MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT & ENGINEERING

Classify

A Mini-Project Report

Under

Implementation of Technology

Submitted by

Videet Parekh

Chaitanya Parsana

Vinay Pisharody

Devesh Ratho

Under The Guidance Of

Prof. Ratnesh Chaturvedi

in partial fulfillment for the award of the degree

of

Bachelor of Technology

IN

Computer Engineering

at

MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT AND ENGINEERING, NMIMS

May 2017

CERTIFICATE

This is to certify that the project entitled “Classify” is the bonafide work carried out by Videet Parekh, Chaitanya Parsana, Vinay Pisharody and Devesh Ratho of B.Tech (Computer Engineering), MPSTME (NMIMS), Mumbai, during the IV semester of the academic year 2014-2015, in partial fulfillment of the requirements for the award of the Degree of Bachelors of Technology as per the norms prescribed by NMIMS. The mini-project work has been assessed and found to be satisfactory.

Prof. Ratnesh Chaturvedi

Examiner 1

Examiner 2

Dean

Dr. Sharad Mhaiskar

DECLARATION

We, Videet Parekh, Chaitanya Parsana, Vinay Pisharody and Devesh Ratho, Roll No. E025, E027, E031, E034 respectively, of B.Tech (Computer Engineering), IV semester 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 from any source, published or unpublished, including the internet.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. (Source: IEEE, The institute, Dec. 2004)
4. I have made sure that all the ideas, expressions, graphs, diagrams, etc., that are not a result of my 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. I affirm that no portion of my work can be considered as plagiarism and I take full responsibility if such a complaint occurs. I understand fully well that the guide of the seminar/ project 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: Videet Parekh, Chaitanya Parsana, Vinay Pisharody and Devesh Ratho.
Roll No. E025, E027, E031, E034 (respectively)

Place:
Date:

Acknowledgement

We would like to show our gratitude to all the faculty that has taught us the subjects necessary for us to be able to complete this Project. We would like to thank Prof. Ratnesh Chaturvedi, who has left no stone unturned for imparting the knowledge of application development for Android Devices during the course of Implementation of Technology classes, in the best way possible. We would also like to thank the administration of this college and our Dean, DR. Sharad Mhaiskar for overseeing the functioning of this great institution without which our cause would no home.

Table of contents

List of Figures

List of Tables

Abbreviations

Abbreviation	**Description**
ADB	Android Debugging Bridge
ADT	Android Development Tools
Android SDK	Android Software Development Kit
ART	Android Runtime
CPU	Central Processing Unit
CR	Class Representative
GPU	Graphic Processing Unit
GUI	Graphic User Interface
HoD	Head of Department
IPS	In-Plane Switching
LCD	Liquid Crystal Display
RAM	Random Access Memory
SQL	Structured Query Language
SR	Student Representative
XML	Extensible Markup Language

Introduction

There is no unified system that exists currently for communication between students in a class. The parts of the system include Blackboard, WhatsApp etc. Students to need to keep a track of all the ongoing activities of the class using all the systems simultaneously. Blackboard shows the announcements by teachers. WhatsApp is used for instant communication and reminders of submissions.

Classify is built with the purpose of bridging that gap with the help of an active notification system at its core, which will notify all users of the application on any activity that is taking place on a class, university and committee level while also helping them put up requests for coaching on various subjects from their peers or seniors.

Project Overview

Classify is an android application which is built with the intention of having a common source of communication to communicate efficiently and to remove the discrepancies caused by the lack of effective communication. Adding to this, to have a request system to request fellow classmates for training in a particular subject or a particular topic as required. Having a notification system for institute level in a different window and having a notification system for the different events being held in the college.

It is designed keeping in mind the different levels of access that students, CRs, SRs, Administration (Dean, HoD, Course Coordinator) and Committees would require. Thus, the applications provides the following logins:

- Class level – Student/SR/CR
- Institute level –Dean/ HoD/Course Coordinator
- Events – Heads of the respective committees

Considering colleges that don’t have blackboard, this will serve as a means for communication between students in a class. Other than colleges, this can be implemented for a larger community by implementing it for an organization or a company.

Hardware Specification

Classify 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:
  - 1 GHz ARM Cortex A8 based CPU core
  - PowerVR SGX 540 GPU.
- Memory:
  - 512 MB of 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, OnePlus One and LG Nexus 4 were used to develop the application on mobiles and ASUS Google Nexus 7 was used to develop the application on a tablet. Following are the specifications:

- LG Nexus 4:
  - Processor:
    - Quad-core 1.5 GHz Krait Snapdragon APQ8064
    - Adreno 320 GPU
  - Memory:
    - 2GB of RAM
    - 16GB of Flash Memory
  - Screen:
    - 4.7-inch IPS LCD
- OnePlus One:
  - Processor:
    - 2.5 GHz Qualcomm Snapdragon 801
    - Adreno 330 GPU
  - Memory:
    - 3GB of RAM

        - 64GB of Flash Memory
    - Screen:
        - 5.5-inch IPS LCD display

- ASUS Google Nexus 7:
    - Processor:
        - 1.2 GHz NVIDIA Tegra 3 CPU
        - ULP GeForce GPU
    - Memory:
        - 1 GB of RAM
        - 16GB of Flash Memory
    - Screen:
        - 7-inch IPS LCD display

Software Specifications:

Classify is made on Android SDK API 22, which is primarily for android 5.0 Lollipop. This also has backward compatibility to previous APIs. Classify is designed to work on Android 4.0, Ice Cream Sandwich and above.

Apart from Android 4.0 and above, the application, like most android applications can run on the following Operating Systems:

- Blackberry OS 10.2
- Blackberry OS 11
- Sailfish OS
- Chrome OS
- Color OS
- IUNI OS
- YUN OS
- Nokia X mobile Platform
- Amazon Fire OS
- Tizen OS
- Android x86 (Intel Based devices)
- DuOS-M

Some of these are based on android, while some (BBos 10.2/11, Sailfish OS) are made compatible to run android applications.

Following is the list of Android versions on which Classify has been tested:

- 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.

Review of Literature

Database Management

A database management system (DBMS) is a collection of programs that enables you to store, modify, and extract information from a database. There are many different types of DBMSs, ranging from small systems that run on personal computers to huge systems that run on mainframes. The following are examples of database applications:

- Computerized library systems
- Automated teller machines
- Flight reservation systems
- Computerized parts inventory systems

For the purpose of this application, SQLite was used for all purposes of database management.

Java

Java is a general-purpose computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. Java code can run on all platforms that support Java without the need for recompilation.

The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licences.

Version History:

Major release versions of Java, along with their release dates:

- JDK 1.0 (January 21, 1996)
- JDK 1.1 (February 19, 1997)
- J2SE 1.2 (December 8, 1998)
- J2SE 1.3 (May 8, 2000)
- J2SE 1.4 (February 6, 2002)
- J2SE 5.0 (September 30, 2004)
- Java SE 6 (December 11, 2006)
- Java SE 7 (July 28, 2011)
- Java SE 8 (March 18, 2014)

Introduction to Android

Android is an operating system for mobile devices such as smartphones and tablet computers. It is developed by the Open Handset Alliance led by Google. It's is built on a Linux foundation. Google purchased the initial developer of the software, Android Inc., in 2005. The unveiling of the Android distribution on November 5, 2007 was announced with the founding of the Open Handset Alliance, a consortium of 84 hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices.

This alliance shares a common goal of fostering innovation on mobile devices and giving consumers a far better user experience than much of what is available on today's mobile platforms. By providing developers a new level of openness that enables them to work more collaboratively, Android will accelerate the pace at which new and compelling mobile services are made available to consumers. Android is often symbolized by the green robot to the right.

Android has evolved rapidly since its launch. Google has named all projects after a dessert. The main releases are listed below, this is nothing you have to memorize, and it’s just to illustrate the rapid pace of development and all the innovations. Android is developed "on Internet time", that is much faster than the old style of development (for example Windows releases which are typically several years apart).

The Android Open Source Project (AOSP) is tasked with the maintenance and further development of Android. Android consists of a kernel based on the Linux kernel, with middleware, libraries and APIs written in C and application software running on an application framework which includes Java-compatible libraries based on Apache Harmony. Android uses the Dalvik virtual machine with just-in-time compilation to run compiled Java code. Android has a large community of developers writing applications ("apps") that extend the functionality of the devices. Developers write primarily in a customized version of Java.

