

ПЛОВДИВСКИ УНИВЕРСИТЕТ „ПАИСИЙ ХИЛЕНДАРСКИ”
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА
КАТЕДРА „ОБУЧЕНИЕ ПО МАТЕМАТИКА, ИНФОРМАТИКА И
ИНФОРМАЦИОННИ ТЕХНОЛОГИИ”

НЯКОИ МЕТОДИЧНИ АСПЕКТИ НА СЪЗДАВАНЕ НА
ИНТЕРАКТИВЕН ИНТЕРФЕЙС И
ПРЕПОДАВАНЕТО МУ В УЧИЛИЩЕ

ДЕТЕЛИНА СЕРГЕЕВА МИЛКОТЕВА

А В Т О Р Е Ф Е Р А Т

на дисертация за присъждане на
образователна и научна степен

ДОКТОР
по докторска програма Методика на обучението по Информатика и
информационни технологии

Научни ръководители:

проф. д-р Коста Гъров

доц. д-р Стефка Анева

Пловдив, 2022



Ключови думи: *Интерактивен интерфейс, интерактивен уеб подход, интерактивност, методични аспекти и принципи, интерактивна презентация, интерактивно приложение, уебразработчик, интерактивни сайтове, интерактивна среда, интерактивен информационен център, структурно стабилен, визуално привлекателен, атрактивни входящи точки, динамика, уебинтерфейс, рендираща машина, независими броячи, аналогични цветове, хексадецимален код, декоративните шрифтове, бяло пространство, интерактивни банери, итеративен дизайн, интерактивни елементи, дигитални интерфейси, интерактивно съдържание, уебтехнологии, интелигентен уебпроект,...*

ВЪВЕДЕНИЕ

Настоящата разработка е предназначена в помощ на учителите и учениците по Информационни технологии, в модул „Уебдизайн“ – дванадесети клас, които в процеса на създаване на интерактивен интерфейс, изграждат нови дигитални компетентности в използването на най-новите софтуерни технологии. Разработена е под формата на ръководство с насоки за планиране, проектиране, създаване на графичен дизайн – софтуер за създаване и избор на цветови схеми, изгаждане, тестване, публикуване на уеб сайт и сигурност в уеб. Предложен е вариант на създаване на уеб сайт, чрез използване на системи за управление на уебсъдържание, както и повишаване на сигурността на сайта, тестването му, създаване на резервно копие на сайта, актуализация, поддръжка, SEO оптимизация, ориентиране към „отзивчив дизайн „и др.

През последните години в средните училища в България се формираха паралелки за профилирано обучение по Информатика и Информационни технологии, в които се преподава програмиране, както и създаване на интерактивен интерфейс, предимно в системата на www. Това налага учителите да съставят сами учебни програми, по които да водят определени дисциплини в областта на информационните технологии, икономически и други подобни. Поради този факт, забавна тематика се оказва програмирането на основите на изграждането на уеб приложения като ASP.NET MVC 5, ASP.NET Core MVC с контролери и изгледи с помощта на Visual Studio 2017 и др. А главно в тази разработка акцентът е върху технологиите – HTML, DHTML, XHTML, CSS, PHP, FLASH, Java, JavaScript, Java applets, plugins, и програмите CorelDraw, Potoshop, Illustrator, Системите за управление на съдържание и др., които правят изграждането на уеб потребителския интерактивен интерфейс по-лесен и по-продуктивен, а най-вече **забавен за учениците**.

Една от най-важните задачи е разработването на навигационната система на информационната презентация, която ориентира и води посетителите през нея. Чрез създаването **интерактивна среда**, чрез хубави графики или други визуални и музикални елементи, лого – връщачо в началната страница, избора на стилове, шрифтове, цветове, движещи се текстове и др. интерактивни елементи се постига сайт, който е атрактивен – **интерактивен**. В никакъв случай обаче не бива да се прекалява и с интерактивността, всичко трябва да е умерено и „прескачането“ от страница в страница не бива да създава впечатление, че се отива извън тази интерактивна платформа, т. е. създават се идентични, вече избрани по динамика, цвят и ефекти страници за цялостния **интерактивен проект, като едно дигитално списание**.

Тук се фокусира към учениците, и тези които желаят да създават графични и информационни интерфейси, да се занимават с уебдизайн от самото му проектиране до цялостното завършване на проекта, надграждане и визуално оформление по тяхно виждане и усет, с използване на най-различни съвременни технологии, спазвайки някои методични аспекти и принципи за изграждането на интерактивния интерфейс. Пред аудитория представям и идея за допълнителна подготовка и подпомагане усвояването на знания и умения у учениците в обучението по Информационни технологии, модул „Уеб дизайн“ 12-ти клас, като учебно помагало под формата на **Наръчник по Webdesign – външно тяло към дисертацията**. В процеса на обучение учениците ще изграждат нови дигитални компетентности с използване на съвременни техники и софтуерни технологии. Съобразен е с учебната програма на модула, включени са възлови теми като фрагменти от учебното съдържание, които детайлно се разискват . Дала съм извадка от тематиката, синтезирани съгласно Учебната програма в

Съдържанието на наръчника като ЕТАПИ – алгоритъм. Представени са основните аспекти, принципи, стандарти и технологии, необходими за изграждането на уебсайт. В приложението може да видите Учебни програми на групи по Уебдизайн.

Ключово по темите се акцентира върху практически насоки в планирането, проектирането, изграждането, тестването, публикуването, оптимизирането и сигурността на уебсайта. Всяка една от темите дава възможност на учениците да се „разгърнат“. Наръчника го изградих на базата на опита ми в групи WEBDESIGN - занимания по интереси тази учебна година в СУ „Софроний Врачански“, както и с ученици от училищата НХГ „Цанко Лавренов“, МГ „Акад. Кирил Попов“... , в групите ми по „WEBDESIGN“ по проект „Образование за утрешния ден“ 2020 имах паралелки с насоченост по Уебдизайн и обучавах учениците да изградят Уеб сайтове. Над 15-сет години работих в тази посока с различни възрастови групи, което породило идеята ми за него. Предишните ми две статии „*Някои методични аспекти на създаване на интерактивен интерфейс и преподаването му в училище*“ и „*Избрани софтуерни технологии за изграждане на интерактивен интерфейс*“, както и *Наръчникът* са свързани с темата на моята дисертация. В Наръчника наблягам на ключови моменти при изграждането на уебсайт, подкрепено с практически примери. Давам информация за технологиите като основа за надграждане и начините за усъвършенстване. Той би бил по-добър от класически учебник, тъй като е по-синтезиран, изчистен от към излишна информация и алгоритмизиран. Може непрекъснато да се надгражда, актуализира и обогатява. Сигурността в уеб е важна и може да видите повече информация в следваща ми статия: „*Повишаване нивото на мрежова и информационна сигурност*“.

Наръчникът може да се ползва от начинаещи и от напреднали. Целта на модела на обучение в него е да представи една примерна идея за подпомагане на всеки, който иска да се занимава с Уеб дизайн. (Този наръчник продиктува идеята ми и за „*Наръчник на учителя*“ – последната ми статия.)

В наръчника са включени етапите ОТ планиране, проектиране, създаване на графичен дизайн (посочен е софтуер за създаване и избор на цветови схеми – Just Color Pichker, който може да се свали от сайта <https://annystudio.com/software/colorpicker/#download>), **избор на шрифтове**, (https://drive.google.com/file/d/0B4z_Pjf1EcbgdE9YdDBMQXM3WUU/view?resourcekey=0-EJ30MHAqFDM_uq6a2zFy0Q), **използване на сайта** <https://www.w3schools.com/html/default.asp>, **изработване на графични компоненти въз основа на изработения схематичен дизайн - wireframe, вграждане на анимация, мултимедия, изграждане, тестване, публикуване и оптимизация ДО сигурността му.**

За целта може да се използват онлайн инструменти за изработка на диаграми – www.nicepage.com, www.diagrams.net, MS PowerPoint или **Inkscape**. Други софтуерни технологии като SeaMonkey, Java Script, XML, Sublime, Notepad++, WinSCP, FileZilla, HTML редактора BlueGriffon (<http://www.bluegriffon.org/#download>), безплатни шаблони HTML - www.free-css.com, www.wix.com, технологиите <https://Nicepage.com>, Apache or nginx, PHP, MySQL спомагат за изграждането на уебсайт, чрез използване на система за управление на уеб съдържание, както и повишаване на сигурността на сайта, тестването му, актуализация, поддръжка, SEO оптимизация, популяризация, отзивчив дизайн и др. Посочват се удачни съвременни технологии, които постепенно могат да се надграждат. Споменават се програмите за векторна графика, които спомагат за

изработката на професионално лого или цялостна визия на сайта... , вграждане на мултимедия, навигационно меню, движещ се текст – хоризонтално, вертикално и други форми за „раздвижване“.

Цялостния ми дългогодишен опит в изграждането на интарективен интерфейс, помогна за реализирането на тази дисертация.

Уважаеми ученици, учители, тъй като темата е доста обширна и насочена предимно към Вас, но и към всички желаещи да се занимават с уеб дизайн – ще използвам общото название, обръщение – **Уебдизайнер**.

Основна характеристика в работата и изграждането на уебсайт е балансът – добре прецененият компромис между изискванията и възможностите. Може би най-важният момент при изграждането на сайта са тестовете, проведени най-малко на два различни браузъра.

Друг важен методичен аспект, е спазването на основни принципи. Тук именно влизат в действие Великата Дидактика, според която през последните години възможностите на информационните и комуникационните технологии постоянно се увеличават. Те проникват във всички сфери на човешката дейност, в това число и в сферата на образованието. На учителите им се налага често в педагогическата си практика сами да подготвят дидактически материали, които да използват в обучението. Съществуващите разработки у нас не разглеждат специално проблеми като подготовка на дидактически материали с помощта на новите технологии, формулиране на изисквания (правила) за създаване и използване на съвременните дидактически материали. В съвременното българско училище най-често използваните дидактически материали са: печатни материали, аудио записи, видео записи, мултимедийни презентации, електронни учебници, ресурси и помагала.

Важна цел за съдържанието е то да бъде последователно и всички пътища да са завършени и ясно маркирани. Метафората, която се използва при дизайна на един информационен интерактивен интерфейс, не само оказва влияние върху индивидуалността на проекта, но също така определя какъв ще бъде достъпът до информацията и дали потребителят ще бъде въввлечен в процеса. Информацията не може да бъде оставена просто да се носи по екрана. Тя трябва да се постави в контекст, да се рамкира в прозорци и панели, които са логически и ясно свързани с една основна мрежа. Изображенията, анимацията, видеото, музиката, интерактивните бутони, са основните елементи за истинска интерактивна презентация. Трябва да се внимава с визуалното претоварване (с текст, изображения и др.), защото се счита, че това е тихият убиец на интерактивността.

Да проектираш интерфейс, означава да създаваш настроение и преживяване. Дизайнът включва обмисляне на всички елементи, свързани със структурата на съдържанието, външния вид на интерфейса, начина му на действие, употребата му и посланието, което той предава. Предизвикателство на визията е да познавате толкова добре своята публика, че да можете да задържите нейното внимание и да я направите съпричастна. Готовият софтуер има много предимства пред написания по поръчка, но проблемът е, че трябва да разчитате само на вградените от автора възможности – WordPress, Joomla, Drupal, Plone и други системи за управление на съдържание.

Добре организиранят и визуално привлекателен интерфейс, дава лесен достъп до голямо количество информация, използвайки хипертекст, илюстрации, снимки, анимация, звук, видео, връзки с Интернет, така се създава се интерактивна среда, която приканва към изследване и чрез изследването – генерира знания. Внимателно се избира съдържанието, за да се постигне комуникативен ефект. Необходимо е създаване

на усещане за пространство в интерфейса, така се трансформира плоската повърхност на екрана в достоверно изглеждащо дълбоко пространство, което привлича хората и предава реализъм на преживяването им. Използването на сполучливи дизайнерски подходи и стабилни техники за оформление, за онагледяване и движение в страниците е от съществено значение. Не по-малко важни са и използваните шрифтове, както и осъвременяването и актуализацията на сайта. Един от реалните критерии за успех на един уеб сайт е повтарящият се трафик от верни потребители. Четири са основните причини потребителите да се връщат на някои уеб сайтове, а на други – не. Тези четири критерия лежат в основата на добрия уеб дизайн, тъй като това са неща, които потребителите искат най-много да видят. Те могат да бъдат резюмирани накратко чрез акронима НОМЕ (формиран от първите букви на четирите качества):

- ✓ висококачествено съдържание (High-quality);
- ✓ често обновяване (Often);
- ✓ минимално време за изтегляне на ресурси (Minimal);
- ✓ удобно ползване (Ease).

А включването на богат набор ключови думи и честотата на употребата им в текстовете са едни от най-важните елементи за доброто индексване на сайта в търсачките по естествен път. Следва непрекъснатата поддръжка и актуализация.

Ролята на уеб дизайнерите на съвременния интерактивен интерфейс са последните в дългата редица от художници и майстори, които улавят философското, идейното и физическото и го трансформират в комуникационни пакети, които вдъхновяват, учат, или продават на аудиторията. Дизайнерите „пакетират“ мисли и информация, така че да могат да бъдат транспортирани, споделяни, обмисляни, дискутирани и проигравани. Те не работят във вакуум, когато сглобяват уебсайтове. И така, вместо да осигури рецепта от типа:

“click – and - drag” /натисни и провлачи/ за използване на множеството програми, които обикновено са нужни за подготвяне на една уеб страница или друг интерактивен проект, тук е необходимо да се фокусира или акцентира във интерфейс дизайна, а технологиите винаги допълват комуникативните цели на уеб страницата.

Тази разработка е насочена към учениците в часовете по Уебдизайн, които желаят да създават графични и информационни интерфейси, да се занимават с уебдизайн от самото му проектиране до цялостното завършване на проекта, надграждане и визуално оформление по тяхно виждане и усет, с използване на най-различни съвременни технологии, спазвайки някои методични аспекти и принципи за изграждането на интерактивния интерфейс.

За онези, които днес разучават Мрежата, Интернет представлява едно чудо – място, където има всякаква информация, забавления, игри, текстове, изпълнени с картинки, звук и анимация. Shockwave е нещо нормално, електронната поща е необходимост, скоростта на връзката трябва да е висока, дизайнът на уеб страниците – атрактивен или както вече е актуално **ИНТЕРАКТИВЕН**, навигацията – лесна, намирането на информация – мигновено. Всички тези изисквания на потребителите на практика са изисквания към уеб дизайнерите – учениците, които изграждат уеб съдържанието в Мрежата.

I. ОБЩА ХАРАКТЕРИСТИКА НА ДИСЕРТАЦИОННИЯ ТРУД

Натрупаният опит и променящата се среда около нас породиха идеята използването на ИКТ (информационни и комуникационни технологии) в училище да става по нов начин. Предложеният учебен подход има за цел да изгражда стил на интегриране в процеса на преподаване на дисциплината Уебдизайн съчетан с проверки за ефективност. В днешно време преподавателят не е само преподавател, а *ментор, оратор, вдъхновител*.

Съвременният преподавател трябва да овладее уменията чрез непрекъснат „експеримент“ да съчетава компетентното използване на съвременни софтуерни технологии с методите и средствата за автоматизация на информационните дейности. Новият стил се базира на изработени правила, опит и фундаментални знания. Той дава яснота, точност и организираност в използване на софтуерния и хардуерния ресурс, с който се работи. Стилът на интегриране, който се разглежда в разработката, способства за изграждането на ясни стратегии и креативност в процеса на работа с информация, представянето ѝ по подходящ динамичен, атрактивен, интерактивен начин и насочва към избиране на най-резултатните пътища в тази дейност чрез тяхното анализиране.

Бъдещите висококачествени научни изследвания и иновации зависят от това, знанието и уменията да се поставят предварително на солидна основа и да се насърчат към творческо, новаторско и предприемаческо мислене и използване на дигитални знания умения и компетенции.

Обучението е динамичен процес, при който непрекъснато се променят както човешкият ресурс, научните достижения, техническите средства и методи на преподаване, така и средата, в която ще се прилагат получените знания. Образованието е съвкупност от знания, получени при целенасочено обучение.

Изброените промени налагат преосмисляне на изучавания материал в часовете по ИТ в частност по Уебдизайн и търсене на начини за сравнение с практическата му приложимост в конкретните сфери на трудова реализация за съответните специалисти. Изследователският интерес на разработката е породен от факта, че информационните технологии могат да се използват като обвързващи звена в дейности с различна насоченост.

Цели и задачи на дисертационното изследване:

Целта на настоящата дисертация е, да се изследват възможностите за създаване на интеграционна насоченост на учебния процес за дисциплината Уебдизайн. Има се предвид интердисциплинарно обвързване с други дисциплини, които се изучават в училище и интегриране с предприятия, фирми и свободни професии, за да се предложи ефективно, методически обосновано решение за откриване на оптималния стил за преподаване в учебните групи по Уебдизайн, като се създадат условия за използване на съвременните компютърните технологии с лекота и компетентност. Задачата е да се засили интересът към уебтехнологиите, да се създадат условия учениците да „разгърнат“ своите възможности и да изградят добри интерактивни презентации под различни форми, по-конкретно като уебсайтове със собствени стилове, визии и интерактивни елементи, да използват всякакви продукти, приложения, професионални графични програми, плъгини, програмни езици и най-вече да развият своите дигитални компетенции.

Основните изследователски задачи са:

1. Организира се методически „експеримент“ за преподаване на учебните модули по дисциплината Уебдизайн, изградени от учебни единици, които позволяват гъвкавост и структурираност на преподавания материал.
2. Провежда се анализиране състава на ученическите групи, определяне нивата на владеене на програмни езици, графични програми, системи за управление на уебсъдържание, други платформи и технологии, преценка за предварителните компютърни знания и умения на обучаваните. Трябва да се определят основните характеристики и принципи, които съдействат за по-високо качество в обучението на всяка учебна група от ученици.
3. Извършва се дейност по актуализиране и надграждане на дигиталните компетенции и мислене на учениците от средното образование, като се дават препоръки, насоки, принципи и удачни съвременни технологии, които могат да използват в процеса на изграждане на интерактивната среда.
4. Организира се процес на изследване за откриване и вграждане в учебните единици на теоретичен материал, обвързващ обучението по Уебдизайн с други дисциплини.
5. Прилага се дейност по проучване изходящото ниво на учениците чрез използване на изучените софтуерни технологии (практическо обвързване). Ежегодно се анализират резултатите от получените знания на учениците по основните теми изучавани по Уебдизайн. Получените изводи и препоръки от обучаваните се използват от преподавателя за оптимизация на учебния процес.
6. Организира се процес на изследване на възможностите за обвързване на обучението по Уебдизайн във училище, чрез провеждане на електронни анкети и проучване от учениците в конкретна работна среда.

Като се съчетават различни форми и методи на обучение, в разработката се посочват начини за систематичен контрол на ефективността на учебния процес в часовете по Уебдизайн. Използването на подходящи примери позволява да се избират най-резултатните пътища за изучаване на материала, предвидени в учебната програма, с възможности за индивидуализация и надграждане на обучението.

Обект на изследването е учебно-познавателната и приложна дейност на учениците в училищата: НХГ „Цанко Лавренов“ – Пловдив, СУ „Св. Софроний Врачански“ – гр. Пловдив и др.

Обект на обследването са учениците, обучаващи се по дисциплината Уебдизайн в средните училища. Не се изключват и всички, желаещи да се занимават с тази материя.

Предмет на изследването е ефектът от използване на съвременни софтуерни технологии и методически насоки за създаване на условия за интеграционна насоченост на обучението по дисциплината Уебдизайн за учениците в средните училища. Методическите постановки се основават на базата на предложени насоки за повишаване на качеството на обучението, чрез проектирането на различни модули в учебното съдържание за овладяване на необходимата система от понятия и умения с възможности те да се използват при обучението на учениците по други дисциплини в следващ етап от степен на образование, където компютърните технологии ще облекчат научната и изследователската им дейност, както и ще повиши нивото на тяхната дигитална комуникация и дигитални презентационни умения чрез изграждане на интерактивен интерфейс.

В дисертационния труд съобразно целта, обекта и предмета на изследване се формира следната хипотеза:

Хипотеза

Предполага се, че след проведеното обучение по учебна програма, одобрена от МОН по дисциплината Уебдизайн, учениците, учителите и всички ще ползват с лекота

предложените аспекти за изграждане на интерактивен интерфейс. Разработката има точно това за цел, а именно да е в помощ и допълнителна подкрепа за желаещите да се занимават с изграждането и интерактивни презентации и уебдизайн. Очакваните резултати от учениците, използващи съвременни софтуерни технологии и методически аспекти на създаване на интерактивен интерфейс, както и условия за интеграционна насоченост на обучението по дисциплината Уебдизайн, е да постигнат по-трайно овладяване на учебния материал и да могат успешно да го прилагат и използват в следващ клас или етап от степен на образование, изискващи компютърни умения, в извънучебните си дейности, обвързани с използването на компютри, софтуерни технологии, които дигитални компетенции ще им послужат за реализация във висши училища и за реализиране на пазара на труда.

Теоретичното значение на изследването е в разработената методика за преподаване на дисциплината Уебдизайн, съчетаваща модули за интердисциплинарно обвързване с други учебни дисциплини, които се изучават от учениците и в използваните методи за обвързване на обучението по дисциплината Информационни технологии с компютърните изисквания в конкретни предприятия и фирми. Дава се възможност на учениците ясно и точно да видят използването на компютърни технологии в практиката, с тяхното многообразие в софтуерно и хардуерно отношение, с различна професионална насоченост (бизнес и реклама, електронна търговия, уебдизайн, софтуерно инженерство и др).

Практическото значение на дисертацията може да се класифицира по следния начин:

1. Прилагане на нова методика за обучение по дисциплината Уебдизайн, която е богато запълнена с различни практически задачи за всеки изучаван модул, подпомагащи усвояването на новите знания от учениците, за преговор и самостоятелни разработки – уебсайтове, интерактивни презентации и др.
2. Използване на приложни продукти разработени на MS Word (електронни тестове за контролни, електронни анкети, електронно обобщение на индикаторно оценяване на ученически разработки от самите тях и сравняване с тези на преподавателя и др.). Разработените електронни приложения по естествен начин провокират учениците да навлязат в програмния продукт в дълбочина, за да могат да го ползват и по демонстрирания от преподавателя начин да го прилагат в подобни ситуации.
3. Използването на различни начини за интердисциплинарна обвързаност на Информационни технологии и с други дисциплини. Използваният метод обуславя прилагането и обогатяването на знанията по Уебдизайн непрекъснато, през целия процес на обучение на учениците. Предлага се широк спектър от задачи, които се използват в редица специализиращи дисциплини и изграждат умения за компютърна обработка на входни данни, обработката им и представянето им във вид и стил, който ученикът прецени под формата на дигитална интерактивна презентация.
4. Прилагане на нов метод за изследване на възможностите за обвързване на обучението по Уебдизайн с Информационните технологии, като учениците могат да обогатят знанията си в използването на различни периферни компютърни устройства, на специализирани софтуерни технологии и повишаване на дигиталните им компетенции.
5. Използване на разнообразни, индивидуални задачи, изпълнението на които способства да се изградят умения у учениците за добиване на самостоятелни знания и компетенции за работа с учебна литература. Те трябва да могат умело и задълбочено да се ориентират в учебната литература на хартиен носител или в електронен вид.

СТРУКТУРА И ОБЕМ НА ДИСЕРТАЦИЯТА

Дисертацията се състои от въведение, обща характеристика, три глави, заключение, приложения и списък на използвана литература.

Първа глава включва теоретични основи на изследването, Втора глава – методически инструментариум за реализация на модела, Трета глава - Описание на педагогическото изследване, обработка и анализ на получените данни.

„Чудеса се случват, но човек доста трябва да се потруди за това.“

ГЛАВА ПЪРВА – ОСНОВНИ ТЕОРЕТИЧНИ ИЗСЛЕДВАНИЯ. ОСНОВНИ ЕТАПИ В РАЗРАБОТКАТА НА ИНТЕРАКТИВЕН ИНТЕРФЕЙС

Интерфейсът е тази част от инструмент или технология, с която потребителят комуникира. За отвертката, това е дръжката, за уеб страницата, това е приложената комуникативна среда, в която е поместено съдържанието на страницата и устройствата за навигация, от които потребителят има нужда, за да достигне до съдържанието.

Целта на един интерфейс, е да изясни съдържанието, предложено от страницата и да го „поднесе“ ефективно. Например:

а/ Ако уеб страницата /или друга интерактивна област/ е проектирано като инструмент да позволява на посетителите да получават помощната техническа информация, която търсят или да си купят сезонни билети за влак – интерфейсът в действителност трябва да бъде прозрачен, така че да не пречи на потребителят да получи това, за което е дошъл.

б/ Ако основната цел на страницата е образователна, интерфейсът трябва да бъде достатъчно привлекателен, за да насърчи участието и ученето.

в/ Ако страницата е предназначена да рекламира или да продава продукт или услуга, интерфейсът играе важна роля за улесняването на потенциалните клиенти при получаването на информацията, от която се нуждаят, за да вземат решение.

Освен това той трябва да ги мотивира да пристъпят към действие и да им осигури начин за поръчване online или не. Навигацията е тясно свързана с интерфейса на сайта. Web е система за навигация: основно действие на потребителя е да щраква върху хипер-текстовите връзки, за да се придвижва из огромното информационно пространство от стотици милиони страници. Тъй като това пространство е необятно, навигирането е трудно и се налага да се предоставят на потребителите средства за навигация, които надрасват обикновените хипервръзки от типа „иди там“.

Интерактивния интерфейс предполага да има помощен характер към потребителите, да могат да си отговорят на трите фундаментални въпроса: „къде съм?“, „Къде бях?“, „Къде да отида?“. Вероятно най-важният въпрос при навигацията е „Къде се намирам?“, тъй като потребителят няма шанс да разбере структурата на сайта. Трябва да има начин за връщане в предходна страница и директно в заглавната. Моите проучвания на удобството на ползване показват, че потребителите не се чувстват комфортно в сайт, който се опитва да използва интерфейс за навигация, коренно различаващ се от преобладаващите. В никакъв случай обаче не бива да са идентични

с други, понеже това ще доведе до отегчаване. Вашето лого, вашия идентификатор, трябва да е на всяка уеб страница.

За отговора на въпроса „Къде бях?“ – трябва да се слагат бутони Back. А на въпроса „Къде да отида?“ , най-добре с един клик в логото на сайта да отидете в началото...Препоръчително е използването на технологията CSS за това.

В Web потребителите се движат между сайтовете с висока скорост, а рамките между различните дизайни са размити. Рядко потребител прекарва повече от няколко минути в даден сайт. Именно поради това, те смятат, че използват Web като цяло, не като отделен сайт. Те не искат да четат никакви наръчници и помощна информация. Затова е добре учениците да проектират за съвместимост и със съзнанието, че техния сайт не е център на Вселената на потребителя. Целта на добрия уеб дизайнер е да улесни използването на сайта от потребителите. Трябва да се стреми да намали безредието: постига се чрез съвкупност, резюмиране, филтриране, съкращаване и представяне чрез примери и пр.

Преди всичко, е необходимо уеб-дизайнерът да качи файловете си на сървър – FTP клиент и да прави настройки..Но какво е сървър? - Сървър е всеки компютър, работещ със софтуер, който му позволява да отговаря на запитвания за документи или друг вид данни. Програмите, които дават заявките и показват документите (например браузърите), се наричат клиенти. По отношение на специфични функции като карти с изображения термините „от страна на сървъра“ и „от страна на клиента“, се отнасят до машината, извършваща обработка. Функциите от страна на клиента се изпълняват на потребителския компютър; функциите от страна на сървъра се изпълняват на отдалечената машина. Уеб сървърите отговарят на заявки от браузри (клиентска програма), извикват определения файл и връщат резултатите от документа или скрипта. Уеб браузърите и сървърите общуват помежду си посредством Hypertext Transfer Protocol – протокол за обмен на хипер-текстови документи, за по-кратко наричан още HTTP. Всеки компютър може да бъде сървър, ако на него се стартира софтуер за сървър. В наши дни съществуват множество налични пакети, но лидери си остават Apache и Microsoft Internet Information Server (ISS).

Apache – по настоящем, по-голямата част от сървърите /приблизително 60%/ работят с Apache. Неговата популярност се дължи на факта, че е мощен и притежава много характеристики, и най-вече защото винаги е бил безплатен. Работи предимно на Unix платформи, но бяха създадени версии, обслужващи и други платформи като Windows NT/2000 и Mac OS X. Основната инсталация на Apache е доста ограничена, но лесно може да бъде разширена и персонализирана с помощта на различни модули. Apache извиква съответния модул, за да извърши определена задача, например разпознаване на потребителя или заявки, базирани на данни. Може да изтеглите сървъра Apache, както и неговата документация, от сайта на Apache <http://www.apache.org>.

Internet Information Server (IIS) – Това е продукт на Microsoft, който също е безплатен. Той работи на Windows NT платформи.

Видът на сървъра не влияе върху работата на учениците, например върху създаването на графики или разработването на основните HTML файлове. Той обаче влияе на по-съвременните техники за изграждане на уеб сайтове като Server Side Includes.

Трудно може да си представим съвременния свят без информацията, която получаваме от уебсайтовете. Използваме ги за професионални или домашни цели, за лични нужди, развлечения, за лесна комуникация и т. н. Всеки сайт е съвкупност от страници, по които може да се „скача“. Страниците съдържат различни елементи: текст, изображения, видео, анимация, звук, списъци, таблици, форми, хипервръзки. Много често те се описват на HTML (HyperText Markup Language – език за маркиране

на хипертекст). HTML е създаден през 1989 г. в Церн, Швейцария. Досега има разработени пет версии. Пълните технически спецификации на официалните версии на HTML може да се видят на адрес: <http://www.w3.org/>.

Всяка уебстраница има свой адрес, чрез който е достъпна. Страниците се разглеждат през браузър (Windows Explorer, Mozilla Firefox, Opera, Safari, Google Chrome и др.)

При разработване на сайтове е важно те да бъдат тествани на различни браузъри.

Всеки сайт се нуждае от host – място в сървъра за съхраняване на неговите файлове. Обикновено тези места са платени. Освен хост, нужен е и подходящ домейн-име на сайта.

1. WORD WIDE WEB (Web) е най-популярната услуга в Интернет.

представлява мултимедийна информация под формата на документи, в които се комбинират текст, аудио, видео, графика и анимация. Хипервръзките в тези документи правят възможен достъп до други документи, свързани с тях. Те от своя страна сочат към други, намиращи се на различни сървъри в мрежата, без да имат директна връзка помежду си. По същество се осъществява една паяжина от контекстно-ориентирани връзки. Най-често, уеб сайтът се дефинира като „съвкупност от уеб страници, адресирани на общ URL, който често се състои само от името на домейна или IP адреса и пътя до основната директория (“/”) в мрежа, базирана на протокола IP“. Всеки уеб сайт се публикува на компютър, наречен уеб сървър, който е достъпен през глобална или локална мрежа. Присъствието в уеб пространството в днешно време вече е необходимост и е задължително за всеки, който иска да покаже на света, че съществува, независимо дали е ученик, собственик на малка фирма или става дума за голяма организация.

Процесът на създаване на интерактивен интерфейс, е специфичен и доста различен в зависимост от конкретния проект. Той може да се раздели на няколко основни етапа. Пропускането на някой от тях или слабата работа на специалистите, които отговарят за него, може да доведе до неуспех на целия проект.

В процеса на работа вземат участие различни специалисти, тъй като за изграждането на всеки сайт са необходими знания и умения в няколко различни области – графичен дизайн, мултимедия, програмиране, маркетинг, реклама и мениджмънт. Добрата работа не екипа като цяло е ключов фактор.

По-малките уебпроекти често се изпълняват от неголям екип, като това дава възможност за гъвкавост и експедитивност на процеса и по-нисък бюджет. Задължително условие в този случай е разработчиците да могат да поемат и извършват работа на различни специалисти на ниво. В противен случай се появяват сайтове, които „не звучат“ добре и които нямат голям шанс за успех във все по-силната конкуренция в Мрежата.

Първата стъпка при създаването на уебсайт е дефинирането на цели. Без ясна идея за целите и задачите проектът не би могъл да бъде успешен. Внимателното планиране и следване на отделните етапи от създаването на сайта също са важни условия за неговото добро реализиране.

Преди започване на работата по сайта е необходимо да се определи:

- Какво ще представя – за фирмено представяне, обучение, реклама, развлечение, електронна търговия и т. н.
- Основни цели – да дава обща или конкретна информация, да се увеличават продажбите, да се осъществяват бизнесконтакти и т. н.

- Аудитория – кой ще посещава сайта, каква ще бъде неговата публика (възраст, пол, интереси, езици);
- Основно съдържание – с какви средства ще се постигнат целите на сайта;

След като се постигне яснота по горните въпроси, се определя текстовото съдържание и графичните ресурси (снимки, графики, лого, фирмени елементи и др.) и начина на изпълнение. Следва проектирането на основната структура на сайта и архитектурата на страниците, като се подготвят необходимите блоксхеми и скици. В този момент се извършва изборът на основния дизайн на сайта, създаването на „маска“ и проект, по който ще се работи.

Етап „проектиране“ се разработва съвместно от мениджъра на проекта, маркетинговия специалист, дизайнера, конструктора и клиента.

На базата на подготвените материали и взетите решения при фазата на проектирането се преминава към създаване на страниците от уебсайта, като на този етап едновременно се извършва подготвянето на текстовете и графиките, и тяхното съвместяване. В зависимост от случая се създават и необходимите скриптове и програми, преминава се към детайлно тестване. Това не е единичен цикъл, а се повтаря многократно, докато се постигне нужния резултат.

Тестовите обхващат всички възможни варианти на потребление –

Браузъри и монитори (разделителна способност, цветови настройки). Тестува се скоростта при работа в мрежа, проверяват се текстовете за граматически и смислови грешки, връзките между страниците в сайта и извън него, като е необходимо да се обърне внимание на удобството при работа с него.

В етап „Създаване“ работят едновременно конструкторът, редакторът, дизайнерът, програмистът и разработчикът, ръководени от мениджъра на проекта.

1.1. Първата най-важна и отговорна задача при създаването на един уеб сайт с интерактивен интерфейс:

- Осъзнаване на необходимостта от него;
- След това последователно трябва да се обмислят отговорите на въпросите: Защо трябва да бъде създаден? Кой има нужда от този сайт? Какво ще прави този сайт? С какво ще бъде по-различен от другите сайтове? Какво ще постигне със силата на Интернет?
- Много често уеб сайтовете просто имитират друга медия или други издания, предназначени за печат.
- Създаването на качествен сайт в съвременните условия трябва да пресъздава реалистични цели в добра функционалност, позитивно въздействащ и с атрактивен вид и не на последно място – да използва силата на Интернет в тематичното рекламиране и информизиране на милиони потребители.

1.2. Планирането на уеб сайта е един от съществените етапи в процеса на неговото изграждане и ключов момент за успеха му. Този етап се свързва с изготвянето на продуктов план, който:

Очертава главно предназначението на сайта;

Дефинира целите на сайта;

Определя екипа от хора, които ще участват в разработването на сайта;

Идентифицира целевата аудитория;

- ❖ Анализира съществуващите „конкурентни“ решения;
- ❖ Описва специфичните характеристики на сайта и как те ще работят;

- ❖ Предвижда срокове за изпълнение, приходи и разходи по проекта;
- ❖ Определя екип от хора, които ще участват в популяризирането, поддържането и актуализирането на сайта;
- ❖ Следващият значим етап е проектирането на сайта.

Уеб сайтовете са направени, за да се използват, а не само да се разглеждат. Те трябва да притежава елегантност и стил. На този етап вниманието трябва да се концентрира върху:

- ❖ Обмисляне на посланието на сайта, който се създава;
- ❖ Оформяне на основните теми като заглавия;
- ❖ Определяне на съдържателната част на сайта и подготовка на необходимите материали – текст, изображения, мултимедийни обекти, връзки към източниците на информация;
- ❖ Визуално представяне на структурата на сайта – чрез блок схема;
- ❖ Фокусиране върху графичния дизайн на стила на сайта;

1.3. Предизвикателството към графичния дизайн и основна задача:

Оформлението на началната страница на уеб сайта – Home Page-index.html... На нея трябва да се представи категорично, ясно и точно какъв е сайтът, какво може да се направи на него и на кого принадлежи. Тези послания се дават чрез: името на сайта и неговия URL адрес, кратък пояснителен текст, навигацията и функционалността, подходящи изображения и цветове.

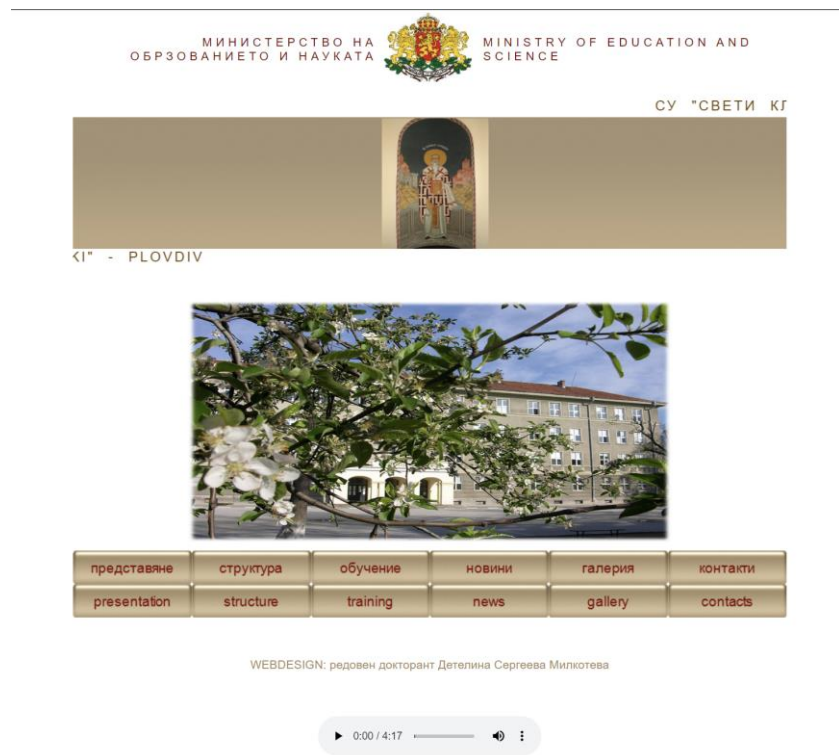
Основната задача на графичния дизайн е да придаде форма на съдържанието, характеристиките и навигацията на сайта. Той подсказва на потребителите кое е най-важното, като поставя по-голяма визуално ударение върху определени елементи от страниците чрез използване на цвят, размер, разположение и анимиране.

Като цяло, уеб сайт представлява повече отколкото сборът на неговите части. Общото впечатление включва в себе си нещо неуловимо, което може да накара потребителя да остане на сайта достатъчно дълго време, да запази адреса му като ценен, да го изпрати на приятел, но може и веднага да го затвори.

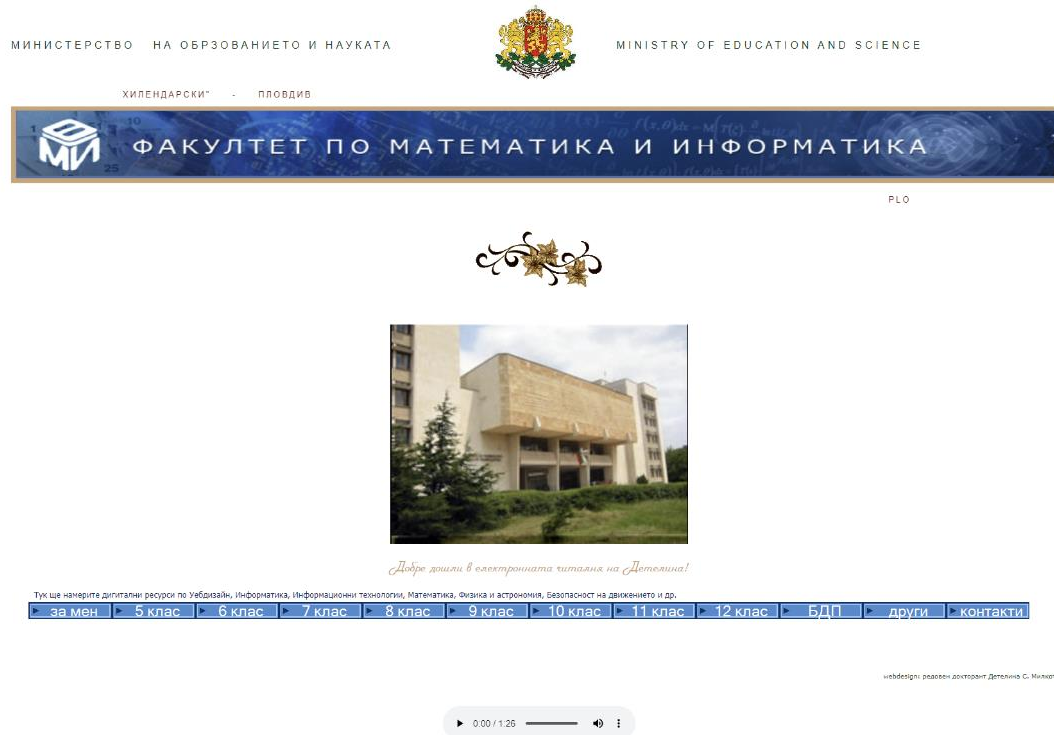
Визията на сайта също е много важна. Централните елементи на всяка страница са логото, основната навигация, допълнителната навигация, различните графични елементи и форми. Началната страница или входящата е най-важната. Тя привлича вниманието, там трябва да присъства името на сайта, връзки към главните подстраници, да дава информация за целта на сайта, да приветства, да дава пълна информация, информацията да се обновява непрекъснато и да накара посетителите да влизат отново и отново. От първата страница трябва да е видимо как ще изглеждат следващите, да е като книга и да не се губи представа къде се намира посетителят при една препратка. Да не се пренебрегват и вътрешните страници, те трябва да задържат вниманието, да следват стила и логиката на началната, при което да са в единство с предходните нива.

Изборът на правилната цветова схема е комбинация от определяне на това какви цветове ще носят правилното послание към потребителите. При изграждане на дизайна, цветовата гама да е съобразена с темата на сайта, да не е натрапчива и отблъскваща. Внимателния и балансиран подбор на цветовете може да създаде великолепна композиция, тъй като цветовете са първото нещо, което се забелязва. За малките елементи се избира по-ярки цветове. Ползвайте повечето от 3 - 4 цвята . Във въведението съм дала и софтуер за графично решение и избор на цветовете.

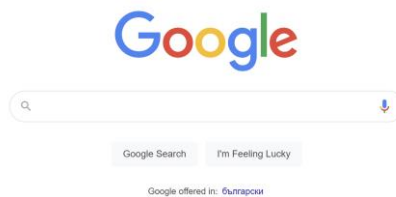
Ето и примери за визия на сайтове, които смятам за изчистени и добре стилизирани –
Примери 1, 2 и 3:



Пример 1: Изчистен от към реклами и други елементи сайт



Пример 2: Изчистен дизайн



Пример 3: Изчистен сайт – изчистените сайтове се правят по-трудно

Софтуерни технологии за избор на шрифт, цветови схеми и други.

За визията на сайта може да се ползват професионални програми графичен дизайн и реклама – CorelDraw, Adobe Illustrator, Adobe Photoshop и др.

За да се изгради уебсайтът е необходимо да се спазват принципите на уебдизайна, които може да видите в статията „ИЗБРАНИ СОФТУЕРНИ ТЕХНОЛОГИИ ЗА ИЗГРАДАНЕ НА ИНТЕРАКТИВЕН ИНТЕРФЕЙС“, вкл. в сборника на СБУ – гр. Пловдив. Освен принципите е нужно да се спазват и уеб стандарти, както и използване на основни уеб технологии. Основният методи за предаване на информация по Мрежата става с хипервръзки. Всяка страница има свой URL адрес, който се състои от име на протокол, име на сървър – домейн, и път до ресурса на сървъра и този URL адрес не се повтаря. Уеб страниците са групирани в уеб сайтове, които са съвкупност от страници, допълващи се функционално, със сходен дизайн и са на един уеб сървър. Всеки сайт представлява папка с файлове, има уникално име по което да е различаван в Интернет.

Организацията на работата и обменът на информация се основават на мрежовата архитектура КЛИЕНТ - СЪРВЪР. От една страна са клиентите – уеб браузъри, които подават заявки и изчакват отговор, а от друга са сървърите – компютри с огромен капацитет от ресурси, които чакат заявки за обработка.

Обменът на хипертекстове – www е най-често използваната услуга. Протоколът е http, който осъществява връзката между клиента и сървъра.

Важно за уеб стандартите е, че те са създадени 70-те години на миналия век, използват се нови технологии, добавят се функционалности, което налага въвеждането на нови стандарти. Уеб страницата може да изглежда напълно различно в различните браузъри, затова възникват УЕБСТАНДАРТИТЕ. Те са нормативни спецификации на технологии методологии за визуализиране на съдържание в уеб среда. Те помагат за структурата, съхраняването и визуализацията на информацията представяна на потребителите. W3C е главната международна организация за стандартизиране. Стандартите се отнасят до това как една страница трябва да бъде проектирана: какви шрифтове и размери да се ползват, кои цветове да се ползват и къде, стил на бутоните, разположение на навигацията, къде се поставя функцията за търсене. Тези стандарти обхващат, както естетическия стил, така и функционалното устройство на сайта.

По принцип кодът може да се напише със всеки текстов редактор, но има специализирани, които улесняват работата и са с по-добра функционалност. Те са текстови и визуални. Текстовите са за ръчно въвеждане на HTML код – Notepad++, Bluefish Editor, UltraEdit, EmEditor. Тези редактори познават синтаксиса и структурата на HTML и свързаните с него други езици. Визуалните реализират технологията

WYSIWYG – каквото виждате, това получавате. Авторът вижда резултатът от действията си. Това дава възможност на уеб дизайнера да вижда какво проектира. Трик в работата е използването на шаблони – предварително създадени уеб страници или части от нея. Те могат да се използват многократно, което улеснява процеса по създаване на уебсайта. Те са лесни за модифициране и спестяват време, разходи. Редактират се с визуални редактори, има шаблони на сайтове за системи за управление на съдържание.

Задължително се зачитат авторските права на източниците на информация, като шаблони и др. уеб елементи, информация и пр. В тази връзка ще посоча Общия публичен лиценз на ГНУ – GNU, лиценз на свободен софтуер, по който авторът предава програмното обезпечение в обществена собственост, като гарантира на потребителите на компютърни програми определени права да ползват програмата за каквато и да е цел, да изучават как работи и да я променят, да отразяват и разпространяват нейни копия, да подобряват програмата и да дават на обществото достъп до подобренията.

В основата на изграждането със съвременните технологии, които изброих по-горе, си остава HTML – hypertext markup language /стандартен маркиращ език за създаване на документи, които да се визуализират в уеб браузър/. HTML елемент е основен структурен компонент в документа. Описва се отварящ и затварящ таг и включва цялото съдържание между тях. А таг е етикет, заграден в ъглови скоби (< >). Таговете указват на браузъра как да интерпретира информацията, описана в уеб документа, като определя началото и края на описанието на елементите, съдържанието, вид и атрибути. Атрибутът задава основни или допълнителни характеристики на HTML елемента. Описва се в отварящ таг, най-често като двойка атрибут = стойност.

Всеки таг има конкретно предназначение – има на адрес www.w3schools.com/tags/.

1.4. Основни принципи в графичния дизайн:

- ❖ Използване на предварително подготвени шаблони, за да следват страниците сходна структура;
- ❖ Използване на невидима решетка, върху която да се планира точното местоположение на всеки елемент от структурата на страницата, така че лесно да може да се обходи с поглед;
- ❖ Внимателен подбор на цветовете, съобразен с асоциациите и чувствата, които те могат да породят;
- ❖ Съобразяване с типографията във виртуалното пространство относно избор на: видове шрифтове – несерифни (Arial, Calibri, Tahoma, Verdana), които потребителите имат системно инсталирани на техните компютри. В противен случай сайтът няма да изглежда по начина, по който е проектиран.
- ❖ Размер на шрифта – от 8 до 12 пункта;
- ❖ Стил на шрифта – нормален /малките по размер символи изглеждат като неясни петна и трудно се четат от екран, ако са получер или стил курсив/;
- ❖ **Цвят на шрифта** – контрастиращ на фона на страницата /най-често тъмен текст на светъл фон/;
- ❖ Дължината на реда – трябва да бъде приблизително равен на дължината на две азбуки – 52 символа. Прекалено дългите редове текст разсейват читателите, ако не могат да намерят къде е началото на следващия ред, а прекалено късите ги отегчават.
- ❖ Отстъпи между абзаците – поне един ред, за да може текстът „да диша“.

Индивидуалната работа по изграждане на уеб сайт на конкретна тема изисква определено време и зависи от познаването и владенето на определена технология или специализиран софтуер за програмната му реализация.

1.5. Ролята на уебдизайнера – ученика.

И така, вместо да осигури рецепта от типа: “click – and - drag” /натисни и провлачи/ за използване на множеството програми, които обикновено са нужни за подготвяне на една уеб страница, CD ROM или друг интерактивен проект, тук е необходимо да се фокусира или акцентира във интерфейс дизайна, а технологиите винаги допълват комуникативните цели на уеб страницата.

ВНИМАНИЕ!

Всеки проект включва екранна диаграма с придружаваща я легенда за най-съществените компоненти на интерфейса.



Фиг. 1 Екранна диаграма на уебстраница

1.6. Концепция - също, както в печатната медия, първият етап на проекта за уеб дизайн е разработването на концепция.

В началния стадий на проекта, трябва да си отговорите на няколко основни въпроси:

- ❖ За какво е нужен този уеб сайт?
- ❖ С кого ще трябва да контактува?
- ❖ Как комуникацията ще се извърши най-ефективно?
- ❖ Какво ще вземат със себе си посетителите на сайта?
- ❖ Кога ще бъде завършен уебсайтът?
- ❖ Моментът подходящ ли е да се впускате в голям **интерактивен проект**?

Щом веднъж сте си отговорили на тези въпроси, вече сте близо до решението.

1.7. Дизайн – По време на дизайнерската фаза, ще изберете това каквото сте решили по въпросите коя е аудиторията, как най-добре да общувате с нея и с какво и с какво искате да си тръгне. Това е и времето за съставяне на план за организация на сайта и за изчисляване на управлението му от посетителите. Различните истории и прототипи могат да ви помогнат да се чувствате сигурни, че идеите Ви ще проработят. Няма значение колко добра е концепцията, с която излизате или колко добри технологии използвате за поддръжката му: ако посетителите не могат да си представят как да достигнат до това, което искат, те скоро ще се отегчат или разочароват.

1.8. Производство - Производствената фаза включва сглобяване на текста, изображенията, аудио и навигационните елементи, които сте планирали през дизайнерската фаза и интегрирането им в работна система.

С други думи: „изграждането на интерактивен интерфейс“ е конструиране на **интерактивна среда**, инсталиране на атракциите, складиране на запасите, поставяне на знаци за указване посоката и включване на светлините. От това колко добре ще се справите с производствената фаза, зависи дали вашият сайт ще се превърне в успешен, добре приет **интерактивен информационен център** или ще бъде просто още една спирка по пътя към нещо друго.

1.9. Маркетинг - Ако искате хората да дойдат на **виртуалния информационен концерт**, който вие сте проектирали и произвели в лицето на Вашият сайт, ще трябва да ги уведоиме, че той съществува. Вие може да имате най-красивия, най-красноречивия, динамичния, най-образователния, практическия или забавния център в мрежата, но ако никой не знае, че той е там, ефектът от трудът се губи.

Актуализирането и рекламирането е жизнено важно за всеки web проект.

1.10. Съдържание на сайта - Съобщението не е единственото съдържание за уеб сайта. Посетителите в мрежата безспорно не са пленници. Те са свободни във всеки един момент да напуснат Вашият сайт и никога да не се върнат. Ето защо, част от съдържанието на уеб сайта трябва да бъде предназначено да дава на хората допълнителна ценност, която те търсят, за да ги задържим в сайта достатъчно дълго, така че да разберат съобщението. Или, за да ги накараме да се връщат отново в бъдеще... За това можете да подсилвате и подновявате съобщението, защото посетителите искат информация, която могат да прилагат, инструменти, които могат да ползват, красота на която да се наслаждават, игри, които могат да играят, информация, която могат да свалят безплатно или връзки към други интересни за посетителите страници.

1.11. Технологиата не е съдържанието - Като се има предвид нуждата от допълнителна ценност и постоянно променящите си технологии на Интернет, може да бъде доста изкушаващо да украсите уеб сайта с всичките си най-нови технологични постижения. Но ако са твърде много, те могат да утежняват съдържанието. Не забравяйте, че нещо което добавяте в съдържанието не бива да разводнява или отнема от основната цел на сайта.

Всяко динамично взаимодействие, което планирате за сайта, винаги трябва да поддържа и подсилва основното съдържание. Следователно технологиата е добра, когато прави информацията по-достъпна, по-ангажираща, по-запомняща се.

1.12. Възможности на технологиата - движението, звукът, взаимодействието и стандартните добри статични графики могат да помогнат да привлечете посетителите, да спечелите място в паметта им, да ги накарате да се върнат отново и да я ползват. Но при сменящи се с шеметни скорости Web технологии /напредването на компютърните възможности, повишаването скоростта на браузърите, plug-in манията и новите анимационни техники/ избирането на най-надеждно средство за създаване на достъпно съобщение е чист трик.

1.13. Съобразяване на технологиата с аудиторията - трябва да имате предвид вашата целева аудитория, когато избирате специфичните видове медии /текст, анимация, аудио

и видео, виртуална реалност и др./ и технологиите за тяхното доставяне /напр. работа с таблици, допълнителни plugins, Java; база данни за търсене и т.н./.

Ако продуктът е High-tech и аудиторията се състои от напреднали в технологията, ще бъде по-подходящо използването на последните online приспособления. Технологията може да бъде част от привлекателността на сайта, като посетителите й вероятно ще имат бързи компютри и plug-ins, за да работят правилно с нея.

Но, ако Вашият продукт е насочен към по-широк аудитория, използването на най-нови технологии може да изолира Вашия сайт от посетителите, които не са добре подготвени за да могат да я използват.

Това може да подразни голяма част от аудиторията, защото с техниката, която ползват, сайтът ще бъде муден, ненадежден, смущаващ.

И така: Защо да използвате shockwave анимация, работеща автоматично с повечето обикновено използвани браузъри?

Много по-вероятно е да си навлечете неприятности, ако използвате твърде комплексна технология, отколкото твърде опростена.

1.14. Създаване на единна метафора индивидуалност и стил - дизайнът на един интерфейс зависи от неговата аудитория и от неговата цел: например да образова, да представя или продава. След като веднъж е разработена концепцията на Web сайта, време е да се проектира „пултът“, „контролното табло“ или „инструмент панел“, който ще използвате, за да доставите съдържанието на сайта на неговите посетители. При това е важно да разберете метафора, индивидуалност или стил, който да обединява съдържанието и да подкрепя и изяснява комуникацията. Вашата цел е да създадете възможно най-директен и полезен, най-ангажиращ и запомнящ се **интерактивен интерфейс**.

1.15 . Избиране на метафора - един от най-ефективните начини на комуникация е използването на познатото, за да се обясни непознатото. И един начин да се направи това е използването на метафора – фигура на речта или изцяло конструирана среда, в която за едно нещо се говори като за друго или се представя като друго. Например описанието в началото на сайта, вместо буквално описание, да представлява метафора. Тя показва как интерактивните компоненти на един on-screen интерфейс трябва да работят успешно. Друг пример: Думата Web /мрежа/, само по себе си е метафора за мрежата от свързани инструкции, личности и информация. Ако дизайнът на Вашият сайт използва метафора, той ще трябва да осигури някакъв вид физическа реалност за духовното пространство, което посетителите обитават, докато са там.

Метафората се използва твърде много в web дизайна. Вероятно най-големият недостатък на метафората е, че подтикват учениците да бъдат изобретателни и да поставят сайта в насока, която изглежда забавна и подходяща за метафората, но изоставя на заден план истинските цели на потребителя. Потребителите не живеят в метафоричен свят – те живеят в истински свят. При такава ситуация обикновено е по-добре да бъдете съвсем буквални и да описвате какво представлява всеки от елементите на интерфейса и какво прави, вместо да се опитвате да съвместите всички елементи с една метафора. Имайки предвид казаното, метафората може да бъде полезна поради две причини. Първо, метафората може да предостави унифицираща структура за дизайна, чрез която ще усети, че той е нещо повече от колекция от отделни теми. Второ,

метафората може да улесни придобиването на познания, като позволи на потребителите да изградят допълнителни знания върху вече познатото от еталонната система, на която се основа метафората.

1.16. Индивидуалност - Ако не използвате физическа метафора за обединяването на сайта, той ще има нужда от своя уникална идентичност и индивидуалност. Какъв „привкус“ ще има доставянето на вашата информация – забавен или официален?!

Два Web сайта, разработени с много сходно оформление, но с различна индивидуалност, могат да осигурят съвсем различно преживяване. И отново, бъдете последователни.

Ако главните части от сайта трябва да представят доста сериозна информация, шеговитата индивидуалност, подходяща за някои части от сайта, трябва да бъде изхвърлена в полза на по-официална. Такава, която да няма нужда да се изменя с цел да се избегне объркване на съобщението, когато посетителят стигне до главната информация.

1.17. Стил - Друго нещо, допринасящо за вкуса на интерфейс дизайна на Web сайта, е неговият стил: разположението му, графики, шрифтове, цветове и т.н.

Разположението на сайта трябва да бъде достатъчно гъвкаво, за да понася промените в съдържанието и различните типове медия. Това изисква една скрита мрежа, която най-малкото да определя зоните за идентификация на сайта или на компанията, навигационните елементи, текста и образите. Така например не центрирайте едни елементи, а да подравнявате други в ляво. По принцип подравняването в ляво или в дясно поддържа по-добре мрежата и се чете по-добре от центрирането и за това е първоначално зададеният по-добър избор за дизайн. Един от начините за отличаване на интерфейса е чрез измеренията на неговите графики: 2D, 2 ½ D или 3D. Стилът 2D е подобен на печатната страница, с нейното подчертаване на структурната организация на графика и текст върху равна повърхност.

Пример за 2 ½ D е подвижният бутон – навигационен елемент, който се поставя релефно и с възможност за натискане. Като използвате този подход, вие може лесно да покажете кои части от интерфейса са интерактивни.

3D стилът очертава виртуално триизмерно пространство, в което за навигация и информационен достъп се използват образи и обекти от реалния свят. Този стил може да включва всичко, от илюстрации, имитиращи перспектива, до виртуална реалност *render-on-the-fly /VRML/*, която позволява на посетителя да се движи свободно из цялата среда. Но не е необходимо метафората и стилът да съвпадат перфектно.

Нужно е уеднаквяване на дизайна чрез набор от стилове. Всички набори от стилове за всеки отделен сайт трябва да бъдат проектирани от една централна дизайнерска група. За това има две причини. Първо – централизираният дизайн е единственият начин да се осигури последователен стил и да се извлече едно от основните предимства на наборите от стилове. Второ, по-голямата част от създателите на web съдържание няма да са в състояние да напишат добри набори от стилове. Опитът с текстови програми, които поддържат набори от стилове, показва, че повечето автори се справят ужасно със своите набори от стилове. Разбирането на ефекта от стила е относително лесно в традиционното настолно издаване, тъй като то е в WYSIWYG1 среда с единна, канонична изходяща форма. Web обаче не е от типа WYSIWYG, поради разнообразието от платформи. Нещо повече, web наборите от стилове са каскадни – CSS, което означава, че наборите от стилове на сайта се сливат с наборите от стилове на потребителя, за да пресъздадат крайно представяне. Поради тези различия е от

съществено значение наборите от стилове да работят, т.е. страниците трябва да продължат да работят дори когато наборите от стилове не могат да се използват от потребителя или от неговия браузър. В никакъв случай не използвайте например трикове, при които повтаряте една и съща дума множество пъти с малко отместване, за да създадете ефект на сянка; без замисления стил текстът се превръща в нещо неразбираемо. Поддържането на добро представяне и без набори от стилове е необходимо за потребителите с по-стари браузъри, за аудитория със зрителни проблеми и за тези, които са забранили използването на набори от стилове в своите браузъри (поради бъгване или просто защото наборите от стилове влизат в конфликт с техните предпочитания). Това правило лесно може да се провери, само като се изключат наборите от стилове от своя браузър и се заредят отново. Важно още при работа със набори от стилове, е да имате предвид още няколко неща:

- не използвайте повече от два шрифта, нито абсолютни размери на шрифта. Вместо това задайте целия текст в съотношение спрямо размера на шрифта по подразбиране, посочен в предпочитаните настройки на потребителя;
- CSS дава възможност в наборите от стилове да се посочи, че определени настройки могат да надделяват над тези от други нива на каскадата.
- Ако имате няколко набора от стилове се уверете, че във всички тях използвате еднакви CLASS имена за една и съща концепция.

1.18. Създаване на план за организация и навигация - за повечето хора предизвикателство от вида: „Какво ще се случи, ако кликна върху това?!“, вече се е изтъркало. Организираните, интуитивните и достъпните ще спечелят пред разпилените, мистериозните и тайнствените почти всеки път.

Съществено е да представите графично структурата и организацията на сайта, независимо дали вертикално организирани като дървовидна структура – вертикална или хоризонтална, или представени радиално като главина на колело.

Картата на сайта трябва да показва нагледно къде всеки елемент е във връзка с другите елементи. И трябва да бъде активна. Всички видими on-screen трябва да бъдат привлекателни и с възможност за кликуване върху навигационните устройства.

Изработката и организирането на потенциалните компоненти на интерфейса, Ви помагат да се уверите, че имате цялата информация и всички навигационни инструменти, от които имате нужда, за да представите желаното съдържание по възможно най-ясния начин. Това, което е важно, не е медията, а посланието.

Съществуват много начини за визуализация на възможните категории и информационните връзки на интерактивния проект, от грубата скица до облепената с бележки стена. Главната цел е да сте сигурни, че съдържанието е последователно и че всички пътища са завършени и ясно маркирани.

За интерактивния информационен дизайн, интерфейс компонентите и технологиите зад тях са платното, боите и рамките на картината, които се използват за посрещане на комуникативното предизвикателство. Метафората, която се използва при дизайна на един интерфейс, не само оказва влияние върху индивидуалността на проекта, но също така определя какъв ще бъде достъпът до информацията и дали потребителят ще бъде въввлечен в процеса. Установяването на графичния тон на екрана със слоя на основния фон /foundation background layer/ е първата стъпка от дългия процес на изграждане на информационните елементи. Една от най-творческите възможности за комуникация е

проектирането и интегрирането на интуитивни и практични управляващи обекти /controls/, с които потребителят да работи и да играе. Информацията не може да бъде оставена просто да се носи по екрана. Тя трябва да се постави в контекст, да се рамкира в прозорци и панели, които са логически и ясно свързани с една основна /подходяща/ мрежа.

Изображенията, анимацията, видеото, музиката, интерактивните бутони..., са основните елементи за истинска интерактивна презентация. Но трябва да се внимава с визуалното претоварване /с текст, изображения,.../, *то е тихия убиец на интерактивността*.

ИЗВОД: Сайтът трябва да бъде завладяващ, просвещаващ, развлекателен, образователен, мотивиращ и запомнящ се. Когато проектирате Web сайтове, запазвайте последователността на метафорите и навигационните елементи във всяка страница, така че хората да разбират къде се намират. Трябва да помнете, че вашите потребители, са както хора които никога не са се включвали в мрежата, така и хора с голям опит в работата с Интернет.

Да проектираш интерфейс, означава да създаваш настроение и преживяване. Ако човек се съсредоточи по-скоро върху създаването на богато изживяване на хората, отколкото върху използването на конкретна технология, продуктът има много по-големи шансове да успее.

Дизайнът включва обмисляне на всички елементи, свързани със структурата на съдържанието, външния вид на интерфейса, начина му на действие, употребата му и посланието, което той предава.

Предизвикателство на дизайна е да познавате толкова добре своята публика, че да можете да задържите нейното внимание и да я направите съпричастна. Готовият софтуер има много предимства пред написания по поръчка, но проблемът е, че трябва да разчитате само на вградените от автора възможности.

Добре организираният и визуално привлекателен интерфейс, дава лесен достъп до голямо количество информация, използвайки хипертекст, илюстрации, снимки, анимация, звук, видео, връзки с Интернет, т.е. създава се интерактивна среда, която приканва към изследване и чрез изследването – генерира знания.

Ученикът трябва внимателно да избира съдържанието, за да постигне комуникативен ефект. Необходимо е създаване на усещане на пространство в интерфейса, така се трансформира плоската повърхност на екрана в достоверно изглеждащо дълбоко пространство, което привлича хората и предава реализъм на преживяването им. Не забравяйте – не претрупвайте зрително сайтът; не го задръствайте с ненужна технология; Използвайте сполучливи дизайнерски подходи и стабилни техники за оформление, за да постигнете фокус и движение в страниците.

Не по-малко важен е шрифтът. Накрая – осъвременяване и актуализация /поддръжка/ на сайта. Някои анализатори обичат да говорят за т. нар. уникални посетители, но тази статистика е фиктивна. При воденето на масивна реклама много лесно се постига голям брой уникални посетители, но това не прави сайта добър, ако тези посетители просто хвърлят един поглед на заглавната страница и си отиват разочаровани, без намерение да се върнат някога. Единственият реален критерий за успеха на един web сайт е повтарящият се трафик от верни потребители. Четири са основните причини потребителите да се връщат на някои web-сайтове, а на други – не. Тези четири критерия лежат в основата на добрия web-дизайн, тъй като това са четирите неща, които потребителите искат най-много да видят. Те могат да бъдат резюмирани накратко чрез акронима НОМЕ (формиран от първите букви на четирите качества):

- ✓ Висококачествено съдържание (High-quality);
- ✓ Често обновяване (Often);
- ✓ Минимално време за изтегляне (Minimal);
- ✓ Удобно ползване (Ease).

Ако можете да осигурите тези четири елемента, потребителите ще бъдат щастливи и ще имате добър сайт.

Интернет навлезе вече повсеместно в живота ни. От ден на ден потребителите стават все по повече. Броят и многообразието на сайтовете в Мрежата също расте и при това с пълна сила важи правилото „Колкото повече – толкова по-добре!“ Но разбира се, не и за сметка на качеството. Как да създаваме добри сайтове – приятни и лесни за разглеждане, чрез които успешно да се продават стоки и услуги и да се получава полезна информация? Целта на настоящата разработка е да се превърне в полезен наръчник за учениците при изграждане на уебсайтове.

Поради специфичния характер на Интернет, най-динамично променящата се медия – е необходимо да се направи уговорка, че всичко представено на тези страници, е актуално към момента.

Уеб услугите променят начина, по който мислим за разпределените софтуерни системи, но те не са всесилни. Уеб услугите осигуряват слой на абстрактност над съществуващите софтуерни системи, като например приложни сървъри, изпращане на съобщения и пакетирани приложения. Уеб услугите работят на ниво на абстрактност, аналогично на Интернет и са в състояние да свържат всяка операционна система, хардуерна платформа или език за програмиране, точно както виртуалното пространство прави това. За разлика от съществуващите системи за разпределено изчисление, уеб услугите са адаптирани към виртуалното пространство. Подразбиращият се мрежов протокол е HTTP. Повечето съществуващи технологии за разпределено изчисление съдържат в себе си комуникационен протокол. Чрез уеб услугите комуникационният протокол е вече там, в необятното виртуално пространство. Подразбиращият се мрежов протокол е HTTP. Повечето съществуващи технологии за разпределено изчисление съдържат в себе си комуникационен протокол. Чрез уеб услугите комуникационният протокол е вече там, в необятното виртуално пространство. Създаването на нови приложения ще стане възможно, когато всичко започне да поддържа уеб услугите. С навлизането на уеб услугите се появяват всякакви видове нови дейности, дискусийни групи, интерактивни форуми и модели за публикуване, които се възползват от тази нова възможност.

1.19 . Същност на интернет, основи за изграждане на интерактивен интерфейс -

Световната компютърна мрежа Интернет възниква през 60-те години на XX достъпни в нея. Потенциалните възможности на световната паяжина WWW (World Wide Web) са огроомни. Мрежата даде нов смисъл на комуникациите въобще. И макар че историята ѝ не е дълга, нейното съществуване и развитие са белязани с уникален в световен мащаб ръст и с необикновена динамика. Влиянието ѝ във всекидневието на българина обаче все още не е осъзнато напълно, както и възможностите, които тя предлага. По-далновидните вече разбират как и в какъв мащаб Мрежата ще промени начина, по който правим бизнес, по който работим, по който живеем.

За да имаме наистина професионално отношение към Интернет, трябва да познаваме неговата история и същност. Историята на Интернет започва през далечната 1969 г., когато в САЩ се формира група от учени от различни области, работещи върху проект, спонсориран от Американското правителство. Тази група е известна под названието Advancet Research Project Agency – ARPA.

По това време съществуват много модели компютри с различни платформи и операционни системи. Основните задачи на ARPA е свеждат до откриване на начин за пренасяне на данни на разстояние и създаване на единни стандарти за комуникация между компютрите. Скоро са открити и други положителни страни на свързани в мрежата компютри, като например съвместно използване на ресурси и обмен на данни. През 1971 г. Мрежата, получила названието ARPA-Net, има вече 23 възела, повечето от които са компютри, намиращи се в университетски центрове. Успоредно се развива и хардуерът. През 1970 г. е открит микропроцесорът, през 1971 г. – флопи дискът, а през 1973 г. – харддискът. Също през 1973 г. е разработен протоколът, който по-късно ще се наложи като най-успешен и масово приложим – TCP/IP. С въвеждането в практиката на този протокол започва истинският бум на компютрите, свързани в мрежа, и на преноса на данни. През 1972 г. е изработена първата компютърна програма за изпращане и получаване на електронна поща (e-mail) и от тогава започва да се измества основното предназначение на компютърните мрежи: учените все повече ги използват за комуникации помежду си вместо за изчисления на разстояние. През следващата година се появяват и първите възли на ARPA-Net, намиращи се извън САЩ – в Англия и Норвегия. Бавно – от гледна точка на това колко бързо следват новостите днес – мрежата нараства и обхваща все повече и повече компютри от целия свят. На дневния ред излизат компютрите, при които обменът на данни се осъществява по телефона.

Междувременно устройствата за осъществяване на връзката между компютрите също претърпяват развитие – през 1985 г. скоростта на модемите е 2400 bps (bit per second), през 1987 г. тя е вече 9600 bps. За сравнение днес максималната скорост на модемите е 56 000 bps и тя е ограничена главно от възможностите за пренасяне на данни по стандартите кабели за телекомуникация.

През 1984 г. растежът на Мрежата е значителен, като броят на включените в нея компютри надминава хиляда. По това време се въвежда и принцип за именуване на отделни възли – DNS (Domain Name System). През 1989 г. световната мрежа се състои едва от 100 000 компютъра, а по приблизителни данни днес в нея са включени над 40 000 000 компютъра. Във вида, в който го познаваме, WWW съществува от 1991 г., а първите версии на популярните днес браузъри се появяват едва през 1994 г. - Netscape 1.0, Internet Explorer 1.0 – 1995 г.

За онези, които днес разучават Мрежата, Интернет представлява едно чудо – място, където има всякаква информация, забавления, игри, текстове, изпълнени с картинки, звук и анимация. Shockwave е нещо нормално, електронната поща е необходимост, скоростта на връзката трябва да е висока, дизайнът на уеб страниците – атрактивен или както вече е актуално ИНТЕРАКТИВЕН, навигацията – лесна, намирането на информация – мигновено. Всички тези изисквания на потребителите на практика са изисквания към всички, които изграждат уеб съдържанието в Мрежата.

Мошта на XML направлява технологиите за уеб услуги като цяло, независимо дали това са основните стандарти, допълнителните технологии или eXtensible Markup Language. XML най-накрая решава проблема за независимостта на данните за езиците за програмиране,

междинните системи и системите за управление на бази данни. Преди това типовете данни и структури бяха специфични за тези типове софтуер и направените опити за общи дефиниции, като например CORBA IDL, не бяха приети много добре. XML е на път да се наложи толкова добре, колкото неговия събрат HTML.

XML не само описва данните независимо, но също така съдържа полезна информация за тълкуването им от всяка софтуерна система или език за програмиране. Уеб услугите притежават почти неограничен потенциал. Всяка програма може да бъде съгласувана с уеб услугите и уеб услугите могат да бъдат съгласувани с всяка програма.

Трансформацията на данните във и от XML е много важна, но XML е достатъчно гъвкав, за да приспособи всеки тип данни и структура и дори ако е необходимо, да създаде нови такива. Когато всички програми и софтуерни системи започнат да поддържат уеб услуги, светът на разпределеното изчисление ще бъде много по-различен от този, който познаваме днес.

XML е разработен, за да се преодолеят ограниченията на HTML, особено за по-добра поддръжка на създаването и управлението на динамично съдържание. HTML е подходящ за дефинирането и поддържането на статично съдържание, но с развитието на виртуалното пространство в посока на софтуерно-базирана платформа, в която данните имат асоциирано значение, съдържанието трябва да бъде генерирано и поддръжано динамично. Чрез XML могат да се дефинират произволен брой елементи, които свързват значение с данни, т.е. описват се данните и какво да се прави с тях, като се използват един или повече елементи, създадени за тази цел. Например:

```
<Company>
<CompanyName region="US">
Skateboots Manufacturing </CompanyName>
<address>
<line>
200 High Street </line>
<line>
Springfield, MA 55555 </line>
<Country>
USA
</Country>
</address>
<phone>
+1 781 555 5000 </phone>
</Company>
```

В този пример XML позволява да се дефинират не само елементи, които описват данните, но също така структури, които групират свързани данни. Лесно е да си представите търсене на елементи, които отговарят на определени критерии, като например <country> (страна) и <phone> (телефонен номер) за дадена компания или за всички елементи <company> (компания), и да върнете списък с тези елементи, които се идентифицират като компании във виртуалното пространство.

Освен това, XML позволява асоциираните схеми да валидират данни поотделно и да описват други атрибути и качества на данните, нещо, което е напълно невъзможно с HTML.

Разбира се, голямата гъвкавост на XML причинява проблеми. Тъй като XML позволява дефинирането на нови елементи, е много трудно да се гарантира, че всички използват едни и същи елементи по един и същи начин, за да изразят едно и също нещо. Ето тук изниква нуждата от взаимно приети, последователни модели на съдържанието.

Две страни, разменящи XML данни, могат да разбират и интерпретират данните по един и същ начин, само ако споделят една и съща дефиниция на данните. Ако две такива страни споделят и една и съща схема, могат да са сигурни, че разбират значението на еднаквите елементи по един и същ начин. Точно това е заложено в основата на уеб услугите.

Начинът, по който изглежда и се използва всеки уеб сайт зависи от използваната технология за неговото разработване. Тук се включват HTML и XHTML, каскаден набор от стилове (CSS), плъгин-модули, скриптове от клиентска страна (JavaScript) и скриптове, разположени на сървъра. Тези технологии се поддържат от основната организация, която установява международни стандарти за www, наречена W3C (World Wide Web Consortium).

За да се избегнат грешки при успоредната работа с HTML и CSS е да се инсталира специализиран софтуер за разработване на уеб приложения, които следят правилното изписване на таговете и въвеждането на основни атрибути. Освен, че определят правилния стандарт за писане, повечето приложения могат да помогнат за откриване на допуснати волни или неволни синтактични грешки. Именно за това може да се използва и периодично обръщане към сайта W3C – консорциума, отговарящи за уеб стандартите. W3C публикува в своята страница online валидатори за HTML и CSS с няколко възможности за проверка.

Зареждайки адреса за валидиране на код (<http://validator.w3.org/>), се виждат три възможности за проверка: Валидиране чрез URL адрес, чрез прикачен файл и чрез поставяне на кода в областта за проверка. След натискане бутона Check в съдържанието на страницата ще се появят допуснатите грешки, с описание къде се намират и от какъв тип са. Ако всичко е наред, се появява надпис за успешно валидиране на уеб страницата: This document was successfully checked as HTML. Аналогично може да се провери и CSS съдържанието на сайт, използвайки адреса: <http://css-validator.org>.

Уеб страниците се създават с помощта на езика HTML (HyperText Markup Language). Чрез него се дефинира структурата на уеб страницата, указва се на брауъра какво съдържа страницата и как трябва да са разположени елементите по нея. HTML документите представляват текстови файлове, които съдържат текст и инструкции за форматирането му (наречени тагове) по време на визуализацията в конкретния брауър. С HTML се описва как да се вмъкнат таблици, изображения, списъци, формуляри и да се задават хипервръзки към различни ресурси в Интернет. Възможностите на HTML се обогатяват при комбиниране с каскаден набор от стилове (CSS – Cascading Style Sheets), който дефинира как трябва да изглежда външния вид на уебстраницата. Същността е в разделянето на съдържанието от представянето, което позволява да се създаде сайт, който е **структурно стабилен, визуално привлекателен** и се зарежда по-бързо.

Ефективността се постига чрез еднократното дефиниране на таговете за шрифт, размери, стил, цвят, подравняване и други характеристики, като този набор от стилове се оформя в отделен CSS файл, който се свързва с всички страници на сайта. Така при необходимост от промяна във форматирането на даден сайт не се налага да се търсят HTML файлове на съответните страници и да се правят многократни промени, а промяната се извършва еднократно – само в CSS файла със стиловете.

* С навлизането на мобилните технологии в ежедневието и необходимостта от показването на едно и също съдържание от уеб пространството на таблети, мобилни телефони и електронни книги, все по-често се използва модифицирана версия на HTML – езикът HTML 5. Той е предназначен да създаде виртуално пространство, което е по-стандартизирано и по-съвместимо с тези приложения и системи.

* Плъгин – модулите са допълнителни технологии, които се смесват незабележимо с брауъра, като дават възможността за възпроизвеждане на аудио, видео и като цяло мултимедийни възможности на страницата. Най-популярна е Flash технологията. Визуализирането става възможно, ако предварително е инсталиран Flash плейър към брауъра. Тази технология позволява да се вградят мултимедийни файлове (аудио, анимация, видео, навигация) директно в уеб страницата, да се задават аткрактивни ефекти на преходи, богати цветове и отлично синхронизиране на звук.

- Скриптовете, изпълнявани от клиентската страна, са малки програми, които се свалят с уеб страницата и носят интерактивност на сайта. JavaScript е главния клиентски скриптов език, който е вграден в повечето брауъри и може да изпълни всякакъв вид функции в рамките на уеб страницата, без да има нужда от помощта на сървъра. Използва се за манипулиране на изображенията и текста, за стартиране на малки изкачащи прозорци, за разширяване и свиване на падащи менюта, за сабиране на информация от формуляри и като цяло за динамични елементи на сайтовете. Скриптовете, изпълнявани от сървърната страна, свързват уеб страницата с база от данни или приложение, позволяващи да се настрои сайта за различни потребители и да се обслужват динамични страници (онлайн магазини, онлайн разплащания и други). Възможни скриптове са PHP, ASP, JSP и др.

Други възможности за изграждане на уеб страници във визуална среда WYSIWYG редактори (What You See Is What You Get – Получаваш това, което виждаш), в който кодът се генерира автоматично. В процеса на работа лесно се превключва между прозореца за редактиране и прозореца за изглед на крайния резултат. Удобството при използването им се дължи на факта, че с тях се работи интуитивно, разполагат с богат набор от шаблони и много сложни задачи като достъп до база данни, създаване на формуляри, електронна търговия и дискуссионни групи, са автоматизирани. Най-популярни продукти са Adobe Dreamweaver, Microsoft Expression web 4, SeaMonkey, системи за управление на съдържание и др. Всеки от тях има силни и конкурентни възможности, които допадат на едни или други потребители, както и известни проблеми (свързани със съвместимостта с брауърите, плагин-модулите, експортирането в Интернет, индексирнето в търсачките машини и др.), разрешаването на които е цел в разработването на нови версии. Съвместимостта между версиите и възможността за адаптиране към новите стандарти са едни от най-търсените предимства за избор на конкретен софтуер в разработката на съвременните уеб сайтове.

1.20. Бъдещето на уеб пространството - Изобретателят на World Wide Web, Тим Бърнърс-Лий е казал, че следващото поколение на уеб пространството ще бъде основано на данни, не на текст; XML е за данните това, което е HTML за текста. Следващото поколение на уеб пространството е предназначено да посрещне някои от недостатъците на съществуващата мрежа, и по-точно: трудността при търсенето на точни съответствия на текстови низове в уеб страниците. Обаче, тъй като уеб пространството е станало толкова успешно, бъдещето му трябва да се реализира като разширение, или еволюция, на текущата версия. Не е възможно да се замени всичко и

да се започне отначало! Решенията за комуникации от тип приложение към приложение трябва да се основат на съществуващите Интернет технологии. Ако бъдещето на уеб пространството зависи от възможността му да поддържа комуникации с данни толкова ефективно и лесно, колкото текстова комуникация, уеб услугите трябва да могат да се обръщат динамично към уеб пространството и към точки, или адреси (URL), и да преобразуват данни към и от XML на заден план. Тези крайни точни, или адреси, предоставят услугите, които обработват XML данните долу-горе по същия начин, по който браузърите обработват HTML текст. Тези адреси могат да бъдат включвани също във всяка програма, която може да разпознава URL и да пар - сва XML. Следователно ще бъде възможно да комуникирате от вашата програма за електронни таблици с отдалечен източник на данни, или пък от програмата ви за управление на сметки с приложение за електронно банкиране, да си уреждате работни срещи с колеги и т.н.

Microsoft и други компании вече са започнали разработката на този тип стандартни услуги, достъпни от всяка програма, и дори голяма част от стратегията .NET на Microsoft е насочена към разработката на инструменти за разработка, позволяващи съшиване на приложения, които използват предварително създадени уеб услуги. За да стане това възможно, са необходими значителни акции по стандартизиране, сравними с усилията, които съпътстват стандартизирането на компоненти за персонални компютри, и следователно могат и да не се случат в бъдещите няколко години. Производителите на софтуер предлагат реализации на уеб услуги за различни архитектури, обикновено според начина, по който разпределят стойността между уеб услугите и съществуващите продукти. Уеб услугите предоставят интерфейси към приложни сървъри, .NET сървъри, брокери за изискване на обекти, междини системи за съобщения, системи за управление на бази данни и пакетирани приложения. Освен това продуктите на някои производители са фокусирани върху стойността на самите стандарти за уеб услуги, като предоставят съвместимост между различни софтуерни домейни, нови механизми за съставяне на приложения и оркестрации за поставяне на различни комбинации от интерфейси на уеб услуги.

Производителите са почти единодушни в поддръжката си на основните стандарти - SOAP, WSDL и UDDI - но мненията им се различават за допълнителните технологии. Всички са съгласни, че сигурността е най-важната следваща стъпка, но вариантите се различават според относителния приоритет на предложенията за транзакциите, последователността на процесите и надеждната обмяна на съобщения.

Що се отнася до ebXML, някои производители предлагат реализации, а други не, макар че често не е ясно какво има предвид производителят под поддръжка на ebXML - дали е само транспортния протокол (който е SOAP with Attachments) или някои от допълнително специфицираните технологии за качество на връзка, регистри и метаданни. Вариантите за производителите за допълнителни технологии и ebXML най-вероятно ще бъдат повлияни от търговски интереси, доколкото производителите се стараят да поддържат технологии, тясно свързани с техните продукти.

На пазара вече има голямо разнообразие от продукти. Основният избор за разработка и внедряване остава между Microsoft .NET и J2EE платформите на Java обществото.

Вижданията за бъдещето също зависят от производителите, като някои от тях предоставят съвсем праволинейни твърдения за влиянието на уеб услугите върху електронната търговия, а други се придържат по-плътено до някоя технология. И в двата случая е ясно, че софтуерните продукти са значително повлияни от реализациите на уеб услугите и че уеб услугите имат огромен потенциал за промяна. Ето и някои уеб услуги: Те са полезни:

- XML (Extensible Markup Language – разширяем език за маркиране), семейство от свързани спецификации, на които са изградени всички технологии за уеб услуги.
- WSDL (Web Services Description Language – Език за описание на уеб услуги), осигуряващ основната и най-важна абстракция на уеб услугите, интерфейса за връзка с други уеб услуги и чрез който уеб услугите се съгласуват с основните програми и софтуерни системи.
- SOAP (Simple Object Access Protocol – Прост протокол за достъп до обекти), осигуряващ комуникационни възможности на уеб услугите за общуване помежду си в Интернет и други мрежи.
- UDDI (Universal Description, Discovery, and Integration – Универсално Описание, Откриване и Интегриране), осигуряващ услуги за регистри и хранилище за съхранение и извличане на интерфейси на уеб услуги.
- eXML (electronic business XML – XML за електронния бизнес), архитектура и набор от спецификации, проектирани за автоматизиране на взаимодействието на бизнес процесите между търговски партньори.
- Допълнителни технологии, отиващи отвъд основните стандарти за уеб услуги, създадени да отговорят на изискванията, свързани със сигурността, надеждността, надеждното изпращане на съобщения, обработката на транзакции и изпълнението на бизнес процесите, за да могат да се използват от по-сложни и по-важни бизнес приложения.
- Реализации на компании, осигуряващи множество реализации, които обикновено съответстват на съществуващи продукти, но в някои случаи са изцяло нови продукти за гъвкави и разширяеми уеб услуги.

2. ПОКОЛЕНИЯ УЕБДИЗАЙН – ИСТОРИЧЕСКИ БЕЛЕЖКИ

Web (Мрежата) поради своето богатство на съдържание и медия - аудио, видео, анимация и музика - популяризира интернет. С течение на времето се развиват и уебсайтовете.

2.1. Първо поколение уебдизайн (1991 - 1994): „Направете го и те ще дойдат“. Това беше всичко необходимо. Всички бяха развълнувани от този привлекателен пълен е информация източник, позволяващ бърз и лесен достъп до допълнителна информация само е няколко кликания. Тези ранни проекти са предимно статични сайтове с основни структури, които се състоят от заглавие и параграфи. Потребителите могат да отидат до различни страници, като щракват върху хипервръзки. Съдържанието на страниците е структурирано в опростено оформление. Личните сайтове са били на мода, все още са. Трябвало е да се напише целият URL (унифициран локатор на ресурси) за достъп до уебсайта. Сайтовете от първо поколение са били ограничени от технологията по онова време. Следващият списък дава някои от характеристиките на технологиите за разработка на сайтове от първо поколение.

- Мониторите са били черно-бели и малки.
- За свързване към интернет са използвани телефонни модеми.
- Интернет скоростите са били бавни.
- Браузърите са използвали собствени разширения, неодобриени от W3C, което води до неправилно въпроизвеждане във всички браузъри.

■ Били са популярни книгите за гости (Guest books).
Сайтовете от първо поколение са били статични - не интерактивни. Включвали са основни структури.

2.2. Второ поколение уебдизайн: „Рекламирай, че продаваш онлайн, и те ще дойдат“.

Тази стратегия за уебдизайн се фокусира върху съдържанието. От време на време съобщението се състезава с технологията. Поради уебинтерактивността, 24-часовия достъп, глобалния обхват, възможността за представяне на продуктова информация за поръчки, както и значителните приходи от онлайн продажби онлайн присъстват вието е от съществено значение за малкия и големия бизнес. Уебсайтовете често включват следните елементи:

- Интуитивна, лесна за използване навигационна система
- Фокус върху съдържанието
- Таблици
- Рамки
- Бутони
- Цветове
- Банери
- Мигащ текст
- Броячи
- Съобщения от типа „Тази страница се вижда най-добре от... съответния браузър“
- Бягащ ред
- Guest books (Книга за гости).

Често срещан проблем по онова време е липсата на стандартизация (за установяване на еднаквост на протоколи и процеси). Сайтове, разглеждани от различни браузъри, невинаги се възпроизвеждат правилно, защото двата гиганта - Netscape и Internet Explorer - предлагат различни функции и разширения в браузърите си.

2.3. Трето поколение уебдизайн: „Дизайн с мисъл за клиентите“.

Сайтовете от трето поколение се опитват да се обърнат към своята публика на базата на предмета на уебсайта, което развива дизайна и структурата. Често в тези сайтове се използва Flash (доставяне на динамично съдържание), за да се предостави на потребителите един вълнуващ интерактивен онлайн достъп. Порталите (Portals) са друго гнездо, което може да откриете в сайтовете от трето поколение. Порталите предоставят врата или точка за достъп, за да откриете информация, продукти и услуги на различни уебсайтове. Отново основният елемент е интуитивна, лесна за използване система. Дизайнът на тези обекти може да включва следните елементи:

- Насоченост към определена аудитория
- Фокус върху съдържанието, а не върху технологията
- Интуитивна, лесна за използване навигационна система
- Използване на колони
- Формуляри
- При отваряне на сайта първо се появява сплаш страница (Splash)²
- CSS (каскадни стилови таблици)
- Портали.

Сайтовете се развиват едновременно в две насоки – от една страна се подобряват техническите възможности на браузърите и се появяват много нови изразни средства (анимиран GIF, CGI, JAVA, REAL AUDIO, REAL VIDEO, FLASH и др.). От друга

страна се променя разбирането на дизайнерите за начините, по които да бъде представена информацията. Именно информацията е в центъра на вниманието още от създаването на първите уебсайтове – как да бъде представена по-атрактивно, какви графични решения ще впечатлят потребителя и ще пояснят съдържанието. Това води до използването на все повече графики, скриптове, бази данни, вграждащи се елементи и т. н.

Сайтовете от първо поколение са предимно текстови, линейни, напомнящи доста на страници от книга, в най-добрия случай с илюстрации. Те включват таблици, хоризонтални разделители, а в отделни случаи и фон. При сайтовете от второ поколение дизайнът съдържа частични графични решения. Започва бум в употребата на бутони, заместващи текстови връзки в менютата. Навлизат в употреба рамките на ярките фонове, както и анимираните елементи с монотонно повтарящо се действие. Думите се заместват от икони и бутони. Информацията следва строга йерархия.

Сайтовете от трето поколение се характеризират с това, че са съобразени с хората, които четат информацията, а не със самата информация. Търси се такова структуриране, което да отговаря на нуждите на интересите на потребителя, като това се съчетава с един лек и в същото време изискан графичен дизайн. Внимателно се определят позицията и връзката между всички елементи в страниците, като се използват много визуални ефекти и метафори, които да направят посещението на сайта приятно префживяване от първата до последната страница.

Съвременният потребител има високи изисквания – той „скача“ от сайт на сайт, не иска да чака трудно е да се привлече и задържи вниманието му. От друга страна, поради спецификата и качеството на връзките скоростта е относителна величина. Затова налагаща се тенденция е сайтовете да се правят с **атрактивни входящи точки**, чието зареждане отнеме не повече от 10 секунди. Ако ученикът желае да сложи по-голям обем за начало, необходима е появата на нещо малко, което да задържи вниманието до зареждането на пълната версия. Размерите на страниците са ограничени или плуващи, с добре обмислено взаимно разположение на елементите, с оглед еднаквото изобразяване на различни монитори, с различна разделителна способност и цветови настройки.

Дизайнът на трето поколение уебсайтове е стилизиран, на моменти пестелив. Навигацията в самия сайт е лесна. Бутоните съществуват, но не се натрапват. Те се сливат и допълват графичния дизайн на сайта, реагират с промяна на цвета или появата на кратко пояснение за страниците, към които водят. Тези пояснения често са чисто текстови. По този начин се постига по-голяма интерактивност и динамика в сайта, като в същото време не доминират, а допълват стила.

Като цяло контролът от страна на ученика при трето поколение сайтове е много голям. Търси се баланс между съдържание и представяне, структура и дизайн. Използваните ефекти съпровождат посетителя при всяка негова стъпка в сайта. Създаването на подобни сайтове е предизвикателство: то е трудно, изисква време и чувство, обвързване не с технологията, а с усещането, което създават.

За мнозина е любопитно какъв ще бъде сайтът от четвърто поколение. Някои казват, че това ще бъдат сайтовете, използващи бази данни и предоставящи индивидуализиран дизайн и съдържание на всеки отделен посетител. Други не са съгласни с това твърдение, защото виждат бъдещето в структурните езици за тримерна графика. Водят се множество спорове за и против сайтове, създадени изцяло с Flash. Трети го свързват с появата на нови периферни устройства и безжични технологии (WAP). Бъдещето ще покаже кой е прав.

С оглед обхващането на всички аспекти на създаването на сайтове е необходимо да се обърне внимание на един специфичен момент – свободата и защитата на авторските права в Интернет. По-голямата част от потребителите считат, че Мрежата е свободно място, и всичко, което е в сайтовете – текстове, графики, скриптове – може да се взима безвъзмездно, да се копира и използва. Всъщност това е така, но само ако е изрично упоменат в сайта. Има много места из Мрежата, предоставящи безплатни ресурси – библиотеки с картинки, различни скриптове, програми за създаване на банери и много други.

Допустимо ли е и доколко професионалните програмисти да използват безплатни ресурси и подобни сайтове?

Процесът е многостранен – на всеки от нас може да се наложи да използва чужди материали, както и неговите да бъдат използвани от други хора. Това е една от спецификите на Мрежата. Трябва да я приемем и да се научим да уважаваме чуждата интелектуална собственост. И ако за лични сайтове използването на чужди ресурси е донякъде приемливо, то при създаването на корпоративни решения това е недопустимо.

Интернет е регулирано и регулируемо място. Пред хората, които се грижат за него, стоят сериозни проблеми – как киберпространството да остане свободно и демократично и как все пак да бъде регулирано.

Не са много сайтовете, предоставящи възможност за електронни разплащания, малко стоки и услуги могат да бъдат намерени в българското уебпространство.

Въпреки всички регулативни или психологически пречки, които стоят пред навлизането на този род търговия процесът не може да бъде спрян. С времето той ще става все по-популярен в световен мащаб и в частност у нас. Разработват се и други системи за електронни разплащания (освен тази на ePay.bg), с което може се ускори процесът на приемане на необходимите законови разпоредби от една страна и ще се създадат значително повече сайтове, предлагащи подобни възможности, от друга. Развитието на електронната търговия даде нов тласък на икономиката като цяло.

„Оптимизмът е стратегия за изграждане на по-добро бъдеще. Защото, ако не вярваш, че бъдещето ще е хубаво, едва ли ще направиш нещо за него и ще поемеш отговорността за това.“

Н. Чомски

Въпреки многообразието на Интернет днес ние сме убедени, че това е само началото на пътя, по който е тръгнало информационното общество и по който след дълги години на търсене ще достигне до нови и неподозирани възможности за комуникация.

Уебсайтовете са съществен елемент от това развитие, тъй като в момента те са посредникът в общуването, а създателите им са хората, от които зависи то да бъде лесно, приятно и ползотворно. Интерактивността, динамиката, добрият дизайн, полезното съдържание, ясната структура са критериите за успеха на сайта. За да бъдете в крак с времето като професионалисти, от учениците се изисква не само да са активни участници в неговото развитие.

- 2.4. Четвърто поколение уеб дизайн: „уикита, блогове, подкасти и още“ -** Това поколение на дизайна е наистина интерактивно. Тези сайтове персонализират потребителското онлайн преживяване. Добре проектираните

сайтове се фокусират върху визуалната организация на съдържанието, използването на интуитивна навигационна система на всяка страница, внимателен подбор на типография (шрифт, цвят, размер и разстояние), съответствие на стандарти и единен и последователен дизайн за по-изтънчен и професионален изглед.

Уебсайтовете вече не са статични, непроменящи се източници. Посетителите имат възможност да създават и променят съдържанието, което гледат в Мрежата. Използват се инструментите на Web 2.0. Често срещаните елементи в Web 2.0 сайтове включват следното:

- Динамично съдържание, създадено от потребителя
- Социално-мрежови функции
- Съответствие с уебстандартите
- Заоблени ъгли
- Изтънчени 3D ефекти
- По-голям текст
- Видеосподеляне
- Инструменти за редактиране на авторско съдържание, коментари и ревюта
- Интернет поверителността позволява на потребителите да управляват своята лична информация, като изтриват потребителското си съдържание или профил
- Уебслужби
- Уикита
- Блогове
- Подкасти
- Уебкасти
- RSS захранвания
- Cascading style sheets (Каскадни стилови таблици)
- Freemium бизнес модел
- Отворен код
- Споделяне на файлове.

Използването на AJAX (Asynchronous JavaScript and XML) подобрява уебдизайна, защото позволява информацията да се обработва без презареждане на страницата (съдържанието се актуализира веднага, когато потребителят извърши действието. Mobile Web (Мобилна мрежа) предоставя безжичен достъп до преносими компютри, телефони или всички преносими устройства, които могат да извличат данни. Едно от предизвикателствата, пред които са изправени потребителите на мобилния интернет, е малкият размер на екрана, който създава проблеми с използваемостта. Друго предизвикателство е оперативната съвместимост в резултат от несъвместими мобилни устройства. Дизайнерите на сайтове вземат предвид тези проблеми поради увеличаващия се брой преносими устройства за свързване към интернет. W3C предприема стандартизация, което е липсващият компонент.

