Software Requirements Specification

for

Placement Cell

Version 2.0

Prepared by Group - 22

Dhirubhai Ambani Institute of Information and Communication Technology

23/04/2023

Revision History

Name	Date	Reason For Changes	Version
Group 22	06/04/2023	-	Version 1.0
Group 22	23/04/2023	Because of bugs found in testing.	Version 2.0

Table of Contents

1.   Introduction

1.1   Purpose

This platform of placement website is designed to ease the process of the current placement drive. It includes most of the functionalities that are helpful in easing up the process by the use of the current technology and managing the large database of students and companies. The placement website is a place to simplify the tasks of the placement drive. A lot of data is managed, and its usage, entry, and deletion are made simple by this form. Our stakeholders - students, placement cell members, and companies' requirements are fulfilled by using this platform for their interaction.

1.2   Intended Audience

The intended audience for this srs document is:
- Developers
- Testers
- Users
  - Admin
  - Students
  - Companies
  - Placement Manager

1.3   Project Scope

The software developed as the placement website is useful for the current placement drive as the key benefits of the system include reducing the administrative workload for placement cell members, increasing efficiency and accuracy in managing student and company data, and providing students with an easy-to-use platform to apply for job placements. The objectives of the system are to improve the overall effectiveness of the placement process and increase the number of successful job placements for college students.

2.   Overall Description

2.1   Product Perspective

As the placement drive comes in any college, there are many functions to be performed and many tasks to take place, so if technology can assist in the process, it would prove really helpful. This product is developed to solve the problems of the current placement process. The product is newly developed and helps the placement process.

2.2   Product Features

1. Student information management: A centralised platform for storing and managing student information such as name, CPI, resume and active backlog number.

2. Secure company access: A secure login system for companies to access student information and job postings.

3. Job postings and scheduling: An online platform for companies to post job opportunities and schedule recruiting drives.

4. Online recruitment process: The ability for students to apply for job opportunities and receive updates throughout the recruitment process.

5. Company profile creation: A platform for companies to create company profiles and gather information about the students who have registered for that particular company.

6. User-friendly interface: An intuitive and user-friendly interface for easy navigation and use by all stakeholders.

7. Real-time notifications: A real-time notification system to keep students updated on the latest developments throughout the recruitment process.

8. Application Timing: Students must register or deregister for the company in a limited time specified by the company.

9. Criterion: Companies should be able to add the criteria based on their needs, e.g. cpi should be greater than 7 so the students who come under the company's criteria can easily register for it.

10. Company Information: Students will be able to see all companies details that are going to visit the campus for recruitment.

11. Company filters: Students should be able to filter the companies based on their interest like roles in the company, location of the job etc.

2.3   User Classes and Characteristics

We have 5 user classes:

1. *Student*
      Users who are seeking job opportunities through the placement cell and can view job postings and apply for positions.

2. *Admin (Placement Committee Members)*
   Users who are responsible for coordinating the placement process and communicating with companies and students.

3. *SuperAdmin (Placement Manager)*
   Users who have the ability to manage and configure the website, such as adding or removing companies and job postings.

4. *Company*
   Users who represent companies that are interested in recruiting students through the placement cell and can post job opportunities.

5. *General/Viewer*
   One who is new to the website and visits the website just for viewing it and getting information about placements at DAIICT.

Characteristics for each user class:

1. Student
   ● Level of access: Students would have access to view job postings and apply for positions through the placement cell website.
   ● Technical expertise: Students may have varying levels of technical expertise and familiarity with the website, so it should be designed to be user-friendly and easy to navigate.
   ● Role within the organization: Students are the primary users of the placement cell website, and its main purpose is to facilitate their job search process.
   ● Frequency of use: Students may use the placement cell website frequently during their job search process to view new job postings and track their applications.

2. Admin(Placement Committee members)
   ● Level of access: Admins would have access to manage job postings and communicate with companies and students.
   ● Technical expertise: Admins would be expected to have a good understanding of the website and its features.
   ● Role within the organization: Admins would be responsible for coordinating the placement process and facilitating communication between companies and students.
   ● Frequency of use: Admins may use the website frequently to manage job postings and track student applications.

3. SuperAdmin (Placement Manager)
   ● Level of access: Super Admin would have the highest level of access to the website, including the ability to manage and configure its settings.

- Technical expertise: Super Admin would be expected to have a high level of technical expertise and familiarity with the website and its features.
- Role within the organization: Super Admin would be responsible for managing and maintaining the placement cell website.
- Frequency of use: Super Admin may use the website frequently to monitor its performance and make any necessary updates or changes.

4. Company
   - Level of access: Companies would have access to post job opportunities and view student applications through the placement cell website.
   - Technical expertise: Companies may have varying levels of technical expertise and familiarity with the website, so it should be designed to be user-friendly and easy to navigate.
   - Role within the organization: Companies are external users who interact with the placement cell website to recruit students for job opportunities.
   - Frequency of use: Companies may use the placement cell website frequently during their recruitment process to post new job opportunities and review student applications.

5. General User / Viewer
   - Level of access: Users won't have access to website functionality except registration.
   - Technical expertise: The technical expertise of the user is assumed to be zero as the user is completely new to the website and has visited the site to know more about it.
   - Role within the organization: They are the external users who would like to know about the placements at DAIICT.
   - Frequency of use: They may use the website once or twice as it is used just to know more about the functioning of placements at DAIICT.

2.4   Operating Environment

1. The website will be designed to run on a Node server engine that supports the required programming languages and frameworks.

2. The website will require a NoSQL database management system such as MongoDB to store user data and job postings.

3. The website will need to integrate with external services, such as email providers to enable admins to perform necessary actions.

4. The website will need to be secured using appropriate security mechanisms, such as encryption, regular security audits like password encryption of users, etc.

5. The website's development and maintenance will require a development environment, including a code editor, a version control system, and a testing environment. The deployment process will require a continuous integration/continuous deployment (CI/CD) pipeline to ensure smooth and error-free releases.

2.5   Design and Implementation Constraints

Database: We have used basic level of MongoDB database which gives us 500MB storage for the web application. This restricted us in collecting every user's resume file. Thus we ask user to input the link of the resume. For more data, we have to pay a certain amount to buy the storage of database.

Nodemailer: We have used nodemailer to send user real-time notifications when a job is posted. This slows down the process of the email notification and we can only email certain number of users.

Host Portal: We have hosted our website on free onrender.com platform. This can lead to limited users on the platform.

2.6   Assumptions and Dependencies

Assumptions:
1. The website will be used to connect job seekers with potential employers for full-time, part-time, and internship positions.
2. Users will be able to create accounts, upload resumes, and apply for jobs through the website.
3. Employers will be able to create accounts, post job openings, and review resumes and applications through the website.
4. The website will need to be compatible with major web browsers.
5. The website will need to comply with relevant laws and regulations, such as those governing user data privacy and job advertising.

Dependencies:
1. The website will require a database to store user information, job postings, applications, student registration, and company registrations.
2. The website will need to integrate with third-party services, such as email providers.
3. The website will need to be hosted on a reliable and scalable web hosting platform to ensure high availability and performance.
4. The website will need to be developed using appropriate web development tools and technologies, such as HTML, CSS, JavaScript, and a web framework like Node.js and Express.js.

