Student Online Attendance

A Major Project Report Submitted in Partial Fulfillment of the Requirement for the Award of Degree of

BACHELOR OF ENGINEERING

(Computer Science & Engineering Department)

To

HIMACHAL PRADESH TECHNICAL UNIVERSITY, HAMIRPUR

Submitted by

Chander Bhanu

(02BTD5050021)

Under the Supervision of

Er Anurag Sharma (Asst. Prof. CSE),
Er Shivani Thakur (Asst. Prof. CSE),
Er Rahul Pal Singh (Asst. Prof. CSE)

COMPUTER SCIENCE & ENGINEERING DEPARTMENT

ATAL BIHARI VAJPAYEE GOVERNMENT INSTITUTE OF ENGINEERING & TECHNOLOGY, PRAGATINAGAR DISTT. SHIMLA (H.P)-171202 INDIA

2015-2019

DECLARATION

I, student of Bachelor of Technology (B. Tech.) Computer Science & Engineering Department, Atal Bihari Vajpayee Government Institute of Engineering and Technology, Pragati Nagar Shimla, having **Roll No – 02BTD5050021,** session 2015-2019, hereby declare that the Project entitled "**Student Online Attendance**", submitted to the Computer Science & Engineering Department as a partial fulfilment of the requirements for the award of the degree of Bachelor of Technology, has been carried out by us under the guidance of Er. Anurag Sharma, Er. Shivani Thakur and Er. Rahul Pal Singh. The Project is an original piece of work and has not formed the basis for the award of any other degree of any other college.

Dated:

Chander Bhanu
(02BTD5050021)

Department of Computer Science & Engineering
Atal Bihari Vajpayee Government Institute of Engineering and Technology, Pragatinagar Distt. Shimla H.P-171202

CERTIFICATE

THIS IS TO CERTIFY THAT THE PROJECT WORK ENTITLED "**STUDENT ONLINE ATTENDANCE**" HAS BEEN CARRIED OUT BY CHANDER BHANU, HAVING EXAMINATION ROLL NUMBER: **02BTD5050021**, BONA-FIDE STUDENT OF BACHELOR OF TECHNOLOGY (B.TECH.), SESSION 2015-2019, COMPUTER SCIENCE & ENGINEERING DEPARTMENT, ATAL BIHARI VAJPAYEE GOVERNMENT INSTITUTE OF ENGINEERING AND TECHNOLOGY, PRAGATINAGAR SHIMLA. THE PROJECT IS A RECORD OF THE WORK ACCOMPLISHED DURING THE B. TECH. AS A PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF DEGREE OF B.TECH. (COMPUTER SCIENCE & ENGINEERING) UNDER MY GUIDANCE AND SUPERVISION. THE WORK IS WORTHY OF ACCEPTANCE. I WISH HIM/HER ALL THE BEST IN FUTURE.

Dated:

Er. Ravi Kumar

Assistant Professor
Computer Science & Engineering

Head of the Department
Computer Science & Engineering Department,
Atal Bihari Vajpayee Government Institute of Engineering and Technology, Pragatinagar Distt. Shimla H.P-171202

ACKNOWLEGEMENT

I wish to express my deep sense of indebtedness and sincerest gratitude to **Er. Anurag Sharma, Er. Shivani Thakur and Er. Rahul Pal Singh** for their invaluable guidance and constructive criticism throughout this dissertation. They had displayed unique tolerance and understanding at every step of progress and encouraged me. We deem it our privilege to have carried out our Dissertation work under their able guidance. As a Final Personal Note, we are grateful to our parents, who are inspirational to us in their understanding, patience and constant encouragement.

Chander Bhanu

(02BTD5050021)

Abstract

"**STUDENT ONLINE ATTENDANCE**" is application developed for taking the attendance of the student on the basis of location. First the app will check trough the student smartphone whether he/she is present within the perimeter of the classroom if he/she is present then the attendance will be sent as a notification to the teacher application. Each Student will be given with a separate username and password, where username will be referred as a Roll no... The message in the notification will be shown in the vertical alignment. This will also help minimizing the attendance proxy. In most educational institutions the attendance is taken manually. It is not only time consuming, but it is also unsecure and unreliable and attendance can be lost. To overcome these problems. I have developed a better system which is Android Application.

LIST OF FIGURES

List of Abbreviation

Symbol.	Abbreviation
Auth	Firebase Authentication
API	Application Programming Interface
Regex	Regular Expressions
FCM	Firebase Cloud Messaging
XML	eXtensible Markup Language
JVM	Java Virtual Machine
JDK	Java Development Kit
JRE	Java Runtime Environment
SMS	Short Message Service
SVP	Speed Variation Period
SSD	Speed Standard Deviation
OBD	On Board Unit
SOA	STUDENT ONLINE ATTENDANCE

Contents

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

The Student Online Attendance application is developed to help lecturers to take the attendance of students using smartphone. In the traditional way attendance is taken on the paper or registers by marking with the pen. But, In the traditional way more time is wasted on taking the attendance by calling student Roll Number or Student's name which results in the wastage of time. Some of the student's helps their friends to mark the attendance of a person who is absent in a lecture or class. This is also one of the major problems in traditional attendance marking system that lecturer has to face because in a class of more than 60 student's teacher cannot remember the face of each and every student. But with this application the student attendance will be marked in an effective way. [1]

Today it is imperative to use the technology for our ease of living in a more efficient way. like we can use the technology like smartphone for marking student attendance easily. This method is eco-friendlier.

Nowadays everybody is using high quality and android supported mobiles. So, in this project we are using Android for marking student attendance. Attendance Management System is software developed for daily attendance of students. Previously, the college relied heavily on paper records. This paper focuses on preventing information in an easy way and intelligible manner which reduces paper and time. The project makes use of database in order to keep a record of attendance and is used while generating a report for individual student. The system is fully controlled by administrator about the staff's profile, student information student attendance. The administrator can add new student, add new staff, view staff and student and view the student attendance report. The faculties are able to directly access all aspects of the student's progress through a secure, online interface embedded in the college website. After the class has been finished, the lecturer can view the student's attendance that has been saved in the faculty server. This system will also help in evaluating attendance eligibility criteria of a student. The faculty can also send messages to parents or guardians about the student performance and attendance by short listing the students. Hence, here communication is made easier with parents and faculty. The messages can be sent through

SMS to the mobile number given in details of the students in the file. The purpose of developing this software is to computerize the tradition way of taking attendance. Another purpose of software is to generate the report automatically at the end of the session. Moreover, graphical user interface is provided in the proposed system, which provides user to deal with system very easily.[1]

The software is developed for the smartphones, because mobile devices is more mobile than other Systems.

The technology used by the project is Android. Android is a Linux based operating used designed primarily for touch screen mobile devices such as smart phones and tablet computers. The operating system has developed a lot in last 15 years starting from black and white phones to recent smart phones or mini computers. One of the most widely used mobile OS these days is Android. The android is an open source operating system means that it's free and any one can use it.

Android applications are written using the Software Development Kit (SDK) and often, the Java programming language. Java may be combined with C/C++, altogether with a choice of non-default runtimes that allow better C++ support. The Go programming language is also supported although with a limited set of application programming interfaces (API). In May 2017, Google announced support for Android app development in the Kotlin Programming language.

The version history of the Android mobile operating system began with the public release of the Android beta in November 5, 2007. The first commercial version, Android 1.0 was released.

1.2 MOTIVATION

In most educational institutions the attendance is taken manually. It is not only time consuming, but it is also unsecure and unreliable and attendance can be lost. Some institutions are using punch card for attendance while this will be difficult for teachers to keep track of the large number of students because by using punch card, a student can help the other students or his/her friend to punch their card even the other student may be absent or come late in class, so it is not reliable. To overcome these problems.

I have developed a better system which is Android Application; It contains 2 modules or 2 application where one is developed for student through where student can login and send the his/her location to the backend. And teacher receives some details in the app with roll no along with some other information. It will be less Time consuming.

The motivation behind making the app is to allow simplicity in taking attendance to the teacher and allow students to view their daily attendance date wise. The teacher can thus take attendance without going through the hassle of paperwork which have its own disadvantages.

Another aspect is to allow teachers to send notifications to students in form of text and images and allow students to get the important information. So, two main points which motivates us to make this application can be given as:

- Taking attendance records for each student is to be calculated as to be necessary. But using this application makes it easy to find out the attendance records of the students which makes it a useful feature.
- Universality of mobile phones motivates to make better use of this technology[1].

CHAPTER 2
LITERATURE SURVEY

2.1 LITERATURE SURVEY

The role of attendance in academic environment:

Attendance was always an important part of final grade of students from the early days of school's existence. But what is the reason for taking attendance? Many researchers conducted researches to find out the relation between students' attendance and result.

