# Synopsis
# on
# GTBIT
# Android Application



Team Members –

Divgun Singh Sethi (05813202712)
Inderdeep Singh Khanna (01113202712)
Baldeep Singh Handa (00913202712)

Under the Guidance of –
Mr. Gaurav Sandhu

Submitted To-
Mr. Parminder Pal Singh Bedi

# Abstract

Our purpose is to design and build an android application for "GTBIT".

The application is developed to run on the Android operating system, using Android Studio. The programming language is Java. The project is tested on Android devices and emulator (which is a tool that allows developers to easily test an application without having to install the application on a real device). This Android based application is designed to bring essential Institute information to the user. It uses the client/server architecture to incorporate a fully working group chat functionality. This application consists of eight different pages with different functionalities solving various problems of students.

User can find various information including basic information, information about various societies of GTBIT, contact information and information about the latest events and fest. Users can also download their respective syllabus, lecture plan, time table and other notes directly from the app.

The group chat functionality provides User can Login into the application using existing Facebook/Google Plus account or Register on the API server. The client can add a room for group chat and send or receive a text message while the users in group are on-line.

Users can also share Images, Location and Emoticons. The person who wants to share any notes from image can send them in textual format using OCR (Optical Character Recognition).

# Modules

The Backbone of our group chat functionality is a service called **Google Cloud Messaging (GCM)**. Using this service you can send data to your application whenever new data is available on a server, instead of making requests to the server in a timely fashion. It can also be used for receiving data from our application. Additionally **Python** has been used as a server-side scripting language and **MySQL** as server-side database. The server will handle all the traffic and GCM requests.

Following is some brief description of the functionalities incorporated in our project.

**About Us Page:** This Page gives basic information about the college campus and history.

**Societies Page:** This Page displays a list of all the societies of college, information about the society and contact details are also provided so that students can join different societies easily.

**Latest Events Page:** This page consist of a live feed which gives information about latest events happening in college campus.

**Downloads:** This Page Consist of download links of syllabus, lecture plan, question bank etc. For each semester of college.

**Gates:** Gates Page Consist of all the information of GTBIT's Annual Event Gates, the page consist of name and posters of each and every event.

**Students Timetable:** Timetable page displays class lecture timetable of students of each semester and year.

**Notice Board:** This Page is designed keeping in mind wastage of paper which can be avoided by posting different announcements, upcoming event or examination on this live feed. This live Feed can be used by societies, teachers or students easily by using the upload notice feature on the page.

**Contact Us:** This page contains navigate to college and feedback feature.

**Push Notifications:** Our application implements push notification mechanism provided by GCM, where server automatically sends messages to the device whenever available so that it need not poll the server continuously for messages which may drain battery power.

**Login API:** As GCM server identifies a particular device by its registration ID, our custom designed Login API will help user for one time registration with the GCM server thus enabling the device to send and receive messages to and from the server. We have also provided the facility to login through ones Facebook or Google Plus account.

**Image & Location Sharing:** Our application provides the facility to share images from gallery with all the members of the group. Also, one can share his/her location with just a single click in chatroom.

**Optical Character Recognition (OCR):** One of the most important and useful feature of our application is **OCR.** OCR implemented using Tesseract Android Tools will allow our application to extract text from the image being captured by the device camera and forward the extracted text.

**Emoticons:** Only text messaging would have been a boring choice, thus to your rescue and to make our application more lively we have implemented emoticons using a custom android library dedicated solely to '**Emojis**'.

**Speech to Text:** User can provide speech input to our application instead of typing the text and see his voice been converted to text automatically. User can also make his device speak out a message simply by long pressing the text and leave rest to the **Text to Speech (TTS)** feature of our application.

# Requirements

**Software Requirements**:

- **Android Studio**- an Integrated Development Environment (IDE) for developing on the Android Platform. Based on JetBrains Intelli J IDEA software, Android Studio is designed specifically for Android development.

- **Android SDK Tools**- Android applications are usually developed in Java programming language using the Android Software Development Kit (SDK). It includes a comprehensive set of development tools.

- **Emulator**- A mobile device Emulator is required for testing and presentation of the application. It comes bundled with Android SDK tools.

- **Tesseract**- Tesseract is probably the most accurate open source OCR engine available. Combined with the Leptonica Image Processing Library it can read a wide variety of image formats and convert them to text. A fork library of tesseract has been used in our application.

- **Emojicon**- Emojicon is an open source android library used to implement emoticons in EditText boxes.

**Hardware Requirements:**

- **PC**- Personal Computer to with atleast 2 Gb of RAM, 15Gb of storage.

- **Android Device**- A physical android device with atleast 512Mb RAM,Minimum Android Version Ice Cream – Sandwich.

# References

The following links were searched and exploited extensively for the project development and implementation.

1. http://www.java.sun.com/products\java
2. http://www.jakarta.apache.org
3. http://www.javaworld.com/
4. http://www.java2s.com/
5. http://www.androidhive.info/
6. http://www.appsrox.com/
7. http://developer.android.com/
8. https://plus.google.com/+AndroidDevelopers/
9. https://www.udemy.com
10. http://www.tutorialspoint.com/android/
11. http://www.vogella.com/tutorials/android.html
12. http://www.codelearn.org/android-tutorial
13. https://code.google.com/p/tesseract-ocr/wiki/TrainingTesseract3
14. http://www.codeproject.com/Tips/840623/Android-Character-Recognition
15. https://en.wikipedia.org/wiki/Optical_character_recognition
16. https://github.com/rmtheis/android-ocr
17. http://gaut.am/making-an-ocr-android-app-using-tesseract/
18. https://github.com/rockerhieu/emojicon