Face Recognition Attendance Management System

Developed by

Raja Rehan Ahmed
4028-FBAS/BSCS/F18

Sheikh Muhammad Jawad
3965-FBAS/BSCS/F18

Supervised By
Dr. Muhammad Nadeem
Assistant Professor

Department of Computer Science and Software Engineering

International Islamic University, Islamabad

(2022)

FINAL APPROVAL

Dated: _____

It is certified that we have read the project report titled **"Face Recognition Attendance Management System"** submitted by **Raja Rehan Ahmed (4028-FBAS/BSCS/F18)** and **Sheikh Muhammad Jawad (3965-FBAS/BSCS/F18).** We judge that this project is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad for a bachelor's degree in Computer Science.

Supervisor

Dr. Muhammad Nadeem

Assistant Professor

Department of Computer Science & Software engineering

International Islamic University, Islamabad _____

Internal Examiner

Mr. Ahsan Tanveer

Lecturer

Department of Computer Science & Software engineering

International Islamic University, Islamabad _____

External Examiner

Mr. Iftikhar Ali Khan

Lecturer

Department of Computer Science & Software engineering

International Islamic University, Islamabad _____

DECLARATION

We hereby declare that this web application, neither as a whole nor as a part thereof has been copied out from any source, it is further declared that we developed this application entirely based on our efforts made under the sincere guidance of our supervisor and teachers.

No portion of the work presented in this report has been submitted in support of any application for any other degree or qualification of this or any other institute of learning.

Raja Rehan Ahmed
4028-FBAS/BSCS/F18

Sheikh Muhammad Jawad
3965-FBAS/BSCS/F18

ACKNOWLEDGEMENT

First of all, we would like to thank Almighty Allah for His blessing in completing this project in a better way and our project supervisor **Dr. Muhammad Nadeem** for his continuous guidance and support.

Acknowledgment is defined as the art of accepting the truth. We have been absorbed by our academic life for the last few years due to our love of study. The underlying fact is that our friends and family recognized this and rallied behind us. This was constantly proved (and continues to be demonstrated) by **Dr. Muhammad Nadeem,** to whom we are thankful.

Finally, we would like to thank all the teachers of the faculty who taught us during our 4 years in university. Without their guidance and knowledge, we would not be on this step today.

PROJECT IN BRIEF

Title	Face Recognition Attendance Management System
Undertaken By	Raja Rehan Ahmed and Sheikh Muhammad Jawad
Supervised By	**Dr. Muhammad Nadeem** **Assistant Professor**
Started on	October 25, 2021
Completed on	August 20, 2022
Tools Used	Visual Studio Code, PyCharm, pgAdmin 4, Draw.io, Github, Lucidchart, MS Word
Languages	HTML, CSS, Bootstrap, Python, Django (BACKEND), PostgreSQL, face_recognition, OpenCV
System Used	Core i7, 8GB RAM, Windows 10

DEDICATION

Our parents, Teachers, siblings, and seniors are the focus of our endeavor. Without their unwavering support, never-ending prayers, and unwavering faith in us, completing this project would have been a mere dream. Dear Parents, Teachers, Siblings, and Friends, we extend our heartfelt condolences. Thank you so much to everyone for your unwavering support. Without their support, prayer, and unwavering faith in both of us, the completion of this project, which is a milestone on the road to the accomplishment they want us to attain, would have been a mere fantasy in comparison.

ABSTRACT

The growing need for automation in every field is increasing day by day. Life is on a fast track and time is very precious. Traditional methods of attendance in our educational sector are manual and tedious. Face Recognition Attendance Management System is an online web based automated attendance system which takes attendance by recognizing the faces of students in a class and helps to avoid proxies in attendance. The system also handles the check-in and check-out of the teachers. The system provides the facility to store the attendance, manage attendance and view reports.

Admin can register Teachers or they can sign up themselves through the signup page and then go to admin for completing the registration process where their face picture and fingerprint samples are taken. Once a teacher is registered completely and has verified the email, he/she can login to the teacher dashboard and be able to see the courses and attendance of students enrolled in those courses. Teachers can also view their own attendance. Admin can register the students by their face image along with other info.

Face Recognition Attendance Management System uses cutting edge Convolutional Neural Network to detect and recognize a face to mark attendance of concerned individuals.

TABLE OF CONTENTS

Contents

Chapter 1

Introduction

1. Introduction:

Face Recognition Attendance Management System is an online web based automated attendance system which takes attendance by recognizing the faces of students in a class and helps to avoid proxies in attendance. The system also handles the check-in and check-out of the visiting teachers. The system provides the facility to store the attendance, manage attendance and view reports.

Teachers can register themselves through the signup page and then go to admin for completing the registration process where their face picture and fingerprint samples are being taken. Once a teacher is registered completely and verified his/her email, he/she can login to the teacher dashboard to view the courses taught by him/her and student attendance. Teacher can also view his own attendance if his/her status is visiting. The System registers the students by their face image along with other information.

Face Recognition Attendance Management System uses cutting edge Convolutional Neural Network to detect and recognize face to mark attendance of concerned individuals.

1.1 Problem Statement:

In our Educational Sector, it's very hard to manage attendance either on paper or in excel. Teachers have to manually call students' names to mark attendance. There is a high chance of proxies. 75% attendance is mandatory for exam presence is standard in most institutions.

In the manual system, it's very hard to implement a 75% attendance policy. It's also tedious for the Program Office to manage the check-ins and check-outs of teachers.

1.2 Proposed Solution:

So, we have developed a system which can facilitate the attendance process by automating the students' attendance and implementing the 75% attendance policy. The system also automates the check-in and check-out of visiting teachers. Hence it can help to maintain a disciplined environment in the institute.

1.3 Project Scope:

The project is of great value for Schools, Academies, Universities etc. These institutions can automate their manual attendance system to this modern and automated system and reduce the workload for the management and the teachers.

This is a web-based project. Teachers can view student's attendance which are enrolled in the courses taught by them and also their own attendance if their status is visiting. The data is stored in a centralized database e.g., user's data, students' data, s, attendance records etc.

1.4 Major Modules:

The major modules of this system include: -

1.4.1 User Management

Our users are Admin i.e., Program Office, and Teacher both Visiting and Permanent. Students are not users; they are only stakeholders.

1.4.2 Attendance

Attendance of Students will be taken through Face Recognition. Check in and check out of teachers will be taken either through Face Recognition or Fingerprint Recognition.

1.4.3 Face Detection and Recognition

AI and machine learning models rely on access to high-end training data. We have used state of the art python library face_recognition to detect faces in the image, extract encodings and then recognize a face by comparing the encodings with known encoding.

1.4.4 Fingerprint Recognition

Fingerprint recognition is achieved by using Scale Invariant Feature Transform (SIFT) by identifying unique key points in the fingerprint image and their descriptors and then comparing them with known key points.

1.4.5 Attendance Reports

This module handles all the attendance reports. It gets the user input and generates reports based on that input. This module also has the feature of downloading the reports as a CSV.

1.5 Project Objectives:

The main goal of our venture is to transfer the manual attendance system of students and the check-in and check out of teachers to an automated system that is fault-tolerant, flexible, and easy to use.

The objectives are as follows: -

- Take Attendance in all the rooms of the department with the only click of a button from the Program Office
- Remove Proxies from the attendance
- Reliable and user-friendly system both for management and teachers
- Saving time by automating the process
- Minimum effort for taking attendance
- Bring students attendance and teachers check-in, check out under one umbrella
- Easy Operations will be used
- All attendance data, students' data, and teachers' data will be stored in the database

1.6 Project limitations:

Project limitations are given below:

- Active internet connection.
- All students must be looking straight to the camera at the time of attendance.
- Students should not be too far away from the camera.
- Good lighting conditions in classrooms.
- Good quality cameras.

Chapter 2

Existing Systems

2. Existing Systems:

This is a web-based system designed to manage attendance of both students and visiting faculty. The students' attendance of all classes is initiated by the Program Office with one click which triggers the system to mark the attendance of students according to the timetable, courses, and student enrollments. The check in and checkout of the visiting teachers is marked either through face or fingerprint recognition.

The face image of each student is taken by the system to register a new student. While both face image and fingerprint impressions of visiting teacher are captured by the system for their registration. This data is used to create profiles of students and teachers.

This project covers the vast domain of Educational Institutes including University, Colleges, Schools etc. Using our system, they can efficiently handle the attendance. The system eliminates the proxies from the attendance and makes it easier for the organization to manage the attendance without any headache of manually taking it on paper or on an excel sheet.

2.1 RFID (Radio Frequency Identification) based Recognition System

A Radio Frequency Identity Card must be carried by the student in the RFID-based system in order to record their attendance for the day. The ID must be placed on the card reader. The system has the ability to connect to RS232 and record attendance in a database that has already been recorded. There is a chance that fraudulent access might take place. When a certain student is missing, some students may use the ID of another student to prove their attendance, or they may even attempt to misuse it occasionally.

The drawbacks of this system are as follows:-

- Materials like metal & liquid can impact signal
- Accuracy is not optimal
- Highly Expensive
- Implementation is difficult & time-consuming

2.2 Fingerprint based Recognition System

A portable fingerprint device that is already set up with the student's fingerprint is required for the existing fingerprint-based attendance system. To confirm their attendance for the day, the student must register their fingerprint on the set device later, either during or before the lecture hours. The issue with this strategy is that it might divert students' attention throughout the lesson. The drawbacks of this system are as follows:-

- The system has an inability to enroll some users.
- The accuracy and working of the system are affected by the skin conditions of people.
- The system is associated with forensic applications.
- There are health issues involved due to the touching of a single scanning sensor device by countless individuals. This has serious implications in pandemic situations such as COVID-19 etc.
- It is difficult to capture complete and accurate fingerprint images in some cases based on age and occupation.
- The collection of high-quality nail-to-nail images needs training and specific skills.

2.3 Iris based Recognition System

The student must stand in front of a camera for the Iris scanning portion of the Iris based student attendance system. The data of the student contained in the database is compared with the scanned iris, and the attendance on their presence needs to be updated. As a result, the institute's faculty members have less work to do with paper and pencil. This lessens the possibility of proxies in the class and aids in keeping the student records secure. It is a wireless biometric method that addresses the issue of fictitious attendance and the difficulty of constructing the necessary network.

The drawbacks of this system are as follows:-

- If the subject is unconscious, the authentication will fail because the scanner detects the movement of the iris.
- It requires a minimum distance between the scanner and the human eye. It cannot scan if the distance is high.
- It requires an IR light source and sensor. It can't use a regular camera.
- Visible light should be minimized for the highest accuracy.
- Eyelashes, lenses, and reflections can cause obstacles.

2.4 Face based Recognition System

A high-resolution digital camera that can detect and recognize faces of students can be used to record attendance using facial recognition technology. The computer matches the identified face with student face photos recorded in the database. The attendance is noted in the attendance database for further calculation once the student's face is matched with the image that has been saved. If the acquired image doesn't match the student's face that is already stored in the database, a new image is added to the database. In this system, the main problem is that every student has to stand in front of the camera separately for attendance which is very time consuming and inefficient if the number of students is very large.

The drawbacks of this system are as follows:-

- Expensive Camera
- Difficult Implementation for a big organization
- No Manual attendance allowed
- Complex UI
- Old Face Recognition techniques being used

Chapter 3

System Analysis

3. System Analysis:

Software Development Life Cycle (SDLC) is a process in Software Engineering that defines various stages and steps involved in the development of high-quality deliverable software.

The different phases of SDLC are given below.

- Requirement elicitation
- Analysis of requirements
- Designing the System
- Implementation
- Verification and Validation
- Testing
- Deployment
- Maintenance

Let's see each step of the SDLC for our system (FRAMS) separately, starting from requirements elicitation and analysis.

3.1 Requirement Analysis:

In the first phase of the SDLC, we need clear-cut requirements about the system, that's what we need to implement in the system and what we don't. After completing this stage, we have a clear picture of the system in mind and we start designing the system.

In the case of our system, we needed to elicit the requirements from the Program Office, Teachers and Students, because these are the stakeholders of the system.

Program office has been consulted many times to gather requirements that we have analyzed and then designed our system in the following way.

3.2 Use-Case Modeling:

The Use-cases of the system describes the proposed functionality of the whole system and the actors that can use certain functionality. In short, use case modeling is the graphical representation of requirements that we have elicited. This model shows which actor/user can interact with the system in certain ways.

3.3 Actors:

The list of actors is given below.

3.3.1 Admin / Program Office

Admin is responsible for all the administrative activities.

The major activities involve:-

- Registering Students & Teachers
- Taking Student Attendance of all the classes & marking Teachers' check in & checkout
- Downloading Reports
- Managing and Updating Database

3.3.2 Teacher

Teacher can see the attendance records of his own as well as of the students enrolled in courses taught by him/her. He can also view individual course reports as well as collective student reports and download these record in a CSV.

Above are the actors of our system. Now we are going to show you their interaction with the system with the help of a use case diagram.

3.4 Use Case Diagram:

Following is the user case diagram of the system. There are two main actors of the system, i.e., Admin / Program Office, Teacher.

