



Sri Lanka Institute of Information Technology

Information Technology Project (IT2080)

Assignment 3

Final Report

WD_B05_ITP_G3

Online Educational Institute Management System

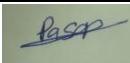
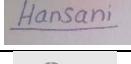
	Name with Initials	Registration Number	Contact Phone Number	Email
1.	Baddewithana P	IT21247804	0706370573	It21247804@my.sliit.lk
2.	Ilesingha I.T.S.	IT21206078	0702234039	it21206078@my.sliit.lk
3.	Rajawasan W.H.H.S.	IT21315664	0769649542	it21315664@my.sliit.lk
4.	Hansani K.J.M.	IT21222672	0762582467	it21222672@my.sliit.lk
5.	De Silva K.T.S.	IT21328916	0772021034	it21328916@my.sliit.lk
6.	Liyanage L.D.P.D.	IT21327780	0717656875	it21327780@my.sliit.lk
7.	Jayasooriya J.M.D.T.	IT21326868	0704642020	it21326868@my.sliit.lk
8.	De Zoysa A.P.S.	IT21219634	0719954022	it21219634@my.sliit.lk

Declaration

This project report, or a section of it, was not a copy of a document created by any organization, university, or other institute, or a former SLIIT student project group, and it was not obtained from the internet or other sources.

Project Details

Project Title	Online Educational Institute Management System
Project ID	Batch 5.1 G03

Student ID	Name	Signature
IT21247804	Baddewithana P	
IT21206078	Ilesingha I.T.S.	
IT21315664	Rajawasan W.H.H.S.	
IT21222672	Hansani K.J.M.	
IT21328916	De Silva K.T.S.	
IT21327780	Liyanage L.D.P.D.	
IT21326868	Jayasooriya J.M.D.T.	
IT21219634	De Zoysa A.P.S.	

Date: 19/05/2023

Abstract

Nawamaga was founded in 2017 as a single Advanced Level Chemistry class. However, due to its reputation for high-quality education, the institution has since expanded to offer classes at the primary, ordinary, and advanced levels. The institution has grown rapidly over the years and now has two branches located in urban areas and over 10000 students enrolled. In the past Nawamaga Institute ran everything manually. To find the best solution for them, we first did a requirement-gathering analysis by visiting their Institute and observing their current process. Since we saw the difficulties with handling physical documents and considering the advantages of an online system, we decided to create a web application to handle the process of the Institute.

The web application was built using MERN stack technology. The system is made up of eight major components that support each individual functionality, Student Management, Staff Management, Notice & Timetable Management, Module Management, Financial Management, Quiz Management, Assignment Management, and Support Services Management.

Acknowledgment

First, many thanks to Mr. Samantha Rajapaksha, the lecturer in our group, for his unconditional support, and for his patience, motivation, dedication, and huge experiences. His advice was valuable throughout the documentation and implementation of this project.

Apart from the adviser, it's necessary to thank the rest of the lecturers' and the instructors' panel for their assistance, guidance, support, and intelligent remarks.

Mr. Dhevin Ekanayake, owner of Nawamaga Monaragala, and his outstanding institute team deserve special thanks for providing this amazing opportunity and collaborating all through the requirement analysis and project journey.

And finally, a special thanks to my fellow WD_B05_ITP_G03 group members who have supported me from the beginning to the end of the project.

Table of Content

Contents

Declaration	1
Abstract	2
Acknowledgment.....	2
Table of Content.....	3
List of Tables	6
List of Figures.....	6
Chapter 1. Introduction	7
1.1 Background.....	7
1.2 Problem and Motivation	7
1.3 Literature Review	10
1.4 Aims and Objectives	12
1.4.1 Aims	12
1.4.2 Objectives	12
1.5 Solution Overview.....	13
1.6 Methodology.....	14
1.6.1 Design Methods.....	14
1.6.2 Technologies	14
1.6.3 Tools	15
1.6.4 Testing Methods.....	15
1.6.5 Integration Methods.....	15
1.7 The Structure of the Report.....	16
1.8 GitHub Repository Link	16
Chapter 2. Requirements	17
2.1 Stakeholder Analysis	17
2.2 Requirements Analysis	18
2.2.1 Functional Requirements	18
2.2.2. Non-Functional requirements	19
2.2.3 Technical requirements	19
2.3 Requirements Modeling	20

2.3.1 User Stories.....	20
2.3.2 Use Case Diagram	21
2.3.2 Use Case Scenario	22
Chapter 3. Design And Development	35
 3.1 System Architecture.....	35
 3.2 Activity Diagrams	36
 3.3 Sequence Diagrams.....	45
 3.4 ER Diagram.....	56
 3.5 Class Diagram.....	57
 3.6 User Interfaces	58
3.6.1 Common UI – IT21247804.....	58
3.6.2 Student Management – IT21247804.....	59
3.6.3 Support Services Management – IT21206078.....	59
3.6.4 Timetable and Notice Management – IT21315664.....	60
3.6.5 Class Fee Management – IT21222672.....	60
3.6.6 Module Management – IT21327780.....	61
3.6.7 Assignment Management – IT21326868	61
3.6.8 Staff Management – IT21219634	62
3.6.9 Quiz Management – IT21328916	62
 3.7 Implementation	63
3.7.1 Implementation Structure.....	64
3.7.2 Database Connection.....	65
3.7.3. Code Special Algorithms	66
Chapter 4: Testing.....	68
4.1 Student Management Test Cases	68
4.2 Support Services Test Cases	72
4.3 Timetable and Notice Management Test Cases	74
4.4 Class Fee Management Test Cases	76
4.5 Module Management Test Cases	77
4.6 Quiz Management Test Cases	79
4.7 Assignment Management Test Cases.....	80
4.8 Staff Management Test Cases	83

Chapter 5: Evaluation and Conclusion.....	84
 5.1 Evaluation.....	84
5.1.1 Introduction	84
5.1.2 Evaluation Techniques.....	84
5.1.2.1 Database testing	84
5.1.2.2. Focus group.....	84
5.1.2.3. Inspection.....	85
5.1.2.4 Walkthroughs	85
5.1.3 Procedures of testing.....	86
5.1.4 User Evaluation	88
 5.2 Conclusion	89
Chapter 6: References.....	90
References	90
Chapter 7: Appendices	91
7.1 Appendix.....	91
7.2 Appendix	92
Gannt Chart.....	92

List of Tables

Table 1 Technologies Used	14
Table 2 Evaluation Table.....	88

List of Figures

Figure 1:0:1System Overview	13
Figure 2:0:1 Onion Diagram	17
Figure 2:0:1Usecase Diagram.....	21
Figure 3:0:1MERN stack Architecture	35
Figure 0:2Class Diagram	57
Figure 0:3 Login	58
Figure 0:4Signup.....	58
Figure 0:5Admin Dashboard.....	58
Figure 0:6 Appointment	59
Figure 0:7 Feedback.....	59
Figure 0:8 View data.....	59
Figure 0:9 Support Services- form.....	59
Figure 0:10 Student View -Notice page.....	60
Figure 0:11 Admin View- Notice Page	60
Figure 0:13 Student View - Timetable	60
Figure 0:12 Admin View - Timetable.....	60
Figure 0:14 Admin View – Class Fees.....	60
Figure 0:15Add class fee and View class fees.	60
Figure 0:17 Student View Module page.....	61
Figure 0:16 Admin View - Module page.....	61
Figure 0:18 Student View - Assignment	61
Figure 0:19 Admin View - Assignment.....	61
Figure 0:20 Admin View	62
Figure 0:21 Add Salary	62
Figure 0:22 Admin View - Add Quiz	62
Figure 0:23 Admin View Quiz home page.....	62
Figure 0:24 Admin View Update Quiz.....	62
Figure 0:26 Folder Structure	64
Figure 0:1Kanban Board.....	91

Chapter 1. Introduction

1.1 Background

The client is Nawamaga Institute, an educational institute that currently handles everything manually and does not offer any online services like online payment, registration, and learning management system. They face difficulties in handling physical documents and are looking for a solution to manage their processes more efficiently. The client has agreed to work with the development team to create a web application to automate their processes and improve their services. The nature of client is an educational institution seeking to improve its operational efficiency and provide better services to its students.

The institution has a staff of teachers and administrators who manage day-to-day operations such as class scheduling, assignment creation and grading, and student record management, but the fact that the institution does not offer any online services suggests that it is lagging behind in terms of technology and infrastructure. This can lead to inefficiencies such as long processing times for administrative tasks and difficulties in managing student records. It also suggests that the institution may be missing out on potential opportunities to attract and retain students who prefer the convenience of online services.

Nawamaga demonstrates its willingness to adapt to changing trends and improve its offerings to students by looking for a solution to automate its processes and provide online services. The decision to work with a development team to create a web application using the MERN technology stack demonstrates a desire to use the latest technology and best practices in software development.

1.2 Problem and Motivation

Currently, Nawamaga Institute is running everything manually. Covid pandemic situation period they launched an online payment system through their website for the student that's why curfew times anyone can't come to the institute to pay monthly fees. but at this present, that system not currently working now, and It doesn't offer any online services such as an Assignment management system, Quiz management system, Student management system, Support services management system, Timetable & notice management system, Staff management System, and Module Management system. They need to launch a new system for these problems.

- Registrations & documentation are done manually.

If someone wants to register for a new class, they should come to the institute physically. Also, they must fill in registration forms manually and hand them over to the Institute. it is a very difficult situation for students far from Monaragala. The institution has a staff of teachers and administrators who manage day-to-day operations such as class scheduling, assignment creation and grading, and student record management, but the fact that the institution does not offer any online services suggests that it is lagging in terms of technology and infrastructure. This can lead to inefficiencies such as long processing times for

administrative tasks and difficulties in managing student records. It also suggests that the institution may be missing out on potential opportunities to attract and retain students who prefer the convenience of online services.

- Implementing online services in the institution can bring benefits to both students and the administration. For students, it will provide convenience by eliminating the need to travel and physically fill in forms. It will also provide accessibility to those who are far away. For the administration, it will streamline administrative processes, resulting in faster processing times, more accurate record keeping, and a reduction in errors. Automation of routine tasks will free up time for teachers and administrators to focus on more important work, such as student engagement and academic excellence. Finally, the use of online services will also help the institution attract and retain tech-savvy students.
- Difficulty in conducting exams.

Since the exam is conducted at least once a month, it costs a lot of paper, and it is difficult to find papers according to the current situation in the country, so having to spend a lot of money on it is a loss for the institution. As the institute hopes to provide education for students outside Monaragala, the fact that students outside Monaragala are not able to attend the examinations conducted by their institute on time is one of the issues.

- By transitioning to online exams, the client can experience a range of benefits, including reduced costs, increased accessibility, improved efficiency, enhanced flexibility, and improved security. Online exams can help the institution save money on paper and printing costs, and reduce the time and labor required to administer exams. They also offer the opportunity for students who are outside of Monaragala to participate in exams at the same time as students who are physically present, increasing access to education. Additionally, online exams can be administered more efficiently, providing faster feedback for students, and improving operational efficiency. They also offer greater flexibility in terms of scheduling and location, reducing the burden on students and allowing them to better manage their schedules. Lastly, online exams can provide better security features, such as randomized question orders, question banks, and timed tests, helping to prevent cheating and ensuring the integrity of the exam.
- Support services are done manually.

Students and parents must come and make inquiries to the institution to get the services and information they need. Since there is only one person for the support service, most of the time students and parents come at the same time, so they must spend a long time in queues. The available time will be wasted and due to the job work of some parents, they will not be able to come to the institution to get some information.

- by building an online institute, the institution can provide a range of benefits to students and parents, including 24/7 access to information, efficient communication, flexibility for working parents, increased engagement, and cost savings. These

benefits can help to improve the overall customer experience and attract and retain students in the long run.

- Limit reach & High overhead costs.

Nawamaga education institute faces some limitations in their capacity to accommodate a larger number of students due to the constraints of their physical infrastructure for some Subjects. Classroom and lab space, limited faculty availability, scheduling conflicts, and resource limitations are some of the reasons that limit the number of students who can access education in these institutions. Maintaining physical infrastructure is a significant expense for educational institutions, adding to their overhead costs. The costs of utilities, maintenance of equipment, and construction and maintenance of buildings can be prohibitively expensive. High overhead costs can strain finances and result in reduced student programs, resources, and class development investments.

➤ By building an online institute system, Nawamaga education institute can overcome the limitations of physical space and faculty availability for high-demand subjects. This will lead to increased enrollment and revenue and expand the institute's reach beyond its physical location. Additionally, an online institute system can significantly reduce overhead costs by eliminating the need for physical infrastructure and utilities, freeing up resources that can be invested in student programs and class development. Overall, an online institute system can provide Nawamaga education institute with the opportunity to expand, thrive and provide a more diverse academic offering.

- Difficult marketing campaign.

There is a shortage of students for some subjects even though there are more students for some subjects. The reason for this is that students from outside Monaragala are not able to come to the institution and participate in educational activities. and another reason is the competitive tuition of other teachers in the country. Since many teachers have become Converted to online education, the marketing business of their educational institution has become fail to the extent that students participate in those online educational institutions for their respective subjects.

➤ Building an online institute system can help to address the shortage of students for certain subjects by providing access to a wider pool of students. This can lead to increased enrollment and revenue. Offering competitive tuition rates for online courses can help to retain existing students and attract new ones. Online education provides the opportunity to reach students who may live too far away to attend in-person classes, increasing the diversity of the student body. Embracing online education can also lead to a more innovative and dynamic learning environment by staying up to date with current trends and technology. [1]

1.3 Literature Review

There are two other solutions that can fulfill clients' requirements: Updating the existing system and purchasing ready-made system. Each of these solutions has its own benefits and drawbacks as follows.

Updating Existing System:

To ensure that this solution satisfies the client's new expectations, the developing team must analyze the existing system and update it. Online examination management, Parental management, Registration and login management, and Support services are necessary additions to this system.

Benefits:

- Economical: Modernizing an old system could be less expensive than creating a brand-new one from the beginning.
- Familiarity: Since the current system is currently in use, employees and students are already familiar with it, which may mean that no further training is required.
- Preserves data: Updating the current system makes sure that old records and historical data are not lost.

Drawbacks:

- Limited functionality: The current system might be functionally restricted, which could make it challenging to satisfy future requirements.
- Lack of Adaptability: The current system could not be adaptable or able to interface with other systems or processes.
- Technical issues: Updating an existing system can be a challenging and time-consuming process that requires a high level of technical understanding. It can also be challenging if it is not done by the system's developers. [2]

Purchasing Ready-made System:

On the internet, pre-built Web apps with a variety of features are for sale. The customer has the option of making purchases that suit their demands.

Benefits:

- Speed: Buying a pre-built system could be quicker than designing a brand-new system.
- Features: An acquired system might offer more powerful features and functions than the one it replaces, which would make managing student data and processes easier.
- Support: Many systems that are purchased include technical support, which can be useful in resolving technical problems or difficulties.

Drawbacks:

- Price: Buying a ready-made system can be expensive, particularly if it needs to be customized to fit certain needs.
- Restricted adaptability: Ready-made systems might not be able to work with other systems or procedures, which could be problematic if the Institute's requirements change in the future.
- Implementation issues: Pre-built systems may not be delivered as a functional web-based application. In that kind of circumstance, Admin needs more assistance from a knowledgeable individual. [3]

Why building a new System would be better than using the existing system:

The current system was developed back then in the pandemic era, and it is only developed to maintain class fee management and module management. As the business grows over time functional requirements like Online examination management, Parental management, and Staff management and non-functional requirements like expandability, and security have emerged. To achieve all these functional and non-functional requirements updating the existing system or purchasing a ready-made system is not enough. As a result, the idea of developing a new system emerged. The requirements of the Institute can be met in the future by creating a new system that can scale as the business expands. Although developing a new system may be more costly and time-consuming, it can end up being the best alternative because it offers the most specialized and effective solution.

1.4 Aims and Objectives

1.4.1 Aims

Past months ago, Students are currently confronted with several obstacles, including maintenance of handwritten documents because of a lack of paper, transportation issues, and the COVID-19 pandemic.

The following solutions are used to solve these issues.

- Students encounter a variety of transport challenges because of limited fuel in the country's current circumstances. Because of this difficulty, Students can do their educational activities through our website.
- This application will be helpful to minimize the number of Students participating in physical classes because of fuel issues.
- Handwritten documents can be lost at any time. hence storing information in a database can be more secure than maintaining handwritten documents.
- Maintaining a database is easier than using handwritten documents.

1.4.2 Objectives

- Always try to provide good education to students for the lowest price.
- Try to seek bigger and new opportunities because of this integrated system and this system helps the company to make virtual appointments between students, teachers, and Parents.
- To provide a convenient payment gateway for students
- To expand a large online student base through this system

1.5 Solution Overview

As a result of the problems mentioned above, we decided to build a web application for the Institute to manage all the necessary sections. The Online Educational Institute system is a good, user-friendly platform that automates and allows easy to manage the company tasks daily. This system's well-designed and adaptable functionalities make it appropriate for computerizing all the institute processes such as Student management, Staff management, Timetable and Notice management, Class Fee management, Module management, Quiz management, Assignment management, and Support Services Management

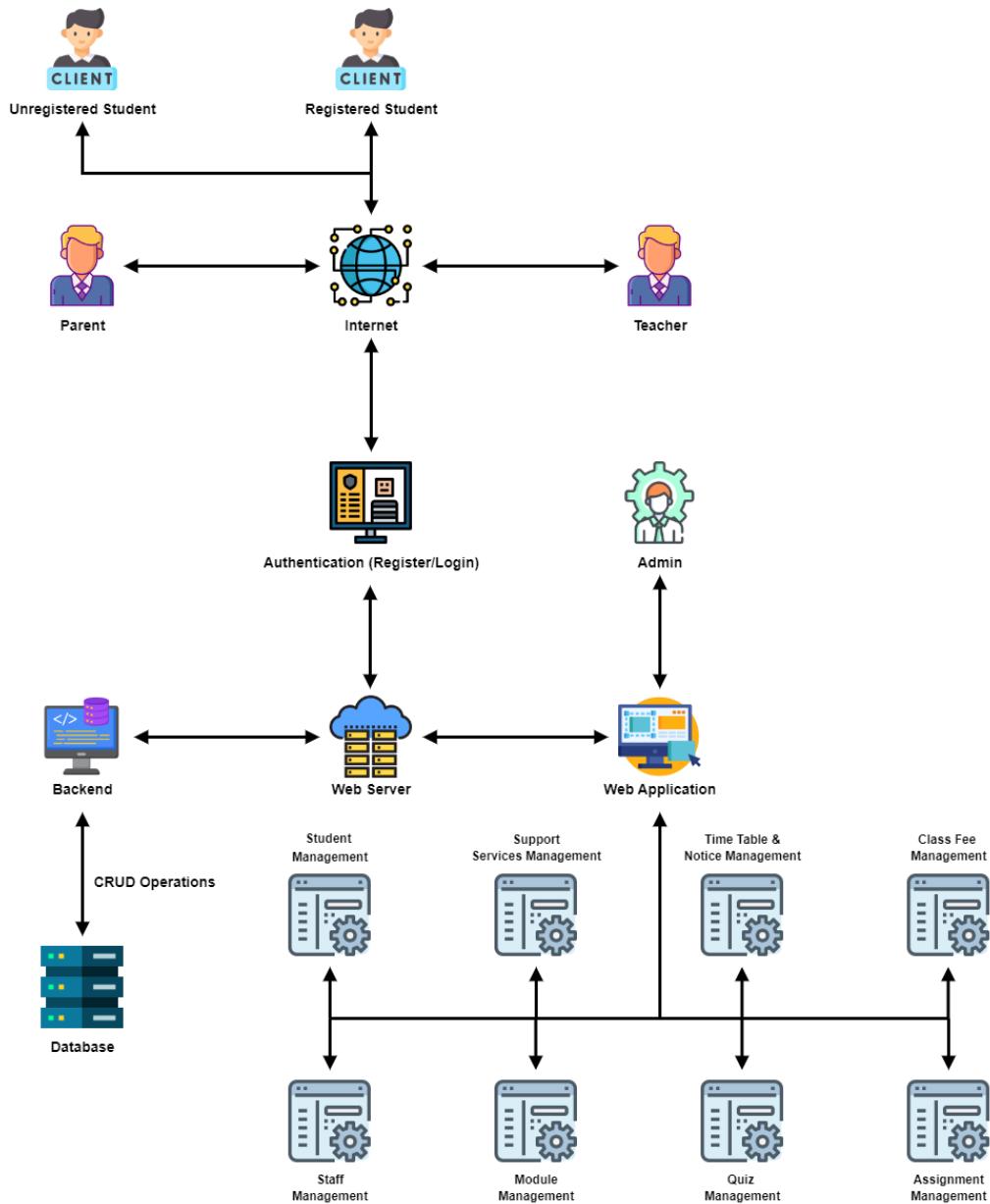


Figure 1:0:1 System Overview

1.6 Methodology

1.6.1 Design Methods

For our project we are using the waterwall method as the design methodology. First, we gathered all the requirements we need to understand the project and we created a design as a blueprint. After that we are planning to do the implementation of the system. After completing the implementation, we are planning to do the verifications by testing the web application. If the system is not verified, we are fixing the errors that cause the verification to fail. After the verification is completed, we are planning to deploy the system so the Institute can use the system for their required purposes. After deploying, we have the responsibility of maintaining the system in its lifespan. If there is any error in the system, we are taking the necessary actions to fix the error and restore the web application to its optimal execution.

1.6.2 Technologies

For this project we are using several software. MongoDB is used in place of conventional relational databases, Express.js and Node.js are used to design web applications and React.js is used to design front-end development.

The following technologies are part of the MERN Stack.

- MongoDB – Database Management
- Express.js – Server-side Management
- Node.js – Server-Side Management
- React.js – Web Application Management

Table 1 Technologies Used

React.js	An open-source JavaScript package called React.js is used to create user interfaces for single-page applications. It manages the view layer for both online and mobile applications. Additionally, React.js enables us to design reused UI components.
MongoDB	An open-source NoSQL database management system is called MongoDB. NoSQL databases are used in place of conventional relational databases. Working with sizable, distributed data sets makes good use of NoSQL databases. MongoDB is a technology that can manage, store, and retrieve information that is document oriented.
Express.js	A Node.js is web application framework called Express.js offers a wide range of functionality for creating web and mobile applications. A single page, multipage, or hybrid web application can be created with it. It is a layer added to Node.js that aids in managing servers and routes.
Node.js	Due to its single-threaded nature, Node.js is typically used for non-blocking, event-driven servers. Although it was created with real-time, push-based architectures in mind, it is utilized for conventional web pages and back-end API services. [4]

According to the requirements we use several dependencies like Bootstrap, React-Scroll for the frontend and Noteman, dotenv for the backend.

1.6.3 Tools

In first, we used diagrams to identify the requirements and specifications of the client. We used draw.io and Mock flow to create models like Use case diagram, Class diagram, ER diagram, Activity diagrams and Sequence Diagrams. [5]

To track the progression of the system development, we are using a Kanban board in Trello web application. In here everyone has the access to update and record the progress of the development and upload the issues and question they have faces during the development. [6]

For the implementation, we use visual studio code IDE. Also, we use MongoDB compass to manage the database of the system.

As the database we use MongoDB to manage our data in the system. For run the implemented system we use Apache server. The connection between the user and the server can be implemented by the Apache server.

1.6.4 Testing Methods

After completing the development, we must do the process of verification and validation of the system. To test the functionality of the system based on the required specification, we are going to execute black box testing on the system. In here we are using the following techniques as,

- Boundary Value Analysis
- Equivalence Partitioning
- Decision Tables
- Domain Test
- State Models [7]

For examine the program structure and the business login of the system we are using white box testing method. In here we are,

- Verifying the execution of all the independent paths within a module.
- Verifying all the logical decisions of the system.
- Discovering any types of errors like logical errors, design errors, typographical and syntax errors of the system.

