**ТЗ**

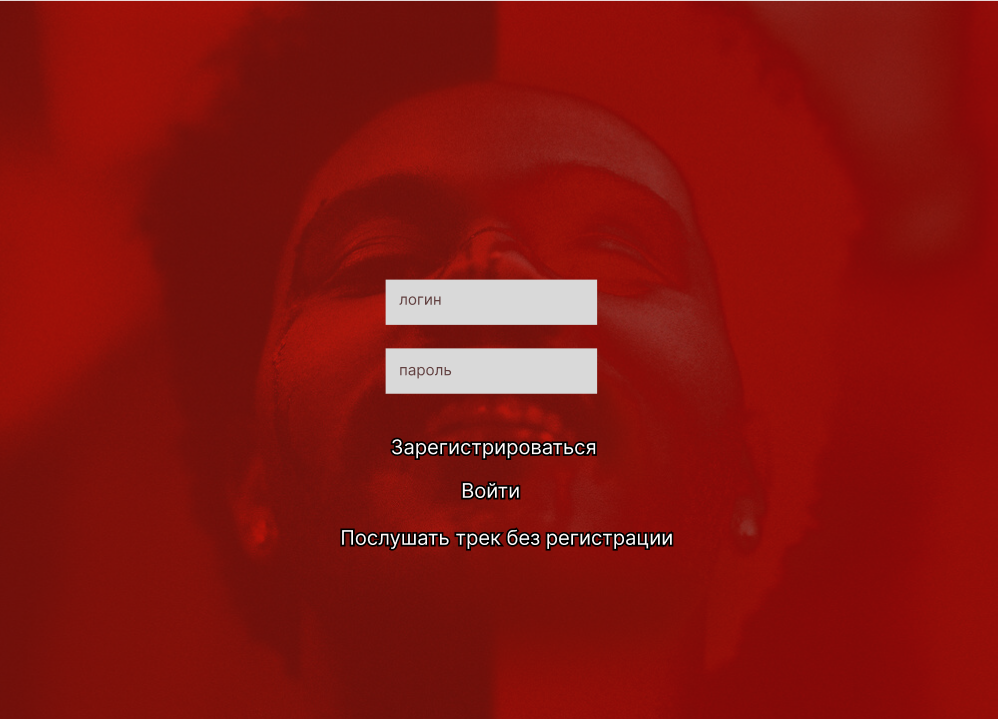
**Music web-site: Weesick**

1. **Область применения**

Сайт подойдет всем любителям музыки, а также тем, кто желает поделиться впечатлениями о прослушанном треке или же просто скачать себе любимую песню.

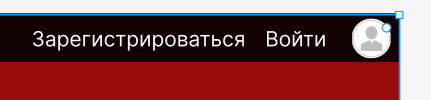
1. **Требования к функциональности**

После перехода на сайт клиент попадает на окно, где регистрируется, либо входит в уже существующий аккаунт, либо просто переходит на сайт. Но в этом случае он сможет лишь слушать треки. Ему будут не доступны любые другие функции, такие как: написание комментариев, создание плейлистов и тд.

****

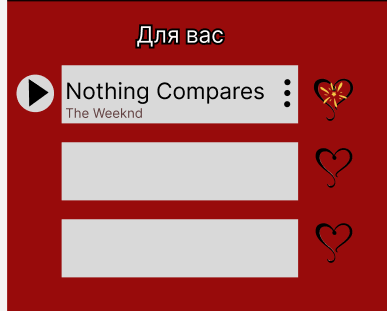
Страница входа/регистрации пользователя

Далее на главной странице в правом верхнем углу клиент сможет также войти/зарегистрироваться или перейти на свой аккаунт.



