

МИНИСТЕРСТВО ОБРАЗОВАНИЯ, НАУКИ И МОЛОДЕЖНОЙ
ПОЛИТИКИ КРАСНОДАРСКОГО КРАЯ

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ КРАСНОДАРСКОГО КРАЯ
«ЦЕНТР ДЕТСКОГО И ЮНОШЕСКОГО ТЕХНИЧЕСКОГО ТВОРЧЕСТВА»

Принято на заседании
методического совета
от «18» 10 2022 г.
Протокол № 1

УТВЕРЖДАЮ

Директор

«Центр детского и юношеского
технического творчества»

/ В.А. Щebetун /

Приказ № 643-об

от «18» 10

2022 г.



МЕТОДИЧЕСКОЕ ПОСОБИЕ
для педагогов дополнительного образования
технической направленности
«Виртуальная и дополненная реальность/
Информационные технологии»

«Создание сайта»

Автор-составитель:

Педагог дополнительного
образования

Константинов Иван Николаевич

г.Новороссийск
2022 г.

Содержание

Введение	3
1. Теоретические основы построения web-сайтов	6
1.1. Основная классификация современных web-сайтов	6
1.2. Структурные элементы web-сайта	9
1.3. Этапы разработки web-сайта	15
1.3.1. Планирование	15
1.3.2. Дизайн	16
1.3.3. Верстка страницы	17
2. Обзор технологий создания сайтов	22
3. Разработка сайта	26
3.1. Разработка структуры сайта	26
3.2. Верстка интерактивного веб-сайта	27
4. Продвижение сайта в поисковых системах	39
Заключение	43
Список использованной литературы и источников	45

Введение

Данное методическое пособие разработано педагогом дополнительного образования, реализующим дополнительные образовательные программы технической направленности в муниципальных образованиях Краснодарского края на базе мобильного комплекса «Кванториум» по направлению «Виртуальная и дополненная реальность/Информационные технологии».

«Создание сайта» - один из разделов дополнительной образовательной программы для обучения детей 10-18 лет. В сети Интернет есть множество платформ, конструкторов для разработки одностраничных и многостраничных сайтов. Изучив сайтостроение, школьники в будущем могут стать программистами, WEB-дизайнерами и быть востребованными на рынке труда.

В последнее время скорость роста в нашей стране количества WEB ресурсов увеличивается в геометрических пропорциях. Вместе с ростом количества WEB ресурсов растет количество пользователей сети Интернет. Для многих намного удобней предварительно ознакомиться с услугами тех, или иных организаций на их WEB узлах, прежде чем выходить на реальный контакт.

Существует много примеров, когда наличие хорошо спланированного ресурса в Интернет позволяло многим фирмам стать известными. Для многих мелких фирм и организаций нет никакой возможности заявить о себе в обычной рекламе, например, если это связано с повышенной информационной емкостью и как следствие с большой ценой за рекламу. Единственной возможностью заявить о себе в полном объеме, предоставить полную информацию о своих возможностях, услугах, товарах и ценах, заключается в наличии собственного Интернет-ресурса. Крупные корпоративные компании вынуждены использовать сложные сетевые технологии, для обеспечения связи между всеми своими членами.

В связи с ростом пользователей, для которых Интернет это не столько работа, сколько способ отдыха и развлечений появилось большое количество развлекательных серверов, которые не столько морально, сколько материально заинтересованы в создании более технически-совершенного и привлекательного, чем у конкурентов WEB-ресурса сервера.

Все эти факты делают вопрос о создании интересного оформленного, технически грамотно спланированного WEB - сайта, крайне актуальным.

Актуальность темы определяется тем, что Интернет - самое быстроразвивающееся средство вещания в истории человечества, по оценкам экспертов, количество пользователей удваивается каждые полгода. Пользователи сети представляют собой область повышенного интереса для большинства рекламодателей. Ни печатная продукция, ни тем более радио или телевидение не могут подавать информацию в таком удобном, презентабельном и убеждающем виде, как это делает WEB-сайт. Исчерпывающая информация, прайс-листы, каталог товаров, возможность

осуществлять всевозможные заказы, размещенные на WEB-сайте, избавляют от необходимости разъяснять одни и те же тривиальные вопросы по телефону или факсу, позволит высвободить рабочее время сотрудников фирмы для более эффективного решения насущных задач. Именно поэтому создание WEB-сайта для коммерческой организации продиктовано современной необходимостью.

Целью работы является создание WEB-сайта.

Для достижения поставленной цели были сформулированы и решены следующие задачи:

- изучена предметная область;
- разработана модель предметной области;
- выбран метод проектирования WEB-сайта;
- построена инфологическая модель WEB-сайта.

Объектом исследования является создаваемый сайт. Он нужен, в основном, для того, чтобы помочь его владельцу донести информацию о себе до интернет-пользователей, а также продать им свои услуги.

Предметом исследования являются технологии, средства разработки и языки программирования для создания WEB-сайта.

Гипотеза исследования: создание веб-сайта позволит агентству повысить эффективность продаж своих услуг.

Методы исследования. Методы работы основываются на принципах информационного моделирования предметной области. Исследование проводилось посредством изучения предметной области. Подбор необходимых материалов по разработке сайтов осуществлялся из Интернет.

Методическое пособие состоит из двух частей. Из теоретической и практической части. Целью аналитической части является рассмотрение существующего состояния предметной области, основных терминов и положений и т. д.

Глава 1 основана на информации, представленной в аналитической части, Глава 2 - обобщает ее. Главы 3, 4 - практическая часть, является решением проблематики, изложенной в аналитической части, на языке информационных технологий.

В заключении сделаны выводы по проекту, определены пути его внедрения на объекте и направления дальнейшего совершенствования.

Методическое пособие поможет педагогам дополнительного образования технической направленности «Виртуальная и дополненная реальность/Информационные технологии» в обучении детей сайтостроению. Так же педагог дополнительного образования, изучив методическое пособие, имеет возможность разработать свой собственный сайт – визитку, или образовательный сайт для размещения учебных материалов, используемый педагогом для более эффективной реализации дополнительных образовательных программ.

1. Теоретические основы построения web-сайтов

1.1 Основная классификация современных web-сайтов

Существует большое количество признаков, по которым можно произвести классификацию сайтов. Классификация сайтов является необходимой, так как зачастую веб-мастеру приходится сталкиваться с непониманием заказчика того, какой сайт должен получиться в итоге. Поэтому целесообразно выделение базовых признаков каждого вида. Кроме того, у сайтов разного типа различная, специфическая структура.

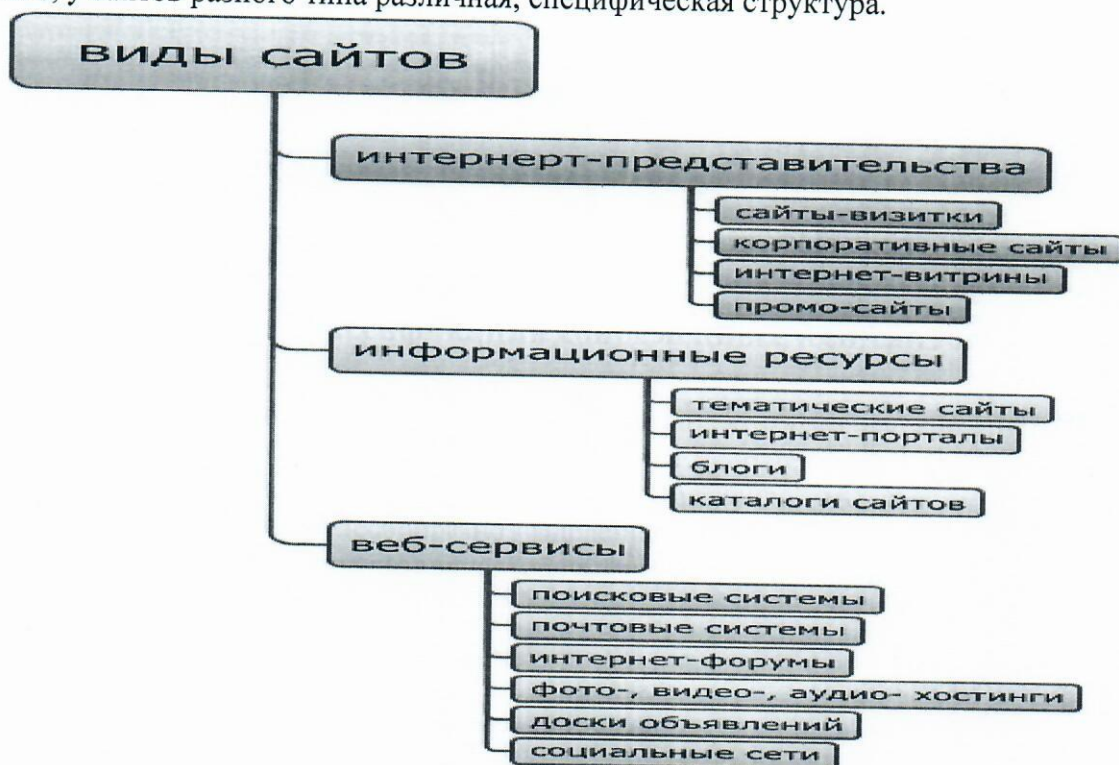


Схема 1. Виды сайтов

В первую очередь, как правило, выделяют сайты коммерческие и некоммерческие. Конечной целью функционирования коммерческого сайта всегда является прибыль, в то время как некоммерческие сайты выполняют, в основном, информационную функцию. Коммерческим можно назвать не только сайт, предназначенный для осуществления продаж либо предоставления услуг, но и промо-сайты, популяризирующие бренд, товар или фирму. Некоммерческие сайты чаще всего создаются общественными организациями, политическими партиями, государственными организациями. Также некоммерческими сайтами можно назвать ресурсы, предназначенные для обмена мнениями и организации общения людей.

