СОДЕРЖАНИЕ

[ВВЕДЕНИЕ](#_mqw0ck8nzvz9) **2**

[1](#_vav131q3oiy9) **3**

[1.1 Обзор проекта](#_7bpmz094y9u) **3**

[1.2 Сроки и этапы проекта](#_84mqc3kexcfu) **3**

[1.3 Создание репозитория](#_4ljstfsngzvr) 4

[1.4 ER-диаграмма](#_n2gjd92khh9c) **4**

[2 Проектная организация](#_3uig5zq0vlnc) **6**

[3 Управление проекта](#_dawafoqbobb) **7**

[4 Технический процесс](#_b9pmndi2z014) **8**

[5 Сметы, график, бюджет](#_7y1jrp2etnbo) **9**

[ЗАКЛЮЧЕНИЕ](#_cs0r7e5ef34y) **10**

# ВВЕДЕНИЕ

Хотите стать волшебником для самого себя и быть уверенным, что любое ваше желание сбудется на 100%? Тогда вам просто необходимо наше приложение – именно оно станет инструментом реализации. Не верите, что это возможно? А вот другие поверили и уже давно пользуются. Работая с данным инструментом каждый день не более 1 минуты, и вы кардинально сможете изменить свою жизнь к лучшему!

# 1

# 1.1 Обзор проекта

Целью данного проекта является создание приложения трекера желаний Make a Wish.

Пользователь может авторизоваться, выбрать вещи, которые бы хотел приобрести сам или чтобы их подарили друзья. Смысл этого приложения - облегчить друзьям и знакомым пользователя выбор подарков на какие-либо праздники, тем самым подарив нужную и необходимую вещь, а также держать все свои желание в одном месте, с возможностью знать цену товара и место, где можно его приобрести. Также присутствует возможность “забронировать” желание друг друга, в личном кабинете пользователей отображается данная отметка, чтобы пользователи знали, чего ожидать на ближайший праздник, однако имя пользователя будет скрыто для сохранения интриги.

# 1.2 Сроки и этапы проекта

Работа над проектом проводилась в бригаде, состоящей из Боевец Алины и Чухровой Анны, и была поделена на этапы с конкретными сроками и задачами, которые необходимо решить. Лидером бригады является Алина.

Также между членами бригады были распределены роли. Алина отвечает за визуальную часть приложения, backend разработку, тестирование приложения и написание отчета. В свою очередь Анна отвечает за backend разработку, хранение данных, также за тестирование приложения и написание отчета.

Сроки и этапы представлены в таблице 1.

Таблица 1 - Этапы проекта

| Этап | Срок | Задачи |
| --- | --- | --- |
| 1 | 10.02.2022 - 17.02.2022 | Выбор темы |
| 2 | 17.02.2022 - 24.02.2022 | Обозначение задач в проекте между членами бригад, составление плана |
| 3 | 24.02.2022 - 10.03.2022 | Структура приложения,  ER- диаграмма, создание репозитория, выбор языка и среды программирования |
| 4 | 10.03.2022 - 24.03.2022 | Визуальная часть приложения, выбор базы данных |
| 5 | 24.03.2022 - 07.04.2022 | Хранение, добавление и удаление желаний |
| 6 | 07.04.2022 - 21.02.2022 | Добавление отметки “забронировано”, редактирование желаний |
| 7 | 21.04.2022 - 05.05.2022 | Создание отчета |
| 8 | 19.05.2022 | Демонстрация проекта |

# 1.3 Создание репозитория

Для совместной работы был создан репозиторий на GitHub, где хранится весь код и источники, необходимые для работы. Репозиторий представлен на рисунке 1.

