



POLITECNICO DI MILANO  
Computer Science and Engineering

# Requirements Analysis and Specifications Document

Students&Companies (S&C)

Software Engineering 2 - Project  
A.Y. 2024 - 2025

22 December 2024  
Version 2.1

*Author:*  
Biagio Fabio Schilirò

*Professor:*  
Matteo Camilli

---

# Contents

---

<b>Contents</b>	I
<b>List of Tables</b>	III
<b>List of Figures</b>	IV
<b>1 Introduction</b>	<b>1</b>
1.1 Purpose . . . . .	1
1.1.1 Goals . . . . .	2
1.2 Scope . . . . .	2
1.2.1 World Phenomena . . . . .	3
1.2.2 Shared Phenomena . . . . .	4
1.3 Definitions, Acronyms, Abbreviations . . . . .	4
1.3.1 Definitions . . . . .	5
1.3.2 Acronyms . . . . .	5
1.3.3 Abbreviations . . . . .	6
1.4 Revision History . . . . .	6
1.5 Reference Documents . . . . .	6
1.6 Document Structure . . . . .	7
<b>2 Overall Description</b>	<b>8</b>
2.1 Product Perspective . . . . .	8
2.1.1 Scenarios . . . . .	8
2.1.2 Class Diagram . . . . .	9
2.1.3 State Diagram . . . . .	12
2.2 Product Functions . . . . .	18
2.2.1 Authentication Functions . . . . .	18
2.2.2 Managing Application Functions . . . . .	18
2.2.3 Managing Internship Functions . . . . .	19
2.2.4 FeedBack Functions . . . . .	20
2.3 User Characteristics . . . . .	20
2.4 Assumptions, dependencies and constraints . . . . .	20
2.4.1 Assumptions and Dependencies . . . . .	20
2.4.2 Constraints . . . . .	21
<b>3 Specific Requirements</b>	<b>23</b>
3.1 External Interface Requirements . . . . .	23
3.1.1 Students Interfaces . . . . .	23

---

## *CONTENTS*

---

3.1.2	Companies Interfaces . . . . .	25
3.1.3	Hardware Interfaces . . . . .	27
3.1.4	Software Interfaces . . . . .	27
3.1.5	Communication Interfaces . . . . .	27
3.2	Functional Requirements . . . . .	27
3.2.1	Use Case Diagrams . . . . .	28
3.2.2	Use Case Description . . . . .	31
3.2.3	Sequence Diagrams . . . . .	39
3.2.4	Requirements . . . . .	50
3.2.5	Mapping on Requirements . . . . .	52
3.3	Performance Requirements . . . . .	53
3.4	Design Constraints . . . . .	54
3.4.1	Standard Compliance . . . . .	54
3.4.2	Hardware Limitations . . . . .	54
3.5	Software System Attributes . . . . .	54
3.5.1	Reliability . . . . .	54
3.5.2	Availability . . . . .	54
3.5.3	Security . . . . .	55
3.5.4	Maintainability . . . . .	55
3.5.5	Portability . . . . .	55
<b>4</b>	<b>Formal Analysis Using Alloy</b>	<b>56</b>
4.1	Signatures . . . . .	56
4.2	Facts, Predicate and Assertions . . . . .	60
4.3	Worlds . . . . .	70
<b>5</b>	<b>References</b>	<b>75</b>

---

# List of Tables

---

1.1 Students&Companies' Goals . . . . .	2
1.2 Students&Companies' World Phenomena . . . . .	3
1.3 Students&Companies' Shared Phenomena . . . . .	4
1.4 Definitions . . . . .	5
1.5 Acronyms . . . . .	5
1.6 Abbreviations . . . . .	6
2.1 Students&Companies' Domain Assumptions . . . . .	21
3.1 Register into the platform Use Case . . . . .	31
3.2 Log-in into the platform Use Case . . . . .	32
3.3 Send Application Use Case . . . . .	32
3.4 See Application Use Case . . . . .	33
3.5 Delete Application Use Case . . . . .	33
3.6 Schedule Interview Use Case . . . . .	34
3.7 Fill Out a Questionnaire Use Case . . . . .	34
3.8 Write a comment Use Case . . . . .	35
3.9 Manage a Compliant Use Case . . . . .	35
3.10 Upload C.V. Use Case . . . . .	36
3.11 Insert an Internship Proposal Use Case . . . . .	36
3.12 Manage an Application Request Use Case . . . . .	37
3.13 Manage a Questionnaire Use Case . . . . .	38
3.14 View Questionnaire's Answers Use Case . . . . .	38
3.15 Evaluate the Selection Process Use Case . . . . .	39
3.17 Requirements, Domain Assumptions and Goals Mapping . . . . .	53

---

# List of Figures

---

2.1	UML Class Diagram . . . . .	11
2.2	Authentication State Diagram . . . . .	14
2.3	Internship Application State Diagram . . . . .	15
2.4	Student Evaluation State Diagram . . . . .	16
2.5	Internship Evaluation State Diagram . . . . .	17
3.1	Student and Company Login . . . . .	23
3.2	Student Sign-Up (A) . . . . .	24
3.3	Student Sign-Up (B) . . . . .	24
3.4	Student Profile . . . . .	25
3.5	Company Sign-Up (A) . . . . .	25
3.6	Company Sign-Up (B) . . . . .	26
3.7	Company Profile . . . . .	26
3.8	Registration Use Case Diagram . . . . .	28
3.9	Student Use Case Diagram . . . . .	29
3.10	Company Use Case Diagram . . . . .	30
3.11	Student Registration Sequence Diagram . . . . .	40
3.12	Company Registration Sequence Diagram . . . . .	41
3.13	Student Log-In Sequence Diagram . . . . .	42
3.14	Company Log-In Sequence Diagram . . . . .	43
3.15	Send Application Sequence Diagram . . . . .	44
3.16	Manage Application Sequence Diagram . . . . .	45
3.17	Sustain Selection Process Sequence Diagram . . . . .	46
3.18	Review Internship Sequence Diagram . . . . .	47
3.19	Create Questionnaire Sequence Diagram . . . . .	48
3.20	Manage Internship Application Sequence Diagram . . . . .	49
4.1	Alloy Wold - Sent Applications . . . . .	71
4.2	Alloy Wold - Under Review and WithDrawn Applications . . . . .	72
4.3	Alloy Wold - Rejected and Selection Process Applications . . . . .	73
4.4	Alloy Wold - Rejected and Internship Applications . . . . .	74

## *Chapter 1*

---

# Introduction

---

This document outlines the Requirements Analysis and Specification Document (**R.A.S.D.**) for the Students&Companies System.

Its primary objective is to provide a comprehensive and precise account of the system's functional and non-functional requirements, while exploring real-world use cases, system constraints and user interactions.

The RASD serves as an essential resource for guiding project planning, development processes and formal agreements with stakeholders.

This document is intended for both developers responsible for creating the system-to-be and the end-users (i.e. students and companies), who will interact with the S&C system itself.

### 1.1 Purpose

In today's fast-evolving job market, practical experience has become more important than ever, especially for university students seeking to bridge the gap between academic theory and real-world applications: internships have become a key stepping stone, providing students with valuable opportunities to apply their skills, gain hands-on experience, explore different career paths and build professional networks that can shape their future.

At the same time, companies across various industries are actively seeking fresh talent who offer innovative ideas, up-to-date knowledge, and enthusiasm to the workplace.

However, despite the clear benefits for both the sides, the process of matching students with the right internships remains complex and time-consuming: students frequently struggle to find internships that match their skills, interests, and career goals, while companies face the challenge of sorting through numerous applications to identify candidates who best fit their projects and organizational culture. Indeed, traditional methods for finding internships, such as job boards, personal connections or university career centers, can be inefficient and overwhelming, often leaving both students and companies with unmet expectations.

In order to address these challenges, the **Students&Companies System** (S&C) has been introduced as an innovative platform specifically designed to simplify and optimize the internship matchmaking process: it will ensure that students are paired with internships that align with their academic background and promote professional growth, while helping companies to quickly identify candidates who will bring the most value to their teams.

### **1.1.1 Goals**

The main goals of the system-to-be are expressed in the following points:

<b>Goal</b>	<b>Description</b>
G.1	Allow students to search and filter internship proposals based on the project descriptions and the terms offered by companies
G.2	Allow students to manage (i.e. send, delete and monitor) sent applications to the chosen suitable internship proposals
G.3	Allow students whose application has been accepted by the company to sustain (i.e. fill questionnaires, schedule an interview, access to the Interview Link) and monitor the selection process made by the company
G.4	Allow companies to manage (i.e. review, start a selection process, accept or reject the request) applications sent to them by students according to some information (i.e. experience, skills, attitude) expressed in the students'CVs
G.5	Allow companies to manage the selection process (i.e. send and check questionnaires, schedule and create an Interview Link, approve/decline the internship request) with the students whose application sent to them have been accepted
G.6	Provide a shared space for students and companies involved in an internship experience where they can leave a comment, report a complaint or the resolution of an existing one and view all the feedback made about the internship itself

Table 1.1: Students&Companies' Goals

## **1.2 Scope**

Students&Companies (S&C) is an innovative and dynamic platform designed to revolutionize the way in which **university students and companies connect for internships**.

With S&C, companies can effortlessly access to key student information, including their experience, skills, and attitudes, all listed in their CVs.

On the other hand, students are provided with detailed insights about internships, including the project's domain, the tasks involved, the technologies used, the potential invaluable benefits such as mentorship and training, and the potential compensation (if present). In such a way, S&C will allow **students to proactively searching for internships** that perfectly align with their goals and aspirations.

Once a student applies, and the company accepts their application, the platform becomes the ideal **space for** a smooth and engaging **selection process**: companies can directly interact with applicants, sending questionnaires or scheduling online interviews, - all within the platform.

An important feature to highlight is the integration of an AI-powered tool during the questionnaire creation process: this tool will allow companies to input a prompt or key topic, which the A.I. will analyze to generate tailored, relevant questions that can be directly submitted to the student or reviewed by the company.

Should a student successfully pass the selection phase, S&C continues to support both sides throughout the internship: indeed it will be provided a **shared environment** where students and companies can offer feedback, either comments or complaints, that arise during the on-

going experience. In the case of a complaint, once the issue has been addressed by the other party, the concerned individual can also indicate its resolution.

### **1.2.1 World Phenomena**

The various World Phenomena observed in the system's settings are illustrated below:

<b>World Phenomena</b>	<b>Description</b>
WP.1	The student creates his own CV, highlighting his experiences, skills and attitudes
WP.2	The student chooses internship proposals that best match his expectations
WP.3	The student participate in interviews with companies
WP.4	The student participates into an internship
WP.5	The student encounters issues during the internship
WP.6	The company opens internship opportunities
WP.7	The company evaluates student applications based on some specific criteria
WP.8	The company evaluates student responses to questionnaires
WP.9	The company participates in interviews with students
WP.10	The company assesses the student's performance and address problems during the internship

Table 1.2: Students&Companies' World Phenomena

### **1.2.2 Shared Phenomena**

In this paragraph the various Shared Phenomena observed in the system's settings are reported.

<b>Shared Phenomena</b>	<b>Description</b>
SP.1	The student looks for internships proposals using filters (e.g. skills and compensation) to find opportunities that match his criteria
SP.2	The student uploads his CV in his profile
SP.3	The student sends an application for an internship to a company
SP.4	The student view the details of a sent application
SP.5	The student deletes a sent application
SP.6	The student suggests a new date for an Interview Proposal to the associated company
SP.7	The student accepts an Interview Proposal sent by the associated company
SP.8	The student fills out a questionnaire
SP.9	The student writes a comment about the on-going internship experience
SP.10	The student reports eventual complaints occurred during the internship
SP.11	The student reports that a complaint has been addressed by the associated company
SP.12	The company posts internship opportunities on its profile
SP.13	The company accesses the student's CV
SP.14	The company accept/reject an application of a student
SP.15	The company create the questionnaires to send to the approved students
SP.16	The company generate a list of questions through the A.I. Tool
SP.17	The company sends an Interview Proposal to the student
SP.18	The company reports eventual complaints occurred during the internship
SP.19	The company reports that a complaint has been addressed by the associated student
SP.20	The companies and the students are notified whenever a new complaint is reported
SP.21	The student is notified when the company send him an Interview Proposal
SP.22	The student is notified whenever his application's status has been changed by the company

Table 1.3: Students&Companies' Shared Phenomena

### **1.3 Definitions, Acronyms, Abbreviations**

In this section there are all the definitions, acronyms, and abbreviations that will be used in the subsequent discussions and that are essential to be clarified.

### **1.3.1 Definitions**

In this section some key definitions, which may be useful to know before proceeding, are listed.

Term	Definition
Students	One of the two main users of the platform, the ones who are actively looking for internship opportunities.
Companies	The other relevant users of the platform, the ones that offer internship opportunities
Internship	A temporary position that allows students to gain practical experience in a professional setting
Selection Process	The set of activities conducted during the evaluation of students, after their application approval; it includes interview scheduling, assessments and skill tests to determine whether they are qualified to participate in the internship
System	The collection of hardware and software tools that deliver the desired service, referred to here as S&C in its entirety.

Table 1.4: Definitions

### **1.3.2 Acronyms**

In order to avoid any misunderstanding, a list of acronyms used in the following sections is provided in the table below:

Acronyms	Meaning
S&C	Students&Companies
CV	Curriculum Vitae
HR	Human Resources
GDPR	General Data Protection Regulation
HTTPS	Hyper Text Transfer Protocol Secure
HTTP	Hyper Text Transfer Protocol
WiFi	Wireless Fidelity
LTE	Long Term Evolution
3G	Third-Generation Wireless
4G	Fourth-Generation Wireless
5G	Fifth-Generation Wireless
TLS	Transportation Layer Security
SHA-256	Secure Hash Algorithm
AI	Artificial Intelligence
NLP	Natural Language Processing

Table 1.5: Acronyms

### 1.3.3 Abbreviations

In this section it is reported the table of the abbreviations used in the document:

Abbreviations	Meaning
WP	World Phenomena
SP	Shared Phenomena
G	Goal
R	Requirement
D	Domain Assumption
w.r.t.	with reference to
e.g.	exempli gratia
i.e.	id est
etc.	etcetera

Table 1.6: Abbreviations

### 1.4 Revision History

This section highlights the updates made to the document throughout its compilation process.

Date	Revision	Notes
22/11/2024	v.3.0	Final Release
07/01/2025	v.3.1	Refinement of some requirements and correction of typos

### 1.5 Reference Documents

The following documents have been indispensable in the creation of this document:

- *Course slides on WeeBeep*
- *RASD assignament document*
- *RASD review by Prof. M. Camilli*

## 1.6 Document Structure

This RASD Document is structured as follows:

1. **Introduction:** this section outlines the purpose of the document, emphasizing the primary objectives, target audience, and the identification of the product and application domain.  
It also covers the description of the world and shared phenomena, along with definitions of key terms.
2. **Overall Description:** this chapter provides an overview of possible scenarios for the platform and details the assumptions regarding the application domain.
3. **Specific Requirements:** this section presents a more detailed explanation of the requirements compared to the "Overall Description".  
It includes functional requirements illustrated through use case diagrams, as well as sequence and activity diagrams.
4. **Formal Analysis Using Alloy:** this chapter contains Alloy models, used to describe the application domain and its properties.
5. **References:** this section lists all the documents and sources referenced in the creation of the RASD.

## *Chapter 2*

---

# Overall Description

---

## 2.1 Product Perspective

This section presents a comprehensive list of real-world scenarios, alongside with diagrams, in order to provide a deeper understanding of shared phenomena.

Thus, each scenario is thoroughly analyzed to highlight key interactions and processes, while the diagrams are used to illustrate the underlying dynamics, relationships, and patterns.

### 2.1.1 Scenarios

#### **Scenario A: The student's approach for the Internship Application**

Bob, a final-year computer science student, is seeking an internship in the field of Artificial Intelligence (A.I.), with a particular focus on working with Transformer architectures in Natural Language Processing (N.L.P.) tasks.

To tailor his search to his exact preferences, he opens the S&C app and filters internships based on the specific criteria listed above.

After reviewing the available options, Bob ultimately chooses to apply to the ClosedAI's internship and submits his application.

Bob can now easily monitor the status of his application and view all the related details, such as the submission date and the internship information, directly from his profile.

#### **Scenario B: A company launches a New Internship Program**

UnderTheData, a leading tech company, is seeking data science interns for an exciting summer project. Due to the summer schedule, the internship offers paid compensation and, furthermore, provides a unique opportunity for interns to engage with prominent figures in the tech industry. Additionally, successful interns may have the chance for future job placement.

To attract highly capable students for this major project, UnderTheData decides to post the opportunity on its profile on the S&C platform, detailing all key aspects of the internship (i.e. compensation, networking opportunities, etc...).

#### **Scenario C: The company's Internship Application Review**

GeneralMech Company recently posted an internship opportunity for mechanical engineering students. Among the applicants, Alice, a highly skilled and experienced student, submitted her application.

Upon receiving it, the company set her application's status to "Under Review" and verified that she met all the minimum requirements by assessing the CV attached to her application.

Impressed by the Alice's qualifications, GeneralMech approved her application by setting its state to "Selection Process" and proceeded to the next stage.

#### **Scenario D: The company's Internship Selection Process**

GeneralMech recently approved the application of Alice, a talented mechanical engineering student, marking the start of her selection process.

Leveraging the S&C platform with its A.I. Questionnaire Generator tool, the company generated a set of questions by providing a tailored prompt to the system; the AI-produced questions were then reviewed and customized by the company to align them with its specific needs and sent to Alice through the platform.

After Alice submitted the answers to the questionnaire, GeneralMech reviewed her impressive responses and, recognizing her outstanding performance, decided to schedule an online interview.

Using the S&C platform, the company proposed a date for the interview but, unfortunately, Alice was unavailable on the suggested date and proposed an alternative; the company accepted her proposal and sent a revised interview proposal with the agreed-upon date.

Upon Alice's acceptance, the system generated an interview link for both parties to access to the meeting on the scheduled date.

After a successful interview, GeneralMech officially confirmed Alice's selection.

#### **Scenario E: Addressing Internship Challenges through real-time Feedbacks**

Carlo, a student who recently started an internship at NewSolution Company, encountered challenges during his experience, primarily due to a significant lack of communication from his supervisor. This communication gap led to poor performance on a project and hindered his progress; concerned about the potential negative impact on his internship, Carlo decided to utilize the feedback and problem-reporting feature of the S&C platform to raise a complaint.

After receiving Carlo's complaint, the company acted swiftly to address the issue and improve the situation.

Once the problem was resolved, Carlo updated the status through the S&C platform, confirming that the complaint had been successfully solved.

Impressed by the speed and efficiency with which the company answered to his needs, Carlo also chose to leave a detailed comment describing his experience and highlighting the positive outcome.

#### **2.1.2 Class Diagram**

This section illustrates the Class Diagram, providing a clear representation of the relationships between the various classes present in the system.

