Государственное АВТОНОМНОЕ профессиональное образовательное учреждение

свердловской области

«Каменск-Уральский политехнический колледж»

Специальность 09.02.07 Информационные системы и программирование

Группа ИСиП20-302

КУРСОВОЙ ПРОЕКТ

по МДК 05.02 разработка кода информационных систем

тема: Разработка клиентской части веб-приложения для автосервиса

Выполнил:

А.М. Горбунов

Проверил:

Н.В. Бодня

2023

Содержание

[Введение 3](#_Toc130311797)

[1.Аналитическая часть 4](#_Toc130311798)

[1.1 Анализ предметной области 4](#_Toc130311799)

[1.2 Постановка задачи 5](#_Toc130311800)

[2. Проектная часть 6](#_Toc130311801)

[2.1 Разработка информационной структуры веб-приложения 6](#_Toc130311802)

[2.2 Разработка макета дизайна веб-приложения 12](#_Toc130311803)

[2.3 Реализация макета веб-приложения 16](#_Toc130311804)

[Заключение 34](#_Toc130311805)

[Список литературы 35](#_Toc130311806)

# Введение

В настоящее время интернет является неотъемлемой частью жизни людей, без которой существование уже и не представляется возможным. Но для взаимодействия через интернет нужны приложения, позволяющие это осуществить, то есть интернет-приложения.

В современном мире практически каждое предприятие имеет свой собственный web-сайт или web-приложение. Также, созданное web-приложение способствует решению некоторых задач, это ускоряет обслуживание клиентов и приводит к увеличению прибыли предприятия.

Автосервис — это комплекс услуг, связанный с техническим обслуживанием и ремонтом авто и мототранспорта.

Как правило, автосервис осуществляется на станциях технического обслуживания (СТО) и ориентирован как на личный автотранспорт физических лиц, так и на автопарк организаций.

Актуальность бизнеса по ремонту и обслуживанию автотранспорта связана со взрывным ростом уровня автомобилизации.

Актуальность исследования обусловлена необходимостью предоставления полной достоверной информации о предоставляемых услугах, а также оформлению заявки на ремонт в автосервис.

Цель данного проекта разработать клиентскую часть веб-приложения для автосервиса.

Для достижения поставленной цели необходимо выполнить следующие задачи:

* + Провести анализ предметной области.
  + Построить функциональную модель IDEF0.
  + Разработать структуру веб-приложения.
  + Разработать макет дизайна.
  + Разработать клиентскую часть веб-приложения.

# 1.Аналитическая часть

# 1.1 Анализ предметной области

Как правило, автосервис осуществляется на станциях технического обслуживания (СТО) и ориентирован как на личный автотранспорт физических лиц, так и на автопарк организаций.

В автомастерской могут оказываться следующие услуги: Диагностика авто, различные ремонтные работы, сварочные работы, замена масла и другое.

Оформление заявки занимает некоторое количество времени, чтобы уменьшить эту потерю времени было решено создать веб-приложение, в котором клиенты смогут записаться на ремонт онлайн.

Объект исследования – процесс разработки web-приложения для оформления заявки на ремонт в автосервис.

Предмет исследования – разработка дизайна и пользовательского интерфейса web-приложения автосервиса.

Цель исследования – разработать web-приложение для автосервиса, обеспечивающее возможность оформления заявки на ремонт автомобиля.

На рисунке 1 представлена функциональная модель процесса «оформления заявки на ремонт».



Рис.1. IDEF0 «Оформление заявки на ремонт»

Для реализации этого процесса:

Входом является: заявка клиента на ремонт

Выходом является: оформленная заявка клиента

Механизмом является: администратор и ИС

Управлением является: нормативная документация и ГОСТ Р 7.0.97-2016

# 1.2 Постановка задачи

Проведя анализ предметной области пришли к выводу что необходимо разработать веб-приложение, которое позволит посетителям сайта онлайн записаться на ремонт авто.

Веб приложение позволит:

* Укрепить бренд.
* Увеличить продажи.
* Привлечь внимание аудитории.
* Сократить издержки по привлечению новых посетителей.
* Увеличить средний чек.
* Удерживать старых клиентов с меньшими затратами.

В данном приложении будет доступно: просмотр информации об автосервисе, прайс его услуг и запись на ремонт. Что позволит любому пользователю быстро получить информацию и решить для себя нужен-ли ему именно этот автосервис или нет.

# 2. Проектная часть

# 2.1 Разработка информационной структуры веб-сайта

Процессу верстки web-приложения предшествует большая работа. Планируя создание web-проекта, мы должны в первую очередь определить задачи, которые он должен решать. От этого зависит и успешность будущего проекта и подход к его созданию. Всё многообразие сайтов и web-приложений в интернете можно разделить на 3 типа: одностраничные, визитки и многостраничные.

Одностраничный

Такой сайт или web-приложение еще называют лэндингом. Это приложение, всё содержимое которого помещается на одной странице, которая просматривается прокруткой. У него нет разветвленной структуры, это просто длинная страница с логическими блоками, которую пользователь пролистывает по мере чтения.

Лэндинг — это специфический сайт, он создается специально для того, чтобы подтолкнуть пользователя к какому-либо действию. Это может быть покупка товара, запись на курс, звонок в офис, регистрация и так далее. Для этого в лэндинге присутствуют контактная форма, различные кнопки (звонок, сообщение в мессенджер), призывы к действию, мотивирующие видео, иллюстрации, списки преимуществ — в общем, всё то, что поможет решить его главную цель.

Цель лэндинга — перевести посетителя web-приложения в статус клиента, покупателя или подписчика. Поэтому иногда его еще называют целевой страницей. В лэндингеb содержится уникальное торговое предложение (УТП). Обычно на одностраничном сайте представлено только одно УТП.

Плюсы лэндинга в том, что он мало весит и быстро делается. Для его заполнения не нужно много текста, главное — красиво оформить смысловые блоки.

Структура — это логическое построение всех страниц сайта или web-приложения, категорий и подкатегорий. Это логическая схема, в соответствии с которой все страницы и разделы расположены относительно друг друга и принцип, по которому они друг с другом взаимосвязаны.

Требования к структуре:

* Присутствие навигационных цепочек (хлебных крошек).
* Соблюдение правила 3-х кликов
* Использование категорий для группировки связанного контента.
* Использование HTML и CSS для навигации.
* Наличие ссылок на главную со всех страниц.
* Неизменность структуры при увеличении или уменьшении количества категорий и подкатегорий.
* Наличие карты сайта для посетителей.

Исходя из вышеперечисленного была создана следующая структура web-приложения:

Список страниц: Главная страница, о нас, услуги.