3. Functional and Non-Functional Requirements

3.1 Functional Requirements

General:
- The system should be able to manage and store all the information.
- The system should display past drive placement statistics.
- The system should have a contact us section.
- The system should have an about us page to know more about the placement cell.
- The system should display a basic introduction about the college and its placements on its home page.

For Students:
- Students should be able to register, log in, and log out of the system.
- Students should be able to see their profile and change their password.
- Students should be able to edit their profile only before getting verified.
- Students should be able to upload the latest version of their resume whenever needed.
- Students should be able to register and deregister for various Jobs after their verification.
- Students should be able to filter the jobs according to their eligibility and interest.
- Students who are not eligible for the job should not be allowed to register for the same.
- Students(Not placed and registered) will be notified through mail when a job posted by the company is verified by the placement manager.

For Companies:
- Companies should be able to register, log in, and log out of the system.
- Companies should be able to see their profile and change their password.
- Companies should be able to edit their profile only before getting verified.
- Companies should be able to hire students and these students gets marked as placed.
- Companies should be able to post the jobs after the verification is done by the placement manager.
- A company should be able to post multiple jobs.
- Companies should be able to edit/delete the job posting until it is not verified.
- Companies should be able to view the registered student's details for their postings.

For Admin/Placement cell members:
- Admin should be able to log in and log out of the system.
- Admin should be able to change their password.
- Admin should be able to verify the students registered to the system.
- Admin should be able to view the full list of registered students and their details.
- Admin should be able to view the full list of registered companies.
- Admin should be able to verify the jobs posted by the companies.
- Admin should be able to view the full list of jobs posted by the companies.

- Admin should be able to schedule the registration period for each job.
- Admin should be able to view the current drive statistics.
- Admin should be able to download the Excel sheet of students' details who are not placed yet.

For Super Admin/Placement Manager:
- The super admin will have access to all the admin functionalities.
- The super admin should be able to verify the companies registered to the system.

3.2 Non-Functional Requirements

1. **Authentication** - While login, the system should check to see if a user's credentials match the credentials in a database.
2. **Scalability** - The system will be used by a huge number of students as well as companies, so it must adapt to handle such a high number of users.
3. **Reliability**
   - The database must be kept up to date so it does not display companies not currently recruiting.
   - This will ensure that the students can only apply for the companies that are currently recruiting and those for which the students are eligible.
4. **Maintainability** - The web application should have the capability of modifications and updates so that the technology used does not get outdated too quickly. It can be easily updated even if the website developer team changes after the final product's deployment.
5. **Compatibility** - The website should be able to run on the latest versions of browsers and some older ones.
6. **Usability** - The UI/UX of the website should be simple yet elegant so that students can filter companies according to their choice and perform other actions easily without any special training. User comfort and easy interaction with the website should be ensured.
7. **Accuracy** - The information that is recorded about the companies and the students that are registered for particular companies must be accurate, reliable, and consistent.
8. **Security** - User credentials should not leak in a data breach, as it is a privacy concern.

4. Use Cases

4.1 General

- Know about placements at DAIICT

4.2 Students

- Manage Profile

- Update Password
- Update Resume
- View jobs
- Register/Deregister for jobs
- Filter jobs

4.3   Companies

- Post and manage Job
- View posted jobs
- View registered students' details
- Hire Students

4.4   Admin/Super Admin

- Verify student
- View student list
- Verify jobs
- View job list
- View company list
- Verify company(*Super Admin Exclusive*)
- View current placement drive statistics
- Edit job schedule
- Download unplaced verified student's Datasheet

5.   Use Cases Descriptions

5.1   Know about placements at DAIICT

- ➢ Actors: Any user who visits the website
- ➢ Goal: Gain the placement related information for DAIICT
- ➢ Pre-conditions:
  - ○ The user is coming for the first time and is unknown to the website and placement drive at DAIICT.
- ➢ Description:
  1. Open the website.
  2. The system will display the homepage of the website.
  3. Analyze the placement Statistics.
  4. Read the Director's Message and the Student Achievement Section.
  5. Scroll down and see the list of past recruiters.
  6. Contact details are displayed at the bottom of the page.
  7. Go to the "About Us" page from the nav bar.

8. Read the placement cell details and download the brochure.

➢ Related Functional Requirements:
   ○ The system should display past drive placement statistics.
   ○ The system should have a contact us section.
   ○ The system should have an about us page to know more about the placement cell.
   ○ The system should display a basic introduction about the college and its placements on its home page.
➢ Related Non-Functional Requirements: Usability, Reliability, Compatibility

5.2 Manage Profile

➢ Actors: Student, Company
➢ Goal: To enable the student and company to edit their profile.
➢ Pre-conditions:
   ○ The student and company have registered to the system.
   ○ The student and company are not verified by the admin/super-admin.
   ○ The student or company has logged into the system using their login id.
➢ Description:
   1. Go to the "Edit Profile" section from the sidebar.
   2. The system presents the edit form where the existing details are prefilled.
   3. Edit the desired details and select the "Save Changes" option.
   4. The system saves the changes and displays them in the "Profile" section.
➢ Exception:
   4.a If the student/company leaves any required fields blank, the system prompts them to fill in the missing information before submitting the changes.
➢ Post-Conditions:
   ○ The system should display the changes.
➢ Related Functional Requirements:
   ○ Students should be able to edit their profile only before getting verified.
   ○ Companies should be able to edit their profile only before getting verified.
➢ Related Non-Functional Requirements: Usability, Accuracy

5.3 Update Password

➢ Actors: Student, Company, Admin, Super Admin
➢ Goal: To enable the actors to change their profile password.
➢ Pre-conditions:
   ○ The actor has registered to the system.
   ○ The actor logs into the system using their login id.
➢ Description:
   1. Go to the "Update Password" section from the sidebar.

2. The system displays the update password screen.
3. Enter the old password.
4. Enter the new password and retype it in the confirm password.
5. Click on the "Save Password" button.
➢ Exception:
   ○ 3.a If the old password is incorrect, the system will throw an error.
   ○ 4.a If the new password and confirm password does not match, the password will not change.
   ○ 4.b if the new password is not of valid pattern specified, the password will not change.
➢ Post-conditions:
   ○ The password will be updated in the database.
➢ Related Functional Requirements:
   ○ Companies should be able to see their profile and change their password.
   ○ Students should be able to see their profile and change their password.
   ○ Admin/Super-admin should be able to change their password.
➢ Related Non-Functional Requirements: Security, Authentication

5.4   Update Resume

➢ Actors: Student
➢ Goal: To enable the students to update their resume.
➢ Pre-conditions:
   ○ The student is logged in the system.
➢ Description:
   1. Go to the "Update Resume" section from the sidebar.
   2. The system displays the update resume screen.
   3. The student enters the updated resume's URL.
   4. Click on the "Save Resume" button.
➢ Exception:
   ○ 3.a The link provided is not a valid URL.
➢ Post-conditions:
   ○ The resume link will be updated in the database.
➢ Related Functional Requirements:
   ○ Students should be able to upload the latest version of their resume whenever needed.
➢ Related Non-Functional Requirements: Accuracy, Reliability

