

Requirements Analysis and Specification Document (RASD)

InternHub - Students&Companies (S&C) Platform

Version 2.0

*Prepared by: Shreesh Kumar Jha, Samarth Bhatia and Satvik Sharma
19 December 2024*

Table of Contents

1. [Introduction](#)
 2. [Overall Description](#)
 3. [Specific Requirements](#)
 4. [Formal Analysis Using Alloy](#)
 5. [Effort Spent](#)
 6. [References](#)
-

1. Introduction

1.1 Purpose

This Requirements Analysis and Specification Document (RASD) provides a comprehensive overview of the InternHub - Students & Companies (S&C) platform. Its primary purpose is to serve as both a guide for developers responsible for implementing the system specifications and as a contractual reference point for clients and contractors. Additionally, it offers a clear, precise, and unambiguous explanation of the platform's features and limitations, empowering students, companies, and academic institutions to confirm that the system meets their needs and requirements.

The S&C platform's overarching goal is to transform how university students connect with companies for internships. To this end, it focuses on:

1. Establishing an efficient system that matches students with suitable internship opportunities.
2. Streamlining the entire internship lifecycle—from application through completion—to simplify both student and company workflows.
3. Utilizing smart recommendation algorithms to align student skills with company requirements, ensuring more accurate and beneficial matches.
4. Providing robust monitoring and feedback tools to enhance transparency, accountability, and continuous improvement.
5. Ensuring effective complaint management and maintaining high-quality standards throughout the internship process.

By offering a seamless and impactful experience, the S&C platform aims to serve as a trusted solution that addresses the requirements of all stakeholders—students, businesses, and universities—thereby ensuring a more efficient, productive, and rewarding internship ecosystem.

1.1.1 Goals

ID	Description
G1	Enable students to create detailed profiles, including their CVs, skills, academic achievements, and interests.
G2	Allow companies to post comprehensive internship opportunities, detailing roles, requirements, benefits, and timelines.
G3	Provide intelligent recommendations that align student skills and preferences with internship opportunities.
G4	Equip universities with tools to effectively monitor, manage, and track student internship progress and performance.
G5	Implement feedback and rating systems to promote accountability and continuous improvement for both students and companies.
G6	Facilitate seamless communication between students, companies, and universities for better collaboration and coordination.
G7	Integrate a secure document management system for handling internship-related paperwork, such as contracts and certificates.
G8	Offer analytics and reporting tools to provide insights into internship trends, success rates, and areas for improvement.

G9	Support multilingual functionality to ensure accessibility for a diverse user base across regions.
G10	Implement a grievance redressal mechanism to resolve disputes and ensure fair treatment for all users.
G11	Provide training modules or resources to prepare students for internships, such as interview tips and skill-building exercises.

1.2 Scope

World Phenomena:

The project addresses core operational areas that shape the interactions and processes among students, companies, and universities. These include:

1. **Student Internship Process:** Students search for and apply to internships, creating personal profiles and uploading their CVs.
 - *ID: WP1* Students create profiles and upload their CVs.
 - *ID: WP4* Students apply for internships.
2. **Company Internship Management:** Companies post internship opportunities, define requirements, and review applications to select suitable candidates.
 - *ID: WP2* Companies post internship opportunities.
 - *ID: WP5* Companies review applications and select candidates.
3. **University Oversight:** Universities maintain a supervisory role, monitoring and managing internship activities to ensure quality and compliance.
 - *ID: WP3* Universities monitor internship progress.
4. **Interview Coordination and Selection:** The platform facilitates the scheduling and management of interviews and other selection procedures to ensure a smooth hiring process.
5. **Feedback and Quality Assurance:** Feedback is collected from students, companies, and universities to ensure continuous improvement, enhanced user experience, and adherence to quality standards.
 - *ID: WP6* Feedback is collected from students and companies.
6. **Complaint Handling:** A robust complaint-handling system ensures issues are addressed efficiently, maintaining trust and transparency in all interactions.
7. **Fair and Transparent Interactions:** The system is designed to foster an environment of fairness, transparency, and accountability among students, companies, and universities.

Shared Phenomena:

The platform provides a suite of shared functionalities, ensuring seamless interaction and data exchange among all stakeholders (Students, Companies, and Universities):

1. User Account Management and Profiles:

- *ID: SP1* Students create accounts on the platform (Controller: Student, Observer: Platform).
- *ID: SP2* Companies create accounts on the platform (Controller: Company, Observer: Platform).
- *ID: SP3* Universities create accounts on the platform (Controller: University, Observer: Platform).

2. Students maintain detailed profiles, including CVs, skills, and achievements, while companies and universities manage their respective institutional profiles.

3. Internship Postings and Applications:

Students can apply directly to posted internships, and companies can review and manage these applications.

- *ID: SP4* Students apply for internships (Controller: Student, Observer: Platform).
- *ID: SP5* Companies review and manage applications (Controller: Company, Observer: Platform).
- *ID: SP6* Universities track student applications (Controller: University, Observer: Platform).

4. Feedback and Rating System:

A comprehensive feedback mechanism enables stakeholders to exchange feedback, ratings, and reviews to ensure accountability and continuous improvement.

- *ID: SP7* Feedback is exchanged between stakeholders (Controller: All, Observer: Platform).

5. Interview Scheduling and Notifications:

Interviews are coordinated efficiently, with automated reminders and updates for both students and companies.

6. Communication and Support Tools:

The platform offers seamless communication channels for queries, updates, and issue resolution among students, companies, and universities.

7. Document Management:

Secure storage and sharing capabilities for documents, such as internship agreements, certificates, and other relevant files, ensure easy access and proper record-keeping.

8. Analytics and Insights:

Real-time analytics and reporting tools help all parties make informed decisions, monitor internship progress, and assess performance metrics.

9. Multilingual Support:

The platform supports multiple languages, accommodating a diverse global user base.

10. Training and Preparation Resources:

Students have access to resources like training materials and interview preparation tools, aiding them in securing and succeeding in internships.

1.3 Definitions, Acronyms, Abbreviations

Term/Acronym	Definition
S&C	Students&Companies Platform
RASD	Requirements Analysis and Specification Document
CV	Curriculum Vitae
UI	User Interface
API	Application Programming Interface
DBMS	Database Management System
SLA	Service Level Agreement
GDPR	General Data Protection Regulation

1.4 Revision History

Version	Date	Description	Authors
0.1	8 December 2024	Initial Release	Shreesh Kumar Jha, Samarth Bhatia, Satvik Sharma
1.0	17 December 2024	Structure Fix and Added Use Cases	Shreesh Kumar Jha Samarth Bhatia
2.0	19 December 2024	Alloy Modelling	Shreesh Kumar Jha Samarth Bhatia

1.5 Document Structure

As shown below, the document is organized into six sections, each with a distinct focus:

1. Introduction:

The project's goals, purpose, and a succinct analysis of common and worldwide occurrences are presented in the introduction, containing acronyms and definitions to help you grasp the problem domain.

- 2. Overall Description:**
Provides a thorough rundown of the issue, potential domains, and features of the product. Explains limitations, dependencies, and assumptions as well.
- 3. Specific needs:**
Provides a detailed description of both functional and non-functional needs, including those pertaining to external interfaces.
- 4. Formal Evaluation Using Alloy:**
Provides assertions and checks to validate the model outlined in previous parts.
- 5. Effort Spent:**
Describes how each team member contributed to the writing of this paper.
- 6. References:**
Provides a list of all the supplementary materials and references that were utilized to produce the document.

2. Overall Description

2.1 Product Perspective

2.1.1 | Scenarios

Scenario 1: A student registers

A university student named Amelia Young chooses to work for S&C while searching for an internship. After accessing the platform's home page, she selects the "Sign-Up" option and inputs her email address, name, and a strong password. She gets a confirmation email after completing the form. Amelia activates her account by clicking on the confirmation link. After finishing, she logs on to look for internship possibilities.

Scenario 2: A Company posts an internship

Interns are needed for the data analytics team at InnovateCorp, a mid-sized software startup. After logging in to S&C, an HR representative from the organization fills out the "Post Internship" part, which includes the job title, description, length, and needed qualifications. They also establish a deadline for applications. After submission, S&C gives InnovateCorp a confirmation email and contacts the appropriate students based on their profiles.

Scenario 3: A University administrator monitors internships

Greenfield University academic coordinator Dr. Olivia Cruz uses S&C to monitor the development of her students' internships. She goes to the "Internship Monitoring" section, where she may see comprehensive reports, examine student and company comments, and respond to complaints made by either side.

Scenario 4: A Student applies for an internship

Liam Chen, a computer science major, looks for software development internships on S&C. He discovers a position at InnovateCorp that suits him after using filters to refine the results. After

reading the job description, Liam selects "Apply Now" and sends in his resume. On his dashboard, the application status is updated.

Scenario 5:A Company reviews applications

Notifications of new applications are sent to InnovateCorp's HR department. They access Liam's profile and resume by logging in to the "Application Management" area. Following screening, Liam is placed on their shortlist, and his dashboard is automatically updated with the interview time.

System Context

The Students & Companies (S&C) platform is a dynamic web-based solution designed to streamline the internship process by bridging the gap between students, companies, and universities. The platform caters to three primary user groups, each with distinct roles and requirements:

1. University Students:

Students utilize the platform to search for internship opportunities, create detailed profiles with CVs and skills, and track their application progress.

2. Companies:

Organizations post internship positions, outline role requirements, and select suitable candidates based on student profiles and recommendations.

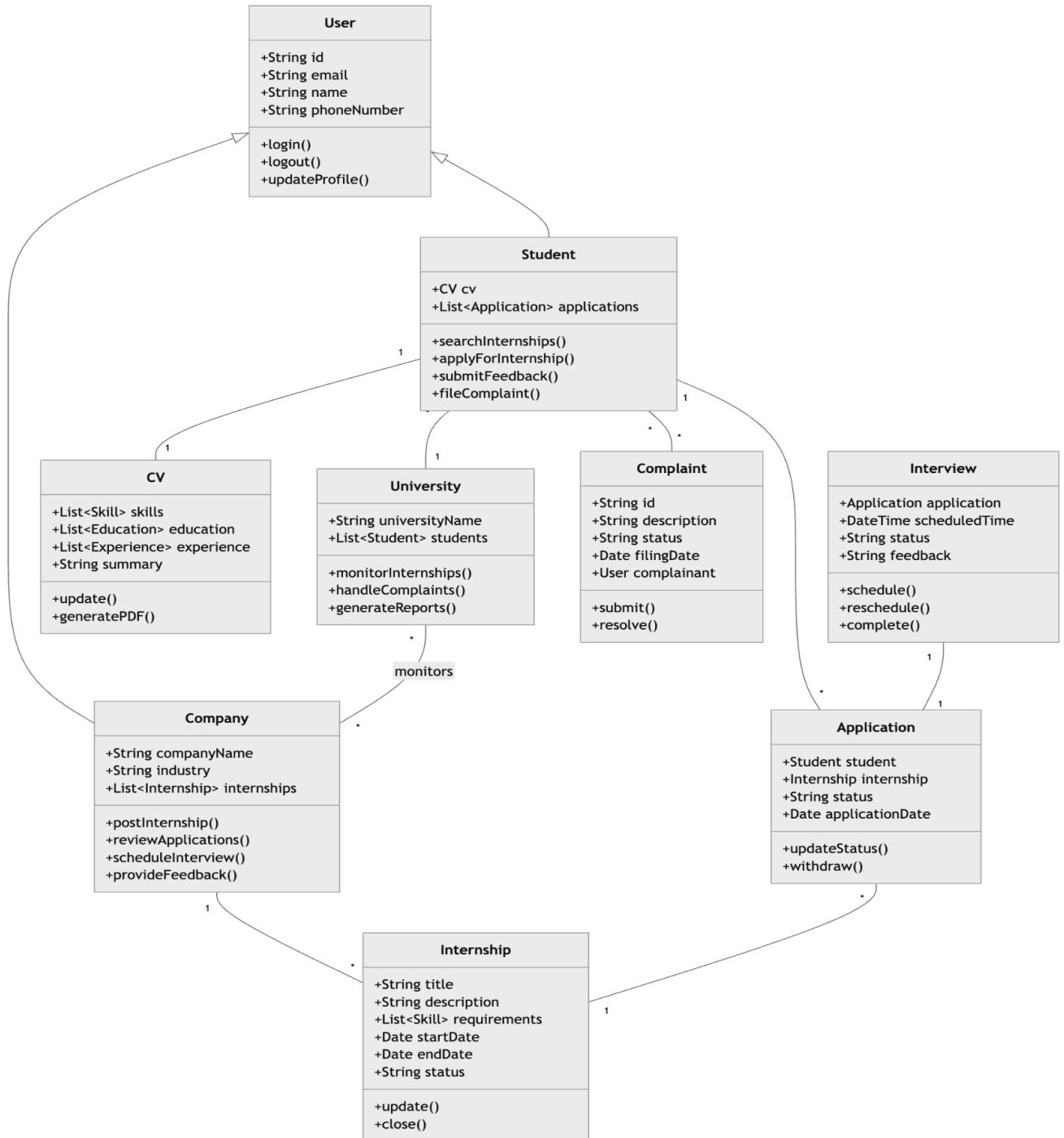
3. University Administrators:

Universities leverage the platform to oversee the entire internship process, monitor student performance, and ensure alignment with academic standards.

