Пояснительная записка

к проекту

«Blackland Radio»

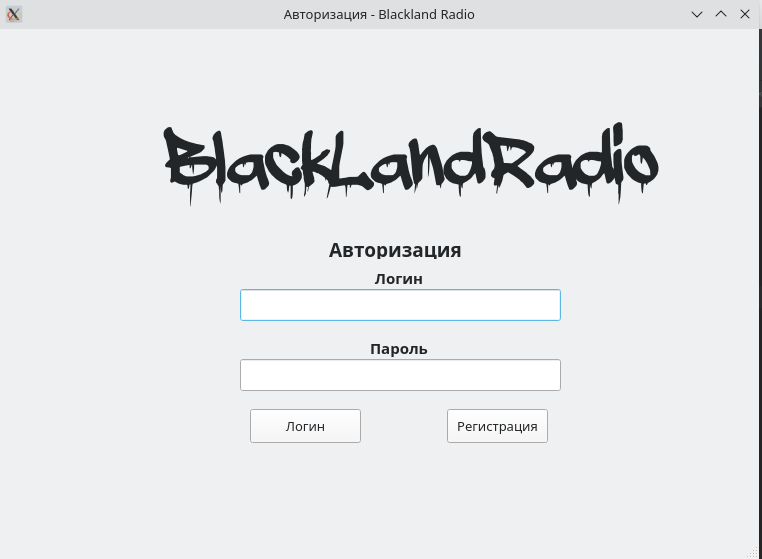
Автор: Пронюк Георгий

**Идея проекта**

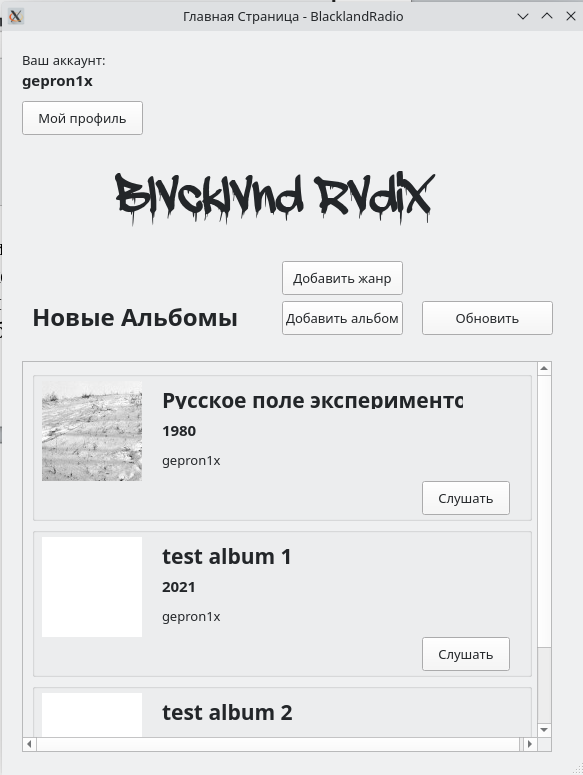
Blackland Radio — музыкальный сервис, где пользователи свободно могут делиться некоммерческой музыкой.

**Функционал**

По запуску программы нас встречает окно авторизации:

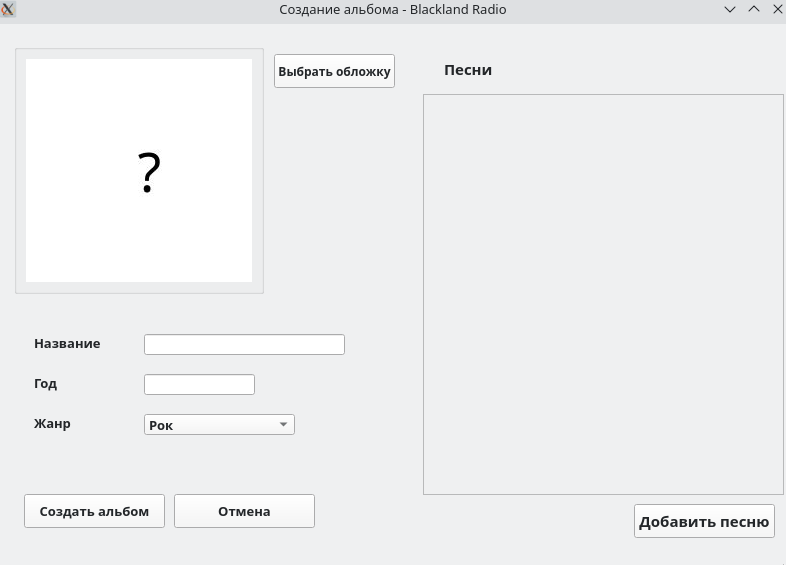
  
Здесь можно зарегистрировать или войти в уже существующий аккаунт. Если аккаунт с таким же именем существует или пароль/имя пользователя введено неправильно, в statusBar будет отправлено соответсвенное сообщение об ошибке.

