**PRESENT!**

**A Mini-Project Report**

**Under**

**IMPLEMENTATION OF TECHNOLOGY**

***Submitted by***

**ANYUN SHARMA (N030)**

**PUNEET VARMA (N038)**

**PRITEE SAROJ (N041)**

**MANNAT BAGGA (N042)**

***Under The Guidance Of***

**PROF. RATNESH CHATURVEDI**

***in partial fulfillment for the award of the degree***

***of***

**MBA TECH**

**IN**

**COMPUTER ENGINEERING**

**at**



**MPSTME NMIMS VILE PARLE**

**APRIL 2016**

**CERTIFICATE**

This is to certify that the project entitled **PRESENT!** is the bonafide work carried out by **ANYUN SHARMA, PUNEET VARMA, PRITEE SAROJ, and MANNAT BAGGA** B.Tech (Computer Engineering), MPSTME (NMIMS), Mumbai, during the fourth semester of the academic year 2015-2016, in fulfillment of the requirements for the award of the Degree of Bechelors of Technology as per the norms prescribed by NMIMS. The project work has been assessed and found to be satisfactory.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Ratnesh Chaturvedi

Internal Mentor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Examiner 1 Examiner 2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dean

Dr. Sharad Y. Mhaiskar

**DECLARATION**

We, ANYUN SHARMA, PUNEET VARMA, PRITEE SAROJ, and MANNAT BAGGA B.Tech (Computer Engineering), semester- IV, understand that plagiarism is defined as anyone or combination of the following:

1. Un-credited verbatim copying of individual sentences, paragraphs or illustration (such as graphs, diagrams, etc.) from any source, published or unpublished, including the internet.

2. Un-credited improper paraphrasing of pages paragraphs (changing a few words phrases, or rearranging the original sentence order)

3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did wrote what.

4. We have made sure that all the ideas, expressions, graphs, diagrams, etc., that are not a result of our work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.

5. We affirm that no portion of my work can be considered as plagiarism and we take full responsibility if such a complaint occurs. We understand fully well that the guide of the seminar/ seminar report may not be in a position to check for the possibility of such incidences of plagiarism in this body of work.

Signature of the Students:

\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

Name: \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

Roll No. \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

Place: Mumbai

Date: April 2016

**ACKNOWLEDGMENT**

We take this opportunity to express our sincere deference and gratitude to Prof. Ratnesh Chaturvedi for his constant guidance and motivation provided during the course of this work. It was a unique privilege to work under his valuable guidance and supervision.

We would like to express our gratitude towards our parents and friends for their kind co-operation and encouragement which helped us in completion of this project.

Our thanks and appreciations also go to our batch mates in developing the project and people who have willingly helped us out with their abilities.

**Table of contents**

**CHAPTER NO. TITLE PAGE NO.**

List of Figures 6

Abbreviations 7

Abstract 8

1. INTRODUCTION 9

1.1 Project Overview 12

1.2 Hardware Specification 13

1.3 Software Specification 14

2. REVIEW OF LITERATURE 15

3. ANALYSIS & DESIGN 16

4. METHODS IMPLEMENTED 25

5. RESULTS & DISCUSSION 26

6. CONCLUSION & FUTURE SCOPE 27

REFERENCES 28

**List of Figures**

**CHAPTER NO. TITLE PAGE NO.**

1. INTRODUCTION

Fig 1.1 Android application 9

Fig1.2 icon 11

3. ANALYSIS & DESIGN

Fig 3.1 Application icon on phone screen 17

Fig 3.2 application screen 1 18

Fig 3.3 application screen 2 19

Fig 3.4 application screen 3 20

Fig 3.5 application screen 4 21

Fig 3.6 application screen 5 22

Fig 3.7 application screen 6 23

Fig 3.8 application screen 7 24

\*\*\*\*\*\*\*\*\*\*\*\*\*

*.*

**Annexure VII**

**Abbreviations**

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| ADB | Android Debugging Bridge |
| ADT | Android Development Tools |
| Android SDK | Android Software Development Kit |
| ART | Android Runtime |
| CPU | Central Processing Unit |
| GPU | Graphic Processing Unit |
| IPS | In-Plane Switching |
| LCD | Liquid Crystal Display |
| RAM | Random Access Memory |
| SQL | Structured Query Language |
| XML | Extensible Markup Language |

**ABSTRACT**

Almost all academic institutions have a compulsion of attendance to appear for the final exams. If this attendance criterion is not met, the students are not allowed to appear for the exam. Though because of some valid reason students do miss class. So it becomes very important for students to keep a record of their attendance.

