



Assiut University
Faculty of Computers and Information

Smart Attendance System

Software Requirements Specification

Version 1.0 approved

Under supervision of:

Prof. Dr. Taysir Hassan Abdel Hamid

TA: Amjad Dife

Prepared by:

- 1. Doaa Ahmed Gharib [IS]**
- 2. Fady Essam Saeed [CS]**
- 3. Aya Ahmed Elsaied Mohamed [IS]**
- 4. Dina Ahmed Ali [IS]**
- 5. Gorg Hosny Alfy [IS]**
- 6. Fady Karmy Khella [IS]**
- 7. Yasmeen khaled farhan [IS]**
- 8. Keroles Ibraheem Nageh [IS]**

2023-2024

Table of Contents

Table of Contents	ii
Revision History.....	ii
1. Introduction.....	1
1.1 Purpose.....	1
1.2 Intended Audience and Reading Suggestions	2
1.3 Product Scope	2
1.4 References	4
2. Overall Description	6
2.1 Product Perspective	6
2.2 Product Functions	6
2.3 User Classes and Characteristics	9
2.4 Operating Environment	10
2.5 Design and Implementation Constraints	11
2.6 User Documentation	12
2.7 Assumptions and Dependencies.....	12
3. External Interface Requirements	13
3.1 User Interfaces	13
3.2 Hardware Interfaces	24
3.3 Software Interfaces	24
3.4 Communications Interfaces	24
4. System Features	25
4.1 E-R Diagram	25
4.2 Use Case Diagram	25
4.3 Use Case Scenarios	26
4.4 Activity Diagram	34
5. Other Nonfunctional Requirements	40
5.1 Performance Requirements	40
5.2 Safety Requirements	40
5.3 Security Requirements	40
5.4 Software Quality Attributes	40
5.5 Business Rules	41
6. Other Requirements	42
Appendix A: Glossary	43
Appendix B: Analysis Models	45
Appendix C: To Be Determined List	46

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

- Smart attendance systems serve several purposes, depending on the context in which they're used. Here are some of the key benefits:

- *Increased Efficiency and Accuracy:*

- Automation: Smart systems eliminate manual **attendance recording**, saving time and reducing errors.
- Real-time data: Attendance data is captured and processed instantly, providing immediate insights.
- Reduced paperwork: Less reliance on paper-based attendance sheets means less clutter and easier record keeping.

- *Improved Security and Transparency:*

- **Biometric** verification: Technologies like **facial recognition** or **fingerprint scanners** ensure accurate identification, preventing "buddy punching" or unauthorized access.
- Time tracking: Detailed records of arrival, departure, and breaks help ensure accurate payroll and compliance with labor regulations.
- Enhanced security: Controlled access points and real-time monitoring can improve overall security in workplaces or educational institutions.

- *Enhanced Management and Analysis:*

- Detailed reports: Attendance data can be analyzed to identify trends, track individual performance, and optimize scheduling.
- Data-driven decision making: Insights from attendance data can inform policy changes, resource allocation, and employee engagement strategies.
- Improved communication: Real-time attendance updates can be shared with parents, managers, or other stakeholders.

- *Specific Applications:*

- Schools and universities: Track student attendance, identify absenteeism patterns, and improve engagement.
- Workplaces: Ensure accurate payroll, monitor employee punctuality and productivity, and improve time management.
- Events and
- Conferences: Manage registration, track attendee participation, and gain insights into event effectiveness.

Overall, smart attendance systems offer a range of benefits for organizations and individuals by streamlining processes, improving accuracy, enhancing security, and providing valuable data for informed decision-making.

1.2 Intended Audience

The audience of this system will be:

1. Students
2. Faculty members
3. Registration office.

This project will be managed by the college's staff, and will be developed by the student who are in the fourth year along with the staff and other specialized people in the technology, such as programming, web design and others.

1.3 Project Scope

Developing a smart attendance system involves various considerations and decisions to ensure a successful project. Here's a breakdown of the key aspects to define the scope:

1. System Goals and Requirements:

- The primary purpose of the system is to avoid the role of paperwork permanently and to automate all procedures equally in all organizations and it is more efficient and reliable method of recording attendance. Traditional methods, such as manual attendance sheets or sign-in registers. This way is saving time and effort spent in attendance and departure processes in the country. Achieving social distancing and privacy for people working under the current global conditions and Covid-19.
- Core features include user registration, attendance marking (facial recognition, fingerprint scanner, etc.), time tracking, and data reporting.

2. Target Users and Use Cases:

The target users include: employees, students, teachers, event organizers, administrators working with different scenarios such as, marking attendance in a classroom, at workstations, during events, or remotely.

3. Technology Stack and Architecture:

- The type of attendance marking technology that we used is: facial recognition, fingerprint scanners, and **QR codes**, with high accuracy and security, reduced cost, and user acceptance.
- Hardware and software components that are needed are mobile apps, servers, databases, and any necessary integrations.

4. Data Management and Reporting:

- Attendance data will be stored and secured by **Cloud storage**, local servers, or a hybrid approach.
- Kind of reports and analytics that are needed such as: individual attendance records, trends, lateness reports, overtime calculations, etc.
- The data will be **visualized** and presented to users in: reports, **dashboards**, and mobile app interfaces.

5. Project Deliverables:

The final products or outcomes of the project are a functional smart attendance system, documentation, training materials, and ongoing support.

6. Scalability and Future Considerations:

The system has been considered to accommodate a future growth or changes in requirements according to the flexible design and architecture of the system.

- Additional Considerations:

- Privacy concerns: Address data privacy concerns with transparent policies and user consent mechanisms.
- Accessibility: Ensure the system is accessible to users with disabilities.
- Ethical considerations: Implement the system ethically and responsibly, avoiding biases or discrimination.

1.4 Reference

- [1] Prof. Dr. Vijay Mane, Shafaque Sheikh, Mrunal Shinde, Prasad Shevale, Prem Shejole, Sahil Salve, “Smart attendance using face recognition”, 2023.
- [2] Dwi Sunaryono, Joko Siswantoro, Radityo Anggoro, "An android based course attendance system using face recognition", 2021.
- [3] Sikandar Khan, Adeel Akram, Nighat Usman “Real time automatic attendance system for face recognition using Face API and OpenCV”, 2020.
- [4] Kaneez Laila Bhatti, Laraib Mughal, Faheem Yar Khuhawar, Sheeraz Ahmed Memon, “Smart attendance management system using face recognition”, 2018.
- [5] Raneem Alghamdia, Wafaa Alsubaieb, Renad Alharthic, Wedyan Alghamdid, Dabiah A. Alboaneene and Nourah F. Alqahtanif , “A Smart Attendance System for Imam Abdulrahman Bin Faisal University Using Facial Recognition”, 2020.

- [6] Vivek Seelam, Akhil kumar Penugonda, B. Pavan Kalyan, M. Bindu Priya, M. Durga Prakash, “Smart attendance using deep learning and computer vision”, 2021.
- [7] Jarugula.Vamsikrishna, Kollipara.Anudeep, L.Jegan Antony Marcilin, V. Balamurugan, “An Advanced Attendance Marking system using facial Recognition”, 2019.
- [8] Habibulrahman Azizi, Numan Amin, Sayed Shoaibullah Shams, “Smart Attendance using Real-Time Face Recognition”, 2022.
- [9] Khawla Alhanaee, Mitha Alhammadi, Nahla Almenhali, Maad Shatnawi “Face Recognition Smart Attendance System using Deep Transfer Learning”, 2021.

2. Overall Description

2.1. Product Perspective

The current attendance management process in universities faces challenges in terms of efficiency, accessibility, and user experience. There is a need for a comprehensive solution that addresses these issues and leverages advancements in web technology, mobile applications, and AI. The proposed system aims to overcome existing limitations by providing a streamlined and user-friendly approach to attendance management, enhancing accessibility, and offering improved interfaces for a more efficient and satisfactory user experience.

2.2. Product Functions

The attendance management system will encompass a variety of functions to facilitate efficient operation:

1. Verification Mechanisms:

- Verification of fingerprint
- Verification of Face recognition
- Scanning QR code

2. Dashboard Display:

- Display a dashboard tailored for Professor and Student roles

3. Course Management:

- Add a new course
- Update existing course details
- Delete a course

4. Lecture Management:

- Add a new lecture
- Update existing lecture details
- Delete a lecture

5. QR Code Generation:

- Generate QR codes for various purposes

6. User Authentication:

- Sign up for a new account
- Log in to the system

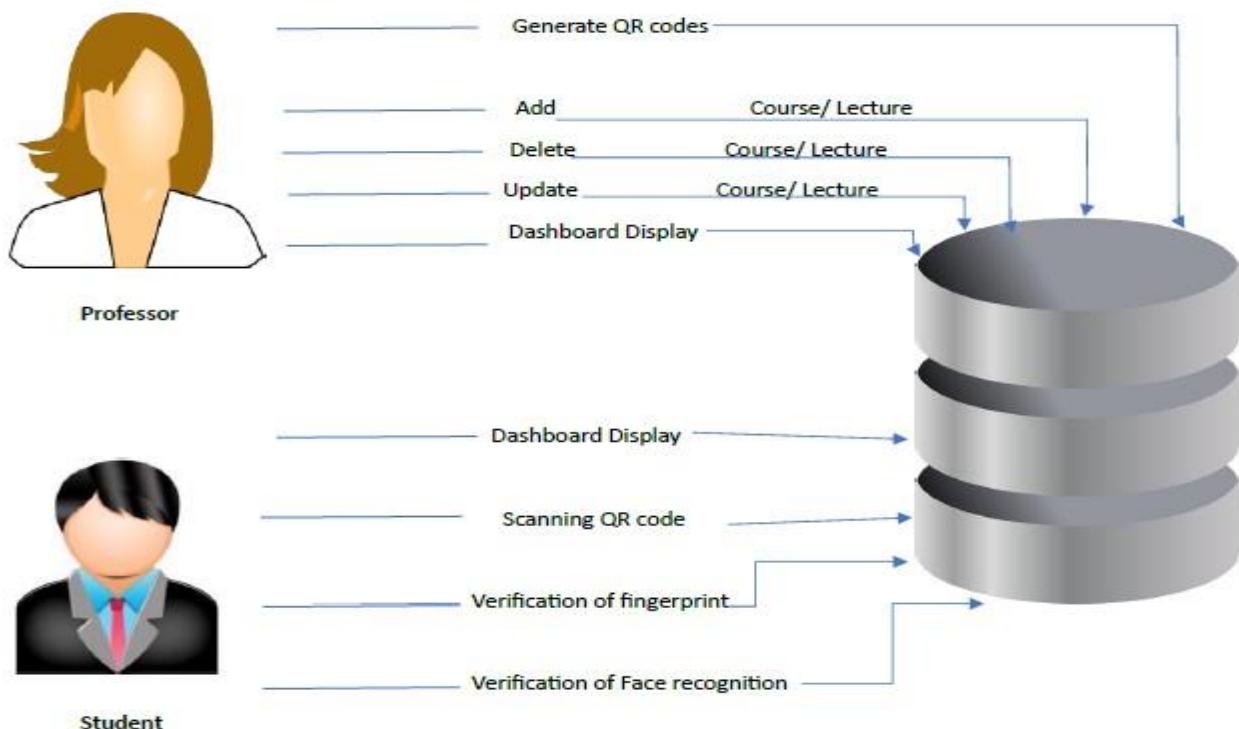


Figure 2.1.1 High-Level Architecture

Fig (1): User Authentication

Functional Requirements

1. Verification Mechanisms:

Function #1	Details
Input	Biometric input (fingerprint), Facial image, QR code image
Output	Verification result (True/False), Decoded information (for QR code)
Process	Biometric matching process (for fingerprint and facial recognition), Scanning process (for QR code)

2. Dashboard Display:

Function #2	Details
Input	User role (Professor/Teaching Assistant/Student)
Output	Customized dashboard for the specified user role
Process	Data retrieval and presentation

3. Course Management:

Function #3	Details
Input	Course details, Course ID and updated details (for updates and deletions)
Output	Confirmation of successful addition, update, or deletion
Process	Database update

4. Lecture Management:

Function #4	Details
Input	Lecture details, Lecture ID and updated details (for updates and deletions)
Output	Confirmation of successful addition, update, or deletion
Process	Database update

5. QR Code Generation:

Function #5	Details
Input	Data for QR code
Output	Generated QR code
Process	QR code generation algorithm

6. User Authentication:

Function #6	Details
Input	User details (for sign up), User credentials (for log in)
Output	Confirmation of successful registration (for sign up), Access granted or denied (for log in)
Process	User registration process (for sign up), Authentication process (for log in)

2.3. User Classes and Characteristics

2.3.1 User Classes

There are two user levels in smart attendance System.

I. Professor

II. Student

2.3.2 Characteristics of User Classes

Professor:-

Professors engaging with the attendance management system are expected to possess a moderate to high level of technical proficiency. This proficiency enables them to navigate the system effectively and leverage its features for course management, attendance tracking, and data analysis. A key requirement for professors is the ability to customize their dashboard, tailoring it to emphasize functionalities relevant to their responsibilities. This includes tools for data analysis and reporting, allowing professors to derive valuable insights from attendance records and monitor student participation. Additionally, professors should have comprehensive capabilities for managing courses and lectures within the system. Professors are also equipped with the functionality to generate QR codes. This feature empowers them to create QR codes for various purposes, such as facilitating attendance tracking during lectures or generating unique identifiers for course-related activities. The QR code generation capability adds a valuable tool to the professor's toolkit for efficient and secure management of attendance and course-related tasks.

Student: -

Students interacting with the attendance management system prioritize a user-friendly interface that simplifies the process of accessing attendance records, course information, and relevant updates. Their dashboard should provide a comprehensive overview of their courses, attendance status, and upcoming lectures. Students may benefit from functionalities such as QR code scanning for attendance tracking during lectures. To enhance the user experience, customizable **notification preferences** would allow students to receive alerts tailored to their individual needs and preferences.

2.4. Operating Environment

Hardware Requirements:

- The system should be compatible with commonly used computing devices such as laptops, desktops, and tablets.
- Minimum hardware specifications include a modern processor, sufficient RAM, and suitable storage capacity for efficient performance.

Software Requirements:

- The system is compatible with major operating systems, including Windows, macOS, and Linux.
- Web browser compatibility should include but is not limited to Google Chrome, Mozilla Firefox, and Microsoft Edge.
- Mobile applications are designed for compatibility with Android operating systems.

Networking:

- Reliable internet connectivity is required for accessing the system's web-based functionalities.
- The system should be compatible with standard networking protocols for data transmission.

Display Devices:

- Professors should have access to display devices such as monitors or TVs with sufficient screen size and clarity.
- The display devices may be equipped with projectors to enhance visibility in larger classrooms.

Connectivity:

- Ensure compatibility with common display interfaces such as HDMI or VGA for seamless connectivity between the system and the display devices.