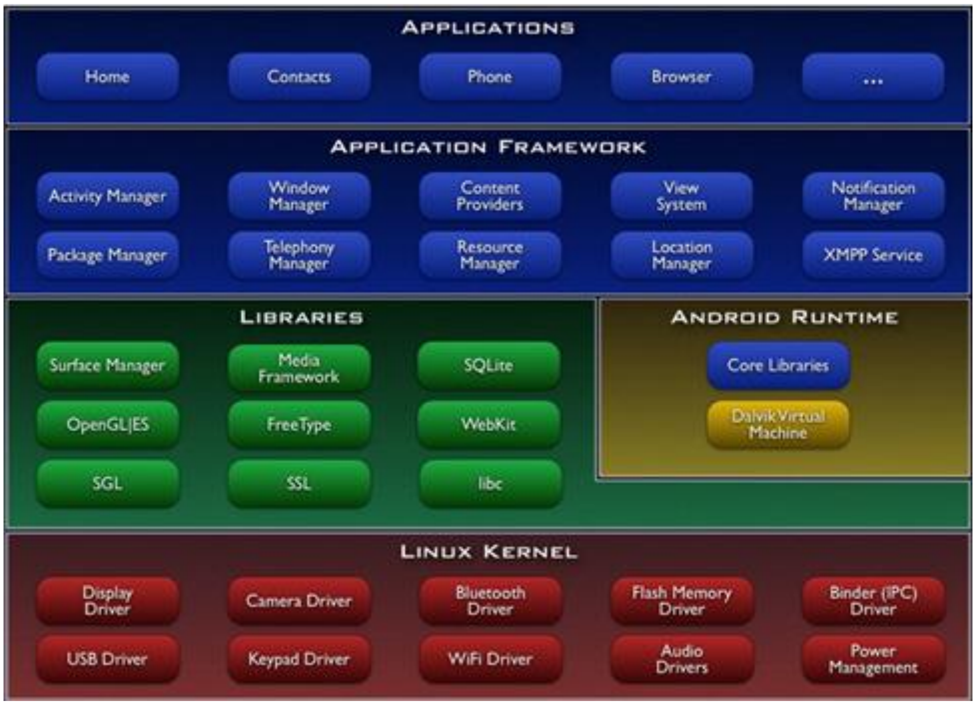


Fig. 2.1. Android system libraries

Material Design

Material design is a unified system of visual, motion, and interaction design that adapts across different devices. Material design is inspired by tactile materials, such as paper and ink. Material surfaces interact in a shared space. Surfaces can have elevation (z-height) and cast shadows on other surfaces to convey relationships. It is a design language developed by Google and announced at the Google I/O conference on June 25, 2014.

Analysis, Design and Implementation.

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. Classify is designed using a hybrid design of Holo Design and Material Design elements. Material Design was introduced very recently with the launch of Android 5.0 (Lollipop), whereas Holo design was used since Android 3.0 (Honeycomb). The design also involved the organization of the GUI and the application's databases.

Requirements:

When analyzing the requirement of the application, following points came to light for the need of Classify:

1. Blackboard's Mobile Client is not adopted as widely by the students.
2. Blackboard is cluttered with information, where notifications are missed out quite frequently by a student.
3. There is no central communication line piercing through all the hierarchy levels of the institution and people have to rely on word of mouth more often than not.
4. The announcements on class groups and forums are lost among a hoard of other messages of lesser significance.
5. There is no central portal for peers to ask each other for help in studies, and the potential of such sessions isn't completely used, as many people miss out on them due to lack of notification in any form.
6. The existing methods of announcement and notifications don't reach the audience fast enough, in a noticeable manner.
7. Students don't check their emails very frequently, where the existing mechanism of updates relies on.
8. There is no way for committees to make students aware of their events apart from posters on the walls and forwarded messages, both of which take up a lot of time.

Features:

1. **Notify lectures for the next day:**

   The Application is designed to notify the students of the lectures taking place every day on their notification panel, so they don't have to refer the time table again and again.

2. **Notify rescheduling of lectures:**

   Similarly, if the lecture has been rescheduled, it will notify the users. This is to avoid confusion among students and teachers and to utilize time properly. Both the first and the second features are accessible by the CR/SR and the Course Coordinator.

3. **Notify cancelations of lectures:**

   It notifies the cancellation of lectures to students and the Course Coordinator with its central notification system.

4. **Notify submission dates and other important dates:**

   It notifies the students about the important dates for various academic submissions, viva-voce and exams.

5. **Notify urgent submissions to the user:**

   It notifies the user about any urgent submissions that are to be done to any teacher or the administration.

6. **Request for training for a topic.**

   It allows the students to seek tuition from their peers for certain topics and/or subjects.

7. **Different window for event notifications.**

   It has a dedicated window for events of several college committees to post their announcements and distribute the messages to all the users, increasing their reach manifold.

Functions:

There are broadly 4 functions of Classify. They are as follows:

Class Notifications:

Class Notifications can be posted by the CR, SR and Course Coordinator. It can range from rescheduling or cancelling of lectures to assignment submissions. It has been implemented using a slide out menu which is toggled by the button on top.

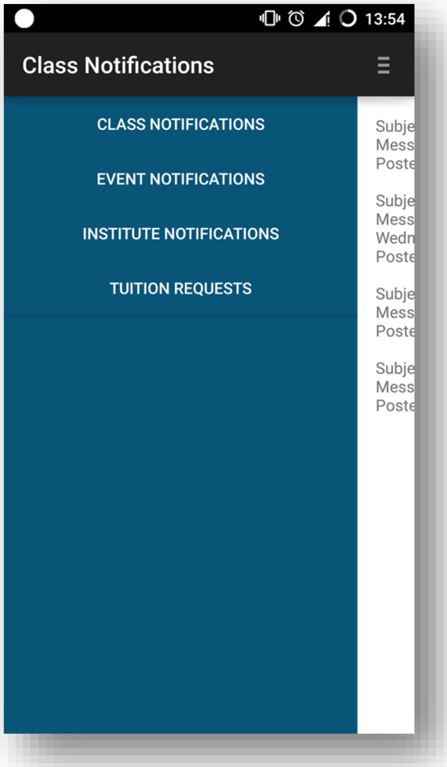
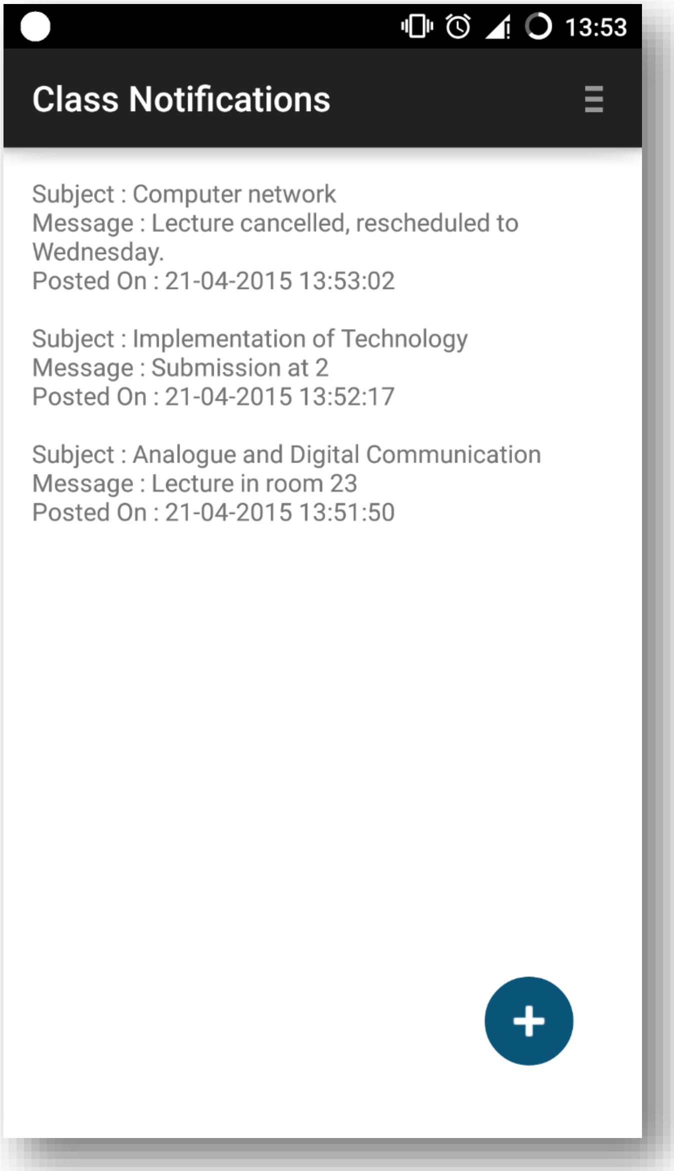


Fig.3.1: Slide out page for the Class Notification Tab.

After going to the Class Notifications window, the summary of all class notifications is shown, with the newest one on top.

There is also an FAB, which can be used to add new notifications.



