Career & Internship Portal

(Team Members)
Alingcomot, Liezel Anne P.
Fajarillo, Jaira Q.
Gonzales, Raymhar S.
Navera, Mervin M.
Repulles, Karyle Scott D.
Silapan, Francess Kyla V.

Software Requirements Specification

Document

Version: (n) **Date:(January/11/2025)**

Table of Contents

1. Introduction 5

1.1 Purpose *4*

1.2 Scope *4*

1.3 Definitions, Acronyms, and Abbreviations *5*

1.4 References *6*

1.5 Overview *7*

2. The Overall Description

2.1 Product Perspective *8*
2.1.1 System Interfaces *8*
2.1.2 Interfaces 9
2.1.3 Hardware Interfaces 9
2.1.4 Software Interfaces 9
2.1.5 Communications Interfaces 9
2.1.6 Memory Constraints 10
2.1.7 Operations 10
2.1.8 Site Adaptation Requirements 10

2.2 Product Functions *10*

2.3 User Characteristics *11*

2.4 Constraints *11*

2.5 Assumptions and Dependencies *13*

3. Specific Requirements 14

3.1 External interfaces *15*

3.2 Functions *16*

3.3 Performance Requirements *18*

3.4 Logical Database Requirements *19*

3.5 Design Constraints *21*
3.5.1 Standards Compliance 21

3.6 Software System Attributes *22*
3.6.1 Reliability 24
3.6.2 Availability 24
3.6.3 Security 25
3.6.4 Maintainability 25
3.6.5 Portability 26

3.7 External Actor Descriptions *29*
3.7.1 Human actors 29
3.7.2 Hardware actors 29
3.7.3 Software system actors 30

3.7.4 Use-Case Descriptions 58
3.7.4.1 Use case 1 58
3.7.4.2 use case n 59
3.8 Sequence Diagram *59*
3.8 Activity Diagram *60*
6. Supporting Information 61

1. Introduction

In the digital age, the pursuit of career advancement and skill development has become a critical priority for both job seekers and students. However, many existing platforms are limited in scope, focusing solely on job or internship listings without providing essential resources to streamline the job application process or enhance career readiness. The lack of integrated solutions often leaves users juggling multiple tools to prepare for interviews, build resumes, or gain the skills necessary for competitive opportunities.

To address these challenges, this study proposes the development of a "Career & Internship Portal"—a centralized, user-friendly platform designed to optimize the career development process. This system not only simplifies access to job and internship listings but also integrates essential tools such as resume builders, mock interview schedulers, and comprehensive career resources. By offering these functionalities in one cohesive platform, the portal aims to empower users at every stage of their professional journey.

1.1 Purpose

The purpose of "Career & Internship Portal" is to develop a centralized and comprehensive platform that integrates job and internship listings with career preparation resources. The system seeks to streamline the job search and application process, helping users enhance their career readiness. By offering features such as resume-building tools, mock interview scheduling, and task management, the portal addresses the current limitations of job platforms, which often lack integrated career development tools.

The platform is designed to cater to a diverse user base, including students and busy parents, providing tailored resources that align with their unique career goals and responsibilities.

1.2 Scope

The "Career & Internship Portal" is envisioned as a web-based and mobile-friendly application aimed at addressing the gaps in existing job boards and career platforms. Its scope includes:

• **Centralized Job and Internship Listings:** Users can easily search, filter, and apply for positions based on their preferences.
• **MResume-Building Tools:** Provides customizable templates and real-time editing capabilities to assist users in creating effective resumes.
• **Mock Interview Scheduling:** Enables users to book and practice interviews using an integrated calendar system with reminders.
• **Career Preparation Resources:** Offers access to tutorials, industry guides, and tips for skill-building and professional growth.
• **Task Scheduling and Management:** Helps users organize and prioritize career-related activities, such as application deadlines and interview schedules.

• **Accessibility:** Ensures a seamless user experience on both desktop and mobile platforms.

This system will serve as a one-stop career development tool for students and job seekers.

1.3 Definitions, Acronyms, and Abbreviations.

Definitions

1. **Career & Internship Portal:**

A comprehensive online platform designed to assist users in finding job and internship opportunities, building resumes, preparing for interviews, and accessing career development resources. It integrates job listings, career resources, and tools like resume builders and mock interview scheduling in one centralized system.

2. **User:**

An individual who interacts with the Career & Internship Portal. This can include students, parents, job seekers, or anyone using the portal to search for jobs, internships, or career development resources.

3. **Resume Builder:**

A tool within the portal that helps users create and format resumes, often using templates and step-by-step guidance to produce professional documents tailored to specific job applications.

4. **Mock Interview:**

A simulated interview session designed to help users practice their interview skills. It is integrated into the portal as a scheduling tool that allows users to book mock interview sessions with career coaches or automated systems.

5. **Job and Internship Listings:**

A section within the portal where available job and internship opportunities are displayed. Users can browse, filter, and apply to relevant positions.

6. **API (Application Programming Interface):**

A set of protocols and tools that allows different software systems to communicate with each other. In this project, APIs like Google Maps API and Google Calendar API will be integrated for location-based services and scheduling features.

7. **User Profile:**

A personalized space within the portal where users store their information, including contact details, resume, job preferences, and application history.

Acronyms

1. **SRS (Software Requirements Specification):**

A detailed document that defines the software's functionality, design, and behavior as required for development. It serves as a blueprint for developers, stakeholders, and project teams.

2. **UI (User Interface):**

The visual elements and layout through which users interact with the portal, including buttons, menus, forms, and other interactive components.

3. **UX (User Experience):**

The overall experience a user has when interacting with the portal, encompassing usability, design, accessibility, and the efficiency of interactions.

4. **DBMS (Database Management System):**

A system used to manage and store data, such as MySQL or PostgreSQL, used to store user profiles, job listings, applications, and other portal data.

5. **HTTP (HyperText Transfer Protocol):**

A protocol used for transmitting hypertext via the internet, essential for the functioning of web-based applications, including the Career & Internship Portal.

6. **SSL (Secure Sockets Layer):**

A security protocol used to establish an encrypted link between the user's browser and the portal, ensuring that sensitive data (such as resumes and application information) is transmitted securely.

7. **API (Application Programming Interface):**

A set of protocols that enables the integration of external tools and services (e.g., Google Maps, Google Calendar) with the portal for enhanced functionality.

Abbreviations

1. **EC2 (Elastic Compute Cloud):**

An Amazon Web Services (AWS) product used to host web applications, providing scalable computing resources for the portal.

2. **S3 (Simple Storage Service):**

An AWS service for storing and managing user data, such as resumes, documents, and media files within the Career & Internship Portal.

3. **RDS (Relational Database Service):**

An AWS-managed service for setting up, operating, and scaling a relational database in the cloud. It will store the Career & Internship Portal's data, including job listings, user profiles, and applications.

4. **HTML (HyperText Markup Language):**

The standard markup language used to create the structure of web pages, such as the Career & Internship Portal's website.

5. **CSS (Cascading Style Sheets):**

A language used to style the layout and appearance of web pages. It will be used to create a responsive, visually appealing interface for the portal.

6. **JavaScript (JS):**

A programming language that enables interactivity on web pages. It will be used to add dynamic functionality to the Career & Internship Portal, including form handling, real-time content updates, and interactive features.

7. **PDF (Portable Document Format):**

A widely-used document format that will allow users to download their resumes in a consistent and secure format through the portal.

1.4 References

1. **Career Development Tools and Strategies**

• Heintz, J., & Weber, R. (2019). Effective Career Preparation: Building Skills for Workplace Success. New York: Career Strategies Press.

- National Association of Colleges and Employers (NACE). (2022). Career Readiness Competencies. Available at: www.naceweb.org

2. Web and Mobile Application Development

- Flanagan, D. (2020). JavaScript: The Definitive Guide. O'Reilly Media.
- Duckett, J. (2014). HTML and CSS: Design and Build Websites. Wiley Publishing.
- Google Developers (2023). Google Maps and Calendar APIs Documentation. Available at: https://developers.google.com/

3. System Design and Database Management

- Connolly, T., & Begg, C. (2020). Database Systems: A Practical Approach to Design, Implementation, and Management. Pearson Education.
- AWS Documentation (2024). Amazon Web Services Cloud Hosting Solutions. Available at: https://aws.amazon.com/documentation/

4. User-Centered Design and Usability

- Krug, S. (2014). Don't Make Me Think: A Common Sense Approach to Web Usability. New Riders.
- Nielsen, J. (2012). Usability Engineering. Academic Press.

5. Security Standards and Practices

- Stallings, W. (2019). Cryptography and Network Security: Principles and Practice. Pearson Education.
- Open Web Application Security Project (OWASP). (2023). OWASP Top Ten Web Application Security Risks. Available at: https://owasp.org/

6. Project Management and Software Development Lifecycle (SDLC)

- Sommerville, I. (2015). Software Engineering. Pearson.
- Royce, W. W. (1970). Managing the Development of Large Software Systems. Proceedings of IEEE WESCON.

7. Career Services and Job Portals

- LinkedIn Talent Solutions (2023). Features of LinkedIn for Job Seekers and Employers. Available at: https://business.linkedin.com/
- Indeed Help Center (2023). Using Job Boards for Career Growth. Available at: https://www.indeed.com/help/

8. Accessibility and Inclusivity Standards

- W3C. (2018). Web Content Accessibility Guidelines (WCAG) 2.1. Available at: https://www.w3.org/WAI/standards-guidelines/wcag/

9. Relevant Surveys and Case Studies

- Pew Research Center. (2022). Workplace Trends and Employment Challenges. Available at: https://www.pewresearch.org/
- Case Study: CareerOneStop (2023). Integrating Career Services into Digital Platforms. CareerOneStop.org.

1.5 Overview

The portal is designed to bridge the gap between job seekers and career development resources. Unlike traditional job boards, the "Career & Internship Portal" integrates career preparation tools into a single platform. It offers:

1. Ease of Use: A user-friendly interface for both web and mobile devices.
2. Enhanced Career Readiness: By providing tools for resume creation and mock interviews, the portal equips users with the skills needed for success in the competitive job market.
3. Streamlined Process: Simplifies job and internship searches by providing relevant listings, tutorials, and integrated scheduling tools.

The system aims to empower users to take control of their career journeys by offering a comprehensive and efficient solution to their needs.

2. The Overall Description

The Career & Internship Portal is a centralized platform designed to streamline the career development process for students, job seekers, and professionals. It offers a comprehensive suite of features, including job and internship listings, a resume-building tool, mock interview scheduling, and access to career resources such as industry guides and tutorials. The portal caters to diverse user needs by simplifying job searches, enhancing career readiness, and supporting skill development through an intuitive and user-friendly interface. Accessible via web or mobile application, the system ensures flexibility and convenience, integrating essential tools to help users confidently navigate the competitive job market. By addressing common challenges such as fragmented resources and inefficient application processes, the platform empowers users to achieve their career goals effectively.

2.1 Product Perspective

The "Career & Internship Portal" is a self-contained platform designed to streamline career development by integrating job and internship listings, resume-building tools, mock interview scheduling, and career resources into one centralized system. Unlike traditional job boards, which primarily focus on job postings, this portal includes features to enhance user preparedness and increase employability.

The system will function as an independent web-based platform but can also integrate with external systems such as job listing APIs, career service tools, and external scheduling software. Comparatively, the system differs from existing platforms like LinkedIn or Indeed by prioritizing a comprehensive, user-focused approach to career preparation.

2.1.1 System Interfaces

The system interfaces with several external services and tools to deliver its functionalities effectively. Key system interfaces include:

- **Google Calendar API**: Enables users to schedule and track career events, such as application deadlines and mock interviews.
- **Google Maps API**: Provides location-based services to help users navigate to interviews and networking events.

- **Authentication APIs** (e.g., OpenID, Google Sign-In): Ensure secure user login and data access.
- **Job Listing APIs**: Allow the system to fetch and display job and internship opportunities from external sources.

These interfaces ensure that external systems and services seamlessly contribute to the platform's features.

2.1.2 Interfaces

The system employs a **Graphical User Interface (GUI)** to interact with users. The interface is designed to be intuitive and user-friendly, ensuring accessibility for all user demographics, including students and busy parents. Key features of the user interface include:

- Responsive design for compatibility across desktops, tablets, and smartphones.
- ADA-compliant features to ensure usability for users with disabilities.
- Interactive dashboards for easy navigation of tools, such as the resume builder and job application tracker.

By optimizing the interface for simplicity and accessibility, the platform caters to diverse user needs.

2.1.3 Hardware Interfaces

The system has no specific hardware interface requirements beyond access to standard internet-enabled devices. These include:

- Personal computers, tablets, or smartphones for accessing the portal.
- Network connections for communication with external APIs and services.

This minimal dependency on specialized hardware ensures ease of use and broad compatibility.

