Рекомендации к тексту курсача (о чем вообще писать):

1. Обзор используемых технологий, в частности HTML, CSS, JS, системы контроля версий и базовые особенности их применения. Сравнение с другими технологиями, если нужно.
2. Краткое описание сферы деятельности для которой разрабатывается сайт.
3. Описание структуры разрабатываемого сайта в виде того что он ДОЛЖЕН содержать. (В народе это называется ТЗ).
4. Описание процесса разработки ключевых элементов сайта. Что и в какой последовательности вы разрабатывали.

Курсовая должна содержать:

* Титульный лист
* Задание
* Реферат
* Содержание
* Введение
  + Краткая актуальность
  + Цель и задачи (мин. 4)
  + Объект и предмет
* 1 Название раздел
* 2 Название раздел
* Заключение
* Список использованных источников

Введение

основная часть

1. разбор используемых технологий веб-программирования

Веб-разработка — процесс создания веб-сайта или веб-приложения. Основными этапами процесса являются веб-дизайн, верстка страниц, программирование на стороне клиента и сервера, а также конфигурирование веб-сервера.

Верстка и дизайн страниц осуществляется с применением веб-технологий – комплекса современных инструментальных и программных средств разработки web-сайтов и серверного программного обеспечения.

Современные веб-технологий включают в себя языки разметки гипертекста, средства создания визуального оформления, языки программирования. Далее будет проведен обзор использованных технологий.

1.1 html

HTML (HyperText Markup Language) – язык разметки гипертекста, язык создания веб-страниц. HTML-документ содержит текст и команды языка HTML, которые называются дескрипторы разметки или теги. Теги языка HTML определяют внешний вид документа, структуру документа (взаимное расположение текстовой, графической и другой мультимедийной информации), ссылки на другие интернет-ресурсы. При написании кода страницы идентификаторы тегов заключаются в угловые скобки < >.

HTML позволяет формировать на странице сайта текстовые блоки, включать в них изображения, организовывать таблицы, управлять отображением цвета документа и задавать форматирование текста, добавлять звуковое сопровождение, а также задавать взаимное расположение текстовых блоков и других компонент страницы. Любой web-браузер содержит интерпретатор языка HTML, что позволяет ему корректно отображать web-страницу вместе со всем ее содержимым на экране.

В современной веб-разработке принято придерживаться принципа разделения описания логической структуры web-страницы (которое производится с помощью языков разметки) и описания ее внешнего вида. Такой подход позволяет отделить содержимое HTML-документа от дизайна и верстки.

1.2 css

Для описания внешнего вида элементов web-страницы на экране браузера используют технологию стилевого оформления – таблицы каскадных стилей CSS (Cascading Style Sheets).

Стиль оформления включает в себя все то, что определяет внешний вид документа: цвета, шрифты, рамки и границы, отступы, другое оформление и выравнивание текста, расположения отдельных блоков, обтекание и т.д. Стиль определяется набором правил отображения тегов, задаваемых таблицей стилей.

Таблица стилей – это шаблон, который управляет форматированием тегов HTML в web-документе. Таблица стилей состоит из набора правил описания стиля. Любое правило каскадных таблиц стилей состоит из двух частей: селектора и определения. Селектор – это любой элемент или группа элементов web-страницы, для которых определяется форматирование. Определение описывает конкретный вид форматирования и состоит из двух частей: свойства и значения, разделенных знаком двоеточия.

Использование CSS целесообразно, поскольку возможности оформления, предоставляемые этой технологией, гораздо шире стандартных средств форматирования языка HTML. CSS позволяет определить стиль сразу для группы элементов, позволяет сохранить стиль отдельной страницы и применить его сразу к нескольким страницам, задав для них, таким образом, единое оформление. Это значительно повышает гибкость определения и модификации дизайна страницы.

Использование только технологий HTML и CSS позволяет создавать только статические страницы сайтов с низкой функциональностью. Страницы статических сайтов создаются заранее, для их модификации код редактируют вручную и затем заново загружают на сайт.

Если информация, размещенная на сайте, требует постоянной актуализации, и если на сайте необходимо осуществлять взаимодействие с пользователем, статический сайт не достаточно удобен и необходимо прибегать к технологиям создания динамического сайта.

Динамический сайт – это сайт с динамическим информационным наполнением. Динамические сайты также основываются на статических данных и HTML-разметке, но дополнительно содержат программную часть – скрипты, благодаря которым страница собирается из отдельных фрагментов в режиме реального времени. Таким образом, динамический сайт состоит из набора «строительных блоков» динамических страниц – шаблонов дизайна, информационного наполнения, скриптов, хранящихся в виде отдельных файлов. При работе с динамическим сайтом для изменения страницы достаточно изменить ее информационное наполнение, а сам механизм формирования и вывода страницы остается неизменным. Кроме информационного наполнения, динамически могут создаваться также и элементы навигации по сайту.