5.5   View Jobs

➢ Actors: Student
➢ Goal: To enable the students to view verified and applicable jobs.
➢ Pre-conditions:

- ○ The student has registered to the system.
- ○ The student is verified.
- ○ The student logs into the system using their login Id.
➢ Description:
    1. Go to the "Companies" section from the sidebar.
    2. The system displays all the verified and available jobs.
➢ Exception:
    2.a If no jobs are available, the system will display no information.
➢ Post-Conditions:
    - ○ The student should be able to apply for the companies in which (s)he is interested and meets the criteria.
➢ Related Functional Requirements:
    - ○ The system should be able to manage and store all the information.
➢ Related Non-Functional Requirements: Reliability, Usability

5.6   Register/Deregister for jobs

➢ Actors: Student
➢ Goal: To enable the student to register/Deregister for the jobs.
➢ Pre-conditions:
    - ○ The student has logged into the system.
    - ○ The student is verified.
    - ○ The posted job is verified.
    - ○ The job has been scheduled.
    - ○ The student meets the criteria of that particular job.
➢ Description:
    1. Go to the "Companies" section from the sidebar.
    2. The system displays all the verified and available jobs.
    3. The student can register for the jobs of their interest.
    4. The student can deregister for the jobs they have registered.
➢ Exception:
    3.a The rejected and placed students will not be able to register for any jobs.
    3.a Students who are not eligible for the jobs will not be allowed to register for those jobs.
    4.a If the registration deadline has passed, the student will not be able to register/deregister.
➢ Post-Conditions:
    - ○ The student will be registered for that particular job.
    - ○ The details of the student will be displayed to the company for that job.
    - ○ If (s)he deregisters for a job, the system will update that, and they will be able to register again.
➢ Related Functional Requirements:

- ○ Students should be able to register and deregister for various Jobs after their verification.
- ○ Students who are not eligible for the job should not be allowed to register for the same.
➢ Related Non-Functional Requirements: Reliability, Usability, Scalability

5.7   Filter jobs

➢ Actors: Student
➢ Goal: To enable the student to filter the jobs according to their priority and eligibility.
➢ Pre-conditions:
- ○ Student is logged into the system.
➢ Description:
1. Go to the 'Companies' section from the sidebar.
2. The System will display all the jobs' information.
3. Choose the filters you want to apply from CPI, Location, Branch, CTC, and Job Title.
4. Apply the filter.
➢ Exception:

  4.a If "any" is selected, then the filter will not be applied to that section.
➢ Post-Conditions:
- ○ Filtered list of companies according to the priorities applied in the filter will be displayed.
➢ Related Functional Requirements:
- ○ Students should be able to filter the jobs according to their eligibility and interest.
➢ Related Non-Functional Requirements: Reliability, Usability, Scalability

5.8   Post and manage Job

➢ Actors: Company
➢ Goal: To enable the companies to post and edit jobs.
➢ Pre-conditions:
- ○ The company is registered to the system and is verified.
- ○ The company has logged into the system.
➢ Description:
1. Go to the "Post Job" section from the sidebar.
2. The system displays a form for the company to enter details about job openings.
3. The company fills in the required fields and submits the job posting to the system.
4. The system saves the job posting.
5. The company can edit the job from the "Edit Jobs" section.
6. The company can delete the job from the "Delete jobs" section.
➢ Exceptions:

  3.a If the company leaves any required fields blank, the system prompts the company to fill in the missing information before submitting the job posting.

5.a The company can edit the job until the job is not verified.
➢ Post-conditions:
  ○ The company should see its job in its posted job list after posting it.
  ○ The system displays the job in the "Job list" and "Verify jobs" pages of the admin section.
➢ Related Functional Requirements:
  ○ Companies should be able to post the jobs after the verification is done by the placement manager.
  ○ A company should be able to post multiple jobs.
  ○ Companies should be able to edit/delete the job posting until it is not verified.
➢ Related Non-Functional Requirements: Reliability, Usability, Accuracy

5.9 View posted jobs

➢ Actors: Company
➢ Goal: To enable the company to view their posted jobs.
➢ Pre-conditions:
  ○ The company has logged into the system.
  ○ The company is verified.
➢ Description:
  1. Go to the "Posted Jobs" section from the Profile page.
  2. The system will display all the jobs posted by the company.
➢ Exception:
    2.a If the company hasn't posted any job, it will display no information in that section.
➢ Post-Conditions:
  ○ All the jobs posted by that company will be displayed; the company can select any job for details.
➢ Related Functional Requirements:
  ○ The system should be able to manage and store all the information.
➢ Related Non-Functional Requirements: Reliability, Usability

5.10 View registered students' details

➢ Actors: Company
➢ Goal: To enable the company to view students' details who have registered for that particular company's posted job.
➢ Pre-conditions:
  ○ The company logged into the system.
  ○ The company has posted a job.
  ○ The posted job is verified.
➢ Description:
  1. The company selects the particular job in the "Posted Jobs" section on the Profile page.

2. The company selects the 'Student List' option.
➢ Exception:
2.a No details will be displayed if no student has registered for that job.
➢ Post-Conditions:
○ The system will display the registered students' details for that particular job.
➢ Related Functional Requirements:
○ Companies should be able to view the registered student's details for their postings.
➢ Related Non-Functional Requirements: Reliability, Usability

5.11 Hire Students

➢ Actor: Company
➢ Goal: To enable the company to hire students based on their choice.
➢ Pre Condition:
○ The company has logged into the system
○ The company has posted a job and is verified.
○ The job registration is scheduled.
➢ Description:
1. Go to the "Posted Jobs" section from the Profile page.
2. The system displays all the jobs posted by that company.
3. The company selects a job.
4. The system displays the job's details.
5. Selects the "Students List" option.
6. The system displays the list of students who have registered for that job.
7. The company can select any student and can hire them by selecting the button "Hire".
➢ Exception:
○ 6.a No information will be displayed if no student has registered for that job.
➢ Post Condition:
○ The student will get placed in that company for the selected job.
○ The job's details will be displayed in the student's profile section.
○ The company will be able to see the hired student from the student list.
○ The admin will be able to see the hired student from the student list.
➢ Related Functional Requirements:
○ Companies should be able to hire students and these students gets marked as placed.
➢ Related Non-Functional Requirements: Reliability, Usability, Accuracy

5.12 Verify student

➢ Actors: Admin, Super Admin
➢ Goal: To enable admin/super admin to verify students based on the details provided by them.
➢ Pre-conditions:

- ○ A new student has registered to the system.
- ○ Admin/Super Admin must be logged in to the system.
- ➢ Description:
  1. Go to the "Verify Students" section from the sidebar.
  2. The details of a newly registered student are displayed with accept and reject buttons.
  3. Admin/Super Admin can choose to accept or reject the student based on the details provided.
- ➢ Post-Conditions:
  - ○ The new student's profile is verified/rejected.
  - ○ The system will display the result in the Student's profile page.
- ➢ Related Functional Requirements:
  - ○ Admin/Super Admin should be able to verify the students registered to the system.
- ➢ Related Non-Functional Requirements: Reliability, Usability, Accuracy

