

Государственное бюджетное профессиональное
образовательное учреждение Московской области
«Физико-технический колледж»

Документация к веб-приложению

«Распознавание лиц по фотографии человека»

Работу выполнил:
Студент группы ИСП-22
Титова София

СОДЕРЖАНИЕ

УСТАНОВКА НЕОБХОДИМЫХ ИНСТРУМЕНТОВ	3
РАБОТА С ГРАФИЧЕСКИМ ПРИЛОЖЕНИЕМ	8

УСТАНОВКА НЕОБХОДИМЫХ ИНСТРУМЕНТОВ

Для начала работы с приложением нужно организовать среду для его запуска. Первым делом скачайте редактор исходного кода Visual Studio Code с [официального сайта](https://code.visualstudio.com). Для этого нажмите на «other platforms» и выберите подходящую для своей системы версию.

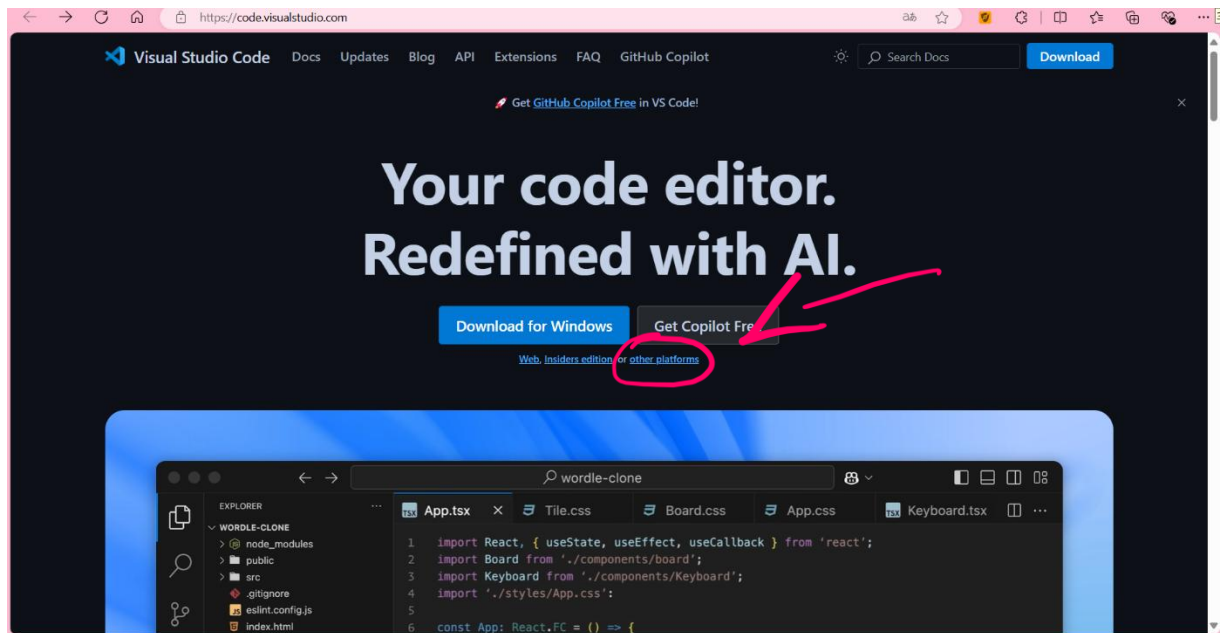


Рисунок 1. Главная официального сайта Visual Studio Code



Рисунок 2. Страница для скачивания Visual Studio Code

Затем найдите скаченный файл в проводнике и установите программу, после чего откройте её от имени администратора, нажав на правую кнопку мыши и выбрав нужный пункт.

