Adaptive Multi-Agent Chatbot System
Execution Guide

Install Libraries

First, install the packages and libraries. The project dependencies can be found in `.\docs\requirements.tsx` of the project directory, and are also listed below:
- · fastapi
- · uvicorn
- · langchain
- · wikipedia
- · ollama
- · langchain_ollama
- · langchain-community
- · chromadb
- · sentence-transformers

Install *Ollama*. Next, run the following command in the project's terminal to install the necessary dependencies:

```
pip install -r .\docs\requirements.tsx
```

Or run the command in the terminal:

```
pip install fastapi uvicorn langchain ollama wikipedia langchain_ollama
langchain-community chromadb sentence-transformers
```

If the IDE still cannot recognize the libraries and packages, install them independently, using the IDE suggestions or refresh your Python Interpreter.

Run and query the Multi-Agent Chatbot System

If the model has not yet been downloaded
1. Pull the LLMs you will be using from Ollama
   [In your computer's CLI]
      - · Run the command `ollama` to confirm it is installed.
      - · Run the command `ollama pull mistral` to download the model `mistral` from the internet. It may take some time and make sure you have the space on your computer for it. `mistral` is a smaller LLM and will be used in the project.

- Confirm that the model is downloaded with the command `ollama list` and view the model.
- Run *Ollama* by clicking its application icon in your local environment.

If the model has been downloaded already

1. Run *Ollama* by clicking its application icon in your local environment.
2. Run *FastAPI*
   [In the IDE's CLI]
   - Run `uvicorn api.main:app --reload`
   - Open a browser and enter the link `http://127.0.0.1:8000/docs` to access *SwaggerUI* resource.
3. Query the multi-agent chatbot
   - Navigate to the `POST /chat endpoint` on the webpage.
   - Click the button "Try it out".
   - Input a query into the "input_text" text box. An example input can be "Where is Montreal?"
   - Click the button "Execute" and wait for the relevant agent to generate its response. This may take some time.
   - Once the page has finished loading, scroll down to the "Response" section of the page and view the "Server Response".
   - The response generated by the agent can be viewed in the block "Response body" in json format, along with other details including the curl, request URL and response headers. An example response is below for the query posed,

```
{
   "response": " Montreal is located in Canada, specifically in the province
of Quebec. It is centrally located on the Island of Montreal and several
smaller peripheral islands, with the largest being Île Bizard. In terms of
geographical proximity, it is approximately 196 kilometers (122 miles) east of
Ottawa, the national capital, and 258 kilometers (160 miles) southwest of
Quebec City, the provincial capital."
}
```

4. View chat history
   - Navigate to the `GET /history endpoint`
   - Click the button "Try it out"
   - Click the button "Execute"
   - Scroll down to the "Server Response" section to view the entire chat history with the chatbots in json format.
5. When finished, close all running processes
   - Stop running *FastAPI* in your IDE terminal.

- Stop running *Ollama*.