Emotional Intelligence Assessment Chatbot
Software Requirement Specifications

Course Information	
Course Code	CS-3009
Instructor	Farrukh Hasan Syed
Project Team	Ashad Abdullah - 21k-3296 Maaz Imam - 21k-3218 Muhammad Usman - 21k-4890
Date	April 7, 2024

Contents

1 Introduction

1.1 Motivation

The motivation behind this project is to innovate the hiring process by integrating an advanced chatbot capable of assessing a candidate's emotional intelligence (EI). This initiative aims to streamline candidate evaluation for job applications by allowing companies to submit job descriptions, based on which the chatbot generates customized behavioral tests. Leveraging state-of-the-art language models like Google Gemini or Mistral 7b, the chatbot is designed to understand and evaluate candidate responses, the chatbot ensures a precise alignment of candidates' emotional intelligence with job-specific requirements. This approach not only enhances the efficiency of the hiring process but also supports better-informed hiring decisions and contributes to fostering a successful workplace environment.

1.2 Stakeholders

The stakeholders of this project include:

- HR Departments: Responsible for overseeing the hiring process. Requirement elicitation will involve interviews and surveys to understand their needs and challenges in assessing candidates' EI.
- Candidates: Individuals applying for positions. Elicitation will include user testing and feedback sessions to ensure the chatbot's assessments are fair and understandable.
- Development Team: Comprises software engineers, AI specialists, and project managers who will build and maintain the chatbot. Regular meetings and iterative feedback sessions will be used for requirement elicitation.
- Legal and Compliance Officers: Ensure the chatbot adheres to data protection and privacy laws. Elicitation from this group will be through direct interviews to understand legal constraints and requirements.

1.3 Assumptions and Dependencies

Assumptions:

- Access to high-quality, diverse EI benchmark datasets is assumed for validating candidate responses.
- It is assumed that companies have a clear understanding of the EI requirements for their job postings.

Dependencies:

- The project's success is dependent on the availability and performance of LLMs like Google Gemini or Mistral 7b.
- Continuous access to updated EI benchmark datasets is a crucial dependency.

2 Functional Requirements

2.1 Job Description Submission

- **Primary Requirement:** The system shall allow HR departments to submit job descriptions at any time.
- **Event-driven Requirement:** Upon receiving a job description, the system shall generate a customized EI assessment test tailored to the specific requirements of the role.
- **Optional Feature Requirement:** The system may provide suggestions for enhancing the job description to better align with desirable EI traits.

2.2 EI Assessment Test Generation and Evaluation

- **Primary Requirement:** The chatbot shall generate EI assessment tests based on the submitted job descriptions, leveraging LLMs for nuanced understanding and question formulation.
- **Sate-driven Requirement:** When a candidate completes an EI assessment test, the system shall evaluate the responses against established EI benchmarks and generate a comprehensive report detailing the candidate's EI strengths and areas for improvement.
- **Unwanted Behavior Requirement:** The system shall not allow for any biases in test generation and evaluation, ensuring fairness and objectivity in assessing candidates' emotional intelligence.

2.3 Operating Environment

The chatbot should be accessible through any platform through web and has internet connectivity.

3 Non-functional Requirements

3.1 Performance

The chatbot should provide responses within a reasonable time frame to ensure a smooth user experience during candidate assessment.

3.2 Scalability

The chatbot should be capable of handling multiple concurrent users and job submissions without experiencing performance degradation.

3.3 Security

The chatbot should implement appropriate security measures to protect sensitive candidate and company data, ensuring confidentiality and integrity throughout the assessment process.

3.4 Usability

The chatbot interface should be intuitive and user-friendly, allowing stakeholders to easily submit job descriptions and interpret candidate responses without extensive training or technical knowledge.

3.5 Reliability

The chatbot should operate reliably without frequent interruptions or failures, ensuring continuous availability during critical assessment periods.

3.6 Compatibility

The chatbot should be compatible with a wide range of devices and operating systems, facilitating accessibility for stakeholders across various platforms.

3.7 Maintainability

The chatbot codebase should adhere to standard code style guidelines such as PEP 8, ensuring consistency and readability across the codebase. It should be thoroughly documented to facilitate understanding and modification by developers. Additionally, the code should utilize type hints where applicable to improve clarity and maintainability.

