Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Новгородский государственный университет имени Ярослава Мудрого»

ИНСТИТУТ ЭЛЕКТРОННЫХ И ИНФОРМАЦИОННЫХ СИСТЕМ

**ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ**

РАзработка веб-сайта для прослушивания музыкальных композиций

Курсовой проект по учебной дисциплине

«Технология разработки программного обеспечения»

по специальности 09.02.03 Программирование в компьютерных системах

ПТК.КП 2993 13.000ПЗ

|  |
| --- |
| Руководители  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ФИО  (подпись)  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 года |
| Студент группы 2993  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ФИО  (подпись)  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 года |

Аннотация

Данный проект представляет собой разработку веб-приложения, которое предлагает пользователям возможность прослушивать песни, а также расширенное использование сервиса для композиторов. Пользователи могут проигрывать аудио файлы композиций, следить за выходом треков разных исполнителей. Поддерживать определенных исполнителей через подписку. Для создания сайта используются HTML для структурирования контента, CSS для стилизации и JavaScript для реализации функциональной части сайта, взаимодействия с локальным хранилищем и работы с аудио файлами.

Введение

В современном мире, где музыка играет не последнюю роль в жизни человека, в независимости от любимого жанра отдельно взятого человека. Композиторы вкладывают в свои произведения много эмоций, которые отражаются в слушателях. Но, к сожалению, не все могут сочинять новое творчество, так как это требует не только навыков, а ещё многих вложений и хороших специалистов в других областях музыкального мастерства.

Наличие удобной платформы, как для пользователя, так и для композитора становится с каждым годом актуальнее. Настроение, новые идеи, новые желания и свершения в жизни часто зависит от музыки. С точки зрения этих фактов разработка удовлетворяющего приложения для прослушивания и опубликования музыки, является приоритетом развития данного направления.

Целевой аудиторией данного приложения являются люди, которым необходима возможность из любой точки пространства запустить приложение для прослушивания песни. Это могут быть путешественники, телекоммуникационные службы, спортсмены, а также обычные люди, для которых погодные данные важны в повседневной жизни. В основную часть аудитории так же входят музыкальные артисты, которым в свою очередь, необходимо контролировать отчетность прослушиваний, создание качественного материала формата “.mp3” и правильного продвижения. Разработанное веб-приложение предоставляет им удобный и эффективный способ прослушивания или создания музыки.

В рамках данного проекта будет разработано веб-приложение, позволяющее пользователям не только прослушивать музыку, но и приобретать ее, полностью следить за деятельностью музыканта в границах эстрады и другое. В связи с одной из главных проблем – безопасностью в интернете, сайт предоставляет как композитору, так и слушателю возможность контроля монетизации песен. Слушатель может выбирать кому из музыкантов, которых пользователь слушает, будет отправлены деньги, полученные с подписки на платформу. Композиторы в свою очередь, смогут полностью отслеживать поток монетизации в любое время, без третьего юридического лица.

Это сделает мир музыкального творчества ближе к широкой аудитории, что не только привлечёт много новых пользователей на платформу, но и увеличит количество хороших музыкальных произведений. В целом создание сайта с такими возможностями для стороны слушателей и стороны исполнителей является необходимым переходом на новую ступень удобства в данной сфере развлечений и улучшения качества отечественной музыки.

Для достижения поставленной цели будут решены следующие задачи:

1. Анализ предметной области и требований к приложению;
2. Выбор технологий и инструментов разработки;
3. Проектирование архитектуры и функциональности приложения;
4. Разработка макета для выявления самых удобных комбинаций интерфейса;
5. Разработка пользовательского интерфейса;
6. Тестирование и отладка приложения;
7. Развертывание и публикация приложения.

Раздел 1. Общая (теоретическая) часть

1.1 Постановка задачи (ТЗ)

1.1.1 Характеристики бизнес-процессов

Бизнес-процесс - совокупность взаимосвязанных мероприятий или работ, направленных на создание определенного продукта или услуги для потребителей.

Основными бизнес-процессами, связанными с разработкой приложения для прослушивания музыки, являются работа с хранилищем аудио файлов из надежных источников, а также их воспроизведение в формате удобного интерфейса. Эти процессы должны выполняться в режиме реального времени.

Для приложения музыкальной платформы были выявлены следующие бизнес-процессы:

1. Получение данных об композиции из объекта, являющимся хранилищем, с возможностью изменения;
2. Обработка и анализ композиции: процесс запроса к собранным ранее данным в удобный для работы и вывода формат;
3. Воспроизведение музыки на основе функции, на основе полученных данных при обращении к объекту с данными;
4. Предоставление изменения воспроизведения пользователям: процесс визуализации кнопок изменения и повторного запроса к алгоритму воспроизведения.

