**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)

Оглавление

[ВВЕДЕНИЕ 4](#_Toc184984014)

[Глава 1. Теоретическая часть 5](#_Toc184984015)

[1.1. Введение в Frontend-разработку 5](#_Toc184984016)

[1.2. Основы веб-дизайна 6](#_Toc184984017)

[1.3. Адаптивный веб-дизайн 7](#_Toc184984018)

[Глава 2. Практическая часть 8](#_Toc184984019)

[2.1. Анализ требований 8](#_Toc184984020)

[2.2. Загрузка библиотеки 8](#_Toc184984021)

[2.3. Создание стилей (styles.css) 13](#_Toc184984022)

[ЗАКЛЮЧЕНИЕ 18](#_Toc184984023)

# ВВЕДЕНИЕ

В современном мире музыка становится неотъемлемой частью жизни миллионов людей. Успех музыкальной платформы зависит от удобного интерфейса, простой навигации и визуальной привлекательности, позволяющей пользователям легко находить информацию об артистах и их релизах. Именно на этих аспектах построена концепция нашего сайта.

Проект ориентирован на создание информационного портала, посвященного музыкантам и их творчеству. Особое внимание уделено разработке пользовательского интерфейса, который включает структуру и стилизацию страниц, удобную навигацию и интерактивные элементы, упрощающие доступ к информации.

Особое внимание уделяется адаптивности дизайна, чтобы сайт был удобен на различных устройствах, от смартфонов до компьютеров. Для этого применяются современные технологии, такие как HTML, CSS и медиазапросы. Использование библиотек, например Google Fonts, обеспечивает гармоничное визуальное оформление и комфортное восприятие текста.

Основная цель проекта – создать удобный и функциональный ресурс для всех любителей музыки, предоставляющий актуальную и полную информацию о музыкантах и их релизах. Наш сайт станет местом, где каждый сможет легко найти интересующего артиста или открыть для себя новую музыку.

# Глава 1. Теоретическая часть

## Введение в Frontend-разработку

Frontend-разработка охватывает создание визуальной части веб-сайтов и приложений, с которой взаимодействуют пользователи. Эта сфера включает такие аспекты, как проектирование интерфейса, верстка страниц и реализация функциональных возможностей, доступных через браузер. Основу Frontend-разработки составляют такие технологии, как HTML, CSS и JavaScript.

HTML (HyperText Markup Language) служит языком разметки для создания структуры веб-страницы. Он позволяет формировать заголовки, абзацы, списки, ссылки и другие элементы, составляющие основу контента страницы.(Рисунок 1.1.1).



Рисунок 1.1.1 – Пример кода *HTML.*

CSS (Cascading Style Sheets) — это язык, предназначенный для стилизации веб-страниц. Он позволяет настраивать цвет, тип и размер шрифта, отступы, а также другие визуальные характеристики элементов HTML. (Рисунок 1.1.2)

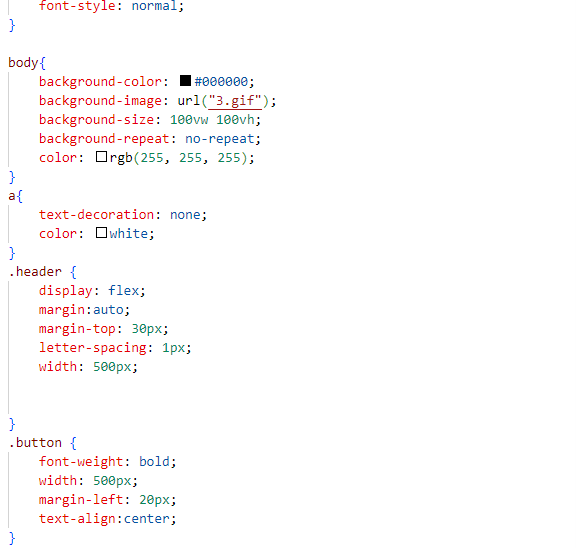


Рисунок 1.1.2 - Пример кода CSS.

Вместе эти технологии формируют полноценный фронтенд, который отвечает за визуальное оформление и функциональность веб-приложений.

## Основы веб-дизайна

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

Удобство навигации: Это один из главных факторов, определяющих качество пользовательского опыта. Посетитель должен без труда ориентироваться на сайте, находить нужную информацию и перемещаться между страницами. Добиться этого можно за счет логичной структуры меню, четко выделенных ссылок и кнопок.