5.13 View student list

- ➢ Actors: Admin, Super Admin
- ➢ Goal: To enable admin/super admin to see the list of the students whose profile is verified.
- ➢ Pre-conditions:
  - ○ Admin/Super Admin must be logged in to the system.
- ➢ Description:
  1. Admin/Super Admin navigates to the Student List page on the placement cell website.
  2. The system displays a list of students.
  3. Admin/Super Admin can view the details of each student on the list, including their name, cpi, and resume link.
- ➢ Exception:
  2.a System will display no information if no students have registered yet.
- ➢ Post-Conditions:
  - ○ Admin/Super Admin is shown the list of students.
- ➢ Related Functional Requirements:
  - ○ Admin/Super Admin should be able to view the full list of registered students and their details.
- ➢ Related Non-Functional Requirements: Reliability, Usability

5.14 Verify jobs

- ➢ Actors: Admin, Super-admin
- ➢ Goal: To enable the admin/super admin to verify the posted jobs by the companies if everything is correct.
- ➢ Pre-conditions:
  - ○ Admin/super-admin should be logged into the system.
  - ○ A new job is posted by a company.

➢ Description:
  1. Go to the "Verify Jobs" section from the sidebar.
  2. Accept or reject the job according to the description.
➢ Post-Conditions:
  ○ The job is verified or rejected based on the choice made by Admin/Super Admin.
  ○ The job should not be displayed on the screen once accepted or rejected.
  ○ The verified job will be displayed in the scheduled job section.
➢ Related Functional Requirements:
  ○ Admin/super-admin should be able to verify the jobs posted by the companies.
➢ Related Non-Functional Requirements: Reliability, Usability, Accuracy

5.15 View job list

➢ Actors: Admin, Super-admin
➢ Goal: To enable the admin/super admin to view the list of posted jobs.
➢ Pre-conditions:
  ○ Admin / super-admin should be logged in to the system.
  ○ A new job should be posted.
➢ Description:
  1. Go to the "Job List" section from the sidebar.
  2. The system will display the Job list.
  3. By clicking on the job, a dropdown of the information about the job will be displayed.
➢ Post-Conditions:
  ○ List of the jobs posted by the companies will be displayed.
➢ Related Functional Requirements:
  ○ Admin should be able to view the full list of jobs posted by the companies.
➢ Related Non-Functional Requirements: Reliability, Usability

5.16 Verify company

➢ Actors: Super Admin
➢ Goal: To enable the super admin to verify/reject the registered company.
➢ Pre-conditions:
  ○ The company should have registered on the placement website.
  ○ Super-admin should have logged into the system.
➢ Description:
  1. Go to the "Verify Company" section from the sidebar.
  2. The system will display the list of registered companies.
  3. Accept or reject the company as per the details shown by the company.
➢ Post-Conditions:
  ○ Companies are not displayed in the list once accepted or rejected.
  ○ The companies should be able to post jobs.
➢ Related Functional Requirements:
  ○ The super admin should be able to verify the companies registered to the system.

➢ Related Non-Functional Requirements: Reliability, Usability, Accuracy

5.17   View company list

➢ Actors: Admin, Super-admin
➢ Goal: To enable admin/super admin to view the full list of registered companies.
➢ Pre-conditions:
    ○ The admin/super admin has logged into the system.
➢ Description:
    1. Go to the "Company List" option from the sidebar.
    2. The system will display all the companies that have registered in the system.
➢ Exception:
    2.a If no company has registered for the system, the system will display no information.
➢ Post-Conditions:
    ○ List of companies will be displayed, and you can select any company for more details.
➢ Related Functional Requirements:
    ○ Admin/super-admin should be able to view the full list of registered companies.
➢ Related Non-Functional Requirements:Reliability, Usability

5.18   View current placement drive statistics

➢ Actors: Admin, Super-admin
➢ Goal: to enable the admin/super admin to view the current placement drive statistics.
➢ Pre-conditions:
    ○ Admin/super-admin should be logged in to the system.
➢ Description:
    1. Go to the "Admin Statistics" section from the sidebar.
    2. System displays the analysis of the current drive in the form of charts.
    3. By hovering over the charts, you can see information about that part of the chart.
    4. Analyze the chart and current placement drive state.
➢ Post-Conditions:
    ○ Statistical charts will be visible.
➢ Related Functional Requirements:
    ○ Admin/super-admin should be able to view the current drive statistics.
➢ Related Non-Functional Requirements: Reliability, Usability

5.19   Edit job schedule

➢ Actors: Admin, Super-admin
➢ Goal: To enable the admin/super admin to edit the job schedule.

➢ Pre-conditions:
  ○ Admin/ super-admin should be logged in to the system.
  ○ A job must be posted by the company and is verified.
➢ Description:
  1. Go to the "Schedule Jobs" section from the sidebar.
  2. The system will display the verified jobs.
  3. Select the start and the end date of the job that you want to schedule.
  4. Click on the submit button, and all the students will be mailed about that job posting.
➢ Post-Conditions:
  ○ All the students are mailed or notified once the job is scheduled.
➢ Related Functional Requirements:
  ○ Admin should be able to schedule the registration period for each job.
➢ Related Non-Functional Requirements: Reliability, Usability, Accuracy