Содержание главной страницы: Меню, лозунг, краткая информация об автосервисе, популярные услуги, наши мастера, форма записи на ремонт, контактная информация.

Целевая аудитория. Возраст: 25-40, пол: мужской, сфера деятельности: разнообразная, потребности: квалифицированный сервис и починка авто, уровень дохода: от 50 тыс. Рублей в месяц.

Цель: коммерческая;

Задачи: реклама автосервиса, дать возможность клиентам записаться на ремонт онлайн.

На рисунке 2 представлена карта web-приложения для автосервиса:

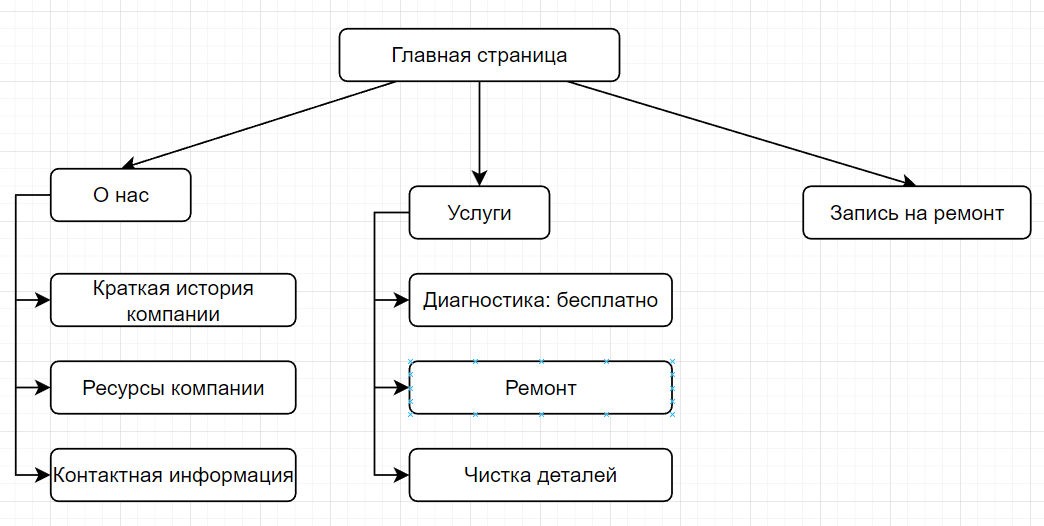


Рис.2. «Карта сайта»

В структуру будут входить три страницы, которые будут связаны между собой ссылками в меню навигации.

Каркас – это базовая платформа. Что-то в духе скелета для проекта, на который впоследствии будут натягиваться другие детали приложения или сайта. Это оболочка, несущая в себе базовую функциональность.

Каркас рассказывает о том, как будет выглядеть или работать ваше web-приложение, но при этом не бросает в глаза красками и стилевыми решениями. Но это не совсем уж абстрактный набросок, который невозможно будет при необходимости использовать в ходе практической задачи.

Наличие качественного плана действий всегда в значительной степени упрощает работу над проектом. Многие разработчики и предприниматели отказываются от этого этапа, ссылаясь на то, что уйдет слишком много времени и средств на проработку идей, и лучше бы сразу преступить к реализации.

Наличие четкого плана дает следующие преимущества:

1. Путь, по которому можно двигаться, точно зная, какую функцию надо реализовать и как это сделать.
2. Возможность быстрее вести разработку, так как можно практически полностью вычеркнуть планирование из рабочей программы программистов.
3. Отсутствие примитивных ошибок и тупиков, возникающих из-за того, что команда сама не знает, как должен выглядеть итоговый продукт.

На рисунке 3 представлен каркас главной страницы сайта для автосервиса.

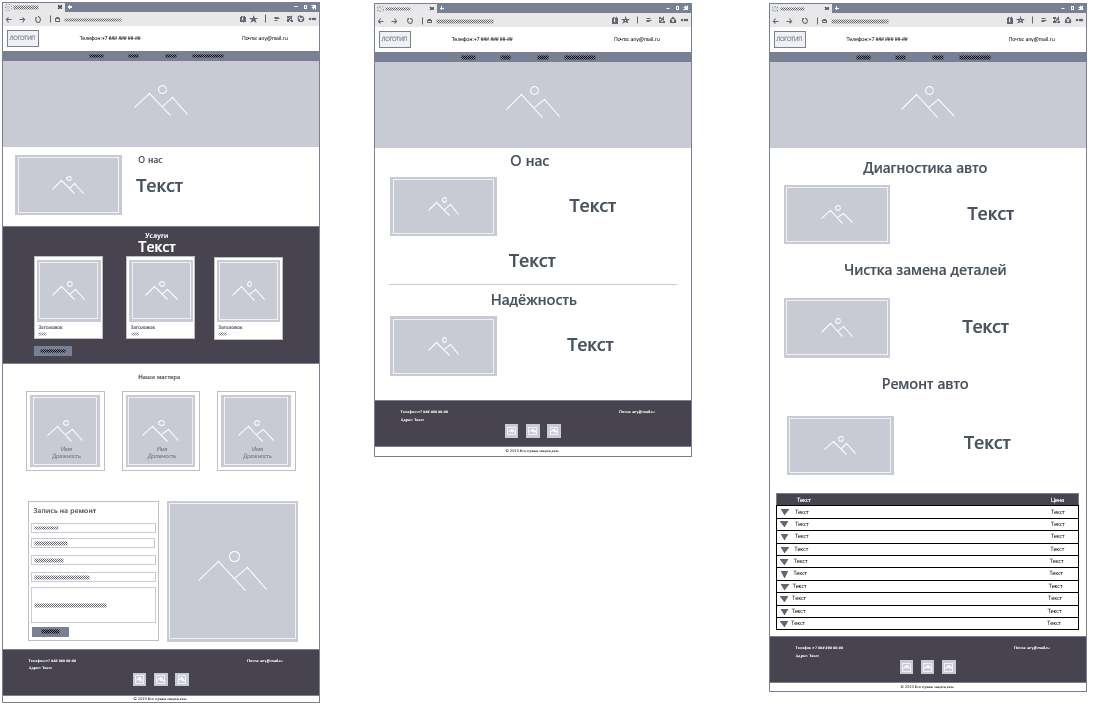


Рис.3. «Каркас главной страницы»

На рисунке 4 представлен каркас страницы сайта с информацией об автосервисе.