Визуальная иерархия: Принцип размещения элементов на странице, который помогает сфокусировать внимание пользователей на наиболее значимых частях контента. Шрифты, цвета и размеры используются для выделения заголовков, кнопок и других ключевых элементов.

Типографика: Важный компонент веб-дизайна, влияющий на удобство чтения и восприятие текста. Грамотно подобранные шрифты, их размеры и интервалы между строками способствуют улучшению восприятия информации.

## Адаптивный веб-дизайн

Адаптивный веб-дизайн (Responsive Web Design) стал стандартом в разработке современных сайтов. Этот подход обеспечивает возможность веб-страниц автоматически подстраиваться под разные устройства и разрешения экранов. Он включает использование гибких сеток, медиа-запросов и адаптивной графики.

Гибкие сетки: Формируют динамические макеты, которые изменяются в зависимости от размеров экрана. Это исключает необходимость горизонтальной прокрутки и делает использование сайта более комфортным.

Медиа-запросы: CSS-правила, которые позволяют применять различные стили в зависимости от характеристик устройства, отображающего сайт. Этот элемент адаптивного дизайна обеспечивает корректировку оформления и структуры страниц под экраны разных размеров.

# Глава 2. Практическая часть

## Анализ требований

Перед началом разработки сайта о просмотре музыкантов необходимо определить ключевые страницы и их элементы:

1. Главная страница

* Ссылки на дискографию и биографию

1. Страница биографии

* Краткая информация
* Фото артиста

1. Страница дискографии

* Изображение релиза
* Информация о релизе

## Загрузка библиотеки

Сначала создадим структуру каталогов и файлов для проекта. Для этого создадим папку проекта с именем HTML\_HRT и следующие подкаталоги и файлы (Рисунок 2.2.1):

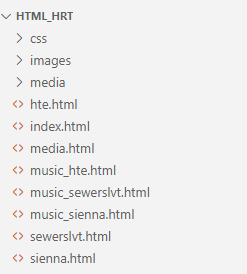


Рисунок 2.2.1 - Подкатологи и файлы проекта

Начнем с создания главной страницы index.html, где будет описана структура HTML. Для повышения читаемости кода мы используем семантическую разметку: элемент <header> обозначит верхнюю часть страницы и будет служить для реализации навигации. (Рисунок 2.2.2)

Рисунок 2.2.2 - Элемент <header> с описанной навигацией.

Для навигации по сайту, используем элемент <header> и <details> и добавим в них ссылки для перемещения по сайту (Рисунок 2.2.2):

После окончания написания главной страницы переходим к написанию страниц биографии.(Рисунок 2.2.3):

Рисунок 2.2.3 - sewerslvt.html

После описания биографии одного артиста, проделываем аналогичную работу с другими.

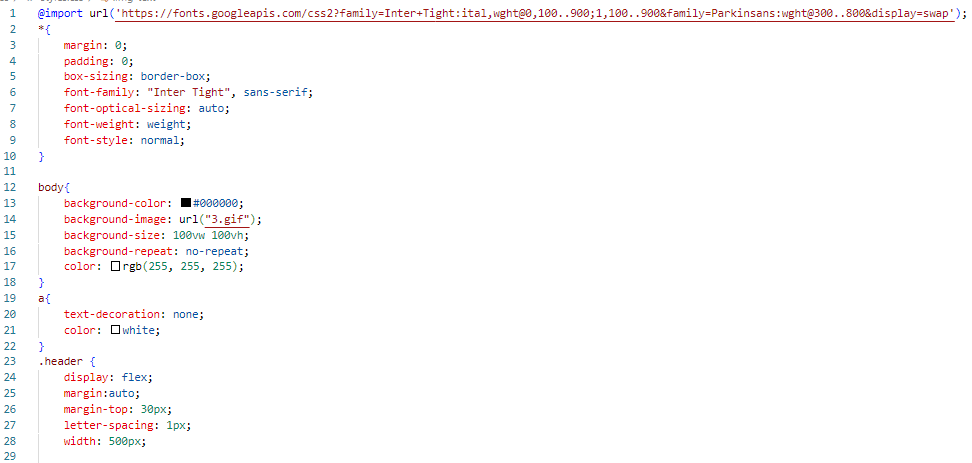
Оформив разделы с биографией, переходим созданию страниц с релизами музыкантов (Рисунок 2.2.4):



Рисунок 2.2.4 - music\_sewerslvt.html

## Создание стилей (styles.css)

Теперь мы создадим файл стилей styles.css, чтобы оформить элементы, которые мы добавили в index.html и других разделах. Добавим ссылку на текст и <header> (Рисунок 2.3.1):

Рисунок 2.3.1 - Оформление <header> и текста.

