Tribhuvan University

Faculty of Humanities and Social Sciences

FACE DETECTION ATTENDANCE SYSTEM USING EUCLIDEAN DISTANCE

A PROJECT REPORT

Submitted To

Department of Computer Application

Aadim National College

In partial fulfillment of the requirements for the Bachelors in Computer Application

Submitted By:

Nigam Rai (105902063)

Jestha 2081

6th Semester

Under the supervision of

Er. Prem Subedi

Abstract

One of the main motivations behind this project was to improve how attendance is taken and tracked, especially in environments where scheduling and accuracy are important. Traditional attendance systems are often manual, time-consuming, and prone to errors or even tampering. To overcome these challenges, this project set out to build an automated face detection attendance system that could identify and verify individuals without the need for manual input.

The goal was to create a system that could recognize faces and automatically mark attendance. This would help reduce human involvement, prevent proxy attendance, and allow real-time tracking of attendance records. Additionally, the system aimed to make it easy for both users and administrators to access attendance history and reports through a modern, user-friendly platform.

To build the system, several technologies were used. React Native was chosen for the mobile front end because it offers a smooth and responsive experience on Android devices. The backend was developed using Node.js and Express.js to handle API requests and overall server logic. Python was used for the face recognition component, with the help of machine learning libraries like face-recognition for processing images and detecting facial features. Attendance data was stored in MongoDB, a flexible NoSQL database suited for this kind of application.

In the end, the system worked well. It was able to detect and recognize faces accurately, mark attendance automatically, and keep the records safe. The project met its goals by offering a simple, efficient, and reliable way to track attendance. It made the whole process easier, reduced the need for manual work, and proved to be a useful tool in real-world situations.

Keywords: *Traditional Attendance Systems, Automated Face detection, Proxy Attendance, Real-Time Tracking, Attendance History, React, Node.js, Express.js, API, Python, Face-Recognition, MongoDB, NoSQL, Database*

Acknowledgements

I would like to express my heartfelt gratitude to Tribhuvan University (TU) and Aadim National College for providing me with the opportunity and resources to undertake this project. The support and environment offered by the institution have played a vital role in the successful completion of this work.

I am deeply thankful to my supervisor, Er. Prem Subedi, for his continuous guidance, valuable feedback, and unwavering encouragement throughout the project. His insights and expertise have been instrumental in shaping the direction and outcome of this work.

I would also like to extend my sincere thanks to the Head of Department, Er. Prem Subedi, for his support and for facilitating a productive learning environment.

My sincere appreciation goes to all the IT staff for their cooperation and for providing me with sufficient information and assistance during the course of the project.

Lastly, I am immensely obliged to my friends for their uplifting inspiration, encouraging guidance, and kind support, which helped me remain focused and motivated throughout the completion of this project.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Introduction

A Face Detection Attendance System is a biometric system that uses face detection technology to for easy and simple attendance marking. Unlike the traditional ways that use manual signtaures, or fingerprint scanners, this system marks attendance based on the basis of individual's unique facial features for a fast and contact way of marking attendance.

By using artificial intellingence and computer webcam, the system captures and processes face photographs in real-time and verifies the person. This helps to remove proxy attedance and human error, making attendance management more reliable and faster.

Face detection attendance systems are generally used in offices, colleges, hospitals etc where accurate attendance management is needed.

1.2 Problem Statement

To record attendance, the traditional methods include manual registers, and fingerprint scanning. These methods suffer from various inefficiencies, inaccuracies, and security breaches. In manual systems, attendance recording consumes time, is prone to error, and is susceptible to manipulation, whether in proxy attendance or buddy punching, in which Person A marks Person B's attendance. Similarly, both RFID and fingerprint systems require physical contact on some level and hence raise hygiene concerns and wear-and-tear issues over the years.

Such limitations give birth to a whole series of issues grappling schools, universities, offices, and organizations-ie., disparities in data set, administrative burden, and degradation of security. Furthermore, arranging attendance in a large setup can be a herculean task and hence diminish efficiency and productivity.

To address such problems, the Face Detection Attendance System is the contactless automated attendance system with high accuracy that prevents any human intervention, fraudulent attendance, and security breach. Employing AI-based facial recognition technology for itself keeps the attendance system real-time, readily integrated, and another efficient way of keeping attendance.

