



Automated PhD Theses Management System

**A dissertation submitted for the Degree of Master of
Information Technology**

T. L. K. Arachchi

**University of Colombo School of Computing
2023**



Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: T. L. K. Arachchi

Registration Number: 2020/MIT/007

Index Number: 20550073



Signature:

Date: 07th February 2024

This is to certify that this thesis is based on the work of
Mr./Ms.

Under my supervision. The thesis has been prepared according to the format stipulated and is
of acceptable standard.

Certified by:

Supervisor Name: Dr. A. P. Sayakkara



Signature:

Date: 2024-03-16

Abstract

The management of PhD theses represents a formidable challenge within the academic realm. This complex and time-intensive process demands meticulous attention to detail, involving various stakeholders, including students, supervisors, and reviewers.

Across local educational institutes in Sri Lanka, the predominant approach to managing undergraduate and postgraduate theses has remained rooted in tradition, a manual system reliant on hardcopy thesis submissions, supervisor-led evaluations, and manual distribution of thesis copies. However, this antiquated method has proven to be both inefficient and prone to errors, resulting in a significant drain on resources and an overall lack of transparency and progress tracking.

The absence of an automated PhD thesis management system has created substantial inefficiencies, leading to an inadequate utilization of resources and a dearth of visibility into the thesis management process. Educational institutions grapple with challenges such as a lack of transparency, resource misallocation, and a perpetual inability to effectively monitor progress, culminating in a dire need for a transformative solution.

Enter the proposed Automated Theses Management System, a groundbreaking initiative designed to provide an efficient and secure platform for stakeholders to collaborate seamlessly, ensuring that all related tasks are executed with precision and timeliness. This system addresses a glaring gap in the management of PhD theses, offering respite to educational institutions grappling with the complexities of the process.

A PhD thesis is a monumental endeavor, characterized by years of research, writing, and relentless revision. However, even after the thesis is meticulously prepared, the management process from the initial submission to the final evaluation entails an intricate web of tasks that must be meticulously executed in a timely and accurate fashion.

This journey can be a daunting and stressful experience, not only for students but also for supervisors, evaluators, and reviewers. It is all too easy to lose track of the intricate details and the progress of the thesis in the midst of this complexity.

The introduction of an automated thesis management system offers a lifeline to those embroiled in the thesis management process. By providing a centralized platform, it equips stakeholders with the tools they need to track thesis progress, facilitate collaboration with peers, and harness features tailored to simplify the management process.

For the implementation of the designed web based system, HTML, CSS, Bootstrap technologies have been used as the front-end technologies to facilitate the client side, and PHP, Apache server as the backend technologies for server side, while the database has been implemented using phpMyAdmin.

These technologies were chosen considering their compatibility, flexibility and robustness which facilitate the implementation of the needed features and characteristics of the project such as quick and user friendly front-end development, flexible and efficient data handling and easy cross project integration. The system represents a very simple, efficient, intuitive platform upon which to manage academic theses.

In summary, the Automated Theses Management System is poised to reshape the landscape of PhD thesis management, offering a transformative solution to educational institutes. With its advent, inefficiencies will be minimized, transparency maximized, and the progress of PhD theses will be conducted with newfound precision, all contributing to the enrichment of the academic landscape.

Table of Content

Declaration-----	2
Abstract-----	3
Table of Content-----	5
List of Figures-----	6
List of Tables-----	7
Chapter 01: Introduction-----	8
1.1 Project Overview-----	8
1.2 Motivation-----	8
1.3 Aims and Objectives-----	9
1.4 Structure of the Dissertation-----	10
Chapter 02: Background-----	11
2.1 Introduction-----	11
2.2 Requirement Analysis-----	11
2.2.1 Functional Requirements:-----	11
2.2.2 Non-Functional Requirements:-----	13
2.3 Review of Similar Systems-----	14
2.4 Feature Comparison of the Similar Systems-----	15
Chapter 03: Analysis and Design-----	17
3.1 Design Overview-----	17
3.2 Proposed High-level System Architecture-----	17
3.3 Proposed System Use Case Diagram-----	18
3.4 Proposed Class Diagram-----	20
3.5 Sequence Diagrams-----	21
3.6 Activity Diagram-----	23
Chapter 04: Implementation-----	26
4.1 Proposed Solution:-----	26
4.2. Software Development Life Cycle and Methodology:-----	26
4.3 System Architecture-----	27
4.4 Technology Stack-----	27
4.5 User Interfaces-----	28
Chapter 05: Evaluation-----	34
5.1 Introduction-----	34
5.2 Software Testing-----	34
5.2.1 Testing Strategy-----	34
5.2.2 Test Cases and Results-----	35

Chapter 06: Conclusion and Future Works	47
6.1 Overview	47
6.2 Problems Encountered and Lessons Learned	47
6.3 Critical Evaluation of the System	47
6.4 Future Works	48
References:	49

List of Figures

Figure 3.3.1 - Use Case Diagram	19
Figure 3.3.1 - Use Case Diagram	19
Figure 3.4.1 - Class Diagram	21
Figure 3.5.1 - Thesis Submission Sequence Diagram	21
Figure 3.5.2 - Reviewer Assignment Sequence Diagram	22
Figure 3.5.3 - Final Grading Generation Sequence Diagram	23
Figure 3.6.1 - Thesis Submission Activity Diagram	24
Figure 3.6.2 - Thesis Submission Activity Diagram	25
Figure 3.6.3 - Final Grade Generation Activity Diagram	25
Figure 4.2.1 - Prototyping Software Development Life Cycle	26
Figure 4.5.1 - Login Interface	28
Figure 4.5.2 - Admin Landing Page Interface	29
Figure 4.5.3 - New Program Adding Interface	29
Figure 4.5.4 - Student Account Registration Interface	30
Figure 4.5.5 - Supervisor Account Registration Interface	30
Figure 4.5.6 - Coordinator Account Registration Interface	31
Figure 4.5.7 - Created Accounts List Table View Interface	31
Figure 4.5.8 - Student Submitted Thesis View Interface	32
Figure 4.5.9 - Student New Thesis Submission Interface	32
Figure 4.5.10 - Student Thesis Progress View Interface	33

List of Tables

Table 5.2.1.1 - Test Plan for the evaluation of the implemented system-----	35
Table 5.2.2.1 - Test Case Student Submit the Thesis-----	35
Table 5.2.2.2 - Test Case Create Student Account-----	36
Table 5.2.2.3 - Test Case Create Supervisor Account-----	36
Table 5.2.2.4 - Test Case Create Examiner Account-----	37
Table 5.2.2.5 - Test Case Create Coordinator Account-----	37
Table 5.2.2.6 - Test Case Track Progress Overview-----	38
Table 5.2.2.7 - Test Case View Updated Progress Overview-----	38
Table 5.2.2.8 - Test Case The supervisor views the detailed view of the thesis-----	39
Table 5.2.2.9 - Test Case Supervisor's feedback submission-----	40
Table 5.2.2.10 - Test Case The examiner views the detailed view of the thesis-----	40
Table 5.2.2.11 - Test Case Examiner's feedback submission-----	41
Table 5.2.2.12 - Test Case Student receive and view the feedback received from the supervisor-----	41
Table 5.2.2.13 - Test Case Add new programme to the system-----	42
Table 5.2.2.14 - Test Case Assign examiner-----	43
Table 5.2.2.15 - Test Case Assign supervisor-----	43
Table 5.2.2.16 - Test Case Upload the marks-----	44
Table 5.2.2.17 - Test Case Calculate the Grade-----	44
Table 5.2.2.18 - Test Case The student receives the grade-----	45
Table 5.2.2.19 - Test Case Update assigned the examiners-----	46
Table 5.2.2.20 - Test Case Receive notification reminders for the thesis evaluation-----	46

Chapter 01: Introduction

1.1 Project Overview

The PhD thesis management process is a complex, time-consuming task that requires a lot of attention to detail (Garip, 2020). The process can be laborious and may involve multiple stakeholders, such as the student, the supervisor and reviewers.

Almost all the local educational institutes in Sri Lanka are handling the thesis management system of their undergraduates and post graduates manually in the traditional way. So, in brief it is submission of the hardcopy of the thesis document and proceeding with its marking by the supervisor then sharing copies of the same with the reviewers and all other steps manually in the process. This current system for managing PhD thesis is inefficient, error prone, and time consuming (Arslan, 2018).

So, currently, they lack an efficient system for managing PhD thesis submissions, resulting in inefficiencies in the overall thesis process. This has led to an inefficient use of resources, a lack of transparency, an inability to track progress, and an overall lack of visibility into the thesis process.

The lack of an automated PhD thesis management system poses a challenge to educational institutes, as it hinders their ability to effectively manage the thesis process. So, this problem will be addressed by the proposed automated system to provide an efficient and secure platform for the various stakeholders to collaborate and interact, and ensure that all related tasks are completed in an accurate and timely manner (Khan and Khan, n.d.).

1.2 Motivation

A PhD thesis is a major undertaking, requiring years of research, writing, and revision. Even after preparing the thesis, its management process starting from the initial submission, to the final evaluation involves multiple tasks that need to be handled in a timely and accurate manner.

This process can be daunting and stressful, and it is easy for students as well as for the supervisors, evaluators and reviewers to lose track of the progress of the thesis they manage or to get bogged down in the details.

An automated thesis management system could help to address these challenges by providing the involved people with a centralized platform to track the progress of their thesis, to collaborate with other parties, and to access implemented features to help them with their management process (Kaushik, 2017).

1.3 Aims and Objectives

The aim of this research is to develop an automated thesis management system that will help to address the challenges of thesis management and evaluation with the base of following objectives.

Objective 1: To develop an automated PhD thesis management system for educational institutes to make that process productive, efficient and convenient.

Objective 2: To create a user-friendly platform for educational institutes to manage and track the progress of PhD thesis evaluation and management process.

Objective 3: To enable efficient and secure data exchange between stakeholders.

Objective 4: To create a convenient mechanism instead of the traditional system for tracking the progress of PhD thesis evaluation and analyzing their quality.

Objective 5: To create a system that provides an efficient and secure platform for the various stakeholders to collaborate and interact, and ensure that all related tasks are completed in an accurate and timely manner.

1.3 Scope of the Project

Inscope

Creating an automated PhD thesis management system for educational institutes as a general system to be used by any educational institute. The system will provide an efficient way of managing the entire process of a PhD thesis, from submission to evaluation.

The system will be able to handle tasks such as:

- Create and maintain a database of theses
- Submission of the theses to the supervisors through the system
- Track progress of thesis evaluation and grading
- Generate a view on the progress of theses
- Provide notifications to supervisors and reviewers when tasks need to be completed
- Provide an interface for supervisors to communicate with students and provide feedback
- Generate final grading for the thesis once the marks and feedbacks are provided by the supervisor and reviewers
- Create and manage thesis templates

So, in brief, this system will handle only the main steps that are involved from the point of thesis submission, until the completion of the final evaluations grading of the submitted theses.

Outscope

The proposed system will not handle the PhD preparation and registration process of theses. The proposed system is not capable of handling the intermediate submissions of the PhD process and the progress of a PhD student will be tracked through the system.

1.4 Structure of the Dissertation

The dissertation will be structured as follows:

- Chapter 1: Introduction
- Chapter 2: Background
- Chapter 3: System Analysis and Design
- Chapter 4: Implementation
- Chapter 5: Evaluation
- Chapter 6: Conclusion

Chapter 02: Background

2.1 Introduction

This section provides an overview of the current state of the field in relation to the topic of automating the PhD thesis management system for educational institutes and relevant background and literature related to the topic.

There is a growing body of literature on the topic of automated PhD thesis management systems. This literature covers the benefits of automated systems, the challenges of developing such systems, and the different approaches that have been taken to automating the thesis management process.

2.2 Requirement Analysis

2.2.1 Functional Requirements:

The functional requirements for the Automated Theses Management System are crucial to ensure its successful development and implementation. Here are the identified key functional requirements for the system:

1. Thesis Database Management:

- The system must create and maintain a comprehensive database of all theses submitted by students.
- It should support the storage and retrieval of essential thesis information, including titles, authors, submission dates, and related documents.