3.8 Legal & Ethical Considerations

The chatbot should comply with relevant laws and ethical guidelines regarding candidate assessment and data privacy, ensuring fair and unbiased evaluation throughout the hiring process.

4 Constraints

- **Software Constraints**: The chatbot requires access to advanced language models such as Google Gemini or Mistral 7b for nuanced understanding of responses. Additionally, an internet connection is necessary for accessing external datasets like the one provided on Kaggle.
- **Hardware Constraints**: The chatbot can be accessed from any electronic device capable of running Python applications and connecting to the internet.
- **User Constraints**: Users must be stakeholders such as companies or organizations seeking to assess candidate emotional intelligence in job applications.

5 Architecture Design

5.1 Overview

User Interface (UI):

- Google Form: Users fill out a Google Form to provide their information.
- Job Interview Session: HRs submit their job position description constraints and a choose the time slots they wish for the interviews to be conducted on.
- Chatbot Interface: After submitting the form, users interact with the chatbot interface for the interview session.

Chatbot System (Internal Logic):

- Form Data Processing: Upon form submission, the system extracts and processes user information from the Google Form.
- Interview Session Management: The chatbot formulates interview questions based on user data and conducts the interview session.
- Emotional Intelligence Testing: The chatbot analyzes user responses to assess emotional intelligence, possibly using predefined criteria or algorithms.
- Scoring System: Each user response is scored based on predefined criteria or algorithms.
- Report Generation: Generates a report summary of the interview session, including the final score and insights on emotional intelligence.

Data Access Layer:

- Database: Stores user information, interview responses, scores, and report summaries.

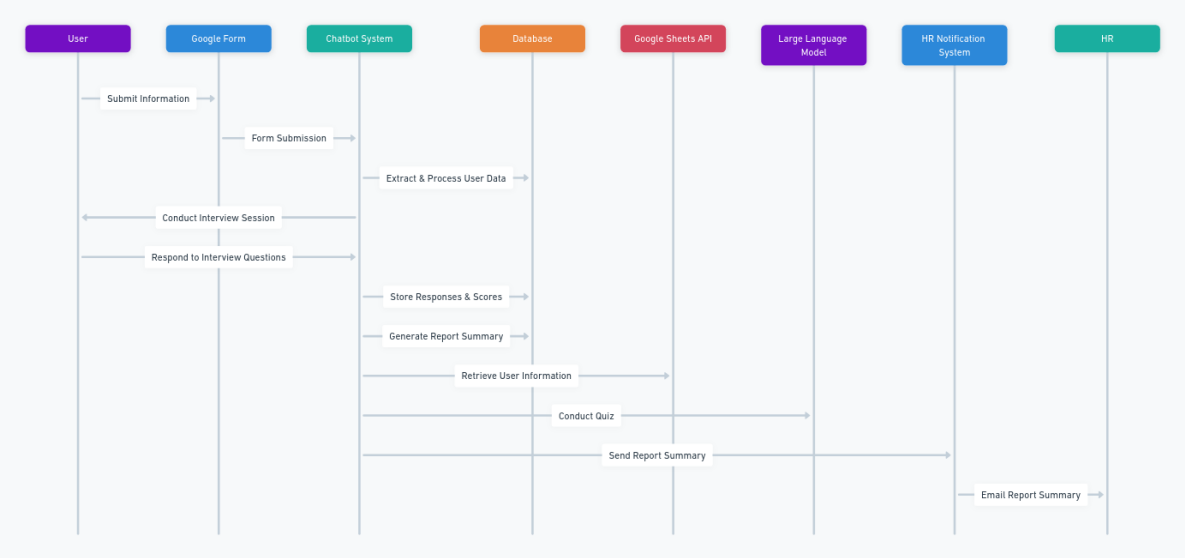
External Interfaces:

- Google Sheets API: Integrates with the Google Sheets API to retrieve user information from the Google Form response sheet.
- Large Language Model: Usage of a Large Language Model such as Gemini for conducting the quiz.
- HR Notification System: Sends report summaries to HR email addresses.

