SECJ2203: Software Engineering

Project Proposal

FYP Management System

8 April 2023

Faculty of Computing

Prepared by: Software Geeks

NO.	NAME	MATRIC NUMBER
1	Nur Immal Hayati Binti Hasmi Anuar (L)	A21EC0111
2	Ain Batrisyia Binti Norazlan	A21EC0009
3	Jeliza Justine A/P Sebastin	A21EC0034
4	Nur Aisyah Fatihah Binti Mohamed Rozi	A21EC0107

Revision Page

a. Overview

This current version of System Documentation contains all materials related to this project. Starting with the purpose of creating this system and scope cover in implementing the system. In this version, it includes all the specific requirements for the system such as the user characteristics, system features that have been interpreted in form of use case diagram and come along with activity diagram and system sequence diagram. It also included the software system attributes, performance and other requirements and lastly the system design constraints.

b. Target Audience

This documentation targeted all the relevant stakeholders.

i. Students of Data Engineering Course (SECPH)
ii. Coordinator from Faculty of Computing
iii. University Coach
iv. Industry Coach
v. Faculty of Computing Management Department

c. Project Team Members

Member Name	Role	Task	Status
Nur Immal Hayati binti Hasmi Anuar	Project Leader	1. UC001: Use Case - User Module 2. UC002: Use Case - Materials Module	Completed
Ain Batrisyia binti Norazlan	Active Member	1. UC003: Use Case - Submission Portal 2. Design Constraints	Completed
Jeliza Justine A/P Sebastin	Active Member	1. UC004: Use Case - Plagiarism Checker 2. Software system attributes, performance and other requirements	Completed

Nur Aisyah Fatihah binti Mohamed Rozi	Active Member	1. UC005: Use Case - Grading System 2. User Characteristic	Completed

d. Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Nur Immal Hayati binti Hasmi Anuar	Completed Chapter 1 to 5	20/5/2023

Table of Contents

1. Introduction

1.1 **Purpose**

The purpose of this System Documentation (SD) is to give a detailed overview of the system development process, which includes the System Requirements Specification (SRS), System Design Document (SDD), and System Testing Document (STD). The SD serves as a reference for individuals involved in system development, such as developers, testers, project managers, and clients. It provides a clear understanding of the system requirements, design, and testing processes, ensuring that all stakeholders are on the same page and working towards common objectives. Through its structured and comprehensive documentation, the SD contributes to the success of the system development process by minimizing errors, improving efficiency, and enhancing overall project outcomes.

1.2 Scope

This project's scope is to grab more FYP-i/PSM-i involved in the industry-integrated project using FYP Management System as a platform to complete their project-based learning activities. The user of this system will be fourth-year Data Engineering students in the Faculty of Computing, an industry coach, a university coach, and the examiners and coordinators. This FYP system allows students to submit their proposals to be reviewed and evaluated by the instructors assigned to them. Generally, the system should provide all necessary information and guidance for users to complete their work. Moreover, it should contain multiple functions for different types of users.

The scope of the FYP Management System web-based software includes the following:

- The software architecture, components, and modules would be designed, considering factors like compatibility with different browsers and operating systems.
- The design would also incorporate best practices in software engineering, including coding standards, testing procedures, and quality assurance processes.
- The software product would be developed following the waterfall approach, where coding is done based on the finalized design. The coding would adhere to established coding standards and best practices.
- Ongoing support, bug fixes, and updates would be provided to ensure that the software continues to meet the specified requirements and perform as expected.
- The system will be tested using a combination of manual and automated testing methods to ensure that it meets all specified requirements and performs as expected.

1.3 Definitions, Acronyms and Abbreviation

Definitions of all terms, acronyms and abbreviations used are to be defined here.

Term	Definition
Integrated Project Proposal System	The system is used to manage the project proposal of SECPH students.
FYP-i/PSM-i	Final Year Projects with Industry
SD	System Documentation
SDD	Software Design Description
SRS	Software Requirement Specification
Coordinators	The person who manages the whole system and has access to every platform.

Table 1.0: Definitions of all terms, acronyms, and abbreviations

1.4 **References**

- Thakur, D. (2013). *Software Design Documentation (SDD). Computer Notes.* https://ecomputernotes.com/software-engineering/softwaredesigndocumentation

- Rome,P. (2023), *What are Non Functional Requirements - With Examples* Retrieved May 18, 2023, from https://www.perforce.com/blog/alm/what-are-non-functional-requirements-examples

- Rajkumar. (2023), *What are Quality Attributes in Software Architecture* Retrieved May 18, 2023, from https://www.softwaretestingmaterial.com/quality-attributes-in-software-architecture/

1.5 **Overview**

This System Documentation contains all materials related to build this Industry Integrated Project Proposal System which is also known as Final Year Project Management System. This document has 3 main sections which are System Requirements Specification (SRS), System Design Document(SDD), and System Testing Document(SDS).

This System Documentation is organized as follows:

Purpose and Scope System - The objective of creating the system and all the tools used in implementing the system.

Specific Requirements - The user characteristics, and system features that have been interpreted in the form of a use case diagram and come along with an activity diagram and system sequence diagram.

Software System Attributes, Performance, and Other Requirements - Comprises all the relevant attributes for the system and performance requirements.

System Design Constraints - List of all the design constraints that need to pay more attention in developing the system.

This document is used to give all the stakeholders involved a clear view of this project. It provides all the information about the software development, design, and testing procedures throughout the project lifecycle.

2. Specific Requirements

2.1 **User characteristics**

The "FYP Management Systems" software will be used by five types of users: data engineering students, coordinators, examiners, university coaches, and industrial coaches.

2.1.1 Data engineering students

- Data engineering students who will use the software are expected to have basic computer skills, including familiarity with e-learning web-based applications.
- They need to submit their draft of the FYPi project proposal step by step before making the final submission to be graded by the examiners.
- Some FYP projects may be plagiarism from other resources so a plagiarism checker is a must for students to verify the authenticity of their project
- Sometimes students may not be aware if the project was successfully submitted or not, but the system will notify the students if they successfully submitted so they can be certain that their work has been received.

2.1.2 Coordinators

- The coordinator must undergo comprehensive training to understand the system's functions.
- All mandatory requirement forms such as rubric, e-evaluation, no-plagiarism endorsement form, and comment form for students should be uploaded in a single location with file tagging provided by the system.
- The system will also provide features for the coordinator to create a submission, and change the time and due date of an existing submission platform.
- Coordinator releases the official students' grades after verifying marks given by examiners, industrial coaches as well as university coaches.

2.1.3 Examiners

- Examiners are responsible for evaluating papers and are expected to uphold fairness, as the system will automatically reject submissions from students that are submitted after the deadline.
- Examiners are permitted to download and evaluate files uploaded by students, as the system grants them access to these files for grading purposes.
- The system requires examiners to submit an approval form for each student, indicating that all mandatory requirements have been fulfilled by the student, before allowing the examiners to allocate marks.

2.1.4 Industrial Coach

- Capable of delivering an accurate draft to the student within the anticipated time frame.
- System provides details and contact information of the industrial coach.

2.1.5 Universities Coach

- The system will provide a list of students assigned to a particular supervisor and set a limit on the number of students that the supervisor can access.
- The system sends reminders to the university coach to review the progress of their supervisees.

2.2 **System Features**

The Final Year Project Proposal Management system is a software that operates on a browser that the students from the Data Engineering course will use as their platform for updating the progress of their final year project proposal. This system is proposed as the need to enhance the process and management system efficiently compared to the previous system. In this project, a web-based software solution will be introduced to eliminate the current tedious way of managing students' final-year projects by developing a computerized system that assists in managing all the operations of the final-year project. It also provides a medium for easy communication between students and lecturers or coordinators during all the phases.

The system features are illustrated in Figure 1.1 below. The detailed description of each module and function is tabulated in Table 1.1.

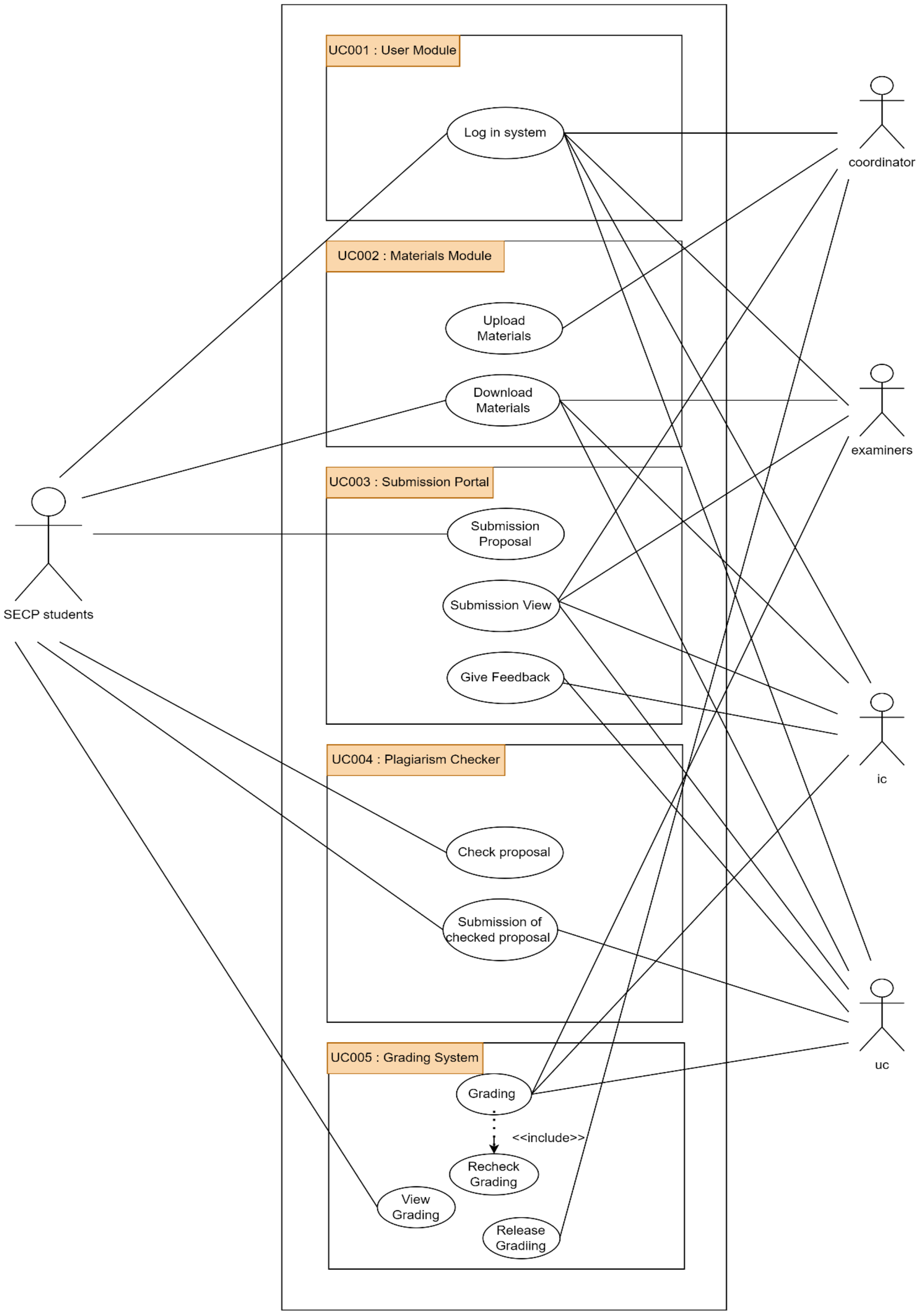

Figure 1.1: Use Case Diagram for The Final Year Project Proposal Management System