2. Thesis Submission Process:

- Users (students) should be able to submit their theses electronically through the system.
- The system must validate and store submitted theses securely.

3. Progress Tracking:

- Provide a feature to track the progress of each thesis, including its current status (e.g., under review, in progress, completed).
- Allow authorized users to view and monitor the progress of individual theses or all theses collectively.

4. Progress Overview:

- Generate a comprehensive view or dashboard displaying the overall progress of all theses.

- Include visual indicators or reports to highlight pending tasks, completed reviews, and any delays.

5. Notification System:

- Implement a notification system to alert supervisors, reviewers, and students about upcoming tasks, deadlines, and status changes.
- Notifications should be sent via email or in-app notifications.

6. Supervisor-Student Communication:

- Provide a communication interface for supervisors to interact with students, discuss thesis-related matters, and provide feedback.
- Support document sharing and discussion threads.

7. Thesis Evaluation and Grading:

- Allow supervisors and reviewers to assess and grade the theses.
- Enable the collection and storage of evaluation data, including comments, marks, and feedback.
- Calculate the final grading based on the input from supervisors and reviewers.

8. Thesis Template Management:

- Include a feature for managing thesis templates, ensuring that students adhere to formatting and structure guidelines.
- Allow administrators to update and maintain these templates as needed.

9. User Roles and Permissions:

- Implement role-based access control to ensure that only authorized personnel can perform specific actions within the system.
- Define roles such as student, supervisor, reviewer, and administrator.

10. Audit Trail and Logging:

- Maintain an audit trail or activity log to record all user interactions and system events.
- This log can be useful for tracking changes, debugging issues, and ensuring accountability.

11. Reporting and Analytics:

- Provide reporting capabilities to generate custom reports on thesis progress, evaluation results, and other relevant metrics.
- Support data analysis for continuous improvement of the thesis management process.

12. User Training and Support:

- Offer user training and provide comprehensive support to ensure that all users can effectively utilize the system.

2.2.2 Non-Functional Requirements:

Non-functional requirements are essential for ensuring the overall performance, reliability, and usability of your "Automated Theses Management System." These requirements complement the functional aspects and help define how the system should perform and operate. Here are the identified non-functional requirements:

1. Performance:

- Response Time: The system should respond to user interactions promptly, with minimal delay, ensuring that users do not experience significant wait times.

2. Reliability:

- Availability: The system should be available 24/7, with planned downtime kept to a minimum for maintenance and updates.

3. Security:

- Ensure data security through access controls, and regular data backups.
- Implement strong user authentication and authorization mechanisms to protect sensitive thesis data.

4. Usability:

- The user interface should be intuitive and user-friendly, with consistent design and navigation.

5. Scalability:

- The system should be able to store a large volume of theses and related data efficiently.

6. User Training:

- Provide training and support materials for users to learn how to use the system effectively.
- Offer ongoing user support to address queries and issues.

By incorporating these non-functional requirements into the "Automated Theses Management System," makes it possible to ensure that the system not only meets its core functionality but also delivers a reliable, high-performance, and secure platform for managing the thesis process in educational institutes.

2.3 Review of Similar Systems

In recent years, the need for automated PhD thesis management systems has become increasingly evident as universities and research institutions worldwide have experienced a surge in the number of PhD students. As a result, many universities and research institutions have developed their own automated PhD thesis management systems to address the growing demand.

Electronic Thesis and Dissertation by the Networked Digital Library of Theses and Dissertations

One such system is the (Electronic Thesis and Dissertation (ETD), 2023) system developed by the Networked Digital Library of Theses and Dissertations (NDLTD). The ETD system is an open-source system that provides a platform for managing and archiving electronic theses and dissertations. It allows students to submit their theses and dissertations electronically and also allows faculty and staff to manage the review and approval process.

ProQuest Dissertations

Another example is the (ProQuest Dissertations, 2023) and Theses Global database, which is a commercial system that provides access to over 5 million theses and dissertations from around the world. The system allows users to search and access theses and dissertations from various universities and research institutions, and also provides tools for managing and archiving these documents.

Thesis and Dissertation Services (TDS) system developed by Texas A&M University

In addition, some universities have developed their own custom automated PhD thesis management systems, such as the (Thesis and Dissertation Services (TDS), no date) system developed by Texas A&M University. The TDS system provides a platform for students to submit their theses and dissertations electronically, and also allows faculty and staff to manage the review and approval process.

The Thesis Management System of the University of New South Wales

The Thesis Management System of the University of New South Wales (TMS: UNSW, no date) is an online platform used by students and faculty members to manage the submission and examination of higher research degree (HDR) theses. The system was developed in-house by the UNSW Graduate Research School, in collaboration with the university's IT department. This system facilitates features such as online submission of theses, integration with other systems and customizable workflows for managing the submission and examination of higher research degree theses.

Thesis Management System of the Erasmus University Rotterdam

Another such system developed by a university is The Thesis Management System of the Erasmus University Rotterdam (TMS: EUR, no date). It is an online platform designed to manage the submission and examination of master's theses and doctoral dissertations. The system is used by students, faculty members, and examiners to streamline the thesis submission and examination process. This system also provides a set of features similar to the above discussed system from the University of New South Wales.

In addition to the above related systems, a few more literature sources have also been studied in order to have a better understanding of the background upon this field. Following are a few of the researches conducted on the related topics:

- (Thesis Process Automation System, 2018)
- (Design of a University Thesis and Project Automation System (UTPAS), 2023)
- (ThesisFS: Online Document Management System, 2014)

2.4 Feature Comparison of the Similar Systems

Below table is about a comparison of a few similar systems based on some of the major features against the proposed system. Following are the systems considered for the below comparison:

- 1) The University of New South Wales: Thesis Management System (UNSW: TMS)
- 2) The EUR Thesis management system (TMS: EUR)
- 3) ProQuest: ETD Dissemination
- 4) TDS by Texas A&M University

	UNSW: TMS	EUR Thesis Management System	ProQuest: ETD Dissemination	TDS by Texas A&M University	Proposed System
Provision of Thesis Templates	No	No	No	Yes	Yes
Tracking view for the progress of each submission	No	Yes	No	Yes	Yes
Automated email notifications to the supervisor and reviewers	No	No	Yes	No	Yes
Possible to add the desired external reviewers	No	No	Yes	No	Yes

Grading publishing through the portal	No	No	No	No	Yes
Interface for student and supervisor to collaborate	No	Yes	No	Yes	Yes

Table 2.3.1 - Feature comparison of the proposed system against the similar systems

Above background study provides information on how such automated thesis management systems should be organized and the components to be included. The aim of the proposed system is to come up with a better and more convenient system with the knowledge gained through this literature and the evaluation of other currently available related systems.

Chapter 03: Analysis and Design

3.1 Design Overview

This chapter focuses on the design of the proposed system using object-oriented methodologies. It includes various diagrams and models to clarify the system's structure, such as use case diagrams, class diagrams, and sequence diagrams. Additionally, entity-relationship diagrams depict the data model. The user interface design section justifies design decisions using wireframes and UI mockups, presenting the transition to final interfaces. The chapter showcases a comprehensive approach to system design, incorporating object-oriented methodologies for an intuitive user interface.

3.2 Proposed High-level System Architecture

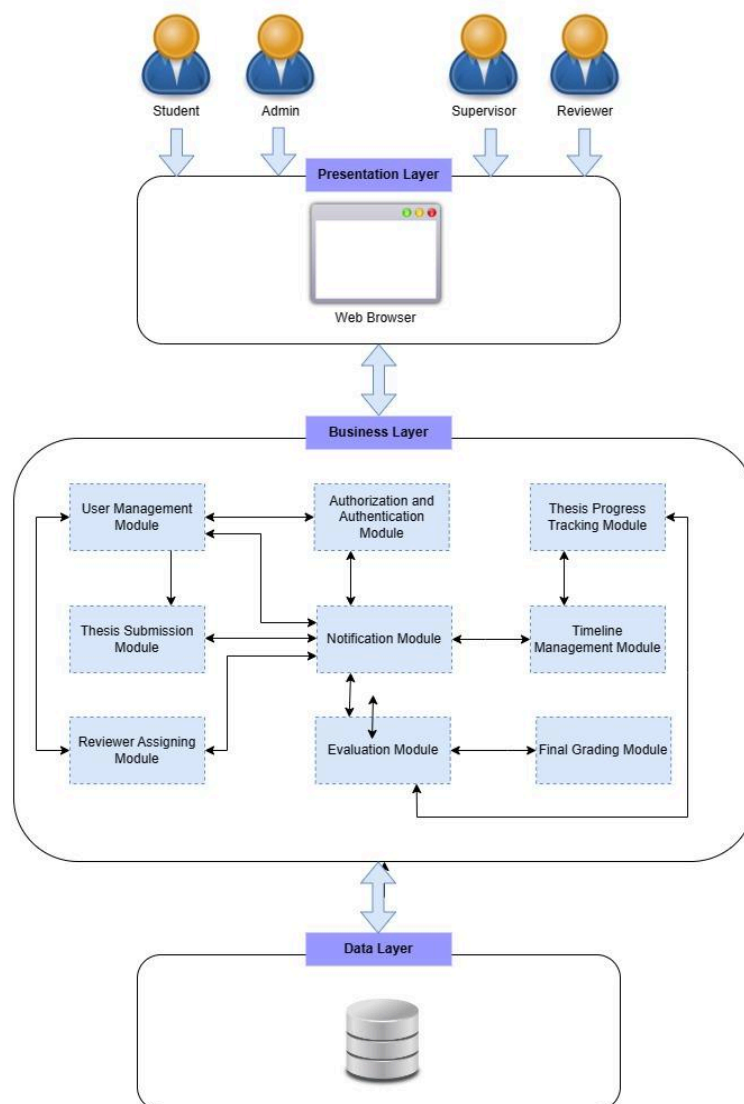


Figure 3.2.1 - High-level System Architecture

The high-level system architecture for the proposed solution is illustrated in the above Figure 3.2.1. The figure presents an overview of the interaction between different components and modules in a single diagram. This diagram is a visual presentation of the structural organization of the system and represents the relationships and dependencies between the different layers, namely the presentation layer, business layer, and data management layer. This architectural diagram serves as a guiding framework, facilitating clear communication of the system's design rationale and technical intricacies to both technical and non-technical stakeholders and readers.

3.3 Proposed System Use Case Diagram

The Figure 3.3.1 is the use case diagram for the proposed system which provides a high-level view of the multiple interactions between users and the system. The diagram highlights the core functions and user roles in an insight and efficient visual that captures all the required aspects of the system. Each of the use cases described in the diagram represents an action that a particular user undertakes. These actions start from registering in the system to submit and evaluate a thesis. In addition to the actions, the diagram portrays the users involved, including students, supervisors, coordinators, and administrators, and how they are connected and dependent on each other. This graphical illustration provides stakeholders with the perfect view and simulation of what the system can do and how the users can interact. This is essential in identifying the requirements from a user and system behavior perspective and ensuring that the system is developed based on the user needs.

Furthermore, the use case diagram is a valuable tool for understanding requirements and ensuring effective communication between stakeholders and development teams. By visually representing user actions, interconnections with different user actions and system features, the diagram allows stakeholders to confirm that all essential user tasks are addressed and the system aligns with their intended workflows and goals. Additionally, the diagram helps identify edge cases and exceptional scenarios, promoting a robust and comprehensive system design. Through iterative refinement and validation, the use case diagram contributes to developing a solution that meets user expectations and fulfills the intended objectives of the proposed system.