**Present!** enables a student to keep a record of their attendance on this application. It asks for your time table, according to which it asks whether you have attended all the lectures of that day, at the end of the day. It also distinguishes between lectures, tutorials and practicals. It shows the number of lectures attended in terms of percentage. Thus making it user-friendly. It also shows subject vise attendance percentage. If practical is missed, it reminds to compensate those practicals. As we are students too,this application is by the students to the students.

**INTRODUCTION**

**Introduction to Android:**

Android is an open source and Linux-based operating system for mobile devices such as smart phones and tablet computers. Android was developed by the Open Handset Alliance, led by Google, and other companies.

Android provides a rich application framework that allows you to build innovative apps and games for mobile devices in a Java language environment.

Android Architecture:

Android operating system is a stack of software components which is roughly divided into five sections and four main layers as shown below in the architecture diagram.

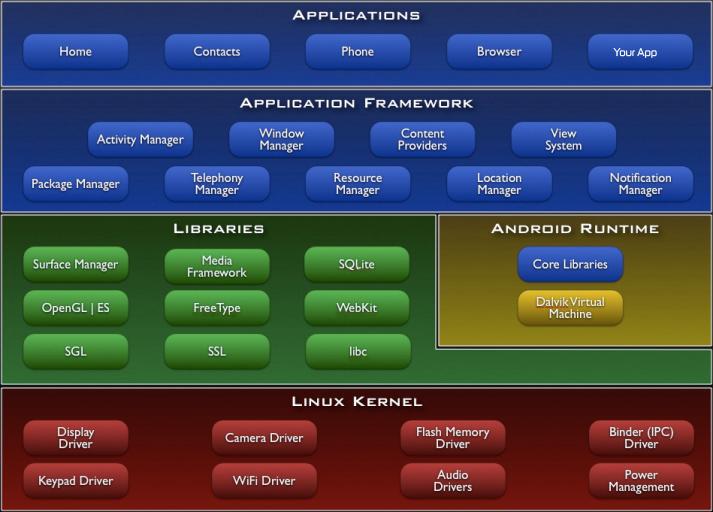


Fig 1.1 Android application

Android gives you a world-class platform for creating apps and games for Android users everywhere, as well as an open marketplace for distributing to them instantly. Android operating system is a stack of software components which is roughly divided into five sections and four main layers as shown below in the architecture diagram.

You will find all the Android application at the top layer. You will write your application to be installed on this layer only. Examples of such applications are Contacts Books, Browser, Games etc. These comprise both the native applications provided with the particular Android implementation (for example web browser and email applications) and the third party applications installed by the user after purchasing the device. Applications created by third party users or developers will be installed here.

**Introduction to Application:**

**Present!** is an application that enables students to keep a record of their attendance by simply creating an ID on the app. It is an attendance management application specifically designed for students. It asks for your name and criteria of attendance i.e. percentage of attendance compulsory as input from user friendly interface.

In many colleges students are required to fulfil the attendance criteria. If they fail to do so, they are blacklisted and exempted to appear for semester end exams wasting an entire year. There are various reasons because of which students might miss their classes. If a student is sick or have to prepare for tests, there is nothing that they can do to attend their classes. Colleges which follow attendance criteria strictly do not take any excuses this makes important to keep a record of attendance. Students often try doing it manually which is very difficult because then you have to recollect how many times the lectures have been missed. This application makes it very simple



Fig 1.2 Icon

The user needs to mention their name and attendance criteria. Then it asks for time table. The time table is simply made by adding the subjects and specifying whether it’s a practical, lecture or tutorial. As the timetable specifies the end of the day, it gives a notification to input the lectures attended. The lectures attended is marked by the user and it also shows cumulative attendance in terms of percentage subject vice. It also reminds of the practical missed so that it can be compensated. The attendance is marked by a single click using notification.