W3C разработва следните Mobile Web Best Practices (Най-добрите практики за мобилен интернет), които поддържат дизайна и предоставянето на съдържание към мобилни устройства. Според W3C десетте начина да създадете достъпен интернет за мобилните устройства са:

- Дизайн за Web, разглеждан от много устройства.
- Доверете се на уебстандартите.
- Стойте настрана от известни опасности.

- Добрият дизайн намалява проблемите с ползваемостта (малки екрани и клавиатури).
- Премахване на pop-up (изскачащи) прозорци.
- Не използвайте рамки или таблици.
- Бъдете внимателни с ограниченията на мобилните устройства.
- Оптимизиране на навигацията.
- Проверете графиките и цветовете.
- Оставете ги малки.
- Използвайте мрежата пестеливо.
- Помогнете и напътствайте въвеждането на потребителя.
- Помислете за потребителя в движение.

Не всичко на разположение онлайн е напълно точно. Всеки може да създаде и публикува уебсайт без надзор или без да е доказал, че съдържанието е достоверно. Поради липсата на редакционен контрол отговорността за проверка на точността на информацията се пада на потребителите. Важно е да се оцени онлайн ресурсът, преди да се използва.

Има няколко важни критерия при определяне на валидността на даден сайт:
 Прецизност Цитирани ли са източниците? Добре обучен и с широки познания ли е авторът? Има ли линкове, предоставящи допълнителна информация в подкрепа на съдържанието в сайта?

Има ли информация за контакт с автора - име, пощенски и/или e-mail адрес? Има ли информация за копирайт на автора?

Обстойно и уместно ли е съдържанието?

Колко често се обновява сайтът? Има ли неработещи линкове? Остаряла ли е информацията?

Каква е целта на сайта? Представени ли са предубедени мнения?

Проверете URL, за да определите вида на сайта. Като цяло .gov, .org, и .edu, .bg, сайтове дават достатъчно обективна информация, защото не рекламират продукт или услуга. Определете дали това е личен сайт. Знак, че сайтът е личен, е тилда (~), % или думата member (член) или user (потребител) в URL. Не забравяйте, че няма допълнителен борд за одит на уебсъдържанието. Проверете за линкове към основната информация за организацията - About Us (За нас), Mission (Мисия) или Purpose (Цели). Вижте какви организации имат връзка към уебсайта. За да откриете информация за сайта, можете да използвате <http://www.alexa.com>.

Допълнителните въпроси, които да зададем при оценяване на уебсайт, включват: Има ли сайтът професионален изглед? Лесен и интуитивен ли е за употреба? Забелязвате ли грешки в правописа, граматиката и/или пунктуацията?

Интернет безопасността също е важна.

Интернет предоставя на потребителите богата информация, ресурси и инструменти.

Това може да бъде и опасен форум. Трябва да сте наясно с рисковете, когато работите онлайн. Някои от рисковете включват:

- Кибер тормоз
- Онлайн хищници
- Кражба на идентичност
- Застрашаване на приемането Ви във ВУЗ или при постъпване на работа.

Често използвани техники от нарушителите

- Phishing (Фишинг) - изпращане на измамни електронни писма от предполагаем кредитор, който иска вашата лична и финансова информация. Фишингът често се разпознава по грешките като неправилно изписани думи и сбъркана граматика.
- Търсене на онлайн профили за лична информация.
- Изпращане на неуместни електронни писма или съобщения.

Основните принципи, на които е изграден протоколът TCP/IP и чрез него – Интернет, са два: децентрализацията и пакетният трафик. Децентрализацията (т. е. равноправен статут) означава, че всички възли в Мрежата са с еднакъв статут – всеки от тях да може да приема и изпраща съобщения, без за това да е необходим координиращ или субординиращ “център”. Пакетният трафик е втората основна характеристика на протокола TCP/IP – предаваните данни се разпределят на части, в пакети. Те изпращат от един възел в друг. Препращането им става чрез специални програми, които обикновено работят на компютри, наречени маршрутизатори или рутери. При тях се проверява къде трябва да отиде всеки пакет и чрез специални таблици се насочва към друг маршрутизатор. Пакетите пътуват директно само между два най-близки възела от Мрежата. В останалите случаи се извършва неколккратно прехвърляне, докато пакетът стигне до целта, като се използва не най-близката, а най-бързата в момента връзка. Не е важно по кой път ще премине пакетът. Ако един или няколко възела в Мрежата престанат да функционират, пакетът пак ще бъде предаден, стига да има маршрут от началната до крайната точка. Интернет е свободно организирано международно взаимодействие от свързани, но автономни мрежи, осигуряващи комуникация от типа компютър – компютър чрез доброволно използване на протоколи и процедури. Световната мрежа е базирана на протокола TCP/IP, който е рутиращ и организиращ мрежов протокол и има архитектура, улесняваща свързването на компютри от хетерогенна среда.

IP – адресът е 32 битов логически адрес, уникален за всеки компютър в световната мрежа. Той се изписва с четири числа, разделени с точки, всяко от които може да приеме стойност 0 до 255. Те оформят две части – мрежова идентификация (първите две цифрови групи от адреса). Мрежовата идентификация обхваща всички хосткомпютри в дадената физическа мрежа.

В Световната мрежа съществуват т.нар. Първични сървъри (с адреси, започващи с числата от 1 до 126-най-големите световни мрежи и мрежови сървъри), и в които се пазят мрежовите идентификации на всички сървъри в Интернет. Те се групират на второ ниво по национални или локални принципи във вторични сървъри. За удобство в началото на 90-те се създаде възможности IP – адресите да се заменят с имена. За да може да се организира разпознаването на имената от сървърите, се налага използването на области - домейни и именни сървъри (Name Servers). Основният домейн се регистрира и изписва с име и разширение, разделени с точка от вида:

www.yahoo.com

Основните, международно възприети домейни са два типа: световни –

gTLD (generic Top-Level Domains) - .com, .net, .org, и регионални, специфични за всяка държава – ccTLD (country code Top-Level Domains) - .bg за България, .ca за Канада, .dk за Дания, .fr за Франция и т. н.

Организацията, която се занимава в момента за раздаването на тези имена е ICANN (Internet Corporation for Assigned Names and Numbers) – www.icann.org.

2.5. Фактори, водещи до разлики при възпроизвеждането –

По-долу са посочени основните фактори, водещи до разлика при възпроизвеждането на уебстраниците, както и тяхната тежест и необходимостта да се съобразяваме с тях.

1. Браузърите са продукт на различни производители и това води до разлики в интерпретирането на един и същ HTML – код. С течение на времето тези разлики (противно на очакванията) стават все повече, а едновременно с това дизайнерите стават все по-взискателни и искат все по-голям контрол върху изгледа на страницата. Въпреки усилията на World Wide Web Consortium различията остават, защото техните стандарти се интерпретират твърде широко в борбата за извоюване на по-голям пазарен дял. По този начин W3C върви след браузърите вместо пред тях, а единият стандарт остава под въпрос.

ИЗВОД: Необходимо е уебстраниците да бъдат съобразени с разликите между отделните браузъри. Прието е различните версии на софтуерните продукти да се бележат с цифри след името. Браузърите не правят изключение от този стандарт – например обикновено получава номер, по-голям с единица от предходната.

Извършва се непрестанна разработка и обновяване на браузърите, поправят се бъгове, въвеждат се нови функции и на пазара непрекъснато се появяват нови версии. По този начин се предоставят нови възможности и изразни средства, които липсват в предишните версии. Същевременно страниците, използващи тези възможности, са почти непригодни за употреба с някой от по-старите браузъри. Не е необходимо уебстраниците да са съобразени с най-старите версии на браузърите – те вече са излезли от употреба, защото не отговарят на потребностите в Мрежата. Все пак активни в момента са няколко от последните версии, така че съобразяването с тях е желателно.

2. Операционната система също оказва влияние върху работата на браузъра и респективно върху вида на страниците. Netscape for Windows и Netscape for Unix са различни програми, макар и създадени от един производител. Основното влияние, което оказват операционните системи, е върху шрифтовете, които са системно заложили и които се използват от браузъра за изобразяването на текстовото съдържание върху екрана. Докато при Windows се ползват Arial и Verdana, то при Unix се ползват Geneva и Helvetika. С течение на времето имената се уеднаквяват, но се появяват разлики в големината – шрифт с големина 12 пункта при Unix визуално е по-малък от същия при Windows.

При изграждането на страниците разликата в операционните системи може да бъде пренебрегната, тъй като, от една страна, не е възможно да се направи нещо по въпроса, а от друга – потребителите на даден сайт обикновено използват един тип операционна система. Необходимо е сайтът да е разработен и изпробван на нея.

3. Въпросът за големините на отделните елементи в страницата (текст, графика и др.) става още по-сериозен, когато се отчете фактът, че потребителите работят на различни по големина монитори и респективно различни настройки на разделителната способност на екрана – например 640x480, 800x600, 1024x768 и т. н. Тъй като графиката в мрежата е предимно растерна, тя се изобразява по начин, при който всеки пиксел от нея съответства на един пиксел от екрана на монитора. Така реалната големина на графиките в страницата зависи първо, от разделителната способност и, второ, от големината на монитора. Това често води до появата при някои потребители на хоризонтална лента за позициониране или до нежелано разместване на взаимното положение на елементите. Затова, задължително е уебстраниците да бъдат съобразени с най-разпространените резолюции.
4. Влияние оказват и настройките на самия браузър. От менюто Options всеки отделен потребител може да зададе големината и вида на шрифтове, които да използва, цвета на връзките, може да изключи зареждането на графиките, за да ускори работата си в мрежата, и т. н. В някои браузъри се предлагат и възможности за изключване на фон, таблици, настройки на параметрите за сигурност с изключване на т. нар. cookies, скриптове и аплети.

Задължително по време на проектирането на уебстраниците или сайтовете е невъзможно да се отчетат всички възможни индивидуални настройки. По тази причина те обикновено се пренебрегват. Потребителите рядко правят промени в настройките по подразбиране, а ако има такива, те в повечето случаи срещат проблеми при сърфирането в по-голямата част от сайтовете в мрежата.

- 2.6. **Верният подход** - Задачата на учениците и разработчиците е да представят информацията по най-добрия начин на възможно най-голям брой потребители и съобразно предназначението на сайта. За съжаление те са в много деликатна ситуация, тъй като нямат пълен контрол върху изгледа на страниците. Главната причина за това, както бе споменато, си остава езикът HTML. **Основна характеристика в работата на учениците е балансът** – добре прецененият компромис между изискванията и възможностите. Необходимо е добро познаване на HTML, на разликите между браузърите и съобразяване с всички опционни ситуации при ползване на уебстраницата. Единият възможен подход е да се ползват само общите за всички браузъри команди. Това прави страницата универсална, но лишава дизайнерите от възможността да си служат с някои добри интерактивни възможности. Другият подход е да се използва програмен код, който да определи вида на браузъра и след това да зареди страницата, специално подготвена за него. Очевидно наред с предимството да се осъществи по-добър контрол върху

страницата тук съществуват и много недостатъци – възможно е браузърът изобщо да не стартира въпросния програмен код, от една страна, а от друга изработката на сайта се усложнява, тъй като трябва да се подготвят не една, а няколко версии. Окончателното решение е изцяло в ръцете на екипа, разработващ сайта, като решаващи са конкретните детайли на проекта.

Може би най-важният момент при изграждането на сайта са тестовете, проведени най-малко на два различни брауъра и на поне две техни версии: по-старата и най-новата. Тестовете са единственият начин, по който може да се гарантира качеството на крайния продукт и да се открият допуснатите неточности. Препоръчително е да се изберат тези брауъри, които са най-използвани в момента. Добре е актуална информация за това да се търси директно в Интернет.

2.7. Оценяване - Кои са критериите, по които може да бъде оценено качеството на даден сайт? Това е мнението на конкурентите и на журито в евентуален конкурс по уебдизайн. Там се налагат определени критерии, съобразени с нуждите и възможностите на потребителите в Мрежата, които ще го оценят по същите тези критерии, но интуитивно поради липсата на повече познания в областта на уебдизайна. Оценяването на един сайт е сложен процес – най-добрият съдник са посетителите. Могат да се дадат различни мнения, съществуват различни гледни точки, но основните критерии се заключават в следното:

- Уебстраницата трябва да изглежда еднакво добре независимо дали е отворена с Netscape, Internet Explorer, Opera, Mozilla или друг браузър;
- Уебстраницата трябва да е съвместима с по-старите версии на браузърите – JavaScript, Java, DHTML, XML, Flash и др. често извеждат съобщения за грешки или променят дизайна на страницата при брауъри, които не ги поддържат;
- Уебстраницата трябва да изглежда еднакво добре независимо дали мониторът на компютъра е с разделителна способност 640x480, 800x600, 1024x768 и т. н.;
- Уебстраницата трябва да се зарежда бързо, защото потребителите са нетърпеливи и често се отказват, преди тя да се е визуализирала докрай.
- Графичният дизайн на уебстраницата трябва да е съобразен с основните принципи на композицията и цветознанието;
- Навигацията и връзките в сайта не трябва да създават чувство за обърканост – потребителят интуитивно трябва да знае в коя част на сайта се намира и как да стигне до следващата.
- Статистиките показват среден престой на страница 10 секунди! Този тест трябва да се провежда онлайн, а не от харддиска на компютъра. При средна скорост на връзката 36 60000 bps за секунда се предават максимум 4,47 KB, т. е. за страница с големина 50 KB (заедно с графиката към нея) ще са необходими минимум 10 секунди.

3. ПРИНЦИПИ НА УЕБДИЗАЙНА.

Дизайн, насочен за ползване с компютър – въпреки, че все повече устройства предлагат поддръжка за Уеб, целта ни като уебдизайнери е да създадем уебстраници, предназначени за ползване чрез компютър. Трябва да съобразим как оформлението, шрифтовете и цветовете ще се изобразят на компютърния монитор. Като автор на HTML трябва да имаме предвид и нелинейната същност на хипертекстовите документи и да интегрираме логически връзки в самото съдържание. Трябва да се даде възможност на потребителите да проследят желаната от тях информация, като свържете сродните теми с подходящи хипертекстови връзки. Потребителят трябва да се чувства сигурен, докато разглежда сайта, така че във всеки момент да знае къде точно се намира и на къде да продължи.

- 3.1. **Изглед и усещане на сайта:** Интерфейсът, чрез който потребителите се придиждат из сайта, често се нарича изглед и усещане (look and feel) на сайта. Посетителите виждат и усещат сайта, дакото разучават неговото информационно съдържание. Те четат текста, проследяват асоциации чрез връзките, разглеждат графиките и ако вашият дизайн позволява, си създават собствен „път“ през предоставената от вас информация. Изгледът и усещането на сайта включват начина на работа с него и характерната атмосфера, която той създава за посетителите. Когато създавате уебдизайн, трябва да планирате не само конкретния изглед и усещане на сайта, но и неговото поведение при сблъсък с променливата природа на Уеб. Трябва да се убедите, че максимален брой потребители могат да ползват сайта успешно!
- 3.2. **Напревете дизайна си преносим:** За да има успех, дизайнът на сайта трябва да бъде преносим и достъпен за потребител, ползващи различни по вид браузъри, операционни системи и компютърни платформи. Много дизайнери допускат грешката да тестват само с една конфигурация като самятат, че страниците им ще изглеждат по същия начин при всички потребители. Независимо колко опитен дизайнер сте, никога не забравяйте да проверите работата си при различни условия, дори когато сте уверен в постигнатия резултат. За да сте уверени, че сайтът ви ще бъде достъпен за максимален брой хора, винаги го преглеждайте с браузърите, които очакват потребителите ви да ползват. Тествайте го с най-популярните операционни системи и на повече от една компютърна платформа. Анализирайте аудиторията и си изградете профил на типичния потребител.
- 3.3. **Съобразете дизайна с бавна Интернет връзка:** планирайте страницата си така, че да бъдат достъпни при различна скорост на връзката. Ако страниците ви се зареждат бавно, защото съдържат големи изображения или сложни анимации, потребителите Ви ще напуснат сайта още преди да видят съдържанието му. Средният потребител затваря даден сайт, ако той не се е заредил в рамките на 10-20 сек. Мнозинството от потребителите вече ползват постоянен и бърз достъп до Уеб, но много хора все още разчитат на сравнително бавна връзка.

Добре е да ги имате предвид, когато обмисляте изгледа и усещането на вашия сайт.

Повечето потребители просто не биха чакали повече от 10-20 секунди, за да се зареди една страница. Ако страниците Ви се зареждат бавно, посетителите вероятно биха продължили към друг сайт, преди да са видели дори част от съдържанието Ви. Много дизайнери допускат грешката да не тестват страниците си при различни скорости. Ако не направите тест, не бихте могли да разберете как потребители с различна скорост на връзката зареждат Вашия сайт и по този начин ще загубите ценни за Вас посетители. Макар повечето потребители в САЩ вече да разполагат с високоскоростен достъп до Интернет, близо една четвърт от тях все още ползват комутираема връзка (dial-up). Достъпът чрез кабелен модем беше до скоро най-сигурното средство за високоскоростна връзка при домашните потребители, но напоследък нараства популярността на DSL (Digital Subscriber Line) – услуга, предлагана от телефонните компании. Корпорациите все още разчитат предимно на T1 или ISDN (Integrated Services Digital Network) връзки. В таблицата по-долу е показано най-често срещаните типове свързаност.

Технология	Скорост	Описание
Стандартна телефонна линия	До 56 Kbps	Около 20% от потребителите в САЩ все още ползват този метод за връзка с Интернет. Ако с тази връзка успеете да задържите постоянна скорост над 44 Kbps, по-скоро се счита, че сте късметлии.
Базов ISDN	от 64 Kbps до 128 Kbps	ISDN предлага добра скорост и постоянна връзка с Интернет, но е сравнително скъп метод. Използва се предимно в градовете, най-често от фирми.
DSL	от 512 Kbps до 8 Mbps	DSL използва съществуващите телефонни линии за пренос едновременно на глас и на данни. Позволява постоянна връзка с Интернет.
Кабелен модем	от 512 Kbps до 52 Mbps	Кабелните модеми са бързи, осигуряват постоянна връзка с Интернет и не е необходимо да набирате, за да се свържете.

Таблица 1. Най-популярни типове технологии за Интернет връзка

Въпреки нарасналия брой потребители с високоскоростна връзка, уебдизайнерите би трябвало да се стремят да запазят страниците си достатъчно

прости, за да се зареждат бързо. Единственият значим фактор, който влияе на скоростта на зареждане, е размерът и броят на графичните елементи в уебстраниците Ви. Като практическо правило никое изображение на уебсайт не трябва да надвишава размер от 10-15 KB. Ако сте сигурни, че всичките Ви посетители имат бърза връзка към Интернет, можете да съобразите страниците си с този фактор. При сайт с широк кръг от потребители е най-добре да приемете 56Kbps като базова скорост, тъй като много хора все още ползват комутируема връзка.

- 3.4. Предвидете лесен достъп до информацията:** Информационната структура на сайта е най-важният фактор за неговия успех. Тя определя колко лесно потребителите ще се ориентират в съдържанието. Целта ви е да организирате и представите информацията по логичен и лесен за навигация начин. Трябва да предоставите на потребителите си разнообразни навигационни средства, без да ги отклонявате от крайната им цел. Всеки посетител може да реши за себе си дали просто ще разглежда сайта, или ще търси конкретна информация. Често на страницата попадат хора, търсещи информация, която не е от първостепенно значение за самия сайт. Някои потребители посещават даден сайта заради конкретна част от съдържанието, например телефонен номер или формуляр за поръчка. Помислете предварително кои са действията и начините, по които потребителите ви ще обхождат сайта ви. Осигурете преки връзки към най-търсените раздели. Помнете, че потребителите искат да отворят сайта, да извлекат необходимата информация и да продължат нататък.
- 3.5. Представете ясно информацията:** Ниската разделителна способност на компютърните монитори ги прави неудобни за четене. Светлината, която идва иззад текста, изморява очите. Потребителят се натовазва допълнително и от влиянието на други фактори, като отраженията или разстоянието от монитора. За да избегнете тези проблеми, направете дизайна си лесен за четене. Много уебсайтове не отговарят на критерия за яснота, тъй като ползват твърде много различни шрифтове, цветове и дълги пасажки текст. Разделяйте текста на малки откъси, за да улесните четенето от екрана. Използвайте контрастни цветове, например тъмни букви на светъл или бял фон, тъй като те са по-лесни за възприемане от човешкото око. Вземете под внимание, че потребителите имат различни навици, когато четат онлайн текстове. За разлика от печатния текст, текстът в Уеб повече се преглежда и по-малко се чете. Потребителят плъзга поглед по дългите текстове, докато превърта страниците. За да го ориентирате в съдържанието, ползвайте повече вътрешни заглавия. Ограничете ширината на текста, оформете ясни, лесни за четене колони. Спазвайте правилото „седем“ (плюс-минус две) - потребителите не могат да възприемат повече от седем (плюс минус) две стъпки или сегменти информация на веднъж. Например една добре написана процедура не трябва да съдържа повече от девет стъпки. Вместо да създавате дълги, превъртащи се страници, разделете информацията на по-малки сегменти и ги свържете посредством хипертекст.

3.6. Плавни преходи: Планирайте единен облик за всички секции и страници на вашия сайт. Подсилете елементите, придаващи индивидуалност, и създайте плавни преходи от една страница към друга чрез повтарящи се цветове и рифтове и чрез деформление на страниците, което позволява йерархични нива. Избягвайте неочакваните резки промени, освен ако това не е целта ви. Вътрешната съгласуваност в дизайна осигурява плавен преход между отделните страници, дава увереност на потребителите, че все още се намират на вашия сайт, и им помага да открият търсената информация.

Дайте на посетителите си опорна точка, като поставяте навигационните елементи ванига на едно и също място. Те бързо ще се ориентират в навигационната структура. Използвайте едни и същи графични елементи за навигация в целия сайт. Това ще допринесе за вътрешната съгласуваност на сайта и ще елиминира нуждата от изтегляне на множество различни изображения.

Когато минават от една уебстраница към друга, потребителите все едно разлистват списание. Въпреки че всяка страница трябва да бъде завършена изцяло, тя същевременно е и част от целия сайт. Дизайнът на отделната страница трябва да отразява идентичността на сайта като цяло, независимо къде е позиционирана в йерархията на сайта.

3.7. Препоръчва се решетъчна структура на сайта: структурата на уебстраницата е продиктувана от това дали ще решите да ползвате решетъчна структура или шаблон за страници. Решетъчната структура е метод за оформление, при който елементите по страницата се организират в редове и колони. Можете да използвате решетъчна структура, за да постигнете визуално единство на всички страници в сайта. Когато искате да внесете разнообразие или да подчертаете важна информация, бихте могли да нарушите решетъчната структура и да поднесете тази информация по друг начин. Страниците, които ползват решетъчна структура и я спазват при подреждането на текста и графичните елементи, изглеждат по-елегантни и завършени от страниците с липса на такава подредба. Повечето съвременни сайтове използват таблици по един или друг начин, за да постигнат структурираност и консистентност на страниците си. Ако на таблицата е зададено да не се показва рамка, потребителите няма как да разберат, че съдържанието е разположено в таблица. За тях това е просто една ясно структурирана и подредена страница. Таблиците като средство за позициониране в крайна сметка ще отпадне, тъй като все повече нараства използването на по-новите браузъри, които поддържат позициониране на съдържанието в колони без употреба на таблици. Повече за позиционирането - чрез CSS.

3.8. Използвайте активно бяло поле: Бели полета се наричат празните пространства в страницата, независимо от това какъв цвят им е зададен. Стремете се да използвате белите полета целенасочено при изграждане на дизайна, а не като допълнение към него. Правилната употреба на бели полета

помага на потребителя да се ориентира по-добре, като в същото време разделя отделните части на страницата. Белите полета, които са използвани целенасочено за структуриране и за отделяне на дадено съдържание от друго, се наричат активни бели полета и са неделима част от дизайна на сайта. Понякога те имат определяща роля при структурирането на дизайна. Пасивни бели полета са празните пространства, които граничат с външната част на екрана или са резултат от грешна подредба на елементите. Визуализацията на съдържанието може да стане объркваща, ако не се използват достатъчно активни бели полета за разделяне и обособяване на отделните му елементи. Липсата на активни бели полета създава усещането, че съдържанието на страницата е твърде много и че откриването на търсената информация ще бъде трудно.

3.9. Направете дизайн, ориентиран към потребителя. Подчинете и съобразете дизайна изцяло с нуждите на потребителите си. Познавайки аудиторията си, ще можете да отговорите на почти всички въпроси, свързани с дизайна: ако нещо е полезно за потребителите – реализирайте го, ако е възможно да ги разсейва или дразни – премахнете го. Установете какво очакват посетителите от сайта ви. Ако е възможно, пуснете анкета на сайта. Съберете отговорите на основните въпроси и на тяхната база създайте профил на типичния потребител. Какво очакват потребителите ви от сайта? Дали са дошли да получат поддръжка или помощ за някакъв продукт, или желаят да купят нещо? Искат ли да четат статии или да търсят информация? Веднъж разбрали интересите на потребители си, ще можете да прецените доколко дизайнът на сайта ги задоволява и отразява техните потребности. Погледнете заглавната страница на www.google.com, най-популярната Интернет търсачка. Тя не съдържа реклами, има много малко връзки и дизайнът ѝ предполага само една крайна цел – да позволи на потребителите бързо да въведат дума за търсене. Изводът при сравнение на началните страници на няколко сайта е следния: Дизайнът трябва да съответства на визуалните очаквания на аудиторията – това е изгледът на сайта. Важно е обаче да се съобразите и с начина, по който потребителят взаимодейства със съдържанието – усещането от сайта!

3.10. Как да направим унифициран дизайн за Уеб? Когато създавате уебсайт, планирайте обща тема и структура за всички страници в него. Изборът на цветове, шрифтове, изображения и оформление трябва да формира единна визия, която да улесни потребителите на сайта ви. Темата трябва да съответства на впечатлението, което искате да създадете за вашата организация. Изборът на меки цветове, четливи шрифтове и отделно структурирани колони подчертава двете основни теми – информационна и новинарска – ако разгледаме сайта www.cdc.gov. За разлика от сериозността, която излъчва сайтът за възрастни, детският комбинира ярки цветове, широк забавен шрифт, раздвижена структура и привличащи окото картинки. Така се постига повече жизненост и игровост.

Когато създавате дизайна на сайт, не трябва да мислите за всяка страница поотделно. За да бъде сайтът ви хармоничен и унифициран, предвидете плавни

преходи, базирайте визуалната структурар върху решетка и използвайте активни бели полета.

- 3.11. Важно е взаимодействието:** Помислете как потребителят иска да взаимодейства с информацията на вашата уебстраница. Изградете дизайн според типа на съдържанието и вземете под внимание дали посетителят ще чете, или просто ще преглежда страниците ви. Например, ако страницата е заглавна за сайта или за дадена секция, тя ще се състои предимно от връзки. Потребителят взаимодейства с такъв тип страници, като преглежда съдържанието, превърта при необходимост, посочва графиките, за да провери дали са връзки, и щраква върху текстовете с връзки. За такова взаимодействие дизайнът на страницата трябва да бъде с разбираеми заглавия, хипервръзки в текста и кратки описания. Организирайте връзките тематично в групи, а тях разделете с бяло поле, графични елементи или фон цвят.
- 3.12. Разположението на елементите също е много важно:** Трудно е да се предвиди в какъв ред потребителят ще прегледа страницата. Все пак, съществува общоприето схващане за относителната тежест на отделните области от екрана.

2		
5	1	3
4		

- 3.13. Използвайте хипертекстови връзки:** за разлика от авторите на хартиени издания, създателят на хипертекст има предимството да добавя при необходимост връзки в текста или изображенията и така да води потребителя през съдържанието. Това преимущество върви ръка за ръка с допълнителната доза отговорност. Вие вземате решение как потребителите ще се придвижват през сайта и как ще възприемат информацията. Читателят на списание може да отгърне на произволна страница винаги, когато пожелае. Вие можете да възпроиветете този нелинеен начин на четене като поставите връзки в сайта, които позволяват на потребителя да се придвижва от страница към страница и от секция към секция в произволен ред. Чрез внимателно използване на хипервръзките бихте могли да завладеете потребителите по един напълно нов начин. Много сайтове имат отделни колони с връзки, докато тези, които предоставят връзки в самия текст, са малко на брой. Това е една мощна възможност на хипертекста, която не се използва достатъчно. Вплетете връзки в текста, за да предложите на читателя различни пътища до информацията. Избягвайте безсмислени изрази като „Натиснете тук“. Заменете го с кратък, но ясен и полезен текст, който дава информация накъде води тази връзка. Всички сме влизали в сайта Wikipedia – обърнете внимание как хипертекстовите връзки са вмъкнати директно в текста. Когато потребителят избере дадена връзка, попада на друга страница с информация. От там той може или да се върне обратно, или да продължи към следващата страница с информация и така нататък. Наличието на достатъчно хипертекстови връзки позволява всеки

потребител да си създаде уникален поглед върху съдържанието на сайта. Създайте повече връзки, за да могат посетителите да се движат из сайта по-бързо. Подсигурете връзки, които да ги връщат към навигационната част, картата на сайта или към заглавната страница. Не карайте потребителя да превърта дълги страници текст. Осигурете връзки, които да го препращат директно към долната част на страницата, такива, които да го връщат най-горе, и други, които да сочат обратно към по-високите нива на сайта. Осигурете хипертекстово съдържание за текста, за да може потребителите да прескочат директно към темата, която ги интересува. Ползата от хипертекстовото съдържане е в различното оцветяване на връзките, което показва на потребителите кои страница вече са посетили и кои не. По подразбиране непосетените връзки са в син цвят, а след посещение стават лилави. Хипертекстовото съдържание отбелязва ясно и разбираемо за потребител къде е бил и какво още му остава да посети. С помощта на хипертекста речниците и други документи с висока информационна плътност стават много по-лесни за ползване.

3.14. Не претрупвайте страниците си: Възможно е да претрупате дадена страница с твърде много информация. Вземете под внимание „прага на любознателността“ на потребителя. Въздържайте се от изкушението да поднесете на посетителите на сайта си твърде много информация. Осигурете подходяща навигация, за да бъде локализирано бързо търсеното съдържание. Разделете съдържанието на малки порции и използвайте хипертекстови връзки, за да го разпределите между страниците.

3.15. Не забравяйте хората с увреждания! Всеки сайт, ориентиран към широката публика, трябва да очаква, че потребители с физически увреждания също ще искат да разгледат съдържанието му. Осигуряването на достъпност при дизайна на даден сайт означава страниците да бъдат конструирани по такъв начин, че да са достъпни независимо от физически, сензорни или когнитивни увреждания, работни ограничения или технологични пречки от страна на потребителя. По думите на Тим Бърнърс Лий: „Силата на Мрежата е в нейната всеобхватност. Достъпът за всеки, независимо от неговото състояние, е ключов аспект за това.“ Значителна част от масовите уебсайтове са с толкова силно присъствие на мултимедийно и графично съдържание, че са на практика неизползваеми от повечето помощни устройства от типа на екранни четци, гласови браузъри и Брайлови преводачи. Създаването на достъпно съдържание не означава, че трябва да се откажете от по-интересните и предизвикателни възможности на съвременния уебдизайн. Много от препоръките за разработване на леснодостъпно съдържане се съчетават съвсем естествено със създаването на добър дизайн. Към момента съществуват два наръчника по темата, които са на разположение на уебдизайнерите. „Инициативата за достъпност в Уеб“ на W3C издава „Насоки за достъпно уебсъдържание“ и скоро ще убликува неговото второ издание. Правителството на САЩ имат собствен наръчник.

3.16. Съобразете дизайна с компютърните монитори: компютърния монитор, на който ще се изобразяват вашите страници, е много различен от печатните медии. Когато планирате уебсайт, трябва да имате предвид следните разлики:

- Формата на компютърния екран. За разлика от печатните издания, които обикновено са портретно ориентирани, компютърния екран е пейзажно ориентиран – това означава, че е по-широк отколкото висок. Дизайнът на страницата ви трябва да съобразен с екрана, на който се показва и от който се чете.
- Докато хартията отразява светлината, компютърния екран излъчва светлина. Това влияе на избора на цветове и нивото на контраст. Страниците ви трябва да бъдат достатъчно контрастни за четене, но не чак толкова, че цветовете да разсейват потребителите от съдържанието. **Избягвайте светлите текстове на светъл фон и тъмните текстове на тъмен фон.**

Независимо колко добре сте планирали интерактивния си интерфейс на Вашия сайт, никога не можете да предвидите как точно как точно потребителите виждат Вашата работа, защото не знаете каква е разделителната способност на техните екрани. Разделителната способност на компютърните монитори е широчината и височината на екрана, измерени в пиксели. Повечето монитори могат да ползват поне два различни разделителни способности, като по-големите от тях имат и по-богат избор. Разделителната способност на екрана на потребителя е фактор, върху който Вие нямате никакъв контрол.

Разделителната способност е функция на възможностите на монитора и на видеокартата на компютъра. В момента най-популярната разделителна способност (традиционно се отбелязва с широчина x височина в пиксели) е 1024x768. Тя измести по-стария стандарт 800x600 пиксела. С поевтиняването на големи монитори някои потребители избират дори по-висока разделителна способност – 1280x1024 и нагоре, което им позволява да виждат повече съдържание на екрана в даден момент. Като уебдизайнер, Вие трябва да се съобразите и с потребителите с по-стари компютри, които ползват разделителна способност от 800x600. Въпреки че повечето сайтове променят дизайна си според модела 1024x768, много други все още се придържат към 800x600.

Публикуването на уеб сайта в Интернет е последният етап в процеса на разработването му, но реално съзнателната работа не приключва с това. Необходимо е да се положат усилия за реклама и популяризиране на сайта, за анализиране на процесите в използването му, за актуализиране на информацията в него и на базата на всичко това, да се направи оценка на реалната приложимост на сайта.

ВНИМАНИЕ: За да се проектира Web сайта трябва да се вземат под внимание някои важни черти на:

*Структурата

*Общият дизайн

*Графичният изглед

*Навигацията

*Текста

*Фреймовете

3.17. За структурата на сайта и страниците:

Уебсайтът представлява съвкупност от уебстраници. Тази съвкупност не е маханична. Добрата логика на организация на страниците трябва да ги обединява, от една страна, по съдържание, а от друга – графично. Сайтът има йерархични нива, чиито вътрешни връзки е необходимо да се обмислят предварително и да се зложат още при дизайна. Важно е да се изяснят структурата и архитектурата на страницата и да се създаде формат за всяко отделно ниво. Употребата например на единен стил в обръщението към посетителите (на текстово ниво), на сходни графични детайли или еднакви филтри при обработката на снимковия материал (на ниво визия), създава усещането за цялост, прави сайта консистентен, комуникативен, стилиен, приятен за навигация.

Подготвянето на основната структура на сайта изисква много добър баланс между творческия и аналитичния подход. Необходимо е да се организира информацията и да се определи как тя ще се появи на екрана на компютъра.

Интуитивно най-разбираема за хората е пирамидалната структура, при която има едно входно централно ниво, което води към няколко поднива, препращащи потребителя още „по-надълбоко“ в сайта. Става въпрос за създаване на една логична и лесна за разбиране йерархична структура, по която потребителят да се движи с лекота от външните нива (съдържащи по-обобщена информация) към по-вътрешните (където са конкретните данни) и обратно. Същевременно, за да се спести ненужно движение по сайта, се създават и преки връзки между нивата.

Най-добри резултати се получават, когато се подготви блок-схема, на която страниците се изобразяват като кутийки, а връзките се показват с линии, които ги съединяват. В горната част са главните страници, а в долната – вътрешните. Между страниците в сайта съществуват множество връзки, невинаги подчинени на йерархичния принцип. Не е необходимо всички те да бъдат показани – изобразяват се само основните и важните направления.

НАЧАЛНА СТРАНИЦА

Основен елемент в структурата е началната страница (home page), която представлява своеобразен „център“ на информацията в сайта. Тя в някои случаи може да бъде предхождана от splash screen –

-нулева страница, чиято цел е да впечатли посетителя, да го въведе в тематиката на сайта, и която води единствено към началната. От една страна, потребителите на Интернет обикновено са онлайн не за да се впечатляват, а за да ползват информация, и в този смисъл наличието на нулева страница независимо от нейните качества трябва да бъде много добре обмислено. От друга страна, все повече хора „сърфират“ за удоволствие, а не за да търсят определена информация. Напоследък се наблюдава тенденция на увеличаване броя на нулевите страници, тъй като те увеличават интереса на посетителя.

Създаването на блок-схема е етапът, в който от многото възможни и същевременно добри варианти ние избираме само един. Именно това е моментът, в който интуитивно се вземат множество решения, които придават определена насока в развитието на проекта.

В зависимост от сайта може да се наложи обособяването на няколко основни раздела, предлагащи твърде различно по вид съдържание, които биха изисквали създаването на няколко различни входящи страници. В този случай се прави т. нар. „метавходяща страница“, която да препраща посетителя към входа на интересувания го раздел. Това се определя от информацията. Задачата на дизайнера е да направи единен мотив за различните части и по този начин да ги обедини идейно, като в същото време всеки раздел трябва да запази своята идентичност.

Началната страница е разпределително място аз целия сайт. От нея има връзки към основните вътрешни страници (рубрики) и се дава акцент на някои конкретни страници по-надолу в йерархията.

НАВИГАЦИЯ

От навигацията на страниците зависи дали съдържанието им ще бъде достъпно за потребителите. Основна задача е намирането на необходимата информация да се направи възможно най-лесно за потребителите. Необходимо е включването на „навигационната лента“, „меню“ – еднакво за всички страници от сайта. В него се съдържат връзките към основните рубрики и началната страница. Желателно е да има индикация за позицията в йерархията на текущата страница. При подготвянето на навигацията ключови са следните два въпроса: „Интуитивно ли намират новите посетители информацията, която ги интересува, т. е. лесно достъпна ли е тя?“ и „Удобен ли е сайтът (навигацията му)“ за редовните посетители, които обикновено се интересуват от информацията на вътрешните нива?

Желателно е съдържанието на сайта да бъде променяно постоянно, за да знаят потребителите че при всяко следващо посещение ще намерят нещо ново, и това да ги накара да се връщат отново и отново. Още при планирането на сайта е необходимо да бъдат заложени възможностите за бъдещи промени. Трябва да се мисли глобално, в категории, да се създадат основните принципи на сайта – гъвкава структура, податлива на промени. Добре е графичният дизайн също да позволява лесното добавяне на нови рубрики и елементи.

Съдържание на сайт: Съдържанието е сърцето на уеб сайта. Дали той ще бъде полезен и интересен за посетителите зависи преди всичко от съдържанието, което предлага – текстове, снимки, допълнителна информация, интерактивни приложения, мултимедийни формати и т. н.

Създаването на релевантно и оригинално съдържание има ключова роля за успеха на сайта. Когато не е възможно тази задача да бъде възложена на външен доставчик (рекламна или уеб агенция, PR специалист), тя остава основен ангажимент на компанията, собственик на сайта. В практиката този етап често се оказва подценен от възложителя – в най-добрия случай сайтът се запълва с „нещо, колкото да има“, а в най-лошия – забавя старта на иначе готовия сайт. За да може сайтът да изпълни максимално ефективно поставените му задачи, е важно към съдържанието му да се подходи отговорно.

В началото на работата по сайта се прави проучване какво съдържание би било интересно и полезно за целевите публики, какво в повече може да предложи новият сайт в сравнение с по-стари такива или спрямо стандартното за неговия тип. След това се търсят пресечните точки между важното за аудиторията съдържание и целите на самия сайт – резултатът е списък с елементи, които искаме новият сайт да съдържа. И така стигаме до първото сериозно предизвикателство в създаването на сайта – осигуряване на съдържанието.

Най-лесно е в сайта директно да се прехвърли информацията, с която така или иначе вече разполагаме. За един корпоративен сайт например често се адаптира корпоративният профил, използван в брошурите и презентациите, разделът с информация за медиите се попълва с готовия архив на новините и т. н. За малкия или нов бизнес, който не разполага с PR и маркетингов отдел и още по-малко – с подкрепата на професионален копирайтър, подготвянето на съдържание за сайта е трудна задача. Затова е по-добре желанията и амбициите по отношение на съдържанието да бъдат съобразени с възможностите на екипа. Уеб сайтът е динамична структура и спокойно може да бъде обогатяван и допълван „на живо“.

Обемът и типът на информацията, с която разполагаме, може да наложи промяна в концепцията за съдържанието на сайта. Когато цялото желано съдържание не може да се осигури навреме, обемът и оттам структурата на сайта се променят. Все пак е добре, когато се постигне съгласие, че е важно да присъства даден тип информация – например видеоразходка в хотела или рубрика статии с полезни съвети – да се предвиди в плана на сайта и да бъде добавена на следващ етап.

Независимо, че адаптирането на готовата информация може да бъде достатъчно за първия етап от живота на сайта, все по-често виждаме разработване на оригинално съдържание, предназначено специално за уеб. Това могат да бъдат статии (например за корпоративния блок), интерактивни сравнителни таблици за продукти, видео и анимации, илюстриращи различни процеси, онлайн инструменти като калкулатори, игри, различни приложения. Това именно е и правилната практика, гарантираща по-добър сайт като цяло.

Естествено, тук се изисква допълнителна инвестиция на ресурси – хора, време, знания, техника. Затова и в момента тази практика все още не е широко разпространена извън онлайн медиите, за които пък е определяща.

Текстово съдържание

Текстът е най-широко разпространеният тип съдържание в уеб. Основната част от съдържанието на повечето сайтове е текст. Разбира се, има и силно визуални сайтове, където текстът е на втори план, но те са по-скоро изключения от общото правило, доколкото се отнасят до специфични сфери – дизайн, фотография, визуални изкуства. Ето защо текстът носи сериозна отговорност за цялостното впечатление от сайта и за цялостното му представяне относно поставените му задачи. В този смисъл е важно да се имат предвид спецификите на текста за уеб.

Текстът за уеб е:

Кратък и стегнат. Често само три изречения в една уеб страница вършат по-добра работа от три стандартни страници. Ако някой има нужда от допълнителна информация, дайте му възможност да си я поиска и му я осигурите.

Информативен. Понякога под формата на инструкции – започвайте с най-важното, със заключението, повечето посетители са дошли в сайта, за да свършат работа, и нямат време за дълги обяснения и прелюдии.

Лесен за сканиране с един поглед. Използва се подчертаване на важните думи, къси параграфи, отстъпи при изброяване.

Богати хипервръзки. Линковете като основна характеристика на уеб са важни, в т. ч. и поставяне на множество връзки към други страници в сайта, документи за даунлоад или външни препратки, когато е необходимо да се даде по-широко обяснение или допълнителна информация.

С минимално форматиране. За по-лесно възприемане от екран следва да се използват основно получер, несерифни и системни шрифтове, изчистени и консистентни, подравняването от двете страни на текста, както в традиционен документ не е задължително.

Планиране на сайт: Сайтът е съвкупност от разположени на един сървър свързани с помощта на хипервръзки уеб страници.

Реализацията на сайта се предшества от планиране, което включва следните стъпки:

- ✓ Изясняване целта на сайта;
- ✓ Анализ на аудиторията;
- ✓ Планиране на началната страница;
- ✓ Работа върху съдържанието на страниците.

Изясняване на целта на сайта – Всяка задача има цел. Например – представяне на организация, обучение, реклама, галерия, електронен магазин и др. За да се изясни целта на сайта, се провеждат разговори с организацията / лицето, което поръчва сайта.

Анализ на аудиторията

Тук се изяснява:

- ✓ Кой ще посещава най-често този сайт – каква е аудиторията;
- ✓ Какви са основните характеристики на аудиторията – технически компетенции, интереси, използван език/езици и др.;
- ✓ Какво очаква аудиторията, каква е нейната нагласа – предварителни интервюта с потенциални потребители;
- ✓ Как създаденият сайт се възприема от аудиторията – предварително тестване от потенциални потребители, преди сайтът да се публикува. Въз основа на резултатите може да се направят промени.

Планиране на началната страница (Home Page)

- ✓ Обикновено е първа страница;
- ✓ Отразява целта на сайта и очакванията на потребителя;
- ✓ Съдържа хипервръзки към важни страници от сайта;
- ✓ Информацията – на един екран (без скролинг);
- ✓ Ясна комуникация – валидно за всяка страница и особено за началната;

АРХИТЕКТУРА НА САЙТА

Как ще бъде подреден сайтът? Каква ще е навигацията? На колко нива ще бъде разположена информацията? Това са само част от въпросите, които трябва да си зададете, за да изясните подреждането на съдържанието на вашия сайт, неговата информационна архитектура и цялостен изглед.

Когато блок-схемата е завършена и одобрена от клиента, се преминава към планиране на основния дизайн, този началната и вътрешните страници на сайта. Дизайнът на началната страница е много важен. Статистиката сочи, че повечето от две трети от посетителите на даден сайт виждат само нея и напускат. Следователно от това, как е изработена началната страница, зависи дали посетителят въобще ще продължи да разглежда сайта. Следователно тя трябва да грабне вниманието, а вътрешните страници – да го задържат. Тук в пълна сила важи поговорката “По дрехите посрещат, по ума изпращат!” – добрия дизайн на входната страница е решаващ за успеха на даден сайт.

Към съдържанието на началната страница могат да бъдат включени рубрики „новини“ или „актуална информация“ – това подсказва, че сайтът се поддържа и провокира посетителя да се върне отново.

Съдържанието на началната страница трябва да бъде динамично, различно от познатото от други сайтове, добре оформено. Това може да се постигне с различни похвати – от една страна, интересна информация, от друга – пленителен дизайн, добър подбор на цветове, шрифтове, графики или снимки и т. н. Посетителят трябва да получи информация от началната страница за името на сайта, за идеята, за целта на сайта, да прочете встъпителни думи, датата на последното актуализиране и т. н. Необходимо е да се добави информация за контакти и обратна връзка с авторите на сайта, осъществяване на възможност за оценяване съдържанието и представянето на сайта. Желателно е съдържанието на първата страница да е разположено на един екран, за да може да бъде обхванато с един поглед, без да се налага придвижване. Това означава, че тя трябва да е съобразена с разделителната способност и на по-голямата част на мониторите, която днес е 800x600.

След като е известно какво искаме да съдържа сайтът, е време да се поработи по неговата структура и информационна архитектура, а от тях да излязат и важни решения относно дизайна на сайта.

Цялото съдържание следва да бъде „налято“ в определена структура, която е добре да бъде логична и естествена. Най-често тя е йерархична и непомня на структурата в дадена организация. Именно това е начинът за изграждане и организиране на големи по обем уеб проекти. С подобна структура информацията и сайтът като цяло ще бъдат най-добре възприети от потребителя.

В някои случаи, макар и по-рядко, е възможно да имаме нужда не от йерархична структура, а от подреждане, при което се върви в строга последователност, както при слайдшоу.

В идеалния случай уеб сайтът има структура на паяжина, при която всяко нещо е свързано с множество други. Тази структура е най-подходяща за малки сайтове или сайтове от типа онлайн списания или образователни ресурси.

Подобна схема на свързаности следва да се направи още в началото на работата по всеки уеб проект. Добре организираният уеб сайт поднася цялата си информация само на няколко клика разстояние. Колкото по-лесно се използва един сайт, толкова по-голяма е вероятността потребителят да остане по-дълго на него. За създаване на сайт с

добра структура е необходимо страниците да не бъдат на повече от две нива дълбочина от началната страница. Това улеснява не само потребителите, които бързо се ориентират и намират цялото им необходимо съдържание, но и търсещите машини, които индексират съдържанието. Архитектурата на уеб сайта включва технически, естетически и функционален поглед върху сайта като цяло, с фокус към потребителя и неговите изисквания и очаквания. Понятието включва едновременно ползваемост, съдържание, интерактивност, информационна архитектура и дизайн. Информационната архитектура на сайта определя как ще бъде организирана информацията на сайта – страници, документи, данни. Тук следва да се изясни също конвенцията за файловете имена в сайта (важни за оптимизацията за търсачки), поместването на файловете за сайта в отделни папки и тяхната структура (например всички графични файлове да се намират в папка, „/images/“). Както и конвенцията за линковете (например всички външни линкове да се отварят в нов прозорец). За да се изгради сайтът, е необходимо да се отговори на множество въпроси. Коя е основната информация, която следва да бъде изведена на първа страница? Това, особено при по-големи сайтове, е наистина труден въпрос. От гледна точка на потребителя, а и на търсещите машини, добре е на първа страница да бъдат изнесени основните акценти и най-важните неща, които имате да кажете. Например вместо общ текст за добре дошли да се изнесат най-важните продукти и услуги на фирмата, текущи промоции и новини. Кое е съдържанието, което ще е полезно на различните публики на сайта, как то е взаимно свързано и как да се подреди оптимално? Това е основната част от сайта, която следва да е добре подредена, структурирана и поднесена. Кои други задължителни елементи следва да бъдат включени в сайта? Тук влизат страници от типа „Карта на сайта“, „Отговорности“, „Авторски права“, „Защита и сигурност на информацията“ и други.

Още за Структурата:

1. Включете потребители, за да ви помогнат да дефинирате потребителско-ориентиран дизайн на вашия сайт.
2. Създайте диаграма, показваща зависимостите между отделните елементи в сайта.
3. Използвайте хоризонтална, а не вертикална йерархия, тъй като така потребителят ще може по-лесно да се върне в началото, ако се загуби.

Общ дизайн

4. Проектирайте сайта така, че да бъде достъпен за възможно най-много потребители. За целта осигурете поддръжка от повече браузъри.
5. Включете начална страница в сайта, която да съдържа въведение в съдържанието.
6. Осигурете всяка информация да бъде на три стъпки от главната страница - иначе потребителите могат да изпуснат важна информация.
7. Всяка страница от сайта трябва да бъде достъпна от главната - директно или индиректно.
8. Ограничете големината на страниците в сайта до екран и половина - иначе потребителят ще загуби контекста на информацията.
9. Заглавието на всяка страница трябва да бъде уникално и подсказващо съдържанието ѝ.
10. Включете информация за контакт, така че потребителите да могат да изкажат мнението си.
11. Направете адреса на сайта колкото е възможно по-къс. за да се избегнат грешки при писане
12. Предоставете търсене в сайта, за да подобрите достъпа до информация, особено в случаите на голям и сложен сайт.

Текст

13. Слагайте подходящи заглавия.
14. Разположете важната информация в началото, така че потребителят да може да я прочете веднага и да продължи за повече детайли.
15. Използвайте връзки, за да добавите по-подробна информация.
16. Използвайте внимателно цветовете, за да може текста да се чете лесно на избрания фон. Убедете се, че има добър контраст между текста и фона. Най-добре е фонът да има цвят с ниска интензивност, за да не дразни очите.
17. Избягвайте да използвате цветен и подчертан текст, за да не си помисли потребителя, че това е връзка.
18. Използвайте bold текст, за да наблегнете на по-важна дума или фраза, но не го използвайте за цяло изречение, тъй като става трудно за четене.
19. Използвайте Times New Roman и Arial шрифтовете
20. Не използвайте повече от 3 различни шрифта в сайта, тъй като текста ще стане труден за четене.
21. Използвайте ляво подравнен текст, а не дясно подравнен или центриран.
22. Проверете четливостта чрез използване на различни монитори и промяна на различни фактори като шрифт и големина на шрифта, цветовете контраст.

3.18. Направете дизайна отзивчив (responsive web design) и адаптивен уебдизайн - същност и спецификации.

Отзивчив уеб дизайн (Responsive Web Design или RWD; „responsive“ от английски – отзивчив), представлява вид стилизиране на уеб страници, целящо да осигури оптималната ѝ визуализация и улеснено взаимодействие с нея, което да е подходящо за различни устройства (мобилни телефони, таблети, настолни компютри и др.) и техните различни резолюции на екрана. Целта на това адаптиране е да се минимизира нуждата от преоразмеряване и скролване.

Уебсайт, изработен чрез Отзивчив дизайн, адаптира съдържанието си към средата, на която се визуализира чрез използването на гъвкави, базирани на пропорции решетки (fluid, proportion-based grids), гъвкави изображения (flexible images) и CSS3 media queries, по следните начини:

- Концепцията за „гъвкава“ или „течна“ решетка (fluid grid) изисква оразмеряването на елементите на уебстраницата да бъде в относителни единици като проценти, а не в абсолютни като пиксели (pixels) или точки (points);
- „Гъвкавите“ изображения (flexible images) също се оразмеряват в относителни единици за да се предотврати показването им извън елемента, в който се съдържат;
- Медийните заявки (Media queries) позволяват на уебсайта да използва различни стилове (CSS) според характеристиките и типа на устройството, на което се визуализира, като ширината на брауъра е най-често използваната характеристика.

Отзивчивия уеб дизайн (Responsive web design) е все по-използван и придобива все по-голямо значение, тъй като количеството на мобилния трафик възлиза на повече от половината от общия интернет трафик. Тази тенденция е толкова разпространена, че Google започна повишаване на рейтингите на сайтове, които са с Отзивчив уеб дизайн (Responsive web design).

Предимства

- по-голяма достъпност от различни уеб поддържащи устройства;

- изработва се дизайн наподобяващ основния (десктоп дизайн), но за устройства с различни резолюции. Дизайнерите могат да използват един шаблон за всички устройства и просто използвайки CSS да се определи как съдържанието да се нагоди към различните размери екрани;
- по-добро индексране от Google (SEO оптимизация);
- разработва се само един сайт, а не няколко различни според резолюцията на устройството;
- добро дългосрочно решение, от гледна точка бързо развиващия се и разнообразен пазар на мобилни устройства;
- удобно и добре изглеждащо съдържание.

Недостатъци

Основен недостатък е скоростта на зареждане. Ако сайтът има видеоматериали и слайдове, то става много труден за нормално ползване. Много от недостатъците се проявяват при сайтове с големи размери. При някои от най-известните сайтове за електронна търговия от този тип се наблюдава голямо утежняване на началните страници. Въпреки че ненужните тагове са скрити за по-малките мобилни устройства, браузърите зареждат целия сайт на заден план и това прави отзивчивия дизайн излишен в този случай. Определящ фактор е и бюджетът. При отзивчивото оформление е нужно пълно пренаписване на кода, което обикновено излиза скъпо. Другият недостатък е при бавен мобилен **интернет** или слабо устройство. Зареждането дори на сайтове с малки размери се превръща с досадно чакане. Фазата на проектиране и тестване може да се окаже доста продължителна, понеже е трудно да се изработи сайт, който да бъде удобен за визуализиране на всички размери дисплеи. Има отзивчиви сайтове, които на екрани с ниска резолюция приличат на парчета от пъзел, които не си подхождат напълно.

При отзивчивия уеб дизайн се създава гъвкаво оформление на сайта за визуализация от всякакво устройство, свързано с Интернет. Възпроизвежда се пълното му кодиране, дори някои от неговите по-големи елементи да не се показват на брауъра или изображенията с големи размери да се смаляват автоматически на екрана. Всичко това намалява скоростта на зареждане на сайта.

При адаптивния уеб дизайн са налични няколко оформления на сайта с определени фиксирани размери за различните устройства:

- 320, 360 пиксела за смартфони;
- 768 – за планшети;
- 960 – за компютри.

3.19. Интерактивен интерфейс в уебстраница.

Основен инструмент за привличане и задържане на вниманието на потребителя е използването и „вграждането“ на **интерактивност в web**. Тъй като самата дума **интерактивност** се използва в много различни ситуации и за обозначаване на множество различни неща, е необходимо да определим какво ще се разбира под интерактивност в web design, а именно: **Интерактивна ще бъде наричана тази web страница, която реагира по определен начин на действията на потребителя.**

Самия HTML не създава предпоставки за интерактивност, с негова помощ може да се бъде изградено единствено статично съдържание, което да се ползва подобно на книгите и други печатни материали. Възможността за хипервръзки не може да се разглежда като интерактивност – това е същностно определена характеристика на Интернет.

Логично е да се предположи, че за да започне web страницата да реагира, е необходимо тя да бъде програмирана, т. е. в нея да бъдат заложени някакви команди и условията, при които те да бъдат изпълнявани. Може да се използва програмен код, който работи в брауъра при потребителя (client side script). Това са така наречените скриптове и при тях действието се извършва от компютъра на потребителя и обикновено без допълнителен трафик и обмен на данни със сървъра в Интернет. Целия необходим код бива предаден заедно с основното съдържание на web страницата. Той най-често е на езика Java Script и по-рядко на VB Script.

3.20. Мултимедия в уеб страница

Съвременните уеб приложения са изпълнени с атрактивни елементи, допълващи визията на уеб сайта. Такъв тип елементи, освен текст и изображения, могат да бъдат и видео, аудио, плъгини, динамичен образ, динамични галерии, анимация, флаш елементи.

- **SHOCKWAVE FLASH**

Елементите на Flash са нещо като добрата алтернатива на аpletите на Java. Те също представляват елементи, които се вграждат в уебстраниците, но имат предимството да работят много бързо и много добре.

При технологията Flash, за разлика от класическото въвеждане на графични елементи, се работи предимно с векторна графика, което от своя страна позволява направата на динамично променящи се обекти – възможност, която липсва както при стандартния HTML, така и при неговите подобрения с JavaScript, CSS и т. н. Елементите на Flash на практика нямат възможност да комуникират и да влияят на останалите елементи на уебстраницата – тяхното действие се затваря изцяло в рамките на вградения елемент с изключение на факта, че в тях може да има връзки. За сметка на това, те предоставят големи възможности за движение на образи, преплитане, смесване, затъмняване, съчетано със звук и т. н. Употребата на Flash е подходяща при развлекателни сайтове, или такива, целящи да привлекат вниманието. Водят се множество спорове „За“ и „Против“ използването на Flash, особено за създаването на цели сайтове само с тази технология. Вярно е, че с Flash се създават много „живи“ *интерактивни сайтове*, но понякога те стават обемни и тежки за зареждане. За множеството потребители с по-стари версии брауъри, които не поддържат тази технология, се предлага възможност да изтеглят от мрежата малка програма – Plug-in, с което да направят възможно стартирането на Flash.

Елементите Flash се създават с помощта на специализиран софтуер на фирмата Macromedia. Повече информация можете да намерите на адрес www.macromedia.com.

Препоръки: *Зърване на първия екран със съдържанието*

Заглавната страница е лицето на сайта и затова трябва да бъде проектирана по начин, различаващ се от този на останалите страници. Разбира се, заглавната и вътрешните страници трябва да притежават един и същи стил, но има определени различия. В заглавната страница например не е необходимо да има бутон Home, тъй като е доста неприятно да натискаш бутон, който не води към друга страница. Освен това Home Интерактивната уеб страница, обикновено има по-голямо лого и името на компанията

или на сайта са разположени на видно място. Непосредствената цел на всяка заглавна страница е да отговори на въпросите „Къде се намирам?“ и „За какво е предназначен този сайт?“ – и двата изискват ясна и разширена версия на името. Друг не по-малко важен въпрос, на който трябва да отговаря заглавната страница е „Какво прави този сайт?“. И все пак най-важната функция на заглавната страница е да служи като отправна точка към схемата за навигация – **интерактивния интерфейс**, който предлага сайта.

Към заглавната страница са прикрепени още няколко вътрешни. На всяка от тях в горен ляв ъгъл или долен ляв – препоръчвам да се прави връзка към заглавната. Дори често пъти е най-добре, самото лого да препраща към заглавната страница. Имам предвид, логото на уеб сайта да присъства като неминуема част във всяка вътрешна, но от него да се препраща към заглавната страница, бързо, удобно, приятно за потребителя, както посочих и по-горе.

Най-важния аспект по отношение времето за отговор, е кога потребителят започва да вижда екран с полезна информация. Не е от голямо значение фактът, че пълната страница с всичките ѝ илюстрации се зарежда по-дълго, стига потребителят да е в състояние да започне бързо някаква информация. Напътствията за бързо първоначално зареждане са: -

- **Заглавия на страниците обикновено се пишат с цел да бъдат намерени от хората.** За хората, използващи машини за търсене, вашият сайт съществува единствено като заглавие на страница, показано в панела с резултатите от търсенето. В HTML всяка страница има заглавие, определено в заглавната ѝ част (header). Важно е да определите добри заглавия за страниците си, тъй като именно заглавията се използват като основна препратка към тях. Заглавието дори може да е от една дума максимум до шест. Или разполагате с 40 до 60 символа или знака да обясните на хората какво ще намерят на вашата интерактивна уеб страница.
- **Анимацията не е най-надеждното средство за привличане вниманието на потребителя върху уеб сайта.** Анимацията може да има положителен ефект за интерактивния интерфейс в уеб страницата, ако целта ѝ е да предупреди потребителя за обновена информация, но след това да без повече разсейване, да го остави да прочете текста на спокойствие. По принцип разбира се, страницата трябва да бъде проектирана да показва най-важната информация в горната си част, което автоматично освобождава другите страници от присъствието на анимация.
- **Горната част на страницата да е смислена,** дори когато изображенията все още не са заредени (това означава повече текст и по-малко изображения);
- **Използвайте текстови атрибути alt за изображенията,** така че потребителите да разберат за какво служат съответните изображения, докато те се възпроизвеждат. Повечето браузъри показват алтернативния текст на ALT преди изображението да се зареди, на мястото, където то трябва да се появи.
- **Браузърът трябва бързо да очертае горната част на изображението.** Той може да направи това обаче единствено ако разполага с пълната необходима информация за оформлението. Не забравяйте да включите атрибутите на width и height към всички изображения и колони на таблици.
- **Сложните таблици често отнемат повече време** за възпроизвеждане, затова за да намалите сложността на таблиците си, разделете информацията в няколко таблици. По-специално, най-горната таблица в страницата трябва да бъде проста и да се възпроизвежда бързо.

- **Препоръки за заглавия на връзки** – целта на едно заглавие на връзка е да помогне на потребителя да предположи какво ще се случи, ако последва връзката. Подходяща информация за включване в заглавие на връзка би била: името на сайта, към който води връзката (ако е различен от текущия сайт), името на под-сайта, към който води връзката (ако остават в същия сайт, но се преместват в друга негова част); Допълнителни подробности относно вида информация, съдържаща се в посочената страница, както и какво отношение има тя към текста – котва и съдържанието на текущата страница. Заглавията на връзките трябва да са по-малки от 80 символа и по принцип рядко да надвишава 60 знака. Кратките заглавия на връзка са по-добрия вариант.
- **В никакъв случай обаче не препоръчително да се използват фреймове**, големи изображения и видео, които по-скоро забавят скоростта на зареждане на уеб страниците...
- **Пишете кратко**, изследванията показват, че четенето на компютърен екран е с около 25% по-бавно от четенето на хартия. Дори потребителите, които не познават тези изследвания на човешките фактори, обикновено казват, че не им е приятно да четат онлайн текст. Изводът е, че хората не искат да четат големи количества текст от компютърния екран. Следователно, би трябвало да пишете около 50% по-малко текст, мислейки за добрия комфорт на потребителя. Също така, те не обичат да превъртат екрана, затова се старайте да правите кратки страници.
- **Видео** - Поради ограниченията на пропускателната способност, настоящото използване на видео в Web трябва да бъде намалено. Евентуално видеото ще се използва повече след време, към момента бавят скоростта на уеб страницата. Ето как се вгражда обикновен видеофайл: `<object data="video.avi" type="video/avi">
<embed width="240 height="196" controller="false" src="video.avi">
bgcolor="#000000" border="0">
</embed></object>`

Филми и аудиофайлове

Когато планирате уебсайт, видеото или музикалният откъс може да е елементът, който добавя привлекателност или допълва посланието ви. Преди да добави този вид медия, дизайнерът трябва да вземе предвид размера на файла. Видео и музикалните файлове може да забавят достъпа до един сайт и да доведат до излизане на зрителя от него. Вместо да вграждате тези файлове в уебсайта, предложете връзка (link) към желания видео или музикален файл, която да елиминира забавянето.

Видеоформатите включват Windows Media (WM V), AVI, филмови файлове (MPEG или MPG), QuickTime (QT и MOV) и Flash Video Files (FLV). Когато видеото е готово да бъде записано, изберете Web video. Това ще компресира видеофайла до 320 x 240 формат.

Аудиофайловете формати включват WAV, AIF, MP3, Real Audio (RA, RAM, или RPM) и QuickTime (QT или QTM). Тези файлове може да бъдат възпроизвеждани от повечето браузъри. Ако имате нужда да конвертирате един аудиофайл, за да бъде възпроизведен от сайта ви, следният сайт предоставя лесен инструмент за конвертиране. Той също така конвертира други видове медия като изображения, видео и документи.

Съществуват много добри онлайн конвертори, но с ограничения, за това може да свалите подходящата за целта програма.

- **Аудио** – Основното предимство на звука е, че той предоставя канал, който е отделен от канала на изображенията. Говорът може да се използва за предоставяне на коментар или помощ, без да се засенчва информацията на екрана. Аудиото е подходящо да се използва за предоставяне на усещането за място или настройка. Звукът с добро качество неминуемо подсилва преживяването на потребител, така че инвестицията в звукова продукция с професионално качество си струва. Звуков файл може да се вгради с кодът:


```
<embed src="v.mid" hidden="true"width"0"
height="0"
autostart="true"loop="true"></embed>
<noembed><bgsound src="v.mid">.
autostart="true" loop="infinite"></noembed>
```

Flash технологията се появява през 1997 г. и предизвиква истински бум в развитието на web дизайна и Интернет. Чрез нея се осъществява мечтата на много web дизайнери за повече **динамика в съдържанието на уеб страниците**. Flash позволява, както наличието на растерна, така и на векторна графика; Тази технология дава възможност за „меки“ движения с преливане на форми и цветове, възможност да се добави звук, който да се синхронизира с образа, да се следят действията на потребителя и да се реагира спрямо тях.

Въртящи се емблеми, скучни „зареждащи“ екрани и безкрайни, нежелани „въведения“ изградиха лошо име на Flash сред потребителите. Но глупавата злоупотреба с мощта на новия инструмент в крайна сметка даде възможност за създаването на изтънчени потребителски интерфейси, създадени от престижни магазини. По-малко талантиливи и по-малко иновативни агенции се надяваха да се възползват от Flash, като често създаваха доста по-слаби сайтове, но не можете да обвинявате чука и пироните за некачествената дърводелска изработка. Flash завладяваше богато приложно пространство по същия начин, по който браузърът на Microsoft превземаше позициите на Netscape.

Извод: В способни ръце Flash улеснява създаването на **богати интерактивни интерфейси**, които биха били трудни за емулиране с използването на стандартно маркиране, CSS, SVG (Scalable Vector Graphics) и др.

Съществува класификация на техниките за тестване на софтуера, която може да подпомогне програмистите при опитите за ориентиране в тази противоречива и не много добре разработена и изяснена област на процеса за разработване на софтуер. Като пример за добра поддръжка е тестването на софтуерни продукти е дискутиран Фау моделът (V-model). Подходите за аналитично осигуряване на качеството на системата са ориентирани към структури тестове, известни още като white-box тестове. Представени са двата основни вида структурни тестове – ориентирани към контролния поток и ориентирани към потока от данни тестове. Тестването на функционалността е един от съществените методи за осигуряване на качество на софтуер.

В уебстраниците, макар и рядко, се срещат вградени видео или аудиообрази. И двете възможности изискват сравнително големи обеми от информация, които трябва да се предадат по Мрежата и това е основната причина, поради която тяхната употреба се избягва. Звукът може да се включи под формата на MPEG-, WAV – или MIDI-файлове и да бъде или за фон, или с помощта на Java Script да се синхронизира с действия на потребителя. Просвирването на звуковите файлове става най-често с помощта на

програмите Windows Media Player, Real Player, Win amp, Flash Player. То може да се осъществи по два начина:

- След като файлът се зареди напълно;
- Онлайн, т. н. стрийминг – технология, с помощта на която звуков файл се възпроизвежда, докато се зарежда от Интернет. Проблем в случая е скоростта, която често е бавна и това води до накъсване на звука в някои случаи. Стандартът за разпространение на видеофайлове е AVI. Техния обем е доста голям, поради което те почти не се използват в България. Елементарни анимации могат да бъдат вградени в страниците под формата на GIF - анимации. Те съдържат няколко изображения, които се появяват последователно на едно и също място. Подготвянето на отделните кадри става със специализиран редактор, а включването на крайния резултат в HTML-код се извършва както при обикновените убизображения, с помощта на тага .

Ресурси за изображения

Графична колекция (Clip Art)

Microsoft Clip Art on Office Online

<http://office.microsoft.com/en-us/clipart/default.aspx>

Безплатни анимации <http://www.free-animations.co.uk>, 33B.ru

Безплатни бутони <http://www.freebuttons.com/>, [HTML Tutorial \(w3schools.com\)](http://www.w3schools.com/html/html_tutorial.asp)

Снимки

Microsoft Clip Art on Office Online

<http://office.microsoft.com/en-us/clipart/default.aspx>

Flickr

<http://www.flickr.com/>

MorgueFile

<http://www.morguefile.com/>

Google, Yahoo Images

<http://images.google.com/imehp?hl=en&tab=wi> <http://www.vahoo.com/>

Видео

YouTube

<http://youtube.com>

Online Video Gallery (Онлайн видеогалерия) <http://onlinevideogallery.com/>

Аудио

Microsoft Clip Art on Office Online

<http://office.microsoft.com/en-us/clipart/default.aspx>

Безплатни аудиоклипове <http://freeaudioclips.com/>

Free Play Music (Безплатна музика) <http://freeplaymusic.com>

Безплатни звукови файлове

http://www.wvomingwebdesign.com/files/pages/free_sound_files.html

3.21. Проблеми със съвместимостта на браузърите. Едно от основните предизвикателства пред използващите HTML дизайнерите на **интерактивен интерфейс** е да създадат страници, които да бъдат визуализирани правилно от различни браузъри. Всеки браузър притежава подпрограма, наречена **рендираща машина**, която интерпретира таговете в HTML файла и показва съответния резултат в браузъра. В различните браузъри логиката на интерпретация варира и резултатът при изобразяване на един HTML файл е потенциално различен. Като уебдизайнер Ви трябва задължително да тествате работата си под възможно най-голям брой браузъри, за да се уверите, че всичко се вижда по замисления от вас начин. Дори и на пръв поглед да Ви се струва, че Вашият код на HTML е съвместим с повечето браузъри, вероятно ще се изненадате колко различен може да бъде резултатът. За да осигурите максимална съвместимост на Вашите уебстраници с множеството браузъри, следвайте правилата:

Препоръки:

- Спазвайте стандартите на консорциума за уеб W3C
- Валидирайте кода си
- Познавайте аудиторията си
- Тествайте **интерактивния си интерфейс** и последващите страници с възможно повече браузъри

Повечето създатели на HTML не разполагат с информация за възрастта, типа или операционната система на браузърите, които ще ползват потребителите, когато разглеждат техния код. Изборът на конкретен браузър и неговата версия могат да варират значително в зависимост от редица фактори. Много организации и отделни потребители не са склонни да сменят софтуера си само защото е излязла по-нова версия. За други може да е невъзможно да преминат на нова версия, ако компютрите им са стари и не разполагат с достатъчно бърз процесор или голям твърд диск. Тестването с най-новите браузъри е добра идея, но също така е благоразумно да проверите работата си с по-стари браузъри, за да се уверите че максимален брой потребители ще могат да достигнат до Вашия **уебинтерфейс**.

Важна стъпка в изграждането на един сайт е познаването на неговата потенциална аудитория. Например, ако правите сайт на тема последни тенденции в програмните езици, той би привлякъл компютърни специалисти, така че можете да съобразите кода си с най-съвременни браузъри. Ако обаче сайтът Ви е насочен към широката публика, ще трябва да го тествате и със старите браузъри, за да се уверите, че страници Ви се изобразяват по предвидения от Вас начин както от нов, така и от по-стар браузър. Уверете се, че визуализирането на най-важното съдържание от сайта не зависи от нови технологии. По този начин то ще достигне и до потребителите, използващи по-стари браузъри. И накрая, ако страницата ще се използва във вътрешна мрежа и имате контрол върху софтуера, който ползват потребителите Ви, тогава бихте могли да се съобразявате само с един браузър.

Факт е, че излизат постоянно нови браузъри, но и много потребители все още ползват стари. По-новите браузъри поддържат привлекателни възможности като CSS, които в същото време не се поддържат от старите. Когато включвате подобни отскоро поддържани възможности в дизайна на страницата си, това може значително да повлияе на изобразяването ѝ от по-старите браузъри, тъй като те не могат да интерпретират правилно последните нововъведения. Съществуват също и някои фини разлики в поведението на браузърите при различните компютърни платформи.

Най-новите браузъри, поддържат много по-добре въведените от W3C стандарти. Компаниите производители на тези браузъри са открили, че и те, като част от обществото на уебразработчиците, имат полза от спазването на общите стандарти. Браузърите, които не се отклоняват от приетите норми, помагат за създаването на **по-добър визуален интерактивен интерфейс** и неразривна връзка с уебпотребителите.

3.22. Промоция на сайта

След като работата по сайта е напълно приключена, сайтът е качен и достъпен в Интернет, следва да започне неговото популяризиране и рекламиране във и извън Интернет. Това става по различни начини – включване в търсещи машини, обмяна на банери, включване в уебрингове и др. Данните за посещаемостта са важен индикатор за това какво се търси на сайта, как се осъществява „движението“ на потребителите и т. н. Статистиката е полезен източник на информация за развитие на сайта.

3.23. Включване в търсещи машини

Търсещите машини и порталните сайтове са два различни типа сайтове и е добре завършеният сайт да бъде регистриран във всички, както в регионален, така и в световен мащаб.

„Започнахме да индексирате цялата информация от Мрежата просто за удоволствие.“ Наречете го хоби или страст. Наречете го инстинкт. Но това не е бизнес, не се печели пари от това, може би в основата е възможността да се влияе светкавично на хилядите хора, които посещават сайта. Не се знае, че това е удоволствие и може да донесе успех. Основните български портали са:

www.dir.bg

www.gyvetch.bg

www.search.bg

www.alle.bg

portal.bol.bg

www.triada.bg

От скоро започна развитието на регионални портали.

www.bourgas.org

www.russe.net

www.ida.bg

Има сайтове, които извършват регистрацията едновременно в множество портали и търсачки. За да е успешно включването в търсачките, следва да се спазват няколко правила:

- МЕТА – таговете да са добре формулирани и достатъчно пълни;
- Рубриките, в които се регистрира сайтът, да са подбрани внимателно;
- Да се има предвид проблемът с кирилицата;
- Внимателно да се изписва адресът на сайта;

3.24. Банери

Банерите и обмяната на връзки са основен източник за трафик и средство за реклама на Мрежата. Интернет е истинско предизвикателство за въображението и творческите сили на хората, създаващи реклама. Новата медия предоставя възможност за чести промени. От друга страна (в сравнение с рекламите в другите медии например), изискванията

към качеството непрекъснато се изменят. Всичко това подсказва, че маркетингът онлайн е не само добро средство, но и сериозно предизвикателство.

Най-успешните сайтове, ще са тези, които мотивират хората само с един добър банер. Постепенно се наложиха няколко стандартни размера на банери:

- Малки – 88x38 или специално за България – 90x30;
- Големи: 400x40 или 468x60.

Освен по размера си банерите се различават и по това, дали са статични или съдържат анимация. Не винаги банерът с анимация е по-ефектен от статичният. Понякога раздвижените банери дори водят до обратен ефект – раздразнение и отблъскване на посетителя. Не е лесно да се създават свежи, запомнящи се банери, които същевременно да имат малък обем. Ето няколко съвета, спазването на които би осигурило успеха на един банер:

- Използвайте два до три ключови цвята;
- Имайте предвид, че част от хората са далтонисти, други използват черно-бели монитори – изберете цветовете внимателно;
- Създайте динамика с форми и цвят, дори при статичен банер;
- Уверете се, че посланието Ви е кратко, разбираемо и ясно;
- Използвайте символи, за които сте сигурни, че ще бъдат разбрани от посетителите.

В световен мащаб amazon.com е лидер с 360 уникални реклами, MSN е на второ място с 330 банера.

Освен, че можете да раздавате банери, които да се поставят на други сайтове, можете да поставяте такива на своя сайт. Обикновено се използва система „обмяна“. Въпреки всичко, бъдете убедени, че си заслужава да поставите съответния банер на сайта си. Обмислете не губи ли дизайнът на страниците ви с появата на банер. Разположението на банерите, особено на първа страница, е от стратегическо значение. Представете си един много атрактивен банер, който се зарежда бързо, значително преди останалата част от заглавната Ви страница. Това би могло да накара посетителите да не дочакат зареждането на страницата, а да щракне върху банера и да напусне сайта Ви преди да го е видял.

3.25. Рекламни електронни съобщения

Един от начините да се направи промоция на сайт е изпращането на електронни съобщения до множество потребители, които биха проявили интерес към него – т. нар. мейлингкампания, която има, разбира се, своя аналог и в реалния живот, с разпращането на рекламни писма и брошури. Този метод не е много популярен у нас, но напоследък все по-често се използва.

- Текстът на съобщението трябва да бъде кратък и ясен;
- Адресът на сайта и името на подателя се дават задължително – анонимните писма трудно предизвикват интерес;
- Не бива да се изпращат прикрепени файлове – с графика мейлът ще изглежда добре, но обемът му ще бъде по-голям, а при файлове с разширение .doc съществува риск да не бъде прочетен, поради възможността да пренася макровируси;
- Адресите на получателите не бива да се изброяват в полета. Добре е да се използва специализиран софтуер.

Към използването на мейлингкомпанията следва да се пристъпи внимателно. Рискът тя да предизвика обратен ефект е голям. Получаването на подобни електронни съобщения се нарича „спам“ и често се приема негативно, особено от „ветераните“ в Мрежата. И все пак (както и в реалния живот), има много потребители, които не биха имали нищо против да получат рекламно съобщение, информиращо ги за нов сайт, с промоция на нови услуги в Интернет или каквото и да било друго.

3.26. Реклама офлайн

Освен всички изброени по-горе методи за промоция онлайн на уебсайт е добре да се планира и проведе и неговата офлайнреклама, за да бъде той максимално ефективен. Най-малкото, с което една фирма трябва да започне да рекламира сайта си, е да постави неговия адрес на своите визитки, бланки и други печатни материали, които издава.

Все повече фирми правят промоция на уебсайтовете си, използвайки останалите медийни канали. По билбордове, брошури, рекламни карета във вестници и списания, радио и телевизия, и наред с други детайли се цитират и корпоративни адреси в Интернет. Присъствието в Мрежата е въпрос на престиж и е удобство както за потребителите, така и за фирмите, които благодарение на своя уебсайт спестяват време и средства.

Свободата на въображението, съчетано с правилното използване на познатите маркетингови тактики в конвенционалните рекламни компании, могат да предизвикат интерес към сайта както от страна на хората, които са ежедневно в Мрежата, така и на тези, които по-рядко ползват Интернет. За да се задържи този интерес, самият сайт трябва да е атрактивен, професионално направен, добре поддържан и актуален.

3.27. Данни за посещенията и статистика

За мнозина 6-7 цифрено число, гръмка анонсиращо, че показва броя на посетителите на сайта, и разположено на видно място на началната страница, е силно необходимо. Дали наистина е така? В епохата на прохождение на уебсайтовете, когато тяхното количество не беше голямо, както и това на сърфиращите в Мрежата, показването на броя на посещенията предизвикваше интерес. Днес ситуацията е различна. Статистиката за посещенията на сайта е важен индикатор за създателите на собствениците на сайта. От каква полза би била тази информация за тях? Ако сайтът наистина е добър и стойностен, то броят на посетителите няма никакво значение. Още по-малко значение може да има в случай, че не е стойностен. Например големите корпоративни сайтове не излагат на показ броя на посетителите му, но същевременно имат пълна статистика за тях, за техния „път“ из сайта, т. е. кои страници предпочитат, колко време са прекарвали на тях, каква е разделителната способност на монитора или версията на браузъра, и т. н. Всичко това се получава от т. нар. server log, който регистрира всяко повикване на страниците от даден сайт.

На практика от статистиката може да се научи от кои препратки от други сайтове са дошли посетителите и от кои страници са напуснали. Ако всичко това се анализира внимателно, *може да се подсказже в коя посока е необходимо да се работи, какво привлича или отблъсква посетителите.*

Съществува и т. нар. „независими броячи“, които дават данни за броя на посещенията. На базата на този тип броячи се правят класации с най-посещаваните за дадена страна сайтове. Такива безплатни броячи са ExtremeTracker (www.extreme-dm.com/tracking), WebCounter (counter.search.bg), Fast Counter

(www.fastcounter.com). Те предоставят информация за вида, броя на посещенията и т. н., и определят рейтинга на сайтовете.

4. РИСКОВЕ И ПРЕПОРЪКИ ПРИ РАЗРАБОТКАТА НА УЕБ

4.1. Как може да се изложите на риск?

- Предоставяне на лична информация (фамилно име, възраст, телефонен номер, адрес или номер на социална осигуровка)
- Предоставяне на финансова информация (номера на кредитни карти или банкови сметки)
- Отговаряне на неподходящи съобщения
- Договаряне на тайни срещи (онлайн или лично)
- Предоставяне на твърде много информация в онлайн профила ви
- Мигновени съобщения (след като бъде изпратено, съобщението в реално време не може да бъде върнато - например при чатване или във Фейсбук)
- Публикуване на неподходящо видео и материал (представящи образ, за който по-късно бихте съжалявали).

Комуникация: Крайният резултат от уебдизайна авната цел на един уебсайт е да предоставя информация на получателя относно темата, която е представена. Уебдизайнерът трябва да вземе предвид всички възможни интерпретации на материала, за да се гарантира правилното представяне на идеята в сайта.

Уебдизайнерът трябва да помисли за публиката, за която е създаден уебсайта. Кой ще гледа или ще получава уебсайта? Предимно за местна, национална или световна публика ли е предназначен? Културата също играе важна роля в дизайна на един уебсайт. Текст, изображения и цвят, които са приемливи за една култура, могат да бъдат напълно неприемливи за друга. За да избегнете тези проблеми, винаги имайте предвид публиката, когато създавате уебсайт.

Зрение: Как човешкото тяло обработва комуникацията?

Почти 80 % от информацията, която човешкото тяло обработва, идва от окото. Човешкото око е много сложен орган, който е от съществено значение за информационния поток към мозъка.

Цветовата чувствителност се използва за описание на диапазона на цвета, който човешкото око може да види. Тази чувствителност значително се различава при всяко човешко същество. Като цяло женските очи имат по-голяма цветова чувствителност от мъжките. Уебдизайнерът трябва да е наясно с тази цветова чувствителност и да я включи в проектирането на сайта.

Цветовата чувствителност може да определи контраста, който е нужен на получателя на информацията, предоставена в уебсайта. За да се чете лесно текстът на уебстраницата, цветът му трябва да контрастира с фона. Погледнете някои от уебсайтовете, които посещавате ежедневно. Какви са често срещаните цветове за фон? Какви са често срещаните цветове на текста? Черното и бялото често се избират за цвят на текст и фон, но какво да кажем за използването на другите цветове уебсайтовете? Когато гледа облекло, жената може да определи цвета като тъмносин, докато мъж може да приеме същия цвят като черен. Причината е, че дължината на вълните на черно и тъмносиньо са толкова близки, че мъжкото око има трудности при различаването им. Женското око е по-добре пригодно, за да направи тази разлика. Имайте предвид аудиторията на вашия уебсайт. Въпреки че аудиторията Ви може да е предимно от

жени, неволно можете да създадете затруднения за мъже, които ще посетят вашия сайт. Може да включите повторения в сайта за неща, които искате зрителят да запомни. Включването на снимки и таблици на материала също ще помогне на зрителя да запомни вашата информация. Когато говорите за части от нещо, включването на диаграми с етикети за всяка от частите ще помогне на зрителя да закодира снимка на материала в собствения си ум. Създаването на интерактивна дейност, където зрителят трябва да изпълни задача, използвайки информацията, включва допълнителни сетива и поради това увеличава способността да се запомни информацията. Начинът, по който дизайнерът - ученикът използва елементи на дизайн, за да създаде уебсайт, може също така да повлияе на информацията, която зрителят получава. Също както проблемите, свързани с комуникацията в началото на този раздел, така и визуалното възприятие може да повлияе на информацията в уебсайта.

4.2. Закон на Фиц

Законът на Фиц казва, че времето, което отнема на един индивид да се премести към даден обект, е свързано с разстоянието и размера на обекта.

В уебдизайна този закон може да бъде полезен в организацията на сайта. Ако движението към един обект зависи от размера и разстоянието, тогава уебдизайнерът ще иска да постави елементи, които са желани в непосредствена близост до фокуса на потребителя и с по-голям размер. Обратното би било вярно за нежелани предмети. Например при проектиране на уебсайт, където потребители ще поръчват продукти, дизайнерът ще иска да направи бутона „Add to my order“ (Добави към поръчката ми) по-голям от околните елементи и на място, което е близо до поръчвания предмет. Бутонът за отказ на поръчката може да бъде по-малък и не толкова достъпен.

4.3. Петте елемента на дизайна

Цвят, форма, линия, текстура и типография са петте елемента на дизайна. Това са нещата, които изграждат визуалната част на уебсайта ви. Използването на тези елементи по правилен начин ще подобри сайта Ви и ще му придаде по-високо ниво на професионализъм и авторитет. Изучаването на правилната употреба на тези елементи ще ви помогне да създадете уебсайт, който да влезе в списъка на най-добрите в интернет.

Петте принципа на дизайна (**движение, баланс на композицията, близост, повторение и единство**) са всички начини, по които ще използвате елементите на дизайна във Вашия уебсайт. Точно както има трикове за използване на елементите, така има и трикове за прилагане на принципите.

4.4. Цвят

При избор на цветове за уебсайт дизайнерът трябва да използва такива, които се допълват. Това означава цветове, които са един срещу друг на цветното колело. Когато дизайнерът използва *допълващи се цветове (Complementary colors)*, един цвят позволява на другия да изпъкне. Ако погледнете схемата на цветното колело, ще видите следните двойки допълващи се цветове: червено и зелено, оранжево и синьо и жълто и виолетово. Това са само няколко примера за допълващи се цветове. Обърнете внимание на някои от уебсайтовете, които посещавате ежедневно, и вижте дали използват допълващи се цветове.

За да разберете тази идея, помислете как би изглеждал даден уебсайт, ако използвате аналогични цветове. ***Аналогични цветове (Analogous colors)*** са тези, които са един до друг. Изберете някакъв цвят от цветното колело и се преместете е един цвят наляво или

надясно. Как би изглеждал уебсайтът ви, ако използвахте тези цветове за текстов и фонов цвят? Ще бъде много трудно да се различи текстът от фона поради близката дължина на вълните на цвета. Не забравяйте, че мъжете срещат по-големи трудности в различаването на цветове с подобна дължина на вълните. Поради което, ако изберете да използвате жълто-зелено за цвят на фона със зелен текст, вашият уебсайт ще бъде много труден за четене. От друга страна, ако използвате син фон с оранжев текст, това ще доведе до високо ниво на контраста, което ще направи текста лесен за четене. Въпреки че трябва да имате предвид допълващи се цветове за създаване на контраст, те винаги дават добър избор за цвят на текста и фона. Син фон с оранжев текст или оранжев фон със син текст може да бъдат непоносими за зрителя. Може да искате да изберете цветове в рамките на същата тоналност за цвят на вашия фон и текст, но винаги трябва да имате предвид идеята за използване на допълващи се цветове, за да създадете контраст.

Изборът на цветове е важна част от проектирането на вашия уебсайт, защото това ще бъде едно от първите впечатления, които зрителя ще си създаде за вашия сайт. Зле избраните цветове ще оставят зрителя с лоши впечатления от сайта ви, което ще омаловажи посланието ви. Има няколко уебсайта, които могат да помогнат при избора ви. Използвайте уебтърсачка, за да потърсите „уебцветове“ (web colors), или нещо подобно. Това търсене трябва да доведе до сайтове, които ще ви помогнат да изберете правилните цветове за вашия уебпроект.

При използването на уебресурси за избор на цвят ще срещнете термините RGB и HEX. RGB е съкращение от червено (red), зелено (green) и синьо (blue), като всеки цвят има определен номер. Този номер показва количеството светлина от всеки цвят (червено, зелено или синьо), която е добавена, за да се създаде той. Използването на RGB номер ще гарантира, че сте получили точния цвят, който искате за вашата работа. Смесването на светлина от различните цветове ще бъде това, което зрителят на уебсайта ви ще вижда, тъй като телевизори и компютърни монитори го използват за изграждане на цветовете, които виждате.

RGB кодовете варират от 0:0:0 (черно или липсата на всички цветове) до 255:255:255 (бяло или смес от всички цветове). Има няколко цветови таблици, които може да бъдат намерени в Мрежата, за да ви помогнат да изберете вашите цветове. Тези таблици включват списък с набора от трите числа, за да ви помогнат да зададете цветове за вашия уебсайт. Заедно с този RGB код също така ще срещнете HEX или хексадецимален код.

Хексадецималният код (HEX) е много подобен на RGB кода, но използва серия от цифри и букви за идентифициране на всеки цвят. Всяка от трите стойности на цвят (0 - 255) се изразява в двуцифрен хексадецимален (основа - 16) код, което води до шестцифрен ред, който представлява кодът. (Хексадецималните цифри се състоят от числата 0 - 9 и буквите A - F, които представляват стойностите от 10 до 15, тъй като нямаме едноцифрено представяне за тези стойности в нашата десетична бройна система.) Например хексадецималният код за черно е 000000, а за бяло е FFFFFFFF. Всички други цветове имат собствена идентификация. RGB или HEX класификациите може да се използват, за да помогнат в създаването на вашия уебсайт, но ще искате да разберете кой код използва вашата програма за уебдизайн. (Някои софтуери позволяват въвеждането или на RGB, или на HEX код.)

След като сте избрали контрастиращи цветове за вашата работа, може да искате да използвате различни цветови нюанси за други части от уебсайта си. Ще забележите, че много уебсайтове използват повече от един цвят за текста или блоковете от цветове, за да

привлекат вниманието към определена част. Дизайнерите на уебсайтовете са склонни да използват цветовете в рамките на определена тоналност. Отново Мрежата може да бъде важен инструмент за откриване на тези цветове. След като сте избрали основен цвят за уебсайта, можете да видите какви други цветове ви се предлагат. Уебсайтовете, които може да използвате, за да направите своя избор, трябва да предлагат RGB или HEX цветови модели.

4.5. Форма

Формата може да направи по-атрактивен уебсайта ви и да помогне в организирането на текста или на други елементи. Погледнете някои от любимите си уебсайтове. Ще видите, че повечето използват правоъгълна форма. Използването на правоъгълници (или карета) създава затворена форма и позволява на нашето око да се фокусира върху тази област. Заглавията (headings) може да са в цветни карета; начинът, по който текстът е подравнен отляво или отдясно, създава каре от текст; заглавията може да бъдат в каре в по-голям правоъгълник, обграждащ придружаващия текст; или дълги колони от правоъгълници може да се използват, за да разделят сайта на секции. Често се използват правоъгълни колони, за да разделят уебсайта на секции. Лявата секция е обикновено за бутони, които позволяват на потребителя да се придвижва от една страница в друга. Средната секция обикновено е за основното съдържание на сайта, докато в дясната колона обичайно ще откриете реклами или вторична информация за сайта. Това не са строги правила за уебдизайн, но ще откриете много сайтове, които следват тази идея.

Рекламирането има собствен начин за използване на форми. Повечето реклами се поставят в каре в горната част на страницата, в колона от лявата или от дясната страна. Използвайте любимата си търсачка, за да потърсите често срещана тема (например кучета). Когато резултатите се покажат на екрана, опишете тяхната форма. Много често ще видите, че ако очертаете външната страна на текста, ще създадете правоъгълник. Десният край например може да е назъбен, но общият контур на текста остава правоъгълник.

Повечето сайтове използват правоъгълник, за да организират своята навигация. Вижте бутоните, които ви отвеждат от една част на уебсайта към друга. Формата на тези бутони е най-вероятно правоъгълна, докато други може да бъдат овални. Всеки от тях използва определена форма, за да очертае областта на бутона.

4.6. Линия

Използването на линията е подобна идея на тази за формата. Изберете който и да е уебсайт и вижте линиите, които са използвани. Някои може да бъдат прави, докато други са криви. Линиите може да бъдат цветни, а други - черни или бели. Потърсете и определете към коя от следните категории попадат линиите в сайта Ви.

Най-разпространеният начин за използване на линии е за организиране на идеите. Линията може да се използва за разделяне на една идея от друга. Тя дава на зрителя чувство за завършеност преди преминаване към следващата идея. Може да откриете тънка линия на заден фон, която се показва между навигационните бутони. Тази линия помага за организирането на сайта чрез определяне на границите на навигационния бутон.

Линии може да се използват за показване на посока на даден обект или движението му. Може да показват движение на обект чрез илюстрация. Помислете си за сайт, в който се разглежда храносмилателната система. Илюстрация на разбираема храносмилателна система може да се направи лесно чрез добавяне на стрелка към устата, където влиза храната, друга линия - през хранопровода надолу към стомаха, и т.н. Друг пример за

използване на линия за показване на движение е, когато един карикатурист ги рисува до изображението, за да създаде илюзия, че обектът се движи в определена посока. Докато не се научите да създавате анимация в сайта си, трябва да използвате линии, за да внушите това движение.



Използването на линии за организация може да изглежда, че разделя части от сайта, но линията може също така да ги свързва. Тя може да помогне на зрителя да разбере, че идеите, обединени с линията, са свързани.

Накрая, друг популярен начин за използване на линия е да подчертаете текст, за да подскажете на зрителя, че той е хипервръзка към друг сайт или раздел от сайта. Много пъти това се прави чрез комбинация от цветен текст и подчертаване. Много е важно за един уебдизайнер да разбере, че зрителите предполагат, че подчертаният текст е хипервръзка, върху която могат да кликнат. Използването на подчертаване на друго място в сайта ще причини неудовлетвореност в зрителя.

4.7. Текстурата

Текстурата е използването на дълбочина, така че дадено изображение да не изглежда плоско. Дълбочината е прилагане на определени дизайнерски техники за създаване на триизмерен външен вид върху плоска повърхност - понякога се нарича 3D. Това е начин да се накара окото да възприеме, че изображението има някакво обемно измерение. Текстурата може да се постигне чрез използване на сенки, детайлни линии или подчертаване с цвят.

Използването на текст със сенки създава илюзия у зрителя, че текстът е повдигнат от страницата или на екрана. Ако текстът е плосък, тогава няма да има сянка.

Въпреки че текстурата може да направи по-интересен сайта, прекалената ѝ употреба вероятно ще разсейва зрителя. Уверете се, че използвате текстурата изборително в области, където бихте искали да направите малко по-атрактивен сайта си.

Друг начин за използване на текстура във Вашия уебсайт е да използвате текстурен фон вместо плътен цвят. Най-добрият начин да го направите е да поставите текста и други визуални елементи в каре, което да ограничи текстурния фон. Тук бихте могли да използвате идеята за форма (каре) в комбинация с текстура (фона), за да създадете atrактивен изглед на уебсайта си. Друг начин за използване на текстурен фон е да създадете избледняла или по-светла негова версия, където текстът ще контрастира с фона.

4.8. Типография

Типографията е може би най-важният елемент от Вашия уебсайт. Тя включва всички неща, които правите с шрифта в рамките на уебсайта. Отново погледнете някои от любимите си уебсайтове. Как Вашият сайт използва шрифтове? Как привлича вниманието към определени части от информацията? Как се използват шрифтовете, за да помогнат на организацията на сайта? Как сайтът използва шрифтове, за да предаде дадено послание? Типографията е много важна част от предаването на посланието към получателя.

Използването на правилния шрифт, за да доставите вашето послание на зрителя, е важна част от самото послание. Има два основни вида шрифтове, които се използват в ежедневните ви срещи с текста. Серифните шрифтове (*serif*) използват малки черти в края на буквите, докато безсерифните шрифтове (*sans-serif*) нямат такива черти. (*Sans* е френската дума за *без*.) В печатни съобщения и в по-дълги текстове е нормално да се използва серифен шрифт. При четене на информация от екрана е прието да се

използват безсерифни шрифтове за по-дълги пасажии. Погледнете любимия си уебсайт. Има ли на страницата смесица от шрифтове *serif* и *sans-serif*? Ако е така, къде обикновено откривате серифните шрифтове? Къде обикновено откривате безсерифните шрифтове?

AaBbCcDd AaBbCcDd

Буквите вляво са *Times New Roman*, един пример за серифен шрифт. Буквите вдясно са в *Arial* - пример за безсерифен шрифт.

Третият вид популярни шрифтове се нарича декоративен. **Декоративните шрифтове са ръкописни и символни.** Добре е ръкописните шрифтове да се използват за заглавия, но прекадената им употреба може да бъде разсейваща за сайта. Трудно е да се прочете по-дълъг текст, написан с ръкописен шрифт. Символните шрифтове са предназначени за декорация, а не за четене; поради което трябва да ги използвате само на подходящи места.

