



Department of Computer Science
Barani Institute of Management Sciences,
Rawalpindi Pakistan
BIMS Management System(BMS)

By

Sajid Inayat	18-ARID-5264
M. Najam Sadiq	19-ARID-437
Aqsa Imran	20-ARID-317

Supervisor:

Mr. Muhammad Hassan

Bachelor of Science in Computer Science (2020-2024)

DECLARATION

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software documentation and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other. We will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

Aqsa Imran

Muhammad Najam Sadiq

Sajid Inayat

CERTIFICATE OF APPROVAL

It is to certify that the final year project of BS (CS) “**BIMS Management System**” was developed by “**Aqsa Imran, 20-arid-317**” and “**Muhammad Najam Sadiq, 19-arid-437**” “**Sajid Inayat, 18-arid-5264**” under the supervision of “**Mr Muhammad Hassan**” and that in their opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Computer Science.

Supervisor

Administrator BIMS

Executive Summary

In public places, there is often a need for monitoring people and different activities going on, which can be referred later for many reasons including security. Appointing humans for this task involves many problems such as increased employee hiring, accuracy problem, trust, no proof for later use, and also the fact that a human can remember things till a certain time limit. Talking about the current security system, they use dumb still cameras with a continuous recording facility irrespective of the fact that any event may happen or not. Moreover, they are usually pointing at a specific user defined locations so more than one cameras are required to cover the entire region.

To prevent all these problems from prevailing, the CSCS is developed. It is a surveillance system, which provides solution to many of these problems. It is a stand-alone application which doesn't require any computer to operate. It monitors different situations using a camera which is able to rotate intelligently based on sensor messages and captures the scene in the form of video or photos later reference as well.

Customizable Surveillance Control System (CSCS) is a surveillance system that can be assigned a sensor type as in our case a heat sensor is used, it works accordingly, rotates the camera upon event detection and perform user defined actions like capturing video and stores them, for the future use.

It is an embedded system consisting of Linux fox kit with embedded a running server application also a camera, USB storage device and a sensor node base station is attached with fox kit. LAN communication is used by user to download the videos and to operate the system manually.

Acknowledgement

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor “Mr. Muhammad Hassan” for personal supervision, advice, valuable guidance and completion of this project. We are deeply indebted to **them** for encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us the values of honesty & hard work.

Aqsa Imran

Muhammad Najam Sadiq

Sajid Inayat

Abbreviations

SRS	Software Requirement Specification
PC	Personal Computer
BMS	BIMS Management System
DB	Data Base
MS	Management System

Table of Contents

Introduction.....	1
1.1 Brief	1
1.2 Relevance to Course Modules	1
1.3 Project Background.....	1
1.4 Literature Review.....	2
1.4.1 Analysis from Literature Review	2
1.5 Methodology and Software Life Cycle.....	2
1.5.1 Rational Behind Selected Methodology	2
Problem Definition.....	3
2.1 Problem Statement	3
2.1.1 Problem Solution	3
2.2 Deliverables and Development Requirement	3
2.2.1 User Authentication and Authorization	3
2.2.2 Course Management	3
2.2.3 Communication and Collaboration	3
2.2.3 Course Management	3
2.2.4 Enrollment and Registration	3
2.2.5 Reporting and Analytic	3
2.3 Current System.....	4
2.3.1 Edmodo	4
2.3.2 Classroom	4
2.4 Purposed System.....	5
2.4.1 Improved administration.....	5
2.4.2 Enhanced security	5
2.4.3 Customization options	5
2.4.4 Improved Communication	5

Requirement Analysis.....	7
3.1 Use cases	7
3.1.1 Use Case Diagarm.....	7
3.1.1 Use Case Discription.....	8
3.1.2 Use Case Discription.....	9
3.1.3 Use Case Discription.....	10
3.1.4 Use Case Discription.....	11
3.1.5 Use Case Discription.....	12
3.1.6 Use Case Discription.....	13
3.1.7 Use Case Discription.....	14
3.2 Functional Requirments	14
3.2.1 Adminstrative Level Requirment.....	14
3.2.2 Teacher Level Requirment.....	15
3.2.3 Student Level Requirment	15
3.3 Non – Functional Requirments	15
3.3.1 Usability	15
3.3.2 Performance	15
3.3.3 Security	15
3.3.2 Reliability.....	15
3.4 Sequence Diagram	16
The Design and Architecture	21
4.1 System Architecture	21
4.1.1 Architecture Overview	21
4.1.2 Data Management	21
4.1.3 Functionsl Components	21
4.1.4 Scailibilty anf Performance.....	22
4.2 System Design	22
4.3 Class Diagram	23
Implementation	24
5.1 Component Diagram.....	24

5.2	Network and Protocol Choice	25
5.2.1	Network Architecture Choice	25
5.3	Choice of Object Middleware	25
5.4	User Interface	26
5.4.1	User Interface Design	26
5.4.2	Data Base Design	27
5.4.3	Application Modules	28
5.4.3	Data Flow Integration	29
5.4.4	Security Measures	29
Testing and Evaluation.....		30
6.1	Verification	30
6.1.1	Codes Defects	30
6.1.2	Fuctional Defects	30
6.1.3	Usability Defects.....	30
6.2	Validation.....	30
6.3	Unit Testing	30
6.4	Integration Testing	30
6.5	Test Cases	31
Conclusion and Future Work		37
7.1	Summary	37
7.2	Conclusion	37
7.3	Future Work	38
7.3.1	Integration with other educational tools and platforms	38
7.3.2	Enhanced analytics and reporting	39
7.3.3	Improved accessibility	39
7.3.3	Expanded course content	39
References.....		40

List of Figures

Fig 2.1 Software development life cycle	6
Fig 3.1 Use Case Diagram	7
Fig 3.2 Sequence Diagram of Show Performance	16
Fig 3.3 Sequence Diagram of Course	17
Fig 3.4 Sequence Diagram of View Profile	17
Fig 3.5 Sequence Diagram of Login	18
Fig 3.6 Sequence Diagram of Perform Task.....	19
Fig 3.7 Sequence Diagram of Assign Performance	20
Fig 4.1 Class Diagram.....	23
Fig 5.1 Entity Relation Diagram.....	24
Fig 5.2 Login Interface	26
Fig 5.3 Admin Dashboard.....	27
Fig 5.4 Teacher List	27
Fig 5.5 Course List.....	28
Fig 5.6 Modules of Admin.....	28

List of Tables

Table 2.1 Tools and Technologies	6
Table 3.1 Login	8
Table 3.2 View enrolled courses.....	9
Table 3.3 Zoom meeting links	10
Table 3.4 Perform Task.....	11
Table 3.5 Event Management	12
Table 3.6 View Profile.....	13
Table 3.7 View Chat	14
Table 6.1 TC001	31
Table 6.2 TC002	31
Table 6.3 TC003	32
Table 6.4 TC004	32
Table 6.5 TC005	33
Table 6.6 TC006	33
Table 6.7 TC007	34
Table 6.8 TC008	35

Chapter 1: Introduction

The BIMS Management System is a multi-environmental website which is using the web services of web based open source BMS of Model. It provides all the necessary functionality to students, teachers and director. Teachers can share learning material with students, they can also view the assignments, quizzes submitted by students. Meanwhile students can also view, download learning material. A manager (director) can monitor the students and teacher activities.

1.1 Brief

The advanced in technology gives many advantages to the people. All kinds of information and be shared easily by the technology through the Internet. This kind of information sharing will lead to the new learning behavior. University students are mostly independent in their learning as lectures usually give out lecture notes, and further information are left for the students to discover on their own, as it is not a one- way learning process which is practiced in the primary and secondary school system. The learning process at the university level is a two-way process, lectures share their knowledge and students give their opinions or thoughts in return a topic in a class discussion. Therefore, university students need to constantly broaden their knowledge by searching for information. This system (BMS) provides an avenue for classroom materials or activities to be shared easily. It is also portal that enables lectures and students to interact out of the classroom, having discussions though forums that could otherwise take up too much of the time supposed to be spent learning in the classroom.

1.2 Relevance to Course Modules

Our project will provide a better user-friendly interface to the students. It will help to use the more effectively. We will work on the new features in the process. The BMS website development project encapsulates key concepts and techniques from various course modules, such as web design, user interface optimization, and information management systems. This Model based MS server that performs the base functionality and a user interface that is operated by instructors, students and administrators. This system will allow instructors and administrators to efficiently manage elements such as user registration, study content, calendars, user access, communication and notifications.

1.3 Project Background

The basic feature of uploading courses, assigning roles, generating various report about performance appraisal. It also helps in managing the course activities, like courses calendar, different tasks related to course etc. Users and trainers can send reminders and notification, including upcoming training and events. System also helps to conduct the assessment of the students through quizzes and assignments. With the help of this project one can track and check the skills set of the team members and compare then against business goals.

The project aims to address the limitations of traditional administrative methods, promote efficient communication, and provide a user-friendly interface for seamless interaction among students, faculty, and administrative staff.

1.4 Literature Review

There are many systems used for the same purposes to manage their database, working, authorize people etc. They all refer to our project as they work only for the communities/organizations that use them. Our project will also manage our workplace environment i.e., BMS. A thorough literature review was conducted to explore existing research, studies, and best practices in the field of educational management systems and student portals. The review examined various approaches, technologies, and frameworks implemented in similar institutions to gather insights and identify trends. It provided valuable knowledge about successful implementations, potential pitfalls, and emerging technologies that influenced the design and development of the BMS. Extensive research into educational portal systems and user interface design underscored the importance of intuitive navigation, comprehensive data management, and interactive interfaces in promoting effective learning and administrative efficiency.

1.4.1 Analysis from Literature Review

Based on the findings from the literature review, an analysis was conducted to evaluate the suitability of different approaches and technologies for the BMS. This analysis considered factors such as scalability, security, usability, and compatibility. The insights gained from the analysis guided the selection of appropriate technologies and the design of the portal architecture to ensure optimal performance and user experience. Another very important thing to keep in mind is that BMS are very good in keeping track of learning and progress of students, but they are not a tool to encourage useful interactive learning. There needs to be a desire and motivation from the students to learn only then these systems will achieve their desired objectives.

1.5 Methodology and Software Life cycle for This Project

The development of the BMS follows a waterfall methodology and software life cycle approach, encompassing comprehensive phases of requirement analysis, system design, iterative development, rigorous testing, deployment, and ongoing maintenance. By adhering to industry best practices and standards, the project ensures the creation of a robust, scalable, and user-centric portal tailored to the specific needs of BIMS Management System.

1.5.1 Rationale behind Selected Methodology

The selection of the methodology is based on its ability to accommodate evolving project requirements, promote effective collaboration, and ensure timely project delivery. The chosen methodology enables the project team to prioritize user feedback, adapt to changing technological trends, and deliver a BMS that reflects the institute's commitment to technological advancement and educational excellence.

Chapter 2: Problem Definition

The problem definition section provides a concise overview of the key challenges and issues identified within the existing administrative and communication systems at Barani Institute of Management and Science (BIMS). This section serves as a precursor to the subsequent delineation of the specific problem statement and proposed solutions within the context of the BMS development project.

2.1 Problem Statement

The problem statement encapsulates the core issues and inefficiencies within the current administrative and communication frameworks at BMS. It outlines the constraints and limitations faced by students, faculty, and administrative staff, emphasizing the need for a comprehensive digital solution to streamline operations, improve accessibility, and enhance user experience.

2.1.1 Problem solution:

The problem solution segment presents an overview of the proposed resolutions and strategies to address the identified challenges. It highlights the key features and functionalities of the BMS that aim to mitigate the existing limitations, foster effective communication, and streamline administrative processes within the institute.

