**МИНИСТЕРСТВО ОБРАЗОВАНИЯ МОСКОВСКОЙ ОБЛАСТИ**

**Государственное бюджетное профессиональное образовательное  
учреждение Московской области  
«Авиационный техникум имени В.А. Казакова»**

ПЦК Профессионального цикла

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Программист

КУРСОВОЙ ПРОЕКТ (РАБОТА)

по междисциплинарному курсу  
МДК.01.01. Разработка программных модулей

(полное наименование МДК)

на тему: «Разработка backend для веб-конференций»

Специальность 09.02.07 Информационные системы и программирование

Выполнил  
Обучающийся 4 курса

группы ИП-1 Кирилов Назари Андреевич

преподаватель Громов Виталий Каприянович

*(подпись)* (Фамилия Имя Отчество)

Курсовая работа защищена с оценкой

(оценка прописью)

г. Жуковский, 2024г.

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ МОСКОВСКОЙ ОБЛАСТИ**

**Государственное бюджетное профессиональное образовательное  
учреждение Московской области  
«Авиационный техникум имени В.А. Казакова»**

ПЦК Профессионального цикла

Специальность 09.02.07 Информационные системы и программирование

Квалификация Программист

УТВЕРЖДАЮ «\_\_» 20\_\_ г.

/ /

(дата) (подпись)

**ЗАДАНИЕ НА ВЫПОЛНЕНИЕ КУРСОВОГО ПРОЕКТА (РАБОТЫ)  
по междисциплинарному курсу**

**МДК.01.01. Разработка программных модулей**

***Студентке(ту)* Кирилову Назари Андреевичу**

Специальность **09.02.07 Информационные системы и программирование**

Группа ИП-1 курс 4

1. Тема Разработка backend для веб-конференций
2. Дата выполнения курсового проекта (работы) «30» ноября 2024 г.
3. Содержание курсового проекта (работы):

**Цель:** Создание backend для проекта по веб-конференциям

преподаватель Громов Виталий Каприянович

*(подпись)* (Фамилия Имя Отчество)

*Задание принято к исполнению* «30» ноября 2024 г.

(подпись обучающегося)

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](#_Toc184132821)

[Глава 1. Теоретическая часть 5](#_Toc184132822)

[1.1. Введение в Frontend-разработку 5](#_Toc184132823)

[1.2. Основы веб-дизайна 6](#_Toc184132824)

[1.3. Адаптивный веб-дизайн 7](#_Toc184132825)

[Глава 2. Практическая часть 13](#_Toc184132829)

[1.1. Анализ требований 13](#_Toc184132830)

[1.2. Загрузка библиотеки 15](#_Toc184132831)

[1.3. Создание CSS(Styles.css) 19](#_Toc184132832)

# ВВЕДЕНИЕ

# В современном мире, где цифровые технологии проникают во все сферы жизни, наличие качественного веб-интерфейса становится неотъемлемой частью успешного бизнеса. Особенно это актуально для предприятий, работающих в сфере торговли, таких как мясные лавки. В условиях растущей конкуренции и изменяющихся потребительских предпочтений, создание удобного и функционального веб-интерфейса позволяет не только привлечь новых клиентов, но и удержать существующих, обеспечивая им комфортный и быстрый доступ к услугам и товарам.

# Цель данной курсовой работы заключается в разработке веб-интерфейса для мясной лавки, который будет отвечать современным требованиям пользователей и обеспечивать эффективное взаимодействие между продавцом и покупателем. В процессе работы будет проведен анализ существующих решений, выявлены ключевые потребности целевой аудитории, а также разработаны рекомендации по созданию интуитивно понятного и привлекательного интерфейса.

# В первой части работы будет рассмотрена теоретическая база, связанная с принципами проектирования веб-интерфейсов, а также исследованы особенности работы мясных лавок и их целевой аудитории. Во второй прототипа веб-интерфейса, его функциональных возможностей и дизайна. В заключении будут подведены итоги работы и даны рекомендации по дальнейшему развитию веб-интерфейса.

# Глава 1. Теоретическая часть

**1.1.Введение в Frontend-разработку**

Frontend-разработка охватывает создание визуальной составляющей веб-приложений и сайтов, с которой взаимодействуют конечные пользователи. Эта сфера включает в себя множество аспектов, таких как проектирование интерфейса, верстка страниц и реализация функционала, доступного пользователям через браузер. Ключевыми технологиями, применяемыми в Frontend-разработке, являются HTML, CSS и JavaScript.

HTML — это язык разметки, который используется для создания структуры веб-страницы. С его помощью формируются заголовки, параграфы, списки, гиперссылки и другие элементы, составляющие основу страницы. Далее идет пример простейшего кода на HTML (Рисунок 1.1.1).

