
Software Requirements Specification

for

Course Registration and Time Table Generation System

Prepared by:

Project Manager- Pranjal Jain (21UCS155)

Pranjal Bansal (21UCS154)

Pranjal Mishra (21UCS156)

Vaibhav Khamesra (21UCS224)

Vaibhav Rai (21UCS225)

The LNM Institute of Information Technology

Date: October 24, 2023

Table of Contents

Table of Contents

1.	Introduction	
1.1	Purpose	4
1.2	Document Conventions	4
1.3	Intended Audience and Reading Suggestions	4
1.4	Product Scope	4
1.5	References	5
2.	Overall Description	
2.1	Product Perspective	6
2.2	Product Functions	6
2.3	User Classes and Characteristics	7
2.4	Operating Environment	8
2.5	Design and Implementation Constraints	8
2.6	User Documentation	8
2.7	Assumptions and Dependencies	8
3.	External Interface Requirements	
3.1	User Interfaces	9
3.2	Hardware Interfaces	14
3.3	Software Interfaces	15
3.4	Communications Interfaces	15

4. System Features	
4.1 Course Registration	16
4.2 Admin Course Management	16
4.3 Timetable Generation	17
4.4 Email Notification	18
5. Other Nonfunctional Requirements	
5.1 Performance Requirements	19
5.2 Safety Requirements	19
5.3 Security Requirements	20
5.4 Software Quality Attributes	20
5.5 Business Rules	20
6. Other Requirements	
6.1 Database Requirements	21
6.2 Legal Requirements	21
6.3 Reuse Requirements	21
Appendix A: Glossary	22
Appendix B: Analysis models	23
Appendix C: To Be Determined List	33

1. Introduction

1.1 Purpose

The project aim is to implement an online course registration system in place of manual course registration for university students with a user-friendly interface and in a cost-effective manner, and to develop a timetable generator that can schedule classes across all batches and courses.

1.2 Document Convention

The document is made using the Calibri font, where:

- Headings are represented in bold letters with a font size of 25
- Sub-headings are represented as underlined and in bold letters with font size 13
- Text is represented with a font size of 11.
- The font used in the entire document is *Verdana*.

Appendix A is added as a glossary, which consists of abbreviations and acronyms.

1.3 Intended Audience and Reading Suggestions

The SRS document is a reference document for anyone interested in software requirements for web-based systems for Course Registration and Timetable generation but should be of special interest to designers, project managers, users, development teams, and organizational staff

1.4 Product Scope

The scope of this project is to present a detailed description of the Online Course Registration System and a timetable generator. Students will use the online registration system. The application of the software must be user-friendly for both our students and administrators. The Course Registration System will enable only those students who have paid fees to register for courses online. Each student may indicate program electives (one or more) according to their semester. It allows

faculty members to choose their teaching courses and to maintain student grades. The course registration system will help students gather information about a particular course. Then they can easily register themselves for that course. The management of the institution can see the records of the students, courses, and fees.

The timetable generation system would be used by the academic administration to automate the intricate process of creating lecture schedules while also ensuring that proper breaks are given to the students and lecturers and there are no clashes.

1.5 Reference

- Lucid Chart for creating and analyzing different diagrams
- Software Engineering by Ian Sommerville, Pearson Education LPE, 8th or 9th Edition:- To analyze advantages and disadvantages of various requirements (like database downtime, etc.)
- MongoDB - For establishing a database
- Express.js - For API calls and backend
- React.js - Frontend
- Node.js - Backend

2. Overall Description

2.1 Product Perspective

The Online Course Registration System and Time Table Generator is a self-contained software product that will be developed to overcome the problems that have occurred due to the manual process. This system will provide easy access to the registration process, and it will consist of user-friendly functions with attractive interfaces. The system will give better options for the problem of handling large-scale physical file systems and thus reduce errors that occur manually. The outcome of this project will increase the cost-effectiveness of almost all the tasks related to the registration process at the college level in a much more convenient manner. The timetable generator would take course details, faculty details, and details of available lecture halls along with the time of their availability to schedule the lectures.

2.2 Product Functions

The key features of this system can be abstracted as follows:

- Authentication will be done through username and password for both admins and students separately.
- The administrator can choose from three options add/ delete/ modify courses available; initiate the registration process; and generate a timetable based on the added courses.
- Admin will add details regarding courses available for registration and register students in their respective core courses.
- Students will verify their details, choose whether to drop or add a course, and will get a confirmation slip/ course registration slip as an acknowledgment.
- Proof will be provided in the form of a registration report once the process has been submitted and terminated.

- The course instructor, with her/his credentials, can access the list and details of registered students.
- The Timetable generator will generate a timetable based on the data received through the process of Course registration, i.e., the number of students registered per course and the availability of lecture halls and lecturers

2.3 User Classes and Characteristics

There are two user levels in the Course Registration System: -

2.3.1 Student

- The user can access the online course registration system.
- Since the admin will already register users, they will have an account.
- The user enters a username and password to access the course registration process.
- The user has the functionality to change their password.
- The user views the respective course list as per their branch.
- Users choose from the list of available program electives.
- Submits registration and receives proof of registration

2.3.2 System Admin

- The administrator will create an account for the admitted students at the university and assign their username as the student roll number and a random password.
- Responsible for adding and changing the courses in the course catalog for the respective department and semester.
- Admin registers students in their core courses and provides them with the option to choose program electives.
- The admin would also have the control to generate a timetable but it would be done by first validating the course registrations and then, following which, a timetable would be automatically generated by the most optimal scheduling algorithms.

2.4 Operating Environment

Any device that can access the high-quality internet and support web applications. This system can run on any kind of OS, as it is not particularly dependent on the software.

2.5 Design and Implementation Constraints

- Using this system is fairly simple and intuitive. A user familiar with basic browser navigation skills should be able to understand all the functionality provided by the system.
- The system should work on most home desktop and laptop computers that support JavaScript and HTML5.

2.6 User Documentation

User documentation will consist of the several components usually expected of a modern web-based software application, including a tutorial, help pages, and a complete user manual.

2.7 Assumptions and Dependencies

The assumptions and dependencies relevant to the system are as follows.

- All users should have a computer or any device that has access to a reliable network.
- Users should have been registered with the university and should have a student roll number.
- Users should read the course catalog carefully in order not to make mistakes, cause they are un-reversible.