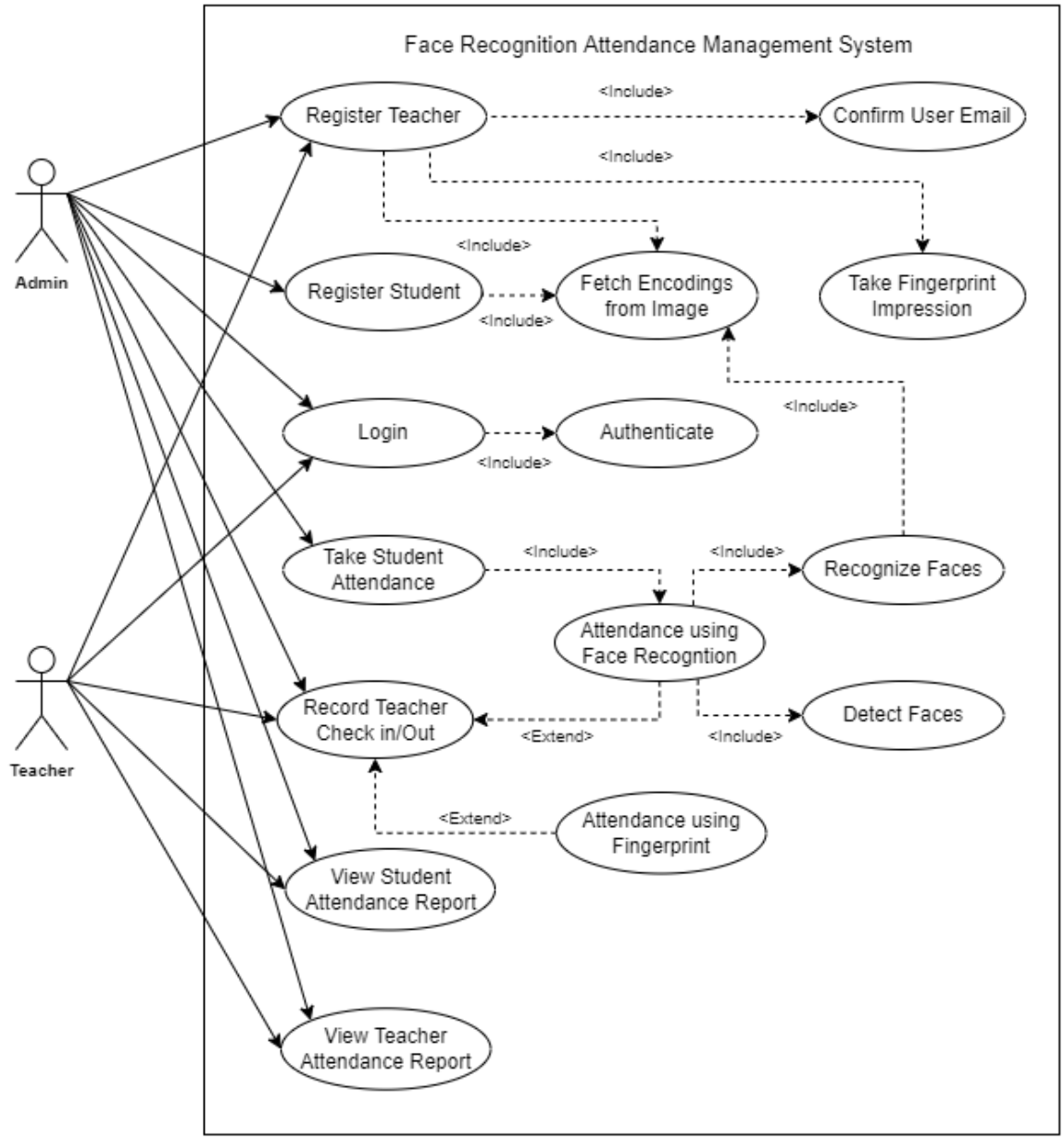

Figure 3.1: System Use Case Diagram

3.5 Use Cases in Brief Format:

Register User

Use Case Name	Register User
Actor	Admin, Teacher
Type	Primary
Description	Make a user profile to be able to use the system.

Login

Use Case Name	Login
Actor	Admin, Teacher
Type	Primary
Description	Verify the user and give access to the system.

Register Student

Use Case Name	Register Student
Actor	Admin
Type	Primary
Description	Register the student in system in order to take the attendance later.

Take Student Attendance

Use Case Name	Take Student Attendance
Actor	Admin
Type	Primary
Description	Admin takes the attendance of all the classes by going to attendance page and pressing the submit button.

Mark Teacher Check in, Check out

Use Case Name	Mark Teacher Check in, Check out
Actor	Admin, Teacher
Type	Primary
Description	Admin accesses the Visiting Teachers Attendance Page; System marks check-in or check-out of Visiting Teacher either through thumb impression or face recognition with a timestamp.

View Student Attendance Report

Use Case Name	View Student Attendance Report
Actor	Admin, Teacher
Type	Secondary
Description	User accesses the Student Attendance Report page and enters the query information; System generates and displays the report based on query information.

View Teacher Attendance Report

Use Case Name	View Teacher Attendance Report
Actor	Admin, Teacher
Type	Secondary
Description	User accesses the Teacher Attendance Report page and enters the query information; System generates and displays the report based on query information.

Confirm User Email

Use Case Name	Verify Medicines
Actor	Admin, Teacher
Type	Primary
Description	Teacher clicks the link emailed to him/her on registration. System verifies the link and makes the Teacher profile active.

Take Fingerprint Impression

Use Case Name	Take Fingerprint Impression
Actor	Admin, Teacher
Type	Primary
Description	Admin accesses the Register User Page; System asks to provide fingerprint samples along with other User's data if the User is a Visiting Teacher. Admin enters the fingerprint samples of the Teacher. System registers the user.

Take Face Image Sample

Use Case Name	Take Face Image Sample
Actor	Admin, Teacher, Student
Type	Primary
Description	Admin accesses the Register User Page; System asks to provide a face image sample along with other User's data if the User is a Visiting Teacher; Admin enters the face image sample of Teacher. Or Admin accesses the Register Student Page; System asks to provide the face image sample along with other Student's data; Admin enters the face image sample of the student; System registers the user / student.

Fetch Encodings from Image

Use Case Name	Fetch Encodings from Image
Actor	Admin, Teacher
Type	Primary
Description	Admin access the Register User Page or Add Student Page; System extracts the encodings from face image and also from fingerprint samples while registering a Visiting Teacher and stores them in the database for recognizing the Visiting Teachers and Students for attendance.

Authenticate

Use Case Name	Authenticate
Actor	Admin, Teacher
Type	Primary
Description	System receives a Login request with username and password; System authenticates the User by matching the username and password with the User's data in the database. If the User's data matches, the User gets authenticated.

Take Attendance Using Face Recognition

Use Case Name	Take Attendance Using Face Recognition
Actor	Admin, Teacher, Student
Type	Primary
Description	Admin accesses the Take Student Attendance Page; The system marks the students' attendance by recognizing faces from the classroom images following a timetable and course enrollments. Or Visiting Teacher comes to the Program Office for Check-in or Check-out. Admin accesses the Visiting Teacher Check-in and Check-out Page; Teacher sees straight to the camera. System marks the Check-in or Check-out of Visiting Teacher by recognizing the face.

Detect Faces

Use Case Name	Detect Faces
Actor	Admin
Type	Primary
Description	System gets an image; System detects the faces from the image using face_recognition HOG method; System crops the detected faces from the picture and forwards them to the recognizer for recognition.

Recognize Faces

Use Case Name	Recognize Faces
Actor	Admin
Type	Primary
Description	System receives the detected face; System extracts the face encodings and matches with faces already present in the database. If the face gets a match in the database, then the attendance of that Student or Visiting Teacher gets marked.

Take Attendance Using Fingerprint

Use Case Name	Take Attendance Using Fingerprint
Actor	Admin, Teacher
Type	Primary
Description	Teacher comes to the Program Office for check-in or check-out. Admin accesses the Teacher Check-in and Check-out Page; Teacher presses his thumb on the Fingerprint Scanner. System marks the Check-in or Check-out of the recognized Visiting Teacher.

3.6 Use Cases in Extended Format:

The expanded format of use cases describes the sequence of messages from the actors of the system and the system's response to each message during the execution of a use case. This format also shows the structure and content of the message passed between actors and the system.

The following are the extended formats of use cases:

Register User

Use Case Name	Register User.
Actor	Admin, Teacher
Purpose	Make a user profile to be able to use the system.
Overview	Teacher must register. Admin already has access.
Type	Primary
Pre-Condition	Admin is logged in.
Typical Course of Events	

Actor Actions	System Response
iii. User enters the required information, and if required photo and fingerprint samples into the system.	i. Displays registration form. ii. System asks to provide details about the Teacher including an email ID, username and password and if the Teacher is a Visiting Teacher and Admin is registering Teacher, System also asks to take face and fingerprint samples of the Teacher. iv. System verifies the form data. v. Verification successful.
Post-Condition	The user registered successfully and accessed the functionalities related to it. If the Teacher himself/herself registered through the signup page, System got the pending registration request.

Login

Use Case Name	Login.
Actor	Admin, Teacher
Purpose	Verify the user and give access to the system.
Overview	Teacher must register. Admin already has access.
Type	Primary

Pre-Condition	User is registered already.
Typical Course of Events	
Actor Actions	**System Response**
ii. Enter credentials. v. Direct to home page.	i. Displays the login form. iii. System verifies and authenticates the User. iv. Authentication successful.
Post-Condition	User logged in successfully.

Register Student

Use Case Name	Register Student
Actor	Admin
Purpose	Register the student in the system in order to take the attendance later.
Overview	System registers the student along with face picture in order to recognize the student at the time of attendance.
Type	Primary
Pre-Condition	Admin is logged in.
Typical Course of Events	
Actor Actions	**System Response**

iii. Admin enters the required information.	i. Displays the student registration form. ii. System asks to provide the student's information including a sample face image. iv. System verifies the form data. v. Verification successful.
Post-Condition	Student registered successfully.

Take Student Attendance

Use Case Name	Take Student Attendance.
Actor	Admin, Student
Purpose	Mark student attendance.
Overview	Admin takes the student attendance of all the classes.
Type	Primary
Pre-Condition	i. Admin is logged in. ii. All students are registered and the system has face image samples of all the students. iii. Students are present in their assigned classroom and looking straight at the camera at the time of attendance.
Typical Course of Events	
Actor Actions	**System Response**

i. Admin accesses the Take Attendance Page. iii. Admin presses the Take Attendance Button and send the classroom images to server.	ii. Displays Student Attendance Page. iv. System uses the classroom images of all classes to mark the attendance of students by detecting and recognizing students' faces from images following a timetable and course enrollments.
Post-Condition	Attendance of Students from all classes is marked successfully.

Mark Teacher Check in, Check out

Use Case Name	Mark Teacher Check in, Check out.
Actor	Admin, Teacher
Purpose	Keep record of teacher attendance.
Overview	Mark teacher check in and checkout.
Type	Primary
Pre-Condition	i. Admin is logged in. ii. Teacher is already registered and System has already stored the encodings of the face and fingerprint impression of the teacher.
Typical Course of Events	
Actor Actions	**System Response**

i. Admin accesses the Teacher check in, checkout page.	ii. System marks check-in or check-out of Visiting Teacher either through thumb impression or face recognition with a timestamp.
Post-Condition	Teacher checked in or checked out successfully.

View Student Attendance Report

Use Case Name	View Student Attendance Report
Actor	Admin, Teacher
Purpose	Show the student(s) attendance data.
Overview	Admin or teacher views the student(s) report.
Type	Secondary
Pre-Condition	User is logged in.
Typical Course of Events	
Actor Actions	**System Response**
i. User access the Student Attendance Report Page. iii. User enters their query information.	ii. System asks for query information about the required report. iv. System generates and displays the required Attendance Report.
Post-Condition	System successfully generates and displays the Attendance Report.

View Teacher Attendance Report

Use Case Name	View Teacher Attendance Report
Actor	Admin, Teacher
Purpose	Show the teacher attendance data.
Overview	View the teacher attendance report.
Type	Secondary
Pre-Condition	i. User is logged in. ii. Teacher is already registered.
Typical Course of Events	
Actor Actions	**System Response**
i. User accesses the Teacher Attendance Report Page. iii. User enters their query information.	ii. System asks for information about the required Report. iv. System generates and displays the required attendance report.
Post-Condition	System successfully generates and displays the Attendance Report.

Confirm User Email

Use Case Name	Confirm User Email
Actor	Admin, Teacher
Purpose	Verify and validate the email and make the Teacher active and be able to login.
Overview	A unique link is sent to the Teacher email. Teacher clicks on the link and verifies his/her email.
Type	Primary
Pre-Condition	Admin is logged in.
Typical Course of Events	
Actor Actions	**System Response**
ii. Teacher clicks the sent link.	i. System sends a confirmation link to the provided email address. iii. System verifies the link. iv. System makes the Teacher profile active.
Post-Condition	Teacher email is verified successfully.

Take Fingerprint Impression

Use Case Name	Take Fingerprint Impression
Actor	Admin, Teacher
Purpose	Take fingerprint impression for registering a Teacher or marking Teacher check in, check out.

Overview	Fingerprint impression is taken and stored in the database.
Type	Primary
Pre-Condition	Admin is logged in.
Typical Course of Events	
Actor Actions	**System Response**
i. Teacher presses his finger on the fingerprint scanner.	ii. System records the fingerprint and stores it if it's intended for Teacher registration or matches it with stored fingerprints if it's intended for teacher check in, checkout.
Post-Condition	Fingerprint impression is successfully taken.

Take Face Image Sample

Use Case Name	Take Face Image Sample
Actor	Admin, Teacher, Student
Purpose	Take face image sample for registering teacher or marking teacher check in, check out.
Overview	Face image sample is taken and stored in the database.
Type	Primary
Pre-Condition	Admin is logged in.
Typical Course of Events	
Actor Actions	**System Response**

i. Admin enters the face image sample.	ii. System fetches the image and stores it if it's intended for Student or Teacher registration or matches it with stored face samples if it's intended for teacher check in, check out.
Post-Condition	Image is taken and saved.

Fetch Encodings from Image

Use Case Name	Fetch Encodings from Image
Actor	Admin
Purpose	Make the system able to recognize students or teachers at the time of attendance.
Overview	Fetch the encodings from face image or both face and fingerprints while registering a teacher.
Type	Primary
Pre-Condition	Admin is logged in.
Typical Course of Events	
Actor Actions	**System Response**
i. Admin enters the face image.	ii. System extracts the encodings from face image and also from fingerprint samples while registering a Visiting Teacher and stores them in the database for recognizing the Visiting

	Teachers and Students for attendance.
Post-Condition	Encodings are fetched and stored in the database successfully.