2.2 Deliverable and Development Requirements

The BMS will provide the following key functions:

- 2.2.1 User Authentication and Authorization:** Secure login and role-based access control to ensure that only authorized users can access relevant features and data.
- 2.2.2 Course Management:** Faculty can manage course materials, assignments, quizzes, and grades. Students can access and submit assignments, participate in discussions, and view their progress.
- 2.2.3 Communication and Collaboration:** Real-time messaging, discussion forums, and announcement features to facilitate seamless communication and collaboration among students, faculty.
- 2.2.4 Enrollment and Registration:** Streamlined processes for student enrollment, course registration.
- 2.2.5 Reporting and Analytic:** Generate reports on student performance, course evaluation, and administrative data analysis to support decision-making and performance monitoring. The project deliverable subsection outlines the tangible outcomes and results expected upon the successful implementation of the BMS. It includes a detailed list of specific deliverable, such as a fully functional user interface, integrated administrative tools, comprehensive data management systems, and effective communication channels, highlighting their significance in addressing the identified problem areas and fulfilling the institute's digital need.

2.3 Current System

Following are the projects which have some common features of BMS and have simpler interface:

2.3.1 Edmodo

Edmodo is geared more towards educators. More specifically, teachers, students (mainly K12) and parents would benefit from using this tool because of its collaborative nature. It allows students to practice communication skills in a controlled environment which is so important to student development. Edmodo also provides a secure environment for students to use develop positive social networking skills. Students can post content, collaborate with each other and connect with their teachers in meaningful ways. Using Edmodo is a great way to promote and teach online safety, digital citizenship and online netiquette in younger students. The use of online badges is also a great motivator for student learning. I believe that creating blog entries in Edmodo can allow students to express their creativity and themselves through writing. Blog entries can challenge students to utilize higher order thinking skills and therefore increase student engagements with the lesson. The use of Edmodo has allowed for evaluations of students learning through the elements that the corporate author of Edmodo facilitates on its platform, Schoology: The objectives of this free platform is to create a learning strategy for students and to motivate studying online. On this platform students and teachers are able to create groups and courses, administer resources, set courses created as public or private, integrated resources from external platforms and present statistics of the progress of each student. From a technical perspective, Edmodo fit three very important criteria:

Usability: Ease of user Students/teacher/parents/administrators learn to use the tool in a short period of time and therefore will be more inclined to use it for classroom discussion, content and learning management.

Accessibility: Accessibility to all users Students/teacher/parents/administrators are able to access to the tool using various devices with all browser types and from anywhere, anytime and anyplace.

Compatibility: Compatibility with multiple devices and equipment. Since users will not all have the same devices or equipment is important that the tool is flexible enough to be used with a variety of devices or equipment.

2.3.2 Classroom:

Google Classroom is a free web services developed by Google for schools that aim to simplify creating, distributing and grading assignments. The primary purpose of Google Classroom is to streamline the process of sharing files between teachers and students. It is estimated between 40 to 100 million people use Google Classroom. As the speed of uploading files is perfect for working inside and outside the classroom.

It is free for educational centers, designed for teachers and students includes email, storage and forums for collaboration between professors and students in the same class or different classes and allows users to share or integrate videos, PDFs and images. The user can also incorporate the URLs of virtual libraries or external resources. You can use Google Classroom in your school to streamline assignments, boost collaboration and foster communication.

Classroom is available on the web or by mobile app. You can use Google Classroom with many tools that you already use, such as Gmail, Google Docs and Google Calendar.

There are following functionalities which user can perform by using Google Classroom.

Teachers

- Start a video meeting.
- Create and manage classes, assignments and grades online without paper.
- Add materials to your assignments such as YouTube videos, Google Forms survey and other items from Google Drive.
- Give direct, real-time feedback.
- Use the class stream to post announcements and engage students in question-driven discussion.
- Invite parents and guardians to sign up for email summaries with students upcoming or missing work.

Students

- Track classwork and submit assignments.
- Check originality, feedback and grades.
- Share resources and interact in the class stream or by email.

Guardians

- Get an email summary of your student's work.
- Review announcements and activities.

Administration

- Protect data and set permissions for your users.
- Set up classes and rosters.
- Add or remove students and teacher from classes.
- Get 24/7/support.

2.4 Purposed System

The Purposed System has many advantages like:

2.4.1 Improved administration:

Admins will have more control over the system, allowing for more efficient and effective management of course content, student information, and other critical data.

2.4.2 Enhanced security:

With increased control, the security of sensitive data will be improved, ensuring the confidentiality and privacy of students and staff.

2.4.3 Customization options:

The system can be customized to meet the specific needs of the institution, providing a tailored solution that is better suited to their requirements. Better reporting and analytics. With greater control, administrators will have access to a wider range of data and reporting tools, enabling them to make informed decisions and track performance metrics more effectively.

2.4.4 Improved communication:

The new system will allow for improved communication between administrators, directors, teachers, and students, streamlining the flow of information and enhancing collaboration.

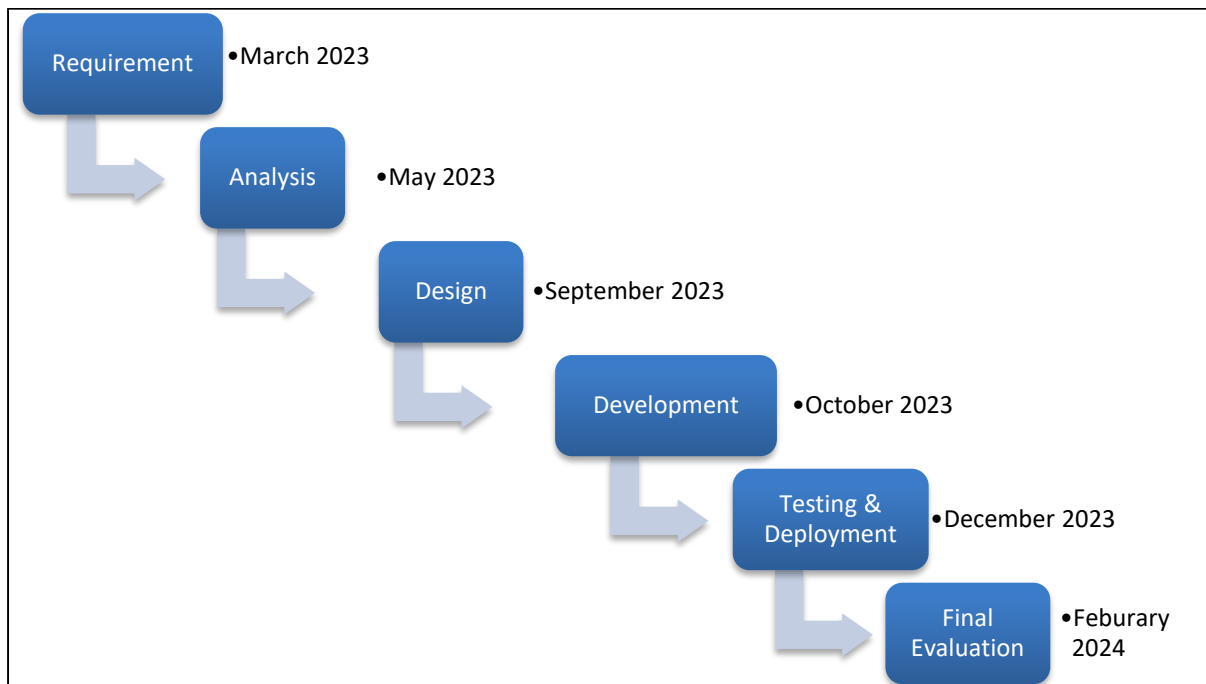


Figure 2.1: Software development life cycle

Table 2.1: Tools and technologies

Tools and technologies	Tools	Version
	Visual Studio	2023
	Xampp	8.2.12
	Laravel Framework	5.2.6
	Technology	Versions
	My SQL	

Chapter 3: Requirement Analysis

The focus is on requirements analysis. The chapter begins with an overview of the use case analysis, which includes the identification and description of various use cases within the website. All the concepts and flow of data that are used to develop this website are discussed in this chapter. Data store represents stored information (either temporary or permanently) that is used by software. Mostly websites are difficult to learn and use for the beginners and it is problem for the new user. Additionally, the chapter covers non-functional requirements, which focus on the qualities and characteristics of the BMS beyond its specific functionalities. Non-functional requirements may include usability, performance, security, reliability.

3.1 Use Cases

Use case diagram is a graphical depiction of the interaction among the elements of BMS. It represents the methodology use in system analysis to identify, clarify, and organize system requirements of the BMS. Next the use case analysis, the chapter delves into the functional requirements of the BMS. These requirements outline the specific functionalities and features that the application must possess.

