Software Requirements Specification

Master Dissertation Management System

Version 1.0

Printing Date

Software Engineering, Faculty of Computer

Prepared by: Arif Suffian Bin Muhamad Zaili

Revision Page

a. Overview

Describe the content of the current version.

b. Target Audience

State the targeted audience.

c. Project Team Members

List the team members and respective assigned module.

d. Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
<Current Version>			

Note:
This Software Requirements Specification (SRS) template is based on IEEE Std 830-1998, organized by modules according to system features (Appendix A.5 of the IEEE Std, 830-1998, Section 5) and customized to meet the need of SCSJ2203 course at Faculty of Computing, UTM. Compiled and checked by Shahida Sulaiman, PhD on 20 March 2016. Examples of models are from Satzinger (2011).

Table of Contents

1. Introduction

1.1 Purpose

This SRS describes the functional requirement accompanied with the use case description and the non-functional requirements for Master Dissertation Management System. The intended audience for the SRS is the stakeholders and the developers.

1.2 Scope

The scope of this project is to develop a management system for master's student dissertation that includes the following module:

- Thesis Submission Module
- Data management module
- Viva management module

All these modules will interact with each other in order to achieve the goal of making the product can providing the following functionalities:

- Allowing users to login the application with username and password
- Students are able to manage their thesis submission (upload, edit, delete submission)
- Supervisors are able to monitor the progress of their supervisee
- Supervisors are able to nominate examiners for student evaluation and viva
- Examiners are able to accept nomination request and evaluate students
- Admin is able to download, receive upload notifications and manage student data.

1.3 Definitions, Acronyms and Abbreviation

UC	–	Use Case
SRS	–	Software Requirement Specification
HTML	–	Hyper Text Markup Language
CSS	–	Cascading Style Sheet
UTM	–	Universiti Teknologi Malaysia
CRUD	–	Create, Read , Update, Delete Function

1.4 References

This subsection should:

a) Provide a complete list of all documents referenced elsewhere in the SRS;
b) Identify each document by title, report number (if applicable), date, and publishing organization;
c) Specify the sources from which the references can be obtained.

Specify complete list of references using a standardized reference format.

1.5 Overview

The Software Requirements Specification (SRS) document serves as a comprehensive guide that outlines the functional and non-functional requirements of a software system. It provides a detailed overview of the system's purpose, features, and architecture, while also specifying user interactions, system constraints, and performance objectives. By documenting the system's requirements, interfaces, and dependencies, the SRS acts as a foundation for the development team, stakeholders, and users, ensuring a clear understanding of the software's scope and functionality throughout the software development lifecycle.

2. Overall Description

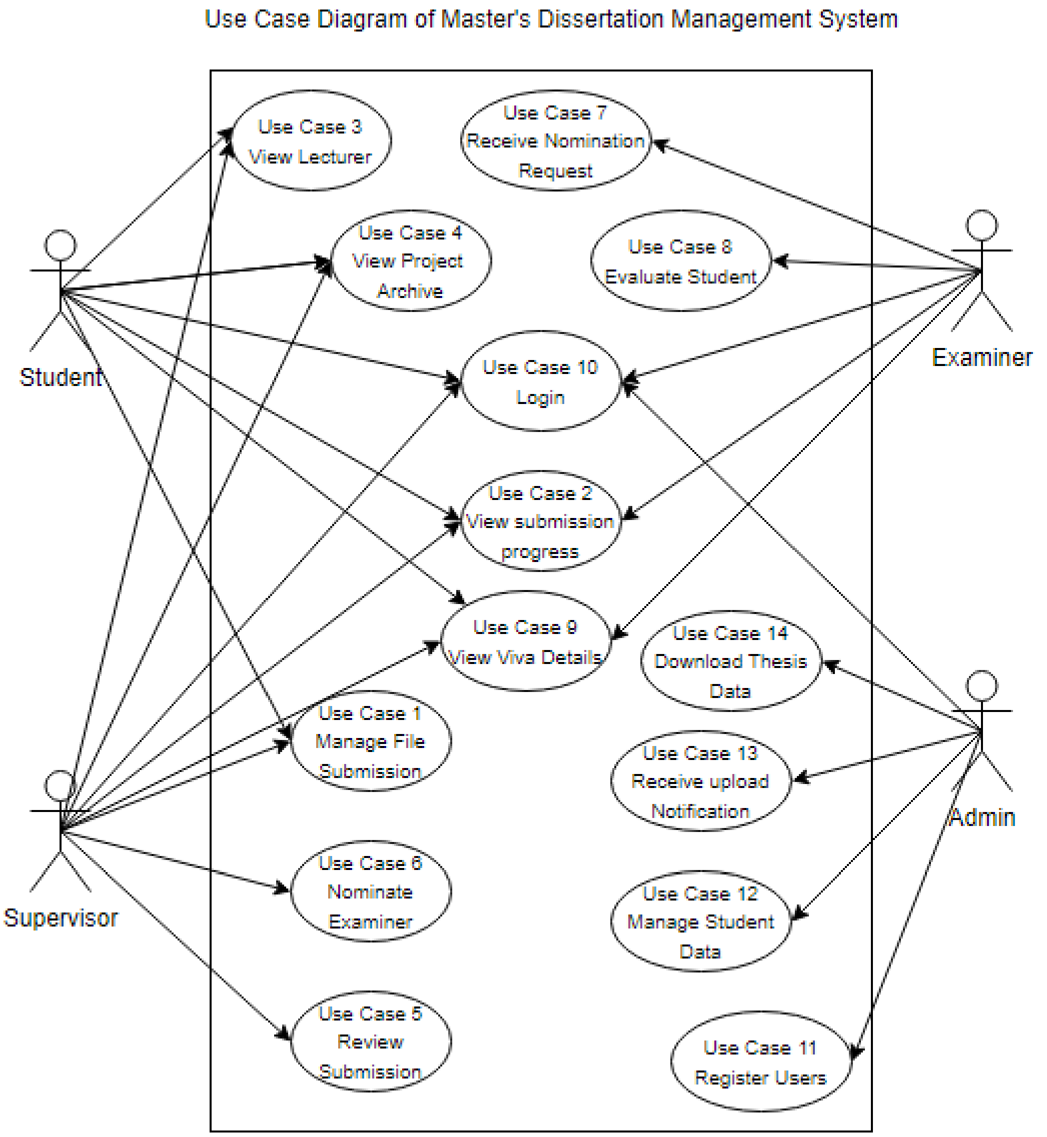


Figure 2.1: Use Case Diagram of Masters Dissertation Management System

2.1 Product Perspective

The web-based Masters Dissertation management system is a comprehensive platform that revolutionizes the management of the thesis process in educational institutions. By providing a centralized hub for students, supervisors, examiners, and administrators. It simplifies the entire lifecycle of thesis management. From proposal submission to progress tracking, and viva evaluation, the system enhances collaboration, streamlines workflows, and ensures efficient and transparent management of the thesis process, ultimately facilitating successful completion of academic projects.

2.1.1 System Interfaces

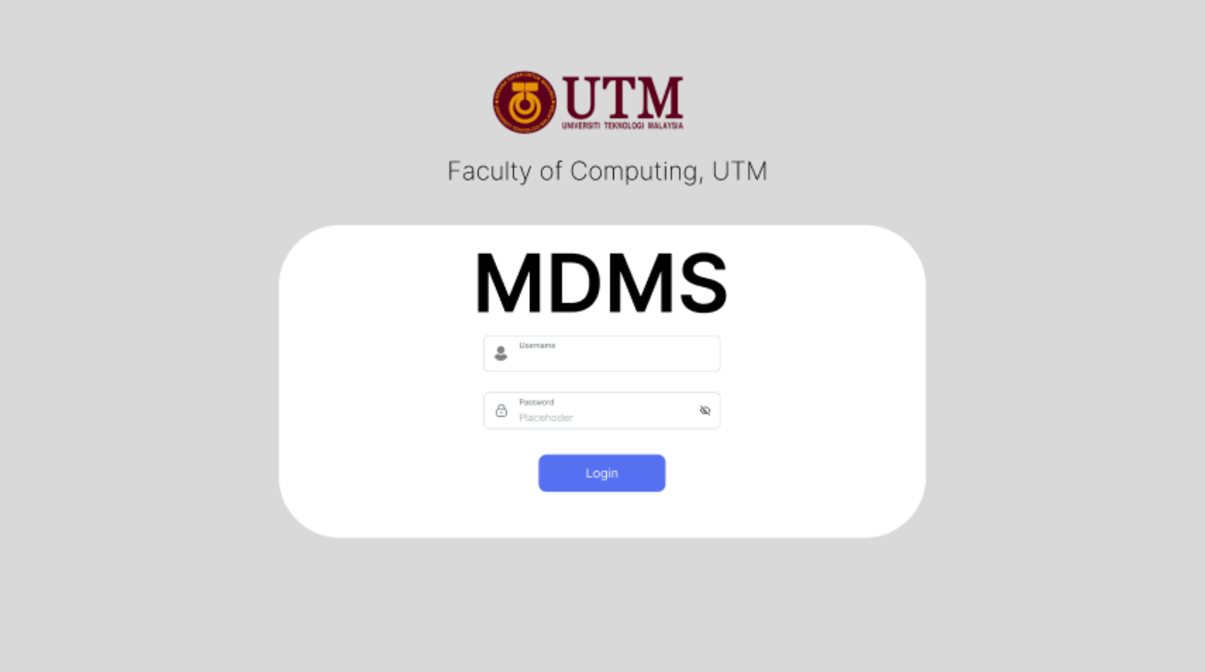
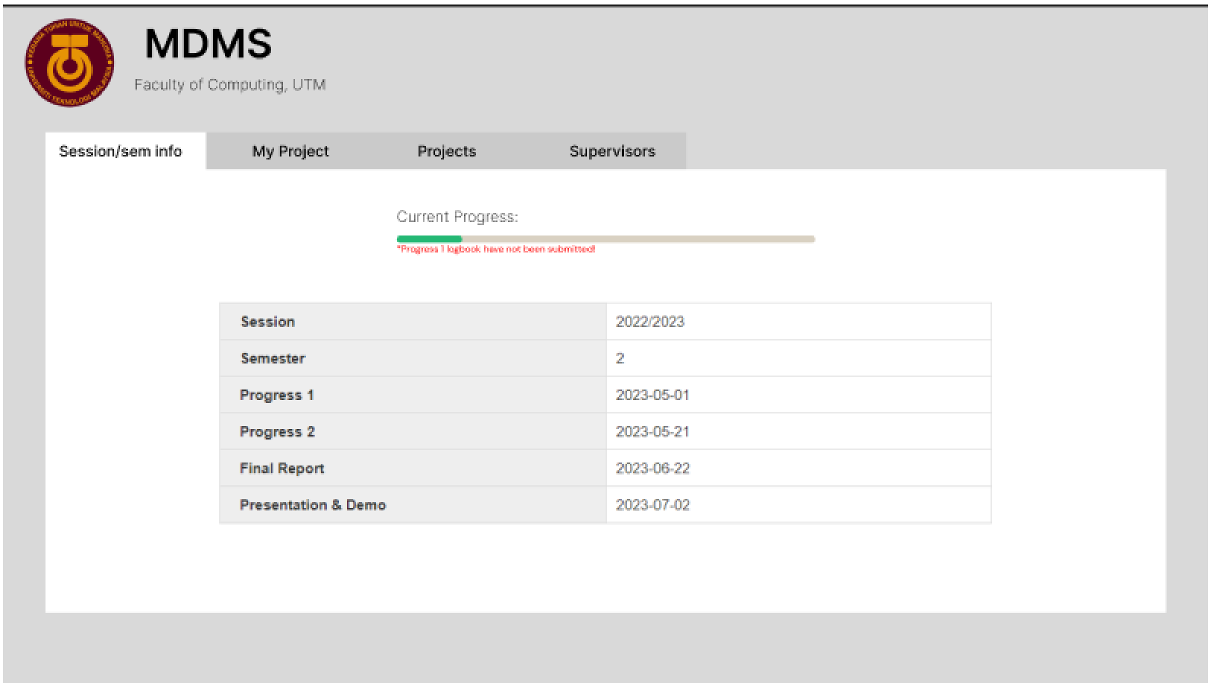


Figure 2.2 Login Menu



MDMS

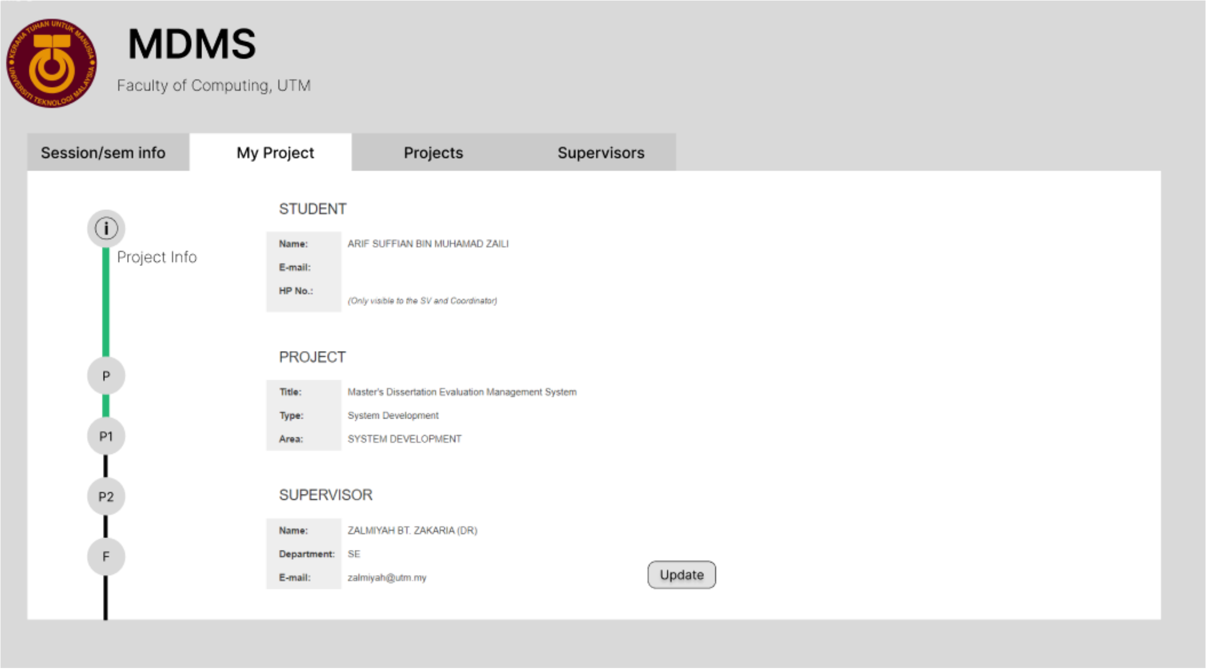
Faculty of Computing, UTM

Current Progress:

*Progress 1 logbook have not been submitted!

Session	2022/2023
Semester	2
Progress 1	2023-05-01
Progress 2	2023-05-21
Final Report	2023-06-22
Presentation & Demo	2023-07-02

Figure 2.3 View Submission Progress Page (Student Perspective)



MDMS

Faculty of Computing, UTM

STUDENT

Name: ARIF SUFFIAN BIN MUHAMAD ZAILI
E-mail:
HP No.:
(Only visible to the SV and Coordinator)

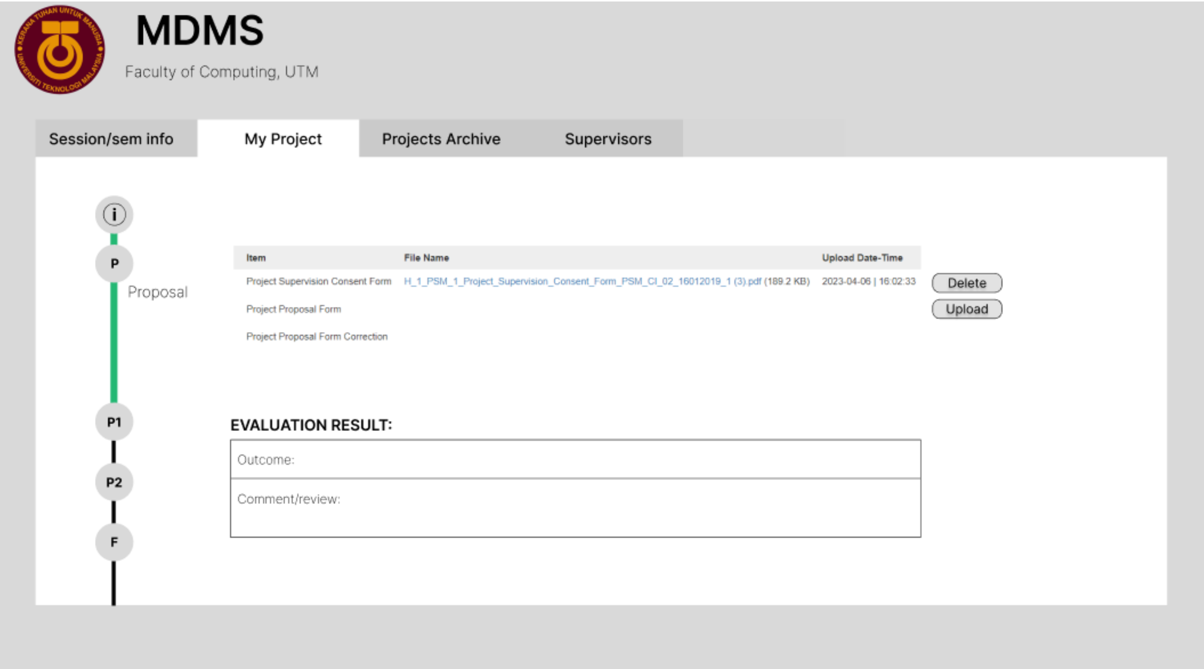
PROJECT

Title: Master's Dissertation Evaluation Management System
Type: System Development
Area: SYSTEM DEVELOPMENT

