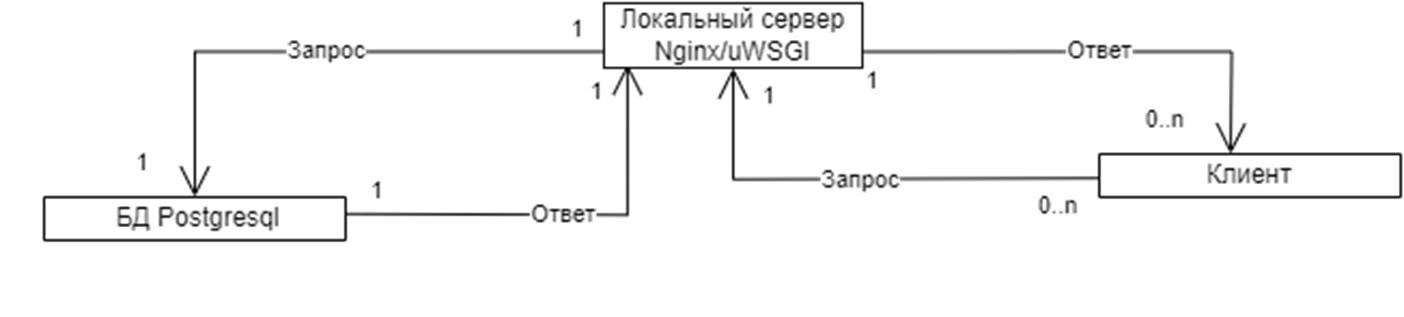
**Контрольная точка 0**

Группа 6303-010302D

Щетинин Михаил и Маргарян Гор

В качестве web-сайта было решено реализовать сайт с минииграми с возможностью регистрации и фиксирования личных рекордов игрока. Так же будет реализована система глобальной таблицы рекордов игроков, каждый пользователь сможет увидеть лучшие результаты других игроков в каждой из игр.

**Модель клиент – сервер**



Для хранения информации будем использовать СУБД PostgreSQL. Сервер будет локальным на нашем оборудовании, с использованием ПО Nginx. Для запуска Django приложения будет использоваться ПО uWSGI.

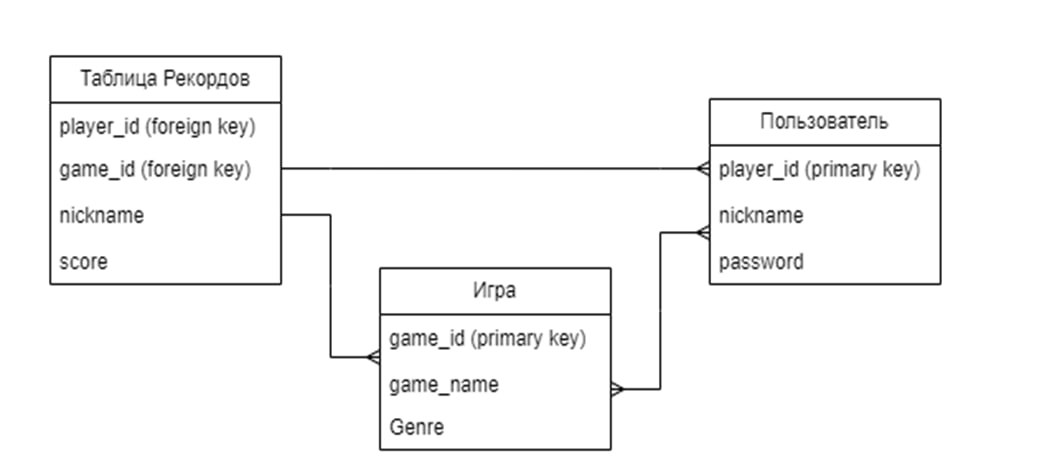
В качестве языка программирования backend будем использовать Python в связке с фреймворком Django. Игры будут написаны с использованием библиотеки Pygame. Frontend разработка будет осуществлена при помощи CSS и фреймворка bootstrap.

Файлы с кодом игр, иконки и картинки хранятся на яндекс диске.

В качестве системы контроля версий будем использовать сервис GitHub.

Ссылка на репозиторий: <https://github.com/Amin71518/ArcadeZone.git>

**База данных**



База данных будет содержать 3 сущности: пользователь, таблица рекордов, игра.

В таблице пользователей будет располагаться информация о всех юзерах, зарегистрированных на сайте.

В таблице игра будет располагаться список всех игр, доступных на сайте.

В таблице рекордов будет вся информация о рекордах пользователей в представленных играх.

Для работы нашего web-сайта будем использовать **REST**.

REST – архитектурный стиль взаимодействия компонентов веб-приложений. Основой является соблюдение 6 принципов:

Клиент – Серверная архитектура.

Stateless.

Кэширование.

