# Conversational AI Chatbot - Design Document

## 1. Introduction

This document outlines the architecture and design of the Conversational AI Chatbot with context-aware retrieval, NER-based query refinement, and SQL schema integration. The system is designed to provide coherent, memory-aware responses while ensuring that SQL queries are generated with an understanding of the database schema.

## 2. System Overview

### 2.1 Objective

The chatbot aims to:  
- Maintain short-term and long-term memory of conversations.  
- Extract relevant contextual information using FAISS and a predefined variable (chat\_context).  
- Convert user queries into standalone questions for improved understanding.  
- Enhance query processing using Named Entity Recognition (NER).  
- Integrate a database schema from an SQL file (context.sql) for SQL-aware responses.

### 2.2 Workflow

The chatbot operates as follows:  
1. The user inputs a query.  
2. The system retrieves past interactions from memory and FAISS.  
3. The query is rewritten into a standalone question.  
4. Key concepts are extracted using spaCy NER.  
5. The retrieved context and schema are included in the prompt.  
6. The LLM generates a response based on memory, context, and schema.  
7. The chatbot displays the response and stores the interaction for future reference.

## 3. Key Modules

### 3.1 Context Retrieval

Extracts historical conversation data from the chat\_context variable and uses FAISS semantic search to retrieve related information.

### 3.2 Standalone Question Rewriting

Ensures user queries are self-contained to improve understanding by the LLM. Uses GPT-4 for rewriting user input into standalone questions.

### 3.3 Concept Extraction Using NER

Named Entity Recognition (NER) is applied to extract key topics from standalone questions to refine search retrieval accuracy.

### 3.4 FAISS Vector Search

Uses FAISS with HNSW indexing for efficient semantic similarity search and retrieval of relevant past interactions.

### 3.5 Memory Management

Manages short-term conversation memory, keeping only the last few exchanges to optimize performance.

## 4. System Constraints & Future Enhancements

\*\*Constraints:\*\*  
- Requires GPT-4 API access.  
- FAISS indexing may need optimization for larger datasets.  
- Assumes consistent formatting of chat\_context.  
  
\*\*Future Enhancements:\*\*  
1. Dynamic Schema Updates - Reload context.sql if modified.  
2. SQL Execution Engine - Execute AI-generated SQL queries against a live database.  
3. Custom NER Model - Improve concept extraction for domain-specific terminology.  
4. User Intent Recognition - Predict query type (SQL vs. general knowledge).

## 5. Conclusion

This chatbot integrates contextual memory retrieval, concept-based query refinement, and SQL schema awareness to provide intelligent, SQL-aware responses. The combination of FAISS, memory buffers, and LLM rephrasing ensures accurate, context-aware conversational AI.

## 6. References

1. Johnson, J., & Goyal, A. (2019). "FAISS: Facebook AI Similarity Search."

2. Vaswani, A., et al. (2017). "Attention Is All You Need." NeurIPS.

3. SpaCy Documentation - https://spacy.io

4. LangChain Documentation - https://docs.langchain.com