A Project Report On

Collegify

Submitted by

Hitanshu Panchal (1680026)

Dhruwangi Shrotriya (1680054)

Priyanka Walekar (1680056)

Under the guidance of

Mr. Pratik Shah

Term-December'18 to May'19

Department of Computer Engineering

SVKM's

Shri Bhaghubhai Mafatlal Polytechnic Irla, N.R.G Marg, Vile Parle (W), Mumbai-400057

ACKNOWLEDGEMENT

We take imense pleasure in thanking Professor. Name of Principal, Principal of Shri Bhagubhai Mafatlal Polytechnic for having permitted us to carry out this project work.

We wish to express our deep sense of gratitude to my Internal Guide, Mr.PRATIK SHAH for his able guidance and useful suggestions, which helped us in completing the project work in time. We would like to thank him for his valuable assistance in the project work. Words are inadequate in offering my thanks to the entire staff of I.T/C.S.E for providing us with all amenities and facilities.

The success of this project would not have been possible without the constant encouragement, advice and support from a vast number of people. Ideas they say remain ideas. Very few of them,if find support are turned into working models. And the rest of them are forgotten and remain what they started off as ideas. This idea as well would have remained, but for one person, The real debt of gratitude that we owe to him without whom this project would have not seen light of the day. We, the members of the team, who developed "COLLEGIFY" are very delighted to take the opportunity to acknowledge whole heartedly the innumerable guidance and support extended to us by our guide.

Finally, yet importantly, we would like to express our heartfelt thanks to our beloved parents for their blessings, our friends/classmates for their help and wishes for the successful completion of this project.

ABSTRACT

With the increase in demand for smartphones and efficiency of wireless networks, the demand for mobile application has increased incredibly. Android is one of the most popular open source platforms that offer the developer's full access to the framework API's so as to build innovative applications.

The project is focused on providing users with an interface to search for colleges and get suitable information. This application has a feature to provide user a way to connect with the college in their region. Also, the user will be able to browse through photos, reviews and ratings and bookmark the colleges.

Contents

List of Figures

1 INTRODUCTION

The era of mobile technology opens the window to android app. The websites are diminishing, and the mobile phones are emerging. It's the time to change from conventional website to apps, which has become the daily part of our routine. We are introducing "Collegify" to provide people with information regarding the colleges in Mumbai. It recommends the nearby colleges to their residents and provides the GPS location of the colleges. Our application tries to connect to the people and provide them basic details like phone numbers, directions, college websites, courses offered, address of the respective colleges and GPS location.

This application is also beneficial for the users who are new to Mumbai City and have less knowledge about the colleges, so by using our application, they can find different colleges and its information. User gets recommendation as per the GPS location. Also provides nearby bus stops to colleges. One can also bookmark the college they like. In this way, our application provides the information about different colleges to the parents or student in more organized manner.

2 GANTT CHART

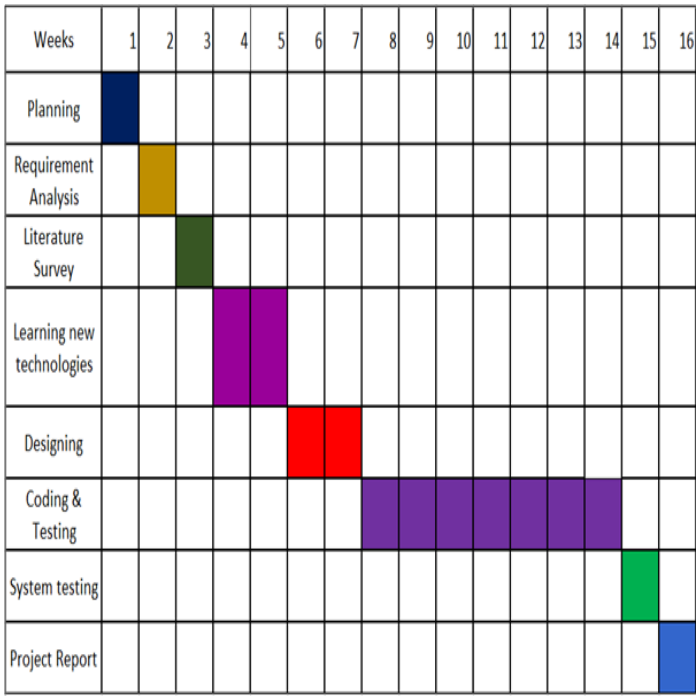


Figure 1: Gantt Chart

3 PROBLEM DEFINITION

In any field, even after scoring well in secondary/higher secondary education, the biggest problem in front of students and their parents is finding the right college for the field they want. Also, for students with average score it becomes difficult to find colleges. It happens that for searching the appropriate colleges the students have to search on the internet extensively which consumes a lot of time.

Information such as a list of all the colleges of different fields, their cut off lists and nearby colleges near to the student's resident, which sometimes may not be easily available.

Rather than searching the web for basic information about the college, Collegify is an application that will help these students to get the most suitable college for them in whatever field they are aiming to develop their career. This will reduce the efforts as well as the stress of the students and their parents. The application will also provide filter option through the user can sort out colleges as per his/her requirements.

3.1 Feasibility Study

1. Technical feasibility

    - The technologies used to develop the application are Android studio and Firebase. Firebase is Google's mobile platform that provides backend as a service. Android studio is an open source software that allows to develop an application easily. The project can work with an active internet connection, it doesn't require any special infrastructure. The maintenance of the application requires keeping the database , in this case the real-time firebase database, clean and updated.

2. Economic feasibility

    - The software Android Studio thats is used to develop the application is open source and free to use. However, Firebase which provides the backend service will need to have a paid plan and also the Google Places API will impose some charges for its services. The application can earn using in app advertisement and also in future can implement a feature to collaborate with colleges for offline admission forms.

3. Market feasibility

    - In todays busy world, there is very less time for people to research about the colleges or the courses each colleges provide. It is very hectic for parents or students to find different colleges that provide courses they are looking for.

4. Schedule feasibility

    - The given deadline for the development of Collegify was 16 weeks. There were many unplanned activities that happen during this development period however all the major features of the android application were successfully developed. Hence, it was feasible to develop the application in the given time.

4 LITERATURE SURVEY

4.1 Introduction To Android OS

Android is an open source and Linux-based Operating System for mobile devices such as smartphones and tablet computers. Android is a software bunch comprising not only operating system but also middleware and key applications. Android Inc. was founded in Palo Alto, California, in October 2003 by Andy Rubin, Rich Miner, Nick Sears, and Chris White.In July 2005, Google acquired Android Inc. After the original release there have been number of updates in the original version of Android. The goal of android project is to create a successful real-world product that improves the mobile experience for end users. Android is a powerful OS supporrting a large number of applications in smartphones. These applications make life more comfortable and advanced for the users. Hardware that support Android are mainly based on ARM architecture platform

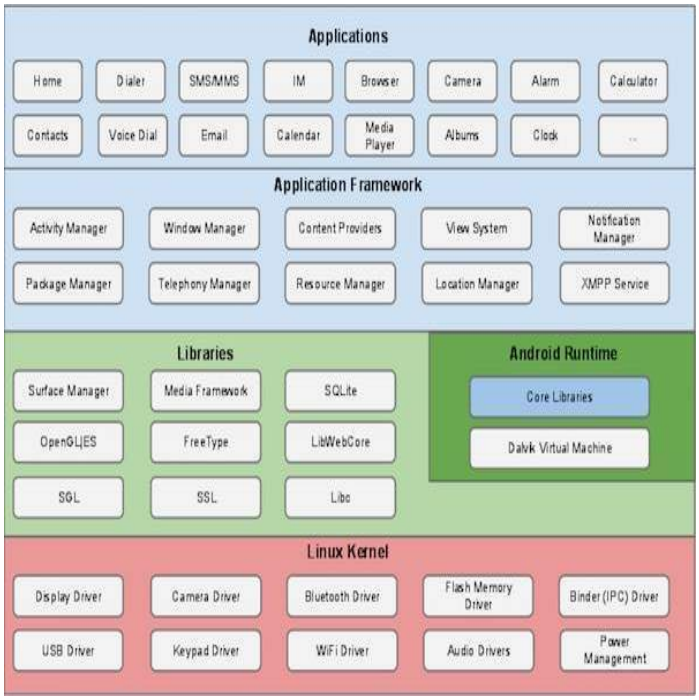


Figure 2: Android Architecture

4.1.1 Key Features of Android OS

Android is a powerful operating system competing with Apple 4GS and supports great features.

1. Beautiful UI
   - Android OS basic screen provides a beautiful and intuitive user interface.
2. Connectivity
   - GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC and WiMAX.
3. Storage
   - SQLite, a lightweight relational database, is used for data storage purposes.
4. Media support
   - H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, and BMP.
5. Messaging
   - SMS and MMS.
6. Web browser
   - Based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine supporting HTML5 and CSS3.
7. Multi-touch
   - Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero.
8. Multi-tasking
   - User can jump from one task to another and same time various application can run simultaneously.
9. Resizable widgets
   - Widgets are resizable, so users can expand them to show more content or shrink them to save space.
10. Multi-Language
    - Supports single direction and bi-directional text.
11. GCM
    - Google Cloud Messaging (GCM) is a service that lets developers send short message data to their users on Android devices, without needing a proprietary sync solution.
12. Wi-Fi Direct
    - A technology that lets apps discover and pair directly, over a high-bandwidth peer-to-peer connection.
13. Android Beam
    - A popular NFC-based technology that lets users instantly share, just by touching two NFC-enabled phones together.

4.2 Android Applications

Android applications are usually developed in the Java language using the Android Software Development Kit. Once developed, Android applications can be packaged easily and sold out either through a store such as Google Play, SlideME, Opera Mobile Store, Mobango, F-droid and the Amazon Appstore. Android powers hundreds of millions of mobile devices in more than 190 countries around the world. It's the largest installed base of any mobile platform and growing fast. Every day more than 1 million new Android devices are activated worldwide.

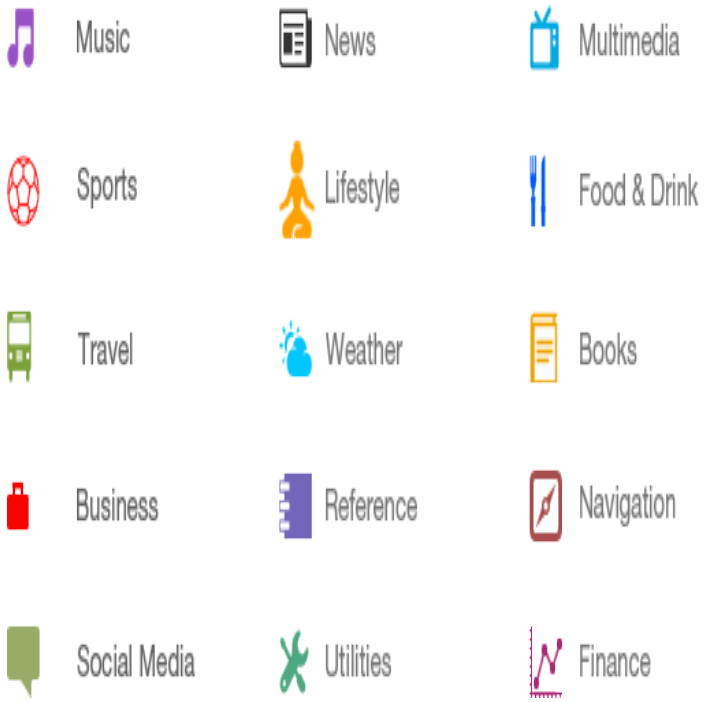


Figure 3: Categories of Android Application

4.2.1 Components Of Android Application

Application components are the essential building blocks of an Android application. These components are loosely coupled by the application manifest file AndroidManifest.xml that describes each component of the application and how they interact. There are following four main components that can be used within an Android application-

1. Services
   - They handle background processing associated with an application.
2. Activities
   - They dictate the UI and handle the user interaction to the smart phone screen.

3. Broadcast Receivers
    - They handle communication between Android OS and applications.
4. Content Providers
    - They handle data and database management issues.

4.3 Firebase

Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014. Firebase helps you quickly develop high-quality apps and grow your business. It provides tools and infrastructure to enable a developer to deliver functionality across multiple platforms more easily and efficiently. The product range mainly addresses application developers and vendors who can use a SDK to address various APIs, and thus the features offered by Firebase, and use them in their application.

4.3.1 Products And Services by Firebase

Firebase offers as a platform various, partly free, services and tools. In the following the individual offers are listed and described:

1. Analytics
    - Analytics is Firebase's flagship product and offers unlimited analytics solutions for free. Unlimited in this case means that the solutions provided by Firebase can collect and manage an unlimited amount of Analytics data. There are 500 event types with up to 25 attributes available. The collected data can be evaluated in dashboards and various graphics and thus provide information about user behaviour. In addition, an export of the data to the Google Cloud Platform service Big Query is possible to run own evaluations on the data. The use of Big Query is not free and is not included in Firebase, but must be billed separately.
2. Cloud Messaging
    - Formerly known as Google Cloud Messaging (GCM), Firebase Cloud Messaging (FCM for short) provides a cross-platform way to deliver and receive notifications for android, iOS, and the Web using JavaScript.
3. Authentication
    - Firebase Authentication provides a cross-platform system for different authentication options. Included are the registration with Facebook, Twitter, Google, GitHub as well as username and password. The integration into an existing authentication system should be possible without any problems.
4. Real-time database
    - With the NoSQL real-time database, data can be saved as JSON and synchronized across multiple devices. Access is also possible when the device is currently offline.
5. File storage
    - A cloud-based storage for user-generated data. The data is stored in Google Cloud Storage.

6. Hosting
    - With hosting, Firebase provides a content delivery network (CDN) available for web platforms to deliver resources faster to users (e.g. images or JavaScript code).
7. Test Lab
    - A test platform for Android apps.
8. Crash Reports
    - Firebase's Crash Reporting service provides the ability to gather detailed information if the app crashes due to an unexpected error on a device or if the system stops responding when it stops responding.
9. Notifications
    - Send notifications as a method to increase user interaction with an app.
10. Remote Configuration
    - Change configuration parameters of the app without having to create a new release and distribute it through the Google Play Store.
11. App Indexing
    - Helps to make the content of the app index able in Google Search, thus increasing user interaction and finding app content.
12. Dynamic Links
    - Dynamic links are ultimately the same as deep links to specific content in the app. The difference is that these dynamic links also survive the app's installation process, leading new users to the content they wanted to see before installing the app.
13. Invitations
    - An out-of-the-box solution to allow existing users to share an app and invite friends or acquaintances to the app.

4.4 Version Control System Using Git

Git is the best choice for most software teams today. While every team is different and should do their own analysis, here are the main reasons why version control with Git is preferred over alternatives:

1. Git is good
    - Git has the functionality, performance, security and flexibility that most teams and individual developers need. These attributes of Git are detailed above. In side-by-side comparisons with most other alternatives, many teams find that Git is very favourable.
2. Git is a de facto standard
    - Git is the most broadly adopted tool of its kind. This is makes Git attractive for the following reasons. Vast numbers of developers already have Git experience and a significant proportion of college graduates may have experience with only Git. While some organizations may need to climb the learning curve when migrating to Git from another version control system, many of their existing and future developers do not need to be trained on Git. In addition to the benefits of a large talent pool, the predominance of Git also means that many third party software tools and services are already integrated with Git including IDEs, and our own tools like DVCS desktop client Source tree, issue and project tracking software, Jira, and code hosting service, Bit bucket.
3. Git is a quality open source project
    - Git is a very well supported open source project with over a decade of solid stewardship. The project maintainers have shown balanced judgment and a mature approach to meeting the long term needs of its users with regular releases that improve usability and functionality. The quality of the open source software is easily scrutinized and countless businesses rely heavily on that quality. Git enjoys great community support and a vast user base. Documentation is excellent and plentiful, including books, tutorials and dedicated web sites. There are also podcasts and video tutorials.

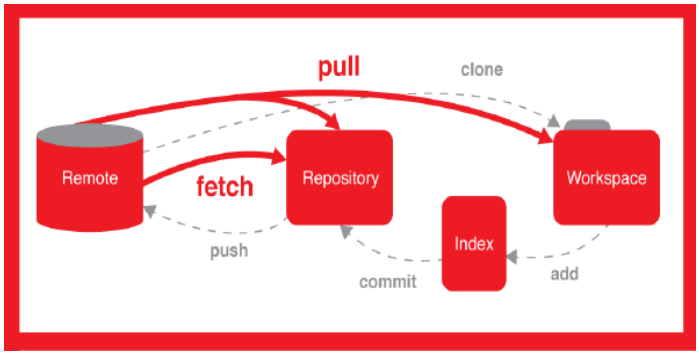


Figure 4: Basic Architecture of Git

4.5 GitHub

GitHub is the single largest host for Git repositories, and is the central point of collaboration for millions of developers and projects. GitHub is a Web-based Git version control repository hosting service. It is mostly used for computer code. It offers all of the distributed version control and source code management (SCM) functionality of Git as well as adding its own features. It provides access control and several collaboration features such as bug tracking, feature requests, task management, and wikis for every project.

4.6 Google APIs

Google APIs is a set of application programming interfaces (APIs) developed by Google which allow communication with Google Services and their integration to other services. Third-party apps can use these APIs to take advantage of or extend the functionality of the existing services. The APIs provide functionality like analytics, machine learning as a service (the Prediction API) or access to user data (when permission to read the data is given). Another important example is an embedded Google map on a website, which can be achieved using the Static maps API, Places API or Google Earth API.

4.6.1 Google Play Services

With Google Play services, applications can take advantage of the latest, Google-powered features such as Maps, Google+, and more, with automatic platform updates distributed as an APK through the Google Play store. This makes it faster for your users to receive updates and easier for you to integrate the newest that Google has to offer. Google Play services gives you the freedom to use the newest APIs for popular Google services without worrying about device support. Updates to Google Play services are distributed automatically by the Google Play Store and new versions of the client library are delivered through the Android SDK Manager. This makes it easy for you to focus on what's important: your users' experience.

4.6.2 Google Maps API

Build full-featured Android apps for users. Google Maps APIs for Android are available via Google Play services so your app can be location-aware, include data-rich maps, find relevant places nearby and more. Add maps to your Android app. Integrate base maps, 3D buildings, indoor floor plans, Street View and Satellite imagery, custom markers and more.

4.6.3 Google Places API

With the Google Maps Android API, maps can be added based on Google Maps data to your application. The API automatically handles access to Google Maps servers, data downloading, map display, and response to map gestures. You can also use API calls to add markers, polygons, and overlays to a basic map, and to change the user's view of a particular map area. These objects provide additional information for map locations, and allow user interaction with the map.

5 LIBRARIES

5.1 Android Support Library

When developing apps that support multiple API versions, developers may want a standard way to provide newer features on earlier versions of Android or gracefully fall back to equivalent functionality. Rather than building code to handle earlier versions of the platform, developers can leverage these libraries to provide that

compatibility layer. In addition, the Support Libraries provide additional convenience classes and features not available in the standard Framework API for easier development and support across more devices.

Library: com.android.support:appcompat-v7:28.0.0

5.2 FirebaseAuth Library

Firebase Authentication, you can outsource your entire authentication system to Firebase so that you can concentrate on building great features for your app. Firebase Authentication makes it easier to get your users signed-in without having to understand the complexities behind implementing your own authentication system. It offers a straightforward getting started experience, optional UX components designed to minimize user friction, and is built on open standards and backed by Google infrastructure. Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to your app. It supports authentication using passwords, phone numbers, popular federated identity providers like Google, Facebook and Twitter, and more.Firebase provides Authentication service that you can use-

1. Authenticate users using Email/password or Google/Facebook/Twitter /Phone (SMS) etc. social authentication.
1. It creates Identity for the users. Manages password resets. It lets you switch between social auth to username/password.
1. Its integrated with Firebase, and other Google Cloud Services like Cloud Storage. So yo can easily enforce authorization rules based on the same identity.

Library: com.google.firebase:firebase-auth:16.1.0

5.3 Firebase Database

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.Firebase Realtime Database is a cloud-hosted NoSQL database that lets you store and sync data between your users in realtime.

Library: com.google.firebase:firebase-database:16.0.6

5.4 Picasso

Picasso allows for hassle-free image loading in your application—often in one line of code!

Library: com.squareup.picasso:picasso:2.71828

Name: Picasso

Use: To download images through URL.

Contributors: 85 contributors

Link: https://github.com/square/picasso

5.5 RecyclerView Library

The RecyclerView is a new ViewGroup that is prepared to render any adapter-based view in a similar way. It is supossed to be the successor of ListView and GridView, and it can be found in the latest support-v7 version.

Library: com.android.support:recyclerview-v7:28.0.0

6 SYSTEM DESIGN

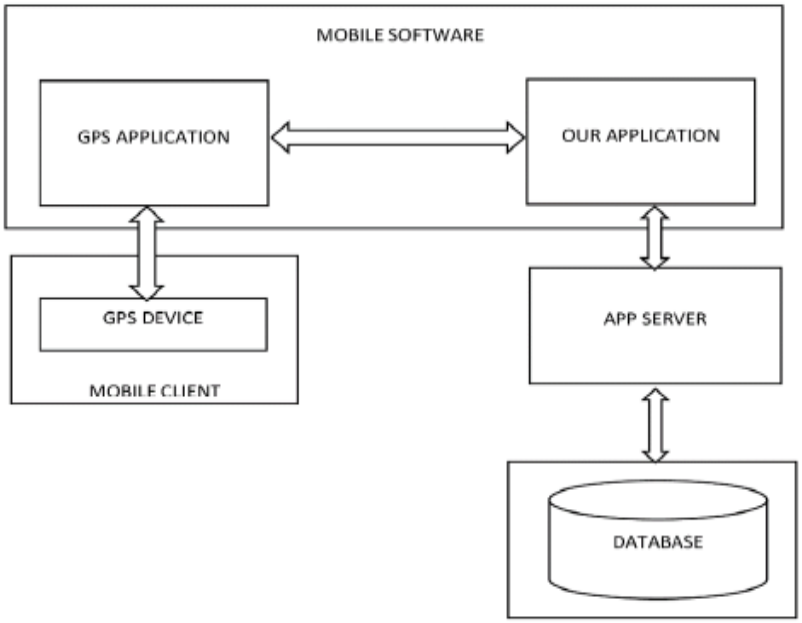
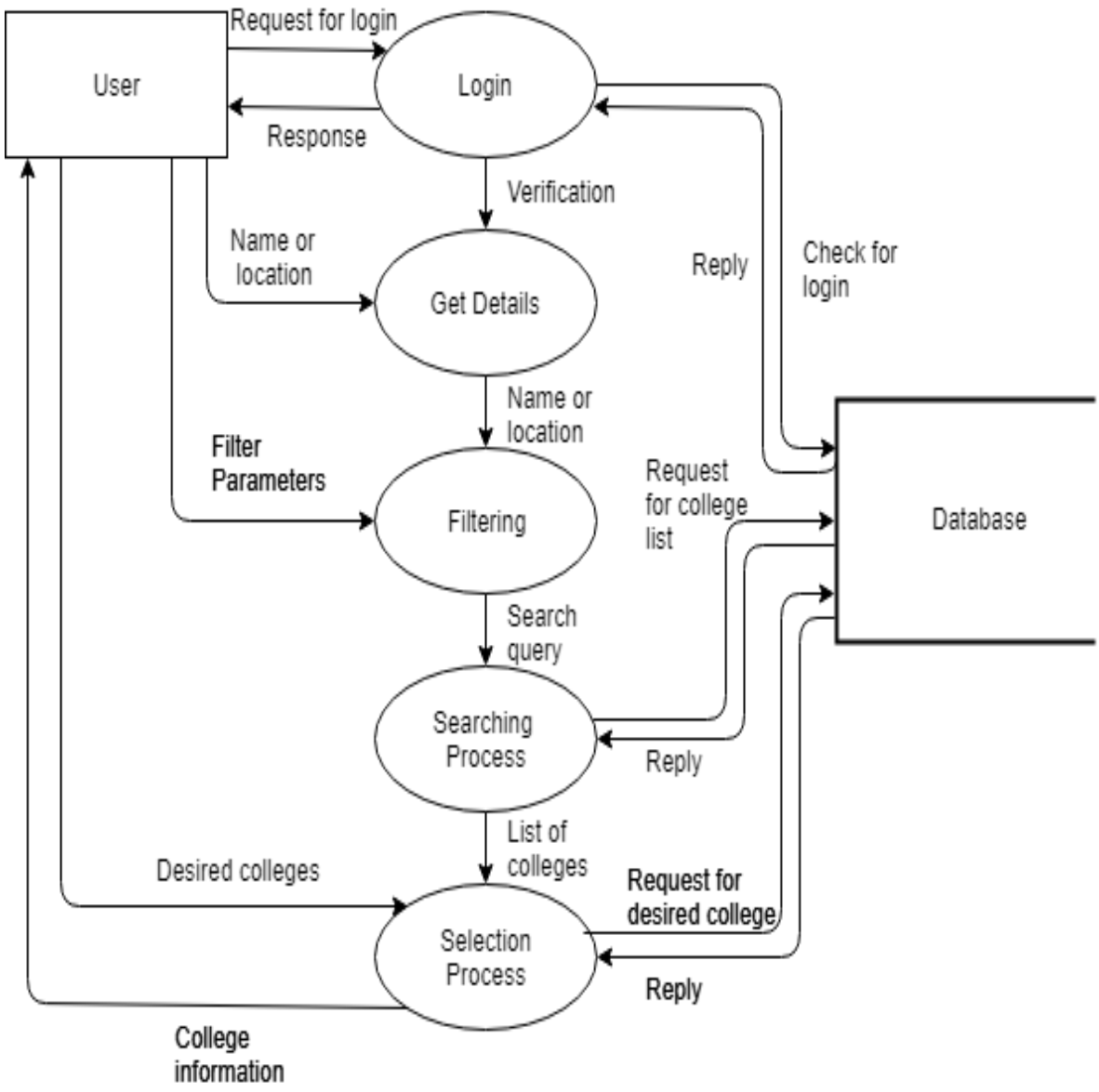


