Recruitment System

B Pavan Kumar – 22bd1a6608

B Laxmi Sai Praneeth – 22bd1a6606

Ch Thirush Reddy– 22bd1a660A

T Rajesh – 22bd1a661T

K Kartheek – 22bd1a660Q

Bramha – 22bd1a660Y

Problem Statement:

The recruitment process is often time-consuming, complex, and riddled with inefficiencies. Traditional recruitment methods involve manually sifting through piles of resumes, scheduling interviews, and managing multiple stages of the hiring process, which can lead to delays, errors, and missed opportunities. Such inefficiencies result in a prolonged hiring cycle, increased operational costs, and challenges in finding the right candidates.

A Recruitment Management System aims to solve these problems by providing an automated, centralized platform for managing the end-to-end hiring process. It streamlines candidate sourcing, application tracking, interview scheduling, and communication, reducing the workload on HR teams and enabling them to focus on selecting the best talent. This digital solution enhances the recruitment experience for both employers and candidates by ensuring faster, more efficient, and data-driven hiring decisions, ultimately leading to improved talent acquisition and reduced time-to-hire.

Software Requirement Specification

For

Recruitment System

Version 1.0 approved

Prepared by:

1. B Pavan Kumar – 22bd1a6608
2. B Laxmi Sai Praneeth – 22bd1a6606
3. Ch Thirush Reddy– 22bd1a660A
4. T Rajesh – 22bd1a661T
5. K Kartheek – 22bd1a660Q
6. Bramha – 22bd1a660Y

Keshav Memorial Institute Of Technology

19-10-2024

Table of Contents

Revision History

Name	Date	Reasons for Changes	Version
Week-1	26-09-2024	SRS-Template Creation(Introduction)	1.0
Week-2	19-10-2024	SRS Documentation-Use case , class diagram	1.1
Week-3	25-10-2024	SRS Final Document	1.2

1.Introduction

1.1 Purpose

The Campus Recruitment System is designed to streamline and automate the process of recruitment for educational institutions. The system brings together students, companies, and administrators into a unified platform that facilitates job posting, application management, and status tracking. It allows students to search and apply for jobs, companies to post jobs and manage applications, and administrators to oversee the entire recruitment process, including managing both student and company profiles. The system ensures that all recruitment activities are handled efficiently, improving the overall process for both students and recruiters. It aims to bridge the gap between students seeking employment and companies seeking talent by providing a centralized, secure, and user-friendly platform.

1.2 Document Convention

Heading:

Font-Size:16

Font-Style: Bold

Font: Times New Roman

Subheading:

Font-Size:14

Font-Style: Bold

Font: Times New Roman

Content:

Font-Size:12

Font: Times New Roman.

1.3 Intended Audience and Reading Suggestions

This document is intended for multiple stakeholders:

Developers: Should focus on functional and nonfunctional requirements, class diagrams, and use case diagrams to understand how different actors (students, companies, and admin) will interact with the system. These sections will help developers design the system’s architecture and implement it effectively.

Project Managers: Should review timelines, resource allocation, and potential risks. The sections on performance criteria and functionality will provide insight into delivering the system on schedule, within budget, and managing associated risks.

Campus Administrators: Should understand how the system automates student and company management, job postings, and application tracking, allowing for smoother recruitment operations.

Recruiters/Companies: Should focus on the features that enable them to post job vacancies, view student applications, and manage their recruitment process through the system efficiently.

Testers: Can use this document to define functional and non-functional test cases, ensuring that the system meets all operational criteria and performs as expected under various conditions.

1.4 Product Scope

The Campus Recruitment System (CRS) offers a centralized platform for managing recruitment-related activities. The system includes three main types of users: Students, Companies, and Administrators.

For Students: The system allows students to create profiles, search for available jobs, apply for positions, view their application status, and update their details as necessary.

For Companies: Companies can create job postings, view student applications, and call selected students for interviews. They can also update their profile information and manage their recruitment process efficiently.