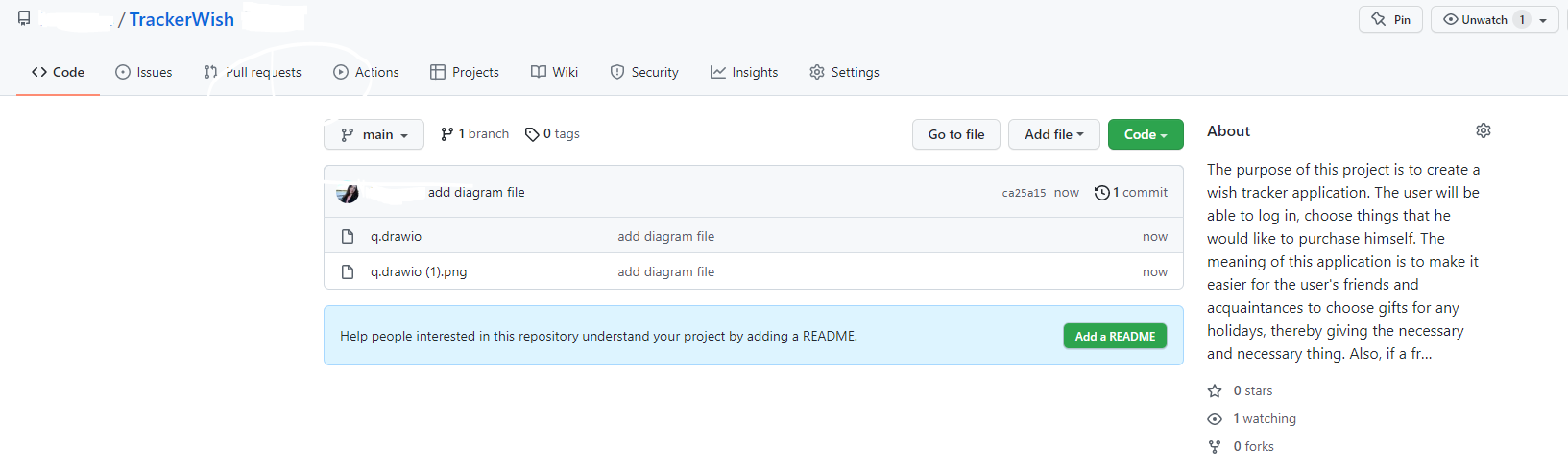


Рисунок 1 - Репозиторий

# 1.4 ER-диаграмма

В первую очередь необходимо составить ER-диаграмму– это концептуальная модель модели данных высокого уровня. ER моделирование помогает систематически анализировать требования к данным для создания хорошо спроектированной базы данных. ER-диаграмма отображает отношения набора сущностей, хранящиеся в базе данных. Другими словами, мы можем сказать, что ER-диаграммы помогут вам объяснить логическую структуру баз данных. Цель ER Diagram – представить инфраструктуру структуры объекта. Таким образом, считается наилучшей практикой завершить моделирование ER перед внедрением вашей базы данных [https://coderlessons.com/tutorials/bazy-dannykh/osnovy-subd/5-diagramma-er-v-subdhttps://coderlessons.com/tutorials/bazy-dannykh/osnovy-subd/5-diagramma-er-v-subd].

ER-диаграмма представлена на рисунке 2.

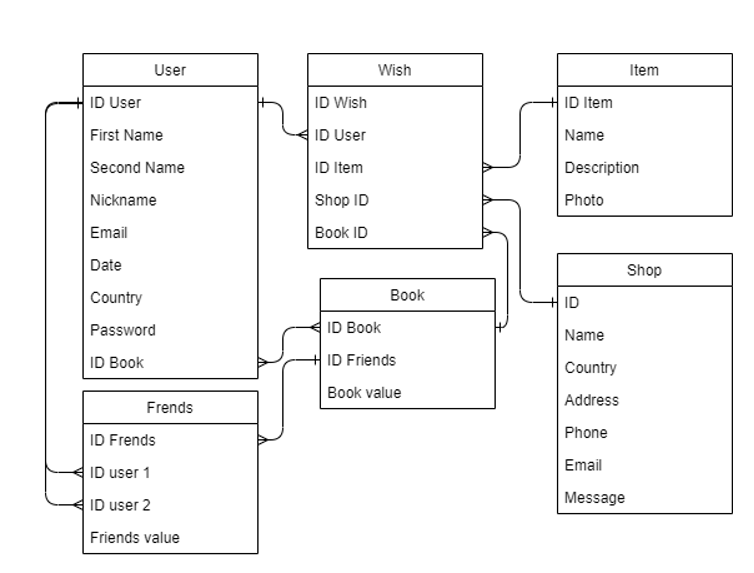


Рисунок 2 - ER-диаграмма

1.5 Выбор среды разработки, языка программирования и базы данных

Для разработки веб-приложения необходимо выбрать среду разработки, язык программирования и базу данных.

