
Software Requirements and Design Document

for

FlexJobs

Muhammad Shaif Imran (22i-1024)

Adeel Mahmood Ansari (22i-0979)

Abdul Wasay (22i-1198)

TechNova

27/11/2024



1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document outlines the requirements for **FlexJobs Student Portal**, a career and educational opportunity platform tailored specifically for FAST University students. The platform aims to centralize access to job opportunities, internships, exchange programs, scholarships, and educational programs relevant to FAST students. This document covers the features, objectives, and benefits of the platform, emphasizing its role in enhancing career growth and academic success for its target audience.

1.2 Product Scope

FlexJobs Student Portal is a web-based application that bridges the gap between students and career opportunities by consolidating resources into a single, easy-to-use interface. It is designed to cater to the specific needs of FAST University students by providing a curated list of job postings, internships, and academic programs.

Key benefits include:

- A centralized hub for job opportunities, internships, and educational opportunities.
- Tools to track applications and updates from companies.
- A registration portal for organizations to post opportunities targeted at FAST students.
- Enhanced communication between students and organizations to streamline opportunity management.

By addressing these gaps, FlexJobs aligns with the university's mission to support students' career aspirations and academic goals.

1.3 Title

FlexJobs Student Portal: Centralizing Career and Educational Opportunities for FAST University Students

1.4 Objectives

The primary objectives of the FlexJobs Student Portal are:

- To create a unified platform for FAST students to access career and academic opportunities.
- To streamline the application process for internships, scholarships, and jobs.
- To enable students to track application statuses and receive updates from companies.
- To provide organizations with tools to post opportunities and manage student applications.
- To foster career growth and reduce the chances of missed opportunities due to scattered information sources.

1.5 Problem Statement

The lack of a centralized platform for career and educational opportunities at FAST University creates significant inefficiencies for students.

Challenges include:

- Fragmented information across multiple platforms leads to missed deadlines and opportunities.
- Limited tools for tracking application statuses result in confusion and disorganization.
- Lack of a dedicated portal for organizations to target and engage FAST University students.

A successful solution is **FlexJobs Student Portal**, which will consolidate all relevant career-related information into one platform, making it easier for students to find and apply for opportunities. By providing tools for application tracking and updates, the platform will improve organization and efficiency in managing career growth.

2. Overall Description

2.1 Product Perspective

FlexJobs is a self-contained system specifically developed for FAST University. It is not part of any existing product family but is designed to address the institution's unique requirements.

FlexJobs is composed of interconnected subsystems:

1. **User Management System:** For managing user roles (students, admins, and organizations).
2. **Opportunity Management Module:** Enabling organizations to post and manage opportunities.
3. **Application Tracking Module:** Allowing students to apply for opportunities and track statuses.
4. **Notification and Communication System:** Sending updates and announcements to students.

The system integrates with:

A secure database for storing user and opportunity information.

2.2 Product Functions

FlexJobs provides the following primary functions:

User Management:

Students, admins, and organizations can log in and perform role-specific tasks.

Opportunity Management:

Organizations can post job, internship, and scholarship listings.

Application Tracking:

Students can apply for opportunities and monitor application statuses.

Notification System:

Automated notifications for application updates and new listings.

2.3 List of Use Cases

Register Student
Register Organization Representative and its Organization
Verify Organization and Organization Representative
Post Opportunities
Filter Applicants
Apply for Jobs
Track Applications
Manage Student Profile
Generate Reports
Schedule Interviews
Manage ChatBox
Filter Opportunities

2.4 Extended Use Cases

1. Register Student

Use Case Name:

Register Student

Scope:

FlexJobs

Level:

User Goal

Primary Actor:

Student

Stakeholders and Interests:

Students: Want to create an account and access job, internship, and educational opportunities.

University: Ensures only valid students register.

Preconditions:

The student must have a valid FAST University email address.

Post conditions:

The student's account is created, and they can log in to the application.

Main Success Scenario:

Actor	System
1) The student navigates to the Student Registration section.	2) The system displays the registration fields
3) The student provides the information	4) The system validates and stores the information if valid.

Extensions:

4a. The student already exists or invalid information provided, the student is prompted to re-enter the information.

2. Register Organization Representative and its Organization

Use Case Name:

Register Organization Representative and its Organization

Scope:

FlexJobs

Level:

User Goal

Primary Actor:

Organization Representative

Stakeholders and Interests:

Organizations: Want to register to post opportunities.

Admin: Needs to verify the legitimacy of organizations.

Preconditions:

The user must have valid details.

Post conditions:

The organization and organization representative registration is pending verification by the admin.

Main Success Scenario:

Actor	System
1) The Organization Representative navigates to “Register Organization Representative”.	2) The system displays the registration fields.
3) The Organization Representative writes the details.	4) The system checks if the organization is in the system or not.
	5) If yes, the organization representative gets registered.
	6) If not, the organization representative is prompted to enter Organization details.
7) The Organization Representative fill the Organization data.	8) The system checks the details and verifies it and add it to the System and return the successful message.

Extensions:

- E1: The user enters wrong information:
 - The system prompts the representative to write data in a correct format.

3. Verify Organization

Use Case Name:

Verify Organization

Scope:

FlexJobs

Level:

User Goal

Primary Actor:

Admin

Stakeholders and Interests:

Admin: Needs to ensure that only legitimate organizations are allowed to post opportunities.

Organizations: Want to have their account verified to start posting opportunities.

Preconditions:

The organization must have completed the registration process, and the admin must have proper authorization to perform the verification.

Post conditions:

The organization is either verified, rejected, or asked for additional information. This allows the organization to proceed (or not) with posting opportunities.

Main Success Scenario:

Actor	System
1. The admin logs into the verification dashboard.	2. The system displays a list of pending verifications.
3. The admin selects an organization to verify.	
4. The admin reviews submitted information (e.g., legal documents, registration data).	
5. The admin either approves or rejects the organization and its representative.	6. The system updates the organization's status based on the admin's decision.

4. Post Opportunities

Use Case Name:

Post Opportunities

Scope:

FlexJobs

Level:

User Goal

Primary Actor:

Verified Organization Representative

Stakeholders and Interests:

Organizations: Want to post job, internship, and educational opportunities for students.

Preconditions:

The organization and the organization representative must be verified and logged in.

Post conditions:

The opportunity is posted and visible to eligible students.

Main Success Scenario:

Actor	System
1. The representative go to "Post New Opportunity" section. 3. The representative enters the opportunity details (title, description, requirements, etc.). 4. The representative specifies eligible departments/majors.	2. The system displays the Post opportunity fields. 5. The system check the details and if correct it stores it in the system and displays the successful message

Extensions:

- E1: Missing information:
 - The system prompts the representative to complete missing fields before posting.

5. Filter Applicants:**Primary Actor:**

Company/Organization Representative

Scope:

FlexJobs Student portal

Level:

User goal

Stakeholders and Interests:

- **Company Representative:** Wants to find qualified students based on relevant filters (e.g., skills, GPA, experience).
- **Students:** Want to ensure their profile is easily searchable by companies based on relevant skills and academic performance.
- **Admin (University):** Ensures the system operates efficiently and helps match students with employers.

Preconditions:

- The company must be registered and logged into the system.
- Applicant profiles must be complete with academic and professional details for filtering.
- The company should have posted job/internship opportunities.

Postconditions:

The company successfully filters and reviews qualified applicants based on the set criteria.

Main Success Scenario (Basic Flow):

Actor	System
1. The company representative wants to filter applicants who applied for the job on FlexJobs. 3. The company representative selects filter criteria. 5. The company representative reviews the filtered list of students. 6. The representative can now select and contact qualified students for the next steps.	2. The system presents the representative with a list of applicants who applied for the job. 4. The system filters the applicants according to the chosen criteria.

Extensions (Alternate Flows):

3a. No Filtered Applicants Found:

- The system informs the company that no applicants meet the set criteria.
- The company can adjust the filter or expand the search to more general criteria.

5a. Incomplete Student Profiles:

- The system notifies the company if any profiles are incomplete or missing key details.
- The company can choose to notify students to update their profiles or disregard incomplete profiles.

Special Requirements:

- The filtering process should be fast, with results provided within seconds.
- Filters should include GPA, skills, graduation year, and relevant work experience.

6. Apply for Jobs:**Primary Actor:**

Student

Scope:

FlexJobs Student portal

Level:

User goal

Stakeholders and Interests:

- **Students:** Want a straightforward way to find and apply for opportunities.
- **Companies:** Want students to easily apply for posted positions.

Preconditions:

- a) The student must be registered and logged in.
- b) The student's profile must be complete with relevant academic and professional details.
- c) The company must have posted job or internship openings on the portal.

Post conditions:

The student has successfully applied for the desired job or internship, and the application has been submitted to the company.

Main Success Scenario:

Actor	System
1) The student picks a job and views its description.	2) The system retrieves the job description and displays it to the user.
2) The student chooses to apply.	3) The system checks user validity, submits application and returns successful message.
	4) The system sends the application to the relevant organization.

3a. The user has already submitted an application: The system displays ineligibility message to the user.

4a. The application submission was unsuccessful: The user is notified of the error.

Special Requirements:

1. The job listings should be easy to navigate.
2. Application submission should be simple, with confirmation messages provided to the student.

7. Track Applications:**Primary Actor:**

Student

Scope:

FlexJobs Student portal

Level:

User goal

Stakeholders and Interests:

- o **Students:** Want to track the progress of their job or internship applications easily.
- o **Companies:** Want to provide updates to students regarding the status of their applications.

Preconditions:

1. The student has successfully applied for a job or internship through the portal.

Post conditions:

The students can track the status of their applications.

Main Success Scenario (Basic Flow):

Actor	System
1) The student navigates to the Track Applications section.	2) The system displays the list of current applications and their status (In progress, rejected, accepted, called for interview).
3) The student chooses an application to view details.	4) The system displays the application specifics.

4a. **Application in Progress:** The system displays the current status as "In Progress."

4b. **Application Accepted:** The system displays the current status as "Accepted."

4c. **Application Rejected:** The system displays the current status as "Rejected" along with feedback provided by the employer.

4d. **Called for Interview:** The system displays the current status as "Called for Interview." The system shows the interview details, including date, time, location, and any additional instructions.

Special Requirements:

1. The portal should send automated notifications to students when there is a status update on their application.
2. The system should clearly display all stages of the application process.

8. Manage Student Profile:**Primary Actor:**

Student

Scope:

FlexJobs Student portal

Level:

User goal

Stakeholders and Interests:

- **Students:** Want to easily update and maintain their profiles with relevant information.
- **Companies:** Want to view detailed and accurate student profiles when filtering and reviewing applicants.

Preconditions:

1. The student must be logged in.
2. The student's profile must initially contain basic information provided during registration.

Post conditions:

The student has successfully updated their profile the latest information.

Main Success Scenario (Basic Flow):

Actor	System
1) The student navigates to the Manage Profile section.	2) The system displays the editable fields
3) The student provides the new information(password, cgpa, resume, department, semester).	4) The system validates and stores the information in the database if valid.
	5) The system displays the successful/unsuccessful update message to the student.

Extensions:

4a. The student enters invalid information and is prompted to enter again.

Special Requirements:

1. The profile management system should be easy to navigate and user-friendly.
2. Profiles should automatically be flagged if critical information is missing, prompting the student to complete their profile.

9. Generate Reports:**Primary Actor:**

Admin

Scope:

FlexJobs

Level:

User Goal

Stakeholders and Interests:

- **University Administration:** Needs data for analysis and decision-making.
- **Department Heads:** Require placement and application statistics.
- **Career Office:** Needs performance metrics.

Preconditions:

1. User must have administrative privileges.
2. Sufficient data must exist in the system.

Post conditions:

1. The requested report is generated and available for viewing or download.

Main Success Scenario:

Actor	System
1. Admin goes to the Generate Report section. 2. The admin selects the entity for which report will be generated (e.g. student, organization, organization representative). 4. The admin receives the information and agrees to it. 6. The admin previews the generated report to check its content.	3. The system generates the required report and prompts the admin of the path where the report is saved.

Extensions:**1) Report generation timeout:**

The system notifies the admin of a report generation timeout.

10. Schedule Interviews

Scope:

FlexJobs

Level:

User Goal

Primary Actor:

Organization Representative

Stakeholders and Interests:

- **Organizations:** Want to efficiently schedule interviews with candidates.
- **Students:** Need clear interview schedules and details.
- **University Career Office:** Requires an overview of interview activities.

Preconditions:

1. The organization must have shortlisted candidates.
2. Students must have applied to opportunities.

Post conditions:

1. The interview is scheduled, and both parties are notified.

Main Success Scenario:

Actor	System
1. The representative navigates to Schedule Interview section	2. The system displays the fields.
3. The representative selects the application for the interview, interview type, time and date.	

	4. The system saves the interview details and sends notifications.
--	--

Extensions:

4) Schedule conflict:

- The system notifies the representative of scheduling conflicts.
- The system suggests alternative time slots to resolve conflicts.