Subject : Computer network
Message : Lecture cancelled, rescheduled to Wednesday.
Posted On : 21-04-2015 13:53:02

Subject : Implementation of Technology
Message : Submission at 2
Posted On : 21-04-2015 13:52:17

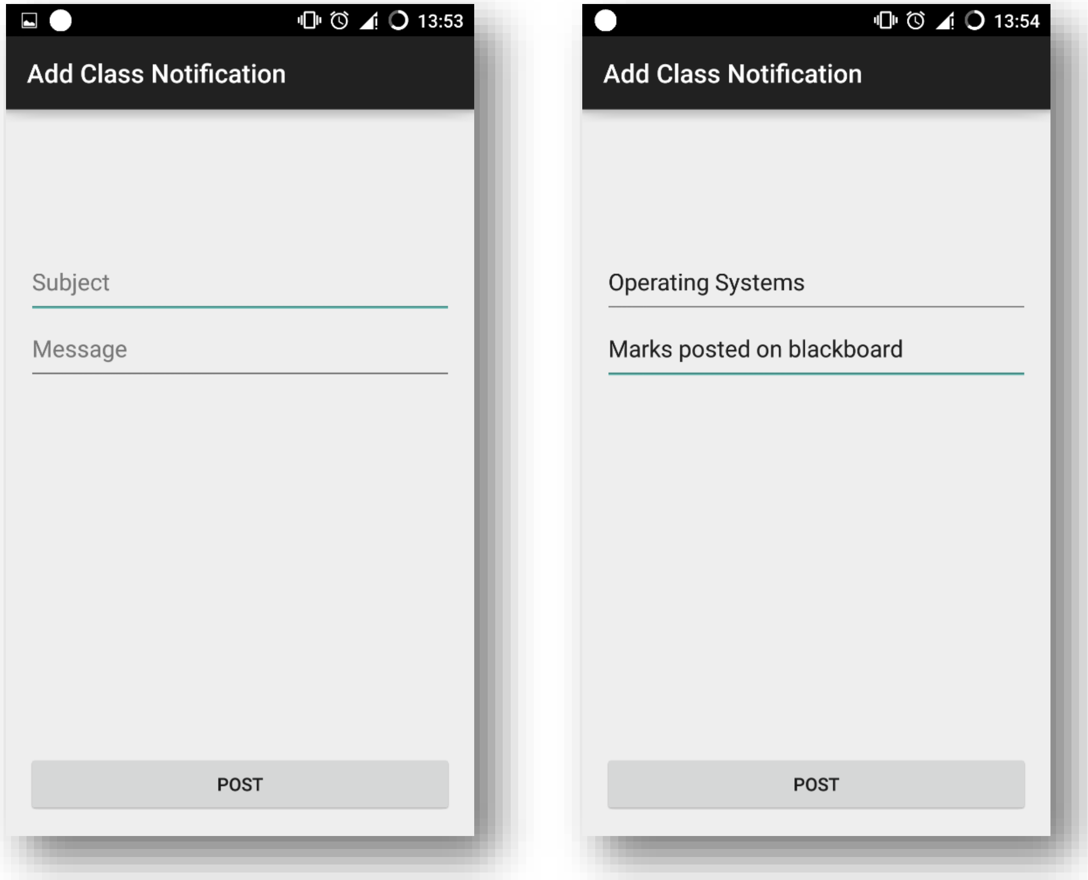
Subject : Analogue and Digital Communication
Message : Lecture in room 23
Posted On : 21-04-2015 13:51:50

Fig. 3.2: Class Notifications Summary Page.

Selecting the FAB, navigates to the ad notification page, where new notifications can be added using a text box.

There are 2 fields in Class notifications:

- ➢ Subject
- ➢ Message



Add Class Notification

Subject

Message

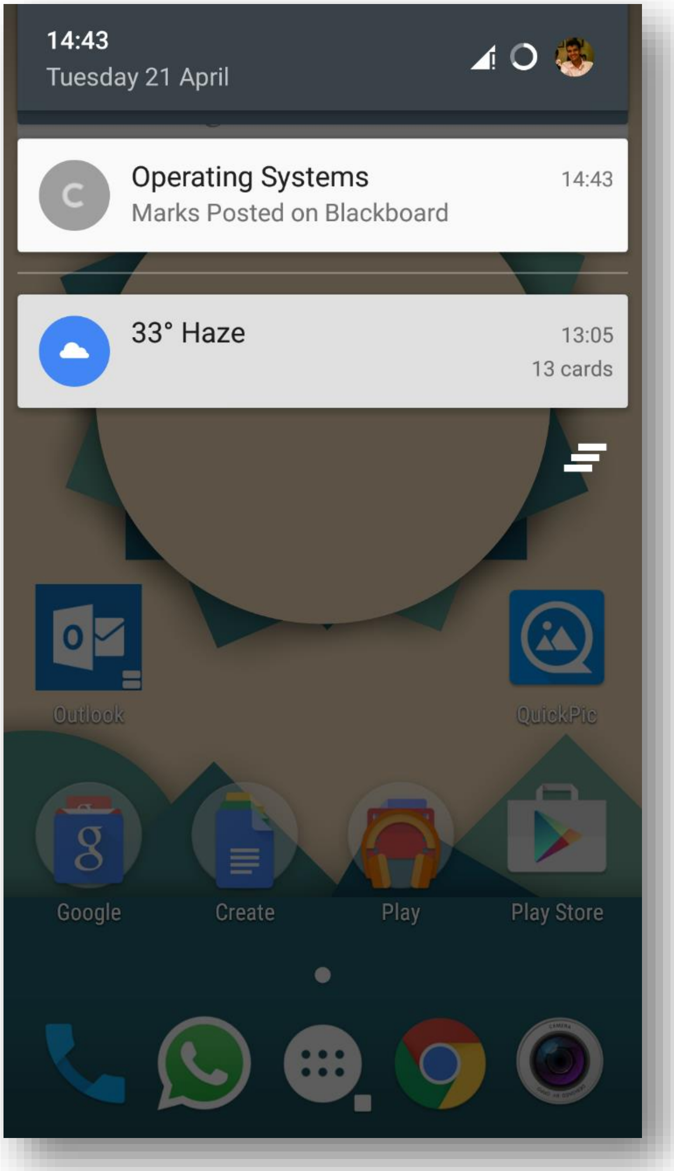
Add Class Notification

Operating Systems

Marks posted on blackboard

Fig 3.3: New Class Notification Page

After a new input has been made, there is a notification generated on the central notification panel.



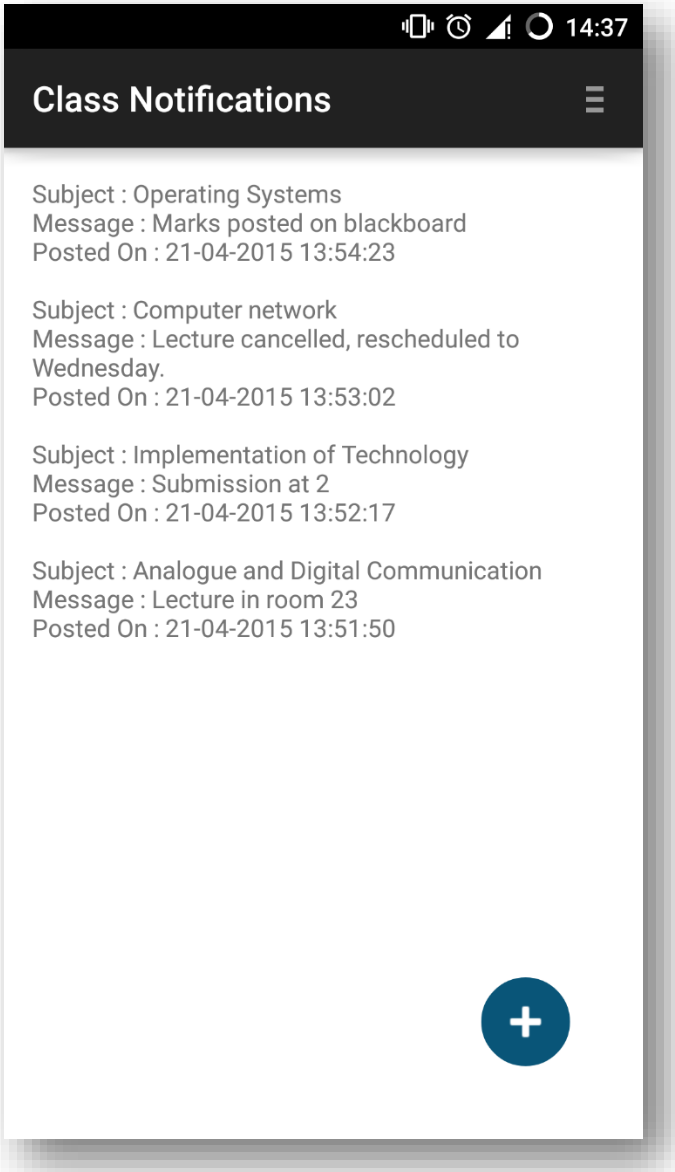
14:43
Tuesday 21 April

Operating Systems 14:43
Marks Posted on Blackboard

33° Haze 13:05
13 cards

Fig 3.4: Notification in the Notification Panel

The newest Notification is added on top, as shown below.



Subject : Operating Systems
Message : Marks posted on blackboard
Posted On : 21-04-2015 13:54:23

Subject : Computer network
Message : Lecture cancelled, rescheduled to Wednesday.
Posted On : 21-04-2015 13:53:02

Subject : Implementation of Technology
Message : Submission at 2
Posted On : 21-04-2015 13:52:17

Subject : Analogue and Digital Communication
Message : Lecture in room 23
Posted On : 21-04-2015 13:51:50

Fig 3.5: Class Notifications Summary after new input

Event Notifications:

Event Notifications can be posted by the Heads of any Committee in College. They can consist of any activity of the committee that they want to publicize. It has been implemented using a slide out menu which is toggled by the button on top.

