

Проект состоит из основных частей: серверной(server.py) и клиентской части (app.py). Серверная часть реализована с помощью технологии FasAPI, а клиентская – с использованием Streamlit.

## Серверная часть (server.py)

**FastAPI:** Используется для создания REST API, который обрабатывает запросы от клиентской части.

**Модели данных:** Расположены в директориях models/request\_models.py и models/response\_models.py. Они используются для валидации входных и выходных данных API.

**Обучающее ядро:** Расположено в директории trainer\_core. В модуле upload\_dataset.py реализованы функции upload\_emotion\_class, is\_image и upload\_dataset\_inframe, которые отвечают за извлечение данных с датасета для предварительной обработки:

upload\_emotion\_class – извлекает классы из датасета. В корневом каталоге датасета должны находиться каталоги в количестве не менее двух, названия которых соответствуют названию классов.

is\_image – проверка, является ли файл изображением

upload\_dataset\_inframe – извлечение данных из датасета и их преобразование в структурированный датафрейм

Модули dataset.py и extraction.py отвечают за обработку и извлечения признаков из изображений.

**Хранение данных:** Управляется через services/storage\_service.py, который отвечает за загрузку, удаление и перечисление моделей и датасетов. Датасеты загружаются и хранятся в директории datasets, а модели в директории models\_train.

**Логирование:** Настроено через core/logger.py

Ендпоинты (endpoints) расположены в service.py. В нем находятся:

/load\_dataset - отвечает за загрузку датасета на сервер;

/eda – отвечает за аналитику датасета;

/fit – отвечает за обучение модели;

/load\_model – отвечает за загрузку модели в инференс;

/list\_models – отвечает за вывод списка моделей, которые хранятся в директории models\_train;

/list\_datasets – отвечает за вывод списка уже обученных моделей, которые хранятся в директории /datasets;

/remove\_model – отвечает за удаление модели из директории models\_train;

/remove\_dataset – отвечает за удаление датасета из директории datasets;

/remove\_all\_datasets – отвечает за удаление всех датасетов из директории datasets;

/remove\_all\_models – отвечает за удаление всех моделей из директории models\_train;

/predict – отвечает за инференс;

## Клиентская часть (app.py)

**Streamlit:** Используется для создания веб-интерфейса, который взаимодействует с серверной частью;

**Функционал:** Включает загрузку датасетов, обучение моделей, инференс, удаление моделей и датасетов, а также проведение EDA;

**Загрузка датасета:** Позволяет пользователю загрузить zip-файл с датасетом;

**EDA:** Показывает статистику по выбранному датасету;

**Обучение модели:** Позволяет выбрать датасет, указать параметры модели и обучить её;

**Инференс:** Позволяет загрузить zip-файл с датасетом для предсказания;

**Удаление моделей и датасетов:** Позволяет удалить выбранные модели и датасеты;