

Multiple PDF Query Chatbot

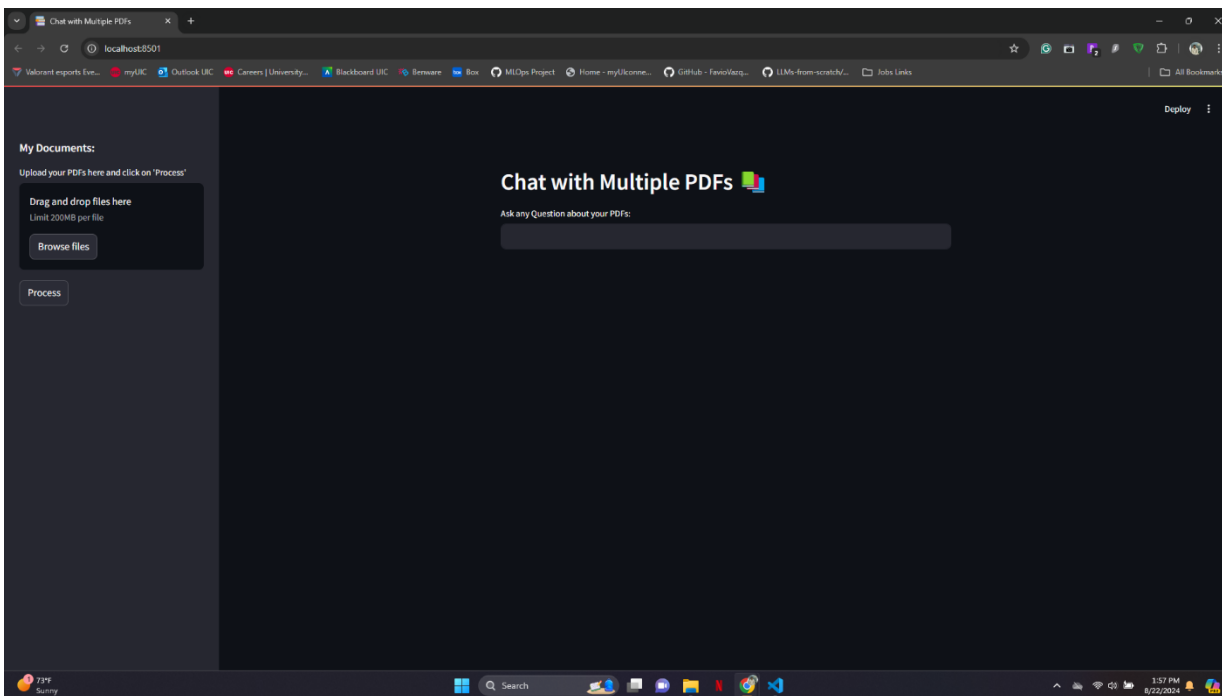
Language: Python

Libraries Used: Streamlit, Langchain, OpenAI API.

Outcome: Developed a RAG (Retrieval Augmented Generation) based web application using Streamlit, which powers a sophisticated chatbot capable of answering questions directly from uploaded PDF documents. Leveraged LangChain to extract and convert textual data from PDFs into high-dimensional embeddings, subsequently stored in a vector database 'FAISS' for efficient retrieval. Integrated OpenAI's GPT API for generating contextually relevant answers by querying the stored embeddings, ensuring accurate and insightful responses. The system is designed to handle complex queries and provides a seamless user experience, demonstrating advanced NLP capabilities and scalable architecture.

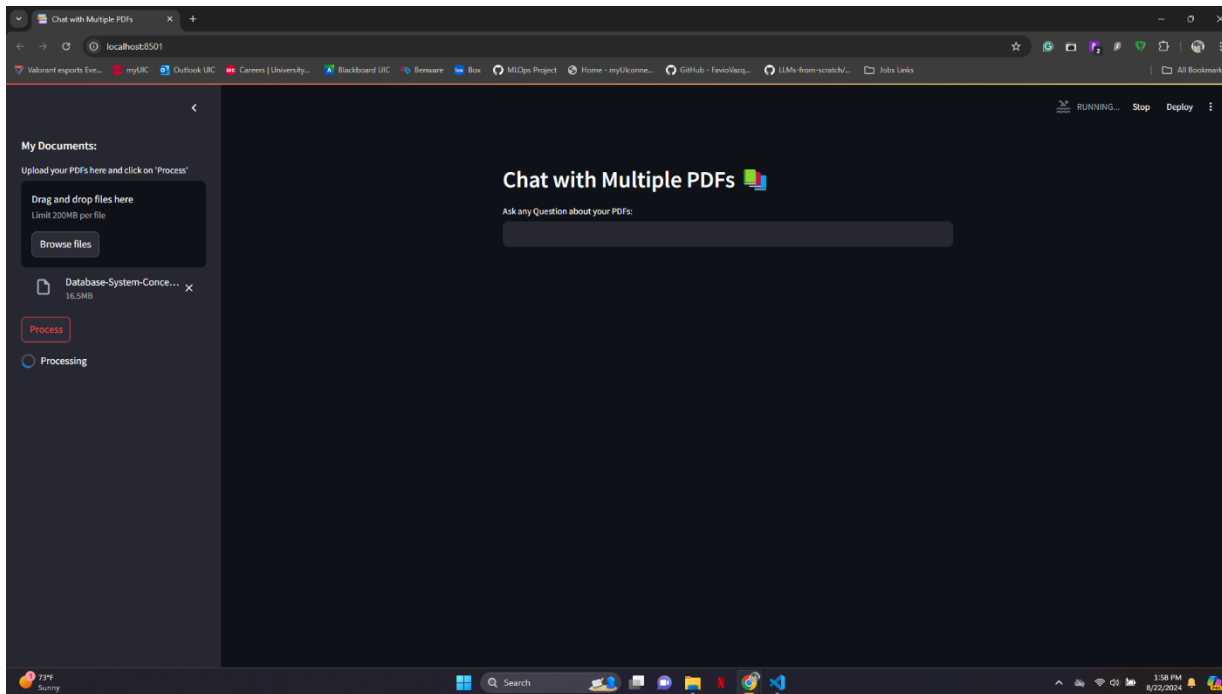
Demo:

Step 1



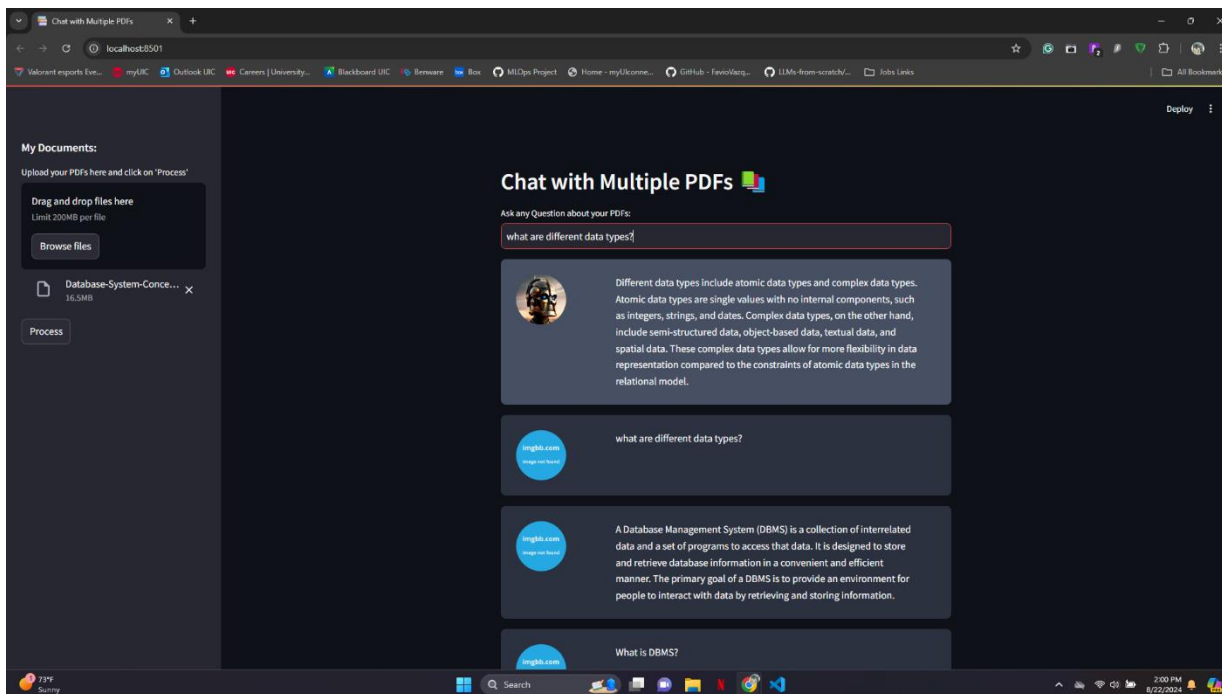
- Upload the PDFs by using the 'Browse files' option and click 'Process'.