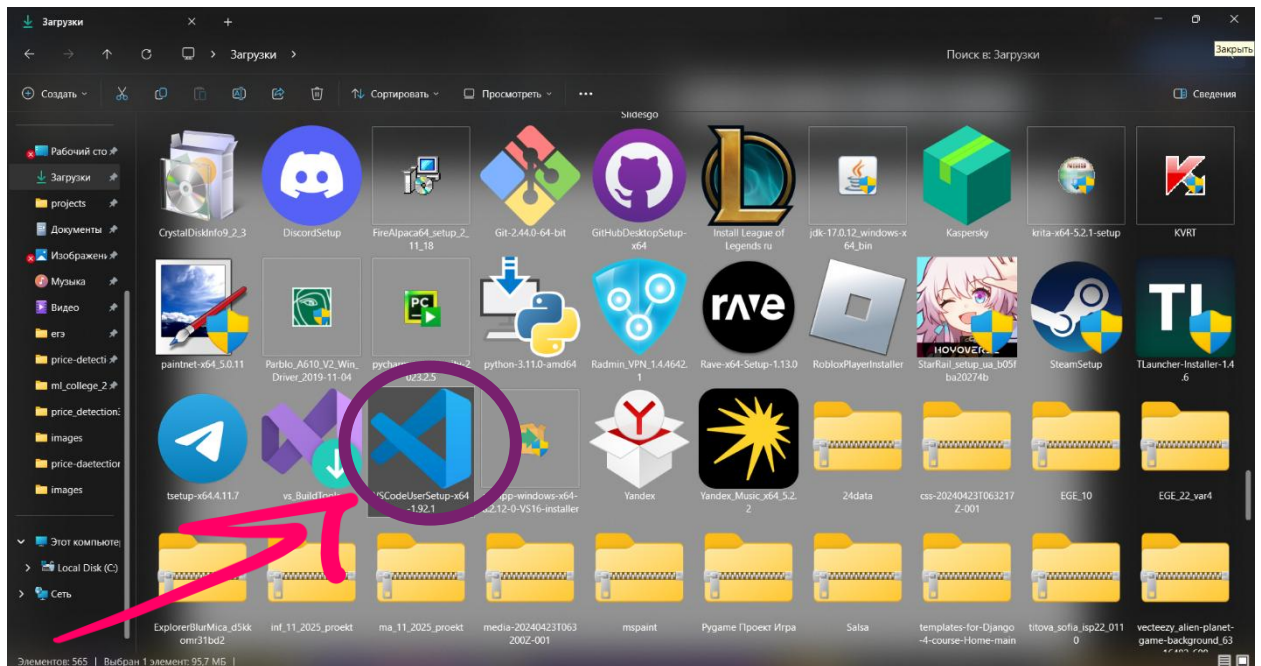


Рисунок 3. Проводник

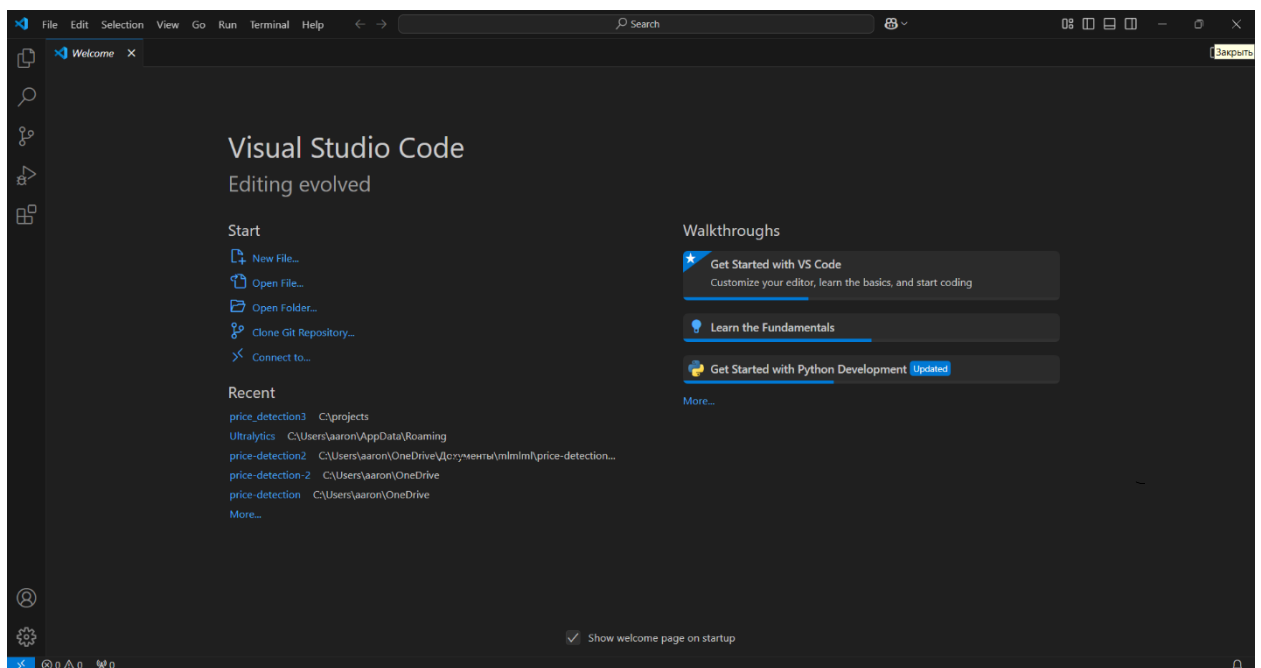


Рисунок 4. Начальное окно Visual Studio Code

Нажмите на «Open Folder...» и выберите нужную папку с файлами приложения.

Необязательно, но желательно сделать виртуальное окружение - изолированную среду для разработки и выполнения программного обеспечения. Она позволяет изолировать зависимости, библиотеки и пакеты для каждого проекта, минимизируя конфликты между различными проектами. Информацию о том, как создать виртуальное окружение, можно найти в [документации самого VS Code](#).

Теперь нужно открыть терминал комбинацией клавиш `ctrl+shift+~` (если комбинация не сработает, нажмите `Terminal > new Terminal`):