For Admins: Admins can manage both student and company profiles, oversee job postings, and ensure that the recruitment process runs smoothly. They can update the system's details and maintain overall system integrity.

The system provides login, logout, update details, and password change functionalities for all users. Additionally, it offers the following core services:

Students: Apply for jobs, view job status, and search jobs.

Companies: Post jobs, view applications, and call for interviews.

Admins: Manage students and companies.

This integrated solution simplifies recruitment by providing an all-in-one platform for managing job applications and recruitment activities. It enhances the recruitment process's transparency, security, and efficiency.

1.5 References

SHRM (Society for Human Resource Management) – Guidelines on recruitment strategies and efficient hiring processes.

MySQL/PostgreSQL documentation – For database management, schema design, and API integrations.**React and Node.js Documentation** – Framework guides for building web applications and managing asynchronous operations.

2.Overall Description

2.1 Product Perspective

The Recruitment Management System (RMS) serves as an integrated platform to address inefficiencies in traditional recruitment processes. It simplifies the complex hiring procedures by centralizing job postings, candidate applications, and employer interactions. The system provides specific interfaces for students (candidates), admins (HR or university staff), and companies (recruiters), as illustrated in the use case diagram. It ensures smoother management of the recruitment lifecycle, from application to hiring, with roles and functionalities defined in the class diagram.

2.2 Product Functions

1. Student Functions:

- Search and apply for jobs.
- Track the status of job applications.
- Update personal and academic details.

2. Admin Functions:

- Manage student and company records.
- Approve or manage job postings.
- Monitor system activities and ensure smooth operation.

3. Company Functions:

- Post job openings and manage listings.
- Review and manage student applications.
- Update company profile details.

4. Centralized Login System:

- Secure login for all user types (students, admins, companies).
- Role-based access control to ensure appropriate permissions.

5. Real-Time Communication:

- Instant notifications for job applications, updates, and system activities.
- Streamlined communication to enhance the hiring process.

2.3 Operating Environment

The RMS is designed to operate within an institutional or corporate IT infrastructure. It is expected to run as a web-based application accessible from different devices (desktops, laptops, mobile devices) via standard browsers. The backend system relies on database management

for storing user data, job postings, and application records. The environment must support internet access and secure login protocols to ensure data protection and smooth system operations for all users.

2.4 User Characteristics

1. Students:

- Job seekers with basic knowledge of digital platforms.
- Apply for jobs and track the status of their applications.
- Update personal and academic details.

2. Administrators:

- HR professionals or university staff.
- Manage student and company records.
- Oversee and monitor the recruitment process.

3. Companies:

- Recruiters or employers.
- Post job listings and manage open positions.
- Review applications and communicate with candidates.

2.5 Design and Implementation Constraints

The system must ensure data integrity, security, and scalability. Constraints include the need for a robust user authentication mechanism, efficient handling of large datasets (e.g., job applications), and secure storage of sensitive user information. Additionally, the system must support concurrent users, handle multiple job applications at once, and integrate with external systems such as email services for notifications and updates.

2.6 User documentation

Comprehensive user documentation will be provided for all types of users. This includes guides for students on how to search for jobs, apply, and manage their profiles. Admins will have documentation explaining how to manage student and company records, post jobs, and oversee the recruitment process. Companies will be guided on how to post jobs, view applications, and schedule interviews. Troubleshooting FAQs and system maintenance manuals will also be available.

2.7 Assumptions and Dependencies

The system assumes that users (students, admins, and companies) have access to stable internet connections and standard devices for accessing the web application. It is also dependent on reliable hosting services for seamless operation and may require integration with third-party systems (e.g., email servers, data storage platforms) for notifications and backups. Security protocols such as HTTPS and data encryption are assumed to be in place for protecting user data.

3.External Interface Requirements

3.1 User Interfaces

The Recruitment Management System (RMS) offers three primary user interfaces:

Student Interface: Provides features such as job search, application submission, and status tracking. The UI is intuitive, with clear navigation for viewing available jobs, updating personal details, and applying for positions.

