

Тестовое задание для backend - разработчика

Цель:

Разработать модуль для связи и обработки сообщений между Телеграм (через telegram-api) и чат-ботом OpenAI через их API. В качестве альтернативы можно использовать другую языковую модель с которой есть возможность работать.

Готовая работа должна включать себя модуль следующей функциональности:

- 1) Сам модуль, отправляющий и получающий сообщения между тг-ботом и OpenAI API.
- 2) Механизм модуля, который может определить, что пользователь выразил желание поговорить не с ботом, а с оператором.
- 3) В случае, если модуль определяет, что пользователь хочет поговорить с человеком, необходимо передать в тг-бота соответствующее сообщение: «переключаю на человека», после чего подключить к диалогу тг-бота любого пользователя.

Задачи:

1. Настройка Telegram бота:

- Используйте официальный HTTP API Telegram для приема и отправки сообщений.
- Настройте webhook для передачи входящих обновлений от Telegram на ваш сервер.

2. Разработка логики бота:

- Используйте OpenAI для генерации ответов на запросы пользователей.
- Реализуйте механизм, по которому бот будет определять, желает ли пользователь разговаривать с живым человеком.

3. Работа с базой данных и кратковременной памятью:

- Используйте PostgreSQL через SQLAlchemy для хранения данных о диалогах и пользователях.
- Используйте Redis для хранения кратковременной памяти о статусах диалогов, например, кто должен отвечать: бот или оператор.

4. Разработка API:

- Используйте FastAPI для создания эндпоинтов:
- Принимать обновления от Telegram.
- Отправлять запросы в OpenAI и возвращать ответы пользователю.
- Передавать управление оператору, если бот не может обработать запрос.

5. Контейнеризация:

- Используйте Docker для сборки приложения.
- Настройте Docker Compose для оркестрации контейнеров, включая веб-сервер, базу данных, Redis и любые другие необходимые сервисы.

Технические детали:

- Некоторые моменты, требующие сложной и долгой реализации, могут быть просто описаны в комментариях в коде, указывая, как бы вы реализовали их на практике.
- Подходите творчески к решению задачи, особенно к механизмам определения возможности бота ответить на запрос и передачи диалога оператору.
- Для реализации использовать Python от 3.10

Ожидаемые результаты:

- Код проекта должен быть доступен на GitHub с README файлом, который описывает, как настроить и запустить проект в Docker.

- README должен также включать описание основных функций и примеры использования API.

Мы ожидаем, что вы продемонстрируете свои навыки работы с API, асинхронным программированием, управлением базами данных, контейнеризацией приложений и применением систем хранения кратковременных данных для управления состоянием.