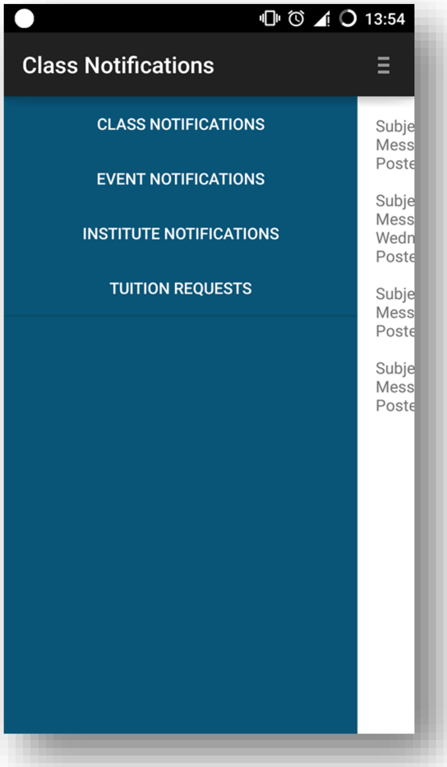
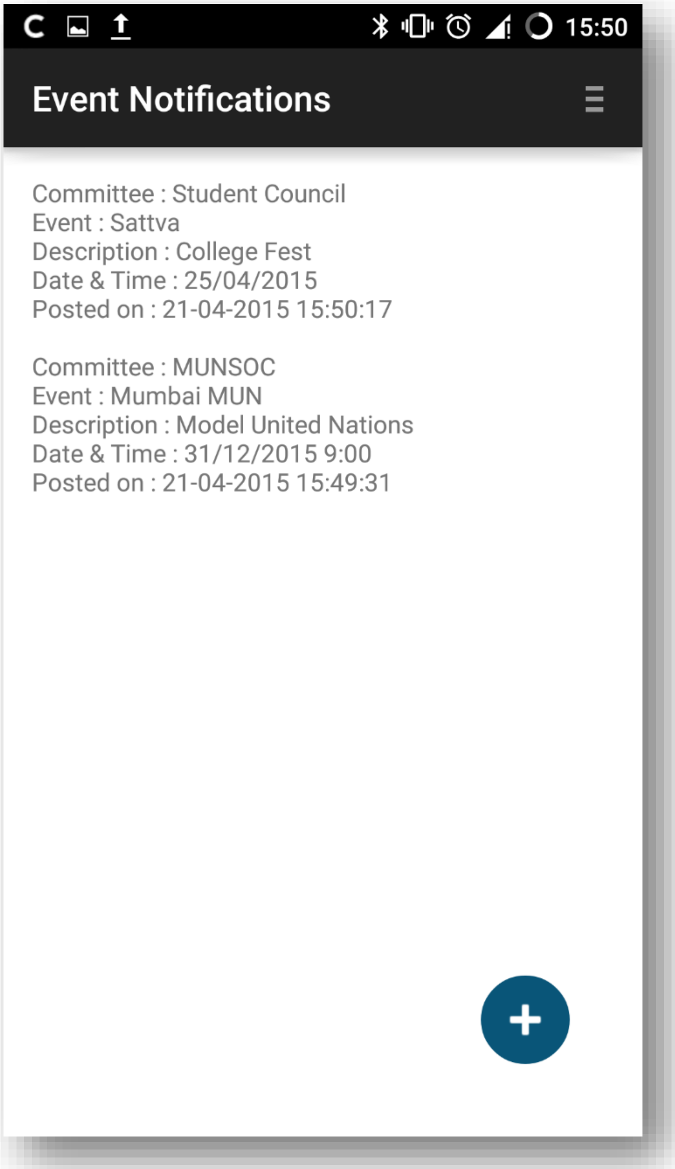


Fig. 3.6: Slide out page for the Event Notification Tab.

After going to the Event Notifications window, the summary of all Event Notifications is shown, with the newest one on top.

There is also an FAB, which can be used to add new notifications.



Event Notifications

Committee : Student Council
Event : Sattva
Description : College Fest
Date & Time : 25/04/2015
Posted on : 21-04-2015 15:50:17

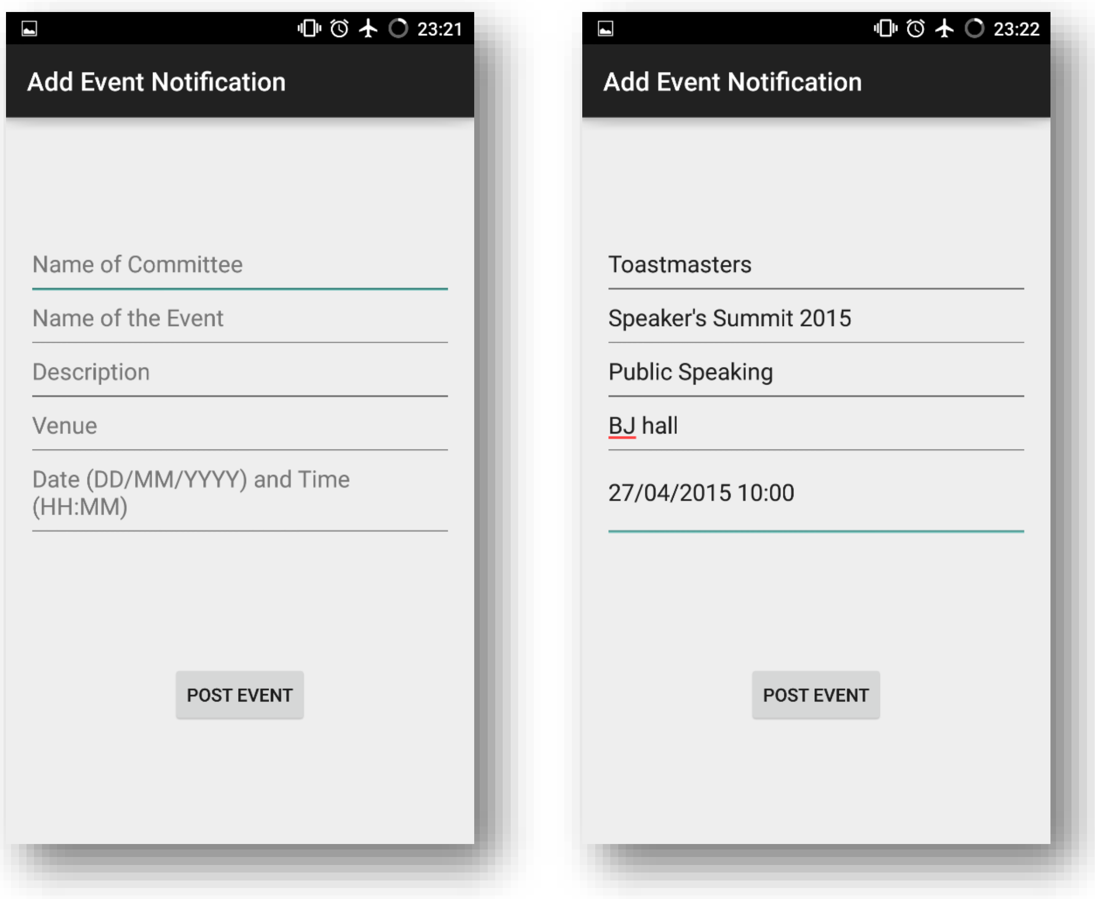
Committee : MUNSOC
Event : Mumbai MUN
Description : Model United Nations
Date & Time : 31/12/2015 9:00
Posted on : 21-04-2015 15:49:31

Fig. 3.7: Event Notifications Summary Page.

Selecting the FAB, navigates to the ad notification page, where new notifications can be added using a text box.