2.1.4 Software Interfaces

The system interacts with various external software to enhance functionality, including:

- **Google Maps API**: Used for location-based features like navigation to interviews.
- **Google Calendar API**: Synchronizes career events and deadlines.
- **jsPDF**: Enables users to export resumes in PDF format.
- **Database Systems**: MySQL or PostgreSQL will manage user profiles, application histories, and system data.

The choice of these software components ensures scalability, interoperability, and ease of implementation for developers.

2.1.5 Communications Interfaces

The system utilizes secure **HTTPS protocols** for all user interactions, ensuring encrypted communication between the client and the server. Additionally, RESTful APIs facilitate communication with external services, such as:

- Authentication (OpenID or Google Sign-In).
- Job Listing APIs for fetching data from external job boards.
- Scheduling APIs for managing interview and application events.

These communication interfaces maintain system reliability and user data privacy.

2.1.6 Memory Constraints

The system relies on cloud-based storage solutions for scalability and efficiency. Memory constraints include:

- Support for storing large amounts of user-generated data, such as resumes, application records, and user profiles.
- Optimization to ensure quick data retrieval, even as user data volume increases.
  There are no strict memory constraints for primary or secondary memory on user devices.

2.1.7 Operations

The platform supports the following operations to ensure a seamless user experience:

- **Interactive Operations**: Users can perform tasks such as job searching, resume building, and scheduling mock interviews.
- **Automated Notifications**: Reminders for deadlines, interviews, and career-related tasks will be generated.
- **Backup and Recovery**: Cloud-hosted databases will feature automatic backups to ensure data reliability and recovery in case of system failures.

These operations enhance system reliability and ensure uninterrupted user access to essential tools.

2.1.8 Site Adaptation Requirements

● **Deployment:** The system must be deployable on cloud platforms or university-hosted servers to ensure reliability and accessibility.
● **Customization:** Adaptable to different institutional or organizational requirements.

2.2 Product Functions

Job and Internship Listings
●Users can browse and apply for job or internship opportunities.
●Listings can be filtered by location, industry, and user preferences.

Resume-Building Tools
●Provides pre-designed templates and editing tools for creating professional resumes.
●Allows users to download resumes in PDF format using libraries like jsPDF.

Mock Interview Scheduling
●Integration with Google Calendar for booking interview practice sessions.
●Users receive automated reminders and feedback.

Career Preparation Resources
●Access to tutorials, tips, and industry-specific guides.
●Provides insights on how to improve job application success rates.

Task Management
●Scheduling tools to organize job applications, interviews, and deadlines.
●Tracks missed events and offers feedback for improved time management.

Accessibility
●A mobile-responsive interface that ensures usability on various devices.

2.3 User Characteristics

Students

- **Profile:** Typically balancing academic schedules with career preparation.
- **Needs:** Simplified tools for finding internships, building resumes, and practicing interview skills.

Busy Parents

- **Profile:** Working or stay-at-home parents seeking flexible employment.
- **Needs:** Resources for updating resumes and searching for jobs that fit their schedules.

2.4 Constraints

1. Regulatory Policies

- The system must comply with the **Philippine Data Privacy Act of 2012 (Republic Act No. 10173)**, including:
  - **Fair Information Processing Principles** to ensure lawful, fair, and transparent data collection and processing.
  - **Data Subject Rights** for users to access, rectify, and delete their personal data.
  - **Data Security** protocols to protect user data from unauthorized access and breaches.
  - **Breach Notification Procedures** to notify the National Privacy Commission (NPC) and affected users in case of a data breach.
- Accessibility standards must adhere to **Web Content Accessibility Guidelines (WCAG)** to support users with disabilities, including:

    - Alternative text for images.
    - Sufficient color contrast.
    - Keyboard navigation and assistive technology compatibility.

2. Hardware Limitations

- The system is designed to function on standard internet-enabled devices such as personal computers, laptops, smartphones, and tablets.
- No specialized hardware is required, ensuring compatibility with commonly used devices.
- The system relies on stable network connections for real-time features like job searches, calendar integration, and notifications.

3. Interface to Other Applications

- The system integrates with external tools and APIs to enhance functionality, including:
    - **Google Calendar API** for scheduling and reminders.
    - **Google Maps API** for location-based navigation.
    - **Authentication APIs** (e.g., Google Sign-In, OpenID) for secure user login.
    - **Job Listing APIs** for retrieving job and internship data from external sources.
- These interfaces rely on **RESTful APIs** over HTTPS for secure and reliable communication.

4. Parallel Operation

- The system must support simultaneous usage by multiple users, including operations like job searching, resume building, and mock interview scheduling.
- It must maintain optimal performance even during peak usage periods, with efficient resource management to avoid bottlenecks.

5. Audit Functions

- A robust logging system must track user actions, system activities, and changes to data for audit and troubleshooting purposes.
- Logs must be securely stored and accessible only to authorized personnel to ensure compliance and maintain system integrity.

6. Control Functions

- Portal administrators must have access to control functions, including:
    - User account management.
    - Oversight of job postings and career resources.
    - Monitoring and managing system health and performance.
- Administrative controls must be restricted to authorized users to ensure secure operations.

7. Higher-Order Language Requirements

- The system should be developed using a standard, portable programming language such as **Python** or **Java**, ensuring compatibility and maintainability.
- Code must adhere to consistent standards (e.g., **PEP 8** for Python) and follow modular architecture for ease of updates and scalability.

8. Signal Handshake Protocols

- While the system does not explicitly depend on low-level signal handshake protocols like **XON-XOFF** or **ACK-NACK**, reliable data transmission mechanisms must be implemented for interactions between the client and server.

9. Reliability Requirements

- The system is critical for career development and must maintain an **uptime of 99.9%**.
- Automated failover mechanisms should redirect users to backup servers during outages.
- A disaster recovery plan must ensure:
    - **Recovery Time Objective (RTO):** 4 hours.
    - **Recovery Point Objective (RPO):** 12 hours.
- Regular backups and system monitoring must be in place to minimize downtime and data loss.

10. Criticality of the Application

- As a platform central to users' career growth, any failure or downtime could significantly impact user trust.
- Notifications for scheduled maintenance or unexpected outages must be communicated promptly to users.

11. Safety and Security Considerations

- The system must prioritize data protection through:
    - **Strong encryption standards** (e.g., AES-256) for securing sensitive data.
    - **Secure authentication mechanisms**, such as multi-factor authentication (MFA).
- Regular **security audits** and penetration tests must be conducted to identify and resolve vulnerabilities.
- User sessions should have inactivity timeouts to reduce unauthorized access risks.

2.5 Assumptions and Dependencies

Assumptions

- The system assumes that users will have access to a stable internet connection to utilize real-time features such as job searches, calendar integration, and notifications.

- The majority of users will access the system through commonly used devices such as personal computers, laptops, smartphones, or tablets.
- External APIs (e.g., Google Calendar, Google Maps, job listing APIs) will remain available and functional throughout the system's lifecycle.
- Users will provide accurate and truthful information during account creation and profile updates.
- The system's hosting infrastructure will support scalability to handle an increasing number of users and concurrent operations.
- Security measures implemented in the system will remain effective against foreseeable cyber threats.

Dependencies

- **APIs and Third-Party Services**: The system depends on external APIs for key functionalities, including:
    - **Google Calendar API** for scheduling and reminders.
    - **Google Maps API** for navigation and location-based services.
    - **Job Listing APIs** for fetching job and internship data from external sources.
    - **Authentication Services** (e.g., Google Sign-In, OpenID) for secure user login.
- **Cloud Infrastructure**: The system relies on cloud services (e.g., AWS S3) for data storage, backups, and recovery.
- **Regulatory Compliance**: The system depends on adherence to laws such as the Philippine Data Privacy Act and other relevant data protection regulations.
- **User Devices**: The system is dependent on the availability of modern internet-enabled devices with updated browsers for optimal performance.
- **Hosting Environment**: The system's availability and uptime depend on reliable hosting infrastructure capable of 24/7 operation with minimal downtime.
- **Development Tools and Frameworks**: The system relies on chosen tools, programming languages (e.g., Python, Java), and frameworks to meet performance and compatibility requirements.

3. Specific Requirements

Software Requirements

1. **Frontend Technologies:**
- HTML, CSS, JavaScript, and React for responsive design.
2. **Backend Technologies:**
- Node.js or Django for server-side operations.
- MySQL or PostgreSQL for database management.
3. **APIs and Integrations:**
- Google Maps API for location-based job recommendations.
- Google Calendar API for mock interview scheduling.

Database Requirements

1. Store user profiles, job listings, resume data, and mock interview schedules.
2. Implement indexing for faster searches.

Development and Deployment Tools:

1. Version control via GitHub or GitLab.
2. Continuous integration and deployment pipelines using Jenkins or GitHub Actions.

Special Requirements

1. Mobile Application:
   - Develop an Android app with the same functionality as the web version.
   - Ensure compatibility with Android 8.0 and above.
2. Analytics Dashboard:
   - Provide admins with data on user activities, popular listings, and resource usage.
   - Generate monthly reports for strategic planning.
3. Accessibility:
   - Ensure compliance with WCAG 2.1 standards for accessibility, such as screen reader support and keyboard navigation.
4. Error Handling:
   - Provide clear error messages and fallback mechanisms for system failures.

3.1 External Interfaces

This section provides a detailed description of all inputs and outputs of the "Career & Internship Portal" software system, focusing on technical aspects relevant to the development team.

Name of item:

- Identifies each data element or interface (e.g., "Job Listings API," "User Profile Data," "Payment Gateway").

Description of purpose:

- Briefly explains the function or role of each interface.
  - Example: "Job Listings: To provide access to job postings from various sources."

Source of input or destination of output:
Specifies the origin or destination of data for each interface.

- Example: "Job Listings: External job boards (e.g., Indeed API), company websites, user submissions."

Valid range, accuracy and/or tolerance:

- Defines acceptable limits and expected accuracy for each data type.

- Example: "User Profiles: Valid email addresses, phone numbers, accurate education and employment history."

Units of measure:

- Specifies units of measurement where applicable (e.g., file size in KB/MB).

Timing:

- Defines the expected timing of data exchange (e.g., real-time, batch updates).

Relationships to other inputs/outputs:

- Describes how different interfaces interact and influence each other.
  - Example: "User Profiles are used to personalize the user experience and for targeted job recommendations."

Screen formats/organization:

- Specifies the desired presentation of data within the user interface.
  - Example: "User-friendly layouts for job/internship listings, easy navigation, clear calls to action."

Window formats/organization:

- Defines how data will be presented across different devices (e.g., responsive design).

Data formats:

- Specifies the expected data formats for input and output (e.g., JSON, XML, CSV).

Command formats:

- Defines the communication protocols and commands used for data exchange (e.g., RESTful APIs).

End messages:

- Specifies the types of messages returned to the system after an interface interaction (e.g., success/failure messages, error codes).

3.2 Functions

This section defines the fundamental actions that the "Career & Internship Portal" software system must perform. This maybe changed as the system is developed.

1. Job/Internship Listing Management

- **The system shall** retrieve job and internship listings from external sources (e.g., APIs, web scraping).
- **The system shall** validate the accuracy and completeness of retrieved job and internship data.
- **The system shall** store job and internship listings in the system database.
- **The system shall** allow users to search for job and internship listings based on various criteria (e.g., keywords, location, company, industry).
- **The system shall** filter search results based on user preferences (e.g., job type, experience level, salary range).
- **The system shall** display relevant job and internship listings to users.
- **The system shall** generate job and internship alerts based on user preferences.

2. User Profile Management

- **The system shall** allow users to create and manage their profiles.
- **The system shall** validate user input for accuracy and completeness (e.g., email address, phone number).
- **The system shall** store user profile data securely in the system database.
- **The system shall** allow users to update their profile information at any time.
- **The system shall** ensure the confidentiality and security of user profile data.

3. Resume Builder Functionality

- **The system shall** provide a user-friendly interface for resume creation.
- **The system shall** offer a selection of resume templates for users to choose from.
- **The system shall** allow users to import existing resumes from their local devices.
- **The system shall** enable users to edit and format resume content within the system.
- **The system shall** allow users to download their resumes in various formats (e.g., PDF, DOCX).
- **The system shall** generate cover letters based on resume data and job/internship descriptions.

4. Mock Interview Scheduling

- **The system shall** allow users to schedule mock interviews with mentors or other qualified individuals.
- **The system shall** validate and confirm interview availability with both the user and the interviewer.
- **The system shall** send automated reminders to both the user and the interviewer prior to the scheduled interview time.
- **The system shall** facilitate the exchange of feedback between the user and the interviewer after the interview.

5. Career Resources Management

- **The system shall** store and retrieve career resources (articles, guides, videos) from various sources.