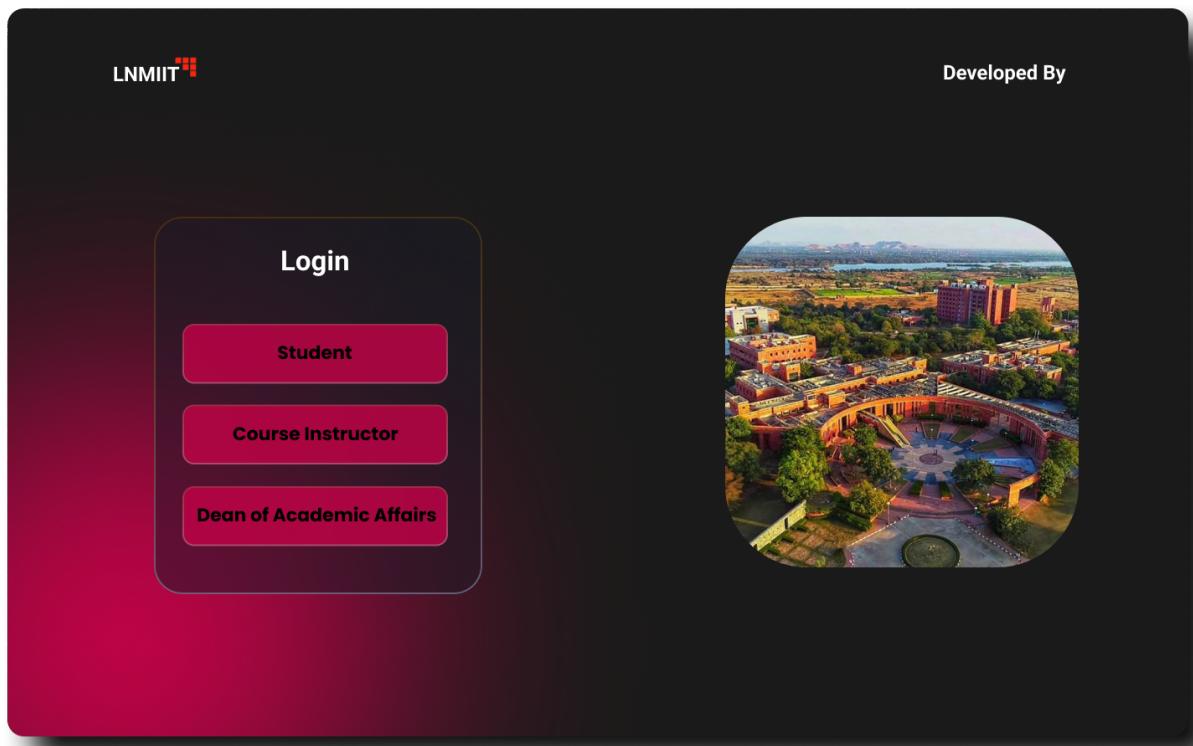
3. External Interface Requirements

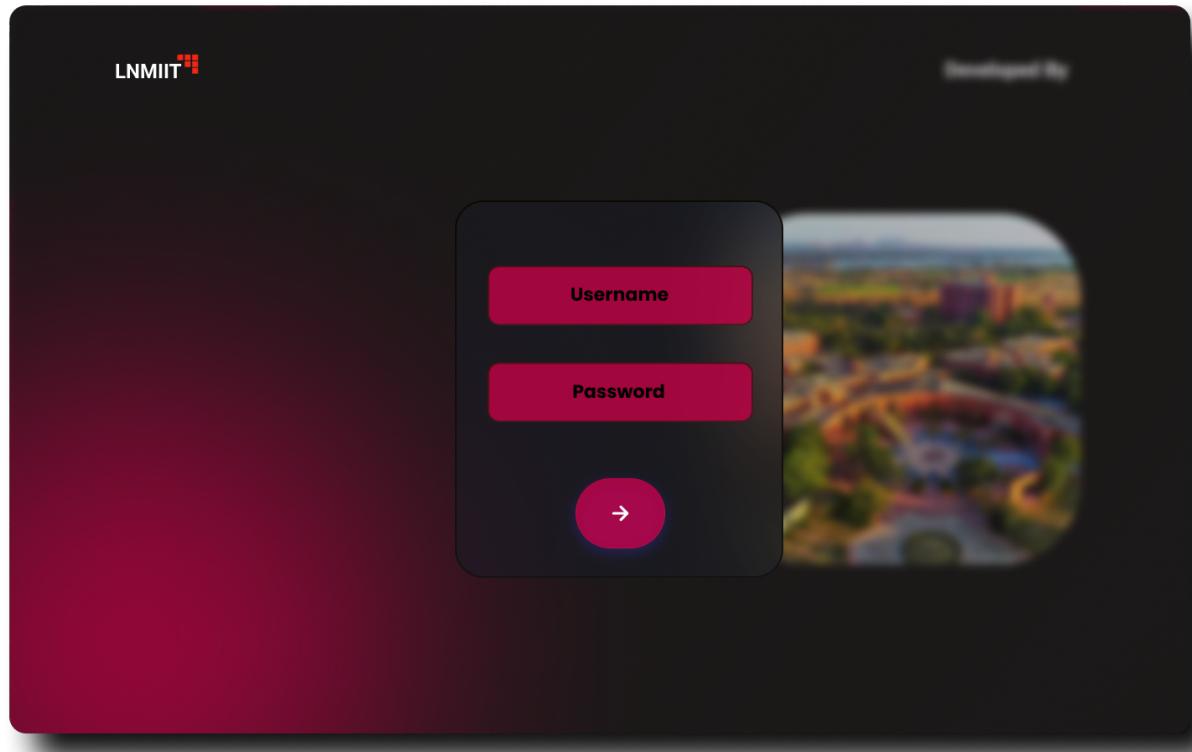
3.1 User Interfaces

The user interface of the course registration and timetable generation web application is designed to be intuitive, user-friendly, and visually appealing. Below, we describe the key interfaces and provide details on the images to be included.

3.1.1 Login Page

The login page serves as the entry point for users, requiring them to provide valid credentials to access the system. It features a simple and secure login form with fields for username and password. Users will also find a "Forgot Password" option for password recovery.





3.1.2 Student Course Details

The student course details page offers a comprehensive view of a student's enrolled courses. It presents detailed information about each course, including course name, instructor, schedule, and location. Students can easily access this information to plan their academic schedules effectively.

Student Details



Course Details



DSA
ABCD

Credits:- 4



DSA
ABCD

Credits:- 4



DSA
ABCD

Credits:- 4



DSA
ABCD

Credits:- 4



DSA
ABCD

Credits:- 4



DSA
ABCD

Credits:- 4



DSA
ABCD

Credits:- 4

3.1.3 Courses Offered

The "Courses Offered" interface allows students to explore available courses. It presents a list of courses, searchable by various criteria such as department, course code, and availability. This intuitive interface simplifies the course selection process.

The screenshot displays a dark-themed user interface for course selection. At the top left is the LNMIIT logo. The main title "Courses Offered" is centered above two sections: "Core Courses" and "Program Electives".

Core Courses:

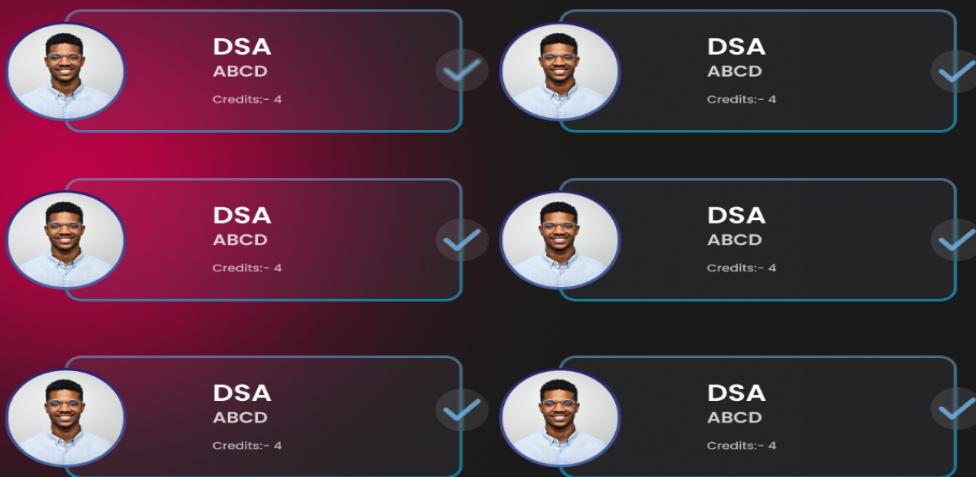
- SWE** Faculty 1 Credits:- 4
- DSA** Faculty 2 Credits:- 4
- PTS** Faculty 3 Credits:- 4
- ISDL** Faculty 4 Credits:- 4
- EFE** Faculty 5 Credits:- 4
- COA** Faculty 6 Credits:- 4

Program Electives:

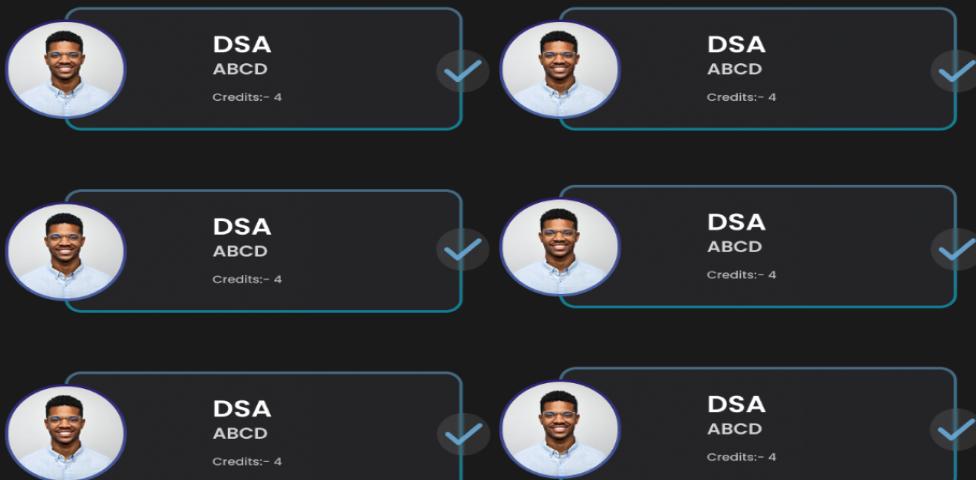
- PAVFM** Faculty 10 Credits:- 4
- DL** Faculty 13 Credits:- 4
- SC** Faculty 20 Credits:- 4
- NLP** Faculty 18 Credits:- 4
- DIP** Faculty 8 Credits:- 4
- DPIOT** Faculty 5 Credits:- 4

