# Мой подход к решению задачи

## Исследование и подготовка

Разбил задачу на ресерч; ноутбук; имплементация

• Провёл ресерч по методам Image Retrieval: классические, глубокие DL-архитектуры (ResNet, EfficientNet), CLIP, metric learning, DINO + всякие экзотические уже нашел имплементации но явно не на тестовое  
• Имеется опыт в metric learning: настраивал триплетные выигрывал ФИЦ, собрал ноутбук с полноценным pipeline на Flowers Recognition там шло все как по маслу  
• Jupyter Notebook отработал как по маслу: предобработка, аугментации, обучение, валидация, метрики Precision@5/mAP.

Правда там не все метрики я добавил под конец более быстрый metric learning resnet но GPU на прогон всех метрик красиво разом кончились но сиды я фиксировал

## 2. Перенос в production

• Перенос модели в API (FastAPI) и фронтенд (Streamlit).  
• Настройка Dockerfile и docker-compose: решал конфликты зависимостей   
• Работа с Faiss: хранение и загрузка \*.faiss-файлов, он-флай построение индекса для metric модели.  
+Усложнили работу скачивания тк на мак почему то не хотел multiprocess вставать

## 3. Личные обстоятельства

• В течение недели трижды возникало воспаление у моей собаки (бигль) в ухе, пришлось экстренно ехать в клинику в районе 6 вечера; из‑за этого один из этапов переноса и отладки занял примерно на час больше и + дорога поэтому в 9 загрузил вышел форс мажор(

Задача выполнена полностью: готовый API, Streamlit UI, Docker-контейнер, все коммиты задокументированы и воспроизводимы.