
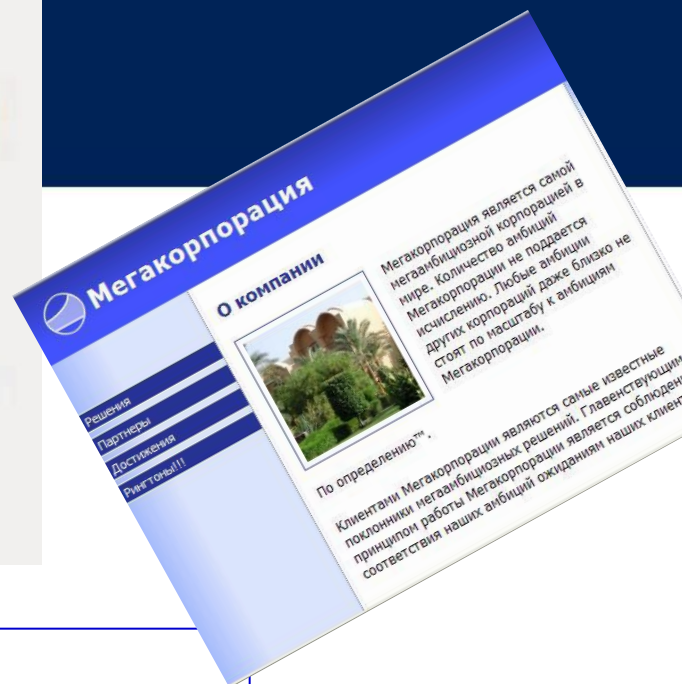


# Разработка электронного портала





## Интернет представительства и их структура

- Применительно к WEB, **портал** - это крупный сайт с большим количеством разнообразной информации, охватывающей все (или почти все) области тематики портала.
- Портал представляет собой централизованное хранилище обновляемой информации, к которой более или менее регулярно обращается большое число пользователей.

# Еще определение....

- Портал – portal – сайт, организованный как системное многоуровневое объединение разных ресурсов и сервис, который
- дает пользователю четкую информацию, осуществляет мгновенный доступ к таким сервисам, как
- поисковые системы,
- электронный шоппинг,
- бесплатная электронная почта,
- торговая реклама,
- и т.д.

# РЭП-

- Дисциплина посвящена всем аспектам разработки клиентской части веб-проектов: дизайн, юзабилити, вёрстка, JavaScript, веб-медиа и так далее.

- В современной социокультурной среде, веб-инфраструктура является одним из наиболее важных источников информации, а также средств общения.
- При этом развитие сети интернет, произошедшее за небольшое даже по современным меркам время, привело к бурной эволюции самих информационных ресурсов — узлов сети.
- 
- Вследствие развития пользовательского опыта и повышения важности веб-среды для общества в целом, инфраструктура сетевых ресурсов сети интернет находится в постоянном развитии.

- Интернет находится в постоянной трансформации
- по причине развития самой инфраструктуры сети, обеспечивающей все большие скорости и объемы обмена информацией увеличивающей проникновение сети вовсе более бедные и отдаленные уголки планеты;
- по причине появления инновационных устройств создания и потребления контента.

- Основные направления создания и потребления контента в сети Интернет подвержены **Моде**.
- К числу основных контентных направлений можно отнести:
  - новостные ресурсы,
  - ресурсы интернет-коммерции (различные магазины и торговые площадки — агрегаторы),
  - поисковые ресурсы,
  - онлайн-библиотеки,
  - игровые порталы,
  - научные порталы,
  - корпоративные информационные ресурсы.



# Модели ориентации порталов

- ориентированные на компании;
- ориентированные на пользователей;
- ориентированные на технологии; (применение всевозможных технологических решений)
- ориентированные на дизайнеров, его создающих; (демонстрация уровня самих дизайнеров)

# Примеры сайтов современной верстки

- <https://getbeagle.co/>
- <https://alfabank.ru/>
- [нет-коллекторам.рф](http://нет-коллекторам.рф)
- <http://vaalentin.github.io/2015/>
- <http://blog.guillaumecassuto.com/>
- <http://www.romainbriaux.fr/>
- <http://pirogov.ru/>

## Основные проблемы, объединяющие все веб-ресурсы

- Необходимость постоянного добавления и обновления контента.
- Проблема удобства использования информационных ресурсов.
- Проблемы жизненного цикла ресурса, от его появления на свет, через развитие, к расцвету и увеличению числа пользователей.

# ТИПИЧНЫЙ ЖИЗНЕННЫЙ ЦИКЛ ВЕБ-САЙТА:

- Формирование идеи веб-сайта — на этом этапе происходит разработка концепции ресурса, определение аудитории, формируются общие принципы контентного наполнения

- Разработка бета-версии (или прототипа веб-сайта).
- На этом этапе апробируются концепции и идеи первого этапа.
- Одной из наиболее важных задач данного этапа, является быстрое создание системы, выполняющей базовые и наиболее важные для разрабатываемого сайта функции.
- *Предполагаемая, на этом этапе, нагрузка на систему — невысока, инвестиции на данном этапе так же скромные, поэтому, как правило используются бесплатные системы разработки, системы хранения данных и т. п.*

- Продвижение созданного веб-сайта (рекламные усилия, продвижение через социальные сети и т. п.).
- Нарращивание функционала, устранение существующих проблем и ошибок.