QR Code Visibility:

- Consider factors such as font size, color contrast, and resolution to ensure QR codes are clearly visible to students, even in varying lighting conditions.

2.5. Design and Implementation Constraints:

- **Compatibility:** the system must be compatible with a variety of web browsers and operating systems.
- **Security:** stringent security measures must be implemented to protect sensitive student data and ensure the integrity of attendance records.
- **Scalability:** the system should be scalable to accommodate growth in the number of users and volume of attendance data.

2.6. User Documentation:

2.6.1. User Manual:

- *The user manual shall be provided to guide users on how to use the System effectively.*
- *The manual shall include step-by-step instructions for:*
 1. Logging in and accessing the system.
 2. Marking on attendance records.
 3. Viewing attendance records.
 4. Any other relevant functionalities.

The manual shall be written in clear, concise and simple understandable language.

2.6.2. Online Help:

- An online help system shall be integrated within the System for quick access to assistance.
- The help system shall provide contextual guidance and explanations for various features and functionalities.

2.7 Assumptions and Dependencies:

- The successful operation of the Smart Attendance System depends on reliable internet connectivity within the institution.
- The system must support English language.
- The system must work on different smart devices and different browsers.

3. External Interface Requirements

3.1 User Interfaces

3.1.1. Web design

Smart Attendance

In light of the need for social distancing following Covid-19 we have designed a system that addresses the problem of student attendance in faculty. They were utilizing a paper-based approach to keep track of their attendance. For students to take attendance, a (face recognition & fingerprint & QR code) will be provided. The professor is in charge of determining whether or not all of the students in the group or class are present, To authenticate their attendance and exit.

REGISTER

Features

- Social Distancing**: Social distancing after Covid-19 and not exchanging tools (absence paper or pens)
- Regular Manner**: Facilitate the process of taking absence student and to show it the professor in a regular manner
- Detect Location**: Not taking the presence of an absent student who is not present in the lecture (detect location)

Services

- Fully Responsive Admin Panel**: Flexible Admin panel and user friendly administrative interface to manage almost everything. Every instructor & student consist of his/her own access to the back-end to see the previous records and history of their attendance and other activities.
- Highly Interactive Dashboard**: Easy-to-use administrative panel with a highly efficient dashboard that aims to provide a quick overview of the latest happenings and keep you updated about the status of the app.
- User Management**: User management consist of instructors & students and their profiles accordingly. Add new instructor or students or edit details of profile and take any action from back-end.
- Reporting Management**: The part of management that enables the instructor to access and view the reports of the student's activity of check-in and check-out on monthly and individually basis.
- Optional Manual Attendance System**: This part is optional and to mark attendance manually in case if any student forgot to mark his/her attendance.

About Us

Welcome To Our Website!

Take attendance in no time! The first ever automatic student attendance tracker which is secured and fully monitored web-based system.

Learn More

Contact

Name: _____

Email: _____

Phone: _____

Subject: _____

Message: _____

SEND

Smart ATTENDANCE

Copyright © 2024 - Attendance_AI
Privacy Policy

Fig (2): Intro page

Smart Attendance

SMART ATTENDANCE

In light of the need for social distancing following Covid-19 we have designed a system that addresses the problem of student attendance in faculty. They were utilizing a paper-based approach to keep track of their attendance. For students to take attendance, a (face recognition & fingerprint & QR code) will be provided. The professor is in charge of determining whether or not all of the students in the group or class are present, To authenticate their attendance and exit.

REGISTER

Fig (3): First page

Smart Attendance

Features

- Social Distancing**: Social distancing after Covid-19 and not exchanging tools (absence paper or pens)
- Regular Manner**: Facilitate the process of taking absence student and to show it the professor in a regular manner
- Detect Location**: Not taking the presence of an absent student who is not present in the lecture (detect location)

Fig (4): Features

Smart Attendance

Services

- Fully Responsive Admin Panel**: Flexible Admin panel and user friendly administrative interface to manage almost everything. Every instructor & student consist of his/her own access to the back-end to see the previous records and history of their attendance and other activities.
- Highly Interactive Dashboard**: Easy-to-use administrative panel with a highly efficient dashboard that aims to provide a quick overview of the latest happenings and keep you updated about the status of the app.
- User Management**: User management consist of instructors & students and their profiles accordingly. Add new instructor or students or edit details of profile and take any action from back-end.
- Reporting Management**: The part of management that enables the instructor to access and view the reports of the student's activity of check-in and check-out on monthly and individually basis.

Fig (5): Service

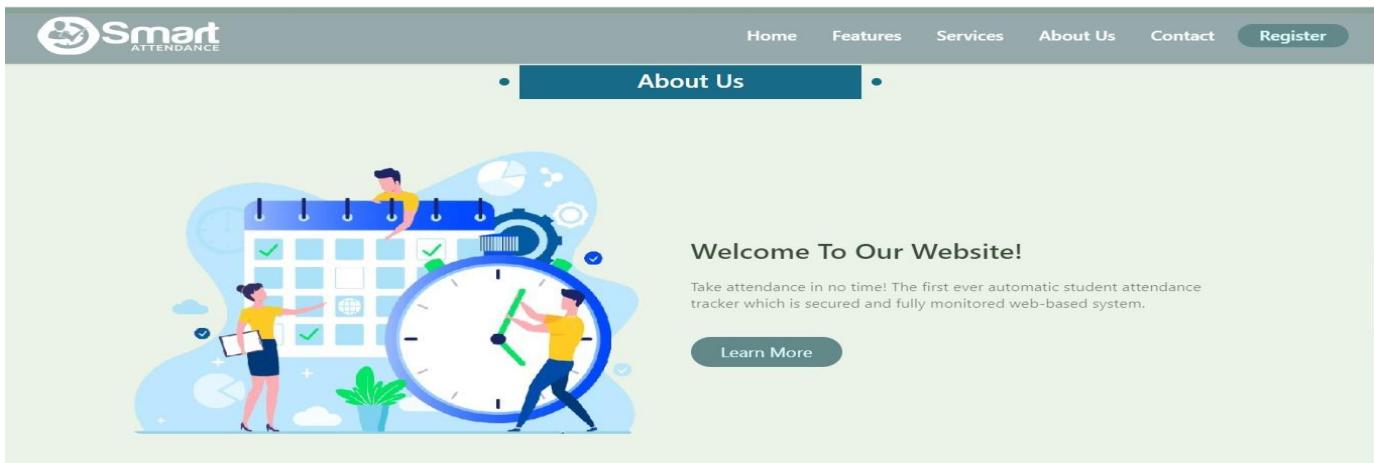


Fig (6): First about us

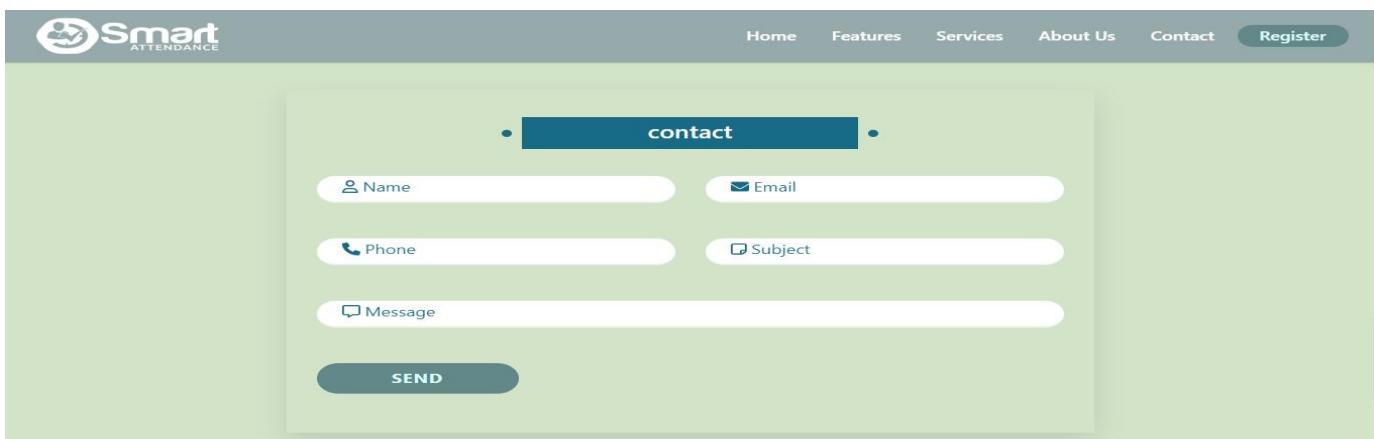
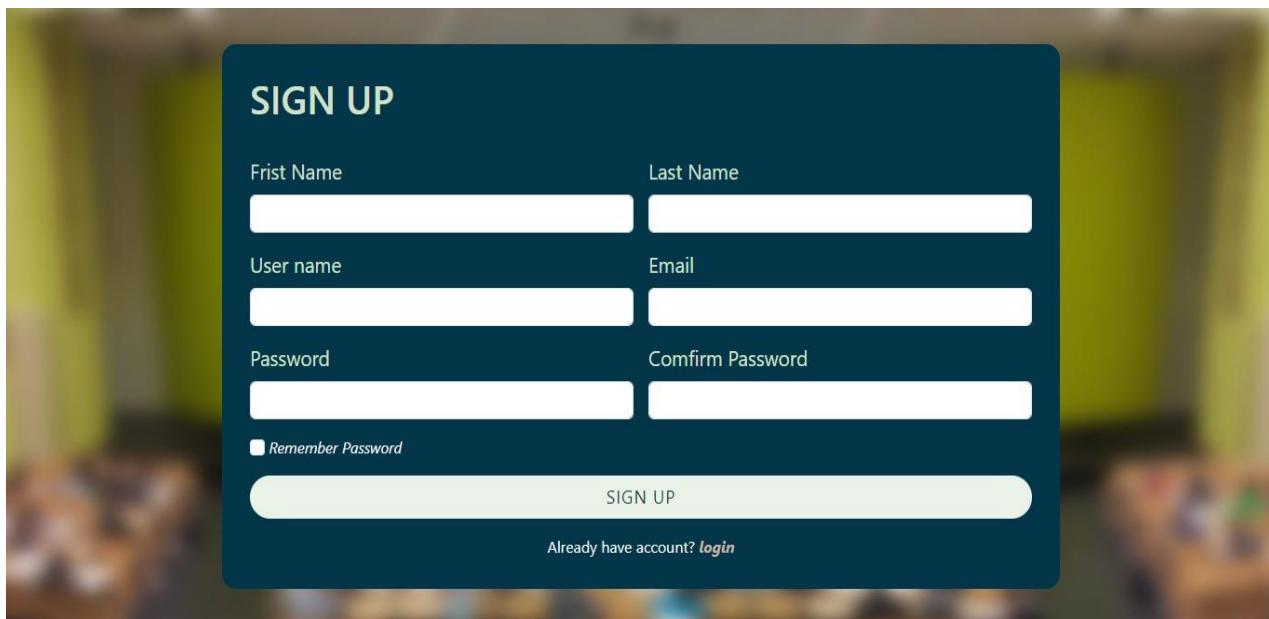


Fig (7): First contact

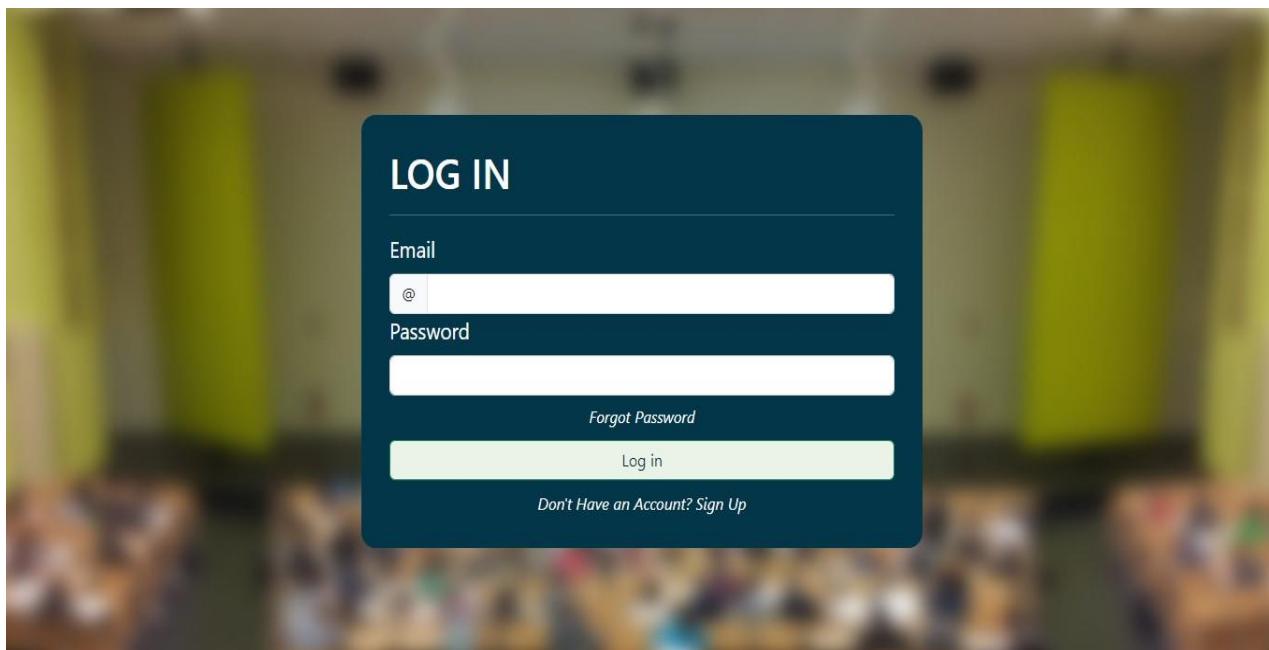


Fig (8): Footer



The image shows a 'SIGN UP' form on a dark teal background. It contains four input fields: 'First Name' and 'Last Name' in the top row, and 'User name' and 'Email' in the bottom row. Below these are two more input fields: 'Password' and 'Comfirm Password'. A 'Remember Password' checkbox is located below the password fields. At the bottom is a large 'SIGN UP' button. Below the button, a link says 'Already have account? [login](#)'.

Fig (9): Sign up page



The image shows a 'LOG IN' form on a dark teal background. It has two input fields: 'Email' and 'Password'. Below the password field is a 'Forgot Password' link. At the bottom is a large 'Log in' button. Below the button, a link says 'Don't Have an Account? [Sign Up](#)'.

