Task

You are tasked with building a **FastAPI** service that exposes endpoints for a classification chatbot. The chatbot should leverage the **OpenAI API** to generate results. Additionally, the service must store history in PostgreSQL and allow retrieval of this history. The dataset with classification examples, to make sure there is no overfitting to the data, we have a separate dataset which we will use to test the results of your **API**. This is the test data you will submit with the APIs output.

The mapping of values is the following: Politics = 0; Sport = 1; Technology = 2; Entertainment =3; Business = 4.

The goal is to test:

1. **Clean coding** and project structure.
2. **API design** using FastAPI.
3. **OpenAI API** usage.
4. **SQL database** integration and schema design.
5. **Best practices** around configuration management, error handling, and security (API keys, etc.).

Requirements

- You must use Python
- You must use OpenAI's API
- You must document how to run your code
- Expose at least the following endpoints:
  - `POST /classify`:
    - Accepts a user message.
      - `message` is the user's query.
    - Response
      - `response` classification result.

Additional notes

- You can use the **openAI API** to generate embeddings and classify using the embeddings if you please .
- You can fine-tune a model if you please

Extra points

- **Testing**: Add unit tests or integration tests with `pytest`.
- **Docker**: Provide a `Dockerfile` to containerize the application.
- **Asynchronous** code with FastAPI (`async def`) where appropriate.

Delivery

- **Github** public repository link
    - code and how to run it
    - *test.csv* following the structure of the *train.csv* but with the values predicted by your **API**.