In order to build the class diagram for the S&C platform, it is essential to define the key classes involved:

- **Student:** It represents university students seeking internship opportunities
- **Company:** It represents companies offering internship opportunities via the S&C platform

- **Internship:** It represents the positions offered by companies to students
- **Application:** It represents the student's submission to a company, expressing his request to participate in an internship experience proposed by the company itself
- **Selection Process:** It outlines the process between a student and a company to determine the student's eligibility for internship participation
- **Feedback:** It represents the channels for providing feedback, reporting issues, and monitoring the progress of the internship between students and companies. Feedback serves as a general class, which can be further specialized into two distinct categories:
  - \* **Complaint:** It represents feedback that requires action from the receiving party (i.e. if it is submitted by the student, then the company is responsible for addressing the issue, and viceversa)
  - \* **Comment:** It represents a straightforward insight into the ongoing experience, without requiring any action
- **C.V.:** It summarizes a student's skills, experience and attitudes

## CHAPTER 2. OVERALL DESCRIPTION

---

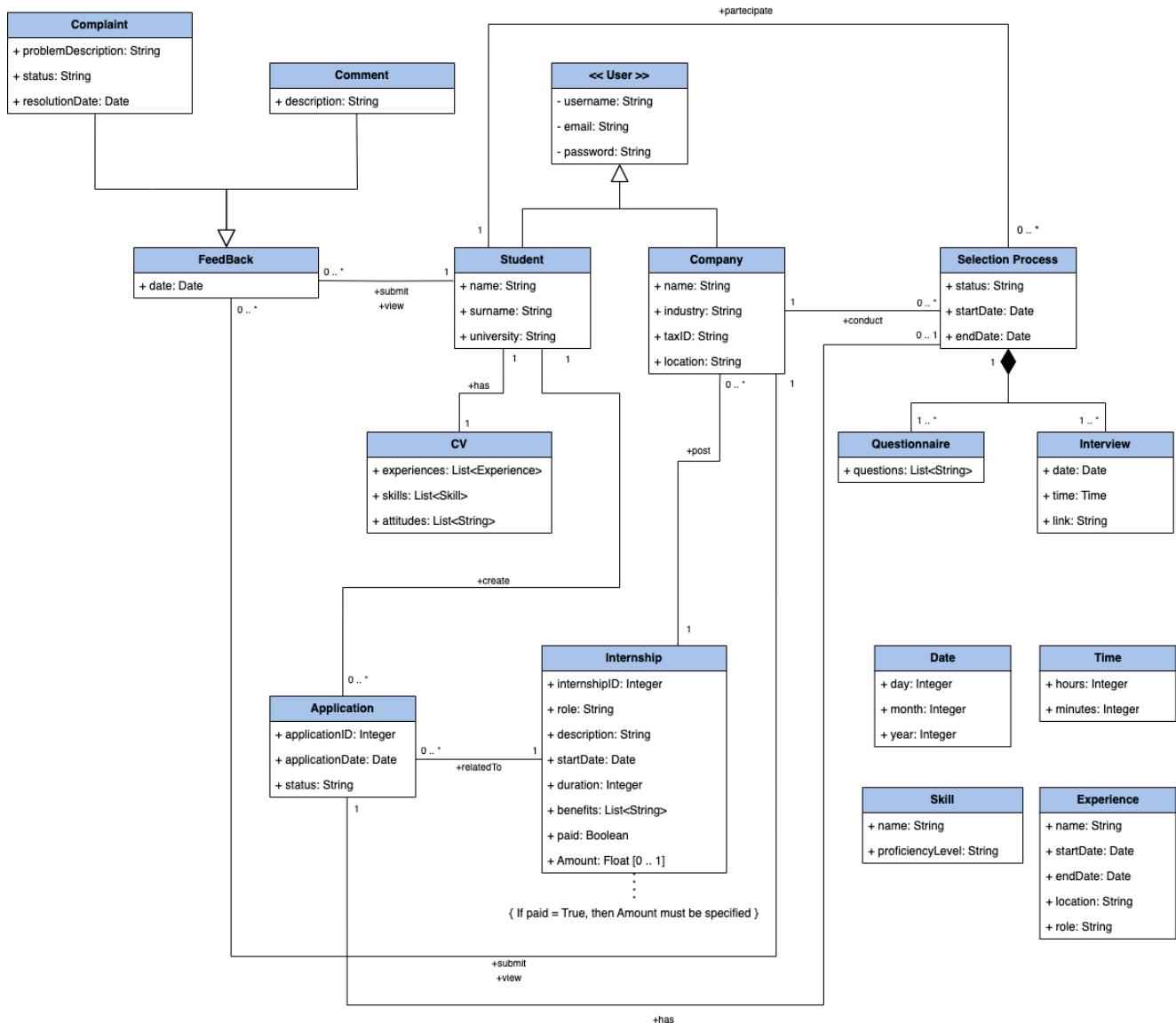


Figure 2.1: UML Class Diagram

### **2.1.3 State Diagram**

State diagrams represent the dynamic behavior of individual objects with complex lifecycles, showing the sequence of states they move through in response to triggering events and the actions that follow each transition.

In this section, the state diagrams representing the system's overall behavior are presented.

#### **Authentication**

In Figure 2.2, the authentication process for both students and companies is illustrated. Initially users, either the students and the companies, have the option to register: if the registration fails, then the system asks them to retry the process; otherwise, if the registration is successful, users can proceed to log in.

In the case in which the student or company already has an account, they can directly log in: upon entering valid credentials, the system displays the homepage related to their specific role; if the credentials are incorrect, the user is required to re-enter them.

This state diagram is essential, as it establishes the foundational step required for performing any action on the platform: indeed, each subsequent state diagram will reference this process.

#### **Internship Application**

As shown in Figure 2.3, the state diagram illustrates the process of creating an internship application. After logging in successfully, the student can choose either to view the complete list of available internship opportunities or to filter them based on specific criteria.

If the student finds a suitable internship opportunity, he can submit an application and view the application's dashboard, where he can basically monitor the relevant information and track the status of his application. Additionally, the system provides the option to delete the application if needed.

#### **Student Evaluation**

The state diagram in Figure 2.4 illustrates the process of evaluating and selecting a student's internship application, focusing primarily on the internship status for a given student from the company's perspective (i.e. Application Evaluation -> Student Evaluation -> Questionnaire Creation / Interview Scheduling -> Result). Although this process involves two elements, the student and the company, and other types of diagrams may be needed (as will be discussed in later sections), this diagram is valuable because it emphasizes the steps from the company's viewpoint, allowing us to track how the internship application status evolves in response to specific actions.

Once logged in, the company can review a student's application for an open internship position: if the initial evaluation is positive, the selection process begins and it may include submitting questionnaires to the student or scheduling interviews. Based on the evaluation of these elements, the company then decides whether to accept or reject the student for the internship position.

### **Internship Evaluation**

This state diagram (see figure 2.5) represents the progression of the shared interaction space between the student and the company during the internship. Notably, this diagram applies to both students and companies, as either party can report issues and provide feedback on the ongoing experience; however, for the sake of clarity, the process is described from the student's perspective.

After completing the authentication phase, the student can choose either to leave feedback on his internship experience or to file a complaint. In the case of a complaint, once it is submitted, the issue is expected to be addressed and resolved; after the resolution, the complaint is formally closed.

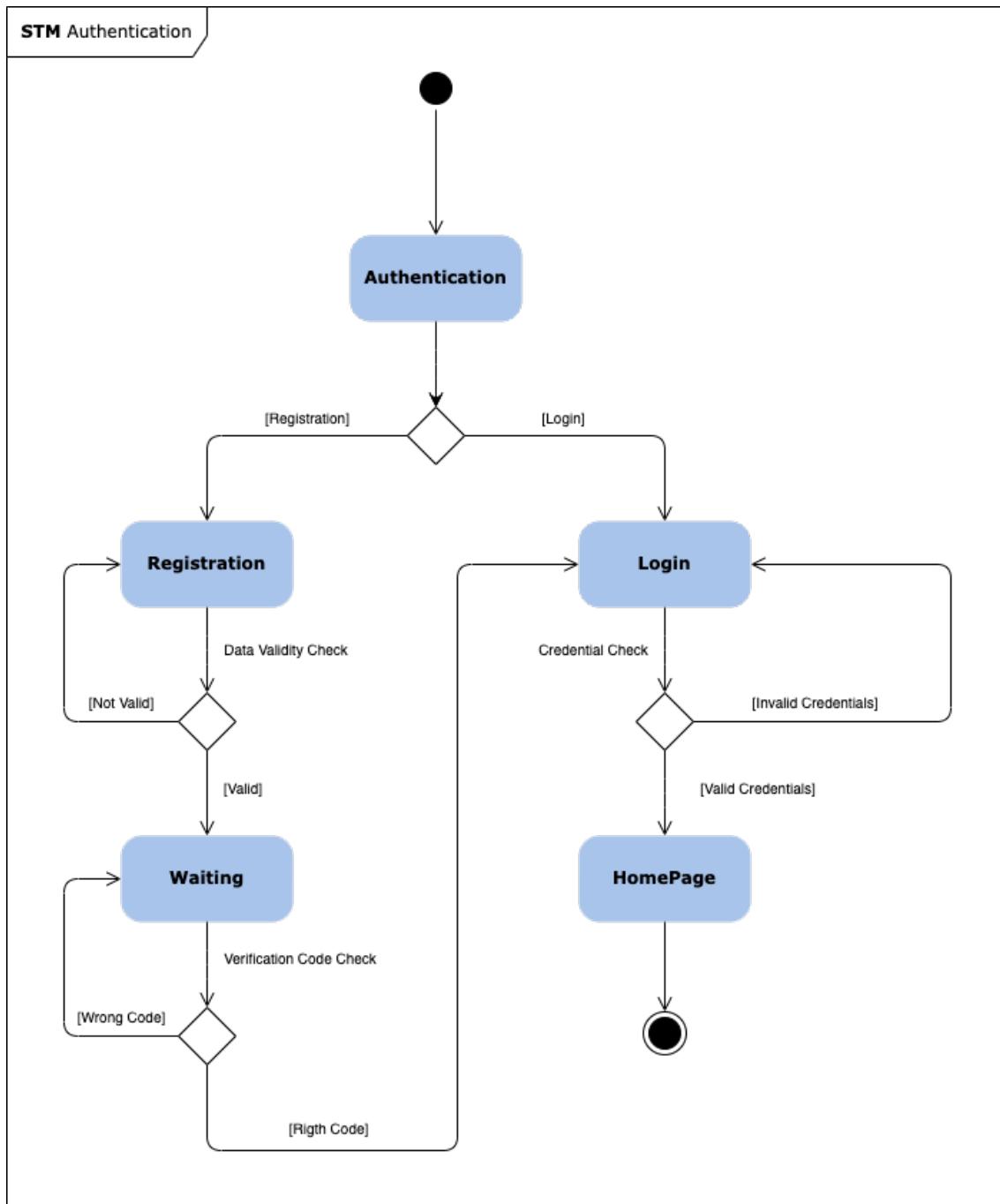


Figure 2.2: Authentication State Diagram

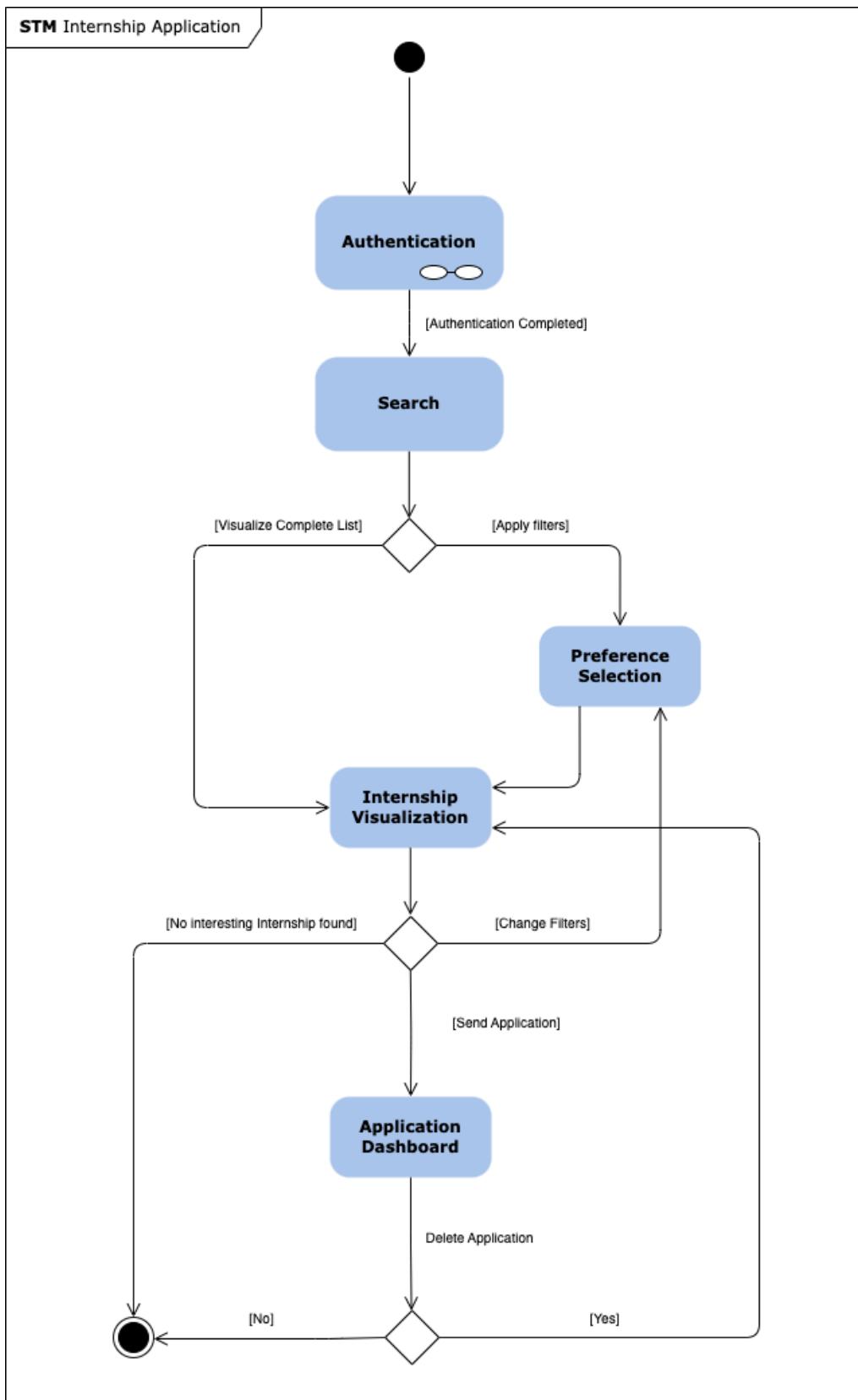


Figure 2.3: Internship Application State Diagram

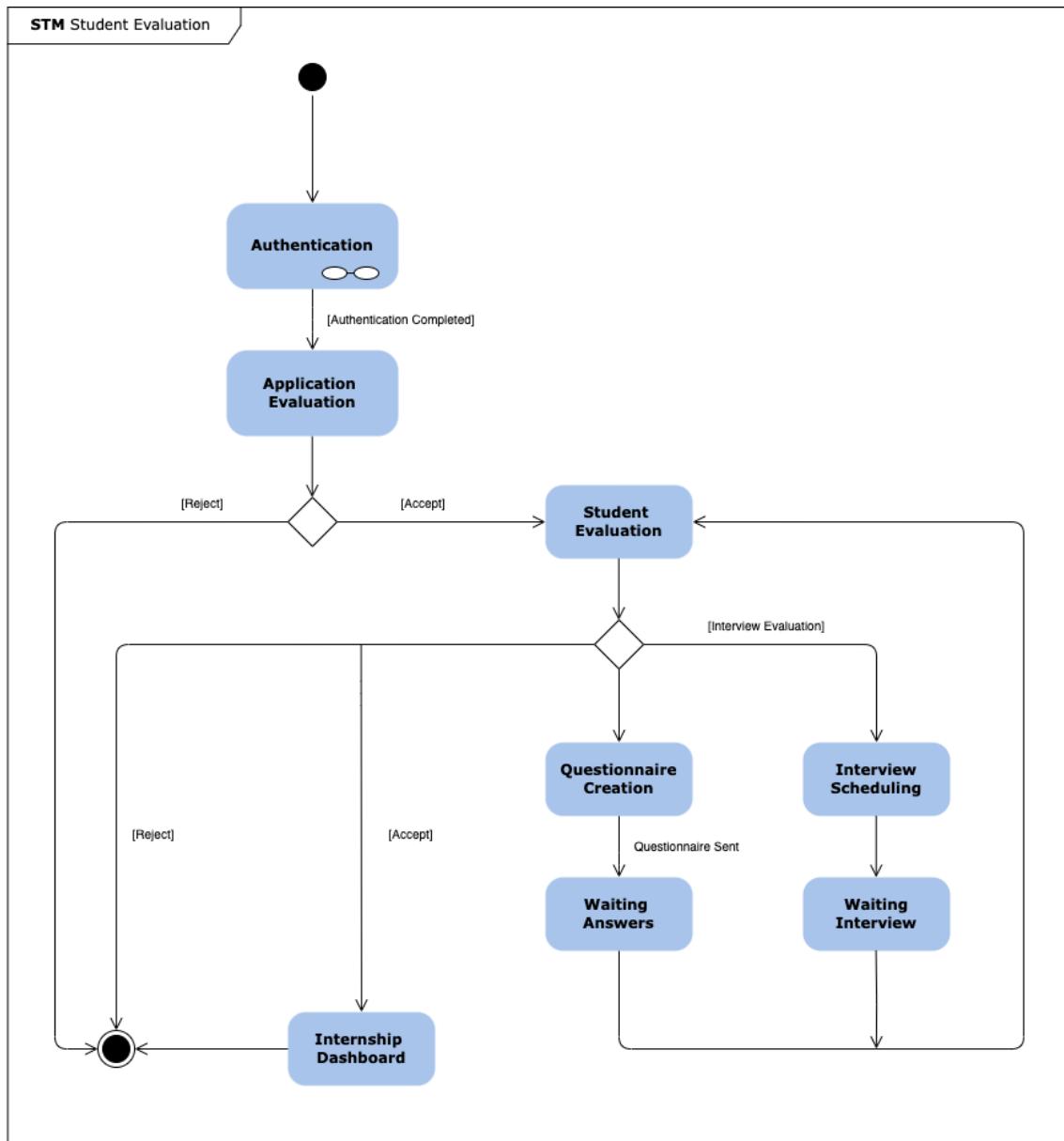


Figure 2.4: Student Evaluation State Diagram

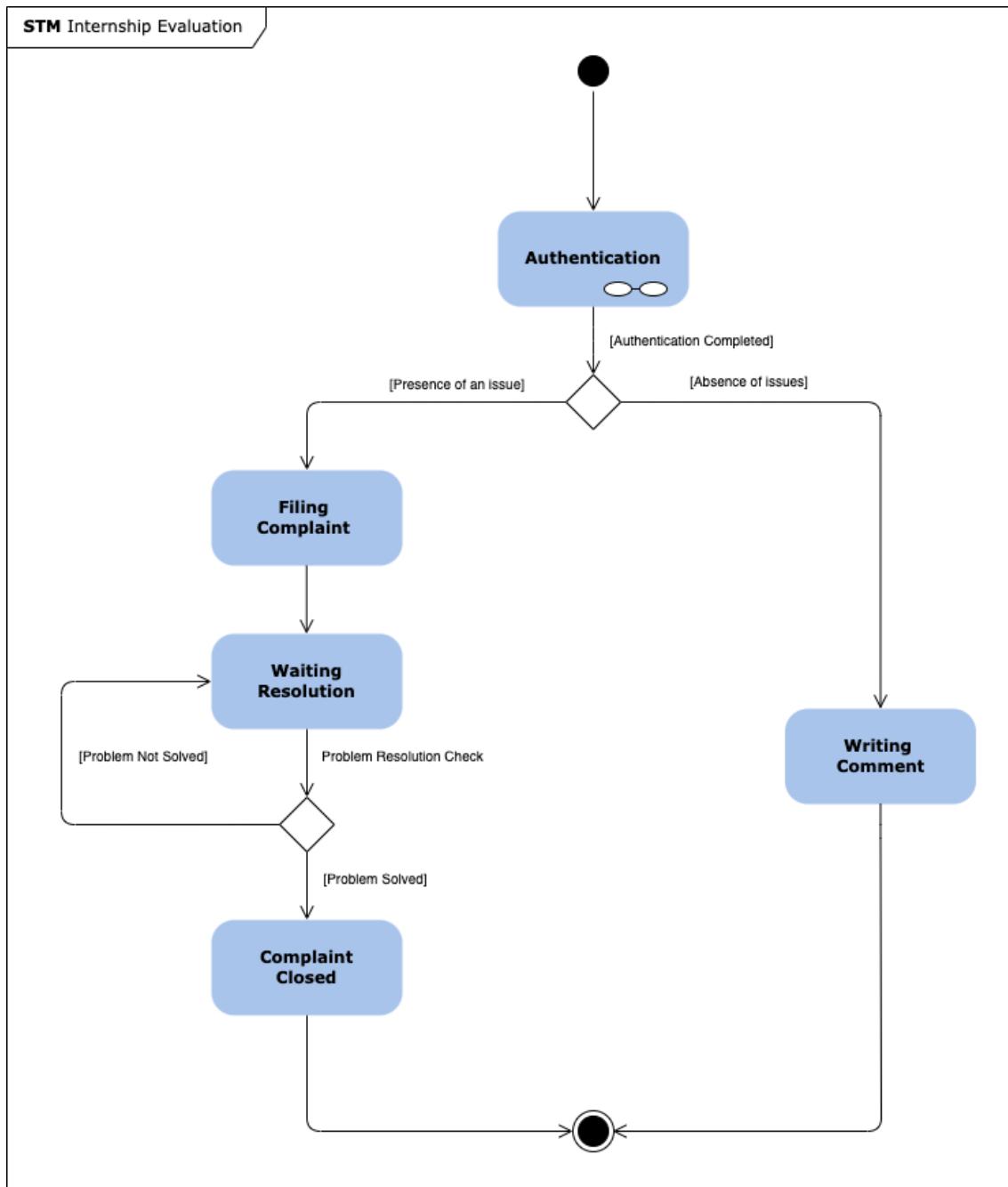


Figure 2.5: Internship Evaluation State Diagram

## **2.2 Product Functions**

This section provides a detailed description of each functionality offered by the platform in relation to the goals outlined in section 1.1.1

### **2.2.1 Authentication Functions**

The system allows both students and companies to create accounts and subsequently log in, though the registration process differs for each of them, requiring different kinds of information. Furthermore, the permissions and access levels granted to companies and students are also completely different.