Fig (10): Log in page

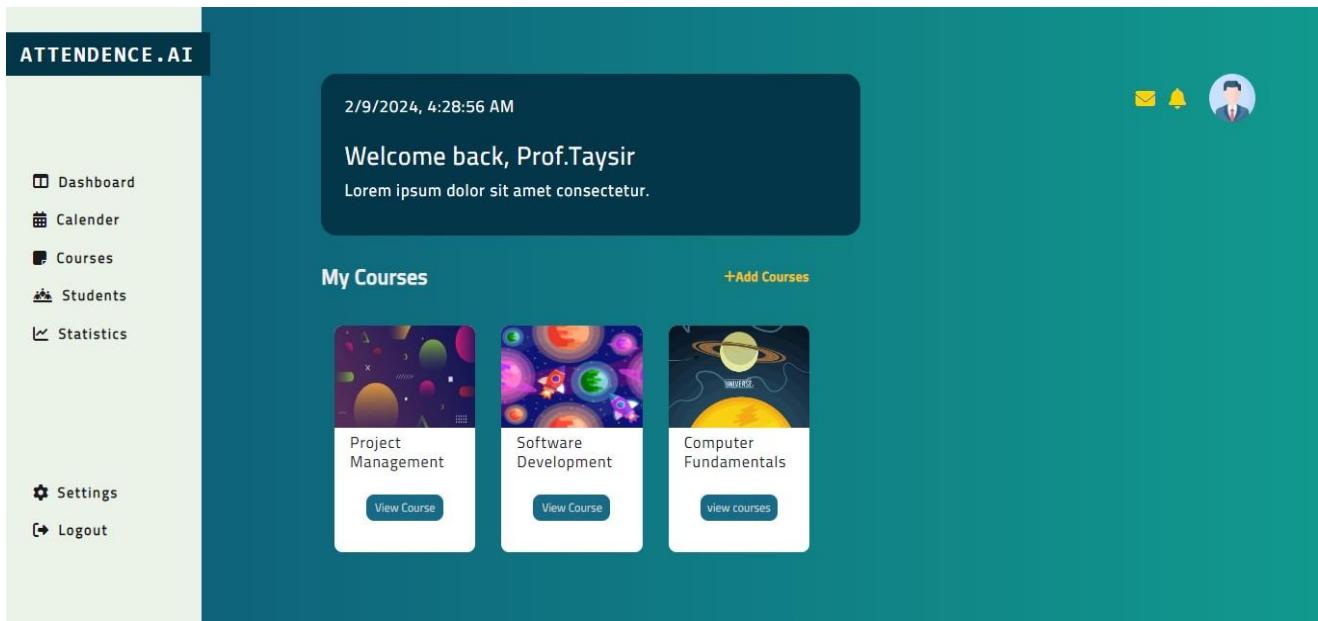
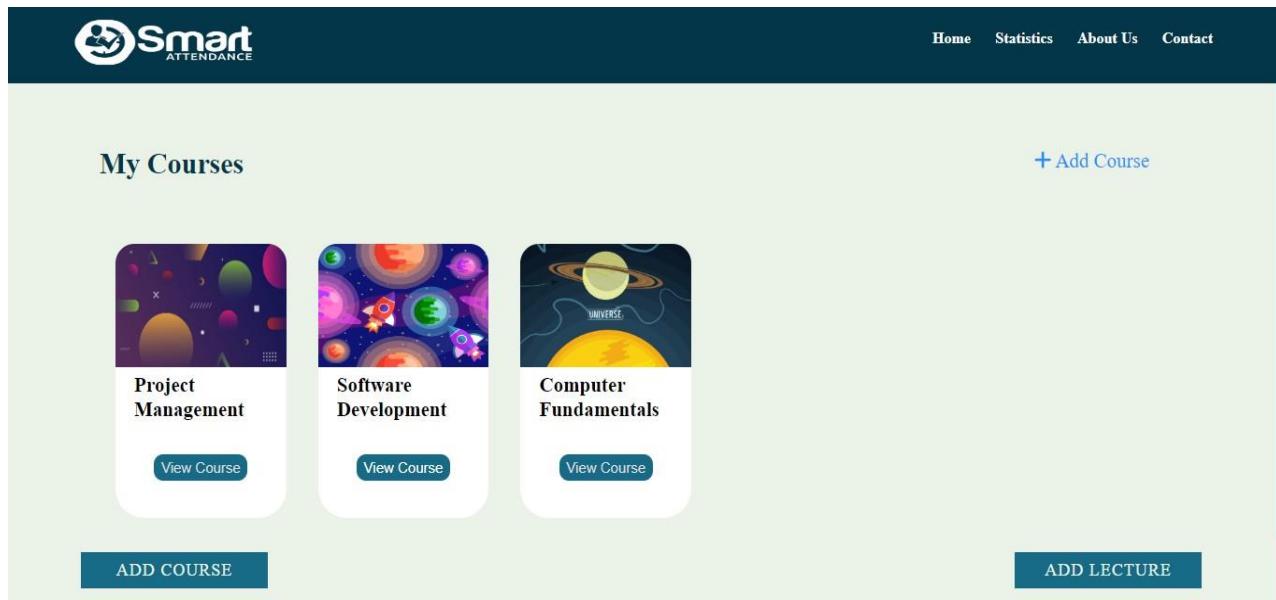


Fig (11): main page

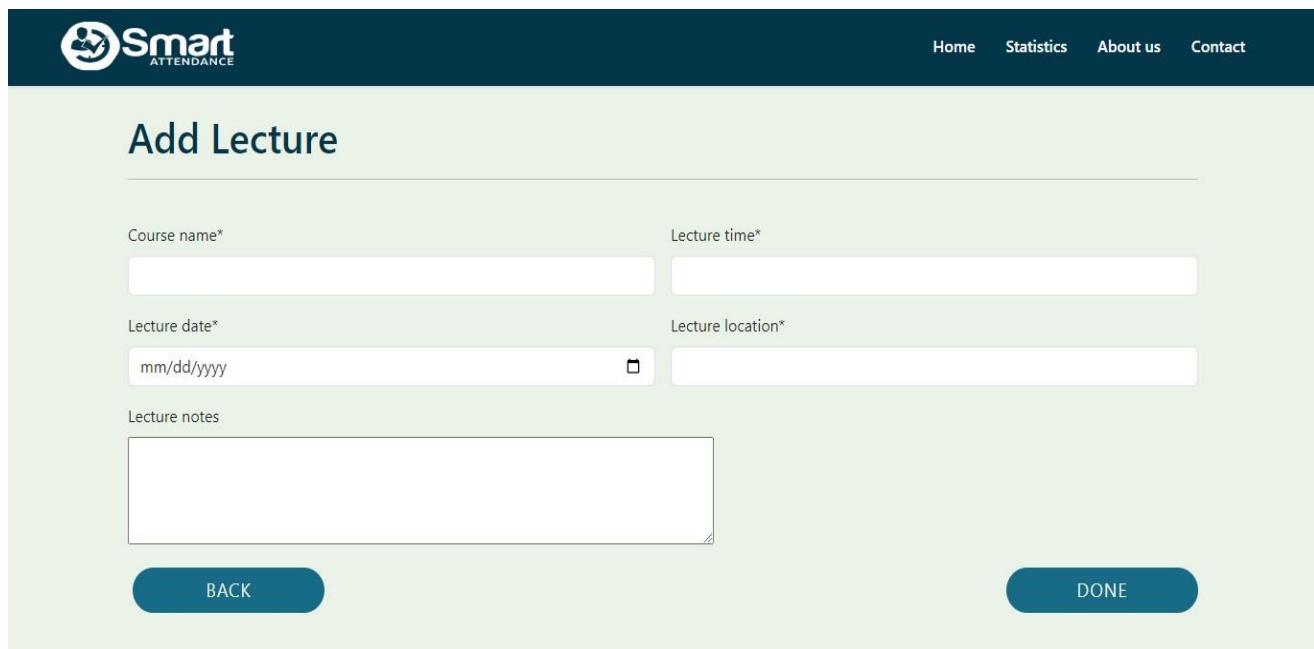
A screenshot of the 'Add Course' form from the Smart ATTENDANCE application. The header features the 'Smart ATTENDANCE' logo on the left and navigation links for 'Home', 'Statistics', 'About Us', and 'Contact' on the right. The main title 'Add Course' is centered above the form fields. The form consists of several input fields: 'Course name*' with a text input field; 'Credit hours*' with a text input field; 'Semester*' with a text input field; 'Level*' with a text input field; 'Minimum Attendance Rate' with a dropdown menu showing 'from 1 to a'; 'Departement*' with a text input field; and 'Attendance instructions' with a text area. At the bottom are two buttons: 'BACK' on the left and 'ADD' on the right.

Fig (12): Add course



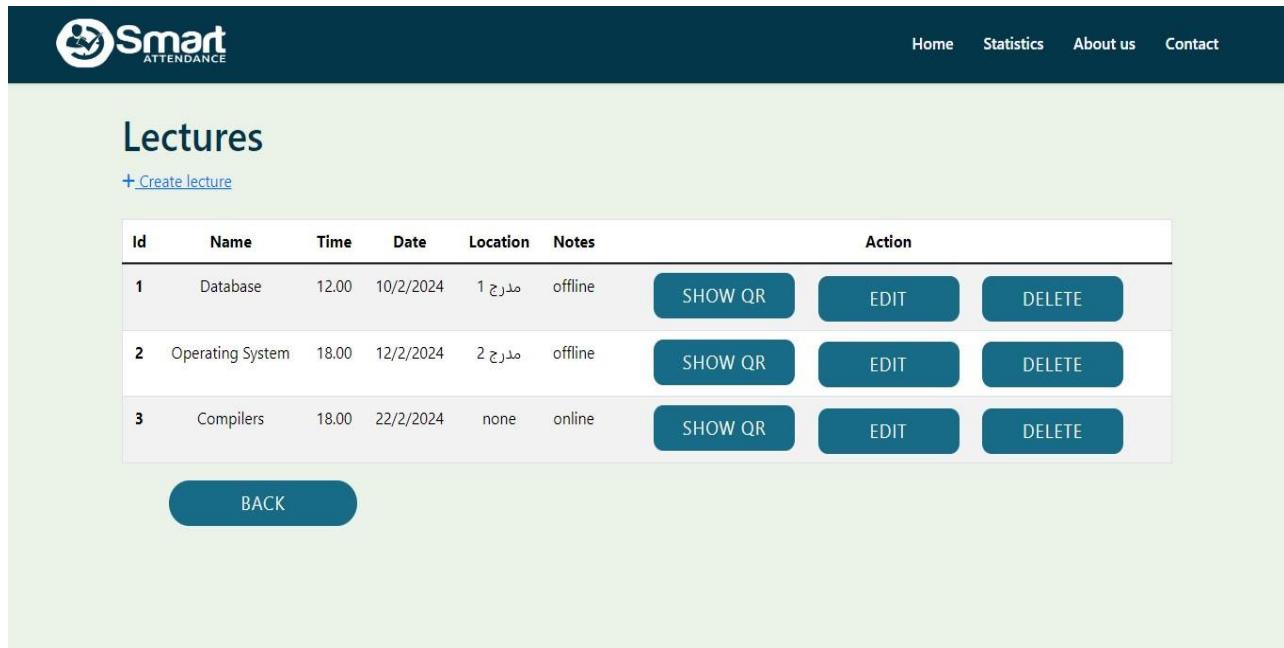
The screenshot shows the 'My Courses' section of the Smart Attendance application. At the top right, there are navigation links: Home, Statistics, About Us, and Contact. Below this, a heading 'My Courses' is followed by a '+ Add Course' button. Three course cards are displayed: 'Project Management' (with a space-themed icon), 'Software Development' (with a rocket and code icon), and 'Computer Fundamentals' (with a solar system icon). Each card has a 'View Course' button at the bottom. At the bottom left is an 'ADD COURSE' button, and at the bottom right is an 'ADD LECTURE' button.

Fig (13): View course



The screenshot shows the 'Add Lecture' form. At the top right, there are navigation links: Home, Statistics, About us, and Contact. The main title is 'Add Lecture'. The form fields include: 'Course name*' with an input field; 'Lecture time*' with an input field; 'Lecture date*' with an input field and a calendar icon; 'Lecture location*' with an input field; and 'Lecture notes' with a large text area. At the bottom are two buttons: 'BACK' on the left and 'DONE' on the right.

Fig (14): Add lecture

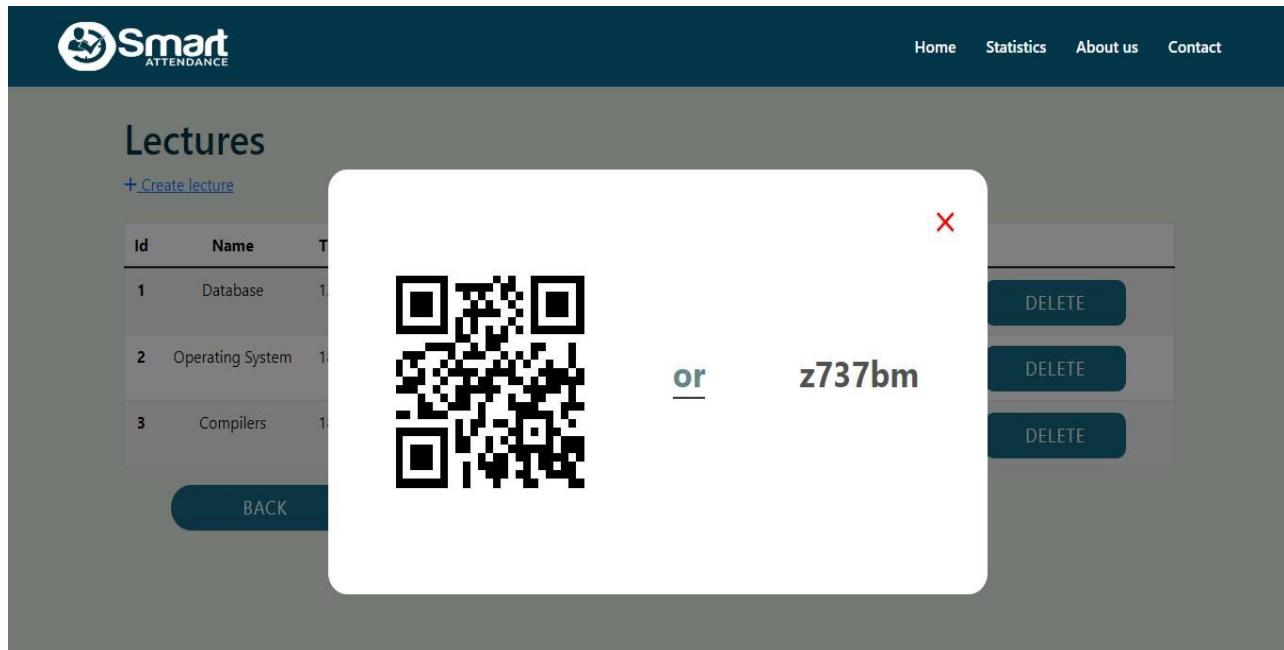


The screenshot shows the 'Lectures' section of the Smart Attendance application. At the top, there is a navigation bar with links for Home, Statistics, About us, and Contact. Below the navigation bar, a heading 'Lectures' is displayed, followed by a link '+ Create lecture'. A table lists three lectures with columns for Id, Name, Time, Date, Location, Notes, and Action. The table rows are as follows:

Id	Name	Time	Date	Location	Notes	Action
1	Database	12.00	10/2/2024	1 مدرج	offline	<button>SHOW QR</button> <button>EDIT</button> <button>DELETE</button>
2	Operating System	18.00	12/2/2024	2 مدرج	offline	<button>SHOW QR</button> <button>EDIT</button> <button>DELETE</button>
3	Compilers	18.00	22/2/2024	none	online	<button>SHOW QR</button> <button>EDIT</button> <button>DELETE</button>

A large blue 'BACK' button is located at the bottom left of the main content area.

