COLLEGE APP

A Project Report Submitted to
RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA
BHOPAL (M. P.)

In Partial fulfillment of the requirement for the award of
Degree of
BACHELOR OF ENGINEERING
In Computer Science & Engineering

2017-2018

Submitted By

FARIHA FATIMA (0114CS141024)
JAI KUMAR KHERAJANI (0114CS141038)

Under the Guidance of
PROF. AMIT SAXENA
(Head of Dept. CSE)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
TRUBA INSTITUTE OF ENGINEERING & INFORMATION TECHNOLOGY
BHOPAL (M. P.)

TRUBA INSTITUTE OF ENGINEERING
&
INFORMATION TECHNOLOGY, BHOPAL

(Affiliated to Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal)

(Session 2017-2018)

Department of Computer Science & Engineering

CERTIFICATE

This is to certify that **Fariha Fatima, Jai Kumar Kherajani** students of VIII semester in Truba Institute of Engineering & Information Technology have completed their Major Project on "**Trubian**", as per the syllabus and has submitted a satisfactory report on this project as a partial fulfillment towards the award of degree of Bachelor of Engineering in Computer Science & Engineering under Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal.

Prof. Amit Saxena
Project Guide &
H.O.D., CSE Dept.

Dr. Rajeev Arya

Director

ACKNOWLEDGEMENT

It is a proud privilege for me to express deep sense of gratitude towards my guide and the head of Computer Science Dept. Prof. Amit Saxena, for their deep interest, constant encouragement, valuable suggestions and constant supervision at making this project study purposeful and making “TRUBIAN” worthy to stand to the test of merit.

I am also thankful to our director, Dr. Rajeev Arya, for encouraging me and providing all the necessary facilities.

At last, I want to thank my team, parents and friends who treasured me for my hard work and encouraged me and finally to God who made all the things possible for me till the end.

TABLE OF CONTENTS

INTRODUCTION

1. INTRODUCTION

The project is based on the current scenario where teachers and students are having a problem of sharing the information like time table, syllabus and other important information. This will provide a medium to connect the teachers and the students, where information can only be posted by teachers, Students will only be able to view it, no changes can be made in the posted information by the students. So, the information is authentic and credible, plus ensures that you have the best experience because the app is designed for the specific purpose right from the start.

It would be a great relief for the teachers and students when they can share the information in just one click on their mobile phones. And hence an application that comes to their rescue and save their time and energy.

Also, people prefer an easy to use and manage application rather than a complex one and here we are presenting “TRUBIAN” to make work easy.

1.1 Overview and issues involved

This project is of great use for the teachers and students. An android application for the ease of the communication and sharing of multiple information between them. We came through some situations a lot of time when teachers posted some information in the class group and because of other chatting going on in the group already we are unable to see the information and faced troubles because of that. In this application you would observe that only the things, information, news etc. which are important will only posted and nothing else.

Every semester teachers and students face a problem of connectivity, groups have to be made, and with so much happening in so many groups in so many apps it is easy to miss an important update from college. So, in order to overcome that this app will be useful. This will provide a medium to connect, where information can only be posted by teachers and students can only view it. So, the information is authentic and credible, plus ensures that you have the best experience because the app was designed for the specific purpose right from the start. All sorts of information like time tables, syllabus, and other important information which is to be shared among the students will be uploaded on this application by the teachers.

So, there would be a relief for the students and even the teachers that all the information will be shared so that there is no chance of denial by the students that they did not receive the specific piece of information.

1.2 Problem Statement

Every semester teachers and students face a problem of connectivity, groups have to be made, and with so much happening in so many groups in so many apps it is easy to miss an important update from college. So, in order to overcome that this app will be useful. This will provide a medium to connect, where information can only be posted by teachers and students can only view it. So, the information is authentic and credible, plus ensures that you have the best experience because the app was designed for the specific purpose right from the start. This is an application that comes to rescue and save time from making groups and adding hundreds of students in it. The information can be seen as soon as it is uploaded. There will be zero chance of missing the information which was the biggest problem before.

1.3 Proposed Solution

An android application for the ease of the communication and sharing of multiple information between teachers and students. It would be a great relief for teachers if they could convey the message, share the information by posting it on with just one touch on their mobile phones as well as it is easy for the students to check the information at anytime and from anywhere. And hence an application that comes to their rescue and save time from making groups and adding hundreds of students in it. The information can be seen as soon as it is uploaded.

With the help of this application there will be a good communication between the teachers and the students. Sometimes there are students who are not in groups where teachers share the information so by this application they can login from anywhere at anytime and look up for the information. It is totally different and a new concept application, mostly because there will only be the sharing of information and nothing else. Only the time table related and important announcements will be made on this

The App needs user to register and sign in, where they have to enter their email id and roll number. Once they login they can see the newsfeed, time table and all the information regarding the college. But the student should have signed up for the account, without having an account they cannot log in or see any information. Students will just be allowed to look up the information they can never change anything or any information, that will only be done by teachers.

LITERATURE SURVEY

2. LITERATURE SURVEY

2.1 Methodology

We have done a detailed study of various solutions that are already present in the market and are in use by other educational institutions. Some of the apps available on the google play store that fall under the same category are One School, My School.

The project is based on the current scenario where teachers and students are having a problem of sharing the information like time table, syllabus and other important information. This will provide a medium to connect the teachers and the students, where information can only be posted by teachers, Students will only be able to view it, no changes can be made in the posted information by the students. So, the information is authentic and credible, plus ensures that you have the best experience because the app is designed for the specific purpose right from the start.

It would be a great relief for the teachers and students when they can share the information in just one click on their mobile phones. And hence an application that comes to their rescue and save their time and energy.

The App needs user to register and sign in, where they have to enter their email id and roll number. Once they login they can see the newsfeed, time table and all the information regarding the college. But the student should have signed up for the account, without having an account they cannot log in or see any information.

2.2 Technology and Tools

- **ANDROID**

The Technology used in this software is android. We chose to work on android since it is the most popular and widely used OS available in today's date. Most of the smart phone users use android. Also, the version 4.4 of android i.e. KitKat, is being used by almost 74% of the users.

Last but not the least, there is a vast scope of knowledge in this field and this is the reason why we chose to work on android.

Android is a mobile operating system (OS) currently developed by Google, based on the Linux kernel and designed primarily for touch screen mobile devices such as smart phones and tablets. Android's user interfaces mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with the keyboard text input. In addition to touch screen devices,

Applications, which extend the functionality of devices, are written using the Android software development kit (SDK) and, often, the Java programming language that has complete access to the Android APIs. Initially, Google's supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) plugin; in December 2014, Google released Android Studio, based on IntelliJ IDEA, as its primary IDE for Android application development. Other development tools are available, including a native development kit (NDK) for applications or extensions in C or C++, Google App Inventor, a visual environment for novice programmers, and various cross platform mobile web applications frameworks. In January 2014, Google unveiled a framework based on Apache Cordova for porting Chrome HTML 5 web applications to Android, wrapped in a native application shell.

- **ANDROID STUDIO**

Android Studio is the official integrated development environment (IDE) for Android platform development.

It was announced on May 16, 2013 at the Google I/O conference. Android Studio is freely available under the Apache License 2.0. Android Studio was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. Based on JetBrains' IntelliJ IDEA software,

Android Studio is designed specifically for Android development. It is available for download on Windows, Mac OS X and Linux, and replaced Eclipse Android Development Tools (ADT) as Google's primary IDE for native Android application development.

New features are expected to be rolled out with each release of Android Studio. The following features are provided in the current stable version:

- Gradle-based build support.
- Android-specific refactoring and quick fixes.
- Lint tools to catch performance, usability, version compatibility and other problems.

- ProGuard integration 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
- Built-in support for Google Cloud Platform, enabling integration with Google Cloud Messaging and App Engine.

- **JAVA**

There are lots of different ways to write Android programs but the official language of Android is Java. Not only is Java the official programming language for app development, Java itself is used by Google for large parts of the Android internals. There are two distinct parts to writing an Android app. One is the Java programming language itself; the other understands how to create an app in terms of its user interface, the Android OS, and the Android Software Development Kit (SDK). 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).

Java is a widely used programming language expressly designed for use in the distributed environment of the internet. It is the most popular programming language for Android smartphone applications and is among the most favored for edge device and internet of things development.

Java was designed to have the look and feel of the C++ language, but it is simpler to use than C++ and enforces an object-oriented programming model. Java can be used to create complete applications that may run on a single computer or be distributed among servers and clients in a network. It can also be used to build a small application module or applet for use as part of a webpage.

In this session, you'll become familiar with Java, the programming language used to develop Android applications. Our goal is to prepare those already familiar with one programming language, such as PHP or Objective-C, to become comfortable working with the Java programming language and dive into Android app development. In this session, you'll get a brief introduction to Java fundamentals, including object-oriented programming, inheritance and more.

- **XML**

XML was designed to transport and store data. XML (Extensible Markup Language) is a set of rules for encoding documents electronically. It is defined in the XML 1.0 Specification produced by the W3C and several other related specifications; all are fee-free open standards. XML's design goals emphasize simplicity, generality, and usability over the Internet. It is a textual data format, with strong support via Unicode for the languages of the world. Although XML's design focuses on documents, it is widely used for the representation of arbitrary data structures, for example in web services.

There are a variety of programming interfaces which software developers may use to access XML data, and several schema systems designed to aid in the definition of XML-based languages.

- XML stands for Extensible Markup Language
- XML is a markup language much like HTML
- XML was designed to carry data, not to display data
- XML tags are not predefined. You must define your own tags
- XML is designed to be self-descriptive

For example, XML documents can be very simple, such as the following:

<? xml version="1.0" standalone="yes"?>

<conversation>

<greeting>Hello, world! </greeting>

<response>Stop the planet, I want to get off! </response>

</conversation>.