1.6.5 Integration Methods

As the integration method for this system, we are using the Rest API method. It uses HTTP requests to access and manipulate data and typically return data in JSON or XML format. REST API provides a standardized way for different applications to communicate other over the web. [8]

1.7 The Structure of the Report

The rest of the report includes four other main chapters (Chapter 2, Chapter 3, Chapter 4, Chapter 5) which include furthermore details. Also, references and appendixes are mentioned at the end of the report.

- ✓ Chapter 2

This chapter includes requirements analysis, stakeholder analysis and requirements modeling.

User stories and use case diagrams are used to explain the project's functional requirements. There are also non-functional requirements.

- ✓ Chapter 3

This chapter contains designs and development details. All the diagrams of components, processes, workflow, databases, and development aspects are included. The design aspect of the project is illustrated here with diagrams. The software design is demonstrated using UML diagrams such as class diagrams, activity diagrams, and high-level architecture diagrams. The ER diagrams show the software's database design. User interface designs are used as diagrams of components to clarify the UI design.

- ✓ Chapter 4

Chapter 4 includes the test cases and results. The software's test strategy is described here. Test cases are used to validate that all major functions work correctly.

- ✓ Chapter 5

This chapter provides an idea about the scope of the project. The limitations and shortcomings of the proposed methodology and solution are also explained here. Evaluation and conclusions are discussed in this section.

At the end of the report all the references are mentioned and in the post-body section, appendices are mentioned to identify how the appendices are referred to in the document.

1.8 GitHub Repository Link

https://github.com/SLIITITP/y2_s2_wd_it_01-itp_wd_b05_g03

The above repo link belongs to the GitHub repository which created inside the SLIIT/ITP organization Classroom. Visibility of that repository is set to Private by the Classroom owners and team members cannot change the visibility to public. Because of that there is a doubt whether the examiner who marks this report would be able to see that repository.

As a solution we created a separate personal GitHub repo and uploaded our codes and other documents. Here is the link for that repository.

https://github.com/IT21247804/Nawamaga_Institute_management_System

Chapter 2. Requirements

2.1 Stakeholder Analysis

Members of the different functions in Nawamaga Institute are the main stakeholders of the system. This includes Student Management, Staff Management, Notice and Timetable Management, Financial Management, Assignment and Quiz Management and Support Services.

In addition, Student, Admin and Staff of the institute have shown interest of the newly developing system.

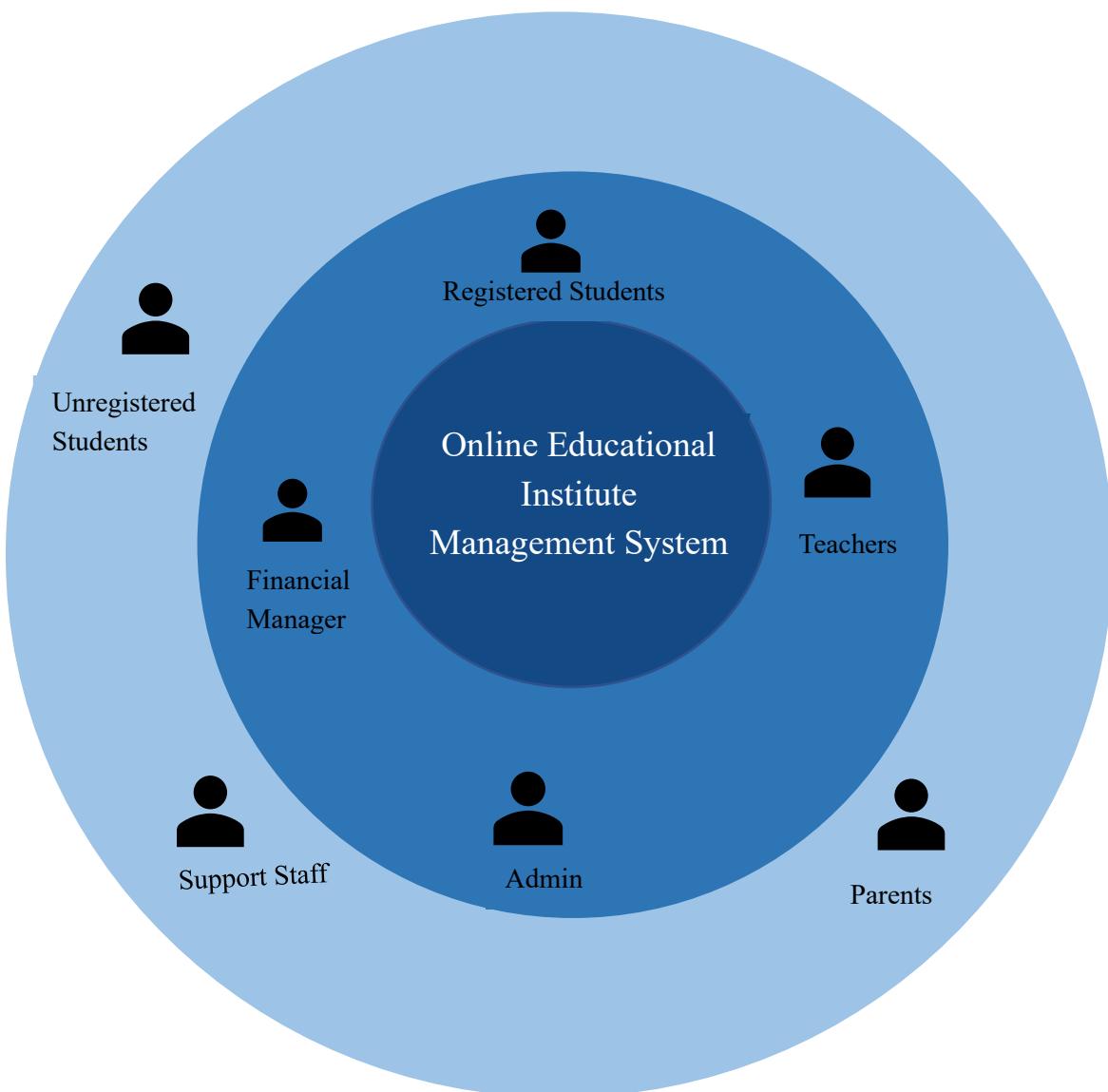


Figure 2:0:1 Onion Diagram

2.2 Requirements Analysis

Our group has decided to develop a fully functional web-based application for Nawamaga Educational Institute. All the functional and non-functional requirements were gathered after analyzing the existing manual system, and the expected outcome was to create an effective and dependable online Educational Institute.

2.2.1 Functional Requirements

Admin

- Login
- Update Student profile
- Verify Students
- Send verification emails.
- Add new Staff Members and Teachers
- Delete Staff Members
- Edit Staff Member profiles.
- Pay Salary for staff members.

- Manage Student Payments
- Add timetable.
- Delete timetable.
- Update timetable
- View timetable
- Calculate recent questions from students.
- Get a list of all tickets & Questions.
- Tracking Student Question readability
- Get Reports

Student

- Login
- Enroll Module
- Download notes.
- Download recordings.
- View Timetable
- Submit assignments.
- Edit /Delete Submissions
- Enroll for the quizzes.
- Answer the quizzes.
- Submit answer paper.
- View the marks.
- Join the classes.
- Raise Ticket
- Delete and Edit/Update the ticket.
- Answer the Q&A.
- Pay Class Fees
- Request Refund

Teacher

- Login
- View Student Details
- Upload quizzes
- Update /Delete quiz.
- Recheck the Quizzes
- Download the Results sheet.
- Generate mark sheets.
- Searching Module Content.
- Add module.
- Upload or Delete Recordings
- Upload or Delete Notes
- Upload assignment.

2.2.2. Non-Functional requirements

- **Security**
Sensitive data about the business will be stored in the database. So, we needed to consider the level of security.
- **Ease of use**
User interface and functions should be convenient and easy to identify.
- **Availability**
The probability of the system crashing should be minimal. Therefore, users can use this application at any time (24/7).
- **Performance**
This application should navigate to the correct page without any defects because patient queues must not be delayed.
- **Reliability**
The application should be implemented using reliable and tested functions with minimal unforeseen errors.

2.2.3 Technical requirements

- MongoDB
- Express.js
- ReactJS
- NodeJS

2.3 Requirements Modeling

2.3.1 User Stories

- As a student, I want to register to the system so that I can be a registered student.
- As an Admin I want to login to the system so that I can interact with the system.
- As a student, I want to give my feedback so that I can share my experience about the Institute with admin.
- As an Admin I want to see Feedback from students so that, I can improve the services of the Institute.
- As a Student I want to make an appointment with teachers so that, I can take advice.
- As an admin I want to see parent appointments so that I can give them instructions.
- As a student, I want to access module management pages so that I can refer to and download older lecture recordings to develop my knowledge.
- As an Admin I want to upload lecture recordings so that I can provide previous lecture materials to students.
- As an admin I want to generate a report so that I can get an idea about what modules were uploaded previously.
- As a teacher, I want to add new assignments for the students. So, then I can assign marks to the students when they submit the completed assignment.
- As a Student I can submit completed assignments to the system using the given link So that student can understand their levels.
- As a Tutor I need to schedule a questionnaire. I should be able to edit any questions on the quiz module page. After the quiz ends, I want a report like Attendance of Students, Questions skipped by students, and The result sheet as a document.
- As an admin I want to make a suitable timetable for classes So that Students and Teachers can manage their schedules
- As a Student I want to view my class timetable So that I can join classes without any issues
- As an admin I want to send notifications about class changes and published notices about class fees so that they can always get updated information.
- As an admin I want to add teachers and staff members So that users can manage their own functions
- As an admin I pay teachers' and staff members' salaries on time. So that they can easily do their financial work.
- As a student I want to pay my class fees So that I can easily pay class fees online.

2.3.2 Use Case Diagram

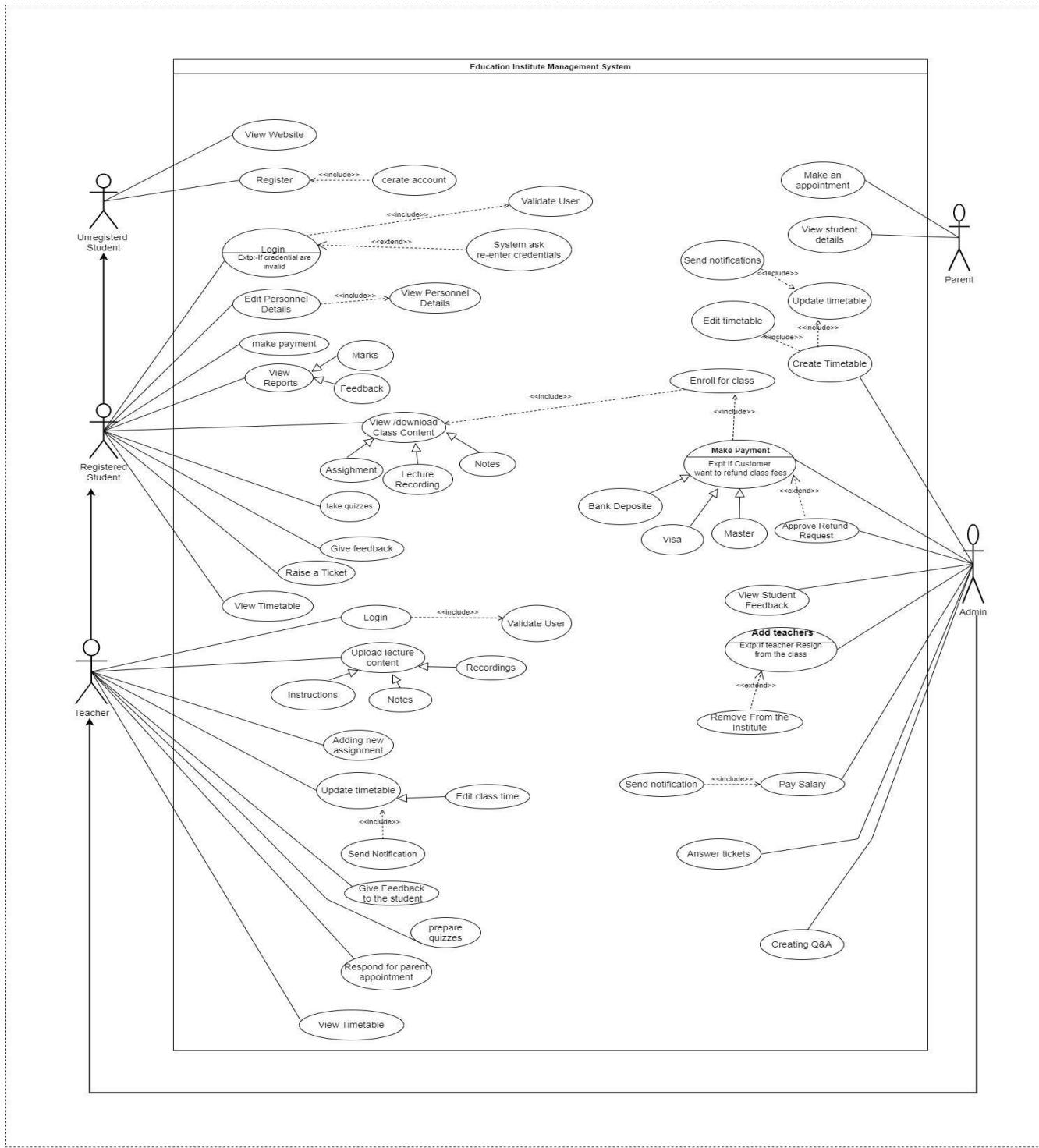


Figure 2:0:1 Usecase Diagram

2.3.2 Use Case Scenario

2.3.2.1 Use case Scenario for Student Management IT21247804

Use Case Name	Student Register to the system
Actor	Student
Goal	Be a registered student of the system.
Overview	Student gives details and system store student details.
Pre-conditions	Student views the Institute website using web browser
Post-conditions	Student details are stored in the database. System redirects to login page
Basic path	1.Student views the Login page. 2.Student press “Register” button. 3.System asks to enter a password. 4.Student enter the password. 5.Student press finish button. 6.System store details. 7.System shows “Success message”.
Alternative Path	7.a) If error occurred System shows “error” message.
NFRs and TRs	NFR1: Security NFR2: Availability NFR3: Response time NF4: Language – Support TR1: Basic skills in computer or mobile. TR2: Stable Internet connection. TR3: Web-browser

Use Case Name	Student makes an appointment
Actor	Student
Goal	Receive a time to meet the tutor
Overview	Make an appointment to meet the tutor
Pre-conditions	Student must login to the system
Post-conditions	
Basic path	<p>1.Student login to the system using respective credentials.</p> <p>2.System redirects to the home page.</p> <p>3.Parent click on “Make an appointment” Button.</p> <p>4. System shows the appointment form.</p> <p>5.Paent fill the Appointment form.</p> <p>6.Parent click on the finish button.</p> <p>7.Sytem store details.</p> <p>8.System shows success message.</p>
Alternative Path	<p>1.a) If credentials are invalid System shows to “RE -Enter” message.</p> <p>8.a) If there is an error system shows “Error” message.</p>
NFRs and TRs	<p>NFR1: Availability</p> <p>NFR2: Response time.</p> <p>NFR3: Scalability</p> <p>NF4: Language - Support</p> <p>TR1: Basic skills in computer or mobile.</p> <p>TR2: Stable Internet connection.</p> <p>TR3: Web – browser.</p>

Use Case Name	Student gives feedback.
Actor	Student
Goal	to Give feedback on the classes, teachers, and the system.
Overview	Student logs in to the system and fills the feedback form.
Pre-conditions	Student must log in to the system.
Post-conditions	Entered details should be showed on the website.
Basic path	<p>1.Student validated to the system.</p> <p>2.System redirects to home page.</p> <p>3.Student press “feedback button”</p> <p>4.System redirects to the feedback form.</p> <p>5.Student enter details.</p> <p>6.Student press submit button.</p> <p>7.System Store details.</p> <p>8. System display “Success” message.</p>
Alternative Path	1.a) If credentials are invalid System shows to “RE -Enter” message.
NFRs and TRs	<p>NFR1: Availability</p> <p>NFR2: Response time.</p> <p>NFR3: Scalability</p> <p>NF4: Language - Support</p> <p>TR1: Basic skills in computer or mobile.</p> <p>TR2: Stable Internet connection.</p> <p>TR3: Web – browser.</p>

2.3.2.2 Use case Scenario for Support Services Management – IT21206078

Use Case Name	Raise a ticket
Actor	Student
Goal	Student raise tickets adding their general and academic issues.
Overview	Student login to the system to raise a ticket.
Pre-conditions	Web sites should be available. Student should have an account
Post-conditions	Ticket details are stored in the database. System redirects to view ticket
Basic path	<ol style="list-style-type: none"> 1. Website displays login page. 2. Login as a Student. 3. Navigate to support service page. 4. Raise the ticket. 5. Delete the ticket. 6. Update the ticket. 7. Log out.
Alternative Path	<p>2.a Website notifies user about Invalid Login Data.</p> <p>4.a Send Email to the relevant student.</p> <p>5.a student can delete or edit the ticket.</p>
NFRs and TRs	<p>NFR1: Security NFR2: Availability NFR3: Usability NFR4: Maintainability</p> <p>TR1: Basic skills in computer or mobile. TR2: Stable Internet connection. TR3: Web-browser</p>

2.2.2.3 Use case Scenario for Timetable and Notice Management – IT21315664

Use Case Name	Creating/updating Timetables
Actor	Admin (Primary), Teacher/Student (Secondary)
Goal	Admin creates and updates timetables without clashing
Overview	Admin creates timetable if any changes happen edit the timetable and update it. Then send notifications to teachers/students
Pre-conditions	Admin logging into the system.
Post-conditions	Admin creates timetable/
Basic path	<ol style="list-style-type: none"> 1. Admin logging to the system 2. The system navigates to the admin dashboard. 3. Admin selects the timetable and clicks the create button. 4. System navigates to the timetable dashboard. 5. Admin fill in the relevant form for subject, teachers, and classroom. 6. System automatically creates a timetable. 7. System sends notifications to teachers/students when happening changes in timetables.
Alternative Path	<ol style="list-style-type: none"> 1 a. Invalid logging 2 a. Incomplete form filling.
NFRs and TRs	<p>NF1: Time for complete transaction</p> <p>NF2: Language support</p> <p>NF3: Availability</p> <p>NF4: Maintainability</p> <p>NF5: Reliability</p> <p>TR1: MongoDB</p> <p>TR2: Node JS</p> <p>TR3: React JS.</p> <p>TR4: Express</p>

2.2.3.4 Use case Scenario for Class Fee Management– IT21222672

Use Case Name	Pay Salary
Actor	Admin (Primary), Staff Member (Secondary)
Goal	Admin pays salary successfully for staff members.
Overview	First Admin adds staff members successfully for staff members successfully. After admin provide bank account details of staff members and pay suitable salary for everyone.
Pre-conditions	Admin logging into the system.
Post-conditions	Staff Member has received their salary.
Basic path	<p>Admin registered staff members</p> <p>Admin creates staff members' profile.</p> <p>Staff members received notification about creating their profiles.</p> <p>Admin selects the pay salary button.</p> <p>System navigates the salary payment page.</p> <p>Admin enters the account details about staff members.</p> <p>Admin enters pay button.</p> <p>Staff members received transaction pdf via email.</p>
Alternative Path	<p>2a. Staff members want to update details.</p> <p>6a. Enter the wrong account number.</p> <p>8a. Type the wrong email address.</p>
NFRs and TRs	<p>NF1: Time for complete transaction</p> <p>NF2: Language support</p> <p>NF3: Availability</p> <p>NF4: Maintainability</p> <p>NF5: Reliability</p> <p>TR1: MongoDB</p> <p>TR2: Node JS</p> <p>TR3: React JS.</p> <p>TR4: Express</p>

2.2.2.5 Use case Scenario for Quiz Management– IT21328916

Use Case Name	Q & A session	
Actor	Management member	
Goal	Staff members give a respond for them as soon as possible.	
Overview	Staff member login to the system for providing answers	
Preconditions	The web site should be available. Student should have an account	
Postconditions	View questions	
Basic path/ alternative path	1	Login as a Staff member
	2	Navigate to support service page
	3	providing answers
	4	Log out
	2a	Website notifies user about Invalid Login Data
	4a	Send Email to the Management
NFRs and TRs		NF1 Useability
		NF2 Reliability
		NF3 Efficiency
		NF4 Maintainability
		TR1 Web browsers using skills.
		TR2 Language support
		TR3 Time for complete raising tickets

2.2.2.6 Use case Scenario for Module Management– IT21327780

Use Case Name	Student Login to the system
Actor	Student
Goal	Be a registered student of the system.
Overview	Student gives details and system store student details.
Pre-conditions	Student views the Institute website using web browser
Post-conditions	Student details are stored in the database. System redirects to login page
Basic path	1.Student views the Login page. 2.Student press “Register” button. 3.System asks to enter a password. 4.Student enter the password. 5.Student press finish button. 6.System store details. 7.System shows “Success message”.
Alternative Path	7.a) If error occurred System shows “error” message.
NFRs and TRs	NFR1: Security NFR2: Availability NFR3: Response time NF4: Language – Support TR1: Basic skills in computer or mobile. TR2: Stable Internet connection. TR3: Web-browser

Use Case Name	Read & Download lecture contents
Actor	Student
Goal	Read and download lecture recordings and study materials
Overview	The student can watch and download pre-recorded lecture recordings through the module page
Pre-conditions	<ul style="list-style-type: none"> • The student logged in to the system using valid credentials and entered the module page. • Students read and download the study material contents.
Post-conditions	<ul style="list-style-type: none"> • The study materials are read. • The study materials are downloaded.
Basic path	<ol style="list-style-type: none"> 1. The student logged in to the system using login credentials. 2. The student chooses the module page. 3. The system navigates to the module page. 4. The system navigates to the Content page. 5. The student chooses the lecture recording or study materials that he/she wants. 6. The system navigates to the recorded lecture-uploading page. 7. The student downloads the files. 8. Student redirect module page.
Alternative Path	1a. Login credentials invalid.
NFRs and TRs	<p>NF1: Downloading file.</p> <p>NF2: Language Support</p> <p>NF3: Response time</p> <p>TR1: General computer or mobile skills.</p> <p>TR2: Downloading files.</p> <p>TR3: Web browsers using skills.</p> <p>TR4: Searching skills.</p> <p>TR5: Stable internet connection.</p>

Use Case Name	Upload lecture recordings
Actor	Tutor
Goal	Upload lecture recordings and study materials
Overview	The Tutor uploads pre-recorded lecture recordings and study materials to the system through the module page
Pre-conditions	The Tutor logged in to the system using valid credentials and uploaded the study material contents.
Post-conditions	<ul style="list-style-type: none"> • The study materials are uploaded. • The system sends an “uploading process is successful” message.
Basic path	<ol style="list-style-type: none"> 1. The Tutor logged in to the system using login credentials. 2. The Tutor chooses the module page. 3. The system navigates to the module page. 4. The Tutor chooses what he/she wants to upload (lecture recordings/study materials). 5. The Tutor chooses the lecture recording. 6. The system navigates to the recorded lecture-uploading page. 7. The Tutor chooses the correct file that he/she wants to upload. 8. The Tutor gives the content title and description. 9. The Tutor clicks the upload button. 10. The system shows uploading successful messages. 11. Tutor redirect module page.
Alternative Path	<p>1a. Login credentials are not valid.</p> <p>10a. Invalid file type.</p>
NFRs and TRs	<p>NF1: Valid file format NF2: Time to go to upload the file. NF3: Language Support NF4: Maintainability NF5: Response time NF6: Uploading file.</p> <p>TR1: General computer or mobile skills. TR2: Uploading files. TR3: Web browsers using skills. TR4: Identifying the correct file format. TR5: Stable internet connection</p>

2.2.2.7 Use case Scenario for Assignment Management– IT21326868

Use Case Name	Adding new assignments	
Actor	Teacher	
Goal	Teacher adding new assignments to the system.	
Overview	Checking student submissions and add marks.	
Preconditions	Web site should be available. Teacher should have an account	
Post conditions	Generate mark sheets.	
Basic path	1	Website displays login page
	2	Login as a Teacher
	3	Navigate to homepage
	4	Navigate teacher to add assignment page
	5	Add new assignments
	6	Saved added assignments.
	7	Log out
Alternative Paths	2a	If invalid login details, system display an error message
NFRs and TRs		NF1- Data integrity NF-2 Availability NF-3 Security
		TR1- Stable Internet Connection TR2-Uploading Files TR3- Downloading Files

Use Case Name	Read and submit assignments	
Actor	Student	
Goal	Read and complete assignments	
Overview	Student login to the system to submit complete assignments.	
Preconditions	Web site should be available. Teacher should have an account	
Post conditions	Checking submitted assignments and edit if it has errors.	
Basic path	1	Website displays login page
	2	Login as a Student
	3	Navigate to homepage
	4	Navigate student to submit assignment page
	5	Submit complete assignments
	6	Saved submit assignments
	7	Log out
Alternative Paths	2a	If invalid login details, system display an error message
NFRs and TRs	NF1- Data integrity NF-2 Availability NF-3 Security NF4-Maintainability	
	TR1- Stable Internet Connection TR2-Uploading Files TR3- Downloading Files	

2.2.2.8 Use case Scenario for Staff Management– IT21219634

Use Case Name	Add Salary for Staff Member
Actor	Admin
Goal	Calculate salary and distributed it quickly
Overview	Admin give details
Pre-conditions	Admin can enter Salary Details
Post-conditions	Admin details are stored in the database. System redirects to login page
Basic path	<ol style="list-style-type: none">1.Login to the system2.Redirect to the dashboard3.select the add salary button4.Enter the details5.Calculate Salary6.Click ADD button7.Display the All the records
Alternative Path	4.a) All the input fields have validations
NFRs and TRs	NFR1: Security NFR2: Availability NFR3: Assurance TR1: Basic skills in computer or mobile.