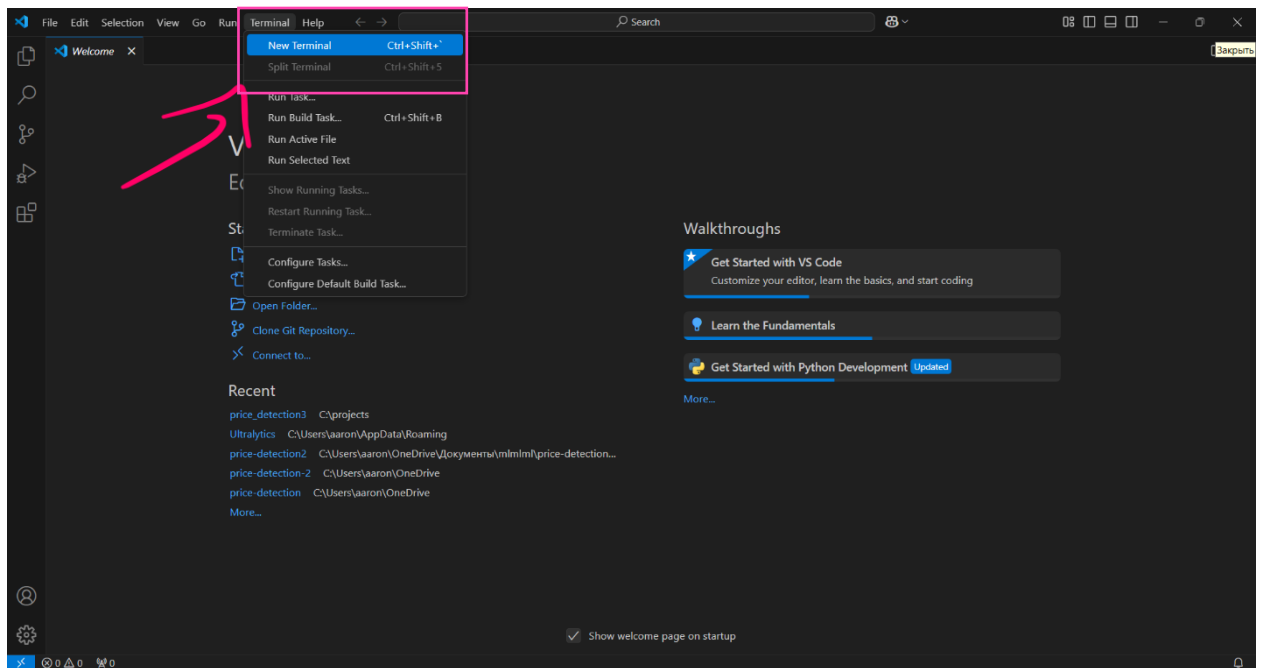


Рисунок 5. Открытие терминала в Visual Studio Code

Теперь можно устанавливать необходимые библиотеки. Это можно сделать, установив файл «requirements.txt». Убедитесь, что файл присутствует в той директории, в которой вы работаете. После выбора папки у вас сбоку должна появиться панель, в которой будут отображены все файлы, находящиеся в открытой папке (отображено на рисунке ниже).

Введите в терминал следующую команду:

```
pip install -r requirements.txt
```

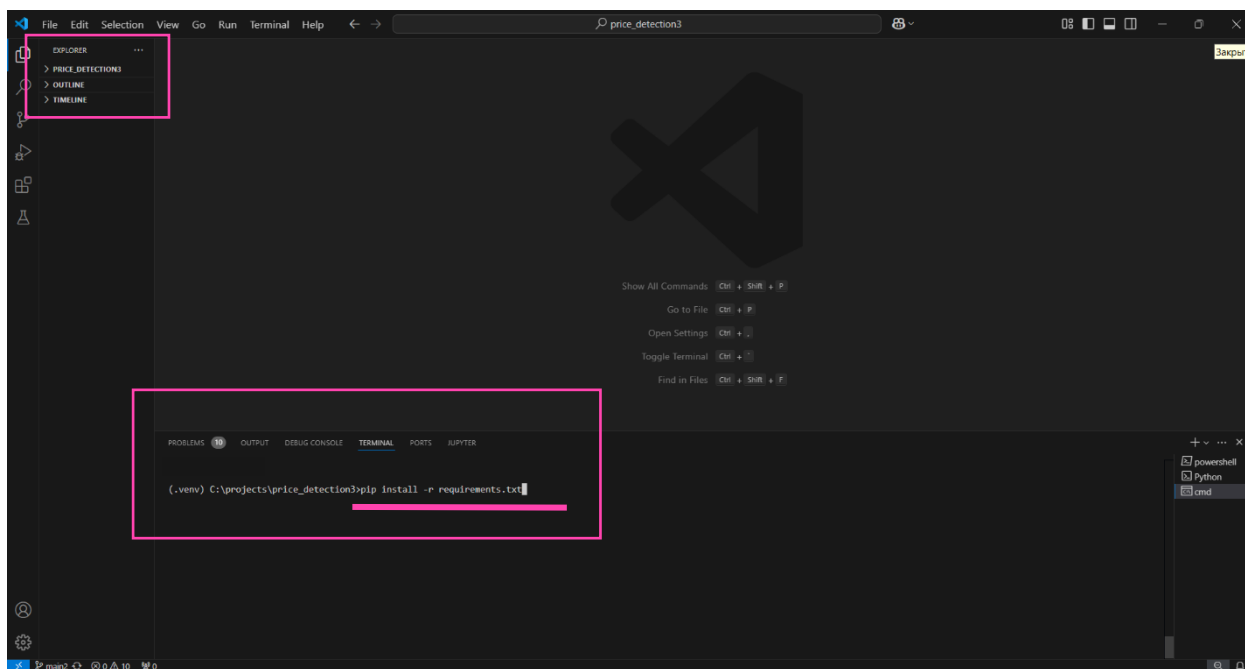


Рисунок 6. Терминал и директория в Visual Studio Code

После набора команды остаётся только нажать клавишу «Enter», чтобы началась установка всех нужных библиотек.

