**OCR Microservice**

**Оглавление**

1. **Описание проекта**
2. **Функционал**
3. **Назначение и примеры использования**
4. **Технологии и зависимости**
5. **Установка и запуск**
6. **Остановка работы скрипта**
7. **Контакты**

**Описание проекта**

Данный микросервис создан для **распознавания текста (OCR)** на изображениях, используя библиотеку [Tesseract](https://github.com/UB-Mannheim/tesseract/wiki) и фреймворк [FastAPI](https://fastapi.tiangolo.com/).  
Он позволяет загружать изображение (через веб-интерфейс или API) и получать расшифрованный текст, который также сохраняется в базе данных (*в данном случае была использована одна из наиболее простых БД – SQLite)*

**Функционал**

* **Загрузка изображений** через веб-форму (HTML-страница) или REST API.
* **Распознавание текста (OCR)** из изображений с помощью Tesseract V5(последняя актуальная на 22.01.25)
* **Сохранение результатов** в базу данных (SQLite).
* **Просмотр списка распознанных изображений** и деталей (дат, имён файлов, расшифрованного текста) через HTML-шаблоны.
* **API**-функции для работы со сторонними скриптами и сервисами.
* Простые **тесты** (pytest) для проверки базовой функциональности. *Картинки результатов тестов представлены в папке проекта.*

**Возможное назначение и примеры использования**

**Основное:** Микросервис планируется к использованию как составная часть дипломного проекта – Многоформатный нейро-помощник по работе с базой знаний (картинки, документы, БД, динамические ресурсы).

**Дополнительные возможные назначения:**

* **Автоматизация документооборота**: распознавание чеков, счетов, квитанций, сертификатов и хранение результатов.
* **Обработка сканов**: извлечение текстовой информации из отсканированных PDF или изображений.
* **Системы поиска**: предобработка изображений (OCR), чтобы осуществлять полнотекстовый поиск по сканам или фотографиям.
* **Аналитика**: можно собирать распознанный текст (например, квитанции, накладные) и дальше анализировать данные (суммы, даты и т.д.).

**Примененные технологии и зависимости(из описания задания)**

* **FastAPI**: веб-фреймворк для создания веб-интерфейса.
* **SQLAlchemy**: доступ к базе данных (SQLite).
* **pytesseract**: Python-обёртка над Tesseract V5.0 для распознавания текста.
* **PIL (Pillow)**: для чтения изображений.
* **Jinja2**: шаблонизатор для HTML.

Полный список библиотек представлен в файле с окружением. Установить все зависимости можно через файл requirements.txt:

pip install -r requirements.txt

**Установка и запуск**

1. **Установите Tesseract.**

Дистрибутив для установке представлен в папрке проекта.

1. **Клонируйте репозиторий** или скачайте исходный код в папку:

git clone https://github.com/Plastynenko/CVPRO\_project.git

cd CVPRO\_project

1. **Создайте виртуальное окружение** (рекомендуется, чтобы не засорять систему):

python -m venv CVPRO

Активируйте окружение:

# Windows (PowerShell):

CVPRO\Scripts\activate

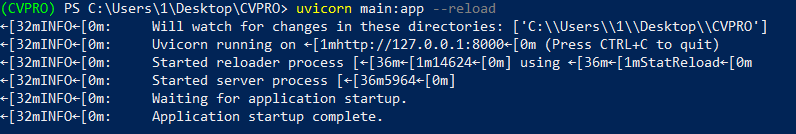
1. **Установите зависимости**:

pip install -r requirements.txt

1. **Запустите приложение** с помощью uvicorn:

uvicorn main:app --reload

Если всё нормально, в терминале будет:



Uvicorn running on http://127.0.0.1:8000

1. **Откройте браузер** и перейдите по адресу <http://127.0.0.1:8000>.

На главной странице вы увидите список ранее обработанных изображений.

**Остановка работы скрипта**

Приложение FastAPI, запущенное через uvicorn, работает в **фоновом режиме** до тех пор, пока вы его не остановите.

Инструкция по остановке скрипта:

* 1. Вернитесь в консоль или терминал, где запущен сервер.
  2. Нажмите **Ctrl + C**, чтобы отправить сигнал прерывания.
  3. Сервер завершит работу.

**Контакты**

* Автор: *Пластуненко Л.Ю.*
* Email: *plastynenko@gmail.com*
* GitHub: github.com/Plastynenko

Дополнительно: Структура обязательных папок проекта:

CVPRO/

├── CVPRO/ # Виртуальное окружение

├── templates/

│ ├── base.html

│ ├── index.html

│ ├── upload.html

│ └── details.html

├── tests/

│ └── test\_main.py

├── main.py

├── database.py

├── models.py

├── schemas.py

├── crud.py

├── ocr\_service.py

├── requirements.txt

└── Readme\_OCR\_Microservice.docx