- **No SQL**

No SQL encompasses a wide variety of different database technologies that were developed in response to the demands presented in building modern applications:

1. Developers are working with applications that create massive volumes of new, rapidly changing data types — structured, semi-structured, unstructured and polymorphic data.

2. Long gone is the twelve-to-eighteen-month waterfall development cycle. Now small teams work in agile sprints, iterating quickly and pushing code every week or two, some even multiple times every day.

3. Applications that once served a finite audience are now delivered as services that must be always-on, accessible from many different devices and scaled globally to millions of users.

4. Organizations are now turning to scale-out architectures using open source software, commodity servers and cloud computing instead of large monolithic servers and storage infrastructure.

5. Relational databases were not designed to cope with the scale and agility challenges that face modern applications, nor were they built to take advantage of the commodity storage and processing power available today.

The Benefits of No SQL:

When compared to relational databases, No SQL databases are more scalable and provide superior performance, and their data model addresses several issues that the relational model is not designed to address:

- Large volumes of rapidly changing structured, semi-structured, and unstructured data
- Agile sprints, quick schema iteration, and frequent code pushes
- Object-oriented programming that is easy to use and flexible
- Geographically distributed scale-out architecture instead of expensive, monolithic architecture

2.3 Other Requirements

Model followed here is the prototype as the feature is first implemented, tested and again for adding another feature another prototype is created and tested as a whole unit to see whether the newly added module fits in its place properly or not.

ANALYSIS

3. ANALYSIS

3.1 Software Requirements

1. Android Studio

2. Firebase APIs

3. Windows 10

4. JDK

5. Android OS v16 or above

6. Fresco by Facebook

7. Glide by Bumptech

8. Expresso

9. Material Showcase View

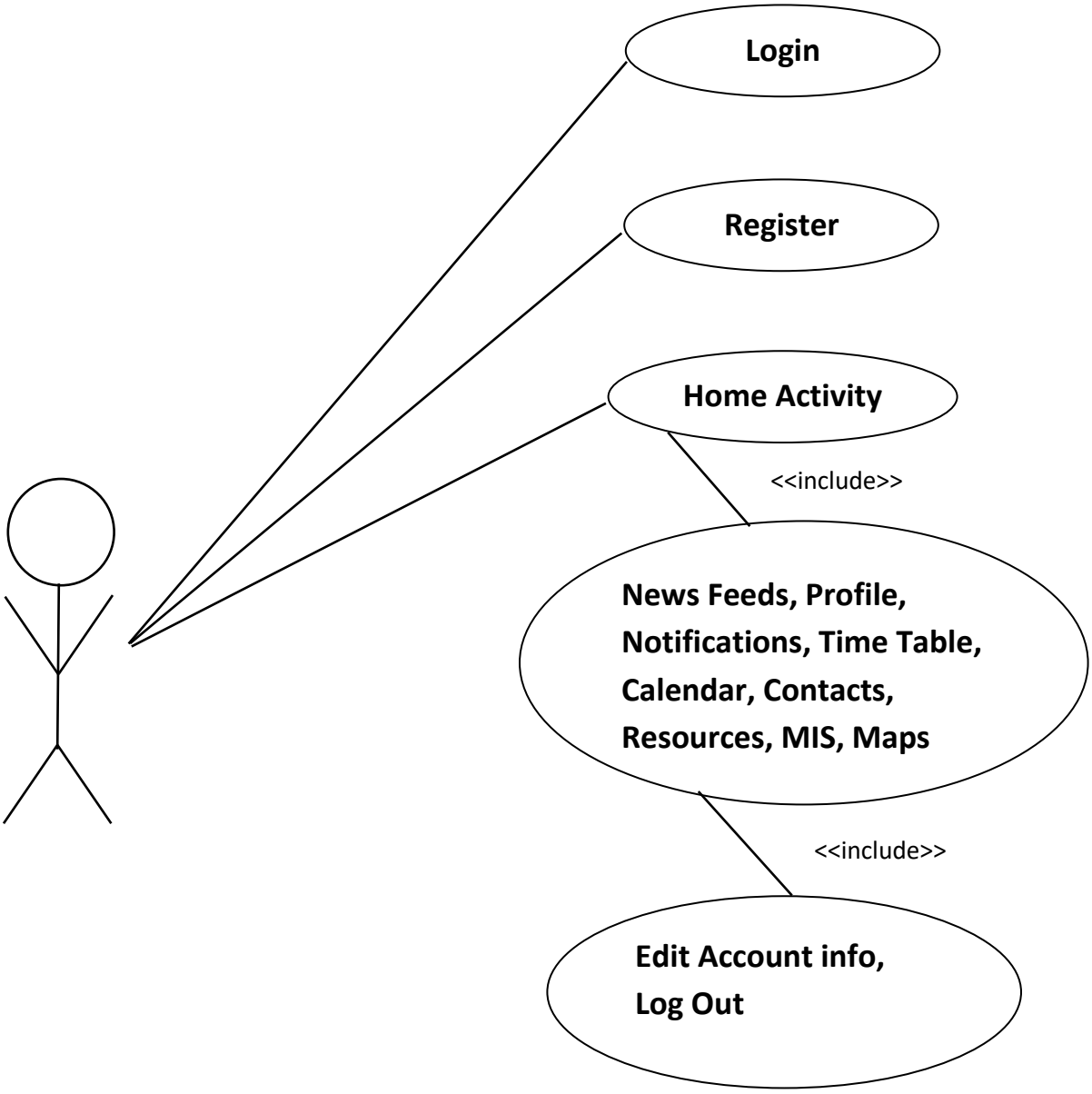
10. Android Support Libraries

3.2 Hardware Requirements

1. Desktop PC (8GB RAM)

2. Android operating system enabled phone

3.3 Use Case Model



3.4 Use Case description

Use-Case Diagram: A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.

Due to their simplistic nature, use case diagrams can be a good communication tool for stakeholders. The drawings attempt to mimic the real world and provide a view for the stakeholder to understand how the system is going to be designed.

A use case diagram consists of Actors, system, Use-cases, Associations, include relationships and Extend relationships.

Actors are the external users which are going to perform various operations on the system.

Use-Cases are the operations that the actors can perform on the system.

Include relationships is a generalization relationship denoting the inclusion of the behavior described by another use case. The best way to think of an include dependency is that it is the invocation of a use case by another one.

Extend relationships is a generalization relationship where an extending use case continues the behavior of a base use case. The extending use case accomplishes this by conceptually inserting additional action sequences into the base use-case sequence. This allows an extending use case to continue the activity sequence of a base use case when the appropriate extension point is reached in the base use case and the extension condition is fulfilled. When the extending use case activity sequence is completed, the base use case continues.

In software and systems engineering, a use case is a list of actions or event steps typically defining the Interactions between a role (known in the Unified Modeling Language as an actor) and a system to achieve a goal. The actor can be a human or other external system. In systems engineering use cases are used at a higher level than within software engineering often representing missions or stakeholder goals. The detailed requirements may then be captured in the Systems Modeling Language (SysML) or as contractual statements.

Advantages:

Since the inception of the agile movement, the user story technique from Extreme Programming has been so popular that many think it is the only and best solution for agile requirements of all projects. Alistair Cockburn lists five reasons why he still writes use cases in agile development.

The list of goal names provides the shortest summary of what the system will offer (even than user stories). It also provides a project planning skeleton, to be used to build initial priorities, estimates, team allocation and timing.

The main success scenario of each use case provides everyone involved with an agreement as to what the system will basically do and what it will not do. It provides the context for each specific line item requirement (e.g. fine-grained user stories), a context that is very hard to get anywhere else.

The extension conditions of each use case provide a framework for investigating all the little, niggling things that somehow take up 80% of the development time and budget. It provides a look ahead mechanism, so the stakeholders can spot issues that are likely to take a long time to get answers for. These issues can and should then be put ahead of the schedule, so that the answers can be ready when the development team gets around to working on them.

The use case extension scenario fragments provide answers to the many detailed, often tricky and ignored business questions: “What are we supposed to do in this case?” It is a thinking/documentation framework that matches the if…then…else statement that helps the programmers think through issues. Except it is done at investigation time, not programming time. The full use case set shows that the investigators have thought through every user’s needs, every goal they have with respect to the system, and every business variant involved.

DESIGN

4. DESIGN

4.1 Technology Selection

1. **Java Platform, Standard Edition or Java SE:** It is a widely used platform for programming in the Java language. It is the Java Platform used to deploy portable applications for general use. In practical terms, Java SE consists of a virtual machine, which must be used to run Java programs, together with a set of libraries needed to allow the use of file systems, networks, graphical interfaces, and so on.

JVM (JAVA VIRTUAL MACHINE), use of JSP does not look us into using specific hardware platform, operating system, or server software.

FEATURES OF JAVA: -

- **SIMPLE: -**Java was designed for the professional programmer to learn and use effectively. It is easier because it has some features of C++ and C which is known to us. Beyond this it has also some extra features which make it easier than other languages.
- **OBJECT-ORIENTED: -**JAVA is object-oriented language. In JAVA everything is treated as object. Java manages to strikes a balance between purists "Everything is an object" object paradigm. The object model in JAVA is simple and easy to extend, while simple types, such as integers, are kept as high performance non-object.
- **ROBUST: -**The program must execute reliably in variety of systems. Thus, the ability to create robust programs gives a high priority in the design of JAVA.
- **MULTITHREADED: -**JAVA supports multithreaded programming, which allows writing program that do many things simultaneously.
- **PORTABILITY: -**For program to be dynamically downloaded to all various platforms Connected to the internet, some means of generating portable executable code is needed. The same mechanism that helps ensure security also helps create portability.
- **SECURITY: -**JAVA achieves this protection by confining a JAVA program to the JAVA execution environment and allowing it access to other parts of the computer.
- **DYNAMIC: -**JAVA programs carry with them substantial amounts of run-time type information that is used to verify and resolve accesses to objects at run time. This makes it possible to dynamically link code in a safe and expedient manner.

