

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

Факультет компьютерных наук

Кафедра информационных системы и технологии

Разработка музыкального приложения “MusicZone”

Курсовой проект

*09.03.02 Информационные системы и технологии*  
*Программная инженерия в информационных системах*

Зав. кафедрой \_\_\_\_\_ С. Д. Махортов, д.ф.- м.н., доцент  
\_\_.\_.20\_\_

Обучающийся \_\_\_\_\_ Р.С. Шевцов, 3 курс, д/о

Обучающийся \_\_\_\_\_ М.В. Гончаренко, 3 курс, д/о

Обучающийся \_\_\_\_\_ М.С. Артемьев, 3 курс, д/о

Руководитель \_\_\_\_\_ В.С. Тарасов, ст. преподаватель

Руководитель \_\_\_\_\_ И.В. Клейменов, ассистент

Воронеж 2023

## Содержание

Введение .....	4
1 Постановка задачи.....	5
1.1 Цели создания системе.....	5
1.2 Требования к разрабатываемой системе .....	5
1.3 Задачи проекта.....	5
1.3.1 Для не зарегистрированного пользователя .....	5
1.3.2 Для зарегистрированного пользователя .....	5
1.3.3 Для администратора .....	5
2 Анализ предметной области .....	6
2.1 Терминология .....	6
2.2 Обзор аналогов .....	8
2.2.1 Яндекс.Музыка.....	8
2.2.2 YouTubeMusic .....	9
2.2.3 Spotify .....	11
2.3 Моделирование системы.....	12
2.3.1 Диаграмма в стиле методологии IDEF0 .....	12
2.3.2 Диаграмма прецедентов .....	13
2.3.3 Диаграммы классов.....	14
2.3.4 Диаграммы последовательности .....	15
2.3.5 Диаграмма развертывания.....	19
2.3.6 Диаграмма состояния.....	19
2.3.7 Диаграмма объектов .....	19
2.3.8 Диаграммы активности.....	20
2.3.9 Диаграмма сотрудничества .....	21

3 Реализация .....	21
3.1 Средства реализации .....	21
3.2 Реализация базы данных .....	23
3.2.1 ER-диаграмма.....	23
3.2.2 Физическая модель базы данных .....	24
3.3 Реализация клиентской части .....	24
3.3.1 Форма для поиска музыки .....	24
3.3.2 Форма экрана плеера .....	25
3.3.3 Форма экрана авторизации.....	26
3.3.4 Форма экрана регистрации.....	27
3.3.5 Форма экрана восстановления пароля .....	27
3.3.6 Форма экрана личной страницы.....	29
3.3.7 Форма экрана редактирования профиля.....	30
3.3.8 Форма экрана загруженной музыки.....	31
3.3.9 Форма экрана добавления музыки .....	31
3.3.10 Форма экрана редактирования музыки.....	32
3.4 Серверная часть .....	33
3.4.1 Основные положения.....	33
4 Тестирование .....	34
Заключение .....	35

## **Введение**

В современном мире мобильные приложения стали неотъемлемой частью нашей повседневной жизни, предоставляя нам доступ к различным сервисам и удобствам на расстоянии нажатия кнопки. Одной из популярных категорий мобильных приложений являются музыкальные приложения, которые с успехом сочетают в себе развлекательную и практическую функции.

Музыкальные приложения – это программные приложения для мобильных устройств, предназначенные для прослушивания, организации и обмена музыкой. Они позволяют пользователям наслаждаться музыкой в любое время и в любом месте, предлагая широкий спектр функций, включая потоковую передачу музыки, создание плейлистов, поиск новых артистов и треков, радиостанции и многое другое. Благодаря мобильным музыкальным приложениям, музыка стала более доступной и персонализированной, позволяя нам настроиться на нужное настроение или создать собственный саундтрек к повседневным моментам нашей жизни.

В данной курсовой работе рассматривается процесс разработки собственного мобильного музыкального приложения

В рамках работы будут рассмотрены различные аспекты разработки мобильного музыкального приложения, начиная с анализа предметной области, определения его концепции и основных функциональных требований. Затем будет изучено проектирование пользовательского интерфейса и пользовательского опыта, с учетом современных тенденций и личных практик в этой области. Важное внимание будет уделено выбору и интеграции соответствующих технологий и API для обеспечения необходимых функций, таких как передача музыки, поиск треков и другие.

## **1 Постановка задачи**

### **1.1 Цели создания системе**

Целью данной работы является создание мобильного музыкального приложения для поиска и прослушивания музыки.

### **1.2 Требования к разрабатываемой системе**

- Обеспечение авторизации и аутентификации пользователей
- Использование механизмов защиты от SQL-инъекций
- Использование протокола передачи данных HTTP
- Приложение должно быть построено на трехуровневой архитектуре

### **1.3 Задачи проекта**

#### **1.3.1 Для не зарегистрированного пользователя**

- Поиск музыки
- Прослушивания найденной музыки

#### **1.3.2 Для зарегистрированного пользователя**

- Поиск музыки
- Прослушивания найденной музыки
- Добавить музыку на личную страницу

#### **1.3.3 Для администратора**

- Загрузка, редактирование и удаление музыки со стороны администратора приложения
- Поиск музыки
- Прослушивания найденной музыки
- Добавить музыку на личную страницу

## **2 Анализ предметной области**

### **2.1 Терминология**

- Мобильное приложение — программное обеспечение, предназначенное для работы на смартфонах, планшетах и других мобильных устройствах, разработанное для конкретной платформы (iOS, Android, WindowsPhone и т. д.)
- Android-приложение — программное обеспечение, предназначенное для работы на смартфонах, планшетах и других мобильных устройствах, разработанное для платформы Android
- Клиент — это аппаратный или программный компонент вычислительной системы, посылающий запросы серверу
- Сервер — выделенный или специализированный компьютер для выполнения сервисного программного обеспечения
- База данных — это упорядоченный набор структурированной информации или данных, которые обычно хранятся в электронном виде в компьютерной системе. База данных обычно управляется системой управления базами данных (СУБД)
- HTTP — это протокол, позволяющий получать различные ресурсы, например HTML-документы. Протокол HTTP лежит в основе обмена данными в Интернете
- SQL-запросы — это наборы команд для работы с реляционными базами данных
- Дизайн-макет — это схематичное изображение финальной идеи с указанием всех деталей. В нем указываются концепция, шрифты, тексты, изображения, расположение всех элементов и общая картина продукта
- Аутентификация — процедура проверки подлинности, например, проверка подлинности пользователя путем сравнения введенного им пароля с паролем, сохраненным в базе данных

- Авторизация — предоставление определенному лицу или группе лиц прав на выполнение определенных действий
- Android — это операционная система с открытым исходным кодом, созданная для мобильных устройств на основе модифицированного ядра Linux
- Фреймворк — программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта
- SQL-инъекция — внедрении в запрос произвольного SQL-кода, который может повредить данные, хранящиеся в БД или предоставить доступ к ним
- HTTPS — расширение протокола HTTP для поддержки шифрования в целях повышения безопасности
- Пользователь — человек, который использует приложение
- Аккаунт или учетная запись — это персональная страница пользователя или личный кабинет, который создается после регистрации на сайте
- Frontend — клиентская сторона пользовательского интерфейса к программно-аппаратной части сервиса
- Backend — программно-аппаратная часть сервиса, отвечающая за функционирование его внутренней части
- REST — архитектурный стиль взаимодействия компонентов распределённого приложения в сети
- API — описание взаимодействия одной компьютерной программы с другой
- Трек — это любая звуковая дорожка, электронная музыкальная композиция (мелодия, вокал)
- Песня — это композиция, состоящая из поэтического текста и мелодии. Они выполняют равные смысловые функции

## **2.2 Обзор аналогов**

Разрабатывая приложение, основной задачей которого является хранение, распространение и прослушивание музыки, необходимо рассматривать разработку с точки зрения актуальности и уникальности проекта. Для оценки этих качеств необходимо прибегнуть к рассмотрению аналогов разрабатываемого приложения, адекватно оценивая все положительные и негативные черты того или иного продукта.

Помимо мировой практики разработки продуктов данного направления, необходимо учитывать специфику текущего времени, когда некоторая продукция, сервис и т.д. может быть не доступна на территории страны, которая является целевой аудиторией разрабатываемого продукта.

### **2.2.1 Яндекс.Музыка**

Яндекс.Музыка — это онлайн-сервис, предоставляющий простой и удобный доступ к миллионам аудио-треков, подборкам, плейлистам, радиостанциям и музыкальным подкастам. Яндекс.Музыка рекомендует своим пользователям новинки и композиции, исходя из их личных предпочтений, возраста, настроения и времени суток. Также сервис позволяет создавать персональные плейлисты, синхронизировать музыку между разными устройствами. Яндекс.Музыка доступна в большинстве стран и на большинстве релевантных устройств.