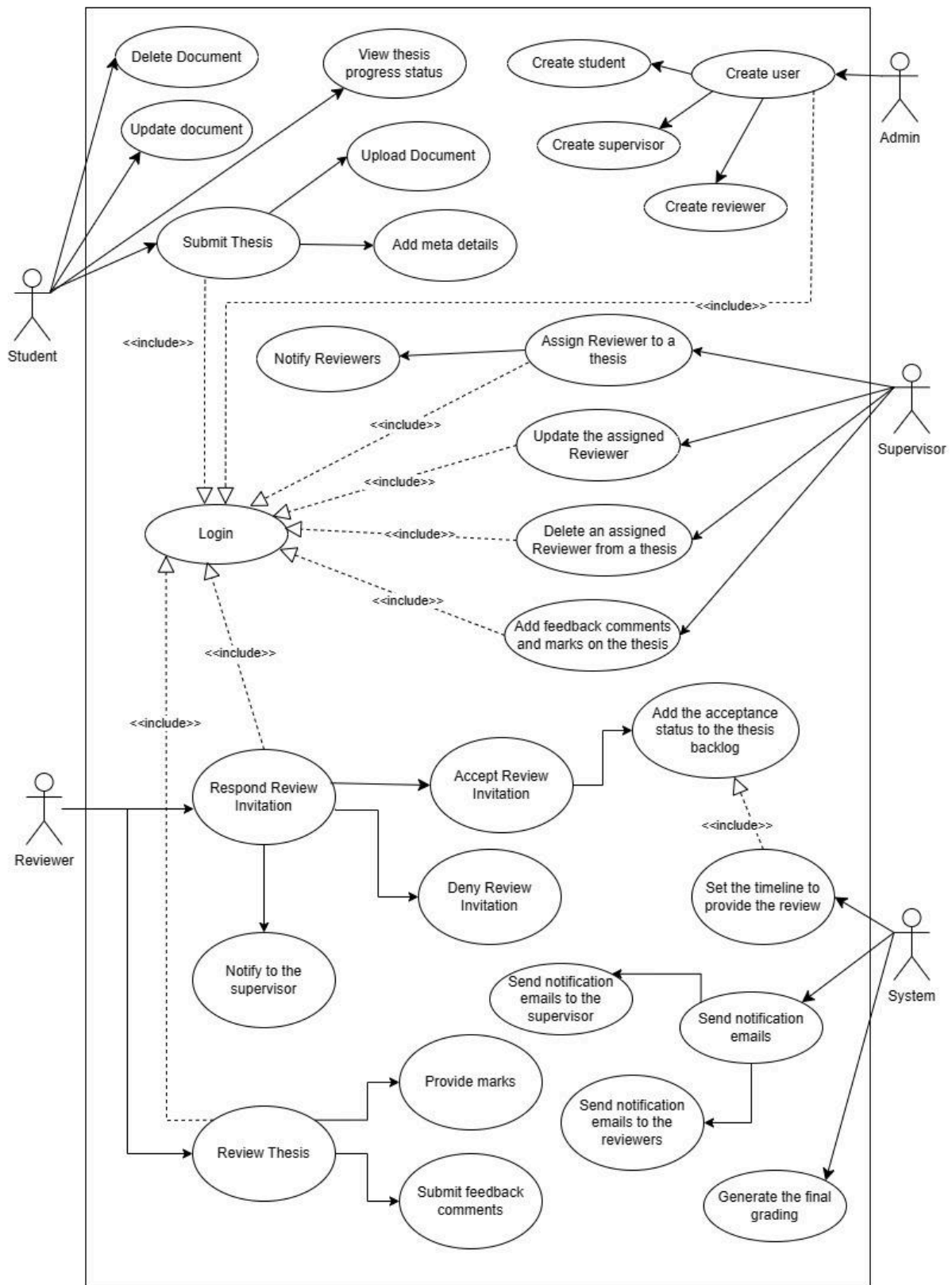


Figure 3.3.1 - Use Case Diagram

3.4 Proposed Class Diagram

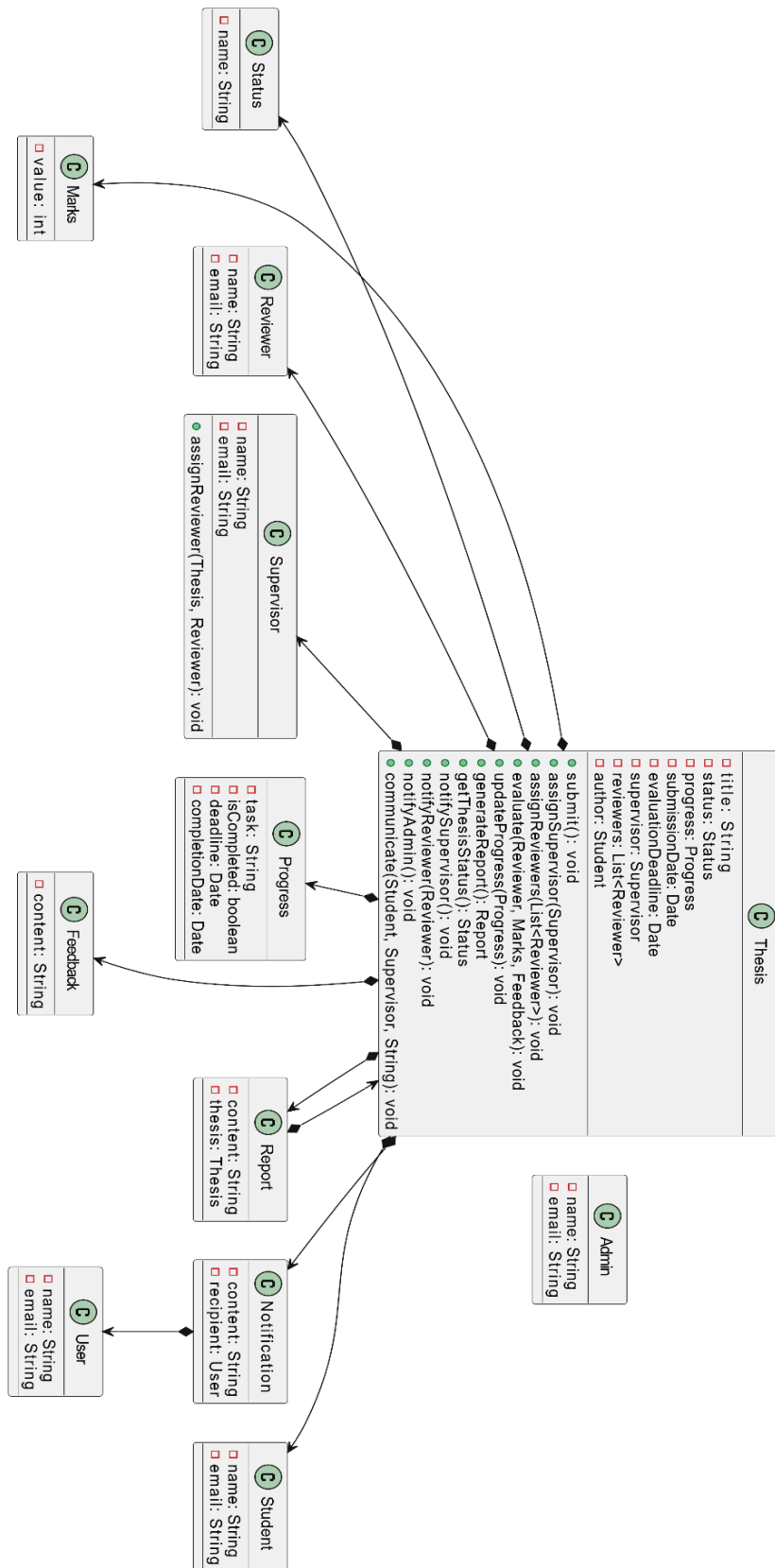


Figure 3.4.1 - Class Diagram

3.5 Sequence Diagrams

Sequence Diagram 1 - Thesis Submission:

Figure 3.5.1 is the sequence diagram for the Thesis Submission functionality by Student which offers a detailed picture of the sequence of interactions between various system components and actors involved in the thesis submission process. Beginning with the initiation of the submission request by the student, the diagram illustrates the sequential flow of events, including form submission, document upload, validation, and confirmation. Through a series of lifelines representing the involved entities, such as the student, user interface, backend system, and database, the diagram captures the exchange of messages and data flows between these elements. By visualizing the dynamic behavior of the system during thesis submission, stakeholders gain a clear understanding of the underlying processes, dependencies, and system responses, aiding in validation, refinement, and optimization of the submission functionality.

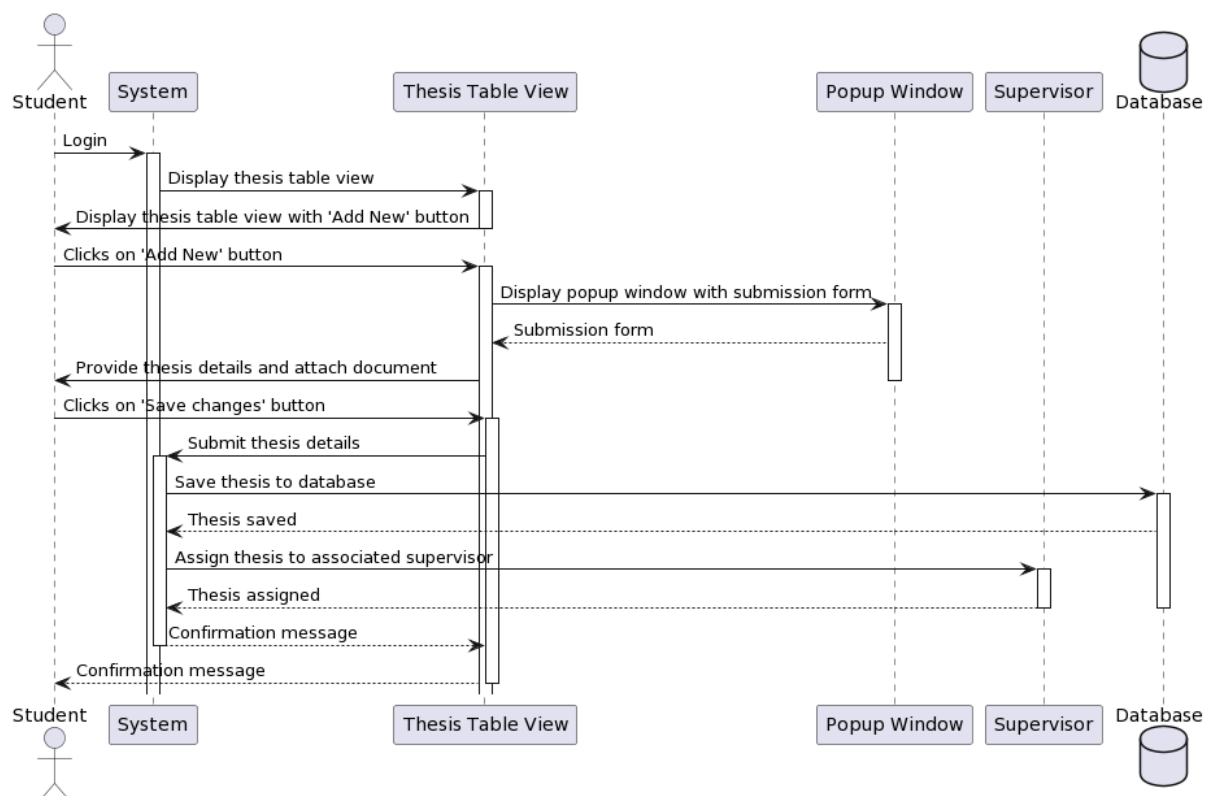


Figure 3.5.1 - Thesis Submission Sequence Diagram

Sequence Diagram 2 - Reviewer Assignment:

The sequence diagram in Figure 3.5.2 shows the process of how a reviewer get assigned to evaluate a thesis. It starts with the Supervisor reviewing the student's thesis and marking it as 'Approved'. Once the status is updated, the Coordinator will be notified once logged into the system. The Coordinator then selects the thesis record and sees a list of registered reviewers. They choose the appropriate reviewer from this list and assign them to evaluate the thesis. This workflow helps in efficiently assigning reviewers to provide timely and effective feedback on the student's academic work.

