

ANDROID CLASS ATTENDANCE MANAGEMENT SYSTEM

Mrs. V. ABINAYA M.C.A., M. Phil., (Ph. D)

Assistant Professor, Department of Commerce with Computer Applications

DR. N. G. P Arts and Science College, Coimbatore-48.

Mail id: v.abinaya@drngpasc.ac.in

Ms. GLADYS GISHA L

Student of III B COM (CA), Department of Commerce with Computer Applications

Dr. N. G. P Arts and Science College, Coimbatore-48.

Mail id: gladysgisha@gamil.com

ABSTRACT

The Android is a growing technology which have started to fulfil need with lots of application to make things handy. Smart Attendance application deals with the faculty members and to make the student attendance system more secure and portable. The Faculty member by make use of their login details they mark the attendance and then upload the report to mobile app. The faculty members can also be able to view the report of the absentees or presentees in accordance with date. The project application will be more compact and it is supported by most of the android mobiles.

The user interface is the important factor for an application. And the project provides an interactive design and user-friendly app, to obtain the user's attention. The admin of the college has the permission to enroll departments, the students and their basic details in each department. Admin only have the permission to make changes in each department. Admin also enroll the staff and their workflow and provide separate login for each staffs.

The staff by their proper user id and password, they enter login into this app and mark the attendances for their enrolled students. They only have the permission to put attendance for the current date attendance for each student. The staff can able to generate attendance report on basis of date for the presentees or absentees. The project provides the option to view report in mobile and also the report can be viewed and verified by the administrator.

Keywords:

Students, Admin, Faculty, android management system, mark attendance.

1. INTRODUCTION

The topic managers' primary responsibilities in the current situation are to record student attendance and mark attendance during class sessions since doing so can control student attendance and confirm the number of students in the class. A crucial duty for any university is the management and upkeep of student data. Traditionally, marking attendance and entering information for the exam are done manually in a log book. Later, desktop programmes take over for this duty. The desktop application is a standalone programme that is installed on a certain desktop or laptop, and only that desktop system may be used to carry out the duties. The machine digital attendance system, which was first launched in 1970, is the second generation of attendance systems. Digital attendance machines typically contain a few buttons (numerical and alphabetic) and a digital display that indicates the time or the displayed text [9]. Employees can input a password or barcode using the machine's number and alphabet keys. Because they may share the PIN with friends to take attendance, employees who cheat are still likely to do so when using digital attendance machines. Additionally, there is a sizable queue in front of the digital machine.

2. OBJECTIVES

For the daily activities of several organisations, the college web portal now offers online attendance. Through applications, the environment offers a simple way to interact with and update the data. A static web portal has a number of interactions and the option to change the entire application's content for any improvement.

3. EXISTING SYSTEM

The proposed system offers the user the convenience that the current method does not. A mobile student attendance system based on a smartphone app is maintained by the staff. The timetable shown in the Android mobile application created by the students and staff does not influence the user's preference for login.

4. PROPOSED SYSTEM

In contrast to the current approach, the proposed solution provides the user with convenience. The staff takes care of a mobile student attendance system built on a smartphone app. The timetable displayed in the by students and staff-created Android mobile application has no bearing on the user's preferred login method.

5. ANDROID ATTENDANCE MANAGEMENT SYSTEM

5.1 Android

Android is a Linux-based operating system with a Java programming language. It offers tools like a device emulator, a debugger, and its own Java Virtual machine (Dalvik Virtual Machine - DVM), among others. The Open Handset Alliance, which is headed by Google, developed Android. A unique virtual machine, such as the Dalvik Virtual Machine, is used by Android. Dalvik use unique bytes of code. As a result, Android cannot run conventional Java byte code. A programme called "dx" is included with Android that enables the conversion of Java Class files into "dex" (Dalvik Executable) files. The "aapt" (Android Asset Packaging Tool) programme condenses Android applications into an.apk (Android Package) file to streamline development. The Android Development Tools (ADT) for Eclipse are made available by Google. Automatic performance of the ADT.

