**Пояснительная записка YLStudent’sChat**

**Авторы: Щетинин Михаил, Беликова Люба.**

Идея проекта: Если кратко, то результатом данного проекта должен стать сервис для поиска единомышленников, учащихся в яндекс.лицее, для совместных проектов как в самом лицее, так и в будущих проектах.

Все крупные разработки в профессиональном программировании ведутся командами, и ведь действительно, легче делать что-то, когда каждый отвечает за свою часть и делает ее на 100%, а не, когда один человек мечется от back-end к front-end, потом к логистике, потом снова к дизайну и в итоге продукт получается неполностью проработанный, так как, когда вся работа лежит на плечах одного человека, то сложно уследить за всем сразу.

Из этих соображений в яндекс.лицее есть возможность создавать отчетные проекты командами. Но есть один нюанс: «Как найти людей, которые тебе подходят?». Кажется, что это задача легковыполнима – подошел к паре ребят из своей группы, да и создали команду. Но давайте посмотрим шире.

Мы занимаемся в яндекс.лицее всего два раза в неделю, да и в это время какое-то более менее нормальное общение не с кем наладить нельзя, так как мы все время занятий проводим в компьютерах, а людям очень тяжело начать общаться вживую с людьми, с которыми они до этого не имели практически никаких контактов. В нашей нынешней ситуации ко всему прочему прибавляется еще и то, что мы занимаемся онлайн, то есть шанса установить с кем-то из своей группы связь практически нет(про другие группы я вообще молчу). И даже, если вы не из стеснительных и легко можете начать общение с человеком, есть шанс, что в вашей группе не найдется подходящего для вашего проекта человека, а может вы просто не сойдетесь с одногруппниками характерами.

Вы будете пытаться найти людей из других групп(это не запрещено правилами яндекс.лицея), но найти связь с ними очень сложно, ведь в лицее нет никаких специальных сообществ или чатов для поиска напарников.

Для того, чтобы решить эту проблему, мы создали YLStudent’sChat - специализированный сервис для поиска единомышленников и общения учеников яндекс.лицея.

В итоге, цель нашего проекта – создать удобный сервис для общения людей из разных групп и даже городов, сделать возможным поиск напарников для проектов по всему яндекс.лицею, не ограничиваясь только своей группой.

Описание реализации: back-end приложения написан на Flask. Для оформления сайта использовался html, css. В программе используется метод шаблонов. Присутствует база данных sqllite. Вся программа состоит из отдельных функций, каждая из которых подключена к своей страничке. Вот основные функции:

1)Начальная страница (app.route(“/“)) – страничка, которая появляется, когда пользователь только входит на сайт. На ней он может авторизироваться и получить доступ ко всем главным функциям сервиса.

2)Регистрация (app.route(“/register“) – страничка, которая используются для регистрации новых пользователей. На этой страничке, как и при авторизации используется технология flask-login, а также база данных sqllite. При регистрации пользователь может выбрать себе аватарку, эту функцию надо было отдельно прописывать с помощью jacascript. Все данные о пользователе хранятся в табличке users базы данных под уникальным id.

3)Чаты (app.route(“/chat”)) – одна из двух главных страничек сайта. На ней пользователе могут общаться. Для всех пользователей предусмотрен общий чат, но также можно создавать свои чаты, редактировать и удалять их. Все данные о чатах также хранятся в базе данных. Это страничка была самой трудной в написании, так как для создания полноценного чата, с отображением сообщений в реальном времени, надо было изучить websockets(библиотека flask-socketio). Чтобы было понятнее, это можно объяснить так: «с помощью http запроса пользователь отправляет сообщение на удаленный сервер, а тот возвращает ответ только этому пользователю. Websockets же позволяют серверу отправлять сообщению сразу всем пользователям подключенным к определенной страничке. Этой методики нет в нашем курсе, так что пришлось много гуглить, и изучать, но методом проб и ошибок, после десятков дней работы, мы смогли реализовать real-time чат. Добавили удаление сообщений в реальном времени, отображение времени отправки сообщений, а также аватарки пользователя, который это сообщение отправил. Конечно, кроме питона socketio надо было добавить и в виде js кода, так что пришлось разбираться в основах javascript.

Websockets – являются самой интересной фишкой проекта, потому что с ее помощью мы сумели добиться чата как в самых популярных мессенджерах(WhatsApp, Telegram).