Yao and Chiang[3] conducted a research to find out the relationship between students' class attendance and their overall grades between students who participated in computer science classes. The analysis of their data showed that students who has more class attendance, have higher grades. Also, they indicated that there is a strong correlation between absence from the first day of class and overall result. Broucek and Bass indicated that the GPA is significantly correlated with attendance.

Credé, Roch, and Kleczka conducted a research to resolve the debate about the importance of attendance and explore the relationship between attendance and students' grade and characteristics. They found out that there is a strong correlation between attendance and both individual grade and college CGPA. Also, they indicated that the students' characteristics have a small effect on attendance.

Attendance systems:

Since the computer enters the educational environment, many researchers performed different researches to simplify the attendance task based on the technology that was available. Mar and Lan caster designed and develop based application to save students' attendance data and generate reports. In their system, administration should enter the students' information and groups, and then tutors should print the class list names and tick who is absent and who is not and finally enter the data to the system. They just focused on developing a system to generate attendance report in their research and did not try to facilitate the process of entering data. Kizildag[3] et al. used barcode to keep track of students 'attendance in seminars. They printed the students' ID on their ID card and scan those by using a barcode reader. Nawaz, Assad, and Khalil[3] used

image processing technique to read the attendance list data and based on that data generate the overall attendance of students. In order to implement this system, they designed a standard attendance list and specified a special not repeated number to each student. Lecturers should know that which number belongs to which student and also, he should tick the list properly and obey some rules. The mentioned systems tried to make the attendance process easy, but in order to use these, still lecturers shouldspend their time to do that. In the following author reviewed some attendance systems which are commercially in the market and many organizations use these, but they have some disadvantages too.

Finger print:

According to Jain, Hong, and Kulkarni,[3] "A fingerprint is the pattern of ridges and furrows on the surface of a fingerprint. The uniqueness of a fingerprint is exclusively determined by the local ridge characteristics and their spatial relationships.". Fingerprint is one of the biometric systems that used widely. There are other biometric systems such as: Iris, Facial, Voice, DNA, Hand, Gait and Signature verification. Nawaz et al. developed an attendance system based on fingerprint. Their system consisted of fingerprint sensor to capture the fingerprints, LCD to show the submission of attendance and a server to save all data. In order to use this system student should obey some roles that one of them is to scan finger correctly on the fingerprint censorship, Khan, Munir, and Baloch introduced a fingerprint attendance system over the network. Their system read the fingerprint by a sensor and generates a fingerprint template on the client, then the client check that with templates that saved in the server over the network and if it finds a match, attendance will be marked. They also generate a report at the end of the attendance process and email it to the lecturer, furthermore; If the number of absences of a student exceeds the limit, an SMS will be sent to student's parent's hand phone. Zang and Designed and implemented a wireless fingerprint management system. In their research they designed a mobile devise that can scan fingerprint and check the template with saved templates on itself and save all data and finally transfer the data to the server in order to preparing the reports by using a wireless communication module.

RFID

According to Hunt, Puglia, and Puglia, "RFID is anacronym for radio frequency identification, which is a wireless communication technology that is used to uniquely identify tagged objects or people.". A RFID system includes two parts: readers and tags. Readers can be stationary or mobile and tags are attached to objects. When a tag is within the range of

reader, reader can read tag's information. Each tag can save up to 2KB information on its microchip. Kassem et al. developed a prototype of a RFID[3] system for monitoring and identifying attendance at Notre Dame University and they believed that their system is reliable, time saving and easy control. Silva, Filipe, and Pereira developed an attendance system based on RFID prototype which consists of two parts: first part includes database to save the data and a web service server and second part includes RFID readers which installed in the classrooms and these two parts connected by the LAN. They had a concern about student's privacy in their research. They were worried about the issue that they can know where the student is in the university.

.

CHAPTER 3

PROPOSED PROJECT WORK

3.1 PROBLEM DEFINITION

Attendance Management System is software developed for daily student attendance in schools, colleges and institutes. It facilitates to access the attendance information of a particular student in a particular class. This system will also help in evaluating attendance eligibility criteria of a student. By just a click on the app, the system will be able to produce the students' attendance report thus reducing the need for manual labour which is prone to human errors and time consuming.

This application is built for automating the processing of attendance. It also enhances the speed of performing attendance task easily.

The student and staff have unique user login id and password available. The student can only view the attendance record on weekly, monthly, and whole semester basis. The staff can view as well as modify the attendance record. Printing facility for attendance record is available for both students and staff.

With respect to managing the attendance records of student records it has other features included. One includes “Status” feature. This allow the teacher to send notifications regarding any important information student needed to know.

This can be sent in the form of text or in the form of image. The image may include any important notification sent on office orders.

The teacher first registers in the app and then can add the subjects he teaches. Based on number of students a new activity is generated which allow mark the attendance of the student.

The attendance marked can be viewed by teacher in the table format. The names of student can be updated by the teacher in the other activity.

The subject can be deleted by the teacher if wrong subject has been added or wrong details added.

The student can then login with his email and password to get access to the attendance marked by different teachers on basis of departments of college. The student can also view the notifications sent by the teachers.

Apart from managing the attendance records of student records it have other features included. One includes “Status” feature. This allow the teacher to send notifications regarding any important information student needed to know.

These notifications may be related to work related to class work or any other information related to tests. By having attendance on mobile it becomes easy for teachers to easily take attendance without any problem of paperwork which can be conflicting. These types of problems are solved by this application developed in Android which solve such problem. This attendance can be viewed in easy layout giving details regarding attendance which include students total present and total absent details of students as well as percentage details of students.

Some major problem at present state are:

- No software is present at all.
- Time is consumed in taking the attendance.
- Existing system requires lot of paper work. Loss of even a single register/record led to difficult situation No direct way to get the contact information of faculty members.
- Student marking the proxy of fellow students.

3.2 OBJECTIVES

The objective of this project is to include the features that a student might need in his day to day life in college.

The main objective of this project is to computerize the manual system & reduce the time consumption. In other words, we can say that our project has the following objectives: -

1. Make the attendance system computerized.
2. Reduce time consumption.
3. Reduce error scope.
4. Paper work is reduced as we need not to maintain registers for future references.
5. To avoid wastage of papers by taking attendance through app.
6. Simplicity in the taking of attendance.

With these the aim is to reduce the operational time, and increase accuracy, reliability and operational efficiency.

3.3 METHODOLOGY

The students will be able to see all the updates that the department will publish in real time. And the Administrator will be maintaining daily updates. Administrator must be an authorized user. He can publish new updates using a dedicated application. The students will also be served with a dedicated application.[4]

This system also uses access points' signal strength data to determine the indoor position but in a different way. Each access point has two main properties which are: media access control (MAC) address which is a world unique address and received signal strength indicator (RSSI) which is a measurement of the power present in a received radio signal. RSSI is an indication of the power level being received by the device antenna and that means the higher the RSSI number, the stronger the signal

Through analysis of the researched methodologies it has become apparent that either Prototyping or the Waterfall model with iterative feedback is best suited for the development. While prototyping help to produce a solution quickly and involve the user during the physical development of the application, the waterfall model clearly sets out a stage to establish user requirements, which I believe will be critical to the success of my project.

For this reason, the waterfall model with iterative feedback will be used to structure the development of the application. Iterative feedback will allow for stages of the development to be returned to should a change in user requirements to be needed. The development will also try to exploit the advantages of prototyping by integrating it within the implementation stage of the waterfall model to receive feedback from the user on the design and main functionality of the system. The section logically illustrates the mechanism of the proposed system structure together with each module that constructs the overall system architecture. The proposed system, called **Attendance management system** consists of sub modules:

Teacher Login: to be used by the teacher to register and login to get access to homepage. Then mark attendance of the students by adding subjects and student details. It also allows to send the notification to the students.

Student Login: to be used by the students to get their attendance records marked by teachers. It also includes to view the status sent by the teachers.

These two are explained below:

3.3.1 TEACHER LOGIN

This application can only be used by the faculty members. Here the main purpose of this application is to add and modify the data into the cloud database. The data included attendance marked by teachers of their respective subjects. Only the Teacher login is given the privilege to add or modify the database content. Thus, the admin can post the information into the database and can retrieve the data. But the student can only retrieve the data. He cannot modify the data. Thus, this adds to a bit security to the overall system. In the **Attendance management system**, teacher can manage the attendance of the students in the table format.

Teacher can also send status to the students which can be termed as notifications. In addition to the above features it allows to change password of the user by re authenticating the password. It also allows to view the profile of the teacher and edit the personal details added by teacher. Logout feature allow to log out from the application to the Home screen.

Teacher can then mark the attendance of the added students and this attendance can be viewed in a table-view. Apart from viewing attendance, teacher can change the student names and also can edit student subject details.

Also, there is status activity for teacher from where there is option to upload an image and an edit text where teacher can type the message which is to be sent to the server. From there data can be get on Student login.