Теперь более подробно рассмотрим каждый вид сайта.

1) Сайт визитка — самый простой вид сайта. Сайт такого типа можно сделать даже на простом HTML, без использования системы управления сайтом. Обычно сайт-визитка содержит от 1 до 5 страниц. Сайты этого вида, как правило, включают в себя только общую информацию о владельце сайта и его контактные данные.

Простота разработки такого вида сайта делает стоимость его создания сравнительно дешевой, что является очевидным преимуществом для заказчика.

2) Корпоративные сайты — это полнофункциональные представительства компаний в интернете. Этот тип сайта лучше всего подходит для серьезных средних и крупных фирм. Корпоративные сайты содержат полную информацию о компании и её

деятельности. Такой тип сайта иногда называют виртуальным офисом, так как посещение такого сайта сравнимо с общением с менеджером по работе с клиентами.

Корпоративные сайты нужны, в первую очередь, для формирования имиджа компании и предоставления посетителям и клиентам наиболее полной информации.

3) Интернет-витрина или интернет-каталог товаров — это вид сайтов, основная задача которых — продавать. На таких сайтах размещается информация о товарах и контакты, обычно телефоны, по которым следует звонить, желающим приобрести предлагаемый товар. На таких сайтах размещаются технические характеристики товаров, отзывы, рекомендации экспертов и т.д.

4) Интернет-магазины. Этот вид сайтов аналогичен интернет-витринам, но имеет дополнительный функционал: возможность заказать предлагаемый товар прямо через сайт.

5) Промо-сайты предназначены для раскрутки и продвижения какого-либо товара или бренда.

6) Тематические сайты. Данный тип интернет сайтов характеризуется тем, что содержит информацию по какой-либо конкретной тематике. Сюда можно отнести интернет-энциклопедии.

7) Интернет-порталы — это тип сайтов, содержащих большое количество разнообразной информации. Как правило, порталы схожи по структуре с тематическими сайтами, но имеют более развитый функционал и большее количество сервисов и разделов. Также на порталах часто бывают разделы для общения пользователей: чаты, блоги и форумы.

8) Блог — это тип сайтов, на которых владелец или редактор блога пишет посты со своими новостями, идеями или другой постоянно поступающей информацией. Отличительной особенностью блогов является актуальность публикуемой информации.

9) Каталоги сайтов - это вид сайтов, основным содержанием которых являются структурированные ссылки на другие сайты, а также их краткие описания.

10) Поисковые системы - вид сайтов, предназначенных для поиска страниц в интернете по определённым запросам.

11) Почтовые сервисы - этот тип сайтов предоставляет интерфейс для работы с электронной почтой.

12) Интернет-форумы. На сайтах этого вида пользователи могут создавать темы, а также комментировать их. Как правило, форумы ограничены одной специфической тематикой, хотя встречаются и форумы «обо всём».

13) Сайты-хостинги. На сайтах этого типа реализована функция хранения каких-либо файлов. Также часто встречаются сайты-хостинги с возможностью просмотра загруженных файлов прямо через браузер.

14) Доски объявлений. На таких сайтах пользователи могут размещать или искать информацию в виде каких-либо объявлений, например — о покупке-продаже.

15) Социальные сети. Тип сайтов, созданных для общения пользователей между собой. Как правило, на таких сайтах есть рейтинги, страницы пользователей, группы и множество других сервисов.

Таким образом, при планировании и разработке сайта в первую очередь необходимо определиться с типом сайта в зависимости от поставленных целей. И только после этого можно переходить к разработке структуры и содержания сайта.

1.2. Структурные элементы web-сайта

Любая web-страница содержит определенный набор стандартных элементов, являющихся обязательными компонентами каждого ресурса Интернета. Безусловно, ассортимент и количество подобных объектов могут варьироваться в зависимости от тематической направленности сайта, объема опубликованных на нем материалов, а также от целей и задач, которые ставит перед собой создатель данного ресурса. Компоновка

таких элементов, проектирование их взаимного расположения и составляет одну из главных задач web-мастера.

Основные элементы страницы.

Зачастую основными элементами страницы являются:

- ✓ содержащий блок
- ✓ логотип
- ✓ навигация
- ✓ контент
- ✓ Футер, нижний колонтитул
- ✓ свободное пространство (это не элемент дизайна, но понятие, помня о котором при составлении макета страницы, проект не будет выглядеть как нагромождение блоков).

Содержащий блок - контейнер. Роль контейнера на странице может выполнять непосредственно элемент `body` или же `div`. Ширина содержащего блока может быть резиновой - `fluid`, а может быть фиксированной - `fixed`.

Изначально разработчику сайта ширина окна браузера пользователя неизвестна, поскольку она может меняться в самых широких пределах. Ширина зависит от разрешения монитора, длины его диагонали, размера окна и еще некоторых варьируемых данных. Иными словами предугадать ее заранее простыми средствами не представляется возможным. С учетом этой особенности утвердилось два способа верстки: фиксированный и резиновый.

При работе с фиксированным макетом устанавливаем общую ширину макета жестко заданной и равной определенной величине. Если взять некоторую обобщенную статистику посетителей сайтов и посмотреть, какое разрешение монитора они преимущественно используют, то узнаем, что это 800 x 600 и 1024 x 768. Получается, что ширина монитора пользователей в основном 800 и 1024. Возьмем за ориентир 800, тогда общая ширина макета за вычетом вертикальной полосы прокрутки и рамки браузера окажется 770–790 пикселей. На этот размер ориентируемся и устанавливаем ширину макета, например 770 пикселей.

Резиновый макет основывается на том, что в качестве одной из единиц измерения выступают проценты. Общая рабочая ширина окна браузера — 100%, и колонки макета в сумме не должны ее превышать, поэтому для удобства, как правило, везде применяют процентную запись. При изменении размеров окна происходит переформатирование данных страницы, чтобы они вписались в новую ширину.

Этот вид верстки набирает все большую популярность и практически все известные сайты выбрали именно его в силу того, что эффективно задействуется вся площадь веб-страницы.

Логотип - текстовая или графическая составляющая проекта и выделяющая его среди других. Логотип чаще всего располагается в верхнем левом углу страницы или же посередине.

Навигация. Основная навигационная панель содержит ссылки на основные разделы сайта. Навигационная панель часто располагается в верхней части страницы.

Контент – это основная составляющая веб-страницы. Он занимает главенствующую роль в дизайне страницы, поэтому занимает большее пространство, подкреплён, помимо текста, графикой.

Нижний колонтитул. Данный элемент располагается внизу страницы и обычно содержит информацию о правообладателе, контактные и юридические данные, ссылки на основные разделы сайта, ссылки на социальные сети, форму обратной связи и пр.

Среди всего многообразия составления макета веб-страницы можно выделить наиболее распространённые:

1. Макет веб-страницы с навигацией в левом столбце (см.рис. 1).
2. Макет веб-страницы с навигацией в правом столбце (см.рис. 2).
3. Макет веб-страницы с навигацией в трёх столбцах (см.рис. 3).

