1.1 CCET-IPD APP

1.1 Objective :

Help participants to know about the CCET's Innovative product design event and also to register for the event.

1.2 About the App:

For an event to organize well, participants are must and for that being this app is developed. CCET college yearly organizes an event namely Innovative product design i.e. IPD and for that purpose, this app named, **CCET-IPD** provides the interface to the user for both registration and rules for the event. This app beautifully present about the event, instructions for the event and people behind this event to the user. The user can register himself for the event also by giving all the necessary details and project related docs

1.3 Tools used:

Languages used:

Java, Android, XML

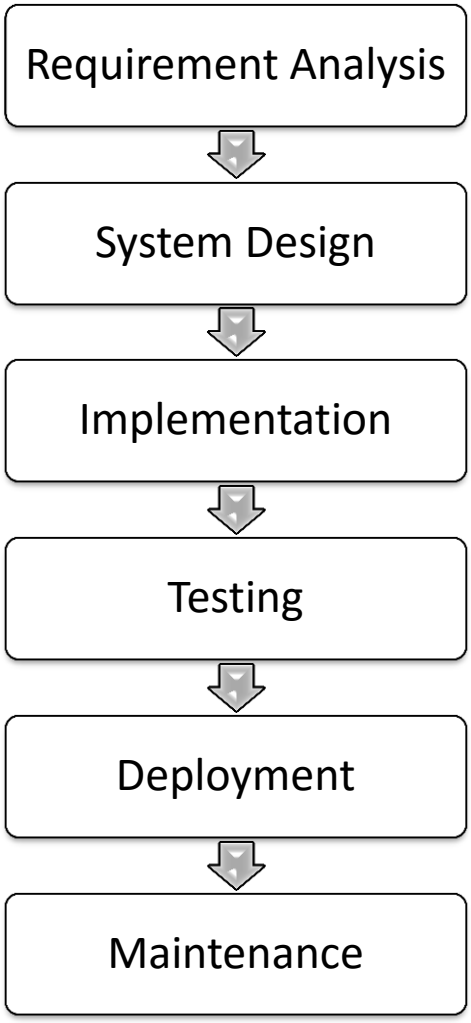
Platform used :

Android Studio

Database by:

Firebase

1.4 Waterfall Model

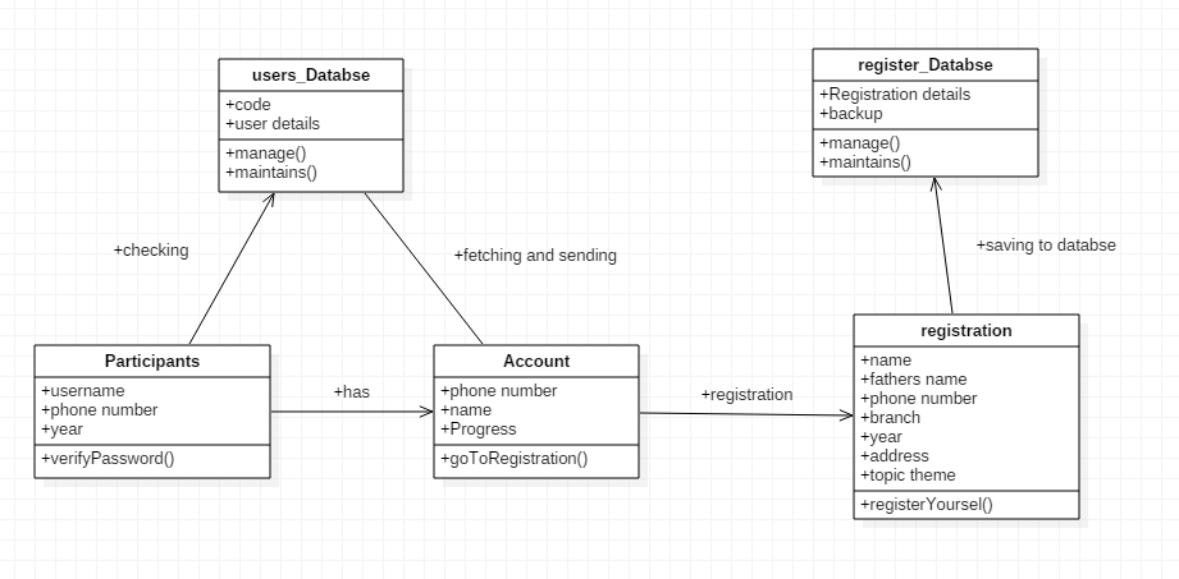


1.4.1 Requirement Analysis:

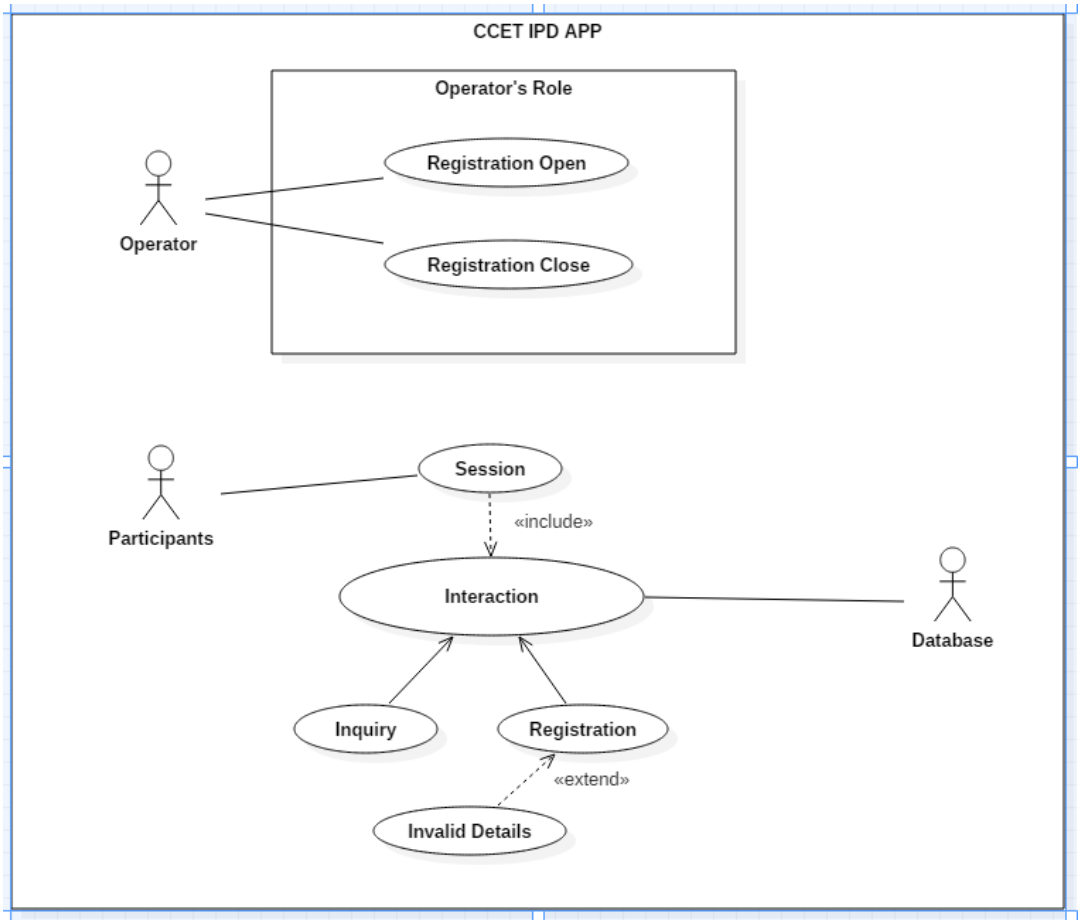
Student detail's attributes, About event, instructions, rules and regulations, data flow diagram and SRS.

