# TE-KER: Student Attendance Checker Mobile Application Using Face Recognition

A Capstone Project

Presented to the Faculty of the

College of Computer Studies

Lyceum of the Philippines University Batangas

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Information Technology

Specialized in Multimedia Technologies

by:

Ken Alvin B. Alvarez

Aleci Andrei M. Marasigan

Renz Joshua L. Miral

Mariel Angelie P. Perez

December 2018

**APPROVAL SHEET**

In partial fulfilment of the requirements for the degree Bachelor of Science in Information Technology (Specialized in Multimedia Technologies), this capstone project entitled "**Te-ker: Student Attendance Checker Mobile Application using Face Recognition"** has been prepared and submitted by **Ken Alvin B. Alvarez**, **Aleci Andrei M. Marasigan**, **Renz Joshua L. Miral, Mariel Angelie P. Perez** and is hereby recommended for oral examination.

_____
**Engr. Joselito A. Dolot, MSc**
Adviser

Defended in an oral examination before a duly constituted panel with a grade of _____

_____
**Mrs. Maria Cristina M. Ramos, MSCS**
Chairman

_____ _____
**Mrs. Melody R. Dimaano, MDM, MSCS** **Miss Maria Julieta A. Saldua, MSIT**
Member Member

_____
**Dr. Imelda L. An**
Member

Accepted in partial fulfillment of the requirements for the degree Bachelor of Science in Information Technology (Specialized in Multimedia Technologies).

_____
**Mrs. Roselie B. Alday, MCS, PhD Cand.**
Dean, College of Computer Studies

# TE-KER: Student Attendance Checker Mobile Application Using Face Recognition

**Ken Alvin Alvarez**
Lyceum of the Philippines University
Batangas City
(+63)9292194593
kenalvinalvarez@lpubatangas.edu.ph

**Aleci Andrei M. Marasigan**
Lyceum of the Philippines University
Batangas City
(+63)9550385660
aleciandreimarasigan@lpubatangas.edu.ph

**Mariel Angelie P. Perez**
Lyceum of the Philippines University
Batangas City
(+63)9358697974
marielangelieperez@lpubatangas.edu.ph

**Renz Joshua Miral**
Lyceum of the Philippines University
Batangas City
(+63)9155153711
renzjoshuamiral@lpubatangas.edu.ph

## ABSTRACT

This capstone project entitled "TE-KER: Student Attendance Checker Mobile Application Using Face Recognition" aim at providing a modern way of checking attendance using facial recognition. The main goal of this application is to help the teachers regarding the attendance of the students. Apparently, teachers may have some difficulty when it comes to checking attendance. Its either they need to call the names of the students one by one or they need to pass around a piece of paper just for the students to sign for confirmation of attendance and it is not accurate because some of the students may ask their friends to sign their attendance even if they are not around. Hence, the study can help the teachers not only to lessen their time in checking attendance but also an easy way of generating attendance report. It also has a feature where you can notify the parents of the student if the students have absences.

**KEYWORDS:** *attendance checking system, face recognition , mobile application*

## ACKNOWLEDGEMENT

The success of this study required the help of various individuals. Without them, the researchers might not meet their objectives in doing this study. The researchers want to give gratitude to the following people for their invaluable help and support

To Jesus Christ, our Lord and Savior, for giving the wisdom, strength, support and knowledge in exploring things; for the guidance is helping surpass all the trials that the researchers encountered and for giving determination to pursue their studies and to make this study possible.

Second, to our research adviser, Mr. Joel Dolot. He has been there providing his heartfelt support and guidance at all times and has given us invaluable guidance, inspiration and suggestions through our capstone.

Third, to our panelist Mrs. Maria Cristina M. Ramos, Mrs. Melody R. Dimaano and Ms. Maria Julieta A. Saldua for giving ideas for the improvement of the study, for their patience, understanding and encouragement over this past semester.

Nobody has been more important to us in the pursuit of this project than the members of our family. We would like to thank our parents, whose love and guidance are with us in whatever we pursue. Lastly, to the people who helped and contribute great ideas and advices, especially classmates and close friends for without them, this study would not be possible.

## DEDICATION

This capstone project is wholeheartedly dedicated to almighty God, who provides us strength, inspiration, power of mind and goodwill.

We also dedicate this research to our parents, brothers and sisters who were supportive during the making of our study, our source of inspiration and gave us strength when we thought of giving up, who continually provide their moral, spiritual, emotional, and financial support

Most of all, we dedicate this study to ourselves and fellow researchers for great time, for the love and participation they have shown as we reached the fulfillment of this study.

Ken

Aleci

Renz

Mariel

## TABLE OF CONTENTS

## LIST OF FIGURES

# TE-KER: Student Attendance Checker Mobile Application Using Face Recognition

**Ken Alvin Alvarez**
Lyceum of the Philippines University
Batangas City
(+63)9292194593
kenalvinalvarez@lpubatangas.edu.ph

**Aleci Andrei M. Marasigan**
Lyceum of the Philippines University
Batangas City
(+63)9550385660
aleciandreimarasigan@lpubatangas.edu.ph

**Mariel Angelie P. Perez**
Lyceum of the Philippines University
Batangas City
(+63)9358697974
marielangelieperez@lpubatangas.edu.ph

**Renz Joshua Miral**
Lyceum of the Philippines University
Batangas City
(+63)9155153711
renzjoshuamiral@lpubatangas.edu.ph

## ABSTRACT

This capstone project entitled "TE-KER: Student Attendance Checker Mobile Application Using Face Recognition" aim at providing a modern way of checking attendance using facial recognition. The main goal of this application is to help the teachers regarding the attendance of the students. Apparently, teachers may have some difficulty when it comes to checking attendance. Its either they need to call the names of the students one by one or they need to pass around a piece of paper just for the students to sign for confirmation of attendance and it is not accurate because some of the students may ask their friends to sign their attendance even if they are not around. Hence, the study can help the teachers not only to lessen their time in checking attendance but also an easy way of generating attendance report. It also has a feature where you can notify the parents of the student if the students have absences.

**KEYWORDS:** *attendance checking system, face recognition, mobile application*

## 1.0 INTRODUCTION

A research conducted by Elaine R. Manalo entitled "Absenteeism of the Students of Lyceum of the Philippines University

photo in the database to be compared when the attendance checking is done. Using excel format where the system will base to check

Batangas for the First Semester of SY 2008-2009" got a total of 1915 student with absencences on the said academic year. Until today, most of the professors in school are still using the conservative way of taking students' presence one of the best examples of these is either by calling their names one by one or going around a participation sheet to assign affirming their essence. Furthermore, it is a tedious task . It is also greatly opposing reducing amount of paper resources needed in data management. [1] In addition, instructors through big lecture may discover the disturbance of taking the presence sheet being passed everywhere in the class and the manual validation of attendance of students. [2] Additional matter of taking the attendance record in a firm copy form is that an instructor may misplaced the sheet. [3]

**TE-KER: Attendance Checker Mobile Application Using Facial Recognition** is a mobile application that will check the attendance of the students using facial recognition. This application will save the attendance of the students automatically and they can also add some information about the students such as name, section, time, subject taken and the face Id of the student. It is used for the comparison of the student photo in the their attendance. It can also generate the attendance report of the students. The proponents will be using SQL Database for the Check Attendance feature, Class Management feature and View Attendance feature. This application is also a cloud-based app where the pictures will be stored to minimize the storage usage. The professor should create an account in Azure to get the face Id of each students. This application has a feature of Attendance Report where they can find if the student has absences and the professor can send a text message to inform the student.

The limitation of this application is it can only check around 15-20 students in a class. The students must be near to the instructor, so their face will be recognized

This application will only be available for Android user having an 8.0 OS, the latest version. The RAM and external storage or internal storage must at least be 2 GB. Flash is also imperative in order to make sure that the picture to compare with the database is clear and bright.

### 1.1 OBJECTIVES OF THE STUDY

1. To provide teachers with a mobile application that will monitor attendance of students and generate reports.

2. To develop the mobile application using facial recognition.

3.To use SQL for the database and Android Studio as a programming language in the development of the application

## 2.0 LITERATURE REVIEW

A mobile application, usually stated as an app, is a kind of application software intended to run on a portable device, such as mobile phone or computer. Mobile application often helps to offer users with comparable amenities to those retrieved on PCs. Apps are usually minor, distinct software parts with partial meaning. This use app software was formerly promoted by Apple Inc. and its App Store, which bids thousands of applications for the iPhone, iPad and iPod Touch. A mobile application is also known as an app web app, online app or mobile phone. [4]

Mobile application is a change since the unified software systems mostly originated on PCs. As a replacement, each app offers partial and inaccessible functionality such as game, calculator or mobile web browsing. Through applications might have evaded multitasking because of the incomplete hardware properties of the premature smartphone devices, their specificity is today portion of their prestige because the allow consumers to select what their devices are gifted to do.

The modest smartphone application takes computer related apps and port them to a smartphone device. As mobile apps develop extra vigorous, this method is rather missing. An additional classy method includes emerging exactly for the portable setting, captivating benefit together its boundaries and returns. In case applications that use location-based features are characteristically constructed form the smartphone, this specified that the manipulator prepares the similar thought of site on a computer. [5]

Furthermost, smartphone application is a key vital initiative to indicate among cradle-based management over a chain line system or a clean available wireless result (PENTA Group, 2010). Through portable applications are a fast-increasing part of the international mobile market, this contains the software runs on a mobile device and achieves convinced responsibilities for the customers. Also, the numerous meanings with user interface for rudimentary telephone service

in addition to progressive facilities smartphone application are broadly used by imposts. Smartphone application is a great and endlessly rising market and helped by cumulative amount of portable app inventers, producers and workers (The Mobile Marketing Association Group, 2008). Innovative study proposes that the universal market for mobile app will be finished in the upcoming two centuries. Study has been located done for Getjar, the biosphere's next major Apps store.

iPhone has represented as an uprising of worldwide smartphones. It has usual a stage to advance all kinds of mobile apps serving an improved image in portable computing. The roles of iPhone are that it is abundant with multitouch edge, acidometer, GPS, proximity sensor, dialer, sqlite3 database, OpenGL ES, Quartz, Encryption, Audio, Game and Animation, Address book and Calendar including latest features of iPhone Gaming.

Windows mobile application is the newest technology every portable device should develop. This technology permits the handler to look at the internet, lead obtain correspondences, check with timetables acquaintances, and make performances, in small accomplish entire corporate with the custom of a mobile. Mobile phone has been making waves in the approaching eras above practical interface in the mobile biosphere. This newest modernization benefits the procedures to bring applications straight to the manipulator and informal copies of application can be attained. [6]

**Attendance Checking**

Each university spends much time monitoring student's attendance very extremely. Several universities already have procedures in classrooms to record and follow up insistent presence as an essential part of their responsibility.