11. Manage Chat Box:

Scope:

FlexJobs

Level:

User Goal

Primary Actor:

Student, Organization Representative

Stakeholders and Interests:

- **Students:** Want to communicate with organizations.
- **Organizations:** Need to interact with potential candidates.
- **University Administration:** Requires oversight of communications.

Preconditions:

- Both parties must be registered users.
- Connection must be established.

Post conditions:

- Messages are exchanged and stored in the system.

Main Success Scenario:

Actor	System
1. The user accesses the chat interface to view active conversations.	2. The system loads the chat interface, displaying the list of active conversations.
3. The user selects the desired conversation to chat.	4. The system loads the chat history or initiates the conversation.
5. The user composes and sends a message.	6. The system delivers the message in real-time.

Extensions:**4) User is offline**

- The system stores the message.
- The system delivers the message when the user comes online.

12. Filter Opportunities:**Scope:**

FlexJobs

Level:

User Goal

Primary Actor:

Student, Organization Representative

Stakeholders and Interests:

- **Students:** Need relevant opportunities to further their careers (jobs, internships, scholarships, exchange programs).
- **Organizations:** Want candidates to discover and apply for their roles.
- **University Administration:** Requires data on student engagement with opportunities.

Preconditions:

- The user must be authenticated.

Actor	System
1) The student goes to the Opportunities section. 3) The user selects the category of interest: 3a. If Jobs, see Jobs section 3b. If Educational, see Educational Opportunities Section.	2) The system displays the categories, i.e. Jobs and Educational Opportunities.

Extensions:

Jobs Section:

Actor	System
2) The student may select a category based on their requirement. 4) The user selects the opportunity and calls “Apply for Jobs” use case.	1) The system displays a list of available jobs 3) The system retrieves and display the filtered jobs.

1a, 3a: No Job Opportunities available: The system displays an error message to the user.

Educational Section:

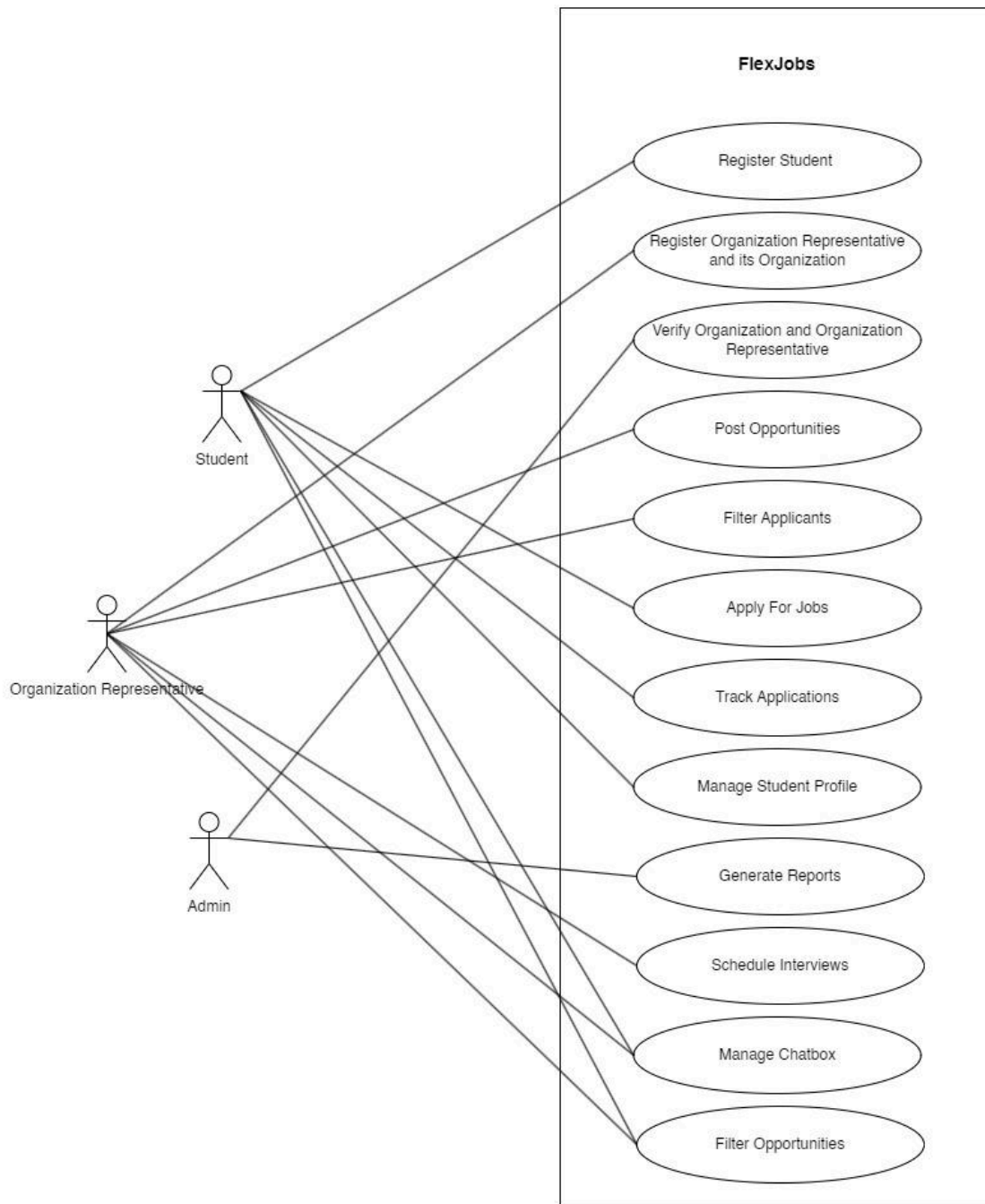
Actor	System
2) The student views a particular educational opportunity details.	1) The system displays a list of available Educational Opportunities. 3) The system retrieves and displays the relevant information.

1a: No Educational Opportunities available: The system displays an error message to the user

Postconditions:

- For jobs, the user submits an application.
- For scholarships and exchange programs, the user is provided with information

2.5 Use Case Diagram



3. Other Nonfunctional Requirements

3.1 Performance Requirements

- The system should support at least 500 concurrent users without performance degradation.
- Responses to user actions (e.g., applying for a job) should occur within 2 seconds.

3.2 Safety Requirements

- All user data must be encrypted during storage and transmission to prevent unauthorized access.
- The platform should include measures to prevent data loss in case of a system failure.

3.3 Security Requirements

- User authentication through secure login credentials.
- Role-based access control to ensure only authorized users can access specific features.

3.4 Software Quality Attributes

- **Usability:** The interface should be intuitive and accessible for all users.
- **Maintainability:** The system should allow for seamless updates and bug fixes.
- **Reliability:** The platform should ensure 99.9% uptime.

3.5 Business Rules

- Only verified organizations can post opportunities.
- Students must be enrolled at FAST University to register on the platform.

3.6 Operating Environment

- The system is a Java-based application designed to run on personal computers.
- It will operate on the Windows, macOS, and Linux operating systems, ensuring cross-platform compatibility.
- The application requires a Java Runtime Environment (JRE) version 11 or higher.

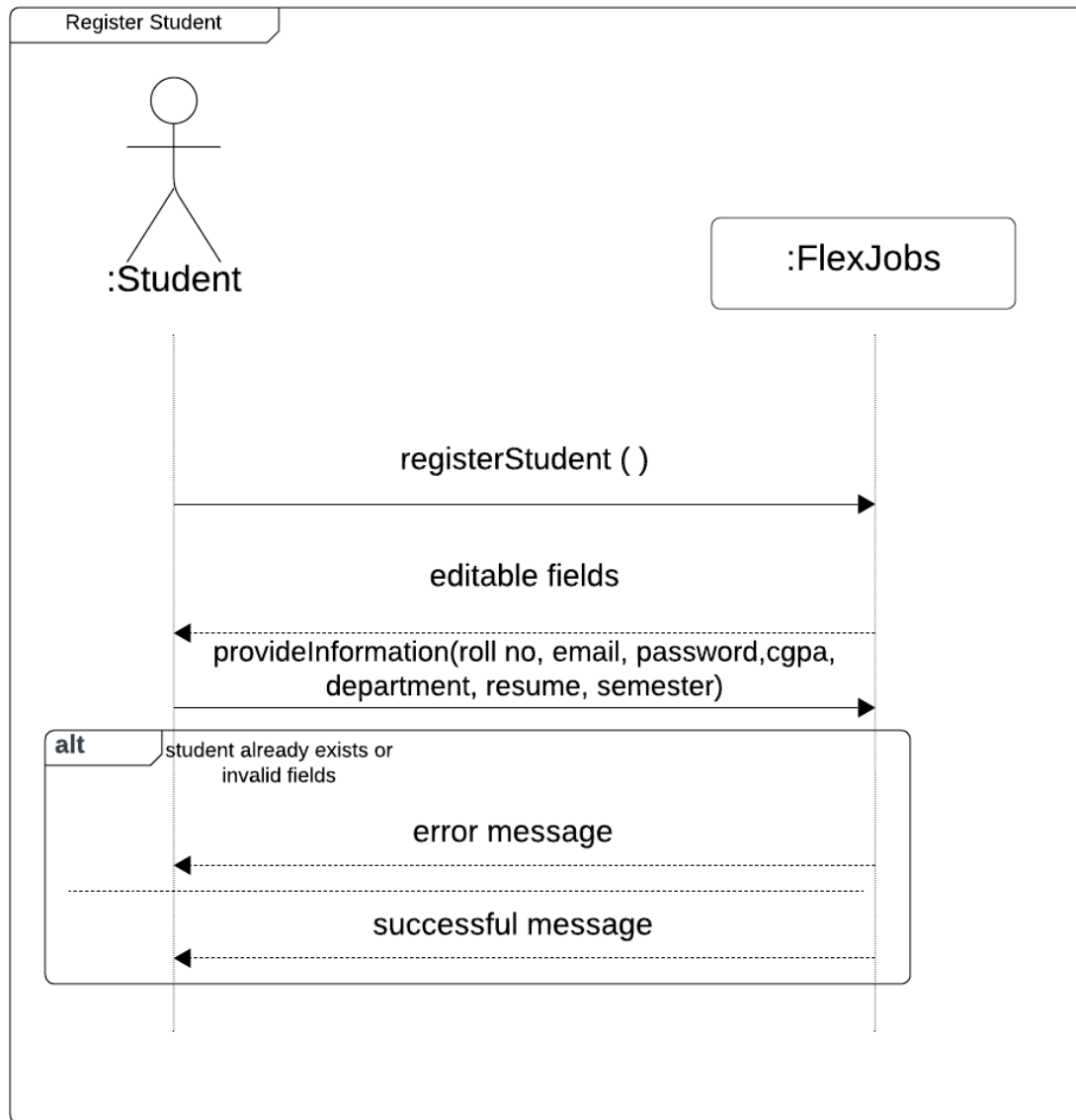
3.7 User Interfaces

- A dashboard for students showcasing opportunities, application statuses, and updates.
- A submission portal for organizations to create and manage postings.
- Admin tools for monitoring system activity and managing user roles.