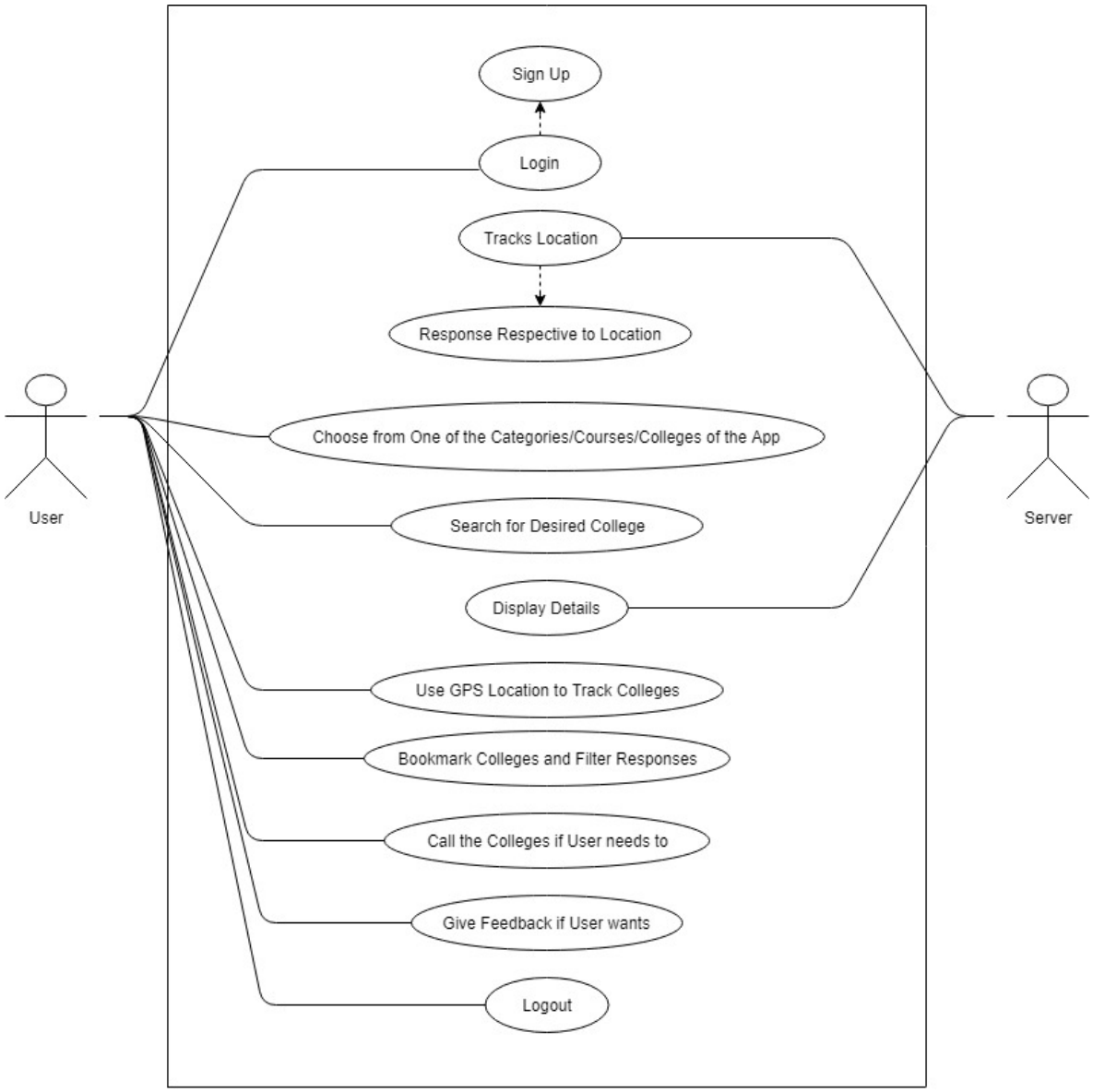
Figure 5: Basic Architecture of the Application

1. The user will have to register himself/herself with email and phone number to the application. A two-step authentication will be done using One Time Password (OTP) and the OTP will be sent to the phone number of the user. The OTP will be generated and the details of the user will be stored in the database at the server side.
2. The user's location will be detected by Google Maps API and it will be sent to server. Colleges located near the user's location will be detected and the location of the colleges will be responded to the application. The user will have to choose the type of college on that, also the desired courses and accordingly colleges will be displayed along with the necessary information. Also user can filter the type of colleges and courses.
3. The location of the colleges will be displayed to the user in the application. When the user will click on the location of the colleges, he/she will be redirected to the Google Maps.

6.1 Flowchart



6.2 Use Case Diagram

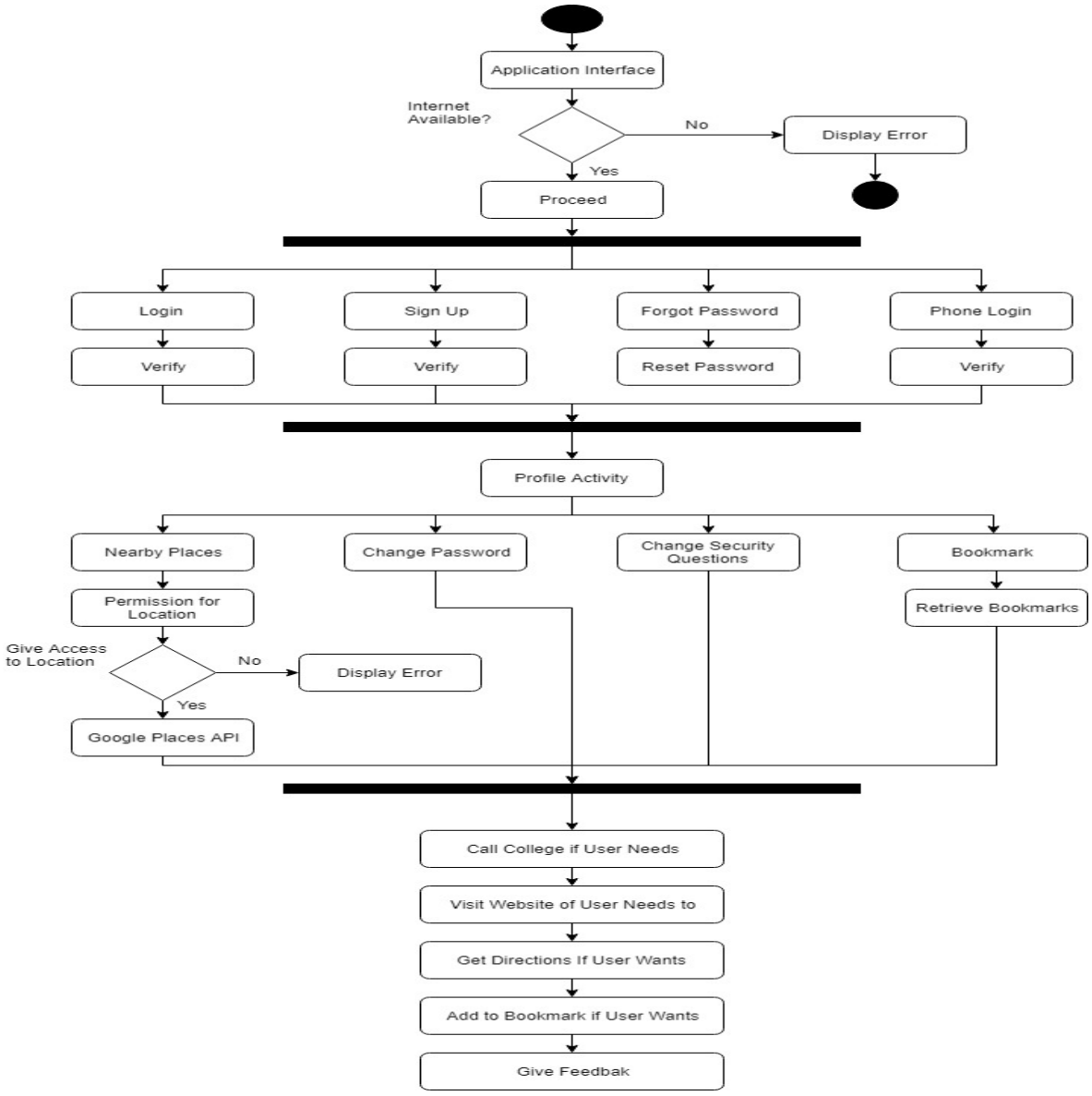


1. A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

2. This use case describes the overall structure of our application with different interactive functions of the user and use cases of the diagram.

3. Here the application provides functionalities like login, signup, different modules, map view, ratings, review and this diagram shows somewhat exact replication of interaction between users and use cases.

6.3 Activity Diagram



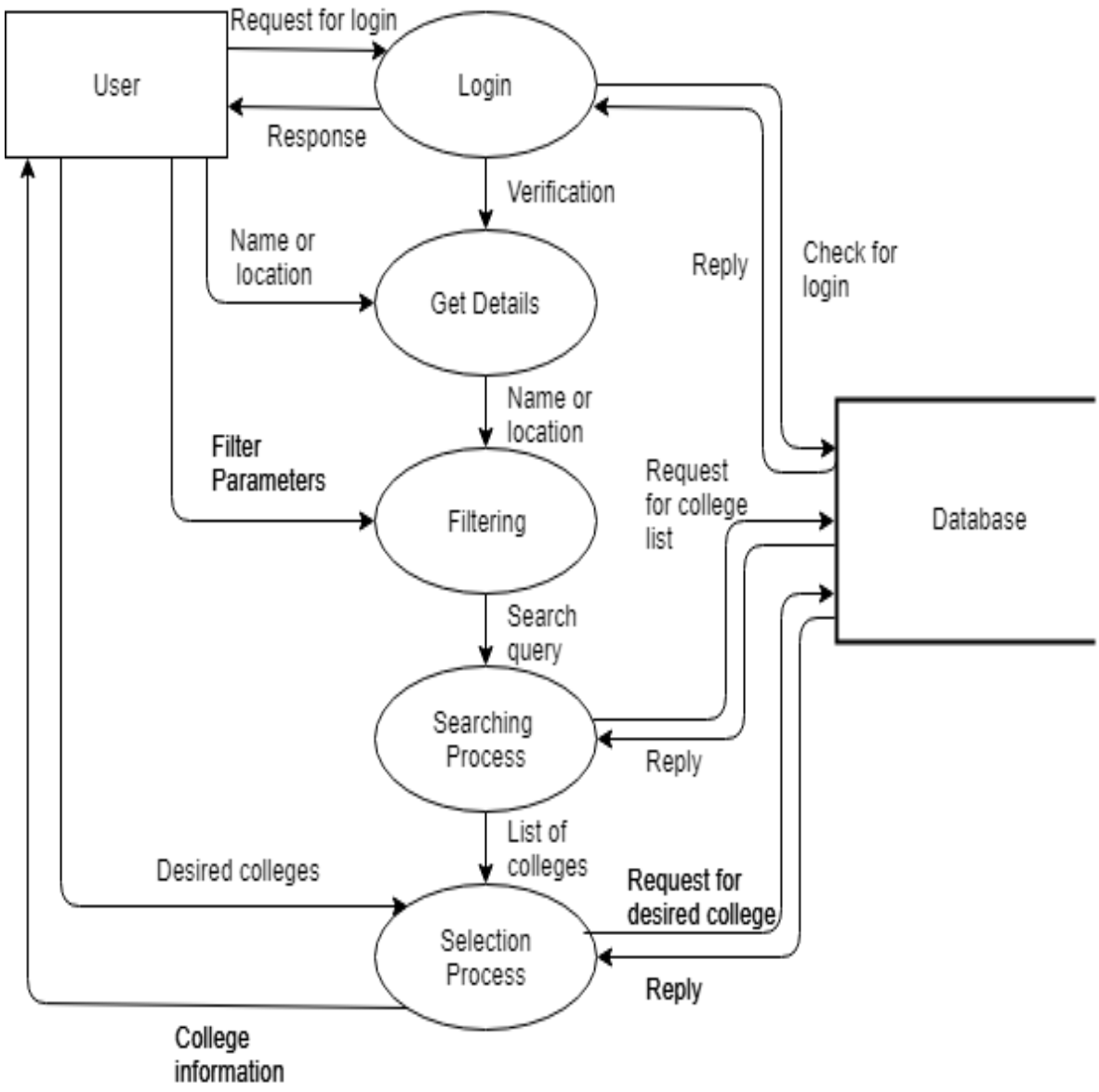
1. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

2. The control flow is drawn from one operation to another. This flow can be sequential, branched, or

concurrent.

3. This activity diagram describes the overall operations/activities of our application.
4. The flow of our activity diagram is sequential.
5. It has many different operations like login, signup, reviews, etc.
6. It deals with all flow control with different elements such as fork, join, etc.

6.4 Data Flow Diagram



7 DESIGN AND IMPLEMENTATION

7.1 Requirements

7.1.1 Hardware Requirements

1. Mobile Phone
   - The mobile phone should be an Android powered phone.
   - The mobile phone should have at least Android API 19 (Kitkat).
   - The mobile phone should have an Internet Connection.
2. For Development
   - Microsoft Windows 7/8/10 (32- or 64-bit).
   - 3GB RAM minimum.
   - 2GB of available disk space minimum.
   - i3 Processor.

7.1.2 Software Requirements

1. Android Studio:

   It will be used to design the layout of all the user-interface screen. Java will be used to develop the application for passing different values between different activities and intents. 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 Firebase Cloud Messaging (Earlier 'Google Cloud Messaging') and Google App Engine.
   - Android Virtual Device (Emulator) to run and debug apps in the Android studio.

   Android Studio supports all the same programming languages of IntelliJ (and CLion) e.g. Java, C++, and more with extensions, such as Go and Android Studio 3.0 or later supports Kotlin and "Java 7 language features and a subset of Java 8 language features that vary by platform version."[20] External projects backport some Java 9 features. While IntelliJ that Android Studio is built on supports all released Java versions, and Java 12, it's not clear to what level Android Studio supports Java versions up to Java 12 (the documentation mentions partial Java 8 support). At least some new language features up to Java 12 are usable in Android.

2. Firebase:

   Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014. As of October 2018, the Firebase platform has 18 products, which are used by 1.5 million apps.We will use Firebase Authentication, Database and Storage.

7.2 Implementation

7.2.1 XML Code

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.collegify">
<!--
The ACCESS_COARSE/FINE_LOCATION permissions are not required to use
Google Maps Android API v2, but you must specify either coarse or fine
location permissions for the 'MyLocation' functionality.
-->
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

<application
android:allowBackup="true"
android:icon="@drawable/collegify_logo"
android:label="@string/app_name"
android:roundIcon="@drawable/collegify_logo"
android:supportsRtl="true"
android:theme="@style/AppTheme">
<activity android:name=".GiveReviewActivity"></activity>
<activity android:name=".ReviewActivity" />
<activity android:name=".FeedbackActivity" />
<!--
The API key for Google Maps-based APIs is defined as a string resource.
(See the file "res/values/google_maps_api.xml").
Note that the API key is linked to the encryption key used to sign the APK.
You need a different API key for each encryption key, including the release key that is used to
sign the APK for publishing.
You can define the keys for the debug and release targets in src/debug/ and src/release/.
-->
<meta-data
android:name="com.google.android.geo.API_KEY"
android:value="@string/google_maps_key" />

<activity
android:name=".GoogleMapsActivity"
android:label="@string/title_activity_google_maps" />
<activity android:name=".ChangePasswordActivity" />
<activity android:name=".VerifySQ" />
<activity android:name=".ChangeSecurityQuestionActivity" />
<activity android:name=".SecurityQuestionActivity" />
<activity android:name=".FilterActivity" />
<activity android:name=".CollegeActivity">
<meta-data
android:name="android.support.PARENT_ACTIVITY"
android:value=".CollegeListActivity" />
```

```
</activity>
<activity android:name=".CollegeListActivity" />
<activity android:name=".BookmarksActivity" />
<activity android:name=".SubCategory" />
<activity android:name=".ProfileActivity">
<intent-filter>
<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
</activity>
<activity android:name=".VerifyPhone" />
<activity android:name=".PhoneLogin" />
<activity android:name=".ForgotPasswordActivity">
<meta-data
android:name="android.support.PARENT_ACTIVITY"
android:value=".MainActivity" />
</activity>
<activity android:name=".SignUpActivity" />
<activity android:name=".MainActivity" >
<intent-filter>
<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
</activity>
</application>

</manifest>
```

7.2.2 Layouts

ActivityBookmarks.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
tools:context=".CollegeListActivity">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />

<android.support.v4.widget.DrawerLayout
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/drawer"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_below="@+id/toolbar"
style="@style/Theme.AppCompat.Light.NoActionBar"
tools:context="com.example.drawer_menu.MainActivity">

<android.support.v7.widget.RecyclerView
android:id="@+id/recyclerview"
android:layout_width="match_parent"
android:layout_height="match_parent"></android.support.v7.widget.RecyclerView>

<android.support.design.widget.NavigationView
app:headerLayout="@layout/header"
android:layout_width="wrap_content"
android:layout_height="match_parent"
android:background="@color/white"
app:itemTextColor="@color/black"
app:itemIconTint="@color/black"
app:menu="@menu/drawer_menu"
android:layout_gravity="start"
android:id="@+id/navgview">
```

```
</android.support.design.widget.NavigationView>
</android.support.v4.widget.DrawerLayout>

</LinearLayout>
```

ActivityChangePassword.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".ChangePasswordActivity"
android:id="@+id/chgpasspg">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />


<EditText
android:id="@+id/passwd"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_centerHorizontal="true"
android:layout_marginTop="200dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/pass"
android:drawablePadding="10dp"
android:hint="Enter New Password"
android:inputType="textPassword"
android:paddingLeft="10dp" />

<EditText
android:id="@+id/confpasswd"
android:layout_width="300dp"
android:layout_height="40dp"
```

```
android:layout_centerHorizontal="true"
android:layout_marginTop="10dp"
android:background="@drawable/edit_round"
android:layout_below="@id/passwd"


android:drawableLeft="@drawable/pass"
android:drawablePadding="10dp"
android:hint="Re-Enter New Password"
android:inputType="textPassword"
android:paddingLeft="10dp" />

<Button
android:id="@+id/chgpass"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_below="@id/passwd"
android:layout_alignStart="@+id/passwd"
android:layout_marginTop="100dp"
android:background="@drawable/bg"
android:drawableLeft="@drawable/send"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:text="Change Password"

android:textAllCaps="true"
android:textColor="#ffffff"
android:textStyle="bold" />

</RelativeLayout>
```

ActivityChangeSecurityQuestions.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".ChangePasswordActivity"
android:id="@+id/chgpasspg">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
```

```
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />


<EditText
android:id="@+id/passwd"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_centerHorizontal="true"
android:layout_marginTop="200dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/pass"
android:drawablePadding="10dp"
android:hint="Enter New Password"
android:inputType="textPassword"
android:paddingLeft="10dp" />

<EditText
android:id="@+id/confpasswd"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_centerHorizontal="true"
android:layout_marginTop="10dp"
android:background="@drawable/edit_round"
android:layout_below="@id/passwd"


android:drawableLeft="@drawable/pass"
android:drawablePadding="10dp"
android:hint="Re-Enter New Password"
android:inputType="textPassword"
android:paddingLeft="10dp" />

<Button
android:id="@+id/chgpass"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_below="@id/passwd"
android:layout_alignStart="@+id/passwd"
android:layout_marginTop="100dp"
android:background="@drawable/bg"
android:drawableLeft="@drawable/send"
android:paddingLeft="10dp"
android:paddingRight="30dp"
```

```
android:text="Change Password"

android:textAllCaps="true"
android:textColor="#ffffff"
android:textStyle="bold" />

</RelativeLayout>
```

ActivityCollege.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".CollegeActivity">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />

<android.support.v4.view.ViewPager
android:id="@+id/view_pager"
android:layout_width="match_parent"
android:layout_height="200dp"
android:layout_below="@id/toolbar">

</android.support.v4.view.ViewPager>

<TextView
android:id="@+id/collegename"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignBottom="@+id/view_pager"
android:layout_marginBottom="15dp"
android:textSize="25dp"
android:textColor="#fff"
android:textStyle="bold"
android:fontFamily="@font/serif"
android:layout_marginStart="10dp" />
```

```
<Button
android:id="@+id/call"
android:layout_width="35dp"
android:layout_height="30dp"
android:layout_below="@id/view_pager"
android:layout_toStartOf="@+id/addbk"
android:background="@drawable/circlebutton"
android:drawableLeft="@drawable/ic_phone"
android:paddingLeft="6dp"
android:paddingRight="20dp"
android:layout_marginEnd="5dp"
android:layout_marginTop="10dp"
android:textAllCaps="true"
android:textColor="#000"
android:textStyle="bold" />

<Button
android:id="@+id/website"
android:layout_width="wrap_content"
android:layout_height="30dp"
android:layout_below="@id/view_pager"
android:layout_marginStart="5dp"
android:layout_alignStart="@+id/collegename"
android:background="@drawable/circlebutton"
android:drawableLeft="@drawable/ic_website"
android:paddingLeft="20dp"
android:paddingRight="20dp"
android:layout_marginTop="10dp"
android:text="   Website"
android:textAllCaps="true"
android:textColor="#000"
android:textStyle="bold" />

<Button
android:id="@+id/review"
android:layout_width="35dp"
android:layout_height="30dp"
android:layout_below="@id/view_pager"
android:layout_marginTop="10dp"
android:layout_alignParentRight="true"
android:layout_marginEnd="10dp"
android:background="@drawable/circlebutton"
android:drawableLeft="@drawable/ic_review"
android:paddingLeft="6dp"
android:textAllCaps="true"
android:textColor="#000"
android:textStyle="bold" />
```

```
<Button
android:id="@+id/addbk"
android:layout_width="35dp"
android:layout_height="30dp"
android:layout_below="@id/view_pager"
android:layout_toStartOf="@+id/review"
android:layout_marginEnd="5dp"
android:background="@drawable/circlebutton"
android:drawableLeft="@drawable/ic_bookmark"
android:paddingLeft="6dp"
android:layout_marginTop="10dp"
android:textAllCaps="true"
android:textColor="#000"
android:textStyle="bold" />

<Button
android:id="@+id/dir"
android:layout_width="wrap_content"
android:layout_height="30dp"
android:layout_below="@id/call"
android:layout_alignStart="@+id/website"
android:background="@drawable/circlebutton"
android:drawableLeft="@drawable/ic_directions"
android:paddingLeft="20dp"
android:paddingRight="20dp"
android:layout_marginTop="10dp"
android:text="   Directions"
android:textAllCaps="true"
android:textColor="#000"
android:textStyle="bold" />

<Button
android:id="@+id/bus"
android:layout_width="wrap_content"
android:layout_height="30dp"
android:layout_below="@id/call"
android:layout_toEndOf="@+id/dir"
android:layout_marginEnd="10dp"
android:layout_marginTop="10dp"
android:layout_marginStart="5dp"
android:layout_alignParentRight="true"
android:background="@drawable/circlebutton"
android:drawableLeft="@drawable/ic_busstop"
android:paddingLeft="20dp"
android:paddingRight="20dp"
android:text="   Bus Stops"
android:textAlignment="center"
android:textAllCaps="true"
android:textColor="#000"
```

```
android:textStyle="bold" />

<View
android:id="@+id/view1"
android:layout_width="wrap_content"
android:layout_height="2dp"
android:layout_below="@id/dir"
android:layout_marginTop="8dp"
android:layout_marginBottom="340dp"
android:layout_marginEnd="5dp"
android:layout_marginStart="5dp"
android:background="@color/black" />

<TextView
android:id="@+id/addr"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@+id/dir"
android:text="Address :"
android:layout_marginTop="18dp"
android:textSize="20dp"
android:textColor="#000"
android:textStyle="bold"
android:fontFamily="@font/serif"
android:layout_marginStart="10dp" />

<TextView
android:id="@+id/collegeaddr"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@+id/addr"
android:text=""
android:layout_marginTop="10dp"
android:textSize="15dp"
android:textColor="#000"
android:textStyle="bold"
android:fontFamily="@font/serif"
android:layout_marginStart="10dp" />