#### **Function A: Student Authentication**

In order to manage the student access, the system can distinguish between two key operations:

- **Registration:** a new student must provide a username, password, email and university-related information (e.g. institution's name, student ID, etc...).  
To complete the registration, the student must confirm his email address by entering an OTP code sent to the email immediately after submission.
- **Log-In:** students can access to the platform using username/email and password

#### **Function B: Company Authentication**

In order to manage the company access, the system can distinguish between two key operations:

- **Registration:** a new company must provide a company name, password, corporate email and some specific business information (i.e. business registration number, tax ID, etc...).  
To complete the registration, the company must confirm his corporate email address by entering an OTP code sent to the email immediately after submission.
- **Log-In:** companies can access to the platform by entering their corporate email and password

### **2.2.2 Managing Application Functions**

The S&C platform provides a set of functionalities that are used by the students to manage their applications.

#### **Function A: Internship Search**

The platform enables students to proactively search for internship opportunities: they can either browse the full list of available internships or use filters to narrow down options based on specific preferences such as location, role, field of interest, additional benefits and more.

### **Function B: Internship Application**

Through the S&C platform, students can submit applications directly to the companies offering the selected internships.

The system will automatically attach the CV uploaded to the student's profile to each of his application, ensuring a smooth and efficient process.

### **Function C: Application Monitoring**

Students can use the platform to monitor the detailed information on all the submitted applications, including application status updates.

Additionally, they can withdraw applications that have not yet been processed by the company.

### **Function D: Changing Application Status**

The system automatically notifies via email the student whenever there is a change in the status of one of their application.

### **Function E: Sustaining Selection Process**

Once an application reaches the "Selection Process" stage, students can participate in further screening activities through the platform, that are completing questionnaires and scheduling interviews proposed by the hiring company.

## **2.2.3 Managing Internship Functions**

The system provides companies with a suite of features designed to manage internship opportunities and evaluate student applications efficiently.

### **Function A: Internship Offer Management**

Companies can create, update, and manage their internship listings directly on the platform, specifying important details such as role, location, qualifications, benefits and more.

Additionally, they have the ability to monitor all applications received for each specific internship position.

### **Function B: Application Review**

The platform enables companies to review student applications, from which it is possible to consult each candidate's CV.

### **Function C: Questionnaire Management**

Companies have access to a robust set of tools for creating structured questionnaires; these questionnaires can be customized and sent to applicants as part of the selection process.

### **Function D: Interview Scheduling**

The platform provides a complete set of features to facilitate the interview process, enabling companies to schedule interviews with candidates.

#### **2.2.4 FeedBack Functions**

The platform includes features that allow both students and companies to share feedback about their ongoing experience and report any issues: this enables open communication and helps to address any concerns with the other party.

##### **Function A: Comment**

Users involved in the internship, either students or company, can share their thoughts on the internship's progress by leaving comments.

They can also view comments posted by the other party.

##### **Function B: Managing Issue**

Both students and companies can report issues encountered with the other party; once a reported problem is resolved, the user can mark the complaint as "Solved".

##### **Function C: Complaint Creation**

The system automatically notifies interested parties when a complaint is created about them.

### **2.3 User Characteristics**

The Student&Company platform supports two primary user categories:

- **Students:** the primary users of the system, students use the platform to search for and apply to available internship opportunities
- **Companies:** representing the providers of internship opportunities, companies use the platform to post internships, review applications and interact with potential candidates

### **2.4 Assumptions, dependencies and constraints**

#### **2.4.1 Assumptions and Dependencies**

This section outlines the key assumptions made regarding the domain of interest for the Student&Company platform.

<b>Assumptions</b>	<b>Description</b>
D.1	Both students and companies must have an internet connection
D.2	Students must provide consent for the system to store their data, such as their CVs and other relevant information
D.3	Students must have an email address to sign-up
D.4	Companies must have a corporate email address and all the business information required to sign-up

Next Page→

← continued from the previous page

<b>Assumptions</b>	<b>Description</b>
D.5	Companies publish truthful and detailed descriptions of internship proposals, accurately describing tasks to be performed, application domain, relevant technologies that will be used, etc...
D.6	Both students and companies will use the platform's feedback and complaint features responsibly, providing truthful and unbiased comments
D.7	When a complaint is reported, the responsible party, either the student or the company, will address the issue
D.8	When a complaint is solved, the responsible party, either the student or the company, will report its resolution
D.9	Companies ensure that their internship proposals comply with all applicable local labor laws and regulations
D.10	Companies actually check the list of applications for their internship and evaluate them
D.11	Applications are evaluated in a fair way, without any form of preference / discrimination
D.12	There is some interest, student-side, into applying in internship proposed by companies that use this software
D.13	There is some interest, company-side, into looking for students that use this software
D.14	When a student receives an interview proposal, he will respond to it
D.15	When a student propose a new date for an interview, the company will respond to the proposal
D.16	Students must upload their CV

Table 2.1: Students&Companies' Domain Assumptions

#### **2.4.2 Constraints**

This section highlights the key limitations and constraints that our system-to-be will encounter.

##### **Privacy Constraints**

The S&C system collects and utilizes student information to provide companies with all the relevant data needed to assess an internship application; consequently, the students' data must not be, under no circumstances, disclosed to third parties or used for marketing purposes.

##### **Hardware Constraints**

The S&C platform is relatively simple and doesn't require highly complex components to function properly, but, despite this setting, it is not compatible with all types of devices. Below there is a list of essential elements needed for the system to operate effectively.

- LTE/3G/4G/5G or Wi-Fi 4+ Connection
- Web Browser (e.g. Opera, Safari, Chrome, Firefox) that supports HTML 5

### **Software Constraints**

The system can operate without the need of any additional software but, it is important to notice that, in order to access and leverage its full range of features, some additional integrations are necessary:

- **Meeting APIs:** these APIs are essential for enabling companies to generate interview links, facilitating the scheduling process
- **A.I. APIs:** these APIs optimize the questionnaire creation process by generating potential context-aware questions derived from the prompts provided by the company

## *Chapter 3*

# Specific Requirements

## 3.1 External Interface Requirements

This section analyzes in detail the specific functional requirements of the S&C system, highlighting its interactions and integrations with other components.

### 3.1.1 Students Interfaces

This section addresses the specific interfaces associated with the 'student' user.

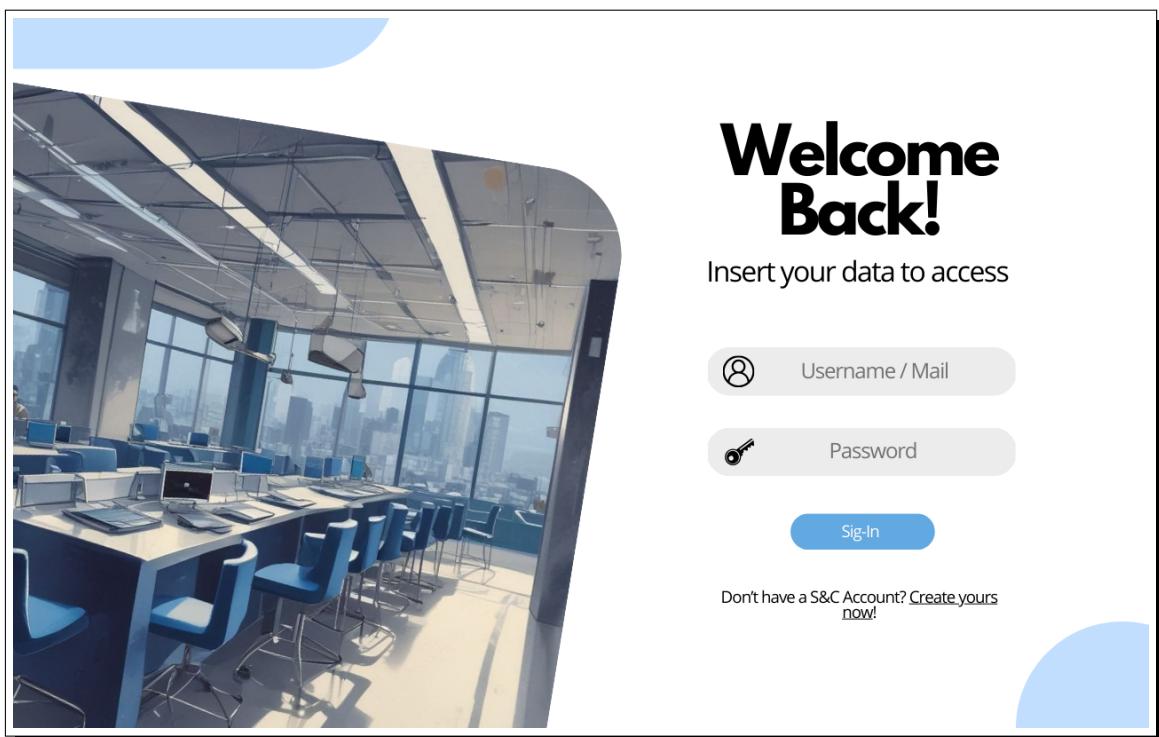


Figure 3.1: Student and Company Login

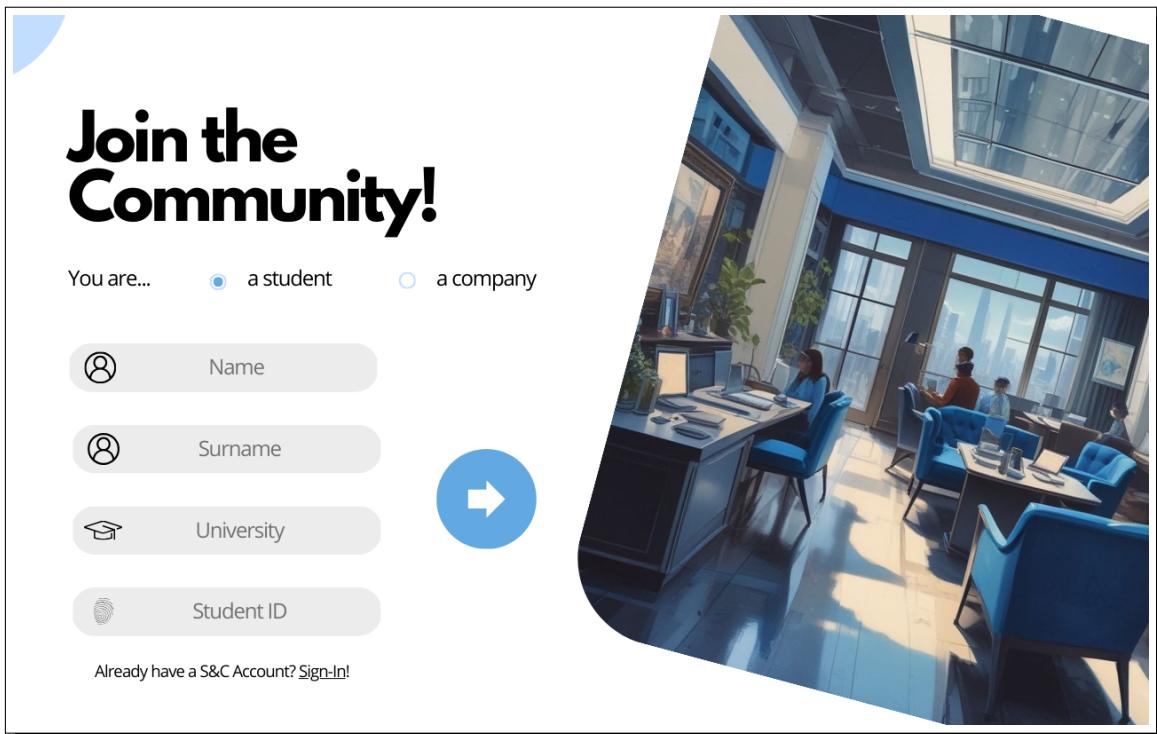


Figure 3.2: Student Sign-Up (A)

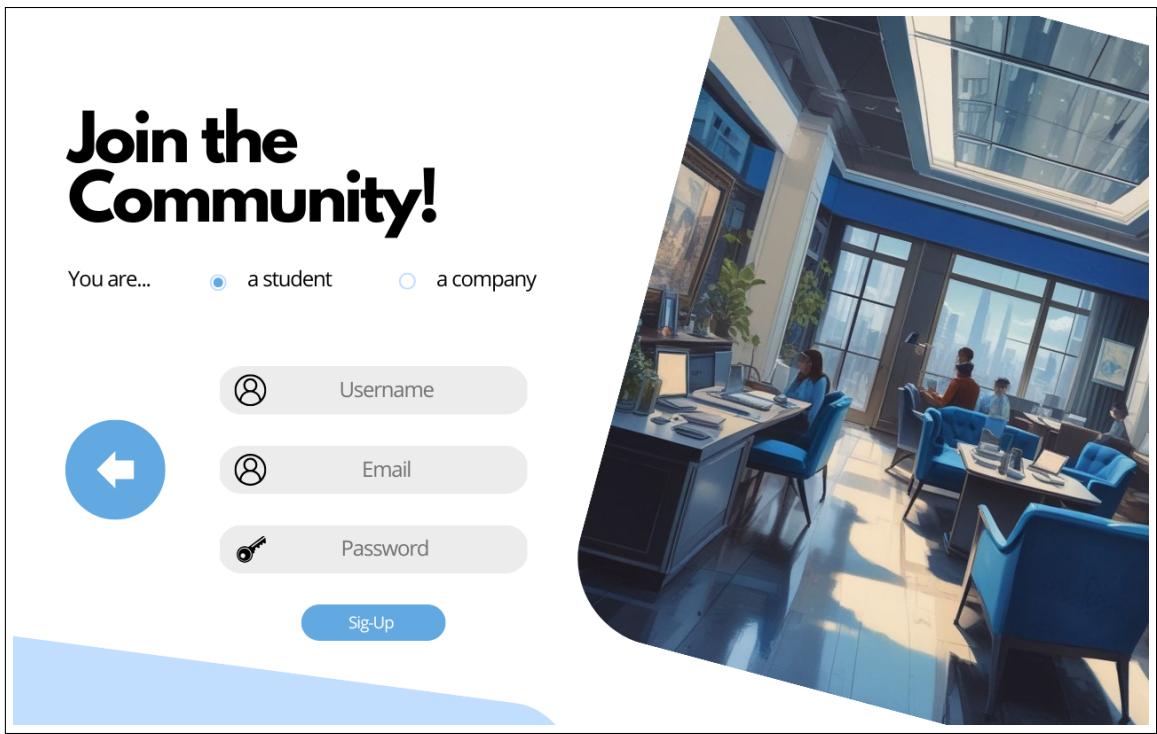


Figure 3.3: Student Sign-Up (B)

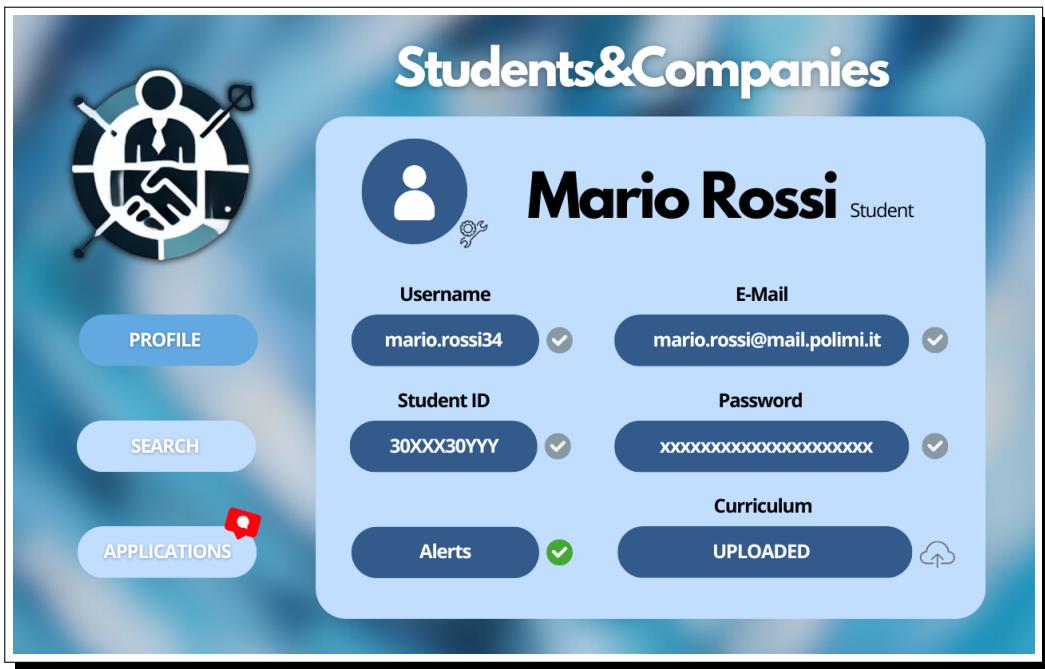


Figure 3.4: Student Profile

### 3.1.2 Companies Interfaces

This section addresses the specific interfaces associated with the 'company' user.

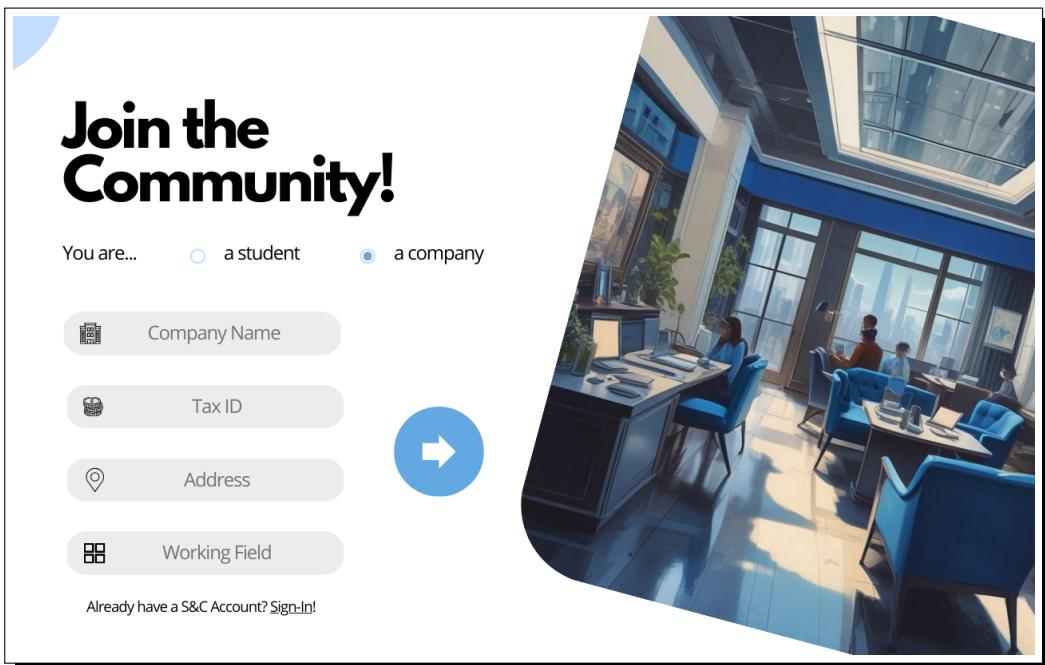


Figure 3.5: Company Sign-Up (A)

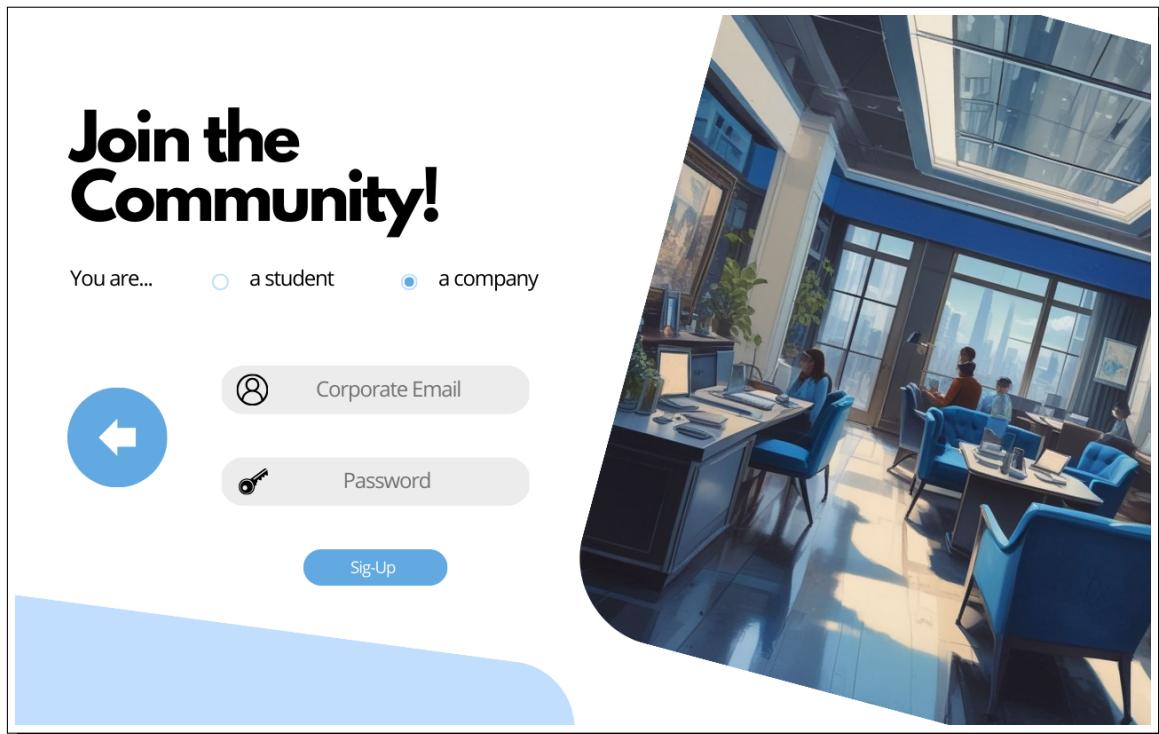


Figure 3.6: Company Sign-Up (B)