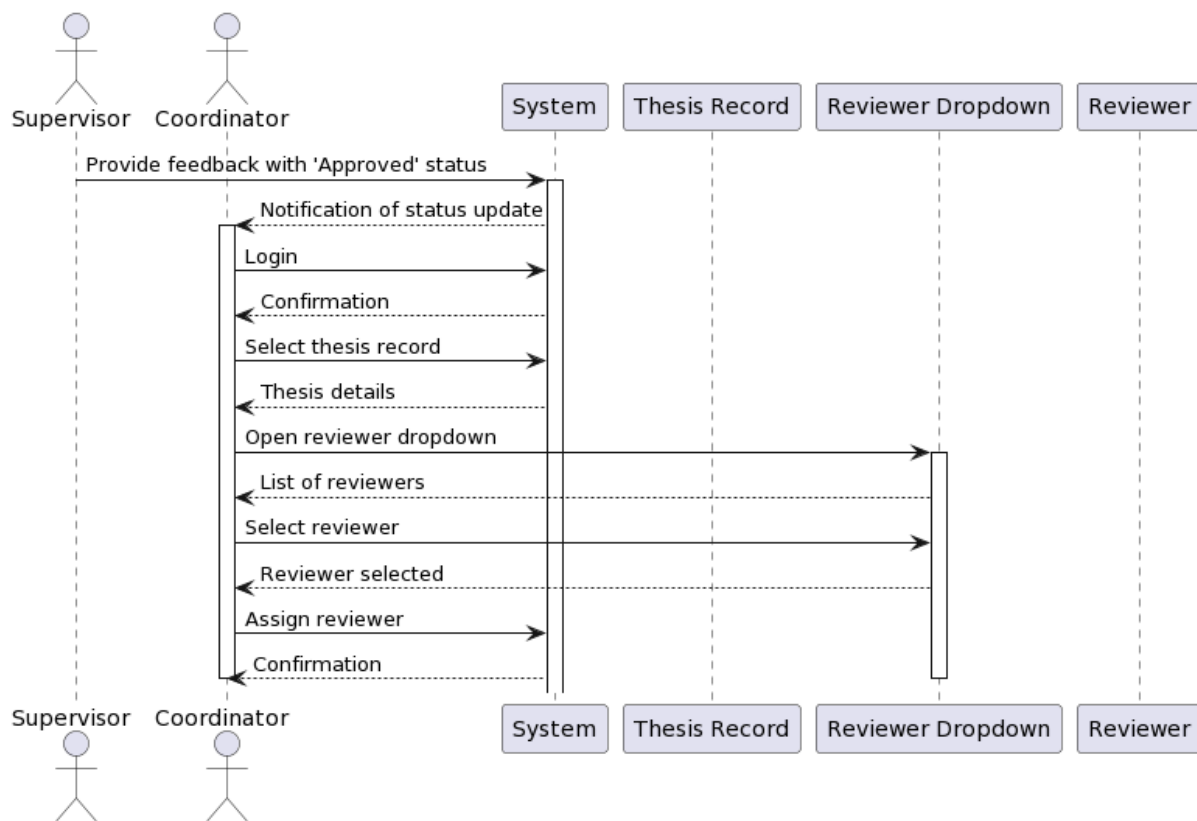


Figure 3.5.2 - Reviewer Assignment Sequence Diagram

Sequence Diagram 3 - Generating the Final Grading of the Thesis:

Figure 3.5.3 is the sequence diagram for the 'Generate Final Grade' process. First, the Supervisor and Reviewer independently evaluate the student's work and mark it as 'Approved' in the system. Then, the Coordinator logs into the system, accesses the student's profile, and enters the marks from both evaluators. When the Coordinator clicks 'Save,' the system

calculates the final grade based on the established logic. This workflow ensures a thorough and fair evaluation, leading to the final grade being generated within the system.

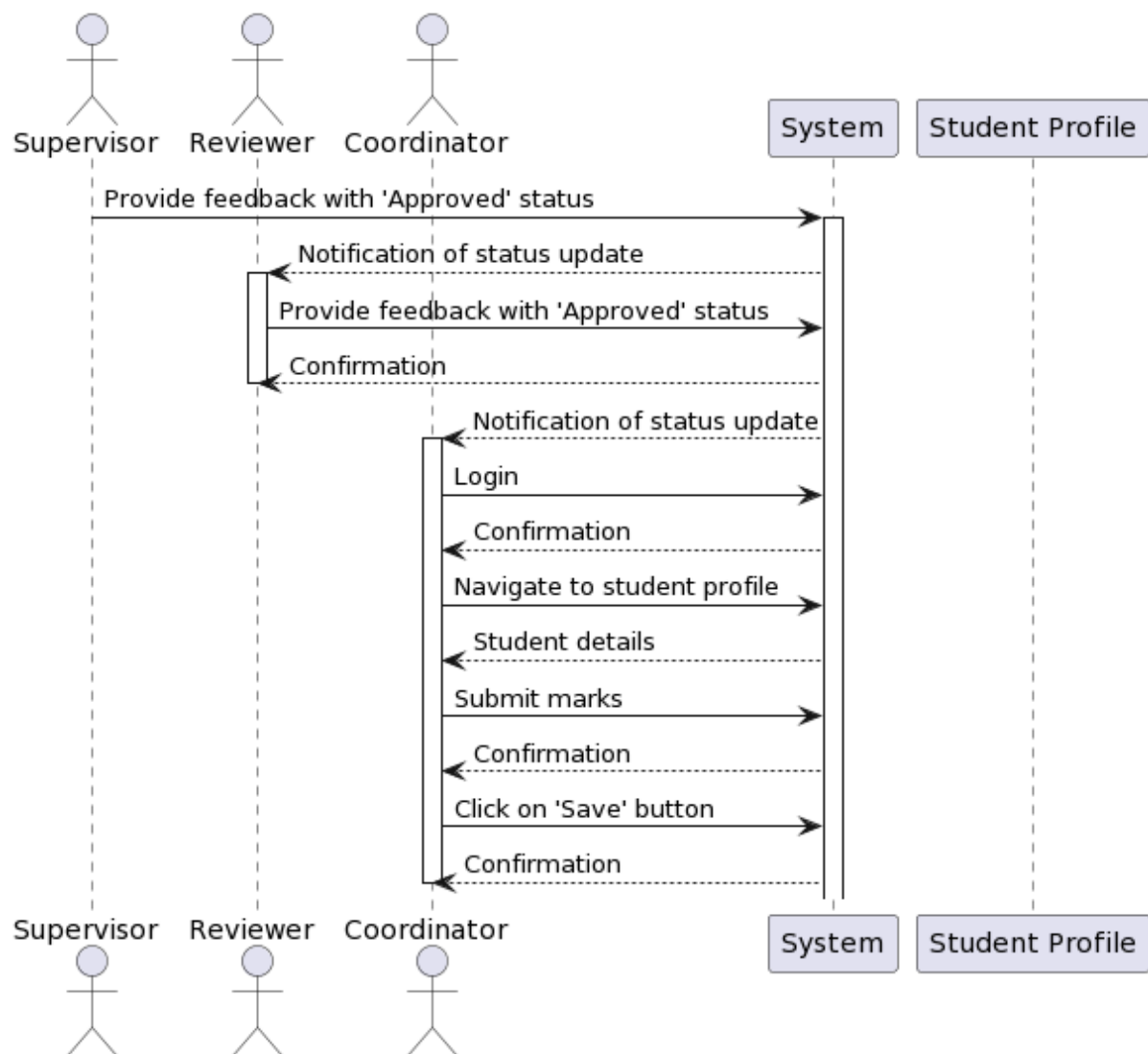


Figure 3.5.3 - Final Grading Generation Sequence Diagram

3.6 Activity Diagram

Activity diagrams are graphical representations used in software engineering to model the dynamic aspects of a system's behavior. They illustrate the flow of control or data between various activities or processes within a system. These diagrams are part of the Unified Modeling Language (UML) and are particularly useful for visualizing the procedural flow of activities, including decision points, parallel activities, and loops.

The following Figure 3.6.1, Figure 3.6.2 and Figure 3.6.3 are the corresponding activity diagrams for the above sequence diagrams with Figure 3.5.3, Figure 3.5.3 and Figure 3.5.3 respectively.

Activity Diagram 1 - Thesis Submission:

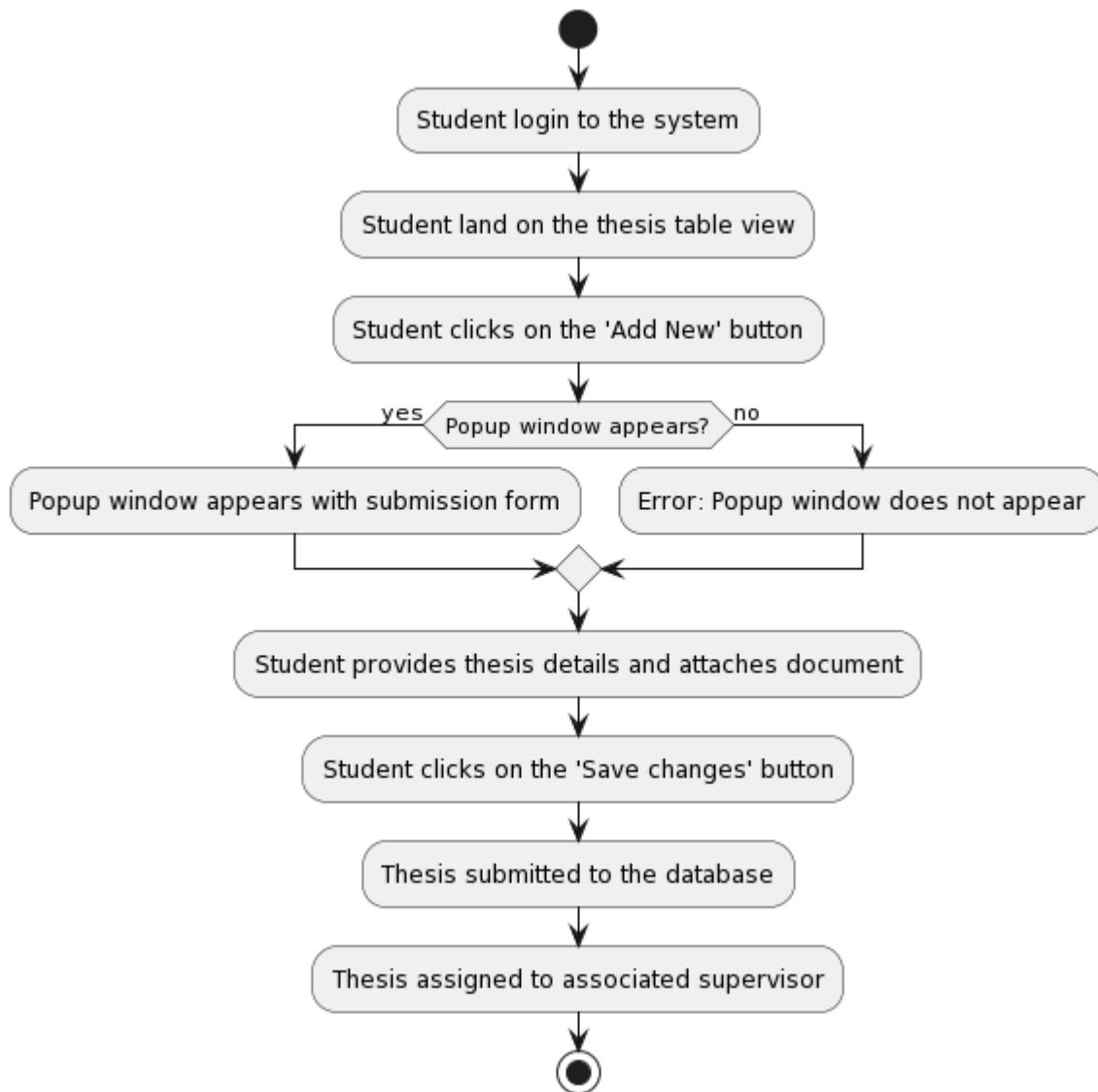


Figure 3.6.1 - Thesis Submission Activity Diagram

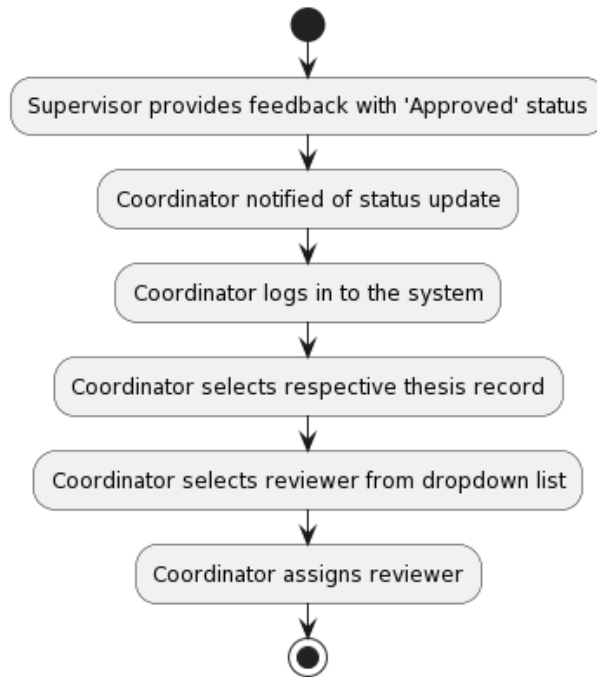


Figure 3.6.2 - Thesis Submission Activity Diagram

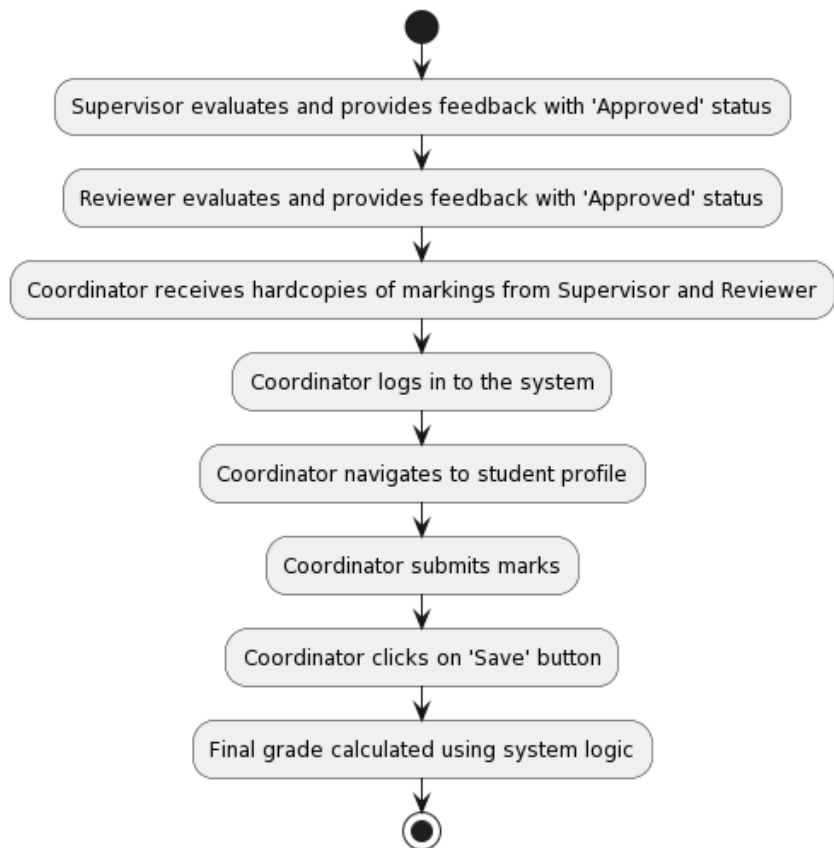


Figure 3.6.3 - Final Grade Generation Activity Diagram

Chapter 04: Implementation

4.1 Proposed Solution:

This system is planned to be implemented and delivered as a web based application because of the advantages that a web based automated thesis management system can provide over a traditional system. It will allow for easier access and greater convenience for students, supervisors and other stakeholders. The system will enable users to access the system from any computer with an internet connection, plus easily accessible from any location. Additionally, a web based system will enable more efficient management of theses and other documents, since access to the system is more secure, and information can be updated in real time. Overall, a web based system will provide improved efficiency, convenience, and collaboration for all involved, making it an ideal choice for the proposed project.

4.2. Software Development Life Cycle and Methodology:

In order to automate the PhD thesis management system for educational institutes, the **Prototyping development** methodology will be used. The Prototype Model is an iterative software development model, which is used to quickly develop a working prototype of a product.

This model is ideal for the implementation of an Automated PhD Thesis Management System, as it allows for quick development of a working prototype of the system, enabling the stakeholders to assess the system and make necessary modifications before the completion of the project.

This model also allows for early testing of the system, which will help to identify any potential issues before they become more serious. Additionally, since the prototype model is iterative in nature, it makes it easier to make changes to the system during development, as the changes can be tested and modified quickly. Finally, the Prototype Model is cost effective and results in shorter development cycles, allowing for a faster completion of the project.

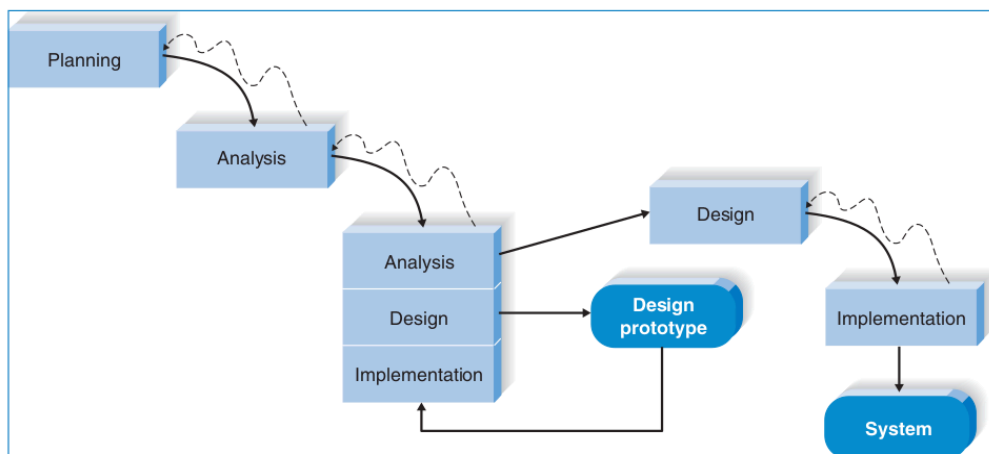


Figure 4.2.1 - Prototyping Software Development Life Cycle

4.3 System Architecture

Client-server architecture will be used for the implementation of this project as it is an ideal platform for implementing an automated thesis management system because it provides the necessary scalability and reliability needed for such a system. Client-server architecture allows the creation of multiple user interfaces for the different processes associated with the management system. Furthermore, the use of a server in a client-server architecture allows for the storage of large amounts of data that can be accessed by multiple clients. Additionally, the use of a client-server architecture can provide the ability to quickly and reliably update the system when needed. Finally, the use of this architecture allows for the separation of the user interface from the business logic, which can facilitate the development of an automated thesis management system.

4.4 Technology Stack

The planned technological stack for the project is as follows:

- Frontend: HTML, CSS
- Backend: PHP
- Server: Apache
- Database: PHPMyAdmin

Frontend: HTML and CSS are widely used and well-established technologies for building web-based user interfaces. They are relatively easy to learn and use, and there is a large community of developers and resources available for support. In addition, HTML and CSS are platform-agnostic and can be accessed from any device with a web browser, making them a flexible choice for building a system that needs to be accessible to a wide range of users.

Backend: PHP is a server-side scripting language that is widely used for web development. It is a popular choice for building web applications due to its ease of use, speed, and scalability. PHP is also open source, which means that it is free to use and there is a large community of developers and resources available for support.

Server: Apache is a widely used web server software that is known for its reliability, security, and performance. It is an open-source software, meaning that it is free to use, and has a large community of developers and resources available for support. Apache is also known for its flexibility and can be customized to meet the specific needs of your project.

Database: PHPMyAdmin is a web-based tool for managing MySQL databases. It is a popular choice for managing databases because it is easy to use, flexible, and provides a range of features for managing and manipulating data. PHPMyAdmin is also open source, which means that it is free to use and there is a large community of developers and resources available for support.

Overall, the selected technology stack for the proposed Automated PhD thesis management system is a solid choice because it is widely used, well-established, and has a large community of developers and resources available for support. The technologies are also free to use and open source, which can help to keep costs down and allow for greater flexibility in customization. Additionally, the technologies are all well-suited for building web-based applications and can be used together to create a cohesive and integrated system.

4.5 User Interfaces

The figure 4.5.1 is the login interface provided for all the user types to provide a secure gateway for users to access the system, requiring authentication through username and password fields.

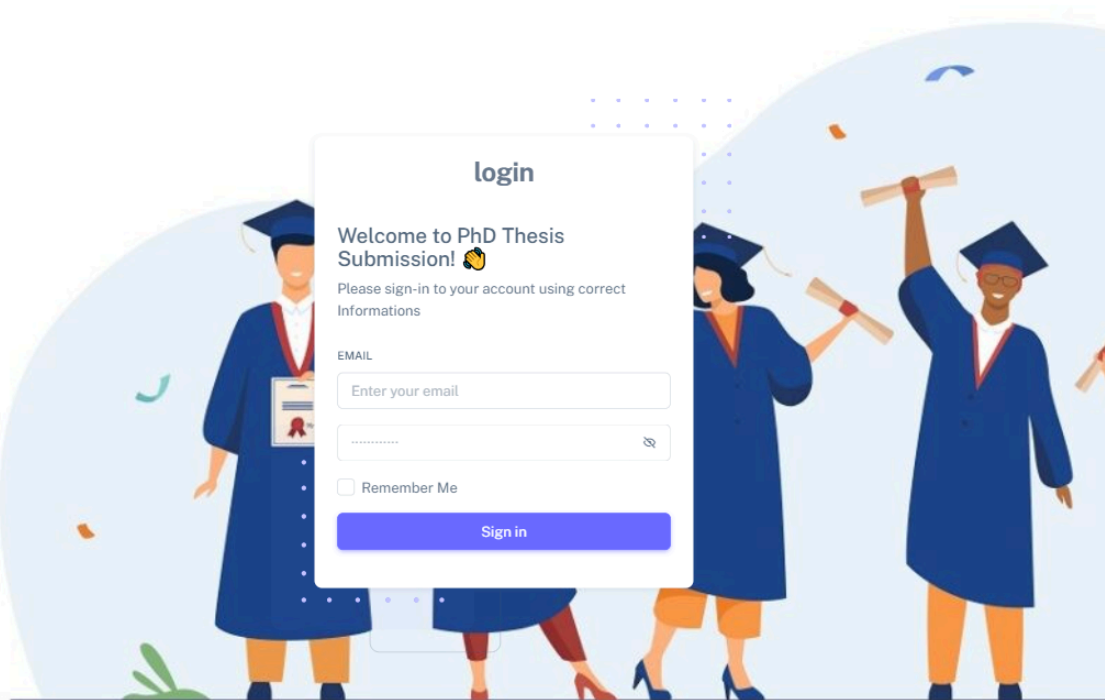


Figure 4.5.1 - Login Interface

The figure 4.5.2 is the admin landing page interface that serves as a centralized hub for major administrative tasks, offering quick access to system management functionalities and information. The left side panel had the menu to navigate to different functions.