<View
android:id="@+id/view2"
android:layout_width="wrap_content"
android:layout_height="2dp"
android:layout_below="@id/collegeaddr"
android:layout_marginTop="8dp"
android:layout_marginBottom="340dp"
android:layout_marginEnd="5dp"
android:layout_marginStart="5dp"
android:background="@color/black" />
```

```
<TextView
android:id="@+id/courses"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@+id/collegeaddr"
android:text="Provided Courses :"
android:layout_marginTop="18dp"
android:textSize="20dp"
android:textColor="#000"
android:textStyle="bold"
android:fontFamily="@font/serif"
android:layout_marginStart="10dp" />

<android.support.v7.widget.RecyclerView
android:id="@+id/recyclerview"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@+id/courses"
android:layout_marginTop="18dp">

</android.support.v7.widget.RecyclerView>


</RelativeLayout>
```

ActivityCollegeInfo.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
tools:context=".CollegeListActivity">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
```

```
android:theme="?attr/actionBarTheme" />

<android.support.v4.widget.DrawerLayout
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/drawer"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_below="@+id/toolbar"
style="@style/Theme.AppCompat.Light.NoActionBar"
tools:context="com.example.drawer_menu.MainActivity">

<android.support.v7.widget.RecyclerView
android:id="@+id/recyclerview"
android:layout_width="match_parent"
android:layout_height="match_parent"></android.support.v7.widget.RecyclerView>

<android.support.design.widget.NavigationView
app:headerLayout="@layout/header"
android:layout_width="wrap_content"
android:layout_height="match_parent"
android:background="@color/white"
app:itemTextColor="@color/black"
app:itemIconTint="@color/black"
app:menu="@menu/drawer_menu"
android:layout_gravity="start"
android:id="@+id/navgview">
</android.support.design.widget.NavigationView>



</android.support.v4.widget.DrawerLayout>


</LinearLayout>
```

ActivityFeedback.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:id="@+id/feedbackpg"
tools:context=".FeedbackActivity">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
```

```
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />

<EditText
android:id="@+id/Feedback"
android:layout_width="300dp"
android:layout_height="80dp"

android:layout_centerHorizontal="true"
android:layout_marginTop="200dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/ic_feedback"
android:drawablePadding="10dp"
android:hint="Enter Feedback"
android:inputType="textMultiLine"
android:paddingLeft="10dp" />

<Button
android:id="@+id/submit"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_below="@id/Feedback"
android:layout_centerHorizontal="true"
android:layout_marginTop="150dp"
android:background="@drawable/bg"
android:drawableLeft="@drawable/send"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:text="Submit"
android:textAllCaps="false" />

</RelativeLayout>
```

ActivityFilter.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
```

```
android:orientation="vertical"
tools:context=".ProfileActivity">

<TextView
android:id="@+id/text"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_centerInParent="true" />

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />


<android.support.v4.widget.DrawerLayout
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/drawer"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_below="@+id/toolbar"
style="@style/Theme.AppCompat.Light.NoActionBar"
tools:context="com.example.drawer_menu.MainActivity">

<android.support.v7.widget.RecyclerView
android:id="@+id/recyclerview"
android:layout_width="match_parent"
android:layout_height="match_parent"
>
</android.support.v7.widget.RecyclerView>

<android.support.design.widget.NavigationView
android:id="@+id/navgview"
app:headerLayout="@layout/header"
android:layout_width="wrap_content"
android:layout_height="match_parent"
android:background="@color/white"
app:itemTextColor="@color/black"
app:itemIconTint="@color/black"
app:menu="@menu/drawer_menu"
android:layout_gravity="start">
```

```
</android.support.design.widget.NavigationView>

</android.support.v4.widget.DrawerLayout>


</RelativeLayout>
```

ActivityForgotPassword.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:id="@+id/forgotpasspg"
tools:context=".ForgotPasswordActivity">

<EditText
android:id="@+id/email"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_alignParentTop="true"
android:layout_centerHorizontal="true"
android:layout_marginTop="189dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/user"
android:drawablePadding="10dp"
android:hint="Enter e-mail address"
android:inputType="textEmailAddress"
android:paddingLeft="10dp" />

<TextView
android:id="@+id/message"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@+id/email"
android:layout_centerHorizontal="true"
android:layout_marginTop="10dp"
style="@style/textviewLinks"
android:text="Enter your registered email to get a \npassword reset link via e-mail" />

<Button
android:id="@+id/forgotpass"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_centerHorizontal="true"
```

```
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:drawableLeft="@drawable/send"
android:layout_below="@id/email"
android:text="OK"
android:textAllCaps="true"
android:textStyle="bold"
android:layout_marginTop="150dp"

android:textColor="#ffffff"
android:background="@drawable/bg"
/>

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />


</RelativeLayout>
```

ActivityGoogleMaps.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
android:layout_height="match_parent"
android:layout_width="match_parent"
xmlns:android="http://schemas.android.com/apk/res/android">

<RelativeLayout
android:layout_width="match_parent"
android:layout_height="wrap_content">

<EditText
android:id="@+id/location_search"
android:layout_width="310dp"
android:layout_height="40dp"
android:layout_marginTop="10dp"
android:layout_centerVertical="true"
android:ems="10"
android:background="@drawable/edit_round"
android:paddingLeft="10dp"
```

```
android:drawableLeft="@drawable/ic_location_searching"
android:drawablePadding="10dp"
android:hint="Write Any Place Name..."/>

<ImageButton
android:id="@+id/search_addr"
android:layout_toEndOf="@+id/location_search"
android:layout_width="wrap_content"
android:layout_height="40dp"
android:layout_centerVertical="true"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:onClick="onClick"
android:background="?android:selectableItemBackground"
android:src="@mipmap/search"/>

</RelativeLayout>

<fragment xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:map="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/map"
android:name="com.google.android.gms.maps.SupportMapFragment"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_marginTop="45dp"
android:layout_alignParentStart="true"
android:layout_alignParentTop="true"
tools:context=".GoogleMapsActivity" />

<RelativeLayout
android:id="@+id/second_relative_layout"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentBottom="true">
<ImageButton
android:id="@+id/nearby_colleges"
android:layout_width="100dp"
android:layout_height="wrap_content"
android:layout_marginBottom="10dp"
android:layout_centerHorizontal="true"
android:onClick="onClick"
android:background="@drawable/round_corner"
android:src="@mipmap/college"/>
</RelativeLayout>

</RelativeLayout>
```

ActivityMain.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity"
android:id="@+id/loginpg">


<TextView
android:id="@+id/title"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_centerHorizontal="true"
android:layout_marginTop="80dp"
android:fontFamily="@font/pacifico"
android:text="Collegify"
android:textColor="@color/black"
android:textSize="80dp" />

<TextView
android:id="@+id/logintitle"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/title"
android:layout_alignStart="@+id/email"
android:fontFamily="@font/sketchy"
android:text="Login"
android:textColor="@color/black"
android:textSize="30sp" />

<EditText
android:id="@+id/email"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_below="@id/title"
android:layout_centerHorizontal="true"
android:layout_marginTop="40dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/user"
android:drawablePadding="10dp"
android:hint="Enter email address"
android:inputType="textEmailAddress"
android:paddingLeft="10dp" />
```

```
<EditText
android:id="@+id/passwd"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_below="@id/email"
android:layout_centerHorizontal="true"
android:layout_marginTop="10dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/pass"
android:drawablePadding="10dp"
android:hint="Enter Password"
android:inputType="textPassword"
android:paddingLeft="10dp" />

<Button
android:id="@+id/submit"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_below="@id/passwd"
android:layout_alignStart="@+id/email"
android:layout_marginTop="82dp"
android:background="@drawable/bg"
android:drawableLeft="@drawable/send"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:text="Submit"

android:textAllCaps="true"
android:textColor="#ffffff"
android:textStyle="bold" />


<TextView
android:id="@+id/signup"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignTop="@+id/submit"
android:layout_centerHorizontal="true"
android:layout_marginTop="62dp"
style="@style/textviewLinks"
android:text="Not registered? Sign Up here" />

<TextView
android:id="@+id/forgotpass"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignTop="@+id/passwd"
```

```
android:layout_centerHorizontal="true"
android:layout_marginTop="50dp"
style="@style/textviewLinks"
android:text="Forgot Password?" />



<TextView
android:id="@+id/phone"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignTop="@+id/forgotpass"
android:layout_centerHorizontal="true"
android:layout_marginStart="130dp"
android:layout_marginTop="30dp"
style="@style/textviewLinks"
android:text="Login via Phone Number" />


</RelativeLayout>
```

ActivityPhoneLogin.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/loginbg"
tools:context=".PhoneLogin">

<TextView
android:id="@+id/title"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_centerHorizontal="true"
android:layout_marginTop="80dp"
android:fontFamily="@font/pacifico"
android:text="Collegify"
android:textColor="@color/black"
android:textSize="80dp" />

<EditText
android:id="@+id/editTextMobile"
android:layout_width="300dp"
android:layout_height="40dp"
```

```
android:layout_below="@id/title"
android:layout_centerHorizontal="true"
android:layout_marginTop="40dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/phone"
android:drawablePadding="10dp"
android:hint="Enter phone number"
android:inputType="textEmailAddress"
android:paddingLeft="10dp" />

<TextView
android:id="@+id/message"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/editTextMobile"
android:layout_centerHorizontal="true"
android:layout_marginStart="72dp"
android:layout_marginTop="30dp"
style="@style/textviewLinks"
android:text="You will receive a verification code soon!!" />

<TextView
android:id="@+id/emailLogin"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@+id/editTextMobile"
android:layout_centerHorizontal="true"
android:layout_marginStart="130dp"
android:layout_marginTop="70dp"
style="@style/textviewLinks"
android:text="Login with E-mail" />

<Button
android:id="@+id/buttonContinue"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_centerHorizontal="true"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:drawableLeft="@drawable/send"
android:layout_below="@id/editTextMobile"
android:text="Continue"
android:textAllCaps="true"
android:textStyle="bold"
android:layout_marginTop="150dp"

android:textColor="#ffffff"
android:background="@drawable/bg" />
```

```
</RelativeLayout>
```

ActivityProfile.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
tools:context=".ProfileActivity">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />


<android.support.v4.widget.DrawerLayout
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/drawer"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_below="@+id/toolbar"
style="@style/Theme.AppCompat.Light.NoActionBar"
tools:context="com.example.drawer_menu.MainActivity">

<android.support.v7.widget.RecyclerView
android:id="@+id/recyclerview"
android:layout_width="match_parent"
android:layout_height="match_parent"
>
</android.support.v7.widget.RecyclerView>

<android.support.design.widget.NavigationView
android:id="@+id/navgview"
app:headerLayout="@layout/header"
android:layout_width="wrap_content"
android:layout_height="match_parent"
```

```
android:background="@color/white"
app:itemTextColor="@color/black"
app:itemIconTint="@color/black"
app:menu="@menu/drawer_menu"
android:layout_gravity="start">
</android.support.design.widget.NavigationView>

</android.support.v4.widget.DrawerLayout>


</RelativeLayout>
```

ActivitySecurityQuestion.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".SecurityQuestionActivity"
android:id="@+id/sqpg">


<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />

<EditText
android:id="@+id/squestion"
android:layout_width="300dp"
android:layout_height="80dp"

android:layout_centerHorizontal="true"
android:layout_marginTop="200dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/ic_questionmark"
android:drawablePadding="10dp"
android:hint="Enter Security Question"
android:inputType="textMultiLine"
android:paddingLeft="10dp" />
```

```
<EditText
android:id="@+id/answer"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_below="@+id/squestion"
android:layout_centerHorizontal="true"
android:layout_marginTop="20dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/ic_answer"
android:drawablePadding="10dp"
android:hint="Enter the answer"
android:inputType="textMultiLine"
android:paddingLeft="10dp" />

<Button
android:id="@+id/submit"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_below="@id/answer"
android:layout_centerHorizontal="true"
android:layout_marginTop="150dp"
android:background="@drawable/bg"
android:drawableLeft="@drawable/send"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:text="Submit"
android:textAllCaps="false" />



</RelativeLayout>
```

ActivitySignUp.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/loginbg">

<TextView
android:id="@+id/title"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_centerHorizontal="true"
android:layout_marginTop="80dp"
```

```
android:fontFamily="@font/pacifico"
android:text="Collegify"
android:textColor="@color/black"
android:textSize="80dp" />

<TextView
android:id="@+id/signtitle"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/title"
android:layout_alignStart="@+id/email"
android:fontFamily="@font/sketchy"
android:text="SignUp"
android:textColor="@color/black"
android:textSize="30sp" />

<EditText
android:id="@+id/name"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_below="@+id/title"
android:layout_centerHorizontal="true"
android:layout_marginTop="40dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/user"
android:drawablePadding="10dp"
android:hint="Enter your Username"
android:inputType="textEmailAddress"
android:paddingLeft="10dp" />

<EditText
android:id="@+id/email"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_below="@+id/name"
android:layout_centerHorizontal="true"
android:layout_marginTop="10dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/user"
android:drawablePadding="10dp"
android:hint="Enter email address"
android:inputType="textEmailAddress"
android:paddingLeft="10dp" />

<EditText
```

```
android:id="@+id/passwd"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_below="@id/email"
android:layout_centerHorizontal="true"
android:layout_marginTop="10dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/pass"
android:drawablePadding="10dp"
android:hint="Enter Password"
android:inputType="textPassword"
android:paddingLeft="10dp" />


<EditText
android:id="@+id/confpasswd"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_below="@id/passwd"
android:layout_centerHorizontal="true"
android:layout_marginTop="10dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/pass"
android:drawablePadding="10dp"
android:hint="Re-enter Password"
android:inputType="textPassword"
android:paddingLeft="10dp" />

<Button
android:id="@+id/submit"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_below="@id/passwd"
android:layout_centerHorizontal="true"
android:layout_marginTop="150dp"
android:background="@drawable/bg"
android:drawableLeft="@drawable/send"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:text="Submit"
android:textAllCaps="false" />

<TextView
android:id="@+id/log"
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
android:layout_alignTop="@+id/confpasswd"
android:layout_centerHorizontal="true"
android:layout_marginTop="50dp"
android:paddingStart="60dp"
android:paddingEnd="50dp"
style="@style/textviewLinks"
android:text="Already Registered? Login" />

</RelativeLayout>
```

ActivitySubCategory.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
tools:context=".SubCategory">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />

<android.support.v4.widget.DrawerLayout
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/drawer"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_below="@+id/toolbar"
style="@style/Theme.AppCompat.Light.NoActionBar"
tools:context="com.example.drawer_menu.MainActivity">

<android.support.v7.widget.RecyclerView
android:id="@+id/recyclerview"
android:layout_width="match_parent"
android:layout_height="match_parent"></android.support.v7.widget.RecyclerView>

<android.support.design.widget.NavigationView
```

```
app:headerLayout="@layout/header"
android:layout_width="wrap_content"
android:layout_height="match_parent"
android:background="@color/white"
app:itemTextColor="@color/black"
app:itemIconTint="@color/black"
app:menu="@menu/drawer_menu"
android:layout_gravity="start"
android:id="@+id/navgview">
</android.support.design.widget.NavigationView>



</android.support.v4.widget.DrawerLayout>


</LinearLayout>
```

activity_verify_phone.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/loginbg"
tools:context=".VerifyPhone">


<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_centerHorizontal="true"
android:layout_marginTop="129dp"
android:text="You will receive a code shortly"
android:textSize="25dp"
android:textColor="@color/black"
android:textStyle="bold"
/>

<ProgressBar
android:visibility="gone"
android:id="@+id/progressbar"
android:layout_below="@id/editTextCode"
android:layout_centerHorizontal="true"
android:layout_marginTop="20dp"
android:layout_width="wrap_content"
android:layout_height="wrap_content" />
```

```
<EditText
android:id="@+id/editTextCode"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_alignParentTop="true"
android:layout_centerHorizontal="true"
android:layout_marginTop="189dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/user"
android:drawablePadding="10dp"
android:hint="Enter verification code"
android:inputType="textEmailAddress"
android:paddingLeft="10dp" />

<Button
android:id="@+id/buttonSignIn"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_centerHorizontal="true"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:drawableLeft="@drawable/send"
android:layout_below="@id/editTextCode"
android:text="Login"
android:textAllCaps="true"
android:textStyle="bold"
android:layout_marginTop="150dp"

android:textColor="#ffffff"
android:background="@drawable/bg" />
</RelativeLayout>

activity_verify_sq.xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:id="@+id/verifySQpg"
tools:context=".VerifySQ">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
```

```
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />

<TextView
android:id="@+id/asksq"
android:layout_width="300dp"
android:layout_height="wrap_content"
android:layout_centerHorizontal="true"
android:layout_marginTop="200dp"
android:text=""
android:textStyle="bold"
android:fontFamily="@font/serif"
android:textAlignment="center"
android:textSize="18sp" />

<EditText
android:id="@+id/answer"
android:layout_width="300dp"
android:layout_height="40dp"

android:layout_below="@+id/asksq"
android:layout_centerHorizontal="true"
android:layout_marginTop="20dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/ic_answer"
android:drawablePadding="10dp"
android:hint="Enter the answer"
android:inputType="textMultiLine"
android:paddingLeft="10dp" />

<Button
android:id="@+id/submit"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_below="@id/answer"
android:layout_centerHorizontal="true"
android:layout_marginTop="150dp"
android:background="@drawable/bg"
android:drawableLeft="@drawable/send"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:text="Submit"
android:textAllCaps="false" />
```

```
</RelativeLayout>
```

CardviewItem.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent"
android:layout_height="100dp"
android:layout_margin="5dp"
android:id="@+id/cardview"
android:layout_centerInParent="true"

app:cardCornerRadius="20dp"
android:innerRadius="0dp"
android:shape="ring"
android:thicknessRatio="1.9">
<!-- Last four lines to make cardview round corners-->

<RelativeLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical">


<ImageView
android:id="@+id/categorybg"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:scaleType="fitXY"
>
</ImageView>
<ImageView
android:id="@+id/categorybg2"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/cardview_gradient"
android:scaleType="fitXY">
</ImageView>

<TextView
android:id="@+id/category"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_centerInParent="true"
android:fontFamily="@font/serif"
android:text="College Category"
android:textAlignment="center"
android:textColor="@color/black"
```

```
android:textSize="20dp"
android:textStyle="bold" />


</RelativeLayout>



</android.support.v7.widget.CardView>
```

CategoryItem.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent"
android:layout_height="300dp"
android:layout_margin="5dp"
android:id="@+id/cardview"
android:layout_gravity="center"

app:cardCornerRadius="20dp"
android:innerRadius="0dp"
android:shape="ring"
android:thicknessRatio="1.9">
<!-- Last four lines to make cardview round corners-->

<RelativeLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical">


<ImageView
android:id="@+id/categorybg"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:scaleType="fitXY"
android:background="#FFF"
>
</ImageView>
<ImageView
android:id="@+id/categorybg2"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/cardview_gradient"
android:scaleType="fitXY">
</ImageView>

<TextView
```

```
android:id="@+id/category"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_centerInParent="true"
android:fontFamily="@font/serif"
android:text="College Category"
android:textAlignment="center"
android:textAllCaps="true"
android:textColor="@color/black"
android:textSize="20dp"
android:textStyle="bold" />


</RelativeLayout>

</android.support.v7.widget.CardView>
```

CourseItem.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:orientation="horizontal"
android:layout_width="match_parent"
android:layout_height="80dp">

<TextView
android:layout_width="match_parent"
android:layout_height="match_parent"
android:text="Course Name"
android:textSize="20dp"
android:textStyle="bold"
android:gravity="center"
android:layout_marginLeft="10dp"
android:fontFamily="@font/serif"
android:id="@+id/cname"
/>


</LinearLayout>
```

header.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:orientation="vertical"
android:layout_width="match_parent"
android:layout_height="160dp"
android:background="@drawable/headerbg"
android:padding="20dp">
<ImageView
```

```
android:src="@drawable/profilepic"
android:layout_height="75dp"
android:layout_width="75dp"
/>
<TextView
android:id="@+id/user"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="10dp"
android:textStyle="bold"
android:textColor="@color/black"
android:text="user"
android:textSize="25dp"/>
</LinearLayout>
```

Swipe.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:orientation="vertical"
android:layout_width="match_parent"
android:layout_height="match_parent">


<ImageView
android:id="@+id/image_view"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_alignParentStart="true"
android:layout_alignParentTop="true"
android:layout_marginStart="0dp"
android:layout_marginTop="0dp"
android:scaleType="fitXY" />
</RelativeLayout>
```

DrawerMenu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto">
<item android:id="@+id/Home"
android:title="Home"
android:icon="@drawable/ic_home"/>
<item
android:id="@+id/Bookmarks"
android:title="Bookmarks"
android:icon="@drawable/ic_round_bookmarks_24px"/>
<item
android:id="@+id/gmaps"
```

```
android:title="Nearby Places"
android:icon="@drawable/ic_visited"/>
<item
android:id="@+id/ChangePassword"
android:title="Change Password"
android:icon="@drawable/ic_chgpass"/>
<item
android:id="@+id/ChangeSecurityQuestions"
android:title="Change Security Questions"
android:icon="@drawable/ic_securityquestion"/>
<item
android:id="@+id/HelpSupport"
android:title="Help &amp; Support"
android:icon="@drawable/ic_help_support"/>
<item
android:id="@+id/Feedback"
android:title="Feedback"
android:icon="@drawable/ic_feedback"/>
<item
android:id="@+id/Logout"
android:title="Logout"
android:icon="@drawable/ic_logout"/>

</menu>
```

SearchMenu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto">

<item
android:id="@+id/search"
android:title="Search"
android:orderInCategory="100"
android:icon="@drawable/ic_search"
app:actionViewClass="android.widget.SearchView"
app:showAsAction="always"/>

</menu>
```

ActivityReview.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".ReviewActivity">
```

```
<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />



<android.support.v7.widget.RecyclerView
android:id="@+id/recyclerview"
android:layout_below="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="match_parent">
</android.support.v7.widget.RecyclerView>


<Button
android:id="@+id/grev"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_alignParentBottom="true"
android:layout_centerHorizontal="true"
android:layout_marginBottom="37dp"
android:background="@drawable/bg"

android:drawableLeft="@drawable/send"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:text="Give Review"
android:textAllCaps="true"
android:textColor="#ffffff"
android:textStyle="bold" />


</RelativeLayout>
```

ActivityGiveReview.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
```

```
android:layout_width="match_parent"
android:layout_height="match_parent"
android:id="@+id/reviewpg"
tools:context=".ReviewActivity">

<android.support.v7.widget.Toolbar
android:id="@+id/toolbar"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_alignParentEnd="true"
android:layout_marginTop="0dp"
android:layout_marginEnd="0dp"
android:background="@drawable/gradient"
android:minHeight="?attr/actionBarSize"
android:theme="?attr/actionBarTheme" />

<EditText
android:id="@+id/grev"
android:layout_width="300dp"
android:layout_height="80dp"

android:layout_centerHorizontal="true"
android:layout_marginTop="200dp"
android:background="@drawable/edit_round"

android:drawableLeft="@drawable/ic_feedback"
android:drawablePadding="10dp"
android:hint="Enter Review"
android:inputType="textMultiLine"
android:paddingLeft="10dp" />

<Button
android:id="@+id/submit"
android:layout_width="300dp"
android:layout_height="40dp"
android:layout_below="@id/grev"
android:layout_centerHorizontal="true"
android:layout_marginTop="150dp"
android:background="@drawable/bg"
android:drawableLeft="@drawable/send"
android:paddingLeft="10dp"
android:paddingRight="30dp"
android:text="Submit"
android:textAllCaps="false" />

</RelativeLayout>
```

ReviewItem.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android" android:layout_width="match_parent"
android:layout_height="match_parent">

<TextView
android:id="@+id/userid"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentStart="true"
android:layout_alignParentTop="true"
android:layout_marginStart="10dp"
android:layout_marginTop="10dp"
android:text="user"
android:textColor="#000"
android:textStyle="bold"
android:textSize="20dp"/>

<TextView
android:id="@+id/review"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@+id/userid"
android:layout_marginStart="30dp"
android:layout_marginTop="10dp"
android:text="review"
android:textColor="#000"/>
</RelativeLayout>
```

7.2.3 Java Codes

BookmarksActivity

```
package com.example.collegify;

import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.GridLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;

import com.example.collegify.Adapters.CollegeListAdapter;
import com.example.collegify.Lists.CollegeList;
import com.example.collegify.Lists.common;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

public class BookmarksActivity extends AppCompatActivity {

private Toolbar toolbar;
private DrawerLayout mDrawerlayout;
private ActionBarDrawerToggle mToggle;
private TextView user;

private NavigationView navigationView;
List<CollegeList> collegeList;


private RecyclerView recyclerView;
private RecyclerView.LayoutManager layoutManager;
```

```
private CollegeListAdapter SubAdapter;

