Оглавление

1. Структура проекта	2
2. Зависимости, требования проекта	
2.1 Зависимости	
2.2 Структура БД и таблиц	
3. Запуск проекта	
3.1 Установка переменных окружения	
3.2 Запуск сервера	
4. Клиентские запросы	
5. Общая логика сервиса	
5.1 Запросы find_predictions	
5.2 Запросы find_data	
5.2 Surpoch Inia_data	•••••

1. Структура проекта

Проект содержит несколько модулей:
data_service.py-главный модуль, интерфейс API
find.py-обработка запросов
crawler.py-модуль взаимодействия с commoncrawl.org
prepare_write_data.py-модель обработки и записи информации в БД

logging.conf — конфигурация журналирования (лога) requirements.txt — зависимости проекта server.log — журнал отладочных сообщений (лог) setenv.sh — учётные и конфигурационные данные

2. Зависимости, требования проекта

2.1 Зависимости

Bce зависимости, для работы написанного сервиса, указаны в файле requirements.txt. Устанавливаются через систему управления пакетами рір командой «рір install -r requirements.txt» в директории проекта Для хранения информации требуется СУБД Postgresql 12.3

2.2 Структура БД и таблиц

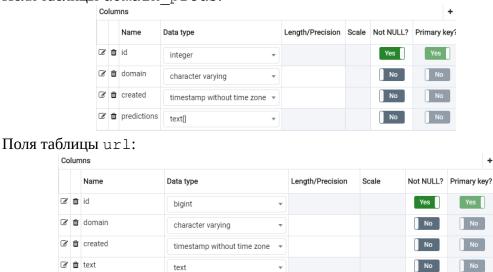
☑ is_accompanying

☑ mi url

boolean

character varying

Сервисом используется одна БД с названием data_service, которая имеет две таблицы: Поля таблицы domain preds:



No

No

No

Для работы с БД должен быть создан пользователь с паролем и соответствующими ролями для добавления информации в БД

3. Запуск проекта

3.1 Установка переменных окружения

Так как сервисом используются учётные данные для доступа к БД и конфигурационные параметры, то в целях безопасности и гибкости проекта все данные были вынесены в файл setenv.py, который представляет из себя bash-скрипт. Перед началом запуска сервиса, находясь в директории проекта, необходимо выполнить скрипт командой «. ./setenv.sh». После выполнения скрипта в рабочее окружение ОС экспортируются все переменные, которые указаны в скрипте. Код сервиса считывает переменные окружения посредством библиотеки оs (os.getenv(<имя_переменной>)). Скрипт set.env занесён в .gitignore и выгрузке в удалённый git-репозиорий не подлежит.

3.2 Запуск сервера

Сервис запускается на веб-сервере uvicorn. Для запуска необходимо выполнить следующую команду в директории проекта:

```
«uvicorn data_service:app --reload --host <ip> --port <port>»,
где: <ip>- ip-адрес сервера, на котором будет запущен сервис
<port> - порт сервера
```

например: команда «uvicorn data_service:app --reload --host 172.16.10.1 --port 22345» запускает сервер по адресу 172.16.10.1 на порту 22345. Следовательно все запросы от клиента должны приходить на адрес 172.16.10.1:22345

4. Клиентские запросы

5. Общая логика сервиса

5.1 Запросы find_predictions

При запросе find_predictions сервис возвращает найденные записи в таблице "domain_preds", в противном случае возвращается сообщение о том, что записей не найдено.

5.2 Запросы find_data

При запросе find_data сервис возвращает найденные записи в таблице "url", в противном случае производится поиск информации по доменному имени в индексах commoncrawl.org. В случае, если информация по доменному имени имеется в индексах commoncrawl.org, то она извлекается, обрабатывается, записывается в таблицу "url", сервис возвращает найденные записи в таблице "url".

В случае, если информации по доменному имени отсутствует в индексах commoncrawl.org, то возвращается сообщение о том, что записей не найдено.