Техническое задание на разработку веб-приложения «Аудиохостинг»

1. Введение

1.1 Цель документа

Этот документ описывает требования к разработке веб-приложения «Аудиохостинг» для повышения удобства поиска музыки под свои предпочтения и ее прослушивания без скачивания с возможностью поделиться впечатлениями с другими слушателями.

1.2 Актуальность и назначение проекта

У слушателей часто возникают сложности в поиске подходящей музыки без ее скачивания, а у исполнителей возникают сложности с ведении обратной связи со слушателями. Основной целью системы является автоматизация этих процессов для повышения удобства пользователей.

1.3 Основные пользователи системы

Система предназначена для использования такими группами пользователей:

- слушатели (поиск музыки, обмен сообщениями);

- исполнители (загрузка музыки);

- администратор (удаление музыки).

2. Цели и задачи проекта

2.1 Основные цели

- повышение удобства слушателей в поиске подходящей музыки без ее скачивания;

- повышение удобства исполнителей в работе с аудиозаписями.

2.2 Конкретные задачи

- разработка функциональности для создания, редактирования и удаления треков;

- организация системы фильтрации, сортировки и поиска треков;

- организация системы пагинации треков;

- реализация функциональности обмена сообщениями между пользователями;

- разработка функции прослушивания треков без скачивания;

- реализация скачивания треков;

- реализация функциональности добавления и удаления друзей для обмена сообщениями;

- реализация функциональности сохранений треков у пользователя.

3. Требования к системе

3.1 Функциональные требования

- управление треками: возможность создания, редактирования и удаления;

- модерация треков: возможность прослушивания треков и удаления не прошедших модерацию.

- прослушивание треков: пользователь может ставить трек на паузу или возобновить прослушивание.

- фильтрация: пользователь может отфильтровать треки по исполнителям;

- поиск: пользователь может искать треки по названию;

- сортировка: пользователь может отсортировать треки по новизне;

- пагинация: данные разделены на страницы;

- обмен сообщениями: пользователь может отправить сообщение, прочитать историю сообщений;

- статистика для исполнителей: исполнитель может просмотреть количество сохранений для каждого своего трека;

- сохранения треков: пользователь может сохранить трек в «Мою музыку» и удалить его из «Моей музыки»;

- друзья: пользователь может добавлять и удалять друзей.

3.2 Нефункциональные требования

- производительность: отклик интерфейса не более 2 секунд при загрузке задач;

- масштабируемость: возможность расширения на 5000+ пользователей без ухудшения производительности;

- безопасность: поддержка HTTPS и защита данных через SSL.

4. Требования к пользовательскому интерфейсу

4.1 Основные экраны

Главная страница: отображение списка треков с фильтрацией по исполнителю, поиском, сортировкой и пагинацией треков.

Страница «Моя музыка”: отображение списка треков пользователя с фильтрацией по исполнителю, поиском, сортировкой и пагинацией треков.

Страница диалогов: список диалогов с пользователями, которые есть в друзьях, возможность перейти к определенному диалогу и прочитать историю сообщений.

Страница друзей: список друзей пользователя с возможностью удалить друзей.

Страница поиска друзей: список пользователей с возможностью поиска друзей и добавления их к себе в друзья.

Страница личного кабинета: редактирование данных пользователя.

Панель исполнителя: интерфейс для управления треками, а также просмотра статистики.

Панель администратора: интерфейс для просмотра и удаления треков.

4.2 Навигация

Система должна включать меню навигации с разделами: «Моя музыка», «Сообщения», «Главная», «Личный кабинет», «Панель исполнителя», «Панель администратора».

4.3 Юзабилити

Простой и понятный интерфейс для быстрого доступа к трекам.

5. Требования к технической реализации

5.1 Языки и технологии

Backend: Laravel.

Frontend: JavaScript.

База данных: MySQL.

5.2 Архитектура системы

Хранение данных в реляционной базе данных.

Микросервисная архитектура для взаимодействия между модулями.

5.3 Интеграции

Отсутствует.

6. Требования к безопасности

6.1 Аутентификация и авторизация

Аутентификация: валидация данных пользователя.

Авторизация: разграничение прав доступа - администратор, исполнители, слушатели.

6.2 Шифрование данных

HTTPS: для всех передаваемых данных.

Хранение паролей: хэширование через bcrypt.

6.3 Политики доступа

Защита от SQL-инъекций и XSS.

Ограничение по роли для доступа администратора и исполнителей.

