**Тема проекта:**

Разработка системы мониторинга перемещения курьеров.

**предметная область:**информатика

**Выполнил:**Ученик 10А класса - Селезнёв Роман

**Руководитель:**Учитель информатики - Чубакова Т.В.

**Муниципальное Бюджетное Общеобразовательное Учреждение**

**Средняя Общеобразовательная Школа №55**

**Г.Брянск**

**2021 г.**

**Введение**

**Цель проекта:** Создать систему мониторинга перемещения курьеров.

**Целевая аудитория:** Администрация кафе/ресторанов, курьеры, доставляющие еду.

В современном обществе огромное множество людей предпочитает заказать еду на дом походу в ресторан/кафе. Их можно понять, ведь это весьма удобно. Не нужно одеваться, идти/ехать по отвратительной погоде до любимого заведения, тратя при этом своё личное время.

В то же время еда сама себя не доставит. Для этого есть специальные люди - курьеры. Им и так приходится не легко, доставляя заказы, а теперь представьте, что они ещё и не имеют наглядного представления о том, где находится заказ(имеется лишь адрес, нельзя посмотреть местоположение заказа на карте), как до него добраться, какие заказы доступны на данный момент времени для выполнения и т.д.

Также у администрации может возникнуть вопрос относительно месторасположения их курьеров, их статуса и текущих заказов.

Все вышеперечисленные проблемы призвана решить система мониторинга перемещения курьеров.

**Основные этапы реализации проекта:**

1. Планирование приложения.
2. Поиск и изучение технологий, которые помогут в реализации проекта.
3. Создание MWP(минимальный работающий продукт).
4. Поиск и устранение ошибок.
5. Добавление дополнительных функций в проект.
6. Работа над дизайном проекта.
7. Помещение финальной версии проекта на удалённый сервер.

**Технологии, используемые в проекте.**

1. Высокоуровневый язык программирования Python 3.

Используется как основной инструмент при создании данного веб-приложения.

1. Язык гипертекстовой разметки HTML 5.

Используется для создания и редактирования веб-страниц в данном проекте.

1. Каскадные таблицы Стилей CSS.

Используются для создания красивого и понятного интерфейса данного веб-приложения.

1. API карт 2Gis.

Используется для реализации карты и всех действий, связанных с ней.

1. Фреймворк Flask.

Используется для реализации веб-сервера на языке Python 3.

1. Фреймворк Bootstrap 5.

Используется для создание более качественного дизайна веб-приложения.

1. Шаблонизатор Jinja 2.

Используется для упрощения веб-разработки. Позволяет выполнять код прямо в HTML шаблоне.

1. Удалённый веб-сервер Heroku.

Используется в качестве удалённого сервера, на котором будет располагаться веб-приложение.

**Разработка продукта.**

Я разрабатывал каждую часть проекта в отдельном модуле, чтобы в дальнейшем облегчить процесс исправления ошибок и процесс добавления новых функций продукта. Также это сделает код более читабельным, и, как следствие, более понятным для восприятия.

Для начала я создал модуль с СУБД( систему управления базой данных ) и БД( базу данных ) чтобы хранить в ней все данные моего проекта. СУБД включает в себя функции выдачи информации( местоположение курьера, все заказы и т.д.( подробнее см: ->приложение->папка с кодом->data\_base2.py) ). Также в СУБД находятся классы Курьера и Администратора, благодаря которым реализованы некоторые функции веб-приложения.

После я занялся разработкой сервера и всех прилагающихся к нему функций( вход в систему, мониторинг курьеров и т.д. ( подробнее см: ->приложение->папка с кодом->main.py ) ).

Далее я создал HTML шаблоны и стили для них( подробнее см: ->приложение->папка с кодом->папка templates( в ней все HTML шаблоны )/папка static->styles( в ней все CSS стили )).

После вышеперечисленных действий начался процесс устранения ошибок и добавления новых функций в продукт.

**Проблемы при разработке продукта.**

Первой проблемой стал выбор способа реализации БД. Выбор был между языком структурированных запросов к базе данных(SQL) и стандартной работой с файлами в языке Python в связке с библиотекой для считывания двоичных данных (Pickle).

Я выбрал второй вариант т.к. лучше знаю данные технологии.

Следующей проблемой стала реализация карты. Для начала я использовал библиотеку folium, но позже пришлось заменить её на API карт 2gis, так как реализация карт через folium требовала написания отдельного модуля(подробнее см: ->приложение->папка с кодом->map\_module.py), а API карт 2gis можно было сразу же подключить к проекту и использовать без написания отдельного модуля.

Также карты folium-а были статическими, что вызывало затруднения с обновлениями данных.

На этапе устранения ошибок было множество мелких проблем, устрание которых редко занимало более 30-ти минут.

**Итоги.**

Продукт создан и готов к применению на практике, что значительно облегчит работу курьерам и администрации того или иного заведения.

**Приложение.**

Содержит:

1. папку с полным исходным кодом продукта.
2. презентацию.
3. текстовый файл с ссылками на доп. источники.