

Отчет по домашнему заданию №9 курса Highload Architect

Запуск:

```
git clone https://github.com/filatkinen/socialnet
```

```
cd socialnet/labs/lab09
```

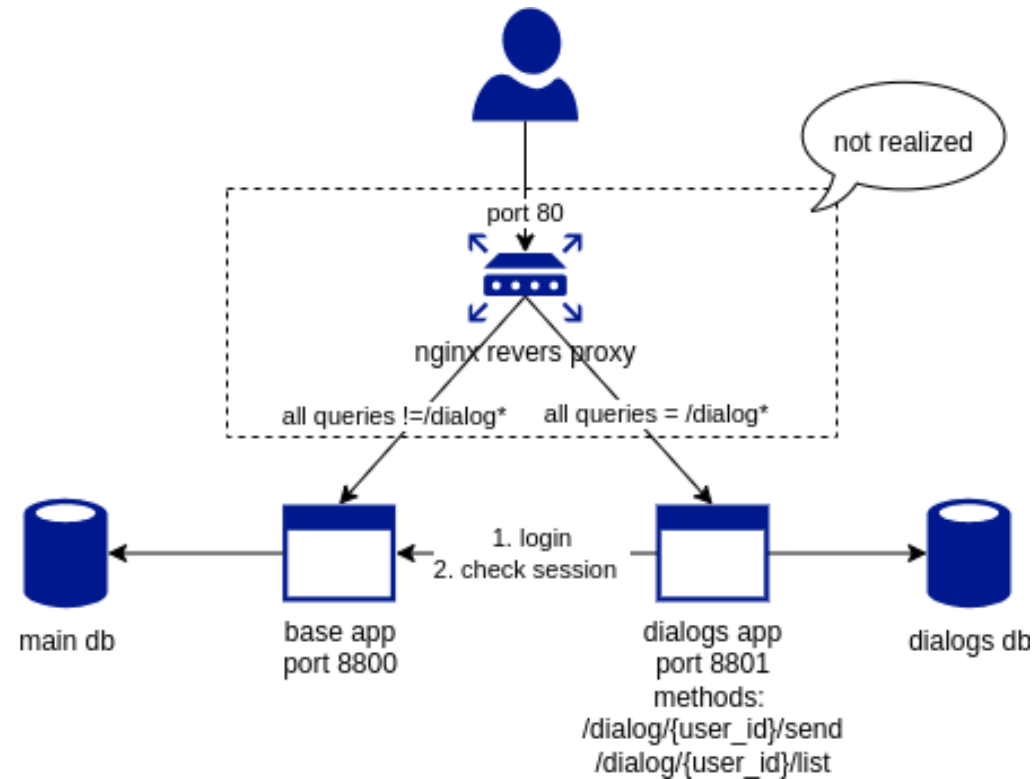
```
docker-compose build
```

```
docker-compose up -d
```

Проверка:

```
./check.sh
```

Схема разделения сервиса:



Сервис диалогов вынесен в отдельное приложение со своей БД.

Обращается к основному сервису за авторизацией либо за проверкой сессии пользователя.

По хорошему надо было бы сервис авторизации выделить в отдельный сервис, но это не в рамках данного ДЗ

На схеме изображен вариант с реверс прокси, который раскидывает запросы в зависимости от URL запроса.

Относительно пунктов требования ДЗ:

- Сквозное логирование. Его можно организовать по разному, в частности для трассировки можно добавлять на каждом этапе в заголовки дополнительную информацию. Которую можно по разному либо логировать, либо в зависимости от добавленной информации использовать ее для трейсов. Минимально условия выполнены: основной сервис логирует запросы от сервиса диалогов:

```
2023/09/28 16:48:58 Starting HTTP server at:0.0.0.0:8800
2023/09/28 19:48:58 Starting GRPC subsystem...
2023/09/28 16:49:14 127.0.0.1 [28/09/2023 16:49:14 UTC] RPC /dialog.Dialog/checkToken - - - 728.004µs grpc-go/1.55.0
```

- Требования по сохранению доступности сервиса для старых клиентов. Не знаю как это увязать с требованиями с разделением сервиса, так как мы в принципе выделяем новый сервис. Но сделать это можно так:
 - GRPC Если изменения в контракте не аффектуют старых клиентов, то они продолжат работу до обновления. Но для GRPC к имени пакета можно добавлять номер версии. Бизнес-логику нужно отделить от кода grpc. И использовать мультиплексирование -т.е. на одном порту сразу несколько сервисов и старый и новый. А так как пакеты разные(вызов функции в grpc — это имя пакета и название функции), то в принципе особой проблемы в этом нет.
 - Для REST можно использовать заголовок в header с версией API, нужно чтобы этот заголовок отправлялся клиентом и на основании этой информации обрабатывать запросы от клиента(но говорят что не советуют так делать, так как заголовки могут кэшироваться) ну либо вынести обращение к API на отдельный поддомен(v1.api.ch.ru)/путь в адресе(ch.ru/api/v1).