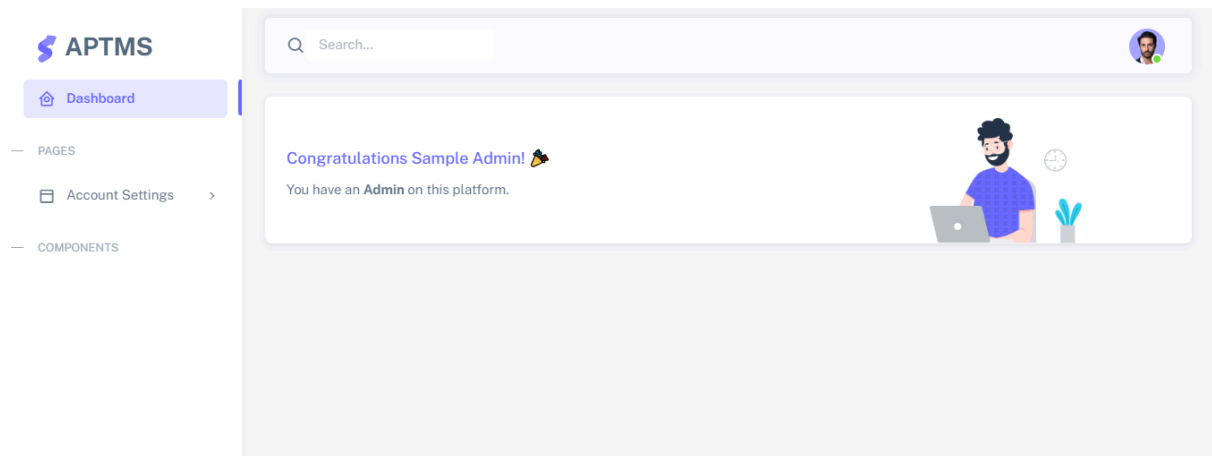


Figure 4.5.2 - Admin Landing Page Interface

The Figure 4.5.3 is the interface to empower administrators to seamlessly add new academic programs to the system, facilitating the expansion of available offerings. Also, it facilitates deleting the added programs from the existing programs list.

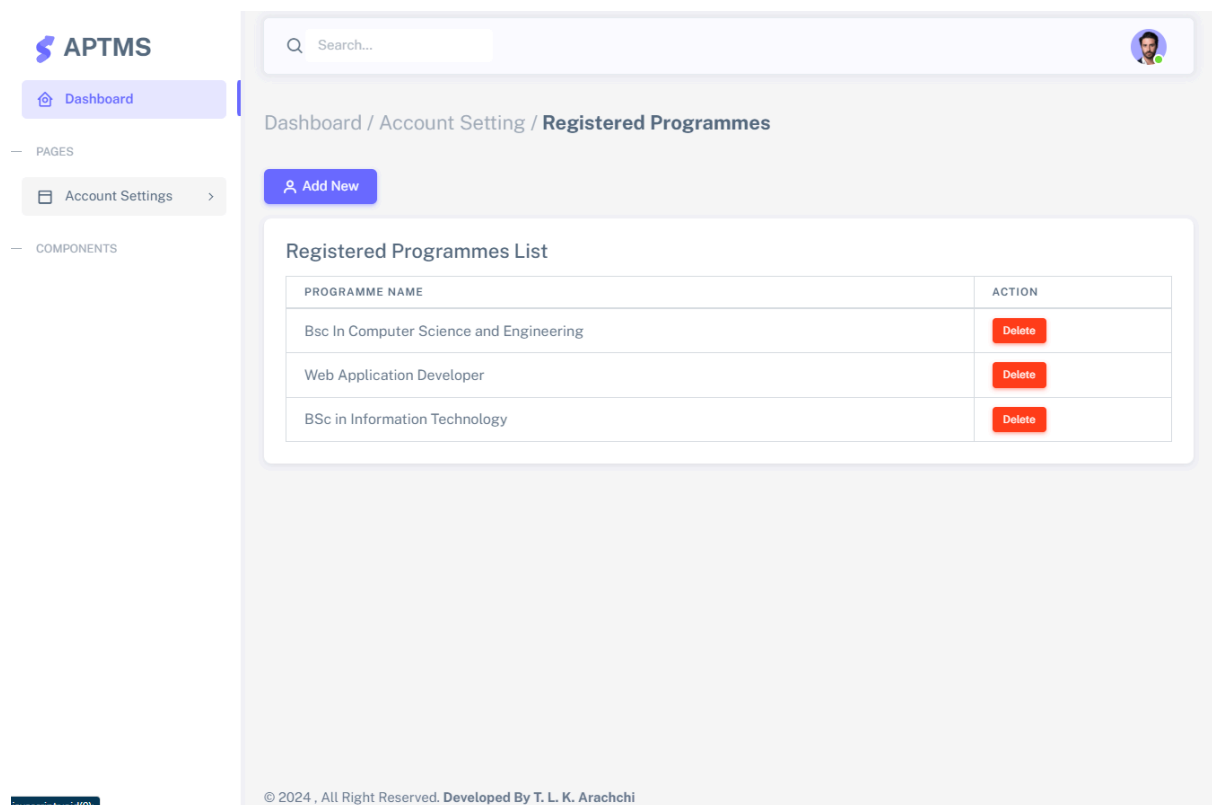


Figure 4.5.3 - New Program Adding Interface

Figure 4.5.4 is the student account registration interface provided for the admin where the admin can register students for all the programs by providing individual's details and by assigning the respective supervisors.

The screenshot shows the 'Student Registration Form' within the APTMS system. The interface includes a sidebar with 'Dashboard' and 'Account Settings' options. The main content area has a breadcrumb trail 'Dashboard / Account Setting/ Account Registration' and a tabbed interface with 'Supervisor', 'Students', 'Coordinator', 'Examiner', and 'Administrator' tabs. The 'Students' tab is active, displaying the 'Student Registration Form' with the following fields: 'STUDENT NAME' (with a dropdown for 'Mr.'), 'REGISTRATION NUMBER', 'REGISTERED DATE' (with a date picker), 'REGISTERED PROGRAMME' (with a dropdown 'Select the Program'), 'ASSIGNED SUPERVISOR' (with a dropdown 'Select the Supervisor'), 'EMAIL', and 'PASSWORD'. A 'Register' button is located at the bottom right of the form.

Figure 4.5.4 - Student Account Registration Interface

The admin utilizes the interface in Figure 4.5.5 to register supervisor accounts within the system, so that the registered supervisors can be assigned for the students at their account registration.

The screenshot shows the 'Supervisor Registration Form' within the APTMS system. The interface includes a sidebar with 'Dashboard' and 'Account Settings' options. The main content area has a breadcrumb trail 'Dashboard / Account Setting/ Account Registration' and a tabbed interface with 'Supervisor', 'Students', 'Coordinator', 'Examiner', and 'Administrator' tabs. The 'Supervisor' tab is active, displaying the 'Supervisor Registration Form' with the following fields: 'NAME' (with a dropdown for 'Mr.'), 'NUMBER', 'EMAIL', and 'PASSWORD'. A 'Register' button is located at the bottom right of the form.

Figure 4.5.5 - Supervisor Account Registration Interface

The Figure 4.5.6 is the interface given to the admin to register coordinators. Once the coordinator profile is created, then can log into the system to handle the intermediate tasks for students profile management.

APTMS

Dashboard

PAGES

Account Settings

COMPONENTS

Search...

Dashboard / Account Setting/ Account Registration

Supervisor

Students

Coordinator

Examiner

Administrator

Coordinator Registration Form

NAME

Mr.

EMPLOYEE NUMBER

EMAIL

PASSWORD

Register

Figure 4.5.6 - Coordinator Account Registration Interface

The Figure 4.5.7 is the interface implemented for the admin to view the list of all the created user profiles at a single place. Also, the admin can delete the created user profiles if needed within the same interface using the functionality implemented for the delete button.

APTMS

Dashboard

PAGES

Account Settings

COMPONENTS

PASSWORD

Register

Users List

Search:

Name	Email Address	Phone	User Role	Action
Mr. Admin	admin@gmail.com	001122	Admin	<button>Delete</button>
Mr. Coordinator	coordinator@gmail.com	12346	Coordinator	<button>Delete</button>
Mr. Coordinator 1	coordinator1@gmail.com	12344	Coordinator	<button>Delete</button>
Mr. Dr. Asanka Sayakkara	lakshikatimansi@gmail.com	123456	Supervisor	<button>Delete</button>
Mr. Examiner 1	examiner1@gmail.com	12347	Examiner	<button>Delete</button>
Mr. Examiner 2	examiner2@gamil.com	12348	Examiner	<button>Delete</button>
Mr. Examiner 3	examiner2@gmail.com	12349	Examiner	<button>Delete</button>
Mr. nazmul	example@domain.com	990999	Admin	<button>Delete</button>
Mr. Student User 1	teststudent1@gmail.com	112233	Student	<button>Delete</button>
Mr. Student User 2	teststudent2@gmail.com	112244	Student	<button>Delete</button>

Showing 1 to 10 of 10 entries

© 2024 , All Right Reserved. Developed By T. L. K. Arachchi

Figure 4.5.7 - Created Accounts List Table View Interface

31

The Figure 4.5.8 is the interface designed for the student users where each student can see a summary of the last submitted thesis document version. This is the landing page for the student once logged in.

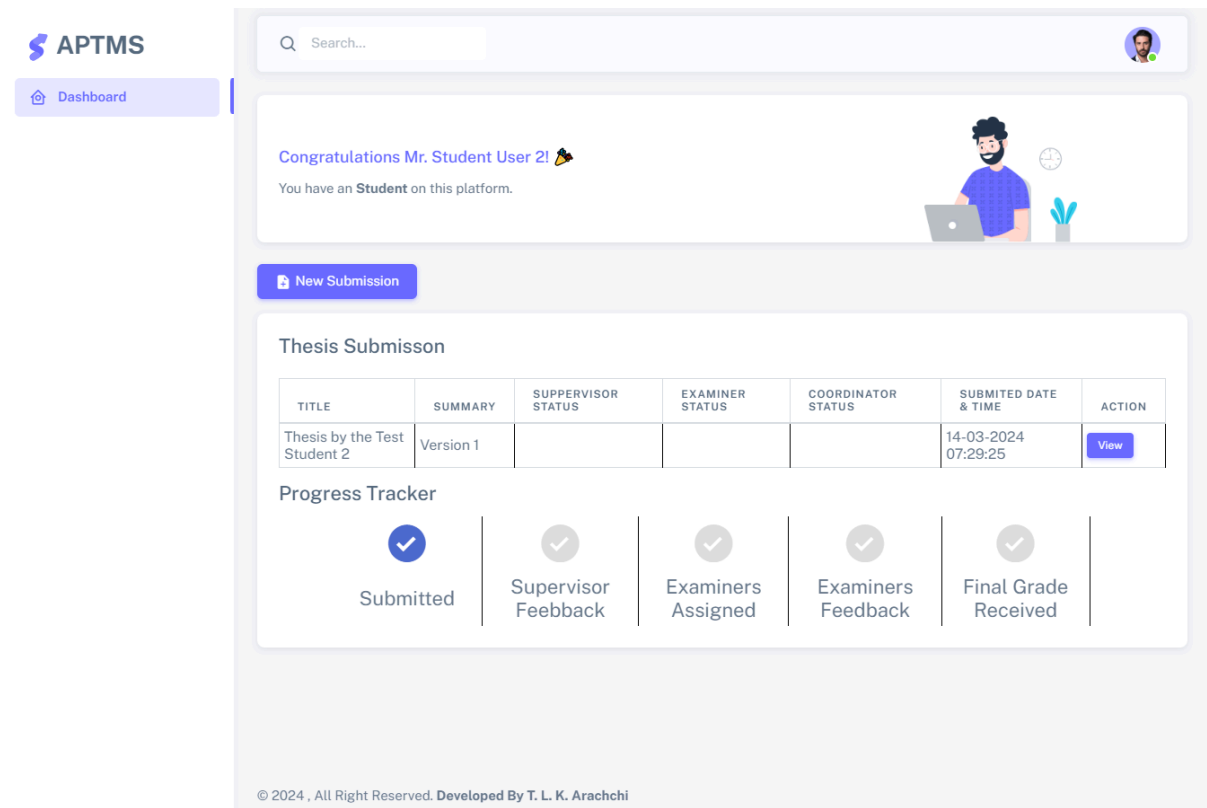


Figure 4.5.8 - Student Submitted Thesis View Interface

The Figure 4.5.9 below is the interface designed for students for the submission of new thesis document versions. There, the student can provide the thesis title, summary for the submission and attach the thesis file for their new submission