Так же делаем оформление для кнопок на странице (Рисунок 2.3.2):

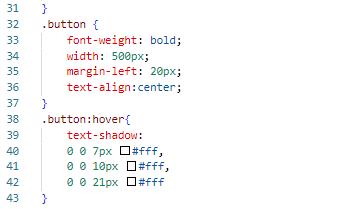


Рисунок 2.3.2 - Оформление кнопок.

Так же оформляем <details> и <summary> (Рисунок 2.3.3):

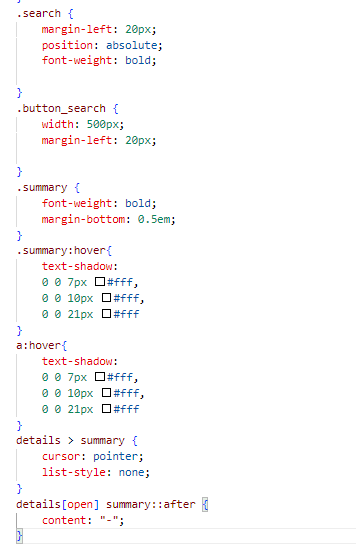


Рисунок 2.3.3 - оформление <details> и <summary>.

После оформленных ссылок, оформляем текст и картинки на странице (Рисунок 2.3.4):

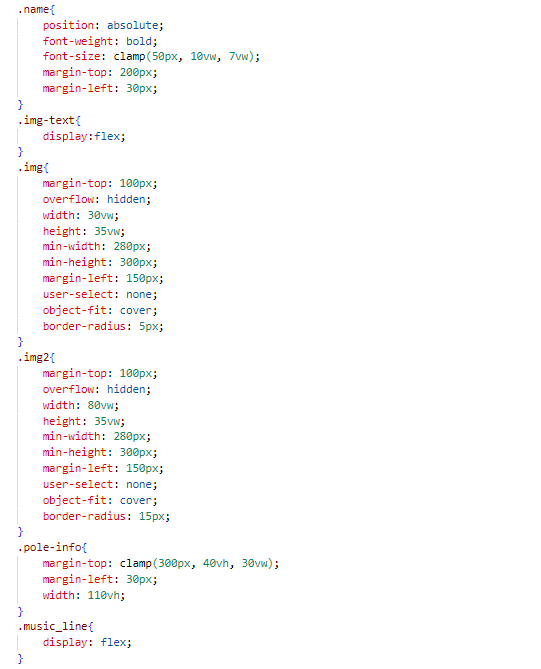


Рисунок 2.3.4 - Оформление текста и картинок

Далее можем перейти к оформлению страницы с дискографией. (Рисунок 2.3.5):

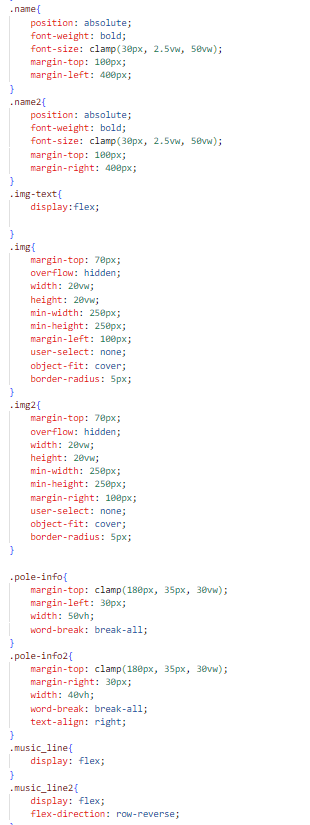


Рисунок 2.3.5 - Оформление страницы с дискографией

Так же добавляем стилизацию <footer> на этой странице. (Рисунок 2.3.6):

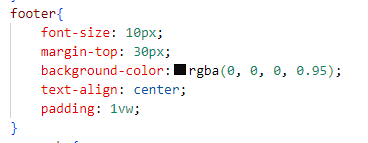


Рисунок 2.3.6 - Оформление <footer>

На данном этапе все страницы созданы и оформлены. Теперь добавляем фотографии к релизам артистов и тестируем сайт на различных устройствах (мобильных и настольных), чтобы проверить его адаптивность. Также важно убедиться, что все ссылки на страницах работают правильно и ведут на соответствующие разделы.