SUPERVISOR

Name: ZALMIYAH BT. ZAKARIA (DR)
Department: SE
E-mail: zalmiyah@utm.my

Figure 2.4 Manage File Submission page and Review Submission



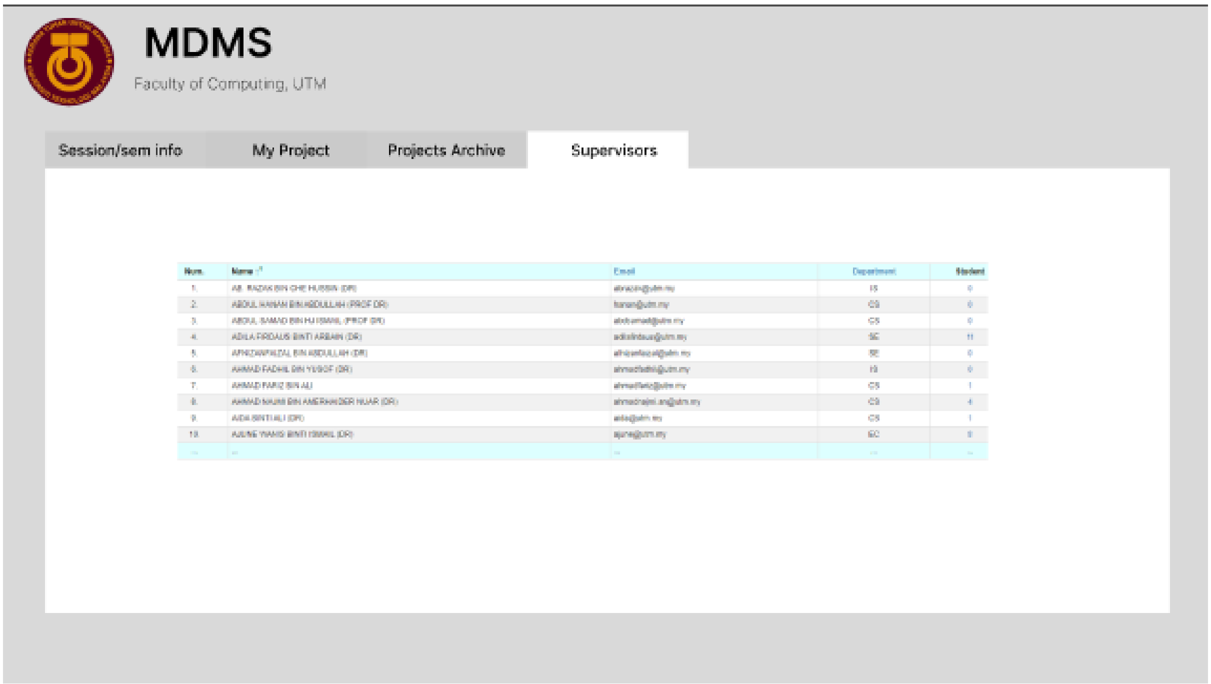
Item	File Name	Upload Date-Time
Project Supervision Consent Form	H_1_PSM_1_Project_Supervision_Consent_Form_PSM_CI_02_16012019_1 (3).pdf (189.2 KB)	2023-04-06 | 16:02:33
Project Proposal Form		
Project Proposal Form Correction		

EVALUATION RESULT:

Outcome:

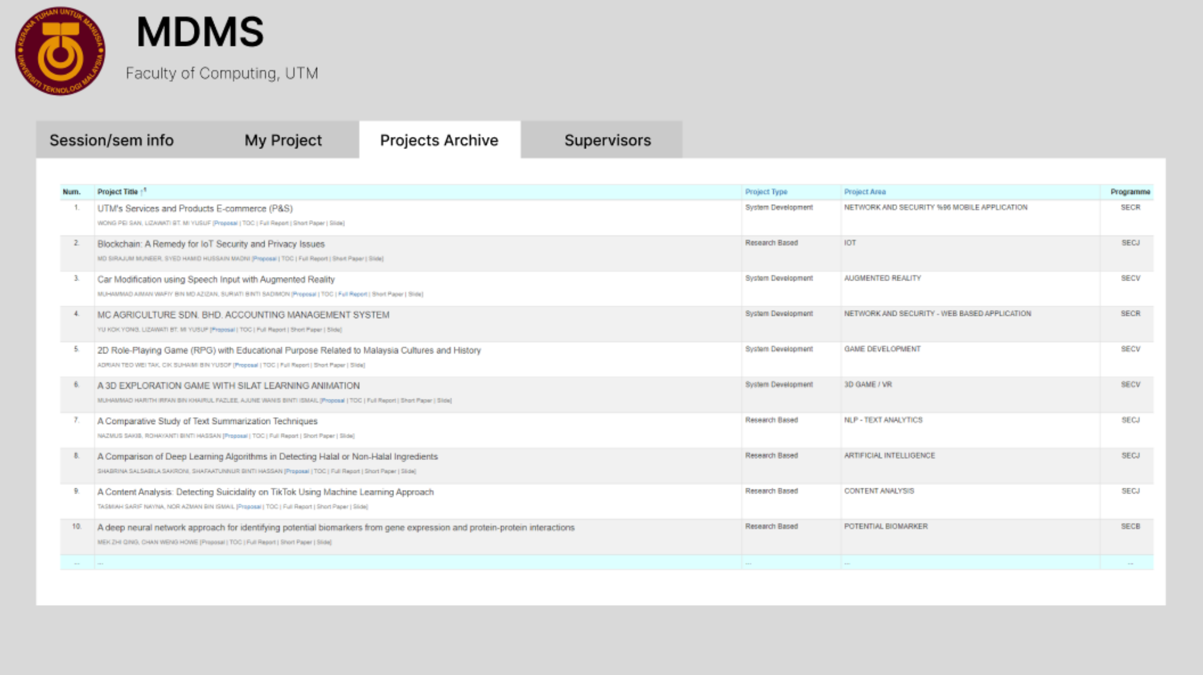
Comment/review:

Figure 2.5 Manage File Submission page and Review Submission



Num.	Name	Email	Department	Student
1.	AB. RAZAK BIN CHE HUSSIN (DR)	abrazak@utm.my	IS	0
2.	ABDUL HANAN BIN ABDULLAH (PROF DR)	hanan@utm.my	CS	0
3.	ABDUL SAMAD BIN HJ ISMAIL (PROF DR)	abdsamad@utm.my	CS	0
4.	ADILA FIRDAUS BINTI ARBAIN (DR)	adilafirdaus@utm.my	SE	11
5.	AFNIZANFAIZAL BIN ABDULLAH (DR)	afnizanfauzi@utm.my	SE	0
6.	AHMAD FADHIL BIN YUSOF (DR)	ahmadfadhil@utm.my	IS	0
7.	AHMAD FARIZ BIN ALI	ahmadfariz@utm.my	CS	1
8.	AHMAD NAJMI BIN AMERHAIDER NUAR (DR)	ahmadnajmi.an@utm.my	CS	4
9.	AIDA BINTI ALI (DR)	aida@utm.my	CS	1
10.	AJUNE WANIS BINTI ISMAIL (DR)	ajune@utm.my	SC	0
...	...	...	...	...

Figure 2.6 View Lecturer page



Num.	Project Title ↑¹	Project Type	Project Area	Programme
1.	UTM's Services and Products E-commerce (P&S) WONG PEI SAN, LIZAWATI BT. MI YUSUF [Proposal | TOC | Full Report | Short Paper | Slide]	System Development	NETWORK AND SECURITY %96 MOBILE APPLICATION	SECR
2.	Blockchain: A Remedy for IoT Security and Privacy Issues MD SIRAJUM MUNEER, SYED HAMID HUSSAIN MADNI [Proposal | TOC | Full Report | Short Paper | Slide]	Research Based	IOT	SECJ
3.	Car Modification using Speech Input with Augmented Reality MUHAMMAD AIMAN WAFIY BIN MD AZIZAN, SURIATI BINTI SADIMON [Proposal | TOC | Full Report | Short Paper | Slide]	System Development	AUGMENTED REALITY	SECV
4.	MC AGRICULTURE SDN. BHD. ACCOUNTING MANAGEMENT SYSTEM YU KOK YONG, LIZAWATI BT. MI YUSUF [Proposal | TOC | Full Report | Short Paper | Slide]	System Development	NETWORK AND SECURITY - WEB BASED APPLICATION	SECR
5.	2D Role-Playing Game (RPG) with Educational Purpose Related to Malaysia Cultures and History ADRIAN TEO WEI TAK, CIK SUHAIMI BIN YUSOF [Proposal | TOC | Full Report | Short Paper | Slide]	System Development	GAME DEVELOPMENT	SECV
6.	A 3D EXPLORATION GAME WITH SILAT LEARNING ANIMATION MUHAMMAD HARITH IRFAN BIN KHAIRUL FAZLEE, AJUNE WANIS BINTI ISMAIL [Proposal | TOC | Full Report | Short Paper | Slide]	System Development	3D GAME / VR	SECV
7.	A Comparative Study of Text Summarization Techniques NAZMUS SAKIB, ROHAYANTI BINTI HASSAN [Proposal | TOC | Full Report | Short Paper | Slide]	Research Based	NLP - TEXT ANALYTICS	SECJ
8.	A Comparison of Deep Learning Algorithms in Detecting Halal or Non-Halal Ingredients SHABRINA SALSABILA SAKRONI, SHAFAATUNNUR BINTI HASSAN [Proposal | TOC | Full Report | Short Paper | Slide]	Research Based	ARTIFICIAL INTELLIGENCE	SECJ
9.	A Content Analysis: Detecting Suicidality on TikTok Using Machine Learning Approach TASMIAH SARIF NAYNA, NOR AZMAN BIN ISMAIL [Proposal | TOC | Full Report | Short Paper | Slide]	Research Based	CONTENT ANALYSIS	SECJ
10.	A deep neural network approach for identifying potential biomarkers from gene expression and protein-protein interactions MEK ZHI QING, CHAN WENG HOWE [Proposal | TOC | Full Report | Short Paper | Slide]	Research Based	POTENTIAL BIOMARKER	SECB
---	---	---	---	---

Figure 2.7 View Project Archive

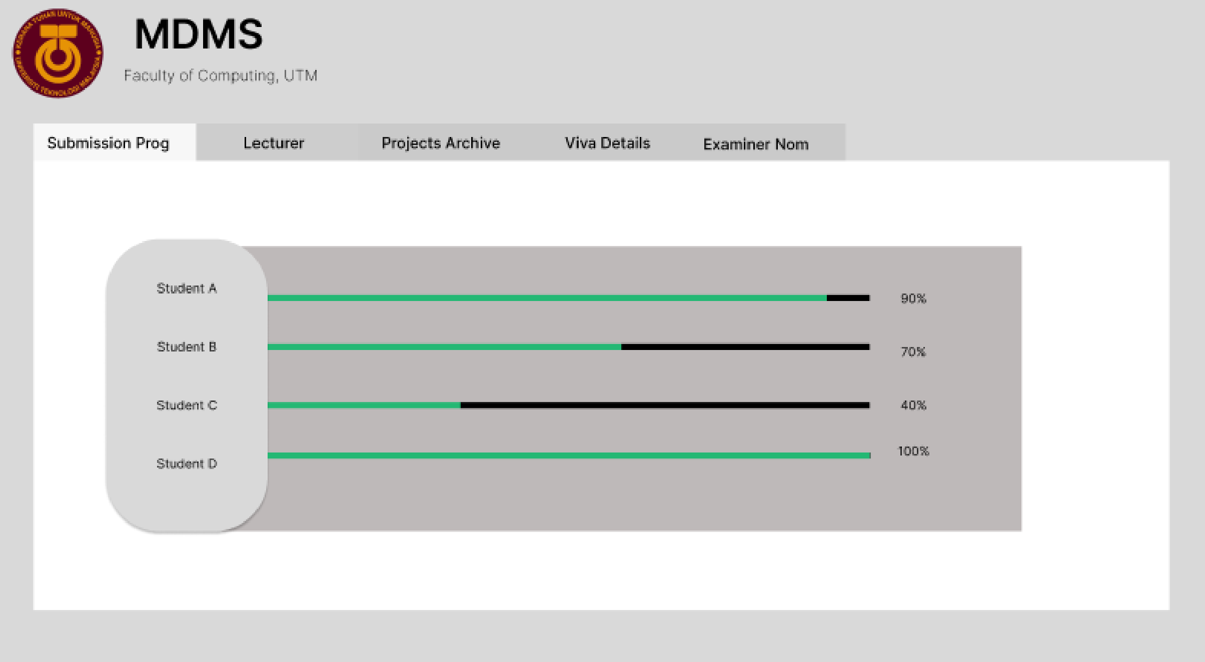
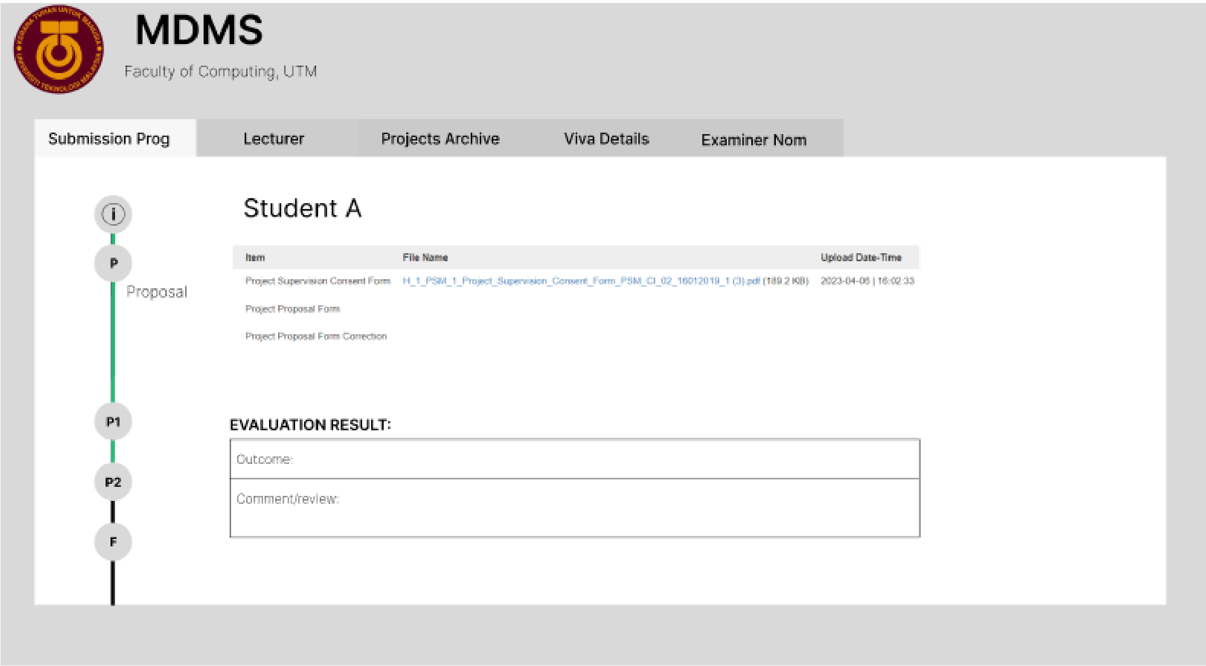


Figure 2.8 View Submission Progress (supervisor Perspective)



Student A

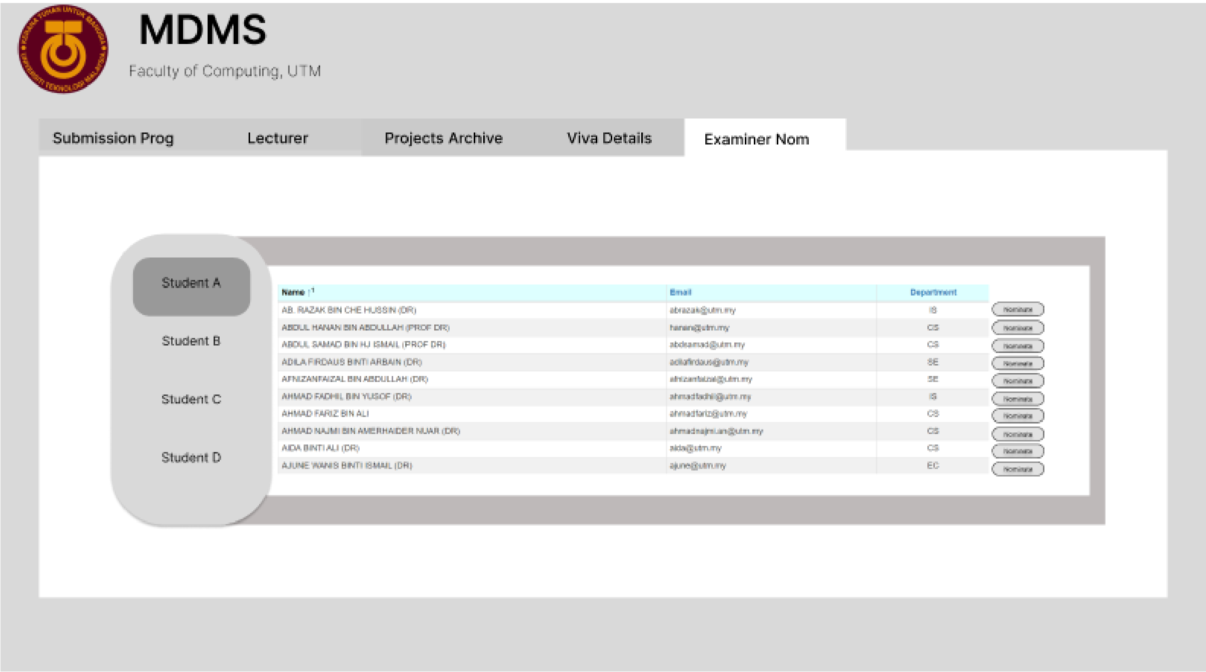
Item	File Name	Upload Date-Time
Project Supervision Consent Form	H_1_PSM_1_Project_Supervision_Consent_Form_PSM_CI_02_16012019_1 (3).pdf (189.2 KB)	2023-04-06 | 16:02:33
Project Proposal Form		
Project Proposal Form Correction		

