

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет компьютерных наук

Кафедра программирования и информационных систем

Разработка стримингового музыкального веб-приложения «Musicman»

Курсовая работа

09.03.04 Программная инженерия

Разработка программного обеспечения

Зав. кафедрой _____ С.Д. Махортов, д. ф.-м. н., профессор

Обучающийся _____ А.Р. Сашина, 3 курс, д/о

Обучающийся _____ З.С. Казмиров, 3 курс, д/о

Обучающийся _____ Е.М. Охрямкин, 3 курс, д/о

Руководитель _____ В.С. Тарасов, старший преподаватель

Воронеж 2023

Содержание

Введение.....	3
1 Постановка задачи.....	4
1.1 Постановка задачи	4
1.2 Средства реализации.....	4
2 Анализ предметной области	6
2.1 Терминология (гlossарий).....	6
2.2 Обзор аналогов	6
3 Реализация.....	7
3.1 Реализация логики	7
3.2 Реализация интерфейса	7
Заключение	8
Список использованных источников	9

Введение

В современном мире стриминговые музыкальные сервисы являются неотъемлемой частью нашей жизни и получают все большую популярность. Они предоставляют пользователю удобный доступ к миллионам аудиозаписей в любое время и в любом месте, что позволяет наслаждаться музыкой в комфортных условиях.

Цель данной курсовой работы - разработка веб приложения музыкального стримингового сервиса, позволяющего пользователю в удобной форме наслаждаться любимой музыкой, создавать собственные плейлисты и делиться ими с другими пользователями.

В работе будет рассмотрен процесс разработки приложения, включая его проектирование, функциональность, дизайн и архитектуру системы. А также анализ конкретной среды, исследование потребностей и предпочтений пользователей и определение пути улучшения пользовательского опыта.

Для реализации проекта будут использованы современные технологии и инструменты, необходимые для создания качественного веб приложения. Работа будет состоять из нескольких этапов, включая анализ потребностей пользователей, проектирование интерфейса и архитектуры, разработку функциональности и интеграцию с внешними сервисами, а также тестирование и оптимизацию приложения для улучшения его производительности и удобства использования.

Результатом данной работы будет функциональное и удобное веб приложение музыкального стримингового сервиса, удовлетворяющее потребности пользователей и соответствующее современным требованиям в области веб разработки.

1 Постановка задачи

1.1 Постановка задачи

Стриминговый сервис предназначен для прослушивания песен с возможностью настройки приложения под нужды пользователя.

Основными целями создания системы являются:

- возможность регулировать настройки прослушивания песен;
- показ песен пользователю на основе его музыкальных предпочтений;
- возможность рекомендации песни другим пользователям.

1.2 Средства реализации

Система должна состоять из сервера приложения, реляционной базы данных, клиентской части.

Основной используемый стек технологий:

Back-end (серверная часть):

- Java 17;
- Spring Framework;
- PostgreSQL, Liquibase.

Java является кроссплатформенным языком, т.е. для запуска достаточно иметь Java Virtual Machine. Самым популярным фреймворком для Java является Spring. Основным преимуществом является огромное наличие компонент и внутренних библиотек, которые уже реализованы, а значит позволяет быстро и качественно написать код.

В качестве базы данных была выбрана Postgres, т.к. умеет работать с различными типами данных и позволяет ускорять запросы с помощью индексов. Для управления и применения изменений в базу данных будет использоваться библиотека Liquibase. Основным преимуществом является поддержка написания миграционных файлов в виде yaml или xml файлов.

Front-end (клиентская часть):

- CSS3 + HTML5;

- React.js, TypeScript;
- Effector;
- Material UI.

CSS3 + HTML5 были выбраны потому, что это самый современный стандарт вёрстки и разметки. Он поддерживается большинством браузеров и предоставляет множество новых свойств, упрощающих разработку.

React является популярным фреймворком для разработки клиентской части. Главным его преимуществом является виртуальная объектная модель документа (DOM), занимающая мало места. А значит позволяет быстрее обновлять страницу с изменениями и повышает производительность приложения.

Язык TypeScript был выбран, потому что это компилируемый в JavaScript код, который нивелирует недостатки слабой типизации JavaScript и берёт множество проверок безопасности кода на себя во время трансляции в JavaScript код.

Effector является стейт-менеджером, который хорошо взаимодействует с React-ом и позволяет легко разделять работу с данными по разным хранилищам (декомпозиция). Material UI является библиотекой готовых компонент для React, которые обладают приятным дизайном.

2 Анализ предметной области

2.1 Терминология (гlossарий)

Администратор сайта – специалист, осуществляющий информационную поддержку сайта, управление контентом.

Веб-браузер (браузер) - клиентская программа, поставляемая третьими сторонами и позволяющая просматривать содержимое веб-страниц.

Контент – совокупность информационного наполнения веб-сайта.

Неавторизованный пользователь (гость) – человек, который может авторизоваться в системе, если был зарегистрирован ранее, или пройти регистрацию.

Пользователь – человек, который зарегистрирован в системе и имеет доступ к личному кабинету и основному функционалу системы.

Система администрирования – закрытая от посетителей часть сайта. Управляется администратором.

Эквалайзер – программа, позволяющая регулировать громкость отдельных зон частотного диапазона и выравнивать амплитудно-частотную характеристику звукового сигнала.

Плейлист – подборка аудиоконтента.

2.2 Обзор аналогов

3 Реализация

3.1 Реализация логики

3.2 Реализация интерфейса

Заключение

Список использованных источников

1. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы ГОСТ 34.602-2020 — Взамен ГОСТ 34.602-89; введен 01.01.2020
2. Официальный сайт Spring: сайт. — URL: <https://spring.io/> (дата обращения: 21.03.2023). — Текст: электронный.
3. Официальный сайт React.js: сайт. — URL: <https://react.dev/> (дата обращения — 21.03.2023). — Текст: электронный.
4. Официальный сайт PostgreSQL: сайт. — URL: <https://www.postgresql.org/> (дата обращения: 21.03.2023). — Текст: электронный.
5. Официальный сайт Liquibase: сайт. — URL: <https://www.liquibase.org/> (дата обращения: 21.03.2023). — Текст: электронный.
6. Официальный сайт Effector: сайт. — URL: <https://effector.dev/> (дата обращения: 21.03.2023). — Текст: электронный.
7. Официальный сайт Material UI сайт. — URL: <https://mui.com/> (дата обращения: 21.03.2023). — Текст: электронный.