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_bookmarks);

navigationView = (NavigationView) findViewById(R.id.navgview);
View headerView = navigationView.getHeaderView(0);
user = (TextView) headerView.findViewById(R.id.user);
user.setText(common.username);

toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Bookmarks");
mDrawerlayout = (DrawerLayout) findViewById(R.id.drawer);
mToggle = new ActionBarDrawerToggle(this, mDrawerlayout,
R.string.open, R.string.close);
mDrawerlayout.addDrawerListener(mToggle);
mToggle.syncState();
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

collegeList = new ArrayList<>();
recyclerView = (RecyclerView) findViewById(R.id.recyclerview);
layoutManager = new GridLayoutManager(this, 1);
recyclerView.setLayoutManager(layoutManager);
recyclerView.setHasFixedSize(true);

DatabaseReference mDb = FirebaseDatabase.getInstance().
getReference("Users/"+common.username+"/
Bookmarks");
mDb.addValueEventListener(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot)
 {

for (DataSnapshot postSnapshot: dataSnapshot.getChildren())
{


String name = postSnapshot.getValue(String.class);
collegeList.add(new CollegeList(name));

}
SubAdapter = new CollegeListAdapter(BookmarksActivity.this,
collegeList);
recyclerView.setAdapter(SubAdapter);
}
```

```
@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e,
Toast.LENGTH_SHORT).show();
}
});

navigationView = (NavigationView) findViewById(R.id.navgview);

navigationView.setNavigationItemSelectedListener(new NavigationView.OnNavigationItemSelectedListener()
{
@Override
public boolean onNavigationItemSelected(@NonNull MenuItem menuItem)
 {
switch (menuItem.getItemId()) {
case R.id.Home:
Intent HomeIntent = new Intent(getApplicationContext(),
ProfileActivity.class);
startActivity(HomeIntent);
finish();
break;
case R.id.Bookmarks:
Intent BookmarkIntent = new Intent(getApplicationContext(), BookmarksActivity.class);
startActivity(BookmarkIntent);
finish();
break;
case R.id.gmaps:
Intent GoogleMapsIntent = new Intent(getApplicationContext(), GoogleMapsActivity.class);
startActivity(GoogleMapsIntent);
break;
case R.id.ChangePassword:
Intent ChgPassIntent = new Intent(getApplicationContext(), ChangePasswordActivity.class);
startActivity(ChgPassIntent);
break;
case R.id.ChangeSecurityQuestions:
Intent ChgSecQuest = new Intent(getApplicationContext(),
ChangeSecurityQuestionActivity.class);
startActivity(ChgSecQuest);
break;
case R.id.Feedback:
Intent FeedbackIntent = new Intent(getApplicationContext(),FeedbackActivity.class);
startActivity(FeedbackIntent);
break;
case R.id.Logout:
FirebaseAuth.getInstance().signOut();
finish();
startActivity(new Intent(getApplicationContext(),
MainActivity.class));
```

```
}
return false;
}
});

}

// So that the 3 line in toolbar work
@Override
public boolean onOptionsItemSelected(MenuItem item) {
if (mToggle.onOptionsItemSelected(item)) {
return true;
}
return super.onOptionsItemSelected(item);

}
}
```

ChangePasswordActivity

```
package com.example.collegify;

import android.annotation.SuppressLint;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.drawable.BitmapDrawable;
import android.renderscript.Allocation;
import android.renderscript.Element;
import android.renderscript.RenderScript;
import android.renderscript.ScriptIntrinsicBlur;
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
```

```
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;

public class ChangePasswordActivity extends AppCompatActivity
 {

private EditText passwd;
private EditText confpasswd;
private Button chgpass;
private ProgressDialog progressDialog;
private FirebaseAuth firebaseAuth;

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_change_password);

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Change Password");
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

//To Blur the Background image
Bitmap bitmap = BitmapFactory.decodeResource(this.getResources(),R.drawable.loginbg);
//Drawable to bitmap
Bitmap blurredBitmap = blurRenderScript(this,bitmap, 10);//second parametre is radius
findViewById(R.id.chgpasspg).setBackground(new BitmapDrawable(getResources(), blurredBitmap));
//Assign the bitmap as background

passwd = (EditText) findViewById(R.id.passwd);
confpasswd = (EditText) findViewById(R.id.confpasswd);
chgpass = (Button) findViewById(R.id.chgpass);

progressDialog = new ProgressDialog(this);
firebaseAuth = FirebaseAuth.getInstance();

chgpass.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
changePassword();
}
});
}

private void changePassword() {

String pass = passwd.getText().toString().trim();
String confpass = confpasswd.getText().toString().trim();
FirebaseUser user = FirebaseAuth.getInstance().getCurrentUser();
```

```
if (TextUtils.isEmpty(pass)) {
//password is empty
//Toast.makeText(this, "Please enter password", Toast.LENGTH_SHORT).show();
passwd.setError("Please enter password");
passwd.requestFocus();
//stopping the function execution further
return;
}
if (TextUtils.isEmpty(confpass)) {
//password is empty
//Toast.makeText(this, "Please enter password", Toast.LENGTH_SHORT).show();
confpasswd.setError("Please re-enter password");
confpasswd.requestFocus();
//stopping the function execution further
return;
}
if(confpass.equals(pass)){
if(user!=null){
progressDialog.setMessage("Changing Password...");
progressDialog.show();

user.updatePassword(pass).addOnCompleteListener(new OnCompleteListener<Void>()
{
@Override
public void onComplete(@NonNull Task<Void> task) {
progressDialog.dismiss();
if(task.isSuccessful()){
Toast.makeText(ChangePasswordActivity.this, "Password Changed Successfully", Toast.LENGTH_LONG).show();
finish();
}
else {
Toast.makeText(ChangePasswordActivity.this,
"Unknown error occured,Password Change Unsuccessful" ,Toast.LENGTH_LONG).show();
}
}
});
}

}
else {
Toast.makeText(this, "The re-entered password doesn't match", Toast.LENGTH_SHORT).show();
return;
}

}
@Override
public boolean onOptionsItemSelected(MenuItem item) {
switch (item.getItemId()) {
```

```
// Respond to the action bar's Up/Home button
case android.R.id.home:
//NavUtils.navigateUpFromSameTask(this);
finish();
return true;
}
return super.onOptionsItemSelected(item);
}
// For blurring Images
@SuppressLint("NewApi")
public static Bitmap blurRenderScript(Context context, Bitmap smallBitmap,
int radius)
 {
try {
smallBitmap = RGB565toARGB888(smallBitmap);
} catch (Exception e) {
e.printStackTrace();
}

Bitmap bitmap = Bitmap.createBitmap(
smallBitmap.getWidth(), smallBitmap.getHeight(),
Bitmap.Config.ARGB_8888);

RenderScript renderScript = RenderScript.create(context);

Allocation blurInput = Allocation.createFromBitmap(renderScript, smallBitmap);
Allocation blurOutput = Allocation.createFromBitmap(renderScript, bitmap);

ScriptIntrinsicBlur blur = ScriptIntrinsicBlur.create(renderScript,
Element.U8_4(renderScript));
blur.setInput(blurInput);
blur.setRadius(radius); // radius must be 0 < r <= 25
blur.forEach(blurOutput);

blurOutput.copyTo(bitmap);
renderScript.destroy();

return bitmap;

}

private static Bitmap RGB565toARGB888(Bitmap img) throws Exception
{
int numPixels = img.getWidth() * img.getHeight();
int[] pixels = new int[numPixels];

//Get JPEG pixels.  Each int is the color values for one pixel.
img.getPixels(pixels, 0, img.getWidth(), 0, 0, img.getWidth(),
img.getHeight());
```

```
//Create a Bitmap of the appropriate format.
Bitmap result = Bitmap.createBitmap(img.getWidth(), img.getHeight(), Bitmap.Config.ARGB_8888);

//Set RGB pixels.
result.setPixels(pixels, 0, result.getWidth(), 0, 0, result.getWidth(),
result.getHeight());
return result;
}
}
```

ChangeSecurityQuestionActivity

```
package com.example.collegify;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.drawable.BitmapDrawable;
import android.renderscript.Allocation;
import android.renderscript.Element;
import android.renderscript.RenderScript;
import android.renderscript.ScriptIntrinsicBlur;
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.example.collegify.Lists.common;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
```

```
public class ChangeSecurityQuestionActivity extends AppCompatActivity {

private TextView user;



private String email;
private TextView security_question;
private EditText ans;
private EditText newans;
private EditText question;
private Button submit;

final DatabaseReference mDatabase = FirebaseDatabase.getInstance().
getReference("Users");
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_change_security_question);

//To Blur the Background image
Bitmap bitmap = BitmapFactory.decodeResource(this.getResources(),R.drawable.loginbg);
//Drawable to bitmap
Bitmap blurredBitmap = blurRenderScript(this,bitmap, 10);
//second parametre is radius
findViewById(R.id.ChangeSecurityQuestions).
setBackground(new BitmapDrawable(getResources(), blurredBitmap));
//Assign the bitmap a


Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Change Security Question");

getSupportActionBar().setDisplayHomeAsUpEnabled(true);

security_question = (TextView)findViewById(R.id.quest);
ans = (EditText) findViewById(R.id.oldanswer);
question = (EditText) findViewById(R.id.squestion);
newans = (EditText) findViewById(R.id.answer);
email = common.e_mail;

mDatabase.addListenerForSingleValueEvent(new ValueEventListener()
{
@Override
public void onDataChange(@NonNull DataSnapshot
 dataSnapshot)
  {
for(DataSnapshot postSnapshot: dataSnapshot
```

```
.getChildren())
{
if(email.equals(postSnapshot.child("email").
getValue(String.class)))
{
security_question.setText(postSnapshot.child
("question").getValue(String.class));
submit = (Button) findViewById(R.id.submit);
submit.setOnClickListener
(new View.OnClickListener()
{
@Override
public void onClick(View v) {
changeSQ();
}
});
break;
}else {
Toast.makeText
(ChangeSecurityQuestionActivity.this,
"User doesn't exist" ,Toast.LENGTH_LONG ).show();

}
}

}

@Override
public void onCancelled(@NonNull DatabaseError
databaseError)
{
Toast.makeText(getApplicationContext(),
databaseError.getMessage() ,Toast.LENGTH_LONG ).show();
}
});

}
private void changeSQ(){
final String answer = ans.getText().toString();
if (TextUtils.isEmpty(answer)) {
//answer is empty
ans.setError("Please enter the answer");
ans.requestFocus();
//stopping the function execution further
return;
}
mDatabase.addListenerForSingleValueEvent(new ValueEventListener()
{
@Override
```

```
public void onDataChange(@NonNull DataSnapshot dataSnapshot)
 {
for(DataSnapshot postSnapshot: dataSnapshot.getChildren())
{
if(email.equals(postSnapshot.child("email").
getValue(String.class))){
if(answer.equals(postSnapshot.child("answer").
getValue(String.class))){
final String q
= question.getText().toString().trim();
final String a
= newans.getText().toString().trim();
mDatabase.child(common.username)
.child("question").setValue(q);
mDatabase.child(common.username).
child("answer").setValue(a);
Toast.makeText(ChangeSecurityQuestionActivity.this,
"Security Question Changed Successfully!!!" ,
Toast.LENGTH_LONG ).show();

return;
}else{
Toast.makeText(ChangeSecurityQuestionActivity.this,"
The Answer for the Old Security Question may be incorrect" ,Toast.LENGTH_LONG ).show();
}
break;
}else {
Toast.makeText(ChangeSecurityQuestionActivity.this,
"User doesn't exist" ,Toast.LENGTH_LONG ).show();

}
}

}

@Override
public void onCancelled(@NonNull DatabaseError databaseError)
{
Toast.makeText(getApplicationContext(),databaseError.getMessage() ,Toast.LENGTH_LONG ).show();
}
});
}
@Override
public boolean onOptionsItemSelected(MenuItem item) {
switch (item.getItemId()) {
// Respond to the action bar's Up/Home button
case android.R.id.home:
//NavUtils.navigateUpFromSameTask(this);
finish();
```

```
return true;
}
return super.onOptionsItemSelected(item);
}
// For blurring Images
@SuppressLint("NewApi")
public static Bitmap blurRenderScript(Context context, Bitmap smallBitmap,
 int radius) {
try {
smallBitmap = RGB565toARGB888(smallBitmap);
} catch (Exception e) {
e.printStackTrace();
}

Bitmap bitmap = Bitmap.createBitmap(
smallBitmap.getWidth(), smallBitmap.getHeight(),
Bitmap.Config.ARGB_8888);

RenderScript renderScript = RenderScript.create(context);

Allocation blurInput = Allocation.createFromBitmap(renderScript,
smallBitmap);
Allocation blurOutput = Allocation.createFromBitmap(renderScript, bitmap);

ScriptIntrinsicBlur blur = ScriptIntrinsicBlur.create(renderScript,
Element.U8_4(renderScript));
blur.setInput(blurInput);
blur.setRadius(radius); // radius must be 0 < r <= 25
blur.forEach(blurOutput);

blurOutput.copyTo(bitmap);
renderScript.destroy();

return bitmap;

}

private static Bitmap RGB565toARGB888(Bitmap img) throws Exception {
int numPixels = img.getWidth() * img.getHeight();
int[] pixels = new int[numPixels];

//Get JPEG pixels.  Each int is the color values for one pixel.
img.getPixels(pixels, 0, img.getWidth(), 0, 0, img.getWidth(),
 img.getHeight());

//Create a Bitmap of the appropriate format.
Bitmap result = Bitmap.createBitmap(img.getWidth(), img.getHeight(), Bitmap.Config.ARGB_8888);

//Set RGB pixels.
```

```
result.setPixels(pixels, 0, result.getWidth(), 0, 0, result.getWidth(),
result.getHeight());
return result;
}
}
```

CollegeActivity

```
package com.example.collegify;

/*
* Created by Hitanshu on 15-03-2019.
*/

import android.content.Intent;
import android.graphics.Color;
import android.net.Uri;
import android.os.Handler;
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.view.ViewPager;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import com.example.collegify.Adapters.CollegeListAdapter;
import com.example.collegify.Adapters.CourseRVAdapter;
import com.example.collegify.Adapters.CustomSwipeAdapter;
import com.example.collegify.Lists.CollegeInfoList;
import com.example.collegify.Lists.CollegeList;
import com.example.collegify.Lists.common;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.List;
import java.util.Timer;
import java.util.TimerTask;
```

```
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class CollegeActivity extends AppCompatActivity {

private Toolbar toolbar;
List<CollegeInfoList> collegeInfo;
private ApiInterface apiInterface;

private TextView collegenm;
private TextView collegeaddr;

private String clgname;
private String clgname1;
private String clgaddr;
private String phoneno;
private String link;
private double lat;
private double lg;
private List<String> courses;
private List<String> images;

ViewPager viewPager;
CustomSwipeAdapter adapter;
private Timer timer;
private int current_position = 0;

private RecyclerView recyclerView;
private RecyclerView.LayoutManager layoutManager;
private CourseRVAdapter cAdapter;


@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_college);

toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);

getSupportActionBar().setDisplayHomeAsUpEnabled(true);

collegenm = (TextView) findViewById(R.id.collegename);
collegeaddr = (TextView) findViewById(R.id.collegeaddr);
viewPager = (ViewPager) findViewById(R.id.view_pager);

collegeInfo = new ArrayList<>();
```

```
recyclerView = (RecyclerView) findViewById(R.id.recyclerview);
layoutManager = new LinearLayoutManager(this);
recyclerView.setLayoutManager(layoutManager);
recyclerView.setHasFixedSize(true);

final String clgnm = getIntent().getStringExtra("collegenm");
getSupportActionBar().setTitle(clgnm);

////////////////////////////////////////////////////////////////////////////////////////
final DatabaseReference mDatabase = FirebaseDatabase.getInstance().
getReference("Colleges");
mDatabase.addValueEventListener(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
for (DataSnapshot postSnapshot : dataSnapshot.getChildren()) {
String name = postSnapshot.child("name").getValue(String.class);
if (name.toLowerCase().equals(clgnm.toLowerCase())) {
clgname = postSnapshot.child("name").getValue(String.class);
clgname1 = postSnapshot.child("name1").getValue(String.class);
clgaddr = postSnapshot.child("addr").getValue(String.class);
phoneno = postSnapshot.child("contact").getValue(String.class);
link = postSnapshot.child("link").getValue(String.class);

collegenm.setText(clgname);
collegeaddr.setText(clgaddr);
lat = postSnapshot.child("lat").getValue(double.class);
lg = postSnapshot.child("long").getValue(double.class);

courses = new ArrayList<>();
images = new ArrayList<>();
for (DataSnapshot post : postSnapshot.child("course").getChildren()) {
courses.add(post.getValue(String.class));
}

cAdapter = new CourseRVAdapter(courses);
recyclerView.setAdapter(cAdapter);

for (DataSnapshot post : postSnapshot.child("image").getChildren())
{
images.add(post.getValue(String.class));
Toast.makeText(getApplicationContext(), post.getValue(String.class),
Toast.LENGTH_SHORT).show();
}
if (images.isEmpty()) {
collegenm.setTextColor(Color.parseColor("#000000"));
}
adapter = new CustomSwipeAdapter(getApplicationContext(), images);

viewPager.setAdapter(adapter);
```

```
if (images.size() != 0) {
createSlideshow();
}


}
}
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
}
});

final DatabaseReference db = FirebaseDatabase.getInstance().
getReference("Users/" + common.username + "/Bookmarks");
db.addListenerForSingleValueEvent(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

if (dataSnapshot.child(clgname1).exists()) {
db.child(clgname1).setValue(null);
Toast.makeText(getApplicationContext(), "Bookmark Removed",
Toast.LENGTH_SHORT).show();
} else {
//Add the bookmark
db.child("Sample").setValue(null);
db.child(clgname1).setValue(clgname);
Toast.makeText(getApplicationContext(), "Bookmark Added",
Toast.LENGTH_SHORT).show();
}
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
}
});
}
});

Button btn = (Button) findViewById(R.id.call);
Button dirbtn = (Button) findViewById(R.id.dir);
Button busbtn = (Button) findViewById(R.id.bus);
Button webBtn = (Button) findViewById(R.id.website);
Button revBtn = (Button) findViewById(R.id.review);
```

```
btn.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent = new Intent(Intent.ACTION_DIAL);
intent.setData(Uri.parse("tel:" + phoneno));
startActivity(intent);
}
});
dirbtn.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent = new Intent(Intent.ACTION_VIEW);
//it broadcasts the request to all apps asking can anyone provide the service
intent.setData(Uri.parse("geo:" + lat + "," + lg + "?q=" + clgname));
startActivity(intent);
}
});
busbtn.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent = new Intent(Intent.ACTION_VIEW);
//it broadcasts the request to all apps asking can anyone provide the service
intent.setData(Uri.parse("geo:" + lat + "," + lg + "?q=Bus Stops"));
startActivity(intent);
}
});
webBtn.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
if (!link.isEmpty()) {
Uri webaddress = Uri.parse(link);
Intent launchWeb = new Intent(Intent.ACTION_VIEW, webaddress);
//it broadcasts the request to all apps asking can anyone provide the service
if (launchWeb.resolveActivity(getPackageManager()) != null) {
startActivity(launchWeb);
}
} else {
Toast.makeText(getApplicationContext(), "Website Not Available",
Toast.LENGTH_SHORT).show();
}
}
});
revBtn.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent = new Intent(getApplicationContext(),ReviewActivity.class);
intent.putExtra("clgnm", clgname);
startActivity(intent);
```

```
}
});
}

private void createSlideshow() {


final Handler handler = new Handler();
final Runnable runnable = new Runnable() {
@Override
public void run() {
if (current_position == images.size()) {
current_position = 0;
}
viewPager.setCurrentItem(current_position++, true);

}
};
timer = new Timer();
timer.schedule(new TimerTask() {
@Override
public void run() {
handler.post(runnable);
}
}, 400, 2500);
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
switch (item.getItemId()) {
// Respond to the action bar's Up/Home button
case android.R.id.home:
//NavUtils.navigateUpFromSameTask(this);
finish();
return true;
}
return super.onOptionsItemSelected(item);
}
}
```

CollegeListActivity

```
package com.example.collegify;

/*
* Created by Hitanshu on 12-03-2019.
*/

import android.content.Intent;
```

```
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.GridLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.SearchView;
import android.widget.Toast;

import com.example.collegify.Adapters.CollegeListAdapter;
import com.example.collegify.Lists.CollegeList;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.List;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

public class CollegeListActivity extends AppCompatActivity {

private Toolbar toolbar;
private DrawerLayout mDrawerlayout;
private ActionBarDrawerToggle mToggle;
List<CollegeList> collegeList;
private ApiInterface apiInterface;

private RecyclerView recyclerView;
private RecyclerView.LayoutManager layoutManager;

private CollegeListAdapter SubAdapter;

private NavigationView navigationView;

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

```
setContentView(R.layout.activity_college_info);

toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Colleges");
mDrawerlayout=(DrawerLayout) findViewById(R.id.drawer);
mToggle = new ActionBarDrawerToggle(this, mDrawerlayout,
 R.string.open, R.string.close);
mDrawerlayout.addDrawerListener(mToggle);
mToggle.syncState();
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

collegeList = new ArrayList<>();
recyclerView = (RecyclerView)
findViewById(R.id.recyclerview);
layoutManager = new GridLayoutManager(this, 1);
recyclerView.setLayoutManager(layoutManager);
recyclerView.setHasFixedSize(true);

String subcat = getIntent().
getStringExtra("subcategory");
Toast.makeText(getApplicationContext(), subcat, Toast.LENGTH_LONG).show();

apiInterface = ApiClient.getApiClient().
create(ApiInterface.class);
Call<List<CollegeList>> call = apiInterface.
sendCourse(subcat);

call.enqueue(new Callback<List<CollegeList>>() {
@Override
public void onResponse(Call<List<CollegeList>> call,
Response<List<CollegeList>> response) {
collegeList = response.body();
SubAdapter = new CollegeListAdapter
(CollegeListActivity.this,
collegeList);
recyclerView.setAdapter(SubAdapter);
}

@Override
public void onFailure(Call<List<CollegeList>>
call, Throwable t)
{
String error = t.getMessage();
Toast.makeText(CollegeListActivity.this,"Error:
"+error,Toast.LENGTH_SHORT).show();
}
});
```

```
navigationView = (NavigationView) findViewById(R.id.navgview);

navigationView.setNavigationItemSelectedListener(new NavigationView.OnNavigationItemSelectedListener()
{
@Override
public boolean onNavigationItemSelected
(@NonNull MenuItem menuItem)
{
switch (menuItem.getItemId()) {
case R.id.Home:
Intent HomeIntent =
new Intent(getApplicationContext(),
 ProfileActivity.class);
startActivity(HomeIntent);
finish();
break;
case R.id.Bookmarks:
Intent BookmarkIntent =
 new Intent(getApplicationContext(), BookmarksActivity.class);
startActivity(BookmarkIntent);
finish();
break;
case R.id.gmaps:
Intent GoogleMapsIntent = new Intent(getApplicationContext(),
GoogleMapsActivity.class);
startActivity(GoogleMapsIntent);
break;
case R.id.ChangePassword:
Intent ChgPassIntent = new Intent
(getApplicationContext(),
ChangePasswordActivity.class);
startActivity(ChgPassIntent);
break;
case R.id.ChangeSecurityQuestions:
Intent ChgSecQuest =
new Intent(getApplicationContext(),
ChangeSecurityQuestionActivity.class);
startActivity(ChgSecQuest);
break;
case R.id.Feedback:
Intent FeedbackIntent =
new Intent(getApplicationContext(),
FeedbackActivity.class);
startActivity(FeedbackIntent);
break;
case R.id.Logout:
FirebaseAuth.getInstance().signOut();
finish();
```

```
startActivity
(new Intent(getApplicationContext(),
MainActivity.class));
}
return false;
}
});
}
@Override
public boolean onCreateOptionsMenu(Menu menu){
final MenuInflater inflater = getMenuInflater();
inflater.inflate(R.menu.search_menu, menu);
MenuItem item = menu.findItem(R.id.search);
SearchView searchView = (SearchView) item.getActionView();

searchView.setOnQueryTextListener
(new SearchView.OnQueryTextListener()
 {
@Override
public boolean onQueryTextSubmit(String query)
 {
String user_input = query.trim();
Intent intent =
 new Intent(getApplicationContext(),
FilterActivity.class);
/*Bundle bundle = new Bundle();
bundle.putParcelableArrayList
("AllColleges",allCollegeList );
bundle.putParcelableArrayList
("AllCourses", allCourseList);
intent.putExtras(bundle);*/
intent.putExtra("userinput",user_input);
startActivity(intent);

return false;
}

@Override
public boolean onQueryTextChange(String newText)
{
return false;
}
});

return super.onCreateOptionsMenu(menu);
}
// So that the 3 line in toolbar work
@Override
public boolean onOptionsItemSelected(MenuItem item)
```

```
 {
if(mToggle.onOptionsItemSelected(item)){
return true;
}
return super.onOptionsItemSelected(item);

}
}
```