Chapter 3. Design And Development

3.1 System Architecture

A software application architecture that organizes applications into three logical and physical computing tiers. Since each tier runs on its own infrastructure, each tier can be developed simultaneously by a separate development team. Later, can be updated or scaled as needed without impacting the other tiers.

- 1). Presentation tier This is the user interface and communication layer of the application. end users interact with the application using the presentation tier. The main purpose of the presentation tier is to display information and collect information from the user.
- 2). Application tier This is also known as the logical tier or the middle tier. This can also be known as the heart of the application. This layer can add delete and modify data in the data tier. Information collected from the presentation tier is processed against other information in the data tier using business logic.
- 3). Data tier This is where the information process by the application is stored and managed. This can be a rational database management system or a no SQL database server such as mongo DB. Since Data Tier and presentation tier cannot communicate with each other or the communications go through the application tier. Fast development, Improved scalability, improved reliability and improve security are some of the main advantages in the three-tier architecture.

3.1.1 Three tier architecture in web development

Web server – Presentation Tier Web servers provide the user interface. Usually this is a web page or a website. The content can be static or dynamic and it is usually developed using HTML CSS and JavaScript.

Application server – Middle Tier This is where business logic is used to process user input.

Database server – Data Tier

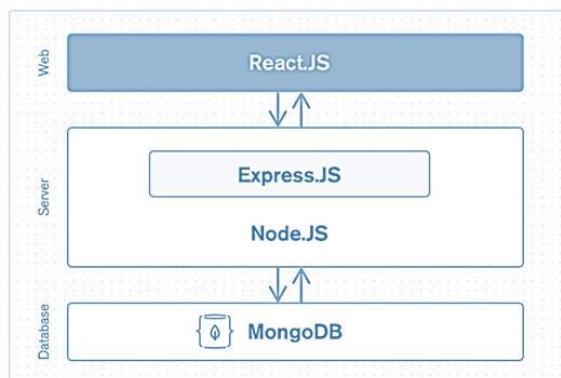


Figure 3:0: IMERN stack Architecture

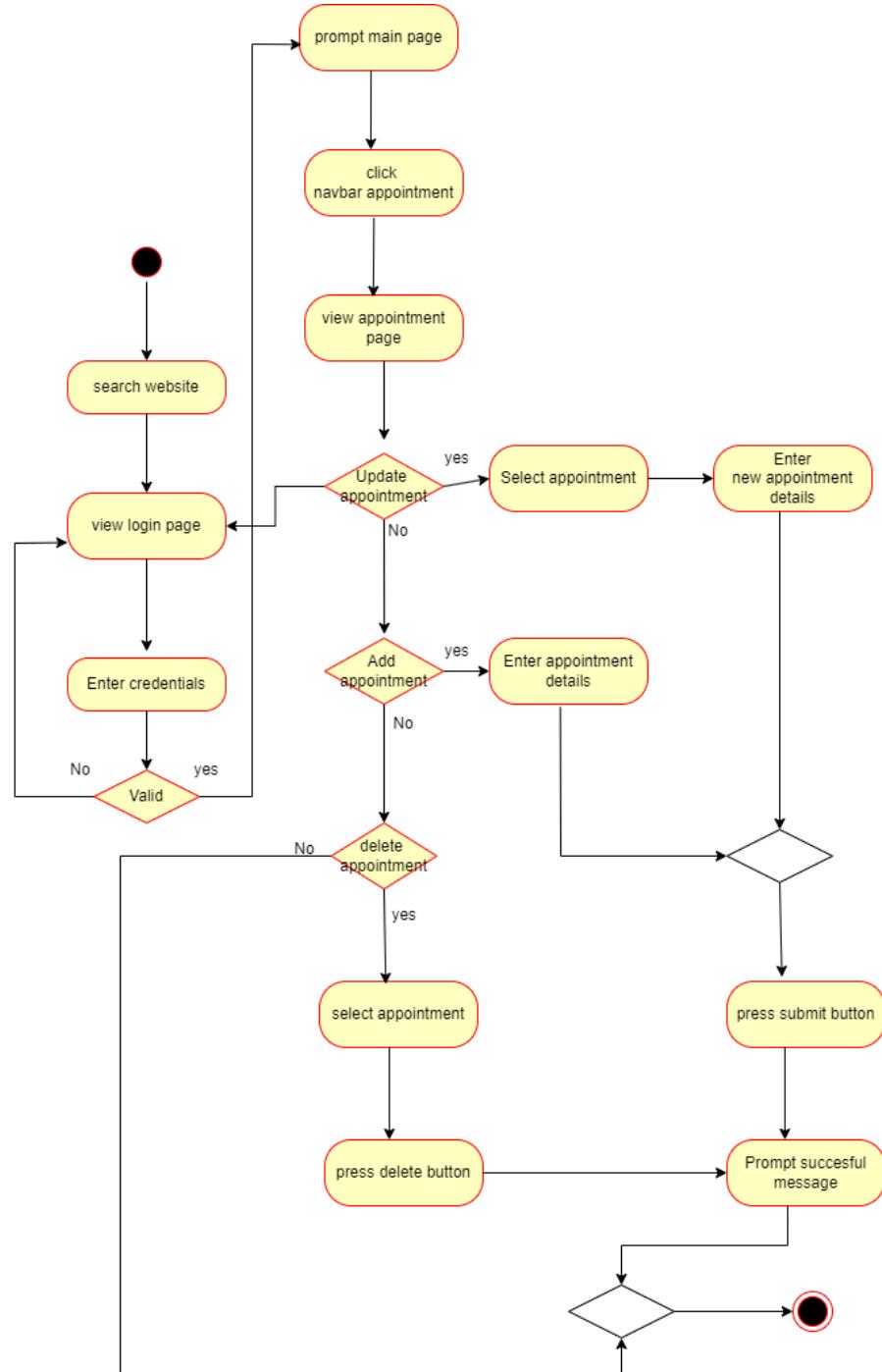
3.2 Activity Diagrams

3.2.1 Activity Diagram for Student Management – IT21247804

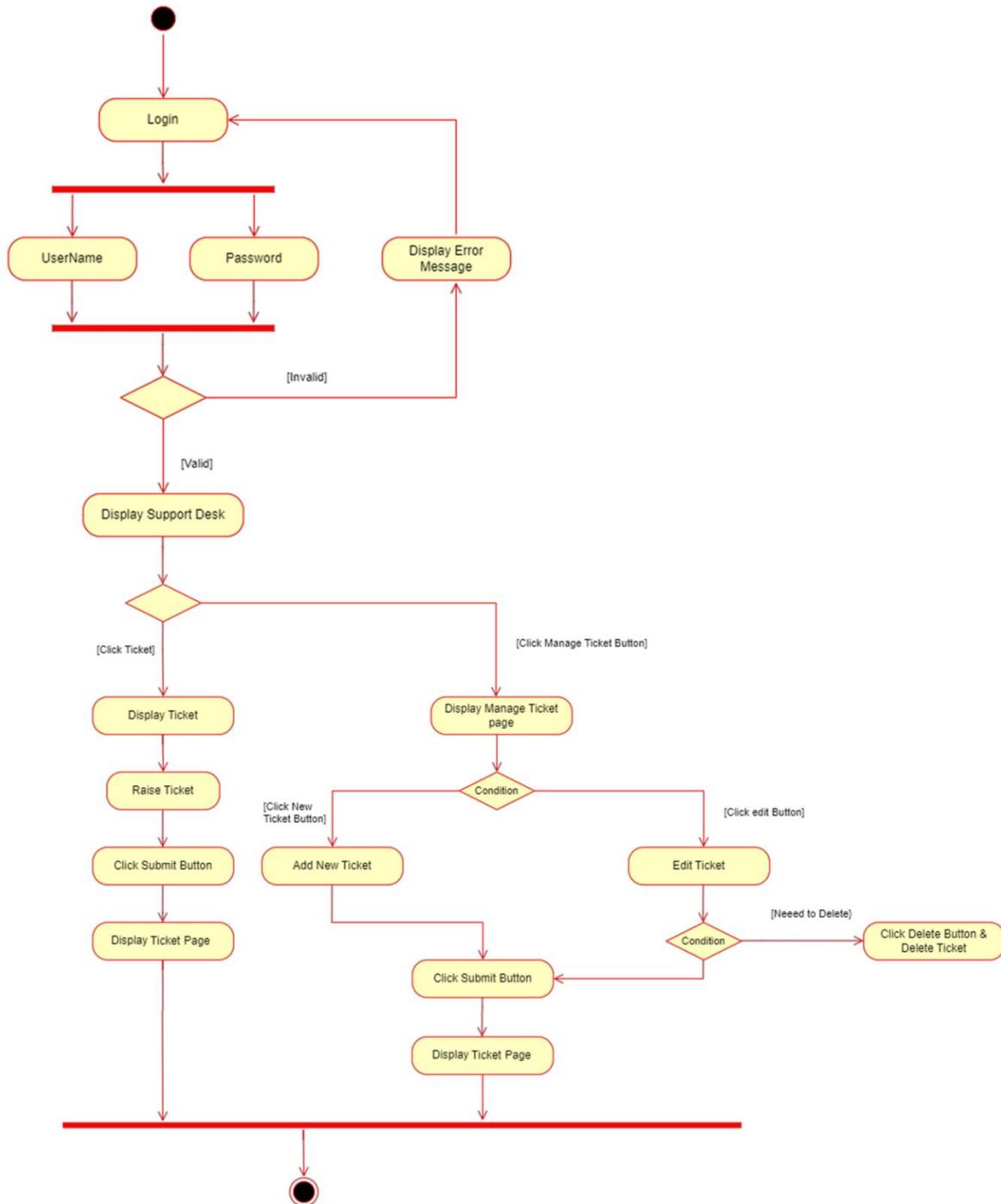
- ✓ Student Feedback



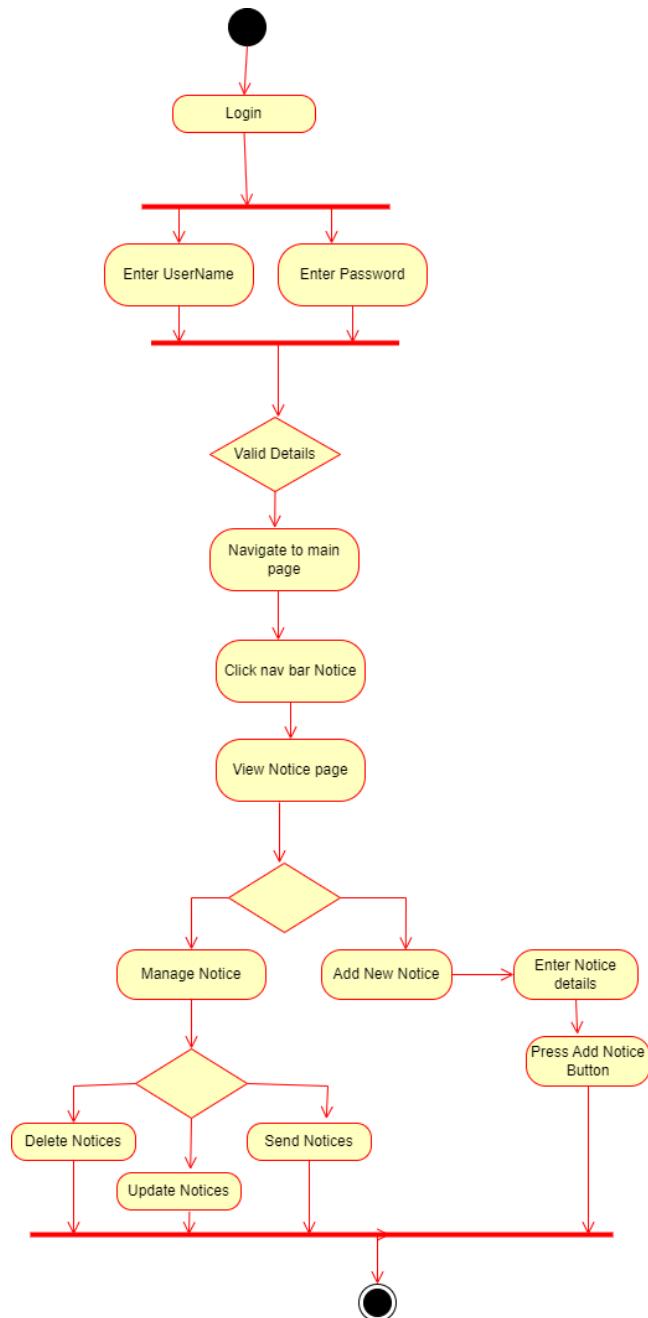
✓ Student Appointment



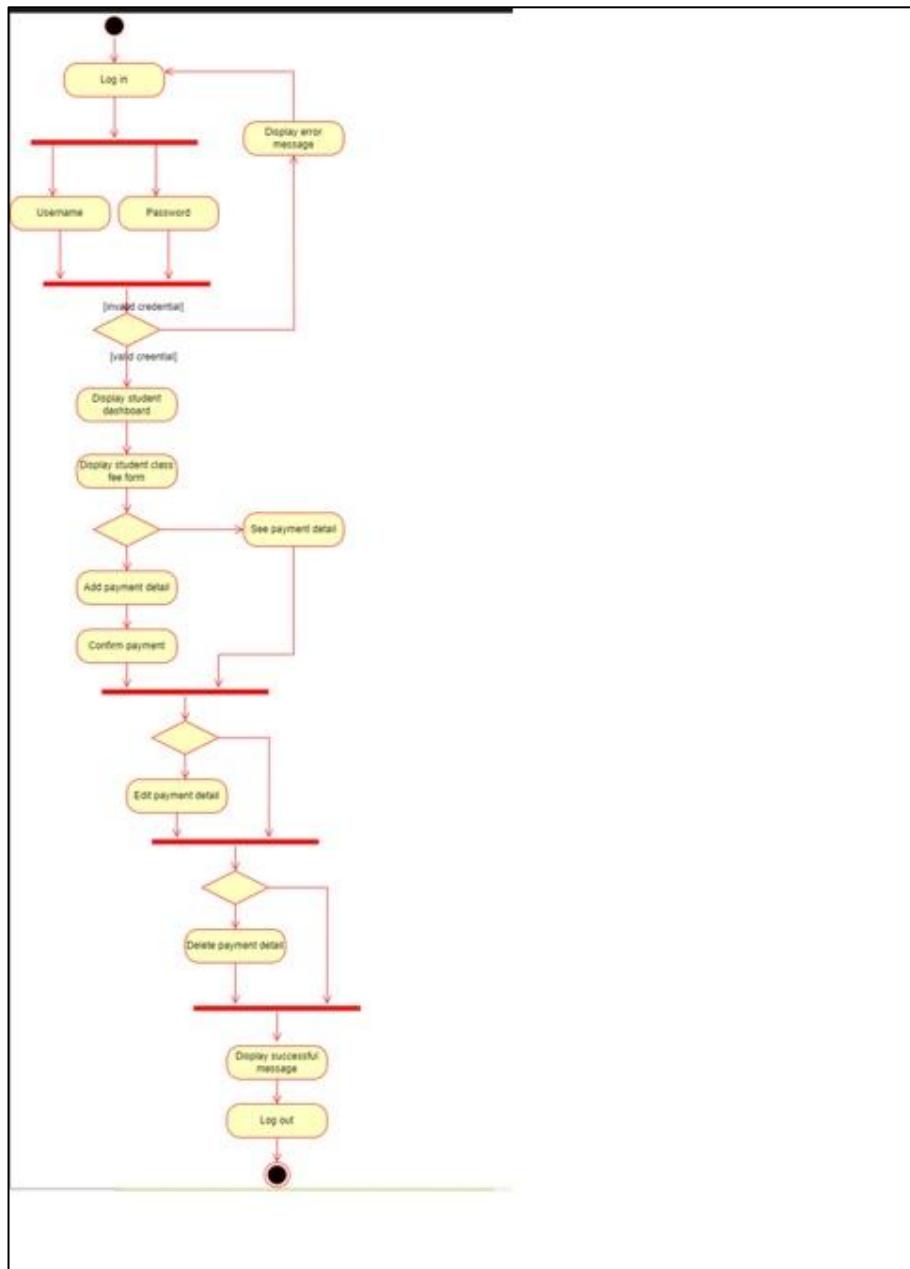
3.2.2 Activity Diagram for Support Services– IT 21206078



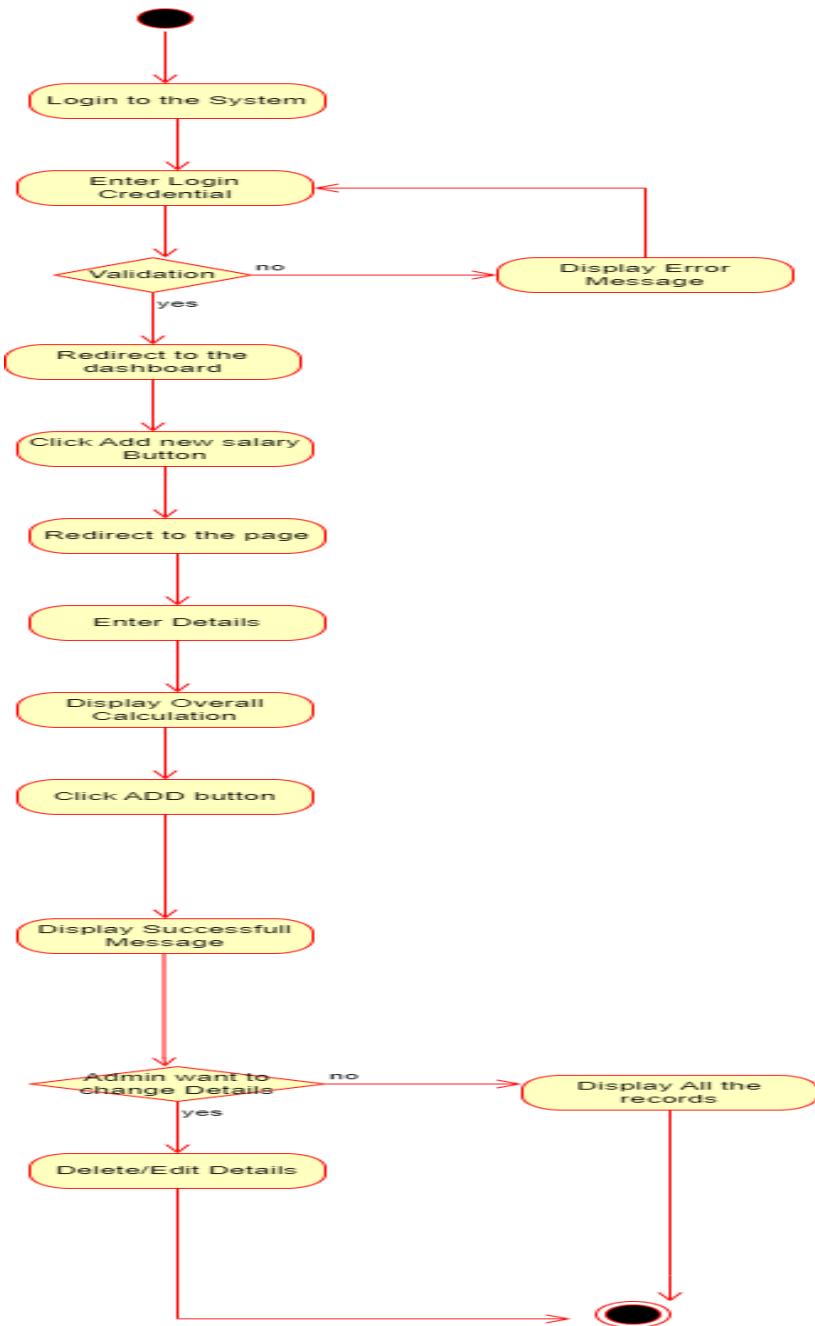
3.2.3 Activity Diagram for Notice and Timetable Management– IT21315664