This application can be used by any student who is willing to maintain their attendance. The features that it provides helps students to better manage their attendance record.

**PROJECT OVERVIEW**

Present! is an attendance management application specially made for students. Present! Is one touch solution to all the attendance related problems faced by college students. Because of many reasons like need to study for upcoming tests, being sick, traffic, etc. students may miss their lectures. This application helps students to better manage their attendance and enable them to meet the desired attendance criteria and not fail for falling short of it.

.

**HARDWARE SPECIFICATION**

Present! is not a hardware intensive application, thus requires very basic level of hardware, as its application ranges across all devices, low end to high end, phones to tablets. Thus, the following are the minimum hardware specifications required to run the application:

* Processor:
  + Quad-core 1.2 GHz
  + PowerVR SGX 540 GPU.
* Memory:
  + 768 MB RAM
  + 1 GB of Flash Memory
  + Micro-SD card slot (Optional)
* Screen:
  + 3.5-inch LCD display
  + Capacitive or Resistive touch

During Development of the application, a one plus 2 model was used to test the application.

**SOFTWARE SPECIFICATION**

Present! is designed to work on Android 5.1.1, lollipop.

Following is the list of Android versions on which Present! can be used:

* Android 4.0 (Ice Cream Sandwich)
* Android 4.2 (Jellybean)
* Android 4.3 (Jellybean)
* Android 4.4 (KitKat)
* Android 5.0 (Lollipop)
* Android 5.0.2 (Lollipop)
* Android 5.1 (Lollipop)

This gives the application a broad platform, as these operating systems are in majority of smart phones being used by prospective users.

The application does not require internet connection.

**Review of Literature**

A literature review is a text of a scholarly paper, which includes the current knowledge including substantive findings, as well as theoretical and methodological contributions to a particular topic. Most often associated with academic-oriented literature, such as a thesis, dissertation or a peer-reviewed journal article, a literature review usually precedes the methodology and results section although this is not always the case. Literature reviews are also common in a research proposal or prospectus. Its main goals are to situate the current study within the body of literature and to provide context for the particular reader. Literature reviews are a basis for research in nearly every academic field.

**Java:**

Java is a set of several computer software and specifications developed by Sun. Microsystems, later acquired by Oracle Corporation, that provides a system for developing application software and deploying it in a cross-platform computing environment. Java is used in a wide variety of computing platforms from embedded devices and mobile phones to enterprise servers and supercomputers. While less common, Java applets run in secure, sandboxed environments to provide many features of native applications and can be embedded in HTML pages.

Writing in the Java programming language is the primary way to produce code that will be deployed as byte code in a Java Virtual Machine (JVM); byte code compilers are also available for other languages, including Ada, JavaScript, Python, and Ruby. In addition, several languages have been designed to run natively on the JVM, including Scala, Clojure and Groovy. Java syntax borrows heavily from C and C++, but object-oriented features are modeled after Smalltalk and Objective-C.Java eschews certain low-level constructs such as pointers and has a very simple memory model where every object is allocated on the heap and all variables of object types are references. Memory management is handled through integrated automatic garbage collection performed by the JVM.

**ANALYSIS & DESIGN**

During the conception of the application, the requirements were mapped out first. This included analyzing the need of this application, its features and figuring out a list of functions it would perform during operation. This largely consisted of the analysis.

After performing a complete analysis, the design of the application was pursued. Abnegate was designed using material design. Material design is a comprehensive guide for visual, motion, and interaction design across platforms and devices.

