

# AI Student Support Chatbot for myUMBC Portal

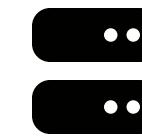
Deliverable #2 - Real world problem statement

Course: 636 Structured Systems Analysis and Design

Group 2: Pavithra (MI31417) , Sakshi (KN34693), Andrew (CO60011)

Date of Submission: December 7<sup>th</sup>, 2025

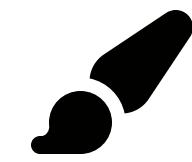
# Team Roles & Contributions



## Pavithra

### Backend + NLP Integration Lead

Led the development of the core chatbot engine, NLP model integration, and server-side logic.



## Sakshi

### UI/UX Design Lead

Led the design of the user interface, interaction flow, and developed high-fidelity prototypes.



## Andrew

### Documentation & Testing Lead

Led project documentation, presentation flow, user research, and comprehensive system testing.

## Leadership Rotation Timeline

Pavithra (Early Phase) → Sakshi (Mid Phase) → Andrew (Final Phase)

# Executive Summary

## Project Goal & Scope

To build and integrate an AI-powered chatbot within the myUMBC portal, designed to provide fast, accurate, 24/7 support to students.

This initiative aims to reduce support ticket volume, streamline student services, and enhance the overall student experience by providing a single, reliable point of contact for common inquiries.

### Key Scope Areas

- Student Support
- Library Services
- Career Services
- Quick Campus Resources

### Core Features

- Natural Language Query
- Knowledge Base Search
- Escalation Workflow
- Analytics Dashboard

# Outcomes & Future

## Major Outcomes & Challenges

**Achieved:** Functional prototype with core NLP and UI components.

**Challenges:** Data access limitations, scope refinement, and integration complexities.

**Problems:** Aligning requirements with technical feasibility and available data.

## Future Roadmap

**Enhanced NLP:** Improve intent recognition and contextual understanding.

**Secure API Integration:** Connect with live campus systems for real-time data.

**Mobile-First Design:** Optimize for mobile app integration and accessibility.

# Project Description

## The Problem Statement & Opportunity

Students face fragmented and scattered support systems , leading to frustration and repetitive inquiries that burden staff, spam of emails and doubts regarding basic campus resources - across various departments.

This project seizes the opportunity to provide a single, reliable, and intelligent point of contact, offering fast, accurate, 24/7 answers to improve student satisfaction, easy access to campus resources and operational efficiency.

## Key Stakeholders

-  Students & Staff (enrolled or have myUMBC Login)
-  Career Center & Handshake
-  IT Help Desk
-  ISSS
-  Library Services
-  myUMBC Portal Team

# System Request & Feasibility Analysis



## Technical Feasibility

Viable through secure NLP/LLM model hosting, integrated knowledge base, and API connectivity within the UMBC ecosystem.



## Economic Feasibility

No extra-cost prototype with significant long-term cost savings potential via reduced support ticket load and increased staff and resources efficiency.



## Organizational Feasibility

Strong alignment with UMBC's strategic goals for efficiency and student experience, with staff support for escalation workflows.

# Management & Planning Evolution

## Development Model: Agile/Iterative

**Pros:** Allowed for flexibility, continuous feedback, and adaptation to changing requirements.

**Cons:** Required high-end tech team led to some scope adjustments mid-semester.

## Communication Plan

Weekly meetings, Google Chat for instant communication, and clear email threads for task division.

**Outcome:** We experienced no communication issues.

After the Mid-Term review, we changed the plan of individual training the data-set to using NLP, LLM and Hugging Face. Milestones were adjusted from the initial plan to a revised plan to better accommodate technical discoveries and integration challenges.

# Requirements Elicitation & Summary

## Elicitation Methods

- Internal brainstorming sessions
- Reference analysis of UMBC support portals
- Review of sample documents for incoming students.