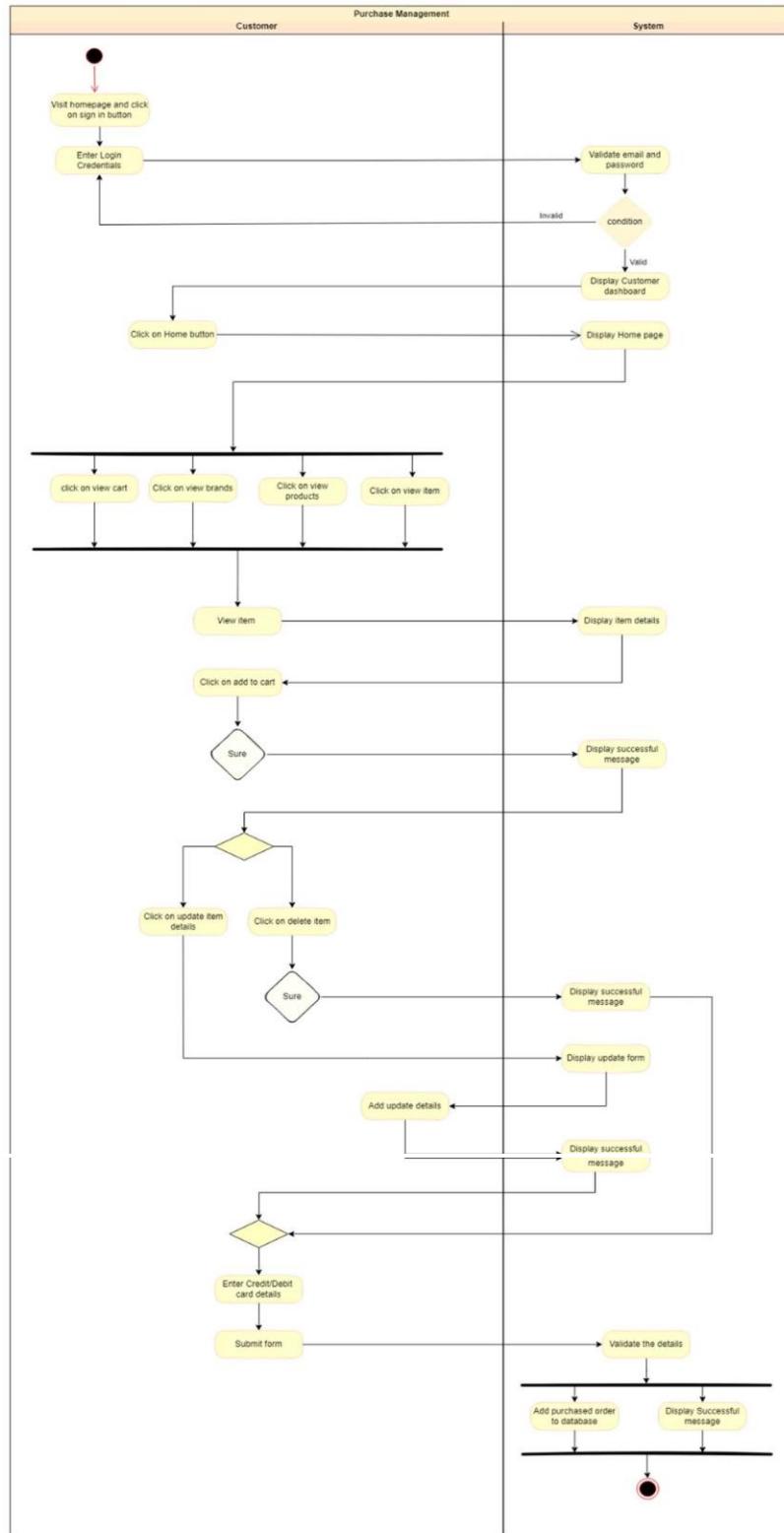
3.2.4 Activity Diagram for Class Fee Management – IT21222672



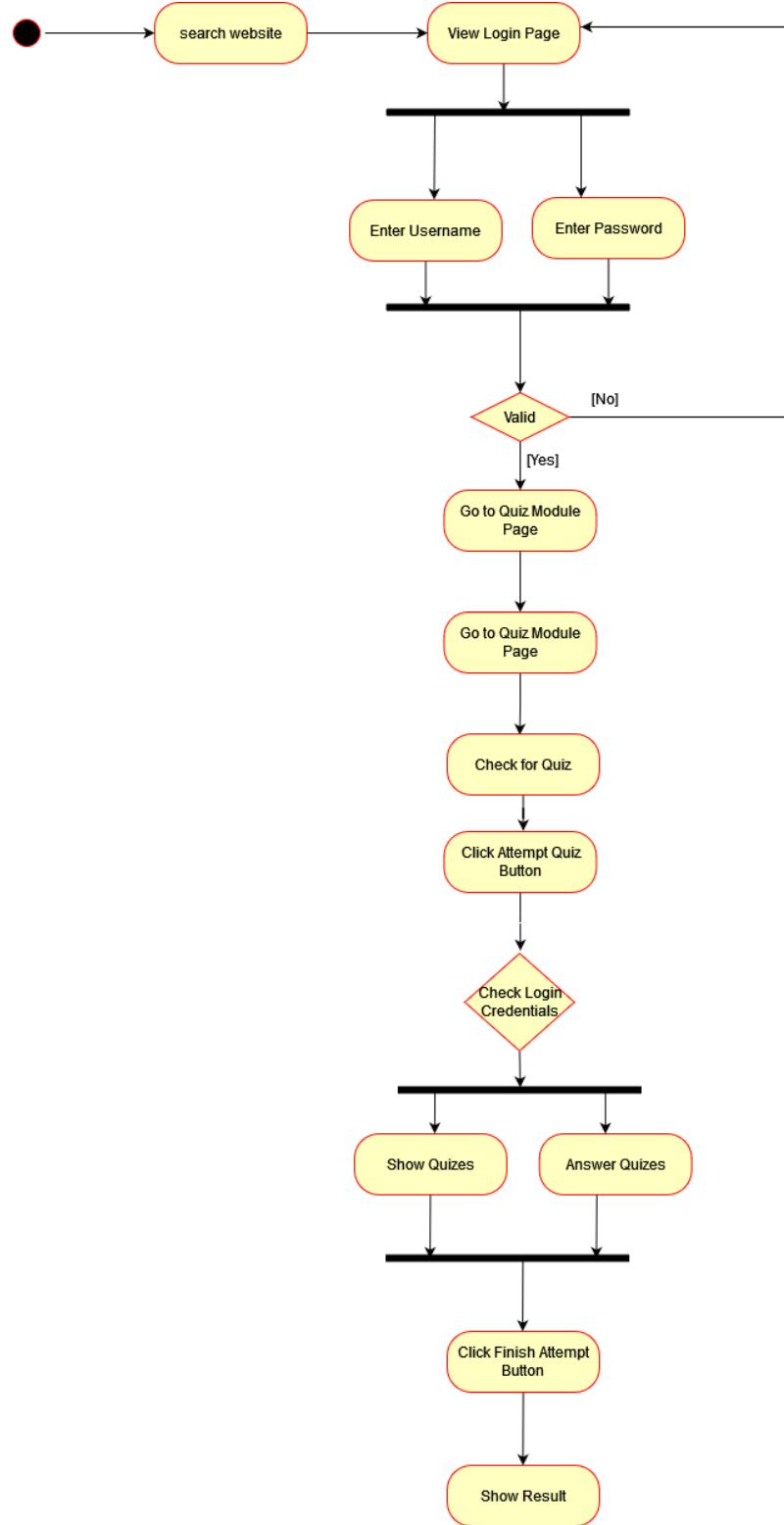
3.2.5 Activity Diagram for Staff Management – IT21219634



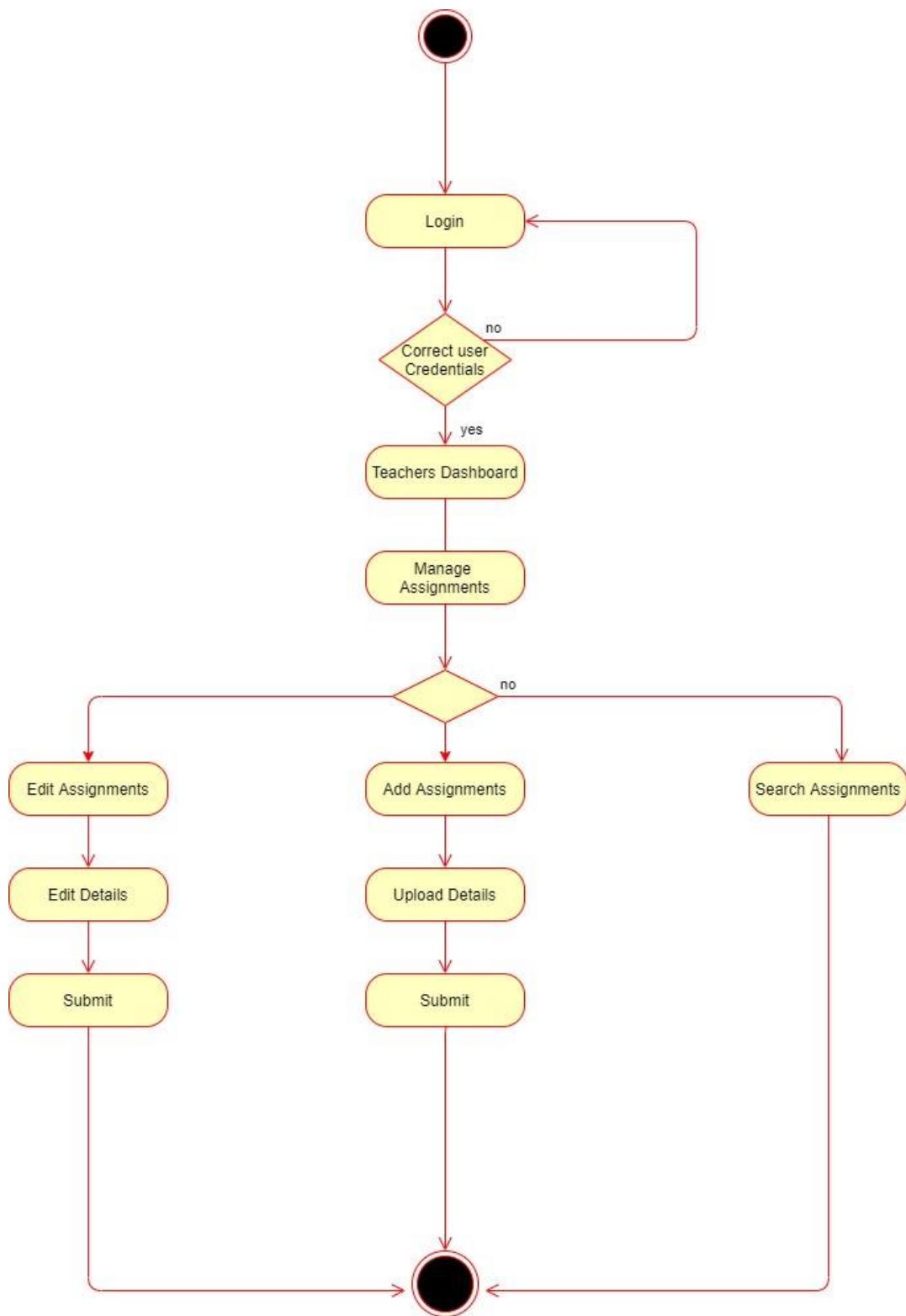
3.2.6 Activity Diagram for Module Management – IT21327780



3.2.7 Activity Diagram for Quiz Management – IT21328916



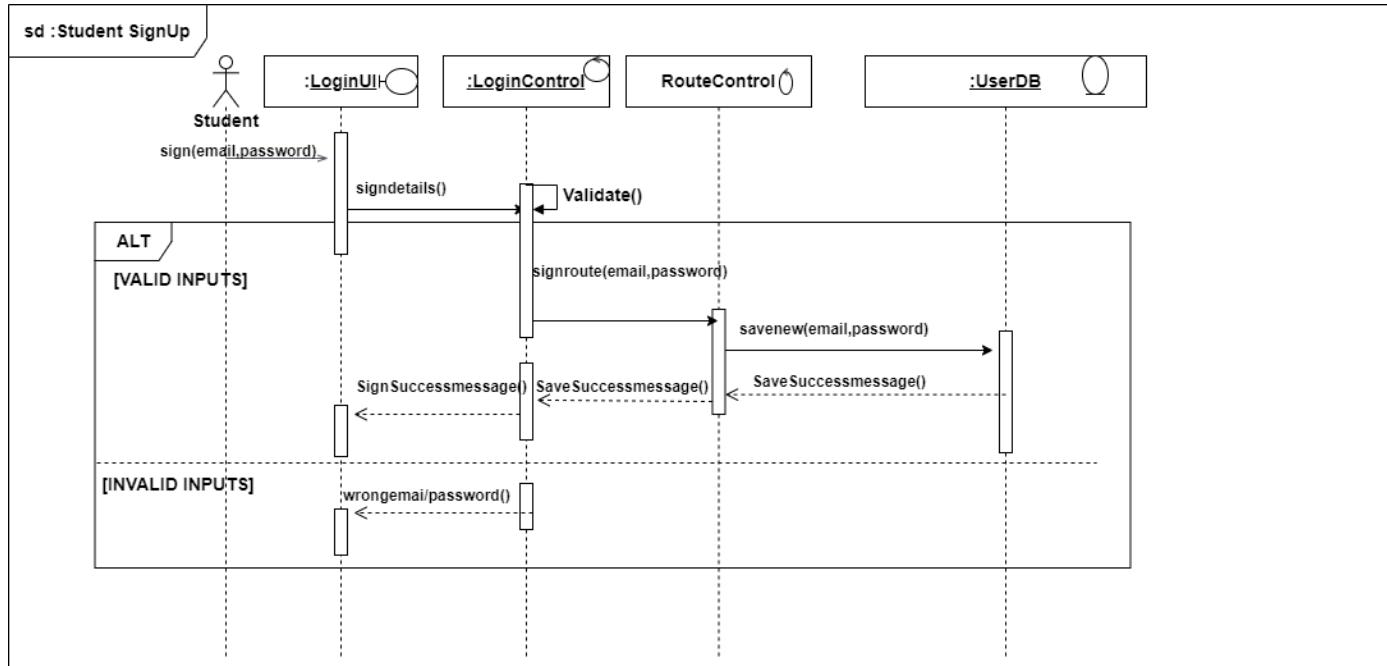
3.2.8 Activity Diagram for Assignment Management – IT21326868