Authenticate

Use Case Name	Authenticate
Actor	Admin, Teacher
Purpose	Authenticate the user and let him/her login.
Overview	Verify the username and password entered to authenticate the user.
Type	Primary
Pre-Condition	System received a login request.
Typical Course of Events	
Actor Actions	**System Response**
i. User sends the login request.	ii. System authenticates the User by matching username and password with User's data in the database. iii. If the User's data matches, the User gets authenticated and logged in.
Post-Condition	User is logged in and redirected to his/her specific Homepage.

Take Attendance Using Face Recognition

Use Case Name	Take Attendance using Face Recognition
Actor	Admin, Teacher, Student
Purpose	Taking attendance using face recognition automates the attendance process.
Overview	Classroom images or Teacher image is sent to the System. System marks the attendance by detecting and recognizing face(s) following a timetable in case of student attendance.
Type	Primary
Pre-Condition	Admin is logged in.
Typical Course of Events	
Actor Actions	**System Response**
i. Admin enters the teacher image or classroom images in case of student attendance.	ii. The system marks the students' attendance by recognizing faces from the image(s) coming from all the classes following a timetable and sample course enrollments. iii. Or the system marks the check in or check out of the Teacher by recognizing the face.
Post-Condition	Attendance is marked successfully.

Detect Faces

Use Case Name	Detect Faces
Actor	Admin
Purpose	Detect faces and make the image ready to fetch face encodings.
Overview	System gets a picture; system detects the faces present in the picture.
Type	Primary
Pre-Condition	Admin is logged in.
Typical Course of Events	
Actor Actions	**System Response**
i. Admin enters image(s).	ii. System detects the faces from the picture(s) using the HOG (Histogram of Oriented Gradients) method. iii. System extracts the face encodings from the detected faces for further processing.
Post-Condition	Faces are detected accurately and successfully.

Recognize Faces

Use Case Name	Recognize Faces
Actor	Admin
Purpose	Recognize faces to mark attendance of Students or

	Teachers.
Overview	System gets the detected faces; system recognize the faces by comparing with known face encodings present in the system.
Type	Primary
Pre-Condition	i. Admin is logged in. ii. System receives detected faces.
Typical Course of Events	
Actor Actions	**System Response**
i. Admin enters the image(s).	ii. System extracts the encodings of the face(s) from the image(s) and matches it with the encodings of all visiting teachers if the attendance is of a visiting teacher and with encodings of all the students enrolled in the course of which attendance is to be taken. If a face gets a match in the database, then the attendance of that Student or check-in or check-out of the Visiting Teacher gets marked.
Post-Condition	Faces are recognized accurately and successfully.

Take Attendance Using Fingerprint

Use Case Name	Take Attendance using Fingerprint
Actor	Admin, Teacher
Purpose	Taking attendance using fingerprint automates the

	attendance process.
Overview	Teacher presses the finger on the fingerprint scanner. System marks the attendance by comparing the fingerprint with fingerprints of registered visiting teachers.
Type	Primary
Pre-Condition	i. Admin is logged in. ii. Teacher comes to the Program office for check-in or check-out.
Typical Course of Events	
Actor Actions	**System Response**
i. Visiting Teacher comes to the Program Office for Check-in or Check-out. ii. Admin accesses the Teacher check in and check out Page. iv. Teacher presses his thumb on the Fingerprint Scanner.	iii. Displays the Teacher check in and check out page. v. System marks the check in or checkout of the Teacher by comparing the fingerprint with fingerprints of all the visiting teachers registered in the system.
Post-Condition	Attendance is marked successfully.

Chapter 4

System Design

4 System Design

The process of defining the architecture, product design, modules, interfaces, and data for a system in order to meet specific criteria is system design. The application of system theory to product development is known as system design. The discipline of system analysis, system architecture, and systems engineering have some overlap.

System design is the process of defining the elements of a system such as the architecture, modules, and components, the different interfaces of those components, and the data that goes through that system. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

System design implies a systematic approach to the design of a system. It makes a bottom-up or top-down approach, but either way, the process is systematic wherein it considers all related variables of the system that needs to be created from the architecture to the required hardware and software, right down to the data and how it travels and transforms throughout its travel through the system. System design then overlaps with systems analysis, systems engineering, and system architecture.

The system design approach first appeared right before World War II, when engineers were trying to solve complex control and communication problems. They needed the ability to standardize their work into a formal discipline with proper methods, especially four new fields like information theory, operations research, and computer science in general.