Attendance is very significant in every student, only inattentive is large change in presentation in the school. Typically, students of high school are disposed to absences, it is because about the details that they deliberate is a boring class, idleness to join in the class, approximately students desire is to go to computer shops playing games, somewhat coming in the class and around, students waste the effect of an acquaintance or peer pressure.

Furthermost, the schools these days are applying computerized techniques and procedures to encounter their rising requirements, and it can be best distinct by

commonly used of computers and other devices. Attendance is a recording of statistics of student present during actions, usually defined as methodical accurate all. It is a responsibility of every student as completest potential. It is very significant to the others to keep their presence in school or at work. [7] Inspiring consistent school attendance is one of the greatest influential habits one can make for teen's achievement in cooperation with university. Once school attendance is made an urgency, the child become well assisted to his success and grow well natural life behaviors, evade unsafe performance and take a healthier unintended of proceeding from high school. It is a statistic that students who join in school frequently, study more and become positive in university than students who do not. Guardians who make consistent school attendance a precedence are also serving their children study to take charge, and that is a significant example for a fruitful lifetime. Attendance forms in premature lifetime. Students who grow decent attendance behaviors in the initial age will be further possible to remain the during their school career, besides into their selected profession. As soon as student are inattentive for less days, their marks and understanding services frequently recover – even among those students who are stressed in school frequently also sense more communal services and relationships, and are communal services and relationships, and are meaningfully more possible to progress form high school, set them up for a solid upcoming.

**Methods in Checking Attendance**

Instructors use paper sheet to check presence and it is not an effective way because it is time fascinating matter. They will be consuming more time for calling the students names and checking the present and absent. Sometimes, dishonest student announces a friend is present while in fact, the student is absent.

According to Dobson, tracking attendance can be time-consuming and tedious chore. Typically, the professor takes attendance manually by requesting that every student says "here"; when his or her track of the students' physical location name is called or by scanning the classroom to figure out which student there are. The professor then records the data and it is transmitted to school organization, frequently by hand. Despite the fact that it is an incredible idea, it likewise experiences its own shortcomings as it does not give any verification means for the

attendance data integrity. Some errors could emerge for various reasons, for example, if the student forget, his or her card, or if he swaps the tags or carried the tags of an absent student. Likewise, the past framework could not export or generate report various formats that the school administration is requiring. The system is not capable of providing the teachers to generate instant attendance reports. [8]

One of the greatest communal parts where laboratory performs distinct faculty member variation is attendance rule Approximately, professors necessitate attendance. Around faculty tally attendance completely in rating purpose, whereas others, total the absence of attendance in contradiction with the student's ranking. Furthermore, faculty who not need attendance by their students inspire attendance in a diversity of conducts. Fundamentally, faculty who trust the attendance is significant in student's accomplishment but most can offer one unreliable sign to care confidence. This newssheet covers précises of about utmost new study on the part of class attendance on student presentation. Nearly new study will likewise be discovered that validates the effect of class attendance on further variables that mark the general speculative achievement of an organization. [9] Presence of students into laboratories or meeting places is significant and can be measured as the preliminary fact to achieving a decent teaching. This procedure consequently needs devotion of the teacher. Among the traditional procedures of checking presence of the students are considerating out the student designations and mark if they are currently in the class or authorized a sheet of the students are existing in the room. The frauds of the traditional technique is communication among students and the instructor over how the teacher needs to know the students in a effort less time consuming while marking the presence of the students in a full class [10].

**Methods in Checking Attendance using Mobile Applications**

According to Chauhan, a biometric (fingerprint recording) system is their way of monitoring and checking the attendance. It is used to register the attendance of students and teachers while a Global Positioning System (GPS) will trail food grains for the Mid-Day Meal Scheme, Minister of State for HRD D Purandeshwari told HT. As for the researchers, they developed their project with the use of the latest trends of technology within this generation. Android is one of the

most efficient technologies to use. Almost everyone has this kind of technology to use for their personal purposes like gaming, entertainment, education and etc. The researchers take this advantage and grab the opportunity because it would be a great help to fellow students and professors to have an easy way of checking the attendance of their classes. Attendroid can easily check attendance by the use of portable hotspot that is provided in every android device. Unlike the use of Bluetooth technology, portable hotspot has an advantage when it comes to speed and radius that make it easier for the instructors to check the attendance of the students. Attendroid provides generation of report that is not a feature of the application listed in the proponent's research literatures. Synchronization of data is also one of the feature of Attendroid. Management of student records is also a part of Attendroid. [11] Rastogi and Gupta established an application to record attendance through smartphone device. The goal of their plan was to make an Android mobile application for attendance system that can be used by all teachers for their own courses. The project of the application was that the information, facts of the developments selected to the particular facility, and the student registered in the course are done to be thru from the server using an internet connection and store it in the mobile database. This application stores the attendance in the smartphone internal database, where the professors can view and be informed by the attendance whenever required. Additionally, the researchers decided that the attendance system through mobile devices is a very real tool which can be used to a countless extent. The system is transportable and can simply be connected and used on any mobile phone supporting an Android operating system (OS). The routine of this system outcomes in a discount in the amount of periods consumed in attendance details into the server database. Likewise, it delivers an interface which is informal to appreciate by the users and importantly supports in familiarizing the routine of this system.

**Studies about Mobile to Check Attendance**

Researchers about checking attendance have greatest significant method to record and track the attendance of students in schools and universities. Attendance checking in dissimilar procedures has been in routine in numerous groups to record the attendance of the students. This benefits the group in creating their month-end staff, and extra actions. Such structures might be physical or automatic. Biometric-based

system and card-based system are the samples of automatic attendance structure. On the other hand, in the manual attendance system where the professors check the attendance of the students who are registered in dissimilar programs were occupied by the facility. The data concerning the presence is then approved on to the academic unit of the organization where the data is feed into the organization server file. The application is connected in each professor's smart phones and can be used to take presence in disconnected Wi-Fi. This request raises the facts of the profession selected to the individual professor and the students registered in the courses from the server using the internet connection and stores it in the mobile database. The professors can access and keep informed the attendance when vital. At the end of the month, once the professor wanted to upload the presence, they can straight upload it coming from their mobile to the server providing the Wi-Fi connection is accessible. [12]

K. Akhila et al made an android-based mobile application for student attendance tracking system. It bids consistency, less time consuming, and it is handy to use; taking the attendance using android mobile phones. It can decrease the labors of the operate associates near attendance conservation. It is a well-organized and accessible mobile application for attendance monitoring. Rakhi Joshi et al established a mobile-based presence management with knowledge structure. The web-based mobile application is established in a SQL server. The system is easy to mark attendance of the students through mobile phone and make a preceding suggestion to student as soon as their presence drives under the specified like SMS. Furthermore, Amita Dhale et al made a study on "smart connect", mobile and web-based application for school management system. It is established by means of SQL server. It is mostly used to supply the facts vital for the organizations.

In calculation to provide the verification in administrations, the location-based attendance management system is also trained. Mohammad Salah et al (2010) offered a mobile application for period and presence system built on the site. This application is used to check the presence of the staffs created on whether they are in the similar position of group or not. This application is established using mobile. Bluetooth and Wi-fi are used with wireless.

A mobile application for attendance management system by Riya Lodha et al (2014) established an application for attendance management system using Bluetooth allowed devices. This application purpose with the wireless skill is using Bluetooth to check the attendance. Hence, it decreases the period occupied for the presence marking. Freya. J. Vora planned a basis of android-based mobile attendance system. It uses Wi-Fi technology to check the attendance in android based phones. It is permitting to stock and manage the attendance.

Student's attendance tracking is a vigorous matter in demand to check students' presentation in the laboratory even in their lessons. It develops an important apprehension since the college specialist upholds a law that single student can only join in the examination if his/her attendance is advanced or equivalent to numerous proportions (60%,70% or 80% etc.) or else not. The old-style presence system wants students to actually mark the presence piece each time for the presence of each lesson. This is needlessly inefficient to sign and spot student's name on the presence sheet. Correspondingly, it occurs that students may unintentionally or freely sign the student's name similar as substitution. The firm reproduction of presence sheet might become lost. By means of Mobile Phones similar as Android Technology, the professor will be gifted to check attendance effortlessly by planned mobile application and import the attendance in the phone as well as in server and can mark the average and as well as they can provide a hard copy. By means of kept data, this structure is able to check attendance, check intruders' admission, presence proportion, direct communications, and direct message to the custodian to retain them efficient about their child's attendance at the Institution. The planned system has connected access since any room might extremely support the progression of the professor with custody path of their student's presence. [13]

## 3.0 METHODS

### 3.1 Research Method

This research undertakes the checking attendance using face recognition, its procedures and methods. For appropriate organizations, the researchers discuss that the method or model that will be used is Mobile App Development Lifecycle. This covers initial planning, prototyping, testing and development.

Creating a mobile application is known through creating a fruitful mobile application. This is a procedure which includes fairly wide planning. This lifecycle also discourses about unique features of mobile applications like lifespan composite functionalities, less corporal interfaces and additional. Initial testing of the MADLC designates that this lifecycle will assist the creators professionally to perform developments and transport resolution on period. [14]

**Initial Planning.** As with most any development project, the first step is go through an initial planning. There will be amount of time considering what the end user wants to achieve, who are the users and what is the purpose of building an application. Researchers will present topics with their adviser. They will discuss the usual problems and how can the researchers solve it. After the discussions, the topics should be consulted to heir adviser, Mr. Joselito Dolot and be approved by the Dean.

**Design.** In this phase, the researchers will create an interface that can easily understood with the use of Android Studio as their development tool, Adobe Photoshop and Adobe Illustrator for creating layout designs. The researchers must already cite functionalities, features and appropriate guidelines of the application software. The researchers should start visualizing flowcharts because it can help them with the flow of their application.

**Building.** If all the phases of designing process were completely done, the researchers will proceed to the development phase which is the draft of the system and construct then develop.

**Prototyping.** In this phase, the useful necessities of each model are evaluated; the prototypes are verified and directed to the user response. Afterwards, the response is received from the client, the vital changes are applied through the development phase. When the next prototype is prepared, it is combined with the primary prototype, verified and directed to the client. The last prototype is directed to the client for a last response. In the phase also, the work is completed and documented. It will be forwarded to the testing phase.

**Testing.** Testing is one of the significant phases of any development. The

prototype testing is performed on an emulator or simulator and it is often provided in SDK. If the application is already done, this will be the last phase where the researchers will debug the whole program to determine the errors and complications. They will also do some test to distinguish if it is working properly like installing it to different android phones. All the features inside the application must be working properly.

**Deployment.** After the testing is completed and the final feedback is obtained from the client, the application is ready for deployment. The application is uploaded to the appropriate application store/market for user communication. Before the application is deployed, the following steps are to be checked. The application should be completely working, and it is time to launch the application in Google Play Store.