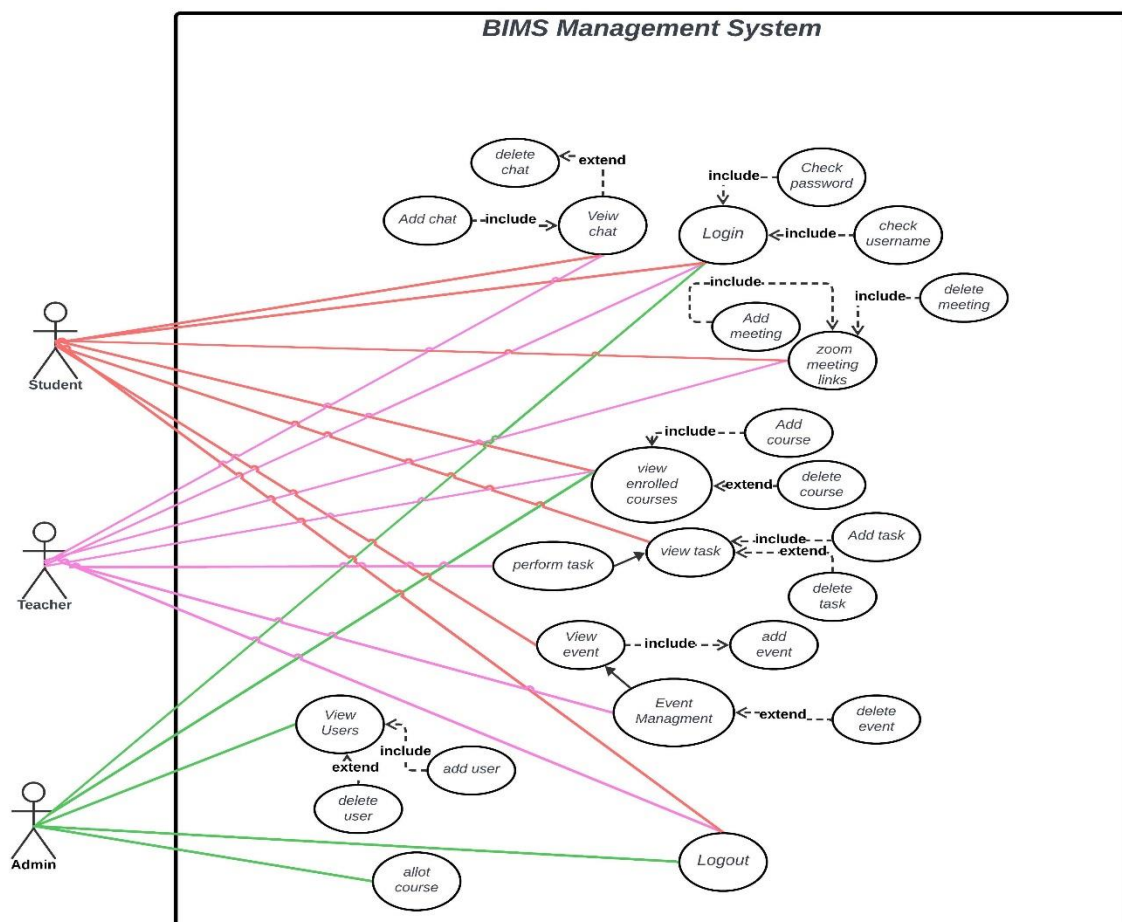


Figure 3.1: Use Case Diagram

3.1.1 Use Case Description:

Table 3.1: Login

Use Case ID:	UC-1
Use Case Name:	Login
Actors:	Student, Teacher, Admin
Description:	It will allow users to sign up to the website.
Trigger:	A user wants to use the BMS management system.
Pre-conditions:	The BMS is accessible, and the user has a valid email address.
Post-conditions:	The user's registration information is stored in the database, and the user can log in to the website.
Normal Flow:	<ul style="list-style-type: none">• The user navigates to the home page.• The user provides the required information, including name, email address, and password.• The user submits the login/sign up form.• The system validates the user's information.• If the information is valid, the system creates a new user account and sends a verification email to the provided email address.• The user verifies the email by clicking on the verification link.• The system activates the user account.• The user receives a confirmation message and can now log in to the website.• On the other hand, old user simple login by their user name and password.
Alternative Flows: [Alternative Flow 1 – Not in Network]	A1: Invalid Information If the user provides invalid or incomplete information, the system displays an error message and prompts the user to correct the information. A2: Existing Email Address If the user provides an email address that is already registered in the system, the system displays an error message and prompts the user to use a different email address.
Exceptions:	E1: Email Verification Failure If the user fails to verify the email within a specified time period, the system disables the account and prompts the user to re-register.
Includes:	Add dates, Valid Emails
Special Requirements:	User must login or sign up to the website.
Assumptions:	The user has access to a valid email address for registration.
Notes and Issues:	User is only the student/ teacher of BIMS and Manager. And he/she have a valid email.

3.1.2 Use Case Description:

Table 3.2: View Enrolled Course

Use Case ID:	UC-2
Use Case Name:	View enrolled courses
Actors:	Student, Teachers
Description:	The student can see the courses semester wise which they're enrolled in.
Trigger:	A user can see the courses which are assign to them.
Pre-conditions:	The student must be enrolled in the subject.
Post-conditions:	The user's enrollment form must be filled.
Normal Flow:	<ul style="list-style-type: none">• The user can see the enrolled courses.• The student will have provided by the course content like files and lecture.• The uploaded lectures can be seen by the students only.• The teacher can upload lectures on the BMS.
Alternative Flows: [Alternative Flow 1 – Not in Network]	A1: View semester courses The students can only view the courses which they're currently enrolled in. A2: Add/ Delete course The teacher will be able to delete or add a course accordingly to the task performance. Admin can also add or delete courses.
Exceptions:	E1: Drop out If the student fails, the exam he/she can be drop out of the university. E2: freeze semester The student can also freeze the semester it will allow them to view only past performance.
Includes:	None
Special Requirements:	The teacher and student must be registered in the university.
Assumptions:	The student and teacher should be enrolled in the specific courses.
Notes and Issues:	If the student is enrolled, he/she will be able to view course.

3.1.3 Use Case Description:

Table 3.3: Zoom meeting links

Use Case ID:	UC-3
Use Case Name:	Zoom meetings links
Actors:	Student, Teacher
Description:	This use case involves the online zoom meetings links.
Trigger:	A teacher can provide its personal zoom meetings links to the student.
Pre-conditions:	The teacher has to provide the links of their personal meeting room to the student.
Post-conditions:	The links will be view and accessed by the students to join the meetings.
Normal Flow:	<ul style="list-style-type: none">• The teacher can add the meetings links which can be viewed by the students.• The students can copy those links to join the meetings.• Moreover, it is provided whenever there are online classes schedule.• Once the meeting is expiring the student can join the meeting again by using the same link.• Only teachers can add or delete a link.
Alternative Flows: [Alternative Flow 1 – Not in Network]	A1: Links expire If the links expires the teacher have to provide the new link to join the meetings. A2: Waiting room If the teacher has opened the waiting room setting the student will have to wait until the teacher let the student to join the request.
Exceptions:	E1: Wrong Passcode: If there is a wrong passcode provided by the teacher, then they'll have to provide the right passcode.
Includes:	Valid zoom meeting links.
Special Requirements:	Student have to join the links in case there are online classes are schedule.
Assumptions:	The students have access to the links.
Notes and Issues:	None.

3.1.4 Use Case Description:

Table 3.4: Perform Task

Use Case ID:	UC-4
Use Case Name:	Perform task
Actors:	Student
Description:	This use case involves the process of performing task like Quiz, Assignment by the students.
Trigger:	A Student want to perform the specific task.
Pre-Conditions:	The assignment or quiz is available to the student when the teacher upload its.
Pos-conditions:	The students can upload answer file.
Normal Flow:	<ul style="list-style-type: none">• The student can perform assignment.• The student can also perform quiz.
Alternative Flows: [Alternative Flow 1 – Not in Network]	A1: Perform quiz If student can't perform quiz it means he/she isn't enrolled in the course or they're not allowed by the admin/ teacher to perform the task. A2: Perform assignment If student can't perform assignment, it means he/she isn't enrolled in the course or they're not allowed by the admin/ teacher to perform the task.
Exceptions:	E1: Quiz not submitted If the user fails to submit quiz he/she can submit again or check his/her network connection.
Includes:	Add quiz and Add assignment
Special Requirements:	The File can be in pdf, word ,jpg file.
Assumptions:	The user can perform task.
Notes and Issues:	User can perform task if he/she has enrolled in that course.

3.1.5 Use Case Description:

Table 3.5: Event Management

Use Case ID:	UC-5
Use Case Name:	View event
Actors:	Student, Teacher
Description:	This use case involves the event management. It can be viewed by the students. And the teacher can add or delete the event happening at the campus.
Trigger:	A user can view the events that are added by the teachers.
Pre-Conditions:	The students can view the event if it is added by the teacher.
Pos-conditions:	The events information is stored in the event management module can be seen by the student.
Normal Flow:	<ul style="list-style-type: none">• The events are shown in the module.• They're only be edited by the teacher.• They're only viewed by the student.
Alternative Flows: [Alternative Flow 1 – Not in Network]	None.
Exceptions:	None.
Includes:	Add events, delete events
Special Requirements:	None.
Assumptions:	The teacher can add or remove them.
Notes and Issues:	The student can see the event if they're added.

3.1.6 Use Case Description:

Table 3.6: View Profile

Use Case ID:	UC-6
Use Case Name:	View profile
Actors:	Admin
Description:	This use case involves the profile home page which can be seen by the users if they're successfully login to the website.
Trigger:	A user initiates the action to view their profile.
Pre-conditions:	The user must log in or register on the BMS. The user's requirements must be validated to access and view the profile
Post-conditions:	Upon successful login, the user gains access to their profile and can utilize the functions available within it.
Normal Flow:	<ul style="list-style-type: none">• User logs in to the system.• User navigates to the profile section.• The system displays the user's profile.
Alternative Flows: [Alternative Flow 1 – Not in Network]	If the user isn't connected to the network, an appropriate error message or prompt should be displayed, guiding the user to connect to the network.
Exceptions:	E1: Unsuccessful Login If the user doesn't list in the DB he/she won't be able to view the profile. E2: System Error: If there is a technical error the system will show the error message according to what the problem is.
Includes:	Any functionalities or modules that are included in the profile viewing process. Example: view event, courses, teacher lists, chat etc.
Special Requirements:	The user connectivity to the network can give access to the profile.
Assumptions:	The teacher can add or remove them.
Notes and Issues:	Additional features like profile editing, settings, or additional permissions could be considered for future enhancements.

3.1.7 Use Case Description:

Table 3.7: View Chat

Use Case ID:	UC-7
Use Case Name:	Chat
Actors:	Student, Teacher
Description:	It will solve the queries of the students about the BMS.
Trigger:	A user have any query related BMS it can chat with the Teacher.
Pre-conditions:	The chat box is accessible 24/7 to the students.
Post-conditions:	The teacher will be able to solve the queries of the students.
Normal Flow:	<ul style="list-style-type: none">• The chat box will help students to interact with the teacher.• The student can upload file if he/she has submitted wrong file in the quiz or assignment they can chat with the teacher and upload the file again.
Alternative Flows: [Alternative Flow 1 – Not in Network]	A1: Invalid Information If the user provides invalid or incomplete information, the system displays an error message and prompts the user to correct the information.
Exceptions:	E1:
Includes:	Student can upload file and the teacher can response to the specific student.
Special Requirements:	User must add chat in order to get the response.
Assumptions:	The queries are solved.
Notes and Issues:	It will not work if there is not any chat in the chat list.

3.2 Functional Requirements

A requirement specification is a collection of all requirements that are to be imposed on design and verification of the product. The specification also contains other related information necessary for the design, verification, and maintenance of our application. According to nature of attaining specific behavior of the application following specifications were made by the supervisor.

3.2.1. Administrative level requirements

1. Login
2. Register user
3. Enroll users of their respective course
4. View course details
5. View course participants
6. Listing of Teachers
7. Listing of active students
8. Listing of courses

3.2.2. Teacher level requirements

1. Login
2. Listing of students
3. Create assignments and quizzes
4. Listing of all courses
5. Upload files
6. Remove files
7. Late submission and In-time submissions control
8. Event management

3.2.3. Student level requirements

1. Login
2. View courses
3. View assignments and quizzes
4. Submit assignment and quizzes
5. Download file
6. View event management

3.3 Non-Functional Requirements

3.3.1 Usability:

The BMS shall have a user-friendly and intuitive interface. Users, including students, faculty, and administrators, should be able to navigate the application easily and perform tasks without confusion. The system shall provide clear instructions and guidance for various operations within the application.

3.3.2 Performance:

The BMS shall exhibit optimal performance to handle multiple users concurrently. Response times for user actions, such as login, course enrollment, and grade retrieval, should be fast and efficient. The system shall be able to handle a significant number of concurrent users without significant performance degradation.

3.3.3 Security:

The BMS shall ensure the security and privacy of user data. User authentication and access control mechanisms shall be implemented to prevent unauthorized access. The system shall encrypt sensitive information, such as passwords and personal data, to protect against unauthorized disclosure.

3.3.4 Reliability:

The BMS shall be reliable and available to users without frequent downtime. The system shall implement backup and recovery mechanisms to ensure data integrity and minimize data loss in the event of system failures. Regular maintenance and monitoring procedures shall be in place to identify and resolve issues promptly.