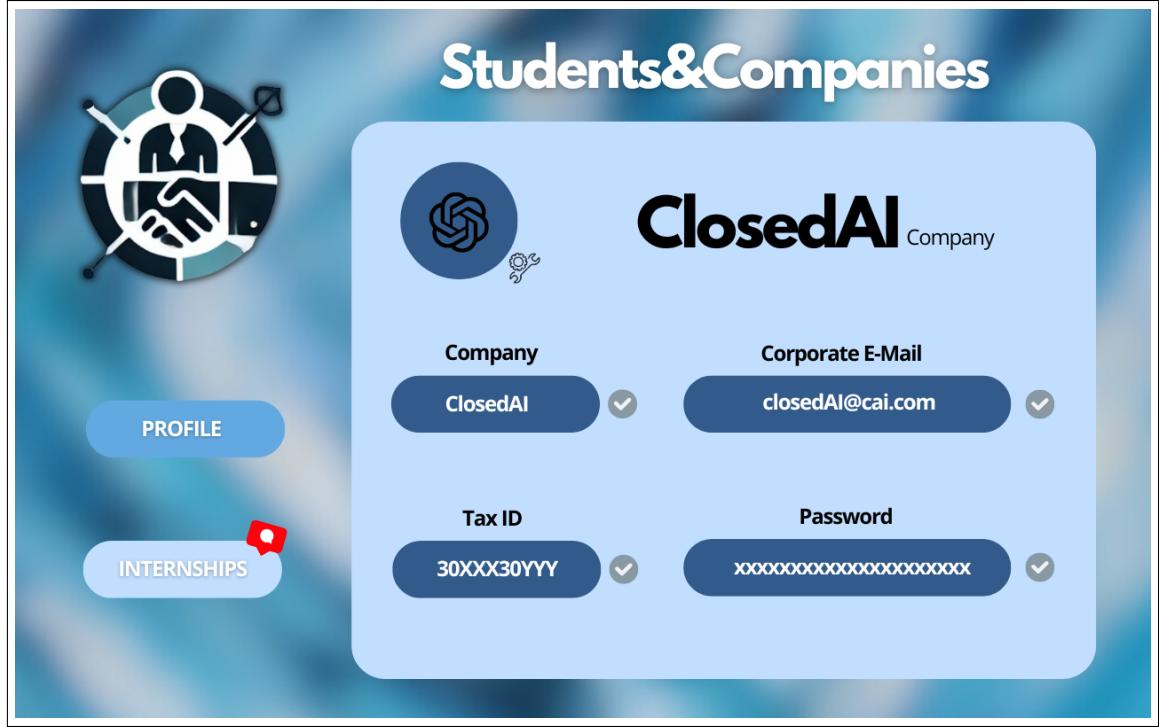


Figure 3.7: Company Profile

### **3.1.3 Hardware Interfaces**

In order to effectively use the system, one of the following devices is required:

- **Smartphones**
- **PCs**

These devices enable users, both students and companies, to access the platform through a web browser.

### **3.1.4 Software Interfaces**

This section provides an overview of the essential connections between the system-to-be and the external software components, highlighting the critical functionalities that require such an integration:

- **Meeting APIs Integration:** in order to enable companies to schedule interviews with candidates applying for internship positions, the system must be integrated with widely used meeting platforms.

This includes APIs for applications such as Microsoft Teams and Google Meet.

It is important to notice that by supporting multiple APIs, the system ensures both redundancy for uninterrupted operations and flexibility for companies to choose their preferred meeting platform.

- **A.I. APIs Integration:** the system uses ChatGPT APIs to facilitate the generation of interview questions based on user-provided prompts.

In this case ChatGPT is exclusively chosen due to its superior ability to provide high-quality, contextually relevant answers compared to other available A.I. systems

### **3.1.5 Communication Interfaces**

This section outlines the requirements for all communication functions within the system.

The S&C system will use the HTTPS application layer protocol to perform all the communications. Accordingly, all the devices interacting with the platform must be connected to the internet via Wi-Fi or via mobile network (i.e. LTE, 3G, 4G or 5G).

## **3.2 Functional Requirements**

This section will define the Use Case Diagrams, provide a detailed description of the various use cases and link each of them to its corresponding sequence diagram.

In addition, it will also report the requirements for the S&C System.

### 3.2.1 Use Case Diagrams

In order to guarantee clarity and readability, the Use Case Diagrams are presented from three distinct perspectives:

- **Platform Registration Use Case** (Figure 3.8): this diagram involves two primary actors, the unregistered User and the unregistered Company, and represents the process of signing up within the system
- **Student Use Case** (Figure 3.9): in this diagram, the Student is the main actor; it provides a detailed analysis of the various operations that the Student performs while interacting with the system, capturing all the relevant use cases from the student's perspective
- **Company Use Case** (Figure 3.10): this diagram focuses on the Company as the main actor and examines in detail the operations that the Company performs when it interacts with the system

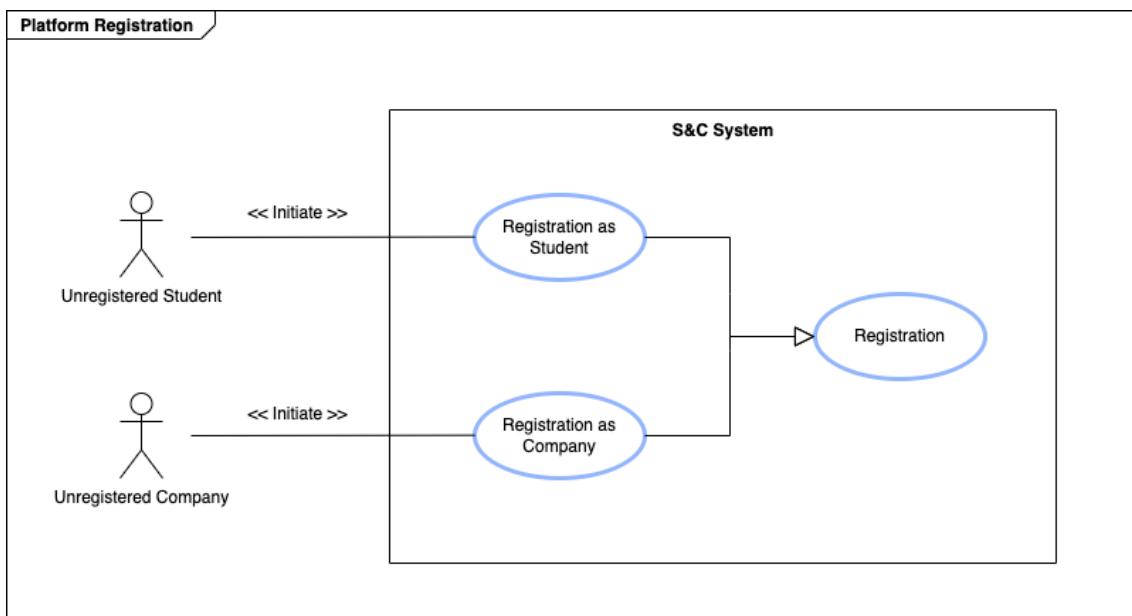


Figure 3.8: Registration Use Case Diagram

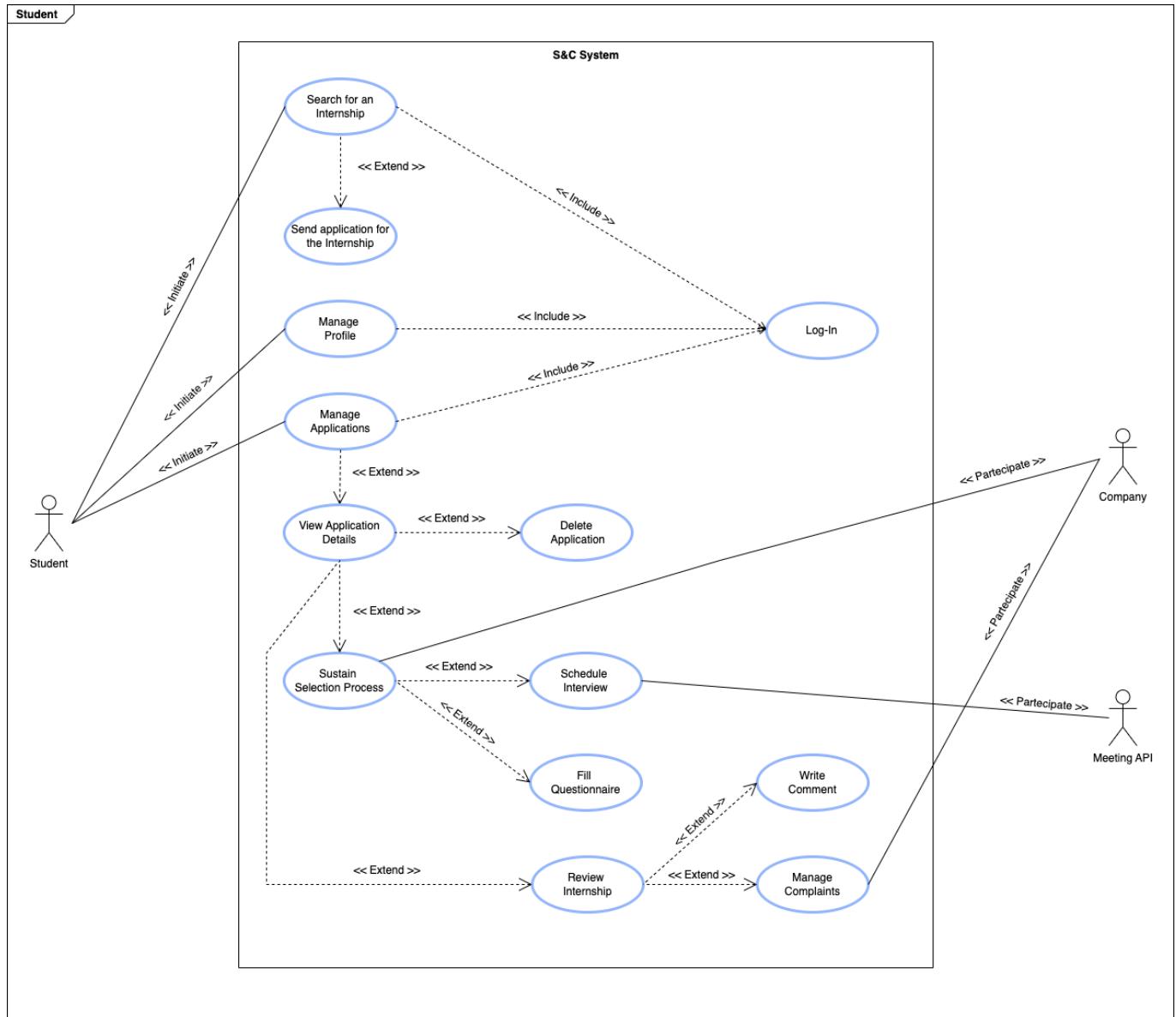


Figure 3.9: Student Use Case Diagram

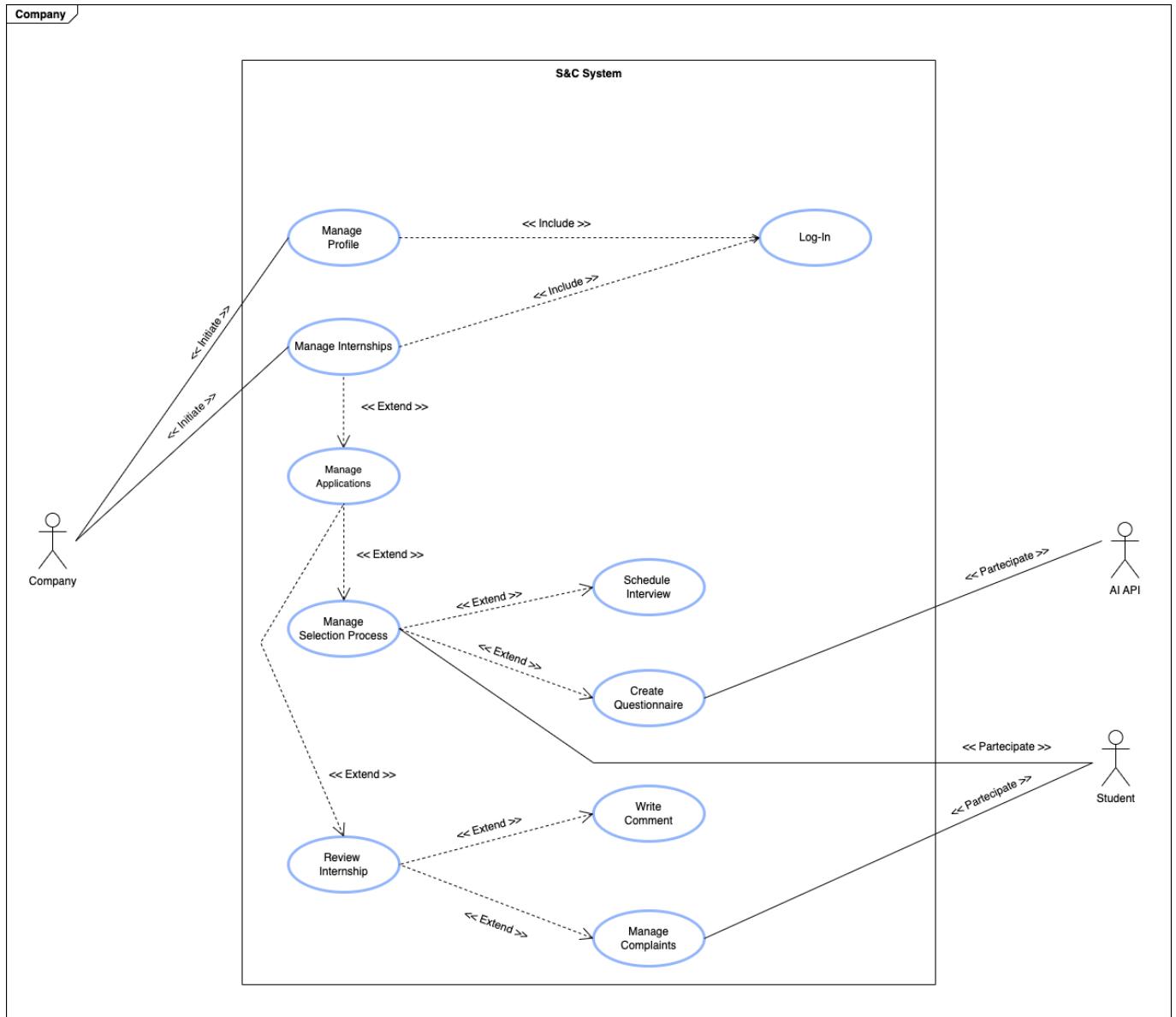


Figure 3.10: Company Use Case Diagram

### 3.2.2 Use Case Description

This section provides a comprehensive description of the various use cases involving the S&C system, offering detailed insights into the interactions and processes associated with each scenario.

*Notice that, in order to keep a concise analysis and maintain clarity, simple use cases (e.g. 'Manage Profile') and redundant ones (e.g. 'Schedule Interview', 'Submit a Comment', or 'Handle a Complaint' from the company's perspective) have been intentionally omitted.*

#### UC.1 - Registration into the system

<b>Name</b>	Register into the platform
<b>ID</b>	UC.1
<b>Actors</b>	Unregistered Student, Unregistered Company
<b>Entry Condition</b>	<p>The actor navigates to the sign-up page</p> <p>1) The actor chooses the related sign-up option (i.e. either student or company)</p> <p>2a) The unregistered student fills the sign-up form with name, surname, University, Student ID, email, username and password</p> <p>2b) The unregistered company fills the sign-up form with company name, tax ID, address, working field, corporate email and password</p> <p>3) The actor submits the registration</p> <p>4) The system sends a confirmation code to the provided email to verify the validity of the actor's email itself</p> <p>5) The actor inserts the received code on the related field</p> <p>6) The system verifies the validity of the code</p> <p>7) The system stores the actor's data</p> <p>8) The system redirects the actor to the appropriate log-in page (i.e. either student or company)</p>
<b>Event Flow</b>	
<b>Exit Condition</b>	The actor signed-up correctly
<b>Exceptions</b>	<p>1) The unregistered student provided an username or email already present in the system → The system notified the unregistered student of the error and the flow returns to step 2a</p> <p>2) The unregistered company provided a tax ID or a corporate email already present into the system → The system notified the unregistered company of the error and the flow returns to step 2b</p> <p>3) The actor inserted a wrong confirmation code → The system notifies the actor of the error and the flow returns to step 4</p>

Table 3.1: Register into the platform Use Case

**UC.2 - Access to the system**

<b>Name</b>	Log-in into the platform
<b>ID</b>	UC.2
<b>Actors</b>	Student, Company
<b>Entry Condition</b>	1) The actor navigates to the log-in page 2) The actor is already registered
<b>Event Flow</b>	1a) The student fills the log-in form with his username/mail and password 1b) The company fills the log-in form with his corporate mail and password 3) The actor submits the log-in 4) The system redirects the actor to the appropriate homepage (i.e. either student or company)
<b>Exit Condition</b>	The actor is correctly logged-in
<b>Exceptions</b>	1) The student provided an username/email not valid → The system notifies the student of the error and the flow returns to step 1a 2) The student provided a wrong password → The system notifies the student of the error and the flow returns to step 1b 3) The company provided a corporate email not valid → The system notifies the company of the error and the flow returns to step 1b 4) The company provided a wrong password → The system notifies the company of the error and the flow returns to step 1b

Table 3.2: Log-in into the platform Use Case

**UC.3 - Find an Internship and Apply**

<b>Name</b>	Send Application
<b>ID</b>	UC.3
<b>Actors</b>	Student
<b>Entry Condition</b>	1) The Student is logged-in 2) The Student navigates to the Search Internship Page 3) The student has uploaded his CV
<b>Event Flow</b>	1) The student sets the filters for his search 2) The system returns all the available internships that match the filters 3) The student analyzes the internship proposals 4) The student selects a chosen internship 5) The student press the "Send Application" button 6) The system creates the application 7) The system appends the student's CV to the application 8) The system stores the student's application 9) The system confirms to the student that the application has been sent → Go to exit condition 2
<b>Exit Condition</b>	1) No application is sent 2) The student correctly send the application to the related company
<b>Exceptions</b>	1) The student provided too strict filters and no internship is found → The system notifies the student of the error and the flow returns to step 1 2) The student does not find any interesting internship → Go to exit condition 1

Table 3.3: Send Application Use Case

**UC.4 - Visualize the details of a sent application**

<b>Name</b>	See Application Information
<b>ID</b>	UC.4
<b>Actors</b>	Student
<b>Entry Condition</b>	1) The Student is logged-in 2) The Student navigates to the Sent Application Page 3) The Student has already sent an application
<b>Event Flow</b>	1) The student see the list of the sent applications 2) The student press on a specific application 3) The student visualize the status of the application (e.g. Sent, Under Review, etc...) 4) The student visualize all the information about the related internship proposal 5) The student visualize the date of submission of the application 6) If the application's status is "Selection Process" or "Internship", the student can also view the relevant related elements (i.e. questionnaires/interview links or comments/complaints)
<b>Exit Condition</b>	The student has visualized the Application Information
<b>Exceptions</b>	<i>None</i>