Also, the authorization system has been set up for this app as well, so that even if a student downloads the app, he will not be able to login into the app trying to intrude with the system. So, for authorization or authentication, we have used the *Firebase Authorization* feature of the cloud database. Thus, only the authorized user can access the application in order to add or modify the content of the database.

From the flow chart we can see that, the user is first provided with the login screen, then the user has to enter the email or username and password in order to user the application. Then the user is authenticated with the firebase authentication mechanism. If the user is a valid user, then he is presented with the home screen from where he can access all the features that are provided within the application. If the user is not a valid user, then the user will be

redirected back to login screen.

3.3.2 STUDENT LOGIN

The Attendance management system student application is the one which the students are going to use. This app is created only for the student use although the faculty members can use it well. Here the main purpose of this application is to retrieve the data from the cloud database. That is, the students can only retrieve the data, they cannot add or modify the data contents except their basic information like their branch, name, or phone no.

Student can get all the needed information regarding his attendance from the app itself without having to worry about any of problems involved in seeing the notice board.

Student can also get the notices which are to be sent by the teachers. These notices are viewed by students by clicking on view status activity.

These statuses can be sent by a teacher which can contain important information related to student. During registration activity while entering basic details student is required to specify the department in which he is studying. This department may be cse, ece or others.

The department specified by student will help him in getting the attendance details of his department. The attendance details of all students will be shortlisted based on the department of the students.

On logging into the application student will see the activity in home screen which will have option of Check attendance and a check status option. On clicking on check attendance student will enter into other activity containing details regarding the complete attendance system.

On clicking on Check status student will enter into the facticity containing details regarding status sent by the teachers. This basic information is necessary for customizing the application data according to the user.

Here the data that is stored in the cloud database can be retrieved using the student application. For the application to work, the Internet is needed. The retrieval of the data from the database is very quick.

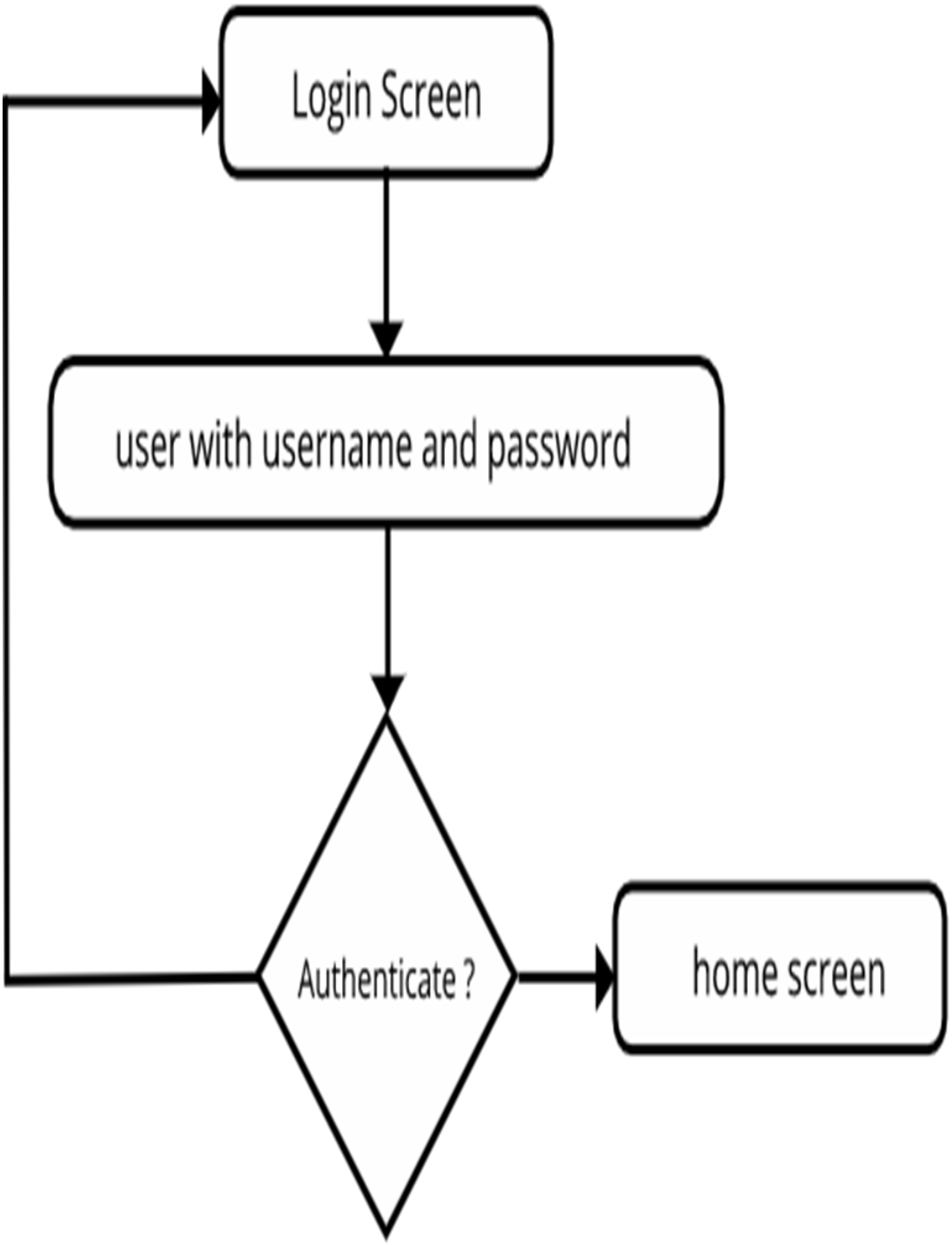

Figure 1 general working of the Student application

All it takes is 2-3 seconds to load the data into the application. The user authentication mechanism is also implemented in this application so that no outsider can access the information of our college.

The overall flow diagram of the application is shown in the figure 4. Here we can see that when the user opens the application, the user is first checked if he is the old user or the new user. By new user, we mean that the user is using this application for the first time. If the user is a new user, then the registration screen is shown, where the user is asked for the basic information. After entering the information, the user details are verified again with the database.

If the details entered are found to be unique, only then the user is registered that is the data is saved in the database. Else if the details entered are not unique that means the user is registered already and then user is provided with an alert and is redirected to the login screen.

In login screen, the user has to enter the email and password as login credentials. After that the user is verified and then in sent to the home screen. Now the user can access the provided features and if the user wants to log out, then he can do so by pressing the log out button.

Then the user will be redirected back to login screen. As shown in figure, where the user is asked for the basic information. After entering the information, the user details are verified again with the database. If the details entered are found to be unique, only then the user is registered that is the data is saved in the database.