1.4.2 System Design:

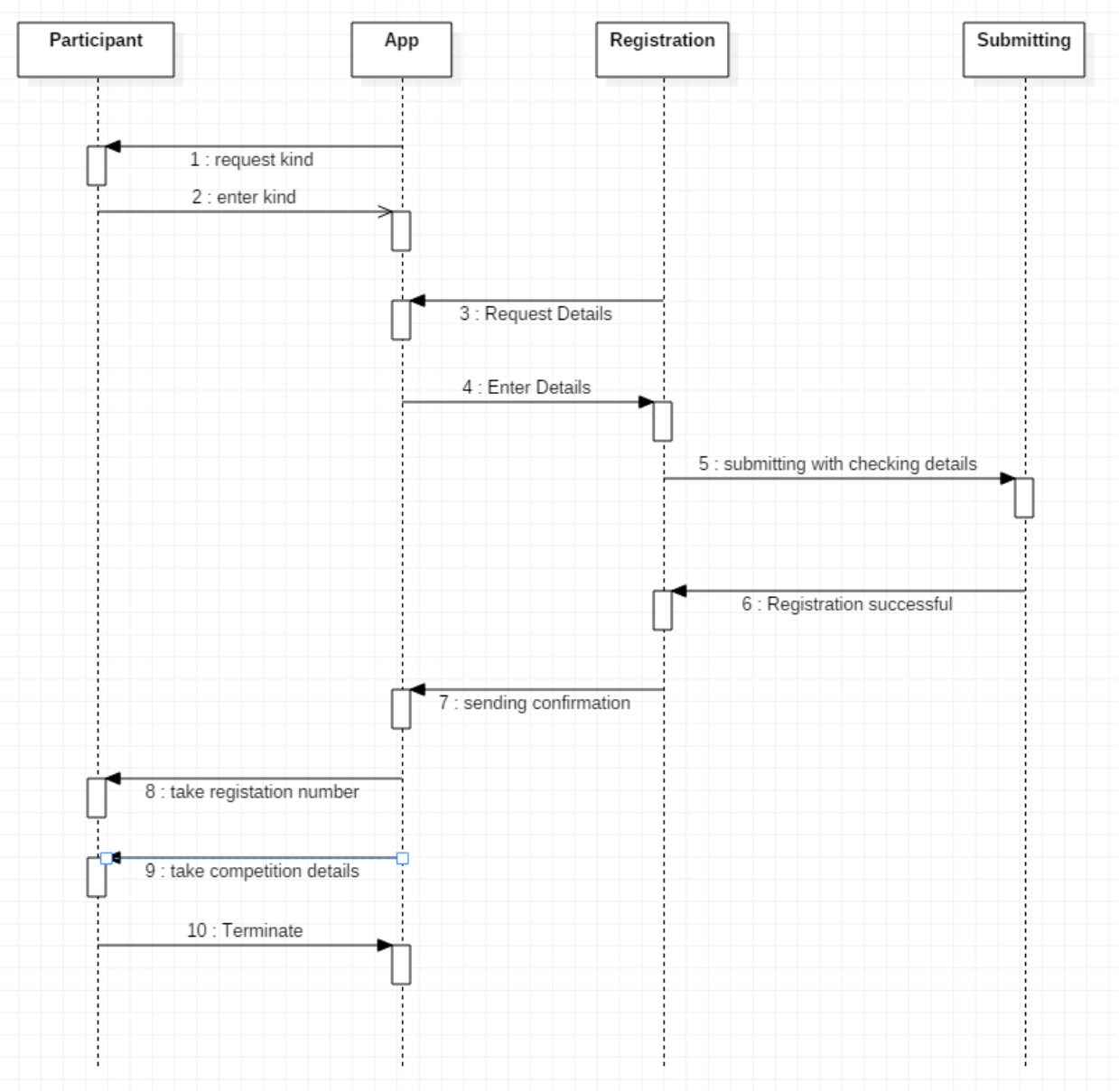
1.4.2.1 Class diagram



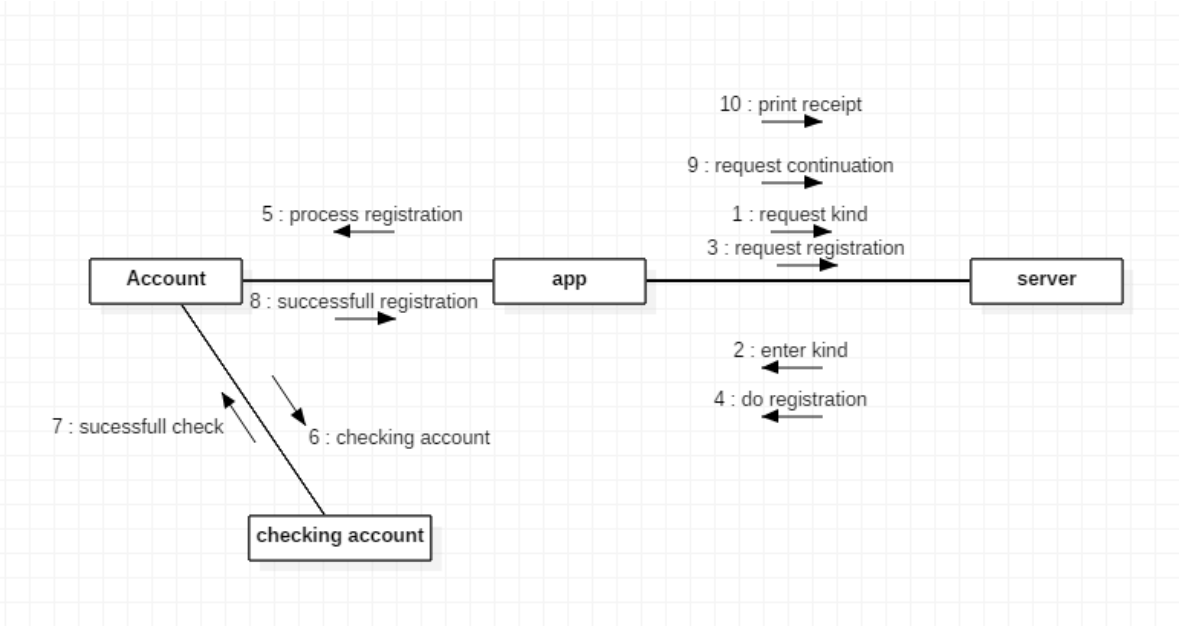
1.4.2.2 Use case diagram



1.4.2.3 Sequence Diagram



1.4.2.4 Data-Flow diagram



1.4.3 Implementation

Frontend : android, java and XML.

Backend: java, Firebase

1.4.4 Testing

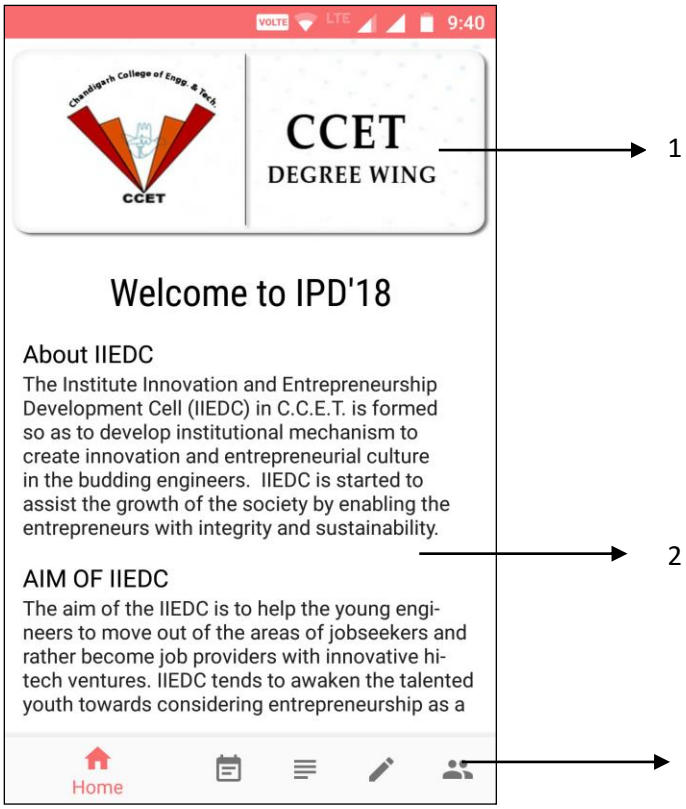
Alpha testing : testing done by developer itself i.e registering and working of an app.