There are 2 fields in Event notifications:

- Name of Committee
- Name of Event
- Description
- Venue
- Date and Time



Toastmasters

Speaker's Summit 2015

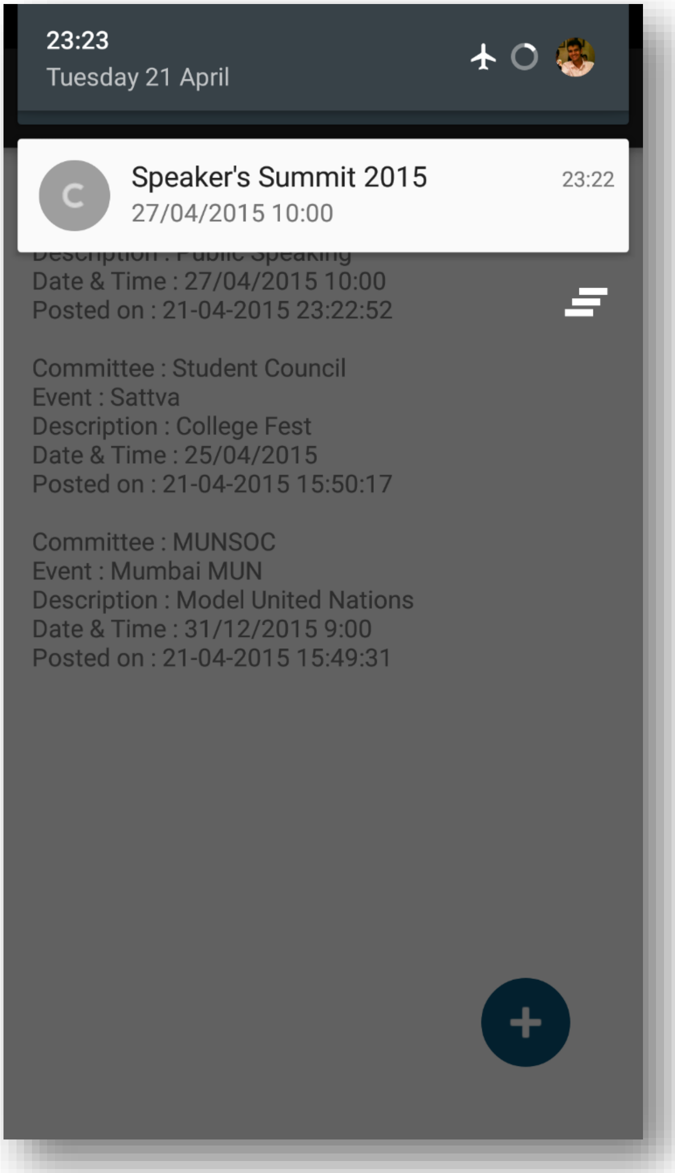
Public Speaking

BJ hall

27/04/2015 10:00

Fig 3.8: New Event Notification Page

After a new input has been made, there is a notification generated on the central notification panel.



23:23
Tuesday 21 April

C
Speaker's Summit 2015
23:22
27/04/2015 10:00

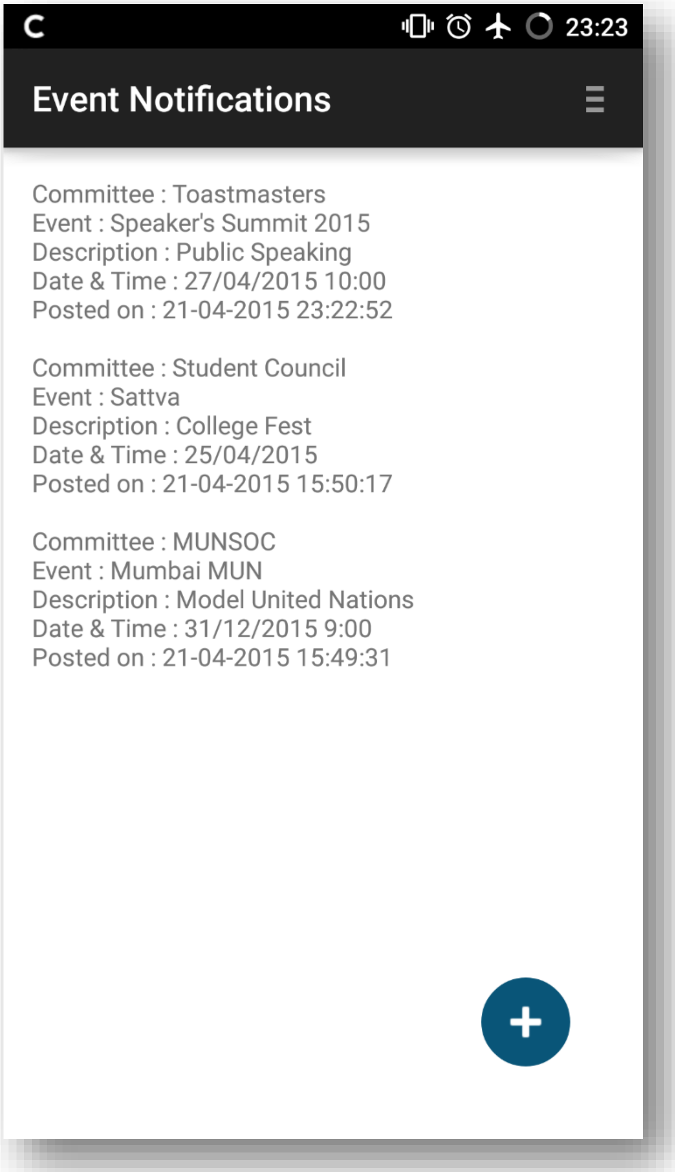
Description : Public Speaking
Date & Time : 27/04/2015 10:00
Posted on : 21-04-2015 23:22:52

Committee : Student Council
Event : Sattva
Description : College Fest
Date & Time : 25/04/2015
Posted on : 21-04-2015 15:50:17

Committee : MUNSOC
Event : Mumbai MUN
Description : Model United Nations
Date & Time : 31/12/2015 9:00
Posted on : 21-04-2015 15:49:31

Fig 3.9: Notification in the Notification Panel

The newest Notification is added on top, as shown below.



Event Notifications

Committee : Toastmasters
Event : Speaker's Summit 2015
Description : Public Speaking
Date & Time : 27/04/2015 10:00
Posted on : 21-04-2015 23:22:52

Committee : Student Council
Event : Sattva
Description : College Fest
Date & Time : 25/04/2015
Posted on : 21-04-2015 15:50:17

Committee : MUNSOC
Event : Mumbai MUN
Description : Model United Nations
Date & Time : 31/12/2015 9:00
Posted on : 21-04-2015 15:49:31

Fig 3.10: Event Notifications Summary after new input

Institute Notifications:

Institute Notifications can be posted by the Course Coordinator and the HoD. They can consist of any announcement that they want to make to the students, on an institute level. It has been implemented using a slide out menu which is toggled by the button on top.

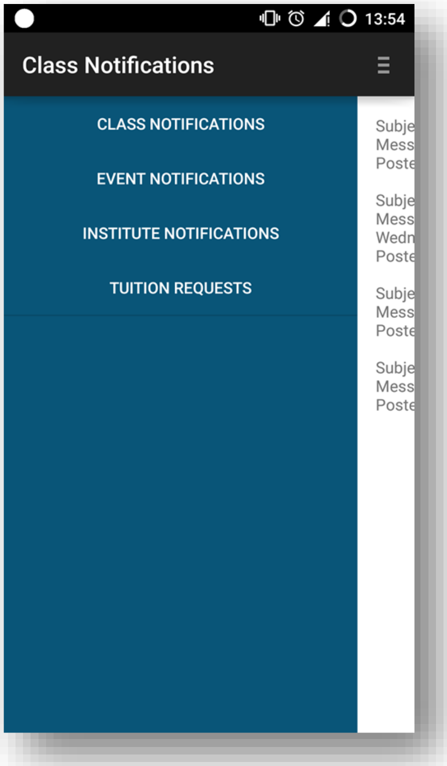
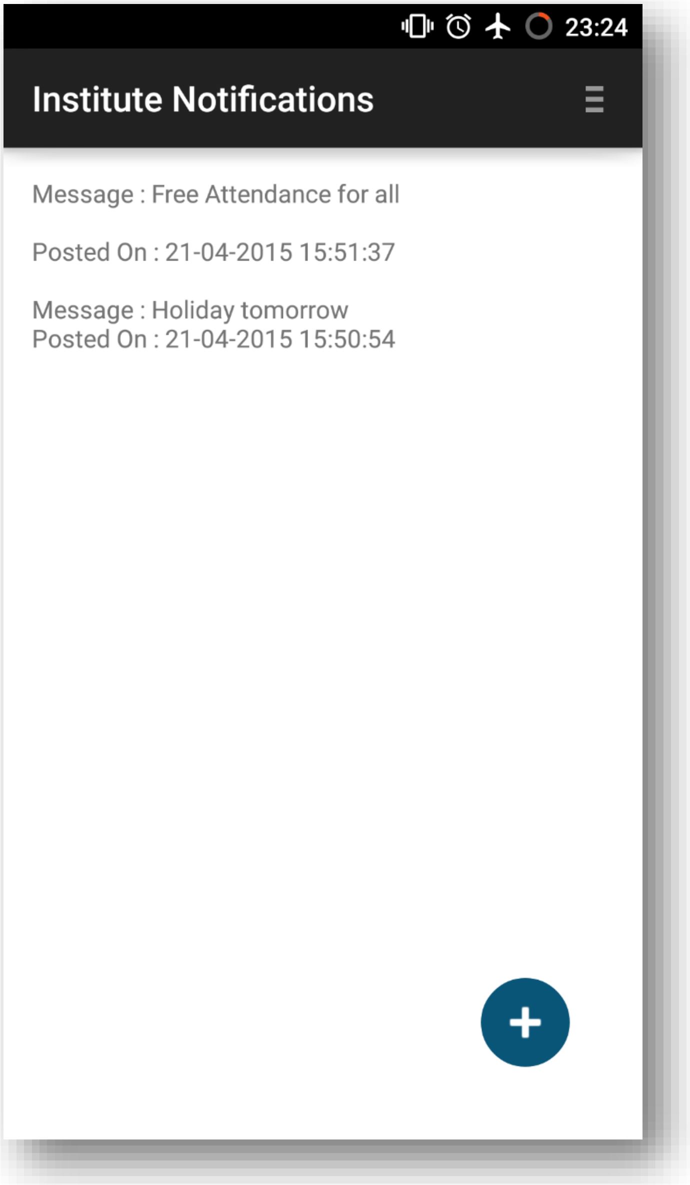


Fig. 3.11: Slide out page for the Institute Notification Tab.