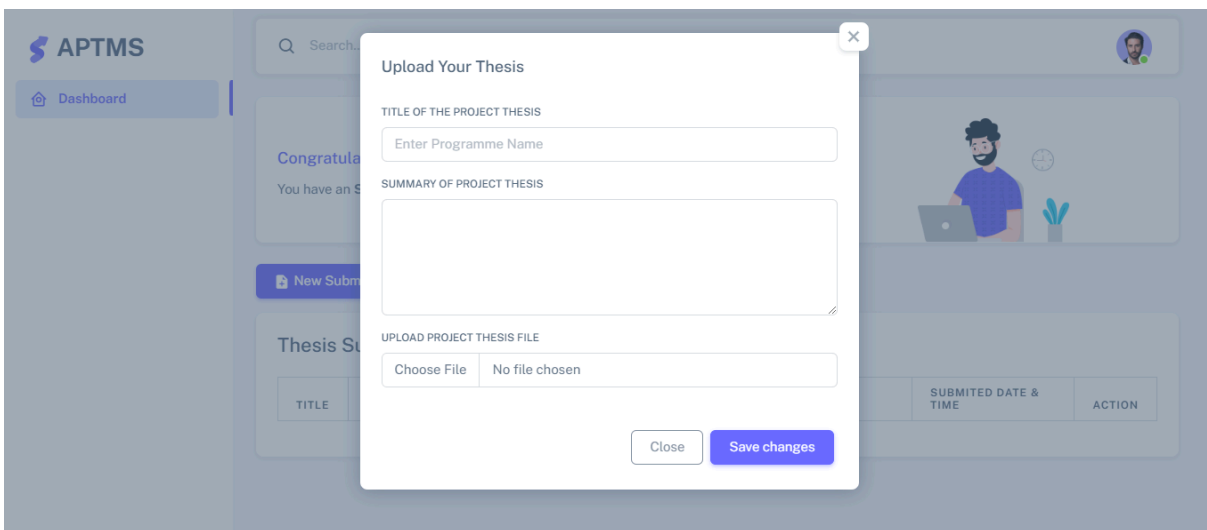


Figure 4.5.9 - Student New Thesis Submission Interface

Figure 4.5.10 shows the interface provided to students to overview the progress of the submitted thesis document where the completed stages will be clearly displayed to see the status of the evaluation process.

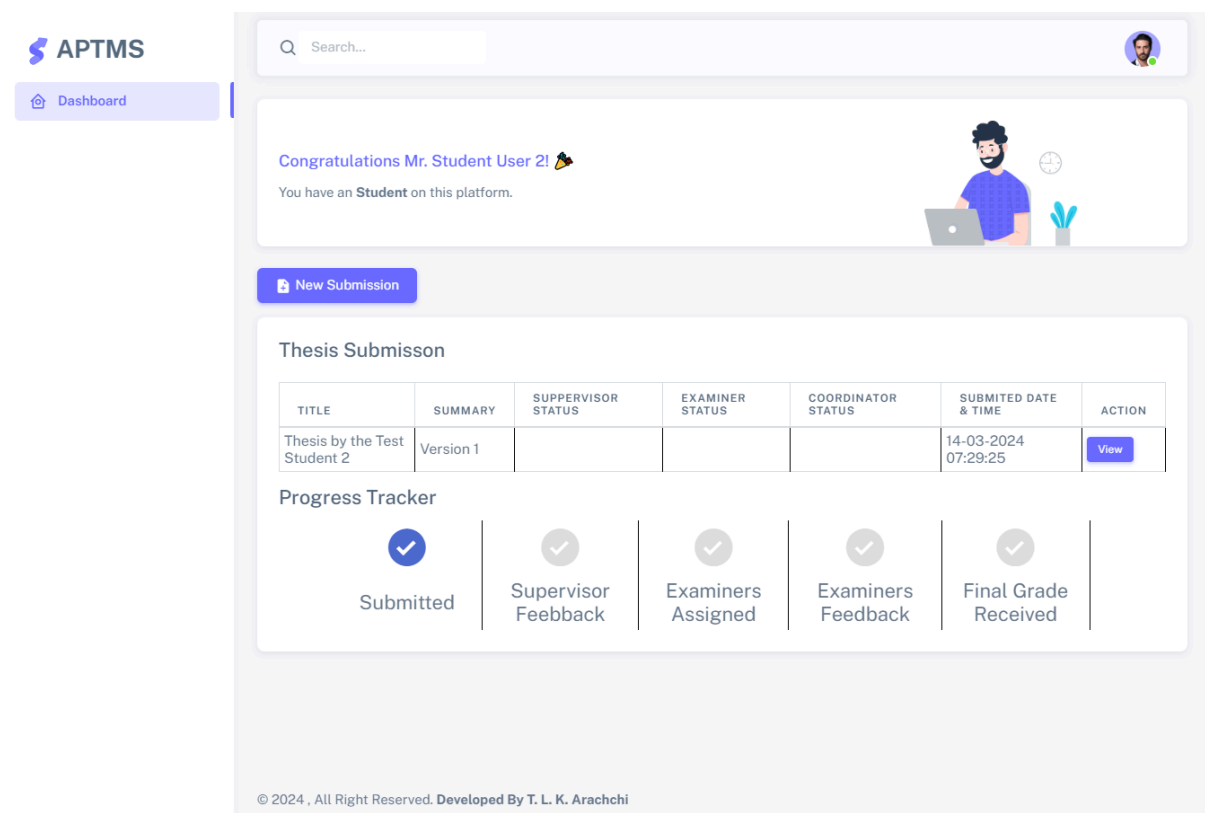


Figure 4.5.10 - Student Thesis Progress View Interface

Chapter 05: Evaluation

5.1 Introduction

This chapter is about how the evaluations of the implemented system were organized and conducted in order to ensure that the system meets the expected requirements and the desired purpose which it was planned for.

5.2 Software Testing

5.2.1 Testing Strategy

At the initial stage of planning the project, a test plan was created to validate the entire process of developing the software, also to verify that the ultimately implemented system is to the expected standards and requirements.

Objectives	<ul style="list-style-type: none">● Ensure the functionality, usability, and reliability of this web-based Automated PhDThesis Evaluation System through comprehensive testing.● Identify the defects in the system and suggest solutions for those.
Scope	<ul style="list-style-type: none">● Testing should cover all aspects of the implemented system, not limiting to the major functionalities of submission process, progress tracking, communication features, evaluation, and grading of thesis.
Testing Approach	<ul style="list-style-type: none">● A combination of different and relevant testing techniques should be applied for functional testing, user interface testing, integration testing, and performance testing.● Testing should happen throughout the project development cycle in each test level.
Resources	<ul style="list-style-type: none">● Test Environment● Testing tools● System users
Schedule	<ul style="list-style-type: none">● Unit testing: 8 weeks● Integration testing: 4 week● System testing: 1 week● User acceptance testing: 1 week

Deliverables	<ul style="list-style-type: none"> • Test plan • Test cases • Test results • Defect reports
Approvals	<ul style="list-style-type: none"> • System Users
Testing Sign-Off	<ul style="list-style-type: none"> • Test sign-off will be provided when all test cases have been executed, and if the system meets the defined acceptance criteria.

Table 5.2.1.1 - Test Plan for the evaluation of the implemented system

5.2.2 Test Cases and Results

This section displays the high level test cases implemented and executed to ensure the overall functionality of the system. Actual and expected results were tracked to keep a track of any discrepancy between the expected and actual behaviors within the system.

Test case ID: TC001		
Test Case Title: Student Submit the Thesis		
Test Case Summary: Each student should be able to submit their thesis document successfully.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Log in as a student. 2. Navigate to the Submissions page. 3. Fill in thesis details and upload related documents. 4. Click on the submit button. 	The thesis should be successfully submitted and stored securely in the database.	The thesis is successfully submitted and stored securely in the database.

Table 5.2.2.1 - Test Case Student Submit the Thesis

Test case ID: TC002		
Test Case Title: Create Student Account		
Test Case Summary: The Administrator should be able to create a new student account successfully.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as administrator. 2. Navigate to the user registration page. 3. Select the 'Student' tab for the account type. 4. Provide student details (e.g., name, registration number, email etc.). 5. Click on the 'Register' button. 	The student account should be created successfully with the entered details.	The student account is created successfully with the entered details.

Table 5.2.2.2 - Test Case Create Student Account

Test case ID: TC003		
Test Case Title: Create Supervisor Account		
Test Case Summary: The Administrator should be able to create a new supervisor account successfully.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as administrator. 2. Navigate to the user registration page. 3. Select the 'Supervisor' tab for the account type. 4. Provide supervisor details (e.g., name, employee number, email etc.). 5. Click on the 'Register' button. 	The supervisor account should be created successfully with the entered details.	The supervisor account is created successfully with the entered details.

Table 5.2.2.3 - Test Case Create Supervisor Account

Test case ID: TC004		
Test Case Title: Create Examiner Account		
Test Case Summary: The Administrator should be able to create a new examiner account successfully.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as administrator. 2. Navigate to the user registration page. 3. Select the 'Examiner' tab for the account type. 4. Provide student details (e.g., name, employee number, email etc.). 5. Click on the 'Register' button. 	The examiner account should be created successfully with the entered details.	The examiner is created successfully with the entered details.

Table 5.2.2.4 - Test Case Create Examiner Account

Test case ID: TC005		
Test Case Title: Create Coordinator Account		
Test Case Summary: The Administrator should be able to create a new coordinator account successfully.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as administrator. 2. Navigate to the user registration page. 3. Select the 'Coordinator' tab for the account type. 4. Provide student details (e.g., name, employee number, email etc.). 5. Click on the 'Register' button. 	The coordinator account should be created successfully with the entered details.	The coordinator account is created successfully with the entered details.

Table 5.2.2.5 - Test Case Create Coordinator Account

Test case ID: TC006		
Test Case Title: Track Progress Overview		
Test Case Summary: The student should be able to see the progress stage of their Thesis evaluation process.		
Pre-conditions: The logged in user should have an already done thesis submission.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as a student. 2. Navigate to the 'Submissions' view. 	The student should be able to see the progress view with the stages with their completed and incomplete statuses.	The student was able to see the progress of their thesis with completed and incomplete stages.

Table 5.2.2.6 - Test Case Track Progress Overview

Test case ID: TC007		
Test Case Title: View Updated Progress Overview		
Test Case Summary: The progress stages of the Thesis evaluation process should be updated corresponding to the stages completion.		
Pre-conditions: The logged in user should have an already done thesis submission.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as a student who had done the submission of the thesis. 2. View the progress stages. 3. No login as the supervisor. 4. Submit the feedback. 5. Again login to the same student account. 6. View the progress overview for the previous submission. 	The progress overview should have been updated by ticking the next stage completed for the supervisors feedback submission.	The progress overview is updated by ticking the next stage as completed after the supervisor's feedback submission.

Table 5.2.2.7 - Test Case View Updated Progress Overview

Test case ID: TC008		
Test Case Title: The supervisor views the detailed view of the thesis.		
Test Case Summary: The supervisor should be able to see the detailed view of the student's thesis submission.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as the supervisor. 2. Navigate to the thesis list view to see the submitted thesis by the students. 3. Select a thesis. 4. Click on the 'View' button of that record. 	The detailed view of the selected record should be successfully displayed.	The detailed view of the selected record is successfully displayed.

Table 5.2.2.8 - Test Case The supervisor views the detailed view of the thesis

Test case ID: TC009		
Test Case Title: Supervisor's feedback submission.		
Test Case Summary: The assigned supervisor should be able to provide the feedback to the student.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as the supervisor 2. Navigate to the thesis list view to see the submitted thesis by the students. 3. Select a thesis. 4. Click on the 'View' button of that record. 5. Then the detailed view should be displayed with the options to provide the text feedback and to submit any feedback document. 6. Type the text feedback. 	<p>The supervisor should be able to submit the feedback as text and document successfully.</p> <p>The submitted text feedback should be displayed real time in the supervisors feedback view.</p>	<p>The supervisor was able to submit the feedback as text and document successfully.</p> <p>The submitted text feedback is displayed real time in the supervisors feedback view.</p>

7. Attach the feedback document. 8. Mark the status. 9. Click on the 'Submit' button		
--	--	--

Table 5.2.2.9 - Test Case Supervisor's feedback submission

Test case ID: TC010		
Test Case Title: The examiner views the detailed view of the thesis.		
Test Case Summary: The examiner should be able to see the detailed view of the student's thesis submission.		
Steps to recreate	Expected result	Actual result
5. Login as the examiner. 6. Navigate to the thesis list view to see the submitted thesis by the students. 7. Select a thesis. 8. Click on the 'View' button of that record.	The detailed view of the selected record should be successfully displayed.	The detailed view of the selected record is successfully displayed to the examiner..