Beta testing: successfull registration of 10 groups simultaneously. Running app in different devices.

1.4.5 Deployment: Once app get accepted then it will be shared to different colleges using google play store.

2. LAYOUTS AND WORK-FLOW OF CCET-IPD APP

2.1 HOME



1. ImageView

   Displays image resources, for example Bitmap or Drawable resources. ImageView is also commonly used to apply tints to an image and handle image scaling.

```
<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@mipmap/ic_launcher"
    />
```

2. TextView

A user interface element that displays text to the user. To provide user-editable text.

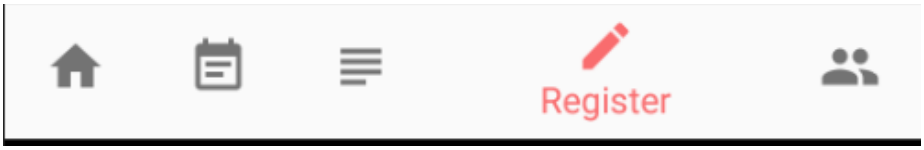
```
<TextView
    android:id="@+id/text_view_id"
    android:layout_height="wrap_content"
    android:layout_width="wrap_content"
    android:text="@string/hello" />
```

3. Bottom Navigation

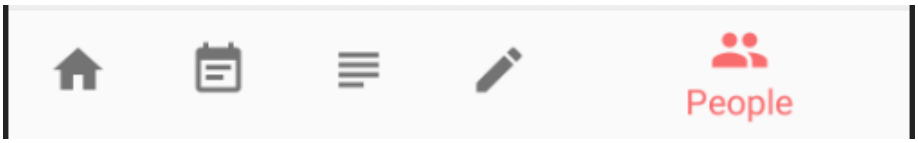
Home- Open the home page in current fragment.

Event - calender icon, which explains everything about the event.

Instructions - Rules and instructions for the event.



Register – Page for the users to register themselves.



People – People behind this event.

2.2 Event

This page give the glipse of the event and time left for the event.

1.

2.

3.

1. CountDown :

   Its not in built, but developed using ImageView and TextView. It tells the time left for the event. It is not static in app but can be controlled from the database and can be set manually.

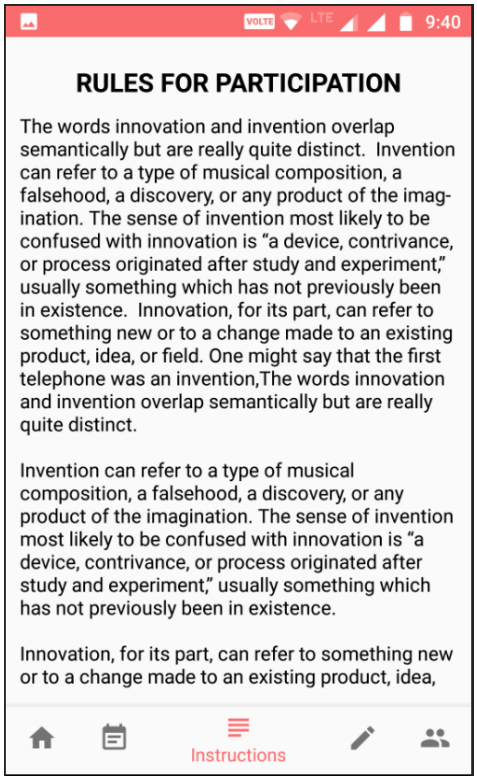
2. Gallery (big Image) :

   It shows the previous glimpse happen in that event. Using ImageView, these images are presented in the app. You can select Images from the bottom ribbon which has buch of images in it, just click the image you want to enlarge.

3. Image Ribbon :

   It has buch of images as in our phone's gallery, just tap on the image to see it in the big picture.

