# AI Chatbot Project – Evaluation and Improvements

This document evaluates the current state of the AI chatbot built using llama.cpp and the DeepSeek model. It highlights existing limitations and suggests areas for enhancement.

## ⚠️ Current Limitations (Cons)

### 🧠 Memory Is Temporary (RAM-Based)

All conversation history is stored in-memory and lost when the server restarts, so there's no continuity across sessions.

### 🪵 Lack of Persistent Logging

No logs are saved for requests/responses/errors, making it hard to debug or analyze performance.

### 🧠 Prompt Engineering Is Manual

Manually crafting the prompt with labels can be brittle and may not align with newer model formats.

### ⚙️ Latency (Due to subprocess + local inference)

Using subprocess.run to invoke llama-cli adds overhead, resulting in slow response times.

### 🚫 No Rate Limiting / Security

The /ask endpoint is exposed to the public with no authentication or usage limits.

### 🧠 Model Limitation

The DeepSeek 1.3B model is small and not fine-tuned for general conversation, limiting its capability.

### 🧪 No Fine-Tuning or Custom Instructions

No persona or system instructions are provided to guide the model's behavior.

### 🖼️ No Support for Multimodal Inputs

The chatbot only supports text input and lacks the ability to process images, audio, or files.

## ✅ Suggestions for Improvement

### 📦 Persistent Memory Storage

Use SQLite, Redis, or file-based storage to retain chat history beyond server restarts.

### 🪵 Add Logging + Analytics

Log all prompts, responses, errors, and token usage using logging libraries.

### 🌐 Use LLM API Instead of Subprocess

Switch to llama-cpp-python or Ollama for in-memory fast inference.

### 🔐 Add Security

Implement API keys, rate limiting, and prompt injection prevention.

### 🚀 Model Upgrade

Use larger and more general-purpose models like LLaMA 3, Mistral, or Yi series.

### 🧠 Add Persona / Instructions

Incorporate system prompts to guide the assistant's behavior and tone.

### 🖥️ Streamlit / Web Dashboard

Improve the frontend UI with history, reset, file upload, and live logs.

### 🧪 Add Tests & Health Monitoring

Set up unit tests, uptime monitoring, and resource dashboards.

## 🌟 Bonus Feature Ideas

|  |  |
| --- | --- |
| Feature | Tools / Tech |
| File uploads (e.g., PDFs) | LangChain, PyMuPDF, PDFMiner |
| Voice support | Coqui TTS, Whisper, Vosk |
| Embedding Search | FAISS, Chroma, LlamaIndex |
| RAG (Retrieval-Augmented Generation) | Combine knowledge with vector search |
| Memory Profiles (per user) | User-based history using auth |