**Screenshots and working**

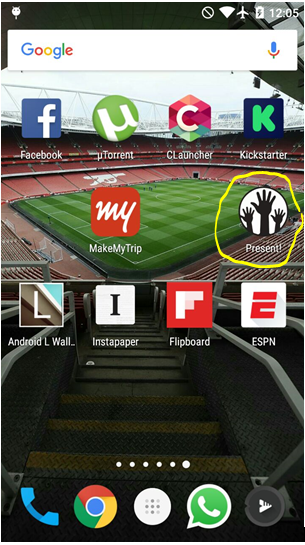


Fig 3.1 Application icon on phone screen



Fig 3.2 application screen 1

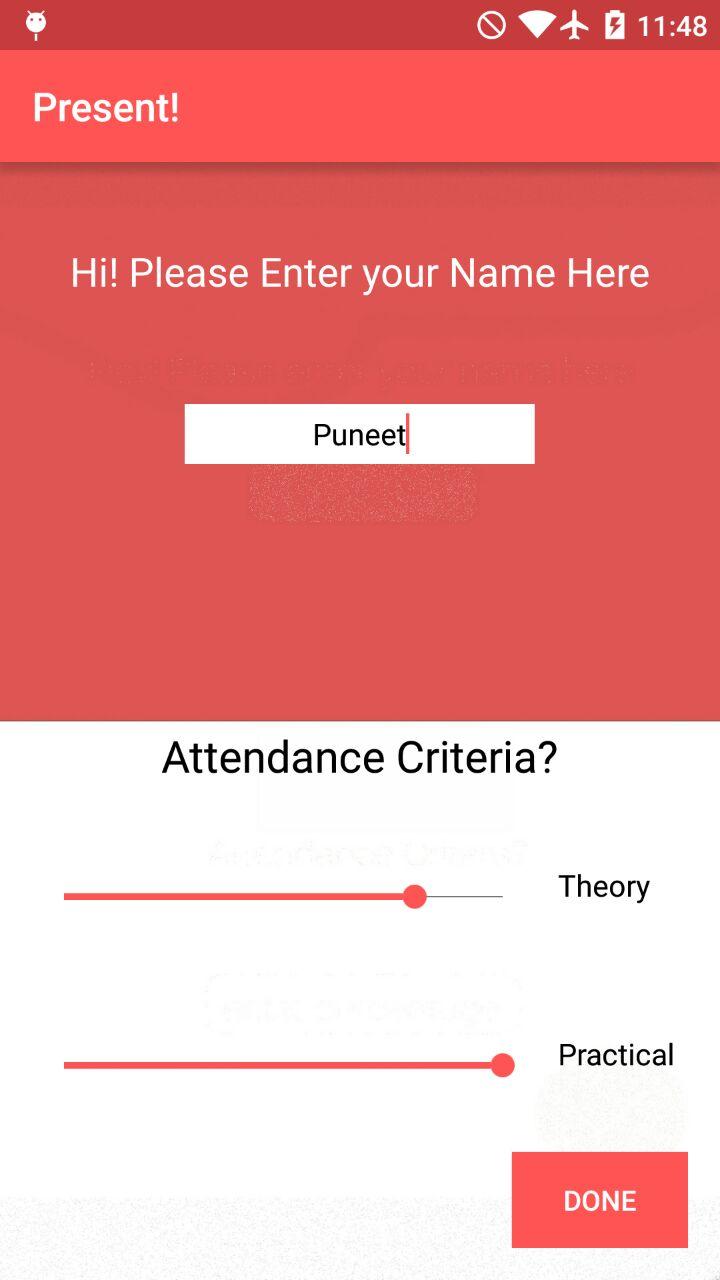


Fig 3.3 Application screen 2

home screen of application Present! It asks to enter your name and specify attendance criterion.

Once it is done click DONE button.