С связи с опытом работы на языке программирования Python, языком программирования был выбран именно он. Python — это скриптовый язык программирования. Он универсален, поэтому подходит для решения разнообразных задач и многих платформ, начиная с iOS и Android и заканчивая серверными ОС. Это интерпретируемый язык — он не компилируется, то есть до запуска представляет из себя обычный текстовый файл. Программировать можно практически на всех платформах, язык хорошо спроектирован и логичен. Разработка идёт в разы быстрее, потому что кода здесь куда меньше, чем на других языках [https://skillbox.ru/media/code/dlya\_chego\_nuzhen\_python].

Для написания приложения на Python были выбраны две среды разработки - это PyCharm (для Анны ) и Visual Studio Code (для Алины). Отличий в этих двух средах практически нет, выбор был сделан из опыта работы для каждого члена бригады.

Логотипы языка и сред программирования представлены на рисунке 3.

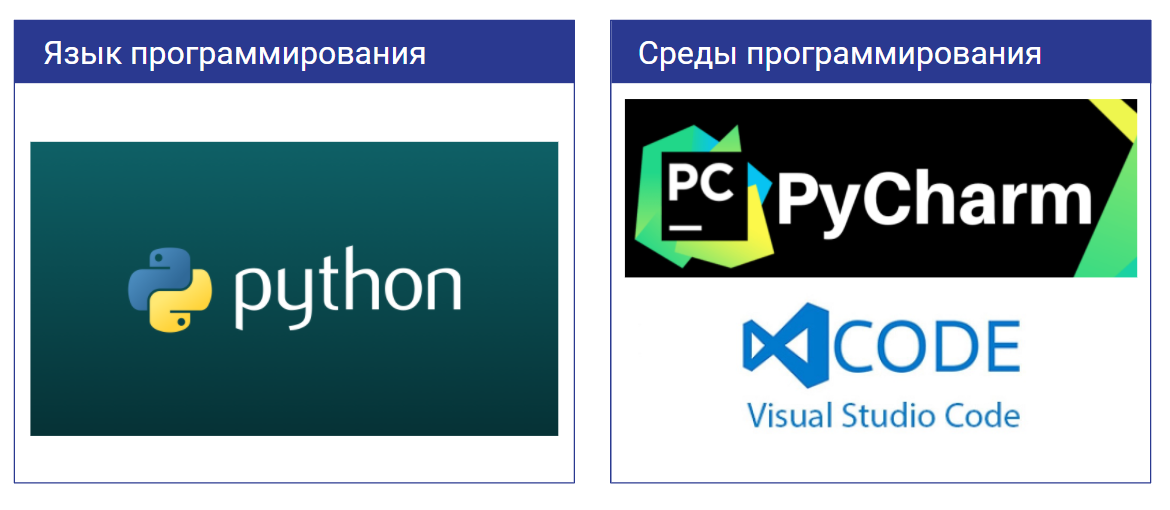


Рисунок 3 - Язык и среды программирования

Для хранения данных также необходима база данных, в данном проекте будет задействована SQLite3.

SQLite3 (часто говорят просто SQLite) — это часть стандартного пакета Python 3, то есть оказывается, что для работы с SQLite из Python-кода не нужно устанавливать некое клиент-серверное ПО, не нужно поддерживать работу какого-то сервиса, отвечающего за работу с СУБД. Достаточно лишь импортировать модуль sqlite3 и приступить к его использованию в программе, получив в своё распоряжение систему управления реляционными базами данных [https://habr.com/ru/company/ruvds/blog/514538/].

Логотип базы данных представлен на рисунке 4.



Рисунок 4 - База данных

Также необходимо отметить, что данное веб-приложение будет разрабатываться с помощью Flask. [Flask](http://flask.pocoo.org/) — это небольшой и легкий веб-фреймворк, написанный на языке Python, предлагающий полезные инструменты и функции для облегчения процесса создания веб-приложений с использованием Python. Он обеспечивает гибкость и является более доступным фреймворком для новых разработчиков, так как позволяет создать веб-приложение быстро, используя только один файл Python. Flask — это расширяемая система, которая не обязывает использовать конкретную структуру директорий и не требует сложного шаблонного кода перед началом использования.Flask использует [механизм шаблонов Jinja](http://jinja.palletsprojects.com/) для динамического создания HTML-страниц с использованием знакомых понятий в Python, таких как переменные, циклы, списки и т. д. [<https://www.digitalocean.com/community/tutorials/how-to-make-a-web-application-using-flask-in-python-3-ru>]

Логотип Flask представлен на рисунке 5.