Table 3.4: See Application Use Case

**UC.5 - Delete an unwanted Application**

<b>Name</b>	Delete Application
<b>ID</b>	UC.5
<b>Actors</b>	Student
<b>Entry Condition</b>	1) The Student is logged-in 2) The Student navigates to the Sent Application Page 3) The Student has already sent an application
<b>Event Flow</b>	1) The student see the list of the sent applications 2) The student press on a specific application 3) The student press the delete button 4) The system requires the confirmation of the operation 5) The student confirms the operation 6) The system changes the application's status to "Withdrawn" 7) The system confirms to the student that the operation has been successfully performed
<b>Exit Condition</b>	The student has successfully deleted the application
<b>Exceptions</b>	1) The application is already being reviewed by the company → The system notifies the student of the error and redirect him on the Specific Application Page

Table 3.5: Delete Application Use Case

**UC.6 - Schedule an interview during the Selection Process**

<b>Name</b>	Schedule Interview
<b>ID</b>	UC.6
<b>Actors</b>	Student, Company
<b>Entry Condition</b>	1) The Student is logged-in 2) The Student navigates to the Specific Application Page 3) The Company has set the status of that application to Selection Process 4) The Company has sent an Interview Request with a date
<b>Event Flow</b>	1) The student opens the Interview Proposal 2) The student visualize the proposed date 3) The student is unavailable on the proposed date 4) The student chooses another date 5) The student submits the change date request 6) The system sends to the company the date change request 7) The company replies with a new Interview Proposal 8) The system delivers to the user the Interview Proposal 9) The student confirms the Interview Proposal 10) The system notifies the company that the proposal has been accepted 11) The company press the "Generate Interview Link" Button 12) The system generates the link 13) The system sends to the company and to the student the generated link
<b>Exit Condition</b>	The student has correctly scheduled the interview
<b>Exceptions</b>	1) The company sends an interview proposal with another not feasible date → The flow returns to step 1

Table 3.6: Schedule Interview Use Case

**UC.7 - Answer to questionnaires sent by the company**

<b>Name</b>	Fill Out the Questionnaire
<b>ID</b>	UC.7
<b>Actors</b>	Student, Company
<b>Entry Condition</b>	1) The Student is logged-in 2) The Student navigates to the Specific Application Page 3) The Company has set the status of that application to Selection Process 4) The Company has sent a questionnaire to the student
<b>Event Flow</b>	1) The student opens the received questionnaire section 2) The student visualize the sent questionnaire 3) The student fill out the questionnaire 4) The student submit his answers 5) The system stores the student's answers 6) The system notifies the company that the questionnaire has been filled out 7) The system notifies the student that the questionnaire has been successfully submitted
<b>Exit Condition</b>	The student has successfully submitted the questionnaire
<b>Exceptions</b>	1) The student had not opened the questionnaire before the closing date → The system notifies the student of the missed deadline and alerts the company about the problem

Table 3.7: Fill Out a Questionnaire Use Case

**UC.8 - Write a Feedback about the on-going Internship**

<b>Name</b>	Write a comment
<b>ID</b>	UC.8
<b>Actors</b>	Student
<b>Entry Condition</b>	1) The Student is logged-in 2) The Student navigates to the Specific Application Page 3) The status of the application is set to "Internship"
<b>Event Flow</b>	1) The student opens the Feedback section 2) The student press the "Add a comment" button 3) The student fill the corresponding text block 4) The student submit the form 5) The system checks the validity of the comment 6) The system stores the student's comment 7) The system notifies the company that a new comment has been added 8) The system notifies that the comment has been successfully saved
<b>Exit Condition</b>	The student has successfully submitted the comment
<b>Exceptions</b>	The student submits an empty comment → The system notifies the student of the error and the flow returns at step 2

Table 3.8: Write a comment Use Case

**UC.9 - Manage a complaint about the on-going internship**

<b>Name</b>	Manage a Complaint
<b>ID</b>	UC.9
<b>Actors</b>	Student, Company
<b>Entry Condition</b>	1) The Student is logged-in 2) The Student navigates to the Specific Application Page 3) The status of the application is set to "Internship"
<b>Event Flow</b>	1) The student opens the Feedback section 2) The student press the "Add a complaint" button 3) The student fill the corresponding text block 4) The student submit the form 5) The system checks the validity of the complaint 6) The system stores the student's complaint 7) The system notifies the company that a new compliant has been added 8) The system notifies that the complaint has been successfully saved 9) The student waits the resolution of the problem 10) The company address the notified issue 11) The student marks the complaint as "Solved" 12) The system stores the change of the complaint's status 13) The system notifies the company about the change in the complaint's status 14) The system confirms to the student that the operation has been successfully performed
<b>Exit Condition</b>	The student has successfully submitted a compliant and solved the problem
<b>Exceptions</b>	1) The student submits an empty comment → The system notifies the student of the error and the flow returns at step 2

Table 3.9: Manage a Compliant Use Case

**UC.10 - Upload the Curriculum Vitae**

<b>Name</b>	Upload C.V.
<b>ID</b>	UC.10
<b>Actors</b>	Student
<b>Entry Condition</b>	1) The student is logged-in 2) The Company navigates to the Manage Profile Page
<b>Event Flow</b>	1) The student press the "Upload C.V." button 2) The submits the C.V. file 3) The system checks the file's validity 4) The system stores the student's C.V. 5) The system confirms to the student that the operation has been successfully performed
<b>Exit Condition</b>	The student has successfully uploaded his C.V.
<b>Exceptions</b>	The student has already uploaded a C.V. → The system asks the student whether he wants to replace it: if the student confirms, the new C.V. is uploaded; otherwise, the process is aborted

Table 3.10: Upload C.V. Use Case

**UC.11 - Publish a new Internship Proposal**

<b>Name</b>	Insert an Internship Proposal
<b>ID</b>	UC.11
<b>Actors</b>	Company
<b>Entry Condition</b>	1) The Company is logged-in 2) The Company navigates to the Published Internships Page
<b>Event Flow</b>	1) The company press the "Add an Internship" button 2) The company fill the corresponding text blocks with all the required information about the internship 3) The company submit the form 4) The system performs all the validation checks 5) The system stores the new internship proposal 6) The system confirms to the company that the operation has been successfully performed
<b>Exit Condition</b>	The company has successfully submitted a new internship proposal
<b>Exceptions</b>	The company has not correctly filled in the required fields → The system notifies the company of the incorrect field entries and the flow returns at step 2

Table 3.11: Insert an Internship Proposal Use Case

**UC.12 - Visualize and Evaluate an Internship Application**

<b>Name</b>	Manage an Application Request
<b>ID</b>	UC.12
<b>Actors</b>	Company, Student
<b>Entry Condition</b>	1) The Company is logged-in 2) The Company navigates to the Specific Internship Page 3) The Company has received an application for the considered internship
<b>Event Flow</b>	1) The company open the list of the applications for that internship 2) The company select an application to evaluate 3) The company set the state of the application to 'Under Review' 4) The system updates the application's status 5) The system notifies the student that the application's status has been changed by the company 6) The system confirms to the company that the operation has been successfully performed 7) The company analyzes the C.V. appended to the application request 8) The company accepts the application request 9) The company set the state of the application to 'Selection Process' 10) The system notifies the student that the application's status has been changed by the company 11) The system confirms to the company that the operation has been successfully performed
<b>Exit Condition</b>	The company has correctly visualized and evaluated an internship application
<b>Exceptions</b>	1) The application's status has been set to "Withdrawn" before step 3 → The system notifies the company of the update 2) The company rejects the application request → The company sets the application's status to 'Rejected'

Table 3.12: Manage an Application Request Use Case

**UC.13 - Create and Send Questionnaires to an Internship Applicant**

<b>Name</b>	Manage a Questionnaire
<b>ID</b>	UC.13
<b>Actors</b>	Company, Student, A.I. API
<b>Entry Condition</b>	1) The Company is logged-in 2) The Company navigates to the Specific Application Page 3) The Company has changed the status of an application to 'Selection Process'  1) The company opens the questionnaire section 2) The company press the 'Create a new questionnaire' button 3) The company submits a prompt that describes the kind of questions to generate 4) The system uses the A.I. API to generate a set of questions 5) The system sends the set of generated questions to the company 6) The company considers the set of generated questions valid 7) The company manually review (i.e. modify, add or delete) the questions 8) The company submits the questions 9) The system generates the structured questionnaire with those questions 10) The system stores the questionnaire 11) The system notifies the student about the submission of a new questionnaire 12) The system confirms to the company that the operation has been successfully performed
<b>Event Flow</b>	
<b>Exit Condition</b>	The company has correctly created and sent a questionnaire
<b>Exceptions</b>	1) The company does not consider the generated questions valid → The flow returns to step 3

Table 3.13: Manage a Questionnaire Use Case

**UC.14 - Analyze the answers of a Questionnaire**

<b>Name</b>	View Questionnaire's Answers
<b>ID</b>	UC.14
<b>Actors</b>	Company, Student
<b>Entry Condition</b>	1) The Company is logged-in 2) The Company navigates to the Specific Application Page 3) The Company has changed the status of an application to 'Selection Process' 4) The Company has sent a questionnaire to the student 5) The Student has answered to the questionnaire
<b>Event Flow</b>	1) The company opens the questionnaire section 2) The company opens the specific questionnaire 3) The company visualizes the answers of the questionnaire 4) The company evaluates the answers
<b>Exit Condition</b>	The company has correctly visualized and evaluated the answer to a questionnaire
<b>Exceptions</b>	<i>None</i>

Table 3.14: View Questionnaire's Answers Use Case

**UC.15 - Evaluate the Selection Process of an Internship Applicant**

<b>Name</b>	Evaluate the Selection Process
<b>ID</b>	UC.15
<b>Actors</b>	Company, Student
<b>Entry Condition</b>	1) The Company is logged-in 2) The Company navigates to the Specific Application Page 3) The Company has changed the status of an application to 'Selection Process' 4) The Company has sent at least one questionnaire or set at least one interview
<b>Event Flow</b>	1) The company decides to approve the student into the internship 2) The company sets the Application's Status to "Internship" 3) The system updates the application's status 4) The system notifies the student that the application's status has been changed by the company 5) The system confirms to the company that the operation has been successfully performed
<b>Exit Condition</b>	The company has correctly evaluated the Student's Selection Process
<b>Exceptions</b>	The company decides to reject the student for the internship → The company sets the Application's Status to "Rejected"

Table 3.15: Evaluate the Selection Process Use Case

### 3.2.3 Sequence Diagrams

To further clarify the interactions between the actors and the system, this section presents the sequence diagrams associated with the most relevant use cases outlined above.