EVALUATION RESULT:

Outcome:

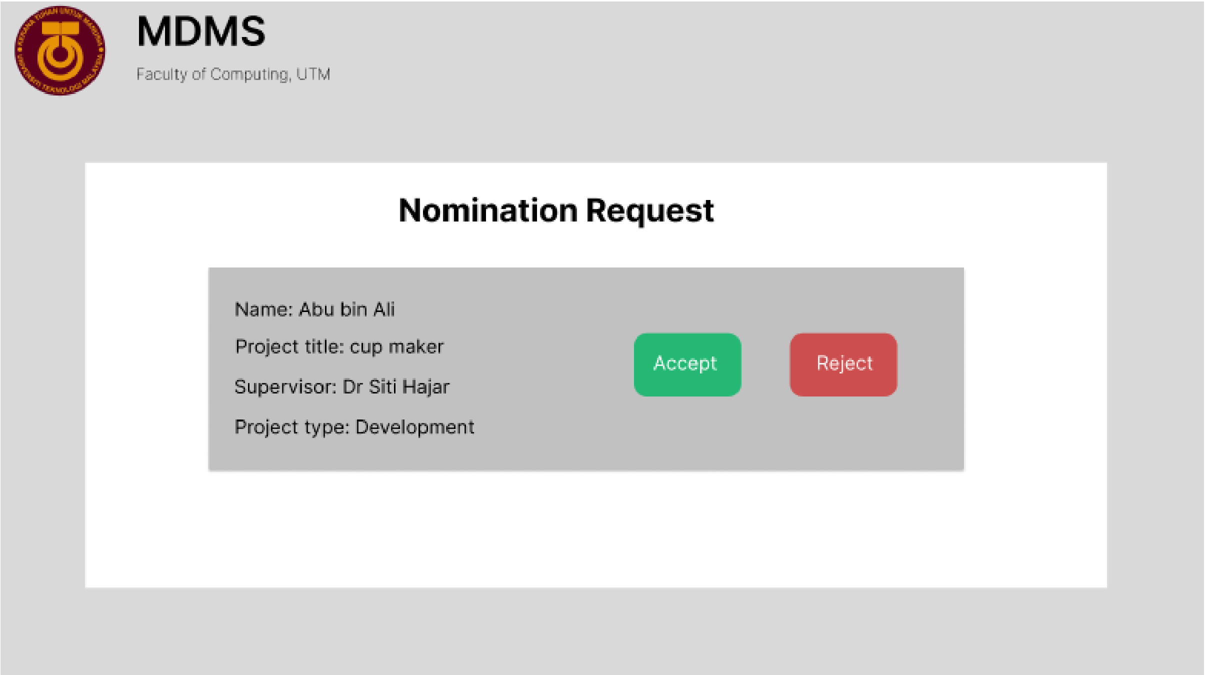
Comment/review:

Figure 2.9 View Submission Progress (supervisor Perspective)



Name ↑1	Email	Department	
AB. RAZAK BIN CHE HUSSIN (DR)	abrazak@utm.my	IS	Nominate
ABDUL HANAN BIN ABDULLAH (PROF DR)	hanan@utm.my	CS	Nominate
ABDUL SAMAD BIN HJ ISMAIL (PROF DR)	abdsamad@utm.my	CS	Nominate
ADILA FIRDAUS BINTI ARBAIN (DR)	adilafirdaus@utm.my	SE	Nominate
AFNIZANFAIZAL BIN ABDULLAH (DR)	afnizanfaizal@utm.my	SE	Nominate
AHMAD FADHIL BIN YUSOF (DR)	ahmadfadhil@utm.my	IS	Nominate
AHMAD FARIZ BIN ALI	ahmadfariz@utm.my	CS	Nominate
AHMAD NAJMI BIN AMERHAIDER NUAR (DR)	ahmadnajmi.an@utm.my	CS	Nominate
AIDA BINTI ALI (DR)	aida@utm.my	CS	Nominate
AJUNE WANIS BINTI ISMAIL (DR)	ajune@utm.my	EC	Nominate

Figure 2.10 Nominate Examiner



Nomination Request

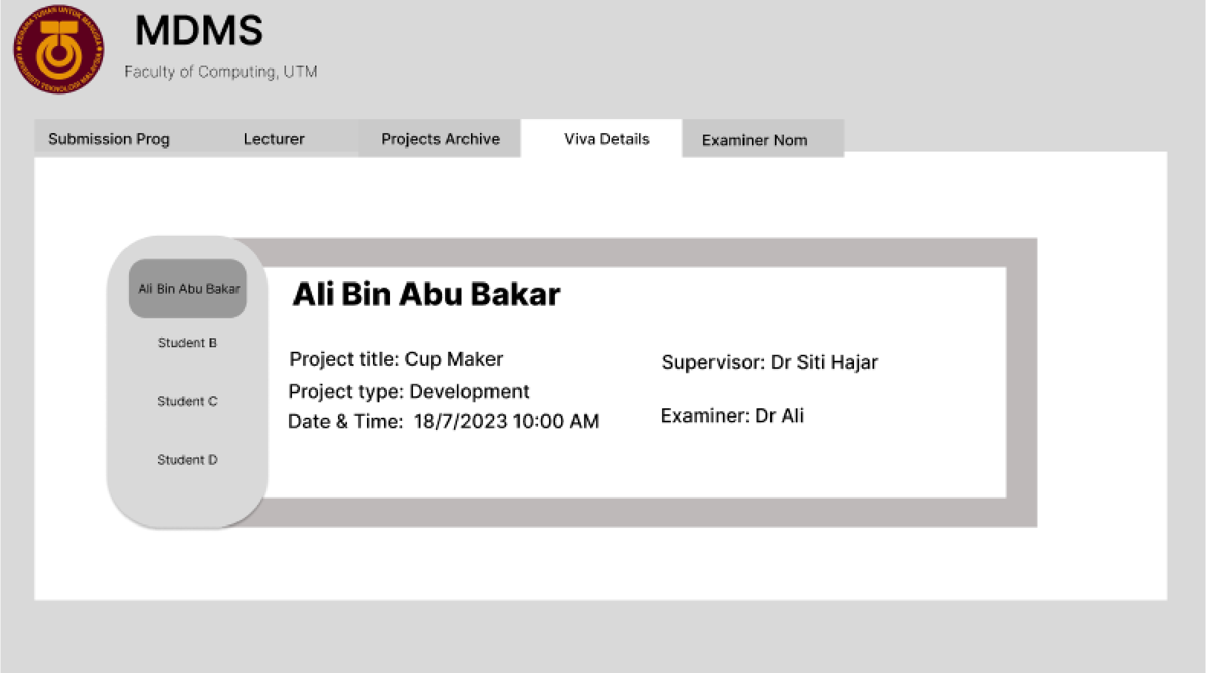
Name: Abu bin Ali

Project title: cup maker

Supervisor: Dr Siti Hajar

Project type: Development

Figure 2.11:Receive Nomination Request



Ali Bin Abu Bakar

Project title: Cup Maker

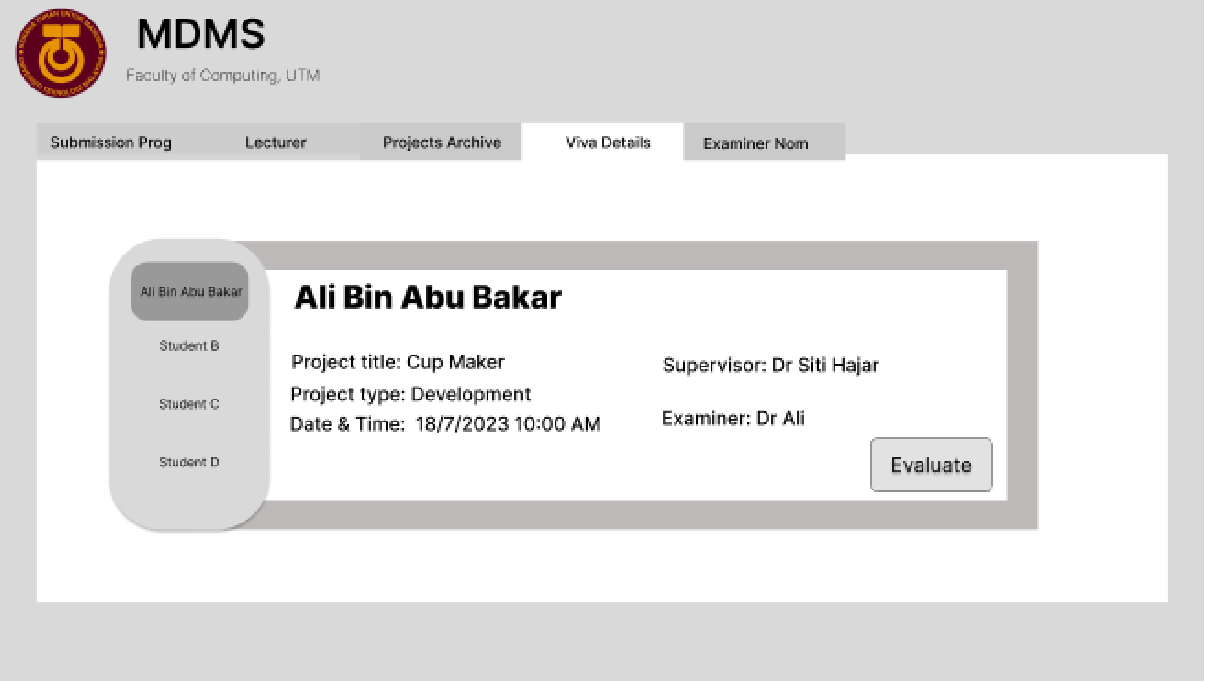
Project type: Development

Date & Time: 18/7/2023 10:00 AM

Supervisor: Dr Siti Hajar

Examiner: Dr Ali

Figure 2.12: View Viva Details



Ali Bin Abu Bakar

Project title: Cup Maker

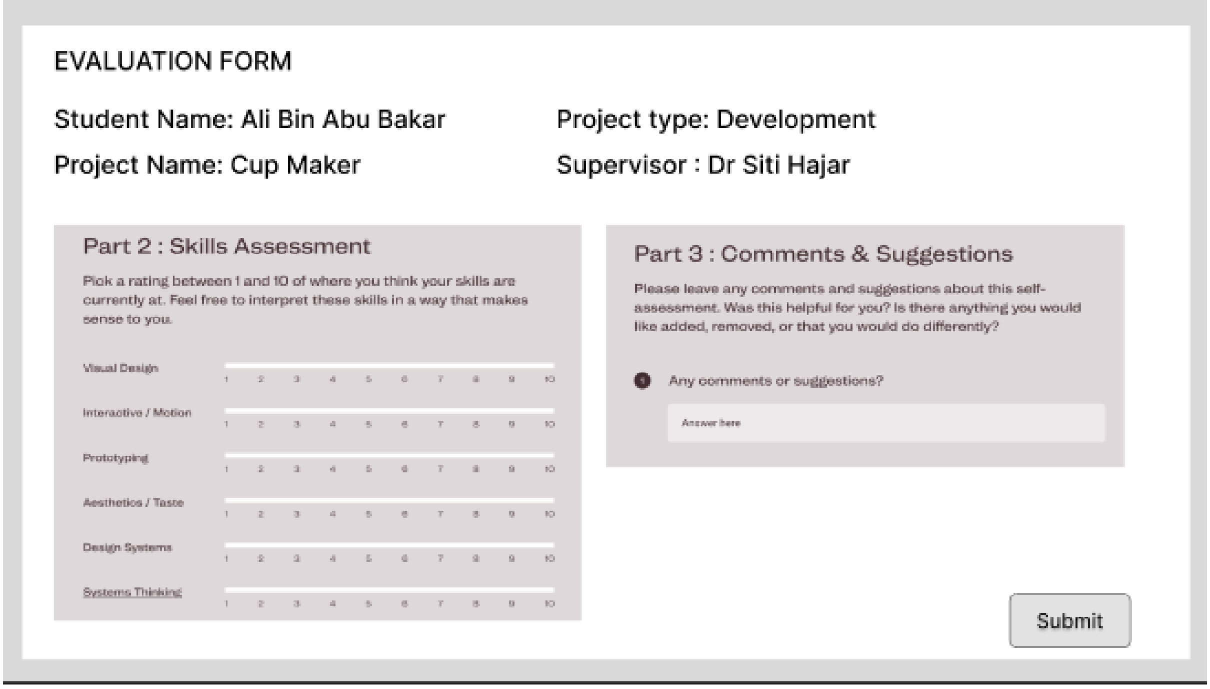
Project type: Development

Date & Time: 18/7/2023 10:00 AM

Supervisor: Dr Siti Hajar

Examiner: Dr Ali

Figure 2.13 : View Viva Details (Examiner Perspective)



EVALUATION FORM

Student Name: Ali Bin Abu Bakar

Project Name: Cup Maker

Project type: Development

Supervisor : Dr Siti Hajar

Part 2 : Skills Assessment

Pick a rating between 1 and 10 of where you think your skills are currently at. Feel free to interpret these skills in a way that makes sense to you.

Visual Design 1 2 3 4 5 6 7 8 9 10

Interactive / Motion 1 2 3 4 5 6 7 8 9 10

Prototyping 1 2 3 4 5 6 7 8 9 10

Aesthetics / Taste 1 2 3 4 5 6 7 8 9 10

Design Systems 1 2 3 4 5 6 7 8 9 10

Systems Thinking 1 2 3 4 5 6 7 8 9 10

Part 3 : Comments & Suggestions

Please leave any comments and suggestions about this self-assessment. Was this helpful for you? Is there anything you would like added, removed, or that you would do differently?

1 Any comments or suggestions?

Answer here

Figure 2.14 : Evaluate Form

2.1.2 User Interfaces

The user will interact with the system via desktop or laptop screen while using keyboard and mouse. For handheld devices such as mobile phones and tablets, the user interacts via touch screen functionality

2.1.3 Hardware Interfaces

This system is a web-based application where the user can access from a desktop or mobile device. The user interface will be displayed for interaction in browsers of all devices that are able to access the system.

2.1.4 Software Interfaces

Node JS framework would be the software interface for the system. This framework is designed to create scalable network applications using an event-driven, non-blocking I/O approach, which makes it quick and resource-efficient. It enables developers to use a single programming language (JavaScript) on both the server-side and client-side. Inevitably, this increases the speed and efficiency with which programmes may run, while cutting their development and maintenance costs.

2.1.5 Communication Interfaces

There are 2 ways of communication. Firstly, would be the communication between the user and the system itself. Secondly, The structure's many components will interact with one another, which is required because they are dependent on one another. For web-based apps, the underlying operating systems handle most of the communication.

2.1.6 Memory

There is no minimum requirement of primary or secondary memory to run the system. However, the use of browser is essential to the system thus the minimum memory requirement to run a browser must be met.

2.1.7 Operations

The system should be easy to use for all users. Also, there are no need for additional software and tools to use the system besides the operating system. No special technical skills required to interact with the system tools.

2.1.8 Site Adaptation Requirements

There are no specific size or mission related features that should be modified to adapt to this software.

2.2 Product Functions

Use Case	Summary
UC 001: Manage File Submission	This use case consists of all CRUD function of file submission.
UC 002: View Submission Progress	This use case consists of the function where the appropriate users can view submission progress of the thesis.
UC 003: View Lecturer	This use case consists of function where lecturer list will be displayed.
UC 004: View Project Archive	This use case consists of function where the system will display all project archives.
UC 005: Review Submission	This use case consists of a feature whereby the supervisor can provide feedback and review for the submissions.
UC 006: Nominate Examiner	This use case consists of a feature whereby the supervisor nominates the examiner for viva.
UC 007: Receive Nomination Request	This use case consists of a feature whereby the examiner receives and answers the nomination request for viva.
UC 008: Evaluate Student	This use case consists of a feature whereby the examiner evaluates the student for their work and viva.

UC 009: View Viva Details	This use case describes the scenario where the appropriate users will get to view the details of a viva session.
UC 010: Login	This use case consists of the login functionality for all users
UC 011: Register Users	This use case consists of scenario where by the admin register all users directly to the database.
UC 012: Manage Student Data	This use case consists of scenario whereby the admin can edit and delete student data.
UC 013: Receive Upload Notification	This use case describes the scenario where the admin receives notification whenever a submission is uploaded to the database.
UC 014: Download Thesis Data	This use case describes the scenario where the admin downloads the thesis data from the database.

2.3 User Characteristics

Actor	Description
Student	The student would be the candidates that have access to view most information related function such as view submission progress, lecturer, project archives and manage file submissions
Supervisor	The supervisor would have all related task to assist their students by having accessibility to all view functionality and the privilege to nominate an examiner for the student viva.
Examiner	The examiner would have their core responsibility which is within viva related functionality such as view viva details, accept nomination request and evaluate students
Admin	The admin will have privilege over all data related functionality such as manage student and thesis data, receive upload notification and registration.

2.4 Constraints

This system maintains the data every day. The system viewed by the desktop at its workstation, which is in the office or from home.

2.5 Assumption and Dependencies

The system will not work as intended if there is no internet connection.

The system must be active for 24 hours.