4.1 System Sequence Diagrams

4.1.1 Register Teacher

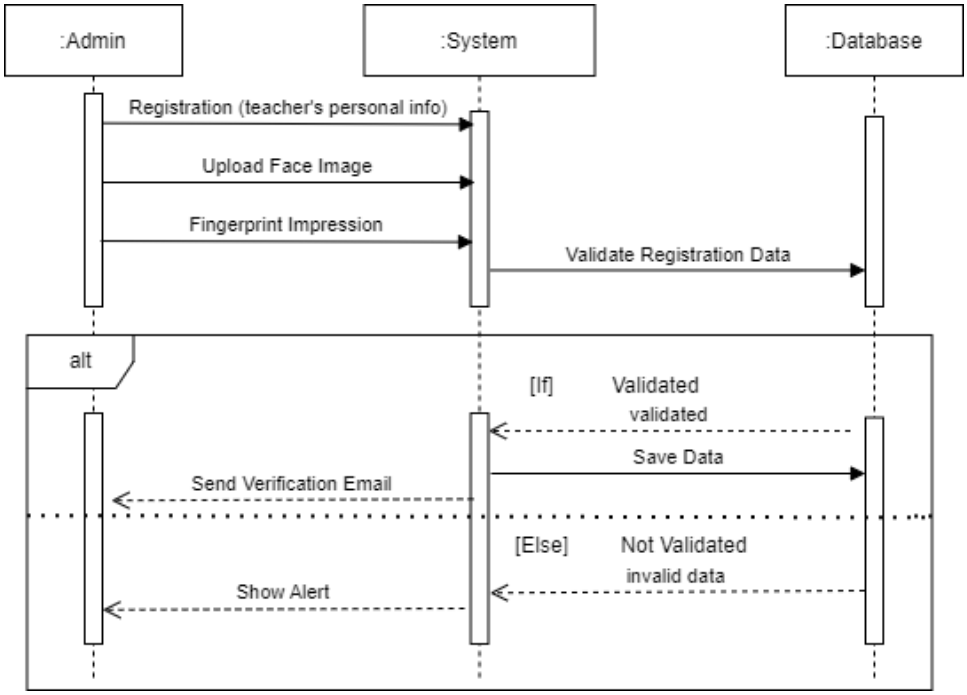

Figure 4.1: System Sequence Diagram

4.1.2 Register Student

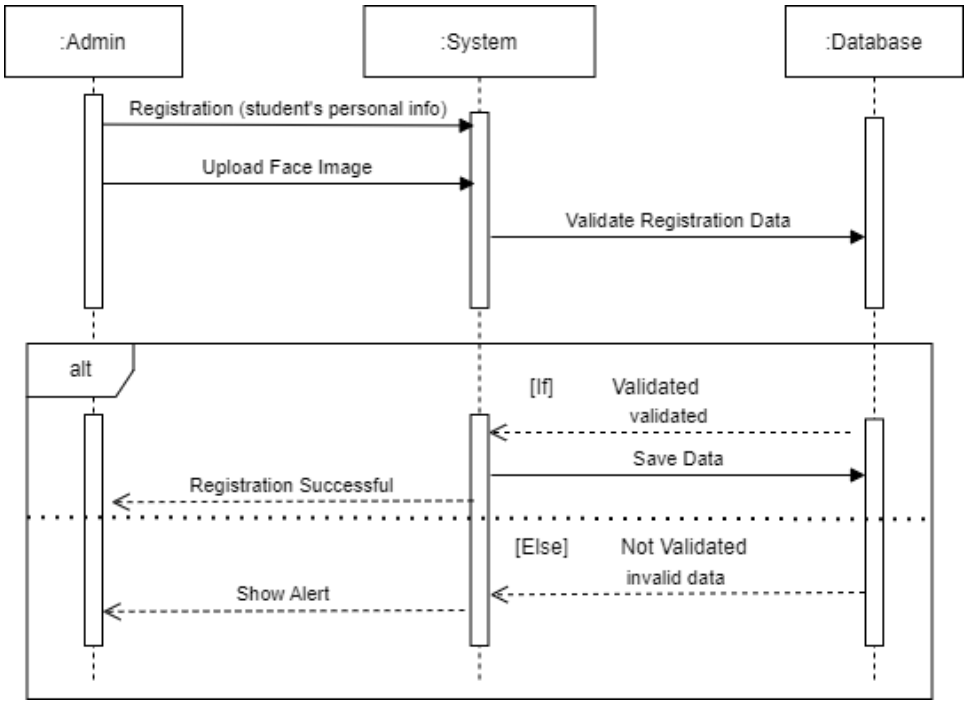

Figure 4.2: System Sequence Diagram

4.1.3 Login

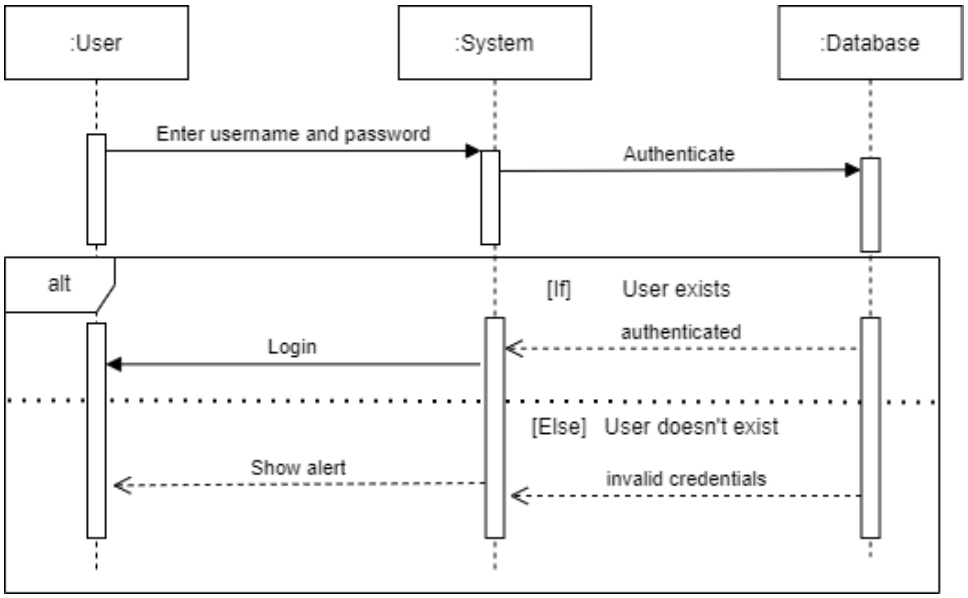

Figure 4.3: System Sequence Diagram

4.1.4 Take Student's Attendance

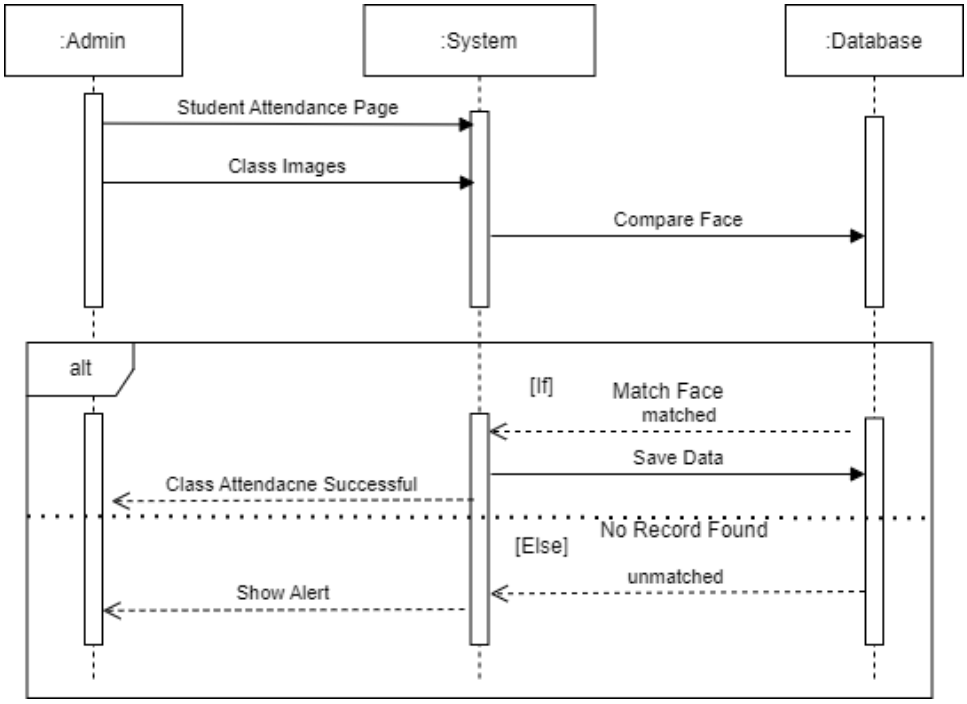

Figure 4.4: System Sequence Diagram

4.1.5 Record Visiting Teacher's Check-in/Check-out

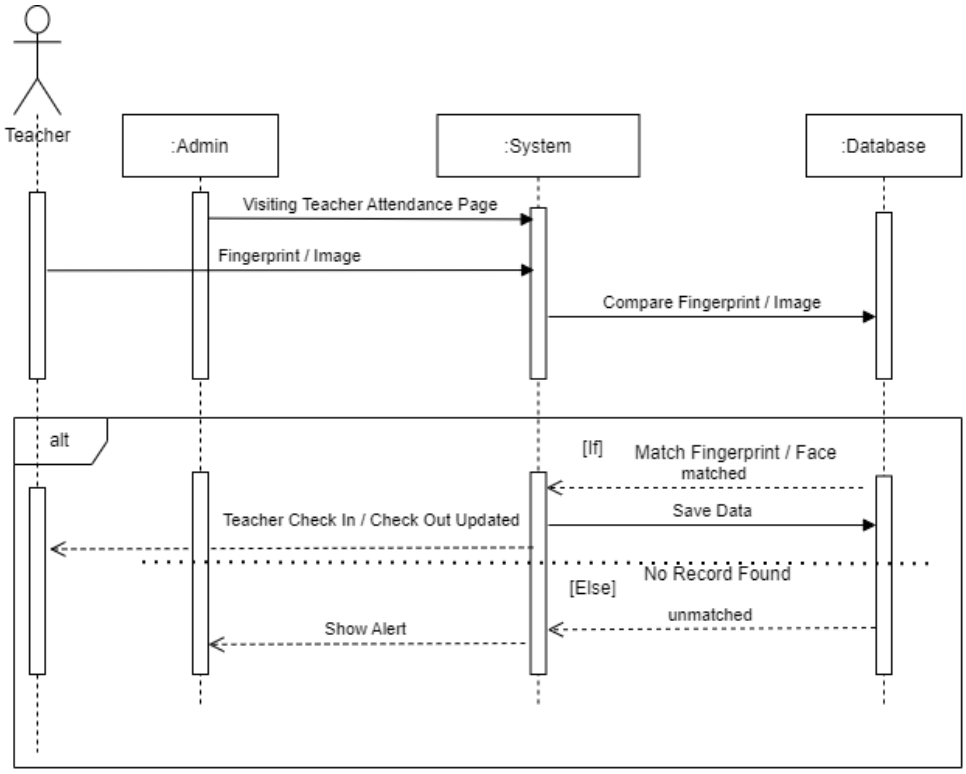

Figure 4.5: System Sequence Diagram

4.1.6 View Student's Attendance

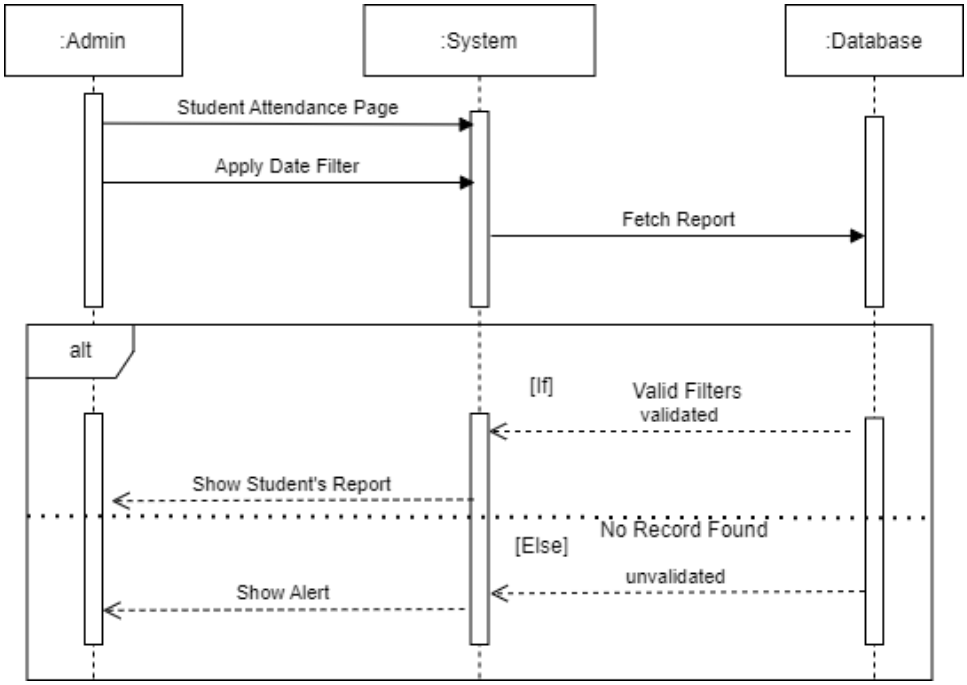

Figure 4.6: System Sequence Diagram

4.1.7 View Visiting Teacher's Attendance

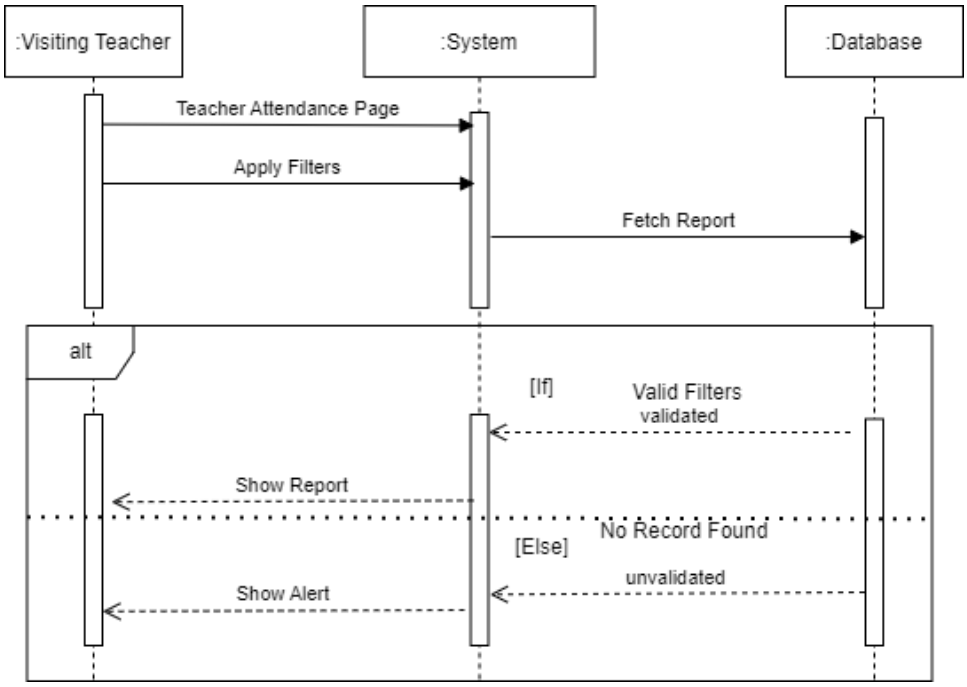

Figure 4.7: System Sequence Diagram

4.2 ERD

Databases are absolutely an integral part of the software system. To fully utilize ER Diagram, database engineering guarantees you to produce high-quality database design to use in database creation, management, and maintenance. An ER model also provides a means for communication.

Entity Relationship Diagram, also known as ERD, ER Diagram, or ER model, is a type of structural diagram for use in database design. An ERD contains different symbols and connectors that visualize two important information "The major entities within the system scope, and the inter-relationships among these entities". And that's why it's called "Entity Relationship diagram.

When we talk about entities in ERD, very often we are referring to business objects such as people/roles (e.g., Student), tangible business objects (e.g., product), intangible business

objects (e.g., Log), etc. “Relationship” is about how these entities relate to each other within the system.

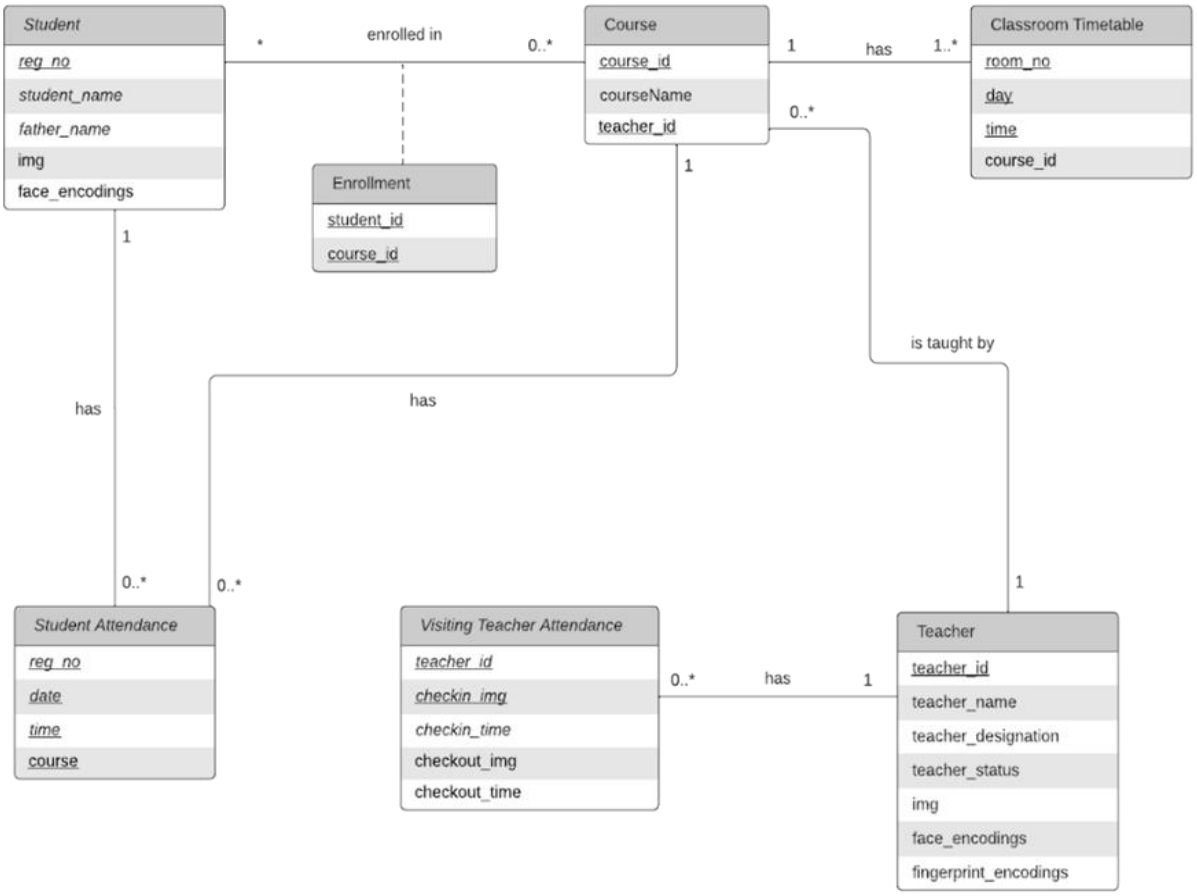

Figure 4.8: Entity Relationship Diagram

Chapter 5

System Implementation

5. System Implementation

This is the core part of our application lifecycle. The phase describes the technology stack being used to implement the project successfully. This web application is built using the Django full stack framework which is widely used python framework. It is based on the Model View Template architecture. 5.1 Tools and Technologies

5.1 Technologies

Following technologies are being used in the project:-

5.1.1 HTML

HTML stands for HyperText Markup Language. It is used to create the user interface of this project. The majority of the frotend is being built in HTML. We have used HTML in this project.

5.1.2 CSS

CSS stands for cascading style sheets. It is used to make the web page attractive. It is used to describe layouts, colors, font, etc. We have used the most interactive CSS styling to make our project more appealing.

5.1.3 JAVASCRIPT

JavaScript is a programming language used for both client and server sides. JS enables us to add dynamic behavior to the webpage.

5.1.4 Bootstrap

Bootstrap enables designers and developers to build completely responsive websites quickly. It can be considered the most popular CSS framework for developing responsive applications.

5.1.5 Python

Python is a high-level, interpreted, general-purpose programming language. It's design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected

5.1.6 Django

Django is a free and open-source, Python-based web framework that follows the model–template–views architectural pattern.

5.1.7 PostgreSQL

PostgreSQL which is an advanced, enterprise class open-source relational database that supports both SQL (relational) and JSON (non-relational) querying.

5.2 Algorithms / Libraries Used

5.2.1 Face Detection using HOG (Histogram of Oriented Gradients)

This technique breaks up the image into small squares of pixels replacing pixels or the square with arrows which shows in which direction the image is getting darker. These arrows are called gradients. These gradients show the flow of light in the image. By doing this process we get a very simple representation of the image showing a basic structure of a face. Now to detect a face, we've to identify the part of the HOG image which looks the same as the known HOG pattern fetched from tons of training faces.