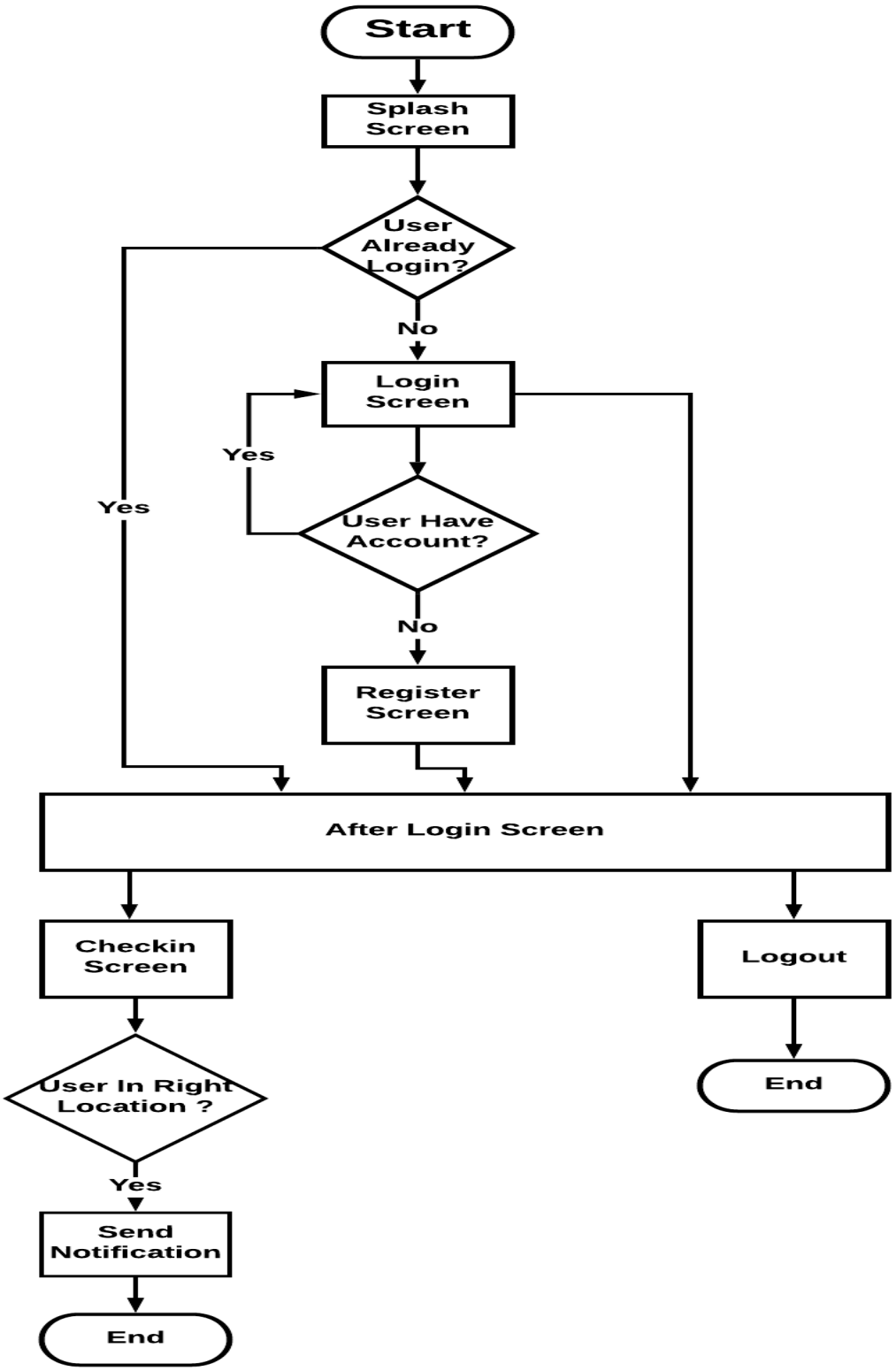

Figure.2 Student Online Attendance Flow Diagram

3.4 EXPECTED OUTCOMES

After the implementation of this project, we will have an application that will provide the user to quicker access of the data from the cloud, which will increase the reliability and the overall user experience.

After the implementation of project. The final application will have following features:

• Easy to use GUI.

• Teachers will be able to mark attendance easily or attendance will be able automatically transferred to the teacher app.

• The false attendance being awarded to a student by means of his /her friends or batchmates (Proxy) can be reduced.

• The last and most important of all i.e. it will be able to reduce the time for marking the attendance.

3.5 HARDWARE AND SOFTWARE REQUIREMETS

3.5.1 HARDWARE REQUIREMENTS

Following are the specifications for the machine on which the project will run successfully:[5]

For Development:

64-bit distribution capable of running 32-bit applications.

1.8GHz processor or Higher

4 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator.

2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system images.

For Deployment:

1GHz processor or Higher

512 MB RAM minimum, 1 GB RAM recommended

25 MB of available disk space minimum

Android Device with minimum Android version 5.1

3.5.2 SOFTWARE REQUIREMENTS

Following are the software requirements in order to develop the project:

This software is required during the development of the application of the android.
These can be explained be

Java

Java is a general purpose and the most popular object-oriented programming language. Java was developed by James Gosling and his colleagues at Sun Microsystems in the early 1990‟s. Due to its simplicity and easy to learn and advanced features, we opted this language for our project. This language supports many interesting features that make it an ideal language for software development.

Android Studio

Android Studio is the official integrated development environment (IDE) for Google 's Android operating system, built on Jet Brains ' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, mac OS and Linux based operating systems.

XML

Android layouts are written in extensible Markup Language, also known as XML. Much like HTML (or Hypertext Markup Language), XML is also a Markup language. It was created as a standard way to encode data in internet-based applications. However, unlike HTML, XML is case-sensitive, requires each tag is closed properly, and preserves whitespace. XML is commonly used as a data format on the Internet. If you want to access data from the Internet, chances are that the data will be in the form of XML.

Android SDK

The Android SDK (software development kit) is a set of development tools used to develop applications for Android platform. The Android SDK includes the Required libraries, Debugger, An emulator, Relevant documentation for the, Android application program interfaces (APIs) and Sample source code.

Firebase (Back end)

Firebase is a mobile and web app development platform that provides developers with a plethora of tools and services to help them develop high-quality apps, grow their user base, and earn more profit. The Firebase Real-time Database is a cloud-hosted No Sql database that lets you store and sync between your users in real time. The cloud database is really one big JSON object that the developers can use to store the large amount of data. Firebase Authentication provides back end services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to your app.

Firebase Cloud Messaging (FCM) provides a reliable and battery-efficient connection between your server and devices that allows you to deliver and receive messages and notifications on iOS, Android, and the web at no cost.

Realtime Database is shipped with mobile and web SDKs, allowing you to build your apps without the need for servers.

When your users go offline, the Realtime Database SDKs use local cache on the device to serve and store changes. When the device comes online, the local data is automatically synchronized.

The Realtime Database can also integrate with Firebase Authentication to provide a simple and intuitive authentication.

Chapter 4

PROJECT IMPLEMENTATION

4.1 FRONT END & BACK END DETAILS

4.1.1 FRONT END: JAVA

Java is a general purpose and the most popular object-oriented programming language. Java was developed by James Gosling and his colleagues at Sun Micro systems in the early 1990‟s. Due to its simplicity and easy to learn and advanced features, we opted this language for our six months industrial training. This language supports many interesting features that make it an ideal language for software development. In addition to the object-oriented features as following: -

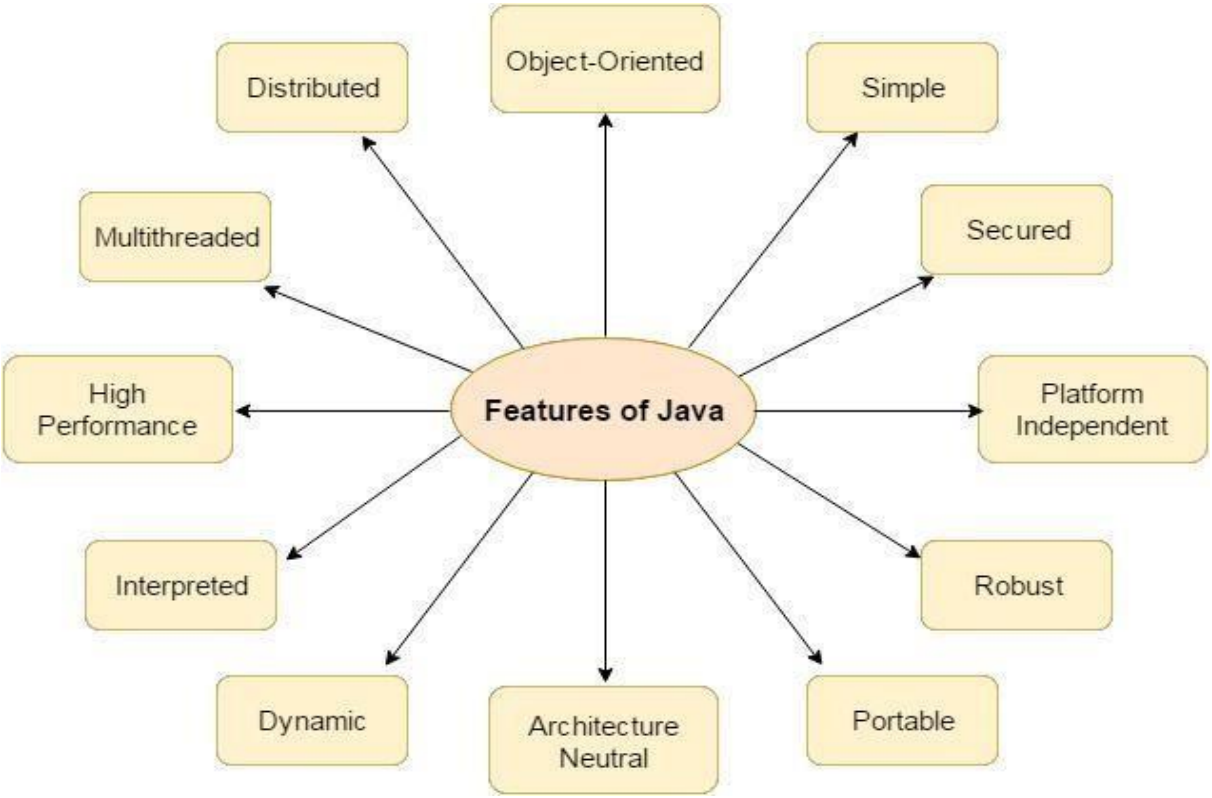

Fig 3. Features of Java

Java Features:

Simple: Java is a compact and simple language. Programs are easy to write and debug as it omits many clumsy, poorly understood and confusing features of other programming languages such as C++.

Object-Oriented: Java is an object-oriented language because programming in java is centred on creating objects; manipulating objects and making objects work together.

Distributed: Java is a distributed language which means that the programs can be designed to run on computer networks.

Robust: Java is designed for writing programs that are highly robust. By robust, we mean reliable.

Secure: As java is intended to be used in networked/distributed environments so it implements several security mechanisms to protect you against malicious code that might try to invade your file system.