Яндекс.Музыка обладает широким спектром предоставляемых услуг и с точки зрения авторского контента, выплачивая хорошие royalty



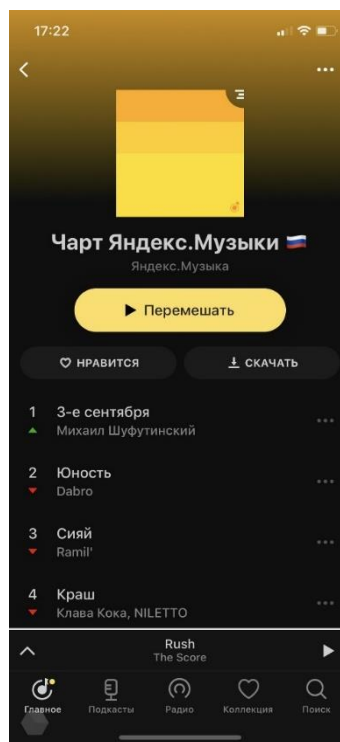


Рисунок 1 - Интерфейс страницы «Яндекс.Музыка»

Недостатки:

- Ограниченный доступ к музыкальным трекам
- Высокая стоимость подписки на сервис
- Рекламные блоки на платформе могут быть раздражающими
- Неудобство в использовании приложения на некоторых устройствах
- Ограниченный доступ к облаку и потеря данных
- Неполное совпадение исполнителей, альбомов и треков
- Не удастся найти все композиции и исполнителей
- Невозможность загрузить личную музыку в плейлисты
- Вирусные подборки, появляющиеся в рекомендациях, не соответствующие запросам пользователя

### 2.2.2 YouTubeMusic

YouTubeMusic — это онлайн-музыкальный сервис, представленный компанией YouTube в 2018 году. Сервис предоставляет своим пользователям доступ к миллионам песен и музыкальным видео, а также плейлистам, радиостанциям и рекомендациям на основе предпочтений пользователя.

Стоит отметить, что на фоне многих известных зарубежных стриминговых сервисов именно продукт Google отличается хорошо работающими рекомендациями, что идеально могут скрасить вечер слушателя. В плане работы с рекомендациями данный продукт конкурирует с такими стриминговыми сервисами как Spotify, Dizzer и т.д.

YouTubeMusic доступен для использования на различных устройствах, таких как компьютеры, смартфоны и планшеты, и поддерживает различные операционные системы, включая Android и iOS. Он также интегрирован с другими сервисами Google, такими как GooglePlay Музыка.

Сервис предлагает возможность прослушивания музыки как в режиме онлайн, так и офлайн, в зависимости от настроек пользователя. Он также включает функции автоматического плейлиста и адаптивного режима, который позволяет пользователям настроить оптимальное качество звука в зависимости от доступной связи с интернетом.

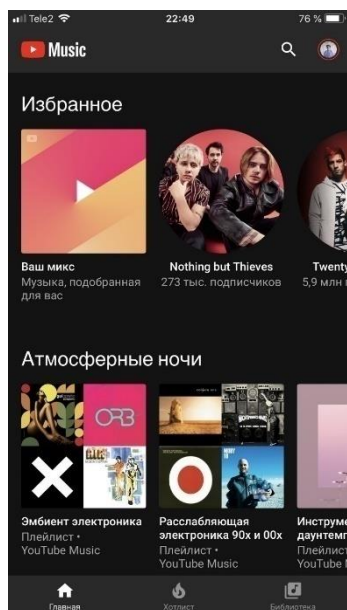


Рисунок 2 - Интерфейс страницы «YouTubeMusic»

Недостатки:

- Громоздкий интерфейс
- Плохая работа с карточками музыкантов
- Создание темы исполнителя без возможности привязать актуальную страницу исполнителя в YouTube

- Различное качество звука от трека к треку
- Отсутствие некоторых функций, таких как создание временного плейлиста

### **2.2.3 Spotify**

Spotify — это стриминговый сервис, предлагающий доступ к миллионам песен, подкастам и аудиокнигам. Spotify позволяет пользователям создавать плейлисты, слушать музыку в режиме онлайн или офлайн (сохранение в кэш), делиться треками с друзьями, искать новых исполнителей и многое другое. Spotify также имеет мобильное приложение и многочисленные интеграции с другими платформами и устройствами.

На данный момент Spotify — это флагман от мира стриминговых сервисов. Его алгоритмы подборки очень точны, а работа с артистами, посредством взаимодействия через цифровой музыкальный дистрибьютор, самая удобная из существующих на данный момент.

Spotify предлагает широкий спектр услуг по распространению музыки определенных исполнителей, включая треки даже малоизвестных исполнителей по жанрам и стилю подходящие под образец в подборки, которые рекомендуются пользователям с соответствующими музыкальными предпочтениями.

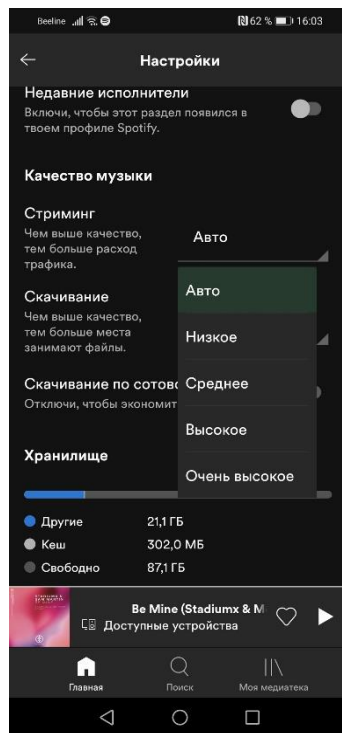


Рисунок 3 - Интерфейс страницы «Spotify»

Недостатки:

- Возможно сохранение не более 10 тысяч треков
- Spotify не доступен в некоторых странах мира, что ограничивает число потенциальных пользователей
- Ограниченный доступ к музыке, в частности, к новым альбомам известных артистов
- Spotify потребляет намного больше интернет-трафика, чем другие сервисы

## 2.3 Моделирование системы

### 2.3.1 Диаграмма в стиле методологии IDEF0

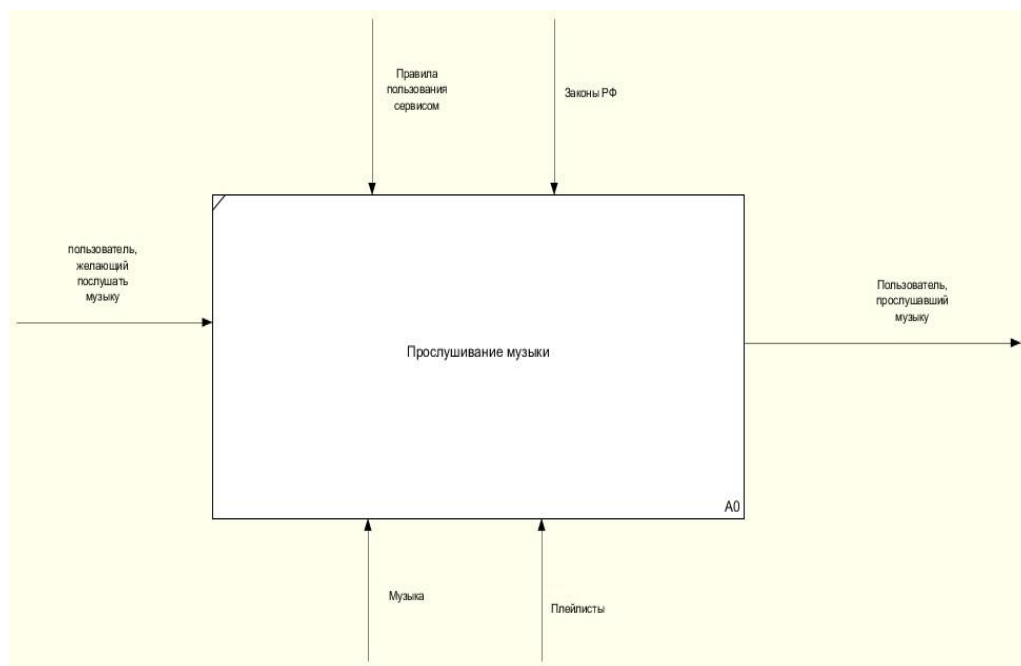


Рисунок 4 - Диаграмма в стиле методологии IDEF0

### 2.3.2 Диаграмма прецедентов

Рассмотрим полную диаграмму для использования приложения разными типами пользователей (неавторизованный, авторизованный, исполнитель). В данном случае необходимость составления диаграммы прецедентов продиктована прежде всего тем, что use-case диаграмма — это инструмент для моделирования системы и понимания ее функциональности и потребностей пользователей. Они помогают в определении основных действий, которые пользователь должен совершить в системе, чтобы достичь определенных целей. Они также позволяют определить возможные риски и проблемы, которые могут возникнуть в ходе использования системы.