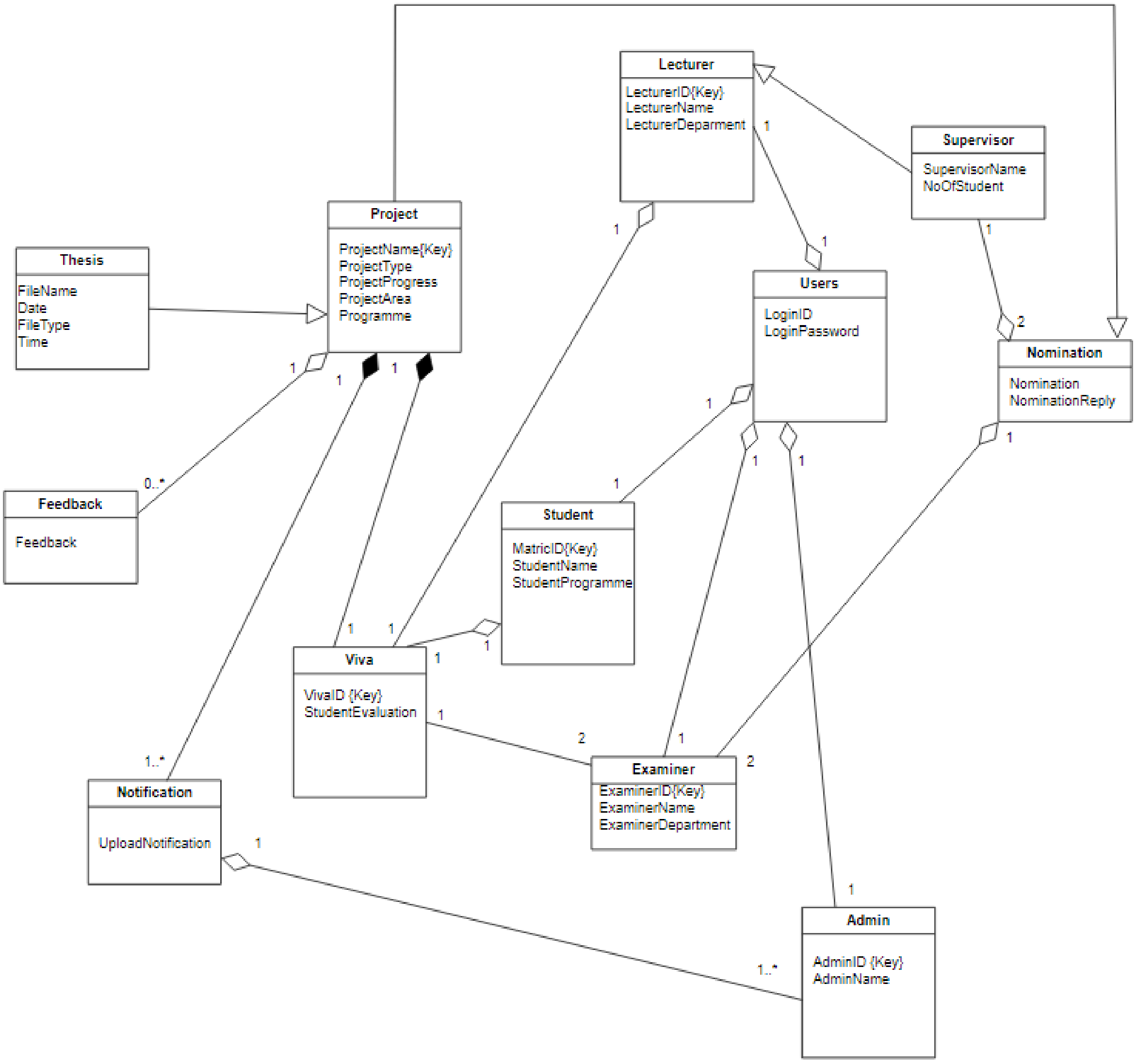


Figure 3.1: Domain Model of <Master's Dissertation Management System>

2.6 External Interface Requirements

2.6.1 User Interfaces

Provide the details for Section 2.1.2.

2.6.2 Hardware Interfaces

Provide the details for Section 2.1.3.

2.6.3 Software Interfaces

Provide the details for Section 2.1.4.

2.6.4 Communication Interfaces

Provide the details for Section 2.1.5.

2.7 System Features

2.7.1 Module <Thesis Management Module>

This module consists of 5 use cases which all of them are related with thesis related functionality.

- Manage File Submission
- View Lecturer
- View Submission Progress
- View Project Archive
- Review Submission

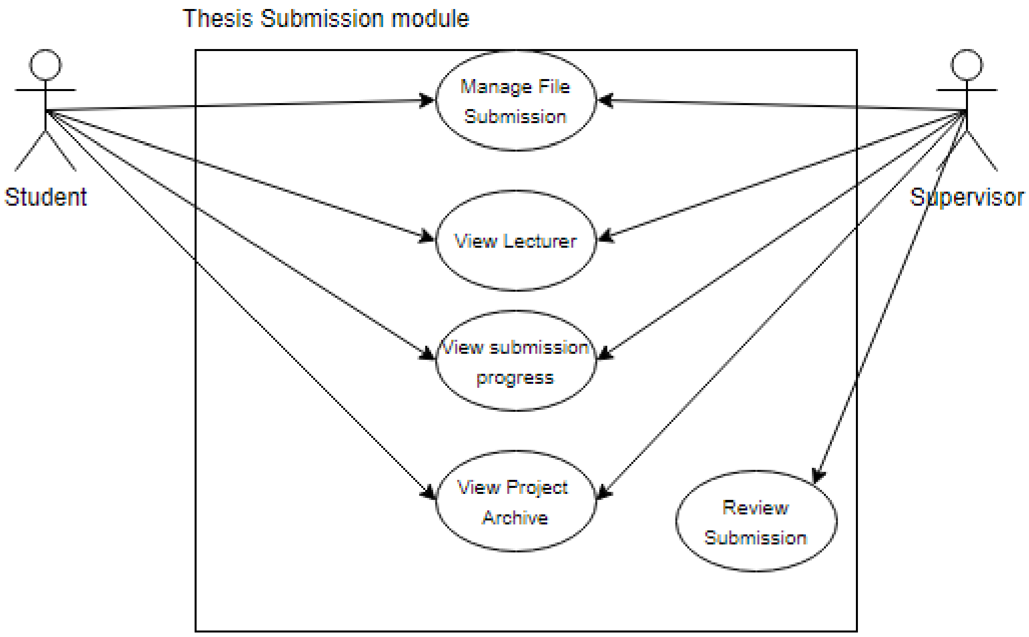


Figure 3.2: < Thesis Management Module >

2.7.1.1 UC001: Use Case 001 <Manage File Submissions>

Table 3.1: Use Case Description for <Use Case 001: Manage File Submissions>

Use Case Name	Manage File Submissions
Brief Description	functionality of managing file submissions within a system. It allows users to submit files, view submitted files, update file information, and delete files when necessary. This use case is particularly relevant in systems that require users to upload and manage various types of files or documents.
Actors	Student
Preconditions	• The user is authenticated and authorized to access the file submission functionality. • The system is operational and capable of handling file uploads and storage.
Post Conditions	• The user can successfully submit, view, update,

	and delete files as required. • The system maintains the integrity of the submitted files and their associated metadata.
Basic Flow	**User Submits File:** 1. The user initiates the file submission process. 2. The system presents a file upload interface or prompts the user to select a file from their device. 3. The user selects the file and uploads it to the system. 4. The system validates the file, ensuring it meets any required criteria or constraints. 5. If the file passes validation, the system saves it in the appropriate location or storage system. **User Updates File Information:** 1. The user selects a specific file from the list of submitted files. 2. The system retrieves the file's details and presents them to the user for editing. 3. The user updates the desired information, such as file name, description, or tags. 4. The system validates the changes and updates the file's metadata accordingly. **User Deletes File:** 1. The user selects a file from the list of submitted files. 2. The system presents a confirmation dialog to ensure the user's intention to delete the file. 3. If the user confirms deletion, the system removes the file from the storage system and deletes its

	associated metadata.
Alternative Flow	• If the file fails validation during the submission process (e.g., file format not supported, size exceeds limits), the system displays an error message and prompts the user to submit a valid file.
Exception Condition	• If there are no submitted files available for the user to view, the system displays an appropriate message indicating the absence of files. • In case of any system errors or technical issues, the system notifies the user and provides guidance on how to proceed.

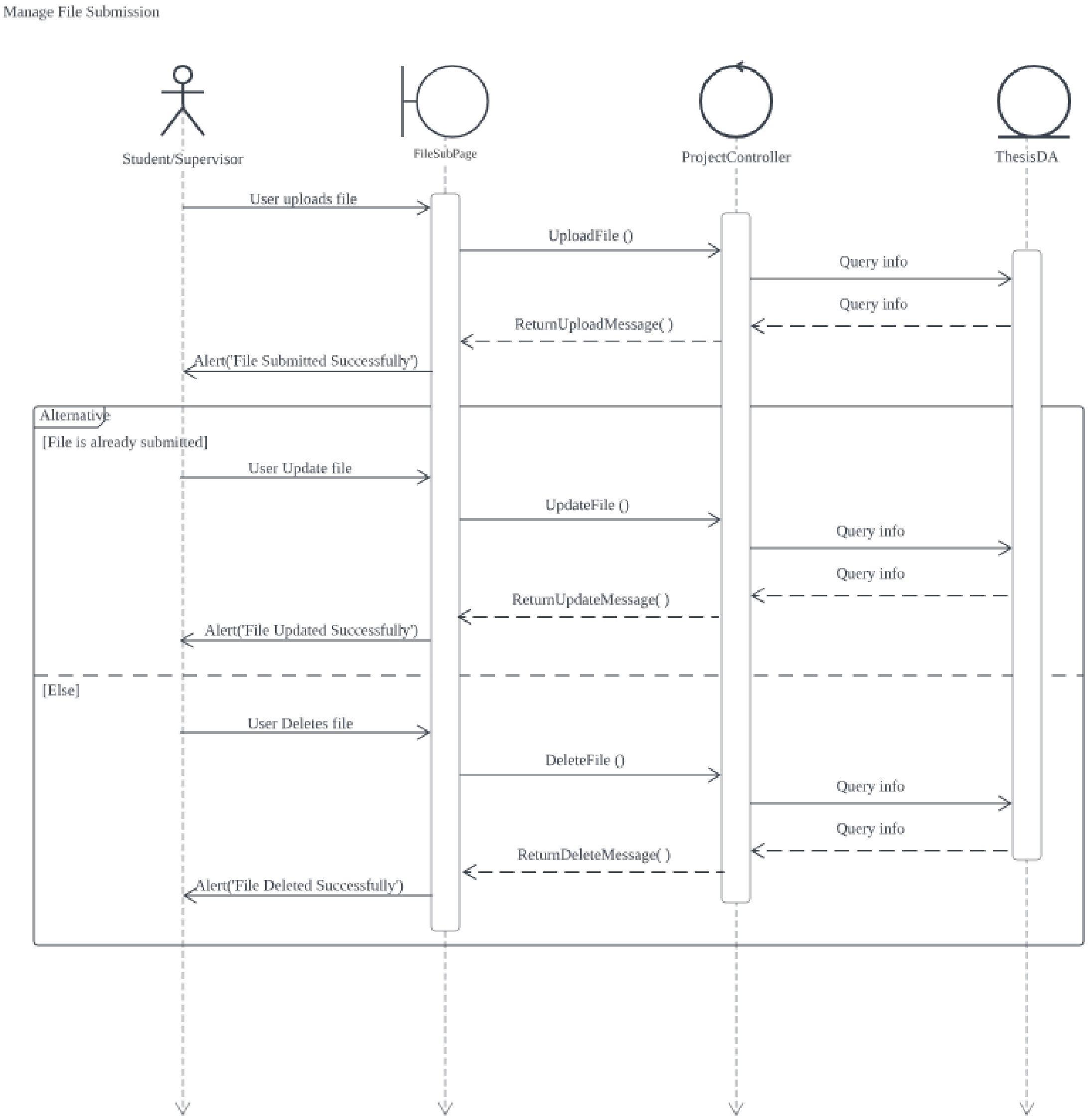


Figure 3.3: System Sequence Diagram of < Manage File Submissions >

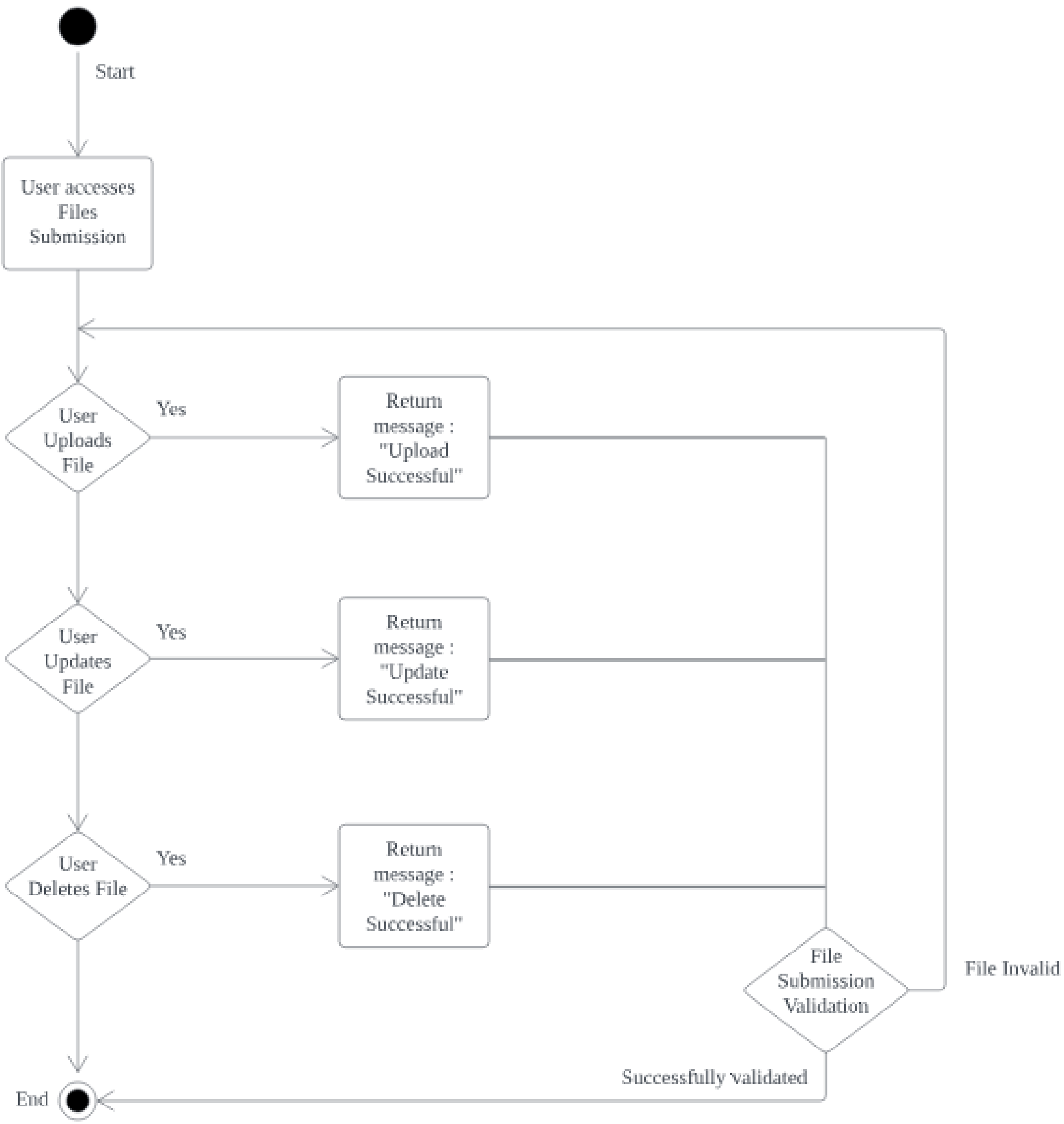

Start
User accesses Files Submission
User Uploads File
Yes
Return message : "Upload Successful"
User Updates File
Yes
Return message : "Update Successful"
User Deletes File
Yes
Return message : "Delete Successful"
File Submission Validation
File Invalid
Successfully validated
End

Figure 3.4: Activity Diagram of < Manage File Submissions >

2.7.1.2 UC002: Use Case 002: <View Submission Progress>

Use Case Name	View Submission Progress
Brief Description	The use case involves the functionality of providing feedback on a submission. It allows authorized users, such as supervisors, to review the content.
Actors	Supervisors
Preconditions	• The examiner or advisor is authenticated and authorized to access the system. • The management system is operational and capable of handling thesis review and feedback processes.
Post Conditions	• The system may trigger subsequent actions
Basic Flow	**Supervisor Accesses File Submissions for Review:** 1. The supervisor logs into the system and accesses the assigned thesis for review. 2. The system presents the thesis document or relevant details for the examiner to review. **Supervisor Provides Feedback:** 1. The supervisor records their feedback, comments, and suggestions regarding the thesis. 2. The system may provide input fields, comment boxes, or other means to capture the examiner's feedback.
Alternative Flow	• The System may incorporate a deadline mechanism to ensure timely submission of the examiner’s review and feedback.
Exception Condition	• Error in submitting feedback due to connection error may cause the system to initiate an error message