От време на време може да искате определени думи да се открояват от останалата част на текста. Това може да бъде постигнато чрез типография. Чрез промяна на цвета на определена дума, добавяне на курсив (*italic*), промяна на размера на шрифта или добавяне на получерно (**bold**) форматиране авторът на уебсайт може да привлече вниманието към определен текст. Същите техники може да се използват за организиране на сайта.

Уебсайтът може да се разглежда като учебник. Когато материалът преминава от една идея или тема към друга, авторът може да използва заглавие в получерен (**bold**) шрифт, като размерът му е малко по-голям. Това помага да се префокусира вниманието към темата и позволява по-добра организация на материала. Заглавията в получерен шрифт с по-голям размер улесняват по-бързото намиране на информация в рамките на сайта. Помислете върху примера, където сте извършили търсене за широко разпространена тема като кучета. Ако един уебсайт представя различните породи кучета, но вие се интересувате само от йоркширски териери, вероятно ще искате бързо да прегледате материала, за да откриете това, което искате. Авторът, който използва заглавия (**headings**), ще улесни откриването на такава информация. Ето няколко насоки за избор на размер на шрифт: размери, **по-малки от 10 pt (пункта), са трудни за четене**; широко разпространеният размер е 12 точки; размер на шрифт над 14 точки трябва да се използва за заглавия (**headings**). Разгледайте любимия си уебсайт и забележете къде е използван получерен шрифт в различни размери.

Последният въпрос, който трябва да вземете предвид в типографията, е дали вашият избор за стил на шрифт изпраща посланието, което искате. Стилът на шрифта, цветът и главните букви влияят върху посланието, което изпращате с текста. Погледнете следните примери на текст и се опитайте да установите какво послание според вас авторът се опитва да внуши на читателя.

Помислете за електронните съобщения, които сте изпращали на приятели, написани само с главни букви. Когато искате да крещите за нещо, вие пишете само с главни букви. Ако сте ядосани, пишете само с главни букви. Ако пишете електронно съобщение на приятел, че искате да поръчате пица за вашето вечерно събиране, най-вероятно няма да го крещите чрез съобщението.

4.9. Петте принципа на дизайна Движение

Има две неща, които трябва да се вземат предвид, когато са занимавате с движение във вашия уебсайт: естествените движения на очите в сайта и движението, създадено от

автора на сайта. Всяко от тях може да бъде повлияно от това как авторът използва елементите на дизайна в уебсайта.

По навик ние започваме да разглеждаме даден текст от горния ляв ъгъл. Започваме да четем от горната част и се движим хоризонтално по страницата, отляво надясно, отгоре надолу. Същият навик по естествен начин се прилага, когато гледаме една уебстраница. Читателят автоматично ще погледне първо към горния ляв ъгъл на текста в страницата. Там обикновено е мястото, където се намира заглавието. След това читателят ще следва първия хоризонтален ред на страницата. После автоматично ще се премести надолу по страницата към втория хоризонтален ред, който привлича вниманието му, не е задължително да е следващият ред от текста. Окоето отново ще следва хоризонтално този втори ред. Най-накрая читателят ще се върне обратно към лявата част на страницата и ще позволи на окоето си да потърси надолу по нея още информация. Този модел на движение на окоето създава формата на F. Първото движение от горната лява част по страницата е най-горното стъпало на F. Второто движение по страницата създава второто стъпало на F и най-накрая движението на окоето надолу надясно създава стъблото. Други разновидности на този модел на движение включват модел E и обърнат модел L.

Това движение на окоето казва на уебдизайнера, че информацията, която ще привлече читателите към останалата част от сайта, трябва да бъде поставена на видни места. Тези области ще бъдат горният ляв ъгъл на текста и левият край на страницата. Уебдизайнерът може допълнително да привлече вниманието на читателя, като използва цвят, различни стилове и получен шрифт.

Вторият тип движение е създаден по-директно от дизайнера. Чрез употребата на форми, линии и цветове той може да създаде йерархия. Йерархията е начинът, по който дизайнерът предава важността на идеите на читателя. При използването на форми дизайнерът може да избере да включи идеята е най- малка важност в малък кръг, а идеите от нарастващо значение - в малко по-големи кръгове. Естественото движение на очите е да изберат една посока и да я следват. Окоето може да започне е по-големия обект, да следва преминаването към по-малките обекти и да поеме по обратния път. С която и да е от двете посоки читателят ще има чувството за йерархия, докато идеите преминават от един кръг към друг. Линията също може да се използва, за да насочи погледа на читателя в определена посока в рамките на уебсайта. Окоето по естествен начин ще следва линията, която свързва определени елементи на сайта.

4.10. Баланс на композицията

Балансът е разпределението на елементи в пространството, а в случая е уебдизайна това е разположението на елементи в рамките на уебстраницата. Мислете за обекти, които са по-големи или по-тъмни, като за тежки, докато по-малките и по-бледи обекти са по-леки. Вижте любимия си уебсайт и се опитайте да откриете типа баланс, който е използван. Има три основни типа композиция, които можете да откриете: симетрична, асиметрична или кръгова.

Симетричната композиция ще създаде уебстраница е равни тежести от всяка страна на линията, която е начертана вертикално по средата. Това е същият тип симетрия, която се появява толкова често в природата.

Асиметричната композиция изисква повече планиране от уебдизайнера. В тази техника дизайнерът балансира обекти на страницата чрез разпределение на размерите и

цветовете им. Няколко обекта е по-малък размер и по-леко оцветени могат да бъдат балансирани чрез един по-голям и с по-тъмен цвят на другата страна на страницата.

Кръговата композиция е просто поставяне на обектите в кръг около една централна точка. Този метод помага да фокусираме вниманието на читателя, тъй като всички обекти сочат към центъра на кръга.

Подравняването може да ви помогне да използвате баланса при създаването на Вашия уебсайт. Това е ориентация на елементите един към друг. Подравняването най-често се използва за текст, но същата идея може да се приложи върху изображения, както и други елементи. Има четири основни типа подравняване: ляво, дясно, центрирано и двустранно.

Първият и най-често използван тип подравняване в европейската култура е лявото. Тъй като четем отляво надясно, лявото подравняване е естествено за нас. Дясното подравняване е точно противоположното. То се среща често в арабската култура, където текстът се чете отдясно наляво. Третото подравняване е центрираното. Често сте виждали центрирано подравняване да се използва при заглавия. Четвъртото подравняване изравнява текста в левия и десния край и най-често се открива в списания и вестници. То се нарича *двустранно (Justified) подравняване*.

4.11. Близост

Близостта е мярка за разстоянието между обектите и посланията, които се изпращат чрез него. Близостта може да се използва, за да ни помогне да определим йерархия чрез поставяне на идеи със сходна важност на една и съща линия. Разделянето на обекти може да покаже, че те са напълно различни в идеите си. Чрез употребата на близост уебдизайнерът експериментира с използването на **бяло пространство** или области без текст и изображения. По-малката употреба на бяло пространство внушава по-голяма връзка между обектите, докато увеличеното бяло пространство създава по-голямо разделение между тях.

Помислете за използването на бяло пространство в писането. Когато писателят минава към друга мисъл, започва нов параграф. Началото на нов параграф предоставя бяло пространство и подсказва на читателя, че информацията се променя и преминава към следващата тема.

Близостта може също така да демонстрира връзка между частите и цялото.

4.12. Повторение

Уебдизайнерът може да използва принципа на **повторение**, за да акцентира върху една идея или изображение. Често използван начин за повторение е свързването на една идея или дума с графика.

Друга техника е повторната употреба на даден цвят. Цветовете и оформлението в различни страници на уебсайта също трябва да се повтарят. По този начин добавяте елемент на приемственост и казвате на читателите, че са все още в сайта, създаден от същия автор.

4.13. Единство

Последният принцип на дизайн е **единство**. Единството е начинът, по който дизайнерът обединява всички части (цвет, форма, линия, текстура и типография), което води до последователност, за да оформи крайния продукт.

Близост и повторение са два принципа, които играят важна роля за създаването на единство в рамките на един уебсайт. Чрез използване на близостта дизайнерът поставя заедно нещата, които са свързани. Чрез повторението придава на сайта последователност. Погледнете любимия си сайт. Докато сърфирате из него, забележете

дали дизайнерът повтаря цвят и оформление в страниците. Ако е така, значи той е създал единство чрез повторение на цвета и оформлението.

4.14. Работа с изображения и медии

Важен аспект от дизайна на една уебстраница са създаването и обработката на вашите изображения. Когато търсите в Мрежата, ще откриете сайтове с ефектни графики, които внушават желаното послание. Когато създавате своя уебстраница, първо вземете решение за целите и публиката ѝ и след това започнете да събирате и създавате необходимата медия. Тази медия трябва да съвпада с посланието, така че да не обърква публиката. Каква медия може да използвате? Може да използвате професионални графични програми за създаване на уникални изображения, лого, банери, бутони и други като например: MS PhotoShop, Adobe Illustrator, Coreldraw за векторна графика и други.

4.15. Използване на изображения в уебдизайна Лого (logo)

Логото е търговска марка, текст или символ, който представя една компания, организация или лице. Логото представлява посланието или целта на компанията. То може да включва изображение, текст или символ. Сещате ли се за лого във вашата среда? Ето няколко примера:

Ето едно лого, което използва само символ. Познахте ли? Да, Windows!

Фиг. 2 Лого на Windows



Лого на Пловдивския
Университет „Паисий
Хилендарски“
гр. Пловдив;



*Фиг. 3 Лого на ПУ „Паисий
Хилендарски“*



*Фиг. 4 Лого със стилизиран
Хилендарски“*

текст на ФМИ на ПУ „Паисий

Ето едно лого, което използва само стилизиран текст. Познахте ли? Разбира се, ФМИ на ПУ „Паисий Хилендарски“ – гр. Пловдив!

Защо да създаваме лого? Проектиране

Логото ускорява обработването на паметта, като въздейства по-силно върху публиката. Умовете ни ще си припомнят изображението по-лесно, отколкото само текст.

Публиката тогава автоматично ще свърже този символ с продукта или услугата на компанията. Логото е „първото впечатление“ за една компания; то доставя трайно послание без всички други детайли.

Първо, помислете за целите на компанията. Ако тя набляга на запазване на природните ресурси, какви цветове и форми ви идват на ум? Възможните форми за това лого биха били зелени, кафяви и естествени. Ако мисията на компанията е да създаде мислещи играчки за деца, бихте могли да използвате чисти цветове с основни геометрични форми. Преди да започнете работа, разберете каква ще бъде публиката и къде ще бъде използвано логото (в мрежата и/или в печата). Вашата цел е да създадете лого, което е уникално и свързано с тази компания.

След като имате изградена идея за компанията, можете да започнете да създавате опростен дизайн. Логото трябва да бъде опростено за лесно разпознаване. Не забравяйте, логото е първоначалният поздрав, а не реч. Вашето завършено лого за Мрежата ще бъде малко и с резолюция между 72 и 96 dpi (точки на инч).

Когато се направи както трябва, едно лого е най-простият начин за създаване на асоциация между името и продукта.

4.16. Банери (Banners)

Уебсайтовете са покрити с пъстроцветни банери, някои се използват като заглавия, други - като реклами, а трети изискват да бъдат кликнати с мишката. Банерите са форма на реклама в Световната глобална мрежа и са предназначени за привличане на трафик към определен уебсайт.

Има три вида банери: статични, анимирани и **интерактивни банери**.



Фиг. 5 Статичен банер

Това е пример за статичен банер. Статичните банери са създадени за заглавието на уебстраница и е възможно да съдържат логото.

Пример за анимирани банери. Стрелките, текстът и обектите се движат, за да привлекат погледа на потребителя.



Фиг. 6 Анимирани банер

Пример за интерактивен банер. Този вид банер изисква от потребителя да кликне върху самия банер и отива директно в началната страница (в случая index.htm).

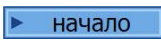


Фиг. 7 Банер водещ към началната страница

Често банерите са предназначени за рекламни кампании. Компаниите желаят висок коефициент на кликвания (CTR-click-through ratio), които ще доведат до високи продажби. Коефициентът на кликвания (CTR) се базира на броя потребителски кликвания върху банера за събиране на допълнителна информация. За да подобрите коефициента на кликвания (CTR), използвайте ярки цветове и думи в получерен (bold) шрифт, добавете думата „безплатно“ (free), използвайте стимул за кликвания върху банера и направете посланието опростено и кратко.

4.17. Бутони

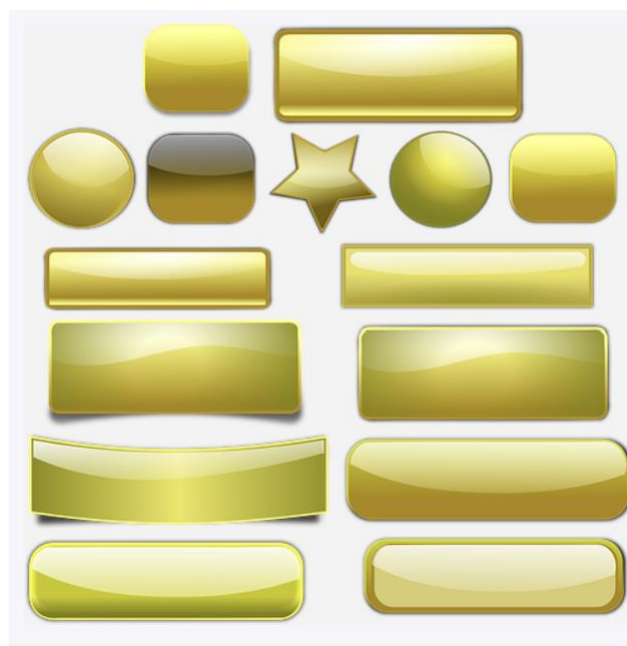
Бутоните са ключови компоненти в уебдизайна за навигация или връзка към други ресурси (сайтове, страници, видеоклипове или документи). Бутоните могат да бъдат с различна форма и да съдържат текст и/или изображения. Когато създавате уебстраница, екипирайте дизайна си, включително навигационната област с бутони. Използвайте последователност във форма и цвят, която да допълни вашия дизайн.

Това са примери за различни стилове бутони: 



Фиг. 8 Различни видове бутони

Когато използвате бутони във вашия уебдизайн, последователността е от жизненоважно значение за вашата тема. Ако уебстраницата е неприятна за гледане с твърде много цветове, форми на бутони или анимации, зрителят незабавно ще излезе. За достъп до готови бутони използвайте любимия си браузър, за да потърсите „безплатни уеббутони“ (Free Web buttons). Ще бъдете изумени!



4.18. Ролоувър (Rollovers)

Ролоувър е термин за промяната във външния вид, когато потребител позиционира показалеца на мишката върху графичен бутон, текст или изображение. Тази промяна може да се изразява в изскачаща дума, описваща функцията на бутона или промяна в изображението (цвят, избледняване, увеличение). Ролоувърите помагат на потребителя при навигация и добавят движение към един уебсайт. Може да видите пример в сайта ucd.bg.

представяне	структура	обучение	новини	галерия	контакти
presentation	structure	training	news	gallery	contacts

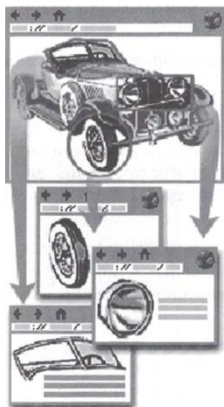
Фиг. 9 Примерен сайт, от който може да видите ролоувърите посочвайки тези бутони

Примерът показва първоначалния бутон с тъмен към светъл градиент и отблясък. След като потребителят отиде върху бутона, градиентът се обръща, така че става от тъмен към светъл. Когато кликнете върху бутона, отблясъкът изчезва и градиентът става по-тъмен. Този ролоувър има три изображения, включени в HTML кода.

4.19. Карти на изображение (Image Maps)

Картата на изображение (image map) е друга техника за навигация в уебсайт, Потребителят кликва върху части от изображението, за да осъществи достъп до различни страници в рамките на сайта. Това е креативен начин за навигация, но дизайнерът трябва да има предвид следното:

- Картата на изображение може да бъде по-голяма от набор бутони за навигация, което да доведе до по-бавно зареждане.
- Потребителят може да е изключил графиките в своя браузър или да използва текстово базиран браузър, затова не забравяйте да предвидите алтернативен начин за навигация.
- За да може една търсачка да получи достъп до всички страници в уебсайта, поставете текстови връзки и други малки графики на всички страници, за да бъдат те лесни за намиране.



<http://en.wikipedia.org/wiki/Imagemap>

Фиг. 10 Карта на изображение

Потребителят кликва върху една част от колата в изображението и бързо бива отпратен към друга уебстраница, която разказва за тази част.

Рязане на изображение (Image Slicing)

Едно от предизвикателствата за включване на графики в уебстраница е, че големите изображения водят до бавно зареждане. Рязането на изображения (Image slicing) може да помогне да разрешите този проблем. Рязането на изображение е процесът на разделянето му на по-малки правоъгълници, наречени резени (slices). Всеки резен се поставя в една клетка от HTML таблица, след което се оптимизира за създаване на най-малкия размер на файл. Чрез този процес качеството на изображението се запазва.

4.20. Видове графични файлове и употреба

Всеки тип файл е създаден, за да съхранява специфични видове данни и обикновено е означен чрез разширението след името му (няколко букви след точката в името на файла). Например файл, който завършва с .txt, е създаден, за да съхранява текст.

При проектиране на уебсайт ще се натъкнете на три типа графични файлове: GIF (.gif), JPEG (.jpeg), и PNG (.png).

GIF

Файловият тип **GIF** (Graphics Interchange Format - Формат за обмен на изображения) е предназначен за прости графики и анимации. Това е 8 бита на пиксел растерно изображение и използва палитра до 256 цвята. GIF форматът поддържа анимации и позволява палитра от 256 цвята във всеки кадър.

Уебдизайнерите често избират GIF файлов формат поради преносимостта и широката му приемственост. GIF файловете са компресирани чрез процес без загуба на данни, наречен LZW (Lempel-Ziv-Welch), в който файлът се намалява, без изображението да губи качеството си.

GIF файлът е страхотен за лого, проста рисунка с линии, изображения с плътни цветове и опростени анимации. **GIF /Graphics Interchange Format/** е създаден за представяне на графични изображения в Интернет. Използва техника на компресиране „без загуба“, което означава, че информацията за цветовете не се губи по време на самата компресия. Това е форматът на отлично компресиране и изобразяванена плоски /без светлосенки/ цветове, което го превръща в логичния избор за скици /прости изображения/ и цветни графики. Поради ограниченията си в дълбочината на цвета, този формат не е най-добрият избор за фотографии или по-сложни изображения, които включват градиента на цвета, светлосенки или преливане. Може да изберете един от цветовете на GIF изображението да е прозрачен за брауъра. Така цветът ще се слива със този на уебсайта. С малко повече въображение може да създавате всякакви ефекти, включително превъртане на текст, смяна на ц ветове, анимирани икони и презентации, използвайки Adobe Shockwave или Flash, които включват и звук в синхрон с анимацията и позволяват на потребителите да взаимодействат с нея. Инструменти за създаване на GIF анимации: GIF Construction Set professional – www.mindworkshop.com/alchemy/gifcon.html, Ulead GIF Animator – www.ulead.com/ga/runme.htm, GIFMation – www.boxtopsoft.com/gifmation.html.

JPEG

Файловият тип **JPEG** (Joint Photographic Experts Group: името на организацията, която го е разработила - Обединение на експертите фотографи) е предназначен за снимки с плавни тонове и различни нюанси. Той е най-често срещаният тип файл за цифрови апарати. JPEG също така се използва за снимки в уебдизайна.

Файловият формат JPEG използва метод на компресия със загуби (lossy). Това означава, че изображението ще загуби част от качеството си, което не може да бъде възстановено. Обикновено тази загуба не може да се забележи в Мрежата, освен ако изображението не е компресирано няколко пъти.

Очевидно един уебдизайнер ще избере JPEG файлове за снимки в уебсайта си. **JPG**

/Join photographic Experts Group/ - е най-добрият избор за фотографии или нерастерни изображения. JPG са 24-битови изображения с възможност за милиони цветове. За разлика от GIF, за показване на цвят форматът не използва палитра.

Форматът използва компресия „със загуба“, създадена специално за фотографии: когато снимката се компресир, някои цветове отпадат, в резултат на което се губи част от качеството на оригиналното изображение. Тъй като резултатът се изобразява на компютърен монитор с малка разделителна способност, загубата на качество обикновено не се забелязва. Нещо повече, по ниското качество се компресира по-

кратко време за зареждане на изображението. Може да се трансформират фотографии във JPG като се използват Adobe Photoshop или друг софтуер за обработка на изображения. Дори може ръчно да задаваме ниво на компресия за сметка на качеството. Внимание-независимо дали работите с GIF или JPG, съхранявайте оригинала, компресията и при двата формата води до безвъзвратни загуби на качество и не може да се възстановява оригинала.

PNG

Файловият тип **PNG** (Portable Network Graphics - Преносими мрежови графики) е създаден, за да подобри и замести GIF файла. Той е проектиран за прехвърляне на изображения в Мрежата, така че поддържа RGB цвятова палитра и сиви (grayscale) изображения. Това е формат за растерно (bitmap) изображение, което има възможност за компресия на данни без загуба (lossless). Този файлов тип поддържа статични изображения и се използва за анимации.

Файловият тип PNG е чудесен избор за съхранение на щрихова графика (line art), текст или други изображения с остри ръбове.

Графични файлови формати: към настоящия момент подходящи са три формата: GIF, JPG, PNG. SVG предложен от консорциума W3C е все още твърде нов и употребата му е ограничена. Всички формати компресират изображенията, за да намалят размера на файла. Правилният избор на формат е от голямо значение. Ако допуснете грешка при избора си, изображението няма да се компресира или да изглежда така, както очаквате.

PNG /Portable Network Graphics/ - създаден специално за Интернет. Липсва му поддръжка от браузърите. Безплатен файлов формат, предвиден да замени GIF, защото компресира „без загуба“ 8-битови изображения във файлове с по-малък размер спрямо GIF. Може да се ползва за формат за разпечатване, понеже поддържа 8-битов индексен цвят, 16-битова сива скала 24-битов цвят. Не е най-подходящ за фотографии, защото технологията „без загуба“ не е толкова ефективна в сравнение с тази на JPG. Но пък поддържа прозрачност и постепенно зареждане, но не и анимация. Повечето браузъри поддържат този формат, затова все по-често се използва от уебдизайнерите. **SVG /Scalable Vector Graphics/** е нов стандарт на W3C. Той все още не се поддържа от всички браузъри, но се очаква да стане широко популярен. Той е език за описание на двуизмерни графики чрез XML. Това е формат за векторна графика.

4.21. Палитри (Palettes)

Ще разгледаме уеб, системна и оптимизирана палитра. Най-често уебдизайнерът ще избере уебпалитрата.

Уебпалитра (Web): Световната глобална мрежа е стандартизирала предварително определена палитра от 256 цвята. Това се показва като куб 16 x 16 x 16 с 4096 цвята. Тези цветове би трябвало да се показват във всеки браузър и са безопасни за мрежата.

Системна палитра (System): Тя съдържа цветовите стойности, които може да бъдат показани от устройството, което се използва.

Оптимизирана палитра (Optimized): Това е палитра с определени избрани цветове. Дизайнерът може да иска да намали дълбочината на цвета до 8 бита, което е 256 цвята. Това ще намали размера на файла. Често се среща е при GIF изображения и анимации.

4.22. Цифрова фотографии

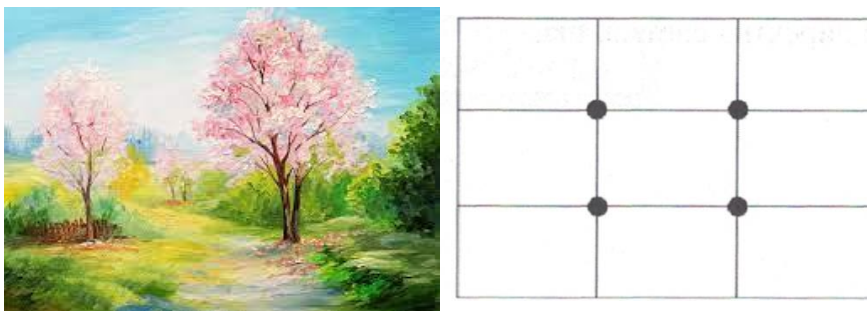
Цифровите изображения могат да бъдат визуализирани, принтирани, обработвани, предавани и съхранявани чрез използване на компютърни техники. Нека разгледаме

някои техники и концепции, които ще помогнат в създаването на висококачествени изображения.

4.23. Техники и концепции

Отрез от кадър (Factor Framing): Когато е установен централният обект, създайте естествена рамка, за да го заградите. Например едно дърво може да бъде естествена рамка за портрет. Отрезът от кадър (Factor framing) създава „неоновия знак“, който казва, „Виж ме!“.

Правило на третините (Rule of Thirds): Това е добре известно композиционно правило, което измества обекта от центъра на снимката. Представете си снимка, разделена на девет равни части. Обектът трябва да е поставен в една от четирите пресичащи се точки. Тази асиметрична композиция е приятна за окото на зрителя.



Фиг. 11 Приятна асиметрична композиция

Посока на движение (Leading Room): Тази концепция се основава на идеята, че зрителят иска да види накъде отива движещ се обект, а не къде е бил. Например куотърбек, който хвърля топката, трябва да бъде в дясната третина на снимката. Зрителят иска да види къде е хвърлена топката.

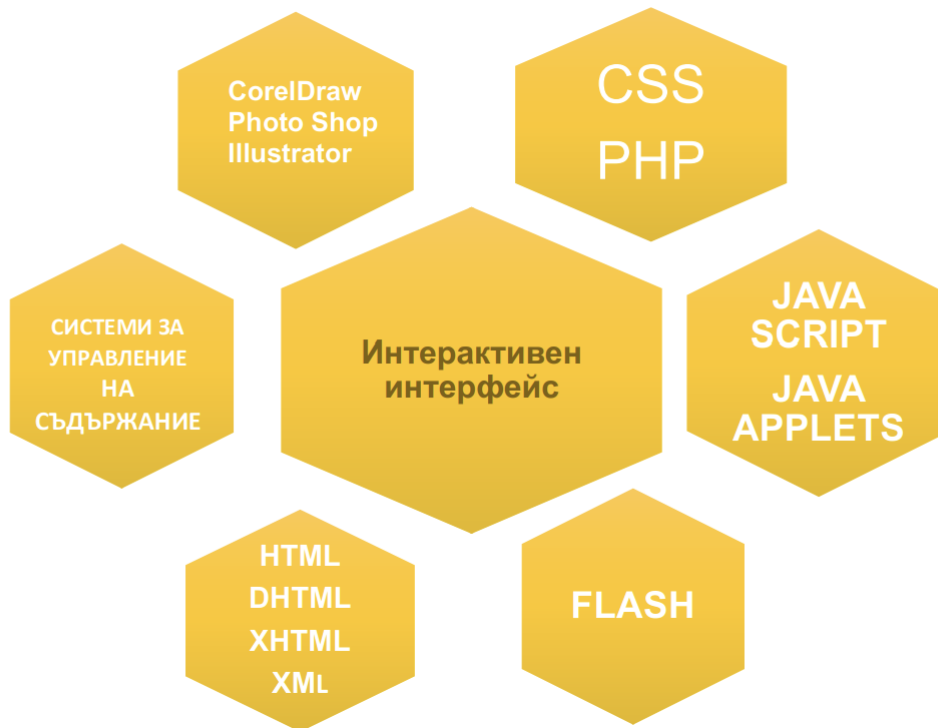


Фон (Background): Когато правите снимка, трябва да сте наясно какво има зад централния обект. Ако фонът се пренебрегне, е възможно сливане на нещо от задния план с централния обект. Снимката трябва да е опростена без разсейване във фона.

Осветление (Lightning): Зазоряване и здрач са идеалното време, за да правите снимки. Заснемане по обяд или при ярка слънчева светлина ще доведе до груби сенки. Снимането в ясен ден дава красиви цветове в обектите ви. Ако снимките са направени в затворени помещения, използвайте колкото се може повече естествена светлина и избягвайте директна светкавица.

Ориентация (Orientation): Тази техника се отнася за положението на камерата: портретна или пейзажна ориентация. Вертикалният (портретен) формат подчертава височина и поради това е подходящ за водопад или изправен човек. Хоризонталният (пейзажен) формат е добър за пейзажи или група от хора.

ГЛАВА ВТОРА. МЕТОДИЧЕСКИ ИНСТРУМЕНТАРИУМ ЗА РЕАЛИЗАЦИЯ НА МОДЕЛА



1. ПРОЦЕСЪТ НА ПРОЕКТИРАНЕ. СХЕМА НА РАБОТНИЯ ПРОЦЕС

Добре проектираните сайтове интегрират визуалната организация на съдържанието, интуитивна система за навигация, внимателен подбор на типография (шрифт, цвят, размер и разстояние), спазване на стандартите на W3C с цел единен, последователен и професионален дизайн.

Уебдизайнът е неразделна част от по-големия процес на уебразработване. Уеб-дизайнерът създава външния вид и навигацията на даден уебсайт с помощта на HTML или други програмни езици, както и редица програми за компютърна графика. Тяхната работа определя това, което хората виждат и с което взаимодействат, когато посещават сайта. Една от най-важните задачи на дизайнера е разработването на навигационната система, която напътства и води посетителите през сайта. Уебдизайнерите са отговорни за създаването на хубави графики или други визуални елементи и за избора на стилове и шрифтове, за да направят сайта привлекателен и да помогнат на дадена компания да прогресира в бизнеса си.

Все повече и повече до интернет може да се осъществява достъп чрез различни връзки и устройства (безжичните устройства като лаптопи, поддържащи Wi-Fi или Bluetooth, мобилни телефони и PDA - персонални цифрови помощници).

Работата на един уебдизайнер е да направи сайта функционален и приятен за потребителя. В същото време дизайнерът трябва да работи за постигане на целите на клиента; един корпоративен уебсайт трябва да продава или да търгува с печеливши за бизнеса продукти или услуги. В момента актуална специалност в училище е Електронната търговия. Уебдизайнерът, който създава корпоративен интернет сайт за служители например, трябва да осигури лесен достъп до съответната информация. Уебдизайнерът, който изгражда сайт за електронна търговия, трябва да гарантира, че

потребителите разпознават продукта на компанията и да направи процеса на пазаруване възможно най-лесен.

Създаването на сайта преминава през няколко етапа, в които се включват различни специалисти. Всеки етап е значим и към него следва да се подходи с разбиране и професионализъм. Пропускането на някой от етапите или недоброто изпълнение на дейностите по тях води до неуспех на сайта като цяло. Етапите и специалистите, включени в тях са дадени в таблица 2.

	Проектиране		Създаване			Разпространение	
	Стратегия	Структура	Съдържание	Разработка и програмиране	Тестване	Публикуване	Популяризиране
	Цел, аудитория, изисквания	Блок-схема на сайта, архитектура и дизайн на страниците	Текст, графика	HTML, Java Script, CSS, DHTML, ASP, GGI, PHP	Браузъри, резолюции, цветове, скорости, навигация	Домейни, хостинг, FTP	Регистрация в търсачки, банери, промоция, офлайн реклама
Клиент	■	■	■			■	■
Мениджър	■	■	■	■	■	■	■
Маркетинг	■	■	■				■
Конструктор		■	■		■		
Дизайнер		■	■		■		■
Редактор		■	■	■			
Програμισ		■	■	■	■	■	■
Разработчик			■	■	■	■	■

Таблица 2. Етапи, специалисти, технологии

■ - незадължително

■ - задължително

Навигация и разположение

Навигацията на сайта е много важна, доколкото от нея зависи ориентирането на посетителите в съдържанието му. Колкото по-добре е направена тя, толкова повече отделни страници има възможност да посети търсещият и да разгледа повече от това, което му предлагате.

Да се създаде добра навигация е истинско предизвикателство. При необходимост от включване на повече теми е добре да се разпределят в основна и допълнителна навигация или хоризонтална и вертикална такава. Добре е всяка навигационна лента да включва между 5 до 7 елемента, иначе става трудна за възприемане и ориентация. Също така е важно да се обмислят добре менютата от типа „падащи“ или „разгръщащи се“, тъй като те биха могли да останат невидени от част от потребителите или трудна за употреба.

Ето няколко правила за навигацията и разположението (layout - лейаута) на страниците, засягащи силно и тяхната оптимизация за търсещите машини:

1. Основната навигация е най-добре да се изгражда с CSS (Cascading Style Sheets) и да е текстова, т. е. добре е да се избягват графичните или флаш бутони.
2. Линковете, включени в основната навигация, следва да са интуитивни, разбираеми и кратки, както и добре разпределени в категории.
3. Обикновено долу на страницата се повтаря основната навигация, като тук може да се използват и по-разширени описания. Тази навигация е важна за търсещите машини.
4. Допълнителната навигация обикновено е по-дребна и се поставя най-долу, при затварянето на страницата. Тя включва линкове към страниците от типа „Карта на сайта“, „Контакт“, „Отговорности“.

Дизайнът – как ще изглежда сайтът?

Дизайнът на сайта е ключов момент в неговото възприемане от хората, за които е предназначен, или както казва поговорката – „По дрехите посрещат!“ Сложността в случая произтича от факта, че дизайнът не е самоцел, а е свързан и зависи от редица фактори, като това какви са целите и задачите на сайта, за кого е предназначен, какви са техническите изисквания, решения, структура и информационна архитектура, функционалности. Дизайнът е в зависимост и от даденостите извън конкретния уеб сайт, като например логото на фирмата, корпоративния дизайн и идентичност, съществуват ли изисквания по отношения на дизайна, цветове, стил и т. н. Всичко това прави задачата на дизайнерите на сайта нелека.

Дизайнът и за кого се прави той. За кого да мислим най-вече, създавайки дизайна на сайта? Имаме няколко възможности да изберем за кого правим сайта и съответно неговата видима част – дизайна. Най-грешната е да мислим за себе си, да направим нещо, което ние харесваме, и да вярваме, че то ще се хареса на всички останали. Разбира се, в процеса на творческа работа, каквато е създаването на дизайн, е задължително личното удовлетворение на дизайнера и това той да харесва работата си, но работата следва да подчинена и на всички останали елементи, цитирани по-горе. Друга голяма и често допускана грешка е дизайнът да се създава „по вкуса на шефа“ и за него.

Отново, разбира се, ръководителят на фирмата възложител има най-добра представа за бизнеса си, но мнението му не бива да е водещо и сайтът да се разработва единствено по неговия вкус. За жалост това е доста често срещана практика у нас. Най-вероятно решение е по време на създаване на сайта основният фокус да бъде насочен към потребителя – човека, който ще посещава нашия сайт, т. е. нашата публика. Често срещаща грешка е работата по сайта да започне директно с дизайна или да се провежда конкурс за избор на изпълнител на уеб сайта, като от кандидатите се изисква предложение за дизайн. Защо този подход е грешен? Дизайнът е, разбира се, много важен, но е произведен на множество други фактори, определяни в подготвителния етап. Целите и задачите на сайта, неговата аудитория определят съдържанието и информационната архитектура, след което идва ред на дизайна. Ако се започне директно с дизайна, могат да се пропуснат важни елементи, които биха имали ключова роля в работата на сайта в бъдеще при изпълнението на неговите цели.

Стилът. Какъв ще е стилът на сайта, зависи от множество фактори – от типа на нашия бизнес, от нашите публики и техните вкусове, от целите, които сме си поставили да изпълним с този сайт. Важното е, изберем ли стил, да го следваме и да го запазим консистентен в целия сайт. Стилът се отнася както до визуалните елементи и тяхната цялост, така и до текстовете, съдържанието на сайта.

Снимките. Сред въпросите, които идват след отговор на описаните по-горе, са и тези какви снимки или други графични елементи ще се използват – дали има вече закупени, дали ще се снимат и разработват специално за сайта, дали ще се използват вече разработени графични решения от други носители, като например каталози или мултимедийни представяния, и как те ще се адаптират за сайта. Често срещана грешка е да се залага на безплатни или случайно намерени в Интернет снимки, които не са с уредени авторски права. Това със сигурност не подхожда на един държач на себе си бизнес и неговата онлайн презентация в рамките на корпоративния уеб сайт.

Анимираните елементи. Понякога има залитане по отношение на анимираните елементи или цялостните Flash решения. Важно е да се знае, че човешкото възприятие има определени граници, и ако искаме да фокусираме вниманието на посетителя на сайта върху конкретни послания, е добре да анимираме или тях, или да не анимираме прекалено много елементи в сайта. В този смисъл – движение на сайта - да, но умерено и с мисъл.

Простотата. „Далай Лама веднъж каза, че простотата е ключът към щастието в модерния свят. Тази философия може да се приспособи към областта на уеб дизайна и *дизайна на дигитални интерфейси*“.

Така започва статията „Направи го просто, тъпако!“ на Пар Амквист (цялата статия е на адрес www.groovemanifesto.net). Простотата е изключително важна в много аспекти на днешния живот, сам по себе си претоварен от комуникации и информация. Но всички ще се съгласят, че постигането на простотата е трудно. В този смисъл дизайнът не бива да се натрапва и да доминира, а следва да изведе в правилна емоционална посока посетителите и да позволи на цялото съдържание, и най-вече на важното от него да изпъкне по адекватен начин.

Начална страница. Началната страница на сайта, или т. нар. „Home Page“ е ключова и трудна за изграждане, доколкото следва да наложи стила на целия сайт, да включи в себе си всички основни комуникационни елементи – лого, основна и помощна навигация, акценти, евентуално новини, промоции и др. Без да е решаващо правило, повечето начални страници на корпоративни сайтове се стремят да обхванат цялото съдържание в един екран, т. е. без вертикален скрол. Това често е нелека задача. При другите типове сайтове това правило не съществува и често виждаме съдържание разгърнато вертикално в няколко екрана.

Вътрешни страници и темплейти. Изграждането на дизайна на началната страница е ключов, доколкото диктува като стил и архитектура дизайна на вътрешните страници. Обикновено се изграждат темплейти за различните типове вътрешни страници, които са производни един на друг и взаимно свързани, с малки различаващи се компоненти в зависимост от необходимостите на конкретното ниво. Основните комуникационни елементи – лого и навигация, остават в най-честия случай не променени на първа, както и на вътрешните страници.

Функционалност. Какво ще прави сайтът?

Функционалността е тази, която различава уеб сайта от брошурата, презентацията, мултимедийния диск. Тя прави сайта интерактивен. Различни функционални модули и

елементи дават възможност потребителят на сайта (посетител, клиент или администратор) да взаимодейства с неговото съдържание и да изпълни конкретна задача (да си плати сметката за тока онлайн, да си купи пица, да се свърже с конкретна фирма или неин служител, да кандидатства за работа, да остави обратна връзка, да изтегли документи, които по-късно да попълни, и т. н., и т. н.). Функционалните модули в един сайт могат да бъдат много и различни, всеки се среща в множество вариации според конкретните изисквания на проекта. Затова е трудно да се представят изчерпателно всички варианти и възможности.

1.1. Единен дизайн, използваемост и маркетинг

Уебсайтовете, ориентирани към потребителя, са проектирани да отговарят на нуждите на целевата аудитория. Интуитивната, лесна за използване навигационна система е основен елемент. Дизайнът на ориентирани към потребителя уебсайтове има следните характеристики:

- Проектиран за конкретна целева аудитория.
- Фокусира се върху съдържанието, а не върху технологията.
- Предлага лесна за използване навигационна система.
- Организира съдържанието чрез колони и други разделители.
- Събира информация чрез лесни за използване формуляри.
- Използва CSS (каскадни стилкови таблици).



Фиг. 12 Уебдизайн, ориентиран към потребителя

Планирането на сайта винаги се извършва преди каквато и да е разработка. С помощта на потребителя сайтът може да бъде изграден така, че да включва функции, които са най-важни за клиентите.

Проучването на желанията на потребителя и прилагането на идеите в дизайна на сайта ще повишат удовлетвореността и често увеличават печалбите на комерсиалните уебсайтове.

Разходите за поддръжка на компанията са

намалени, защото има по-малко телефонни обаждания и имейл съобщения, изпратени до бюрата за помощ (helpdesk) от клиенти, които срещат трудности в сайта.

Следните стилове дизайн са проблематични и крайният продукт неизменно ще представлява слаб уебсайт.

- **Фирмено ориентирани (Company-centered).** Тези уебсайтове се фокусират върху интересите на компанията и служителите или собствениците вземат решение относно тяхното съдържание. Проблемът, който често възниква, е, че това, което те мислят, че трябва да присъства в сайта, не отговаря на нуждите на клиента. Често продуктите и услугите се пренебрегват за сметка на корпоративната структура.

- **Технологично ориентирани (Technology-centered).** Тези уебсайтове често изглеждат, сякаш са създадени от инженери или много неопитни дизайнери. Очевидни са недостатъчното проучване и щитиране и в резултат - недостатъчно информация, която да е на разположение на клиента.

- **Дизайнерски ориентирани (Designer-centered).** Тези уебсайтове са витрини за работата на дизайнерите, но често не успяват да задоволят нуждите нито на клиента, нито на купувача. Важно е да запомните, че работата на един дизайнер на уебсайт е да

улесни комуникацията между клиента си и купувача/потребителя. Потърсете информация от потребителя и се фокусирайте върху нуждите на клиента.

1.2. Влиянието на дизайна върху използваемостта и достъпността.

Модели в уебдизайна

Проектирането на уебсайтове е трудна работа и изисква проучване. Процесът на проектирането може да се направи по-лесно управляем, като проектът се разбие на по-малки, по-лесно контролируеми задачи. Моделът на дизайн е формален начин за документиране и записване на стъпките за разрешаване на проблеми с дизайна. Важен аспект от този процес е да се идентифицират и документират основните качества, които правят едно добро решение различно от едно лошо решение. Въпросите, на които трябва да се отговори по време на този процес, са:

- Какви основни компоненти се срещат в един успешен уебсайт?
- Каква информация трябва да предава страницата?
- Каква е аудиторията?
- Има ли други съображения относно дизайна?

След като се отговори на тези въпроси и резултатите се документират, дизайнерът трябва да определи кои функции ще задоволят нуждите и дали някои от тях могат да се съчетаят и да се комбинират. След това дизайнерът трябва да определи приоритета на тези основни компоненти, за да реши кои от тях са важни и кои може да се смятат за „допълнителни“, които би било хубаво да се включат, ако времето и други ресурси го позволяват.

Може да бъдат обобщени няколко качества за един добър уебсайт. Всички страници трябва да са представени с описателни заглавия и линкове в началната. Навигацията трябва да е проста и интуитивна. Графиките трябва да се използват по подходящ начин и програмният код да отговаря на стандартите. Уебсайтът трябва да е достъпен за всички посетители. Трябва да са включени подходящи метатегети (meta-tags) за оптимизиране на работата на търсачките.

Следващата стъпка е да се очертае реалният „облик“ на сайта. Той може да включва лични стилове и стратегии за дизайн. Това започва с груб проект на сайта, след което дизайнерът прецизира схемата върху действителен чертеж, които включва определени изображения, навигационна система, съдържанието и долните колонтитули (footers).

1.3. Организация на сайта

Организацията на един уебсайт включва внимателно планиране. Целта на всеки сайт е да има завръщане на посетители. Съдържанието трябва да е полезно, лесно за намиране и правилно. Съдържанието е най-важната част във всеки уебсайт. Организацията, включително възможността за достъп до всички части в сайта, е на второ място. Първата страница на всеки уебсайт се нарича начална (homepage). Началната страница трябва ясно да очертава целите на сайта, да предоставя навигация до основни подстраници и да включва информация за контакти, както и кога за последно е актуализирана.

След като съдържанието на целия сайт е било избрано, частите (chunks) трябва бъдат организирани в карта на сайта, която показва взаимовръзките между тях. Проектира се навигационната система на сайта, която включва бутони или връзки към основните страници и е последователна. Това ще позволи на посетителите лесно да се придвижват и ще им създаде усещане за подреденост на сайта. Ако навигационните връзки са бутони, на страницата трябва да се включат и текстови връзки. Това ще даде възможност на хора с увредено зрение да използват четец на уебстраници или на тези,

които са деактивирали изображенията в браузъра си, да се придвижват в сайта с помощта на текстовите връзки.

Добър тест за използваемост на един сайт е да установите дали от началната страница може да се стигне до всяка подстраница с не повече от три кликания на мишката. Ако това не може да се направи, **сайтът трябва да се реорганизира**. Многократното кликане в сайта и ненамирането на търсената информация или попадането в задънена улица ще разочарова потребителя и ще намали възможността за повторно посещение.

1.4. Мерки и насоки за използваемост

Жизненият цикъл на създаването на уебсайт е идентичен с този на традиционния софтуер: спецификации на изискванията, анализ, дизайн, осъществяване, тестване и внедряване.

Използваемостта може да се дефинира със степента, в която даден софтуер помага на човека, стоящ зад клавиатурата, да изпълни някаква задача, а не да се превърне в допълнителна пречка за изпълнението ѝ. Използваемостта често се оценява с помощта на следните основни критерии:

- Лесно за учене (Колко лесно е за един посетител, посещаващ сайта за пръв път, да получи желаната информация?).
- Съхраняване на наученото във времето (Когато посетителите се върнат в сайта, колко лесно могат да възстановят уменията си за навигация в него?).
- Скорост на изпълнение на задачите (Колко бързо даден посетител може да завърши определена задача?).
- Процент на грешки (Колко често посетителите правят грешки в сайта? Колко време им отнема да се възстановят от тези грешки?).
- Субективна удовлетвореност на потребителите (Докладват ли посетителите, че е лесно да се навигира в сайта?).

Ето три основни правила, които може да следвате за добра използваемост:

- Хората искат добре проектиран уебсайт. Те имат много малко търпение за сървърни грешки или за проблеми с разработката.
- Превъртането (scrolling) трябва да се избягва. Информацията, която не е на първия екран, често се пренебрегва.
- Потребителите не четат. Изследванията са установили, че степента на четене на компютърен монитор е с 25 % по-бавна отколкото на хартия. Това налага да пишем с 50 % по-малко, съдържанието да се рационализира и несъществените подробности да се отстранят.

1.5. Достъпност

Почти всички потребители харесват достъпни уебстраници, които позволяват оразмеряване на текста, гъвкави ширини, мишени за лесно кликане и смислена терминология.

2. ПРОЦЕС НА РАЗРАБОТКА „ИТЕРАТИВЕН ДИЗАЙН“

Едно от най-важните неща, които трябва да запомните в процеса на разработка, е фокусът ви постоянно да е насочен към нуждите на клиентите и крайните потребители. Итеративният дизайн предполага работа, преразглеждане и преработване на сайта, докато нуждите на клиента и потребителя бъдат изпълнени. Ключът към итеративния дизайн е бързото създаване на прототип, който ще осигури обратна връзка от всички заинтересовани страни. Той трябва да бъде достатъчно гъвкав, за да може лесно да се правят промени. Съществуват три основни стъпки към процеса на итеративен дизайн.

- Дизайн. Среща с клиента, определяне на измерими цели, разработване на концепции и обсъждане на нуждите на потребителите.

- Прототип. Създаване на карта на сайта, сториборд (storyboards) и схеми (wireframes) за уебсайта.
- Оценяване. Оценяване на прототипите, създадени в Стъпка 2; преработете, ако е необходимо.



Фиг. 13 Процес на итеративен дизайн

Ако има съществуващ уебсайт, процесът на итеративен дизайн може да бъде полезен в актуализиране на информацията и поддържане на дизайна в крак с технологичните стандарти.

Пример за този итеративен процес може да се види на адрес <http://www.useit.com/papers/sun/pagedesign.html>.

2.1. Етапи в процеса на дизайн

Процесът на дизайн по принцип следва много организиран модел на действие. Четирите етапа - откритие, разработване на стойностно предложение, проучване и бързо създаване на прототип, са типични дейности, които един уебдизайнер извършва при работа с клиента. Всеки етап има специфични цели и е от решаващо значение за ефективното и успешно развитие на сайта.

Започнете с определяне на целите на уебсайта. Какви са общите цели на уебсайта?

Определете тези цели, като проучите клиента и работите с екипа по разработката.

Трябва да познавате потребителите. В рамките на проучването трябва да определите отговорите на следните въпроси по отношение на потребителя:

- **Хора/Роли.** Кой ще посещава уебсайта? Разбиране на нуждите на потребителите. Можете да използвате тази информация, за да определите какво ще бъде включено в сайта.
- **Задачи.** Какво послание иска да предаде Вашият клиент чрез сайта? След като нуждите на клиента са установени, разработчикът трябва да създаде това съдържание, за да достави посланието.
- **Технологичен достъп.** Как ще бъде разпространявано посланието? Какви технологични инструменти ще бъдат използвани, за да се позволи на потребителите да взаимодействат с уебсайта или да получат информация?
- **Социални въпроси.** Защо потребителите ще посещават уебсайта? Наложително е да се познава и разбира ио-големият социален и организационен контекст, в който те работят, играят или живеят (социални въпроси).
- **Контекст/Сценарии.** Може да бъдат използвани истории или сценарии за проучване на целите и дейностите на вероятните потребители. Посетителите на уебсайта оценяват това. Ако потребителите усетят, че компанията е съпричастна към нуждите им, те ще бъдат по-склонни да посетят сайта и да останат по-дълго.

Планиране на стилове, граматика и нива на четене. Често е лесно да се пренебрегне очевидното. Винаги имайте предвид целевата аудитория, докато разработвате съдържанието за уебсайта. Важно е да комуникирате с аудиторията по начин, който е важен за нея и на подходящо ниво на четене. Винаги проверявайте граматиката и правописа. Нищо няма да отблъсне един посетител по-бързо от непохватен уебсайт с грешно написани думи.

В края на етап откриване дизайнерът трябва да има ясна представа за очакванията на клиента. Определете обхвата на проекта и проучете аудиторията, която ще посещава сайта. Оценете други сайтове, които допълват или са конкуренция на клиента.

- **Разработване на стойностно предложение**

Това е кратко и точно изявление, което отличава продуктите на клиента от тези на конкуренцията. То трябва да включва какво иска клиентът и кои са продуктите. Накратко уникалното стойностно предложение отговаря на въпроса „Защо трябва да купувам от вас, а не от конкуренцията?“.

Целта на уебсайта е да се направи така, че продуктът или услугата на клиента да се отличава от останалите. Създайте непреодолимо послание, което не може да се изтълкува погрешно.

Проучване

Проучете различни варианти за уебдизайн, като генерирате множество груби дизайни, от които ще бъде избран един за по-нататъшна разработка. Работете съвместно с членове от екипа и винаги комуникирайте с клиента, за да сте сигурни, че неговите очаквания са изпълнени.

Генериране на множество дизайни:

- Визуализиране на решения за открити проблеми
- Създаване на дизайн, базиран на информация и навигация
- Разработване на ранен графичен дизайн.

- **Бързо създаване на прототип**

Разработване на модел за уебсайта, който ще илюстрира дизайна. Това ще позволи обратна връзка с клиента, така че лесно да бъдат направени промени. Разработването на прототип в ранен етап от процеса позволява обратна връзка с клиента. По-късно може да бъде включен потребителският интерфейс.

Клиентите искат да знаят как ще изглежда крайният продукт. Работата на дизайнера е да превърне тези идеи в реалност. Концептуалната работа е много трудна, но има някои модели, които позволяват итеративност и гъвкавост на дизайна.

Основният процес протича по следния начин: проучване, мислене, скициране на дизайн, преосмисляне, преразглеждане на скицата (повторете толкова пъти, колкото можете да си позволите), след това създайте! Целта е да научите колкото се може повече в началото на един проект чрез „предварително проучване“, модели на дизайн и разговор между участващите страни.

Някои основни модели, използвани от повечето дизайнери, включват карти на сайта (site maps), сториборд (storyboards) и схеми (wireframes). Въпреки че дефинициите лесно може да се припокриват, има малки разлики и всички те може да помогнат в процеса на разработване на дизайна.

- Карта на сайта (**site map**) е текстово изложение на целия сайт.
- Сториборд (**storyboard**) е графично изложение на целия сайт. Може да бъде на хартия или да се създаде във всяка програма за графичен дизайн. Често програмата ще предостави изглед на карта на сайта, за да илюстрира навигацията.

- Схемата (**wireframe**) представя конкретните страници, които трябва да бъдат създадени в уебсайта. Всяка страница е представена чрез чертеж, който показва информационните компоненти, които ще се появят на екрана, особено навигацията. Схемата може да предложи основно визуално оформление, но не се ангажира с конкретен външен вид.

2.2. Карти на сайт (Site Maps)

При по-големи сайтове често се използва карта на сайта (site map), която позволява на посетителите да се ориентират лесно и да имат достъп до цялото съдържание. Картата наподобява „съдържанието“ на книгите и затова е по-лесно разбираема. Понякога картата се извежда в отделна страница или в нов малък прозорец (pop - up). Добре е да има връзка към нея от всички страници или най-малкото от началната страница.

Интерактивна карта

Карта на местоположението срещахме почти винаги до адреса на офиса или магазина в секция „Контакти“ или „Къде да ни намерите“. Когато обаче бизнесът е представен в повече градове и дори с няколко обекта в едно населено място, информацията за местоположението и съответно адресите и телефоните за контакт трябва да се организират по по-удобен начин.

Едно от практичните решения е създаването на интерактивна карта-обикновено на държавата, разделена на региони и с указани графично точките на присъствие. Картата е интерактивна, защото изисква потребителят да избере конкретно населено място, за да види информацията за контакт с местния офис.

Най-често подобно решение се изгражда на Flash, свързано с админ панела и база данни, в която се въвеждат обектите.

Понякога при по-сложно географско местонахождение се интегрира решение с Google Maps, което позволява възможности за увеличение и други, предоставяни от картите на Google. Друга разновидност на **интерактивната карта** е карта, с която потребителят взаимодейства по отношение на изгледа ѝ – приближава, отдалечава, променя мащаба, търси конкретен адрес или зона и т. н.

- Картите на сайта са едни от най-важните и широко използвани инструменти за уеб дизайн. Тъй като показват цялостната структура и йерархия на уебсайта, те се използват за разработване на оформлението и предоставят рамката (framework), върху която да се базира навигацията. Започнете, като начертаете груба карта на сайта. Включете всички планирани страници. Не включвайте специфични подробности в картата на сайта. Очаквайте дизайнът да се развива по време на проекта.
- Друга полезна практика е да се създаде HTML карта на сайта, която може да се използва в завършения уебсайт. Това е особено полезно за много големи сайтове. HTML картата на сайт може да се състои от една или много страници, които показват всички или най-важните страници. HTML картите на сайт са най-често текстови, но понякога са графични. Всяка страница е представена от заглавието си, което всъщност е връзка към нея. Създайте добро описание за заглавията (headings) в картата на сайта. Добрите заглавия за всички връзки във вашия уебсайт също помагат. Свържете повечето страници на сайта към HTML картата на сайта, за да му придадете високо ниво на значимост за търсачките.
- Картите на сайт са особено полезни в ситуации, когато потребителите нямат достъп до всички области на даден уебсайт. В този случай търсачката може да не открие всички подходящи страници. Когато дизайнерите използват AJAX или Flash, търсачките не могат да навигират, за да достигнат до

съдържанието. Това прави картата на сайта много важна и позволява на търсачките да получат информация от наскоро актуализираните страници.

- Имайте предвид, че картите на сайт не заместват механизмите, базирани на обхождане (crawl-based mechanisms), които търсачките използват, за да откриват URL адреси.

Съставяне на карта на сайта

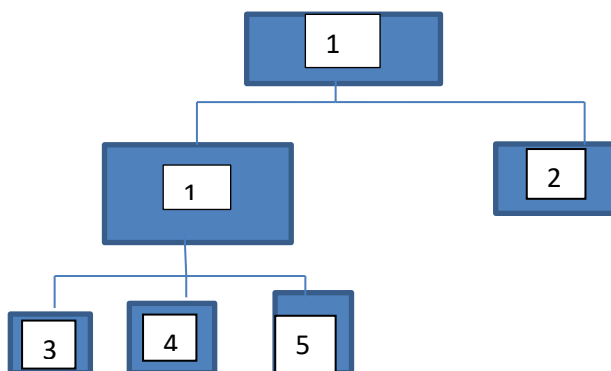
- ✓ Картата на сайта е добро средство за планиране;
- ✓ Страниците се представят с правоъгълници, хипервръзки със стрелки;
- ✓ Картата дава представа за навигационната структура на сайта;
- ✓ Началната страница е най-отгоре;

Картата се създава ръчно или с помощта на софтуер, като се използват графични редактори. Тя не само определя какви страници ще съдържа сайтът, но и неговата структура, т. е. връзките между страниците.

Страниците може да се свързват в различни структури. Най-простата структура е линейната. Използва се за представяне на линейни инструкции. Среща се и при сайтове с учебни материали (страница напред – страница назад).

Примери:

1. Йерархична структура на сайт (най-често срещаната)



2. Линейна структура



Работата върху съдържанието на страниците

Това включва създаване на всички компоненти на страниците – текст, изображения, аудио, видео, таблици, хипервръзки и т. н.

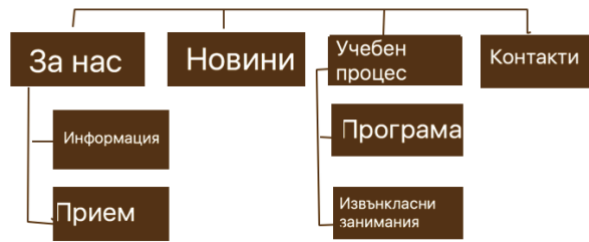
След тази стъпка сайтът е готов. Той се тества от разработчиците и от група потребители. Когато е готов окончателно, сайтът се публикува. С това работата не е приключила. Сайтът изисква определена поддръжка – да се следи за неговата посещаемост, за реакцията на аудиторията, да се актуализира съдържанието и др. Обявяването на датата на последната актуализация е добра практика, защото буди доверие у посетителите.

И така, за да се създаде един добър несложен сайт, е необходимо да започнете с неговото планиране, да намерите точните средства за реализация и подходящото естетическо оформление. Не забравяйте да го тествате за грешки.

След публикуването сайтът става видим и за други потребители в уебпространството. Има различни начини да привлечете вниманието към него. Вече се запознахте с ролята на метатаговете. Добавете и възможностите за реклама. Ето и някои начини:

- ✓ Разпространете картички, календари, флаери с адреси на сайта;
- ✓ Включете адреса на сайта като елемент на подписа, съпровождащ изпращаните от вас съобщения чрез електронна поща;
- ✓ Стремете се да има повече хипервръзки от други сайтове към Вашия;
- ✓ Използвайте възможността да рекламирате Вашия сайт в други сайтове, особено в безплатни;

Много има още за сайтовете и техния интерактивен интерфейс. Но най-важното е да се оформят така, че да създават настроение на тези, които го посещават, а също и на Вас самите.



Фиг. 14 Примерна карта на сайт от училище

- **Сториборд (Storyboards)**
- В допълнение към картата на сайта сторибордът често се използва, за да помогне при дизайна, бюджета и избора на технология. Добрият сториборд може да даде ясна представа за обхвата на сайта на всички, които разработват проекта.

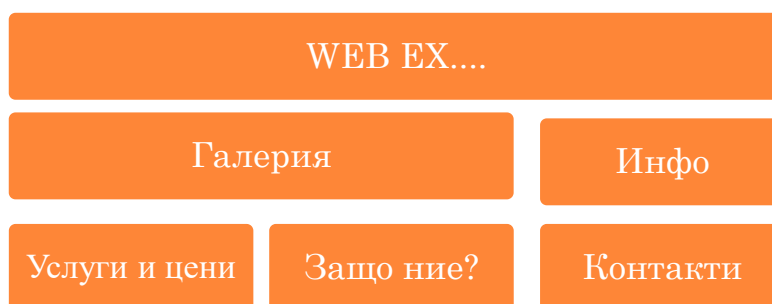
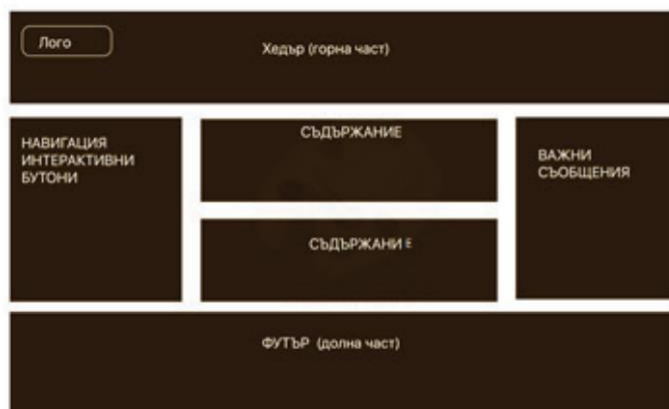
Често едно информативно, а не декоративно оформление на сайта се изгражда с помощта на основни HTML страници. Въпреки че сторибордът не включва всяка отделна страница от сайта, той покрива основните функционални области и главни раздели. Всяка страница в сториборда ще включва основната навигация, полезни връзки (related links), функционално описание на проектираната страница и някои бележки относно съдържанието ѝ. Въпреки че сторибордът не показва елементи на дизайн, той включва всички основни елементи, които ще се показват на всяка страница.

Подравняването на местоположението на тези елементи може да даде по-добра визия на крайния сайт. Всяка от страниците ще има HTML файл. Въпреки че страниците са ограничени откъм детайли, един сториборд предлага възможност на клиента да изпробва първоначално сайта. Тези страници могат да бъдат създадени с помощта на хартия/молив или във всяко графично приложение. Не е необходимо да бъдат показвани конкретни изображения на тези страници, но трябва да има очевидна тема или функция на дизайна, която да отразява последователността между страниците.

2.3. Схеми (Wireframes)

В уебдизайна схемата (wireframe) е опростено изображение на елементите, които трябва да присъстват на всяка страница от уебсайта или на екрана на програмата. Основното за тези схеми е да бъдат кратки и достъпни. Схемите са мястото за експериментиране с нови идеи, тестване и адаптиране, а не за показване на обширни детайли. Те са прости рисунки, които показват положението на елементите в уебстраницата. Схемите позволяват на клиента ви да се фокусира върху оформлението, без да бъде разсейван от изображения, стил на шрифта и други елементи на дизайна. Клиентът може да се концентрира върху съдържанието на всяка страница и мястото, което всеки елемент ще използва. Много по-евтино е да промените дизайна на този етап, отколкото след като сте разработили детайлите.

Важните елементи за уебсайта трябва да се виждат в схемата. Използвайте прости форми вместо реални графики и ги обозначете. Включете навигация, лого на компанията или банер за сайта и всички области със съдържание. Може да искате да включите поле за търсене (search box) или област за вход (user login) на потребителя.



Фиг. 15 Примери за схема на уебстраница

Вижте снимки на схеми на страница, създадени от професионален уебдизайнер, на адрес <http://deeplinking.net/paper-web/>.

2.4. Усъвършенстване

След като дизайнът на уебсайта е разработен, започва усъвършенстването. Този процес също е **итеративен**. Схемата е отправна точка в процеса за избиране на облика на заглавия (headers) и текст, изглед на изображения и цветови модели. След като всички страни са работили върху усъвършенстването на сайта, трябва да има подробен прототип на уебсайта и сторибордът и схемите ще се обогатят с много повече детайли. Процесът на усъвършенстване включва много стъпки за оценяване.

- **Експертни мнения - „Евристична оценка“**

Целта на евристичното оценяване е да се намерят проблемите на използваемостта в дизайна, за да може да бъдат коригирани по време на итеративния процес.

Евристичната оценка включва оценители, които преглеждат интерфейса и преценяват съответствието му с приетите принципи за използваемост. Евристичното оценяване е трудно за един човек, защото той никога няма да успее да открие всички проблеми, свързани с използваемостта на един интерфейс. Различни хора откриват различни проблеми с използваемостта. Обикновено се препоръчва от трима до петима души да оценяват един сайт, тъй като при по-голям брой оценители често не се придобива допълнителна информация.

- **Корекции и проверка за грешки**

Тестови формуляри и други интерактивни елементи - ако сайтът включва формуляри, анкети, проучвания или често задавани въпроси, покажете на клиента как да ги използва. Включете план за актуализиране на тези интерактивни елементи.

■ **Неформално оценяване „Мислете на глас“**

Когато двама души си сътрудничат при употребата на един уебсайт, проблемите му може много бързо да станат очевидни. Единият човек използва сайта и устно обсъжда процеса, докато другият записва коментарите и насърчава първия да изразява своите мисли и въпроси. Единственият начин да разберете дали дизайнът работи наистина е да помолите няколко души да го използват и да ви докладват какво мислят, докато го правят. Основните грешки са очевидни веднага, а по-малко очевидните може да бъдат допълнително обсъдени с повече дискусии с оценителите.

■ **Официални проучвания за използваемостта „Окончателни данни“**

Тестването за използваемост е процес, чрез който характеристиките на взаимодействието човек - компютър на една система се измерват аналитично и се определят слабостите, които трябва да се коригират. Регистрите на стандартни уебсървъри са ценен източник на информация относно моделите на използване, след като един уебсайт е качен на даден сървър. В този етап оценителите трябва да намерят експерти по използваемост или представители на потребители; сесиите с реални потребители се отразяват с големи статистически подробности и са материал за анализи. Подобно тестване може да бъде строго структурирано и доста скъпо. След всичките използвани стратегии за усъвършенстване сайтът трябва да се публикува в непублично пространство, за да може клиентът да го тества. Сайтът трябва да се прегледа на няколко различни компютъра, на различни размери монитори и е различни браузъри. Изключително важно е външният вид и обликът на сайта да останат последователни във всички сценарии на преглед и от страница на страница. Всички промени трябва да бъдат завършени по време на етапа на усъвършенстване, преди действителното публикуване на сайта.

2.5. Писане за ефективна комуникация в Мрежата

Има много правила за „добро“ писане. Те включват правилен правопис и граматика, както и интересно съдържание. Обаче за да пишат в Мрежата, авторите трябва да са наясно как хората четат уебстраници, както и с естеството на хипертекста сам по себе си. Писането за Мрежата коренно се различава от това за печатни медии. Потребителите на Мрежата по принцип сканират съдържанието, а не четат всяка дума. Дизайнерите трябва да пишат сбито, след това да съкратят, редактират и перифразират. Класическото академично писане следва пирамидална структура. Първо полага основите, след което подкрепя информацията от проучвания или данни и накрая следва заключение. Този стил не работи за уебсъдържание. Работният процес е обърнат, като първо се поставя важната информация. Следват детайлите, като се позволява на потребителя сам да реши дали да прочете останалата част от страницата. Когато една страница е по-дълга от типичния екран на монитор, поставянето на важната информация на първо място е от съществено значение. Изследванията показват, че повечето потребители не превъртат (scroll). Пишете от гледната точка на посетител и отговаряйте на въпроси, които са в съзнанието на потребителя. Групирайте съдържанието в малки късчета информация. Това прави текста по-лесен за четене в уебпространството. Използвайте функцията на заглавието (headline) във ваша полза. Атрактивното заглавие в горната част на страницата не само привлича вниманието на читателя - доброто

използване на заглавия ще позволи на браузъра да открие и категоризира сайта Ви правилно. Заглавията в рамките на един документ подобряват способността на читателя да сканира сайта. Техниките, които са в помощ на комуникацията, включват маркиране на важен текст, добавяне на хипертекстови връзки и различни шрифтове и цветове. Поставете думи за действие като абониране (subscribe), направи поръчка (place order) и излез (quit) в началото на фрази или изречения. Активният глагол помага да представите по-бързо гледната си точка, подобрява яснотата и води до по-кратки изречения. **Наблегнете на фактите.** Не смятайте, че читателят внимава. Един читател прекарва много кратко време в един сайт при първо посещение, затова изложете фактите ясно и отчетливо.

Потребителите са склонни да четат във F модел и да се фокусират върху информацията, която е представена под формата на списък с водещи символи (bulleted lists).

Качественото писане за Мрежата включва подходящо използване на хипертекстови връзки. Обобщената информация за всяка връзка е от съществено значение. Давайте достатъчно информация, за да гарантирате, че читателят ще следва връзките, които действително я съдържат. Може да бъде включена допълнителна информация, но рискувате да загубите потребителя, когато връзката го отведе към друг уебсайт. Чрез включването на полезни връзки вие показвате на читателите, че сте направили някои проучвания и сте готови да им позволите да проучат и други гледни точки.

Сканируемите, стегнати и обективни стилове на писане въздействат положително на способността на потребителите на Мрежата да навигират в даден уебсайт и да останат доволни. Всеки уебсайт се конкурира с хиляди други за вниманието на читателя. **Всяка дума има значение.**

3. ОСИГУРЯВАНЕ НА КАЧЕСТВОТО

След като вече сте създали уебсайт, нямате търпение да покажете своето произведение в интернет. Въпреки това, преди да позволите на други да видят Вашия сайт се уверете, че той покрива определени стандарти. Тази глава ще ви преведе през тестването, необходимо за осигуряване на качествен сайт.

3.1. Техническо тестване

Преди да покажете своя уебсайт на света, уверете се, че той функционира правилно:

- Водят ли връзките зрителя до местата по предназначение?
- Вашият сайт еднакво ли изглежда в Internet Explorer и във Firefox или в другите уеббраузъри?
- Може ли вашият сайт да бъде разглеждан при различни размери и разделителни способности на екрана, без това да променя въздействието му?
- Проверихте ли си още веднъж за последно правописа и граматиката?

Едно от най-лесните неща за тестване на вашия уебсайт е точността на връзките. За да го направите, трябва да разгледате сайта си в режим за преглед в браузър, за да видите как ще изглежда, когато бъде публикуван в интернет.

Докато преглеждате своя сайт, отговорете на въпроса: „Изглежда ли той по начина, по който планирахте?“. Нахвърляйте си набързо бележки за промените, които може да поискате да направите на сайта, преди да го публикувате в интернет.

След това тествайте бутоните и връзките. Това може да отнеме известно време в зависимост от броя им, понеже ще трябва да проверите всички бутони и връзки на всяка страница. Щракнете върху всеки от тях, все едно че сте посетител в сайта. Води ли ви всеки бутон или връзка до очакваното местоназначение? Отново си запишете кои бутони или връзки все още изискват допълнителна работа. Ще трябва също да

прегледате бутоните и връзките, за да определите дали адекватно описват къде биват прехвърлени посетителите. Видно ли е за зрителите дали се придвижват към друга страница в рамките на вашия сайт, или го напускат и отиват към друг? Ако сте използвали повторение и единство в рамките на Вашия сайт, потребителите трябва да могат да кажат кога са в него и кога го напускат. Предоставили ли сте на посетителите възможност за лесно връщане към конкретна страница в рамките на вашия сайт? Едновременно с тестването на бутоните и връзките направете проверка, за да се уверите, че всички изображения се появяват според предназначението си. Преименуването и преместването на файловете с изображения може да създадат проблеми, затова внимавайте, когато го правите. Водете си бележки, ако някое изображение не се показва според очакванията и е заменено от червен X на страницата.

Ако вашият сайт премине успешно тестването на връзките и бутоните, опитайте да го разгледате в други уеббраузъри. Сред по-популярните са Internet Explorer, Mozilla Firefox, Opera, Safari.

Уверете се, че тествате сайта си, за да видите как ще изглежда той при различните разделителни способности. Този процес е много сходен с предните два, които вече завършихте.

Например може да видите следните възможности за предварителен преглед на вашия сайт:

- Internet Explorer (640 x 480)
- Internet Explorer (800 x 600)
- Internet Explorer (1024 x 768).

Числата след уеббраузърите се отнасят за броя на пикселите, които се показват на монитора. Разгледайте вашия сайт във всяка от тези разделителни способности за екрана, за да откриете евентуални вариации в изгледа. Ако не сте сигурни за разделителната способност за екрана на Вашата целева аудитория, най-добрият избор би бил 800 x 600, тъй като тя може да бъде визуализирана на екраните с по-висока разделителна способност и най-често се среща при по-старите монитори.

Имайте предвид своята аудитория при всеки от техническите тестове, използвани за оценяването на Вашия уебсайт. Кой ще го използва? Този сайт за световна аудитория ли го създавате? Ще поискате да се уверите, че Вашата аудитория ще получи възможно най-доброто преживяване, когато посети сайта Ви. Настройте своя сайт, за да отговаря на потребностите на аудиторията.

Една от последните проверки, които винаги трябва да правите, преди да публикувате своя сайт, е да прегледате отново **правописа и граматиката**. По-голямата част от програмите за редактиране на уебсайтове използват средство за проверка на правописа и граматиката. Използвайте този инструмент дори когато вече няколко пъти сте го правили. Внасянето само на една малка промяна може да създаде грешка в правописа или граматиката, за която по-късно ще съжалявате. Грешките в правописа и граматиката създават лошо първо впечатление и намаляват доверието към уебсайта. След като завършите техническото тестване, използвайте бележките си, за да коригирате грешките.

3.2. Тестване на използваемостта

Тестването на използваемостта ви помага да откриете как другите ще реагират на сайта ви и как ще го използват. Вие искате вашият сайт да бъде лесен за ползване и посетителите да се връщат отново и отново.

Преди да започнете да тествате използваемостта на сайта, трябва да прегледате принципите, набелязани от Вас при първоначалното му създаване. Какво се опитвахте да постигнете с уебсайта? Искате ли да информирате, да забавлявате или да продавате

нещо на човека, който разглежда вашия сайт? Трябва да се фокусирате върху тези неща при тестването на използваемостта на сайта. Дали зрителят е информиран, забавляван или убеден да купи нещо?

Най-добрият възможен начин за тестване на използваемостта на сайта, е да намерите някого, който е представител на целевата аудитория, никога не е виждал сайта и не знае нищо за предназначението му. Този човек ще играе ролята на *зрител изпитател* на Вашия сайт. Накарайте го да седне и да разгледа сайта в режим „Преглед“. Вашата роля в този процес е тихо да стоите близо и да си водите записки за неговото преживяване. Трябва да го помолите да говори на глас за сайта, докато го разглежда. Като дизайнер не трябва да отговаряте на въпроси, отнасящи се до сайта, но трябва да запишете коментарите, направени от зрителя изпитател. Помнете, Вие искате зрителят изпитател да разглежда сайта без Вашето съдействие, точно както някой в интернет ще го направи. Какви са първите впечатления на зрителя за сайта? Дали разбира защо сте го създали? Кои части на сайта зрителят харесва и кои не разбира? Може ли зрителят лесно да навигира в сайта, или се губи?

Имайте предвид, че зрителят си изгражда впечатление за сайта през първите 30 секунди от разглеждането му. Ако не направите добро впечатление през тези 30 секунди, може да го загубите и той да отиде в друг сайт. Точно тази е причината Вашата начална страница да бъде толкова важна. Тя трябва да дава ясна индикация за предназначението на сайта, да изглежда организирана и да има лесен за ползване и разбиране интерфейс, за да примамва зрителя да продължи да го проучва. Водете си подробни записки относно впечатлението му за началната страница.

Забележете как Той реагира на елементите на дизайна (цвят, размер на шрифта, подчертаване), които сте използвали. Дали тези елементи са полезни и атрактивни за него, проучвайки сайта, или са объркващи или отблъскващи? Помнете, че очите на повечето зрители следват F модел, когато разглеждат даден уебсайт. Дали важната информация е в рамките на този F модел? Забелязва ли я зрителят изпитател? Освен това обръщайте внимание не само на това какво казва зрителят изпитател, а и на местата във Вашия сайт, на които той се колебае или се връща назад по стъпките си. Зрителят може да не Ви каже, ако е объркан, но Вие можете да забележите това по начина, по който реагира на сайта.

Оставете го да разглежда сайта ви най-много 30 минути. Ако му трябва повече от 30 минути, това ще бъде индикация, че трябва да направите някои промени в сайта, преди да го публикувате, за да изясните или да реорганизирате част от информацията. Разгледайте записките, които си направихте. Вземете решение за промените в сайта и ги направете **веднага**.

4. ТЕХНОЛОГИЯТА HTML

Организиране на информацията чрез хипертекст:

Най-привлекателния аспект на Уеб са изградените връзки между сходни по съдържание теми посредством хипертекст – един нелинеен метод за организиране на информацията. Когато използвате хипертекст, можете да прескачате на свързани помежду си теми, бързо да намерите търсената информация и да се върнете в изходната си позиция или да продължите към друга тема. Като създател на хипертекст, вие определяте кои думи да превърнете в хипертекстови връзки и къде ще се озоват потребителите, когато изберат дадена връзка. В Уеб хипервръзките под формата на текст или изображение могат да ви отведат на друга уебстраница или да ви позволят да отворите или изтеглите файл, например музика, изображение, филм

или програма. Макар еднопосочната същност на хипертекстовите връзки да не се е променила от зората на Уеб, същността на съдържанието, към което те отвеждат, се е променила драстично. С превръщането на Уеб във всяка по-богата и интерактивна среда типовете свързано съдържание постоянно се развиват. За да се възползват от всички нови технологии, предлагани от уебсайтовете, често е необходимо потребителите да разполагат с по-бърза интернет връзка от тази, използвана при комутируема (dial-up) линия. Скоростта на Интернет връзката определено влияе върху дизайнерската работа.

HTML е файлов формат, в смисъла, в който това може да се каже например за .doc или .xls. HTML документите са текстови файлове /.txt/, които могат да бъдат прочетени и редактирани с произволен текстов редактор. В тях, изписани с текст, се съдържат команди, предназначени за брауъра-програма, която реализира тяхното съдържание на екрана, под формата на web страница. По тази причина файловото разширение, което се използва за тях е .htm, .html --- по избор.

Това е необходимо, за да бъдат разпознавани като такива, а не като обикновени текстови страници. Следователно всяка web страница може да бъде отворена и създадена с произволен текстов редактор, поддържащ запис в .txt формат.

При стандартния запис в .doc формат Word добавя към самият текст своя системна информация, за това с коя версия се използва текстов редактор от висок клас, трябва да се внимава и при всяко записване да се указва, че файлът от типа „plain text“ е само текст. По тази причина за записването на HTML, се препоръчва използването на най-простите текстови редактори в Windows или /PE2 в DOS/, Note pad++, е по-съвременния текстов редактор.

Добро решение е да се използва някоя от помощните програми за създаване на Web страници, като е необходимо тя да предлага възможност за редактиране и надграждане на самия HTML код.

Такива са например Macromedia Dream Weaver, Home Site, Web Studio Pro...

Има непрофесионални програми, които не следят за HTML код.

Ако искате да видите HTML код на дадена страница, която е отворена в брауъра, можете да изберете менюто View Source. Вероятно това е най-добрият начин да влезете в темата. На професионален език, командите в HTML се наричат „тагове“ и се различават от останалото съдържание на страницата по това, че са оградени от символите „<“ и „>“ .

Когато брауърът приеме за обработка /визуализиране/ една html страница, той я изчита последователно от началото до края.

Таговете, които среща, не се показват на екрана – те биват възприемани като инструкции, указващи дадена функция или формат. Останалото текстово съдържание се извежда на екрана на потребителя, като се съблюдава зададените до този момент в страницата команди и форматиране.

Грешно е да се мисли, че HTML е език, който описва дизайна на страницата. При него се създават единствено описания за типа на данните, за тяхното относително разположение едни спрямо други и за форматирането на включения между таговете текст.

В HTML редовете и интервалите нямат значение. Това означава, че HTML кодът може да бъде изписан на един или повече редове, без това да повлияе на визуализирането му на екрана, което ще зависи единствено от използваните тагове. Понякога е подходящо в HTML кода да се оставят команди или информация, която обаче не бива да се визуализира под никаква форма. За тази цел се използват „коментари“ – началото се дава с <!-- а краят с -->.

Например: <!-- това е коментар -->.

Отличават се от основния текст в HTML файла, по това че започват с „<“ и завършват с „>“, така наречените квадратни скоби. Има два вида HTML тагове – отварящи и затварящи. Отварящия таг съобщава на браузъра, че от това място в страницата започва от него форматиране, а затварящият съобщава, че при него то завършва. В отварящия таг се включват и параметрите, определящи неговото действие. Ако такива не бъдат зададени, за визуализирането се използват стойности по подразбиране, като при някои от таговете задаването на определени параметри е задължително. Според стандарта стойностите, които се присвояват на параметрите, трябва да стоят в кавички. Между отделните параметри и самия таг се оставя по един празен интервал за разделител.

Напр.: .

Затварящият таг, винаги е същия като отварящия, но без параметрите и със знака „<“ веднага след „>“.

Например: .

Таговете могат да се изписват, както с малки, така и с големи букви. Всички браузъри не правят разлика между малки и главни букви /case insensitive/, т. е. Употребата на малки или големи букви не влияе върху работата им. Но прието е, в даден HTML документ да се пише или само с малки или само с главни букви. Вероятно стана ясно, че всеки таг, който е бил отворен, следва да бъде затворен. Нарушаването на това изискване може да доведе до неправилно визуализиране в някои браузъри и дори до показването на екрана на празна страница. Затварянето им е необходимо от една страна, за да се укаже краят на определено действие и от друга – за да се запази структурата на езика.

Всяко правило си има изключение: - следва противоречие.

Таговете могат да бъдат отваряни един в друг, т. е. не е необходимо един таг да бъде затворен, за да се отвори друг и по този начин може да се съчетае тяхното действие.

В този случай е необходимо вътрешния таг да се затвори преди външния, например:

<I>...</I> е правилно!

<I></I> е грешно!

Тук съществува още една особеност – някои тагове имат по-висок приоритет от други. Такива са например <TABLE>, <P>, <DIV>, , , <DL>.

Това означава, че те не бива да попадат вътре в тагове с по-нисък приоритет, но могат да се вграждат свободно едни в други. Например ако искаме да форматираме едно заглавие, ще използваме центриране и удебелен шрифт, както следва:

<DIV ALIGN="CENTER"> заглавие </DIV>. Обратното подреждане на тагове <DIV ALIGN="CENER"> заглавие </DIV> е неправилно.

ВНИМАНИЕ: Съвременните браузъри имат множество вградени алгоритми за корекции на грешки в кода. Ако HTML разрешава подобни „отклонения“, то при XML това е недопустимо.

4.1. Цветове

Може да използвате инструкцията за форматиране <body>, за да зададете цвят на фона или на елементите от текста (вижте табл. 2). Цветът, зададен за връзка, се появява и като цвят на рамката на графиките в документа.

<i>Елемент на страница</i>	<i>HTML инструкция</i>	<i>Описание</i>
Цвят на фона	<BODY BGCOLOR="color">	Задава цвят на фона на цялата страница;
Обикновен текст	<BODY TEXT="color">	Задава цвят на обикновения текст на целия документ. Цветът на текст по подразбиране е черен.
Връзки	<BODY LINK="color">	Задава цвят на хипервръзките. Цветът на хипервръзките по подразбиране е син.
Посетени връзки	<BODY VLINK="color">	Задава цвят на връзките, върху които вече сте щракнали с мишката. Цветът на посетените връзки по подразбиране е лилав.
Действаща връзка	<BODY ALINK="color">	Задава цвят на връзката, върху която щраквате в момента. Цветът на действащата връзка по подразбиране е червен.

Таблица 3. Атрибути за задаване на цветове с инструкцията за форматиране <body>

На почти всички елементи в уеб страниците може да бъде зададен цвят, различен от този по подразбиране. Може да бъде сменен цветът на фона на страницата, текста, връзките, да бъде сложен фонов цвят на таблиците или само на отделни техни полета и т.н. Въпреки, че за всеки отделен случай се използват различни параметри и тагове, съществува единна система за определяне на конкретния цвят или нюанс.

В HTML са предвидени два начина за задаване на цвета – с ключова дума или RGB формат.

На таговете и параметрите в тях, които отговарят з определен цвят могат да се присвояват стойности от типа red, yellow, blue, black, green, white и т. н., които представляват названията на съответните цветове на английски език. Но този начин не е много удачен, защото първо се ограничава броят цветове и второ – не всички браузъри разпознават ключовите думи.

Цвят в RGB формат: RGB е абривиатура от първите букви на трите основни цвята red, green, blue. Това е един от възприетите при компютрите стандарти за цвят, при които по отделно се задава интензитетът на трите основни цвята със стойности за всеки един от тях от 0 /най-слабият/ до 255 /най-силен/.

От тяхното комбиниране могат да се получат $256*256*256=16\ 777\ 216$ различни цвята /нюанси/.

Общия запис се извършва в 16-на Бройна система.

16-на БС={0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F}.

Записът на интензитета на отделния основен цвят се извършва винаги в две позиции и ако числото е едноцифрено, се добавя 0 отпред. Това означава, че стойностите варират от 00 до FF, а за да се отбележи, че записът е в 16- на БС, най-отпред се поставя знакът #. Например: #40FOCC или #PRGGBB. Ето основните RGB кодове:

Черно	#000000		
Бяло	#FFFFFF		
Сиво	#COCOCO	Тъмно сиво	#80808
Синьо	#0000FF	Тъмно синьо	#00008
Червено	FF0000	Тъмно червено	#80000
Зелено	00FF00	Тъмно зелено	#00800
Жълто	FFFF00		
Бибамо	00FFFF		
Синьозелено /циан/			

Таблица 4. Кодове за цветовете

4.2. Общи елементи на страница

Всеки HTML документ започва с тага <HTML>, завършва с тага </HEAD> и е изграден от две основни и принципно различни предназначени части – глава /HEAD/ и тяло /BODY/. В главата се поставят тагове, описващи съдържанието и начините за неговото интерпретиране. Никаква част от нея не се изобразява от брауъра на екрана. Тя започва с тагът <HEAD> и завършва с </HEAD>. Непосредствено след това с тага <BODY> се обявява началото на тялото на документа, в което се съдържа цялата текстова информация, командите за нейното форматиране, както й таговете, с които се включват на съответните места графики и други възможни компоненти от типа на Java Aplet и т.н. Тялото завършва с тага <BODY>.

Като цяло структурата на всеки такъв HTML документ, изглежда по следния начин:

<HTML>

<HEADK>...съдържание...

</HEAD>

<BODY>... съдържание...

</BODY>

</HTML>

4.3. Глава на HTML страница

Заглавната част на документ, оградена от инструкции за форматиране <head>, съдържа информация, която описва HTML документа. Тази инструкция няма свои собствени атрибути; то просто служи като контейнер за други инструкции за форматиране, които спомагат за дефинирането и управлението на съдържанието на документа.

Заглавията: най-често използваният елемент вътре в заглавната част е заглавието на документа (поместено между началната и крайната инструкция за форматиране <title>.

Заглавията трябва да съдържат само ASCII символи (букви, числа и основна пунктуация). Специалните символи в заглавието трябва да се изписват със своите символни обекти, например:

```
<TITLE>The Adventures of Peto & Fleck</TITLE>
```

Заглавията се появяват на потребителския маркер в текста или в „горещите списъци“. Освен това търсачките също разчитат на заглавието на документа. Затова е много важно да поставяте описателни и съдържателни заглавия на вашите страници и да избягвате неясните заглавия от сорта на „Добре дошли“ или „Моята страничка“.

В главата се поставят информацията, която не се изобразява на екрана, но носи информация за тези, които разглеждат кода:

```
<HEAD>
<TITLE>...</TITLE>
<META NAME="Autor" CONTENT="...">
<META NAME="Description" CONTENT="...">
<META NAME="Keywords" CONTENT="..">
<META HTTP-EQUIV="Content-Type"> CONTENT="text/html;
</HEAD>
```

Забележка: На мястото на многоточието стои обикновено тестово съдържание.

4.4. Тяло на HTML документ

В тялото на документа, оградено от инструкцията за форматиране <body>, е поместено съдържанието на документа – частта, която желаете да бъде показана в прозореца на браузъра. Тялото на документа може да се състои само от няколко текстови параграфа, едно единствено изображение или сложна комбинация от текст, изображения, таблица и мултимедийни обекти. Какво ще има на Вашите страници, зависи само от Вас.

В началото на инструкцията за форматиране <body> беше проектирана с цел да определя тялото на документа. Впоследствие нейните възможности се разшириха и сега чрез нея вие можете да контролирате цвета на фона и на текста на вашите документи. Тези настройки са общи – това означава, че те важат за целия документ.

С този таг се определя съдържателната част на информацията. В самия таг <BODY>, могат да се включат параметри, с които да се зададат основните цветове и текста и фона, например:

```
<BODY BGCOLOR="#FFFFFF" TEXT="#000000">
```

```
LINK="#0000FF" VLINK="#0000FF"
```

```
ALINK="#0000FF">, където:
```

- BG COLOR="" е цвят на фона на страницата /по подразбиране/;
- TEXT="" е основния цвят на текста /по подразбиране/;
- LINK="" е цвят на връзките в страницата;
- VLINK="" е цвят на посетените връзки /по подразбиране/;
- ALINK="" е цвят на връзките в момента на натискането им;

В HTML съществува възможност за фон на страницата, да бъде зададено изображение, а не просто цвят. Това става с параметъра BACKGROUND="", където в кавичките се задава името на графичния файл /изображение/:

```
<BODY BACKGROUND="photo.jpg">
```

4.5. Защитаване в HTML

Задаването на вида, цвета и големината на шрифта, с който да бъде изобразен определен пасаж от текста, става с помощта на тага може да има и един от следните три параметъра:

COLOR="" – с помощта на него се задава цветът на текста в RGB формат, например COLOR="#336699".

FACE="" – определя шрифта. Това може да бъде име на шрифт, но ако при потребителя той отсъства, от тагът няма да има смисъл – текста ще бъде изведен на екрана с шрифта по подразбиране, зададен в настройките на браузъра.

FACE="Verdana, Geneva, Arial, helvetica".

SIZE="" – определя големината на шрифта (от 1 до 7); (-2; +4);

Внимание! Шрифтът, валиден за целия текст в web страницата може да се зададе с <BASE FONT>. Това не изключва възможността някои параграфи да бъдат форматирани допълнително, с помощта на .

<BASE FONT> Има същите параметри, както FONT, но не изисква затваряне и се поставя в глава <HEAD>.

Стилът на шрифта се задава с таговете , <I>, <U>, <S>, bold (удебелен), (italic - наклонен), U – underline (подчертан) и strike (задраскан).

4.6. Някои специални символи

Код	На екрана се появява....
<	<
>	>
 	Space /празен интервал/
©	©
®	®
™	™
°	°
±	±
&	&

Таблица 5. Някои специални символи

4.7. Форматиране

Заглавие – за определяне на текста, който представлява заглавие или подзаглавие за HTML са предвидени комплект от тагове <H1>, <H2>, <H3>, <H4>, <H5>, и <H6>. Изискват задължително затваряне. Най-често използвани са <H3> и <H4>. Например: <H3>Заглавие</H3>.

Нов ред – С оглед автоматичното пренасяне на думите на нов ред, когато се запълни текущият, възниква и необходимостта от тага <NOBR>. Използва се за телефонен номер. Изисква се задължително затваряне.

Трябва да се има предвид, че ако ограденият с <NOBR> текст е много дълъг, това ще доведе до излизането на реда от екрана и изобразяването на хоризонтална лента за придвижване по съдържанието (хоризонтален скролбар).

Параграф – Започва с <p> и може да не бъде затварян, но е препоръчително.

Подравняване – в ляво, вдясно, център и т. нар. Justify подравняване от двете страни на текста. За тази цел се използва тагът <DIV> изисква задължително затваряне </DIV>.

<DIV ALIGN="center"> </CENTER>.

Хоризонтален разделител – в HTML е на разположение специалният таг <HR>, използването на който довежда до изобразяването на релефна хоризонтална линия. Този таг не изисква затваряне и може да се използва и с някои от следните параметри:

SIZE – задава дебелина на линия: Например SIZE="1";

WIDTH – ширина на линията; (може по два начина в пиксели – например WIDTH="100") или относително спрямо широчината на екрана – например WIDTH="50%".

ALIGN – задава позиционирането на хоризонталния разделител; (right, left, center, например: ALIGN="LEFT");

NOSHADA – използването на този параметър, без да е необходимо да му се задават (параметри) стойности, означава, че линията ще бъде изобразена без релеф.

 София

Пловдив

<HR SIZE "1"WIDTH="100"ALIGN="left" NOSHADE>.

Неномериран списък – disk – точка, square – квадрати

<UL TYPE="circle"> кръг

Номериран списък -

София

Пловдив

<OL TYPE="A">.

4.8. Таблици в HTML

4.8.1. Тагове в таблица

За създаването на таблици в HTML се използва комбинация от три тага, подреждането на които следва строги изисквания. Ето пример на таблица с три реда и две колони:

```
<table>
<tr>
<td> колона 1, ред 1 </td>
<td> колона 2, ред 1 </td>

</tr>
<tr>
<td> колона 1, ред 2 </td>
<td> колона 2, ред 2 </td>
</tr>

<tr>
<td> колона 1, ред 3 </td>
<td> колона 2, ред 3 </td>
</tr>
</table>
```

Използваните тагове са:

<table> - указва началото на таблицата, следва затварящ таг </table>, указващ край на таблицата;

<tr> - съкратено от table row „ред на таблицата“, указва началото на реда от таблицата; Затваря се задължително с </tr>;

<td> - съкратено от table data „поле с данни“ – указва началото на клетка от таблицата, което е в рамките на дефинирания с помощта на <tr> ред. Следва </td> - край на съответното поле.

4.8.2. Параметри на таблица

width – широчина на таблицата

width="100" - в пиксели или width="80%" в проценти;

height – височина на таблицата в пиксели (height="100");

border – определя дебелината на рамката в точки (border="5");

cellpadding – определя разстоянието в точки (пиксели) от рамката на полето до съдържанието в него, т. е. задава широчината на празното поле – cellpadding="3";

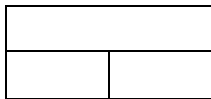
bgcolor - запазва по стандартния начин цвета за фон на таблицата, например bgcolor="#0080FF";

Внимание: отделните полета на таблицата, дефинирани с помощта на <td>, могат да имат цвят различен от общия за таблицата – например <td bgcolor="#00FF00">. На всяко поле може да му бъде зададен размер с width и height.

За позициониране на съдържанието се използват align и valign.

- За align – left, center, right;
- За valign – top, bottom, middle - в средата;

Обединяване на редове и колони:



a/ <table border="1">

<tr>

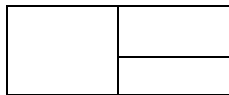
<td colspan="2"> ред 1 </td> </tr>

<tr><td>поле 1, ред 2 </td>

<tr><td>поле 2, ред 2 </td>

</tr>

</table>



б/ <table border="1">

<tr>

<td> колона 1, горно поле </td>

<td colspan="2"> колона 2 </td>

</tr>

<tr>

<td> колона 1, долно поле </td>

</tr></table>.

4.8.3. Създаване на невидима таблица за графика

Това е една по-особена употреба на невидимите таблици.

Използва се за съединяване на разделена на отделни части графика. Това е необходимо най-често случаите, когато се прави меню от графични бутони, разположени върху общ фон. Тяхното разделяне на отделни файлове се извършва с цел за всеки от тях да се зададе самостоятелна връзка. В този случай при изобразяването на екрана не трябва да остават разстояния между отделните полета в таблицата, нито пък да се появява рамка. Това би развалило дизайна и би разместило отделните негови елементи. Следователно основните характеристики на невидимата таблица са следните: няма фонов цвят, няма рамки около полетата и няма разстояние както между отделните полета, така и между границата на полето и съдържанието в него. Казано на езика HTML, това означава, че липсва (не се използва) параметърът BGCOLOR, а присъстват задължително параметрите BORDER, CELLPADDING и CELLSPACING със стойности 0. С употребата на невидимата таблица за графика е възможно да се „разчупи“ правоъгълната структура на всяко едно графично изображение. Методът на работа се състои в това съответната графика (която представлява правоъгълник) да се раздели с помощта на прави разрези на отделни части (отново правоъгълници), всеки от които се подставя в отделно поле на таблицата. Пример за HTML – кода на подобна страница е даден по-долу:

Пример: <HTML>

```

<HEAD><TITLE>...</TITLE></HEAD>
<BODY>
<TABLE BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR>
<TD><IMG SRC="part1.gif"></TD>
<TD><IMG SRC="part2.gif"></TD>
</TR>
<TR>
<TD><IMG SRC="part3.gif"></TD>
<TD>Hello! This is cool!</TD>
</TR>
</TABLE>
</BODY></HTML>

```

„Невидима таблица,, - няма общ фонов цвят и няма рамка около полетата или чрез код – липсва bgcolor, а присъства border="0".

```

<table border="0"> cellpadding="10"
<tr>
<td valign="top"> тук е съдържанието на колона 1</td>
<td valign="top"> тук е съдържанието на колона 2</td>
<td valign="top"> тук е съдържанието на колона 3 </td>
</tr>
</table>

```

Внимание! Нормална големина (ширина) на таблицата:

```

<table width="590" border="0">
<tr>
<td> съдържанието на страницата е тук </td>
</tr>
</table>

```

4.9. Рамки

Употребата на Frames (в превод от английски „рамки“) означава браузърът да отвори няколко HTML страници в рамките на един екран. По този начин изображението се формира на базата на няколко HTML документа, изобразени едновременно в съседни области на екрана. Принципът е следният: указва се в какви пропорции и как да се раздели екранът на браузъра и във всяка една от така получените части – кой HTML документ (уебстраница) да се зареди.