Table 1.1: Description of Module and Functions for Final Year Project Proposal Management System

Module	Function	Description
User Module	UC001 – Log in	This use case allows users to log in to the system.
Materials Module	UC002 - Access Materials	This use case allows users to upload and download all the important materials for the proposal.
Submission Portal	UC003 - Submission	This use case allows users to submit the document for their project.
	UC004 - View and Feedback	This use case allows the user to view the submission and add feedback to give comments.
Plagiarism Checker	UC005 - Check Plagiarism	This use case allows users to check the percentage of similarity with the resources they used to refer.
Grading System	UC006 - Grading	This use case allows users to do grading, recheck, release the grading,view the grading.

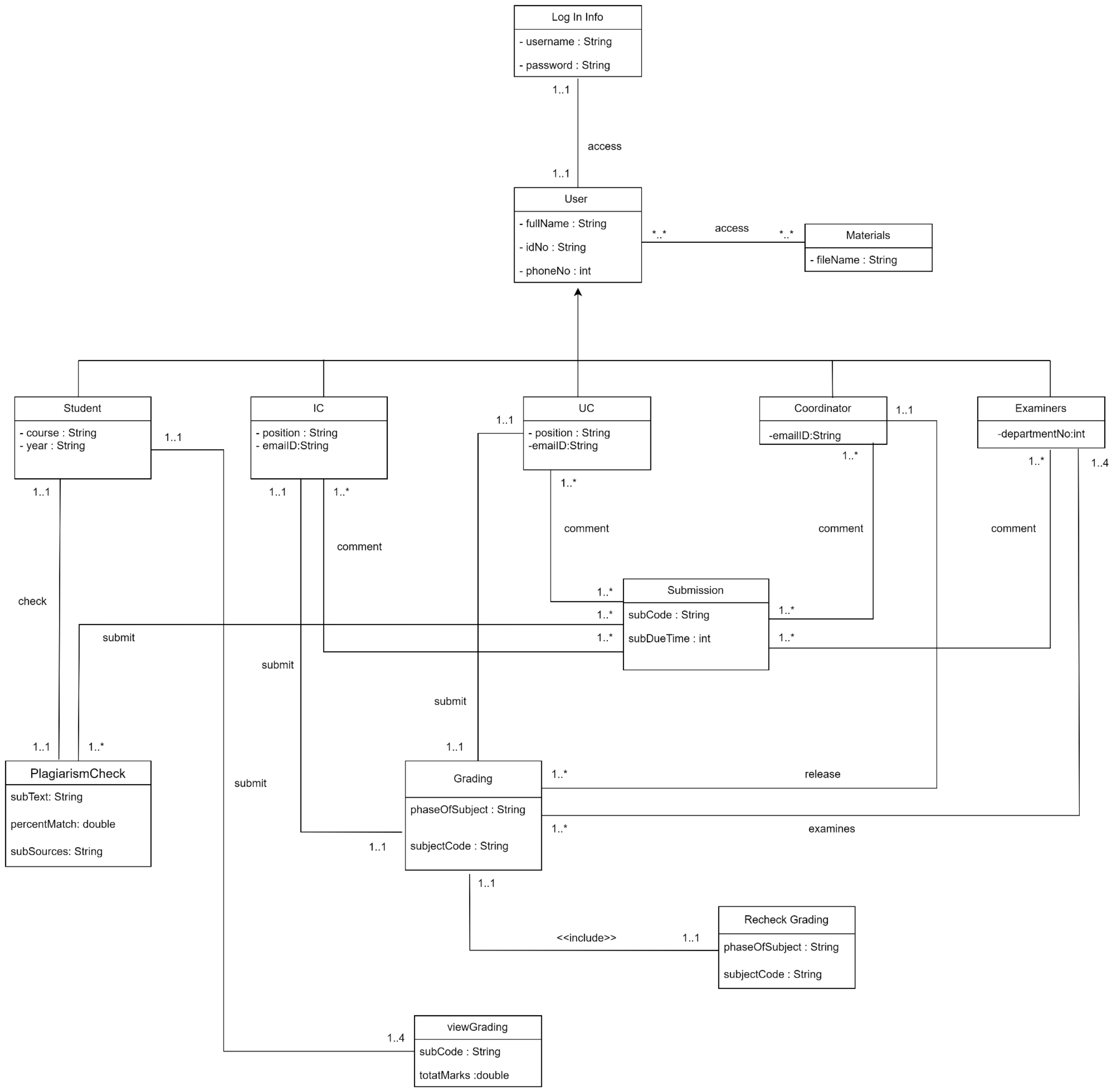

Figure 1.2: Domain Model Diagram for The Final Year Project Proposal Management System

2.2.1 **UC001: Use Case - User Module**

Use case: User Module
ID: UC0001
Actors: SECP Student, Coordinator, UC, IC, Examiner
Preconditions: Has stable internet connection to access the platform
Flow of events: 1. The use case starts when the user key in their username. 2. Then, the user will key in their authorized password. 3. The user then clicks the button Log In. 4. The system will validate the username and password entered whether it is valid or not. 5. If the account is invalid, Exception 1 will follow. 6. If the account is valid, the user will be directed into the home page of the system. 7. The use case ends.
Exception flow: 1. Invalid password or username 1.1. Display error message

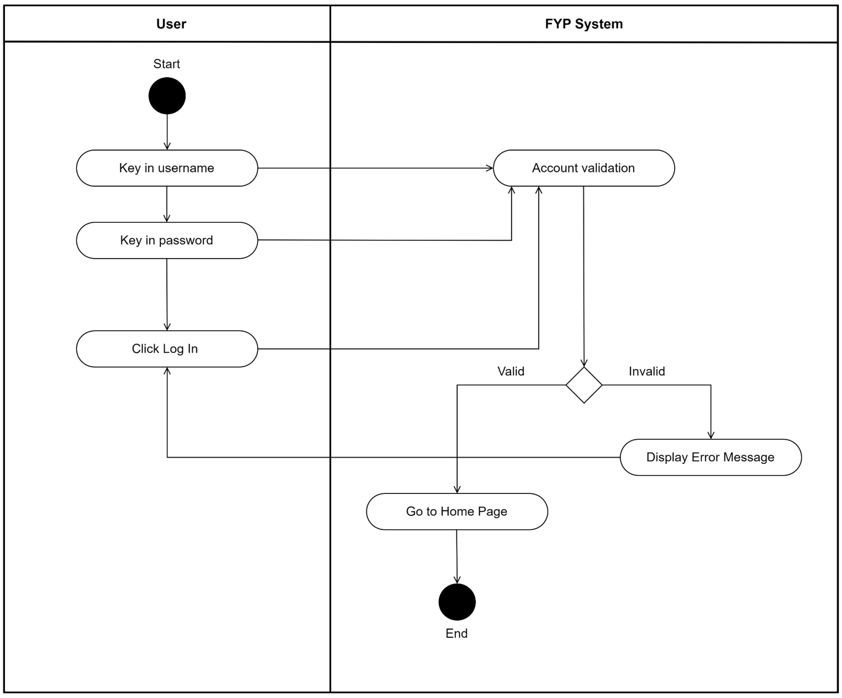

Figure 1.3: Activity Diagram for User Modules

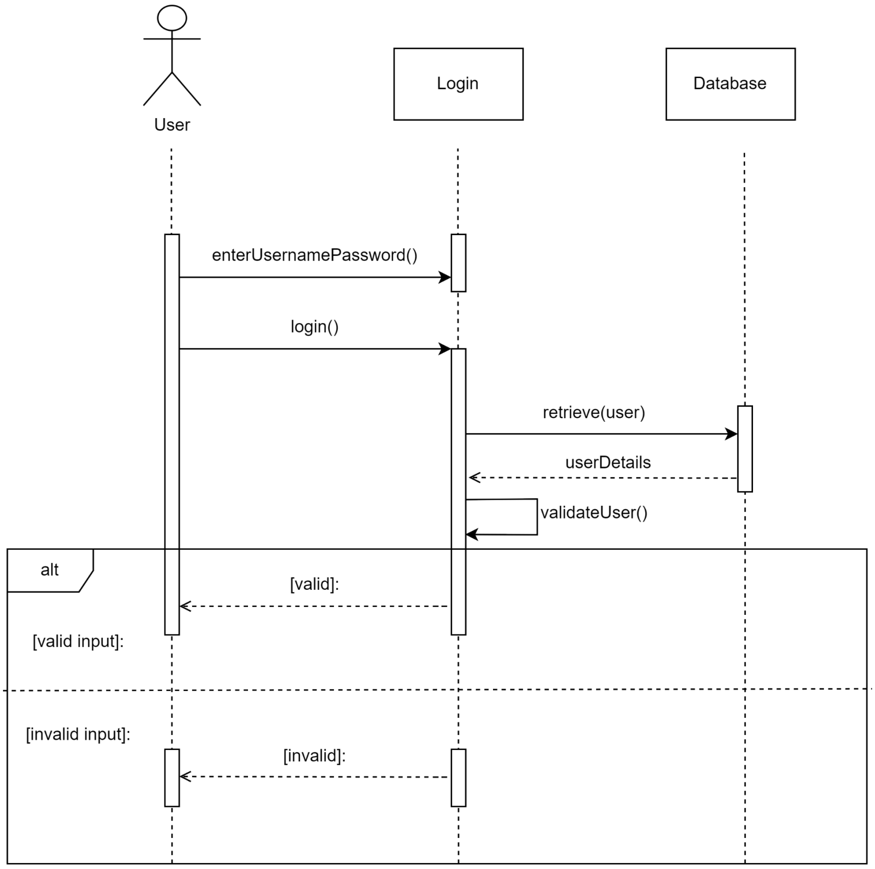

Figure 1.4: Sequence Diagram for User Modules

2.2.2 UC002: Use Case - Materials Module

Use case: Materials Module
ID: UC0002
Actors: SECP Student, Coordinator, UC, IC, Examiner
Preconditions: Users already log in into the system.
Flow of events: 1. The use case starts when the user clicks the Materials button. 2. Then, the system will display the list of materials. 3. Coordinator will click the Add File button. 4. The system will display an upload file window pane. 5. Coordinator will then choose the file from the browser to upload into the system. 6. Then, the coordinator will click the Upload button. 7. The other user then chooses a file to download it. 8. Then the user will download the file. 9. The system will display a Download Successful message. 10. The use case ends.
Postconditions:

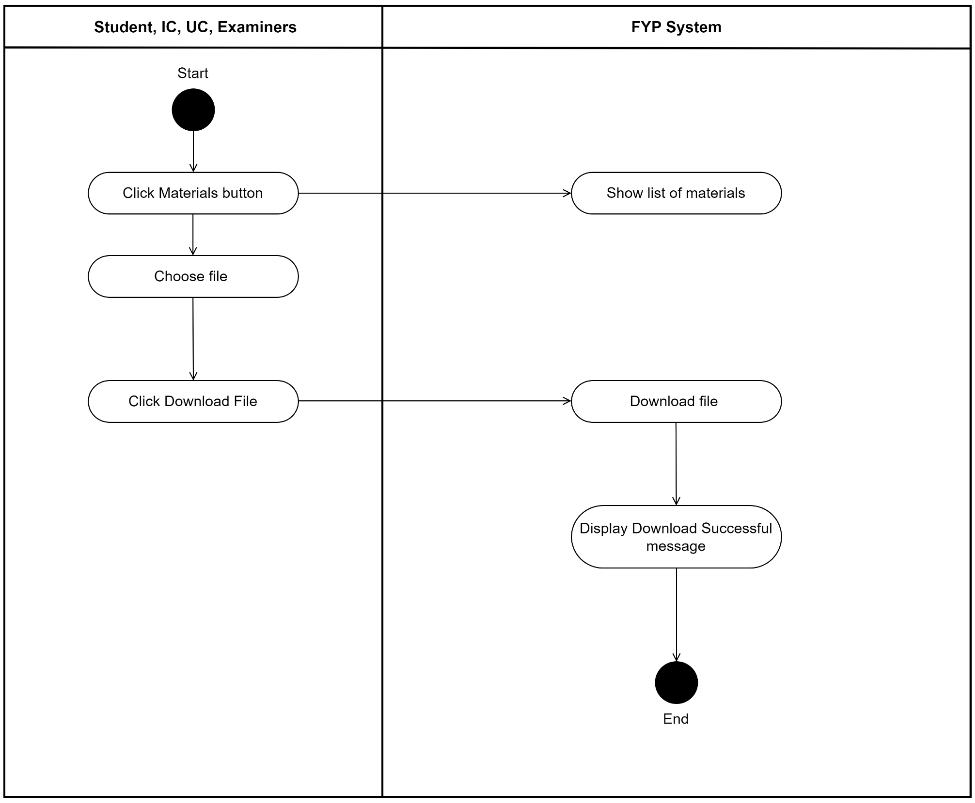

Figure 1.5: Activity Diagram for Materials Modules (Download Materials)

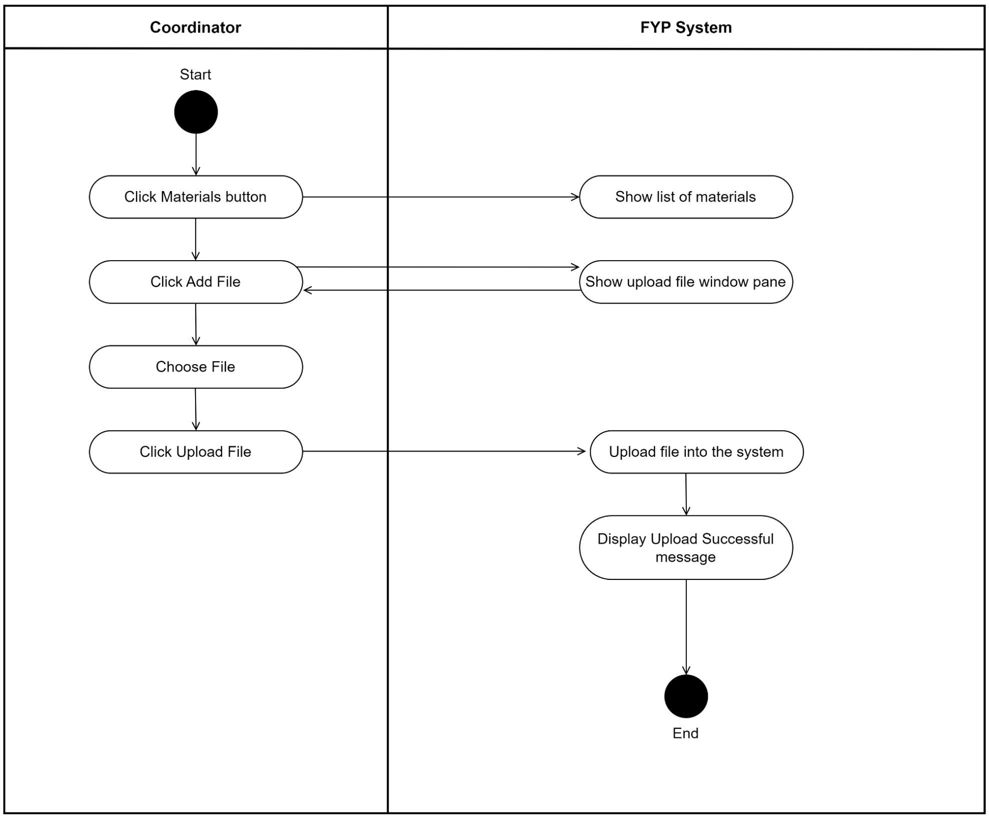

Figure 1.6:Activity Diagram for Materials Modules (Upload Materials)

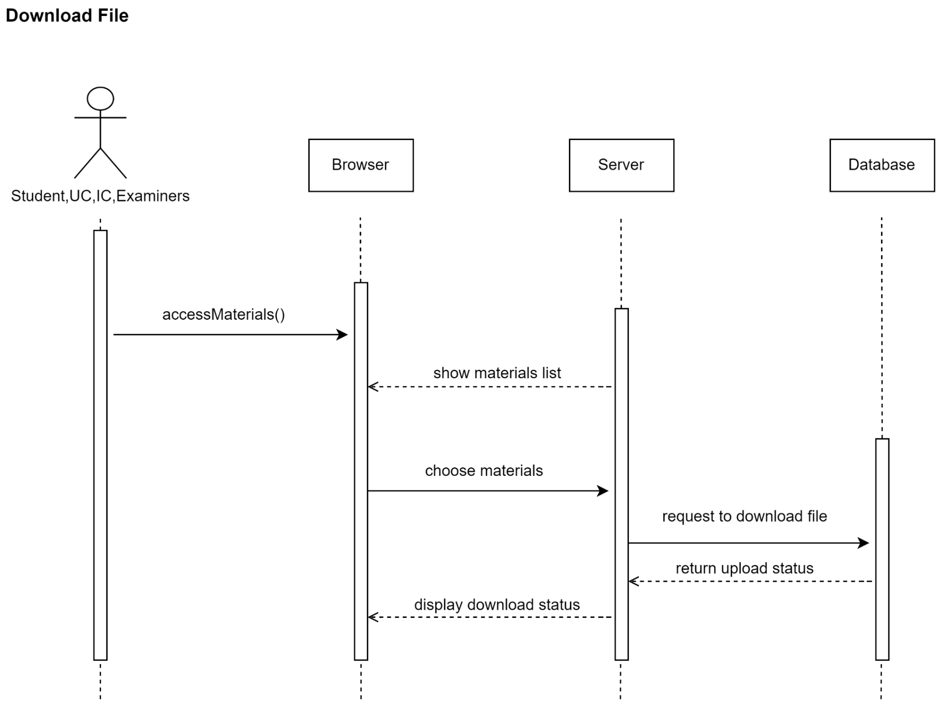

Figure 1.7: Sequence Diagram for Materials Modules (Download Materials)

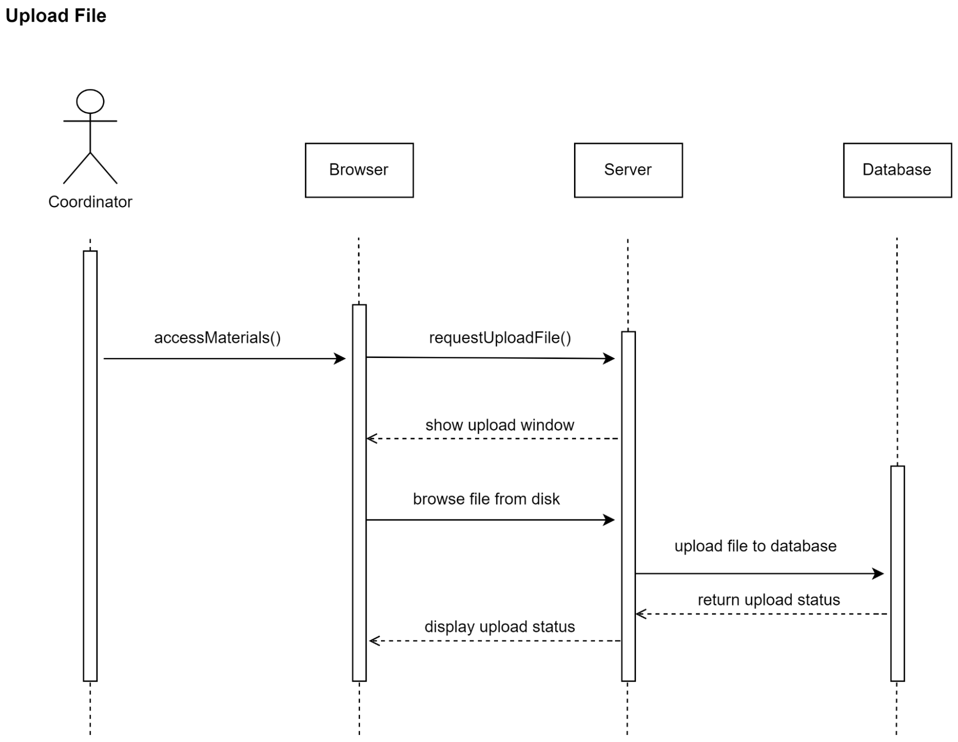

Figure 1.8: Sequence Diagram for Materials Modules (Upload Materials)