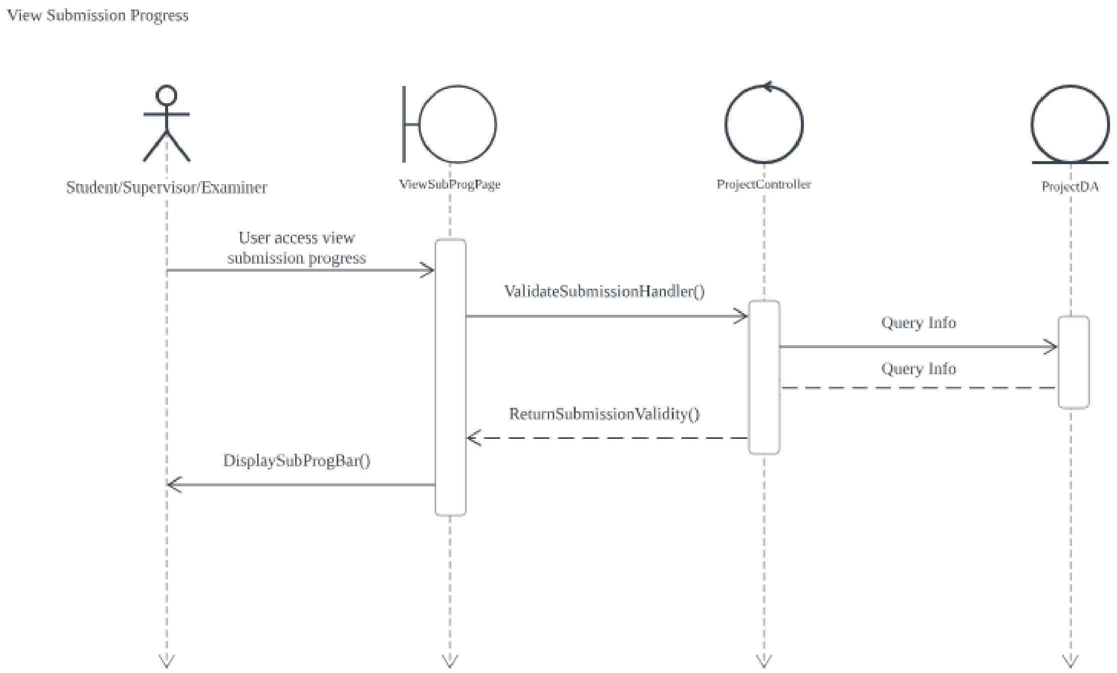

View Submission Progress
Student/Supervisor/Examiner
ViewSubProgPage
ProjectController
ProjectDA
User access view submission progress
ValidateSubmissionHandler()
Query Info
Query Info
ReturnSubmissionValidity()
DisplaySubProgBar()

Figure 3.3: System Sequence Diagram of < View Submission Progress >

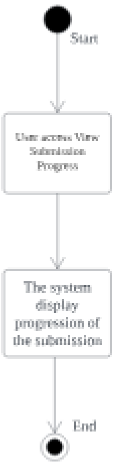

Start
User access View Submission Progress
The system display progression of the submission
End

Figure 3.4: Activity Diagram of < View Submission Progress >

2.7.1.3 UC003: Use Case 003 <View Lecturer>

Use Case Name	View Lecturer
Brief Description	The use case involves the functionality of accessing and viewing a list of available lecturers within the system. It allows users, such as students or administrators, to obtain information about lecturers who are available to supervise or advise on their thesis.
Actors	Students
Preconditions	• The student or administrator is authenticated and authorized to access the system. • The system is operational and capable of storing and presenting lecturer information.
Post Conditions	• The student or administrator gains access to a comprehensive list of available lecturers within the system. • The system provides accurate and up-to-date information about the lecturers, allowing users to make informed decisions when selecting a supervisor for their thesis
Basic Flow	**User Requests Lecturer List** 1. The student or administrator accesses the "View Lecturer List" functionality within the management system. **System Presents Lecturer List** 1. The system displays the list of lecturers based on the user's request and any predefined criteria. 2. The lecturer list may be presented in a tabular format or a visually organized interface, allowing users to browse and find individual lecturer.

Alternative Flow	None
Exception Condition	- If there are no available lecturers in the system, the system may display an appropriate message indicating the absence of any lecturers or advisors.

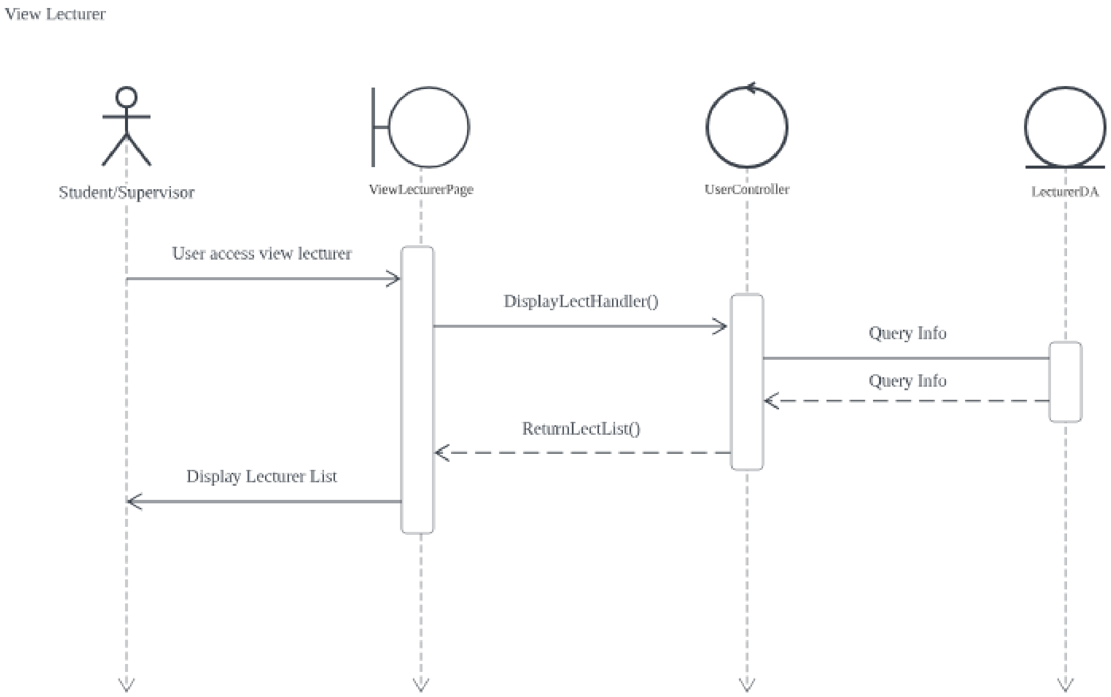


Figure 3.3: System Sequence Diagram of < View Lecturer >

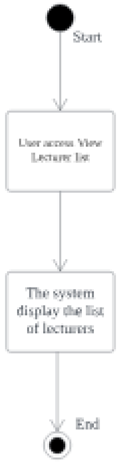

Start
User access View Lecturer list
The system display the list of lecturers
End

Figure 3.4: Activity Diagram of < View Lecturer >

2.7.1.4 UC004: Use Case 004 <View Project Archive>

Use Case Name	View Project Archive
Brief Description	The use case involves the functionality of accessing and viewing a list of project archive within the system. It allows users, such as students or administrators, to obtain information about other thesis titles that is available to view.
Actors	Students, Supervisors
Preconditions	• The student or administrator is authenticated and authorized to access the system. • The system is operational and capable of storing and presenting thesis information.
Post Conditions	• The student or administrator gains access to a comprehensive list of available thesis within the system. • The system provides accurate and up-to-date information about the thesis, allowing users to make informed decisions when selecting a supervisor for their thesis.
Basic Flow	**User Requests Project Archive** 1. The student or administrator accesses the "Project Archive" functionality within the management system. **System Presents Project Archive** 1. The system displays the list of thesis based on the user's request and any predefined criteria. 2. The lecturer list may be presented in a tabular format or a visually organized interface, allowing

	users to browse and find specific thesis.
Alternative Flow	None
Exception Condition	• If there are no available lecturers in the system, the system may display an appropriate message indicating the absence of any lecturers or advisors.

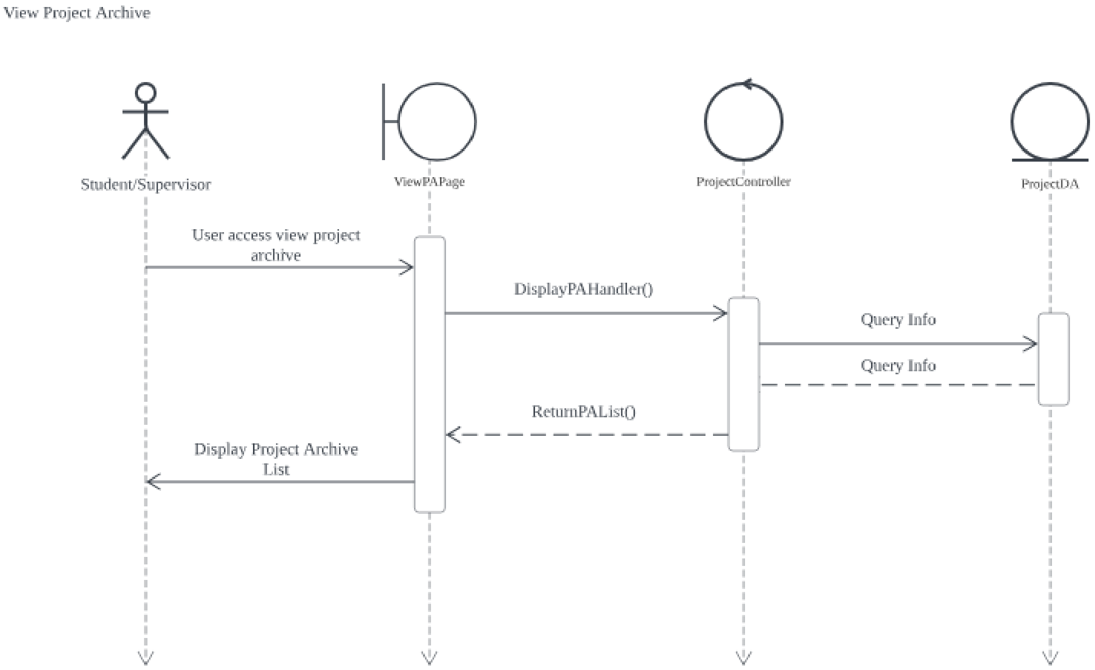


Figure 3.3: System Sequence Diagram of < View Project Archive >

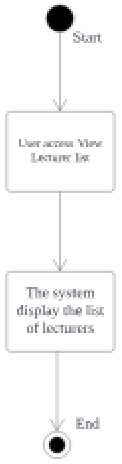

Start
User access View Lecturer list
The system display the list of lecturers
End

Figure 3.4: Activity Diagram of < View Project Archive >

2.7.1.5 UC005: Use Case 005 <Review Submission>

Use Case Name	Review Submission
Brief Description	The use case involves the functionality of providing feedback on a submission. It allows authorized users, such as supervisors, to review the content.
Actors	Supervisors
Preconditions	• The examiner or advisor is authenticated and authorized to access the system. • The management system is operational and capable of handling thesis review and feedback processes.
Post Conditions	• The system may trigger subsequent actions
Basic Flow	**Supervisor Accesses File Submissions for Review:** 3. The supervisor logs into the system and accesses the assigned thesis for review. 4. The system presents the thesis document or relevant details for the examiner to review. **Supervisor Provides Feedback:** 3. The supervisor records their feedback, comments, and suggestions regarding the thesis. 4. The system may provide input fields, comment boxes, or other means to capture the examiner's feedback.
Alternative Flow	• The System may incorporate a deadline mechanism to ensure timely submission of the examiner’s review and feedback.
Exception Condition	• Error in submitting feedback due to connection error may cause the system to initiate an error message

Review Submission

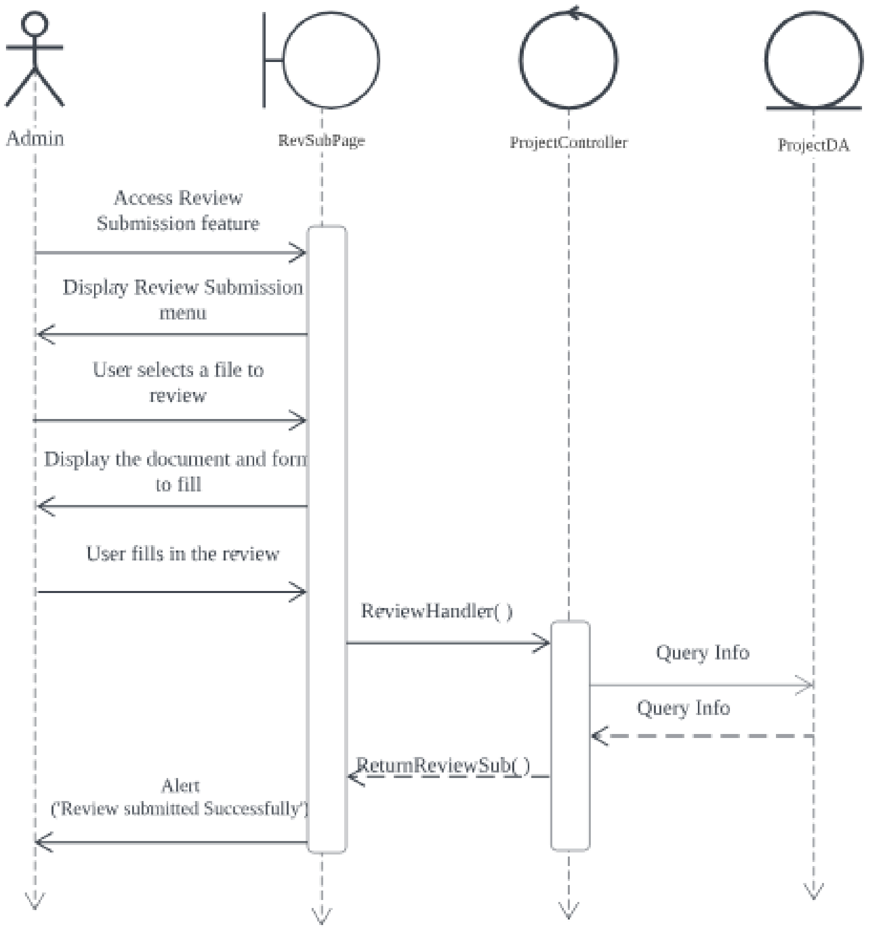

Admin
RevSubPage
ProjectController
ProjectDA
Access Review Submission feature
Display Review Submission menu
User selects a file to review
Display the document and form to fill
User fills in the review
ReviewHandler( )
Query Info
Query Info
ReturnReviewSub( )
Alert ('Review submitted Successfully')

Figure 3.3: System Sequence Diagram of < Review Submission >

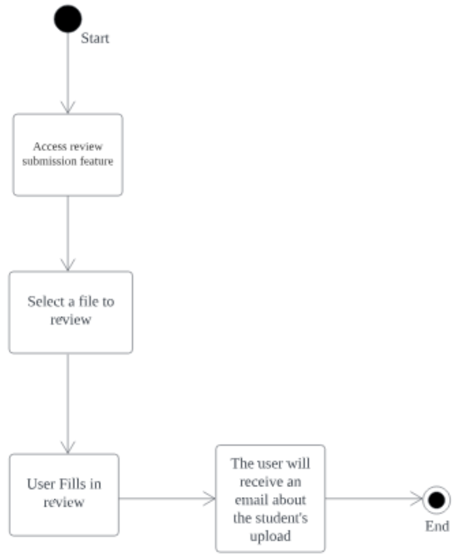

Start
Access review submission feature
Select a file to review
User Fills in review
The user will receive an email about the student's upload
End

Figure 3.4: Activity Diagram of < Review Submission >

2.7.2 Module <Viva Management Module>

This module consists of 4 use cases which all of them are related with viva management module

- Nominate Examiner
- View Viva Details
- Accept Nomination Request
- Evaluate Student

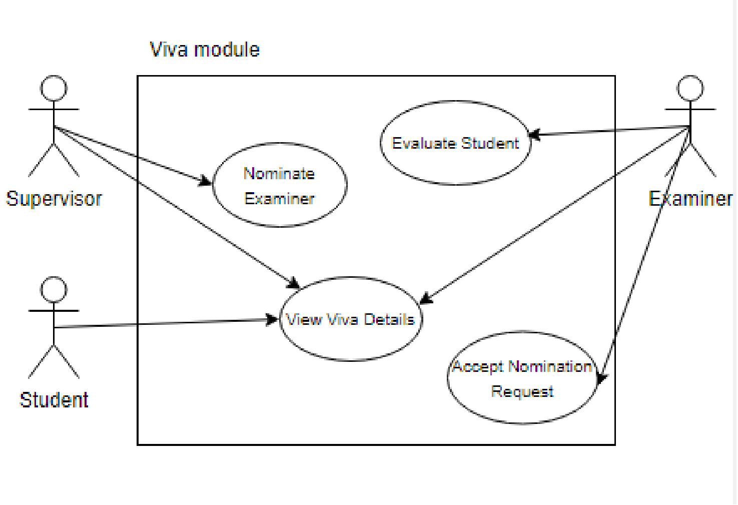


Figure 3.2: < Thesis Management Module >

2.7.2.1 UC006: Use Case 006 <Nominate Examiner>

Use Case Name	Nominate Examiner
Brief Description	The use case involves the functionality of nominating examiners for thesis evaluation within a thesis management system. It allows authorized users, such as supervisors, to propose examiners to evaluate a student's thesis.
Actors	Supervisors, Admin

Preconditions	• The administrator or advisor is authenticated and authorized to access the system. • The thesis management system is operational and capable of handling examiner nominations.
Post Conditions	• The nominated examiner(s) are associated with the chosen thesis within the thesis management system. • The system maintains a record of the examiner assignment for future reference and tracking. • The nominated examiner will receive their nomination request by email
Basic Flow	**Supervisor initiates Examiner Nomination:** 1. The supervisor accesses the examiner nomination functionality within the system. 2. The system presents a list of thesis that require examiner nomination. 3. The supervisor selects the thesis for which an examiner needs to be nominated. **Supervisor Assigns Examiner**: 1. The supervisor assigns the selected examiner(s) to the chosen titles. 2. The system records the examiner assignment and associates it with the relevant thesis 3. The supervisor will assign 2 examiners for a project
Alternative Flow	**Supervisor did not assign enough examiner:** 1. The system will display appropriate messages 2. The nomination process will continue until 2 examiners have been selected
Exception	• If there are no available potential examiners or

Condition	insufficient examiners to choose from, the system may display appropriate messages