Например ако на всички страници от даден сайт има повтарящ се елемент (лого горе, меню вляво или вдясно, навигация долу) той би могъл да се даде в отделна рамка. За нея се подготвя самостоятелна уебстраница.

Предимства и недостатъци на рамките – Въпросът с употребата на рамки е доста спорен и дискутиран. По-долу са представени няколко аргумента „За“ и „Против“. Положителните страни са няколко. Отделните рамки са напълно самостоятелни – имат своя собствена лента за позициониране и съдържание, определено в отделен HTML – документ. Когато се щракне върху някоя връзка, тя обикновено се отваря в същата рамка, а останалите рамки не се променят. По този начин не е необходимо всяка от страниците да се „грижи“ за цялостното съдържание на екрана. Това води намаляване обема на страниците, респективно на работата по тях, и раздвижва съдържанието. Потребителят остава с впечатление, че „тъче на няколко стана едновременно“. Използването на рамки има и сериозни недостатъци. На първо място това е слабия контрол, който дизайнерът има върху цялостната визия. Често се срещат сайтове, при които, ако се промени размерът на прозореца на браузъра, изведнъж се получават множество ленти за позициониране, чието присъствие не е предвидено. Ако пък тяхната поява е забранена (което може да се направи), части от текста и изобщо от съдържанието се скриват и по никакъв начин не могат да бъдат видени. Друго неудобство представлява фактът, че през цялото време докато потребителят „се разхожда“ из сайт, организиран с рамки, полето на браузъра, показващо URL адреса на страниците, не се променя. То непрекъснато показва именно страницата, създава разделянето на екрана, а не адресите на страниците, които се появяват във вече разделения екран. Това естествено оказва влияние и в случаите, когато файлът се прибавя към списъка с предпочитанията (bookmark). Той в последствие ще води към началната страница, отваряща рамките.

Една от причините да не се използват рамки е, че не всички версии на браузърите и дори не всички браузъри ги поддържат. Това също представлява голям проблем.

4.9.1. Кога се използват рамки?

Използването на рамки с един от най-спорните моменти в уебдизайна. Те имат както своите горещи привърженици, така и своите яростни противници. Често се практикува разработването на две версии и сайта - едната без рамки, а другата - с рамки, като при влизането на потребителя се предоставя възможност сам да направи избор между една от двете. Това означава двойна работа и респективно много по-високи разходи за разработването на сайта.

Много често базираните на HTTP - протокол пощенски услуги предоставят възможност посетителят да избере вариант със или без рамки. В конкретния случай вариантът с рамки е по-удобен поради това, че менюто, което е ключов елемент, е самостоятелно и достъпно. От друга страна, потребителите с по-стари браузъри и тези, които предпочитат да нямат рамки, са улеснени. Така са реализирани безплатните пощенски.

Смята се ,че по-удобно за клиентите да ползват сайтове без рамки. В някой случаи все пак употребата на рамки е за предпочитане, но това трябва да стане след внимателно обмисляне на евентуалните проблеми, които могат да възникнат при използването на сайта(малък екран на брауъра и други).

4.9.2. Как се създават рамки?

Създаването на рамки се извършва с помощта на специални тагове, включени в HTML – документа. На практика те са единственото съдържание на ключовата страница, указваща рамките, като чрез тях се задава кои други страници да се заредят в създадените рамки. Това означава, че те трябва предварително да се подготвят и при създаването на ключовата страница вече да присъстват в директорията, за да могат да бъдат отворени. Таговете, които се използват, са следните:

<RAMESET> - указва начало на дефинирането по разделяне на страницата на две части. Задължително се затваря с </RAMESET>. <FRAMESET> може да раздели екрана само на две, като в зависимост от параметрите това ще стане хоризонтално или вертикално. Ако е необходимо екранът да се раздели на повече от две части, в рамките на единия FRAMESET се прави втори.

Задължително в тага трябва да присъства един от следните два параметъра: COLS="" или ROWS"". Именно те указват дали екранът да бъде разделен съответно вертикално или хоризонтално. Вътре в кавичките стоят две стойности, разделени със запетая, които задават размера на първата и на втората част. Могат да се използват следните стойности: число – означава абсолютен размер, зададен в пиксели; цифра % - означава относителен размер; символът звезда * -означава останалото свободно пространство.

Пример:

```
COLS="100,*"
```

```
COLS="20%,80%"
```

```
COLS="30%,*"
```

```
COLS="*,400"
```

<FRAME> - този таг указва последователно за всяка от двете части какво и как трябва да се зареди. Той се използва задължително със следните два параметъра:

- SRC="" - в кавичките се задава страницата „етикет“, кодово име на съответната рамка;
- NAME="" – в кавичките се задава „етикет“, кодово име на съответната рамка;

<NOFRAMES> - този таг се използва, за да се предвиди съдържание, което да се появи, в случай че брауърът не поддържа рамки.

Пример:

Създаването на рамки става с едновременната употреба на трите тага, и то в точно определен ред, който може да се служи и за шаблони (template):

```
<HTML>
```

```
<HEAD><TITLE>...</TITLE></HEAD>
```

```

<FRAME SRC="page1.htm"NAME="frame1">
<FRAME SRC="page2.htm"NAME="frame2">
</FRAMESET>
<NOFRAMES>
<BODY>...</BODY>
</NOFRAMES></HTML>

```

Чрез така представения пример ще се създадат рамки с параметри по подразбиране, които са следните: дебела рамка, разделяща двете полета, и лента за позициониране (появяват се единствено ако съдържанието излиза от рамките на екрана).

Параметри – за да се премахне напълно рамката, разделяща двете полета, във <FRAMESET> се използват едновременно параметрите BORDER=0 и FRAMEBORDER=0, тъй като единият от тях работи само в Netscape, а другият – в MISE. За всяка отделна рамка може да се укаже статусът на лентата за позициониране, като за целта в съответният таг <FRAME> се включи параметърът SCROLLING="" с една от следните стойности: auto (автоматично), (Y) да или (No) не.

Пример:<.FRAMESET SRC="page1.htm"NAME="frame1"SCROLLING="no">

4.10. Разширени възможности на HTML

С течение на времето и с развитието на технологиите възможностите на уебстраниците се обогатиха и разшириха многократно. Ако се върнем във времето до 1991 г., когато е изобретен WWW, ще видим, че тогава първите уебстраници са представлявали просто много текст и малко връзки, без графика, без движение и т. н. С това естествено ни най-малко не подценяваме факта, че те са били революционна крачка напред в развитието на Интернет. Днес уебстраниците са интерактивни, почти изцяло графични, обогатени със звук, мултимедийни елементи, формуляри за попълване, възможности за електронна търговия и какво ли още не. Всичко това е резултат на неуморната работа и богатото въображение на милиони специалисти по целия свят.

Тук са представени накратко елементите, които оформят днешния облик на Интернет, но представляват второ ниво на познанията по уебдизайн. Разясненията ще бъдат предимно в областта на общата култура, необходима на всеки уебдизайнер. За доброто овладяване на тези възможности е необходимо преминаването на специализиран курс, а за пълното им разбиране – развиване на програмистки умения.

4.11. Връзки в HTML

Създават се с помощта на тагът <A> и .

Всичко, включено в HTML кода между таговете <A> и , представлява хипервръзка, независимо дали съдържа текст или графика. На къде да води тя се задава с параметъра href="" , където в кавичките стои адресът в Интернет (URL).

```

<A href="http://www.abcdg.com/"> </A>
<A href="http://wwwabcdg.com/index.asp">abc design </A>

```

Какво е URL? –

Всеки един обект в мрежата има свой уникален адрес: Uniform Resource Locator, чрез който може да бъде извикан и разгледан с помощта на браузър. Най-отпред се поставя името на протокола за пренос на данни по мрежата, който за web страницата е http (hyper text transfer protocol). Следва името на регистрирания домейн, директорията, поддиректориите и името на html документа (файл) разделени с наклонена черта (“/”).

Ако се свързва файл от същата директория, се пише само неговото име:

Go to this page ;

- Ако се свързва файл от поддиректория, се пише нейното име, наклонена черта (/) и името на файла:

< Go to This page

- Ако се свързва файл от по-горна директория, се пише “/” и името на файла: <A href "../page.htm">Go to This page .

Тим Бърнъс-Лий е казал, че ако е знаел предварително каква популярност ще придобие Web, би помислил по-сериозно за намирането на алтернатива на двете наклонени черти в URL адреса, които са особено досадни при диктуването им по телефона. По принцип, URL е код за четене от машините и не би трябвало да има нищо общо с дизайна на потребителския интерфейс. В действителност обаче е тъжна истина това, че URL адресите се разкриват пред потребителите в много от случаите на ползване на Web, затова трябва да ги обмислим като аспект на дизайна. Най-добре е сливане на имената на сайтовете – домейните, колкото по-кратко, толкова по-запомнящо се е. Например Jacob-nielsen.com може да се изпише jnielsen.com. За разделители може тирета, но не и долни тирета. Ако потребителите са в състояние лесно да запомнят името на домейна Ви, то те могат поне да стигнат до заглавната страница, от където чрез търсене и навигация спокойно ще намерят необходимата им страница, дори да не знаят останалата част от URL адреса. Притежаването на неясен домейн струва доста загуба на клиенти или посетители. В Web хората са изключително целеустремени. Те искат да направят нещо конкретно и не са толерантни към нещо, което се изправя между тях и целта им. Затова водещият принцип на web-дизайна трябва да бъде разчистване на пътя към успеха на посетителите, колко е възможно най-бързо.

Връзка за изпращане на e-mail: contacts .

Примери за свързване на текст:

Към локален файл: ...

Към външен файл: ...

Към поименен репер: ...

Към поименен репер в текущия файл: ...

За изпращане на съобщение по електронна поща: ...

Към файл, намиращ се на FTP сървър:

...

4.12. Графични изображения и връзки

Ако е необходимо едно изображение в уеб страницата да бъде и хипервръзка, т.е. цялата негова повърхност да препраща на определен адрес в Интернет: достатъчно е да се използва тага <A>:

```
<a href="http://www.yahoo.com">
```

```
</A>
```

4.13. Фреймове

Технология, спомагаща за по-голямо удобство при използване на информацията и изграждане на навигацията на сайта.

<frameset> - указва начало на разделяне, задължително се затваря с </frameset>.

Разделянето на екрана може да бъде само в едно направление – хоризонтално или вертикално. Ако е необходимо да се постигне по-сложна комбинация от фреймове, то тогава в рамката единия frameset се поставя в другия. В тага трябва да присъства само един от следните два параметъра cols=" " или

rows=" ". Именно те указват дали екрана да бъде съответно вертикално или хоризонтално. В кавичките могат да стоят някакви стойности, разделени със запетайка, които задават размера на съответни части, например:

```
cols="100,200","*" или cols="20%,80%"  
cols="30%,*"; cols="*,400".
```

Ако са посочени две стойности, то ще се получат две части със съответните размери, ако са три – три части и т. н. Символът „*“, означава каквото остане и е задължително да се използва, когато другите размери са в пиксели, тъй като не може да се предвиди точно колко е голям екранът на брауъра при даден потребител, а дори създаването на фреймове не може да промени размера му.

<frame> - с този таг се указва последователно за всяка от частите какво и как трябва да се зареди. Той се използва задължително с два параметъра: src=" " – в кавичките се задава страницата, която трябва да се зареди;

name=" " – в кавичките се задава „етикет“, кодово име на съответната рамка;

<noframes> - използва се за да се предвиди съдържанието, което да се появи в случай, че брауърът не поддържа фреймове.

Създаването на фреймове става с едновременната употреба на трите тага и то в точно определен ред:

```
<html>
```

```
<head>...</head>
```

```
<frameset cols="40%,60%">
```

```
<frame src="page1.htm name="frame1">
```

```
<frame src="page2.htm" name="frame2">
```

```
</frameset>
```


<noframes>

<body>...</body>

</noframes></html>

За да се премахне напълно рамката, разделяща двете полета във <frameset>, е необходимо да се добавят параметрите border="0" и frame border="0".

4.14. Форми в HTML

Отварящ таг <FORM> и следователно затварящ таг </FORM>.

1. Падащо меню: <select><option>1<option>2</select>;
2. Списък: <select size="5"><option>1<option>2</select>;
3. Поле за въвеждане на текст: <textarea rows="5" cols="23"></textarea>;
4. Ред за въвеждане на текст: <input type="text">;
5. Ред за въвеждане на парола: <input type="password">;
6. Ред за въвеждане на файл: <input type="file">;
7. Бутон: <input type="button">, <input type="submit">, <input type="reset">;
8. Чек бокс: <input type="checkbox">;
9. Радио бутон: <input type="radio">;
10. Изображение: <input type="image"="photo.jpg">;

Скрит обект на данни: <input type="hidden">. На всички тези обекти обикновено се задават имена с параметър name в съответния таг, като позволява въведената информация, да е достъпна за ползване и в последствие.

4.15. Стиллове в HTML

HTML е гръбнака, върху който се изгражда цялото съдържание в мрежата. Затова е необходимо HTML кодът да бъде „украсен“, допълнен с технологии и езици, които могат да се интегрират с него, да го използват като основа и да внесат в websites допълнително функционалност и възможности.

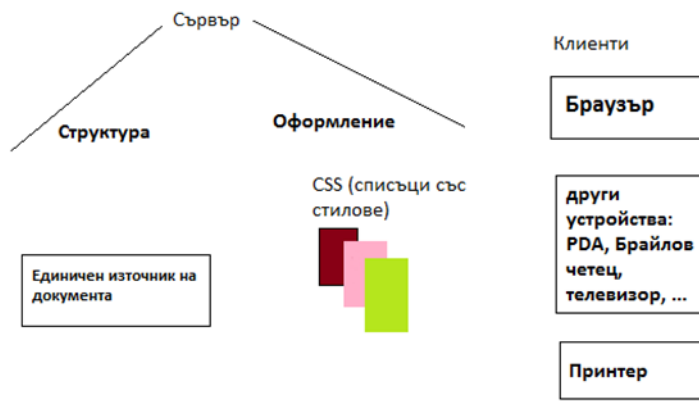
CSS – технологията: контролираме параметрите за визуализиране на всеки един HTML таг. Може да се направи примерно тагът да има действието на тагът <I>, но много по-полезно е например с нейна помощ да се премахне подчертаването на хипервръзките. Не е много естетичен подчертания текст.

Добавянето на CSS технологията към HTML се основава на тага <STYLE>, който се поставя в главата на уеб документа – страницата.

```
<style type="text/css"><! - -  
A {text-decoration:underline;}  
A :hover{text-decoration:underline;}  
B {background:#000080; color:#FFFFFF;}  
- -></style>.
```

За да помогнат на уебдизайнерите да преодолеят ограниченията в HTML по отношение на оформлението, създателите на браузри въвеждат стилните елементи от типа на . Свикнали с днешните мощни текстообработващи програми, дизайнерите и авторите на съдържание искат да разполагат със същите възможности за манипулация и стриктно позициониране на обектите по уебстраницата, каквито имат и при дизайна на една печатна страница. Отново HTML не е предвиден за това. Също както SGML,

задачата на HTML е да опише структурата на документа, а не неговия стил на представяне. При смесване на информацията за стиловете в самата структура на документа, както е било в ранни периоди на уебразработките, се ограничава междуплатформената преносимост на съдържанието. Вградените в уебстраниците инструкции иза визуализиране са специално съобразени с конкретен тип изходно устройство – компютърния екран. Чрез обособяване на списъци със стилове спецификите на оформлението се отделят от самото съдържание. По този начин се обхваща огромното разнообразие от устройства и потребители, които влизат в Уеб. Уебсървърът може да определи типа на устройството, което се обръща към него, и да предостави съответстващия му списък със стилове. Фигура --- илюстрира тази концепция:



Фиг. 16 различни крайни устройства

Препоръки за W3C за стиливия уебезик, наречен CSS (Cascading Style Sheets), позволяват на авторите на уебстраници да създават стилови правила за различните елементи и да ги записват в отделен документ, наречен style (списък със стилове). CSS премахва необходимостта да се ползват нестандартни начини за писане на код, които до скоро доминираха в уебдизайна. Например да приемем, че искате всички заглавия в уебсайта от типа `<h1>` да бъдат центрирани и изобразени в зелено. Без CSS ще трябва навсякъде, където се среща елементът `<h1>`, да добавите следния код: `<h1 align="center">Някакво заглавие</h1>` Използвайки CSS, можете да постигнете същото оформление така: `h1 {color: green; text-align: center;}`

Можете да сложите това правило във външен списък със стилове и да свържете всички страници от сайта си към него. Така едно-единствено правило контролира всички елементи `<h1>` в целия уебсайт. По-късно, ако решите да смените цвета на `<h1>` от зелен на червен, е достатъчно просто да промените правилото, за да се отрази това във всички уеб страници на сайта. Доскоро приемането на CSS като стандарт беше ограничено, най-вече поради слабата му и ненадеждна поддръжка от страна на основните браузъри. Поновите браузъри, като Internet Explorer 7.0, FireFox 2.0 и Netscape Navigator

8.0, предлагат по-пълна и последователна поддръжка с CSS, което дава свобода на дизайнерите да използват този мощен стилев език.

4.16. ИЗБОР НА HTML РЕДАКТОР

Можете да създавате или да генерирате HTML код за Уеб по много начини. В първите дни на Уеб най-широко използваният инструмент беше Notepad – прост текстов редактор, включен във всички версии на Windows 95 до Vista. Неговия еквивалент за Macintosh е TeachText или SimpleText. Много сайтове в Уеб са създадени с тези лесни за използване текстови редактори, но които някои от разработчиците на HTML код все още разчитат. Те са и най-добрия начин да се научи HTML, защото всеки отделен таг трябва да се опише на ръка. Все пак днес малцина дизайнери ползват прост текстов редактор, когато на пазара се предлагат все по-мощни пакети за създаване на HTML.

Съществуват множество HTML редактори, например AdobeDreawever, Adobe GoLive и Microsoft Expression Web (който замести FrontPage). Ориентираните към чист код HTML редактори, като Adobe HomeSite, се отказват от визуалния подход. Те станаха популярни, защото предлагат много мощни подобрения, липсващи в Notepad, като например разнообразни възможности за търсене и заменяне, както и синтактична проверка. Същевременно тези редактори запазват и възможността за манипулация на кода на ниво таг. Най-новите инструменти предлагат също и валидиране и преобразуване на кода, което значително редуцира задачите около почистването на стария HTML, така че да съответства на новите синтактични правила и стандарти на XHTML.

Бележка:

<p>В Уеб се предлагат на изгодна цена множество HTML редактори, лицензирани като софтуер с гратисен период (който може да се изпробва преди да бъде закупен). Посетете www.download.com, потърсете за “HTML editors” (HTML редактори) и ще видите списък със софтуер, който може да изтеглите.</p>
--

Много нови офис пакети също преобразуват документи в HTML. Например можете да създадете брошура с текстообработваща програма и да я експортирате в HTML страница. Може дори да създадете презентация на Microsoft PowerPoint и да я експортирате в HTML. За един разработчик на HTML, подобна автоматизация не е приемлива, защото така се губи контрол върху генерирания код. Нещо повече, практиката да се преобразува съдържанието на даден документ от приложение като Microsoft Word в HTML е известна с изобилието от нестандартен HTML код. Ако възнамерявате сериозно да създавате привлекателни и спазващи стандартите уебсайтове, най-добре е да се ориентирате към специализираните пакети за създаване на HTML, а не към офис приложения.

Също като браузрите, HTML редакторите интерпретират таговете съобразно собствената си вградена логика. Затова страница, която сте създали с редакторски интерфейс, може да изглежда доста различно, заредена в браузър. Нещо повече, много редакторски пакети създават сложен и нестандартен код, за да се постигне даден ефект, указан от потребителя. Този сложен код може да предизвика проблеми със съвместимостта на различните браузри. Свикналите да пишат HTML код на ръка (с Notepad или Notepad++, ...), често се изненадват, когато видят колко сложен е кодът, генериран от редакторски пакет за HTML.

За да бъдете ефективни, трябва да умеете свободно да работите на ниво HTML код.

Дори и да предпочете да ползвате от многото редакторски анкети за разполагане на основните елементи, генериране структурата на страницата или за изграждане на

сложна таблица, имайте готовност да редактирате на ниво код при нуждата от отстраняване на проблеми. В крайна сметка, докато стигнете до крайни вариант на страницата си, най-вероятно ще сте работили с комбинация от няколко различни софтуерни инструмента, за да придадете **интерактивност на Вашия уеб интерфейс**.

Създаване на качествен код

В миналото уебдизайнерите се базираха на принципа „**важното е да работи**“ и пишеха нестандартен код, с който да накарат брауъра да изобрази желанието от тях резултат. Илюстрация за това е използването на заглавните тагове от <h1> до <h6> изцяло заради размера на буквите, а не според логическото им предназначение в даден документ. Друг пример е използването на <table> като инструмент за позициониране на обекти вместо като контейнер за данни, каквато е действителното му предназначение.

Подобни лоши навици, за щастие, вече са в миналото. Необходимо е да създавате код, който да е максимално съобразен със стандартите и да представя съдържанието на уеб сайта по най-добрия и **интерактивен уеб подход**.

- Придържане към стандартите
- Използване на семантично маркиране
- Избор на правилния тип документ
- Валидиране на кода

Чрез използването на предвидени команди в HTML, като CSS, Java Script, Flash аплети, видео стрийминг и т. н., което образно казано са горните етажи на къща, в основата на която си остават сравнително простите и „консервативни“ възможности на HTML. Брауърите четат и визуализират именно този код!

За всички, които не са доволни от ограничения контрол над презентацията на уеб документите, който ви предлага стандартния HTML, Cascading Style Sheets (CSS) са едно чудесно разрешение. Освен това те са официалният стандарт на W3C за контролиране на цялата мултимедийна уеб динамична презентация, като оставят на HTML първоначално отредената му роля – да указва структурата. Ето някои **важни аспекти** в употребата на листовите със стилове:

Подобно на своите двойници в програмите за оформление на страници за настолна издателска дейност, листовите със стилове в HTML дават възможност на авторите да прилагат типографски стилове и инструкции за празни пространства към елементите на дадена страница. Думата Cascading (каскаден) показва какво се случва, когато няколко източника на стилистична информация се надпреварват за контрола над елементите на дадена страница – стилистичната информация се подава от листовите със стилове на по-високо ниво (и от родителските към дъщерните елементи вътре в документа), докато приоритетът не бъде предаден на друга стилистична команда.

Листовите със стилове се оказаха добра новина както за дизайнерите, които искат да имат повече контрол над презентацията, така и за HTML пуристите, които се придържат към принципа, че стилът трябва да е отделен от съдържането и структурата на HTML документа. По този начин листовите със стилове превърнаха в реалност мечтите и на двете групи. Важно обаче е, да знаем всички преимущества и недостатъци.

Консорциумът W3C дефинира CSS като „прост механизъм за добавяне на стил (например шрифтове, цветове, разстояние между буквите/редовете) към уеб документите“ (www.w3.org/style/css/). Струва си да се споменат някои подробности, които са пропуснати в това обобщение:

- CSS е стандартен език за разположения за виртуалното пространство – такъв, който контролира цветовете, типографията и размера и разположението на елементите и изображенията.
- Въпреки че е точен и мощен, CSS код може лесно да се пише на ръка, както ще бъде показано в тази глава.
- CSS приятелски настроен към честотната лента; един – единствен CSS файл може да контролира външния вид на целия сайт, който се състои от хиляди страници и стотици мегабайти.
- Създателите (W3C) на CSS отдавна целят той да замести HTML разположенията, базирани на таблици, и друг презентационен код, но CSS може да бъде много ефективен в хибридни, преходни разположения.
- Чисто CSS разположение, комбинирано със структурен XHTML, може да помогне на дизайнерите да разделят представянето от структурата, правейки сайтовете по-достъпни и по-лесни за поддръжка.

ИЗВОД:

Преимущества

Листовете със стилове предлагат на уеб дизайнерите следните преимущества:

- По-добър контрол върху типографията и оформлението на страниците.
Със листовете със стилове Вие можете да задавате основните типографски атрибути като размер на шрифта, разстояние между редовете и разстояние между буквите. Също така листовете със стилове предлагат методи за задаване на отстъпки, полета и позициониране на елементи и дори използват терминологии от традиционната и настолната издателска дейност като точки и пространства, изчислени в em – мерна единица за дължина.
- Стильът е разделен от структурата.
HTML е създаден с цел да обозначава структурата на уеб документа, към който се прилага презентацията от устройството на потребителския край. Обаче през последните години предназначението на HTML беше разширено с по-голям контрол върху презентацията (инструкцията за форматиране е възможно най-скандалният пример). Листовете със стилове, когато бъдат направени правилно, връщат първоначалния замисъл на HTML и ги поставят в отделен, избран от вас участък.
- Потенциално по-малки документи.
Задаването на шрифтови спецификации в началото на документа вместо многократното използване на може драстично да намали броя на символите в документа, а по този начин и неговия размер. А както знаете, в Мрежата малките файлови размери са за предпочитане (както и по-краткото време за сваляне на данните от сървъра).
- По-лесна поддръжка на сайта. Възможно е да свържете няколко уеб страници към един лист със стилове, което означава, че можете не само да правите по една промяна, влияеща върху определен елемент от дадена уеб страница, а да прилагате стилови промени на стотици и дори хиляди уеб страници, като редактирате един единствен файл.
- Можете лесно да го научите. След като сте разучили синтактичните правила, създаването на основни листовете със стилове е не по-сложно от оставянето на инструкции за форматиране в документ. За да се улесни още повече вашата

работа, сега функциите със стилове са вградени в уеб обработващите инструменти като Macromedia Dreamweaver, Adobe Golive...

5. ТЕХНОЛОГИЯТА CSS. СТИЛОВИ ПРАВИЛА В CSS

Стиловите правила в CSS определят стиловите характеристики на даден елемент на HTML. Когато се групират тези правила образуват списък със стилове (style sheet). Стиловите правила са лесни за писане и разбиране. Кодът по-долу показва просто стилово правило за елемента <p>. Стиловите правила се съдържат в елемента <style> в разряда <head> на документа. Това правило определя, че текстът на всички елементи – параграфи в документа ще бъдат сини и с размер 24:

```
<head>
<style type="text/css">
P {color: blue; font-size: 24 pt}
</style></head>
```

Стиловото правило се състои от две части: селектор и декларация: съдържа информация за стила на даден елемент. Селекторът посочва елемента, за който се отнася правилото. CSS предлага множество мощни техники за селекция. Декларацията съдържа точната стойност на свойството. Декларацията съдържа свойство и стойност. Стойността е състоянието на свойството, например синьо за цвят, 125% за размер на шрифта или 30 px за бяло поле. След стойността се поставя точка и запетая (;). CSS съдържа богат набор от свойства, всяко от които може да приеме определени стойности. Общо взето селекторите и декларациите могат да се комбинират по различни начини. Може да се комбинира CSS и HTML по **три начина**: чрез атрибута style, чрез елемента <style>, чрез външен списък със стилове.

Атрибутът style: Задаван стил на даден елемент - <h1 style="color": blue>Some text </h1>. Атрибутът style се използва, когато искаме за определен стил да го заменим с дефиниран стил от по-високо ниво за конкретен елемент, т. е. ако трябва цветът на някое заглавие да бъде различен от цвета на всички останали заглавия в документа. Атрибутът е полезен за експерименти при разработки на документи.

Елемент style: <style> винаги се съдържа в раздела <head> на документа. Правилата, които се съдържат в елемента <style>, влияят само върху документа, в който се намират. <head>

```
<title>примерен документ</title>
<style type="text/css">
H1 {color: red;}
</style>
```

</head>. Стойността "text/css", показва, че използваният стил е CSS. Дори да не е задължителен, type е добре да се включва винаги към елемента <style>, за да бъде съвместим.

Външен списък със стилове: Декларирането на списъци със стилове във външен документ дава възможност да се задават правила за много уебстраници наведнъж. Това е лесен начин за употреба на стиловете, защото създава условия за контрол на целия сайт от един единствен файл със стилове. Външния списък със стилове е обикновен текстов документ, който съдържа стилови правила. Разширението на файла е .css.

```
H1 {color: white; background-color: green;}
```

```
H2 {color: red;}
```

Този файл не е HTML документ и не съдържа никакъв код на HTML, а само стилови правила на css. Тук не е необходимо да включвате елемента <style>.

```
<head>
```

```
<title> Sample Document </title><link href="style.css" rel="stylesheet" type="text/css">
```

```
</head>
```

Атрибутът rel посочва взаимоотношението между текущия и свързания документ. Браузърът показва страницата на базата на информацията, подадена от CSS. Предимството е, че можете да опишем стиловите правила в един документ и те ще се приложат едновременно на всички страници и промяната е лесна – само от един файл. В CSS може да се добавят коментари: Започват с „/*“ и зъвършват с „*/“.

Каскадност на стиловете: - Множество стилови правила и списъци със стилове могат да се приложат към един и същ документ. Авторът на HTML код може да прикачи определена група от стилове, докато потребителят може да предпочете собствените си стилови настройки поради лични или технически ограничения. Само един стил може да бъде приложен върху даден елемент. Каскадността определя кои стилови правила да бъдат приложени върху елементите на документа, като им придава определен приоритет:

- *използване на ключова дума important,
- * произход на правилото,
- * обхват на селектора,
- *място на правилото в списъка със стилове.

При каскадното наслагване на правилата, които идват от автора на документа, са с най-голям приоритет: това е външния вид на страницата, който повечето потребители биха желали да видят, а именно предвидения от автора дизайн. Списъкът със стилове е на второ място. Заложените в брауъра стилове са с най-малък приоритет: това е списъкът със стилове, който определя начина на показване на документа по подразбиране. Например заглавието <h1> се показва с шрифт Times Bold и с нов ред преди и след него. Заложените в брауъра стилове определят как ще се покажат елементите, към които няма прикачени други стилове.

Наследяване: елементите в един документ на HTML са организирани в йерархична структура. Най-висшия елемент – родителя е <html>. Родителските елементи съдържат в себе си други елементи, наречени деца. <head> и <body> са непосредствени деца на <html>. Същевременно са и родителски елементи, защото съдържат в себе си други вложени елементи. Ако проследите надолу йерархията на документа, ще откриете още елементи, които са едновременно родители и деца, като например <p> и . По

подразбиране свойствата в CSS се наследяват от родителските елементи към децата. Възползвайки се от наследствеността, можем със съвсем малко правила да приложим стил на множество елементи в документа.

```
<style type="text/css">
```

```
h1 {color: red;}
```

```
p {color: red;}
```

```
ul {color: red;}
```

```
em {color: red;}
```

```
li {color: red;}
```

```
</style>
```

Правилото работи, защото всички елементи са деца на `<body>` и всички правила са еднакви. Далеч по-ефективно е да се напише едно правило за родителския елемент, което децата да наследят. `<body>` е родителския елемент на съдържателната част от всеки документ на HTML. Това е селекторът, който ще приложи общ стил на целия документ.

Техники за селекция – основните. Селектори на тип: селекторът посочва елемента, за който се отнася декларацията на стила – `body {color: grey;}`

```
h2 {color: red;}
```

```
p {font-size: 85%;}
```

- **групирани селектори:** За да бъдат стиловете Ви по-стегнати, може да групирате селекторите, към които прилагате едни и същи правила. Например следната група стилове има еднаква декларация на различни елементи – задава червен цвят на елементите `<h1>` и `<h2>`.

```
<h1> {color: red;} <h2>{color: red;}
```

Двата селектора могат да се запишат лесно: като се разделят отделните селектори със запетая: `h1, h2 {color: red;}`.

- **Комбинирани декларации:** Често в един и същ селектор се прилагат декларации за различни свойства.

```
P {color: blue;}
```

```
P {font-size: 125%;}
```

Следните стилови правила показват, че текстът на елемента в параграфа трябва да бъде син и с размер 125 % от размера п подразбиране. Двете декларации може да се запишат по-просто, чрез комбинацията им в едно правило. Декларациите се разделят с “;”. `p {color: blue; font-size:125%;}`

- **йерархични селектори-контекстен селектор:** базира се на йерархичната структурирана елементите в дървото на документа. Позволява да посочваме елементи, които се намират в други елементи. Следното правило селектира само елементи ``, които са части на `<p>`. Всички останали елементи в `` в документа остават непроменени. `p b {color: blue;}`. Селекторът съдържа няколко елемента, разделени само с интервал. Ако искаме да е по-конкретен, може да използваме повече от два елемента.

Уебтипография: До скоро уебтипография означаваше употреба на безброй тагове и текстови изображения, но CSS днес предлага ефикасен стилев език, с който да манипулираме множество свойства на текста и да постигнем професионален резултат, без да се налага да прибъгваме до изображения, които увеличават времето за зареждане на уебстраницата ни. Оформлението на текста в уеб със сигурност поражда емоция у потребителите. Фактът, че има разнообразие от шрифтове не означава да ползвате всички. Ако изберем шрифт, който не е наличен, браузърът ще го замени със стандартен, ползван по подразбиране. Трябва:

- да избираме ограничен брой шрифтове и размери
- да използваме общодостъпни шрифтове
- да направим текста лесен за четене
- да избъгваме употреба на текстови изображения, които изискват много време за зареждане

Страниците ще са по-изчистени, ако използваме малко шрифтове и размери. Изберете шрифт за всяко ниво от йерархията на документа – за заглавие, за вътрешно заглавие и за основен текст. Представете различната структура на документа чрез промяна в размера, дебелината и цвета на буквите. Например заглавието на страницата би трябвало да ползва по-голям и дебел шрифт, докато вътрешните заглавия могат да ползват същото начертание, но с по-малки и тънки букви. Изберете няколко размера и дебелина от едно и също семейство шрифтове: три степени, голям за заглавията, по-малък за подзаглавията и стандартен за основния текст. Не правете произволни промени в избора си на шрифт. Прилагайте едно и също оформление на текста и комбинация на стилове в целия уебсайт. Последователността в дизайна създава визуална идентичност на сайта.

Стандартни шрифтове: шрифтовете са слаба страна на HTML, защото информацията за тях се съдържа при потребителя. Браузърът и ОС определят как и дали ще се избрани даден шрифт, ако използвате в страницата си шрифт, който потребителят не е инсталирал, браузърът ще го замени със стандартния Times за Macintosh или Times New Roman за Windows. Проблем е, че масовите шрифтове изглеждат с различен размер през различните ОС. Може единствено да разгледаме как изглежда сайтът ни през различни платформи. За това ползвайте цели семейства шрифтове, за предпочитане безсерифни: Arial, Helvetica, Times New Roman.

Избъгвайте текстови изображения, макар че на моменти се оказват доста полезни: ако трябва да ползвате специален шрифт, с който подозирате, че потребителят няма да има инсталиран на компютъра си, може да създадем текстово изображение с Adobe Photoshop, ..., съхранявайте текста в GIF, JPG формат. Така може да добавите сянка или други ефекти към текста – само за важни места в уеб като лого или повтаряща се графична навигация. Но имайте предвид, че те от тези текстови изображения не могат да сърфират, за това използвайте оформление чрез HTML. CSS разполага с много мерни единици, вкл. абсолютни (пунктове), относителни (пиксели) и процент от основния размер на шрифта. Изборът на мерни единици зависи от това къде ще се изобразява текста. За печатна медия – абсолютни единици, например пункт или сантиметър. Като създавате дизайн на уеб – относителни стойности, които се адаптират според екрана на потребителя – em, px.

Таблица 6. Мерни единици в CSS

единици	съкращение	Описание
Абсолютни		
Сантиметър	cm	Стандертен метричен сантиметър
Инч	in	Стандертен американски инч
Милиметър	mm	Стандартен метричен милиметър
Пика pc	pc	Типографска единица, равна на 12 пункта
Пункт	pt	Типографска единица, 72 пункта правят един инч
Относителни		
Ем	em	Ширина на главна буква М. Обикновено съответства на размера на конкретния шрифт
Екс	ex	Височина на буквата x в конкретния шрифт
Пиксел	px	Размер на пиксел за конкретния монитор
Процент	150%	Размер според базовия шрифт. 150 % е 1,5 пъти размерът на базовия шрифт

Общо свойство за шрифт:

Стойност: `{{<font-style> || <font-variant> || <font-weight> }?`

`<font-size>[/<line-height>?<font-family>]`

База: виж отделни свойства

Отнася се за: всички елменти

Наследено: да

Проценти: позволени само при `<font-size>` и `<line-height>`

Свойството font позволява да зададем стойности за `font-style`, `font-variant`, `font-size`, `line-height`, `font-family` в една и съща декларация. Само две от тях са задължителни – за размера и семейство на шрфита, които трябва да са посочени в правилния ред: `p {font: 12pt arial;} h1 {font: 2em sans-serif;}`

Разстояния в текста в CSS: чрез свойствата на текста в CSS се променя разсткоянието около и мемжду буквите:

Отстъп на абзац – свойството `text-indent`

p {text-indent: 2em;}

blockquote {text-indent: - 2em;}

Свойство text-align – подравняване: служи за избор на хоризонтално подравняване на текста в даден елемент – left, right, justify, center. P {text-align: justify;}.

Свойство Декорация / text-decoration/ - подчертаване на текст, ефект, който има особено значение за хипертекстови документи. Това свойство предлага стойност blink, което прави текста да мига. Ако искаме да махнем подчертаването от елементите <a>, то:

a {text-decoration: none;}.

Свойство височина на ред /line-height/ - в CSS може да посочите или точна стойност, или процент за височина на ред – междуредие /разредка, бяло поле между редовете текст/. Процентът зависи от размера на шрифт. Ако посочим 150%, при размер на шрифта 12pt, височината на реда ще бъде 18 pt. P {line-height:150%;}. Това свойство може да направи текста по-лесен за четене.

Вертикално подравняване /vertical-align/: може да ползваме вертикално подравняване за разполагане на текст спрямо изображение. Следното правило добавено към елемента с атрибут style, задава стойност text-top на вертикално подравняване.

Разстояние между буквите / свойство letter-spacing/ - използва се за промяна на разстоянието между буквите. В печатната терминология се нарича кернинг. h1 {letter-spacing: 4pt;}

Разстояние между думите /свойството word-spacing/ - за промяна на разстояние между думите в текста. Разстоянието, което зададете в стилово правило, се добавя към стандартно зададената стойност. Следния код задава word-spacing fj 2 em.

{word-spacing: 2em;}

5.1. Графика и цвят при уебдизайна. Основи на компютърните цветове. Контрол на изображенията и цвета чрез CSS. Контрол на фонове изображения чрез CSS.

В една уебстраница може да се комбинират текст, цвят и изображения, затова едно от причините Интернет да бъде толкова атрактивен и популярен. Ако използваме подходящ начин комбинация от тези елементи, може да създаваме приятен и атрактивен сайт. Но ако включим твърде много големи и сложни изображения, не подберем подходящи цветове и фон, ще отблъснем посетителите на сайта си, поради дълго зареждане на страниците, нечетим текст и объркваща навигация. Трябва да се търси баланс между изображения и текст. Това се постига с CSS контрол на разстоянията и подравняването. Чрез фонове изображения в CSS може да подбирате външния вид на сайта и да придавате фирмена идентичност и марка. Нужно е да се ползват цветове разумно, така че да комуникирате и да водите читателите към определени секции от сайта. Проверявайте дали цвета е добър за всички браузъри, тествайте при различни скорости на връзката, за да се уверите, че това няма да оттегли времево посетителите на сайта Ви.

Основи на компютърните цветове: Преди да създаваме или прикачваме изображения, трябва да се знае как компютърния монитор работи с цветовете. Мониторът показва цвят, като смесва трите основни : червен, зелен и син, които често ползват съкращението RGB. Всеки от тези три цвята се нарича цветен канал. Мониторът може да излъчи всеки канал с различен интензитет от 0 до 255 наситеност на цвета. Цветовете зависят както от настройките на потребителя, така и от производителя и могат да варират значително на различни монитори. Количеството данни, които се използват за постигане на определен цвят, се нарича **дълбочина на цвета**. Ако мониторът може да изобрази 8 бита данни за всеки от трите цветни канала, той има 24 – битова дълбочина на цвета / $8 \times 3 = 24$ /. 24 - битовите изображения могат да съдържат близо 17 млн. цвята и се наричат true-color. Двата формата поддържат 24 - битов цвят. Ако потребителите имат 24 - битови цветове, ще видят пълната дълбочина на цвета на вашите изображения.

Дифузия – като заредите 24 - битово изображение на 8 - битов монитор или ползвате файлов формат, който не поддържа 24 - битов цвят, браузърът трябва да използва смесица от свои цветове. Тъй като 8 - битовите монитори работят с по-малки цветове /256/, браузърът опитва да постигне някакво приближение до липсващите чрез смесване на цветовете, с които разполага. Това смесване се нарича дифузия. Прилага се като браузърът попадне на цвят, който не поддържа, например, когато се опитвате да превърнете 24 - битова фотография в 8 - битово изображение с 256 цвята. Дифузните изображения често изглеждат зърнести и пикселизирани. Една от възможностите за контрол на дифузията е да използвате недифузни цветове. Недифузната палитра се отнася само за GIF и 8 - битовия PNG формат, но не и за 24 - битов JPG. Поради променливата същност на цвета в Интернет винаги тествайте цветовете, които сте избрали, и бъдете предпазливи при добавянето им към дизайна на сайта. Цветовете изглеждат различно на всяка марка монитор и ОС. Лошият избор на цвят дразни и разсейва посетителя от съдържанието. Тъмен фон, крещящи цветове и нечетливи хипервръзки са само част от множеството примери в Интернет за неразумна употреба на атрибут `color` в HTML.

Таблица 7 . Имена на цветове, познати за повечето браузъри (цветовете с получен шрифт са част от уебпалитрата)

Име на цвят	Шестнадесетична стойност (Hex)	Име на цвят	Шестнадесетична стойност (Hex)
Aqua	00FFFF	Navy	000080
Black	000000	Olive	808000
Blue	0000FF	Purple	800080
Fuchsia	FF00FF	Red	FF0000
Grey	808080	Silver	C0C0C0
Green	008000	Teal	008080
Lime	00FF00	White	FFFFFF
Maroon	800000	Yellow	FFFF00

- 5.2. Контрол на цвета чрез CSS: Стойност на цвят /свойство color/ -** позволява да определите цвета на всеки елемент от уебстраницата. То едновременно задава цвета на текста и рамката на елемента, освен ако отделно не сте посочили цвят на рамката чрез нейните свойства:

P {color: blue;} /*име на цвят*/

P {color: #0000FF;} /*шеснадесетична стойност*/

P {color: rgb (0, 0, 255);} /*rgb числа*/

P {color: rgb (0%с 0%, 100%) /*rgb процент*/

Стандартен цвят на текста – Цветовете се наследяват от родителските към дъщерните елементи. Ако посочите цвят за <body>, всички елементи на страницата ще наследят неговия цвят, което е ефективен начин за избор на стандартен цвят на текста за целия сайт. Следното правило определя цвета на елемента <body>: body {color: #006633;}.

Промяна цвета на връзките – можете да промените цвета на хипервръзки, като използвате следните специални класове от CSS: link – цвят на непосетена връзка, active – цвят на активна връзка, visited – цвят на посочена връзка/по подразбиране е лилав/.

Цвят на фона /свойство background-color/: позволява да задаваме цвят за фон на всеки елемент от страницата. Това включва и всяко вътрешно бяло поле. Може да се прилага както при елементи от тип блок, така и при съдържателните елементи. Следното стилово правило селектира елемент span и му задава цвят на фона: span {background-color: #ffffff;} Правилото е приложено към елемента span, който огранчжава името името на автора в кода: Detelina Milkoteva - съдържателен елемент с цвят на фона.

Цвят на фона на страницата: за да изберете фонов цвят на цялата страница, използвайте body селектор. Така цветовете ще се отнасят за цялата съдържателна част. По подразбиране цветът на всеки елемент е прозрачен. Затова всички елементи показват фона на самата страница, освен ако отделно не им е зададен друг цвят. Следното правило посочва цвят за фона на елемента <body>: body {background-color: #cccccc;} . Не винаги стандартният фон на потребителите е бял, затова вместо да разчитате на потребителските настройки, използвайте свойството background - color.

Обръщане цветовете на текст и фон: смяна на цвета на текста (обикновено черен) и фона (обикновено бял) е често срещан ефект в печатните издания. В Интернет това може да се постигне с избрани от Вас произволни цветове. Този похват най-често се прилага за заглавия не толкова при стандартен текст. Осъществява се лесно чрез стилово правило. Следното правило прави цвета на фона на елемент <h1> син, а цвета на текста – бял. H1 {color:#ffffff; background-color: blue; padding: .25 em;}. Вътрешното бяло поле на елемента е увеличено на 0,25 em, за да има по-голяма видима площ на фона.

Цвят на фона на таблици: лесно се променя цвета на фона на таблици с различен цвят, но винаги като използвате свойството background - color. То може да се прилага към всички елементи: таблица <table>, ред <tr>, заглавна клетка <th>, <td> клетка за данни.

table {background-color: #ffffff;} /*фонът на таблицата е бял*/

tr.hilite {background-color: yellow;} /*фонът на редовете от клас “hilite” са жълти*/

th.header {background-color:#cc9966;} /*фонът на клетките от клас “header” е светлокафяв*/

Добовяне на елемент : непосредствено след отварящия таг <p>:

```
<html>
<head>
<title>заглавие</title>
</head>
<body>
<h1>.....</h1>
<p>/*може да се опише картинката*/
</p> </body> </html>.
```

Контрол на фонови изображения с CSS: Поставяне на фоново изображение /background-image/ - позволява да определим изображението, което е избрано за фоново. В стандартния HTML единствената възможност е изображението да се наслагва многократно, докато запълни целия прозорец на брауъра. Това е и стандартното поведение на свойството background - image в CSS.

URL адрес на фоново изображение – за да поставите изображение като фон на страницата, използвайте за селектор елемента <body>, защото той е родителски елемент на цялата съдържателна част на документа. В стилното правило трябва да посочите относителен адрес на файла с изображението. В CSS има специален синтаксис за описване на URL адрес: body {background-image: url (paper.gif);}

Фон на страница – за да се повтаря изображението на цялата страница, използвайте за селектор body: body {background-image:url (avi.jpg);}

Фон на елемент – може да използвате свойството background - image за да добавите изображение като фон на произволен елемент: h1 {background-image: url (blutex.jpg); padding: .25 em;}

Свойството padding добавя допълнително бяло поле около елемента <h1>, за да се вижда по-голямата част от изображението.

Повторяемост /background-repeat/ - можем да контролираме наслагването на фоновете изображения на страницата или в конкретен елемент. За да има ефект, трябва вече да сме посочили фоново изображение посредством background - image. Затова background - repeat винаги се използва редом с background - image.

- **вертикална** – като използваме стойността repeat на свойството background - repeat, можем да създаваме графичен елемент, който се повтаря по вертикалната ос: body {background-image: url (ivy.jpg); background - repeat: repeat-y}

h1, h2, h3, p {image-left: 70 px;}

- **хоризонтална** – като използваме стойността repeat - x на свойството background - repeat, можем да създаваме графичен елемент, повтарящ се по

```
горизонталната ос. Body {background - image: url (imy.jpg); background - repeat: repeat-x;}h1 {margin-top: 50px;}
```

Еднократно фоново изображение – за да се покаже изображението само веднъж, можем да използваме стойността no-repeat на свойството background-repeat. Така може да постигнете воден знак върху страницата си, т. е. изображението може предварително да е обработено както искаме с графична програма.

```
body {background-image: url (ivy.jpg);
```

```
background-repeat: no-repeat;}
```

```
h1 {margin-top: 50px;}
```

В горния списък със стилове свойството margin-top има стойност 50 px, за да разположи елемента <h1> под фоново изображение. За да преместим това изображение на друго място в страницата, можем да използваме свойството background-position. Три вида позициониране: процент, разстояние и ключова дума: left, left top, top left, top, center, bottom...

Центриране на фоново изображение: използва се свойството background-position, за да разполагате вертикални, хоризонтални и еднократни изображения. Една от най-често срещаните възможности е комбинацията на това свойство с background-repeat за центриране на фоново изображение и постигане имитация на воден знак на печатна страница. body {background-image: url (ivy.jpg); Background-repeat: no-repeat; Background-position: center;}

Позициониране на вертикални и хоризонтални изображения – можем също да позиционираме изображения, които се повтарят по вертикалната или хоризонталната ос на уебстраницата. Следното стилово правило позиционира вертикално повтарящо се изображение в десния край на прозореца на браузъра:

```
Body {background-image: url (ivy.jpg);
```

```
Background-repeat: repeat-y
```

```
Background-position: right;}
```

```
P {margin-right: 70px; }
```

Можете да позиционирате и хоризонтално повтарящи се изображения. Следното стилово правило позиционира хоризонталното повтарящо се изображение в долния край на прозореца на браузъра:

```
Body {background-image: url (ivy.jpg);
```

```
Background-repeat: repeat-x Background-position: bottom;
```

5.3. Стандарти за визуално оформление със CSS

Визуално оформление със CSS – определя се как блоковете със съдържание да бъдат изобразени от браузъра. Той се базира на йерархичната структура на HTML и на типа на елемента. Делят се на три вида елементите на HTML според начина на показване:

1/Блок - елементите от тип блок се изобразяват като блокове от съдържание, например параграф. В тях стоят карета със съдържанието на блока;

2/Списък - тези елементи се намират в елемент от тип блок. Те съдържат текст, без да формират отделен блок;

3/Списък - елементите от тип списък обединяват съдържателни елементи, оформени като списък. Те се изобразяват като номериран или неномериран списък. В CSS можем да избираме рамка, вътрешно и външно бяло поле на всички елементи от тип блок. В някои случаи стойностите зависят от съдържащия блок (родителя) на елемента, който искате да промените. Например, ако зададете процентна стойност за външно бяло поле, тя ще се изчисли спрямо съдържащия блок. Свойството **display** указва типа при показване на елемента. Когато работите с HTML, вероятно не използвате често това свойство, защото браузърът ползва стандартните типове по подразбиране за всеки един елемент. Например елементите `<h1>`, `<p>` и `<blockquote>` са от тип блок, а `` и `<i>` са съдържание. В повечето случаи ще искате браузърът да показва всеки елемент според стандартния му тип. Дори може да се наложи да промените това. Следното стилово правило променя типа на елемента `<h1>` от тип блок в тип съдържание: `h1 {display: inline;}` В резултат на това правило елементът `<h1>` започва да реагира като съдържателен елемент. Свойството `none` на свойството `display` позволява да скриете даден елемент, така че да не се показва в браузъра. Това може да ви бъде от полза, да се вижда само част от информацията от Вашата уебстраница.

Блоков модел на CSS: - описва правоъгълните блокове, в които се разполага дадено съдържание от страницата. Всеки елемент от тип блок, който създадете, се изобразява в прозореца на браузъра под формата на пълно със съдържание каре. Всяко каре може да има рамка, вътрешно и външно бяло поле. Карето със съдържанието е най-вътрешният правоъгълник, заобиколен от вътрешно бяло поле, рамка и външно бяло поле. Цветът на вътрешното бяло поле е същия като този на фона на елемента, докато външното бяло поле винаги е прозрачно. Рамката разделя вътрешното от външното бяло поле.

```
p {margin: 2em;
```

```
padding: 2em;
```

```
border: solid thin black; background-color: white; }
```

Свойството margin /външно бяло поле/ и **padding /вътрешно бяло поле/** имат стойност `2em` и за четирите страни на блока. Свойството `border` е обобщаващо и задава стойност `solid` за `border-style`, `thin` за `border-width` и `black` за `border-color`. Свойството **background-color** прави фона на параграфа бял. В CSS може да промените свойствата `margin`, `padding`, `border` за всяка страна на блока поотделно:

```
p {margin-left:50px;
```

```
margin-right:20px;
```

```
padding-left:10px;
```

```
padding-right:20px;
```

```
border-left:solid thick black;
```

```
border-right:solid thick double;
```


background:white;} Може да ползваме два вида мерни единици за свойствата – размер или процент. Чрез свойствата margin може да контролираме външното бяло поле на блока, то е прозрачно – показва фона на съдържащия елемент. Външните бели полета подобряват четимостта на текста, създават изместени елементи и добавят разстояние около изображенията.

Обобщаващото свойство margin на външното бяло поле , с което можем да променим едновременно и четирите отделни свойства margin – от четирите страни на елемента. Можем да посочим размер или процентна стойност. Най-често свойството се ползва за едновременна промяна на размера на белите полета от всички страни на блока: p {margin: 2em;}

Вътрешно бяло поле /свойство padding/- контролира вътрешното бяло поле на блока. Вътрешното бяло поле се намира между съдържанието на елемента и рамката. То наследява цвета на самия елемент – ако елементът <p> има бял фон, цветът на вътрешното поле също ще бъде бял. Ако добавите рамка около елемента, почти винаги ще Ви се наложи да използвате padding, за да увеличите разстоянието между елемента и съдържанието.

Padding е обобщаващо свойство, с което можем да променим всички четири отделни стойности за padding само с едно правило. Посочваме точен размер или процентна стойност. За разлика от margin, вътрешните бели полета не се обединяват и не можете да укажете отрицателни стойности. Най-често свойството се ползва за едновременна промяна на размера на белите полета от всичките четири страни на съдържанието. Може да посочим различен размер вътрешни бели полета в едно общо правило.

Рамка: чрез свойството border се контролира външния вид на рамката около елементите. Тя е разположена между вътрешното и външното бяло поле. Съдържа 20 различни свойства, много от които са твърде специфични и се прилагат рядко: border, border-left, border-right, border-top, border-bottom. Тези свойства позволяват да се сменят вида, цвета и дебелината на рамката на всяка една от четирите страни на блока, както и за целия блок.

Вид на рамката: border-style е най-важното свойство, тъй като, ако не е посочено, рамката няма да се вижда. Използват се ключовите думи: none-не се показва рамка, това е стойността по подразбиране; dotted – пунктир, dashed – прекъснатата линия, solid – плътна линия, double – двойна, groove – триизмерна рамка, която изглежда гравирана в страницата, ridge – триизмерна рамка, която изглежда релефна на страницата, inset – триизмерна рамка, която изглежда вградена в страницата, outset – триизмерна рамка, която създава илюзия, че целия блок е релефен. Ето пример за свойството border-style: p {border-style:solid;}

Дебелина - чрез ключова дума или с точна стойност: thin-тънка, medium-средна, това е стойността по подразбиране, thick-дебела. Дебелината на рамката зависи от браузъра. Точната стойност може да бъде абсолютна или относителна. Проценти не са позволени: p {border-width: 1px; border-style:solid; }

!!!Рамка няма да се покаже, ако не се посочи стойност за border-style.

Цвят: чрез свойството border-color може да променяме цвета на елемента border: p {border-color: red; border-width: 1px; border-style: solid;} Стандартния цвят на рамката съответства на цвета на съдържанието. Например долното правило задава червен цвят на елемента. Рамката също ще е червена, защото не е указана друга стойност: p {color: red; font: 12pt arial; border: solid;} Може да ползваме различен цвят

за всяка от четирите страни на рамката: border-left-color, border-right-color, border-top-color, border-bottom-color. Така конкретна част от рамката ще се оцвети както желаем.

Обобщаващо свойство за рамка: свойството border променя едновременно четирите страни на рамката. Може да задаваме border-width, border-style, border-color в произволен ред. Задължително е да се посочи border-style. Ако не укажем border-width, стойността по подразбиране е medium. Ако не посочим border-color, цветът на рамката ще бъде като цветът на елемента. Рамката ще бъде черна: p {border: 1px solid black;}

Следното правило задава стойност solid на border-style. Border-width е 1 пиксел. Border-color е blue. Подредбата на стойностите не е от значение: p {border: double blue 1px solid;}

Плаващ текстов блок: - Свойството float може също да се използва за подравняване на плаващ текстов блок вляво или вдясно на основния текст. Като използвате width, height, може да създаваме текстово каре:

```
.floatbox {width:125px;
Height: 200px;
Float: left;
Background-color:#ffff66; /*жълто*/}
```

Позициониране на елементи - свойствата на CSS дават контрол на начина, по който се разполагат блоговите елементи в уеб. Може да избираме 4 различни вида позициониране, които осигуряват създаване на страниц, чиито елементи са разположени на различни места, и то без да прибягваме до таблици. Така съдържанието е по-достъпно и запазва употребата на таблици само за таблични данни.

Позиционирането в CSS върви със скриптове, за повишаване **интерактивния интерфейс на уебстраниците**. Скриптовете са малки програми, написани на скриптов език – Java Script. Комбинацията между скриптов език и свойствата за позициониране в CSS предлага повече възможности за взаимодействие с потребителя от простите хипертекстови връзки. Скриптовете могат да променят съдържанието на уебстраницата, след като бъде заредена от браузъра, да създават интерактивни формуляри и игри, както и да симулират анимация.

Нормално позициониране на елементите: По подразбиране браузърът разполага елементите на страницата един след друг, в зависимост дали са от тип блок, или съдържание. При нормалното разполагане на елементите блоковете се подреждат вертикално един под друг, като започват от най-горния съдържащ блок. Лявата страна на всеки блок допира лявата страна на съдържащия елемент, освен ако е плаващ или схемата е обърната от дясно на ляво, като е в дясноподравнените езици, например иврит /Hebrew/. Разстоянието между блоковете се определя от стойността на свойството margin. Нормално разполагане на елементите съответства на последователността при стандартния HTML:

```
<body>
<h1>The document heading</h1>
<p>A paragraph of some text</p>
```

`<p>Another paragraph of text</p></body>` Елементът `<body>` е съдържащият елемент за цялото съдържание на уебстраницата. Елементите в него се изобразяват точно в реда, в който са записани в кода - от горе надолу. В конкретния пример това е елемент `<h1>`, следван от два елемента `<p>`. Елементите не стоят един до друг, освен ако са плаващи или са от тип съдържание. Нормалното разполагане на съдържателните елементи е хоризонтално, като започва от горната част на съдържащия блок. Елементите от тип съдържание обхващат редове текст, например в елемента `<paragraph>`. Браузърът разполага текста в елемента, като го пренася на следващия ред според ограниченията на съдържащия блок или на размера на прозореца.

Вид на позиционирането: следващото стилово правило използва свойството `position` със стойност `absolute`, за да позиционира елемента `<div>`: `div {position: absolute;}`

Отместване – чрез свойствата за отместване може да се посочат точните стойности за позициониране на всеки елемент, който има стойност `absolute`, `fixed`, `relative` на свойството `position`. Свойствата за отместване определят позицията на елемента спрямо съдържащия го блок, като посочват разстоянието от краищата му. Например следния код на HTML и стилово правило създават абсолютно позициониран елемент `<div>`, който е отместен спрямо съдържащия си блок със 130 пиксела от левия край и със 100 пиксела от горния край:

```
<html>
<head>
<title>Absolute Positioning</title>
<style type="text/css">
div {position: absolute;
left: 130px;
top:100px;
background-color:#ffcc33;}
</style>
</head>
<body>This is an absolutely positioned division element.</div>
</body>
</html>
```

В този пример съдържащият блок е елементът `<body>`. Разделът е отместен със 130 пиксела спрямо левия край и със 100 от горния. Накрая е добавен цвят на фона. Препоръчва се пиксел за размер на отместванията.

Абсолютно позициониране – когато сте избрали абсолютно позициониране, разположенето на елемента се определя спрямо съдържащия му блок. Абсолютното позиционираният елемент излиза от нормалната схема на документа, като това не влияе на другите елементи. Създава се нов съдържащ блок за елементите в него. Те могат да

се разполагат според нормалната схема или да имат собствено позициониране. Абсолютните елементи могат също да припокриват съдържанието на друг блок. По-долу се вижда код на HTML и стилни правила. Вътрешният елемент се намира в съдържащия външен елемент:

```
<html>
<head>
<title>Absolute Positioning</title>
<style type="text/css">
div.outside {position: absolute;
left:130px;
top:100px;
width:200px;
height:200px;
background-color: #ffcc33;}
div.inside {position: absolute;
top: 150px;
left:150px;
width:150px;
height:50px;
border:soli thin black;
background-color:#ffffff;}
</style>
</head>
<body><div class="outside">Outside
<div class="inside">Inside</div>
</div>
</body>
</html>
```

Фиксирано позициониране – вид абсолютно позициониране с условието, че съдържащият блок на фиксирания елемент винаги е прозорецът на брауъра. Фиксираните елементи стоят винаги на едно и също място на екрана, независимо как потребителят превърта документа. Фиксираното позициониране позволява създаването на разположение от тип фрейм, където съдържанието може да се превърта независимо от навигационния елемент:

```
div.fixed {position: fixed;
left:10px;
top:10px;
background-color: #cccccc;
width: 100px;
height:200px;
padding-top: .5em;
padding-left: .5em;}
div.content {position: absolute; left:130;}
```

Фиксираният елемент има точна ширина и височина, които определят размера на елемента в прозореца на брауъра. Отместванията left, top посочват фиксираната позиция на елемента. Елементът, в който се намира съдържанието на страницата, няма конкретна ширина и височина, което позволява размерите му да се променят спрямо прозореца на брауъра. Отместването left позиционира елемента, така че да не припокрива фиксирания елемент.

Относително позициониране – стойността relative на свойството position позволява изместването на нова позиция даден елемент от нормалното му място в документа. Новата позиция е относителна спрямо стандартната позиция на елемента. Чрез свойствата за отместване може да нагласяме точното местоположение на елемента. Относителното позициониране понякога е полезно за създаване на колонна структура чрез CSS, но като цяло се използва много рядко. Като позиционираме елемент относително, мястото където е заемал в нормалното разполагане на елементите, остава резервирано, това е разликата с абсолютното позициониране, където елементът напълно излиза от стандартната схема.

Включването на CSS в страницата става по два начина:

1. Директно, с употребата на тага <STYLE> в главата на документа:

Пример

```
<STILE TYPE="text/css"><!--
A:hover{text-decoration:none;}
B {color:#0000FF;}
--></STILE>
```

Чрез включването на показания стил връзките в страницата ще бъдат без подчертаване, получерният текст (с употребата на тага) да бъде със син цвят.

2. В отделен файл, в който са включени само данните за стила и който може да бъде свързан със съответната страница посредством тага <LINK>.

Пример:

```
<LINK REL="stylesheet"TYPE="text/css"href="ezine.css">
```

Обикновено се използва именно този метод, тъй като към един и същ файл, съдържащ описание на стила, могат да се свържат всички страници от сайта и по този начин с малки корекции в него може да се променя едновременно видът на целия сайт.

Прилагането на каскадните стилове надгражда, улеснява и добавя нови възможности при използването на таговете в даден HTML документ. Силата на CSS е използването му като външен файл, контролиращ съдържанието и външния вид на целия уеб сайт. Въвеждането на ново форматиране или промяна на съществуващо такова в CSS съдържанието автоматично ще се наложи върху всички страници. Много по-лесно е глобално да се контролира HTML таговете от един каскаден файл, отколкото на всяка страница ръчно да се променя кодът. За създаването на каскаден файл може да се използва обикновен текстов редактор – notepad, Notepad++, Sublime... но да се запази файлът със разширение css. Връзката между HTML документа и css файла се описва в секцията head чрез тага link и атрибут, указващ пътя до стилизиращия файл

```
(<link rel="stylesheet" href="css/styles.css" type="text/css">).
```

Предефиниране на HTML таг в CSS файл започва с въвеждане на избрания таг, а в големи фигурни скоби се описват атрибутите и техните стойности за постигане на крайния резултат.

Пример 1:

```
Body  
{  
background-color:silver;  
font-family:verdana;  
font-size:16px;  
color:#cccccc;  
}
```

Има тагове, позволяващи форматиране на текстово съдържание, а също така и отпадането на поддръжката на тага font, чрез който основно се форматира текст на ниво символи. Ето други възможности за стилизиране, т. е. атрибути за **форматиране на текст**:

- Color – определя цвета на символите, стойностите може да се въвеждат чрез шеснайсетичен код (#FFFFFF), наименование на цвят (white) или чрез функцията rgb (255,255,255);
- Text-align - подравнява съдържанието на текста спрямо размера на тага, според стойността:
Left – ляво, right – дясно, center – среда и justify – двустранно;
- Font-family - задава шрифт на текста;
- Font-size – определя размера на символите, като мерни единици могат да са px – пиксели или pt – пойнтове;
- Font-weight – форматира текста като получер (bold) или определя символите да се виждат в нормален вид. Възможните стойности са: normal или bold;

- Font-style – форматира текста като курсив (italic) или определя символите да се виждат в нормален вид. Възможните стойности са: underline или none;
- Text-transform – променя контрола на символите: uppercase - всички символи са главни, lowercase - всички символи са малки, capitalize - главна буква на всяка дума и none - без контрол;
- Text-indent - отстъп на изречение;
- Letter-spacing – отстъп между символите (px), (pt);
- Line-height – определя височина на ред/междуредие, стойностите може да са цели числа, дробни числа, проценти, px, pt или cm.

Пример:

```
{
text-align:left;
font-family:Verdana;
font-size: 14 px;
color:black;
font-weight:bold;
line-height:1,5;
}
```

5.4. Форматиране на фон със CSS

Избирането на добър външен вид на уеб страниците е свързано с background оформлението, както на цяла страница, така и на отделни елементи от нея. Два са начините за създаване на стилин фон: да използваме цвят или изображение. Но трябва да се съобразяваме с вида на изображенията, ако са векторни са за предпочитане, запазват качеството си при трансформация.

```
body { background-color:yellow}
h1 {background-color:#00ff00}
```

```
#yellow{background-color:yellow}
.fon{background-color:#00ff00}
```

Изображенията като част от информационното съдържание на HTML страниците много често намират приложение и като фон. Предварително избраното изображение трябва да се постави в директория, до която ще има достъп уеб страницата.

- Background-image – определя използването на изображение за фон чрез атрибут url (път до изображение);
- background-repeat – определя начина на повторение на изображението при по-голям размер на страницата от този на изображението. Стойностите са repeat – y: повторение вертикално по страницата, repeat – x: повторение хоризонтално по страницата, no-repeat: без повторение на изображението;
- background-position – позициониране на изображение, ако то е с по-малък размер от този на страницата. Местоположението се задава с хоризонтална и вертикална стойност в проценти, пиксели или служебни думи, разделени с интервал: left top, left center, left bottom, right top, right

center, right bottom, center top, center center, center bottom, 30% 70%, 20px 80px;

- background-attachment – фиксира изображението винаги във видимата област на страницата. Независимо от размера на страницата, стойността на атрибута е fixed;

Пример:

```
Body
{background-image:url('images/planet.jpg');
background-repeat:no-repeat;
background-position:center;
}
```

5.5. Рамки със CSS

Рамките са едни от най-силните ефекти в уеб програмирането като с тяхна помощ лесно могат да се открият блокове, меню бутони или разделяне на уеб страницата на логически области. Рамката е универсален стил, който може да се приложи към изображенията, inline тагове и текстови области. За създаването на такъв стил трябва да се определят познатите характеристики: цвят на рамката, стил и размер. В новата версия на CSS рамките имат едно ново допълнение – радиус. Тази характеристика определя заоблянето на върховете на правоъгълната област, подходяща за създаване на менюта или бутони за сайта.

Атрибути и стойности на рамка са:

- border-style – стил на рамката с осем варианта (dotted, dashed, solid, double, groove, ridge, inset, outset). При единична стойност за атрибута всички страни ще се визуализират с еднаква рамка, но ако се разделят с интервал няколко стойности, те ще заемат местата в следния ред: горе-дясно-долу-ляво, като при два параметъра ще се редуват през една страна.
- border-color - цвят на рамката, който се определя чрез шеснайсетичен код, служебна дума или функция rgb. По аналогия при стила на рамката може да се посочат няколко цвята, разделени с интервал след атрибута, като редът на оцветяване ще е същият.
- border-width – плътност на рамката, която се определя в пиксели. С единична стойност се задава еднакъв размер на всички страни. Ако зададете две стойности, първа ще определи размер на рамката отгоре и отдолу, а втората – вляво и дясно на селектора. Могат да се определят и четирите страни, като се изброят четири стойности, разделени с интервал.
- Border - radius – определя радиуса на заобляне на върховете при селекторите в пиксели. Както и при останалите стилове на рамките, и тук възможните описания на заобляне могат да са с една, две или четири стойности. Тази нова възможност на рамките изисква допълнително описание за различните версии на браузърите, особено за по-старите. Използват се още два атрибута: единият е – moz – border - radius за Mozilla FireFox, а другият – webkit – border - radius за Safari и Google Chrome, така ако се наложи да използвате тази характеристика, е желателно да посочите и трите атрибута с идентични стойности.

Пример:

```
div.button
```




```

{
text-align:center;
border-style:solid;
border-width:3px;
border-color:#000000;
background-color:#0094FF;
width:160px;
color:#ffffff;
border-radius:25px;
-moz-border-radius:25px;
-webkit-border-radius:25px;
}

```

Начинът, по който изглежда и се използва всеки уеб сайт зависи от използваната технология за неговото разработване. Тук се включват HTML и XHTML, каскаден набор от стилове (CSS), плъгин-модули, скриптове от клиентска страна (JavaScript) и скриптове, разположени на сървъра. Тези технологии се поддържат от основната организация, която установява международни стандарти за www, наречена W3C (World Wide Web Consortium).

За да се избегнат грешки при успоредната работа с HTML и CSS е да се инсталира специализиран софтуер за разработване на уеб приложения, които следят правилното изписване на таговете и въвеждането на основни атрибути. Освен, че определят правилния стандарт за писане, повечето приложения могат да помогнат за откриване на допуснати волни или неволни синтактични грешки. Именно за това може да се използва и периодично обръщане към сайта W3C – консорциума, отговарящи за уеб стандартите. W3C публикува в своята страница online валидатори за HTML и CSS с няколко възможности за проверка.

Зареждайки адреса за валидиране на код (<http://validator.w3.org/>), се виждат три възможности за проверка: Валидиране чрез URL адрес, чрез прикачен файл и чрез поставяне на кода в областта за проверка. След натискане бутона Check в съдържанието на страницата ще се появяват допуснатите грешки, с описание къде се намират и от какъв тип са. Ако всичко е наред, се появява надпис за успешно валидиране на уеб страницата: This document was successfully checked as HTML. Аналогично може да се провери и CSS съдържанието на сайт, използвайки адреса: <http://css-validator.org>.

Уеб страниците се създават с помощта на езика HTML (HyperText Markup Language). Чрез него се дефинира структурата на уеб страницата, указва се на брауъра какво съдържа страницата и как трябва да са разположени елементите по нея. HTML документите представляват текстови файлове, които съдържат текст и инструкции за форматирането му (наречени тагове) по време на визуализацията в конкретния брауър.

С HTML се описва как да се вмъкнат таблици, изображения, списъци, формуляри и да се задават хипервръзки към различни ресурси в Интернет. Възможностите на HTML се обогатяват при комбиниране с каскаден набор от стилове (CSS – Cascading Style Sheets), който дефинира как трябва да изглежда външния вид на уебстраницата. Същността е в разделянето на съдържанието от представянето, което позволява да се създаде сайт, който е структурно стабилен, визуално привлекателен и се зарежда по-бързо.

Ефективността се постига чрез еднократното дефиниране на таговете за шрифт, размери, стил, цвят, подравняване и други характеристики, като този набор от стилове се оформя в отделен CSS файл, който се свързва с всички страници на сайта. Така при необходимост от промяна във форматирането на даден сайт не се налага да се търсят HTML файлове на съответните страници и да се правят многократни промени, а промяната се извършва еднократно – само в CSS файла със стиловете.

С навлизането на мобилните технологии в ежедневието и необходимостта от показването на едно и също съдържание от уеб пространството на таблети, мобилни телефони и електронни книги, все по-често се използва модифицирана версия на HTML – езикът HTML 5. Той е предназначен да създаде виртуално пространство, което е по-стандартизирано и по-съвместимо с тези приложения и системи.

Плъгин – модулите са допълнителни технологии, които се смесват незабележимо с брауъра, като дават възможността за възпроизвеждане на аудио, видео и като цяло мултимедийни възможности на страницата. Най-популярна е Flash технологията. Визуализирането става възможно, ако предварително е инсталиран Flash плейър към брауъра. Тази технология позволява да се вградят мултимедийни файлове (аудио, анимация, видео, навигация) директно в уеб страницата, да се задават атрактивни ефекти на преходи, богати цветове и отлично синхронизиране на звук.

- **Скриптовете**, изпълнявани от клиентската страна, са малки програми, които се свалят с уеб страницата и носят интерактивност на сайта.
- **JavaScript** е главния клиентски скриптов език, който е вграден в повечето брауъри и може да изпълни всякакъв вид функции в рамките на уеб страницата, без да има нужда от помощта на сървъра. Използва се за манипулиране на изображенията и текста, за стартиране на малки изкачащи прозорци, за разширяване и свиване на падащи менюта, за събиране на информация от формуляри и като цяло за динамични елементи на сайтовете. Скриптовете, изпълнявани от сървърната страна, свързват уеб страницата с база от данни или приложение, позволяващи да се настрои сайта за различни потребители и да се обслужват динамични страници (онлайн магазини, онлайн разплащания и други). Възможни скриптове са PHP, ASP, JSP и др.

5.6. Задаване на име и шрифт, размер и цвят

Не си струва да се сменя шрифтът, указан по подразбиране в брауъра, защото по този начин ще се гарантира наличието на съответния шрифт на компютъра на потребителя и коректното показване на уебстраницата. По-голямата част от читателите не могат да назоват нито един от шрифтовете например в операционната система UNIX. След проучване сред учениците ми, с които съм имала възможността да работя, искаха на техните страници да се показва именно шрифтът **Arial**. И ето че възниква парадокс. Предвиждайки такива проблеми, създателите на CSS са съобразили да включват CSS свойството `font-family`, което се използва за задаване на шрифт по избор. Това свойство може да приема до три стойности едновременно. Първата и втората стойност указват локалното име на шрифта (името, с което шрифта е познат на операционната система на потребителя). Последната стойност указва име константа, което име във всеки брауър съответства на реално съществуващ шрифт. По този начин ако шрифтът, указан в първата стойност, липсва на компютъра на потребителя, брауърът се опитва

да намери втория (указания с втората стойност). Ако и той липсва, тогава браузърът ще използва шрифт, съответстващ на зададената константа.

За третата стойност е най-добре да използвате или константата serif, която съответства на стандартен шрифт, или константата monospace, която съответства на шрифт, при който ширината на всички символи е еднаква. Ако трябва да измените кегела на шрифта, най-добре е да използвате стандартните за тази цел CSS свойства (по-подробно вижте „Задаване на кегел на шрифта“ малко по-нататък в тази глава.)

За да зададете име на шрифта, може да използвате едно от следните CSS свойства:

- Font – family – задава името на шрифта;
- Font – задава някой от параметрите на шрифта, в това число и име (фактически дублира използването на свойството font-family, но едновременно с това можете да задавате и други параметри на шрифта).

```
<HTML>
<HEAD>
<TITLE>Font family</TITLE>
</HEAD>
<BODY>
<DIV>ABCabc</DIV>
<DIV style="font-family: serif;">ABCabc (serif)</DIV>
<DIV style="font-family: monospace;">ABCabc (monospace)</DIV> <DIV
style="font-family: Arial, serif;">
bABCabc (Arial or serif)</DIV>
<DIV style="font-family: 'Times New Roman', monospace;">
bABCabc (Times New Roman or monospace) </DIV>
</BODY>
</HTML>
```

ABCabc ABCabc (serif)
ABCabc (monospace)
ABCabc (Arial or serif)
.ABCabc (Times New Roman or monospace)

Задаване на размера на шрифта

Размера на шрифта можете да зададете чрез използването на едно от следните две CSS свойства:

- font-size — задава размера на шрифта;
- font — задава някой от параметрите на шрифта, в това число и размера (фактически дублира използването на свойството font-size, но имате възможност да зададете допълнителни параметри.)

Стойността за размера най-често се задава в пиксели или пунктове.

Внимание! При използване на CSS СВОЙСТВОТО font трябва да укажете и някой допълнителен параметър (например име на шрифта и или цвят). В противен случай използването на това свойство е неоправдано.

Задаване на размер на шрифта:

```
-- <CD>\Samaouchitel\font-size.html -->
```

```

<DOCTYPE HTML PUBLIC "-//VJ3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
<TITLE>Font size</TITLE>
</HEAD>
<BODY>
<DIV style="font: 16px black; ">16px black</DIV>
<DIV style="font: 25px serif;">25px serif</DIV>
<DIV style="font: 30pt serif;">30pt serif</DIV>
<DIV style="font-size: 150%;">150%</DIV>
<DIV style="font-size: 200%;">200%</DIV>
</BODY>
:</HTML>
16px black
25px serif
30pt serif
150%
200%

```

Задаване на цвят на шрифта

Цвят на шрифта можете да зададете чрез някое от следните две SS свойства:

- font-color — задава цвета на шрифта;
- font — задава някой от параметрите на шрифта, в това число и цвета (фактически дублира използваното на свойството font-color, но имате възможност да зададете допълнителни параметри).

Стойността на цвета се задава чрез:

- във формат RGB (в пълна или съкратена форма),
- с помощта на константни цветове (вижте по-подробно в приложение Г, “Стандартни цветове”).

```

<!-- <CD>\Samouchitel\font-size.html -->
. : Г7УРЕ HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
-TУЕ>
<HEAD>
<TITLE>Font size</TITLE>
</HEAD>
<BODY>
<DIV style="font: 16px black;">16px black</DIV>
<DIV style="font: 25px serif;">25px serif</DIV>
<DIV style="font: 30pt serif;">30pt serif</DIV>
<DIV style="font-size: 150%;">150%</DIV>
<DIV style="font-size: 200%;">200%</DIV>
</BODY>
<HTML>

```

```

16p\ black
25px serif

```

30pt serif

150%

200%

Безсмислено е да публикуваме резултата от примера (цветно изображение) в черно-бяла книга.

5.7. Управление на абзаци, отстъп за нов ред

Отстъпът за нов ред обикновено се използва, за да се укаже, че започва нов абзац. Можете да го реализирате, като поставите един или няколко символа за непрекъсваем интервал (), но правилният начин е, като използвате средството на CSS - свойството `text-indent`, със съответната стойност, указваща големината на отстъпа. В примера по-долу ще видите: (отстъпът е зададен за всички параграфи в HTML документа.

```
<!-- <CD>\Samouchitel\text _ indented _ line.html -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
<TITLE>Indented line</TITLE>
<STYLE type="text/css">
<! —
BODY {
text-indent: 40px;
}
—>
</STYLE>
</HEAD>
<BODY>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus auctorem purus a massa.
Fusce tincidunt. Duis ipsum arcu, pharetra vel.
</BODY>
</HTML>
```

Ако искате да придадете на текста привлекателен вид, може да направите първата буква по-голяма. В наше време, когато Интернет прониква все по-дълбоко в живота ни, се появяват сайтове, съдържащи електронни версии на различни литературни произведения. Уебдизайнерите се нуждаят от средство, което да им предостави възможност да постигнат подобен ефект. Разбира се, може да разделите текстът като на първата буква приложите CSS клас, но по-правилното е, ако се използват готови стандартни CSS свойства.

За задаване на определени параметри на първата буква е създаден псевдоелементът: `first-letter`. В примера за първата буква (съответно първия ред) се задава отстъп от 40 пиксела и размер от 40 пиксела. Освен това се задава син цвят на символа и сив за фона.

```
<!-- <CD>\Samouchitel\text _ first _ letter.html -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
<TITLE>First letter</TITLE>
```

```

<STYLE type="text/css">
<! —
:first-letter {
margin-left: 40px; font-size: 50px; color: blue;
background-color: silver;
}
—>
</STYLE>
</HEAD>
<BODY>
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
</BODY>
</HTML>

```

5.8. Управление на заглавия

В тази глава ще разгледаме създаването на заглавия с помощта на таговете <H1>-<H6>. Това са тагове от блоков тип - след тяхното изпълнение следва преход на нов ред. Подредени са йерархично, като на най- високото ниво отговаря <H1>, а на най-ниското <H6>, като размерът на шрифта намалява от <H1> към <H6>. По подразбиране заглавието се изобразява с получер шрифт.

Рано или късно ще си зададете въпроса: “Защо въобще ми трябва тагове за заглавия, след като мога да използвам <DIV>, или <P>?” Най-напред, това е необходимо от гледна точка на структурата на документа, например защото с тяхното използване значително се улеснява работата на синтактичните анализатори на страницата. От друга страна, ако копирате текст от браузъра и го поместите в текстов редактор, лесно можете да определите кое е заглавие (например MSWord автоматично разпознава заглавията, ако копирате текст от Internet Explorer).

Задаване на размера на заглавието

Заглавията (тагове <H1>-<H6>) са кратки фрагменти от текста, които указват началото на раздели или глави в относително дълги документи. Към тях можете да прилагате практически всички CSS свойства, отнасящи се до текст и шрифт.

```

<!-- cCO\Samouchitel\header_size.html -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
<TITLE>Header size</TITLE>
<STYLE type="text/css">
<! —
H1 { font-size: 25px; }
H2 { font-size: 20px; }
—>
</STYLE>
</HEAD>
<BODY>
<H1>1. Introduction</H1> text...
<H2>1.1. Purpose</H2> text...
<H2>1.2. Scope</H2> text...

```

```
<H1>2. Main idea</H1> text...
<H1>3. Overview</H1> text...
</BODY>
</HTML>
```

Внимание! Може да срещнете използване на един и същ размер на шрифта за няколко йерархични нива. В този случай разликата между заглавията се явява единствено номерацията.

5.9. Управление на изображения

Ще се спрем на използването на HTML тага . Това е типичен инлайн таг, т.е. заема толкова място от реда, колкото е необходимо за визуализиране на неговото съдържание. Съществува само една негова алтернатива, а именно показване на изображения в качеството им на фон (най-често за елементи, дефинирани с HTML таговете и <DIV>). Този начин на визуализиране на изображения обаче не е изцяло равнозначен на използването на тага , но има един сериозен недостатък. В случай че фоновото изображение не може да бъде намерено (или браузърът на потребителя е настроен да не показва изображения), то за потребителя ще бъде невъзможно да види изображението чрез контекстното меню (опция Show picture). Освен това изображението не може да се мащабира, ако се показва като фоново.

Задаване на размер на изображение

Тъй като блокът на тага е правоъгълен (както и блоковете на останалите HTML тагове), хоризонталните и вертикалните размери на изображението се задават съответно с CSS свойствата width и height. Проблеми с използването на тези свойства могат да възникнат само при относително по-старите браузъри.

Алтернатива на тези CSS свойства се явяват средствата на HTML - едноименните атрибути width и height на HTML тага .

```
<!-- <CD>\Samouchitel\image _ size.html -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
<TITLE>Image size</TITLE>
</HEAD>
<BODY>
<IMG src="images/rose.jpg" style="width: 20px; height:
^ 20px;">&nbsp;
<IMG src="images/rose.jpg" style="width: 40px; height:
^ 40px;">&nbsp;
<IMG src="images/rose.jpg" style="width: 60px; height:
'b 60px;">&nbsp;
<IMG src="images/rose.jpg" width="60" height="60">
</BODY>
</HTML>
```

Поставяне и премахване на рамка около изображение

Всеки дизайнер решава самостоятелно дали да постави рамка около изображението. Когато изображението има малки размери и се явява част от общия скелет на страницата, отговорът е очевиден - рамка не се поставя. Ако обаче става въпрос за

големи изображения (например фотографии, схеми или рисунки), въпросът е спорен и отговорът обикновено зависи от естетичните предпочитания на дизайнера.

Рамка на изображение, зададено с HTML таг , се задава с помощта на производните на CSS свойството border, които предоставят на разработчика възможността да манипулира отделните страни на рамката, както прецени за добре. Най-често се използват CSS свойствата border-width, border-style, border-color.

Алтернатива на тези CSS свойства се явява атрибутът border на HTML тагът .

```
<!-- <CD>\Samouchitel\image _ border.html -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
<TITLE>Image border</TITLE>
<STYLE type="text/css">
<! —
IMG {
width: 60px; height: 60px;
}
//—>
</STYLE>
</HEAD>
<BODY>
<IMG src="images/rose.jpg">&nbsp;
<IMG src="images/rose.jpg" style="border-style:
'b none; ">&nbsp;
<IMG src="images/rose.jpg" style="border: 10px solid
^ red;">&nbsp;
<IMG src="images/rose.jpg" border="0"> <IMG src="images/rose.jpg" border="10">
</BODY>
</HTML>
```

Задаване на външни отстъпи

И тук положението е както при рамките - за малки изображения, явяващи се част от общата композиция на страницата, отстъпи не се задават. За големите изображения (фотографии, схеми или рисунки) отстъпи се задават според личните предпочитания на дизайнера.

Задаването на външни отстъпи около изображението се осъществява с помощта на CSS свойството margin и неговите производни за горния, десния, долния и левия край: margin - top, margin - right, margin - bottom, margin - left (виж раздел “Външни отстъпи” в глава 1). Обикновено стойностите се задават в пиксели. Алтернатива на това CSS свойство са атрибутите на HTML тага :

- hspace — задава размера на полето отляво и отдясно на изображението;
- vspace — задава размера на полето отгоре и отдолу на изображението.

```
<!-- <CD>\Samouchitel\image _ margin.html -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
<TITLE>Image margin</TITLE> <STYLE type="text/css">
<! —
```



```
TABLE {
border: 1px solid gray;
}
//—>
</STYLE>
```

5.10. Управление на таблици

В тази глава ще разгледаме създаването на таблици посредством HTML -тага <TABLE>. Той е типичен представител от групата на блоковите тагове, т.е. за неговите елементи се отделя цял ред от браузъра. Тагът <TABLE> се разглежда като контейнер за редове (дефинирани с HTML - таг <TR>), които на свой ред се явяват контейнери за клетките на таблицата (дефинирани с HTML - тагове <TH> и <TD>).

Задаване на размера на таблицата

Както всички други маркери (HTML тагове), така и тагът <TABLE> има правоъгълна структура на блока. От това следва, че задаването на хоризонталните и вертикалните размери на таблицата се осъществява с използването съответно на CSS свойствата width и height. Като алтернатива на CSS свойството width може да послужи едноименният атрибут на HTML тага <TABLE>, предназначен за задаване на ширина на таблицата. Тагът <TABLE> не притежава атрибут, позволяващ задаването на височина на таблицата.

```
<!-- <CD>\Samouchitel\table_size.html -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
<HTML>
<HEAD>
<TITLE>Table size</TITLE>
<STYLE type="text/css">
<!--
TABLE, TD {
border: 2px solid gray;
}
//—>
</STYLE>
</HEAD>
<BODY>
<TABLE style="width: 300px; height: 100px;" <TR><TD>Minsk</TD><TD>1.
900.000</TD></TR> <TR><TD>Kiev</TD><TD>2.700.000</TD></TR>
</TABLE>
<BR>
<TABLE width="300">
<TR><TD>Minsk</TD><TD>1. 900.000</TD></TR>
<TR><TD>Kiev</TD><TD>2.700.000</TD></TR>
</TABLE>
</BODY>
</HTML>
```

Хоризонтално и вертикално подравняване на съдържанието на клетките

Хоризонталното подравняване на текста в клетките на таблицата има преди всичко декоративен смисъл. Като изключение може да се посочи използването на таблици като

скелет на уебстраницата. Такъв тип скелет обикновено се нарича табличен - от името на HTML тага <TABLE> (съществува също и “авно” оформление — от името на HTML тага <Div>).

Центрираното хоризонтално подравняване обикновено се използва за заглавия на колоните (това е стойност по подразбиране за клетки, дефинирани с HTML тага <TH>), дясното подравняване се използва за числа, а подравняването по левия край (стойността по подразбиране) - за текст. Хоризонталното подравняване се задава с CSS свойството `text-align`, което може да приема следните стойности:

- `left` — подравняване по левия край;
- `center` — центрирано;
- `right` — подравняване по десния край.

Вертикалното подравняване на съдържанието на клетки, дефинирани с HTML таговете <TH> и <TD>, по подразбиране е центрирано. Това често е неудобно - например при съставяне на отчети е необходимо текстът да е подравнен по горната граница на клетката. Вертикалното подравняване се задава с помощта на CSS свойството `vertical-align`, най-често със следните стойности:

- `top` — подравняване по долния край;
- `middle` — центрирано;
- `bottom` — подравняване по горната граница.

Като алтернатива на това CSS свойство понякога се използват атрибутите на HTML таговете <TH> и <TD>:

5.11. Предимства и недостатъци на CSS

CSS има доста предимства! CSS не е създаден за други цели и не е предназначен за далечното бъдеще. Използван добре в момента, CSS предлага практически предимства, включително:

- 1) Пести честотната лента на потребителя, като ускорява скоростта на зареждане на страниците, особено при dial-up връзки.
- 2) Намалява натоварването върху сървъра и нужната честотна лента на собственика, като по този начин спестява пари.
- 3) Намалява времето за проектиране и разработка. Създаването на сайта отнема само няколко часа. Спестяване има само от времето, прекарано в разработка, разбира се: създаването на съдържание и художественото оформление отнемат също толкова време, колкото обикновено.
- 4) Намалява времето за актуализация и поддръжка:
 - Хората, създаващи съдържанието, вече не трябва да се тревожат за сложни таблици, тагове за шрифт и други остарели компоненти на разположението, които могат да се нарушат, когато текста бъде актуализиран. Тъй като няма никакви (или почти никакви) такива елементи, няма какво да се наруши.
 - Дизайнерите вече не трябва да се тревожат за възможността клиентите да развалят сайта.
 - Глобалните промени могат да бъдат реализирани за минути. Текстът е прекалено тъмен? Създавате едно или две правила в CSS файла и целият сайт моментално ще отрази промяната.
- 5) Увеличава съвместимостта, като се придържа към препоръките (уебстандартите) на W3C.
- 6) Увеличава достъпността, като премахва някои, повечето или всички презентационни елементи от маркирането.

- 7) Предоставя възможности за потребители, които не могат да ползват мишка – това са хора с увредени двигателни функции; те могат да ползват браузъри, които поддържат JavaScript, но има вероятност да не могат да щракват с мишка или други движения с мишка. Ето го и кодът:

```
<input type="button" onclick="setActiveStyleSheet('default')>;  
return false;"onkeypress="setActiveStyleSheet('default');  
return false;" /> Това приканва всеки да разгледа сайта, не да ги гони.
```

Ето един пример за CSS код за навигационна лента:

```
/*Navigation bar components*/  
table#nav {  
border-bottom: 1px solid #000;  
border-left: 1px solid#000;  
}  
table#nav td {  
font: 11px verdana, arial, sans-serif;  
text-align:center;  
border-right: 1px solid#000;  
border-top: 1px solid#000;  
}  
table#nav td a {  
font-weight:normal;  
text-decoration: none;  
display: block;  
margin:0;  
padding: 0;  
}  
#nav td a:link, #nav td a:visited {  
background: transparent url (/images/bgpat.gif) repeat;  
display: block;  
margin: 0;  
width: 100px;  
line – height: 25px;  
}  
#nav td a:hover {  
color:#f60;  
background: white url (/images/nopat.gif) repeat;  
}
```

```

td#home a:link img, td#home a:visited img {
color: #c30;

background: transparent url (/images/bg/pat.gif) repeat;
width:400px;
height: 75px;
}

td#home a:hover img {
color: #f60;

background: white url (/images/nopat.gif) repeat;
width: 400px;
height: 75px;
}

```

CSS има и недостатъци:

Информацията в листовите не се поддържа от браузърите преди Microsoft Internet Explorer или Netscape Navigator. Това обаче не е толкова дразнещо, колкото непостоянството на поддръжка от браузърите, които твърдят, че наистина поддържат CSS.

- Ако използвате цвят, за да подчертаете някаква информация, подсилете я и с някакви други методи, например с удебеляване или подчертаване на препратките. Ако сте изключили подчертаването чрез CSS, можете да направите препратките по-удебелени от нормалния текст. В този случай избягвайте да удебелявате неподчертан текст, за да не се чудят далтонистите кой текст е препратка и кой е просто удебелен.
- Не вярвайте само на един браузър – пишете валиден CSS код и го тествайте в различни браузъри. Неправилния CSS код може да направи страницата нечетима.
- Внимавайте, когато задавате размери на текста. Избягвайте да задавате размера на текст в пиксели, защото не може да се оразмерява ...
- Не вярвайте на резултатите от тестове за проверка на достъпност, т. е. не допускайте, че CSS кодът ви е наред, само защото страниците преминават успешно теста за достъпност на Bobby. Страница, която използва нечетим шрифт с големина 7 px, спокойно може да премине теста.

6. HTML и XHTML

HTML отбелязва значителен прогрес от времето, когато е приет за първи път през 1992 г. След първоначалния бурен интерес към HTML и Уеб назрява необходимостта от организация, която да създаде препоръчителни правила и стандарти, гарантиращи отворената същност на Уеб. За да се отговори на тази потребност, през 1994 г. в Масачузетския технологичен институт е създаден консорциумът W3C.

Под ръководството на Тип Бърнъс - Лий той определя стандартите в HTML и предоставя отворен, необвързан форум, чрез който представителите на индустрията и научните среди да допринасят за развитието на тази нова медия. W3C е в незавидната роля да стои на гребена на вълната на една много динамична

индустрия. Различните комисии, от които се състои консорциумът, целят да разширят и наложат определени стандарти за множеството нововъзникващи уебтехнологии. Това включва XHTML (Extensible Hypertext Markup Language), xml (Extensible Markup Language), CSS (Cascading Style Sheets) и други описателни стилове и езици. Има и ред други съпътстващи HTML технологии. На сайта на www.w3c.org може да откриете повече за HTML, XHTML, XML, CSS, както и други справки за дефиницията на определени елементи, да тествате валидността на кода си или да се информирате за най-новите уебтехнологии.

XHTML (съкращение от eXtensible HyperText Markup Language) е език от семейството на XML - базираните езици, който наследява HTML - езикът, на който най-често се пишат уебстраниците в Интернет.

Докато езикът HTML е приложение, базирано на SGML (Standard Generalized Markup Language), много гъвкав markup language framework, то XHTML е базиран изцяло на XML - езикът, който наследява SGML. За да бъдат разчитани от парсера, XML - базираните документи (включително и XHTML) трябва да бъдат с добро оформление и без грешки, когато се проверяват с валидатора на W3C - <http://validator.w3.org>

XHTML 1.0 излиза на 20 януари 2000 година, създаден от работната група на World Wide Web консорциума (W3C). Официално XHTML 1.1 е пуснат на 31 май 2001 година. XHTML5 в разработка от септември 2009 година насам и ще бъде част от спецификацията на HTML5.

Езикът е създаден, за да направи HTML по-гъвкав език с по-големи възможности. Друга причина за създаването на XHTML е оперативната съвместимост на различните приложения и формати от данни. Докато HTML4 е базиран изцяло на SGML, XHTML е написан изцяло като XML - базиран език.

- За разлика от HTML, XHTML не позволява пропускането на затварящите тагове на някои елементи като paragraph, heading и други.
- В HTML целия код е чувствителен към регистъра на малките и главни букви (case - insensitive). В XML само имената на таговете и атрибутите са case - insensitive, но не е позволено тагове да се наименуват или да започват с главни букви.
- HTML позволява скриването на някои от атрибутите и писането само на техните стойности (Пример: <option selected>). В XML - базираните езици този код би изглеждал така: <option selected="selected">..
- За разлика от XML, в HTML някои елементи като <html>, <head>, или <body> могат да бъдат пропуснати, а след това автоматично добавени от парсера.
- В старите версии на HTML е позволено стойностите да бъдат задавани без кавички. Пример: <body lang=en>

Пример:

This is an example of a form for Japanese address input.

```

Family name: <input name="name" inputmode="kanji" />
(in kana): <input name="namekana" inputmode="katakana" />
Given name: <input name="Given" inputmode="kanji" />
(in kana): <input name="Givenkana" inputmode="katakana" />
Postal code: <input name="zip" inputmode="latin digits" />
Address: <input name="address" inputmode="kanji" />
(in kana): <input name="addresskana" inputmode="katakana" />
Email: <input name="email" inputmode="latin lowerCase" />
Telephone: <input name="tel" inputmode="latin digits" />
Comments: <textarea name="comments" inputmode="user predictOn" />

<input name="doit" type="submit" value="OK" />

```

W3C е организация в Интернет, която се опитва да въведе ред в бъркотията от браузъри и версии – W3C /World wide web Consortium, www.w3c.org и която от 1992 г. е разработила три официални HTML стандарти/. HTML 2.0 е първия официален стандарт, създаден по времето, когато Интернет едва започва да придобива популярност. Командите, описани в него, предоставят на разработчиците твърде скромни възможности за развитие на визията и това довежда до необходимост за неговото преработване. HTML е типичен структурен език, създаден с идеята да описва организацията и типа на данните. Налични са няколко версии на HTML, които остават само на чернова, не отговарят на изискванията на новата медия. Чак HTML 4.0.1 дава възможност за по-пълен контрол на изгледа на страниците /CSS/, които намират реализация. Java Script 1.2 създава условия за изграждане на много по-динамични уеб страници /включително движение на отделни HTML елементи/, от което се поражда названието DHTML. Следва разработката на много структурни езици – XML, VRML, SMIL и др.

