ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Mixture of Doctors

an agentic-RAG system for self-management of chronic diseases

Lucia Castellucci, Roberto Mitugno, Luca Samorè

GitHub Repository

{lucia.castellucci2, roberto.mitugno, luca.samore}@studio.unibo.it

Apice Page

Why Mixture of Doctors?

Chronic diseases require continuous management and support, often over a lifetime.

Patient self-management improves outcomes and reduces healthcare costs.

Large Language Models (LLMs) can power chatbots to assist patients with guidance and motivation.

Two main challenges:

- Hallucinations (factually wrong answers).
- System complexity for multi-disease, secure deployment.

Motivation

Fine-tuning LLMs is costly, slow, and prone to “catastrophic forgetting.”
RAG is a more scalable solution: knowledge can be updated without retraining.
Ensures **accuracy** and **up-to-date answers** while protecting patient data privacy.
Open-source models allow compliance with **GDPR** and **HIPAA**.

Concept

Specialized “Doctor” modules for each chronic disease (diabetes, multiple sclerosis, hypertension).

Each Doctor uses **RAG**: retrieves validated medical knowledge before generating answers.

The **Orchestrator** routes queries to the right Doctor(s), enabling cross-disease reasoning.
Responses are combined by a **Synthesizer** into a single coherent answer.

Entirely built on **open-source** LLMs for privacy and local deployment.