Architectural Neutral: This means that the programs written on one platform can run on any other platform without having to rewrite or recompile them. It follows "Write-once-run anywhere" approach.

Portable: In Java, the size of the primitive data types is machine independent. These consistencies make java program portable among different platforms such as Windows, UNIX and Mac.

Interpreted: Java is such a language that is both compiled and interpreted. The two steps of compilation and interpretation allow extensive code checking and improved security.

High Performance: Java programs are complied with portable intermediate form known as byte codes, rather than to native machine level instructions and JVM executes java byte codes on any machine on which it is installed. This architecture means that java programs are faster.

Multi-threaded: Java is also a multi-threaded programming language. It allows you to write a program that can do many tasks simultaneously.

Dynamic: Java is designed to be dynamic. Classes are stored in separate files and are loaded into the Java Interpreter only when they are needed.

Robust: Java is designed for writing programs that are highly robust. By robust, we mean reliable.

Java Development Kit (JDK)

JDK is Kit which provides the environment to develop and execute(run) the Java program.
JDK is a kit (or package) which includes two things:
Development Tools (to provide an environment to develop your java programs).
JRE (to execute your java program).

Java developers are initially presented with two JDK tools, java and javac. Both are run from the command prompt. Java source files are simple text files saved with an extension of .java. After writing and saving Java source code, the javac compiler is invoked to create .class files. Once the .class files are created, the 'java' command can be used to run the java program.

For developers who wish to work in an integrated development environment (IDE), a JDK bundled with NetBeans can be downloaded from the Oracle website. Such IDEs speed up the development process by introducing point-and-click and drag-and-drop features for creating an application. There are different JDKs for various platforms. The supported platforms include Windows, Linux and Solaris. Mac users need a different software development kit, which includes adaptations of some tools found in the JDK. After writing and saving Java source code, the javac compiler is invoked to create .class files. Once the .class files are created, the 'java' command can be used to run the java programs. Java developers are initially presented with two JDK tools, java and javac. Both are run from the command prompt. Java source files are simple text files saved with an extension of .java. After writing and saving Java source code, the javac compiler is invoked to create .class files. Once the .class files are created, the 'java' command can be used to run the java program.

* For developers who wish to work in an integrated development environment (IDE), a JDK bundled with NetBeans can be downloaded from the Oracle website. Such IDEs speed up the development process by introducing point-and-click and drag-and-drop features for creating an application.

Java Runtime Environment (JRE)

JRE is an installation package which provides environment to only run (not develop) the java program (or application) onto your machine. JRE is only used by them who only wants to run the Java Programs i.e. end users of your system.

JRE 1.0 has evolved with a variety of class and package additions to the core libraries, and it has grown from a few hundred classes to several thousand in Java 2 Platform, Standard Edition (J2SE). Entirely new APIs have been introduced, and many of the original version 1.0 APIs have been deprecated.

Java Virtual Machine (JVM)

JVM is a very important part of both JDK and JRE because it is contained or inbuiltin both. Whatever Java program you run using JRE or JDK goes into JVM and JVM is responsible for executing the java program line by line hence it is also known as interpreter. The JVM is detailed by a specification that formally describes what is required in a JVM implementation. Having a specification ensures interoperability of Java programs across different implementations so that program authors using the Java Development Kit (JDK) need not worry about idiosyncrasies of the underlying hardware platform. The JVM reference implementation is developed by the OpenJDK project as open source code and includes a JIT compiler called Hotspot. The commercially supported Java releases available from Oracle Corporation are based on the OpenJDK runtime. Eclipse OpenJ9 is another open source JVM for OpenJDK.

Why JAVA?

Android runs on many different hardware platforms. We would need to compile and optimize our native code for each of these different platforms to see any real benefits.

There are a large number of developers already proficient in Java.

Java has huge open source support, with many libraries and tools available to make developers life easier.

Java protects us from many of the problems inherent in native code, like memory leaks, bad pointer usage, etc.

Java allows them to create sandbox applications, and create a better security model so that one bad App can't take down your entire OS.

4.1.2 FRONT END: XML

XML stands for Extensible Markup Language. XML is a markup language much like HTML used to describe data. XML tags are not predefined in XML. We must define our own Tags. Xml as itself is well readable both by human and machine. Also, it is scalable and simple to develop. In Android we use xml for designing our layouts because xml is lightweight language so it doesn't make our layout heavy.

In Android there are several xml files used for several different purposes. Below we define each and every one.

Layout XML files

Layout xml files are used to define the actual UI (User interface) of our application. It holds all the elements(views) or the tools that we want to use in our application. Like the TextView's, Button's and other UI elements.

Manifest XML file

This xml is used to define all the components of our application. It includes the names of our application packages, our Activities, receivers, services and the permissions that our application needs. For Example – Suppose we need to use internet in our app then we need to define Internet permission in this file.

Strings XML file

This xml file is used to replace the Hard-coded strings with a single string. We define all the strings in this xml file and then access them in our app (Activity or in Layout XML files) from this file. This file enhances the reusability of the code.

Styles XML file

This xml is used to define different styles and looks for the UI (User Interface) of application. We define our custom themes and styles in this file.

Drawable XML file

These are those xml files that are used to provide various graphics to the elements or views of application. When we need to create a custom UI we use drawable xml files. Suppose if we need to define a gradient color in the background of Button or any custom shape for a view then we create a Drawable xml file and set it in the background of View.

Color XML file

This file is used to define the color codes that we used in our app. We simply define the color's in this file and used them in our app from this file.

4.1.3 Project Folder Structure

Basic Android Project would have six directories such as: assets, bin, gen, libs, res, src. Also, there are some files in project root directory such as: AndroidManifest.xml, licenses, project. Properties and other files.

The most important for the developers are **"res"** and **"src"** directories.**res** directory contains all the current project resources such as: images, layouts, custom strings and other values. Images are stored in different directories depending on their size that application can automatically choose right image depending on the device specifications. Layouts are stored in the "layout" folder. Basically, layout file example would be an XML file which would specify elements and their position in current view. Also, it is possible to code custom strings and colors so the parser can display them in application. It is recommended approach to store them in values directory rather than hard code to the actual code or XML file. It would make easy further development at translating the application to other languages.

The other important directory is **src** directory. This directory would usually consist of Java files which are adding functionality to the application. Then developer would create classes as separated Java files. If the class is created in GUI ADT environment the tool would generate automatically the statement in "Android Manifest" file. If other programming environment is used, user must specify any new class activity by hard coding the Android Manifest file.

Android Manifest File usually would be placed to the root directory of the project and state required version of android, needed permission and all activities which are run within the application.

Android application uses XML layout for displaying its content. XML document would consist of several tags with given properties. The parent tag would state type of view which is the main for the document. Also, it is allowed to use several views inside of one main view.

Like any other XML tag, this tag is given several properties which will define its identification, style, onclick action, etc. Identification is one of the important part of the tag. By defining the "id" programmer can use it in the java code. Styles can be hard coded in the statement or linked to separate file which will specify style for this element. Action properties can call the event action which is defined in the code part of the project.

Java programming language is used in android development. All the activities are organized

in classes and triggered by certain events. Every android application has its starting activity. In the default project it would be main activity class.

Java folder will contain all the java source code (.java) files which we'll create during the application development, including JUnit test code. Whenever we create any new project / application, by default the class file MainActivity.java will create automatically under the package.

Layout folder will contain all XML layout files which we used to define the user Interface of our application.

AndroidManifest.xml: Every project in Android includes a manifest file, which is AndroidManifest.xml, stored in the root directory of its project hierarchy. The manifest file is an important part of our app because it defines the structure and metadata of our application, its components, and its requirements.

This file includes nodes for each of the Activities, Services, Content Providers and Broadcast Receiver that make the application and using Intent Filters and Permissions, determines how they co-ordinate with each other and other applications.

Mipmap: Mipmap folder contains the Image Asset file that can be used in Android Studio application. You can generate the following icon types like Launcher icons, Action bar and tab icons, and Notification icons.

Colors.xml: colors.xml file contains color resources of the Android application. Different color values are identified by a unique name that can be used in the Android application program.

Strings.xml: The strings.xml file contains string resources of the Android application. The different string value is identified by a unique name that can be used in the Android application program. This file also stores string array by using XML language.

Styles.xml: The styles.xml file contains resources of the theme style in the Android application. This file is written in XML language

4.1.4 BACK END: FIREBASE

Firebase is a technology that allows us to make web applications with no server-side programming so that development turns out to be quicker and easier. It supports the web, iOS, OS X, and Android clients. Applications using Firebase can just utilize and control information, without thinking about how information would be put away, and synchronized crosswise over different examples of the application in real time. With Firebase, we don't need to stress over-provisioning servers or building REST APIs with just a little bit of configuration; we can allow Firebase to make a chance to take every necessary step: storing data, verifying users, and implementing access rules.