1.3 Objectives

- To develop a web interface where individuals can simply walk up to the system, have their face scanned, and their attendance recorded without manual intervention
- To allow users and the admins to view attendance record.

1.4 Scope and Limitation

1.4.1 Scope

- The project will capture and verify to mark attendance without manual intervention.
- The project will store the attendance and allowed to be viewed by user and admin

1.4.2 Limitations

1.5 Development Methodology

For the methodology of the proposed system, CRISP-DM methodology is being used.

CRISP-DM stands for cross-industry process for data mining. The CRISP-DM methodology provides a structured approach to planning a data mining project. It is a robust and well-proven methodology. We do not claim any ownership over it. We did not invent it. We are however evangelists of its powerful practicality, its flexibility and its usefulness when using analytics to solve thorny business issues. It is the golden thread than runs through almost every client engagement. The CRISP-DM model is shown on the right.

This model is an idealised sequence of events. In practice many of the tasks can be performed in a different order and it will often be necessary to backtrack to previous tasks and repeat certain actions. The model does not try to capture all possible routes through the data mining process.

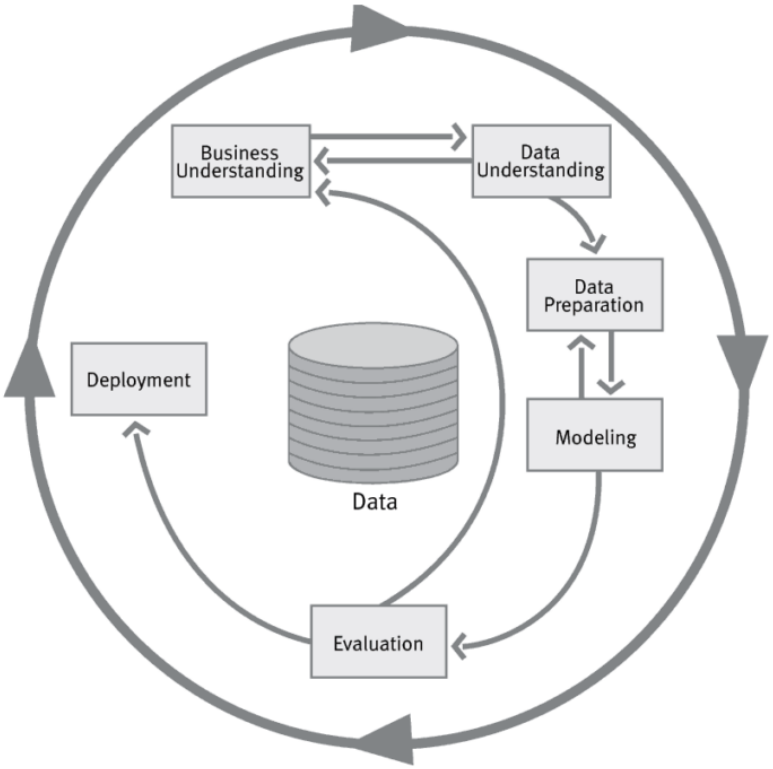


Figure 1.1 CRISP-DM Methodology

1.6 Report Organization

The first part of the report contains the summarized introduction of the whole report. It includes the overview, scope and limitation, problem statement and objectives of this project.

The second chapter includes background study i.e., description of fundamental theories, general concepts and terminology related to the project. It also includes the literature review i.e., review of the similar projects, research and theories done by other researchers.

The third chapter includes the system analysis and design phase in which the report of functional and non-functional requirements of the project is stated using use case and system diagrams. It also includes the feasibility study about the system which explains whether the system development process is affordable and within the knowledge range of the developers. It shows the technical, operational and economic feasibility of the project development phase. The explanation of the designing of the system is also done in this chapter. It includes data modeling and process modeling which is explained by using ER diagram and Data Flow Diagram. The architectural design, database design and the user interface design are also listed in this chapter.

Chapter 2: Background Study and Literature Review

2.1 Background Study

Attendance recording is a fundamental operation in various atmospheres such as schools, universities, corporate offices, and events. Traditionally, this task is accomplished by using manual methods like attendance sheets and punch cards or by semi-automated methods such as using RFID cards and biometric fingerprint scanners.