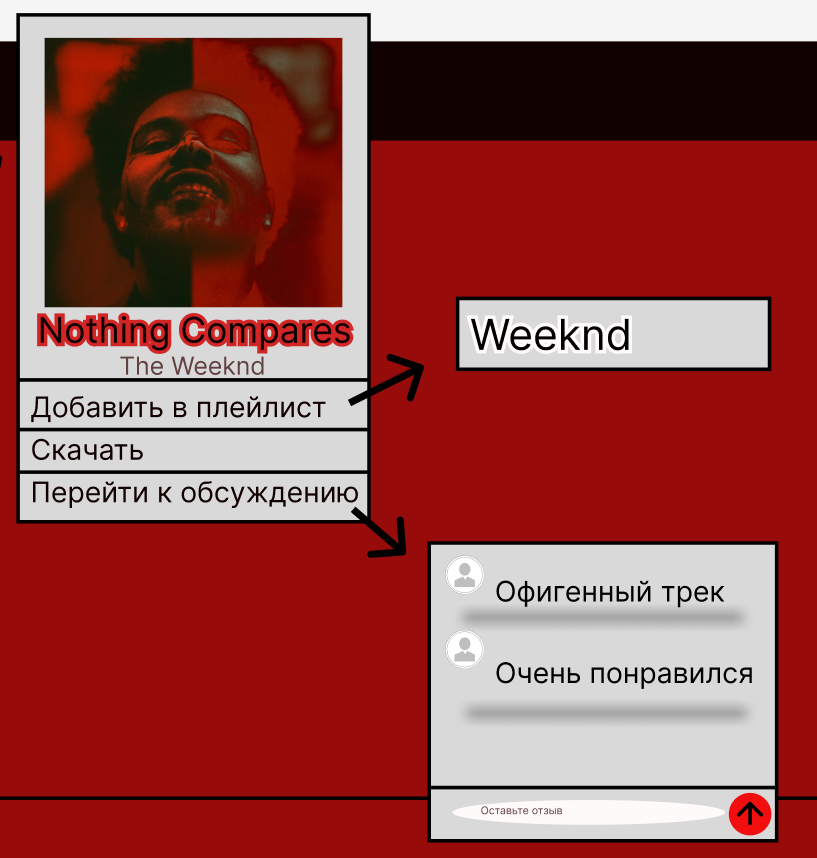
Вход/регистрация/аккаунт

В верхней части главного экрана пользователю будут предложены треки, которые он может послушать или лайкнуть.



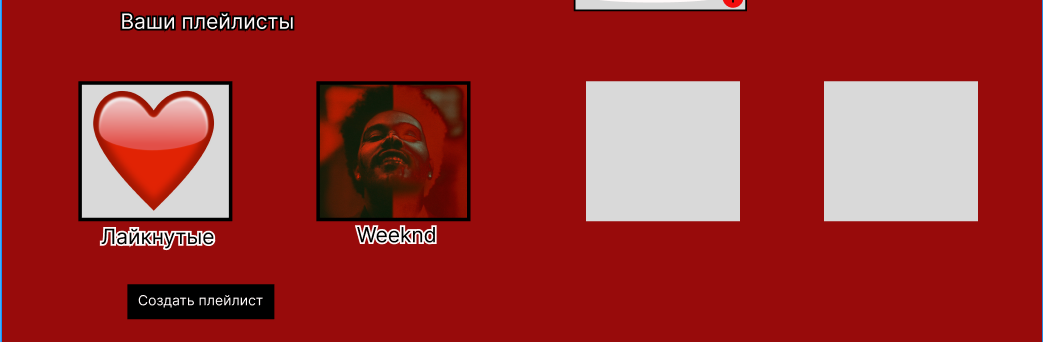
Рекомендации треков

Если клиент нажмет на три точки справа от названия песни, то сможет скачать трек себе, добавить в плейлист или перейти к обсуждению трека.



Верхняя часть главного экрана

Также плейлист для начала нужно создать, чтобы добавить туда трек. Это делается с помощью кнопки *«Создать плейлист»* снизу экрана. После вводится его название и готово. Плейлист *«Лайкнутые»* заполняется автоматически, когда пользователь лайкает треки.



Плейлисты

Также, если клиент нажмет на плейлист, то перейдет на новую страницу сайта и там уже сможет увидеть список треков в плейлисте, послушать их, или удалить трек из плейлиста по желанию.

1. **Требования к эксплуатации и надежности**

Пользователю необходимо иметь базовые навыки в пользовании веб-браузером. Для того чтобы обеспечить защиту аккаунта клиента, пароль от аккаунта пользователя при регистрации сохраняется в базу данных в хешированном состоянии.

1. **Требования к исходным кодам и языку программирования**

Приложение написано на языке программирования Python с использованием фреймворка Flask, библиотеки sqlite3, json и requests.