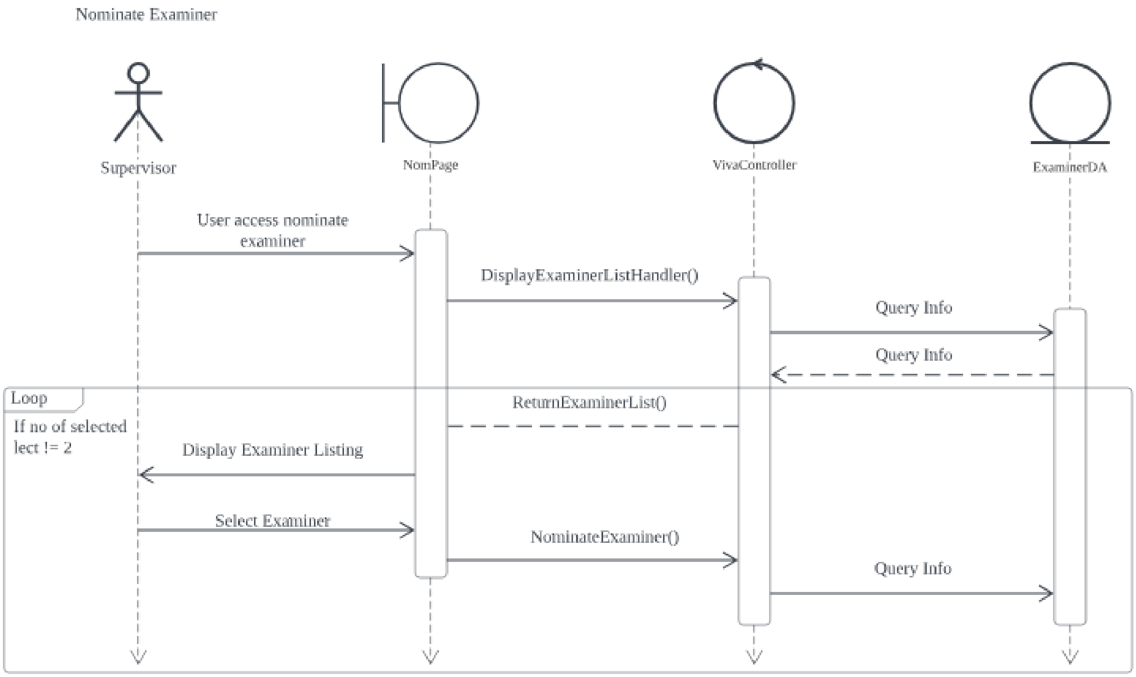


Figure 3.3: System Sequence Diagram of < Nominate Examiner >

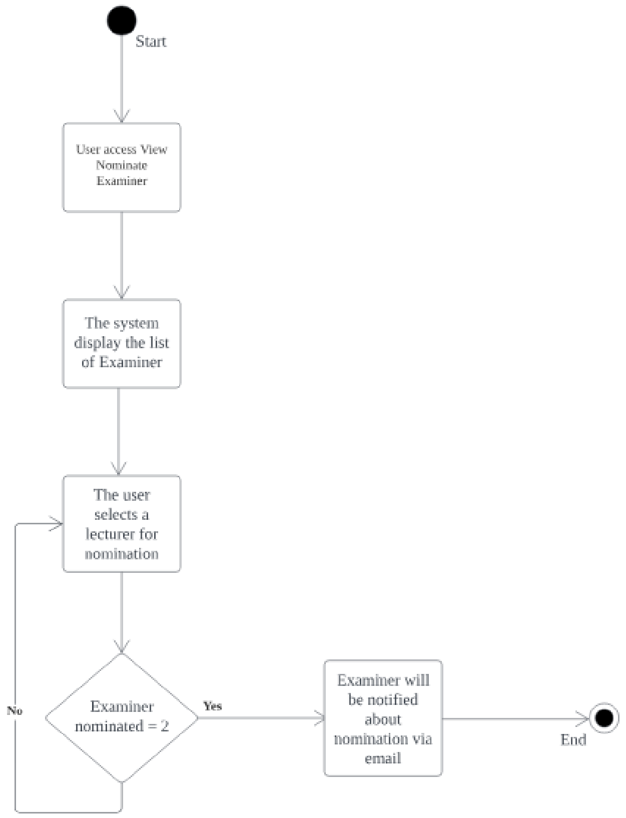

Start
User access View Nominate Examiner
The system display the list of Examiner
The user selects a lecturer for nomination
Examiner nominated = 2
No
Yes
Examiner will be notified about nomination via email
End

Figure 3.4: Activity Diagram of < Nominate Examiner >

2.7.2.2 UC007: Use Case 007 <Receive Nomination Request>

Use Case Name	Receive Nomination Request
Brief Description	The use case involves the functionality of accepting a nomination request to serve as an examiner to evaluate the student for their thesis. It allows authorized users to accept nomination requests from supervisors to fulfill the role of an examiner for a specific student
Actors	Examiner
Preconditions	• The examiner is authenticated and authorized to access the system. • The thesis management system is operational and capable of handling nomination requests and approvals.

Post Conditions	• The nominee’s acceptance of the nomination is recorded in the thesis management system. • The system will display the status of acceptance, the supervisor will be notified via an email
Basic Flow	**Nominee Receives Nomination Request:** 1. The nominee is notified or receives a request to serve as an examiner for a specific thesis. 2. The system will notify the nominee regarding the request via email **Nominee Reviews Thesis Details:** 1. The nominee accesses the nomination request in the thesis management system. 2. The system presents the details of the nominated thesis, including the thesis topic, student information, and any additional relevant details. **Nominee Considers Nomination:** 1. The nominee evaluates their availability, expertise, and any potential conflicts of interest. 2. The nominee may review their workload, research interests, and any other factors that might influence their decision. **Nominee Accepts Nomination:** 1. If the nominee decides to accepts the nomination ,they will indicate their acceptance within the thesis management system. 2. The system records the nominee's acceptance and associates them with the nominated thesis.
Alternative Flow	• If the nominee decides to decline the nomination,

	they can indicate their rejection within the system, providing an explanation if required.
Exception Condition	• Error in submitting feedback due to connection error may cause the system to initiate an error message

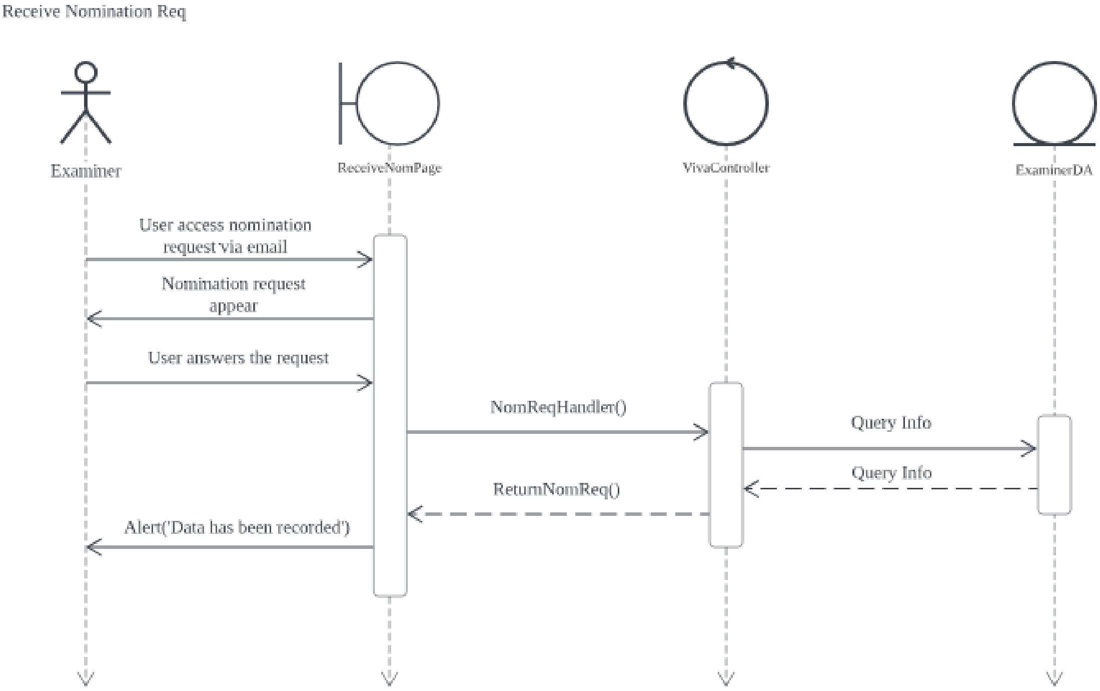


Figure 3.3: System Sequence Diagram of < Receive Nomination Request >

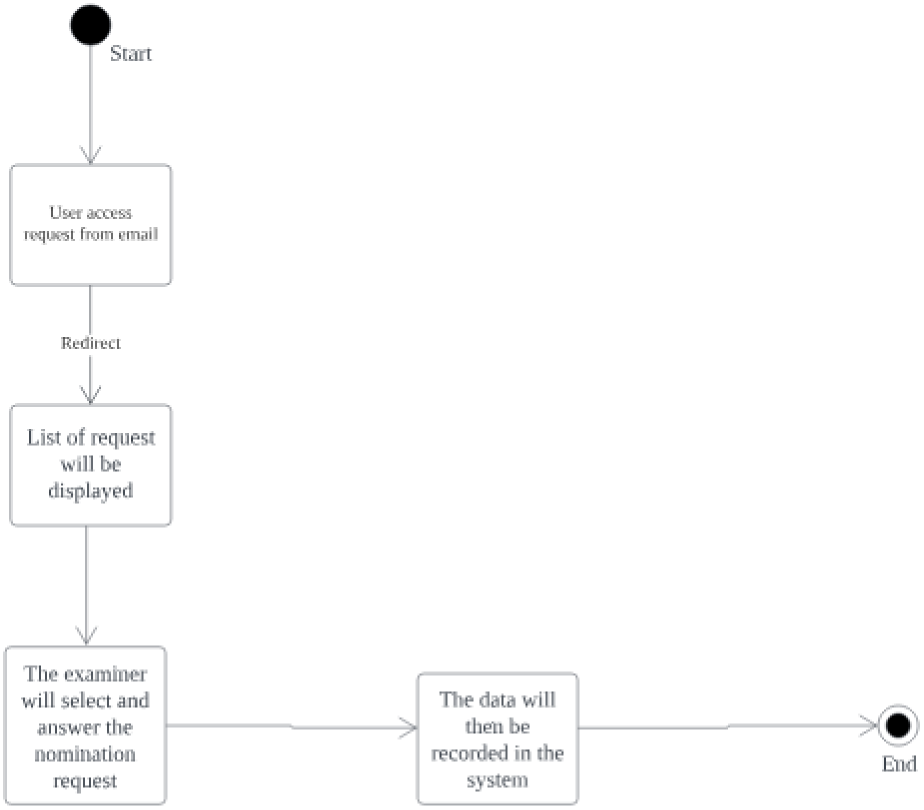

Start
User access request from email
Redirect
List of request will be displayed
The examiner will select and answer the nomination request
The data will then be recorded in the system
End

Figure 3.4: Activity Diagram of < Receive Nomination Request >

2.7.2.3 UC008: Use Case 008 <Evaluate Student>

Use Case Name	Evaluate Student
Brief Description	The use case involves the functionality of accepting a nomination request to serve as an examiner to evaluate the student for their thesis. It allows authorized users to accept nomination requests from supervisors to fulfill the role of an examiner for a specific student
Actors	Examiner
Preconditions	• The examiner is authenticated and authorized to access the system. • The thesis management system is operational and capable of handling nomination requests and approvals.

Post Conditions	• The examiner evaluates the student's thesis work within the system. • The system maintains a record of the evaluation, including feedback, scores, and any additional notes or comments. • The evaluation results contribute to the overall assessment of the student's thesis and can be used for administrative purposes, grading, and decision-making regarding the student's progress.
Basic Flow	**Examiner Accesses Student Evaluation**: 1. The examiner accesses the evaluation functionality within the thesis management system. 2. The system provides a list of students assigned to the examiner for evaluation. 3. The examiner selects a specific student from the list to initiate the evaluation process. **Examiner Reviews Student's Thesis Work:** 1. The system presents the student's thesis document and any associated materials for the examiner to review. 2. The examiner reads through the thesis, analyzes the research methodology, evaluates the quality of writing, and assesses the adherence to academic standards. **Examiner Provides Evaluation and Feedback:** 1. The examiner applies predefined evaluation criteria or rubrics to assess different aspects of the student's thesis work.

	2. The system provides input fields or evaluation forms for the examiner to provide detailed feedback and scores for each criterion. 3. The examiner fills in the evaluation form, providing constructive feedback, strengths, areas for improvement, and an overall evaluation score. **System Records Evaluation:** 1. The system captures and records the examiner's evaluation, feedback, scores, and any additional comments. 2. The system updates the student's evaluation section with the provided information, making it accessible to other authorized users.
Alternative Flow	• none
Exception Condition	• Error in submitting feedback due to connection error may cause the system to initiate an error message

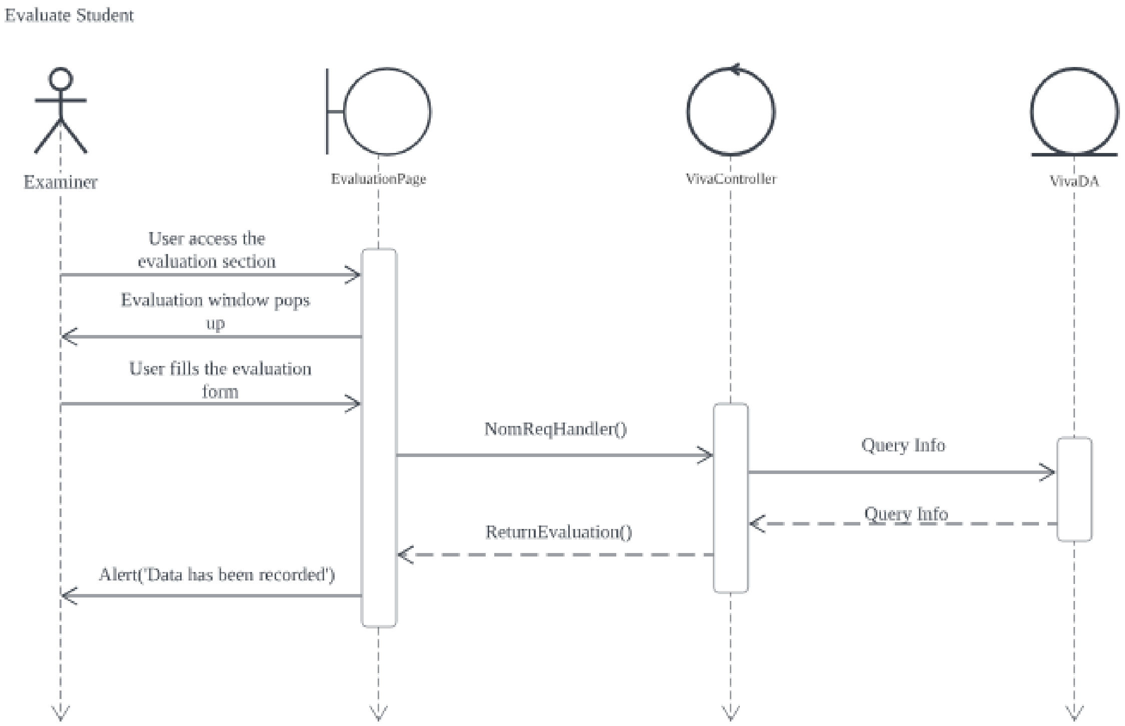

Evaluate Student
Examiner
EvaluationPage
VivaController
VivaDA
User access the evaluation section
Evaluation window pops up
User fills the evaluation form
NomReqHandler()
Query Info
Query Info
ReturnEvaluation()
Alert('Data has been recorded')

Figure 3.3: System Sequence Diagram of < Evaluate Student >

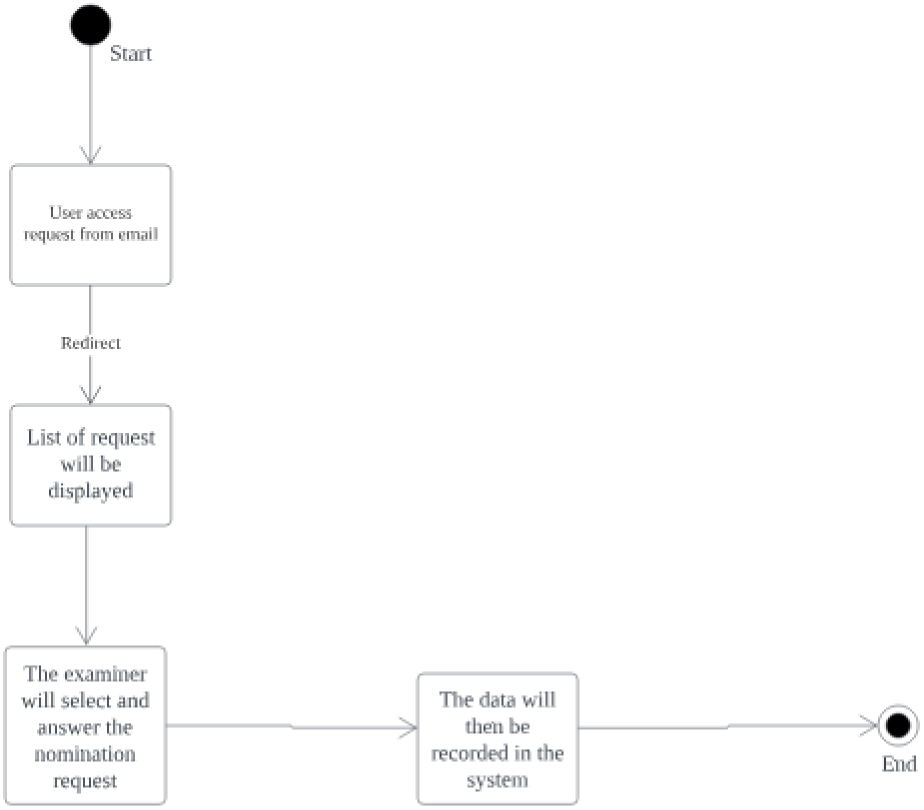

Start
User access request from email
Redirect
List of request will be displayed
The examiner will select and answer the nomination request
The data will then be recorded in the system
End

Figure 3.4: Activity Diagram of < Evaluate Student >

2.7.2.4 UC009: Use Case 009 <View Viva Details>

Use Case Name	View Viva Details
Brief Description	The use case functions to gather all data related to the viva into a centralized section. The function will assist examiners and students by having a section to refer to in regards of their viva details.
Actors	Student, Supervisor, Examiner
Preconditions	• The user is authenticated and authorized to access the system. • The thesis management system is operational and capable of displaying viva related data.
Post Conditions	• The system will display a detailed information about viva

Basic Flow	**User Requests Viva Details** 1. The student or administrator accesses the "View Viva details" functionality within the management system. **System Presents Viva Details** 1. The system displays the viva details 2. The lecturer list may be presented in a tabular format or a visually organized interface, allowing users to browse and find individual lecturer.
Alternative Flow	- None
Exception Condition	- If the requested viva data is not available or inaccessible due to permissions or other reasons, the system may display an appropriate error message to the user.