Admin Interface: Admins have a dashboard where they can manage students and companies, post updates, and review system logs. The interface includes options for adding, editing, and deleting student and company profiles.

Company Interface: Designed for recruiters, this interface allows companies to post job listings, view applications, and call for interviews. Job management tools are accessible, enabling recruiters to manage job postings efficiently. All interfaces are responsive, allowing use across desktops, tablets, and mobile devices.

3.2 Hardware Interfaces

Servers: The system requires hosting on cloud or local servers to manage databases, authentication, and application processing.

Client Devices: Students, admins, and companies can access the system via PCs, laptops, tablets, or smartphones that support modern web browsers.

Database Servers: A database server is needed for storing user information, job applications, and associated data, ensuring efficient retrieval and secure data management. The system is designed to support basic hardware requirements with no need for specialized equipment.

3.3 Software Interfaces

The system interacts with several software components:

Web Browsers: The user interfaces are accessible through web browsers such as Chrome, Firefox, and Edge, ensuring cross-platform compatibility.

Database Management Systems (DBMS): The backend integrates with databases such as MySQL or MongoDB to store and retrieve user details, job postings, and application records.

Authentication System: The login functionality interacts with a secure authentication module (OAuth or custom authentication) to verify user credentials.

Email Services: For notifications, the system may integrate with email APIs like SMTP, SendGrid, or similar, to notify users about job statuses, interviews, or account updates.

3.4 Communication Interfaces

HTTP/HTTPS: All client-server interactions occur over HTTP/HTTPS to ensure secure data transmission, particularly for login, job posting, and application processes.

Email Communication: The system sends automated emails to students, admins, and companies for updates on applications, interviews, and general notifications via SMTP or integrated email services.

4. System Features

4.1 Global Database

4.1.1. Description and Priority

The global database feature manages all system data, including student profiles, company information, job postings, and application records. It serves as the central repository for storing and retrieving recruitment-related data. This feature is of High Priority, as the database is integral to the entire system's functioning.

4.1.2. Stimulus/Response Sequences

- **Stimulus**: The system receives input data from students, companies, and administrators.
- **Response**: The system securely stores the data in the global database and ensures it is accessible for future actions, such as job searches and application submissions.

4.1.3. Functional Requirements

- REQ-1: The system must store all student, company, job, and application details in a secure database.
- REQ-2: The system should allow administrators to manage (add, update, delete) entries in the database.
- REQ-3: The database must be optimized for fast search and retrieval to ensure efficient operations across the system.

4.2 Login Interface

4.2.1. Description and Priority

The login interface allows different user groups (students, companies, and administrators) to securely access the system based on their roles. This is a High Priority feature since system access must be restricted to authorized users only.

4.2.2. Stimulus/Response Sequences

- **Stimulus**: A user (student, company representative, or administrator) enters their credentials on the login page.
- **Response**: The system verifies the credentials, grants or denies access based on the role, and redirects the user to the respective dashboard.

4.2.3. Functional Requirements

- REQ-4: The system must validate user credentials against the database and provide appropriate access control.
- REQ-5: The system should support different user roles (e.g., student, admin, recruiter), providing different interfaces and permissions based on the role.

- REQ-6: The system must lock out users after multiple failed login attempts and notify them with an option to reset their password.

4.3 Job Posting and Application Management

4.3.1. Description and Priority

This feature allows companies to post job listings, while students can search and apply for jobs. High Priority as it is essential to the recruitment process.

4.3.2. Stimulus/Response Sequences

- **Stimulus**: A company posts a new job listing, or a student applies to a job.
- **Response**: The system adds the job listing to the database, or records the student's application, and provides a confirmation.

4.3.3. Functional Requirements

- REQ-7: The system must allow companies to create, update, and remove job postings.
- REQ-8: The system should allow students to search and filter job listings by various criteria (e.g., location, role).
- REQ-9: The system must track all student applications and allow recruiters to review, shortlist, or reject them.

