

Intern Task: AI-Powered Chat Application using RAG, Redis, and Database

Objective

Build a smart chat application that stores user messages, answers questions using AI, and supports real-time updates or fast access using Redis.

What You Need to Do

1. Create an API to send and receive messages.
2. Store chat messages in a database (MongoDB or PostgreSQL).
3. Use RAG (Retrieval-Augmented Generation) with OpenAI or LangChain to answer questions from a document (like PDF or TXT).
4. Use Redis to either:
 - Cache recent messages for faster access, or
 - Handle real-time messaging using Pub/Sub.
5. (Optional) Build a simple front-end UI or host your backend API online.

What to Submit

- GitHub link with all code
- README with clear setup instructions
- Sample screenshot or video of your app working
- Postman collection or curl commands to test your API
- (Bonus) Hosted version or working link

Tools You Can Use

- Backend: Python (Flask or FastAPI) or Node.js
- AI: OpenAI, LangChain, FAISS or ChromaDB
- Database: MongoDB or PostgreSQL
- Redis: For cache or real-time Pub/Sub
- File: Use a sample PDF or text file to extract answers from

Tips for Success

- Keep your code clean and simple

- Use comments to explain your logic
- Make sure your API returns proper responses
- Think of how to make it user-friendly and smart

Deadline & Time Estimate

Estimated Time: 2 days (submit on 22 june)

Accept partial: “If you complete just one endpoint and DB setup properly, that’s fine — submit what you have”

All the best!

- The Cosmic Soul Team.