## Flowcharts

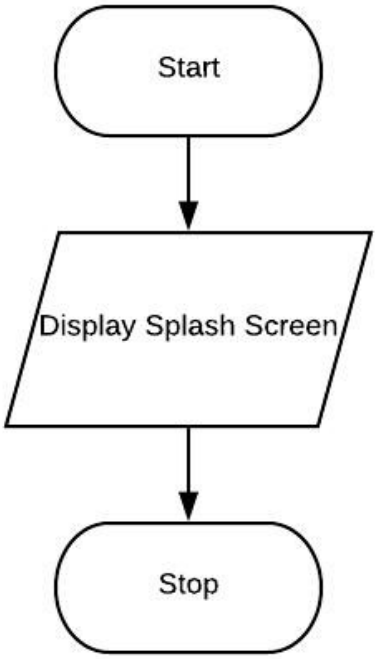


**Figure 1. Splash Screen**

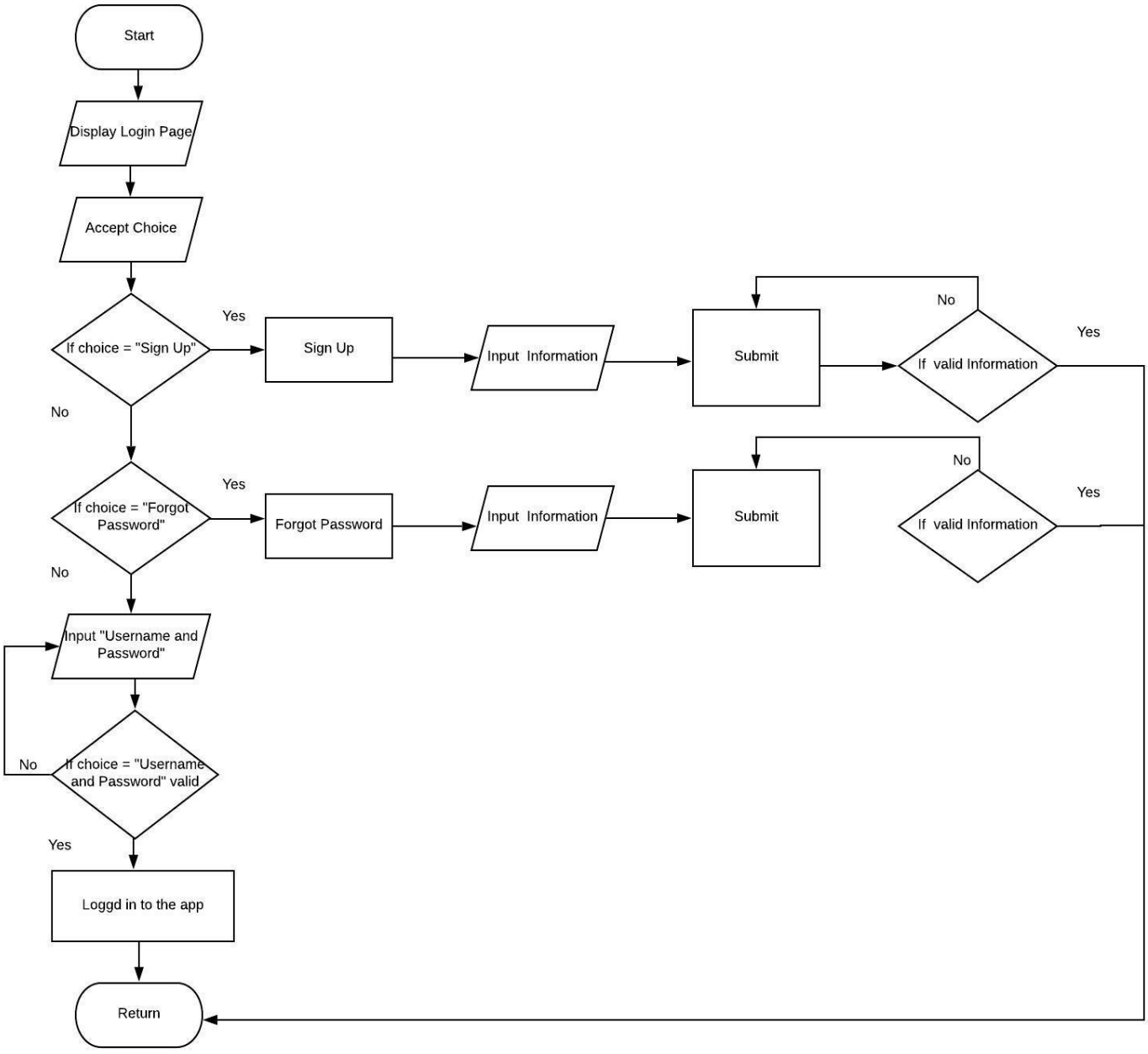


**Figure 2. Login Page**

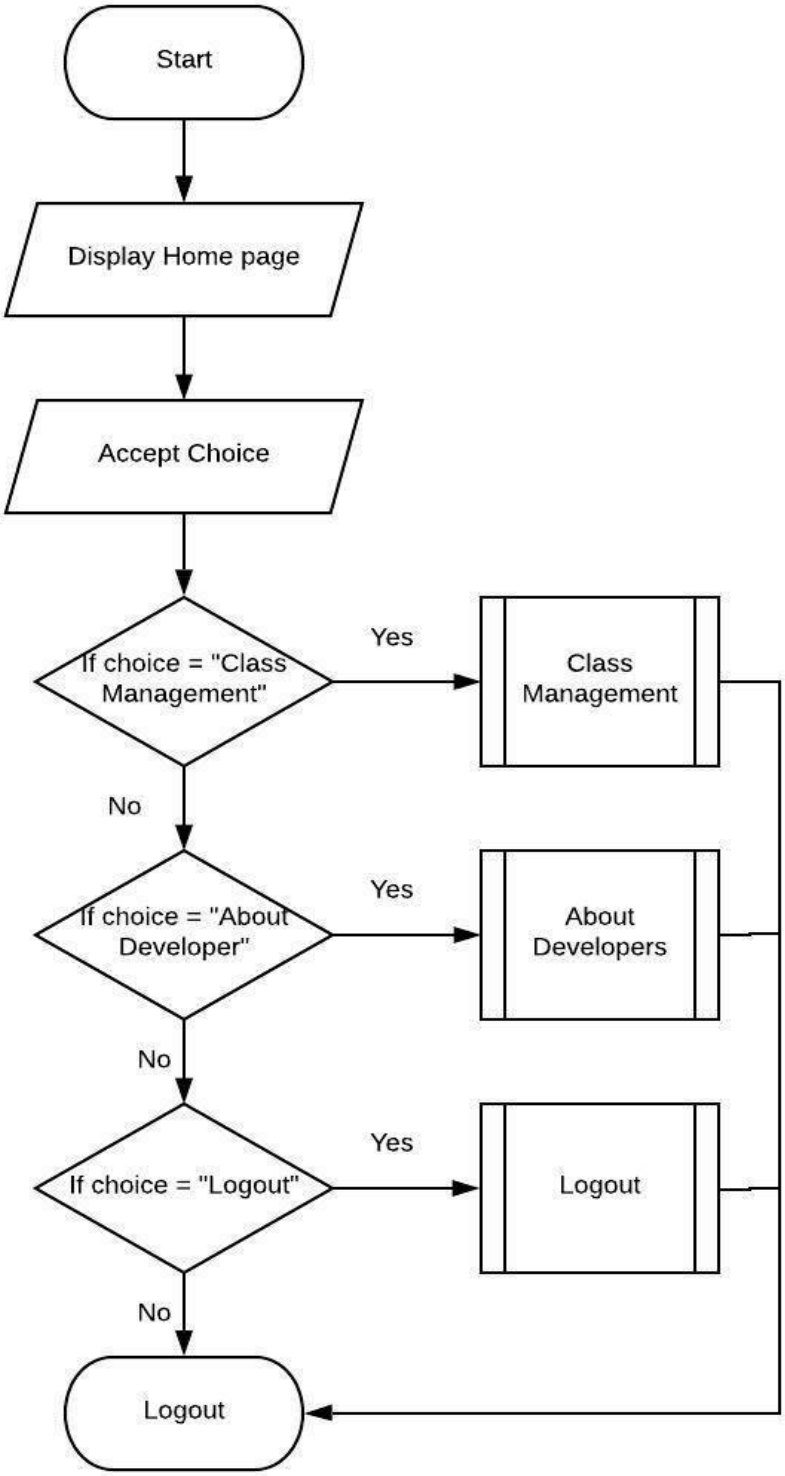


**Figure 3. Home Page**

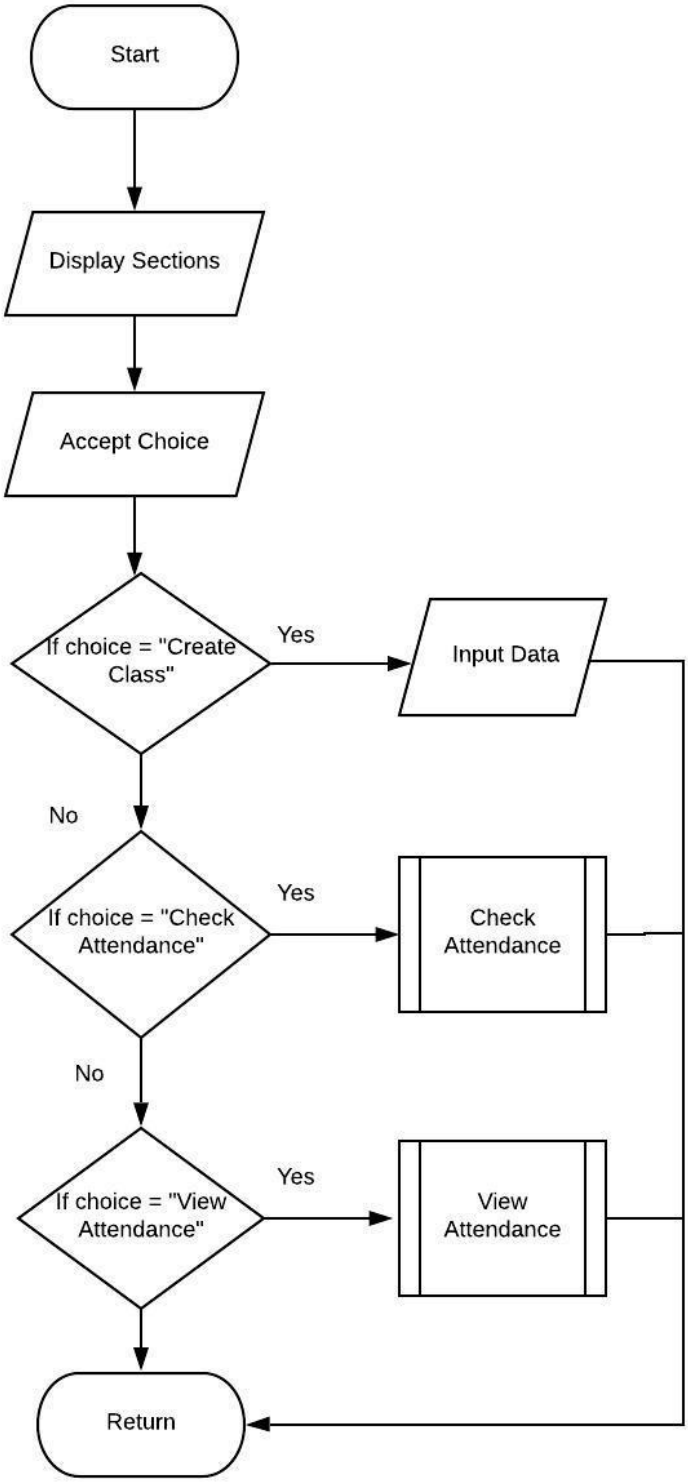


**Figure 4. Class Management**

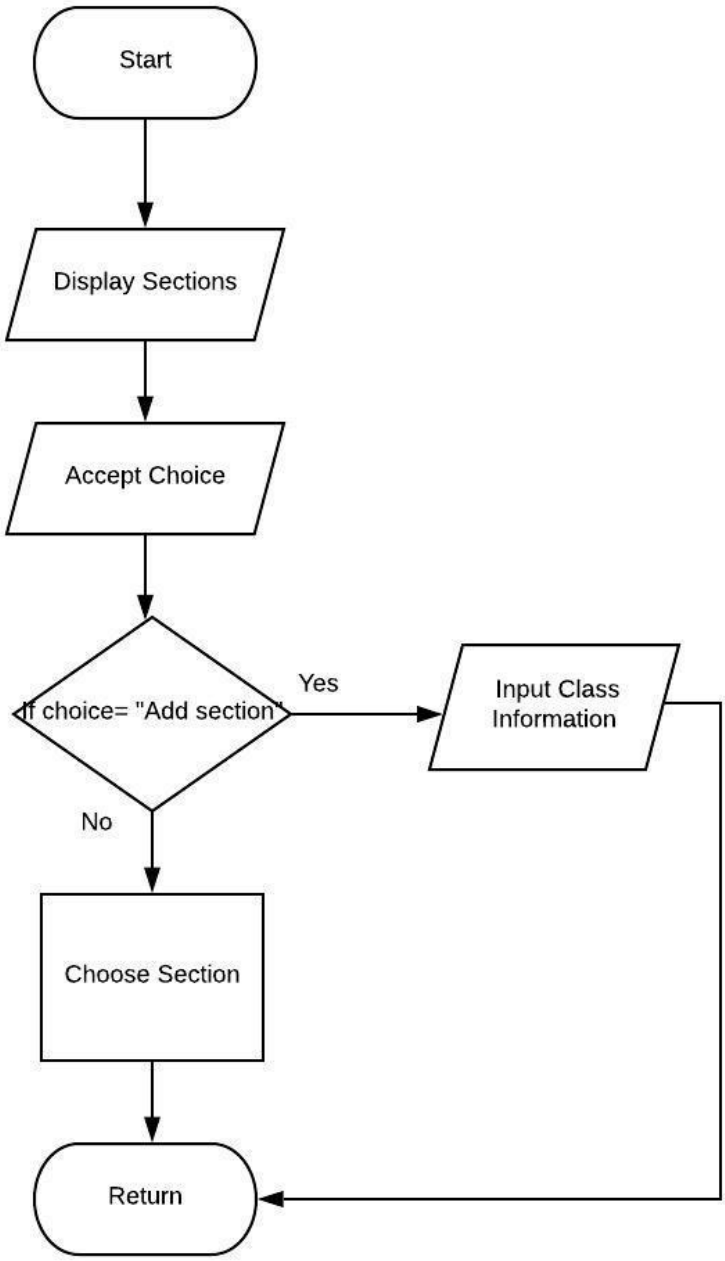


**Figure 5. Add Section**

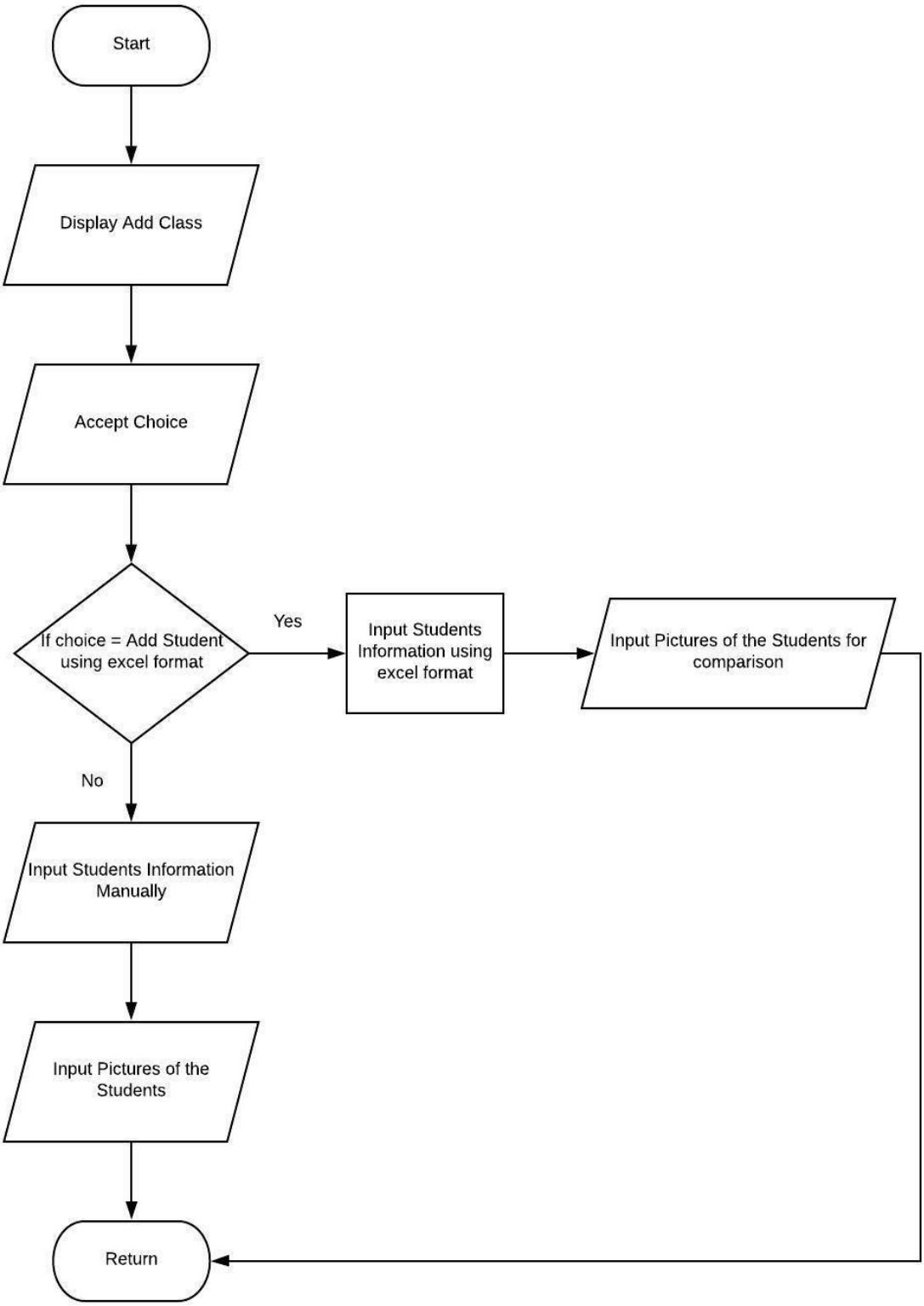


**Figure 6 . Add class**

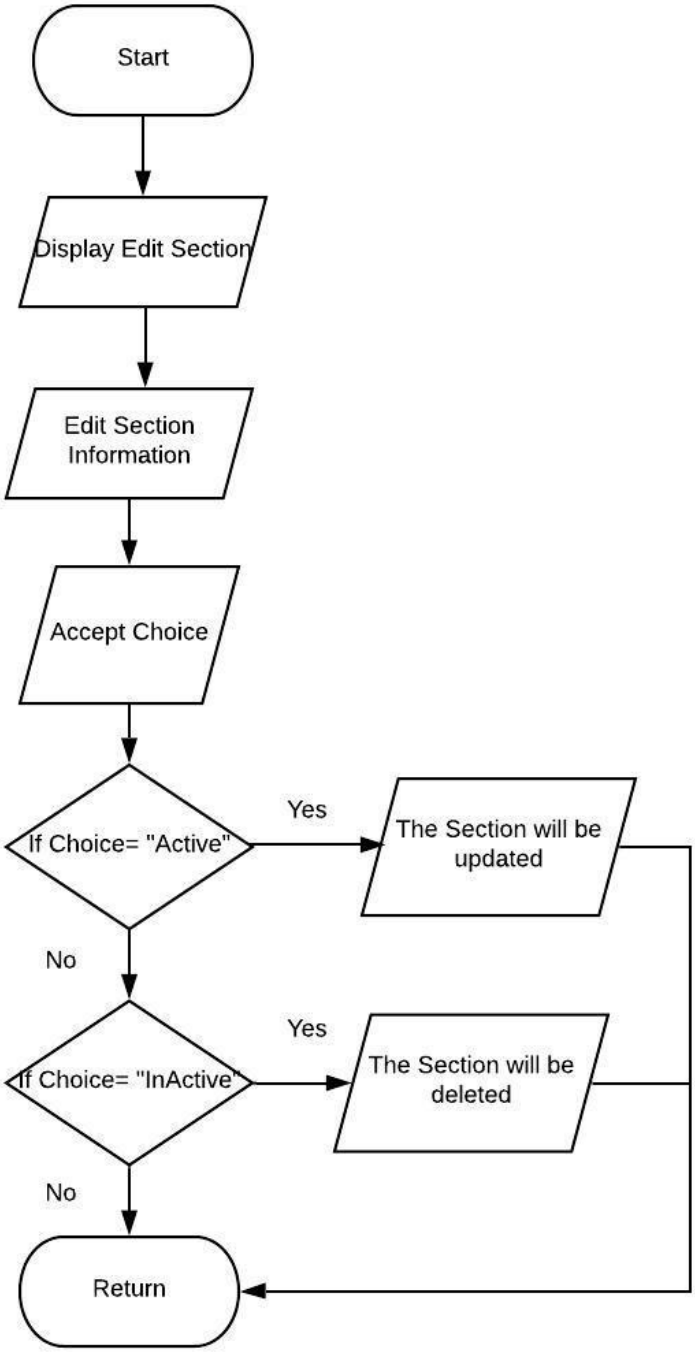


**Figure 7. Edit section**

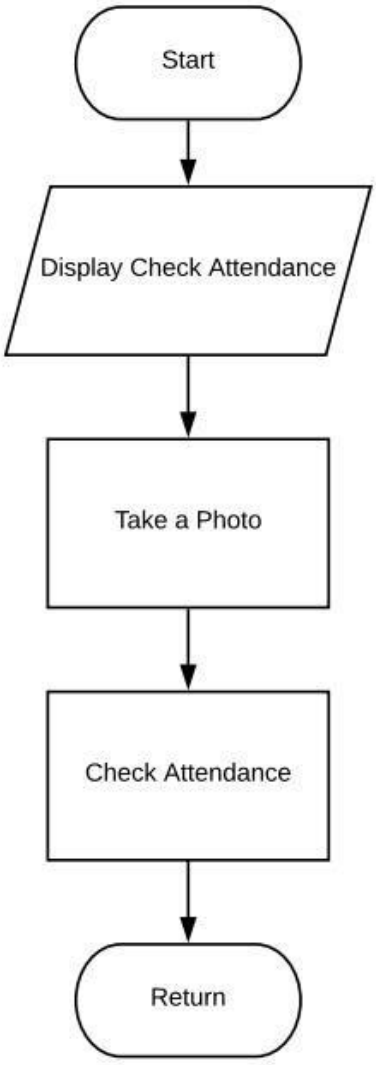