FeedbackActivity

```
package com.example.collegify;

import android.annotation.SuppressLint;
import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.drawable.BitmapDrawable;
import android.renderscript.Allocation;
import android.renderscript.Element;
import android.renderscript.RenderScript;
import android.renderscript.ScriptIntrinsicBlur;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EdgeEffect;
import android.widget.EditText;
import android.widget.Toast;

import com.example.collegify.Lists.common;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.io.BufferedReader;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;

public class FeedbackActivity extends AppCompatActivity
 {
```

```
private EditText feedback;
private Button submit;

final DatabaseReference feedDatabase = FirebaseDatabase.getInstance().getReference("Feedback");

@Override
protected void onCreate(Bundle savedInstanceState)
{
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_feedback);

//To Blur the Background image
Bitmap bitmap = BitmapFactory.decodeResource(this.getResources(),R.drawable.loginbg);
//Drawable to bitmap
Bitmap blurredBitmap = blurRenderScript(this,bitmap, 10);
//second parametre is radius
findViewById(R.id.feedbackpg).setBackground
(new BitmapDrawable(getResources(), blurredBitmap));
//Assign the bitmap a


Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("FeedBack");

getSupportActionBar().setDisplayHomeAsUpEnabled(true);

feedback = (EditText) findViewById(R.id.Feedback);
submit = (Button) findViewById(R.id.submit);

submit.setOnClickListener(new View.OnClickListener()
 {
@Override
public void onClick(View v) {

storeFeedback();

}
});
}

private void storeFeedback() {
final String feedb = feedback.getText().toString();
final String date = new SimpleDateFormat("dd-MM-yyyy", Locale.getDefault()).format(new Date());
if (TextUtils.isEmpty(feedb)) {
//Feedback is empty
feedback.setError("Please enter a Feedback");
feedback.requestFocus();
//stopping the function execution further
```

```
return;
}
feedDatabase.addListenerForSingleValueEvent(new
ValueEventListener()
{
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot)
 {
feedDatabase.child(common.username).child(date).setValue(feedb);
Toast.makeText(getApplicationContext(),"Your feedback is recorded,
Thank you!" ,Toast.LENGTH_LONG ).show();
finish();
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError)
{
Toast.makeText(getApplicationContext(),databaseError.getMessage() ,Toast.LENGTH_LONG ).show();
}
});
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
switch (item.getItemId()) {
// Respond to the action bar's Up/Home button
case android.R.id.home:
//NavUtils.navigateUpFromSameTask(this);
finish();
return true;
}
return super.onOptionsItemSelected(item);
}
// For blurring Images
@SuppressLint("NewApi")
public static Bitmap blurRenderScript(Context context,
 Bitmap smallBitmap, int radius) {
try {
smallBitmap = RGB565toARGB888(smallBitmap);
} catch (Exception e) {
e.printStackTrace();
}

Bitmap bitmap = Bitmap.createBitmap(
smallBitmap.getWidth(), smallBitmap.getHeight(),
Bitmap.Config.ARGB_8888);

RenderScript renderScript = RenderScript.create(context);
```

```
Allocation blurInput = Allocation.createFromBitmap
(renderScript, smallBitmap);
Allocation blurOutput = Allocation.createFromBitmap
(renderScript, bitmap);

ScriptIntrinsicBlur blur = ScriptIntrinsicBlur.
create(renderScript,
Element.U8_4(renderScript));
blur.setInput(blurInput);
blur.setRadius(radius); // radius must be 0 < r <= 25
blur.forEach(blurOutput);

blurOutput.copyTo(bitmap);
renderScript.destroy();

return bitmap;

}

private static Bitmap RGB565toARGB888(Bitmap img)
throws Exception {
int numPixels = img.getWidth() * img.getHeight();
int[] pixels = new int[numPixels];

//Get JPEG pixels.  Each int is the color values for one pixel.
img.getPixels(pixels, 0, img.getWidth(), 0, 0, img.getWidth(),
img.getHeight());

//Create a Bitmap of the appropriate format.
Bitmap result = Bitmap.createBitmap(img.getWidth(),
img.getHeight(), Bitmap.Config.ARGB_8888);

//Set RGB pixels.
result.setPixels(pixels, 0, result.getWidth(), 0, 0,
result.getWidth(), result.getHeight());
return result;
}
}
```

FilterActivity

```
package com.example.collegify;

import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
```

```
import android.os.Bundle;
import android.support.v7.widget.GridLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.SearchView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.collegify.Adapters.CollegeListAdapter;
import com.example.collegify.Adapters.SubCategoryAdapter;
import com.example.collegify.Lists.AllClgList;
import com.example.collegify.Lists.ClgCategories;
import com.example.collegify.Lists.CollegeList;
import com.example.collegify.Lists.common;
import com.google.firebase.auth.FirebaseAuth;

import java.util.ArrayList;
import java.util.List;

public class FilterActivity extends AppCompatActivity
 {

private Toolbar toolbar;
private DrawerLayout mDrawerlayout;
private ActionBarDrawerToggle mToggle;

private RecyclerView recyclerView;
private RecyclerView.LayoutManager layoutManager;
private SubCategoryAdapter filteradapter;
private CollegeListAdapter Cfilteradapter;

private NavigationView navigationView;

List<ClgCategories> filteredList;
List<CollegeList> filteredList1;


@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_filter);


final String userinput = getIntent().getStringExtra
("userinput");
```

```
toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle(userinput);
mDrawerlayout = (DrawerLayout) findViewById(R.id.drawer);
mToggle = new ActionBarDrawerToggle(this, mDrawerlayout,
 R.string.open, R.string.close);
mDrawerlayout.addDrawerListener(mToggle);
mToggle.syncState();
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

//allCourseList = new ArrayList<>();
//allCollegeList = new ArrayList<>();

recyclerView = (RecyclerView) findViewById(R.id.recyclerview);
layoutManager = new GridLayoutManager(this, 1);
recyclerView.setLayoutManager(layoutManager);
recyclerView.setHasFixedSize(true);

applyfilters(userinput);

navigationView = (NavigationView) findViewById(R.id.navgview);

navigationView.setNavigationItemSelectedListener(new NavigationView.OnNavigationItemSelectedListener()
{
@Override
public boolean onNavigationItemSelected(@NonNull
MenuItem menuItem)
{
switch (menuItem.getItemId()) {
case R.id.Home:
Intent HomeIntent = new Intent(getApplicationContext(),
ProfileActivity.class);
startActivity(HomeIntent);
finish();
break;
case R.id.Bookmarks:
Intent BookmarkIntent = new Intent(getApplicationContext(), BookmarksActivity.class);
startActivity(BookmarkIntent);
finish();
break;
case R.id.gmaps:
Intent GoogleMapsIntent = new Intent(getApplicationContext(), GoogleMapsActivity.class);
startActivity(GoogleMapsIntent);
break;
case R.id.ChangePassword:
Intent ChgPassIntent = new Intent(getApplicationContext(), ChangePasswordActivity.class);
startActivity(ChgPassIntent);
break;
case R.id.ChangeSecurityQuestions:
```

```
Intent ChgSecQuest = new Intent(getApplicationContext(),
ChangeSecurityQuestionActivity.class);
startActivity(ChgSecQuest);
break;
case R.id.Feedback:
Intent FeedbackIntent = new Intent(getApplicationContext(),FeedbackActivity.class);
startActivity(FeedbackIntent);
break;
case R.id.Logout:
FirebaseAuth.getInstance().signOut();
finish();
startActivity(new Intent(getApplicationContext(),
MainActivity.class));

}
return false;
}
});
}
public void applyfilters(String userinput){
/*Bundle bundle = getIntent().getExtras();
allCourseList = bundle.getParcelableArrayList("AllCourses");
allCollegeList = bundle.getParcelableArrayList("AllColleges");
*/

filteredList1 = new ArrayList<>();
filteredList = new ArrayList<>();

for (AllClgList temp : common.allCourseList) {
if (temp.getName().toLowerCase().
contains(userinput.toLowerCase())) {
// || temp.getName1().toLowerCase().
contains(user_input.toLowerCase())
//user_input = temp.getName();

filteredList.add(new ClgCategories(temp.getName(),
temp.getImage()));
//Toast.makeText(getApplicationContext(), user_input,
Toast.LENGTH_LONG).show();
}

}
for (AllClgList temp : common.allCollegeList) {

if (temp.getName().toLowerCase().
contains(userinput.toLowerCase())|| temp.getName1().
toLowerCase().contains
(userinput.toLowerCase())) {
//|| temp.getName1().toLowerCase().
```

```
contains(userinput.toLowerCase())
//userinput = temp.getName();
filteredList1.add(new CollegeList(temp.getName()));
//  filteredList1.add(new CollegeList(temp.getName(),
 temp.getImage()));

}

}
Toast.makeText(getApplicationContext(), "Search Finished", Toast.LENGTH_LONG).show();

if(filteredList == null || filteredList.isEmpty()){

if(filteredList1 == null || filteredList1.isEmpty()) {
TextView message = (TextView) findViewById(R.id.text);
message.setText("No result found");
}
else {
Cfilteradapter = new CollegeListAdapter
(FilterActivity.this, filteredList1);
recyclerView.setAdapter(Cfilteradapter);
}
} else{
filteradapter = new SubCategoryAdapter
(FilterActivity.this, filteredList);
recyclerView.setAdapter(filteradapter);
}
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
MenuInflater inflater = getMenuInflater();
inflater.inflate(R.menu.search_menu, menu);
MenuItem item = menu.findItem(R.id.search);
SearchView searchView = (SearchView) item.getActionView();

searchView.setOnQueryTextListener
(new SearchView.OnQueryTextListener() {
@Override
public boolean onQueryTextSubmit(String query) {
String user_input = query.trim();
applyfilters(user_input);
return false;
}

@Override
public boolean onQueryTextChange(String newText) {
return false;
}
```

```
});

return super.onCreateOptionsMenu(menu);
}

// So that the 3 line in toolbar work
@Override
public boolean onOptionsItemSelected(MenuItem item) {
if (mToggle.onOptionsItemSelected(item)) {
return true;
}
if (item.getItemId() == R.id.search) {

}
return super.onOptionsItemSelected(item);

}
}
```

ForgotPasswordActivity

```
package com.example.collegify;
import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.drawable.BitmapDrawable;
import android.renderscript.Allocation;
import android.renderscript.Element;
import android.renderscript.RenderScript;
import android.renderscript.ScriptIntrinsicBlur;
import android.support.annotation.NonNull;
import android.support.v4.app.NavUtils;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;

public class ForgotPasswordActivity extends AppCompatActivity
{
```

```
private EditText email;
private Button forgotpass;


@Override
protected void onCreate(Bundle savedInstanceState)
{
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_forgot_password);

//To Blur the Background image
Bitmap bitmap = BitmapFactory.decodeResource
(this.getResources(),
R.drawable.loginbg);
//Drawable to bitmap
Bitmap blurredBitmap = blurRenderScript
(this,bitmap, 10);
//second parametre is radius
findViewById(R.id.forgotpasspg).setBackground(new BitmapDrawable(getResources(), blurredBitmap));
//Assign the bitmap as background

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Forgot Password");
toolbar.setSubtitle("Password Recovery");
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

email = (EditText) findViewById(R.id.email);
forgotpass = (Button) findViewById(R.id.forgotpass);

forgotpass.setOnClickListener(new View.OnClickListener()
 {
@Override
public void onClick(View v) {

Intent VerifySecQ = new Intent
(ForgotPasswordActivity.this,VerifySQ.class);
VerifySecQ.putExtra
("email",email.getText().toString() );
startActivity(VerifySecQ);
finish();

}
});

}
// For blurring Images
@SuppressLint("NewApi")
```

```
public static Bitmap blurRenderScript(Context context,
Bitmap smallBitmap, int radius) {
try {
smallBitmap = RGB565toARGB888(smallBitmap);
} catch (Exception e) {
e.printStackTrace();
}

Bitmap bitmap = Bitmap.createBitmap(
smallBitmap.getWidth(), smallBitmap.getHeight(),
Bitmap.Config.ARGB_8888);

RenderScript renderScript =
RenderScript.create(context);

Allocation blurInput =
Allocation.createFromBitmap(renderScript, smallBitmap);
Allocation blurOutput =
Allocation.createFromBitmap(renderScript, bitmap);

ScriptIntrinsicBlur blur =
ScriptIntrinsicBlur.create(renderScript,
Element.U8_4(renderScript));
blur.setInput(blurInput);
blur.setRadius(radius); // radius must be 0 < r <= 25
blur.forEach(blurOutput);

blurOutput.copyTo(bitmap);
renderScript.destroy();

return bitmap;

}

private static Bitmap RGB565toARGB888(Bitmap img)
 throws Exception {
int numPixels = img.getWidth() * img.getHeight();
int[] pixels = new int[numPixels];

//Get JPEG pixels.
Each int is the color values for one pixel.
img.getPixels(pixels, 0, img.getWidth(), 0, 0,
img.getWidth(), img.getHeight());

//Create a Bitmap of the appropriate format.
Bitmap result = Bitmap.createBitmap(img.getWidth(),
 img.getHeight(), Bitmap.Config.ARGB_8888);

//Set RGB pixels.
```

```
result.setPixels(pixels, 0, result.getWidth(), 0, 0,
result.getWidth(), result.getHeight());
return result;
}
@Override
public boolean onOptionsItemSelected(MenuItem item) {
switch (item.getItemId()) {
// Respond to the action bar's Up/Home button
case android.R.id.home:
NavUtils.navigateUpFromSameTask(this);
return true;
}
return super.onOptionsItemSelected(item);
}
}
```

GoogleMapsActivity

```
package com.example.collegify;

import android.Manifest;
import android.content.pm.PackageManager;
import android.location.Address;
import android.location.Geocoder;
import android.location.Location;
import android.os.Build;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.support.v4.app.ActivityCompat;
import android.support.v4.app.FragmentActivity;
import android.os.Bundle;
import android.support.v4.content.ContextCompat;
import android.text.TextUtils;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.Toast;


import com.example.collegify.GoogleMaps.GetNearbyPlaces;
import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
```

```
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.tasks.TaskExecutors;

import java.io.IOException;
import java.util.List;

public class GoogleMapsActivity extends FragmentActivity implements
OnMapReadyCallback,
GoogleApiClient.ConnectionCallbacks,
GoogleApiClient.OnConnectionFailedListener,
LocationListener
{

private GoogleMap mMap;
private GoogleApiClient googleApiClient;
private LocationRequest locationRequest;
private Location lastLocation;
private Marker currentUserLocationMarker;
private static final int Request_User_Location_Code = 99;
private double latitude,longitude;
private int ProximityRadius = 10000;


@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_google_maps);

if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.M){
checkUserLocationPermission();
}

// Obtain the SupportMapFragment and get notified
when the map is ready to be used.
SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager()
.findFragmentById(R.id.map);
mapFragment.getMapAsync(this);
}

public void onClick(View v) {

Object transferData[] = new Object[2];
GetNearbyPlaces getNearbyPlaces = new GetNearbyPlaces();

switch (v.getId()){
```

```
case R.id.search_addr:
EditText addressField = (EditText) findViewById(R.id.location_search);
String address = addressField.getText().toString();

List<Address> addressList = null;
MarkerOptions userMarkerOptions = new MarkerOptions();

if(!TextUtils.isEmpty(address)){
Geocoder geocoder = new Geocoder(this);

try {
addressList = geocoder.getFromLocationName(address,6 );

if(addressList != null){
for(int i = 0; i < addressList.size(); i++){
Address  userAddress = addressList.get(i);
LatLng latLng = new LatLng(userAddress.getLatitude(),
userAddress.getLongitude() );
userMarkerOptions.position(latLng);
userMarkerOptions.title(address);
userMarkerOptions.icon(BitmapDescriptorFactory.defaultMarker
(BitmapDescriptorFactory.HUE_MAGENTA));

mMap.addMarker(userMarkerOptions);
mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
mMap.animateCamera(CameraUpdateFactory.zoomTo(10));
}
}else {
Toast.makeText(this,"Location Not Found" ,
Toast.LENGTH_SHORT).show();

}
} catch (IOException e) {
e.printStackTrace();
}
}else {
Toast.makeText(this,"Please Write Any Location Name" , Toast.LENGTH_SHORT).show();
}
break;
case R.id.nearby_colleges:
mMap.clear(); //Clear all markers
String url = getUrl(latitude,longitude,"university");
transferData[0] = mMap;
transferData[1] = url;
getNearbyPlaces.execute(transferData);
Toast.makeText(this,"Searching for nearby Colleges" ,
 Toast.LENGTH_SHORT).show();
Toast.makeText(this,"Showing nearby Colleges" ,
 Toast.LENGTH_SHORT).show();
```

```
break;
}
}

private String getUrl(double latitude,double longitude,
 String college){

StringBuilder googleurl = new StringBuilder
("https://maps.googleapis.com/maps/api
/place/nearbysearch/json?");
googleurl.append("location=" + latitude + "," + longitude);
googleurl.append("&radius=" + ProximityRadius);
googleurl.append("&type=" + college);
googleurl.append("&sensor=true");
googleurl.append("&key=" + "AIzaSyBwZKOd6FJe58qVuCAVKHMyEuSDiHzmcbs");

Log.d("GoogleMapsActivity","url="+googleurl.toString());

return googleurl.toString();

}
@Override
public void onMapReady(GoogleMap googleMap) {
mMap = googleMap;

if(ContextCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED){
// TODO: Consider calling

buildGoogleApiClient();

mMap.setMyLocationEnabled(true);
}

}

public boolean checkUserLocationPermission(){
if (ContextCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION ) !=
PackageManager.PERMISSION_GRANTED) {
//if the permission is not granted
if (ActivityCompat.shouldShowRequestPermissionRationale(this, Manifest.permission.
ACCESS_FINE_LOCATION)) {
ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.ACCESS_FINE_LOCATION},
Request_User_Location_Code);
} else {
ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.ACCESS_FINE_LOCATION},
Request_User_Location_Code);
}
```

```
return false;
} else {
return true;
}
}

@Override
public void onRequestPermissionsResult(int requestCode,
@NonNull String[] permissions, @NonNull int[] grantResults)
{
switch (requestCode){
case Request_User_Location_Code:
if(grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED){
if (ContextCompat.checkSelfPermission(this,Manifest.permission.
ACCESS_FINE_LOCATION ) == PackageManager.PERMISSION_GRANTED){
if (googleApiClient == null){
buildGoogleApiClient();
}
mMap.setMyLocationEnabled(true);
}
}else {
Toast.makeText(this, "Permissions Denied...",Toast.LENGTH_SHORT ).
show();
}
return;
}
}

protected synchronized void buildGoogleApiClient(){
//Build the client
googleApiClient = new GoogleApiClient.Builder(this)
.addConnectionCallbacks(this)
.addOnConnectionFailedListener(this)
.addApi(LocationServices.API)
.build();

googleApiClient.connect(); // Connect to the client
}

@Override
public void onLocationChanged(Location location) {

latitude = location.getLatitude();
longitude = location.getLongitude();

lastLocation = location;

//Remove the location marker if it is already set to another location
if(currentUserLocationMarker != null){
```

```
currentUserLocationMarker.remove();
}
LatLng latLng = new LatLng(location.getLatitude(),location.
getLongitude() );
MarkerOptions markerOptions = new MarkerOptions();
markerOptions.position(latLng);
markerOptions.title("User's Current Location");
markerOptions.icon(BitmapDescriptorFactory.defaultMarker
(BitmapDescriptorFactory.HUE_ROSE));

currentUserLocationMarker = mMap.addMarker(markerOptions);

mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
mMap.animateCamera(CameraUpdateFactory.zoomBy(12));

if(googleApiClient != null){
LocationServices.FusedLocationApi.removeLocationUpdates
(googleApiClient,this);
}
}

@Override
public void onConnected(@Nullable Bundle bundle) {

locationRequest = new LocationRequest();
locationRequest.setInterval(1100);
locationRequest.setFastestInterval(1100);
locationRequest.setPriority(LocationRequest.
PRIORITY_BALANCED_POWER_ACCURACY);

if(ContextCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED){
LocationServices.FusedLocationApi.requestLocationUpdates
(googleApiClient, locationRequest, this);
}

}

@Override
public void onConnectionSuspended(int i) {

}

@Override
public void onConnectionFailed(@NonNull
 ConnectionResult connectionResult) {

}
}
```

MainActivity

```
package com.example.collegify;

import android.annotation.SuppressLint;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Paint;
import android.graphics.drawable.BitmapDrawable;
import android.renderscript.Allocation;
import android.renderscript.Element;
import android.renderscript.RenderScript;
import android.renderscript.ScriptIntrinsicBlur;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.example.collegify.Lists.common;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

private Button submit;
private EditText email;
private EditText passwd;
private TextView signup;
private TextView forgotpass;
private TextView phone;

private ProgressDialog progressDialog;
private FirebaseAuth firebaseAuth;

@Override
```

```
protected void onCreate(Bundle savedInstanceState)
{
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);

firebaseAuth = FirebaseAuth.getInstance();
progressDialog = new ProgressDialog(this);

//To Blur the Background image
Bitmap bitmap = BitmapFactory.decodeResource(this.getResources(), R.drawable.loginbg);
//Drawable to bitmap
Bitmap blurredBitmap = blurRenderScript(this, bitmap, 10);
//second parametre is radius
findViewById(R.id.loginpg).setBackground(new BitmapDrawable
(getResources(), blurredBitmap));
//Assign the bitmap as background

submit = (Button) findViewById(R.id.submit);
email = (EditText) findViewById(R.id.email);
passwd = (EditText) findViewById(R.id.passwd);
signup = (TextView) findViewById(R.id.signup);
signup.setPaintFlags(signup.getPaintFlags() |
Paint.UNDERLINE_TEXT_FLAG);
forgotpass = (TextView) findViewById(R.id.forgotpass);
forgotpass.setPaintFlags(forgotpass.getPaintFlags() |
 Paint.UNDERLINE_TEXT_FLAG);
phone = (TextView) findViewById(R.id.phone);
phone.setPaintFlags(phone.getPaintFlags() | Paint.UNDERLINE_TEXT_FLAG);


submit.setOnClickListener(this);
signup.setOnClickListener(this);
forgotpass.setOnClickListener(this);
phone.setOnClickListener(this);


}

private void loginUser() {
final String eml = email.getText().toString().trim();
final String pass = passwd.getText().toString().trim();

if (TextUtils.isEmpty(eml)) {
//email is empty
//Toast.makeText(this, "Please enter email",
Toast.LENGTH_SHORT).show();
email.setError("Please enter email");
email.requestFocus();
//stopping the function execution further
```

```
return;
}
if (TextUtils.isEmpty(pass)) {
//password is empty
//Toast.makeText(this, "Please enter password",
Toast.LENGTH_SHORT).show();
passwd.setError("Please enter password");
passwd.requestFocus();
//stopping the function execution further
return;
}
//if validations are ok
//Register the user
progressDialog.setMessage("Logging In...");
progressDialog.show();

firebaseAuth.signInWithEmailAndPassword(eml, pass)
.addOnCompleteListener(this, new OnCompleteListener<AuthResult>()
{
@Override
public void onComplete(@NonNull Task<AuthResult> task) {

progressDialog.dismiss();
if (task.isSuccessful()) {
Toast.makeText(getApplicationContext(), "CP1", Toast.LENGTH_SHORT).show();

//open the profile activity
final DatabaseReference mDatabase = FirebaseDatabase.getInstance().getReference("Users");
mDatabase.addListenerForSingleValueEvent(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

String email = "";
String usrname = "";

for (DataSnapshot postsnapshot : dataSnapshot.getChildren()) {
if (postsnapshot.child("email").getValue().equals(eml)) {
email = postsnapshot.child("email").getValue(String.class);
usrname = postsnapshot.child("uname").getValue(String.class);
}
}

Intent startIntent = new Intent(getApplicationContext(),
 ProfileActivity.class);
startIntent.putExtra("email", email);
startIntent.putExtra("uname", usrname);
startActivity(startIntent);
finish();
Toast.makeText(getApplicationContext(), "Login Successfully", Toast.LENGTH_SHORT).show();
```

```
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
}
});

} else {

Toast.makeText(getApplicationContext(), task.getException().getMessage(), Toast.LENGTH_SHORT).show();
}
}
});

}

@Override
public void onClick(View view) {
switch (view.getId()) {
case R.id.submit:
loginUser();
break;
case R.id.signup:
//open Signup activity

Intent startIntent = new Intent(getApplicationContext(),
SignUpActivity.class);
startIntent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
startActivity(startIntent);


//finish();
break;
case R.id.forgotpass:
//open forgot password activity

Intent forgotpassIntent = new Intent(getApplicationContext(), ForgotPasswordActivity.class);
startActivity(forgotpassIntent);
break;
case R.id.phone:
//open phone login activity

Intent phoneLoginIntent = new Intent(getApplicationContext(),
PhoneLogin.class);
startActivity(phoneLoginIntent);
break;
}
```

```
}

// For blurring Images
@SuppressLint("NewApi")
public static Bitmap blurRenderScript(Context context, Bitmap smallBitmap, int radius) {
try {
smallBitmap = RGB565toARGB888(smallBitmap);
} catch (Exception e) {
e.printStackTrace();
}

Bitmap bitmap = Bitmap.createBitmap(
smallBitmap.getWidth(), smallBitmap.getHeight(),
Bitmap.Config.ARGB_8888);

RenderScript renderScript = RenderScript.create(context);

Allocation blurInput = Allocation.createFromBitmap
(renderScript, smallBitmap);
Allocation blurOutput = Allocation.createFromBitmap
(renderScript, bitmap);

ScriptIntrinsicBlur blur = ScriptIntrinsicBlur.create(renderScript,
Element.U8_4(renderScript));
blur.setInput(blurInput);
blur.setRadius(radius); // radius must be 0 < r <= 25
blur.forEach(blurOutput);

blurOutput.copyTo(bitmap);
renderScript.destroy();

return bitmap;

}

private static Bitmap RGB565toARGB888(Bitmap img) throws Exception {
int numPixels = img.getWidth() * img.getHeight();
int[] pixels = new int[numPixels];

//Get JPEG pixels.  Each int is the color values for one pixel.
img.getPixels(pixels, 0, img.getWidth(), 0, 0, img.getWidth(),
 img.getHeight());

//Create a Bitmap of the appropriate format.
Bitmap result = Bitmap.createBitmap(img.getWidth(), img.getHeight(), Bitmap.Config.ARGB_8888);

//Set RGB pixels.
result.setPixels(pixels, 0, result.getWidth(), 0, 0, result.getWidth(), result.getHeight());
```

```
return result;
}
}
```