Courses Offered

Core Courses



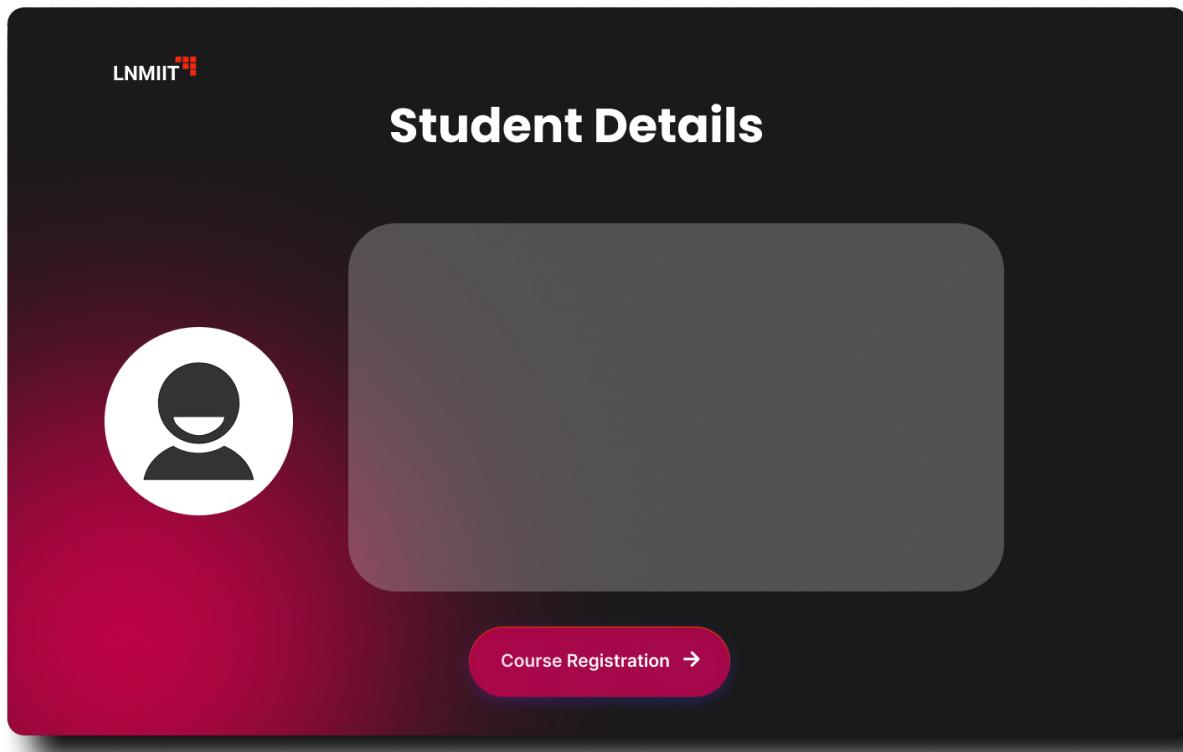
Program Electives



Confirm →

3.1.4 Student Profile

The student profile page enables users to manage their personal information, including contact details, academic history, and account settings. It ensures that students can maintain accurate and up-to-date profiles.



The user interface design adheres to modern web design principles, with a responsive layout for optimal usability on various devices, including desktops, tablets, and mobile phones. The visuals are carefully crafted to enhance the user experience and facilitate easy navigation.

3.2 Hardware Interfaces

The software will interact with the hardware components of the system in the following ways:

- **Supported Devices:** The software will be accessible through standard web browsers on various hardware devices, including desktop computers, laptops, tablets, and smartphones.

- Data and Control Interaction: Data transfer and control interactions between the software and the hardware will be through standard HTTP and HTTPS protocols.
- Communication Protocols: The software will use HTTP/HTTPS for communication between the client and server.

3.3 Software Interfaces

The software will interface with the following software components and services:

- Database: The system will use a MongoDB database to store course information, user profiles, and scheduling data. Data will be accessed using MongoDB drivers.
- Server Framework: The software will utilize Node.js with Express.js as the server framework for handling API requests.
- Web Client: The user interface will be developed using React for a responsive and interactive web experience.
- External Services: The system will integrate with email services for sending notifications and confirmations. Email services will be accessed via standard email protocols (e.g., SMTP).

3.4 Communications Interfaces

The software will have the following communication requirements:

- Email Communications: The system will use SMTP for sending email notifications to users. Email content will follow industry-standard formatting.
- Web Communications: Communication between the client and server will occur over secure HTTPS connections.
- Data Transfer Rates: The software will ensure efficient data transfer to provide a responsive user experience.
- Security: Communication security will be implemented using HTTPS for data encryption, and email communications will follow industry security standards.

4. System Features

4.1 Course Registration

4.1.1 Description and Priority

- This feature allows users to register for courses and create a time-table. It is of high priority as it is a core functionality of the system.

4.1.2 Stimulus/Response Sequences

- The user logs in or registers.
- The user selects courses and preferences.
- The system validates course availability, prerequisites, and scheduling constraints.
- The user submits the registration request.
- The system generates and displays the timetable.
- The system sends a confirmation message to the user.

4.1.3 Functional Requirements

- REQ-1: The system shall provide user authentication and registration mechanisms.
- REQ-2: Users shall be able to select courses from the available catalog.
- REQ-3: The system shall validate course prerequisites, capacity limits, and scheduling conflicts.
- REQ-4: Users should be able to submit their course registration requests.
- REQ-5: The system shall generate a timetable for the user based on their course selections.

4.2 Admin Course Management

4.2.1 Description and Priority

- This feature allows administrators to manage course offerings. It is of high priority.

4.2.2 Stimulus/Response Sequences

- The administrator logs into the system.

- The administrator accesses the admin dashboard.
- The system provides options for managing courses.
- The administrator adds, modifies, or removes courses.
- The system updates the course catalog.

4.2.3 Functional Requirements

- REQ-6: Administrators shall have privileged access to the admin dashboard.
- REQ-7: The system shall allow administrators to add, modify, or remove courses from the catalog.
- REQ-8: Changes to the course catalog shall be immediately reflected in the user registration process.

4.3 Timetable Generation

4.3.1 Description and Priority

- This feature allows administrators to generate timetables for registered students. It is of high priority.

4.3.2 Stimulus/Response Sequences

- The administrator logs into the system.
- The administrator accesses the admin dashboard.
- The system provides an option for timetable generation.
- The system generates and displays the timetable.

4.3.3 Functional Requirements

- REQ-9: Administrators shall have a function to generate timetables for registered students.
- REQ-10: The system shall validate scheduling constraints and conflicts during timetable generation.
- REQ-11: The generated timetable shall be displayed in a clear and user-friendly format.
- REQ-12: Timetables shall be saved and associated with the respective student profiles.

4.4 Email Notifications

4.4.1 Description and Priority

- This feature allows the system to send email notifications to users for important updates or changes to their registrations. It is of medium priority.

4.4.2 Stimulus/Response Sequences

- The system detects important updates (e.g., course changes and timetable modifications).
- The system sends email notifications to affected users.

4.4.3 Functional Requirements

- REQ-13: The system shall have a function to send email notifications to users.
- REQ-14: Email notifications shall include details of the update and instructions for the user.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The performance requirements for the course registration and timetable generation software are as follows:

- Response Time: The system shall respond to user actions (e.g., course registration, timetable generation) within 2 seconds in normal operational conditions to ensure a smooth and responsive user experience.
- Concurrent Users: The system should support a minimum of 1000 concurrent users during peak registration periods without significant degradation in performance.
- Timetable Generation: The system shall generate timetables for students within 10 seconds for a typical course schedule.
- Historical Data: The system should maintain a history of registrations and schedules without affecting system performance.

Rationale: These performance requirements ensure that the system can handle a significant number of users efficiently and provide a seamless experience during peak usage times.