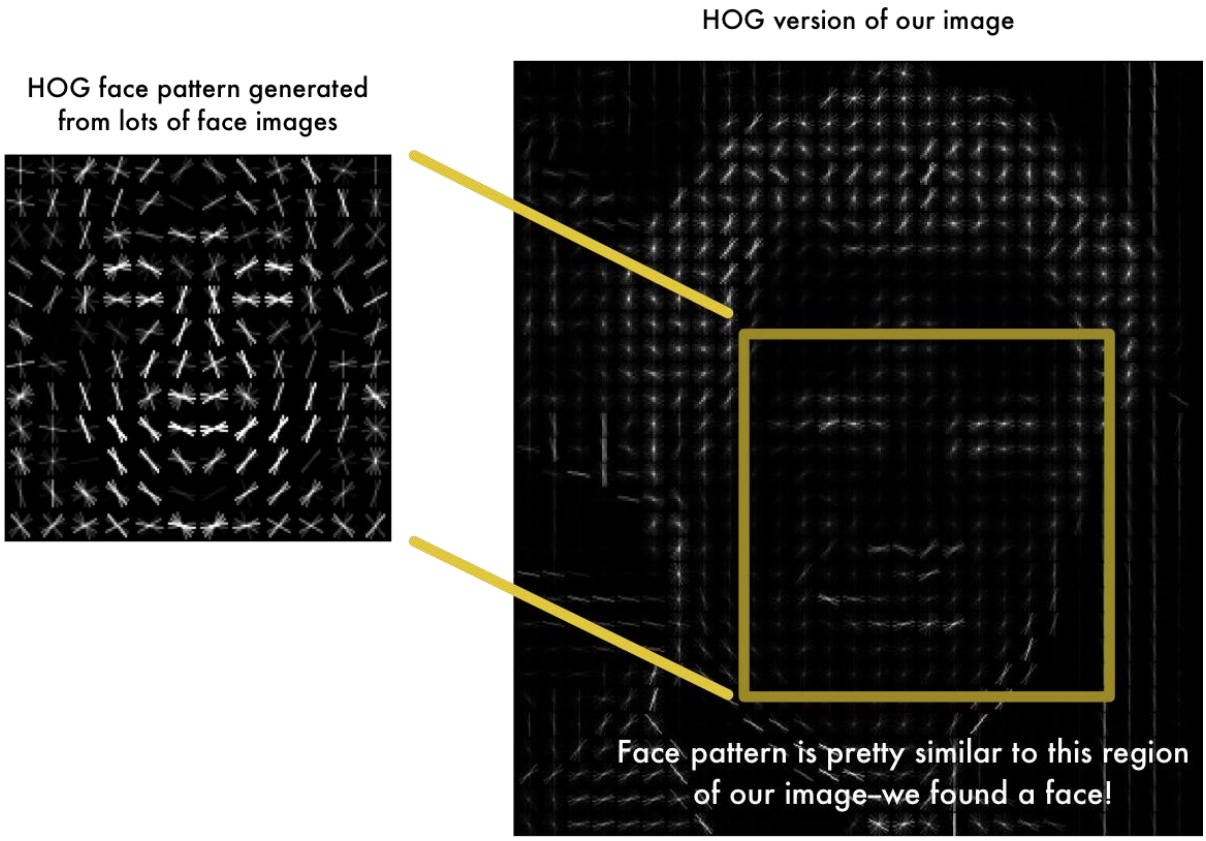

Figure 5.1: HOG Face Pattern

5.2.2 Encoding a Face Image

The process of training CNN (Convolutional Neural Network) to make it capable of extracting the face embeddings or encodings demands a lot of processing and training data and it might take more than a day with high-end hardware to train the model to achieve high accuracy. However, this part is already done and we have a trained model from the face-recognition library for extracting the face embeddings. We give the image and optionally face locations in the image and it finds and returns 128 face measurements for each face in the image as numpy.ndarray.

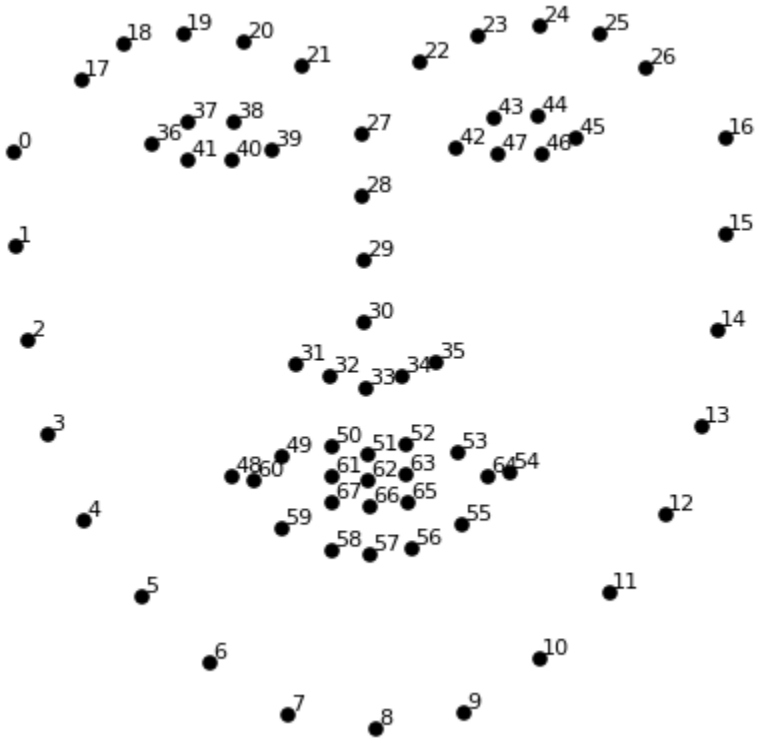

Figure 5.2: 128 Unique Face Measurements

5.2.3 Classifying a Face

This process is a lot simpler than the previous ones. We again use the face_recognition library for comparing the extracted encodings from the test image with known encodings. It takes a list of known face encodings and the face encoding to check and optionally tolerance which is the distance between faces to consider it a match. Lowering the tolerance increases the strictness of comparing. 0.6 is the optimal value for tolerance. This returns a list of True / False values which specifies which known encodings are matched with the face encoding to check.

5.3 Student Attendance Workflow

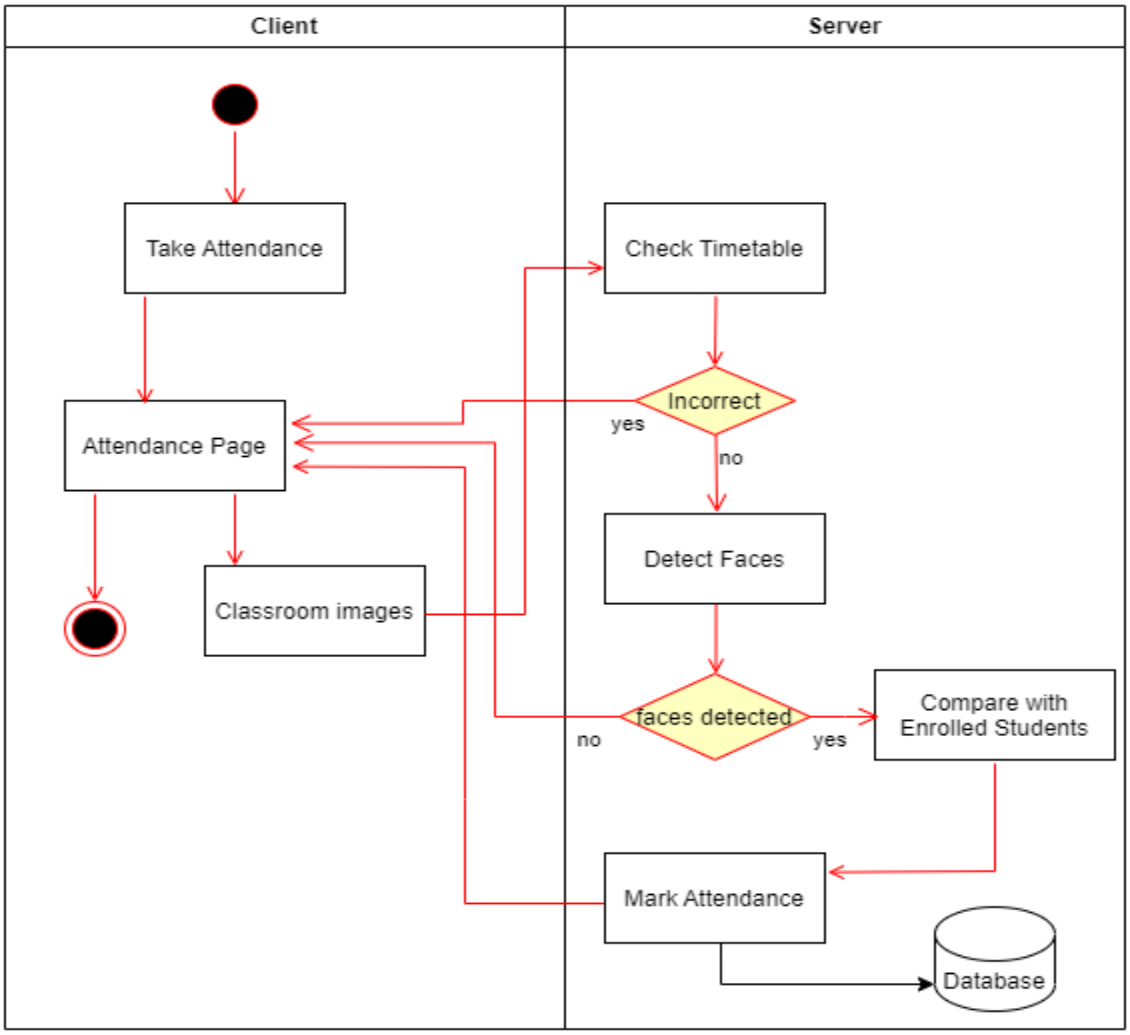

Figure: 5.3: Student Attendance Workflow

5.4 Student Attendance Pseudocode

```
for classroom in ClassRoom.objects.all() {
    course = classroom.course
    img = cv2.imread(f'{path}/{filename}')
    faces = face_recognition.face_locations(img)
    encodings_test = face_recognition.face_encodings(img, faces)

    for student in students_enrolled {
        encodings_known.append(student.encodings)
    }
    for encoding in encodings_test {
        matches = compare_faces(encodings_known, encoding)
        if matches {
            presentCount+=1
            attendance=StudentAttendance(student=student,class_timing=classTime,time=now, course=course, status='P')
            attendance.save()
        }
    }
    for student in students_enrolled {
        alreadyMarked=StudentAttendance.objects.filter(student=student,class_timin=classTime, time__year=now.year, time__month=now.month, time__day=now.day, course=course)
        if alreadyMarked.count < 1 {
            attendance = StudentAttendance(student=student, time=now, class_timing=classTime, course=course, status='A')
            attendance.save()
        }
```

```
            else if alreadyMarked.count() == 1 {
                marked = alreadyMarked
                if marked.status == 'A':
                marked.time = now
                marked.save()
            }
        }
}
```

5.5 Deployment Diagram

A deployment diagram is used to describe the connection between software and hardware components of the system and the physical distribution of the processing. In our system, the software component is a user interface connected to a hardware component that is served through the internet.

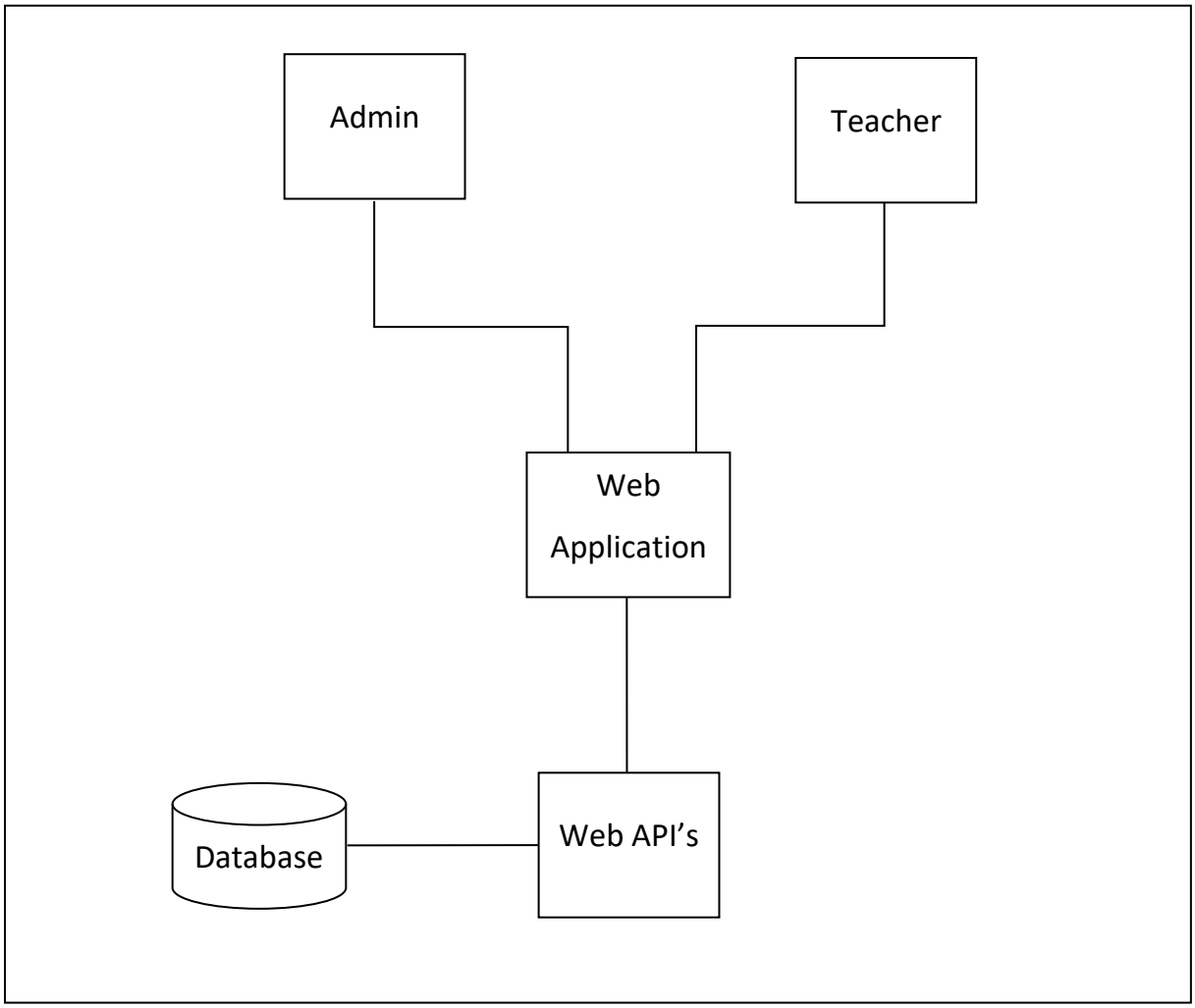

Figure 5.4: Deployment Diagram

Chapter 6

System Testing

6. System Testing

Software testing comes next after the development stage. This is done to test the system to detect possible errors and uncertainties removing them. I have tested all use cases and each is approved. Software testing ensures both development and user satisfaction and provides quality to all stakeholders concerned.