Рисунок 5 - Use-Case диаграмма пользования приложением

### 2.3.3 Диаграммы классов

В данном пункте будут рассмотрены диаграммы классов-сущностей серверной части приложения. Диаграмма классов позволяет легко понять структуру проекта и отношения между классами. Это позволяет всем участникам проекта быть на одной волне, что позволяет сократить количество ошибок и повторных разъяснений в процессе разработки, помимо этого классы, которые созданы на диаграмме, обычно хорошо описываются, что позволяет легче отлаживать код и тестировать приложение в целом.

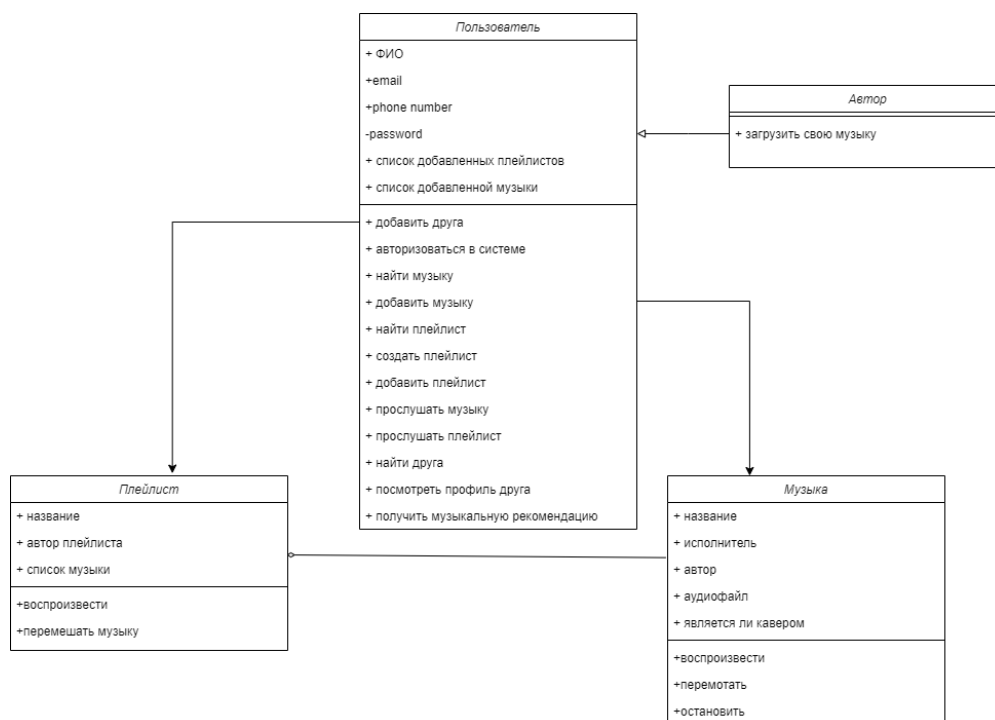


Рисунок 6 - Диаграмма классов-сущностей серверной части приложения.

### 2.3.4 Диаграммы последовательности

Диаграмма последовательности является важным инструментом для проекта, который помогает более глубоко понимать процесс, улучшать его эффективность и упрощать взаимодействие.

Рассмотрим диаграмму последовательности администратора приложения.

