Amlgo Labs – Junior AI Engineer Assignment

Project Title: Document-based RAG Chatbot
Candidate Name: Ravishankar Singh
Submission Date: [05-07-2025]

---

1. Project Overview

Objective:
To build a chatbot that can answer questions based on document content using Retrieval-Augmented Generation (RAG) architecture.

Problem Statement:
Manual searching through long PDFs or policy documents is time-consuming. This chatbot enables interactive, fast, and document-accurate responses.

---

2. Technical Architecture

Pipeline:
User Query → Retriever → Vector DB (Chroma) → Prompt Template → LLM (phi via Ollama) → Response (Streamed)

Steps:

- PDF is cleaned and preprocessed
- Text is split into chunks of 100–300 words
- Each chunk is converted to vector embeddings (`all-MiniLM-L6-v2`)
- Chunks are stored in Chroma vector database
- Semantic retriever finds best chunks for a query
- Custom prompt + query + chunks go to LLM
- Response is streamed to user with source text shown

---

3. Sample Interactions

Q: What is the return policy?

Q: Who can access my data?

Q: Is cancellation allowed?

---

4. Observations and Limitations

Positives:

- **Real-time streaming of answers** enhances user experience by simulating natural conversational flow.
- **Responses are grounded in the actual content of the uploaded document**, ensuring high factual accuracy and reduced hallucination.
- **The system is lightweight and efficient**, designed to run smoothly on local machines without requiring GPU or cloud dependencies.

Limitations:

- The Phi model performs reliably for most queries; however, in some complex or edge-case scenarios, responses may be brief or partially accurate. This presents an opportunity for further fine-tuning or model upgrades.
- The current implementation is optimized for single-document QA, ensuring focused retrieval and low latency. Support for multi-document inputs can be added in future iterations.
- Chunking is currently static and based on sentence lengths to maintain context. While it works well in general, adaptive chunking based on document structure could further improve retrieval precision and model output.

---

5. Conclusion

This chatbot demonstrates how RAG pipelines can be implemented efficiently using open-source tools.

Tools Used:
LangChain, HuggingFace Embeddings, ChromaDB, Streamlit, Ollama