After going to the Institute Notifications window, the summary of all Institute Notifications is shown, with the newest one on top.

There is also an FAB, which can be used to add new notifications.



Institute Notifications

Message : Free Attendance for all

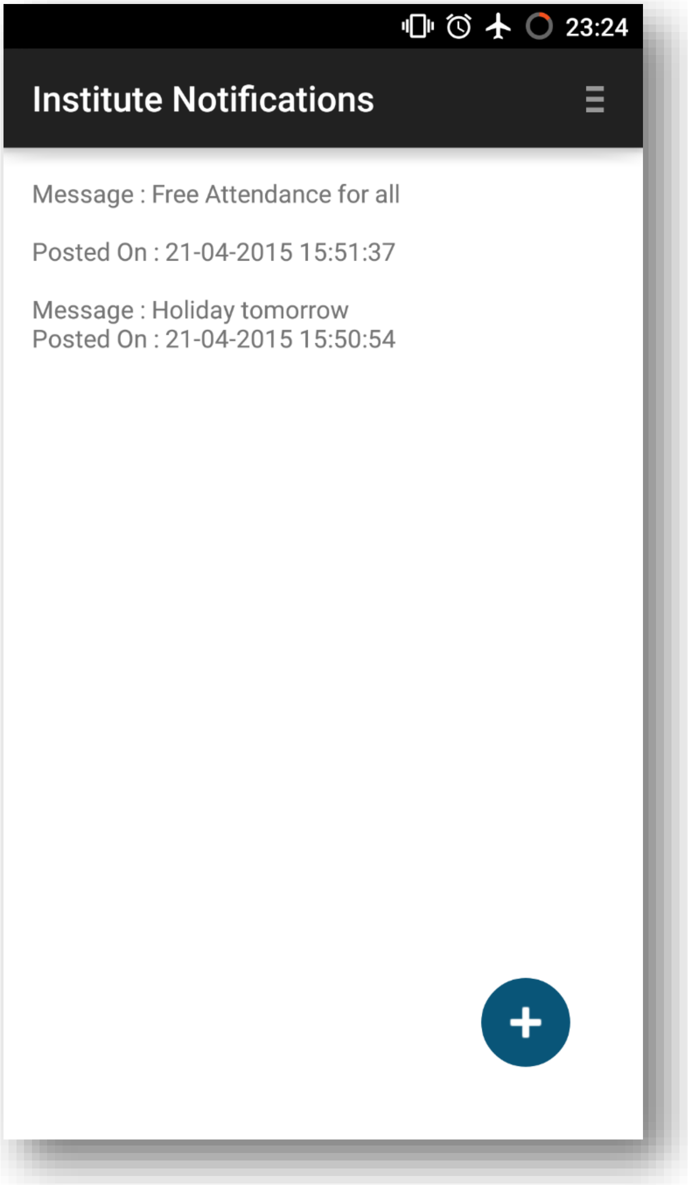
Posted On : 21-04-2015 15:51:37

Message : Holiday tomorrow
Posted On : 21-04-2015 15:50:54

Fig. 3.12: Institute Notifications Summary Page.

Selecting the FAB, navigates to the add notification page, where new notifications can be added using a text box.

There is only 1 fields in Institute notifications, which is message.



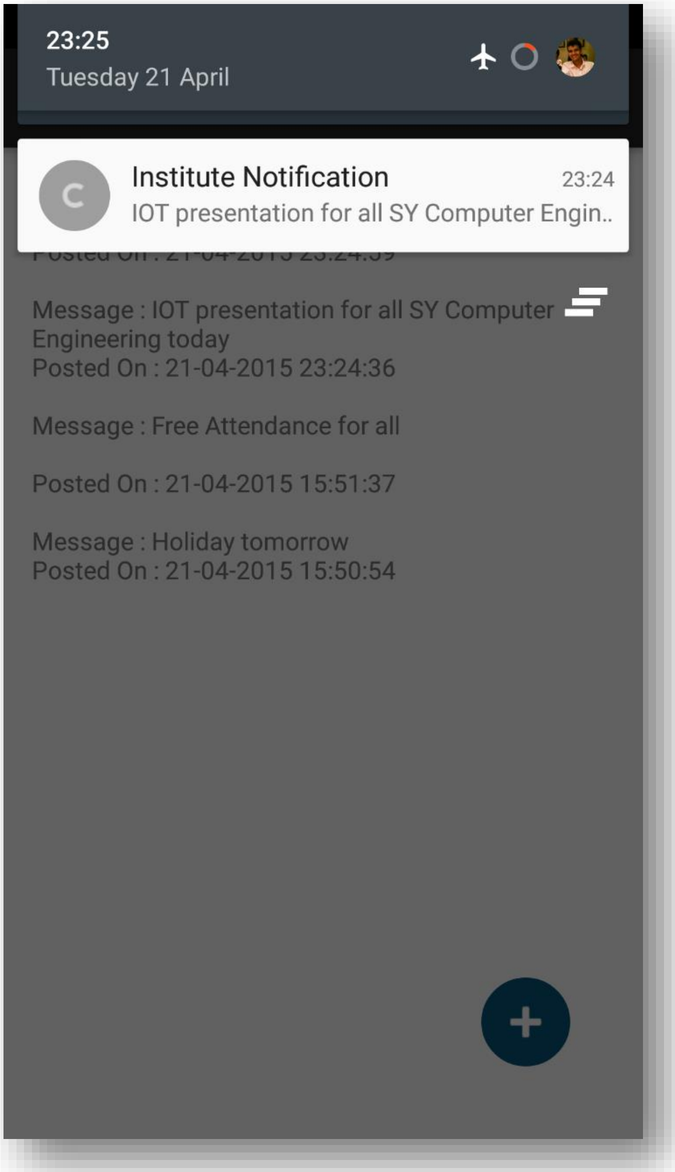
Message : Free Attendance for all

Posted On : 21-04-2015 15:51:37

Message : Holiday tomorrow
Posted On : 21-04-2015 15:50:54

Fig 3.13: New Institute Notification Page

After a new input has been made, there is a notification generated on the central notification panel of all users.



23:25
Tuesday 21 April

Institute Notification 23:24
IOT presentation for all SY Computer Engin..

Posted On : 21-04-2015 23:24:59

Message : IOT presentation for all SY Computer Engineering today
Posted On : 21-04-2015 23:24:36

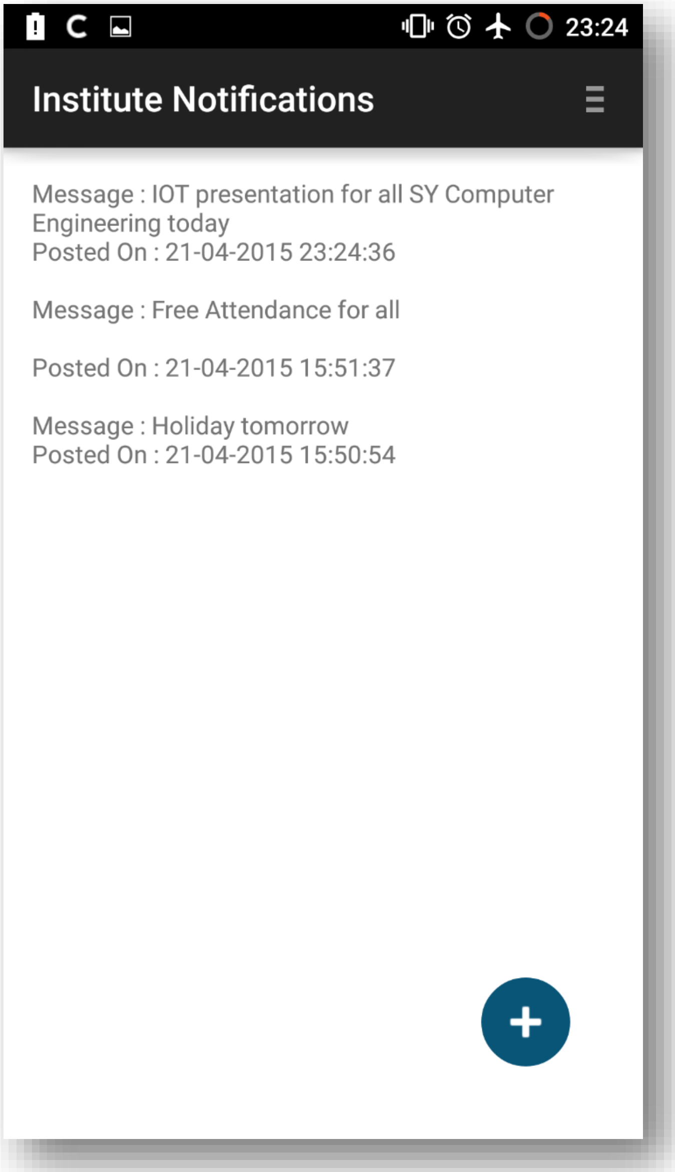
Message : Free Attendance for all

Posted On : 21-04-2015 15:51:37

Message : Holiday tomorrow
Posted On : 21-04-2015 15:50:54

Fig 3.14: Notification on Notification Panel

The newest Notification is added on top, as shown below.