- **The system shall** categorize and organize career resources for easy user access.
- **The system shall** allow users to search and filter career resources based on their interests and career goals.
- **The system shall** display career resources in a user-friendly and engaging format.

6. User Messaging and Communication

- **The system shall** enable secure and reliable communication between users and support staff.
- **The system shall** allow users to send and receive messages within the portal.
- **The system shall** facilitate the resolution of user inquiries and support requests.

7. Payment Processing (if applicable)

- **The system shall** integrate with a secure payment gateway (e.g., Stripe, PayPal).
- **The system shall** process payments for premium features (if applicable) accurately and securely.
- **The system shall** generate and store transaction records for all payments.

8. Data Security and Privacy

- **The system shall** implement robust security measures to protect user data from unauthorized access and breaches.
- **The system shall** comply with all relevant data privacy regulations (e.g., GDPR, Philippine Data Privacy Act).

9. Error Handling and Recovery

- **The system shall** detect and handle system errors gracefully.
- **The system shall** display informative error messages to users.
- **The system shall** implement mechanisms for system recovery in case of failures.

3.3 Performance Requirements

Given that the project is in the planning stage, these requirements should be met for the system.

Job Search Response Time: Job search results should be displayed within 5 seconds of the user submitting a query.
Resume Builder Load Time: The resume builder interface should load within 2 seconds.
Mock Interview Scheduling Response Time: Mock interview scheduling requests should be processed and confirmed within 3 seconds.
User Profile Update Processing Time: User profile updates should be processed and reflected on the screen within 3 seconds.

3.4 Logical Database Requirements

This section outlines the logical database requirements for the "Career & Internship Portal" using a SQL database schema.

3.4.1 Tables

- **Users**
    - user_id INT PRIMARY KEY AUTO_INCREMENT,
    - first_name VARCHAR(255) NOT NULL,
    - last_name VARCHAR(255) NOT NULL,
    - email VARCHAR(255) UNIQUE NOT NULL,
    -
    - password_hash VARCHAR(255) NOT NULL,
    -
    - date_of_birth DATE,
    - phone_number VARCHAR(20),
    - address TEXT,
    - education TEXT,
    - work_experience TEXT,
    - skills TEXT,
    - career_goals TEXT,
    - is_active BOOLEAN DEFAULT TRUE,
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
    -
- **Jobs**
    - job_id INT PRIMARY KEY AUTO_INCREMENT,
    - title VARCHAR(255) NOT NULL,
    - company_name VARCHAR(255),
    - description TEXT,
    - location VARCHAR(255),
    - job_type ENUM('Full-time', 'Part-time', 'Contract', 'Internship'),
    - required_skills TEXT,
    - salary_range DECIMAL(10, 2),
    - application_deadline DATE,
    - company_website TEXT,
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
    -
- **Internships**
    - internship_id INT PRIMARY KEY AUTO_INCREMENT,

    - title VARCHAR(255) NOT NULL,
    - company_name VARCHAR(255),
    - description TEXT,
    - location VARCHAR(255),
    - start_date DATE,
    - end_date DATE,
    - application_deadline DATE,
    - required_skills TEXT,
    - department VARCHAR(255),
    - supervisor VARCHAR(255),
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
    -
- **Resumes**

    - resume_id INT PRIMARY KEY AUTO_INCREMENT,
    - user_id INT,
    - file_path VARCHAR(255),
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
    - FOREIGN KEY (user_id) REFERENCES Users(user_id)
    -
- **CareerResources**

    - resource_id INT PRIMARY KEY AUTO_INCREMENT,
    - title VARCHAR(255),
    - resource_type ENUM('Article', 'Guide', 'Video'),
    - url TEXT,
    - category VARCHAR(255),
    - keywords TEXT,
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
- **MockInterviews**

    - interview_id INT PRIMARY KEY AUTO_INCREMENT,
    - user_id INT,
    - interviewer_id INT,
    - scheduled_time TIMESTAMP,
    - interviewer_feedback TEXT,
    - user_feedback TEXT,
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - FOREIGN KEY (user_id) REFERENCES Users(user_id),
    - FOREIGN KEY (interviewer_id) REFERENCES Users(user_id)
- **Messages**

- message_id INT PRIMARY KEY AUTO_INCREMENT,
- sender_id INT,
- recipient_id INT,
- message TEXT,
- created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
- is_read BOOLEAN DEFAULT FALSE,
- FOREIGN KEY (sender_id) REFERENCES Users(user_id),
- FOREIGN KEY (recipient_id) REFERENCES Users(user_id)

3.5 Design Constraints

This section outlines the constraints that may impact the design and development of the "Career & Internship Portal".

3.5.1 Standards Compliance

- **Data Privacy:** The system must comply with all relevant data privacy regulations, such as the **Philippine Data Privacy Act of 2012 (Republic Act No. 10173)**. This includes:
  - **Fair Information Processing Principles:** Ensuring data collection and processing are lawful, fair, and transparent.
  - **Data Subject Rights:** Implementing mechanisms for users to access, rectify, and delete their data.
  - **Data Security:** Implementing appropriate security measures to protect user data from unauthorized access, use, disclosure, alteration, or destruction.
  - **Data Breach Notification:** Establishing procedures for notifying the National Privacy Commission (NPC) and affected users in case of a data breach.
- **Accessibility:** The system must adhere to accessibility standards such as the **Web Content Accessibility Guidelines (WCAG)** to ensure usability for individuals with disabilities. This includes:
  - Providing alternative text for images.
  - Ensuring sufficient color contrast.
  - Making the interface navigable using only a keyboard.
  - Supporting assistive technologies (screen readers, screen magnifiers).
- **Usability:** The system must prioritize user experience and usability by:
  - Following established usability principles (e.g., consistency, simplicity, learnability).
  - Conducting user testing and incorporating feedback into the design.
  - Ensuring the system is easy to navigate and use for all target users.
- **Security:**

- Implement robust security measures to protect the system from cyber threats such as hacking, malware, and data breaches.
- Use strong encryption for sensitive data (e.g., passwords).
- Regularly conduct security audits and penetration testing.

3.6 Software System Attributes

Software System Attributes for the Career & Internship Portal

The software system attributes define the quality characteristics of the platform, ensuring it meets the expectations for performance, usability, and reliability. Below are the key attributes:

1. Performance

- The system must handle 1,000 concurrent users without performance degradation.
- Job search queries should return results within 3 seconds.
- Resume generation should complete within 5 seconds of input submission.
- The portal should support high-speed data access, even with large-scale data (e.g., 1,000+ job listings).

2. Scalability

- The platform must be designed to scale horizontally to handle an increasing user base and growing database size.
- Capable of supporting additional features or integrations without major system redesigns.
- Cloud-based infrastructure will allow dynamic scaling of resources based on demand.

3. Usability

- The system must have a user-friendly interface that is intuitive for both technical and non-technical users.
- Responsive design ensures usability across devices (desktop, tablets, smartphones).
- Include tooltips, tutorials, and an easy-to-navigate menu for better user experience.

4. Security

- User data (e.g., resumes, account information) must be encrypted using AES-256 encryption.
- Secure authentication via OAuth 2.0 for Google integration.
- Protect against common vulnerabilities such as SQL injection, cross-site scripting (XSS), and CSRF attacks.
- Implement role-based access control (RBAC) to restrict unauthorized actions.
- Regular security audits and penetration testing to identify and address vulnerabilities.

5. Reliability

- Ensure 95% uptime through the use of load balancers, redundant servers, and robust cloud hosting.
- Provide automated data backups to prevent loss in case of system failures.
- Incorporate failover mechanisms for uninterrupted service during server issues.

6. Maintainability

- Modular design to allow easy updates or bug fixes without affecting the entire system.
- Clean and well-documented codebase to simplify maintenance.
- Use version control systems like GitHub for code management and change tracking.
- Support continuous integration and deployment (CI/CD) pipelines for streamlined updates.

7. Accessibility

- Compliant with WCAG 2.1 standards for users with disabilities.
- Provide screen reader support for visually impaired users.
- Ensure all buttons and links are accessible via keyboard navigation.
- High-contrast mode for users with visual impairments.

8. Portability

- The web version should work seamlessly on modern browsers (Chrome, Firefox, Safari, Edge).
- The mobile app should be compatible with Android 8.0+ and iOS 12+.
- Ensure easy deployment on different hosting platforms (e.g., AWS, on-premises servers).

9. Interoperability

- Seamlessly integrate with third-party APIs such as:
- Google Maps API for location-based job recommendations.
- Google Calendar API for scheduling mock interviews.
- Provide standard data exchange formats (e.g., JSON, XML) for compatibility with external systems.

10. Availability

- The platform should provide 24/7 availability with robust monitoring tools like Amazon CloudWatch or New Relic.
- Scheduled maintenance periods should be communicated to users in advance.
- Incorporate disaster recovery mechanisms to restore services within 30 minutes of a failure.

11. Adaptability

- The platform must be designed to easily incorporate new features (e.g., AI-powered career coaching).

- Allow localization for multi-language support based on user region.
- Flexible enough to adapt to industry trends or user demands.

These attributes ensure the Career & Internship Portal is efficient, reliable, secure, and scalable while meeting the diverse needs of its users.

3.6.1 Reliability

Minimized downtime through the use of scalable cloud services.

3.6.2 Availability

System Availability Requirements

1. ***Availability Target****:*
    - *The system shall maintain an uptime of 99.9% during its operational hours (24/7 availability) to ensure consistent access for users worldwide.*
    - *Scheduled maintenance windows must be minimized and communicated to users at least 72 hours in advance.*

2. Failure Recovery Requirements

- *In the event of a failure, the system shall recover and resume normal operation within **15 minutes**.*
- *The system shall implement an automatic failover mechanism to redirect users to a backup server or data center in the event of a primary server failure.*

3. Data Recovery and Input Loss

- *The system shall ensure that:*
    - *User input loss during failures is minimized to **at most 10 characters of input** or equivalent actions within a session.*
    - *For incomplete form submissions (e.g., job applications or resume edits), the system shall auto-save drafts every **30 seconds** to reduce data loss.*

4. Checkpoint Mechanism

- *The system shall implement **transactional checkpoints** to save user progress at key interaction points, such as:*
    - *After submitting a job application.*
    - *After saving or updating a resume.*
    - *After scheduling a mock interview.*
- *These checkpoints ensure data consistency and facilitate smooth recovery after failures.*

5. Restart Capabilities

- *Upon restart after a failure, the system shall:*
    - *Restore the user's last active session state (if applicable) using session persistence mechanisms.*
    - *Notify the user of the recovery process and any data that might have been impacted.*
    - *Provide users with a clear error message and steps to re-engage with the system, ensuring transparency and trust.*

6. Redundancy and Scalability

- *The system shall use a **load-balanced, distributed architecture** to handle high traffic and prevent single points of failure.*
- *All critical components, including databases, application servers, and network services, shall have redundancy to ensure uninterrupted service.*

7. Monitoring and Alerts

- *A real-time monitoring system shall:*
    - *Detect failures immediately and notify administrators within **5 minutes**.*
    - *Generate automated alerts for potential performance issues, enabling proactive measures to prevent downtime.*

8. Disaster Recovery

- *The system shall have a documented **disaster recovery plan** with the following features:*
    - *Offsite backups updated every **12 hours** to ensure minimal data loss during major outages.*
    - *Recovery Time Objective (RTO): **4 hours**.*
    - *Recovery Point Objective (RPO): **12 hours**.*

3.6.3 Security

Data encryption and secure authentication mechanisms to protect user information.

3.6.4 Maintainability

Modularity:

- Modular architecture with logically separated components.
- Well-defined interfaces for minimal interdependencies.

Documentation:

- Comprehensive and version-controlled documentation, including design, API references, and maintenance guides.

- Code comments explaining non-obvious logic and decisions.

Code Quality and Standards:

- Adherence to consistent coding standards (e.g., PEP 8 for Python).
- Code review process and use of static analysis tools.

Testing and Validation:

- Unit test coverage of at least 80%.
- Integration and regression testing for system stability.
- Automated testing pipelines integrated with CI/CD.

Complexity Management:

- Avoid complex algorithms; use established design patterns.
- Cyclomatic complexity limited to a score of 10.
- Scheduled refactoring to reduce technical debt.

Interfaces:

- Standards-compliant internal and external interfaces.
- Versioned APIs for backward compatibility.

Configuration and Customization:

- Externalized and editable configuration settings.
- Support for modular upgrades with minimal system impact.

Dependency Management:

- Use dependency management tools (e.g., pip, npm).
- Regular updates to dependencies for security and compatibility.

Logging and Monitoring:

- Comprehensive logging for debugging and maintenance.
- Centralized logs and real-time monitoring dashboards.

Maintenance Handoff:

- Formal handoff meeting for knowledge transfer.
- Checklist of routine maintenance tasks for smooth transitions.