5.2 Safety Requirements

The safety requirements for the software are as follows:

- Data Backup: The system shall perform daily automated data backups to prevent data loss.
- Error Handling: The software should provide robust error-handling mechanisms to prevent critical system failures.
- Security Measures: The software shall implement security features to protect user data and prevent unauthorized access or breaches.

External Policies: Compliance with data protection regulations and industry best practices

5.3 Security Requirements

The security requirements for the software are as follows:

- User Authentication: Users shall be required to authenticate using secure methods, such as username and password, for access.
- Data Encryption: User data shall be encrypted during transmission (HTTPS) and storage to ensure data privacy.
- Authorization: Users shall only have access to data and functionality that they are authorized to use based on their roles and permissions.
- Intrusion Detection: The system shall have an intrusion detection system (IDS) to monitor and respond to security breaches.

External Policies: Adherence to data protection regulations and security best practices.

5.4 Software Quality Attributes

The software quality attributes for the product are:

- Usability: The software should be user-friendly, with clear and intuitive navigation.
- Reliability: The system shall operate without unexpected interruptions and provide accurate data.
- Maintainability: The software code shall follow best practices and be well-documented for easy maintenance and future enhancements.
- Flexibility: The software shall be designed to accommodate changes or updates in requirements with minimal impact on the overall system.
- Availability: The system should be available for use during specified uptime hours, ensuring that users have access when needed.

Relative Preferences: Usability is preferred over ease of learning, while maintainability is important for long-term sustainability.

5.5 Business Rules

The business rules for the product include:

- Administrator Privileges: Only administrators can generate timetables, manage course offerings, and modify student registrations.
- Registration Rules: The system shall enforce university policies on course prerequisites, capacity limits, and scheduling constraints during registration.

6. Other Requirements

6.1 Database Requirements

- The software shall use a MongoDB database for storing course information, user profiles, and scheduling data.
- Database backups shall be taken daily and stored for a defined retention period.
- Data retention policies for student records and historical data shall be defined and implemented.

6.2 Legal Requirements

- The software shall comply with all relevant data protection regulations, such as GDPR or HIPAA, depending on the geographical location and user data processed.
- Privacy policies and terms of use shall be provided to users, and they must agree to these terms before using the application.

6.3 Reuse Objectives

- The development team shall explore opportunities for reusing existing code libraries and components to enhance efficiency and reduce development effort.
- Code reusability shall be encouraged, and a reusable code repository shall be maintained for future projects.

Appendix A: Glossary

Definitions of terms, acronyms, and abbreviations used in the SRS:

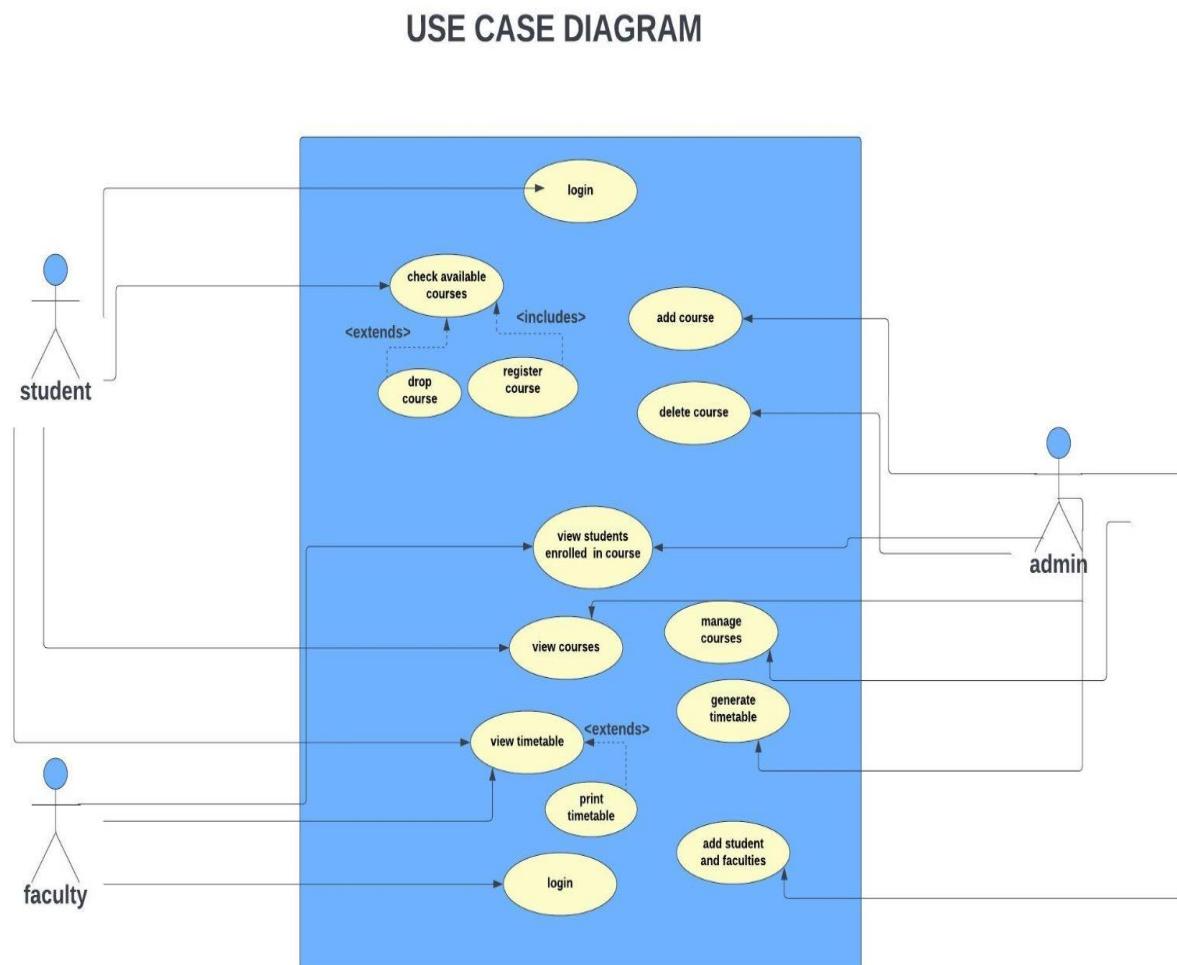
- SRS: Software Requirements Specification
- IDS: Intrusion Detection System
- HTTPS: Hypertext Transfer Protocol Secure
- REQ: Requirement
- SMTP: Simple Mail Transfer Protocol
- API: Application Programming Interface
- HTTP: Hypertext Transfer Protocol
- OS: Operating System

Appendix B: Analysis Models

In this section, we present a summary of various analysis models that have been developed to support the design and understanding of the course registration and timetable generation web application.

1 Use Case Diagram

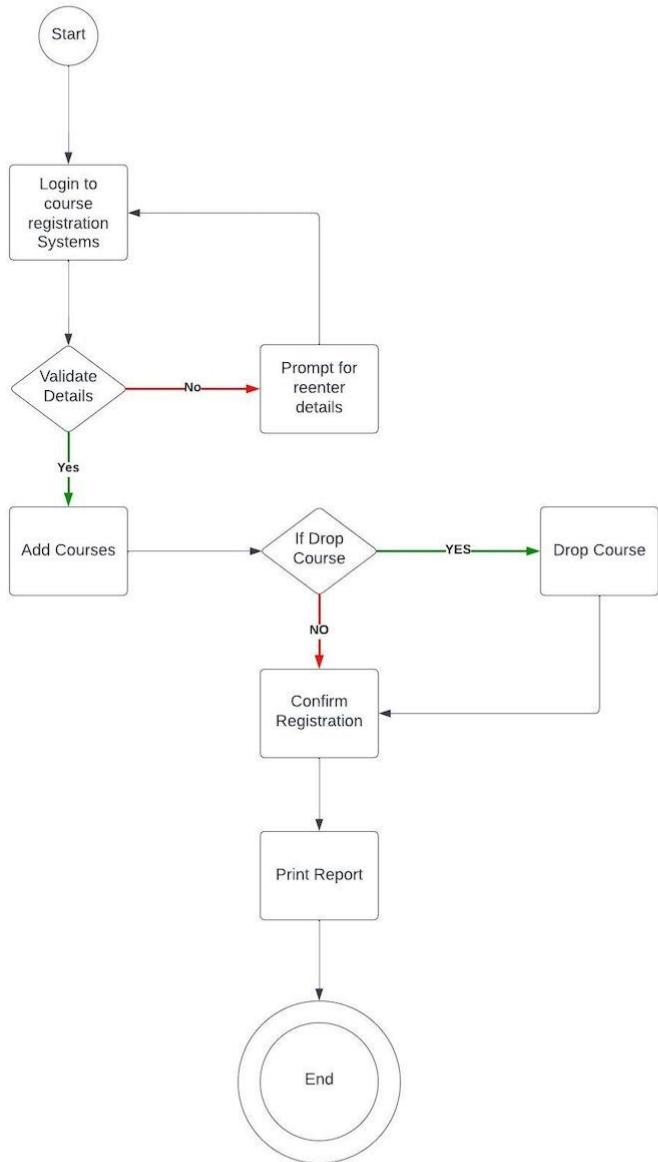
The use case diagram provides a high-level representation of user interactions with the system. It outlines the primary functionalities and features of the application, illustrating how users interact with the system and the main use cases.