4. Макет веб-страницы с горизонтальной навигацией

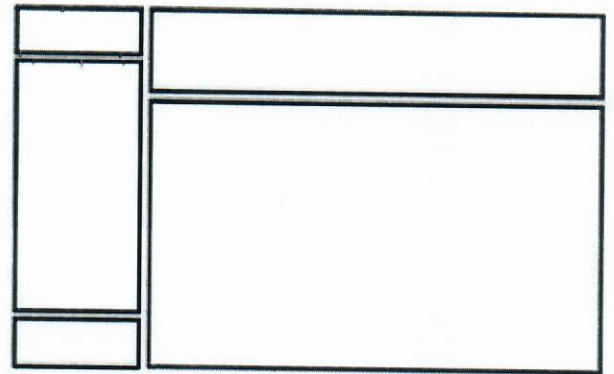
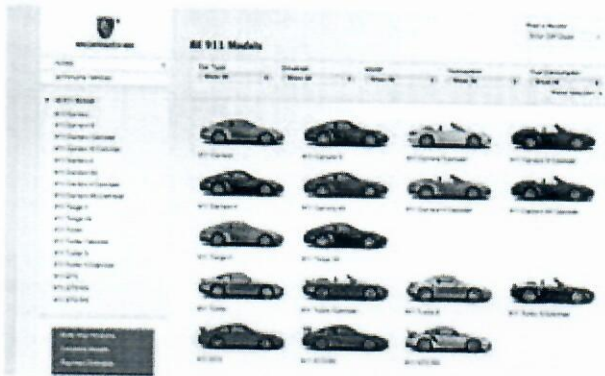


Рисунок 1. Навигация в левом столбце

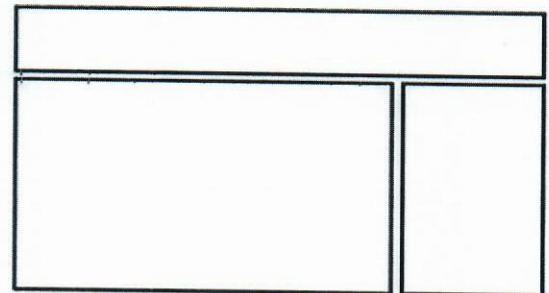
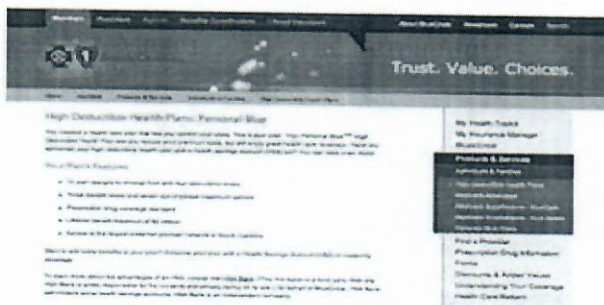


Рисунок 2. Навигация в правом столбце



Рисунок 3. Навигация в трёх столбцах

Горизонтальная навигация. На данном этапе сайты с таким типом навигации составляют большинство. Удобство такого подхода легко объяснить тем, что в данном случае у нас остаётся больше пространства для контента, составляющего наш сайт.

Нельзя не упомянуть о некоторых наметившихся трендах последнего времени в качестве компоновки и дизайна страниц.

Во-первых, стоит упомянуть о так называемых лендинговых страницах, которые подразумевают под собой длинную страницу, разделённую на соответствующие секции и знакомящие пользователя с основным контентом сайта. Часто лендинг является единственной страницей, на которой сразу удаётся показать всю необходимую

информацию, не заставляя пользователя переходить по страницам. Лэндингам обычно сопутствует хороший дизайн, выверенная и продуманная подача информации, элементы call-to-action, интерактивность.

Стоит так же сказать о внешнем виде страниц и вспомнить такие понятия как скевоморфизм и плоский дизайн.

Скевоморфизм уже продолжительное время уступает свои позиции плоскому дизайну. Данное понятие означает наделение интерактивных элементов качествами реальных. Например, оформление страницы с книгами в виде книжной полки, оформление кнопки на подобии настоящей с соответствующей имитацией нажатия, использование реальных текстур и пр. Данный принцип активно использовался при создании страниц буквально несколько лет назад, но затем тенденции сменились, и на первый план вышел плоский дизайн (см. рис. 4).



Рисунок 4. Плоский дизайн

Плоский дизайн использует минимум текстур и оформление, создание дизайна базируется на понятиях контраста, соотношения цветов и размеров.

Немаловажную роль на тренды в веб-дизайне оказало развитие мобильных операционных систем. Чаще всего двигателем прогресса становятся решения от компании Apple, которая в своих дизайн-решениях iOS сначала использовала имитацию реальных объектов, а затем всё упростила до плоского дизайна (Flat UI). Сейчас в вебе главенствует Flat дизайн и Material дизайн, который активно развивает Google (см.рис. 5).

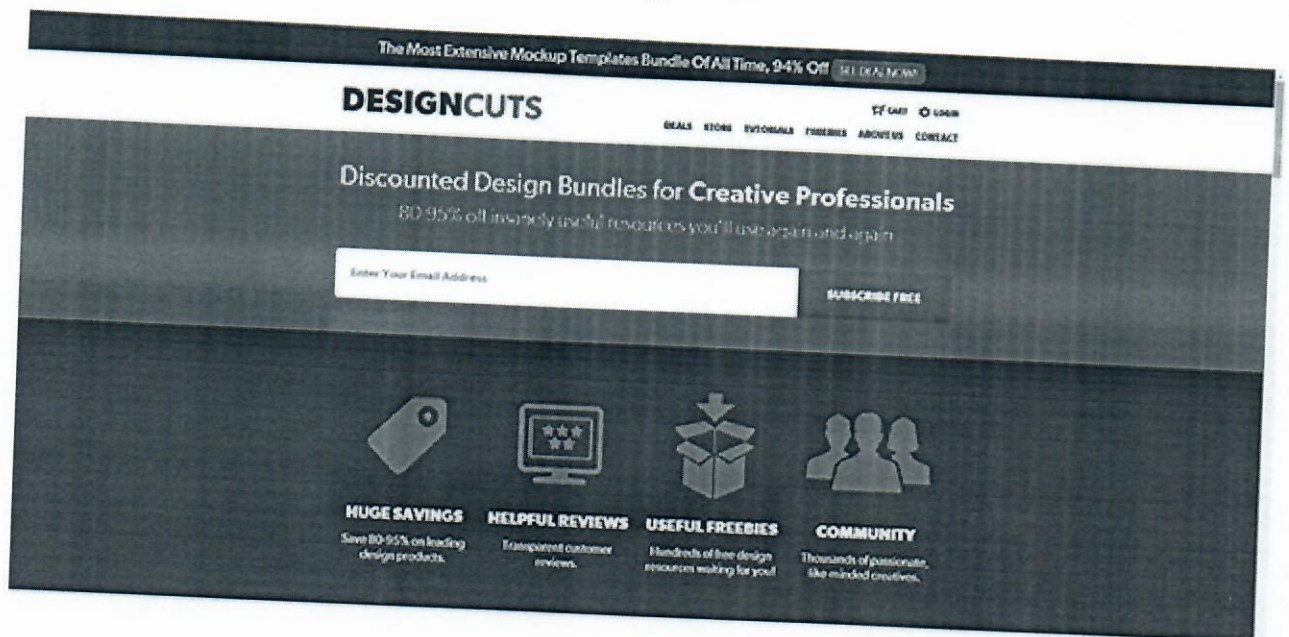


Рисунок 5. Поисковое окно

Под структурой проекта понимается хранение файлов проекта в его директории. Часто приходится видеть, когда все файлы сосредоточены вместе, названия файлам даны «капсом», цифрами или русскими буквами и пр. Во-первых, это банальное неуважение к тому, кто будет работать с проектом далее, во-вторых, чем больше будет проект, тем больше будет становиться файлов и, в конце концов, можно просто запутаться, что к чему относится и что нужно, а что нет.

Лучше всего отдельные категории файлов помещать в свои папки: картинки в папку *images* или *img*, *css* в папку *css*, *javascript* в папку *js*. В корне будет лежать только *index.html* и страницы сайта, либо только *index.html*, а страницы в отдельной папке *pages*.

Так же стоит сказать и об именовании файлов проекта. Чаще всего применяются следующие имена: главная страница – это *index.html*, стили проекта *styles.css*, скрипты *scripts.js* или *app.js*, минимизированные версии файлов имеют префикс *.min*, картинки носят не пространные названия на русском языке или набора цифр, а отражают то, что на них изображено, например, *download-icon.png*, *button.png*, *logo.png*, и т.д. (см. рис.6).

Рисунок 6. Файлы проекта

Итак, мы разобрали все основные компоненты web-страницы и их возможное расположение относительно друг друга. Безусловно, все, что было сказано в данном разделе, является не панацеей, а руководством к действию. Мы попытались изложить лишь общие принципы, которые применяются при компоновке структуры сайта, окончательное же решение всегда остается за web-мастером.

1.3 Этапы разработки web-сайта

Условно процесс создания сайта можно разделить на 3 этапа:

1. Планирование
2. Дизайн
3. Верстка страницы

1.3.1. Планирование

Данный этап можно разделить на несколько подэтапов:

- Создание идеи

На данном этапе необходимо определиться с тематикой проекта. Далее, в соответствии с выбранной темой, необходимо собрать соответствующие материалы: текстовые, графические.