## Functional Requirements

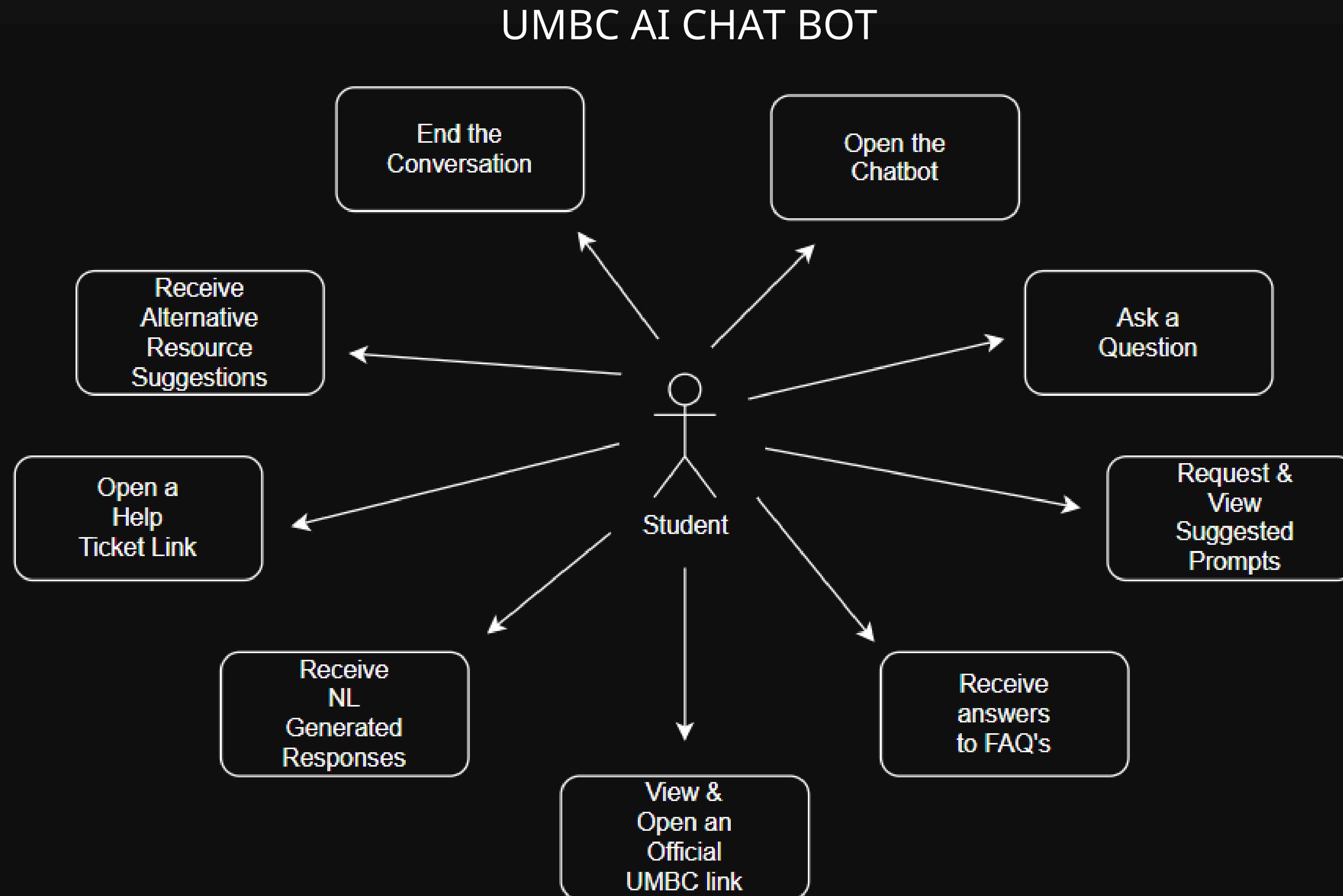
- The system will have an intuitive chatbot interface
- The system will perform knowledge-base search & retrieval
- The system will perform automated escalation workflow
- The system will provide ease of access to frequently used external portals linked with UMBC (Handshake, ISSS, Transit)