- Успех веб-сайта, резкое повышение нагрузки.
- Как следствие, редизайн системы хранения данных и доступа к данным, с сохранением пользовательских интерфейсов (модернизация подсистем хранения данных и подсистемы реализации бизнес — логики)



- Модернизация визуальной подсистемы в соответствии с последними трендами.

- Разделение ресурса на три независимых функциональных части позволяет оптимальным образом разделять работы между различными группами разработчиков.
- Обычно разделяют разработчиков, занимающихся
  - пользовательскими интерфейсами (front side),
  - разработкой серверной части — бизнес-логикой,
  - хранением данных (server side).
- *В наиболее нагруженных проектах, так же отдельно выделяются группы разработчиков ответственные за поддержание и оптимизацию баз данных.*

# Современные проблемы визуального дизайна сайтов

- Поскольку на наш взгляд, сегодня, на первое место выходит не проблема создания “правильного” визуального дизайна веб-ресурса, а скорее проблема создания дизайна учитывающего новации пользовательского опыта доступа к веб-ресурсам с использованием множества новых устройств, таких как планшеты, смартфоны, смарт-телевизоры и другие устройства
- Таким образом, одно из ключевых требований к дизайну современных веб сайтов — размещение всей важной функциональности, в крайне ограниченном объеме, и максимально полная поддержка инновационных возможностей устройств.

# Типы структурных элементов

- Шапка страницы
- Навигация
- О компании
- Контакты
- Информация общего характера для всех посетителей
- Часть обычной информационной страницы
- Страницы с формами ввода
- Баннеры и информеры
- Закрытая часть сайта
  - Модули, реализующие основную деятельность:
    - Информационное взаимодействие
    - Выполнение бизнес действий

## Факторы, влияющие на дизайн

- Задачи, которые должен решать сайт
- Технологии которые необходимо использовать на сайте
- Люди, которые должны будут пользоваться сайтом

# Принципы определения посетителей

*Основная проблема создателей сайтов:*

- Команда разработчиков считает, что она и есть посетитель и делает как удобно ей;*
- Опыт разработчиков;
- Понимание окружающей среды;
- Стилль жизни и менталитет;
- Логика мышления;
- Стилль изложения информации

## Базовые требования к специалистам при создании сайтов

- Подготовка по дисциплинарным направлениям:
  - Графический дизайн;
  - Информационная или научная направленность;
  - Usability инжиниринг;
  - Маркетинг;
  - Компьютерное образование

Приступая к разработке сайта, частью которого является Интернет приложение, необходимо:

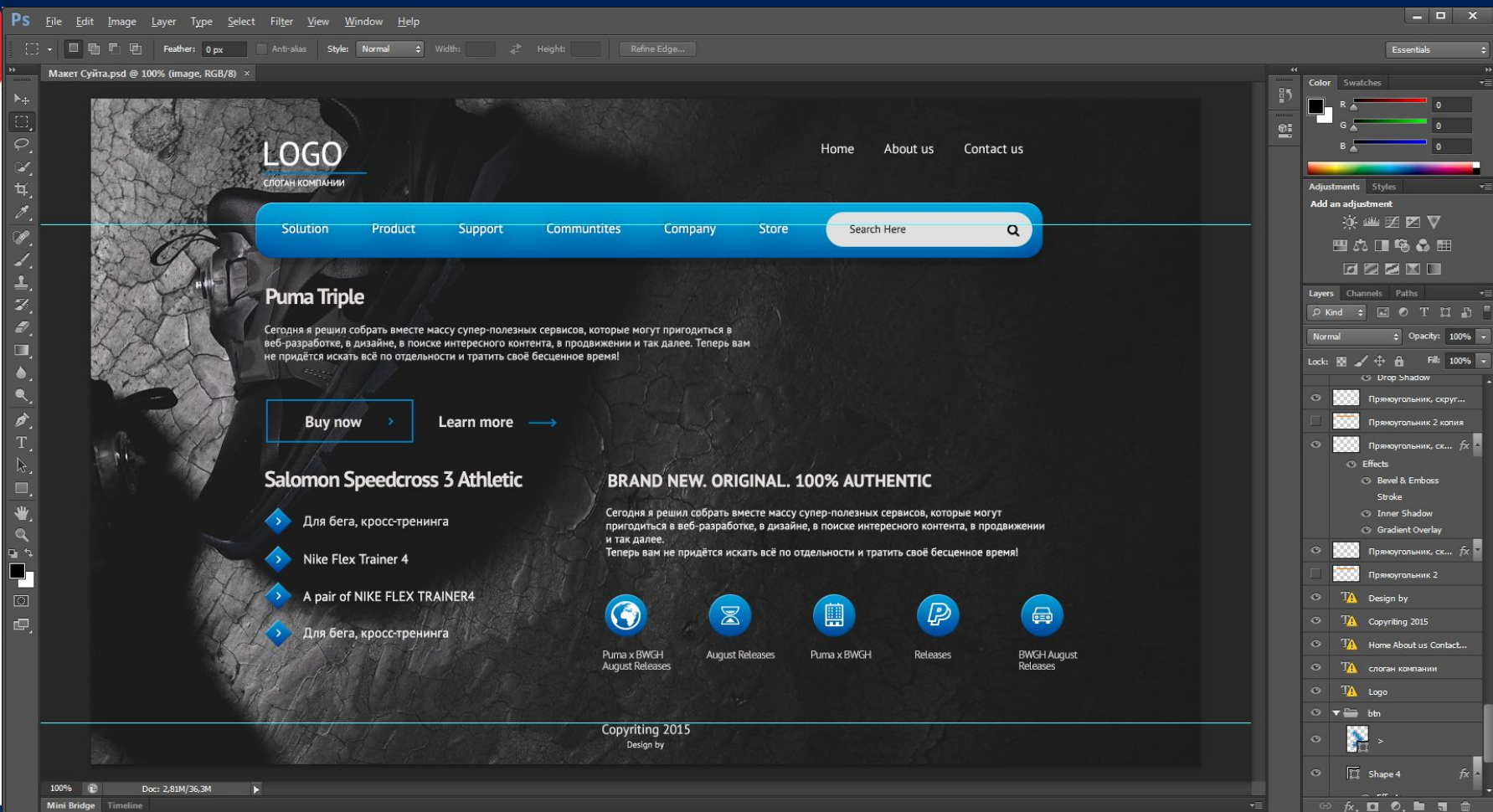
- четко представлять структуру навигации;
- четко представлять структуру информационного наполнения.