Android SDK

The Android platform is provided under one of the most forward-thinking, developer-friendly open-source licences, giving mobile operators and device makers a great deal of creative freedom. The Alliance has made an early access software development kit available to give developers the resources they need to produce unique and engaging platform apps. For users, mobile service and device developers, and manufacturers, Android holds the promise of hitherto unheard-of advantages. Wireless carriers and handset makers will have complete freedom to alter Android in order to offer cutting-edge new devices more quickly and cheaply.

5.2 SQLite in Android

Android includes an open-source database called SQLite. Standard relational database capabilities including prepared statements, transactions, and SQL syntax are supported by SQLite. Additionally, at runtime, it uses very little memory. No database administration or setup is needed to use SQLite on Android. The database is automatically handled for the user and the user defines the SQL to use while working with the database. The data types text (similar to Java's string), integer (similar to Java's long), and real (similar to Java's double) are supported by SQLite. Before being stored in the database, all other kinds must be transformed into one of these fields. The types supplied to the columns are not verified by SQLite to ensure they are the defined kinds; a user must do so.

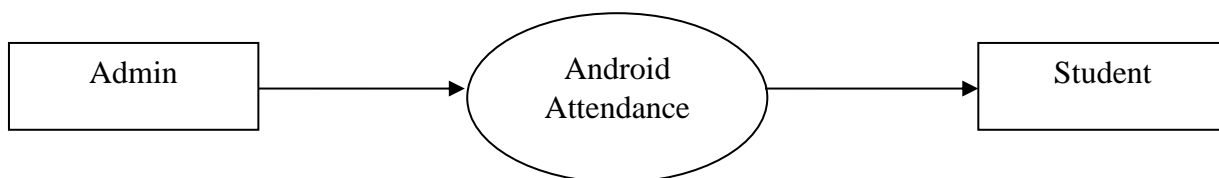
SQLite Database and Cursor

The insert(), update(), and delete() methods are offered by SQLite Database. The methods.rawQuery(), which accepts SQL, query(), which offers an interface for specifying dynamic data, and SQLite Query Builder can all be used to generate queries. Since a content provider's interface resembles that of SQLite Builder, it is frequently used for them. A cursor is always returned by a query. The outcome of a query is represented by a cursor. Use the getCount() function to determine how many elements there are. The methods moveToFirst() and moveToNext() allow you to navigate between different data rows individually.