## Non-Functional Requirements

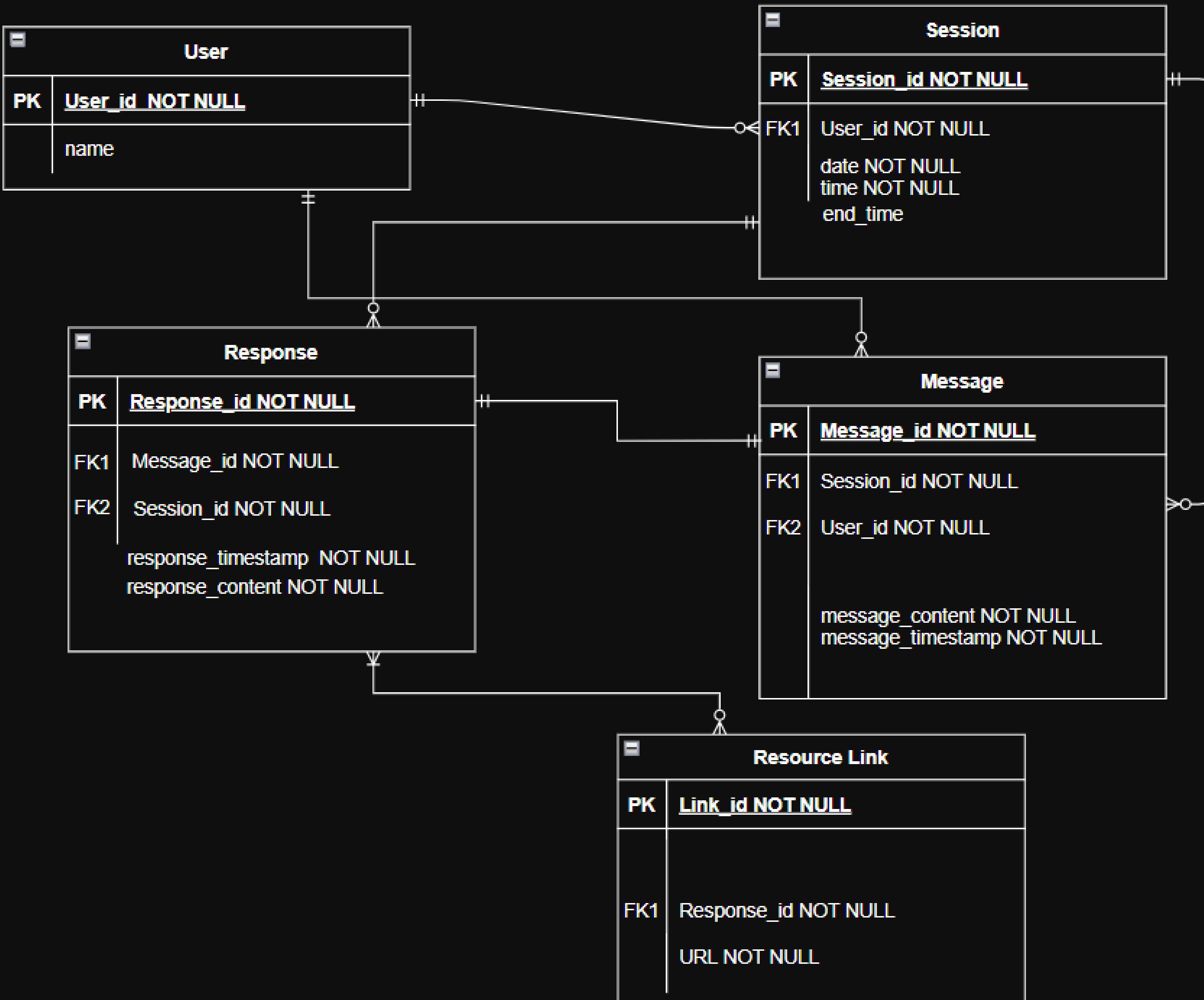
- Response time: **3-5 seconds**
- Concurrent users: **1 per computer as of now it's based on code**  
(In the future >15k users will be able to access at a time.)
- FAQ accuracy: **95%**

- Desktop compatibility
- Accessibility (WCAG 2.1 AA) compliance
- No Personally Identifiable Information collection or retention