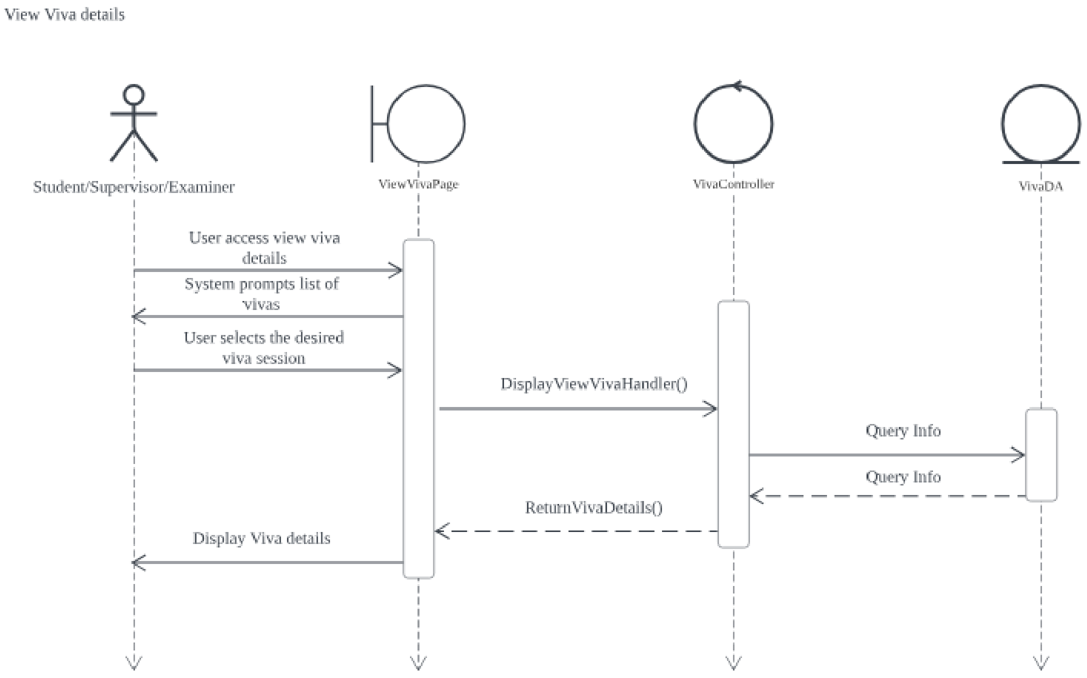


Figure 3.3: System Sequence Diagram of < View Viva Details >

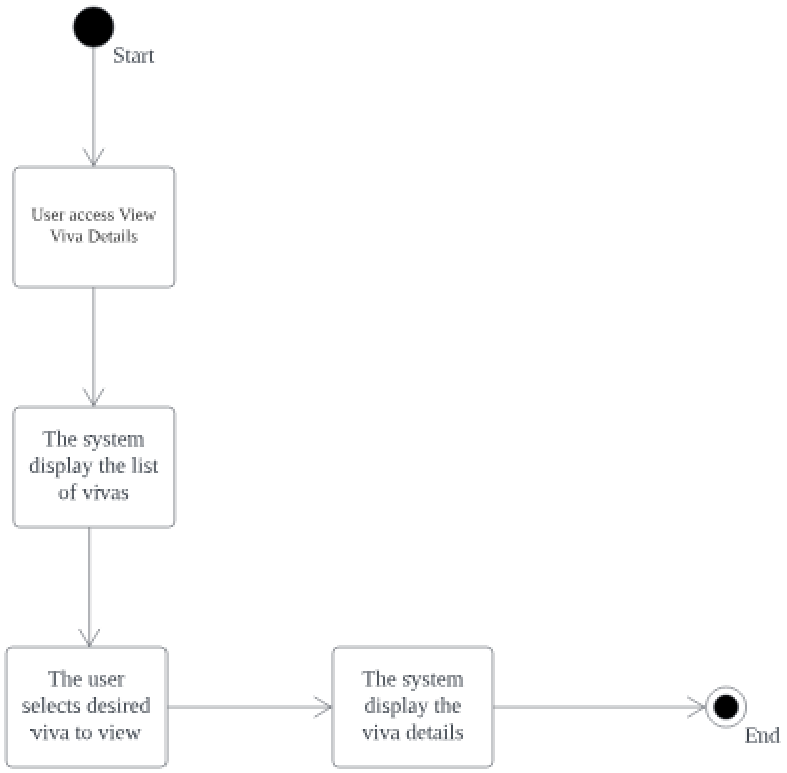

Start
User access View Viva Details
The system display the list of vivas
The user selects desired viva to view
The system display the viva details
End

Figure 3.4: Activity Diagram of < View Viva Details >

2.7.3 Module <Data Management Module>

This module consists of 5 use cases which all of them are related with data management module

- Login
- Register User
- Manage Student Data
- Receive Upload Notification
- Download Thesis Data

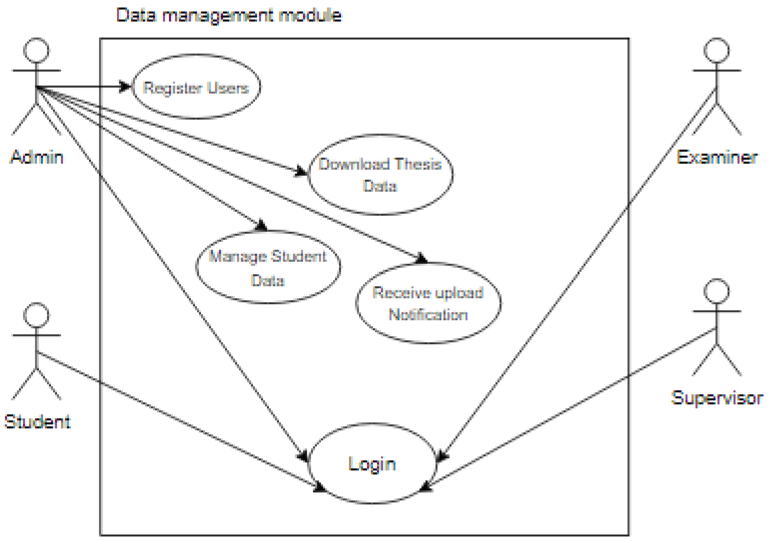


Figure 3.2: < Data Management Module >

2.7.3.1 UC010: Use Case 010 <Login>

Use Case Name	Login
Brief Description	The use case involves the functionality of authenticating users and granting access to the thesis management system. It allows users, such as students, advisors, or administrators, to provide their credentials and securely log into the system to perform various actions related to thesis management. This use case is essential for ensuring that only authorized individuals can access and interact with the system's functionalities.
Actors	Students, Supervisors, examiners ,admin
Preconditions	• The user has a registered account in the thesis management system.. • The system is operational and capable of securely authenticating user credentials.
Post Conditions	• The user successfully logs into the thesis management system, gaining access to the authorized functionalities based on their user role and permissions.

	• The system maintains the user's session and allows them to perform actions within the system until they explicitly log out or the session expires.
Basic Flow	**User Enters Credentials** 1. The user accesses the login page or interface provided by the thesis management system. 2. The system prompts the user to enter their username and password. **System Validates Credentials:** 1. The system verifies the authenticity of the provided credentials by comparing them with the stored user account information. 2. If the credentials are valid, the system proceeds to grant access; otherwise, it displays an appropriate error message. **System Grants Access:** 1. Upon successful validation of the user's credentials, the system allows the user to access the system. 2. The system may redirect the user to a specific landing page or the last accessed page before the login process.
Alternative Flow	• If the user enters an incorrect username or password, the system displays an error message and allows the user to retry entering the correct credentials.
Exception Condition	• In Case of repeated failed login attempts, the system may display appropriate messages

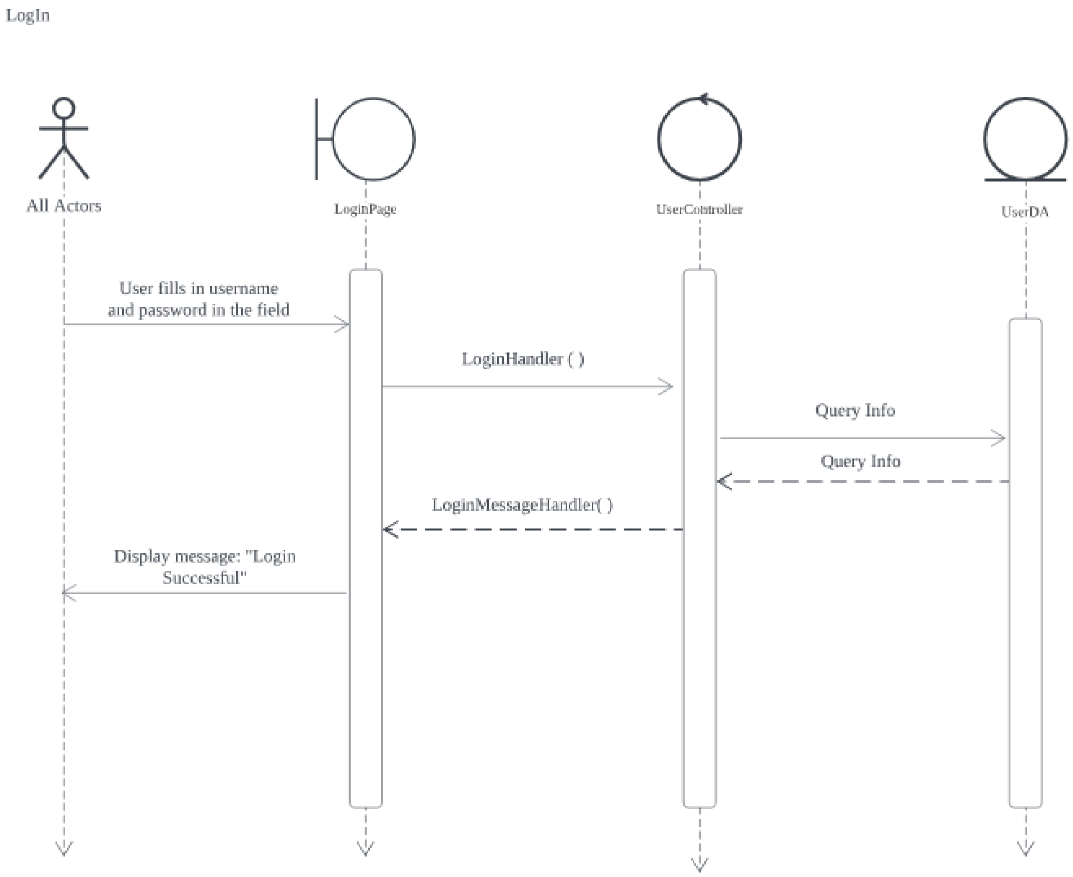


Figure 3.3: System Sequence Diagram of < Login >

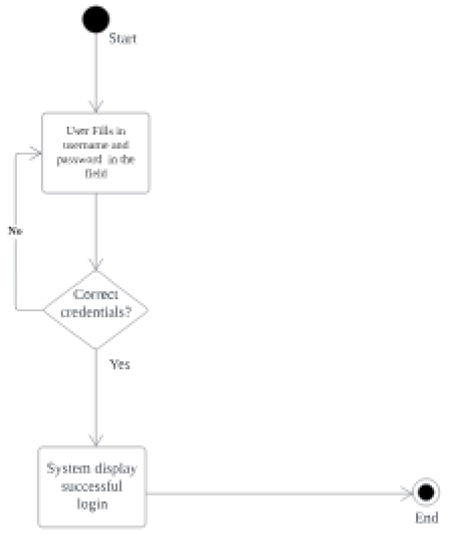

Start
User Fills in username and password in the field
No
Correct credentials?
Yes
System display successful login
End

Figure 3.4: Activity Diagram of < Login >

2.7.3.2 UC011: Use Case 011 <Register Users>

Use Case Name	Register Users
Brief Description	The use case for a thesis management system involves the functionality of the system administrator registering new students into the system. It allows the system administrator to create user accounts for prospective students, capture their academic information, and enable them to participate in the thesis program. This use case ensures accurate and efficient student registration within the system.
Actors	Admin
Preconditions	• The administrator is authenticated and authorized to access the system. • The thesis management system is operational and capable of handling new student registrations.
Post Conditions	• The system administrator successfully registers a new student into the thesis management system. • The new student is provided with login credentials to access the system and participate in the thesis program. • The system maintains an accurate record of student registrations for administrative and tracking purposes.
Basic Flow	**Administrator Accesses Registration:** 1. The administrator accesses the registration functionality within the thesis management system. 2. The system presents the registration form or user interface for capturing student information. **Administrator Enters Personal Information:** 1. The administrator enters the prospective student's personal details, such as name, contact

	information, and identification number, into the registration form. 2. The system may include validation mechanisms to ensure the accuracy and completeness of the provided information. **Administrator Enters Academic Information:** 1. The administrator captures the academic details of the prospective student, including the program of study, institution, and any relevant academic history. 2. The system may incorporate dropdown menus or pre-defined options to simplify the selection process. **Administrator Sets Account Credentials:** 1. The administrator creates a username and password combination for the student's thesis management system account. The common ones would be their name as the username and matric ID as the password. 2. The system may enforce password strength requirements and verify the uniqueness of the username. **System Registers Student:** 1. The system validates the entered information for completeness, accuracy, and adherence to any predefined rules or requirements. 2. Upon successful validation, the system registers the student as a new user in the thesis management system.

Alternative Flow	- The system may incorporate additional steps for verifying the eligibility of prospective students, such as checking admission requirements or reviewing supporting documents.
Exception Condition	- The system may include error handling mechanisms to address cases where incomplete or incorrect information is provided during registration.

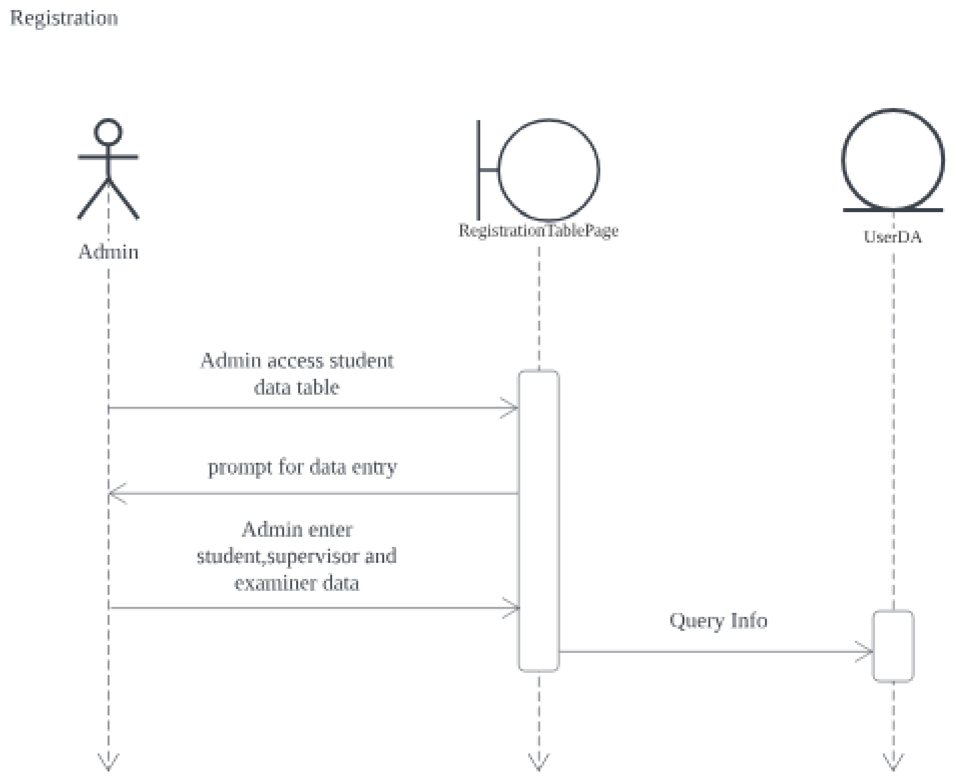
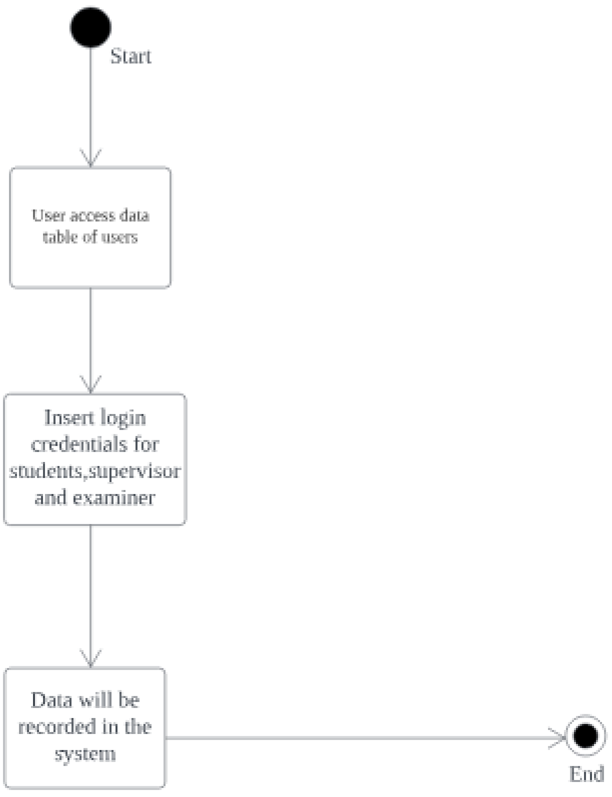