Table 5.2.2.10 - Test Case The examiner views the detailed view of the thesis

Test case ID: TC011		
Test Case Title: Examiner's feedback submission.		
Test Case Summary: The assigned examiner should be able to provide the feedback to the student.		
Steps to recreate	Expected result	Actual result
10. Login as the examiner 11. Navigate to the thesis list view to see the submitted thesis by the students. 12. Select a thesis. 13. Click on the 'View' button of that record.	The examiner should be able to submit the feedback as text and document successfully. The submitted text feedback should be displayed real time in the examiners feedback view.	The examiner was able to submit the feedback as text and document successfully. The submitted text feedback is displayed real time in the examiners feedback view.

14. Then the detailed view should be displayed with the options to provide the text feedback and to submit any feedback document. 15. Type the text feedback. 16. Attach the feedback document. 17. Mark the status. 18. Click on the 'Submit' button		
---	--	--

Table 5.2.2.11 - Test Case Examiner's feedback submission

Test case ID: TC012		
Test Case Title: Student receive and view the feedback received from the supervisor		
Test Case Summary: The student should receive all the feedback received from the supervisor and examiners.		
Pre-Conditions: <ol style="list-style-type: none"> 1. The student should have submitted a thesis for evaluation. 2. The supervisor should have received the submitted document from the student. 		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as the supervisor 2. Provide the feedback on the thesis both in text format and through attaching a document 3. Click on the submit button 4. Now login as the student 5. Open the detailed view of the last submission 	The provided feedback from the supervisor should be displayed with both text feedback and submitted feedback documents.	The student was able to view the supervisor provided both text and document feedback.

Table 5.2.2.12 - Test Case Student receive and view the feedback received from the supervisor

Test case ID: TC013		
Test Case Title: Add new programme to the system		
Test Case Summary: Administrator should be able to add the new programs to the system.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as the Administrator 2. Navigate to the Registered Programs View 3. Click on 'Add New' button 4. The popup view to add the New program should appear 5. Type the new programme's name 6. Click on the 'Add' button 	The newly entered program should appear in the programs list view.	The newly added program appears in the below programme's list view.

Table 5.2.2.13 - Test Case Add new programme to the system

Test case ID: TC014		
Test Case Title: Assign examiner		
Test Case Summary: Coordinator should be able to assign examiners to the supervisor approved thesis for further evaluation.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as the coordinator 2. Navigate to the 'Student Profiles' view 3. Select a student 4. Then the student's detail view should appear 5. Click on the 'Assign Examiner' button under the Examiner field 6. The set of available 	The selected examiner's name should appear under the Student's Profile - Examiners field.	The selected examiner's name appears under the Student's Profile - Examiners field.

<p>examiners should appear</p> <p>7. Select an examiner from the list</p> <p>8. Click on the 'Assign' button</p>		
--	--	--

Table 5.2.2.14 - Test Case Assign examiner

Test case ID: TC015		
Test Case Title: Assign supervisor		
Test Case Summary: The administrator should be able to assign the supervisor at the student account creation.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as the Administrator 2. Navigate to the Student account registration view 3. Move to the 'Supervisor' field and click on it 4. Available supervisors list should appear as a dropdown 5. Select a 'Supervisor' from the list. 6. Fill the other fields 7. Click on the 'Register' button 	<p>Once the user clicks on the 'Register' button, the selected supervisor should be successfully assigned to the student.</p>	<p>Once the user clicked on the 'Register' button, the selected supervisor was successfully assigned to the student.</p>

Table 5.2.2.15 - Test Case Assign supervisor

Test case ID: TC016		
Test Case Title: Upload the marks		
Test Case Summary: The coordinator should be able to submit the marks for the student user account.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as the coordinator 	<p>Entered marks should be saved under the respective</p>	<p>Entered marks were successfully saved under the</p>

2. Navigate to the 'Student Profile' view 3. Select the student profile 4. Detailed view of the student profile should appear to the coordinator 5. There the marking grid should appear at the bottom 6. Provide the marks 7. Click on the 'Save' button	student profile.	respective student profile.
--	------------------	-----------------------------

Table 5.2.2.16 - Test Case Upload the marks

Test case ID: TC017		
Test Case Title: Calculate the Grade		
Test Case Summary: The accurate grade should be calculated corresponding to the submitted marks.		
Steps to recreate	Expected result	Actual result
1. Login as the coordinator 2. Navigate to the 'Student Profile' view 3. Select the student profile 4. Detailed view of the student profile should appear to the coordinator 5. There the marking grid should appear at the bottom 6. Provide the marks 7. Click on the 'Save' button	Then the grade should be calculated with respect to the entered marks and display the final grade.	The final grade was generated after the calculation with respect to the entered marks and displayed.

Table 5.2.2.17 - Test Case Calculate the Grade

Test case ID: TC018		
Test Case Title: The student receives the grade.		
Test Case Summary: The student should receive the final grade at the submission of the final marks.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as the coordinator 2. Navigate to the 'Student Profile' view 3. Select the student profile 4. Detailed view of the student profile should appear to the coordinator 5. There the marking grid should appear at the bottom 6. Provide the marks 7. Click on the 'Save' button 8. Now login as the Student 9. Go to the submitted thesis status view 	The received grade should be displayed under the Thesis status view.	The received grade is displayed under the Thesis status view.

Table 5.2.2.18 - Test Case The student receives the grade

Test case ID: TC019		
Test Case Title: Update assigned the examiners		
Test Case Summary: The coordinator should be able to update the assigned examiners.		
Steps to recreate	Expected result	Actual result
<ol style="list-style-type: none"> 1. Login as the coordinator 2. Navigate to the 'Student Profiles' view 3. Select a student 4. Then the student's detail view should 	The newly selected examiner's name should appear under the Student's Profile - Examiners field.	The newly selected examiner's name appears under the Student's Profile - Examiners field.

appear 5. Click on the 'Assign/Edit Examiner' button under the Examiner field 6. The set of available examiners should appear 7. Remove the already assigned examiners. 8. Select an examiner from the list 9. Click on the 'Assign' button		
--	--	--

Table 5.2.2.19 - Test Case Update assigned the examiners

Test case ID: TC020		
Test Case Title: Receive notification reminders for the thesis evaluation		
Test Case Summary: The examiners should receive the notifications as a reminder for the thesis evaluations.		
Steps to recreate	Expected result	Actual result
1. Login as the supervisor 2. Provide the feedback and submit it with the 'Approved' status 3. At the submission of the Supervisors feedback, the thesis should be forwarded to the Coordinator 4. The login as the coordinator 5. Assign an examiner to the thesis 6. Save the changes done on the student profile	The assigned examiner should receive an email notification on this.	The examiner received the email notifying that a new thesis has been assigned for evaluation.

Table 5.2.2.20 - Test Case Receive notification reminders for the thesis evaluation

Chapter 06: Conclusion and Future Works

6.1 Overview

This section provides an evaluation and assessment of the Web-Based Automated PhD Theses Management System targeted for educational institutes and universities. The objective is to evaluate the system's performance, determine the extent to which project goals have been achieved, and identify potential areas for improvement. Also, the lessons learnt during the development process and suggestions for future enhancements will also be discussed here.

6.2 Problems Encountered and Lessons Learned

Through the life cycle of this project Automated PhD Theses Management System, provided invaluable opportunities to acquire knowledge about various technologies and how to apply those technologies in the implementation of a web based system. Throughout the project, several important lessons were learned. Emphasis was placed on the significance of user interface design, as intuitive and visually appealing interfaces significantly contribute to user satisfaction.

Creating a robust and efficient database structure was highlighted during the database design phase. Effective time management and balancing project work with other responsibilities and studies were crucial skills developed throughout the project. Overall, the project provided valuable learning experiences in web development and project management.

6.3 Critical Evaluation of the System

This Automated PhD Theses Management System, in its current state, effectively addresses the challenges students and other stakeholders face in effective management of the evaluation process of the theses of the students. It provides a user-friendly interface for creating and managing profiles, maintaining different user roles and their functionalities, thesis submission, tracking evaluation process till the provision of the final grading on the evaluated theses. However, some areas can be improved to enhance the system's flexibility to perform in customized platforms, system's effectiveness and competitiveness in the market.

6.4 Future Works

The current system has successfully prioritized and addressed the basic workflow and core functionalities for the requirement and achieved the project objectives. Further enhancements can be made to meet evolving user expectations and industry demands.

Following are several potential areas that can be further improved to take this system to its next level:

1. Improvement of the user interface designs by enhancing the interface's visual appeal, integrity, and intuitiveness would further elevate the user experience.
2. Incorporating advanced features such thesis template maintenance
3. Provision of multiple grading or marking structures can be considered as another area of improvement to make the system more competitive and to immerse in this field as a mandatory tool to be used.
4. Increasing the efficiency of the system by including more functionalities to be handled within the system itself other than handling middle tasks externally.
As an example, supervisors can provide their marks within the system itself in order to generate the final gradings.
5. Integration with external systems or platforms: Explore opportunities to integrate the system with external systems or platforms, such as virtual learning environments of the educational institutes.
6. Development of a dedicated mobile application: Create a mobile application to offer a seamless and optimized user experience on mobile devices, catering to the increasing popularity of mobile platforms.
7. Support for multiple languages: Enhance the system's accessibility and usability by providing multilingual support, enabling users from different regions to utilize the platform effectively.

These potential future improvements further enhance the system's functionality, user experience, and competitiveness in the education industry. Implementing these enhancements can be pursued in subsequent development phases, considering the availability of resources and the evolving needs of users.

References:

Garip, O., 2020. Web-based Thesis Workflow Management System. Youngstown State University.

Arslan, M., Demirci, G., & Kaya, A. (2018). An automated thesis management system for educational institutions. *Journal of Computer Science*, 14(10), 1583-1592.

Khan, M. A., & Khan, S. U. (2018). Automated thesis management system: A review. *International Journal of Advanced Research in Computer Science and Software Engineering*, 8(11), 2345-2352.

Kaushik, S., & Singh, S. (2017). Automated thesis management system: A need of the hour. *International Journal of Innovative Research in Computer Science and Engineering*, 5(6), 116-123.

Global ETD Search (no date). Available at: <http://search.ndltd.org/>. (Accessed: 2nd March 2023).

ETD Dissemination (no date). Available at: <https://about.proquest.com/en/dissertations/include-dissertations-and-theses/>. (Accessed: 2nd March 2023).

Thesis & Dissertation Services (TDS) - Texas A&M University-Commerce (no date). Available at: <https://inside.tamuc.edu/academics/graduateschool/thesis%20and%20dissertation%20services/default.aspx>. (Accessed: 5th March 2023).

Thesis management system (no date). Available at: <https://www.eur.nl/en/about-eur/vision/community-learning-and-innovation/innovation-projects/digitalisation-projects/thesis-management-system>. (Accessed: 6th March 2023).

Helmiawan, M.A. and Firdaus, Y. (2018) "Thesis Process Automation System," *ResearchGate* [Preprint]. Available at: https://www.researchgate.net/publication/328118384_Thesis_Process_Automation_System. (Accessed: 6th March 2023).

Habib, Md.R. et al. (2023b) "Design of a University Thesis and Project Automation System (UTPAS)," in *Lecture notes in networks and systems*. Springer International Publishing, pp. 969–978. Available at: https://doi.org/10.1007/978-981-19-7874-6_71. (Accessed: 6th March 2023).

Noel, J.C. G, Yu, W. and Tagle, P. (2014) “ThesisFS: Online Document Management System,” *Ateneo* [Preprint]. Available at:
https://www.academia.edu/2601142/ThesisFS_Online_Document_Management_System.
(Accessed: 8th March 2023).

Thesis Management System (no date). Available at:
<https://www.creatrixcampus.com/thesis-management-system>. (Accessed: 10th March 2023).