Теперь, когда всё установлено, можно запускать приложение. Внимание! Лучше ещё раз убедиться, что вы находитесь в нужной директории. Название директории так же прописано в терминале в самом начале строки.

Для запуска веб-приложения наберите в терминале команду и нажать кнопку «Enter»:

`streamlit run st.py`

«st.py» - название скрипта с графическим приложением.

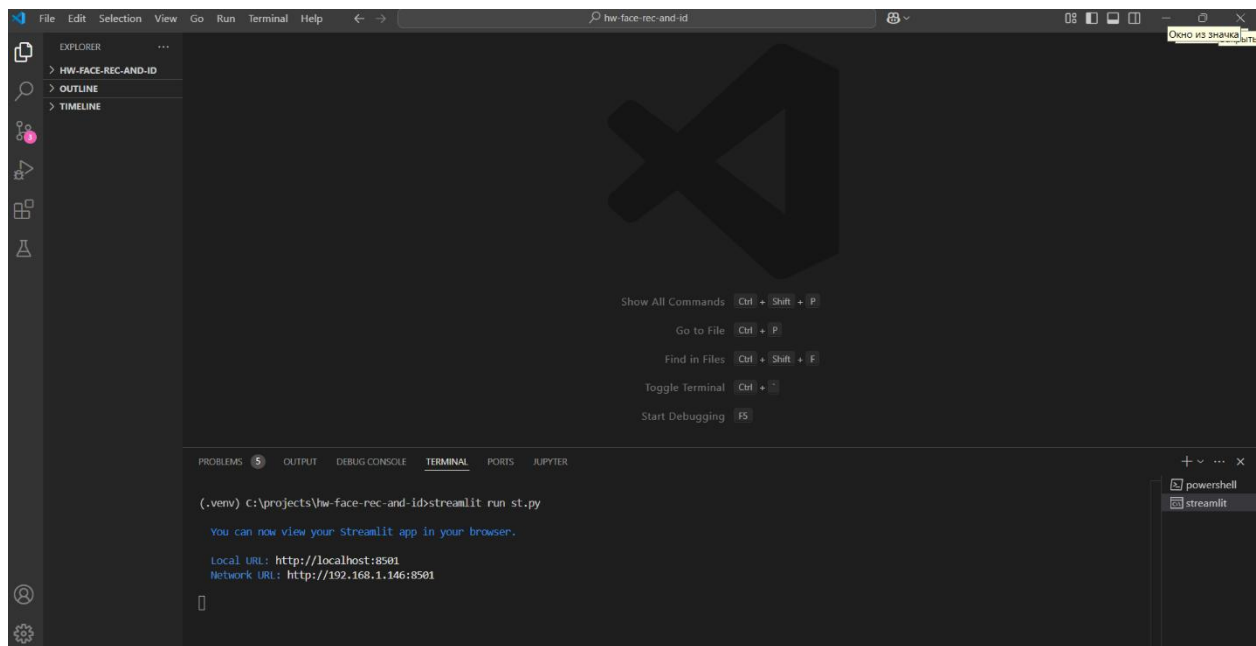


Рисунок 7. Запуск графического приложения через терминал

РАБОТА С ГРАФИЧЕСКИМ ПРИЛОЖЕНИЕМ

Если всё было сделано верно, в браузере у вас откроется сайт с ссылкой «<http://localhost:8501/>». Так же эта ссылка отображается в терминале. Загрузка приложения может занять некоторое время.