Such methods, though performing the task to a large extent, have many restrictions:

- Manual systems are prone to instances of proxy attendance, forgery, and human error. It becomes very cumbersome and inefficient to keep large volumes of paper-based attendance records and even to analyze them.
- Biometric systems, such as fingerprint scanners, though offering stronger security, demand physical contact, thus posing health concerns, especially in a crisis situation such as the COVID-19 pandemic. They also fail to work effectively if a user's finger is dirty, wet, or injured.
- Card-based systems such as RFID or ID cards are easy to lose and allow the holder to share or misuse the card; thus, there are reliability and security concerns.

In recent years, the realms of artificial intelligence (AI) and computer vision have opened new possibilities for modern and contactless identity verification systems. Within this spectrum, face detection and recognition have grown popular as being non-invasive, fast, and relatively accurate.

2.2 Literature Review

This part contains the literature review of the previous research done on face detection attendance system.

a. NCheck

NCheck by Neurotechnology is a suite of biometric time and attendance management software. It leverages biometric technologies, such as fingerprint, facial, and iris recognition, to provide accurate and secure employee attendance tracking. It is commonly used in workplaces, schools, and other institutions where precise timekeeping and identity verification are essential.

Strengths:

- Advanced Biometric Accuracy: Reliable fingerprint, face, and iris recognition algorithms.

- Multi-Biometric Support: Flexible options for different environments and user needs.
- Cross-Platform Availability: Compatible with Windows, Android, and iOS.
- Offline Functionality: Attendance tracking without an internet connection.

Weakness:

- Initial Setup Cost: High upfront costs for biometric devices and licensing.
- Hardware Dependency: Requires compatible biometric devices for operation.
- Environmental Limitations: Performance issues in poor lighting or with damaged fingerprints.

b. Lystface

Lystface is a face recognition app that leverages advanced facial recognition technology for various applications. While there isn't a great deal of detailed public information about Lystface specifically, it is typically categorized as a tool for facial recognition-based attendance systems and identity verification. Apps like Lystface are increasingly popular in both personal and professional settings due to the growing demand for secure, automated processes.

Strengths:

- Efficiency: Automates the attendance process, reducing time spent on manual roll calls or sign-ins.
- Accuracy: Facial recognition can significantly reduce errors associated with traditional attendance methods, such as mistaken identity or manual errors.
- Security: Prevents proxy attendance (e.g., having someone else sign in for another person), offering more secure verification of attendance or identity.

Weaknesses:

- Privacy Concerns: The use of facial recognition raises privacy issues. Users might be concerned about how their biometric data is stored, who has access to it, and whether it could be misused.
- Technical Limitations: Facial recognition systems can sometimes struggle with factors like low lighting, angle, or changes in appearance (e.g., facial hair, glasses, or masks). These challenges can reduce the reliability of the app in certain scenarios.
- Data Security: While the app may offer encryption, storing biometric data can still be a security risk. If not properly secured, there’s potential for data breaches or hacking attempts.

Others: FaceIt Systems, Buddy Punch etc

Chapter 3: System Analysis and Design

3.1 System Analysis

This chapter discusses about the proposed model of the system, the methodology used in building the system, tools and techniques, requirement analysis, requirement specification. It also talks about the functional requirements of the system, the system design. The application architecture of the system will also be shown in this chapter, the use case diagram, data design, activity diagrams, dataflow diagram, control flow diagram, entity-relationship diagram (ERD) and user interface design will all be shown in this chapter.

3.1.1 Requirement Analysis

The following section presents the complete set of functional requirements of PARG system.

i. Functional Requirements

Following figure shows the requirements:

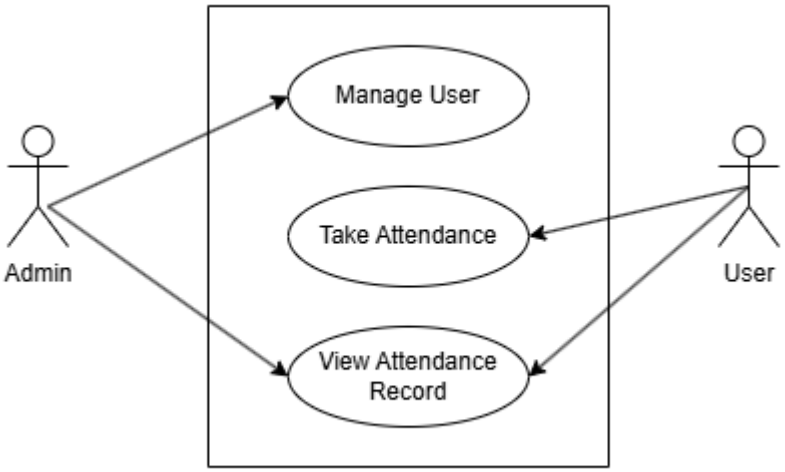


Figure 3.1 Use case Diagram

Table 3.1 Functional Requirements

Req. No.	Description	Type
R-101	The web application will contain user interface.	Functional
R-102	The web application will require internet to operate.	Configuration

