

# FACE RECOGNITION ATTENDANCE MANAGEMENT SYSTEM USING FIREBASE ML-KIT WITH ANDROID



By

Hafiz Muhammad Umer

70063633

Basit Shahzad Tatari

70064897

Asif Sultan Bhatti

70066389

A Project Report submitted to the  
DEPARTMENT OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY  
in partial fulfilment of the requirements for the degree of  
BACHELORS OF SCIENCE IN COMPUTER SCIENCE

Faculty of Computer Science & Information Technology  
University of Lahore  
Islamabad  
March, 2020

Copyright © 2020 by UOL Student

All rights reserved. Reproduction in whole or in part in any form requires the prior written permission of Hafiz Muhammad Umer, Basit Shahzad Tatari, and Asif Sultan Bhatti.

## ***DEDICATION***

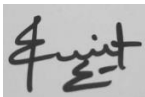
*Challenges in life needs motivation, self-effort as well as advices from elders. My humble dedication is to those who are very dear to me, my parents, my siblings, my teachers, my friends and all those who guide me to the right path, support me, encourage me and motivate me to do better in my life.*

# DECLARATION

I hereby declare that this application neither whole or part of it has been copied out from any kind of source. I further declare that I have developed this application and written this report on the basis of my own effort, under the supervision of my project supervisor. If any part of this system is proved to be copied out from any product found to be the reproduction of someone else, I shall be held responsible.



Hafiz Muhammad Umer  
70063633



Basit Shahzad Tatari  
70064897



Asif Sultan Bhatti  
70066389

March, 2020

# CERTIFICATE OF APPROVAL

It is certified that the project titled “Face Recognition Attendance Management System Using Firebase ML-kit With Android” carried out by Hafiz Muhammad Umer, Reg. No. 70063633, Basit Shahzad Tatari, Reg. No. 70064897, and Asif Sultan Bhatti, Reg. No. 70066389 under the supervision of Mr. Khurram Shahzad and Ms. Itrat Fatima, University of Lahore, Islamabad, is fully adequate, in scope and in quality, as a final year project for the degree of BS of Computer Science.

Supervisor:

-----  
Mr. Khurram Shahzad  
Lecturer  
Dept. of CS & IT  
University of Lahore, Islamabad

Co-Supervisor:

-----  
Ms. Itrat Fatima  
Lecturer  
Dept. of CS & IT  
University of Lahore, Islamabad

Internal Examiner 1:

-----  
Mr. Danial Zafar  
Lecturer  
Dept. of CS & IT  
University of Lahore, Islamabad

Internal Examiner 2:

-----  
Ms. Zubaria Asma  
Lecturer  
Dept. of CS & IT  
University of Lahore, Islamabad

Project Coordinator:

-----  
Mr. Muneeb Muzammil  
Lecturer  
Dept. of CS & IT  
University of Lahore, Islamabad

HOD:

-----  
Dr. Muhammad Yaqoob Wani  
Associate Professor  
Dept. of CS & IT  
University of Lahore, Islamabad

## **ACKNOWLEDGMENT**

I would like to thank my supervisor Mr. Khurram Shahzad and co-supervisor Ms. Itrat Fatima in assisting me and helping me fulfill the requirements of this project and for being cooperative with me and my colleagues Basit Shahzad and Asif Sultan during the course of this project.

# ABSTRACT

Marking and maintaining attendance is one of the biggest problems faced by the institutes. Attendance maintaining is one of the important tasks in any institute because it measures the presence of students as well as helps in the growth of the institute. I and my colleagues thought of this problem and try to contribute to the solution to this problem. We thought to solve this problem by marking the attendance of students by facial recognition using android phones and maintaining this attendance in a secure database.

For solving this we have collected a dataset of faces of our class fellows and train a machine learning model on their faces using Firebase ML kit. We have integrated this model with an android app. The attendance is being stored and maintained in the Firebase Realtime Database. We have created three android apps, an admin app, which will be used by admins of the department to register teachers, students, and subject. A teacher app which will be used by teachers to mark the attendance of students. A student app that will be used by students to check their daily attendance reports.

We have tested our app on the different faces of students and found that our model giving the results of over 80% accuracy. We have learnt so many things during the project, how machine learning algorithms work, how to manage a large-scale project, and much more.

We have tried to put our best effort to contribute to solving the problem of attendance marking and maintenance. We are quite satisfied with our contribution and our project is doing quite an impressive job. We are going to contain this project in the future and try to make it more accurate, scale it to larger platforms, and sell it to different institutes.

# TABLE OF CONTENTS

DECLARATION .....	iv
ACKNOWLEDGMENT .....	vi
ABSTRACT .....	vii
TABLE OF CONTENTS .....	viii
LIST OF FIGURES .....	xiii
LIST OF TABLES.....	xiv
LIST OF ACRONYMS .....	xv
Chapter 1 .....	1
INTRODUCTION .....	1
1.1 Overview .....	1
1.2 Statement of Problem.....	2
1.3 Definitions, Acronyms, and Abbreviations .....	2
1.4 General Description .....	2
1.4.1 Product Perspective.....	2
1.4.2 Product Functions .....	3
1.4.3 User Characteristics .....	3
1.4.4 General Constraints.....	3
1.4.5 Assumptions and Dependencies .....	3
1.5 Purpose of the project .....	3
1.6 Applications of the project.....	4
1.7 Theoretical bases and Organization .....	4
1.8 Summary .....	5
Chapter 2.....	6
LITERATURE REVIEW .....	6
2.1 Related Technologies.....	6
2.1.1 Firebase ML Kit.....	6
2.1.2 Machine Learning .....	7
2.1.3 Android .....	7
2.2 Related Projects .....	8
2.3 Related Studies.....	8
2.4 Project Limitations and Bottlenecks .....	8



2.5 Summary .....	9
Chapter 3.....	10
TOOLS AND TECHNIQUES .....	10
3.1 Hardware used with technical specifications .....	10
3.1.1 Lenovo IdeaPad 100S Specifications.....	10
3.1.2 HP Elitebook 8470p Specifications .....	11
3.1.3 Android Devices: .....	11
3.2 Software and simulation tools used .....	12
3.2.1 Windows 10 .....	12
3.2.2 Android Studio.....	13
3.2.3 Java .....	13
3.2.4 XML.....	14
3.2.5 Firebase ML kit.....	14
3.3 Summary .....	14
Chapter 4.....	15
METHODOLOGIES .....	15
4.1 Design of the investigation/Algorithms/Hardware .....	15
4.1.1 Use Cases .....	16
4.1.1.1 Use case # 1 .....	16
4.1.1.2 Use case # 2 .....	16
4.1.1.3 Use case # 3 .....	17
4.1.1.4 Use case # 4 .....	17
4.1.1.5 Use case # 5 .....	18
4.1.1.6 Use case # 6 .....	18
4.1.1.7 Use case # 7 .....	19
4.2 Analysis Procedures.....	19
4.2.1 Sequence Diagrams.....	20
4.2.1.1 Sequence Diagram # 1 .....	20
4.2.1.2 Sequence Diagram # 2 .....	21
4.2.1.3 Sequence Diagram # 3 .....	22
4.2.2 Flow Charts.....	23
4.2.2.1 Flow Chart # 1 .....	23
4.2.2.2 Flow Chart # 2 .....	24
4.2.2.3 Flow Chart # 3 .....	25

4.3 Implementation Procedures .....	26
4.3.1 Details about Hardware.....	27
4.3.2 Details about Software/algorithms.....	27
4.3.2.1 Android Studio.....	28
4.3.2.2 Algorithm.....	28
4.3.3 Details about control etc .....	28
4.4 Verification of functionalities .....	29
4.5 Summary .....	30
Chapter 5.....	31
SYSTEM TESTING .....	31
5.1 Objective testing .....	31
5.2 Usability Testing.....	32
5.2.1 Natural Mapping .....	32
5.2.2 Task Analysis.....	33
5.3 Software Performance Testing.....	33
5.4 Compatibility Testing .....	33
5.5 Load Testing .....	34
5.6 Security Testing .....	34
5.7 Installation Testing.....	35
5.7.1 Activity Testing .....	35
5.7.2 Operation Testing.....	35
5.8 Test Cases .....	36
5.9 Specific Requirements .....	39
5.9.1 Functionalities.....	39
5.9.2 External Interface Requirements.....	44
5.9.2.1 User Interfaces .....	44
5.9.2.2 Hardware Interfaces .....	44
5.9.3 Software Interfaces .....	44
5.9.4 Communications Interfaces .....	44
5.9.5 Functional Requirements .....	44
5.9.5.1 Functional Requirement 1 .....	45
5.9.5.2 Functional Requirement 2.....	45
5.9.5.4 Functional Requirement 3.....	45
5.9.5.5 Functional Requirement 4.....	45

5.9.5.6 Functional Requirement 5 .....	45
5.9.5.7 Functional Requirement 6 .....	45
5.9.5.8 Functional Requirement 7 .....	46
5.10 Non-Functional Requirements .....	46
5.10.1 Performance .....	46
5.10.2 Reliability .....	46
5.10.3 Availability .....	46
5.10.4 Security .....	46
5.10.5 Maintainability .....	47
5.10.6 Portability .....	47
5.10.7 Efficiency .....	47
5.11 Inverse Requirements .....	47
5.11.1 Interface Requirements .....	47
5.11.2 Modification .....	47
5.11.3 Subject enrollment Limit .....	47
5.11.4 Report Reading .....	48
5.12 Design Constraints .....	48
5.12.1 Limited Storage .....	48
5.12.2 Targeted Android Versions .....	48
5.12.3 Limited users .....	48
5.13 Logical Database Requirements .....	48
5.13.1 Offline Database .....	48
5.13.2 Online Database .....	49
5.13.3 Images Storage .....	49
5.14 Other Requirements .....	49
5.14.1 Scrum Development Model .....	49
Chapter 6 .....	50
RESULTS AND CONCLUSIONS .....	50
6.1 Presentation of the findings .....	50
6.1.1 Hardware Results .....	50
6.1.2 Software results .....	50
6.1.3 Evaluation Details .....	50
6.1.4 Confusion Matrix .....	51
6.2 Discussion of the findings .....	51

6.2.1 Comparison with initial Goal.....	52
6.2.2 Reasoning for shortcomings .....	52
6.3 Limitations .....	52
6.4 Recommendations.....	52
6.5 Summary .....	53
6.6 Conclusion .....	53
Chapter 7 .....	54
FUTURE WORK.....	54

# LIST OF FIGURES

Figure 4.1 Dataset Example .....	15
Figure 4.2 Use Case Diagram .....	19
Figure 4.3 Sequence Diagram 1 .....	20
Figure 4.4 Sequence Diagram 2 .....	21
Figure 4.5 Sequence Diagram 3 .....	22
Figure 4.6 Flow Chart 1 .....	23
Figure 4.7 Flow Chart 2 .....	24
Figure 4.8 Flow Chart 3 .....	25
Figure 4.9 Admin Dashboard & Register Teacher .....	26
Figure 4.10 Register Student & Register Course .....	27
Figure 4.11 Control of Modifications .....	28
Figure 4.12 Firebase Real Time Database .....	29
Figure 5.1 Objective Testing .....	32
Figure 5.2 Database Security Rules .....	34
Figure 6.1 Evaluation Details .....	51
Figure 6.2 Confusion Matrix .....	51

## LIST OF TABLES

Table 3.1: Lenovo IdeaPad 100S Specifications .....	10
Table 3.2: HP Elitebook 8470p Specifications .....	11
Table 4.1: Use Case # 1 .....	16
Table 4.2: Use Case # 2 .....	16
Table 4.3: Use Case # 3 .....	17
Table 4.4: Use Case # 4 .....	17
Table 4.5: Use Case # 5 .....	18
Table 4.6: Use Case # 6 .....	18
Table 4.7: Use Case # 7 .....	19
Figure 4.7 Flow Chart 3 .....	25
Table 5.1 Compatibility Testing .....	34
Table 5.2: Test Case # 1 .....	36
Table 5.3: Test Case # 2 .....	37
Table 5.4: Test Case # 3 .....	37
Table 5.5: Test Case # 4 .....	38
Table 5.6: Test Case # 5 .....	38
Table 5.7: Test Case # 6 .....	39
Table 5.8: User Login .....	40
Table 5.9: Registering of Subject, Teachers, & Students .....	40
Table 5.10: Face Recognition .....	41
Table 5.11: Automatic Attendance Marking .....	42
Table 5.12: Editing of Attendance .....	43

## **LIST OF ACRONYMS**

UOL	University of Lahore
HOD	Head of Department
SDK	Software Development Kit
IDE	Integrated Development Environment
ML	Machine Learning
CNN	Convolution Neural Network
DBMS	Database Management System

# Chapter 1

## INTRODUCTION

Face recognition is an important application it is used in numerous fields. This application is use for maintain and monitoring the attendance records. It is play important role in any organization.

Marking and managing attendance is a difficult task by educational institutes, which is timing-consuming and leads to many student and teacher's related issues. To solve these issues there should be an automatic system of managing and marking attendance. The purpose of this application Faces Recognition Base Attendance Management System.

This system will be used the face of the students and mark the attendance. It will be very helpful in resolving many issues faced by students and teachers.

There are usually two ways to maintain an attendance system.

- Manually Attendance System (MAS)
- Automatically Attendance System (AAS)

The Manually Student's Attendance Management system is a way where a teacher attentive to a particular subject need to call a student roll no and manually mark his attendance. Manually attendance may be a time-consuming way of marking attendance or sometimes the teacher may miss someone or the student may answer the presence of his friends.

There are a lot of problems that come with the manually taking process of attendance in the classroom. To solve these problems, we are going to automate the attendance system. Through the camera to detect face and recognize it and mark the attendance for the student. It helps in stopping the proxy issue, and then attendance of the students will maintain effectively.

### 1.1 Overview

This project is used to helpful for Android users in eliminating Manual Attendance System. In the modern age, organizations are using Face Recognition Attendance



Management System. There are many methods of automatic attendance. One of which is Biometric Attendance system. This system wastes time for the student.

The main goal of our work is to reduce marking and managing attendance issues in any institute. This project is very effective for educational institutes after the completion of this project, teachers can be able to install this app and it will assist to mark and manage the attendance easily.

## 1.2 Statement of Problem

Manual Attendance is a major problem in education institutes, teachers and students both are facing many issues daily. This System will be able to mark the attendance of the students using face recognition. It is used to save time and resolve the issues of the student and teachers related to attendance. This is also very helpful to restrict students to take classes properly and in-time.