7. Употреба на Flash в уеб страниците – flash филмите могат да се поставят на уеб страници или могат да се използват като уеб страници. Още повече, че с усъвършенстваните възможности за писане на скриптове, въведени във Flash 5, приложенията на Flash филми могат да бъдат ограничени единствено от въображението Ви. Някои възможни приложения включват:

- Анимации на страници
- Интерактивни навигационни ленти
- Анимирани рекламни плакати
- Интерактивни и мащабируеми карти
- Интерактивни форми
- Цели интерфейси на уеб сайтове, които да заместят традиционните HTML страници
- Анимационни филмчета
- Музикални видео филми
- „Джубокс“ – интерфейс за изпълнение на MP3 файлове

Въпреки че Flash предоставя значителен брой подобрения спрямо това, което можем да направим само с HTML, съществуват и известен брой недостатъци. Нека разгледаме обстойно двете страни на монетата при употребата на Flash на Вашите страници.

Преимущества

Голяма част от аспектите на файловия формат Flash го правят подходящ за добавяне на интерактивно съдържание към веб страници:

- Малък размер на файловете. Векторния формат на Flash се асоциира с малки файлове и бързото им сваляне от сървъра.
- Мащабируем е. Изображенията и анимациите на Flash могат да бъдат преоразмерени без загуба на детайли, като по този начин се улеснява запълването на целия прозорец на брауъра с Flash интерфейса, без да се увеличава файловият размер. Flash може да се използва за създаването на статични изображения, например карти, където увеличаването на мащаба е желателно.
- Високо качество на изображенията. Заглаждането в реално време изглажда краищата на графиките и текста, независимо какъв е размерът на визуализация. Потребителите могат да увеличат мащаба на векторната графика без загуба на качеството на изображението.
- Използва поточна технология. Flash файловете започват да се изпълняват бързо и продължават да се изпълняват, докато протича трансферът на данни, така че те могат да бъдат псевдопоточни от даден HTTP сървър.
- Използва вграден звук. Flash е един добър начин за предоставяне на фонен звук и насочени към потребителя звукови ефекти на веб сайта.
- Flash е добре поддържан формат. Възпроизвеждащото устройство Flash, което е нужно за изпълнението на Flash файловете, работи на Windows, Mac OS платформи. То се инсталира директно на Windows машини.
- Поддръжка на скриптове. Flash използва езика за писане на скриптове ActionScript за контролиране поведението на съдържанието.
- Той е свободно достъпен формат. Macromedia направи файловия формат Flash общодостъпен, което означава, че другите разработчици могат да вградят поддръжка на Flash в своите приложения. Едно от първите такива приложения е LiveMotion на Adobe (инструмент за създаване на интерактивни бутони и анимирани обекти), което съхранява файловете си във Flash формат, за да могат в последствие да бъдат изпълнени от възпроизвеждащото устройство на Flash.

Недостатъци

- Нужен е plug-in. Стандартните Flash файлове изискват възпроизвеждащо устройство – или Shockwave Essentials, или Flash – за да бъдат инсталирани на потребителската машина. Въпреки че на пръв поглед това може да изглежда като малко препятствие, особено след като възпроизвеждащите устройства Shockwave и Flash са едни от най-широко разпространените и глобални plug-in, думите „задължителен plug-in“, са абсолютно достатъчни, за да накарат клиента да каже „в никакъв случай“, без дори да обмисли предложението.
- Съдържанието се губи при неграфични брауъри. Използването на Flash филми за заглавията и навигацията се изправя пред същите проблеми, както и употребата на статични графики на мястото на текст. Хората, които не могат да визуализират вашите Flash анимации, няма да могат да прочетат вашето съдържание.
- Визуализацията винаги започва от началната страница на филма. Потребителите не могат да се придвижат към определена част или

сцена от Flash филма. Връзките отвеждат потребителите само до първата страница от Flash филма.

- Поддръжката на Unix е ограничена. Въпреки, че съществува Netscape plug-in for Linux Red Hat 6 и по-високи версии и за Solaris, другите потребители на Unix нямат възможност да визуализират Flash филми. Едно от решенията на този проблем е Flash Player Java Editor. Не съществува версия за Unix на Flash инструмента за обработка.
- Отпечатването може да се окаже проблематично. Могат да се появят проблеми при разпечатването на Flash съдържание, най-вече от Netscape браузъри на Mac.
- Нужен е скъп обработващ софтуер. По настоящем се нуждаете от Flash софтуер на Macromedia, за да създавате Flash файлове.

Мощни Flash инструменти

Можете да създавате доста модерни конфигурации с Flash направо в прозореца на програмата посредством вградените му характеристики. За голяма част от потребителите това е абсолютно достатъчно. Но Flash не се използва само за анимация и звукови ефекти. С помощта на усъвършенстваните скриптові характеристики или допълнителните софтуери, Flash филмите могат да бъдат програмирани така, че да изпълняват интерактивни функции и дори да служат като край за динамичното генерирано съдържание.

Основи на Flash интерфейса

Обикновено Flash филмите се добавят към веб страници посредством комбинация от инструкциите за форматиране <object> и <embed> заедно с техните параметри и атрибути за контролиране на визуализацията и плейбека. И двете инструкции за форматиране се използват с цел да се избегнат несъвместимостта на Internet Explorer и Netscape Navigator. Internet Explorer for Windows използва инструкцията за форматиране <object>, която автоматично му позволява да изтегли ActiveX контроли за изпълнението на Flash медията.

Добавяне на Flash към HTML документ

За да кодирате страницата си по начин, достъпен за най-голям брой потребители, използвайте комбинация от инструкциите за форматиране <embed> и <object>. След малко ще прочетете и обясненията към всяка от двете инструкции. Забележете, че технологиите променят бързо и Macromedia коригира своите инструкции от време на време. За актуализация се обърнете към страниците за поддръжка на Macromedia /<http://www.macromedia.com/support/director/internet.html/>.

Инструкцията за форматиране <embed>:

Стандартна инструкция за форматиране <embed> изглежда ето така:

```
<EMBED SRC="PATH/FILE.SWF" WIDTH=X HEIGHT=X  
PLUGINSOURCE="http://www.macromedia.com/shockwave/download/  
Index.cgi?P1_Prod_Version=ShockwaveFlash">  
</EMBED>
```

Стойностите на width и height са задължителни и определят размера на изображението или филма в пиксели. Забележете, че също така може да зададете и процентни

стойности (съответстващи на процентната част от прозореца на браузъра, която филмът ще запълни).

Инструкцията за форматиране <object>

Инструкцията за форматиране <object> казва на Internet Explorer да свали и инсталира определено ActiveX възпроизвеждащо устройство за Flash файлове. Ето пример на основна инструкция за форматиране <object>:

```
<OBJECT CLASSID="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"  
CODEBASE=http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=5,0,0,0 WIDTH=300 HEIGHT=150>  
<PARAM NAME="MOVIE" VALUE="movienamе.swf">  
</OBJECT>
```

Параметърът classid определя конкретна Flash ActiveX контрола, а codebase предоставя на браузъра нейното местоположение, от което може да бъде свалена. Стойността на атрибута classid трябва да бъде изписана във вашият HTML файл, точно както е показано по-горе; тя се прилага към всички версии на Flash. Вижда се, че атрибутът codebase сочи към възпроизвеждащото устройство за версия 5. Възпроизвеждащите устройства за другите версии и подиздания могат да бъдат активирани посредством този метод само чрез промяната на номера на версията. Атрибутите width, height са задължителни. Първият параметър (<param>) установява името и местоположението на вашия Shockwave Flash файл. Хубава книга за Flash Ви препоръчвам – Flash 5 от Robert Reinhardt, Jon Warren Lentz (*Hungry Minds, 2001*).

8. PHP – вграждане в HTML

КАКВО Е PHP?

PHP (www.php.net) е пригоден за различни цели скриптов език с отворен код, който е идеално приспособен към уеб разработка и може да бъде вляган в XHTML. Синтаксисът му е заимстван от C, Java, Perl и е сравнително лесно за научаване. PHP (което доста объркващо означава Предпроцесор за хипертекст), притежава много възможности, но е станал изключително популярен заради една от тях: когато се използва в комбинация с база данни на MySQL (www.mysql.com), PHP позволява на дизайнерите и разработчиците лесно да създават динамични сайтове и да изграждат уеб приложения. PHP е проект на Apache Software Foundation (www.apache.org) и е безплатен – факт, на който дължи част от популярността си. Разработчиците на отворен код и независимите уеб дизайнери са очаровани от езика и непрекъснато създават PHP-базирани приложения, които предоставят безплатно на своите колеги.

Първото, с което трябва да се запознаете, е взаимовръзката между PHP и HTML.

```
<HTML>  
<HEAD>Sample PHP Scrip</HEAD>  
<BODY>  
The following prints "Hello, World":  
<?php print "Hello, World";  
>
```

```
</BODY>
```

```
</HTML>
```

От примера се вижда, че PHP кодът е вграден в HTML кода. Когато интерпретаторът на PHP достигне отварящия таг `<?php`, той изпълнява кода до маркера `?>`. След това PHP заменя този PHP код с резултата от изпълнението му (ако има такъв), а останалият текст (HTML текста) се пропуска непроменен до уеб клиента. Така споменатият скрипт ще даде следния резултат:

```
<HTML>
```

```
<HEAD>Sample PHP Script</HEAD><BODY>
```

The following prints "Hello, World":

Hello, World

```
</BODY>
```

```
</HTML>
```

Съвет: Ако разрешите INI опцията `short_open_tags`, можете да използвате и съкращения отварящ таг `<?;`; това обаче не е препоръчително, тъй като може да възникне конфликт с някои XML тагове. Затова опцията е изключена по подразбиране.

Коментари в PHP

Има три начина за писане на коментари:

В стил C

```
/* Това е коментар в стил C,
```

```
*който може да заема няколко
```

```
*реда, до достигане на затварящите символи*/
```

В стил C++

```
//Това е коментар в стил C++, който завършва в края на реда
```

В стил обвивка

```
#Това е коментар в стил обвивка. Завършва в края на реда
```

Интерфейси в PHP

Наследяването на класове позволява описанието на взаимоотношение родител-наследник между класове. Например може да имате базов клас `Shape` (фигура), от който произлизат `Square` (квадрат) и `Circle` (окръжност). Често обаче са налага да добавяте допълнителни „интерфейси“ към някои класове, което в общи линии означава допълнителни договори, с които трябва да се съобразява класа. В C++ това се постига с множество наследяване от два класа. В PHP като алтернатива на множественото наследяване са избрани интерфейсите, които позволяват сключването на допълнителни договори, които трябва да бъдат спазвани от класа. Интерфейс се декларира аналогично на клас, но съдържа само прототипи на функции (без реализация) и константи. Всеки клас, който реализира този интерфейс, автоматично придобива дефинираните константи

на интерфейса и, като реализиращ клас трябва да предоставя дефиниция на функциите-прототипи от интерфейса, които са абстрактни методи (освен ако декларирате реализация клас като абстрактен).

За да реализирате интерфейс, използвайте следния синтаксис:

```
Class A implements B, C,... {  
}
```

Класовете, които реализират даден интерфейс, имат взаимоотношение instanceof (“е от тип”) с интерфейса; например ако клас А реализира myInterface, следният код ще отпечата “Sobj is-A myInterface”:

```
Sobj=new A();  
if (Sobj instanceof myInterface){  
print ‘Sobj is-A myInterface’;  
}
```

Следващият пример дефинира интерфейс, наречен Loggable, който може да бъде реализиран от класовете, за да дефинират каква информация ще бъде вписвана в дневник от функцията MyLog(). Обектите от класове, които не реализират този интерфейс, но се подадат на функцията MyLog(), ще доведат до показване на съобщение за грешка:

```
interface Loggable {  
function logString();  
}  
class person implements Loggable{  
private Sname, Saddress, SidNumber, Sage;  
function logString(){  
}  
}  
class Product implements Loggable{  
private Sname, Sprice, SexpiryDate;  
function logString(){  
return “classProduct: name, price=Sthis->price\n”;  
}  
}  
function MyLog(Sobj){  
if (Sobj instanceof Loggable){  
print Sobj->logString();  
}  
}  
Sperson=new Person();  
//...
```

Sproduct=new Product();

MyLog(Sperson);

MyLog(Sproduct);

Забележка: Интерфейсите винаги са public; затова не можете да поставяте модификатори на достъпа пред прототипите на методите в декларацията на интерфейса.

Забележка: Не можете да реализирате няколко интерфейса, които си пречат, (например интерфейси, които дефинират еднакви константи или методи).

Шаблони за дизайн

И така, какво точно означава, че един език е обектно-ориентиран (ОО)?

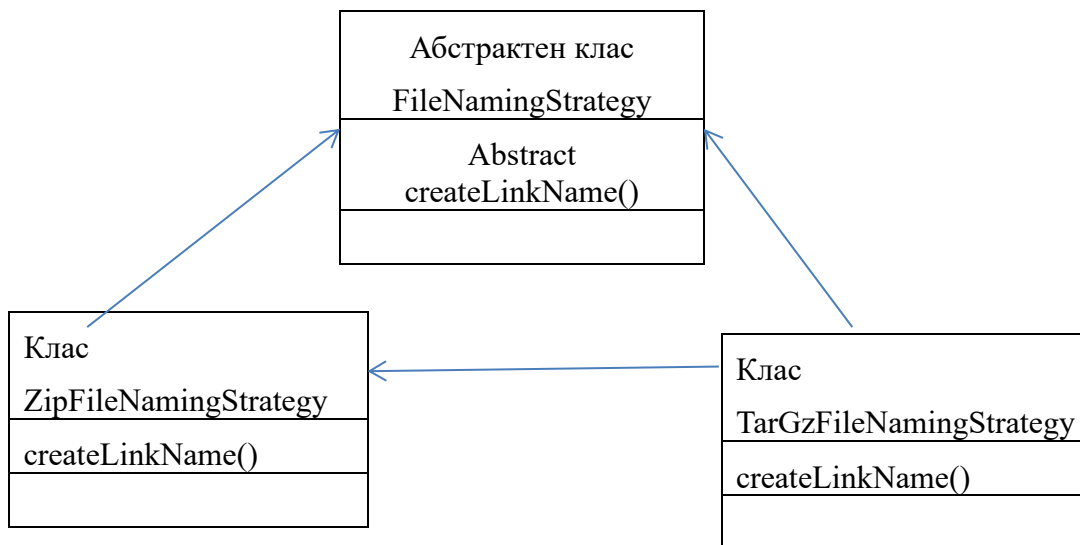
Някои хора вярват, че всеки език, който разполага с обекти, капсулиращи данни и методи, може да се счита за обектно-ориентиран. Други биха добавили към определението полиморфизма чрез наследяване и добавянето на модификатори на достъпа към дефинициите. Обектно-ориентираните приложения не се свеждат само до използване на точно определен синтаксис и възможности на езика, а е по-скоро начин на мислене. Въпреки че и на функционалните езици като С се пишат някои изключително професионални приложения (РНР например), хората, разработващи с обектно-ориентирани езици, отдават на дизайна на софтуера доста по-голямо значение. Една от причините за това вероятно е, че обектно-ориентираните езици обикновено предлагат възможности, които подпомагат фазата на проектиране, но главната причина е по-скоро културна, тъй като ОО общността винаги е придавала голямо значение на добрия дизайн.

Когато проектирате софтуер, често се срещат повтарящи се шаблони. Някои от тях са известни на общностите за проектиране на софтуер и за тях има готови приети решения. Тези повтарящи се проблеми се наричат **шаблони за дизайн**. Предимството от познаването и използването на тези шаблони е не само в спестяването на време от нуждата да преоткривате колелото, но и в предоставянето на разработчиците на общ език на софтуерния дизайн. Има доста шаблони, но ние ще разгледаме само един: шаблон **Стратегия (Strategy)**.

Шаблонът стратегия обикновено се използва, когато един алгоритъм трябва да може да се заменя с други негови вариации. Ако например имате код, който създава изображения, ще има случаи, в които създавате JPEG файлове, а в други-Gif файлове.

Шаблонът Стратегия обикновено се реализира чрез деклариране на абстрактен базов клас с метод за алгоритъма, който след това се реализира наследяване на конкретни класове. В кода се решава коя конкретна стратегия е подходяща; след това тя се реализира и използва при нужда. По-долу ще видите как един сървър за сваляне може да използва различна стратегия за сваляне на файл, според типа на уеб клиента. При създаване на HTML кода с препратките за сваляне, той ще създаде препратки или към tar.gz файлове, или към zip файлове, според посочената операционна система в идентификацията на брауъра. Разбира се това означава, че на сървъра трябва да бъдат записани и двата вида файлове. За по-просто ще приемем, ако в S_SERVER[HTTP_USER_AGENT] присъства думата "Win", значи клиентът използва Windows, и се нуждае от препратки към zip файлове;

В този пример бихме имали две стратегии: Стратегия tar.gz и стратегия zip. Получава се следната йерархия на стратегиите (виж фиг. 16).



Фиг. 17 Йерархия на стратегиите

От следния код ще придобиете идея за начина на употреба на този шаблон **Стратегия**:

```

abstract class FileNamingStrategy {
  abstract function createLinkName (filename);
}
class ZipFileNamingStrategy extends FileNamingStrategy {
  function createLinkName (Sfilename)
  {
    return http://downloads.foo.bar/Sfilename.zip;
  }
}
class TarGzFileNamingStrategy extends FileNamingStrategy {
  function createLinkName (Sfilename){
    return http://downloads.foo.bar/Sfilename.tar.gz;
  }
}
if (strstr(S_SERVER["HTTP_USER_AGENT"], "Win")){
  sfileNamingObj=new ZipFileNamingStrategy();
} else{
  SfileNamingObj=new TarGzFileNamingStrategy();
}
Scalc_filename=SfileNamingObj->createLinkName("Calc101");
Sstat_filename=SfileNamingObj->createLinkName("Stat2000");
  
```

```
print<<<EOF
<h1>The following is a list of great downloads<</h1>
<br>
<a href="Scalc_filename">A great calculator</a><br>
<a href="Sstat_filename">The best statistics application</a><br>
<br>
EOF;
```

При достъп до скрипта от система с Windows ще се създаде следния HTML код:

```
<h1>The following is a list of great downloads<</h1>
<br>
<a href="http://downloads.foo.bar/Calc101.zip">A great calculator</a><br>
<a href="http://downloads.foo.bar/Stat2000.zip">The beststatistics
Application</a><br>
<br>
```

PHP е скриптов език върху сървърната (обслужваща) страна език с отворен код, който е проектиран за уеб програмиране и е широко използван за създаване на сървърни приложения и динамично уеб-съдържание. Автор на езика е канадецът от датски произход Размус Лердорф. PHP е рекурсивен акроним от PHP: Hypertext Preprocessor (като в самото начало има значение, дадено от създателите му, на Personal Home Page).

PHP може да работи с повечето модерни бази данни – MySQL, PostgreSQL, Microsoft SQL Server, Oracle, SQLite и д.р.

На официалния сайт на PHP се намира обширна библиотека с информация за езика и модулите му, която може да се използва както за основно запознаване с езика, така и като справочник по време на работата с него.

PHP действа главно като филтър, който взема съдържанието на файл и изпълнява специални PHP инструкции, описани във файла. PHP скриптът има начален и краен таг, между които е разположено съдържанието. Отварящият таг е `<?php`, а затварящия е `?>`. Цялата инструкция трябва да завършва с `;`.

Тук има пример за код, който визуализира „Здравей, свят“:

```
<?php
echo 'Здравей, свят!'; /* Hello world */
?>
```

пример за свързване към база данни и изпълняване на проста заявка към нея:

```
<?php
```

```

session_start();
$dbUser = "root";
$dbPassword = "password";
$dbHost = "localhost";
$conn = mysql_connect($dbHost, $dbUser, $dbPassword) or die("Error in MySQL: " +
mysql_error());
mysql_select_db("database", $conn);
$rs = mysql_query("SELECT UserId, Username, UserPassword FROM Users WHERE
`Username` = " . $uname . " AND UserPassword = " . $upass . " LIMIT 1", $conn);
if(mysql_num_rows($rs) == 1)
    $_SESSION['logged'] = true;
mysql_close($conn);
?>

```

Предимство на PHP е, че позволява командите му да бъдат вградени в HTML документите, като крайните потребители не могат да видят изходния код написан на PHP, който преработва всичко извън рамките му:

```

<?php
// PHP код
?>
обикновен HTML код
<?
// още PHP код
?>

```

Предимства на езика: PHP работи на множество операционни системи (Unix, Linux, Windows, BSD, Mac OS X) и множество уеб сървъри – Apache, lighttpd, IIS

- PHP е лесен за разработване
- PHP е безплатен и се разпространява под лиценза на BSD
- PHP може да бъде лесно модифициран и адаптиран към нуждите на прилагашата го организация
- PHP е създаден и пригоден за разработката на уеб приложения
- PHP не изисква особени умения от разработчици работили на структурни езици – езикът е с прост и интуитивен синтаксис за такива разработчици
- PHP е широко разпространен поради простотата си. Има много програмисти, което води до по-евтино платен персонал във фирмите, по ниска цена на приложенията за клиентите и още по-голяма използваемост. Поддръжката за PHP разработчици е гарантирана от множеството форуми и приложения на общността.

По аналогия с Perl към стандартните класове на PHP могат да бъдат писани и много допълнителни модули

- PHP поддържа следните системи за управление на бази от данни:

IBM DB2 – formix – Ingres, Microsoft SQL Server (MS SQL),
mSQL, MySQL, Oracle, PostgreSQL, Sybase

- PHP поддържа и ODBC

Недостатъци:

- В по-старите версии на езика по подразбиране съществува функционалност, чрез която крайните потребители могат да подават направо променливи към PHP скрипта (`register_globals`). Настройката по подразбиране е премахната от версия 4.2.0, но съществуват все още много стари приложения, които предполагат употребата на тази функция.
- PHP не поддържа напълно Unicode. Пълната поддръжка се предвиждаше в PHP 6, но не беше осъществена и се премина към версия PHP 7.
- PHP не поддържа именни пространства до версия 5.3 на езика – т.е. всички функции се намират в едно именно пространство.
- Съществуват много функции на библиотеки с идентична функционалност.
- Няма изградена конвенция за именуване и подреждане на аргументите на вградените функции в PHP, което прави необходимо постоянно консултиране с документацията при работа с PHP.
- PHP няма вградена поддръжка на нишки (threads), което прави обработката на големи масиви от данни много трудна: страницата не се зарежда преди целият масив от данни да е преработен и ако потребителят прекъсне зареждането ѝ (и уеб сървърът разрешава прекъсване на заявката), прекъсва и обработката на данните.
- Обръщанията към неинициализирани променливи и елементи на масив са разрешени в PHP, което прави възможно допускането на трудни за откриване правописни грешки.
- PHP използва т. нар. „magic quotes“, които са създадени с цел да предпазват начинаещите програмисти, но объркват по-напредналите.
- Понякога се среща неконсистентност между поведението на PHP в различните версии, което предизвиква необходимостта от пренаписване на скриптовете, писани за стари версии на езика.

Най-честата употреба на PHP е за изграждане на уеб сайтове. PHP прави уеб приложенията динамични, даващи възможност на потребителите да взаимодействат със сайта. Уеб приложението събира информация от потребителя, използвайки HTML формуляри, и след това я обработва. Част от информацията, събрана от потребителите и съхранявана на уеб сайта, е поверителна, което прави сигурността основен проблем. PHP предоставя средства, даващи възможност за събиране на информация от потребителите, без да се застрашава сигурността. На вас остава да разработите пълно приложение, използвайки частите, предоставяни от PHP. Настоящата глава описва как се използва функционалността на PHP за изграждане на динамично уеб приложение.

Вграждане в HTML

Разбира се, не е наложително PHP да се вгражда в HTML файл; няма никакъв проблем да се създаде PHP файл, който не съдържа никакъв HTML код. Когато обаче се създава уеб приложение, PHP и HTML често се използват заедно в общи файлове. PHP е разработен предимно за използване в уеб пространството, за вграждане в HTML

файлове като език за създаване на шаблони. Когато в даден файл се включи PHP код, той трябва да носи разширение на PHP (разширение, което сигнализира на уеб сървъра да очаква PHP код); Обикновено това е php, но при конфигурирането на уеб сървъра има възможност да се дават и други разширения, като например html или php5.

Следва пример за PHP код, вграден в HTML код:

```
<html>
<head><title>Example 1</title></head>
<body>
<?php
/* Ако е първи април, показваме цитат */
if (date('md')== '0401') {
echo 'a bookstore is one of the only pieces of evidence we have'.
'that people are still thinking:<i>Jerry Seinfeld</i>';
} else {
echo 'Good morning!';
}
?>
</body>
</html>
```

С <?php започва PHP секцията, вградена в HTML кода; редът ?> приключва PHP секцията. Обърнете внимание, че кодът използва echo, за да изпрати резултата. Когато текстът е така прост, конструкциите echo са приливи. Когато обаче се налага да отпечатвате текстови низове, съдържащи апострофи или кавички, кодът става по-сложен. Ако отпечатваният текст в примера беше конструкция с препратка (например), примерът нямаше да работи правилно, тъй като апострофите в текста щяха да влязат в противоречие с апострофите, ограждащи текстови низ. В такъв случай, може да прекратите PHP секцията точно преди извеждането на текста и да започнете нова, точно преди PHP кода, който завършва блока и започва блока else:

```
<html>
<head><title>Example 2</title></head>
<body>
<?php>
/* Ако е първи април, показваме цитат */
if (date('md')== '0401') {
?>
A bookstore is one of the only pieces of evidence we have that people are still thinking.
<i>Jerry Seinfeld</i>
<?php
} else {
```

```
?>
Good morning!
<?php
}
?>
</body>
</html>
```

Но този начин за писане на код е доста разхвърлян. С него се нарушава един от основните принципи на програмирането: „Отделяй логиката от съдържанието“.

Качване на файлове

Все още не сме разгледали един вид въвеждане на данни – качването на файлове PHP предоставя възможност за качване на файлове, като например изображения или други прилежащи материали. Тъй като браузърът трябва да свърши малко повече работа от това просто да изпрати POST със съответните данни, за качването на файлове ще трябва да използвате специално изработен формуляр. Ето един пример за такъв:

```
<form enctype="multipart/form-data" action="handle_img.php">
method="post">
<input type="hidden" name="MAX_FILE_SIZE" value="16000"/>
Send thi file:<input name="book_image" type="file"/><br/>
<input type="submit" value="Upload"/>
</form>
```

Различията между формулярите за качване на файлове и нормалните такива са маркирани в разпечатката. Първо, атрибутът enctype, включен в тага на формуляра, инструктира браузъра да изпрати различен вид заявка POST.

Всъщност това е нормална заявка POST, но тялото, съдържащо кодираните файлове (и другите полета на формуляра) е напълно различно. Вместо елементарен синтаксис от типа поле=променлива: поле2=променлива2, в тялото се изпраща нещо наподобяващо електронно писмо с „текст и HTML“, като всяка негова част е поле от формуляра.

Самото поле за качване на файл е от тип file и се изобразява като поле за въвеждане и бутон за браузване, който дава възможност на потребителя да търси файла във файловата система. Текстът върху бутона може да бъде променен, така че обикновено е локализиран. (Английската версия на Mozilla използва “Browse”, холандската версия на IE използва “Bladeren” и т. н.).

Скритото поле за въвеждане изпраща MAX_FILE_SIZE на браузъра, което определя максималния допустим размер на зареждания файл. Повечето браузъри обаче игнорират това допълнително поле, така че от Вас зависи обработващия скрипт да приеме или отхвърли този файл.

Обработка на грешки

Броят на грешките в приложенията може да бъде намален чрез използването на добри навици при програмиране. Съществуват обаче множество фактори, които причиняват грешки и които са извън контрола на създателя на скрипта. Мрежови проблеми, пълни

твърди дискове, хардуерни аварии, програмни грешки в други PHP компоненти и програми, с които си взаимодейства приложението, могат да причинят проблеми, които не се дължат на грешка във Вашия PHP код.

Ако не предвидите грешките, подразбиращото се поведение на PHP е да покаже съобщение за тях на потребителя, заедно с препратка към страница от ръководството с описание на функцията, в която е възникнал проблема, името на файла и реда от кода, генерирал грешката. В много случаи PHP продължава да работи и след показването на съобщението. Това съобщение за грешка всъщност е предназначено за вас – разработчика, а не за потребителите на сайта. Потребителите биха оценили по-добре една страница, обясняваща на нормален език какво се е объркало, и съвсем не се вълнуват от препратки към документацията или местоположението в кода.

PHP предоставя редица възможности за разрешаване на такива проблеми, по-конкретно:

- Какви са различните видове грешки, с които могат да се сблъскат потребителите;
- Какви възможности имате Вие, в качеството на разработчик, за обработката им с PHP;
- Как да пишете Ваши собствени хендлъри на грешки;
- Преобразуването към различни механизми за докладване за грешки;

Видове грешки:

- Програмистки грешки

Понякога грешките се появяват поради грешки във Вашия код. В някои случаи това са най-лесните за поправяне грешки, тъй като в повечето случаи могат да бъдат открити чрез елементарното тестване на всички операции, извършвани от приложението. Премахването на тези грешки се свежда до поправка на кода.

- Грешки в синтаксиса/парсването

Синтактичните грешки и другите такива при парсването се улавят при компилирането на файла, преди PHP въобще да започне да го изпълнява.

```
<?php
Print"Hello!\n";
?>
```

Този пример съдържа XML таг на място, където PHP очаква да немири код. Изпълнението му води до грешка:

```
Parse error: parse error in test.php on line 4
```

Както виждате, скриптът дори не е отпечатал Hello! Преди да покаже съобщението за грешка, тъй като синтактичната грешка е открита по време на компилирането, още преди PHP да започне изпълнението на скрипта.

- Eval

При компилирането се улавят всички синтактични грешки или такива при парсването, с изключение на грешките в код, изпълняван с eval(). В случая eval, кодът се компилира по време на изпълнение на скрипта. Тук сме променили предишния пример, така че да съдържа eval:

```
<?php
```

```
print"Hello!\n";  
eval("gobbledygook/>");  
?>
```

Този път резултатът е различен:

Hello!

Parse error: parse error in /home/ssb/test.php(4): eval()'d code on line 1

Както виждате, този път грешката се прояви по време на изпълнението.

Това е така, тъй като кодът, изпълняван с eval() не се компилира докато не се изпълни самата конструкция eval().

PHP се вгражда лесно в HTML файлове, показващи HTML формуляри за събиране на данни от потребителите и файлове, качвани от тях. Събирането на данни от потребителите поражда въпроси, свързани със сигурността на уеб сайта, както и за всякаква информация за потребителите, която се съхранява от уеб сайта, както и за всякаква информация за потребителите, която се съхранява от уеб сайта. За по-висока сигурност PHP трябва да бъде с изключено регистриране на глобалните променливи (register_globals Off). За да атакуват Вашия уеб сайт или за да откраднат данните Ви, злонамерените потребители използват техники като изпълнение на скриптове от друг сайт (изпълнение на части от клиентски скриптове върху вашия уебсайт) и инжектиране на SQL (вмъкване на злонамерен код в заявки, изпълнявани от Вашата база данни). За да се защитите от атаки, трябва да сте подозрителни към всички данни, постъпващи от потребителите. Трябва внимателно да ги проверявате, за да сте сигурни, че са безопасни и не са опасни за вашия уеб сайт. Можете да защитите своя уеб сайт при качването на файлове от страна на потребителите, като проверявате размера и типа на качваните файлове. Освен това, можете да защитите информацията, която е видима в адресното поле на уеб браузъра – информацията, подавана в URL адреса – като я хеширате посредством един от няколко възможни метода (например с класа Crypt_HMAC от PEAR, който е разработен точно с тази цел). Хеширането е полезно и за защита на паролите, съхранявани на сайта с цел удостоверяване на потребителите. Друга полезна мярка за защита на уеб сайта от потребителски грешки или атаки от злонамерени хора е да разработите ваш собствен хендлър на грешки, който разпознава когато нещо не е наред и разрешава проблеми.

За да бъде едно уеб приложение полезно, данните му трябва да бъдат на разположение на всички уеб страници по време на потребителската сесия.

Един от начините за предаване на данни от една страница към друга е чрез бисквитки. Когато потребителят поиска достъп до уеб страници, вместо нея се появява страница за влизане и въведените от потребителя име и парола се сравняват с името и паролата, записани за този потребител. Ако потребителят бъде удостоверен, се записва бисквитка. Информацията от бисквитката се предава автоматично с всяка заявка за страница. Друг метод за запазване на данните между отделните уеб страници е да се използват сесии. В PHP сесията могат да се съхраняват променливи, които са достъпни и за други скриптове от тази сесия.

След като определите нужните части за Вашето **интерактивно уеб приложение**, трябва да ги организирате в едно цяло. Един от често използваните методи за организиране се нарича „един скрипт за всичко“, което означава, че index.php обработва всички заявки за различни страници. Друг често срещана организация е „скрипт за всяка функция“.

Общият принцип е да се отделя разположението от логиката. След като организирате различните части цялостно приложение, сте „готови за бой“

9. Java аплети

Java е обектно-ориентиран програмен език, разработен от Sun Microsystems . Трябва да се отбележи, че той не е свързан с JavaScript, който е език за скриптове, разработен от Netscape Navigator, и който работи в HTML документите в брауъра. Тъй като Java е пълен програмен език /подобно на C, C++/, той може да се използва за създаването на цялостни приложения. Първоначалния принос на Java към Мрежата обаче, беше под формата на Java аплети – самосъдържащи се, малки изпълними програми. Тези програми, именувани с наставката class, могат да се поставят направо в уеб страницата, подобно на графики.

Предимства и недостатъци на Java аpletите:

Аpletите са идеални за уеб разпространение поради следните причини:

- Те не зависят от платформата;
- Те се свалят и изпълняват изцяло на клиента, така че сървърът не се натоварва продължително;
- Обикновено файловете на аpletите са доста компактни и бързо се свалят от сървъра;
- Те не изискват специален патентен plug-in за инсталирането си. Сега всички основни брауъри поддържат Java, което означава, че шансовете потребителите да видят вашия аplet са доста големи.

Разбира се, всяка утопична технология си има своя тъмна страна и за съжаление в реалния свят брауърите могат да проявят темпераментно отношение по отношение на начините, по които манипулират Java applets. Брауърите са прословути със своите сринове в присъствието на жадните за изчисления аплети. Най-общо казано, на брауърите им отнема много време да инициализират Java, което означава че прогонва потребителите. Когато Java за първи път се появи на сцената, настана голяма суматоха сред уеб разработчиците, но след това ентузиазмът намаля и отстъпи мястото си на въпросите, свързани с изпълнението и разработването на други уеб мултимедийни решения.

Java applets могат да се използват за всичко! От елементарни анимации до симулатори на битки. Тъй като Java позволява моментни изчисления, те са подходящи за програми, които взаимодействат с потребителския вход. Не е учудващо, че голям процент от игрите представляват Java applets. Аpletите също така се използват и за по-практични цели като калкулатори и електронни таблици. Още по-интересно е, че те могат да доставят данни и на живо (заглавия на новините, борсови котировки, спортни резултати,..) и позволяват на потребителите да се придвижват през сложни информационни връзки. Вероятно съществуват стотици Java applets. Това е просто най-общ списък на нещата, които те правят:

- Помощни програми – калкулатори, календари, часовници, електронни таблици;
- Текстови ефекти – превъртащи се подвижни текстове, пулсиращ текст, блестящи цветни текстови съобщения;
- Аудио ефекти – дигитални „китари”, радиобутони;
- Игри – Asteroid, Hangman, Minesweeper, кръстословици;

- Разни – диаграми на биоритми, симулатори на битки, ежедневни котировки и др.

Ако Ви трябва специално пригоден за вашата страница аplet, най-добре е да си наемете програмист, който да създаде един според Вашите изисквания. Обаче съществуват и известен брой безплатни аплети или аплети с лицензна такса, които можете да си свалите от библиотеките в Мрежата.

Едно добро начало може да се окаже разделът за аплети в сайта на Java на Sun на адрес: <http://java.sun.com/applets/>. Тази страница предоставя списък от връзки към източници за аплети.

Ако търсите полезни аплети, които да можете да ползвате на минутата, проверете в JavaBoutique на адрес: <http://javaboutique.internet.com/>. Тук ще намерите стотици налични аплети, придружени с ясни инструкции за употребата им. Този сайт предоставя чудесна възможност да добавите интерактивни ефекти на страниците си, без да учите програмиране.

Добавяне на аплети в страница: По настоящем съществуват два начина за добавяне на аплети към уеб страница: с инструкцията за форматиране `<object>`, която се препоръчва от HTML 4.01, и с по-добре поддържаната инструкция за форматиране `<applet>`. W3C обезцени инструкцията за форматиране `<applet>` и всичките ѝ атрибути в полза на инструкцията за форматиране `<object>`. Въпреки това инструкцията за форматиране `<applet>` е по-добрият избор, тъй като поддръжката на аплети, вградени с `<object>`, все още не е постоянна и не може да се открие метод, който да работи на всички браузъри. Освен това някои аплети изискват употребата на инструкцията за форматиране `<applet>`, така че първо прочетете документацията на аплета. Ето и двата метода:

Добавяне на аплети с `<object>`:

Може да добавите елементарен самосъдържащ се аplet към HTML документ с помощта на инструкцията за форматиране `<object>`, ето така:

```
<OBJECT CLASSID="applet.class"CODEBASE="http://somedomain.com/classes/">
```

An applet with some useful function should display in this space.

```
</OBJECT>
```

Добавяне на аплети с `<applet>`:

Инструкцията за форматиране `<applet>` представлява контейнер за някакъв брой параметрични инструкции за форматиране (`<param>`). Ето как би изглеждала инструкцията за форматиране `<applet>` за някаква игра:

```
<APPLET CODEBASE=class CODE="Wacky.class" WIDTH=300 HEIGHT=400>
```

```
<PARAM NAME="Delay" VALUE="250">
```

```
<PARAM NAME="Time"VALUE="120">
```

```
<PARAM NAME="playSounds" VALUE="YES">
```

```
</APPLET>
```

10. Необходимост от списъци със стилове

Тъй като XHTML е приложение на XML, е необходимо да се използват списъци със стилове, в които да се съдържа информацията за оформлението. Разделянето на данните

от стиловете позволява една и съща информация да бъде представена на различни устройства единствено посредством смяна на списъка със стилове. При използването на различни списъци съдържанието на една и съща уебстраница може да се изобрази на компютърен монитор, телевизионен екран, джобно устройство или дисплей на мобилен телефон.

Моделът „едни данни – много устройства“ премахва ограниченията пред използването на съдържанието и структурата на документите на XHTML от множество различни приложения. Когато получи заявка от сървъра, скрипт или аplet (малка програма, написана на Java, която работи в брауъра) се грижи да подаде към съответното потребителско устройство правилния списък със стилове. В момента съществуват два езика за писане на списъци със стилове към XML или XHTML:

- CSS (Cascading Style Sheets) – поради по-добрата и широка поддръжка от страна на брауърите този език се използва все по-често.
- XSL (Extensible Style Language) - като приложение на XML не само описва оформлението на страницата, но и позволява документи на XML да се трансформират към XSL и обратно. В рамките на списък със стилове на XSL е възможно да се използват стилови правила на CSS, така че двата езика да се допълват взаимно.

10.1. Миграция от HTML към XHTML

Кога да изберем XHTML пред HTML? Ако създавате нов сайт, може да използвате строго форматиран XHTML, CSS за оформлението му. Ако вече имате сайт, Ви предстои повече работа. Миграцията на работещ сайт към XHTML трябва да бъде постепенен процес, а не рязък преход към нещо ново. XHTML позволява да започнете да прилагате новия синтаксис, докато старият HTML все още контролира някои атрибути, като например цветовете на страницата и връзките. Мигрирайки и спазвайки строго правилата в XHTML, все повече ще изчиствате кода на съществуващите страници, ще планирате структурата на кода за новите страници, ще премахвате отпадналите вече елементи и ще отделяте оформлението в отделен CSS файл. Ето и стъпките, през които трябва да преминете при миграция от HTML към XHTML:

1. **Оценка на съществуващия код** – Проверете за спазването на основните синтактични правила на XHTML. Използвани ли са затварящи тагове? Написани ли са всички тагове изцяло с малки букви? Атрибутите поставени ли са в кавички? Колко работа ще е необходима, за да се приведе кодът в строго форматиран вид? По-голямата част от тази работа може да се автоматизира посредством различни HTML редактори.
2. **Оценка на съществуващото оформление** – Колко често в кода Ви се срещат отпаднали елементи, като , и отпаднали атрибути, като „bgcolor“, „face“ и „size“? При много сайтове тази информация може да достигне до 50% от целия код. Започнете да планирате как да изнесете информацията за оформлението в отделен CSS файл.
3. **Правила за писане на код** – Създайте си правила за писане на код и ги спазвайте навсякъде в сайта. Уверете се, че новодобавеното съдържание в сайта следва актуалните стандарти на кодиране и тези на CSS. Колкото по-стриктно спазвате стандартите, толкова по-лесна ще бъде поддръжката след това.
4. **Използване на CSS** – Започнете от по-прости списъци със стилове, които да съдържат основните характеристики, като цветове, шрифт и размер на буквите. Обмислете използването на някои от по-сложните възможности на CSS като

класове, които да Ви позволяват да именуваме и стандартизираме различни стилове в сайта. Докато изграждате новите си стилови правила, започнете да премахвате стария форматиращ код на сайта.

5. **Съвместимост със старите версии** – Старите браузъри пренебрегват новия синтаксис на XHTML, но все пак трябва да проверите, че съдържанието се чете добре. Тествайте внимателно стиловите си правила в CSS, за да се уверите, че се интерпретират добре от старите браузъри. Ако не се поддържат, ги променете, като се съобразявате с това, или обмислете да направите отделна група от списъци, които да подавате към по-старите браузъри. Това е възможно при някои уебсървъри.

11. Универсалния език XML

XML (произнася се екс-ем-ел, от англ. eXtensible Markup Language — разширяем маркиращ език (PME)) е стандарт (метаезик), дефиниращ правила за създаване на специализирани маркиращи езици, както и синтаксисът, на който тези езици трябва да се подчиняват. Сам по себе си той е безполезен, защото указва само как да бъде структуриран един документ (чрез маркиране с етикети), но не и какво означават отделните маркери (етикети). Ето защо е по-правилно да се нарече метаезик отколкото език — обикновените езици имат семантика, т.е. предават някаква информация, докато XML указва само граматиката (по-точно синтаксиса) на езиците, базирани на него.

Ползата от XML се състои в това, че синтаксисът (структурирането) на документите се измисля само веднъж, а специализираните маркиращи езици само дефинират семантиката т.е. набора от маркиращи етикети и тяхното значение. Другата изгода е, че XML прави възможно разработването на семантично агностични програмни средства които могат да обработват всички XML базирани езици, дори и такива които не са съществували по време на съставянето на софтуера. Тези програмни средства могат след това да се използват и в изработването на софтуер който е написан за конкретен XML език. Или накратко — разделянето на синтаксиса от семантиката е поредното въплъщение на принципа „Разделяй и владей“.

XML е създаден като олекотена версия на SGML, целящ по-лесното му внедряване и налагане сред разработчиците. Създаден е от Уеб консорциума (World Wide Web Consortium) и съответно е добре пригоден за употреба в Интернет пространството.

В XML информацията се маркира като се загражда с етикети:

```
<name>Иван Димитров Георгиев</name>
```

Тук текстът Иван Димитров Георгиев е маркиран с етикет <name> като по този начин указваме, че маркираният къс информация искаме да бъде разглеждан като едно цяло, имащо определено значение. Етикетите вървят по двойки — отварящ и затварящ, маркиращи съответно началото и края на маркирания текст.

Освен това етикетите могат да се влагат един в друг, така че да са получи йерархична организация на информацията:

```
<person>  
<name>Иван Димитров Георгиев</name>
```



```
<country>България</country>
<language>български</language>
<language>руски</language>
</person>
```

По горния начин бихме могли да маркираме информация, свързана с отделни хора. В примера цялата информация е маркирана с етикет `<person>`, тъй като искаме да укажем, че се отнася за човек.

Това, което XML указва, е как да бъдат използвани етикетите. Например всеки документ трябва да започва с коренов етикет, така че цялата информация ще бъде маркирана от поне един етикет. Друго правило е изискването етикетите да не се припокриват (кръстосват) като в следващия пример:

```
<!-- Неправилно оформен документ! -->
<person><name>Иван Димитров Георгиев</person></name>
```

Правилно е следното:

```
<person><name>Иван Димитров Георгиев</name></person>
```

Документ, който се подчинява на тези (и много други) правила описани в XML-стандарта, се нарича добре оформен XML-документ. Това, че един документ е добре оформен, гарантира, че всички програми, написани според стандарта, ще могат да го манипулират така, както всички останали валидни XML-документи. Обикновено първата стъпка преди програмната обработка на един документ е проверката за оформеност. Има много широкодостъпни програмни средства, които извършват тази проверка.

Тъй като XML не се занимава със семантичната страна на документите, следният текст също е част от добре оформен XML-документ:

```
<name>
  <person>България</person>
</name>
```

Разбира се, най-вероятно бихме искали етикетът `<name>` да се среща само в `<person>` етикет, но не и в `<language>` етикет, тъй като с него бихме искали да маркираме име на човек. Също така ще е добре, ако държавата, в която живее човекът, бъде маркирана винаги с етикет `<country>`, а не с някой друг.

Именно това е ролята на XML разширенията — да опишат допълнителни синтактични правила като това кои етикети къде могат да се срещат, както и да се изясни тяхната семантика (значение). Тези допълнителни синтактични правила са в тясна зависимост от семантиката на етикетите и по това се различават от общите XML правила. Описването им лесно може да се формализира, за която цел са създадени специални езици. Тези езици се наричат XML схеми (XML schemas) и задават допълнителните

синтактични правила, на които трябва да отговаря документ, написан според някои от разширяващите XML стандарти.

Документи, които отговарят на правилата, описани в дадена XML схема, се наричат валидни. Валидните документи винаги са и добре оформени. Една от най-простите XML схеми е DTD (Document Type Definition — Дефиниция на документен тип). Схемата, на която отговаря XML документът, се посочва в началото му и следва определен от XML синтаксис:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Горният код показва, че документът е написан според стандарта на Web Consortium, XHTML 1.0 Transitional (който представлява разширение на XML). Формалността на описанието в XML схемите позволява проверката за валидност на документите да се автоматизира.

Втората задача на разширяващите стандарти — описване на семантиката — е по-трудна и обикновено се описва в изказан, полуформален текст, предназначен да бъде четен от хора (за разлика от XML схемите, които са формални и могат да се разбират от програмите). Спазването на семантиката е изцяло задължение на този, който съставя документа, и не може да бъде проверено от програмата.

За да разберем HTML, е необходимо кратко въведение в XML. Подобно на HTML, XML също произхожда от SGML, но няма предварително дефинирани елементи като <h1> или <p>. Основната разлика между тях е, че HTML представлява набор от предварително дефинирани елементи, които браузърът разбира, докато XML е метаязык. Според PC Webopedia представката “мета”, означава „относно“, т. е. метаязык означава “**език на езика**“. Следователно XML е език, който описва описващия език, позволявайки ни да създаваме собствени елементи за посрещане на специфичните ни информационни нужди. Тази гъвкавост позволява:

- Да се добавят нови елементи или свойства, с които да се разширят възможностите на HTML.
- Да се създават нови браузри или софтуерни приложения за алтернативни начини на достъп до Интернет ресурси.

Кодът на XML много прилича на кода на HTML с някои синтактични разлики. Основната разлика между двата езика е, че **XML позволява създаването на елементи, които могат да опишат всякакъв тип информация, какъвто бихте пожелали**. Например да предположим, че поетите искат да създадат свой описателен език, който да представя различните части на едно стихотворение ето така:

```
<poem>
<title>Ода за Интернет</title>
<stanza>
<line>Уебдизайнът е безкраен</line>
<line>Да го видиш няма време</line>
<line>Сайтове да критикувам</line>
```

<line>първа радост е за мене!</line>

</stanza></poem>

Забележете, че този код е много сходен със стандартния код на HTML, но таговете са специфични за типа съдържание, което обхващат. За разлика от стандартния HTML, браузърът не знае как да изобрази тази информация, освен ако документът не е придружен от списък със стилове, указващ например, че елементът <line> трябва да бъде показван като текст с шрифт Helvetica и размер 12.

Ползи от прехода към XHTML: Едно от най-важните предимства при прилагане на правила от XML в HTML е по-строгий синтаксис на кода при XML. Преминавайки към XHTML, все повече сайтове ще трябва да актуализират кода си, който не отговаря на стандартите, включително и да премахнат отпадналите елементи, които W3C обяви за остарели и в бъдеще няма да бъдат поддържани от по-новите версии на HTML. Колкото повече уебсайтовете използват стандартизиран код, толкова по-лесно браузърите ще разпознават правилния код от некоректния. Освен това XHTML е замислен така, че да се интерпретира правилно и от браузъри, които поддържат HTML 4.01. първоначално HTML е създаден да описва ограничен тип документи и не успява да се адаптира добре към лавинообразния интерес към Уеб. От друга страна, тъй като XHTML е базиран на XML, той е разширяем, което означава, че дизайнерите могат да обогатяват възможностите му, а разработчиците да реагират лесно на бъдещите изисквания. Също така XHTML е замислен да поддържа разнообразието от бъдещи устройства, които тепърва ще осъществяват достъп до Интернет с развитието на технологиите. Всеки съвместим с XHTML софтуер трябва да може да чете и да показва XHTML независимо от типа на компютъра или монитора.

Тъй като XML разполага с по-добър механизъм за обработка на данни, новата версия на XHTML обещава да работи гладко с приложения за бази данни и за автоматизиране на работните процеси. Следващото поколение HTML ще включат подобрена поддръжка на елементите за формуляри, наблягайки по-скоро на обработката на данните, отколкото на тяхната визуализация. Обменът на данни между различните приложения и устройства също ще бъде улеснен. Таблиците ще поддържат модел на данните, който позволява различно изобразяване на съдържанието според изходното устройство. Например таблица с цените на фондовата борса може да бъде изпратена към различни крайни устройства, като джобен компютър (PDA) или мобилен телефон, и да се визуализира по подходящ начин върху всеки от съответните дисплеи. Това решение придава допълнителна полезност на данните – те се генерират еднократно, но могат да бъдат визуализирани по различен начин.

11.1. Синтактични правила на XML, XHTML

XML и съответно XHTML съдържат редица синтактични правила, различни от тези в HTML. XHTML приема следните задължителни правила от XML.

- Документите са строго форматиращи
- Елементите са вложени симетрично
- Използването на главни и малки букви в имената на елементите е от значение
- Затварящите тагове са задължителни
- Празните елементи се маркират със затваряща наклонена черта

- Стойностите на атрибутите са заградени в кавички

Документите са строго форматирани

Старо форматирани документ е този, който се подчинява на синтактичните правила, описани по-горе. Документ, който не отговаря на тези правила, не се приема за валиден XHTML.

Елементите са вложени симетрично

Таговете в XHTML могат да се влагат един в друг, но не трябва да се припокриват. Всяка двойка отварящ и затварящ таг трябва напълно да обхваща вложените в нея елементи. Например следния XHTML е некоректен:

```
<p><strong>някакъв текст...</p></strong>
```

Затварящият таг на атрибута за получер шрифт трябва да бъде преди затварящия таг за параграф. Ето правилното влагане на елементите:

```
<p><strong>някакъв текст...</strong></p>
```

XHTML е чувствителен към регистъра на буквите

За разлика от HTML, в XHTML главните и малките букви се интерпретират различно. Ако браузърът е съвместим с XHTML, той ще интерпретира <BLOCKQUOTE> и <blockquote> като два различни елемента. Макар и да не е задължително, общоприет принцип в XHTML е да се използват изцяло малки букви при именуването на елементи и атрибути.

Затварящите тагове са задължителни. За някои елементи HTML, където всеки непразен елемент се нуждае от затварящ таг. Например следните два елемента <p> нямат затварящи тагове:

```
<p> Това е първи абзац.
```

```
<p> Това е втори абзац.
```

Правилното изписване в XHTML е:

```
<p>Това е първи абзац</p>
```

```
<p>Това е втори абзац</p>
```

Празните елементи се маркират със затваряща наклонена черта

Празните елементи трябва да ползват затварящ таг или да се маркират като празни чрез добавяне на дясно наклонена черта (/) в края на тага им. Например елементът
, когато се използва в XHTML, се преобразува в
. Елементът изглежда по следния начин:

```

```

Обърнете внимание на дясно наклонената черта в края на елемента. По-старите браузъри я игнорират, затова няма опасност празните елементи да бъдат представени неправилно, ако ги направим съвместими с XHTML. За да сте сигурни, че старите браузъри ще пропуснат наклонената черта, задължително поставяйте интервал пред нея. Например
 би трябвало да се преобразува в
.

Стойностите на атрибутите са заградени в кавички

За разлика от HTML, всички стойности на атрибути в XHTML задължително се поставят в кавички. Ето пример за некоректен XHTML:

```
<h1 align=center> Заглавие </h1>
```

Правилно изписване е:

```
<h1 align="center"> Заглавие</h1>
```

Прилагане на синтактичните правила на XHTML

Ако предвиждате в бъдеще да работите с XHTML, трябва да следвате тези синтактични правила в кода си на HTML. Това ще Ви гарантира, че кодът, който създавате днес на HTML, в бъдеще ще работи и с XHTML. Ако имате стара код на HTML, помислете дали да не го преработите в съответствие със стандартите на XHTML. Код, подобен на долния, е много често срещан в Уеб днес. Въпреки че е синтактично некоректен, той се обработва правилно от браузъра:

```
<H1>Прост код на HTML</h1>
```

```
<P>Това е един абзац.
```

```
<IMG SRC="xml.gif">
```

```
<H3>Неномериран списък</H3>
```

```
<UL>
```

```
<LI>първи елемент
```

```
<LI>втори елемент
```

```
<LI>трети елемент
```

За да превърнете този код в коректен XHTML, е необходимо да приложите изброените по-горе правила. Резултатът е следният:

```
<h1>Прост код на HTML</h1>
```

```
<p>Това е един абзац.</p>
```

```
<h3>номериран списък</h3>
```

```
<ul>
```

```
<li>Първи елемент</li>
```

```
<li>Втори елемент</li>
```

```
<li>Трети елемент</li>
```

```
<ul>
```

Много безплатни и платени софтуерни програми могат да Ви помогнат при превръщането на вашия код в стандартен XHTML. Сред тях са HTML Tidy

<http://tidy.sourceforge.net/> и Tidy GUI - <http://perso.wanadoo.fr/ablavier/TidyGUI/>.

Стандартът XML, въведен през февруари 1998 г., завладя софтуерната индустрия. За първи път се предлага универсален, приспособим формат за структуриране на документи и данни не само във виртуалното пространство, но и на всякъде. Светът го възприе много добре.

11.2. Сравнение на XML и HTML:

Въпреки, че е базиран на същата технология, която даде началото на добрия старт HTML (и точно както HTML, използва тагове, атрибути и стойности, за да форматира структурираните документи), XML е доста различен от уважавания език за маркиране, който трябва да замени.

HTML е базов език за маркиране на уеб страници. Разполага с фиксиран брой тагове и малък набор от до известна степен несъгласувани правила. В HTML трябва да заграждате в кавички някои тагове, за други пък не трябва да го правите, а трети можете да заграждате или да не заграждате в зависимост от настроението Ви. Тази свобода позволява на всеки лесно да създава уеб страница, дори и да не знае какво точно прави – което, разбира се, беше целта. И това беше добра цел, когато виртуалното пространство се нуждаеше от базово съдържание и почти нищо повече. Но в днешното по-усложнено виртуално пространство, където страниците често пъти се създават с използването на инструменти за публикуване, а съдържанието трябва да лежи между базите данни, уеб страниците, безжичните устройства и принтерите, липсата на унифицирани правила в HTML пречи на използването на данните за други цели, без да бъдат променяни. Лесно е да се преобразува текст в HTML, но не трудно да се преобразуват данни, маркирани в HTML, в някакъв друг формат.

Освен това HTML е просто език за форматиране и не е особено точен. Не съдържа никаква информация за съдържанието, подлежащо на форматиране, което отново ограничава възможността за повторното използване на това съдържание при други условия. И разбира се, HTML се използва единствено във виртуалното пространство.

За разлика от него, базираното на XML маркиране е обвързано със съгласувани правила и е в състояние да се използва далеч извън виртуалното пространство. Когато маркирате даден документ в XML, вие не просто го подготвяте за визуализиране върху дадена уеб страница. Вие го кодирате с тагове, които могат да бъдат разбрани от всяка среда, поддържаща XML. Наподобява един родител, много деца или казано по точно, XML е език, предназначен за създаването на други езици. Стига да се придържат към неговите правила, библиотекарите имат възможността да създават XML маркиране, чиито потребителски тагове спомагат за посещането на нуждите на каталогизирането. Музикалните компании могат да създават XML маркиране, чиито тагове са изпълнител, запис, композитор, продуцент, данни за авторско право, данни за платен хонорар и т. н. Композиторите могат да организират своите партитури в потребителски XML език за маркиране, наречен MusicML. Тези потребителски XML езици се наричат приложения (application) и са съвместими един с друг, тъй като всички те са XML. т. е. всеки XML парсер може да разбира тези приложения и приложенията лесно могат да обменят данни помежду си. По този начин данни от XML база данни на звукозаписна компания могат да се озоват в каталога със музикални записи на дадена библиотека, без необходимостта от човешки труд, без грешки и без затъването в софтуерни несъвместимости.

11.3. Синтаксис на XML документа

Ако сте добре запознати с HTML, няма да Ви е трудно да разчетете всеки елементарен XML документ.

```
<?xml version="1.0" standalone="no"@?>
<!DOCTYPE accounts SYSTEM ""simple.dtd">
<accounts>
<costumer>
```

```
<name>Bobby Five</name>
<accountNumber>4456</accountNumber>
<balance>111.32</balance>
</dustomer>
</accounts>
```

Първия ред кодът между `<?xml и ?>` се нарича XML декларация. Тази декларация съдържа специална информация относно XML процесора (програмата, която чете XML), показваща, че документът отговаря на изискванията на версия 1.0 от XML стандарта. Като допълнение атрибутът `standalone="no"` казва на програмата, че е нужен външен DTD, за да бъде интерпретиран документа правилно. Втория ред DOCTYPE декларацията. Тя обозначава основния елемент (в нашия пример `accounts`) и DTD на документа. Основният елемент е елементът, който съдържа всички останали елементи в документа. Той трябва да е уникален, което означава, че може да се използва само веднъж в документа. Последната част от декларацията е указател към самото DTD. Идентификаторът SYSTEM сочи към местоположението на DTD – неговото URL. В нашия пример DTD на документа се намира в отделен файл, наречен `simple.dtd`.

Добре оформен XML

Браузърите често възстановяват небрежно написан или неправилен HTML. Но при XML документите, нещата стоят по различен начин. Тъй като XML езиците варират, правилата за кодирането на документите трябва да се спазват изцяло, за да бъде документът правилно интерпретиран от XML правилата за описание, той се нарича добре оформен.

Основните правила за добре оформен XML документ са следните:

- Не трябва да има празно пространство (шпации или преминаване на нов ред) преди XML декларацията.
- Всички стойности на атрибутите на елементите трябва да бъдат поставени в кавички (или единични, или двойни).
- Инструкциите за форматиране и атрибутите различават главни от малки букви; например `<par>`, `<PAR>` и `<Par>` се приемат като три различни инструкции за форматиране. Всеки елемент трябва да притежава начална и крайна инструкция за форматиране, освен ако не е празен елемент.
- Ако една инструкция за форматиране е самостоятелен празен елемент, в края на записа си трябва да има наклонена черта (например ``).
- Всички начални и крайни инструкции за форматиране трябва да са вградени правилно и да не се припокриват.
- Документът трябва да притежава един основен елемент – уникален елемент, в който е поставен целия документ. Основният елемент може да се използва само веднъж в документа.
- Отделни описателни символи (`<`, `&`, `>`) не се допускат в текста; използвайте еквивалентните им стандартни символни обекти. Допълнителна литература може да се намери на: <http://www.xml.com>.

Разширяемият език за маркиране (XML), подобно на Hypertext Markup Language (Език за маркиране на хипертекст), произлиза от Standard Generalized Markup Language (GSML – Стандартен обобщен език за маркиране). Една от характеристиките на GSML е разделянето на формата от съдържанието. Независимо дали документът е създаден във формат A4 или във формат за писмо, форматът се описва независимо от съдържанието на документа. По тази причина един и същи документ може да бъде представен в

множество формати, без да се променя съдържанието. Този принцип на езиците за маркиране се използва при уеб услугите чрез разделението на инстанцията на документа, която съдържа данните, и схемата, която описва структурата и типа на данните, включително семантичната информация, която се използва за съгласуване на документа с множество езици за програмиране и софтуерни системи.

XML представя голям брой спецификации, повечето от които се отнасят до обработката на документа, отколкото до обработката на информацията. Тази глава описва тези XML спецификации и технологии, които са най-важни за уеб услугите и които по принцип могат да се определят като простиращи се „отвъд маркирането“, за да осигурят средства за структуриране и сериализиране на данните.

XML е представен от набор от спецификации, публикувани от World Wide Web Consortium (W3C). Много приложения – по-мощни програми, документни формати и инструменти – включват една или повече от тези спецификации. Два аспекта на XML са от съществена важност за уеб услугите: XML формата за представяне на данни, или инстанцията на данните, и асоциираните схеми, които дефинират семантиката на данните. Разбира се, много други свързани с XML технологии представляват интерес и са важни за уеб услугите, поради което са включени в технологии за уеб услуги и XML програмите и инструментите, които ги реализират.

XML се използва в широк кръг приложения. Главните му приложения при уеб услугите са форматиране на данни, сериализация и трансформация. Уеб услугите комуникират, като обменят форматирани инстанции на XML документи, които пренасят данни.

Уеб услугите се описват, като се използват XML схеми, които дефинират типове данни, структури и семантика. В дефинициите на технологиите за уеб услуги се използва голямо разнообразие от XML спецификации – схеми, пространства от имена, разширяеми набори от стилове и др. Главното предимство, което XML предлага на уеб услугите, е **независимост на данните**, което означава, че типовете данни и структури не са обвързани с основните реализации на услугите. Преди това типовете данни и структурите за разпределено изчисление са били дефинирани в отделни езици за програмиране, езици за описание на бази данни и езици за описаниена междинни интерфейси. За да се възползват от независимостта на данните, приложенията трябва да преобразуват данните в XML и да трансформират данни от XML в естествения им формат. Въпреки че XML е замислен като език за маркиране на документи, сега голяма част от XML общността разглежда XML като технология за описание и сериализация на данни.

12. Технологията DHTML

Когато се създава нова уебстраница, една от целите е тя да бъде с нещо по-добра от предишните. Дизайнерските решения непрекъснато се развиват – от обикновено структуриране на информация, през включване на атрактивни графични елементи, до изграждане на цялостна концепция, реализирана с елегантен дизайн и употреба на символика и визуални метафори, характерни за сайтовете от третото поколение. Но ако до скоро потребителите на Интернет се интересуваха главно от съдържанието и информацията, която можеха да намерят там, то днес все повече хора сърфират за удоволствие и за да се забавляват. Това предполага промяна на подхода при изграждане на уебстраниците, който до сега изискваше единствено владение на пространството, умение за разпределяне и оформяне на съдържанието в подходящ и интуитивно понятен

стил. С оглед новите тенденции в потреблението все по-често се налага владеене не само на пространството, но и на времето, т. е. на последователността при представяне на информацията – не само като навигация в сайта, но и в рамките на всяка една страница.

Уебдизайнът все повече се отдалечава от изобразителното изкуство и се доближава към похватите в киното и телевизията. Представянето на уебстраниците се режисира, определя се подходящ сценарий за поднасяне на съдържанието, който да отговаря на поставени цели. Уебдизайнерите все повече се стараят да привлекат и задържат вниманието на потребителите точно както масмедииите и все по-малко разчитат на стойността на информацията, която предлагат.

За реализиране на динамично съдържание, разпределено както в пространството, така и във времето, се използва технологията DHTML. Названието представлява абревиатура на Dynamic Hyper Text Markup Language или по-понятно Dynamic HTML, динамичен вариант на основния формат за създаване на уебстраница – HTML. За да се премине от твърде статичния и ограничен според общото мнение HTML към динамичен, в него се въвежда възможността за слоеве, всеки от които носи определено съдържание и възможност да бъде преместен, показан или скрит в даден момент. По този начин може да се изгради уебстраница, чието съдържание да се променя динамично според действията от страна на потребителя. Всички движения се осъществяват от браузъра, без презареждане и респективно, без да се извършва трафик по Мрежата. С помощта на DHTML могат да се разработят уебстраница, съдържащи както атрактивни промоции, така и забавни игри и пъзели. Технически погледнато създаването на слоевете при DHTML се реализира с CSS (Cascading Style Sheets), а тяхното управление – с JavaScript. Употребата на DHTML става възможна едва когато се появяват браузъри, поддържащи новия стандарт HTML 4.0, CSS –технологията за работа с таговете и JavaScript 1.2 –втора версия на разпространения език за програмиране в уебстраниците.

Проблеми с уебстраниците, базирани на DHTML, срещат както потребителите с по-стари браузъри, така и уебдизайнерите. Те се дължат на факта, че двата най-употребявани браузъра – MSIE (Microsoft Internet Explorer) и Netscape – използват различни команди за едни и същи операции. Докато при първия обръщение към даден обект се извършва с document.all, то при втория това става с document.layers. На практика се налага разработването на усложнен JavaScript, който в зависимост от браузъра да използва единия или другия вариант за работа със създадените в уебстраницата слоеве. По данни за България на ABC Design&Communication към месец юни тази година потребителите на Интернет, чиито браузъри не поддържат DHTML, са едва 5-8%. Това означава, че пречките в това отношение са почти преодолени и вече е на разположение едно от най-мощните средства за създаване на **динамични, „интелигентни“ или интерактивни уебстраница.**

13. Език за описание на уеб услуги: WSDL

Уеб услугите предлагат софтуерно-ориентиран вариант на бизнес или потребителска функция, с която приложенията могат да взаимодействат в мрежа. Едно такова взаимодействие трябва да бъде описано и представено на потенциалните клиенти, за да може да бъде успешно реализирано с използването на уеб услуги. Нещо повече, потребителите трябва да са в състояние да разберат как да взаимодействат с услугата: какви данни се очаква да получат, да ли услугата връща някакви резултати и какъв комуникационен или транспортен протокол поддържа.

Уеб услугите съществуват отдавна, но в доста примитивен вариант. Възможно е да се изпрати низ от търсене или заявка за нивата на акциите, като се добави информация към

URL адресите. Този формат обаче има ограничена полезност и обхват по много причини, една от които е ограниченията на размера и използваемостта на един URL адрес за предаване на данни. Освен това създаването на какъв адрес е трудно и досега не е дефиниран последователен, стандартен подход за описването му.

Стандартната форма на уеб услугите поставя заявка и отговора в XML документ, т. е. в източниците, към които се обръщат URL адресите, а не в параметрите на самите URL адреси. Навлизащите стандарти за уеб услуги използват по-универсален подход, така че не се налага всеки уеб сайт да решава индивидуално как да използва разширените URL адреси. В следващия пример е илюстриран най-примитивният тип уеб услуга:

Тук входните данни се пренасят като параметър към URL адреса на уеб страницата, която реализира услугата – в този случай функция за търсене на служител, която връща e-mail адрес и телефонен номер.

Основни принципи на WSDL: Езикът за описание на уеб услугите, е създаден за описание и публикуване по стандартен начин на форматите и протоколите на всяка уеб услуга. Стандартите за интерфейси на уеб услуги са необходими, за да се гарантира, че не трябва да създавате специални взаимодействия с всеки сървър във виртуалното пространство, както в днешни дни с разширения URL от браузър.

WSDL елементите съдържат описание на данните, като обикновено се използват една или повече XML схеми, които се предават на уеб услугата, така че изпращачът, и получателят да могат да разберат данните, които се обменят. WSDL елементите съдържат също описание на операциите, които трябва да бъдат извършени с данните, така че получателят да знае как да ги обработи, както и обвързване с транспортен протокол, така че изпращачът да знае как да ги изпрати. Обикновено WSDL се използва със SOAP, а WSDL спецификацията включва методи за обвързване със SOAP.

WSDL е разработен от Microsoft, Arrib и IBM.

WSDL елементи:

WSDL разделя уеб услугите на три специфични, разпознаваеми елементи, които могат да се комбинират или използват, след като бъдат дефинирани. Преоразуването от съществуващи приложения означава съгласуване с тези елементи, които определят съдържанието и типовете данни на съобщението, операциите, които се извършват с него, и специфичните транспортни протоколи за обмен на съобщенията с операциите в мрежата. В рамките на тези елементи има други поделементи или части:

- **Типове данни:** типовете данни – под формата на XML схеми или друг механизъм, който се използва в съобщенията.
- **Съобщение:** абстрактна дефиниция на данните под формата на съобщение, представено или като цял документ, или като аргументи които се асоциират с извикването на метод.
- **Операция:** абстрактна дефиниция на операция за съобщение, като например именуване на метод, опашка със съобщения или бизнес процес, който ще приеме и обработи съобщението.
- **Тип порт:** абстрактен набор от операции, асоциирани с една или повече крайни точки, които дефинират колекцията от операции за методи за обвързване; тъй като колекцията от операции е абстрактна, може да бъде асоциирана с множество транспортни протоколи чрез различни методи за обвързване.
- **Метод за обвързване:** конкретният протокол и формати на данните за операциите и съобщенията, дефинирани за определен тип порт.

- **Порт:** комбинация от метод за обвързване и мрежов адрес, осигуряваща целевия адрес на съобщението на услугата.
- **Услуга:** колекция от свързани крайни точки, обхващаща дефинициите на услугите във файла; услугите съгласуват метода за обвързване с порта и включват дефиниции за разширяване.

За щастие тези части на WSDL обикновено се генерират, като се използват инструменти, които трансформират метаданните от съществуващите бекенд технологии в информация за XML схеми, която в последствие се слива с WSDL файла. Обикновено WSDL файловете се генерират от продукти и инструменти за уеб услуги, като например XMLBus Edition.

Web Services Description Language (WSDL), осигурява механизма, чрез който дефинициите на уеб услугите са достъпни за външния свят, и който реализаторвите на уеб услуги трябва да спазват, когато изпращат SOAP съобщения. WSDL описва типовете данни и структури за уеб услуги, обяснява как да се съгласуват типовете данни и структурите в съобщенията, които се обменят, и включва информация, която свързва съобщенията с основните реализации.

WSDL е дефиниран така, че частите му могат да бъдат разработвани отделно и да бъдат комбинирани за създаване на подробен WSDL файл. Типовете данни и структурите могат да бъдат споделяни между множество съобщения, както и дефиницията на услугите, съставляващи интерфейса. WSDL описва интерфейсите и в рамките на даден интерфейс асоциира всяка услуга с основно приложение. За постигане комуникацията при уеб услугите, WSDL ги съгласува с комуникационните и транспортни протоколи. При взаимодействие на уеб услуги и двете страни споделят общ WSDL файл. Изпращачът използва WSDL файла, за да генерира съобщения в съответния формат и да използва подходящия комуникационен протокол. Получателят използва WSDL файла, за да разбере как да получи и анализира съобщението и как да го съгласува с основните обекти или програма.

13.1. Достъп до уеб услуги: SOAP

SOAP (Simple Objects Access protocol – Прост протокол за достъп до обекти) е може би най-важната от всички технологии за уеб услуги. Вярно е, че уеб услугите не биха съществували без начин за абстрактно представяне на данните и публикуване на дефиниции на интерфейси, но несъмнено SOAP реализира най-важният аспект на уеб услугите: пренасяне на данните от едно място на друго през мрежата.

Тъй като виртуалното пространство се превръща в най-важната мрежа в света, а XML се превръща в най-важният формат за представяне на данни, е логично транспортните протоколи на уеб услуги да изискват комбинация и на двете. SOAP осигурява точно това.

SOAP позволява на подателя и получателя на XML документ да поддържат общ протокол за пренасяне на данни през виртуалното пространство за ефективна мрежова комуникация. От гледна точка на виртуалното пространство, SOAP е един видразширение на протокола HTTP (HyperText Transport Protocol), който поддържа обмен на XML съобщения. Вместо да се използва HTTP, за да се поиска свалянето и визуализирането в браузър на даден HTML страница, SOAP изпраща XML съобщение чрез HTTP заявка и получава отговор чрез HTTP отговор. За да обработи правилно XML съобщението, HTTP слушателят, като например Apache или Microsoft Internet Information Server (IIS) трябва да осигури SOAP *процесор*. С други думи, всеки HTTP слушател, който получава SOAP съобщения, трябва да включва възможност за

обработка на XML. Още по-точно, HTTP слушателят, който получава SOAP съобщение, трябва да е в състояние да валидира и разбере конкретния формат на XML документ, дефиниран в спецификацията на SOAP. Спецификацията на SOAP позволява съгласуването на протокола за обмен на SOAP съобщения с одруги транспортни протоколи, въпреки че съгласуването с HTTP е единствената асоциация, която е дефинирана в спецификацията.

Въпреки името, SOAP не включва обектен модел. SOAP дефинира еднопосочен протокол за обмен на XML съобщения, върху който могат да се изграждат допълнителни приложения, включително стила на взаимодействие от тип заявка/отговор, който е аналогичен на обектно – и процедурно-ориентирана обработка, асинхронен обмен на съобщения и нотификация за събития, аналогична на ориентираните към съобщения междинни системи, непотвърдени съобщения и препращане чрез SOAP междинни точки. SOAP взаимодействията се моделират, когато възникват между SOAP възли, които могат да бъдат податели, получатели на SOAP съобщения или и двете. Специален вид SOAP възел може да изпълнява ролята на междинна точка между изпращач и получател за целите на обработка на специални хедъри. Всеки SOAP възел поддържа един или повече SOAP процесори и отговаря за обработката на SOAP блокове, когато се получи съобщение.

След като е дефиниран интерфейсът за уеб услугите, на тях има е необходим начин да комуникират помежду си и да обменят съобщения: Simple Object Access protocol (SOAP) дефинира общ формат за XML съобщения през HTTP или други транспортни протоколи. SOAP е проректиран като прост механизъм, който може да бъде разширен така, че да обхваща допълнителни характеристики, функции и технологии.

Тази глава описва частите на SOAP и предназначението на всяка една от тях. SOAP представлява еднопосочна асинхронна технология за изпращане на съобщения, която може да бъде адаптирана и използвана в множество интерактивни стилове за обмен на съобщения: ориентирани към дистанционно извикване на процедури (remote procedure call - RPC), документно ориентирани, публикуване и абонамент и др.

Ако нещо дефинира минималния критерий за уеб услуга, то трябва да се придържа към SOAP е съществена за уеб услугите. SOAP е дефиниран на много високо ниво на абстрактност и може да бъде съгласуван с произволен брой софтуерни системи, включително приложни сънвдъри, .NET сървъри, междинни системи, системи за управление на база данни и пакетирани приложения.

13.2. Откриване на уеб услуги: UDDI регистър

Инициативата на универсално описание, откриване и взаимодействие (UDDI) създава спецификации и работеща реализация на хранилище за описания на уеб услуги. UDDI регистърът може да бъде претърсван по различни критерии за категоризиране, за да се намери информация за фирми, предлагащи услуги, които представляват интерес. UDDI осигурява обществено средство за съхранение и извличане на информация за интерфейси и реализации на уеб услуги.

Тази глава описва формата на данните на UDDI и API интерфейсите на SOAP, който се използват за съхранение и извличане на информация от UDDI. Главата осигурява информация за UDDI, която подпомага физическия регистър и процеса, чрез който UDDI спецификациите и технологиите ще бъдат възприети. Виртуалното пространство има нужда от UDDI, за да осигури хранилище за информация за уеб услугите, така че издатели и потребители да могат да се откриват взаимно. Само тогава може да се оцени

истинската стойност на уеб услугите: когато потребителите на уеб услуги могат лесно и бързо да откриват и да осъществяват достъп до реализации на уеб услуги навсякъде по света.

13.3. Алтернативният подход: ebXML

Инициативата на ebXML започна почти по едно и също време с разрастването на общността на уеб услуги. През първите няколко месеца ebXML беше изцялоотделна и паралелна развиваща се кампания. Повечето от целите на ebXML са общи с целите на уеб услугите и повечето технологии се припокриват в идейно отношение. По принцип ebXML е насочен повече към индустриалното и корпоративното изчислително ниво, чиято основна цел е решаването на въпроса с дефиницията на бизнес процесите. Тук се обяснява произхода на целите на ebXML и подробно разгледана е архитектурата му. Описани са отделни спецификации и са поставени в подходящия контекст в рамките на цялостната архитектура. Архитектурата на ebXML включва повече нещата, които включват технологиите за уеб услуги, но отива отвъд тях и дефинира изискванията за качеството на услугите за надеждно изпращане на съобщения, сигурност и преговори между търговски партньори. Изпълнявайки ролята на заместител, ebXML се опитва да избегне някои проблеми.

14. JAVASCRIPT

С помощта на **JavaScript** могат да се зададат определени функции на някои елементи от уебстраницата или иначе казано – тя да се програмира. Създадено с възможностите на HTML, изображението в екрана на брауъра е статично, като единственото движение е когато се придвижваме по него. С добавянето на JavaScript в HTML – кода това може да бъде преодоляно и определени бутони или други елементи да започнат да реагират на движението на мишката.

С употребата на JavaScript уебстраницата става интерактивна –

Реагира спрямо действията на потребителя, като по този начин задържа или насочва неговото внимание в определена насока.

JavaScript е типичен пример за т. нар. скриптове, които работят на компютъра на „клиента“ – client side script. Командите на JavaScript се интерпретират и изпълняват от брауъра, което означава че те пристигат еднократно и наведнъж заедно с HTML-съдържанието на страницата и след това при тяхното действие не се осъществява трафик по мрежата.

Прието е JavaScript да се добавя по два начина към уебстраниците, за които е предназначен:

1. Директно – чрез включването му в HTML – код на страницата.

За целта се използва тагът <SCRIPT>, а самото съдържание на скрипта се поставя в таг за коментар, за да не се появява на екрана, ако брауърът не го разпознае като програмен код.

Пример: <SCRIPT LANGUAGE="JavaScript">

<!--

тук стои програмния

//-->

</SCRIPT>

2. В отделен файл – кодът, записан на JavaScript, е в отделен файл, който се свързва с уебстраницата с помощта на тага <SCRIPT> и употребата в него на параметъра SRC.

Пример:

```
<SCRIPT LANGUAGE="JavaScript" SRC="code.js"></SCRIPT>
```

Трябва да се отбележи, че действието на скриптовете е в пряка зависимост от възможностите и вида на браузъра, тъй като именно той ги изпълнява. В това отношение съществуват известни различия между Netscape и MSIE, но много по-съществени са тези между по-старите и по-новите версии. Употребата на определени команди на JavaScript води до появата на съобщения за грешки и объркване на работата на браузъра, ако неговата версия е по-ниска от тази, за която е предназначена и изпробван скриптът.

Създаването на JavaScript (т. е. на самия програмен код) е много интересно и приятно занимание, което обаче изисква познаване на основите на обектноориентираното програмиране (ООП) и на самия език JavaScript. По тази причина по-често се среща употребата и включването в страниците на готови скриптове, които се разпространяват безплатно в Интернет. Все по-често се срещат уебстраници, в които може да се попълни информация в определени за целта полета. Тази информация може да се изпрати към сървъра и на базата на това да се получи резултат. Става дума за формуляри за регистрации, за обратна връзка, за търсене, за включване в mailing list за пазаруване (вече и България) и т. н. Ще наричаме тези елементи форми.

Създаването на форми има две страни – еднакво важни и отговорни и същевременно много различни. Първо е необходимо да се включат съответните полета за попълване и бутони за изпращане на данните в уебстраницата. След това трябва да се направи и съответната програма, която ще получава данните от заявката, ще ги обработва и ще връща резултата към потребителя.

1. Създаване на полетата в уебстраницата се осъществява с включването в HTML на определени тагове. За да се обяви начало на формата, се използва тагът <FORM>, следван от таговете задаващи съответните полета, както и текстовото и графично съдържание около тях. Най-накрая тагът <FORM > трябва да бъде затворен </FORM>. По-долу е даден списък с таговете, създаващи определени елементи на формите:

<INPUT TYPE="TEXT"> - създава в уебстраницата поле с размери, определени от стойностите на параметрите ROWS И COLS, в което потребителят може да въведе информация.

<TEXTAREA> е едно от изключенията при таговете за форми и изисква употребата и на затварящ таг: </TEXTAREA>.

<INPUT TYPE="CHECKBOX"> - създава поле за поставяне на отметка.

<INPUT TYPE="BUTTON" VALUE=""> - създава бутон с надпис, зададен в параметъра VALUE.

<INPUT TYPE="SUBMIT"> - създава бутон за изпращане на попълнената форма.

<INPUT TYPE="RESET"> - създава бутон за изчистване на всички полета във формата.

<INPUT TYPE="PASSWORD"> - създава поле за въвеждане на парола.

<INPUT TYPE="IMAGE" SRC=""> - включва на съответното място графичен файл, описан в параметъра SRC.

<INPUT TYPE="FILE"> - създава поле, в което потребителят може да избере файл от твърдия диск, който да бъде изпратен заедно с данните от формата.

<SELECT> - създава падащо меню. Отделните елементи в него се задават с тага <OPTION>, като накрая таг <SELECT> се затваря със </SELECT>.

2. Създаването на програмата, приемаща и обработваща данните от формата, се осъществява предимно на езика за програмиране Perl или на ASP.

В много редки случаи обработката на формата може да бъде направена с JavaScript и потребителят да получи резултата от това веднага.

На езика Perl се правят т. нар. server side scripts – скриптове, които работят на сървър и носят също така общото название CGI – Common Gateway Interface. ASP е съкращение от Active Server Pages (активни сървърни страници) и представлява начин да се съвместят програмните HTML – кодове в един и същи файл. ASP – страниците могат да се поддържат само на MicroSoft NT сървъри. За създаването на Perl – скриптове, както и на ASP са необходими много повече знания, отколкото за JavaScript. Повечето от фирмите освен хостинг предлагат и определени стандартни програми безплатно, което е необходимо уебдизайнерът да подготви единствено уебстраницата с формата, спазвайки определените изисквания на конкретната програма. На практика в повечето случаи се налага подготвянето на специфичен скрипт, като за целта се търсят програмисти, запознати с възможностите на сървърите.

15. Аплети на JAVA

Аплетите на Java представляват самостоятелни програми, написани обикновено с помощта JDK (Java Development Kit), които тъй като тук става дума за програмиране, с което опитният специалист може да постигне много. За съжаление аpletите на Java имат основния недостатък, че са много тромави, бавно се стартират и доста често водят до срив на брауъра. По тази причина те се употребяват рядко и тяхното използване не е препоръчително.

Други възможности за изграждане на уеб страници във визуална среда

WYSIWYG редактори (What You See Is What You Get – Получаваш това, което виждаш), в който кодът се генерира автоматично. В процеса на работа лесно се превключва между прозореца за редактиране и прозореца за изглед на крайния резултат. Удобството при използването им се дължи на факта, че с тях се работи интуитивно, разполагат с богат набор от шаблони и много сложни задачи като достъп до база данни, създаване на формуляри, електронна търговия и дискуссионни групи, са автоматизирани. Най-популярни продукти са Adobe Dreamweaver, Microsoft Expression Studio, Incomedia WebSite X5 и др. Всеки от тях има силни и конкурентни възможности, които допадат на едни или други потребители, както и известни проблеми (свързани със съвместимостта с брауърите, пльгин - модулите, експортирането в Интернет, индексирването в търсещите машини и др.), разрешаването на които е цел в разработването на нови версии. Съвместимостта между версиите и възможността за адаптиране към новите стандарти са едни от най-търсените предимства за избор на конкретен софтуер в разработката на съвременните уеб сайтове. За първоначално запознаване с технологията за разработване на уеб сайт с учебна цел ще се използва уеб редактора WebSite X5, Free 10, който е от типа WYSIWYG. Това е визуален софтуер, в който чрез последователни стъпки ще осъществява проектиране, оформяне на съдържателната част и публикуване на професионален уеб сайт. Програмния код, който се генерира автоматично, е създаден с XHTML.

Не е задължително да разбирате Java, за да поставите Java аплети върху страниците си. Можете да се сдобиете с аплети от много места и след това да използвате HTML елемента APPLET, за да ги поставите, където желаете. В тази глава ще видите къде могат да се намерят аплети в Интернет (често безплатно) и как да ги поставите върху страниците си. Изненадващо, но можете интензивно да използвате Java върху страниците си без да е нужно да знаете езика Java.

```

<HTML>
<HEAD>
<TITLE>Hey it's Java </TITLE>
</HEAD>
<BODY BGCOLOR=WHITE>
<APPLET CODEBASE="classes">
CODE= "HelloWorld.class"WIDTH=200
HEIGHT=50>
</APPLET>
</BODY>
</HTML>

```

За да добавите аplet към вашата Web страница:

1. <APPLET CODEBASE="classes" CODE -> ="HelloWorld.class" WIDTH= -> 200 HEIGHT=50>

Елементът APPLETT е стандартна част от HTML. Атрибутите, които използваме тук са CODEBASE (директорията върху сървъра, където се намира аплета), CODE (името на класа на файла аplet), HEIGHT и WIDTH (които указват на брауъра, колко място ще заеме аплета върху страницата).

В този случай ние викаме аплета HelloWorld. Ширината е 200, а височината е 50.

2. </APPLET>

Това затваря елемента APPLETT.

Съвет: Ако имате аплети, използвани от множество страници върху вашия сайт, използвайте атрибута CODEBASE на елемента APPLETT, за да укажете аплета в обща директория. По този начин, ако смените аплета по-късно, няма да е необходимо да промените HTML файловете.

Ето код за изобразяване на съобщение за неподдържащи Java брауъри:

```

<HTML>
<HEAD>
<TITLE>Hey, it's Java</TITLE>
</HEAD>
<BODY BGCOLOR=WHITE>
<APPLET CODEBASE="classes" CODE=
"HelloWorld.class" WIDTH=200 HEIGHT=50>
<H1> If you had Java, you'd be seeing a
Cool applet now</H1>
</APPLET>
</BODY>
</HTML>

```

Въпреки че понастоящем повечето брауъри поддържат Java, някои хора използват стари брауъри, а някои сърфират по мрежата с изключена Java.

1. <H1> If you had Java, you'd be -> seeing a cool applet now</H1>

Сигурно предполагате, че това изглежда като нормален HTML и сте прави - това е така. Тъй като той е в елемента APPLETT, това се показва само ако потребителя не може да изобрази Java аплета.

За да позиционирате Java аплет:

■ `<APPLET CODEBASE="classes" CODE ->="HelloWorld.class" WIDTH=200 HEIGHT=50 ALIGN=RIGHT>`

„Hello World“ ще се визуализира в дясната страна на прозореца. Това се получава, защото атрибутът ALIGN на елемента APPLET е настроен на RIGHT.

■ `<APPLET CODEBASE="classes" CODE ->="HelloWorld.class" WIDTH=200 HEIGHT=50 HSPACE=100 VSPACE=100>`

Аплетът Hello World е заобиколен от 100 пиксела разстояние бяло пространство от всички страни, което се дължи на настройките на атрибутите HSPACE и VSPACE.

Съвети

■ Други валидни стойности за ALIGN са LEFT, RIGHT, TOP, TEXTTOP, MIDDLE, ABSMIDDLE, BASELINE, BOT - TOM и ABSBOTTOM.

■ HEIGHT, WIDTH, ALIGN, HSPACE и VSPACE са атрибути, които елементът APPLET използва подобно на IMG. За да научите повече за тези атрибути, виж книгата на Елизабет Кастро, HTML 4 за World Wide Web.

■ Намиране на аплети в Интернет

Има няколко източника в Интернет, откъдето можете да се снабдите с безплатни или евтини аплети. Един от най-старите и добре известни е архива Gamelan, на адрес <http://www.gamelan.com>. Преди да навлезете в мъчението да пишете собствени аплети, няма да ви заболи ако проверите дали някой друг не е открил точно това колело.

■ Има и много други места в Интернет, където можете да намерите аплети.

Изграждане на потребителски интерфейс

Една от главните причини да поставите Java аплет върху страниците си е възможността да взаимодействате с потребителя. За да взаимодействате с потребителя трябва да имате *потребителски интерфейс*, полета, в които потребителят може да въвежда информация, полета за маркиране, радио бутони и т.н.

15.1. Защита на страницата с парола:

```
<HTML>
<HEAD>
<TITLE>Listing</TITLE>
</HEAD>
<BODY BGCOLOR=WHITE>
<APPLET CODEBASE="classes"
CODE="Applet1.class"
WIDTH=500 HEIGHT=300>
</APPLET>
</BODY>
</HTML>
```

Често срещано е желанието да се защити някоя страница или област от Web сайта Ви от очите на случайни посетители. **Листинг** и **Аплет** показват прост Java аплет, който изисква парола. Ако се въведе правилната парола се зарежда защитената страница.

За да защитите страница с парола:

```
1. try {
passwordURL = new URL("http://->/www.chalcedony.com/java/
}
catch (MalformedURLException -> err) {
passwordURL =
```

```
->this.getDocumentBase();
}
```

Първо се създава нова URL променлива, на която се присвоява адреса, който ще се зареди ако е въведена правилната парола. В случая, URL, на който искаме да отидем е <http://www.chalcedony.com/java>. Конструкцията try/catch е задължителна в случай, че не се въведе валиден URL, което ще предизвика MalformedURLException грешка. В този случай ще зададем URL да бъде адреса на текущата страница.

```
■      add (new Label (-"Password"));
passwordField =
-4 new TextField(10);
passwordField.setEchoCharacter -> (*');
add(passwordField);
```

Първият ред добавя етикета „Password“ на екрана. След това създаваме текстово поле с дължина 10 наречено passwordField, което ще изобразява звездички (*), когато в него се въвежда текст, и го изобразяваме на екрана.

```
■      if (badPass) {
g.drawString("Incorrect ->Password - try again", 200, ->40);
}
```

Ако се активира флага за неправилна парола се изобразява реда „Incorrect Password - try again“.

```
■      if (e.target instanceof -4 TextField) {
if (passwordField.get -4 Text().equals("peachpit")) -4 { getAppletContext().
-4 showDocument -4 (passwordURL);
}
```

Този аплет защитава следната страница като изисква правилна парола:

```
import java.awt.*
import java.applet.Applet;
import java.net.*;
public class Applet1 extends Applet {
    TextField passwordField;
    Boolean badPass=false;
    URL passwordURL;
    Public void init () { setBackground (color.white);try { passwordURL=new URL
(http://www.chalcedony.com/java/");}
catch (MalformedURLException err) { passwordURL=this.getDocumentBase();}
add (nes Label (“Password”)); passwordField.setEchCharacter (“*’);
add (passwordField);
}
```

```

Public void paint (Graphics g) {
If (badPass){ g.drawString (Incorrect pasword – try again”, 200, 40);} }
Public boolean action(Event e, Object arg) {
If (e.target instanceof textField){
If (passwordField.getText ().equals
(“peachpit”)) {
getAppletContext ().showDocument
(password URL);
} else {
hadPass=true;
repaint();
}
return true;
}
return false;
}

```

Създаване на анимации с FLASH

От дълго време Flash е добре известен като програма за създаване на анимация - отчасти, защото се справя изключително добре в анимирането на рисунки. Flash се използва често за уеб анимация, защото позволява създаването на висококачествени анимации с малък файлов размер, което е идеално за онлайн съдържание. В последните години Flash се използва за създаването на анимации за излъчване под формата на телевизионни реклами например. Причината Flash да бъде откриван на толкова много места, е това, че е чудесен инструмент за създаване и анимиране на векторни графики. **Анимацията е поредица от изображения**, създаващи впечатление за движение при разглеждане в бърза последователност. Във Flash, тези изображения се формират от съдържание, поставено в кадри от времедиagramата (Timeline). За да създадете анимация или движение във вашите FLA файлове, бихте могли да използвате рисунки, снимки или дори графики, генерирани чрез ActionScript.

Съществуват няколко различни начина за създаване на специални движения и анимации във Flash. Имате възможност да създавате *движение по междинни фази* на фигури и да създадете *покадрова анимация*. *Междинните фази* се постигат чрез изчисленията, извършвани от Flash за анимиране, преместване или промяна на свойствата или формата на даден обект от SWF файла. Началото и края на дадено преобразуване се задават посредством ключови кадри. Когато приложите движение по

междинни фази върху набор от ключови кадри, Flash самостоятелно създава движението като запълва промените, които се случват между двата кадъра. Промяната на фигури по междинни фази се използва за промяна на векторна линия или форма, като например промяната на квадрат в кръг. Движението по междинни фази се използва и за промяна на свойства, например местене на обекти на сцената или тяхното скриване и появяване. В настоящия урок ще създадете както движение, така и промяна на фигури по междинни фази.

Покадровата анимация представлява по-традиционен начин за създаване на анимация. Вместо да карате Flash да създава междинни фази посредством автоматични изчисления, вие рисувате ръчно всяка фаза на движението, което Ви позволява да създавате по-сложни ефекти, особено когато автоматичната функция не Ви дава възможност да създадете вида анимация, която искате. Анимациите от тип кадър по кадър обикновено отнемат повече време за създаване, а и добавят повече размер към SWF файла.

Имате възможност да създавате и скриптова анимация посредством ActionScript и (обикновено) филмови клипове. Съществува част от ActionScript, наречена Drawing API, която се използва за скриптово управление на линии и форми на сцената, и която може да бъде използвана по много сложни начини. Въпреки че това е извън предмета на настоящата книга, определено си заслужава да разгледате темата, ако и когато, желаете да научите повече за ActionScript.

Както разбрахте току-що, движението по междинни фази се използва за промяна на свойствата на даден обект в рамките на определен брой кадри. Имате възможност да променяте свойства като позиция, размер, прозрачност, цвят и ротиране, както и да задавате движение по междинни фази по траектория. Винаги трябва да променяте обекта в символ преди да използвате тази техника за негова инстанция. Това ще Ви помогне да запазите файловия размер малък, когато експортирате SWF файла.

Файловият размер се запазва малък, защото Flash има възможност да използва многократно елементи от библиотеката (Library) вместо да пресъздава данните при всяко срещане във времедиagramата (Timeline).

Имате възможност да използвате промяна на форма по междинни фази за сурови данни, но не и за инстанция или битмап. Първо трябва да ги “разглобите” на сурови данни като изберете Modify > Break Apart

Какво е уебсайт по начин, по който може да не си мислил?

Уебсайтът е сбор от уебстраници. Уебстраниците са документи. Също като книжката с инструкции за кафемашината или гаранционната карта. Като документи, в тях има стандарти и правила. Ако не спазваме правилата, нямаме качествен документ, уебстраница и уебсайт. Това е академичното дефиниране.

Уебсайтовете са комплекс от технологии, но този комплекс и резултат са следствие от нуждата за абстрактно мислене, извън кутията на неразбираемите за хората технологии. Уебсайтът, като продукт на нашия развит живот, е комбинация от всичко емоционално, мисловно и инстинктивно, което сме. Той е проекция на нашия напредък и нужда за изява пред обществото, от което така силно имаме нужда и с което сме обвързани ежедневно с битови и бизнес цели.

Също както, когато едно дете нарисова картинка, и психолозите могат да видят неговото състояние, нужди, тревоги и етап на развитие, уебсайтът разкрива състоянието, нуждите, тревогите и етапа на развитие на една фирма.

Уебсайтът е проекция на качества, начин на мислене и опит.

Какво издава уебсайтът за една фирма?

Хората изразяват себе в заобикалящата ги среда със специфично поведение, начин на говорене, облекло. Реагират на някои теми по определен начин, за други - имат друга реакция. Понякога преднамерено, понякога несъзнавано, хората демонстрират противно на вътрешното си разбиране поведение, мнение и действия.

Всичко това е изразяване на собствената личност. Повечето хора имат нужда да покажат вътрешния си свят чрез познати модели на поведение. Останалите, които нямат желание да се изразяват на околните, имат нужда да покажат на себе си, че не са зависили от публиката.

Уебсайтът за една фирма е един от множеството безбройни начини за изразяване.

Независимо дали е пасивно или активно, съзнавано или несъзнавано, отношението към фирмения уебсайт, съществуващ или все още не, е израз на мнение и отношение към околните.

Какво може да види всеки човек в уебсайта на една фирма? Безброй неща. Зависи от степента на вглеждане, адекватна преценка и допълнително проучване за първоначалните впечатления. Всеки може да разбере нещо различно. Вярно или невярно, коректно в определена степен или грешно в някои отношения.

Изолирайки основни аспекти на демонстриране на фирмено състояние чрез фирмения уебсайт, получих следния списък:

- Бизнес качества и осигуряване на нуждите на клиентите на фирмата
- Изразяване на имидж и поглед към качеството
- Финансово състояние
- Отделяне на време за уебсайта
- Наличие на персонал
- Компетенция в сферата на дейност
- Способност за комуникация с клиенти
- Партньорски отношения
- Публична дейност
- Скорост на фирмено развитие.