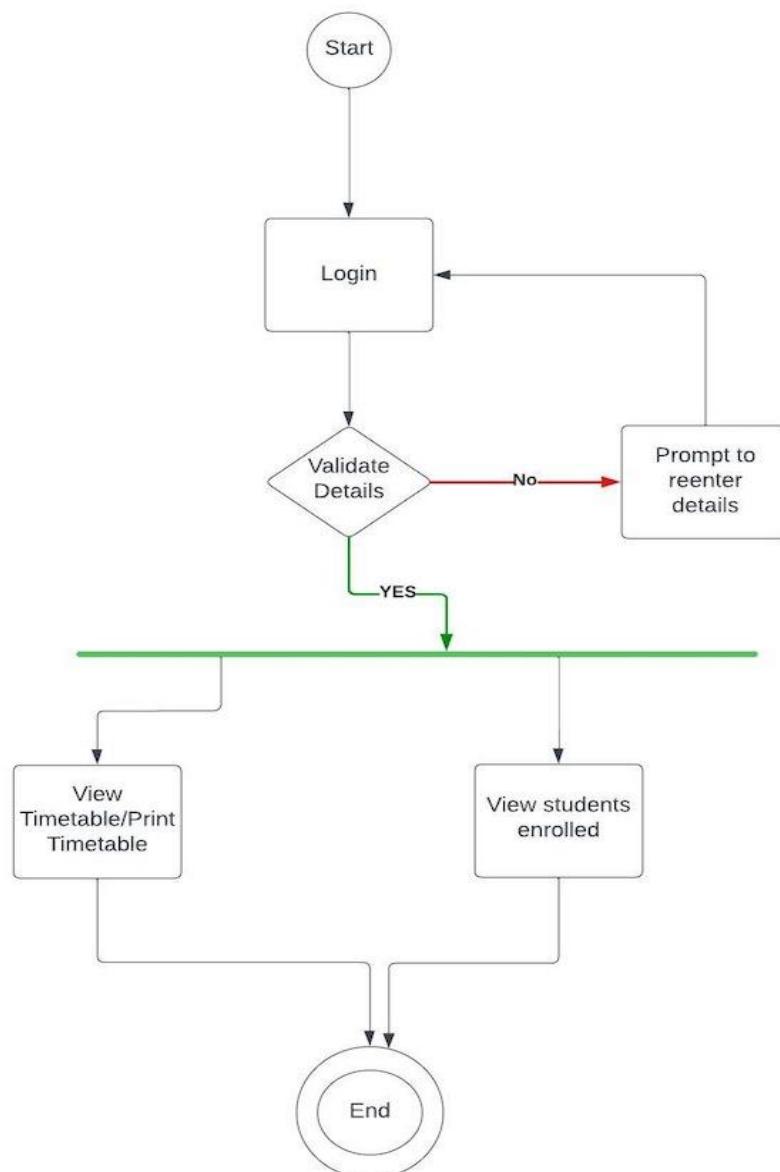
2 Use Case Activity Diagram (UCAD)

UCAD, or Use Case Activity Diagram, is a comprehensive design model that consolidates various design and drafting tools into a unified platform. This model is instrumental in architectural design, enabling the creation of detailed plans and schematics.

