Project: - **Ask Question from PDF**

* **Model Used:** [llama3.2:1b](https://ollama.com/library/llama3.2)
* **Platform Used:** [Ollama](https://ollama.com)
* **Required Libraries:**

langchain, langchain\_community

faiss-cpu

PyMuPDF

dotenv

tiktoken

ollama-embeddings

langchain\_core

**Sprint Plan**

* **Day 1 [19/11/2024]: Document Loading and Processing**

Tasks:

* Implement PDF ingestion functionality.
* Traverse the directory to locate PDF files.
* Load PDF files using PyMuPDF.
* Split documents into chunks with overlap for better context continuity.
* Tokenize and evaluate chunk sizes using tiktoken.
* **Day 2 [20/11/2024]: Embeddings and Vector Store**

Tasks:

* Set up the Ollama embedding model to generate embeddings for text chunks.
* Create a FAISS vector store to store embeddings.
* Implement functionality to add documents to the vector store and test similarity retrieval.
* **Day 3 [21/11/2024]: Retrieval and Question Answering**

Tasks:

* Configure the retriever.
* Set up the LLM for question answering using ChatOllama.
* Design the RAG pipeline:
* Retrieval of relevant chunks.
* LLM-based answer generation using retrieved context.
* Test the pipeline with sample questions.

**Dependencies**

* Ollama server for embeddings and LLM inference.
* FAISS for vector similarity search.
* Pre-installed libraries listed in the requirements section.

**Foundational Beliefs**

* All PDFs contain textual data.
* LLM server and embedding service are correctly set up and accessible.
* The system will be used on a local machine or environment with necessary compute resources.

Name: - **Sawan Kumar**