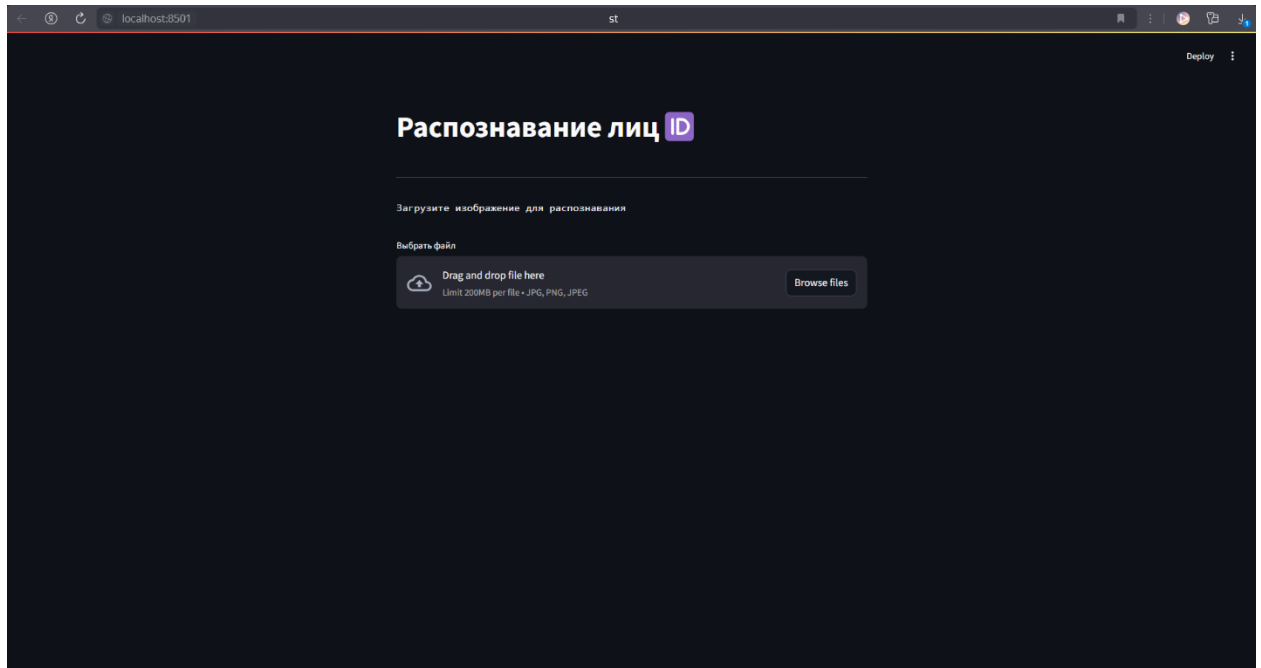


Рисунок 8. Главная страница приложения

Немного о том, как работать с графическим приложением: вам нужно выбрать файл с фотографией, на которой изображён один (!!!) человек с расширением .jpg или .png. Будьте аккуратны, загружая фотографии людей в очках, масках и прочих аксессуарах, которые закрывают лицо, так как на подобных фотографиях нейронная сеть может не обнаружить лицо.

Порядок действий для взаимодействия с API:

1. Нажмите на кнопку «Browse files».
2. Выберите файл, на котором нужно распознать ценник. Файл должен быть с расширением .jpg или .png.
3. Немного подождите: изображение передано на распознавание.
4. Результат! Вам выведется ваше изображение с обведённым найденным лицом, имя и значение совпадения, а так же наиболее близкая фотография к вашей из базы данных.

Если вы получили картинку с надписью «Error» вместо результата, то вам так же выведется проблема: было найдено либо больше 1-го лица, либо не было найдено ни одного.