Это позволит определить структуру Интернет-приложения и используемые технологии .



- При этом важно отметить, что сегодня в наличии у веб-разработчиков имеются подходящие инструментальные средства, позволяющие реализовывать предъявляемые к веб-ресурсам требования.
- Например, язык HTML 5 — позволяет реализовать все вышеперечисленное и к тому же является гибкой и расширяемой спецификацией, что гарантирует добавление новых возможностей, появляющихся вследствие развития технологий в сфере пользовательских устройств.

- **Вёрстка** – это создание на основе нарисованных макетов (как правило PSD) структуры страницы с помощью HTML и CSS, корректно отображающейся в различных браузерах. Созданная структура, определяет дальнейшее отображение текстовой и графической информации на сайте.



Внешний вид и размещение

• HTML+CSS+JS

Создание объектов

событий

Изменение при наступлении

# Фреймворки

- Bootstrap
  - Foundation
    - WordPress

# Препроцессор —

- это компьютерная программа, принимающая данные на входе и выдающая данные, предназначенные для входа другой программы (например, компилятора).
- О данных на выходе препроцессора говорят, что они находятся в препроцессированной форме, пригодной для обработки последующими программами (компилятор).
- Результат и вид обработки зависят от вида препроцессора; так, некоторые препроцессоры могут только выполнить простую текстовую подстановку, другие способны по возможности сравниться с языками программирования.

# Предпроцессоры CSS

<http://zencoder.ru/css/css-sass-less-stylus/>

- [Sass](#)
- [LESS](#)
- [Stylus](#)

- Препроцессоры CSS были созданы с одной единственной целью - добавить стилевым таблицам CSS мощь и одновременную гибкость без нарушения кросс-браузерности. Все препроцессоры компилируют созданный с помощью их синтаксиса код в стандартный CSS-код, который понимает и использует любой браузер, каким бы древним он (*браузер*) не был.




## Преимущества использования предпроцессоров(Хабрахабр ):

- делают код CSS чистым и логичным
- позволяют создавать переменные и повторно использовать свойства CSS

# Сниппет

- - это небольшой блок информации о найденном документе того или иного сайта, который отображается в результатах поиска. Именно от привлекательности и релевантности сниппета зависит перейдет ли пользователь на сайт.

 [Печи для дома, бани и саун - ...цены, большой каталог печи для дома, бань](#)  
Покупайте **печи для** дома, бани и саун с удовольствием, доставляем :) Викимарт, +7 (495) 641-58-85. ... Назначение: Для дома, **Для дачи** Материал: Металл, Чугун Размеры (ДхШхВ): 450x370x900...  
[garden.wikimart.ru](#) > Каминь > **Печи для дома** ————— Навигационные цепочки

- Снippet состоит из заголовка, описания документа и дополнительной информации - url, телефон, регион, адрес организации, ссылки на страницы в соц.сетях.
- Для больших и хорошо структурированных сайтов, в снippetе могут присутствовать быстрые ссылки и навигационные цепочки.

- В сниппете может отображаться информация о пользователях социальных сетей — с фотографиями и личными данными, представленными их в профиле (например: образование, место работы, возраст и т. д.). Любые изменения в профиле, в том числе и смена аватара, отобразятся в выдаче Яндекса, только после того как система проиндексирует обновленные данные.

- На сегодняшнее время поведение пользователей оказывает весьма серьезное влияние на ранжирование документов в результатах поиска. Причем учитывается поведение не только на сайте, но и непосредственно в самой поисковой выдаче (SERP). Большое внимание поисковых систем уделяется кликовым факторам.
- Показателем кликабельности является **CTR**, который определяется как отношение числа кликов к числу показов. Чем больше кликов по вашему документу по отношению к числу его показов, тем больше ваш CTR. Чем больше ваш CTR, тем лучше ваш сайт выглядит “в глазах” поисковых систем.