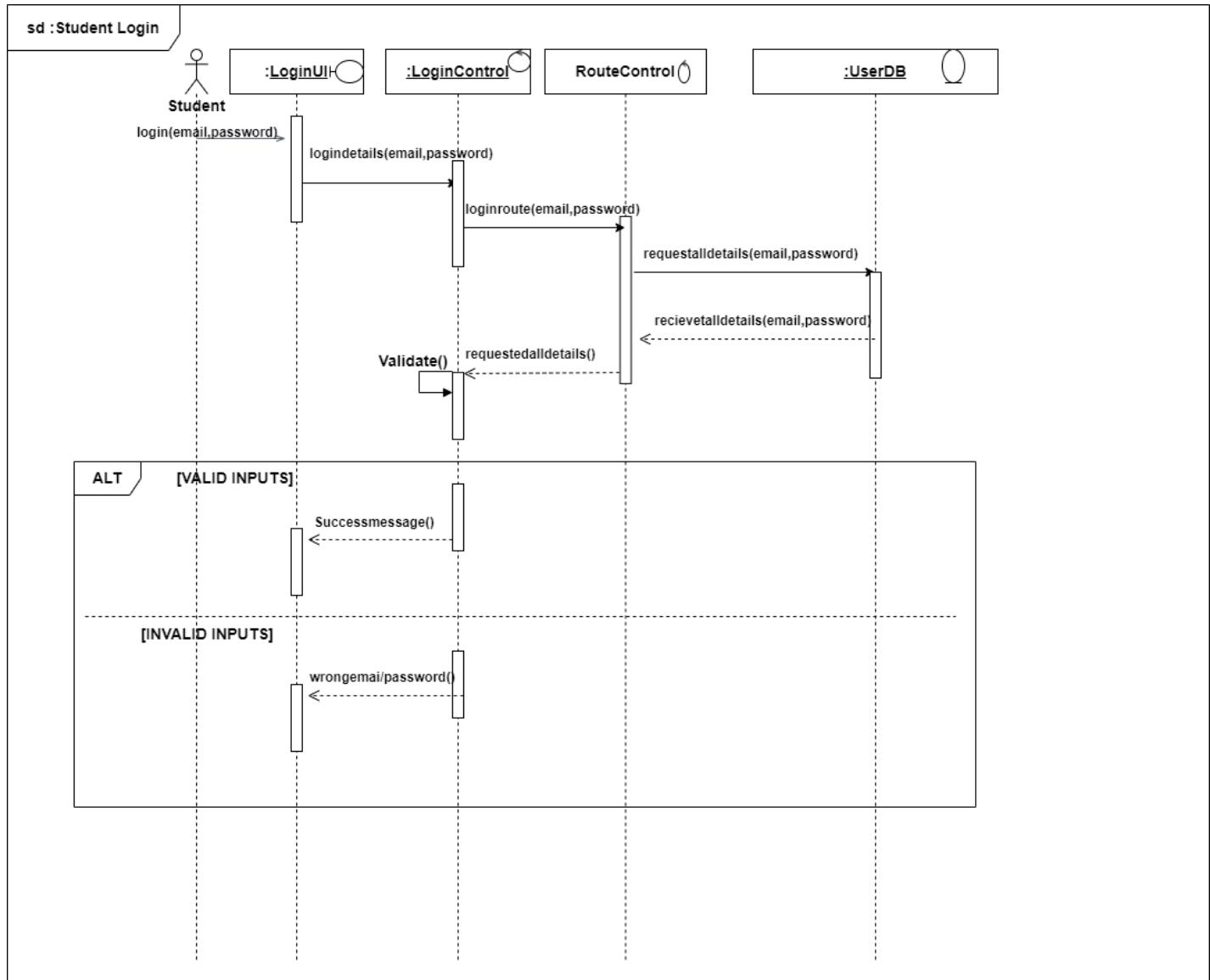
3.3 Sequence Diagrams

3.3.1 Sequence Diagram for Student Management – IT21206078

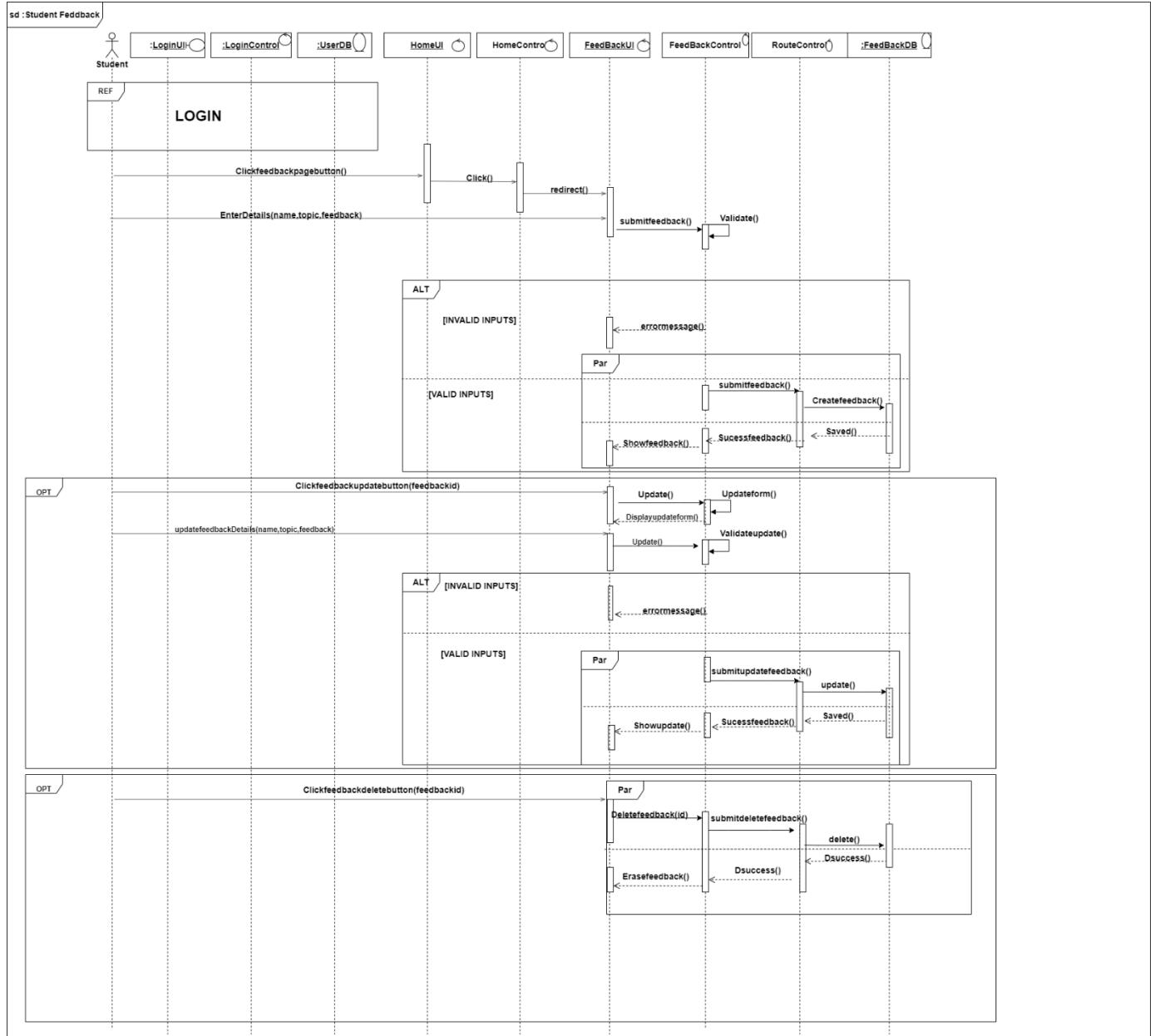
- Signup



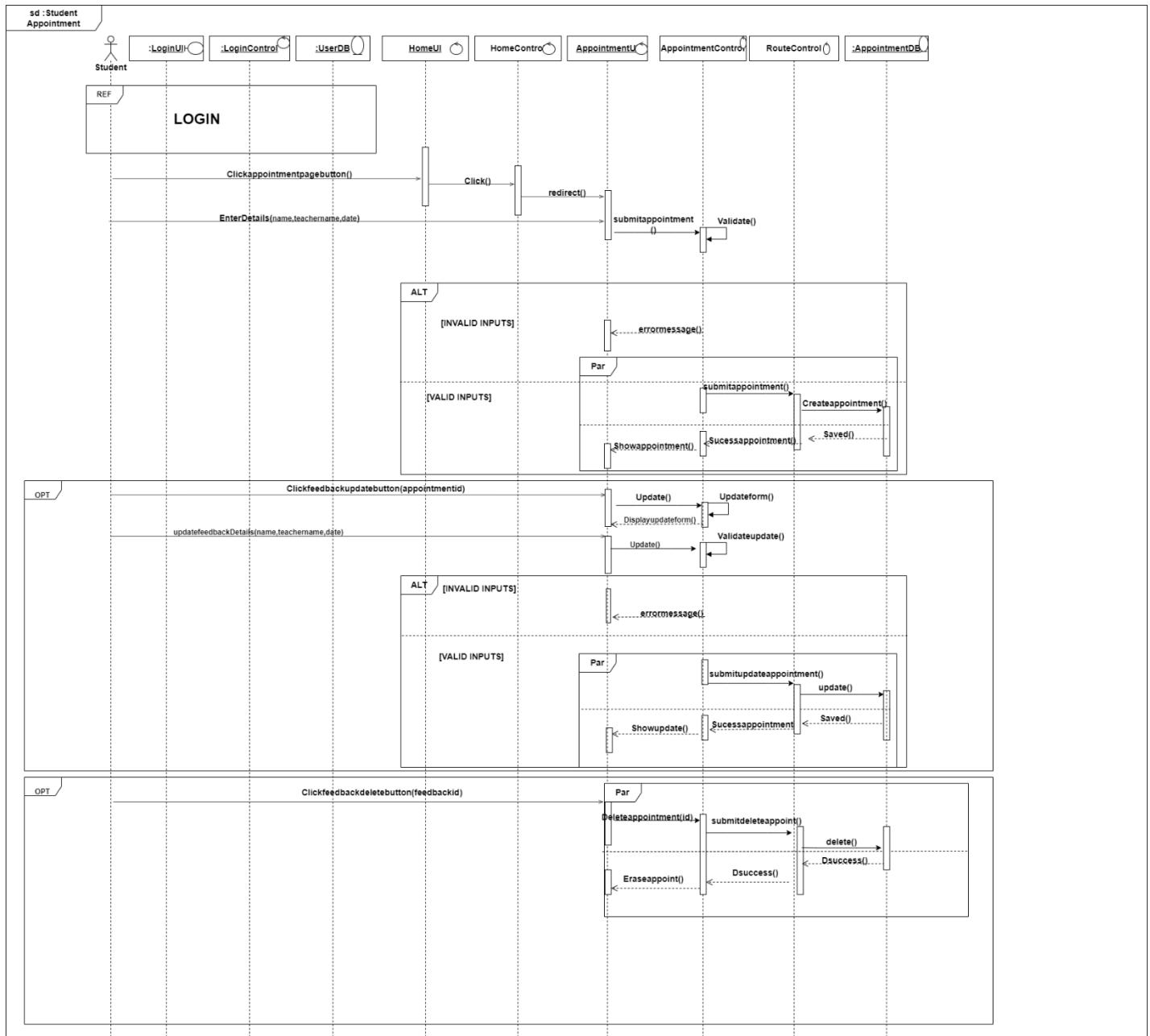
- Student Login



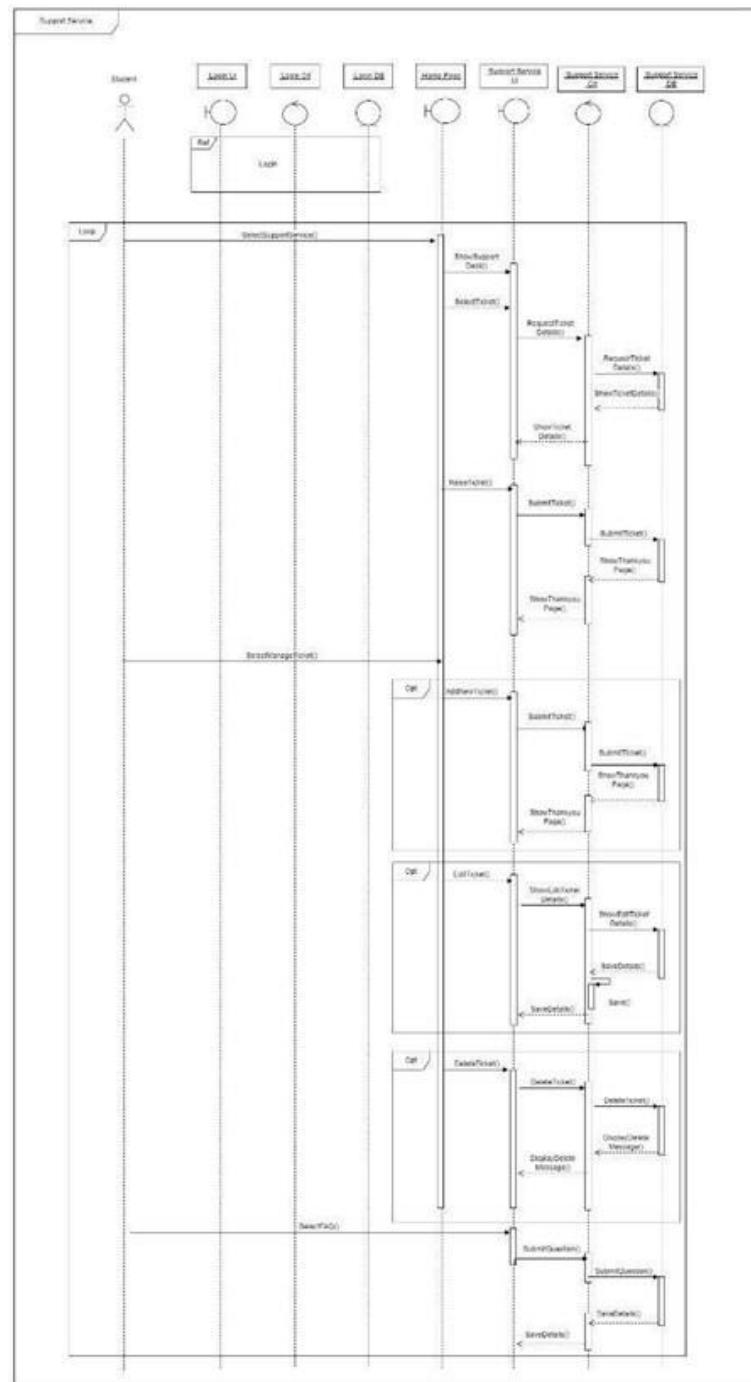
■ Student Feedback



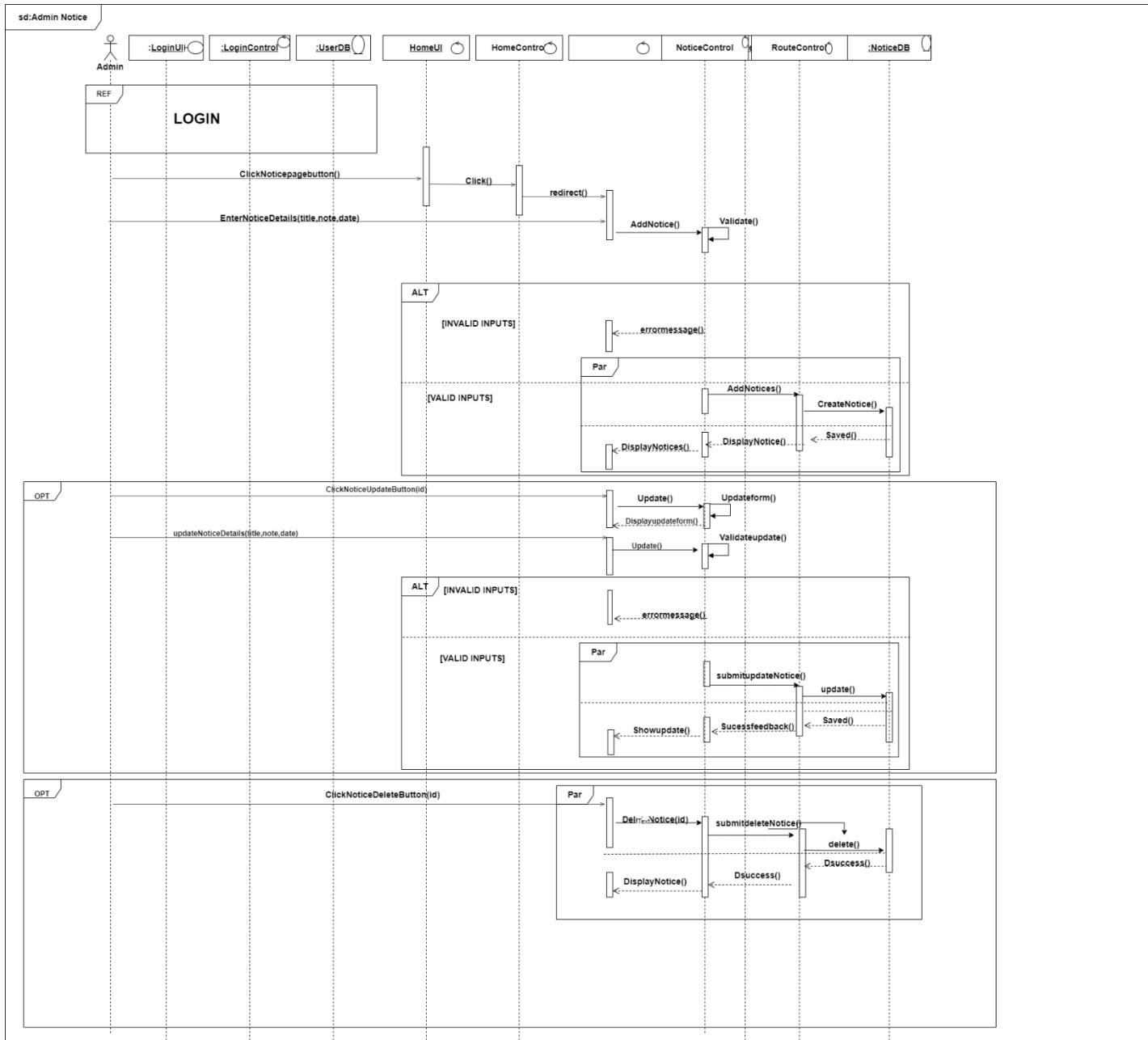
■ Student Appointment



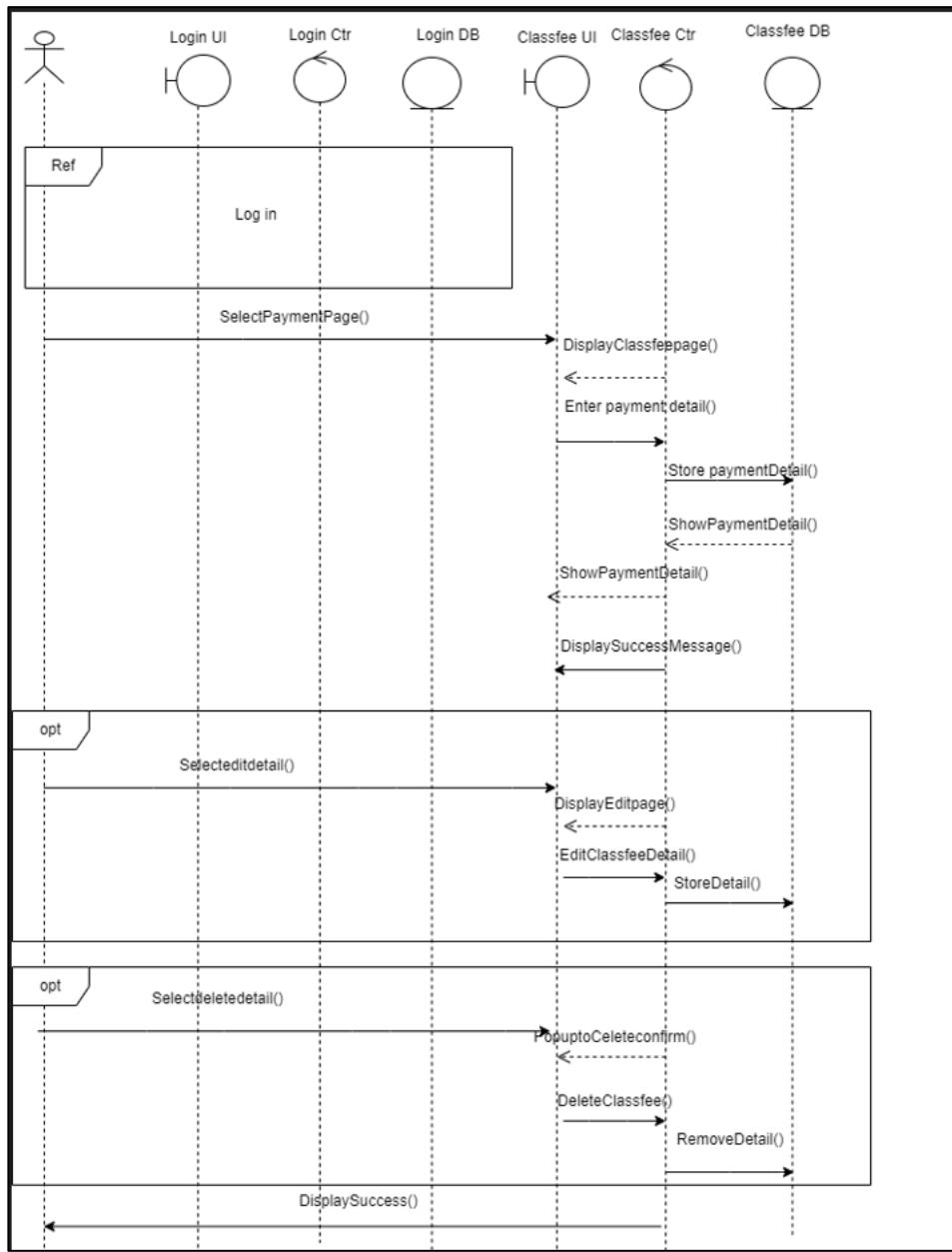
3.3.2 Sequence Diagram for Support Services Management – IT21206078



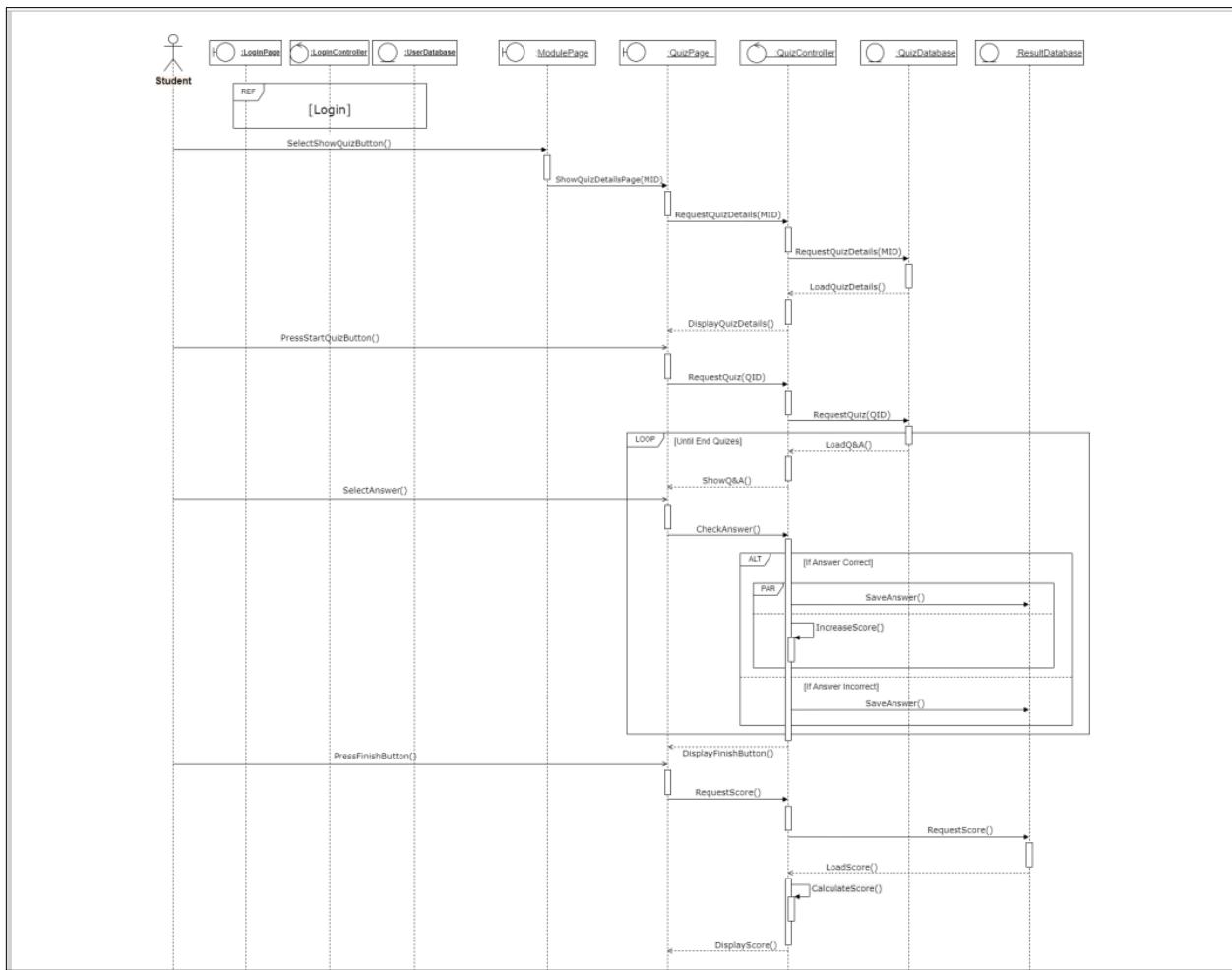
3.3.3 Sequence Diagram for Timetable and Notice Management – IT21315664



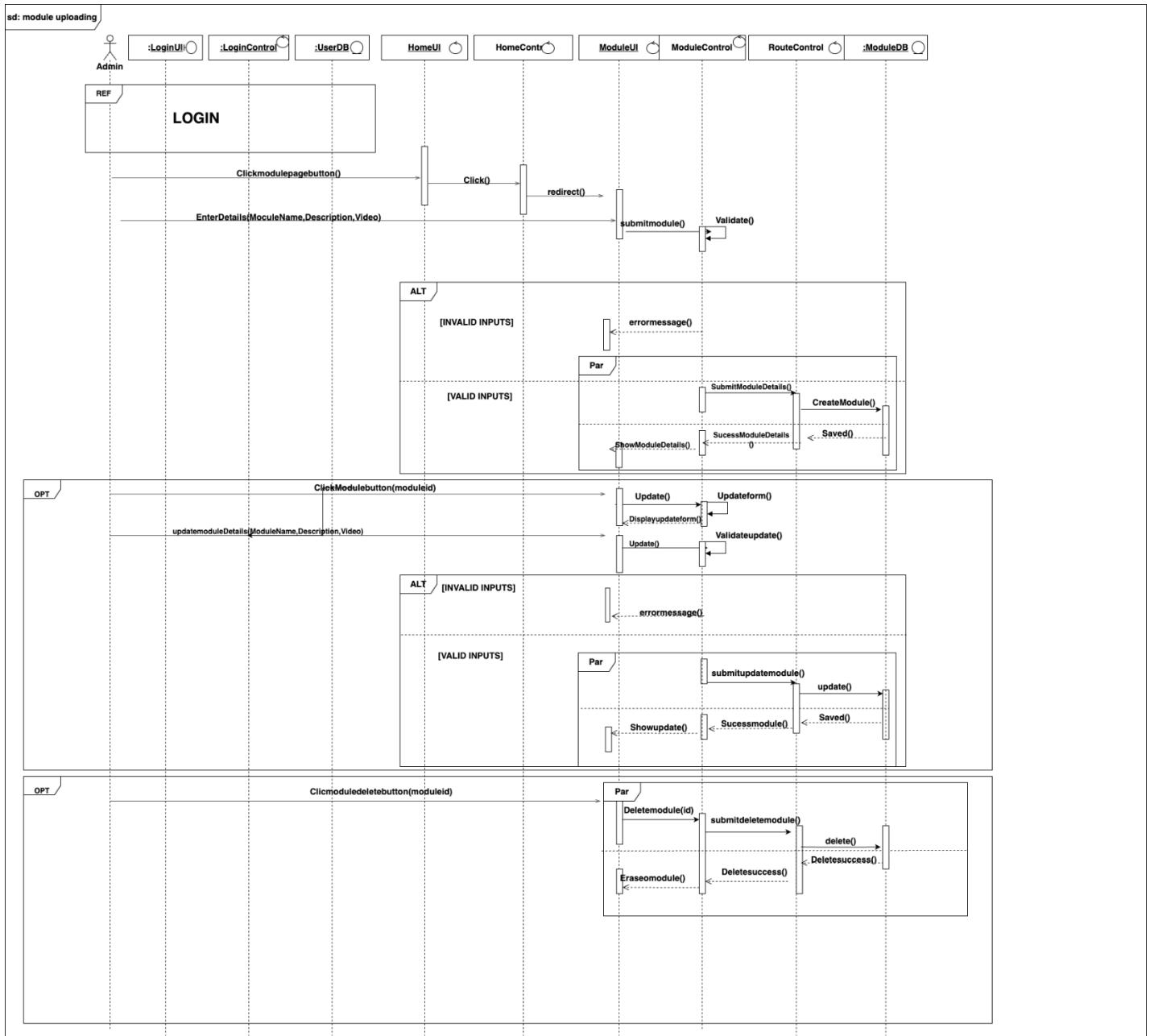
3.3.4 Sequence Diagram for Class fee Management – IT21222672



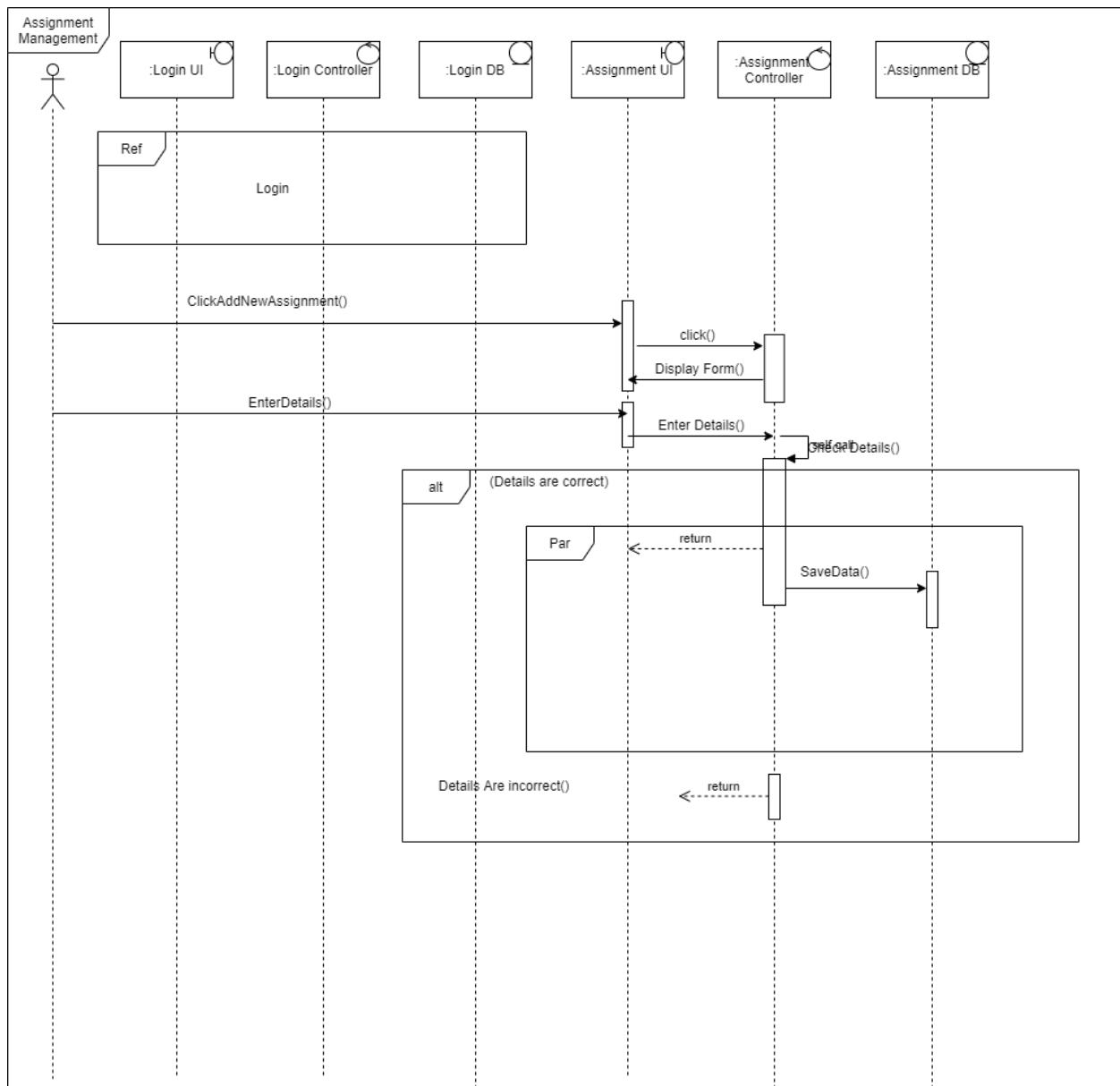
3.3.5 Sequence Diagram for Quiz Management - IT21328916