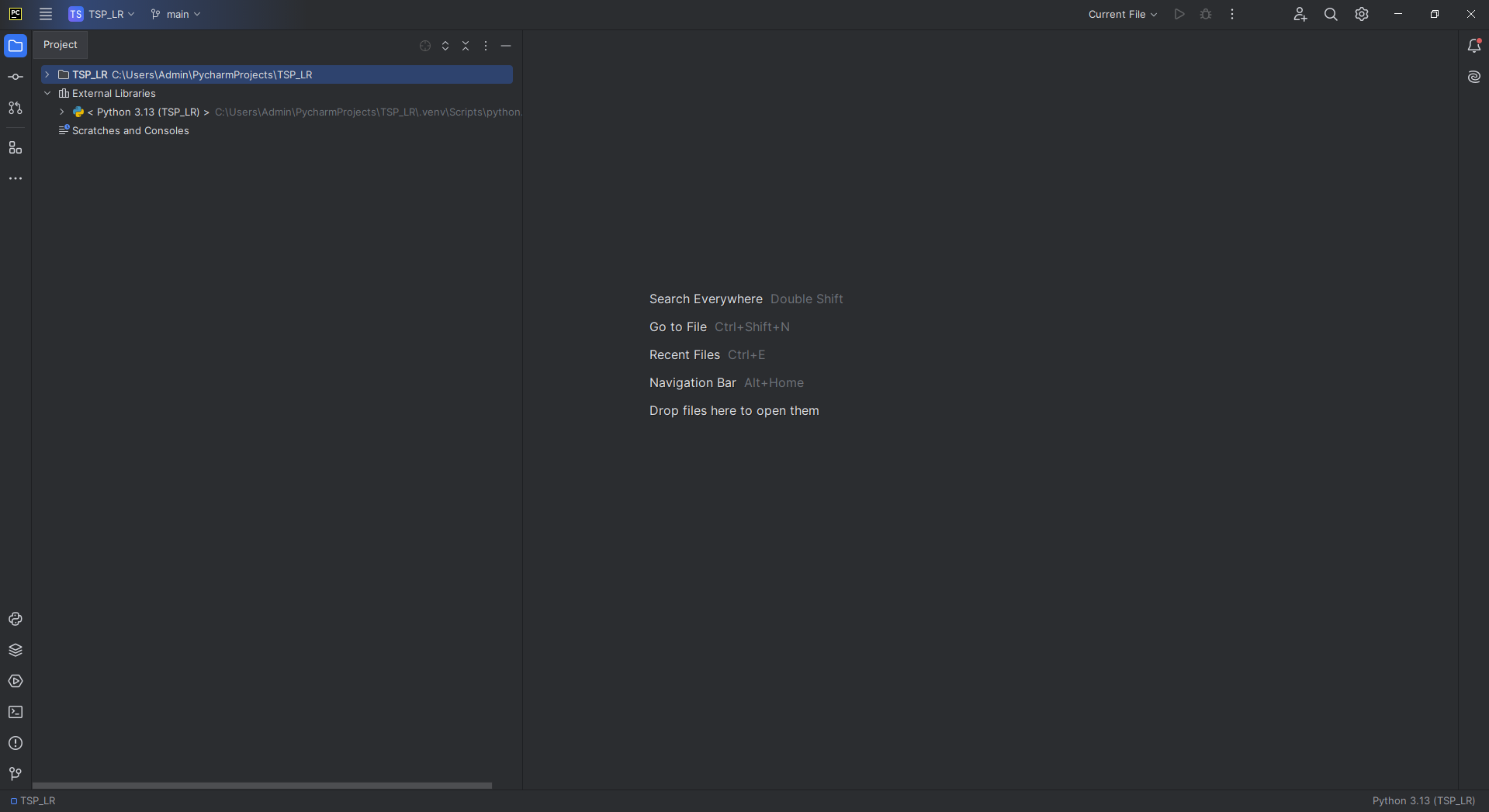
Единообразие интерфейса.

Layered system.

Код по запросу.

**Подготовка окружения**

В качестве среды разработки будет использоваться PyCharm Community Edition, версия Python 13.2.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Метод | url | описание | Тело запроса (json) | Ответ (json) |
| POST | /register/ | Регистрация пользователя | {  "nickname": "player1",  "password": "pass"  } | 201 Created  {  "message": "User registered successfully",  "player\_id": 1  } |
| POST | /login/ | аутентификация | {  "nickname": "player1",  "password": "securepassword"  } | 201 Created  {  "message": "Login successful",  "token": "JWT\_TOKEN"  } |
| GET | /games/ | Получение списка игр | - | 200 OK  {  "game\_id": 1,  "game\_name": "Space Shooter",  "genre": "Action",  "picture\_url": "https://yandex.disk/path/to/picture1.png",  "code\_url": "https://yandex.disk/path/to/code1.zip"  } |
| GET | /games/{game\_id}/start/ | Запуск игры. Сервер создает новую запись в таблице session и возвращает ссылку на код игры | - | 200 OK  {  "message": "Game started",  "session\_id": 123,  "code\_url": "https://yandex.disk/path/to/code1.zip"  } |
| POST | /sessions/{session\_id}/end/ | Сохранение результатов сессии | {  "score": 5000  } | 201 Created  {  "message": "Session ended successfully",  "session\_id": 123,  "score": 5000  } |
| GET | /users/{player\_id}/results/ | Получение результатов игрока |  | 200 OK  {  "session\_id": 123,  "game\_name": "Space Shooter",  "start\_time": "2023-10-01T12:00:00Z",  "end\_time": "2023-10-01T12:10:00Z",  "score": 5000  } |