3.4 Sequence Diagram

A Sequence Diagram is a type of interaction diagram in the Unified Modeling Language (UML) that shows how processes operate with one another and in what order. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

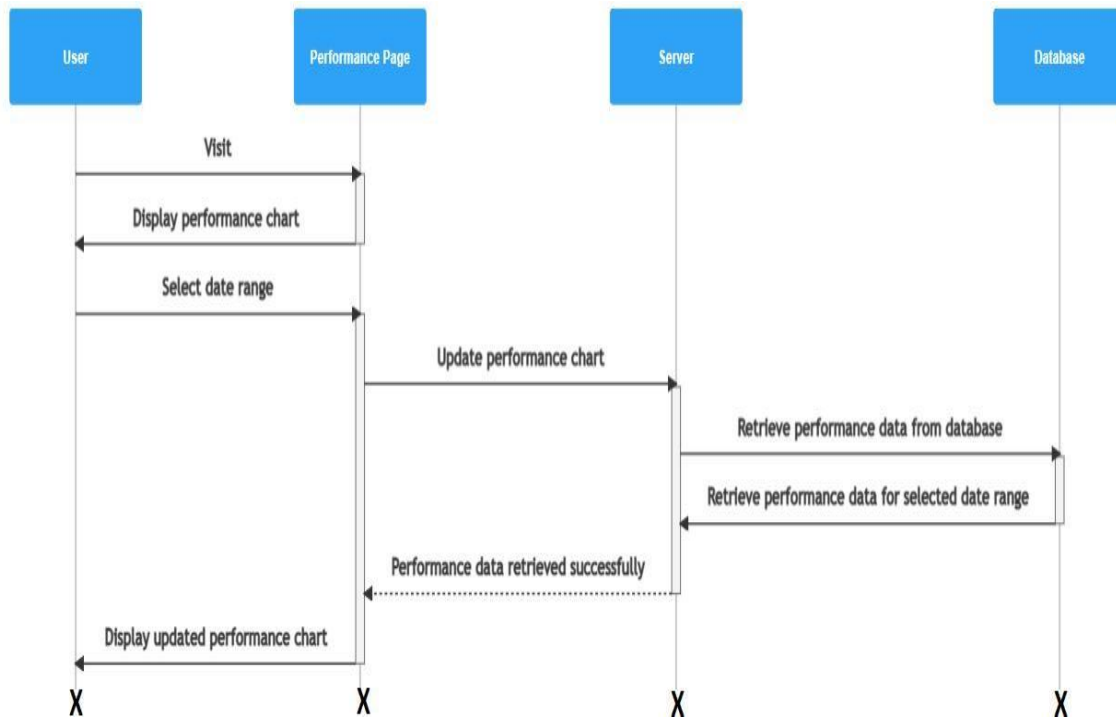


Figure 3.2 Sequence Diagram of Show Performance

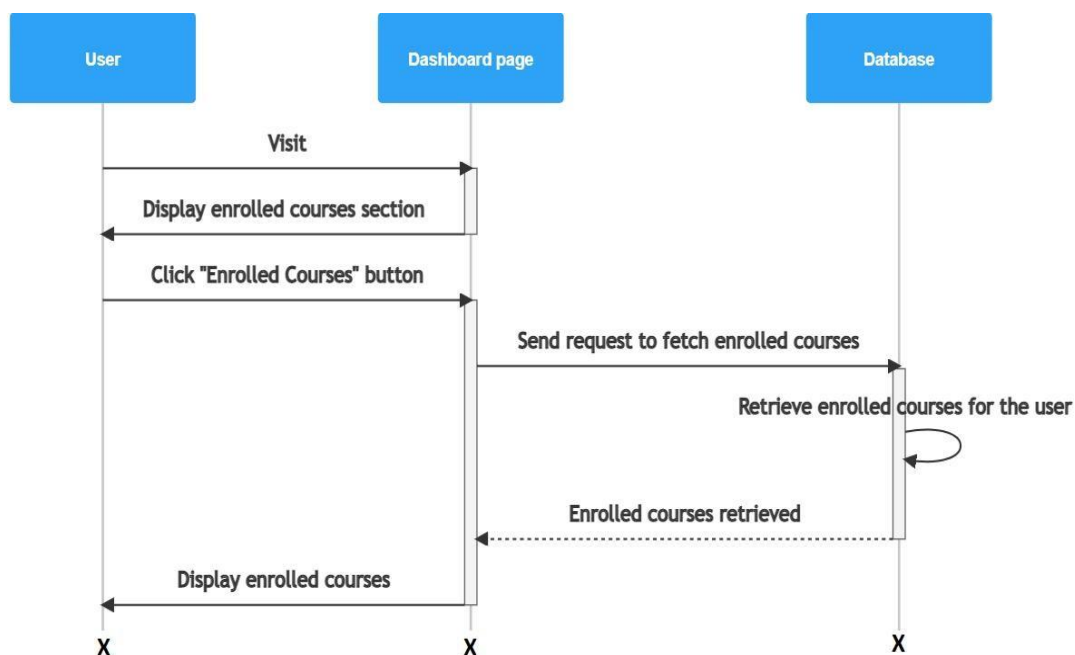


Figure 3.3 Sequence Diagram of Courses

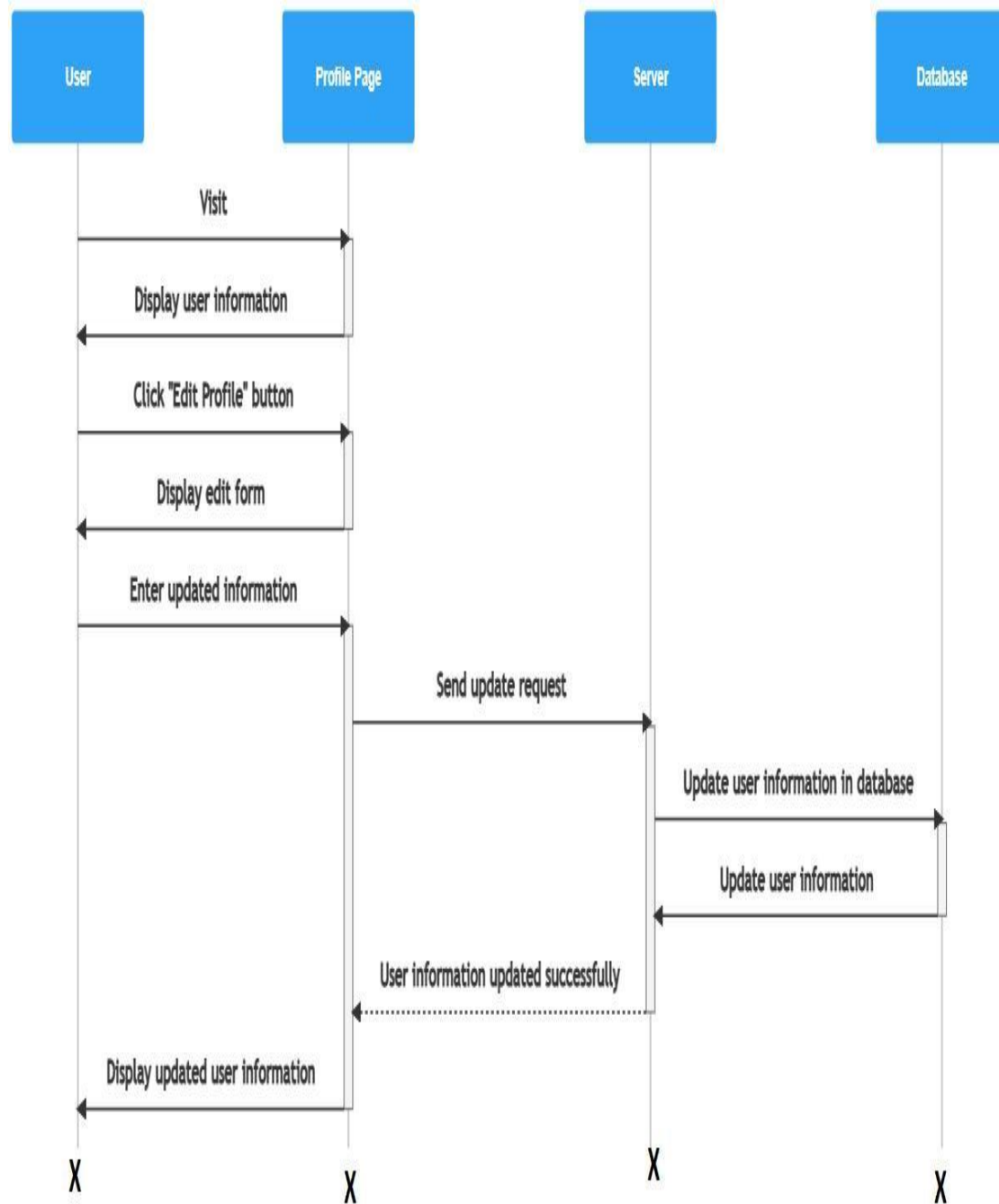


Figure 3.4 Sequence Diagram of View Profile

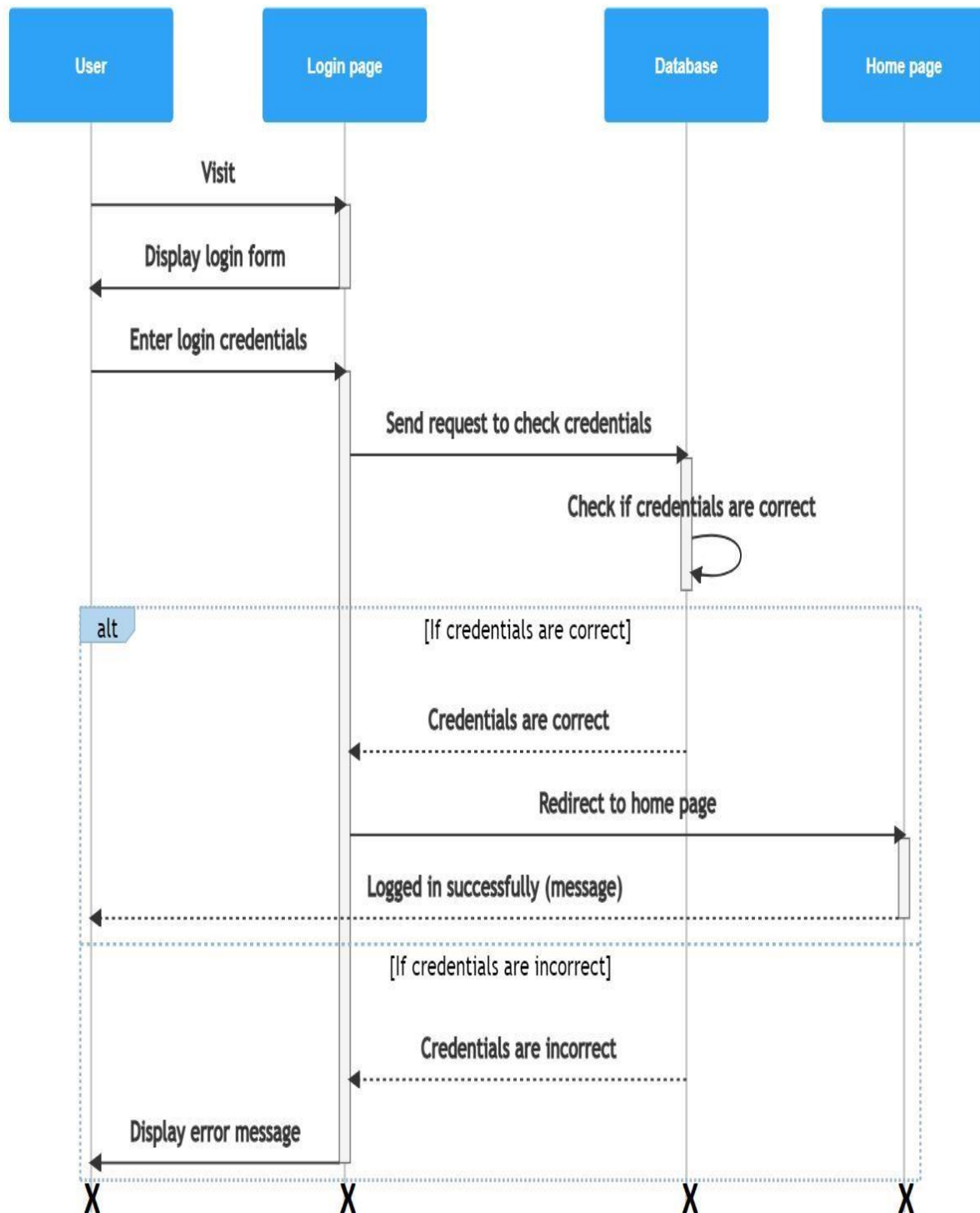


Figure 3.5 Sequence Diagram of login

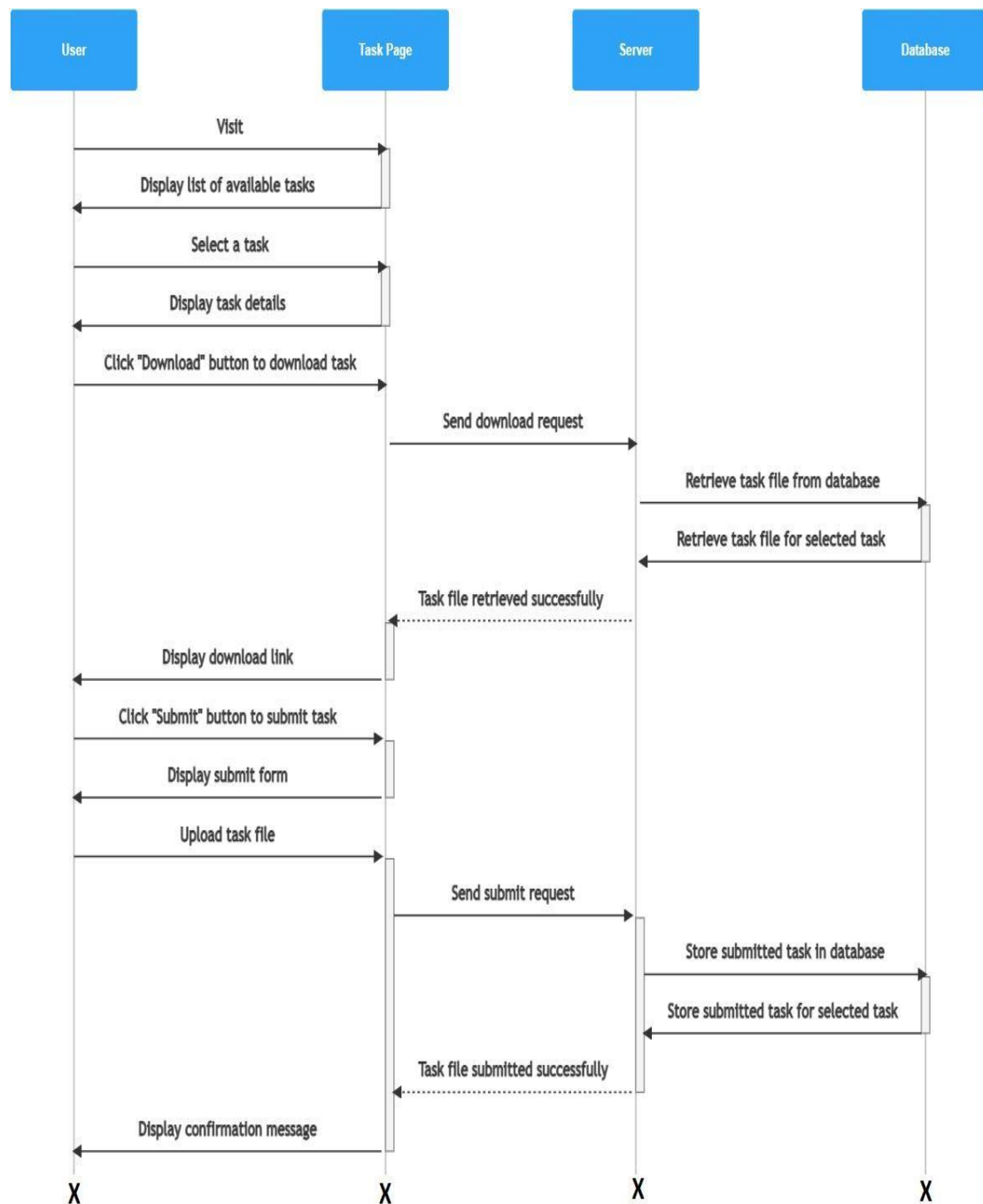


Figure 3.6 Sequence Diagram of Perform Task

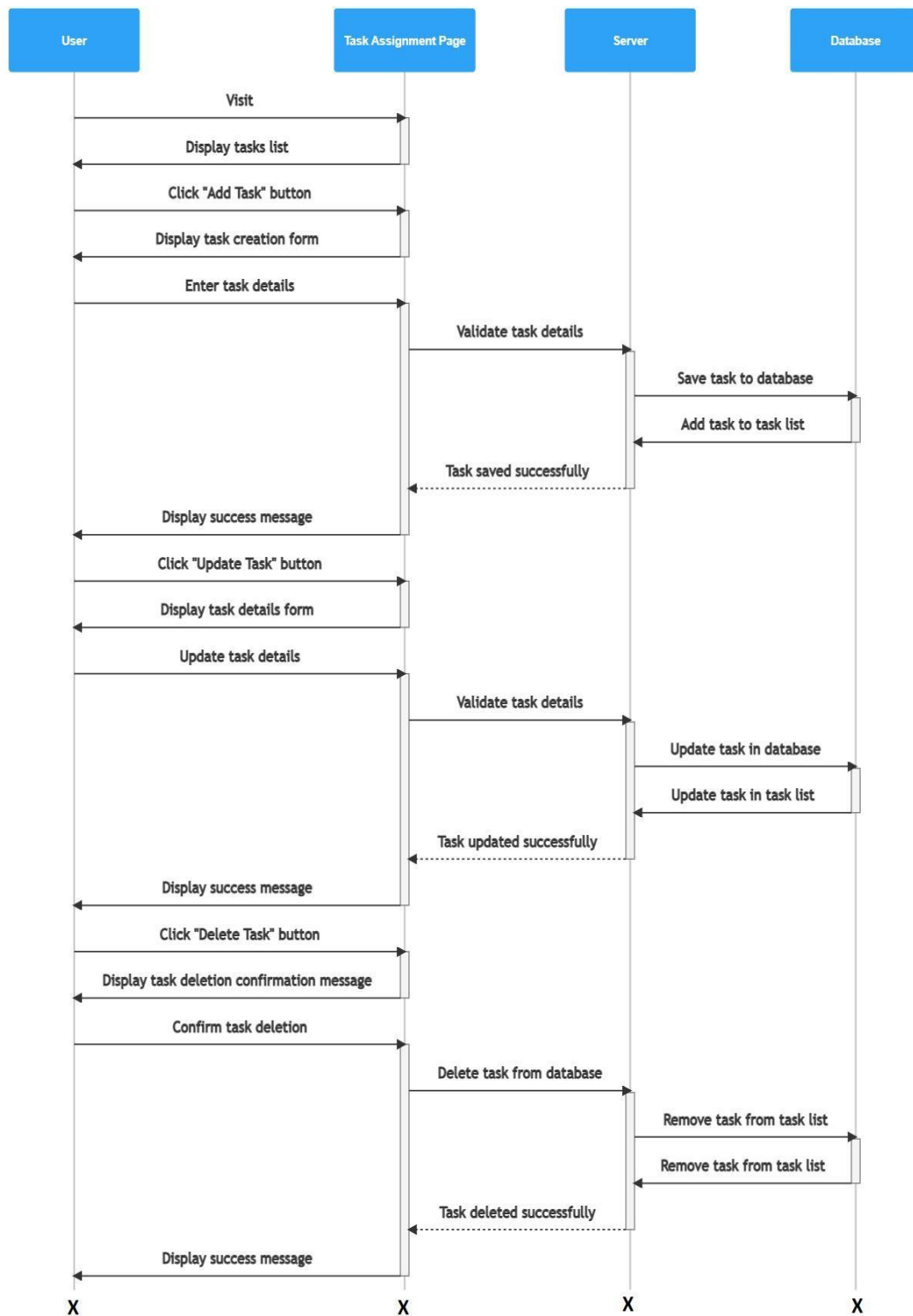


Figure 3.7 Sequence Diagram of Assign Task

Chapter 4: Design and Architecture

The system architecture of the BMS is designed to ensure seamless integration, data security, and efficient communication between different components. One of the primary goals of the BMS is to facilitate effective communication and collaboration between teachers and students beyond the traditional classroom setting. Through this Website, teachers can effortlessly share learning materials, such as lecture notes, presentations, and supplementary resources, enabling students to access and review them at their convenience. Additionally, students can engage in discussions, ask questions, and seek clarification from their peers and instructors, fostering a dynamic learning environment. The BMS offers a comprehensive suite of tools for managing courses, ensuring seamless coordination and organization of course activities. Instructors can effortlessly upload course content, including lectures, assignments, quizzes, and exams, and students can conveniently access and submit their work through the app. Furthermore, the Website supports automated reminders and notifications, keeping students informed about upcoming deadlines, important announcements, and course updates, thus promoting accountability and timely completion of tasks. It encompasses the following key elements:

4.1 System Architecture

4.1.1 Architecture Overview:

The BMS follows a client-server architecture. The client-side incorporates HTML, CSS, and PHP for interactive and responsive user interfaces. The server-side employs a Model-View-Controller (MVC) architecture, utilizing technology Laravel and Express for handling user requests, and a MySQL database for data storage and management. The communication between the client and server occurs through the HTTPS protocol to ensure secure data transmission.

4.1.2 Data Management:

The database structure is organized into separate tables for students, faculty, courses, and administrative data, ensuring efficient data storage and retrieval. Data processing involves real-time validation of user inputs, and scheduled data updates to maintain data accuracy and reliability. Data security is implemented through user authentication protocols, encrypted data transmission, and regular security audits to protect sensitive information from potential threats.

4.1.3 Functional Components:

The user authentication module enables secure user registration, login authentication, and session management, ensuring that only authorized users can access the portal's features and resources. The course management module allows faculty members to create, update, and manage course content, assignments, and assessments, facilitating a structured and organized learning environment. The communication module facilitates real-time messaging, announcement broadcasts, and discussion forums, promoting effective communication and collaboration among students, faculty, and administrative staff. The administrative module provides administrative staff with the necessary tools to manage student records, financial data, and institutional resources efficiently within the portal.

4.1.4 Scalability & Performance:

The system architecture is designed to be scalable, employing load balancing techniques, data caching mechanisms, and horizontal scaling to accommodate an increasing number of users and data volume while maintaining optimal system performance. Performance optimization measures, such as code optimization, database indexing, and resource caching, are implemented to minimize latency, enhance response times, and ensure a smooth and seamless user experience.

4.2 System Design

The system design of the BMS is meticulously crafted to ensure a user-friendly interface, robust data management, and efficient communication. It encompasses the following key elements: The BMS is an innovative Website designed to revolutionize the learning experience for students, teachers, and directors. One of the primary goals of the BMS is to facilitate effective communication and collaboration between teachers and students beyond the traditional classroom setting. Through this app, teachers can effortlessly share learning materials, such as lecture notes, presentations, and supplementary resources, enabling students to access and review them at their convenience. Additionally, students can engage in discussions, ask questions, and seek clarification from their peers and instructors, fostering a dynamic learning environment. The BMS offers a comprehensive suite of tools for managing courses, ensuring seamless coordination and organization of course activities. Instructors can effortlessly upload course content, including lectures, assignments, quizzes, and exams, and students can conveniently access and submit their work through the app. Furthermore, it supports automated reminders and notifications, keeping students informed about upcoming deadlines, important announcements, and course updates, thus promoting accountability and timely completion of tasks. The assessment module within the BMS empowers instructors to create and administer various types of assessments, including quizzes, tests, and exams. The app provides a flexible and customizable assessment creation interface, allowing instructors to design questions with different formats, such as multiple-choice, true or false, and essay questions. It also offers instant grading and feedback, enabling students to receive prompt evaluation of their performance and providing valuable insights for improvement. The project scope of the BMS encompasses a user-friendly interface tailored to the frequently used features of the existing BMS. This customization aims to streamline the director's experience by providing quick access to essential functionalities, such as course management, user administration, and data analytics. The app's design prioritizes ease of use, ensuring a seamless user experience for directors, instructors, and students alike.

