

AI Intern / Junior AI Developer

Mini Project: Build a Local GPT Assistant (RAG-Style System)

Goal

Build a small AI assistant that:

- Lets users **upload documents or datasets** (TXT, PDF, CSV, DOCX).
- **Indexes and stores** the data locally.
- **Answers questions only** from the uploaded data (no hallucination).
- If the answer isn't found, it should respond:

👉 "I don't know based on the provided documents."

Key Tasks

1. Data Upload & Processing

- Accept file uploads (.txt, .pdf, .csv, .docx).
- Convert content to text and split into small chunks (~300–500 words).
- Store chunks with metadata (file name, chunk ID).

2. Embedding & Search

- Use an embedding model such as **SentenceTransformers (all-mpnet-base-v2)** or **OpenAI Embeddings**.
- Store embeddings in a **vector database** (FAISS / Chroma).

3. Query & Response

- When a user asks a question:
 - Retrieve top relevant chunks (top 3–5).
 - Use an **LLM (OpenAI / Hugging Face)** to generate an answer **only using retrieved chunks**.
- If no relevant context found (below similarity threshold), return:

"I don't have enough information in the uploaded documents."

4. Simple Interface

- Build a small **FastAPI** or **Streamlit** app where:
 - Users can upload files
 - Ask questions
 - View the assistant's answers and referenced sources

Tech Stack (Suggested)

- **Language:** Python 3.10+
 - **Libraries:** SentenceTransformers / OpenAI API, FAISS / Chroma, FastAPI / Streamlit
 - **Optional Tools:** LangChain / LlamaIndex for RAG pipeline support
-

Deliverables

- Working **application link** (live demo)
 - **GitHub repository** with clean, documented code
 - Short **README (1 page)** including:
 - Overview of approach
 - Libraries used
 - Example input & output
 - How out-of-scope queries are handled
-


Timeline

- Duration: **6–8 hours of active work**
 - Submission Deadline: **Within 24 hours**
-

Submission Instructions

Send your submission to:

 relation@developerbazaar.com

 **+91 83193 64008**

Include the following in your email:

1. **Working URL** – (Streamlit / FastAPI hosted link)
2. **GitHub Repository Link** – (complete project code)

Subject: AI_RAG_Assignment_<YourName>

Important Note

- Submissions **without a working demo link and GitHub repo** will not be considered for the next round.

- Ensure the link is publicly accessible for testing (Streamlit Share, Render, Hugging Face Spaces, or localtunnel).
-

Tip for Candidates

Keep it simple, logical, and functional.

We're looking for clear thinking, clean code, and meaningful results — not complexity.