Firebase provides lots of services. These can be divided into two groups: **Develop & Test** and **Grow & Engage.**

Fig 4. Firebase Services

Most widely used firebase services are described as follows:

Realtime Database:

The Firebase Realtime Database is a cloud-hosted NoSQL database that lets us store and sync between our users in Realtime. The Realtime Database is really just one big JSON object that the developers can manage in Realtime. With just a single API, the Firebase database provides our app with both the current value of the data and any updates to that data.

Realtime syncing makes it easy for our users to access their data from any device, be it web or mobile. Realtime Database also helps our users collaborate with one another.

Another amazing benefit of Realtime Database is that it ships with mobile and web SDKs, allowing us to build our apps without the need for servers.

When your users go offline, the Realtime Database SDKs use local cache on the device to serve and store changes. When the device comes online, the local data is automatically synchronized. The Realtime Database can also integrate with Firebase Authentication to provide a simple and intuitive authentication process.

Authentication:

Firebase Authentication provides back end services, easy-to-use SDKs, and ready- made UI libraries to authenticate users to our app. Normally, it would take us months to set up our own authentication system. And even after that, we would need to keep a dedicated team to maintain that system. But if we use Firebase, we can set up the entire system in under 10 lines of code that will handle everything for us, including complex operations like account merging.

We can authenticate our app's users through the following methods:

- Email & Password
- Phone numbers
- Google
- Facebook
- Twitter

And More.

Using Firebase Authentication makes building secure authentication systems easier, while also improving the sign-in and onboarding experience for end users.

Firebase Authentication is built by same people who created Google sign in, Smart Lock and Chrome Password Manager.

Firebase Storage:

Firebase Storage is a standalone solution for uploading user-generated content like images and videos from an iOS and Android device, as well as the Web.

Firebase Storage is designed specifically to scale your apps, provide security, and ensure network resiliency.

Firebase Storage uses a simple folder/file system to structure its data.

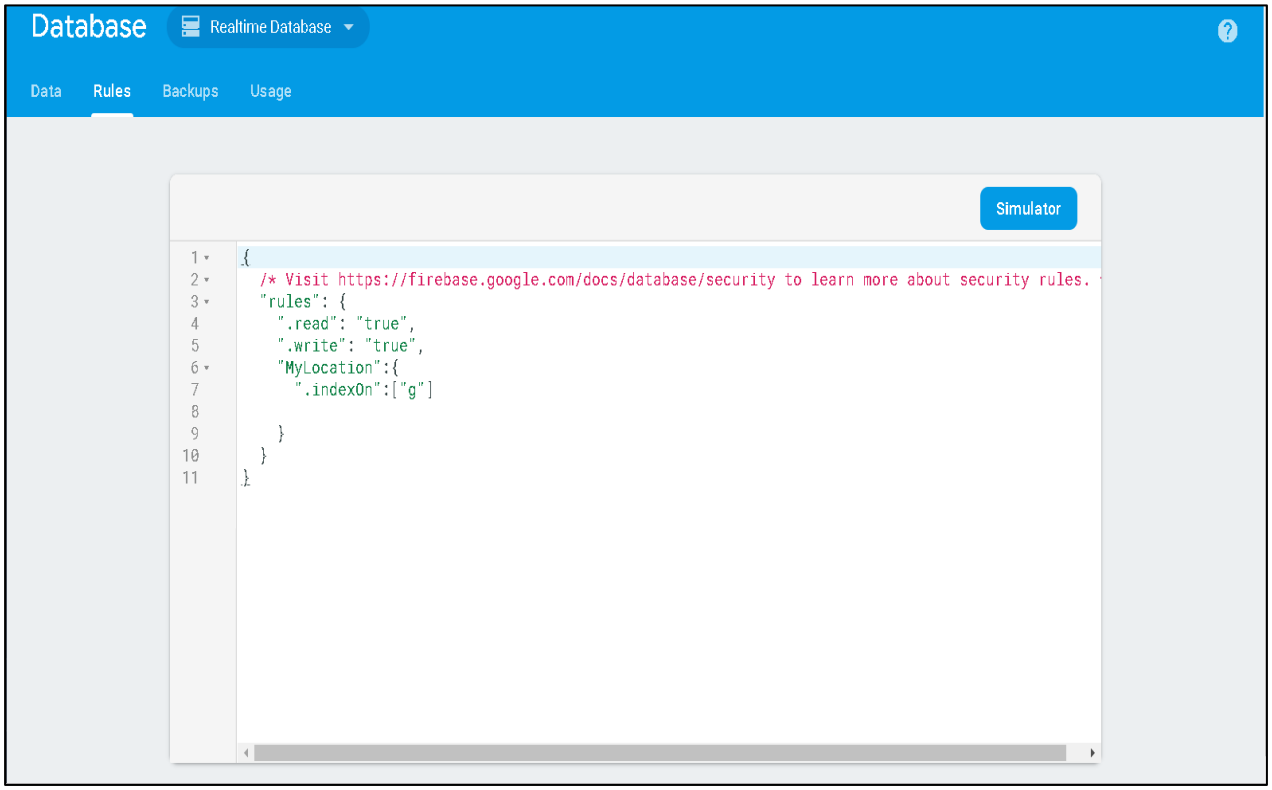

Fig 5. Firebase Storage rules

Firestore:

Cloud Firestore is a NoSQL document database that lets us easily store, sync, and query data for our mobile and web apps—at a global scale. Though this may sound like something similar to the Realtime Database, Firestore brings many new things to the platform that makes it into something completely different from Realtime Database.

Improved Querying and Data Structure:

Where Realtime Database stores data in the form of a giant JSON tree, Cloud Firestore takes a much more structured approach. Firestore keeps its data inside objects called documents. These documents consist of key-value pairs and can contain any kind of data, from strings to binary data to even objects that resemble JSON trees (Firestore calls it as maps). The documents, in turn, are grouped into collections.

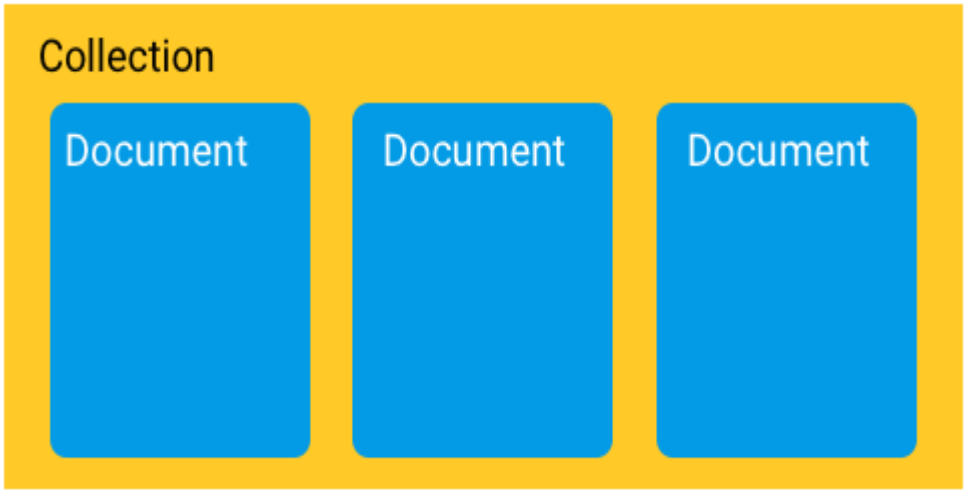

Fig 6. Firestore single collection

Firestore database can consist of multiple collections that can contain documents pointing towards sub-collections. These sub-collections can again contain documents that point to other sub-collections, and so on.

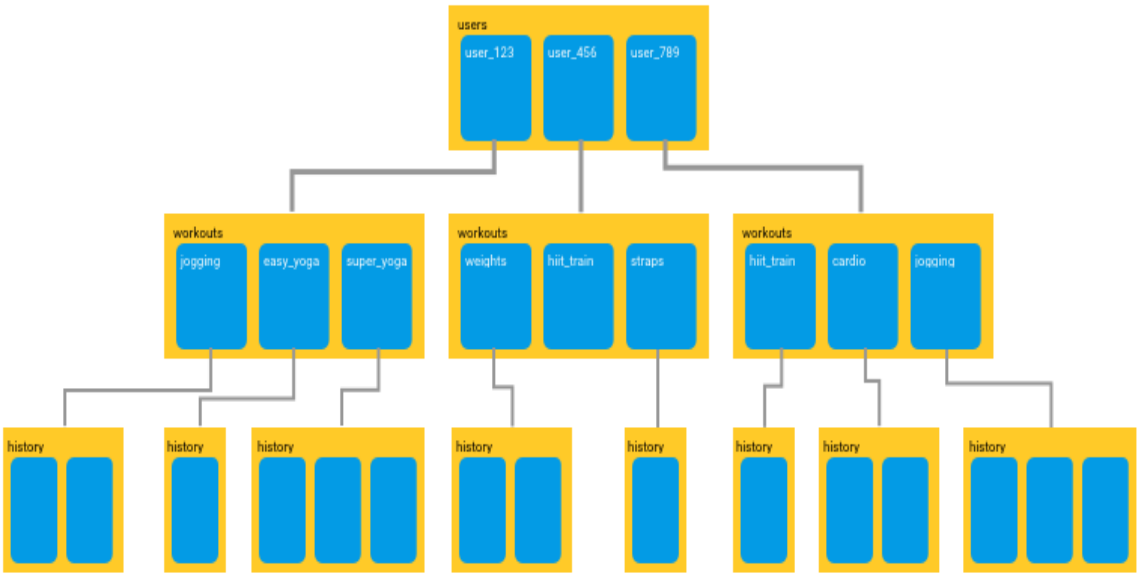

Fig 7. Firestore Multiple collections

You can build hierarchies to store related data and easily retrieve any data that you need using queries. All queries can scale with the size of your result set, so your app is ready to scale from its first day itself.