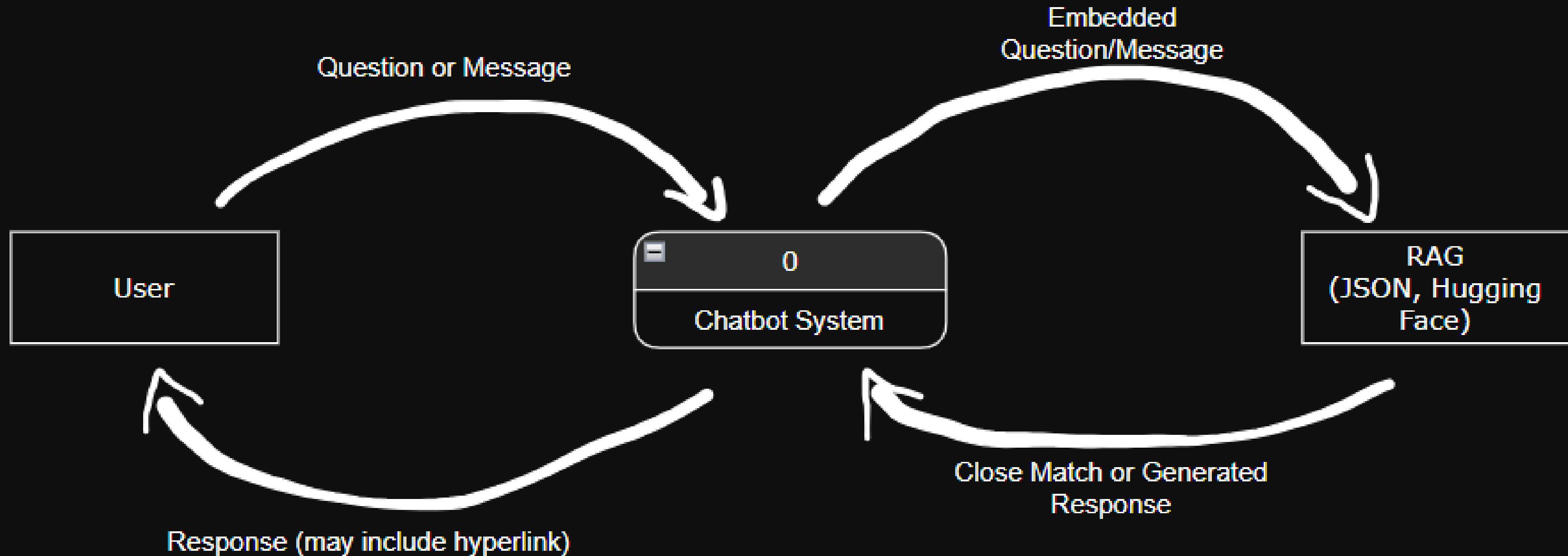
# Use Case Diagram



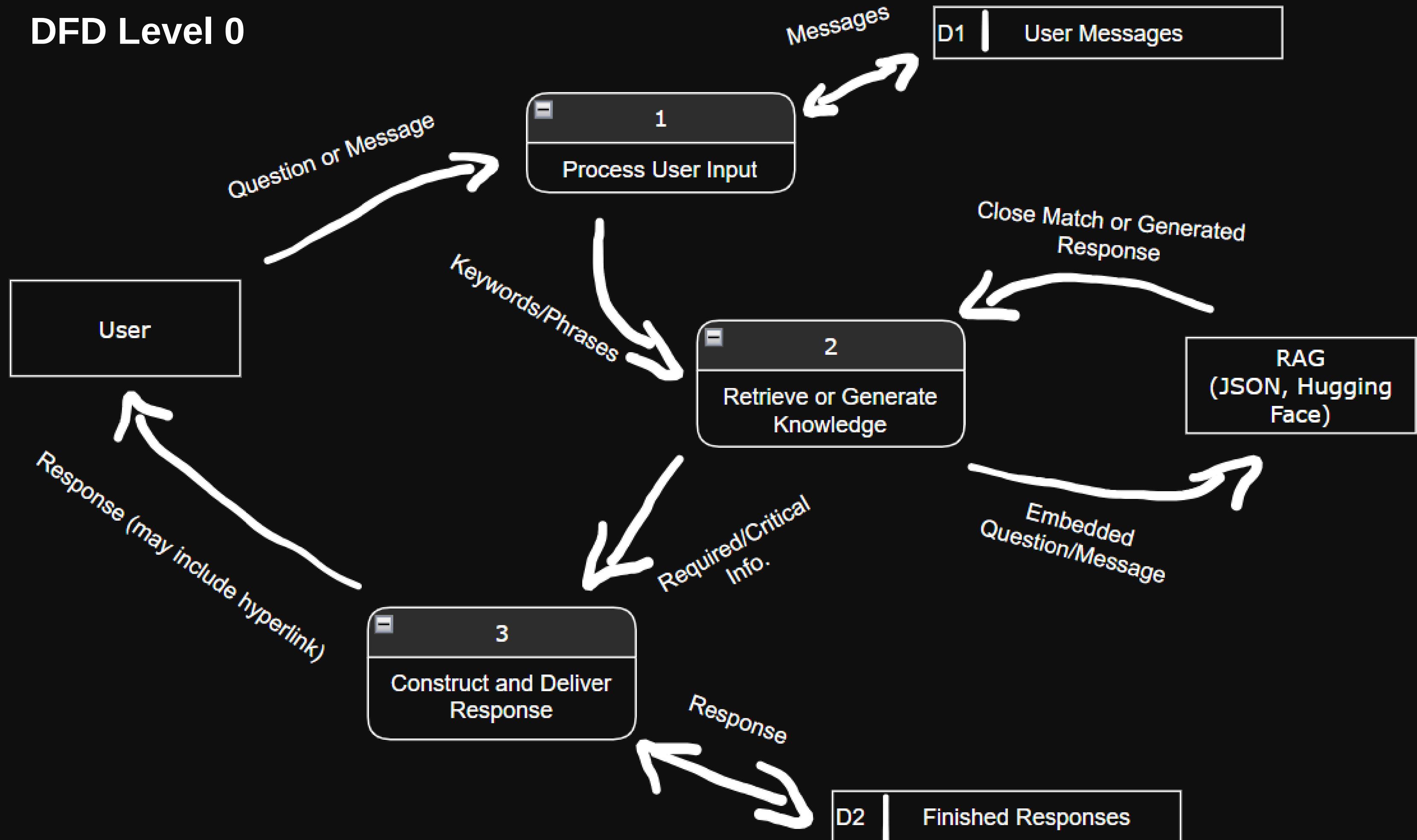
# Entity-Relationship Diagram



# DFD Context-Level



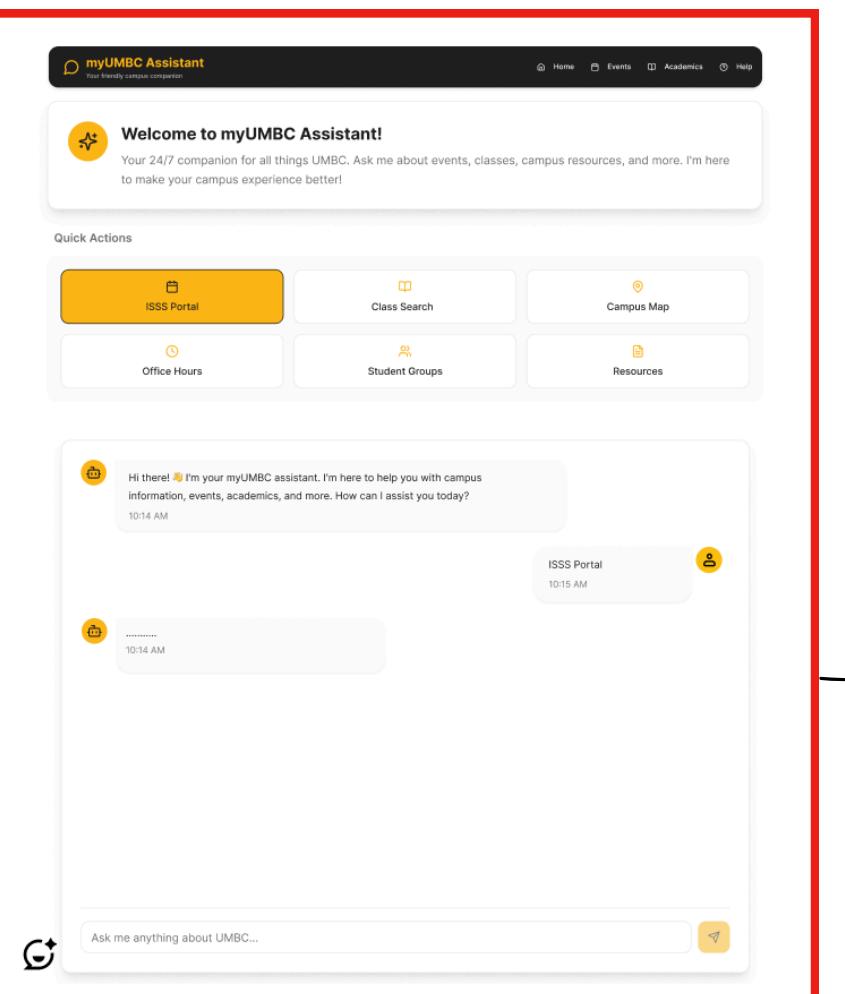
## DFD Level 0



# User Interface & Mockup

## UI Mockup Screenshot

The screenshot shows the myUMBC homepage with a dark header bar. The top navigation includes 'myUMBC' (with a logo), a search bar, and links for 'Profile', 'Guide', 'Events', 'Groups', and 'Help'. A 'Log In' button is on the far right. The main content area features a date banner 'TUESDAY November 25, 2025'. Below it are several sections: 'HGST WINTER COURSE SCHEDULE 2026' with a thumbnail, 'Being Black Abroad' with a thumbnail, 'HGST Course Offerings Winter 2026' with a thumbnail, 'Picturing Mobility: Black Tourism and Leisure during the J...', 'Public Forum: The Ethics of Cryptocurrency' with a thumbnail, 'NSF HDR: Scientific-MOOD