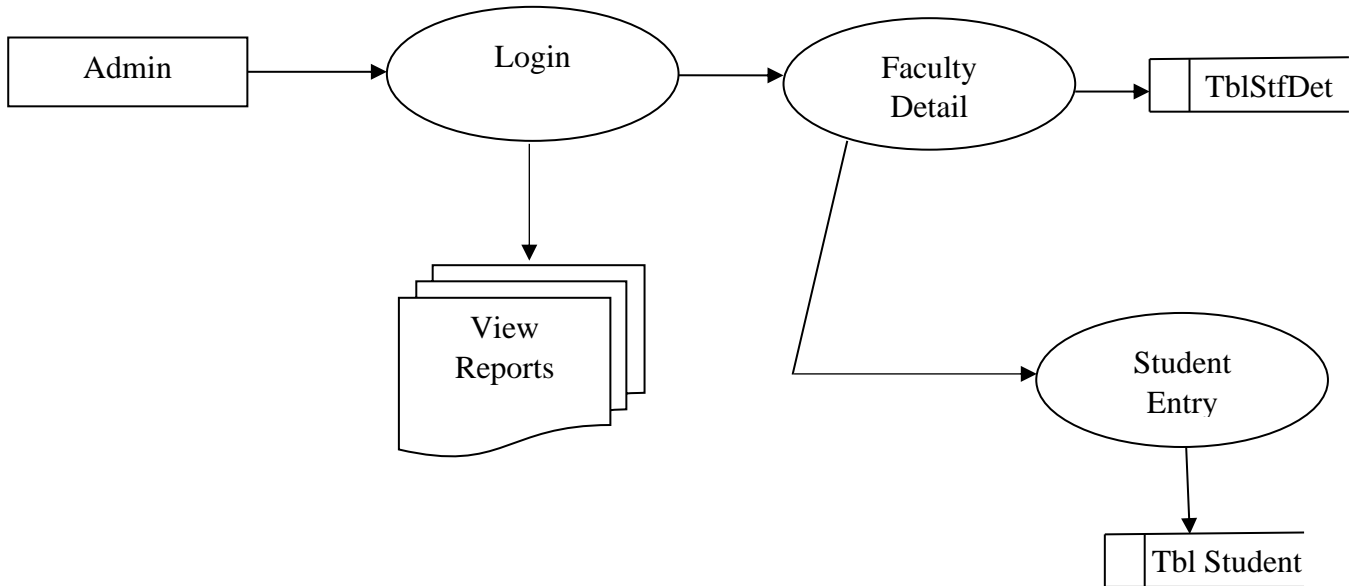
6. DATA FLOW DIAGRAM

A data flow diagram is a visual representation of how information moves from input to output as well as the transformations that are used. Any level of abstraction of a system or piece of software can be represented using a data flow diagram. DFDs can be divided into tiers that correspond to escalating functional and informational complexity.

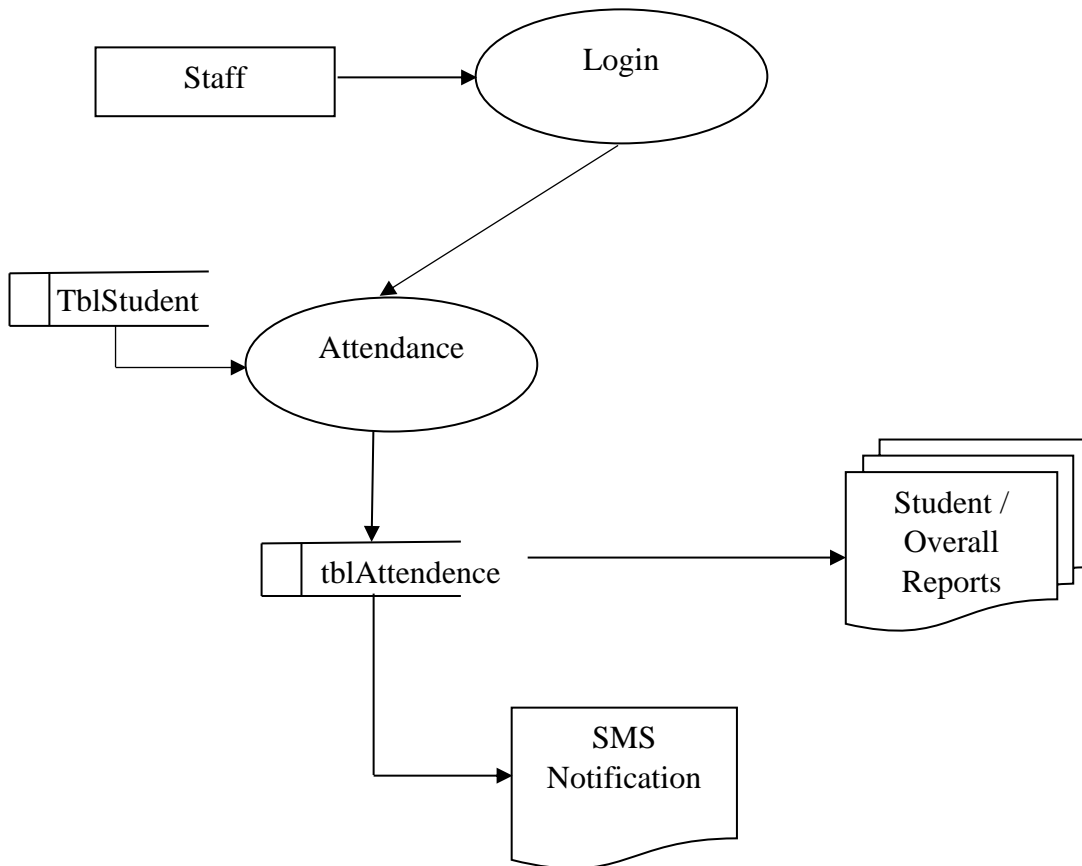
6.1 LEVEL 0:



6.2 LEVEL 1:



6.3 LEVEL 2:



7. SYSTEM TESTING

Software testing is an important element on S/W quality assurance and represents the ultimate review of specification, design and coding. The increasing visibility of S/W as a system element and the costs and associated with a S/W failure are motivating forces for well planned, through testing. Though the test phase is often thought of as separate and distinct from the development effort – first develop, and then test – testing is concurrent process that provides valuable information for the development team.

There are at least three options for integrating Project Builder into the test phase

- The testers import the same project or projects that the developers use.
- Create a project based on the development project but customized for the testers (for example, it does not include support documents, specs, or source), who import it.

- A combination of the second and third options works best. Associating the application with a project can be useful during the testing phase, as well.

UX testing

It's crucial to test how your Android application's user interface (UI) behaves when it is running on a device in addition to unit testing the many parts that make up your programme, including as activities, services, and content providers. UI testing makes ensuring that your programme responds to a series of user activities on a device—such as pressing toolbars, menus, dialogue boxes, images, and other UI controls—in the right way.

Testers only need to be aware of the app's intended result when a user takes a certain action or provides a specific input while conducting functional or "black-box" UI testing.

8. IMPLEMENTATION AND MAINTANENCE

The system must then be translated from a designed state into actual code in order to fulfil the needs of the user. End-user training is part of implementation. The programme that has been implemented needs to be maintained for continued use. The technology was initially used in conjunction with a manual system. The system has undergone data testing and shown to be both error-free and user-friendly. The end-user received instruction on how to utilise the software and its capabilities.

The software engineering tasks that take place after the software product is delivered to the client are referred to as "system maintenance" in this context. The period during which a software product is functional is known as the maintenance phase of the software lifecycle. All actions taken to keep the system operational after software installation are referred to as maintenance. As was previously said, software frequently has design problems. Adaptive maintenance and corrective maintenance are the two basic categories of maintenance activities.

9. CONCLUSION

It is discovered that the "Android Class Attendance Management System" is a lot faster, more dependable, more user-friendly than the current system. The technology has undergone successful testing after being carefully planned and constructed. The system has been tested with all conceivable input combinations, and the outcomes have been found to match those anticipated. Android and SQLite are both used in the system's full development. With messages and control information outputs made available in the necessary format, it has been designed with user friendliness in mind.

10. REFERENCES

- B. Soewito and F.L Gaol, E.Simanjuntak and F.E. Gunawan, “Attendance System on Android Smartphone” in 2015.
- S.A.M. Noor, N. Zaini, M.F.A Laptip, and N. Hamzah, “Android based attendance Management System in 2015.
- Karwan Jacksi, Falah Ibrahim, Shahab Ali, “Student Attendance management System”, University of Zakho, Iraq, 2018.
- Mohammad Ausaf Anwar, Durgaparsad Gangodkar, “Design and Implementation of Mobile phones based Attendance Marking System, India, 2015.

Sites Referred

<http://developer.android.com/guide/index.html>

<http://www.codeproject.com/Articles/102065/Android-A-beginner-s-guide>

<http://mobile.dzone.com/articles/fundamentals-android-tutorial>

<http://mobile.tutsplus.com/tutorials/android/java-tutorial/>