**Figure 8. Check Attendance**

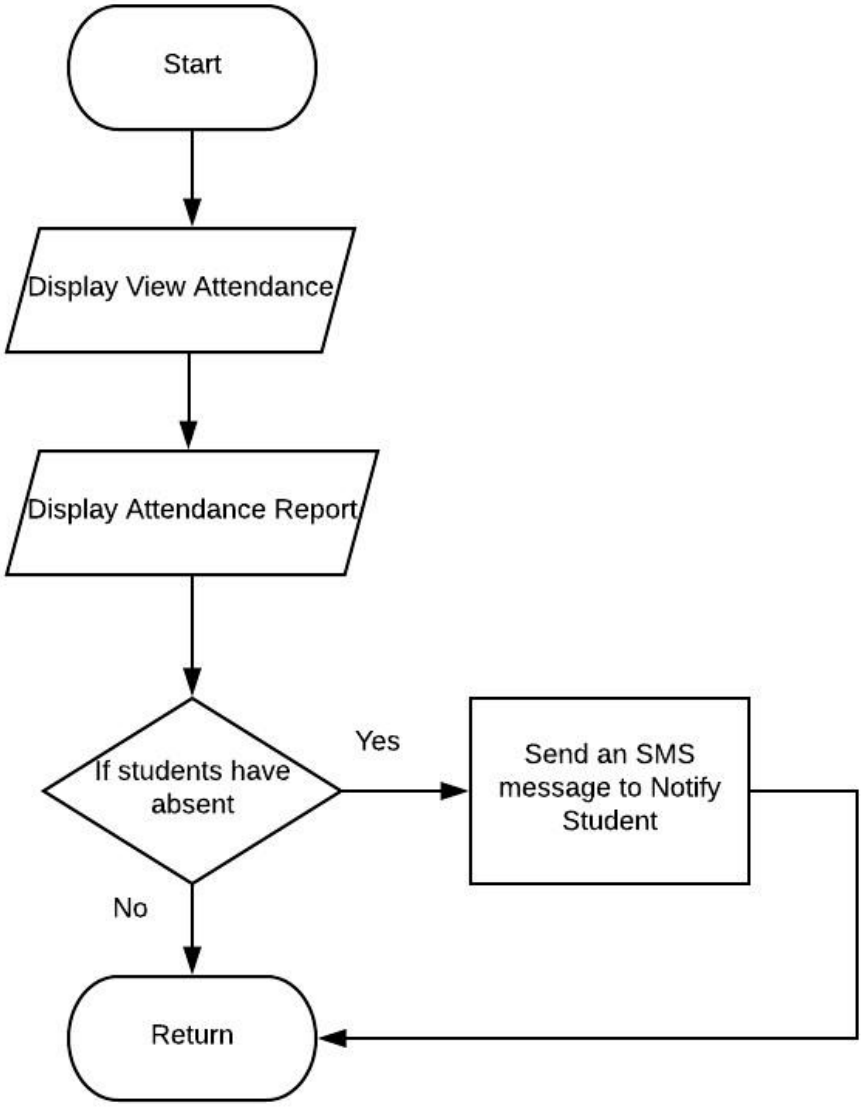


**Figure 9. View Attendance**

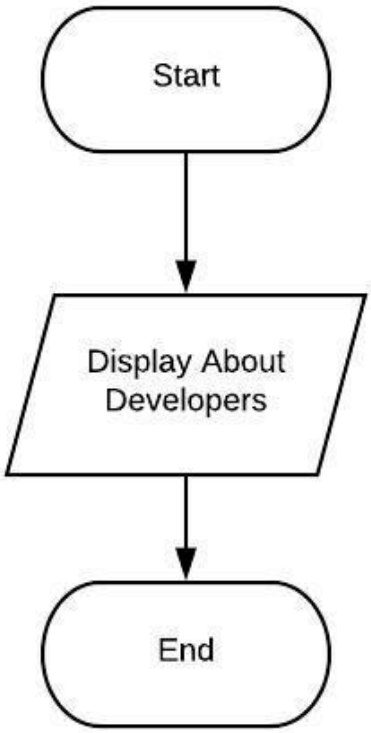


**Figure 10. About Developers**

## 4.0 RESULT AND DISCUSSION


**Figure 11. Splash Screen**

Upon clicking the icon, the system will display a starter screen containing the

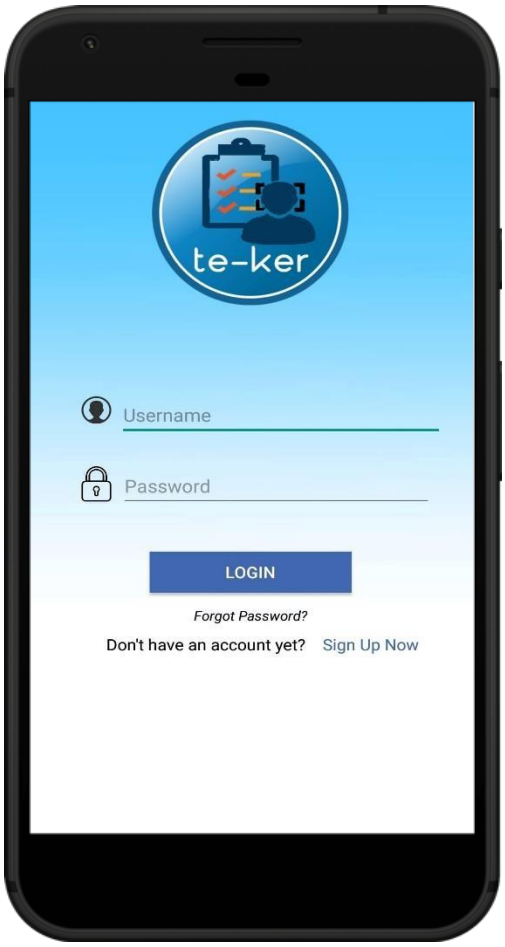


**Figure 12: Login**

Once the Splash Screen has successfully loaded, the user will be redirected to the Login Page. Wherein the user must