2.2.3 UC003: Use Case <Submission>

Use Case: <Submission>
ID: UC003
Actors: Student
Preconditions: 1. Has active internet connection. 2. Must submit in PDF format for report and pptx format for presentation slide.
Flow of events: 1. The use case starts when the user taps on the submission page. 2. System displays link submission for each chapter and clicks the chosen chapter. 3. System displays the submission page containing the submission form. 4. The user clicks on the ‘add submission’ button. 5. Choose documents from their folder. 6. System validate the document. If the document is invalid, Exception 1 is followed. 7. Click the submit button. 8. System display submission successful. 9. The use case ends.
Postconditions:

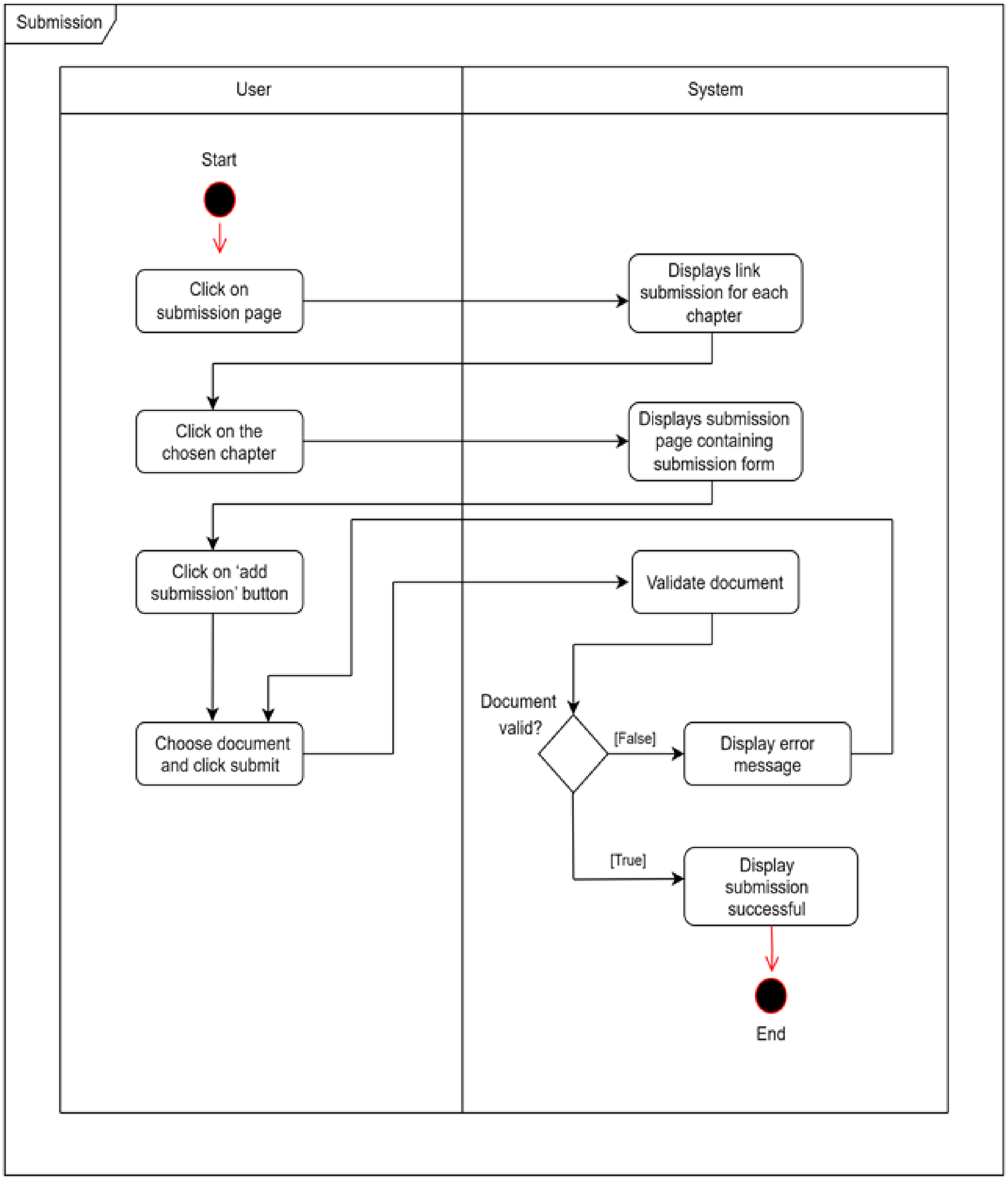
Submission
User
System
Start
Click on submission page
Displays link submission for each chapter
Click on the chosen chapter
Displays submission page containing submission form
Click on 'add submission' button
Validate document
Choose document and click submit
Document valid?
[False]
Display error message
[True]
Display submission successful
End

Figure 1.9: Activity Diagram for Submission Portal

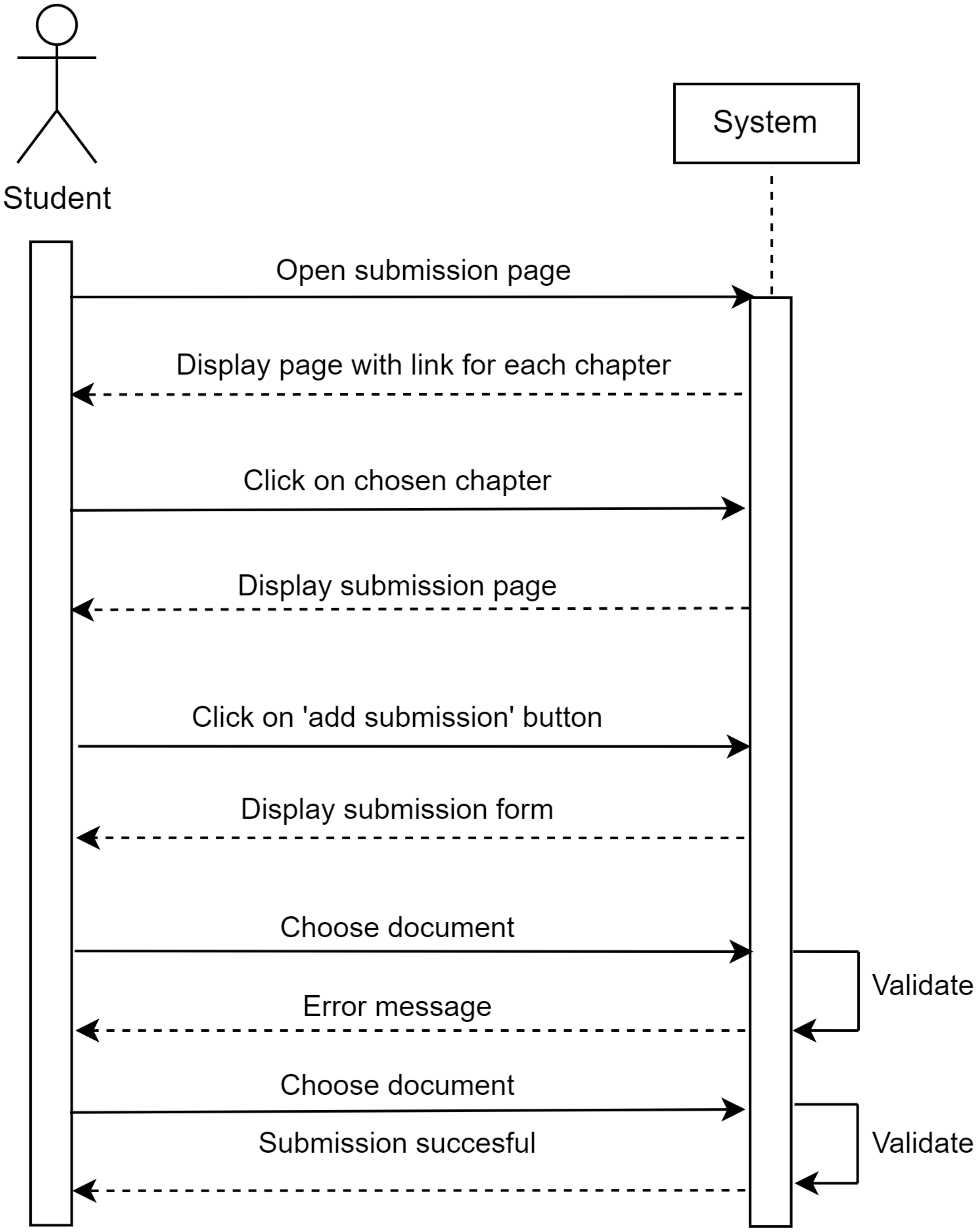

Figure 2.0: Sequence Diagram for Submission Portal

2.2.4 UC004: Use Case <View and Feedback>

Use Case: <View and Feedback>
ID: UC004
Actors: University Coach, Industry Coach, Examiner
Preconditions: 1. Has active internet connection. 2. There is a document to be reviewed.
Flow of events: 1. The use case starts when the user taps on the submission page. 2. System displays a list of student names assigned to each user. 3. System displays four gray buttons beside each student's name with each button representing each chapter. 4. If the button display green 4.1 Students have submitted the assignment. 5. Else 5.1 There has been no submission from the student yet. 6. The user clicks on the student name that they want to review. 7. System displays the overall submission from the chosen student. 8. The user clicks on the ‘add feedback’ button. 9. Write a feedback comment to help students improve their work 10. Submit the feedback. 11. System display feedback has been added successfully. 12. The use case ends.
Postconditions:

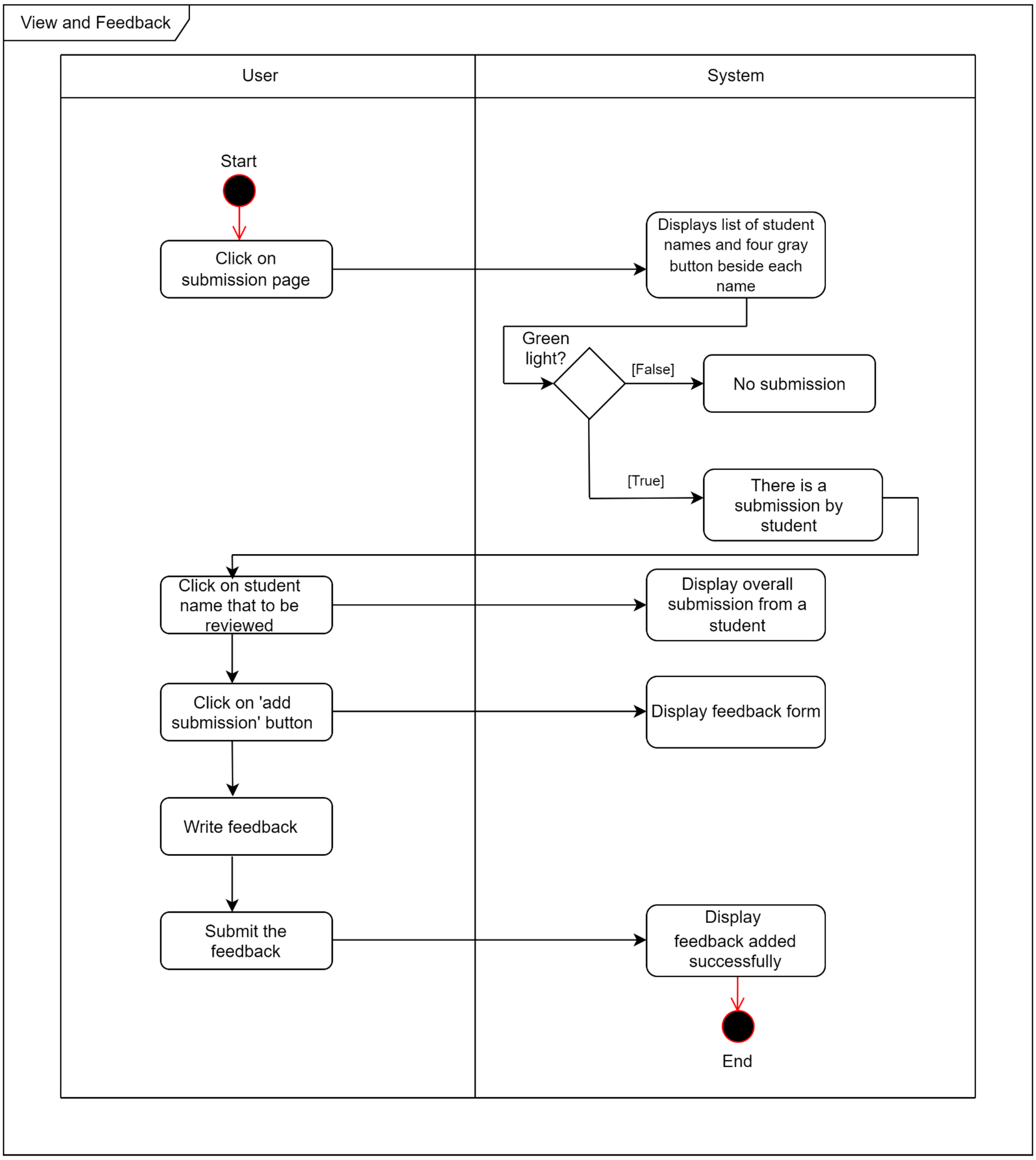
View and Feedback
User
System
Start
Click on submission page
Displays list of student names and four gray button beside each name
Green light?
[False]
No submission
[True]
There is a submission by student
Click on student name that to be reviewed
Display overall submission from a student
Click on 'add submission' button
Display feedback form
Write feedback
Submit the feedback
Display feedback added successfully
End

Figure 2.1: Activity Diagram for View and Feedback

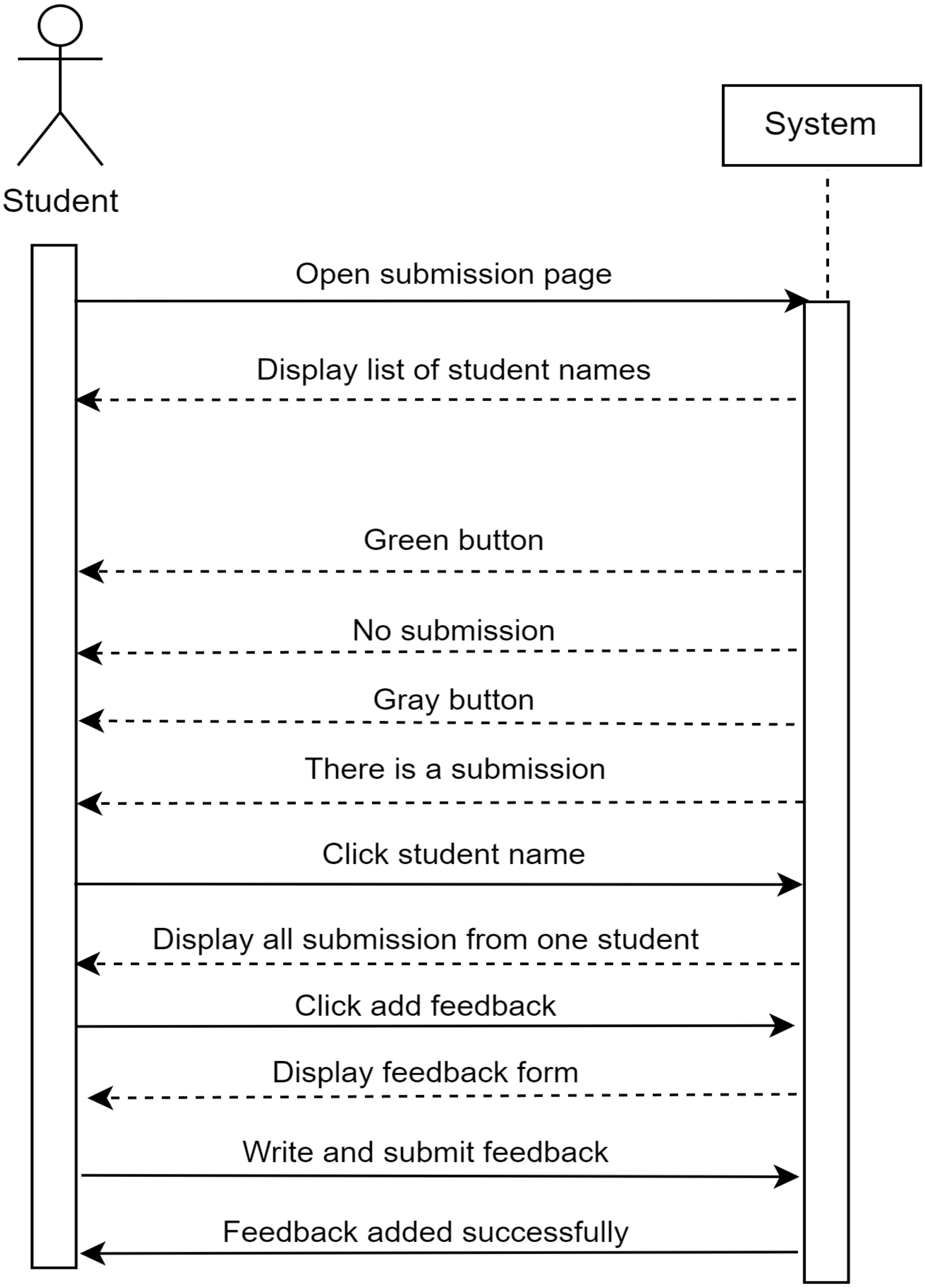

Figure 2.2: Sequence Diagram for View and Feedback

2.2.5 UC005: Use Case <Plagiarism Checker>

Use Case: <Plagiarism Checker>
ID: UC005
Actors: UTM Student
Preconditions: 1. The student is logged on to the system.
Flow of events: 1. The use case starts when the user taps on "Plagiarism Checker" in the menu after logging in. 2. System displays the plagiarism checker page containing the file upload section. 3. The user uploads his/her file. 4. The user taps on "Check Plagiarism". 5. System shows the file uploaded with the percentage of matching detected and its sources. 6. System displays the download option at the bottom center of the page. 7. If the percentage of matching is more than the percentage set by the management system then 7.1 User modify or re-do the project phase. 7.2 Check again in the plagiarism checker. 8. Else 8.1 Users download the file and proceed to the submission portal.
Exception Flow: 1. Percentage of matching exceeds the fixed value 1.1 System displays a warning message.