1. **Требования к структуре базы данных**

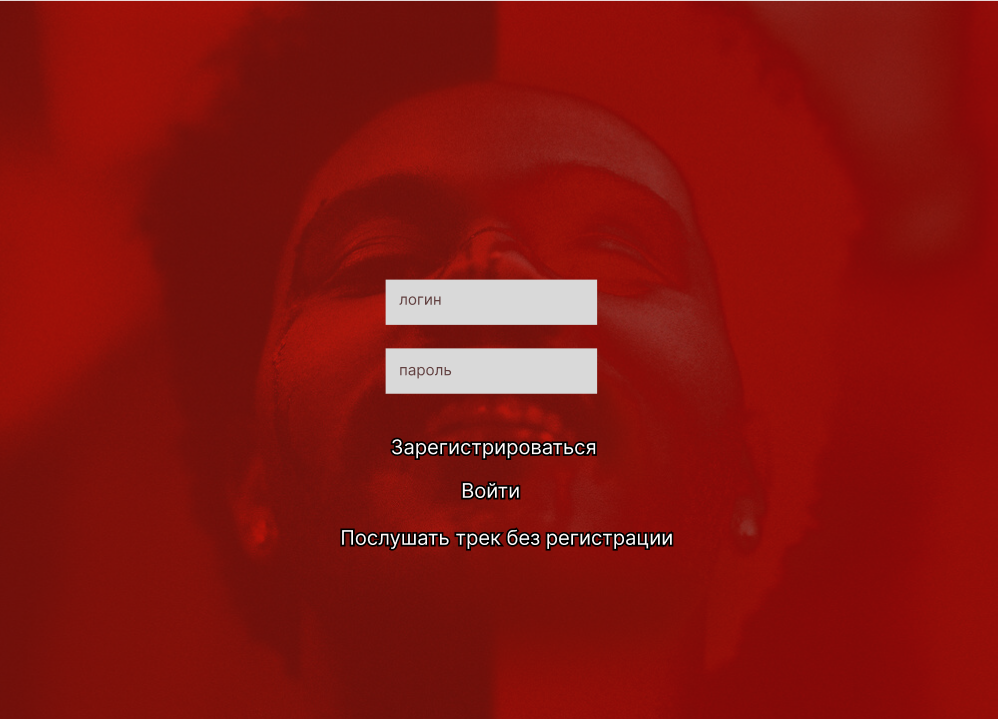
База данных, созданная с помощью библиотеки sqlite3, описана в несколько таблиц:

1. Таблица пользователей (id, логин, пароль)

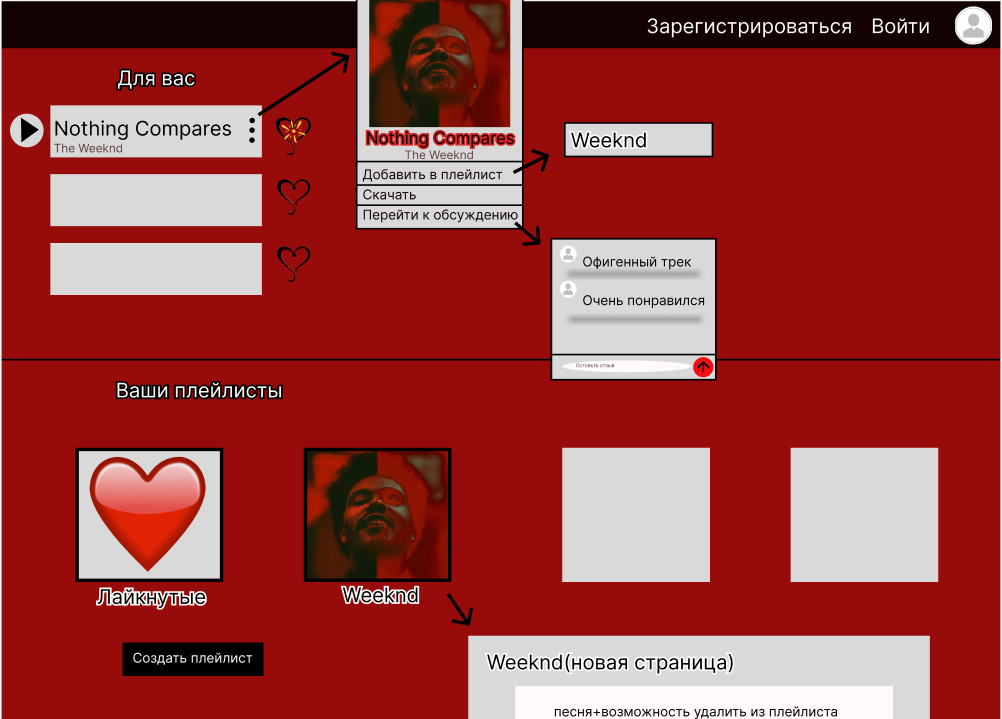
2. Таблица хранения треков (id, автор, название, картинка, комментарии)

3. Таблица сохраненных плейлистов и треков у пользователей (id пользователя, id плейлиста, id песен)

1. **Внешний вид приложения**

****

Вход/регистрация



Главная страница