R-103	The web applications will allow navigating menus.	Functional
R-104	The web applications will allow admin to manage (add, update, view, delete) users and attendances	Functional
R-105	The web application will allow users to take attendance and view attendance records	Functional

ii. Non-functional Requirements

Some of the contents of non-functional requirements are shown table below.

Table 3.2 Non-functional requirements

Req. No.	Description	Type
NR-101	The web application shall ensure sensitive information is secure	Security
NR-102	The web application will be user friendly	Usability
NR-103	Unauthorized usersvwill not be able to access the system.	Security
NR-104	The web application shall run well on desktop and mobile devices	Configuration

3.1.2 Feasibility Analysis

Some important feasibility studies are mentioned below:

i. Technical Feasibility

It is technically feasible as I already have hardware and software required for development of a software. Also, I have technical knowledge on how the project is made through programming language like JavaScript and python.

ii. Operational Feasibility

It is operationally feasible as I am making this system by removing the threats and weakness of existing non manageable system which is reliable for the users.

iii.Economic Feasibility

The system does not require extra software and hardware. So, there is no recurring cost than just the internet connection.

3.1.3 Data Modeling (ER Diagram)

Since we are using a NoSql database (MongoDB), there will be no ER Diagram

3.2 System Design

3.2.1 Architecture Design

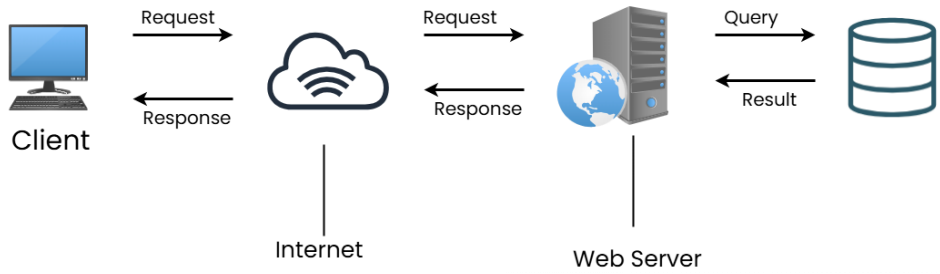


Fig 3.2 Architecture Design

3.2.2 Database Schema Design

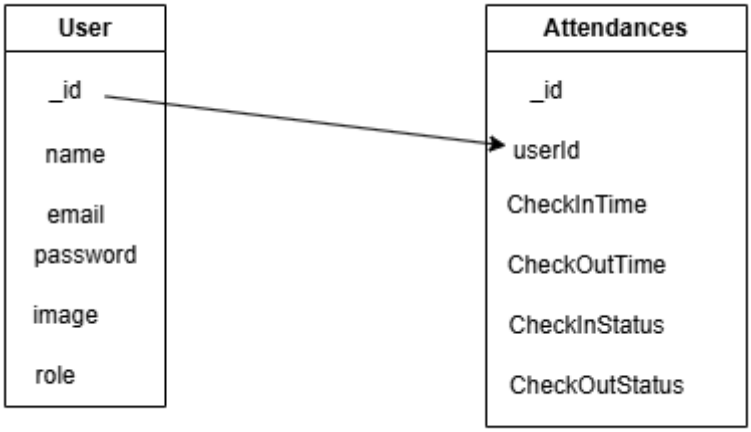
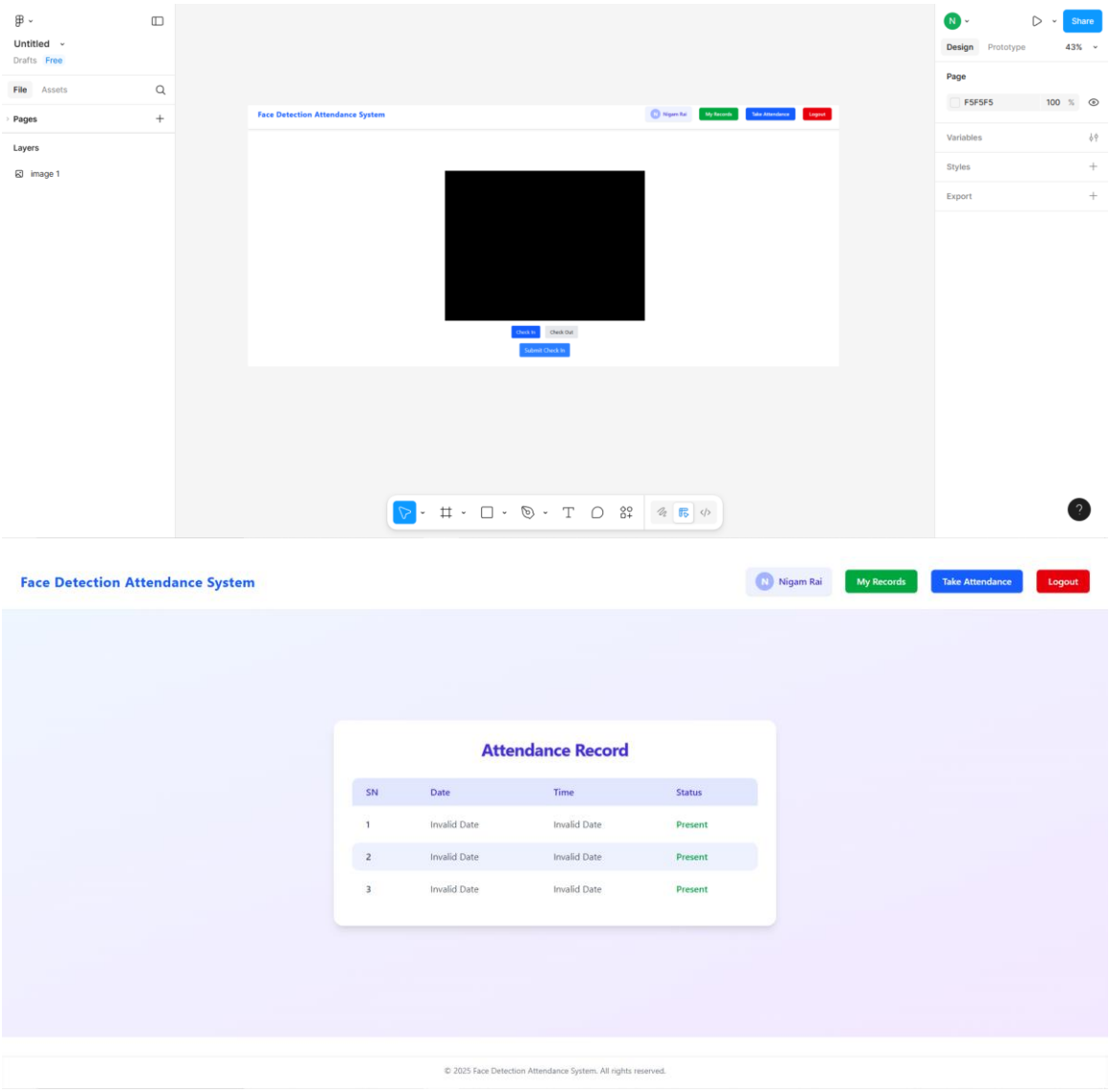


Fig 3.3 Database Schema Diagram

3.2.3 Interface Design

Some of the screenshots of interface design of Face Detection Attendance System is shown below:



3.3 Algorithm Details

In a face detection attendance system, once a face is detected and extracted from an image, the next crucial step is face matching — determining whether the detected face matches a known (enrolled) face in the database. One of the most common techniques for this is Euclidean distance.

How It Works

1. Face Embedding Generation:

First, each face image is passed through a pre-trained face recognition model (e.g., FaceNet, Dlib, DeepFace) to generate a face embedding — a high-dimensional vector (typically 128 or 512 values) that uniquely represents facial features.