3.6.5 Portability

1. **Percentage of Components with Host-Dependent Code**

- **Rationale**: Minimizing host-dependent components reduces the effort required to adapt the software to different environments.
- **Testing/Measurement**:
  - Identify and quantify components that rely on platform-specific APIs or features.
  - Target: Host-dependent components should constitute less than **5%** of the overall system.

2. **Percentage of Code That Is Host-Dependent**
   - **Rationale**: A low percentage of host-dependent code ensures higher reusability and ease of deployment across different systems.
   - **Testing/Measurement**:
     - Use static analysis tools to scan for platform-specific code.
     - Target: Host-dependent code should be limited to **10% or less**.
3. **Use of a Proven Portable Language**
   - **Rationale**: Languages with established portability (e.g., Python, Java) reduce dependencies on specific platforms and compilers.
   - **Testing/Measurement**:
     - Verify language portability by testing on multiple operating systems (e.g., Linux, Windows, macOS).
     - Target: The system should run seamlessly on at least **three major operating systems**.
4. **Use of a Particular Compiler or Language Subset**
   - **Rationale**: Restricting the use of non-standard or platform-specific compiler features ensures consistent behavior across environments.
   - **Testing/Measurement**:
     - Compile the code using at least two standard-compliant compilers (e.g., GCC, Clang) and validate outputs.
     - Target: No compiler-specific extensions should cause incompatibility.
5. **Use of a Particular Operating System**
   - **Rationale**: Designing for cross-platform compatibility ensures that the software is not locked to a single OS.
   - **Testing/Measurement**:
     - Perform functional tests on all target operating systems.
     - Target: The system should demonstrate equivalent functionality and performance across all platforms

ID	**Characteristic**	**H/M/L**	1	2	3	4	5	6	7	8	9	10	11	12

1	Correctness	H												
2	Efficiency	M												
3	Flexibility	H												
4	Integrity/Security	H												
5	Interoperability	M												
6	Maintainability	H												
7	Portability	H												
8	Reliability	H												
9	Reusability	M												
10	Testability													

		H												
11	Usability	M												
12	Availability	H												

Definitions of the quality characteristics not defined in the paragraphs above follow.

- *Correctness -Ensure code correctness with automated tests.*
- *Efficiency - Use lightweight frameworks for performance.*
- *Flexibility - Modular design for easier adaptability.*
- *Interoperability - Use standard APIs and protocols.*
- *Reliability - Implement robust error-handling mechanisms.*
- *Reusability - Design components for independent usage.*
- *Testability - Include automated test coverage.*
- *Usability - User-centric design for intuitive operation.*

3.7 External Actor Descriptions

3.7.1 Human actors

- **Job Seekers**: Students, graduates, or professionals seeking internships or career opportunities. They search, apply, and track job applications.
- **Recruiters/Employers**: Company representatives who post job openings, review applications, and manage hiring processes.
- **University/College Administrators**: Personnel who validate student profiles, manage partnerships with employers, and oversee the portal's operations.
- **Portal Administrators**: Individuals responsible for maintaining the system, managing user accounts, and addressing technical issues.
- **Career Counselors**: Professionals providing guidance to job seekers on their career paths and application processes.

3.7.2 Hardware actors

- **User Devices**: Personal computers, laptops, smartphones, and tablets used by human actors to access the portal.
- **Web Servers**: Hardware running the portal's web application, hosting the front-end and back-end services.
- **Database Servers**: Servers storing user data, job postings, application records, and analytics.
- **Network Devices**: Routers, firewalls, and other network infrastructure ensure secure and efficient communication between users and the system.

3.7.3 Software System Actors

- **Job Aggregator APIs**: Services like LinkedIn, Indeed, or Glassdoor for aggregating job postings.
- **Authentication Services**: Tools like Google Sign-In, LinkedIn OAuth, or custom authentication for user login and registration.
- **Payment Gateways**: Systems handling payments for premium services (e.g., advanced job recommendations, resume optimization).
- **University/College Systems**: Integration with educational institutions' internal systems for student verification and performance tracking.
- **Applicant Tracking Systems (ATS)**: Employer-side systems that synchronize with the portal to manage job applications.
- **Email/Notification Services**: Third-party tools like Twilio or SendGrid for sending application updates, job alerts, and reminders.
- **Analytics Platforms**: Tools like Google Analytics or custom analytics engines for tracking user behavior and system performance.

3.7.4 Use-Case Descriptions

3.7.4.1 Use case 1

8

2.1.2 Interfaces
2.1.3 Hardware Interfaces
2.1.4 Software Interfaces
2.1.5 Communications Interfaces
2.1.6 Memory Constraints
2.1.7 Operations
2.1.8 Site Adaptation Requirements

2.2 Product Functions

2.3 User Characteristics

2.4 Constraints

2.5 Assumptions and Dependencies

3. Specific Requirements

3.1 External interfaces

3.2 Functions

3.3 Performance Requirements

3.4 Logical Database Requirements

3.5 Design Constraints
3.5.1 Standards Compliance

3.6 Software System Attributes
3.6.1 Reliability
3.6.2 Availability
3.6.3 Security
3.6.4 Maintainability

3.6.5 Portability

3.7 External Actor Descriptions
3.7.1 Human actors
3.7.2 Hardware actors
3.7.3 Software system actors
3.7.4 Use-Case Descriptions
3.7.4.1 Use case 1
3.7.4.2 use case n

3.8 Sequence Diagram

3.8 Activity Diagram

6. Supporting Information

1. Introduction

In the digital age, the pursuit of career advancement and skill development has become a critical priority for both job seekers and students. However, many existing platforms are limited in scope, focusing solely on job or internship listings without providing essential resources to streamline the job application process or enhance career readiness. The lack of integrated solutions often leaves users juggling multiple tools to prepare for interviews, build resumes, or gain the skills necessary for competitive opportunities.

To address these challenges, this study proposes the development of a "Career & Internship Portal"—a centralized, user-friendly platform designed to optimize the career development process. This system not only simplifies access to job and internship listings but also integrates essential tools such as resume builders, mock interview schedulers, and comprehensive career resources. By offering these functionalities in one cohesive platform, the portal aims to empower users at every stage of their professional journey.

1.1 Purpose

The purpose of "Career & Internship Portal" is to develop a centralized and comprehensive platform that integrates job and internship listings with career preparation resources. The system seeks to streamline the job search and application process, helping users enhance their career readiness. By offering features such as resume-building tools, mock interview scheduling, and task management, the portal addresses the current limitations of job platforms, which often lack integrated career development tools.

The platform is designed to cater to a diverse user base, including students and busy parents, providing tailored resources that align with their unique career goals and responsibilities.

1.2 Scope

The "Career & Internship Portal" is envisioned as a web-based and mobile-friendly application aimed at addressing the gaps in existing job boards and career platforms. Its scope includes:

• **Centralized Job and Internship Listings:** Users can easily search, filter, and apply for positions based on their preferences.
• **MResume-Building Tools:** Provides customizable templates and real-time editing capabilities to assist users in creating effective resumes.
• **Mock Interview Scheduling:** Enables users to book and practice interviews using an integrated calendar system with reminders.
• **Career Preparation Resources:** Offers access to tutorials, industry guides, and tips for skill-building and professional growth.
• **Task Scheduling and Management:** Helps users organize and prioritize career-related activities, such as application deadlines and interview schedules.

• **Accessibility:** Ensures a seamless user experience on both desktop and mobile platforms.

This system will serve as a one-stop career development tool for students and job seekers.

1.3 Definitions, Acronyms, and Abbreviations.

Definitions

1. **Career & Internship Portal:**

A comprehensive online platform designed to assist users in finding job and internship opportunities, building resumes, preparing for interviews, and accessing career development resources. It integrates job listings, career resources, and tools like resume builders and mock interview scheduling in one centralized system.

2. **User:**

An individual who interacts with the Career & Internship Portal. This can include students, parents, job seekers, or anyone using the portal to search for jobs, internships, or career development resources.

3. **Resume Builder:**

A tool within the portal that helps users create and format resumes, often using templates and step-by-step guidance to produce professional documents tailored to specific job applications.

4. **Mock Interview:**

A simulated interview session designed to help users practice their interview skills. It is integrated into the portal as a scheduling tool that allows users to book mock interview sessions with career coaches or automated systems.

5. **Job and Internship Listings:**

A section within the portal where available job and internship opportunities are displayed. Users can browse, filter, and apply to relevant positions.

6. **API (Application Programming Interface):**

A set of protocols and tools that allows different software systems to communicate with each other. In this project, APIs like Google Maps API and Google Calendar API will be integrated for location-based services and scheduling features.

7. **User Profile:**

A personalized space within the portal where users store their information, including contact details, resume, job preferences, and application history.

Acronyms

1. **SRS (Software Requirements Specification):**

A detailed document that defines the software's functionality, design, and behavior as required for development. It serves as a blueprint for developers, stakeholders, and project teams.

2. **UI (User Interface):**

The visual elements and layout through which users interact with the portal, including buttons, menus, forms, and other interactive components.

3. **UX (User Experience):**

The overall experience a user has when interacting with the portal, encompassing usability, design, accessibility, and the efficiency of interactions.

4. **DBMS (Database Management System):**

A system used to manage and store data, such as MySQL or PostgreSQL, used to store user profiles, job listings, applications, and other portal data.

5. **HTTP (HyperText Transfer Protocol):**

A protocol used for transmitting hypertext via the internet, essential for the functioning of web-based applications, including the Career & Internship Portal.

6. **SSL (Secure Sockets Layer):**

A security protocol used to establish an encrypted link between the user's browser and the portal, ensuring that sensitive data (such as resumes and application information) is transmitted securely.

7. **API (Application Programming Interface):**

A set of protocols that enables the integration of external tools and services (e.g., Google Maps, Google Calendar) with the portal for enhanced functionality.

Abbreviations

1. **EC2 (Elastic Compute Cloud):**

An Amazon Web Services (AWS) product used to host web applications, providing scalable computing resources for the portal.

2. **S3 (Simple Storage Service):**

An AWS service for storing and managing user data, such as resumes, documents, and media files within the Career & Internship Portal.

3. **RDS (Relational Database Service):**

An AWS-managed service for setting up, operating, and scaling a relational database in the cloud. It will store the Career & Internship Portal's data, including job listings, user profiles, and applications.

4. **HTML (HyperText Markup Language):**

The standard markup language used to create the structure of web pages, such as the Career & Internship Portal's website.

5. **CSS (Cascading Style Sheets):**

A language used to style the layout and appearance of web pages. It will be used to create a responsive, visually appealing interface for the portal.

6. **JavaScript (JS):**

A programming language that enables interactivity on web pages. It will be used to add dynamic functionality to the Career & Internship Portal, including form handling, real-time content updates, and interactive features.

7. **PDF (Portable Document Format):**

A widely-used document format that will allow users to download their resumes in a consistent and secure format through the portal.

1.4 References

1. **Career Development Tools and Strategies**

• Heintz, J., & Weber, R. (2019). Effective Career Preparation: Building Skills for Workplace Success. New York: Career Strategies Press.

- National Association of Colleges and Employers (NACE). (2022). Career Readiness Competencies. Available at: www.naceweb.org

2. Web and Mobile Application Development

- Flanagan, D. (2020). JavaScript: The Definitive Guide. O'Reilly Media.
- Duckett, J. (2014). HTML and CSS: Design and Build Websites. Wiley Publishing.
- Google Developers (2023). Google Maps and Calendar APIs Documentation. Available at: https://developers.google.com/

3. System Design and Database Management

- Connolly, T., & Begg, C. (2020). Database Systems: A Practical Approach to Design, Implementation, and Management. Pearson Education.
- AWS Documentation (2024). Amazon Web Services Cloud Hosting Solutions. Available at: https://aws.amazon.com/documentation/

4. User-Centered Design and Usability

- Krug, S. (2014). Don't Make Me Think: A Common Sense Approach to Web Usability. New Riders.
- Nielsen, J. (2012). Usability Engineering. Academic Press.

5. Security Standards and Practices

- Stallings, W. (2019). Cryptography and Network Security: Principles and Practice. Pearson Education.
- Open Web Application Security Project (OWASP). (2023). OWASP Top Ten Web Application Security Risks. Available at: https://owasp.org/

6. Project Management and Software Development Lifecycle (SDLC)

- Sommerville, I. (2015). Software Engineering. Pearson.
- Royce, W. W. (1970). Managing the Development of Large Software Systems. Proceedings of IEEE WESCON.