7. Ограничения и допущения

7.1 Технические ограничения

Отсутствуют.

7.2 Финансовые ограничения

Отсутствуют.

7.3 Сроки выполнения

Полная реализация проекта — 1 месяц с начала разработки.

8. Требования к тестированию и приемке

8.1 Типы тестирования

Функциональное тестирование: проверка всех реализованных функций.

Нагрузочное тестирование: тестирование на работе с 5000+ пользователями.

Тестирование безопасности: проверка на уязвимости.

8.2 Критерии приемки

Полное выполнение всех функциональных и нефункциональных требований.

Прохождение не менее 95% тестов по результатам приемочных испытаний.

9. Требования к документации

9.1 Пользовательская документация

Руководство пользователя с инструкциями по созданию, редактированию и управлению задачами.

9.2 Техническая документация

Описание архитектуры, структура базы данных.

10. План реализации

10.1 Этапы разработки

Анализ и проектирование — 1 месяц.

Разработка прототипа — 1 месяц.

Полноценная разработка и тестирование — 1 месяц.

Внедрение и обучение — 1 месяц.

10.2 Сроки выполнения этапов

Дата начала: 21 ноября 2024 года.

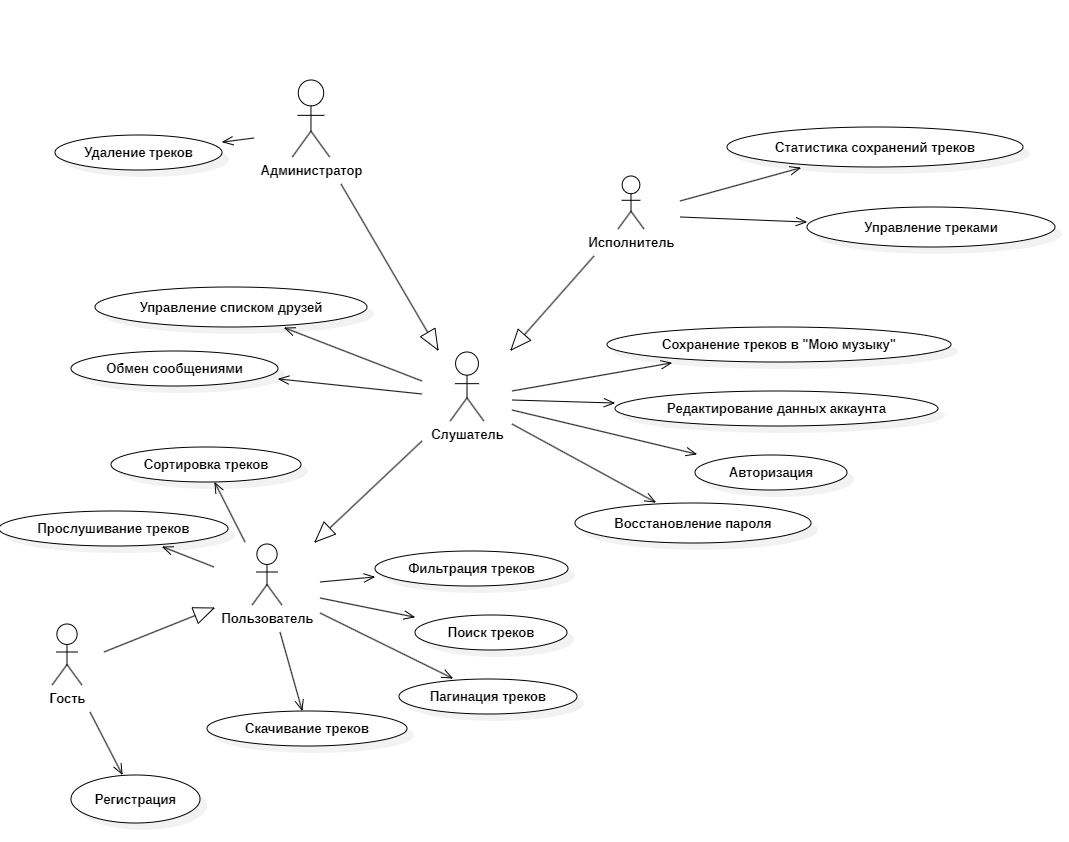
Дата завершения: 21 декабря 2024 года.

10.3 Ответственные лица

Менеджер проекта: Нургалеева Э.Р.

Ведущий разработчик: Нургалеева Э.Р.

**Use Case:**



**Структура базы данных:**

