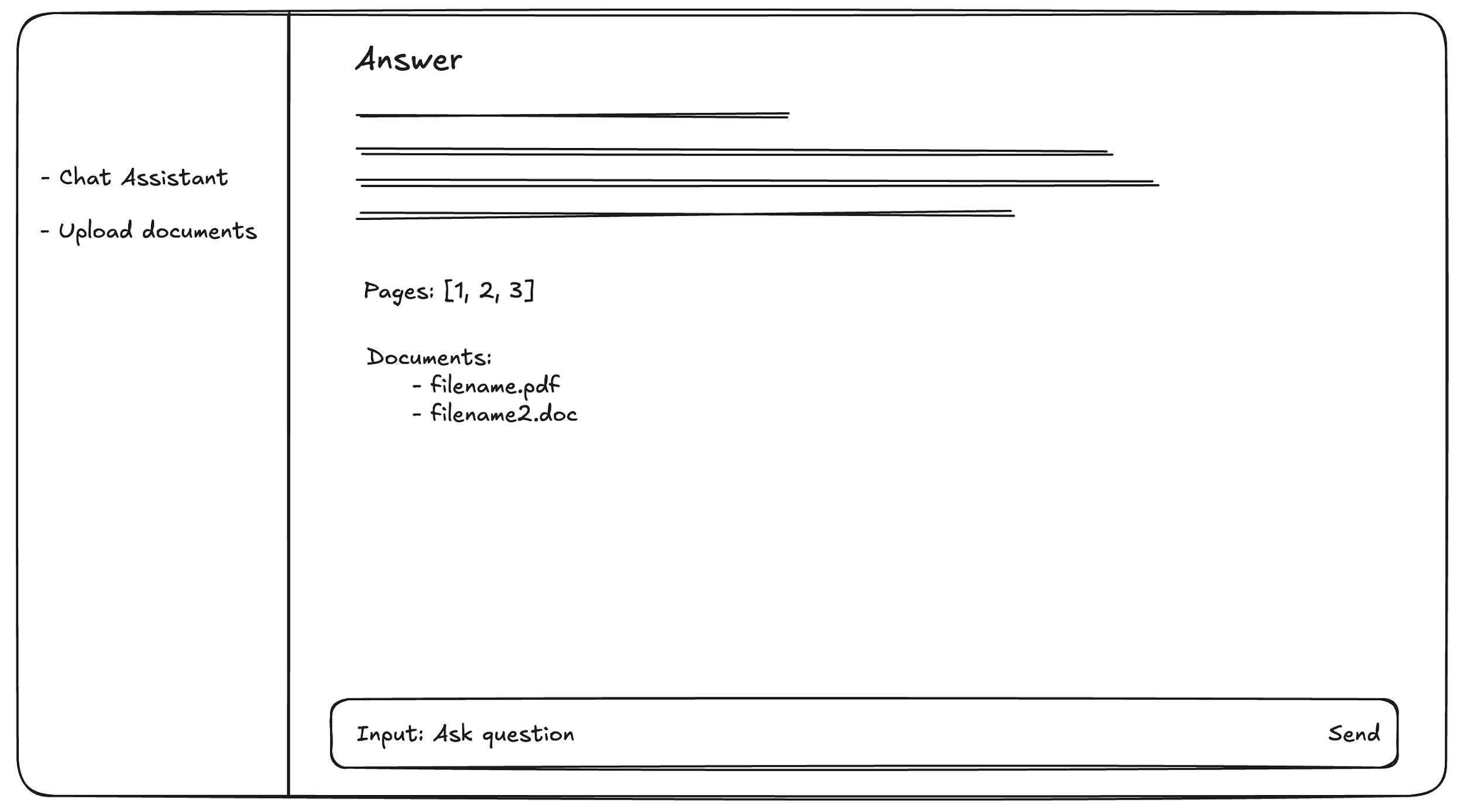
**Technical Assignment: Build an Agentic RAG Application**

**Objective**

Build an Agentic RAG application that enables users to upload documents, interact through a chat interface, and receive accurate answers with citations including page numbers from the original sources.



**Requirements**

**Core Features**

* **Chat Interface**:
  + Implement an **interactive interface**.
  + The interface should be **simple and functional** (design is not a priority).
* **Question Answering**:
  + Users upload documents and ask questions.
  + The system should:
    - Provide answers based on uploaded documents.
    - Include file name and page number as citation.
    - If no answer exists, respond clearly with “No answer found” (or similar).
    - Avoid AI hallucinations.
* **Core Components**:
  + Use **LlamaIndex** framework for retrieval and indexing (mandatory).
  + Use any **Vector Database** for storing embeddings, You may choose any DB you prefer (e.g., Neo4j, Weaviate, etc).
* **Optional Components:**
  + **LlamaParse** for document parsing and extraction.
  + Additional tools, frameworks, or APIs that enhance performance or user experience.
* **Agentic Behaviour** (Optional):
  + Support **multi-step reasoning** for complex queries.
* **Conversation Memory**:
  + Support memory so the system can understand **follow-up questions** in context.

**Technical Specifications**

* **Language**: Python
* **Main Tools**: LlamaIndex + (You can use additional libraries or tools if they improve functionality).
* **Containerization**: Docker
* Deploying the app on **a free cloud platform** (like Streamlit Cloud, Render, or similar) if possible (read deliverables part).

**Deliverables**

* **GitHub Repository** (or compressed folder) including:
  + Source code.
  + **Dockerfile** and configurations for deployment.
  + **README** with:
    - Setup and run instructions.
    - Architecture overview.
    - Environment variables and configuration details.
* **Optional**: If deployment is not possible, a short screen recording (max 5 minutes) showing the app in action.
* Include the total duration (how many hours/days you spent on the task).

**Test Data**

We will provide:

* A set of **PDF documents**
* A **list of questions with expected answers** for evaluation.

### **Additional Notes on Evaluation**

* A list of **sample questions with expected answers** will be provided for testing.
* These questions and answers serve as **guidelines**, not strict benchmarks.
  + Your answers should be **correct and sourced from the documents**.
  + They **do not need to match the expected answer word-for-word**, but the **accuracy and precision** of the response matter.
  + Avoid fabricating or guessing. If no relevant information exists, return **“No answer found”**.