В результате нашей работы мы разработали сайт с информацией о музыкантах, используя только HTML и CSS. В процессе создания были реализованы основные страницы и компоненты, такие как хедер, футер, релизы и страница с биографиями музыкантов.

Этот проект можно расширить, добавив дополнительные функции, такие как JavaScript для удобного управления контентом, а также серверные технологии для обработки данных и взаимодействия с базой данных. Тем не менее, базовая структура и оформление уже готовы к использованию и демонстрации.

# ЗАКЛЮЧЕНИЕ

В процессе выполнения данного проекта по созданию сайта с информацией о музыкантах был реализован полный цикл разработки, начиная с анализа требований и проектирования структуры, и заканчивая реализацией и тестированием интерфейса. В проекте активно использовались HTML и CSS, что позволило создать интуитивно понятный и визуально привлекательный интерфейс.

Основные компоненты, такие как хедер, навигационное меню, карточки артистов и страница с биографиями, были разработаны с учетом современных стандартов веб-дизайна. Благодаря применению медиа-запросов удалось обеспечить адаптивность страниц, что критически важно для удобства пользователей на различных устройствах. Каждый элемент интерфейса был тщательно спроектирован, чтобы предоставить пользователям легкий доступ ко всем функциям сайта.

Кроме того, в процессе работы над проектом были изучены успешные практики фронтенд-разработки, такие как использование семантической разметки, структурирование контента и организация навигации. Эти аспекты способствовали созданию более доступного и удобного сайта для пользователей с разными потребностями.

Полученные результаты не только продемонстрировали основные принципы работы с HTML и CSS, но и стали основой для дальнейшего изучения более сложных технологий и подходов в веб-разработке, таких как JavaScript для динамического взаимодействия и различные фреймворки для создания клиентских приложений.

В заключение, разработка сайта с информацией о музыкантах подтвердила важность комплексного подхода, включающего проектирование, верстку и стилизацию, что в итоге обеспечивает не только функциональность, но и высокую эстетическую ценность веб-приложения.

Проект служит отличной основой для дальнейшего расширения функциональности и улучшения пользовательского опыта.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. **Кастеллос, М. А.** (2019). HTML и CSS: Разработка и дизайн веб-сайтов. — Издательство "Питер". — 640 с.  
   - Эта книга предоставляет все необходимое для освоения веб-разработки с использованием HTML и CSS, включая множество примеров и практических заданий.  
  
2. **Фриман, Э., Робин, Э.** (2020). Динамический веб: HTML и CSS в действии. — Издательство "Вильямс". — 672 с.  
   - Книга охватывает как основы HTML и CSS, так и более сложные темы, включая адаптивный и семантический веб-дизайн.  
  
3. **Рэйчел Анджел, Д.** (2016). HTML и CSS: Краткое руководство. — Издательство "Олимп-Бизнес". — 320 с.  
   - Практическое руководство для начинающих, в котором подробно описаны основные концепции и элементы HTML и CSS.  
  
4. **Сполдинг, Р.** (2018). CSS: The Definitive Guide. — O'Reilly Media. — 700 с.  
   - Всеобъемлющая книга по CSS, которая охватывает различные аспекты стилизации веб-документов.  
  
5. **Моррисон, Э.** (2021). Веб-дизайн и разработка: Полное руководство по HTML и CSS. — Издательство "КУДИЦ-Пресс". — 456 с.  
   - Учебное пособие, которое включает в себя все аспекты веб-дизайна и разработки, делая акцент на современных подходах к стилизации и верстке.  
  
6. **Максвелл, Т.** (2020). Responsive Web Design with HTML5 and CSS. — Packt Publishing. — 588 с.  
   - Книга посвящена созданию адаптивного дизайна с использованием HTML5 и CSS, что является критически важным для современных веб-приложений.  
  
7. **Моррис, А.** (2019). HTML, XHTML и CSS: дизайн и программирование веб-страниц. — Издательство "БХВ-Петербург". — 560 с.  
   - Хороший ресурс для изучения web-разработки, включая практические примеры и задания.  
  
8. **Карасев, А. В.** (2018). Веб-разработка: Учебное пособие по HTML, CSS и JavaScript. — Издательство "ДЛА". — 512 с.  
   - Учебное пособие обеспечивает подробное введение в веб-разработку с акцентом на использование HTML и CSS.