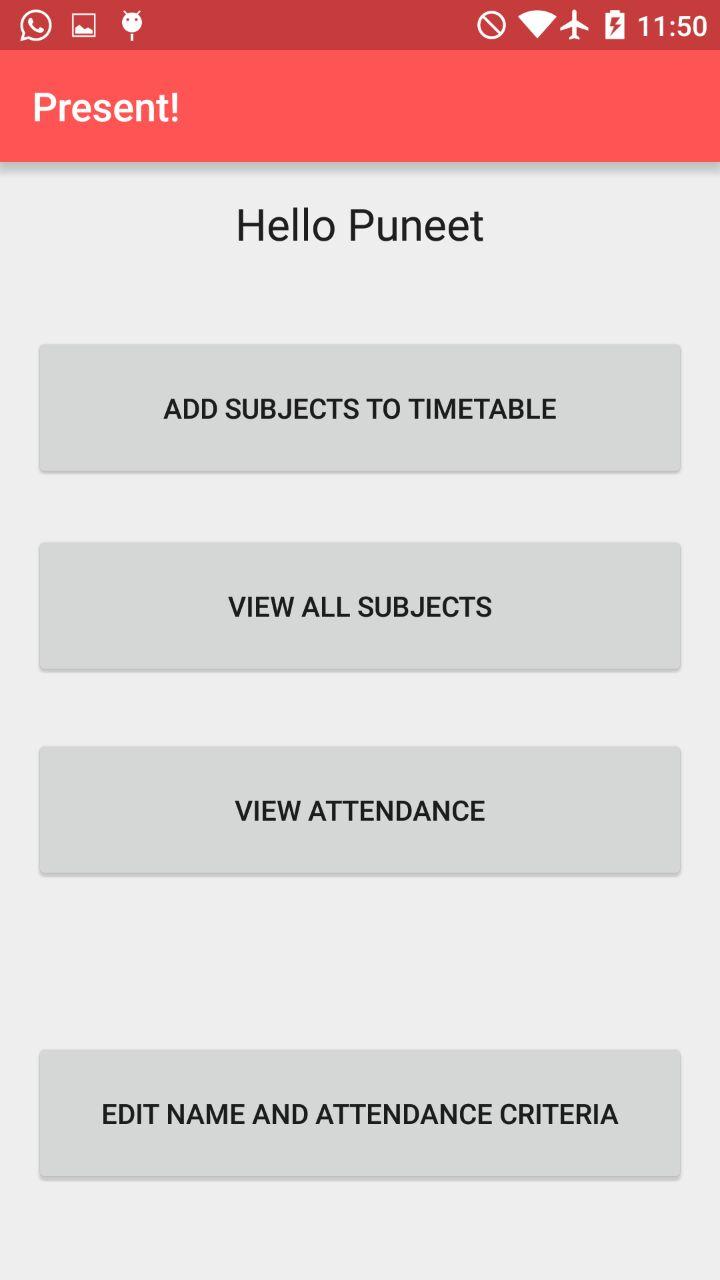


Fig 3.4 Application screen 3

The second screen has various options like add subjects to timetable, view all subject, view attendance and edit name and criterion. All of these options are self-explanatory. For the first time login add subject to timetable.