5.20 Download Unplaced Verified Students' Datasheet

➢ Actors: Admin, Super-admin
➢ Goal: To enable the admin/super admin to download the datasheet.
➢ Pre-conditions:
  ○ Admin/ super-admin should be logged in to the system.
➢ Description:
  1. Click on the "Download Datasheet" option from the sidebar.
➢ Post-Conditions:
  ○ The datasheet is downloaded.
➢ Related Functional Requirements:
  ○ Admin should be able to download the Excel sheet of students' details who are not placed yet.
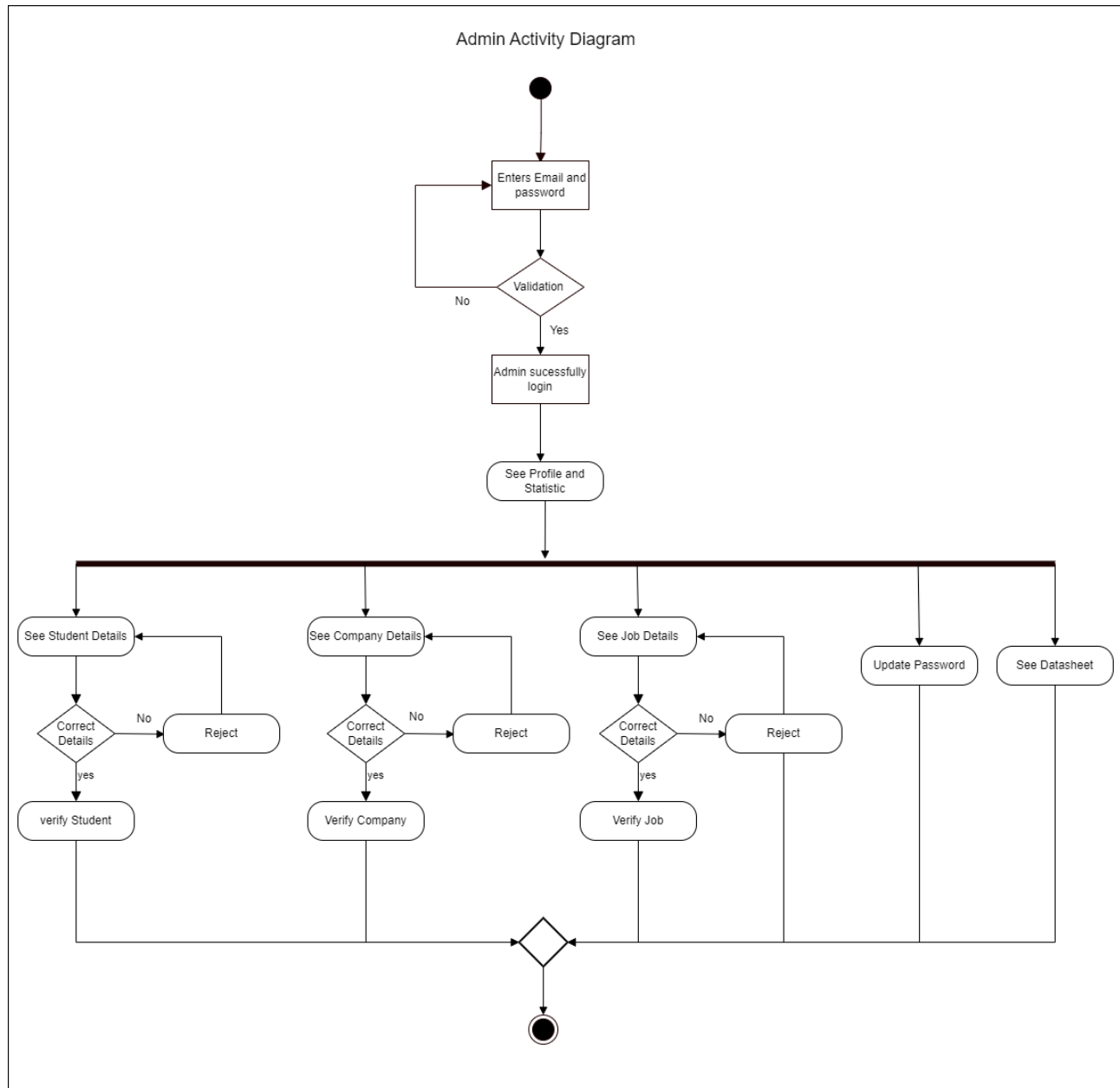➢ Related Non-Functional Requirements: Reliability
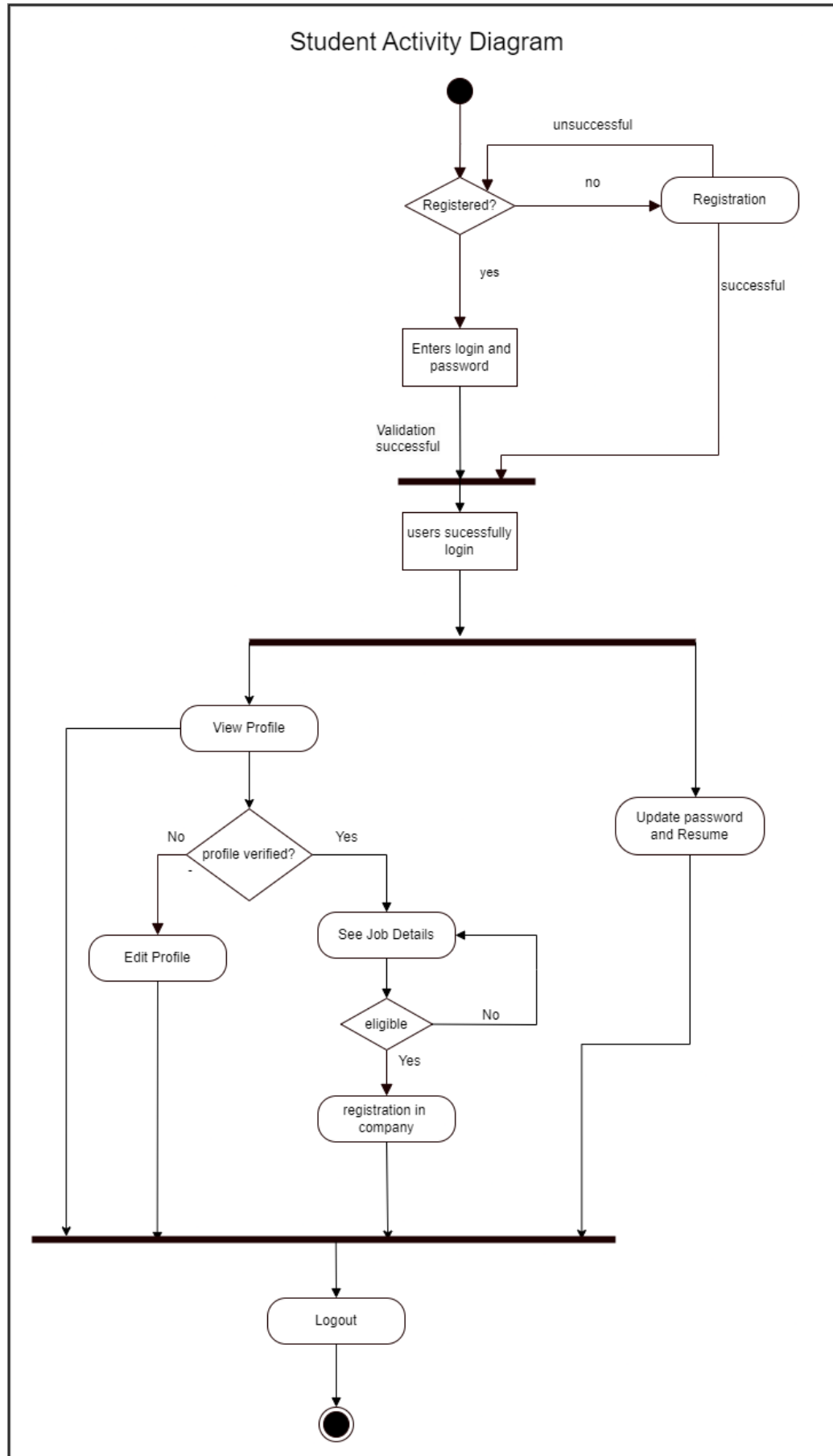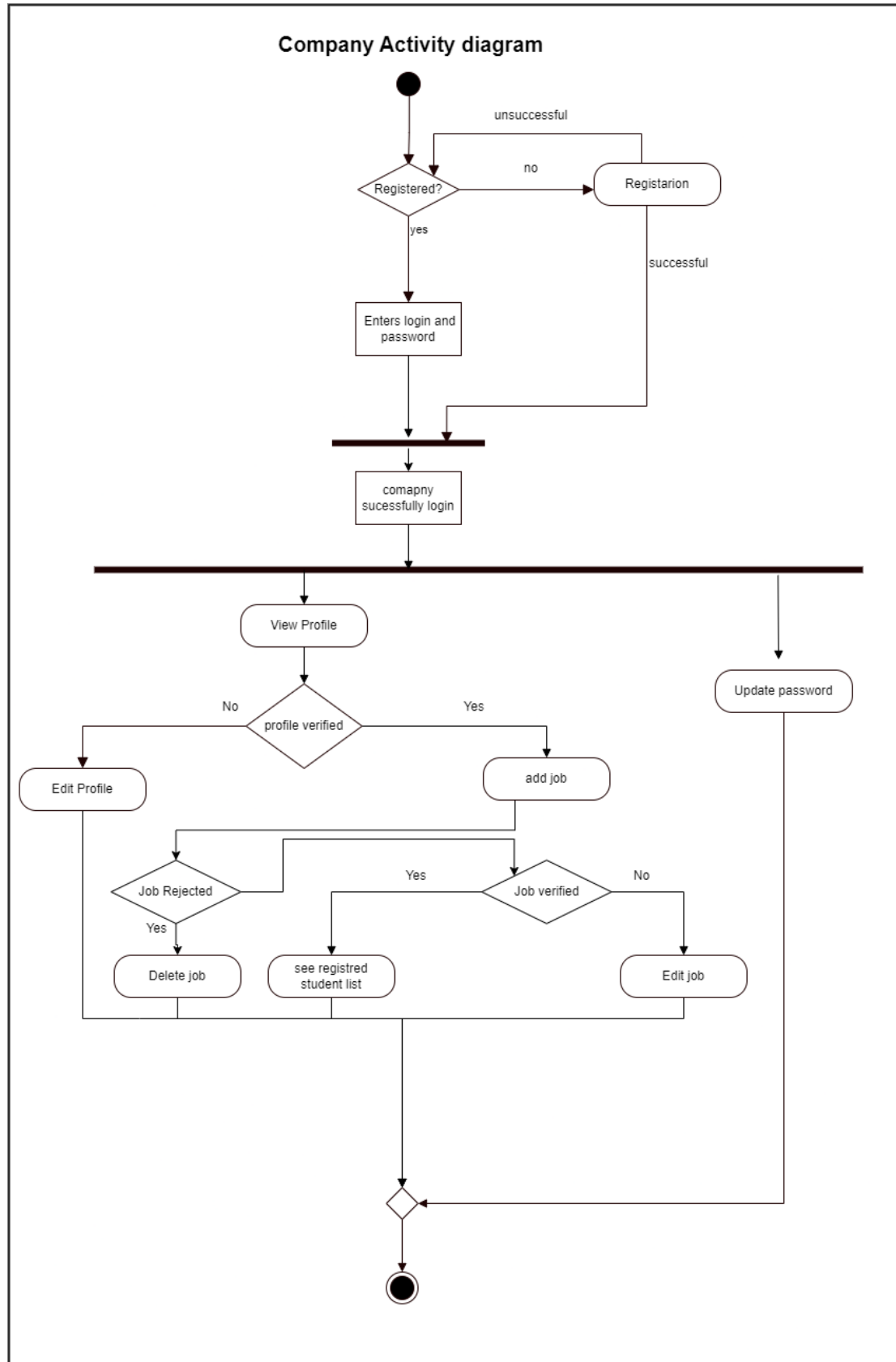
Appendix A: Analysis Models