## 1.3 Definitions, Acronyms, and Abbreviations

- **Firestore ML Kit:** Firestore Machine Learning kit provides an easy and ready-to-use many learning-based models in mobile (android and iOS), some of these models include: detecting faces, labelling images, recognizing text, and much more.
- **Firestore DBMS:** Firestore Database Management is a Real-time Database that works with instant update and changes in real-time with synchronization of users to see the real-time changes.
- **IDE:** Integrated Development Environment.
- **Android OS:** An Operating System On which our application will run.
- **Android studio:** An IDE to develop Android OS based apps, which we are using in our project to develop apps.

## 1.4 General Description

This section will describe the function and objectives of the system. It also includes the requirements and constraints of the system.

### 1.4.1 Product Perspective

The product will be an android based application implemented with firestore. The android based face recognition attendance management system will provide a simple

method for use to mark and maintain attendance. We are following main features that are included in the system.

- Multiple android API level support: The products will support from android API level 19 to 29 which includes more than 95% device on the Google Play Store.
- User account: It is use to create their account according to their position e.g. Student or Teacher.

#### 1.4.2 Product Functions

We made the system capable of identifying the faces of the students and base on the matched faced will give in the database and marked attendance automatically. The automatic system will maintain the record of attendance for future use it will also calculate the short attendance of the students.

#### 1.4.3 User Characteristics

This system will be used by educational institutes. The effect was that he should have access to a mobile phone and the internet. The administrator will get a training on how to use admin side to register student, teacher, and subject and how to upload images of the face of the students.

#### 1.4.4 General Constraints

This system is built using firebase. Which is very highly flexible for Android. Firebase DBMS will be used to store students, images, classes, and attendance. The system will be only able to perform its functionalities on android OS i.e. Android version should be Between 4.4 to 10.

#### 1.4.5 Assumptions and Dependencies

This software is highly depending on the version of the Android OS and the active internet connection. The android version of device should between 4.4 to 10.

### **1.5 Purpose of the project**

This software will be used in any organization for security purposes. This software is very effective for educational institutes to store faces that are detected and automatically marks attendance. This system is easy and secure for the institutes as well as for the users. It saves time and effort. Many offices use a biometric system and also

at the Government institute to logging at work hours and restricting access to only those people that are authorized. Now in many countries, organizations using a facial recognition system. This is much different from fingerprint sensors, due to this there is very little chance of duplication and fraud. In many aspects recording a fingerprint is a task that requires relatively more effort. The Most interesting implementation of facial recognition in attendance systems is that of event attendance. There are many benefits after creating this system, especially for large events. The Current situation big conferences and music consult still largely rely on tickets, passes, and scanning of barcodes. When the audience numbers hit the late hundreds and even the thousand, are millions Face recognition is very good for all this.

Introducing face recognition cut down these efficiencies since most modern facial recognition systems can perform the task in under two seconds.

This device takes the pictures and attached them to the system and then identifying which student and where is sitting in the classroom. This also habits the procedure that can be used for attendance. And when the teacher comes to class and speaks about attendance and press attendance button and choose the concerned subject to start the attendance procedure. This method is a very safe method you cannot mark anyone's else attendance in his.

## **1.6 Applications of the project**

There are numerous fields where face recognition can work and help in reducing the human effort, some of these applications are:

- Institutes use face recognition to mark attendance.
- Organizations can also use it to mark attendance and record time of their employees.
- Face recognition can be used in public places for crime identification.
- Face recognition can be used in airports to recognize their recurring passengers.

There are many other fields in which face recognition can be use to help humans in their work.

## **1.7 Theoretical bases and Organization**

This report is included in 7 Chapters altogether. The first chapter will explain the introduction, a brief overview, purpose, and the problem statement of the project. The

second chapter will contain the literature review of our project and will explain any kind of technologies or other work that is related to the field base on which we are making our project. The third chapter will give a brief description of the tool and techniques we will be using to complete the required objectives and goals of our project. The fourth chapter will include the methodologies, implementation procedures use and detail about the hardware and software that will be used in our project. The fifth chapter will include brief explanations about the system testing. The sixth chapter explains in the findings, results, and the conclusion of our project. The seventh chapter and the final chapter of this report explain what the future work yet to be done and how we will complete it. The final pages of this document explain the references that guided us throughout the project and the terminologies of the report.

The entire report is based on the project. Face Recognition attendance system Application and covers all the steps taken to develop the app such as the system used and what their hardware specification is. The software tools that were used in the duration of the project, as well as programming languages the project was implementation it.

## **1.8 Summary**

This chapter covers the introduction and a brief overview of our project, the purpose behind it, and why we chose this topic. It also explains in detail the general description of the project what difference it will bring in the future. The Face Recognition Attendance system an Android App that can be utilized by Android smartphone users. The main goal of the project to save the time of users.

## **Chapter 2**

### **LITERATURE REVIEW**

This chapter includes an explanation of the literature review of our project. This chapter of our literature review also covers the brief study of other researches works regarding this field. A literature review is an overall comprehensive summary of previous studies that are related to this particular problem. The literature review of our project will also briefly explain the related work studies, books, articles, and different fields that are pertinent to certain areas of different researches.

Before starting our project, we have studied the different research papers regarding our project to understand how the Firebase machine learning kit actually works. To review these articles is to know about the basic idea of how face detection and machine learning work to recognize the face as well as which type of other resources that need to design and overall development of our Android mobile application. The main thing about our project is we are designing an android based application which is a new fact in our project. After completing the research study, we started to collect data related to our project.

#### **2.1 Related Technologies**

There are numerous technologies that are very nearly related to our project where a lot of work already be done, but we are developing an android based mobile application. There are some main technologies which are uses in our project. These include Firebase ML Kit, Firebase DBMS, IDE, Android OS, Android Studio. The Convolution Neural Network CNN also use to detect multiple faces at the same time. These are such techniques that are playing a vital role in face detection and face recognition.

##### **2.1.1 Firebase ML Kit**

Firebase Machine Learning Kit provides an easy and ready-to-use many machine learning based models for mobile in order to recognize the text, faces detection, identifying to landmarks, scanning the bar-codes, labeling images, and as well as identifying the language of the text that is used in the learning process of data. The Firebase ML-Kit make able to develop more powerful apps that are used in android

devices. The Firebase ML-Kit is able to process data more quickly even when there is no access to the network connection. This thing of ML-Kit makes sure it's a high level of accuracy and precision while detection and recognizing the faces. It also uses a Tensor flow model to make sure more powerful and accurate results while processing is being processed. By using the ML-Kit model in our project data, it detects the images and then recognizes the faces with database faces on the base of features that we are using in our project.

### 2.1.2 Machine Learning

Machine learning is scientific knowledge that uses different types of algorithms and some models that are based on statistical to perform specific functions within the computer system. It is considered as the branch of Artificial intelligence. Machine learning depends on the presently available data, which is in the form of pictures, images, sounds, and texts on which machine learning learns from the environment of the available data. For the working of facial recognition also need to be trained data using the Machine learning algorithms. The DeepFace algorithm is used for facial recognition that depends on a deep convolutional neural network. It was created by the Facebook research team in 2014. The main purpose of this identifies the human faces from the digital images with an accuracy of 97%. The main advantage of Deep learning is this it can be trained on a large amount of data to detects the multiple faces from the given data.

### 2.1.3 Android

Android is such an operating system that is used in mobile phones. It is open-source software and an altered version of the Linux kernel that is specially created for touchscreen mobile devices like smartphones and tablets. It is a very powerful mobile phone OS because it can support a huge number of applications in smart devices. An android plays a vital role in the solution of daily life problems in an efficient way. It also plays a key role in the growth of the business. The development of android supports the complete Java language. The 1.0 is the first android version of was released in 2008, but currently, android 10 is the latest and stable version of the Android development kit that was launched on September 3, 2019. Android uses multiple libraries to perform

specific functions such as audio, video recording, etc. The SQLite database used to store and share android applications.

## **2.2 Related Projects**

There are some projects that are related to our projects such as face detection, biometric verification and other types of facial recognition projects, but there is no specific project in the android application. Therefore, we are doing our project by using firebase ML-Kit with the android application to solve identification problems very efficiently.

## **2.3 Related Studies**

There are numerous research studies have been done since the last decade in the field of face recognition. These studies include artificial intelligence machine learning as well as deep learning that is closely related to our project. Many other studies are interacting within our projects such as face detection, face recognition, image processing, and feature extraction. After studying different studies on face recognition, almost all refer that face recognition is the study of picking up facial features to identify the face. The idea which is used in biometric is also related to our project. Firebase ML-Kit and android technologies also play a vital role in face detection and face recognition. There is no android base specific application that has been developed in the past to solve the identification issues in respective face recognition. Therefore, our main focus is that we develop a powerful android-based application in order to solve identification problems.

## **2.4 Project Limitations and Bottlenecks**

This system is provisioned to be built on the Firebase for Android which is highly flexible. Firebase DBMS will be used in regards to storing the student's images, classes, and attendance. The system will only be able to perform its functionalities on android OS portrait mode. There are the following assumptions and dependencies of our system. This software highly depends on the version of the Android OS i.e. Android version should be Between 4.4 to 10. The device should be enabled with the Internet and should have a high-resolution camera to ensure better results. The system has also needed to have access to the internet. Initially, the system will be used for educational institutes. It means that the user should have the basic knowledge of operating the smartphone. Our system has the ability to support Multiple android API levels its means that our

system will support from android API levels 19 to 29. This means that almost more than 95% of devices on the Google Play Store on the API levels 19 to 29. The system also limits the users to get an account from an admin before using the system will be according to their position such as Student or Teacher.

## **2.5 Summary**

In the above chapter, we have briefly discussed the overview of our project. Before starting the project, we discussed the research work that we did and what type of articles we read which are associated with face recognition. We have also discussed the related technologies and related projects that are done before our project. To start our project, we study various research papers about Machine Learning and face recognition to analyze what is actually be doing. There are several technologies that are nearly associated with the work done regarding our project. There some technologies that are being used in our project including Firebase ML-Kit, Android, and Machine Learning. These technologies played a vital role in the success of Machine learning and face recognition in our project.



## Chapter 3

### TOOLS AND TECHNIQUES

We used several software applications to begin the commencement of our work. We did our practical work on the android operating system.

#### 3.1 Hardware used with technical specifications

We used two different laptops as hardware technology during our project, and HP Elitebook for practical programming and design interface of our project whereas Lenovo notebook was preferred for data collection and compilation. Since our app was designed in Android studio, which is very heavy software, a gaming laptop was preferred as it has very high and heavy hardware specifications because much Android application required lots of space as well as memory to run successfully. We did most of our work on Windows 10 Education operating systems. The two main software we used was Android Emulator and Android Studio. PC equipment details are critical representations of the PC's hardware and software. Processor speed, model, and maker. The normal speed of the processor is GHz. RAM is ordinarily shown in (GB). If the computer has more RAM, it can do it more at a time. A Hard circle is also called ROM. This is regularly shown in GB. Minimum System Requirement is the specs your pc needs just to play the game on full resolution and get 60fps. When developing a project, whether it is an app or website, the developer should always keep in mind the hardware specifications of his system for these have a great role play for the execution as well as the proper running of the software. As there are many heavy software tools in today's age of computer technology that may run very slowly on the systems with the minimum specification or for the worst-case scenario, may not run at all, and this can cause some serious problem for the professional developer any kind of software development.

##### 3.1.1 Lenovo IdeaPad 100S Specifications

**Table 3.1: Lenovo IdeaPad 100S Specifications**

Type	Notebook
SSD	No

HDD	512 GB
RAM	8GB DDR3L 1333 MHz
Graphics Card	Intel® HD Graphics
GPU	2 GB
Processor	Intel® Atom™ Z3735F Processor
CPU	Intel Pentium N3710

### 3.1.2 HP Elitebook 8470p Specifications

**Table 3.2: HP Elitebook 8470p Specifications**

Type	Elitebook
SSD	None
HDD	320 GB
RAM	8 GB
Graphics Card	Intel® HD Graphics 4000
GPU	None
Processor	Intel® Core™ i5-3340M
CPU	2.70 GHz

Table 3.1 and 3.2 explain detail the two systems that were used. They explain the amount of random-access memory (RAM) used in both laptops, what processors both systems have, whether there is a graphic card or not, the GPU and CPU, as well as the amount of storage.

### 3.1.3 Android Devices:

We have used android devices to test our results and accuracy of the model. The android device which are used to test the model on real time has the android operating system

version 8.1.0. The tests results are enclosed in the chapter 6. Another android device which is an android emulator use to test all the functionalities of the project, the emulator has android operating system version 7.

### **3.2 Software and simulation tools used**

A Software is a program that assists in the administration, plan, coding, testing, assessment or services of the other projects. Globally available tools are categorized in the size and intricacy from the simple software for beginner to intermediate level programmers and end-users to extremely high-end and complex software that can support many software projects simultaneously and are mainly used in major businesses or organization. Online programming support software-These software allow developers, programmers, and clients to rapidly right and alter application projects and test program results. Test information generators-These devices investigate a program and product documents of information expected to test the rationale of the program. We did the work regarding our project across multiple operating systems and software applications such as windows 10 Ubuntu OS. The software we used is Firebase, firebase ML Kit, and Android Studio. The languages we used in our projects are Java and XML.

#### **3.2.1 Windows 10**

Windows 10 is a Microsoft based operating system mostly used in PCs. It belongs to the Windows NT family of operating systems. It is a closed source which means that is must be licensed before it is fully installed and used on any PC or laptop. It was developed using three different programming languages, C, C++, and C# and it is of hybrid kernel type. Its default user interface is Window shell. It uses Windows API and the Net framework. The features of this operating system are mentioned below.

- Hardware Independence
- Task Management
- File Management
- Disk Management
- Network Capability
- Hardware Adaptability
- Windows Defender Anti-virus
- Stock software (Ms Office)

- Personal AI assistant
- Good for gamers

Our Window operating system has the following specification:

- Edition: Windows 10 Education
- Version: 1903
- Type: 64-bit
- OS Build 18362.418

### 3.2.2 Android Studio

Android Studio is a software tool developed by Google Inc. for its Android operating system. It was developed on Jet Brain's IntelliJ IDEA software. Its specification is to develop full fledged Android applications. It is free to use software with no license and hence no payment is required. It can be used on window, Linux and even MacOS. It was written in three different programming languages, Java, Kotlin, and C++. It requires a minimum 5 GB for installation and 8 GB RAM to run smoothly. It consists of the following specification.