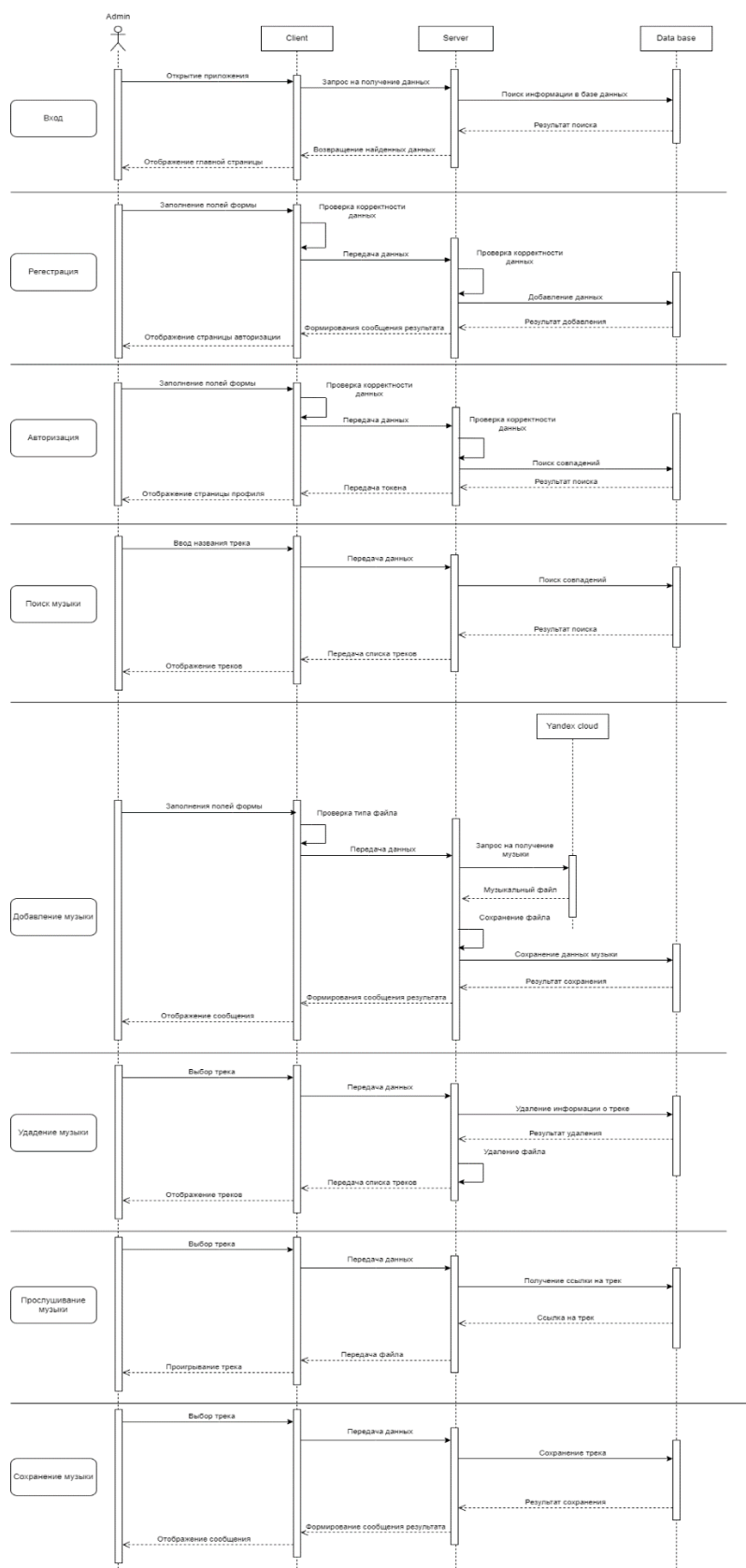


Рисунок 7 - Диаграмма последовательности администратора



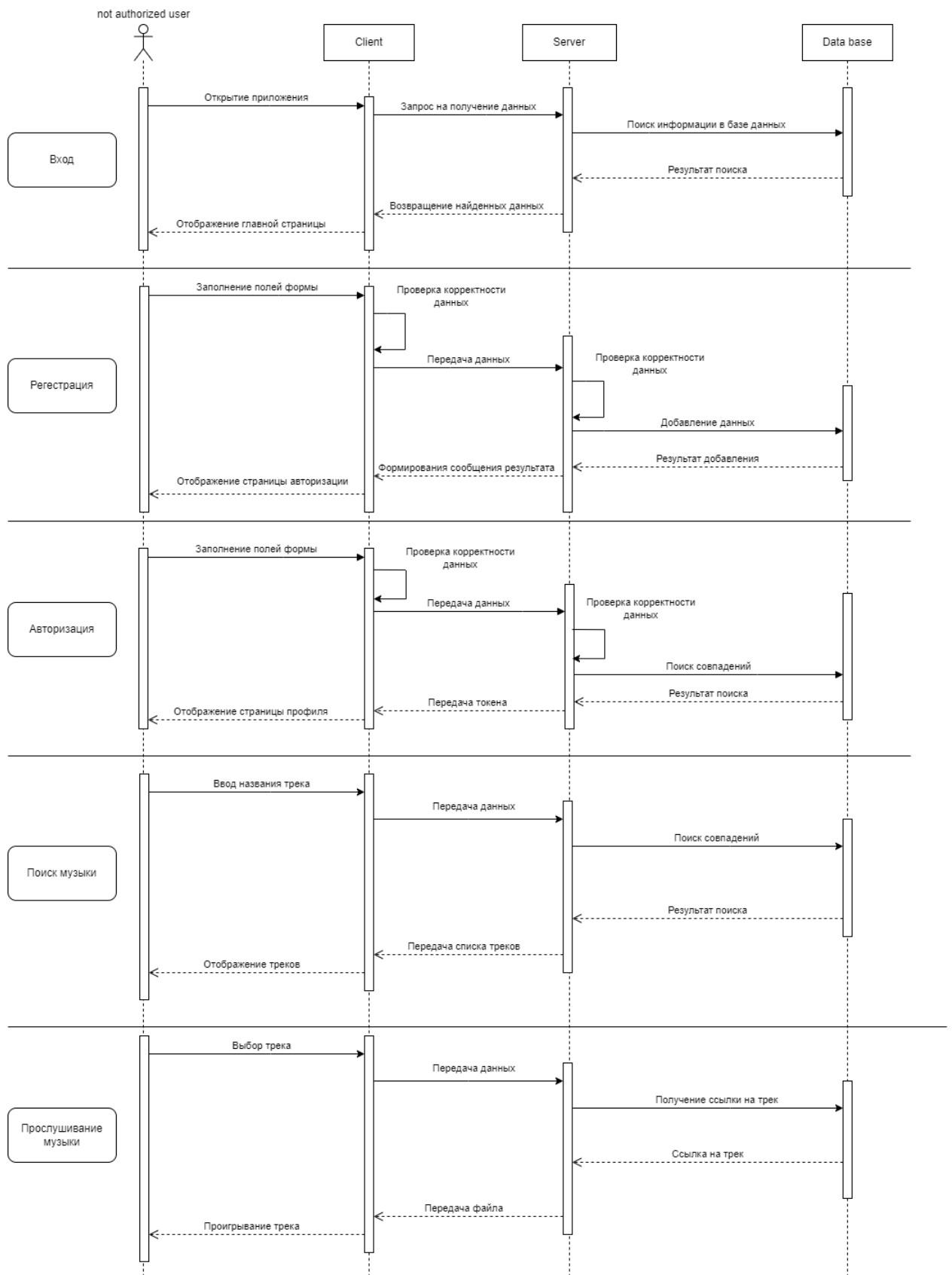


Рисунок 8 - Диаграмма последовательности неавторизованного пользователя

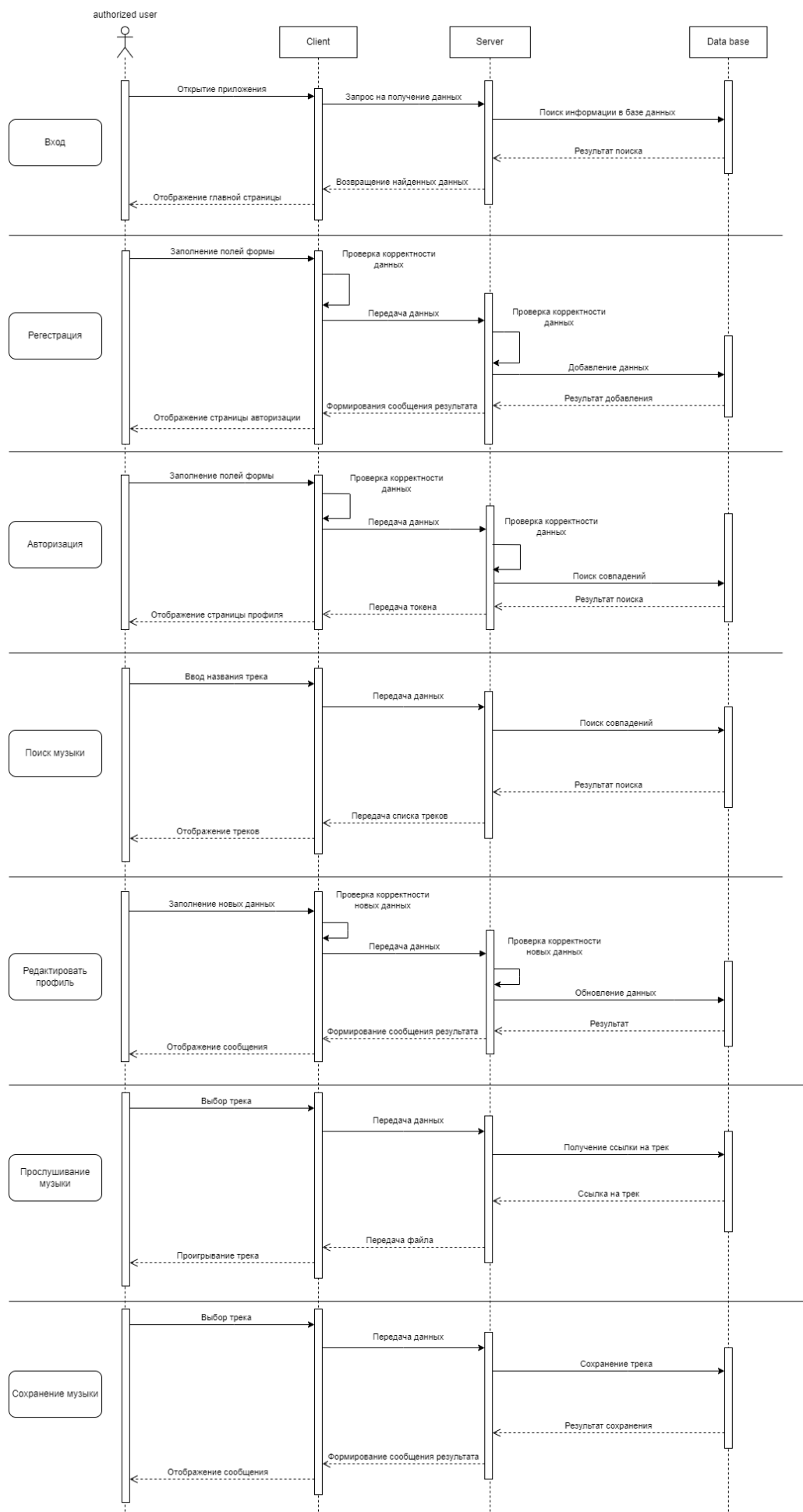


Рисунок 9 - Диаграмма последовательности авторизованного пользователя

### 2.3.5 Диаграмма развертывания

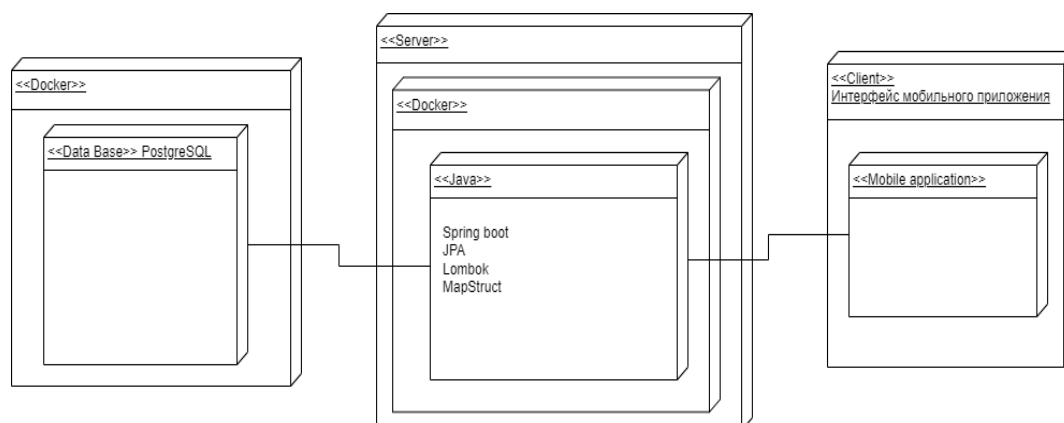


Рисунок 10 - Диаграмма развертывания приложения

### 2.3.6 Диаграмма состояния



Рисунок 11 - Диаграмма состояния

### 2.3.7 Диаграмма объектов

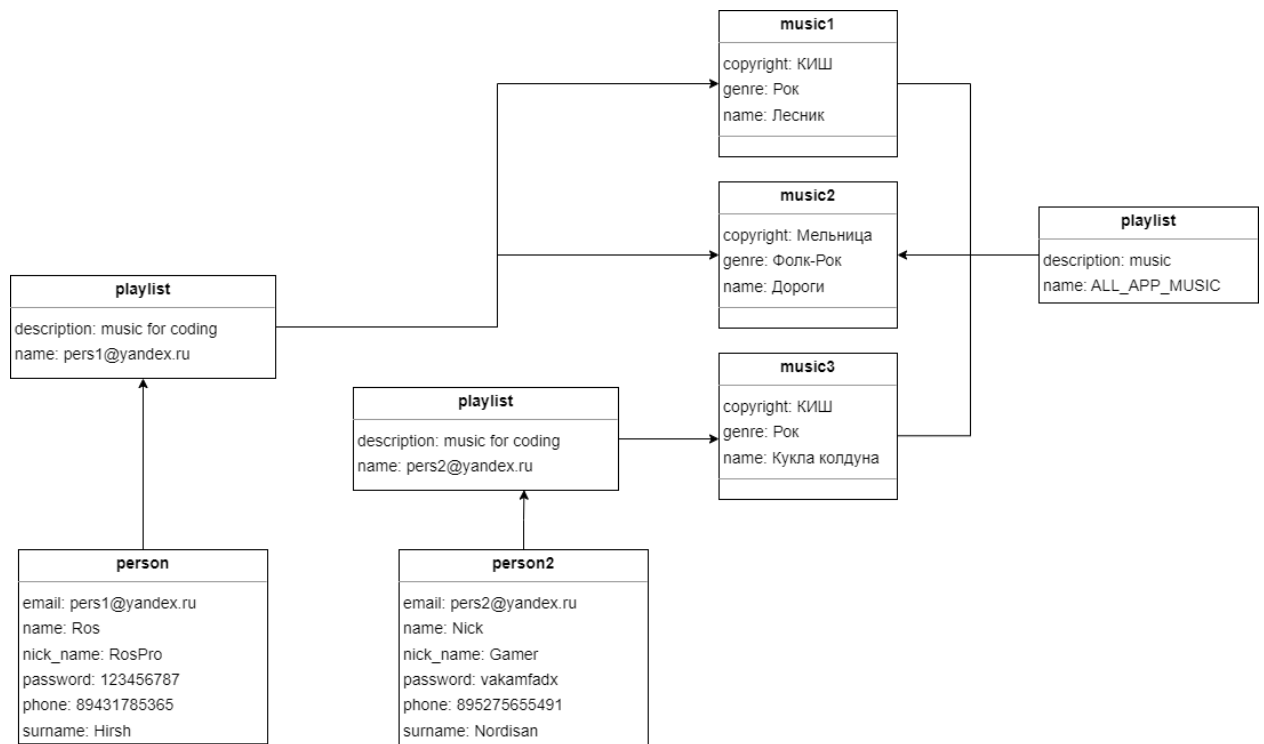


Рисунок 12 - Диаграмма объектов

### 2.3.8 Диаграммы активности

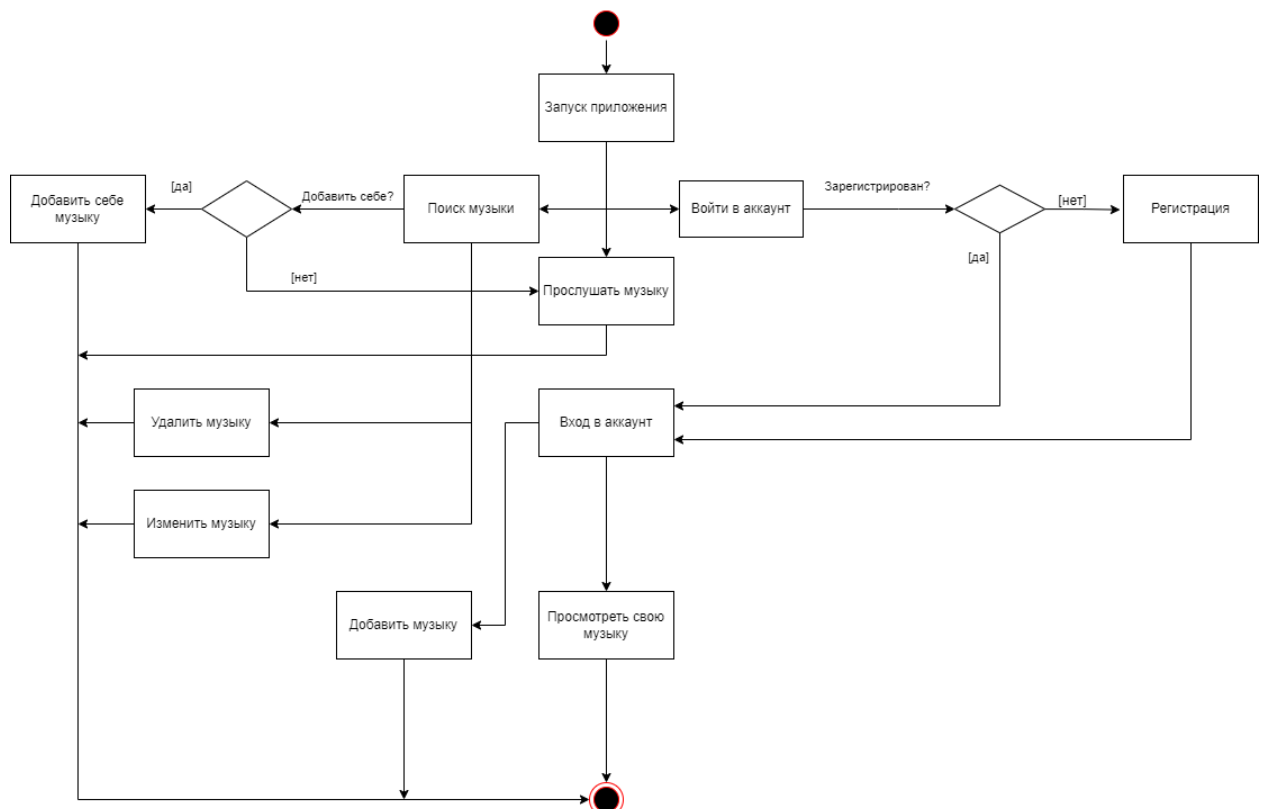


Рисунок 13 - Диаграмма активности администратора

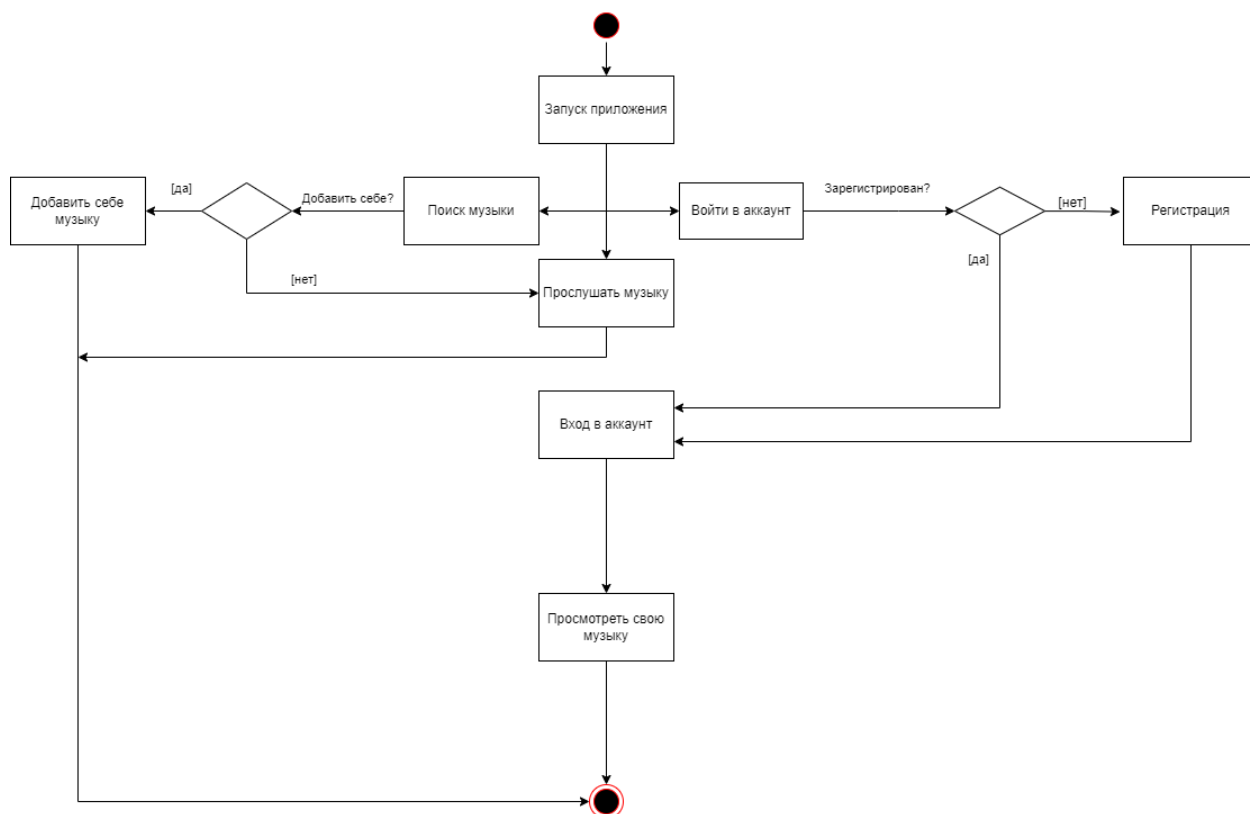


Рисунок 14 - Диаграмма активности пользователя

### 2.3.9 Диаграмма сотрудничества

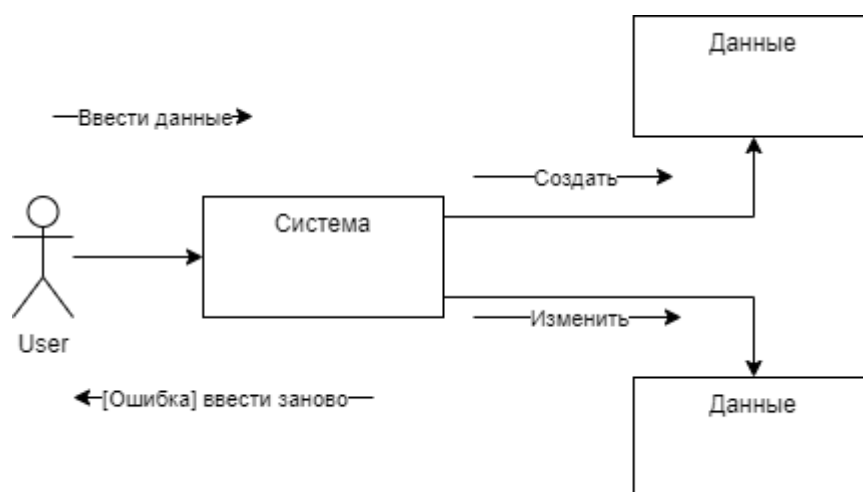


Рисунок 15 - Диаграмма сотрудничества

## 3 Реализация

### 3.1 Средства реализации

Ниже приведен перечень используемых технологий, который в ходе разработки может расширяться.

#### Backend

— Java — строго типизированный объектно-ориентированный язык программирования. Был выбран в качестве основного, т.к. он