Use Case Diagram

Here after the user has registered the login works and once the user has logged in the other use cases would work.
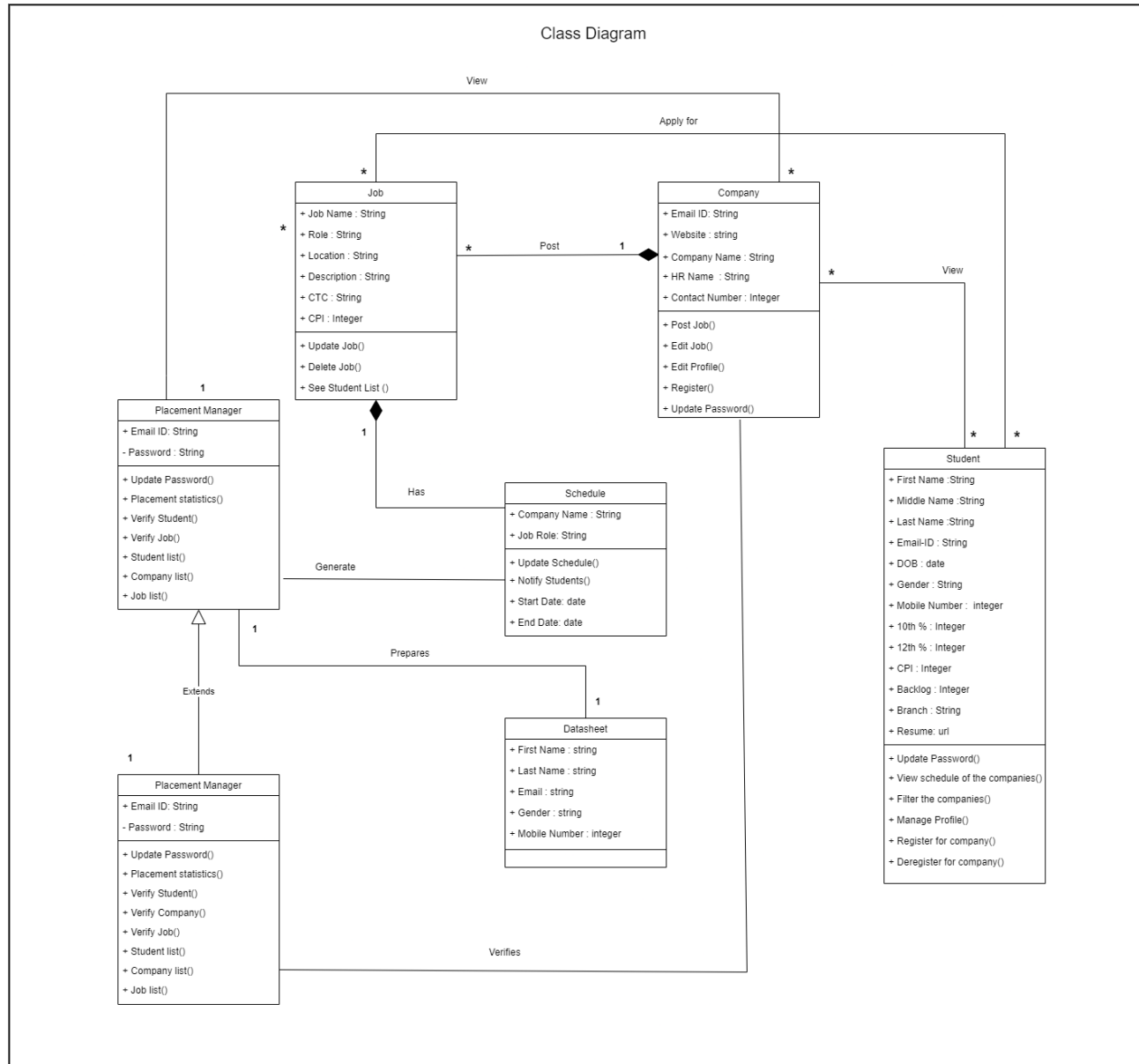
Activity Diagram