have to enter information like username and password to access the application.

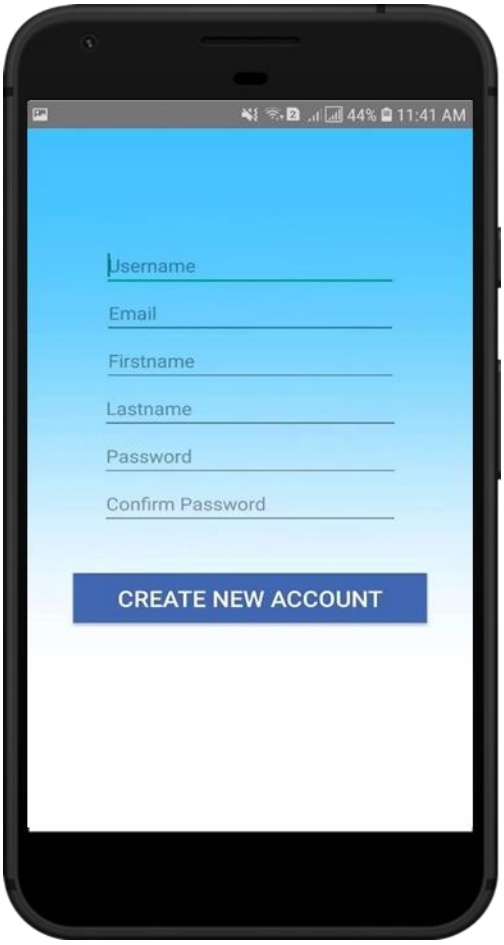


**Figure 13. Login (Continuation)**

If the user does not have an account yet, the user must sign up first to access the application.

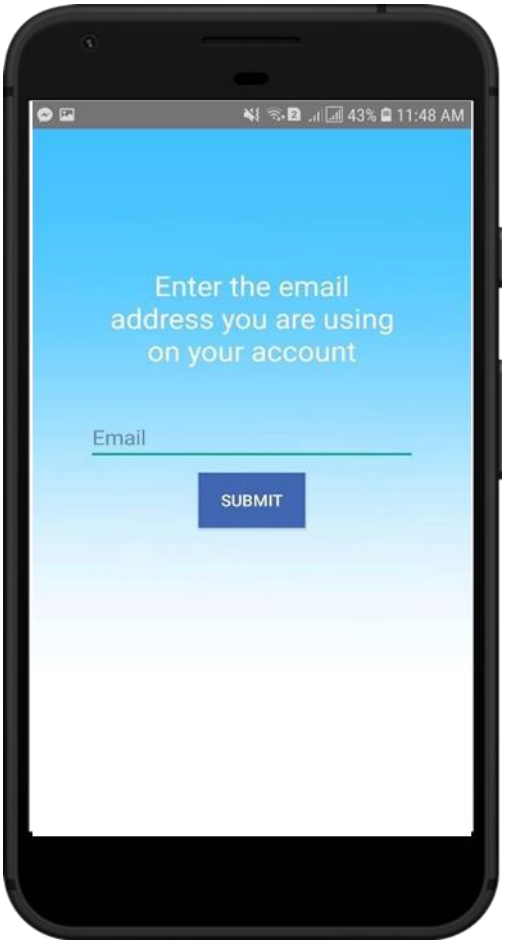


**Figure 14. Login (Continuation)**

Just in case the user forgot his/her password, the user may recover the account by just entering the registered email address and the application will send a code to reset the password.