FAIR Challenge' with a thumbnail, 'LIVE EVENT VIRTUAL HACKATHON' with a thumbnail, 'Cycle w/ Alex 25' with a thumbnail, and 'The Ethics of 4 Cryptocurrency' with a thumbnail. A sidebar on the left lists 'Trending' posts: 'Honoring Alice Wong (1974-2025)', 'Meet Thalia Anora', and 'NEW! UMBC International Student Short Film Screening'. Another sidebar on the right lists 'Discussion Groups' such as 'Academics', 'Campus Life', 'Classifieds', etc. The footer contains copyright information and links for 'Use of Student Data', 'Equal Opportunity', and 'Safety Resources'.



A screenshot of the myUMBC Assistant mobile application showing a conversation with the AI assistant. The assistant's messages are in blue bubbles, and user messages are in grey bubbles. The first message from the assistant is 'Hi there! 😊 I'm your myUMBC assistant. I'm here to help you with campus information, events, academics, and more. How can I assist you today?'. The user responds with '10:14 AM'. The assistant replies with '10:15 AM'. The user then types 'Ask me anything about UMBC...' and sends it at 10:14 AM. The assistant responds with a circular arrow icon and the text 'Ask me anything about UMBC...'. The interface includes a 'Quick Actions' section with cards for 'ISSS Portal', 'Class Search', 'Campus Map', 'Office Hours', 'Student Groups', and 'Resources'.

# System Design & Future Roadmap

## Acquisition Strategy

A custom build leveraging modular components and integrating with existing campus APIs and authentication systems.

## UI Design Summary

A clean, UMBC-branded interface featuring a central chat window, quick action links, and clear fallback options for escalation.

## Limitations & Assumptions

Prototype-level NLP, limited training dataset, and assumed access only staff or students who have myUMBC.