- Разработка структуры проекта

Когда мы тема проекта определена, подобран необходимый материал, следующим этапом будет разработка структуры проекта. Структура проекта подразумевает под собой разделы сайта, в соответствии с которыми будет формироваться навигационное меню и строиться дизайн проекта. На данном этапе можно классифицировать материал по темам и разделам.

- Проработка макета проекта.

Далее составляется макет проекта. Для отрисовки наброска можно использовать бумагу и ручку, Photoshop или любой другой редактор графики. Важно отметить, что данный этап – это не отрисовка готового дизайн-макета, а всего лишь схематичный набросок, выполненный для понимания того, как на сайте будут располагаться основные информационные блоки, графика и прочие элементы дизайна.

1.3.2. Дизайн

После создания макета проекта можно переходить непосредственно к созданию дизайн-макета. На данном этапе начать стоит с определения цветовой гаммы проекта. Один из способов определения основного цвета в проекте – это составление mood board. Для этого необходимо выписать себе все синонимы, связанные с темой проекта, а затем каждый синоним набрать в поиске по картинкам Google. На основе найденных изображений выписать себе цвета, которые чаще всего встречаются на них. Найденные цвета будут составлять визуальное восприятие нашего проекта и вызывать у пользователя соответствующие чувства.

Для работы с выбранным цветом и составлением палитры цветов нашего сайта можно использовать следующие инструменты:

- Color Scheme Designer 3. Помимо выбора цветовой схемы данный сервис позволяет посмотреть пример того, как выбранные цвета будут смотреться на сайте
- Adobe Color CC. Данный ресурс, в отличие от Color Scheme Designer 3, позволяет создавать палитры ещё и на основе загруженных изображений. Так же данный сервис обладает большим архивом палитр других пользователей.
- COLORlovers. Обширное сообщество, где можно подобрать различные палитры.

Важно отметить, что при подборе цветов для палитры всегда стоит выбирать как минимум 2 контрастирующих цвета. Достижение нужного контраста между цветами – необходимое условие для того, чтобы получился хороший интерактивный дизайн.

При работе над дизайном главной и внутренних страниц стоит помнить о некоторых основных принципах.

Элементы Call to Action. Понятие призыва к действию относится к интерактивным элементам сайта: кнопки, баннеры и пр. Данные элементы оформляются таким образом, что пользователю должно хотеться непременно на них нажать. Например, это может быть кнопка с призывом к действию, яркий баннер с заманчивым предложением, яркой картинкой и пр.

Таким образом, становится понятен принцип построения дизайна, опирающегося на данное понятие: например, яркая картинка, баннер должны привлечь внимание пользователя, сопутствующий посыл в тексте должен вызвать в нём интерес и желание, а завершающим аккордом должна стать, например, кнопка с призывом к действию.

Но и данный принцип не работает сам по себе без некоторых других: схема просмотра страницы, визуальные направляющие.

Довольно часто встречается Z-схема просмотра страницы. В соответствии с этим элементы страницы обычно располагают следующим образом: логотип слева вверху, меню справа вверху, информационные блоки, картинки слева внизу, кнопка с призывом к действию справа внизу.

1.3.3. Верстка страницы

Итак, процесс дизайна макета страницы плавно перетекает в процесс «оживления» сделанного на предыдущих этапах.

Прежде всего стоит отметить, что вёрстка страницы делается поэтапно: сначала пишется HTML-структура, затем добавляются стили, а после, если необходимо, пишутся скрипты, добавляются необходимые плагины и библиотеки.

Учитывая вышесказанное, мы можем условно разделить работу над проектом на следующие этапы:

1. Написание HTML
2. Написание CSS
3. Написание JS

Сейчас при написании HTML кода уже смело можно использовать тэги и элементы разметки, которые появились вместе с стандартом HTML5, если необходимо поддерживать старые браузеры, то можно использовать, например, плагин `html5shiv`, который обеспечивает поддержку новых стандартов в старых браузерах или библиотеку `Modernizr` (`html5shiv` входит в сборку `Modernizr`), которая определяет возможности браузера, с помощью которого просматривается сайт.

На этапе написания HTML мы, как бы, создаём скелет страницы, её абстрактную модель при помощи тэгов. Стоит отметить, что структуру может быть проще написать, если у нас есть прототип, составленный на первом этапе или же, если мы сами, глядя на дизайн-макет, на бумаге схематично разрисовали себе все блоки страницы.

При написании разметки мы так же сразу можем прописывать элементам классы и идентификаторы.

В проекте во всём должен быть порядок: от структуры проекта до имен классов, разметки и написания кода. Если при разметке важно следить за типом информации и размещением её в соответствующих блоках, то при именовании классов и идентификаторов важно соблюдать здравый смысл. Классы должны давать абстрактное понятие о блоке, к которому они относятся, чтобы код было легче читать, а затем и писать стили. В принципе здесь не должно быть ничего сложного, если мы размечаем меню, то логично содержащему блоку дать класс `.nav` или `.navigation`, если это блок с текстом, то можно дать ему класс `.block-text` и т.д.

Написание CSS

Правила именования классов подводят нас к следующему этапу. Когда написана html структура проекта, определены классы можно переходить к написанию CSS стилей и нарезке макета.

Стоит упомянуть о 2-х CSS файлах-дополнениях: `normalize.css` и `reset.css`.

Изначально в проектах повсеместно использовался `reset.css`, написанный Эриком Мейером. Цель данного свода правил – сбросить стили браузера, которые он по умолчанию использует для отображения элементов разметки. Таким образом, при использовании `reset.css` нам не нужно переписывать стили браузера, по сути мы работаем с «чистым листом» и можем сосредоточиться на написании собственных стилей с нуля.

`Normalize.css` наоборот, не сбрасывает все стили «в ноль», а нормализует их, приводит отображение стилей проекта к более-менее однообразию во всех современных браузерах.

У обоих сводов правил есть свои плюсы и минусы, на данный момент популярен `normalize.css`. Популярность данного свода правил обусловлена ещё и тем, что вам не приходится при разработке заново прописывать основные свойства для базовых элементов страниц, а лишь модифицировать их по необходимости.

Так же стоит упомянуть о, так называемом, «быстром сбросе», когда мы абсолютно для всех элементов страницы с помощью селектора `*` прописываем свойства `{margin: 0; padding: 0}`. Таким образом, мы сбрасываем все внутренние и внешние отступы для всех элементов страницы. Так же стоит отметить, что все размеры и отступы берутся непосредственно из дизайн-макета. Для получения данных значений нам необходимо использовать инструмент «линейка» и направляющие, а затем переносить полученные значения в код. Если мы работаем с фиксированным макетом, то значения переносятся в пикселях как есть, если же у нас «резиновый», то значения нужно переводить в проценты.

Основная формула – это ширина элемента, делённая на ширину контекста. Например, если блок, содержащий текст и картинку имеет на макете ширину $400px$, а блок с текстом в нём должен иметь ширину $340px$, то в процентном выражении это будет $(340 / 400) * 100\%$, то есть 85% будет занимать блок с текстом (см.рис.7).

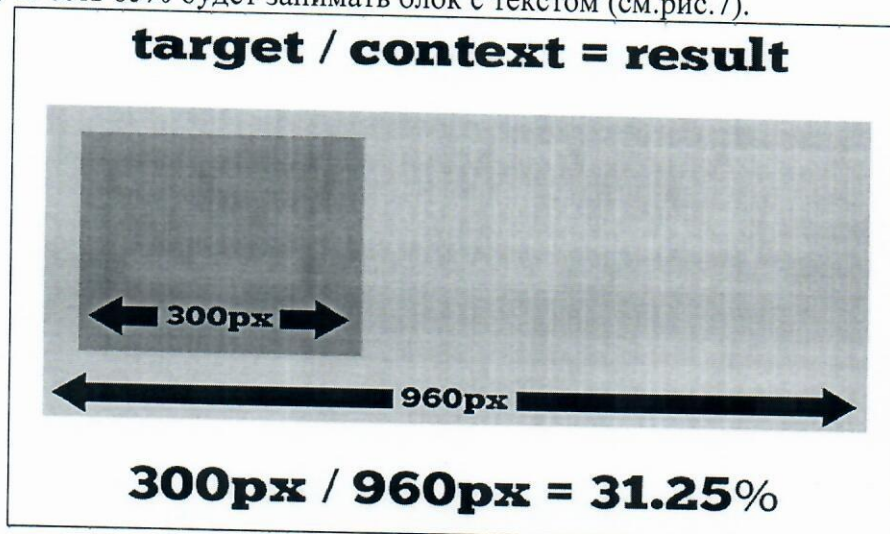


Рисунок 7. Размер экранного блока сайта

Возвращаясь к Mobile First, стоит сказать, что при работе в данной концепции написание стилей стоит начинать с мобильной версии, а затем с помощью медиа запросов добавлять правила, которые будут работать на других разрешениях. Зачастую, при расширении правил от мобильной версии к десктопной, приходится писать гораздо меньше кода, чем при работе наоборот: от десктопной версии к мобильной.

