MCP Workshop: Exercise 1

File: src/main.py

Clone the repo from https://github.com/DiegoLigtenberg/workshop-langgraph-mcp

This repo contains contains an agent in src/main.py. Please get this file running using poetry, docker or podman (see README.MD). The agent in main.py hosts a web-based chatinterface that connects to local and external mcp servers using stdio. This means that the code from the mcp servers will be locally extracted from a filepath (src/local_mcp_servers), or by using a uv download (in the case for word-mcp).

The MCP server that will be downloaded in this workshop is safe and can be found on this github page: https://github.com/GongRzhe/Office-Word-MCP-Server

These exercises provide a way to familiarize yourself with connecting chat interfaces with (external) MCP servers. Please start with **exercise 1**. Here you can play around with an existing chat-bot hosting a Vibify mcp server.

From exercise 2 onwards, you can run your own local version of this connected to other MCP servers. Accesses to this FastAPI web interface is done at http://localhost:8000/chat or https://127.0.0.1:8000/chat.

Exercise 1: This introduction exercise shows an MCP server connected to the Vibify website shown in the presentation. Go to mcp-workshop- server.up.railway.app. To get familiar with the application, ask the chatbot to 1) Name a song in the database, 2) What song is streamed the most, 3)
Can you increase the streamcount of this song by 10. Why is question 3 not possible?

____

Exercise 2.a: Go to src/main.py. Setup the environment using poetry/docker/podman. Run the file and view the tool calls are loaded from the Word-Mcp server. Check the console output when the server starts and scan the list of all the tools available from the office-word-mcp-server.
How many are loaded, and do you think the LLM has enough con- text for when to use what tool?

____

Exercise 2.b: *How could we guide the LLM to have a better understanding of when to use what tool? Hint: system prompt*

____

Exercise 3: Try to make a query that combines the local weather server and math server (src/local_mcp_servers) and creates a word document with the answer of your query. Make sure to give the word document a name.docx. For example: "What's the weather in Paris and what's 15 * 7? Create a word document called results.docx with both answers."
What tools did the program use?

____

Exercise 4: Compare the server configuration for local servers vs external packages.
Where is the uv package installed? What does this imply for security of downloaded MCP servers?

____

Bonus: *Can you come up with a way to create a business proposal word document (with in-text references and a reference list)?*

____

NOTE: you may need to remove the truncate_messages_safely() function as this can require quite some memory