7. Career Services and Job Portals

- LinkedIn Talent Solutions (2023). Features of LinkedIn for Job Seekers and Employers. Available at: https://business.linkedin.com/
- Indeed Help Center (2023). Using Job Boards for Career Growth. Available at: https://www.indeed.com/help/

8. Accessibility and Inclusivity Standards

- W3C. (2018). Web Content Accessibility Guidelines (WCAG) 2.1. Available at: https://www.w3.org/WAI/standards-guidelines/wcag/

9. Relevant Surveys and Case Studies

- Pew Research Center. (2022). Workplace Trends and Employment Challenges. Available at: https://www.pewresearch.org/
- Case Study: CareerOneStop (2023). Integrating Career Services into Digital Platforms. CareerOneStop.org.

1.5 Overview

The portal is designed to bridge the gap between job seekers and career development resources. Unlike traditional job boards, the "Career & Internship Portal" integrates career preparation tools into a single platform. It offers:

1. Ease of Use: A user-friendly interface for both web and mobile devices.
2. Enhanced Career Readiness: By providing tools for resume creation and mock interviews, the portal equips users with the skills needed for success in the competitive job market.
3. Streamlined Process: Simplifies job and internship searches by providing relevant listings, tutorials, and integrated scheduling tools.

The system aims to empower users to take control of their career journeys by offering a comprehensive and efficient solution to their needs.

2. The Overall Description

The Career & Internship Portal is a centralized platform designed to streamline the career development process for students, job seekers, and professionals. It offers a comprehensive suite of features, including job and internship listings, a resume-building tool, mock interview scheduling, and access to career resources such as industry guides and tutorials. The portal caters to diverse user needs by simplifying job searches, enhancing career readiness, and supporting skill development through an intuitive and user-friendly interface. Accessible via web or mobile application, the system ensures flexibility and convenience, integrating essential tools to help users confidently navigate the competitive job market. By addressing common challenges such as fragmented resources and inefficient application processes, the platform empowers users to achieve their career goals effectively.

2.1 Product Perspective

The "Career & Internship Portal" is a self-contained platform designed to streamline career development by integrating job and internship listings, resume-building tools, mock interview scheduling, and career resources into one centralized system. Unlike traditional job boards, which primarily focus on job postings, this portal includes features to enhance user preparedness and increase employability.

The system will function as an independent web-based platform but can also integrate with external systems such as job listing APIs, career service tools, and external scheduling software. Comparatively, the system differs from existing platforms like LinkedIn or Indeed by prioritizing a comprehensive, user-focused approach to career preparation.

2.1.1 System Interfaces

The system interfaces with several external services and tools to deliver its functionalities effectively. Key system interfaces include:

- **Google Calendar API**: Enables users to schedule and track career events, such as application deadlines and mock interviews.
- **Google Maps API**: Provides location-based services to help users navigate to interviews and networking events.

- **Authentication APIs** (e.g., OpenID, Google Sign-In): Ensure secure user login and data access.
- **Job Listing APIs**: Allow the system to fetch and display job and internship opportunities from external sources.

These interfaces ensure that external systems and services seamlessly contribute to the platform's features.

2.1.2 Interfaces

The system employs a **Graphical User Interface (GUI)** to interact with users. The interface is designed to be intuitive and user-friendly, ensuring accessibility for all user demographics, including students and busy parents. Key features of the user interface include:

- Responsive design for compatibility across desktops, tablets, and smartphones.
- ADA-compliant features to ensure usability for users with disabilities.
- Interactive dashboards for easy navigation of tools, such as the resume builder and job application tracker.

By optimizing the interface for simplicity and accessibility, the platform caters to diverse user needs.

2.1.3 Hardware Interfaces

The system has no specific hardware interface requirements beyond access to standard internet-enabled devices. These include:

- Personal computers, tablets, or smartphones for accessing the portal.
- Network connections for communication with external APIs and services.

This minimal dependency on specialized hardware ensures ease of use and broad compatibility.

2.1.4 Software Interfaces

The system interacts with various external software to enhance functionality, including:

- **Google Maps API**: Used for location-based features like navigation to interviews.
- **Google Calendar API**: Synchronizes career events and deadlines.
- **jsPDF**: Enables users to export resumes in PDF format.
- **Database Systems**: MySQL or PostgreSQL will manage user profiles, application histories, and system data.

The choice of these software components ensures scalability, interoperability, and ease of implementation for developers.

2.1.5 Communications Interfaces

The system utilizes secure **HTTPS protocols** for all user interactions, ensuring encrypted communication between the client and the server. Additionally, RESTful APIs facilitate communication with external services, such as:

- Authentication (OpenID or Google Sign-In).
- Job Listing APIs for fetching data from external job boards.
- Scheduling APIs for managing interview and application events.

These communication interfaces maintain system reliability and user data privacy.

2.1.6 Memory Constraints

The system relies on cloud-based storage solutions for scalability and efficiency. Memory constraints include:

- Support for storing large amounts of user-generated data, such as resumes, application records, and user profiles.
- Optimization to ensure quick data retrieval, even as user data volume increases.
  There are no strict memory constraints for primary or secondary memory on user devices.

2.1.7 Operations

The platform supports the following operations to ensure a seamless user experience:

- **Interactive Operations**: Users can perform tasks such as job searching, resume building, and scheduling mock interviews.
- **Automated Notifications**: Reminders for deadlines, interviews, and career-related tasks will be generated.
- **Backup and Recovery**: Cloud-hosted databases will feature automatic backups to ensure data reliability and recovery in case of system failures.

These operations enhance system reliability and ensure uninterrupted user access to essential tools.

2.1.8 Site Adaptation Requirements

● **Deployment:** The system must be deployable on cloud platforms or university-hosted servers to ensure reliability and accessibility.
● **Customization:** Adaptable to different institutional or organizational requirements.

2.2 Product Functions

Job and Internship Listings
●Users can browse and apply for job or internship opportunities.
●Listings can be filtered by location, industry, and user preferences.

Resume-Building Tools
●Provides pre-designed templates and editing tools for creating professional resumes.
●Allows users to download resumes in PDF format using libraries like jsPDF.

Mock Interview Scheduling
●Integration with Google Calendar for booking interview practice sessions.
●Users receive automated reminders and feedback.

Career Preparation Resources
●Access to tutorials, tips, and industry-specific guides.
●Provides insights on how to improve job application success rates.

Task Management
●Scheduling tools to organize job applications, interviews, and deadlines.
●Tracks missed events and offers feedback for improved time management.

Accessibility
●A mobile-responsive interface that ensures usability on various devices.

2.3 User Characteristics

Students

- **Profile:** Typically balancing academic schedules with career preparation.
- **Needs:** Simplified tools for finding internships, building resumes, and practicing interview skills.

Busy Parents

- **Profile:** Working or stay-at-home parents seeking flexible employment.
- **Needs:** Resources for updating resumes and searching for jobs that fit their schedules.

2.4 Constraints

1. Regulatory Policies

- The system must comply with the **Philippine Data Privacy Act of 2012 (Republic Act No. 10173)**, including:
  - **Fair Information Processing Principles** to ensure lawful, fair, and transparent data collection and processing.
  - **Data Subject Rights** for users to access, rectify, and delete their personal data.
  - **Data Security** protocols to protect user data from unauthorized access and breaches.
  - **Breach Notification Procedures** to notify the National Privacy Commission (NPC) and affected users in case of a data breach.
- Accessibility standards must adhere to **Web Content Accessibility Guidelines (WCAG)** to support users with disabilities, including:

  - Alternative text for images.
  - Sufficient color contrast.
  - Keyboard navigation and assistive technology compatibility.

2. Hardware Limitations

- The system is designed to function on standard internet-enabled devices such as personal computers, laptops, smartphones, and tablets.
- No specialized hardware is required, ensuring compatibility with commonly used devices.
- The system relies on stable network connections for real-time features like job searches, calendar integration, and notifications.

3. Interface to Other Applications

- The system integrates with external tools and APIs to enhance functionality, including:
  - **Google Calendar API** for scheduling and reminders.
  - **Google Maps API** for location-based navigation.
  - **Authentication APIs** (e.g., Google Sign-In, OpenID) for secure user login.
  - **Job Listing APIs** for retrieving job and internship data from external sources.
- These interfaces rely on **RESTful APIs** over HTTPS for secure and reliable communication.

4. Parallel Operation

- The system must support simultaneous usage by multiple users, including operations like job searching, resume building, and mock interview scheduling.
- It must maintain optimal performance even during peak usage periods, with efficient resource management to avoid bottlenecks.

5. Audit Functions

- A robust logging system must track user actions, system activities, and changes to data for audit and troubleshooting purposes.
- Logs must be securely stored and accessible only to authorized personnel to ensure compliance and maintain system integrity.

6. Control Functions

- Portal administrators must have access to control functions, including:
  - User account management.
  - Oversight of job postings and career resources.
  - Monitoring and managing system health and performance.
- Administrative controls must be restricted to authorized users to ensure secure operations.

7. Higher-Order Language Requirements

- The system should be developed using a standard, portable programming language such as **Python** or **Java**, ensuring compatibility and maintainability.
- Code must adhere to consistent standards (e.g., **PEP 8** for Python) and follow modular architecture for ease of updates and scalability.

8. Signal Handshake Protocols

- While the system does not explicitly depend on low-level signal handshake protocols like **XON-XOFF** or **ACK-NACK**, reliable data transmission mechanisms must be implemented for interactions between the client and server.

9. Reliability Requirements

- The system is critical for career development and must maintain an **uptime of 99.9%**.
- Automated failover mechanisms should redirect users to backup servers during outages.
- A disaster recovery plan must ensure:
    - **Recovery Time Objective (RTO):** 4 hours.
    - **Recovery Point Objective (RPO):** 12 hours.
- Regular backups and system monitoring must be in place to minimize downtime and data loss.

10. Criticality of the Application

- As a platform central to users' career growth, any failure or downtime could significantly impact user trust.
- Notifications for scheduled maintenance or unexpected outages must be communicated promptly to users.

11. Safety and Security Considerations

- The system must prioritize data protection through:
    - **Strong encryption standards** (e.g., AES-256) for securing sensitive data.
    - **Secure authentication mechanisms**, such as multi-factor authentication (MFA).
- Regular **security audits** and penetration tests must be conducted to identify and resolve vulnerabilities.
- User sessions should have inactivity timeouts to reduce unauthorized access risks.

2.5 Assumptions and Dependencies

Assumptions

- The system assumes that users will have access to a stable internet connection to utilize real-time features such as job searches, calendar integration, and notifications.

- The majority of users will access the system through commonly used devices such as personal computers, laptops, smartphones, or tablets.
- External APIs (e.g., Google Calendar, Google Maps, job listing APIs) will remain available and functional throughout the system's lifecycle.
- Users will provide accurate and truthful information during account creation and profile updates.
- The system's hosting infrastructure will support scalability to handle an increasing number of users and concurrent operations.
- Security measures implemented in the system will remain effective against foreseeable cyber threats.

Dependencies

- **APIs and Third-Party Services**: The system depends on external APIs for key functionalities, including:
    - **Google Calendar API** for scheduling and reminders.
    - **Google Maps API** for navigation and location-based services.
    - **Job Listing APIs** for fetching job and internship data from external sources.
    - **Authentication Services** (e.g., Google Sign-In, OpenID) for secure user login.
- **Cloud Infrastructure**: The system relies on cloud services (e.g., AWS S3) for data storage, backups, and recovery.
- **Regulatory Compliance**: The system depends on adherence to laws such as the Philippine Data Privacy Act and other relevant data protection regulations.
- **User Devices**: The system is dependent on the availability of modern internet-enabled devices with updated browsers for optimal performance.
- **Hosting Environment**: The system's availability and uptime depend on reliable hosting infrastructure capable of 24/7 operation with minimal downtime.
- **Development Tools and Frameworks**: The system relies on chosen tools, programming languages (e.g., Python, Java), and frameworks to meet performance and compatibility requirements.

3. Specific Requirements

Software Requirements

1. **Frontend Technologies:**
- HTML, CSS, JavaScript, and React for responsive design.
2. **Backend Technologies:**
- Node.js or Django for server-side operations.
- MySQL or PostgreSQL for database management.
3. **APIs and Integrations:**
- Google Maps API for location-based job recommendations.
- Google Calendar API for mock interview scheduling.

Database Requirements

1. Store user profiles, job listings, resume data, and mock interview schedules.
2. Implement indexing for faster searches.

Development and Deployment Tools:

1. Version control via GitHub or GitLab.
2. Continuous integration and deployment pipelines using Jenkins or GitHub Actions.

Special Requirements

1. Mobile Application:
   - Develop an Android app with the same functionality as the web version.
   - Ensure compatibility with Android 8.0 and above.
2. Analytics Dashboard:
   - Provide admins with data on user activities, popular listings, and resource usage.
   - Generate monthly reports for strategic planning.
3. Accessibility:
   - Ensure compliance with WCAG 2.1 standards for accessibility, such as screen reader support and keyboard navigation.
4. Error Handling:
   - Provide clear error messages and fallback mechanisms for system failures.