Fig (15): View lecture



This screenshot shows the 'Show QR code' interface for a lecture. The background is dark, and the foreground features a white callout box containing a QR code and a string of characters. The QR code is positioned above the word 'or', and the string 'z737bm' is positioned below it. In the top right corner of the callout box is a red 'X' icon. To the right of the callout box, a vertical stack of three teal-colored buttons, each labeled 'DELETE', is visible. The background of the page shows a list of lectures, with the first two rows partially visible. A 'BACK' button is located at the bottom left of the main content area.

Fig (16): Show QR code



Courses/Database

#	Name	Attendance	Delay
1	Mark	✗	4-minute
2	Jacob	✗	
3	Larry	✓	33-minute
4	Mark	✗	4-minute
5	Jacob	✓	
6	Larry	✓	33-minute
7	Mark	✗	4-minute
8	Jacob	✗	
9	Larry	✓	33-minute
10	Mark	✗	4-minute
11	Jacob	✗	
12	Larry	✓	33-minute
13	Mark	✗	4-minute
14	Jacob	✗	
15	Larry	✓	33-minute

BACK

Fig (17): Show sheet



About Us



Welcome To Our Website!

Take attendance in no time! The first ever automatic student attendance tracker which is secured and fully monitored web-based system.

Learn More

Fig (18): About page



Contact

Name Email

Phone Subject

Message

SEND **BACK**

Fig (19): Contact page

3.1.2. Mobile app

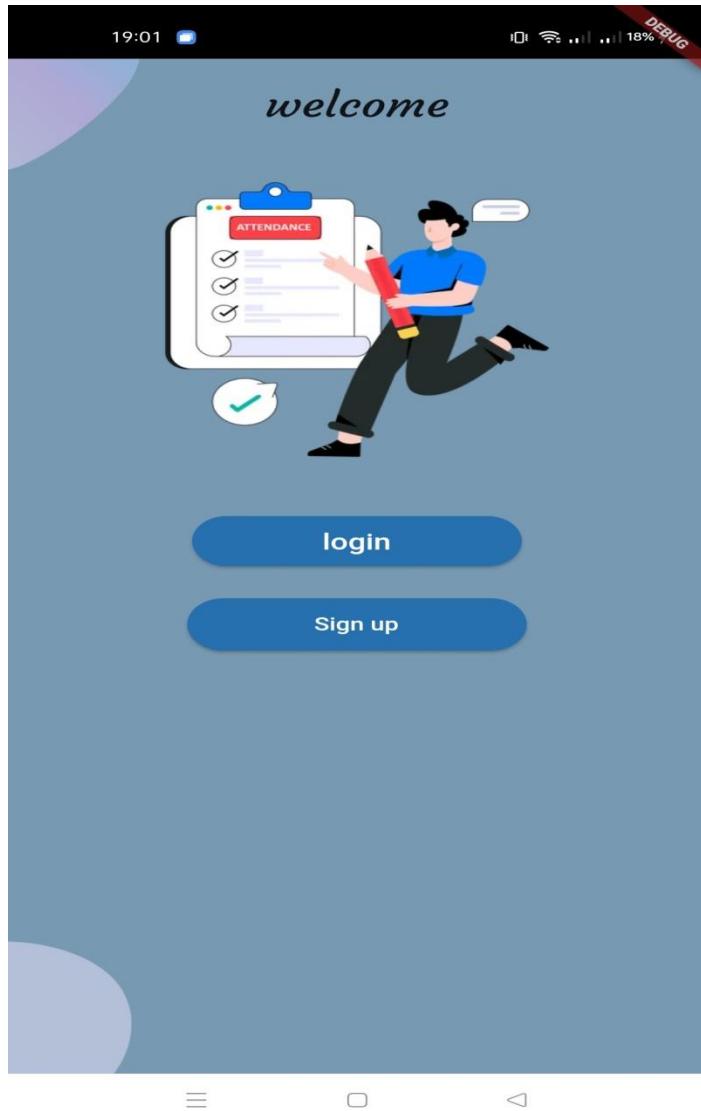


Fig (20): Welcome

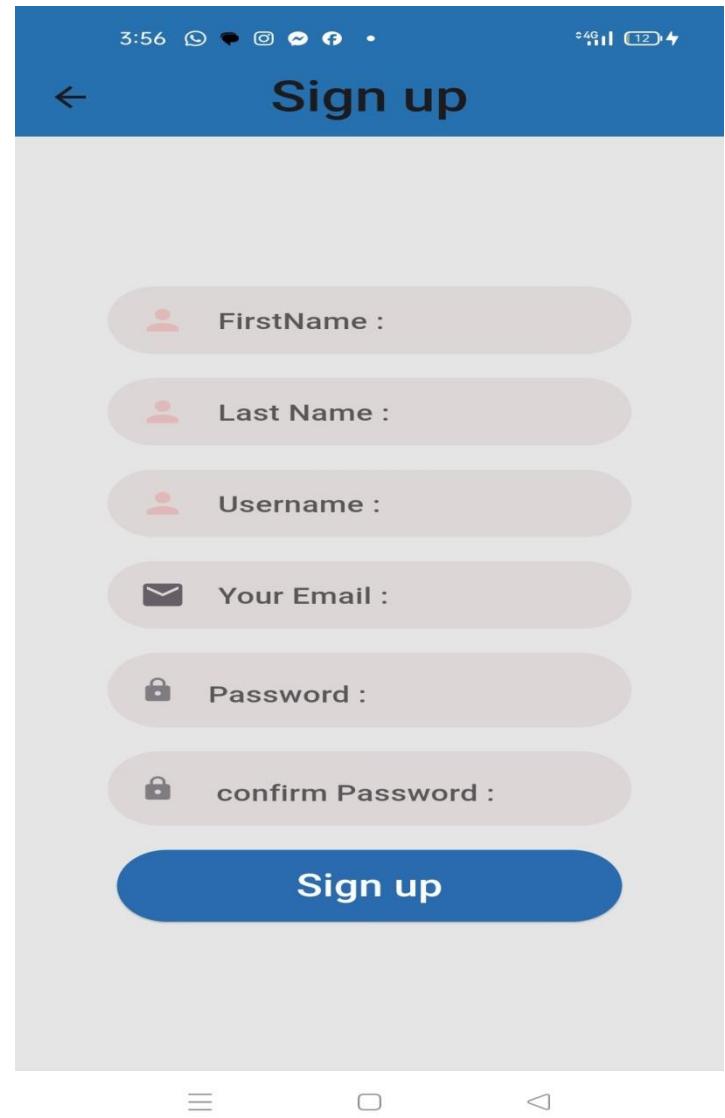


Fig (21): Sign up

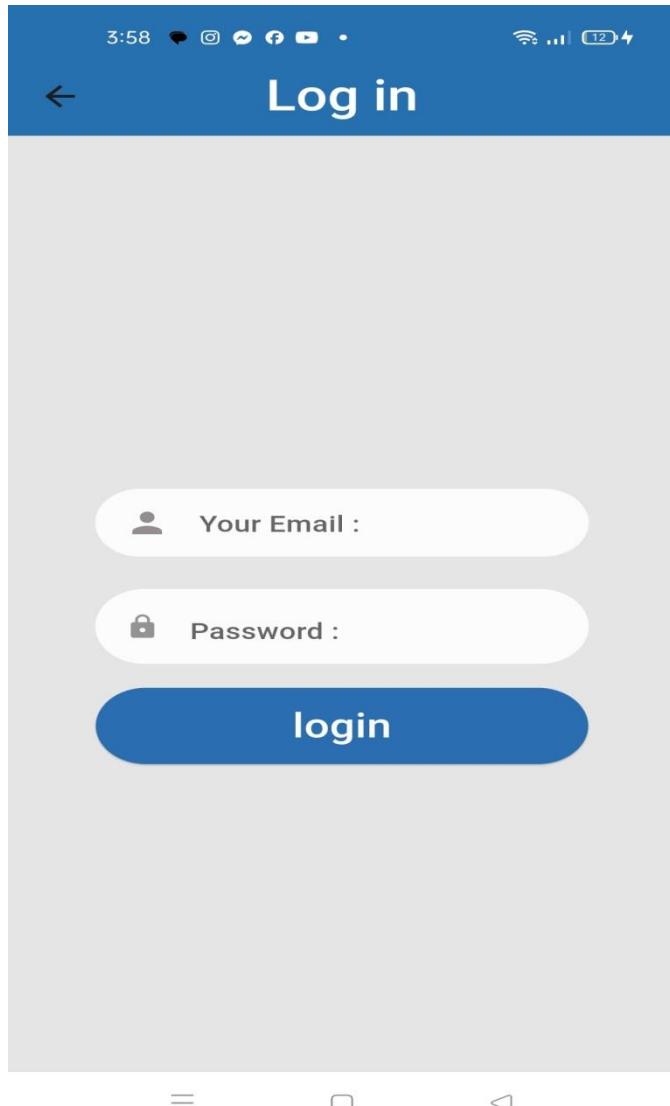


Fig (22): Log in

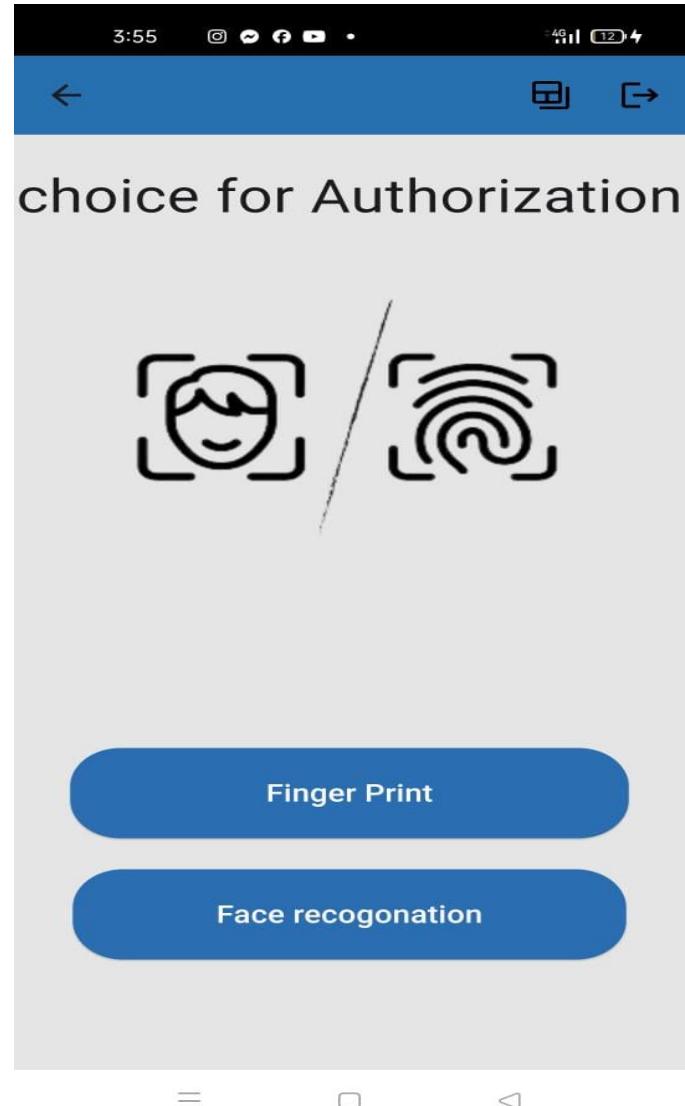


Fig (23): Main page

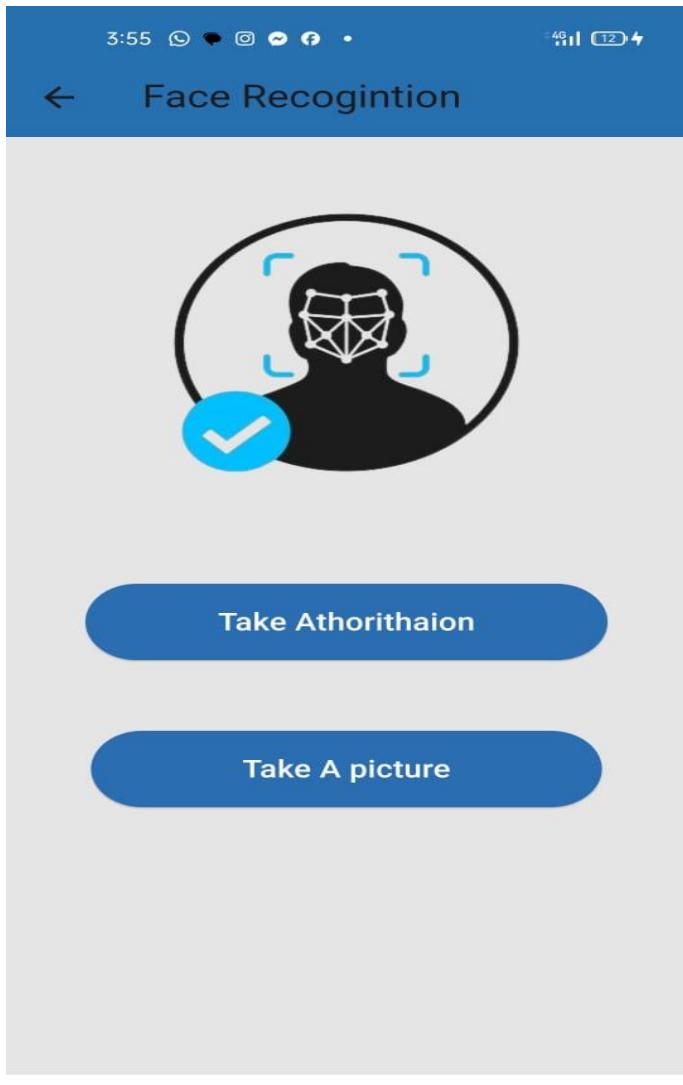


Fig (24): Face recognition



Fig (25): Fingerprint

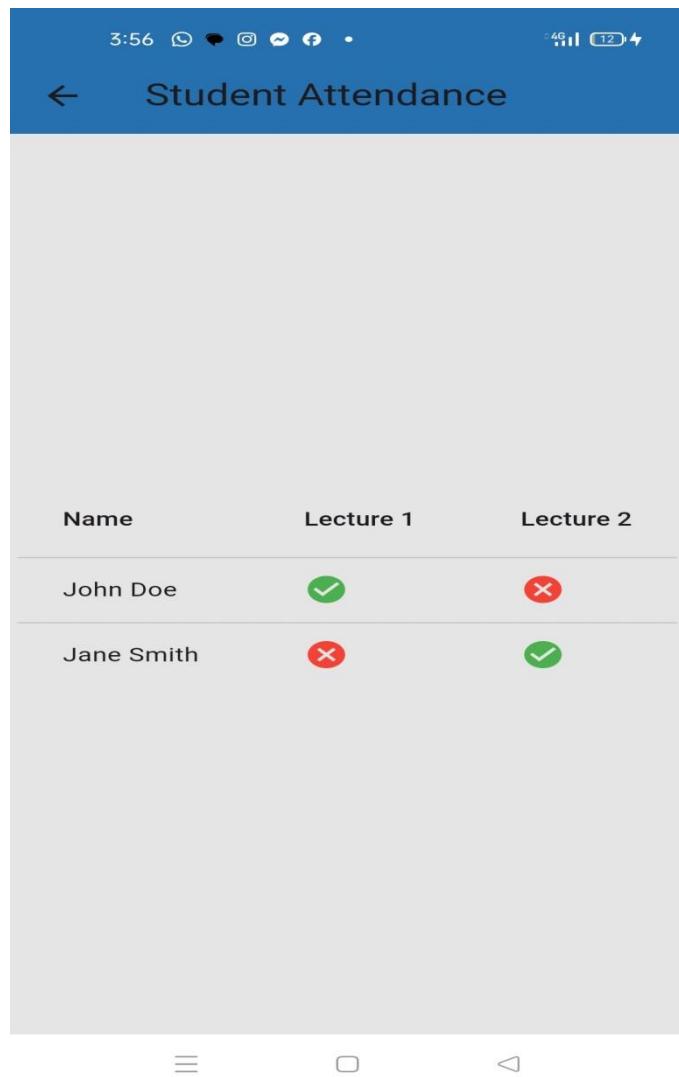


Fig (26): Dashboard

3.2 Hardware Interfaces:

The requirements of the desktop computer where the system going to be installed.