16. Системи за управление на съдържанието (CMS)

Системата за управление на съдържанието (Content Management System), наричана още админ панел или контрол панел, е задължителен компонент на всеки съвременен сайт. Той представлява вътрешната, защитена (вход с потребителско име и парола) част от сайта.

Добрият админ панел дава възможност на собственика на сайта сам да актуализира съдържанието на сайта и да управлява голяма част от неговите функционалности. Така необходимостта от намеса от страна на разработчика възниква доста по-рядко и обикновено във връзка с добавяне или корекции на техническо ниво.

Ето основните неща, които можете да искате да се изпълняват чрез админ панела на Вашия сайт:

- Редакция и форматиране на текстове;
- Автоматично оразмеряване на снимките и илюстрациите (при дефинирани точни размери и по колко варианта ще има всяка една снимка);
- Добавяне на нови страници и категории;

Всички останали функционалности в сайта, част от които ще разгледаме в тази глава, също се интегрират в админ панела.

Обикновено уеб агенциите използват собствена разработка за CMS, която се адаптира за нуждите на конкретния проект. Освен това е възможно използването на готови CMS решения, обикновено софтуер с отворен код, като Joomla, Mam boo, Wordpress, OSCommerce (за е – магазин специално) и др. Интегрирането на готова система е по-евтино, но е възможно да доведе до известни компромиси с дизайна и лекотата на сайта. Кой подход ще бъде избран, зависи от изискванията на конкретния проект и от възможностите, които дава неговият бюджет.

Макар и не съвсем критични за съществуването на сайта, има три важни допълнителни елемента на CMS-а, които е препоръчително да получат специално внимание:

Дизайнът и ползваемост на админ панела. Тъй като админ панелът е по-сложен от самия сайт и с него оторизираният потребител работи всекидневно, е още по-важно да има приветлив изглед, интуитивна архитектура и допълнителни пояснителни бележки, където е необходимо. Админ панелът ще се използва ежедневно или често, затова е важно да има всички необходими за поддръжката на сайта възможности, добре подредени и видими. А често се оказва, че сложността на CMS системите отблъсква администратора от регулярна работа по сайта, което не е в услуга на бизнеса.

Сигурността на админ панела. Един „случаен“ достъп на външен човек може да доведе до подмяна на съдържанието на сайта и до изтичане на вътрешна информация, която може да е обект на търговска тайна или да представлява лични данни за потребителите на сайта или клиентите на компанията.

Документацията и инструкции за използване. Случва се служителите, които отговарят за сайта, след като той заживее онлайн да се сменят, затова са необходими допълнителни обучения и изготвяне на Ръководство за употреба.

Статии и новини

Новините обикновено са най-често обновяваната част от съдържанието на един уеб сайт. Всеки бизнес, т. е. всеки сайт, има какво да каже и да сподели. Това могат да бъдат както корпоративни новини, така и промоции, съобщения, насочени към крайните потребители и медиите. Обикновено те с представят в следния формат: всяка новина има заглавие, водеща снимка, дата, резюме и пълен текст, като в сайта новините излизат в хронологичен ред – най-новата е най-отгоре в списъка с всички. При натрупване на повече новини се обособява „Архив“. Там публикациите могат да бъдат подредени и намерени по дата, тема или друг критерий, определен предварително. Когато се очаква по-голям обем съдържание и покриване на по-широк спектър от теми, е добре да се обособят рубрики или категории.

Същият, или много подобен формат, използван при „Новини“, се използва и за публикуване на статии. Статиите се групират по-често не по хронологичен ред, а по теми. Като цяло, публикуването на статии в корпоративен сайт е специфично и се прилага само при някои бизнеси.

Допълнително към всяка новина или статия може да се създаде възможност за прикачване на файлове (документи, таблици), добавяне на изображения (снимки, схеми), видео или аудио файлове и др.

Добра практика е новините да бъдат изведени в началната страница на сайта – заглавията, първите едно – две изречения и снимката. Това не само ще улесни потребителя, но и ще допринесе за по-доброто индексирание на сайта в търсачките

заради натрупването на повече ключови думи и текст, както и заради честотата на актуализация.

Коментари

Добавянето на възможност потребителите да оставят своето мнение към статиите в сайта под формата на коментари е може би ключовата разлика между традиционните и онлайн медиите. Ето защо повечето онлайн медии и сайтовете на онлайн обществата включват тази функционалност. Рядко се случва фирмен сайт да отвори страниците си за дискусии – по-скоро ще ползва формата на корпоративния блок или форум специално за тази цел. Чрез коментарите в сайта могат да попаднат различни неща – автентични мнения и впечатления (позитивни и негативни), комерсиални послания, спам. За да се ограничи възможността от злоупотреба с „отворената линия“, каквато представляват коментарите онлайн, обикновено се изисква идентификация на коментирация (има, имейл адрес) и задържане на публикуването до одобрение от администратор на сайта.

Коментарите са важен елемент, когато целта е създаване на общност и лоялна аудитория на сайта, но за да изпълнят предназначението си, е важно да се осигури достатъчно ресурс за поддръжката им.

Архив

Когато сайтът или негов раздел предлагат често допълвано съдържание и за целите на проекта е важно да се проследи промяната и натрупването във времето, тогава следва да се обмисли по какъв начин да бъде представен онлайн този архив. Най-често онлайн се пази архив на новини, публикации, съобщения. Той може да бъде организиран по месеци, години, теми, автор, тип съдържание (текст, фото, видео) и др. Добре е точните параметри на архива да бъдат уточнени при планирането на сайта и изготвянето на информационната архитектура. От гледна точка на поддръжката онлайн архивът не изисква специални усилия, тъй като се генерира автоматично.

В тази връзка е добре да има фотоархив и видеоархив, които архиви също да бъдат добре организирани.

Зони с ограничен достъп

Един сайт често има за задача да обслужва различни целеви групи. Обикновено те имат нужда от различно съдържание и функционалност. Например на лекарите можем да покажем пълната информация за всяко лекарство, докато съществуват законови ограничения при комуникацията директно към пациенти на лекарствени продукти, които се купуват само с рецепта и по лекарско предписание.

За да се реализира подобно разграничение, е необходимо потребителят да се регистрира с потребителско име и парола в сайта. По този начин той се идентифицира в системата и тя прави проверка дали конкретният потребител има права за достъп до защитените зони.

Регистрацията на нов потребител може да се прави директно от него самия – например когато се записва да използва услугите на уеб поща, сайт за запознанства, онлайн магазин, или от администратора на сайта – когато броят на потребителите не е голям и те са предварително известни (дилъри, служители и др.).

Регистрацията на потребители е важен функционален модул, който следва да бъде подробно дефиниран при започване на работата по сайта. Основните въпроси, чиито отговори се търсят на етапа на планирането, са:

Кои ще бъдат полетата, изисквани за регистрация на потребителя?

Имейл, парола, телефон, адрес и др. Правилото е колкото по-малко задължителни полета, толкова повече регистрации.

Как ще се идентифицират регистрираните потребители? С име и парола, имейл и парола и др.

Какво ще виждат и правят потребителите, след като влязат в системата?

Кои секции, какво съдържание, какви допълнителни функционалности; едно и също за всички или специфично за всеки потребител и др.

Какво ще виждат и правят потребителите, след като влязат в системата? Кои секции, какво съдържание, какви допълнителни функционалности; едно и също за всички или специфично за всеки потребител и др.

Как ще се възстановява забравена парола? Дали автоматично ще се изпраща по имейл, или администраторът ще я възстановява.

Кой и как ще има достъп до данните? От изключително значение е кой ще има достъп до информацията и данните на регистрираните потребители, както и как ще се съхраняват.

Какви справки ще се правят? Ще се проследява ли дейността на потребителите след входа – колко често влизат, каква информация разглеждат или свалят и други.

За да се насърчат свободните регистрации, е добре да се обясни на потребителя какво ще се получи и защо е необходимо да предостави исканите данни. Линк „Защо да се регистрирам?“ върши чудесно тази работа – нека не го пропускате.

Системата за управление на уеб съдържание (CMS) е софтуер, който подпомага потребителите в процеса на управление на съдържанието, а именно неговото създаване, публикуване, редактиране и модифициране. Уеб съдържанието е всяко текстово, визуално или звуково съдържание, което е част от уеб сайт, т. е. това са документи, данни, приложения, услуги, изображения, аудио и видео файлове, лични уеб страници, архивирани e-mail съобщения и други. Системите за управление на съдържанието обхващат цялостния жизнен цикъл на уеб страниците на сайта – от осигуряването на прости инструменти за създаване на съдържанието до публикуването му и архивирането му. Те осигуряват и възможността за управление на структурата на уеб сайта, външния вид на публикуваните страници, навигацията, достъп до възможности за търсене и т. н.

CMS е динамичен уеб сайт за създаване и промяна на съдържанието чрез администраторски панел. Системата за управление на съдържанието улеснява съвместното създаване на документи от различни потребители по едно и също време. В зависимост от функционалността, която предоставят, системите за управление на уеб съдържание биват:

***системи за управление на корпоративно съдържание** (Enterprise content management systems, ECMS) – използват се за събиране, управление, съхраняване и доставяне на съдържание /документи/, свързано с процесите на организацията;

***системи за управление на учебно съдържание** (Learning Content Management Systems, LCMS) - използват се за разработване, управление, поддържане и доставяне на учебно съдържание и

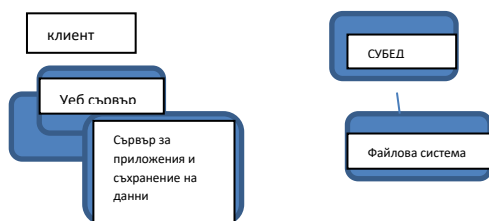
мултимедия;

***системи за управление на компонентно съдържание (Component content management system, CCMS)** -специализирани средства в създаването на документи от съставни части. Тези компоненти могат да бъдат използвани в друг документ или в множество документи. Всеки компонент се съхранява само един път в системата за управление на съдържанието и се използва многократно в рамките на един документ или в множество документи. Това гарантира, че съдържанието е в съответствие навсякъде в

документацията;

***системи за управление на уеб съдържание (Web content management systems, WCMS, Web CMS или WCM система)** - системи, подпомагащи поддържането, контрола и промените в съдържанието на уеб сайт.

Архитектура на системата за управление на уеб съдържание: В основата на CMS технологията стои трислойна архитектура от типа клиент/сервър, изградена от – обработка на данни от клиента (уеб браузър); уеб сервър; сервър за приложения и съхранение на данни. За разлика от традиционната архитектура от две нива, при системите за управление на съдържанието присъства сервър за приложения като посредник между клиента и хранилището на данни. В системата има две хранилища. В първото хранилище /обикновено релационна система за управление на бази от данни/ се съхраняват всички данни, които са публикувани на Интернет страницата. Във второто /обикновено файлова система/ се съхраняват елементите – шаблон, графики и др.



Фиг.18 Трислойна архитектура на системата за управление на съдържание

Последователността от действия при трислойната архитектурата може да се опише така:

1. Потребителят чрез уеб браузъра изпраща заявка за определена уеб страница към уеб сервър
2. Уеб сервърът получава заявката, извлича файла и го подава на сервъра за приложения
3. Сервърът за приложения започва да изпълнява програмния код във файла и се свързва с хранилищата на данни и му изпраща заявка
4. Хранилищата получават заявката, обработват я и изпращат резултатите обратно на сервъра за приложения
5. Сервърът за приложения довършва програмния код, което обикновено включва форматиране на резултатите в HTML. След това връща получения резултат на уеб сервъра
6. Уеб сервърът изпраща на потребителя само резултата във вид на HTML файл

По този начин сервърът за приложения играе ролята на стандартизирана платформа за доставяне на динамично съдържание. Сервърите за приложения могат да бъдат много

и връзката помежду им е чрез уеб сървър. Сървърните приложения са написани на скриптовите езици или инструменти като PHP, JSP, ASP, Python, Coldfusion, Perl и др., които обработват данните и ги преобразуват във визуално съдържание.

16.1. Най-известни системи за управление на уеб съдържание

Системите за управление на съдържание отделят графичния дизайн от самото съдържание чрез използване на т. н. „тема“. Тя контролира графичните елементи,

Характеристика	WordPress	Joomla	Drupal	Plone
Лесна инсталация	+++	+++	+++	+
Лекота при създаване на елементарен сайт	+++	++	+	+
Лекота при създаване на сложен сайт	+++	+++	++	++
Лесна употреба от администратора на съдържание	+++	++	++	+++
Графична гъвкавост	+++	+++	+++	+++
Достъпност и оптимизация за търсачки	++	++	+	+++
Структурна гъвкавост	++	++	+++	+++
Сигурност	+	++	++	+++

шрифт и стилове за навигация. Темата може да се избере предварително, да се добави готова допълнително или да се разработи персонализирана такава. Системите за управление на съдържание с отворен код са предназначени да бъдат модулни. При изтеглянето и инсталирането на системата тя получава набор от функции, които група от разработчици е решила, че трябва да е базово ниво на функционалност, наричано още „ядро“ в системата. То се надгражда чрез т. нар. add-on модули, които осигуряват допълнителна функционалност на системата. Популярни системи за управление на съдържание са WordPress, Joomla, Drupal, Plone...

Таблица 8: Сравнителна таблица на WordPress, Joomla, Drupal, Plone
/+задоволителна, ++ много добра, +++ отлична/

WordPress е отличен избор за сравнително малки, елементарно подредени уеб сайтове. Това е най-лесната система за инсталиране и изучаване, лесно се поддържа и актуализира. Има много предварително зададени графични теми на разположение, адаптирането им към специфични нужди може да бъде много лесно, ако темата позволява да се избере свой собствен цвят и добавяне на собствено лого. Системата притежава ограничена поддръжка за разграничаване на потребителски роли.

Joomla е система за управление на съдържанието, която е относително лесна за инсталиране и настройка. Допълнителните модули на системата поддържат широко разнообразие от функционалности- от директории до колички за пазаруване и функции за общуване, осигурявайки солидна основа за много различни видове сайтове. Има ограничение до една категория на статия, което може да доведе до необходимостта от дублиране, ако е необходимо да бъде включена в повече от един раздел на сайта.

Drupal е гъвкава и мощна система и е невероятен избор за по-сложни уеб сайтове. Поддържа голямо разнообразие от структури за сайт. Има особено силна поддръжка за

Web 2.0 и функции за общуване, включително съдържание добавено от потребителя. Административните екрани за конфигуриране на сайта имат огромен брой опции и настройки, което ги прави по-трудни за разбиране. Drupal не е особено силна при контрола на работния поток и не може да бъде най-добрият избор за организации, които искат да разполагат с много хора с различни роли за собственост върху съдържанието.

Plone – е мощна и стабилна система за управление на съдържанието, която е подходяща за организации с много комплексни нужди. Тя се използва от главните печатни издания и големите фирми. Системата предлага висока степен на гъвкавост и контрол и поддържа почти безкрайно сложни работни потоци. Инструментите на администраторите на съдържанието са с дружелюбен интерфейс и е лесно за административни скретари да актуализират основния текст, изображения и странични площи. Да разучава Plone за изграждането на един уеб сайт не е удачно. По-подходящо е да се наеме консултант, който вече е запознат с нея и има познания за работа с езика Python.

Предимства при използването на системите за управление на уеб съдържание:

- Намаляване на разходите по поддръжка на уеб сайта
- Добре организиран процес по създаване на съдържанието
- По-бързо създаване на нови страници и тяхното обновяване
- По-голяма съгласуваност и сигурност
- Подобряване на навигацията и гъвкавостта на сайта
- Поддръжка на децентрализирано създаване на съдържание

Инсталиране на система за управление на уеб съдържание - Системите за управление на съдържанието и скриптовете (script) са написани на програмен език (PHP, Java, Python и др.). Изпълняват се на уеб сървър, а не в брауъра и изискват специализиран софтуер, наличен на уеб сървъра. Необходимостта от наличието на софтуер като Apache, PHP, MySQL е свързана с факта, че за работата на системите за управление на съдържанието са необходими уеб сървър, който да управлява приложението (Apache), PHP интерпретатор (или друг в зависимост на какъв език е написан CMS) и уеб сървър за управление на базата данни (MySQL). Примерни ftp клиенти: FileZilla, Cyberduck, SmartFTP, WinSCP... Голяма част от компаниите, предлагащи хостинг услуги, разполагат с автоматични инсталатори като Softaculous, Fantastico, Magento и др., с помощта на които може да се инсталира бързо и лесно WordPress или друга система за управление на съдържанието. Това е вид софтуер, който позволява създаване, публикуване и редактиране на уебсайт. Целта на такива системи е да улеснят изграждането на динамичен уеб сайт с възможности за лесна промяна в съдържанието по всяко време не само от програмисти – уеб дизайнери, а и от хора без специализирани технически знания. Тази технология цели да се намали максимално нуждата от писане на код. Тя позволява на ползвателите да се съсредоточат върху създаването и редактирането на самото съдържание, а не върху ползваните технологии. Тези технологии се използват за съхраняване, контролиране и публикуване на новини и съобщения, упътвания, рекламни материали и др. Чрез тези системи:

- Лесно и бързо се публикуват документи и материали;
- Идентификация на потребителите и разпределянето им в групи с различни права за достъп;
- Възможността да се дава определено ниво на достъп до категории от съдържанието или до отделни документи в него;
- Съхранение онлайн на всякакви видове материали – видео, изображения, документи, програми и др.;

При избор на CMS, трябва да се обърне внимание на:

- Броя на лицата, използващи системата;
- Сложност на дизайна – ако всички страници използват подобно оформление, тогава системата за управление на съдържание може много по-лесно да бъде внедрена.

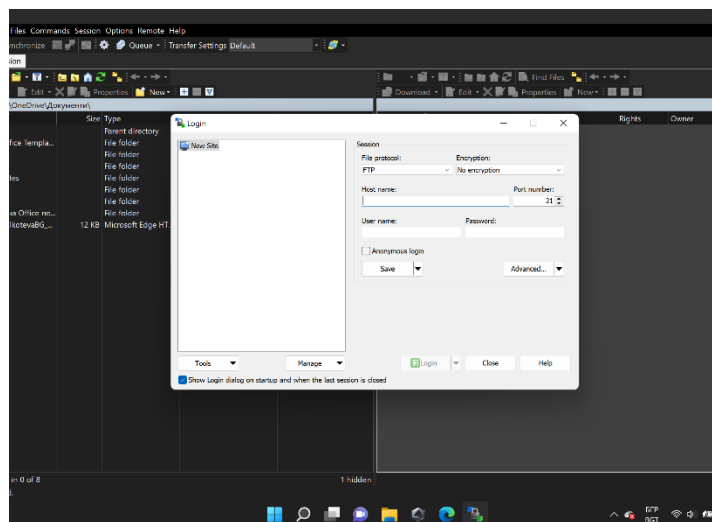
Препоръчвам Ви WordPress, само инсталацията може да Ви затрудни, а тя всъщност отнема пет минути.

Тази система може да се ползва по два начина – директно в интернет, в сайта wordpress.com или за индивидуална работа – пакетът се инсталира на собствен компютър, който изпълнява функциите на сървър.

За да инсталирате на Вашия компютър – сървър, трябва да инсталирате на него следният софтуер – Apache or Nginx, PHP, MySQL or MariaDB.

Голяма част от фирмите Dom.bg, ICN.bg предлагат хостинг услуги, разполагат с автоматични инсталатори, чрез които системата се инсталира лесно, като се следват стъпки и те най-вероятно ще Ви помогнат за това.

Друг начин да извършите инсталация е да качите файловете си на сървъра посредством FTP клиент. Но трябва да имате хостинг, който да отговаря на изискванията. След като имате такъв сървър и сте си инсталирали необходимия софтуер, в тази форма попълвате данните за уеб сайта, хост, домейн, парола.



Фиг. 19 Програма за прехвърляне файловете от локалния компютър на уеб сървър

Всяка редакция става на момента, в който се преместят файловете от локалния Ви компютър в папката, качена уеб сървъра. Страницата се рефрешва и обновява. Самата редакция става по различни начини, ето пример на редакция на кода, където може и да се вграждат заготовки и други кодове. Но след това отново файлът – уебстраницата трябва да се качи на сървъра в определената папка за уеб сайта.

```
index.htm
<body> <div>
726
727
728
729
730
731
732
733
734
735
736 <tr>
737
738
739
740
741
742
743
744
745
746
747
748 <td colspan="3" valign="bottom" align="center" style="height:240px">
749 <span style="letter-spacing: 3px;word-spacing:10px">
750
751
752
753
754 <marquee class="cg" scrolldelay="0" scrollamount="4" height="26px" style="width: 1343px; color: #511104; font-size: small;">ПЛОВДИВСКИ&nbsp;университет&nbsp;ПАСИЙ&nbsp;ХИЛЕНДАРСКИ!
755 <br>
756
757
758
759
760
761
762
```

Фиг. 20 Програма за редакция на код

Ето и стъпките за същинското инсталиране на WordPress:

1. Изтеглете WordPress
2. Разархивирайте пакета. Качете папката в директорията на сървъра, която сте определили за бъдещ веб сайт. Може да използвате контролния панел или FTP клиент.
3. Създайте MySQL база данни през контролния панел на хостинга: от раздела Databases избираме MySQL Databases. В полето New Database посочваме име на новата база данни и кликваме върху бутона Create.
4. Стартирайте в брауъра URL адреса, който отговаря на директорията. Отваря се форма, в която трябва да напишете името на базата данни, потребителското си име и паролата. Тях ще намерите в настройките на акаунта от раздела Accounts на Вашия профил. В следващата форма задавате заглавие на сайта, име и имейл на администратора.

Следва да натиснете бутона Инсталиране и при успешна инсталация, ще се появи прозорец за логване. Може да редактирате на български и всеки от тук насетне може да работи по веб сайта.

Администраторски панел – достъпът до административната част на системата за управление на съдържанието WordPress се осигурява от директорията wp-admin.

16.2. Сигурност при системите за управления на съдържание

Системите за управления на съдържане с отворен код (open source) са едни от най-често атакуваните веб сайтове в мрежата. Тъй като кодът е общо достъпен, ако в системата има пробив в сигурността, то той лесно ще се забележи от недоброжелателни потребители, които биха използвали това с различни цели. Положителна страна на общодостъпния код е, че голяма група от експерти по сигорнустта, които разработват системи или предлагат услуги на база open source, имат интерес те да са максимално сигурни. При голяма част от системите за управления на съдържанието практиката е, след като се забележи потенциална заплаха за сигурността, тя да бъде отстранена, независимо от това дали реално някой може да се възползва от тази слабост, а общностите, които се грижат за сигорнустта, следят световните тенденции и вграждат най-надежните технологии в тези системи.

Сигурност на CMS: нито една система за управление на съдържанието с/без отворен код не гарантира пълна защита от атаки.

Основни начини на защита на системи за управление на съдържанието:

- Допълнително потребителско име и парола в Административния панел. Защитата на администрацията на сайта може да се извърши по два начина, но те се ползват единствено, ако опцията за регистрация на нови потребители не е активна (тъй като новите потребители няма да могат да достъпват акаунта си).
- Ако желаете администраторския панел да се достъпва само от IP адрес на администратора, е необходимо във файл с име .htaccess, разположен в директорията, където се намира login скрипта, да се поставят редовете: DenyfromAll Allowfromxxx.xxx.xxx.xxx, трябва да се замени с IP адреса на администратора.
- Защита на достъпа с допълнително потребителско име и парола. Използвайте опцията от контролния панел с Panel - „Директории, защитени с парола /PasswordProtectDirectories
- Промяна на потребителско име за достъп до Административния панел По подразбиране, на повечето системи за управление на съдържанието с отворен код потребителските имена с права на администратор са admin, adm1, administrator и т.н., както и с еднакви ID-та в базата данни.
- Променете потребителските имена и/или ID-то на потребителското име чрез адм ския панел на системата за управление на съдържанието.
- Промяна на паролата за достъп до Административния панел
- Сменете паролата по подразбиране, която се задава от системата при първоинсталация на системата за управление на съдържанието.
- Редовно променяйте паролата за достъп до администрацията. Използвайте пароли, които съдържат малки и големи букви, цифри и специални символи.
- Архив на уеб сайта - Редовното създаване на архив на уеб сайта (backup) и неговата база данни е сигуре съхраняване на последните корекции по системата при евентуални атаки.
Създайте архив на уеб сайта от с Panel -> Файлове >Бекъпи
 - Плъгини за допълнителна защита - Налични са и много плъгини за допълнителна защита на системата. Има приложения за промяна на URL адреса на администрацията за брой неуспешни опити за достъп, след което се блокира възможността за достъп до админи на сайта и др.
 - LimitLoginAttempts;
 - UserLocker;
 - Better WP Security;
 - LoginLock;
 - HideLogin и др.

Лично аз предпочитам изчистени, като сайта на Google машината. Този дизайн се постига много трудно, докато при тези системи – всичко е лесно, но имаме ограничения, рамки... или с други думи „Всяко нещо си има положителни и отрицателни аспекти!“.

17. Безопасност на уебсайта

При разработването на уебсайт винаги трябва да имаме предвид дали той гарантира безопасно ползване от потребителите и особено от непълнолетните потребители. Стандартите за сигурност на уебсайтове се повишават непрекъснато под натиска на

редица международни организации и институции. През 2009 г. Европейската комисия подписа с 14 от най-популярните сайтове социални мрежи в света „Принципи за безопасно ползване на социалните мрежи“, поставящи редица изисквания, за да се гарантира защитата на непълнолетните потребители от онлайн посегателства и опити за злоупотреба.

Европейската мрежа от центрове за безопасен интернет INSAFE оказва натиск върху държавните институции в държавите членки чрез националните центрове за безопасен интернет (в България това е Safenet.bg към Фондация „Приложни изследвания и комуникации) да въведат по-строг регламент и контрол върху уебсайтовете с цел защита на непълнолетните потребители.

Тук ще Ви представим някои от основните правила, които ще ви позволят да гарантирате, че сайтът ви отговаря на стандартите за безопасност.

17.1. Мерки за защита срещу кибератаки

1. Защита от нерегламентиран достъп и кражби на конфиденциална и класифицирана информация;
2. Предотвратяване на опитите за промяна на съществуващи данни, заличаване, триене;
3. Формиране на киберхигиена за безопасна работна среда с ИКС;
4. Защита при използване при електронна поща и социални мрежи;

Техническите средства са хардуерна и софтуерна защита на ИКС:

1. Средства за физическо осигуряване на КС срещу кражба, несанкциониран достъп и некоректно използване;
2. Средства за контрол на достъпа – защитни стени, пароли, задаване на биометрични данни;
3. Средства за превенция-откриване на непозволени пробиви NIDS;
4. РКІ – частни ключове, средства за кодиране;
5. Средства за автентификация – цифрови сертификати, маркери, електронни подписи;
6. Средства за защита от въздействие от електромагнитни смущения и импулси – екраниране EMI/RFI;
7. Средства за контрол на мрежата – подходящи софтуерни и хардуерни средства като скенери, снифери - сменя MAC адреса на компютъра ,и по този начин спира интернетa на цялата мрежа;
8. За защита на **уеб сървъри трябва да се инсталират сертификати от доверени системи – TLS, TTPS.**
9. Защита на крайните мрежови устройства – трябва да се обръща внимание на абсолютно всички обстоятелства и процеси, свързани с използването на съответното крайно устройство;
10. Защита на устройства с ОС Win: като Уиндоус е най-известната и най-разпространената система, тя е обект номер 1 на кибератаки. Microsoft се грижи, но има и препоръки: ъпгрейд на програмите към тяхната последна версия, криптиране на данните в Win 10, Update Assistant.
11. Win10 – криптиране на данни: включване на BitLocker; Search/BitLocker (control panel/System and Security/BitLocker Drive Encryption);
12. Memory Integrity – включване на функцията за цялостност на данните. Ако се прави запис на данни върху дяловете на ОС, се изчислява под формата на хеш-стойност, която определя каква информация е записана и в какво състояние е ОС

след използването ѝ. Ако при последно сравнение на хеш-стойностите не съвпадат, то е имало промяна на паметта. Отново от win Security.

13. Включване на защита от вируси - AVX;
14. DMZ, IPS/IDS – защитни стени;
15. Мулти-агентни интелигентни системи Smart Grid Distributed Intrusion Detection System (йерархична система, откриваща и класифицираща зловредни кодове и кибератаки);

ВНИМАНИЕ! Най-слабото звено в една инфраструктура си остава личността, за това е силно препоръчително да се осъществяват обучения, да се спазват задълженията. Киберсигурността е високоскоростен процес и е необходим ресурс от образовани хора, техника, софтуер, време. Пътищата за атаки са много и разнообразни, а най-ценния актив е времето постоянно и своевременно отговорност на киберзаплахите. Ето и следните ПРАВИЛА:

- Ползвайте повече от една парола;
- Пазете картовите си данни;
- Обръщайте внимание на заглавието на имейла; Ако съдържа граматически грешки или има несъответствия на вашия роден език – пратете го в кошчето, без да го отваряте;
- Следете отблизо профила си в банката; В случай на подозрителни трансакции, свържете се с вашата банка. Проверявайте трансакциите по банковата си сметка поне веднъж седмично. Най-лесно това става с мобилно и онлайн банкиране. С банката в телефона ви за секунди можете да следите наличността по сметката си отвсякъде и по всяко време;
- Внимавайте с линкове и прикачени файлове;
- Проверявайте кой ви изпраща съобщението;
- Бъдете внимателни, когато използвате обществени wi-fi мрежи;
- Правописните грешки често са предупредителен знак;
- Фишинг имейлите често съдържат правописни грешки в текста или малки грешки в изписването на предполагаемото име на подателя;
- Винаги актуализирайте софтуера на устройствата си;
- Използвайте силни пароли;
- За случаите, когато сте извън офиса
- Включете в пощата си автоматичното изпращане на съобщения само за получатели в рамките на своята работа, но не и за външни получатели. По този начин ще избегнете потенциални хакери да узнаят, че няма да ви има за известно време.
- Винаги проверявайте URL адреса. Ако сайтът включва "https://", тогава на сайта може да се има доверие и е защитен. Ако URL адресът е "http://" и липсва последната буквичка "s", то тогава избягвайте да въвеждате чувствителна информация като номера на вашата кредитна или дебитна карта, пароли и т.н.
- Внимавайте в социалните мрежи.
- Спазвайте етичен интернет етикет.

18. Общи условия за ползване

Условията за ползване на даден уебсайт са сбор от правила, които уреждат взаимоотношенията между собственика на сайта и потребителите. Условията за ползване

трябва да бъдат публикувани на видимо място в сайта. Те трябва да са възможно най-кратки, ясни и написани на език, който е разбираем за целевата група потребители. Ако сайтът предвижда регистрация и потребителят не приеме тези правила, той не трябва да има достъп до услугата. Някои от най-важните правила, които обикновено се включват в условията за ползване, са: данни за доставчика на услугата; описание на самата услуга (предназначение, целева група); допустима употреба на услугата според действащото законодателство; услуги, изискващи регистрация (изисква ли част от сайта или целият сайт регистрация, защо, какви данни се изискват); авторски права (може ли информацията да се копира, възпроизвежда, изменя и при какви условия); препращане към други уебстраници и сайтове (има ли такива връзки, какви са тези сайтове, гарантира ли доставчикът на услуга тяхната сигурност); как се гарантира поверителността на личните данни на потребителите.

19. Поверителност на личните данни

На сайта трябва да бъде подробно описано как се обработват данните на потребителите при използването на предлаганите продукти и услуги. Това включва видовете лични данни, които се събират (лични данни при регистрация, „бисквитки“, сървърни регистрационни файлове, данни за местоположение), кой има достъп до тях, както и за какво се използват. Изрично трябва да е посочено дали данните се предоставят на трети лица и организации и с каква цел. Видът, в който лични данни се подават към трета страна, не трябва да позволява идентификацията на конкретен потребител. При съхранение и обработка на лични данни трябва да се спазва Законът за личните данни, както и Наредбата за минималното ниво на технически и организационни мерки и допустимия вид защита на личните данни. За събиране, съхраняване и ползване на лични данни е нужна официална регистрация като оператор на лични данни. Ако сайтът позволява създаването на потребителски профили, потребителят трябва да има възможност да променя настройките за поверителност и сигурност или да изтрива трайно целия профил или части от съдържанието, което е качил.

20. Правила за съдържание

На сайта трябва да бъде публикувана и политиката относно материалите, които се публикуват от администраторите и от потребителите, ако те имат възможност да създават съдържание. Ако сайтът е информационен, трябва да се укаже дали се гарантират достоверността и обективността на съдържанието, както и откъде е взето. Ако в сайта се публикуват мнения или предположения, това трябва да бъде ясно описано и видимо за потребителите. Ако позволявате на потребителите да генерират мнения и съдържание, Ваша отговорност е да осигурите модератор с ясни правомощия, които не застрашават правото на потребителите на собствено мнение и свобода на словото.

Правилата за съдържание трябва да включват описание на недопустимите действия (публикуване на незаконно или вредно съдържание), както и мерките, които се предприемат в такива случаи (изтриване на съдържанието, изтриване на профила на потребителя, блокиране на достъпа до услугата, сигнализиране на съответните органи). За удобство на потребителя е полезно да се изреди конкретно какъв вид съдържание е незаконно и какво сайтът приема за вредно и за кого - за непълнолетни или за всички потребители. С цел осигуряване на безопасността на потребителите администраторите на сайта трябва да проявяват нулева толерантност към съдържание или линкове към съдържание, което експлоатира деца (детска порнография или педофилско съдържание), насажда омраза, популяризира хранителни разстройства (булимия и анорексия), самонараняване или подтикване към самоубийство, злоупотреба с

наркотици. Ако смятате, че съдържанието в сайта или негови части са неподходящи за деца, използвайте междинна страница с ясно обозначение на минималната възраст за достъп до това съдържание и помислете за механизъм за проверка на декларираната от потребителя възраст.

21. Специални препоръки за сайтове, насочени към деца

- В „Правилата за ползване“ на сайта още в самото начало трябва бъде изброено всичко, което е разрешено, и всичко, което е забранено да правят регистрираните потребители в сайта. Ясното посочване и възможно най- изчерпателно изброяване на всичко, което е изрично забранено и безусловно разрешено, е необходимо, защото, както показват изследванията за ползването на интернет от подрастващи, мнозинството от младежите до 18 г. не правят голяма разлика между „възможно“ и „разрешено“. Повечето от тях са склонни да смятат, че „щом функционалността на сайта позволява да се направи нещо, значи то е разрешено“.
- Ако сайтът изисква регистрация, редно е от непълнолетния потребител да се изисква само псевдоним, а не собствено име или да се предвиди възможност за регистрация само с валиден имей адрес и парола. Да не се изисква информация, която може да представлява заплах а за сигурността на детето, като позволява неговото идентифициране (град, училище, квартал, улица, телефонен номер). Необходимо е да се предвиди възможност всеки регистриран потребител сам да премахне своя профил, както и автоматично изтриване на профили, които не са били активни през определен период от време.
- „Правилата за ползване“ присъстват в пълния си размер в самото начало на процедурата по регистрация и потребителят няма как да продължи нататък, без поне да премине с поглед през текста им - за да стигне в техния край до линк, през който да продължи процеса на регистрацията си.
- Нужно е да се разработи система за сигнализиране за неподходящо съдържание, с по-конкретни опции за обозначаване защо потребителят изпраща сигнала (да може да се посочва, че съответното изображение представлява „порнография“, „онлайн тормоз“, „насилие“ или е „непозволено използване“).
- Необходимо е да се предвиди секция, съдържаща основните правила за безопасност, както и подробно описание на настройките за сигурност и поверителност и как се работи с тях. Добре е да бъдат посочени и връзки към източници за съвет и подкрепа при проблем в онлайн пространството: Онлайн горещата линия за сигнализиране на незаконно и вредно за деца съдържание и поведение в интернет (web12.net), Националният център за безопасен интернет (www.safenet.bg). Българската линия за онлайн безопасност - БЛОБ (www.blob.bg).
- Препоръчителни са и ограничаването и модерирването на възможността на малолетни и непълнолетни да споделят лична информация, снимки или видео с възрастните потребители на сайта. Редно е да се предвиди и филтриране на рекламното съдържание според възрастта на потребителя.

22. Списък с правни документи, които имат отношение към безопасността в интернет

Закон за личните данни - урежда защитата на правата на физическите лица при обработването на личните им данни.

Закон за електронните съобщения - урежда условията за предоставяне на услуги през компютърни системи.

Закон за електронната търговия - урежда обществени отношения, свързани с осъществяването на електронна търговия.

Закон за авторското право и сродните му права - урежда отношенията, свързани със създаването и разпространението на произведенията на литературата, изкуството и науката.

Наказателен кодекс на Република България - има задача да защитава от престъпни посегателства личността и правата на гражданите и цялостния установен в страната правов ред, определя кои общественоопасни деяния са престъпления и какви наказания се налагат за тях и установява случаите, когато вместо наказание може да бъдат наложени мерки за обществено въздействие и възпитание.

Конвенция за престъпления в кибернетичното пространство - насочена към закрилата на обществото от престъпността в киберпространството.

23. Интелектуална собственост и авторско право

Интелектуалната собственост е творение на нечий човешки ум - нещо, което човек произвежда със своя мозък и творчески умения. Интелектуалната собственост е защитена от закона в повечето страни и може да включва художествени произведения - музика, филми, пиеси, поеми, романи, картини, фотографии и други творби. С популярността на Мрежата, незабавния достъп до информация, както и лекотата за копиране и поставяне, потребителите трябва да са много внимателни да не нарушат правата върху интелектуалната собственост на други. Вижте дребния шрифт в уебсайтовете, за да проверите дали материалът, който искате да използвате, е безплатен или ограничен за използване. Нарушаването на авторско право или друга защитена интелектуална собственост е нарушаване на националното законодателство! Материал като текст, музика и видео, който е етикетан като обществено достояние, може да се използва без разрешение от автора. Някои материали може да се използват с разрешение от автора. Невъзможно е да знаете разрешенията, които имате, без да прочетете дребните букви!

Пример за раздел „Условия за ползване“ от сайта на Microsoft

Когато използвате изображения от други източници, не забравяйте първо да проверите законите за авторско право. Когато слушате музика или гледате изображения и филми в Мрежата, предположете, че те не са обществено достояние. Вместо това искайте разрешение от създателя или отидете в сайт, който предоставя безплатни ресурси специално за обществено ползване. Доктрината за справедлива употреба позволява използването на малко количество от даден файл. Следните три сайта ще предоставят информация относно законите за авторското право.

Авторът може обаче да упълномощи други да използват творбите му в ограничен вид. Законът за авторско право включва всички оригинални творби, дори електронни и дигитални медии. Оригиналната авторска творба е защитена през целия живот на автора и 70 години, след като е починал. Добре е да запомните правилото, че законите за авторско право винаги се прилагат по отношение на творба, освен ако изрично авторът не се е отказал от това, или е използвана в съответствие с публикуваните условия, като например Creative Commons License (Лиценз по авторско право).

Обърнете се към тези сайтове за информация относно законите за авторско право за изображения:

<http://www.biovisuals.com/copyright.html>

<http://fairuse.stanford.edu/C0pvri2ht and Fair Use Overview/chapter9/index.html>

Най-добрият съвет е винаги да искате разрешение, за да използвате авторски материал. Има обаче изключения за образователни цели:

- Доктрина за честна употреба

Материалите с авторски права може да бъдат използвани в определени ситуации, за образование и научни изследвания. Оригиналното произведение не може да се използва за търговски цели, копиране на работни книги или учебници е забранено, когато материалите може да бъдат купени и е позволено да се използват ограничени части от оригинална творба, включително цитати. Може да се ползват пасажи и цитати, но да се впише в списъка с използваната литература.

■ **Face-to-Face Teaching** (традиционен начин на обучение лице в лице)

Законът за авторско право не позволява представянето или показването на творби с авторско право по време на обучение лице в лице.

■ **Distance Learning** (Дистанционно обучение)

При този тип обучение само учениците от класа могат да имат достъп до съдържанието на творбите и произведенията трябва да са свързани с учебното съдържание.

Производните творби произлизат от оригиналните и единствено собственикът на авторските права има право да разреши нова версия на оригиналната творба. Производната творба трябва да се различава от оригиналната. Незначителни изменения или корекции няма да я квалифицират като производна.

Произведения, обозначени за обществено достояние, се използват безплатно от обществото.

Отворен код

Отвореният код е метод за разработване на софтуер, в който интелектуалната собственост може да бъде споделена публично, което позволява на всеки безплатно да използва и разпространява софтуера, както и да променя изходния код. Open Source Initiative (Инициатива за отворен код) е неправителствена организация, която насърчава използването на отворен код за разработване на софтуер и го предоставя безплатно на хората. Изходният код също трябва да се предостави за подобряване на качеството и надеждността на софтуера. Според сайта на OSI: „Една от най-важните ни дейности като орган за стандартизация е да поддържа дефиницията за отворен код за доброто на обществото“ (<http://www.opensource.org/>).

Open Source Alternative (Алтернатива за отворен код) (<http://xvwww.osalt.com>) предоставя достъп до алтернативи на добре известни комерсиални програми. Могат да бъдат открити различни заглавия за удовлетворяване на специфични нужди.

24. Поддръжка

Как сайтът ще се развива във времето?

Макара до тук, т. е. до пускането на сайта в Интернет, да е свършено вече много, всъщност не можем да оставим сайта просто така – за да има смисъл от него, трябва да го поддържаме и развиваме. Поддръжката може да е просто техническа, но много по-смислено и важно е да се актуализира сайтът на определен период от време, да се показват нови неща, да се допълва, за да е винаги интересен и полезен на своите посетители. От друга страна, е важно да знаем, че търсачките харесват повече сайтове, в които има ново и ново съдържание, и подминават тези, които не са променяни с месеци или години. С други думи, дори само от гледна точка на SEO поддръжката е важна.

Какво може и трябва да се поддържа в един сайт?

Анонсиране на новости. На първо място това са новините. Често бизнесът казва: „Но при нас няма толкова новини“, което далеч невинаги отговаря на истината. Най-малкото

фирмата има свой щанд на изложение и може да покани своите партньори и клиенти чрез сайта си на това изложение, след което да публикува кратка информация как то е преминало. Понякога представянето на вътрешно-скипна информация също е добър вариант за запълване на рубрика „Новини“. Получена награда за конкретен продукт, за фирмата като цяло или за неин представител, сертификат за качество или други подобни са важни и достойни новини, които не бива да остават извън знанието на нашите публики, и именно сайтът е мястото, където да се анонсират.

Във всички случаи новите продукти, проекти, нововъведения следва да бъдат анонсирани и в уеб сайта. Задължително е те да намерят място на първата страница на сайта – например в рубрика „Актуално“ или „Промоции“, където да се извеждат най-добрите предложения за текущия период.

Сред тези дейности спада и актуализацията на по-динамичните секции, като „Галерия“, „За медиите“, „Работа при нас“ или „Често задавани въпроси“. Своевременното добавяне на нови текстове, снимки, информация за медиите и др. е от ключово значение както за посетителите на сайта, така и за търсещите машини, които го индексират.

Техническа поддръжка. Разбира се, сайтът е софтуерно изграден и техническата поддръжка в повечето случаи се свежда до редовно подновяване на домейна и хостинга на сайта. Следва да се имат предвид характеристиките на сървъра, за който е разработен сайтът, защото при евентуални нови версии на сървъра е възможно някои функционалности да не работят коректно. Добре е да се изисква от хостинг компанията да предоставя в аванс информация при подобрения и изменения в сървърните конфигурации с оглед своевременно отстраняване на възможни нежелани последици в работата на сайта.

25.1. Обратната връзка

Обратната връзка разкрива една от възможностите за **интерактивност**, които предоставя Интернет. Нейната употреба прави сайта „жив“, показва отношение към посетителя и би могло да е от полза при негово доразработване.

Зад понятието „**обратна връзка**“ се крие възможността потребителят да изпрати своето мнение на създателите и собствениците на сайта (аналог на „Книга на посетителя“ в големите магазини, музеи и т. н.). Обратната връзка (feedback) обикновено се оформя на самостоятелна страница, на която се поставя както пълната информация за контакт със собственика на сайта (адрес, телефон, e-mail и др.), така и „форма“, която може да бъде попълнена и изпратена веднага, без да се стартира mail-програма.

Обратната връзка създава у посетителя усещане, че мнението му е ценено и необходимо. Възможността за обратна връзка следва да е на разположение на посетителите през цялото време, т. е. да се вижда на повечето вътрешни страници и задължително на началната.

Обслужването на сайта включва важната част на връзката с хората, които го посещават. **Интерактивността** и възможността за бърза обратна връзка са ключови предимства на Интернет като медия и частично могат да се постигнат чрез корпоративния сайт. В тази връзка своевременният отговор на всяко постъпило по имейл запитване, например чрез специална форма за контакт, е добра практика. В някои фирми и институции, като например в банковия сектор, където бързата реакция е от изключително значение, съществуват правила за отговор на подобни онлайн и отговор до 48 часа от постъпване

на запитването се счита за приемлив. Оставянето на писмо без отговор е груба грешка, която не бива да се допуска.

Статистика. Към поддръжката на сайта или проследяването на неговата работа спада и регулярният преглед на статистиките за посещенията, от които могат да се извадят доста изводи за ефикасността на сайта и посоката, в която е добре той да се развива. С каква честота да се следят статистиките, зависи от целите на сайта и от ресурса, който може да се отдели, но както и при самата актуализация и допълване на съдържанието, така и тук – темпо от минимум един или два пъти в месеца е приемливо.

Грешките. Често срещана грешка е, след като е готов, сайтът да се изостави. Той е отнел достатъчно усилия и след това екипът, който следва да отговаря за поддръжката му, трудно намира мотивация да продължи работата по него. В това отношение е добре да се състави план за поддръжката на сайта, който да се следва редовно. Подобен план може да се направи за година и да се актуализира на месечна или тримесечна база с оглед конкретните необходиминости на бизнеса.

Други допускани грешки са:

- Създава се сайтът, вдига се много шум около него, привличат се посетители, след което не се полагат достатъчно усилия той да се актуализира своевременно и да доставя необходимата информация на хората, които го посещават;
- Инвестира се голяма сума в началното създаване на сайта, без да е предвиден достатъчен ресурс за неговата поддръжка;
- Забравя се да се актуализира важна променена информация на сайта като например телефонни номера за контакт, премахване на стари продукти или обяви за свободни работни позиции и др.;
- Очаква се сайтът да продава или води нови клиенти, без да се полагат достатъчно усилия в тази посока след началното му създаване;
- Влагат се прекалено много усилия в началото, сайтът се пуска в „бета-версия“ и екипът, съответно компанията губят интерес от финализиране на усилията и довеждането на проекта до край.

Кога сайтът има нужда от редизайн или бизнесът ни – от нов сайт?

Обикновено още при първите разговори за изграждането на нов уеб сайт става дума за очакваният за времето, когато той ще върши работа и няма да е необходима активна намеса по него, отвъд рутинната поддръжка. Често инвеститорът и клиентът искат сайтът им да служи завинаги или поне за идните пет години. Разбира се, подобна възможност съществува, но това не означава че след пет години сайтът ни ще изпълнява правилно поставените му цели и задачи. Защо ли? В динамичното време, в което живеем, не само технологиите, но и бизнесът ни се променят в рамките на година-две, понякога дори за месеци. Както, когато наемаме нов офис, не можем да сме сигурни, че след две години, така и сайтът ни расте и се променя с времето. Ясно е, че след пет години, ако не сме поне преободисали офиса, той ще изглежда плачевно – така е и със сайта.

Няколко съвета при редизайна:

- Не отлагайте много редизайна или подмяната на сайта си;

- Създаването на нова версия на сайта е добър повод да поразгледате добре стария, да прецените кое да отпадне и кое да остане, да осмислите от друга гледна точка бизнеса си;
- Отделете достатъчно време за новия сайт, не пришпорвайте създаването му, за да стане наистина добър;
- Разгледайте сайтовете на конкуренцията и на водещи чужди фирми във вашата област, изведете основните елементи, които ви харесват и липсват в стария ви сайт;
- Помислете какво още да включите с оглед на това сайтът да е максимално полезен на вашите публики и винаги го гледайте с техните, а не с вашите очи;
- Не променяйте прекалено често дизайна на сайта си – това ще доведе до объркване у посетителите, ако част от тях не ви навещават по-често;
- Не допускайте новата версия да изглежда по-зле или да върши по-малко работа от старата.

25.2. Кога сайтът е успешен?

Кога да считаме, че сайтът е успешен? Колко посетители трябва да има той, за да сме сигурни, че инвестицията в сайта е добра и си е заслужавала? Критерий ли е броят посещения за успеха на един сайт? Това са важни въпроси, които съвсем закономерно си задава всеки собственик или мениджър, инвестирал в създаването на свое корпоративно онлайн присъствие.

Сайтът има много посетители. За да има един сайт много посещения, може да има различни причини – може да е направен много добре, да има добра оптимизация за търсачките и качествено, полезно съдържание, поднесено по правилния за аудиторията начин. Но трафикът към един уеб сайт би могъл да бъде висок, без това задължително да прави сайта успешен.

Сайтът има малко посетители. Дори сайтът ни да има 5 или 10 посетители дневно, ако това са именно хората, за които сме създали сайта, то той ще е успешен. Или с други думи, ако сайтът има квалифициран трафик, дори и няколко посетители ще осмислят съществуването му. За да измерим това, все пак трябва да поставим измерими цели. Като цел можем да си поставим да накараме хората да се обаждат по телефона например за повече информация; да се регистрират, участват в игра, отговарят на въпроси или да свалят конкретни документи, оферти, каталози и т. н. и т. н. Всичко зависи от целите, които сме си поставили по отношение на сайта като цяло още в самото на работата по създаването му.

Сайтът печели награда. Участието в конкурси от типа на „БГ сайт“ (www.bgsite.org) – конкурсът за най-добър български уеб сайт, Webby Awards – Световните награди за уеб, или други специализирани форуми са основава на олимпийският принцип, че е важно участието. Ако с това сайтът спечели и награда – още по-добре. При всички случаи не е редно сайтът да се създава специално с цел да спечели награда, а наградата да дойде като допълнително признание за труда на екипа, създал сайта. Разбира се, подобна награда е знак, че сайтът е добър. Но невинаги е гаранция, че е успешен.

Сайтът реално генерира бизнес. В някои случаи сайтът се създава просто защото никой бизнес не може без сайт в днешно време. Създава се, без да му се възлагат големи надежди. Факт е, че и при по-малки фирми, и при големи корпорации сайтът може да има ключово значение и дори от първия си ден онлайн да генерира бизнес и да

изпълнява успешно поставените му цели. В този смисъл, ако сме вложили достатъчно ресурс и сме изградили един добър сайт, а той не изпълнява своите задачи, може би е добре да се замислим, да направим някои тестове и проучвания и да видим дали сме заложили неправилни цели, изискваме от сайта прекалено много или може би сме създали сайта с неправилни послания и функционалности и потребителите ни очакват от нас и от сайта друго. При всички случаи е добре на сайта да се гледа като на важен компонент от нашите комуникации и следва своевременно да се оптимизира спрямо изискванията на бизнеса и средата.

Основните въпроси за уеб сайта в резюме

В резюме на всичко, казано до тук, ето основните въпроси, на които следва да се отговори преди началото на работа по изграждане на един уеб сайт.

1. Защо се прави сайтът? Какви са неговите цели и задачи?
2. Каква ще е аудиторията на сайта? Кой ще са публиките на сайта? Кой са основните типове потребители, които очакваме на сайта? Какъв е техният профил?
3. Под какъв домейн ще се изгражда сайтът?
4. На какъв хостинг ще е? Има ли специфични изисквания към хостинга?
5. В какъв стил ще се изгражда сайтът?
6. Каква ще е структурата на сайта? Какво съдържание ще се предлага на него? Как ще е изградено и свързано това съдържание? Какви ще бъдат акцентите?
7. Какви функционалности ще има на сайта? Какви възможности ще са предоставени на различните типове посетители на сайта?
8. Кой, за колко време и при какви възможности ще изгради уеб сайта?
9. Кой ще поддържа сайта?
10. Каква ще е стратегията за популяризиране на сайта във виртуалното и реалното пространство.

Създаването на сайт е нелека задача, изискваща комплексен подход, екип от различни специалисти и заделяне на немалък ресурс от средства, време и труд. Затова е добре към него да се подходи сериозно, както и сериозно да се изисква да доставя заложените резултати, след като вече е готов.

След като преминахме през целия процес по създаване на един сайт, следва да проследим и какви типове сайтове съществуват и какви са техните специфики в едно или друго отношение.

Ето и видовете:

- Корпоративен
- Продуктов
- Промоционален
- Електронен магазин
- Онлайн общности
-

- Портални – <http://www.hit.bg> - сега е премахнат;
- Информационни – <http://slovo.orbitel.bg>
- Новинарски – <http://www.netinfo.to>
- Развлекателни – <http://pozdravi.dir.bg>
- Търговски – <http://www.91111.com>
- Фирмени – <http://www.ciela.net>
- Лични – <http://ravanov.cjb.net> и т. н.

Дизайнът и структурата на страниците и сайтът като цяло зависят до голяма степен от типа сайт, който се създава. Изискванията към графиката и текстовото съдържание е различно при различните сайтове.

25. Представяне на сайта на друг език

При необходимост от дублиране на съдържанието на един сайт на друг език са възможни подходи в зависимост от целта на сайта. Възможно е да се направи втори сайт, като се използват вече подготвени графични решения и структури. Правят се навигационни HTML – страници на база на основните, като текстът в тях се изписва на друг език. Графиките, в които съществува текст на първия език, се преработват, а останалите се ползват такива, каквито са.

Друго решение е създаването на различно съдържание на двата езика в зависимост от конкретните цели, които сайтът си поставя. В този случай също е възможно да се използват голяма част от графичните материали, подготвени вече за първия езиков вариант, но се налага съдържанието да бъде съобразено с този факт. Как се решава това в контекста на структурата на сайта?

Обикновено на входящата страница се поставя меню за смяна на езика. Не се препоръчва поставянето на такво меню на вътрешните страници, защото се предполага, че потребителят е направил своя избор още при влизането в сайта.

Понякога началната страница съдържа информация и менюта на два (и повече) езика. Така потребителят избира възможността направо, без да е необходимо да преминава на следващо ниво.

26. Хост

Хостът е физическото място, сървърът на доставчика на Интернет-услуги, където ще се съхраняват файловете от уебсайта и от където те ще се зареждат при обръщение към тях от Интернет. Хостът е тясно свързан с домейна, тъй като при запазването на собствен домейн е необходимо да се укаже къде ще се хоства в бъдеще сайтът, т. е. фиксираният IP адрес от набора на съответния доставчик, на който ще съответства домейнът. Това не е в сила при безплатния хост, при който се ползва поддиректория на чужд домейн.

Публикуването включва избор на фирма, която предлага хост и домейн, същинско публикуване и управление на съдържанието на уеб сървъра. Ще Ви изброя безплатните хостинг услуги:

www.freehosting.co/
<https://googiehost.com/>

https://freehostingnoads.net/
www.awardspace.com/
www.000webhost.com/

Добре е да знаете, че по-добър хостинг е платеният: Най-евтини хостинг услуги и домейни ще намерите в ICN.bg, dom.bg.

За разлика от безплатното уебпространство платеното не предполага намеса от страна на доставчика, т. е. налагането на рекламни банери и др. Освен това платеното пространство може да е много по-голямо спрямо необходимостите на сайта. Доставчикът в този случай е длъжен да предоставя пълна статистика за посещенията на сайта (Server Log), както и да позволява употребата на активни обръщения към сървъра. Цената на уебпространството е различна на различните сървъри.

.com	28.75 лв. 37.75 лв.	28.75 лв. 37.75 лв.	37.75 лв.	1 година
.bg национален	58.75 лв. 70-40 лв.	58.75 лв. 70-40 лв.	64.75 лв. 70-40 лв.	1 година
.eu супер цена	7.75 лв. 22.75 лв.	7.75 лв. 22.75 лв.	22.75 лв.	1 година
.net	33.75 лв. 39.75 лв.	33.75 лв. 39.75 лв.	39.75 лв.	1 година
.online супер цена	3.75 лв. 78.75 лв.	78.75 лв.	78.75 лв.	1 година
.euo	7.75 лв. 22.75 лв.	7.75 лв. 22.75 лв.	22.75 лв.	1 година

DOMBG v2.0 dom.bg

Сигурна връзка

Актуално

Домейни

Хостинг

Цени за домейни

Цени за хостинг

Цени за SSL сертификати

Плащане

Риселъри

ТОВА Е МЯСТОТО ЗА РЕГИСТРАЦИЯ НА ДОМЕЙНИ, ХОСТИНГ, SSL СЕРТИФИКАТИ

Начало Домейни Хостинг Други услуги Информация Риселъри Безплатно

В УСЛОВИЯ НА ИЗВЪНРЕДНО ПОЛОЖЕНИЕ ВЪВ ВРЪЗКА С COVID-19 ОФИСЪТ НИ НЕ РАБОТИ С КЛИЕНТИ

dom.BG е лидер при регистрация на домейни и хостинг услуги! Досега наши клиенти са регистрирали и хостват на наш домейни сумарно за повече от **210,000** години!

Може да платите по **всички възможни начини** достъпни в България.

dom.BG: безплатен хостинг

С всеки домейн получавате **1 месец безплатен хостинг по план ОК!**

Ако имате домейн, направете заявка за безплатен хостинг

Проверете, дали желаният от вас домейн е свободен

Домейн: Провери!

пример: bulgaria.com или ime.familia.name

проверка и за домейни с други разширения

.EU ОТСТЪПКА: 9.50 лв отстъпка за .EU домейни: вземи бонус код тук

Домейни

31.69 лв - 16.91 лв на година | [Ценообразуване](#)

COM, INFO, BIZ, NET, ORG, MOBI, EU, NAME, US, DE, TV, CC, UK, CA, CN, CH, NL, BE, AT, ME, LI - Може да регистрирате нови домейни или да **подновите** при нас домейн регистриран другаде; заявка за регистрация на едро; може да използвате **безплатните** услуги. [Връзки за домейни.](#)

Хостинг

3.90 лв - 2.90 лв на месец

Бърз хостинг в и извън България за вашия уеб сайт и... Всички пакети включват **FTP** достъп, **WWW** достъп, **SSL**, **Perl**, **MySQL**, **POP3** имейл адреси, **пренасочване** и **имейл**, **статистика** и др. [Връзки за хостинг.](#)

Фиг. 21 примерен сайт, от където да купите хост и домейн

Всеки уеб дизайнер трябва да спазва етични и правни норми като цяло в разработката си на уеб сайт, публикуването му. Съдържанието на сайта не трябва да съдържа материали, които са обект на авторско право, ако има такива трябва да се впише в използвана литература.

Коректно цитирани източници, може да се посочи хипервръзка и автор, информацията в сайта трябва да отговаря на истината, всичко в сайта не трябва да е нецензурно, не се заплашва, обижда, накърняват права или достойнство на други. Не бива да се поражда омраза към етническа, расова или верска, към престъпления или нарушаване на закона.

В интернет има още много сайтове, предоставящи безплатно пространство. Такива услуги предоставят например web.dir.bg, web.bg, www.freeyellow.com, www.geocities.com и т. н. Безплатният уебхостинг всъщност не е съвсем безплатен, тъй като собственикът на сървъра обикновено си запазва правото да поставя свои рекламни банери, фреймове или поп-ап рекламни прозорци из сайта, което забавя зареждането на страниците, нарушава визията, не прави добро впечатление на посетителите и не на последно място – съществува риск посетителят да последва рекламния линк преди още да е видял сайта. Обикновено отдаваното безплатно уебпространство е ограничено. При българските доставчици на подобни услуги то е от 2 до 5 МВ, което е своеобразно ограничение и често е незадоволително. Някои сървъри предоставят безплатно и по-големи обеми. Друг недостатък на безплатното хостване е дължината и сложността на URL. Ако сайта ще има достъп само чрез връзки, което е твърде хипотетично и нереално предложение, то може би дължината на URL няма значение. Но в действителност потребителите с нежелание пристъпват към набирането на дълъг и със сложно изписване адрес, какъвто е случаят със сайта PhiloTrain – <http://members.tripod.com/philotrains/index.htm>.

Обикновено безплатно уебпространство не позволява употребата на скриптове и активно обръщение към сървъра. В последните месеци това, което може да се използва на някои сървъри е езикът PHP. За съжаление почти няма безплатен хостинг, предоставящ възможност за ASP или CGI.

Понякога при безплатен хост се поставя изискване графичните файлове да бъдат в отделна директория, което трябва да се има предвид при изграждането на сайта.

Безплатно уебпространство се използва най-вече за лични сайтове. Не е препоръчително фирмени или сериозни сайтове да се съхраняват на безплатен уебсървър и е напълно неприемливо сайтове за електронна търговия да използват тази услуга.

27. Тестване и публикуване

Тестването на уебсайт е процес, допълващ неговата изработка и управление. Това гарантира функционалността на изготвения сайт. Прави се проверка на неговото цялостно поведение в работни условия, което включва:

- Проверка дали даден сайт е написан според установените стандарти и правила на езика на програмите – валидиране.
- Тестване на визуализацията и функционалността на уеб сайта, независимо от уеб браузъра и операционната система, която се използва.
- Тестване на скоростта на зареждане и достъпността на уебсайта
- Проверка на сигурността на уеб сайта.
- Тестване на администраторската част – контролен панел, достъп и регистрация на потребителите, пароли и др.
- Тестване на сайта за мобилна съвместимост с различните видове дисплеи и монитори на различни мобилни и настолни устройства.

28. Регистрация на сайт в интернет търсачки и директории

Ако искате сайтът Ви да бъде откриваем, когато потребители правят търсене по ключови думи в интернет търсачки като Google, Yahoo! и т.н., трябва да регистрирате сайта си в съответните търсачки. Преди да направите това обаче, добра идея е да оптимизирате сайта си за търсачки с т.нар. SEO оптимизация.

Когато решите, че сайтът Ви е оптимизиран, съобщете на търсачките за неговото съществуване - можете да подадете заявка индивидуално към всяка от интернет търсачките. Ето списък с хипервръзки към адресите за регистрация в някои от най-големите търсачки:

- [Google](#)
- [Yahoo](#)
- [Bing](#)

Google предлагат услуга webmaster tools, с която можете да управлявате и следите процеса на индексирание на страниците си. Чрез тези инструменти например можете да укажете къде се намира файла sitemap.xml за по-бързо индексирание на страниците на сайта, да видите кои са най-честите заявки за сайта Ви по ключови думи или да подадете молба за повторно включване на сайта в индекса, ако по някава причина бъдете изключени.

Голяма част от информацията, която изискват търсачките при регистрация на сайт, е еднаква за различните търсачки. Това позволява автоматизирането на процеса и едновременно регистриране в голям брой търсачки. В интернет можете да намерите голям брой подобни инструменти, но ето един от по-известните: selfpromotion.com

Сайтът предлага 4 седмична безплатна регистрация на акаунт, чрез който можете да подадете автоматична молба за регистрация към повече от 70 от най-значимите търсачки. Ако искате да запазите акаунта си, ще трябва да платите известна сума - толкова, колкото решите че си заслужават предлаганите услуги. Selfpromotion предлага и доста подробен анализ на SEO оптимизацията - факторите за добро класиране на сайт от интернет търсачки.

Пишете по блогове за сайта си - така ще привлечете още посетители, а като добавка имате добър шанс сайта Ви да бъде индексирани и от търсачки, за които не сте подавали молба за регистрация. Причината е, че много от търсачките често посещават блоговете, където информацията често се обновява, и откритите на блога хипервръзки към сайтове се проверяват и добавят към списъка за индексирание.

Регистрация в директории

Включването на сайта Ви в популярни директории е един от най-сигурните начини да осигурите посетители за сайта си. Още едно предимство на директориите е натрупването на входящи хипервръзки към сайта - основно изискване за добро класиране от търсачките.

Много от директориите имат някои основни изисквания за добавяне на сайт - страниците да са завършени, да няма неработещи хипервръзки, да няма страници с неподходящо съдържание - напр. подтикващо към насилие, противоречащо на законите на страната или морално неприемливо и пр.

Често добавянето на сайтове е безплатно, но повечето директории предлагат и платено включване, което осигурява по-добра видимост на сайта. Предлагам Ви списък с някои популярни директории с кратко описание и хипервръзка към страницата за добавяне на сайт:

- [DMOZ](#) е една от най-авторитетните директории и включването на сайта Ви в нея автоматично означава включването на сайта Ви в ред търсачки, които работят съвместно с DMOZ.

Всеки сайт се преглежда от няколко редактора преди да получи одобрение. Поради огромният наплив от кандидатури и факта, че редакторите работят на доброволни начала, процесът на обработка на заявките за включване на сайта е пословично бавен - добавянето на сайта Ви може да отнеме месеци. Въпреки това, престижът който носи добавянето на сайта в DMOZ в повечето случаи си заслужава усилията и търпението.

- [SVEJO](#) - една от най-големите български директории, която се управлява изцяло от потребителите, които добавят нови сайтове и определят класацията им с гласуването си за тях
- [Lubimi.com](#) - "социална мрежа за споделяне на любимите препратки" по думите на самия сайт.
- [Гювеч](#) - указател на български сайтове, който е същевременно и портал с новини. Работи съвместно със Svejo. Можете да добавите безплатно сайта си ("Добави сайт") или да осигурите по-добра видимост на сайта с платената услуга "Добави VIP"
- [Dir.bg](#) - също много популярен портал с каталог на български сайтове.
- [All.BG](#) - това е всъщност портал, който освен директория със сайтове предлага и новини. За да добавите сайт, влезте в категорията, която смятате за подходяща за сайта си и от там изберете "добави сайт" от главното меню.
- [BGtop.net](#) - класация на български сайтове. За да бъде добавен сайт, той задължително трябва да има вариант на български език. Интересна особеност на класацията е нулирането на гласовете за всички сайтове всеки месец. Така всеки новодобавен сайт има възможност за равен старт с останалите сайтове от категорията си.
- [Yes.bg](#) - каталог с актуалните български сайтове. Можете да регистрирате сайта си както на български, така и на руски и английски, но само в една категория.
- [Dao.bg](#) - тази социална мрежа е специализирана за интернет - SEO, дизайн, интернет реклама и т.н. Включването на сайт изисква предварителна регистрация, която е безплатна.
- [Idg.bg](#) - посветен на информационните технологии, компютри, софтуер и интернет портал. Директорията включва български и чуждестранни сайтове
- [Nasoki.com](#) - портал с директория на български сайтове, в която можете да създавате свои категории за включване на сайт
- [Българска директория](#) - директория с подбор на качествени сайтове от България

- [OnlineCategory.Com](#) - Deep Link уеб директория, т.е. можете да добавите към директорията освен началната си страница и някои от вътрешните страници на сайта си. Този т.нар. deep linking подход придава по-голяма тежест на сайта пред интернет търсачките. Формата за регистрация позволява много детайлно описание на сайта, както и добавяне на рекламни страници.
- [AddBg.com](#) : [Уеб Директория](#) - Една съвсем нова директория за български сайтове, в която вече има регистрирани близо 3000 сайта.
- [БГ Директория](#)
- [Уеб директория](#) - по думите на самия сайт, "Свободна уеб директория с избрани полезни български сайтове". Администрацията е много коректна и бързо ще получите отговор на кандидатурата си за включване. Правилата за приемане на сайт са доста стриктни и добавените сайтове не са много, но това облагодетелства както читателите (понеже могат да избират само между по-качествени сайтове), така и авторите на сайтове (понеже ги поощрява да създават по-качествено съдържание за сайтовете си - в крайна сметка именно смисленото и ползено съдържание определя успеха на един сайт).

Следват няколко англоезични директории. Изискването за включване на сайт към тях е сайтът да е написан на английски или поне да има версия на английски.

- [Tsection Web Directory](#)
- [Free web directory](#)
- [Free Web Directory](#) - директорията предлага както безплатно, така и платено включване на сайтове.
- [Directory World](#) - Включването тук е безплатно ако предоставите на сайта си обратна хипервръзка към директорията.

Можете да подадете и заявка за включване към няколко български социални мрежи автоматично от сайта [BgSocialMarker.com](#)

Писане на статии

Още един интересен начин за популяризиране на сайта Ви предствлява написването на статия на тема, близка до съдържанието на сайта, в която да включите препратка към него. Няколко български сайта са специализирани в публикуване на качествени статии и позволяват включването на препратки към сайтове:

За да публикувате статии в сайта, трябва да се регистрирате - това е безплатно и става много бързо. Статията Ви трябва да е оригинална, да не е публикувана на друго място, да е граматически правилно написана и сама по себе си да бъде съдържателна. Всяка одобрена от редакторите и публикувана статия може лесно да бъде добавена към Свежо с предложения бутон в дъното на страницата.

- [Директория за статии - публикувайте статии с линкове безплатно. \(direktoriqzastatii.com\)](#) - Безплатна Директория за Статии

За публикуване на статия и тук се изисква регистрация, която е безплатна. Статията Ви ще бъде публикувана след одобрение от страна на администратор.

Уязвимости в уеб-приложения: една от основните уязвимости в използвани приложения в съвременни КМС е нерегламентирано вмъкване на код в отворени форми, които се използват за попълване на формуляри и придават интерактивност на уеб сайта. Всеизвестно е, че езикът за програмиране JavaScript придава живина и интерактивност на уебсайта. Но функционирането на скриптовете са уязвими точки не само към сървъра, а и към клиентската част в инструментариума на интернет брауъра. . При кражба на данни от сървърната част, хакерите и кракерите могат да заразяват клиентската част с малуер посредством открити уязвимости в приложния код.

ВАЖНО!!! Една от най-популярните инструменти за изследване на уязвимости е OWASP ZED Attack Proxy – ZAP. Той се поддържа активно от стотици международни доброволци и това помага автоматизирането при търсене на уязвимости в уебсайтове и уеб приложения.

За съжаление комуникацията в Мрежата е отворена и неконтролируема. Това е сериозна предпоставка за голямо разнообразие от всевъзможни атаки. За да се обезпечи сигурността на информацията, трябва да се избегне неоторизирания достъп до уеб данните. Уеб сигурност е защита на нашата информация в Мрежата срещу достъп, използване, разкриване, увреждане, промяна, преглед, запис или изтриване от неупълномощени лица. Следните заплахи са:

- Загуба на целостта на данните – нарушителят добавя, изтрива или променя информацията;
- Загуба на конфиденциалност на данните – информацията става достъпна за неоторизирани лица;
- Загуба на услуги – дадени услуги не могат да се изпълняват в резултат от действията на атакуващия.
- Загуба на контрол – услугите се използват от оторизирани потребители по неоторизиран начин.

Срещу нормалното функциониране има различни атаки:

- DoS атаки – от английски Denial of Service - отказ на услуга. Атаката може да стане с изтощаване на ресурсите или чрез възползване от грешка в софтуера на жертвата. По време на такива атаки рязко се повишава разходът на ресурси на процесора и намалява пропускателната възможност на канала за свързване, което може да доведе до силно забавяне на работата на цялата компютърна система, на отделни задачи или изобщо спиране на услуги. Нужен е специализиран софтуер, ограничаващ IP адресите, които изпращат неограничен брой заявки към сървъра за зареждане на сайта. Този софтуер се инсталира на малък сървър.
- Sniffing атаки – от английски – подушване. Те събират или прихващат данни чрез улавяне на информацията с помощта на програма, насочена към улавяне на мрежови пакети. Ако пакетите данни, които се предават през мрежата, не са криптирани, те могат да бъдат почетени от това приложение. По този начин нападателят може да анализира мрежата и да получи информация, с която мрежата да се срине или да се повреди, или да прочете комуникациите, случващи се в мрежата.
- IP Spoofing – от английски – измама. Включва промяна на заглавието на IP пакета и на изпращаните съобщения. То ги кара да изглеждат така, сякаш идват от IP адрес, различен от реалния адрес на първоизточника. Целият поток от данни към и от даден компютър или мрежов сегмент може да бъде пренасочен да минава през определен компютър – компютъра на атакуващия. Така потокът

от информация може да бъде прегледан, претърсван, подправян. Това е метод за придобиване на неоторизиран достъп до компютър или мрежа за започване на атака, за кражба на данни или за унищожаването им.

Има рискове при използване на бисквитки и веб маяци.

HTTP бисквитките от английски HTTP cookie – това е съвсем малък по размер файл, който се създава и изпраща от веб сайт -сървъра, и се записва от веб браузъра на потребителя в системното пространство. При всяка следваща заявка, от браузъра към същия сайт, информацията от бисквитката се изпраща към сървъра. Бисквитките се използват от веб сайтовете, за да „различават“ и „запомнят“ своите посетители, регистрирани или гости, да запомнят предпочитанията им и състоянието на дадена функционалност, услуга или особеност на сайта си. Тези бисквитки може да са сесийни или постоянни. Сесийните са временни и се изтриват при затваряне на браузъра и се използват за запомняне на текуща информация при сърфиране, а постоянните се съхраняват в браузъра за определен срок, зададени при създаването им. При атака към тях, може да се вземат мерки.

Но атакуващия може да „инжектира“ код чрез модификация, като по този начин атакува и целия сайт.

Веб маяк е малко графично изображение, което изследва веб активността на потребителя. Често се съдържа в прозрачно графично изображение с размери един на един пиксела, намиращо се в имейл или сайт. Когато потребителят посети сайт или изпрати имейл, съдържащ веб маяк, се записва информация за потребителя. Тази информация може да съдържа IP адрес, по кое време е посетена страницата, тип на устройството, браузър, и др. често се ползва в комбинация с бисквитки. Тези маяци могат да се открият в HTML кода на веб страница при търсене в тагове . Но за разлика от другите изображения, те се зареждат от друг сървър. За да спрат това проследяване, много от приложенията за електронна поща не показват изображения по подразбиране.

Съществуват сертификати за сигурност - TLS R SSL. Те се използват, за да криптират данни, защитават канала на информацията между веб браузърите и веб сървърите, където са разположени сайтовете. Така всички данни са криптирани и защитени. Инсталирайте си на сървъра такъв сертификат!

29. Оптимизация и Реклама

Оптимизация на сайта - достъпност
Достъпността на Вашия сайт е съществено важна за това сайтът да бъде използван и да бъде посещаван отново и отново.

Започнете с проверка на времето, необходимо на Вашия сайт да се зареди на екрана. Времето, необходимо на страницата да се визуализира на компютъра на потребителя, е пряко свързано с размера ѝ. Има два начина, по които можете да намалите размера на



Вашият сайт, за да съкратите времето за зареждане: да оптимизирате сайта и да компресирате изображенията.

Оптимизация е процесът, чрез който правите подобрения на вашия сайт, за да намалите времето за зареждане за зрителя. Ако използвате Expression Web за създаване на вашия уебсайт, в менюто Инструменти (Tools) щракнете върху Оптимизирай HTML (Optimize HTML). Ще видите диалогов прозорец, който съдържа нещата, които можете да премахнете, за да оптимизирате своя уебсайт. Вторият начин, по който бихте могли да намалите размера на своя сайт, е да компресирате изображенията. Включването на много изображения на Вашия уебсайт ще увеличи времето за зареждане. Използването на анимирани изображения още повече увеличава това време. Простото правило е: **не използвайте прекалено много изображения или анимации в своя сайт.** Използвайте ги, където те са подходящи, за да помогнат на посетителя да разбере информацията. Избягвайте да използвате изображения само за да направите сайта привлекателен. Помнете: **колкото повече изображения включвате, толкова повече се увеличава времето за зареждане.** Изображенията трябва да заемат възможно най-малко място.

За създаването на уебсайтове, които са достъпни за хора с увреждания, е необходимо дизайнерите да осигурят текстова алтернатива за позициите без текстово съдържание (най-често изображения или други визуални елементи). Чрез осигуряването на текстови алтернативи (или алтернативен текст - alt text), както най-често се наричат, позициите без текстово съдържание може да бъдат транслирани във форма, която да се използва от човек със зрителни затруднения чрез уебчетец.

Втората употреба на алтернативния текст е чрез търсачките. Те могат да „четат“ алтернативния текст, поставен върху дадено изображение, за да го идентифицират. Използвайки този алтернативен текст, Вие увеличавате възможността Вашият сайт да се придвижи напред в класацията на страницата с резултати от търсенето и следователно да бъде разгледан от повече потребители.

За да определите използването на алтернативния текст във вашия уебсайт, визуализирайте сайта в Preview в режим Browser. Ако сте използвали алтернативен текст за изображенията във Вашия сайт, той ще се появи, когато прекарате курсора върху картинката. Помнете: **алтернативният текст трябва да бъде кратко, но конкретно описание на изображението.**

Преди да качите сайта в Мрежата, уверете се, че всички файлове са на правилното място с точните си имена. Проверете отново името на началната страница, за да се уверите, че спазва изискването за наименуване, което ви е дадено от Вашата услуга за уебхостинг. Също така се уверете, че всички файлове, които ви трябват, се съхраняват на място, до което имате лесен достъп по време на процеса за качване. Бъдете особено внимателни при преместването на файлове. Преименуването или преместването на файлове означава, че трябва да промените пътя към тях в своя уебсайт. Файловете със сгрешени места и неточните имена ще накарат зрителя да получи съобщение за грешка по време на навигацията в сайта.

Някои програми за уебдизайн като Microsoft Expression Web проследяват имената на файловете и местоположението им в процеса на работа. Препратките и връзките към страниците се актуализират автоматично, когато се правят промени във файловете в рамките на работния „сайт“.

Окончателната задача, преди да публикувате сайта, е да го модифицирате така, че търсачките да го откриват и класират на по-високи позиции. Има два начина за това. Първият е с помощта на метаелементи (наричани понякога метатагове). Метаелементът се състои от <meta> таг с конкретна информация за сайта. Тази информация може да бъде под формата на заглавие на страницата, описание на сайта и ключови думи, които

идентифицират информацията, съдържаща се в него. Тази информация няма да бъде видима на вашия уебсайт. Когато обаче зрителят използва търсачка и напише ключови думи, информацията ще бъде използвана, за да включи Вашия сайт в списъка на показаните връзки по темата. Описанието, което ще въведете в метаелемента на вашия уебсайт, ще се използва още за насочване на потребителя към него. Ако не включите описание във Вашия уебсайт, търсачката ще използва автоматично първите 250 думи на началната страница, за да го категоризира.

Най-добре е да включите тези метаелементи във всичките страници на сайта не само в началната. Всяка страница може да съдържа различна информация и затова ще поискате да смените ключовите думи и описанието ѝ. Вашият зрител изпитател е ценен източник при определянето на някои ключови думи за Вашия уебсайт и за отделните страници. Попитайте зрителя кои думи трябва да се използват, за да се опише всяка страница.

Ако използвате Microsoft Expression Web като софтуер за вашия уебдизайн, можете да въведете описанието на страницата и ключовите думи чрез диалоговия прозорец Page Properties. За тази цел щракнете с десния бутон на мишката върху някоя част от страницата и изберете Page Properties (Свойства на страницата) от менюто за бърз достъп. В диалоговия прозорец ще можете да въведете заглавие, описание на страницата и ключови думи. С помощта на функцията Page Properties софтуерът автоматично ще въведе метаелементите. В други програми за създаване на сайтове може да има подобна функция за задаване на авторски настройки. Използвайте документацията или менюто Help, за да намерите този инструмент.

След като сте публикували своя сайт в световната мрежа, се уверете, че той е достъпен от вашия браузър, за да тествате всяка страница и връзките. Ако се налага да правите поправки, ще трябва да промените файловете на вашия компютър и да следвате отново процеса, за да качите променените файлове.

Започнете да споделяте адреса (URL) на вашия уебсайт с целевата аудитория. Бихте могли да го направите, като съобщите на аудиторията си, че сайтът ви вече е достъпен, или чрез поставени връзки към него в други сайтове. И при двата начина трябва да разпространите съобщение, че сайтът е достъпен за посетители.

Поддръжка: Дори когато сайтът ви се намира в световната мрежа, вашата работата не е напълно завършена. Трябва да продължите да го проверявате, за да се уверите, че функционира според предназначението си.

Важно за SEO: Включването на богат набор ключови думи и честотата на употребата им в текстовете са едни от най-важните елементи за доброто индексване на сайта в търсачките по естествен път.

Рядко текстът е единственият формат на съдържанието в един уеб сайт. Обикновено се търси оптималната комбинация между информативността, която дава текстът, и емоционалната ангажираност, която създават изображенията. Една снимка струва повече от 1000 думи дори и в уеб – стига да е подбрана правилно и използването ѝ да не нарушава нечии авторски права. За корпоративните сайтове е особено важно всички снимки да са с уредени авторски права, тъй като обратното би било сериозна заплаха за имиджа на компанията. Затова почти винаги се използват собствени снимки на компанията или такива, за които тя може да докаже, че има правото да използва (закупени, предоставени от партньори и т. н.). Разбира се, в специализираните банки на изображения има и безплатни снимки, но тяхното разнообразие и качество не винаги успяват да покрият изискванията на проекта. Заимстването на снимки и текстове от чужди сайтове или „от Интернет“ е некоректна практика и професионалните уеб

агенции и уебразработчици препоръчват на клиентите си да се въздържат от прилагането ѝ.

Какви са ползите от SEO?



Чрез SEO постигате по-добро класиране по ключовите думи в уеб търсачките, което е от голямо значение, защото най-таргетираният Ви трафик идва точно оттам.

Каква е цената за постигането на такива резултати?

Цената зависи от конкуренцията, от броя на ключовите думи, които трябва да се оптимизират, от вашия бюджет за реклама, от времето за изпълнение и т.н. Поради тази причина подходиме индивидуално към всеки клиент.

Google AdWords

Реклама в Търсачката на Google: Страницата за търсене на Google е най-посещаваният сайт в света. Всеки ден няколко милиарда потребители се обръщат към Google, търсейки това, от което се нуждаят. За бизнеса това означава най-вече едно – присъствието е задължително.

Изграждане и управление на рекламна кампания в Търсачката на Google

Цели:

- Генериране на качествен трафик от посетители към Вашия сайт
- Постигане на топ 3 позиции
- Подобряване на имиджа на компанията чрез рекламно присъствие в Google
- Активното използване на платени резултати с времето води и до увеличаване на органичния (неплатен) трафик.

Рекламата в търсачката на Google представлява текстови карета, които се задействат при въвеждането на определени ключови думи (търсения) от Вашите потребители. Всяко едно търсене задейства търг, който се провежда за части от секундата. Ако Вашата реклама отговаря на условията да участва в съответния търг, тя се задейства и позиционира съобразно нейната релевантност, рейтинг, представяне до момента, а също така и Вашия залог (bid) и качеството и бързината на сайта Ви. Плащането за рекламните карета се случва единствено тогава, когато потребител кликне на дадена реклама. Този модел на плащане е още известен като PPC (pay-per-click, или плащане на клик). Google притежава високо ефективна система за засичане на невалидни кликове – т.е. ако някой конкурент реши да клика многократно на Ваши рекламни карета, системата го засича и възстановява сумата, похарчена за невалидните кликове.



Самите рекламни карета съдържат заглавие (25 символа), 2 реда описание (2x35 символа) и линк към страницата Ви. Страницата за търсене предоставя 3 горни позиции и допълнително още 7 позиции, които могат да бъдат разпределени отдясно, но и в най-долната част на страницата. Топ 3 позициите са за предпочитане, тъй като потребителите се спират първо на тях и съответно кликат най-вече на тях.

Реклама в Дисплейната мрежа на Google С повече от 2 милиона партньорски уебсайта на издатели и много собствени като YouTube, Gmail и Android услугите на Google, Дисплейната мрежа предлага

множество начини за ангажиране на клиентите и феновете Ви.

Изграждане и управление на рекламна кампания в Дисплейната мрежа на Google

Цели:

- Генериране на качествен трафик към сайта Ви
- Подобряване на имиджа на компанията
- Позициониране на рекламното съдържание на 1-во място

Също както при рекламата в Търсачката на Google, и тук можем да ползваме текстови карета. Освен тях имаме опции за графични банери (вкл. и анимирани), видео и т.нар. rich media формати.

Рекламите в Дисплейната мрежа се задействат не при търсене от страна на потребител, а докато той разглежда сайт от мрежата на Google, който е тематично свързан с рекламата Ви.

Начините за достигане до Вашите клиенти са различни – чрез ключови думи (контекстуално), по теми и интереси, а също така и чрез директно позициониране – избор на сайтове, които позволяват Google реклама и поставяне на рекламното съдържание в техните страници.

Основни предимства на дисплейната реклама са ниските цени, широкият обхват от сайтове и богатият асортимент от начини за таргетиране на аудиторията. Плащането отново се извършва на клик.

Рекламите Ви ще излизат с приоритет в топ сайтовете на групата (abv.bg, vbox7.com, sinoptik.bg и други). Докато потребителят гледа видео, проверява си пощата или времето – Вашата Adwise реклама ще бъде там.



Набелязване на ключовите думи/фрази и планиране SEO оптимизация на сайт е важен етап от популяризацията на уеб сайта. Веднъж оптимизиран уеб сайта, той ще започне да привлича посетители и клиенти. Клиентите означават, че вие си връщате инвестицията за изработката на уеб сайта и в даден момент започвате да печелите от тази инвестиция. Но както се

казва „апетитът идва с яденето“ – вие ще искате да оптимизирате уеб сайта си повече и повече, защото това ще носи още повече клиенти и по-голяма печалба!

Ето защо набелязването на максимален набор от ключови думи и изработването на стратегия за популяризация и SEO оптимизация е важен етап, който след първоначалната инвестиция се превръща в „безплатна реклама“ – вие имате посещения безплатно, базирайки се на досегашната работа (вече заплатена), без да плащате за всяко посещение, както е при плати-на-клик (pay-per-click) рекламата! Всъщност, при силна конкуренция, може да имате нужда от ежемесечна оптимизация и съответно това да бъде под формата на месечен абонамент (но това значи много на брой клиенти, поради което бизнеса е конкурентен).

Ако планирате дългосрочен бизнес, не пропускайте ключовите думи/фрази за SEO оптимизация – те се изплащат и носят печалба с години наред, особено в по-малко конкурентни ниши, където конкуренцията е по-слаба.

Планиране на SEO оптимизация на уеб сайт е сложна дейност и се прави от експерти с опит. Нека вземем предния пример, в който конкурент инвестира много средства, за да бъде първи в Гугъл.

Предимството на даден конкурент поради много вложени средства, може да се компенсира по други начини. Например вие може да акцентирате на друг продукт/услуга повече или друга ключова дума (фраза). Например: ако конкурент излиза на първа страница по ключова дума „**лаптопи**“ (това наистина е много трудно), вие можете да акцентирате върху по-непопулярни думи и фрази като например: „лаптопи леново“, „лаптопи леново цена“, „лаптоп core i7“, „преносими компютри“, „тънък лаптоп“ и т.н. Така ако конкурента плаща на месец по 2 хиляди лева, за да излиза по общата дума/фраза „лаптопи“, вие можете срещу едва 150-300 лева да оптимизирате уеб сайта си за конкретните търсения за марки лаптопи, марка/модел процесор и т.н.

Вашият риск в случая е в пъти по-малък, и печалбата сигурно също би била по-малка, но последното не е сигурно! Реално погледнато, въпреки по-малкия брой посещения спрямо

конкурента, вашият уеб сайт ще хваща точно конкретни търсения – това са хора, които търсят конкретен модел лаптоп! Съответно ако конкурента има 20 продажби от 3000 „обща“ търсения за „лаптоп“, вие можете да реализирате също 20 продажби, но от 250 по-специфични търсения. В този случай, виждате, че въпреки по-малкия брой посещения, вие успявате с малки инвестиции да се включите в много конкурентна ниша!

Имайте предвид, че горният пример е само теория! На практика много зависи кой и как ще направи планирането на SEO оптимизацията и след това реализирането на SEO оптимизацията по изготвената стратегия!

Най-често нямате „money-back“ гаранция (за връщане на парите) за SEO оптимизация, така че рискът е за ваша сметка. Също така имайте предвид, че оптимизацията, за разлика от рекламата, изисква време за да може да даде резултати. Най-често ще трябва да планирате бюджет за минимум 3-6 месеца, с готовност да продължите безсрочно, ако възвръщаемостта е добра или отлична.

Планиране на SEO оптимизация в социалните мрежи (Фейсбук)

Въпреки че от Google имаше изявление, че линкове от социалните мрежи (Facebook, Twitter и други) не се вземат предвид за SEO оптимизация, ние не им вярваме! Досегашният ни опит показва, че е добре да се отдели поне малко внимание на това перо в SEO оптимизацията, защото има полза, но разбира се не е нужно да се прекалява и да се набляга прекалено на SEO оптимизация в социалните мрежи.

Планиране на маркетинг и реклама

Планиране на маркетинг и реклама е важна част от популяризацията на уеб сайт. Ако това е тепърва стартиращ сайт, то рекламата ще доведе първите посетители и първите потенциални клиенти в сайта, които ще завъртят колелото на Вашия бизнес в Интернет.

Уеб дизайнът на уеб сайта и лекотата на работа за посетителите под различни устройства (мобилни телефони, таблети, лаптопи и десктоп компютри), са важни както за SEO оптимизацията, а в още по-голяма степен за платената реклама. При платената реклама вие губите пари на всяко посещение (плащане на клик/Pay per click) и ако уеб сайтът не е „достатъчно атрактивен“ и „достатъчно лесен за разглеждане“ за посетителите, възвръщаемостта ще е много ниска и просто няма да успеете да върнете вложените пари за реклама.

От друга страна, съдържанието на сайта е следващото след дизайна – съдържанието трябва да е представено максимално четливо, да не натоварва очите, да се чете лесно и бързо. Само при тези изпълнени условия имате шанс за успешна реклама, за която възвръщаемостта ще е няколко пъти инвестицията.

Планиране на маркетинг и реклама в социалните мрежи (Фейсбук)

Планиране на маркетинг и реклама в социалните мрежи най-вече се отнася за Фейсбук (Facebook). Facebook е най-разпространената социална мрежа в България, с дял повече от сбора от следващите 5-6 социални мрежи взети заедно.

Маркетинг стратегията във фейсбук е по-специална в сравнение с други типове реклама и също изисква известен опит.

ОПТИМИЗАЦИЯ НА САЙТА СИ МОЖЕ ДА НАПРАВИТЕ НА ТОЗИ ЛИНК –Така ще се качи нагоре в списъка в търсачките!!!

https://www.google.com/webmasters/tools/site-message-view?hl=bg&authuser=0&siteUrl=http://www.su-ohridski.eu/&message-filter=all&grid.r=0&msgid=AB9YKzL8JNpf6lt-3tIzZ_R-szQggnJhTnLU725Bc3JfZ2eU0YPdqu0vNIQbtNtyNtY2rpt1y5byO4a8qg4-hsafXyUmYDjxng

30. Популяризиране

След като бъде създаден, сайтът трябва да бъде „публикуван“ в мрежата, т. е. да стане достъпен за своите бъдещи потребители. Необходимо е да се организира неговото „хостване“ – място в мрежата или „сървър“, на който ще се намира. Регистрира се „домейн“ (domain name), който представлява кратък и лесно запомнящ се адрес в Интернет. Уточняването на хостинга и домейна обикновено започва още по време на изграждане на сайта, тъй като това е процес, който отнема време.

Последната необходима стъпка е „**популяризирането**“ на сайта, защото неговото физическо присъствие в Интернет все още не означава, че той ще има потребители и осигурен висок трафик. Необходимо е той да бъде включен във възможно повече „търсачки“, (банери, връзки), така и по медиите (печатни и електронни). Трябва да се използват всички възможни начини, чрез които информация за съществуването на новия сайт може да достигне до потенциалните потребители.

Преработка и обновяване: Основен принцип в Мрежата е периодичното обновяване и осъвременяване на информацията и дизайна на сайта. Това е ключът за поддържане на постоянния интерес на потребителите. Промените не трябва да са драстични – това води до объркване, но трябва да са редовни.

Новини: Поставянето на новини или любопитна информация на началната страница от сайта и честото им актуализиране е своевременна покана, която „връща“ посетителите и показва, че сайтът не е „мъртъв“ (както значителна част от съдържанието в Мрежата), ако се обновява редовно.

Често задавани въпроси: Обикновено потребителите срещат еднотипни проблеми и отправят към създателите и собствениците на сайта едни и същи въпроси, с оглед улесняването на посетителите. Създаването на страница „Често задавани въпроси“, понякога под названието (FAQ) (от английски – Frequently Asked Questions), е от една страна проява на любезно отношение към посетителите, които проявяват интерес, а от друга – спестява време и усилия, необходими за отговор на всяко конкретно питане.

31. Учебна програма 1 по Уебдизайн

УТВЪРДИЛ!

Директор:

.....

(Пенко Рисков)

УЧЕБНА ПРОГРАМА (ЗАНИМАНИЯ ПО ИНТЕРЕСИ) УЕБДИЗАЙН

2 часа седмично = 72 часа общо

КРАТКО ПРЕДСТАВЯНЕ НА УЧЕБНАТА ПРОГРАМА

Обучението в модул „Уебдизайн“ е насочена към овладяване на базисни знания и умения, свързани с използването на техники и съвременни софтуерни технологии за изграждане на интерактивен интерфейс и ключови дигитални компетентности на учениците. Програмата е предназначена за обучение на ученици, които имат интереси свързани със създаването на динамични уебсайтове, както и за профилирана подготовка на ученици в гимназиален етап.

Учебното съдържание е представено в следните основни теми:

- Планиране на уебсайт
- Структура, динамика и графичен дизайн на уебсайт
- Изграждане, тестване и публикуване на уебсайт
- Софтуер за създаване на уебсайт. Системи за управление на уеб съдържание
- Уеб сигурност

В този модул, акцентът е поставен върху планирането, проектирането, графичния дизайн и изграждането на уеб сайтове чрез използване на съвременни уеб технологии и стандарти. Специално внимание е отделено и на работата със системи за управление на съдържание, като вариант за удобно и лесно създаване и поддръжка уеб сайтове. Естествено, този вариант има и недостатъци, но учениците сами ще могат да правят своя избор от технологии, след като са запознати с тях.

Първи учебен срок – 18 седмици по 2 часа седмично = 36 часа

(1)	(2)	(3)
Теми	Компетентности като очаквани резултати от обучението	Нови понятия

(1)	(2)	(3)
<p>Тема 1. ПЛАНИРАНЕ НА УЕБСАЙТ 1. Основните етапи в планирането на уеб сайт</p>	<ul style="list-style-type: none"> • Описва основните етапи и дейности в планирането на уеб сайт. • Определя темата, целите и целевата група на сайта. • Определя изискванията, на които да отговоря сайтът. • Разчита задание за изграждане на уеб сайт. 	<p>Карта на сайта;</p>
<p>2. Избор на име и регистриране на уеб сайт</p>	<ul style="list-style-type: none"> • Разбира системата за имена на домейни в интернет. • Избира подходящо име на сайт и да го регистрира. 	<p>Домейни-видове; Стандартизиран адрес на ресурс (Uniform resource locator, URL); Система за имена на домейни – Domain Name System (DNS); IP адрес; Хостинг;</p>
<p>3. Структура на уебсайта</p>	<ul style="list-style-type: none"> • Да визуализира идеите си чрез подходящи техники и технологии 	<p>Структура на сайта;</p>
<p>Тема 2. ПРОЕКТИРАНЕ И ГРАФИЧЕН ДИЗАЙН В УЕБСАЙТ 4. Основни етапи при проектирането на уебсайт.</p>	<ul style="list-style-type: none"> • Умее да избира подходящо име на сайт и да го регистрира. • Познава услугата уеб хостинг и нейните характеристики. 	<p>Dom.bg Icn.bg www.freehosting.com/ www.googlehost.com/ www.awardspac.e.co/ www.000webhost.com https://byet.host www.freehostingoads.net</p>

(1)	(2)	(3)
	<ul style="list-style-type: none"> • Умее да описва основните етапи и дейности в планирането на уеб сайт. • Умее да определя темата, целите и целевата група на сайта. 	
<p>5. Създаване на графичен модел на уебстраница</p>	<ul style="list-style-type: none"> • Описва основните етапи в проектирането на уеб сайт. • Създава диаграма, показваща зависимостите между отделните страници. • Използва подходящ софтуер за изграждане на структурата на уеб сайт. • Проектира подходяща навигационна структура на сайт и структура на системата от файлове и директории с цел оптимизиране на уеб сайта за търсещи машини. • Аргументира избраната структура на уеб сайт. • Осъзнава необходимостта от създаването на качествено съдържание за уеб сайт. 	<p>Тест за самопроверка;</p>

(1)	(2)	(3)
<p>6. Създаване на графичен модел на уебстраница-упражнение</p>	<ul style="list-style-type: none"> • Описва основните етапи в проектирането на уеб страница. • Прилага основните стандарти, свързани със структурата, графичния дизайн и графичния интерфейс на уеб страница. • Изброява специфични изисквания при проектирането на начална страница на уеб сайт. • Посочва специфични изисквания при изграждането на сайтове за хора с увреждания. 	
<p>7. Проектиране на уеб страница (упражнение)</p>	<ul style="list-style-type: none"> • Умее да създава схематичен дизайн на уеб страница чрез използване на подходящ софтуер. • Умее да изгражда структура на уеб сайт. 	
<p>8. Създаване на графичен модел на уеб страница</p>	<ul style="list-style-type: none"> • Обяснява ролята и значението на цветовете в уеб дизайна. • Използва софтуер за създаване и избор на цветови схеми. • Обяснява значението на шрифтовете в уеб дизайна. • Създава графичен модел на уеб страница чрез използване на подходящ софтуер. 	