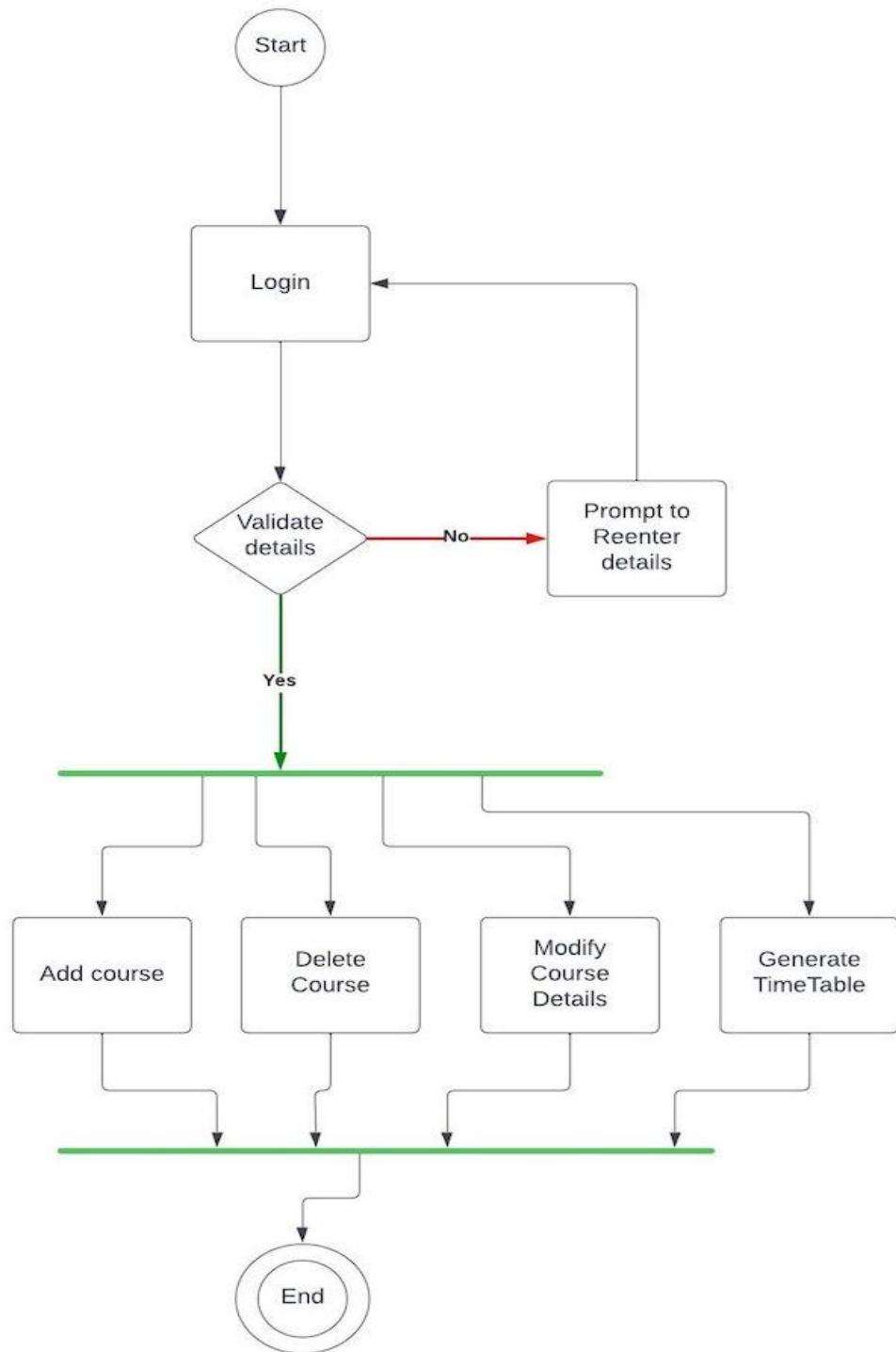
UCAD Student



UCAD Course Instructor

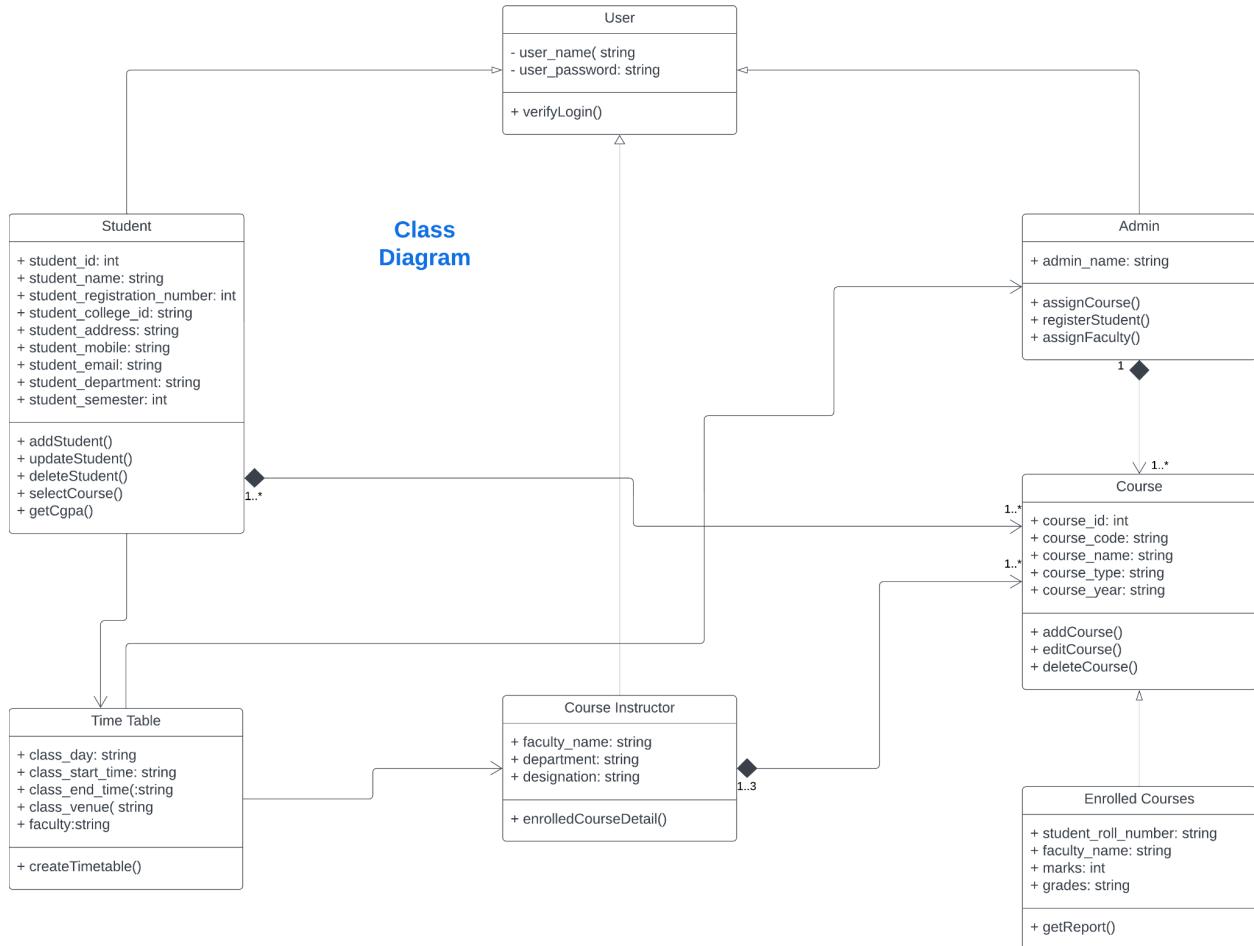


UCAD Admin



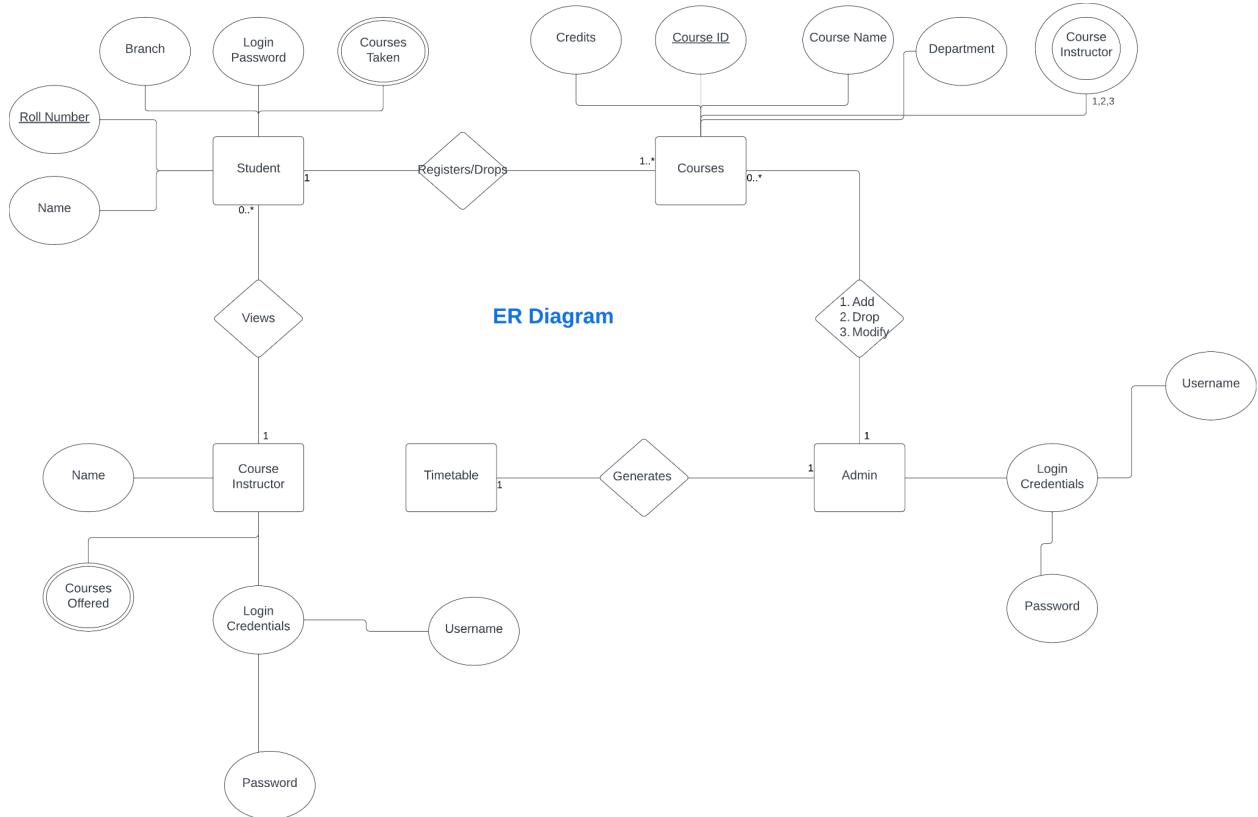
3 Class Diagram

The class diagram offers insights into the system's structure by depicting classes, attributes, and their relationships. It is crucial for understanding the object-oriented design of the application and the organization of classes.