Firestore's queries are shallow. By this, I mean to say that in Firestore, you can simply fetch any document that you want without having to fetch all of the data that is contained in any of its linked sub-collections.

Query with Firestore:

Imagine that you have created a collection in Firestore that contains a list of Cities. So, before you can send out a query, you will have to store the database inside a variable.We can write a query to get into the location of data to use it later for our application.We can create several sub-collections which can be used for complex data structure.

Some applications require last number of collections to be managed on firebase,firestore can be useful in such cases.

Fig 8. Querying collection

Here, citiesRef is that variable that contains your collection of cities. Now, if you want to find a list of capital cities, you would write a query like this:

Fig 9. Querying collection with condition

But Cloud Firestore can make querying even easier! In some cases, Cloud Firestore can automatically search your database across multiple fields.

Better Scalability:

Though Firebase's Realtime Database is capable of scaling, things will start to get crazy when your app becomes really popular or if your database becomes really massive.

Cloud Firestore is based on Googles Cloud infrastructure. This allows it to scale much more easily and to a greater capacity than the Realtime Database.

Multi-Region Database:

In Firestore, your data is automatically copied to various regions. So, if one data center goes offline due to some unforeseen reason, you can be sure that your app's data is still safe somewhere else.

Firestore's multi-region database also provides strong consistency. Any changes to your data will be mirrored across every copy of your database.

Different Pricing Model:

The Realtime Database charges its users based on the amount of data that you have stored in the database.

Cloud Firestore also charges you for the same, but the cost is significantly lower than that of Realtime Database and instead of basing the cost on the amount of data stored, Firestore's pricing is driven by the number of reads/writes that you perform.

Firebase Cloud Messaging (FCM):

Firebase Cloud Messaging (FCM) provides a reliable and battery-efficient connection between your server and devices that allows us to deliver and receive messages and notifications on iOS, Android, and the web at no cost.

We can send notification messages (2KB limit) and data messages (4KB limit). Using FCM, we can easily target messages using predefined segments or create our own, using demographics and behavior. We can send messages to a group of devices that are subscribed to specific topics, or we can get as granular as a single device.

FCM can deliver messages instantly, or at a future time in the user's local time zone. We can send custom app data like setting priorities, sounds, and expiration dates, and also track custom conversion events.

Advantages of using Firebase in General:

- Super easy and quick to implement.
- No server-side configuration needed. No PHP Scripts and No Database Designs.
- Auto scaling built-in.
- Built-in support for authentication services like Facebook, Google and Twitter.
- Declarative Security Rules model allows us to enforce read/write privileges and data validation throughout the tree.
- Realtime update without using GCM.
- Robust APIs for Android.
- Cloud database access.
- Easy to manage a lot of data within database.
- Cloud Messaging Access.
- Firebase storage which allows to store image and videos on the server to use in applications.
- Crash Reporting feature.
- Dynamic Links access.
- Remote configuration.
- Firebase Test Labs.

4.2 WORKING OF THE PROJECT

The College Companion (Student) and College Companion (Admin) both are the sub apps of this project. Together they communicate with the cloud database. The admin app is responsible for adding or deleting the data from the database while the student app is responsible for only retrieval of the data from the database.

4.2.1 SMARTPHONE SIDE IMPLEMENTATION

Following are some pictures of the application implemented on the smart phone side:

Login/Registration Screen

The Login screen act as an authentication phase to authenticate the valid users of the product. Only the users who have already registered themselves can login. The login contains full validation check with the help of Textiles so that invalid user cannot by pass the screen. If in case the user is a new user, the he can register himself by click on Signup option.

The sign-up screen will allow user to enter details like name of email id as a roll number and password to be used for the student.

The login screen will simply take email address and the password which is obtained during registration process.

It also has forgot password feature in the login page which helps to reset the password of the user obtained during registration of the student or teacher, provided email address is correct. There is also a register text view which helps to redirect to the register page which helps to help student register.

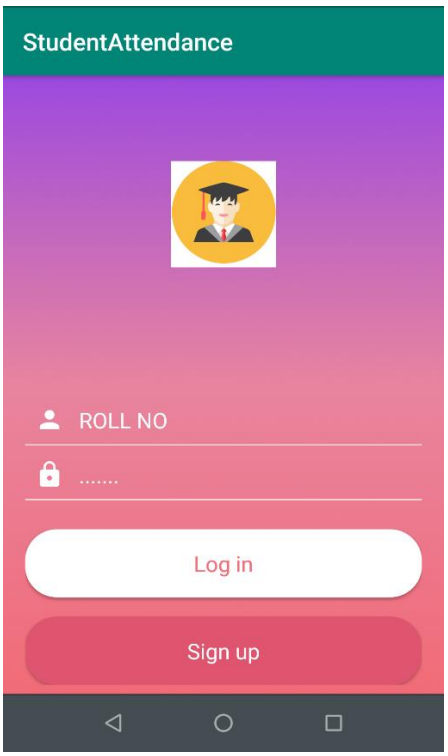

Fig 10. Log in Activity

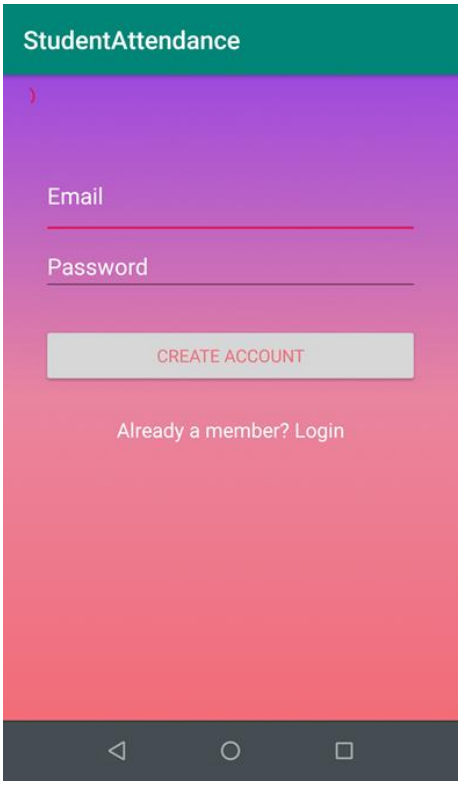

Fig. 11. Sign Up Activity

In background both these screens use the **Firebase Authentication** via email to create and authenticate new users. The access rules are custom so that only authenticated users can write and read from the database.

Home Screen

This is the main screen of the application. This contain all the options that the application is made to serve. There exist two options and the student can select any of them according to the needs. This screen will be displayed directly after the student has logged in successfully. With it the student needs not to login every time he opens the application.

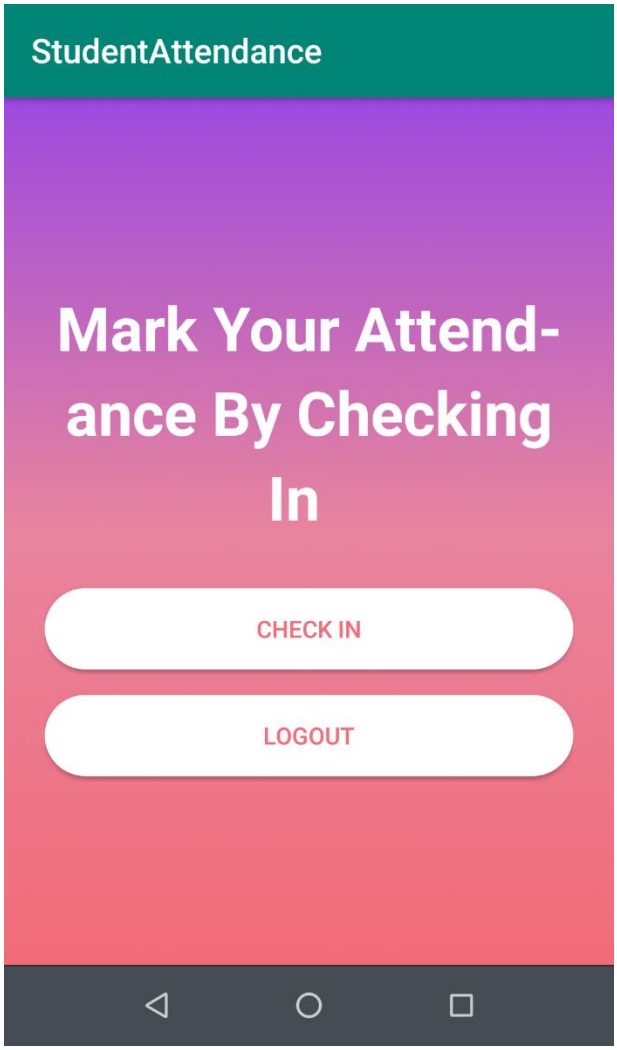

Fig 12. Main Home Screen

In background all the authentication is managed and the user specific information is retrieved form the cloud database.