- Android specific refactoring and quick fixes
- Can create apps for all devices
- Can create apps for all devices
- Building Android Watches applications
- APK support
- Visual layout Editor

### 3.2.3 Java

Java is a class-based, object-oriented programming language structured with scarcely any execution condition as could be. Java application code can run on java virtual machine. Pc programs are written in Java language and it is moveable in light of fact. Java is generally utilized for PCs workstation, gaming support, vehicle route frameworks, restorative observing gadgets, leaving meters, lottery terminal, and cell phones. It is likewise utilized for the server farms to store and move Web-based information. Five initial goals in the information of the language are as follows:

- Easy operate

- Robust and portable
- Architecture, threaded, and spirited
- Excellent performance
- Work well with other languages
- Commonly used in Android app development
- Can be exported to other software tools like Android studio.

### 3.2.4 XML

XML is a software and hardware independent tool that transfer and transporting data. XML stands for extensible Mark-up language. The XML stores the data in plain text format. It provides hardware and software independent way sharing and store data. XML easily upgrade to a new operating system, application without losing data.

- Simplicity,
- Extensible
- Self-description
- Upgrade Easily
- Openness

### 3.2.5 Firebase ML kit

Firebase Machine learning ready-to-use APIs for common mobiles use. It is use for labelling images, identify landmark, and detecting face. Firebase is working for the backend service. Firebase send data and at the same time, that data gets updated. When a customer tells you that change all data, that time all connected clients receive the update data.

## 3.3 Summary

This chapter discussed the tools that were used in the project. The hardware used includes two sets of the systems, HP Elitebook. The OS involved in the project were Windows OS, a closed-source operating developed by Microsoft, written in C, C++, and C# and belong to the Windows NT family of operating systems while the others operating system that was involved is Android OS, an open-source software and an altered version of the Linux kernel that is specially created for smartphones. And software application use for the project were Android studio.

## Chapter 4

### METHODOLOGIES

In this chapter, we will discuss the methodology that we were followed in our project. Several types of methods can be used to achieve better and expected results that we were wanted for our project. To ensure the desired results we were first starting to collect data related to our project and then after collecting data, we separated it into two parts a training dataset and a testing dataset. Once data was collected and then it was loaded. We have trained a Firebase ML-Kit model on the collected dataset and then integrated this model into our android project to achieve our goals.

#### 4.1 Design of the investigation/Algorithms/Hardware

Data collection was the first step of our project. The Data collection was in the form of the student's face images. In the collection phase, all relevant data was collected, then the collected data were separated based on students' ids used as labels. when the data was loaded, then the next phase was to compiled using Firebase ML-Kit in order to obtain the desired results from our project. And at the end, we have integrated this model into the Android-based application that was developed in Android studio by using Java Programming Language.



Figure 4.1 Dataset Example

### 4.1.1 Use Cases

Use cases are used to represent a list of actions and steps that define the interactions between user and the system. Following are the tables for use cases and a use case diagram.

#### 4.1.1.1 Use case # 1

**Table 4.1: Use Case # 1**

<b>ID</b>	1
<b>Description</b>	Registration of Subjects
<b>Actors</b>	Admin
<b>Preconditions</b>	<ol style="list-style-type: none"><li>1. Admin should be logged in to register a new subject.</li><li>2. There should be a minimum of one registered teacher and student before registering the subject.</li></ol>
<b>Basic Steps</b>	<ol style="list-style-type: none"><li>1. Enter the subject name.</li><li>2. Enter the subject id.</li><li>3. Assign subject to a teacher from the list of registered teachers.</li><li>4. Enter semester of the subject.</li><li>5. Enroll students into the subject.</li><li>6. Click done.</li></ol>
<b>Exception</b>	If the subject with the same id already exists, the app will not allow registering the new subject with the same id.

#### 4.1.1.2 Use case # 2

**Table 4.2: Use Case # 2**

<b>ID</b>	2
<b>Description</b>	Registration of Teacher
<b>Actors</b>	Admin
<b>Preconditions</b>	<ol style="list-style-type: none"><li>1. Admin should be logged in to register a new subject.</li></ol>

<b>Basic Steps</b>	<ol style="list-style-type: none"> <li>1. Enter the Name of the teacher.</li> <li>2. Enter the id of the teacher.</li> <li>3. Enter Department of the teacher.</li> <li>4. Enter the education &amp; area of expert of the teacher.</li> <li>5. Click done.</li> </ol>
<b>Exception</b>	If the teacher with the same id already exists, the app will not allow registering the new teacher with the same id.

#### 4.1.1.3 Use case # 3

**Table 4.3: Use Case # 3**

<b>ID</b>	3
<b>Description</b>	Registration of Students
<b>Actors</b>	Admin
<b>Preconditions</b>	1. Admin should be logged in to register a new subject.
<b>Basic Steps</b>	<ol style="list-style-type: none"> <li>1. Enter the Name of the student.</li> <li>2. Enter the id of the student.</li> <li>3. Enter Department &amp; semester of the students.</li> <li>4. Upload 3 pictures of the student (left, front, and right side).</li> <li>5. Click done.</li> </ol>
<b>Exception</b>	If the student with the same id already exists, the app will not allow registering the new student with the same id.

#### 4.1.1.4 Use case # 4

**Table 4.4: Use Case # 4**

<b>ID</b>	4
<b>Description</b>	Modification
<b>Actors</b>	Admin
<b>Preconditions</b>	Admin should be logged in to modify.
<b>Basic Steps</b>	<ol style="list-style-type: none"> <li>1. Choose any subject, teacher or student to modify.</li> <li>2. Perform modifications.</li> </ol>



	3. Click done.
<b>Exception</b>	<ul style="list-style-type: none"> <li>If any id got modified then the entered id should be unique otherwise app will not update the changes.</li> </ul>

#### 4.1.1.5 Use case # 5

**Table 4.5: Use Case # 5**

<b>ID</b>	5
<b>Description</b>	Mark Attendance
<b>Actors</b>	Teacher
<b>Preconditions</b>	The teacher should be logged in to the app.
<b>Basic Steps</b>	<ol style="list-style-type: none"> <li>Choose a subject to mark attendance.</li> <li>Click the picture of the student one by one or place the camera at the entrance.</li> <li>Click done to make the report of the attendance marked.</li> </ol>
<b>Exception</b>	<ul style="list-style-type: none"> <li>Low light and low-resolution camera can affect the attendance marking.</li> </ul>

#### 4.1.1.6 Use case # 6

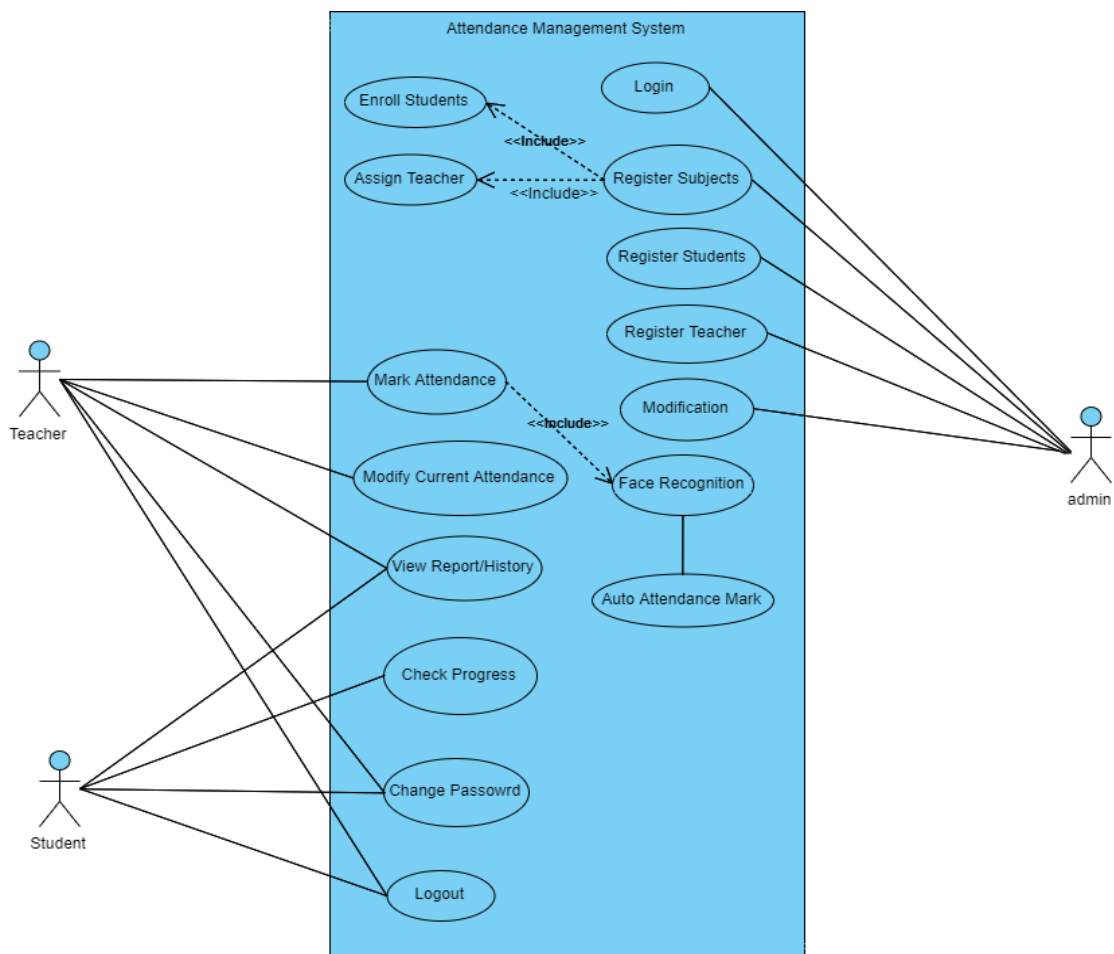
**Table 4.6: Use Case # 6**

<b>ID</b>	6
<b>Description</b>	Modify Current Attendance
<b>Actors</b>	Teacher
<b>Preconditions</b>	<ol style="list-style-type: none"> <li>The teacher should be logged in to the app.</li> <li>The teacher should have marked the attendance of a subject.</li> </ol>
<b>Basic Steps</b>	<ol style="list-style-type: none"> <li>Choose presence of a student (Absent, Present, or Leave).</li> <li>Click save.</li> </ol>
<b>Postconditions</b>	Attendance cannot be modified after it get save.

#### 4.1.1.7 Use case # 7

**Table 4.7: Use Case # 7**

<b>ID</b>	7
<b>Description</b>	Change Password
<b>Actors</b>	Admin, Teacher, & Student
<b>Preconditions</b>	1. The actor should have logged into its account.
<b>Basic Steps</b>	<ol style="list-style-type: none"> <li>1. Go to profile setting.</li> <li>2. Click change password.</li> <li>3. Enter current password, new password, &amp; confirm new password.</li> <li>4. Click done.</li> </ol>
<b>Postconditions</b>	The actor has to login again after changing the password.



**Figure 4.2 Use Case Diagram**

## 4.2 Analysis Procedures

We have analyzed each requirement and evaluated collected dataset based on the requirements. We cropped students' images to only to the face of the student. Then we have analyzed the requirements and thought of how the flow of the project should work. We have taken a small survey to verify that the flow will work possibly or not. Then we again analyzed the flow and change it according to the needs.

### 4.2.1 Sequence Diagrams

Sequence diagrams use to show the sequence of a user to perform his desired operations. User has to follow a sequence of steps in order to achieve his desired task. Following diagrams show sequences for each type of user.