4)Анкеты (app.route(“/ankets”)) – вторая из двух главных страничек сайта. На ней пользователь видит все анкеты, опубликованные учениками яндекс.лицея. Анкеты отсортированы по группе обучения и дате публикации, так что пользователь всегда будет видеть самые свежие анкеты. На анкетах есть информация о давшем анкету пользователе, теме проекта и группе обучения автора анкеты. При нажатии на анкету пользователя перебрасывает на страничку полного описания того, что ищет автор анкет, люди с какими навыками ему нужны. Если пользователь нажмет на кнопку «откликнуться на анкету», то автоматически создастся чат с ним и автором анкеты, в котором они смогут обсудить все нужные детали и договорится о совместной работе. Кроме того, хочу отметить, что если пользователю не понравится анкета и он захочет вернутся назад ко всем анкетам, то кнопка «отмена» с помощью js прописана так, что пользователь возвращается именно к тому моменту, куда он долистал, чтобы не пришлось вновь начинать поиск сначала.

5)Профиль (app.route(“/profile”)) – профиль пользователя. На этой страничке пользователь может изменить личные данные, пароль, аватарку, удалять и редактировать анкеты, а также выйти из аккаунта или удалить его.

В итоге строк кода: python: ~1000; html: ~1000; css: 2161

Библиотеки, необходимые для запуска:

aniso8601==9.0.1

bidict==0.22.0

certifi==2021.10.8

charset-normalizer==2.0.12

click==8.1.0

colorama==0.4.4

Flask==2.1.0

Flask-Login==0.5.0

Flask-RESTful==0.3.9

Flask-SocketIO==5.1.1

Flask-WTF==1.0.0

greenlet==1.1.2

heroku==0.1.4

html5==0.0.9

idna==3.3

importlib-metadata==4.11.3

itsdangerous==2.1.2

Jinja2==3.1.1

MarkupSafe==2.1.1

Pillow==9.0.1

pip==21.1.2

python-dateutil==1.5

python-engineio==4.3.1

python-socketio==5.5.2

pytz==2022.1

requests==2.27.1

setuptools==57.0.0

six==1.16.0

SQLAlchemy==1.4.32

SQLAlchemy-serializer==1.4.1

urllib3==1.26.9

webencodings==0.5.1

websockets==10.2

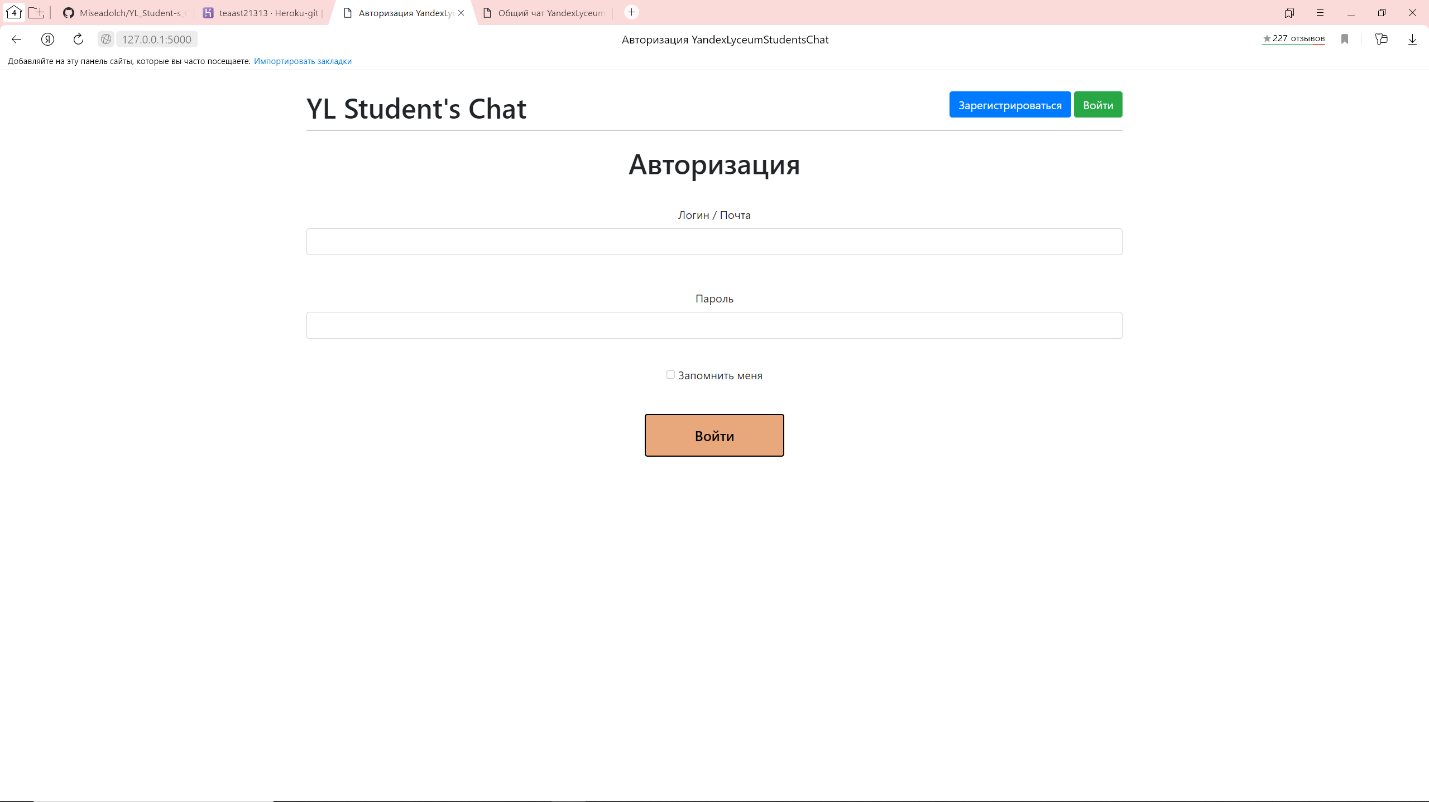
Werkzeug==2.0.3

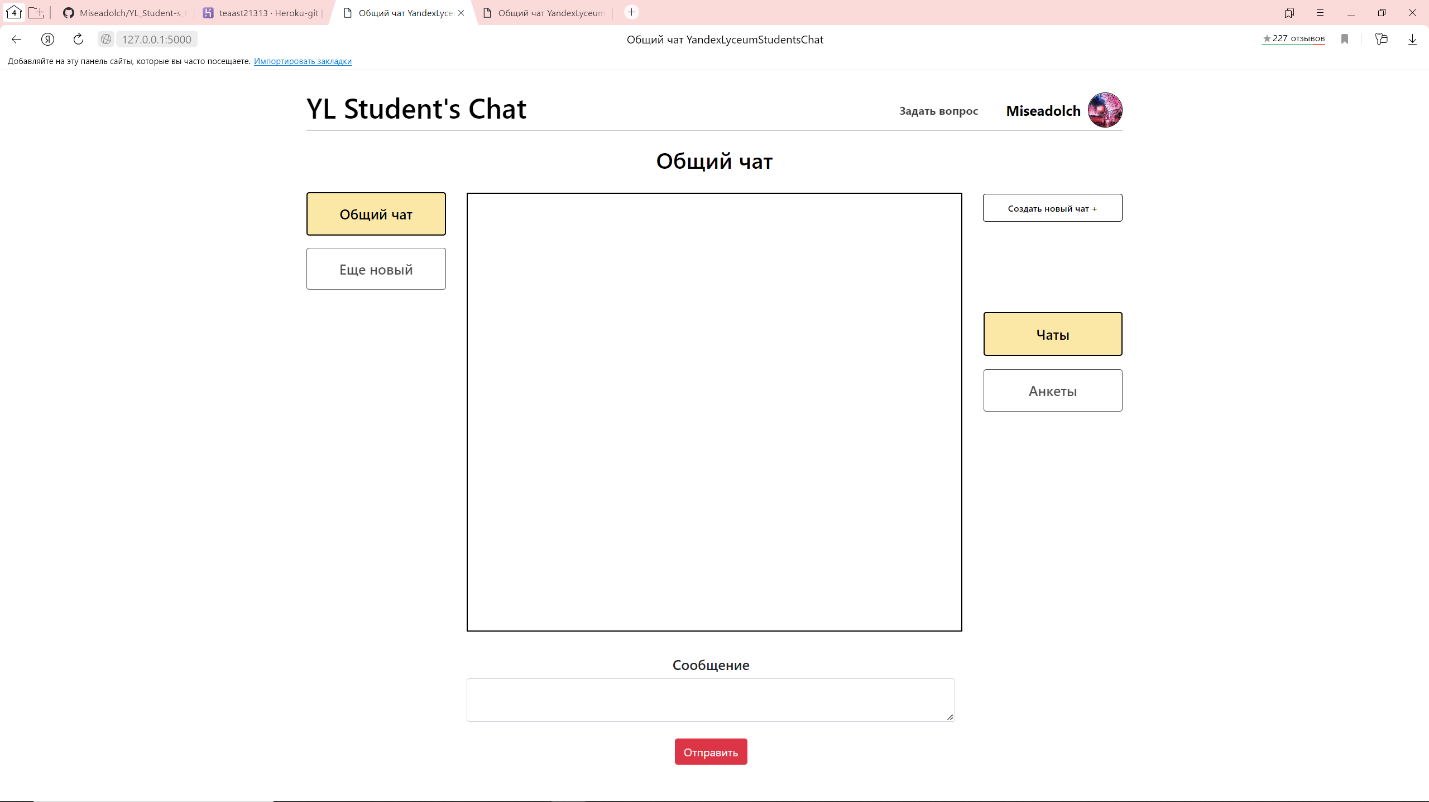
wheel==0.36.2

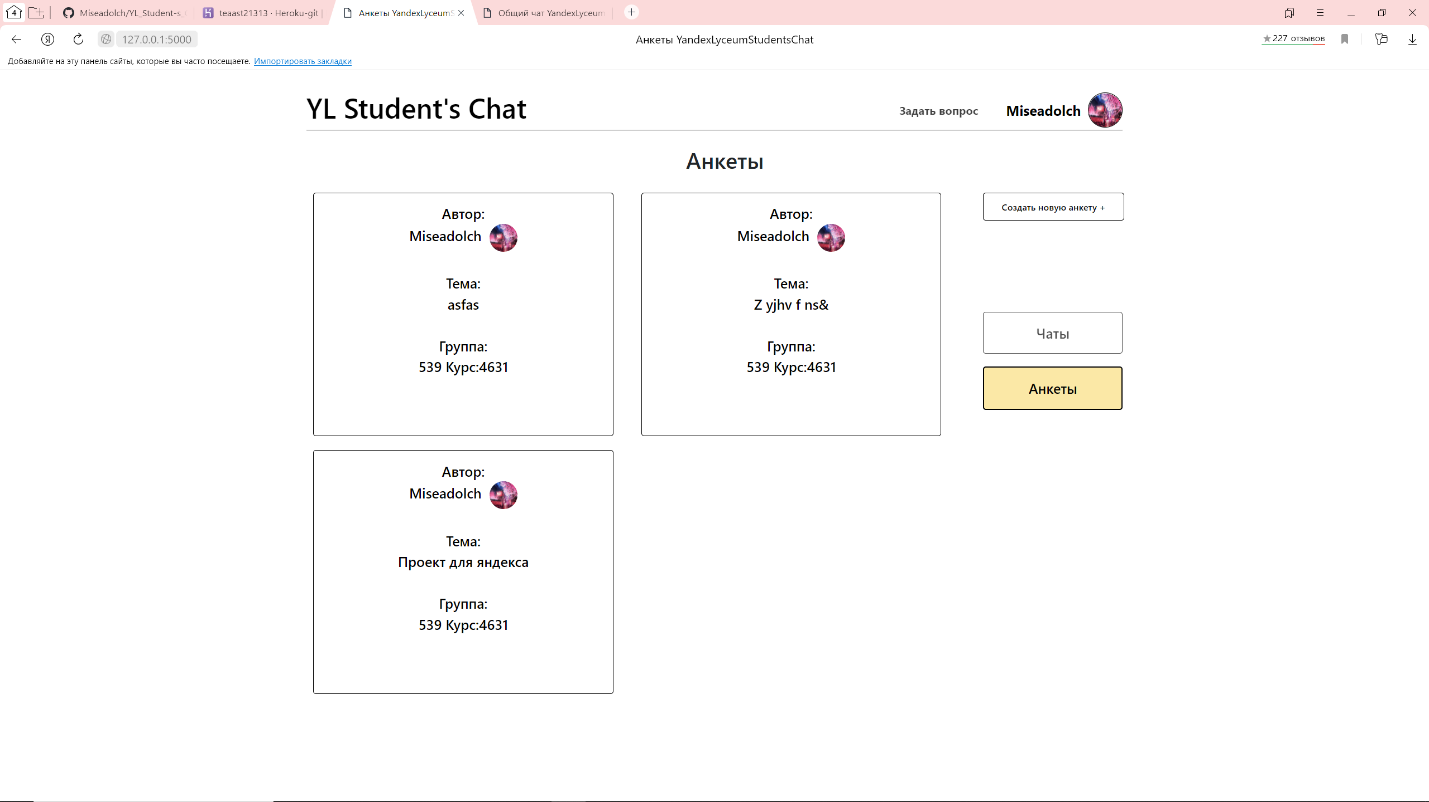
WTForms==3.0.1

zipp==3.7.0

Также все зависимости находятся в файле requirements.txt на гитхабе

Авторизация

Чаты 

Анкеты

Гитхаб проекта: <https://github.com/Miseadolch/YL_Student-s_Chat>