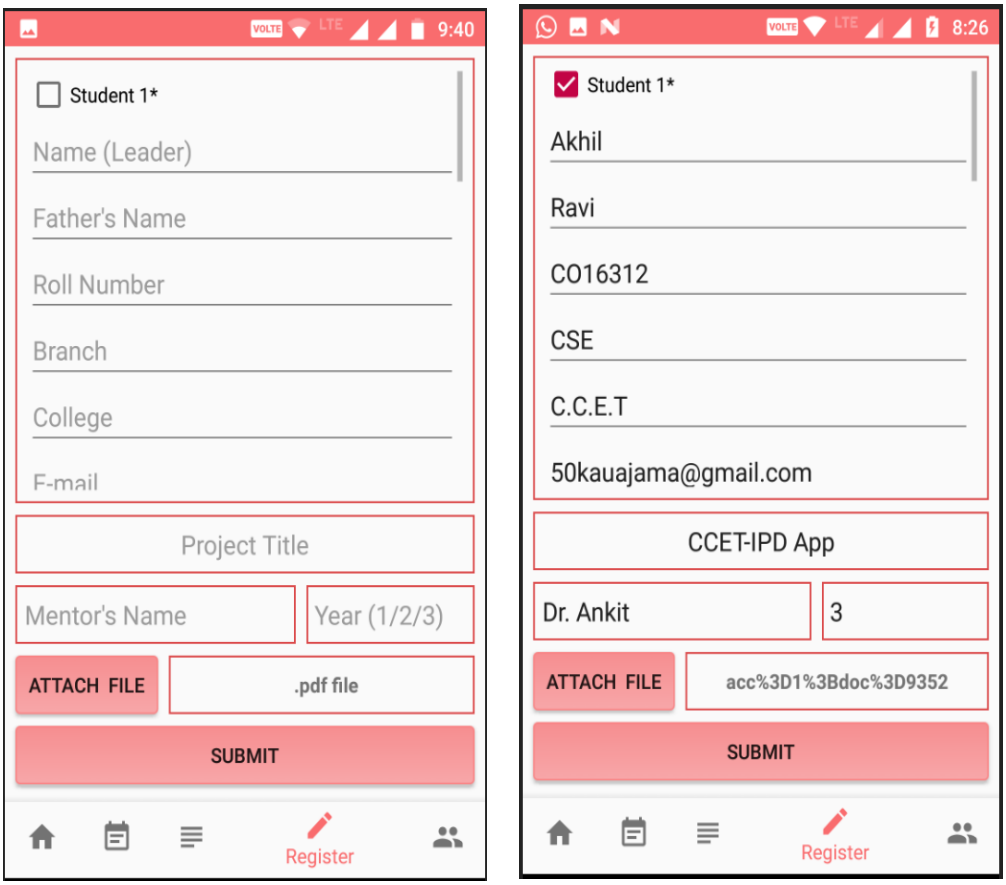
2.3 Instructions



It has the rules, regulations and intructions for the event. That how the event will conduct, selection process, location, themes, how to register etc.

2.4 REGISTER

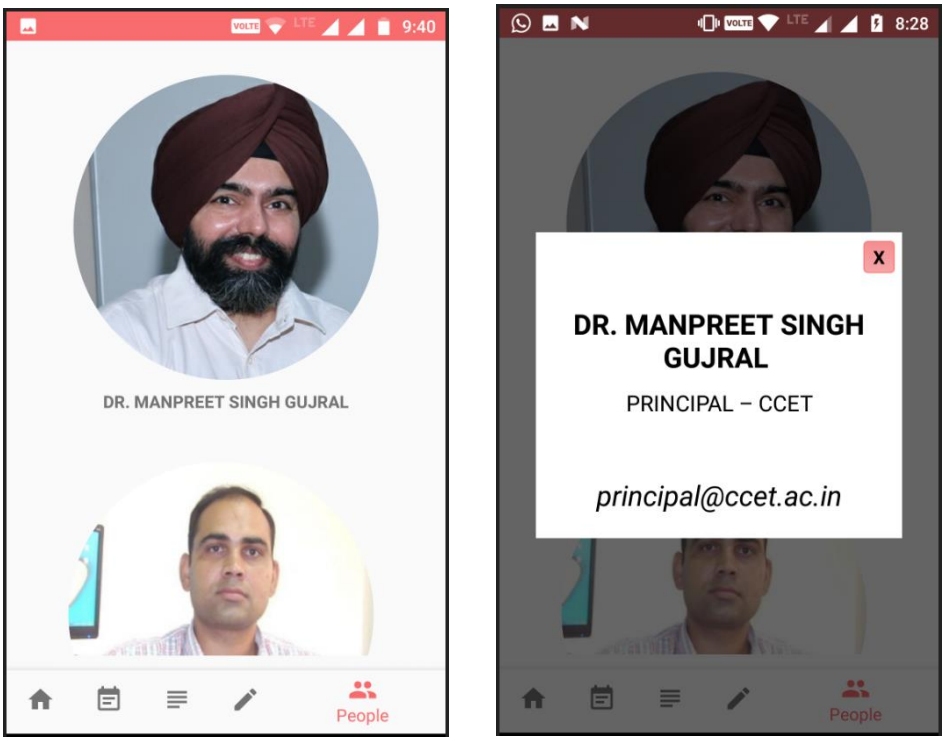
Every thing is explained in the **INSTRUCTIONS** page for this form i.e how and what to fill in this along with size and format of doc.



1. Student first check the **checkbox** then It ask for his/her details required by the officials to get them register like name, fagher's name, college etc.
2. Then the team need to give the name of their project in **Project Title** box.
3. Then name of mentor and year.
4. Then they are required to attach file of their project. Click the **ATTACH FILE** button to do so.
5. Once everything's done, click the **SUBMIT** button.

All this data is send to server and stored in a schema as explained in the later chapter.

2.5 PEOPLE



This page tells us the main people behind this event. It has round **BITMAP** images showing the picture of the Sir/Faculty dedicated for this event. To know more about them just tap on the image a **DIALOG BOX** will appear on the same page showing more details about them.