Student Activity Diagram

Company Activity diagram

Class Diagram

Class Diagram

View

Apply for

Job
+ Job Name : String
+ Role : String
+ Location : String
+ Description : String
+ CTC : String
+ CPI : Integer
+ Update Job()
+ Delete Job()
+ See Student List ()

Company
+ Email ID: String
+ Website : string
+ Company Name : String
+ HR Name : String
+ Contact Number : Integer
+ Post Job()
+ Edit Job()
+ Edit Profile()
+ Register()
+ Update Password()

Post

View

Placement Manager
+ Email ID: String
- Password : String
+ Update Password()
+ Placement statistics()
+ Verify Student()
+ Verify Job()
+ Student list()
+ Company list()
+ Job list()

Schedule
+ Company Name : String
+ Job Role: String
+ Update Schedule()
+ Notify Students()
+ Start Date: date
+ End Date: date

Student
+ First Name :String
+ Middle Name :String
+ Last Name :String
+ Email-ID : String
+ DOB : date
+ Gender : String
+ Mobile Number : integer
+ 10th % : Integer
+ 12th % : Integer
+ CPI : Integer
+ Backlog : Integer
+ Branch : String
+ Resume: url
+ Update Password()
+ View schedule of the companies()
+ Filter the companies()
+ Manage Profile()
+ Register for company()
+ Deregister for company()

Has

Generate

Prepares

Extends

Datasheet
+ First Name : string
+ Last Name : string
+ Email : string
+ Gender : string
+ Mobile Number : integer

Placement Manager
+ Email ID: String
- Password : String
+ Update Password()
+ Placement statistics()
+ Verify Student()
+ Verify Company()
+ Verify Job()
+ Student list()
+ Company list()
+ Job list()

Verifies

Sequence Diagram:

Student

:Authentication

:Database

:profile

:Job

:Admin

Student Registration

Verify Details

Student details added

Student Login

Verify Credentials

alt

Valid Credentials

[If credentials are valid]

Student profile page

[else]

Invalid Credentials

Error Page

edit Profile

alt

[If student is not verified]

Profile updated

profile updated

[else]

can't update profile

see company details

Apply Job

alt

[if criteria matched]

register successfully

added student list of company job

[else]