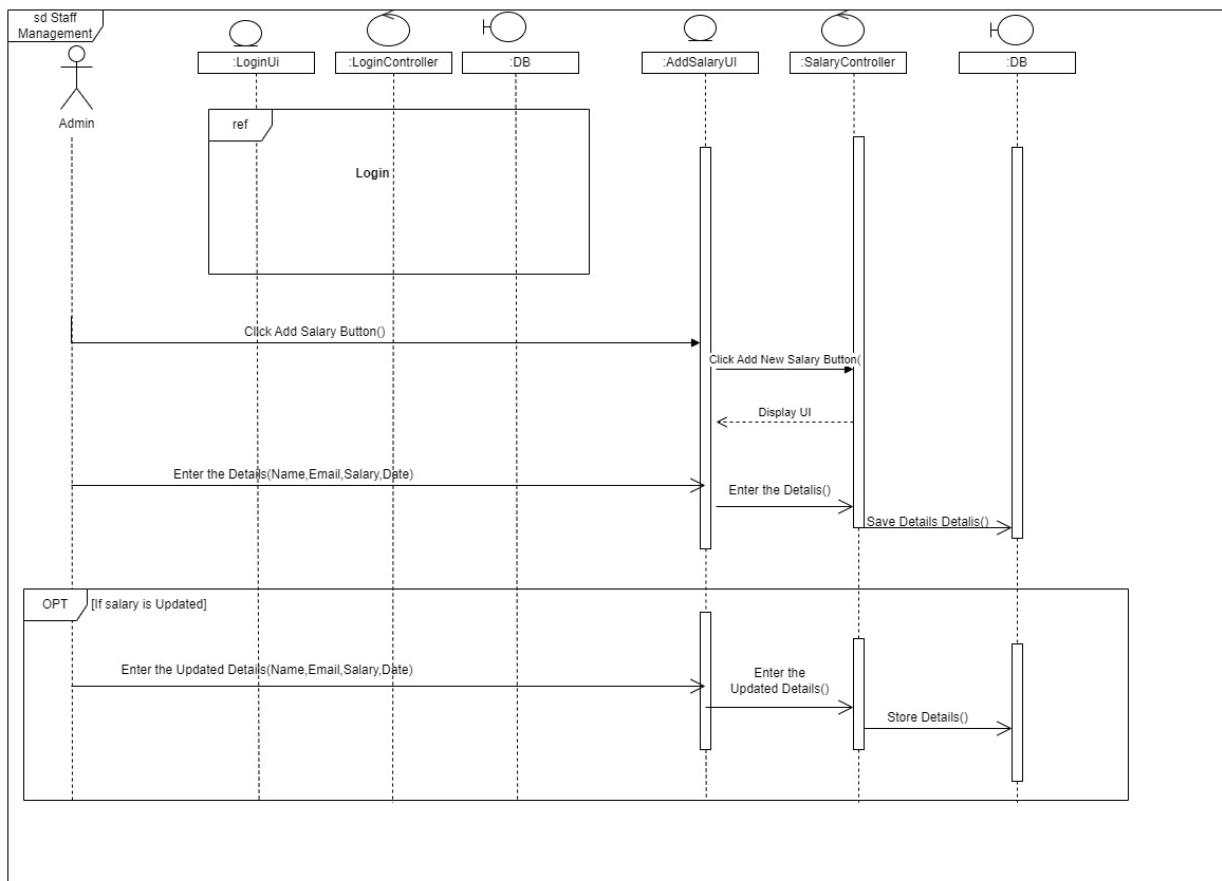
3.3.6 Sequence Diagram for Module Management - IT21327780



3.3.7 Sequence Diagram for Assignment Management – IT21326868



3.3.8 Sequence Diagram for Staff Management – IT21219634



3.4 ER Diagram

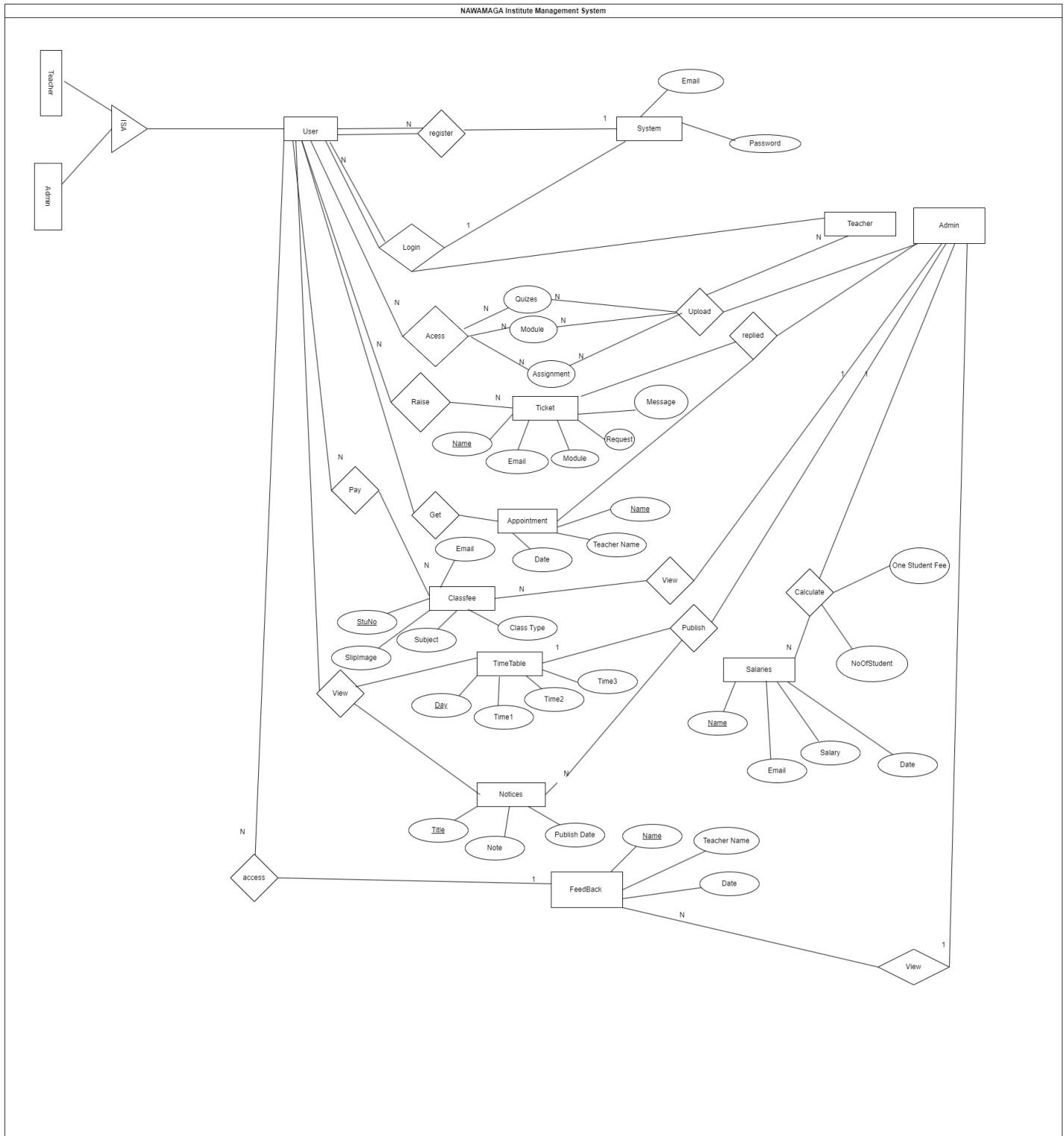


Figure 3.4 ER Diagram

3.5 Class Diagram

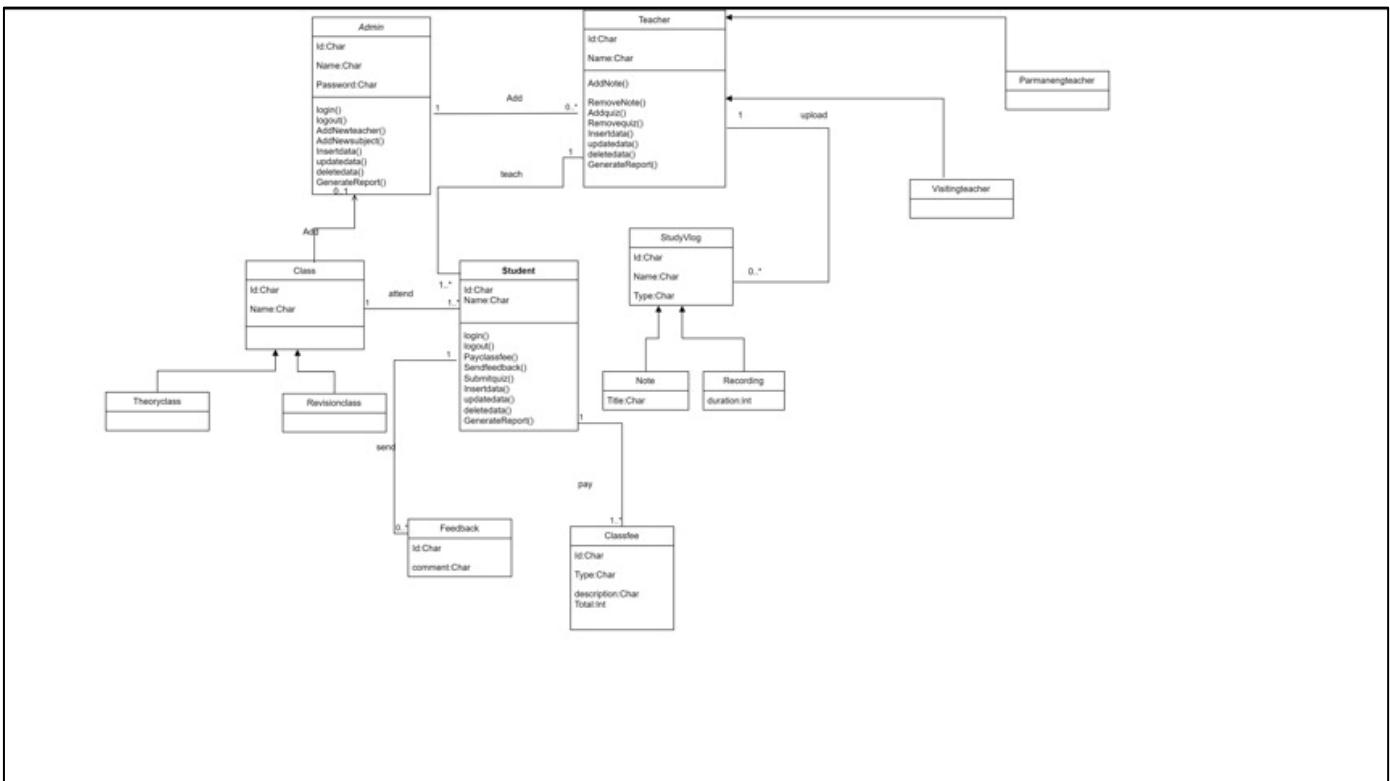


Figure 0:2 Class Diagram

3.6 User Interfaces

3.6.1 Common UI – IT21247804

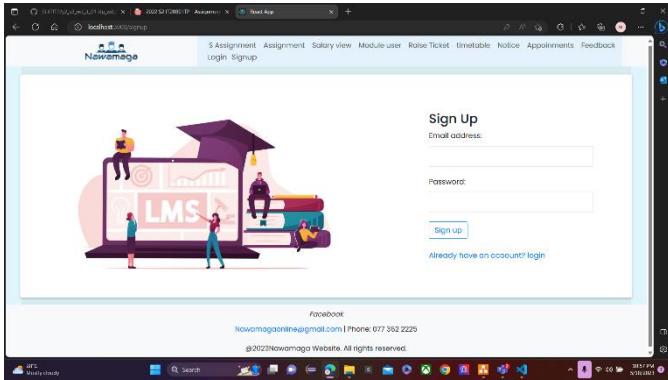


Figure 0:4 Signup

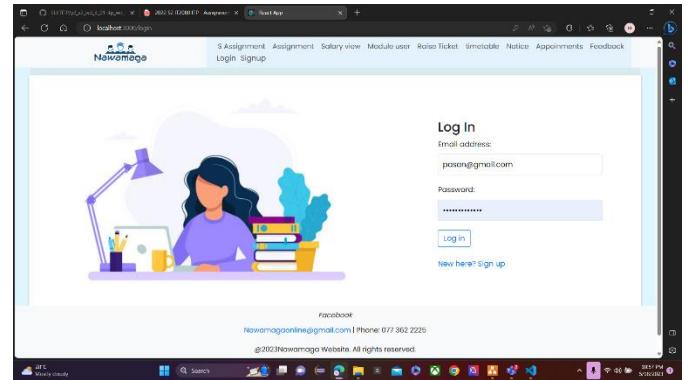


Figure 0:3 Login

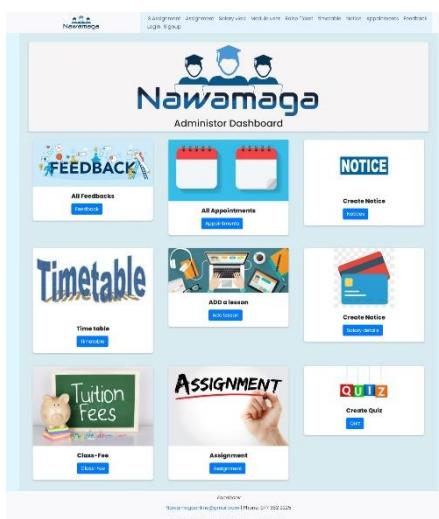


Figure 0:5 Admin Dashboard

3.6.2 Student Management – IT21247804

This screenshot shows the 'Feedback' section of the student management application. It displays a list of feedback entries from students. Each entry includes the student's name, topic/reason, and a brief description. There are 'Update' and 'Delete' buttons for each entry. A modal window titled 'FEEDBACK' is open, showing fields for 'Name', 'Topic/Reason', and 'Description', along with a 'Add Feedback' button.

Figure 0:7 Feedback

This screenshot shows the 'Appointment' section of the student management application. It displays a list of appointments with teacher names and details. A modal window titled 'Appointment' is open, showing fields for 'Teacher name', 'date' (mm/dd/yyyy), and a 'Add Appointment' button.

Figure 0:6 Appointment

3.6.3 Support Services Management – IT21206078

This screenshot shows the 'Support Services' form. It includes fields for 'SDT Name', 'Email', 'Contact Number', 'Module', 'Request/Inquiry type' (a dropdown menu), 'Message', and a 'Submit ticket' button. To the right of the form is a cartoon illustration of two people interacting with a smartphone displaying various icons like a mail envelope and a warning sign.

Figure 0:9 Support Services- form

This screenshot shows the 'My Tickets' view. It lists a single ticket for 'STU: thamindu'. The ticket details include email (i21206078@myiltik), contact number (797287432), module (b20), request/inquiry type (Other), and message (There are too many distractions). Below the ticket list is a message box containing a thank you note and a reference number (KMOV-4232-OROR).

Figure 0:8 View data

3.6.4 Timetable and Notice Management – IT21315664

The screenshot shows a table of notices:

Notice title	Notice note	Notice date	Actions
Exam	exams will be start from tomorrow	2023/03/04	
Assignment Submission for Nimal sir's class	All Student should handover assignment in tomorrow class	2023/05/06	
Assignment of shen sir	Assignment submission for shen sir's grade 6 class extended	2023/04/05	
Maths Class quiz	Students can take math class quiz online.	23/02/2023	

Figure 0:11 Admin View- Notice Page

The screenshot shows a table of notices:

Notice title	Notice note	Notice date
Exam	exams will be start from tomorrow	2023/03/04
Assignment Submission for Nimal sir's class	All Student should handover assignment in tomorrow class	2023/05/06
Assignment of shen sir	Assignment submission for shen sir's grade 6 class extended	2023/04/05
Maths Class quiz	Students can take math class quiz online.	23/02/2023

Figure 0:10 Student View -Notice page

The screenshot shows a weekly timetable grid:

Timetable day	6.00 a.m. ~ 9.00 a.m.	9.00 a.m. ~ 12.00 p.m.	12.00 a.m. ~ 3.00 p.m.	3.00 p.m. ~ 6.00 p.m.	Actions
Monday	Nothing	Nothing	Geography - Sandun Sir	Tamil - Anoja miss	
Tuesday	English- Nimal sir	Sinhala- Senthil sir	Geography - Sandun Sir	science- Sahon sir	
Wednesday	maths- Thian Sir	Tamil - Anoja miss	Sinhala- Senthil sir	English-nimal sir	
Thursday	Nothing	maths- Thian Sir	Geography - Sandun Sir	Tamil - Anoja miss	
Friday	Nothing	science- Sahon sir	English-nimal sir	Tamil - Anoja miss	

Figure 0:13 Admin View - Timetable

The screenshot shows a weekly timetable grid:

Timetable day	6.00 a.m. ~ 9.00 a.m.	9.00 a.m. ~ 12.00 p.m.	12.00 a.m. ~ 3.00 p.m.	3.00 p.m. ~ 6.00 p.m.
Monday	Nothing	Nothing	Geography - Sandun Sir	Tamil - Anoja miss
Tuesday	English-Nimal sir	Sinhala- Senthil sir	Geography - Sandun Sir	science- Irfan sir
Wednesday	maths- Thian Sir	Tamil - Anoja miss	Sinhala- Senthil sir	English-nimal sir
Thursday	Nothing	maths- Thian Sir	Geography - Sandun Sir	Tamil - Anoja Miss
Friday	Nothing	science- Sahon sir	English-nimal sir	Tamil - Anoja miss

Figure 0:12 Student View - Timetable

3.6.5 Class Fee Management – IT21222672

The screenshot shows a form titled "Add New Subject For Payment" with fields for Student No., Subject streams, Type, and Email. Below the form is a table of existing subjects:

Student	Subject	Class Type	Email	Actions
Mathematics	Normal	Normal	horanamogaonline@gmail.com	

Figure 0:15Add class fee and View class fees.

The screenshot shows a table of subjects:

student No	Subject	Class Type	Email
9	Mathematics stream	Revision	horanamogaonline@gmail.com

Figure 0:14 Admin View – Class Fees

3.6.6 Module Management – IT21327780

The screenshot shows the 'Add a new Video' form. It includes fields for 'Module Title' and 'Module Description', and a 'Add Details' button. Below this, there's a video player showing a scene with rabbits in a grassy field. A 'Choose File' button is present, but no file has been selected. The interface is clean with a light blue header and white background.

Figure 0:17 Admin View - Module page

The screenshot shows a video thumbnail of a rabbit in a field. The video player controls are visible at the bottom. The page includes sections for 'Engineering' and 'Information Technology'. At the bottom, there's a Facebook-like footer with contact information: 'Nowomagaonline@gmail.com | Phone: 077 362 2225' and '©2022 Nowomaga Website. All rights reserved.'

Figure 0:16 Student View Module page

3.6.7 Assignment Management – IT21326868

The screenshot shows the 'Add a New Assignment' form. It includes fields for 'Assignment Topic', 'Assignment Number', 'Assignment Description', and 'Due Date' (mm/dd/yyyy). A 'Generate PDF' button is also present. On the right, there's a preview of an assignment titled 'essay' with details: Number: 8, Due date: 2023-05-29T00:00:00Z, and a 'Moths assignment' link. The interface is consistent with the previous module screenshot.

Figure 0:19 Admin View - Assignment

The screenshot shows a list of assignments. The first assignment is 'essay' with Number: 8, Due date: 2023-05-29T00:00:00Z. The second assignment is 'Maths assignment' with Number: 2, Due date: 2023-05-29T00:00:00Z. Both assignments have a 'View' link next to them. The interface follows the same design as the other student view pages.

Figure 0:18 Student View - Assignment

3.6.8 Staff Management – IT21219634

The screenshot shows a form titled 'Add a New Salary'. It includes fields for 'Full Name', 'Email', 'No Of Students' (set to 'One Student Fee'), 'Salary', 'Net Salary', and 'Payment Date' (set to '06/06/2023'). Below the form is a footer with social media links (Facebook, Instagram, YouTube) and copyright information: 'Navanmagaeonline@gmail.com | Phone: 077 362 2225' and '©2023 Navanmaga Website. All rights reserved.'

Figure 0:21 Add Salary

The screenshot displays a table with columns: 'Full Name', 'Email', 'NetSalary', and 'Date'. The data shows three entries: 'try' (email: poson@gmail.com, NetSalary: Rs.2770, Date: 2023-06-09), 'Sapna De Zoysa' (email: sapna@gmail.com, NetSalary: Rs.2770500, Date: 2023-06-09), and 'sAPNA' (email: posangami.com, NetSalary: Rs.9000, Date: 2023-06-09). A red button labeled 'Print report' is visible above the table.

Full Name	Email	NetSalary	Date
try	poson@gmail.com	Rs.2770	2023-06-09
Sapna De Zoysa	sapna@gmail.com	Rs.2770500	2023-06-09
sAPNA	posangami.com	Rs.9000	2023-06-09

Figure 0:20 Admin View

3.6.9 Quiz Management – IT21328916

The screenshot shows a dashboard with three cards: 'M002 GRADE 11' (Update, Print), 'M004 GRADE 11' (Update, Print), and 'M003 GRADE 10' (Update, Print). Below the cards are buttons for 'Add New Quiz', 'Logout', and 'View open to quiz'. The footer contains social media links and copyright information: 'Navanmagaeonline@gmail.com | Phone: 077 362 2225' and '©2023 Navanmaga Website. All rights reserved.'

Figure 0:23 Admin View Quiz home page

The screenshot shows a form for adding a quiz. It includes fields for 'Module Code' (M002), 'Selected Grade' (11), 'Teacher Id' (0031), 'Quiz Start Date' (2023-05-06T00:00:00), 'Quiz End Date' (2023-05-06T11:59:59), 'Number of Ques' (2), and 'Questions' (with 4 questions listed: 1, 2, 3, 4). Below the form is a 'Add Question' section with fields for 'Question Text' (Q1) and 'Correct Answer Value' (1). A green 'Submit' button is at the bottom.

Figure 0:22 Admin View - Add Quiz

The screenshot shows a form for updating a quiz. It includes fields for 'Module Code' (M002), 'Teachers Id' (0031), 'Student Grade' (11), 'Quiz Start Time' (2023-05-06T00:00:00), 'Quiz End Time' (2024-05-06T11:59:59), and 'Quiz Start Date' (2023-05-06T11:59:59). Below the form is a green 'Update' button. The footer contains social media links and copyright information: 'Navanmagaeonline@gmail.com | Phone: 077 362 2225'.

Figure 0:24 Admin View Update Quiz

3.7 Implementation

The project was implemented using a combination of HTML, CSS, and JavaScript languages, adhering to the traditional three-tier architectural pattern of the MERN stack technology. HTML, known as the Hypertext Markup Language, was used to define the structure of web pages, while CSS, or Cascading Style Sheets, handled the layout and presentation. These two languages formed the foundation for creating web pages and applications, complemented by graphics and coding. The development process also incorporated the Bootstrap framework to enhance design capabilities.

JavaScript, a versatile programming language that is free and cross-platform, was employed to build network-centric applications. Its simplicity and comprehensibility made it a suitable choice for this project. The MERN stack, a widely adopted technology stack, was selected for development. The acronym MERN represents the following technologies: MongoDB as the database, Express.js and Node.js for the backend, and React.js for the frontend. React.js, a JavaScript framework, facilitated the development of dynamic client-side applications using HTML. Express.js, a server-side framework, operated within Node.js. MongoDB served as the NoSQL database management system, storing data as documents with key-value pairs in a JSON-like format. In the context of video uploading, Firebase database was utilized.

Although the MEAN stack is another popular choice, the decision was made to use MERN for several reasons. MEAN utilizes the AngularJS framework for the frontend, while MERN utilizes React.js. React.js was found to perform better and be more beginner-friendly compared to AngularJS. Additionally, AngularJS uses the Typescript language, whereas React.js uses JavaScript. MERN was chosen because of its better documentation and suitability for managing larger projects.

For development purposes, Visual Studio Code was utilized as the Integrated Development Environment (IDE). Visual Studio Code supports multiple languages and provides features such as syntax highlighting, bracket matching, simple customization, and collaborative capabilities.