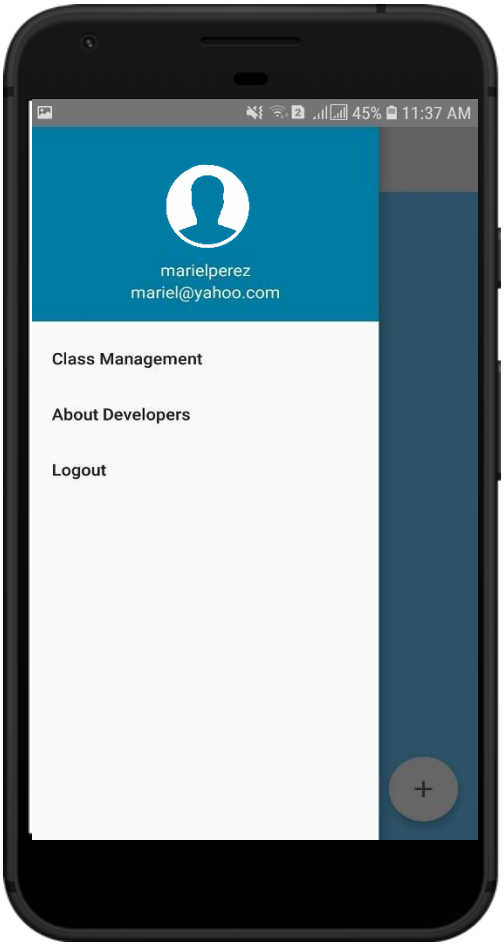


**Figure 15. Home Page**

The users will choose if they want to go to class management, about developers or if they want to exit from the application.

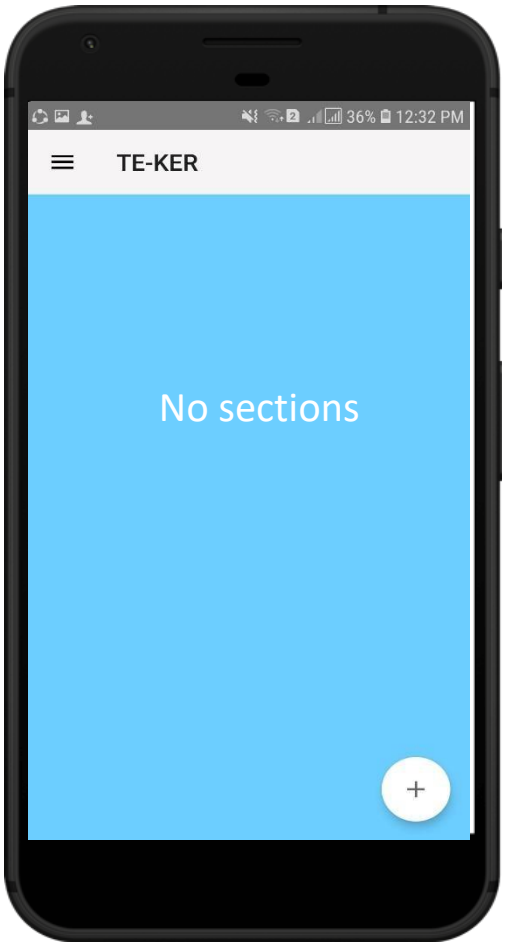


**Figure 16. Class Management**

Upon clicking the Class Management, it will display this figure which means that there is no section yet. If the user wants to add section, just click the button below.

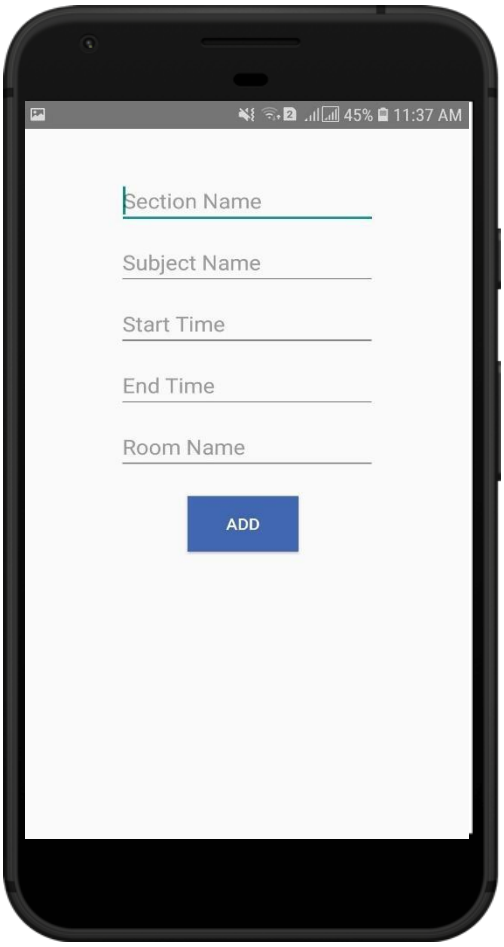


**Figure 17. Add Section**

By clicking the add button on the Class Management, the user will be redirected to this page, wherein the user must enter information on the given fields.

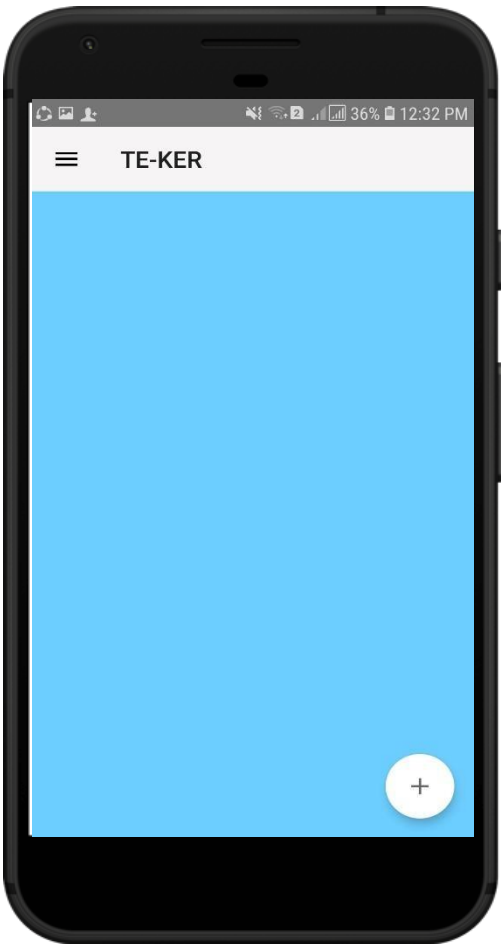


**Figure 18. Add Section (Continuation)**

After adding Section, the user can choose whether to check attendance of a particular section or exit the application.

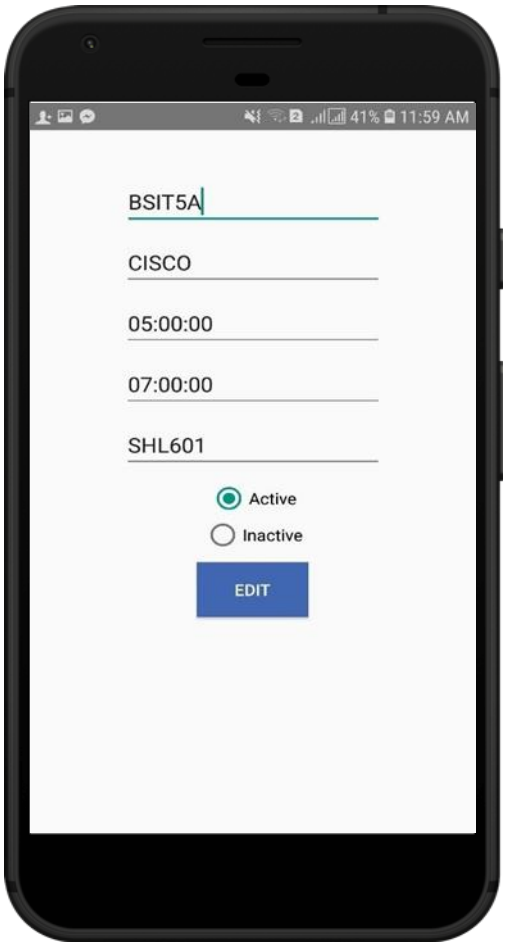


**Figure 19. Edit Section**

In case the user enters a wrong information, the user can edit it. The user can also select on the radio button if the section is still active or inactive.

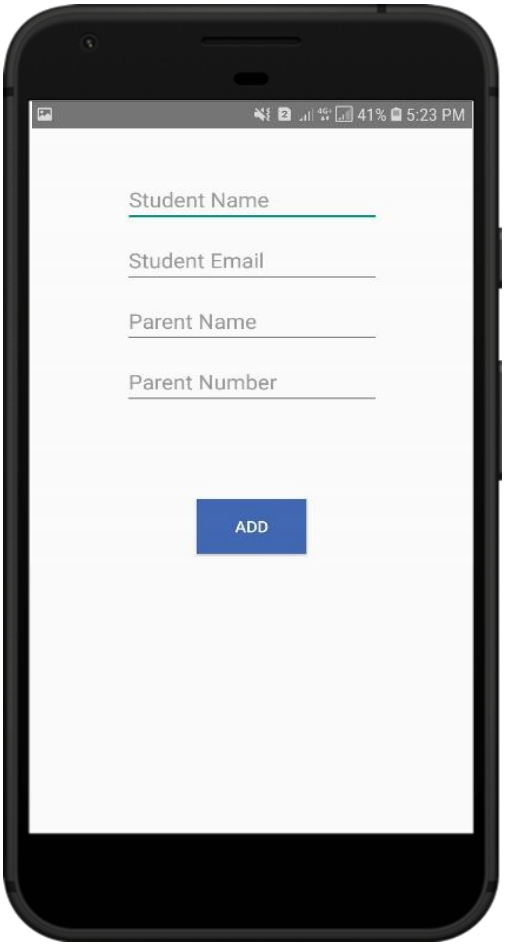


**Figure 20. Add Student**

The user can add student's information manually or import a file using excel format.


**Figure 21. Check Attendance**

Upon clicking the check button, it will display this figure which means the user can now check the attendance.

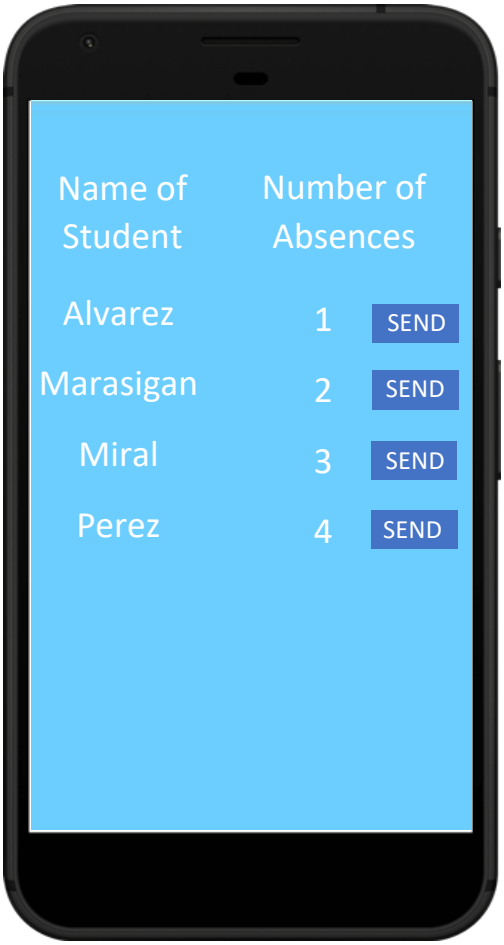


**Figure 22: View Attendance**

After checking the attendance, the user can now view if the student is absent or not. The user can also send a text message to the parent of the student.