1.1.2 Технико-математические описание задачи

Для реализации были использованы следующие средства:

HTML - стандартизированный язык разметки документов в Интернете. Большинство веб-страниц содержат описание разметки на языке HTML. Язык HTML интерпретируется браузерами; полученный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного устройства. Язык HTML до 5-й версии определялся как приложение SGML (стандартного обобщенного языка разметки по стандарту ISO 8879). Спецификации HTML5 формулируются в терминах DOM (объектной модели документа). В интернете HTML-страницы, как правило, передаются браузерам от сервера по протоколам HTTP или HTTPS, в виде простого текста или с использованием шифрования.

CSS - формальный язык описания внешнего вида документа, написанного с использованием языка разметки. Преимущественно используется как средство описания, оформления внешнего вида веб-страниц, написанных с помощью языков разметки HTML и XHTML, но может также применяться к любым XML-документам, например, к SVG или XUL.

Протокол передачи данных HTTP (англ. HyperText Transfer Protocol - «протокол передачи гипертекста») - протокол прикладного уровня передачи данных. Основой HTTP является технология «клиент-сервер». Основным назначением HTTP является передача веб-страниц (текстовых файлов с разметкой HTML), хотя с помощью него с успехом передаются и другие файлы, как связанные с веб-страницами (изображения и приложения), так и несвязанные с ними. HTTP предполагает, что клиентская программа - веб-обозреватель - способна отображать гипертекстовые веб-страницы и файлы других типов в удобной для пользователя форме.

1.1.3 Требования к программе

Функциональные требования:

* Возможность регистрации пользователя.
* Воспроизведение аудио файлов.
* Переход между страницами сайта.
* Функционирующие и стилизованные модальные окна.

Нефункциональные требования:

* Простой и интуитивно понятный интерфейс.
* Быстрая загрузка и отзывчивость приложения.
* Кроссбраузерная совместимость.

1.2 Обоснование проектных решений

1.2.1 Обоснование выбора языков программирования

Выбор языков программирования зависит от многих факторов, например задачи и функции, которые нужно решить и реализовать. Для разработки музыкального веб-сайта обусловлен следующими факторами:

HTML был выбран для создания структуры и разметки веб-страницы, так как он является стандартом для веб-разработки и обеспечивает семантическую структуру контента.

CSS, используется для стилизации и оформления интерфейса приложения. CSS, является основным, а главное понятным средством стилизации проекта, позволяющий быстро, удобно и понятно создать красивый дизайн.

JavaScript применяется для реализации интерактивного функционала, обработки данных и взаимодействия с локальным хранилищем. Он имеет простой синтаксис и большое сообщество разработчиков, что делает его удобным для изучения и использования. Кроме того, JavaScript имеет множество библиотек и фреймворков, которые упрощают разработку веб-приложений. JavaScript является мощным и гибким языком программирования, широко используемым в веб-разработке.

Таким образом, выбор языков программирования обоснован тем, что они являются популярными и широко используемыми языками программирования в веб-разработке.

1.2.2 Инструментальные средства

Для разработки приложения использовались следующие инструментальные средства:

* Visual Studio Code - кроссплатформенная интегрированная среда разработки (IDE), обеспечивающая удобный редактор кода, отладчик и множество полезных расширений для веб-разработки. Она предоставляет широкий набор инструментов для отладки, рефакторинга кода и автоматизации задач.
* Git - распространенная система контроля версий, позволяющая отслеживать изменения в коде, производить откаты и совместную работу над проектом.
* Figma - это онлайн-редактор дизайна, который позволяет создавать макеты и прототипы веб-сайтов, приложений и других пользовательских интерфейсов.

1.2.3 Обоснование выбора среды программирования

Браузер был выбран в качестве среды выполнения для данного веб-приложения, так как он предоставляет все необходимые возможности для отображения и выполнения HTML, CSS и JavaScript-кода. Современные веб-браузеры поддерживают новейшие стандарты и технологии, что делает их идеальной средой для разработки и запуска веб-приложений.

1.2.4 Информационное обеспечение

Для сайта для прогноза погоды были выявлены следующие элементы информационного обеспечения:

Для разработки логики работы приложения для прослушивания музыкальных файлов используется язык программирования JavaScript.

Для получения первичных данных аудио файлов используется множество интернет-платформ, включая и официальные ресурсы музыкантов. Что в свою очередь обеспечивает надежность и качество данных.