The requirements of a smartphone that has a camera or fingerprint and installing the application for the student to be able to attend.

3.3 Software Interfaces:

The computer this software going to be install need to have windows 10 operating system. On that Windows platform

.Net 7.0 will be installed and that will be the platform the particular software will be run. There will be Microsoft SQL Server Management Studio 2018 that should be installed. The operating system for Mobile devices should be Android.

3.4 Communications Interfaces:

When a student scans the QR code after verifying the identity using the fingerprint or facial recognition feature, it will be registered with the professor in the attendance table.

4. System Features

4.1. ER-Diagram:

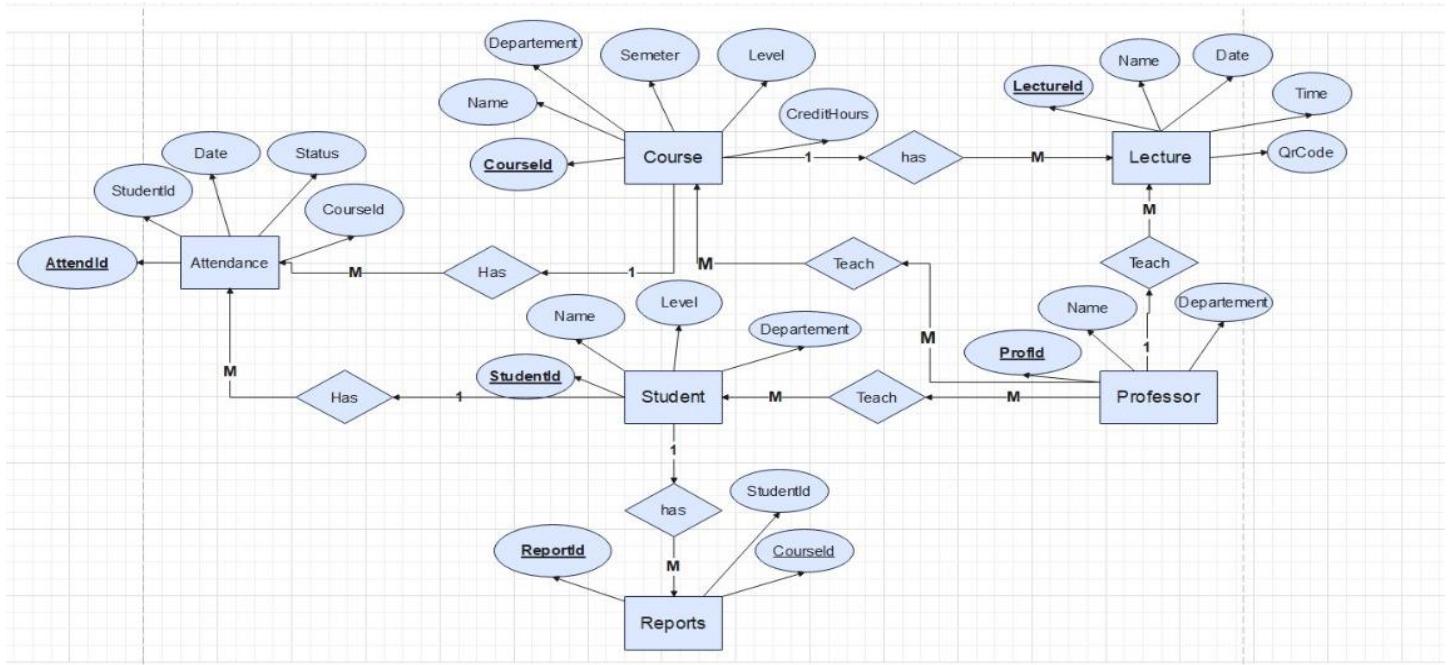


Fig (27): ERD

4.2. Use case diagram:

Use case diagram

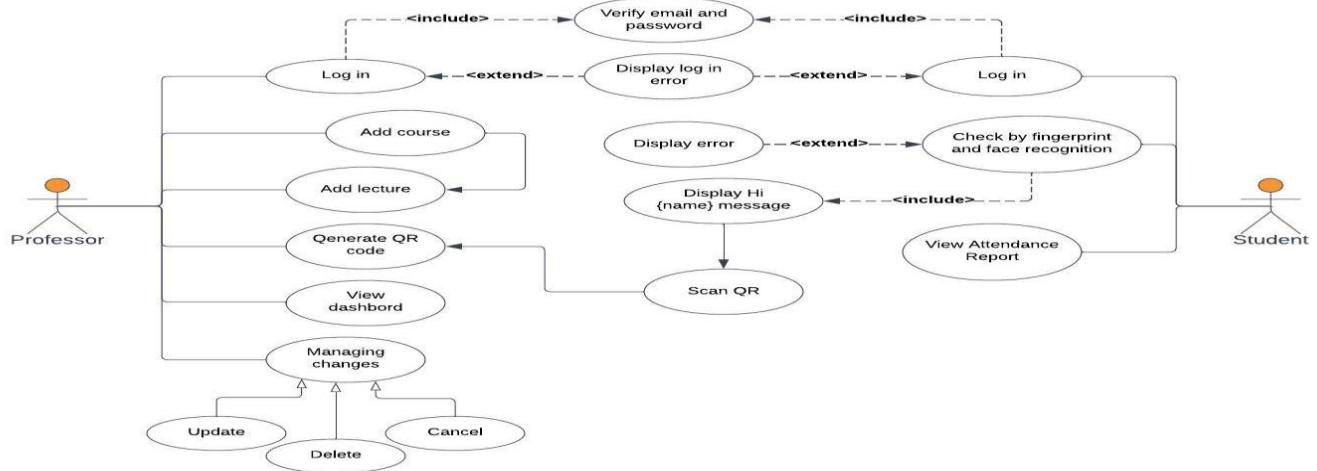


Fig (28): Use case diagram

4.3. Use Case Scenarios:

4.3.1. Sign up Use case scenarios:

Use case Name	Sign up	
Goal	Create account	
Primary actors	User (Professor or Teaching Assistan or Student)	
Secondary actors		
preconditions	Connect to the internet User has E-mail	
Post conditions		
Triggers		
Main Flow	Step	Actions
	1	Turn on data or WIFI
	2	Open the system (app, web)
	3	Click on the sign up
	4	Enter information Requirements
Extensions	4.1	Check about email and password valid or invalid

4.3.2. Log in Use case scenarios:

Use case Name	Log in	
Goal	Log in to app	
Primary actors	User (Professor or Teaching Assistant or Student)	
Secondary actors		
preconditions	Connect to the internet User has E-mail	
Post conditions		
Triggers		
Main Flow	Step	Actions
	1	Turn on data or WIFI
	2	Open the system (app, web)
	3	Click on the login
	4	Enter email and password
Extensions	4.1	Invalid email or password Enter again

4.3.3. Verification of fingerprint

Use case Name	Verification of fingerprint	
Goal	Verify the student's fingerprint	
Primary actors	Student	
Secondary actors	None	
preconditions	Log in or sign up	
Post conditions		
Triggers	Authorization click Mobile fingerprint	
Main Flow	Step	Actions
	1	Log in or sign up
	2	Take fingerprint authorization
Extensions		

4.3.4. Verification of Face recognition

Use case Name	Verification of Face recognition	
Goal	Verify the student's face recognition	
Primary actors	Student	
Secondary actors	None	
preconditions	Log in or sign up	
Post conditions	Click on button	
Triggers	Authorization face recognition of student	
Main Flow	Step	actions
	1	Log in or sign up
	2	Take Face recognition authorization
	3	Update database
Extensions		

4.3.5. Scan QR code

Use case Name	Scan QR code	
Goal	Scan QR code to register attendance	
Primary actors	Student	
Secondary actors	None	
preconditions	Log in to the system and Verification of fingerprint or Face recognition	
Post conditions		
Triggers	Click to the button to scan	
Main Flow	Step	actions
	1	Verification of fingerprint or Face recognition
	2	Click to button
	3	Scan the QR code
Extensions	1.1	2.1 If confirmation is not confirmed by fingerprint or face

4.3.4. Show dashboard (Student)

Use case Name	Show dashboard	
Goal	To see how many times student is absent and present	
Primary actors	Student	
Secondary actors	None	
preconditions	Student registered attendance	
Post conditions		
Triggers	Scan QR code	
Main Flow	Step	actions
	1	After scan QR code
	2	System save in database
	3	Only show dashboard Attendance
Extensions		

4.3.5. Add course

Use case Name	Add course	
Goal	Add a new course to the system	
Primary actors	Professor	
Secondary actors	None	
preconditions	Log in the system	
Post conditions	None	
Triggers	Professor clicks on "Add Course" button	
Main Flow	Step	actions
	1	Click on "Add Course" button
	2	System prompts the owner to fill out Course details
	3	System validates new course information
	4	System creates a new course
	5	Update database
	6	Display Successful Message
Extensions	3.1	Course Details are incorrect. Display The message "Unsuccessful" and display Course management option.

4.3.8. Add lecture

Use case Name	Add lecture	
Goal	Add lecture to a course	
Primary actors	Professor	
Secondary actors	Teaching Assistant	
preconditions	Log in the system	
Post conditions	None	
Triggers	Professor clicks on "Add Lecture" button	
Main Flow	Step	actions
	1	Click on "Add Lecture" button
	2	System prompts the owner to fill out Lecture details
	3	System validates new lecture information
	4	System creates a new Lecture
	5	Update database
	6	Display Successful Message
Extension	3.1	Lecture Details are incorrect. Display The message "Unsuccessful" and display Lecture management option.

4.3.9. Generate OR code

Use case Name	Generate OR code	
Goal	How generate OR code to scan by student	
Primary actors	Professor	
Secondary actors		
preconditions	Add lecture	
Post conditions		
Triggers	Professor clicks on "Generate" button.	
Main Flow	Step	actions
	1	Professor select the course or create course
	2	Click on " Generate " button
	3	View QR code
	4	Professor share QR code with student
Extensions		

4.3.10. Show dashboard

Use case Name	Show dashboard	
Goal	To see how many times student is absent and present	
Primary actors	Professor	
Secondary actors	Teaching Assistant	
Preconditions	Student registered attendance	
Post conditions		
Triggers	Student scan QR code	
Main Flow	Step	actions
	1	After scan QR code
	2	System save in database
	3	Professor can show, update and delete dashboard Attendance
Extensions		

4.3.11. Update Course

Use case Name	Update Course	
Goal	Update course properties	
Primary actors	Professor	
Secondary actors		
preconditions	Log in the system	
Post conditions	None	
Triggers	Professor clicks on "Update Course" button.	
Main Flow	Step	actions
	1	Click on "Update Course" button
	2	System prompts professor to select course
	3	Professor select the course
	4	System display course properties
	5	Professor update course properties
	6	Update from database
	7	Display Successful Message
Extensions		

4.3.12. Update Lecture

Use case Name	Update Lecture	
Goal	Update lecture properties	
Primary actors	Professor	
Secondary actors		
preconditions	Log in the system	
Post conditions	None	
Triggers	Professor clicks on "Update Lecture" button.	
Main Flow	Step	actions
	1	Click on "Update Lecture" button
	2	System prompts professor to select lecture
	3	Professor select the lecture
	4	System display lecture properties
	5	Professor update lecture properties
	6	Update from database
	7	Display Successful Message
Extensions		

4.3.13. Delete Course

Use case Name	Delete Course	
Goal	Delete course properties	
Primary actors	Professor	
Secondary actors		
preconditions	Log in the system	
Post conditions	None	
Triggers	Professor clicks on "Delete Course" button.	
Main Flow	Step	actions
	1	Professor select the course
	2	Click on "Delete Course" button
	3	System display options (yes or no)
	4	Professor delete course (yes) or cancel (no)
	5	Delete from database
	6	Display Successful Message
Extensions		

4.3.14. Delete Lecture

Use case Name	Delete Lecture	
Goal	Delete lecture properties	
Primary actors	Professor	
Secondary actors		
preconditions	Log in the system	
Post conditions	None	
Triggers	Professor clicks on "Delete Lecture" button.	
Main Flow	Step	actions
	1	Professor select the lecture
	2	Click on "Delete Lecture" button
	3	System display options (yes or no)
	4	Professor delete lecture (yes) or cancel (no)
	5	Delete from database
	6	Display Successful Message
Extensions		