4.4 Notifications and Alerts

4.4.1. Description and Priority

The notification system sends real-time alerts to students and recruiters about job postings, application statuses, and upcoming interviews. This is a Medium Priority feature for user engagement.

4.4.2. Stimulus/Response Sequences

- **Stimulus**: A student submits an application, or a recruiter schedules an interview.
- **Response**: The system sends an email/SMS notification to the student or recruiter.

4.4.3. Functional Requirements

- REQ-10: The system must send email/SMS notifications for critical events such as job postings, application status changes, and interview scheduling.
- REQ-11: The system should allow users to configure notification preferences.
- REQ-12: The system must log all notifications sent for future reference.

4.5 Staff and Role Management

4.5.1. Description and Priority

This feature generates reports on student applications, job postings, and recruitment success rates. It helps administrators and recruiters make data-driven decisions. This is a High Priority feature for effective system management.

4.5.2. Stimulus/Response Sequences

- **Stimulus:** An admin requests a report on job applications or recruiter activities.
- **Response:** The system fetches relevant data and generates the report for the admin..

4.5.3. Functional Requirements

- REQ-13: The system must maintain detailed records for all hospital staff, including roles, work hours, and shifts.
- REQ-14: The system should generate payroll reports based on staff roles and attendance.
- REQ-15: The system must allow hospital administrators to assign and modify staff roles and work schedules.

5. Nonfunctional Requirements

5.1 Performance Requirements

The system must handle up to 1,000 concurrent users without performance degradation, ensuring smooth interaction for all users (students, recruiters, administrators). The system should respond to user requests (e.g., login, job search, application submission) within 2 seconds under normal load conditions. The database should support efficient query handling, ensuring search results are returned in less than 3 seconds for up to 100,000 job postings.

5.2 Safety Requirements

The system must ensure that no critical data (such as student or recruiter information) is lost during system crashes or failures. Automatic backups should be scheduled daily. In case of a system failure, the RMS should recover within 15 minutes, ensuring minimal disruption to ongoing recruitment activities. The system should handle any invalid inputs or actions gracefully, providing users with meaningful error messages without causing data corruption.

5.3 Security Requirements

The system must enforce role-based access control (RBAC) to ensure that only authorized users can access certain functions (e.g., only recruiters and administrators can view or modify job postings).All sensitive data, such as passwords, user details, and medical records (if applicable), must be encrypted both at rest and in transit using industry-standard encryption methods (e.g., AES-256).The system should log all user activities (such as login attempts, changes to job postings, etc.) and store these logs securely for audit purposes for at least 12 months.Multi-factor authentication (MFA) should be implemented for all administrative and recruiter accounts to enhance security.The system must comply with relevant data privacy regulations, such as GDPR or HIPAA (if applicable), ensuring user consent is obtained before collecting personal information

5.4 Software Quality Attributes

Reliability: The system must have an uptime of at least 99.9%, ensuring continuous availability during peak usage periods such as recruitment drives.

Scalability: The system should be designed to scale easily to accommodate an increasing number of users and job postings as the recruitment process grows.

Usability: The user interface should be intuitive and accessible, requiring minimal training for first-time users. It should adhere to accessibility standards (e.g., WCAG 2.1) to accommodate users with disabilities.

Maintainability: The system should be modular and well-documented to allow easy updates and bug fixes, ensuring future enhancements can be applied without disrupting current operations.

Portability: The system should be deployable on various environments, including on-premises servers and cloud platforms, ensuring flexibility in deployment strategies.

5.5 Business Rules

Each student can only apply to a maximum of 5 job postings simultaneously. Once a decision (acceptance or rejection) is made for an application, the student can apply to additional jobs. Recruiters must verify their company details (e.g., registration number, industry classification) before posting job openings to ensure only legitimate organizations use the platform. The system must charge a subscription fee to companies for premium services (such as featured job postings or access to advanced analytics). Job postings must remain active for a maximum of 30 days, after which they are automatically archived unless the recruiter chooses to renew them. All user data, including students, recruiters, and administrators, must be retained for at least 5 years from the last activity date for legal and auditing purposes.