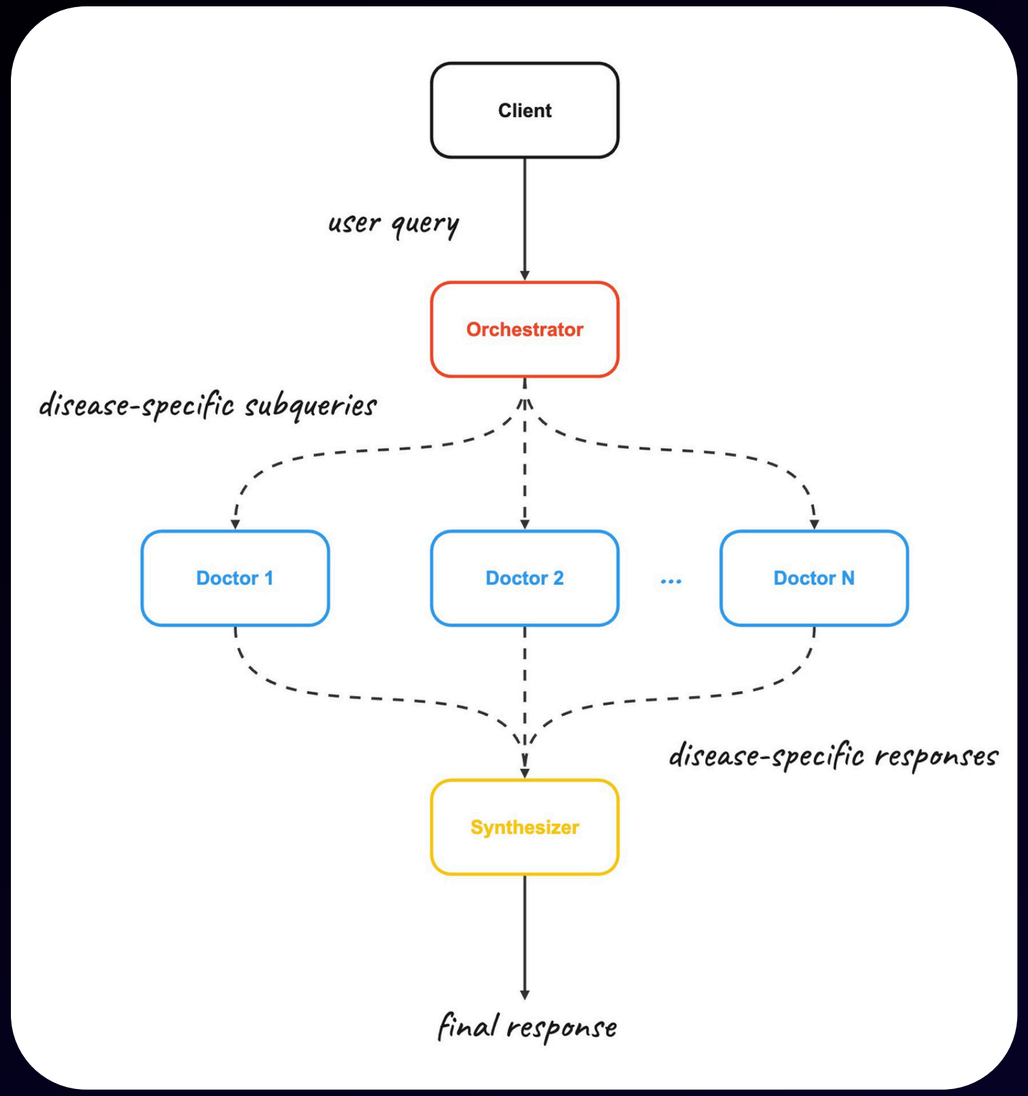
Requirements

Functional

- Query submission through CLI.
- Accurate response generation with RAG.
- Specialized Doctors per disease.
- Logging and persistence of user conversations.
- Support for multi-user identification.

Non-Functional

- GDPR-compliant privacy.
- High availability (≥ 99%).
- Scalability: add new Doctors without re-architecting.
- Usability: intuitive CLI.
- Performance: most responses < 5 seconds.