## Sign-Up and Log-In

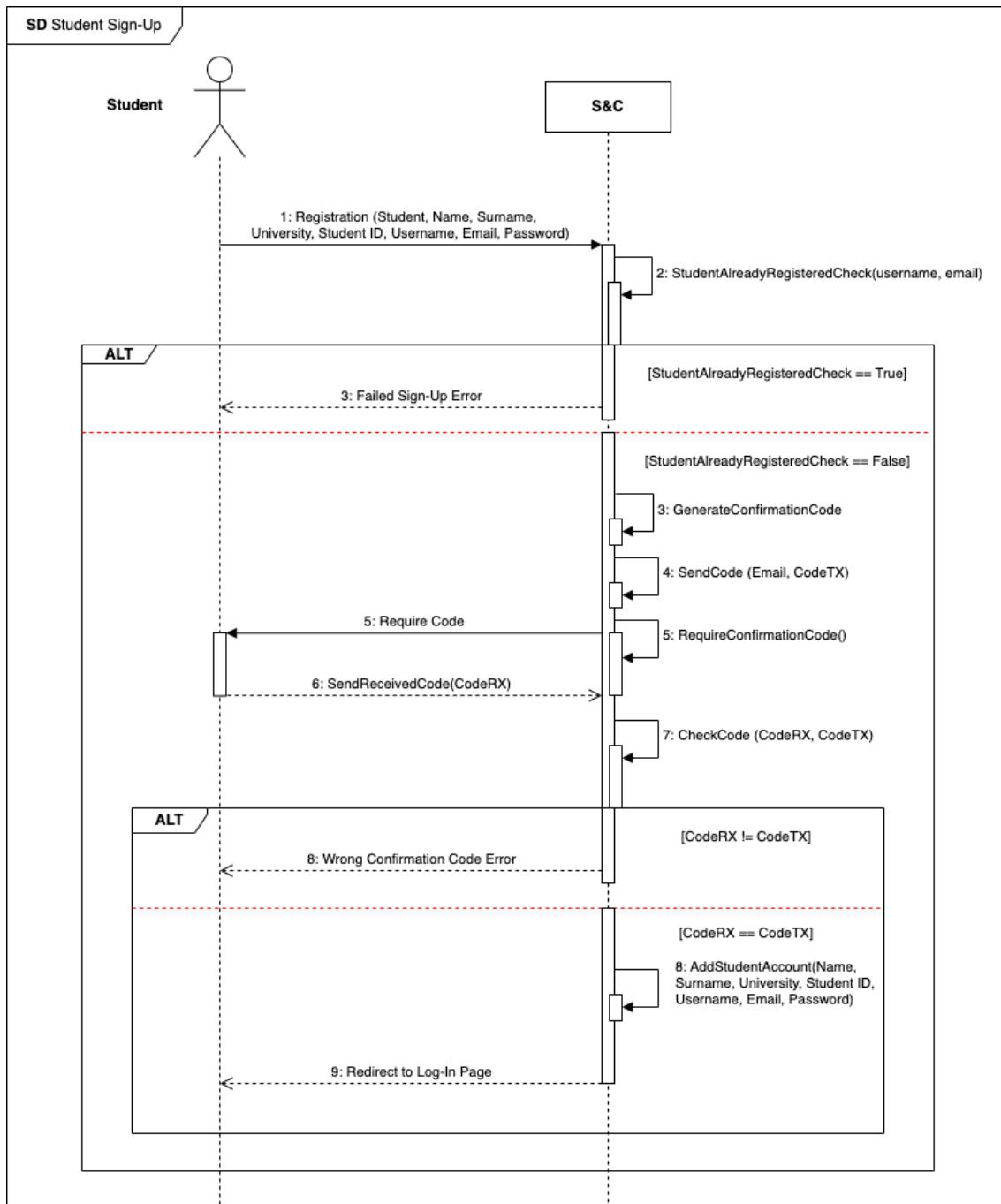


Figure 3.11: Student Registration Sequence Diagram

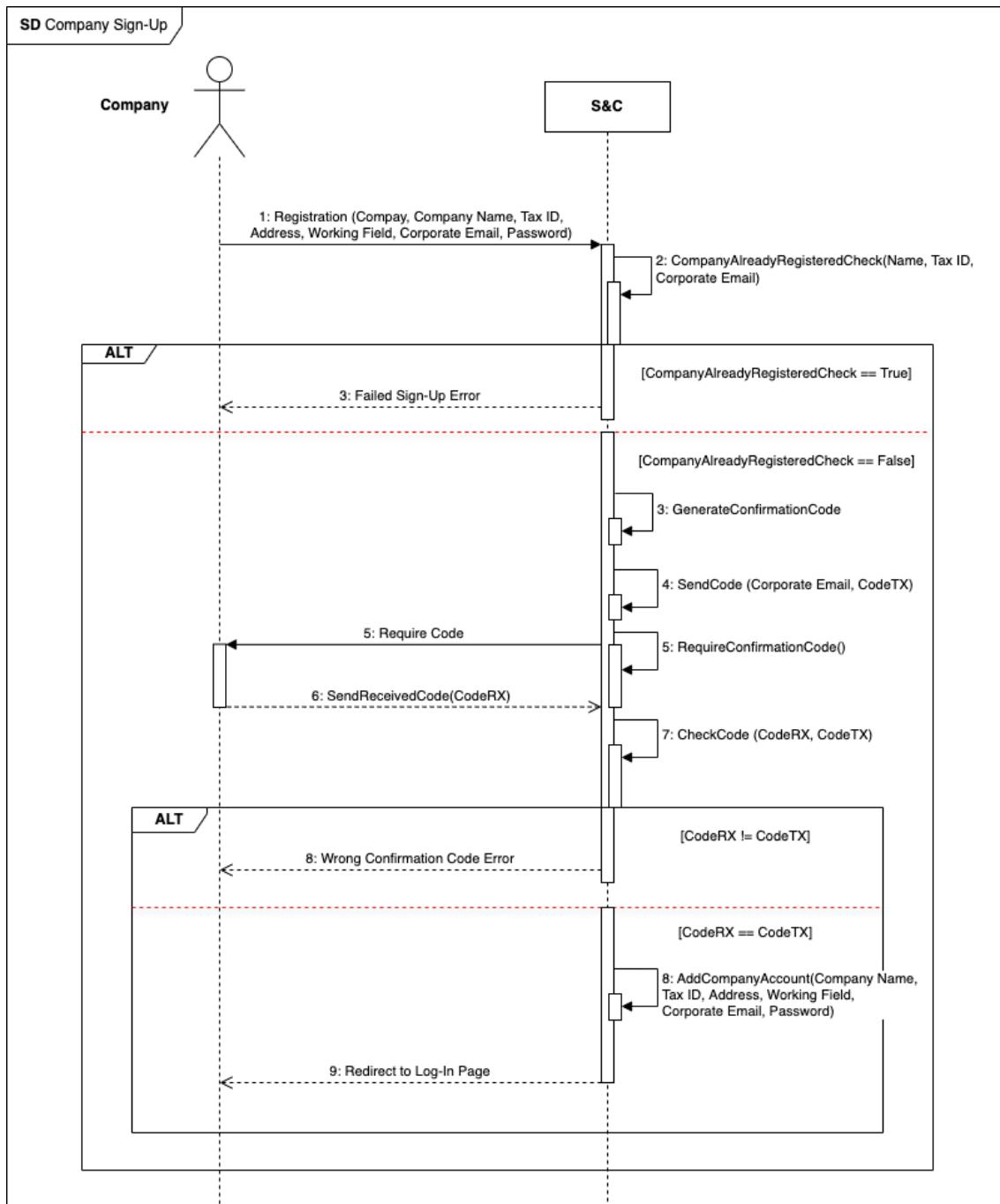


Figure 3.12: Company Registration Sequence Diagram

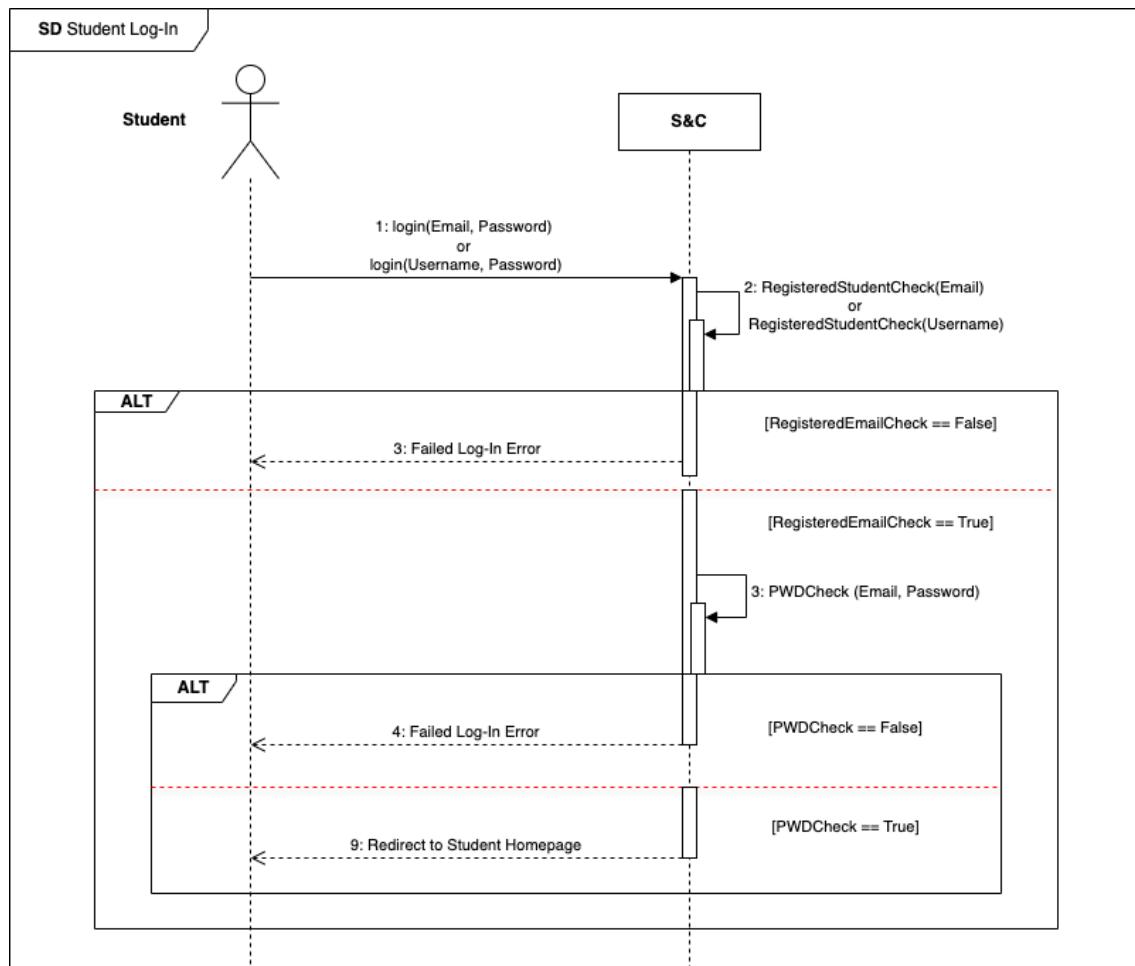


Figure 3.13: Student Log-In Sequence Diagram

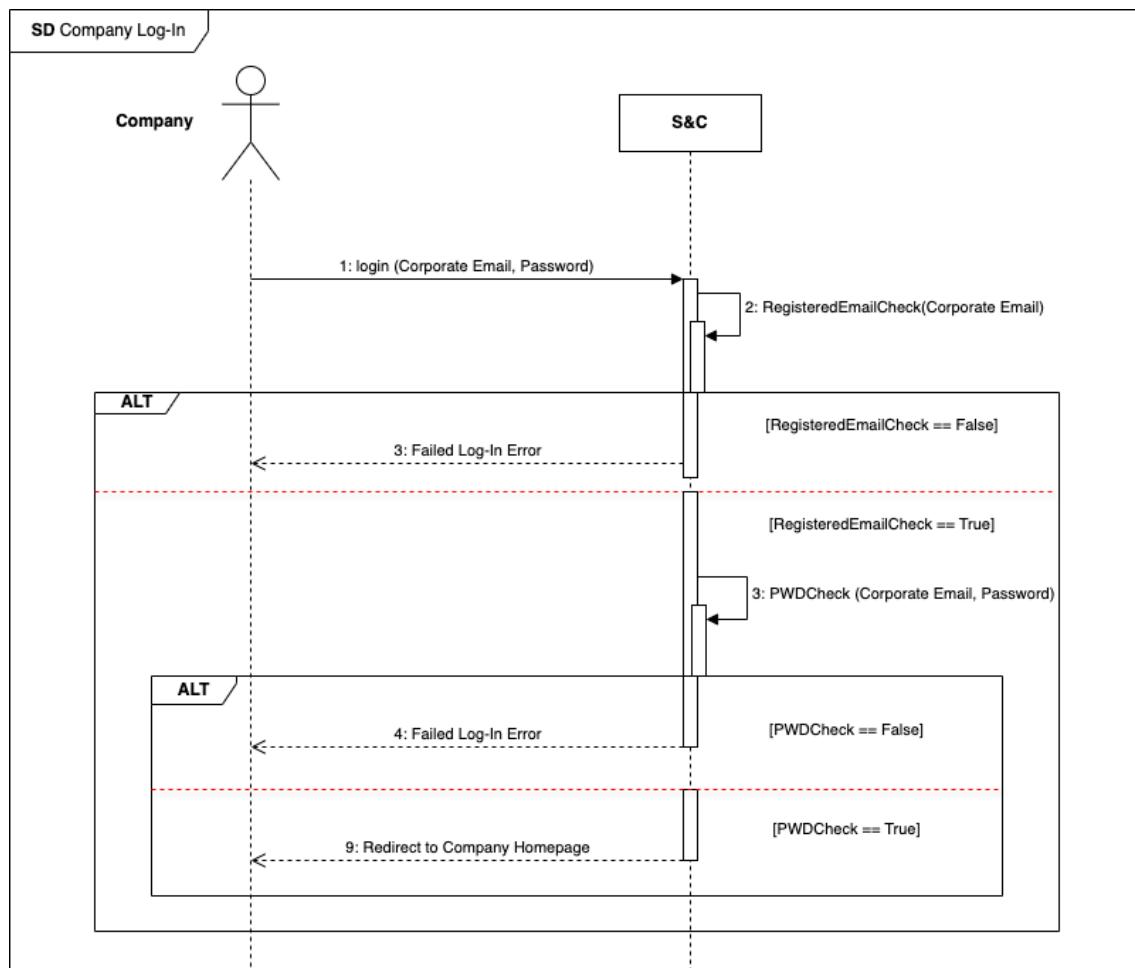


Figure 3.14: Company Log-In Sequence Diagram

### Student Operations as main interactions

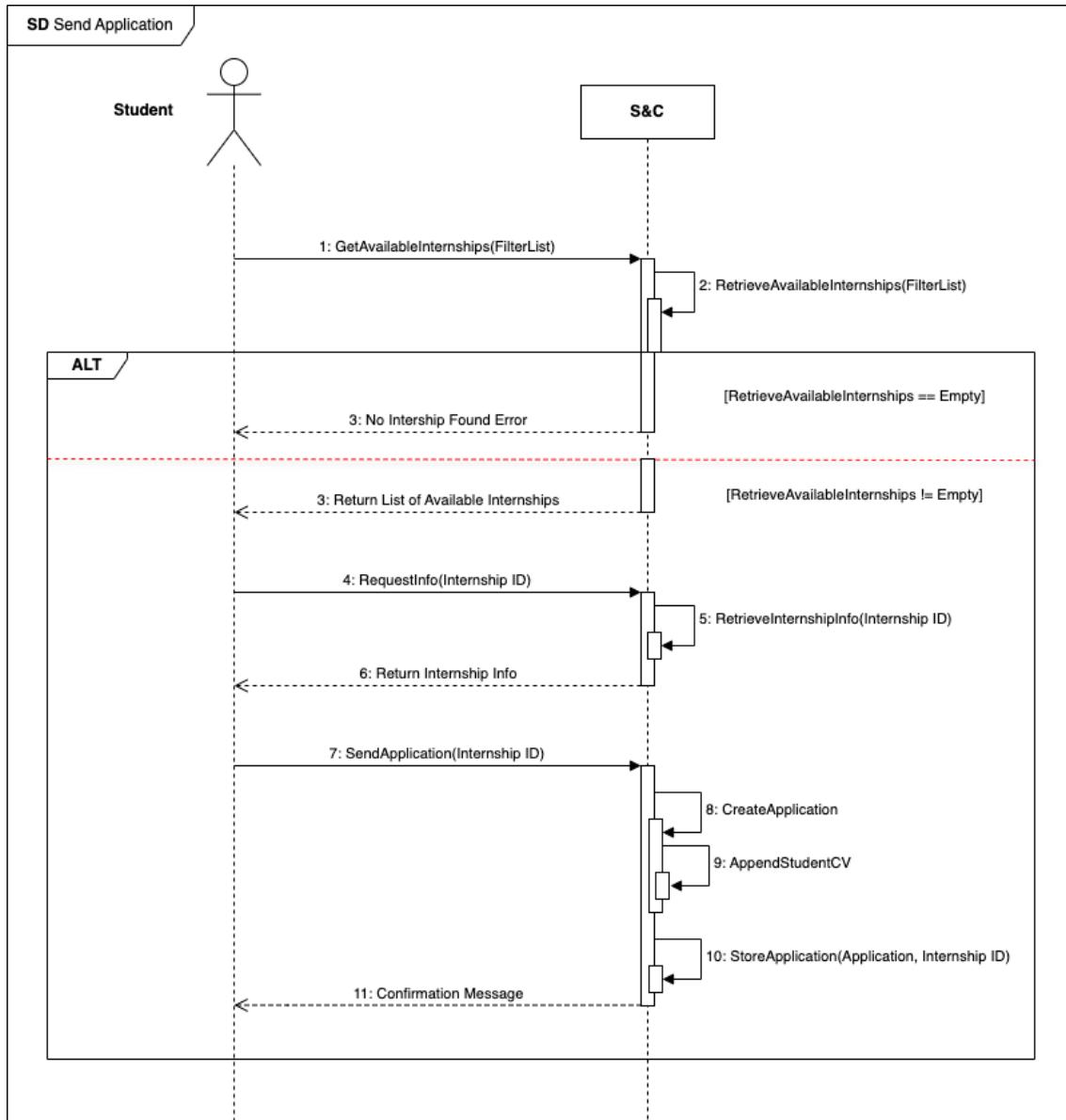


Figure 3.15: Send Application Sequence Diagram

CHAPTER 3. SPECIFIC REQUIREMENTS

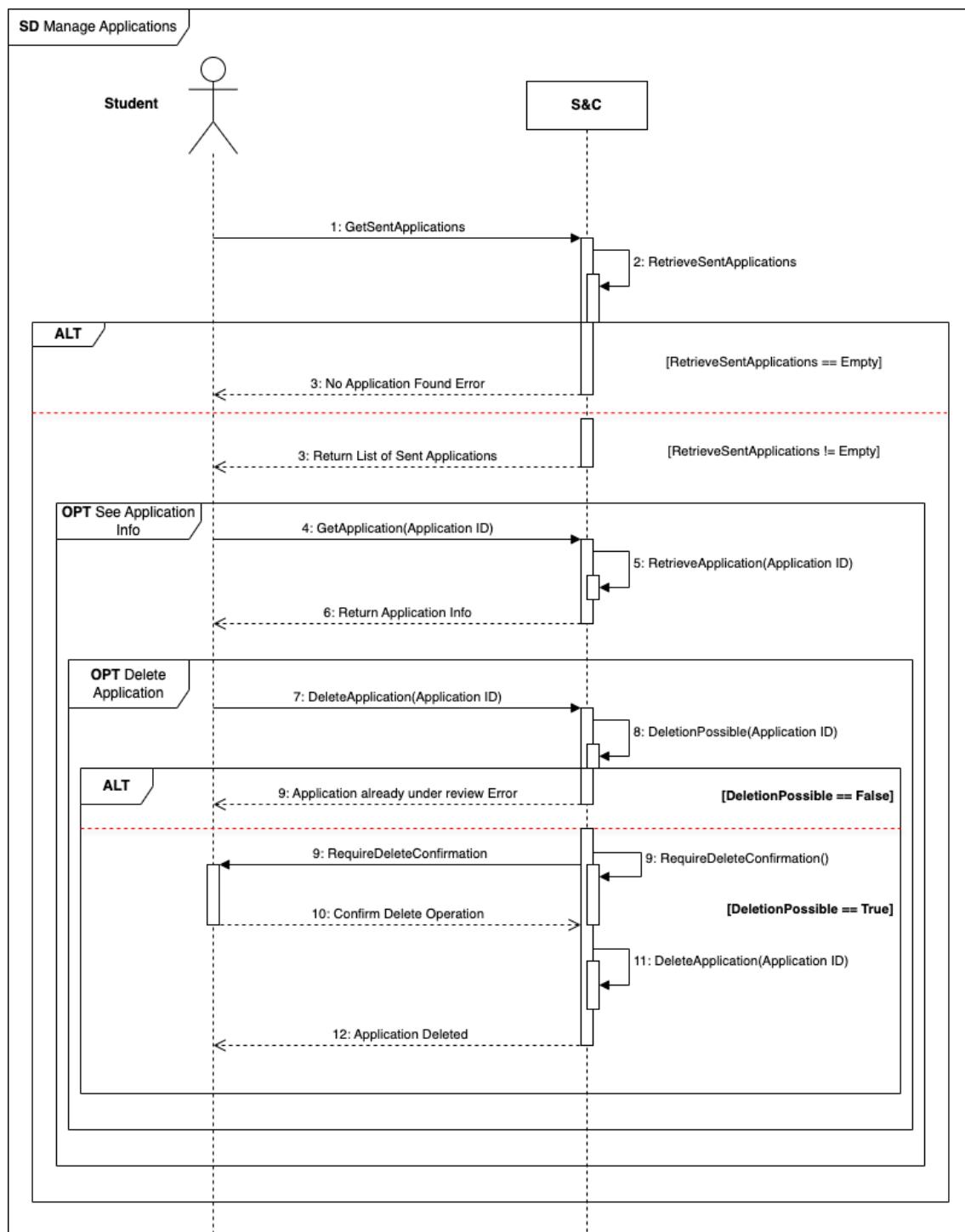


Figure 3.16: Manage Application Sequence Diagram

CHAPTER 3. SPECIFIC REQUIREMENTS

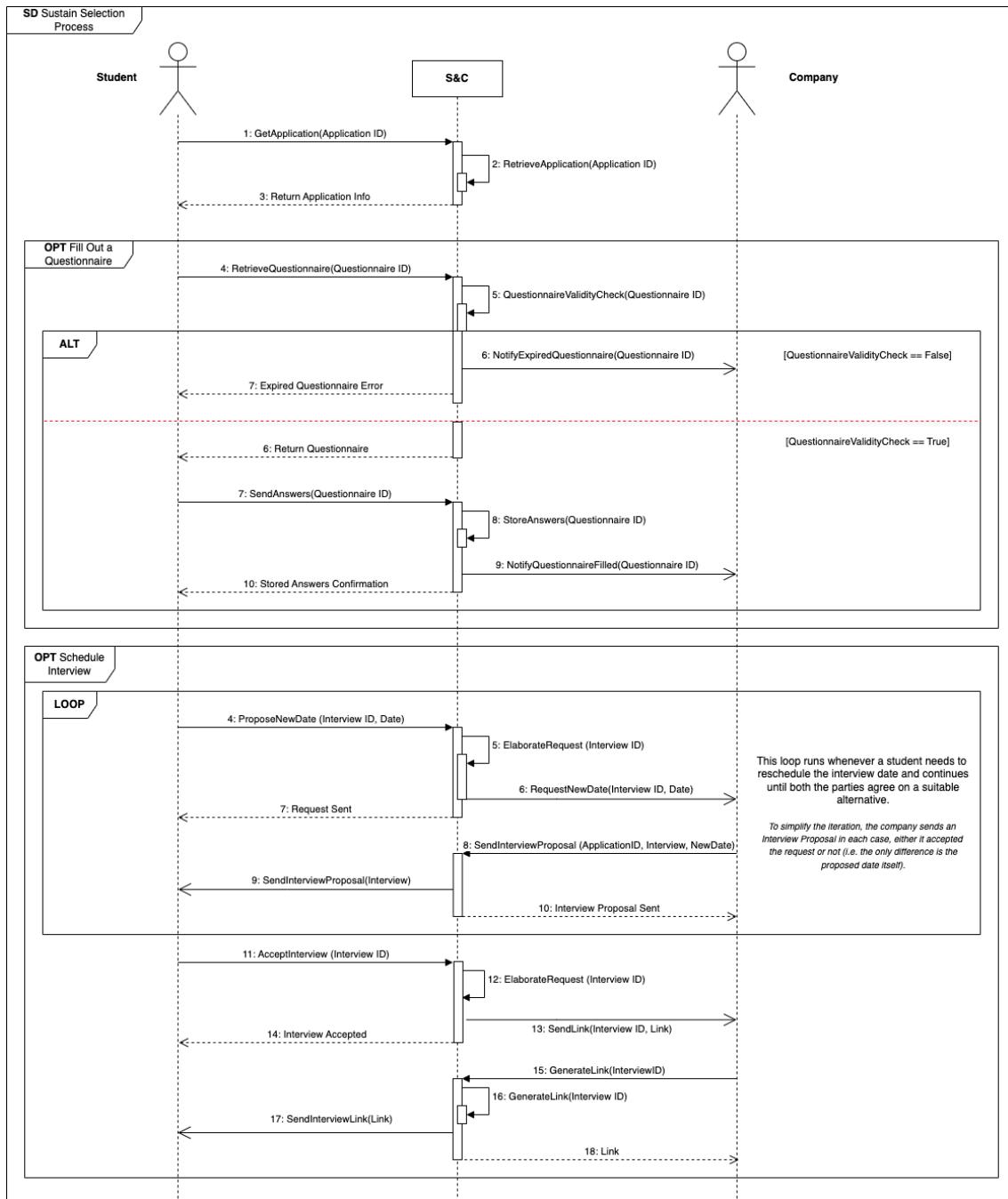


Figure 3.17: Sustain Selection Process Sequence Diagram

## CHAPTER 3. SPECIFIC REQUIREMENTS

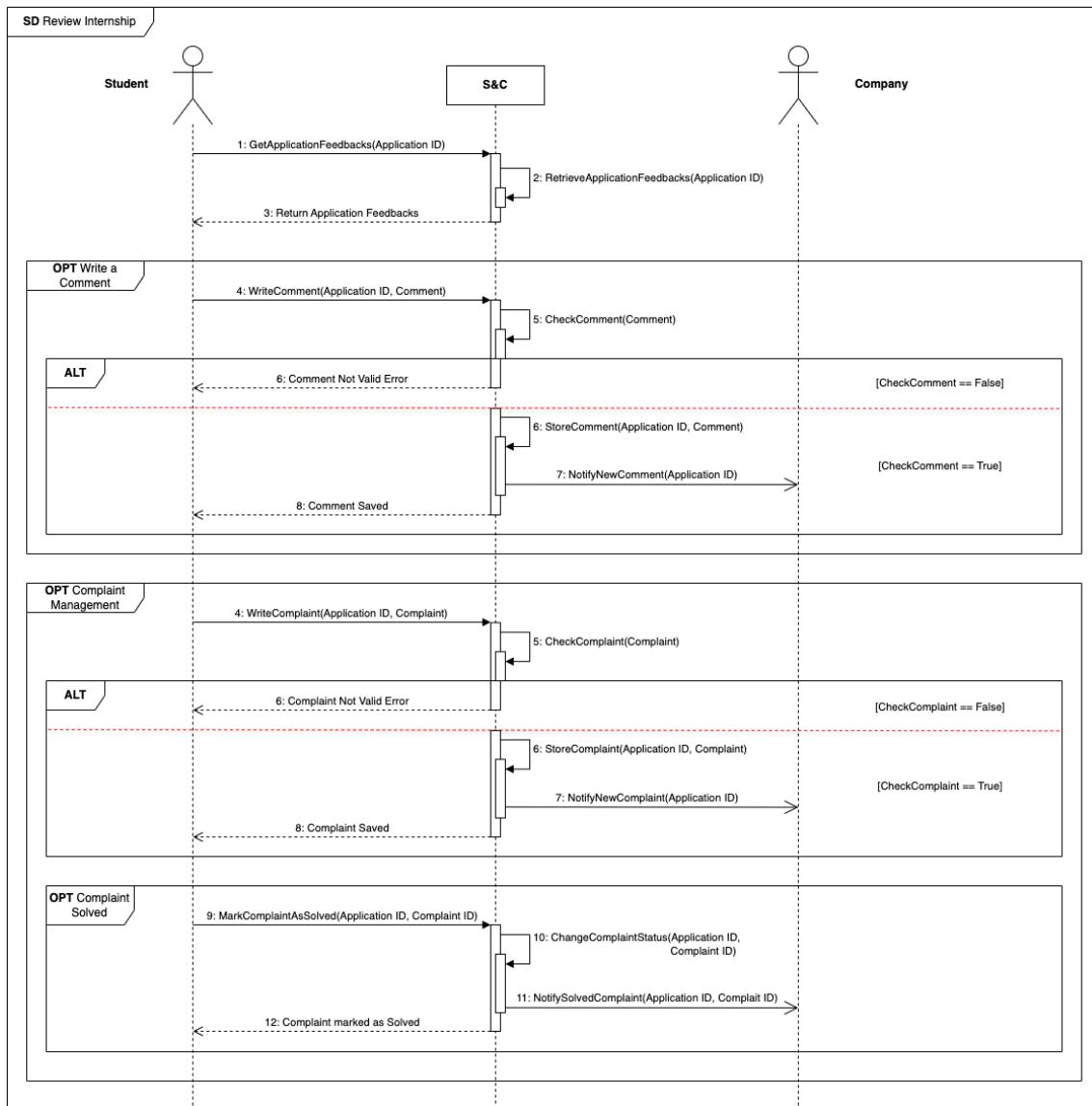


Figure 3.18: Review Internship Sequence Diagram

### Company Operations as main interactions

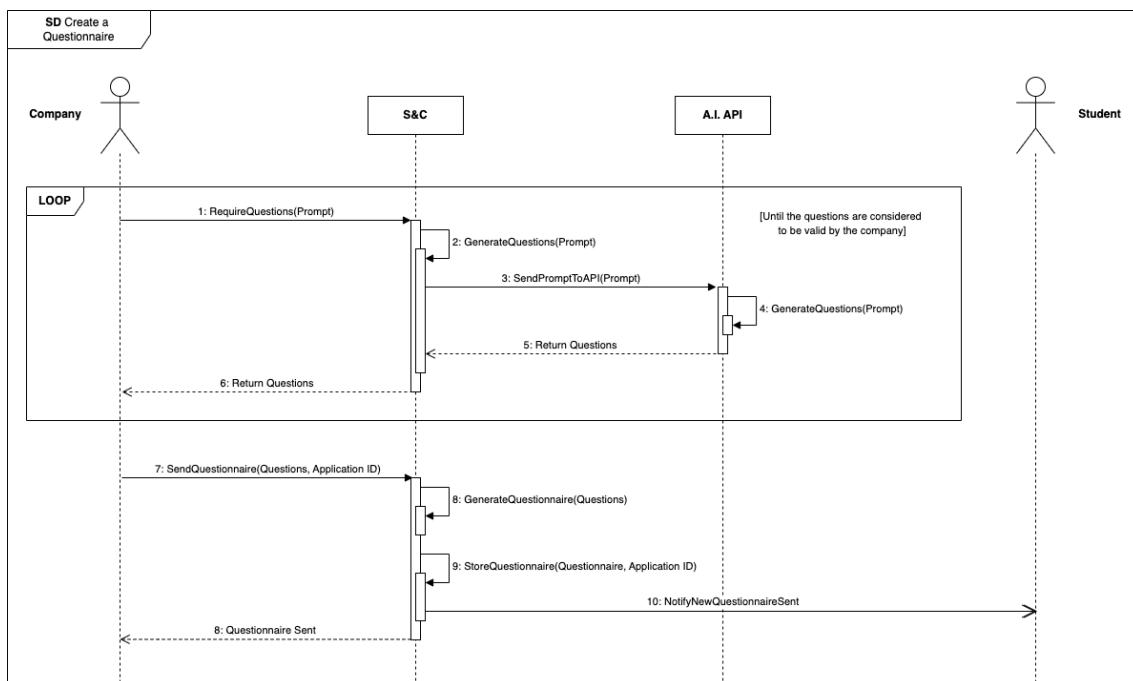


Figure 3.19: Create Questionnaire Sequence Diagram

## CHAPTER 3. SPECIFIC REQUIREMENTS

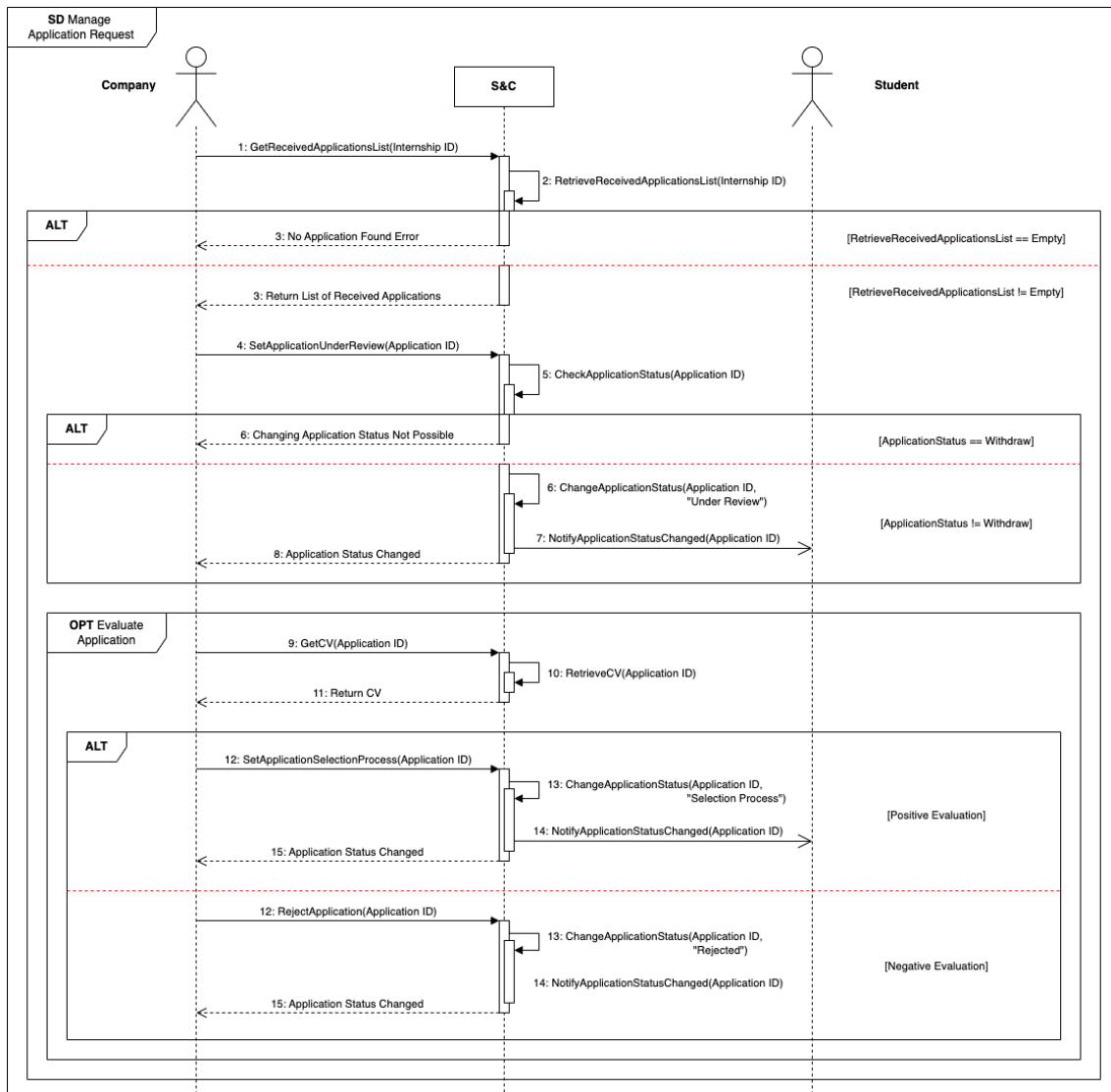


Figure 3.20: Manage Internship Application Sequence Diagram

### **3.2.4 Requirements**

The table below provides a detailed overview of the functional requirements identified for the S&C system:

<b>Requirements</b>	<b>Description</b>
R.1	The system must allow the student to register into the system by providing all the mandatory information (i.e. Name, Surname, Student ID, etc...)
R.2	The system must allow the company to register into the system by providing all the mandatory information (i.e. Name, Tax ID, Working Field, etc...)
R.3	During the registration process, the system must validate the uniqueness of the Username and the Email provided by the student who is registering into the system
R.4	During the registration process, the system must validate the uniqueness of the Name, Tax ID and the Corporate Email provided by the company who is registering into the system
R.5	Once the student has entered his registration details, the system must send a verification code to the provided email address
R.6	Once the company has entered his registration details, the system must send a verification code to the provided corporate email address
R.7	The system must allow the student to input the verification code sent to his email address
R.8	The system must allow the company to input the verification code sent to its corporate email address
R.9	When a verification code is given in input by the user (either student or company), the system must verify whether the provided code matches the one sent to the user's email
R.10	The system must allow the student to log into his account by entering his Username/Email and Password
R.11	The system must allow the company to log into its account by entering its Corporate Email and Password
R.12	Once the log-in information has been validated, the system must redirect the user (either student or company) to his corresponding Profile Page
R.13	The system must allow the student to search for available internship proposals
R.14	When searching for internship proposals, the system must allow the student to apply specific filters to refine and limit the search results
R.15	The system must allow the student to send an application for the chosen internship proposals
R.16	When the student submits an application, the system must attach the student's CV to the application itself
R.17	When a new application is submitted, the system must set the application's status to "Sent"
R.18	The system must allow the student to view the list of the sent applications

R.19	The system must allow the student to view detailed information about a specific application (i.e. submission date, internship proposal details, application status, etc...)
R.20	The system must allow the student to delete his application for an internship proposal
R.21	When the student attempts to delete an application, the system must prevent this action if the application status has been set to "Under Review" by the company
R.22	If the user is allowed to delete the application, the system must update the application's status to "Withdrawn"
R.23	The system must allow the company to update the application's status in the following sequence: Sent → Under Review/Rejected → Selection Process → Internship/Rejected → Internship Completed
R.24	The system must allow the student to upload his C.V.
R.25	The system must allow the student to upload a C.V. only if no CV has been previously uploaded or if the student explicitly agrees to replace the existing CV
R.26	The system must allow the company to publish a new internship proposal
R.27	The system must allow the company to view the list of the published internship proposal
R.28	The system must allow the company to view the list of applications submitted for a published internship proposal
R.29	The system must allow the company to view the details (e.g. student's CV, submission date, etc...) of a submitted application
R.30	When the company attempts to change the application's status, the system must prevent this action if the status is set to "Withdrawn"
R.31	When the application's status is updated, the system must notify the associated student via email
R.32	When the company attempts to change the application's status from "Selection Process," the system must prevent this action unless at least one interview or one questionnaire has been completed
R.33	The system must allow the company to visualize the list of sent questionnaire for a submitted application
R.34	The system must allow the company to provide a prompt to an AI tool to generate a list of questions to include in the questionnaire
R.35	The system must allow the company to view the questions generated by the A.I. Tool
R.36	The system must allow the company to manage (i.e. modify, add and delete) the list of questions generated by the A.I. Tool
R.37	The system must allow the company to create a questionnaire by submitting a list of questions
R.38	When the company submits a list of questions, the system must automatically generate a structured questionnaire based on the provided questions
R.39	After the structured questionnaire is generated, the system must send it to the associated student

R.40	The system must allow the company to view the student's answers associated to a sent questionnaire
R.41	When the student receives a questionnaire, the system must notify him via email
R.42	The system must allow the company to view the list of scheduled interviews of a submitted application
R.43	The system must allow the company to send an Interview Proposal in which it is specified a potential date
R.44	When the student receives an Interview Proposal, the system must notify him via email
R.45	The system must allow the student to manage (i.e. accept or propose a new date) an interview proposal
R.46	The system must allow the company to generate the Interview Link after the student has accepted the Interview Proposal
R.47	After the Interview Link has been generated, the system must send a mail notification to the associated student
R.48	The system must allow the company to send a Questionnaire/Interview Proposal only if the application's status has been set to "Selection Process"
R.49	When the application's status is "Internship", the system must allow both the associated company and the associated student to submit feedback (i.e. comments or complaints) about the ongoing experience
R.50	The system must allow both the student and the company involved in an internship to view the feedbacks (i.e. comments or complaints) made by both of them
R.51	When the user, either the student or the company, submits a complaint, the system must notify the counterpart of the new submission
R.52	When the user, either the student or the company, submits a comment, the system must notify the counterpart of the new submission
R.53	Once the user, either the student or the company, has submitted a complaint, the system must allow them to mark it as "Solved" if the issue has been addressed by the counterpart
R.54	When the student receives a questionnaire, the system must allow the student to complete and submit the questionnaire