4.4. Activity Diagram

4.4.4. Sign up user (web - app)

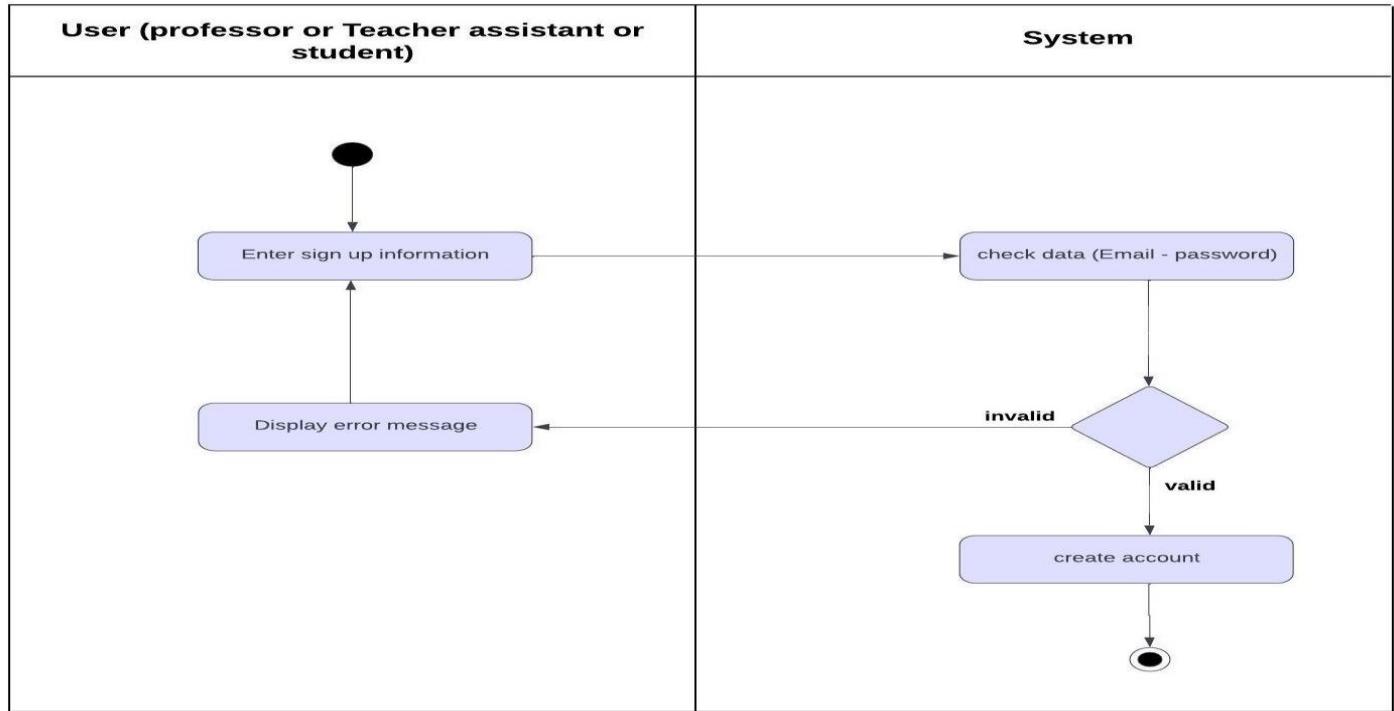


Fig (29): Sign up

4.4.2. Log in user (web - app)

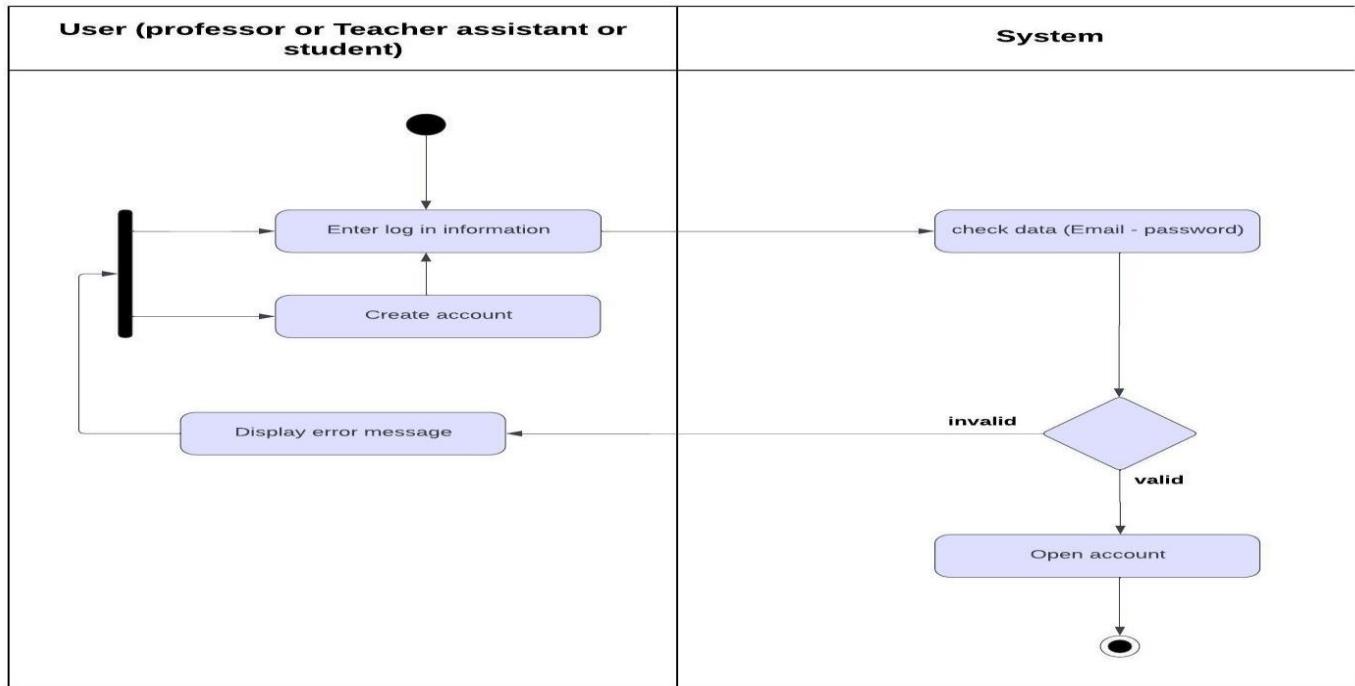


Fig (30): Log in

4.4.3. Professor add course

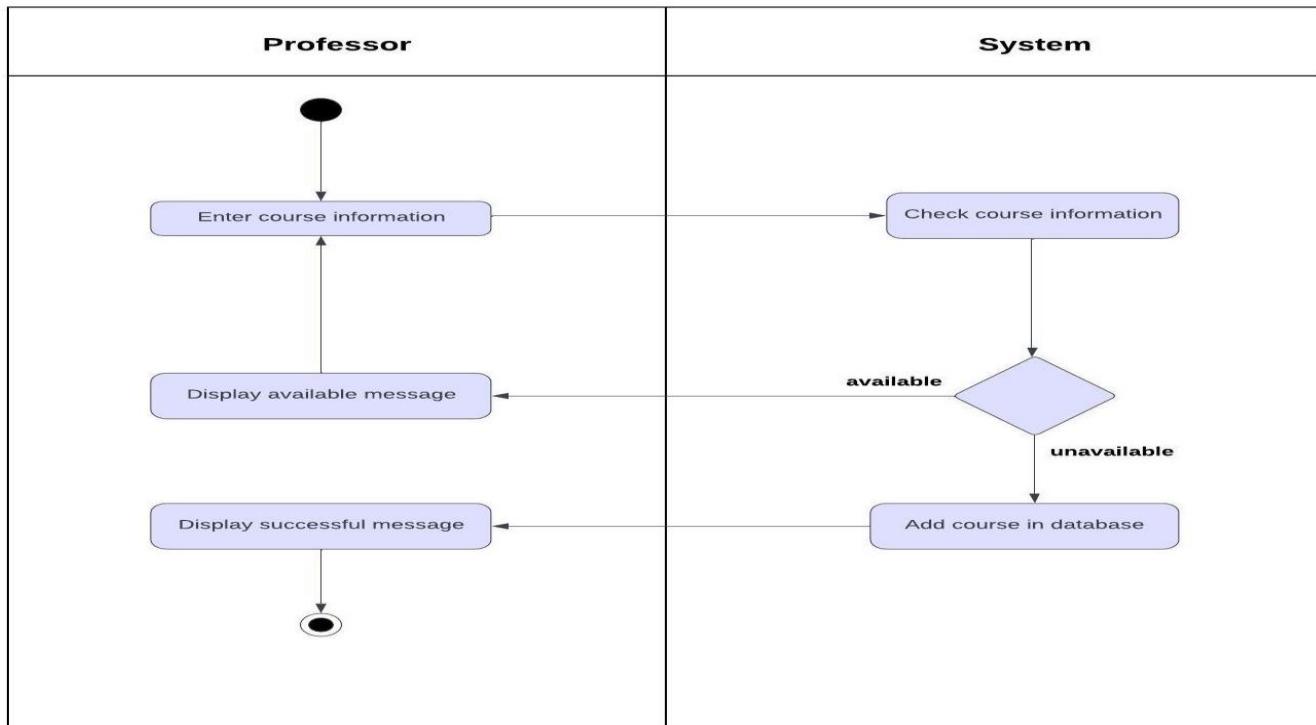


Fig (31): Add course

4.4.4 Professor add lecture

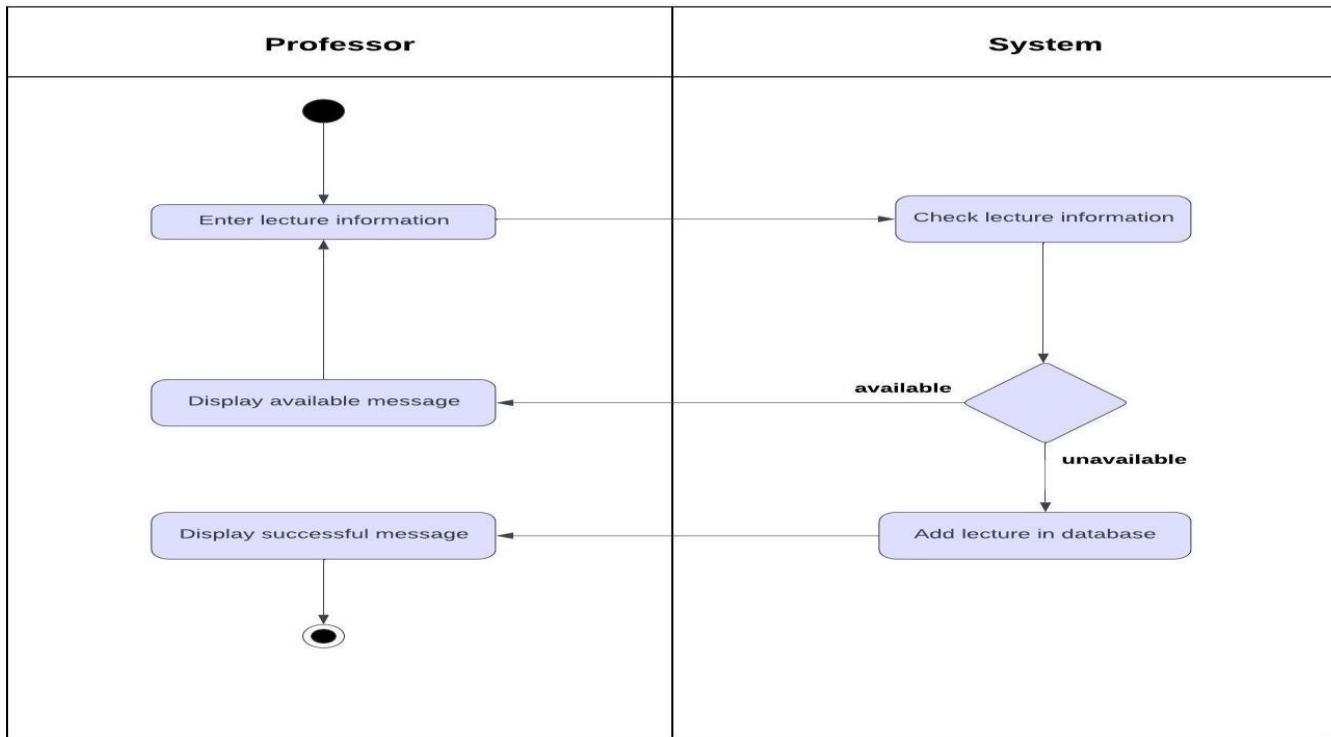


Fig (32): Add lecture

4.4.5. Professor update course

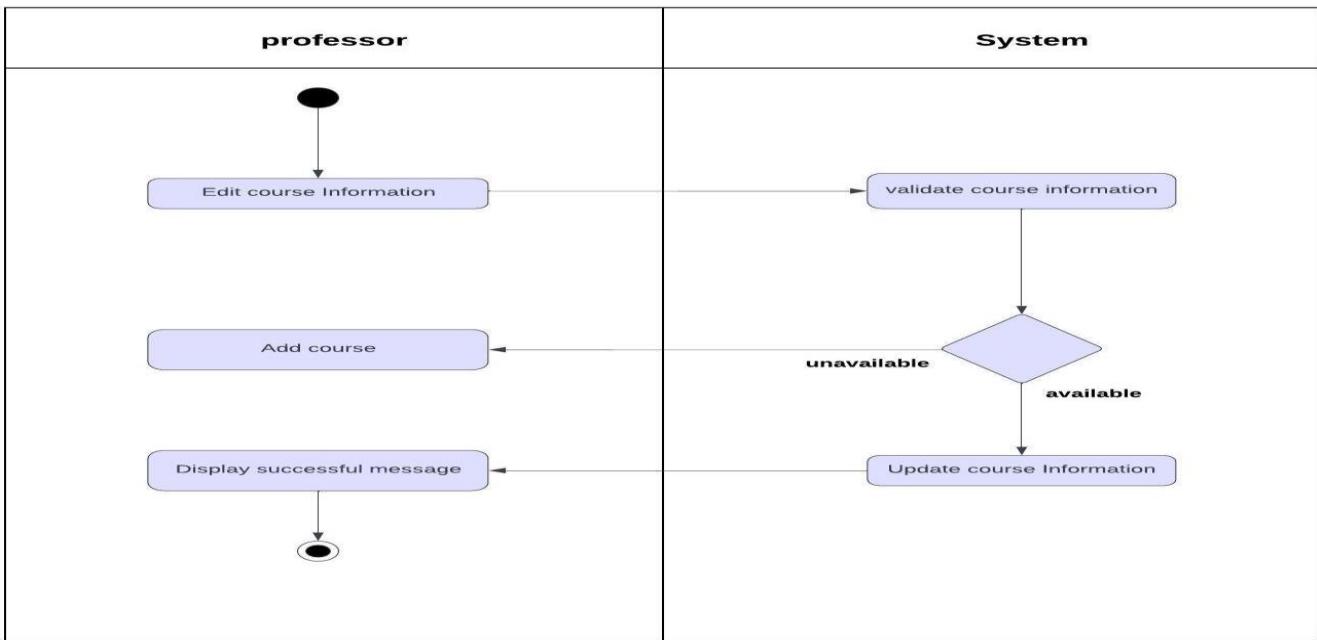


Fig (33): Update course

4.4.6. Professor update lecture