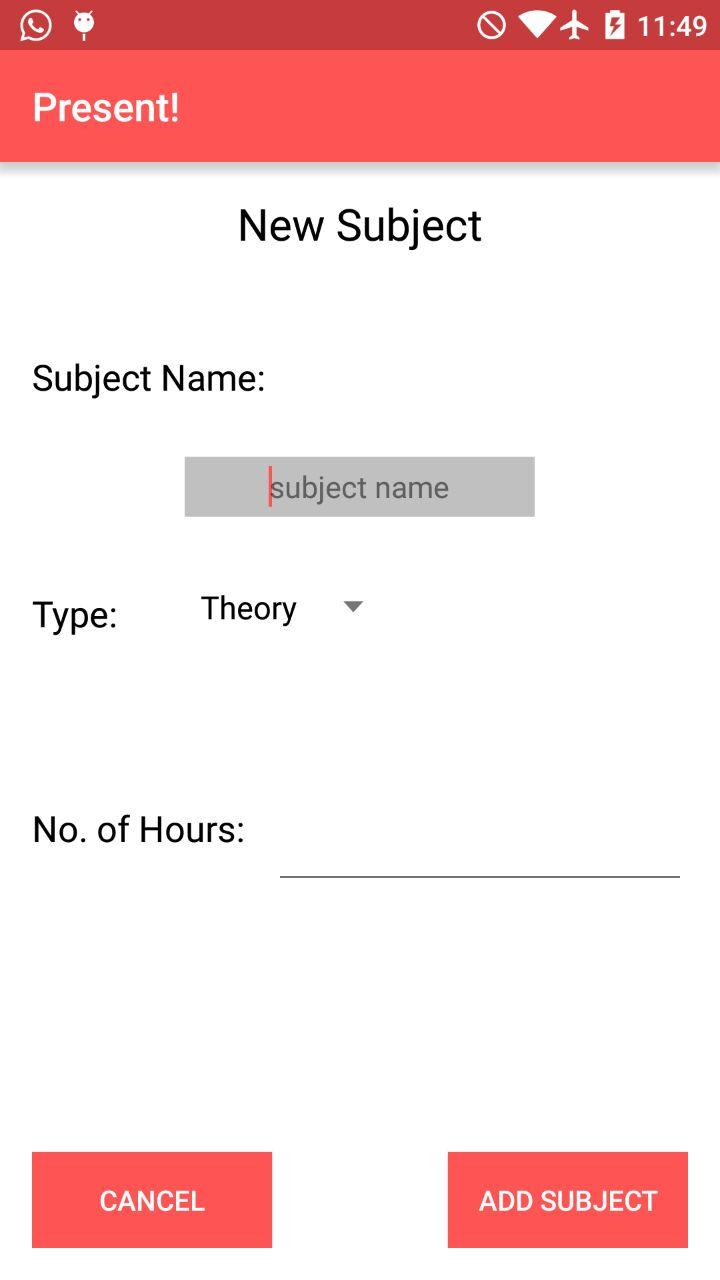


Fig 3.5 Application screen 4

This screen shows up when user clicks on add subject. Fill in the information as specified. Once done click on add subject.

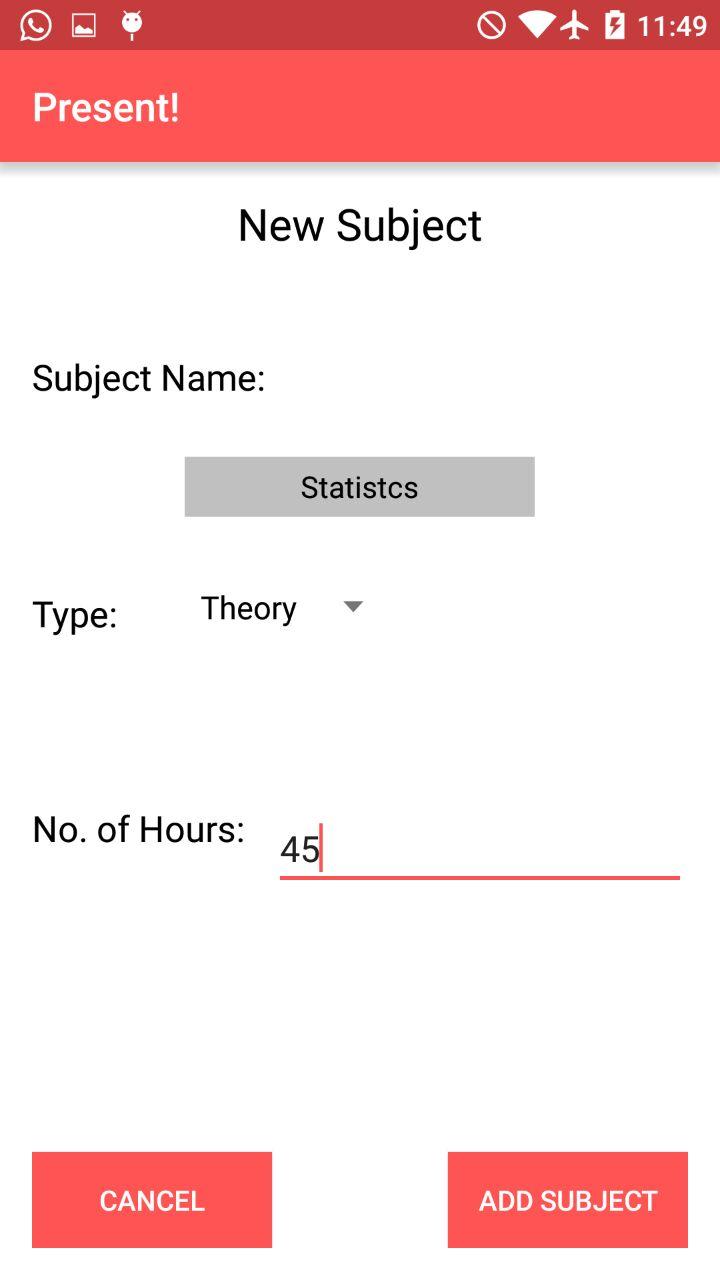


Fig 3.6 Application screen 5

Fill in the form as said and click on add subject.

