CSI5180 Project Definition

Group 20

Ovais Azeem (300112311)
Chikamso Olomukoro (300455876)

1. The title of our project is EventSphereAi.

2. a) We will produce a functional full-stack QA system web application that allows users to discover events happening in their area. It provides clear, personalized, and conversational answers to queries such as "What concerts are happening this weekend?" or "Are there any free workshops near me tomorrow?" The system simplifies event discovery by fetching real-time information about concerts, workshops, festivals, and other activities. Users can specify their preferences, like the type of event, location, budget, and date, and EventSphere delivers tailored recommendations, making it easy to find events that match their interests. It also provides event details such as timing, location, and ticket prices in a user-friendly format. We both have a strong foundation in programming, algorithms, and software development. This is as a result of both us having a bachelor's degree in computer science and an idea of fundamental AI and ML concepts like NLP, entity recognition, and intent classification.

b) We will be using intent detection to identify the purpose of the user's query. Named entity recognition will be used to extract specific entities like "concert", "Ottawa" from the user's query. APIs will be used to fetch real-time event data from Ticketmaster or other third-party sources. Response Generation will be used to structure event data into user-friendly responses. Finally, for testing and evaluation, standard QA benchmarks will be used like precision, and recall for intent detection.

c) The final deliverable is a functional full-stack QA system web application that allows users to discover events happening in their area. Anyone will use this application to find an intuitive way to find events based on preferences like type, location, and budget. Also, event platforms like Ticketmaster or local tourism boards to enhance user engagement. EventSphereAi contributes to the field by demonstrating the practical use of QA systems with API integration to solve real-world problems. It sets a precedent for applying conversational AI to personalized event discovery.

d) We will define what is included or excluded to ensure the project is achievable within the 50 hours allocated for 2 members. What is included is: core functionalities, response generation, API integration, Basic UI, and testing and debugging. What is excluded is advanced features like

conversational memory, extensive personalization, scalability (not able to handle large-scale user traffic), and a polished UI (the UI will be minimal).

3. We will develop EventSphereAI as a full-stack web application. The backend will be built using Python for handling API integrations (e.g., Ticketmaster, Eventbrite) and natural language processing with libraries like SpaCy and HuggingFace for intent detection and entity recognition. The frontend will leverage React to create an intuitive user interface, enabling natural language queries and displaying event details. Communication between the frontend and backend will utilize RESTful APIs. For response generation, we will use templating tools like Jinja2, and event data will be fetched and processed in real-time. Deployment and hosting will utilize AWS or Heroku, ensuring scalability and accessibility. These technologies collectively enable a seamless and user-friendly event discovery experience.

4.

Activity	Why	Time Planned	Deliverable
Research event APIs (e.g., Ticketmaster, Eventbrite)	Understand the available event APIs and select one for integration.	3h	API documentation summary and chosen API for the project.
Study React and frontend design	Familiarize with React for creating the user interface.	3h	Notes or tutorials completed with sample UI components.
Design application architecture	Define the overall system flow and component interactions.	4h	Diagram or document detailing system architecture.
Implement API integration	Build the backend to fetch event data from selected APIs.	6h	Backend code that retrieves and processes API data.
Develop intent detection logic	Enable understanding of user queries (e.g., intent, entity extraction).	6h	Code that classifies user intents and extracts entities like date and location.

Build response generation module	Generate user-friendly event recommendations in natural language.	5h	NLG module that creates conversational answers.
Create the frontend interface	Develop the user-friendly web interface for the QA system.	8h	React-based frontend with input and output functionality.
Integrate frontend and backend	Enable smooth communication between frontend and backend using REST APIs.	6h	Fully integrated web application with data flow.
Test and debug system	Ensure the system is functional and handles edge cases effectively.	5h	Bug-free system with documented test cases and results.
Prepare final deliverables	Compile project demo, write documentation, and create a video.	4h	Project documentation, video demonstration, and presentation slides.