Registration Unsuccessful

Update password

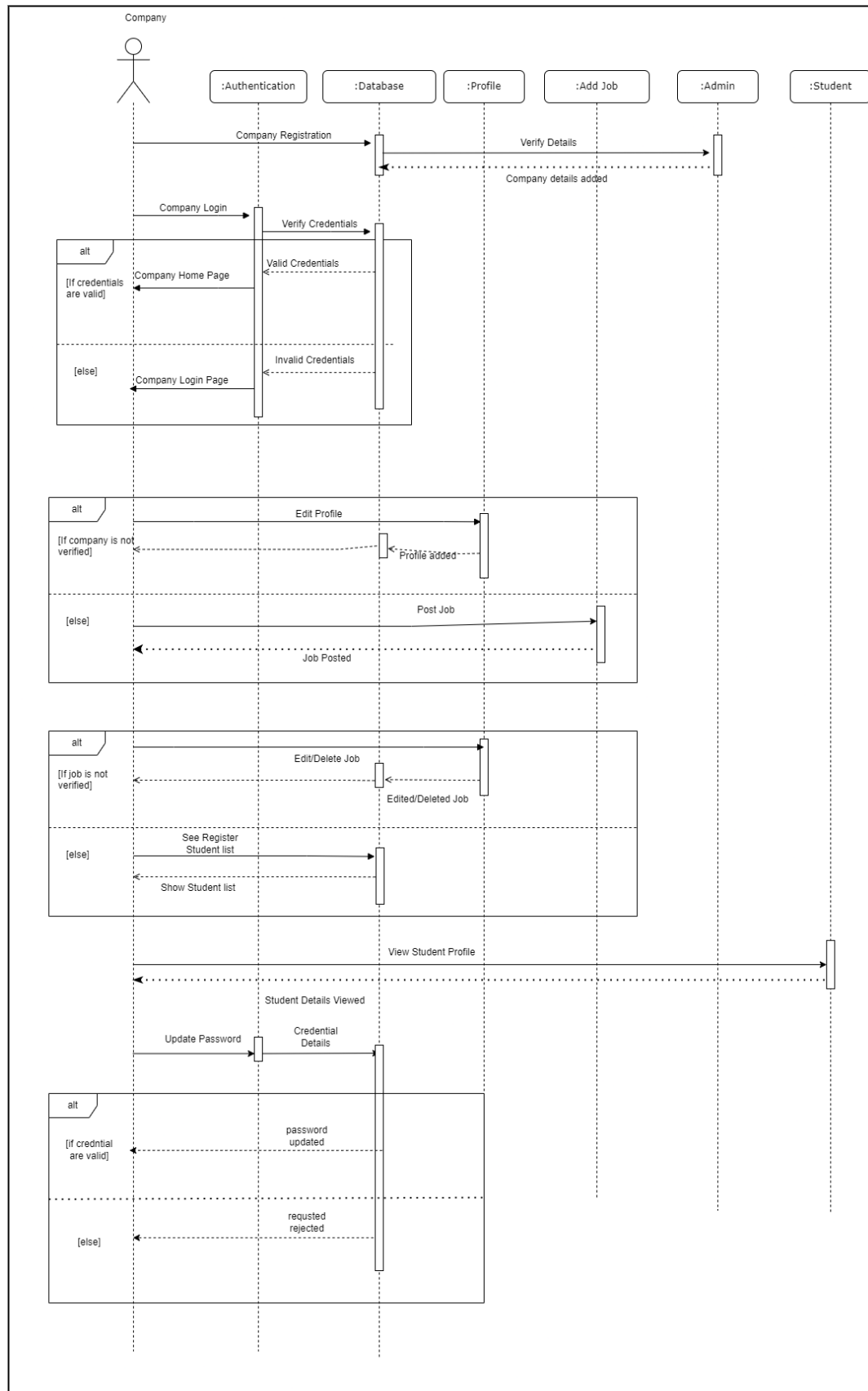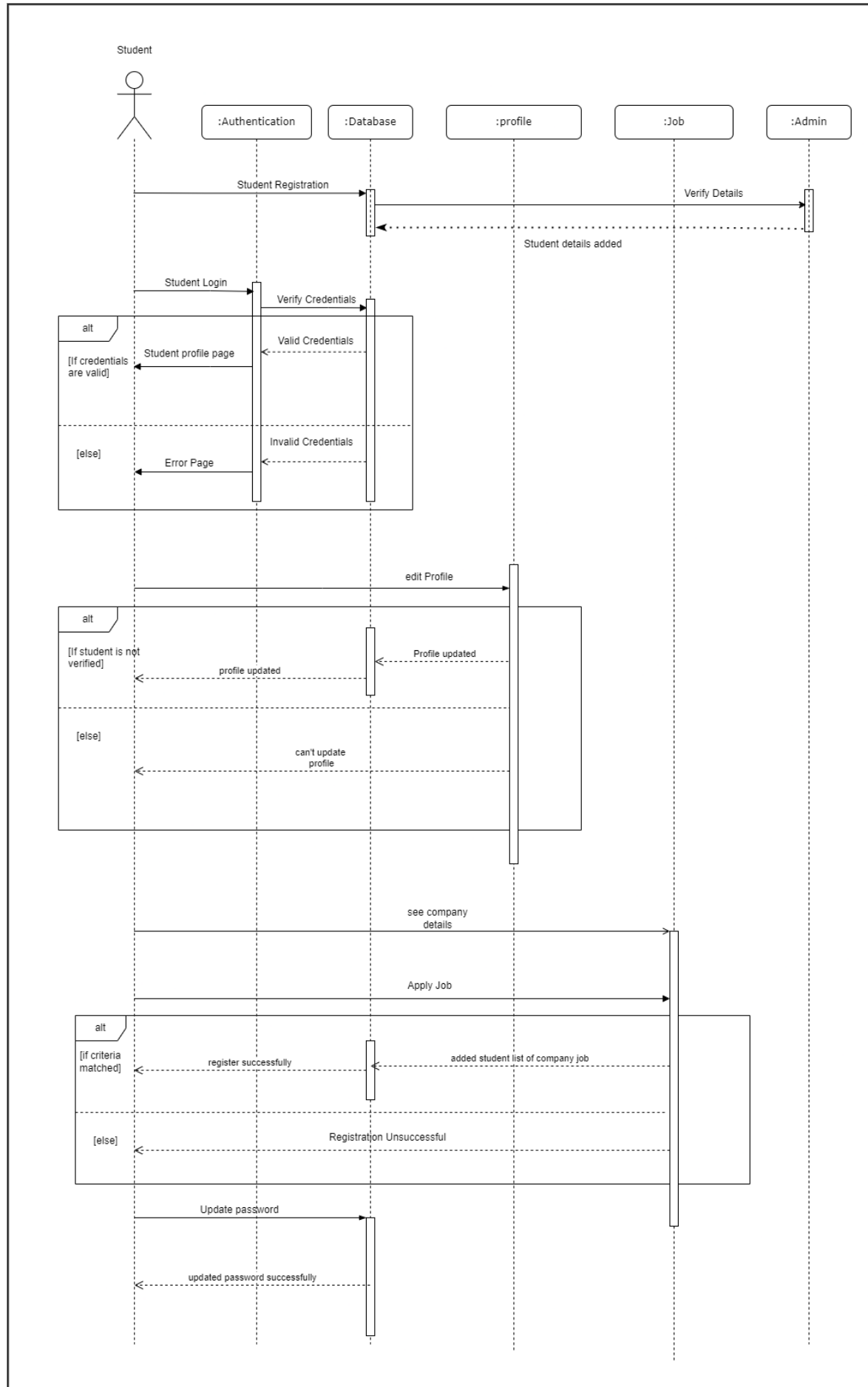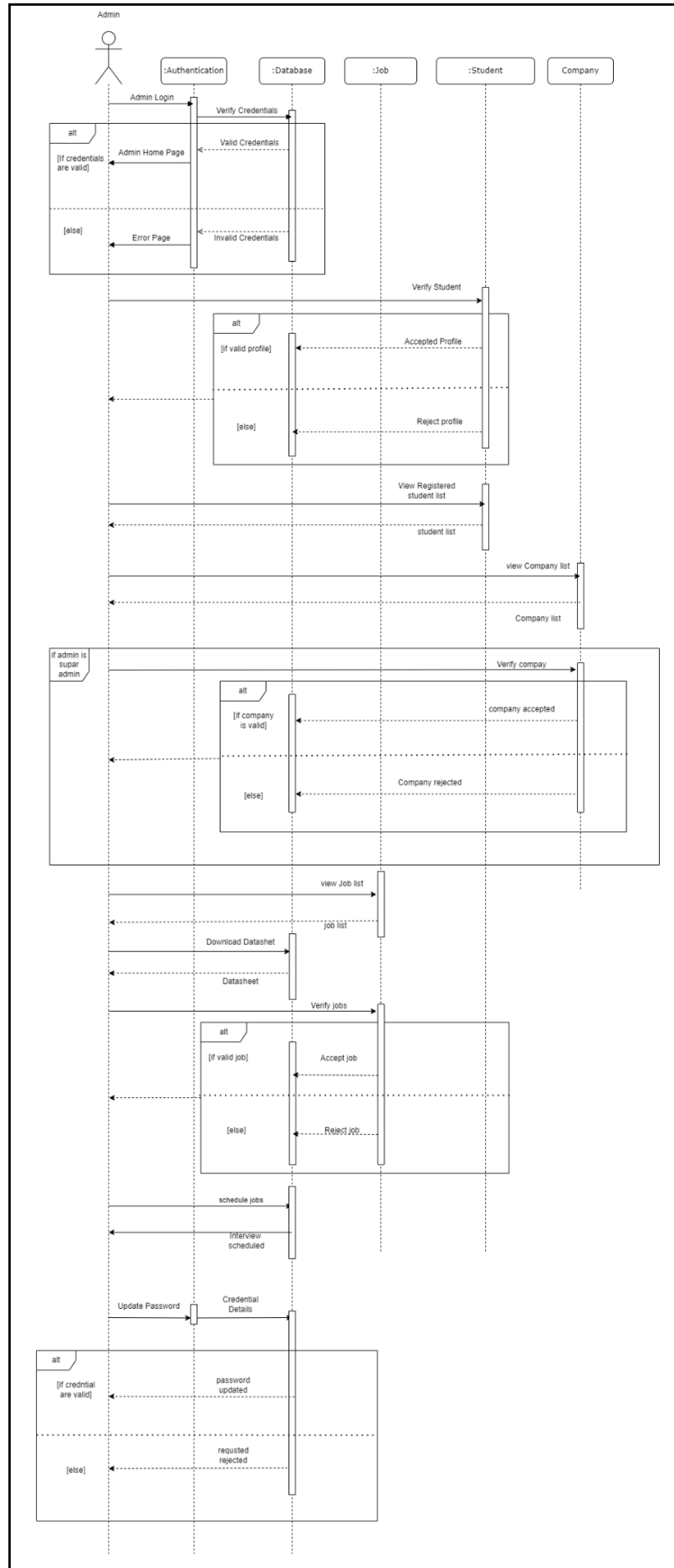updated password successfully

Appendix B: Issues List

Issues to be Solved

- In the Job schedule section of the admin, the end date can be less than the start date.
- In Student Registration, the student can enter future dates as the "Date of Birth".
- If the student or the company forgets the password, he/she won't be able to log in. We do not have the forgot password option.
- In the registration and manage profile pages, the user can enter numbers in the text-only section.

Requirements not implemented

- Interview Scheduling:
    - According to the interview timing, there should be a video conference call on the website at the time of the interview.
- Student course information and the endorsements of the faculty are not shown on the portal.
- Data Analysis:
    - A system for tracking and analyzing student information and recruitment trends, allowing for the placement cell to spot trends and identify areas for improvement.
- Resume Builder:
    - A tool for students to create and manage their resumes, which can be made available to recruiters.
- Transportation/Location:
    - Companies will be able to view various transportation options to reach the college campus.