После успешной авторизации нас встречает главная страница:

  
Тут и сконцентрирован главный функционал программы: Прослушивание чужих и добавление своих альбомов, создание жанров, редактирование своего профиля.

**Добавление альбома**

Открывается окно где пользователь указывает название, год, жанр, обложку и песни альбома:

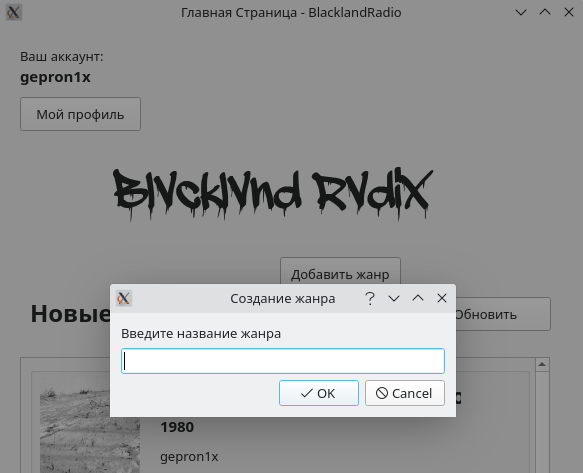


В statusBar’е будет сообщение об ошибке, если:

* Название пустое.
* Год не является числом.
* У песни нету названия или файл не существует.

**Добавление жанра**

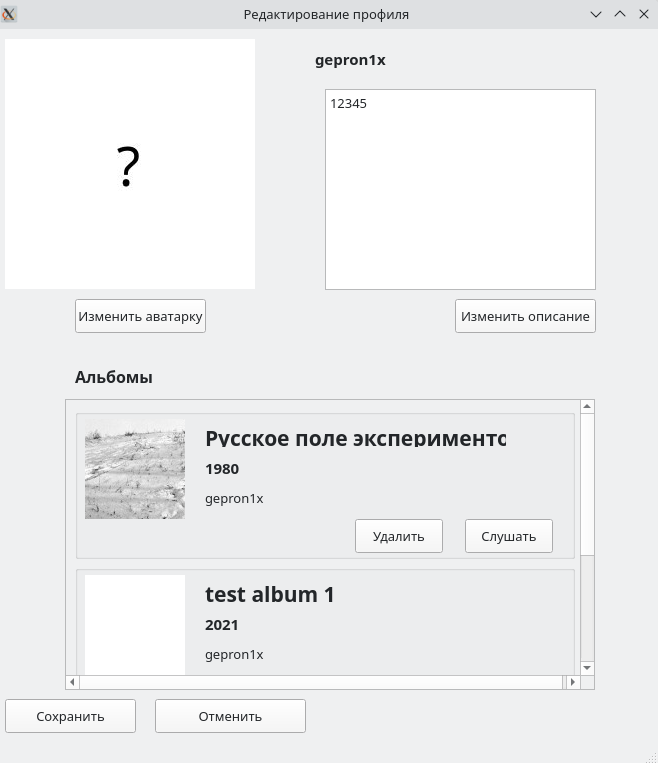
Открывается диалог, в котором пользователя просят ввести название нового жанра:



Если жанр с таким же названием уже существует или юзер попросту не ввёл название, внизу отобразится сообщение об ошибке. В противном случае, жанр сохранится в базу и его можно будет использовать при создании альбома.

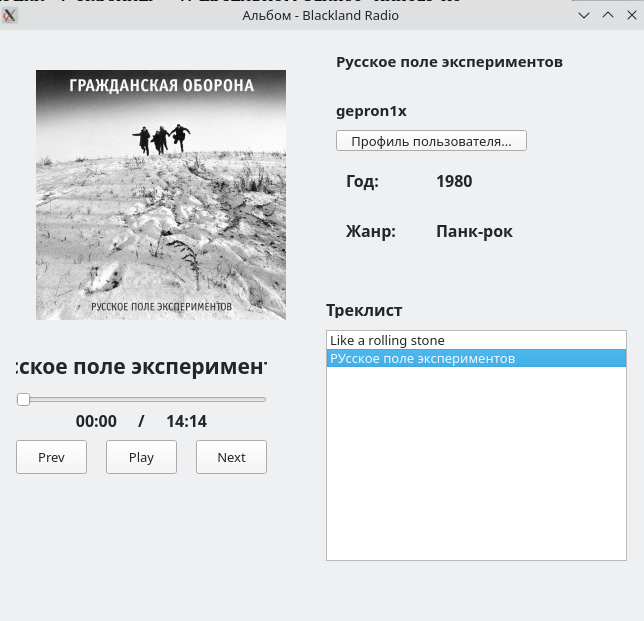
**Редактирование профиля**

Открывается окно в котором можно изменить аватар и описание профиля, а также удалить добавленные на площадку альбомы.

  
Для того, чтобы подтвердить изменения, следует нажать кнопку «Сохранить». В противном случае, ничего не изменится.