2. Android

Android is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics. Android is an open source software stack created for a wide array of devices with different form factors. The primary purposes of Android are to create an open software platform available for carriers, OEMs, and developers to make their innovative ideas a reality and to introduce a successful, real-world product that improves the mobile experience for users. In a nutshell, an Android phone is a powerful, high-tech smartphone that runs on the Android operating system (OS) developed by Google and is used by a variety of mobile phone manufacturers. Pick an Android mobile phone and you can choose from hundreds of great applications and multitask with ease. The latest version is Android 7 Nougat, which will become available for more devices over the coming months. Marshmallow is 6.x, Lollipop is 5.x, and KitKat is version 4.4.x, with 4.3, 4.2 and 4.1 all coming under the codename name Jelly Bean. Mobile Operating System: A mobile operating system, also referred to as mobile OS, is an operating system that operates a smartphone, tablet, PDA, or other mobile device. So, a smartphone that runs on Android OS is an Android phone, you may call it A smartphone/android phone/android-smart-phone.

The use of Android application-

An Android app is a software application running on the Android platform. Because the Android platform is built for mobile devices, a typical Android app is designed for a smartphone or a tablet PC running on the Android OS.

Android stack comprises of –

1. **System and user apps**

- System apps have no special status
- System apps provide key capabilities to app developers

2. **Android OS API in Java framework**

- The entire feature-set of the Android OS is available to you through APIs written in the Java language.
- View class hierarchy to create UI screens

- Notification manager
- Activity manager for life cycles and navigation
- Content providers to access data from other apps

3. Expose native APIs; run apps

- Core C/C++ Libraries give access to core native Android system components and services.
- Each app runs in its own process with its own instance of the Android Runtime.

4. Expose device hardware capabilities

- Standard interfaces that expose device hardware capabilities as libraries

Examples: Camera, Bluetooth module etc.

5. Linux Kernel

- Threading and low-level memory management
- Security features
- Drivers

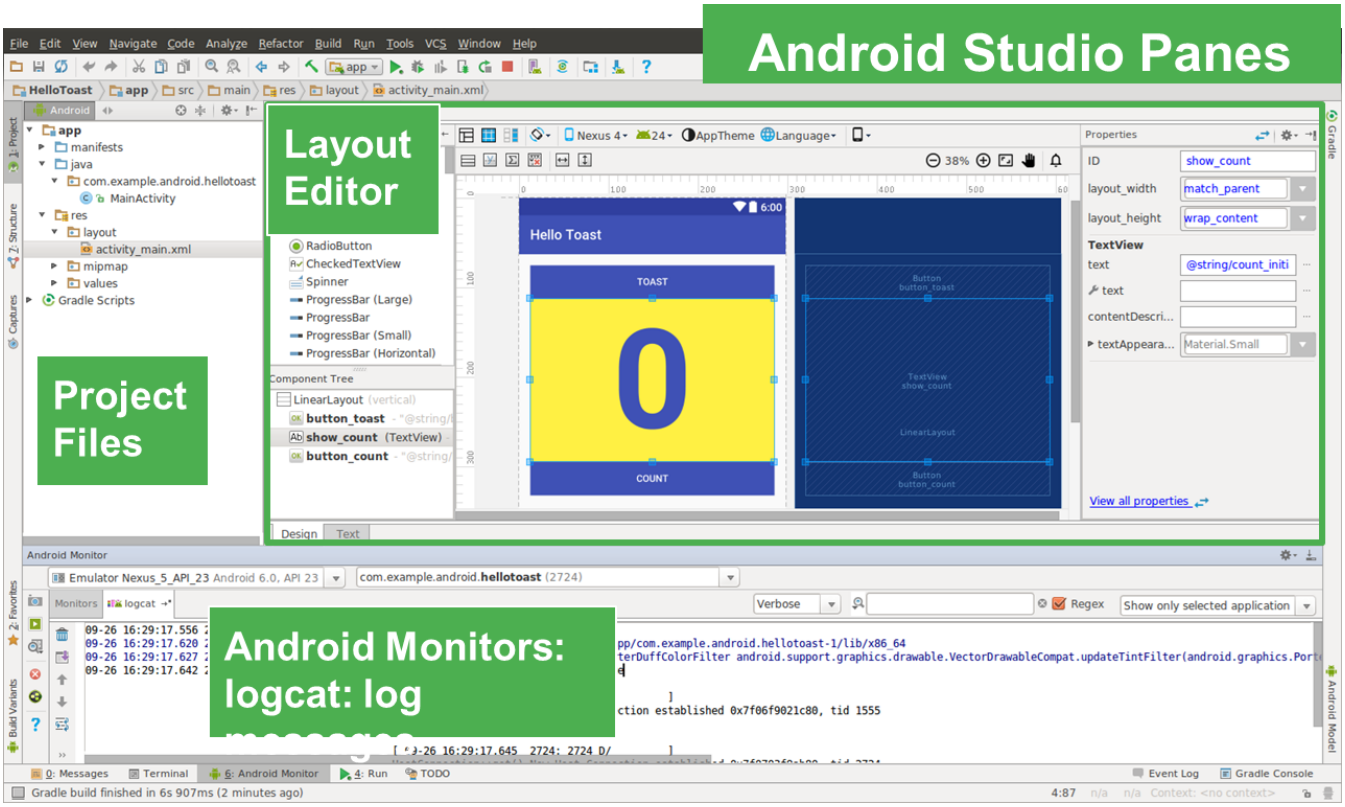
3. Android SDK

- Development tools (debugger, monitors, editors)
- Libraries (maps, wearables)
- Virtual devices (emulators)
- Documentation (developers.android.com)
- Sample code

4. Android Studio

- Android IDE
- Project structure
- Templates
- Layout Editor
- Testing tools
- Gradle-based build

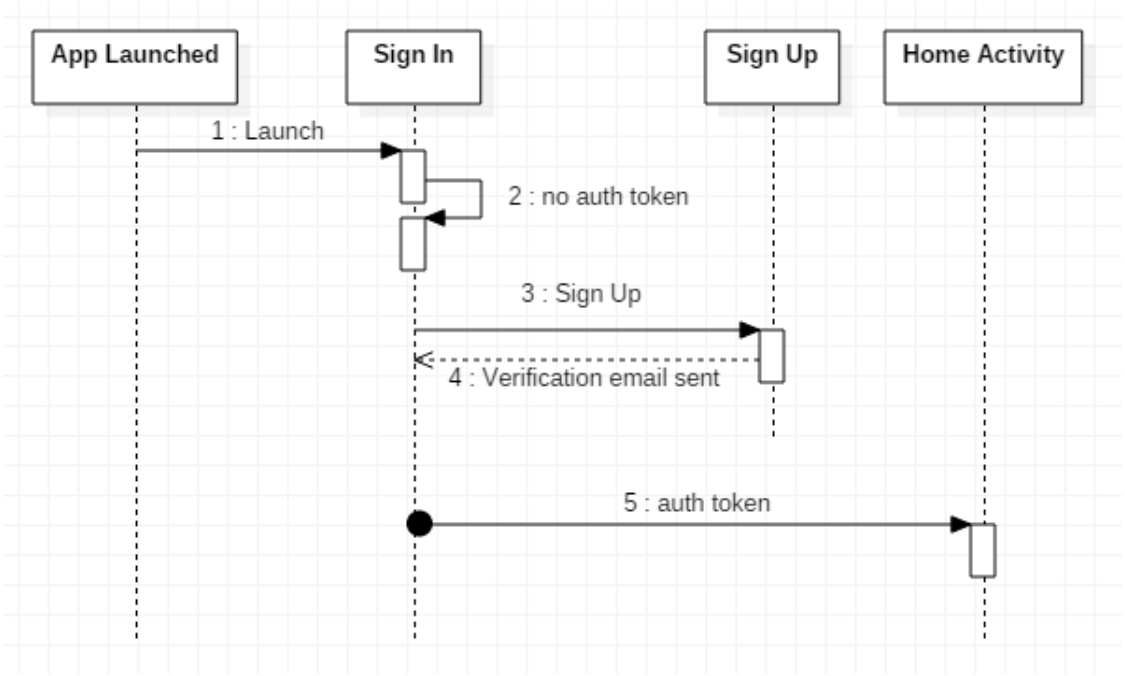
- Log Console
- Debugger
- Monitors
- Emulators