- остается очень популярным языком программирования в этой области благодаря своим мощным возможностям и широкому спектру инструментов для разработки. К тому же существует огромное количество фреймворков и библиотек, написанных на Java, которые в перспективе можно легко интегрировать в проект
- `SpringBootFramework` — универсальный фреймворк с открытым исходным кодом для Java-платформы. Был выбран, т.к. он совместим с большим количеством библиотек и фреймворков, что позволяет использовать его в различных проектах и на различных платформах. Так же он позволяет разработчикам быстро создавать приложения без необходимости тратить много времени на конфигурацию
  - `PostgreSQL` — Данное СУБД было выбрано из-за следующего т.к. является продуктом с открытым исходным кодом, который поддерживается многими серверами. Поддержка различных типов данных. `PostgreSQL` поддерживает множественные типы данных, такие как числа разной точности, тексты с различными кодировками, изображения, звуки, видео, XML-документы, JSON-объекты и многие другие.
  - `FlyWay` — продукт с открытым исходным кодом для обеспечения миграций баз данных. Был выбран, т.к. легко интегрируется со `SpringFramework` и поддерживает `PostgreSQL 14`
  - `Docker` — это программная платформа для быстрой разработки, тестирования и развертывания приложений

#### Frontend:

- `Android SDK` — универсальное средство разработки мобильных приложений для операционной системы Android

#### Инструменты для ведения документации:

- `Miro` — платформа для совместной работы распределенных команд
- `Swagger` — это фреймворк для спецификации REST API.

- Draw.io - Бесплатное кроссплатформенное программное обеспечение для рисования графиков с открытым исходным кодом. Его интерфейс можно использовать для создания диаграмм, таких как блок-схемы, каркасы, диаграммы UML
- Ramus – графическая среда для проектирования и моделирования сложных систем широкого назначения, который может быть использован для создания диаграмм в формате IDEF0
- Figma – онлайн-сервис для дизайнеров, веб-разработчиков и маркетологов. Он предназначен для создания прототипов сайтов или приложений, иллюстраций и векторной графики

Дополнительный инструментарий:

- Git — распределённая система управления версиями.
- GitHub — платформа разработки программного обеспечения с открытым исходным кодом, представляющая систему управления репозиториями кода для Git
- Trello — визуальный инструмент, обеспечивающий эффективность командной работы на любом проекте.