# Корректировка использования тегов и их параметров

# Устаревшие теги

- **<applet>** — добавляет Java-апплет в документ. Вместо него следует использовать **<embed>** или **<object>**.
- **<frame>**, **<frameset>**, **<noframe>** — фреймы более не поддерживаются. Если они вам требуются, используйте другую версию HTML или **<iframe>** совместно со стилями.
- **<strike>** — для зачёркнутого текста применяется **<s>**, а для указания редакторской правки **<del>**.
- **<basefont>**, **<big>**, **<blink>**, **<center>**, **<font>**, **<marquee>**, **<multicol>**, **<nobr>**, **<spacer>**, **<tt>**, **<u>** — вместо этих тегов управляющих видом текста применяются стили.
- А тег **<small>** допустим

# Устаревшие, но поддерживаемые атрибуты

- Атрибут `http-equiv` тега **<meta>** для указания языка должен заменяться атрибутом `lang`.
- Было:
- **<meta http-equiv="content-language" content="ru" />**
- Стало:
- **<html lang="ru">**



- Атрибут `name` тега `<a>` должен заменяться атрибутом `id`.
- Если `name` присутствует, то должен содержать пустую строку или совпадать со значением `id`.
- Было:
- `<a name="p12"></a>`
- Стало:
- `<a id="p12"></a>`

- Атрибут `language` тега **`<script>`** должен быть **опущен**. Если он присутствует, значение совпадает с JavaScript или другим типом, но в таком случае язык скрипта следует указать через `type`.
- Было:
- **`<script type="text/javascript" language="JavaScript">`**
- Стало:
- **`<script>`**

- Атрибут `border` тега `<img>` не указывается, а толщина границы задаётся через стили.
- Было:
  - `<img border="1">`
- Стало:
  - `<img style="border: 1px solid black">`

# Устаревшие атрибуты

- Элемент **<body>**: `alink`, `bgcolor`, `link`, `marginbottom`, `marginheight`, `marginleft`, `marginright`, `margintop`, `marginwidth`, `text`, `vlink`.
- Элемент **<embed>**: `name`.
- Элемент **<html>**: `version`.
- Элемент **<link>**: `charset`, `methods`, `rev`, `target`, `urn`.
- Элемент **<script>**: `event`, `for`, `language`.
- Элемент **<table>**: `datapagesize`.
- Элемент **<td>** и **<th>**: `abbr`, `axis`.

# Использование двойных кавычек

- Значение берётся в двойные кавычки, обычно такая форма указывается для текста.
- `<input type="checkbox">` Использование одинарных кавычек
- Вместо двойных кавычек также допустимо писать одинарные.
- `<input type='checkbox'>`
- Значения атрибутов разделяются между собой пробелом, поэтому если у вас в качестве значения указывается предложение, обязательно берите его в одинарные или двойные кавычки.

# Значения без кавычек

- Значение пишется непосредственно после знака равно идущим вслед за именем атрибута. До и после знака равно можно вставлять любое количество пробелов или обойтись без них.
- `<img src=link.html alt=Картинка>`
- Здесь атрибутами являются `src` и `alt`, а после знака `=` идёт их значение без кавычек. Поскольку атрибуты разделяются между собой одним или несколькими пробелами, то при отсутствии кавычек легко допустить ошибку, когда браузер воспримет предлагаемое нами значение как атрибут.
- `<img src=link.html alt=Картинка в тексте>`
- Здесь значением атрибута `alt` будет слово «Картинка», остальные слова воспринимаются как неверные атрибуты.

# Кавычки внутри значений

- Внутри значений атрибутов не разрешается применять те же кавычки, в которых взято само значение. Но допустимо сочетать разные типы кавычек между собой. Если внутри текста необходимы одинарные кавычки или апостроф, то сам текст следует взять в двойные кавычки.
- `` Соответственно, текст содержащий внутри двойные кавычки надо взять в одинарные.
- `<img src='с.jpg' alt='Такой вид кладки называется "циклопическим" '>`
- Также вместо двойной кавычки в тексте можно использовать спецсимвол `&quot;`, а вместо апострофа — `&apos;`.

# Необязательные теги

- Если какой-то тег не указан, это не означает, что он не представлен вообще. Существуют определённые правила, позволяющие не писать некоторые теги



Ter	Условие
<html>	
</html>	
<head>	Если внутри имеются другие элементы.
</head>	
<body>	Если пустой, а также содержит что-то кроме пробела или комментария.
</body>	
</li>	Если после элемента следует <li> или он последний у родителя.
</p>	Если после элемента следует <address>, <article>, <aside>, <blockquote>, <dir>, <div>, <dl>, <fieldset>, <footer>, <form>, <h1>,..., <h6>, <header>, <hgroup>, <hr>, <menu>, <nav>, <ol>, <p>, <pre>, <section>, <table>, <ul>.
</tr>	Если после элемента следует <tr> или он последний у родителя.
</td>	Если после элемента следует <td> или <th> или он последний у родителя.

- Вызывающие проблемы у пользователей;
- Влияющие на SEO;
- Затрудняющие дальнейшую поддержку и масштабирование сайта и говорящие о низкой квалификации верстальщика.