#### 4.2.1.1 Sequence Diagram # 1

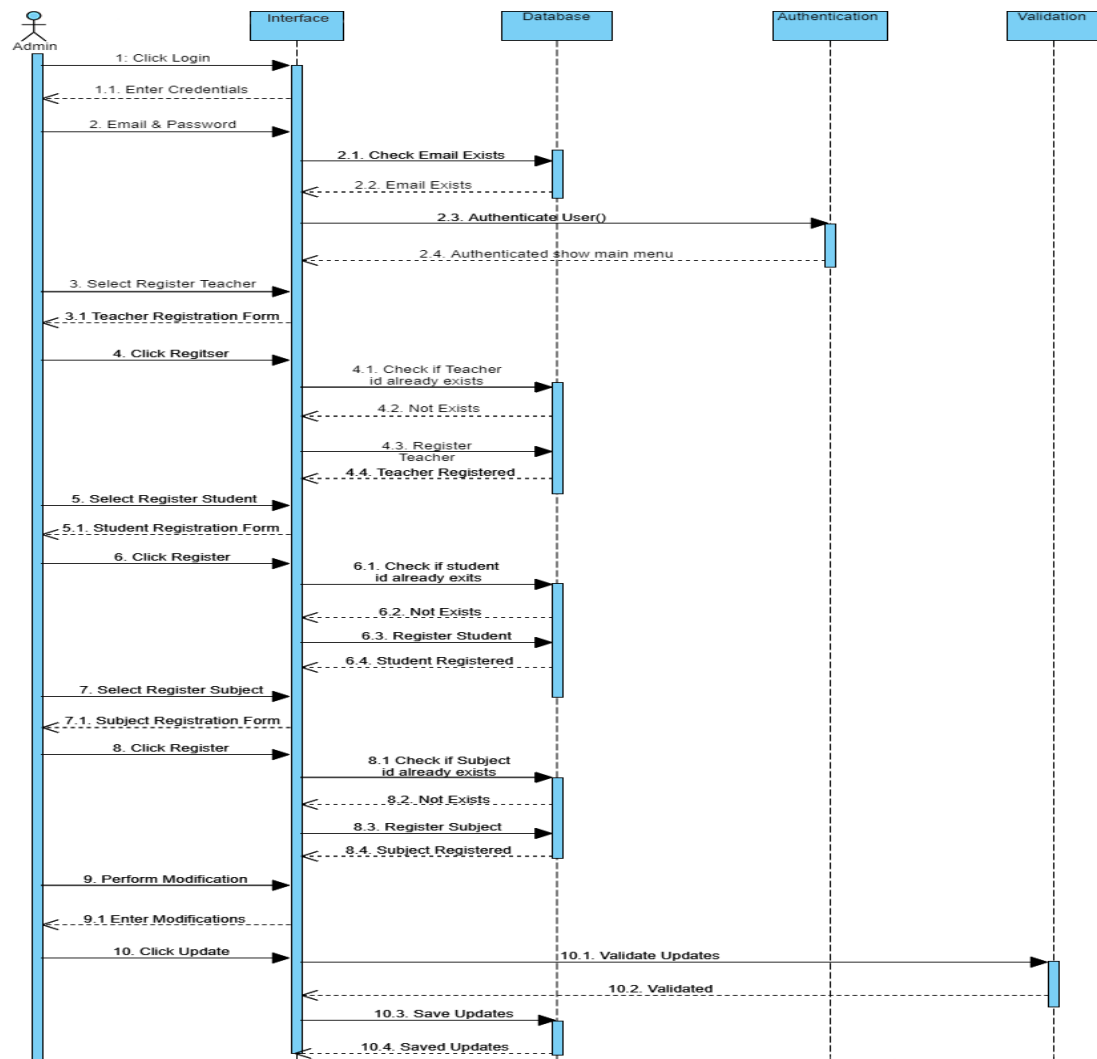


Figure 4.3 Sequence Diagram 1

#### 4.2.1.2 Sequence Diagram # 2

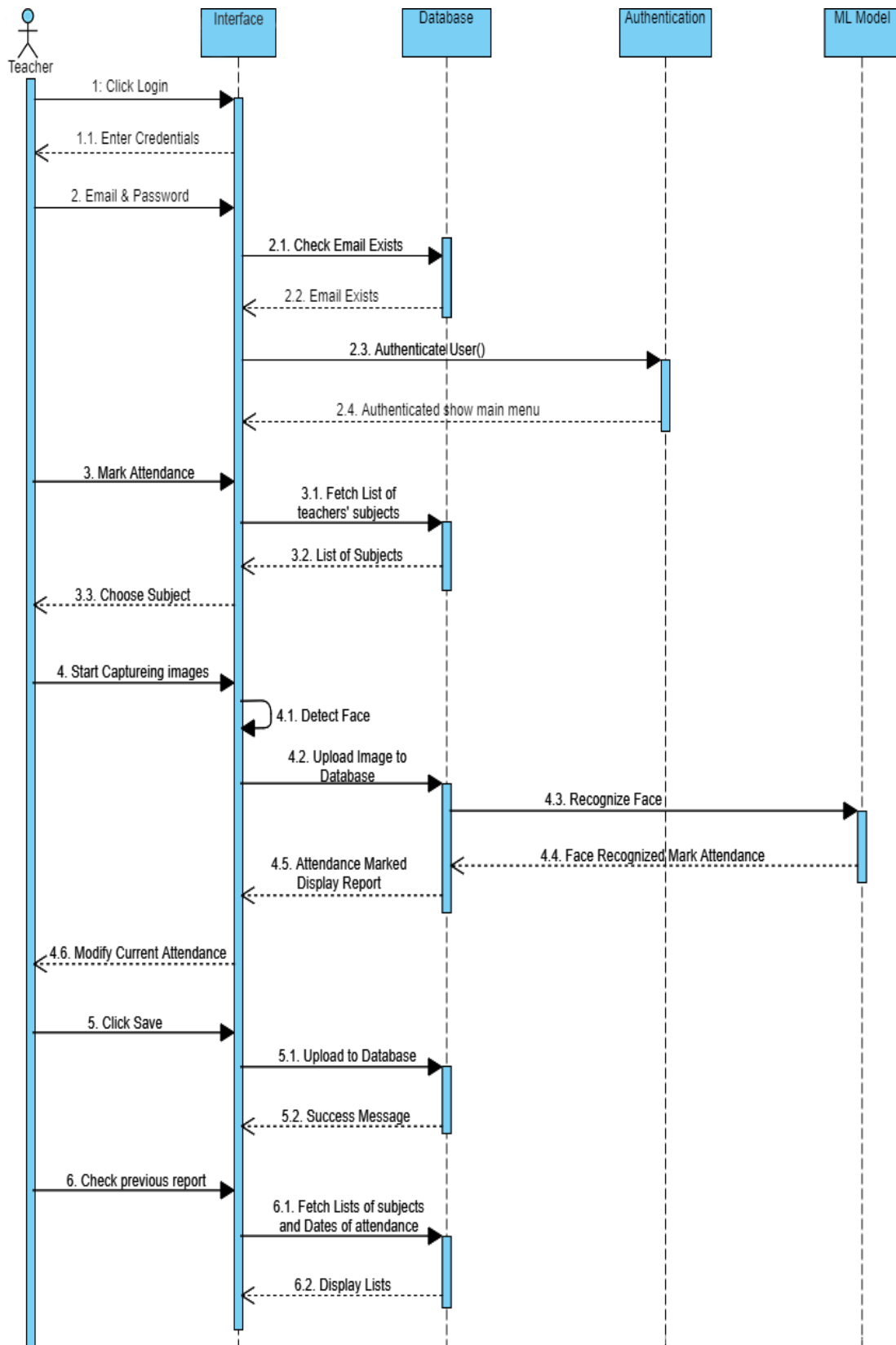


Figure 4.4 Sequence Diagram 2

#### 4.2.1.3 Sequence Diagram # 3

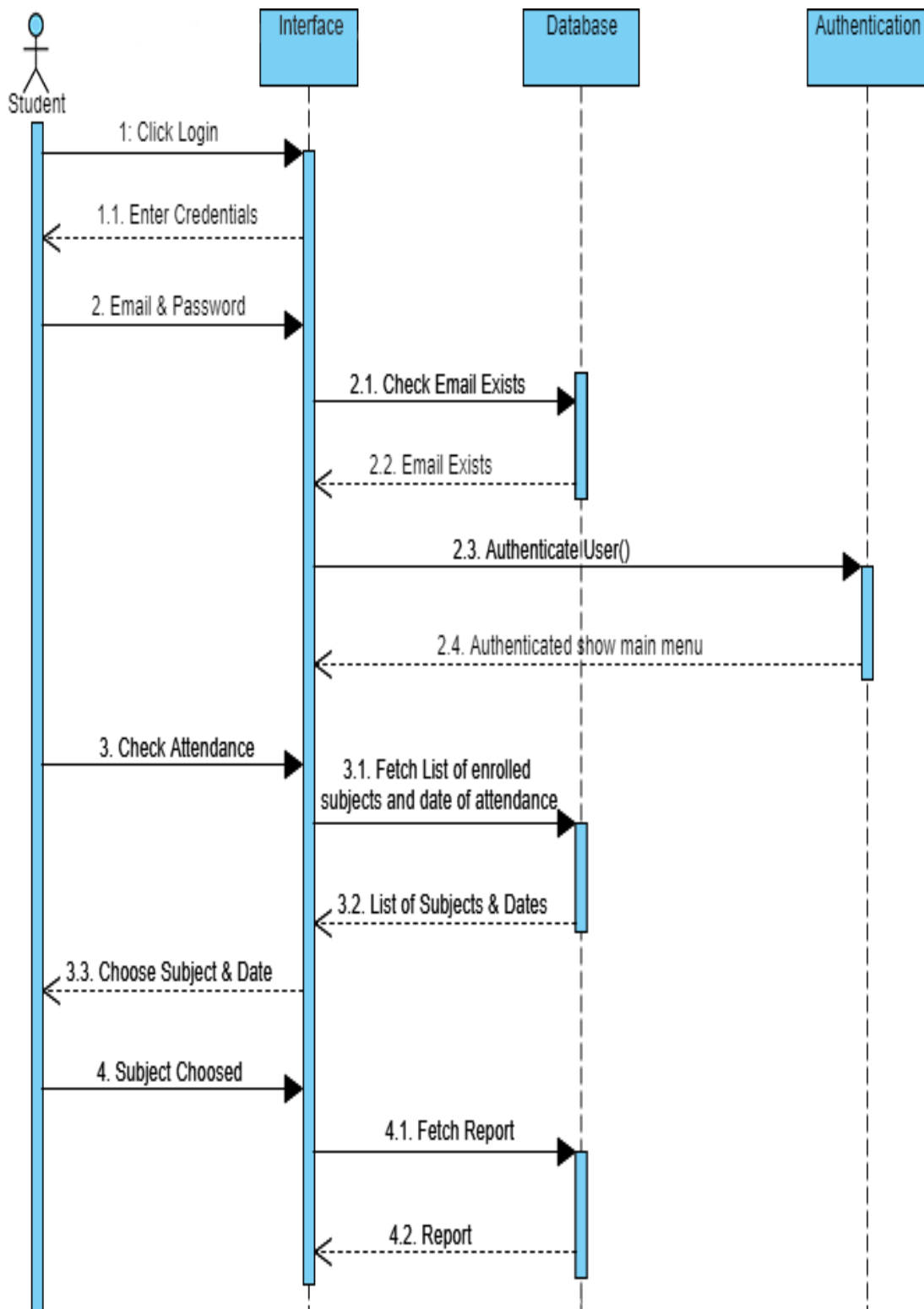


Figure 4.5 Sequence Diagram 3

## 4.2.2 Flow Charts

Flow charts represents the flow of the actions a user perform to reach the desired output. Flow chart also represents that a specific task is succeeded or not. Following are the some diagrams which displays flow charts of each type of user.