6.1 Functional Testing

Function testing is a type of software testing used to verify the validity of a software application against the user's needs. The purpose of functional testing is to verify every functionality in the software by finding the appropriate input and correcting the output to formal requirements.

6.2 White Box Testing

In white box testing, we find the internal structure or workings of an application. We have nothing to do with code in white box testing.

.

6.3 Black Box Testing

In this testing technique, we have checked the errors and bugs using code manipulation. Like checking output for standard input to verify if things go fine or not.

6.4 Test Cases

TC 01

Test Case ID:	TC–01
Test Case Title:	Signup
Description:	Testing the sign-up process.
Pre-conditions:	Web Application is up.

Step No.	Action to perform	Expected Result(s)	Status
1	Click on sign up button.	Signup form displayed.	Pass
2	Fill the form and click on sign up button.	New user request sent to admin.	Pass

TC 02

Test Case ID:	TC–02		
Test Case Title:	Approve Pending Registrations.		
Description:	Testing the pending registration request.		
Pre-conditions:	1. Admin is logged in. 2. Teacher has submitted the signup form.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**
1	Click on pending registration button.	Pending Registrations displayed.	Pass
2	Click View Button.	Complete Teacher Registration Form opened.	Pass
3	Upload Images.	Images uploaded .	Pass
4	Click Register Button.	Email Verification Link sent to the user.	Pass

TC 03

Test Case ID:	TC–03		
Test Case Title:	Discard Pending Registrations.		
Description:	Discarding pending registration request.		
Pre-conditions:	1. Admin is logged in. 2. Teacher has submitted signup form.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**

1	Click Pending Registration button.	Pending Registrations displayed.	Pass
2	Click Discard Button.	Pending Registration request removed.	Pass

TC 04

Test Case ID:	TC–04		
Test Case Title:	Email Verification		
Description:	Testing Email Verification.		
Pre-conditions:	1. Teacher has submitted signup form. 2. Admin is logged in.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**
1	Check Email.	Verification Email received.	Pass
2	Click on verification URL.	Registration completed.	Pass

TC 05

Test Case ID:	TC–05		
Test Case Title:	User Login		
Description:	Testing User Login process.		
Pre-conditions:	1. User is registered.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**
1	Open Login Page.	Login page is live.	Pass
2	Enter login details and click Login button.	User homepage opens/Admin Dashboard opens.	Pass

TC 06

Test Case ID:	TC–06		
Test Case Title:	Teacher Report		
Description:	Testing Teacher Report.		
Pre-conditions:	1. User is logged in.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**
1	Click Teacher Report Button.	Teacher Report Page opens.	Pass
2	Apply Date Filter.	Result shows for the selected date range.	Pass
3.	Download CSV.	CSV Downloaded.	Pass

TC 07

Test Case ID:	TC–07		
Test Case Title:	Student Report		
Description:	Testing Student Report.		
Pre-conditions:	1. User is logged in.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**
1	Click Student Report Button.	Student Report Page opens.	Pass
2	Apply Date Filter.	Result shows for the selected date range.	Pass
3.	Download CSV.	CSV Downloaded.	Pass

TC 08

Test Case ID:	TC–08		
Test Case Title:	Add Student		
Description:	Testing Add Student.		
Pre-conditions:	1. Admin is logged in.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**
1	Click Add Student.	Add Student page opens.	Pass
2	Enter Student Details & upload images.	Shows validation checks.	Pass
3	Click Register.	Database Updated.	Pass

TC 09

Test Case ID:	TC–09		
Test Case Title:	Add Teacher		
Description:	Testing Add Teacher.		
Pre-conditions:	1. Admin is logged in.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**
1	Click Add Teacher.	Add Teacher page opens.	Pass
2	Enter Teacher Details & upload images.	Shows validation checks.	Pass
3	Click Register.	New teacher registered.	Pass

TC 10

Test Case ID:	TC–10		
Test Case Title:	Teacher Check in/out.		
Description:	Testing Teacher Check in/out.		
Pre-conditions:	1. Admin is logged in.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**
1	Click Mark Check in/out.	Teacher Attendance Page opens.	Pass
2	Click Fingerprint Recognition / Face Recognition.	Opens Fingerprint/Face Recognition Page.	Pass
3	Add Face/ Fingerprint.	Face/Fingerprint Added successfully.	Pass

TC 11

Test Case ID:	TC–11		
Test Case Title:	Student Attendance		
Description:	Testing Student Attendance.		
Pre-conditions:	1. Admin is logged in.		
Step No.	**Action to perform**	**Expected Result(s)**	**Status**
1	Click Student / Mark Attendance.	Mark Attendance Page opens.	Pass
2	Choose Room.	Room Attendance Marked successfully.	Pass

Chapter 7

Conclusion

7. Conclusion

FRAMS is a web-based application that caters to modern attendance needs. It is designed to solve the issues of existing manual systems. We have used face recognition technology to mark the attendance of students and used both Face & Fingerprint Recognition to track check-in/check-out of visiting faculty. This system performs sufficiently well in various poses and variations.

This system enables educational institutes to effectively mark the attendance of students and manage visiting faculty, automates reporting and stores their data in the database. Admins can manage attendance through the modern online interface. It saves the administration from the headache of managing huge paperwork which costs both time and money.

The computational models used in this project were selected after careful consideration, and the positive testing outcomes show that our decisions were sound. In this experimental investigation, the system was put through highly rigorous testing, and the real-world performance was far more accurate. The system demonstrated virtually perfect accuracy and further work needed to be conducted in this area.

7.1 Future Enhancement

In future, we are planning to improve and enhance our sytem by adopting more advanced technologies. We are planning to :-

- Utilizing a system of high processing power can result in better performance.
- Developing a mobile application with interactive UI
- Deploying our system to cloud