3.1 External Interfaces

This section provides a detailed description of all inputs and outputs of the "Career & Internship Portal" software system, focusing on technical aspects relevant to the development team.

Name of item:

- Identifies each data element or interface (e.g., "Job Listings API," "User Profile Data," "Payment Gateway").

Description of purpose:

- Briefly explains the function or role of each interface.
  - Example: "Job Listings: To provide access to job postings from various sources."

Source of input or destination of output:
Specifies the origin or destination of data for each interface.

- Example: "Job Listings: External job boards (e.g., Indeed API), company websites, user submissions."

Valid range, accuracy and/or tolerance:

- Defines acceptable limits and expected accuracy for each data type.

- Example: "User Profiles: Valid email addresses, phone numbers, accurate education and employment history."

Units of measure:

- Specifies units of measurement where applicable (e.g., file size in KB/MB).

Timing:

- Defines the expected timing of data exchange (e.g., real-time, batch updates).

Relationships to other inputs/outputs:

- Describes how different interfaces interact and influence each other.
  - Example: "User Profiles are used to personalize the user experience and for targeted job recommendations."

Screen formats/organization:

- Specifies the desired presentation of data within the user interface.
  - Example: "User-friendly layouts for job/internship listings, easy navigation, clear calls to action."

Window formats/organization:

- Defines how data will be presented across different devices (e.g., responsive design).

Data formats:

- Specifies the expected data formats for input and output (e.g., JSON, XML, CSV).

Command formats:

- Defines the communication protocols and commands used for data exchange (e.g., RESTful APIs).

End messages:

- Specifies the types of messages returned to the system after an interface interaction (e.g., success/failure messages, error codes).

3.2 Functions

This section defines the fundamental actions that the "Career & Internship Portal" software system must perform. This maybe changed as the system is developed.

1. Job/Internship Listing Management

- **The system shall** retrieve job and internship listings from external sources (e.g., APIs, web scraping).
- **The system shall** validate the accuracy and completeness of retrieved job and internship data.
- **The system shall** store job and internship listings in the system database.
- **The system shall** allow users to search for job and internship listings based on various criteria (e.g., keywords, location, company, industry).
- **The system shall** filter search results based on user preferences (e.g., job type, experience level, salary range).
- **The system shall** display relevant job and internship listings to users.
- **The system shall** generate job and internship alerts based on user preferences.

2. User Profile Management

- **The system shall** allow users to create and manage their profiles.
- **The system shall** validate user input for accuracy and completeness (e.g., email address, phone number).
- **The system shall** store user profile data securely in the system database.
- **The system shall** allow users to update their profile information at any time.
- **The system shall** ensure the confidentiality and security of user profile data.

3. Resume Builder Functionality

- **The system shall** provide a user-friendly interface for resume creation.
- **The system shall** offer a selection of resume templates for users to choose from.
- **The system shall** allow users to import existing resumes from their local devices.
- **The system shall** enable users to edit and format resume content within the system.
- **The system shall** allow users to download their resumes in various formats (e.g., PDF, DOCX).
- **The system shall** generate cover letters based on resume data and job/internship descriptions.

4. Mock Interview Scheduling

- **The system shall** allow users to schedule mock interviews with mentors or other qualified individuals.
- **The system shall** validate and confirm interview availability with both the user and the interviewer.
- **The system shall** send automated reminders to both the user and the interviewer prior to the scheduled interview time.
- **The system shall** facilitate the exchange of feedback between the user and the interviewer after the interview.

5. Career Resources Management

- **The system shall** store and retrieve career resources (articles, guides, videos) from various sources.

- **The system shall** categorize and organize career resources for easy user access.
- **The system shall** allow users to search and filter career resources based on their interests and career goals.
- **The system shall** display career resources in a user-friendly and engaging format.

6. User Messaging and Communication

- **The system shall** enable secure and reliable communication between users and support staff.
- **The system shall** allow users to send and receive messages within the portal.
- **The system shall** facilitate the resolution of user inquiries and support requests.

7. Payment Processing (if applicable)

- **The system shall** integrate with a secure payment gateway (e.g., Stripe, PayPal).
- **The system shall** process payments for premium features (if applicable) accurately and securely.
- **The system shall** generate and store transaction records for all payments.

8. Data Security and Privacy

- **The system shall** implement robust security measures to protect user data from unauthorized access and breaches.
- **The system shall** comply with all relevant data privacy regulations (e.g., GDPR, Philippine Data Privacy Act).

9. Error Handling and Recovery

- **The system shall** detect and handle system errors gracefully.
- **The system shall** display informative error messages to users.
- **The system shall** implement mechanisms for system recovery in case of failures.

3.3 Performance Requirements

Given that the project is in the planning stage, these requirements should be met for the system.

Job Search Response Time: Job search results should be displayed within 5 seconds of the user submitting a query.
Resume Builder Load Time: The resume builder interface should load within 2 seconds.
Mock Interview Scheduling Response Time: Mock interview scheduling requests should be processed and confirmed within 3 seconds.
User Profile Update Processing Time: User profile updates should be processed and reflected on the screen within 3 seconds.

3.4 Logical Database Requirements

This section outlines the logical database requirements for the "Career & Internship Portal" using a SQL database schema.

3.4.1 Tables

- **Users**
    - user_id INT PRIMARY KEY AUTO_INCREMENT,
    - first_name VARCHAR(255) NOT NULL,
    - last_name VARCHAR(255) NOT NULL,
    - email VARCHAR(255) UNIQUE NOT NULL,
    -
    - password_hash VARCHAR(255) NOT NULL,
    -
    - date_of_birth DATE,
    - phone_number VARCHAR(20),
    - address TEXT,
    - education TEXT,
    - work_experience TEXT,
    - skills TEXT,
    - career_goals TEXT,
    - is_active BOOLEAN DEFAULT TRUE,
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
    -
- **Jobs**
    - job_id INT PRIMARY KEY AUTO_INCREMENT,
    - title VARCHAR(255) NOT NULL,
    - company_name VARCHAR(255),
    - description TEXT,
    - location VARCHAR(255),
    - job_type ENUM('Full-time', 'Part-time', 'Contract', 'Internship'),
    - required_skills TEXT,
    - salary_range DECIMAL(10, 2),
    - application_deadline DATE,
    - company_website TEXT,
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
    -
- **Internships**
    - internship_id INT PRIMARY KEY AUTO_INCREMENT,

    - title VARCHAR(255) NOT NULL,
    - company_name VARCHAR(255),
    - description TEXT,
    - location VARCHAR(255),
    - start_date DATE,
    - end_date DATE,
    - application_deadline DATE,
    - required_skills TEXT,
    - department VARCHAR(255),
    - supervisor VARCHAR(255),
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
    -
- **Resumes**

    - resume_id INT PRIMARY KEY AUTO_INCREMENT,
    - user_id INT,
    - file_path VARCHAR(255),
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
    - FOREIGN KEY (user_id) REFERENCES Users(user_id)
    -
- **CareerResources**

    - resource_id INT PRIMARY KEY AUTO_INCREMENT,
    - title VARCHAR(255),
    - resource_type ENUM('Article', 'Guide', 'Video'),
    - url TEXT,
    - category VARCHAR(255),
    - keywords TEXT,
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
- **MockInterviews**

    - interview_id INT PRIMARY KEY AUTO_INCREMENT,
    - user_id INT,
    - interviewer_id INT,
    - scheduled_time TIMESTAMP,
    - interviewer_feedback TEXT,
    - user_feedback TEXT,
    - created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    - FOREIGN KEY (user_id) REFERENCES Users(user_id),
    - FOREIGN KEY (interviewer_id) REFERENCES Users(user_id)
- **Messages**

- message_id INT PRIMARY KEY AUTO_INCREMENT,
- sender_id INT,
- recipient_id INT,
- message TEXT,
- created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
- is_read BOOLEAN DEFAULT FALSE,
- FOREIGN KEY (sender_id) REFERENCES Users(user_id),
- FOREIGN KEY (recipient_id) REFERENCES Users(user_id)

3.5 Design Constraints

This section outlines the constraints that may impact the design and development of the "Career & Internship Portal".

3.5.1 Standards Compliance

- **Data Privacy:** The system must comply with all relevant data privacy regulations, such as the **Philippine Data Privacy Act of 2012 (Republic Act No. 10173)**. This includes:
  - **Fair Information Processing Principles:** Ensuring data collection and processing are lawful, fair, and transparent.
  - **Data Subject Rights:** Implementing mechanisms for users to access, rectify, and delete their data.
  - **Data Security:** Implementing appropriate security measures to protect user data from unauthorized access, use, disclosure, alteration, or destruction.
  - **Data Breach Notification:** Establishing procedures for notifying the National Privacy Commission (NPC) and affected users in case of a data breach.
- **Accessibility:** The system must adhere to accessibility standards such as the **Web Content Accessibility Guidelines (WCAG)** to ensure usability for individuals with disabilities. This includes:
  - Providing alternative text for images.
  - Ensuring sufficient color contrast.
  - Making the interface navigable using only a keyboard.
  - Supporting assistive technologies (screen readers, screen magnifiers).
- **Usability:** The system must prioritize user experience and usability by:
  - Following established usability principles (e.g., consistency, simplicity, learnability).
  - Conducting user testing and incorporating feedback into the design.
  - Ensuring the system is easy to navigate and use for all target users.
- **Security:**

- Implement robust security measures to protect the system from cyber threats such as hacking, malware, and data breaches.
- Use strong encryption for sensitive data (e.g., passwords).
- Regularly conduct security audits and penetration testing.

3.6 Software System Attributes

Software System Attributes for the Career & Internship Portal

The software system attributes define the quality characteristics of the platform, ensuring it meets the expectations for performance, usability, and reliability. Below are the key attributes:

1. Performance

- The system must handle 1,000 concurrent users without performance degradation.
- Job search queries should return results within 3 seconds.
- Resume generation should complete within 5 seconds of input submission.
- The portal should support high-speed data access, even with large-scale data (e.g., 1,000+ job listings).

2. Scalability

- The platform must be designed to scale horizontally to handle an increasing user base and growing database size.
- Capable of supporting additional features or integrations without major system redesigns.
- Cloud-based infrastructure will allow dynamic scaling of resources based on demand.

3. Usability

- The system must have a user-friendly interface that is intuitive for both technical and non-technical users.
- Responsive design ensures usability across devices (desktop, tablets, smartphones).
- Include tooltips, tutorials, and an easy-to-navigate menu for better user experience.

4. Security

- User data (e.g., resumes, account information) must be encrypted using AES-256 encryption.
- Secure authentication via OAuth 2.0 for Google integration.
- Protect against common vulnerabilities such as SQL injection, cross-site scripting (XSS), and CSRF attacks.
- Implement role-based access control (RBAC) to restrict unauthorized actions.
- Regular security audits and penetration testing to identify and address vulnerabilities.

5. Reliability

- Ensure 95% uptime through the use of load balancers, redundant servers, and robust cloud hosting.
- Provide automated data backups to prevent loss in case of system failures.
- Incorporate failover mechanisms for uninterrupted service during server issues.

6. Maintainability

- Modular design to allow easy updates or bug fixes without affecting the entire system.
- Clean and well-documented codebase to simplify maintenance.
- Use version control systems like GitHub for code management and change tracking.
- Support continuous integration and deployment (CI/CD) pipelines for streamlined updates.

7. Accessibility

- Compliant with WCAG 2.1 standards for users with disabilities.
- Provide screen reader support for visually impaired users.
- Ensure all buttons and links are accessible via keyboard navigation.
- High-contrast mode for users with visual impairments.

8. Portability

- The web version should work seamlessly on modern browsers (Chrome, Firefox, Safari, Edge).
- The mobile app should be compatible with Android 8.0+ and iOS 12+.
- Ensure easy deployment on different hosting platforms (e.g., AWS, on-premises servers).

9. Interoperability

- Seamlessly integrate with third-party APIs such as:
- Google Maps API for location-based job recommendations.
- Google Calendar API for scheduling mock interviews.
- Provide standard data exchange formats (e.g., JSON, XML) for compatibility with external systems.

10. Availability

- The platform should provide 24/7 availability with robust monitoring tools like Amazon CloudWatch or New Relic.
- Scheduled maintenance periods should be communicated to users in advance.
- Incorporate disaster recovery mechanisms to restore services within 30 minutes of a failure.

11. Adaptability

- The platform must be designed to easily incorporate new features (e.g., AI-powered career coaching).

- Allow localization for multi-language support based on user region.
- Flexible enough to adapt to industry trends or user demands.

These attributes ensure the Career & Internship Portal is efficient, reliable, secure, and scalable while meeting the diverse needs of its users.

3.6.1 Reliability

Minimized downtime through the use of scalable cloud services.