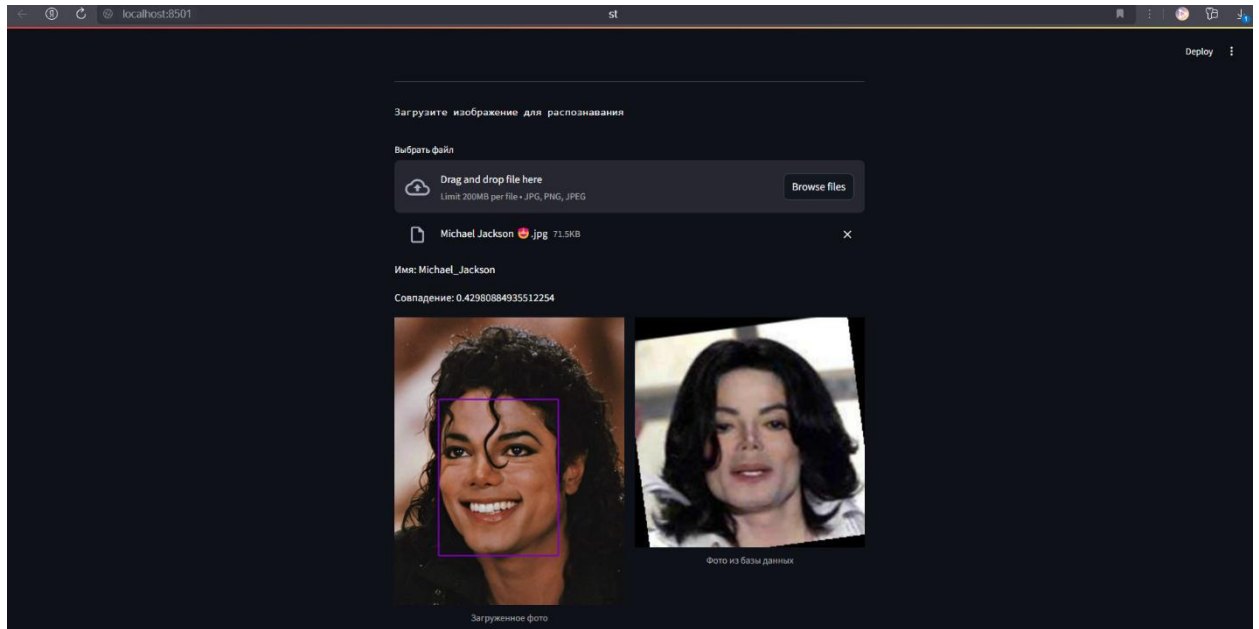


Рисунок 9. Пример вывода результата с лицом

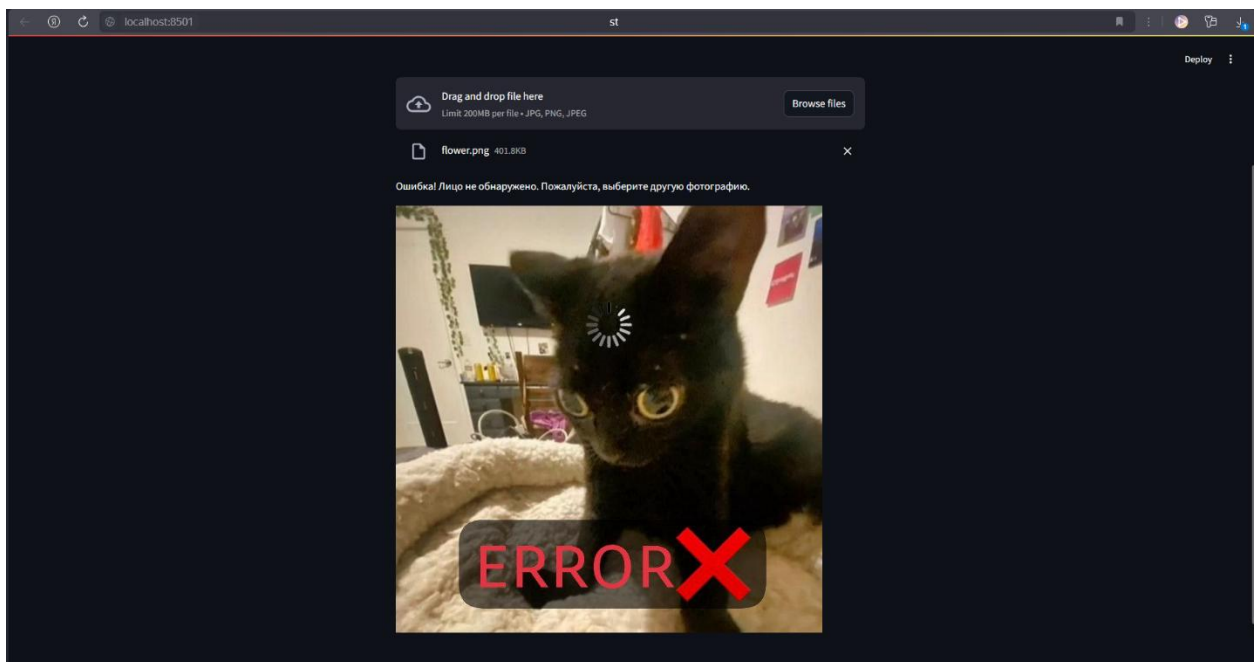