PhoneLogin

```
package com.example.collegify;

import android.content.Intent;
import android.graphics.Paint;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

import com.google.firebase.auth.FirebaseAuth;

public class PhoneLogin extends AppCompatActivity {
private Button buttonContinue ;
private Button submit;
private EditText editTextMobile;
private TextView emailLogin;

FirebaseAuth mAuth;

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_phone_login);

mAuth = FirebaseAuth.getInstance();

emailLogin = (TextView) findViewById(R.id.emailLogin);
emailLogin.setPaintFlags(emailLogin.getPaintFlags() |
 Paint.UNDERLINE_TEXT_FLAG);
editTextMobile = (EditText) findViewById(R.id.editTextMobile);
buttonContinue = (Button) findViewById(R.id.buttonContinue);

buttonContinue.setOnClickListener(new View.OnClickListener()
 {
@Override
public void onClick(View v) {

String code = "91";
String phone = editTextMobile.getText().toString().trim();
if(phone.isEmpty() || phone.length() < 10){
editTextMobile.setError("Valid Phone number is required");
editTextMobile.requestFocus();
```

```
return;
}
String phoneNumber = "+" + code + phone;
Intent intent = new Intent(getApplicationContext(),VerifyPhone.class);
intent.putExtra("phonenumber", phoneNumber);
startActivity(intent);
}
});

emailLogin.setOnClickListener(new View.OnClickListener()
{
@Override
public void onClick(View v) {
//open E-mail login activity
Intent startIntent = new Intent(getApplicationContext(),
MainActivity.class);
startIntent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);//doesnot allow the previous
activity to lose its edit text contrnts
startActivity(startIntent);
}
});

}

}
```

ProfileActivity

```
package com.example.collegify;

import android.content.Intent;
import android.os.Bundle;
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.GridLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.SearchView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.collegify.Adapters.RecyclerViewAdapter;
import com.example.collegify.Lists.AllClgList;
```

```
import com.example.collegify.Lists.ClgCategories;
import com.example.collegify.Lists.common;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.List;


import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

/*
* Created by Hitanshu on 26-02-2019.
*/

public class ProfileActivity extends AppCompatActivity {

private ActionBarDrawerToggle mToggle;

List<ClgCategories> categoryList;

private TextView user;
private RecyclerView recyclerView;
private RecyclerView.LayoutManager layoutManager;
private RecyclerViewAdapter myAdapter;
private NavigationView navigationView;

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_profile);

common.e_mail = getIntent().getStringExtra("email");
common.username = getIntent().getStringExtra("uname");

navigationView = (NavigationView) findViewById(R.id.navgview);
View headerView = navigationView.getHeaderView(0);
user = (TextView) headerView.findViewById(R.id.user);
user.setText(common.username);

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Profile");
```

```
DrawerLayout mDrawerlayout = (DrawerLayout)
 findViewById(R.id.drawer);
mToggle = new ActionBarDrawerToggle(this,
mDrawerlayout, R.string.open, R.string.close);
mDrawerlayout.addDrawerListener(mToggle);
mToggle.syncState();

getSupportActionBar().setDisplayHomeAsUpEnabled(true);

categoryList = new ArrayList<>();
common.allCourseList = new ArrayList<>();
common.allCollegeList = new ArrayList<>();
recyclerView = (RecyclerView) findViewById(R.id.recyclerview);
layoutManager = new GridLayoutManager(this, 2);
recyclerView.setLayoutManager(layoutManager);
recyclerView.setHasFixedSize(true);

common.allCourseList.clear();
common.allCollegeList.clear();
//////////////////////////////
// ////////////////////////////////////////////////////
DatabaseReference mDatabase = FirebaseDatabase.getInstance().getReference("ClgSubCategories");
mDatabase.addValueEventListener(new ValueEventListener()
{
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot)
 {
for (DataSnapshot postSnapshot : dataSnapshot.getChildren())
{

for (DataSnapshot postSnapshot1 : postSnapshot.getChildren()) {
AllClgList allClgList = postSnapshot1.getValue(AllClgList.class);
String name = allClgList.getName().trim();
String name1 = allClgList.getName1();
String image = allClgList.getImage();
common.allCourseList.add(new AllClgList(name, name1, image));
}
}
//Toast.makeText(ProfileActivity.this,"Got Courses!",
Toast.LENGTH_SHORT).show();
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError)
{
// Getting Post failed, log a message
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
```

```
// ...
}
});

mDatabase = FirebaseDatabase.getInstance().getReference("Colleges");
mDatabase.addValueEventListener(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot)
{
for (DataSnapshot postSnapshot : dataSnapshot.getChildren())
{
String name = postSnapshot.child("name").getValue(String.class);
String name1 = postSnapshot.child("name1").getValue(String.class);
String image = "";//postSnapshot.child("image").getValue(String.class);
common.allCollegeList.add(new AllClgList(name, name1, image));
}
//Toast.makeText(ProfileActivity.this,"Got Colleges!",
Toast.LENGTH_SHORT).show();
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
}
});
////////////////////////////////////////////////////////////////////////////////

mDatabase = FirebaseDatabase.getInstance().
getReference("ClgCategories");
mDatabase.addValueEventListener(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
for (DataSnapshot postSnapshot : dataSnapshot.getChildren()) {
String name = postSnapshot.child("name").getValue(String.class);
String image = postSnapshot.child("image").getValue(String.class);
categoryList.add(new ClgCategories(name,image ));
}
myAdapter = new RecyclerViewAdapter(ProfileActivity.this, categoryList);
 //Call the constructor
recyclerView.setAdapter(myAdapter);
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
}
});
```

```
navigationView.setNavigationItemSelectedListener(new NavigationView.OnNavigationItemSelectedListener()
{
@Override
public boolean onNavigationItemSelected(@NonNull MenuItem menuItem) {
switch (menuItem.getItemId()) {
case R.id.Home:
Intent HomeIntent = new Intent(getApplicationContext(),
ProfileActivity.class);
startActivity(HomeIntent);
finish();
break;
case R.id.Bookmarks:
Intent BookmarkIntent = new Intent(getApplicationContext(), BookmarksActivity.class);
startActivity(BookmarkIntent);
finish();
break;
case R.id.gmaps:
Intent GoogleMapsIntent = new Intent(getApplicationContext(), GoogleMapsActivity.class);
startActivity(GoogleMapsIntent);
break;
case R.id.ChangePassword:
Intent ChgPassIntent = new Intent(getApplicationContext(), ChangePasswordActivity.class);
startActivity(ChgPassIntent);
break;
case R.id.ChangeSecurityQuestions:
Intent ChgSecQuest = new Intent(getApplicationContext(),ChangeSecurityQuestionActivity.class);
startActivity(ChgSecQuest);
break;
case R.id.Feedback:
Intent FeedbackIntent = new Intent(getApplicationContext(),FeedbackActivity.class);
startActivity(FeedbackIntent);
break;
case R.id.Logout:
FirebaseAuth.getInstance().signOut();
finish();
startActivity(new Intent(getApplicationContext(), MainActivity.class));

}
return false;
}
});
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
final MenuInflater inflater = getMenuInflater();
inflater.inflate(R.menu.search_menu, menu);
```

```
MenuItem item = menu.findItem(R.id.search);
SearchView searchView = (SearchView) item.getActionView();

searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
@Override
public boolean onQueryTextSubmit(String query) {
String user_input = query.trim();
Intent intent = new Intent(getApplicationContext(), FilterActivity.class);
intent.putExtra("userinput", user_input);
startActivity(intent);
return false;
}

@Override
public boolean onQueryTextChange(String newText) {
return false;
}
});

return super.onCreateOptionsMenu(menu);
}

// So that the 3 line in toolbar work
@Override
public boolean onOptionsItemSelected(MenuItem item) {
if (mToggle.onOptionsItemSelected(item)) {
return true;
}
return super.onOptionsItemSelected(item);

}
}
```

SecurityQuestionActivity

```
package com.example.collegify;

import android.annotation.SuppressLint;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.drawable.BitmapDrawable;
import android.renderscript.Allocation;
import android.renderscript.Element;
import android.renderscript.RenderScript;
import android.renderscript.ScriptIntrinsicBlur;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
```

```
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

public class SecurityQuestionActivity extends AppCompatActivity
{

private EditText question;
private EditText answer;
private Button submit;
private ProgressDialog progressDialog;

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_security_question);

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Security Question");
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

//To Blur the Background image
Bitmap bitmap = BitmapFactory.decodeResource(this.getResources(),R.drawable.loginbg);
//Drawable to bitmap
Bitmap blurredBitmap = blurRenderScript(this,bitmap, 10);
//second parametre is radius
findViewById(R.id.sqpg).setBackground(new BitmapDrawable
(getResources(), blurredBitmap));//Assign the bitmap as background

question = (EditText) findViewById(R.id.squestion);
answer = (EditText) findViewById(R.id.answer);
submit = (Button) findViewById(R.id.submit);

final String uname = getIntent().getStringExtra("uname");

progressDialog = new ProgressDialog(this);
```

```
submit.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
storeSecurityQuestion(uname);
}
});


}

private void storeSecurityQuestion(final String username) {
final String q = question.getText().toString().trim();
final String a = answer.getText().toString().trim();

final DatabaseReference mDatabase = FirebaseDatabase.getInstance().getReference("Users");
progressDialog.setMessage("Please Wait...");
progressDialog.show();
mDatabase.addListenerForSingleValueEvent(new ValueEventListener()
{
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
progressDialog.dismiss();
mDatabase.child(username).child("question").setValue(q);
mDatabase.child(username).child("answer").setValue(a);
Intent intent = new Intent(getApplicationContext(),
MainActivity.class);
startActivity(intent);
return;

}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
progressDialog.dismiss();
Toast.makeText(getApplicationContext(),databaseError.getMessage() ,Toast.LENGTH_LONG ).show();
return;
}
});

}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
switch (item.getItemId()) {
// Respond to the action bar's Up/Home button
case android.R.id.home:
//NavUtils.navigateUpFromSameTask(this);
finish();
return true;
```

```
}
return super.onOptionsItemSelected(item);
}
// For blurring Images
@SuppressLint("NewApi")
public static Bitmap blurRenderScript(Context context,
 Bitmap smallBitmap, int radius) {
try {
smallBitmap = RGB565toARGB888(smallBitmap);
} catch (Exception e) {
e.printStackTrace();
}

Bitmap bitmap = Bitmap.createBitmap(
smallBitmap.getWidth(), smallBitmap.getHeight(),
Bitmap.Config.ARGB_8888);

RenderScript renderScript = RenderScript.create(context);

Allocation blurInput = Allocation.createFromBitmap
(renderScript, smallBitmap);
Allocation blurOutput = Allocation.createFromBitmap
(renderScript, bitmap);

ScriptIntrinsicBlur blur = ScriptIntrinsicBlur.create(renderScript,
Element.U8_4(renderScript));
blur.setInput(blurInput);
blur.setRadius(radius); // radius must be 0 < r <= 25
blur.forEach(blurOutput);

blurOutput.copyTo(bitmap);
renderScript.destroy();

return bitmap;

}

private static Bitmap RGB565toARGB888(Bitmap img) throws Exception
{
int numPixels = img.getWidth() * img.getHeight();
int[] pixels = new int[numPixels];

//Get JPEG pixels.  Each int is the color values for one pixel.
img.getPixels(pixels, 0, img.getWidth(), 0, 0, img.getWidth(),
img.getHeight());

//Create a Bitmap of the appropriate format.
Bitmap result = Bitmap.createBitmap(img.getWidth(), img.getHeight(), Bitmap.Config.ARGB_8888);
```

```
//Set RGB pixels.
result.setPixels(pixels, 0, result.getWidth(), 0, 0, result.getWidth(), result.getHeight());
return result;
}
}
```

SignUpActivity

```
package com.example.collegify;

import android.app.ProgressDialog;
import android.content.Intent;
import android.graphics.Paint;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseAuthUserCollisionException;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

public class SignUpActivity extends AppCompatActivity implements View.OnClickListener {

private Button submit;
private EditText uname;
private EditText email;
private EditText passwd;
private EditText confpasswd;
private TextView login;

private ProgressDialog progressDialog;
private FirebaseAuth firebaseAuth;

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

```
setContentView(R.layout.activity_sign_up);


firebaseAuth = FirebaseAuth.getInstance();
progressDialog = new ProgressDialog(this);

submit = (Button) findViewById(R.id.submit);
uname = (EditText) findViewById(R.id.name);
email = (EditText) findViewById(R.id.email);
passwd = (EditText) findViewById(R.id.passwd);
confpasswd = (EditText) findViewById(R.id.confpasswd);
login = (TextView) findViewById(R.id.log);
login.setPaintFlags(login.getPaintFlags() |
Paint.UNDERLINE_TEXT_FLAG);

submit.setOnClickListener(this);
login.setOnClickListener(this);
}

private void registerUser() {
final String unm = uname.getText().toString().trim();
final String eml = email.getText().toString().trim();
final String pass = passwd.getText().toString().trim();
String confpass = confpasswd.getText().toString().trim();

if (TextUtils.isEmpty(eml)) {
//email is empty
//Toast.makeText(this, "Please enter email",
Toast.LENGTH_SHORT).show();
email.setError("Please enter email");
email.requestFocus();
//stopping the function execution further
return;
}
if (TextUtils.isEmpty(pass)) {
//password is empty
//Toast.makeText(this, "Please enter password", Toast.LENGTH_SHORT).show();
passwd.setError("Please enter password");
passwd.requestFocus();
//stopping the function execution further
return;
}
if (TextUtils.isEmpty(confpass)) {
//password is empty
//Toast.makeText(this, "Please enter password", Toast.LENGTH_SHORT).show();
confpasswd.setError("Please re-enter password");
confpasswd.requestFocus();
//stopping the function execution further
return;
```

```
}
if(confpass.equals(pass)){
//if validations are ok
//Register the user
progressDialog.setMessage("Registering User...");
progressDialog.show();

final DatabaseReference mDatabase = FirebaseDatabase.getInstance().getReference("Users");
mDatabase.addListenerForSingleValueEvent(new ValueEventListener()
 {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
if(dataSnapshot.child(unm).exists()) {
progressDialog.dismiss();
Toast.makeText(getApplicationContext(),"Username is already taken,
Enter a different Username" , Toast.LENGTH_LONG).show();
return;
} else {
progressDialog.dismiss();
mDatabase.child(unm).child("email").setValue(eml);
mDatabase.child(unm).child("uname").setValue(unm);
mDatabase.child(unm).child("Bookmarks").child("Sample").
setValue("sample");
firebaseAuth.createUserWithEmailAndPassword(eml, pass).
addOnCompleteListener(new OnCompleteListener<AuthResult>() {
@Override
public void onComplete(@NonNull Task<AuthResult> task) {

progressDialog.dismiss();
if (task.isSuccessful()) {
Toast.makeText(getApplicationContext(), "Registered Successfully", Toast.LENGTH_SHORT).show();
Intent startIntent = new Intent(getApplicationContext(), SecurityQuestionActivity.class);
startIntent.putExtra("uname", unm);
startIntent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
//doesnot allow the previous activity to lose its edit text contrnts
startActivity(startIntent);
finish();
} else {
if (task.getException() instanceof FirebaseAuthUserCollisionException)
{
Toast.makeText(getApplicationContext(), "You are already registered", Toast.LENGTH_SHORT).show();

} else {
Toast.makeText(getApplicationContext(), task.getException().
getMessage() , Toast.LENGTH_SHORT).show();
}
}

}
```

```
});
}
return;
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
Toast.makeText(getApplicationContext(),databaseError.getMessage() ,Toast.LENGTH_LONG ).show();
}
});

}
else {
Toast.makeText(this, "The re-entered password doesn't match", Toast.LENGTH_SHORT).show();
return;
}
}
@Override
public void onClick(View view) {
if (view == submit) {
registerUser();
}
if (view == login) {
//open login activity
Intent startIntent = new Intent(getApplicationContext(),
MainActivity.class);
startIntent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);//doesnot allow the previous
activity to lose its edit text contrnts
startActivity(startIntent);
}
}



}
```

ReviewActivity.java

```
package com.example.collegify;

import android.content.Intent;
import android.graphics.Color;
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.GridLayoutManager;
import android.support.v7.widget.RecyclerView;
```

```
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

import com.example.collegify.Adapters.CollegeListAdapter;
import com.example.collegify.Adapters.CourseRVAdapter;
import com.example.collegify.Adapters.CustomSwipeAdapter;
import com.example.collegify.Adapters.ReviewAdapter;
import com.example.collegify.Lists.CollegeList;
import com.example.collegify.Lists.ReviewClass;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.List;

public class ReviewActivity extends AppCompatActivity {

private Toolbar toolbar;
List<ReviewClass> reviewList;


private RecyclerView recyclerView;
private RecyclerView.LayoutManager layoutManager;

private ReviewAdapter reviewAdapter;

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_review);

toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Reviews");
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

reviewList = new ArrayList<>();
recyclerView = (RecyclerView) findViewById(R.id.recyclerview);
layoutManager = new GridLayoutManager(this, 1);
recyclerView.setLayoutManager(layoutManager);
recyclerView.setHasFixedSize(true);

final String clgnm = getIntent().getStringExtra("clgnm");
```

```
final DatabaseReference mDatabase = FirebaseDatabase.getInstance().
getReference("Colleges");
mDatabase.addValueEventListener(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
for (DataSnapshot postSnapshot: dataSnapshot.getChildren()) {
String name =  postSnapshot.child("name").getValue(String.class);;
if (name.toLowerCase().equals(clgnm.toLowerCase())) {

DatabaseReference revref = FirebaseDatabase.getInstance().
getReference("Colleges/" + postSnapshot.getKey() + "/reviews" );
revref.addValueEventListener(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
for(DataSnapshot dp: dataSnapshot.getChildren()){
String uname = dp.child("user").getValue(String.class);
String rev = dp.child("review").getValue(String.class);
reviewList.add(new ReviewClass(uname,rev));
}
if(reviewList.isEmpty()){
Toast.makeText(getApplicationContext(),"No Reviews Available" ,
Toast.LENGTH_SHORT ).show();
} else {
reviewAdapter = new ReviewAdapter(ReviewActivity.this, reviewList);
recyclerView.setAdapter(reviewAdapter);
}
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
}
});
}

}


}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
}
});
```

```
Button giverevBtn = (Button) findViewById(R.id.grev);
giverevBtn.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
Intent intent = new Intent(getApplicationContext(),
GiveReviewActivity.class);
intent.putExtra("clgnm",clgnm );
startActivity(intent);
}
});
}
@Override
public boolean onOptionsItemSelected(MenuItem item) {
switch (item.getItemId()) {
// Respond to the action bar's Up/Home button
case android.R.id.home:
//NavUtils.navigateUpFromSameTask(this);
finish();
return true;
}
return super.onOptionsItemSelected(item);
}
}
```

GiveReviewActivity.java

```
package com.example.collegify;

import android.annotation.SuppressLint;
import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.drawable.BitmapDrawable;
import android.renderscript.Allocation;
import android.renderscript.Element;
import android.renderscript.RenderScript;
import android.renderscript.ScriptIntrinsicBlur;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.example.collegify.Adapters.ReviewAdapter;
import com.example.collegify.Lists.ReviewClass;
```

```
import com.example.collegify.Lists.common;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;

public class GiveReviewActivity extends AppCompatActivity {

private EditText rev;
private Button submit;
private String clgnm;

final DatabaseReference mDatabase = FirebaseDatabase.getInstance().
getReference("Feedback");
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_give_review);

//To Blur the Background image
Bitmap bitmap = BitmapFactory.decodeResource(this.getResources(),
R.drawable.loginbg);
//Drawable to bitmap
Bitmap blurredBitmap = blurRenderScript(this,bitmap, 10);
//second parametre is radius
findViewById(R.id.reviewpg).setBackground(new BitmapDrawable
(getResources(), blurredBitmap));//Assign the bitmap a

clgnm = getIntent().getStringExtra("clgnm");

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Review");

getSupportActionBar().setDisplayHomeAsUpEnabled(true);

rev = (EditText) findViewById(R.id.grev);
submit = (Button) findViewById(R.id.submit);

submit.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {

storeRev();
```

```
}
});
}

private void storeRev() {
final String srev = rev.getText().toString();
if (TextUtils.isEmpty(srev)) {
//Feedback is empty
rev.setError("Please enter a Feedback");
rev.requestFocus();
//stopping the function execution further
return;
}
final DatabaseReference mDatabase = FirebaseDatabase.getInstance().
getReference("Colleges");
mDatabase.addListenerForSingleValueEvent(new ValueEventListener()
{
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
for (DataSnapshot postSnapshot: dataSnapshot.getChildren()) {
String name =  postSnapshot.child("name").getValue(String.class);;
if (name.toLowerCase().equals(clgnm.toLowerCase())) {

final DatabaseReference revref = FirebaseDatabase.getInstance().
getReference("Colleges/" + postSnapshot.getKey() + "/reviews" );

revref.addListenerForSingleValueEvent(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
revref.child(common.username).child("user").setValue(common.username);
revref.child(common.username).child("review").setValue(srev);
Toast.makeText(getApplicationContext(),"Your review is recorded, Thank you!" ,
Toast.LENGTH_LONG ).show();
finish();
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
}
});
}

}


}
```

```
@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e, Toast.LENGTH_SHORT).show();
}
});
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
switch (item.getItemId()) {
// Respond to the action bar's Up/Home button
case android.R.id.home:
//NavUtils.navigateUpFromSameTask(this);
finish();
return true;
}
return super.onOptionsItemSelected(item);
}
// For blurring Images
@SuppressLint("NewApi")
public static Bitmap blurRenderScript(Context context,
Bitmap smallBitmap, int radius) {
try {
smallBitmap = RGB565toARGB888(smallBitmap);
} catch (Exception e) {
e.printStackTrace();
}

Bitmap bitmap = Bitmap.createBitmap(
smallBitmap.getWidth(), smallBitmap.getHeight(),
Bitmap.Config.ARGB_8888);

RenderScript renderScript = RenderScript.create(context);

Allocation blurInput = Allocation.createFromBitmap(renderScript, smallBitmap);
Allocation blurOutput = Allocation.createFromBitmap(renderScript, bitmap);

ScriptIntrinsicBlur blur = ScriptIntrinsicBlur.create(renderScript,
Element.U8_4(renderScript));
blur.setInput(blurInput);
blur.setRadius(radius); // radius must be 0 < r <= 25
blur.forEach(blurOutput);

blurOutput.copyTo(bitmap);
renderScript.destroy();

return bitmap;
```

```
}

private static Bitmap RGB565toARGB888(Bitmap img) throws Exception {
int numPixels = img.getWidth() * img.getHeight();
int[] pixels = new int[numPixels];

//Get JPEG pixels.  Each int is the color values for one pixel.
img.getPixels(pixels, 0, img.getWidth(), 0, 0, img.getWidth(), img.getHeight());

//Create a Bitmap of the appropriate format.
Bitmap result = Bitmap.createBitmap(img.getWidth(), img.getHeight(), Bitmap.Config.ARGB_8888);

//Set RGB pixels.
result.setPixels(pixels, 0, result.getWidth(), 0, 0, result.getWidth(), result.getHeight());
return result;
}
}
```

SubCategory

```
package com.example.collegify;

import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.GridLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.SearchView;
import android.widget.Toast;

import com.example.collegify.Adapters.RecyclerViewAdapter;
import com.example.collegify.Adapters.SubCategoryAdapter;
import com.example.collegify.Lists.ClgCategories;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
```

```
import java.util.List;

import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

/*
* Created by Hitanshu on 01-03-2019.
*/

public class SubCategory extends AppCompatActivity {

private Toolbar toolbar;
private DrawerLayout mDrawerlayout;
private ActionBarDrawerToggle mToggle;
List<ClgCategories> subCategoryList;
private ApiInterface apiInterface;

private RecyclerView recyclerView;
private RecyclerView.LayoutManager layoutManager;
private SubCategoryAdapter SubAdapter;
private NavigationView navigationView;