# Проблемы у пользователей

- Доступность сайта.
- Доступность для клиентов:
  - с мобильных устройств;
  - с отключенным javascript'ом;
  - с устаревшими браузерами (IE5.x);
  - с низкой скоростью соединения;
  - с отключенной графикой;
  - с физическими ограничениями (в первую очередь это слепые пользователи).

# Шрифты в рх

```
body  
{font-family:Arial, Tahoma, Verdana, sans-serif; font-size:12px}
```

(пользователи IE не смогут увеличить размер шрифта)

```
надо:  
html  
  {font-size: 75%}  
body {font: normal 1em/1.3 Arial, sans-serif}
```

# Общий CSS для всех media

```
<link href="main.css" rel="stylesheet" type="text/css" />
```

# Почему плохо:

- сайт это в первую очередь документ. Это информация, которая может и должна быть представлена в виде: заголовок, текст, иллюстрация, а
- при печати страницы на принтере получаем не документ, а исковерканный скриншот сайта;
- встроенные браузеры мобильных телефонов (даже старых) вполне могут открывать не только WAP, но и обычные сайты, если они валидны, мало весят и сверстаны семантически, а не презентационно, а
- мобильные устройства не в состоянии отобразить большинство этих стилей и как следствие встроенные в них браузеры вообще не отобразят этот сайт;
- OperaMini попытается конвертировать его в доступный вид, но результат будет весьма далёк от идеала: внешний вид не ахти, много бесполезных на мобильном устройстве элементов и больший чем должен был бы быть трафик у пользователя.

# Как надо:

```
<link rel="stylesheet" href="css/main.css" type="text/css" media="screen,projection" />
```

```
<link rel="stylesheet" href="css/print.css" type="text/css" media="print" />
```

```
<link rel="stylesheet" href="css/handheld.css" type="text/css" media="handheld" />
```

# img вместо background

img вместо background

```
<a href="#"></a>
```

```
<a href="#">
```

```

```

```
</a>
```



# Антисемантика

```
<div class="menu">  
<a title="Kinderspiele" class="kinderspiele" href="#"></a>  
<a title="Artikel" class="artikel" href="#">  
</a>  
</div>
```

```
<ul class="menu"> <li>  
<a title="Kinderspiele" id="kinderspiele" href="#">  
Kinderspiele</a>  
</li>
```

```
<li><a title="Artikel" id="artikel" href="#"> Artikel</a></li> </ul>
```

```
<dl id="t-personal"> <dt>Настоящее  
Имя:</dt> <dd>Иван  
Копейкин</dd> <dt>Дата  
рождения:</dt> <dd>16 сентября  
1977г.</dd> </dl>
```

**Плохо:**

<p><strong class="black">

Предметом бухгалтерского учета</strong> в обобщенном виде выступает хозяйственная деятельность предприятия с точки зрения системы учета ресурсов и результатов финансовой и хозяйственной деятельности предприятия...</p>

<p> <dfn>Предметом бухгалтерского учета</dfn> в обобщенном виде выступает хозяйственная деятельность предприятия с точки зрения системы учета ресурсов и результатов финансовой и хозяйственной деятельности предприятия...</p>

`float: left` для всего

<div></div>

## Неуниверсальные глобальные стили базовых элементов

```
a,body,p,span,td {font-family: Tahoma,Verdana,Sans-serif;  
color: #3C5C92; font-size: 10px;} ul li {margin: 0; padding: 0;  
list-style-type: none}
```

- это грубое нарушение логики документа: например вы зададите стиль для блока – размер шрифта 12px, а текст в параграфах этого блока всё равно будет 10px. Вы зададите дополнительный стиль для параграфов, а ссылки всё равно будут 10px...
- где-то на новой странице контент-менеджер вставит текст со списками... он не увидит списков. Потому что верстальщик обнулil базовый стиль, и создал специальный класс для списков! Контент менеджеру теперь html учить и в код лезть чтобы список добавить?

- <http://blog.ircit.ru/post/vozmozhnosti-foundation-5-kotorykh-net-v-bootstrap/>

# Состав технического задания на сайт

- ТЗ на создание сайта должно состоять не менее чем из 20 разделов, определяющих требования к сайту:



# Разделы технического задания на сайт(1)

- цели и задачи сайта,
- целевая аудитория сайта,
- информационные разделы сайта,
- функциональные разделы сайта,
- взаимосвязи между разделами сайта,
- программная реализация сайта,

## Разделы технического задания на сайт(2)

- дизайн и видимая посетителю структура страниц сайта,
- иллюстрации,
- корректность программного кода и совместимость с браузерами,
- обеспечение «видимости» сайта поисковыми системами,
- рекомендации по составу страниц сайта,

## Разделы технического задания на сайт(3)

- обеспечение удобства для посетителей сайта,
- дополнительные требования к отдельным страницам сайта,
- сбор статистических данных о видимости и посещаемости сайта и его отдельных страниц,
- состав первой и последующих очередей разработки и публикации сайта,

## Разделы технического задания на сайт(4)

- продвижение и раскрутка сайта,
- сопровождение сайта, - развитие сайта,
- выбор имени и регистрация домена, выбор хостинга,
- порядок внесения изменений в ТЗ на сайт,
- порядок согласования и приемки результатов выполнения настоящего ТЗ в целом и отдельных его разделов.