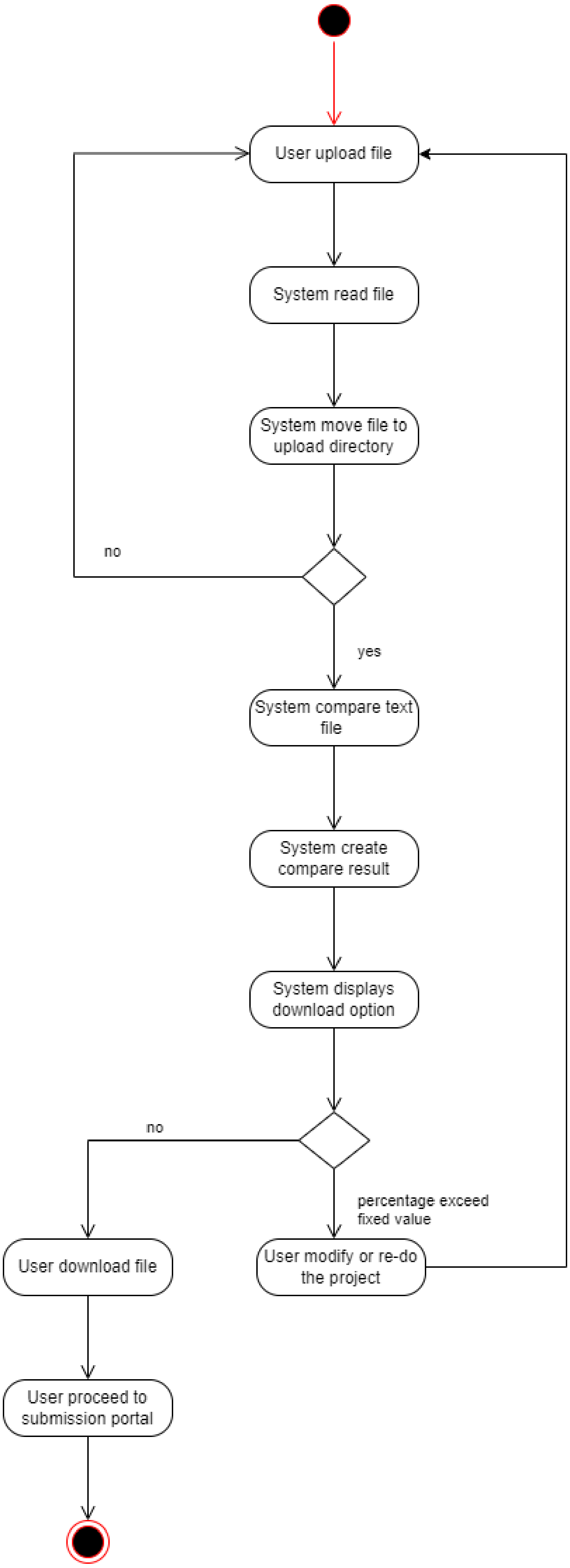
User upload file
System read file
System move file to upload directory
no
yes
System compare text file
System create compare result
System displays download option
no
percentage exceed fixed value
User download file
User modify or re-do the project
User proceed to submission portal

Figure 2.3: Activity Diagram for Plagiarism Checker

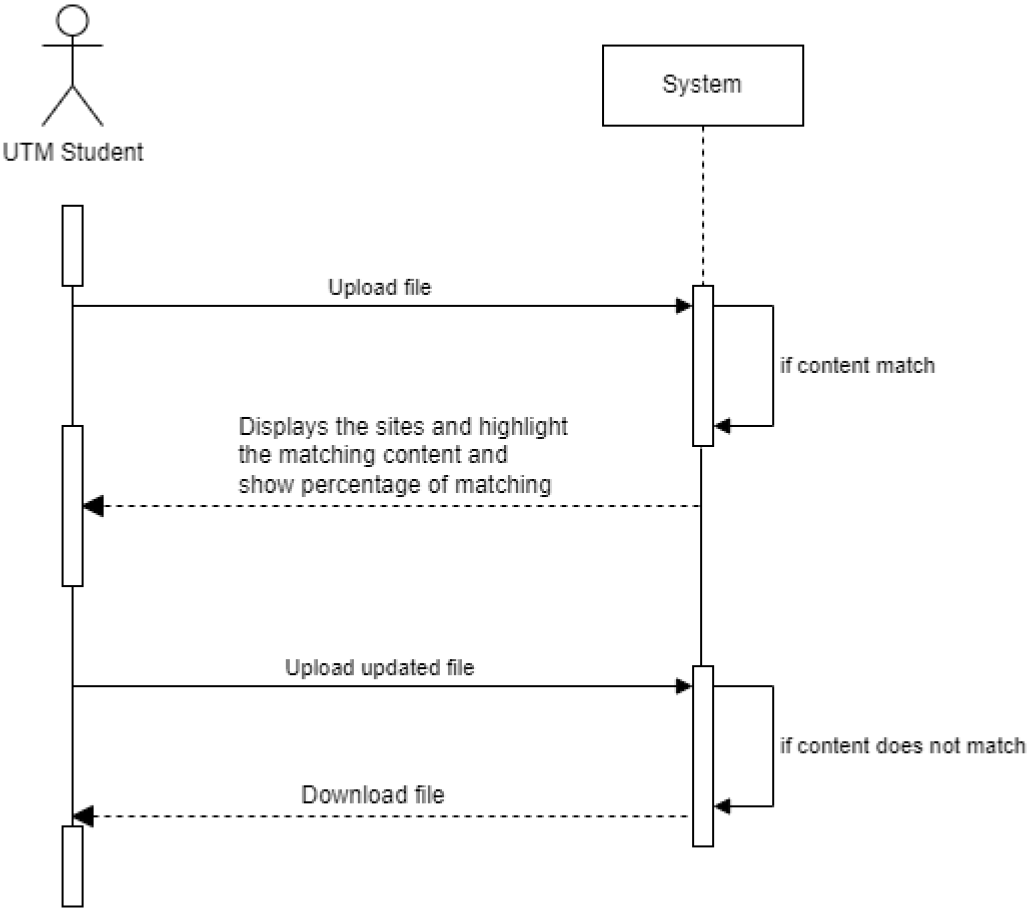
UTM Student
System
Upload file
if content match
Displays the sites and highlight the matching content and show percentage of matching
Upload updated file
if content does not match
Download file

Figure 2.4: Sequence Diagram for Plagiarism Checker

2.2.6 UC006: Use Case <Grading>

Use Case: <Grading>
ID: UC006
Actors: Examiners, Industry Coach, University Coach
Preconditions: Examiners, Industry Coaches, University Coaches username and password to log into the system. A submission has been made by students.
Flow of events: 1. Examiner, Industry Coach, and University Coach login to the system. 2. If the user is examiner 2.1 While the student has few submission topics 2.1.1 Examiner open submission file 2.1.2 Examiner does the grading based on each chapter 2.1.3 If the examiner is done with the grading <<RecheckGrading>> 3. Else 3.1 Industry and University coach does the grading based on the student's skills 3.2 If the University and Industry Coach has done <<RecheckGrading>> 4. Examiner, UC, and IC submit the grading to system 5. The system records and saves the current marks given.
Postconditions: The coordinator receives the grades and needs their verification

Use case: Recheck Grading

ID: UC006

Actors:

Examiner, Industry Coach, University Coach

Includes:

UC006<< Grading>>

Preconditions:

A valid Examiner’s, Industry Coach’s, University Coach’s username and password to log into the system

Students have made a submission.

At least one grading has occurred.

Flow of events:

1.Examiner, Industry Coach, University Coach login to the system.

2. If user is Examiner

2.1 while the student have 4 courses

2.2 Examiner clicks on the recheck button.

2.3 System provides the graded course paper.

2.4 Examiner does the grading based on each course code.

3.Else

3.1 Industry Coach and University click on recheck button

3.2 System provides the graded assessment form.

3.3 Industry Coach and University Coach does the grading based on the skills of each student.

3.4 The Examiner, Industry Coach and Examiner Coach will submit the grading in the System.

3.5 The system saves and records the grades of each student.

Use case: Release Grading

ID: UC006

Actors:

Coordinator

Preconditions:

A legitimate Coordinator login and password, as well as grades, have been supplied.

Each course has received a grade by examiners and UC and IC.

Flow of events:

1.The Coordinator login to the system.
2.The Coordinator navigates to the Grading interface.
3.The Coordinator compares the recheck grades for each course.
4.The Coordinator will compile and write a report about the grades of the students.
5.The Coordinator release their grading when clicking submit button
6.The system displays a confirmation message.
7.If the Coordinator press confirms.
  7.1. The System displays the grades.
8.Else
  8.1. The System returns back to the grading interface.

Postconditions:

The grades are displayed with students' information.

Use case: View Grading

ID: UC006

Actors:

Students

Preconditions:

A valid student's username and password to log into the system.

Submissions have been made by the students.

A grading has been made by the examiner, industry coach and university coach.

Flow of events:

1. The Student login to the system.
2. The Student opens the Grading Interface.
3. System asks students for course code.
4. The Student enters the code.
5. If code entered is valid
   5.1. System will display the grade.
6. Else
   6.1.System will display invalid code.

Postconditions:

The student receives the grades.