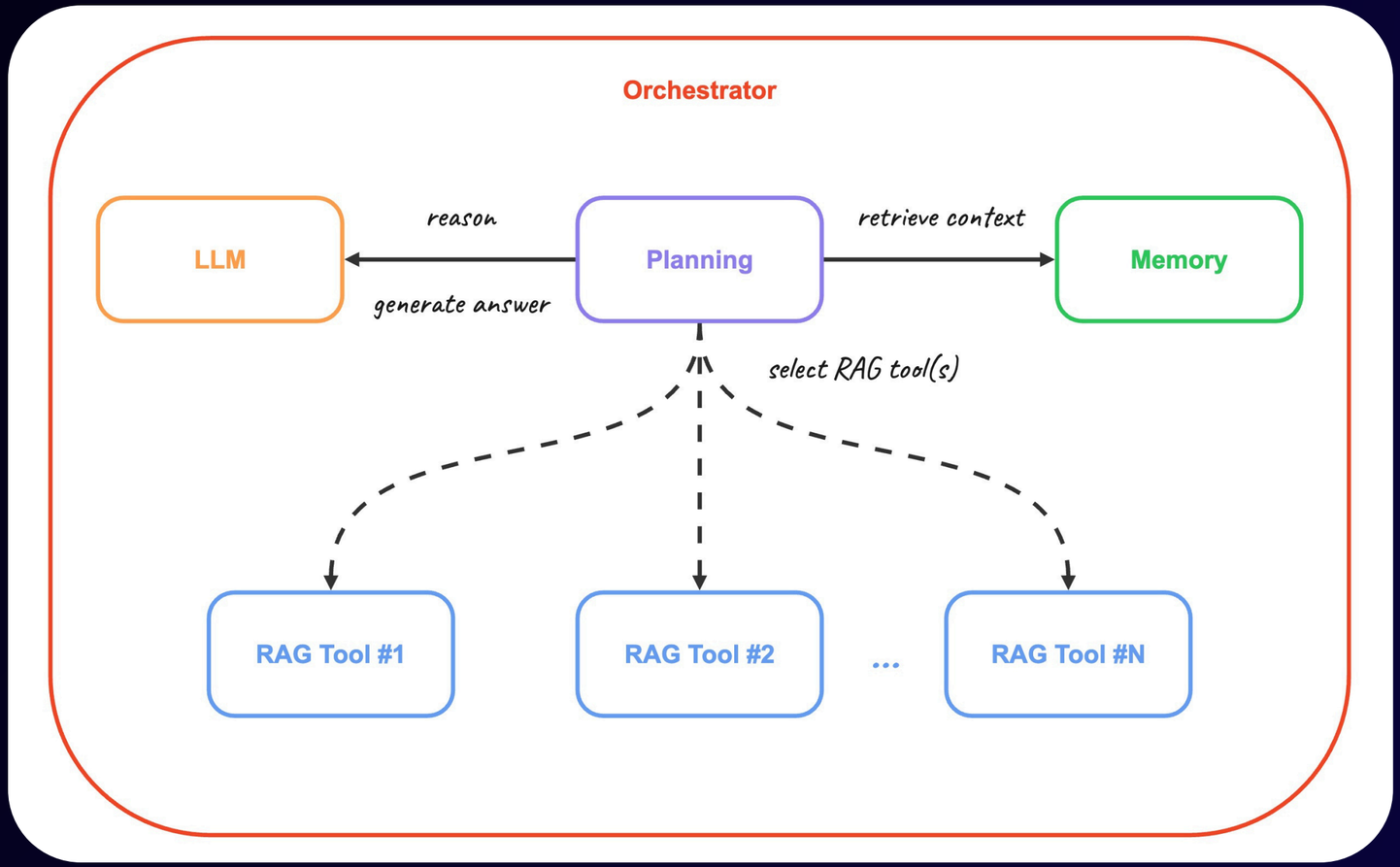


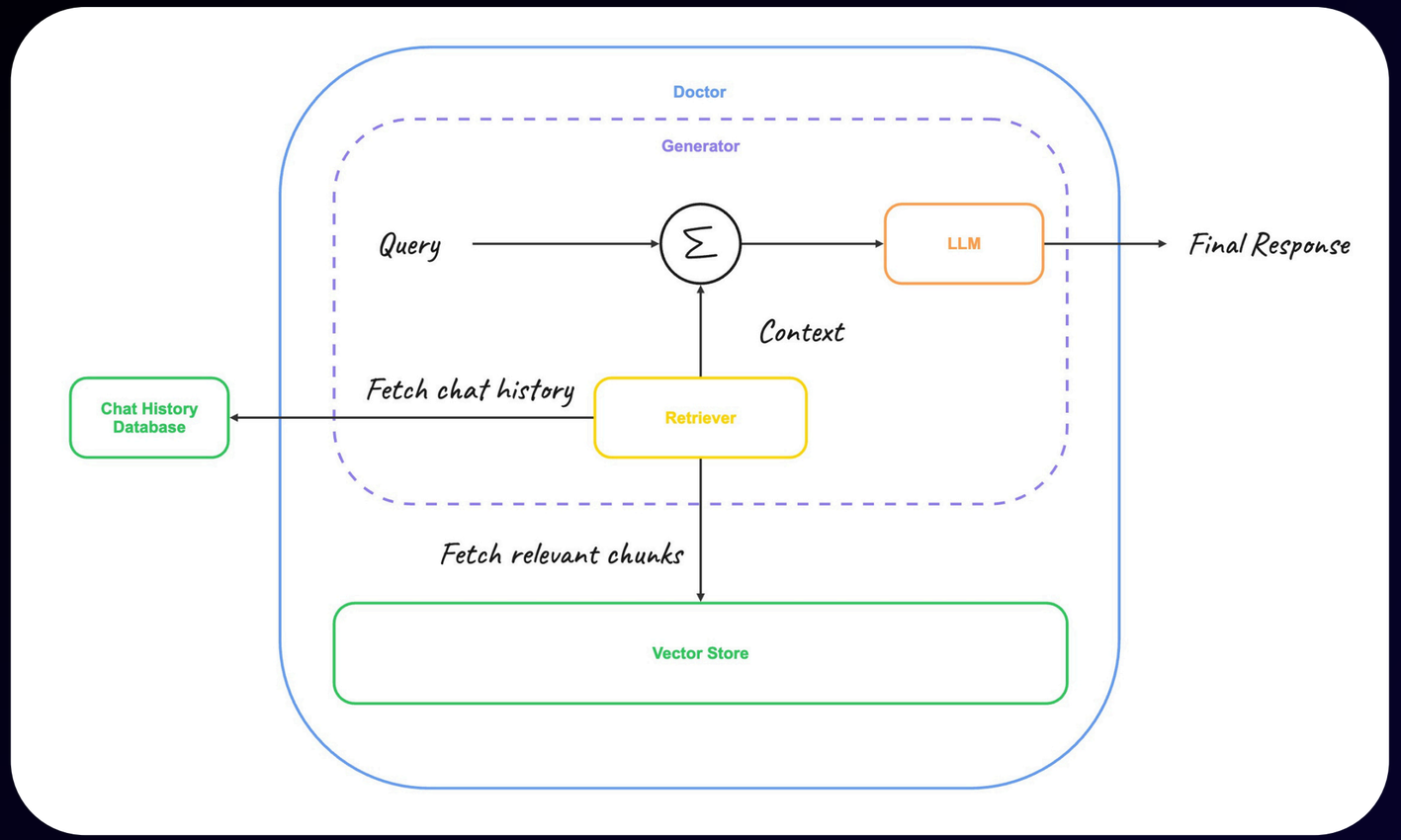
Architecture

- **Orchestrator–Workers model:** Orchestrator delegates tasks to Doctors.
- **Doctors:** disease–specific RAG modules with vector databases.
- **Synthesizer:** merges outputs into one answer.
- **Modular, scalable, and maintainable.**

Orchestrator