## **3.2 Реализация базы данных**

### **3.2.1 ER-диаграмма**

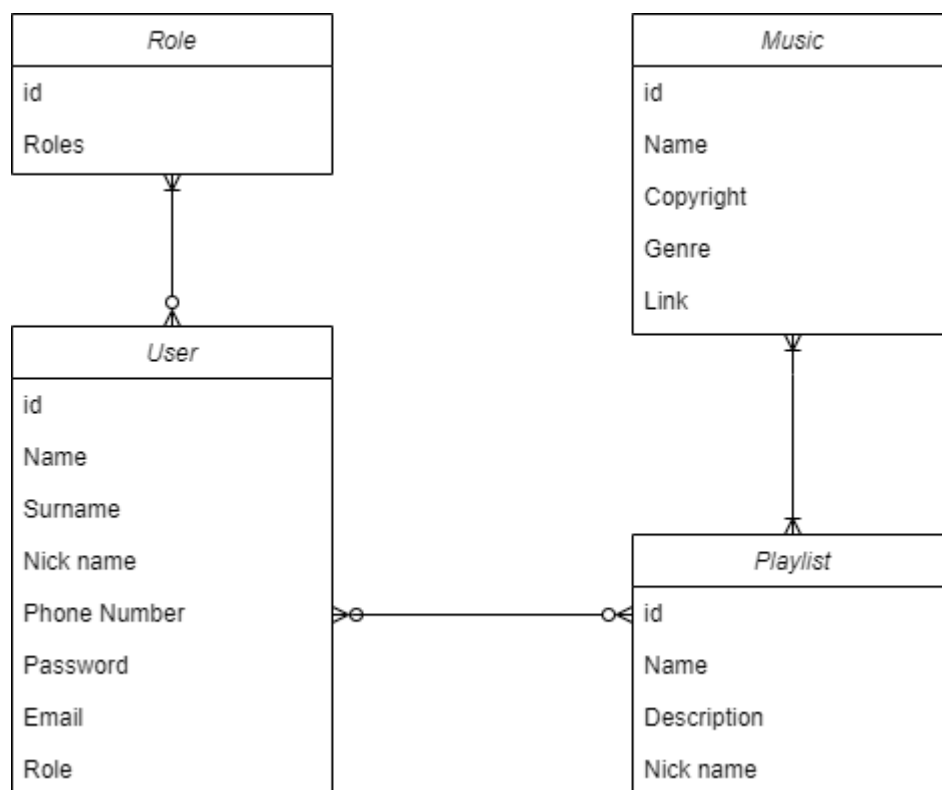


Рисунок 16 - ER-диаграмма базы данных

### 3.2.2 Физическая модель базы данных

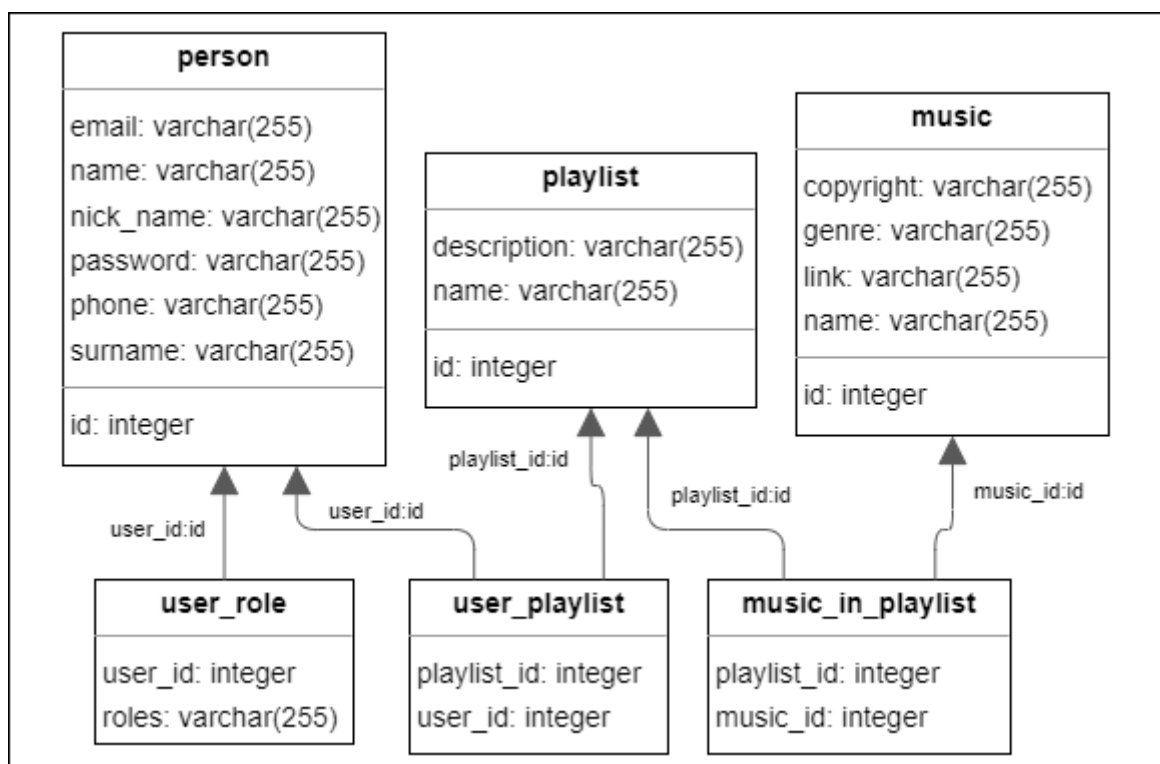


Рисунок 17 - Физическая модель базы данных

## 3.3 Реализация клиентской части

### 3.3.1 Форма для поиска музыки



Пользователь (авторизованный и не авторизованный) имеет возможность найти музыку или для прослушивания по названию или автору. В качестве ответа на его запрос будет выведен список треков. Пользователь может нажать на трек, чтобы прослушать его.

Зарегистрированный пользователь имеет возможность добавить найденную музыку себе в профиль.

Администратор имеет возможность редактировать или удалить музыку.

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: поиск/главная, прослушивание музыки, личная страница.

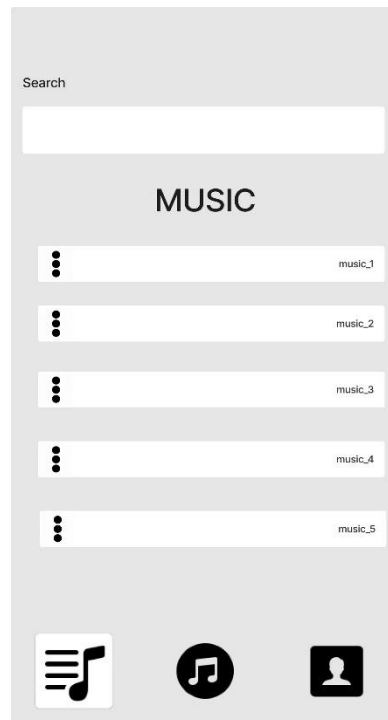


Рисунок 18 - Форма экрана поиска музыки

### 3.3.2 Форма экрана плеера

На данном экране расположена панель с названием трека и его описанием, слайдер, отображающий текущее время трека и кнопки:

- перемотка трека
- стоп/играть
- следующий трек

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: поиск/главная, прослушивание музыки, личная страница.

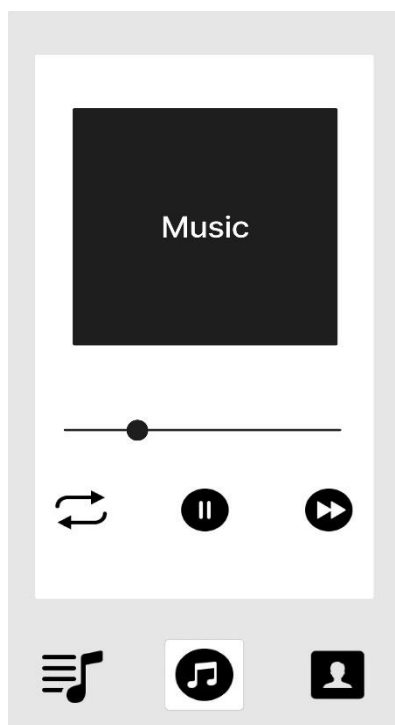


Рисунок 19 - Форма экрана плеера

### 3.3.3 Форма экрана авторизации

На данном экране отображены поля с вводом логина и пароля к аккаунту, кнопка войти в аккаунт, кнопка зарегистрироваться, а также кнопка восстановление пароля.

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: поиск/главная, прослушивание музыки, личная страница.

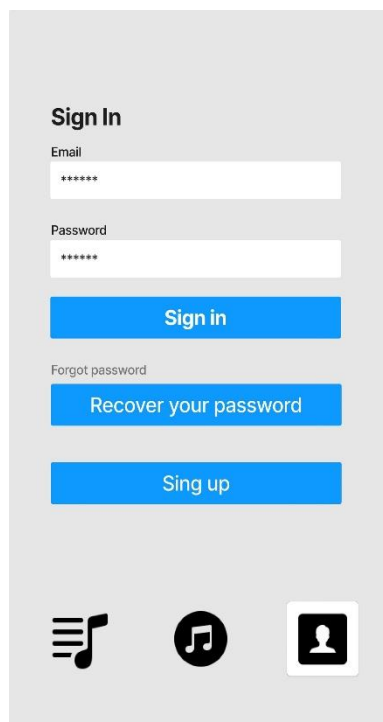
A mobile app screen for signing in. At the top, the title "Sign In" is displayed. Below it are two input fields: "Email" and "Password", both containing six asterisks. A blue button labeled "Sign in" is positioned below the password field. Underneath the button is a link "Forgot password" and a corresponding blue button labeled "Recover your password". At the bottom of the form area is a blue button labeled "Sing up". Below the buttons are three icons: a musical staff, a musical note, and a person silhouette.

Рисунок 20 - Форма экрана авторизации

#### 3.3.4 Форма экрана регистрации

На данном экране отображены поля для ввода имени, фамилии, псевдоним, почты, номера телефона, пароль. Ниже расположена кнопка регистрации.

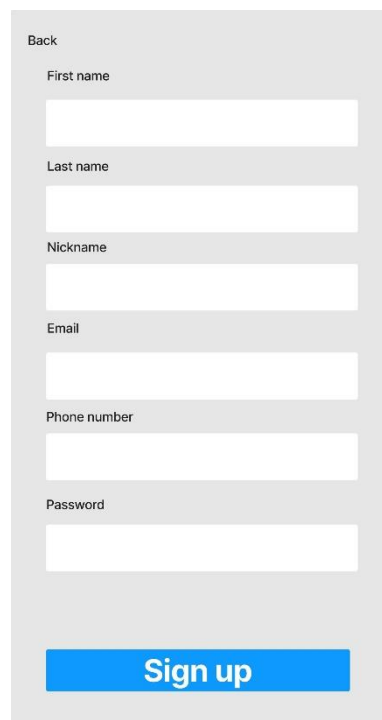
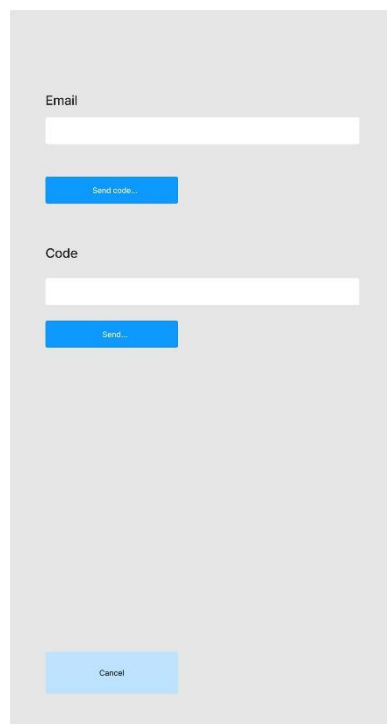
A mobile app screen for registration. At the top left is a "Back" link. Below it are seven input fields: "First name", "Last name", "Nickname", "Email", "Phone number", and "Password". At the bottom is a blue button labeled "Sign up".