### **3.2.5 Mapping on Requirements**

Goals essentially are logical consequences of the conjunction of the requirements and the domain assumptions and, therefore, to ensure that each goal is met (i.e.  $R \text{ and } D \models G$ ), we always validate it against the corresponding requirements and domain assumptions.

The following table provides a detailed mapping of these relationships:

Goals	Domain Assumptions	Requirements
G1	D1 - D3 - D5 - D9 - D13	R1 - R3 - R5 - R7 - R9 - R10 R12 - R13 - R14
G2	D1 - D2 - D10 - D11 - D13 - D16	R1 - R3 - R5 - R7 - R9 - R10 R12 - R15 - R16 - R17 - R18 - R19 R20 - R21 - R22 - R23 - R24 - R25 R31
G3	D1 - D3 - D11 - D14 - D15	R1 - R3 - R5 - R7 - R9 - R10 R12 - R31 - R41 - R44 - R45 - R46 R47 - R53
G4	D1 - D4 - D11 - D12 - D16	R2 - R4 - R6 - R8 - R9 - R11 R12 - R23 - R26 - R27 - R28 - R29 R30
G5	D1 - D4 - D14 - D15	R2 - R4 - R6 - R8 - R9 - R11 R12 - R23 - R32 - R33 - R34 - R35 R36 - R37 - R38 - R39 - R40 - R42 R43 - R46 - R47 - R48
G6	D1 - D3 - D4 - D6 - D7 - D8	R1 - R2 - R3 - R4 - R5 - R6 R7 - R8 - R9 - R10 - R11 - R12 R49 - R50 - R51 - R52

Table 3.17: Requirements, Domain Assumptions and Goals Mapping

*It is worth underlining that certain requirements and domain assumptions may be repeated across different goals: this is perfectly reasonable, particularly when they represent fundamental elements (e.g. login requirements, which basically are always necessary)*

### 3.3 Performance Requirements

The system must meet the following performance requirements to ensure an efficient behaviour:

- **Response Time:** the system shall respond to user interactions (e.g. button clicks, page loads) within 2 seconds under normal traffic conditions
- **Query Processing:** all the queries executed on the platform (e.g. searching for internships) shall return results within 3 seconds under normal traffic conditions
- **Notification:** notifications triggered by system events (e.g. new internship applications, application's status updates) shall be sent to users (both companies and students) within 1 minute of the event occurrence
- **Concurrent User Handling:** the system shall support up to 2,000 concurrent users (a mix of companies and students) during its initial deployment phase without any significant performance degradation

- **Scalability:** the system shall adopt an horizontal scalability strategy, enabling the dynamic addition of resources (e.g. add servers) to maintain peak performance

## **3.4 Design Constraints**

In this section, we analyze all relevant constraints related to the design considerations.

### **3.4.1 Standard Compliance**

The S&C System shall follow the General Data Protection Regulation (i.e. **G.D.P.R.**): it is a key regulation in E.U. law that governs data protection and privacy for all the individuals within the European Union and the European Economic Area (EEA).

As a G.D.P.R.-Compliant system, S&C shall ensure robust measures to safeguard personal data and supporting the user's privacy, meeting the stringent requirements of this legal framework.

Additionally, the system shall adopt the ISO-8601 Standard: it is an internationally recognized standards for date and time formatting that will ensure consistency and interoperability across the various geographical regions.

### **3.4.2 Hardware Limitations**

In order to ensure the proper functionality of the platform, users must have access to an internet connection (e.g. LTE, 3G, 4G, 5G or Wi-Fi) and a compatible device capable of running a web browser supporting HTML 5.

Another critical hardware constraint is represented by the server-side overloading: in the event of high demand, the system must address this issue by adhering to the scalability requirements outlined in the previous section.

## **3.5 Software System Attributes**

This section provides a detailed analysis of the software system attributes.

### **3.5.1 Reliability**

The system shall operate continuously without interruptions over extended periods and, in order to guarantee this requirement, a fault tolerance approach must be used: the backend deployment must incorporate replication and redundancy mechanisms.

Additionally, the system shall maintain offline backups of data storage, enabling effective disaster recovery in the event of data loss.

### **3.5.2 Availability**

The platform must be accessible at all times to accommodate users from different time zones, including international students and global companies.

Furthermore, the S&C system shall ensure an Up-Time of 99.9%, equivalent to an average Down-Time of approximately 8.76 hours per year.

### **3.5.3 Security**

All the data and the information transferred or stored by the system shall be protected using robust encryption methods, such as TLS 1.3 protocol for secure transmission and SHA-256 hashing for data integrity and security.

### **3.5.4 Maintainability**

The system shall ensure an high level of maintainability by employing appropriate design patterns and following established industry standards.

Furthermore, maintenance activities shall be scheduled during periods of low activity (e.g. during the night).

### **3.5.5 Portability**

The platform shall be accessible via web browsers on both desktop and mobile devices without requiring any additional element.

## *Chapter 4*

# Formal Analysis Using Alloy

In this section, the system-to-be has been modeled using a formal modeling language, Alloy, in order to produce a precise specification and analysis of the system properties.

## 4.1 Signatures

All the relevant signatures for our problem are defined and presented below.

```
1  sig Date {}
2  var sig Time {}
3
4  var sig Link {}
5  var sig Questionnaire {}
6
7  sig CV {}
8
9  sig Role {}
10
11 enum ComplaintStatus {Opened, Solved}
12 sig InternshipDescription {}
13 sig ComplaintDescription {}
14 sig CommentDescription {}
15
16 sig Name {}
17 sig Surname {}
18 sig University {}
19 sig StudentID {}
20 sig Industry {}
21 sig Address {}
22 sig Username {}
23 sig Email {}
```

```
24 sig Password {}
25 sig WorkingField {}
26 sig TaxID {}

27
28 -- Certain status names cannot be used because equivalent entities or
29 -- parameters already exist within the model, leading to conflicts
30 -- (i.e. SelectionProcess, Internship)
31 enum ApplicationStatus {Sent, UnderReview, SelectionP,
32                           InternshipExp, Rejected, WithDrawn}

33
34 sig Wage {}
35 enum Paid {Yes, No}

36
37 var sig Interview {
38
39     var date: one Date,
40     var time: one Time,
41     var link: one Link
42
43 }
44
45 var sig SelectionProcess {
46
47     var startingDate: one Date,
48     var endingDate: one Date,
49     var questionnaires: set Questionnaire,
50     var Interviews: set Interview
51
52 } {
53
54     startingDate != endingDate
55
56 }
57
58 abstract sig FeedBack {
59
60     submitDate: one Date
61
62 }
```

```
63
64 var sig Comment extends FeedBack {
65
66     var description: one CommentDescription
67
68 }
69
70 var sig Complaint extends FeedBack {
71
72     var status: one ComplaintStatus,
73     var complaint: one ComplaintDescription,
74     var resolutionDate: lone Date
75
76 } {
77
78     submitDate != resolutionDate
79
80 }
81
82 abstract sig User {
83
84     name: one Name,
85     email: one Email,
86     password: one Password,
87     var submitComments: set Comment,
88     var submitComplaints: set Complaint
89
90 }
91
92 sig Student extends User {
93
94     surname: one Surname,
95     university: one University,
96     studentID: one StudentID,
97     username: one Username,
98     hasCV: one CV,
99     createApplication: set Application
100
101 }
```

```
102
103 sig Company extends User {
104
105     taxID: one TaxID,
106     workingField: one WorkingField,
107     postInternship: set Internship
108
109 }
110
111 sig Application {
112
113     var status: one ApplicationStatus,
114     submitDate: one Date,
115     sentToIntership: one Internship,
116     var associatedSelectionProcess: lone SelectionProcess,
117     var feedBack: set FeedBack
118
119 }
120
121 sig Internship {
122
123     description: one InternshipDescription,
124     role: one Role,
125     startingDate: one Date,
126     endingDate: one Date,
127     paid: one Paid,
128     wage: lone Wage,
129
130 } {
131
132     startingDate != endingDate
133
134 }
```

## 4.2 Facts, Predicate and Assertions

All the relevant facts, predicate and assertions defined for our problem are presented below.