4.3 Class Diagram

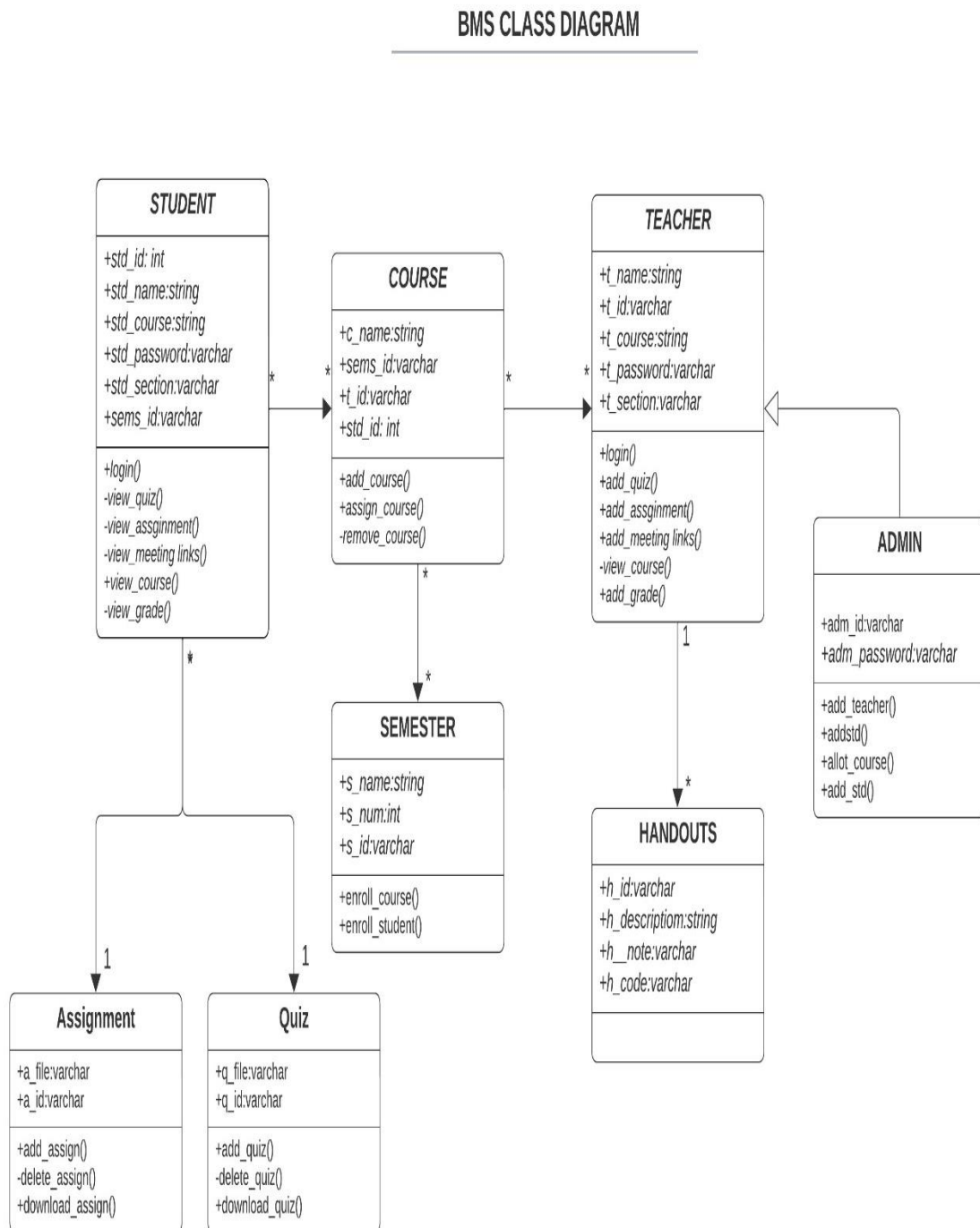


Figure 4.1 Class Diagram

Chapter 5: Implementation

5.1 Entity Relation Diagram:

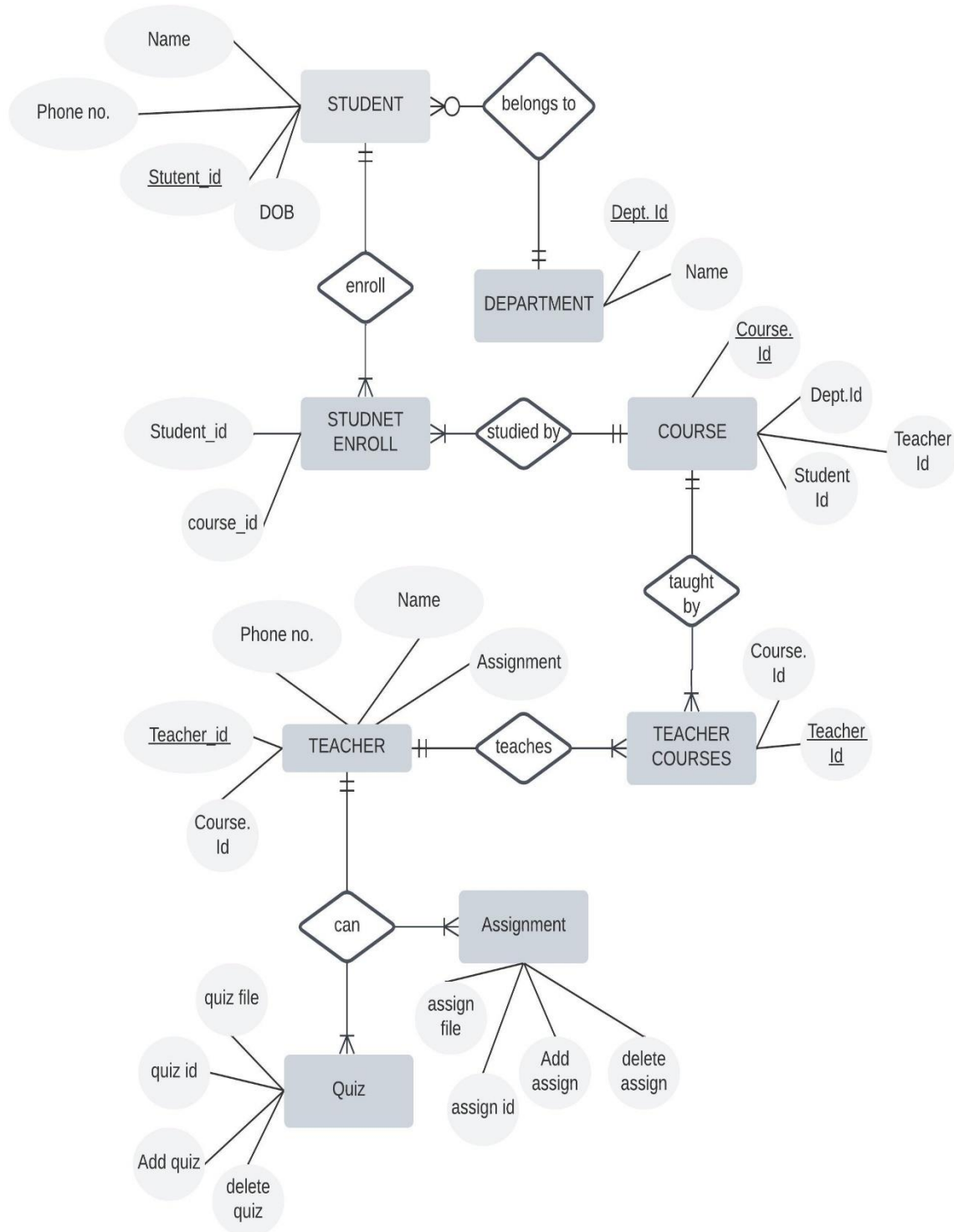


Figure 5.1: Entity Relation Diagram

5.2 Network and Protocol Choice

5.2.1 Network Architecture Choice

The BMS utilizes a scalable and secure client-server architecture to handle the communication between users and the server. This architecture allows for efficient data management and seamless interaction between the client-side and server-side components.

1. Communication Protocols:

HTTP (Hypertext Transfer Protocol) and HTTPS (HTTP Secure) protocols are employed to facilitate communication between the client and the server. HTTPS is implemented to ensure the secure transmission of sensitive data, providing encryption and authentication to prevent unauthorized access and data tampering.

2. Scalability Considerations:

To accommodate the growing user base and increasing data volume, the network architecture is designed to be scalable, allowing for the seamless addition of resources and the distribution of the workload. Load balancing techniques are implemented to evenly distribute incoming traffic and prevent server overload, ensuring consistent performance during peak usage.

3. Security Measures:

The network is secured through the implementation of robust firewalls, intrusion detection systems, and data encryption protocols. Access controls and authentication mechanisms are enforced to verify user identities and regulate access to confidential data. Regular security audits and updates are conducted to identify and mitigate potential vulnerabilities proactively.

4. Performance Optimization:

Performance optimization is achieved through the integration of content delivery networks (CDNs) to reduce latency and enhance the delivery of web content. Caching mechanisms are implemented to store frequently accessed data, minimizing the need for repeated data retrieval and reducing server response times. Additionally, compression techniques are utilized to optimize data transfer and minimize bandwidth usage.

5. Compatibility and Interoperability:

The network architecture and communication protocols are designed to ensure compatibility and interoperability across various devices, browsers, and operating systems. Cross-platform compatibility is maintained to provide users with a consistent and seamless experience, irrespective of their chosen device or platform.

5.3 Choice of Object Middleware

The chosen middleware for the BMS is based on the specific requirements of the application and the need for efficient communication between distributed components. After careful consideration and evaluation, the selected middleware is **RMI** is chosen for its seamless integration capabilities and its ability to facilitate communication between different modules within the BMS. Its robust architecture allows for the transparent invocation of methods on remote objects, enabling the exchange of data and resources across the distributed system. With built-in support for object serialization and deserialization, RMI simplifies the process of transmitting complex data structures, enhancing the overall efficiency and performance of the application.

5.4 User Interface

The user interface (UI) of the BMS is designed to provide an intuitive and engaging experience for all users, including students, teachers, and administrator. It emphasizes user-centric design principles, easy navigation, and accessibility to key features and information. The following aspects are considered in the design of the user interface.

5.3.1 User Interface design

The user interface (UI) design prioritizes intuitive navigation, clear information hierarchy, and responsive layouts to provide an engaging and accessible experience for all users. It incorporates modern design principles, including a consistent color scheme, visually appealing graphics, and easily recognizable icons, ensuring a seamless and visually appealing interface for students, faculty, and administrative staff.

