

Project Structure and File Explanation

README.md

Contains project description, setup steps, and purpose: building a conversational AI using LLM and FAISS for document-based question answering.

app.py (root)

Main entry script. Runs the Streamlit app defined in src.app.

requirements.txt

Lists required Python packages: OpenAI, LangChain, FAISS, Streamlit, dotenv, etc.

src/app.py

Core Streamlit web app logic: handles UI, PDF uploads, querying the AI system, and displaying results.

src/components/learner.py

Handles interaction with LLMs. Builds prompts, sends them to OpenAI, receives and returns generated answers.

src/components/ragl_system.py

Implements Retrieval-Augmented Generation pipeline: loads PDFs, chunks text, embeds using OpenAI, stores in FAISS, and queries them.

src/components/vector_db.py

Manages FAISS vector store: saving, loading, searching. Stores both the index and associated metadata.

src/utils/config.py

Loads configuration from ``.env``, including model parameters, API keys, file paths, etc.

src/utils/model_loader.py

Selects which model to use (e.g. OpenAI vs local) based on config.

__init__.py files

Declare Python packages. These files are usually empty or minimal.