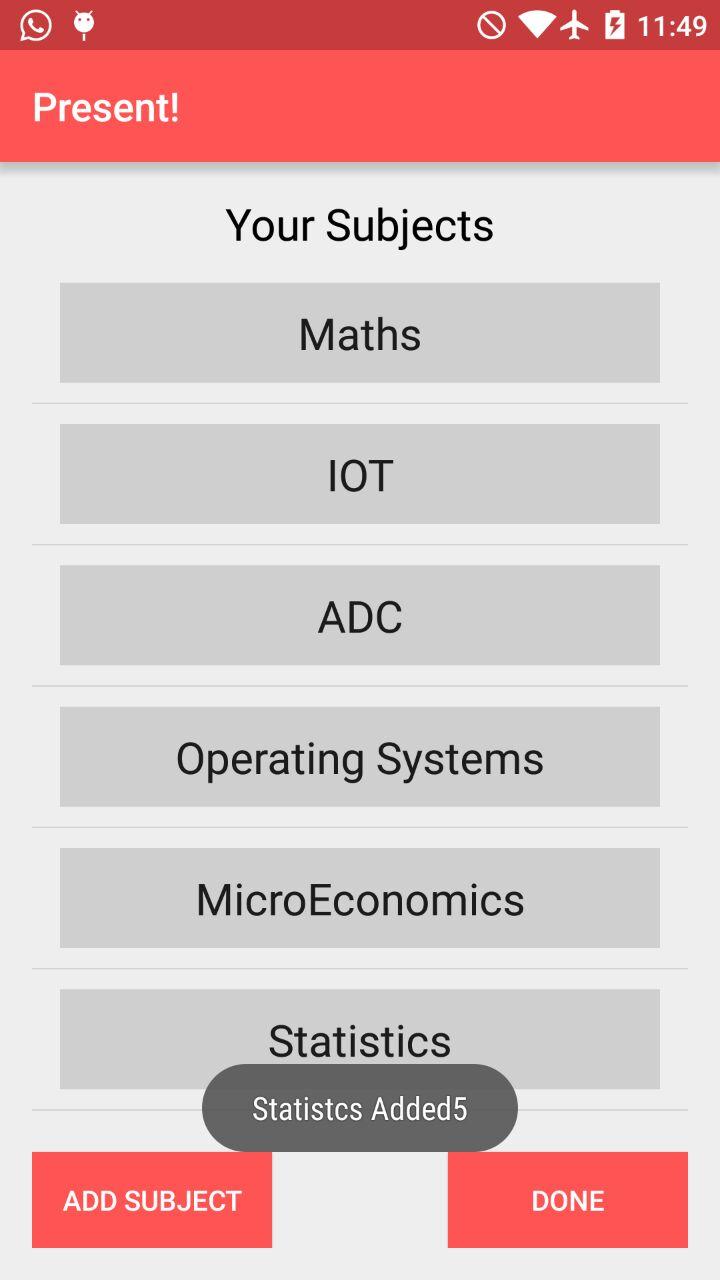


Fig 3.7 Application screen 6

The added subject appears in the timetable list.