**Прослушивание альбомов**

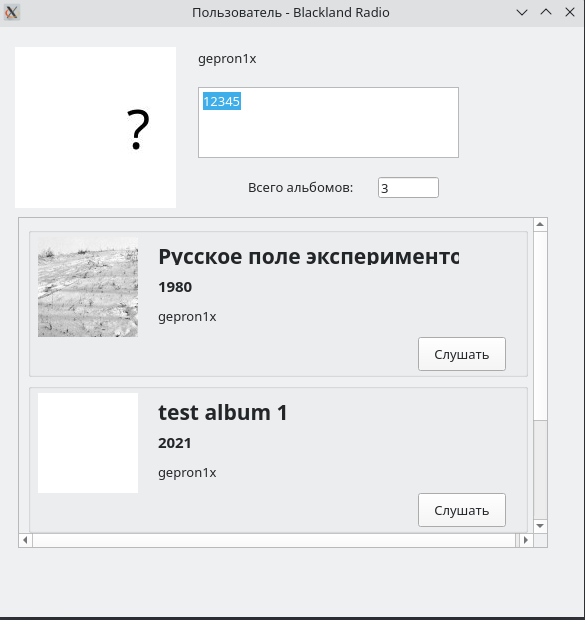
Нас встречает окно плеера. С помощью кнопок Prev/Play/Next можно ставить проигрывание на паузу и переключаться на следующую или предыдущую композицию. Трек так же можно выбрать в треклисте, нажав по нему. По нажатию кнопки «Профиль пользователя» откроется страница пользователя, загрузившего альбом.



**Просмотр профиля**

В окне можно узнать информацию о пользователе:

аватарку, описание и послушать загруженные альбомы:



**Использованные технологии**

Для реализации проекта использовались данные технологии:

* PyQT5 для графического интерфейса программы.
* СУБД SQLite для сохранения, записи и чтения данных.
* QtMultimedia для проигрывания аудио.
* Qt Designer для верстки окон.

**Структура кода**

Код разбит на несколько модулей:

* main.py
* auth.py
* util.py
* api.py
* database.py
* user\_window.py
* album\_window.py
* album\_widget.py
* album\_creation.py

Main

Точка входа. Содержит в себе два класса: BlacklandRadio и MainPage. BlacklandRadio — главный класс программы.

Управляет авторизацией, главной страницей и базой данных. MainPage — главная страница, уже описанная в функционале.

Auth

Авторизация. Содержит в себе класс AuthWindow, управляющий окном входа и регитсрации.

Util

Утилиты, использованные в коде. Содежрит 2 класса:

* WindowHolder — вспомогательный класс-родитель для окон, в которых необходимо открывать другие формы.
* Closable — вспомогательный класс-родитель для тех окон, которые при закрытии вызывают сигнал closed

Api

Содержит в себе классы для удобной работы с результатами запросов из базы данных.

* User — Пользователь
* Album — Альбом
* Song — Песня
* Genre — Жанр

Database

Классы для работы с базой данных.

* BlacklandDatabase — содержит код для взаимодействие с базой: загрузка, сохранение и изменение данных в удобной обертке.
* Editor — предназачен для изменения данных в бд с возможностью отмены. Изменения происходит не сразу,а после вызова метода finish.
* UserEditor — редактор пользователей.

User Window

Окна, связанные с профилями пользователя.

* AbstractUserWindow — родительский класс, содержащий в себе повторяющийся код UserWindow и UserEditorWindow
* UserWindow — окно страницы пользователя.
* UserEditorWindow — окно редактирования профиля.

Album Window

Содержит в себе класс AlbumWindow — окно проигрывателя альбома.

Album Widget

Содержит в себе класс MiniAlbumWidget - мини-виджет альбома, используемый в списках.

Album Creation

Содержит в себе классы, связанные с созданием альбома.

* AlbumCreationWindow — окно создания альбома.
* SongCreationWidget — виджет добавленной песни.

**Структура БД**

Всего в базе 4 таблицы: users, albums, songs, genres

Users

Информация о пользователях. Содержит в себе поля:

* id — id пользователя
* username — имя
* password — пароль
* desciription — описание профиля
* avatar — путь к файлу аватарки пользователя.

Albums

Информация о альбомах. Содержит в себе поля:

* id — id альбома
* user\_id — id пользователя, загрузившего альбом
* name — название альбома
* genre — id жанра
* year — год выпуска альбома
* cover — путь к файлу обложки

Songs

Информация о песнях. Содержит в себе поля:

* id — id песни
* album\_id — id альбома песни
* name — название
* file — путь к mp3 файлу

Genres

* id — id жанра
* name — название

Все таблицы связаны между собой: user\_id в таблице albums указывает на пользователя, загрузишего альбом, album\_id в таблице songs указывает на альбом песни, а genre в albums — на id жанра альбома.

**Планы на будущее**

* Возможность использовать программу на сервере. Разделить серверную и клиентскую часть кода.
* Более функциональная и безопасная авторизация: восстановление доступа и хэширование паролей.
* Больше способов взаимодействия для пользователей: лайки, комментарии, блокировки.
* Глобальный поиск альбомов, пользователей и отдельных песен.