Написание JS. Итак, заключительным этапом является написание JS скриптов. Важно отметить, что не рекомендуется использовать JavaScript для стилизации, то есть не стоит для того или иного объекта DOM добавлять портянку CSS свойств с помощью JavaScript, чтобы выделить его состояние и пр. Рекомендуется использовать классы, то есть заранее в CSS определить классы-состояния и при манипуляции элементами просто добавлять или убирать соответствующие классы.

Не рекомендуется решать задачи, которые явно относятся к уровню CSS с помощью JavaScript, пытаюсь как-то выровнять объект или добавить ему стили при наведении и пр.

Возвращаясь к теме Mobile First, нельзя не упомянуть о 2-х концепциях, которые коррелируют с данной техникой. Progressive Enhancement и Graceful Degradation, что переводится как прогрессивное улучшение и последовательное ухудшение. Данные принципы описывают 2 разных подхода к разработке: в первом случае мы разрабатываем наш сайт, пишем скрипты с учётом старых браузеров и систем или же определяем поведение при невозможности запуска скриптов, а затем постепенно улучшаем скрипт, вводя современные функции и методы, таким образом, мы получаем сайт, который будет одинаково хорошо работать в старых и новых браузерах и средах.

При Graceful Degradation подход противоположный – мы разрабатываем для современных браузеров и, лишь, потом начинаем вносить доработки и изменения с учётом старых версий.

После написания html, css и js для нашей страницы необходимо проверить всё ли сделано верно. Для этого можно использовать online средства.

Исходя из всего вышесказанного, мы можем сделать вывод о том, что разработка веб сайта достаточно сложный процесс, представляющий собой строго определенную последовательность действий.

2. Обзор технологий создания сайтов

Существует несколько основных способов создания сайтов. Рассмотрим их по порядку.

1. Создание сайта на " чистом " HTML. HTML является первым языком web-программирования. Именно на его основе построены многие команды на php, javascript и т.д. Таким образом, можно сказать, что он вечен. Если поисковые системы по ряду признаков могут закрыть доступ к какому-либо движку, то HTML неприкосновенен в этом плане.

При написании сайта на чистом HTML на странице нет ничего лишнего, она быстро загружается, хорошо доступна для поисковых роботов, легче оптимизируется. Однако такой сайт не будет динамическим, внесение любых изменений потребует обновлений файлов на сервере. Такая технология подойдет для создания сайтов-визиток и лендингов.

2. Создание сайта на какой-либо CMS. CMS - система управления контентом, иными словами - "движок ". Минусы: придётся изучать структуру и особенности конкретно выбранной CMS, некоторые CMS не бесплатны и требуют активации лицензии на каждый домен, что может быть весьма накладно. Плюсы: даже начинающий вебмастер может создать динамический сайт без знаний языков программирования; в интернете для популярных CMS можно скачать бесплатные качественные шаблоны. Некоторые опытные вебмастера создают собственные системы управления контентом, удовлетворяющие конкретным условиям и задачам, поставленным разработчиком.

На сегодняшний день CMS WordPress — одна из наиболее популярных систем для создания и ведения блогов. Возможность публикации с помощью сторонних программ и сервисов; моментальная публикация; простота установки и настройки; поддержка веб-стандартов XHTML, CSS.

WordPress позволяет создавать сайты различного типа и функционала - информационные, сайты-портфолио, новостные сайты и т. п. Таким образом, если необходимо быстро создать свой блог, либо, что практически то же самое, вполне функциональный сайт информационного типа, на который периодически будут добавляться новые статьи и пользователям возможно будет их комментировать, то WordPress - вполне качественный и стоящий выбор.

WordPress хорошо русифицирован, для него написано множество дополнительных модулей, плагинов и сделано множество самых разнообразных шаблонов. Минусы WordPress типичны для популярных CMS - не слишком быстрая работа сайта, возможность сбоев при высокой посещаемости и периодическое обнаружение тех или иных дыр в скрипте.

Joomla ещё одна популярная CMS, на ней также весьма часто делают сайты. Несколько сложнее в освоении, чем WordPress, но имеет и большую сферу применения. Для Joomla разработано огромное число модулей, включая форумы, чаты, блоги, интернет-магазины и т. д., поэтому на ней можно смело пытаться делать сложный многофункциональный сайт.

«1С-Битрикс: Управление сайтом» — профессиональная система для создания и управления интернет-проектами: корпоративными сайтами; интернет-магазинами; информационными порталами; интернет-сообществами; социальными сетями и другими сайтами. Особенностью современных версий Битрикса является мощный визуальный HTML-редактор, позволяющий размещать на странице как обычную HTML информацию, PHP код, так и различные динамические компоненты, работу которых обеспечивает CMS.

3. Самые качественные ресурсы создаются, написанием собственного уникального кода и шаблона. Код пишется на одном из языков программирования и оптимизирован под конкретные задачи, что снижает нагрузку на сервер, а это очень важно для серьезных высокопосещаемых ресурсов. Часто применяются базы данных. Шаблон создается отдельно дизайнером или группой дизайнеров и интегрируется с кодом. Индивидуальный код обеспечивает высокую безопасность, так как никто, кроме разработчиков его не знает - сервер передает посетителю только чистый HTML - код.

4. Использование различных конструкторов. Это самый простой способ создания сайтов. В настоящее время существует огромный выбор платформ для создания сайтов с разнообразным функционалом. К недостаткам следует отнести: доменное имя не выше третьего уровня; схожий с другими сайтами дизайн.

WIX – это уникальная международная облачная платформа, которая позволяет создавать сайты любого уровня сложности в режиме онлайн.

Его главное преимущество заключается в том, что большинство услуг, действительно, бесплатные, а дизайн сайтов – привлекательный, современный и яркий. Доступны функции интернет-магазина, добавление картинок HTML-кода, музыки, текстовый редактор и многое другое. Изюминка проекта состоит в уникальном компоновании контента в режиме «Drag and Drop». То есть, пользователю необходимо лишь собрать «пазл» из необходимых компонентов в окне конструктора. Можно перетаскивать компьютерной мышкой любые составляющие, ограничений практически нет.

Относительно молодой конструктор сайтов SETUP в последнее время набирает популярность.

Дизайн сайта неумолимо напоминает нам WIX, как, впрочем, и функционал, что можно скорее отнести к преимуществам. Что касается функционала, то сервис способен удивить многих, так как предлагает на выбор огромное количество привлекательных шаблонов для будущих сайтов, причем, они для комфорта разбиты на тематические категории, например, бизнес, образование и так далее. Плюс ко всему, вы всегда можете заказать индивидуальную разработку, если вам не понравилось то, что предложено по умолчанию. Немаловажная деталь – всегда можно «прикрутить» к своему сайту доменное имя, причем сам сервис предлагает получить его на 100% бесплатно. Как это возможно? Для этого, пользователю придется, как минимум, создать 5 страниц сайта и заполнить их качественным и уникальным контентом.

Еще один известный и очень востребованный конструктор это, безусловно, UCOZ. Он славится своей общедоступностью и высоким качеством создаваемых сайтов. Система имеет довольно простую панель управления. С ее помощью можно будет легко добавлять, удалять и редактировать любой информационный контент. Есть настройки доступа для пользователей и можно будет создать любой дизайн страницы интернет-ресурса. Сервис характеризуется огромным количеством модулей. С их помощью можно будет осуществить автоматическую оптимизацию. На основе системы можно создать как довольно простой сайт, скажем, «визитку», так и интернет-портал, который имеет сложную структуру и обладает большей функциональностью.

Таким образом, в настоящее время практически любой пользователь может создать свой сайт, исходя предназначения интернет-ресурса и собственных навыков.

3. Разработка сайта

3.1. Разработка структуры сайта

Есть несколько простых правил, позволяющих интерфейсу сайта быть понятным посетителю.

1. На сайте не должно быть бесполезной информации, шрифт должен легко читаться. Графические элементы должны быть четкими, выразительными и быстро загружаться. На сайте компании недопустимо использовать анимацию и звук, которые долго загружаются и отвлекают внимание покупателя.