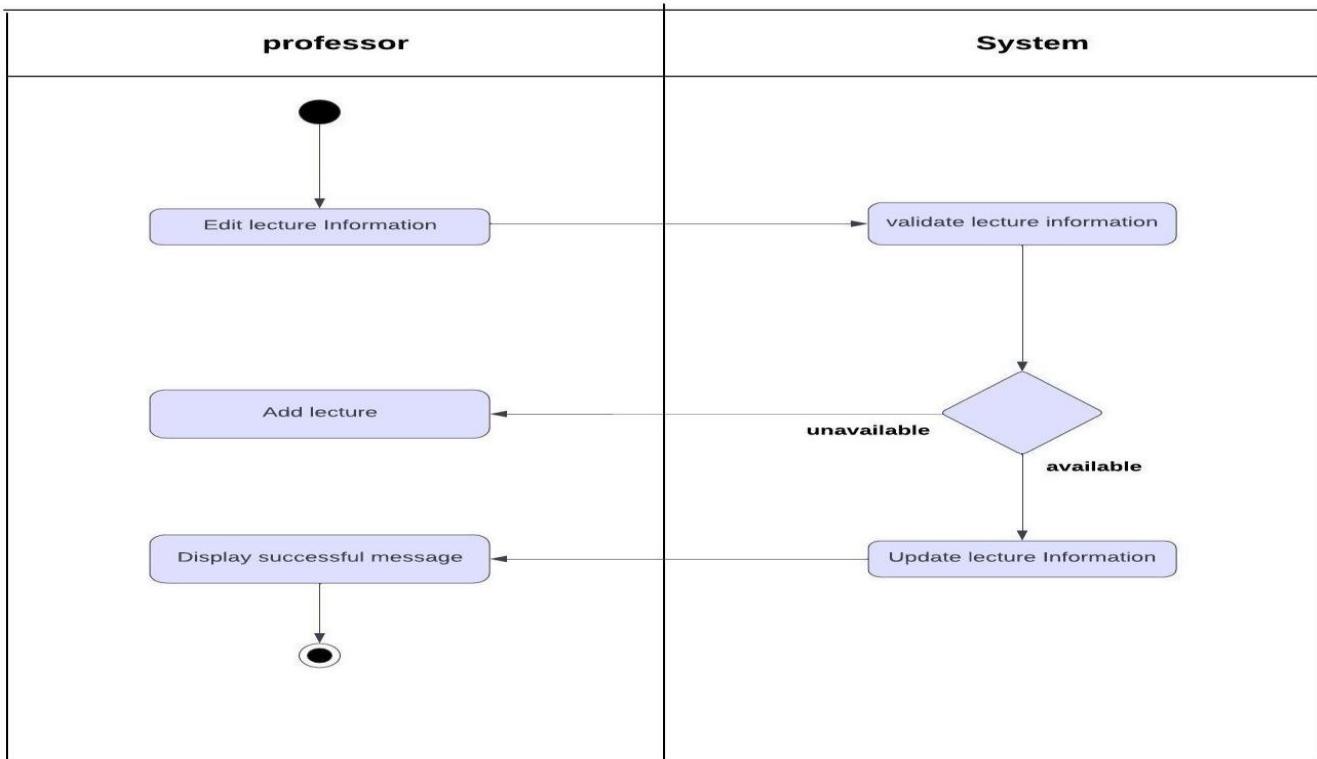


Fig (34): Update lecture

4.4.7. Professor delete course

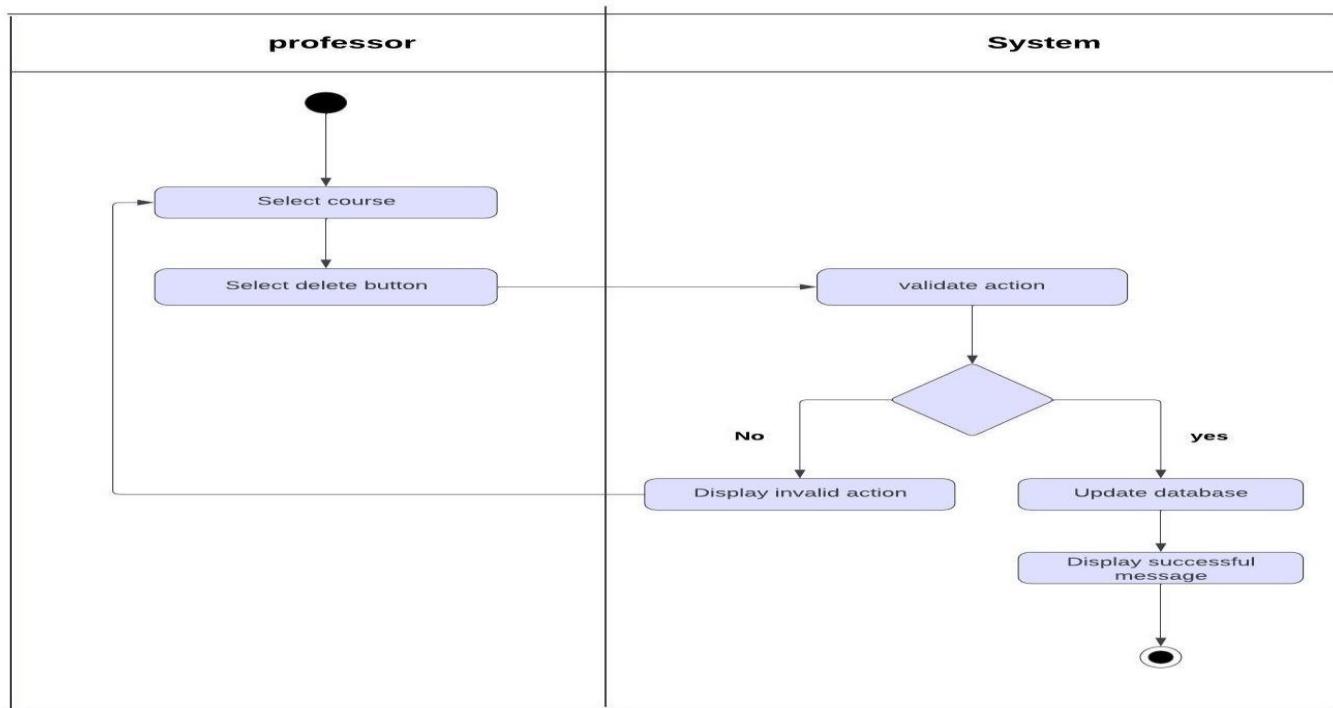


Fig (35): Delete course

4.4.8. Professor delete lecture

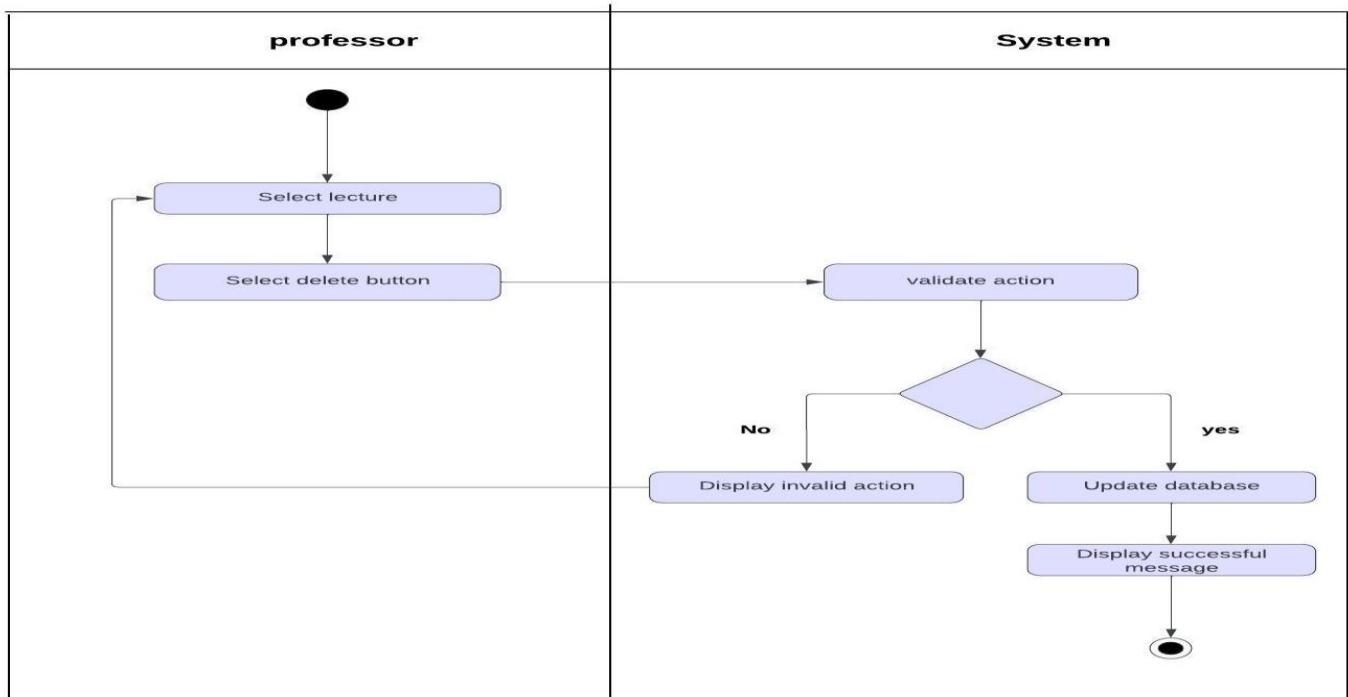


Fig (36): Delete lecture

4.4.9. Professor generate QR code

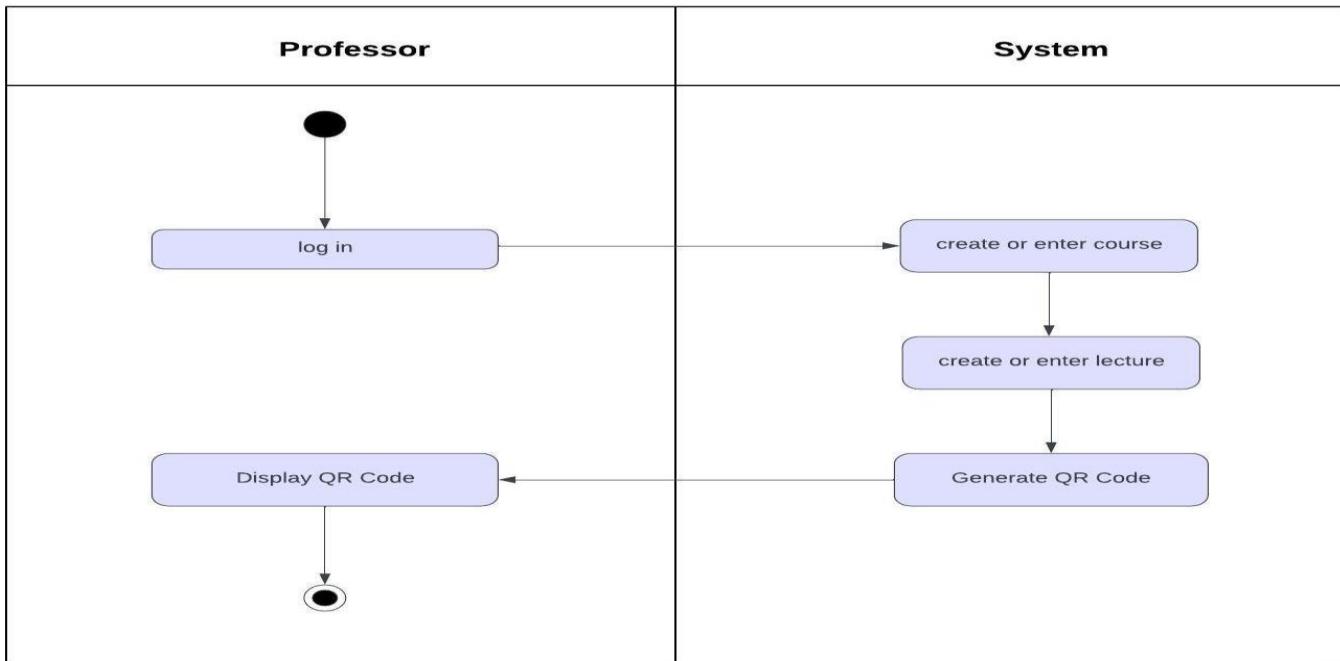


Fig (37): Generate QR code

4.4.10. Face recognition (student)

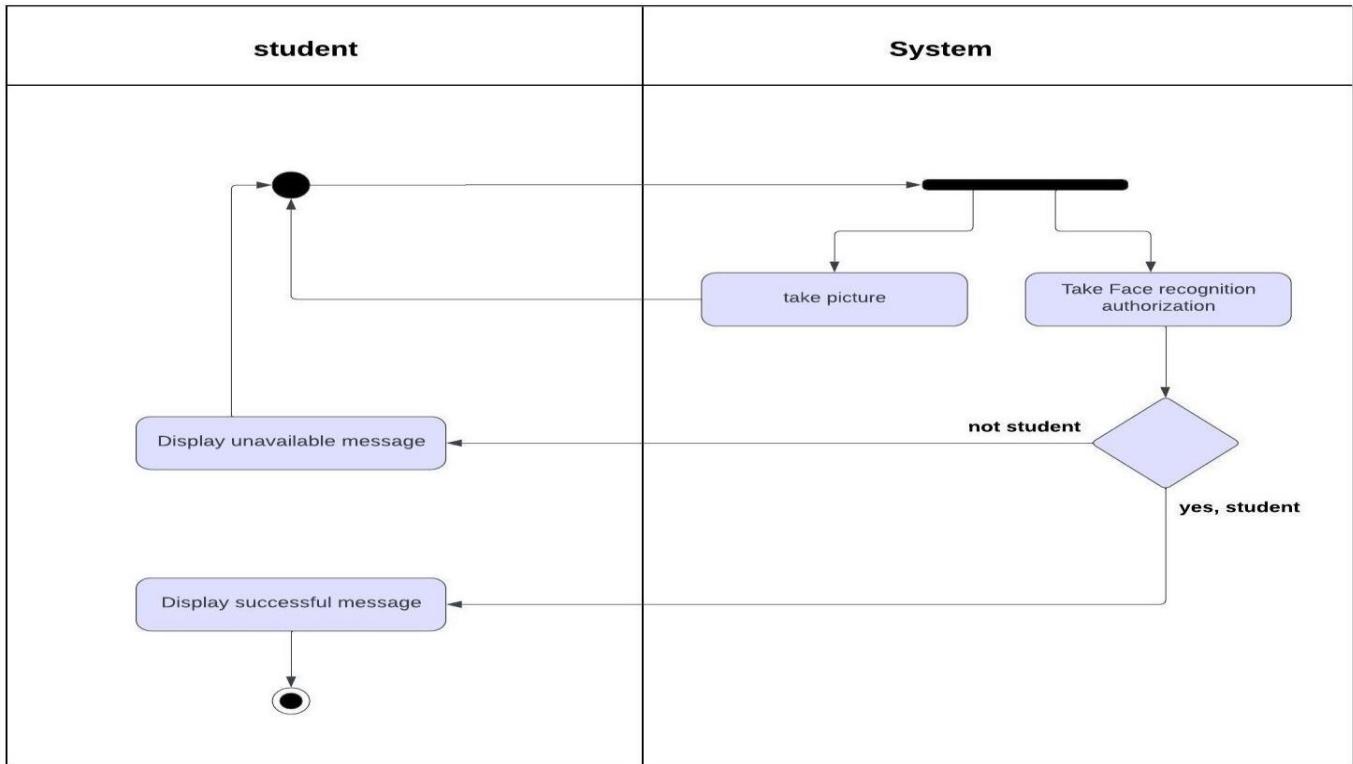


Fig (38): Face recognition

4.4.11. finger print (student)