Example:

Face A → [0.23, -0.12, 0.98, ..., 0.45]

Face B → [0.21, -0.10, 0.97, ..., 0.44]

2. Calculating Euclidean Distance:

The Euclidean distance between two embedding vectors is computed using the formula:

$$d = \sqrt{(b_1 - a_1)^2 + (b_2 - a_2)^2 + \cdots + (b_n - a_n)^2} = \sqrt{\sum_{i=1}^{n}(b_i - a_i)^2}$$

3. Matching Decision:

A threshold value is defined (e.g., 0.6). If the distance between the two face embeddings is less than or equal to the threshold, it is considered a match (i.e., same person). If it's greater, it's a mismatch.

Example:

- Distance = 0.45 → Match
- Distance = 0.82 → No Match

Why Euclidean Distance?

- It is simple and computationally efficient.
- Works well when embeddings are L2-normalized (unit-length vectors).
- It allows fast comparison across many faces, which is crucial in real-time attendance systems.

Use in Attendance System

In the system:

- The user's face is captured at the time of attendance.
- The system generates an embedding for the live face and compares it to stored embeddings in the database.
- If a match is found (distance ≤ threshold), attendance is marked.

This process ensures accuracy, speed, and automation without requiring manual verification.

Chapter 4: Implementation and Testing

4.1 Implementation

The aim of this chapter is to document the process of development of the main features. It gives a detailed breakdown of the problems encountered and how they were resolved. It also goes through the test plan and test report of the project to ensure all the functionalities are functioning properly. This chapter is where the project is going to be implemented. However, the software and hardware components used in the implementation of this project will be analyzed below.

4.1.1 Tools Used

React

React is an open-source JavaScript library developed by Facebook, used for building user interfaces, especially single-page applications. It allows developers to create reusable UI components, improving code maintainability and performance. React follows a component-based architecture and uses a virtual DOM to optimize rendering, making applications more efficient and responsive. In the context of this project, React (or React Native for mobile apps) was used to build a dynamic, user-friendly interface that ensures smooth interaction and real-time updates for users.

Node.js

Node.js is a runtime environment that allows JavaScript to be executed on the server side. Built on Chrome's V8 engine, it is known for its non-blocking, event-driven architecture, which makes it ideal for scalable and real-time applications. In this project, Node.js with Express.js was used to create the backend server, handling API requests, routing, authentication, and real-time communication. It played a critical role in connecting the frontend interface with the database, ensuring secure and efficient data flow.

MongoDB

MongoDB is a NoSQL, document-oriented database that stores data in flexible, JSON-like documents. Unlike traditional relational databases, MongoDB does not require a fixed schema, making it highly adaptable to changes in application requirements. Its scalability and ease of integration with JavaScript-based technologies make it a popular choice for modern web and mobile applications. In this project, MongoDB was used to store and manage user data, messages, attendance records, and other relevant information in a structured yet flexible format.

Chapter 5: Conclusion and Future Recommendations

5.1 Conclusion

The development of the face detection attendance system successfully demonstrated how modern technologies can automate and enhance traditional attendance methods. By utilizing facial recognition for identity verification, the system minimized the need for manual intervention, reduced the chances of proxy attendance, and improved the overall accuracy and reliability of attendance tracking. The integration of technologies like Python for face recognition, React Native for the user interface, Node.js for backend services, and MongoDB for data storage allowed the creation of a scalable and responsive system. The project effectively met its objectives by providing a solution that is not only efficient but also practical for use in educational institutions and organizations.

5.2 Lesson Learnt/Outcome

While developing the application, I learned a lot about creating a modular system, which made the code easier to manage and test. I also realized how important it is to organize data efficiently, especially when using mongodb to handle databases. Connecting the front-end (React) and back-end (Node JS) using REST APIs helped me understand how to sync data smoothly between them. I also gained experience in handling errors and validating user input, which improved the system's reliability.

5.3 Future Recommendations

To further improve the system, several enhancements can be considered. The addition of liveness detection could help prevent spoofing using photos or videos. Integrating push notifications would alert users of successful or missed attendance in real time. Developing a web-based admin panel would make it easier to monitor records and manage users. Furthermore, incorporating cloud storage can ensure data security and scalability. Expanding compatibility to iOS platforms and integrating with institutional management systems would also improve usability and adoption in diverse environments. Continuous training of the recognition model with more diverse datasets will enhance accuracy across a wider range of users.