**Исходный текст и запрос к ИИ**

«И так, начнем разбираться с архитектурой нашего проекта.

В главное директории существует множество разных файлов, но нас интересуют больше всего файлы Python: app.py и loader.py. Это два главных файла, в loader.py у нас будет все импортироваться и будут создаваться основные переменные, с которыми нам предстоит работать. А в файле app.py у нас будет запускаться работа бота.

Также у нас есть Dockerfile, в котором будет прописывается все для докера.

В файле requirements.txt, в котором будут прописывается зависимости, необходимые для запуска бота. Туда входят дополнительные модули, которые нужно будет установить, ну и естественно, нашу библиотеку Aiogram.

Также нужно создать файл .env, чтобы подгружать из него переменные окружения, такие как TOKEN, admin\_id, которые потом будут загружаться в конфигурацию.

Для хранения разных данных, нужно создать несколько папок. Одной из них будет папка data, в которой будут хранится константы, используемые ботом. В файле config.py, в котором будут прогружаться все пароли из переменных окружения, BOT\_TOKEN, ip адрес, который мы должны будем прописать, тут будут параметры для redis и т.д. Также во всех папках будет файл \_\_init\_\_.py, который может быть пустым или в нем будет хранится какая-нибудь информация. В него будут импортироваться главные вещи. Поэтому этот файл нужен для импортов, потому что мы будем из разных файлов импортировать функции, переменные необходимые для работы.

В папке filters будут хранится наш фильтры для хендлеров в боте. Про то и другое поговорим немного позже, но вкратце, фильтры будут фильтровать те апдейты, которые мы будем получать и определять, куда им идти и в какую функцию попадать. Фильтр говорит обработчику в какую функцию и в какой хендлер попасть и получается фильтр для апдейтов и сообщений.

Следующая папка называется handlers, в которой как раз таки будут находится наши хендлеры. В ней есть подпапки channels, errors, groups, users. Обычно, один апдейт попадает в одну функцию, где с ним что-то делают. Поэтому было бы не плохо разделить хендлеры личной переписки, групп, каналов, чтобы они были разделены и были видны, а не смешивались в кучу.

Также у нас есть папка keyboards, в которой находятся подпапки default и inline. Это наши обычные и инлайн кнопки. И, опять-таки, тут есть файлы \_\_init\_\_.py, в которые мы будем импортировать переменные и функции.

Дальше есть папка middlewarres, в которой будут находится наши мидлвари. По своей сути, мидлварь — это программа, которая лежит между операционной системой и приложением. По аналогии, мидлвари это модули, которые будут находится между доставщиком апдейтов и хендлерами. То есть с помощью мидлварей мы сможем влиять на ту информацию, которая идет в хендлер.

Также у нас есть папочка states, в которой будут прописываться состояния. Машина состояний нужна для того, чтобы хранить и обмениваться информацией между хендлерами.

И последняя папка utils. В ней будут вспомогательные модули для работы с базами данных и прочие скрипты, хранящиеся в папке misc. Также в utils есть подпапки db\_api для работы с базами данных, redis, в которой будут хранится какие-то константы для редиса, и файл notify\_admins.py, для рассылки админам бота сообщения о том, что бот запущен.»

*Построй дерево по описанной архитектуре бота и распиши значение каждой папки.*

**final\_bot – папка с финальной версией бота**

├── **bot.py** - файл, в котором запускается работа бота

├── \_\_init\_\_.py

├── **data** - папка, в которой хранятся константы, используемые ботом

│ ├── consts\_and\_vars.py – файл, в котором хранятся все «легкие» постоянные (словари, листы и тд) и функции для получения значений из БД

│ ├── bot\_states.py – файл с состояниями для FSM

│ ├── token\_schoolhelper\_bot.py – файл с токеном для доступа к боту

│ └── \_\_init\_\_.py - файл инициализации папки data

├── **handlers** - папка, в которой хранятся хендлеры для обработки апдейтов и сообщений от пользователей

│ ├── common.py – обычные хэндлеры

│ ├── schedule.py – хэндлеры расписания

│ ├── menu.py – хэндлеры меню

│ ├── scores.py – хэндлеры перевода баллов егэ

│ ├── jokes.py – хэндлеры анекдотов

│ └── \_\_init\_\_.py - файл инициализации папки handlers

├── **keyboards** - папка, в которой хранятся кнопки для бота

│ ├── reply\_kb.py - файл, в котором хранятся обычные кнопки

│ ├── inline\_kb.py - файл, в котором хранятся инлайн кнопки

└── └── \_\_init\_\_.py - файл инициализации папки keyboards