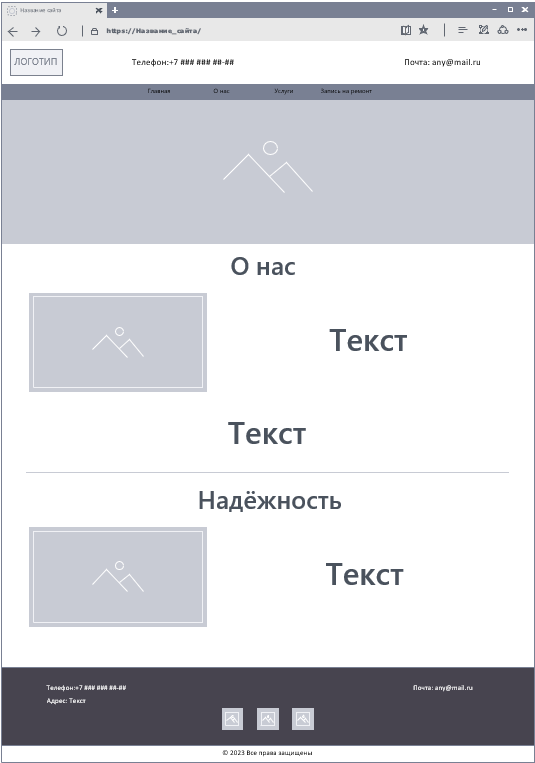


Рис.4. «Каркас страницы с информацией»

На рисунке 5 представлен каркас страницы сайта с информацией об услугах автосервиса.

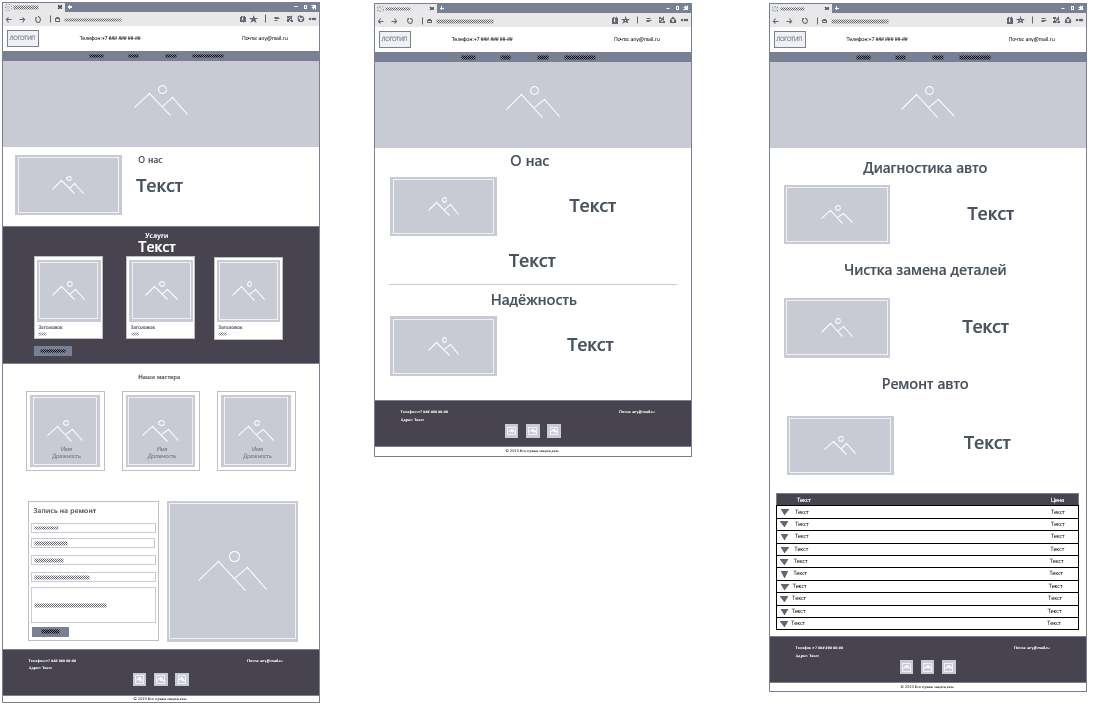


Рис.5. «Каркас страницы с услугами»

Разработан каркас основных страниц на котором изображено расположение контента и навигация.

# 2.2 Разработка макета дизайна веб-приложения

Макет — это предварительный набросок, который определяет внешний вид web-приложения и его функционал. Как перед постройкой дома рисуют чертеж, так и перед разработкой web-приложения сначала нужно визуализировать его в специальной программе.

Макет нужен, чтобы весь процесс прошел быстро: исполнителям пришлось вносить минимум правок, а заказчик сэкономил время и деньги.

Дизайн web-приложения — это визуальное оформление страниц, сочетание всех его графических элементов.

Основные правила web-дизайна:

1. Юзабилити, или удобство использования
2. Читаемые шрифты
3. Умеренная цветовая палитра
4. Единый стиль
5. Больше визуализации
6. Адаптивная вёрстка

Юзабилити (usability) — это качественный показатель, который определяет, насколько просто и удобно пользоваться сайтом или web-приложением.

Юзабилити оценивают по пяти основным параметрам. Определяют, насколько:

1. Новым пользователям легко ориентироваться на сайте или web-приложении.
2. Посетителю ресурса просто совершить целевое действие — прочитать статью или заказать товар — и не запутаться в интерфейсе.
3. Пользователь быстро сможет сориентироваться спустя время.
4. Посетитель часто совершает ошибки при работе с интерфейсом и быстро обнаруживает ошибочность своих действий.
5. Пользователь в целом удовлетворен ресурсом.

На рисунке 6 представлен дизайн макета главной страницы web-приложения для автосервиса.



Рис.6. «Макет главной страницы web-приложения»

На рисунке 7 представлен дизайн макета страницы с информацией об автосервисе.



Рис.7. «Макет страницы с информацией»

На рисунке 8 представлен дизайн макета страницы с информацией об услугах автосервиса.



Рис.8. «Макет страницы с услугами»

# 2.3 Реализация макета web-приложения

Web-приложение автосервиса «AutoHelp» создано с использованием следующих языков программирования: HTML, CSS, JavaScript.

HTML — это код, который используется для структурирования и отображения веб-страницы и её контента. Например, контент может быть структурирован внутри множества параграфов, маркированных списков или с использованием изображений и таблиц данных.

CSS — язык таблиц стилей, который позволяет прикреплять стиль (например, шрифты и цвет) к структурированным документам (например, документам HTML и приложениям XML).

JavaScript — это полноценный динамический язык программирования, который применяется к HTML документу, и может обеспечить динамическую интерактивность в web-приложении.

При создании web-приложения автосервиса «AutoHelp» использовались следующие программы: Visual Studio Code, Figma, Microsoft Visio, веб-браузер Google Chrome.

