**Пояснительная записка WebServer+API проекта “Лабиринт”**

*Проект сделан*

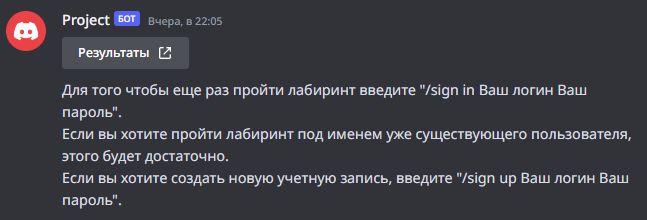
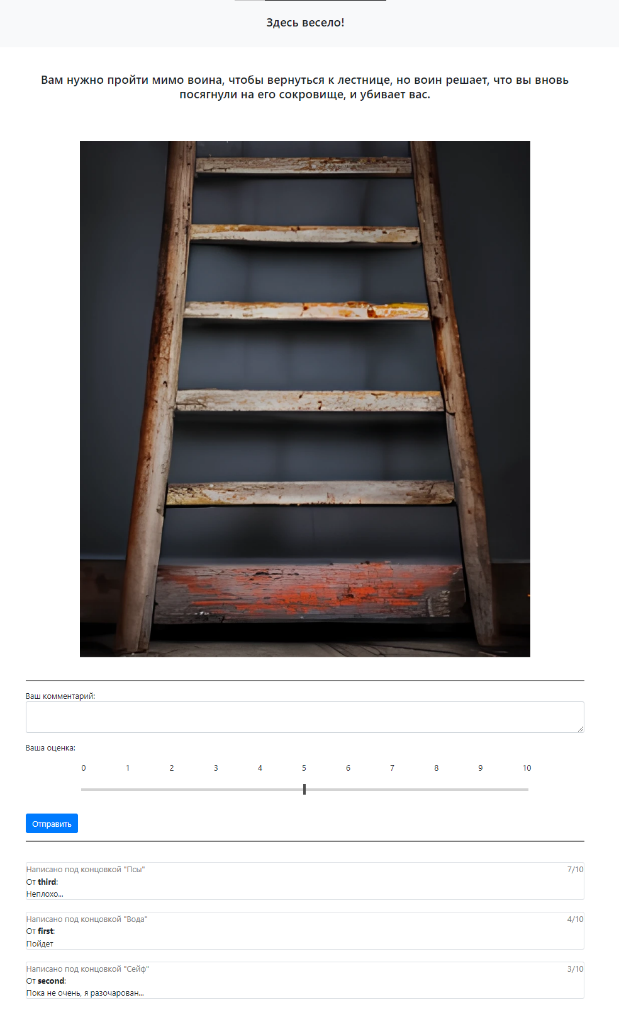
*Шангиной Елизаветой*

*и*

*Хорошавиным Арсением*

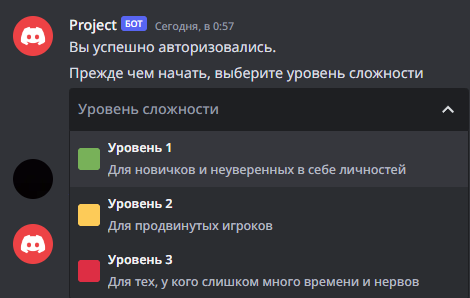
Идеей проекта, которая связывает бот и сайт, является мини-игра лабиринт, с получением концовок и текстовыми описаниями всего происходящего.

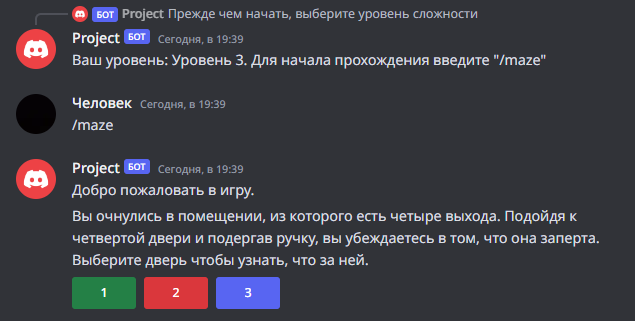
Как уже можно было догадаться, в проекте был реализован сайт, а также бот для дискорда. Бот представляет собой небольшую текстовую игру, а сайт предоставляет информацию о прохождениях пользователя.

Алгоритм работы приложения с пользователем представляется следующим образом: сначала пользователь авторизуется или регистрируется с помощью бота, затем начинает проходить игру, в ходе которой он может получить 14 разных концовок, из которых только одна является выигрышной. Бот использует кнопки, поля выбора и другое. После получения любой концовки бот отправляет ссылку на сайт, который авторизует пользователя и перенаправит его на нужную концовку с помощью специальной одноразовой комбинации, которая используется в адресной строке.

Перейдя по ссылке, вы попадете на сайт с описанем полученной концовки, картинокй к ней, комментариями ко всем концовкам и формой для отправки комментария, которая также включает в себя оценку по десятибалльной шкале (которая еще включает и 0!). На этом сайте вы также можете найти все свои прохождения (которые сделали в этой учетной записи), отсортировать их и вернуться к окну, на которое изначально бот и дает ссылку.

Предусмотрены варинаты перехода по уже использованной ссылке, использование ссылки, которой и не было, а также просто перехода на несуществующую страницу. Для работы сайта использовались вполне стандартные библиотеки для работы с базой данных и реализации сайта, а также другие для некоторых мелочей: flask, flask\_wtf, wtforms, flask\_login, flask\_restful, sqlalchemy, sqlalchemy\_serializer, werkzeug, datetime, random. Для работы бота: disnake и random.

В коде сайта, кроме реализации авторизации с помощью перехода по ссылке от дискорд-бота. Но при этом это и является одной из главных проблем, ведь по ссылке может перейти другой пользователь, ссылка одноразовая да и по реализации, наверняка, были более простые методы. В остальном в коде сайта нет больше особенностей – просто обычные обработчики адресов.

Бот реализован в классе USER в файле discord\_bot\_project.py. Для общения пользователя с ботом используются кнопки, а также меню с выбором сложности (которая, правда, пока ни на что не влияет). С помощью кнопок и текстовых команд пользователь и проходит игру.

Это и является всей реализацией данного проекта.