**Figure 23. About Developers**

Accessible from the Home Screen, this will display the developers of the application.

# 5.0 SUMMARY, CONCLUSIONS AND RECOMMENDATION

## 5.1 Summary

This application is used for checking the attendance of the student using facial recognition, it was developed in an Android Studio platform and SQL for the database. A facial recognition system is a technology capable of identifying or verifying person from a digital image or video from a video source. There are multiple methods in which facial recognition systems work, but in general, they work by comparing selected facial features from given image with faces within a database. This application will allow the professors to check the attendance of their students automatically using facial recognition and it can only be installed in highest android version. The user can also need Internet connection to use the application.

## 5.2 Conclusions

Regarding the information that researchers gathered, the factors we round out, we have come up with TE-KER. It is a Mobile Application consisting of Checking the Attendance of the students. This would be a great help to the Teachers who are looking for an Attendance Checker of their students with just one click. It would be easier for them to check the attendance of their students and to notify them.

## 5.3 Recommendations

Recommendation based on the discovery and conclusion presented. The following recommendation is suggested: Teachers who used the app recommended having an online registration for the students to enroll themselves to a specific section. It is also recommended that the students must have grace period time because of the students being late. The users also recommend that the app should have a show password for the registration and login. For the future researchers, the current researchers recommend that the application can upload a file or list of students using excel format.

**REFERENCES**

[1] Android Based Management System [2015]

[2] Gatsheni, B.N, R.B Kuriakose, and F. Aghdasi. Automating a student class attendance register using radio frequency identification in South Africa. In Mechatronics, ICM2007 4th IEEE International Conference on 2007

[3] Mahyidin, M.F Student Attendance Using RFID system. 2008 December 2012; Available from: http://umpir.ump.edu.my

[4]https://www.techopedia.com/definition/2953/mobile-application-mobile-app

[5]https://www.techopedia.com/definition/2953/mobile-application-mobile-app

[6]https://www.ukessays.com/essays/informationtechnology/literature-review-on-mobile-apps-information-technology-essay.php

[7]http://www.academia.edu/34737457/Attndance_Monitoring Sytem

[8]http://www.apjmr.com/wp-content/uploads/2015/12/APJMR-2015-3.4.4.22.pdf

[9]https://www.mnsu.edu/cetl/teachingresources/articles/classattendance.html

[10]https://www.researchgate.net/publication/321533249_Development_of_Smartphone-based_Student_Attendance_System

[11]http://www.apjmr.com/wp-content/uploads/2015/12/APJMR-2015-3.4.4.22.pdf

[12]http://ethesis.nitrkl.ac.in/5195/1/109CS0146.pdf

[13]https://ijarcce.com/upload/2018/january-18/IJARCCE%2032.pdf

[14]https://www.queppelin.com/2016/09/8-phases-of-mobile-app-development-lifecycle/

**APPENDICES**

**CODE LISTING**

**Main Activity**

```
public class MainActivity extends
AppCompatActivity {
   private static int SPLASH_TIME_OUT =
4000;
   @Override
   protected void onCreate(Bundle
savedInstanceState) {
      super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
      new Handler().postDelayed(new
Runnable(){
         @Override
         public void run(){
            ifLoggedIn();
         }
      },SPLASH_TIME_OUT);
   }
   @Override
   protected void onResume() {
      super.onResume();
      new Handler().postDelayed(new
Runnable(){
         @Override
         public void run(){
            ifLoggedIn();
         }
      },SPLASH_TIME_OUT);
   }
   private void ifLoggedIn() {
      boolean isLoggedIn =
AppPreference.getLogin(this);
      if(isLoggedIn == false) {
         Intent intent = new
Intent(MainActivity.this,
LoginActivity.class);
         startActivity(intent);
         finish();
      } else {
         Intent intent = new
Intent(MainActivity.this,
HomeActivity.class);
         startActivity(intent);
         finish();
      }
   }
}
```

**Login Activity**

```
public class LoginActivity extends
AppCompatActivity {
   Button btnLoginSignin;
   TextView textViewLoginForgot,
btnLoginSignup;
```

```
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

setContentView(R.layout.activity_login);
        initViews();
        initEvents();
        initPreferences();
    }
    private void initPreferences() {
        boolean checker =
AppPreference.getLogin(this);
        if (checker == true) {
            Intent intent = new
Intent(LoginActivity.this,
HomeActivity.class);
            startActivity(intent);
        }
    }
    private void initViews() {
        btnLoginSignin =
findViewById(R.id.btnLoginSignin);
        btnLoginSignup =
findViewById(R.id.btnRegisterSignup);
        editTextLoginUsername =
findViewById(R.id.editTextLoginUsername
);
        editTextLoginPassword =
findViewById(R.id.editTextLoginPassword)
;
        textViewLoginForgot =
findViewById(R.id.textViewLoginForgot);
    }

    private void initEvents() {
        userSignIn();
        userSignUp();
        forgotPassword();
    }

    private void forgotPassword() {

textViewLoginForgot.setOnClickListener(ne
w View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new
Intent(LoginActivity.this,
ForgotPasswordActivity.class);
                startActivity(intent);
            }
```

**Register Activity**

```
public class RegisterActivity extends
AppCompatActivity {
    EditText editTextPassword,
editTextCPassword, editTextFirstName,
editTextLastName, editTextEmail,
editTextUsername;
    Button btnRegisterSignup;
    String url;
```

```
    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

setContentView(R.layout.activity_register);
        initViews();
        userRegister();
    }

    private void initViews() {
        editTextCPassword =
findViewById(R.id.editTextCPassword);
        editTextEmail =
findViewById(R.id.editTextEmail);
        editTextFirstName =
findViewById(R.id.editTextFirstName);
        editTextLastName =

findViewById(R.id.editTextLastName);
        editTextPassword =
findViewById(R.id.editTextPassword);
        editTextUsername =
findViewById(R.id.editTextUsername);
        btnRegisterSignup =
findViewById(R.id.btnRegisterSignup);
    }

    private void userRegister() {
        url = "https://te-
ker.000webhostapp.com/api/v1/register";

btnRegisterSignup.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                final String fname =
editTextFirstName.getText().toString();
                final String lname =
editTextLastName.getText().toString();
                final String password =
editTextPassword.getText().toString();
                final String email =
editTextEmail.getText().toString();
                final String username =
editTextUsername.getText().toString();
                final String cpassword =
editTextCPassword.getText().toString();
        StringRequest stringRequest = new
StringRequest(Request.Method.POST, url,
new Response.Listener<String>() {
                    @Override
                    public void onResponse(String
response) {
                        try {

if(cpassword.equals(password)) {
                            JSONArray jsonArray =
new JSONArray(response);
                            JSONObject jsonObject
= jsonArray.getJSONObject(0);
                            String message =
```

```
jsonObject.getString("message");
                    String code =
jsonObject.getString("code");

Toast.makeText(RegisterActivity.this,
message, Toast.LENGTH_LONG).show();
                    if(code.equals("200")){
                        Intent intent = new
Intent(RegisterActivity.this,
LoginActivity.class);
                        startActivity(intent);
                        finish();
                    } else
if(code.equals("400")) {

editTextCPassword.setText("");
                    } else
if(code.equals("500")) {

editTextCPassword.setText("");

editTextEmail.setText("");

editTextFirstName.setText("");

editTextLastName.setText("");

editTextPassword.setText("");
                    }
                } else {
Toast.makeText(RegisterActivity.this,
"Password don't match!",
Toast.LENGTH_LONG).show();
                    }

                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        }, new
```

**Home Activity**

```
public class HomeActivity extends
AppCompatActivity {
    private DrawerLayout mDrawerLayout;
    private String user_id, url;
    FloatingActionButton fab;
    LinearLayout linearLayout;
    TextView textViewNoSections;

    private RecyclerView mList;

    private LinearLayoutManager
linearLayoutManager;
    private DividerItemDecoration
dividerItemDecoration;
    private List<Section> movieList;
    private RecyclerView.Adapter adapter;

    @Override
```

```
   protected void onCreate(Bundle
savedInstanceState) {
       super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_home);
       linearLayout =
findViewById(R.id.main_layout);
       mList = findViewById(R.id.main_list);
       mList.setVisibility(View.VISIBLE);
       movieList = new ArrayList<>();
       adapter = new
SectionAdapter(getApplicationContext(),mo
vieList);
       linearLayoutManager = new
LinearLayoutManager(this);
linearLayoutManager.setOrientation(Linear
LayoutManager.VERTICAL);
       dividerItemDecoration = new
DividerItemDecoration(mList.getContext(),
linearLayoutManager.getOrientation());


       mList.setHasFixedSize(true);


mList.setLayoutManager(linearLayoutMana
ger);


mList.addItemDecoration(dividerItemDecor
ation);
       mList.setAdapter(adapter);


       getData();
       changeNavHeader();
       initViews();
       initPreferences();
       initEvents();
   }
   private void changeNavHeader() {
       String name =
AppPreference.getName(this);
       String email =
AppPreference.getEmail(this);
       NavigationView navigationView =
findViewById(R.id.nav_view);
       View headerView =
navigationView.getHeaderView(0);
       TextView navName =
headerView.findViewById(R.id.nav_name);
       TextView navEmail =
headerView.findViewById(R.id.nav_email);
       navName.setText(name);
       navEmail.setText(email);
   }
   private void getData() {
       final ProgressDialog progressDialog =
new ProgressDialog(this);


progressDialog.setMessage("Loading...");
       progressDialog.show();
       url = "https://te-
ker.000webhostapp.com/api/v1/get-
sections";
       StringRequest stringRequest = new
StringRequest(Request.Method.POST, url,
```

```
new Response.Listener<String>() {
        @Override
        public void onResponse(String
response) {
            try {
                JSONArray jsonArray = new
JSONArray(response);
                if(response.contentEquals("[]"))
{
textViewNoSections.setVisibility(View.VISI
BLE);

Toast.makeText(HomeActivity.this, "No
sections uploaded yet",
Toast.LENGTH_LONG).show();
                } else {
                    for (int i = 0; i <
response.length(); i++) {
                        JSONObject jsonObject =
jsonArray.getJSONObject(i);
                        Section section = new
Section();
section.setID(jsonObject.getString("id"));
section.setName(jsonObject.getString("nam
e"));
section.setSubject(jsonObject.getString("sub
ject"));
section.setStart_time(jsonObject.getString("
start_time"));
section.setEnd_time(jsonObject.getString("e
nd_time"));
section.setRoom(jsonObject.getString("roo
m"));
section.setStatus(jsonObject.getString("statu
s"));
                        movieList.add(section);
                    }
                }
            } catch (JSONException e) {
                e.printStackTrace();
            }
            adapter.notifyDataSetChanged();
            progressDialog.dismiss();
        }
    }, new Response.ErrorListener() {
        @Override
        public void
onErrorResponse(VolleyError error) {

        }
    }){
        protected Map<String, String>
getParams()
        {
            Map<String, String>  params =
new HashMap<String, String>();
            params.put("user_id", user_id);
            return params;
        }
    };
MySingleton.getInstance(HomeActivity.this
).addToRequestQueue(stringRequest);
```

```
    }
    private void initViews() {
        textViewNoSections =
findViewById(R.id.textViewNoSections);
        fab = findViewById(R.id.fab);
    }

    private void initEvents() {
        initNavigationDrawer();
```