## Future Work

- ❖ Full API integration with live campus systems.
- ⌚ Advanced analytics and user behavior insights.
- 👤 Improved chatbot personalization and learning.
- 📱 Dedicated mobile app integration and optimization.

# Code Structure & Key Modules

## Frontend

Handles user interaction, chat interface, and presentation logic.

## Backend

Manages server-side logic, request routing, and model communication.

## API Connectors

Manages communication with external campus systems and services.

```
4     "answer": "Log into myUMBC, go to 'Student Services', and select 'Register for Courses'.",
5     "topic": "Academics",
6     "link": "https://registrar.umbc.edu/services/registration/"
7   },
8   {
9     "question": "Where can I find career services?",
10    "answer": "Visit the Career Center on campus or check their website for resources and appointments.",
11    "topic": "Careers",
12    "link": "https://careers.umbc.edu/"
13  },
14  {
15    "question": "Where can I find dining options on campus?",
16    "answer": "Visit the UMBC dining website for locations, menus, and hours of operation.",
17    "topic": "Campus",
18    "link": "https://umbc.edu/undergraduate/life-at-umbc/where-youll-eat/"
19  },
```

1. Image: FAQ UMBC data

```
16  BASE_DIR = os.path.dirname(os.path.abspath(__file__))
17  FAQ_JSON = os.path.join(BASE_DIR, "faq.json")
18  INDEX_PATH = os.path.join(BASE_DIR, "faq.index")
19  STORE_PATH = os.path.join(BASE_DIR, "faq_store.json")
```

2. Image: FAQ Vector creation

```

151 def answer_query(q: str) -> str:
263 You are UMBC Buddy – a friendly, conversational assistant.
264
265 You are given several FAQ entries about UMBC. Use them as your main information source.
266 Your job is to paraphrase and combine whatever is useful to answer the student's question.
267
268 Guidelines:
269 - Answer in 2-4 short, friendly sentences.
270 - Talk directly to the student ("you").
271 - Be clear and confident.
272 - Paraphrase (don't copy) the FAQ answers.
273 - If a link from the FAQs would genuinely help (like to see a list of programs, deadlines, hours, or application info),
274 include at most ONE link at the end on a new line starting with "More: ".
275 - Avoid saying things like "I'm not sure" if the FAQ clearly answers it.
276 FAQ CONTEXT:
277 {faq_context}
278
279 Student question: "{q}"
280 Your answer:
281 """ .strip()
282
283     try:
284         if HF_API_KEY:
285             out = hf_generate(prompt)
286         else:
287             out = local_generate(prompt)
288     except Exception as e:
289         print("[error] LLM failed → using raw FAQ:", e)
290         out = faq_main["answer"]
291
292     out = out.strip()
293
294     # If they explicitly asked for link/website and there is a main FAQ link, ensure it's included once
295     if detect_any(LINK_FOLLOWUP_TRIGGERs, q1) and "http" in main_link and main_link not in out:
296         out += f"\nMore: {main_link}"
297
298 return out

```

Our chatbot uses a Retrieval-Augmented Generation (RAG) approach, where a sentence-transformer model retrieves the closest FAQ using vector embeddings, and the Microsoft Phi-3 Mini Instruct LLM generates a fallback response when no close match is found.

### 3. Image: Chatbot Model

# Conclusion & Expected Impact

## Key Contributions

A functional prototype demonstrating a viable, user-friendly AI chatbot tailored for the myUMBC portal.

## Expected Impact

Significant improvement in student experience through instant support and measurable organizational benefits via reduced staff load, spam emails for regular questions, also helping out students to find a go-to AI source related to UMBC.

## We used AI for 1 tasks:

ChatGPT helped us in organizing/building the JSON files. We gave it the information to put into those files, and the AI organized them into a polished version. There were 500+ JSON data entries, so doing all of them would be tedious.

## Team Reflections & Next Steps

The project successfully met its core objectives, though it is limited by its prototype nature. Key lessons include the importance of early data access and iterative testing.

**Next Steps:** Move towards production deployment, expand personalization, and conduct a pilot study with real student users.

# Thank You!