3.6.2 Availability

System Availability Requirements

3. ***Availability Target****:*
    - *The system shall maintain an uptime of 99.9% during its operational hours (24/7 availability) to ensure consistent access for users worldwide.*
    - *Scheduled maintenance windows must be minimized and communicated to users at least 72 hours in advance.*

4. Failure Recovery Requirements

- *In the event of a failure, the system shall recover and resume normal operation within **15 minutes**.*
- *The system shall implement an automatic failover mechanism to redirect users to a backup server or data center in the event of a primary server failure.*

3. Data Recovery and Input Loss

- *The system shall ensure that:*
    - *User input loss during failures is minimized to **at most 10 characters of input** or equivalent actions within a session.*
    - *For incomplete form submissions (e.g., job applications or resume edits), the system shall auto-save drafts every **30 seconds** to reduce data loss.*

4. Checkpoint Mechanism

- *The system shall implement **transactional checkpoints** to save user progress at key interaction points, such as:*
    - *After submitting a job application.*
    - *After saving or updating a resume.*
    - *After scheduling a mock interview.*
- *These checkpoints ensure data consistency and facilitate smooth recovery after failures.*

5. Restart Capabilities

- *Upon restart after a failure, the system shall:*
    - *Restore the user's last active session state (if applicable) using session persistence mechanisms.*
    - *Notify the user of the recovery process and any data that might have been impacted.*
    - *Provide users with a clear error message and steps to re-engage with the system, ensuring transparency and trust.*

6. Redundancy and Scalability

- *The system shall use a **load-balanced, distributed architecture** to handle high traffic and prevent single points of failure.*
- *All critical components, including databases, application servers, and network services, shall have redundancy to ensure uninterrupted service.*

7. Monitoring and Alerts

- *A real-time monitoring system shall:*
    - *Detect failures immediately and notify administrators within **5 minutes**.*
    - *Generate automated alerts for potential performance issues, enabling proactive measures to prevent downtime.*

8. Disaster Recovery

- *The system shall have a documented **disaster recovery plan** with the following features:*
    - *Offsite backups updated every **12 hours** to ensure minimal data loss during major outages.*
    - *Recovery Time Objective (RTO): **4 hours**.*
    - *Recovery Point Objective (RPO): **12 hours**.*

3.6.3 Security

Data encryption and secure authentication mechanisms to protect user information.

3.6.4 Maintainability

Modularity:

- Modular architecture with logically separated components.
- Well-defined interfaces for minimal interdependencies.

Documentation:

- Comprehensive and version-controlled documentation, including design, API references, and maintenance guides.

- Code comments explaining non-obvious logic and decisions.

Code Quality and Standards:

- Adherence to consistent coding standards (e.g., PEP 8 for Python).
- Code review process and use of static analysis tools.

Testing and Validation:

- Unit test coverage of at least 80%.
- Integration and regression testing for system stability.
- Automated testing pipelines integrated with CI/CD.

Complexity Management:

- Avoid complex algorithms; use established design patterns.
- Cyclomatic complexity limited to a score of 10.
- Scheduled refactoring to reduce technical debt.

Interfaces:

- Standards-compliant internal and external interfaces.
- Versioned APIs for backward compatibility.

Configuration and Customization:

- Externalized and editable configuration settings.
- Support for modular upgrades with minimal system impact.

Dependency Management:

- Use dependency management tools (e.g., pip, npm).
- Regular updates to dependencies for security and compatibility.

Logging and Monitoring:

- Comprehensive logging for debugging and maintenance.
- Centralized logs and real-time monitoring dashboards.

Maintenance Handoff:

- Formal handoff meeting for knowledge transfer.
- Checklist of routine maintenance tasks for smooth transitions.

3.6.5 Portability

6. **Percentage of Components with Host-Dependent Code**

- **Rationale**: Minimizing host-dependent components reduces the effort required to adapt the software to different environments.
- **Testing/Measurement**:
  - Identify and quantify components that rely on platform-specific APIs or features.
  - Target: Host-dependent components should constitute less than **5%** of the overall system.

7. **Percentage of Code That Is Host-Dependent**
   - **Rationale**: A low percentage of host-dependent code ensures higher reusability and ease of deployment across different systems.
   - **Testing/Measurement**:
     - Use static analysis tools to scan for platform-specific code.
     - Target: Host-dependent code should be limited to **10% or less**.
8. **Use of a Proven Portable Language**
   - **Rationale**: Languages with established portability (e.g., Python, Java) reduce dependencies on specific platforms and compilers.
   - **Testing/Measurement**:
     - Verify language portability by testing on multiple operating systems (e.g., Linux, Windows, macOS).
     - Target: The system should run seamlessly on at least **three major operating systems**.
9. **Use of a Particular Compiler or Language Subset**
   - **Rationale**: Restricting the use of non-standard or platform-specific compiler features ensures consistent behavior across environments.
   - **Testing/Measurement**:
     - Compile the code using at least two standard-compliant compilers (e.g., GCC, Clang) and validate outputs.
     - Target: No compiler-specific extensions should cause incompatibility.
10. **Use of a Particular Operating System**
    - **Rationale**: Designing for cross-platform compatibility ensures that the software is not locked to a single OS.
    - **Testing/Measurement**:
      - Perform functional tests on all target operating systems.
      - Target: The system should demonstrate equivalent functionality and performance across all platforms

ID	**Characteristic**	**H/M/L**	1	2	3	4	5	6	7	8	9	10	11	12

1	Correctness	H												
2	Efficiency	M												
3	Flexibility	H												
4	Integrity/Security	H												
5	Interoperability	M												
6	Maintainability	H												
7	Portability	H												
8	Reliability	H												
9	Reusability	M												
10	Testability													

		H												
11	Usability	M												
12	Availability	H												

Definitions of the quality characteristics not defined in the paragraphs above follow.

- *Correctness -Ensure code correctness with automated tests.*
- *Efficiency - Use lightweight frameworks for performance.*
- *Flexibility - Modular design for easier adaptability.*
- *Interoperability - Use standard APIs and protocols.*
- *Reliability - Implement robust error-handling mechanisms.*
- *Reusability - Design components for independent usage.*
- *Testability - Include automated test coverage.*
- *Usability - User-centric design for intuitive operation.*

3.7 External Actor Descriptions

3.7.1 Human actors

- **Job Seekers**: Students, graduates, or professionals seeking internships or career opportunities. They search, apply, and track job applications.
- **Recruiters/Employers**: Company representatives who post job openings, review applications, and manage hiring processes.
- **University/College Administrators**: Personnel who validate student profiles, manage partnerships with employers, and oversee the portal's operations.
- **Portal Administrators**: Individuals responsible for maintaining the system, managing user accounts, and addressing technical issues.
- **Career Counselors**: Professionals providing guidance to job seekers on their career paths and application processes.

3.7.2 Hardware actors

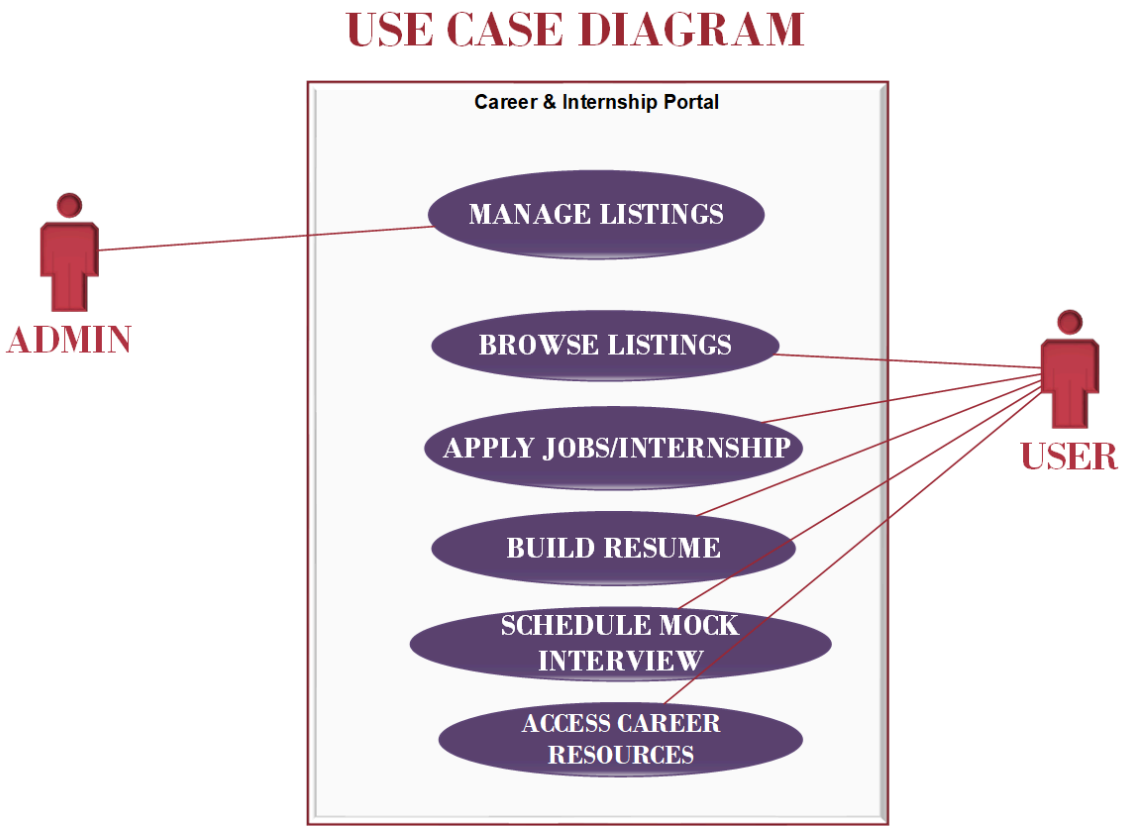
- **User Devices**: Personal computers, laptops, smartphones, and tablets used by human actors to access the portal.
- **Web Servers**: Hardware running the portal's web application, hosting the front-end and back-end services.
- **Database Servers**: Servers storing user data, job postings, application records, and analytics.
- **Network Devices**: Routers, firewalls, and other network infrastructure ensure secure and efficient communication between users and the system.

3.7.3 Software System Actors

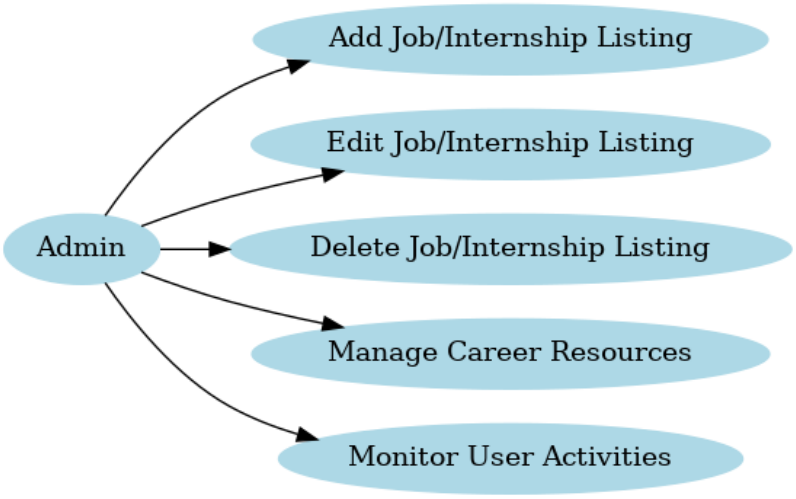
- **Job Aggregator APIs**: Services like LinkedIn, Indeed, or Glassdoor for aggregating job postings.
- **Authentication Services**: Tools like Google Sign-In, LinkedIn OAuth, or custom authentication for user login and registration.
- **Payment Gateways**: Systems handling payments for premium services (e.g., advanced job recommendations, resume optimization).
- **University/College Systems**: Integration with educational institutions' internal systems for student verification and performance tracking.
- **Applicant Tracking Systems (ATS)**: Employer-side systems that synchronize with the portal to manage job applications.
- **Email/Notification Services**: Third-party tools like Twilio or SendGrid for sending application updates, job alerts, and reminders.
- **Analytics Platforms**: Tools like Google Analytics or custom analytics engines for tracking user behavior and system performance.