1.3 Обзор существующих программных систем

На рынке существует ряд приложений и веб-сайтов, предназначенных для прогноза погоды. Некоторые из наиболее популярных и широко используемых:

* Яндекс Музыка

Сайт: https://music.yandex.ru/home

* Веб-сайт и мобильное приложение Яндекс Музыка предлагают популярные платформы для прослушивания, создание плейлистов и оповещения. Они известны удобным интерфейсом.
* Spotify

Сайт: https://www.spotify.com/int/why-not-available/

Этот веб-сайт предоставляет очень удобный и интуитивно понятный интерфейс, понятная система монетизации. Главная проблема заключается в том, что Spotify в данный момент не доступен для русской аудитории.

* Vk Music(BOOM)

Сайт: <https://vk.com/boom>

Как такового сайта нету. Данная платформа встроена в сервисы мессенджера Вконтакте и это является ее главным плюсом.

* Apple Music

Сайт: https://music.apple.com/us/browse

Этот веб-сервис и мобильное приложение имеют приятный интерфейс, проработанную систему и качественные функции.

Преимущества разрабатываемого приложения заключаются в его легкости, быстрой загрузке и предлагаемых услуг, что делает его удобным для использования различных пользователей и в условиях медленного интернет-соединения. Кроме того, приложение будет иметь преимущество гибкости и возможности настройки в соответствии с потребностями пользователей.

Раздел 2. Практическая составляющая

2.1 Реализация

2.1.1 Выбор технических и программных средств

Для разработки музыкального приложения использовались следующие технические и программные средства:

* Операционная система: Windows 10;
* Браузер: Mozilla Firefox (для тестирования и отладки);
* IDE: Visual Studio Code;
* Система контроля версий: Git;
* Инструменты разработки: npm;

2.1.2 Практическая реализация алгоритмов

Алгоритм — это точное предписание, описывающее порядок действий (план действий), определяющих процесс решения задачи или достижения цели.

Основным алгоритмом, реализованным в приложении, является проигрывание музыкальной композиции из представленного списка на странице ”Плеер”. Этот процесс включает следующие шаги:

1. Обращение к кнопки получение ее id/class и создание пути до файла;
2. Поиск данного файла в объекте хранилища;
3. Проверка полученных файлов в зависимости от внесенных изменений;
4. Установка изменений полученных, а именно время, с которого нужно начать воспроизведение и уровень громкости выставленный при прошлом использовании пользователем;
5. Изменение времени окончания воспроизведения и уровень громкости;
6. Обработка различных аудиоформатов;
7. Проверка настроенного файла на поддержку аудиоустройств.

2.1.3 Используемые методы

1. Получение аудио файлов: Код использует объекты данных для хранения, получения и изменения музыкальных композиций для определенного автора. При нажатии на определенный блок происходит обращение к массиву объектов, хранящему как сам аудио файл, так и название и продолжительность произведения. Для получения текущих данных для прослушивания песни.
2. Отображение данных: Полученные данные выбранного произведения отображаются на веб-странице в специальном блоке, отведенном под просмотр произведения, играющей последней композиции. Все данные, кроме самого аудио файла, перезаписываются в этом блоке благодаря манипуляций с DOM. Код обновляет различные элементы HTML, такие как название песни, продолжительность произведения, а также самого исполнителя.
3. Настройки приложения: на данном этапе создания проекта настройки приложения, как таковые не предусмотрены. Только изменение пароля пользователя в локальном хранилище. Данные пользователя такие как e-mail, телефон, логин и пароль сохраняются в локальном хранилище браузера.
4. Обработка событий: Код обрабатывает различные события, такие как отправка формы регистрации и отправки формы связи с создателями проекта, нажатие на кнопки товаров и кнопку прослушивания, так же реализован обработчик событий для перехода по блокам главной странице.
5. Дополнительные функции: Код также включает в себя различные вспомогательные функции, такие как создание небольшого аккордеона с кнопками для удобного перемещения по сайту и слайдер картинок, которые являются обложкой музыкального альбома или трека, для приятной презентации нового материала исполнителей.

В целом, код демонстрирует различные аспекты веб-разработки, включая работу с API, манипуляции с DOM, обработку событий, использование локального хранилища и реализацию различных функций для обработки данных и отображения информации на веб-странице.

2.1.4 Используемые методы (примеры)

2.1.5 Реализация интерфейса

2.2 Тестирование

Раздел 3. Практическая составляющая (продолжение)

3.1 Реализация

3.1.1 Архитектура приложения

3.1.2 Обработка данных и генерация прогноза

3.1.3 Оптимизация и производительность

3.2 Тестирование

3.2.1 Методология

3.2.2 Инструменты тестирования

3.3 Руководство оператора

3.3.1 Условия выполнения

3.3.2 Составные части программы и связи между ними

3.3.3 Техника безопасности при работе на компьютере

Заключение

Список литературы

Приложение А

Приложение Б