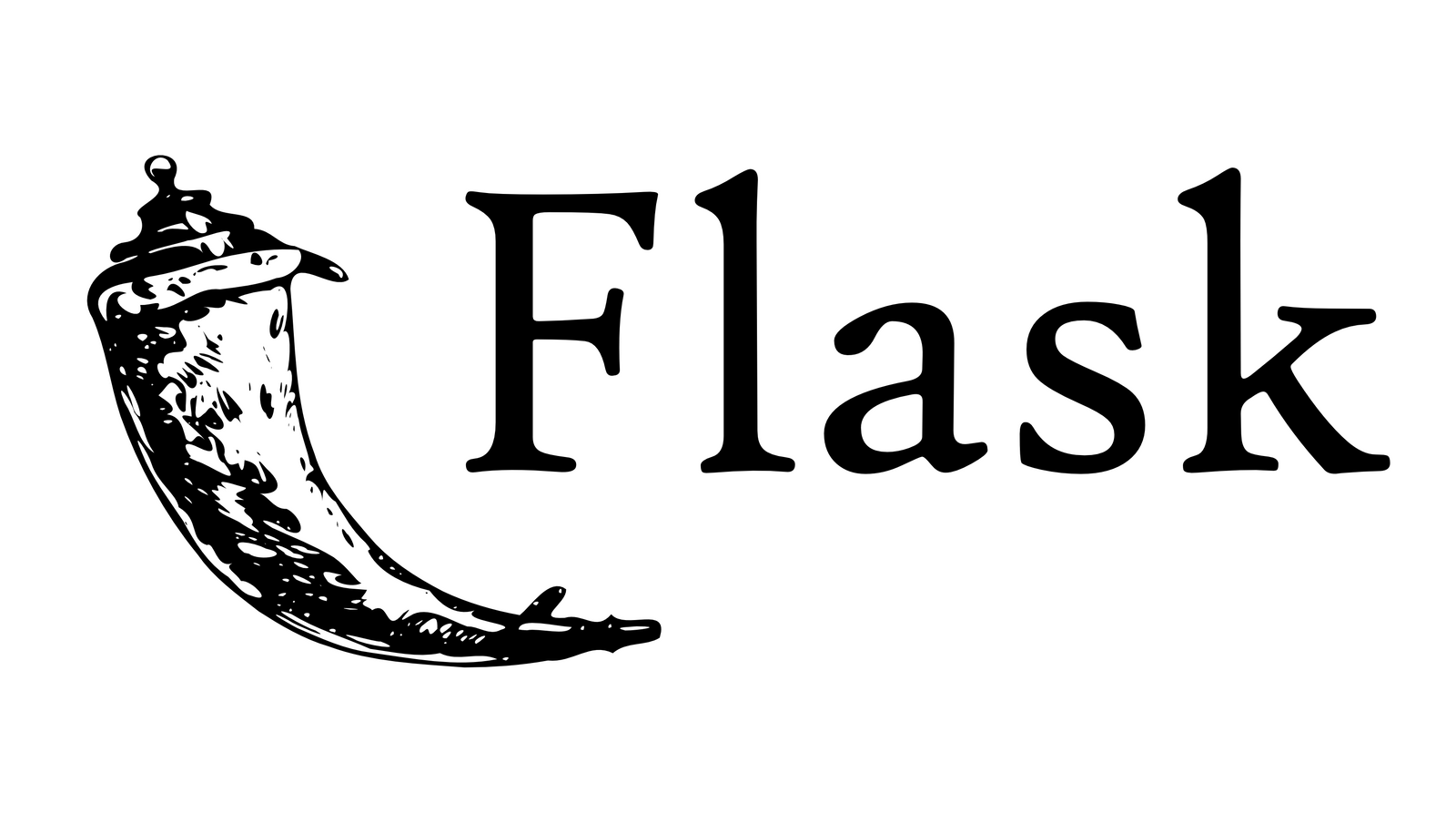


Рисунок 5 - Flask

# 2 Проектная организация

# 3 Управление проекта

Таблица - Релизы

| Релиз | Функционал |
| --- | --- |
| 0.1.0 |  |
| 1 |  |

Таблица - Разделение по языкам

| Язык программирования | Количество строк |
| --- | --- |
| Python |  |
| Html |  |
| Css |  |

# 4 Технический процесс

# 5 Сметы, график, бюджет

# ЗАКЛЮЧЕНИЕ