### 4.2.2.1 Flow Chart # 1

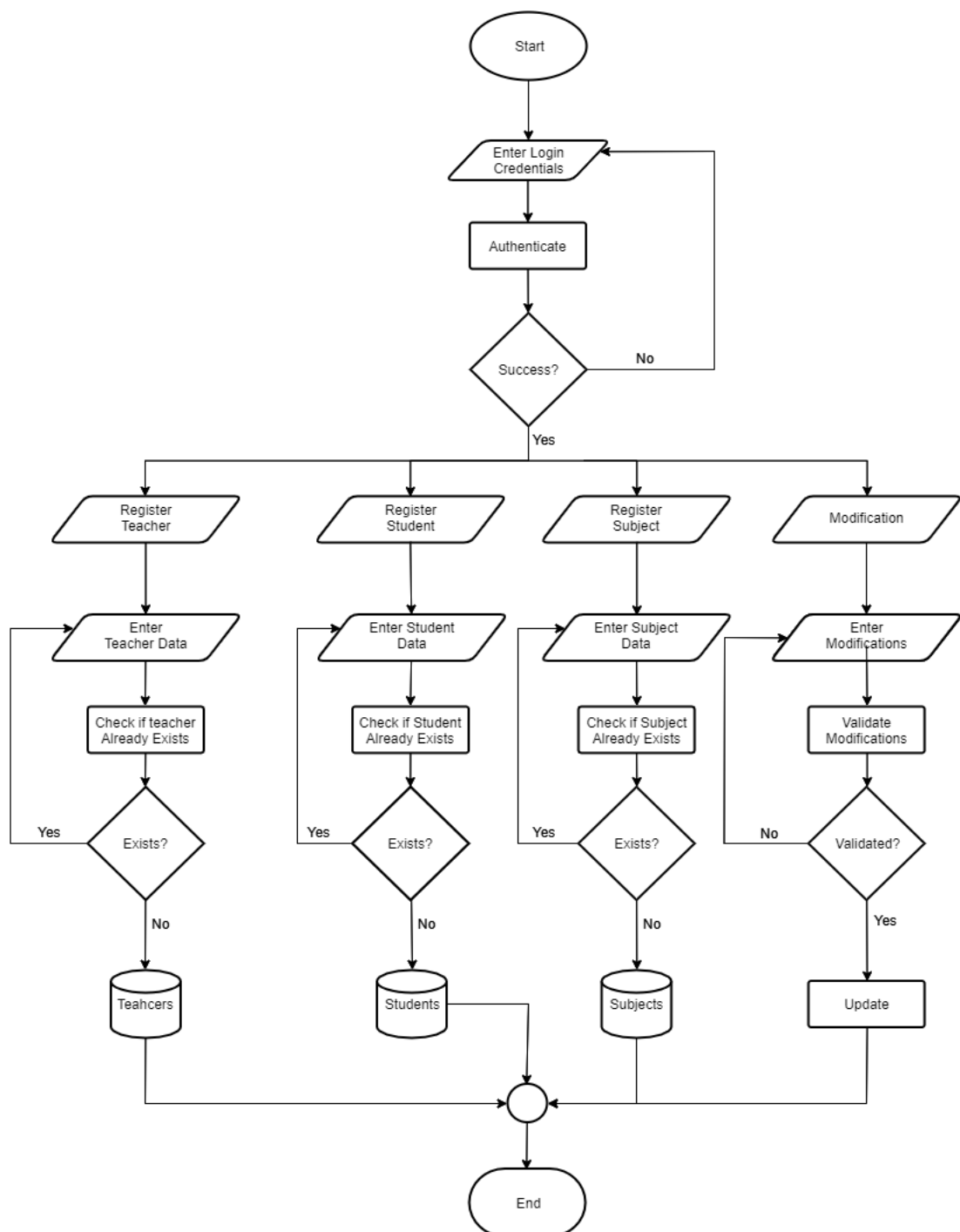


Figure 4.6 Flow Chart 1

#### 4.2.2.2 Flow Chart # 2

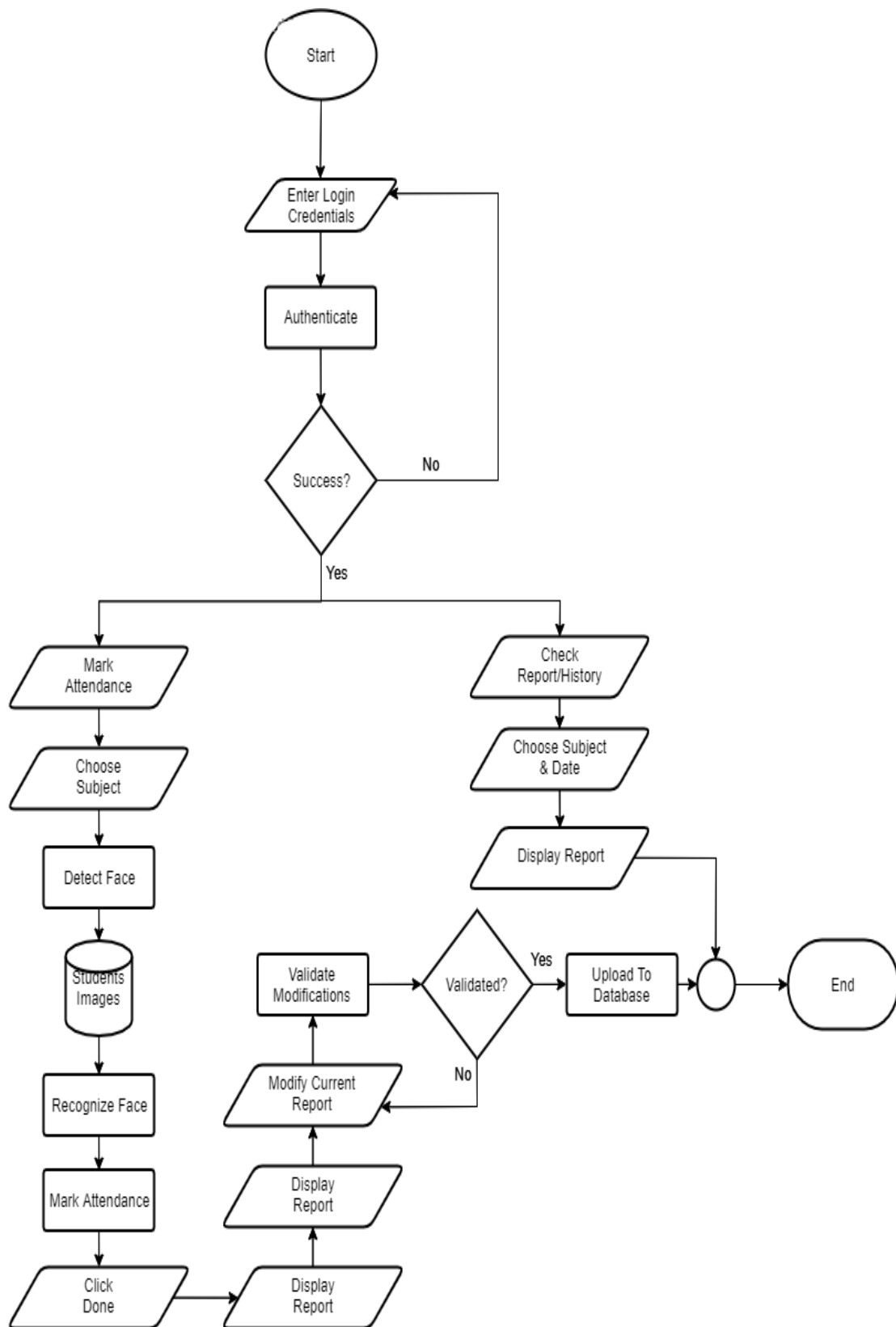


Figure 4.7 Flow Chart 2

#### 4.2.2.3 Flow Chart # 3

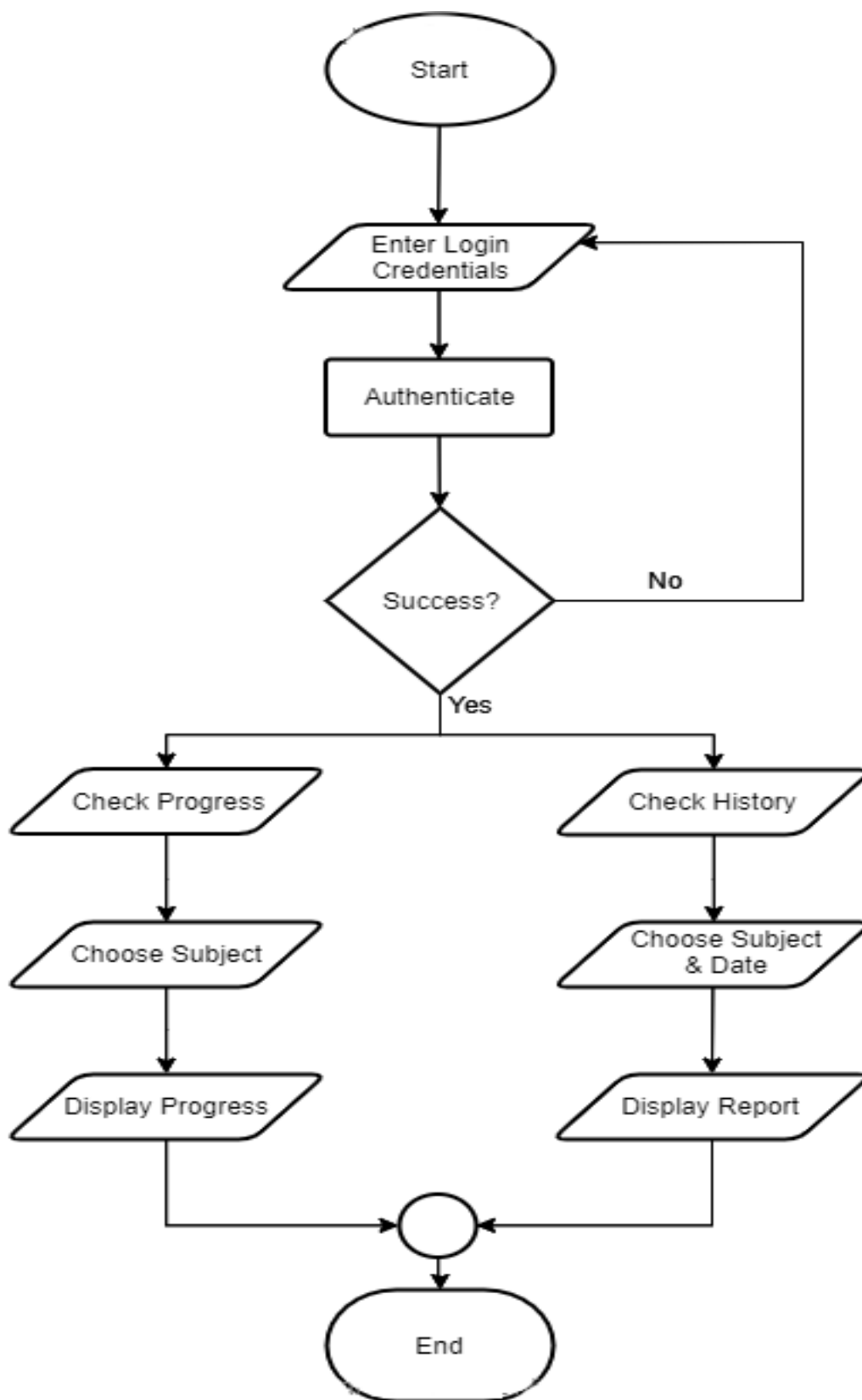


Figure 4.8 Flow Chart 3



## 4.3 Implementation Procedures

After gathering all the requirements and dataset, we started developing the admin app, which will be able to register teacher, student, and subject. The admin app will also be responsible for uploading the images of the student's faces the minimum requirement for student images are 3 more will be better. The admin will then assign an id and password to each teacher and student which they can be used to login to their app. The admin will also be able to modify anything except the attendance.

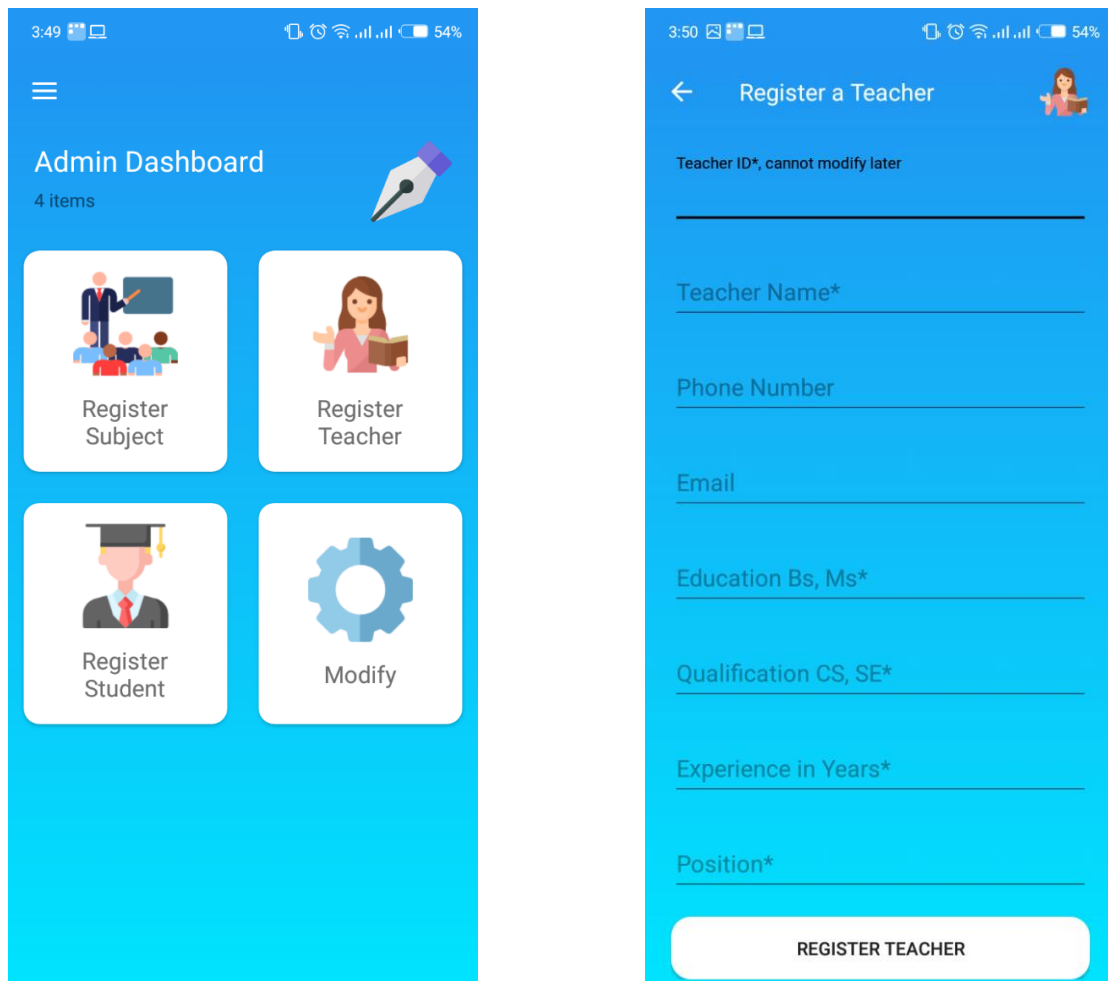
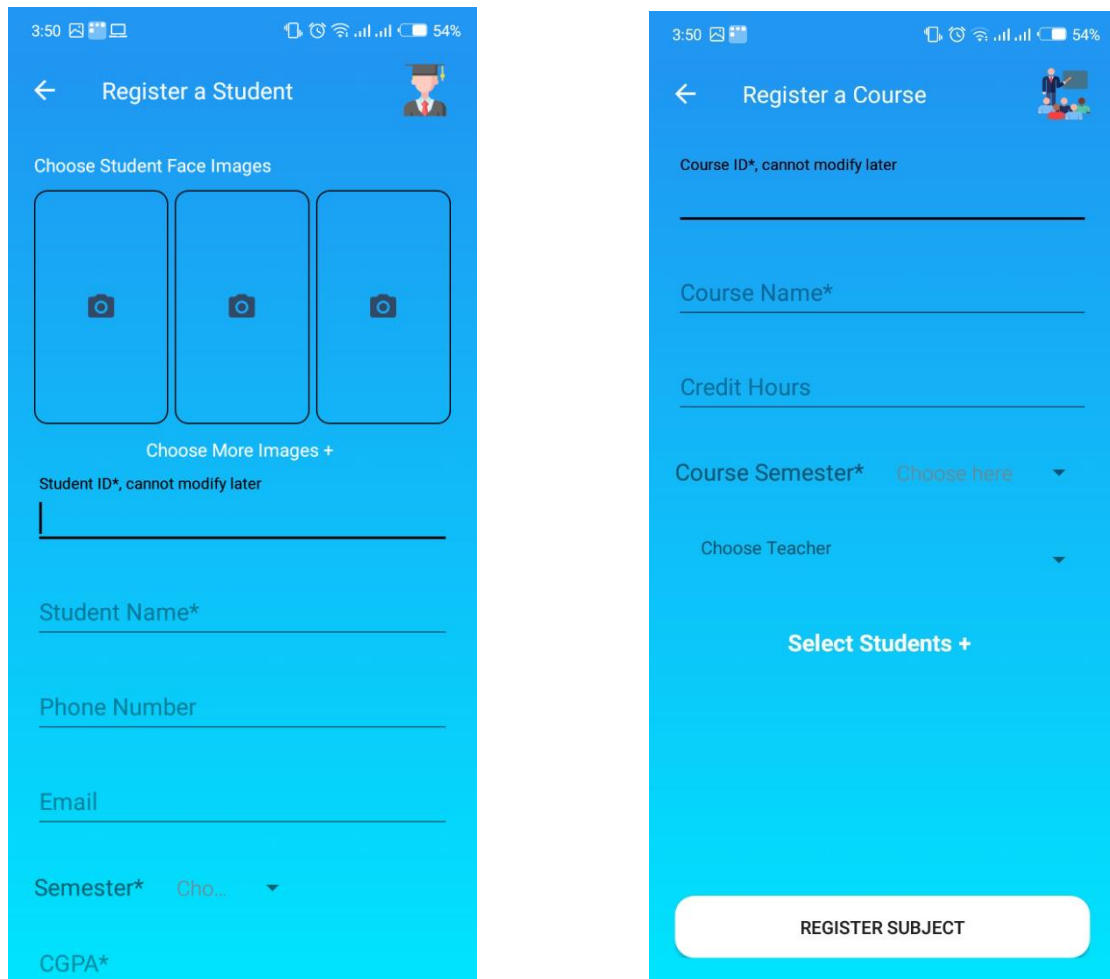


Figure 4.9 Admin Dashboard & Register Teacher



**Figure 4.10 Register Student & Register Course**

#### 4.3.1 Details about Hardware

Table 3.1 and 3.2 shows the details of the hardware that were used to develop the android applications. The first laptop Lenovo IdeaPad 100S was used to gather requirements and dataset for the project. The other laptop HP Elitebook was used to code the apps using Android Studio, which is a powerful software that needs a powerful system to run. HP Elitebook is an i5, 3<sup>rd</sup> generation system with 8 GB RAM and 320 GB Hard disk. A smartphone with android version 8.1.0 was used to test and analyze the model and an android emulator was used to test the design and functionalities of the app.

#### 4.3.2 Details about Software/algorithms

Android studio is the software which was involved throughout the project development, which was used to develop the applications.