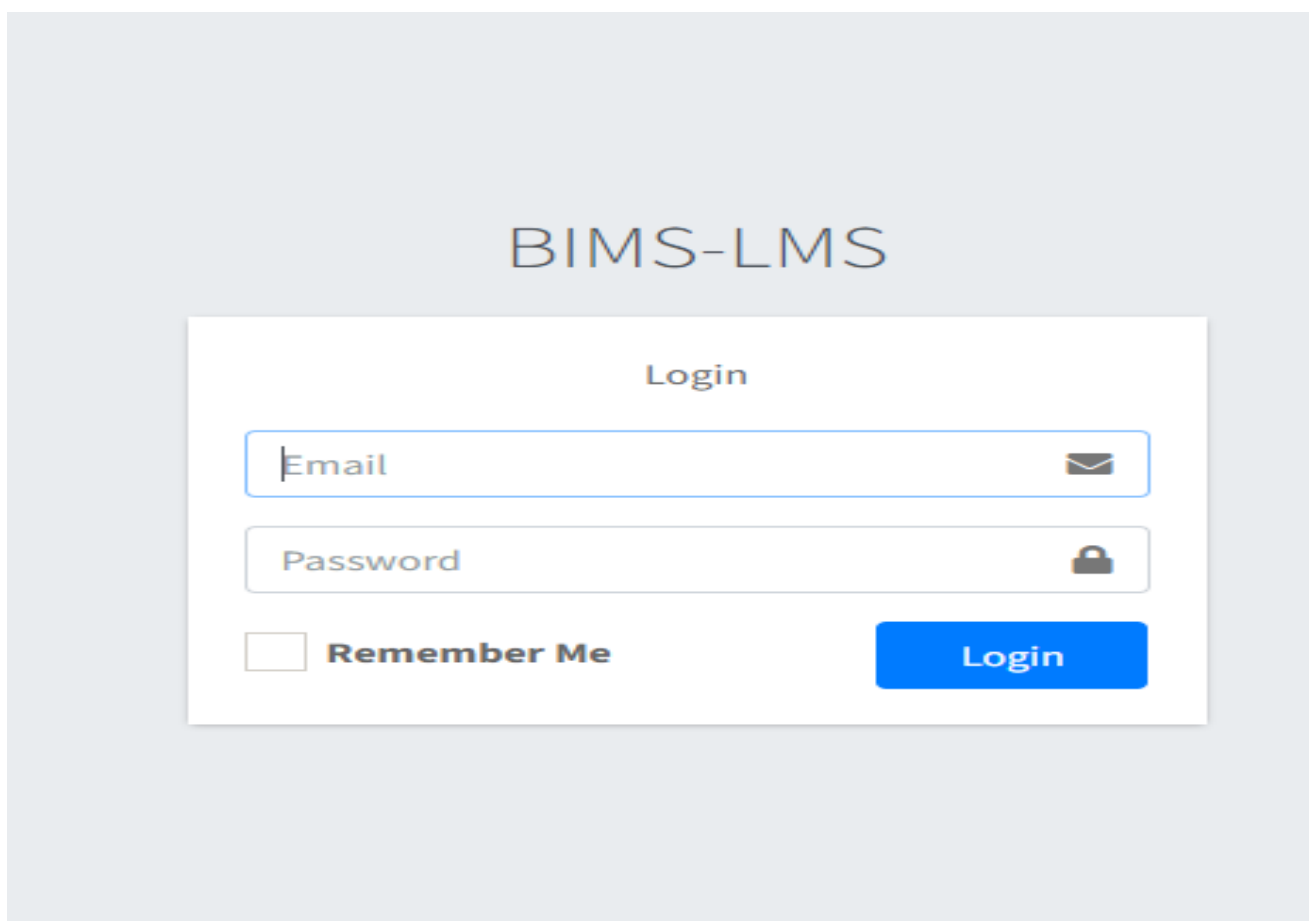


Figure 5.2 Login Interface

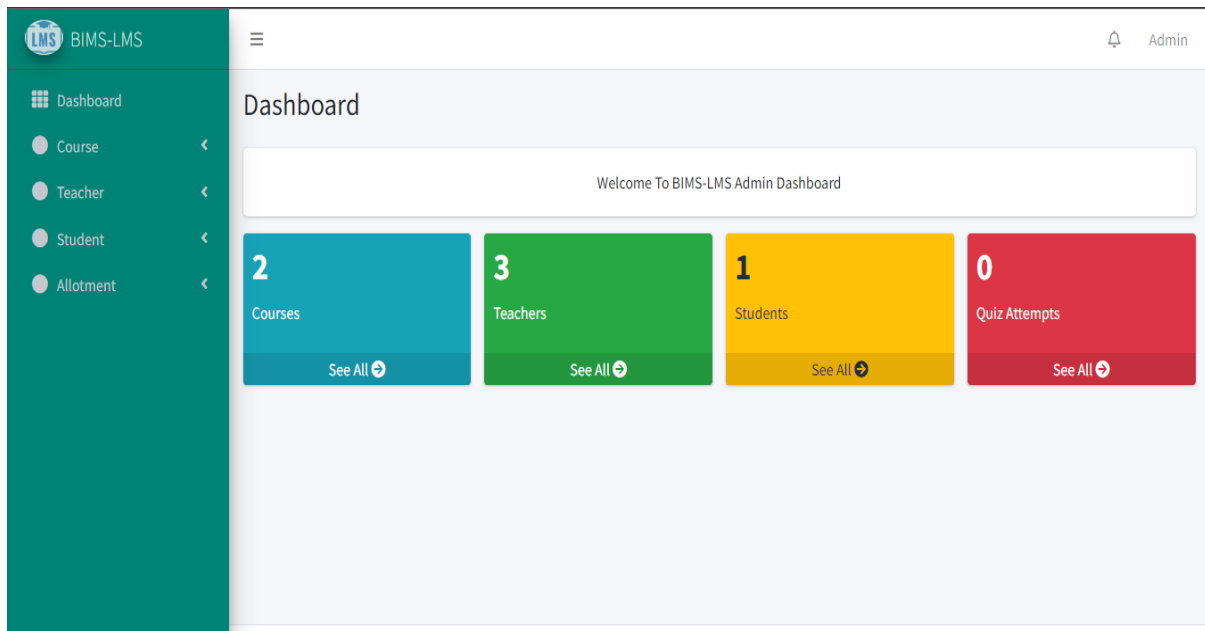


Figure 5.3: Admin Dashboard

5.3.2 Data Base Design

The database design employs a robust and scalable structure to efficiently manage and store various data entities, including student information, course details, faculty records, and administrative data. It utilizes a relational database management system, ensuring data integrity, consistency, and reliability. The database schema is optimized for efficient data retrieval, storage, and manipulation, enabling seamless integration with the application modules and ensuring smooth data processing and management.

Teacher List					
#	Name	Email	Department Name	Status	Action
1	waluraji	xifiwa@mailinator.com	MSC(STATICS)	Inactive	Update Delete
2	subexoky	nytoqum@mailinator.com	MSC(STATICS)	Active	Update Delete
3	gipyd	fana@mailinator.com	MSC(ECONOMICS)	Active	Update Delete

Figure 5.4: Teacher list

#	Name	Code	Description	Action
1	PDC	CS-687	This course is very boring...	Update Delete
2	VP	CS-692	VP is tough	Update Delete

Figure 5.4: Course List

5.3.3 Application Modules

The BMS comprises distinct application modules, each catering to specific functionalities and user requirements. The user authentication module ensures secure user registration, authentication, and authorization processes, guaranteeing the privacy and security of user accounts and information. The course management module facilitates efficient management of course materials, assignments, and schedules, providing faculty members with a comprehensive tool set for organizing and delivering course content. The communication module enables seamless interaction and information exchange among students, faculty, and administrative staff, fostering effective communication channels and collaboration within the academic community.

The administrative module offers administrative staff the necessary tools for managing student records, and institutional resources, streamlining administrative processes and ensuring data accuracy and accessibility.

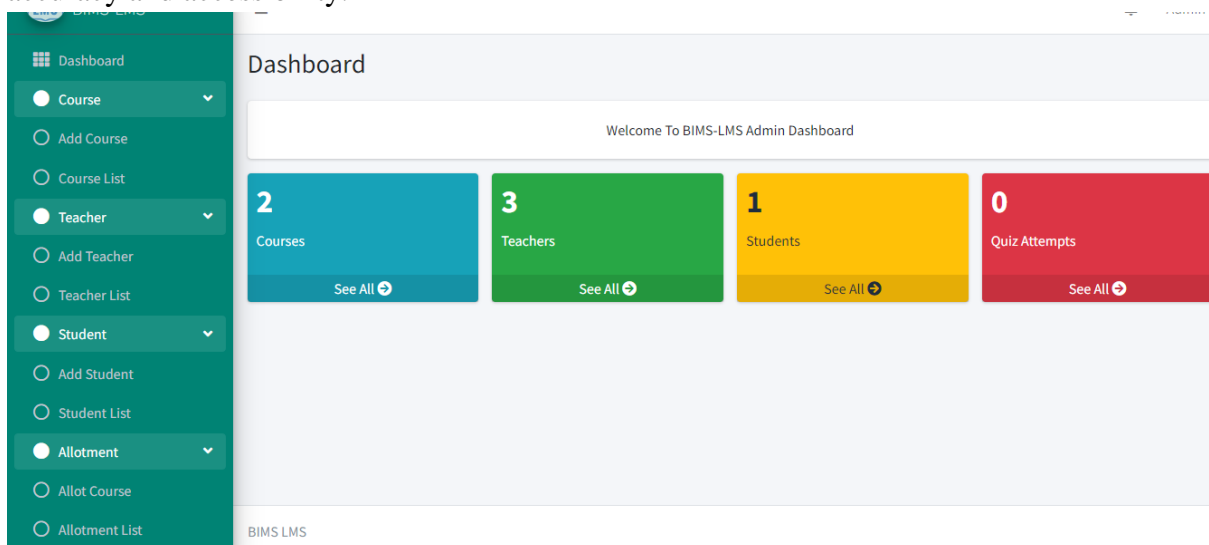


Figure 5.3: Modules of the Admin

5.3.4 Data Flow Integration

The BMS comprises distinct application modules, each catering to specific functionalities and user requirements. The user authentication module ensures secure user registration, authentication, and authorization processes, guaranteeing the privacy and security of user accounts and information. The course management module facilitates efficient management of course materials, assignments, and schedules, providing faculty members with a comprehensive tool set for organizing and delivering course content. The communication module seamless interaction and information exchange among students, faculty, and administrative staff, fostering effective communication channels and collaboration within the academic community. The administrative module offers administrative staff the necessary tools for managing student records, financial data, and institutional resources, streamlining administrative processes and ensuring data accuracy and accessibility.

5.3.5 Security Measures

The system design integrates robust security measures to protect sensitive data and prevent unauthorized access. It implements encryption protocols, multi-factor authentication mechanisms, and role-based access control, ensuring data confidentiality, integrity, and availability. Regular security audits and updates are conducted to identify and mitigate potential security vulnerabilities, ensuring compliance with industry standards and data protection regulations. Security protocols are implemented at both the application and database levels, providing a comprehensive and multi-layered security framework for the BIMS portal website.

Chapter 6: Testing and Evaluation

In system testing, we engage in a comprehensive evaluation of the entire system to guarantee its compliance with specified requirements and expectations. This testing phase encompasses the examination of the integrated system to confirm that it functions as intended and fulfills both functional and non-functional requirements. By conducting system testing, we ensure that the software operates as anticipated and aligns with the business and technical requisites of the system. To effectively carry out system testing, we typically undertake the following activities:

6.1 Verification

Verification aims to proactively identify and address defects within the system. This includes:

6.1.1 Code Defects:

Verification processes such as code reviews and inspections are employed to catch and rectify coding errors, ensuring the integrity of the system's source code.