Visual Studio Code — это редактор кода для разных языков программирования. Он относительно немного весит, гибкий и удобный. В нем можно писать, форматировать и редактировать код на разных языках.

Figma — онлайн-сервис для дизайнеров, веб-разработчиков и маркетологов. Он предназначен для создания прототипов сайтов или web-приложений, иллюстраций и векторной графики. В редакторе можно настроить совместную работу, вносить и обсуждать правки, причем как в браузере, так и через приложение на компьютере.

Microsoft Visio — это векторный графический редактор, позволяющий создавать и редактировать блок-схемы и диаграммы, т. е. решать задачи, с которыми регулярно сталкиваются сотрудники самых разных подразделений любой компании.

Веб-браузер Google Chrome — это кроссплатформенный веб-браузер, разработанный компанией Google.

На главной странице web-приложения автосервиса «АвтоДоктор» для создания шапки, изображенной на рисунке 8.1, был написан код для создания структуры, изображенный на рисунке 8.2 и код для создания стилей шапки, изображенный на рисунке 8.3.

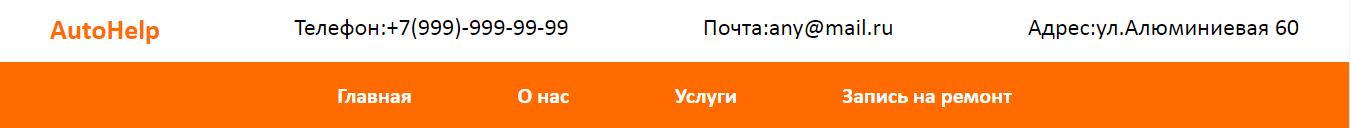


Рис. 8.1 «Шапка главной страницы web-приложения»

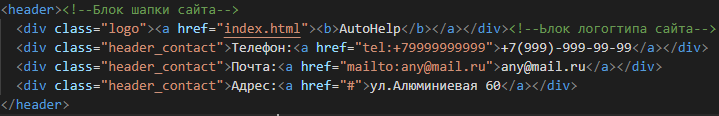


Рис. 8.2 «Код для создания структуры шапки главной страницы web-приложения»

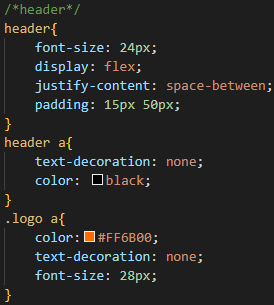


Рис. 5.3 «Код для стилей шапки главной страницы web-приложения»

На главной странице web-приложения автосервиса «AutoHelp» для создания блока «Слайдер», изображенного на рисунке 8.4, использовался код для создания структуры блока, изображенный на рисунке 8.5 и код для создания стилей блока, изображенный на рисунке 8.6.



Рис. 8.4 «Слайдер»

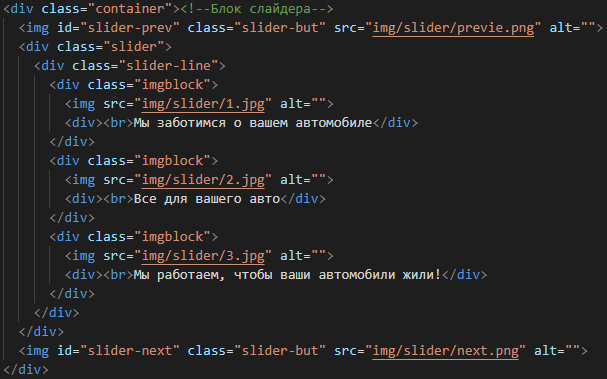


Рис. 8.5 «Код для создания структуры блока «Слайдер»»

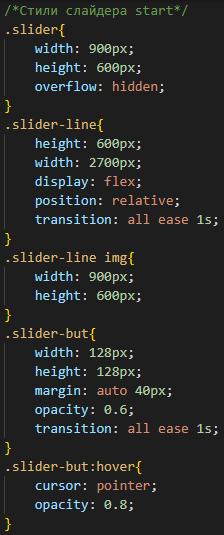


Рис. 8.6 «Код для стилей блока «Слайдер»»

На главной странице web-приложения автосервиса «AutoHelp» для создания блока «О нас» изображенного на рисунке 8.7, использовался код для создания структуры блока, изображенный на рисунке 8.8 и код для создания стилей блока, изображенный на рисунке 8.9.



Рис. 8.7 «Блок «О нас» главной страницы web-приложения»

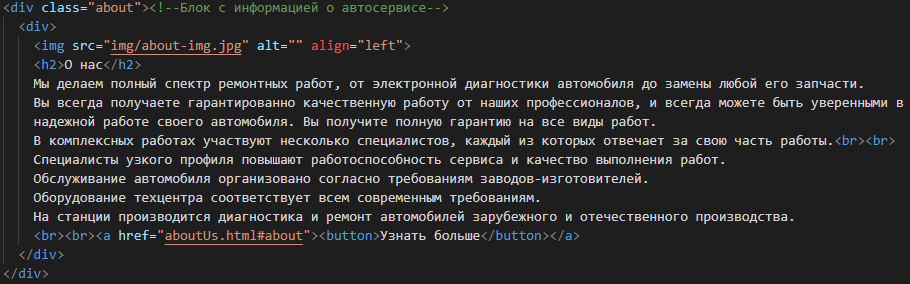


Рис. 8.8 «Код для создания структуры блока «О нас» главной страницы web-приложения»

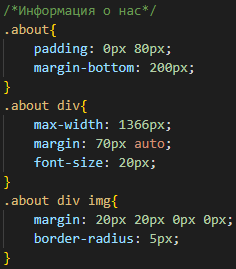


Рис. 8.9 «Код для создания стилей блока «О нас» главной страницы web-приложения»

На главной странице web-приложения автосервиса «AutoHelp» для создания блока «Услуги» изображенного на рисунке 8.10, использовался код для создания структуры блока, изображенный на рисунке 8.11 и код для создания стилей блока, изображенный на рисунке 8.12.