During the testing phase, the Postman API was employed. Postman's Node.js-based runtime facilitated the creation of functional tests, integration tests, regression tests, and other testing types, supporting popular patterns and frameworks.

3.7.1 Implementation Structure

The following picture outlines the folder structure Database Connection

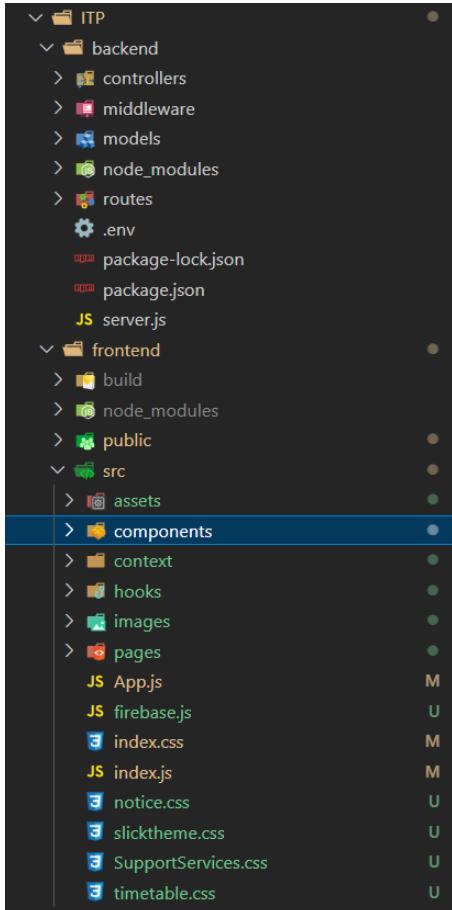
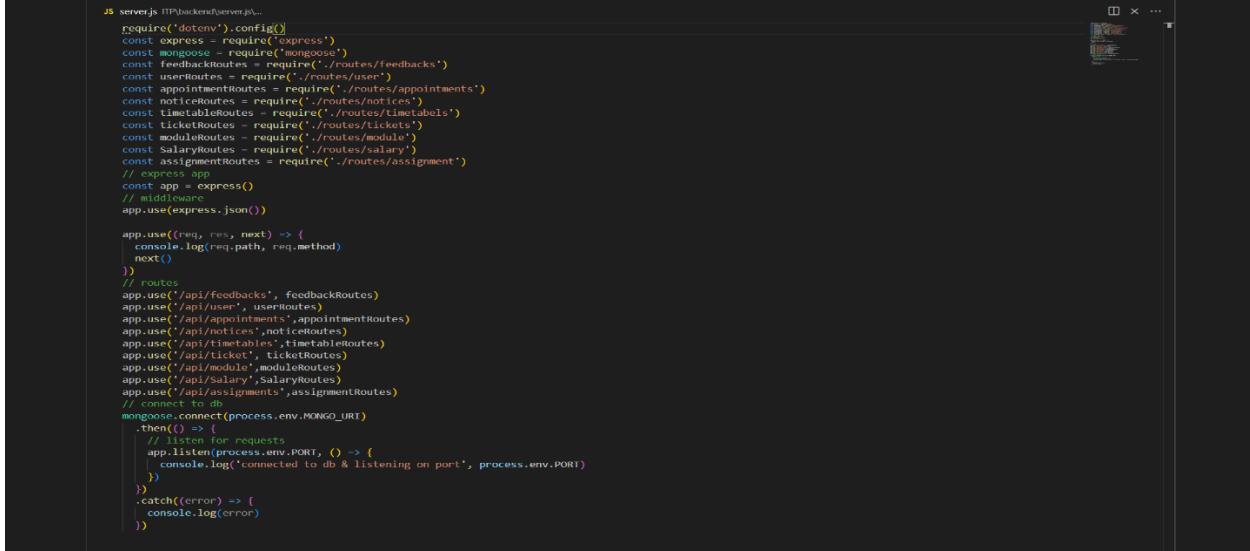


Figure 0:25 Folder Structure

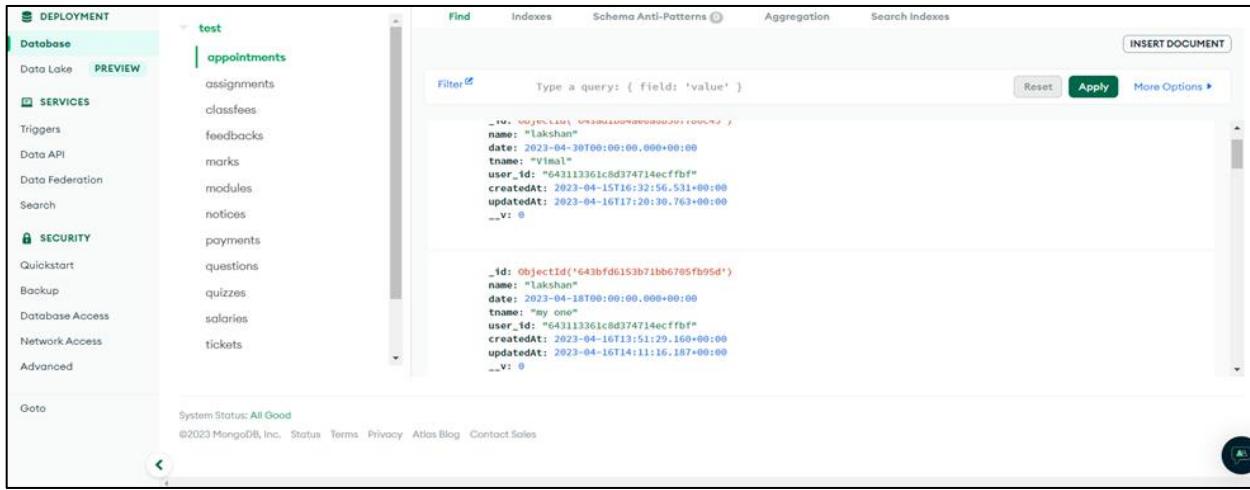
3.7.2 Database Connection

This code segment is used to connect the MongoDB database to the VS Code Project, And a picture of the database.



```
JS server.js (1 file) ...
require('dotenv').config()
const express = require('express')
const mongoose = require('mongoose')
const feedbackRoutes = require('./routes/feedbacks')
const userRoutes = require('./routes/user')
const appointmentRoutes = require('./routes/appointments')
const noticeRoutes = require('./routes/notices')
const timetableRoutes = require('./routes/timetables')
const ticketRoutes = require('./routes/tickets')
const moduleRoutes = require('./routes/module')
const salaryRoutes = require('./routes/salary')
const assignmentRoutes = require('./routes/assignment')
// express app
const app = express()
// middleware
app.use(express.json())

app.use((req, res, next) => {
  console.log(req.path, req.method)
  next()
}) // routes
app.use('/api/feedbacks', feedbackRoutes)
app.use('/api/user', userRoutes)
app.use('/api/appointments', appointmentRoutes)
app.use('/api/notices', noticeRoutes)
app.use('/api/timetables', timetableRoutes)
app.use('/api/ticket', ticketRoutes)
app.use('/api/module', moduleRoutes)
app.use('/api/salary', salaryRoutes)
app.use('/api/assignments', assignmentRoutes)
// connect to db
mongoose.connect(process.env.MONGO_URI)
  .then(() => {
    // listen for requests
    app.listen(process.env.PORT, () => [
      console.log(`connected to db & listening on port ${process.env.PORT}`)
    ])
  })
  .catch((error) => [
    console.log(error)
  ])
})
```



The screenshot shows the MongoDB Atlas interface. On the left, the navigation sidebar includes sections for DEPLOYMENT, SERVICES, and SECURITY. Under the Database section, 'test' is selected, and it lists several collections: appointments, assignments, classifies, feedbacks, marks, modules, notices, payments, questions, quizzes, salaries, and tickets. The main panel displays the contents of the 'appointments' collection. A search bar at the top says 'Type a query: { field: 'value' }'. Below the search bar, two documents are shown:

```
_id: ObjectId("643bfd6153b71bb6705fb95d")
name: "Lakshan"
date: 2023-04-30T00:00:00.000+00:00
tname: "Vimal"
user_id: "643113361c8d374714ecffbf"
createdAt: 2023-04-15T16:32:56.531+00:00
updatedAt: 2023-04-16T17:20:30.763+00:00
__v: 0

_id: ObjectId("643bfd6153b71bb6705fb95e")
name: "Lakshan"
date: 2023-04-18T00:00:00.000+00:00
tname: "my one"
user_id: "643113361c8d374714ecffbf"
createdAt: 2023-04-16T13:51:29.160+00:00
updatedAt: 2023-04-16T14:11:16.187+00:00
__v: 0
```

3.7.3. Code Special Algorithms

1. User validations

```
JS userModel.js // backend/models/userModel.js...
// validation
if (!email || !password) {
  throw Error('All fields must be filled')
}
if (!validator.isEmail(email)) {
  throw Error('Email not valid')
}
if (!validator.isStrongPassword(password)) {
  throw Error('Password not strong enough')
}

const exists = await this.findOne({ email })

if (exists) {
  throw Error('Email already in use')
}

const salt = await bcrypt.genSalt(10)
const hash = await bcrypt.hash(password, salt)

const user = await this.create({ email, password: hash })

return user
}

// static login method
userSchema.statics.login = async function(email, password) {

  if (!email || !password) {
    throw Error('All fields must be filled')
  }

  const user = await this.findOne({ email })
  if (!user) {
    throw Error('Incorrect email')
  }

  const match = await bcrypt.compare(password, user.password)
  if (!match) {
    throw Error('Incorrect password')
  }

  return user
}
```

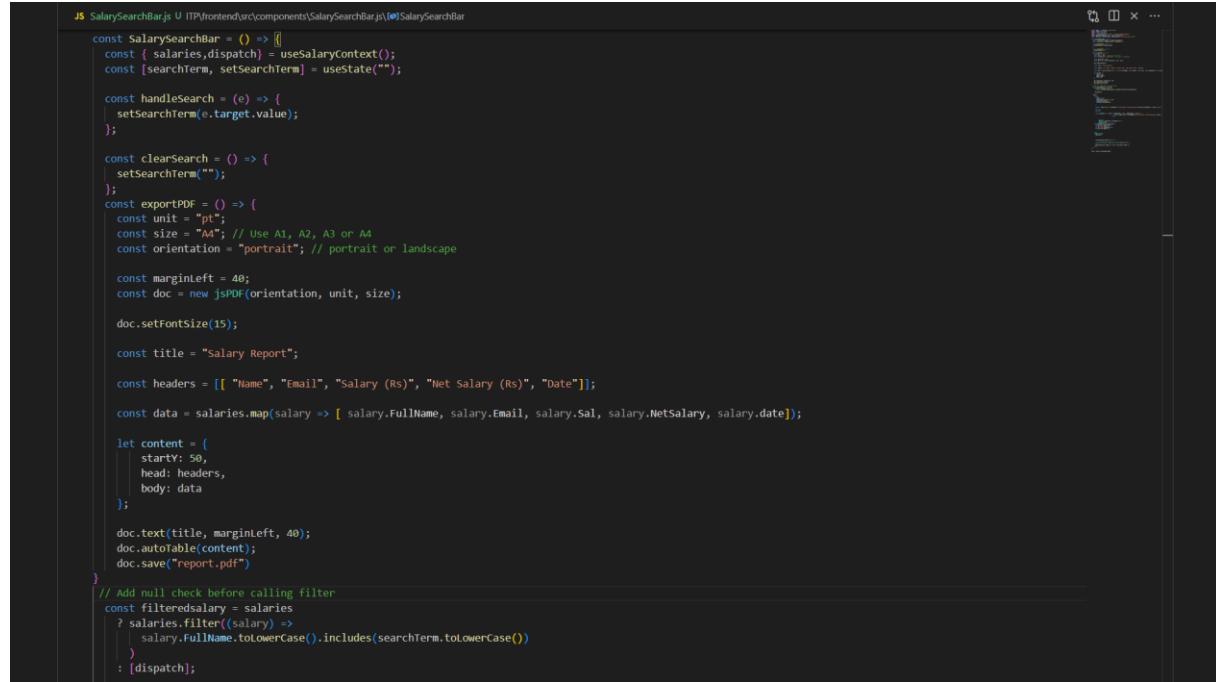
2. Empty fields validations

```
// create new feedback
const createFeedback = async (req, res) => {
  const {name, topic, description} = req.body

  let emptyFields = []

  if(!name) {
    emptyFields.push('name')
  }
  if(!topic) {
    emptyFields.push('topic')
  }
  if(!description) {
    emptyFields.push('description')
  }
  if(emptyFields.length > 0) {
    return res.status(400).json({ error: 'Please fill in all the fields', emptyFields })
  }
}
```

3. Search-Bar and report Generation



```
JS SalarySearchBar.js U ITP/frontend/src/components/SalarySearchBar.js [diff] SalarySearchBar

const SalarySearchBar = () => {
  const { salaries, dispatch } = useSalaryContext();
  const [searchTerm, setSearchTerm] = useState("");

  const handleSearch = (e) => {
    setSearchTerm(e.target.value);
  };

  const clearSearch = () => {
    setSearchTerm("");
  };
  const exportPDF = () => {
    const unit = "pt";
    const size = "A4"; // Use A1, A2, A3 or A4
    const orientation = "portrait"; // portrait or landscape

    const marginLeft = 40;
    const doc = new jsPDF(orientation, unit, size);

    doc.setFontSize(15);

    const title = "Salary Report";

    const headers = [[ "Name", "Email", "Salary (Rs)", "Net Salary (Rs)", "Date"]];

    const data = salaries.map(salary => [ salary.FullName, salary.Email, salary.Sal, salary.NetSalary, salary.date ]);

    let content = {
      startY: 50,
      head: headers,
      body: data
    };

    doc.text(title, marginLeft, 40);
    doc.autoTable(content);
    doc.save("report.pdf")
  }
  // Add null check before calling filter
  const filteredSalary = salaries
    ? salaries.filter((salary) =>
      salary.FullName.toLowerCase().includes(searchTerm.toLowerCase())
    )
    : [dispatch];
}


```

Firebase Connection for Video Uploading in Module Management

```
1 import { initializeApp } from "firebase/app";
2
3 import {getStorage} from "firebase/storage";
4
5 const firebaseConfig = {
6   apiKey: "AIzaSyAfN0SpVjYP5gh4IY7ilp0Jknn0Hcc2Llg",
7   authDomain: "modulevideo.firebaseio.com",
8   projectId: "modulevideo",
9   storageBucket: "modulevideo.appspot.com",
10  messagingSenderId: "9900189263",
11  appId: "1:9900189263:web:5b99aa856b12c93ce52c3a",
12  measurementId: "G-1WT39VF3YC"
13};
14
15 // Initialize Firebase
16 const app = initializeApp(firebaseConfig);
17
18
19 export const storage = getStorage(app);
```

Chapter 4: Testing

4.1 Student Management Test Cases

Test case ID: Test_001	Test designed by: IT21247804-Baddewithana P
Test title: Student Registration	Test designed day: 2023.05.07
Test priority (High/Medium/Low): High	Test executed by: IT21247804 -Baddewithana P
Module name: Student Management	Test executed day: 06.05.2023
Description: Register New Student to the system using a sign-up form.	
Preconditions (if there are any): should have a valid e-mail and browsing facility.	
Dependencies (if there are any):	
Test steps: 1. Click the student registration button. 2. Fill in the required details. 3. Click the submit button.	
Pass Condition: After submitting the student registration form, newly registered students are displayed in the Student Home page.	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_001	User-email: “baddebadde777@gmail.com” Password: ABCabc1234*	Students get registered and prompt to main page.	Students get registered and prompt to main page.	Pass	System let student to submit their details with valid credentials.
Test_001	Password: ABC	Display an error message called “Password not strong enough”	Display an error message called “Password not strong enough”	Pass	System shouldn't let the student to submit the form without strong password.
Test_001	User email: “baddebadde777@gmail.com” Password: ABCabc1234*	Display an error message called “email already exists”	Display an error message called “email already exists”	Pass	System shouldn't let the form if the email already exists.

Test case ID: Test_002	Test designed by: IT21247804-Baddewithana P
Test title: Student login	Test designed day: 2023.05.07
Test priority (High/Medium/Low): High	Test executed by: IT21247804 -Baddewithana P
Module name: Student Management	Test executed day: 06.05.2023
Description: Student login to the system using login form	
Preconditions (if there are any): Student must be registered to the system.	
Dependencies (if there are any):	
Test steps: <ol style="list-style-type: none"> 1. Click the student login button. 2. Fill the required details 3. Click the submit button 	
Pass Condition: After submitting the student login form, already registered students are displayed in the Student Home page.	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_002	User email: Ashan666@gmail.com Password: Ashanshan1234*	Display an error called “E-mail does not exist in the system”.	Display an error called “E-mail does not exist in the system”.	Pass	System shouldn't let the form to submit the form without registering to system.
Test_002	“ baddebadde777@gmail.com ” Password: ABCAbc12*	Display an error message called “Wrong password”	Display an error message called “Wrong password”	Pass	System shouldn't let the student to submit the form without correct password.
Test_002	User email: pasanbadde@gmail.com	Display an error message called “email already exists”	Display an error message called “email already exists”	Pass	System shouldn't let the form if the email already exists.

Test case ID: Test_003	Test designed by: IT21247804-Baddewithana P
Test title: Student give Feedback	Test designed day: 2023.05.07
Test priority (High/Medium/Low): High	Test executed by: IT21247804 -Baddewithana P
Module name: Student Management	Test executed day: 06.05.2023
Description: Registered student gives feedback.	
Preconditions (if there are any): Student must login to the system.	
Dependencies (if there are any):	
Test steps:	
1. Student must login to the system. 2. Student must click Feedback link in the navbar. 3. Student fill out the feedback details. 4. Student click submit button. 5. Student check newly added feedback on feedback page.	
Pass Condition: After submitting the feedback form the newly added feedback is displayed in feedback page along with other feedbacks.	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_003	name: Pasan Title: canteen Description: Ok	Student feedback details published in Feedback page.	Student feedback details published in Feedback page.	Pass	System let student to submit their details with all inputs are filled.
Test_003	Title: canteen Description: Ok	Display an error message called “Fill out all the fields.”	Display an error message called “Fill out all the fields.”	Pass	System shouldn't let the student to submit the form without the name.
Test_003	name: Pasan Title: canteen	Display an error message called “Fill out all the fields.”	Display an error message called “Fill out all the fields.”	Pass	System shouldn't let the student to submit the form without the title.

Test case ID: Test_004	Test designed by: IT21247804-Baddewithana P
Test title: Student make an appointment	Test designed day: 2023.05.07
Test priority (High/Medium/Low): High	Test executed by: IT21247804 -Baddewithana P
Module name: Student Management	Test executed day: 06.05.2023
Description: Registered student make an appointment	
Preconditions (if there are any): Student must login to the system.	
Dependencies (if there are any):	
Test steps: <ol style="list-style-type: none"> 1. Student must login to the system. 2. Student must click appointment link in the navbar. 3. Student fill out the feedback details. 4. Student click submit button. 5. Student check newly added feedback on feedback page. 	
Pass Condition: After submitting the appointment form the newly added feedback is displayed in appointment Page along with other appointments.	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_004	name: Pasan Teacher name: Kamal sir Date: 2023.04.25	Student appointment published in Feedback page.	Student feedback details published in Feedback page.	Pass	System let student to submit their details with all inputs are filled.
Test_004	name: Pasan Teacher name: Kamal sir	Display an error message called “Fill out all the fields.”	Display an error message called “Fill out all the fields.”	Pass	System shouldn't let the student to submit the form without the Date.
Test_004	Teacher name: Kamal sir Date: 2023.04.25	Display an error message called “Fill out all the fields.”	Display an error message called “Fill out all the fields.”	Pass	System shouldn't let the student to submit the form without the name.

4.2 Support Services Test Cases

Test case ID: Test_01	Test designed by: IT21206078-Ilesingha I.T.S.
Test title: Student raise a ticket	Test designed day: 2023.05.02
Test priority (High/Medium/Low): Medium	Test executed by: IT21206078-Ilesingha I.T.S.
Module name: Student Management	Test executed day: 04.05.2023
Description: Registered student raise tickets	
Preconditions (if there are any): Student must login to the system.	
Dependencies (if there are any):	
Test steps: <ol style="list-style-type: none">1. Student logs into the system.2. Student clicks on the support services link in the navbar.3. The support services page is displayed.4. Student fills out the ticket form with the required information.5. Student clicks the submit button.6. The ticket is successfully submitted.7. Student navigates to the ticket management system.8. The ticket management system page is displayed.9. Student verifies that the newly added ticket is displayed in the list of tickets.10. Student checks if the details of the newly added ticket match the information provided during submission.	
Pass Condition: After submitting the ticket, the newly added ticket is successfully displayed in the ticket management system along with other tickets	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_01	Name: Thamindu Email: it21206078@my.sliit.lk Contact no: 767284732 Module: Science Request type: I want to... Message: There are too many...	Newly added ticket is displayed in the ticket management system along with other tickets	The newly added ticket is displayed in the ticket management system along with other tickets	Pass	System let student to submit their details with all inputs are filled.
Test_01	Name: Thamindu Module: Science Request type: I want to...	Display an error message called “Fill out all the fields.”	Display an error message called “Fill out all the fields.”	Pass	System shouldn't let the student to submit the form without the all fields.
Test_01	Email: it21206078@my.sliit.lk Contact no: 767284732	Display an error message called “Fill out all the fields.”	Display an error message called “Fill out all the fields.”	Pass	System shouldn't let the student to submit the form without all fields.

4.3 Timetable and Notice Management Test Cases

Test case ID: Test_01	Test designed by: IT21315664 – Rajawasan W.H.H.S.
Test title: Notice Management	Test designed day: 2023.05.08
Test priority (High/Medium/Low): High	Test executed by: IT21315664 – Rajawasan W.H.H.S.
Module name: Notice Management	Test executed day: 07.05.2023
Description: Create new Notices and send to students	
Preconditions (if there are any): Admin should login to the system using login credentials	
Dependencies (if there are any):	
Test steps: <ol style="list-style-type: none"> 1. Admin login to the system 2. Click the navbar notice button. 2. Navigate to notice page 2. Fill in the required details for the notice form. 3. Click the Add notice button. 	
Pass Condition: After adding the notice details form, registered students can view Notices in Home page.	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_01	<p>Title: Science Class</p> <p>Note: Dear Students, it is to here by inform you that your Science Online Test (Online MCQ based test) will be held from 17-Aug-2020 onwards. It is compulsory for all the students to attempt the test. In case of absence, student will be responsible for non-processing of the result as per the scheme of examination.</p> <p>Published Date: 2023.08.09</p>	The Notice details are successfully added and displayed on the home page	The Notice details are successfully added and displayed on the home page	Pass	The system allows the admin to add a notice and displayed on the home page.

Test_01	<p>Note: Dear Students, it is to here by inform you that your Science Online Test (Online MCQ based test) will be held from 17-Aug-2020 onwards. It is compulsory for all the students to attempt the test. In case of absence, student will be responsible for non-processing of the result as per the scheme of examination.</p> <p>Published Date: 2023.08.09</p>	Display an error message indicating that the notice title field must be filled.	Display an error message indicating that the notice title field must be filled.	Pass	The system does not allow the admin to add notices without entering a notice title.
Test_01		Display an error message indicating that the published date field must be filled. .”	Display an error message indicating that the published date field must be filled	Pass	The system does not allow the admin to add notices without filling published date.