Institute Notifications

Message : IOT presentation for all SY Computer Engineering today
Posted On : 21-04-2015 23:24:36

Message : Free Attendance for all

Posted On : 21-04-2015 15:51:37

Message : Holiday tomorrow
Posted On : 21-04-2015 15:50:54

Fig 3.15: Institute Notifications Summary after new input

Tuition Request:

The tuition request feature has been implemented on a request-acceptance basis.

Anyone who seeks tuition, posts a request and the list is updated accordingly, after which it is updated on all the devices on the request page.

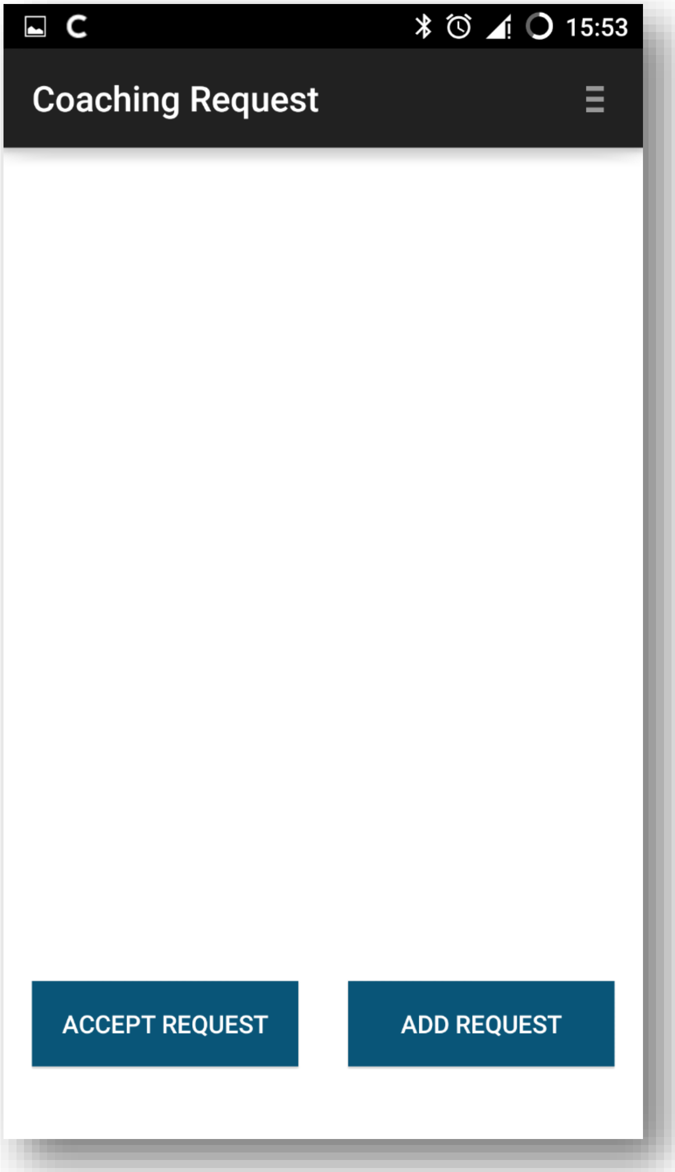
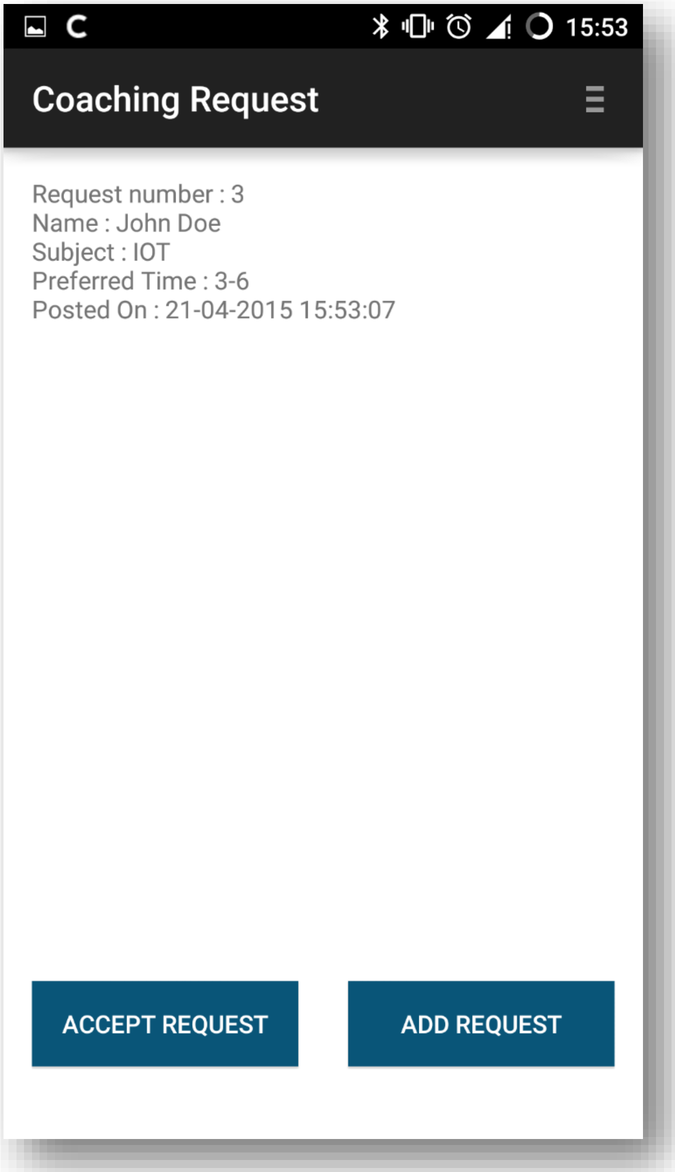


Fig 3.16: Tuition Request Page

A new request can be posted using the 'Add Request' tab. Selecting this tab will toggle the Request Page with the following fields:

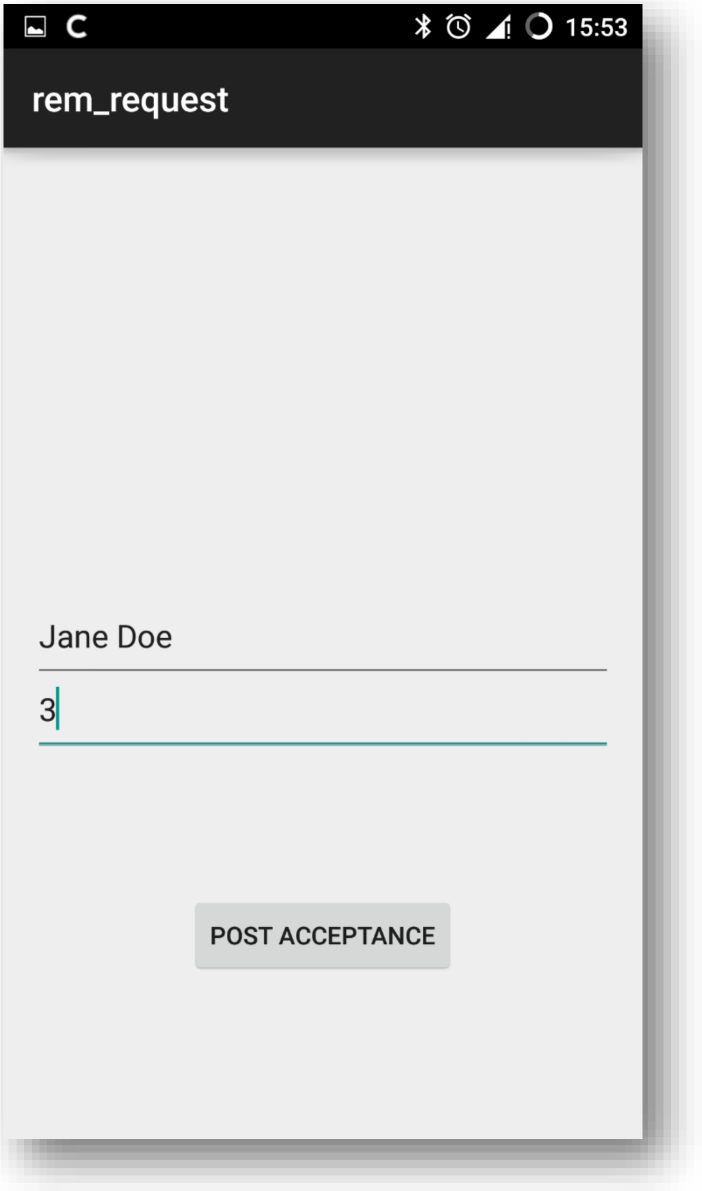
- Name
- Subject
- Preferred Time



Request number : 3
Name : John Doe
Subject : IOT
Preferred Time : 3-6
Posted On : 21-04-2015 15:53:07

Fig 3.17: Add Request Page

A request is assigned a request number. This is used while accepting a request.
A request can be accepted from the ‘Accept Request’ Tab. The request is accepted by putting the request number and the name of the acceptor.