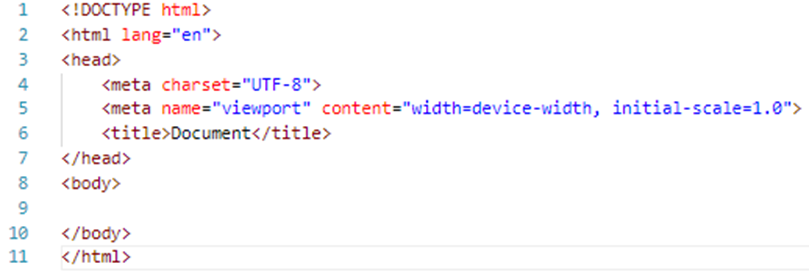


Рисунок 1.1.1 – Пример кода HTML

CSS — это язык стилей, используемый для оформления веб-страниц. Он позволяет задавать такие визуальные параметры, как цвет, шрифт, размеры, отступы и другие характеристики элементов HTML. (Рисунок 1.1.2)

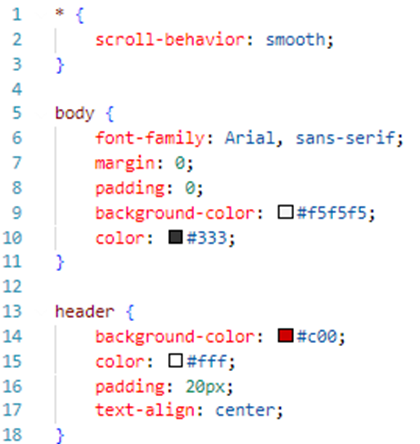


Рисунок 1.1.2 – Пример кода CSS

**1.2.Основы веб-дизайна**

Веб-дизайн играет важную роль в привлечении и удержании пользователей на сайте. Он охватывает такие элементы, как удобство навигации, визуальная иерархия, цветовая палитра и типографика.

Удобство навигации: Это один из ключевых факторов, влияющих на пользовательский опыт. Посетитель должен без труда ориентироваться на сайте, находить необходимую информацию и перемещаться между страницами. Это достигается благодаря логичной структуре меню и хорошо видимым ссылкам и кнопкам.

Визуальная иерархия: Это принцип организации элементов на странице, который помогает акцентировать внимание пользователей на наиболее важных частях контента. Использование различных шрифтов, цветов и размеров позволяет выделить заголовки, кнопки и другие значимые элементы.

Типографика: Это важный аспект веб-дизайна, который влияет на читаемость и восприятие текста. Правильный выбор шрифтов, их размеров и интервалов между строками способствует лучшему восприятию информации.

**1.3.Адаптивный веб-дизайн**

Адаптивный веб-дизайн становится общепринятым стандартом в создании современных сайтов. Этот подход позволяет веб-страницам подстраиваться под различные устройства и разрешения экранов. Он включает в себя использование гибких сеток, медиа-запросов и адаптивной графики.

Гибкие сетки: Они формируют динамические макеты, которые изменяют свои размеры в зависимости от ширины экрана. Это помогает избежать горизонтальной прокрутки и делает сайт более удобным для пользователей.

Медиа-запросы: Это CSS-правила, которые позволяют применять разные стили в зависимости от характеристик устройства, на котором отображается сайт. Они являются ключевым элементом адаптивного дизайна, позволяя изменять внешний вид и структуру сайта для различных экранов.

# Глава 2. Практическая часть

**1.1.Анализ требований**

Общие требования

● Сайт должен быть доступен на всех современных браузерах (Chrome, Firefox, Safari, Edge).

● Должен поддерживать адаптивный дизайн для корректного отображения на мобильных устройствах и планшетах.

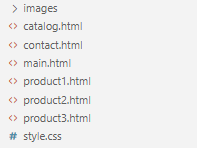
● Время загрузки страниц не должно превышать 3 секунд.

Структура сайта

1. Главная страница
   * популярные товары
2. Страница каталога
   * Категории товаров
   * Элементы навигации
3. Страница товара
   * Изображение товара
   * Описание
   * Кнопка «добавить в корзину»
4. Страница контакты
   * + Контактная информация
     + Форма обратной связи

**1.2.Загрузка библиотеки**

Начнем с создания структуры каталогов и файлов для нашего проекта. Для этого создадим папку проекта с любым именем и следующие подкаталоги и файлы .

****

Начинаем с создания главной страницы index.html, в этом файле мы опишем структуру HTML. Для улучшения читаемости кода для реализации навигации мы применим синематику страницы и импользуем теги <header> для обозночения верхней часчти страницы, так же мы постораемся сделать структуру всех сайтов однотипными чтобы в будующем было легче с стилезацией



После создания главной страницы создаем страницу каталог, на странице каталог у нас будут категории товаров а внутри категорий товаром сами товары, реализованые через <div> , <img> , <a> , <h3> , <p>

****

Следующим шагом создаем страницу контакты. Контакты: это страница на которой будет находится информация о номере телефона потдержки заведения, почта привязаная к заведению, так же форма обратной связи



После создания каталога и категорий внутри него надо заполнить эти категории нужной нам информацией.

