DAY – 16

Project: Product Prompt Assistant

On Day 16, the focus was on all about shifting gears from theoretical concepts to building a robust final product. We dedicated the entire day to designing the core architecture for our AI-Powered Research Assistant (the Capstone Project). The goal was to consolidate our learning into one single, scalable AI solution.

We decided on a modular structure that leverages the strengths of tools studied previously:

- **Backend Core:** Python/Flask housing a LangChain Agent.
- **Data Source:** Multiple private documents processed via RAG (Retrieval-Augmented Generation) to ensure accurate, grounded answers.
- **Functionality:** Function Calling to execute external tools (like live data or calculations).
- **Frontend:** Simple HTML/CSS/JS interface for user interaction (building on previous web app experience).

We began setting up the RAG pipeline:

- Identified sample documents for the knowledge base.
- Wrote the initial script for document loading, chunking (with overlapping chunks) , and generating embeddings.
- Initialized a local vector store (like Chroma DB or FAISS) to store the document embeddings, ready for context retrieval.

CONCLUSION

- Defined a concrete project plan, ensuring the architecture incorporates RAG, agents, and external functions for the final evaluation.
- Successfully implemented initial indexing by splitting documents into chunks and vectorizing them.