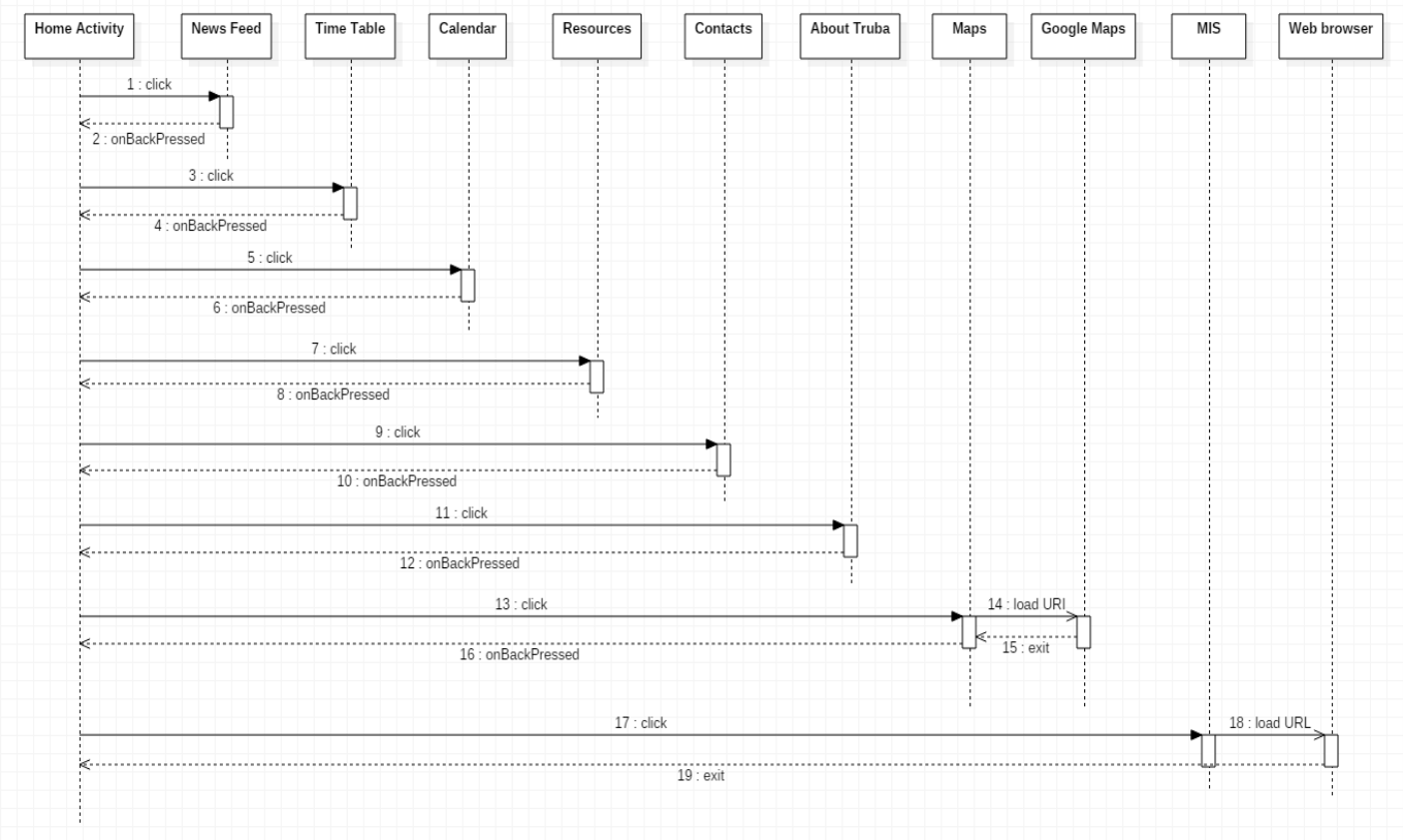
Project folders in android studio –

1. **manifests**—Android Manifest file - description of app read by the Android runtime
2. **java**—Java source code packages
3. **res**—Resources (XML) - layout, strings, images, dimensions, colors...
4. **build.gradle**—Gradle build files