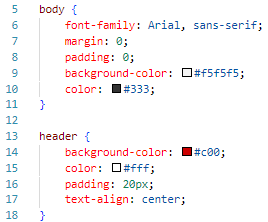
1.Категория – после того как мы заполним эту категорию товаров мы можем точно так же заполнить и другие категории просто изменив названия и фотографии



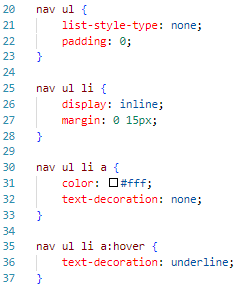
**1.3.Создание CSS(Styles.css)**

Первый шаг перед созданием стилей это создать файл с расширением .css и подключить его к нашему коду, после этого мы можем стилизовать элементы страницы как мы хотим для этого нужна команда <link rel="stylesheet" href="styles.css">.

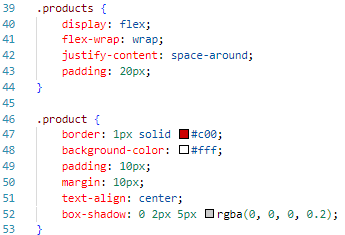
Первым шагом мы стилизуем <body> и <header> так как это первое что бросается пользователю в лицо



Дальше мы стилизуем списки <ul> и <li> а также ссылки <a>,они используются в качестве навигации по нашему сайту

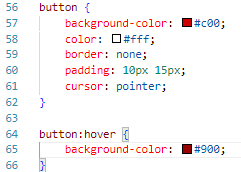


Следующим шагом мы создаем класс .product и .products, таким образом мы сможем придать любому тегу стиль который будет написан в этих классах это нужно для того чтобы стилизовать тег при этом не затронув другой такой же тег

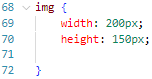


Следующим мы стилизуем кнопки <button> и картинки <img> к <button> мы применяем hover чтобы при наведении на кнопку она выделялась, а для тега <img> мы добавляем ограничение по ширине и высоте чтобы картинки не были на пол страницы

<button>



<img>



# ЗАКЛЮЧЕНИЕ

В ходе разработки web-интерфейса для "Мясной лавки" была создана функциональная и эстетически привлекательная платформа, которая отвечает современным требованиям пользователей и обеспечивает удобство в навигации. Использование только HTML и CSS позволило сосредоточиться на создании чистого и понятного кода, а также на достижении высокой скорости загрузки страниц и адаптивности интерфейса для различных устройств.

Разработанный интерфейс включает в себя все необходимые элементы для успешного функционирования интернет-магазина: от интуитивно понятной навигации и удобного каталога товаров до оформления заказов и обратной связи с клиентами. Применение адаптивного дизайна гарантирует, что пользователи смогут комфортно взаимодействовать с сайтом как на десктопах, так и на мобильных устройствах.

В процессе работы над проектом были учтены принципы юзабилити и современного веб-дизайна, что позволило создать интерфейс, способствующий повышению удовлетворенности клиентов и, как следствие, увеличению продаж.

Таким образом, разработка web-интерфейса для "Мясной лавки" не только продемонстрировала возможности HTML и CSS, но и показала, как с их помощью можно создать качественный продукт, соответствующий требованиям современного рынка. В дальнейшем, для улучшения функциональности и расширения возможностей сайта, можно рассмотреть интеграцию JavaScript и серверных технологий, что позволит реализовать более сложные функции и улучшить взаимодействие с пользователями.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Алан Купер. Дизайн для реального мира - Классическая работа о принципах взаимодействия пользователей с интерфейсами.

2. Дональд Норман. Дизайн повседневных вещей - Книга о том, как дизайн влияет на повседневную жизнь и удобство использования.

3. Илья Бирман. Книга о дизайне - Простой и доступный подход к основам дизайна интерфейсов.

4. Артемий Лебедев. Книга о дизайне - Практические советы и размышления о дизайне от известного российского дизайнера.

5. Стив Круг. Не заставляйте меня думать! - Книга о юзабилити и веб-дизайне, переведенная на русский язык.

6. Аарон Уолтер. Designing for Emotion - Работа о том, как эмоции влияют на восприятие дизайна.

7. Полин Браун. Эстетический интеллект - Книга о важности эстетики в бизнесе и дизайне.

8. Илья Сидоренко. Дизайнер интерфейсов - Личный опыт и советы по созданию интерфейсов.

9. Майкл Микалко. Креативность на заказ - Упражнения и методы для развития креативного мышления в дизайне.

10. W3Schools. (n.d.). HTML Tutorial.https://www.w3schools.com/html/

11. W3Schools. (n.d.). CSS Tutorial. https://www.w3schools.com/css/

12. MDN Web Docs. (n.d.). HTML: HyperText Markup Language. https://developer.mozilla.org/en-US/docs/Web/HTML

13. MDN Web Docs. (n.d.). CSS: Cascading Style Sheets. https://developer.mozilla.org/en-US/docs/Web/CSS