# Этапы проектирования и реализации интернет-сайта

- **Стратегическое планирование**
  - Определение целей и задач
  - Проведение маркетинговых исследований
  - Определение целевого сегмента потребителей
  - Составление бизнес-плана
- **Тактическое планирование**
  - Разработка информационной структуры
  - Разработка навигационной структуры
  - Проектирование средств обновления и добавления информации
  - Разработка и утверждения графического дизайна сайта
- **Реализация**
  - Адаптация графики
  - Верстка
  - Программирование
  - Интеграция с информационной системой предприятия
  - Тестирование
  - Публикация в интернет

# Стратегическое планирование

## 1 Административная документация:

- учредительная (уставы, учредительные договоры и т.п.);
- организационная (организационная структура, положения о подразделениях и т.п.);
- распорядительная (приказы, должностные инструкции и т.п.);
- информационно-справочная (протоколы, акты, письма и т.п.);

## 2 Финансовая документация:

- инвестиционная (бизнес-планы, ТЭО ит.п.);
- отчетная (балансы, отчеты об убытках и т.п.);
- учетная (планы счетов, главная книга и т.п.);
- расчетно-платежная (чековые книжки, платежные поручения ит.п.).

## 3 Стратегическая документация:

- концептуальная (концепция проекта, перспективный план ит.п.);
- декларативная (политика, связанная с качеством, корпоративная культура ит.п.).

# Начальный этап проектирования сайта (концепции)

## **Бизнес-концепция сайта:**

- Какие задачи должен решать сайт?
- Кто является целевыми посетителями сайта?
- Каковы масштабы функционирования сайта;

## **Техническая концепция сайта:**

- разделы и стиль изложения;
- типы информации, которая будет на сайте, частота ее обновления, интерес к ней целевой аудитории;
- навигация (описаны текущие распространенные маршруты и то, чего хотелось бы достичь)
- эргономика (ограничения пользователей, мультимедиа);
- поддержка сайта;
- направления развития;

## **Рекомендации о фирменном стиле.**

**Требования к легкости дизайна** (как в плане скорости загрузки, так и в плане цветового решения)

**Требования к реализации интерфейса** (простой или "продвинутый". простой хорош для пользователей, продвинутый сложнее для понимания, но более впечатляющий, что может быть плюсом для корпоративного сайта)

# Техническая документация

**проектно-конструкторская (технические задания, спецификации ит.п.):**

- конструкторские документы, устанавливающие задачи программирования;
- программные документы, устанавливающие способ программирования;
- ведомости, систематизирующие объекты или предме
- схемы, графически изображающие части, элемент системы, их взаимосвязи;
- инструкции с изложением состава действий и прав использования;
- обоснования, подтверждающие целесообразно принимаемых решений;
- описания, поясняющие назначение системы, ее частей, принципов действия;
- исследования, излагающие и рекомендующие новые
- технологические решения

**проектно-сметная (регламенты и планы, участники работ, сметы и т.п.);**

**нормативно-техническая (стандарты, нормы, правила и т.п.);**

**техно-рабочая (операционные карты, руководства по эксплуатации и т.п.);**

**плановая (планы проекта, календарные графики работ и т.п.);**



# Коммерческая документация

Формируется в процессе техно-рабочего проектирования (а также на этапе внедрения):

- контрактная (протоколы о намерениях, договоры ит. п.);
- транспортная (накладные ит.п.);
- маркетинговая (исследования рынка, образцы рекламных материалов и т.п.).

## В общем случае каждый документ должен содержать

- титульный лист с названием;
- дату составления;
- формулировку назначения;
- область действия;
- категорию специалистов, для которых он предназначен;
- информацию об авторе;
- этапы работ или период, в течение которых документ имеет силу;
- содержательную часть в соответствии с назначением;

- По содержанию каждый документ должен представлять собой связный текст или графическое отображение с указанием смысла условных обозначений которые должны быть реализованы при создании сайта, а также фиксирует ограничения, накладываемые аудиторией, администраторами, ресурсами заказчика. Задача технической концепции (ТК) — установить, какая деятельность и в какой мере нуждается в автоматизации. Другими словами, ТК — ответ на вопрос о том, «что и зачем делать»;

- **техническая концепция (ТК)** устанавливает цели, задачи и идеи,
- **техническое задание (ТЗ)** фиксирует требования к решениям, которые должны быть реализованы при создании сайта. Задача ТЗ — определить способы автоматизации. ТЗ не отвечает не на вопросы о конкретных механизмах реализации, а на вопрос «как делать»;
- **эскизный проект (ЭП)** показывает конкретные способы реализации требований в части архитектуры и интерфейсов. Он является руководством для исполнителей (программистов и дизайнеров) по проектированию алгоритмов и функциональной модели. ЭП показывает, как именно и в какой последовательности разработчики собираются выполнить требования ТЗ заказчика. ЭП уточняет то, «как должен работать сайт»;
- **технический проект (ТП)** уточняет полученные интерфейсные и архитектурные решения на уровне функциональной структуры и алгоритмов взаимодействия отдельных программных элементов между собой. ТП — руководство для программистов и технических исполнителей по реализации архитектурных и интерфейсных решений ЭП. То есть главный вопрос ТП — «как будет работать».