@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_sub_category);



toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Sub Categories");
mDrawerlayout=(DrawerLayout) findViewById(R.id.drawer);
mToggle = new ActionBarDrawerToggle(this, mDrawerlayout, R.string.open, R.string.close);
mDrawerlayout.addDrawerListener(mToggle);
mToggle.syncState();
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

subCategoryList = new ArrayList<>();
recyclerView = (RecyclerView) findViewById(R.id.recyclerview);
layoutManager = new GridLayoutManager(this, 1);
recyclerView.setLayoutManager(layoutManager);
recyclerView.setHasFixedSize(true);

String cat = getIntent().getStringExtra("category");
```

```
DatabaseReference mDatabase = FirebaseDatabase.getInstance().getReference("ClgSubCategories/" + cat);
mDatabase.addValueEventListener(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot)
{
for (DataSnapshot postSnapshot : dataSnapshot.getChildren())
 {
String name = postSnapshot.child("name").getValue(String.class);
String image = postSnapshot.child("image").getValue(String.class);
subCategoryList.add(new ClgCategories(name,image ));
}
SubAdapter = new SubCategoryAdapter(SubCategory.this,
 subCategoryList); //Call the constructor
recyclerView.setAdapter(SubAdapter);
}

@Override
public void onCancelled(@NonNull DatabaseError databaseError)
 {
String e = databaseError.getMessage();
Toast.makeText(getApplicationContext(), e,
Toast.LENGTH_SHORT).show();
}
});
navigationView = (NavigationView) findViewById(R.id.navgview);

navigationView.setNavigationItemSelectedListener(new NavigationView.OnNavigationItemSelectedListener()
 {
@Override
public boolean onNavigationItemSelected
(@NonNull MenuItem menuItem)
 {
switch (menuItem.getItemId()) {
case R.id.Home:
Intent HomeIntent =
new Intent(getApplicationContext(), ProfileActivity.class);
startActivity(HomeIntent);
finish();
break;
case R.id.Bookmarks:
Intent BookmarkIntent =
new Intent(getApplicationContext(), BookmarksActivity.class);
startActivity(BookmarkIntent);
finish();
break;
case R.id.gmaps:
Intent GoogleMapsIntent =
new Intent(getApplicationContext(), GoogleMapsActivity.class);
startActivity(GoogleMapsIntent);
```

```
break;
case R.id.ChangePassword:
Intent ChgPassIntent =
 new Intent(getApplicationContext(), ChangePasswordActivity.class);
startActivity(ChgPassIntent);
break;
case R.id.ChangeSecurityQuestions:
Intent ChgSecQuest =
new Intent(getApplicationContext(),
ChangeSecurityQuestionActivity.class);
startActivity(ChgSecQuest);
break;
case R.id.Feedback:
Intent FeedbackIntent = new Intent(getApplicationContext(),FeedbackActivity.class);
startActivity(FeedbackIntent);
break;
case R.id.Logout:
FirebaseAuth.getInstance().signOut();
finish();
startActivity(new Intent(getApplicationContext(), MainActivity.class));

}
return false;
}
});

}

@Override
public boolean onCreateOptionsMenu(Menu menu){
final MenuInflater inflater = getMenuInflater();
inflater.inflate(R.menu.search_menu, menu);
MenuItem item = menu.findItem(R.id.search);
SearchView searchView = (SearchView) item.getActionView();

searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener()
 {
@Override
public boolean onQueryTextSubmit(String query) {
String user_input = query.trim();
Intent intent = new Intent(getApplicationContext(),
FilterActivity.class);
/*Bundle bundle = new Bundle();
bundle.putParcelableArrayList("AllColleges",allCollegeList );
bundle.putParcelableArrayList("AllCourses", allCourseList);
intent.putExtras(bundle);*/
intent.putExtra("userinput",user_input);
startActivity(intent);
```

```
return false;
}

@Override
public boolean onQueryTextChange(String newText) {
return false;
}
});

return super.onCreateOptionsMenu(menu);
}
// So that the 3 line in toolbar work
@Override
public boolean onOptionsItemSelected(MenuItem item) {
if(mToggle.onOptionsItemSelected(item)){
return true;
}
return super.onOptionsItemSelected(item);

}
}
```

VerifyPhone

```
package com.example.collegify;

import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.android.gms.tasks.TaskExecutors;
import com.google.firebase.FirebaseException;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.PhoneAuthCredential;
import com.google.firebase.auth.PhoneAuthProvider;

import java.util.concurrent.TimeUnit;

public class VerifyPhone extends AppCompatActivity {

private String verificationId;
```

```
private FirebaseAuth mAuth;
private ProgressBar progressBar;
private EditText editText;

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_verify_phone);

mAuth = FirebaseAuth.getInstance();
progressBar = findViewById(R.id.progressbar);
editText = findViewById(R.id.editTextCode);
String phoneNumber = getIntent().getStringExtra("phonenumber");

sendVerificationCode(phoneNumber);

findViewById(R.id.buttonSignIn).setOnClickListener
(new View.OnClickListener() {
@Override
public void onClick(View v) {

//Manual
String code = editText.getText().toString().trim();

if(code.isEmpty() || code.length()<6){
editText.setError("Enter Code...");
editText.requestFocus();
return;
}


verifyCode(code);
}
});
}

private void verifyCode(String code){
PhoneAuthCredential credential
 = PhoneAuthProvider.getCredential(verificationId, code);
signInWithCredential(credential);
}

private void signInWithCredential
(PhoneAuthCredential credential) {
mAuth.signInWithCredential(credential).addOnCompleteListener
(new OnCompleteListener<AuthResult>() {
@Override
public void onComplete
(@NonNull Task<AuthResult> task) {
```

```
progressBar.setVisibility(View.GONE);

if(task.isSuccessful()){
Intent startIntent =
new Intent(getApplicationContext(),
ProfileActivity.class);
startIntent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK);
startActivity(startIntent);
finish();
Toast.makeText(VerifyPhone.this, "Login Successfull!!",Toast.LENGTH_LONG ).show();

}else {
Toast.makeText(VerifyPhone.this,
task.getException().getMessage(),
Toast.LENGTH_LONG ).show();
}
}
});
}

private void sendVerificationCode(String number) {
progressBar.setVisibility(View.VISIBLE);
PhoneAuthProvider.getInstance().verifyPhoneNumber(
number,
60,
TimeUnit.SECONDS,
TaskExecutors.MAIN_THREAD,
mCallBack
);
}

private PhoneAuthProvider.OnVerificationStateChangedCallbacks
mCallBack = new PhoneAuthProvider.OnVerificationStateChangedCallbacks()
{

@Override
public void onCodeSent(String s, PhoneAuthProvider.ForceResendingToken forceResendingToken) {
super.onCodeSent(s, forceResendingToken);

verificationId = s;

}

@Override
public void onVerificationCompleted
(PhoneAuthCredential phoneAuthCredential) {

String code = phoneAuthCredential.getSmsCode();
if(code != null){
```

```
editText.setText(code);
verifyCode(code);
}

}

@Override
public void onVerificationFailed(FirebaseException e) {
Toast.makeText(VerifyPhone.this, e.getMessage(),
Toast.LENGTH_LONG).show();
}


};

}
```

VerifySQ

```
package com.example.collegify;

import android.annotation.SuppressLint;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.drawable.BitmapDrawable;
import android.renderscript.Allocation;
import android.renderscript.Element;
import android.renderscript.RenderScript;
import android.renderscript.ScriptIntrinsicBlur;
import android.support.annotation.NonNull;
import android.support.v4.app.NavUtils;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
```

```
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

public class VerifySQ extends AppCompatActivity {

private String email;
private TextView security_question;
private EditText ans;
private Button submit;


private FirebaseAuth firebaseAuth;

final DatabaseReference mDatabase = FirebaseDatabase.getInstance().
getReference("Users");
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_verify_sq);

//To Blur the Background image
Bitmap bitmap = BitmapFactory.decodeResource(this.getResources(),R.drawable.loginbg);
//Drawable to bitmap
Bitmap blurredBitmap = blurRenderScript(this,bitmap, 10);
//second parametre is radius
findViewById(R.id.verifySQpg).setBackground(new BitmapDrawable(getResources(), blurredBitmap));
//Assign the bitmap as background


Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
getSupportActionBar().setTitle("Forgot Password");
toolbar.setSubtitle("Password Recovery");
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

security_question = (TextView)findViewById(R.id.asksq);
ans = (EditText) findViewById(R.id.answer);
email = getIntent().getStringExtra("email");

firebaseAuth = FirebaseAuth.getInstance();

mDatabase.addListenerForSingleValueEvent(new ValueEventListener(){
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
for(DataSnapshot postSnapshot: dataSnapshot.getChildren()){
if(email.equals(postSnapshot.child("email").
getValue(String.class)))
{security_question.setText(postSnapshot.child("question").
getValue(String.class));
```

```
submit = (Button) findViewById(R.id.submit);
submit.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
changePass();
}
});
break;
}else {
Toast.makeText(VerifySQ.this,"User doesn't exist" ,
Toast.LENGTH_LONG ).show();
}
}

}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
Toast.makeText(getApplicationContext(),databaseError.getMessage() ,
Toast.LENGTH_LONG ).show();
}
});


}
private void changePass(){
final String answer = ans.getText().toString();
if (TextUtils.isEmpty(answer)) {
//answer is empty
ans.setError("Please enter the answer");
ans.requestFocus();
//stopping the function execution further
return;
}
mDatabase.addListenerForSingleValueEvent
(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot)
 {
for(DataSnapshot postSnapshot: dataSnapshot.getChildren())
{if(email.equals(postSnapshot.child("email").
getValue(String.class))){
if(answer.equals(postSnapshot.child("answer").
getValue(String.class))){
firebaseAuth.sendPasswordResetEmail(email)
.addOnCompleteListener(new OnCompleteListener<Void>()
{
@Override
```

```
public void onComplete(@NonNull Task<Void> task) {

if(task.isSuccessful()){
Toast.makeText(getApplicationContext(), "Reset link sent to your email", Toast.LENGTH_LONG).show();
Intent intent = new Intent(getApplicationContext(),MainActivity.class);
startActivity(intent);
finish();
} else {
Toast.makeText(getApplicationContext(),
task.getException().getMessage(), Toast.LENGTH_LONG).show();

}
}
});
}
break;
}
}

}

@Override
public void onCancelled(@NonNull DatabaseError databaseError) {
Toast.makeText(getApplicationContext(),databaseError.getMessage() ,
Toast.LENGTH_LONG ).show();
}
});
}
// For blurring Images
@SuppressLint("NewApi")
public static Bitmap blurRenderScript
(Context context, Bitmap smallBitmap, int radius) {
try {
smallBitmap = RGB565toARGB888(smallBitmap);
} catch (Exception e) {
e.printStackTrace();
}

Bitmap bitmap = Bitmap.createBitmap(
smallBitmap.getWidth(), smallBitmap.getHeight(),
Bitmap.Config.ARGB_8888);

RenderScript renderScript = RenderScript.create(context);

Allocation blurInput = Allocation.createFromBitmap
(renderScript, smallBitmap);
Allocation blurOutput = Allocation.createFromBitmap
(renderScript, bitmap);
```

```
ScriptIntrinsicBlur blur = ScriptIntrinsicBlur.create(renderScript,
Element.U8_4(renderScript));
blur.setInput(blurInput);
blur.setRadius(radius); // radius must be 0 < r <= 25
blur.forEach(blurOutput);

blurOutput.copyTo(bitmap);
renderScript.destroy();

return bitmap;

}

private static Bitmap RGB565toARGB888(Bitmap img) throws Exception {
int numPixels = img.getWidth() * img.getHeight();
int[] pixels = new int[numPixels];

//Get JPEG pixels.  Each int is the color values for one pixel.
img.getPixels(pixels, 0, img.getWidth(), 0, 0, img.getWidth(),
img.getHeight());

//Create a Bitmap of the appropriate format.
Bitmap result = Bitmap.createBitmap(img.getWidth(), img.getHeight(), Bitmap.Config.ARGB_8888);

//Set RGB pixels.
result.setPixels(pixels, 0, result.getWidth(), 0, 0, result.getWidth(), result.getHeight());
return result;
}
@Override
public boolean onOptionsItemSelected(MenuItem item) {
switch (item.getItemId()) {
// Respond to the action bar's Up/Home button
case android.R.id.home:
NavUtils.navigateUpFromSameTask(this);
return true;
}
return super.onOptionsItemSelected(item);
}
}
```

AllClgList

```
package com.example.collegify.Lists;
/*
* Created by Hitanshu on 21-03-2019.
*/

import android.os.Parcel;
import android.os.Parcelable;
```

```
import com.google.gson.annotations.SerializedName;

public class AllClgList implements Parcelable{
@SerializedName("name")
public String name;

@SerializedName("name1")
private String name1;

@SerializedName("image")
private String image;

public AllClgList(String name, String name1, String image){
this.name = name;
this.name1 = name1;
this.image = image;
}
AllClgList()
{}

protected AllClgList(Parcel in) {
name = in.readString();
name1 = in.readString();
image = in.readString();
}

public static final Creator<AllClgList> CREATOR
 = new Creator<AllClgList>() {
@Override
public AllClgList createFromParcel(Parcel in) {
return new AllClgList(in);
}

@Override
public AllClgList[] newArray(int size) {
return new AllClgList[size];
}
};

public String getName() {
return name;
}

public String getName1() {
return name1;
}

public String getImage() {
return image;
```

```
}

@Override
public int describeContents() {
return 0;
}

@Override
public void writeToParcel(Parcel dest, int flags) {
dest.writeString(name);
dest.writeString(name1);
dest.writeString(image);
}
}
```

ClgCategories

```
package com.example.collegify.Lists;

import android.net.Uri;

import com.google.gson.annotations.SerializedName;

/*
* Created by Hitanshu on 01-03-2019.
*/

public class ClgCategories {

@SerializedName("name")
private String ClgCatName;

@SerializedName("image")
private String ClgCatBg;


//Constructor

public ClgCategories(String clgCatName, String clgCatBg) {
ClgCatName = clgCatName;
ClgCatBg = clgCatBg;
}


//Getters

public String getClgCatName() {
return ClgCatName;
}
```

```
public String getClgCatBg() {
return ClgCatBg;
}

//Setters

public void setClgCatName(String clgCatName) {
ClgCatName = clgCatName;
}

public void setClgCatBg(String clgCatBg) {
ClgCatBg = clgCatBg;
}

}
```

CollegeInfoList

```
package com.example.collegify.Lists;
/*
* Created by Hitanshu on 15-03-2019.
*/

import com.google.gson.annotations.SerializedName;

import java.util.ArrayList;
import java.util.List;

public class CollegeInfoList {

@SerializedName("name")
private String ClgName;

@SerializedName("addr")
private String ClgAddr;

@SerializedName("link")
private String website;

@SerializedName("lat")
private double latitude;

@SerializedName("long")
private double longitude;

@SerializedName("contact")
private String phonecall;

@SerializedName("course")
private List<String> courses;
```

```
public String getClgName() {
return ClgName;
}

public String getClgAddr() {
return ClgAddr;
}

public String getWebsite() {
return website;
}

public double getLatitude() {
return latitude;
}

public double getLongitude() {
return longitude;
}

public List<String> getCourses() {
return courses;
}
}
```

CollegeList

```
package com.example.collegify.Lists;
/*
* Created by Hitanshu on 28-03-2019.
*/

import com.google.gson.annotations.SerializedName;

public class CollegeList {

@SerializedName("name")
private String ClgCatName;

public CollegeList(String clgCatName) {
ClgCatName = clgCatName;
}

public String getClgCatName() {
return ClgCatName;
}
}
```

Common

```
package com.example.collegify.Lists;
/*
* Created by Hitanshu on 28-03-2019.
*/

import com.example.collegify.Lists.AllClgList;

import java.util.ArrayList;

public class common {

public static ArrayList<AllClgList> allCollegeList;
public static ArrayList<AllClgList> allCourseList;
public static String username;
public static String e_mail;
}
```

DataParser

```
package com.example.collegify.GoogleMaps;
/*
* Created by Hitanshu on 11-04-2019.
*/

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;

public class DataParser {

private HashMap<String,String>getSingleNearbyPlace(JSONObject googlePlaceJSON)
{

HashMap<String,String> googlePlaceMap = new HashMap<>();
String NameOfPlace = "-NA-";
String vicinity = "-NA-";
String latitude = "";
String longitude = "";
String reference = "";

try {

//Fetching the data
if(!googlePlaceJSON.isNull("name")){
NameOfPlace = googlePlaceJSON.getString("name");
}
```

```
if(!googlePlaceJSON.isNull("vicinity")){
vicinity = googlePlaceJSON.getString("vicinity");
}

latitude = googlePlaceJSON.getJSONObject("geometry").getJSONObject("location").
getString("lat");
longitude = googlePlaceJSON.getJSONObject("geometry").getJSONObject("location").
getString("lng");
reference = googlePlaceJSON.getString("reference");

//Storing the data in Hashmap
googlePlaceMap.put("place_name",NameOfPlace );
googlePlaceMap.put("vicinity",vicinity );
googlePlaceMap.put("lat",latitude );
googlePlaceMap.put("lng",longitude );
googlePlaceMap.put("reference",reference );


} catch (JSONException e) {
e.printStackTrace();
}

return googlePlaceMap;
}

private List<HashMap<String,String>>getAllNearbyPlaces(JSONArray jsonArray)
{

int counter = jsonArray.length();
List<HashMap<String,String>> NearbyPlacesList=new ArrayList<>();

HashMap<String,String> NearbyPlaceMap = null;

for (int i = 0; i < counter; i++){
try {

NearbyPlaceMap = getSingleNearbyPlace
((JSONObject) jsonArray.get(i));
NearbyPlacesList.add(NearbyPlaceMap);

} catch (JSONException e) {
e.printStackTrace();
}
}
return NearbyPlacesList;
}

public List<HashMap<String,String>> parse(String JSONdata){
```

```
JSONArray jsonArray = null;
JSONObject jsonObject;

try {

jsonObject = new JSONObject(JSONdata);
jsonArray = jsonObject.getJSONArray("results");

} catch (JSONException e) {
e.printStackTrace();
}

return getAllNearbyPlaces(jsonArray);

}
}
```

DownloadUrl

```
package com.example.collegify.GoogleMaps;
/*
* Created by Hitanshu on 11-04-2019.
*/

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.MalformedURLException;
import java.net.URL;

public class DownloadUrl {

public String ReadTheUrl(String placeURL) throws IOException{

String Data = "";
InputStream inputStream = null;
HttpURLConnection httpURLConnection = null;

try {
URL url = new URL(placeURL);
httpURLConnection = (HttpURLConnection) url.openConnection();
httpURLConnection.connect();

inputStream = httpURLConnection.getInputStream();
BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(inputStream));
StringBuffer stringBuffer = new StringBuffer();
```

```
String line = "";

while ((line = bufferedReader.readLine()) != null){

stringBuffer.append(line);

}

Data = stringBuffer.toString();
bufferedReader.close();

} catch (MalformedURLException e) {
e.printStackTrace();
} catch (IOException e) {
e.printStackTrace();
} finally {
inputStream.close();
httpURLConnection.disconnect();
}

return Data;
}

}
```

GetNearbyPlaces

```
package com.example.collegify.GoogleMaps;
/*
* Created by Hitanshu on 11-04-2019.
*/

import android.os.AsyncTask;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

import java.io.IOException;
import java.util.HashMap;
import java.util.List;

public class GetNearbyPlaces extends AsyncTask<Object,String,String>{

private String googleplaceData,url;
private GoogleMap mMap;

@Override
```

```
protected String doInBackground(Object... objects) {

mMap = (GoogleMap) objects[0];
url = (String) objects[1];

DownloadUrl downloadUrl = new DownloadUrl();
try {
googleplaceData = downloadUrl.ReadTheUrl(url);
} catch (IOException e) {
e.printStackTrace();
}
return googleplaceData;
}

@Override
protected void onPostExecute(String s) {

List<HashMap<String,String>> nearByPlacesList = null;
DataParser dataParser = new DataParser();
nearByPlacesList = dataParser.parse(s);

DisplayNearbyPlaces(nearByPlacesList);

}

private void DisplayNearbyPlaces(List<HashMap<String,String>>
 nearByPlacesList){

for (int i = 0; i < nearByPlacesList.size(); i++){

MarkerOptions markerOptions = new MarkerOptions();

HashMap<String,String> googleNearbyPlace = nearByPlacesList.get(i);
String nameOfPlace = googleNearbyPlace.get("place_name");
String vicinity = googleNearbyPlace.get("vicinity");
double lat = Double.parseDouble(googleNearbyPlace.get("lat"));
double lng = Double.parseDouble(googleNearbyPlace.get("lng"));

LatLng latLng = new LatLng(lat,lng);
markerOptions.position(latLng);
markerOptions.title(nameOfPlace + " : " + vicinity);
markerOptions.icon(BitmapDescriptorFactory.defaultMarker
(BitmapDescriptorFactory.HUE_BLUE));

mMap.addMarker(markerOptions);
mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
mMap.animateCamera(CameraUpdateFactory.zoomTo(10));
}
}
```

```
}
```

CollegeListAdapter

```
package com.example.collegify.Adapters;
/*
* Created by Hitanshu on 15-03-2019.
*/

import android.content.Context;
import android.content.Intent;
import android.os.StrictMode;
import android.support.annotation.NonNull;
import android.support.v7.widget.CardView;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import com.example.collegify.CollegeActivity;
import com.example.collegify.Lists.CollegeList;
import com.example.collegify.R;

import java.util.List;

public class CollegeListAdapter extends RecyclerView.Adapter<CollegeListAdapter.MyViewHolder> {

private Context myContext;
private List<CollegeList> mData;

//Constructors
public CollegeListAdapter(Context myContext, List<CollegeList> mData){
this.myContext = myContext;
this.mData = mData;
}
@NonNull
@Override
public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int i){

int SDK_INT = android.os.Build.VERSION.SDK_INT;
if (SDK_INT > 8)
{
StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder()
.permitAll().build();
StrictMode.setThreadPolicy(policy);
}
View view;
LayoutInflater mInflater = LayoutInflater.from(parent.getContext());
```

```
view = mInflater.inflate(R.layout.cardview_item, parent,false);
return new CollegeListAdapter.MyViewHolder(view);

}

@Override
public void onBindViewHolder(@NonNull
 final CollegeListAdapter.MyViewHolder myViewHolder, int i){

myViewHolder.category.setText(mData.get(i).getClgCatName());
//Picasso.get().load(mData.get(i).getClgCatBg()).into(myViewHolder.catbg);
myViewHolder.catbg.setAlpha(190);
myViewHolder.cardView.setOnClickListener(new View.OnClickListener(){
@Override
public void onClick(View v) {
final String ClgName = myViewHolder.category.getText().toString();
Intent intent = new Intent(myContext, CollegeActivity.class);
intent.putExtra("collegenm", ClgName);
myContext.startActivity(intent);

}
});

}

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

public static class MyViewHolder extends RecyclerView.ViewHolder{

TextView category;
ImageView catbg;
CardView cardView;

public MyViewHolder(View itemView){
super(itemView);

category = (TextView) itemView.findViewById(R.id.category);
catbg = (ImageView) itemView.findViewById(R.id.categorybg);
cardView = (CardView) itemView.findViewById(R.id.cardview);
}

}
}
```

CourseRVAdapter

```
package com.example.collegify.Adapters;
/*
* Created by Hitanshu on 01-04-2019.
*/

import android.os.StrictMode;
import android.support.annotation.NonNull;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import com.example.collegify.R;

import java.util.List;

public class CourseRVAdapter extends RecyclerView.Adapter
<CourseRVAdapter.MyViewHolder> {

private List<String> mData;

public CourseRVAdapter(List<String> mData){
this.mData = mData;
}

@NonNull
@Override
public MyViewHolder onCreateViewHolder
(@NonNull ViewGroup parent, int i) {
int SDK_INT = android.os.Build.VERSION.SDK_INT;
if (SDK_INT > 8)
{
StrictMode.ThreadPolicy policy
= new StrictMode.ThreadPolicy.Builder()
.permitAll().build();
StrictMode.setThreadPolicy(policy);
}
View view;
LayoutInflater mInflater=LayoutInflater.from(parent.getContext());
view = mInflater.inflate(R.layout.course_item, parent,false);
return new CourseRVAdapter.MyViewHolder(view);
}

@Override
public void onBindViewHolder(@NonNull MyViewHolder myViewHolder, int i){
myViewHolder.crname.setText(mData.get(i));

}
```

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

public static class MyViewHolder extends RecyclerView.ViewHolder{

TextView crname;


public MyViewHolder(View itemView){
super(itemView);

crname = (TextView) itemView.findViewById(R.id.cname);

}

}
}
```