6.1.2 Functional Defects:

Testing procedures, including unit testing, integration testing, and system testing, are conducted to uncover any functional discrepancies or deviations from expected behavior.

6.1.3 Usability Defects:

User interface components are reviewed to identify and address any usability issues that might impact the overall user experience.

6.2 Validation

System testing also aims to validate that the software adequately meets the user requirements. By approaching the testing process from a user perspective, we evaluate the system's functionality, usability, and overall user experience. This validation ensures that the system fulfills its intended purpose and meets the expectations of the end-users.

6.3 Module / Unit Testing

System testing involves evaluating the system holistically to ascertain that all components and modules have been integrated correctly. By examining the system as a unified entity, we verify that various interconnected elements work together seamlessly, ensuring the system's.

6.4 Integration Testing

Integration testing is an integral part of system testing, wherein we focus on verifying the proper integration of all system modules and components. This testing phase ensures that the individual elements of the system interact harmoniously and exchange information accurately, enabling smooth data flow and overall system cohesiveness. The primary objective of system testing is to ensure that the software satisfies all requirements and specifications, exhibits stability and reliability, and performs as expected in the target environment. By subjecting the system to rigorous testing, we aim to identify any potential issues or discrepancies and address them before the software is deployed to end-users.

6.5 Test Cases

Table 6.1 TC001

Test Case ID	TC001
Test Case Description	Test the login functionality of system.
Test Case Steps	<ul style="list-style-type: none">• Navigate to the login page of the system• Enter valid user ID and password in the respective fields• Click on the "Login" button• Verify that the system logs in the user successfully and redirects them to the user dashboard page
Test Case Data	User email: "Aqsasheykh00@gmail.com" Password: "1234"
Test Case Expected Result	The system should authenticate the user with the provided credentials and redirect them to the user dashboard page
Test Case Actual Result	The system logs in the user successfully and redirects them to the user dashboard page.
Test Case Pass/Fail	Pass

Table 6.2 TC002

Test Case ID	TC002
Test Case Description	Test the login functionality with invalid credentials
Test Case Steps	<ul style="list-style-type: none">• Navigate to the login page of the system• Enter an invalid user ID or password in the respective fields• Click on the "Login" button• Verify that the system displays an error message indicating that the credentials are invalid
Test Case Data	User email: "aqsa@gmail.com" Password: "1234"
Test Case Expected Result	The system should generate error message, must include @ in email address. If user email is not registered it will generate error saying "These credentials don't match our record"
Test Case Actual Result	The system displays error saying "These credentials don't match our record". And prevents user from login.
Test Case Pass/Fail	Pass

Table 6.3 TC003

Test Case ID	TC003
Test Case Description	Test the login functionality with empty user ID.
Test Case Steps	<ul style="list-style-type: none"> • Navigate to the login page of the system • Leave the user ID field empty • Enter a valid password in the password field • Click on the "Login" button • Verify that the system displays an error message indicating that the user ID field is required.
Test Case Data	User email: " " Password: "password123"
Test Case Expected Result	The system should display an error message indicating that the user email field is required and prevent the user from logging in.
Test Case Actual Result	The system displays an error message indicating that the user email field is required and prevents the user from logging in
Test Case Pass/Fail	Pass

Table 6.4 TC004

Test Case ID	TC004
Test Case Description	To verify if the user is able to download the task successfully.
Test Case Steps	<ul style="list-style-type: none"> • Login to the system using valid credentials. • Navigate to the task section. • Click on the download button for the desired task. • Check if the download starts automatically or the user is prompted to save the file. • Verify if the downloaded file is the same as the original task file
Test Case Data	Valid login credentials task to be downloaded.
Test Case Expected Result	The task should be downloaded successfully without any errors. The downloaded file should be the same as the original task file.
Test Case Actual Result	The task is downloaded successfully. The downloaded file matches the original task file.
Test Case Pass/Fail	Pass

Table 6.5 TC005

Test Case ID	TC005
Test Case Description	To verify if the user is able to submit the task successfully.
Test Case Steps	<ul style="list-style-type: none"> • Login to the system using valid credentials. • Navigate to the task section. Click on the submit button for the desired task. Browse and select the task file to be submitted. • Click on the submit button. • Verify if the task is submitted successfully.
Test Case Data	Valid login credentials. Task to be submitted
Test Case Expected Result	The task should be submitted successfully without any errors. The user should receive a confirmation message for the successful submission.
Test Case Actual Result	The task is submitted successfully. The user receives a confirmation message for the successful submission.
Test Case Pass/Fail	Pass

Table 6.6 TC006

Test Case ID	TC006
Test Case Description	Add a new task and check if it is successfully added.
Test Case Steps	<ul style="list-style-type: none"> • Open the task assignment page • Click on the "Add Task" button • Fill in the required details of the task (title, description, deadline, priority) • Click on the "Save" button • Go to the task list and check if the task is added successfully
Test Case Data	Title: "Complete project proposal" Description: "Write a proposal document for the upcoming project" Deadline: "2023-04-15"
Test Case Expected Result	The new task should be added successfully with the given details. A success message should be displayed. The new task should be visible in the task list

Test Case Actual Result	The new task is added successfully with the given details. A success message is displayed The new task is visible in the task list
Test Case Pass/Fail	Pass

Table 6.7 TC007

Test Case ID	TC0007
Test Case Description	Update an existing task and check if it is successfully updated.
Test Case Steps	<ul style="list-style-type: none"> • Open the task assignment page • Select an existing task from the task list • Click on the "Edit" button • Update the required details of the task (title, description, deadline) • Click on the "Save" button • Go to the task list and check if the task is updated successfully.
Test Case Data	Task Title: "Complete project proposal" Updated Title: "Finalize project proposal" Updated Description: "Review and finalize the project proposal document" Updated Deadline: "2023-04-20"
Test Case Expected Result	<ol style="list-style-type: none"> 1. The existing task should be updated successfully with the given details. 2. A success message should be displayed 3. The updated task should be visible in the task list with the new details.
Test Case Actual Result	The existing task is updated successfully with the given details. A success message is displayed The updated task is visible in the task list with the new details
Test Case Pass/Fail	Pass

Table 6.8 TC008

Test Case ID	TC008
Test Case Description	Delete an existing task and check if it is successfully deleted
Test Case Steps	<ul style="list-style-type: none">• Open the task assignment page• Select an existing task from the task list• Click on the "Delete" button• Confirm the deletion• Go to the task list and check if the task is deleted successfully
Test Case Data	Task Title: "Finalize project proposal"
Test Case Expected Result	The existing task should be deleted successfully. A success message should be displayed The deleted task should not be visible in the task list
Test Case Actual Result	The existing task is deleted successfully A success message is displayed The deleted task is not visible in the task list
Test Case Pass/Fail	Pass

Chapter 7: Conclusion and Future Work

7.1 Summary

The BMS is a multi-environmental mobile application that uses the web services of Moodle, an open-source Learning Management System (LMS). The app provides functionality to students, teachers, and directors, allowing them to share learning material, view assignments, quizzes and monitor activities. The development of an LMS was motivated by the need for online learning, particularly in light of the COVID-19 outbreak. The webapp is designed to facilitate peer interaction, collaboration, and knowledge construction. It provides an avenue for classroom materials or activities to be shared easily and enables teachers and students to interact outside of the classroom. The project scope involves a software application used to plan, implement and assess a specific learning process. The LMS server performs the base functionality, and a user interface is operated by instructors, students, and administrators. The system allows for efficient management of user registration, study content, calendars, user access, communication, certifications, and notifications. The main features of the project include uploading courses, managing course activities, sending reminders and notifications, conducting assessments, maintaining training records, and tracking and checking the skill set of team members. The project's main objective is to customize the director's screen on the existing LMS and design the app according to frequently used features. System testing is performed to ensure that the software meets its requirements and specifications. Various test cases are used to verify that the software is working as expected and meets the business and technical requirements of the system. Overall, the BMS provides a comprehensive solution for online learning management. Its features and benefits make it a valuable tool for students, teachers, and directors.

7.2 Conclusion

In conclusion, the BMS is an ambitious undertaking aimed at revolutionizing the learning experience for students, teachers, and directors. The project's objective was to develop website application that serves as a powerful learning management system, seamlessly integrating with the open-source LMS Moodle. Throughout the project, careful consideration was given to the needs and requirements of all stakeholders involved in the learning process.

The BMS offers a user-friendly interface, ensuring ease of use and accessibility for instructors, students, and administrators.

The system provides a range of essential features, including course management, communication and collaboration tools, assessment capabilities, and skill tracking functionalities. By implementing the BMS, educational institutions can expect numerous benefits. Instructors gain a streamlined platform for managing course content, organizing activities, and providing timely feedback. Students benefit from convenient access to learning materials, interactive learning experiences, and improved communication channels.

Directors and administrators can efficiently oversee the learning process, track training records, and make informed decisions based on comprehensive data. The successful implementation of the BMS project relies on effective testing, ensuring the system functions as intended and meets all requirements and specifications. System testing is crucial to verify the integration of components, validate user requirements, and guarantee the stability and reliability of the software. The BMS project represents a significant step forward in leveraging technology to enhance the educational landscape. By embracing this innovative learning management system, institutions can foster a more interactive and engaging learning environment, enabling students to reach their full potential. As the project concludes, it is important to acknowledge the collaborative efforts of the development team, project managers, instructors, and students who have contributed to its success. The BMS stands as a testament to their dedication and commitment to advancing education through cutting-edge technology. Moving forward, continuous improvement and updates to the BMS will be essential to adapt to evolving educational needs and leverage emerging technologies. With its strong foundation and comprehensive feature set, the BMS is poised to become a cornerstone in the educational landscape, empowering students and educators alike to embrace a modern and transformative learning experience.

7.3 Future Work

Here are some potential points for future work and updates to the BMS App:

- 7.3.1 Integration with other educational tools and platforms:** The BMS could be updated to integrate with other educational tools and platforms, such as virtual whiteboards, video conferencing software, and online quiz and assessment tools. This would provide users with a more comprehensive and seamless learning experience.
- 7.3.2 Enhanced analytics and reporting:** The app could be updated to provide more detailed analytics and reporting on student performance and engagement. This would allow teachers and directors to better understand how students are interacting with the course material and identify areas for improvement.
- 7.3.3 Improved accessibility:** This Ensure the BIMS Management System is compatible with screen readers. Provide alternative text for images, descriptive links, and proper heading structures for easy navigation. Make sure all functionalities are operable using a keyboard alone. Users should be able to navigate through the interface, fill forms, and interact with elements without a mouse.
- 7.3.4 Expanded course content:** The app could be updated to support a wider range of course content, including multimedia content such as videos, audio recordings, and interactive simulations. This would provide students with a richer and more engaging learning experience.

References

1. Visual Studio Code: Work together using the free Live Share extensions. Edit and debug your code in real-time. Updated November,2023.
https://code.visualstudio.com/updates/v1_85
2. XAMPP: XAMPP with PHP 7.2.0 provides support for creating and manipulating databases in MariaDB and SQLite, among others.
https://www.apachefriends.org/blog/new_xampp_20171220.html
3. Lucid Chart: Diagrams, flowchart maker <https://www.lucidchart.com/pages/>
4. Laravel: A laravel version 10.x is a web framework provides a structure and starting point for creating your application. The Artisan development server, your application will be accessible in your web browser at http://localhost:8000. <https://laravel.com/>
5. Composer: Composer is a tool for dependency management in its latest version requires PHP 7.2.5. It allows you to declare the libraries your project depends on and it will manage (install/update) them for you. <https://getcomposer.org/download/>