#### 4.3.2.1 Android Studio

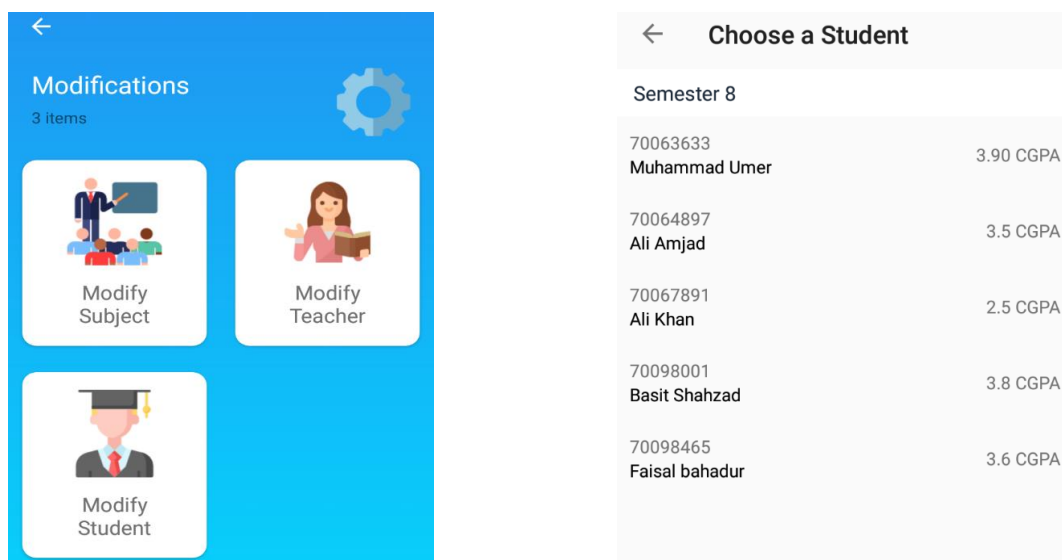
Android Studio is a software tool developed by Google Inc. for its Android operating system. It was developed on Jet Brain's IntelliJ IDEA software. Its specification is to develop full fledged Android applications. It is free to use software with no license and hence no payment is required. It can be used on window, Linux and even Marcos. It was written in three different programming languages, Java, Katlin, and C++. It requires a minimum 5 GB for installation and 8 GB RAM to run smoothly.

#### 4.3.2.2 Algorithm

We have used a linear sort algorithm to store data in the database in a sorted form, which will help in filtering the results when we will analyze our data. We have used linear search algorithm to fetch searched results from the database.

#### 4.3.3 Details about control etc

Everything will be under control of the admin, he can assign a subject to one teacher, he can enroll students into a subject, the minimum requirement of students for a subject is 5 students. The admin cannot control attendance, we are not giving access to the attendance to admin. Attendance will be marked by teachers and get stored into the database after that teachers and students can only see their attendance according to their subject but no one is allowed to edit the attendance.



**Figure 4.11 Control of Modifications**

## 4.4 Verification of functionalities

We have verified each functionality of the app, the functionalities are being verified by the tester using different scenarios, we have also made corrections to the functionalities if a functionality fails to fulfil its task. The functionalities were being test with the implementation as one functionality successfully implemented tester test it, and give remarks. After the complete implementation of the functionalities we have verified the functionalities by three users. Following are the screenshots of the apps captured while verifying the functionalities.

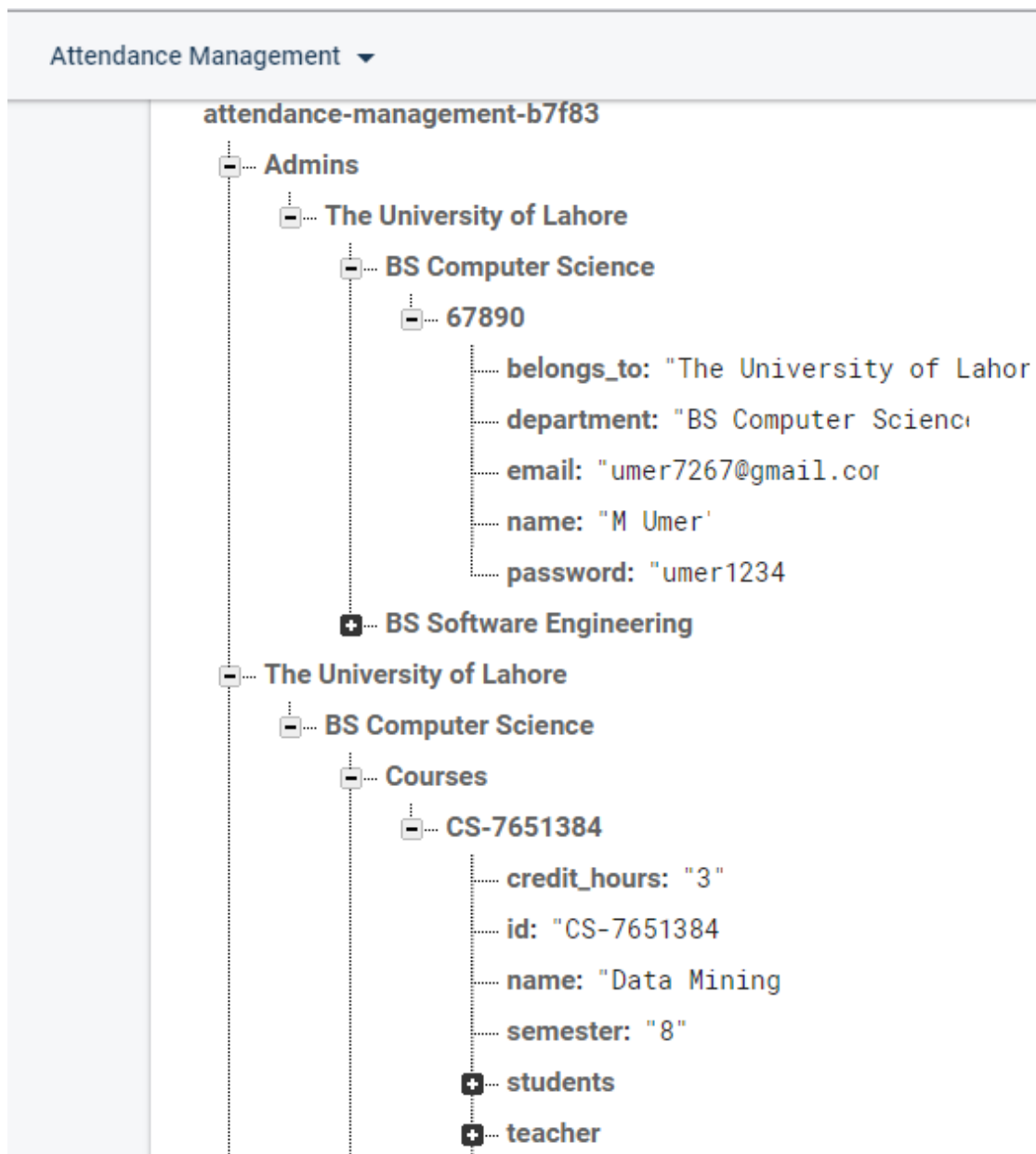


Figure 4.12 Firebase Real Time Database

## **4.5 Summary**

This chapter explains the methodologies used to get the desired results from the application. The first thing was to perform a survey to identify the needs, wants, and requirements for the application. The second thing was to collect the dataset of student's face images. After then analysis was performed on the requirements and the dataset and created a flow of the project. Then the implementation was performed. After the implementation functionalities were verified.

## Chapter 5

### SYSTEM TESTING

System Testing will involve the checking of our system validity. To ensure the quality performance of our app we will apply different kinds of checks on our app in a sequential way. These are Some of the testing phases that are used in our system are mentioned below.

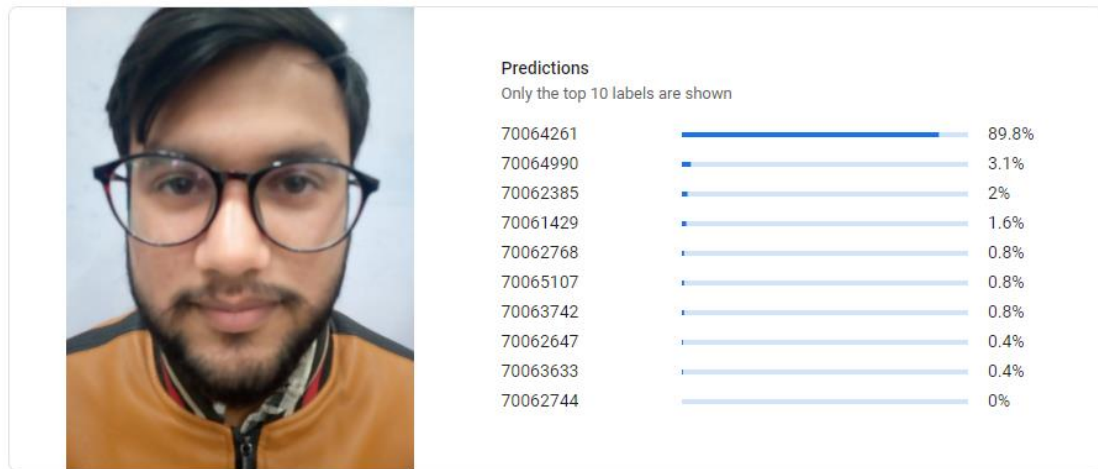
1. Objective testing
2. Usability Testing
3. Software Performance Testing
4. Compatibility Testing
5. Load Testing
6. Security Testing
7. Installation Testing

#### 5.1 Objective testing

In objective testing phase, we have tested our trained model on the images that were never shown to the model during the training of the model to evaluate our model. After this, we were implemented library Firebase Machine Learning Kit that comes with an easy to use set of machine learning models for common mobile use cases some of these models includes: detecting faces, labelling images, recognizing text, and much more.

Once we finished images classification using the Firebase ML-Kit library, we performed image processing on our data. When we have done the training of our model, we were given our model test data in order to check our model whether it is giving the desired output or not. The model that we were using in our system done the testing on our collected data and also display the results on the screen. The testing phase enables our project that our model is working correctly well and also is fulfilling our requirements. The main significant objectives of Software testing are as per the following below: Gaining trust in as well as giving data about the degree of value. An objective test is such a type of test that has correct or incorrect answers, therefore, it can be checked Fair-mind. Our project Software testing has led to different goals and

objectives. After this, we run our app in Android to test its capabilities and also to check whether it was fully functional or not.



**Figure 5.1 Objective Testing**

## 5.2 Usability Testing

This testing phase refers to carried out to measure how much the user will use the system. Our system (app) can easily be used by different Android smartphone users. But in the future, we are going to make it available for almost all the users. For the time being due to lack of technology and there is no common platform of detection as well as recognition of faces for everyone to use such type of app that fulfils the requirements of daily base security problems. The usability testing considers as a non-useful testing technique that is a proportion of how much the system framework can be used by an individual user. Usability testing also estimates the ease of use of particular software. It is to figure out how friendly or interactive is the interface of the app is for the users. We performed usability testing on our app to examine the easiness and friendliness of the app.

### 5.2.1 Natural Mapping

We applied natural mapping in our system Graphical User Interface in order to gain users' attraction and also provided the users' ease so that users can understand what the system is giving them.

Figure 4.2 and 4.3 elaborate that the app is easy to use, everything seems to be very clear, the User interface is attractive and simple, the buttons are showing icon as well as the text to make it very clear to the users.

### 5.2.2 Task Analysis

There we used different types of task analysis to separate each and every task of our system into different steps to ensure that the user does not get any type of annoying or unsatisfactory from our system. We also analyze how the user will understand how to use the system more frequently and more easily. This phase made able our system friendlier as well as getting more attraction from users.

## 5.3 Software Performance Testing

We have got about 90% performance rate from our system (app) after checking and testing again and again. We also ensured that it should not be slow down or any failure while performing required operations on the given data in the form of pictures and operations were ensured the required accurate results. Once Testing is done then there was no error/bug in our system or app. When we got the required results, we observed that our app's performance almost is very nice. It leads to our system's reliability and also very fast. According to our system's users limit, our system or app doesn't crash while processing on given data. It takes just a few seconds or minutes and sometimes it tests without taking any time while it is processing on given data. Here Execution testing is such a type of method that decides the system's efficiency, responsiveness, and dependability of a computer, organize programming system, or electronic device under a remaining task at hand.

## 5.4 Compatibility Testing

This phase performed to analyze the compatibility of our software on different platforms as well as operating systems. Our system software can run on API level 19 Android version 4.4 to the latest API level 29 android version 10. We have tested our system and also examine that our system can efficiently be working on Android versions 6, 8, and 9. Overall our system software is much compatible with all android versions.

The following table elaborate the compatibility testing of our software. The software seems to be compatible with 98% devices of android on the Google Play Store.



**Table 5.1 Compatibility Testing**

Test cases no	Operating System	Different test Support	Compatible
1	Android	Yes	Yes
2	Windows	No	No
3	Mac	No	No
3	Linux	No	No

## 5.5 Load Testing

This testing phase is about how much data can be processed at a time by our system software. There we have tested with 20 users at a time and we see that all users were successfully performing the operations of read and write data from the database. There was no effect on our app's speed and our app was running smoothly. We do not face any error while data was reading and writing during load testing. The response time of our app was very good and overall, all the data was correctly be retrieve and store from the database.

## 5.6 Security Testing

In Security testing, we have ensured our app's security that no one can miss-use data of each other. By applying security tests, we disclose all the weaknesses of our software. We applied several security checks in order to make secure the data of users and we also examine whether data can be accessible or not to admins of different departments within the institution. We also make secure the data and attendance of students that no other one can change or miss-use of their data. Our app provides a secure platform for their users so that all the users can trust and also easily perform their functions.

```
1 {  
2   /* Visit https://firebase.google.com/docs/database/security to learn more about security rules. */  
3   "rules": {  
4     ".read": "auth != null",  
5     ".write": "auth != null"  
6   }  
7 }
```

**Figure 5.2 Database Security Rules**

## **5.7 Installation Testing**

Installation testing is referred to as the installation of the app. It has much importance in the software development life cycle (SDLC). To ensure all activities are working efficiently we first did the installation of our app. Once the installation of our app has successfully be done then we were able to check all the activities of our system whether all the activities are working smoothly or not. It has very significant for system success so there we done further two tests in order to examine all the functionality.

