Software Requirement Specification (SRS) Document for "MyFAANG" - A Mock Interview Web Application using GPT-3, Python, React, Flask, and GCP.

1. Introduction

1.1 Purpose of the Document

This document aims to define the requirements for MyFAANG, a web application that uses GPT-3 to create mock interviews with AI in real-time. This document will outline the functional and non-functional requirements, technical requirements, testing and validation strategies, and project management methodologies.

1.2 Overview of the Project

MyFAANG is a platform that offers job seekers the opportunity to practice and receive feedback on their interview skills. The platform generates realistic mock interviews based on industry and job role selection, then provides AI-powered feedback on user performance. Employers also have access to interview transcripts and performance metrics to aid in the hiring process.

1.3 Scope of the Project

The scope of MyFAANG includes the development of a web application that can generate realistic mock interviews, provide feedback, and store user data securely. The project will use Python, Flask, React, and GCP technologies to build a scalable and user-friendly platform.

1.4 Definitions, Acronyms, and Abbreviations

AI: Artificial Intelligence
API: Application Programming Interface
GCP: Google Cloud Platform
SRS: Software Requirement Specification

2. System Overview

2.1 System Context and Goals

MyFAANG aims to provide job seekers with a platform to practice and receive feedback on their interview skills. The system generates mock interviews and provides AI-powered feedback based on user performance.

2.2 Users of the System

The users of MyFAANG are job seekers and employers. Job seekers will use the platform to practice and improve their interview skills, while employers will have access to interview transcripts and performance metrics to aid in the hiring process.

2.3 Operating Environment

MyFAANG will be deployed on the Google Cloud Platform and accessible through web browsers. The platform will be optimized for desktop and mobile devices.

2.4 Constraints and Assumptions

- The platform will be developed using Python, Flask, React, and GCP technologies.
- The platform will integrate GPT-3 for generating interview questions and providing feedback.

- The platform will store user data securely and comply with data privacy regulations.
- The platform will be optimized for performance and scalability.

2.5 User Interface

The user interface of MyFAANG will be intuitive and user-friendly. The platform will have a simple and easy-to-navigate design, and will provide clear instructions for using the platform.

3. Functional Requirements

3.1 Mock Interview Generation

The MyFAANG web application shall generate a mock interview for the user based on the user's industry and job role selection. The mock interview shall include questions relevant to the selected industry and job role.

3.2 User Authentication and Registration

The MyFAANG web application shall allow users to create an account and authenticate themselves to access the application. The user registration process shall include the user's email, password, and basic information such as name and job role.

3.3 Industry and Job Role Selection

The MyFAANG web application shall allow users to select their industry and job role. The application shall provide a list of industries and job roles for the user to select from.

3.4 Interview Questions and Feedback Generation

The MyFAANG web application shall generate interview questions based on the user's industry and job role selection. The application shall provide feedback to the user on their performance in the mock interview.

3.5 User Performance Analysis

The MyFAANG web application shall analyze the user's performance in the mock interview and provide a report to the user. The report shall include the user's strengths and weaknesses, and areas for improvement.

3.6 Employer Access to Interview Transcripts and Performance Metrics

The MyFAANG web application shall provide employers with access to interview transcripts and performance metrics of the users. Employers shall have access only to the interviews of the users who have applied to their company.

4. Non-functional Requirements

4.1 Performance and Scalability

The MyFAANG web application shall be able to handle a large number of users simultaneously without any significant impact on its performance.

4.2 Security and Privacy

The MyFAANG web application shall ensure the security and privacy of the user's data. User passwords shall be encrypted, and user data shall be stored securely.

4.3 Usability and Accessibility

The MyFAANG web application shall be easy to use and accessible to all users. The application shall be designed with accessibility in mind, and it shall adhere to accessibility standards.

4.4 Compatibility with Different Web Browsers and Devices

The MyFAANG web application shall be compatible with different web browsers and devices. The application shall be responsive and accessible on desktop and mobile devices.

5. Technical Requirements

5.1 Technology Stack (Python, Flask, React, GCP)

The MyFAANG web application shall be developed using Python as the backend programming language, Flask as the web framework, React as the frontend library, and Google Cloud Platform (GCP) for hosting and deployment.

