“AI engineering with LLMs isn’t just prompt-writing—it’s system-building. It’s about designing a thoughtful, end-to-end architecture that aligns the model’s strengths with the core goal. From input handling to retrieval, logic, and output, every part is planned to create a reliable, useful, and scalable system that just—works!”

Knowledge Pre-requisites:

- Python, LLMs, Data management

Task Details:

- We’ll be building a chatbot that helps people learn any language.
- The bot asks what language the user wants to learn, the language they know, their current level of the learning language, then sets a scene based on user selection, and chats with the user in the learning language.
- Over the course of the conversation, the bot helps the user on their mistakes and also maintains a separate list of the mistakes they made. A local database like SQLite may be used for this.
- Finally it gives an overview of the mistakes the user made and a review of what areas they need to focus on to improve.

Task Rules:

1. OpenAI models are recommended to be used.
2. AI Wrapper libraries like LangChain are recommended to be used.
3. Submission can be in the form of a Jupyter Notebook or a Python Script.
4. Personal API keys can be used for testing but please remove them before submission.
5. A brief documentation (1-2 pages) is to be submitted alongside the project. The architecture of the system is to be properly drawn and explained there.
6. A screen recorded video explaining the code, the system, the architecture and the logic behind everything is to be provided. We expect the webcam to be on as well while recording the video. Please refer to this video for guidance on setting it up: Link

Final Deliverables:

Create a **.zip** with the following items and upload in the submission form.

1. Project code as Jupyter Notebook or Python Script.
2. A brief documentation on the architecture and the overall system.
3. Short screen recorded video (with webcam turned on) explaining everything.