Рис. 8.10 «Блок «Услуги» главной страницы web-приложения»

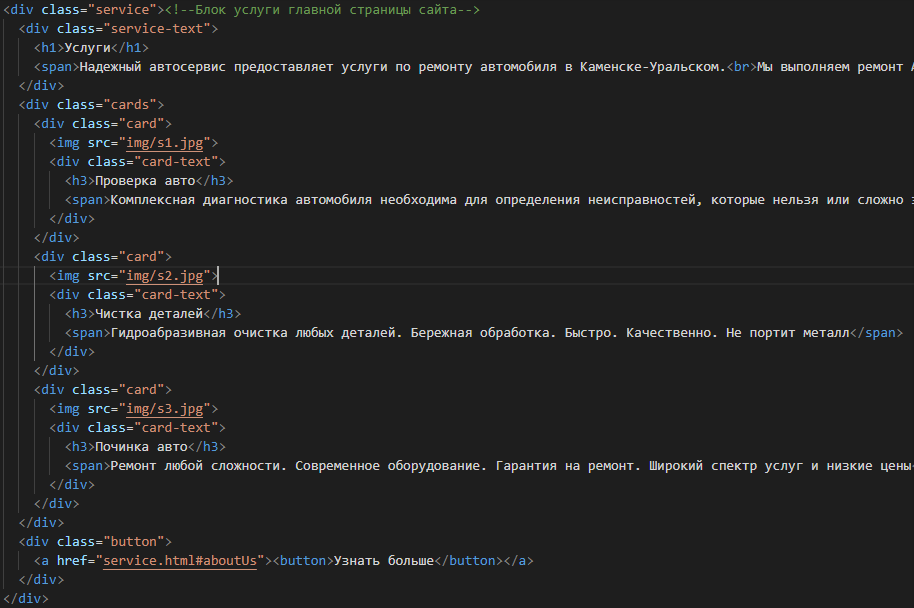


Рис. 8.11 «Код для создания структуры блока «Услуги» главной страницы web-приложения»

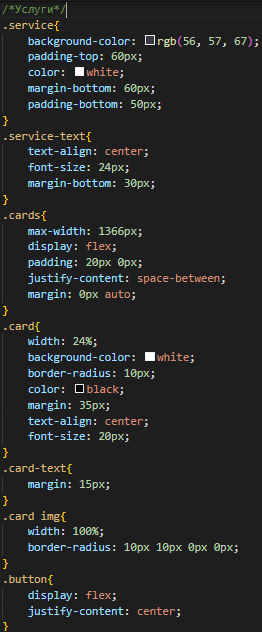


Рис. 8.12 «Код для создания стилей блока «Услуги» главной страницы web-приложения»

На главной странице web-приложения автосервиса «AutoHelp» для создания блока «Наши мастера» изображенного на рисунке 8.13, использовался код для создания структуры блока, изображенный на рисунке 8.14 и код для создания стилей блока, изображенный на рисунке 8.15.

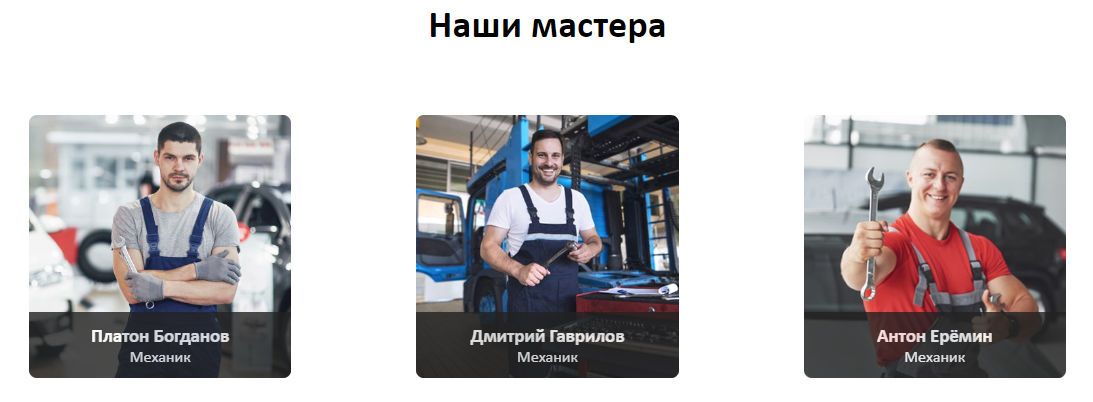


Рис. 8.13 «Блок «Наши мастера» главной страницы web-приложения»



Рис. 8.14 «Код для создания структуры блока «Услуги» главной страницы web-приложения»

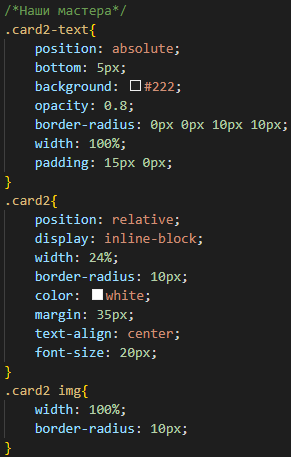


Рис. 8.15 «Код для создания стилей блока «Услуги» главной страницы web-приложения»

На главной странице web-приложения автосервиса «AutoHelp» для создания блока «Форма записи» изображенного на рисунке 8.16, использовался код для создания структуры блока, изображенный на рисунке 8.17 и код для создания стилей блока, изображенный на рисунке 8.18.



Рис. 8.16 «Блок «Наши мастера» главной страницы web-приложения»

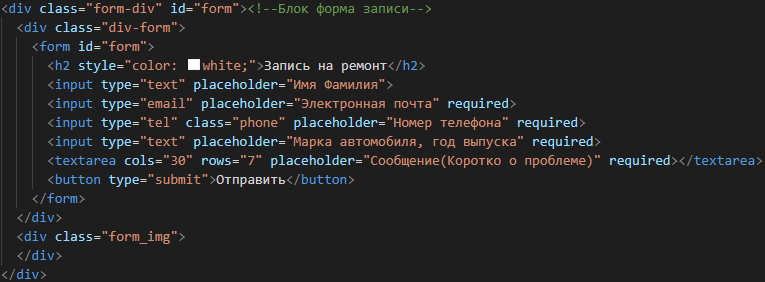


Рис. 8.17 «Код для создания структуры блока «Услуги» главной страницы web-приложения»

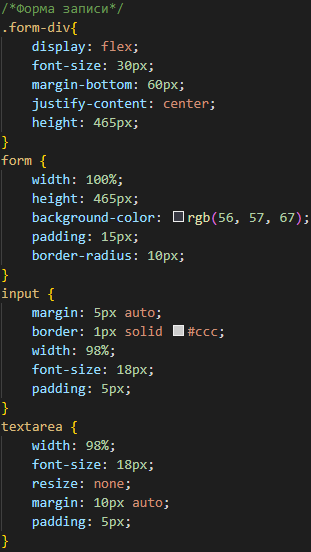
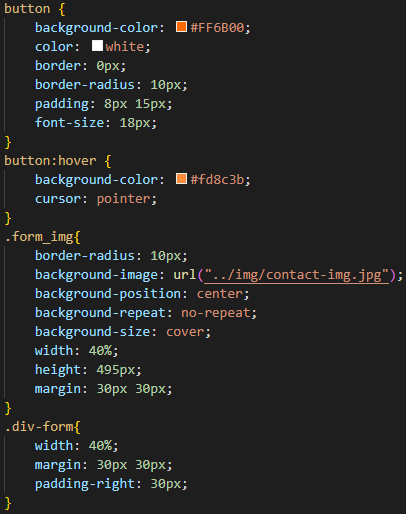
 

Рис. 8.18 «Код для создания стилей блока «Услуги» главной страницы web-приложения»

На главной странице web-приложения автосервиса «AutoHelp» для создания блока «Подвал» изображенного на рисунке 8.19, использовался код для создания структуры блока, изображенный на рисунке 8.20 и код для создания стилей блока, изображенный на рисунке 8.21.