3. SIDE NAVIGATION DRAWER

The navigation drawer is a UI panel that shows your app's main navigation menu. It is hidden when not in use, but appears when the user swipes a finger from the left edge of the screen or, when at the top level of the app, the user touches the drawer icon in the app bar.

This page shows you how to implement a navigation drawer using the DrawerLayout APIs available in the Support Library.

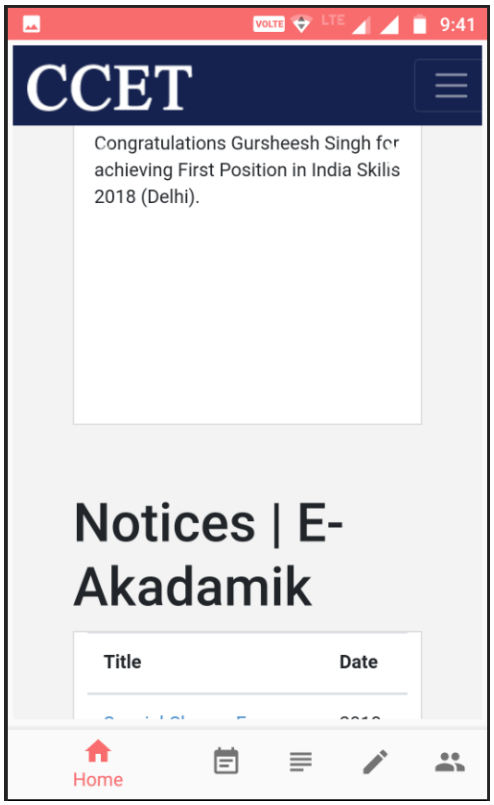
This page uses APIs from on the Android Support Library, so you need to add the following dependencies to your apps module's build.gradle file:

```
dependencies {
  implementation 'com.android.support:appcompat-v7:28.0.0'
  implementation 'com.android.support:design:28.0.0'
}
```

Options in side navigation drawer:

I. Webiste
II. Reach Us
III. Contact Us
IV. Developer

3.1 WEBISTE



In app web page of CCET college is opened in the app itself. This page made students to know more about the college and faculty.

This site is displayed using **WebView**, WebView objects allow you to display web content as part of your activity layout, but lack some of the features of fully-developed browsers. A WebView is useful when you need increased control over the UI and advanced configuration options that will allow you to embed web pages in a specially-designed environment for your app.

XML code snipet:

```
<WebView
        android:id="@+id/activity_main_webview"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
```

Java code snipet:

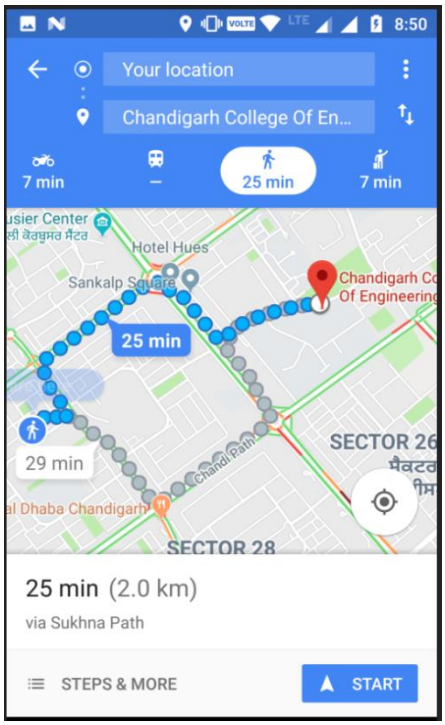
```
//to conncet xml layout with java
mWebView = (WebView) findViewById(R.id.activity_main_webview);
```

Manifest:

```
// as website use Internet to load so we have to give permission to our app to
// access internet. Give permission in the manifest file

<uses-permission android:name="android.permission.INTERNET" />
```

3.2 REACH US



This page uses **GOOGLE MAPS**, helps registered students to get the location of the college. App requests users to acces their location, hence using this location they track the path between their location and college. This feature doesn't open in the app itself, app itself opens google maps. User doesnt need to enter any info. App alone do it for users. It by default fills the your location and destination.

Java code snipet:

```
Uri gmmIntentUri = Uri.parse("geo:37.7749,-122.4194");
Intent mapIntent = new Intent(Intent.ACTION_VIEW, gmmIntentUri);
mapIntent.setPackage("com.google.android.apps.maps");
if (mapIntent.resolveActivity(getPackageManager()) != null) {
    startActivity(mapIntent);
}
```

3.3 CONTACT US

It open the contact us fragment in the app itself. It is just of user having problems in registering or insufficient information for the event then he/she can contact using these particulars.

3.4 DEVELOPER

Finally the last page of this app, tells users about the developer. It has the name and contact of the developer just incase if user is having problem in using the app or incase if filled incorrect details etc.

It has **ImageView** and **TextView** for displaying image and text in this page.