- Central entry point.
- Classifies queries as **easy**, **medium**, or **hard**.
- Uses an **LLM** to decide routing.
- Maintains long-term memory via conversation history database.



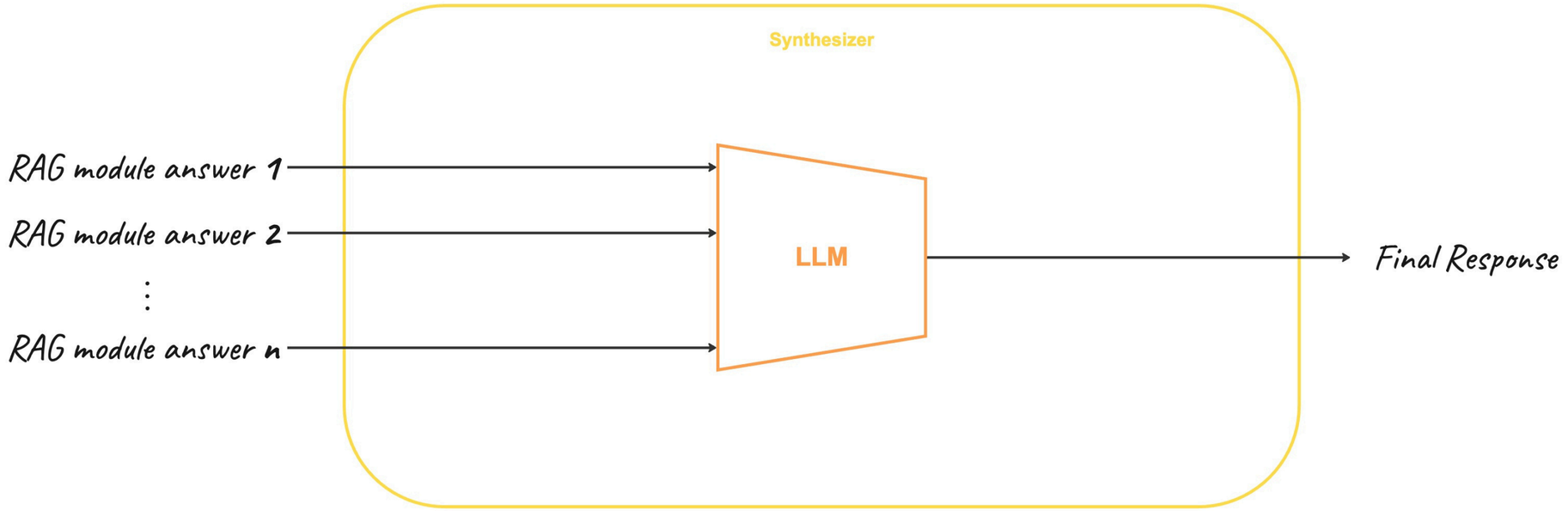


Doctor

- RAG-based expert per disease.
- Retrieves PubMed-sourced chunks from Qdrant vector DB.
- Generates fact-based responses.