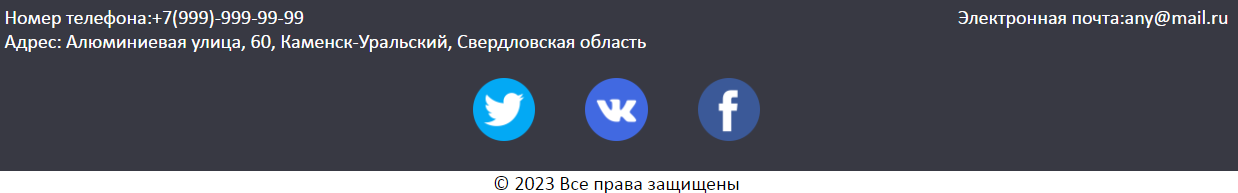


Рис. 8.19 «Подвал главной страницы web-приложения»

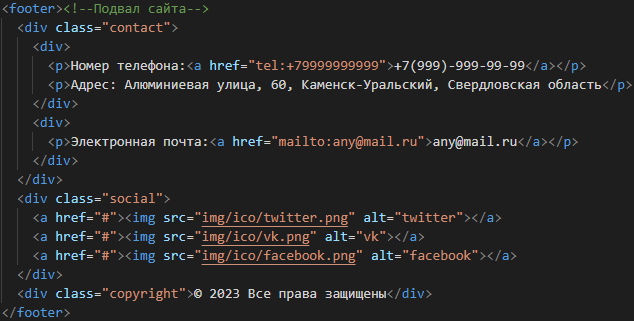


Рис. 8.20 «Код для создания структуры подвала сайта главной страницы web-приложения»

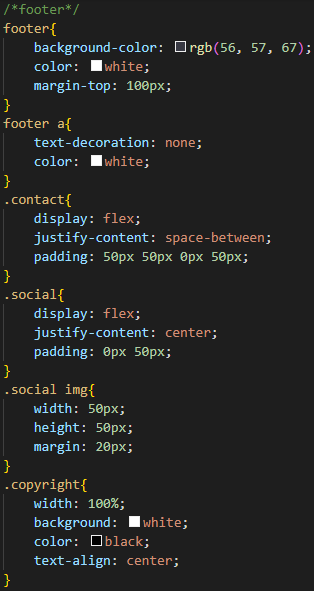


Рис. 8.21 «Код для создания стилей подвала главной страницы web-приложения»

На странице web-приложения с информацией о автосервисе «AutoHelp» для создания блока «Информация о нас» изображенного на рисунке 8.22, использовался код для создания структуры блока, изображенный на рисунке 8.23 и код для создания стилей блока, изображенный на рисунке 8.24.



Рис. 8.22 «Блок «Информация о нас» web-приложения»

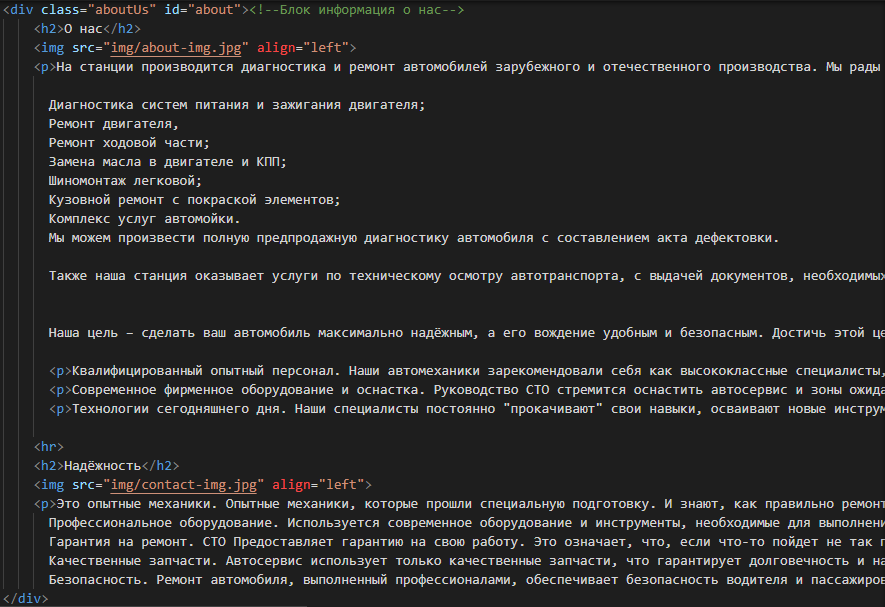


Рис. 8.23 «Код для создания структуры блока «Информация о нас» web-приложения»

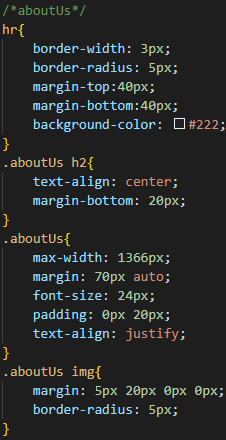


Рис. 8.24 «Код для создания стилей блока «Информация о нас» web-приложения»

На странице web-приложения с услугами автосервиса «AutoHelp» для создания блока «Информация об услугах» изображенного на рисунке 8.25, использовался код для создания структуры блока, изображенный на рисунке 8.26 и код для создания стилей блока, изображенный на рисунке 8.27.