Java code snipet :

```
private Bitmap bmp;
private ImageView img;
img = (ImageView)findViewById(R.id.imageView1);
BitmapDrawable  abmp = (BitmapDrawable)img.getDrawable();
bmp = abmp.getBitmap();
```

4. BACKEND WITH FIREBASE

4.1 About Firebase:

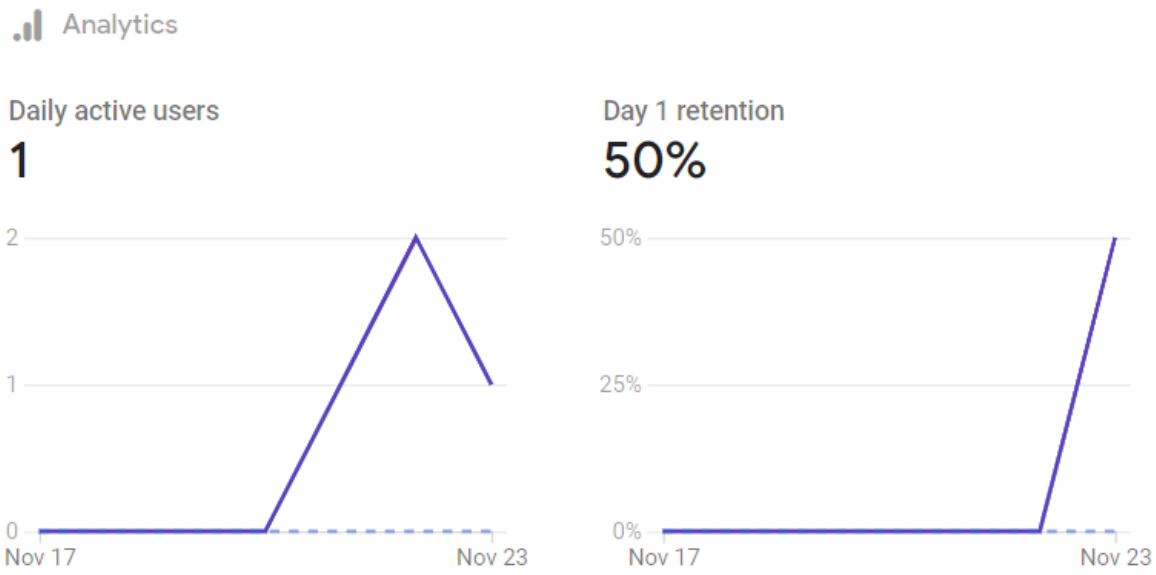
Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014. As of October 2018, the Firebase platform has 18 products which are used by 1.5 million apps.

4.2 Services:

4.2.1 Analytics

Firebase Analytics

Firebase Analytics is a cost-free app measurement solution that provides insight into app usage and user engagement.



4.2.2 Develop

Firebase Cloud Messaging

Formerly known as Google Cloud Messaging (GCM), Firebase Cloud Messaging (FCM) is a cross-platform solution for messages and notifications for Android, iOS, and web applications, which as of 2016 can be used at no cost.

Realtime Database

Firebase provides a realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud. The company provides client libraries that enable integration with Android, iOS, JavaScript, Java, Objective-C, Swift and Node.js applications.

Firebase Storage

Firebase Storage provides secure file uploads and downloads for Firebase apps, regardless of network quality. The developer can use it to store images, audio, video, or other user-generated content. Firebase Storage is backed by Google Cloud Storage.

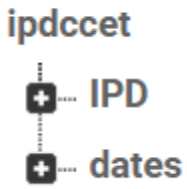
Firebase Hosting

Firebase Hosting is a static and dynamic web hosting service that launched on May 13, 2014. It supports hosting static files such as CSS, HTML, JavaScript and other files, as well as support through Cloud Functions.

5. SCHEMA USED IN APP

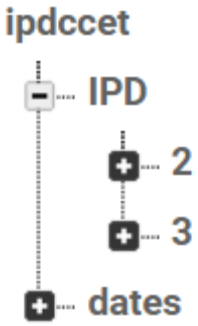
5.1 DATABASE

Parent Directory:



1. **IPD:**

   It is the main directory for the storage, It stores the whole data filled by the students in the registration page.

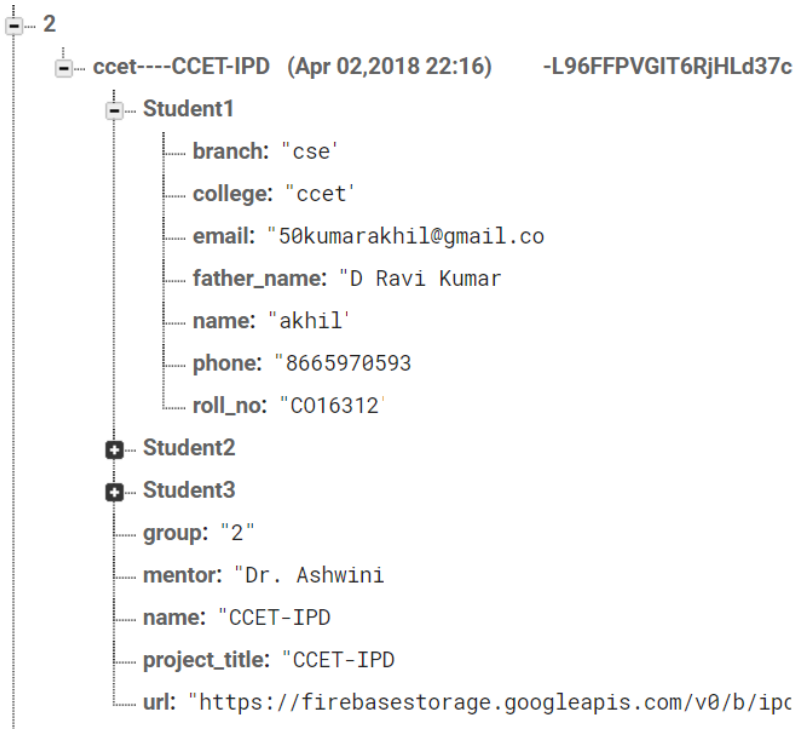


It has further two directories which divide students into their reoective semesters.