6. Other Requirements

6.1 Regulatory Compliance

- The system must comply with applicable educational data privacy regulations, ensuring adherence to laws like:
- Family Educational Rights and Privacy Act (FERPA) for managing student data privacy.
- General Data Protection Regulation (GDPR) and other relevant privacy regulations to protect both student and staff information.
- Local Educational Policies to ensure proper data handling and reporting compliance.

6.2 Data Backup and Recovery

Regular automated backups must be performed to minimize data loss in case of system failures. A disaster recovery plan should be in place, ensuring data replication and quick restoration to guarantee continuous operations of the classroom management system.

6.3 User Accessibility

The system should adhere to WCAG 2.1 accessibility standards to ensure usability for individuals with disabilities.This includes support for screen readers, keyboard navigation, and alternative text for images, making the system accessible to all, including students and teachers with disabilities.

6.4 Localization and Internationalization

The system should support multiple languages to cater to students and teachers from various regions.It should also handle different time zones and formats to ensure compatibility for global use in educational institutions with diverse needs.

6.5 Scalability

The system must be designed for scalability to manage an increasing number of students, teachers, and classrooms as institutions grow.It should maintain high performance even during peak usage periods, such as examination or enrollment times.

6.6 Session Management

The system should implement secure session management, including session timeouts and Multi-Factor Authentication (MFA) for enhanced security, especially in shared educational environments.

Appendix A: Glossary

- **FERPA**: A US federal law that protects the privacy of student education records.
- **API**: Application Programming Interface that allows seamless communication between the system and external educational tools.
- **WCAG**: Web Content Accessibility Guidelines for ensuring websites and applications are accessible to users with disabilities.

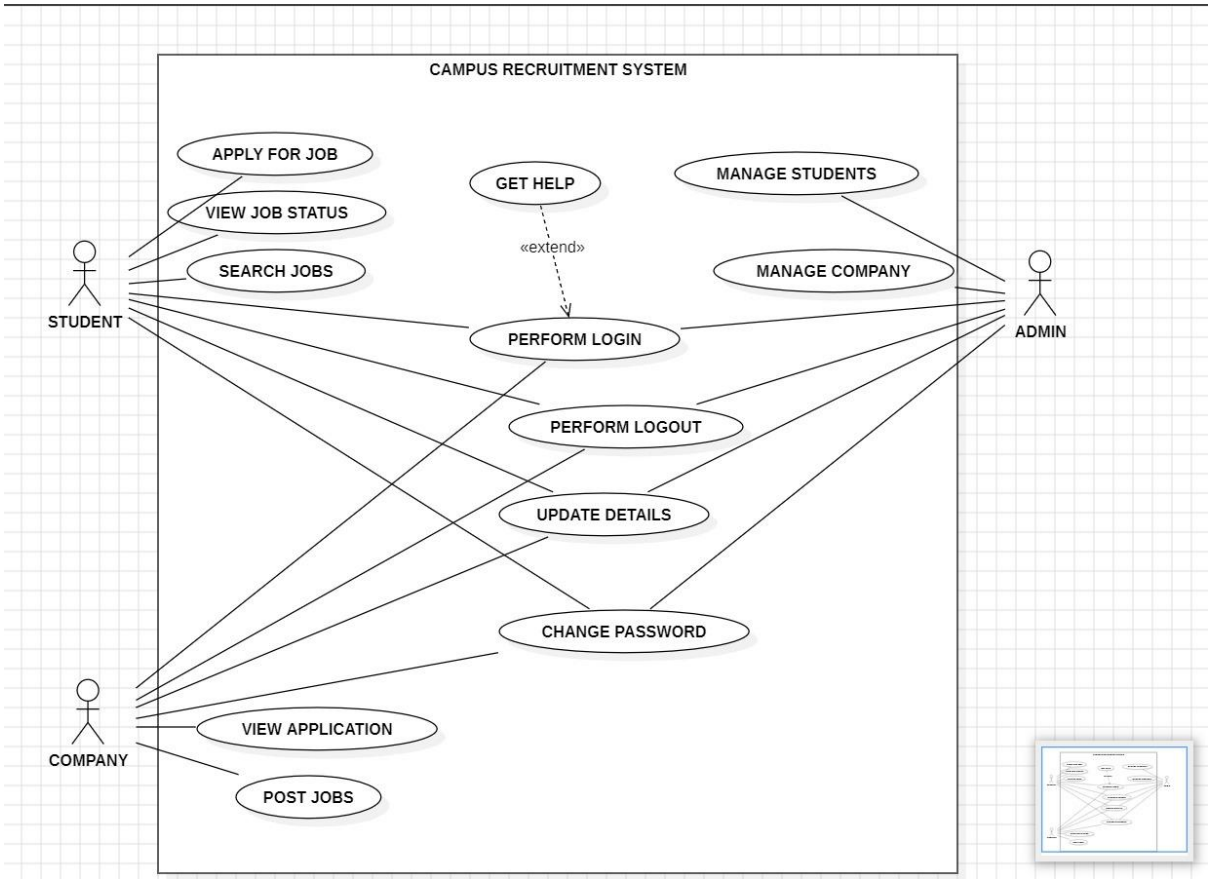
Appendix B:

Use Case Template

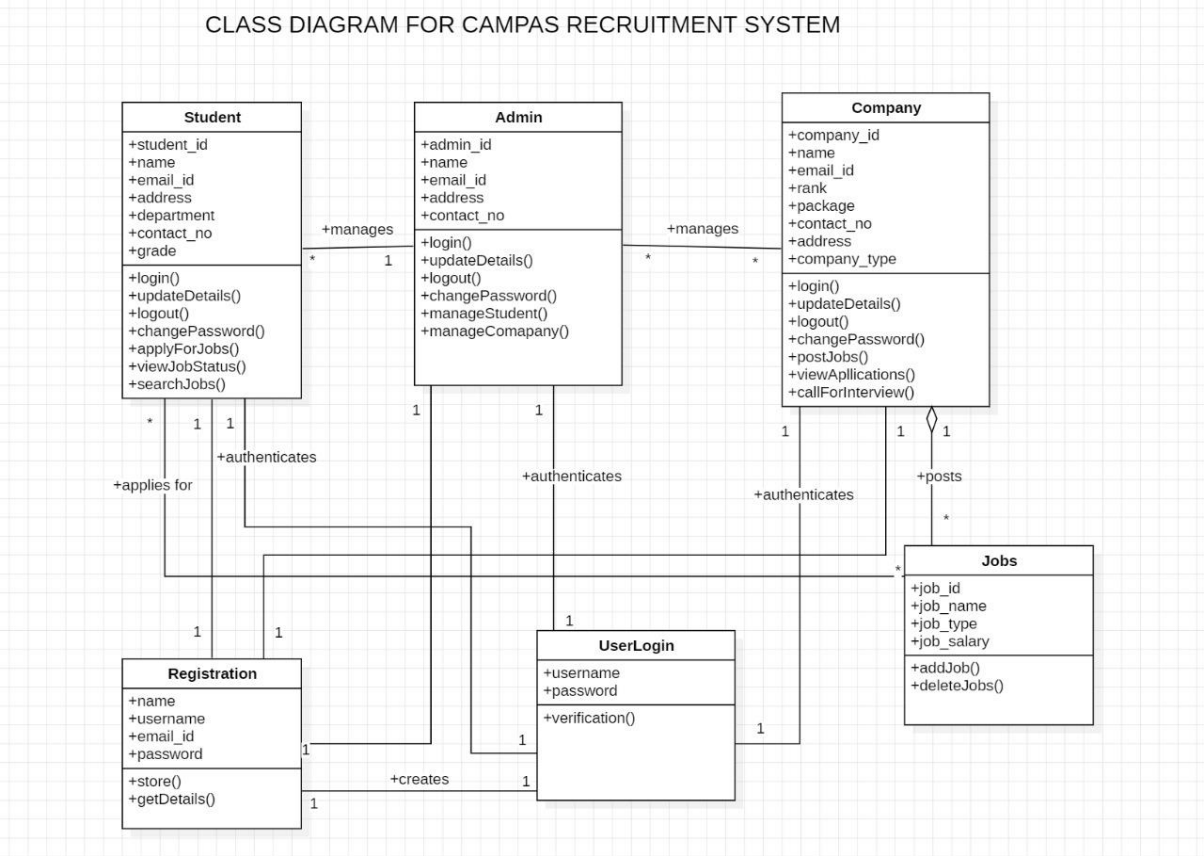
Use Case ID:	040816540236	
Use Case Name:	Recruitment System	
End Objective:	Automate and facilitate the whole process of Recruitment	
Created by:	1.B Pavan Kumar 2.B Laxmi Sai Praneeth 3.Ch Thirush Reddy 4.T Rajesh 5. K Kartheek 6. Bramha	**On (date):** 18th October 2024
User/Actor:	Student,Company and Admin	
Trigger:	Student applying for job by logging in the site	