4.2 Sequence diagrams and Activity diagrams.

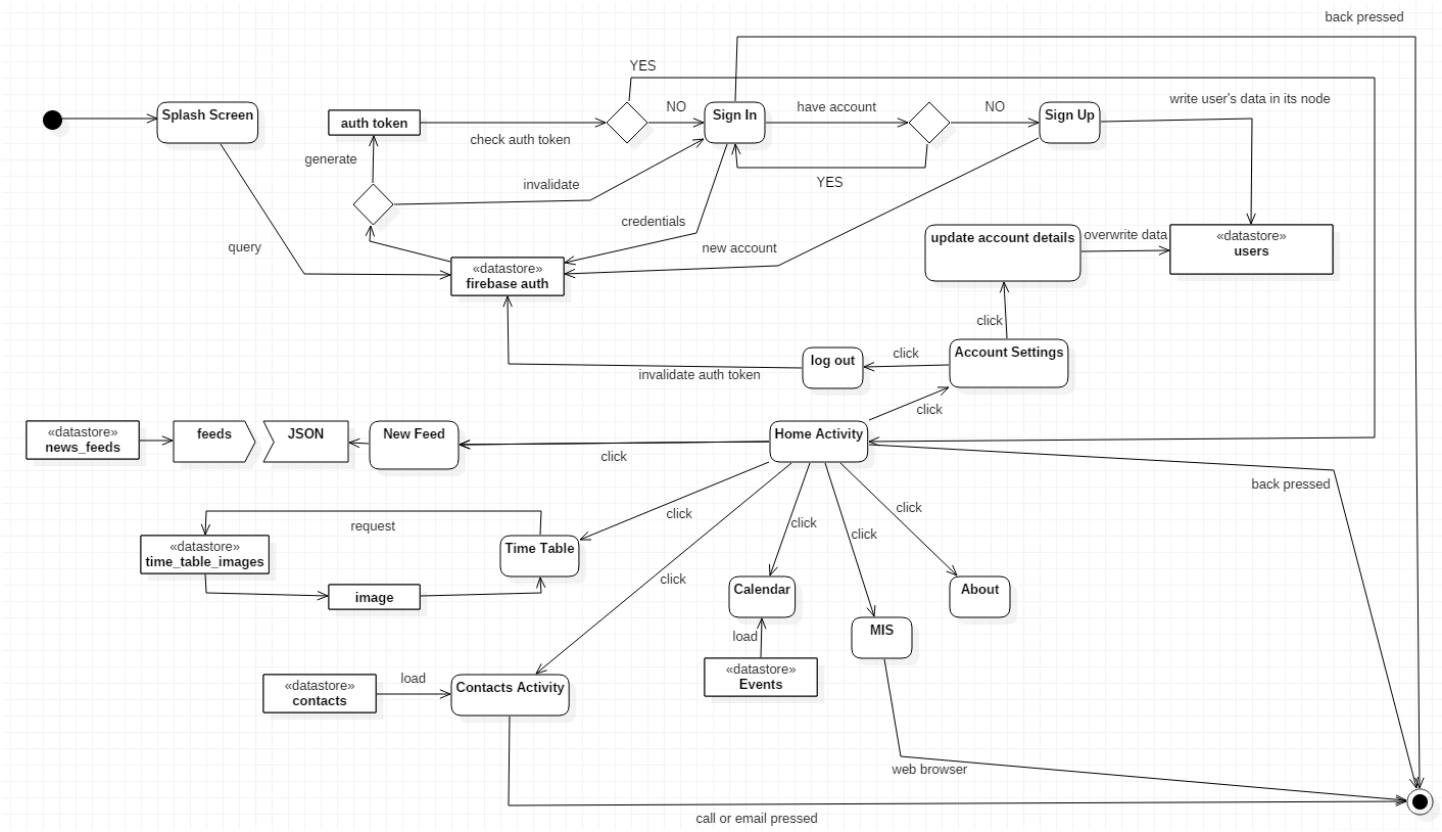


Sequence Diagram of sign in



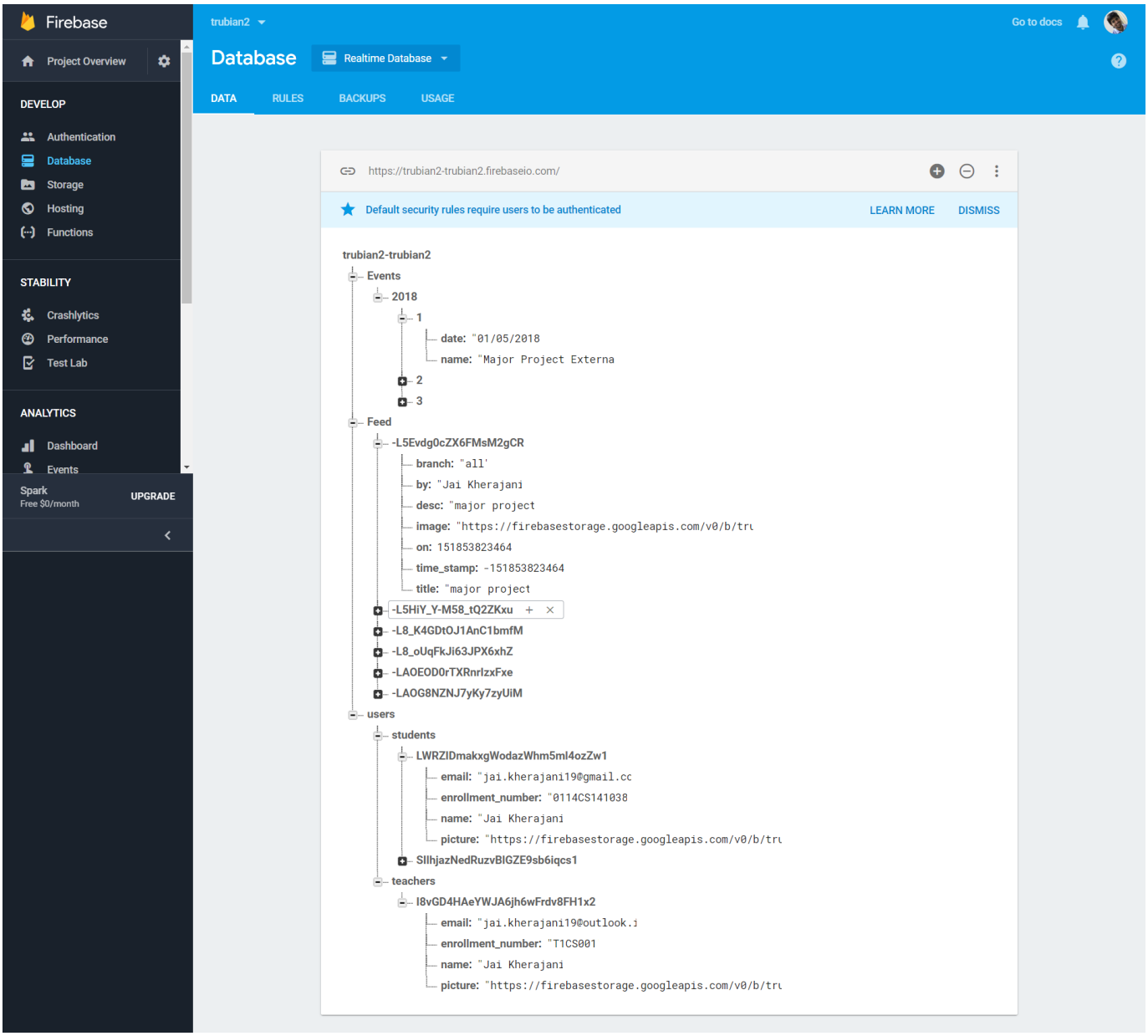
Sequence Diagram for all options

Activity Diagram



4.3 Class Diagram

4.4 Database Design



Firebase Realtime database is a No-SQL database that stores data in JSON format. The data is stored in terms of nodes and can hold any kind of values, any number of values at a particular node irrespective of other nodes. As an be seen from the above picture the details of events are stored at **/Events/{Year}/{unique id}** node, details of accounts are stored at **/users/students/{uid}** for students and **/users/teachers/{uid}** for faculty accounts. Finally, the posts in news feeds are stored at **/Feed/{uid}**.

IMPLEMENTATION & TESTING

5. Implementation and Testing

5.1 Subsystem and Their Dependencies

The subsystems that are included in the application are as follows:

Newsfeed: User can add a post or view post.

Profile: User can view his/her profile and perform various actions like changing password, resetting password, changing profile picture etc.

Time table: User can view time table based on their enrollment number.
Calendar: User can view calendar; all events are stored at a node in firebase database.
MIS: opens MIS URL in a browser
Maps: opens map application and shows college on the map

5.2 Class diagram explanation

A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity.
The purpose of class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction.
UML diagrams like activity diagram, sequence diagram can only give the sequence flow of the application, however class diagram is a bit different. It is the most popular UML diagram in the coder community.
The purpose of the class diagram can be summarized as −

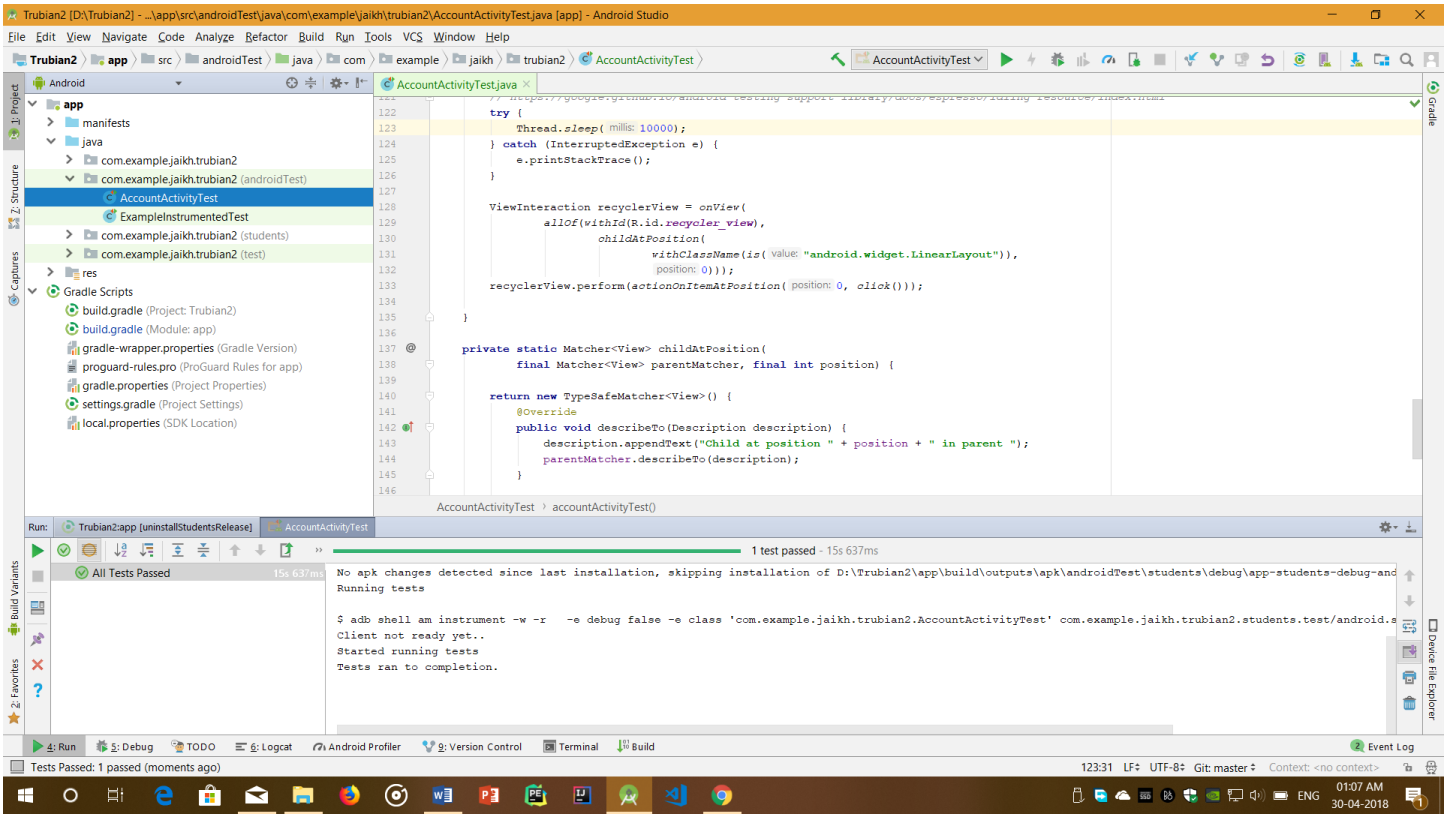
- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

5.3 Expresso Testing

Espresso is a testing framework for Android to make it easy to write reliable user interface tests. Google released the Espresso framework in Oct. 2013. Since its 2.0 release Espresso is part of the Android Support Repository.

Espresso automatically synchronizes your test actions with the user interface of your application. The framework also ensures that your activity is started before the tests run. It also let the test wait until all observed background activities have finished.

It is intended to test a single application but can also be used to test across applications. If used for testing outside your application, you can only perform black box testing, as you cannot access the classes outside of your application.



CONCLUSION

6. CONCLUSION

6.1 Conclusion

Right now, the project comes in only 1 flavor i.e. an android app. But the app can be extended to iOS and other platforms easily. Since the app shares the same data repository, in future by changing the UI or making the app from scratch won't result into loss of data. Also, since the database used is of NoSQL type, in case at later stage we need to add some properties to a specific user group we can do it easily as it is the property of NoSQL that it doesn't follows any schema like relational databases.

6.2 Future Scope

In future, iOS and web clients can be made using the firebase libraries and they don't have to start afresh instead they all will share the common database that is stored in cloud