4. Central Platform:

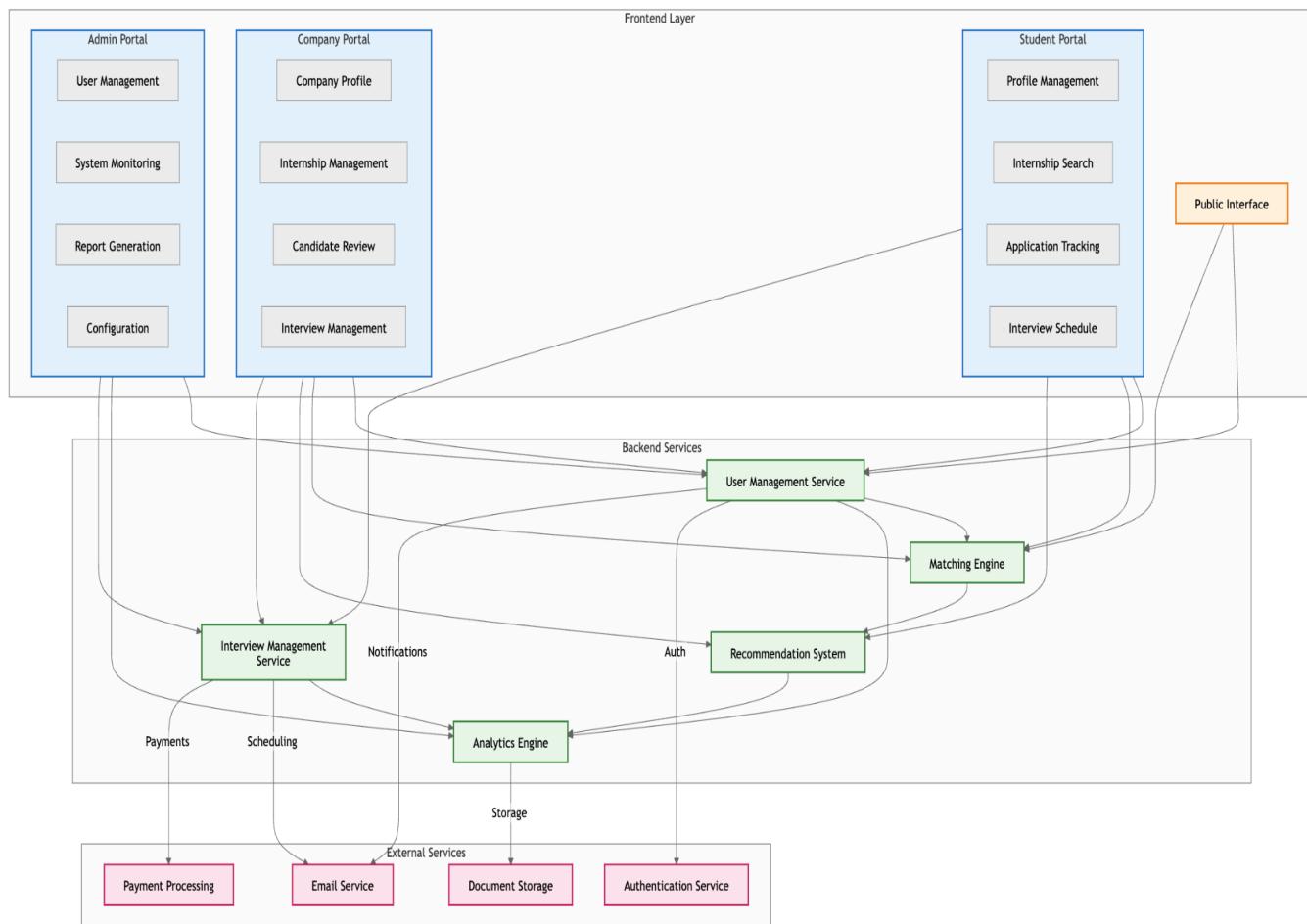
The S&C platform serves as a centralized hub, fostering efficient communication, reducing administrative overhead, and ensuring a seamless experience for all stakeholders involved.

2.1.2 | Class Diagram



Fig(): Class Diagram for InternHub

2.1.3 System Architecture



Fig(): System Architecture Diagram for InternHub

The platform consists of:

The following elements make up the modular architecture of the S&C platform:

- 1) Front-end Layer:
 - a. Student Portal: A place where students may keep track of applications, maintain profiles, and look for internships.
 - b. Company Portal: Resources for employers to advertise openings, evaluate applicants, and offer comments.
 - c. University Administrator Portal: An interface that allows colleges to handle reports and keep an eye on internships.
 - d. The public interface lets users explore internships and discover more about the platform.
- 2) Services for the backend:
 - a) Suggestion for a User Management Service
 - b) Matching Engine
 - c) Recommendation System

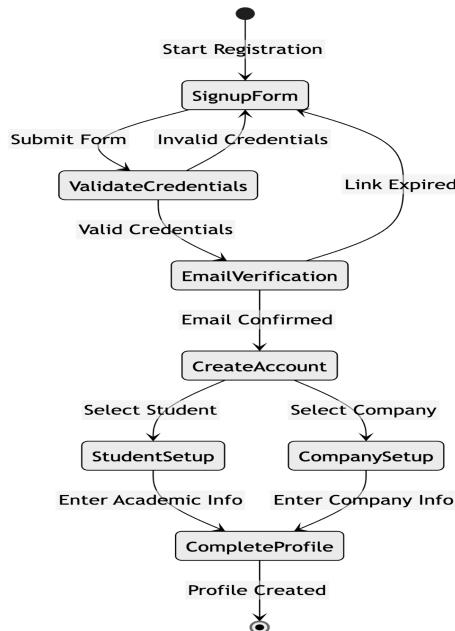
- d) Interview Management Service
 - e) External Integrations for the Service Analytics Engine:
- 3) Email Support
- a) Authentication Service for Document Storage Systems
 - b) Processing payments for internships that are compensated
 - c) Authentication Service
 - d) Payment Processing (for paid internships)

2.1.4 State Diagrams

The State Diagrams of the S&C system, which depict every action a user could take, are shown in this section.

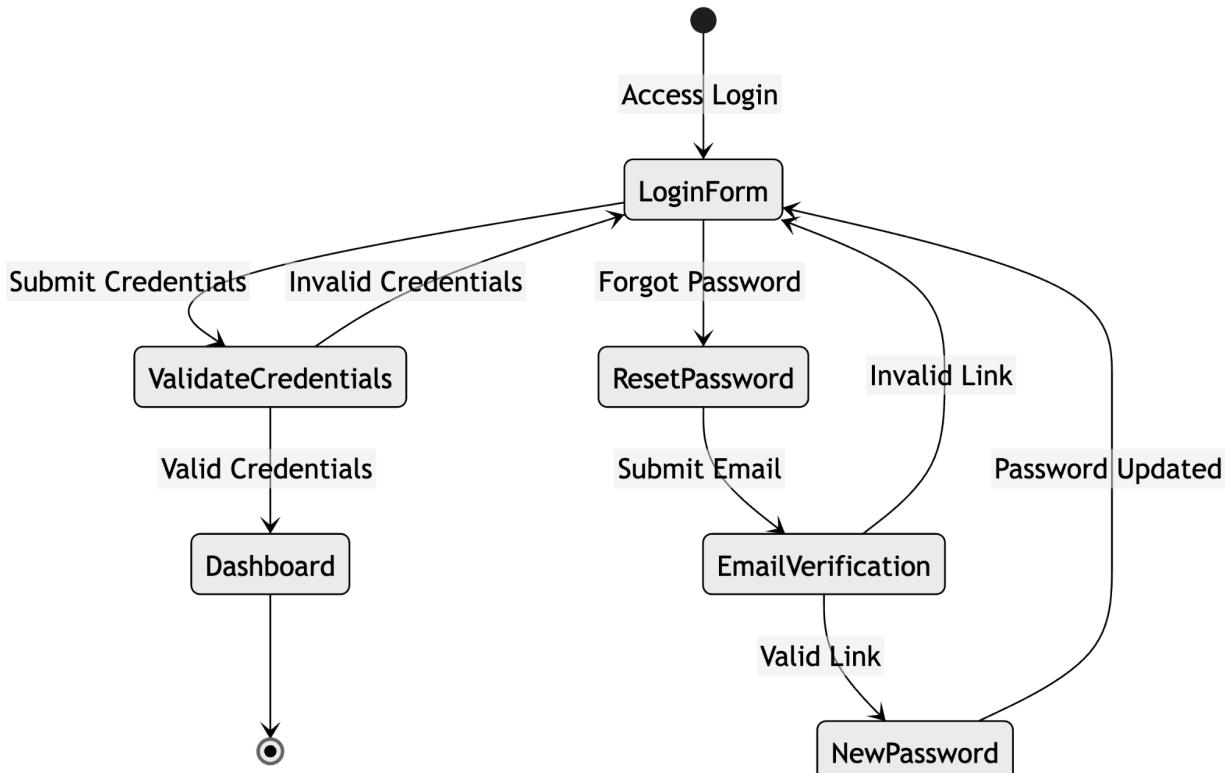
SignUp: When a user wants to register on S&C, they must fill out a registration form with their email address, password, name, and last name. S&C will send the user a verification email if the credentials they have provided are accepted (that is, if the password satisfies security requirements and the email address is not already in use). The new account is successfully created after the user confirms their registration using the email link that was provided. S&C displays an error notice to the user and reroutes them to the signup form page if the credentials are incorrect.

Additionally, S&C will ask for more company-specific details including the company name, industry, and size if the user chooses the "Register as a Company" option during the registration process. To construct a comprehensive company profile, these details are necessary.



Fig(): State Diagram for Sign Up

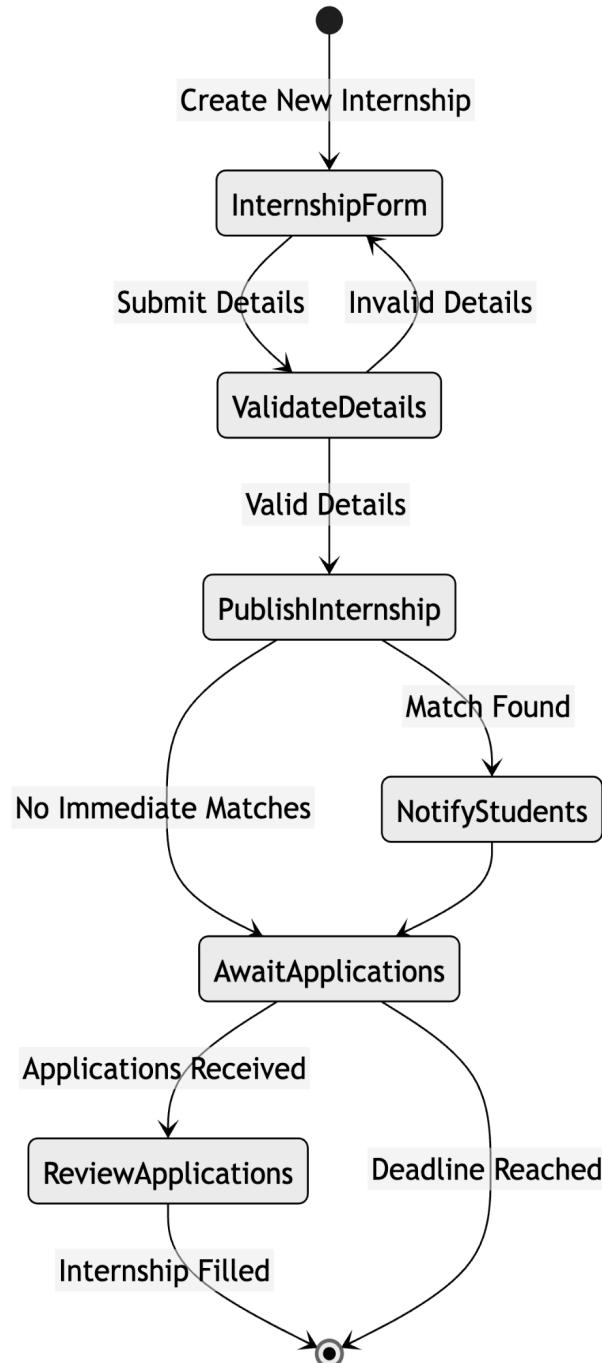
Login: A registered user must fill out a login form using their email address and password in order to access their S&C account. S&C presents the User's dashboard, which highlights pertinent internships and applications according to their function, if the credentials supplied are correct and correspond to those of a registered User in the S&C database. S&C returns the user to the login form page after displaying an error notice if the credentials entered are incorrect.



Fig(): State Diagram for Login

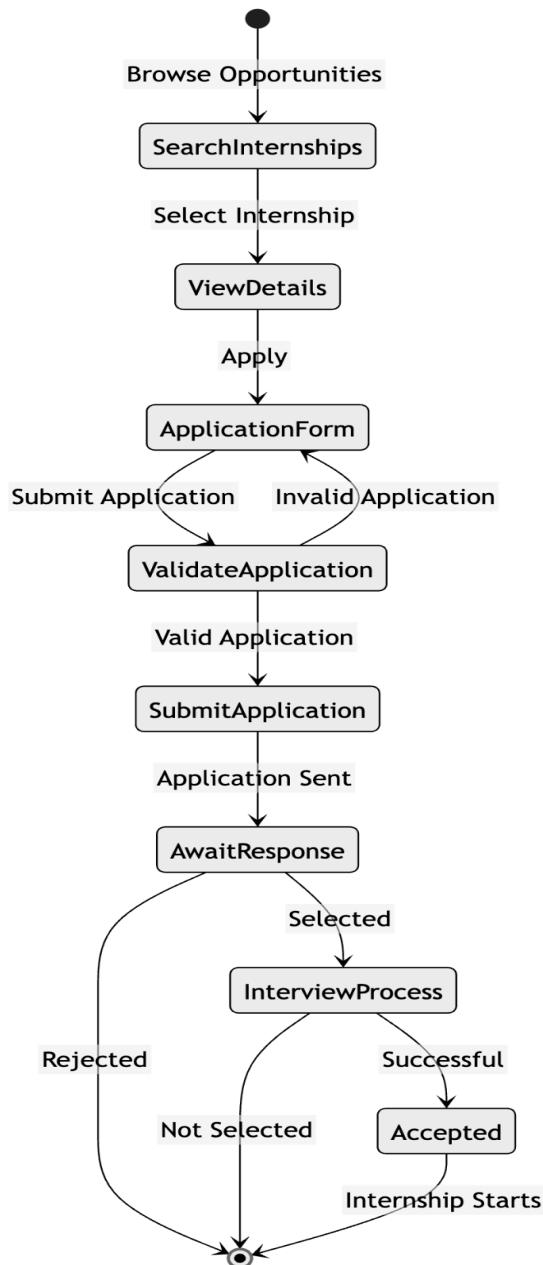
Post Internship: Companies must include a number of parameters in the create internship form when they plan to post a new internship on S&C. The job title, description, requirements, length of service, and any other information, such pay or perks, are examples of these factors. S&C shows the Company an error message and reroutes them back to the creation form if the system determines that any of these parameters are inappropriate.

On the other hand, S&C creates the internship posting if every parameter satisfies the system's requirements. Notifications are subsequently delivered to matching students based on their profiles and preferences after the newly produced posting is published to the Company's dashboard.