Рисунок 21 - Форма экрана регистрации

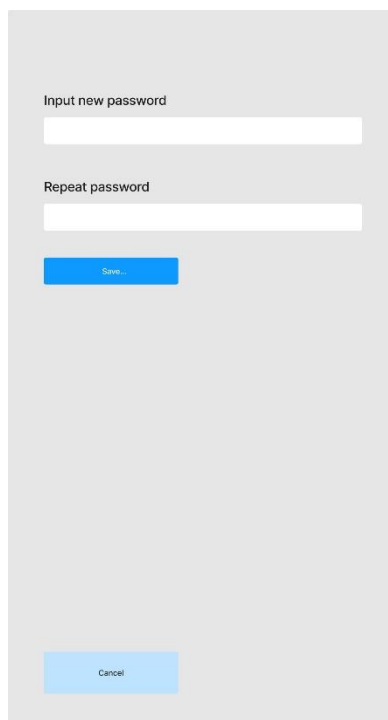
#### 3.3.5 Форма экрана восстановления пароля

На данном экране отображены поля для ввода адреса электронной почты и кнопка отправки кода. После введения данных появляется форма для ввода полученного кода, и кнопка для подтверждения кода. При успешной проверке кода, появляется форма для заполнения нового пароля, и кнопка подтверждения нового пароля. В случае неудачи пользователь будет возвращен на начальную форму восстановления пароля.



The screenshot shows a mobile application interface for password recovery. It features a light gray background. At the top, there is a label "Email" above a white text input field. Below the input field is a blue button with the text "Send code...". Further down, there is a label "Code" above another white text input field. Below this second input field is a blue button with the text "Send...". At the bottom of the screen, there is a light blue button with the text "Cancel".

Рисунок 22 - Форма экрана восстановления пароля



The screenshot shows the next step in the password recovery process. It features a light gray background. At the top, there is a label "Input new password" above a white text input field. Below this input field is a label "Repeat password" above another white text input field. Below the second input field is a blue button with the text "Save...". At the bottom of the screen, there is a light blue button with the text "Cancel".

Рисунок 23 - Форма экрана ввода нового пароля

### 3.3.6 Форма экрана личной страницы

На данном экране указана информация о пользователе (имя, фамилия, псевдоним, почта и номер телефона) рядом расположена кнопка редактирования пользователя и выхода из аккаунта. Ниже находится кнопка для просмотра добавленной музыки.

Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: поиск/главная, прослушивание музыки, личная страница.

Если пользователь является администратором, то на экране появляется кнопка загрузки музыки.

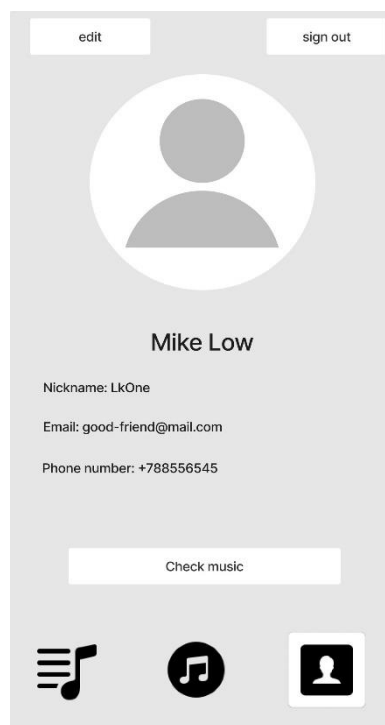


Рисунок 24 - Форма экрана личной страницы

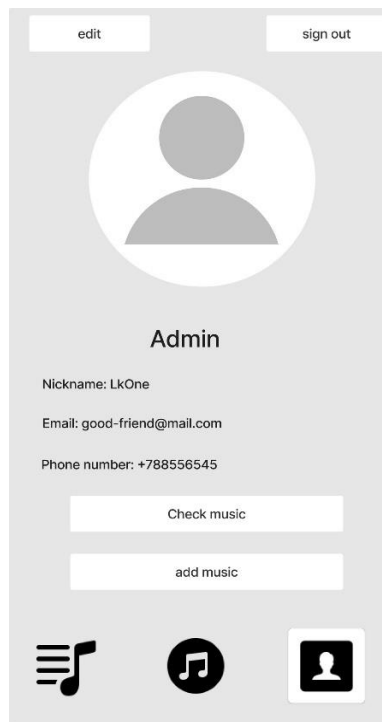


Рисунок 25 - Форма экрана личной страницы администратора

### 3.3.7 Форма экрана редактирования профиля

На данном экране отображены поля для ввода имени, фамилии, псевдоним, почты, номера телефона, пароль с уже введенными соответствующими данными. Ниже расположена кнопка для сохранения данных.

The image shows a mobile application interface for editing a profile. At the top, there is a 'Back' button. Below it are several input fields: 'First name', 'Last name', 'Nickname', 'Email', 'Phone number', and 'Password'. Each field has a corresponding text label and a white input box. At the bottom of the form is a large blue button with the text 'Save'.

Рисунок 26 - Форма экрана редактирования профиля

### 3.3.8 Форма экрана загруженной музыки

На данном экране расположены список загруженной музыки со стороны администратора приложения, и кнопка для добавления музыки.

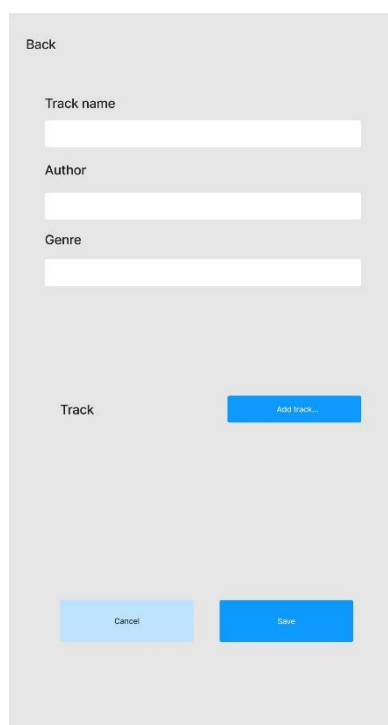
Снизу расположена панель с кнопками, при нажатии на соответствующую кнопку пользователь переходит на страницы: поиск/главная, прослушивание музыки, личная страница.



Рисунок 27 - Форма экрана загрузки музыки

### 3.3.9 Форма экрана добавления музыки

На данном экране расположены поля для заполнения названия трека, автор, жанр музыки и загрузить музыкальный файл, кнопки для сохранения и выхода.



Back

Track name

Author

Genre

Track

Add track...

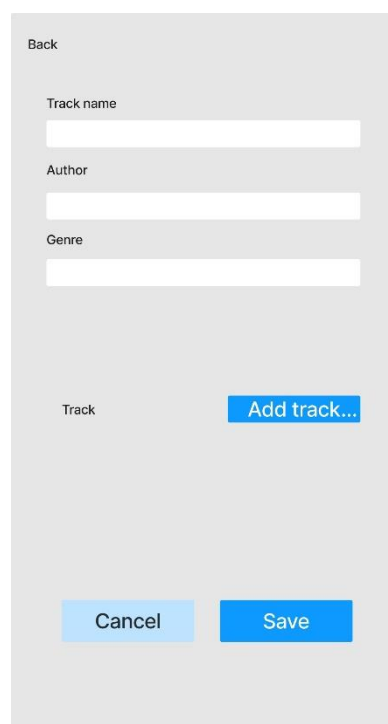
Cancel Save

This is a mobile app form for adding a new track. It features a 'Back' button at the top left. Below it are three input fields for 'Track name', 'Author', and 'Genre'. Further down is a 'Track' label and an 'Add track...' button. At the bottom are 'Cancel' and 'Save' buttons.

Рисунок 28 - Форма экрана добавления музыки

### 3.3.10 Форма экрана редактирования музыки

На экране расположены поля уже заполненными соответствующими данными для редактирования (название музыки, автор, жанр музыки) и загруженный музыкальный файл. В нижней части находятся кнопки для сохранения и выхода.



Back

Track name

Author

Genre

Track

Add track...

Cancel Save

This is a mobile app form for editing an existing track. It features a 'Back' button at the top left. Below it are three input fields for 'Track name', 'Author', and 'Genre'. Further down is a 'Track' label and an 'Add track...' button. At the bottom are 'Cancel' and 'Save' buttons.

Рисунок 29 - Макет экрана редактирования музыки



## **3.4 Серверная часть**

### **3.4.1 Основные положения**

Java Spring Framework является основной библиотекой сервера. Запросы различных типов обрабатываются контроллерами, которые взаимодействуют с базой данных и передают обработку бизнес логики сервисам. Сервисы, в свою очередь, взаимодействуют с репозиториями и сущностями. При завершении работы метода сервиса, результат возвращается в контроллер и далее – в тело-ответ сервера.

## **4 Тестирование**

## **Заключение**

При разработке мобильного приложения в рамках курсового проекта и анализа приложений для рекомендации музыки, было замечено, что необходимо создать новое приложение, лишенное проблем предыдущих аналогов. С целью решения этой проблемы были определены задачи, которые помогли создать клиент-серверное приложение "Muzic Zone". Оно предназначено для прослушивания музыкальных произведений пользователям.