APPENDIX

1. REFERENCE

https://developer.android.com

https://www.w3schools.com

https://guides.codepath.com

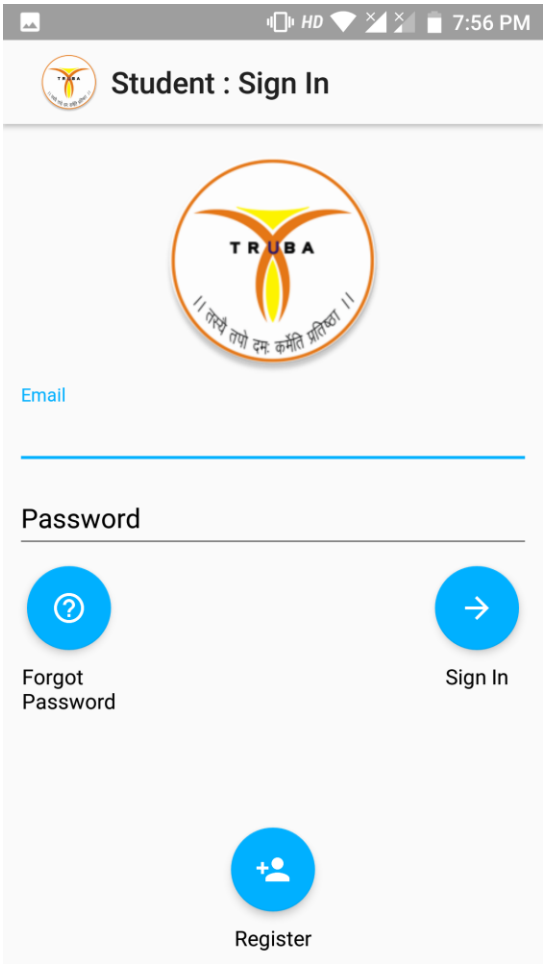
https://www.androidhive.info

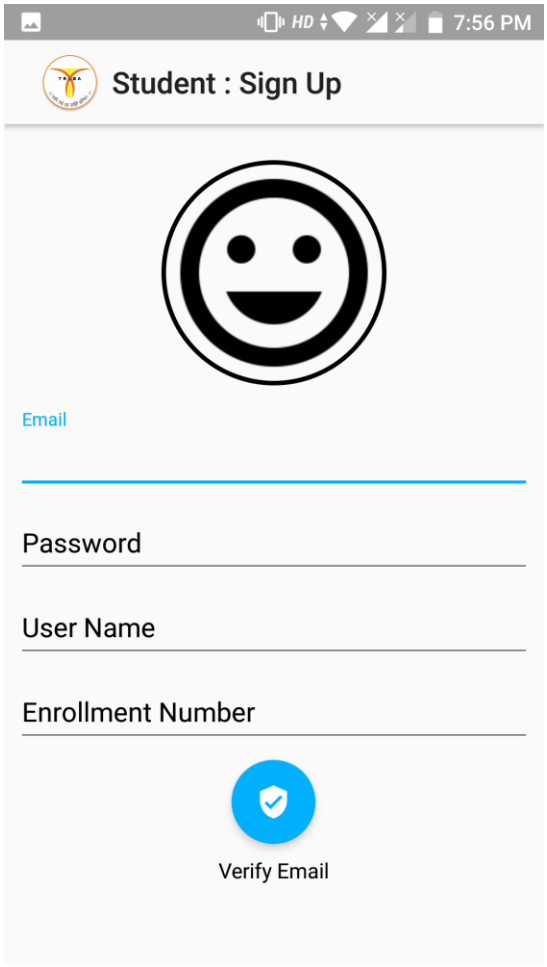
https://firebase.google.com/docs/

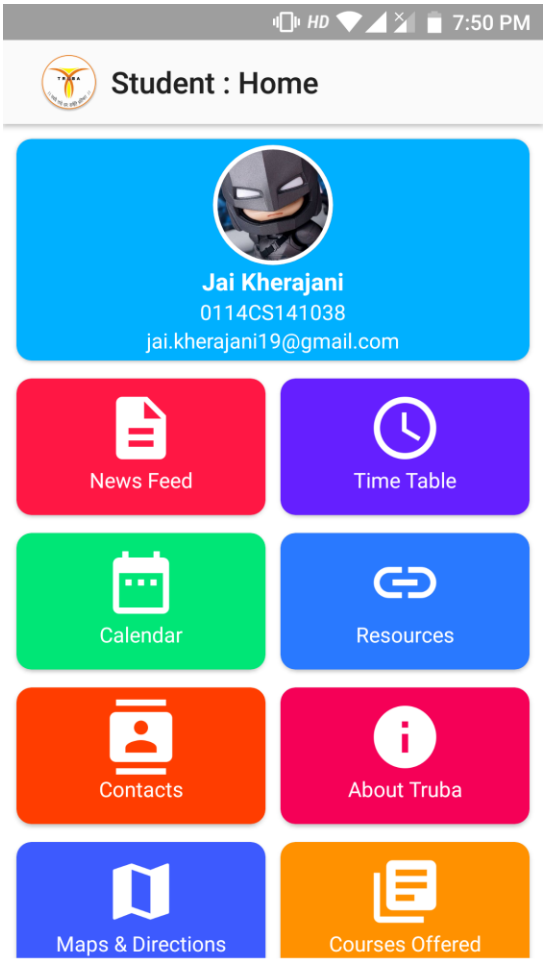
https://in.udacity.com/course/firebase-in-a-weekend-by-google-android--ud0352

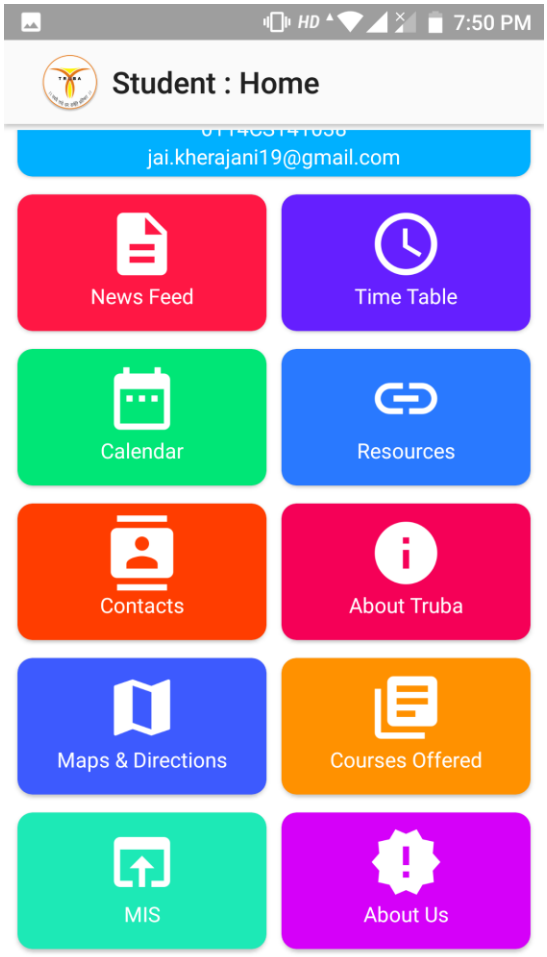
https://codelabs.developers.google.com/codelabs/firebase-android/#0