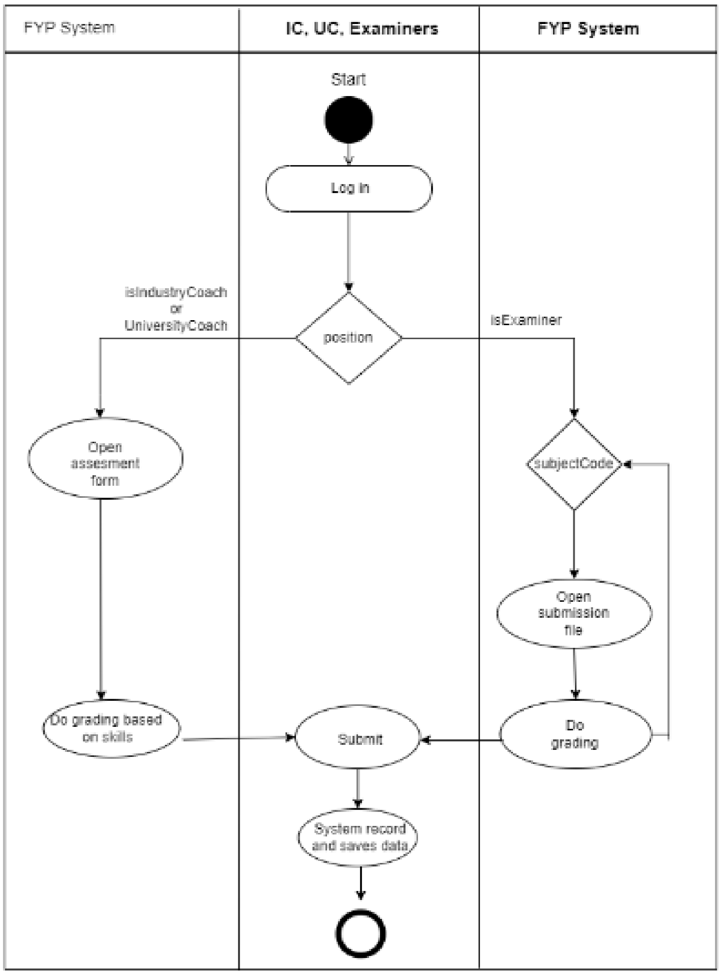

Figure 2.5: Activity Diagram for <Grading>

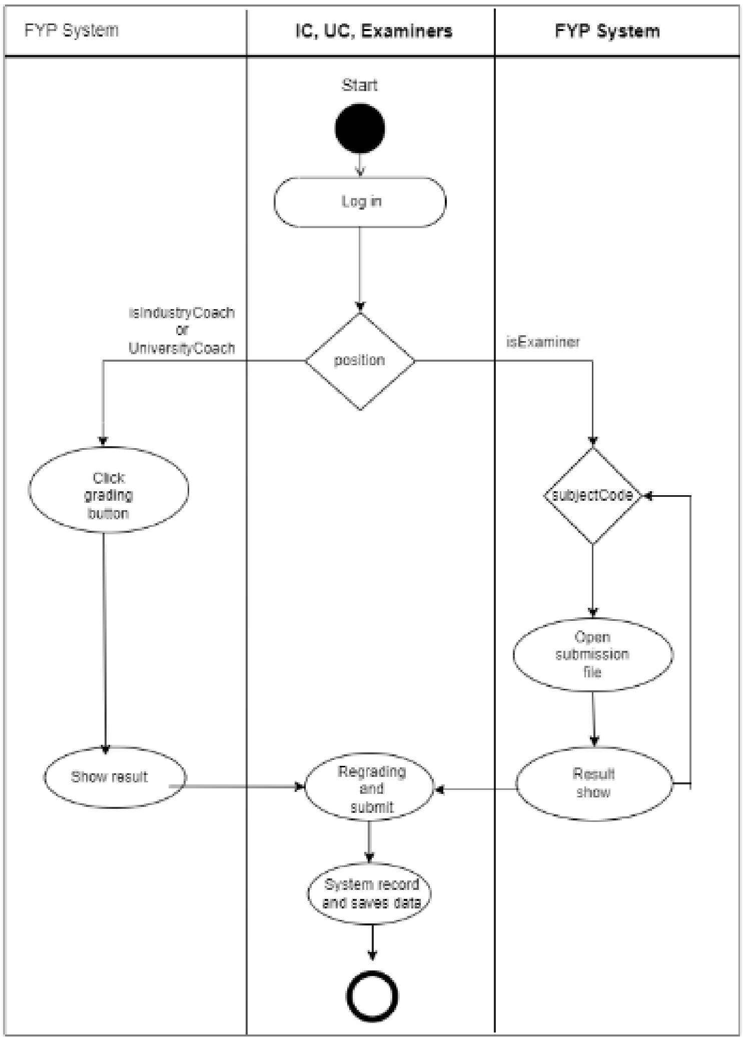

Figure 2.6: Activity Diagram for <Recheck Diagram>

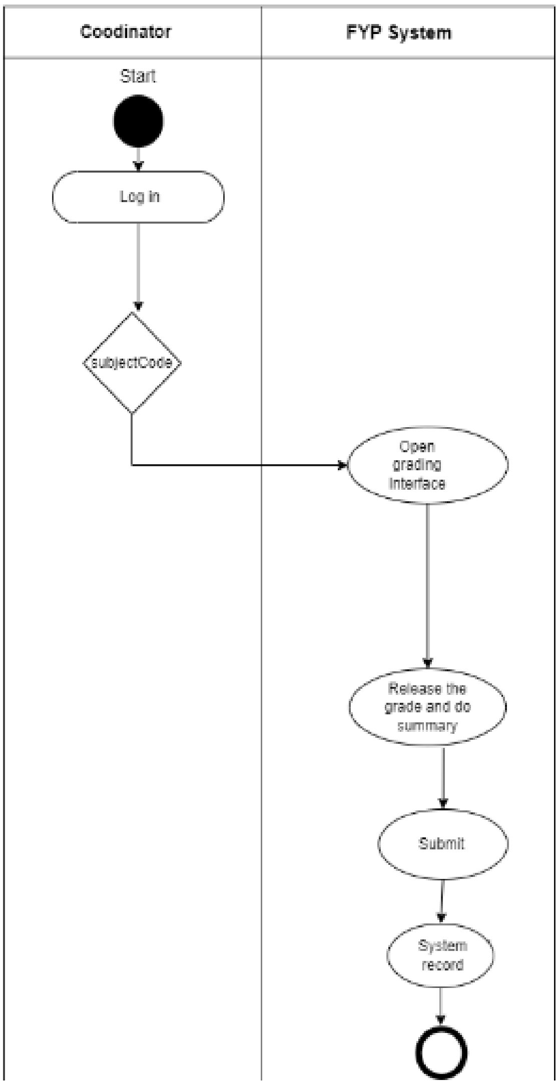

Figure 2.7: Activity Diagram for <Release Grading>

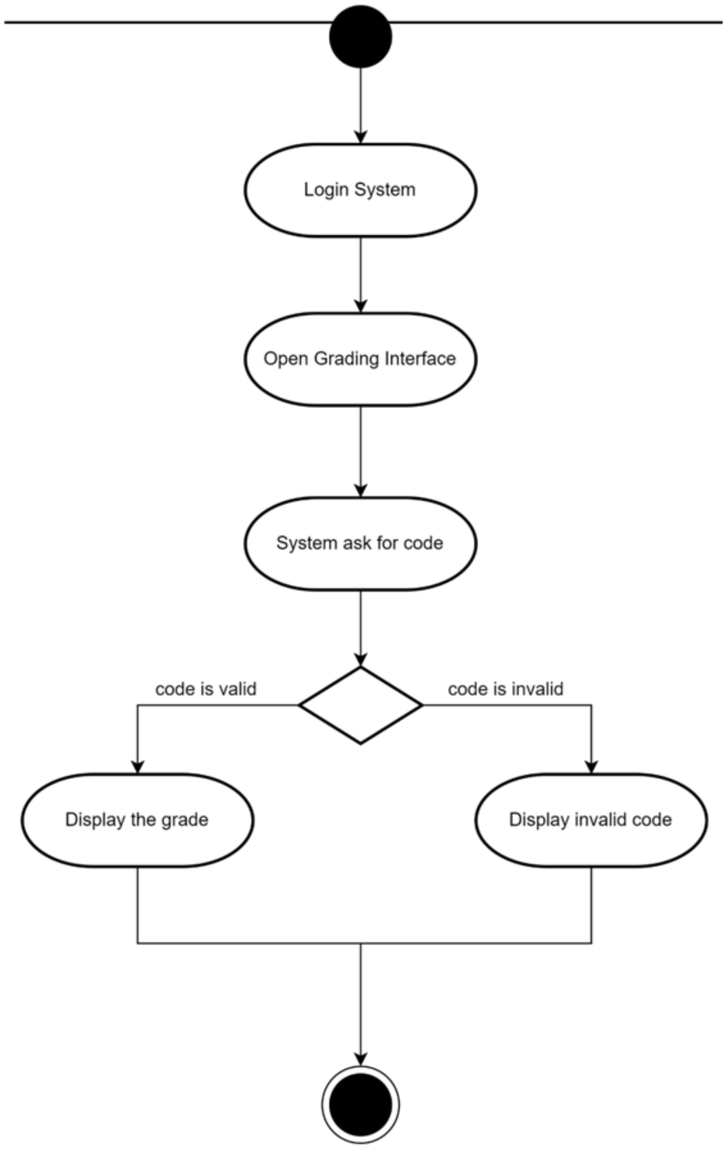

Figure 2.8: Activity Diagram for <ViewGrading>

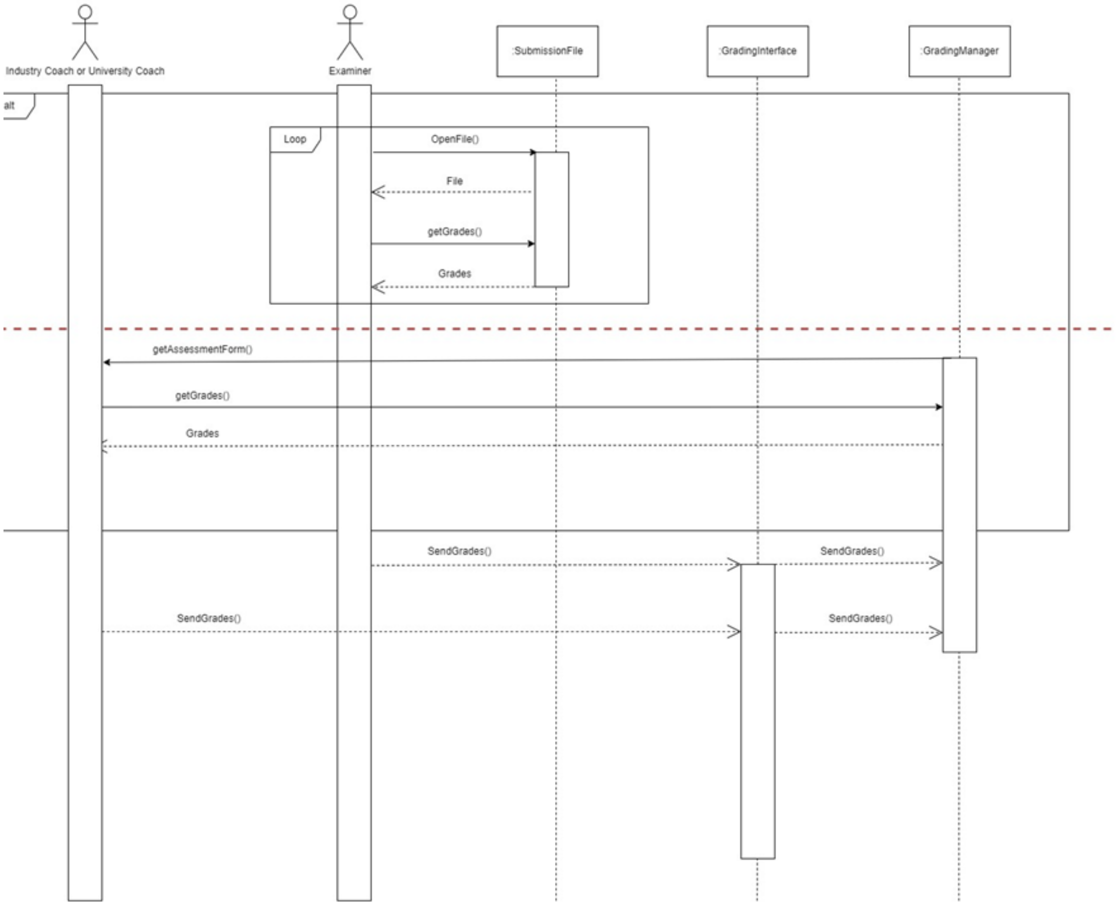

Figure 2.9: Sequence Diagram for <Grading> and <RecheckGrading>

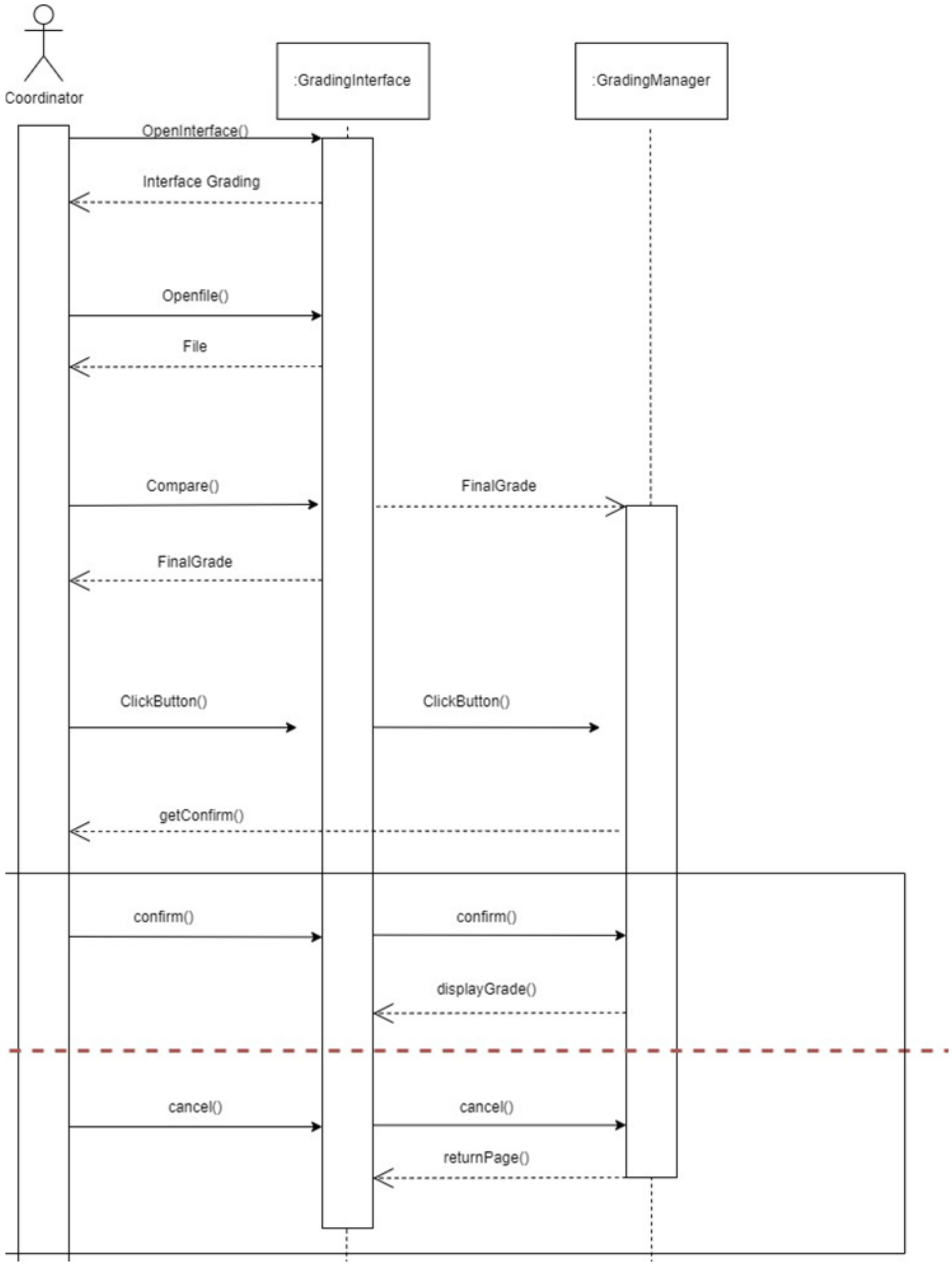

Figure 3.0: Sequence Diagram for <ReleaseGrading>

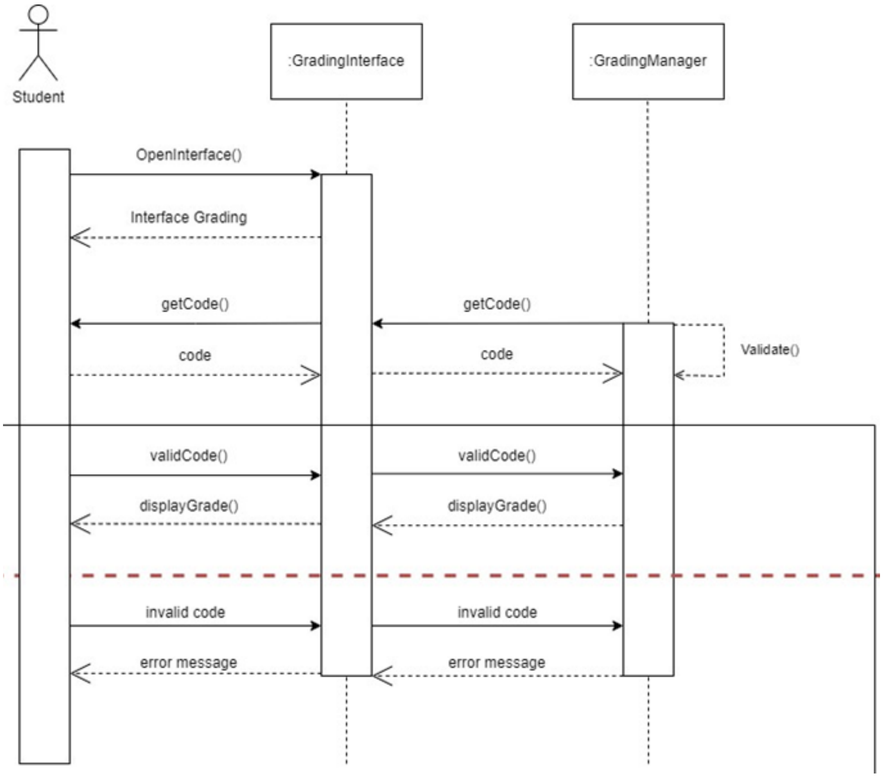

Figure 3.1: Sequence Diagram for <ViewGrading>

2.3 **Software System Attributes, Performance, and Other Requirements**

Non-functional requirements are the requirements that describe how the system should behave or operate, rather than what it should do. They address the characteristics or qualities of the system, such as its usability, reliability, performance, security, maintainability, and compatibility. Non-functional requirements are important to write in the SD because they help ensure that the system meets the stakeholders' needs and expectations. The ISO/IEC/IEEE 29148 standard suggests that non-functional requirements should be specified under three main categories: Software System Attributes, Performance, and Other Requirements.

When designing a final-year project management system, it's important to consider various software system attributes to ensure its effectiveness, efficiency, and usability. Examples of some key attributes of the Software System are as follows:

- Functionality: The system should provide essential features for managing final-year projects, such as modules to access materials, a submission portal to submit documents, a plagiarism checker, and a grading system.
- Usability: The system should have an intuitive and user-friendly interface, making it easy for students, coordinators, examiners, industry coaches ,and university coaches to navigate and use the system effectively.
- Reliability: The system should be dependable, ensuring that it performs its intended functions consistently and accurately. It should have minimal downtime and handle errors gracefully. Data integrity and backup mechanisms should be in place to prevent data loss.
- Maintainability: The system should be designed with maintainability in mind, making it easier to modify, enhance and fix issues in the future.
- Compatibility: The system should be compatible with various platforms and devices, such as desktop computers, tablets, and mobile devices. It should be responsive and adapt well to different screen sizes and resolutions.

Performance requirements define the system's capability to respond to user requests and handle data in a timely manner. Performance is a critical attribute for a final-year project management system as it directly impacts user experience, productivity, and the system's ability to handle increasing workloads. these requirements include the following:

- Response Time: The probability of the system will return results within 1 second and it varies if the internet connection is slow.
- Scalability: The increase in the system's workload that the system should be able to process.
- Capacity: The system should handle a maximum number of users and maximum amount of data.
- Throughput: The system should handle a maximum number of requests in a given period of time.