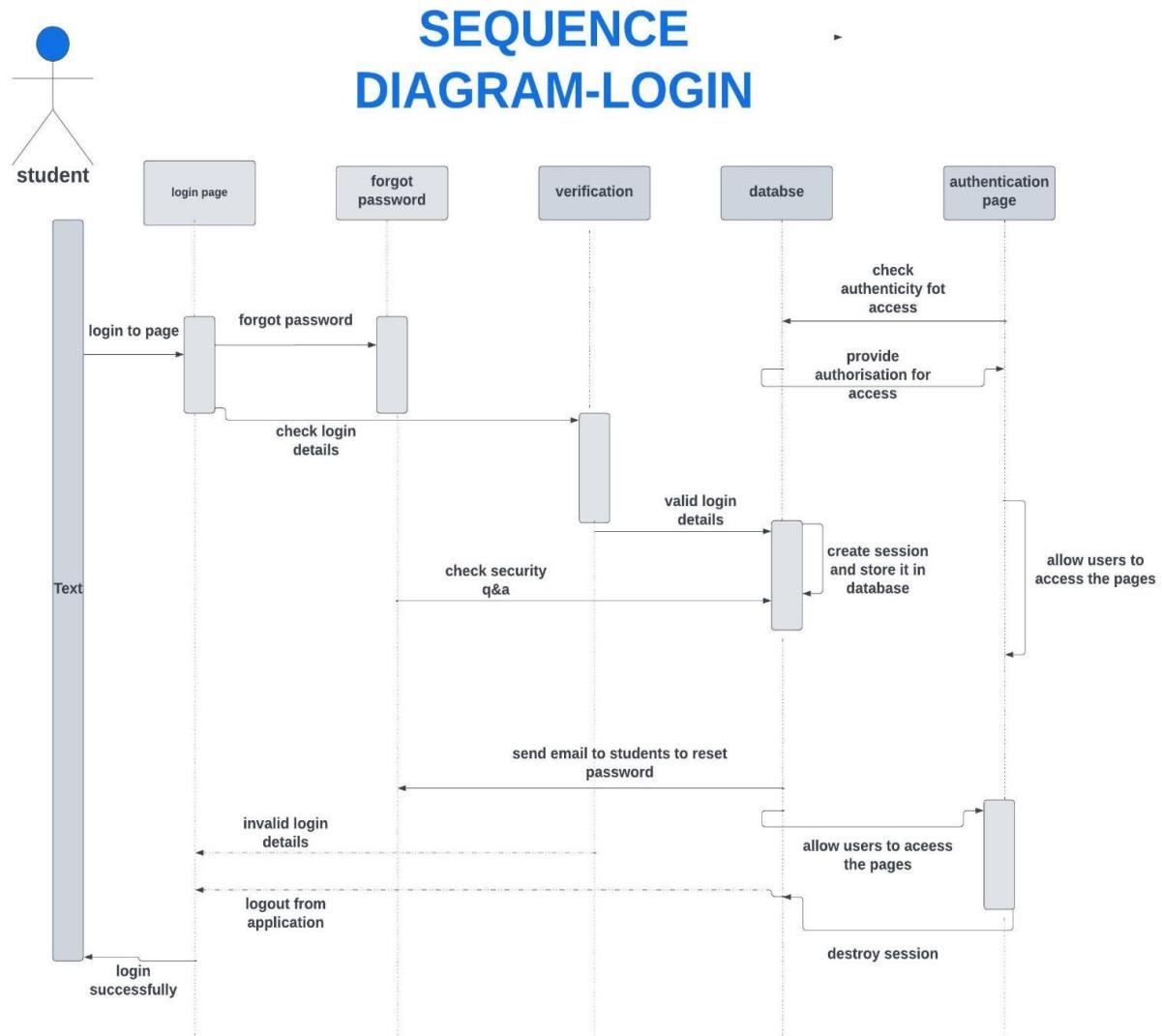
4 Entity-Relationship (ER) Diagram

The ER diagram illustrates the database structure, showcasing entities, their attributes, and the relationships between them. It plays a pivotal role in designing the database schema to ensure efficient data management.

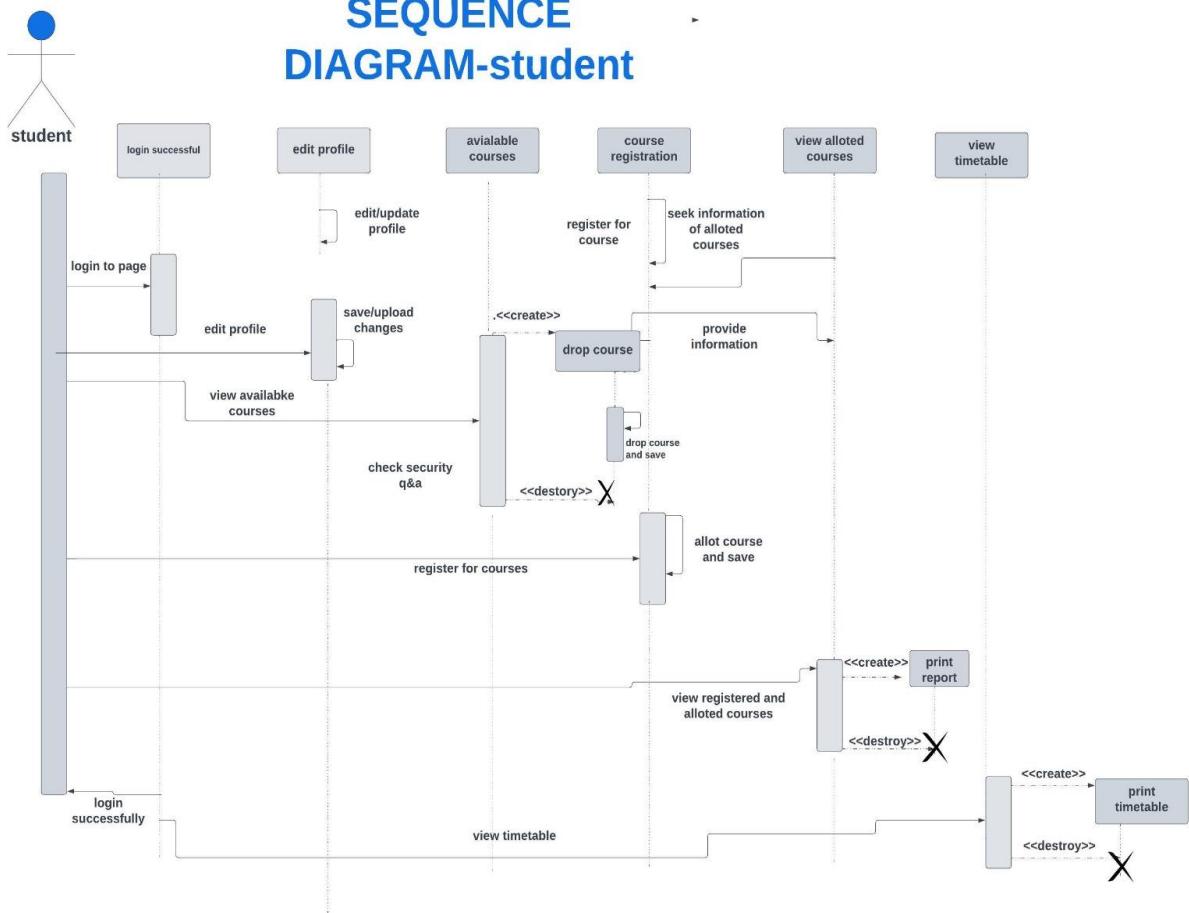


5 Sequence Diagrams

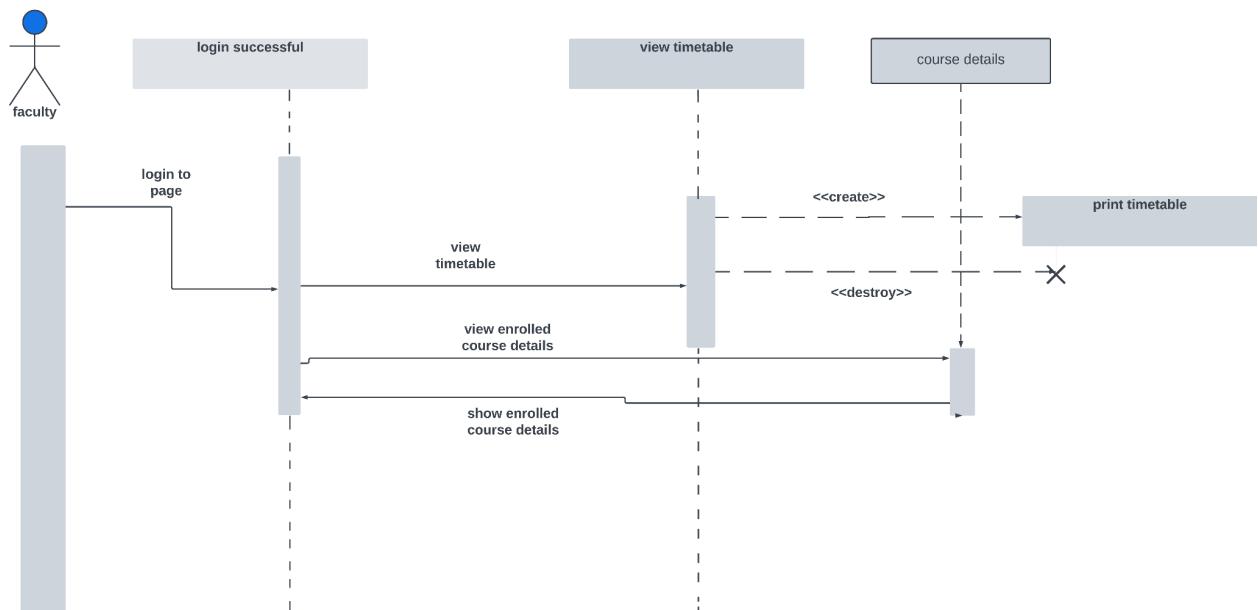
The sequence diagram provides a dynamic view of the system by illustrating interactions and message sequences between different objects or components. It helps in comprehending the flow of operations and the sequence of events within the application.