(1)	(2)	(3)
9. Софтуер за създаване и избор на цветови схеми (упражнение)	<ul style="list-style-type: none"> • Умее да работи с конкретно приложение за създаване и избор на цветови схеми. • Умее да избира подходящ цвят от списък с цветове. • Умее да намира хармонични цветови комбинации. 	Цветова схема;
10. Създаване на графичен модел на веб страница (упражнение)	<ul style="list-style-type: none"> • Умее да създава графичен модел на веб страница с приложението Inkscape. • Проектира графичен модел на начална страница на конкретен веб сайт. 	Графичен модел (mockup); Схематичен дизайн (wireframe); Графичен потребителски интерфейс; Интерактивен интерфейс; Достъпност;
11. Проектиране и графичен дизайн на веб сайт (обобщение)	<ul style="list-style-type: none"> • Познава основните етапи при проектиране на веб сайт. • Умее да създава графичен модел на веб страница. 	Тест за самопроверка;
Тема 3. ИЗГРАЖДАНЕ, ТЕСТВАНЕ И ПУБЛИКУВАНЕ НА УЕБСАЙТ 12. Уебдизайн стандарти в уеб и основни технологии в уеб	<ul style="list-style-type: none"> • Обяснява уеб дизайн стандартите и тяхното предназначение. • Посочва основните уеб технологии в интернет. 	Hyperlink; Hypertext; Webserver; HTTP;
13. Специализирани софтуерни средства за създаване на веб сайтове	<ul style="list-style-type: none"> • Използва редактори и средства за разработка на уеб • Използва готови шаблонни форми. • Обяснява и спазва различните типове права и лицензи по отношение на 	Sublime, Notepad++, WinSCP, FileZilla, MS ExpressionWeb4, Inkscape, CSS, HTML

(1)	(2)	(3)
	използваните шаблони.	
14. Основи на HTML	<ul style="list-style-type: none"> • Използва актуална версия на езика HTML за изграждане на структурата на веб страница. • Вгражда мултимедийни блокове в вебстраница-анимация, видео, аудио. • Аргументира необходимостта от оптимизация на изображенията и мултимедия за веб. 	(Hyper Text Markup Language)HTML;
15. Основи на HTML. Тагове за текст, хипервръзка, изображение (упражнение)	<ul style="list-style-type: none"> • Умее да форматира текст в HTML. • Умее да създава хипервръзка в HTML. • Умее да използва изображения в HTML. 	хипервръзка
16. Основи на HTML. Тагове за таблица, списъци, аудио, видео (упражнение)	<ul style="list-style-type: none"> • Умее да използва тагове за таблица в HTML. • Умее да използва тагове за списъци в HTML. • Умее да използва тагове за аудио и видео в HTML. 	
17. Основи на HTML. Създаване на форми (упражнение)	<ul style="list-style-type: none"> • Умее да създава форми в HTML. 	HTML

(1)	(2)	(3)
<p>18. Основи на HTML (упражнение)</p>	<ul style="list-style-type: none"> • Умее да вгражда мултимедийни блокове в уеб страница – анимация, видео и аудио. • Умее да аргументира необходимостта от оптимизирането на изображения и мултимедия за уеб. 	<p>Мултимедия в web</p>
<p>19. Визуално оформяне на уеб сайт с използване на CSS.</p>	<ul style="list-style-type: none"> • Използва актуална версия на езика Cascading Style Sheets (CSS) за визуално оформяне на уеб страница. • Отделя информацията в уеб сайт от начина за нейното визуално представяне. 	
<p>20. Визуално оформяне на уеб сайт с използване на CSS (упражнение)</p>	<ul style="list-style-type: none"> • Умее да работи с езика Cascading Style Sheets (CSS). • Познава видовете селектори в Cascading Style Sheets (CSS). • Познава основните свойства, които се използват при работа с езика Cascading Style Sheets (CSS). 	
<p>21. Визуално оформяне на уеб сайт с използване на CSS (упражнение)</p>	<ul style="list-style-type: none"> • Умее да форматира текст с езика Cascading Style Sheets (CSS). • Умее да форматира фон с езика Cascading Style Sheets (CSS). • Умее да използва рамки с езика Cascading Style Sheets (CSS). 	

(1)	(2)	(3)
<p>22. Визуално оформяне на уеб сайт с използване на CSS</p>	<ul style="list-style-type: none"> • Добавя и използва шрифтове чрез средствата на CSS. • Използва уеб услуги за шрифтове. 	Font
<p>23. Работа по проект (изработване на уеб сайт) 24. Работа по проект (изработване на уеб сайт) 25. Работа по проект (изработване на уеб сайт) 26. Работа по проект (изработване на уеб сайт) 27. Работа по проект (изработване на уеб сайт) 28. Работа по проект (изработване на уеб сайт)</p>	<ul style="list-style-type: none"> • Сформира екип за реализиране на конкретен проект. • Избира подходящи средства за реализация на конкретен проект. • Проектира конкретен уеб сайт. • Създава информационна архитектура на уеб сайта. • Определя елементите и изгражда навигационна структура на уеб сайта. • Събира и обработва информацията, която ще се включва в уеб сайта – текст, графични изображения, мултимедийни продукти. • Изготвя схематичен дизайн (wireframe) на отделните страници. • Създава графичен модел на уеб сайта с приложението Inkscape. • Изработва уеб сайта с помощта на HTML редактор. 	Редактор на уебсайт;

Втори учебен срок – 18 седмици по 2 часа седмично = 36 часа

(3)	(4)	(5)
<p>29. Тестване и валидиране на кода на веб страница</p>	<ul style="list-style-type: none"> • Обяснява необходимостта от тестване на веб сайт преди и след неговото публикуване. • Използва инструменти за валидиране на веб страници. • Използва емулятори за тестване на веб сайт за работа с различни браузъри. • Използва инструменти за тестване на достъпността на веб сайт. 	<p>Валидиране; Браузър;</p>
<p>30. Тестване на достъпност на веб сайт (упражнение)</p>	<ul style="list-style-type: none"> • Умее да тества скорост и достъпност на веб страница. • Познава различните инструменти за анализ на браузъра Google Chrome. 	
<p>31. Публикуване на веб сайт. Авторски права</p>	<ul style="list-style-type: none"> • Публикува веб сайт. • Обяснява и спазва правните и етичните норми при публикуване на информация в интернет. Авторски права. • Умее да избира хостинг и домейн. • Познава сайтове доставящи безплатни хостинг услуги. 	<p>Авторски права</p>

(3)	(4)	(5)
<p>32. Оценка и оптимизация на уеб сайт. Анализ на посещаемостта на уеб сайт (упражнение)</p>	<ul style="list-style-type: none"> • Обяснява необходимостта от оценка и последваща оптимизация на уеб сайт. • Използва инструменти за анализ на посещаемостта на уеб сайт. • Посочва и използва инструменти за популяризиране на уеб сайт. • Изброява методики и техните специфики за оценяване на уеб сайт. • Оценява уеб сайт на базата на зададени спецификации чрез използване на специализирани инструменти. • Познава инструменти за анализ на посещаемостта на уеб сайт. • Познава различни методики за оценяване на уеб сайт 	<p>Оптимизация на сайт; Search Engine Optimization (оптимизация за търсещи машини); Рангово поддръждане (ранжиране); Really Simple Syndication (RSS);</p>
<p>33. Представяне и защита на проект пред публика</p>	<ul style="list-style-type: none"> • Описва принципите на управлението на проекти с цел проектиране, разработване и поддръжка на комплексни уеб сайтове. • Представя пред публика разработен уеб сайт. • Описва инструменти и стъпки за оценка на уеб сайт. • Оценява уеб сайт. 	
<p>Тема 4. СИСТЕМИ ЗА УПРАВЛЕНИЕ НА УЕБ СЪДЪРЖАНИЕ 34. Същност, предназначение и архитектура на система за</p>	<ul style="list-style-type: none"> • Обяснява предназначението, предимствата и недостатъците на различни системи за управление на уеб съдържание; • Описва трислойната архитектура на система за управление на уеб 	<p>Система за управление на уебсъдържанието (Content Management System, CMS);</p>

(3)	(4)	(5)
управление на уеб съдържание	<p>съдържание;</p> <ul style="list-style-type: none"> • Инсталира система за управление на уеб съдържание; 	
<p>Тема 5. УЕБСИГУРНОСТ 35. Сигурност в уеб.</p> <p>36. Какво научихте в уеб дизайн (обобщение)</p>	<ul style="list-style-type: none"> • Описва различни видове атаки, насочени срещу нормалното функциониране на сайта и начини за тяхното идентифициране. • Обяснява рисковете, които могат да причинят уеб сайтове, които използват графиките на уеб страниците като уеб маяци. • Описва методите за сигурност на информацията в уеб приложения и компрометиране поверителността на данните на потребителите чрез "бисквитки". • Описва начините за увеличаване на надеждността на сайта чрез сертификати за сигурност. • Умее да прилага използването на публичния ключ за сигурност за повишаване на сигурността. • Познава различни средства за повишаване сигурността на уеб сайт. • Познава общите проблеми със сигурността в уеб. • Умее да повишава сигурността на сайт, създаден чрез система за управление на уеб съдържание. 	<p>Бисквитки; Уеб маяци; Публичен ключ за сигурност; (САРТСНА); Сертификат за сигурност; Допълнителна функционалност (Plug-in) за защита;</p> <p>Тест за самопроверка</p>

(3)	(4)	(5)
	<ul style="list-style-type: none"> • Умее да създава резервно копие на сайт. • Умее да планира уеб сайт. • Умее да проектира уеб сайт. • Умее да създава графичен дизайн на уеб сайт. • Умее да изгражда, тества и публикува уеб сайт. • Познава системи за управление на уеб съдържание. • Познава проблемите със сигурността в уеб. 	

ИЗГОТВИЛ: Детелина Милкотева

31.1. За методиката

Представяне на основни елементи на учебно съдържание, ключово по темите:

Тема 1. ПЛАНИРАНЕ НА УЕБСАЙТ

Тема 2. ПРОЕКТИРАНЕ И ГРАФИЧЕН ДИЗАЙН В УЕБСАЙТ

Тема 3. ИЗГРАЖДАНЕ, ТЕСТВАНЕ И ПУБЛИКУВАНЕ НА УЕБ САЙТ

Тема 4. СИСТЕМИ ЗА УПРАВЛЕНИЕ НА УЕБ СЪДЪРЖАНИЕ

Тема 5. УЕБСИГУРНОСТ

Е изложено в началото на Втора глава.

Дисертацията е разработена и предназначена за учениците, учителите в прогимназиален и гимназиален етап, както и в профилирани паралелки с модул по Уебдизайн, също и за абсолютно начинаещи Уеб дизайнери. Надграждат се базисни

знания и умения, получени в задължителната подготовка по Информационни технологии. Обучението се фокусира върху темите по-горе и по-конкретно: изграждането на интерактивен интерфейс – Проект, като учениците сами или групово създават интерактивен интерфейс с предпочитани от тях технологии, след като са запознати с възможностите и палитрата от съвременни уеб средства и уеб стандарти. Проектите може да бъдат за изготвяне на динамично уеб съдържание и със статично.

Част от уроците са за нови знания, има уроци за упражнения, преговор и учебни часове за проверка и контрол-тестове, решаване на конкретни задачи, защита на проект и др.

Изграждане на интерактивен интерфейс е процес, който обхваща няколко различни аспекта, включително оформление на уеб страници, производство на съдържание и графичен дизайн. Докато термините уеб дизайн и уеб разработка често се използват взаимозаменяемо, уеб дизайнът е техническа подгрупа на по-широката категория уеб разработка.

Уеб дизайнът може да бъде доста сложен и труден, но с развитието на технологиите и проникването на интернет, уеб дизайн индустрията е нараснала в епични пропорции. Не е изненадващо, че професията уеб дизайнер се превърна в една от най – актуалните и желани от днешното поколение. Наблюдава се интерес от страна на учениците във всички възрастови групи.

Тази дисертация може да служи и като ръководство за начинаещите, които искат да се занимават с изграждане на интерактивен интерфейс.

Хората често не разбират смисъла на интерактивността. Е, какво точно е ИНТЕРАКТИВЕН ИНТЕРФЕЙС?

Visual + взаимодействие = ИНТЕРАКТИВЕН ИНТЕРФЕЙС

Интерактивния интерфейс се занимава изцяло с решаването на проблеми и това как потребителите взаимодействат с него. Много млади дизайнери често разбират погрешно концепцията за уеб дизайн с тази на изграждането на интерактивен интерфейс.

Запомнете – уеб дизайнът е свързан с дизайна на сайта, а не с кодирането и разработката му. А изграждането на интерактивен интерфейс е свързано именно с кодирането.

Разбира се, това да знаете някои кодиращи езици (HTML, CSS, Java и т.н.) е голям плюс, но не трябва да се впускате дълбоко във front-end или back-end разработката, за да сте фокусиран уеб дизайнер. Това не е ядрото на уеб дизайн.

Често срещаните видове уеб дизайн включват дизайн на електронната търговия, дизайн на фирмени уебсайтове, дизайн за промоционални нови продукти, дизайн на проекта за дейността и дизайн на уеб страницата за общото управление. Може да се сблъскате с тези видове уеб дизайн в ежедневната си работа. Като начало трябва да започнете от специфичните аспекти на софтуерните умения, оформлението, обработката на изображенията и проектирането.

Преди да станете „голям“ в уеб дизайн, първо трябва да сте отличен дизайнер. Трябва да се влюбите в дизайна като цяло – типография, усет за цветове и шрифтове. Влюбете се във всички красиви неща. Открийте красотата. Подобрете способността си да намирате красотата във всичко. Мислете за дизайна като професия, като част от живота

Ви, а не просто като работа. Ето и девет основни умения, които трябва да научите като разработчик на интерактивен интерфейс - **Принципи на визуалния дизайн**:

- Дизайн на оформлението
Цветова комбинация
Дизайн на взаимодействието
Photoshop
Основен език за кодиране
Правила за SEO
Познаване на продукта на компанията и на групата потребители
Овластяване на поне един софтуер за редактиране на кодиране

Познаването на някой език за кодиране ще ви отведе до следващото ниво – Дори и да не можете да пишете код, трябва да знаете кои езици могат да ви помогнат да постигнете желания краен ефект.

Ето и пет ключови елемента на уеб дизайна:

Оформление – Уеб оформлението се фокусира върху високата степен на единство на технологиите и изкуството. Уеб дизайнът се отнася до комбинация от визуални елементи като текст и изображения, които да направят страницата по-красива и лесна за навигация. Тя е важна форма на визуална комуникация и неразделна част от уеб дизайна.

Цвят – Цветовата комбинация на даден уебсайт отразява естетически предпочитания / стил на уеб дизайнера и пряко засяга възприятието на зрителя.

Изображения – Изображенията дават на хората визуално впечатление, което е по-интензивно от текста. Те могат да увеличат шансовете за привличане на вниманието на потребителите.

Шрифт – Персонализираните шрифтове могат да бъдат забавни или изразителни, но не винаги практични. Ако искате сайтът да е четлив и професионален, изберете най-подходящия шрифт за него.

Съдържанието - е основният и съществен елемент на уеб дизайна, тъй като то представя основното послание. Това послание трябва да е ориентирано към колокото се може повече потребители, което се осъществява чрез Responsive design или адаптивен дизайн, както го превеждаме на български език - техника за изработка на уеб сайт. Тя позволява обслужване на широк набор устройства. Адаптивният уеб сайт дизайн се намества и изглежда добре на смартфони, таблети и десктоп машини. Но той не е ориентиран само към смарт устройствата. Той решава и друг тип проблем. Десктоп машините (РС-та, лаптопи) също имат своите особености по отношение на екрани и резолюции. Вероятно сте забелязвали, че някои сайтове изглеждат мънички на екрана. Това е, защото дизайнът им е направен за ниска резолюция. Когато потребител с голям монитор и висока резолюция отвори подобен сайт, то същият изглежда като миниатюра, просто се губи на екрана. Но в същото време Вашите потребители може да използват по-стари компютри. Често срещан казус, сайтът трябва да обслужва клиенти, които се намират в двете крайности. Така поставено предизвикателството може да се разреши с използването на адаптивен дизайн. Всеки един потребител ще вижда сайта

по начин, който е адекватен за неговото компютърно устройство. Проблемът с малките, големите, новите и старите екрани ще бъде елиминиран.

Вашият интерактивен интерфейс трябва да бъде изграден с мисъл за бъдещето!

Адаптивният дизайн не обслужва единствено настоящите нужди на уеб сайта и неговите потребители. Замисълът му е да покрие и утрешните. Пазарът бързо се деформира и мобилният трафик, който получавате, ще нараства стремглаво. Адаптивният уеб сайт дизайн ще Ви помогне да посрещнете тази промяна. Responsive сайтовете гарантират на потребителя, че ще използва по адекватен начин уеб сайта без значение през какво устройство го достъпва.

Направата на адаптивен уеб сайт не е по-различна от изработка на конвенционален сайт. Вие трябва да сте наясно какъв тип потребители привличате със сайта и какви устройства използват те. Допълнително трябва да се вземе предвид, че ако пускате нова услуга или стока на пазара, може да привлечете други клиенти, които използват смартфони и планшети. Съответно тези юзъри ще искат от Вас адекватен интерактивен интерфейс, който да използват през разнородните си устройства. Бихме дефинирали изработката на адаптивен уеб сайт като един **интелигентен уебпроект**.

Водейки се от тези насоки и аспекти, не се изисква предварителни познания по HTML5, CSS3, JavaScript и др., ще успеете да изградите добър **интерактивен проект**.

Целта на обучението по Уеб Дизайн е създаване на динамичен уеб сайт с използването на HTML5, CSS3 и JavaScript, както и с множество други съвременни софтуерни технологии. Друга цел е повишване нивото на дигитални компетенции в областта на информационните технологии и Уеб.

Всяка от темите е илюстрирана с практическа реализация на фрагменти от уеб страници и създаване на функционално пълна уеб страница с HTML, CSS, JavaScript и библиотеки, jQuery и Ajax и други. Има и още технологии, които са препоръчителни:

Amaya, Web Site Wireframe Tools, Microsoft Expression Studio 4 Web, Total Validator, Cryptograph, W3C, Sublime, Notepad++, CMS, MVC.net6, WinSCP, FTP и др.

Персонално отношение при усвояване на преподавания материал.

Всички теми и подтеми са детайлно изработени и включват множество примери. По време на обучението се правят демонстрации, като всеки може да задава въпроси по всяко време, правят се видеоуроци по време на часовете с платформата на MOH MS Teams, така всеки ученик дори да отсъства може да прегледа в удобно за него време случващото се в екип WEBDESIGN. Упражненията преминават под наблюдението на учителя, демонстриращ изграждането на уебсайт, вграждането на елементи и др. Като допълнение към обучението са упражненията и се дават задачи за самостоятелна работа. Добре е учениците да имат склонност към аналитично мислене и да владеят английски на начално ниво.

32. Дидактически принципи, методи и средства

Дидактически принципи и методи на обучението по Уебдизайн:

„Дидактическите принципи са основни изходни положения за функционалната технологична организация на обучението. Правилата на обучение произтичат от

принципите и са частни положения от тях. Правилата конкретизират принципа, подчиняват му се и определят дейностите на обучавания и преподавателя за реализиране на принципа“.

Дидактическите принципи са обект на разглеждане от редица автори.

Принципът за научност при обучението по Информационни технологии означава задължително съответствие на съдържанието и методите на преподаване с равнището и изискванията на ИТ, в частност Уебдизайн, като наука в съвременния ѝ етап на развитие. Терминът научност на обучението изисква да се работи в учебните часове с такива факти, да се формират в мисленето им такива понятия, които днес са признати за научни. Преподавателят следва този принцип, ако успява да изгради коректно формулиране и определяне на основните понятия и построяване на правилни съждения и изводи. Осъществява се надграждане на базисните знания, умения и отношения по ИТ с дигитални компетенции по Уебдизайн – модулно обучение в профилирани паралелки в училищата.

Пример 1.1. Контролни въпроси върху изучавания материал по Уебдизайн:

1. Какво е Уебдизайн?
2. Какво наричаме FTP server?
3. Защо в MS Expression Web 4 е предпочитаната от Вас програма, спрямо CMS системата?

Обучението по Уебдизайн, трябва ефективно да развива мисленето на учениците, да го обогатява с нови понятия и представи, да развива творческото въображение и паметта.

Принципът на нагледност е свързан с обогатяване и разширяване на сетивния опит на учащите, в уточняване на сетивните им представи и развитието на наблюдателността. Класическото описание на принципа е свързано с имената на Ян Коменски и Й. Песталоци и др.

Пример 1.2. Задаване на първата активна форма на обучение – задача № 1.

Всеки ученик по Уебдизайн трябва да разработи за първа задача: Кои съвременни технологии бихте използвали за изграждането на интерактивен интерфейс?

Оформянето на изложението се изисква да отговаря на посочени задължителни критерии за оформяне на доклад, като предварително им са задава шаблон с нужните изисквания, където базисните умения по ИТ ще се надграждат с тези по Уебдизайн.

В часовете за практически занятия всеки ученик трябва да разгледа разработената курсова задача от негов съученик през предходната година и да състави обосновка за оценка. Насоките по изпълнението са, да се изработят индикатори за оценяване.

Включването в заданията за изпълнение на подобни практически задачи се оказва много полезна практика. Принципът на нагледност подпомага студентите да извлекат максимум полза от практическите разработки на своите съученици от предходните години, да видят и осъзнаят както допуснатите грешки, така и положителните приноси в разработката на задачата. Нейното изпълнение им се изяснява и те имат начална база за сравнение.

Принципът на съзнателност се изразява в това обучаваният да разбира всичко, което изучава, владее езика използван от преподавателя, свързва новите знания със старите. Съзнателността може да се разбира като отношение към учебната работа и разбиране на учебното съдържание.

Принципът на активността на познавателната дейност се състои в отразително-преобразуващото отношение на обучаемия към учебното съдържание.

Принципът за активност в обучението се осъществява, като изложението на всеки нов дял или тема започва с теоретична част - кратък увод в темата, в който обучаваните трябва да се включват чрез беседи, дискусии или електронни тестове. Следва

въвеждането на новите понятия, установява се връзка с предходния изучаван материал, показва се мястото и значението на тези знания в общата система от знания, посочват се областите на практическо приложение и се дават задания за практическо изпълнение, степенувани по трудност и обвързани с помощни инструкции за изпълнение.

Всички изучавани тематични модули по дисциплината Уебдизайн в упражненията са структурирани в работни файлове за изпълнение и помощ, които всеки ученик получава предварително в електронен вид. При необходимост се използват и допълнителни материали в електронен вид (техническа документация, образци, анкети, тестове, готови кодове за вграждане в уеб и др.). Всеки ученик започва самостоятелно изпълнение на учебните единици и ако прилага алгоритъма на работа правилно, след приключване на занятието, преподавателят може да анализира и архивира файловете за обратна връзка.

Чрез различни похвати и методи може да се повиши интереса на обучаваните към изучаваната тема. Подходящи в това отношение са класифицираните по трудност задачи за изпълнение в заданията, разнообразието на практическото им приложение и използване след това е допълнителен стимул. С по-голям интерес учениците подхождат и към материала, който ще се използва от тях през следващите семестри в конкретни лабораторни и практически занятия.

Пример 1.3. Използване на приложения.

Включването на аналогията от учебните модули по информатика и ИТ, усъвършенства алгоритмичното мислене на обучаваните и обогатява уменията им за използване на съвременни софтуерни технологии. Има разработени електронни приложения по изброените учебни дисциплини, които подпомагат учебната дейност, като автоматизират специфични работни процеси. Когато учениците знаят, че ще им се наложи да правят интерактивни елементи например и да ги вграждат в уеб, те стават по-внимателни и мотивирани в изпълнението на заданието.

Принципът на системност и последователност може успешно да се приложи в обучението по Уебдизайн. Особеностите на познавателната и практическата дейност при изпълнение на заданията по изучаваните модули обуславят прилагането на този принцип. Последователността в обучението по ИТ и Уебдизайн е подчинена на движение *от простото към сложното, от теорията към практиката, от известното към неизвестното, от знанието към уменията*, от технологиите към използване и прилагане на компютърните умения в дисциплини през следващите години. Този принцип изисква новоусвоените понятия от обучаваните да заемат своето логическо място в системата от познания, които те вече имат.

Принципът на диференциран (индивидуален) подход, се състои в изучаване и съобразяване с индивидуалните особености на всеки обучаван в учебния процес с цел максимално развитие на положителните и преодоляване на отрицателните индивидуални особености.

Преподавателят трябва добре да познава и наблюдава индивидуалните особености на учениците и да се съобразява с тях при избора на средства и методи на обучение .

Дидактически методи и форми на обучение по Уебдизайн - Съвременното развитие на науката и техниката изисква от преподавателите да прилагат ефективни форми на обучение, които да дават знания и умения в променящи се условия, като се акцентува на творческата активност на обучавания.

Методически насоки за обвързване на обучението по Уебдизайн с други дисциплини

Методиката на обучение по Уебдизайн – Някои методични аспекти на създаване на интерактивен интерфейс, е обширна тема, в която са свързани различни методи, форми и средства на обучение, подчинени на целите и различните начини за контрол и оценка на знанията. С проучването в тази разработка се цели разширяването на методиката на

обучение по Уебдизайн, като се създадат възможности за нейното обвързване с други дисциплини. „Естествено е да се потърси аналогия със сродните науки и преди всичко с методиката на обучението по ИТ“. Неявно, в отделните модули на дисциплината ИТ, е изградено интердисциплинарно обвързване (ИДО), а при други то се създава с цел оптимизиране и модернизиране на учебния процес.

Насоки за обвързване с дисциплината Математика

Математиката (елементарна и висша математика) е силно обвързана с ИТ и с Уебдизайн в използване на таблици, писане на формули с Equation, използване на формули, математически изрази и функции в MS Excel, използване на графики в MS Excel, Power Point и др. За да се използва правилно изчислителния ресурс по Уебдизайн от обучаваните, трябва да се следи дали се знаят и спазват основните математически понятия.

Насоки за обвързване с дисциплината Английски език

Фактът, че на английски език е основната терминология на дисциплината Уебдизайн – кодове, команди, менюта, помощна информация, използване ресурсите на Интернет и др., способства да се осъзнае взаимовръзката между двете дисциплини Уебдизайн и Английски език. Преподавателите могат да я използват за ефективно обучение, чрез различни методи базирани на конкретно теоретично и практическо обвързване.

Практическите упражнения по Уебдизайн са по два учебни часа седмично под ръководството на преподавател, като изучаваните теми са посочени в съответната учебна програма. Изборът на учебния материал изисква задълбочен анализ на идеите, методите и съдържанието на самата дисциплина Уебдизайн. Мястото, ролята и приложението на Уебдизайн в системата от дисциплини, предвидени за изучаване по учебен план, е много важно за настоящата разработка. Учениците ползват предварителни знания, които са усвоили от дисциплините Английски език, Математика, Информатика, Компютърна графика, Изобразително изкуство и други предмети изучавани по-рано и паралелно.

Чрез **входяща анкета** се проверява образователния статус и компетенциите на учениците по Уебдизайн, знанията, от които са важни за организацията на учебния процес. Чрез **изходящи електронни анкети** се определя интереса на учениците към изучаваният модул Уебдизайн, мнението им за практическата приложимост на усвоените знания и препоръките за методите и насоките на обучение. Тези анкети са вид обратна връзка с обучаваните, която се използва за оптимизиране на учебния процес през следващите години.

Методика за въвеждане на понятията в дисциплината Уебдизайн

Обучението, което приемаме за „традиционно“, непрекъснато се променя. Причините за това са изменящите се условия на живот, промените в обективните и субективни фактори, участващи в учебния процес и др. Съвременните тенденции наложиха да се разглежда учебния процес като процес на управление. Стремешът за усъвършенстване на учебния процес се оказва обвързан с алгоритмично логичния подход и неговото програмиране.

Алгоритмично логичния подход в обучението, означава да се формулират точни и ясни правила за целия обучителен период, които строго да се спазват. Тук се има предвид и използването на готови алгоритмични схеми, стига те да са се доказали през времето.

Процес на въвеждане и използване на понятията в Уебдизайна - Процесът на въвеждане и използване на понятията за дисциплината Уебдизайн е динамичен, но може да се съкращава или допълва съобразно субективизма на учебните групи и обективно изменящите се условия на обучение. След изследване на много литературни източници може да се обобщи, че всички нови понятия, въведени в процеса на обучение, трябва да изглеждат за обучаваните убедителни и полезни.

За да се постигне такова ниво на преподаване е добре да се спазват следните методически принципи:

- I. Като база да се използват вече известни понятия на учащите се при въвеждане на ново понятие;
- II. Да се класифицира групата от понятия, като всяко ново понятие да е ясно в каква група от известни понятия попада;
- III. Въвеждането на ново понятие да става чрез използване на модели от реалната действителност, а още по-добре свързани с изучаваните специалности. Така преподавателят е убедителен и лесно може да обоснове необходимостта от изучаваното понятие;
- IV. Да предпазваме обучаваните от некоректни тълкувания на свойствата на изучаваните понятия, да се спазва строгостта и систематичността на излагане на учебния материал;
- V. Да се търси връзка с понятия, изучавани в други дисциплини и тяхното ново тълкуване в дисциплината Уебдизайн.

Конкретно индуктивния метод се състои в определяне на обема и съдържанието на понятието чрез разглеждане на множество примери и контрапримери (по възможност разнообразни). При него обикновено най-напред се дават частни примери на разглежданото понятие, посочват се обекти от неговия обем или варианти на неговото приложение. Следва да се формулира дефиниция или общ вид за използването му и се изследват свойствата на въвежданото понятие.

Проблемно – базирания метод, който е конструктивистки конкретно. Индуктивен подход се прилага успешно в компютърното обучение

Градивна структура на основните понятия: Голяма роля в процеса на формиране на понятията играят словесното и символичното им изразяване. Научните термини са думите, чрез които се означават определени понятия в различните области на науката и техниката. Добре е да се изгради навик за използване на речник за научните термини от обучаваните. Например за понятието „УебФайл“ в речника за научните термини на Е. Б. Уваров, Алън Айзакс (издателство „Петър Берон“, София 1992) се правят препратки към памет, външна памет и изчислителна машина. Ако обучаваният има пропуски в усвояването на някои понятия, може лесно и точно да запълни всички празноти и да се усъвършенства.

Когато се изучават понятията е добре да се обсъждат съществуващите синоними и омоними за някои от тях. Това може да се използва за създаване на навици за работа с различните онлайн речници от уметиците.

Типични грешки при овладяване на понятията по Уебдизайн: При овладяване на понятията може да се допуснат грешки в съдържанието, обема, термина или в самото определение.

Рамки за изпълнителни дейности – Структуриране:

Обучението по Уебдизайн, се подчинява на общоприетия стандарт за обучение, съчетаващ лекции, беседи, практически упражнения и задачи за самостоятелна разработка. Планирането и разработването на учебните теми за лекционното съдържание и практическите упражнения на учениците е съобразено с учебния план за Уебдизайн в СУ „Св. Софроний Врачански“ - Пловдив. За всяка тема предварителното обмисляне цели да гарантира нейното изпълнение да протича стройно и организирано. В рамките на една тема градивните елементи се избират от преподавателя в съответствие със съдържанието на темата, нейната продължителност и целите.

Планът за темата се дава в началото на темата или в края на приключващата предходна тема, като обхваща различен брой учебни часове. Целта е, учениците да знаят

върху какъв материал ще работят и при желание да се подготвят теоретично или практически у дома с части от посочените в плана единици.

Въвеждащи дейности: Беседата и дискусиата се използват за припомняне на теоретичния материал, който е включен в темата и ще се използва в личните задания за изпълнение. Провеждат се с цялата работна група, но не са задължителни за всички учебни часове.

Лични задания: Проектирането и изграждането на заданията е сложен процес, но влияе много силно върху стила на преподаване. Трудността е в началото докато се създаде база от подходящо избрани задачи, след това процесът на надграждане е по-лесен. Личните задания са номерирани от 1 до k, и трябва да се отработват последователно. Може в една тема да има само едно задание. Резултатите от изпълнението на заданията формира за всеки ученик нивото, което е достигнал. Този модел на работа е удачен, защото води до прозрачност на естественото отсяване на учениците в различни нива, което се вижда и от самите обучавани. Обучаваните разбират, че отговорността за изпълнение на заданията е лична и трябва да потърсят начини да наваксат пропуските, които имат, ако искат да се представят добре. Най - голяма тежест от изпълнението на индивидуалните задания се дава на задачите за изпълнение.

Задачи за изпълнение – Избор:

Изборът на задачи за изпълнение, тяхното класифициране и вмъкване в структурата на заданията е едно от най-трудоемките задължения за всеки преподавател. В Дурева Д. Проблеми от методиката на обучение по информатика и ИТ, Университетско издателство „Н. Рилски“, Благоевград, 2003 са представени общи изисквания при конструирането на система от задачи за обучение по информатика и ИТ.

Тази дейност е динамична, подчинена на следните критерии - **Критерии за избор на задачи за изпълнение:**

- ✓ Изборът на задачите за изпълнение зависи от съдържанието на изучаваната тема по учебен план, целите на занятието и спецификата на прилаганите методи.
- ✓ Друг критерий е класификация на заданията по трудност. В последното задание са задачите с най-голяма трудност.
- ✓ Проучват се възможностите за обвързаност на задачата с реални практически дейности и производства.
- ✓ Да се търси обучаващия характер на всяка задача и възможностите, чрез нейното решение учениците да надграждат уменията да се ориентират в проблемни ситуации и да излизат от тях като победители. Трябва винаги да се обмисля предназначението на всяка задача, което зависи от мястото и ролята ѝ в множеството от всички задачи. Проследяване броя на учениците, които откриват и използват аналогията при работа с подобни задачи. Обучаваните трябва не само да натрупват определени сведения, които се отнасят до различни методи на решение, но да успяват да подредят и класифицират тази информация по нейната значимост.
- ✓ Разнообразие от задачи, с което да се стимулира устойчив интерес към учебната работа, творческо отношение към поставените учебни цели и начини за формиране в обучаваните на способности за самостоятелно самообучение;
- ✓ Анализ от преподавателя за избора на задачата, неговите очаквания, реалните резултати от изпълнението в предходните години и възможности за оптимизиране. Изводи за използването на наличните до момента знания и методи от обучаваните при решението на задачата. Вземане на решение от преподавателя за даване на допълнителни указания или припомняне на изучаван материал (това зависи и от нивата на знания от страна на учениците в групата по WEGDESIGN).

Постановка на задача за изпълнение: За рационалното построение на учебната дейност е необходимо да знаем какви най-съществени изисквания трябва да

удовлетворяват организацията на съдържанието и методите на обучение. Във всяка задача се проверяват знанията и уменията на учащите се (по материала от миналото или от текущия урок), опознава се новото (формират се понятия, откриват се и се обосновават закономерности и алгоритми) и се извършва затвърдяване на изученото (повторение, прилагане на изученото от други задачи).

SWOT анализ: SWOT анализът може да се използва за изготвяне на преценка и поставяне на оценка върху разработка както от преподавателя, така и от учениците. В теорията на психолога Бенджамин Блум „Таксономия на образователните цели за познавателна дейност“ оценката е поставена на върха в йерархичната класификация на целите и резултатите в обучението.

Индикаторите в учебната дейност по ИТ: Индикаторите се използват за измерване на степента на развитие и служат за проверка последователността и ефективността от определени интервенции. Доброто количествено измерване е основа за осъществяване на ефективен мониторинг и оценка на постиженията от обучаваните и дава ясна представа на всички заинтересовани от процеса на обучение за това какви са целите на учебната програма и какво трябва да се постигне.

Скала за усвояване на учебния материал. Нива към пътя на познанието

Проверката на знанията, уменията и навиците за самостоятелна работа на учащите се по дисциплината Уебдизайн, дава степента на усвояване на учебния материал. Структурирането на задачите за изпълнение в предложената методика предлагат възможност, от изпълнението на заданията в часовете за практически упражнения, всеки от групата да може да си направи точна самооценка за нивото на усвоени знания от всяко практическо занятие по Уебдизайн. Преподавателят има възможност да следи равнището на развитие за всеки обучаван по резултатите от изпълнението на заданията и от участието му в останалите методически дейности.

Системната текуща проверка на знанията на учениците от всяко практическо занятие в разработената методика създава възможност преподавателят да получава непрекъснат обратен поток от данни за пропуски в усвояването на учебния материал, да търси начини за отстраняването им и за подобряване успеваемостта на учебните групи.

Приложение на математическите понятия и методи в дисциплината Уебдизайн Вграждане на математическото мислене в учебните модули -

Всяко общество се нуждае от специалисти, които имат навици за точно логическо мислене, от задълбочени математически знания и умения, да виждат и реализират възможностите за прилагане на математиката в различни конкретни ситуации. Традиционният начин на преподаване и извършване на математическите доказателства се променя, като някои от тях могат да се извършат само с помощта на компютър. Математическите умения дават възможност за изграждане на добър, стилизиран и издържан чисто технологично интерактивен интерфейс.

Езикът на математиката и неговата роля за обучението по Уебдизайн

Едно от най-големите постижения на науката е разработката на изкуствения език на символите и формулите, което е и определящо за по-нататъшното развитие на математиката. С бързото развитие на новите технологии математиката все по-ясно се откроява не като съвкупност от факти и методи, а като език за описване на фактите и методите от най-различни области на науката и практическата дейност.

Изграждане на навици за математическа дейност при обучението по Уебдизайн

Часовете по Уебдизайн са удобни за постепенно формиране и развитие у ученците на фундаменталните логически структури, които са в основата на математическата дейност и са много важно средство за обучение. За преподавателите е по-трудно да преподават в

стил на откривателство, отколкото в посока на заучаване и използване на готови схеми от знания. За учениците е по-леко в процеса на откриване на самата истина, да запомнят системата от твърдения и доказателства, защото разбират техния произход, значение и взаимна връзка. Математиката развива умения като последователност, което се използват и в Уебдизайна.

Изграждането на интерактивен интерфейс е доста обширна тема, включва и много технологични средства и може да се асоциира с Уебдизайн, който от своя страна включва други учебни дисциплини:

Графичен дизайн – световната мрежа е визуална среда, за това се изисква специално внимание на дизайна на външния вид на уебстраниците. Графичния дизайнер взема решения относно всичко, което ще се вижда в уебстраницата: изображения, текст, цветове, разположение на обектите, интерактивните елементи, анимираните банери, бутони и др.

Информационен дизайн – това е аспект от уебдизайна, който много често се пренебрегва. Той включва организиране на съдържанието и определя как да се стигне до него с помощта на диаграми, блок – схеми и др.

Дизайн на интерфейса – определя как работи страницата, възможност за използване, колко лесно потребителят може да постигне целите си. Интерфейса на уебсайта включва бутони, хипервръзки, навигация, цялостно и функционално организиране на страницата.

Основните технологии за изграждането на интерактивен интерфейс са посочени по-долу.

HTML

HTML е езикът, който е в основата на създаването на всяка уеб страница и се разглежда детайлно. На базата на тази основа се изучава HTML графика – canvas и SVG, а така също и HTML медия и HTML API – приложен програмен интерфейс, за създаване на приложения като гео локация, drag/drop и др.

CSS

Cascading Style Sheets (CSS) – каскадните стилове е технология в уеб дизайна за форматиране на съдържанието на документите и за тяхното структуриране – задаване на разположението на HTML елементите в състава на страницата – page layout. CSS3, нови стандарти на базата на придобитите знания. Адаптивен – Responsive уеб-дизайн (RWD) с използване на CSS. Преглед на платформи за RWD – W3.CSS и Bootstrap.

JavaScript

JavaScript е език за програмиране, използван в HTML и Web. В учебните часове се изучават основите на JavaScript. Знанията се надграждат с разглеждане на начините за създаване на потребителски интерфейс и приложения.

В обучението се включва и работа със системи за управление на съдържание, за да могат учениците да разграничат видовете уебсайтове и сами да решават и планират работата си. Очакваните резултати от обучението е изграждане на умения за работа със различни видове технологии, изграждане на цялостен интерактивен проект, работа в екип и повишаване нивото на дигиталните им възможности, като предпоставка за бъдещо развитие.

IV. ГЛАВА ТРЕТА. ОПИСАНИЕ НА ПЕДАГОГИЧЕСКОТО ИЗСЛЕДВАНЕ - ОБРАБОТКА И АНАЛИЗ НА ПОЛУЧЕНИТЕ ДАННИ ОТ ИЗСЛЕДВАНЕТО И ВКЛЮЧВАНЕТО ИМ В УЧЕБНИЯ ПРОЦЕС

1. Общи положения

Чрез настоящата разработка се предлага компютърно базирано структуриране, моделиране, организиране и поднасяне на учебния материал по дисциплината Уебдизайн. Предвидените тематичните модули, според държавните изисквания за обучение, се раздробяват на отделни учебни единици, като се търси за всяка от тях реално практическо приложение. Основните стратегически оси, в този процес на изследователско търсене и прилагане на насоки за интегриране на софтуерните технологии нужни за изграждането на Уебдизайн с други дисциплини или реални практически проблеми, са следните:

- Разбиране основите на съвременните софтуерни технологии, като мост за овладяване на новите уеб технологии и уеб стандарти за умелото им внедряване в компютърно базираното обучение на учениците по Уебдизайн.
- Изграждане на нови стратегии и техники на учене, с цел завършващите ученици изпреварващо да са подготвени и да покриват изискванията от познания в областта на ИКТ, необходими за трудовата им реализация.
- **Да се изградят трайни навици от всеки ученик да използва ИКТ** за непрестанно обогатяване на знанията и уменията си в професионално и научно направление. Подходящи в тази насока се оказват почти всички учебни модули, като специален акцент може да се постави на интернет технологиите за търсене, обработка и съхранение на информация и представянето ѝ във вид на интерактивна презентация, като се осигурява сигурност в уеб.
- **Изграждане на нов тип мислене от учениците в процеса на обучение** - Чрез решаването на конкретни практически или научни проблеми, да се открият начини за стимулиране на личните инициативи от обучаваните в използването на ИКТ, като по този начин се достигнат нови нива на познание. За тази стратегическа ос подходящи градивни елементи на учебните единици се оказват разнообразните курсови задачи, електронните приложения, с които учениците получават самооценка за дейността си и богатия набор от задания в часовете за упражнения по Уебдизайн.
- Повишаване ефективността от дейностите на учениците, като се концентрират усилията им в решаването на конкретни задачи, необходими за изграждането на успешен **интерактивен интерфейс**. Всички учебни модули и техните учебни единици са подчинени на тази стратегическа ос с богат набор от практически задания, указания за изпълнение и електронни образци в часовете за упражнения.
- Непрестанно развитие на комуникативните умения на обучаваните, насочени към овладяването на други знания и умения от различни области на човешкото познание.

В педагогическото изследване се разглежда дисциплината Уебдизайн като звено в съвкупността от знания, предвидени за изучаване от учениците по учебна програма и дава насоки за изграждането на интерактивен интерфейс. Търсят се начините за опознаване на връзките на ИТ със съвременните софтуерни уеб технологии и уеб стандарти, със специализираната за учениците информация за различни приложения и среди за изграждане на **интерактивни презентации**. Стремещът на учениците е да ги

прилагат като образци за реализиране на уеб процеси, за работа с различни технологии през следващите години на обучение и в практическата си дейност след това.

1.1. Организация на методическото изследване

Организираната практическа дейност на учениците в часовете по Информационни технологии като базисни знания, умения и дигитални компетенции, в частност и по Уебдизайн е важен източник на методически умения, навици, които изграждат творческо отношение към използването на съвременни софтуерни технологии и изграждането на интерактивен интерфейс.

Чрез включване на актуални знания по темата в съдържанието на дисциплината Уебдизайн, се възпитава устойчив интерес към изучаваните уебпроцеси, компетентно отношение към учебната дейност, свързана с използване на компютри и се стимулира самостоятелното изучаване на различни софтуерни технологии, среди, средства, стандарти и пр.

Целта на проучването е да се представи планирането и извършването на контролни експерименти относно формулираната в увода хипотеза, оформяне на изводите и обосноваването им. Да се изследва влиянието от прилагането на подобрената методика на обучение по дисциплината Уебдизайн с наличието на разнообразни електронни приложения, методи и средства на преподаване, входящ и изходящ контрол под формата на тестове, задачи, проекти и др. Да се проследи мястото и ролята от използването на технологии, средства, стандарти и анкетното изследване от учениците за обвързване на обучението с бъдеща реализация в ИТ сектора и по-конкретно в Уебдизайна, което напоследък е доста актуално и търсено умение за изграждането му.

1.2. Планиране на експерименти за проверка на хипотезата

За изследването се използва дисциплината Уебдизайн, като уебпроцесът на планиране и реализация на експеримента обхваща следните насоки:

- ✓ Създават се електронни ресурси, които се ползват от всеки ученик за практическите упражнения, които са доближени с предложените задачи по спецификата на изучаваните от учениците теми от учебната програма.
- ✓ Разработени са и се използват електронни тестове за самостоятелна проверка на знанията от всеки ученик, преди контролните работи.
- ✓ Тече непрекъснат процес на проучване, взаимовръзка между дисциплината ИТ с Уебдизайн.
- ✓ Създава се база от нови понятия, технологии и средства, използвани за изучаване по учебна програма и прилагани в процеса на обучение.
- ✓ Поощрява се участието на учениците в процеса на създаване на интерактивен интерфейс, използват се компютри в извънучебни мероприятия и дейности.

Проучват се възможностите за използване уебтехнологии, използване на Internet при нужда от ресурси.

1.3. Организация и провеждане на експериментите - Дисциплината ИТ дава възможности за разширяване и задълбочаване на знанията на учениците по Уебдизайн. Дълги години се занимавах с преподаване на Уебдизайн в моята повече от 15-сет годишна педагогическа практика, още преди да съществуваше модул по Уебдизайн в профилираните паралелки. Оформях групи за допълнително обучение по интереси в училищата, където името на групите беше "WEBDESIGN", а една от предпоследните се казваше „Клуб по изграждане на интерактивен интерфейс“. Целевата група ученици в настоящия експеримент са учениците от НХГ „Цанко Лавренов“ – Пловдив, СУ „Св. Софроний Врачански“ – Пловдив и ОУ „Цар Симеон I“ - Павелско, като се използват възможностите за компютърна обработка, съхранение, събиране и разпространение на данни и информация. Тя може да стане свързващо звено в учебния процес на учениците, ако се постигне успешно

взаимодействие между учениците, програмните ресурси и човешкия интелект за изграждането на интерактивна уебсреда.

Експерименталната работа за проверка на изложената хипотеза започва през учебната 2015/2016 г. в НХГ „Цанко Лавренов“, с оформяне на методически материали за научното изследване. В учебния процес се използват различни електронни приложения (тестове, анкети, учебни модули, различни източници на информация и др.), за разширяване и задълбочаване знанията на учениците, за промените, които предизвикват новите технологии в ежедневието и професионалната дейност, с цел:

- ✓ Бързо и лесно събиране, съхранение, обработка и представяне на информация от различни сфери на теорията и практиката под формата на интерактивни уебсайтове;
- ✓ Помощни средства за логическа организация на учебния процес, за демонстрация на иновативни модели в оценяването на извършени дейности, за анализиране и представяне на данни;
- ✓ Примерни модели на програмни приложения и алгоритмични подходи, които могат да се използват като образци от обучаваните в аналогични ситуации и дейности.

Педагогическото изследване е проведено с всички ученици от осми клас в художествената гимназия „Цанко Лавренов“ - Пловдив, както и със сборни групи в прогимназиален етап в ОУ „Цар Симеон I“ – Павелско и прогимназиален и гимназиален етап в СУ „Св. Софроний Врачански“ - Пловдив.

Група 1: WEBDESIGN – ученици, които се обучават по стандартната методика.

Група 2: Клуб - изграждане на интерактивен интерфейс – ученици, които се обучават с предложената от докторанта методика, с използване на дейности за обвързване на обучението по дисциплината ИТ и работа по проект „Образование за утрешни ден“.

Група 3: WEBDESIGN - ученици, които се обучават с предложената от докторанта методика.

Броят на учениците достига необходимият брой изследвани лица, което осигурява репрезентативността на извадката, а именно групите са били с над минималния брой, който е десет ученика.

Състав на групите по Уебдизайн: Група 1 – 12 ученика, Група 2 – 13, Група 3 - 19. Има и други групи, но сега не ги включвам като цели.

2. Анализ на резултатите от изследването

2.1. Статистическо представяне на резултатите от проведеното изследване

В изследването се анализират и обобщават данните за трите групи по WEBDESIGN. Контролните групи са разгледаните по-горе, отнесени към трите училища (*виж таблица 9*).

Таблица 9. Групи WEBDESIGN и Клуб – изграждане на интерактивен интерфейс за статистическо педагогическо изследване

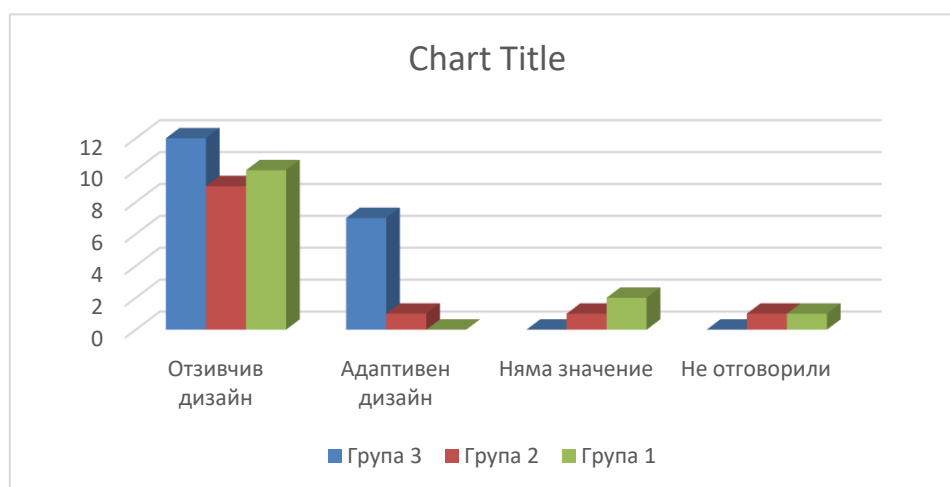
Групи WEBDESIGN и Клуб - изграждане на интерактивен интерфейс в училище				
№	Контролна група	Целева група ученици от училището	Брой ученици	
			Входящо ниво	Изходящо ниво
1	Група 1	НХГ „Цанко Лавренов“ – Пловдив, осми клас	12	12
2	Група 2	ОУ „Цар Симеон I“ – Павелско, сборна група от V, VI и VII клас	13	13
3	Група 3	СУ „Св. Софроний Врачански“ – Пловдив, сборна група V, VII, VIII и IX клас	19	19

Всички групи са наблюдавани на **входящо** и **изходящо** ниво. За посочените групи са събрани и анализирани данни, които позволяват да се проследи използването на общи методически взаимовръзки в обучението по дисциплината Уебдизайн, с цел интегрирането и с други учебни дисциплини.

Учениците от **трета група**, са включени в изследователски дейности по събиране на информация по конкретна тема за проект. Те трябва да проведат проучване чрез електронна анкета и интервю за аспектите от използване на съвременни софтуерни технологии и начините за изграждане на интерактивен интерфейс. Целта е да се изгради в обучаваните представа, че уеб процесът за адаптиране на обучението по Уебдизайн е сложен процес, в който трябва да се включат всички заинтересовани лица и да се повиши качеството на предоставяното обучение.

Въпрос 7: Кой дизайн да изберем? Отзивчив или адаптивен? Защо?	Отзивчив дизайн	Адаптивен дизайн	Няма значение	Не отговорили
Група 3	12	7	0	0
Група 2	9	1	1	1
Група 1	10	0	2	1

Таблица 9.1. Данни за отговорите на Въпрос №7 от изходящата анкета от изследването.



Фиг. 21.1 Графично представяне на отговори на въпрос №7: Отзивчив или адаптивен дизайн?

На базата на информацията за създаване на анкета за педагогическо изследване, представена по-долу, се изготвиха **Входяща анкета: Образователен статус-диагностика и компетенции (виж Приложение 6)**, както и **Изходяща анкета – Интересът по Уебдизайн (Приложение 7)**.

В педагогическите изследвания анкетните методи се използват за установяване на мнение, отношение, оценка и са насочени към разкриването на количествени и качествени характеристики на обекта и предмета на изследването. Анкетните методи са полезни за събиране на големи масиви от данни. По-малко полезни са при изследване на

даден проблем в дълбочина - примерно ако се изследват характеристиките на учебния процес, механизмите за избор на методически варианти и т.н.

Под анкетни методи се разбира голяма група методи, основани на непосредствено (беседа, интервю) или опосредствено (анкета, въпросник) взаимодействие на изследователя с респондента (лице - източник на информация), при което изследователят се обръща към него с въпроси. Всички анкетни методи представляват структурно организирани от единен изследователски замисъл системи от въпроси, непосредствено свързани с целта на изследването.

В педагогическите изследвания анкетните методи се използват за установяване на мнение, отношение, оценка и са насочени към разкриването на количествени и качествени характеристики на обекта и предмета на изследването. Респонденти в педагогическите изследвания могат да бъдат деца, ученици, учители, родители др., имащи отношение към провежданото изследване. Трябва да се има предвид спецификата на децата - респонденти (ниска грамотност, малък жизнен опит, спонтанност на реакцията, особени задръжки и цензурни механизми). Това налага особено щателна подготовка, внимателна работа на изследователя, аргументирано избиране на вида и варианта на метода. Например, когато респондентите са деца от детската градина или ученици от начална училищна възраст е подходящо да се премине към интервю, в което преобладават картинни изображения, схеми, снимков материал и други варианти на визуализация. Има различни класификации на анкетните методи. В настоящата статия приемаме, че те са писмени и устни, контактни и задочни, индивидуални и групови. Анкетните методи са полезни за събиране на големи масиви от данни. По-малко полезни са при изследване на даден проблем в дълбочина - примерно ако се изследват характеристиките на учебния процес, механизмите за избор на методически варианти и т.н.

Анкетата е форма на допитване, осъществяваща се по принцип задочно, т.е. без пряк и непосредствен контакт между интервюиращия и респондентите. Това определение отразява функционалния характер на анкетата и насочва към организацията и принципите на провеждане. Структурата на анкетата проличава от следната дефиниция: Анкетата е система от въпроси и отговори, целяща да установи мнение или отношение на групи изследвани лица (респонденти). Според обхвата на респондентите и начина на провеждане анкетата може да бъде групова и индивидуална, еднократна и панелна, чрез анкетъори и др. Комбинациите на анкета и интервю много широко се използват в педагогическите изследвания.

Друга класификация на анкетите може да бъде направена в зависимост от типа въпроси и предлаганите към тях отговори:

- стандартизираната (само от закрити въпроси),
- нестандартизираната (само от открити въпроси) и
- частично стандартизираната анкета.

В структурно отношение анкетата е съставена от два раздела:

- уводно-конструктивен;
- въпросник.

Уводно-конструктивният раздел е задължителен за успешното провеждане на всяка анкета. Чрез него респондентите се запознават с автора на анкетата, целите на провеждането ѝ, за какво ще се използват резултатите от анкетирането и се ориентират в начините за даване на отговор. Така участниците в анкетирането се убеждават в добронамереността на изследователя и се мотивират за коректно и искрено попълване на въпросника.

2.2. Резултати от входящите анкети на учениците

Входящите анкети се използват за да се провери началното ниво от знания на учениците в използване на:

- Софтуерните технологии, уеб среди, уеб стандарти, уеб принципи и тяхното структурно многообразие;
- Програмни средства и самооценка за степента на владееене;
- Английския език и неговото използване в терминологията, софтуерното и хардуерно обслужване на компютърните системи;
- Internet, дефиниране на индивидуалното разбиране за световна мрежа и ролята и в субективния свят на ученика;
- Собствен персонален компютър, неговите хардуерни характеристики и софтуерна обезпеченост. Начини и продължителност на използване, и работа;
- Използване на електронна поща, актуални електронни адреси, електронни ресурси и др.;
- Въпроси за проверка на логическите и елементарни математически познания, както и на дигиталните компетенции;
- Да се проследи инициативността в задаване на въпроси, препоръки и споделяне на собственото мнение с това на преподавателя;

Резултатите от тези анкети не са определящи в организацията на обучителния процес. Те са възможност, за изграждане на алгоритмична стратегия, за поднасяне на новите знания с етапи на надграждане на открити пропуски в знанията на учениците. Придобива се представа и за броя на учениците, които изпреварват общото ниво с продължителност и по-голям обем на усвоено учебно съдържание от учебната програма.

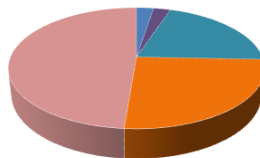
В изследванията се използва вариационния ред от номера на учениците (и съответните им порядкови статистики), които лесно могат да се включат в статистическа обработка по ключовете за учебна година, пореден номер в практическа група по Уебдизайн. Статистическите изводи са построени чрез статистически оценки на параметри. Представянето на данните е автоматизирано чрез програмата MS Excel и надграждано от 2021 година. За всяка учебна година входните данни след анкетирането се задават в MS Excel, отразяват се актуализациите, и надграждането се прави с цел оптимизиране. Обработката става много бързо, като се използва изчислителния и графичен апарат на MS Excel и резултатите се представят чрез презентации по групи. Те се оказват добра база за включване в учебния процес на учениците.

Направените изводи от входящите анкети влияят на:

- ✓ Методическата организация на учебните занятия;
- ✓ Определяне нивата на интердисциплинарна обвързаност с конкретни учебни дисциплини;
- ✓ Подбора на задачи;
- ✓ На моделите за самостоятелна работа в учебните часове и работа в групи;
- ✓ На моделите за обвързване на обучението по ИТ с Уебдизайна.

Примерът за обработка и представяне на данните е удачен да се използва пред учениците в часовете по ИТ като базисни знания за Уебдизайна.

АНКЕТА ПО УЕБДИЗАЙН - ДИАГРАМНО ПРЕДСТАВЯНЕ НА УЧАСТНИЦИТЕ ДАЛИ ОТГОВОРИ НА ВЪПРОСИТЕ С НАЙ-ГОЛЯМ ДЯЛ НА ПОЛОЖИТЕЛНИ ООТГОВОРИ, СЛЕДВАТ ОТГОВОРИ В СРЕДНА СТЕПЕН И НАЙ-МАЛЪК ДЯЛ СЕ ДАВАТ НА ОТРИЦАТЕЛНИТЕ И НА НЕ ОТГОВОРИЛИТЕ



- | | | | |
|--|-----------------|------------------------------------|-----------------------|
| ■ АНКЕТА ПО WEBDESIGN | 05.11.2021 г. № | ■ АНКЕТА ПО WEBDESIGN | 05.11.2021 г. Въпроси |
| ■ 19 ученика Габриела Атанасова Попова | | ■ 19 ученика Кристияна Райковска | |
| ■ 19 ученика Димо сарафов | | ■ 19 ученика Никол Хамбарлийска | |
| ■ 19 ученика Димитър Ташев | | ■ 19 ученика Стефан Евтимов | |
| ■ 19 ученика Алекс Скачков | | ■ 19 ученика Антонио Христов | |
| ■ 19 ученика София Христова | | ■ 19 ученика Димитър Христов | |
| ■ 19 ученика Николета Хилева | | ■ 19 ученика Яна Нейкова | |
| ■ 19 ученика Кристиан Ценов | | ■ 19 ученика Стилян Стойков | |
| ■ 19 ученика Мариета хаджиминева | | ■ 19 ученика Йордан Стоянов Иванов | |
| ■ 19 ученика Златина Станкова | | ■ 19 ученика Невена Станева | |
| ■ 19 ученика Асен Атанасов | | ■ 1 не | |
| ■ 2 да | | ■ 3 Нито Не, нито Да | |
| ■ 4 не отговорил | | ■ 4 ОБЩО | |

Фиг. 22 Анкетирани ученици от Група 3: WEBDEWSIGN – занимания по интереси в СУ „Св. Софроний Врачански“ - Пловдив с дял на положителни, отрицателни и други отговори

Като пример за използване на доверителни интервали за статистическите изследвания, при входящата анкета, може да се разгледа обработката на данните и извеждането на резултатите от **Въпрос № 7**, представени на **фиг. 18**.

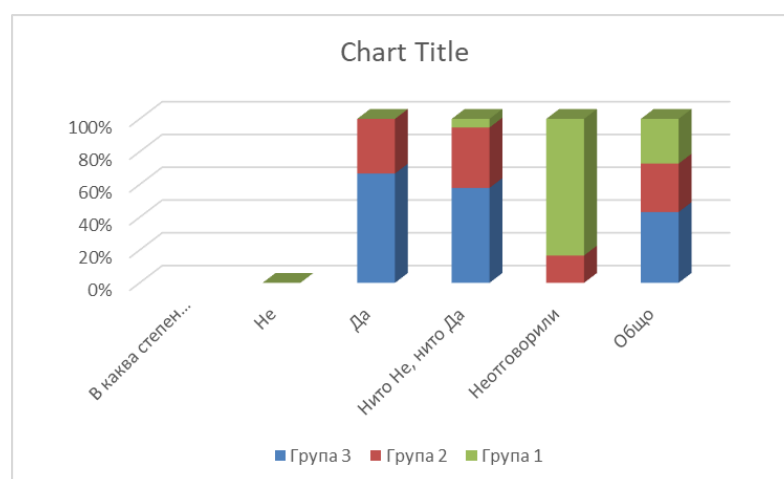
Фиг. 23 Изучавали ли сте дисциплината Английски език, ако „да“ посочете колко години сте го изучавали?





WEBDESIG N	не изуча- вали	изучавали по- малко от 3 год.	изучавали по-малко от 4 год.	изучавали по- малко от 7 год.	няма отговор
Група 1	0	0	1	10	1
Група 2	0	0	1	12	0
Група 3	0	1	7	11	0

На **фиг. 24** е изобразена диаграма на отговорите на учениците от входящата анкета на **Въпрос №4**: В каква степен владеете и използвате програмните продукти на Microsoft Office (Microsoft Word, Microsoft Excel, Power Point, MS Publisher и други от пакета)? .



Фиг. 24 Диаграма на отговорите на учениците на **Въпрос 4**

Доверителните интервали се избират от преподавателя, но независимо дали данните са графично изобразени или таблично представени, те дават много ясна представа за голямата диференцираност в състава на учениците по отношение на знанията им по Английски език. За дисциплината Уебдизайн, голяма част от

терминологията е на английски език. Предварителните знания по езика са важен фактор за доброто боравене с компютърната терминология, с много основни понятия и с помощната информация по изучавания модул. Това е основната причина да се прави проучване и анализиране на данните за степента, до която отделните ученици могат да се справят с чуждоезиковите изисквания, при овладяване на учебния материал по дисциплината Уебдизайн, тъй като доста от източниците и ресурсите за изграждането на интерактивен интерфейс са на английски език.

Основни критерии, включени във входящата анкета:

1. Степен на владееене на алгоритмичните структури за изграждане и структуриране на интерактивен динамичен уебсайт;
2. Изучавани езици за програмиране, самооценка за достигнато ниво;
3. Използване на персонален компютър. Какво се знае за неговите характеристики;
4. Степен на владееене и използване на програмните продукти на Microsoft office (Microsoft Word, Microsoft Excel, Power Point и други уеб технологии, среди, стандарти, принципи);
5. Наличие на електронна поща и нейното използване;
6. Дефиниране на Internet и използване на интернет ресурсите;
7. Владееене на Английски език;
8. Логическо мислене и реакции при необходимост за изразяване на лични позиции и компетенции;
9. Самостоятелна работа и самоподготовка – разучаване на съвременни софтуерни технологии за изграждане на интерактивна среда;
10. Изпълнение на поставените задачи и самоконтрол;

При статистическата обработка на отговорите се използва следното кодиране на данните, което се отразява и в графичното им представяне.

- ▶ Не, означава че отговорът на въпроса е “не”;
- ▶ Да, означава че отговорът на въпроса е “да”, но липсват каквито и да е пояснения, които се изискват при положителен отговор;
- ▶ Нито Да, нито Не, означава че отговорът на въпроса не е само “да” и не е само „не“, има допълнения към отговора, които могат да са верни или не;
- ▶ Липсата на отговор по посочените въпроси се отбелязва и отчита с нула.

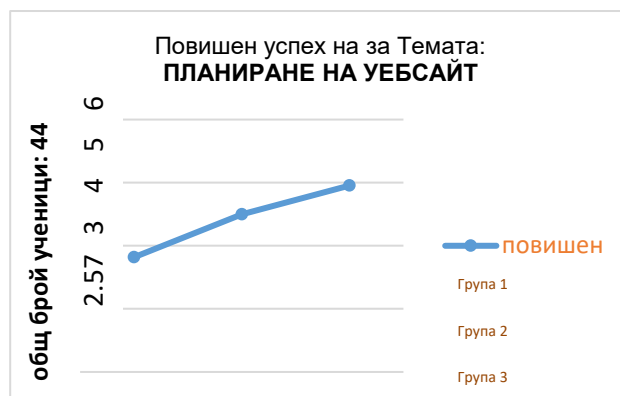
Статистическо изследване на фактора методика

За статистическото изследване на фактора методика е направен сравнителен анализ между трите групи за сравняване на теоретичните знания на обучаемите от проведените тестови изпитвания (първо ниво) и комбинацията от теоретични знания, практически умения и самостоятелна работа (задачи) на изходящото ниво (второ ниво), за установяване ефективността на предложената методика при трите контролни групи. Взема се за изследване извадката от ученици на трите контролни групи, за които се следи за повишаване на оценката от първо към второ ниво, за учебните **Теми**:

- ▶ **Планиране на уеб**
- ▶ **Проектиране и графичен дизайн в уеб**
- ▶ **Изграждане, тестване и публикуване на уеб**
- ▶ **Системи за управление на съдържание**

► Сигурност в уеб

►
►
►
Чрез графичните изображения по специалности и контролни групи се вижда много ясно процентното съотношение на учениците, постигнали повишаване на успеха си в рамките на темите от учебната програма.



Фиг. 25 *Повишаване знанията, уменията и компетенциите по тема Планиране на уеб сайт*

2.3. Статистически резултати от изходящите анкети на учениците

Анализирането на данните от тези анкети разкрива мнението на обучаваните за постигнатите от тях резултати в теоретичните и практически аспекти на изучаваната дисциплина Уебдизайн. Разработените анкети са в електронен вид и са много лесни и удобни за съхранение, обединение и анализиране на данните.

При изработването на анкетата е заложен стремеж за откриване мотивацията на обучаваните в овладяването на предвидения учебен материал и индивидуалните препоръки на учениците за оптимизиране на обучението. Разгледани са пет основни направления при изследване на личното мнение на учениците относно:

- ✚ Интерес към изучаваните теми в модул Уебдизайн;
- ✚ Приложение на изучаваните софтуерни технологии, уеб стандарти, уеб принципи, уеб сради и др.;
- ✚ Предпочитания при изграждането на интерактивен интерфейс, относно използваните технологии, среди, стандарти, принципи и др.
- ✚ Затруднения от теоретично или практическо естество в изучаваните теми от учебната програма;
- ✚ Самооценка за постигнатите теоретични и практически нива от знания по Уебдизайн.

Големият обем от учебна и практическа дейност на учениците при овладяване и използване на дигитални компетенции, както и липсата на предварителни знания в част от обучаваните по този модул е предпоставка, за включването в анкетата на въпрос, изискващ от учениците да изброят какво ги затруднява най-много в изучавания материал:

- ✚ Процесът на планиране на изграждане на интерактивен интерфейс;
- ✚ Изграждане и дизайн на уеб сайта;
- ✚ Работа с различни софтуерни технологии;
- ✚ Работа със системи за управление на уеб съдържание;

Защита на уебсъдържанието;

Отговорите на учениците са разнообразни, но работата с технологии и различни графични програми преобладава, както и начините за използване на всички уеб средства. Посочените затруднения се използват от преподавателя за оптимизиране на учебните дейности през следващите години.

2.4. Използване на софтуерните технологии и приложения за обслужване на учебния процес

Използването на софтуерни технологии, пробужда интереса на част от учениците да овладеят някои допълнителни възможности на изучаваният модул и може да се окаже удачно в работата им след завършване в училище.

Безспорна е необходимостта от овладяване на основните уеб технологии, защото ако се направи равностметка за анкетираните обекти от събеседването със учениците, изводите са че големите фирми/предприятия предпочитат добри дигитални компетенции за служителите си.

Изграждането на интерактивен интерфейс включва електронният обмен на данни и е много важен фактор в съвременните условия на живот. Уебдизайнът е актуален и търсен на пазара на труда. Това обуславя необходимостта да се включи в изследователската дейност на учениците и това проучване. Въпрос №1 и поясняващите възможности за отговори, заедно с резултатите са дадени в **таблица 10**.

Таблица 10. Данни за отговорите от въпрос № 1: Хареса ли Ви да изградите интерактивен интерфейс?

Хареса ли Ви да изградите интерактивен интерфейс?	Да	Не	Не знам	Няма отговор
<i>Група 3</i>	15	1	2	1
<i>Група 2</i>	9	2	1	0
<i>Група 1</i>	8	1	2	1



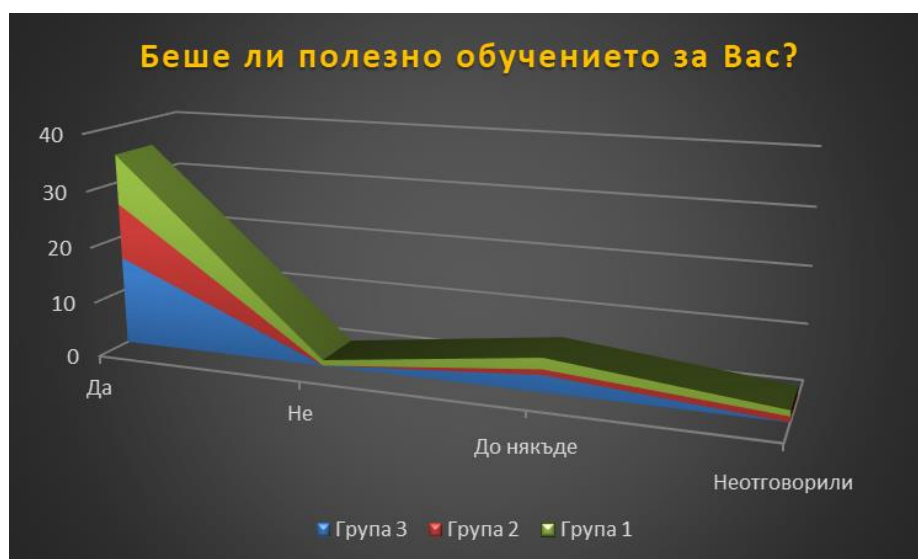
Фиг. 26 Графично представяне на отговорите на учениците по групи на въпрос №1 от изходящата анкета

Интерес за проучването са както използваните електронни документи, така и другите системи за управление на ресурси, за които учениците могат да дадат информация или са описани в анкетата.

Статистическа справка за иновационните софтуерни технологии, които учениците използват в изграждането на интерактивен интерфейс, може да се направи като се обработят отговорите на въпрос № 3 и 4 от изходящата анкета. В изводите, трябва да се отчита с какъв размер на извадката от работни позиции, избрани от ученици се правят заключения, именно 44.

От събеседване с учениците се оказва, че въпросът, който ги интересува най-много е въпрос № 6 (**Какво Ви затруднява най-много?**).

Но Педагогическото проучване касае най-вече Въпрос №5, виж Фиг. 22.



Фигура 27. Графично представяне на резултатите от Въпрос 5: Беше ли полезно обучението за Вас?

Изводи и оценки за резултатите от анкетата

При събеседване със учениците за проведеното проучване, ясно може да се разбере степента на ефективност от изпълнението. Прави впечатление обогатеният речник на изразяване, познават нова терминология и развит стил на изразяване по Уебдизайн. Целта на проучването е постигната.

Учениците успяха да навлязат в материята – **Изграждане на интерактивен интерфейс**, да разберат как и по какъв начин да изградят добър дизайн, да правят сами избор за изграждане на различни видове уебсайтове, като знаят всеки един от тях с какви технологии и ресурси се постига.

Обучаваните са почувствали широкия спектър от изисквания в различните форми на задаване на задачи към тях и са придобили самочувствие на уебдизайнери.

Около 80% от обучаваните могат да формулират ясно и мотивирано оценка на потребностите от обучение по дисциплината Уебдизайн, като се аргументират с конкретни факти.

Фазата „Оценка на резултатите от изследването“ е самостоятелна за всеки ученик и е особено значима за преподавателя, поради необходимостта от:

- ✓ Оценка спрямо очакваната промяна в поведението на участниците на работното им място;
- ✓ Оценка адекватността на проведеното изследване, идентификация на потребностите от знания и умения в областта на информационните технологии и Уебдизайн;
- ✓ Оценка на съдържанието, методите и начина на проучването от учениците;
- ✓ Оценка за качеството на събраната информация и нейното влияние за промени в начина на обучение по модул Уебдизайн;
- ✓ Оценка на възможностите за интегриране на обучението по Уебдизайн с конкретни приложения в бъдещо обучение и реализация.

Фазата се реализира на два етапа:

Етап 1. Оценка на впечатленията на учениците от проучването, след приключването му.

Обучаваните трябва да формулират изводи за полезността на учебния материал от Уебдизайн, за подобряване процеса на обучение и повишаване качеството на образование в тази област. Целта е да се коментират силните и слабите страни от обучението в конкретния практически аспект, степента на готовност да се прилагат на практика усвоените знания.

Етап 2. Оценка на въздействието от проведеното проучване.

Тя се прави от преподавателя, който е външен оценител и може да постигне максимална обективност. Така могат да се получат безпристрастни резултати, които са от полза за учебния процес по Уебдизайн. Оценката се извършва след задълбочено проучване и анализ на всички реализирани фази от изследователския процес – оценка на потребностите от изграждането на интерактивност в уеб, разработка на учебни материали, използване на подходящи технологии спрямо уеб решенията, които се взимат, избор на методи и подходи за представяне на учебното съдържание.

На базата на предложената методика, връзката между преподавателя и учениците се качва на ново, по-високо ниво. Доверието към преподавателя се повишава, защото включването на допълнителната информация при изпълнение на практическите занятия зависи от самия ученик, както и ползването на указания от ПРЕПОДАВАТЕЛЯ, КОЙТО Е МОТИВАТОР, ОРАТОР, МЕНТОР, ВДЪХНОВИТЕЛ.

Проучването показва, че практическата насоченост в обучението Уебдизайн, подчинено на ясна методическа стратегия, относно интердисциплинарно обвързване с изучаваните от учениците дисциплини и изследователска насоченост, които могат да дадат сигурен ориентир на бъдещите софтуерни инженери за изискванията от теоретични и практически умения, прави процеса на обучение по Уебдизайн цялостен, завършен и ефективен, тъй като реално повишава нивото на дигитални компетенции у учениците. Проучването показва също, че интересът към Уебдизайн е завишен в последните години.

ЗАКЛЮЧИТЕЛНИ ИЗВОДИ ЗА ГЛАВА ТРЕТА

От направените анализи на проведеното изследване за създаване на интеграционна насоченост при обучението на учениците, се достига до статистическия извод, че приложената методика за преподаване е довела до постигане на нарастващ резултат, съответно за контролните групи и по-конкретно Група 3: WEBDESIGN при СУ „Св. Софроний Врачански“ гр. Пловдив, 2021-2022 г.

Като се започне от статистическия анализ на входящите групи за обучение, използват се оценките и изводите от статистическия анализ на изходящите групи и се даде в нарастваща скала тежест за използваните софтуерни технологии, проеграми и средства в разглежданите методически групи, може категорично да се обобщи, че предложената методика способства за повишаване на качеството на обучение по Уебдизайн. Самата тематика се оказва интересна за учениците и те имаха възможност да се „разгърнат“ и да постигнат добри уебпроекти, т.е. да развият дигиталните си компетенции.

Разнообразието на използваните методи и програмни приложения в преподаването увеличава вероятността за задържане интереса на обучаваните.

Използването на електронни приложения от учениците в трета група, за изследване на начините за обвързване на обучението по Уебдизайн с различни отрасли на бизнеса и социалните сфери, създава нагласа за позитивно и осъзнато отношение към учебния процес и се превръща във важен източник за нови знания.

Под ръководството на преподавателя, обучаваните имат възможност да осмислят познанията си за Изграждането на интерактивния интерфейс „на живо“, да вникнат в същността на теоретичните и практически аспекти на Уебдизайна, да открият практическата полза и да започнат да планират, как да използват компютърните технологии в бъдеще.

Може да се твърди, че използването на различните електронни средства в предложената методика са ефективен инструмент, който следва да се използва от преподавателя в процеса на преподаване и обучаемите в процес на учене, с цел придобиване на трайни, правилно структурирани знания с ясна представа за тяхното практическо приложение в различните сфери на бъдеща реализация на учениците.

Основният извод след проведените статистически анализи и изследвания е, че предложената методика ефективно съчетава идеи, методи и подходи в начина на преподаване на модул Уебдизайн, които водят до по-трайно овладяване на учебния материал, дават висока практическа насоченост, непрекъснатост в процеса на усвояване на нови знания и умения от обучаваните, както и възможност те да осъзнаят процеса в дълбочина.

Въз основа на проведеното изследване се очертават следните приноси:

1. Научно-приложни:

- 1.1. В резултат на проведени методически проучвания и изследвания се предлага нова организация на обучението по Уебдизайн, която включва провеждане на

непрекъснат процес на „експеримент”, подчинен на определени основни дейности.

- 1.2. Предложена е методика за проучване възможностите при използване на знанията по Уебдизайн в различни учебни дисциплини от учебната програма, даващи възможност за откриване на нови МОЗ (методически обвързващи звена) и ИДО (интердисциплинарно обвързване) с други дисциплини
- 1.3. Създаден е модел за изследване на уеб процесите и дейностите по стандартизация и електронен обмен на данни.

2. Приложни:

- 2.1. Направено е емпирично проучване на голям брой практически дейности, чрез които преподаването по Уебдизайн може успешно да се интегрира.
- 2.2. Създадени са анкети за изследване процеса на обучение и за обвързване на обучението по Уебдизайн с производствени предприятия, фирми и свободни професии.
- 2.3. Предложен е богат набор от примери, позволяващ вграждане на математическото мислене в учебните модули по Уебдизайн и способстващ за изграждане на навици за математическа дейност чрез използване на компютри.

ПОЛЕЗНИ ИНТЕРНЕТ – АДРЕСИ

<http://htowired.lycos.com/webmonkey/> - Webmonkey

- <http://www.webdeveloper.com>
- <http://www.webreferece.com/>
- <http://wdvl.internet.com/WDVL>
- <http://developer.netscape.com/library/documentation/javascript.html> - документация за JavaScript
- <http://www.killersites.com/> - ресурси в помощ на уебдизайнерите
- <http://www.webstandards.org> – уеб стандарти и полезни линкове
- <http://www.freewebtemplates.com/>
- <http://patmclendon.com/html.html> - уроци по HTML и безплатни графики
- <http://www.widearea.co.uk/designer/> - ресурси за създаване на уебграфика
- <http://www.webreference.com>
- <http://www.webdeveloper.ocm>
- <http://www.cgi-resources.com/> - ресурси за създаване на CGI
- <http://javascriptsource.com/> - ресурси за създаване на JavaScript

РЕСУРСИ НА УЕБДИЗАЙНЕРА НА БЪЛГАРСКИ ЕЗИК

- <http://www.abcbg.com/webdesign>
- <http://www.websamba.com/webinfo/developers/index.htm>
- <http://eton.dir.bg/moshenik/kolegi.html> - полезна информация за уеб разработчици от moshenik.bg, на български език
- <http://www.4u.search>
- <http://topcities.com/art/miro/>

СТАТИСТИКИ, ПРОМОЦИИ

- <http://www.linkexchange.com/> - система за размяна на банери
- <http://www.searchenginewatch.com/> - всичко за търсещите машини
- <http://www.selfpromotion.com/> - регистрация в основните световни търсачки
- www.extreme-dm.com/tracking - ExtremeTracker - статистика за посещенията на сайта
- <http://counter.search.bg> – WebCounter – статистика за посещения
- <http://www.fastcounter.com> – FastCounter – статистика за посещения

РЕГИСТРАЦИЯ НА ДОМЕЙН

- <http://www.digsys.bg>
- <http://www.dom.bg>
- <http://www.icn.com>
- <http://domains.com>
- <http://www.jumpdomain.com>

ОРГАНИЗАЦИИ

- <http://www.statmarket.com/> - статистики за мрежата и търсенето от страна на потребителите
- <http://www.internet.com/> - Internet Network Information Center (DNS, IP, names, hosts)
- <http://www.w3.org> – стандартите за разработка на уеб
- <http://www.webstandards.org> – организация, грижеща се за стандартите в уеб

ЕЛЕКТРОННИ ИЗДАНИЯ ЗА МРЕЖАТА

- <http://abcbg.com/ezine> - New Media E-zine – издание с новости за разработчици и дизайнери, на български език
- <http://www.todos.searsh.bg/> Todo s Web – издание на български език за уеб разработчици и дизайнери
- <http://wt.noxis.net/> - Web Today – издание на български език за уеб програмисти
- <http://www.wired.com> – електронна версия на най-популярното издание за уеб дизайн Wired

ДРУГИ интернет връзки

- <http://www.cyberatlas.com> – любопитни неща за мрежата
- <http://www.microsoft.com/ie> - официален сайт на Internet Explorer
- <http://www.netscape.com> – официален сайт на Netscape
- <http://www.macromedia.com> – официален сайт на Macromedia

РЕЧНИК - ОПРЕДЕЛЕНИЯ

- XML (Extensible Markup Language – Разширяем език за маркиране), основата, на която са изградени уеб услугите, предоставя език за дефиниране на данни и начина на обработката им. XML представлява семейство от свързани спецификации, specifications публикувани и поддържани от World Wide Web Consortiou (W3C) и др.
- WSDL (Web Services Description Language – Език за описание на уеб услуги), технология, базирана на XML, която дефинира интерфейси за уеб услуги, типове данни и съобщения, начини за взаимодействие и протоколни съответствия.

- SOAP (Simple Object Access protocol – Прост протокол за достъп до обекти), колекция от технологии, базирани на XML, които дефинират обвивка за комуникацията между веб услугите, която може да се съгласува с HTTP и други транспортни протоколи, и осигурява формат за сериализация за пренасянето на XML документи през мрежа и конвенция за представянето на PRC взаимодействия.
- UDDI (Universal Description, Discovery, and Integration - Универсално описание, откриване на интегриране), механизъм за регистриране и откриване на веб услуги, който се използва за съхраняването и категоризирането на бизнес информация и за извличане на указатели към интерфейси за веб услуги.

ВАЖНО: Понякога веб услугите се описват като решенията – „Сребърен курсум“ за настоящите изчислителни проблеми, като изпълняват ролята, която преди това са изиграли първоначалното виртуално пространство, релационните бази данни, езиците от четвъртото поколение и изкуствения интелект. За нещастие веб услугите не могат да решат самостоятелно почти нищо. Веб услугите са нов слой – друг начин за вършене на нещата – но не представляват основна промяна, която заменя необходимостта от съществуващата изчислителна инфраструктура. Този нов технологичен слой изпълнява нова функция – нов начин на работа, но най-важното - осигурява механизъм за интегриране, който е дефиниран на по-високо ниво на абстракция.

Веб услугите са важни, защото са в състояние да свързват технологични домейни, а не защото заменят някаква съществуваща технология. Бихте могли да кажете, че по-новите езици за програмиране, като например Visual Basic, C# C, C++ и Java, заменят старите езици, като COBOL, ALGOL, FORTRAN, въпреки че има все още много програми написани на тези езици и за тях има създадени аналози в веб услугите. Веб услугите подобно на веб сървърите, допълват, а не са в конфликт със съществуващите приложения, програми и бази данни. За разработката на приложения все още са необходими Java, VB, C#. Новото е начинът за трансформиране на данни във и извън програми и приложения, като се използват стандартни XML формати за данни и протоколи, за да се постигне ново ниво на комуникация и интегриране.

Вероятно разработчиците ще трябва да вземат предвид веб услугите, когато проектират и разработват нови програми и бази данни, но тези програми и бази данни все още ще са необходими зад обвиващия слой на веб услугите. Веб услугите сами по себе си не са изпълними; те разчитат на изпълними програми, написани с използването на езици за програмиране и скриптове. Веб услугите дефинират мощен слой на абстракция, който може да се използва за осъществяването на взаимодействие между програми, като се използва съществуващата инфраструктура на виртуалното пространство, но не могат да постигнат нищо без поддържащата инфраструктура. Веб услугите изискват няколко свързани с XML технологии за пренасяне и трансформиране на данни във и извън програми и бази данни.

ПРИЛОЖЕНИЕ

ПРИЛОЖЕНИЕ 1: УЧЕБНА ПРОГРАМА 2

УТВЪРДИЛ!

Директор:

.....

(Светлозар Чавдаров)

**УЧЕБНА ПРОГРАМА
(ЗАНИМАНИЯ ПО ИНТЕРЕСИ)
УЕБДИЗАЙН**

2 часа седмично = 72 часа общо

Старт: от 05 ноември, всеки петък от 14.00 часа

КРАТКО ПРЕДСТАВЯНЕ НА УЧЕБНАТА ПРОГРАМА

Обучението в модул „Уебдизайн“ е насочена към овладяване на базисни знания и умения, свързани с използването на съвременни софтуерни технологии за изграждане на интерактивен интерфейс и дигитални компетентности на учениците. Програмата е предназначена за обучение на ученици, които имат интереси свързани със създаването на динамични уебсайтове, както и за профилирана подготовка на ученици в гимназиален етап.

Учебното съдържание е представено в следните основни теми:

- Планиране на уебсайт
- Структура, динамика и графичен дизайн на уебсайт
- Изграждане, тестване и публикуване на уебсайт
- Софтуер за създаване на уебсайт. Системи за управление на уеб съдържание
- Уеб сигурност

В този модул, акцентът е поставен върху планирането, проектирането, графичния дизайн и изграждането на уеб сайтове чрез използване на съвременни уеб технологии и стандарти. Специално внимание е отделено и на работата със системи за управление на съдържание, като вариант за удобно и лесно създаване и поддръжка уеб сайтове. Естествено, този вариант има и недостатъци, но учениците сами ще могат да правят своя избор от технологии, след като са запознати с тях.

Първи учебен срок – 12 седмици по 2 часа седмично = 24 часа

(1)	(2)	(3)
Теми	Компетентности като очаквани резултати от обучението	Нови понятия

(1)	(2)	(3)
<p>Тема 1. ПЛАНИРАНЕ НА УЕБСАЙТ</p> <p>1. Основните етапи в планирането на уеб сайт</p>	<ul style="list-style-type: none"> • Описва основните етапи и дейности в планирането на уеб сайт. • Определя темата, целите и целевата група на сайта. • Определя изискванията, на които да отговоря сайтът. • Разчита задание за изграждане на уеб сайт. 	<p>Карта на сайта;</p>
<p>2. Избор на име и регистриране на уеб сайт</p>	<ul style="list-style-type: none"> • Разбира системата за имена на домейни в интернет. • Избира подходящо име на сайт и да го регистрира. 	<p>Домейни-видове; Стандартизиран адрес на ресурс (Uniform resource locator, URL); Система за имена на домейни – Domain Name System (DNS); IP адрес; Хостинг;</p>
<p>3. Структура на уебсайта</p>	<ul style="list-style-type: none"> • Да визуализира идеите си чрез подходящи техники и технологии 	<p>Структура на сайта;</p>
<p>Тема 2. ПРОЕКТИРА НЕ И ГРАФИЧЕН ДИЗАЙН В УЕБСАЙТ</p> <p>4. Основни етапи при проектирането на уебсайт.</p>	<ul style="list-style-type: none"> • Умее да избира подходящо име на сайт и да го регистрира. • Познава услугата уеб хостинг и нейните характеристики. 	<p>Dom.bg Icn.bg www.freehosting.com/ www.googlehost.com/ www.awardspac.e.co/ www.000webhost.com https://byet.host www.freehostingoads.net</p>

(1)	(2)	(3)
	<ul style="list-style-type: none"> • Умее да описва основните етапи и дейности в планирането на уеб сайт. • Умее да определя темата, целите и целевата група на сайта. 	
<p>5. Създаване на графичен модел на уебстраница</p>	<ul style="list-style-type: none"> • Описва основните етапи в проектирането на уеб сайт. • Създава диаграма, показваща зависимостите между отделните страници. • Използва подходящ софтуер за изграждане на структурата на уеб сайт. • Проектира подходяща навигационна структура на сайт и структура на системата от файлове и директории с цел оптимизиране на уеб сайта за търсещи машини. • Аргументира избраната структура на уеб сайт. • Осъзнава необходимостта от създаването на качествено съдържание за уеб сайт. 	<p>Тест за самопроверка;</p>

(1)	(2)	(3)
<p>6. Създаване на графичен модел на уебстраница-упражнение</p>	<ul style="list-style-type: none"> • Описва основните етапи в проектирането на уеб страница. • Прилага основните стандарти, свързани със структурата, графичния дизайн и графичния интерфейс на уеб страница. • Изброява специфични изисквания при проектирането на начална страница на уеб сайт. • Посочва специфични изисквания при изграждането на сайтове за хора с увреждания. 	
<p>7. Проектиране на уеб страница (упражнение)</p>	<ul style="list-style-type: none"> • Умее да създава схематичен дизайн на уеб страница чрез използване на подходящ софтуер. • Умее да изгражда структура на уеб сайт. 	
<p>8. Създаване на графичен модел на уеб страница</p>	<ul style="list-style-type: none"> • Обяснява ролята и значението на цветовете в уеб дизайна. • Използва софтуер за създаване и избор на цветови схеми. • Обяснява значението на шрифтовете в уеб дизайна. • Създава графичен модел на уеб страница чрез използване на подходящ софтуер. 	

(1)	(2)	(3)
9. Софтуер за създаване и избор на цветови схеми (упражнение)	<ul style="list-style-type: none"> • Умее да работи с конкретно приложение за създаване и избор на цветови схеми. • Умее да избира подходящ цвят от списък с цветове. • Умее да намира хармонични цветови комбинации. 	Цветова схема;
10. Създаване на графичен модел на веб страница (упражнение)	<ul style="list-style-type: none"> • Умее да създава графичен модел на веб страница с приложението Inkscapе. • Проектира графичен модел на начална страница на конкретен веб сайт. 	Графичен модел (mockup); Схематичен дизайн (wireframe); Графичен потребителски интерфейс; Интерактивен интерфейс; Достъпност;
11. Проектиране и графичен дизайн на веб сайт (обобщение)	<ul style="list-style-type: none"> • Познава основните етапи при проектиране на веб сайт. • Умее да създава графичен модел на веб страница. 	Тест за самопроверка;
<p>Тема 3. ИЗГРАЖДАНЕ, ТЕСТВАНЕ И ПУБЛИКУВАНЕ НА УЕБ САЙТ</p> <p>12. Уебдизайн стандарти в уеб и основни технологии в уеб</p>	<ul style="list-style-type: none"> • Обяснява уеб дизайн стандартите и тяхното предназначение. • Посочва основните уеб технологии в интернет. 	Hyperlink; Hypertext; Webserver; HTTP;
13. Специализиран и софтуерни средства за създаване на уеб сайтове	<ul style="list-style-type: none"> • Използва редактори и средства за разработка на уеб • Използва готови шаблонни форми. • Обяснява и спазва 	Sublime, Notepad++, WinSCP, FileZilla, MS ExpressionWeb4, Inkscapе, CSS, HTML

(1)	(2)	(3)
	различните типове права и лицензи по отношение на използваните шаблони.	
14. Основи на HTML	<ul style="list-style-type: none"> • Използва актуална версия на езика HTML за изграждане на структурата на уеб страница. • Вгражда мултимедийни блокове в уебстраница-анимация, видео, аудио. • Аргументира необходимостта от оптимизация на изображенията и мултимедия за уеб. 	(Hyper Text Markup Language)HTML;
15. Основи на HTML. Тагове за текст, хипервръзка, изображение (упражнение)	<ul style="list-style-type: none"> • Умее да форматира текст в HTML. • Умее да създава хипервръзка в HTML. • Умее да използва изображения в HTML. 	хипервръзка
16. Основи на HTML. Тагове за таблица, списъци, аудио, видео (упражнение)	<ul style="list-style-type: none"> • Умее да използва тагове за таблица в HTML. • Умее да използва тагове за списъци в HTML. • Умее да използва тагове за аудио и видео в HTML. 	
17. Основи на HTML. Създаване на форми (упражнение)	<ul style="list-style-type: none"> • Умее да създава форми в HTML. 	HTML

(1)	(2)	(3)
<p>18. Основи на HTML (упражнение)</p>	<ul style="list-style-type: none"> • Умее да вгражда мултимедийни блокове в уеб страница – анимация, видео и аудио. • Умее да аргументира необходимостта от оптимизирането на изображения и мултимедия за уеб. 	<p>Мултимедия в web</p>
<p>19. Визуално оформяне на уеб сайт с използване на CSS.</p>	<ul style="list-style-type: none"> • Използва актуална версия на езика Cascading Style Sheets (CSS) за визуално оформяне на уеб страница. • Отделя информацията в уеб сайт от начина за нейното визуално представяне. 	
<p>20. Визуално оформяне на уеб сайт с използване на CSS (упражнение)</p>	<ul style="list-style-type: none"> • Умее да работи с езика Cascading Style Sheets (CSS). • Познава видовете селектори в Cascading Style Sheets (CSS). • Познава основните свойства, които се използват при работа с езика Cascading Style Sheets (CSS). 	
<p>21. Визуално оформяне на уеб сайт с използване на CSS (упражнение)</p>	<ul style="list-style-type: none"> • Умее да форматира текст с езика Cascading Style Sheets (CSS). • Умее да форматира фон с езика Cascading Style Sheets (CSS). • Умее да използва рамки с езика Cascading Style Sheets (CSS). 	

(1)	(2)	(3)
22. Визуално оформяне на уеб сайт с използване на CSS	<ul style="list-style-type: none"> • Добавя и използва шрифтове чрез средствата на CSS. • Използва уеб услуги за шрифтове. 	Font
<p>23. Работа по проект (изработване на уеб сайт)</p> <p>24. Работа по проект (изработване на уеб сайт)</p> <p>25. Работа по проект (изработване на уеб сайт)</p> <p>26. Работа по проект (изработване на уеб сайт)</p> <p>27. Работа по проект (изработване на уеб сайт)</p> <p>28. Работа по проект (изработване на уеб сайт)</p>	<ul style="list-style-type: none"> • Сформира екип за реализиране на конкретен проект. • Избира подходящи средства за реализация на конкретен проект. • Проектира конкретен уеб сайт. • Създава информационна архитектура на уеб сайта. • Определя елементите и изгражда навигационна структура на уеб сайта. • Събира и обработва информацията, която ще се включва в уеб сайта – текст, графични изображения, мултимедийни продукти. • Изготвя схематичен дизайн (wireframe) на отделните страници. • Създава графичен модел на уеб сайта с приложението Inkscapе. • Изработва уеб сайта с помощта на HTML редактор. 	<p>Редактор на уебсайт;</p>

Втори учебен срок – 6 седмици по 2 часа седмично = 12 часа

(3)	(4)	(5)
<p>29. Тестване и валидиране на кода на веб страница</p>	<ul style="list-style-type: none"> • Обяснява необходимостта от тестване на веб сайт преди и след неговото публикуване. • Използва инструменти за валидиране на веб страници. • Използва емулятори за тестване на веб сайт за работа с различни браузъри. • Използва инструменти за тестване на достъпността на веб сайт. 	<p>Валидиране; Браузър;</p>
<p>30. Тестване на достъпност на веб сайт (упражнение)</p>	<ul style="list-style-type: none"> • Умее да тества скорост и достъпност на веб страница. • Познава различните инструменти за анализ на браузъра Google Chrome. 	
<p>31. Публикуване на веб сайт. Авторски права</p>	<ul style="list-style-type: none"> • Публикува веб сайт. • Обяснява и спазва правните и етичните норми при публикуване на информация в интернет. Авторски права. • Умее да избира хостинг и домейн. • Познава сайтове доставящи безплатни хостинг услуги. 	<p>Авторски права</p>

(3)	(4)	(5)
<p>32. Оценка и оптимизация на уеб сайт. Анализ на посещаемостта на уеб сайт (упражнение)</p>	<ul style="list-style-type: none"> • Обяснява необходимостта от оценка и последваща оптимизация на уеб сайт. • Използва инструменти за анализ на посещаемостта на уеб сайт. • Посочва и използва инструменти за популяризиране на уеб сайт. • Изброява методики и техните специфики за оценяване на уеб сайт. • Оценява уеб сайт на базата на зададени спецификации чрез използване на специализирани инструменти. • Познава инструменти за анализ на посещаемостта на уеб сайт. • Познава различни методики за оценяване на уеб сайт 	<p>Оптимизация на сайт; Search Engine Optimization (оптимизация за търсещи машини); Рангово поддръжане (ранжиране); Really Simple Syndication (RSS);</p>
<p>33. Представяне и защита на проект пред публика</p>	<ul style="list-style-type: none"> • Описва принципите на управлението на проекти с цел проектиране, разработване и поддръжка на комплексни уеб сайтове. • Представя пред публика разработен уеб сайт. • Описва инструменти и стъпки за оценка на уеб сайт. • Оценява уеб сайт. 	