# Проектирование БД

Для хранения информации в базе данных необходимо предварительно **определить группы и параметры данных**, свести эту информацию в реляционные таблицы и установить между ними **связи**. Кроме того, необходимо задать **первичные ключи и индексы, нормализовать структуру**.

Конечным результатом проектирования будет схема БД и типовой сценарий SQL на ее основе

# Принципы создания сайтов

- Эффективное размещение структурных элементов на сайте
  - Рассчитывать структурные элементы под самые часто используемые разрешения;
  - Выделять наиболее важные на текущий момент для посетителя структурные элементы;
  - Автоматическое расширение контента под разрешение экрана пользователя;
  - Фиксация максимальной ширины контента;
  - Применение логически обособленных панелей для контента, связанного с текущим;

# Принципы создания сайтов

- Создание сайтов с полноценной навигацией
  - Реализация многонаправленной навигации из каждого из узлов сайта;
  - Разделение контента на логические части и обеспечение возможности перехода по ним;
  - Иерархичность организации навигации;
  - Ориентация иерархии на задачи, которые необходимо решать пользователю;
  - Наличие алфавитных указателей;
  - Наличие хронологических карт контента;
  - Возможность динамической перегруппировки узлов иерархии в соответствии с популярностью;
  - Разделение всего контента сайта на логические участки (категории) и организация каждой из них в соответствии с её спецификой;

# Принципы создания сайтов

- Упрощение навигации

- Создание унифицированной иерархии просмотра сайта;
- Создание навигационной панели;
- Создание логических папок (Tabs);
- Создание кнопок для выполнения определённых действий и их возможное логическое выделение;
- Информирование посетителя о его текущем местонахождении;
- Использование встроенных в текст или картинки ссылок;
- Использование внешних (по отношению к основному текстовому содержанию) ссылок;
- Применение описательных и длинных имён ссылок;
- Адаптация языка предоставления информации под потенциального посетителя;
- Заблаговременное предотвращение ошибок (неактивные ссылки, при заполнении форм и т.д.)
- Осмысленные сообщения об ошибках;
- Показ вместо ошибки 404 (Page Not found) карты сайта;



# Принципы создания сайтов

- Повышение скорости работы сайта
  - Минимизация количества файлов;
  - Создание быстро загружаемых картинок и прочих объектов;
  - Использование множества отдельных независимых HTML таблиц для организации структуры сайта;
  - Применение возможностей HTML и стилей для упрощения логической группировки контента
  - Использование «многоразовых» картинок – применение на сайте не отдельных картинок в каждом из случаев а их «типизация»

# "Кривые" сайты вызывают у людей синдром мышиноного бешенства

Британский центр исследования социальных проблем (Social Issues Research Centre, SIRC) - отчет об отрицательном влиянии Интернета на здоровье человека.

**Пять основных факторов, отягощающих жизнь простых интернетчиков – это!**

1. медленная загрузка страниц («тонкий канал»).
2. неудобная («кривая») навигация.
3. злоупотребление создателя сайта всплывающими окнами («поп-апами»).
4. обилие рекламы («баннеры задолбали») и
5. недоступность нужного ресурса («упал», «умер»).

**Сайт, обладающий хотя бы одним из этих недостатков, вызывает синдром мышиноного бешенства» (mouse rage syndrome).**

**Большинство пользователей, как показали исследования, хотят, чтобы все сайты были похожи на Google.**

- Одна из главных проблем при разработке и выборе психологических методов – основные понятия, которые используются для оценки деятельности.
- В ISO стандартах по usability такими понятиями являются **умственная рабочая нагрузка (*mental workload*)** или **умственная напряженность (*mental tension*)**, а также **негативные функциональные состояния (*mood states*)** – утомление, монотония, снижение бдительности и т.п., или **эмоциональные реакции**, вызываемые **трудностями** при работе с компьютером.

- Иногда еще рассматривают *психофизиологические ресурсы* человека. Видно, что деятельность и функциональные состояния при этом фактически не разделяются

- Методы измерения умственной рабочей нагрузки разделяют на три категории:
  - субъективное мнение,
  - резервные психические возможности,
  - вторичная задача.

# Субъективные мнения

- анализируются с помощью различных оценочных шкал и вопросников.
- Одним из наиболее распространенных методов в настоящее время является индекс нагрузки задачи НАСА. Этот индекс включает шесть шкал: умственных, физических, временных требований, оценки результатов деятельности, усилий и фрустрации (состояние безысходности).

## Для измерения *резервных психических возможностей*

- используются оценки разных психофизиологических показателей  
Например, часто используют вариативность частоты сердечных сокращений, изменение диаметра зрачка, частоту зрительного сканирования и т.д.