Рис. 8.25 «Блок «Информация об услугах» web-приложения»

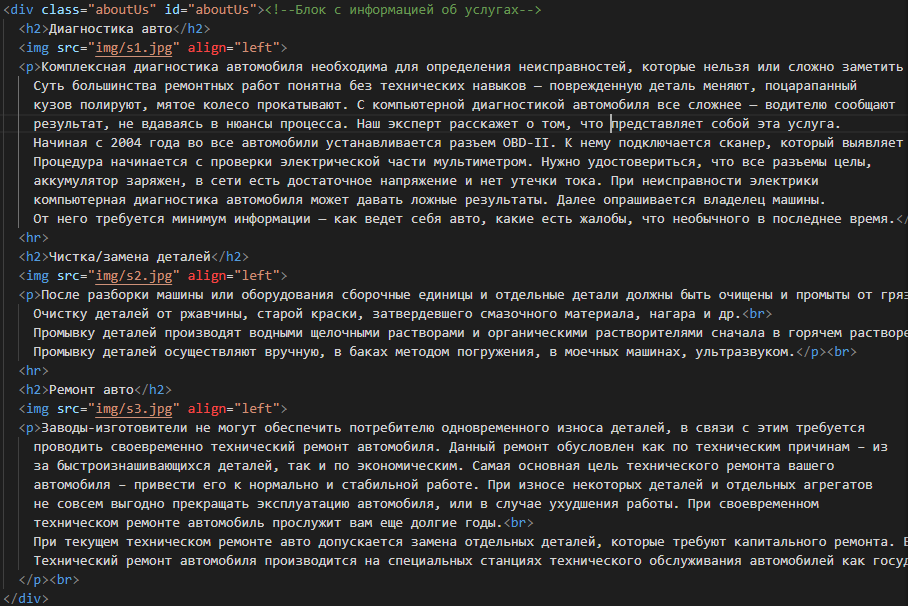


Рис. 8.26 «Код для создания структуры блока «Информация об услугах» web-приложения»

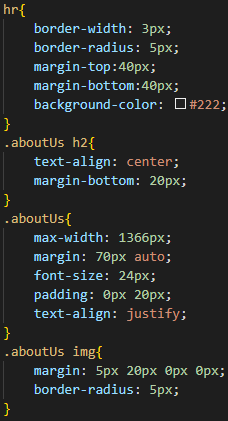


Рис. 8.27 «Код для создания стилей блока «Информация об услугах» web-приложения»

На странице web-приложения с услугами автосервиса «AutoHelp» для создания блока «Прайс услуг» изображенного на рисунке 8.28, использовался код для создания структуры блока, изображенный на рисунке 8.29 и код для создания стилей блока, изображенный на рисунке 8.30.

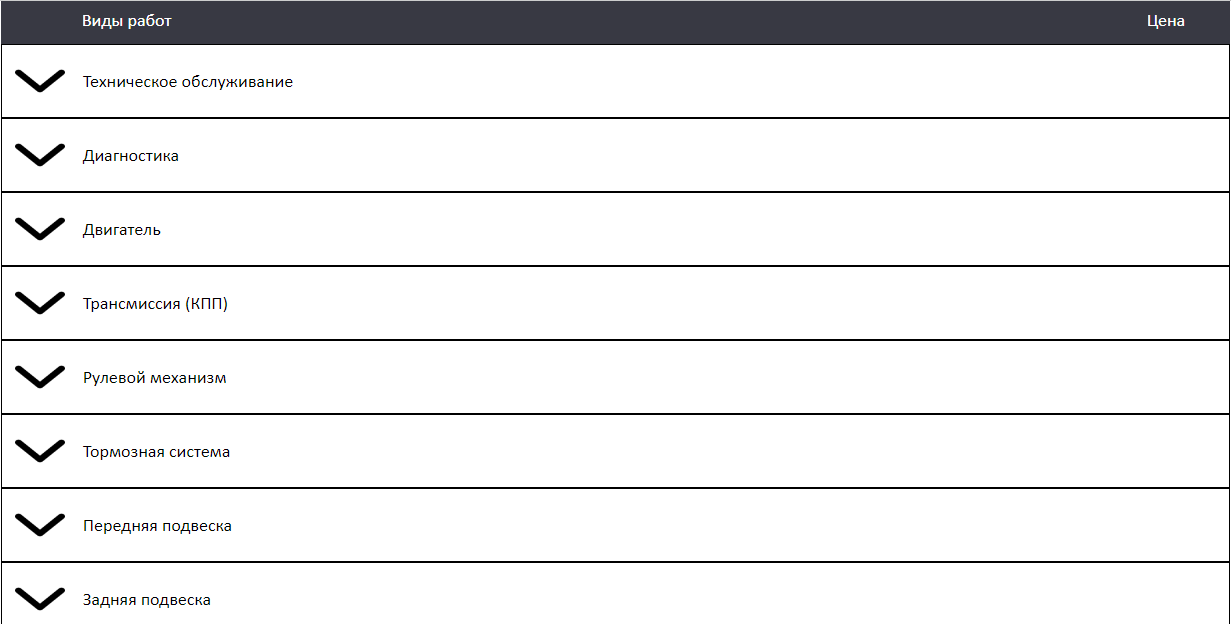
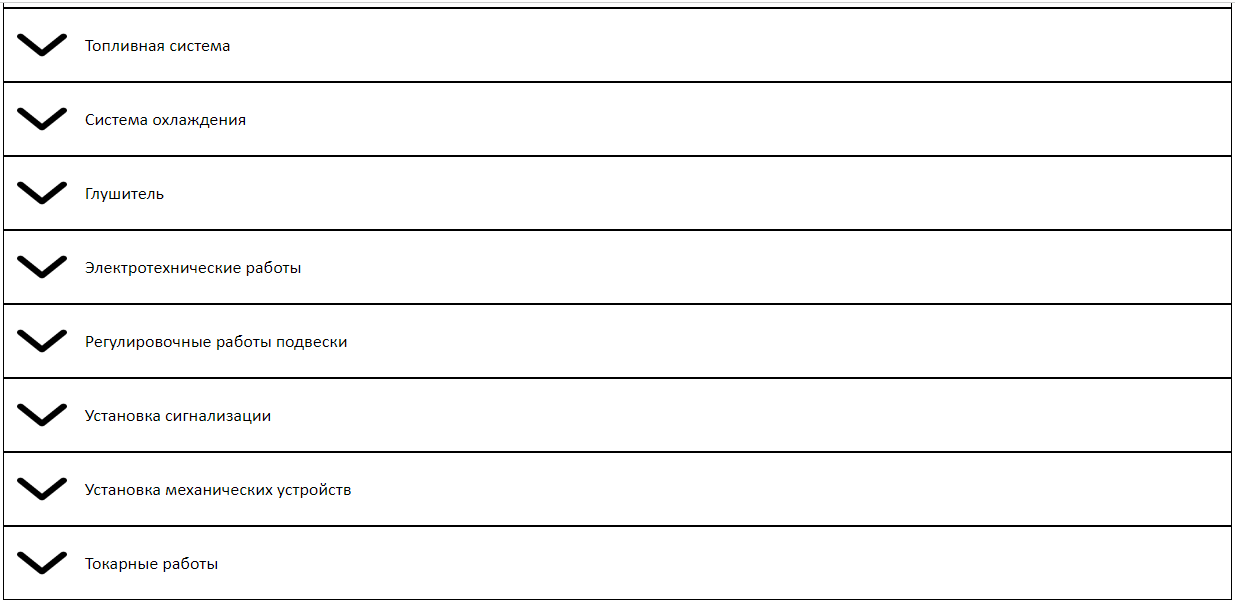
 