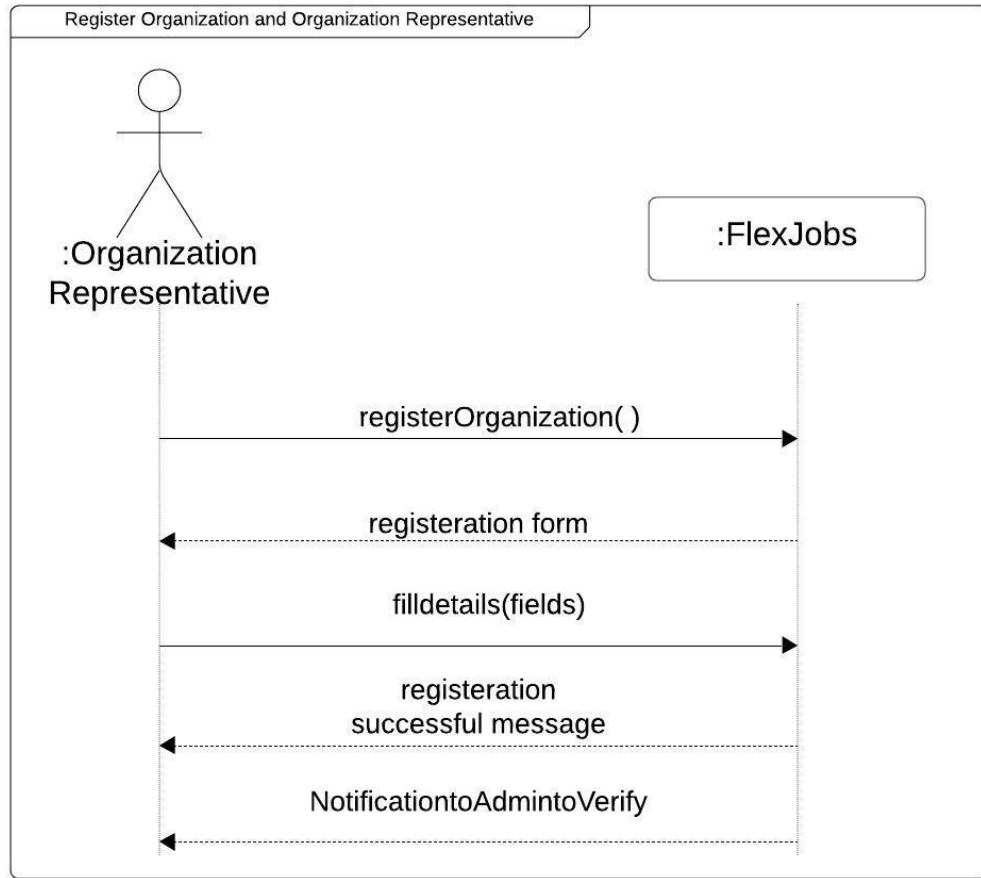
4. Domain Model

5. System Sequence Diagram

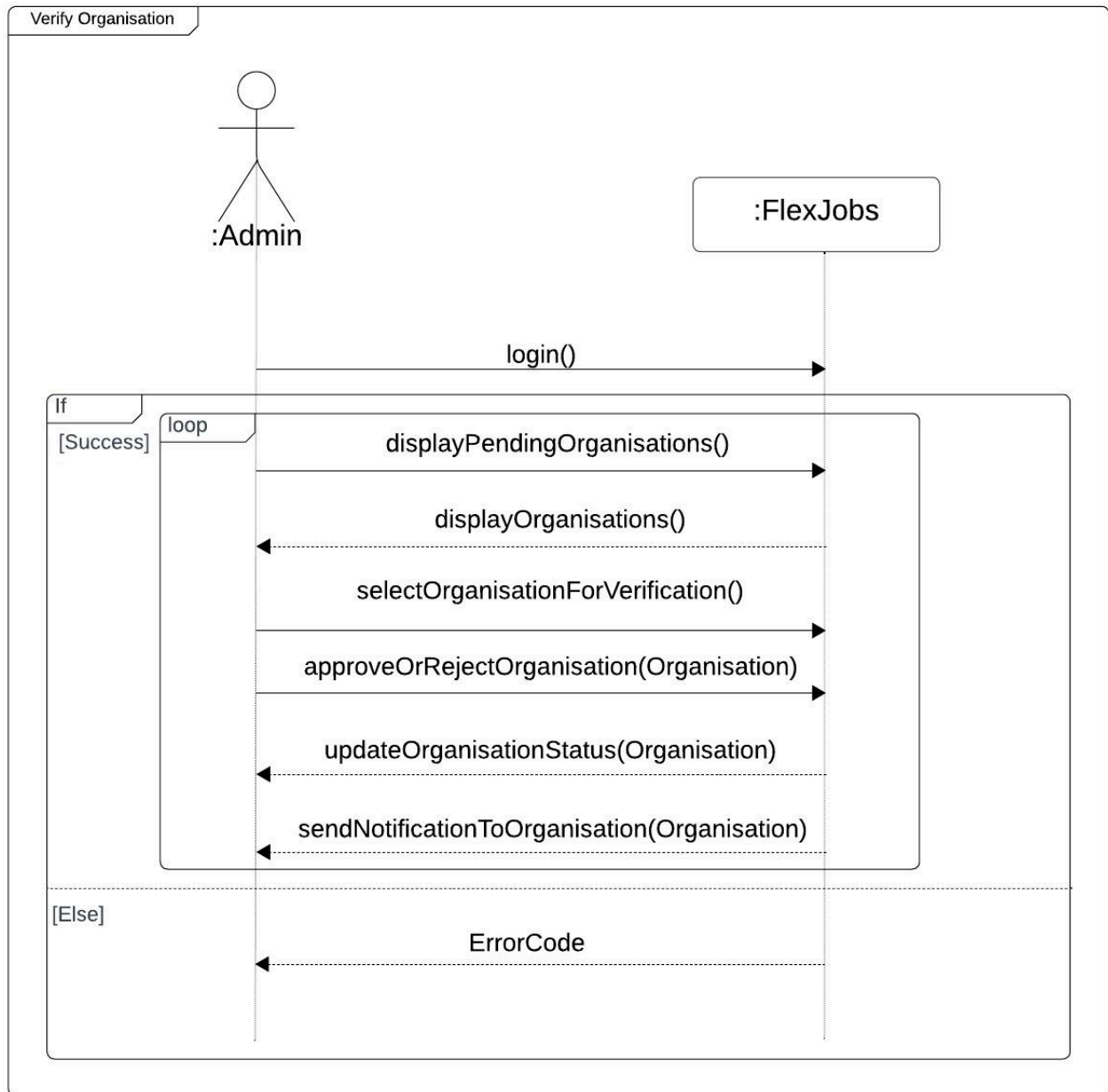
Register Student:



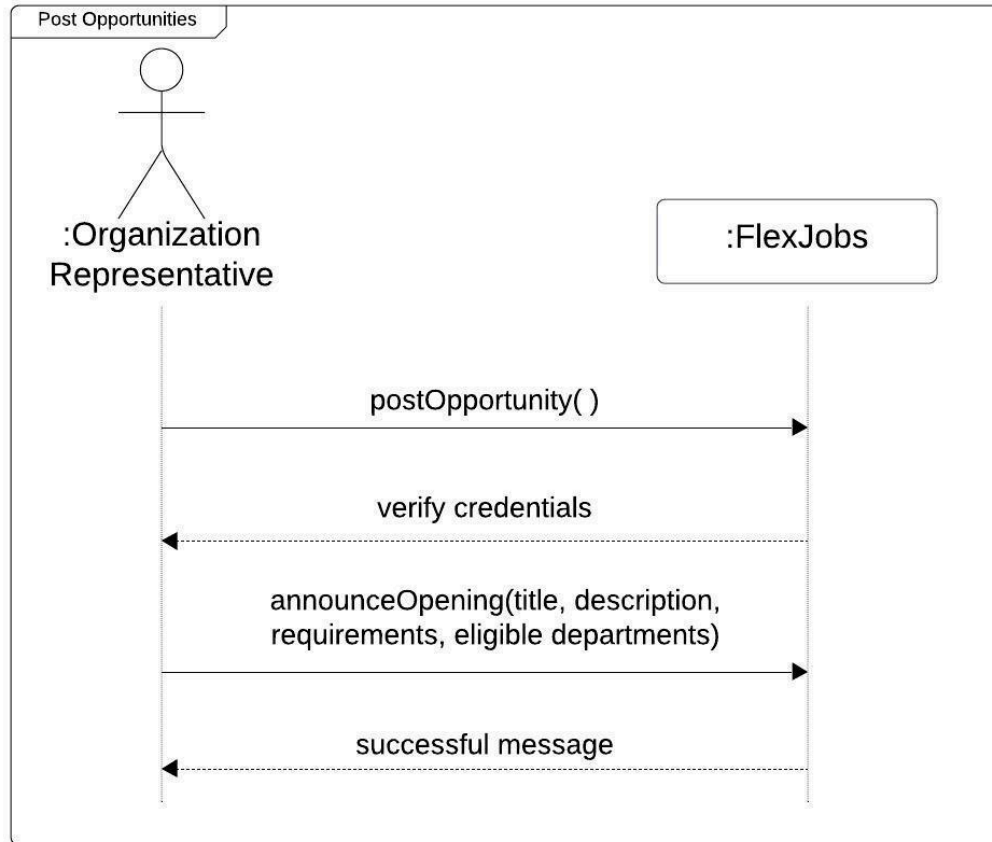
Register Organization and Organization Representative:



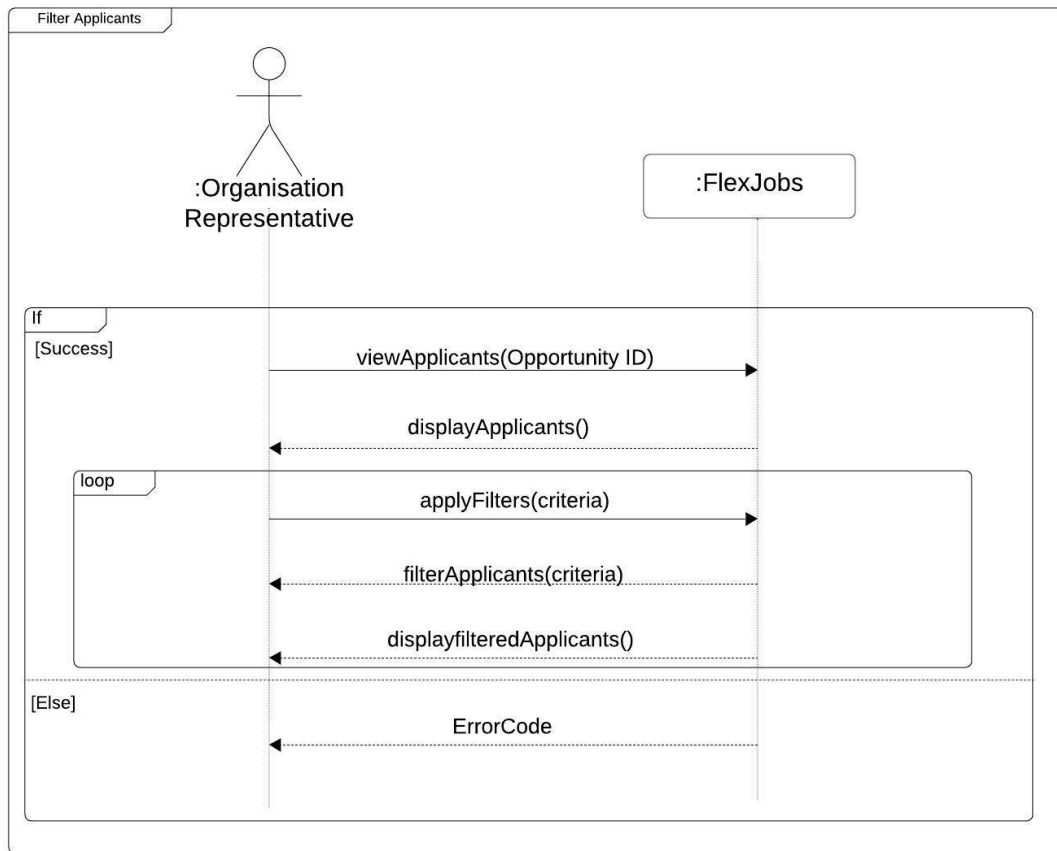
Verify Organization and Organization Representative:



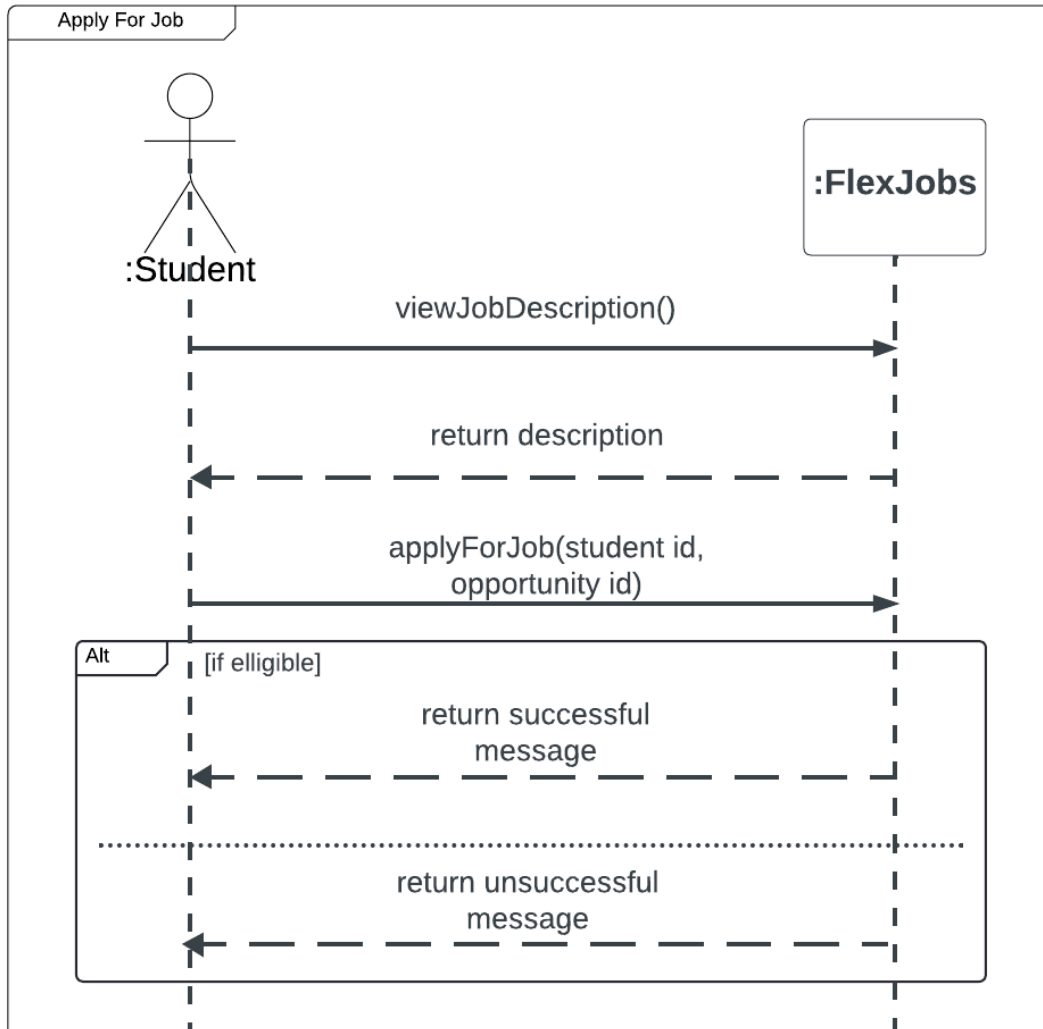
Post Opportunities:



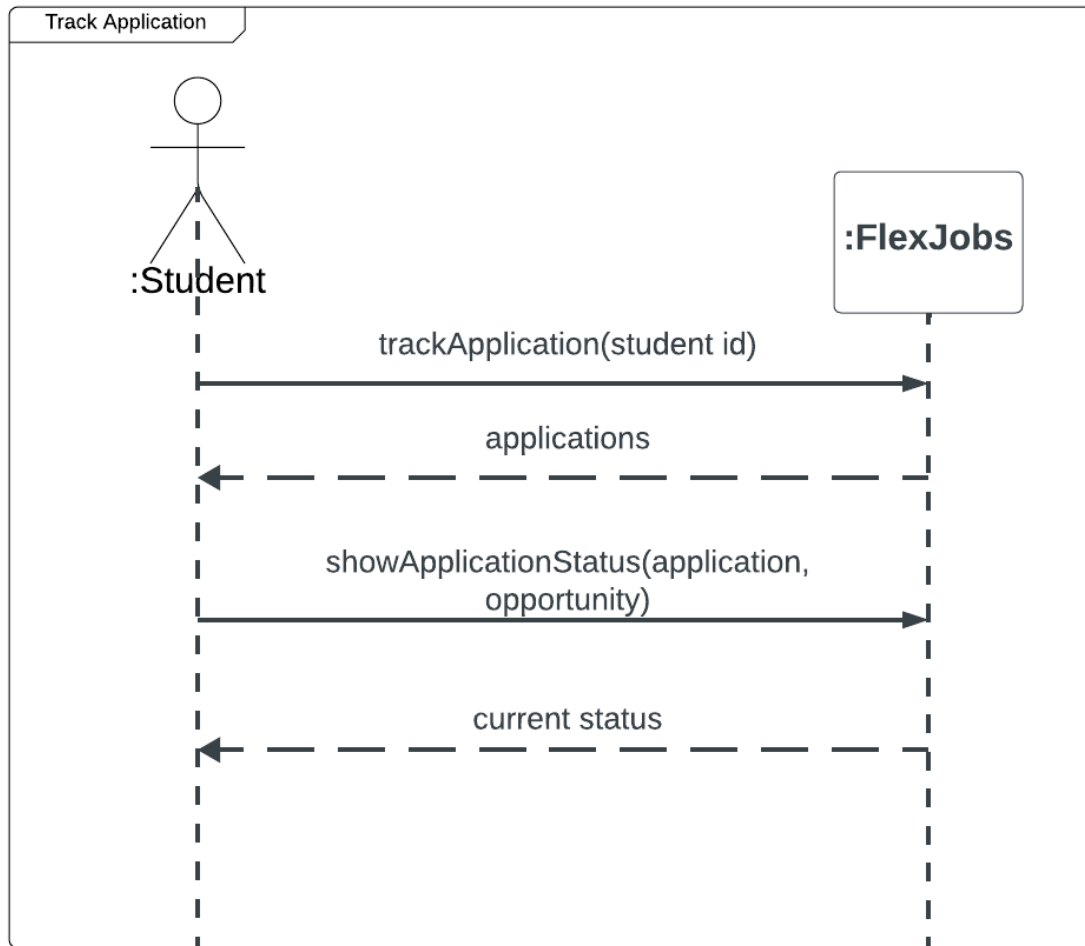
Filter Applicants:



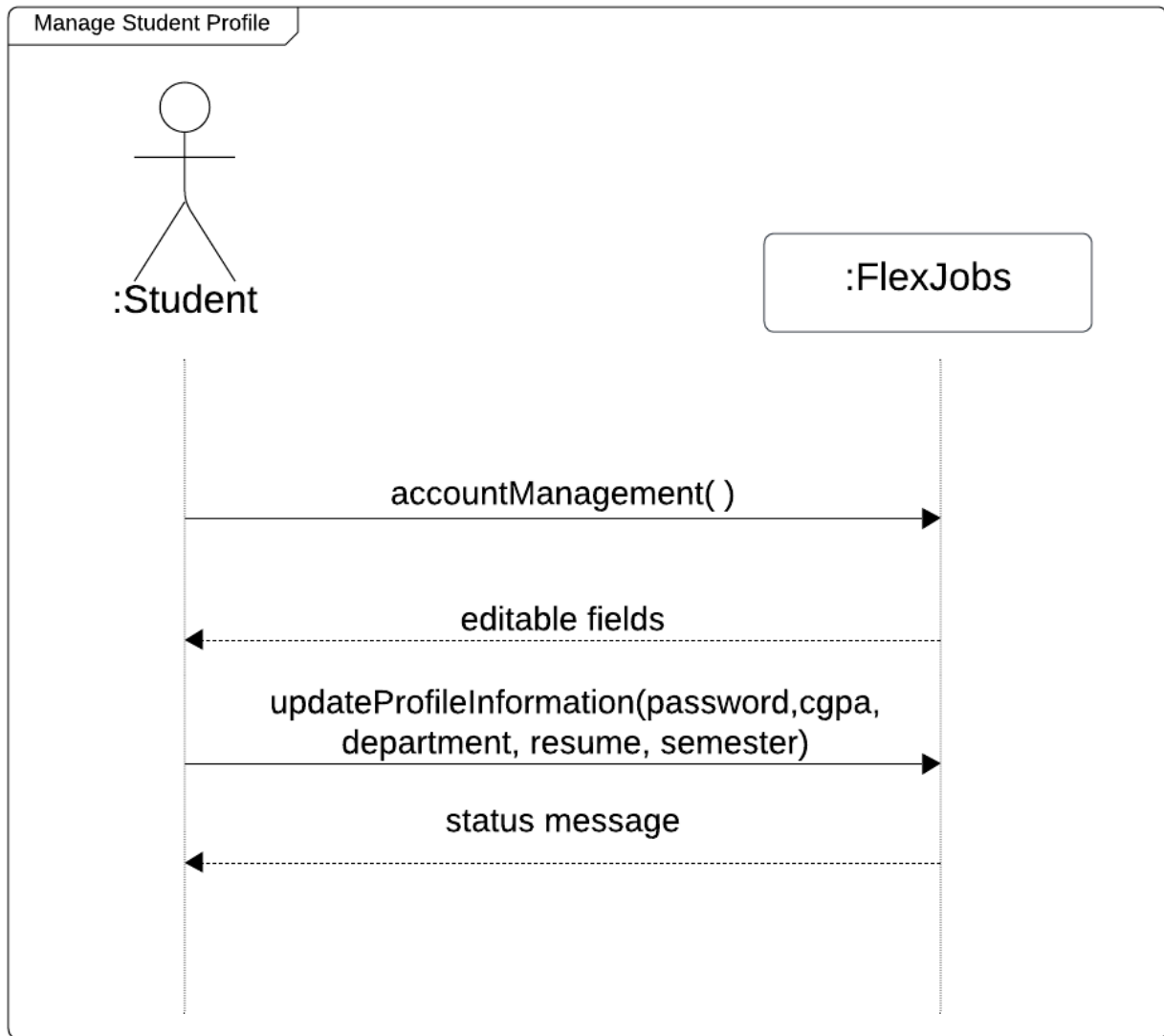
Apply for Jobs:



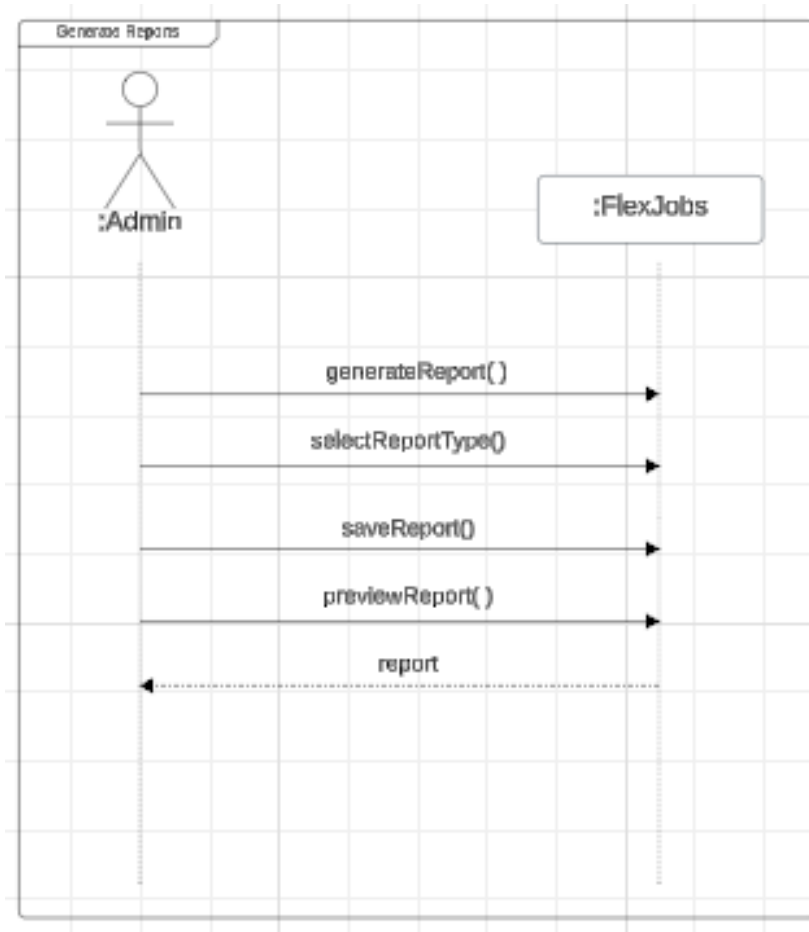
Track Applications:



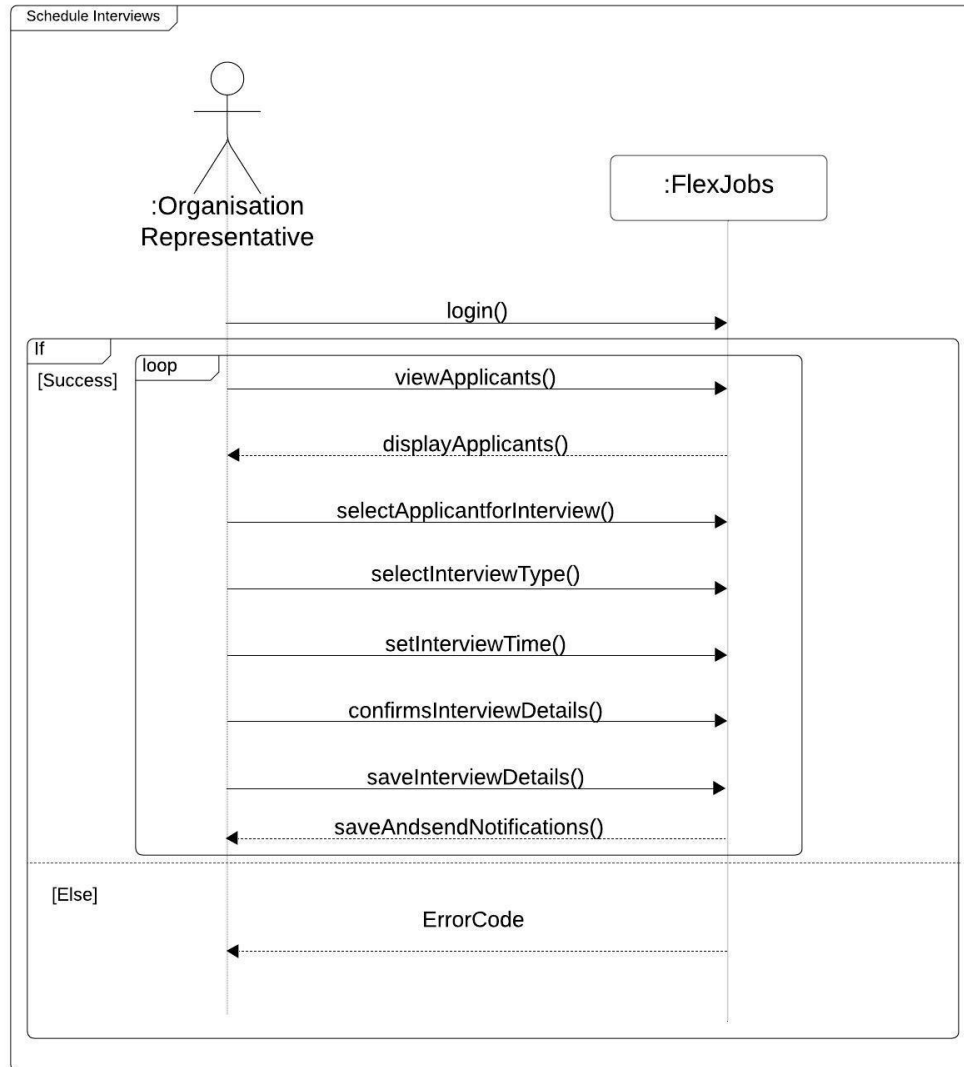
Manage Student Profile:



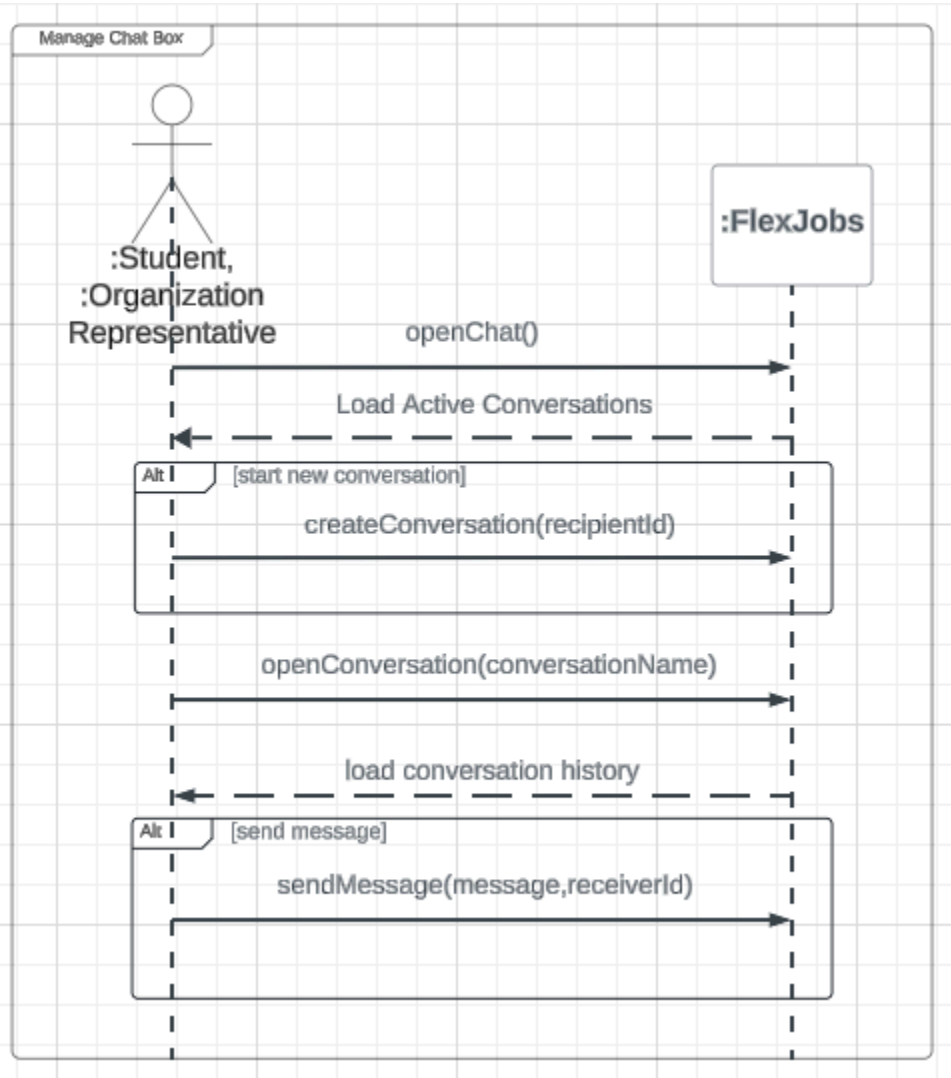
Generate Reports:



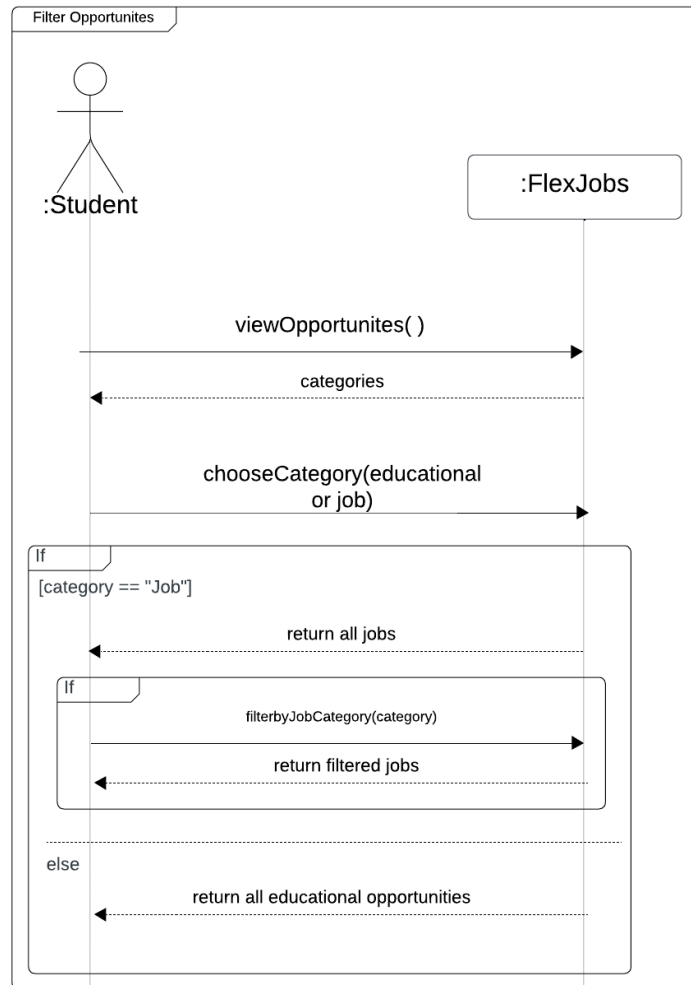
Schedule Interview:



Manage ChatBox:

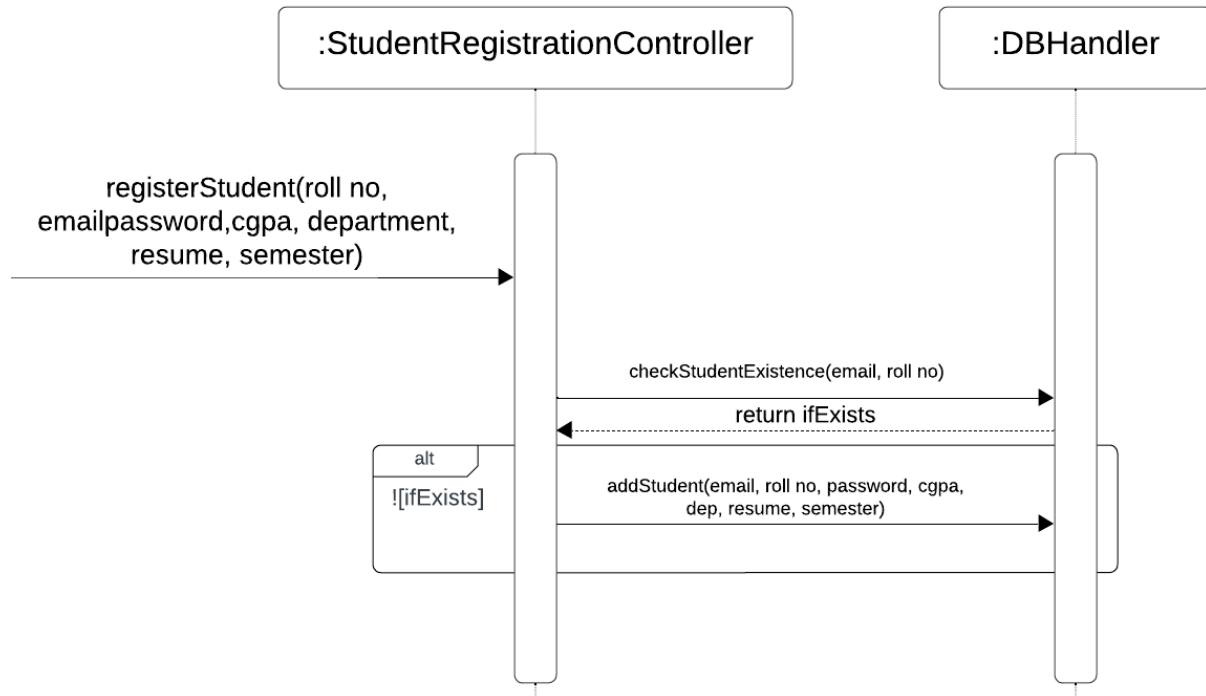


Filter Opportunities:



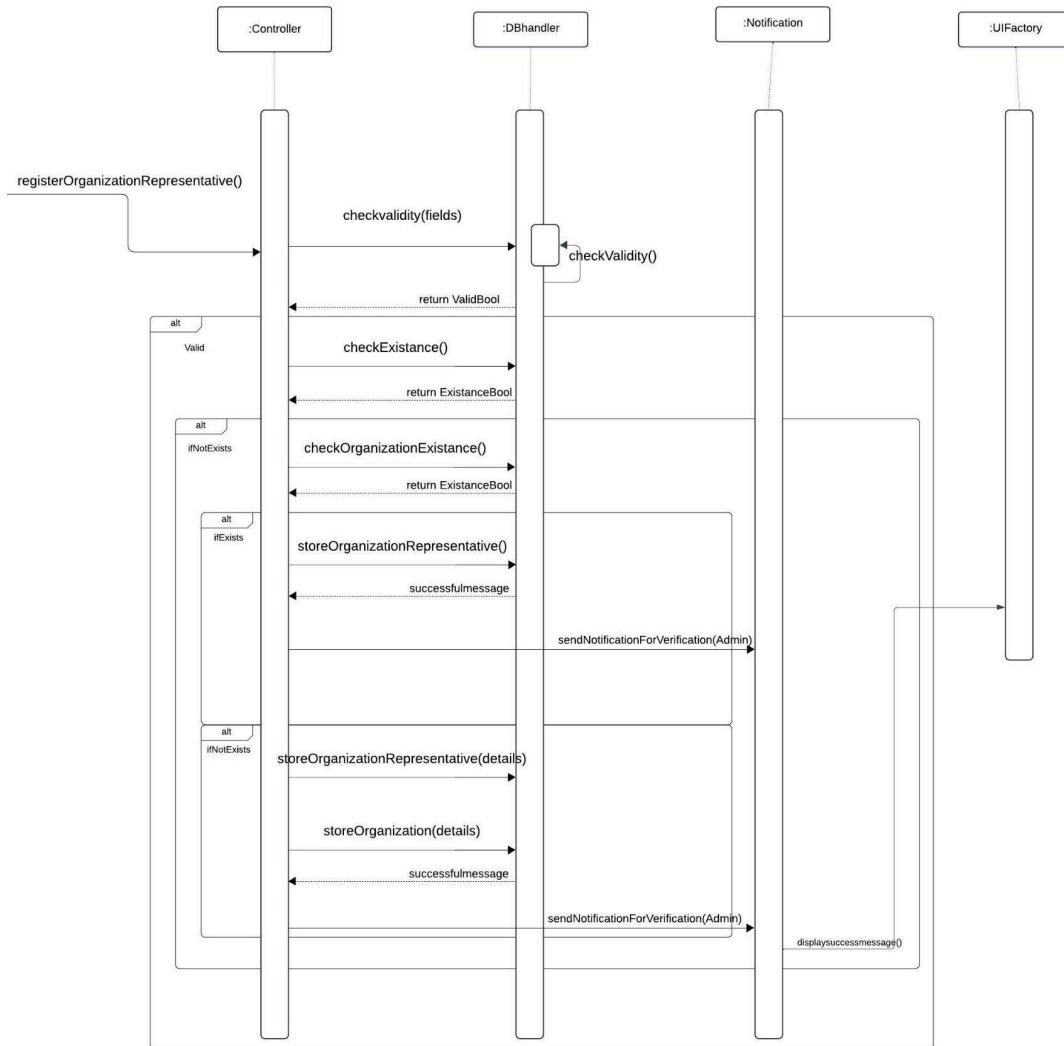
6. Sequence Diagram

Register Student:

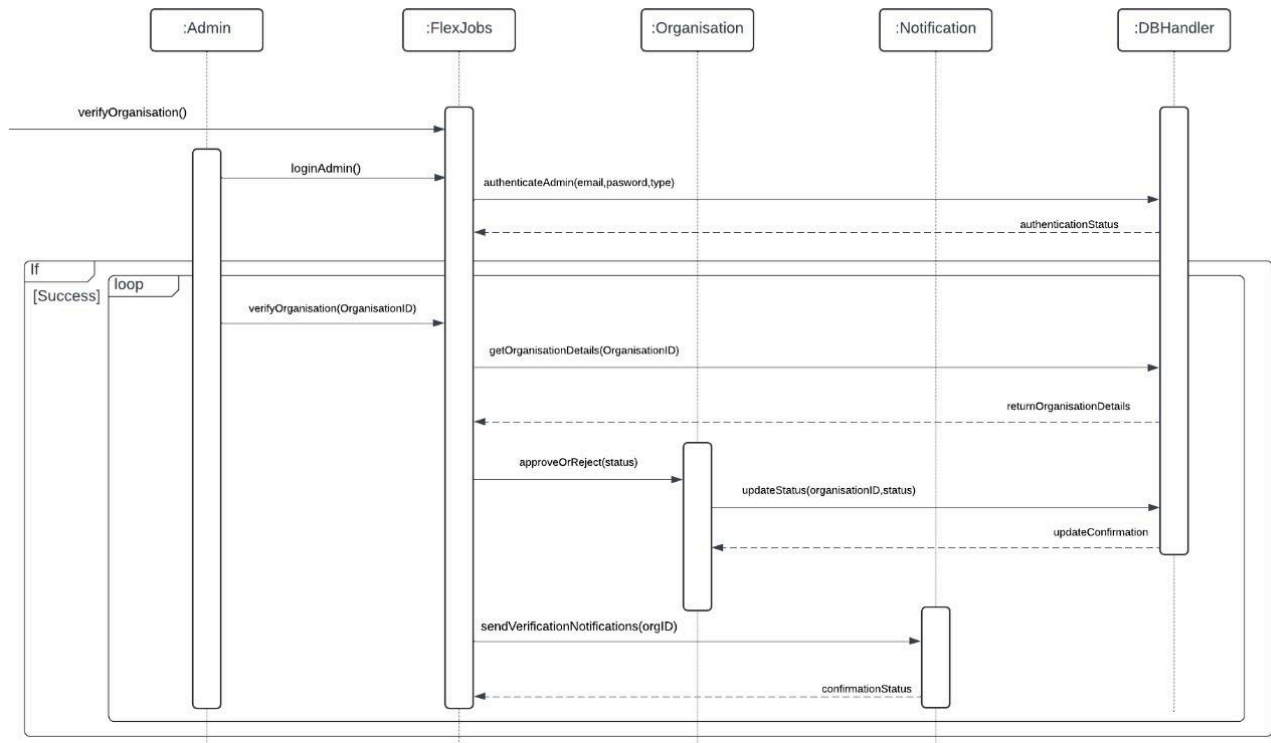


Register Organization and Organization Representative:

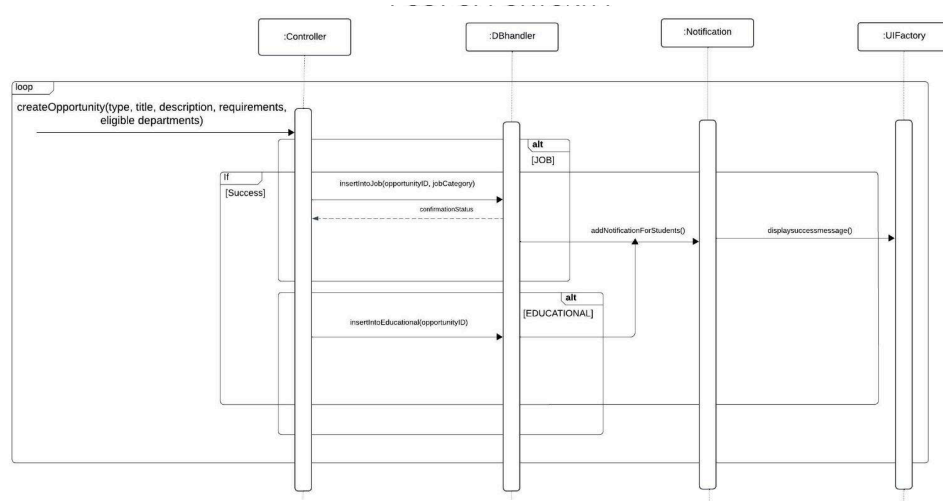
Register Organization and Organization Representative



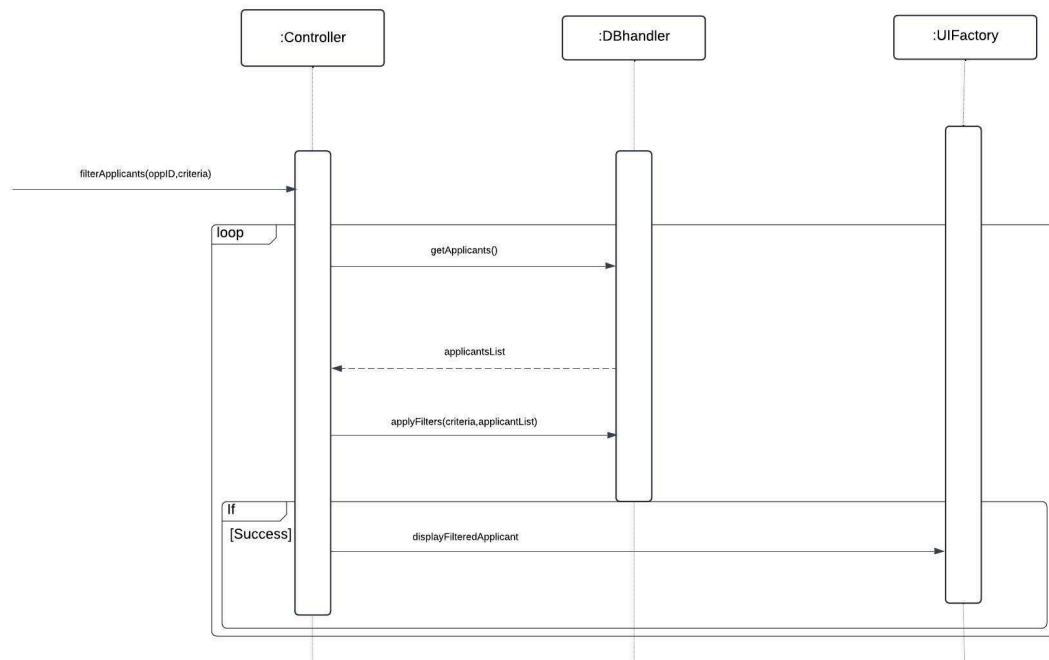
Verify Organization and Organization Representative:



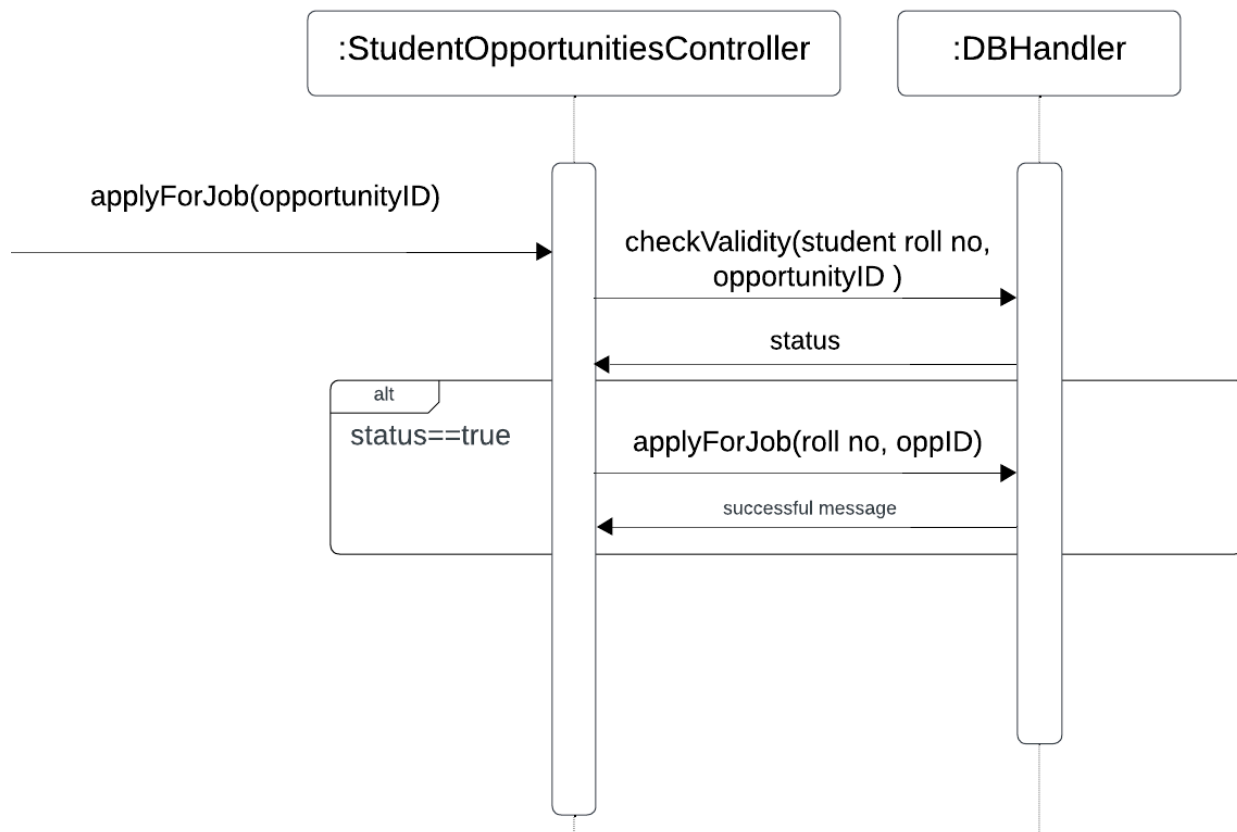
Post Opportunities:



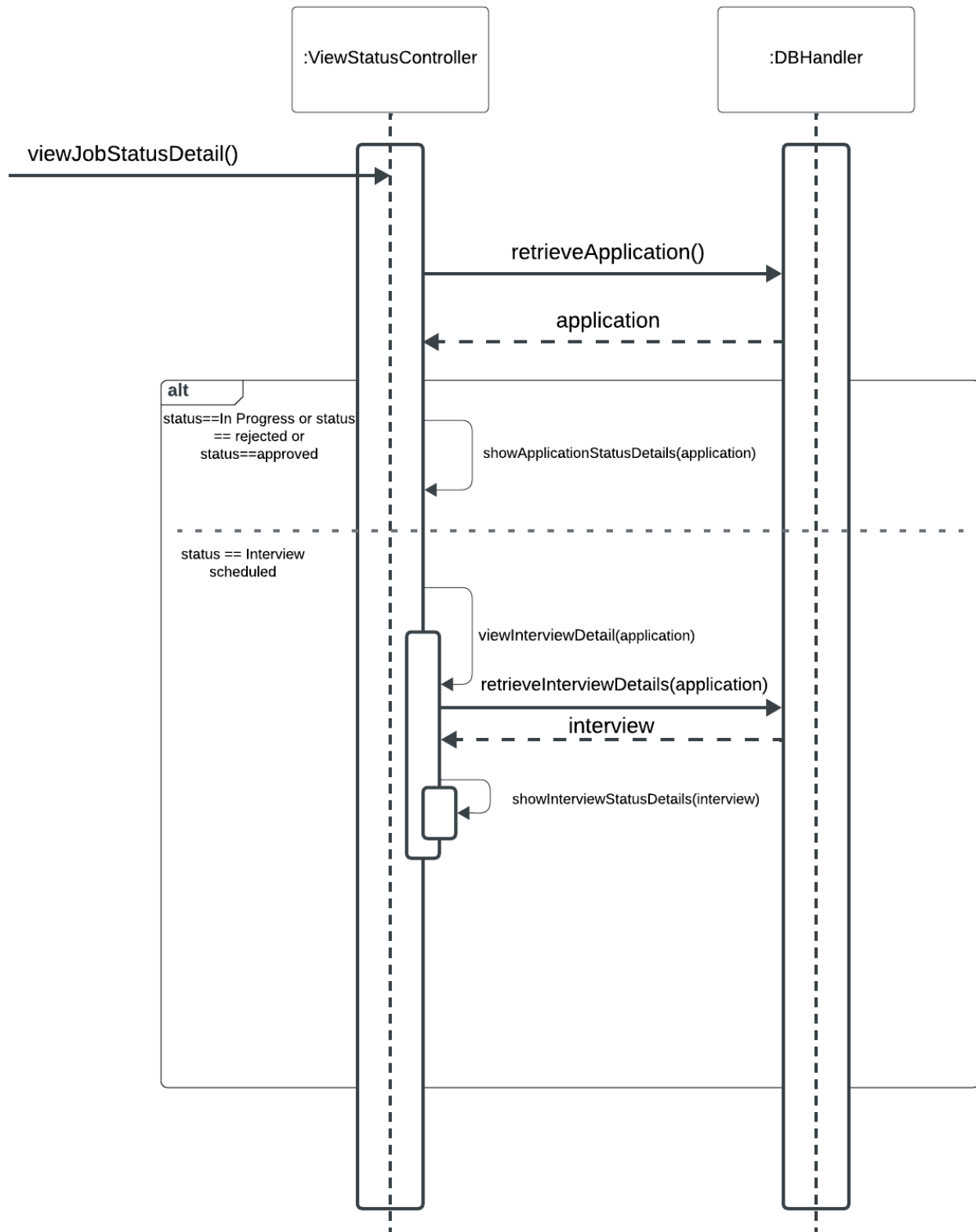
Filter Applicants:



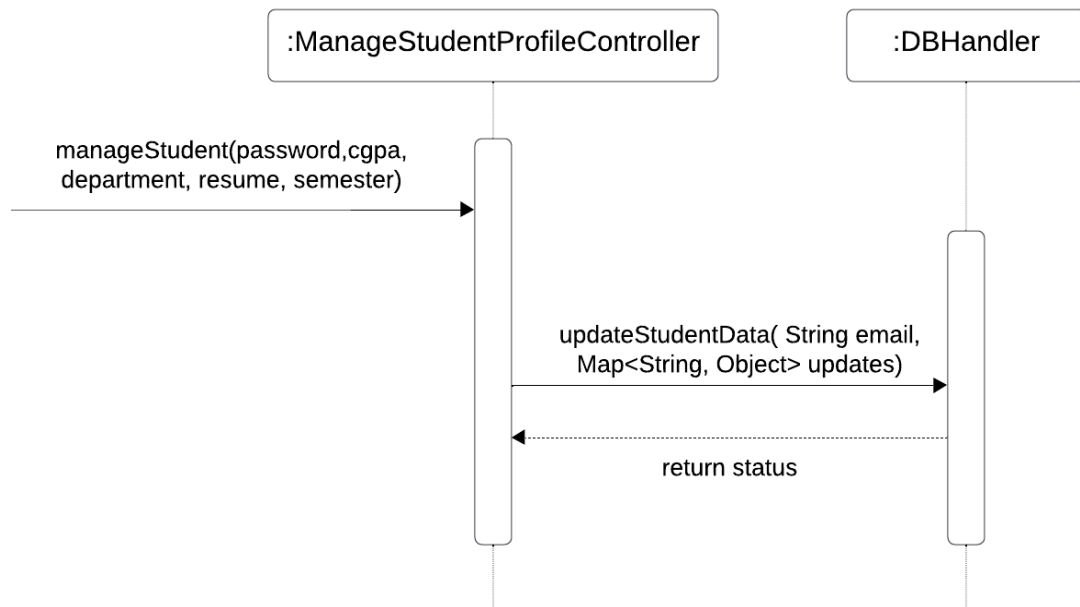
Apply for Job:



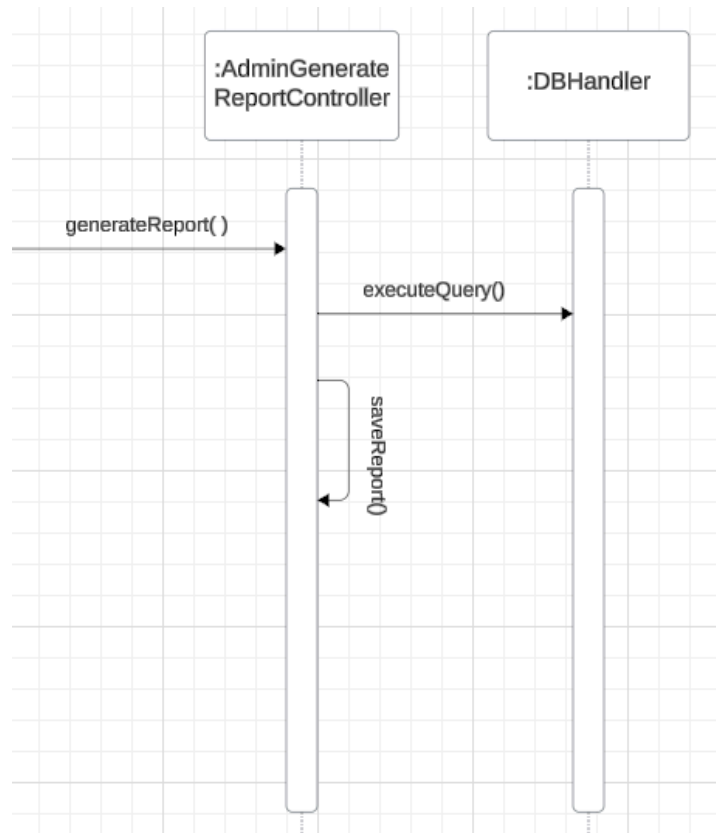
Track Applications:



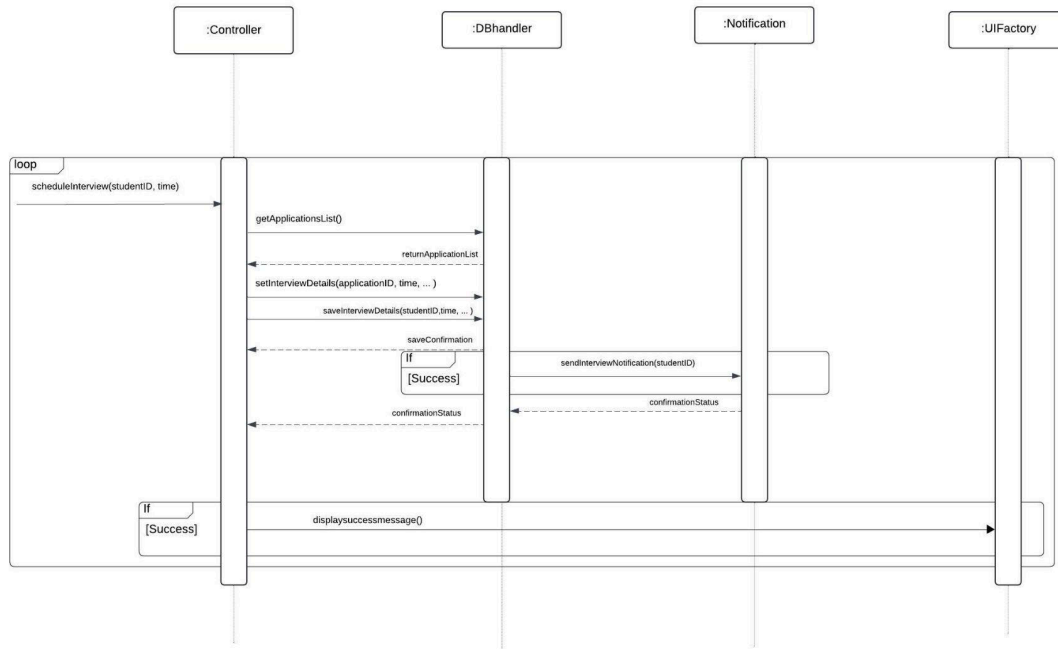
Manage Student Profile:



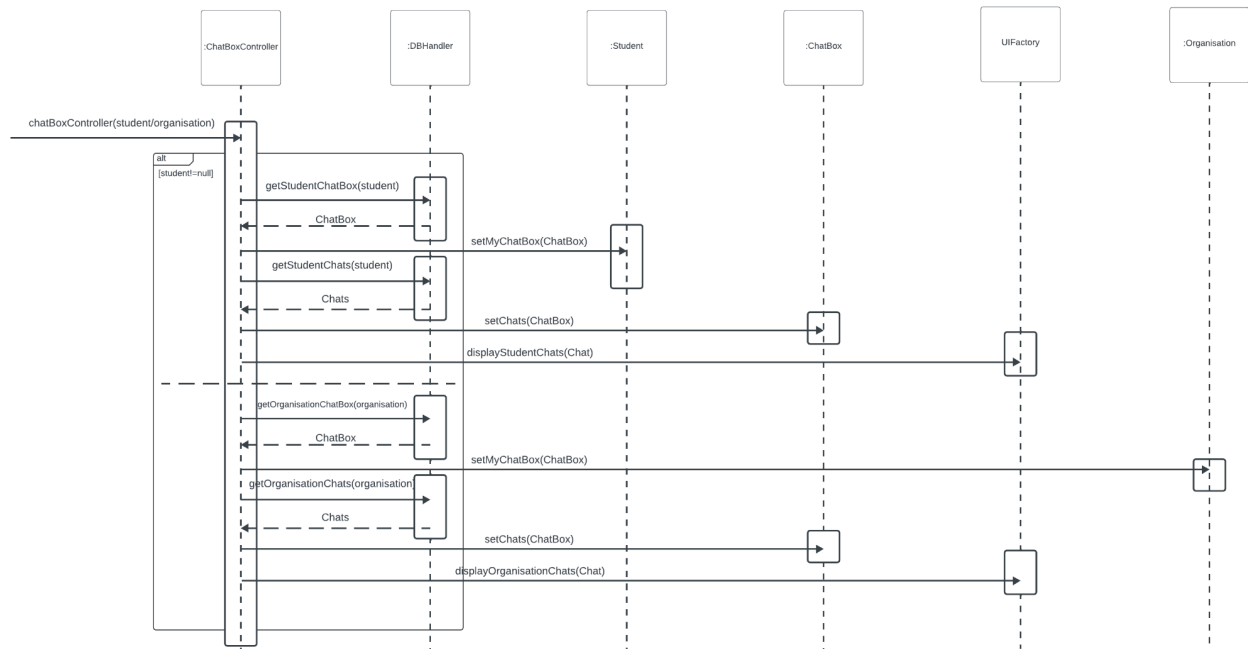
Generate Report:



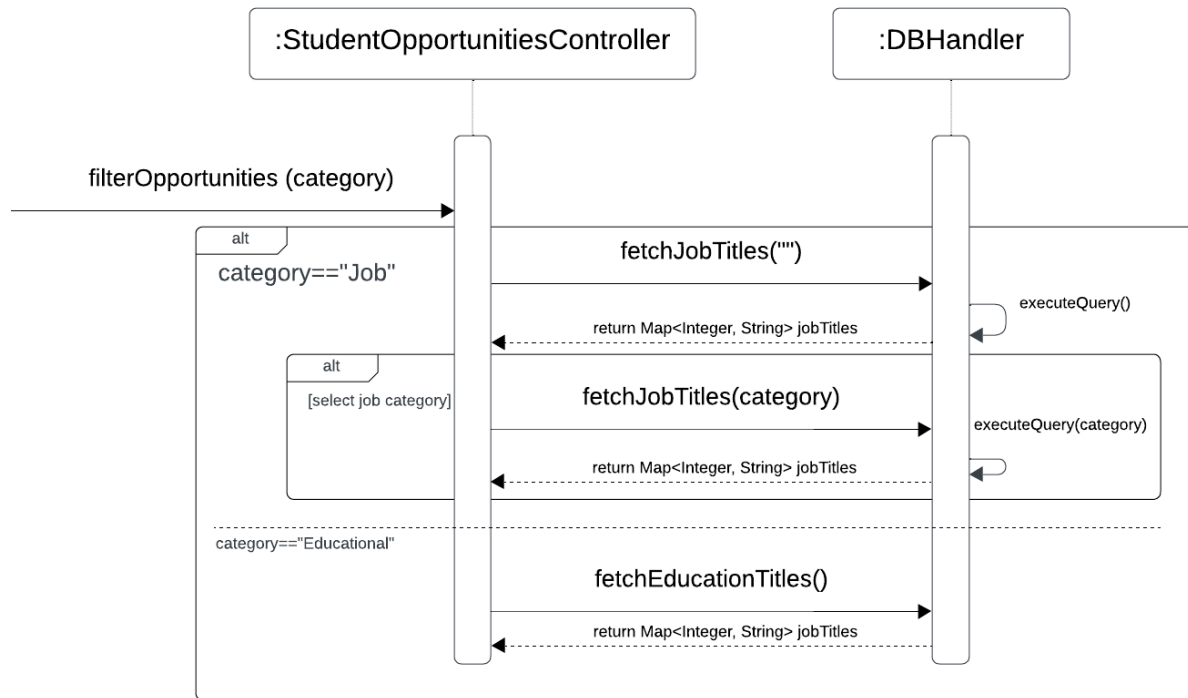
Schedule Interview:



Manage ChatBox:



Filter Opportunities:



3. Other Nonfunctional Requirements

3.1 Performance Requirements

- The system should support at least 500 concurrent users without performance degradation.
- Responses to user actions (e.g., applying for a job) should occur within 2 seconds.

3.2 Safety Requirements

- All user data must be encrypted during storage and transmission to prevent unauthorized access.
- The platform should include measures to prevent data loss in case of a system failure.

3.3 Security Requirements

- User authentication through secure login credentials.

- Role-based access control to ensure only authorized users can access specific features.

3.4 Software Quality Attributes

- **Usability:** The interface should be intuitive and accessible for all users.
- **Maintainability:** The system should allow for seamless updates and bug fixes.
- **Reliability:** The platform should ensure 99.9% uptime.

3.5 Business Rules

- Only verified organizations can post opportunities.
- Students must be enrolled at FAST University to register on the platform.