### **5.7.1 Activity Testing**

We tested each and every activity individually of our app. There almost 15 activities throughout 3 apps. We have created as well tested each and every activity with details to ensure that whether anything is left or not. But we have observed that our app was fully completed from every respect. All the activities are working successfully with simple and user-friendly User Interface (UI) design.

### **5.7.2 Operation Testing**

We have tested the following functions of our app that were working properly. Our app performs the following main operations:

- Log in
- Register teacher
- Register student
- Register subject
- Modification
- Face recognition
- Mark attendance
- Check report.

These are the operation we have checked and all the operations were functional. There was no error or bug in the database while we were performed above operations and the database was also working efficiently.

## 5.8 Test Cases

A test case is very effective in order to check all types of functionalities. By using each and every test case our application is able to work efficiently and properly. Overall, it was a very great experience under which conditions or factors testing being done. The path that was followed by us to find the problems and issues of the application was in sequentially and well mannered.

### Test Case # 1: User Login

#### Software:

Android Operating System 8.1.0

#### Test Description:

User log-in by using his register email by admin. He simply enters his id and password then press the login button.

#### Testing Environment:

An Android mobile phone is with Android version 8.1.0 (Oreo).

#### Test ID: 1

Table 5.2: Test Case # 1

Preconditions	An Android smartphone and an internet connection is required
Actions	User will enter a valid user id and password provided by the admin
Expected Results	User will successfully login in order to perform the required activity
Result	User will successfully login in order to use the system

### Test Case # 2: Registering teachers, students, subjects

#### Software:

Android Operating System 8.1.0

**Test Description:**

Only admin can register the teachers, students, and subjects. The registration would be done according to the position of a user such as a teacher, student, and subjects.

**Testing Environment:**

An Android mobile phone is with Android version 8.1.0 (Oreo).

**Test ID: 2****Table 5.3: Test Case # 2**

Preconditions	Internet should be connected
Actions	Admin can register teachers, students, and subjects
Expected Results	Teachers, Students, and Subjects should register successfully
Result	Teachers, Students, and Subjects were registered successfully

**Test Case # 3: Modifying teachers, students, subjects****Software:**

Android Operating System 8.1.0

**Test Description:**

Only admin can modify the teachers, students, and subjects. The modification would be done according to the position of a user such as a teacher, student, and subjects.

**Testing Environment:**

An Android mobile phone is with Android version 8.1.0 (Oreo).

**Test ID: 3****Table 5.4: Test Case # 3**

Preconditions	The Internet should be connected.
Actions	Only Admin can modify top the teachers, students, and subjects
Expected Results	Teachers, Students, and Subjects should modify successfully
Result	Teachers, Students, and Subjects were modified successfully

#### **Test Case # 4: Face recognition**

**Software:**

Android Operating System 8.1.0

**Test Description:**

The app will recognize the face after detecting an image. Face recognition would be done by the model on images that was used to train the model.

**Testing Environment:**

An Android mobile phone is with Android version 8.1.0 (Oreo).

**Test ID: 4**

**Table 5.5: Test Case # 4**

Preconditions	Internet should be connected
Actions	Face recognition would be done only by the system automatically
Expected Results	Face recognition should be done successfully
Result	Recognizing faces with 80% of accuracy

#### **Test Case # 5: Automatic Attendance Marking**

**Software:**

Android Operating System 8.1.0

**Test Description:**

The app will automatically mark the attendance of student after recognizing his face.

**Testing Environment:**

An Android mobile phone is with Android version 8.1.0 (Oreo).

**Test ID: 5**

**Table 5.6: Test Case # 5**

Preconditions	Internet should be connected
Actions	Automatically attendance marking in the teacher's app.

Expected Results	The attendance should mark automatically.
Result	Automatic attendance gets marked after face recognition.

### **Test Case # 6: Report Generation**

#### **Software:**

Android Operating System 8.1.0

#### **Test Description:**

The report would be generated by the system automatically after the marking attendance from the teacher.

#### **Testing Environment:**

An Android mobile phone is with Android version 8.1.0 (Oreo).

#### **Test ID: 6**

**Table 5.7: Test Case # 6**

Preconditions	Internet should be connected
Actions	The system would generate the attendance report of the student
Expected Results	The report should be generated by the system
Result	The report was generated by the system

## **5.9 Specific Requirements**

This section includes overall all functionality of our product (app), functional and non-functional requirements, test cases, use cases, external interface requirements, inverse interfaces as well as design constraints related to our system.

### **5.9.1 Functionalities**

This section of documentation includes the requirements related to the face recognition automatic attendance management system. It also contains what actually system will do and what functionalities it won't perform. In this section, we will also discuss the features of our system as well as the characteristics that proposed by our system. The Functionalities of our system leads to the completion of our project. If the system does fulfil the functional requirements of a system user, then it is considered as successful

otherwise if the system does not fulfil the user requirements then it is considered as a fail.

**Table 5.8: User Login**

<b>Introduction</b>	A teacher and student should have to get registered in order to use the application. The application will not work until the registration process will complete; the admin will responsible for registering teachers, and students.
<b>Input</b>	The user will click on the log-in button in order to login.
<b>Processing</b>	<ol style="list-style-type: none"> <li>1. The user will enter id and password provided by admin to get log-in.</li> <li>2. When the user will successfully login, then he will be able to use the application.</li> <li>3. Now, the user can perform activities according to his position.</li> </ol>
<b>Output</b>	An android app will display the home page.
<b>Error Handling</b>	If the user uses an invalid id and password then the app will show error and suggest filling a valid user name and password.

**Table 5.9: Registering of Subject, Teachers, & Students**

<b>Introduction</b>	The admin will do all the stuff related to registering new classes, teachers, and students. He will enroll the student in
---------------------	---

	different subjects and assign the subject to a teacher.
<b>Input</b>	The admin will click on the register button of student, teacher, and subject in order to registration respectively.
<b>Processing</b>	<p>The admin will</p> <ol style="list-style-type: none"> <li>1. Enter the student name, subject name and the teacher name</li> <li>2. Enter the subject, student, teacher id.</li> <li>3. Assign subject to a teacher from the list of registered teachers.</li> <li>4. Enter the semester of the subject.</li> <li>5. Enrol students in the subject.</li> <li>6. Click done.</li> </ol>
<b>Output</b>	The registering of students, teachers, and subjects will be done successfully.
<b>Error Handling</b>	If the student, teacher, and subject, with the same id, already exists, the app will not allow registering the new student, new teacher, and new subject with the same id, same teacher name, and same subject name.

**Table 5.10: Face Recognition**

<b>Introduction</b>	After the success of face detection in the image the application will perform a background function of face recognition with the images stored in the database.
---------------------	---



<b>Input</b>	The app will automatically perform face recognition of the detected image with database images.
<b>Processing</b>	<ol style="list-style-type: none"> <li>1. After the detection of the image, the image will be recognized</li> <li>2. The image will be recognized with the images that was used to train model</li> <li>3. Once the recognition of the image will be done, attendance will be marked</li> </ol>
<b>Output</b>	The image of the student will be recognized successfully by the app.
<b>Error Handling</b>	If the system does not contain the image in the database while the system will be recognizing then it may be show error of unavailability of the detected image.

**Table 5.11: Automatic Attendance Marking**

<b>Introduction</b>	The automatic attendance of the students will be marked after a successful recognition of the face of a student. The attendance will get stored in a database for the record and future use.
<b>Input</b>	The app will automatically perform attendance marking after face recognition.
<b>Processing</b>	<ol style="list-style-type: none"> <li>1. In the first step the face will be recognized.</li> </ol>

	2. The attendance will be marked automatically after the image successfully matching.
<b>Output</b>	The automatic attendance will be marked successfully.
<b>Error Handling</b>	When the app does not capture the image or due to low light and low-resolution camera can affect the attendance marking.

**Table 5.12: Editing of Attendance**

<b>Introduction</b>	The application will provide an ability to teachers to edit the current attendance once it marked by the application but a teacher will not be able to edit the attendance once it will get saved in the database.
<b>Input</b>	The teacher can edit the attendance of the students by using edit button
<b>Processing</b>	<ol style="list-style-type: none"> <li>1. In the first step, the attendance will get marked by the app.</li> <li>2. The teacher will also update the attendance after editing attendances</li> <li>3. Once the attendance will be edited, then the use case this activity will be terminated</li> </ol>
<b>Output</b>	The attendance will be edited successfully.

<b>Error Handling</b>	No error occurs in all the attendances will be edited easily in excel sheet
-----------------------	---

## 5.9.2 External Interface Requirements

External interface requirement refers to those requirements that are included in the components of database and programming as well as types of equipment of the system. The components of the database and programming make sure the framework in which the system's interface would be.

### 5.9.2.1 User Interfaces

The user interface of the application will be very simple and easy to use so that a non-technical user can easily understand everything. The following are some basic interfaces of the application.

1. User Login.
2. User Dashboard.
3. Registration forms.
4. Report forms.

### 5.9.2.2 Hardware Interfaces

The device should be enabled with the internet and should have a high-resolution camera.

## 5.9.3 Software Interfaces

The user's android version should be between 4.4 and 10 for running the app.

## 5.9.4 Communications Interfaces

The application uses the internet through mobile data or Wi-Fi to communicate with the Firebase server in order to fetch and store data. The data language in the application will be English since our application is designed to target more audience.

## 5.9.5 Functional Requirements

Functional requirements are considered as the basic functionality of the system that makes it successful or failure. If the system performs all the required functionality and

also meet all the basic requirement of the user then the system is known as successful otherwise when the system does not fulfill desired results then it is considered as failure. Our system has the following functional requirements below.

#### 5.9.5.1 Functional Requirement 1

**User Login:** A teacher and student should have to get registered in order to use the application. The application will not work until the registration process will complete.

#### 5.9.5.2 Functional Requirement 2

**Registering of Subject, Teachers, & Students:** The admin will do all the stuff related to registering new classes, teachers, and students. He will enroll the student in different subjects and assign the subject to a teacher.

#### 5.9.5.4 Functional Requirement 3

**Face Recognition:** After the success of face detection in the image the application will perform a background function of face recognition with the images stored in the database.

#### 5.9.5.5 Functional Requirement 4

**Automatic Attendance Marking:** The automatic attendance of the students will be marked after a successful recognition of the face of a student. The attendance will get stored in a database for the record and future use.

#### 5.9.5.6 Functional Requirement 5

**Attendance Maintenance:** Marked attendance will be maintained in order to check the attendance performance of the students in the institute. Maintained attendance will be used by the institute to perform analysis to improve their processes.

#### 5.9.5.7 Functional Requirement 6

**Editing of Attendance:** The application will provide an ability to teachers to edit the current attendance once it will be marked by the application but a teacher will not be able to edit the attendance once it will get saved in the database.

#### 5.9.5.8 Functional Requirement 7

**Student Progress:** The application will automatically calculate the attendance percentage of the student in each subject and mark the subject red if a student has short attendance in the subject.

### **5.10 Non-Functional Requirements**

Non-functional requirements of the system are related to the system's characteristics that include the design of the system, performance, security, Player experience, availability, portability, maintainability, reliability of the system as well as the effect of run-time behaviour.

#### **5.10.1 Performance**

The system must be interactive and the delays involved must be less. So, in every action-response of the system, there are no immediate delays. In case of delay e.g. saving the attendance, the attendance report should be saved as a draft which can be saved into the database after the delay.

#### **5.10.2 Reliability**

The system should be reliable enough that it should not fail its operations if the load is high. It must be made sure that the system is reliable in its operation and for securing sensitive information.

#### **5.10.3 Availability**

The system should be available 24/7 days. In case of unavailability, it should show a message to users with the proper reason for unavailability and the remaining time of the unavailability. It should be available for the long-range of available android device versions.

#### **5.10.4 Security**

Every user should be signup with email and the teacher must verify their email before using the system. The system should authenticate each user before sign in. It should save users data in a secure server and the data read-write rules should be properly defined for each type of user, only the authenticated users should be allowed to read specific data.

#### 5.10.5 Maintainability

The system should be easy to use and maintain by the users. In case of any confusion and difficulty, there should be help for each function a user can perform. Furthermore, a user should be able to write an email to developers in case of any crashes in the system.

#### 5.10.6 Portability

The system should only run on the android hardware device. It should be portable to run on android tablet devices. It should also support a wide range of more than 90% devices on the google play store.

#### 5.10.7 Efficiency

The system should be able to handle more than 15,000 writes and more than 40,000 read requests in a day. Its response time should not be more than 30 seconds in case of 4G internet speed. The system should have a capacity of storing students' images more than 5 GB and attendance reports more than 1 GB.

### **5.11 Inverse Requirements**

Inverse requirements are used to represent the functionalities that the system should not have for example, the system interface should not in red colour. Following requirements describe the inverse requirements of the system.

#### 5.11.1 Interface Requirements

The interface of the system should not have a dark colour theme. It should not use larger and bold text fonts which make UI less attractive.

#### 5.11.2 Modification

The students should not allow modifying anything into the reports and in case of anything, wrong students should contact admin for correction. A teacher should not be able to modify previous attendance but he can modify the current attendance.

#### 5.11.3 Subject enrollment Limit

The limit for a subject to enroll students should not be more than 50 students, so the admin will only register not more than 50 students in a subject.

#### 5.11.4 Report Reading

A teacher should not be able to check or read reports of the subjects which are not assigned to him. Likewise, a student cannot check progress or report of any other subject in which he did not enroll.

### 5.12 Design Constraints

Design constraints represent the limitations the system will have, in order to overcome these limitations, we may need to bear more cost. Following are the some constraint of the current system.

#### 5.12.1 Limited Storage

As the system back end is developed in a firebase spark plan which is free to use. The firebase gives limited storage of 5 GB in storage and 1 GB in the database in this plan which can affect the system when the user base is high.

#### 5.12.2 Targeted Android Versions

The system will only support Android API level 19 (4.4) to 29 (10) which supports more than 95% devices on the Google Play Store. The developer has to write extra code for older versions of the android which causes large app size which will not really useful for users.

#### 5.12.3 Limited users

Firebase only allows 10 thousand users to be registered in the spark plan if the users are more than this limit then the owner of the system should have to buy other firebase plans in order for the smooth working of the app.

### 5.13 Logical Database Requirements

Database requirements describe the requirements for the data being in the backend server. How the database will maintain and how should we store data in the database.

#### 5.13.1 Offline Database

The system should use an offline database in order to store a report in case of the delay of the internet, so the marked attendance should not get lost. Android SQLite database is one of the great android offline databases which should use in this case.