Registration
Admin
RegistrationTablePage
UserDA
Admin access student data table
prompt for data entry
Admin enter student,supervisor and examiner data
Query Info

Figure 3.3: System Sequence Diagram of < Register Users >



Start
User access data table of users
Insert login credentials for students,supervisor and examiner
Data will be recorded in the system
End

Figure 3.4: Activity Diagram of < Register Users >

2.7.3.3 UC012: Use Case 012 <Manage Student Data>

Use Case Name	Manage Student Data
Brief Description	The use case involves the functionality of managing student-related data within a thesis management system. It allows authorized users such as administrators to create, update, and maintain student records, including personal information and academic details
Actors	Admin
Preconditions	• The administrator or advisor is authenticated and authorized to access the system. • The thesis management system is operational and

	capable of handling student data.
Post Conditions	• The student data within the thesis management system is up-to-date and accurately reflects the student's information, academic progress, and thesis-related details. • The system maintains a record of student profiles, allowing for historical tracking and reference purposes. • Authorized users can access and utilize the student data for various administrative, advisory, and evaluation processes within the thesis management system.
Basic Flow	**Administrator Creates Student Profile:** 1. The administrator accesses the student management functionality within the thesis management system. 2. The system provides input fields to capture essential student information, such as name, contact details, academic program, and enrollment status. 3. The administrator fills in the required information and creates a new student profile within the system. **Administrator Updates Student Data:** 1. The administrator can access and update existing student profiles within the system. 2. The system provides an interface to view and modify student data, including personal details, academic information, and thesis-related progress. **3.** The administrator makes necessary changes or updates to the student's profile and saves the

	updated information. **Advisor Accesses Student Data:** 1. The advisor can access the student profiles assigned to them within the thesis management system. 2. The system presents an overview of the student's details, including contact information, academic history, and thesis-related information. 3. The advisor can review the student's progress, provide guidance, and make informed decisions based on the available data. **System Tracks Thesis Progress:** 1. The thesis management system captures and tracks relevant thesis-related data for each student, such as proposal submission, milestone achievements, evaluation outcomes, and completion status. 2. The system may include features to input and update thesis progress, milestone dates, and any associated documents.
Alternative Flow	• The system may incorporate data validation mechanisms to ensure the accuracy and consistency of student information. • The system may provide options for importing or exporting student data in bulk, such as through CSV files or integration with other institutional systems.
Exception Condition	• If the requested thesis data is not available or inaccessible due to permissions or other reasons, the system may display an appropriate error message to the user.

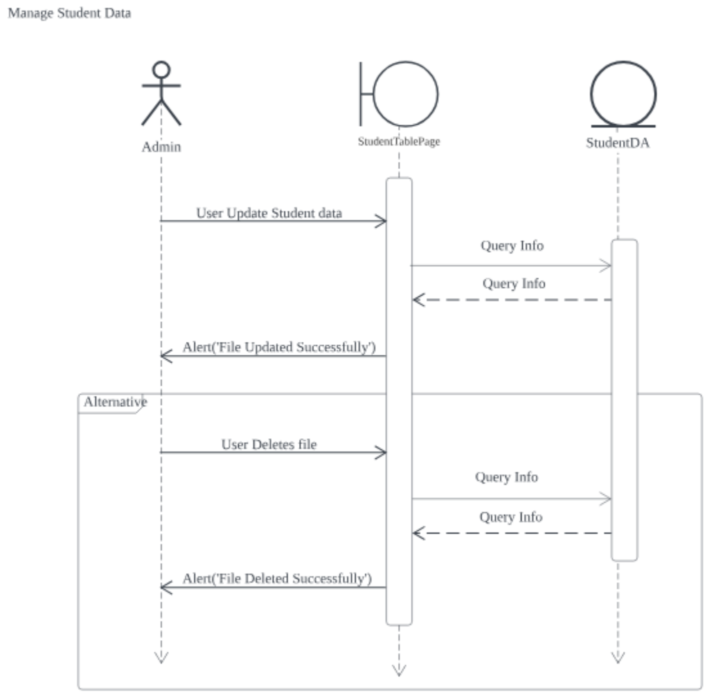


Figure 3.3: System Sequence Diagram of < Manage Student Data >

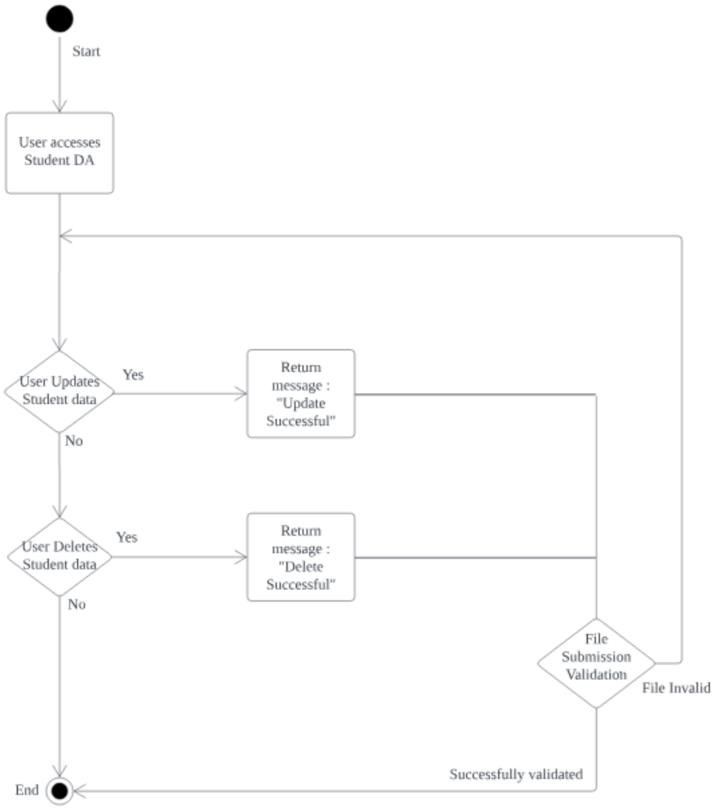

Start
User accesses Student DA
User Updates Student data
Yes
No
Return message : "Update Successful"
User Deletes Student data
Yes
No
Return message : "Delete Successful"
File Submission Validation
File Invalid
Successfully validated
End

Figure 3.4: Activity Diagram of < < Manage Student Data >

2.7.3.4 UC013: Use Case 013 <Receive Upload Notification >

Use Case Name	Receive Upload Notification
Brief Description	The use case involves the functionality of notifying relevant users about the successful upload of a thesis or associated documents within a thesis management system. It enables authorized users, such as administrators to receive timely notifications when a student submits their thesis or related materials through the system. This use case is particularly important to ensure efficient communication and prompt action regarding newly uploaded thesis documents.
Actors	Admin
Preconditions	• The authorized users have subscribed to or enabled notification preferences within the thesis management system. • The thesis management system is operational and capable of handling file uploads and notification delivery.
Post Conditions	• The relevant users receive timely notifications about the successful upload of a thesis or associated documents. • The system maintains a record of the notification delivery for auditing and reference purposes. • Authorized users can promptly review and take necessary actions regarding the newly uploaded thesis within the thesis management system.
Basic Flow	**Student Submits Thesis or Documents:** 1. The student completes the thesis document or prepares other relevant files for submission. 2. The student accesses the thesis management

	system and initiates the upload process. 3. The system validates the uploaded files and confirms successful submission. **System Generates Upload Notification:** 1. Once the student's thesis or documents are successfully uploaded, the system generates an upload notification. 2. The notification includes details such as the student's name, thesis title, and timestamp of the submission. **System Sends Notification to Relevant Users:** 1. The system identifies the relevant users who should receive the upload notification. 2. The system sends the notification to those users via preferred communication channels, such as email, in-app notifications, or push notifications. **Users Receive Upload Notification:** 1. The authorized users, including administrators, advisors, and examiners, receive the upload notification through their selected communication channels. 2. The notification informs them about the newly submitted thesis or documents.
Alternative Flow	
Exception Condition	• The system may incorporate error handling mechanisms to address cases where file uploads fail or encounter technical issues.

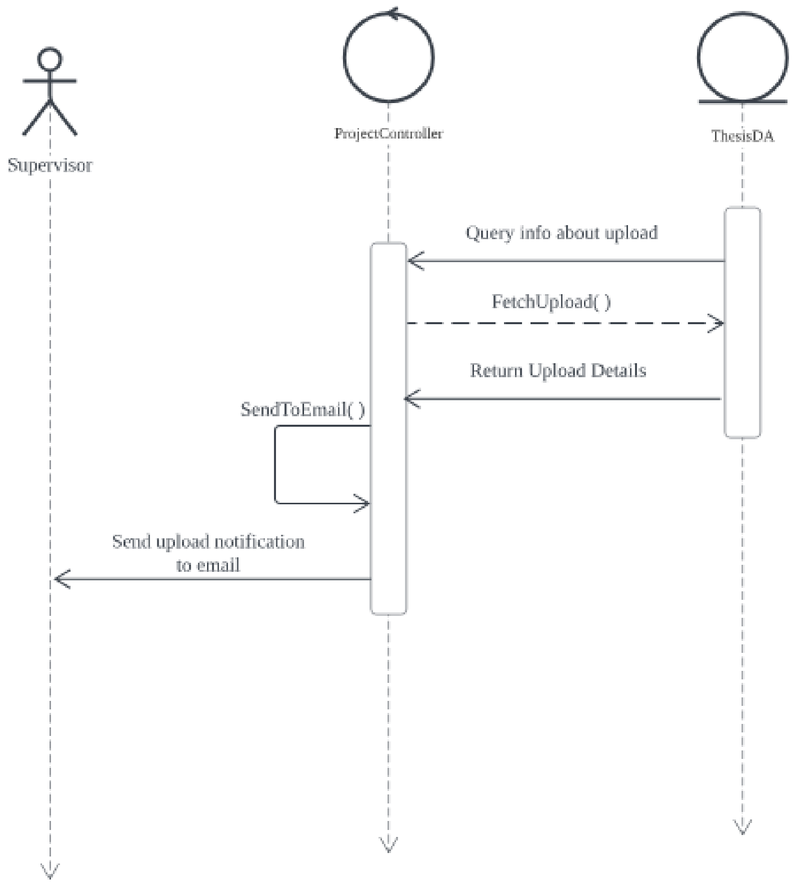


Figure 3.3: System Sequence Diagram of < Receive Upload Notification >

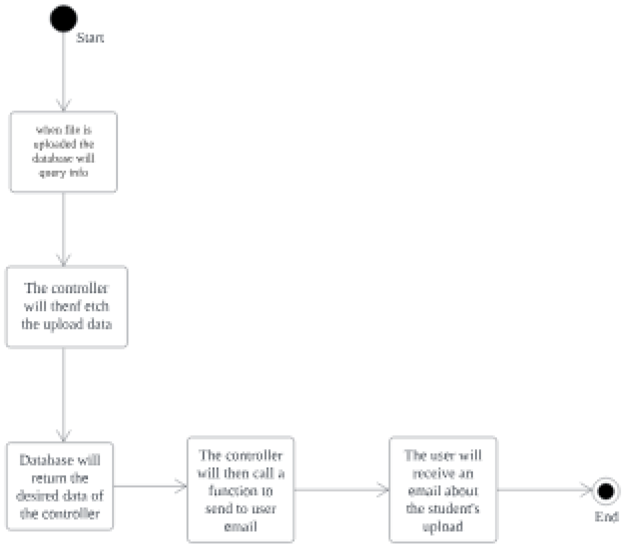

Start
when file is uploaded the database will query info
The controller will thenf etch the upload data
Database will return the desired data of the controller
The controller will then call a function to send to user email
The user will receive an email about the student's upload
End

Figure 3.4: Activity Diagram of < Receive Upload Notification >

2.7.3.5 UC014: Use Case 014 < Download Thesis Data >

Use Case Name	Download Thesis Data
Brief Description	The use case involves the functionality of downloading thesis-related data from the system. It allows authorized users, such as admin to retrieve thesis documents, associated files, or metadata stored within the system.
Actors	Admin
Preconditions	• The user initiating the download is authenticated and authorized to access the system. • The thesis management system is operational and capable of handling file retrieval and data access
Post Conditions	• The user successfully downloads the selected thesis data to their local system. • The system maintains the integrity and security of the remaining thesis data within the thesis management system. • The user can use the downloaded data for further analysis, archiving, or any other purposes as required.
Basic Flow	**User Initiates Download Request:** 1. The user accesses the thesis management system and selects the desired thesis data for download. 2. The system presents options for selecting the data to download, such as specific thesis documents, supplementary files, or metadata. **User Selects Thesis Data:** 1. The user chooses the specific thesis documents, files, or metadata they wish to download. 2. The system validates the selected data and

	prepares it for download. **System Initiates Download:** 1. The system provides a download link or generates a downloadable file that contains the selected thesis data. 2. The user initiates the download process by clicking the download link or saving the generated file to their local storage. **User Receives Downloaded Data:** 1. The user's browser or file management application downloads the selected thesis data. 2. The user may choose a destination location on their local system for saving the downloaded data.
Alternative Flow	• The system may incorporate security measures to prevent unauthorized access or limit the availability of certain thesis data based on user roles or access permissions. • In some cases, the system may impose restrictions on the size or number of files that can be downloaded at once to ensure efficient downloading and system performance.
Exception Condition	• If the requested thesis data is not available or inaccessible due to permissions or other reasons, the system may display an appropriate error message to the user.

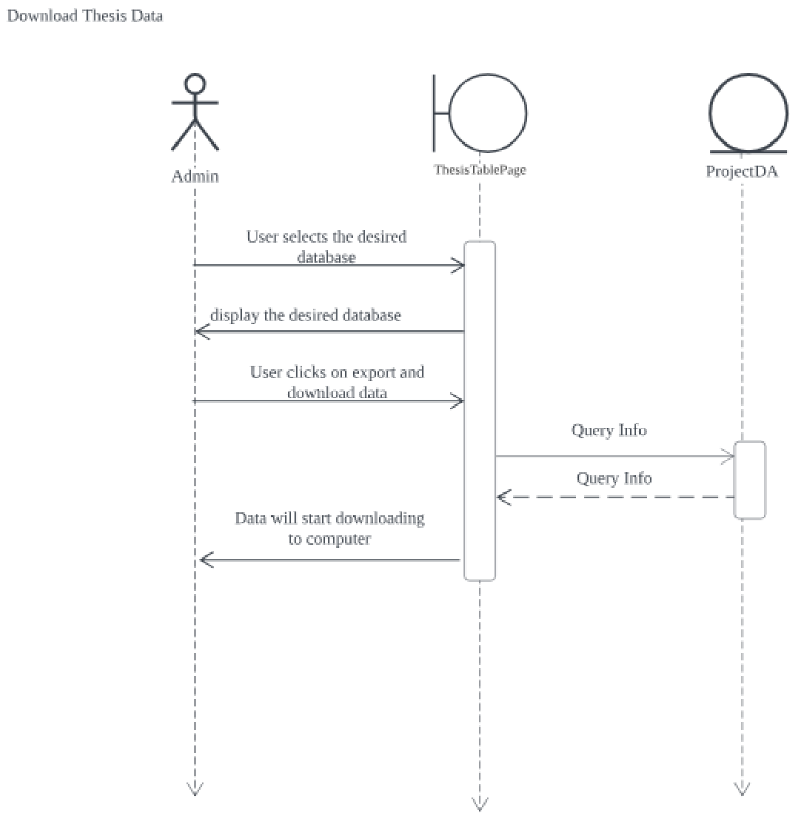

Download Thesis Data
Admin
ThesisTablePage
ProjectDA
User selects the desired database
display the desired database
User clicks on export and download data
Query Info
Query Info
Data will start downloading to computer

Figure 3.3: System Sequence Diagram of < Download Thesis Data >

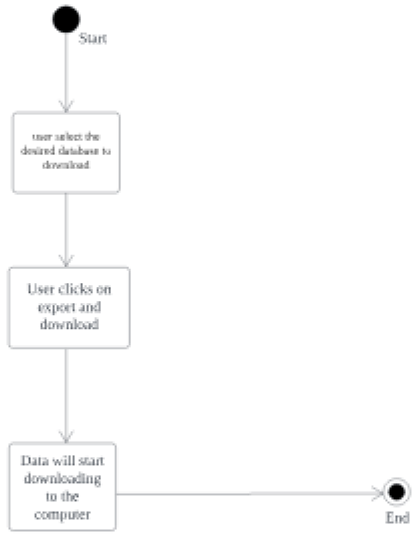

Start
user select the desired database to download
User clicks on export and download
Data will start downloading to the computer
End

Figure 3.4: Activity Diagram of < Download Thesis Data >

2.8 Performance Requirements

State and refer to the specific functional requirement that is related to this non-functional requirement (if any).

2.9 Design Constraints

Explain any constraints imposed by the organization where the software product will be used such as the system must adhere to certain organizational standard and other related non-functional requirements.

2.10 Software System Attributes

Indicate any specific attributes that the customers/users request such as system must be attractive and easy to use for any specific customers.

2.11 Other Requirements

State here other quality characteristics or non-functional requirements for either customers/users or developers such as adaptability, flexibility, interoperability, maintainability, portability, reliability, reusability and usability.