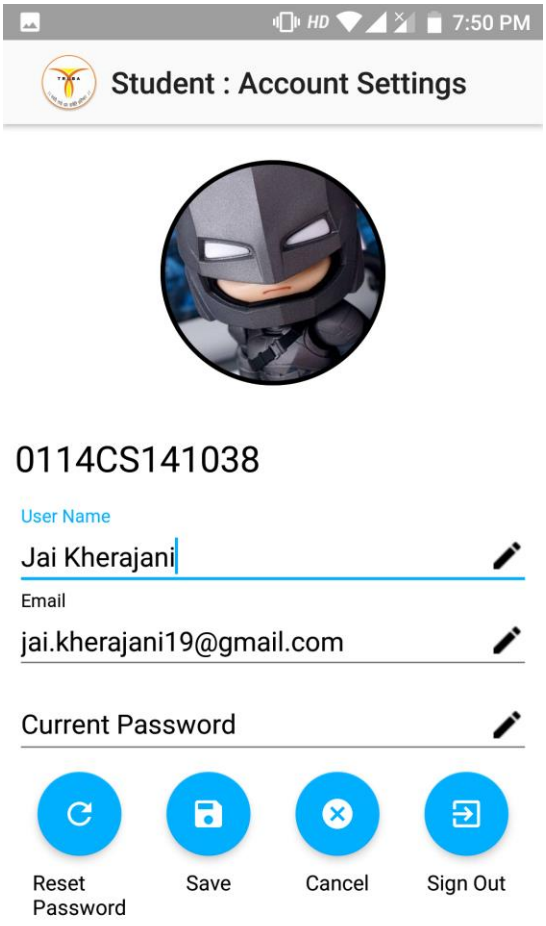
2. Screenshots

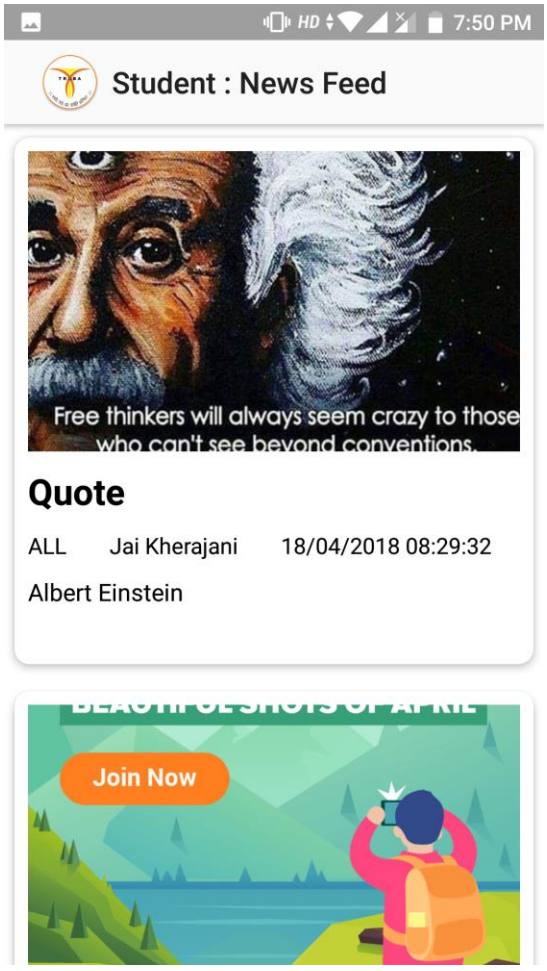


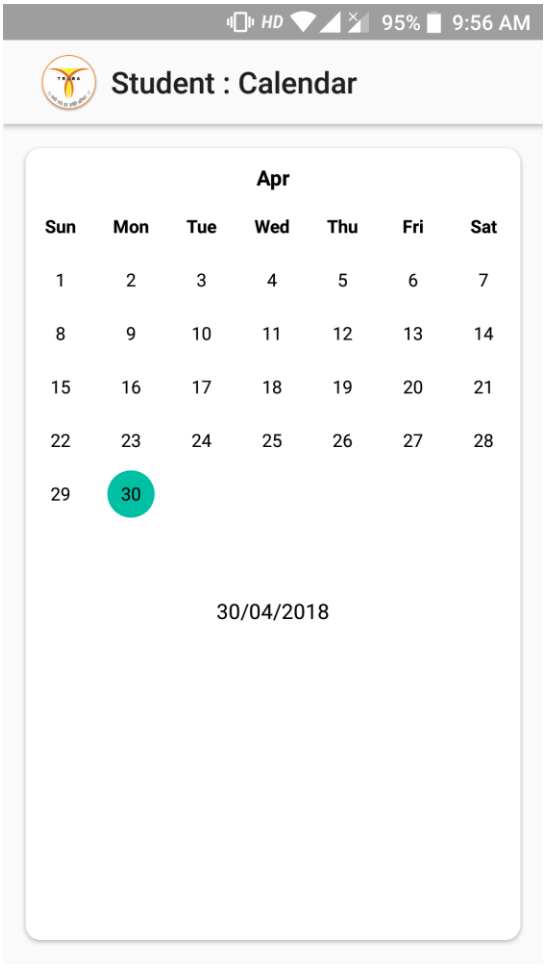


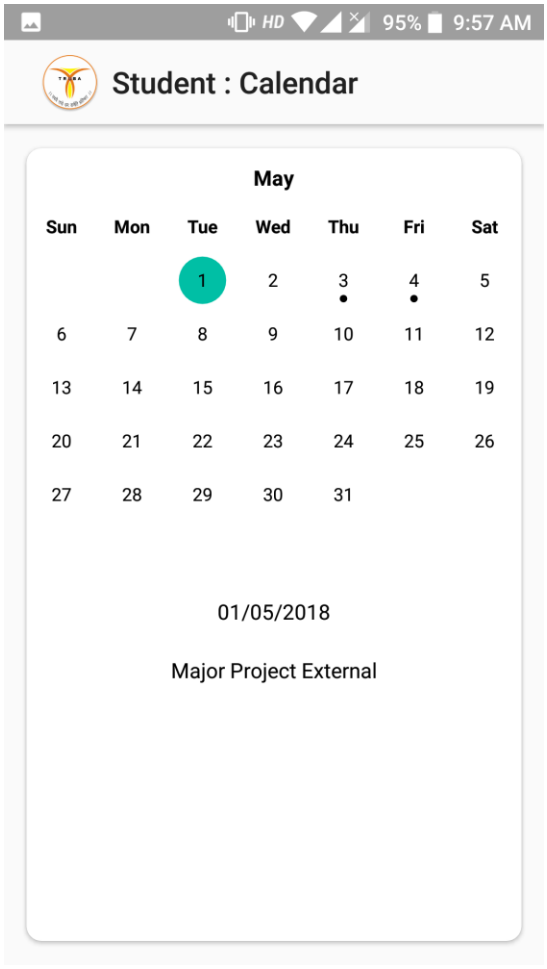

7:50 PM
Student : Home
Jai Kherajani
0114CS141038
jai.kherajani19@gmail.com
News Feed
Time Table
Calendar
Resources
Contacts
About Truba
Maps & Directions
Courses Offered

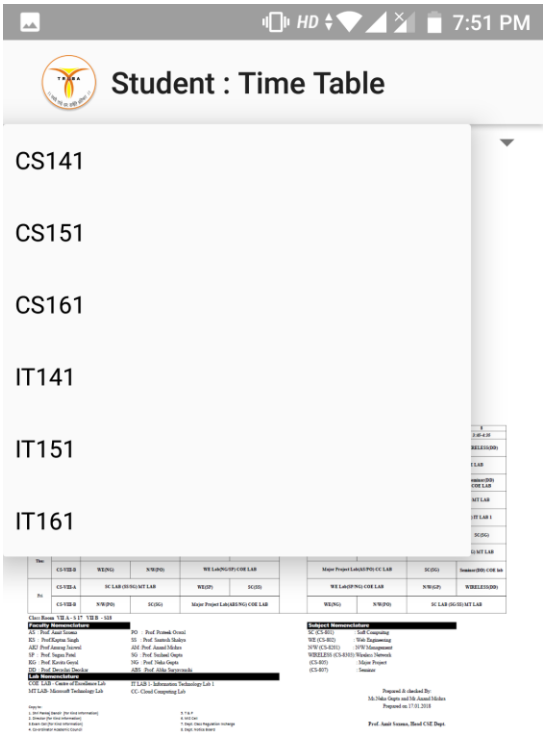

7:50 PM
Student : Home
jai.kherajani19@gmail.com
News Feed
Time Table
Calendar
Resources
Contacts
About Truba
Maps & Directions
Courses Offered
MIS
About Us


7:50 PM
Student : Account Settings
0114CS141038
User Name
Jai Kherajani
Email
jai.kherajani19@gmail.com
Current Password
Reset Password
Save
Cancel
Sign Out


7:50 PM
Student : News Feed
Free thinkers will always seem crazy to those who can't see beyond conventions.
Quote
ALL Jai Kherajani 18/04/2018 08:29:32
Albert Einstein
Join Now

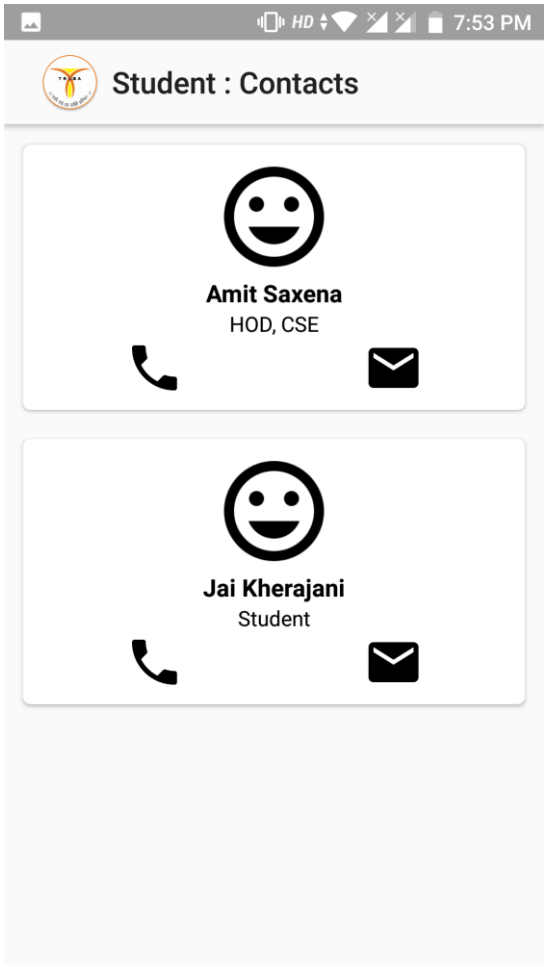

95% 9:56 AM
Student : Calendar
Apr
Sun Mon Tue Wed Thu Fri Sat
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30
30/04/2018

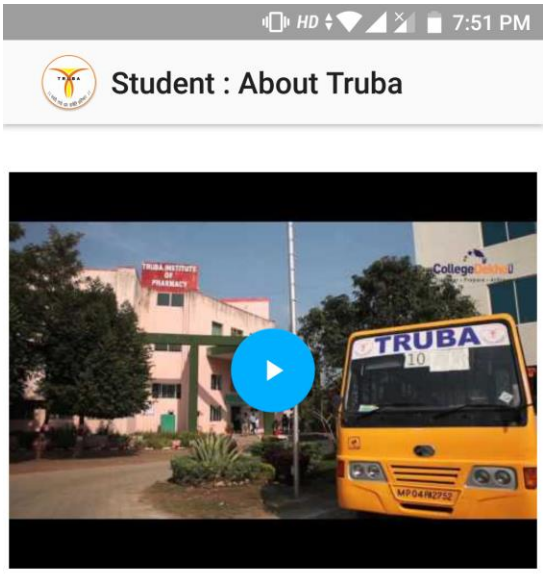

95% 9:57 AM
Student : Calendar
May
Sun Mon Tue Wed Thu Fri Sat
1 2 3 4 5
6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31
01/05/2018
Major Project External