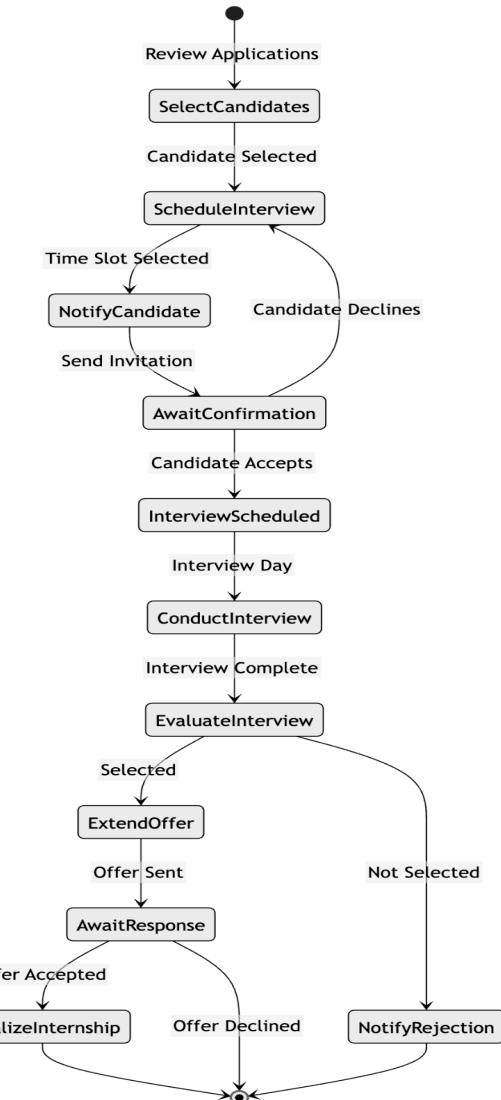
Fig(): State Diagram for Post Internship

Apply for Internship: Students can use the platform to apply for internships when they locate one that interests them. The application form, which could contain extra questions unique to the role, must be filled out by the student. S&C confirms that the student satisfies the prerequisites and that the application is complete. The application is sent to the business if it is accepted, and the student can monitor its progress via their dashboard. S&C lets the student finish the application after displaying an error message if any necessary information is lacking.



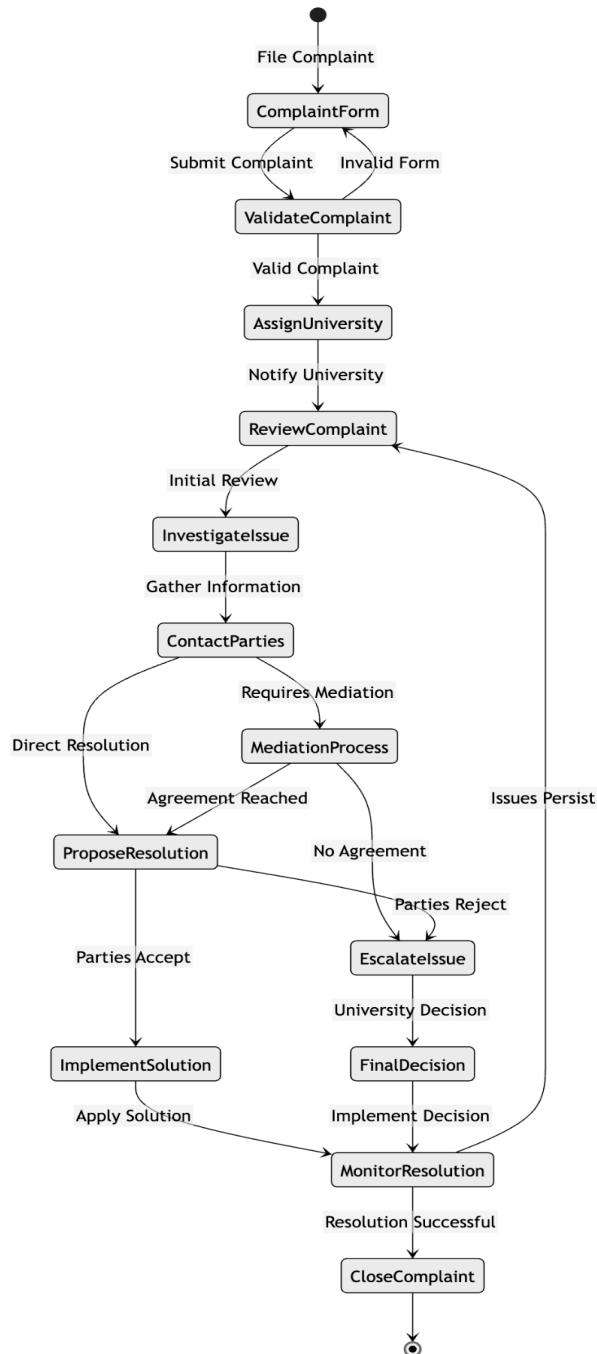
Fig(): State Diagram for Applying for an Internship

Interview Management: A business can use S&C to start the interview process after reviewing applications and choosing applicants for interviews. The candidate is informed of the time slots that are available once the organization makes a selection. Other times can be suggested or accepted by the student. Notifications and calendar invitations are sent to both parties after confirmation. Following the interview, the business can document the results and move on with its choice.



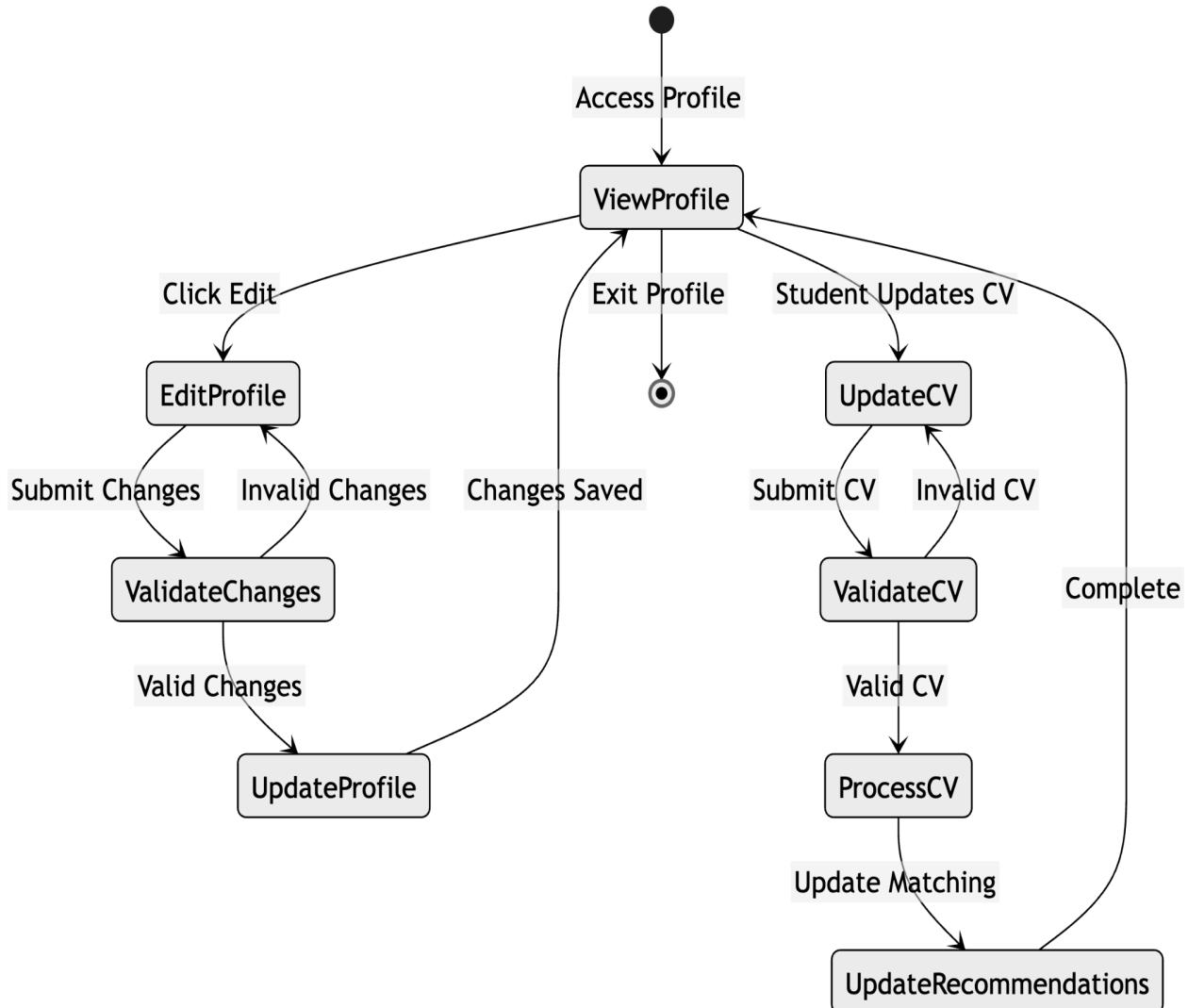
Fig(): State Diagram for Interview Management

Complaint Handling: Through S&C, any party that experiences problems during the internship process can submit a complaint. A description of the problem and any pertinent documentation must be included in the complaint form. The relevant university official receives the complaint once it is filed. The university can look into the matter, get in touch with those involved, and try to find a solution. Every conversation and choice is recorded in the system.



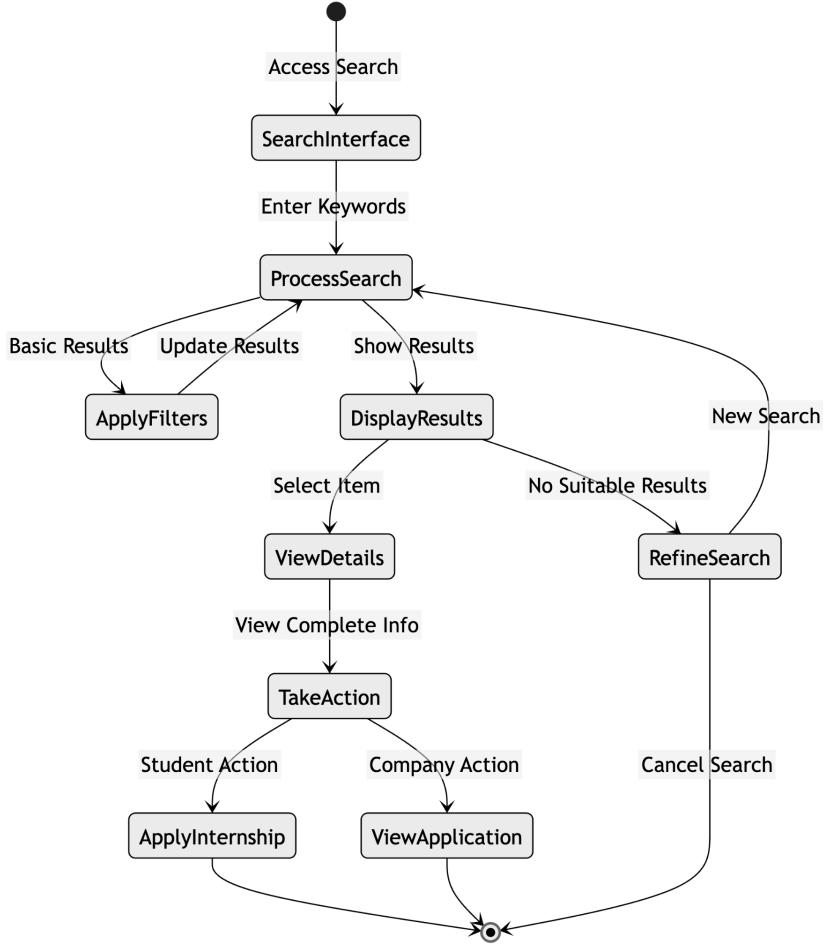
Fig(): State Diagram for Complaint Handling

Profile Management: Anytime they choose, users can view and edit their profiles. For students, this entails revising their resume, preferences, and talents. For businesses, this entails revising internship requirements and company information. S&C verifies all modifications before they are saved. Additionally, in response to changes in profiles, the algorithm immediately updates matching recommendations.



Fig(): State Diagram for Profile Management

Search and Filter: The platform's search feature allows users to look for pertinent information. Students can use industry, geography, keywords, and other parameters to find internships. Employers can look up candidate profiles by experience, education, and skill set. Additional filters can be used to further refine search results, and users can remember their search choices for later use.



Fig(): State Diagram for Search and Filter

2.2 Product Functions

Core Functions

For Students:

1. **Manage Profiles and CVs:** Students have the ability to submit their resumes for prospective employers to see as well as create and edit their profiles.
2. **Search and Filter Internships:** Students can utilize search and filtering options to identify internships that fit their interests, location, and duration.
3. **Monitor Applications:** Students can keep tabs on the progress of their internship applications and get alerts when there are any changes or rulings.
4. **Schedule Interviews:** Depending on the company's availability, students can choose the times they want to interview for internships.
5. **Submit Feedback:** Students can use the platform to provide feedback and comments regarding their internship experience after it has ended.

6. **File Complaints:** Using the portal, students can directly report problems or complaints pertaining to internships.

Management of Company Profiles:

1. **Manage Company Profiles:** Companies have the ability to build, edit, and maintain profiles that contain contact information and organizational details.
2. **Post Internships:** Businesses are able to publish internship openings along with comprehensive requirements and descriptions.
3. **Screen Candidates:** Employers have the ability to examine applications and sort applicants according to their backgrounds, profiles, and other requirements.
4. **Conduct Interviews:** Employers can use the site to immediately schedule and interview shortlisted individuals.
5. **Gather Feedback:** Following an interview or after an internship, employers might offer input on applicants.
6. **Monitor Performance:** During an intern's internship, businesses can keep tabs on their performance.

University Performance:

1. **Observe Student Activity:** Academic institutions have the ability to keep an eye on students' applications, profiles, and internship progress.
2. **Oversee Internships:** Academic institutions can examine information regarding current internships and how well they fit with their objectives.
3. **Handle Complaints:** Universities have the authority to handle and settle complaints about internships made by businesses or students.
4. **Generate Reports:** Academic institutions have the ability to produce reports on internship data, student accomplishments, and employer opinions.
5. **Ensure Quality Assurance:** Academic institutions are able to assess internships' quality and make sure they adhere to institutional requirements.

2.3 User Characteristics

Students

1. Students at universities looking for internships.
2. Demographics: University students from diverse academic backgrounds and skill levels.
3. Motivations: Actively seeking internships to enhance their academic learning, gain practical experience, and develop their careers.
4. Technical Proficiency: Basic computer literacy, with varying levels of technical expertise depending on their academic discipline.
5. Access Requirements: Use web browsers and mobile devices for seamless access to the platform.