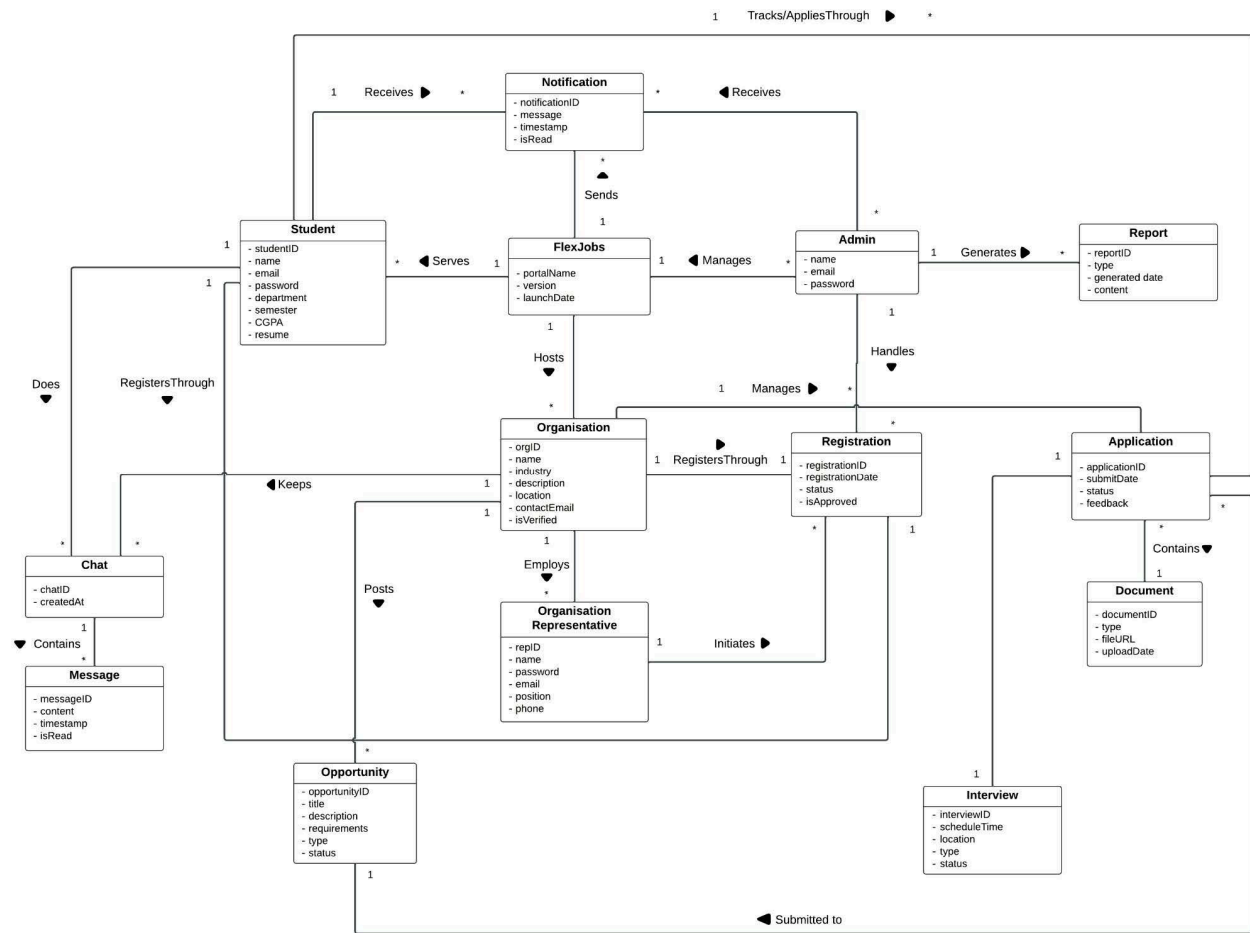
3.6 Operating Environment

- The system is a Java-based application designed to run on personal computers.
- It will operate on the Windows, macOS, and Linux operating systems, ensuring cross-platform compatibility.
- The application requires a Java Runtime Environment (JRE) version 11 or higher.

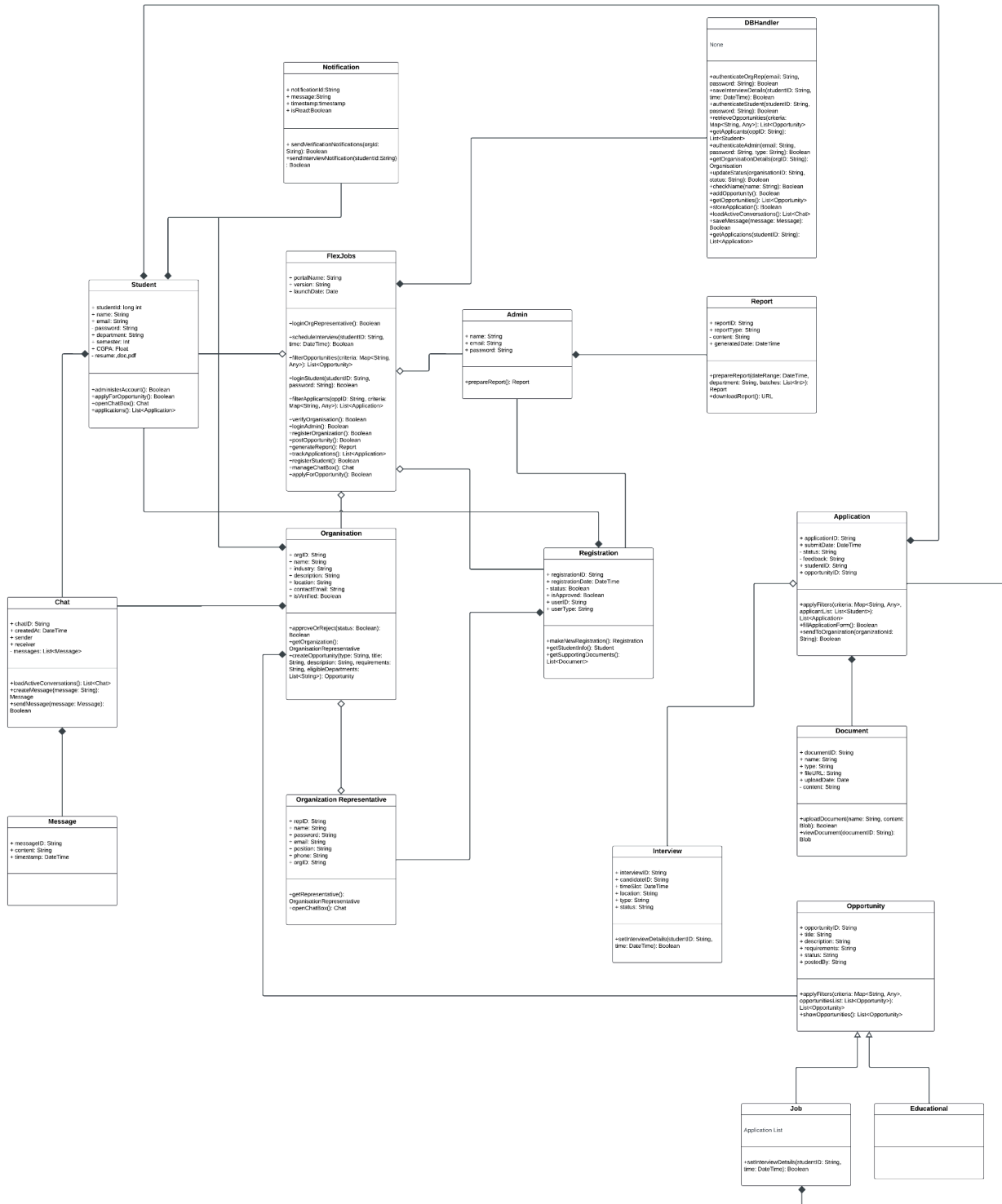
3.7 User Interfaces

- A dashboard for students showcasing opportunities, application statuses, and updates.
- A submission portal for organizations to create and manage postings.
- Admin tools for monitoring system activity and managing user roles.

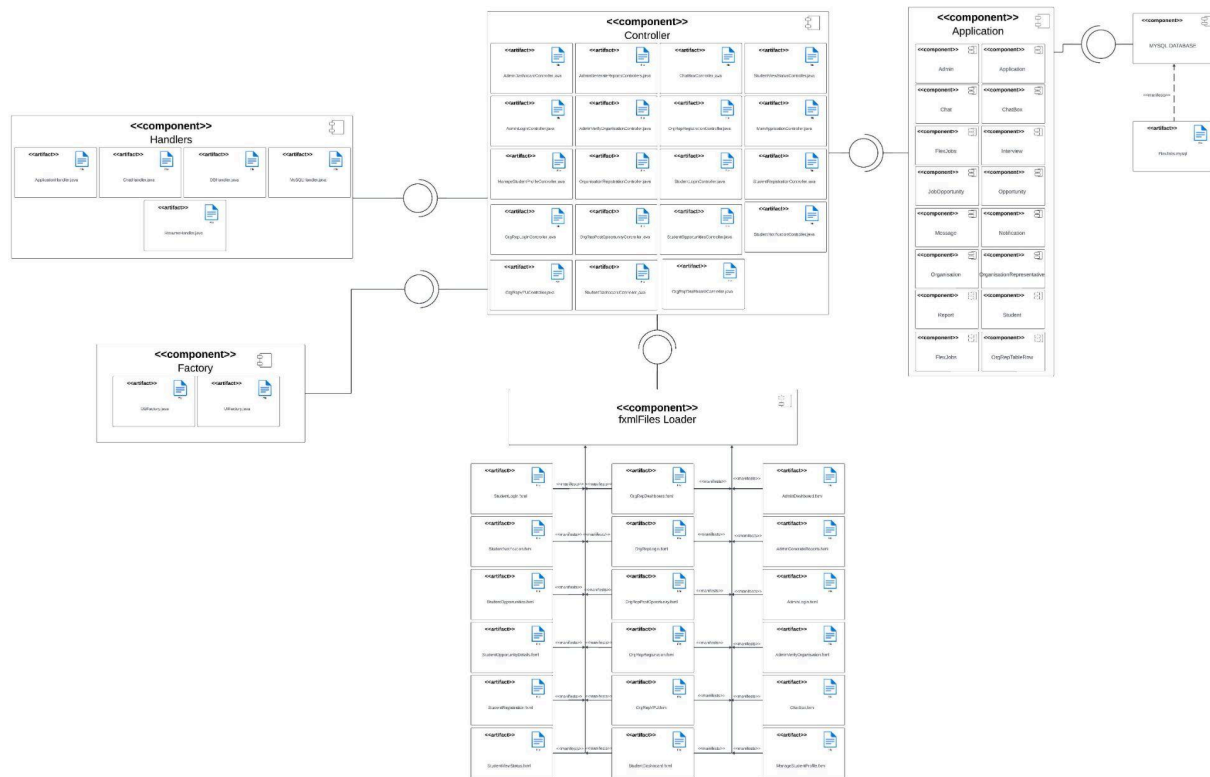
6. Domain Model



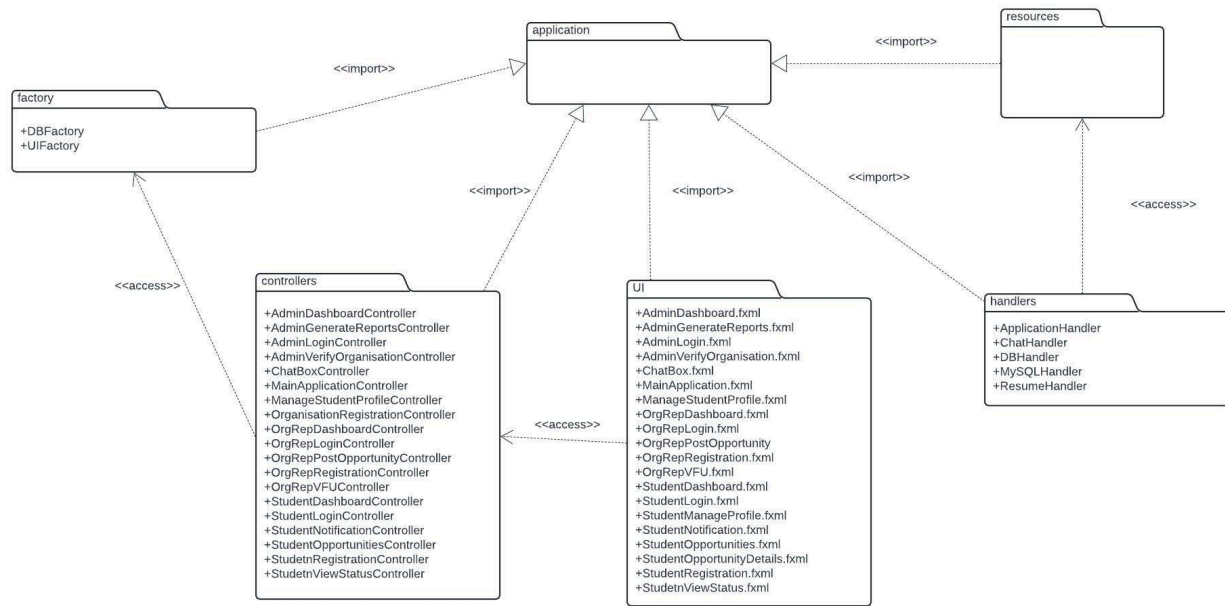
7. Class Diagram



8. Component Diagram



9. Package Diagram



10. Deployment Diagram