CustomSwipeAdapter

```
package com.example.collegify.Adapters;
/*
* Created by Hitanshu on 28-03-2019.
*/

import android.content.Context;
import android.support.annotation.NonNull;
import android.support.v4.view.PagerAdapter;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.RelativeLayout;

import com.example.collegify.R;
import com.squareup.picasso.Picasso;

import java.util.List;

public class CustomSwipeAdapter extends PagerAdapter{

private List<String> images;
private Context ctx;
private LayoutInflater layoutInflater;

public CustomSwipeAdapter(Context ctx,List<String> images ){
```

```
this.images = images;
this.ctx = ctx;
}
@Override
public int getCount() {

return images.size();
}

@Override
public boolean isViewFromObject
(@NonNull View view, @NonNull Object o){
return (view==(RelativeLayout)o);
}

@NonNull
@Override
public Object instantiateItem
(@NonNull ViewGroup container, int position) {
layoutInflater=(LayoutInflater)ctx.getSystemService
(Context.LAYOUT_INFLATER_SERVICE);
View item_view=layoutInflater.inflate
(R.layout.swipe,container,false);
ImageView imageView=(ImageView) item_view.findViewById(R.id.image_view);
Picasso.get().load(images.get(position)).into(imageView);
container.addView(item_view);
return item_view;
}

@Override
public void destroyItem(@NonNull ViewGroup container,
int position, @NonNull Object object) {

container.removeView((RelativeLayout)object);

}
}
```

RecyclerViewAdapter

```
package com.example.collegify.Adapters;

import android.content.Context;
import android.content.Intent;
import android.os.StrictMode;
import android.support.annotation.NonNull;
import android.support.v7.widget.CardView;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
```

```
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import com.example.collegify.ApiInterface;
import com.example.collegify.Lists.ClgCategories;
import com.example.collegify.ProfileActivity;
import com.example.collegify.R;
import com.example.collegify.SubCategory;
import com.squareup.picasso.Picasso;

import java.util.List;

/*
* Created by Hitanshu on 26-02-2019.
*/

public class RecyclerViewAdapter extends RecyclerView.Adapter<RecyclerViewAdapter.MyViewHolder>{

private Context myContext;
private List<ClgCategories> mData;
private ApiInterface apiInterface;
public static final String TAG = ProfileActivity.class.getSimpleName();


//Constructor
public RecyclerViewAdapter(Context myContext,
List<ClgCategories> mData){
this.myContext = myContext;
this.mData = mData;
}

@NonNull
@Override
public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent,
int i){

int SDK_INT = android.os.Build.VERSION.SDK_INT;
if (SDK_INT > 8)
{
StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder()
.permitAll().build();
StrictMode.setThreadPolicy(policy);
}
View view;
LayoutInflater mInflater = LayoutInflater.from(parent.getContext());
view = mInflater.inflate(R.layout.category_item, parent,false);
return new MyViewHolder(view);
```

```
}

@Override
public void onBindViewHolder(@NonNull final MyViewHolder myViewHolder,
final int i) {


myViewHolder.category.setText(mData.get(i).getClgCatName());
Picasso.get().load(mData.get(i).getClgCatBg()).into(myViewHolder.catbg);
myViewHolder.catbg.setAlpha(190);

myViewHolder.cardView.setOnClickListener(new View.OnClickListener(){
@Override
public void onClick(View v) {

final String cat = (String)mData.get(i).getClgCatName();

Intent intent = new Intent(myContext, SubCategory.class);
intent.putExtra("category", cat);
myContext.startActivity(intent);

}
});

//TO blur the image
/*
Bitmap bitmap = ((BitmapDrawable)myViewHolder.catbg.getDrawable()).
getBitmap();//ImageView to bitmap
Bitmap blurred = blurRenderScript(myContext,bit, 25);
//second parameter is radius
myViewHolder.catbg.setImageBitmap(blurred);
//Assign the bitmap to imageview
*/
}

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

public static class MyViewHolder extends RecyclerView.ViewHolder{

TextView category;
ImageView catbg;
CardView cardView;

public MyViewHolder(View itemView){
super(itemView);
```

```
category = (TextView) itemView.findViewById(R.id.category);
catbg = (ImageView) itemView.findViewById(R.id.categorybg);
cardView = (CardView) itemView.findViewById(R.id.cardview);

}

}

}
```

SubCategoryAdapter

```
package com.example.collegify.Adapters;

import android.content.Context;
import android.content.Intent;
import android.os.StrictMode;
import android.support.annotation.NonNull;
import android.support.v7.widget.CardView;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.collegify.Lists.ClgCategories;
import com.example.collegify.CollegeListActivity;
import com.example.collegify.R;
import com.squareup.picasso.Picasso;

import java.util.List;

public class SubCategoryAdapter extends RecyclerView.Adapter<SubCategoryAdapter.MyViewHolder>{

private Context myContext;
private List<ClgCategories> mData;


//Constructor
public SubCategoryAdapter(Context myContext, List<ClgCategories> mData){
this.myContext = myContext;
this.mData = mData;
}

@NonNull
@Override
public SubCategoryAdapter.MyViewHolder onCreateViewHolder(@NonNull
ViewGroup parent, int i) {
```

```
int SDK_INT = android.os.Build.VERSION.SDK_INT;
if (SDK_INT > 8)
{
StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder()
.permitAll().build();
StrictMode.setThreadPolicy(policy);
}
View view;
LayoutInflater mInflater = LayoutInflater.from(parent.getContext());
view = mInflater.inflate(R.layout.cardview_item, parent,false);
return new SubCategoryAdapter.MyViewHolder(view);

}

@Override
public void onBindViewHolder(@NonNull final SubCategoryAdapter.MyViewHolder myViewHolder, final int i){


myViewHolder.category.setText(mData.get(i).getClgCatName());
Picasso.get().load(mData.get(i).getClgCatBg()).into(myViewHolder.catbg);
myViewHolder.catbg.setAlpha(190);

myViewHolder.cardView.setOnClickListener(new View.OnClickListener(){
@Override
public void onClick(View v) {
final String subcat = myViewHolder.category.getText().toString();
Intent intent = new Intent(myContext, CollegeListActivity.class);
intent.putExtra("subcategory", subcat.trim());
myContext.startActivity(intent);
Toast.makeText(myContext, subcat, Toast.LENGTH_LONG).show();
}
});

}

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

public static class MyViewHolder extends RecyclerView.ViewHolder{

TextView category;
ImageView catbg;
CardView cardView;

public MyViewHolder(View itemView){
super(itemView);
```

```
category = (TextView) itemView.findViewById(R.id.category);
catbg = (ImageView) itemView.findViewById(R.id.categorybg);
cardView = (CardView) itemView.findViewById(R.id.cardview);
}

}

}
```

8 APPLICATIONS

1. The application will provide information about the colleges in Mumbai. It will give different information such as the courses available, their duration, fee structure, cut off list of last year, etc.
2. The application will help the user to clarify his/her doubts. The app will provide contact number through which the student of parents will be able to contact the college for enquiry.
3. This application will reduce the overhead of visiting multiple website and display information in more organized manner.
4. This application will also be very useful to people who shifted from other states to Mumbai.

9 FEATURES

1. Search easily for colleges by college name and courses. Also nearby colleges wil be displayed ny detecting user's location and requested college location will be given,
2. View menus, pictures, phone numbers, directions, reviews, and other information you need to find the best college near you.
3. Rate and give feedback the colleges you've liked.
4. Maintain an easy reference list of your favourites or colleges you want to visit with bookmarks.
5. Use map view to explore colleges around you.

10 LIMITATIONS

Like every system has its own limitation, Collegify also has some limitations as follows

1. This application is only available in a particular area i.e. Mumbai, it won't provide information about the colleges outside Mumbai.
2. The application won't be providing a way to apply for admission, the user will have to follow the government defined procedure to get admissions.

11 RESULT AND CONCLUSION

11.1 Results

In this chapter, the results of our project are shown. The screens are as follows: -

Figure 6: SignUp Screen

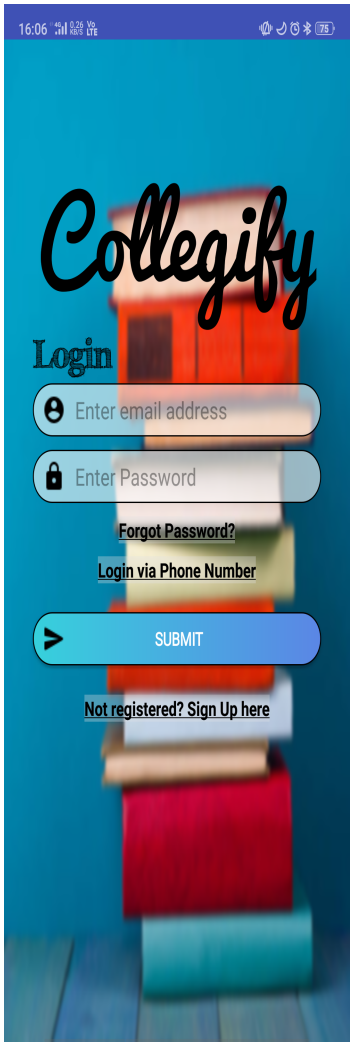


Figure 7: Login Screen

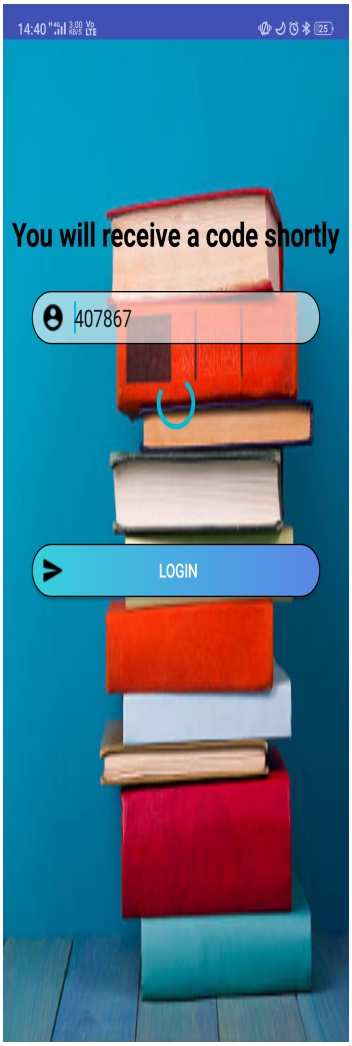


Figure 8: Otp Screen

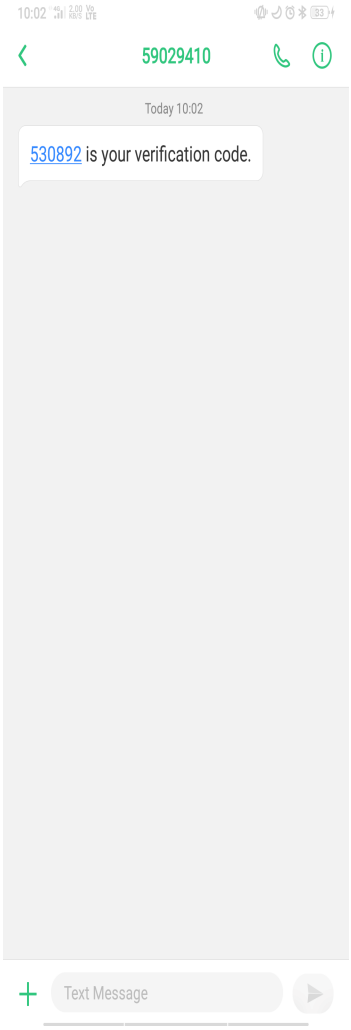


Figure 9: Otp

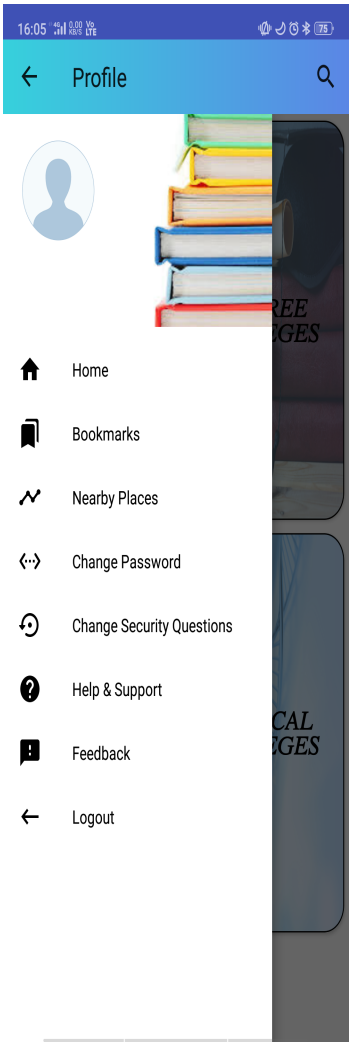


Figure 10: Profile Screen

Figure 11: Home Screen

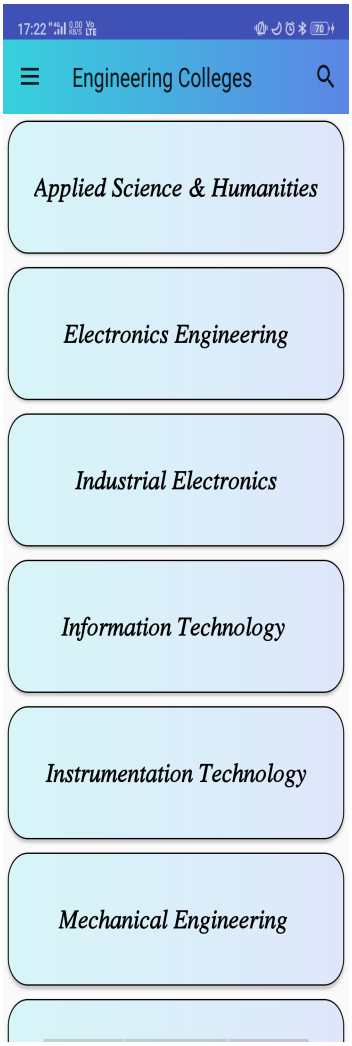


Figure 12: Engineering Courses

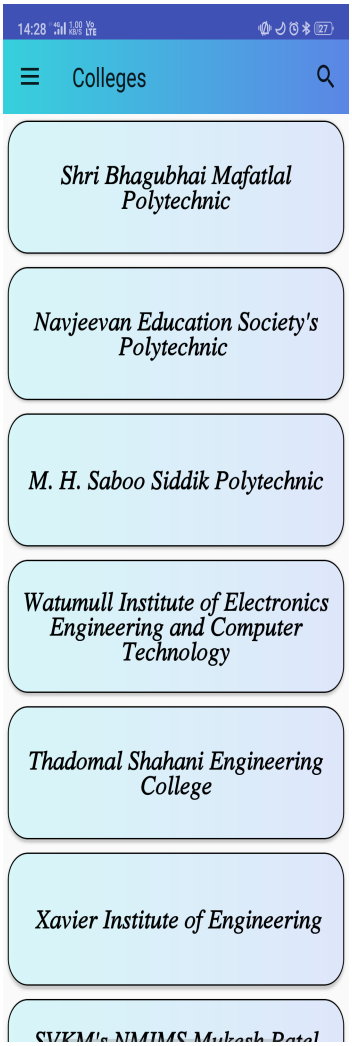


Figure 13: Engineering College List

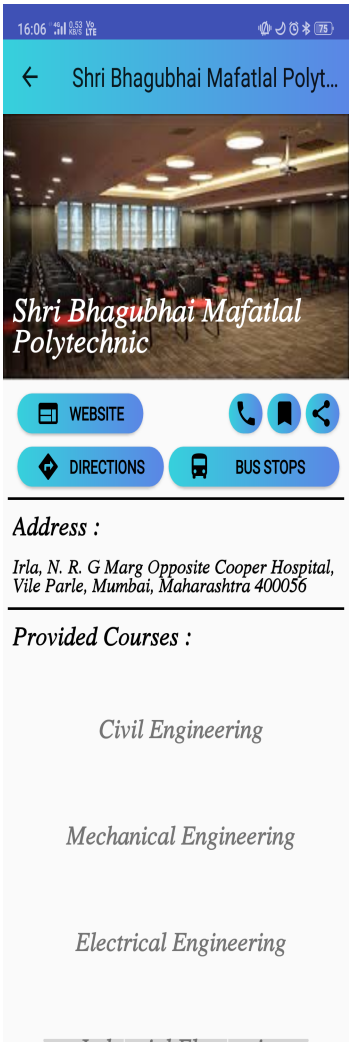


Figure 14: College Details

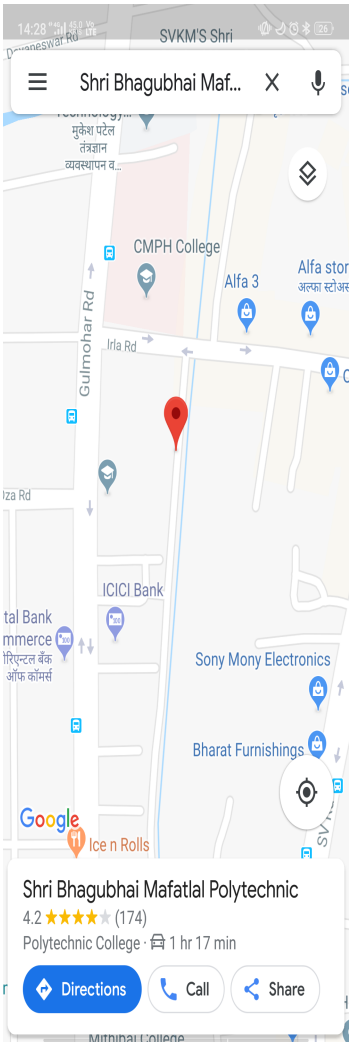


Figure 15: Directions

Figure 16: Bus Stops

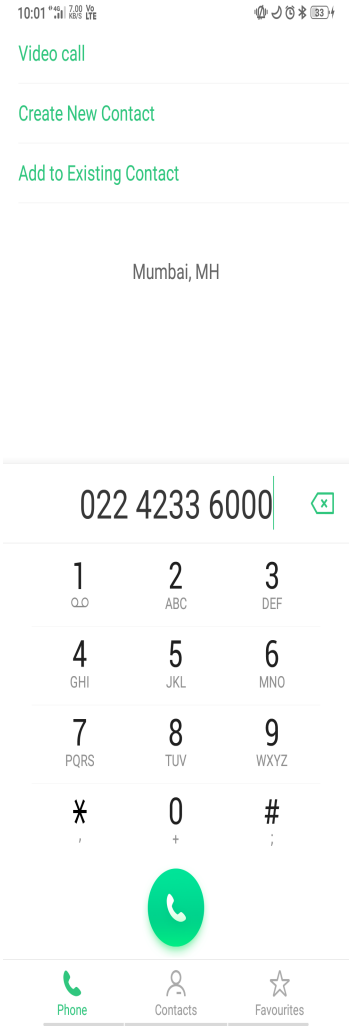


Figure 17: Calling the College

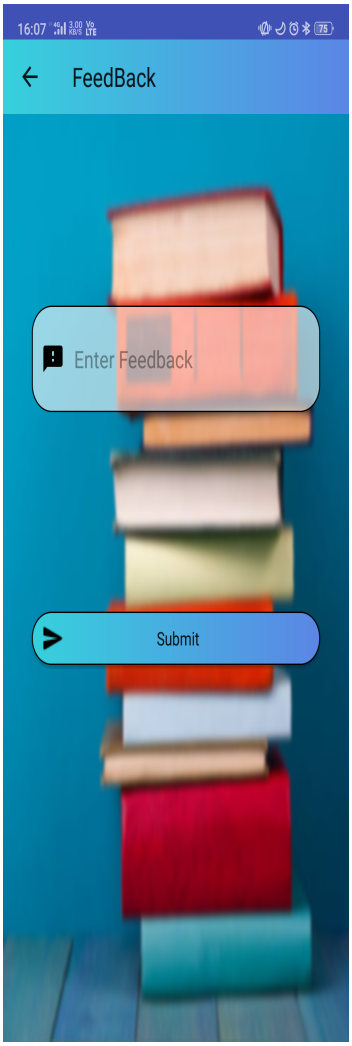


Figure 18: Feedback Screen

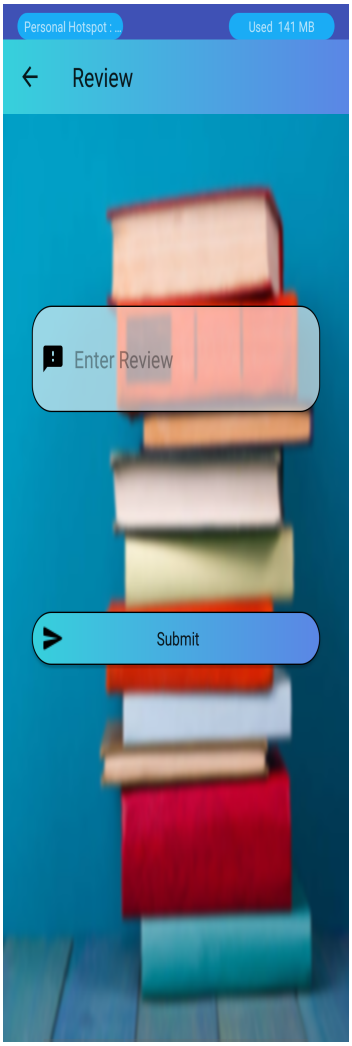


Figure 19: Review Screen

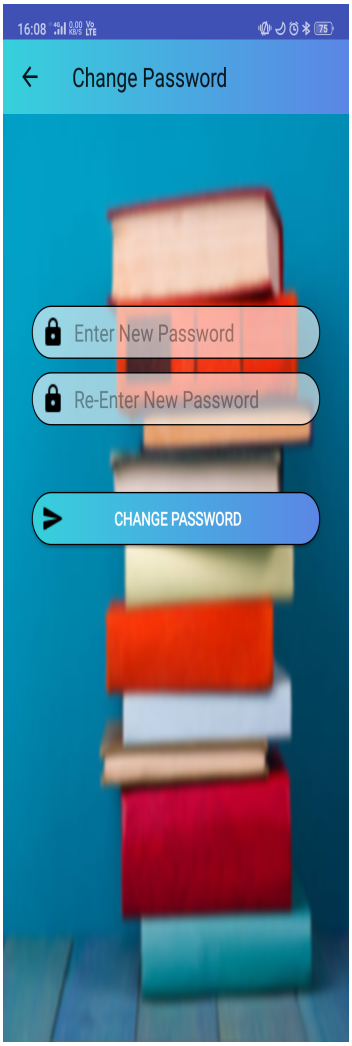


Figure 20: Change Password

Figure 21: Change Security Question Screen

11.2 Future Scope

1. Collaboration with college for admission process, with an agreement with the colleges admission forms can be posted on the application and we can also integrate the process followed for admission on our application. Also, from each college a administrator which will handle the admin module for any modification regarding college informations.

2. Improving coverage area, this application will work specifically in Mumbai city it is not providing any service in other parts of our country, with the increase in popularity and demand we can extend Collegify in other parts of India also.

3. Incorporating Chatbot will prove to be user-friendly after the integration of admission process.

11.3 Conclusion

This is our first attempt in developing a mobile application which gave us a basic understanding of development and challenges of mobile application development. We also understood how to work in a team to fufil the user's needs and how project reports can be done by using LaTeX .We also understood how to use some GitHub Libraries which are developed by the developers from all over the world and to use them in our project. The main aim of the project is to provide an easy to use application for searching the nearby colleges as well as providing the details of the colleges. The application has been implemented and tested on real devices. The application can be further extended to iOS devices from Android by using some cross platform technologies such as React-Native, Flutter, etc. The application can also extended by collaborating with colleges for admission process with the increase in popularity and demand we can extend Collegify in other parts of India also.

12 REFERENCES

[1]. Rajinder Singh, Department of Computer Science and Applications DCSA Panjab University SSGRC Hoshiarpur, An Overview of Android Operating System and Its Security Features, dated February 2014.
https://pdfs.semanticscholar.org/11f4/b8efd1a9af746f17ac5e8d6a789bd3c3a9b7.pdf

[2]. GeekyAnts, Introduction to Firebase, blog post published on December 28, 2017.
https://hackernoon.com/introduction-to-firebase-218a23186cd7

[3]. Stefan Otte, Introduction to Version Control System, Computer Systems and Telematics Institute of Computer Science Freie Universitat Berlin, Germany.
https://pdfs.semanticscholar.org/5093/c4b55ad419379ad22e0c60008f133a2b6e3b.pdf

[4]. Authors: Przemyslaw Gilski, Jacek Stefanski,Publication Name: TEM Journal,Publication Date: 2015.
https://www.academia.edu/11344334/Android_OS_A_Review

[5]. The entire Pro Git book, written by Scott Chacon and Ben Straub and published by Apress.
https://git-scm.com/book/en/v2

[6].The offical firebase documentation provided by Google Developers.
https://firebase.google.com/docs/android/setup

[7]. Google Maps For Business. By Dibyajyoti Pal is a Lead Business Analyst for Manufacturing Practice and is currently assigned for Warranty-on-Demand product at Tavant Technologies.
https://www.tavant.com/sites/default/files/download-center/Tavant_Manufacturing_White_Paper_Google_Maps_Business_Final.pdf

[8]. Hana R. Esmaeel, International Journal of Advanced Research in Computer Science and Software Engineering, Dated – May 2015.
http://ijarcsse.com/Before_August_2017/docs/papers/Volume_5/5_May2015/V5I4-0764.pdf