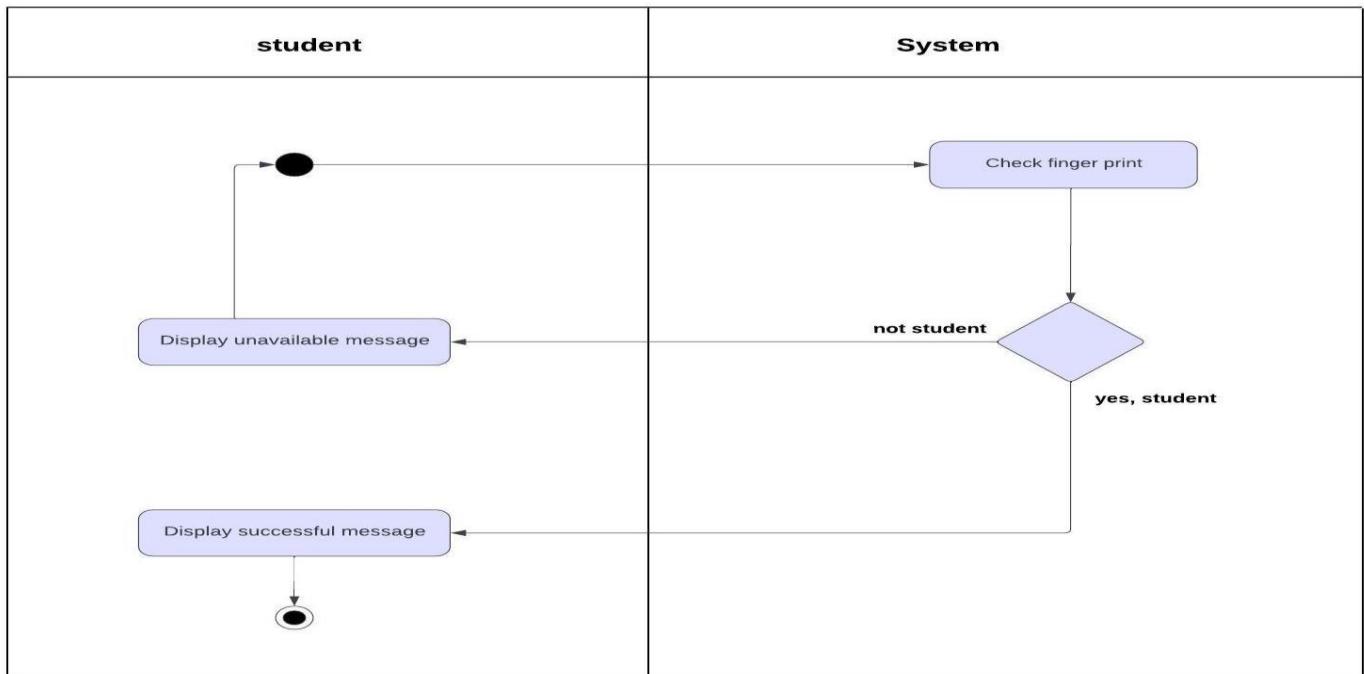


Fig (39): Finger print

4.4.12 Student scans QR code

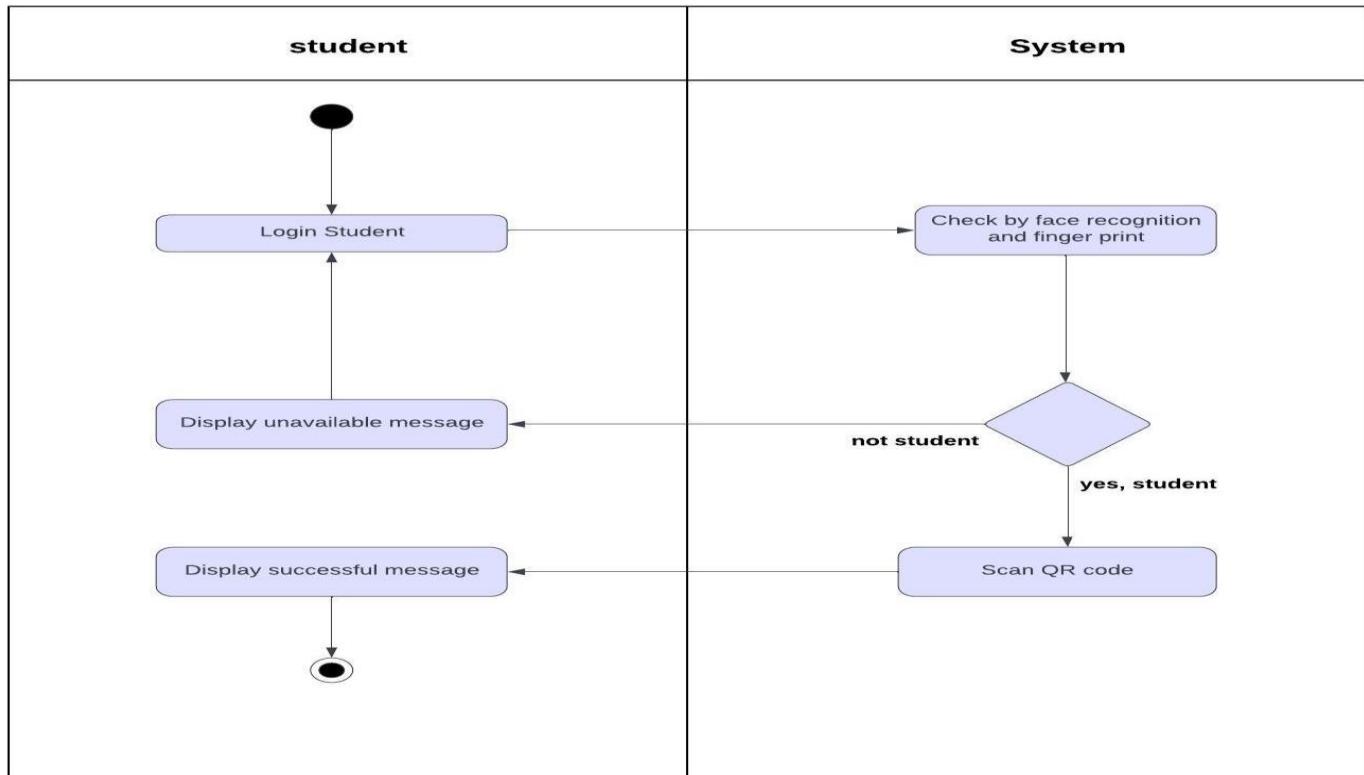


Fig (40): Scan QR code

5. Other Nonfunctional Requirements

5.1. Performance Requirements:

The system should provide a response time of less than 3 seconds for any user action, such as generating QR codes or sending attendance confirmations. The system should be designed to handle a scalable number of users.

5.2. Safety Requirements:

The system must comply with relevant data protection laws and regulations, ensuring the privacy and confidentiality of student and doctor information. Measures should be in place to ensure the integrity of data during transmission and storage, preventing unauthorized tampering or manipulation.

Implement measures to ensure the security of QR codes, preventing unauthorized generation or duplication. In the case of planned system maintenance or unexpected downtime, and any unsaved data should be preserved.

5.3. Security Requirements:

Only individuals with valid academic credentials and affiliations should be allowed to register as students or doctors. Only professor and teaching assistant can register inter to the website. Professor only can perform his/her particular actions. And no one without registered can inter to the app (only student). Implement secure algorithms for QR code generation to prevent malicious attempts to manipulate or counterfeit attendance records.

5.4. Software Quality Attributes:

Availability: The attendance registration system should be available and responsive 99.9% of the time during scheduled class hours.

Flexibility: Scanning Options: The system seems flexible as it supports both fingerprint and facial recognition for identity verification, providing options for students.

Maintainability: Integration with Professor's System: The system's ability to register attendance in the professor's table indicates integration capabilities, contributing to maintainability by ensuring seamless interaction with existing systems.

Integrity: The system should ensure that the data recorded in the attendance table accurately represents the students who have scanned the QR code and successfully verified their identity.

Security Measures: the system should implement robust security measures to prevent unauthorized access or tampering with attendance records.

Portability: The system will run on windows 10 or above and Android Operating System.

Correctness: Both fingerprint and facial recognition features should have a high accuracy rate to prevent false detection.

Efficiency: It takes a student 5 seconds to complete the QR scan.

Testability: testing face recognition and fingerprint.

Reliability: The system must be available and accessible. Students should be able to use the attendance recording system and professors to register without experiencing frequent pauses or interruptions.

5.5. Business Rules:

QR codes generated for attendance confirmation should have a limited validity period (e.g., 3 hours) to prevent misuse or attempts to reuse codes for multiple sessions. Professor must ensure that students attend before the end of the lecture. Implement a data retention policy, specifying the duration for which attendance records and other data will be stored.

6. Other Requirements

6.1. Legal and Compliance Requirements:

6.1.1. Data Protection and Privacy:

- The Smart Attendance System (SAS) must comply with relevant data protection laws, including but not limited to GDPR, ensuring the privacy and security of user data.
- The system should provide mechanisms for obtaining user consent for data processing and clearly communicate the purposes of data collection.

6.2. Performance Requirements:

6.2.1. Response Time:

- The system should respond to user interactions, such as attendance recording and report generation, within an acceptable time frame. Response times should not exceed 2 seconds for common operations.

6.2.2. Scalability:

- The SAS should be designed to handle a scalable number of users, classes, and attendance records. Performance testing should be conducted to ensure the system's responsiveness as the user base grows.

6.3. Security Requirements:

6.3.1. User Authentication:

- The system must implement secure user authentication mechanisms, including password protection and, if applicable, two-factor authentication, to prevent unauthorized access.

6.3.2. Data Encryption:

- All sensitive data, including user credentials and attendance records, must be encrypted during transmission and storage to prevent unauthorized access.

6.4. User Training and Support:

6.4.1. Training Material:

- The development team should provide comprehensive training materials, including user manuals and video tutorials, to facilitate user onboarding and system understanding.

6.4.2. Help Desk Support:

- A help desk or customer support system should be established to assist users with inquiries, issues, or technical difficulties related to the Smart Attendance System.

6.5. System Maintenance and Updates:

6.5.1. Maintenance Schedule:

- A regular maintenance schedule should be established to address system updates, bug fixes, and any necessary improvements. Maintenance activities should be communicated to users in advance.

Appendix A: Glossary

1. Attendance Record:

- A digital or physical record documenting the presence or absence of a user (student or employee) during a specific class or work session.

2. Biometric Authentication:

- Definition: A security measure that uses unique biological characteristics, such as fingerprints or facial recognition, to verify the identity of individuals accessing the system.

3. QR Code:

- A two-dimensional barcode that stores information, which can be scanned and interpreted by devices equipped with QR code readers. In the context of the Smart Attendance System, QR codes are generated for attendance tracking purposes.

4. Dashboard:

- A graphical user interface that provides an overview of important information and functionalities, allowing users (professors, teaching assistants, or students) to access relevant data and features at a glance.

5. Facial Recognition:

- Definition: A technology that identifies and verifies individuals by analyzing unique facial features captured through images or video. In the context of the Smart Attendance System, facial recognition is used for user authentication.

6. Fingerprint Scanner:

- A biometric device that captures and analyzes fingerprint patterns for user identification. In the Smart Attendance System, fingerprint scanners are used as a means of biometric verification.

7. Cloud Storage:

- Online storage services that allow data to be stored and accessed over the internet. In the Smart Attendance System, attendance data may be stored securely in cloud storage.

8. Visualization:

- The presentation of data in graphical or visual formats, such as charts, graphs, or dashboards, to facilitate better understanding and analysis. In the Smart Attendance System, attendance data is visualized to provide insights.

9. Scalability:

- The system's ability to handle a growing number of users, courses, and attendance records without a significant decrease in performance. Scalability ensures the system remains efficient as the user base expands.

10. Notification Preferences:

- User-configurable settings that allow individuals to customize the types and frequency of notifications they receive from the Smart Attendance System. Examples include attendance updates and system notifications.

11. User Manual:

- A document providing step-by-step instructions and guidance on how to use the Smart Attendance System effectively. The user manual assists users in navigating and utilizing various features.

Appendix B: Analysis Models

- **Use Case 1: Log in**
- **Description:** The student or doctor logs into his account.
- **Actors:** Professor or Teaching Assistant or Student.
- **Flow:**
 1. Turn on data or WIFI
 2. Open the system (app, web)
 3. Click on the login.
 4. Enter email and password
- **Note:** we will add screen from user interface
- **Note:** we will add another case

Appendix C: To Be Determined List

1. Technical Decisions:

- Decision: Determine the use of the application or the website
- Status: Pending
- Details: The current use is the website for doctors and the application for students, but both can be done for both doctors and students.

2. Design Choices:

- Choice 1: User Interface Design
- Status: TBD
- Details: The specific design elements and layout for the user interface are yet to be determined. A collaborative effort between the design and development teams is required to create an intuitive and user-friendly interface.

3. System Configuration:

- Configuration 1: Notification Settings
- Status: TBD
- Details: The configuration settings for notifications, including types of alerts and communication channels, are yet to be determined. Input from stakeholders and end-users is needed to define suitable notification preferences.

4. Performance Considerations:

- Consideration 1: Scalability
- Status: Pending
- Details: The scalability requirements for handling a growing number of users and attendance records need to be addressed. Performance testing and discussions on potential bottlenecks are pending.

5. User Roles and Permissions:

- Role 1: Doctor Permissions
- Status: TBD
- Details: The specific permissions and access levels for instructors are yet to be determined. Clarity is needed regarding what actions instructors can perform within the system.

6. Security Measures:

- Measure 1: Data Encryption
- Status: Pending
- Details: Decisions on the implementation of data encryption mechanisms for stored and transmitted data are pending. Security assessments and compliance requirements will influence this decision.

7. User Training:

- Training Module Content
- Status: TBD
- Details: The content and structure of the user training module are yet to be determined. Input from trainers and potential users are needed to ensure comprehensive coverage and effectiveness.

