Пояснительная записка к проекту Flask

“Choose Your Consequence”

Погонцева Данила и Шамиля Анаева

Содержание

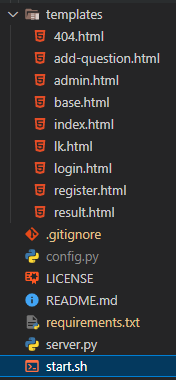
* Введение
  + Идея проекта
  + Процесс использования
* Описание реализации
  + Структура приложения
  + Особенности реализации
  + Используемые технологии
* Заключение
  + Планы на будущее

1.1 Идея проекта

**Choose Your Consequence** - интерактивная платформа, на которой пользователи могут отвечать на философские вопросы, предлагающие два выбора с последствиями, каждый из которых иллюстрирует два разных противоречивых исхода. Пользователи будут заинтересовывать изучать противоречивые моральные и этические вопросы, а также увидеть, какие выборы делают другие люди в подобных ситуациях.

1.2 Процесс использования

Пользователь, заходя на главную страницу сайта, увидит вопрос и два варианта-исхода. Он может сделать свой выбор, после этого перейдет на страницу вопроса и увидит соотношение ответов пользователей: сколько человек выбрало первый вариант, сколько – второй. Пользователь может оставлять комментарии, если войдет в учетную запись, а также лайкать чужие. Кроме того, пользователь может зайти в личный кабинет и увидеть свои ответы на вопросы. Администратор может добавлять вопросы, изменять существующие и удалять их.

Изображение выглядит как текст

Автоматически созданное описание2.1 Структура приложения

* server.py - главный файл приложения
* requirements.txt - файл с перечнем зависимостей
* data - папка с используемыми в приложении моделями и API методами
* static - папка, содержащая статичные ресурсы приложения, такие как библиотеки BootStrap и jQuery, изображения, аватары пользователей и т.п.
* db - папка, в которой находятся база данных
* forms - содержит все формы приложения
* templates - папка со всеми HTML шаблонами
* сonfig.py – файл конфигурации
* start.sh – файл, необходимый для запуска приложения на glitch.com
* LICENSE – файл публичной лицензии проекта

2.2 Особенности реализации и функции

* Возможность регистрации и входа
* Возможность регистрации в качестве администратора с помощью специального токена
* Шифрование пароля пользователя
* Простое добавление новых вопросов, возможность их редактирования и удаления
* Реализован API для взаимодействия с базой данных
* Использование форм и моделей
* Реализована загрузка аватаров пользователей при регистрации
* Возможность оставлять комментарии и лайкать чужие

2.3 Используемые технологии

* Python
* Flask
* Flask-WTF
* SQLAlchemy
* Flask-Login
* Pillow
* HTML
* CSS
* Bootstrap
* JavaScript
* jQuery

3 Планы на будущее

* Добавление возможности редактирования комментариев
* Добавление тёмной темы сайта
* Добавление функции предложения постов от пользователей
* Добавить просмотр автора вопроса и просмотр профилей
* Добавить возможность отвечать на комментарии, создавая ветки ответов
* Другие улучшения