Рисунок 10. Пример вывода результата с ненайденным лицом

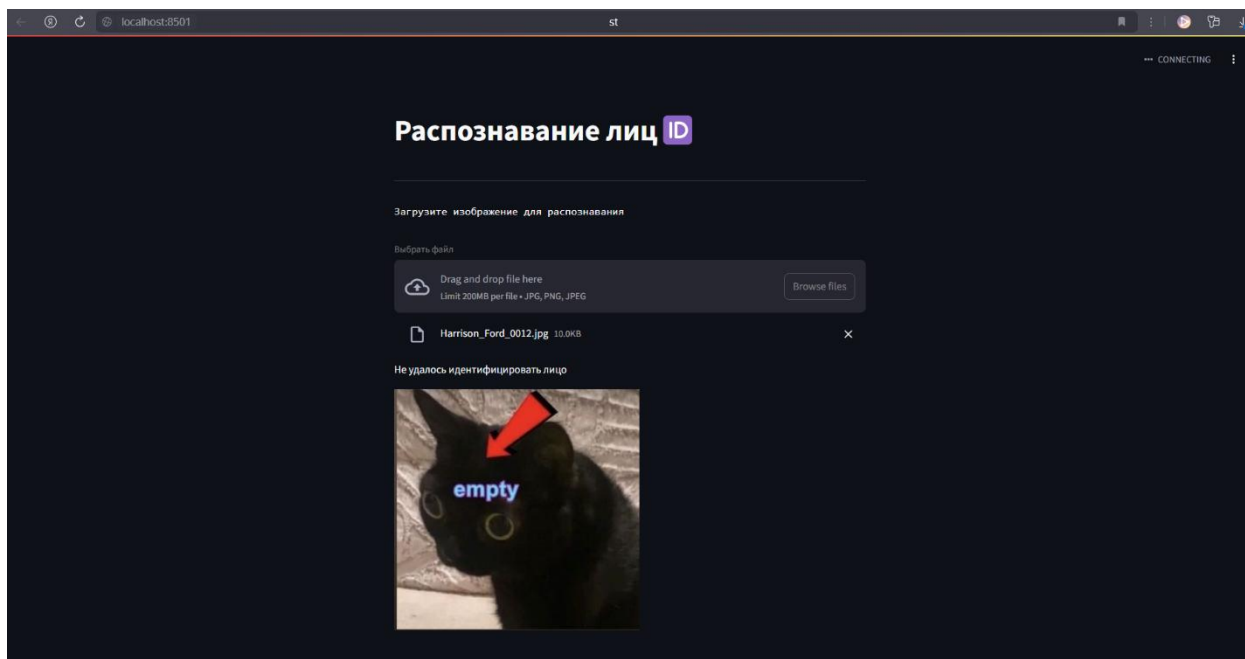


Рисунок 11. Пример вывода результата с найденным, но нераспознанным лицом