APPENDIX

a) USER MANUAL

1. Admin dashboard

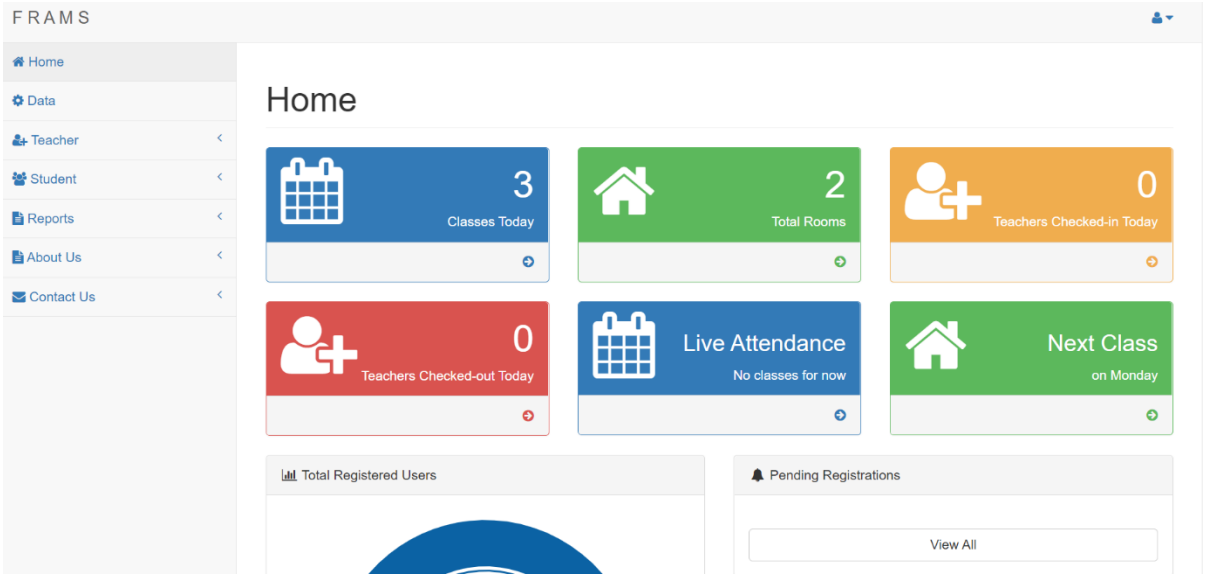

Figure 6.1: Admin Dashboard

2. Admin Side Teacher Registration Page

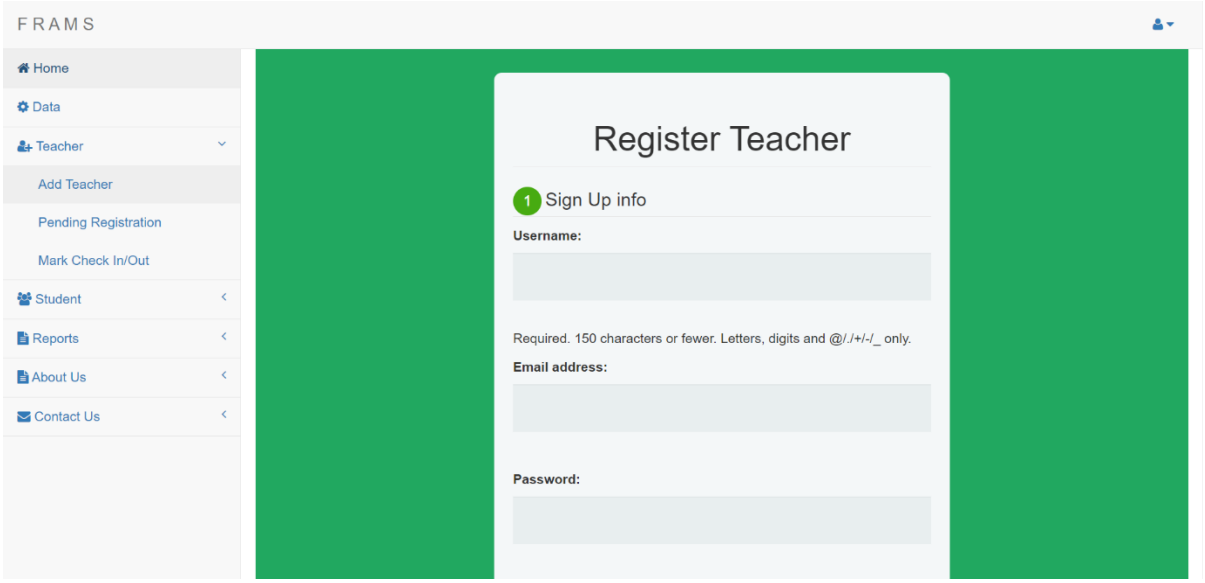

Figure 6.2: Admin Side Teacher Registration Page

3. Pending Registrations Page

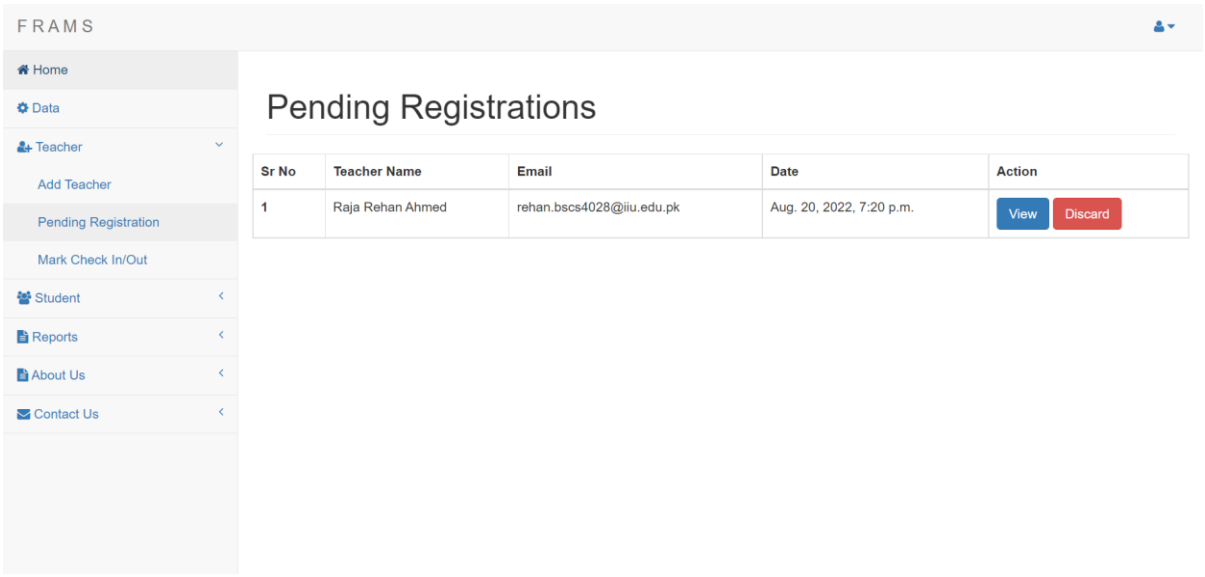

Figure 6.3: Pending Registrations Page

4. Teacher Attendance Page

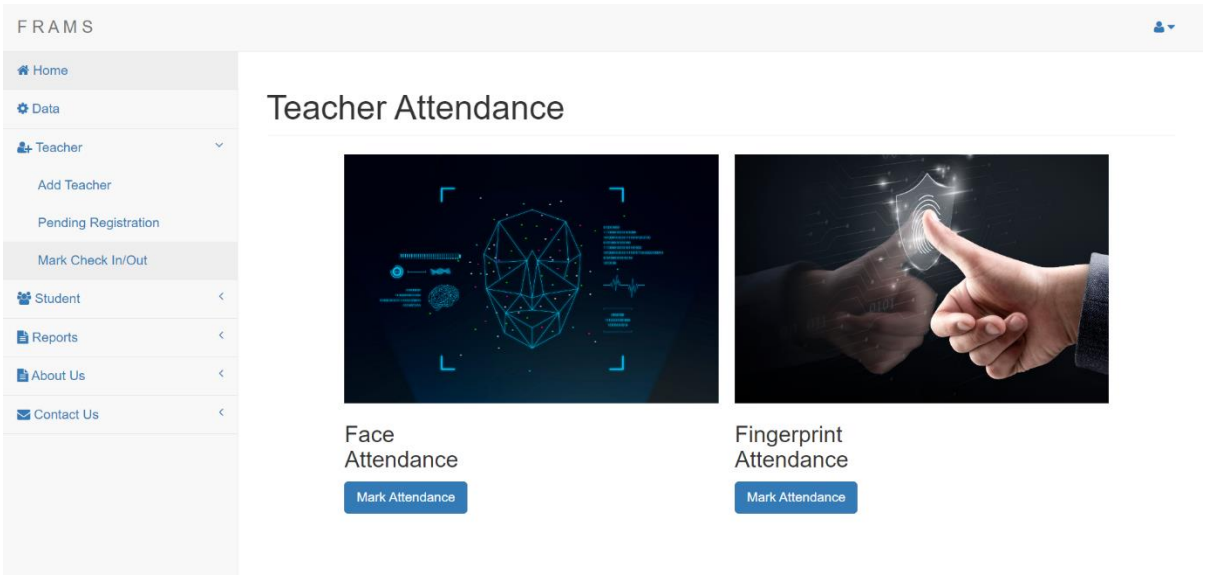

Figure 6.4: Teacher Attendance Page

5. Student Registration Page

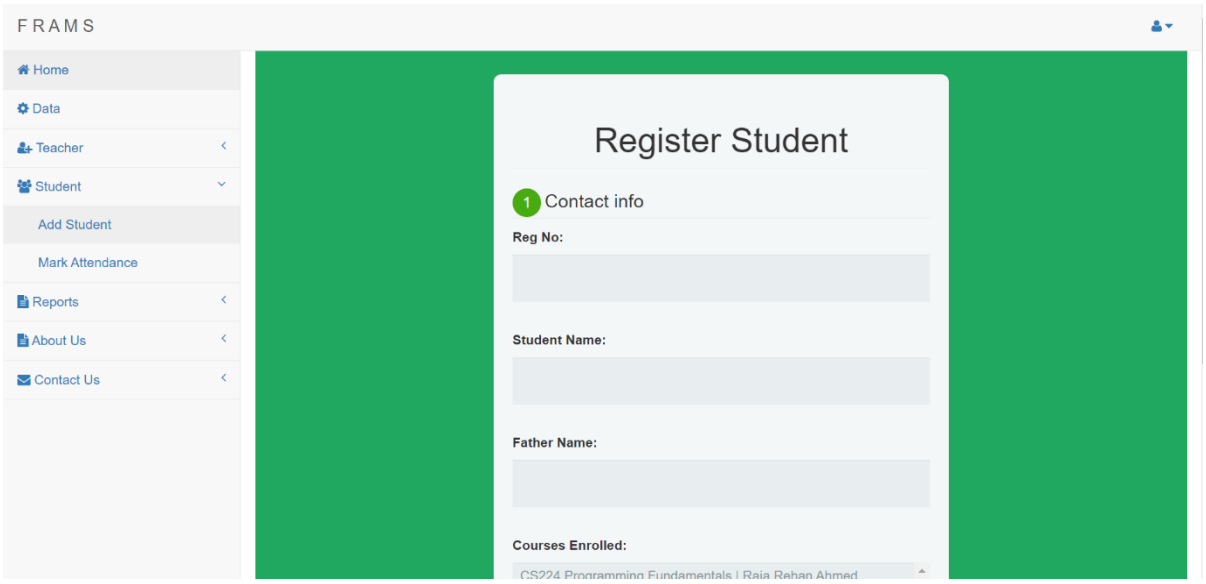

Figure 6.5: Student Registration Page

6. Student Attendance Page

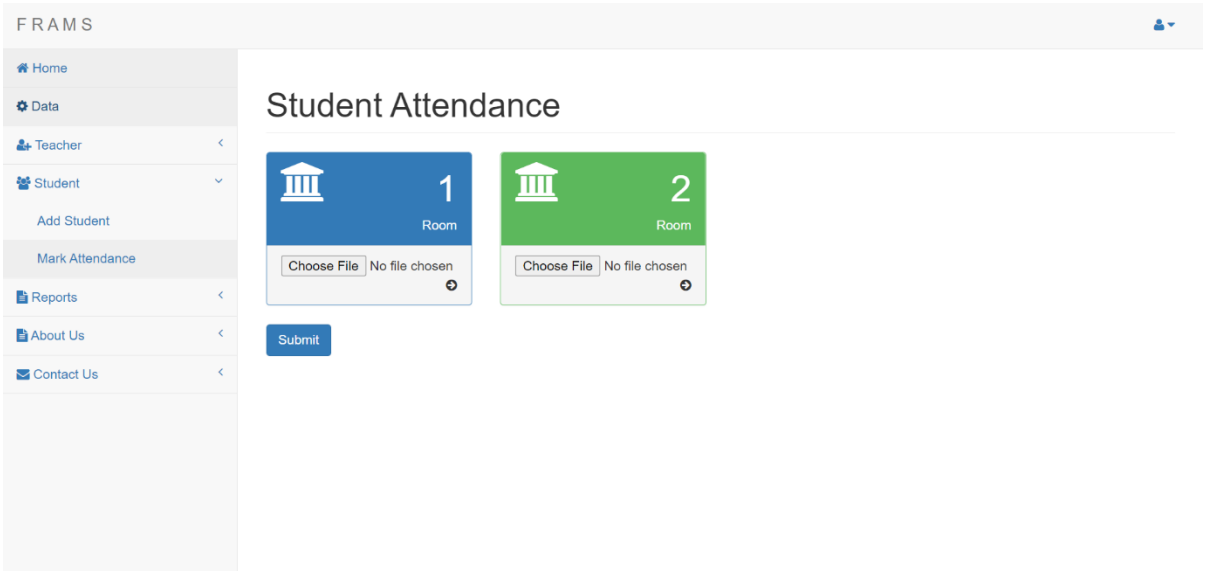

Figure 6.6: Student Registration Page

7. Admin Side Teacher Report Page

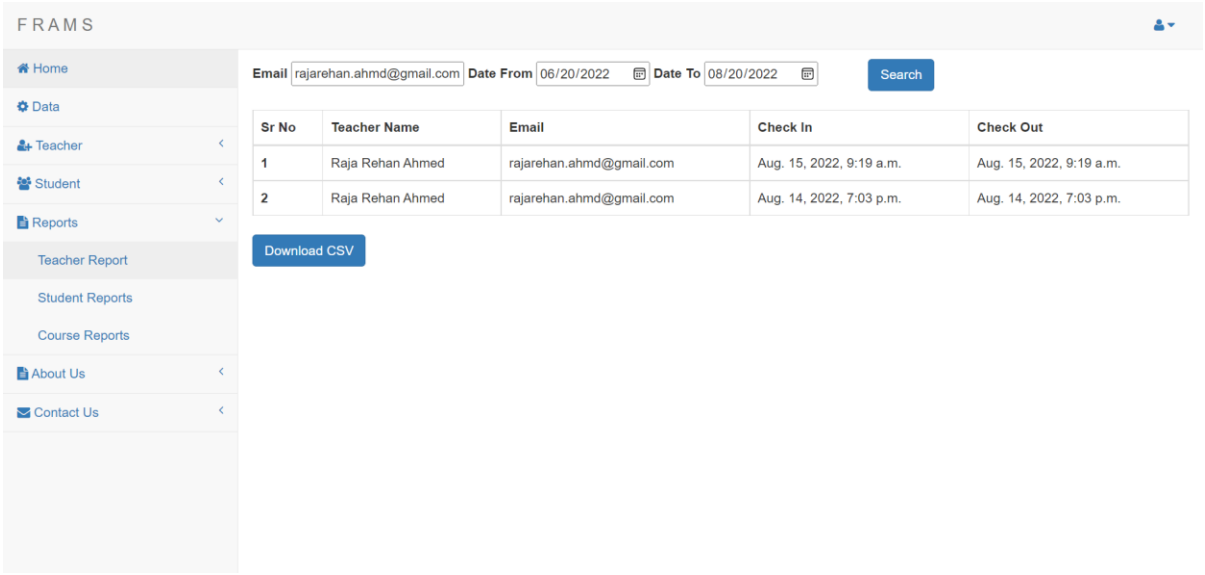

Figure 6.7: Admin Side Teacher Report Page

8. Admin Side Student Report Page

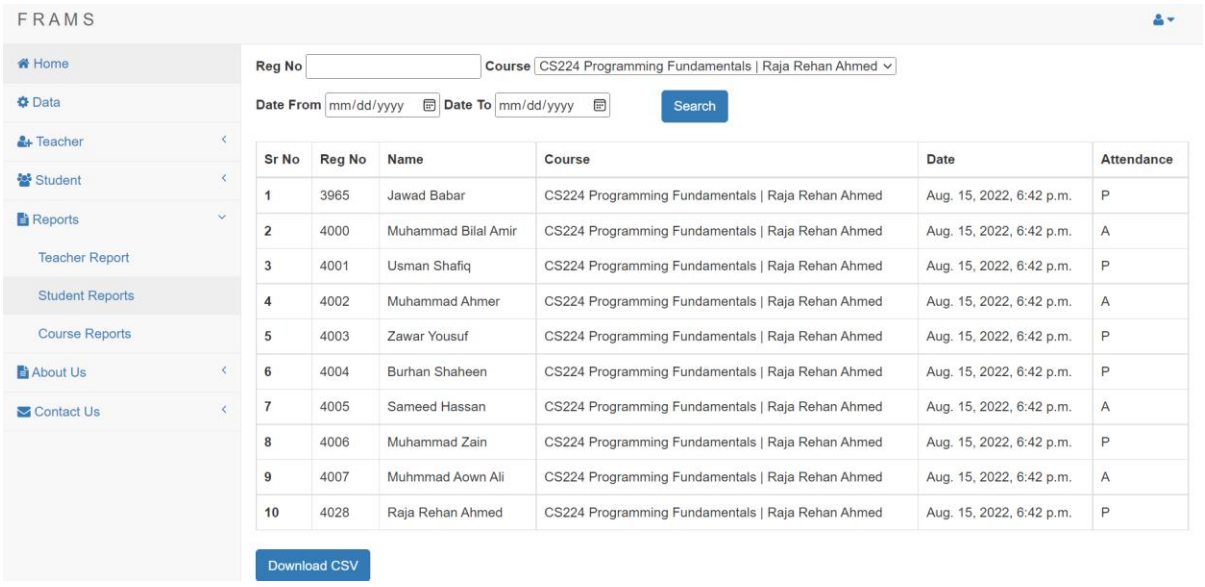

Figure 6.8: Admin Side Student Report Page

9. Admin Side Course Report Page

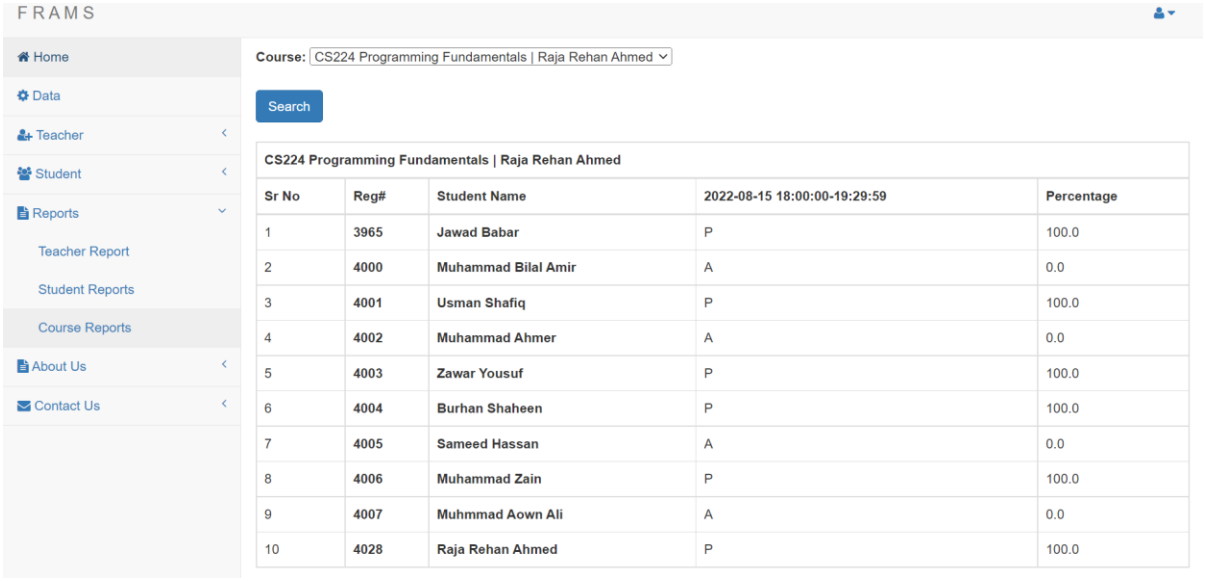

Figure 6.9: Admin Side Course Report Page

10. Teacher Home Page

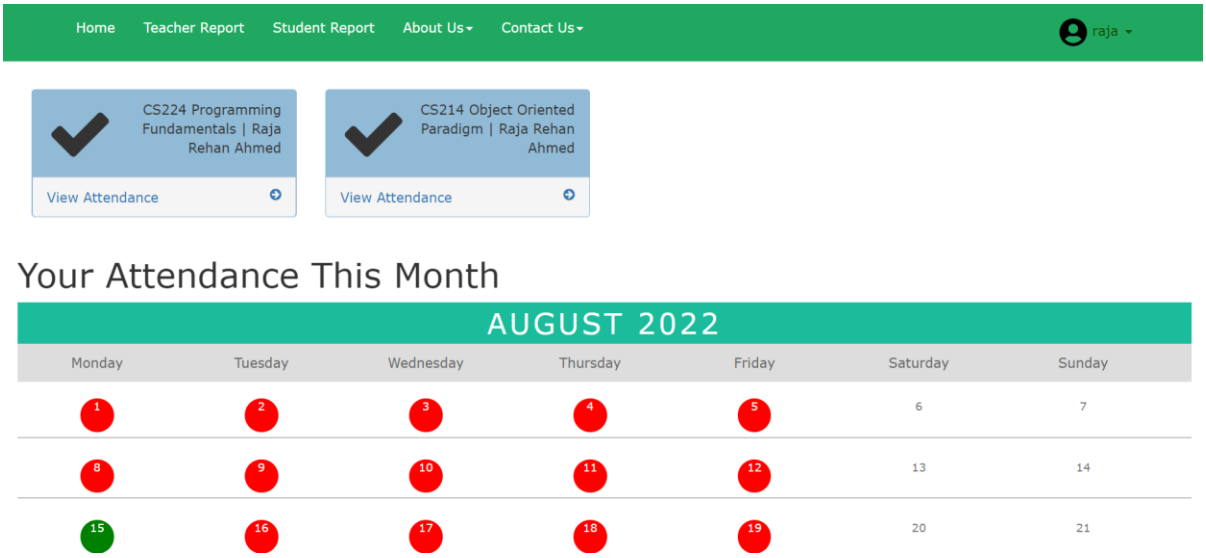