Synthesizer

Independent service that aggregates multiple Doctor outputs into a unified, consistent answer.



Interaction

- **Easy queries**: req-res, handled by the Orchestrator.
- **Medium queries**: pub-sub, handled by one Doctor.
- **Hard queries**: decomposed and processed in parallel by multiple Doctors, then synthesized.

Responses are streamed token-by-token

Infrastructure

- **API Gateway** for requests routing.
- **Message Broker** for async communication.
- **Redis Streams** for streaming responses.
- **Document Database** for chat histories.
- **Vector Databases** for embeddings.

Availability

- Stateless services (Orchestrator, Doctors, Synthesizer) scaled horizontally.
- Load balancing ensures resilience.

Fault-Tolerance

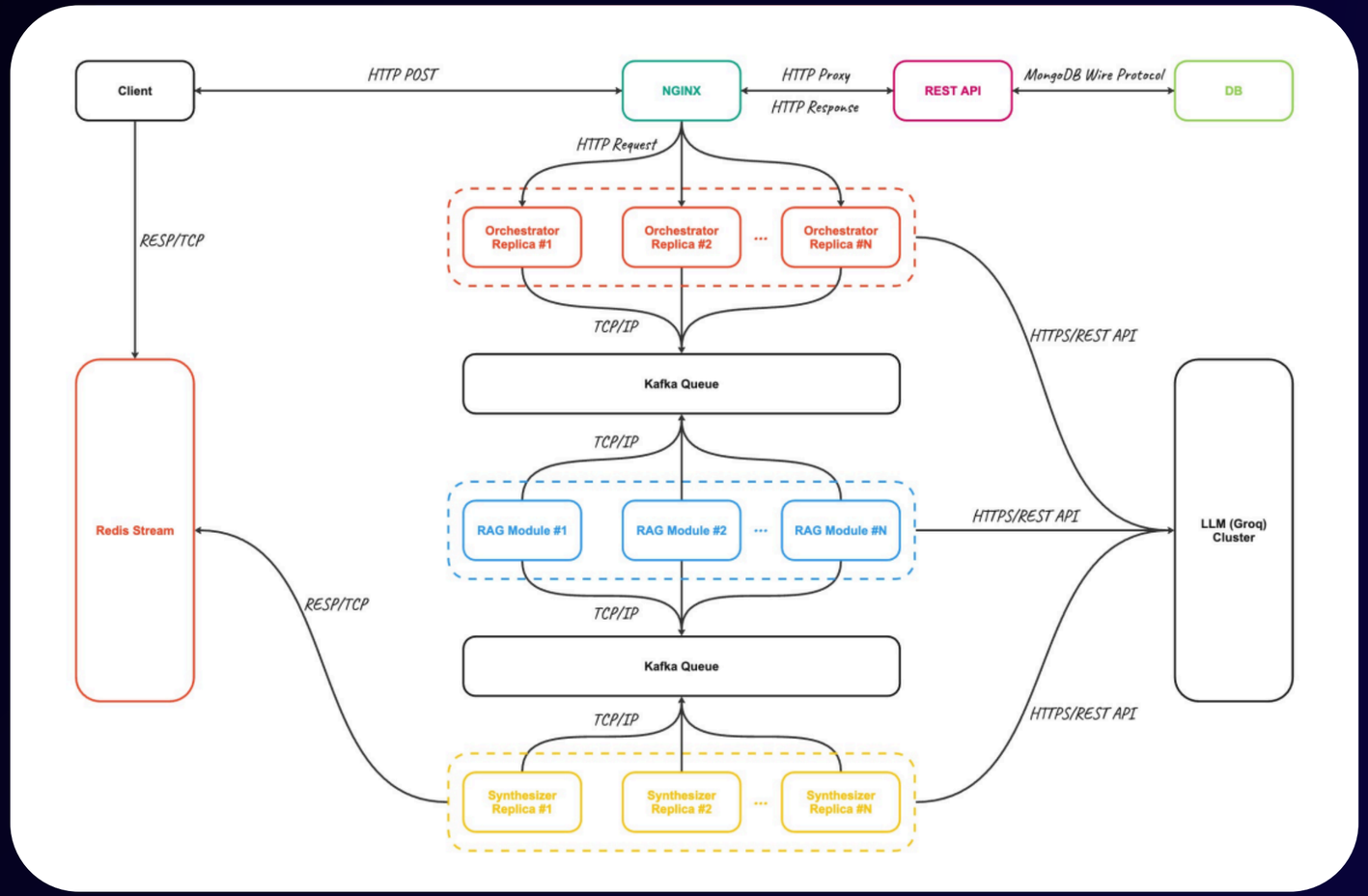
- Databases use leader-follower replication.
- Failed replicas automatically replaced.
- Retry mechanisms prevent permanent loss of queries.