### Facts

```
1 fact UserAttributesConstraints {
2
3     -- A name can exist only if associated to a user
4     always (all n: Name | some u: User | u.name = n)
5
6     -- An email can be associated to a unique user
7     always (all e: Email | one u: User | u.email = e)
8
9     -- A password can exist only if associated to a user
10    always (all p: Password | some u: User | u.password = p)
11
12 }
13
14 fact StudentAttributesConstraints {
15
16     -- A surname can exist only if associated to some student
17     always (all s: Surname | some stud: Student | stud.surname = s)
18
19     -- A university can exist only if associated to some student
20     always (all u: University | some s: Student | s.university = u)
21
22     -- A student ID can exist only if associated to some student
23     always (all sID: StudentID | some s: Student | s.studentID = sID)
24
25     -- Two students can have the same student ID only in the case
26     -- in which they belong to different universities
27     always (all disj s1, s2: Student | s1.studentID = s2.studentID
28             implies
29             s1.university != s2.university)
30
31     -- A username must be associated to a unique student
32     always (all u: Username | one s: Student | s.username = u)
33 }
```

```
34      -- Each CV must be associated to a unique student
35      always (all cv: CV | one s: Student | s.hasCV = cv)
36
37      -- An application must be associated to a unique student
38      always (all a: Application | one s: Student | a in s.createApplication)
39
40  }
41
42
43 fact CompanyAttributesConstraints {
44
45      -- Companies can not have the same taxID
46      always (all tx: TaxID | one c: Company | c.taxID = tx)
47
48      -- A working field can exists only if associated to a company
49      always (all wf: WorkingField | some c: Company | c.workingField = wf)
50
51      -- An internship can not belong to multiple companies
52      always (all i: Internship | one c: Company | i in c.postInternship)
53
54  }
55
56
57 fact ApplicationAttributesConstraints {
58
59      -- A selection process can be associated to a unique
60      -- application
61      always (all sp: SelectionProcess | one a: Application |
62                  a.associatedSelectionProcess = sp)
63
64  }
65
66
67 fact InternshipAttributesConstraints {
68
69      -- An Internship Description can exist only if associated to
70      -- an Internship
71      always (all d: InternshipDescription | some i: Internship |
72                  i.description = d)
```

```
73
74      -- A role can exist only if associated to an Internship
75      always (all r: Role | some i: Internship | i.role = r)
76
77      -- A wage can exist only if associated to an Internship
78      always (all w: Wage | some i: Internship | i.wage = w)
79
80      -- If the internship is paid, the wage must be specified
81      always (all i: Internship | i.paid = Yes
82                  iff
83                  i.wage != none)
84
85  }
86
87 fact InterviewAttributesConstraints {
88
89      -- A link can be associated to a unique interview
90      always (all l: Link | one i: Interview | i.link = l)
91
92      always (all t: Time | some i: Interview | i.time = t)
93
94  }
95
96
97 fact SelectionProcessAttributesConstraints {
98
99      -- A questionnaire can be associated to a unique
100     -- selection process
101     always (all q: Questionnaire | one sp: SelectionProcess |
102                         q in sp.questionnaires)
103
104     -- An interview can be associated to a unique selection process
105     always (all i: Interview | one sp: SelectionProcess |
106                         i in sp.Interviews)
107
108  }
109
110
```

```
112 fact FeedBackConstraints {  
113  
114     -- A feedback must be associated to an application  
115     always (all f: FeedBack | one a: Application | f in a.feedBack)  
116  
117     -- Any kind of feedback can be done only if the application has  
118     -- entered in the internship status  
119     always (all a: Application | all f: FeedBack | f in a.feedBack  
120                         implies  
121                         a.status = InternshipExp)  
122  
123 }  
124  
125 fact CommentAttributesConstraints {  
126  
127     -- A Comment Description can exist only if associated  
128     -- to a comment  
129     always (all d: CommentDescription | some c: Comment | c.description = d)  
130  
131     -- A comment can exists only if associated to a user  
132     always (all c: Comment | one u: User | c in u.submitComments)  
133  
134     -- A comment can be done only by a company or a student  
135     -- involved in that application  
136     always (((all c: Comment | one s: Student | c in s.submitComments  
137                         implies  
138                         s = (c.^feedBack).^createApplication)  
139  
140             or  
141  
142             ((all c: Comment | one co: Company | c in co.submitComments  
143                         implies  
144                         co = ((c.^feedBack).sentToIntership).^postInternship)))  
145  
146 }  
147  
148 fact ComplaintAttributesConstraints {  
149  
150     -- A complaint can exists only if associated to a user
```

```
151     always (all c: Complaint | one u: User | c in u.submitComplaints)
152
153     -- A complaint Description can exist only if associated to
154     -- a complaint
155     always (all d: ComplaintDescription | some c: Complaint |
156                         c.complaint = d)
157
158     -- A complaint has a resolution date only it has been solved
159     always (all c: Complaint | c.resolutionDate != none
160                         iff
161                         c.status = Solved)
162
163     -- A complaint can be done only by a company or a student
164     -- involved in that application
165     always (((all c: Complaint | one s: Student | c in s.submitComplaints
166                         implies
167                         s = (c.^feedBack).^createApplication)
168
169             or
170
171             ((all c: Complaint | one co: Company | c in co.submitComplaints
172                         implies
173                         co = ((c.^feedBack).sentToIntership).^postInternship)))
174
175 }
176
177     -- The company can not have an individual name
178 fact NoSharedName {
179
180     always (all c: Company | no s: Student | c.name = s.name)
181
182 }
183
184     -- A Selection Process will be done only if the application has been
185     -- set to the related status
186 fact SelectionProcessOnlyDuringThatProcess {
187
188     always (all a: Application | a.associatedSelectionProcess != none
189             iff
```

```
190     (a.status = SelectionP or a.status = InternshipExp))
191 }
192
193 fact ComplaintStatus {
194
195     -- All the complaints start with status "Opened" and, after
196     -- some time, will be eventually solved
197     all c: Complaint | c.status = Opened
198         and
199         eventually MarkAComplaintAsSolved[c]
200
201
202     -- Once a complaint has been solved, it can not be open anymore
203     always (all c: Complaint | c.status = Solved
204             implies
205             c.status' = Solved)
206 }
207
208
209 fact ApplicationStatusEvolution {
210
211     -- All the applications start from status "Sent".
212     -- After status "Sent", an application can assume only one of
213     -- the following statuses: "UnderReview" or "WithDrawn"
214     all a: Application | a.status = Sent
215         and
216         eventually (DeleteApplication[a] or AnalyzeApplication[a])
217
218     -- Once an application has been deleted, it can not be
219     -- analyzed anymore
220     always (all a: Application | a.status = WithDrawn
221             implies
222             a.status' = WithDrawn)
223
224     -- Once an application is under review, it will not change
225     -- the status if the company does not perform any action
226     always (all a: Application | a.status = UnderReview
227             implies
228             (a.status' = UnderReview or StartSelectionProcess[a] or
```

```
229     RejectApplication[a]))  
230  
231     -- Once an application has been rejected, it can not be  
232     -- analyzed anymore  
233     always (all a: Application | a.status = Rejected  
234             implies  
235             a.status' = Rejected)  
236  
237     -- Once an application is on the Selection Process Phase, it  
238     -- will not change the status unless the company  
239     -- accepts/reject it  
240     always (all a: Application | a.status = SelectionP  
241             implies  
242             (a.status' = SelectionP or  
243              StartInternship[a] or  
244              RejectInternship[a]))  
245  
246     -- Once an application is on the Internship Phase, it will not  
247     -- change the status anymore  
248     always (all a: Application | a.status = InternshipExp  
249             implies  
250             (a.status' = InternshipExp))  
251  
252 }
```

## Predicates

```
1 pred MarkAComplaintAsSolved [c: Complaint] {  
2  
3     c.status = Opened  
4     and  
5     c.status' = Solved  
6  
7 }  
8  
9 pred DeleteApplication [a: Application] {
```

```
10
11      a.status = Sent
12      and
13      a.status' = WithDrawn
14
15  }
16
17 pred AnalyzeApplication [a: Application] {
18
19      a.status = Sent
20      and
21      a.status' = UnderReview
22
23  }
24
25 pred StartSelectionProcess [a: Application] {
26
27      a.status = UnderReview
28      and
29      a.status' = SelectionP
30
31  }
32
33
34 pred RejectApplication [a: Application] {
35
36      a.status = UnderReview
37      and
38      a.status' = Rejected
39
40  }
41
42 pred RejectInternship [a: Application] {
43
44      a.status = SelectionP
45      and
46      a.status' = Rejected
47
48  }
```

```
49
50 pred StartInternship [a: Application] {
51
52     a.status = SelectionP
53     and
54     #(a.associatedSelectionProcess.questionnaires +
55         a.associatedSelectionProcess.Interviews) > 0
56     and a.status' = InternshipExp
57
58 }
59
60 pred ShowComplaintStatusEvolution {
61
62     some disj c1, c2: Complaint | MarkAComplaintAsSolved[c1] and
63                         c2.status' = Opened
64
65 }
66
67 pred showApplicationStatusEvolution {
68
69     some disj a1, a2, a3, a4: Application|
70     DeleteApplication[a1] and
71     AnalyzeApplication[a2] and
72     AnalyzeApplication[a3] and
73     AnalyzeApplication[a4]
74     and eventually (StartSelectionProcess[a2] and
75                     RejectApplication[a3] and
76                     StartSelectionProcess[a4] and eventually
77                     (StartInternship[a2] and RejectInternship[a4]))
78
79 }
```

## Assertions

```
1 assert FeedbackWrittenByInvolvedUser {
2 }
```

```
3      all a: Application | all f: FeedBack | f in a.feedBack
4          implies
5              (some s: Student | s in a.^createApplication
6                  and
7                      f in s.submitComments + s.submitComplaints)
8          or
9              (some c: Company | c in a.sentToIntership.^postInternship
10                 and
11                     f in c.submitComments + c.submitComplaints)
12
13 }
14
15 assert NoComplaintSolvedBeforeOpened {
16
17     all c: Complaint |
18
19         after always (c.status = Solved implies before c.status = Opened)
20
21 }
22
23 assert ApplicationStatusFlow {
24
25     all a: Application |
26
27         -- If the application starts as "Sent", it must move to
28         -- "UnderReview" or "WithDrawn"
29         (a.status = Sent implies eventually (a.status in
30             UnderReview + WithDrawn))
31
32         and
33
34         -- If the application is "WithDrawn", it remains "WithDrawn"
35         (a.status = WithDrawn implies always (a.status = WithDrawn))
36
37         and
38
39         -- If the application reaches "UnderReview", it can move to
40         -- "Rejected" or "SelectionP"
41         (a.status = UnderReview implies eventually (a.status in
```

```
42     Rejected + SelectionP))  
43  
44     and  
45  
46     -- If the application reaches "SelectionP", it can move to  
47     -- "InternshipExp" or "Rejected"  
48     (a.status = SelectionP implies eventually (a.status in  
49     InternshipExp + Rejected))  
50  
51     and  
52  
53     -- If the application reaches "InternshipExp", it remains  
54     -- in "InternshipExp"  
55     (a.status = InternshipExp implies always  
56     (a.status = InternshipExp))  
57  
58     and  
59  
60     -- If the application reaches "Rejected", it remains  
61     -- in "Rejected"  
62     (a.status = Rejected implies always (a.status = Rejected))  
63 }
```

### 4.3 Worlds

The world generated through the above Alloy modeling is presented in this section. The images represents the progression of an application's status through the several stages:

- **Initial Submission** (Figure 4.1): four applications are created and set to the "Sent" state
- **Withdrawal and Review** (Figure 4.2): one application is withdrawn by the user (changing it to the "Withdrawn" state), while the remaining three move to the "Under Review" state
- **Selection and Rejection** (Figure 4.3): of the three applications under review, one is rejected and the others advance to the "Selection Process"
- **Final Approval** (Figure 4.4): ultimately, only one application progresses to the "Internship Approval" stage

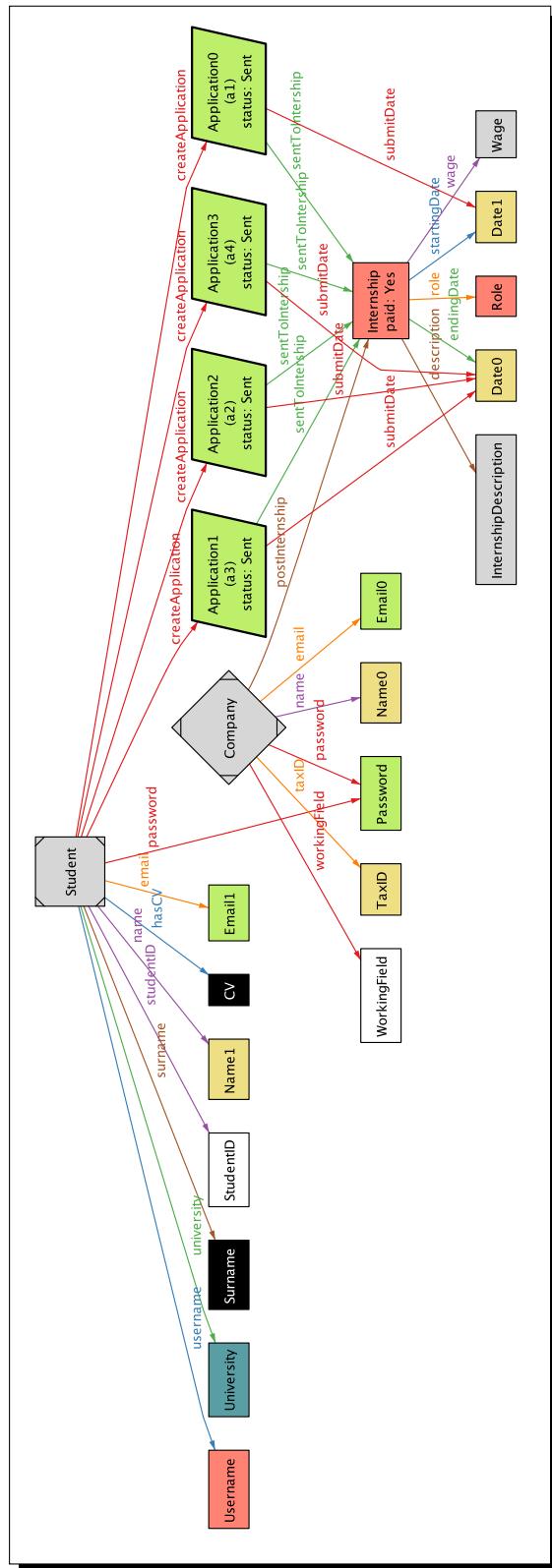


Figure 4.1: Alloy Wold - Sent Applications

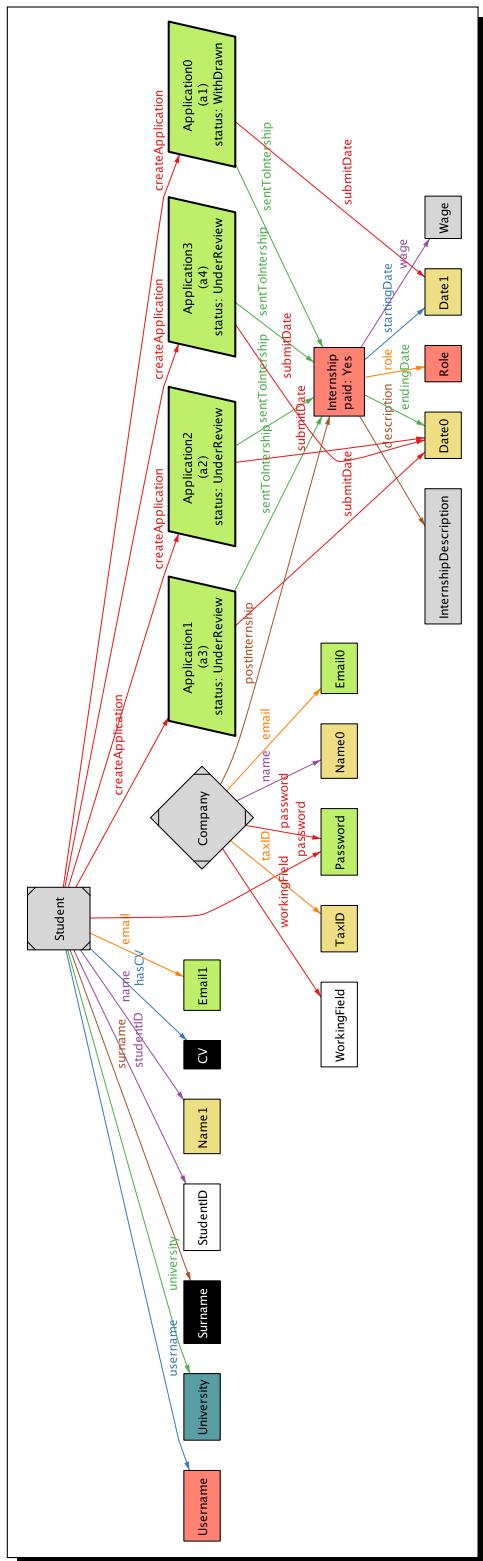


Figure 4.2: Alloy Wold - Under Review and WithDrawn Applications

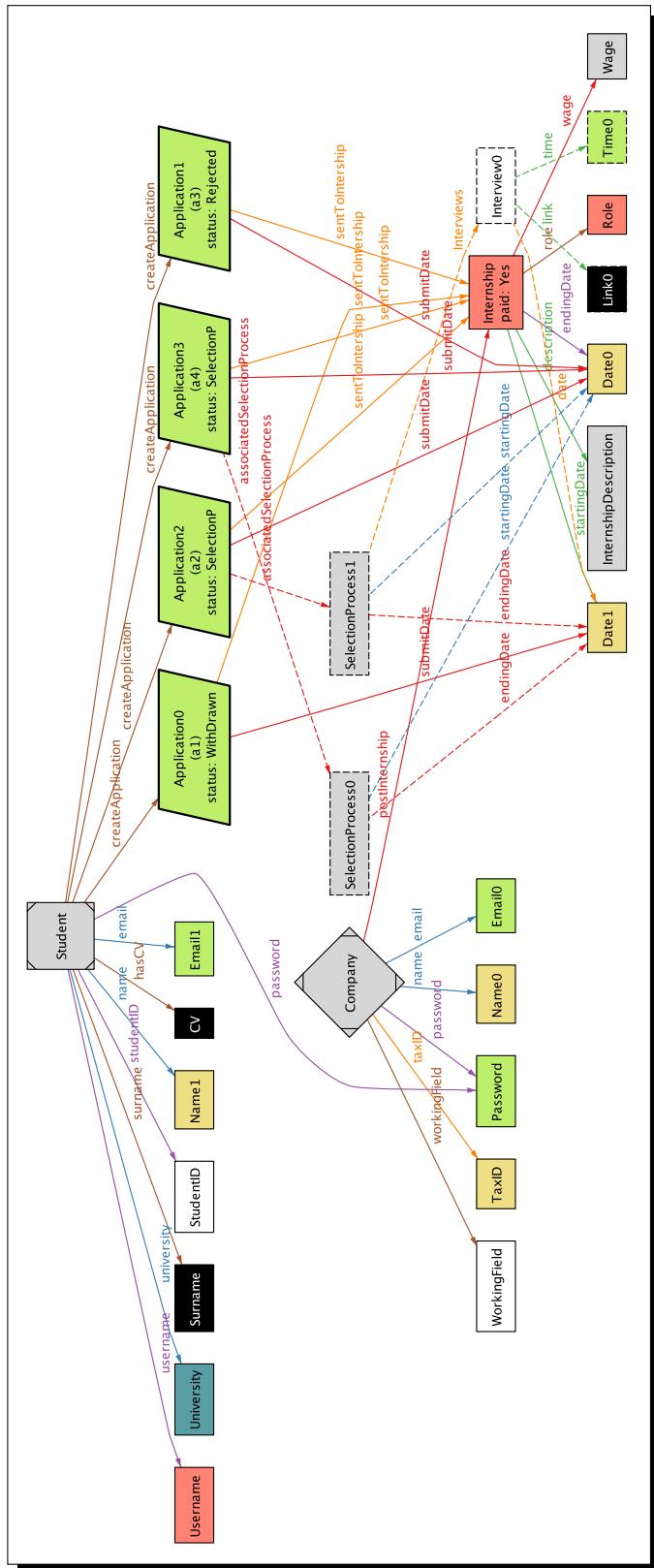


Figure 4.3: Alloy Wold - Rejected and Selection Process Applications

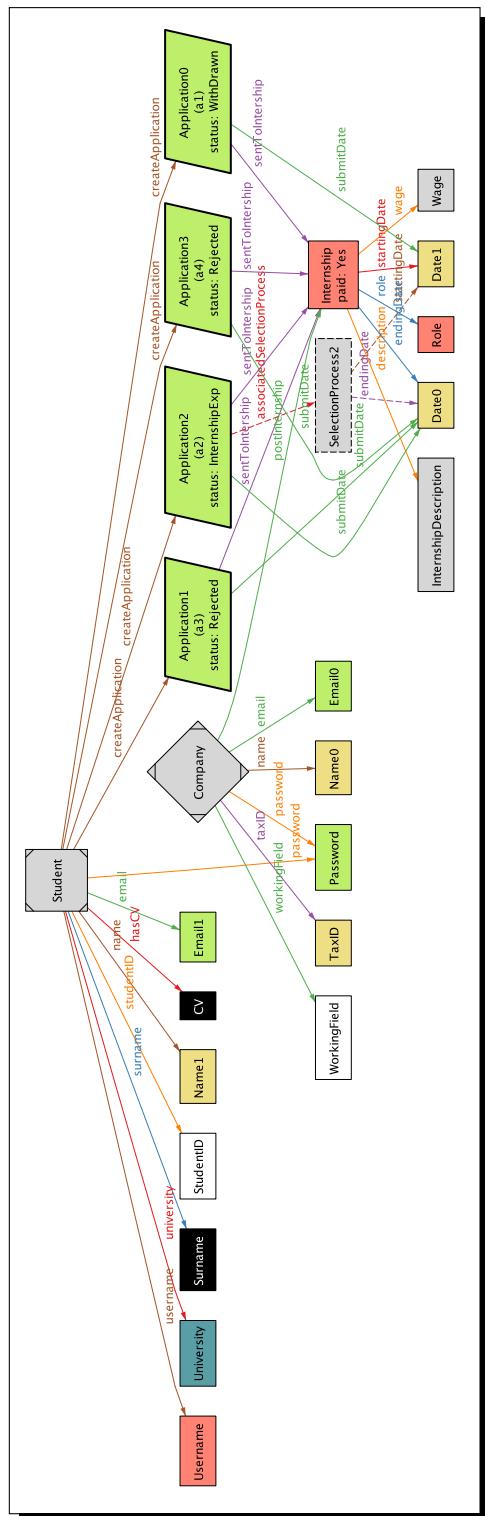


Figure 4.4: Alloy Wold - Rejected and Internship Applications

## *Chapter 5*

---

# References

---

- [\*IEEE Recommended Practice for Software Requirements Specifications\*](#)
- [\*UML Diagrams Examples\*](#)
- [\*Draw.io - Diagram Maker\*](#)
- [\*ChatGPT API - Documentation\*](#)
- [\*Google Meet API - Documentation\*](#)
- [\*Microsoft Teams API - Documentation\*](#)