{EPITECH}

HEPHAESTUS_

< LLMS, LIFE, FORGING STUFF, THE UNIVERSE AND EVERYTHING ELSE />

HEPHAESTUS

> In Greek mythology, Hephaestus was the son of Hera, either on her own or by her husband Zeus. He was cast off Mount Olympus by his mother Hera because of his lameness, the result of a congenital impairment; or in another account, by Zeus for protecting Hera from his advances (in which case his lameness would have been the result of his fall rather than the reason for it).
> As a smithing god, Hephaestus created all the weapons of the gods in Olympus. He served as the blacksmith of the gods, and was worshipped in the manufacturing and industrial centres of Greece, particularly Athens.

Source

And much like Hephaestus, you'll be forging tools. This time for LLMs, giving them a great power: access to live data.

Educational Context

This project runs from January 8th to January 16th, 2026.
You will need to complete this project in a group of 3 to 5 students.

The Hephaestus project aims to introduce you to the design of an intelligent conversational agent capable of executing actions using external tools via a Model Context Protocol (MCP) or a similar protocol.

Introduction

This project will require you to combine skills in AI, web development, and software orchestration, while adhering to best practices in software engineering, documentation, and technical presentation.

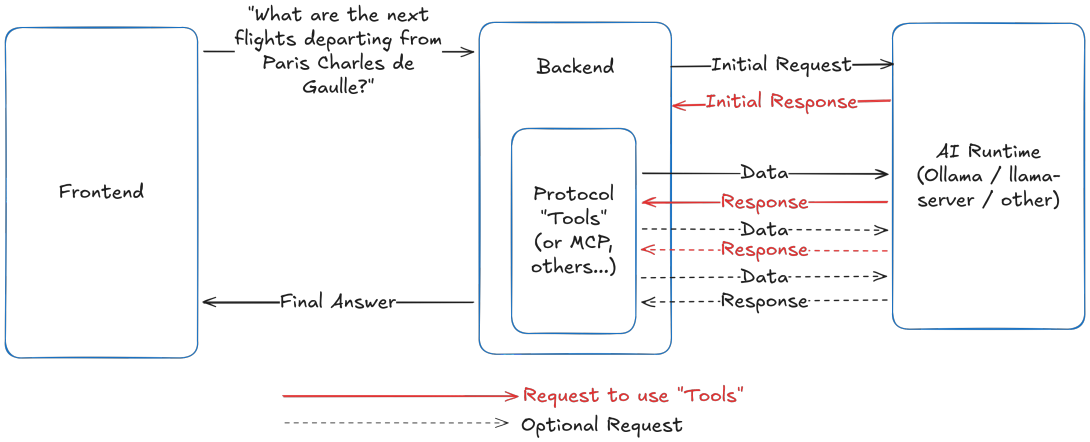
You will be tasked with designing a functional prototype that combines natural language interaction, server-side orchestration of automated actions, and integration into a web interface. **You can pick the theme.** We'll be using airports as example, but feel free to pick whatever you're the most interested in.

We'll need a frontend to interact with as a user. **The frontend must communicate with a backend, which will itself communicate with whatever AI Runtime you end up using.** The reference project was done with Ollama, but if you have a better one go for it!

The backend will need to provide tools that the LLM can use to get additional, live data. Using APIs is nice... But you need to implement every single one of them. What if the user is actually in Tokyo? Seoul? Madrid? Rennes?

How about some live web scraping using something like Selenium (or better, Zendriver) instead?

Example of architecture (simplified, no response streaming):



Suggestions

You are free to choose your tech stack, provided it is coherent and well-documented.

However, the following tools are recommended:
- Backend: Python (FastAPI, Flask)
- Frontend: TypeScript (React / Angular / Vue.js / Svelte)
- Local AI Runtime: Ollama
- LLM: DeepSeek R1 / GPT-OSS / Llama 3.x / Mistral / Qwen3

The use of a local, open-source model is mandatory. No credits or paid API keys will be provided.

The project was fully tested with Qwen3 1.7B in a 40k context.
Using more complex models is possible for testing purposes, but will not be considered in the evaluation criteria.

Naturally, a larger number of parameters and a larger context significantly improve the quality of the results and prevent hallucinations. Consider using at the very least 32k context size if you send raw webpages to the LLM directly.

Presentation

Each group will have 20 minutes to present their work, followed by a 5-10 minute Q&A session.
- A functional demonstration of the website and chatbot;
- A technical explanation of the project architecture (frontend, backend, AI, tools, data flow);
- A critical analysis of the choices made, the difficulties encountered, and the solutions implemented;
- A discussion of potential future developments for the project or reuse of the approach.

All group members must actively participate in the presentation. Any student who does not participate will fail the project.

v1