6. Engagement Needs: A user-friendly interface with clear guidance for application processes, personalized internship recommendations, and progress tracking is required.

Companies

1. Demographics: Organizations ranging from startups to multinational corporations across various industries.
2. Key Users:
 - a. HR Personnel: Responsible for posting internships, managing candidate applications, and coordinating interviews.
 - b. Department Managers: Assess technical and domain-specific skills of applicants.
 - c. Technical and Non-Technical Staff: Engage with interns for mentorship and project collaboration.
3. User Roles: Flexible role-based access for different functionalities, such as job posting, candidate review, and administrative controls.
4. Engagement Needs: Require tools for streamlining recruitment processes, accessing applicant analytics, and maintaining compliance with university policies.

University Administrators

1. Key Roles:
 - a. Academic Coordinators: Oversee the alignment of internships with educational goals and curriculum requirements.
 - b. Internship Program Managers: Monitor internship program effectiveness, gather feedback, and ensure compliance with regulations.
 - c. Student Advisors: Guide students on selecting suitable internships and navigating application processes.
 - d. System Administrators: Maintain the platform, manage user accounts, and ensure data security.
2. Engagement Needs: Require detailed dashboards for tracking student participation, company partnerships, and program outcomes. Need tools for generating reports and managing communications between stakeholders

2.4 Assumptions, Dependencies and Constraints

2.4.1 | Domain Assumptions

ID	Description
DA1	Users have reliable internet access.
DA2	Students maintain updated CVs.

DA3	Companies provide accurate information about internships.
DA4	Universities actively monitor students' progress on S&C.
DA5	All users comply with the platform's terms and policies.

2.4.2 | Dependencies

ID	Description
D1	Reliable web hosting services.
D2	Functional email delivery system.
D3	Secure database system.
D4	File storage system for documents.
D5	Authentication services to manage user logins.

2.4.3 | Constraints

ID	Description
C1	Must comply with GDPR and data protection regulations.
C2	Adhere to university-specific internship regulations.
C3	Ensure system performance under high traffic.
C4	Provide compatibility across major web browsers.

C5	Implement robust security standards for data handling.
----	--

3. Specific Requirements

3.1 External Interface Requirements

User Interfaces:

Students, businesses, and university administrators will be able to access the InternHub - Students&Companies (S&C) platform's user interface via a responsive web application on any device with a current web browser and an internet connection.

Authentication Interface:

1. User login and registration.
2. Password reset functionality.
3. Multi-factor authentication (optional).

Welcome Back

Sign in to your S&C account

Email

Password

Remember me [Forgot password?](#)

Don't have an account?

Create Account

Join S&C to start your journey

 Student

 Company

First Name

Enter first name

Last Name

Enter last name

University

Enter your university

Email

Enter your email

Password

Create password

Create Account

Already have an account?

Sign in

Student Interface:

The dashboard features a top navigation bar with a user profile icon (JD) and a bell icon. Below the header, there are three cards: "Active Applications" (5), "Upcoming Interviews" (2), and "New Messages" (3). The main content area is divided into two sections: "Recent Internship Matches" and "Upcoming Interviews".

Recent Internship Matches

Based on your profile and preferences

- Software Developer Intern**
TechCorp Inc.
React Node.js
[Apply Now](#)
- Software Developer Intern**
TechCorp Inc.
React Node.js
[Apply Now](#)
- Software Developer Intern**
TechCorp Inc.
React Node.js
[Apply Now](#)

Upcoming Interviews

- Frontend Developer**
WebSolutions Ltd
Tomorrow at 10:00 AM
- Frontend Developer**
WebSolutions Ltd
Tomorrow at 10:00 AM

1. Dashboard Overview: Shows notifications, recommended internships, and active applications.
2. Profile management: Add or update qualifications and edit personal information.
3. CV Builder: Use built-in templates to create and manage resumes.
4. Internship Search: Look for and select internships according to talents, domain, and location.
5. Application Tracker: Check the progress of applications that have been submitted.
6. Organize interview dates with the Interview Calendar.
7. Messaging System: Interact with administrators and businesses.

Company Interface:

The dashboard features a header with the TechCorp logo and a 'Post New Internship' button. Below are four summary cards: Active Positions (8), Total Applicants (156), Scheduled Interviews (12), and Hired This Month (5). The main area contains two sections: 'Recent Applications' and 'Today's Interviews'. 'Recent Applications' lists three entries for Sarah Parker, each with a 'Schedule Interview' and 'Review' button. 'Today's Interviews' lists three entries for John Doe, each with a 'Review' button.

Category	Count
Active Positions	8
Total Applicants	156
Scheduled Interviews	12
Hired This Month	5

Section	Applicant / Interviewer	Position	Skills	Action Buttons
Recent Applications	Sarah Parker	Frontend Developer Position	React 3 Years Exp.	Schedule Interview Review
	Sarah Parker	Frontend Developer Position	React 3 Years Exp.	Schedule Interview Review
	Sarah Parker	Frontend Developer Position	React 3 Years Exp.	Schedule Interview Review
Today's Interviews	John Doe	Backend Developer	2:00 PM - 3:00 PM	Review
	John Doe	Backend Developer	2:00 PM - 3:00 PM	Review
	John Doe	Backend Developer	2:00 PM - 3:00 PM	Review

1. Dashboard Overview: Displays applications, postings, and other data.
2. Management of Company Profiles: Modify logos and company information.
3. Management of Internship Postings: Produce and oversee internship advertisements.
4. Managing Candidates: Examine applications and create a shortlist of applicants.
5. Use an interview scheduler to plan and monitor interviews.
6. Analytics Dashboard: Information on posting engagement and candidate performance.

Administrator Interface:

The screenshot displays the University Admin Dashboard with the following sections:

- User Management:** Manages system users and their permissions. It shows 1,234 total users, 23 active complaints, 456 active internships, and 89 reports generated. A search bar and filter button are available. A table lists three users: John Doe (Student, Active) with an Edit button.
- Generate Reports:** Creates custom system reports. It includes sections for Usage Statistics (System activity report), User Analytics (User behavior report), and a Custom Report button.
- Configuration:** System settings and preferences. It includes sections for Notification Settings (Edit), User Permissions (Edit), and System Alerts (Edit).
- Recent Complaints:** Handles and monitors user complaints. It shows two Internship Issue Reports filed by Sarah Parker 2 hours ago, both pending. Each report has Review and Assign buttons.

1. System Overview: Keep an eye on the status and activities of the platform.
2. User management: oversee users, including administrators, businesses, and students.
3. Handling Complaints: Examine and address complaints that have been submitted.
4. Report Creation: Produce reports on internship progress and platform usage.
5. Configuration Settings: Modify settings for the entire system.

Hardware Interfaces

1. Compatible with both desktop and mobile web browsers today.
2. 1280x720 is the minimum screen resolution.
3. Supports mobile devices with touch interfaces.
4. Features for uploading and downloading resumes, job descriptions, and feedback files.

Software Interfaces

1. User information, internships, and system logs are stored in the database management system.
2. Email server: Provides emails for confirmation and notifications.
3. File Storage System: Manages uploaded documents, including resumes.
4. Role-based access and user login are managed by the authentication system.
5. Analytics Engine: Produces use reports and offers insights.

Communication Interfaces

1. Secure communication is ensured using the HTTPS protocol.
2. RESTful API: Enables front-end and back-end communication.
3. Real-time notifications are supported by WebSocket connections.
4. Email SMTP: Manages email correspondence for notifications and alerts.

3.2 Functional Requirements

3.2.1 Requirements List

F1. User Management

ID	Description
F1.1	The system shall allow students, companies, and university administrators to register with verified email addresses
F1.2	The system shall provide secure authentication with optional two-factor verification
F1.3	The system shall allow users to update their profile information including contact details and preferences
F1.4	The system shall enforce role-based access control for students, companies, and administrators
F1.5	The system shall support password reset functionality with email verification

ID	Description
F1.6	The system shall maintain audit logs of all user authentication activities
F1.7	The system shall allow users to manage notification preferences
F1.8	The system shall enforce strong password policies

F2. CV Management

ID	Description
F2.1	The system shall provide customizable CV templates for students
F2.2	The system shall allow students to create and store multiple versions of their CVs
F2.3	The system shall enable students to update their CVs at any time
F2.4	The system shall allow students to control CV visibility to specific companies
F2.5	The system shall provide a skill management interface for students
F2.6	The system shall validate skill entries against a standardized skill database
F2.7	The system shall support document upload for certificates and portfolios
F2.8	The system shall track CV view statistics for students

F3. Internship Management

ID	Description
F3.1	The system shall allow companies to create detailed internship postings

ID	Description
F3.2	The system shall provide an application tracking system for companies
F3.3	The system shall automatically notify students of application status changes
F3.4	The system shall provide advanced search and filtering capabilities
F3.5	The system shall allow companies to set application deadlines
F3.6	The system shall enable bulk application processing for companies
F3.7	The system shall support multiple rounds of application review
F3.8	The system shall maintain a history of all internship postings

F4. Interview Management

ID	Description
F4.1	The system shall provide a calendar interface for interview scheduling
F4.2	The system shall track interview status and progress
F4.3	The system shall collect structured feedback from both parties
F4.4	The system shall send automated interview reminders
F4.5	The system shall support virtual interview link generation
F4.6	The system shall allow rescheduling with mutual agreement
F4.7	The system shall maintain interview history

ID	Description
F4.8	The system shall support multiple interview rounds

F5. Recommendation System

ID	Description
F5.1	The system shall match students with internships based on skills alignment
F5.2	The system shall suggest qualified candidates to companies
F5.3	The system shall learn from user interactions to improve recommendations
F5.4	The system shall generate personalized internship suggestions
F5.5	The system shall consider location preferences in matching
F5.6	The system shall factor in past application success patterns
F5.7	The system shall update recommendations in real-time
F5.8	The system shall explain recommendation reasoning to users

F6. Complaint Handling

ID	Description
F6.1	The system shall provide a structured complaint submission interface
F6.2	The system shall route complaints to appropriate university administrators
F6.3	The system shall track resolution progress

ID	Description
F6.4	The system shall support appeals process
F6.5	The system shall maintain complete complaint history
F6.6	The system shall enable communication between parties
F6.7	The system shall generate complaint resolution reports
F6.8	The system shall enforce resolution timeframes

3.2.2 Priority and Criticality Matrix

Requirement Category	Priority	Implementation Phase	Criticality
User Management	High	Phase 1	Critical
CV Management	High	Phase 1	Critical
Internship Management	High	Phase 1	Critical
Interview Management	Medium	Phase 2	Important
Recommendation System	Medium	Phase 2	Important
Complaint Handling	Low	Phase 3	Standard

3.2.3 Dependencies

F1 Dependencies

- F1.1 must be completed before any other functionality
- F1.2 is required for all secure operations
- F1.4 impacts all other functional areas

F2 Dependencies

- Requires F1 completion
- F2.1 must be completed before F2.2
- F2.5 is required for F5.1

F3 Dependencies

- Requires F1 completion
- F3.1 must be completed before F3.2
- F3.4 is required for F5.4

F4 Dependencies

- Requires F1 and F3 completion
- F4.1 must be completed before F4.2
- F4.3 feeds into F5.3

F5 Dependencies

- Requires F2 and F3 completion
- F5.1 must be completed before F5.4
- Requires continuous data from F4.3

F6 Dependencies

- Requires all other systems to be operational
- F6.1 must be completed before F6.2
- F6.4 requires F6.3 completion