Step 2



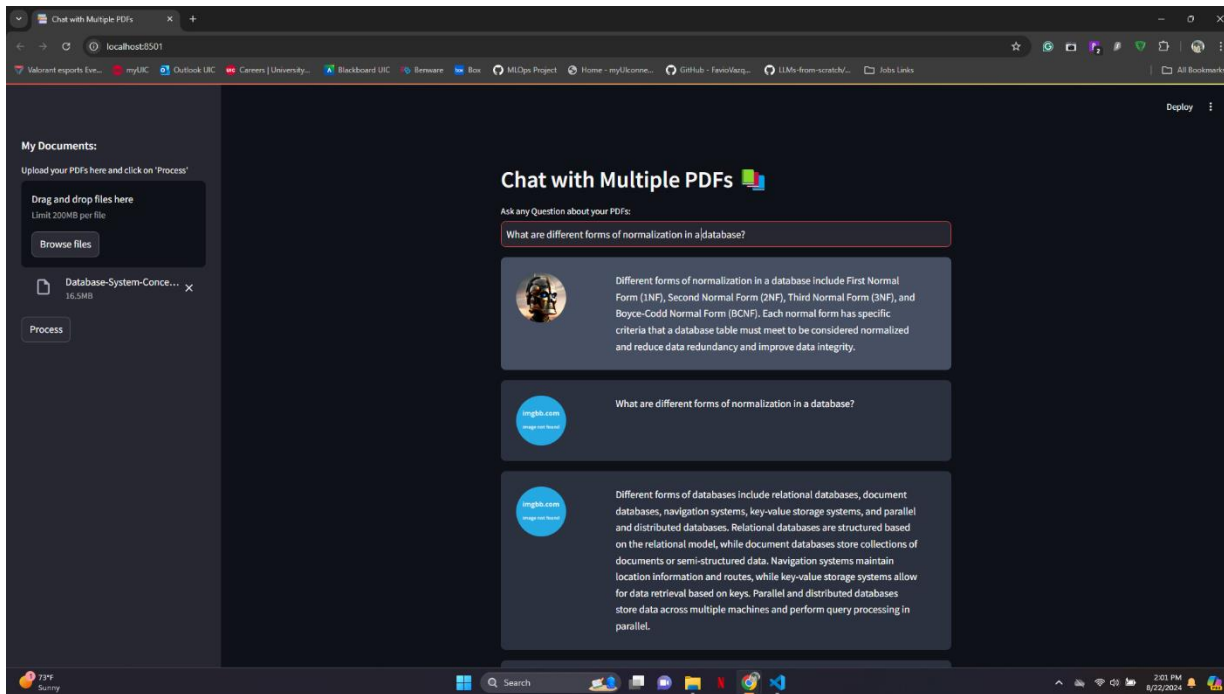
- This step will process the uploaded PDFs into text embeddings and store them into Vector Database FAISS.

Step 3



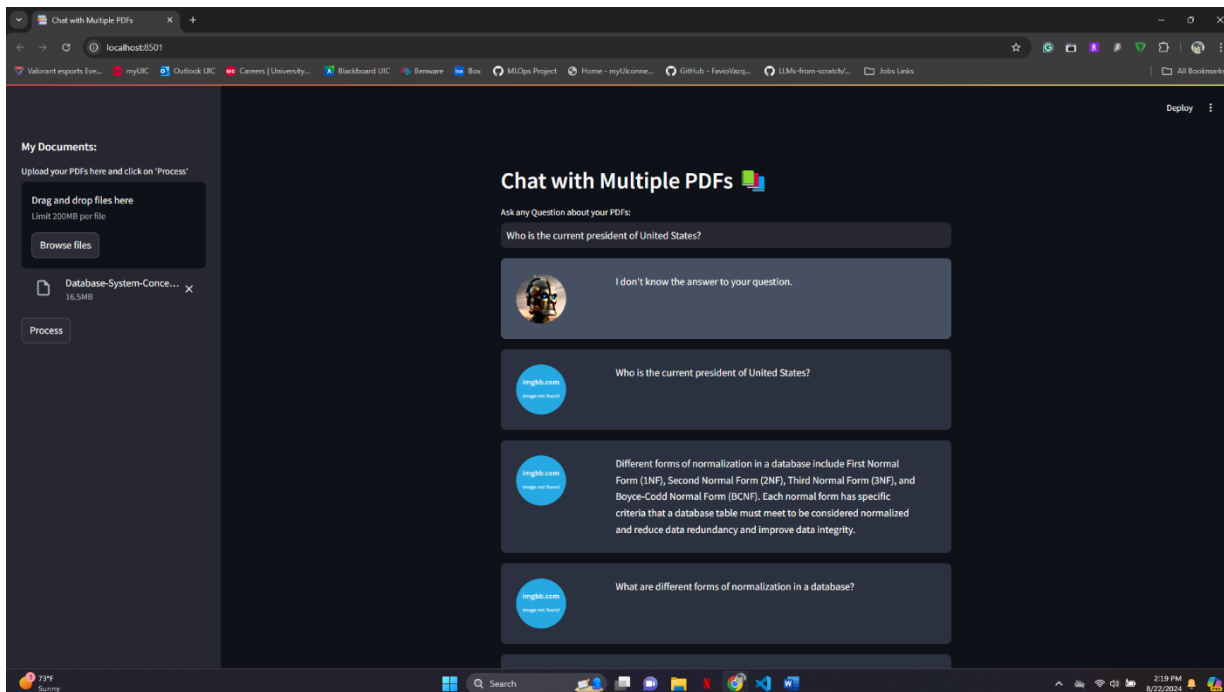
- Once the processing is complete, ask any question related to the PDFs and the chatbot will answer accurately.
- It also remembers previous questions so providing context again is not necessary.

Step 4



- A conversational chain is created as Question and Answer to review the answers and go back to previous questions just like CHATGPT.

Step 5



- For questions that do not relate to uploaded PDFs, it returns no answer which ensures the conversation to remain in context of PDFs.