Человеческий глаз просматривает страницы сайта сверху вниз. Наибольшее внимание сосредотачивается на верхней левой части страницы. Поэтому в верхней части страницы сайта, как правило, размещается наиболее важная информация: название фирмы, логотип, само название сайта и т.д.

Чем проще выполнена верхняя часть страницы, тем легче запомнить название сайта и саму фирму.

2. Посетитель сайта должен без труда находить интересующую его информацию и иметь возможность получить исчерпывающую информацию. На витрине желательно разместить данные о деятельности фирмы.

2. Аккуратная работа с цветом. Правильно примененный цвет может, например, передавать тонкие различия между однородными элементами. Неправильно примененный цвет может мешать работать с программой.

Руководствуясь данными принципами разработки интерфейса, было решено сделать ставку на простоту и информативность, что бы пользователь, попадая на сайт, мог получать четкую информацию о товаре и предстоящих релизах.

В данной методической разработке мы разрабатываем сайт-визитку для рекламного агентства в г. Усмань. В ходе работы разрабатываем логотип компании. Цветовое решение выбрано ярких оттенков для привлечения большего внимания к данному ресурсу и созданию позитивного настроения.

Следуя общим принципам разработки дизайна сайтов, логотип компании мы разместили в верхней части. Ниже мы поместили меню сайта, в котором возможен переход по страницам сайта. Следующим блоком будет новостная лента и основной контент. В нижней части продублируем контакты.

Разработав макет сайта можно переходить к технической части – написанию сайта на языке разметки HTML.

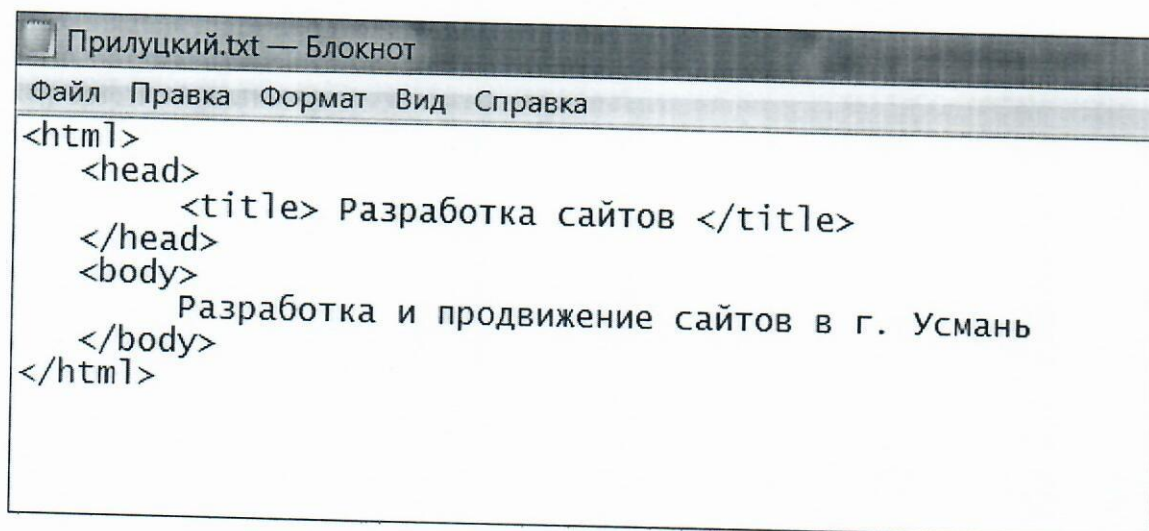
3.2. Верстка интерактивного веб-сайта

Запускаем программу.

Для начала потребуется прописать структуру HTML документа, которая имеет следующий вид:

`<html></html>` — теги, определяющие начало и конец документа;
`<head></head>` — теги, ответственные за заголовок данной страницы;
`<title></title>` — теги, прописывающие название сайта;
`<body></body>` — в этом теге прописывается код сайта.

Пример структуры документа сайта на html (смотри **Рисунок 8**).



The image shows a screenshot of a Notepad window titled "Прилуцкий.txt — Блокнот". The menu bar includes "Файл", "Правка", "Формат", "Вид", and "Справка". The text area contains the following HTML code:

```
<html>
  <head>
    <title> Разработка сайтов </title>
  </head>
  <body>
    Разработка и продвижение сайтов в г. Усмань
  </body>
</html>
```

Рисунок 8. Структура документа сайта на html

В первой строчке документа прописана версия языка html.
 Пропишем данный код в Блокноте и сохраним в формате .html.

Затем откроем в любом браузере и если все верно — откроется страница с названием «Разработка сайтов» и заголовком в верхней части «Разработка и продвижение сайтов».

Следующим шагом добавим нашей веб-странице цвета. Для этого добавим в наш код, в тег `<body>` следующие атрибуты - `bgcolor="blue" text="yellow"`. Здесь `bgcolor` - атрибут, отвечающий за цвет фона страницы, а `blue` - его значение. Атрибут `text` задает цвет текста документа, его значение `yellow` - желтый.

В HTML существуют два способа задания цвета: по имени и указанием шестнадцатеричного кода цвета.

С именными цветами все просто, смотрим в соответствующую таблицу html цветов, выбираем понравившийся и пишем его имя в значение атрибута.

Но гораздо больший выбор предоставляет второй способ. Здесь мы можем выбирать из миллиона цветов, указав его шестнадцатеричный код. Этот код представляет собой 6 цифр и начинается с символа "#". Не будем вдаваться в подробности как формируется код цвета, укажем лишь на то, что получить его можно, например, в программе Photoshop.

Итак, поменяем цвет в атрибуте `bgcolor` на нежно-голубой, а цвет текста - на менее яркий (см. **Рисунок 9**).

Обратим внимание, что атрибуты, указанные в теге `<body>` распространяют свое действие на весь документ, т.е. весь введенный текст будет теперь такого цвета, что не очень удобно. Поэтому лучше убрать атрибут `text` из тега `<body>`, тогда цвет текста будет установлен по умолчанию, т.е. черный. А задавать цвет различных частей текста лучше в самом тексте.

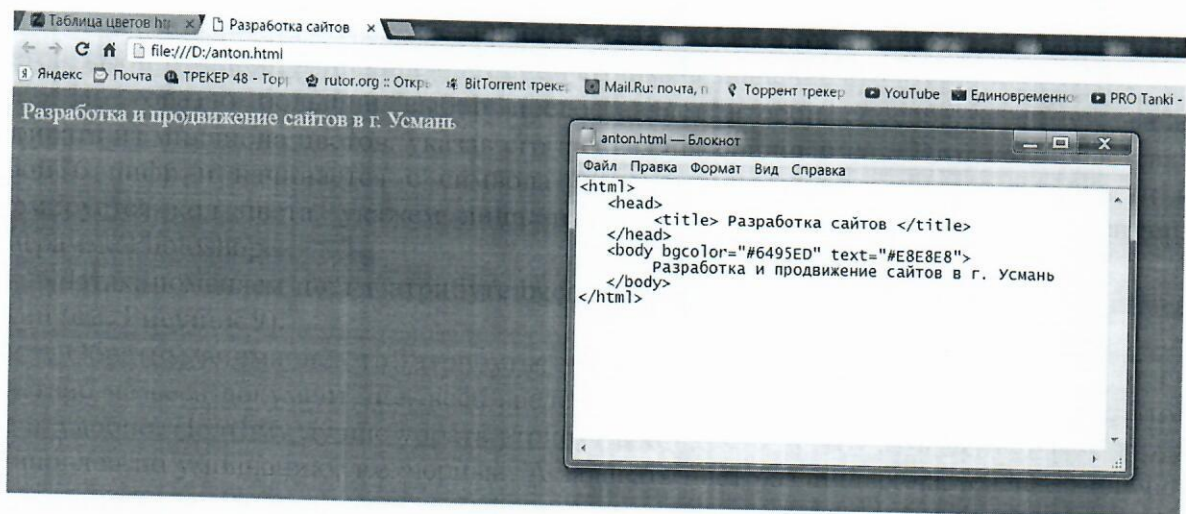


Рисунок 9. Создание цвета в атрибуте bgcolor

Разберем код подробнее:

Теги `<h1> </h1>`, `<h2> </h2>` и т.д. называются заголовками. HTML поддерживает 6 уровней заголовков от `h1` до `h6`. `h1` - самый главный и крупный, далее по убыванию значения и шрифта до `h6` - самого маленького и по значению, и по размеру шрифта.

Теги ` ` управляют шрифтом. Для этого используются атрибуты:

- `color` - задает цвет текста,
- `size` - задает размер шрифта,

- `face` - задает гарнитуру шрифта. В принципе, вы можете указывать здесь любой понравившийся вам шрифт, но учтите, что на компьютере пользователя такого шрифта может и не оказаться. Поэтому рекомендуется использовать стандартные: Times New Roman, Arial, Verdana и Tahoma.