3.2.4 Use Case Diagrams

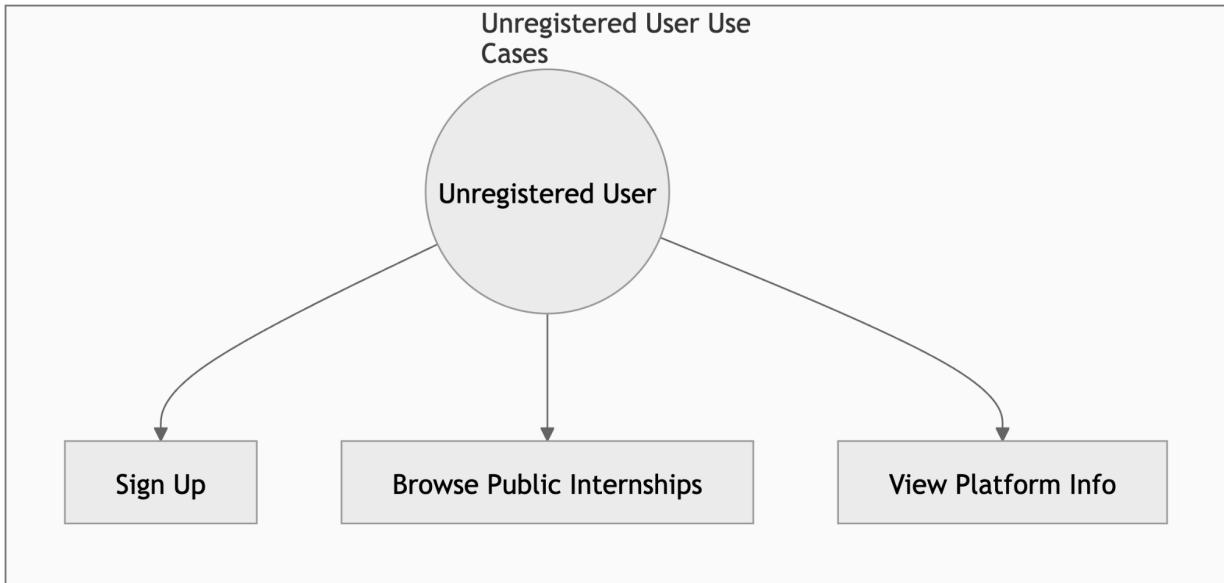


Fig (): Use Case Diagram for Unregistered User

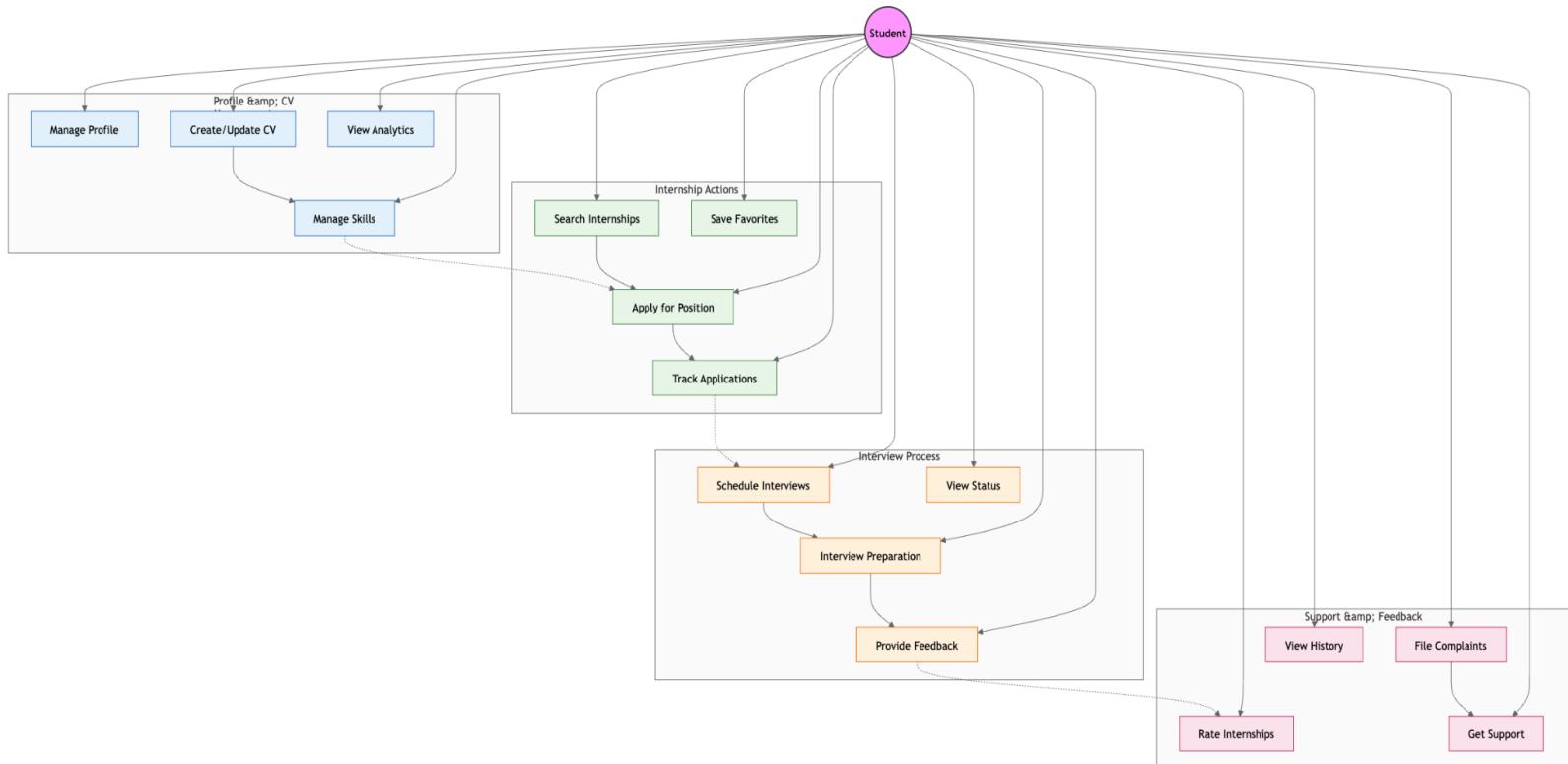


Fig (): Use Case Diagram for Student

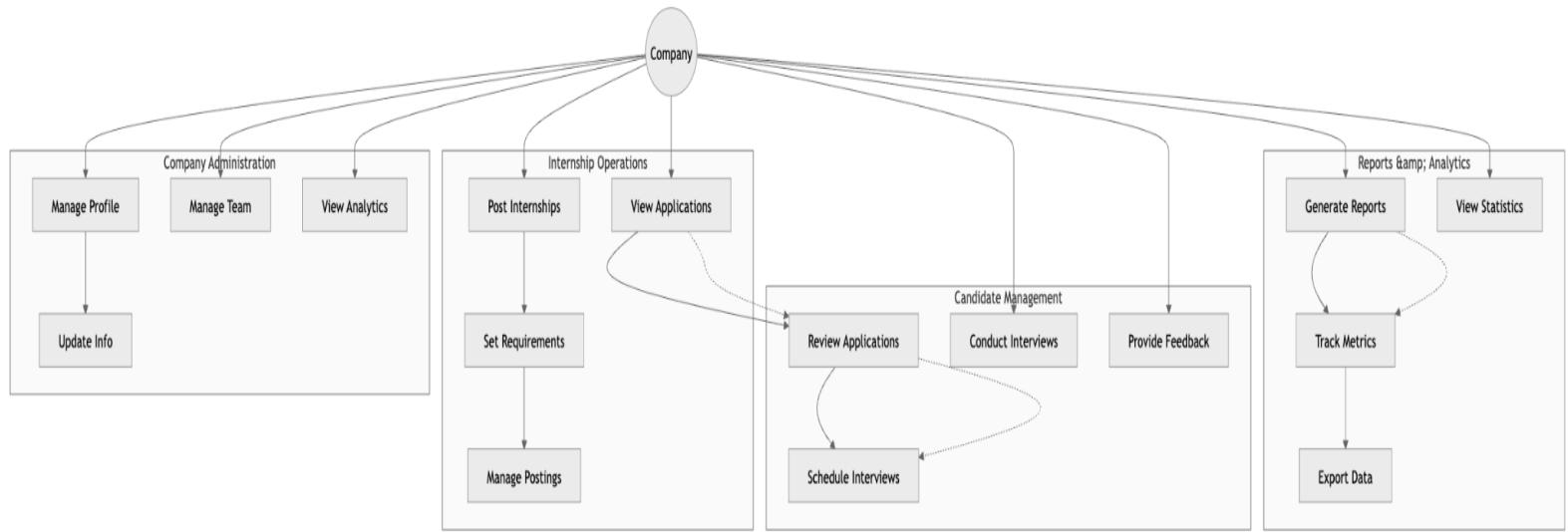


Fig: Use Case Diagram for Company

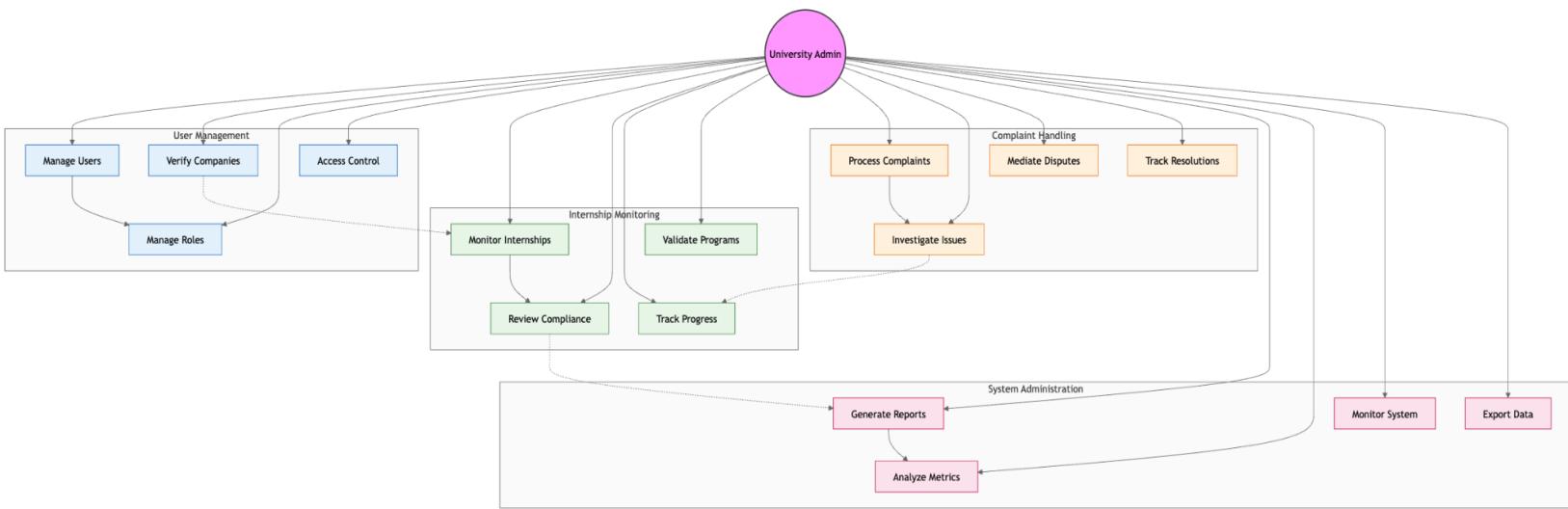


Fig: Use Case Diagram for University Admin

3.2.5 Use Cases

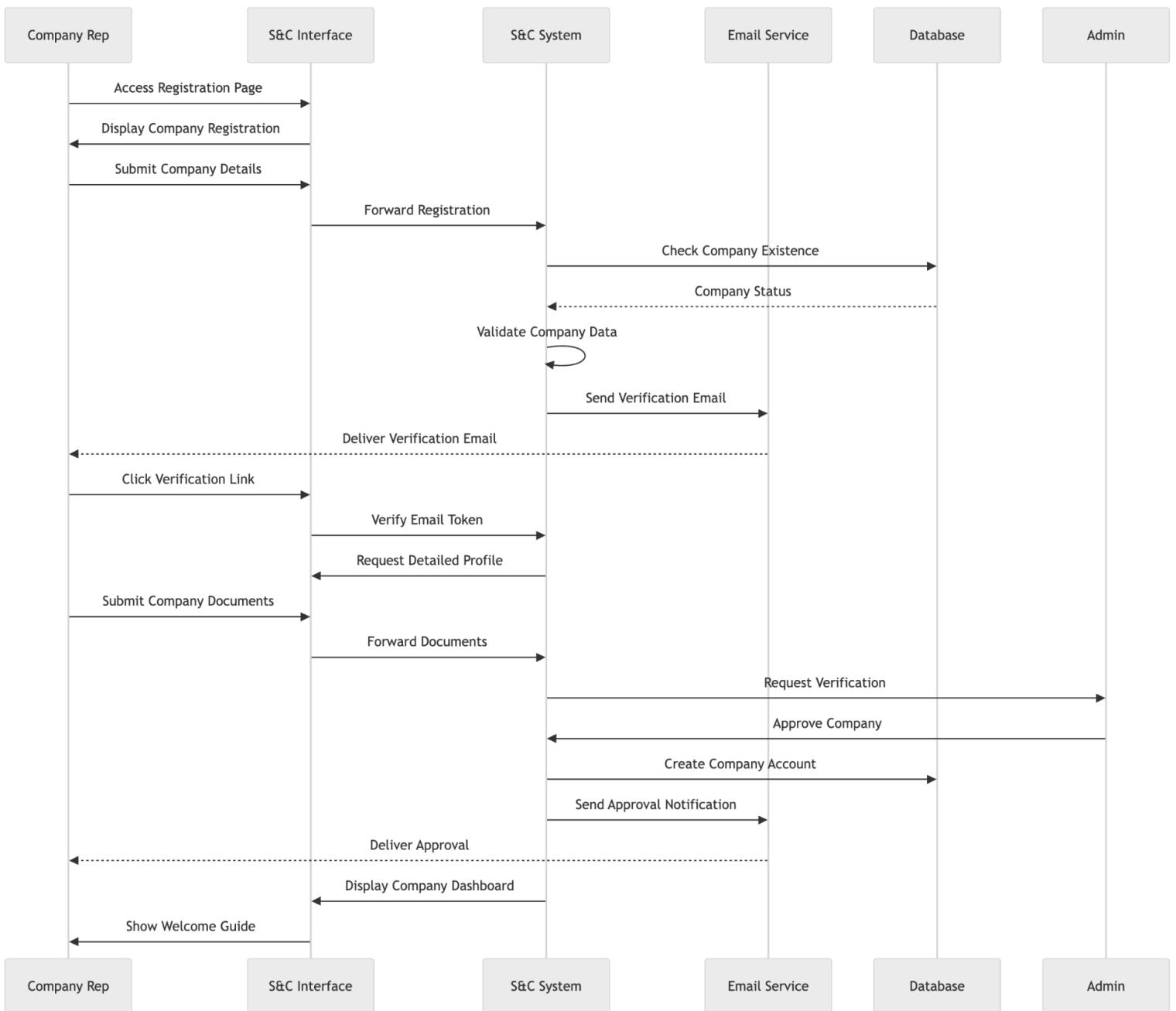
The primary identified use cases are described and illustrated in this section.

Each of them has a table with entry conditions, event row, exit conditions, and exceptions, as well as a sequence diagram that displays the messages sent back and forth between the called functions and the entities.