3.7.4 Use-Case Descriptions

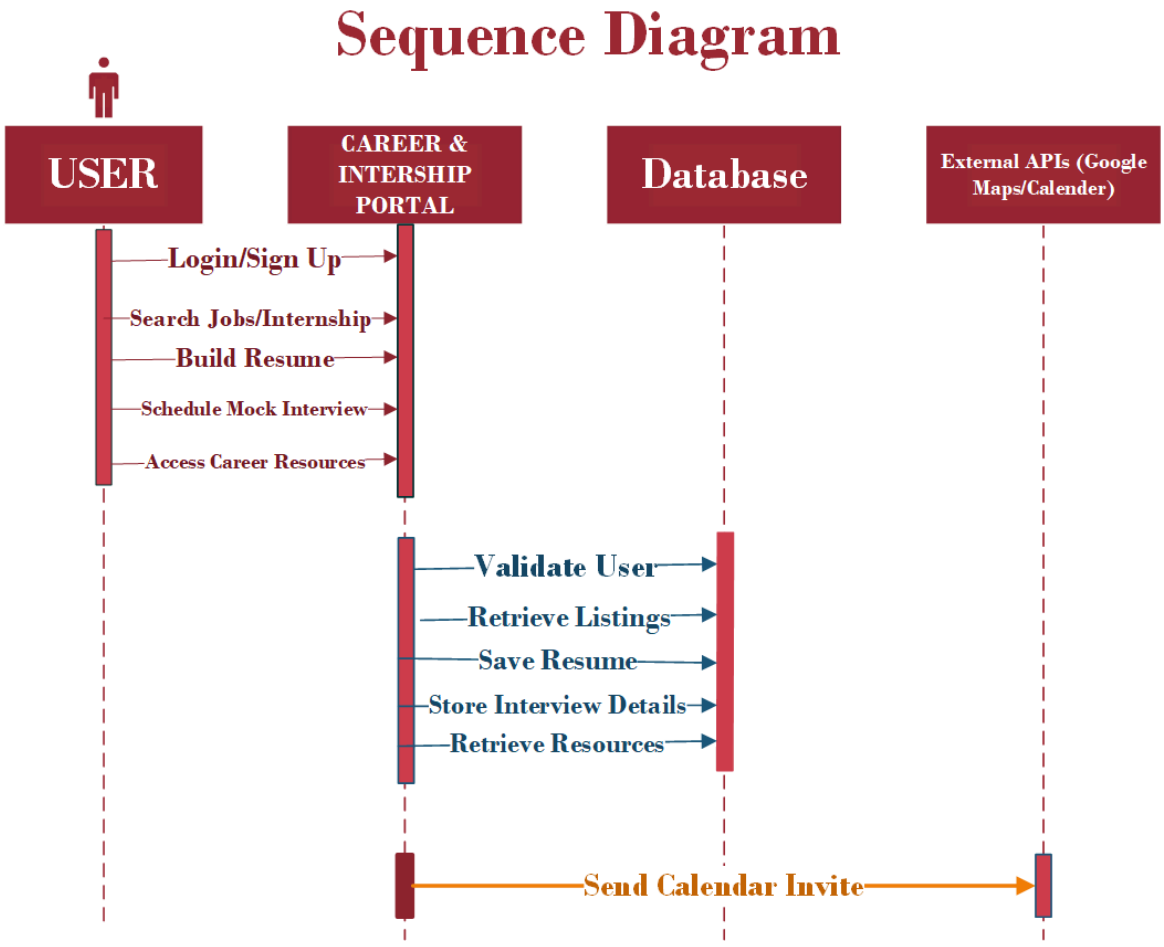
3.7.4.1 Use case 1



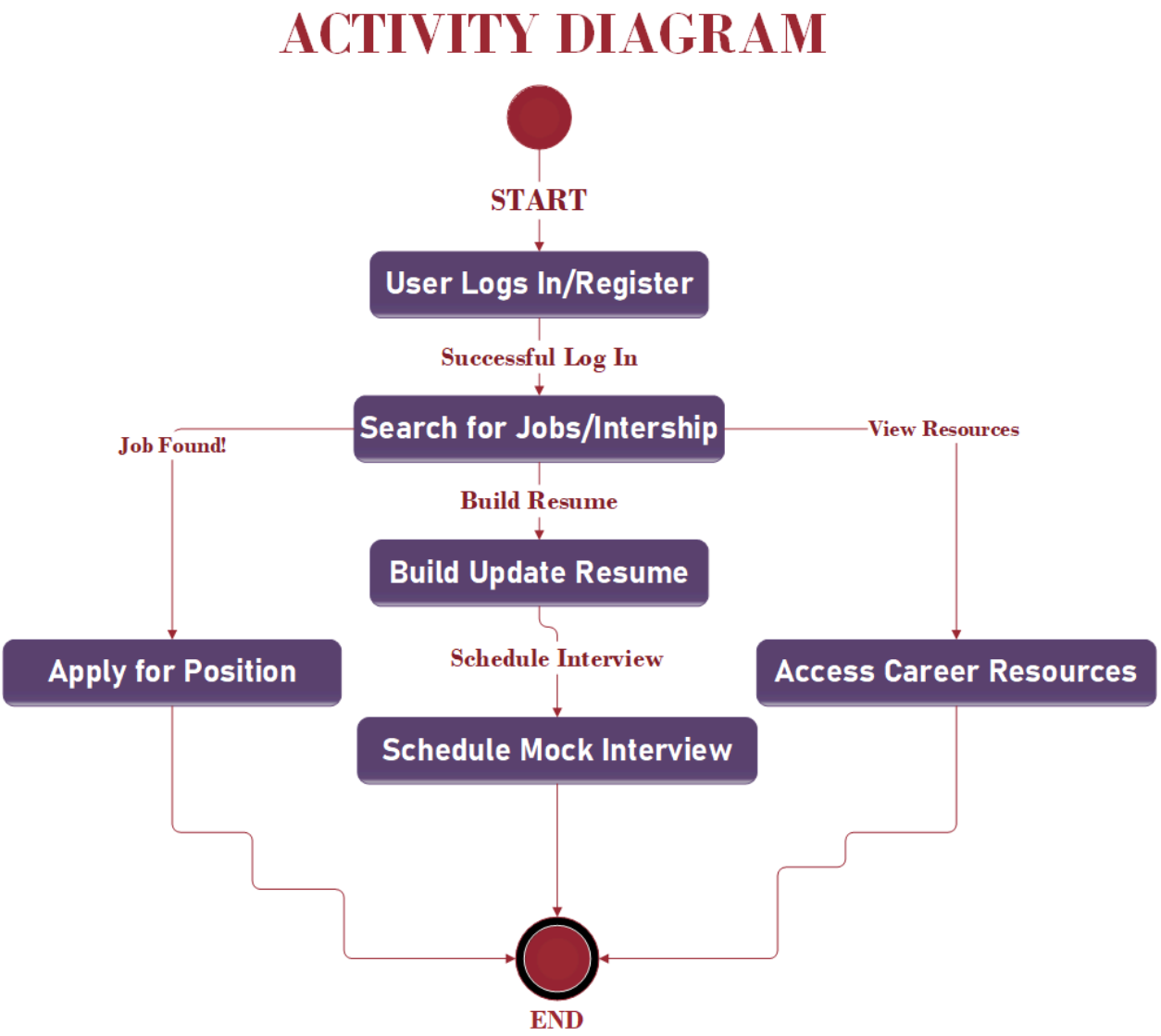
3.7.4.2 Use case 2



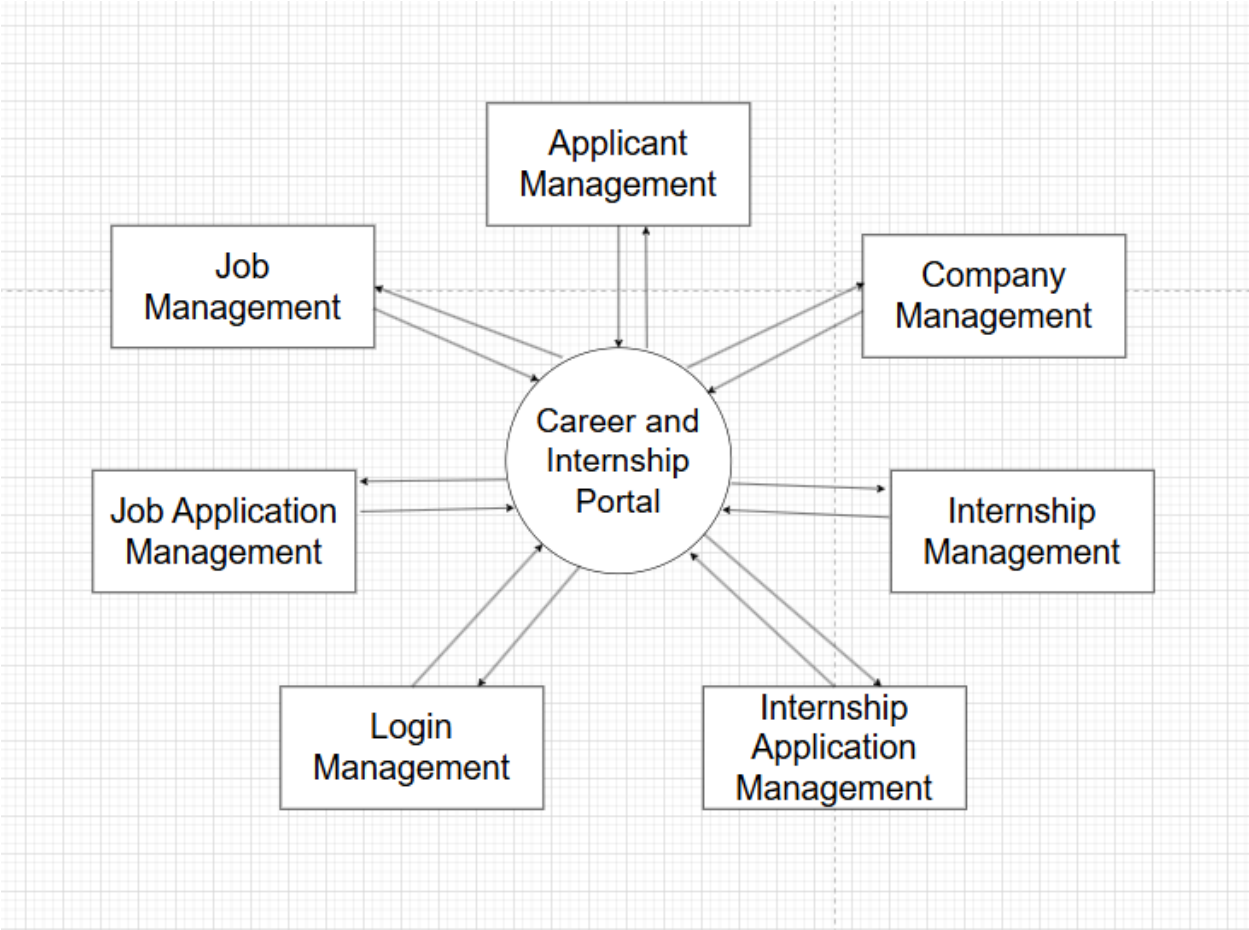
3.7.5 Sequence Diagram



3.7.6 Activity Diagram



3.7.7 Dataflow Diagram



4. Supporting Information

Sample Input:

1. Job Search: Users input preferences such as industry, location, salary range, and skills into search filters.
2. Resume Builder: Users fill forms with personal details, education history, work experience, and certifications.
3. Mock Interview Scheduler: Users input preferred dates, times, and topics for interview preparation.

Sample Output:

1. Job recommendations tailored to user input are displayed in a ranked list.
2. Generated resumes in PDF format are downloadable for direct job applications.
3. Scheduled mock interview sessions are synced to the user's calendar.

(b) Background Information

The Career & Internship Portal was developed to address common challenges in career readiness and job search processes. Research conducted among students and job seekers revealed that:

- Over 70% faced difficulties in finding internships aligned with their academic background.
- Nearly 60% lacked access to integrated tools for resume building and interview preparation.
- Existing platforms were fragmented, requiring users to rely on multiple websites and applications for essential career resources.

This portal was conceptualized as a holistic solution to streamline these processes and support users in their career journeys.

(c) Description of Problems Addressed

1. Disjointed Career Tools:
- Problem: Users must navigate multiple platforms for job listings, resume building, and mock interviews.
- Solution: Integrate all functionalities into a single portal.
2. Limited Career Guidance:
- Problem: Users often lack access to resources that cater to specific industries or roles.
- Solution: Provide curated guides and tutorials for skill-building and professional development.
3. Time Management Issues:
- Problem: Users struggle to manage job application deadlines and interview schedules.
- Solution: Offer task scheduling tools and calendar integration.

(d) Special Instructions

1. **Security Considerations:**
- Implement role-based access control to protect sensitive user data.
- Encrypt stored resumes and application data using AES-256 encryption standards.
2. **Platform Deployment:**
- Ensure mobile applications comply with Google Play and Apple App Store policies.
3. **Integration Requirements:**
- Use Google Maps API for location-based job listings.
- Integrate Google Calendar for scheduling mock interviews and tracking deadlines.
4. **Export Considerations:**
- Ensure resume templates meet international standards for job applications.
- Provide multi-language support for users in non-English-speaking regions.