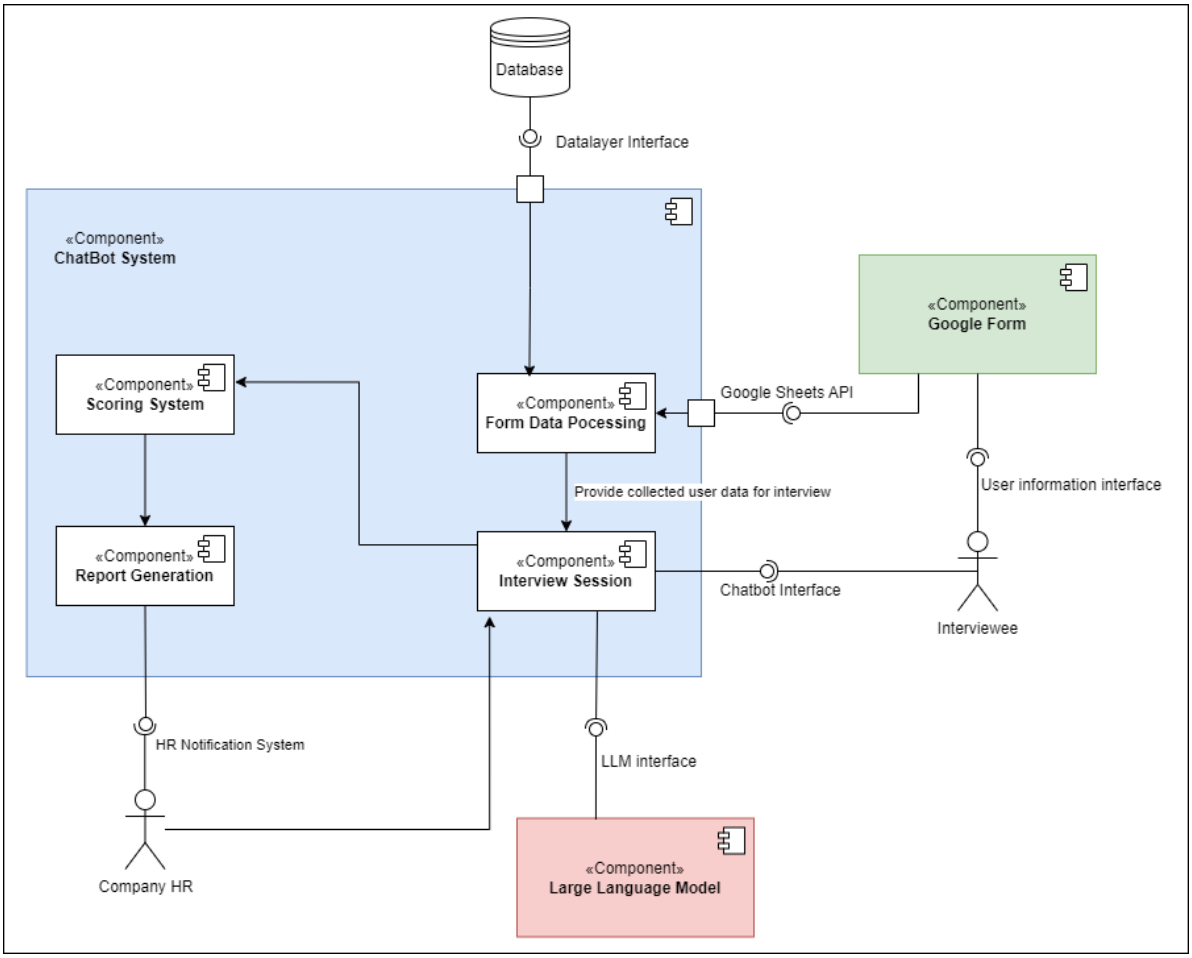
Interactions:

1. User fills out the Google Form with their information.
2. Upon form submission, the system processes the data and initiates an interview session with the chatbot.
3. The chatbot formulates interview questions based on the user's information and conducts the interview session.
4. User responds to interview questions, and the chatbot scores each response.
5. After the interview, the system generates a final score for the user and prepares a report summary.
6. The report summary is sent to HR email addresses for review and analysis.