7:51 PM
Student : Time Table
CS141
CS151
CS161
IT141
IT151
IT161


7:50 PM
Student : Time Table
CS141


7:53 PM
Student : Time Table
CS141

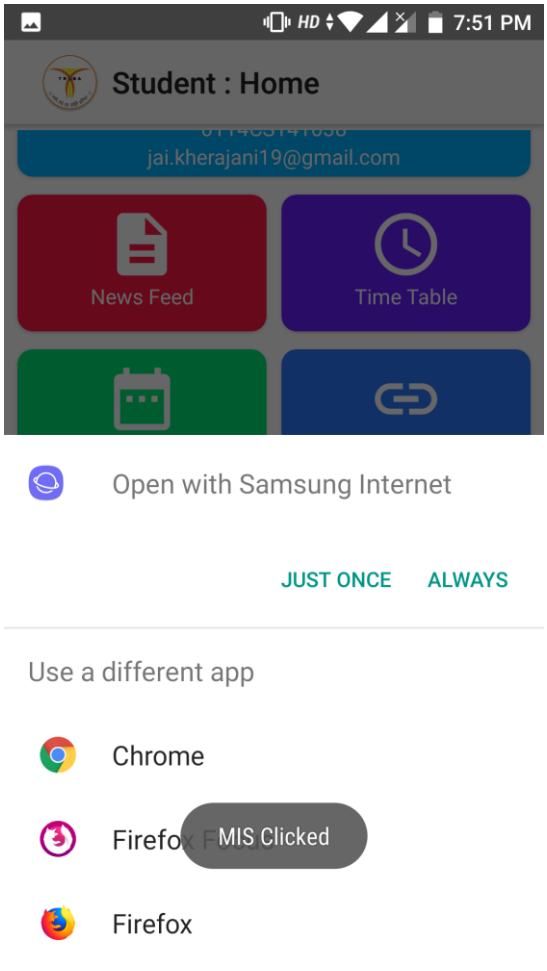

7:53 PM
Student : Contacts
Amit Saxena
HOD, CSE
Jai Kherajani
Student

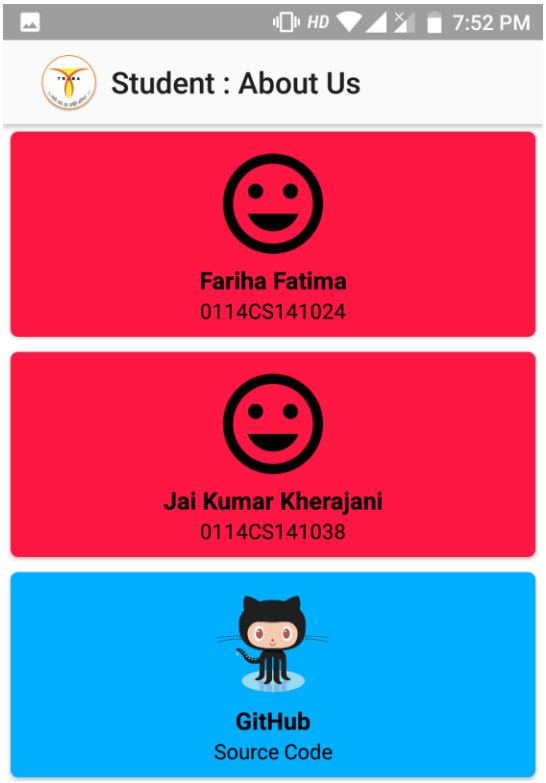


Truba Group: Top engineering college in MP has been set up by Truba Advance Science Kombine (society known as TASKs).

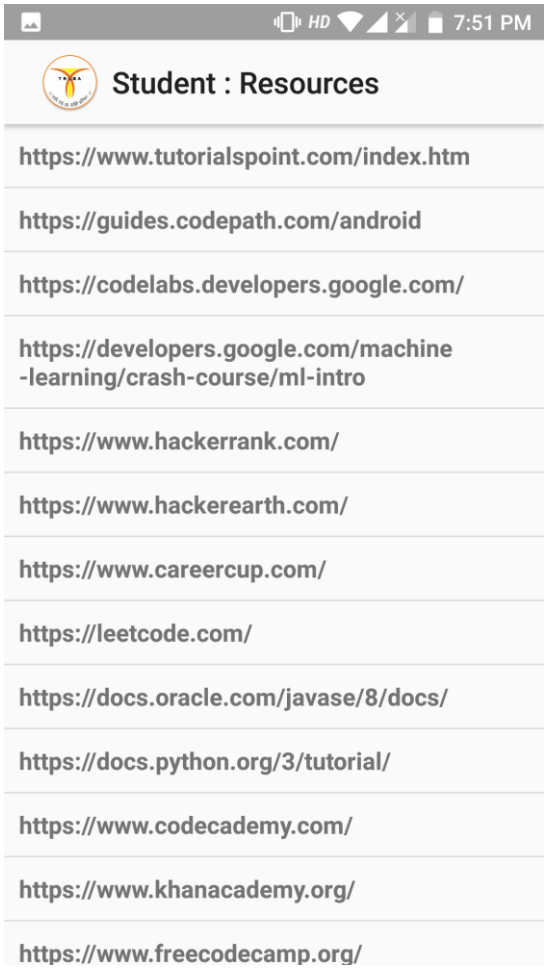
The main reason for the Truba to exist in the world is to impart knowledge to the students who are studying in it not only for obtaining higher marks but for improving students capability and out of the box thinking.

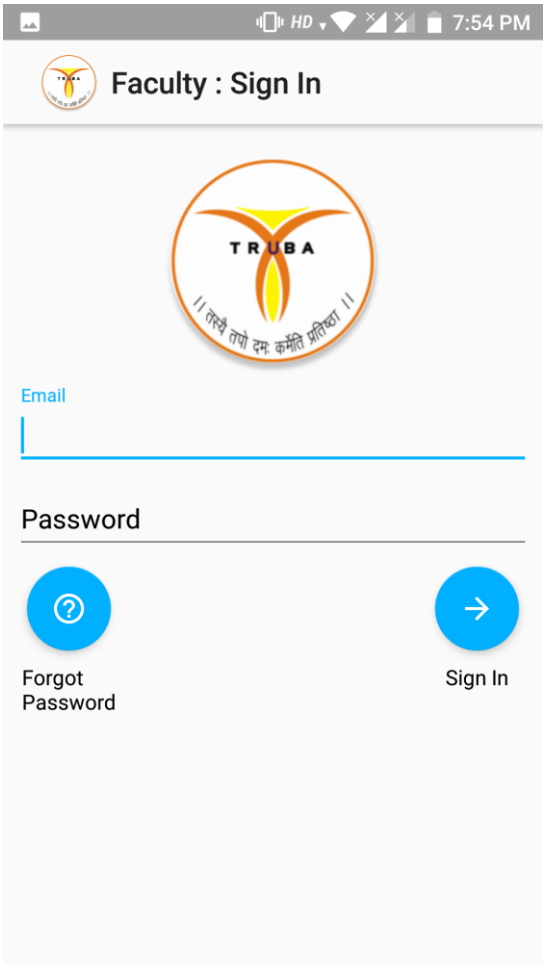
Education is all about creating an environment of academic freedom, where bright minds meet, discover and learn. One would experience top of the world living and learning experience at TRUBA, a place where you have the freedom to flv.

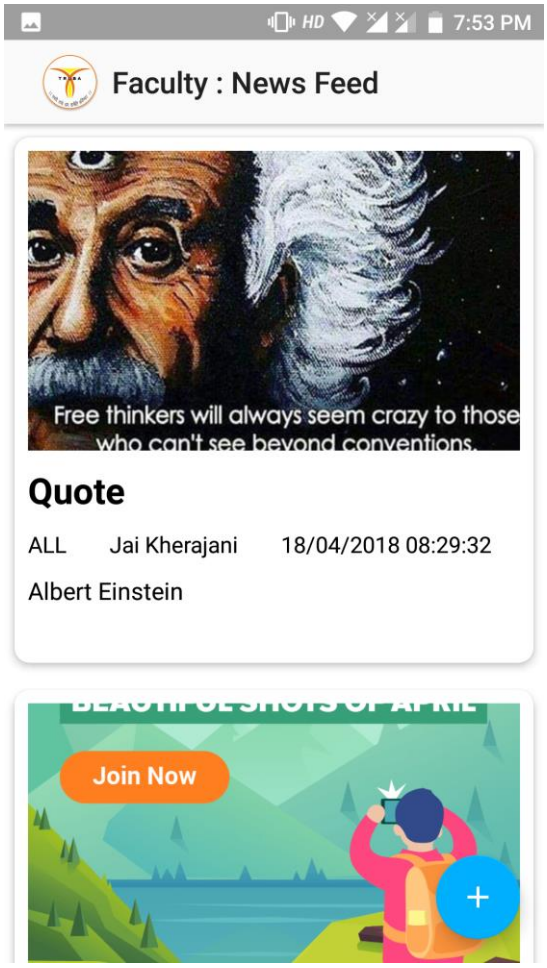


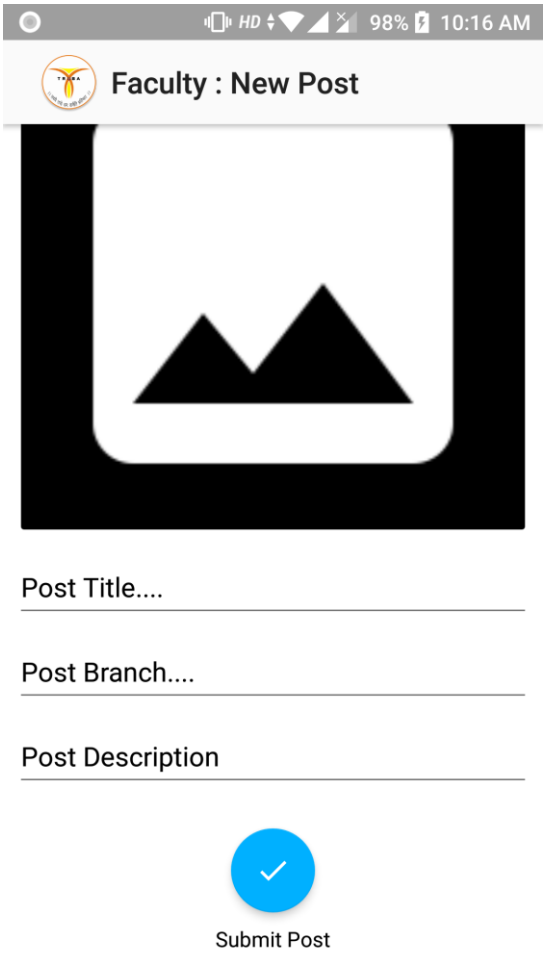


Dedicated to all Trubians. 😍




7:54 PM
Faculty : Sign In
TRUBA
Email
Password
Forgot Password
Sign In


7:53 PM
Faculty : News Feed
Free thinkers will always seem crazy to those who can't see beyond conventions.
Quote
ALL Jai Kherajani 18/04/2018 08:29:32
Albert Einstein
Join Now


98% 10:16 AM
Faculty : New Post
Post Title....
Post Branch....
Post Description
Submit Post


100% 10:26 AM
TRUBA