Other requirements refer to any non-functional requirements that do not fit under the categories of software system attributes or performance. These requirements include the following:

- Security: Security requirements are to protect sensitive project information and user data. The system should include authentication mechanisms, access controls, encryption requirements and secure communication protocols to ensure the system's security and prevent unauthorized access or data breaches.
- Data Management: The system should have data backup and retention policies, data validation rules, data archiving requirements and data privacy regulations to ensure proper management and protection of project-related data.
- Environmental: The system should give a positive impact to the environment. This may include reducing energy consumption, minimizing paper usage through digital workflows or adhering to environmentally friendly practices in system design and operations.
- Legal and Regulatory: The system should ensure compliance with data protection and privacy laws. This includes obtaining appropriate consent for data collection and processing, implementing security measures to protect personal information and providing users with control over their data.

2.4 **Design Constraints**

Design constraints refer to limitations or specifications that influence the design process of a software, product, or system. It sets boundaries and limitations within which the design must operate, and guides the design process by providing a framework for making decisions.

There are lots of constraints that may need to be considered to ensure that the software is effective and appropriate for its intended use. The system will have hardware, security, compatibility, and performance constraints. Hardware constraint Security constraints must address authentication, authorization, data encryption, secure communication, and other relevant security measures. Compatibility constraints

Constraint	Description
Hardware constraints Include considerations such as available memory, processing power, storage capacity, and network connectivity.	The system is designed to accept any accepted format (.pdf or .pptx) with a maximum of 8GB file size and must have an active internet connection.
Security constraints must address authentication, authorization, data encryption, secure communication, and other relevant security measures.	The system is designed to ensure that only authorized students can access and submit documents.
	The system will implement regular data backups to prevent data loss in case of system failures, disasters, or accidental deletion. Establish reliable recovery mechanisms to restore the system and the submitted documents to a functional state.
Compatibility constraints Refer to a requirement or limitation that ensures the compatibility of a software system with other systems, components, or environments	The system will be designed to run on specific operating systems, such as Windows, macOS, Linux, or mobile platforms like Android or iOS. Also compatible with different browsers (e.g., Chrome, Firefox, Safari, Internet Explorer)
Performance constraints To ensure that the system performs optimally across different web browsers, devices, and network conditions	The system must be designed to support a maximum of 1000 concurrent users with a response time of fewer than 5 seconds across various devices, including desktops, laptops, tablets, and mobile phones