Basic/Normal Flows	
User Actions	System Actions
User logs into the system by entering (	System verifies the entered username and password and grants access to the user.
Student searches for available jobs.	System fetches and displays job listings that match the search criteria.
Student views the status of job applications.	System retrieves and shows the application status (e.g., pending, accepted, rejected).
Student applies for a Job	System submits the job application and confirms submission to the student.
Company views student applications.	System displays the list of applicants for job postings.
Admin manages student records.	System provides tools to update, delete, or review student profiles.
Admin manages company records.	System enables the admin to add, update, or remove company profiles.
User logs out of the system.	System ends the current session and redirects to the login page.
User requests help or assistance.	System displays a help page or connects the user to customer support based on the query.

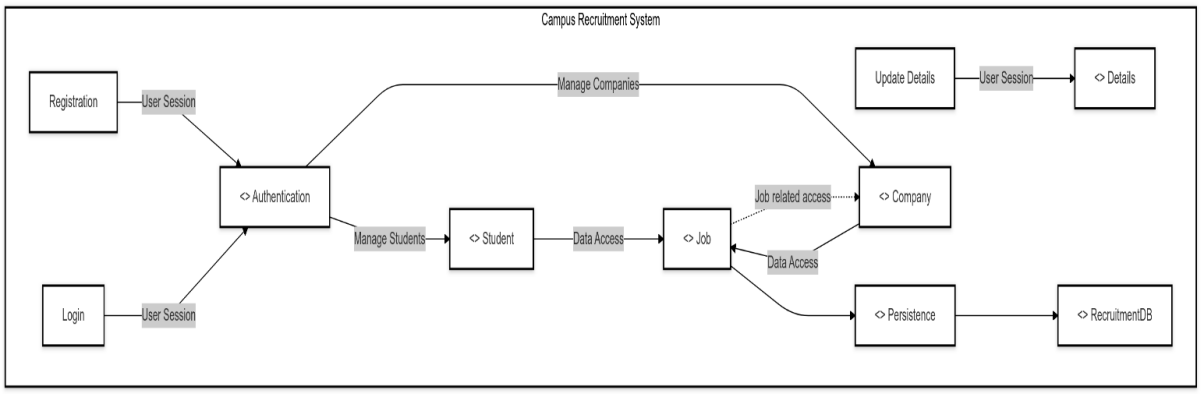
Use Case Diagram



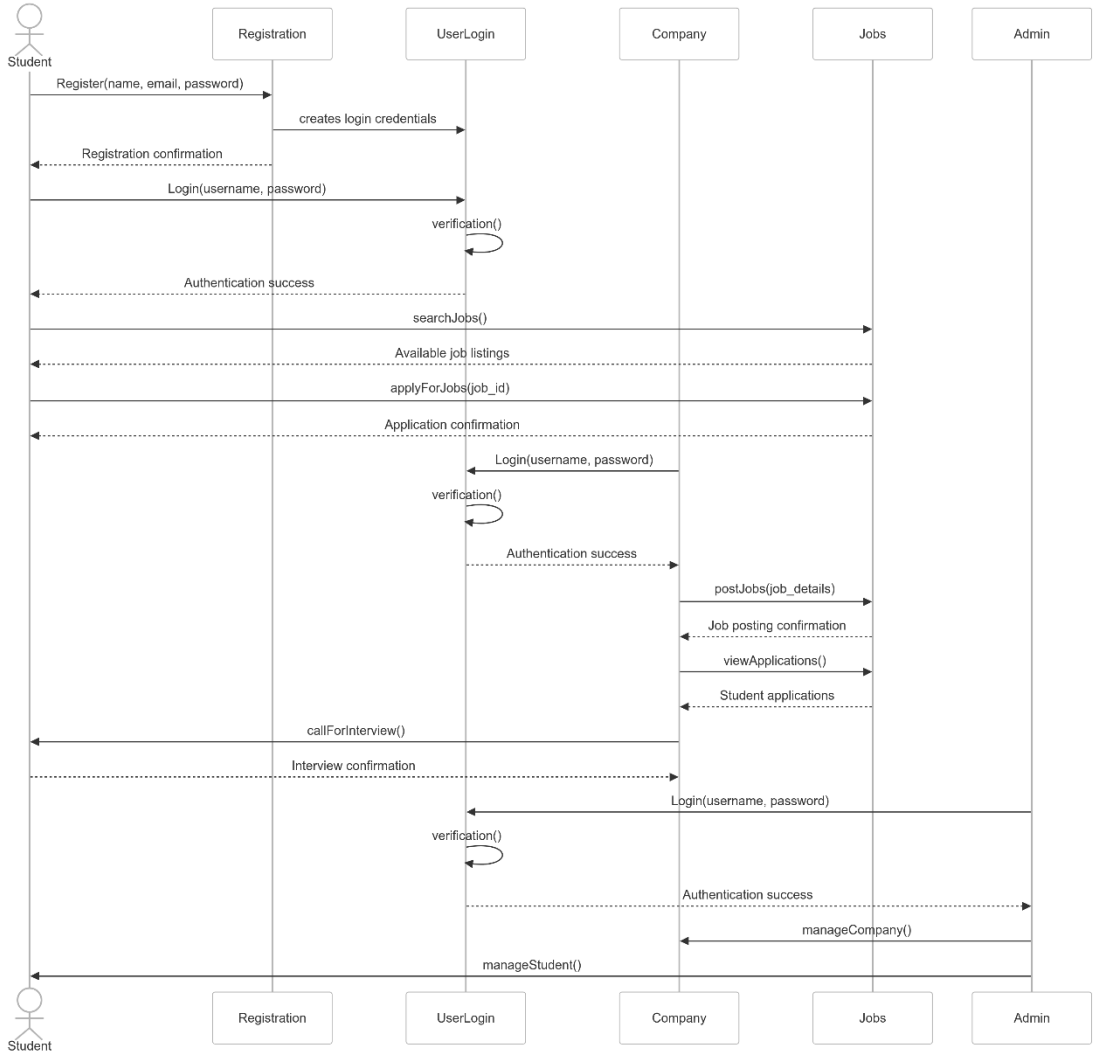
Class Diagram



Component Diagram



Sequence Diagram



Appendix C: To Be Determined List

- Finalized list of third-party educational systems and tools to integrate (e.g., LMS platforms, grading tools).
- Specific user roles and permissions for various staff (administrators, teachers, students).
- Detailed performance benchmarks for handling large classroom data, especially during peak enrollment periods.