Consistency

- Designed as an AP system (availability + partition tolerance).
- Eventual consistency acceptable for chat history.
- Knowledge bases updated offline.

Implementation

- **NGINX**: Reverse proxy and single gateway for incoming traffic.
- **Redis**: Handles real-time streaming of response fragments for token-by-token generation.
- **Qdrant**: Vector database for indexing and searching semantically similar documents.
- **MongoDB**: Flexible persistence layer for storing conversation histories.
- **Groq API**: Provides access to high-performance, low-latency LLM inference.
- **Docker & Swarm**: Used for service containerization (Docker) and orchestration (Swarm).
- **Apache Kafka**: Event streaming platform for decoupled and fault-tolerant messaging.



Conclusions

Limitations

- Finite Conversational Memory
- Standard RAG Implementation
- Minimal Security

Future Work

- Advanced User Interface
- Rigorous RAG Evaluation
- Automated Ingestion Pipeline
- Multi-Agent System

```
MIXTURE OF DOCTORS (MOD)
REPL Interface

Type 'mod' commands directly (e.g., 'mod --help', 'mod new username')
Type 'help' to see available commands
Type 'quit' or 'exit' to exit the REPL

Mixture-of-Doctors> help

Available commands:
  mod [options]           - Execute a MOD CLI command
  help                    - Show this help message
  quit, exit              - Exit the REPL

Examples:
  mod --help              - Show MOD CLI help
  mod new johndoe         - Create a new chat session for johndoe
  mod chat johndoe        - Start or restore a chat session
  mod ask "How are you?"  - Ask a question to the virtual doctor
  mod quit                - Close the current chat session

Mixture-of-Doctors> mod restore luca

=== Chat History ===

Luca: Hello! Can you introduce yourself?
Virtual Doctor: Hello, I'm a specialized medical AI assistant focused on chronic diseases and healthcare topics. I have been designed to provide a
ccurate and comprehensive information on a wide range of medical conditions, treatments, medications, diagnostic procedures, and healthcare practi
ces, with a particular emphasis on chronic diseases. My purpose is to help users understand their health and medical conditions better, and to pro
vide support and guidance in managing chronic illnesses. Please note that while I strive to offer the most current and accurate information, my re
sponses should not be considered a substitute for professional medical advice. If you have specific health concerns or questions, I'll do my best
to provide helpful information, and I always recommend consulting a healthcare professional for personalized guidance and care. What brings you he
re today, and how can I assist you?


Available commands:
  - ask "your question"    : Ask a question to the virtual doctor
  - quit                   : End the current chat session
```

Demo

Demo Link @YouTube

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Thank You

GitHub Repository

Lucia Castellucci, Roberto Mitugno, Luca Samorè

Apice Page