Динамические сайты различаются в зависимости от используемых технологий. Процесс получения динамических страниц также может различаться:

1. Генерация на стороне сервера (осуществляется серверными скриптами на языках PHP, Perl, ASP.NET, Java, Python и др., а информационное наполнение хранится в базах данных).

2. Генерация на стороне клиента (JavaScript).

3. Комбинированная генерация (чаще всего на практике встречается именно комбинация первых двух методов).

1.3 js

Одним из средств добавления динамических эффектов на веб-страницу, задания реакции на пользовательские действия является язык программирования JavaScript.

JavaScript – объектно-ориентированный язык для создания клиентских web-приложений. Сценарии JavaScript внедряются в web-страницу или связываются с ней и после загрузки страницы с сервера выполняются браузером на стороне клиента. JavaScript используется для обработки событий, связанных с вводом и просмотром информации на web-страницах.

JavaScript обычно используется в клиентской части клиент-серверных программ (клиентом выступает браузер, а сервером – web-сервер), имеющих распределённую между сервером и клиентом логику. Клиентская часть среды проектирования web-приложений содержит следующие основные компоненты: браузер – средство просмотра web-страниц, языки разработки web-страниц HTML/XHTML и CSS, языки сценариев (в настоящее время в качестве стандарта принят JavaScript), клиентские расширения, такие как элементы управления ActiveX, Java-апплеты, подключаемые модули, такие как плееры Macromedia Flash или Silverlight и т. д. Язык сценариев JavaScript позволяет создавать интерактивные web-страницы и содержит средства управления окнами браузера, элементами HTML-документов и стилями CSS.

1.4 git

2. описание сферы деятельности

В качестве темы курсовой работы была выбрана разработка адаптивного сайта для производителя автосигнализаций.

Компания, производящая автосигнализации, занимается разработкой и производством устройств для защиты автомобилей от кражи и несанкционированного доступа. Задачи компании включают в себя создание надежных и инновационных продуктов, а также обеспечение высокого уровня сервиса для своих клиентов. Помимо автосигнализаций, выбранная компания автомагнитолы. Основной задачей компании является производство качественных и надежных устройств, которые будут удовлетворять потребности клиентов.

Сайт компании должен отражать ее профессионализм и качество продукции. На главной странице должно быть краткое описание компании и ее продуктов, а также ссылки на разделы сайта, где можно узнать больше о каждом продукте.

Функционал сайта должен включать возможность поиска продуктов по категориям, форму обратной связи, а также информацию о сервисных центрах и гарантийных условиях. Также может быть полезным добавить раздел с отзывами клиентов и статьи о том, как правильно устанавливать и использовать автосигнализации.

В целом, сайт компании должен быть удобным и информативным для потенциальных клиентов, а также обеспечивать легкий доступ к необходимой информации и услугам.

3. описание структуры разрабатываемого сайта

Структура сайта будет включать:

1. Главная страница с кратким описанием компании и ее продуктов, информацией о структуре компании и основных моментах, связанных с ее работой, а также ссылками на разделы сайта.

2. страница с информацией о бренде производителя

3. страница с новостями, где можно посмотреть последние модели, поступившие в продажу, обновления в работе компании и т.п.

4. Каталог продуктов с описанием моделей и фотографиями + страницы для каждого продукта

5. Раздел с информацией о сервисных центрах и гарантийных условиях, страница с формой, где можно подать заявку на получение расширенной гарантии для покупателей.

6. Статьи о том, как правильно устанавливать и использовать автосигнализации.

7. Контакты компании с формой обратной связи.

Каждая страница будет включать:

1. шапка, содержащая меню навигации по сайту
2. блок с основным содержанием страницы
3. подвал сайта с ссылками навигации по сайту

Описание содержания разрабатываемых страниц:

4. описание процесса разработки

4.1 этап планирования

4.2 этап дизайна

4.3 этап разработки

заключение

список источников

<https://cyberleninka.ru/article/n/obzor-sovremennyh-web-tehnologiy/viewer>

<https://lred.ru/webmasteru/573---web->

<https://dfedorov.spb.ru/Web_Tehnologii.pdf>

<https://git-scm.com/book/ru/v2/%D0%92%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5-%D0%9E-%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B5-%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D1%8F-%D0%B2%D0%B5%D1%80%D1%81%D0%B8%D0%B9>