Рис. 8.28 «Блок «Прайс услуг» web-приложения»

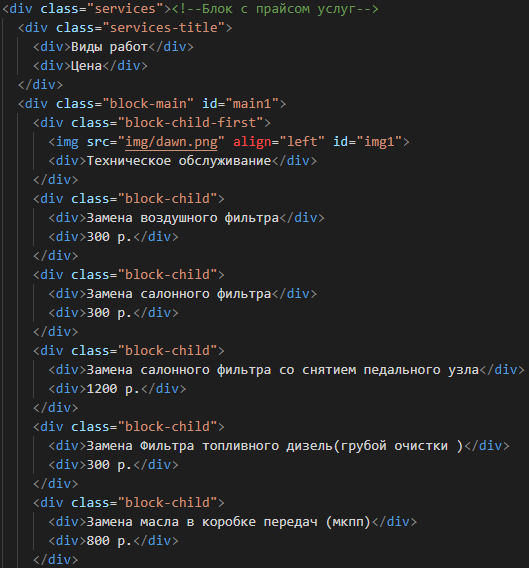


Рис. 8.29 «Код для создания структуры блока «Прайс услуг» web-приложения»

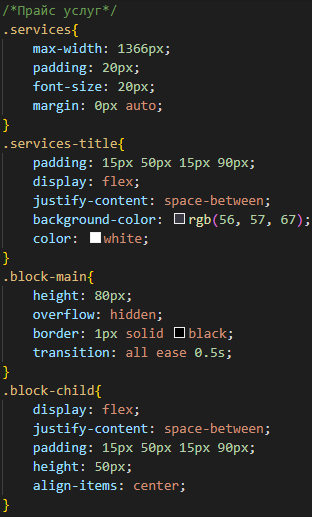
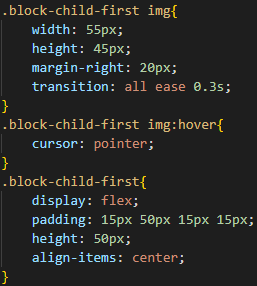
 

Рис. 8.30 «Код для создания стилей блока «Прайс услуг» web-приложения»

# Заключение

В данном курсовом проекте проведён анализ предметной области, разработана информационная структура web-приложения, а также макет дизайна. В итоге разработано web-приложение, который позволит посетителям онлайн записаться на ремонт авто.

Преимущества данного web-приложения: интуитивно понятный дизайн и небольшая цена разработки, для пользователя использование такого web-приложения будет максимально удобным и понятным, в результате чего трафик посещения сайта будет высоким.

Сайт поможет автосервису расширить клиентскую базу. Наличие онлайн-каталога услуг и цен в web-приложении позволит покупателям сделать выбор в пользу Вашего автосервиса. Клиенты могут изучить услуги и цены на без посещения автосервиса, а автосервис, в свою очередь, сократит время на консультации.

Наличие качественного и удобного сайта позволит повысить имидж автосервису, профессионально выглядеть на фоне конкурентов и укрепить доверие клиентов.

# Список литературы

Основная

1. Гниденко, И. Г.  Технология разработки программного обеспечения: учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва: Издательство Юрайт, 2020. — 235 с. — (Профессиональное образование). — ISBN 978-5-534-05047-9. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/453640>
2. Сысолетин, Е. Г.  Разработка интернет-приложений: учебное пособие для среднего профессионального образования / Е. Г. Сысолетин, С. Д. Ростунцев. — Москва: Издательство Юрайт, 2020. — 90 с. — (Профессиональное образование). — ISBN 978-5-534-10015-0. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/456393>

Дополнительная

1. Алексеев А. Введение в веб-дизайн. Учебное пособие. — М.: ДМК Пресс, 2019. — 184 c.
2. [Белов В. В.](http://www.academia-moscow.ru/authors/detail/47801/) , [Чистякова В. И.](http://www.academia-moscow.ru/authors/detail/47802/) [Проектирование информационных систем](http://www.academia-moscow.ru/catalogue/4889/168066/). - Под редакцией: [Белов В. В.](http://www.academia-moscow.ru/authors/detail/47801/) - М.: Академия, 2019
3. Введение в программные системы и их разработку / С.В. Назаров, С.Н. Белоусова, И.А. Бессонова и др. - 2-е изд., испр. - Москва: Национальный Открытый Университет «ИНТУИТ», 2019. - 650 с.: схем., табл., ил. - Библиогр. в кн.;
4. Гарретт Джесс. Веб-дизайн. Элементы опыта взаимодействия. — М.: Символ-Плюс, 2020. — 285 c.
5. Груздев Д.В. Практика ЭВМ-HTML, 3 курс, 2017
6. Дакетт Д. Основы веб-программирования с использованием HTML, XHTML и CSS. – Эксмо, 2018.
7. Долженко, А.И. Технологии командной разработки программного обеспечения информационных систем / А.И. Долженко. - 2-е изд., исправ. - Москва: Национальный Открытый Университет «ИНТУИТ», 2019. - 301 с.: схем., ил. - Библиогр. в кн.;
8. Дэвид Макфарланд. Новая большая книга CSS. — М.: Питер, 2018. — 720 c.
9. Сычев, А.В. Перспективные технологии и языки веб-разработки / А.В. Сычев. - 2-е изд., испр. - Москва: Национальный Открытый Университет «ИНТУИТ», 2019. - 494 с.: ил. ;

Интернет-ресурсы:

1. <https://www.unisender.com/ru/glossary/chto-takoe-yuzabiliti-sajta-osnovnye-principy/>
2. <https://www.unisender.com/ru/blog/sovety/chto-takoe-maket-sayta-i-kak-ego-sozdat/>
3. <https://www.idbi.ru/blogs/blog/osnovnye-pravila-veb-dizayna>
4. <https://mentor.su/struktura-sayta-chto-takoe-kakie-byvayut-i-kak-pravilno-sozdat/>
5. <https://timeweb.com/ru/community/articles/kak-sozdat-dizayn-sayta-gayd-dlya-novichka>