### 5.13.2 Online Database

The Firebase as a backend service provides a RealTime Database that updates modifications in real-time. The system should implement the Firebase RealTime Database in order to store attendance reports, teachers', students', and subjects' information.

### 5.13.3 Images Storage

The Firebase provides a storage services for storing data rather than text, which should use in this system in order to store the images of the students faces. This storage will be used to recognize faces to mark attendance.

## **5.14 Other Requirements**

Other requirements are those requirements which does not fall in any of the above requirement.

### 5.14.1 Scrum Development Model

Scrum development model works on the sprints in which the changes are entertained whenever changes occur. The system should be developed under this model and the maximum time for the completion of one sprint is 3 weeks. Scrum development model involves product owner (the university), Scrum Master (team lead), and the team. The scrum master will be responsible for synchronizing work between the product owner and the team.



## Chapter 6

# RESULTS AND CONCLUSIONS

After completing the project, we were able to achieve the desired results. We have broken the result of our projects into different sub heading below:

### 6.1 Presentation of the findings

Android mobile phone users can download it from the play store. But any user will have to contact us to get the university register. And then we will be registered them and give them. This app is free to use and free to download. After the app is installed, it will be the responsibility of the administrator to register the teacher and the student, and then the teacher will be able to mark the attendance by using face recognition, and the student can also check his attendance report. After the development of the project, we have also tested in our class it working perfectly.

#### 6.1.1 Hardware Results

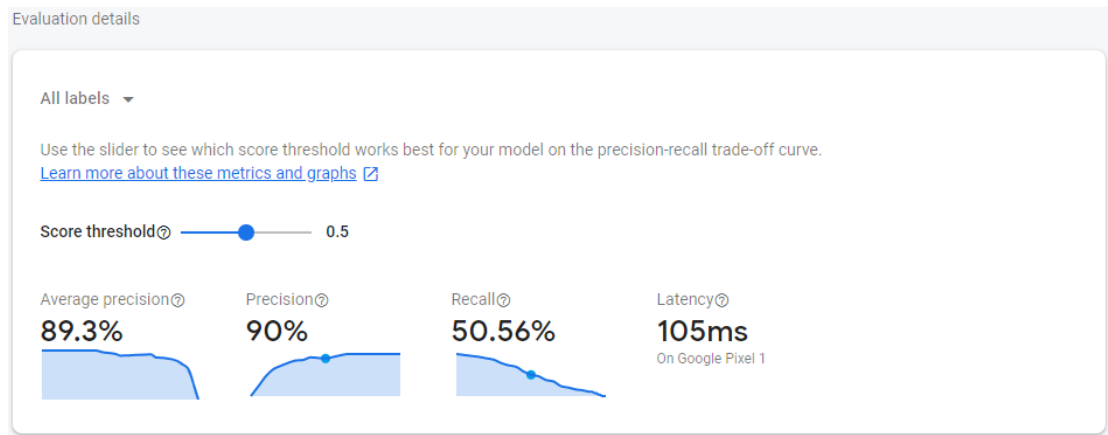
Now we have tested our project in two Android mobiles. The app is working very well on the first mobile which is Samsung s7. The app is working perfectly on this mobile and quick response. Because it is a good Android phone. There is another android mobile in which the app is very slow working. Because the other mobile is not in good quality, because of this the app is not responding quickly. Due to this, the app is delayed in performing the function.

#### 6.1.2 Software results

In this phase, we tested the app in 2 different Android versions. First Android version 9 and the second version is 4.4. Android version 9 App is working perfectly. Android version 4.4 Application not meeting requirements because this app is slow working. But overall, which was our expectation is getting full fill from the app.

#### 6.1.3 Evaluation Details

The below image showing the results of the training model the overall model is evaluated at 89.3% of average precision, the model was also evaluated and test during the training on Google Pixel 1 device which gives the 105ms delay in results.



**Figure 6.1 Evaluation Details**

### 6.1.4 Confusion Matrix

The below image shows the visualized confusion matrix of the model that describe the performance of the model while testing the model on the device.



**Figure 6.2 Confusion Matrix**

## 6.2 Discussion of the findings

We have read a lot of research paper and we have found it people have worked on this project with a laptop camera and CCTV camera. The new thing in our project was that we did it on Android mobile and in this project, we have also included side face recognition. We have read a lot of research papers and their results are also checked

and the results that we have found from side face recognition are the same. Result Accuracy is almost the same as other people working on their projects.

### 6.2.1 Comparison with initial Goal

We have changed somethings from the project proposal we gave. Because the things we wrote in the project proposal are not possible at the time. For example, we were going to work on the project our Low light, but when we tested the project and that time project was not working properly. Then we change many parameters and check. But even then, we were not achieving project achievement on low light. Then we quit working on low light. The new thing in the project was the side face recognition. Very few people worked on Side face Recognition. And we have done one thing to side face recognition. And we have achieved most of the things that we have done in the project proposal.

### 6.2.2 Reasoning for shortcomings

Because of lockdown and Covid-19, we face many difficulties. Due to this reason project slight delay, and that's why the database has been issues of failure. Because of which the database has taken a long time to set up. Now the system is at that stage that it can handle a lot of data. We hope that the future will not have such issues so that this project can be worked on.

## 6.3 Limitations

App will not work without the internet. Registering any subject, it is imperative to have at least 5 students. And if the user has an unexpected issue and it does not help in the App, then he has to contact us.

## 6.4 Recommendations

In case if anyone wants to carry my work on this project in the future to make it more powerful, my recommendations are:

- Train the model on low light images so it can be able to recognize a student's face even in low light.
- Work on distance images, currently the model is trained on very close images of faces, so it should also work on distance.

- Add analytics into the admin app so admin will have the information of about the daily attendance reports.

## **6.5 Summary**

The results of the project were really close to the expected results. We have tested in on two different android devices with different versions. In the lower version the apps are relatively slower than the latest version. The android version of this model was not developed ever before and we also added a side face detection which makes this project an awesome project.

## **6.6 Conclusion**

There are many face recognition base attendance management systems but this project is different from all of them. The project is the very first version of facial recognition in android and also the model was trained on the side face images which also make it unique from other projects. Once app is completed and tested the institutes can get benefits from it by making their attendance system automatic for free of cost.

## Chapter 7

### FUTURE WORK

Our project was to develop a facial recognition based attendance management system using an android device. First, we have collected a dataset of student's faces in the form of images, then we have trained our model using Firebase ML kit. Using this app teacher can mark attendance of his students automatically by simply opening the camera and face it towards students. The model will detect the face of the student and recognize it, and then mark his attendance in the database. Our model was efficient enough to detect and recognize the face of the student from the side. The future of facial recognition base attendance marking has a high scope. We are going to contain this project in the future and expand it to a larger scale. We will work on collecting a large dataset of student's faces, and then test the model on a larger number of classes. We will also work to implement the same project on iOS. There are many other things we have thought to work on it in the future for example.

We will work on making it work under the low light, the basic idea behind this is if a classroom has relatively low light when the lights are off and the teacher is teaching on the projector then the student enters the classroom, we want it to recognize his face. For this, we have to collect a large collection of student's images in low light to train our model on low light images.

There is another scenario where our model should work, the model should be able to detect and recognize student face from distance, currently, we have collected a dataset of student's faces very closely, we will work on to collect dataset from a distance, then we will train our model and test it to check whether we can achieve this or not, but we will try this to make our project work in every possible scenario.

Human eyes are unique from other humans, we will also work on our model to also recognize eyes because many times female students wear a niqab and recognizing their face it is impossible. So, our model should be able to recognize the eyes of the student to mark her attendance.

There are many other cases and scenarios in which our model should work, we will work on to further think of scenarios where our model can fail, and then we will work on to overcome it.

## REFERENCES

- [1]. Khan, S., Akram, A. & Usman, N. Real Time Automatic Attendance System for Face Recognition Using Face API and OpenCV. *Wireless Pers Commun* (2020).
- [2]. Naveed Khan Balcoh, M. HaroonYousaf, Waqar Ahma nd and M. Iram Baig, "Algorithm for efficient Attendance Management: Face Recognition Based approach", *International Journal of Computer Science Issue*, vol. 9, no. 4, July 2012.
- [3]. N. Kar, M. Kanti Debbarma, A. Saha, and D. Rudra Pa: Study of Implementing Automated Attendance System Using Face Recognition Technique, Volume 1, No. 2, international Journal of Computer and Communication Engineering (2012)
- [4]. Liao S., Jain A.K., Li S.Z. Partial face recognition: Alignment-free approach. *IEEE Trans. Pattern Anal. Mach. Intell.* 2012; 35:1193–1205. doi: 10.1109/TPAMI.2012.191
- [5]. International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8, Issue-3S, February 2019 Face Recognition Based Attendance System Nandhini R, Duraimurugan N, S.P.Chokkalingam.
- [6]. Varadharajan, E., Dharani, R., Jeevitha, S., Kavinmathi, B., & Hemalatha, S. (2016). Automatic attendance management system using face detection. In *Online international conference on green engineering and technologies (IC-GET)*.
- [7]. Abdullah, M., Wazzan, M., & Bo-saeed, S. (2012, March). Optimizing face recognition using PCA. *International Journal of Artificial Intelligence & Applications*, 3(2), 23-31.
- [8]. S. Rani K, T.C. Sharma: Face Recognition office security system using lab view 8.6, *IJECIERD*, Volume 3, pp: 195-200 (2013).
- [9]. Adriansyah, A., & Dani, A. W. (2014). Design of Small Smart Home System Based on Arduino. *Electrical Power, Electronics, Communications, Controls, and Informatics Seminar (EECCIS)* (pp. 121-125). Malang, Indonesia: IEEE.
- [10]. Fawaz, A. (2019). A smart classroom of wireless sensor networks for students time attendance system. In *IEEE integrated STEM education conference (ISEC)*.
- [11]. [Phillips and Moon, 1997] Phillips, P. J. and Moon, H., “Comparison of Projection-Based Face Recognition Algorithms,” *Proceedings of 1997 International Conference on Systems, Man, and Cybernetics*, Orlando, Florida, October 12–15, 1997.
- [12]. [Turk and Pentland, 1989] Turk, M. and Pentland, A., “Face Processing: Models for Recognition,” *SPIE Vol. 1192 Intelligent Robots and Computer Vision VIII: Algorithms and Techniques* (1989).
- [13]. Mark Nixon "Eye Spacing Measurement for Facial Recognition", *Proc. SPIE 0575, Applications of Digital Image Processing VIII*, (19 December 1985);

# APPENDICES

## Appendix – A

Android Studio	Android studio is official integrated development environment (IDE) for Google's android operating system, built on JetBrains's IntelliJ IDEA software and designed specifically for android development. It is available for download on windows. It is the replacement for the Eclipse android development tools (ADT) as primary IDE for native android application development
App	A type of mobile application, also known as a computer program designed to run on smartphones
Android	Operating System developed by Google Inc. specifically designed for smartphones and tablets
GUI	Graphic User Interface
Android App	An Android based application developed using Java and Python exclusively for Android Devices
Windows	Operating System
Scripting	A scripting language or script language is a programming language that supports the writing of scripts, programs written for a special runtime environment that can interpret and automate the execution of tasks which could alternatively be executed one-by-one by a human operator.
Graphics	Graphics are visual presentations of app
UX	User experience for using a product.
User	User is the one who use the app to read listen and watch the stories of prophets



System	A system is a set of interacting or interdependent components forming an integrated whole or a set of elements and relationships which are different from Relationships of the set.
Software	A collection of data or instructions that tells a computer to do certain tasks
SRS	SRS software requirement specification
Use Case	List of actions typically defining the interactions between a role and a system to achieve a goal
Sequence Diagram	A Sequence Diagram (SD) is an interaction diagram that shows how object operate with one another and in what order.
Component Diagram	Components are wired together by using an assembly connector to connect the required interface of one component with the provided interface of another component. This illustrates the service consumer - service provider relationship between the two components.
Deployment Diagram	A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.

## Appendix – B

Activity Diagram	An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modeling.
Use Case Diagram	A Use Case Diagram illustrate the usage of each type of users of the system. Which include admin processing on the right side and other users on the left side.
State Diagram	State machine diagram is a behaviour diagram which shows discrete behaviour of a part of designed system through finite state transitions.
Test Case	A test case in software engineering is a set of conditions or variables under which a tester will determine whether an application or software system is working correctly or not.
Python	Python is an <u>interpreted, high-level, general-purpose programming language</u> mostly used in machine learning
Program	A computer program is a collection of instructions that performs a specific task when executed by a computer.
Sublime Text	Sublime Text is a <u>proprietary cross-platform source code editor</u> with a <u>Python application programming interface</u> (API).
Java	Java is a general-purpose programming language that is class-based, object-oriented (although not a pure OO language, as it contains primitive types), and designed to have as few implementation dependencies as possible.
Smartphone	A mobile phone that performs many of the functions of a computer, typically having a touchscreen interface, Internet access, and an operating system capable of running downloaded apps.

**The University of Lahore, Islamabad Campus**

**Department of CS and IT**

**Plagiarism Certificate**

**Face Recognition Attendance Management System**

**Using Firebase ML-Kit with Android**

**By**

**Hafiz Muhammad Umer (70063633)**

**Basit Shahzad Tatari (70064897)**

**Asif Sultan Bhatti (70066389)**

Submission Date: 14-Jul-2020 07:17 PM (UTC+0500)

File name: Plagiarism\_Report\_3.pdf (7.16M)

Word Count: 13379

Plagiarism checked by: Dr. Raja Habib\_\_\_\_\_

Plagiarism Result: 2%

## Plagiarism Report

### ORIGINALITY REPORT

2%

SIMILARITY INDEX

1%

INTERNET SOURCES

0%

PUBLICATIONS

1%

STUDENT PAPERS

### PRIMARY SOURCES

1

aakash-tech-support-  
documentation.readthedocs.io

Internet Source

<1%

2

Submitted to CVC Nigeria Consortium

Student Paper

<1%

3

Ghassan Aouad, Yusuf Arayici. "Requirements  
Engineering for Computer Integrated  
Environments in Construction", Wiley, 2010

Publication

<1%

4

Submitted to University of Wolverhampton

Student Paper

<1%

5

Submitted to Ghana Technology University  
College

Student Paper

<1%

6

B Prabhavathi, V Tanuja, V Madhu  
Viswanatham, M Rajashekhara Babu. "A smart  
technique for attendance system to recognize  
faces through parallelism", IOP Conference  
Series: Materials Science and Engineering,

<1%