**Add Section**

```
public class AddSectionActivity extends
AppCompatActivity {
    EditText editTextSectionName,
editTextSubjectName, editTextStartTime,
editTextEndTime, editTextRoomName;
    Button btnAddSection;
    String url;
    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

setContentView(R.layout.activity_add_secti
on);
        initViews();
        initEvents();
    }
    private void initEvents() {
        url = "https://te-
ker.000webhostapp.com/api/v1/create-
section";
        btnAddSection.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                final String user_id =
AppPreference.getUserId(AddSectionActivi
ty.this);
                final String name =
editTextSectionName.getText().toString();
                final String subject =
editTextSubjectName.getText().toString();
                final String start_time =
editTextStartTime.getText().toString();
                final String end_time =
editTextEndTime.getText().toString();
                final String room =
editTextRoomName.getText().toString();
                StringRequest stringRequest =
new StringRequest(Request.Method.POST,
url, new Response.Listener<String>() {
                    @Override
                    public void onResponse(String
response) {
                        try {
                            JSONArray jsonArray =
new JSONArray(response);
                            JSONObject jsonObject =
jsonArray.getJSONObject(0);
```

```
                    String message =
jsonObject.getString("message");
                    String code =
jsonObject.getString("code");

Toast.makeText(AddSectionActivity.this,
message, Toast.LENGTH_LONG).show();
                    if(code.equals("200")) {
                        Intent intent = new
Intent(AddSectionActivity.this,
HomeActivity.class);
                        startActivity(intent);
                        finish();
                    }
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        }, new Response.ErrorListener() {
            @Override
            public void
onErrorResponse(VolleyError error) {
            }
        }){
            protected Map<String, String>
getParams()
            {
                Map<String, String>  params
= new HashMap<String, String>();
   params.put("user_id", user_id);
            params.put("name", name);
            params.put("subject", subject);
        params.put("start_time", start_time);
         params.put("end_time", end_time);
              params.put("room", room);
                  return params;
              }
            };
MySingleton.getInstance(AddSectionActivit
y.this).addToRequestQueue(stringRequest);
        }
      });
   }
   private void initViews() {
      editTextSectionName =
findViewById(R.id.editTextSectionName);
      editTextSubjectName =
findViewById(R.id.editTextSubjectName);
      editTextStartTime =
findViewById(R.id.editTextStartTime);
      editTextEndTime =
findViewById(R.id.editTextEndTime);
      editTextRoomName =
```

**Add Student Activity**

```
public class AddStudentActivity extends
AppCompatActivity {
   EditText editTextStudentName,
editTextStudentEmail, editTextParentName,
editTextParentNumber;
   Button btnAddStudent, btnImport;
   String url, section_id;
```

```
    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

setContentView(R.layout.activity_add_stude
nt);
        initViews();
        initEvents();
        initExtras();
    }
    private void initExtras() {
        Intent intent = getIntent();
        section_id =
intent.getStringExtra("section_id");
    }
    private void initEvents() {
 url = "https://te-
ker.000webhostapp.com/api/v1/create-
student";
      btnAddStudent.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                final String name =
editTextStudentName.getText().toString();
                final String email =
editTextStudentEmail.getText().toString();
                final String parent_name =
editTextParentName.getText().toString();
                final String parent_number =
editTextParentNumber.getText().toString();
                StringRequest stringRequest =
new StringRequest(Request.Method.POST,
url, new Response.Listener<String>() {
                    @Override
                    public void onResponse(String
response) {
                        try {
                            JSONArray jsonArray =
new JSONArray(response);
```

**Developers Activity**

```
public class DevelopersActivity extends
AppCompatActivity {

    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

setContentView(R.layout.activity_developer
s);
    }
}
```

**Developers Fragment**

```
public class DevelopersFragment extends
Fragment {
    @Nullable
    @Override
    public View onCreateView(@NonNull
```

```
LayoutInflater inflater, @Nullable
ViewGroup container, @Nullable Bundle
savedInstanceState) {
        return
inflater.inflate(R.layout.fragment_developer
s, null);
    }

    @Override
    public void onViewCreated(@NonNull
View view, @Nullable Bundle
savedInstanceState) {
        super.onViewCreated(view,
savedInstanceState);
    }
}
```

**Section Adapter**

```
public class SectionAdapter extends
RecyclerView.Adapter<SectionAdapter.Vie
wHolder> {
    private Context context;
    private List<Section> list;
    public SectionAdapter(Context context,
List<Section> list) {
        this.context = context;
        this.list = list;
    @Override
    public ViewHolder
onCreateViewHolder(ViewGroup parent, int
viewType) {
        View v =
LayoutInflater.from(context).inflate(R.layou
t.section_list, parent, false);
        return new ViewHolder(v);
    }
    @Override
    public void
onBindViewHolder(ViewHolder holder, int
position) {
        Section section = list.get(position);

holder.textViewSectionName.setText(sectio
n.getName());

holder.textViewSectionSubject.setText(secti
on.getSubject());
holder.textViewSectionStartTime.setText(se
ction.getStart_time());
holder.textViewSectionEndTime.setText(sec
tion.getEnd_time());
holder.textViewSectionRoom.setText(sectio
n.getRoom());
    }
    @Override
    public int getItemCount() {
        return list.size();
    public class ViewHolder extends
RecyclerView.ViewHolder implements
View.OnClickListener,
View.OnLongClickListener{
        public TextView
textViewSectionName,
```

```
textViewSectionSubject,
textViewSectionStartTime,
textViewSectionEndTime,
textViewSectionRoom;
        public ViewHolder(View itemView) {
            super(itemView);
            textViewSectionName =
itemView.findViewById(R.id.textViewSecti
onName);
            textViewSectionSubject =
itemView.findViewById(R.id.textViewSecti
onSubject);
            textViewSectionStartTime =
itemView.findViewById(R.id.textViewSecti
onStartTime);
            textViewSectionEndTime =
itemView.findViewById(R.id.textViewSecti
onEndTime);
            textViewSection
```

**Student Adapter**

```
public class StudentAdapter extends
RecyclerView.Adapter<StudentAdapter.Vie
wHolder> {

    private Context context;
    private List<Student> list;

    public StudentAdapter(Context context,
List<Student> list) {
        this.context = context;
        this.list = list;
    }

    @Override
    public StudentAdapter.ViewHolder
onCreateViewHolder(ViewGroup parent, int
viewType) {
        View v =
LayoutInflater.from(context).inflate(R.layou
t.student_list, parent, false);
        return new
StudentAdapter.ViewHolder(v);
    }
    @Override
    public void
onBindViewHolder(StudentAdapter.ViewH
older holder, int position) {
        Student student = list.get(position);

holder.textViewStudentName.setText(stude
nt.getName());
    }

    @Override
    public int getItemCount() {
        return list.size();
    }

    public class ViewHolder extends
RecyclerView.ViewHolder implements
```

```
View.OnClickListener,
View.OnLongClickListener {
        public TextView
textViewStudentName;

        public ViewHolder(View itemView) {
            super(itemView);
            textViewStudentName =
itemView.findViewById(R.id.textViewStud
entName);
            itemView.setOnClickListener(this);

itemView.setOnLongClickListener(this);
        }
        @Override
        public void onClick(View v) {
```

**Edit Section**

```
            public class EditSectionActivity
extends AppCompatActivity {
EditText editTextEditSectionName,
editTextEditSubjectName,
editTextEditStartTime,
editTextEditEndTime,

editTextEditRoomName;

Button btnEditSection;
    RadioButton radioButtonActive,
radioButtonInactive;
    String section_id, name, subject,
start_time, end_time, room, status, url;
    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

setContentView(R.layout.activity_edit_secti
on);
        initExtras();
        initViews();
initEvents();
    }
private void initEvents() {

editTextEditSectionName.setText(name)
editTextEditSubjectName.setText(subject);

editTextEditStartTime.setText(start_time);

editTextEditEndTime.setText(end_time);
editTextEditRoomName.setText(room);
if(status == "0") {

radioButtonInactive.setChecked(true);
        } else {
            radioButtonActive.setChecked(true);
```

**Attendance Adapter**

```
public AttendanceAdapater(Context context,
List<Attendance> list) {
        this.context = context;
        this.list = list;
    }
```

```
@Override
   public AttendanceAdapater.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
       View v = LayoutInflater.from(context).inflate(R.layout.attendance_list, parent, false);
       return new AttendanceAdapater.ViewHolder(v);
   }
@Override
   public void onBindViewHolder(AttendanceAdapater.ViewHolder holder, int position) {
       Attendance attendance = list.get(position);

       holder.textViewAttendanceName.setText(attendance.getName());

       holder.textViewAttendance.setText(attendance.getTotal());
initViews();
      initEvents();
   }    private void initViews() {
      editTextFPEmail = findViewById(R.id.editTextFPEmail);
      btnFPEmail = findViewById(R.id.btnFPEmail);
   }
   private void initEvents() {
      sumbitFP();
   }
private void sumbitFP() {
      url = "https://te-ker.000webhostapp.com/api/v1/check-email";
      btnFPEmail.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View view) {
```

# User's Manual

1. Launch the Google Play Store application on your mobile device.
2. Search for the TE-KER and download the application
3. Launch the application by tapping the TE-KER icon on the application drawer.
4. Create an account and input all the required informations
5. After creating an account the user will be redirected to the Home Page. Tap the Class Management button and then add the section the user wants to add.
6. After adding section, the user can add the enrolled students to the specific section
7. Click the Check button to check the attendance of the students
8. After checking attendance, the user can click the view button to view the absent in the class and can notify them by sending a message
9. Click the About Developers button to know the Developers information
10. Tapping the Logout button on the main menu will close the application.