4.4 Class Fee Management Test Cases

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_001	Student No = null Subject = Biology Type = Theory Email = anna@gmail.com	System sends error message fill all the fields.	System sends error message fill all the fields.	Pass	System should not let the submit the form without student no.
Test_001	Student No = 9 Subject = null Type = Theory Email = anna@gmail.com	System sends error message fill all the fields.	System sends error message fill all the fields.	Pass	System should not let the submit the form without subject.
Test_001	Student No = 9 Subject = Biology Type = null Email = anna@gmail.com	System sends error message fill all the fields.	System sends error message fill all the fields.	Pass	System should not let the submit the form without type.
Test_001	Student No = 9 Subject = Biology Type = Theory Email = null	System sends error message fill all the fields.	System sends error message fill all the fields.	Pass	System should not let the submit the form without email.
Test_001	E mail = annagmail	System send error message email is not correct format.	System send error message email is not correct format.	Pass	System should not let the submit the email without correct format.
Test_001	Student No = aa	System send error message student no is not correct format.	System send error message student no is not correct format.	Pass	System should not let the submit the student no without correct format.

4.5 Module Management Test Cases

Test case ID: Test_01	Test designed by: IT21327780 – Liyanage L. D. P. D
Test title: Tutor added lecture recording	Test designed day: 2023.05.04
Test priority (High/Medium/Low): Medium	Test executed by: IT21327780 – Liyanage L. D. P. D
Module name: Module Management	Test executed day: 2023.05.07
Description: Tutor added new video into Module Page	
Preconditions (if there are any): Tutor must login to the system.	
Dependencies (if there are any):	
Test steps: <ol style="list-style-type: none"> 4. Tutor logs into the system. 5. Tutor clicks on the lecture recordings button in the navbar. 6. The module page is displayed. 7. Tutor fills out required fields and selects video file. 8. Tutor clicks the add video. 9. The video and description were successfully added. 10. The module page is displayed a newly added video. 8. Tutor verifies that the newly added ticket is displayed in the list of videos. 9. Tutor checks if the details of the newly added video and description match the submission. 	
Pass Condition: After submitting the video, the newly added video is successfully displayed in the module page along with other videos	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_01	Video File: sample_video.mp4 Module Name: Machine Learning - Description: This video introduces the basics of machine learning.	The video and its details are successfully uploaded and displayed on the module page	The video and its details are successfully uploaded and displayed on the module page	Pass	The system allows the tutor to upload a video and its details, and the uploaded video is displayed on the module page.

Test_01	<p>video File: sample_video.mp4</p> <p>Module Name: Machine Learning</p> <p>- Description: This video introduces the basics of machine learning</p>	<p>Display an error message indicating that the video title field must be filled.</p>	<p>Display an error message indicating that the video title field must be filled</p>	Pass	The system does not allow the tutor to upload a video without entering a video title.
Test_01	<p>Video File: sample_video.mp4</p> <p>Module Name: (blank)</p> <p>- Description: This video introduces the basics of machine learning</p>	<p>Display an error message indicating that the module name field must be filled.</p> <p>..</p>	<p>Display an error message indicating that the module name field must be filled</p>	Pass	The system does not allow the tutor to upload a video without selecting a module name.

4.6 Quiz Management Test Cases

Test case ID: Test_01	Test designed by: IT21328916 K.T.S. De Silva
Test title: Insert Quizzes	Test designed day: 2023.05.01
Test priority (High/Medium/Low): Medium	Test executed by: IT21328916 K.T.S. De Silva
Function name: Quiz Management	Test executed day: 06.05.2023
Description: Add a new Quiz for quiz module pages	
Preconditions (if there are any): The teacher must log in to the system.	
Test steps: <ol style="list-style-type: none"> 1. Teacher should log into the system. 2. Navigate to teacher dashboard. 3. Click on Quiz upload. 4. Click add new Quiz button. 5. Fill in the required details. 6. Click the submit button. 7. Check if the newly added Quiz is listed in the Quiz Edit page. 	
Pass Condition: After submitting Quiz students can answer it and result should be uploaded to database by system. After that the teacher can see the result and download the result sheet	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_01	Quiz Module Number: M001	Entered values displayed in “Quiz Name” field.	Entered values displayed in “Quiz Name” field.	Pass	System shouldn't let submit the form without having a Name in the “Name Field”
Test_02	Enter No of Quiz: Qwert	Display an error called “Invalid data type!”	Display an error called “Invalid data type!”	Pass	System let submit only number in the “No of Quiz” field
Test_01	Enter Date of Quiz: 2023-05-07	Entered values displayed in “Date of Quiz” field	Entered values displayed in “Date of Quiz” field	Pass	System let submit only 4 digits for Year and 2 digits for Month and Date in the “Date of Quiz” field

4.7 Assignment Management Test Cases

Test case ID: Test_01	Test designed by: IT21326868-Jayasooriya J.M.D. T
Test title: Teacher added assignments	Test designed day: 2023.05.03
Test priority (High/Medium/Low): Medium	Test executed by: IT21326868- Jayasooriya J.M.D. T
Module name: Assignment Management	Test executed day: 2023.05.07
Description: Teacher added new assignments to the Assignment Management page.	
Preconditions (if there are any): Teacher must login to the system.	
Dependencies (if there are any):	
Test steps: <ol style="list-style-type: none"> 1. Teacher logs into the system. 2. Teacher clicks on the Add Assignment button in the navbar. 3. The Assignment Management page is displayed. 4. Teacher fills out required fields and selects image of the Assignment. 5. Teacher clicks the Add Assignments. 6. The image and description were successfully added. 7. The module page displays newly added Assignments. 8. The teacher verifies that the newly added assignments are displayed in the list of assignments. 9. The teacher checks if the details of the newly added assignment and description are correct. If there is any problem, the teacher can edit or delete from the page. 	
Pass Condition: After submitting the Assignment as an image, the newly added assignment is successfully displayed in the Assignment Management page along with other Images.	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_01	Image File: sen.jpg Assignment Number: B001 Assignment topic: Biology Description: Human body, Lesson 1 part 2 Due Date: 08/05/23	Display an error message indicating that the assignment topic title field must be filled.	Display an error message indicating that the assignment topic field must be filled.	Pass	The system does not allow the teacher to upload a image without entering a assignment topic.

Test_01	Image File: dul.jpg Assignment Number: P005 Assignment topic: Physics Description: Force, Lesson 3 part 2 Due Date: 11/05/23	The Image and its details are successfully uploaded and displayed on the page.	The image and its details are successfully uploaded and displayed on the page.	Pass	The system allows the teacher to upload a assignment image and its details, and the uploaded image is displayed on the page.
Test_01	Image File: kav.jpg Assignment Number: C011 Assignment topic: Chemistry Description: Introduction of Lesson 1 Due Date: 9/05/23	Display an error message indicating that the assignment topic name field must be filled. ”	Display an error message indicating that the assignment name field must be filled	Pass	The system does not allow the teacher to upload image without selecting a assignment topic

Test case ID: Test_02	Test designed by: IT21326868-Jayasoorya J.M.D.T
Test title: Student submit assignments	Test designed day: 2023.05.03
Test priority (High/Medium/Low): Medium	Test executed by: IT21326868- Jayasoorya J.M.D.T
Module name: Assignment Management	Test executed day: 2023.05.08
Description: Students upload their complete assignments to the Assignment Management page.	
Preconditions (if there are any): Student must login to the system.	
Dependencies (if there are any):	
Test steps: <ol style="list-style-type: none"> 1. Students log into the system. 2. Students click on the New Assignment button in the navbar. 3. The Assignment page is displayed. 4. Students select an image of the Assignment. 5. Students can view them. 6. The assignment image and description were successfully added. 7. The assignment management page displays newly added Assignments. 8. Student can complete that assignments and upload a photo of answers as image. 	
Pass Condition: After submitting the Complete Assignment as a image, the added assignments are successfully displayed in the Assignment Management page along with other Images.	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_02	Student Number: Bio21713 Image File: sha.jpg	Display an error message indicating that the student number field must be filled.	Display an error message indicating that the student field must be filled.	Pass	The system does not allow the student to upload a image without entering a student number
Test_02	Student Number: phy76508 Image File: wij.jpg	The Image and its details are successfully uploaded and displayed on the page.	The image and its details are successfully uploaded and displayed on the page.	Pass	The system allows the student to upload a assignment image and its details, and the uploaded image is displayed on the page.

4.8 Staff Management Test Cases

Test case ID: Test_001	Test designed by: IT21219634-De Zoysa A.P.S
Test title: Student Registration	Test designed day: 2023.05.07
Test priority (High/Medium/Low): High	Test executed by: IT21219634-De Zoysa A.P.S
Module name: Staff Management	Test executed day: 06.05.2023
Description: Add salary details and calculate salary	
Preconditions (if there are any): should have valid login credential to	
Dependencies (if there are any):	
Test steps: <ol style="list-style-type: none"> 1. Click the add salary button. 2. Fill in the required details. 3. Click the submit button. 	
Pass Condition: After submitting the Salary form, newly salary details are displayed	

Test ID	Test Inputs	Expected Output	Actual Output	Results (Pass/Fail)	Comments
Test_001	Email: “sapnadezoysa7788@gmail.com”	After Displayed Salary details successfully	After Displayed Salary details successfully	Pass	This field input type is email. We should type email correctly
Test_001	Full Name= ” Sapna” Email= sapnadezoysa7788@gmail.com No of Student=100 one Student Fee=10000 Net salary=10000 Salary=10000 Date=2023.05.08	After Displayed Salary details successfully	After Displayed Salary details successfully	Pass	All the fields are required to fill.

Chapter 5: Evaluation and Conclusion

5.1 Evaluation

5.1.1 Introduction

Evaluation, as defined by Scriven, is a systematic and objective process used to assess the merit, value, and significance of a subject or endeavor. It involves the careful examination of various criteria and standards to determine the effectiveness, efficiency, and overall impact of a program, project, intervention, or initiative. Evaluation plays a vital role in decision-making, providing valuable insights and data to inform future actions and improvements.

The primary purpose of evaluation is to gain a comprehensive understanding of the subject under evaluation and its outcomes. By examining its performance, achievements, and adherence to objectives, evaluation enables stakeholders to assess the extent to which the desired results have been achieved. It helps answer critical questions about the value, feasibility, and sustainability of the subject being evaluated. [9]

5.1.2 Evaluation Techniques

The software was evaluated using a variety of evaluation approaches. Several main methods are listed below,

- Testing.
- Interviews.
- Focus groups.
- Inspection.
- Walkthroughs.

5.1.2.1 Database testing

The system stores all the information in a database, organized into different categories and handles various types of data. This database is a critical component of the system because of any loss or corruption of data. To use queries in MongoDB, we can use the `find()` method, which allows us to retrieve documents from a collection based on specified criteria.

5.1.2.2. Focus group.

First, we clearly define the objective of the focus group evaluation. Identify what aspects of the project we want to assess or gather feedback on. By using this we can benefits include usability, user satisfaction, effectiveness, or any specific project goals. After Select participants who represent the target audience or stakeholders of our project. Aim for a diverse group to gather a range of perspectives. Communicate the

purpose and expectations of the focus group to the participants. For that we develop a set of discussion topics or questions that align with the project's evaluation objectives. These topics help prompt participants to share their opinions, experiences, and suggestions related to the project. Facilitate the focus group session, ensuring a comfortable and open environment for participants to express their views. We encourage participants to share their experiences, thoughts, and feedback based on the discussion topics. Capture notes or record the session for later analysis. Review the notes or recordings from the focus group and analyze the data collected. Look for patterns, common themes, and key insights that emerge from the participants' responses. This analysis will help identify strengths, weaknesses, and areas for improvement in the project. Prepare a summary report or presentation that highlights the main findings from the focus group evaluation. Include key themes, participant feedback, and any actionable recommendations that inform project development. Present the evaluation findings to project stakeholders, such as team members, management, clients. Discuss the results, solicit additional input, and collaborate on action steps to address the identified areas for improvement. This feedback loop helps to refine and enhance the project based on the evaluation outcomes. This iterative evaluation approach allows us to gather feedback at various stages of development and make continuous improvements.

5.1.2.3. Inspection

First, we determine the specific criteria or standards that will be used to evaluate the project during inspections. These criteria should align with project requirements, industry regulations, quality standards, or any relevant guidelines. Develop an inspection plan that outlines the scope, objectives, and methods for evaluating the project. Determine the specific areas, components, or processes that will be inspected and establish a timeline for conducting inspections. Perform the inspections according to the established plan. During inspections, assess the project against the predefined evaluation criteria. Record detailed findings and observations during inspections. Take notes, capture photographs, or create inspection reports to document any deficiencies, non-compliance, or areas that require improvement. Include specific details such as location, severity, and potential impact of the identified issues. We should review the collected inspection data and analyze the findings. Identify recurring patterns, common issues, or trends that emerge from the inspections. Assess the severity and potential risks associated with each identified problem. Prioritize the identified issues based on their severity, impact, and urgency. Prepare a summary report that communicates the inspection findings, including both positive aspects and areas for improvement. Collaborate on solutions: Share the evaluation findings and recommendations with relevant project stakeholders, such as the project team, contractors, or management. Collaboratively discuss and develop action plans to address the identified deficiencies and improve the project's quality or compliance.

5.1.2.4 Walkthroughs

First, we identify what aspects of the project you want to assess, such as functionality, usability, efficiency, safety, or any specific project goals. Then we develop a plan that outlines the walkthrough process. Determine the sequence and areas that will be covered during the walkthrough. We consider involving relevant stakeholders, such as project team members, end-users, or subject matter experts, to gain diverse perspectives. Lead the walkthrough session, ensuring that participants understand the purpose and objectives. Guide participants through the project area, highlighting key components,

processes, or functionalities. Encourage participants to ask questions, provide feedback, and share their observations during the walkthrough. We take detailed notes during the walkthrough to document key observations, issues, or areas for improvement. Capture any positive aspects or strengths as well. Note the location, description, and potential impact of each identified observation. We review the collected observations and analyze the findings. Look for patterns, common themes, and critical issues that emerge from the walkthrough. Assess the severity and potential risks associated with each identified observation.

5.1.3 Procedures of testing

Testing is crucial for evaluating system compatibility and effectiveness in addressing the problem. It focuses on software testing and helps identify and minimize errors during development. Quality and reliability are essential for success, and testing methods ensure valid and accurate results. Testing involves validation and verification, playing a vital role in the project.

Validation- focused on ensuring that the system met customer requirements.

Verification - aimed to confirm that the system fulfilled all functional requirements.

Testing was a critical and continuous phase in the software development life cycle, significantly enhancing the system's quality. Various types of testing were employed to ensure the system's effectiveness.

- Unit testing
- Integration Testing
- System Testing
- Regression Testing
- Acceptance Testing [10]

5.1.3.1. Unit Testing

During the project implementation, unit testing was conducted to test individual components in isolation. The login module was an example of a unit that underwent unit testing. Two cases were tested: the first case checked the successful login with valid credentials, while the second case tested the handling of incorrect passwords and the display of appropriate error messages. Unit testing allowed for the examination of individual components to ensure correct functionality and expected results. It helped identify and rectify issues or errors within specific units, such as the login module.

5.1.3.2. Integration Testing

Integration testing takes place after unit testing and focuses on testing the interaction between software modules based on functional specifications. For example, in the case of the login and administration modules, integration testing verifies if the login module grants access to the administration module with correct credentials. This testing ensures seamless interaction and proper functioning of modules when integrated into the software. It identifies issues and inconsistencies that may arise during integration, ensuring smooth overall software operation.

Example:

Login and User Management Integration: Integration testing would involve verifying that the login module integrates correctly with the user management module. This includes testing if user authentication works seamlessly, user roles and permissions are properly assigned, and user information is accurately synchronized between the login and user management modules.

5.1.3.3. System Testing

System testing is conducted after integration testing and aims to validate that the software meets user requirements and functions correctly in its intended environment. It encompasses both functional and non-functional requirements. During system testing, the entire system is tested by performing various processes, such as logging in, to ensure smooth operation. This testing ensures that the integrated software functions as expected and is ready for deployment.

Example: System testing would involve checking the reporting and analytics features, ensuring that the system can generate various reports, such as student progress reports, course evaluation reports, or financial reports. It would also involve testing the system's ability to provide useful analytics and insights based on the collected data.

5.1.3.4. Regression Testing

Regression testing is performed after making changes to a system application to ensure that the changes have been implemented correctly and have not introduced new errors or issues. It involves retesting specific areas of the software affected by the changes. The goal of regression testing is to validate that the existing functionality remains intact and that no unintended consequences have arisen due to the modifications.

Example: If any changes were made to the reporting and analytics features, regression testing would focus on validating that the modifications have not impacted the generation of reports or the accuracy of analytics. This would involve checking if the reports are still generated correctly and analytics insights are consistent with the expected results.

5.1.3.5. Acceptance Testing

Acceptance testing is a black box system testing method where tests are executed to represent expected results from the system. These tests are designed based on the requirements and expectations of customers or stakeholders responsible for verifying the correctness of the acceptance tests. It serves as the final evaluation stage to assess the system against predefined criteria and determine if it meets objectives and fulfills user needs. The purpose of acceptance testing is to ensure that the software is acceptable and ready for deployment based on customer or end-user requirements and expectations.

Example:

1. User Interface: Acceptance testing would involve verifying that the user interface of the Institute Management system is user-friendly, intuitive, and meets the expectations of the users. Testers would assess the layout, navigation, and overall usability of the system, ensuring that it is easy to understand and operate.

2. Functional Requirements: Testers would execute tests to validate that the system performs the necessary functions as specified in the requirements. This could include creating test scenarios to test various features such as user registration, course management, student enrollment, attendance tracking, and generating reports. The goal is to ensure that the system fulfills the functional requirements and operates smoothly.

5.1.4 User Evaluation

Table 2 Evaluation Table

No.	Task	Satisfied?					
		Admin		Teacher		Student	
		Yes	No	Yes	No	Yes	No
01.	Authentication	●		●		●	
02.	Data Insertion	●		●		●	
03.	Data Deletion	●		●		●	
04.	Data Updating	●		●		●	
05.	View User Details	●			●	●	
06.	View Student entered Details (Appointment, Feedback, Ticket)	●		●		●	
07.	View Module Page, Quiz, Assignment Details	●		●		●	
08.	View Employee Salary Details	●		●			●
09.	View Notices & Timetable	●		●		●	
10.	View Class Fee Details	●			●		●
11.	Generate Reports	●		●			●
12.	View Reports	●		●			●

5.2 Conclusion

In conclusion, the Institute Management System is a valuable tool that enhances efficiency, organization, and communication within educational institutions. It streamlines administrative tasks, automates processes, and provides real-time access to information, resulting in improved decision-making and overall effectiveness. With its user-friendly interface and comprehensive features, the Institute Management System simplifies day-to-day operations, promotes collaboration among Teacher, students, and ultimately contributes to the success and growth of the institute. By effectively managing various aspects such as student records, Module Page, Assignments, Quizzes, Raise Tickets, Feedbacks, timetables, Notices, and finance, the system enables administrators to allocate resources efficiently and focus on delivering quality education. Additionally, the system facilitates communication between stakeholders, ensuring seamless interaction and timely dissemination of information. Overall, the Institute Management System proves to be an essential tool for modern educational institutions seeking to streamline processes, optimize resources, and provide a superior learning experience for students.

Chapter 6: References

References

- [1] M. O'Connor, "synergy-learning," [Online]. Available: <https://synergy-learning.com/blog/7-top-benefits-of-using-a-learning-management-system-lms/>.
- [2] A. MAHMOOD, "Pros and Cons of Website Redesign," [Online]. Available: <https://designmodo.com/redesign-website/>.
- [3] Maciej, "Pros and cons of using ready-made themes for website development," [Online]. Available: <https://www.merixstudio.com/blog/pros-and-cons-of-using-ready-made-themes-for-website-development/>.
- [4] MongoDB. [Online]. Available: <https://www.mongodb.com/mern-stack...>
- [5] JAVATPOINT, "UML-TOOLS," [Online]. Available: <https://www.javatpoint.com/uml-tools>.
- [6] M. REHKOPF, ATLASSIAN, [Online]. Available: <https://www.atlassian.com/agile/kanban/boards>.
- [7] E. Aceska, "Types of Black Box Testing Techniques," TestDevLab, [Online]. Available: <https://www.testdevlab.com/blog/types-of-black-box-testing-techniques>.
- [8] "tray.io," [Online]. Available: <https://tray.io/blog/what-is-an-api-integration-for-non-technical-people>.
- [9] A. J. o. Evaluation. [Online]. Available: [https://journals.sagepub.com/doi/pdf/10.1177/1098214020920710#:~:text=Stufflebeam%20\(1973%20p.,the%20product%20of%20that%20process..](https://journals.sagepub.com/doi/pdf/10.1177/1098214020920710#:~:text=Stufflebeam%20(1973%20p.,the%20product%20of%20that%20process..)
- [1] Perfecto. [Online]. Available: <https://www.perfecto.io/resources/types-of-testing>.

Chapter 7: Appendices

7.1 Appendix

Kanban Board

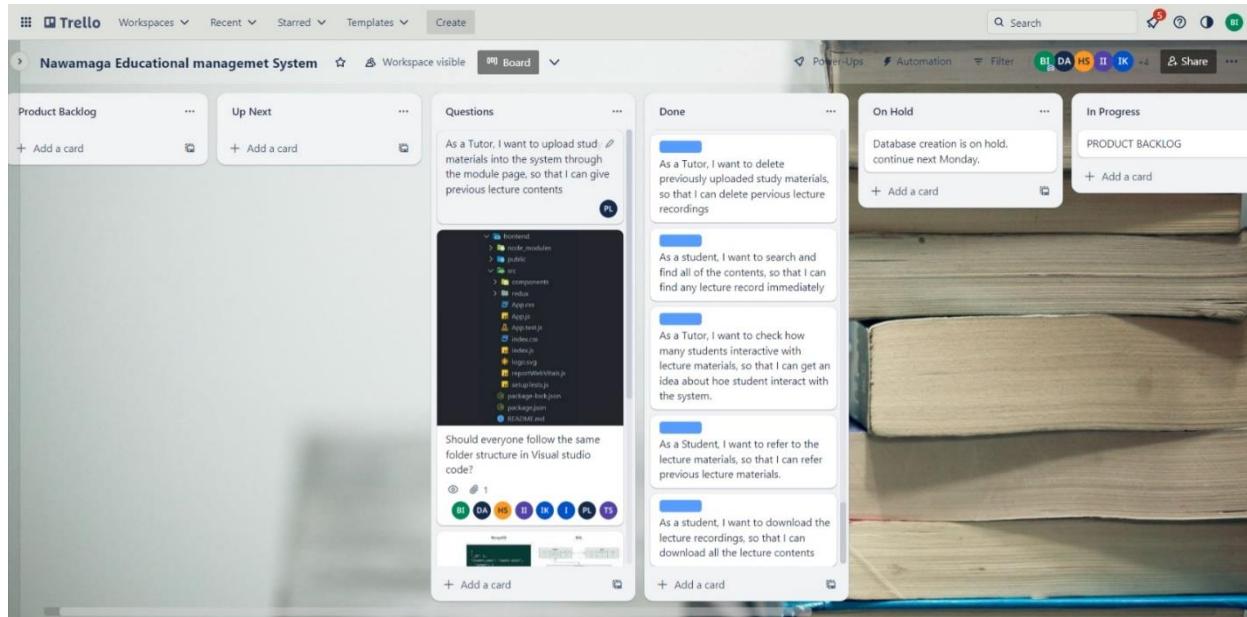


Figure 0:1Kanban Board

7.2 Appendix

Gantt Chart

Task	Feb	Mar	Apr	May
Finding a client	█			
Gather requirement	█			
Requirement analysis	█			
Functionalities		█		
Charter approval			█	
Scrum activity				
Proposal presentation				
UI sketch			█	
Conceptual database				█
Physical database			█	
Implementation				█████
System test				█
Final report				█