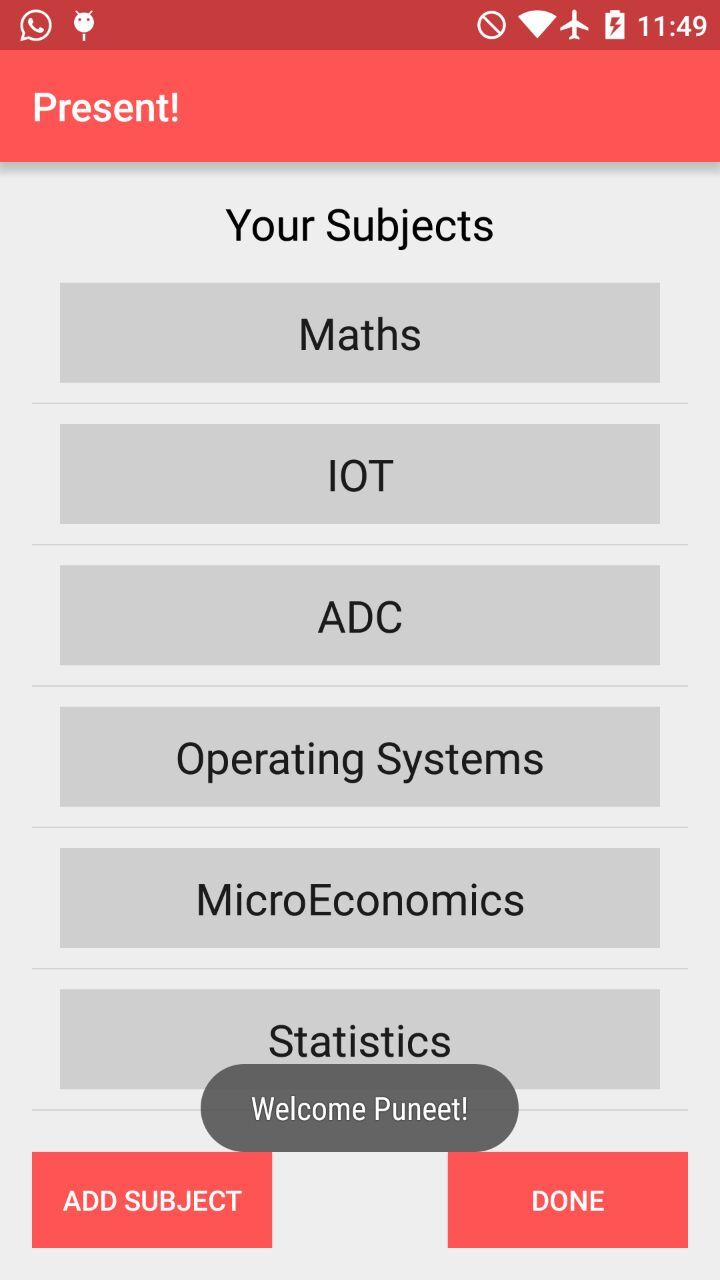


Fig 3.8 Application screen 7

If user selects view all subjects the above screen comes on the screen

**METHODS IMPLEMENTED**

**Android Studio:**

Android Studio is the official IDE for Android application development, based on IntelliJ IDEA. On top of the capabilities you expect from IntelliJ, Android Studio offers:

Flexible Gradle-based build system.

Build variants and multipleapk file generation.

Code templates to help you build common app features.

Rich layout editor with support for drag and drop theme editing lint tools to catch performance, usability, version compatibility, and other problems.

ProGuard and app-signing capabilities.

Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud.

Messaging and App Engine.

**RESULTS AND DISSCUSSION**

The main frame of the application has been created. A user can just need to specify the attendance criterion they have to follow and put in time table and the app is ready to use. It will eliminate the manual calculation of missed hours for each student for each subject and enable him to meet the desired attendance criteria and not fail for falling short of it. We are yet to take the App online. Every app needs to be updated with new features and we will continue to upgrade the app with new features and building more user friendly screens for this app.

**CONCLUSION & FUTURE SCOPE**

* The app is yet to be taken online.
* This app is designed specifically keeping students in mind.
* However, anyone who need to keep their attendance in check can use it.
* Knowing the extent of one’s absence is important so that we can be careful and not end up being held back a year due to something like not meeting the attendance criteria.
* More features like reminder of assignment or practical submission can be added.
* The app is long way from being complete and we will continue to put in effort.

**REFERENCES**

1. Videos provided in course
2. [www.youtube.com](http://www.youtube.com)
3. developer.android.com
4. [www.google.com](http://www.google.com)