5.2 Dependencies and Third-Party Libraries

The MyFAANG web application shall use appropriate third-party libraries and dependencies, such as GPT-3 API for generating interview questions.

5.3 APIs and Data Models

The MyFAANG web application shall use appropriate APIs and data models to generate interview questions, store user data, and provide analytics.

5.4 Deployment and Hosting Requirements

The MyFAANG web application shall be deployed and hosted on the Google Cloud Platform (GCP) and shall use appropriate deployment and hosting configurations.

6. Testing and Validation

6.1 Testing Strategies and Methods

The MyFAANG web application shall be tested thoroughly using appropriate testing strategies and methods, such as unit testing, integration testing, and user acceptance testing.

6.2 Acceptance Criteria

The MyFAANG web application shall meet the acceptance criteria specified by the stakeholders, such as meeting the functional and non-functional requirements, being user-friendly, and providing accurate analytics.

6.3 Quality Assurance Processes

The MyFAANG web application shall follow appropriate quality assurance processes, such as code reviews, automated testing, and documentation.

7. Project Management

7.1 Roles and Responsibilities

The project team will consist of the following roles:

- Project Manager: responsible for overall project planning, scheduling, budgeting, and resource allocation
- Front-end Developer: responsible for developing the user interface using React
- Back-end Developer: responsible for developing the server-side logic using Flask and integrating with GCP services
- Data Scientist: responsible for implementing the machine learning algorithms using GPT-3 API and analyzing user performance data

- Quality Assurance (QA) Engineer: responsible for testing the application and ensuring its quality

7.2 Milestones and Timelines

The project timeline will be divided into the following milestones:

- Milestone 1 (Weeks 1-2): Project planning and requirement gathering
- Milestone 2 (Weeks 3-4): Front-end development and user authentication implementation
- Milestone 3 (Weeks 5-6): Back-end development and integration with GCP services
- Milestone 4 (Weeks 7-8): Machine learning algorithm implementation and performance analysis
- Milestone 5 (Weeks 9-10): Testing, quality assurance, and deployment
- The project will follow an Agile development methodology with weekly sprints to track progress.

7.3 Risk Management and Mitigation

The following risks have been identified for the project:

- Technical Risks: Potential technical issues may arise during the integration of different technologies. To mitigate this risk, the project team will conduct regular code reviews and collaborate closely to ensure smooth integration.
- Schedule Risks: Delays in one milestone may impact the overall project timeline. To mitigate this risk, the project manager will closely monitor progress and adjust schedules as necessary.
- Budget Risks: Unforeseen expenses may impact the project budget. To mitigate this risk, the project team will carefully manage expenses and adjust priorities as needed.
- Security Risks: Security vulnerabilities may arise during the development process. To mitigate this risk, the project team will conduct regular security assessments and implement appropriate measures to ensure data privacy and security.

7.4 Communication and Collaboration

Effective communication and collaboration among project team members are crucial for the success of the project. The project team will utilize communication tools such as email, instant messaging, and video conferencing to stay in touch and collaborate on tasks. The project manager will hold regular meetings to ensure that project goals are being met, milestones are being achieved, and any issues are being addressed promptly.

8. Conclusion

This Software Requirements Specification document outlines the functional and non-functional requirements, technical requirements, testing and validation strategies, and project management plan for the development of the MyFAANG web application. The project team is committed to delivering a high-quality, scalable, and secure application that meets the needs of its users.

9. Future Scope of the Project

Future enhancements to the MyFAANG web application may include:

- Integration with additional machine learning models for more personalized and comprehensive interview feedback
- Integration with social media platforms to allow for easy sharing of interview results and feedback
- Collaboration with job search platforms to provide users with more targeted job recommendations based on their interview performance

10. References and Acknowledgments

The project team would like to acknowledge the following resources and references used in the development of this Software Requirements Specification document:

- IEEE Recommended Practice for Software Requirements Specifications (IEEE Std 830-1998)
- Flask documentation (https://flask.palletsprojects.com/en/2.1.x/)
- React documentation (https://reactjs.org/)
- GPT-3 API documentation (https://beta.openai.com/docs/api-reference/introduction)
- Google Cloud Platform documentation (https://cloud.google.com/docs)