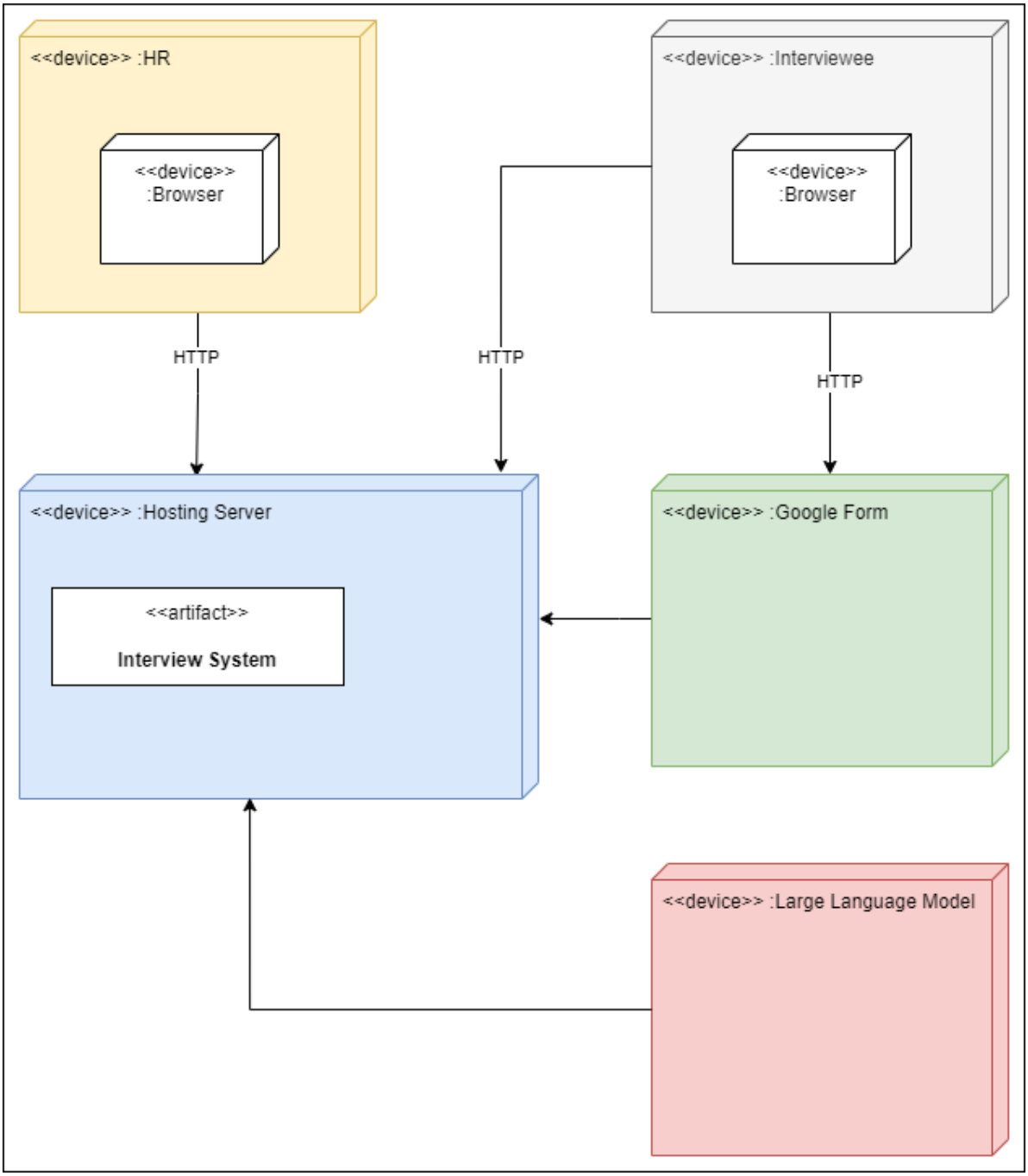
5.2 Sequence Diagram



5.3 Component Diagram



5.4 Deployment Diagram



6 Revision History

Version 1.0 (April 15, 2024)

Description: Initial release of the Software Requirements Specification (SRS) document.

Version 1.1 (April 20, 2024)

Description: Revised section 5.1, replacing the requirement for CV submission with the utilization of a Google Form for gathering user information.

Version 1.1.1 (April 24, 2024)

Description: Added the "Optional Feature Requirement" under section 2.1.

Version 1.2 (April 27, 2024)

Description: Updated section 1.3 regarding dependencies, transitioning from LLaMA to final choice of large language model.

Version 2.0 (May 5, 2024)

Description: Finalized document and released the official version.