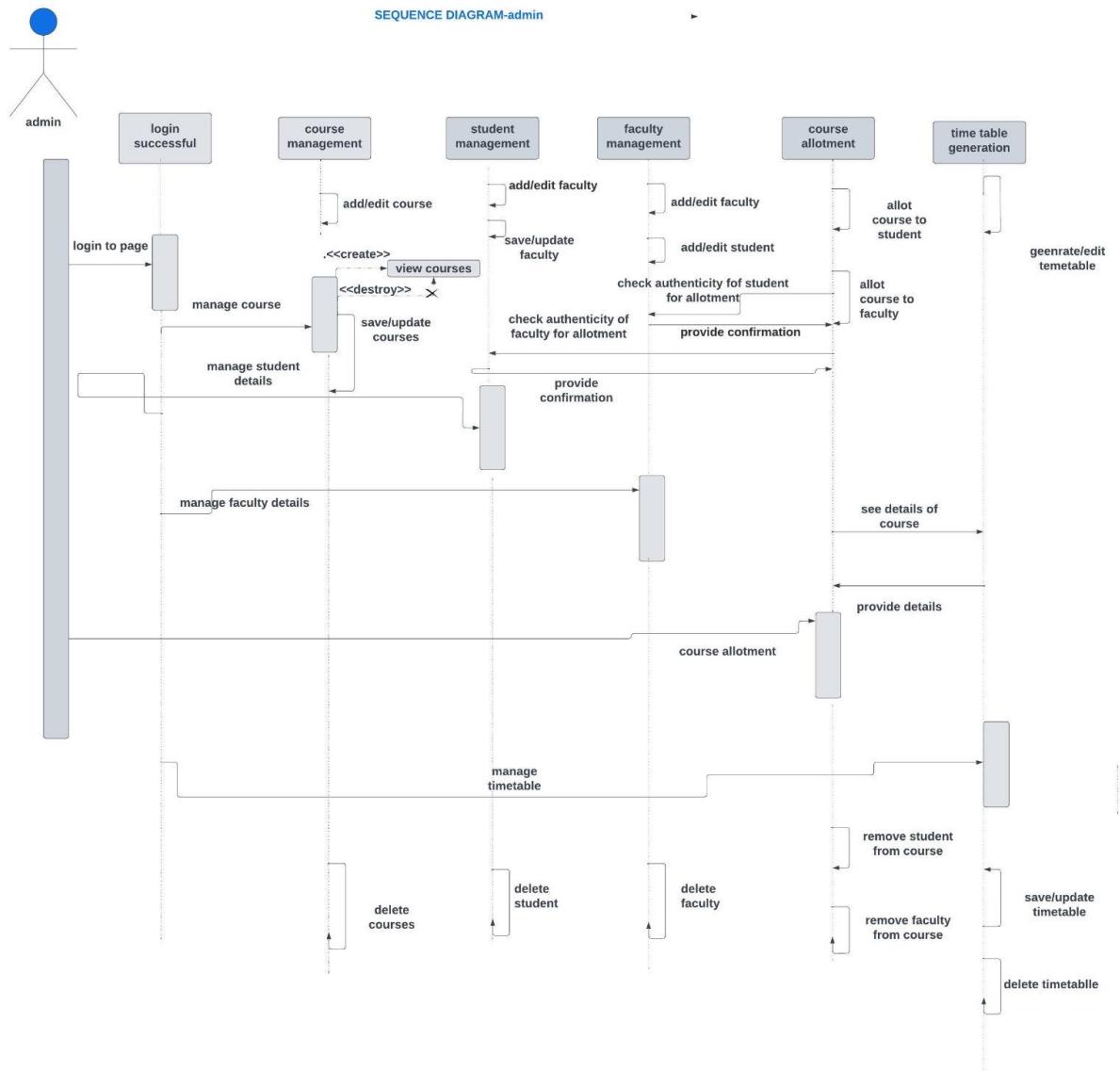


SEQUENCE DIAGRAM-student



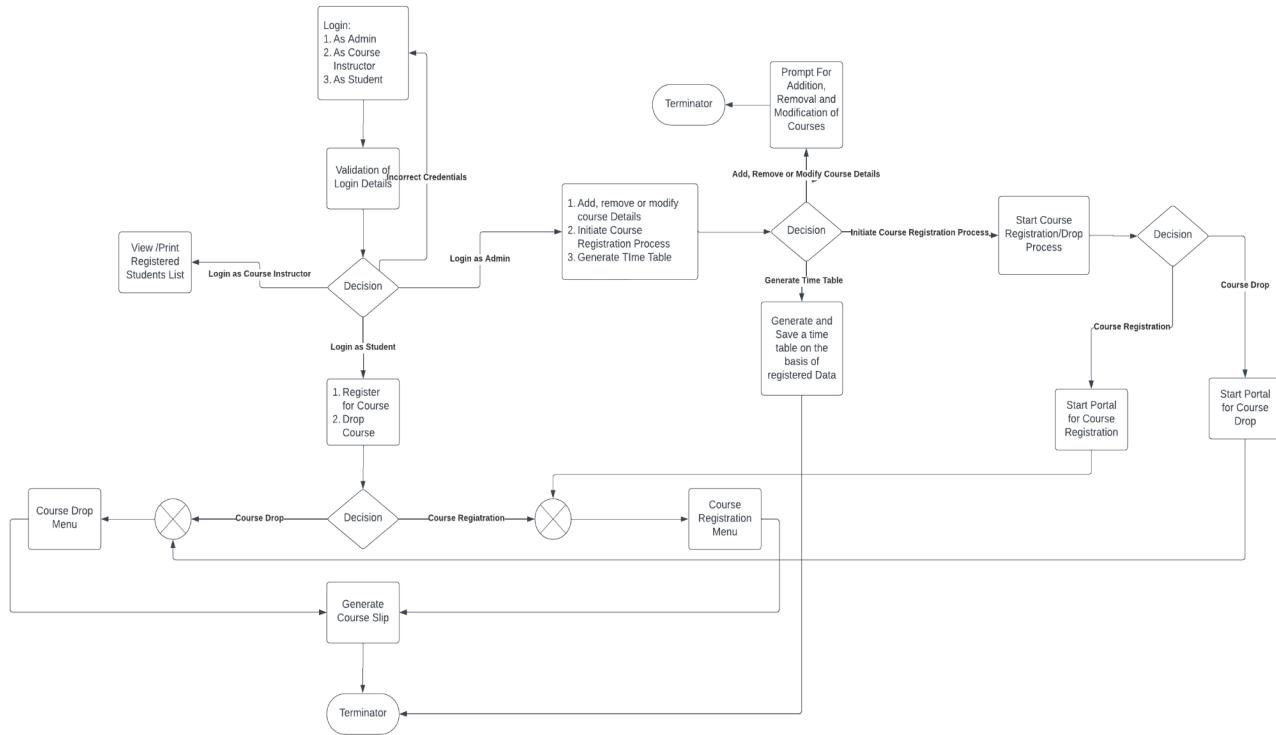
SEQUENCE DIAGRAM-Course Instructor





6 Business Process Model and Notation (BPMN)

BPMN is a standardized graphical notation for modeling business processes. It offers a visual representation of the application's workflow, detailing the sequence of activities, decisions, and interactions within the system. It aids in analyzing process flows and optimizing business processes.



These analysis models collectively contribute to a comprehensive understanding of the course registration and timetable generation web applications. They provide insights into the system's functionality, structure, data management, and business processes.

Appendix C: To Be Determined List

- Timetable Generation Algorithm
 - The algorithm or methodology to be used for automatic timetable generation
- Mobile Responsiveness
 - The level of mobile responsiveness and specific design considerations for mobile devices
- Testing and Quality Assurance Plan
 - Detailed testing methodologies, test cases, and quality assurance procedures
- Admin User Interface
 - The specific user interface design and functionality allow administrators to manage the system.