(3)	(4)	(5)
<p>Тема 4. СИСТЕМИ ЗА УПРАВЛЕНИЕ НА УЕБ СЪДЪРЖАНИЕ</p> <p>34. Същност, предназначение и архитектура на система за управление на уеб съдържание</p>	<ul style="list-style-type: none"> • Обяснява предназначението, предимствата и недостатъците на различни системи за управление на уеб съдържание; • Описва трислойната архитектура на система за управление на уеб съдържание; • Инсталира система за управление на уеб съдържание; 	<p>Система за управление на уеб съдържанието (Content Management System, CMS);</p>
<p>Тема 5. УЕБСИГУРНОСТ В УЕБ</p> <p>35. Сигурност в уеб.</p> <p>36. Какво научихте в уеб дизайн (обобщение)</p>	<ul style="list-style-type: none"> • Описва различни видове атаки, насочени срещу нормалното функциониране на сайта и начини за тяхното идентифициране. • Обяснява рисковете, които могат да причинят уеб сайтове, които използват графиките на уеб страниците като уеб маяци. • Описва методите за сигурност на информацията в уеб приложения и компрометиране поверителността на данните на потребителите чрез "бисквитки". • Описва начините за увеличаване на надеждността на сайта чрез сертификати за сигурност. • Умее да прилага използването на публичния ключ за сигурност за повишаване на 	<p>Бисквитки; Уеб маяци; Публичен ключ за сигурност; (САРТСНА); Сертификат за сигурност; Допълнителна функционалност (Plug-in) за защита;</p> <p>Тест за самопроверка</p>

(3)	(4)	(5)
	<p>сигурността.</p> <ul style="list-style-type: none"> • Познава различни средства за повишаване сигурността на уеб сайт. • Познава общите проблеми със сигурността в уеб. • Умее да повишава сигурността на сайт, създаден чрез система за управление на уеб съдържание. • Умее да създава резервно копие на сайт. • Умее да планира уеб сайт. • Умее да проектира уеб сайт. • Умее да създава графичен дизайн на уеб сайт. • Умее да изгражда, тества и публикува уеб сайт. • Познава системи за управление на уеб съдържание. • Познава проблемите със сигурността в уеб. 	

ИЗГОТВИЛ: Детелина Милкотева

ПРИЛОЖЕНИЕ 2: УЧЕБНА ПРОГРАМА 3

УЧЕБНА ПРОГРАМА ПО УЕБДИЗАЙН ЗА УЧЕНИЦИ В И ПРОГИМНАЗИАЛЕН ЕТАП – ФАКУЛТАТИВНИ ЧАСОВЕ

КРАТКО ПРЕДСТАВЯНЕ НА УЧЕБНАТА ПРОГРАМА по Проект Образование за утрешния ден

УТВЪРДИЛ:

ДИРЕКТОР: Величка Йорданова

Обучението по *УЕБДИЗАЙН*, е насочено към овладяване на знания, умения и отношения, свързани с изграждане на дигиталната грамотност на учениците. Акцентът в обучението пада върху разширяване и надграждане на придобитите дигитални компетентности така, че при завършване на задължителното обучение по компютърно моделиране и информационни технологии учениците да бъдат подготвени за по-нататъшна успешна реализация. Заложените в програмата теми покриват области, свързани със създаването на интерактивен интерфейс под формата на динамичен уеб сайт. Интерактивната комуникация предполага създаване на динамична среда, променящо се пространство, специфична навигация, в зависимост от предпочитанията на потребителя, динамичен дизайн, използване на специален компютърен код на най-високо равнище, мобилни технологии и непрекъснати иновации.

ОБОБЩЕНИ РАЗДЕЛИ В УЧЕБНАТА ПРОГРАМА:

- РАЗДЕЛ 1. ТЕОРЕТИЧНИ ОСНОВИ НА УЕБДИЗАЙНА. ПЛАНИРАНЕ И ТЕХНОЛОГИИ. HTML
- РАЗДЕЛ 2. ПРИНЦИПИ И ИЗГРАЖДАНЕ НА УЕБСАЙТ С MS EXPRESSION WEB4
- РАЗДЕЛ 3. ТЕСТВАНЕ И ПУБЛИКУВАНЕ
- РАЗДЕЛ 4. ОЦЕНКА, АКТУАЛИЗАЦИЯ И ОПТИМИЗАЦИЯ. ПРЕДСТАВЯНЕ

ОЧАКВАНИ РЕЗУЛТАТИ В КРАЯ НА ОБУЧЕНИЕТО

Очакваните резултати във всяка от тези области се постигат в рамките на цялостното обучение в задължителната общообразователна подготовка в училище, в това число и интегрирано в обучението по другите общообразователни учебни предмети. Учениците ще се запознаят с това, какво трябва да знаят преди започване на изграждането на Интерактивен интерфейс, презентация, представяне на информация чрез уеб, минавайки през избора за средства и съвременни технологии, до същинската част на реализацията му. Ще умеят да правят оценка на уебсайтове, да тестват, актуализират, оптимизират.

Акцентът в обучението е поставен върху планирането, проектирането, графичния дизайн и изграждането на интерактивна информационна презентация под формата на уебсайт и избора на технологии за реализацията му.

УЧЕБНО СЪДЪРЖАНИЕ

ОБЛАСТИ НА КОМПЕТЕНТНОСТ	ЗНАНИЯ, УМЕНИЯ И ОТНОШЕНИЯ
РАЗДЕЛ 1. ТЕОРЕТИЧНИ ОСНОВИ НА УЕБДИЗАЙНА. ПЛАНИРАНЕ И ТЕХНОЛОГИИ. HTML	<ul style="list-style-type: none"> - Познават понятията потребителски интерфейс, интерактивен интерфейс. - Знаят основните на уебдизайна. - Посочва примери за дейности, които могат да се приложат в разработката на уеб сайт. - Познават различните начини, техники и средства, с които може да реализират проекта си. - Планират, използват и реализират съвременните технологии и средства за реализацията на проект. - Планиране и изброяване на технологиите за реализация. - Работят с HTML като основа за надграждане. - Посочват примери за дейности, които могат да се опишат с шаблонен метод. Пише HTML код. - Работят с HTML код, създава уеб страници. - Вграждат различни фрагменти – банери, лого, бутони, параграфи, движещи се текстове, фон, таблици, шаблон от Интернет за фон, шрифт, галерии, видео, изображения, коментари, търсачки, интерактивни карти и др. - Умеят по зададена тема да изградят интерактивен уеб сайт.
РАЗДЕЛ 2. ПРИНЦИПИ И ИЗГРАЖДАНЕ НА УЕБСАЙТ	<ul style="list-style-type: none"> - Знаят основните принципи на уебдизайна. - Определят понятията софтуерни технологии и дизайн за реализиране на интерактивен интерфейс. - Описват основните софтуерни технологии: среда с визуално програмиране, създаване на интерактивен уеб сайт с шаблонен метод, CMS - системи, по-сложни техники и методи за реализация, но основно HTML. - Познават начини за изграждане на уебсайт във визуална среда. Работи в MS Expression web 4. - Създават препратки към други уебсайтове и други полезни връзки. - Ползват и създава елементи на готово от други източници и приложения. - Избират и използва изучавани софтуерни приложения за анализ и обработка на данни. - Посочват съвременни технологични средства за дигитално сътрудничество при планирането и оформяне изгледа на интерактивния интерфейс под формата на уеб сайт. - Умеят да предоставя различни варианти на изглед на уеб проекта за потребителите. - Посочват начини за надеждна дигитална идентификация при избора на дизайн за проекта на определена тематика.
РАЗДЕЛ 3. ТЕСТВАНЕ И ПУБЛИКУВАНЕ	<ul style="list-style-type: none"> - Знаят необходимостта от валидиране на уебсайта. - Тестват локално направения уебсайт, променя дизайна. - Познават специализиран софтуер за проверка на допуснатите грешки. - Проверят скоростта на визуализация на уебсайта. - Знаят какво е отзывчив дизайн и се стремят към постигането му. - Умеят да избират хост и domein при публикуване на сайта. - Публикуват.

Тема	Компетентности като очаквани резултати от обучението	Нови понятия
РАЗДЕЛ 1. ОСНОВИ НА УЕБДИЗАЙНА, ПЛАНИРАНЕ И ТЕХНОЛОГИИ. HTML		
ТЕМА 1.1. Въведение. Интерфейс. Интерактивен интерфейс. Интерактивна презентация – уеб сайт;	<ul style="list-style-type: none"> Разбира същността на понятието интерфейс, интерактивност; Разбира механизмите за разпространение на информация и ефективните начин за търсене в онлайн среда, чрез използване на интерактивни уебсайтове; Оценява информация, получена по електронен път, по отношение на достоверност и надеждност. 	<ul style="list-style-type: none"> интерфейс интерактивност
ТЕМА 1.2 Основни етапи при планирането на уеб сайт;	<ul style="list-style-type: none"> Знае основните етапи при планирането на уебсайт. Изброява различни технологии и начини, които допринасят за добрия интерфейс. 	<ul style="list-style-type: none"> HTML, XHTML... CSS PHP JAVA JAVASCRIPT MVC.NET - framework CMS – системи Плъгини Java аплети Dreamweaver MS Expression web 4 Weebly.com....
ТЕМА 1.3. Избор на домейн и хост за регистриране на уеб сайта - препоръки;	<ul style="list-style-type: none"> Определя понятието домейн и хост. Описва основните видове домейни. Избира подходящия домейн за конкретния уеб сайт. Проучва къде е най-добре да се хоства 	<ul style="list-style-type: none"> domain host
ТЕМА 1.4 Обмисляне съдържанието и разположението му в уеб сайта;	<ul style="list-style-type: none"> Знае колко важно е съдържанието в уеб сайта да не бъде с излишна информация. Да се акцентира върху същността ѝ. Поддържа съдържанието във всичките му страници по един и същи начин – шрифт за заглавие и текст с еднакъв формат на всяка страница и пр. 	
ТЕМА 1.5. Основни етапи при проектирането на уебсайта;	<ul style="list-style-type: none"> Прави план как ще изглежда сайта, карта на сайта. Знае етапите на проектирането му. 	
ТЕМА 1.6. Проектиране на (Home page, index.htm, index.html, index.css) уеб страница, шаблон;	<ul style="list-style-type: none"> Реализира уеб страница. Разбира, че може да работи с различен формат на уебстраницата, в зависимост от технологиите, които ще ползва. 	<ul style="list-style-type: none"> Index.html Index.htm Index.css Index.php
ТЕМА 1.7. Създаване на графичен модел на уебсайт;	<ul style="list-style-type: none"> Проектира графичния модел на уебсайта. Ползва различни приложения и техники. Позовава се на опита на добрите уеб дизайнери. Познава и работи с графични редактори като Adobe Illustrator, Adobe Photo Shop, CorelDraw, Photo Paint и др. Прави разлика между растрена и векторна графика. Избира формат на изображенията за уеб сайта. 	<ul style="list-style-type: none"> Adobe Illustrator Adobe Photo Shop Photo Paint CorelDraw SQL XML Perl Бъдещи топ езици – Оре, Fantom, Zimbu, X10, haXe, Chapel.....

ТЕМА 1.8. Уеб дизайн стандарти и основни технологии. Гаранции. Основни стандартизиращи организации;	<ul style="list-style-type: none"> Знае стандартите при разработка на уеб сайт. Какво гарантират стандартите. Кои са стандартизиращите организации. 	<ul style="list-style-type: none"> Функционалност Системно-операционна съвместимост Браузър съвместимост Независимост от резолюцията Независимост от потребителския интерактивен интерфейс
ТЕМА 1.9. Специализирани софтуерни средства за създаване на уебсайтове;	<ul style="list-style-type: none"> Познава набор от софтуерни приложения за реализирането на уебсайт. Прави избор и се аргументира. 	
ТЕМА 1.10. Основи на HTML. Основни тагове и тагове за форматиране;	<ul style="list-style-type: none"> Умее да пише HTML код. Знае основните тагове за форматиране на уебстраница. 	
ТЕМА 1.11. Тагове за създаване и редактиране на таблици, хипервръзки, връзки към изображения и списъци и др. уебсайтове;	<ul style="list-style-type: none"> Знае тагове за редактиране на таблици. Създава препратки към други страници в уебсайта, към изображения, други сайтове. 	
ТЕМА 1.12. Фон, шрифт, бутони, банери, лого и др.;	<ul style="list-style-type: none"> Създава фон на уеб страницата. Форматира шрифт. Вгражда шрифтове от Интернет. Създава бутони, вгражда бутони. Създава и вгражда банери. Рисува лого и „монтира“ в уеб страницата. 	<ul style="list-style-type: none"> Font Background banner button
РАЗДЕЛ 2.		
ПРИНЦИПИ И ИЗГРАЖДАНЕ НА УЕБСАЙТ		
ТЕМА 2.1. Основни принципи на уебдизайна;	<ul style="list-style-type: none"> Знае основните принципи при разработването на уеб сайт. Съобразява се с тях и изгражда успешен уебсайт. 	<ul style="list-style-type: none"> принципи
ТЕМА 2.2. Визуално оформяне на уеб сайт с използването на програмата MS Expression web 4;	<ul style="list-style-type: none"> Практически работи с приложението MS Expression web 4. Познава интерфейса на програмата и създава първата си уеб страница. Разглежда всички режими на работа в приложението. 	<ul style="list-style-type: none"> Режим Design Source code Локално тестване на уебсайта

	<ul style="list-style-type: none"> Преглежда локално уеб страницата си. 	
ТЕМА 2.3. Изграждане на index.html страницата и дизайн на сайта на тема Портфолио.?!;	<ul style="list-style-type: none"> Създаване на уеб страница – Портфолио на ученик... или друга тематика на уебсайта. 	<ul style="list-style-type: none"> Index.htm портфолио
ТЕМА 2.4. Форматиране на фон, текст, таблици, изображения и др.;	<ul style="list-style-type: none"> Вмъкване на фон от програмата, от интернет или собствен. Вмъкване на текст – форматиране. Вграждане на таблици, невидими таблици. Вмъкване на изображения и др. 	<ul style="list-style-type: none"> форматиране
ТЕМА 2.5. Изработка на банер, избор на бутони, хипервръзки и др.;	<ul style="list-style-type: none"> Работа с графичен редактор за изработка на банер, лого. Избор на бутони, вграждане. Създава хипервръзки. 	<ul style="list-style-type: none"> хипервръзки
ТЕМА 2.6. Мултимедия в уеб;	<ul style="list-style-type: none"> Вмъква музика, видео, галерии... Вмъкване на анимирани изображения за подсилване и акцентирание на елементи в уеб – познава сайтове с подобни изображения. Познава програмата Camtasia Studio 8. 	<ul style="list-style-type: none"> Анимирани изображения
ТЕМА 2.7. Работа в режим Design;	<ul style="list-style-type: none"> Работи в режимите предоставени от програмата, вграждане на други кодове. 	<ul style="list-style-type: none"> Вграждане на php Gallery в уебстраница
ТЕМА 2.8. Вмъкване на кодове за раздвижени галерии, бутони и др.;	<ul style="list-style-type: none"> Упражнява се и вгражда кодове на галерии, бутони с падащи менюта и др. 	<ul style="list-style-type: none"> Каскадни стилове Падащо меню
РАЗДЕЛ 3.		
ТЕСТВАНЕ И ПУБЛИКУВАНЕ. ПРЕДСТАВЯНЕ		

ТЕМА 3.1. Предимства на добре валидираната уебстраница. Необходимост от валидиране;	<ul style="list-style-type: none"> Знае предимствата на добре валидирания уебсайт. Осъзнава нуждата от валидиране. 	<ul style="list-style-type: none"> Валидиране
ТЕМА 3.2. Софтуер, който проверява допуснатите ни грешки преди да публикуваме уебсайта, други възможности..;	<ul style="list-style-type: none"> Познава и използва софтуер, проверяващ грешките преди публикуване на уебсайта в Интернет пространството. Ползва уеб сайт, предназначен за тази цел. 	<ul style="list-style-type: none"> http://validator.w3.org/
ТЕМА 3.3. Тестване на визуализацията през различни браузъри, отзивчив дизайн;	<ul style="list-style-type: none"> Познава и ползва софтуер за тестване визуализацията на уебсайта през различни браузъри преди публикуване. Знае какво е отзивчив дизайн и се стреми към постигането му. 	<ul style="list-style-type: none"> Cross-Browser Testing емулятори
ТЕМА 3.4. Тестване скоростта на визуализация;	<ul style="list-style-type: none"> Тества скоростта на визуализация на уеб сайта си. 	<ul style="list-style-type: none"> On- line инструмент за тестване скоростта на визуализация на сайта Web tools приложение на браузър
ТЕМА 3.5. Избор на хост и домейн – препоръки. Публикуване на уебсайта;	<ul style="list-style-type: none"> Прави избор на домейн и хост. Публикува уеб сайт. 	
РАЗДЕЛ 4. ОЦЕНКА, АКТУАЛИЗАЦИЯ И ОПТИМИЗАЦИЯ		
ТЕМА 4.1. Оценка на уебсайта. Актуализация;	<ul style="list-style-type: none"> Избира и използва изучавани софтуерни приложения за анализ и обработка на данни. Прави оценка на уебсайта и сравнява с други. Съзнава необходимостта от непрекъсната актуализация на уебсайта. 	<ul style="list-style-type: none"> Поддръжка на уебсайта
ТЕМА 4.2. Нужда от оптимизация на уебсайта;	<ul style="list-style-type: none"> Знае нуждата от оптимизация на уеб сайта и я прилага. 	<ul style="list-style-type: none"> Оптимизация

Ключови компетентности	Примерни дейности и междупредметни връзки
<i>Компетентности в областта на българския език</i>	<ul style="list-style-type: none"> Създаване на таблици с информация, която е обмислена, форматирана, систематизирана . Анализиране информация, публикувана в интернет, и оценяване надеждността ѝ.
<i>Умения за общуване на чужди езици</i>	<ul style="list-style-type: none"> Използване на помощни материали на чужд език при намиране на решение на технически проблем в дигитална среда. Работа с приложения, където интерфейсът изцяло е на английски език;
<i>Математическа компетентност и основни компетентности в областта на природните науки и на технологиите</i>	<ul style="list-style-type: none"> Създаване на алгоритъм за решаване на математически проблем. Анализиране и обработка на данни, които са свързани с изучаван материал по природните науки и технологиите.
<i>Умения за учене</i>	<ul style="list-style-type: none"> Развиване на критическо мислене чрез търсене и намиране на грешки и тяхното коригиране. Използване на изучавана среда за експериментиране. Използване на онлайн средства за самооценка.
<i>Социални и граждански компетентности</i>	<ul style="list-style-type: none"> Създаване на навици за работа с информация; Създаване на навици за проучване на нагласите на съучениците по актуален проблем. Изграждане на толерантно отношение чрез приемането на различни гледни точки при съвместна работа в екип.
<i>Инициативност и предприемчивост</i>	<ul style="list-style-type: none"> Планиране на дейностите, ефективно разпределение във времето и адекватен избор на начини, средства и материали за постигане на учебната задача. Избор от алтернативни възможни решения при отстраняване на технически проблем.
<i>Културна компетентност и умения за изразяване чрез творчество</i>	<ul style="list-style-type: none"> Създаване на проекти - уеб сайтове с подходящ визуален уеб дизайн, аудио озвучаване, спазвайки уеб стандарти.
<i>Умения за подкрепа на устойчивото развитие и за здравословен начин на живот и спорт</i>	<ul style="list-style-type: none"> Изготвяне проучване за влиянието на дигиталните технологии върху човешкото здраве и околната среда.

РАЗПРЕДЕЛЕНИЕ НА ЗАДЪЛЖИТЕЛНИТЕ УЧЕБНИ ЧАСОВЕ ЗА ГОДИНАТА

Допълнителни уточнения за факултативните часове по Уебдизайн:

С цел осигуряване на постоянство в обучението, се осъществява по два часа блок, един ден в седмицата. По-голямата част от часовете се организират под формата на комбиниран урок, по време на който учениците, освен че получават нови знания, изпълняват и практически задачи. Всеки ученик работи самостоятелно на компютърна работна станция.

Контролните работи се провеждат под формата на тест и не обхващат цял учебен час, тъй като е включена и практическа задача.

Препоръчително процентно разпределение на часовете:

За нови знания и умения	50%
За обобщение, упражнения и работа по проект	30%
За контролни работи	20%

СПЕЦИФИЧНИ МЕТОДИ И ФОРМИ ЗА ОЦЕНЯВАНЕ НА ПОСТИЖЕНИЯТА НА УЧЕНИЦИТЕ

Поради спецификата и разнообразния характер на очакваните резултати при оценяването на знанията и уменията на учениците могат да се използват различни методи и средства за проверка и оценка:

- *Тестове* - хартиен и/или компютърен формат.
- *Изпълняване на практически задачи, които се реализират на компютър в час.* Този тип задачи може да съдържа отделни компоненти, които измерват усвояването на конкретни умения за: работа с изучавания софтуер, извличане на информация, създаване на модели, творческо трансформиране и представяне на различни видове информация в дигитален формат и др.

Съотношение при формиране на срочна и годишна оценка:

Текущи оценки от устни, от писмени и от практически изпитвания върху конкретна задача	40%
Оценки от контролни работи	40%
Оценки от работа по проекти и домашни работи по предварително зададени критерии	20%

ДЕЙНОСТИ ЗА ПРИДОБИВАНЕ НА КЛЮЧОВИТЕ КОМПЕТЕНТНОСТИ, КАКТО И МЕЖДУПРЕДМЕТНИ ВРЪЗКИ

Обучението по Уебдизайн, поставя акцент върху придобиване на дигиталните компетентности, които са нужни на всеки ученик за успешно продължаване на обучението и реализация на пазара на труда. При реализирането на тази основна цел могат да се изпълняват редица дейности, които реализират междупредметни връзки и развиват и останалите ключови компетентности. Те ще умеят да представят информация и проекти по всички учебни дисциплини:

ПРИЛОЖЕНИЕ 3: УЧЕБНА ПРОГРАМА 4



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

Приложение: № 2

ТЕМАТИЧНА ПРОГРАМА И ГРАФИК ЗА РАБОТА НА КЛУБА „СЪЗДАВАНЕ НА ИНТЕРАКТИВЕН ИНТЕРФЕЙС“

Представяне на дейността:

Дигиталните компетенции се явяват съществена част от съвременната техническа култура на човека. Като поколение на XXI век, учениците са наясно с необходимостта от развиване на дигитални умения, които да са предпоставка за успешна реализация в училищната дейност и на по-късен етап, и в живота. Учениците да се създава интерактивна презентация от информация не е просто забавление за учениците, а необходимост. **Дейността на програмата** основно ще заложи върху формиране на базови дигитални умения за търсене и споделяне на информация в интернет, различните начини за комуникация в интернет, авторското право и зачитането му, създаване на творчески проект – интерактивен интерфейс с помощта на различни софтуерни технологии.

I. Цели:

- Развиване и обогатяване на дигиталната култура на учениците.
- Усвояване на ключови понятия свързани с интерактивния интерфейс, използване на средства, методи и технологии за изграждането му.
- Умения за работа с информация, обработката и представянето ѝ по начин – лесно възприемащ се от потребителя.

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

- Да умеят да работят с различни технологии при изграждане на интерактивен интерфейс.
 - Да създават интерактивни презентации и да ги конвертират във видео.
 - Да създават презентационни брошури.
 - Да изработват собствено лого и работа с графична програма.
 - Да изграждат постерна презентация.
 - Да вграждат кодове от различни програмни езици в уеб, да тестват уебсайта.
 - Да работят в реална интерактивна уеб среда.
 - Да използват среди за изграждане и редакция на уебстраници.
 - Да поддържат и оптимизират интерактивния си интерфейс.
-
- Използване на полезни сайтове, разграничаване на достоверната информация в интернет от фалшивата.
 - Познаване на същността на компютърната графика и видовете графични изображения. Работа с графични програми.
 - Развиване на логическото мислене и уменията да се обработва и представя информацията.
 - Умения за работа в екип.
 - Осъзнаване на рисковете в интернет и познаване на начините за предпазване.
 - Развиване на умения за представяне на проект пред публика.

II. Очаквани резултати:

- Да усвоят нови дигитални компетенции.
- Да придобият увереност за боравене с дигиталните устройства и информационните технологии.

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

- Повишаване на успеха на учениците, участващи в групата, като резултат от взаимовръзката на информационните технологии и възлаганите задачи с теми от други учебни предмети.
- Да презентират възможно най-точно, кратко и увлекателно своите проекти-интерактивни интерфейси.
- Да развият умения за търсене на информация в интернет, съобразно възложена задача.
- Да изградят навики за мислене за справяне с възникнал технически проблем.
- Да използват различни софтуерни технологии за изграждане на интерактивен интерфейс.
- Учениците самостоятелно да изработват авторски проекти-интерактивни интерфейси и да ги представят пред публика.
- Учениците да работят с желание и да изпитат удовлетворение от получения краен резултат.
- Работата в екип да развие чувство на отговорност и взаимопомощ.
- Да изграждат интерактивен интерфейс.
- Да създават интерактивни презентации и да ги превръщат във видео.
- Да умеят да съставят брошури.
- Да изграждат постерна презентация.
- Да работят във реална виртуална среда.
- Да познават структурата на един интерактивен интерфейс и да изграждат собствени навигации в уеб пространството.
- Да знаят набора от необходимите програмни средства, техники и технологии за изграждането на интерактивен интерфейс и др.

III. Разпределяне на съдържанието

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

№ по ред	Тема/Ядро на учебното съдържание	Дата на провеждане	Място на провеждане	Брой часове	Начален час
1.	Теоретични обосновки, изясняване основните понятия, свързани с изграждането на интерактивен интерфейс	01.12.2020 г.	Комп. кабинет	1 ч.	10.30 ч.
2.	Примерни технологии, методи, начини и средства за изграждане на интерактивен интерфейс	01.12.2020 г.	Комп. кабинет	1 ч.	11.20 ч.
3.	Избор на технологии за изграждане на интерактивен интерфейс	01.12.2020 г.	Комп. кабинет	1 ч.	12.10 ч.
4.	HTML – основи и изграждане на уеб страница	08.12.2020 г.	Комп. кабинет	1 ч.	10.30 ч.
5.	HTML – Упражнение, вмъкване на текст, картинка и др.	08.12.2020 г.	Комп. кабинет	1 ч.	11.20 ч.
6.	HTML – Упражнение, вмъкване натабличка, невидима таблица...	08.12.2020 г.	Комп. кабинет	1 ч.	12.10 ч.
7.	HTML – Упражнение, вмъкване на навигация	15.12.2020 г.	Комп. кабинет	1 ч.	10.30 ч.
8.	HTML – Упражнение, вмъкване на други интерактивни елементи	15.12.2020 г.	Комп. кабинет	1 ч.	11.20 ч.
9.	HTML – Упражнение, вмъкване на текст с ефекти и др.	15.12.2020 г.	Комп. кабинет	1 ч.	12.10 ч.

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

10.	HTML – Упражнение, тестване през различни браузъри..	22.12.2020 г.	Комп. кабинет	1 ч.	10.30 ч.
11.	Работа с MS Expression web 4 – избор на тема на интерактивния интерфейс	22.12.2020 г.	Комп. кабинет	1 ч.	11.20 ч.
12.	Упражнение – създаване на интерактивен интерфейс с програмата	22.12.2020 г.	Комп. кабинет	1 ч.	12.10 ч.
13.	Упражнение	05.01.2021 г.	Комп. кабинет	1 ч.	10.30 ч.
14.	Тестване през браузър и надграждане	05.01.2021 г.	Комп. кабинет	1 ч.	11.20 ч.
15.	Вмъкване на код в кода	05.01.2021 г.	Комп. кабинет	1 ч.	12.10 ч.
16.	Упражнение – сайтът https://www.w3schools.com/html/	12.01.2021 г.	Комп. кабинет	1 ч.	10.30 ч.
17.	MS Expression Web 4 – създаване на проект	12.01.2021 г.	Комп. кабинет	1 ч.	11.20 ч.
18.	Създаване на още уеб странички ..., вграждане на интерактивност	12.01.2021 г.	Комп. кабинет	1 ч.	12.10 ч.

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

19.	Авторското право и зачитането му. Цитиране на използваните източници, къде и как да се направи..	19.01.2021 г.	Комп. кабинет	1 ч.	10.30 ч.
20.	Работа с MS Expression web 4	19.01.2021 г.	Комп. кабинет	1 ч.	11.20 ч.
21.	Упражнение – създаване на интерактивен интерфейс	19.01.2021 г.	Комп. кабинет	1 ч.	12.10 ч.
22.	Упражнение - вмъкване на елементи	26.01.2021 г.	Комп. кабинет	1 ч.	10.30 ч.
23.	Работа по проекта	26.01.2021 г.	Комп. кабинет	1 ч.	11.20 ч.
24.	Друга среда за създаване на интерактивен интерфейс – Sublime Text 3.0	26.01.2021 г.	Комп. кабинет	1 ч.	12.10 ч.
25.	Работа с Notepad ++, конвертиране UTF 5	09.02.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
26.	Работа с програмите FileZilla	09.02.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
27.	Упражнение	16.02.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
28.	WinSCP	16.02.2021 г.	Комп. кабинет	1 ч.	16.30 ч.

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

29.	Упражнение	23.02.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
30.	FTP сървър, реално манипулиране на готов уебсайт-учебен	23.02.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
31.	Разглаждане на кодове от сайта	02.03.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
32.	Вграждане на кодове – елементи и тестване	02.03.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
33.	Валидиране и тестване през различни браузъри на сайта	09.03.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
34.	Календар в сайта	09.03.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
35.	Рисуване на лично лого с използване на графични програми	16.03.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
36.	Вмъкването на логото	16.03.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
37.	Изработка на постер	23.03.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
38.	Работа с CorelDraw	23.03.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
39.	Упражнение	30.03.2021 г.	Комп. кабинет	1 ч.	15.40 ч.

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

40.	Писане по крива	30.03.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
41.	Оформяне и задаване размери за постер	13.04.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
42.	Палитри, инструменти	13.04.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
43.	Основни опции в програмата	20.04.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
44.	Изработка на елементи като лого, постер и др.	20.04.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
45.	Упражнение в графичната векторна програма. С какво е по-добра от PhotoShop?	27.04.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
46.	Работа с постерната презентация	27.04.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
47.	Упражнение	04.05.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
48.	Създаване на електронна книга с интерактивен интерфейс	04.05.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
49.	Упражнение	11.05.2021 г.	Комп. кабинет	1 ч.	15.40 ч.

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

50.	Упражнение	11.05.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
51.	Упражнение	18.05.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
52.	Създаване на интерактивна презентация и конвертирането ѝ във видео	18.05.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
53.	Упражнение	01.06.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
54.	Упражнение	01.06.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
55.	Представяне на проекта	08.06.2021 г.	Комп. кабинет	1 ч.	15.40 ч.
56.	Системи за управление на съдържание. Видове. Препоръки и предпочитания	08.06.2021 г.	Комп. кабинет	1 ч.	16.30 ч.
57.	https://wordpress.com/	15.06.2021 г.	Комп. кабинет	1 ч.	9.00 ч.
58.	Създаване на интерактивен интерфейс безплатно със системата Wordpress	15.06.2021 г.	Комп. кабинет	1 ч.	9.50 ч.
59.	Упражнение	15.06.2021 г.	Комп. кабинет	1 ч.	10.40 ч.

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

60.	Упражнение	15.06.2021 г.	Комп. кабинет	1 ч.	11.30 ч.
61.	Упражнение	16.06.2021 г.	Комп. кабинет	1 ч.	9.00 ч.
62.	Упражнение	16.06.2021 г.	Комп. кабинет	1 ч.	9.50 ч.
63.	Упражнение	16.06.2021 г.	Комп. кабинет	1 ч.	10.40 ч.
64.	Упражнение	16.06.2021 г.	Комп. кабинет	1 ч.	11.30 ч.
65.	Упражнение	17.06.2021 г.	Комп. кабинет	1 ч.	9.00 ч.
66.	Упражнение	17.06.2021 г.	Комп. кабинет	1 ч.	9.50 ч.
67.	Създаване на брошура	17.06.2021 г.	Комп. кабинет	1 ч.	10.40 ч.
68.	Упражнение	17.06.2021 г.	Комп. кабинет	1 ч.	11.30 ч.
69.	Упражнение	18.06.2021 г.	Комп. кабинет	1 ч.	9.00 ч.
70.	Представяне на проектите	18.06.2021 г.	Комп. кабинет	1 ч.	9.50 ч.

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ
СОЦИАЛЕН ФОНД



ОПЕРАТИВНА ПРОГРАМА
НАУКА И ОБРАЗОВАНИЕ ЗА
ИНТЕЛИГЕНТЕН РАСТЕЖ

		Общо:			

Ръководител: Детелина Милкоцева

Директор: Величка Йорданова

www.eufunds.bg

Проект BG05M2OP001-2.012-0001 „Образование за утрешния ден“, финансиран от Оперативна програма „Наука и образование за интелигентен растеж“, съфинансирана от Европейския съюз чрез Европейските структурни и инвестиционни фондове.

ПРИЛОЖЕНИЕ 4: ПРИМЕРНА HTML СТРАНИЦА, разработана от ученици

```
<HTML>
<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso 8859-1">
<META NAME="descripton" CONTENT="EVRI – Bulgarian Engineering Company">
<META NAME="keywords" CONTENT="EVRI, studies, designs, construction, production, introduction, trade of technologies, machines and equipment, drying lines, grinding, pneumatic transport, pneumatic separation and dust elimination of ores, minerals and tail materials, and using the futher, products, translation, transport services">
<META NAME="author" CONTENT="ABC Design & Communication, 1999-2000">
<TITLE>EVRY – engineering Company</TITLE>
</HEAD>
<BODY TEXT="#000000" BGCOLOR="#FFFFFF" LINK="#0000FF" VLINK="#800080">
<CENTER><TABLE CELLSPACING=10 COLS=2 WIDTH="500"><TR><TD ALIGN=LEFT VALIGN=TOP><FONT FACE="Verdana, Geneva, Arial, Helvetica, Sans" COLOR="#666666" SIZE=-1>
<FONT COLOR="#2233FF" SIZE=5><B>EVRI</B></FONT>
<br><B>Bulgarian Engineering Company</B>
<br>
<br>
<br>Studies, designs, construction, production, introduction, trade of technologies, machines and equipment, drying lines, grinding, pneumatic transport, pneumatic separation and dust elimination of ores, minerals and tail materials, products, transport services.
<br>
<br><A HREF="">evri-tech.html</A>Technologies</A>
<br><A HREF="">evri-dd.html</A>Design Documentation</A>
<br><A HREF="">evri-rw.html</A>Research Works</A>
<br><A HREF="">evri-m.html</A>Manufacture</A>
<br><A HREF="">evri-contact.html</A>Contact us</A>
</FONT></TD>
<TD ALIGN=RIGHT VALIGN=TOP><IMG SRC="filter2.jpg" BORDER=0 HEIGHT=278 WIDTH=228></TD>
</TR></TABLE></CENTER>
<CENTER><A HREF="http://www.abcbg.com/index.htm" TARGET="abc"><IMG SRC="abc.gif" WIDTH=104 HEIGHT=42 BORDER=0></A></CENTER>
</BODY>
</HTML>
```

ПРИЛОЖЕНИЕ 5: HTML ТАГОВЕ И АТРИБУТИ

Tag	Description
<code><!--...--></code>	Defines a comment
<code><!DOCTYPE></code>	Defines the document type
<code><a></code>	Defines a hyperlink
<code><abbr></code>	Defines an abbreviation or an acronym
<code><acronym></code>	Not supported in HTML5. Use <code><abbr></code> instead. Defines an acronym
<code><address></code>	Defines contact information for the author/owner of a document
<code><applet></code>	Not supported in HTML5. Use <code><embed></code> or <code><object></code> instead. Defines an embedded applet
<code><area></code>	Defines an area inside an image map
<code><article></code>	Defines an article
<code><aside></code>	Defines content aside from the page content
<code><audio></code>	Defines embedded sound content
<code></code>	Defines bold text
<code><base></code>	Specifies the base URL/target for all relative URLs in a document
<code><basefont></code>	Not supported in HTML5. Use CSS instead. Specifies a default color, size, and font for all text in a document
<code><bdi></code>	Isolates a part of text that might be formatted in a different direction from other text outside it
<code><bdo></code>	Overrides the current text direction
<code><big></code>	Not supported in HTML5. Use CSS instead. Defines big text
<code><blockquote></code>	Defines a section that is quoted from another source
<code><body></code>	Defines the document's body
<code>
</code>	Defines a single line break
<code><button></code>	Defines a clickable button
<code><canvas></code>	Used to draw graphics, on the fly, via scripting

[HTML Tags \(1\).pdf](#)

	(usually JavaScript)
<u><caption></u>	Defines a table caption
<u><center></u>	Not supported in HTML5. Use CSS instead. Defines centered text
<u><cite></u>	Defines the title of a work
<u><code></u>	Defines a piece of computer code
<u><col></u>	Specifies column properties for each column within a <colgroup> element
<u><colgroup></u>	Specifies a group of one or more columns in a table for formatting
<u><data></u>	Adds a machine-readable translation of a given content
<u><datalist></u>	Specifies a list of pre-defined options for input controls
<u><dd></u>	Defines a description/value of a term in a description list
<u></u>	Defines text that has been deleted from a document
<u><details></u>	Defines additional details that the user can view or hide
<u><dfn></u>	Specifies a term that is going to be defined within the content
<u><dialog></u>	Defines a dialog box or window
<u><dir></u>	Not supported in HTML5. Use instead. Defines a directory list
<u><div></u>	Defines a section in a document
<u><dl></u>	Defines a description list
<u><dt></u>	Defines a term/name in a description list
<u></u>	Defines emphasized text
<u><embed></u>	Defines a container for an external application
<u><fieldset></u>	Groups related elements in a form
<u><figcaption></u>	Defines a caption for a <figure> element
<u><figure></u>	Specifies self-contained content
<u></u>	Not supported in HTML5. Use CSS instead.

	Defines font, color, and size for text
<u><footer></u>	Defines a footer for a document or section
<u><form></u>	Defines an HTML form for user input
<u><frame></u>	Not supported in HTML5. Defines a window (a frame) in a frameset
<u><frameset></u>	Not supported in HTML5. Defines a set of frames
<u><h1> to <h6></u>	Defines HTML headings
<u><head></u>	Contains metadata/information for the document
<u><header></u>	Defines a header for a document or section
<u><hr></u>	Defines a thematic change in the content
<u><html></u>	Defines the root of an HTML document
<u><i></u>	Defines a part of text in an alternate voice or mood
<u><iframe></u>	Defines an inline frame
<u></u>	Defines an image
<u><input></u>	Defines an input control
<u><ins></u>	Defines a text that has been inserted into a document
<u><kbd></u>	Defines keyboard input
<u><label></u>	Defines a label for an <input> element
<u><legend></u>	Defines a caption for a <fieldset> element
<u></u>	Defines a list item
<u><link></u>	Defines the relationship between a document and an external resource (most used to link to style sheets)
<u><main></u>	Specifies the main content of a document
<u><map></u>	Defines an image map
<u><mark></u>	Defines marked/highlighted text
<u><meta></u>	Defines metadata about an HTML document
<u><meter></u>	Defines a scalar measurement within a known range (a gauge)
<u><nav></u>	Defines navigation links

	Use or <s> instead. Defines strikethrough text
<code></code>	Defines important text
<code><style></code>	Defines style information for a document
<code><sub></code>	Defines subscripted text
<code><summary></code>	Defines a visible heading for a <code><details></code> element
<code><sup></code>	Defines superscripted text
<code><svg></code>	Defines a container for SVG graphics
<code><table></code>	Defines a table
<code><tbody></code>	Groups the body content in a table
<code><td></code>	Defines a cell in a table
<code><template></code>	Defines a container for content that should be hidden when the page loads
<code><textarea></code>	Defines a multiline input control (text area)
<code><tfoot></code>	Groups the footer content in a table
<code><th></code>	Defines a header cell in a table
<code><thead></code>	Groups the header content in a table
<code><time></code>	Defines a specific time (or datetime)
<code><title></code>	Defines a title for the document
<code><tr></code>	Defines a row in a table
<code><track></code>	Defines text tracks for media elements (<code><video></code> and <code><audio></code>)
<code><tt></code>	Not supported in HTML5. Use CSS instead. Defines teletype text
<code><u></code>	Defines some text that is unarticulated and styled differently from normal text
<code></code>	Defines an unordered list
<code><var></code>	Defines a variable
<code><video></code>	Defines embedded video content
<code><wbr></code>	Defines a possible line-break

<u><noframes></u>	Not supported in HTML5. Defines an alternate content for users that do not support frames
<u><noscript></u>	Defines an alternate content for users that do not support client-side scripts
<u><object></u>	Defines a container for an external application
<u></u>	Defines an ordered list
<u><optgroup></u>	Defines a group of related options in a drop-down list
<u><option></u>	Defines an option in a drop-down list
<u><output></u>	Defines the result of a calculation
<u><p></u>	Defines a paragraph
<u><param></u>	Defines a parameter for an object
<u><picture></u>	Defines a container for multiple image resources
<u><pre></u>	Defines preformatted text
<u><progress></u>	Represents the progress of a task
<u><q></u>	Defines a short quotation
<u><rp></u>	Defines what to show in browsers that do not support ruby annotations
<u><rt></u>	Defines an explanation/pronunciation of characters (for East Asian typography)
<u><ruby></u>	Defines a ruby annotation (for East Asian typography)
<u><s></u>	Defines text that is no longer correct
<u><samp></u>	Defines sample output from a computer program
<u><script></u>	Defines a client-side script
<u><section></u>	Defines a section in a document
<u><select></u>	Defines a drop-down list
<u><small></u>	Defines smaller text
<u><source></u>	Defines multiple media resources for media elements (<video> and <audio>)
<u></u>	Defines a section in a document
<u><strike></u>	Not supported in HTML5.

HTML Attribute Reference

[< Previous](#)

[Next >](#)

HTML Attribute Reference

The table below lists all HTML attributes and what elements they can be used within:

Attribute	Belongs to	Description
accept	<input>	Specifies the types of files that the server accepts (only for type="file")
accept-charset	<form>	Specifies the character encodings that are to be used for the form submission
accesskey	Global Attributes	Specifies a shortcut key to activate/focus an element
action	<form>	Specifies where to send the form-data when a form is submitted
align	Not supported in HTML 5.	Specifies the alignment according to surrounding elements. Use CSS instead
alt	<area> , , <input>	Specifies an alternate text when the original element fails to display
async	<script>	Specifies that the script is executed asynchronously (only for external scripts)
autocomplete	<form> , <input>	Specifies whether the <form> or the <input> element should have autocomplete enabled
autofocus	<button> , <input> , <select> , <textarea>	Specifies that the element should automatically get focus when the page loads
autoplay	<audio> , <video>	Specifies that the audio/video will start playing as soon as it is ready
bgcolor	Not supported in HTML 5.	Specifies the background color of an element. Use CSS instead
border	Not supported in HTML 5.	Specifies the width of the border of an element. Use CSS instead
charset	<meta> , <script>	Specifies the character encoding
checked	<input>	Specifies that an <input> element should be pre-selected when the page loads (for type="checkbox" or type="radio")
cite	<blockquote> , , <ins> , <q>	Specifies a URL which explains the quote/deleted/inserted text
class	Global Attributes	Specifies one or more classnames for an element (refers to a class in a style sheet)
color	Not supported in HTML 5.	Specifies the text color of an element. Use CSS instead
cols	<textarea>	Specifies the visible width of a text area
colspan	<td> , <th>	Specifies the number of columns a table cell should span
content	<meta>	Gives the value associated with the http-equiv or name attribute
contenteditable	Global Attributes	Specifies whether the content of an element is editable or not
controls	<audio> , <video>	Specifies that audio/video controls should be displayed

coords	<area>	Specifies the coordinates of the area
data	<object>	Specifies the URL of the resource to be used by the object
data-*	Global Attributes	Used to store custom data private to the page or application
datetime	, <ins>, <time>	Specifies the date and time
default	<track>	Specifies that the track is to be enabled if the user's preferences do not indicate that another track would be more appropriate
defer	<script>	Specifies that the script is executed when the page has finished parsing (only for external scripts)
dir	Global Attributes	Specifies the text direction for the content in an element
dirname	<input>, <textarea>	Specifies that the text direction will be submitted
disabled	<button>, <fieldset>, <input>, <optgroup>, <option>, <select>, <textarea>	Specifies that the specified element/group of elements should be disabled
download	<a>, <area>	Specifies that the target will be downloaded when a user clicks on the hyperlink
draggable	Global Attributes	Specifies whether an element is draggable or not
enctype	<form>	Specifies how the form-data should be encoded when submitting it to the server (only for method="post")
for	<label>, <output>	Specifies which form element(s) a label/calculation is bound to
form	<button>, <fieldset>, <input>, <label>, <meter>, <object>, <output>, <select>, <textarea>	Specifies the name of the form the element belongs to
formaction	<button>, <input>	Specifies where to send the form-data when a form is submitted. Only for type="submit"
headers	<td>, <th>	Specifies one or more headers cells a cell is related to
height	<canvas>, <embed>, <iframe>, , <input>, <object>, <video>	Specifies the height of the element
hidden	Global Attributes	Specifies that an element is not yet, or is no longer, relevant
high	<meter>	Specifies the range that is considered to be a high value
href	<a>, <area>, <base>, <link>	Specifies the URL of the page the link goes to
hreflang	<a>, <area>, <link>	Specifies the language of the linked document
http-equiv	<meta>	Provides an HTTP header for the information/value of the content attribute
id	Global Attributes	Specifies a unique id for an element
ismap		Specifies an image as a server-side image map
kind	<track>	Specifies the kind of text track
label	<track>, <option>, <optgroup>	Specifies the title of the text track
lang	Global Attributes	Specifies the language of the element's content
list	<input>	Refers to a <datalist> element that contains pre-defined options for an <input> element
loop	<audio>, <video>	Specifies that the audio/video will start over again, every time it is finished
low	<meter>	Specifies the range that is considered to be a low value
max	<input>, <meter>, <progress>	Specifies the maximum value
maxlength	<input>, <textarea>	Specifies the maximum number of characters allowed in an element
media	<a>, <area>, <link>, <source>, <style>	Specifies what media/device the linked document is optimized for
method	<form>	Specifies the HTTP method to use when sending form-

<code>onkeydown</code>	All visible elements.	Script to be run when a user is pressing a key
<code>onkeypress</code>	All visible elements.	Script to be run when a user presses a key
<code>onkeyup</code>	All visible elements.	Script to be run when a user releases a key
<code>onload</code>	<code><body></code> , <code><iframe></code> , <code></code> , <code><input></code> , <code><link></code> , <code><script></code> , <code><style></code>	Script to be run when the element is finished loading
<code>onloadeddata</code>	<code><audio></code> , <code><video></code>	Script to be run when media data is loaded
<code>onloadedmetadata</code>	<code><audio></code> , <code><video></code>	Script to be run when meta data (like dimensions and duration) are loaded
<code>onloadstart</code>	<code><audio></code> , <code><video></code>	Script to be run just as the file begins to load before anything is actually loaded
<code>onmousedown</code>	All visible elements.	Script to be run when a mouse button is pressed down on an element
<code>onmousemove</code>	All visible elements.	Script to be run as long as the mouse pointer is moving over an element
<code>onmouseout</code>	All visible elements.	Script to be run when a mouse pointer moves out of an element
<code>onmouseover</code>	All visible elements.	Script to be run when a mouse pointer moves over an element
<code>onmouseup</code>	All visible elements.	Script to be run when a mouse button is released over an element
<code>onmousewheel</code>	All visible elements.	Script to be run when a mouse wheel is being scrolled over an element
<code>onoffline</code>	<code><body></code>	Script to be run when the browser starts to work offline
<code>ononline</code>	<code><body></code>	Script to be run when the browser starts to work online
<code>onpagehide</code>	<code><body></code>	Script to be run when a user navigates away from a page
<code>onpageshow</code>	<code><body></code>	Script to be run when a user navigates to a page
<code>onpaste</code>	All visible elements.	Script to be run when the user pastes some content in an element
<code>onpause</code>	<code><audio></code> , <code><video></code>	Script to be run when the media is paused either by the user or programmatically
<code>onplay</code>	<code><audio></code> , <code><video></code>	Script to be run when the media has started playing
<code>onplaying</code>	<code><audio></code> , <code><video></code>	Script to be run when the media has started playing
<code>onpopstate</code>	<code><body></code>	Script to be run when the window's history changes.
<code>onprogress</code>	<code><audio></code> , <code><video></code>	Script to be run when the browser is in the process of getting the media data
<code>onratechange</code>	<code><audio></code> , <code><video></code>	Script to be run each time the playback rate changes (like when a user switches to a slow motion or fast forward mode).
<code>onreset</code>	<code><form></code>	Script to be run when a reset button in a form is clicked.
<code>onresize</code>	<code><body></code>	Script to be run when the browser window is being resized.
<code>onscroll</code>	All visible elements.	Script to be run when an element's scrollbar is being scrolled
<code>onsearch</code>	<code><input></code>	Script to be run when the user writes something in a search field (for <code><input="search"></code>)
<code>onseeked</code>	<code><audio></code> , <code><video></code>	Script to be run when the seeking attribute is set to false indicating that seeking has ended
<code>onseeking</code>	<code><audio></code> , <code><video></code>	Script to be run when the seeking attribute is set to true indicating that seeking is active
<code>onselect</code>	All visible elements.	Script to be run when the element gets selected
<code>onstalled</code>	<code><audio></code> , <code><video></code>	Script to be run when the browser is unable to fetch the media data for whatever reason
<code>onstorage</code>	<code><body></code>	Script to be run when a Web Storage area is updated

<code>min</code>	<code><input></code> , <code><meter></code>	Specifies a minimum value
<code>multiple</code>	<code><input></code> , <code><select></code>	Specifies that a user can enter more than one value
<code>muted</code>	<code><video></code> , <code><audio></code>	Specifies that the audio output of the video should be muted
<code>name</code>	<code><button></code> , <code><fieldset></code> , <code><form></code> , <code><iframe></code> , <code><input></code> , <code><map></code> , <code><meta></code> , <code><object></code> , <code><output></code> , <code><param></code> , <code><select></code> , <code><textarea></code>	Specifies the name of the element
<code>novalidate</code>	<code><form></code>	Specifies that the form should not be validated when submitted
<code>onabort</code>	<code><audio></code> , <code><embed></code> , <code></code> , <code><object></code> , <code><video></code>	Script to be run on abort
<code>onafterprint</code>	<code><body></code>	Script to be run after the document is printed
<code>onbeforeprint</code>	<code><body></code>	Script to be run before the document is printed
<code>onbeforeunload</code>	<code><body></code>	Script to be run when the document is about to be unloaded
<code>onblur</code>	All visible elements.	Script to be run when the element loses focus
<code>oncancel</code>	<code><audio></code> , <code><embed></code> , <code><object></code> , <code><video></code>	Script to be run when a file is ready to start playing (when it has buffered enough to begin)
<code>oncanplaythrough</code>	<code><audio></code> , <code><video></code>	Script to be run when a file can be played all the way to the end without pausing for buffering
<code>onchange</code>	All visible elements.	Script to be run when the value of the element is changed
<code>onclick</code>	All visible elements.	Script to be run when the element is being clicked
<code>oncontextmenu</code>	All visible elements.	Script to be run when a context menu is triggered
<code>oncopy</code>	All visible elements.	Script to be run when the content of the element is being copied
<code>oncuechange</code>	<code><track></code>	Script to be run when the cue changes in a <code><track></code> element
<code>oncut</code>	All visible elements.	Script to be run when the content of the element is being cut
<code>ondblclick</code>	All visible elements.	Script to be run when the element is being double-clicked
<code>ondrag</code>	All visible elements.	Script to be run when the element is being dragged
<code>ondragend</code>	All visible elements.	Script to be run at the end of a drag operation
<code>ondragenter</code>	All visible elements.	Script to be run when an element has been dragged to a valid drop target
<code>ondragleave</code>	All visible elements.	Script to be run when an element leaves a valid drop target
<code>ondragover</code>	All visible elements.	Script to be run when an element is being dragged over a valid drop target
<code>ondragstart</code>	All visible elements.	Script to be run at the start of a drag operation
<code>ondrop</code>	All visible elements.	Script to be run when dragged element is being dropped
<code>ondurationchange</code>	<code><audio></code> , <code><video></code>	Script to be run when the length of the media changes
<code>onemptied</code>	<code><audio></code> , <code><video></code>	Script to be run when something bad happens and the file is suddenly unavailable (like unexpectedly disconnects)
<code>onended</code>	<code><audio></code> , <code><video></code>	Script to be run when the media has reach the end (a useful event for messages like "thanks for listening")
<code>onerror</code>	<code><audio></code> , <code><body></code> , <code><embed></code> , <code></code> , <code><object></code> , <code><script></code> , <code><style></code> , <code><video></code>	Script to be run when an error occurs
<code>onfocus</code>	All visible elements.	Script to be run when the element gets focus
<code>onhashchange</code>	<code><body></code>	Script to be run when there has been changes to the anchor part of the a URL.
<code>oninput</code>	All visible elements.	Script to be run when the element gets user input

<code>onsuspend</code>	<code><audio></code> , <code><video></code>	Script to be run when fetching the media data is stopped before it is completely loaded for whatever reason
<code>ontimeupdate</code>	<code><audio></code> , <code><video></code>	Script to be run when the playing position has changed (like when the user fast forwards to a different point in the media)
<code>ontoggle</code>	<code><details></code>	Script to be run when the user opens or closes the <code><details></code> element
<code>onunload</code>	<code><body></code>	Script to be run when a page has unloaded (or the browser window has been closed)
<code>onvolumechange</code>	<code><audio></code> , <code><video></code>	Script to be run each time the volume of a video/audio has been changed
<code>onwaiting</code>	<code><audio></code> , <code><video></code>	Script to be run when the media has paused but is expected to resume (like when the media pauses to buffer more data)
<code>onwheel</code>	All visible elements.	Script to be run when the mouse wheel rolls up or down over an element
<code>open</code>	<code><details></code>	Specifies that the details should be visible (open) to the user
<code>optimum</code>	<code><meter></code>	Specifies what value is the optimal value for the gauge
<code>pattern</code>	<code><input></code>	Specifies a regular expression that an <code><input></code> element's value is checked against
<code>placeholder</code>	<code><input></code> , <code><textarea></code>	Specifies a short hint that describes the expected value of the element
<code>poster</code>	<code><video></code>	Specifies an image to be shown while the video is downloading, or until the user hits the play button
<code>preload</code>	<code><audio></code> , <code><video></code>	Specifies if and how the author thinks the audio/video should be loaded when the page loads
<code>readonly</code>	<code><input></code> , <code><textarea></code>	Specifies that the element is read-only
<code>rel</code>	<code><a></code> , <code><area></code> , <code><form></code> , <code><link></code>	Specifies the relationship between the current document and the linked document
<code>required</code>	<code><input></code> , <code><select></code> , <code><textarea></code>	Specifies that the element must be filled out before submitting the form
<code>reversed</code>	<code></code>	Specifies that the list order should be descending (9,8,7,...)
<code>rows</code>	<code><textarea></code>	Specifies the visible number of lines in a text area
<code>rowspan</code>	<code><td></code> , <code><th></code>	Specifies the number of rows a table cell should span
<code>sandbox</code>	<code><iframe></code>	Enables an extra set of restrictions for the content in an <code><iframe></code>
<code>scope</code>	<code><th></code>	Specifies whether a header cell is a header for a column, row, or group of columns or rows
<code>selected</code>	<code><option></code>	Specifies that an option should be pre-selected when the page loads
<code>shape</code>	<code><area></code>	Specifies the shape of the area
<code>size</code>	<code><input></code> , <code><select></code>	Specifies the width, in characters (for <code><input></code>) or specifies the number of visible options (for <code><select></code>)
<code>sizes</code>	<code></code> , <code><link></code> , <code><source></code>	Specifies the size of the linked resource
<code>span</code>	<code><col></code> , <code><colgroup></code>	Specifies the number of columns to span
<code>spellcheck</code>	Global Attributes	Specifies whether the element is to have its spelling and grammar checked or not
<code>src</code>	<code><audio></code> , <code><embed></code> , <code><iframe></code> , <code></code> , <code><input></code> , <code><script></code> , <code><source></code> , <code><track></code> , <code><video></code>	Specifies the URL of the media file
<code>srcdoc</code>	<code><iframe></code>	Specifies the HTML content of the page to show in the <code><iframe></code>
<code>srclang</code>	<code><track></code>	Specifies the language of the track text data (required if <code>kind="subtitles"</code>)

<code>start</code>	<code></code>	Specifies the start value of an ordered list
<code>step</code>	<code><input></code>	Specifies the legal number intervals for an input field
<code>style</code>	Global Attributes	Specifies an inline CSS style for an element
<code>tabindex</code>	Global Attributes	Specifies the tabbing order of an element
<code>target</code>	<code><a></code> , <code><area></code> , <code><base></code> , <code><form></code>	Specifies the target for where to open the linked document or where to submit the form
<code>title</code>	Global Attributes	Specifies extra information about an element
<code>translate</code>	Global Attributes	Specifies whether the content of an element should be translated or not
<code>type</code>	<code><a></code> , <code><button></code> , <code><embed></code> , <code><input></code> , <code><link></code> , <code><menu></code> , <code><object></code> , <code><script></code> , <code><source></code> , <code><style></code>	Specifies the type of element
<code>usemap</code>	<code></code> , <code><object></code>	Specifies an image as a client-side image map
<code>value</code>	<code><button></code> , <code><input></code> , <code></code> , <code><option></code> , <code><meter></code> , <code><progress></code> , <code><param></code>	Specifies the value of the element
<code>width</code>	<code><canvas></code> , <code><embed></code> , <code><iframe></code> , <code></code> , <code><input></code> , <code><object></code> , <code><video></code>	Specifies the width of the element
<code>wrap</code>	<code><textarea></code>	Specifies how the text in a text area is to be wrapped when submitted in a form

ПРИЛОЖЕНИЕ 6: ВХОДЯЩА АНКЕТА- ОБРАЗОВАТЕЛНИ СТАТУСИ И КОМПЕТЕНЦИИ

ВХОДЯЩА АНКЕТА: ОБРАЗОВАТЕЛЕН СТАТУС - ДИАГНОСТИКА И КОМПЕТЕНЦИИ

Внимание: Анкетата е анонимна

С тази анкета се цели да се събере допълнително информация за нуждите и възможностите на учениците по Информационни технологии и изцяло дигитални компетенции като базисни знания, умения и отношения, за да можем да предоставим по-добро обучение по Уебдизайн и изграждане на интерактивен интерфейс в часовете – занимания по интереси, модул Уебдизайн, работа по Проект Образование за утрешния ден и други извънкласни форми за уплътняване времето на учениците в направление - развитие и дигитализация. Анкетата се състои от следните части:

- за Вашите познания и опит
- за Вашето натоварване и разписание на учебните часове

Въпрос 1: В каква степен владеете алгоритмичния подход за изграждане на проект?

- а) във висока степен
- б) в средна степен
- в) в ниска степен

Въпрос 2: Владеете ли езици за програмиране, софтуерни технологии за изграждане на динамично представяне на информация, както и на интерактивното ѝ представяне в уеб?

- а) във висока степен
- б) в средна степен
- в) в ниска степен

Въпрос 3: Разполагате ли със собствен персонален или преносим компютър и познавате ли параметрите му?

- а) нямам на разположение компютър, нито лаптоп
- б) имам добър персонален компютър/лаптоп и познавам характеристиките му
- в) имам съвременен лаптоп/компютър, но не познавам характеристиките му

Въпрос 4: В каква степен владеете и използвате програмните продукти на Microsoft Office (Microsoft Word, Microsoft Excel, Power Point, MS Publisher и други от пакета)?

- а) във висока степен
- б) в средна степен

в) в ниска степен

Въпрос 5: Имате ли електронна поща и използвате ли я в работата си в обучението?

а) имам електронна поща, но почти не я ползвам

б) използвам редовно електронната си поща за обмен на електронни ресурси в училище

в) имам създадена електронна поща, но не си спомням паролата

г) не използвам електронна поща

Въпрос 6: Използвате ли Интернет? За какво използвате услугите на Интернет?

а) за игри най-вече

б) за обмен на електронни ресурси в обучението ми по всички предмети

в) за комуникация с приятели

в) за всичко

Въпрос 7: Изучавали ли сте дисциплината Английски език, ако „да” посочете колко години сте го изучавали?

а) по-малко от седем години

б) повече от 4 години

в) по-малко от 3 години

Въпрос 8: Използвате ли математическата си логика за решаване на други проблеми от бита?

а) във висока степен

б) в средна степен

в) в ниска степен

Въпрос 9: Занимавате ли се самостоятелно в къщи или търсите помощ от някого. Разучавате ли съвременни софтуерни технологии за изграждане на интерактивна среда?

а) да

б) не

в) понякога

Въпрос 10: Винаги ли изпълнявате поставените задачи за домашна работа и проверявате ли дали тя е наистина вярна?

а) да б) не в) понякога

Благодаря за отговорите!

ПРИЛОЖЕНИЕ 7: ИЗХОДЯЩА АНКЕТА - ИНТЕРЕС ПО УЕБДИЗАЙН

ИЗХОДЯЩА АНКЕТА: ИНТЕРЕС ПО УЕБДИЗАЙН

Внимание: Анкетата е анонимна

С тази анкета се цели да се събере допълнително информация за това дали са предоставени нужните съвременни софтуерни технологии по подходящ начин, така че да се възприемат от всички ученици. Дали сте получили необходимите дигитални компетенции за изграждане на интерактивен интерфейс. Анкетата се състои от следните части:

- за Вашите познания след обучението и степен на Вашето удовлетворение
- за трудности при работата при изграждането на интерактивния интерфейс

Въпрос 1: Хареса ли Ви да изградите интерактивен интерфейс?

- а) да
- б) не
- в) не знам

Въпрос 2: Владеете ли езици за програмиране, софтуерни технологии за изграждане на динамично представяне на информация, както и на интерактивното ѝ представяне в уеб?

- а) да
- б) не
- в) малко

Въпрос 3: Бихте ли си изготвили Ваше ученическо портфолио под формата на интерактивен интерфейс?

- а) да б) не в) може

Въпрос 4: Какви уебсайтове предпочитате да изградите, динамични с CMS система за управление на съдържание или с други технологии, само за да е уникален Вашия сайт, без реклами и ограничения?

- а) с платформа

б) с кодове, изчистени уебсайтове

в) всякакви

Въпрос 5: Беше ли полезно обучението за Вас?

а) да б) не в) до някъде

Въпрос 6: Какво Ви затруднява най-много?

.....
.....
.....

Въпрос 7: Кой дизайн да изберем? Отзивчив или адаптивен? Защо?

.....
.....

Благодаря за отговорите!

ПРИЛОЖЕНИЕ 8: Тестове по Уебдизайн по теми – раздели с верните отговори

Тест 1 по Уебдизайн от темата ПЛАНИРАНЕ НА УЕБСАЙТ

Въпрос 1: Коя от целите НЕ е присъща за изграждане на уеб?

- а) да се информира обществото за дадена личност б) да се представи определена идея
- в) да се представят дейности на определена организация и фирми
- г) да се тестват периферните устройства на компютъра

Въпрос 2: Как ще наречете хората, към които ориентирате Вашия уебсайт?

- а) всички хора в уеб пространството б) целева група
- в) хора с моите предпочитания г) хората, участващи в уеб процесите

Въпрос 3: През какви етапи се преминава за изграждането на интерактивен интерфейс?

- а) Планиране, проектиране, изграждане, тестване, публикуване
- б) Планиране, проектиране, изграждане, тестване, публикуване, оптимизация
- в) Планиране, проектиране, тестване, публикуване, изграждане
- г) Планиране, изграждане, тестване публикуване, оптимизация

Въпрос 4: Хостингът е:

- а) услуга, която позволява да се качи сайт на сървър – в интернет б) име на сайта
в) домейн на сайта г) категория на сайта

Въпрос 5: Хостингът може да бъде:

- а) файлов и имейлов б) имейлов и споделен в) виртуален
г) файлов, имейлов, споделен, посветен, виртуален, облачен

Въпрос 6: Скалируемост на хостинга е:

- а) оказва влияние на бързината на зареждане на уеб сайта или приложението
б) свързва развитието на уеб сайта с възможността за повишаване на ресурсите на уеб сървъра при нужда – ъпгрейд или скалиране
в) управлява уеб хостинга чрез платформа
г) предоставя количество пространство, на което се съхраняват елементите на уебсайта или приложението

Въпрос 7: Името, с което ще се вижда уебсайта в интернет, се нарича:

- а) IP, URL б) стандартизиран адрес на ресурс в) хостинг г) домейн

Въпрос 8: IP адрес представлява:

- а) името, с което ще се вижда уеб сайта, когато се публикува в интернет
б) число, показващо уникален номер на свързан с интернет компютър
в) адрес на определени ресурси – документи, страници в интернет
г) услуга, която позволява да се качи сайта на сървър, така че да се вижда от потребителите

Въпрос 9: Специална база данни от компютри, услуги или други ресурси, чрез която се получава информация за домейните, се нарича:

- а) база данни б) система за имена на домейни
в) услуги, свързани с информационните технологии г) регистри

Въпрос 10: Адресите на определени ресурси – документ, страница в интернет, са:

- а) домейн на уебсайт б) хост на уебсайт
в) стандартизиран адрес за ресурс г) електронни ресурси

Въпрос 11: Кое НЕ е функция за планиране на уеб сайт?

- а) избор на технология за реализиране и изграждане на уебсайта
б) управлява уеб хостинга чрез система за управление на съдържание
в) определяне на архитектурата и навигацията на сайта

г) свързва развитието на уеб сайта с възможност за повишаване на ресурсите на уеб сървъра при нужда – ъпгрейд и скалиране

Тест 2 по Уебдизайн от темата ПРОЕКТИРАНЕ И ГРАФИЧЕН ДИЗАЙН

Въпрос 1: Визуално ръководство, което представлява най-общата рамка на една страница е:

- а) схематичен дизайн wireframe б) графичен дизайн
в) достъпност в уеб г) графичен потребителски интерфейс

Въпрос 2: Подпомагането на ползвателите с увреждания да получат достъп до информацията и функционалните възможности на уеб сайта, се нарича?

- а) предприемчивост б) лоялност
в) достъпност г) толерантност

Въпрос 3: определянето на дизайн, цетова схема, избор на шрифтове, изработване на графични компоненти, се нарича:

- а) подреждане на компонентите
б) графичен модел mockup
в) стандартизиране
г) уеб процес

Въпрос 4: Създаването на хармонична комбинация от цветовете е:

- а) палитра б) програмата Paint
в) други графични програми г) цетова схема

Въпрос 5: За моя уеб сайт е подходящо да ползвам шрифтът:

- а) Arial, Tahoma, Times New Roman б) Calibri в) артистичен шрифт г) няма значение, стига да е красив

Въпрос 6: Как се нарича етапът на създаване на информационната архитектура и избора на програмните средства за изграждане на интерактивен интерфейс?

- а) изграждане б) тестване в) публикуване и оптимизация г) проектиране

Въпрос 7: Стилното подреждане на обектите – елементите и съдържанието на уеб сайта, се нарича:

- а) достъпност б) схематичен дизайн в) графичен интерфейс г) навигационни обекти

Въпрос 8: Графичен модел е:

- а) изгледът на една уебстраница
- б) избора на софтуерни технологии за реализация и поддръжка
- в)** дизайн, цветовете схема, шрифтове, изработване на други графични компоненти
- г) смесването на цветовете

Въпрос 9: Как назовават първата уеб страницка на Вашия уеб сайт? Възможни са повече от един верен отговор.

- а) index.htm
- б) index.css
- в) index.html
- г) home.page

Въпрос 10: WordPress, Joomla Drupal, Plone и други са:

- а)** системи за управление на съдържание
- б) програми за сърфиране в интернет
- в) вид услуги
- г) електронни ресурси

Въпрос 11: Попълнете думите в текста.

Организацията в уеб сайта се показва с помощта на **блок диаграми**. За по-добра нагледност уеб дизайнерът може да постави **карта на сайта**. За да се визуализира добре на всякакви дигитални устройства уеб сайтът трябва да е **отзивчив – ориентиран към потребителите**. Основни структурни части на уеб страница са: заглавна част, навигационна, съдържание в контекстни полета и **долна част**. Човешкото око обикновено се насочва в **центъра на уеб страницата**, за това трябва да слагаме там **най-важната информация**.

Тест 3 по Уебдизайн от темата ИЗГРАЖДАНЕ, ТЕСТВАНЕ И ПУБЛИКУВАНЕ

Въпрос 1: Връзка към друг текст в интернет, се нарича?

- а)** хипервръзка
- б) пренасочване
- в) онагледяване
- г) отклоняване

Въпрос 2: Нормативни спецификации не технологии методологии за визуализиране на уеб съдържание, се нарича:

- а) уеб спецификации
- б) уеб процеси
- в)** уеб стандарти
- г) уеб приложения

Въпрос 3: Общ публичен лиценз на свободен софтуер, по който авторът предава програмното обезпечение в обществена собственост е:

- а) URL
- б) HTML, XML
- в) WWW.FREE-CSS.COM
- г)** GNU General Public License

Въпрос 4: html tag е:

- а) задава основни или допълнителни характеристики
- б) описва се в отварящия tag
- в) структурен компонент
- г)** етикет, заграден с ъглови скоби

Въпрос 5: Таг за параграф е:

- а) б) <p>...</p> в)
 г) <u>

Въпрос 6: Може ли рамката на таблицата да се направи невидима за уеб:

- а) Да
б) Не
в) сигурно има начин със специална програма
г) не знам

Въпрос 7: Името, с което ще се вижда уебсайта в интернет, се нарича:

- а) IP, URL б) стандартизиран адрес на ресурс в) хостинг г) домейн

Въпрос 8: Често в уеб се налага да се изграждат радиобутони с кодове. Те представляват:

- а) хиперлинкове б) текстова кутия
в) бяло поле за избор на възможности повече от една
г) използват се за избор между няколко възможности, но само една

Въпрос 9: С коя технология може да улесните работата си, като трябва да промените шрифта на целия уеб сайт на всичките му страници?

- а) с технологията CSS б) JavaScript
в) технологията HTML има всички необходими тагове за това г) PHP, FLASH

Въпрос 10: За да изградите интерактивен интерфейс за вашия уеб сайт, който да е унифициран, изчистен и без никакви ограничения и реклами, коя технология ще предпочетете? Възможни са повече от един отговор.

- а) CMS б) MVC.net
в) MS Expression Web г) Среда, в която сам поставям кодове

Въпрос 11: Попълнете пропуснатите думи в текста?

Сайт за тестване е <http://www.webpagetest.org/>. Примерен сайт за безплатен хостинг е www.freehosting.com/.

За да изкача уеб сайтът ми най-отгоре в търсачките, аз трябва да направя **оптимизация** и **регистрация в търсачки и директории**. Подреджане на оценяваните сайтове в реда на тяхното подобряване или влошаване на качество, се нарича **рангово** подреджане. А думите, с които се открива информация от определен сайт, се наричат **ключови**. След като сайтът е изграден, нужно е непрекъснато да се **поддържа и актуализира**.

Тест 4 по Уебдизайн от темата СИСТЕМИ ЗА УПРАВЛЕНИЕ НА УЕБСЪДЪРЖАНИЕ

Въпрос 1: Кое служи за съхранение на данни и тяхното използване?

- а) бизнес слой business tier
- б) гугъл диск google drive
- в) слой за данни back-end**
- г) клиентски слой front-end

Въпрос 2: Система за управление на съдържание е? Възможен е повече от един отговор

- а) CSS
- б) SHOKWAVE
- в) CMS**
- г) Drupal

Въпрос 3: Трислойната архитектура на система за управление на съдържание включва:

- а) клиентски слой
- б) бизнес слой
- в) слой с данни
- г) трите изброени**

Въпрос 4: За колко реално техническо време може да се инсталирате система за управление на съдържание:

- а) ако всичко е наред, за 5 минути**
- б) става за не по-малко от час
- в) 20-30 минути
- г) инсталацията е бавна

Въпрос 5: Ограничена ли е работата с платформа на система за управление на съдържание?

- а) за жалост Да**
- б) Не
- в) зависи
- г) ако е добра интернет връзката - Не

Въпрос 6: Кое от изброените НЕ е CMS?

- а) Drupal
- б) Plone
- в) Joomla
- г) MySQL**

Въпрос 7: В WordPress има модули, какво представляват те?

- а) блокове с информация, които правят така, че определена функция става активна на уеб сайта**
- б) изготвени тематични шаблони
- в) администраторски панели за управление на системата
- г) падащо меню

Въпрос 8: Кой компонент НЕ е необходим за инсталиране на система за управление на съдържание WordPress?

- а) инсталирана поддръжка на PHP
- б) инсталиран HTML редактор за разработка на уеб сайт**
- в) инсталирана система за управление на базата данни MY SQL
- г) уеб сървър инсталиран на компютъра

Въпрос 9: Система за управление на съдържание е:

- а) база данни
- б) система за имена на домейни

в) механизъм за обмен на новини между два сайта г) софтуер за създаване, публикуване и редактиране на уеб

Въпрос 10: Кое не се отнася за системите за управление на съдържание?

- а) визуално ръководство б) лесно публикуване на информация
в) съхранение на онлайн на информация г) възможност да се дава определено ниво на достъп до категории от съдържанието или отделни части от него

Въпрос 11: Попълнете пропуските в текста?

Лесно и бързо създаване, публикуване и редактиране на уеб сайт, става с **платформа на система за управление на съдържание**. **Клиентски слой** служи за пряко взаимодействие с потребителя и изпращане на заявки към бизнес слоя.

А **бизнесслоя**, служи за обработка на данните и работните процеси. Всички данни използват **слой за данни**, в който се съхраняват данните и **тяхното използване**.

Тест 5 по Уебдизайн от темата СИГУРНОСТ В УЕБ

Въпрос 1: Когато правим паролите си, те трябва да:

- а) са силни б) да съдържат главна буква
в) да съдържат число и специални символи г) всичко изброено

Въпрос 2: Защитата на информацията срещу достъп, използване, разкриване, увреждане, промяна, преглед, запис или изтриване от неупълномощени лица, се нарича:

- а) уеб процес б) кибератака в) уеб сигурност г) If spoofing

Въпрос 3: Малък по размер файл, който се създава и изпраща от уеб сайта – сървъра и се записва от уеб браузъра на потребителя в системното пространство, се нарича:

- а) http бисквитки
б) маяци
в) Dos атаки
г) трите изброени

Въпрос 4: Графично изображение, което изследва уеб активността на потребителя е:

- а) маяк б) бисквитка
в) спам г) фишинг

Въпрос 5: Защо понякога в сайтовете като ги ползваме, понякога искат да решим даден тест с автоматизирана програма робот?

- а) за да проверят, че не сме робот б) за да е по-интересен сайта в) не знам г) за проверка на посещаемостта

Въпрос 6: Кой от сайтовете предоставя хост и домейн? Проверете дали няма повече от един отговор.

- a) alle.bg **б) dom.bg** **в) icn.bg** г) ucd.bg

Въпрос 7: Тест за сигурност, използван да се определи дали потребителят на дадена система е човек или компютър се нарича:

- а) публичен ключ за сигурност Captcha** б) Sniffing в) DoS г) IP Spoofing

Въпрос 8: Кой от начините НЕ се отнася за защита на системи за управление на съдържание?

- а) редовно архивиране на сайта
б) инсталиране на антивирусна програма
в) защита на административния панел
г) проверка на потребителите, които се регистрират в сайта

Въпрос 9: Средството за криптиране на информационния канал между клиент и сървър е?

- а) бисквитка б) публичен ключ за сигурност
в) допълнителна защита **г) сертификат за сигурност**

Въпрос 10: Криптографията е:

- а) свързана с кибер атаките** б) калиграфия
в) информационни технологии г) бюротика

Въпрос 11: Попълнете пропуските в текста?

Публичен ключ за сигурност е – тест за сигурност използван да се определи дали потребителят на дадена система е човек или компютър. Технологията за криптиране на данни и защита на канала, по който се извършва комуникацията на сървър се нарича **сертификат за сигурност**.
Допълнителна функционалност за защита Plug-in е независимо компилиран софтуерен **модул**, който се инсталира в допълнение към уеб браузъра. За да се предотврати рискът от триене и повреда на данни в уеб сайта, добре е да се направи **резервно копие backup**.

ИЗПОЛЗВАНА ЛИТЕРАТУРА:

1. Гъргов, К. и А. Ангелов, Уеб дизайн, изд. „Изкуства“, София, 2013, ISBN: 978-954- 9463-86-6.
2. Гъргов, К., Мултимедия, изд. „Изкуства“, София, 2013, ISBN: 978-954-9463-87-3
3. WEB DESIGN – Над 600 цветни примера за вдъхновяващи интерфейс решения; Автори: Джек Дейвис и Сюзън Мерит, изд. „Тракия-М“ София 2001 г.;
4. Как да си направим собствен сайт? Автор: Денис Колисниченко, издателство „Асеновци“;
5. Основи на web дизайна; Автори: Жустин Томс, Венцислав Джамбазов; издателство „Сиела – софт ЕНД ПАБЛИШИНГ“, София 2004;
6. Уеб дизайн, Автори: Коста Гъргов, Ангел Ангелов...;издателство „Изкуства“ ;
7. Информационни технологии; Автор: Ивайло Иванов, Людмила Попова...;изд. „Нова звезда“;
8. Софтуерни технологии; Автори: Нели Манева, Аврам Ескенази; изд. „Анубис“ София 2001 г.;
9. Проектиране с уеб стандарти; Автор: Джефри Зелдман, изд. „Софтпрес“;
10. WEB. ДИЗАЙН – Ръководство на Якоб Нилсен /Световноизвестен експерт по интерфейси и удобство на ползване на Web - сайтове/, изд. „Софтпрес“;
11. Софтуерни технологии – Университетско издателство „Паисий Хилендарски“; Автори: Стоян Стоянов, А. Стоянова – Дойчева...;
12. WEB ДИЗАЙН НАКРАТКО –Настолено бърз справочник; Автор: Дженифър Нийдерст, изд. “O'REILLY”;
13. PHP 5 – Професионално програмиране, Автор: Брус Перънс; изд. „Софтпрес“;
14. <http://www.w3schools.com/html/default.asp>
15. <https://www.sublimetext.com/3>
16. <http://getbootstrap.com/getting-started/>
17. Принципи на уебдизайна, Автор: Джоел Склар – трето издание; издателство DuoDesign;
18. <http://java.sun.com/alets/>
19. <http://WWW.W3.ORG/TR/REC-CSS1>
20. <http://w3.org/style/CSS>
21. The Definitive Guide, автор: Eric Meyer (O' Reilly, 2000)
22. Списание: WebReview (<http://webreview.com>)
23. <http://webreview.com/style/>
24. PHP 5 Поредица за отворен код на Брус Перънс – Професионално програмиране, Автори: Анди Гутманс, Стив Баккен и Дерик Ританс;
25. Принципи на уебдизайна – четвърто издание; Автор: Джоел Склар, издателство DuoDesign;
26. Уебдизайн със SeaMonkey – практическо ръководство; Автор: Петя Асенова, изд. „Просвета“;
27. Jenkins, Sue. Web Design All-in-One, Wiley Publishing Inc., 2009;
28. Evans, Mark, Michael Hamm. Introduction to Web Design Using Dreamweaver, The McGraw-Hill Companiers Inc., 2007;
29. УЕБ сайтът – Мисията задължителна; Автор: Жюстин Томс, Горица Белогушева, издателство „Сиела“;
30. <https://phalconphp.com/bg/>
31. www.html.am

32. Колисниченко, Д., HTML5 & CSS3 – практическо програмиране за начинаещи, ИК “Асеневици”, 2009, ISBN:978-9548898-13-3;
33. <http://w3schools.com>;
34. <http://developer.mozilla.org/en-US/>;
35. <http://docs.webplatform.org>
36. <http://www.maxdesign.com.au/css>;
37. <http://www.thesitewizard.com/>;
38. УЕБ ДИЗАЙН – стратегии за българските уебсайтове; Автори: Жустин Томс и Венцислав Джамбазов, издателство: ”Сиела софт енд павлишинг”;
39. seovisia.com
40. <http://beshared.bg>
41. Създаване на уеб страници – в лесни стъпки, издателство „СофтПрес“, Автор: Колектив на издателство СофтПрес, СофтПрес ООД, 2005; ISBN 954-685-351-8;
42. Web услуги – XML, WSDL, SOAP, UDDI; Автор: Ерик Нюкъмър, издателство „СофтПрес“, СофтПрес ООД, 2005; ISBN 954-685-318-6;
43. Java за World Wide Web, Автор: Дори Смит, превод – Димитър Томов; издателство “Peachpit Press”; ISBN: 0-201-35340-7;
44. Дизайн, хора и взаимодействие; Автор: Антон Радев, 2013; издателство „Парадигма“ 2013, ISBN: 978-954-326-195-6;
45. CSS 3 в примери, книгоиздателство „Асеневици“, 2009, Автор – Сергей Соколов; ISBN: 978-954-8898-13-3;
46. Уебдизайн – Основни принципи; издателство „Просвета - София“ АД, 2012; Автор: Христина Дачева, Николай Нетов, Тодор Георгиев, ISBN: 978-954-01-2630-2;
47. DHTML и CSS за World Wide Web, Автор – Джейсън Кранфърд Тийг; Издателство „СофтПрес“ ООД, 2005; ISBN 954-685-310-0;
48. Macromedia FLASH MX 2004 – официален учебен курс, Автор – Йен Дехаан; издателство „СофтПрес“ ООД, 2004; ISBN 954-685-296-1;
49. <http://www.parvisait.com/site-registration-search-engines.html>;
50. Уебдизайн; Основни принципи; Издателство „Просвета - София“ АД – София 1618, ул. „Земеделска“ 2; ISBN: 978-954-01-2630-2;
51. Уеб страници в лесни стъпки – необходим учебник за всеки от нас, издателство: „Софтпрес“ ООД, 2005 г.; Автор: колектив на издателство СофтПрес; ISBN: 954-685-351-8;
52. Java за World Wide Web – Практическо визуално ръководство; Автор: Дори Смит, превод – Димитър Томов; Издателство: „Инфо Дар“ ЕООД; ISBN: 0-201-35340-7;
53. Macromedia FLASH MX 2004; Официален учебен курс; Автор: Йен Дехан; Изданието на български език е публикувано от Издателство Софт Прес ООД, 2004; ISBN: 954-685-296-1;
54. DHTML и CSS за word wide web; Автор: Джейсън Тийг, изданието на български език е публикувано от издателство Софт Прес ООД 2005; ISBN: 554-685-310-0;
55. Дизайн, хора и взаимодействия; Автор: Антон Радев; Издателство: „Парадигма“, 2013 г.; ISBN: 978-954-326-195-6;
56. https://www.google.com/webmasters/tools/site-message-view?hl=bg&authuser=0&siteUrl=http://www.su-ohridski.eu/&message-filter=all&grid.r=0&msgid=AB9YKzL8JNpf6lt-3tIzZ_R-szQggnJhTnLU725Bc3JfZ2eU0YPdqu0vNIQbtNtyNtY2rpt1y5byO4a8qg4-hsafXyUmYDjxng;

57. www.searchenginewatch.com – списък на машините за търсене, описания, оценки и тестове, както и съвети как сайтът ви да бъде намиран лесно с основните търсачки;
58. <https://bg.wikipedia.org>;
59. Автор: Роджърс Кейдънхед „Научете сами Java“, издателство „Алекс Софт“, ISBN 954656348-X, www.alexsoft.net;
60. „Изграждане на интернет сайт for DUMMIES“. Автор: Джейсън Кумбс, Тед Кумбс, Дейвид Краудър, Ронда Краудър, издателство „Софтпрес“, ISBN 954-685-035-7;
61. Java for Dummies – Автор: Д-р Бари Бърд, издателство „Алекс Софт“, ISBN 954656274-2;
62. HTML5 & CSS3 for Dumies – Автор: Дейвид Карлинс, издателство „Алекс Софт“, ISBN 954656316-1;
63. Адаптивен уеб дизайн с Bootstrap – Автор: Денис Николаевич Колисниченко, издателство „Асеновци“, ISBN 978-619-7356-17-5;
64. HTML for Dummies – Автор: Ричрд огнър, издателство „Алекс Софт“, ISBN 954656228-9;
65. ActionScript 3 – шаблони за дизайн; Автор: Джоуи Лот и Дани Патерсън, издателство „СОФТПРЕС“, ISBN 978-954-685-813-9;
66. „Web услуги – XML, WSDL, SOAP и UDDI“, Автор: Ерик Нюкъмър, издателство „Софт прес“ ООД, 2005. ISBN: 954 685 318 6;
67. Изграждане на интернет сайт for Dumies – Автор: Джейсън Куумбс, Дейвид Краудър и Ронда Краудър, издателство „Софтпрес“, ISBN 954685035-7;
68. Дурева Д. Проблеми от методиката на обучение по информатика и ИТ, Университетско издателство „Н. Рилски“, Благоевград, 2003

СЪДЪРЖАНИЕ

ВЪВЕДЕНИЕ	3
I. Обща характеристика на дисертацията	7
II. ГЛАВА ПЪРВА – ОСНОВНИ ТЕОРЕТИЧНИ ИЗСЛЕДВАНИЯ И ЕТАПИ В ИЗГРАЖДАНЕТО НА ИНТЕРАКТИВЕН ИНТЕРФЕЙС	10
1. Word Wide Web.....	12
1.1. Първата най-важна и отговорна задача при създаването на един уеб сайт с интерактивен интерфейс.....	13
1.2. Планирането на уебсайт.....	13
1.3. Предизвикателство към графичния дизайн и основни задачи.....	14
1.4. Основни принципи на графичния дизайн.....	17
1.5. Ролята на ученикът – уебдизайнерът.....	18
1.6. Концепция.....	18
1.7. Дизайн.....	18
1.8. Производство.....	18
1.9. Маркетинг.....	18
1.10. Съдържание на уеб сайта.....	18
1.11. Технологията не е съдържанието.....	18
1.12. Какво може да направи технологията.....	18

1.13.	Съобразяване на технологията с аудиторията.....	18
1.14.	Създаване на единна метафора, индивидуалност и стил.....	20
1.15.	Избиране на метафора.....	20
1.16.	Индивидуалност.....	21
1.17.	Стил.....	21
1.18.	Създаване на план за организация и навигация.....	22
1.19.	Същност на интернет. Основи за изграждане на интерактивен интерфейс.....	24
1.20.	Бъдещето на уеб пространството.....	28
2.	Поколения уеб дизайн – исторически бележки.....	30
2.1.	Първо поколение.....	30
2.2.	Второ поколение.....	31
2.3.	Трето поколение.....	31
2.4.	Четвърто поколение.....	33
2.5.	Фактори, водещи до разлики при възпроизвеждането.....	37
2.6.	Верният подход.....	38
2.7.	Оценяване.....	39
3.	Принципи на уеб дизайна.....	40
3.1.	Изглед и усещане на сайта.....	40
3.2.	Направете дизайна си преносим.....	40
3.3.	Съобразете дизайна с бавна интернет връзка.....	40
3.4.	Предвидете лесен достъп до информацията.....	42
3.5.	Представете ясно информацията.....	42
3.6.	Плавни преходи.....	43
3.7.	Препоръчва се решетъчна структура на сайта.....	43
3.8.	Използвайте активно бяло поле.....	43
3.9.	Направете дизайна ориентиран към потребителя 1.....	44
3.10.	Как да направим унифициран дизайн за уеб?.....	44
3.11.	Важно е взаимодействието.....	45
3.12.	Разположението на елементите също е много важно.....	45
3.13.	Използвайте хипертекстови връзки.....	45
3.14.	Не претрупвайте страниците си.....	46
3.15.	Не забравяйте хората с увреждания.....	46
3.16.	Съобразете дизайна с компютърните монитори.....	46
3.17.	За сруктурата.....	47
3.18.	Направете дизайна отзивчив 2.....	54
3.19.	Интерактивен интерфейс в уеб страница.....	55
3.20.	Мултимедия в уеб страница.....	56
3.21.	Проблеми със съвместимостта на уеб браузърите.....	61
3.22.	Промоция на сайта.....	62
3.23.	Включване в търсещи машини.....	62
3.24.	Банери.....	62
3.25.	Рекламни електронни съобщения.....	63
3.26.	Реклама офлайн.....	64
3.27.	Данни за посещенията и статистика.....	64
4.	Рискове и препоръки при разработването на уеб сайт.....	65
4.1.	Как може да се изложите на риск?.....	65
4.2.	Закон на Фиц.....	66
4.3.	Петте елемента на дизайна.....	66
4.4.	Цвят.....	66

4.5. Форма.....	68
4.6. Линия.....	68
4.7. Текстура.....	69
4.8. Типография.....	69
4.9. Петте принципа на дизайна. Движение.....	71
4.10. Баланс на композицията.....	71
4.11. Близост.....	72
4.12. Повторение.....	72
4.13. Единство.....	72
4.14. Работа с изображения и медии.....	73
4.15. Използване на изображения в уеб дизайна - Лого.....	73
4.16. Банери.....	74
4.17. Бутони.....	75
4.18. Ролоувър.....	75
4.19. Карти на изображение (Image Maps).....	76
4.20. Видове графични файлови формати.....	77
4.21. Палитри.....	78
4.22. Цифрова фотография.....	79
4.23. Техники и концепции.....	79

III. ГЛАВА ВТОРА - МЕТОДИЧЕСКИ ИНСТРУМЕНТАРИУМ И РЕАЛИЗАЦИЯ НА МОДЕЛА.....80

1. Процесът на проектиране.....	80
1.1. Единен дизайн, използваемост и маркетинг.....	84
1.2. Влиянието на дизайна върху използваемостта и достъпността. Модели в уеб дизайна.....	85
1.3. Организация на сайта.....	85
1.4. Мерки с насоки на използваемост.....	86
1.5. Достъпност.....	86
2. Процес на разработка „Итеративен дизайн“.....	86
2.1. Етапи в процеса на дизайна.....	87
2.2. Карта на сайта.....	89
2.3. Схеми.....	92
2.4. Усъвършенстване.....	93
2.5. Писане за ефективна комуникация в мрежата.....	94
3. Осигуряване на качество.....	95
3.1. Техническо тестване.....	95
3.2. Тестване на използваемостта.....	96
4. Технологията HTML.....	97
4.1. Цветове.....	100
4.2. Общи елементи на страницата.....	102
4.3. Глава на HTML страница.....	102
4.4. Тяло на HTML документ.....	103
4.5. Защиаване в HTML.....	103
4.6. Някои специални символи.....	104
4.7. Форматиране.....	104
4.8. Таблици в HTML.....	105
4.8.1. Тагове в таблици.....	105
4.8.2. Параметри на таблица.....	106

4.8.3.	Създаване на „невидима“ таблица.....	107
4.9.	Рамки.....	108
4.9.1.	Кога се използват рамки?.....	109
4.9.2.	Как се създават рамки?.....	109
4.10.	Разширени възможности на HTML.....	111
4.11.	Връзки в HTML.....	111
4.12.	Графични изображения и връзки.....	112
4.13.	Фреймове.....	112
4.14.	Форми в HTML.....	113
4.15.	Стилове в HTML.....	114
4.16.	Избор на HTML редактор.....	115
5.	Технологията CSS – стилови правила в CSS.....	118
5.1.	Графика и цвят при уеб дизайна. Основи на компютърните цветове. Контрол на изображенията и цвета чрез CSS. Контрол на фонове изображения чрез CSS.....	124
5.2.	Контрол на цветовете със CSS.....	125
5.3.	Стандарти за визуално оформление.....	128
5.4.	Форматиране на фона със CSS.....	136
5.5.	Рамки със CSS.....	136
5.6.	Задаване на име и шрифт, размер на шрифт, цвят на шрифта.....	136
5.7.	Управление на абзаци, отстъп за нов ред.....	141
5.8.	Управление на заглавия, размер.....	142
5.9.	Управление на изображения.....	143
5.10.	Управление на таблици.....	145
5.11.	Предимства и недостатъци на CSS.....	147
6.	HTML и XHTML.....	149
7.	Употреба на FLASH в уеб страниците.....	151
8.	PHP – вграждане в HTML.....	154
9.	JAVA Applets.....	166
10.	Необходимост от списъци със стилове.....	167
10.1.	Миграция от HTML към XHTML.....	168
11.	Универсалния език XML.....	169
11.1.	Синтактични правила.....	172
11.2.	Сравнение на XML и HTML.....	175
11.3.	Синтаксис на XML.....	175
12.	Технологията DHTML.....	177
13.	Език за описание на уеб услуги:	
	WSDL.....	178
13.1.	Достъп до уеб услуги: SOAP.....	180
13.2.	Откриване на уеб услуги: UDDI регистър.....	181
13.3.	Алтернативният подход: ebXML.....	182
14.	JavaScript.....	182
15.	Аплети на Java.....	184
15.1.	Защита на уеб страницата с парола.....	186
16.	Създаване на анимация с FLASH.....	188
17.	Системи за управление на съдържание (CMS).....	190
17.1.	Най-известни системи за управление на съдържание.....	195
17.2.	Сигурност при системите за управление на съдържание.....	198
18.	Безопасност на уеб сайта.....	200
18.1.	Мерки за защита срещу кибератаки.....	200
19.	Общи условия за ползване.....	202

20. Поверителност на личните данни.....	202
21. Правила за съдържание.....	202
22. Специални препоръки за сайтове, насочени към деца.....	203
23. Списък с правни документи, които имат отношение към безопасността в Интернет.....	203
24. Интелектуална собственост и авторско право.....	204
25. Поддръжка.....	205
25.1. Обратна връзка.....	206
25.2. Кога сайтът е успешен?.....	208
26. Представяне на сайта на друг език.....	210
27. Хост.....	210
28. Тестване и публикуване.....	212
29. Регистрация на сайта в интернет търсачки и директории.....	213
30. Оптимизиране и реклама.....	217
31. Популяризиране.....	224
32. Учебна програма 1 по Уебдизайн.....	225
32.1. За методиката.....	236
33. Дидактически принципи, методи и средства.....	239
IV. ГЛАВА ТРЕТА - ОПИСАНИЕ НА ПЕДАГОГИЧЕСКО ИЗСЛЕДВАНЕ – ОБРАБОТКА И АНАЛИЗ НА ПОЛУЧЕНИТЕ ДАННИ.....	247
1. Общи положения.....	247
1.1. Организация на методическо изследване.....	248
1.2. Планиране на експерименти за проверка на хипотеза.....	248
1.3. Организация и провеждане на експериментите.....	248
2. Анализ на резултатите от изследването.....	249
2.1. Статистическо представяне на резултатите от проведеното изследване	
2.2. Резултати от входящите анкети на учениците.....	251
2.3. Статистически резултати от изходящите анкети на учениците.....	255
2.4. Използване на софтуерни технологии и приложения за обслужване на учебния процес.....	256
ЗАКЛЮЧИТЕЛНИ ИЗВОДИ ЗА ГЛАВА ТРЕТА.....	259
ПОЛЕЗНИ ИНТЕРНЕТ АДРЕСИ.....	260
РЕСУРСИ НА УЕБДИЗАЙНЕРА НА БЪЛГАРСКИ ЕЗИК.....	260
СТАСТИКИ, ПРОМОЦИИ.....	261
РЕГИСТРАЦИЯ НА ДОМЕЙН.....	261
ОРГАНИЗАЦИИ.....	261
ЕЛЕКТРОННИ ИЗДАНИЯ ЗА МРЕЖАТА.....	261
ДРУГИ ИНТЕРНЕТ ВРЪЗКИ.....	261

ПРИЛОЖЕНИЕ

Приложение 1: Учебна програма 2.....	263
Приложение 2: Учебна програма 3.....	275
Приложение 3: Учебна програма 4.....	279
Приложение 4: Примерна HTML страница, разработана от ученици.....	285
Приложение 5: HTML тагове и атрибути.....	286
Приложение 6: Входяща анкета - Образователни статуси и компетенции.....	296
Приложение 7: Изходяща анкета – Интерес към Уебдизайн.....	298
Приложение 8: Тестове по Уебдизайн по теми-раздели с верните отговори.....	298