

SDE-1: Programming Assignment

Task: Develop an AI-Powered Document & Multimedia Q&A Web Application

Objective:

Build a **full-stack web application** that allows users to:

- **Upload PDF documents, audio, and video files.**
- Interact with an **AI-powered chatbot** to ask questions based on the uploaded files.
- **Summarize** the content of the uploaded files.
- Extract **timestamps** from audio and video files for specific topics.
- Provide a **button to play** the relevant portion of an audio/video file corresponding to the chatbot's answer.

Requirements:

Backend:

- Use **Python (FastAPI, Django) or Java (Spring Boot, Quarkus)**.
- Implement an **LLM-powered chatbot** using **OpenAI API / LlamaIndex / LangChain**.
- Use **Whisper API / OpenAI ASR / Deepgram** to transcribe audio/video files.
- Store extracted text and metadata in NoSQL (**MongoDB, Elasticsearch, or Pinecone**) or **RDBMS (PostgreSQL, MySQL)**.
- Implement **automated testing with at least 95% coverage**.
- Write a **Dockerfile** to containerize the app.
- Set up a **CI/CD pipeline** using GitHub Actions.

Frontend:

- Build a **React, Vue, or Angular-based UI**.
- Implement an **upload interface** for PDFs, audio, and video files.
- Provide a **chatbot UI** where users can ask questions.
- Display **summaries of uploaded content**.
- Show **timestamps** for specific topics in videos/audio.
- Include a "**Play**" button to jump to the relevant timestamp in an audio/video.

Infrastructure & Deployment:

- Use **Docker Compose** for multi-container setup.
- Deploy on **AWS/GCP/Azure** (Optional but preferred).

Bonus Points:

- Implement **vector search** using FAISS or Pinecone for **semantic search** on documents.
- Provide **real-time chat streaming responses**.
- Enable **multi-user authentication (OAuth, JWT, or API keys)**.
- Implement **rate-limiting and caching** (Redis).

Deliverables:

- **GitHub Repository with:**
 - Source code
 - README.md (◦ setup, API documentation, testing, running instructions)

- **Automated test coverage (95%+)**
- **Live Demo URL**- Record a walkthrough video explaining the application functionality and code. Share the video via YouTube or Google Drive link.