# Посетители\посещения

- Типы (люди\роботы)
- Цели посещения и характер поведения
- Модели перехода по системе и используемые функции



# Определение характеристик посетителей сайта

- Анализ характера взаимодействия пользователя и сайта позволяет определить что нравится людям, а что нет;
- Анализ условий поиска пользователей позволяет определить в чём пользователи заинтересованы;
- Возможность использования опросов, анкетирования или соревнований позволяет более точно выявить восприятие пользователем области сайта;
- Предоставление пользователям возможности ведения личных деталей позволит собрать ценную информацию о клиентской базе;

# Восприятие посетителей сайта как людей

- Все посетители имеют свою индивидуальность
  - Демографическая принадлежность;
  - Отношение к чему либо;
  - Поведение;
  - Знания и навыки;
- У посетителей есть общие особенности
  - Физиологические особенности и познавательные способности;
  - Психологические шаблоны поведения;
  - Эргономическое восприятие;

# Восприятие посетителей сайта как людей

- Понимание задач посетителей
  - Выявление полного перечня того, что может сделать посетитель на сайте и что ему может понадобиться сделать;
  - Выявление путей того, как сайт может помочь в решение текущих проблем посетителя;
  - Для облегчения решения задач целесообразно использовать объекты, привычные посетителям из обычной жизни (например корзина товаров в интернет магазине)

# Сегментация посетителей

Сегментация – группировка посетителей в группы (сегменты) с целью более лёгкого и более точного их понимания, а также дальнейшего обслуживания

## Сегментация основывается на:

### • Отчётности и анализе

- Возможная и фактическая прибыльность посетителя;
- Частота совершения интересующих бизнес действий (например покупки);
- Финансовые взаимоотношения с клиентом;
- Участие в маркетинговых акциях;
- Географический признак;
- Возрастная группа;
- Профессия;
- Риск;
- Пол;
- Национальность;
- Статус взаимоотношений с компанией (в процессе заказа и др.);
- Отношение к компании (внешний человек, сотрудник и др.);
- Поведенческие характеристики:
  - Лояльность и длительность использования сайта;
  - Тип запрашиваемого контента;
  - Объёмы получаемой информации;
  - Дата последнего визита;

### • Маркетинговых кампаниях

### • Персонализации

# Шаблон сегментации посетителей сайта

- Название целевого сегмента;
- Множество идентификационных признаков;
- Важность целевого сегмента для деятельности компании;
- Ключевые причины использования сайта;
- Поведенческие свойства;
- Возможности осуществления персонализации под данный целевой сегмент;

# Этапы проекта по сегментации посетителей

- Определение границ проекта и желаемой целевой аудитории;
- Онлайн исследования и первичная сегментация;
- Тестирование контента под выделенные целевые сегменты;
- Ценовое моделирование и анализ возможной прибыльности;
- Определение правил взаимодействия с представителями целевой аудитории для максимизации прибыли;
- Определение целей взаимодействия с целевой аудиторией;

# Данные посетителей

- Необходимо иметь
  - Контактная информация и информация для отправки заказов;
  - Сферы интереса людей и сегменты работы организаций;
- Важно иметь
  - Альтернативные способы связи;
  - Пол, возраст, национальность;
  - Дата рождения;
- Хорошо бы иметь
  - Откуда посетители узнают о нас;
  - Есть ли у посетителей связи, которые были бы интересны компании;
  - С кем из конкурентов взаимодействовал посетитель;
  - Какие прочие сайты посещает посетитель;



## Посетители интернет магазинов

- Случайные посетители (около 37%)
- Разовые посетители (около 30%)
- Посетители отдельных сайтов (около 25%)
- Активные посетители (около 8%)

## Посетители по степени лояльности

- «целевые» - приходят из поисковиков по целевым запросам
- «постоянные» - обычно ищут только обновлений на сайте, поэтому чаще заходят в раздел новости и если ничего нового нет, то сразу уходят с сайта
- «преданные» - заходят на сайт чаще одного раза в неделю и проводят некоторое время на нём

# Модели поведения посетителей интернет ресурсов

- «Торопливые»
  - Порядка 15 секунд на 1 страницу;
  - Вся работа с сайтом несколько минут;
- «Искатели фактов»
  - На сессию тратится около 9 минут;
  - Страницы также просматриваются очень быстро;
- «Выполнение какой-либо одной миссии»
  - Сессия длится порядка 10 минут
  - Пользователи обычно заходят на незнакомые им сайты;
  - Пользователи открыты для той информации, которые связаны с их целью;
- Повторные посетители
  - Сессия порядка 14 минут;
  - Страница просматривается около 1-2 минут;
- Бесцельный серфинг
  - Сессия длится в районе получаса;
  - На одну страницу уходит от 2х минут;
  - Чаще интересуют сайты с новостями, играми и прочими развлечениями;
- «Любители информации»
  - Время сессии в районе 37 минут;
  - Производится детальный поиск информации (обычно сразу на нескольких сайтах);
  - Возможность воздействовать на посетителя маркетинговыми инструментами;
- Просто серфинг
  - Средняя длительность около 70 минут;
  - На одной странице порядка 1-2 минут;
  - Полное, но не глубокое изучение контента

( <http://sitemaker.ru/promo/internetmarketing/usersmodels/> )

Для оптимизации содержания сайта под его посетителей необходимо:

- Разбить структуру сайта на типы:
  - Физическая
  - Логическая
  - Функциональная
  - Модульная
- Произвести сохранение путей перехода пользователей по сайту;
- Произвести наложение путей перехода на различные типы структур и выявить полученные закономерности;

Наиболее эффективный метод анализа аудитории – глобальные статистические системы:

- Исследование посетительских потоков и управление ими;
- Исследование поведения посетителей на сайтах и вне его
- Исследование удобства и использования посетителями структуры сайта;
- Сравнительный анализ посетительских аудиторий;
- Сравнительный анализ отдельных сайтов, групп сайтов или сегментов сети;