UC1: Student Registration

Actor	Company Representative, Email Provider
Entry Conditions	<ul style="list-style-type: none">- Company not registered in S&C- Representative has company email domain- Company meets platform requirements
Event Flow:	<ol style="list-style-type: none">1. S&C shows the login form with registration option2. Representative clicks on "Create Account"3. S&C displays registration form4. Representative selects "Register as Company"5. Representative enters initial information:<ul style="list-style-type: none">- Company Name- Company Website- Industry Type- Company Size- Company Email Domain- Representative Name- Representative Position- Password6. S&C validates company information7. S&C performs preliminary company verification:<ul style="list-style-type: none">- Website domain matches email domain- Company exists in business registry (if applicable)8. S&C sends verification email9. Representative clicks verification link

	<p>10. S&C prompts for detailed company information:</p> <ul style="list-style-type: none"> - Company Description - Office Locations - Logo Upload - Required Documents <p>11. S&C sends information for admin verification</p> <p>12. Admin reviews and approves company</p> <p>13. S&C activates company account</p> <p>14. S&C guides through internship posting process</p>
Exit Conditions	<ul style="list-style-type: none"> - Company account is created and verified - Company profile is complete - Company can post internships
Exceptions	<ol style="list-style-type: none"> 1. Company already registered <ul style="list-style-type: none"> - Show existing company message - Provide contact for account recovery 2. Invalid company email domain <ul style="list-style-type: none"> - Request valid company email - Provide business verification alternatives 3. Company verification failed <ul style="list-style-type: none"> - Request additional verification documents - Provide support contact 4. Admin rejection <ul style="list-style-type: none"> - Notify reason for rejection - Provide appeal process information 5. Incomplete required documents <ul style="list-style-type: none"> - List missing documents - Save partial progress - Allow later completion



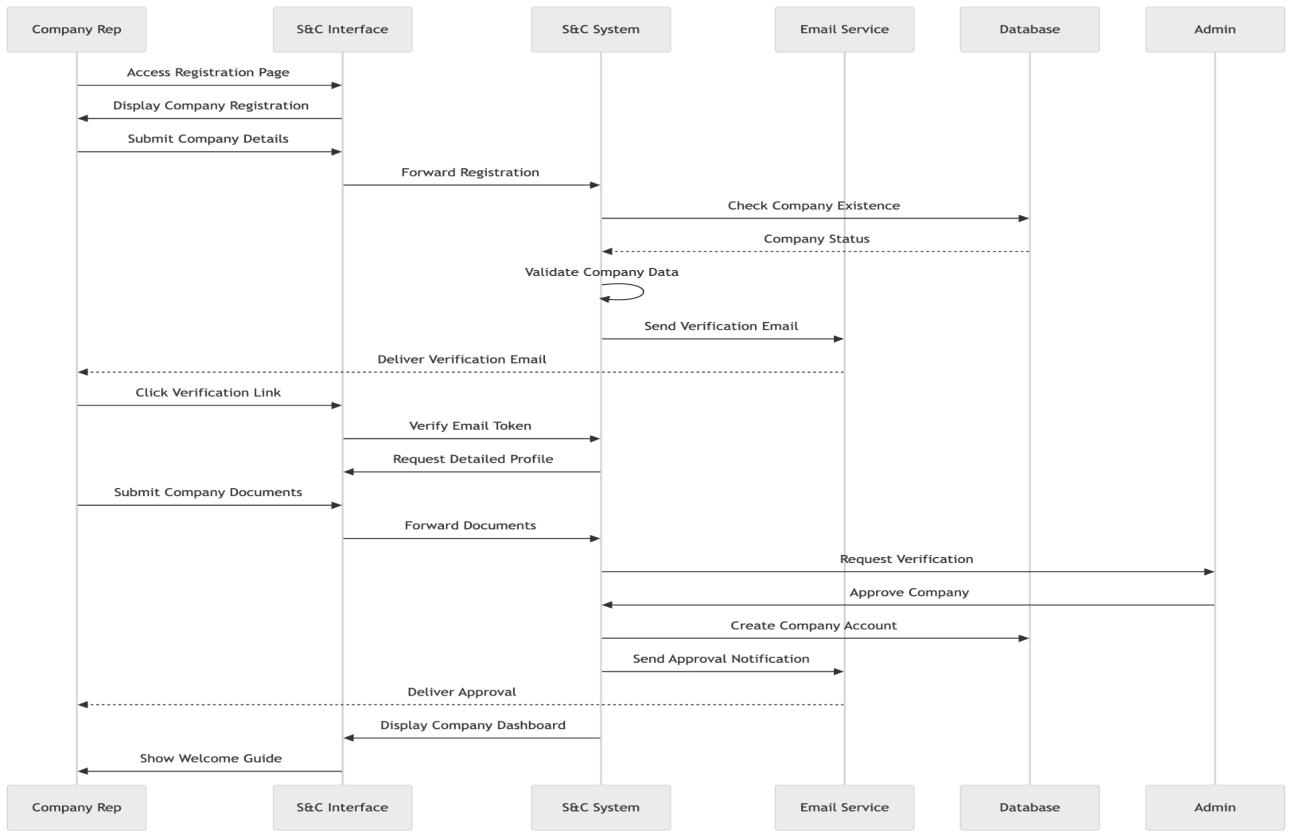
Fig() : Sequence Diagram for Student Registration

UC 2 - Use Case for Company Registration

Actor	Company Representative, Email Provider
Entry Conditions:	<ul style="list-style-type: none">- Company not registered in S&C- Representative has company email domain- Company meets platform requirements
Event Flow:	<ol style="list-style-type: none">1. S&C shows the login form with registration option2. Representative clicks on "Create Account"3. S&C displays registration form4. Representative selects "Register as Company"5. Representative enters initial information:<ol style="list-style-type: none">a. Company Nameb. Company Websitec. Industry Typed. Company Sizee. Company Email Domainf. Representative Nameg. Representative Positionh. Password

	<ol style="list-style-type: none"> 6. S&C validates company information 7. S&C performs preliminary company verification: <ol style="list-style-type: none"> a. Website domain matches email domain b. Company exists in business registry (if applicable) 8. S&C sends verification email 9. Representative clicks verification link 10. S&C prompts for detailed company information: <ol style="list-style-type: none"> a. Company Description b. Office Locations c. Logo Upload d. Required Documents 11. S&C sends information for admin verification 12. Admin reviews and approves company 13. S&C activates company account 14. S&C guides through internship posting process
Exit Conditions	<ul style="list-style-type: none"> - Company account is created and verified - Company profile is complete - Company can post internships

Exceptions	<ol style="list-style-type: none">1. Company already registered<ul style="list-style-type: none">- Show existing company message- Provide contact for account recovery2. Invalid company email domain<ul style="list-style-type: none">- Request valid company email- Provide business verification alternatives3. Company verification failed<ul style="list-style-type: none">- Request additional verification documents- Provide support contact4. Admin rejection<ul style="list-style-type: none">- Notify reason for rejection- Provide appeal process information5. Incomplete required documents<ul style="list-style-type: none">- List missing documents- Save partial progress- Allow later completion
------------	--



Fig() : Sequence Diagram for Company Registration

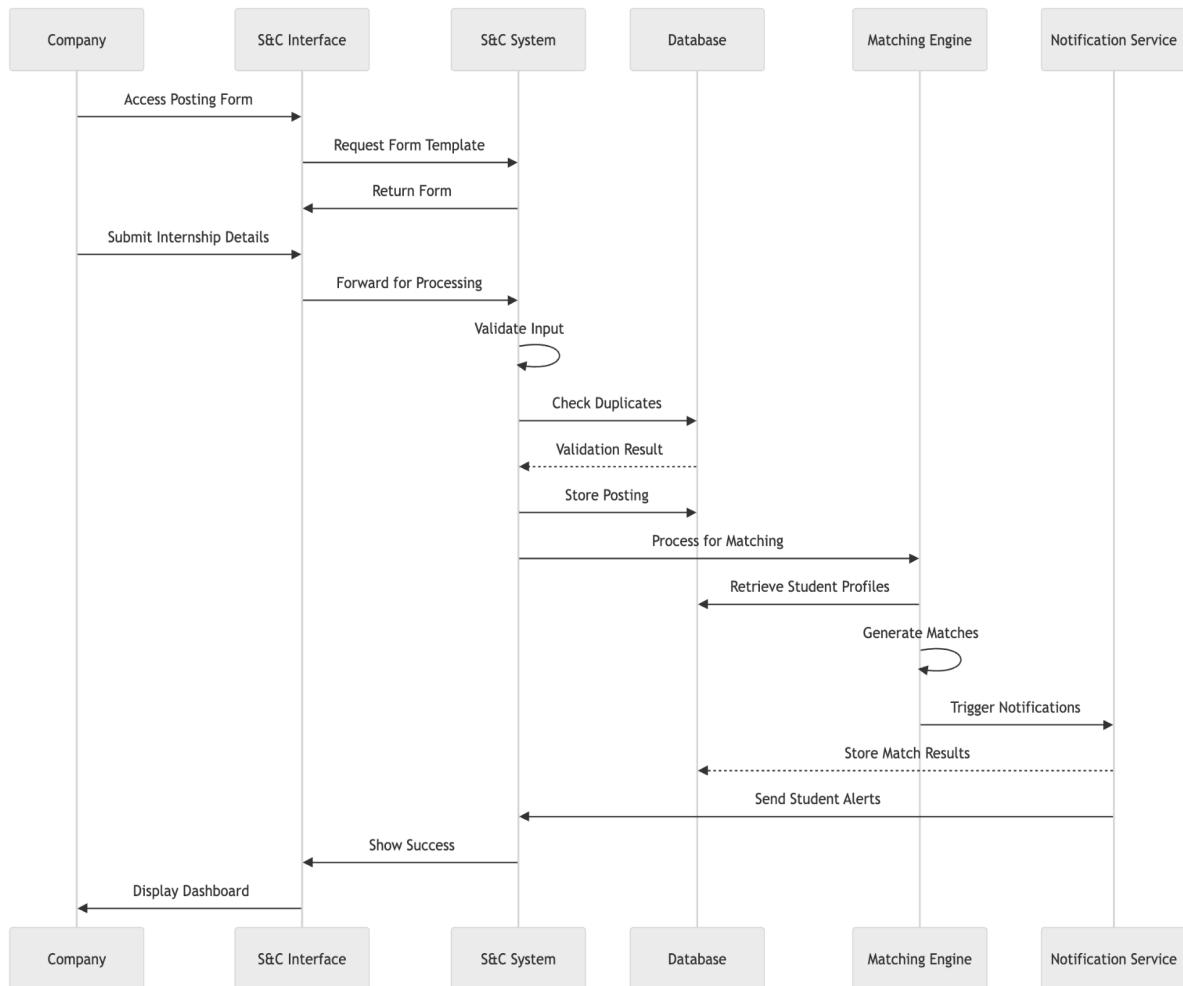
UC 3 - Use Case for Internship Posting

Actor	Company Representative
Entry Conditions	<ul style="list-style-type: none"> - Company is verified and logged in - Company profile is complete - Company has posting privileges

Event Flow	<ol style="list-style-type: none"> 1. Company accesses internship management dashboard 2. Clicks "Post New Internship" button 3. S&C displays internship creation form 4. Company enters internship details: <ul style="list-style-type: none"> - Position Title - Department - Duration - Start Date - Required Skills - Preferred Skills - Educational Requirements - Responsibilities - Compensation Details - Application Deadline - Number of Positions - Location/Remote Status 5. Company sets additional preferences: <ul style="list-style-type: none"> - Interview Process Steps - Required Documents - Assessment Criteria 6. S&C validates all input fields 7. Company previews posting 8. Company submits for review 9. S&C performs automated checks: <ul style="list-style-type: none"> - Content appropriateness - Completeness - Compliance with platform rules 10. S&C processes posting: <ul style="list-style-type: none"> - Indexes for search - Matches with student profiles - Generates recommendations 11. S&C activates posting 12. Notifies matching students
Exit Conditions	<ul style="list-style-type: none"> - Internship is posted and visible - Matching students are notified - Position appears in search results

Exceptions:

1. Incomplete Required Fields
 - Highlight missing fields
 - Save as draft
2. Invalid Date Combinations
 - Show date validation error
 - Suggest valid date ranges
3. Non-Compliant Content
 - Flag specific issues
 - Provide compliance guidelines
4. Duplicate Posting
 - Show existing posting
 - Offer update option
5. Posting Limit Reached
 - Show upgrade options
 - Manage existing posts

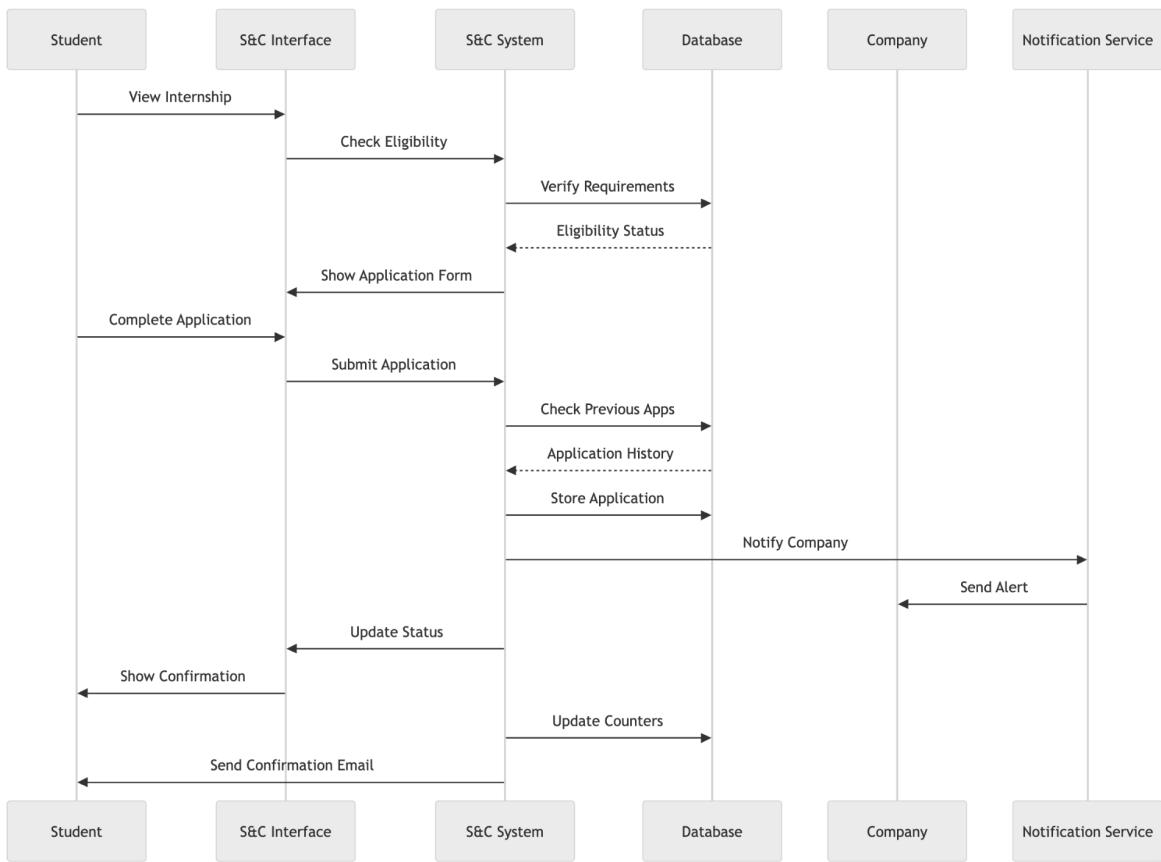


Fig(): Sequence Diagram for Internship Posting

UC 4 - Use Case for Student Application Process

Actor	Student
Entry Conditions	<ul style="list-style-type: none"> - Student is logged in - Profile and CV are complete - Meets basic internship requirements
Event Flow	<ol style="list-style-type: none"> 1. Student finds internship through: <ul style="list-style-type: none"> - Direct search - Recommendations - Email notification 2. Student reviews internship details:

	<ul style="list-style-type: none"> - Company information - Position requirements - Terms and conditions <p>3. Student initiates application:</p> <ul style="list-style-type: none"> - Selects CV version - Customizes cover letter - Answers screening questions - Provides additional documents <p>4. S&C performs preliminary checks:</p> <ul style="list-style-type: none"> - Eligibility verification - Previous applications - Document completeness <p>5. Student reviews application package</p> <p>6. Student submits application</p> <p>7. S&C processes application:</p> <ul style="list-style-type: none"> - Updates application counter - Indexes for company search - Generates application ID <p>8. Company receives notification</p> <p>9. S&C updates student dashboard</p> <p>10. Application tracking begins</p>
Exit Conditions	<ul style="list-style-type: none"> - Application is submitted - Company is notified - Student can track status
Exceptions:	<ol style="list-style-type: none"> 1. Incomplete Profile <ul style="list-style-type: none"> - Redirect to profile completion - Save application draft 2. Missing Required Documents <ul style="list-style-type: none"> - List required documents - Provide upload options 3. Previous Application Exists <ul style="list-style-type: none"> - Show existing application - Offer update option 4. Deadline Passed <ul style="list-style-type: none"> - Show expired notice - Suggest similar positions 5. Requirements Not Met <ul style="list-style-type: none"> - Show specific gaps - Suggest skill development



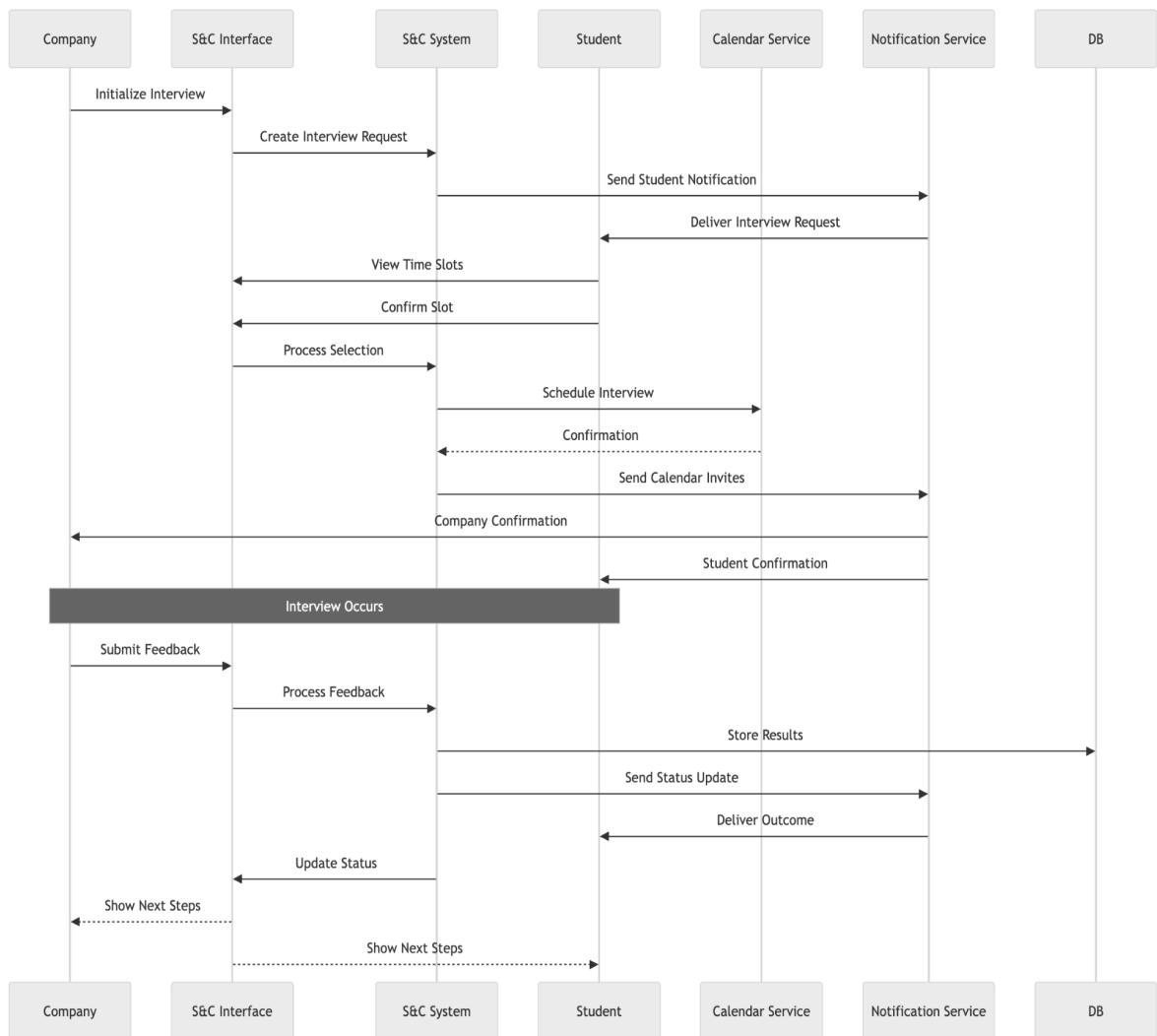
Fig(): Sequence Diagram for Student Application Process

UC 5 - Use Case for Interview Management

Actor	Company Representative, Student
Entry Conditions	<ul style="list-style-type: none"> - Application is shortlisted - Both parties are active users - Interview stage is initiated
Event Flow	<ol style="list-style-type: none"> 1. Company reviews application 2. Company initiates interview process: <ul style="list-style-type: none"> - Selects interview type - Proposes time slots

	<ul style="list-style-type: none"> - Sets interview format <p>3. S&C sends interview request to student</p> <p>4. Student receives notification:</p> <ul style="list-style-type: none"> - Views proposed slots - Checks interview details - Reviews preparations <p>5. Student responds:</p> <ul style="list-style-type: none"> - Accepts time slot or - Proposes alternatives <p>6. S&C coordinates scheduling:</p> <ul style="list-style-type: none"> - Confirms final time - Sends calendar invites - Provides meeting links <p>7. Both parties prepare:</p> <ul style="list-style-type: none"> - Access interview materials - Review guidelines - Check technical setup <p>8. Interview conducted</p> <p>9. Company provides feedback:</p> <ul style="list-style-type: none"> - Evaluation form - Notes and ratings - Decision input <p>10. S&C processes results:</p> <ul style="list-style-type: none"> - Updates application status - Notifies student - Records outcomes
Exit Conditions:	<ul style="list-style-type: none"> - Interview is completed - Feedback is recorded - Next steps are initiated
Exceptions:	<ol style="list-style-type: none"> 1. Schedule Conflicts <ul style="list-style-type: none"> - Rescheduling process - Alternative slot suggestions 2. Technical Issues <ul style="list-style-type: none"> - Backup contact methods - Rescheduling options 3. No-Show Scenarios <ul style="list-style-type: none"> - Record incident - Reschedule policy 4. Cancellation Requests <ul style="list-style-type: none"> - Process cancellation

- Update status
- 5. Feedback Deadline Missed**
- Send reminders
 - Escalation process

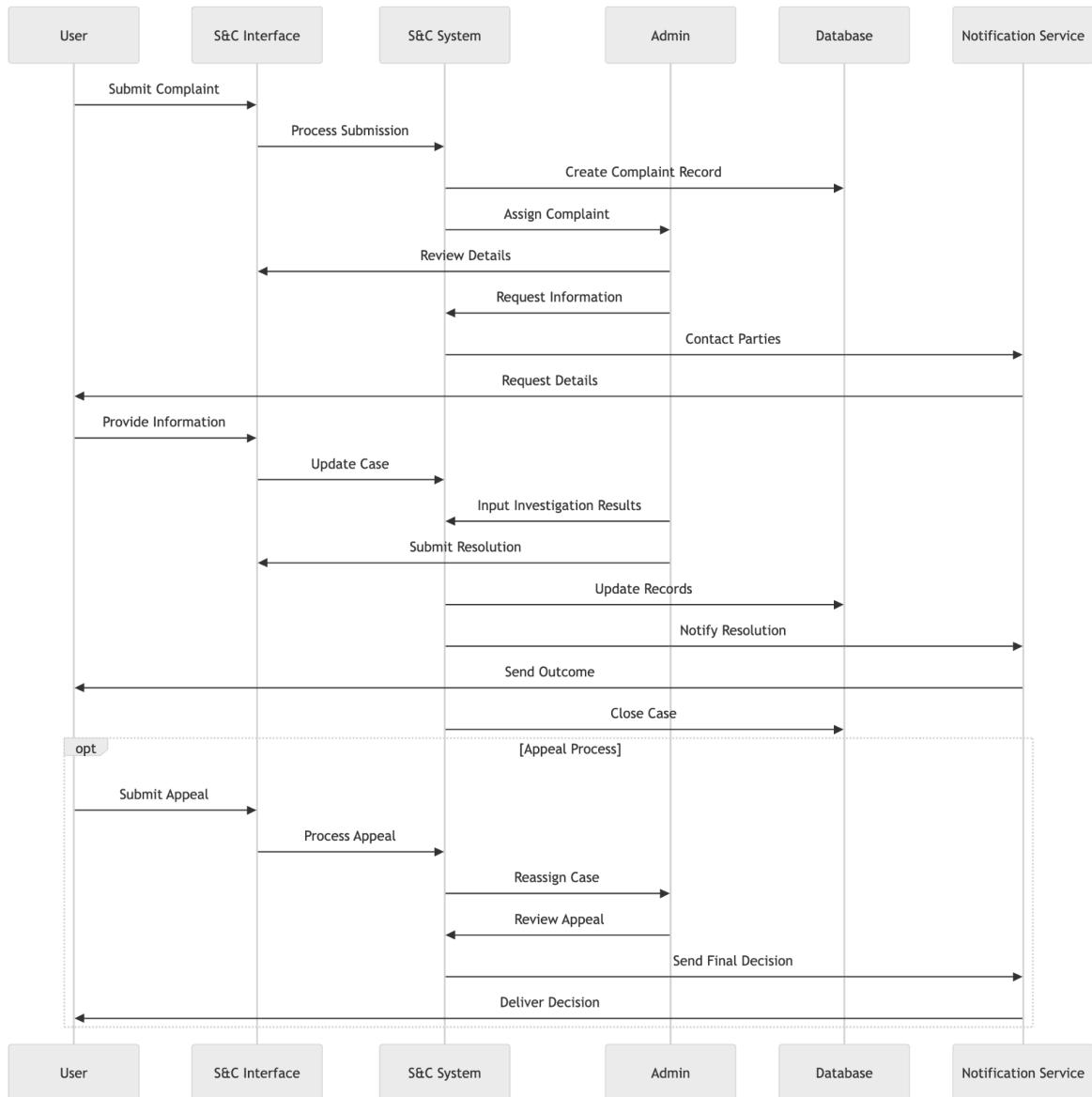


Fig(): Sequence diagram for Interview Management

UC 6 - Use Case for Complaint Handling:

Actor	Student/Company, University Administrator
Entry Conditions	<ul style="list-style-type: none"> - User is registered and active - Incident is within platform scope - Related internship is active/recent
Event Flow	<ol style="list-style-type: none"> 1. User initiates complaint: <ul style="list-style-type: none"> - Selects complaint type - Identifies involved parties - Describes incident - Provides evidence 2. S&C processes submission: <ul style="list-style-type: none"> - Assigns complaint ID - Categorizes severity - Routes to appropriate admin 3. Administrator receives case: <ul style="list-style-type: none"> - Reviews details - Assesses priority - Initiates investigation 4. S&C facilitates investigation: <ul style="list-style-type: none"> - Gathers additional information - Contacts involved parties - Documents communications 5. Administrator actions: <ul style="list-style-type: none"> - Reviews all evidence - Consults policies - Determines resolution 6. Resolution implementation: <ul style="list-style-type: none"> - Notifies all parties - Records decisions - Implements actions 7. Follow-up process: <ul style="list-style-type: none"> - Monitors compliance - Collects feedback - Updates records

Exit Conditions:	<ul style="list-style-type: none"> - Complaint is resolved - Actions are implemented - Resolution is documented
Exceptions:	<ol style="list-style-type: none"> 1. Insufficient Information <ul style="list-style-type: none"> - Request additional details - Hold complaint status 2. Multiple Parties Involved <ul style="list-style-type: none"> - Expand investigation scope - Coordinate responses 3. Policy Violations Found <ul style="list-style-type: none"> - Escalate to higher authority - Implement immediate actions 4. Appeal Submitted <ul style="list-style-type: none"> - Review appeal grounds - Reassign to different admin 5. Resolution Deadline Missed <ul style="list-style-type: none"> - Escalate to supervisor - Update timeline

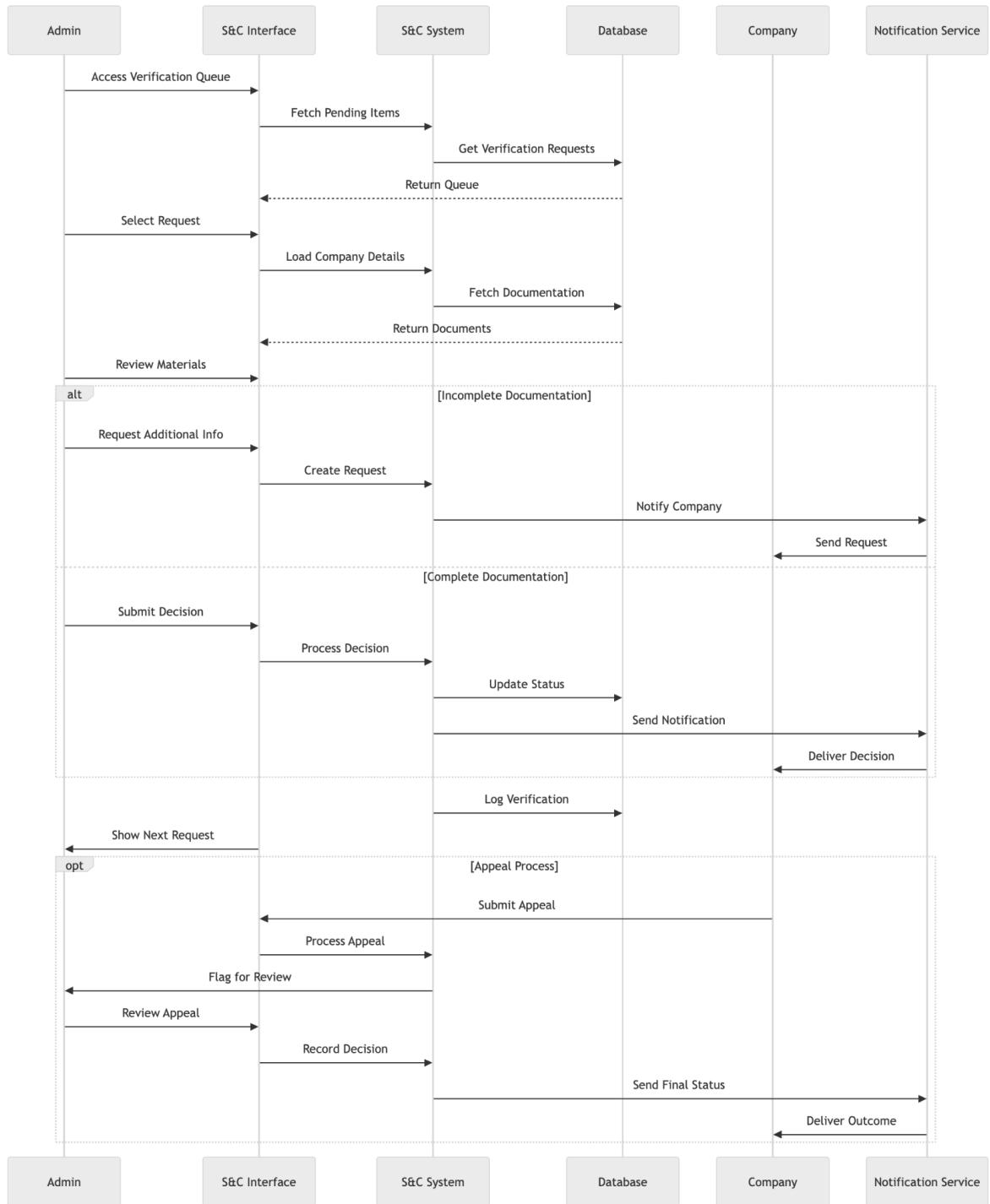


Fig(): Sequence Diagram for Complaint Handling

UC 7 - Use Case for Admin Privileges:

Actor	University Administrator
Entry Conditions	<ul style="list-style-type: none"> - Admin is logged in with verification privileges - Items pending verification exist - Access to verification tools available
Event Flow	<ol style="list-style-type: none"> 1. Admin accesses verification dashboard: <ul style="list-style-type: none"> - Views pending items - Checks verification queue - Reviews priority items 2. For each verification request: <ul style="list-style-type: none"> - Reviews company details - Checks documentation - Validates credentials - Assesses compliance 3. Document verification: <ul style="list-style-type: none"> - Company registration - Business licenses - Insurance certificates - Tax documentation 4. Compliance check: <ul style="list-style-type: none"> - University policies - Legal requirements - Industry standards - Safety regulations 5. Decision process: <ul style="list-style-type: none"> - Approves application - Requests modifications - Rejects with reason 6. Post-approval actions: <ul style="list-style-type: none"> - Sets verification status - Assigns trust score - Enables features - Sets review date 7. Communication: <ul style="list-style-type: none"> - Notifies company - Updates records - Documents decision

Exit Conditions:	<ul style="list-style-type: none"> - Verification decision made - Company notified - Records updated
Exceptions:	<ol style="list-style-type: none"> 1. Missing Documentation <ul style="list-style-type: none"> - Request specific documents - Set pending status 2. Compliance Issues <ul style="list-style-type: none"> - Detail requirements - Provide guidance 3. Verification Timeout <ul style="list-style-type: none"> - Extend deadline - Notify parties 4. Suspicious Activity <ul style="list-style-type: none"> - Flag for investigation - Suspend processing 5. Appeal Request <ul style="list-style-type: none"> - Review appeal - Escalate if needed



Sequence Diagram for Admin Privileges

3.3 Performance Requirements

Response Time

- **Page Load:** < 5 seconds for typical pages during testing.
- **Search Results:** < 2 seconds for small-scale datasets.
- **File Upload:** < 10 seconds for files up to 10MB.
- **Real-time Updates:** < 1 second for key interactive features.

System Capacity

- **Concurrent Users:** Support for up to 50 simultaneous users.
- **Database Transactions:** Designed to handle 10 transactions per second under typical usage.
- **File Storage:** Capacity for up to 10GB, suitable for project scope and testing.
- **Backup Frequency:** Weekly or manual backups for data protection during development.

Availability

- **Uptime:** Target of 95%, considering potential downtime for development and testing.
- **Scheduled Maintenance:** As needed during the project lifecycle.
- **Backup Recovery:** Recovery within 12 hours for small-scale data.
- **Error Rate:** < 1% for prototype-level functionality.

3.4 Design Constraints

Standards Compliance

- 4) GDPR compliance
- 5) WCAG 2.1 accessibility
- 6) ISO/IEC 27001 security
- 7) Browser standards

Development Constraints

- 8) Web-based architecture
- 9) Responsive design
- 10) Modular components
- 11) API-first approach

3.5 Software System Attributes

Reliability

1. Error handling to ensure data integrity.

2. Input validation for all user entries.
3. Transaction integrity for financial processes.
4. System recovery after failures.

Availability

1. Redundant Systems: Ensure continuous, round-the-clock functioning of the platform.
2. Failover Mechanism: Enables the system to handle server disruptions by automatically switching to backup systems.

Security

1. Robust authentication systems.
2. Authorization for sensitive operations based on roles.
3. Data encryption from beginning to end.

Maintainability

1. Modular design for easier updates.
2. Comprehensive documentation for developers and users.
3. Version control for all software components.
4. Automated testing frameworks.

Portability

1. Cross-browser compatibility.
2. Mobile responsiveness.
3. Platform independence to run on various operating systems.
4. Easy deployment for scaling and updates.

4. Formal Analysis Using Alloy

4.1 Model Description

```
// ----- Core Signatures -----
```

```
abstract sig User {  
    id: one ID,  
    email: one ID,  
    name: one ID,  
    phoneNumber: one ID  
}
```

```
// A student user who has a CV and can submit internship applications.
```

```

sig Student extends User {
    cv: one CV,
    applications: set Application
} {
    // Every student must have a CV.
    some cv
}

// A company user that offers internships.
sig Company extends User {
    companyName: one ID,
    industry: one ID,
    internships: set Internship // Internships provided by the company.
}

// A university user managing students and handling complaints.
sig University extends User {
    universityName: one ID,
    students: set Student, // The students enrolled in the university.
    complaints: set Complaint // Complaints registered with the university.
}

// A CV containing a summary, skills, education, and experience details.
sig CV {
    skills: set Skill,
    education: set Education,
    experience: set Experience,
    summary: one ID
}

// An internship with specific details such as title, description, requirements, and dates.
sig Internship {
    title: one ID,
    description: one ID,
    requirements: set Skill, // Skills required for the internship.
    startDate: one Date,
    endDate: one Date,
    status: one Status
}

// An application linking a student to an internship.
sig Application {
    student: one Student,
    internship: one Internship,
}

```

```

applicationDate: one Date,
status: one Status
}

// An interview scheduled for a specific application.
sig Interview {
    application: one Application,
    scheduledTime: one DateTime, // Time slot for the interview.
    status: one Status,
    feedback: lone ID
}

// A complaint filed by a user with a description and status.
sig Complaint {
    id: one ID,
    description: one ID, // Complaint details.
    status: one Status,
    filingDate: one Date,
    complainant: one User // The user who raised the complaint.
}

// Represents textual identifiers for various elements.
sig ID {}

// ----- Enumerations -----

enum Skill { JAVA, PYTHON, CPP, JAVASCRIPT, DATABASE }
enum Education { BACHELORS, MASTERS, PHD }
enum Experience { JUNIOR, MID, SENIOR }
enum Status { OPEN, CLOSED, PENDING, APPROVED, REJECTED }
enum DateTime { MORNING, AFTERNOON, EVENING }

// Represents months as a hierarchy rather than an enumeration.
abstract sig Date {}
one sig JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC extends Date {}

// ----- Utility Function + Predicate -----

// Assigns a numerical index to each month for easy comparison.
fun dateIndex[d: Date]: Int {
    d = JAN => 1
    else d = FEB => 2
    else d = MAR => 3
    else d = APR => 4
}

```

```

        else d = MAY => 5
        else d = JUN => 6
        else d = JUL => 7
        else d = AUG => 8
        else d = SEP => 9
        else d = OCT => 10
        else d = NOV => 11
        else 12
    }

    // Validates that one date comes before another.
    pred dateOrder[d1, d2: Date] {
        dateIndex[d1] < dateIndex[d2]
    }

    // ----- Facts & Constraints -----

    // Enforces unique IDs for all users.
    fact UserIDAreUnique {
        all disj u1, u2: User | u1.id != u2.id
    }

    // Ensures the system always contains at least one student, company, and university.
    fact AtLeastOneOfEach {
        some Student
        some Company
        some University
    }

    // Every student must be associated with exactly one university.
    fact StudentUniversityRelationship {
        all s: Student | one u: University | s in u.students
    }

    // Ensures internships are owned by exactly one company.
    fact InternshipOwnership {
        all i: Internship | one c: Company | i in c.internships
    }

    // Ensures internships have a valid time frame, with the start date before the end date.
    fact InternshipDates {
        all i: Internship | dateOrder[i.startDate, i.endDate]
    }

```

```

// A student can only apply for internships where their CV covers all required skills.
fact ValidApplications {
    all a: Application |
        a.internship.requirements in a.student.cv.skills
}

// Guarantees that open internships have at least one applicant.
fact OpenInternshipsMustHaveApplicants {
    all i: Internship |
        i.status = OPEN implies some a: Application | a.internship = i
}

// Prevents duplicate applications for the same internship by the same student.
fact UniqueApplications {
    all disj a1, a2: Application |
        (a1.student != a2.student) or (a1.internship != a2.internship)
}

// If an application is approved, the corresponding internship cannot remain open.
fact ApprovedApplicationClosesInternship {
    all a: Application |
        a.status = APPROVED implies a.internship.status != OPEN
}

// Ensures interviews are only scheduled if the student's skills meet the internship requirements.
fact ValidInterview {
    all i: Interview |
        i.application.student.cv.skills in i.application.internship.requirements
}

// Complaints (if not rejected) must have a valid complainant.
fact ComplaintResolution {
    all c: Complaint |
        c.status != REJECTED implies some c.complainant
}

// Ensures all complaints have unique IDs and exactly one complainant.
fact ValidComplaints {
    all c: Complaint | one u: User | c.complainant = u
    all disj c1, c2: Complaint | c1.id != c2.id
}

// Suggests that students whose skills match the requirements of an internship should apply.
fact RecommendedApplications {

```

```

all s: Student, i: Internship |
  (i.requirements in s.cv.skills) implies
    (some a: Application | a.student = s and a.internship = i)
}

// ----- Simple Run Command -----

// Generates an instance of the model up to a scope of 5.
run {} for 5

// ----- Analysis Code (Assertions + Predicates) -----

// Assertion: Prevents students from applying to the same internship multiple times.
assert UniqueApplicationsAssert {
  all disj a1, a2: Application |
    a1.student = a2.student and a1.internship = a2.internship
    implies a1 = a2
}
check UniqueApplicationsAssert for 5

// Assertion: Validates that open internships have at least one applicant.
assert OpenInternshipsAssert {
  all i: Internship |
    i.status = OPEN implies some a: Application | a.internship = i
}
check OpenInternshipsAssert for 5

// Assertion: Ensures students are only interviewed if they meet the requirements.
assert ValidInterviewAssert {
  all i: Interview |
    i.application.student.cv.skills in i.application.internship.requirements
}
check ValidInterviewAssert for 5

// Assertion: Complaints (if not rejected) must have valid complainants.
assert ComplaintResolutionAssert {
  all c: Complaint |
    c.status != REJECTED implies some c.complainant
}
check ComplaintResolutionAssert for 5

// Assertion: Approved applications mean the internship is not open anymore.
assert ApprovedApplicationClosesInternshipAssert {
  all a: Application |

```

```

        a.status = APPROVED implies a.internship.status != OPEN
    }
    check ApprovedApplicationClosesInternshipAssert for 5

// Assertion: If a student meets all required skills, an application should exist.
assert RecommendedApplicationsAssert {
    all s: Student, i: Internship |
        (i.requirements in s.cv.skills) implies
        (some a: Application | a.student = s and a.internship = i)
}
check RecommendedApplicationsAssert for 5

//-----
// Scenarios to RUN and visualize
//-----

/**
 * Scenario: A student is applying for an internship.
 * The internship is open, and the student has the required skills for it.
 * We also ensure that the application is properly linked to both the student
 * and the internship.
 */
pred StudentAppliesToOpenInternship {
    some s: Student, i: Internship |
        i.status = OPEN
        and i.requirements in s.cv.skills
        and some a: Application | a.student = s and a.internship = i
}
/** Execute this scenario with up to 5 instances */
run StudentAppliesToOpenInternship for 5

/**
 * Scenario: An internship gets approved for a student.
 * Once approved, the internship is no longer open for applications.
 * This helps verify that the status transitions are consistent.
 */
pred ApprovedInternshipScenario {
    some a: Application |
        a.status = APPROVED
        and a.internship.status != OPEN
}
/** Execute this scenario with up to 5 instances */
run ApprovedInternshipScenario for 5

```

```

/**
 * Scenario: A complaint has been submitted and isn't rejected.
 * If the complaint is still under consideration or resolved,
 * we need to ensure there's a valid complainant tied to it.
 */
pred ValidComplaintScenario {
    some c: Complaint |
    c.status != REJECTED
}
/** Execute this scenario with up to 5 instances */
run ValidComplaintScenario for 5

```

5. Effort Spent

Team Member Contributions

Team Member	Task	Hours Spent
Shreesh Kumar Jha	Requirements Analysis, Frontend Mockup, Frontend Setup, Alloy Modelling	56
Member 1	Requirements Analysis, UML Diagram, Frontend Setup, Mockup Implementation	51
Member 2	Requirements Analysis, UML Diagrams, Backend Setup, Backend Implementation	51

6. References

1. IEEE Standard 830-1998, IEEE Recommended Practice for Software Requirements Specifications
2. Software Engineering 2 Course Materials, A.Y. 2024-2025
3. Daniel Jackson, Software Abstractions: Logic, Language, and Analysis
4. Ian Sommerville, Software Engineering (10th Edition)
5. Assignment RDD AY 2024-2025.pdf
6. GDPR Documentation (<https://gdpr.eu/>)
7. Web Content Accessibility Guidelines (WCAG) 2.1
8. ISO/IEC 27001 Information Security Standards=