Jane Doe

3

Fig 3.18: Accept Request Page

Once a request has been accepted, it is taken off the Request Page, to unclutter the same. This keeps the process streamlined.

Database Structures:

Classify uses several tables in its databases. The databases have been implemented on the native Android Studio database functionality.

The tables are implemented separately for each type of notification. This is done to avoid any mixing of the data in the tables. An auto-incrementing Primary key has been implemented in all the tables, to ensure proper functionality in the long run.

Schema:

The following Schema have been used in the Database:

1. Class Notification:

```
private static final String DATABASE_NAME = "ClassifyDB";
 private static final String DATABASE_TABLE = "ClassNotif";

db.execSQL("CREATE TABLE IF NOT EXISTS " + DATABASE_TABLE + " (" +
KEY_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " + KEY_SUBJECT +"
VARCHAR NOT NULL, " + KEY_MESSAGE + " VARCHAR NOT NULL,
"+KEY_POSTED+" VARCHAR NOT NULL)");
```

This is the table resulting from the schema above:

Primary Key (*Auto Incrementing)*	Subject	Message	Posted

Table 3.1: Schema for Class Notification

2. Event Notification:

```
 private static final String DATABASE_NAME = "EventDB";
 private static final String DATABASE_TABLE = "EventNotif";


db.execSQL("CREATE TABLE IF NOT EXISTS " + DATABASE_TABLE + " (" +
KEY_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
KEY_COMMITTEE +" VARCHAR NOT NULL, " + KEY_EVENTNAME + "
VARCHAR NOT NULL, "+KEY_DESCRIPTION+" VARCHAR NOT NULL,
"+KEY_DATE+" VARCHAR NOT NULL, "+KEY_POSTED+" VARCHAR NOT
NULL)");
```

This is the table resulting from the schema above:

Primary Key (*Auto Incrementing)*	Committee	Event Name	Description	Date	Posted

Table 3.2: Schema for Event Notification

3. Institute notifications:

```
private static final String DATABASE_NAME = "InstututeDB";
private static final String DATABASE_TABLE = "InstituteNotif";

db.execSQL("CREATE TABLE IF NOT EXISTS " + DATABASE_TABLE + " (" +
KEY_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " + KEY_MESSAGE
+" VARCHAR NOT NULL, "+KEY_POSTED+" VARCHAR NOT NULL)");
```

This is the table resulting from the schema above:

Primary Key (*Auto Incrementing)*	Message	Posted

Table 3.3: Schema for Institute Notification

4. Tuition Requests:

```
db.execSQL("CREATE TABLE IF NOT EXISTS " + DATABASE_TABLE + " (" +
KEY_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " + KEY_NAME +"
VARCHAR NOT NULL, "+ KEY_PREFERREDTIME +" VARCHAR NOT NULL,
"+KEY_SUBJECT+" VARCHAR NOT NULL, "+KEY_POSTED+" VARCHAR
NOT NULL)");
```

This is the table resulting from the schema above:

Primary Key (*Auto Incrementing)*	Name	Preferred Time	Subject	Posted

Table 3.4: Schema for Tuition Request

Classify uses standard SQL queries for injecting data into the tables. The data is taken from the user with the text boxes provided.

Design:

Classify has been designed using a hybrid of Holo and Material design techniques, both of which were introduced by Google for use in mobile applications. These design guidelines make sure that the application is optimized for use on devices across all screen sizes, from 3.5 inch cell phones, to 25 inch All-in One computer systems.

The sliding menu pane as well as the Floating Action Button, are both part of Material Design.

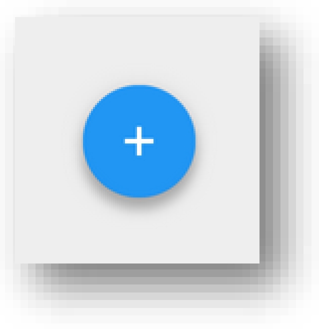


Fig. 3.19: Floating Action Button (with elevation on the Hypothetical Z-Axis)

The action button used in Classify is a flat button, with no elevation on the hypothetical Z-Axis. This is done, because there aren't that many overlapping layers over which the button needs to establish prominence.

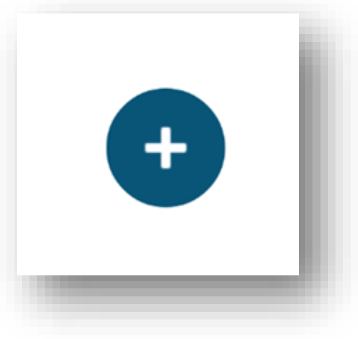


Fig. 3.20: Floating Action Button used in Classify (without elevation on the Hypothetical Z-Axis)

The textboxes, buttons and other minute components continue using the Holo design technique from the older versions of android. This is done because the newer APIs for Material Design haven’t matured yet into daily usage.

Colour Scheme:

In keeping with the material design, we have used a prominent colour, navy blue as the primary colour of the application. This colour is used in the following places:

1. The Icon
2. The Action Button
3. The Page Headers
4. The Sliding window menu.

The hash code for the colour is: **#095578**

Icon:

The icon of Classify is based on Material Design as well. It is circular and uses a shadow effect.

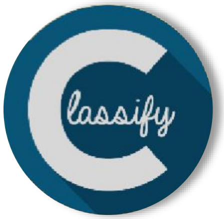


Fig. 3.21: Icon of Classify

Methods Utilized

Android Studio:

Classify has been developed primarily in Java using Android Studio IDE, launched by Google as a replacement to Eclipse. It is a cross platform IDE, running on Linux, Windows and Mac OS X.

Fig.4.1: Android Studio IDE

Android Studio is capable of real time app rendering and has various integrated templates. Which makes developing applications on it easier compared to Eclipse. It also has a drag and drop functionality for UI components, which makes development of the GUI easy. Built-in support for Google Cloud Platform, enabling integration with Google Cloud Messaging and App Engine.

Android studio has many other features as well, some are:

- Gradle-based build support.
- Android-specific refactoring and quick fixes.
- Lint tools to catch performance, usability, version compatibility and other problems.
- ProGuard and app-signing capabilities.
- Template-based wizards to create common Android designs and components.
- A rich layout editor that allows users to drag-and-drop UI components, option to preview layouts on multiple screen configurations.
- Support for building Android Wear apps.

SQLite:

SQLite is used for database management in Classify. The Schemas shown previously are implemented within the Android Studio IDE, as Android Studio has SQLite integrated in it.

SQLite is compatible with over 40 programming languages, this makes it ideal for development on Android.

The Password and Validation was done with the help of SQLite as well, to help maintain multiple users in the future.

Results and Discussion

The result of 3 months of effort and learning has culminated into Classify, which can be very useful to all who use them in college and otherwise. The members involved in the team have learned a great deal about android development, database management, server implementation, java programming, GUI designing, documentation, mobile optimization of applications, graphic design, debugging on several devices and implementation of applications based on a design technique.

Future Scope and Conclusion

Classify is still at version 1.0.0. Following is the list of features it can have and future scope:

1. Implementation of notifications on wearable devices such as smart watches and smart eyewear.
2. Use of the framework of the application in different institutes to bridge the communication gap.
3. Use of this application in corporations for workforce management and notifying them about the same.
4. Full upgrade to Material Design.
5. Notification Synchronization on multiple devices by the same user.
6. Developing a Web Application (HTML based) for cross platform development, so that the application may run on Apple and Windows Phone Devices.
7. Voice typing on the fields.
8. Ability to attach files while posting notifications.

9. Seen acknowledgement for all notifications to confirm receipt by the entirety of the target audience.
10. Integration with Blackboard and Pushbullet services, to put notification on the desktop of the user and redirect directly to the website on clicking.

These are just a few future improvements and future scope for development for what is already a wholesome application for students, teachers and administrators of any institution to use. Classify is a necessity of many educational and private institutions in today’s day and age, given that its basic intention is to streamline notifications and utilize time.

References

[1] developer.android.com
[2] google.co.in/design
[3] developer.android.com/training/material
[4] developer.android.com/reference/android/database/sqlite/package-summary
[5] developer.android.com/guide/components/activities
[6] developer.android.com/training/basics/activity-lifecycle/stopping