TOC 2025 Final Project

Project Description — Intelligent Agents with LLMs

In this project, you will design and implement AI agents capable of interacting with a Large Language Model (LLM) through API-based communication. Your agent should be able to understand user tasks, generate structured prompts, interpret LLM outputs, and execute appropriate follow-up actions. At the end of the project, you will deliver a working prototype, a live demo, and a short report describing your system design, development process, implementation challenges, and potential real-world applications. All code must be uploaded to your GitHub repository.

Requirements

- **LLM API Usage**
    - We will provide each team with a unique API key for accessing our LLM services.
    - You “must” integrate our LLM API into your agent’s workflow.
    - API usage will be logged. Please use the service responsibly and avoid security risks or abusive behavior.
- **System Logic Diagram**
    - You must include a State Machine Diagram or a DAG (Directed Acyclic Graph) illustrating your system logic and workflow.
- **Team Size**
    - Each team may consist of **1 ~ 3 members**.
- **Code Submission**
    - All project code must be uploaded to GitHub.
    - The TA will clone your repository during your demo.
- **Programming Language**
    - No restrictions; choose any language you prefer.
- **Project Demo**
    - Demo sessions will be scheduled at 17th week (暫定).
    - The TA will provide a sign-up sheet for some demo slots then.
    - TA reserves the right to make changes.

Grading (0–100 points)

Essential Level

Failing any of the following results in **0 points**:

- No LLM usage in your system.
- No working demo code available on GitHub.
- No State Machine Diagram or DAG.

Encouragement Level (50 points)

- Must pass all essential-level requirements.
- Implementing the TA’s toy example or existing community example

Advanced Level (50–100 points)

- Complete all essential requirements.
- Implement features beyond the TA’s example.
- Additional scoring:
    - **Innovation & Architecture (0–30 points)**
        - Creativity, system design, and overall implementation quality
    - **Presentation & Demo (0–10 points)**
        - Clarity, communication, and demonstration quality.
    - **Code Quality (0–10 points)**
        - Code cleanliness, readability, documentation, and structure.

Note: All team members receive the same score. Please coordinate fairly.

You are allowed to do this project individually.

Toy Example Workflow

QA Agent — an interactive assistant that calls the LLM and performs automatic web search.

1. The user sends a query to the agent.
2. The agent performs a web search and collects relevant text.
3. The agent sends the collected context to our Ollama LLM API (using your assigned API key) for summarization or answer extraction.
4. The LLM returns a natural-language response, and your agent displays the result (or visualizes it using tools such as matplotlib).

Example Project Ideas

Here are some ideas you may develop or extend:

- **RAG (Retrieval-Augmented Generation) Agent**

  Combine external data sources (documents, databases, websites) with the LLM for more accurate and context-aware answers.

- **Study Helper**

  Summarize articles, generate notes, or explain technical concepts for students.

- **Personal Diary Agent** (e.g., integrated with LINE chatbot)

  Automatically record daily notes and support queries such as：

    - “我之前什麼時候有去吃丹丹遇到莊坤達？”

    - “我計算理論的老師叫什麼名字？”

      The agent retrieves entries from the diary database and provides summaries or answers.