Map Screen:

In this screen the user does not need to perform any action. The App will Automatically fetch the user Location and Apply the condition set for the geofence and if the is either inside the fence or outside the fence .If the user inside the fence then the Notification message will be sent to the teacher application from where teacher can mark it in the student management Application.

And If user is not present inside the fence then no notification will be sent to the teacher app which will be considered as the absent.

User can go to main Screen by simply pressing the return button. Marker in the figure represents the Location of the Student and Circle Represents the fence or perimeter set.

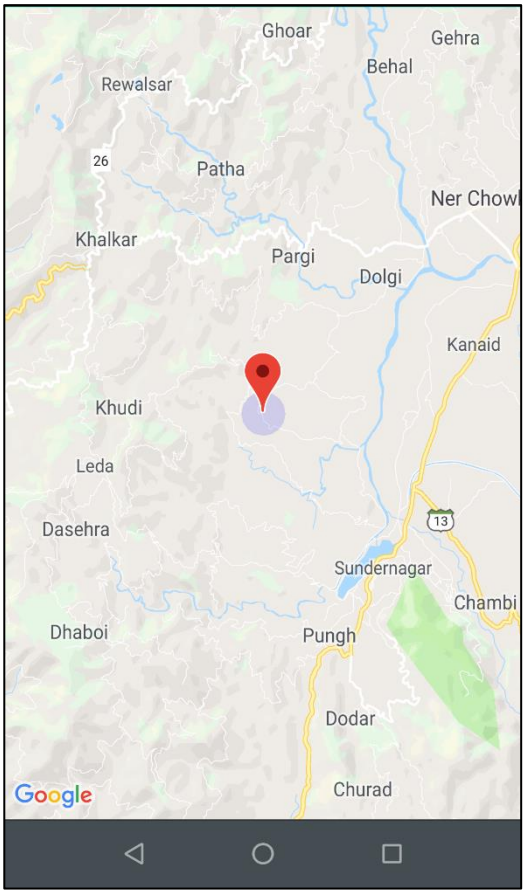

Fig 13. Map Screen

4.2.2 Student Online Attendance (Teacher)

This app is to be used by the faculty members of the college only. They can see who is present in the classroom from the notification or the messages shown in the application. mark the attendance and send some notification on the database. For the test version only one teacher will be given the access to use this application. Because

the application is in its early phase there are still lots of bugs and errors And till date the application is only designed for the one teacher .The application development will progress with time so in future it may be developed for the multiple teachers. But for now, only one teacher will be able to us.

4.2.3 SERVER-SIDE IMPLEMENTATION

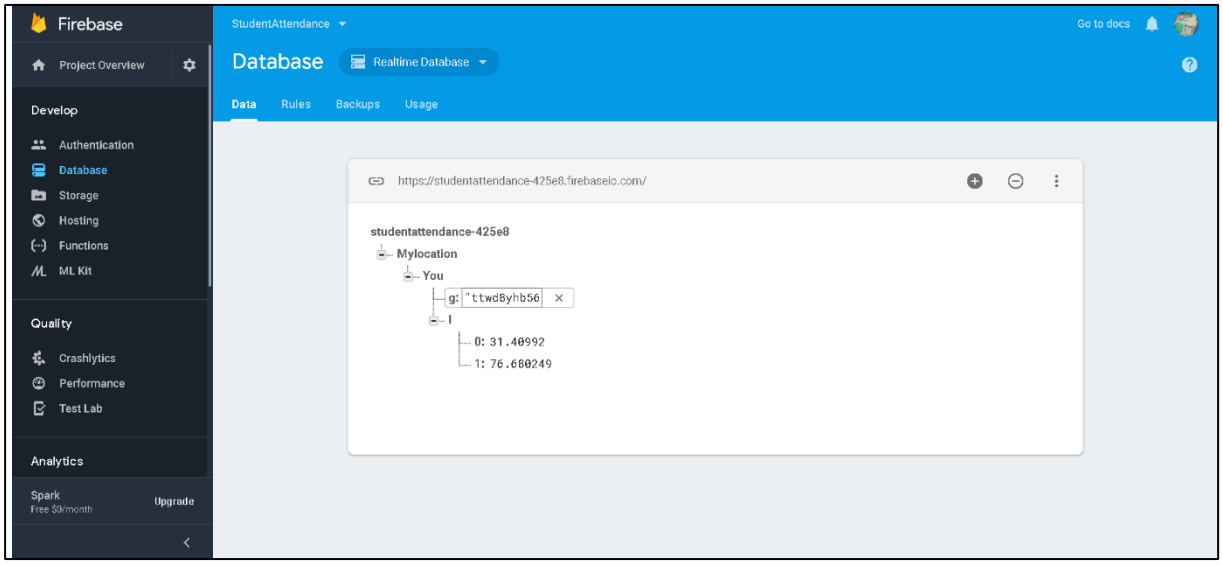

Figure 14. Backend implementation

The figure above shows the database that is being used at the backend. This is a Firebase's Realtime database. Store and sync data with our NoSQL cloud database. Data is synced across all clients in Realtime, and remains available when your app goes offline.

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in Realtime to every connected client. When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.

The Firebase Realtime Database lets you build rich, collaborative applications by allowing secure access to the database directly from client-side code. Data is persisted locally, and even while offline, Realtime events continue to fire, giving the end user a responsive experience. When the device regains connection, the Realtime Database synchronizes the local data changes with the remote updates that occurred while the client was offline, merging any conflicts automatically.

The Realtime Database provides a flexible, expression-based rules language, called Firebase Realtime Database Security Rules, to define how your data should be structured and when data can be read from or written to. When integrated with Firebase Authentication, developers can define who has access to what data, and how they can access it.

The Realtime Database is a NoSQL database and as such has different optimizations and functionality compared to a relational database. The Realtime Database API is designed to only allow operations that can be executed quickly. This enables you to build a great Realtime experience that can serve millions of users without compromising on responsiveness. Because of this, it is important to think about how users need to access your data and then structure it accordingly.

CHAPTER 5

CONCLUSION & FUTURE SCOPE

5.1 CONCLUSION

- We have tried to make our application as easy as everyone can easily understand application in their first use.
- No need to maintain several separate records and manual calculations.
- More Secure than traditional attendance system.
- Less time consuming. Saves a lot of time for the teacher to take attendance.
- In one sentence it is simple, handy, effective, creative and advance software that bring ends to the time-consuming problems to take attendance of each student manually.

5.2 FUTURE SCOPE

- As soon as the student log in the application in the class, they will be available for the attendance.
- In future instead of setting up a circular fence a rectangular shaped fence can be set up.
- If GPS can be improved in future then teacher can take attendance with more accuracy.
- An extension to this technology can be done using any of the face detection algorithms or using biometric scanner devices. This way the administrator has close to a hundred percent accurate information of the staff. For operation and handling a larger database, our system can be used to cope with students' data too.

CHAPTER 6
REFERENCES

[1] Android -based Smart Student Attendance System, Mohammed A. Jabbar Hameed Available:"*http://www.iisjaipur.org/iiimcurrent08/mca_iv_sem_pro_eva/15.project%20attendence%20managemnt%20system.pdf*"

[2] Project Proposal Team 11, Calvin College, Available: "*http://www.calvin.edu/academic/engineering/2013-14-team11/Files/team11_ppfs.pdf*"

[3] Ali Mohammad Hossein Zadeh, Ah-Choo Koo, Hajar Ghadirian ,"*Attendance system based on mobile indoor location*".

[4] Sample Project Synopsis, Grasp Hub, Available: "*http://www.grasphub.com/2012/09/sample-major-project-synopsis-for-final-year-b-tech-b-estudents/"*

[5]. "*http://www.tutorialspoint.com/android/android_studio.html/*"

[6] System requirements, Slide Share, Available: "*https://www.slideshare.net/parryprabhu/android-system-requirements"*

[7]. "*http://www.javatpoint.in"*