In semester directory, the student data is stored as:

College_name----Project_name (date of submission) -registraion_ID

This stored as the name of the directory



It has every data which was filled by the student during registration.

The **url object** has the location of the doc submitted by the student.

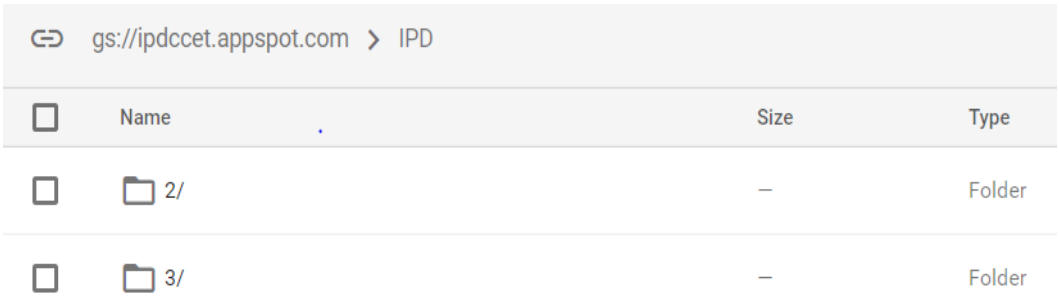
5.2 STORAGE

It stores the doc of the students, It is the real time databse which get updated automatically by the servers of the FIREBASE.

Directory :

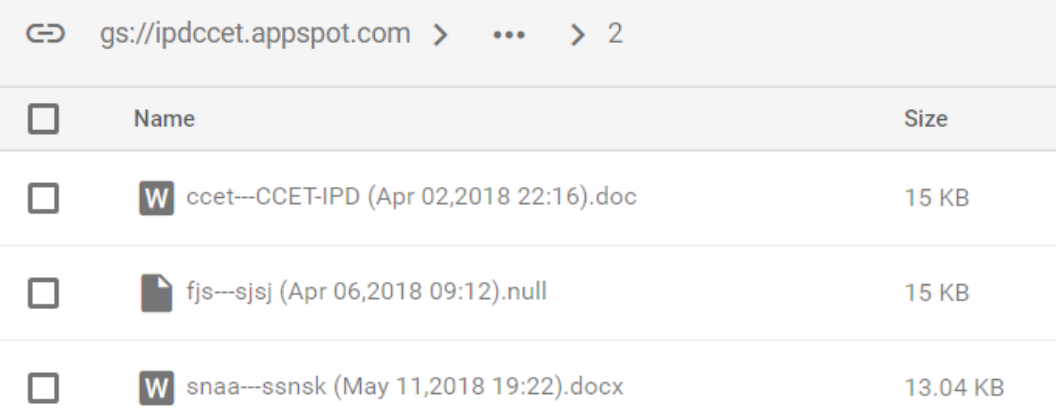
The main root folder is IPD

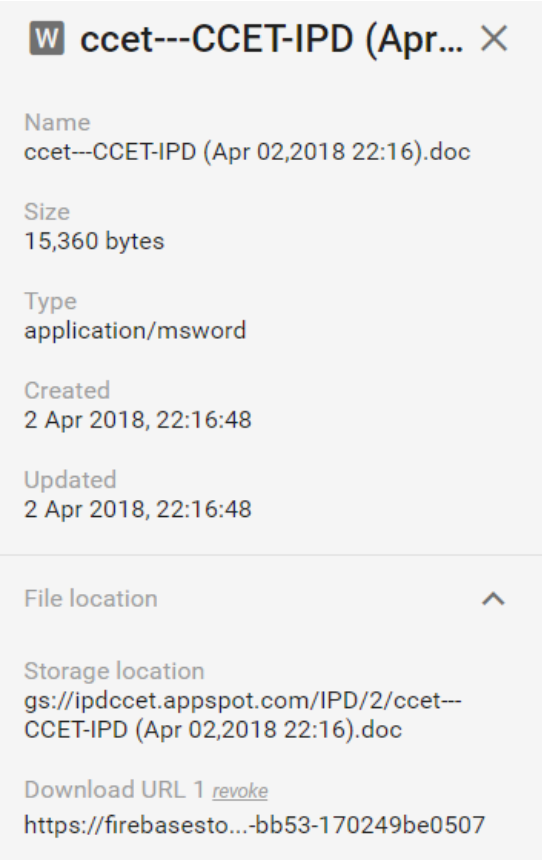
Inside this folder it has semester's folder, which seperates students docs semester wise



Here in semester 2, has all the docs of students of semester 2.

File name : college_name----project_name (date and time of submission).doc





Firebase servers give good details about the doc i.e. doc size, type, date of creation, date when updated, storage collection and download url.Just click the url and download start automatically.