страничка состоит из 5 строк и 2 столбцов. Каждый элемент нашей страницы будет располагаться в своей ячейке таблицы.

Теги `<table>` `</table>` означают начало и конец таблицы, а теги `<tr>` `</tr>` - строки в ней. У нас 5 строк, а потому и 5 пары тегов строк.

Далее у нас 2 столбца, за обозначение столбцов отвечают теги `<td>` `</td>`, но два их только в четвертой строке, в эту строку мы и поместим две пары тегов столбцов. А в остальных строках столбец один, как бы включающий в себя два. На такой случай у тега `<td>` есть атрибут `colspan`, объединяющий столько столбцов, сколько необходимо. В нашем случае их 2, так и напишем `colspan="2"`. Теперь наша страница должна выглядеть так.

```
<html>

  <head>

    <title> Разработка сайтов </title>

  </head>

  <body>

    <table>

      <tr><td colspan="2"> Здесь будет шапка сайта</td></tr>

      <tr><td colspan="2"> Здесь будет меню сайта</td></tr>

      <tr><td colspan="2"> Здесь будет шапка блока новостей</td></tr>

      <tr><td>Здесь - блок новостей </td>
        <td>Здесь - контент</td>
      </tr>

      <tr><td colspan="2">Здесь - низ сайта</td></tr>

    </table>

  </body>

</html>
```

Если сейчас посмотреть на эту страничку в браузере, то увидим, что она очень маленькая. Это связано с тем, что у таблицы нет размера, она меняется в зависимости от содержимого, что не очень удобно. Для этого зададим размеры. За это отвечают два атрибута: `width` - за ширину, `height` - за высоту.

Наша картинка шапки имеет размеры 1000 X 170, их мы и зададим для соответствующей ячейки. Строка меню должна иметь такую же ширину, а высоту - как у соответствующей картинки, т.е. 38. Ширина шапки блока новостей равна также 1000, а высота - 38. Ширина блока новостей и контента должны в сумме давать те же 1000, а высоту зададим 320. Строка низа сайта аналогична строке с меню, т.е. 1000 X 380.

Теперь можно вставить вместо слов "Здесь будет шапка" саму картинку шапки и сделать так для всех остальных ячеек, но есть один нюанс. В остальных блоках мы собираемся что-то писать поверх картинок, т.е. наши картинки должны быть в них фоном.

Для этого у столбцов существует атрибут `background`, в качестве значения которого мы и укажем соответствующие картинки.

Наши картинки лежат в папке `images`. Поэтому путь к ним будет выглядеть так: `background="images/имя картинки"`. Если у вас другая структура хранения страниц и картинок, то вам следует указать ваш путь к картинке.

Наконец, у блока новостей и контента нет фоновых картинок, поэтому в них используем уже знакомый тег `bgcolor`, где укажем номер цвета.

В результате проделанной работы у нас получился вот такой код для главной страницы:

```
<html>
<head>
<title> Разработка сайтов </title>
</head>
<body>
<table>
<tr>
<td colspan="2" width="1000" height="170" background="images/new.jpg">
</td>
</tr>
<tr bgcolor= "#76EE00">
<td colspan="2" width="1000" height="38"
background="images/menu.jpg">
<table border="0" cellpadding="5" cellspacing="0">
<tr><td>
<font size="6">
<font color="#363636" size="6"><a href="index.html">
главная </a> </font> ||
<font color="#363636" size="6"> <a href="portfolio.html">
портфолио </a></font> ||
<font color="#363636" size="6"> <a href="price.html">
прайс-лист </a></font> ||
<font color="#363636" size="6"><a href="kontakt.html">
контакты </a>
</font>
</td></tr>
</table>
```


</td>

</tr>

<tr bgcolor="#FFB6C1"> <td colspan="2"width="780" height="33">

Новости

</td> </tr>

<tr bgcolor=" #FFB6C1"> <td width="200" height="320">

Успей сэкономить!

Только в июне действует скидка на создание сайта 50%

Разработка адаптивных сайтов

В связи с заявлением компании Google, о повышении позиций для сайтов,

адаптированных под мобильные устройства в поисковой выдаче, мы приняли решение о дальнейшем создании для клиентов сайтов обязательно адаптированных под мобильные устройства.

Адаптация входит в стоимость работ по созданию сайта.

Дизайн сообществ.

Создание и оформление сообществ в трех социальных

сетях из четырех на выбор

</td>

<td bgcolor=" #FFEC8B" width="580" height="320">

<i>

Современный коммерческий сайт – это не только визитная карточка компании в сети, но также и мощный инструмент продаж.

Грамотно созданный сайт является хорошим инструментом развития Вашего бизнеса и приносит реальную прибыль.

</i>

Что мы можем вам предложить?

<i>

- Дизайн

</i>

Уникальный, яркий и при этом по-настоящему полезный и функциональный дизайн.

Никаких шаблонов, модных веяний и стереотипов — это признаки костного мышления и неспособности создавать собственные произведения веб-искусства!

Только оригинальные авторские идеи, индивидуальная отрисовка и поразительный результат!

<i>

- Верстка

</i>

Наша верстка построена на новейших технологиях HTML5, CSS3

<i>

- Раскрутка и продвижение сайта

</i>

Продвижение сайтов дает Вашему бизнесу поток потенциальных покупателей.

Когда люди видят результаты поиска, они и не подозревают, что это реклама.

Продвижение сайтов позволяет привлечь заинтересованную, лояльную аудиторию — людей, готовых стать Вашими клиентами.

 </td> </tr>

<tr bgcolor= "#76EE00"> <td colspan="2"width="780" height="38">

<i>

All Inclusive

</i>

<i>

Липецкая обл., г. Усмань, ул. Ленина 62, офис 5

</i>

<i>

e-mail: komserviseplus@mail.ru

</i>

<i>

Наши телефоны

+7 951 308 33 02, +7 920 501 01 67

</i>

</td> </tr>

<table>

</body>

</html>

Аналогично создаем страницы price.html, contact.html, portfolio.html.

Итак, у нас есть четыре html-страницы, которые нужно связать между собой. В HTML для этого используются теги <a> . Все, что пользователь поместит внутри этих тегов станет ссылкой. Это значит, что после нажатия на то, что вы указали в тегах <a> произойдет переход на страницу, которая указана в атрибуте href.

Рассмотрим наш блок меню:

```
<tr>
  <td colspan="2" width="1000" height="38"
    background="images/menu.jpg">
    <tr><td>
      <font color="#FFFFFF" size="5">
        главная || портфолио || прайс-лист || контакты
      </font>
    </td></tr>
  </table>
</td>
</tr>
```

Нам необходимо, чтобы при нажатии на слово "главная" открывалась страница index.html, при клике по "портфолио" - html-страница portfolio.html, а при клике по "контакты" - html-страница contact.html. Внесем соответствующие изменения в эту часть кода:

```
<tr><td>
  <font size="6">
    <font color="#363636" size="6"><a href="index.html">
      главная </a> </font> ||
    <font color="#363636" size="6"> <a href="portfolio.html">
      портфолио </a></font> ||
    <font color="#363636" size="6"> <a href="price.html">
      прайс-лист </a></font> ||
    <font color="#363636" size="6"><a href="kontakt.html">
      контакты </a>
    </font>
  </td></tr>
```

Здесь следует подробнее остановиться на способах задания адреса html-страницы в атрибуте href. Все наши страницы находятся в одной папке, т.е. имеют один уровень. Поэтому мы просто указали имя html-страницы.

Если же html-страница будет находиться в другой папке, то необходимо будет указать путь к ней от данной html-страницы. Например, если в нашей папке site лежат страницы index.html и price.html, а страницу kontakt.html мы поместили бы в папку kon, то

указывая путь со страницы index.html на страницу kontakt.html, мы написали бы следующее: ``.

Если необходимо сделать ссылку на html-страницу, которая не лежит на вашем сайте, то придется указать ее абсолютный адрес, например,

``

Обобщая все вышесказанное, мы можем сделать вывод о том, что для разработку сайта следует начинать с макета. Затем необходимо создать web-страницу в виде файла определенного формата. Если быть более точным – это должен быть файл index с расширением html или htm. Далее, опираясь на базовые знания в области HTML, наполняем сайт информацией.

4. Продвижение сайта в поисковых системах

Для того чтобы, сайт был доступен в поисковых системах его необходимо разместить на хостинге, т.е. на сервере который предоставляет услуги по размещению и хранению ваших файлов. Хостинги бывают платные и бесплатные.

Как правило, хостеры за некоторую ежемесячную плату предоставляют часть своего дискового пространства, куда пользователь размещает свои файлы. Они поддерживают разные технологии и обеспечивают присутствие сайта в интернете 24 часа в сутки.