Figure 6.10: Teacher Home Page

11. Teacher Signup Page

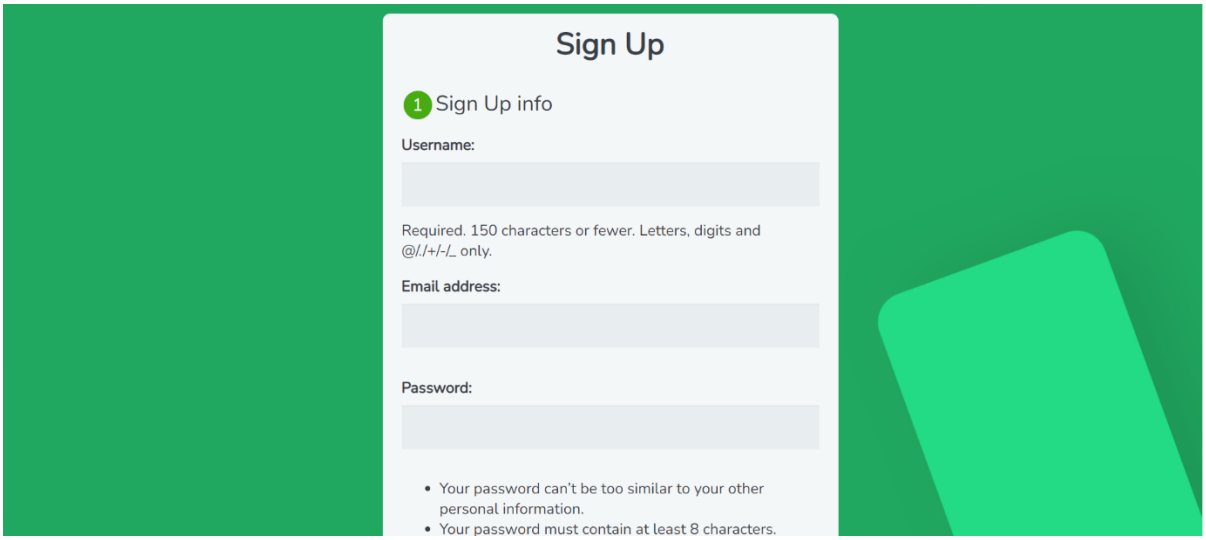

Figure 6.11: Teacher Signup Page

12. Teacher Side Course Report Page

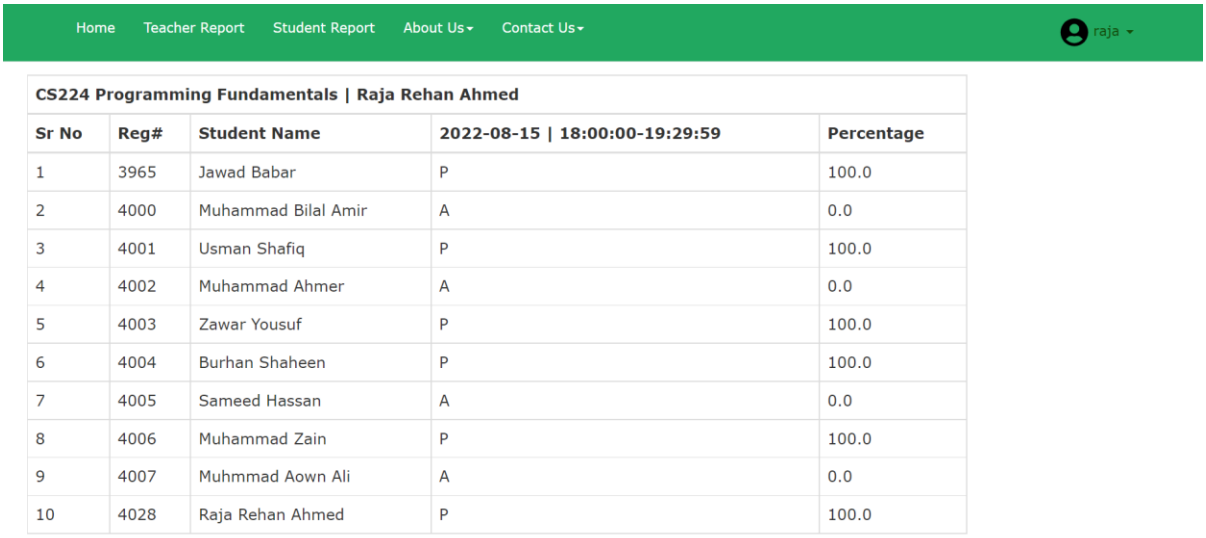

Figure 6.12: Teacher Side Course Report Page

13. Teacher Side Teacher Report Page

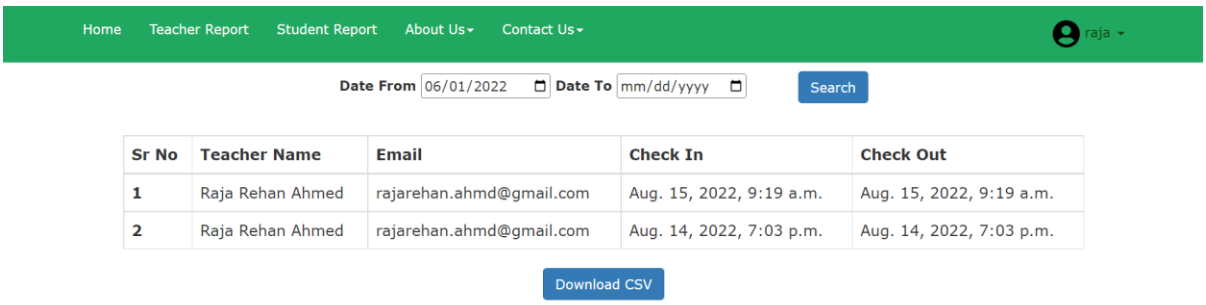

Figure 6.13: Teacher Side Teacher Report Page

14. Teacher Side Student Report Page

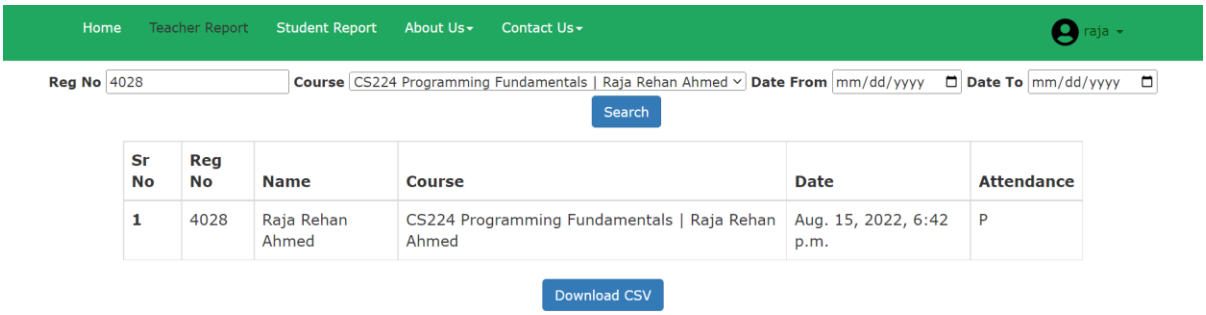

Figure 6.14: Teacher Side Student Report Page

15. Teacher Profile Modal

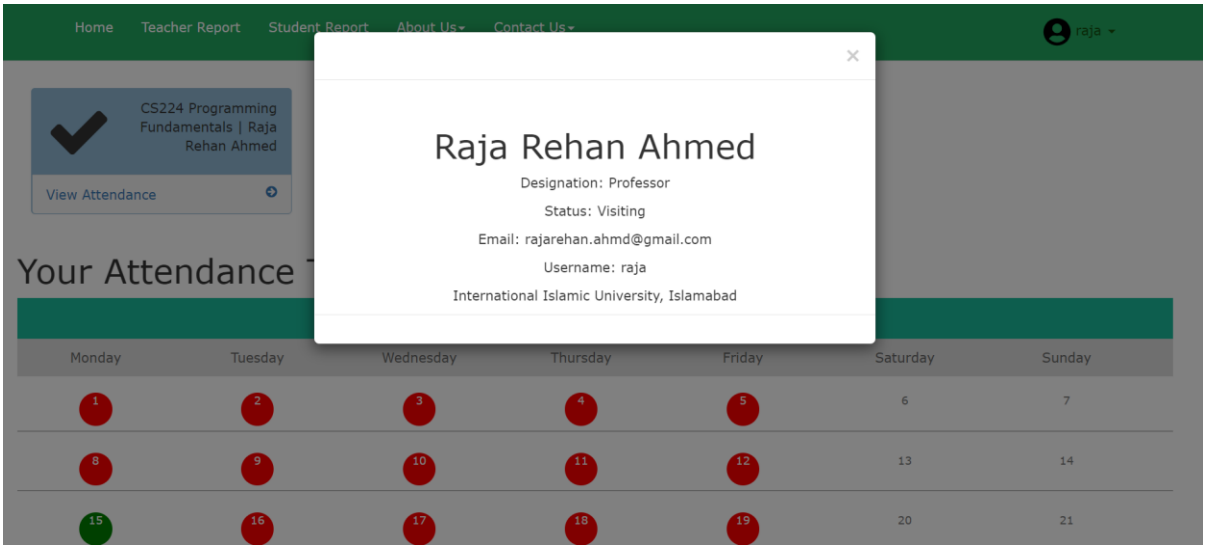

Figure 6.15: Teacher Profile Modal

16. Sign in Page

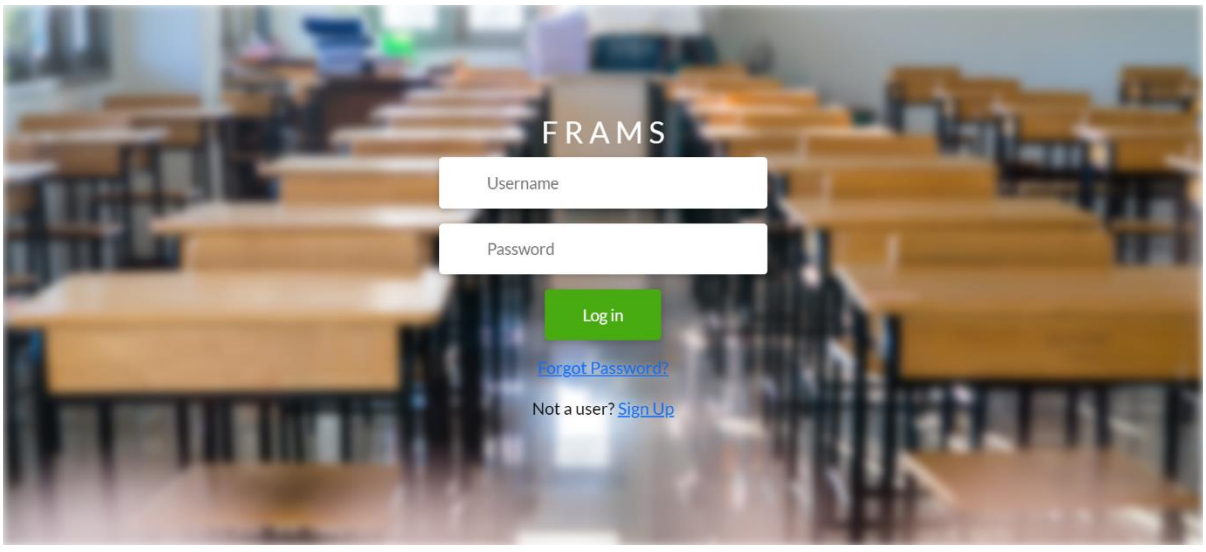

Figure 6.16: Sign In Page

REFERENCES

1. https://medium.com/@ageitgey/machine-learning-is-fun-80ea3ec3c471
2. https://medium.com/@ageitgey/machine-learning-is-fun-part-2-a26a10b68df3
3. https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78
4. https://docs.djangoproject.com/en/4.1/
5. Craig Larman, "Applying UML and Patterns" 3rd Edition, 2004
6. https://opencv.org/
7. https://face-recognition.readthedocs.io/en/latest/face_recognition.html