Существует много сервисов по бесплатному размещению сайтов. Делают они это за счет средств, получаемых от рекламодателей, разместивших рекламу на ваших страницах.

Залогом успеха продвижения абсолютно любого сайта является его правильная внутренняя оптимизация и качество материалов сайта. Если быть кратким, то сайт должен соответствовать следующим характеристикам:

- Качественный, уникальный, а главное полезный и интересный контент.
- Правильная поисковая оптимизация страниц сайта.
- Правильная структура сайта и понятная навигация.
- Правильная внутренняя перелинковка страниц.
- Валидный код, легкость и хороший дизайн.

Качественный, уникальный контент – содержимое сайта должно быть уникальным. Уникальные тексты очень ценятся поисковыми системами, поэтому залогом любого успешного продвижения является именно уникальность информации продвигаемого проекта. Исключений тут практически нет, разве что разнообразные интернет магазины, где информация часто дублируется. Перед публикацией статей, рекомендуется проверять материал на уникальность. Для этого лучше всего воспользоваться специальной программой ЕТХТ-антиплагиат. С ее помощью можно очень быстро проверить уникальность текста и при необходимости исправить недочеты.

Правильная поисковая оптимизация страниц – все страницы сайта должны быть оптимизированы под определенные поисковые запросы. Здесь необходимо придерживаться следующих правил:

- Правильный Title – заголовок страницы должен быть привлекательным и содержать ключевые слова, по которым продвигается данная конкретная страница. Лучше использовать точные вхождения запросов. В этом случае релевантность страницы сильно повышается.

- Использование заголовков – в тексте страниц необходимо использовать заголовки первого – шестого уровня с применением ключевых слов. Самый большой приоритет у заголовка первого уровня, поэтому его нужно оптимизировать в первую очередь. Можно использовать как прямые вхождения ключевых слов, так и морфологические. При этом заголовок первого уровня должен встречаться на странице не более одного раза.

- Использование ключевых слов в тексте страниц – на странице в обязательном порядке должны присутствовать ключевые слова с разными типами вхождений: точными,

прямыми, морфологическими и разнообразными разбавленными вхождениями. Рекомендую использовать каждый тип вхождений, чтобы текст был читабельным и хорошо оптимизированным.

– Выделение ключевых моментов тегами STRONG – позволяет акцентировать внимание на нужных моментах, а также повысить вес выделенных ключевых слов.

– Информативность страниц – продвигаемые страницы должны быть информативными. Обычно, в среднем рекомендуемый объем статьи составляет порядка 300 слов или более. При этом не стоит забывать про графику и другие элементы, которые дополняют информацию на странице.

Сайт должен иметь правильную структуру и понятную навигацию. Вся информация должна быть упорядочена, а не разбросана в произвольном порядке. Одним из примеров правильной структуры будет, например, следующий вариант (смотри Рисунок 12.).

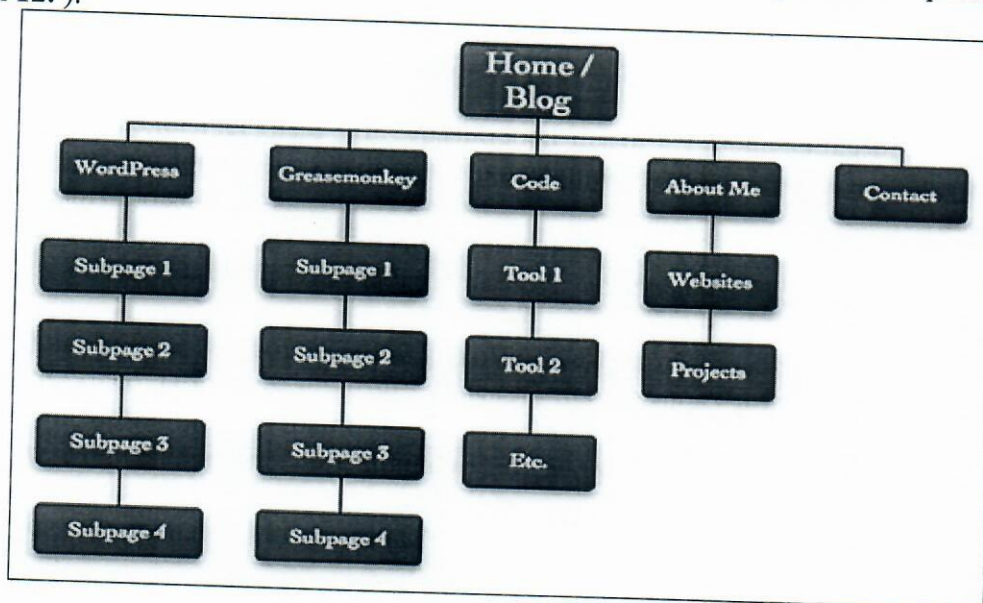


Рисунок 12. Правильная структура сайта

В этом самом простом примере сайт имеет древовидную структуру. На нем есть разделы, которые содержат подразделы, категории, отдельные материалы или любые другие подстраницы. Также с любой страницы сайта есть возможность перейти на страницы контактов, информации о сайте и т.д. Данный вариант является небольшим эталоном правильной структуры сайта и применяется на большинстве web-проектов.

Правильная перелинковка – этот фактор является одним из самых важных в самостоятельном продвижении сайта.

Суть правильной перелинковки состоит в том, чтобы ссылаться с нужными анкерами на тематические страницы и тем самым продвигать их за счет внутренних ссылок. В результате мы получаем разветвленную сеть из ссылок, которая больше походит на паутину. За счет этого мы не только распределяем вес по всем страницам сайта или же смещаем его на нужные нам страницы, но и продвигаем страницы по нужным нам поисковым запросам.

Валидный код, легкость и хороший дизайн – сайт должен иметь хороший внешний вид, быстро грузиться и не содержать ошибок.

Особое необходимо обратить на скорость загрузки сайта, так как с ней у многих бывают проблемы. Чем скорость загрузки меньше, тем больше будет процент отказов, так как пользователи попросту могут не дождаться полной загрузки страницы и уйти. В результате этого вполне можно ожидать скорого падения позиций в поисковых системах.

Исходя из всего вышесказанного, стоит сделать вывод, что больше всего на успех продвижения влияет качество сайта и его внутренняя оптимизация.

Заключение

Web-страница – это лицо той фирмы, того учреждения, человека, который разместил ее в WWW. Именно поэтому сегодня Web-дизайну уделяется такое огромное внимание, ибо от него напрямую зависит популярность того или иного информационного ресурса сети.

Информационный сайт помогает значительно снизить финансовые затраты на рекламу, являясь отличным маркетинговым инструментом. При этом на нем присутствует информация, которая поможет потребителям удовлетворить их требования на необходимые ими запросы. Информационный сайт послужит хорошим инструментом для предприятия.

Результатом методической разработки является создание WEB-сайта для агентства. Разработанный WEB-сайт позволяет узнать об оказываемых услугах и стоимости на них. Для продвижения WEB-сайта нужно зарегистрировать его на поисковых серверах, что позволит пользователям заходить на него по ключевым словам. Сайт должен быть «живым» т.е. он должен обновляться хотя бы 1 раз в неделю.

Подводя итог всему выше сказанному, хочется отметить, что HTML стал тем форматом передачи данных, который наиболее полно и качественно удовлетворяет запросы современного общества. Несомненным фактом является и то, что будущее именно за HTML.

Таким образом, мы изучили возможности языка HTML для создания Web-страниц, узнали, какие HTML и графические редакторы лучше использовать в Web-дизайне, каковы преимущества и недостатки тех или иных программных пакетов.

В данной методической разработке рассмотрены актуальные вопросы разработки и создания современного Web-сайта.

При этом нами были решены следующие частные задачи:

- ознакомление с современными Интернет-технологиями и их использованием в настоящей разработке;
- изучение программного инструментария, применяемого для разработки и создания Web-сайтов;
- выявление и учет методов и способов представления на Web-страницах различных видов информации, не препятствующих их доступности;
- ознакомление с основными правилами и рекомендациями по разработке и созданию Web-сайтов и неукоснительное следование им на практике;
- определение структуры Web-страниц;
- выбор стратегии разработки и создания Web-сайта.

Таким образом, материалы методической разработки показывают, что поставленные в ней задачи полностью решены и помогут педагогу дополнительного образования создать сайт-визитку, или помогут в обучении учащихся при создании учащимися сайта-визитки.

Список использованной литературы и источников

1. Лекции по разработке сайта: [Электронный ресурс] //Лекции.орг.
URL: <http://lektsii.org/4-1517.html>. (Дата обращения: 18.10.2022).