# STUDENT TASK MANAGEMENT APP

**NAVNEET RANJAN**

**17SCSE121006**

***in partial fulfillment for the award of the degree of***

**BACHELOR OF COMPUTER APPLICATION**

GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

IN

COMPUTER SCIENCE

Under the Supervision of

**Mr. Ravindra Kumar Chahar**

**Associate Professor**

GALGOTIAS UNIVERSITY

## SCHOOL OF COMPUTING AND SCIENCE AND ENGINEERING

## BONAFIDE CERTIFICATE

Certified that this project report **"…STUDENT TASK MANAGEMENT APP……..…..."** is the bonafide work of "**……..NAVNEET RANJAN……"** who carried out the project work under my supervision.

**SIGNATURE OF HEAD**

Dr. MUNISH SHABARWAL,
PhD (Management), PhD (CS)
Professor & Dean,
School of Computing Science & Engineering

# Project Title: Student Task Management App


Project Supervisor(Guide):

Mr. Ravindra Kumar Chahar

Associate Professor, SCSE

Submitted By:

Navneet Ranjan

17SCSE121006

## INTRODUCTION

Nowadays, technology is increasingly used by human being in every field. As people move from one place to another, many wireless technologies are available to remain in contact with others, without regard of the location. The increasing popularity of Smart Phones has drawn the attention of almost everybody. Along with making and receiving calls, users can send and receive messages, access the Internet, digital media, incorporate audio/video recording etc. Smart Phones also contain built-in keyboard, high resolution camera, front side camera for video conferencing, touch screen etc. Different smart phones have different operating systems. A mobile app, short for mobile application or just app is an application which runs on smart phones, tablet or mobile phones. Apps are pre installed or downloadable pieces of software that can do almost everything. Apps make mobile more like portable computers having multi core processors, gigabytes of memory and a real operating system. Originally mobile apps are made available for informational purposes that include Gmail, calendar, weather information etc. With increase in technology and user demands, developers started to make apps for other purposes like games, banking, video chats etc. An app can show the data in a similar way as a website, along with other benefits to download the content that can be used offline, in case the Internet is not available. There are many apps available in market today for different Operating Systems i.e. Android, Blackberry and Apple etc., in which Android is having the maximum market share these

days. Fig. 1 depicts market share of different operating systems from 2011 to 2014.

There are three types of Mobile Apps: Native, Web-based and hybrid. Native Apps live on the device. These apps are available through an application store and are developed for a particular platform and fully utilize the features of that device. These Apps work in offline mode and can work on latest API's of that platform. Some different types of Native Apps include calendar, calculator Etc. While Web-based Apps are not real applications, instead these are websites that are designed to have look and feel as Native App. These require a browser as well as Data Connection to run. Users first access them through Web page and in that Web page they have the option of installing them on their device. Hybrid Apps are partly Native App and partly Web App. These Apps live on App store and take advantage of device features. Many users are interested in using Native Mobile Apps as they can be used even when there is no Internet connectivity.

First Android Mobile was T- Mobile G1, launched in United states which contains pull down notification window right from day one, having home screen widgets containing Gmail, Google market widget etc. According to a website, the number of apps available on Google play store (For Android Platform) right from Dec 2009(16000 apps) to February 2015(1,400, 000 apps) is depicted in Fig. 2 [2]. Various domains of Android Applications include Entertainment, educational purposes, retail, financial,

social, travel, education, healthcare etc.

## RELATED WORK

Encouraged by launch of cheaper Smart phones, students have started using Apps for educational purposes. Many Apps are available which provides course material, assignments for practice; text books Etc. for reading. Many Apps provide facility for discussion by chatting instead of going to class for solving their problems. Also students can choose to display information according to their choice. Different famous apps for education include Chemistry Helper, English Helper, Kids numbers, Math Lite, Mathway, High School Physics and Vocabla[3] etc.

Many Apps are available to help the students to prepare for competitive Exams. Most Commonly used Apps include UGC NET [4], in which there are modules for Syllabus of UGC-NET exam, Sample of Objective type questions, Reference books and downloading of previous year question papers etc. A similar App named Indian History [5] is available in which chapter wise tutorials are available for students to learn. They can play the quiz and about 20 questions are asked with four options for each question and student has to select one option and number of correct answers will be shown at the end.

One more App named Gate 2015 [6], is available having modules for Syllabus, Practice questions, previous question papers, references of subjects, guidelines Etc. The Limitation is that there is need to

include all the subjects in a hierarchical manner. A similar App named Net Prep+ [7], is available having modules i.e. timed test, Random Test, Chapter Wise Test. Limitation is that the number of questions are not enough for preparation.

Keeping in view the requirement of students and non availability of suitable App, it is highly required to develop an App that helps the student to prepare for competitive exams like UGC Net and GATE. Existing system available for preparation is to study by sitting at one place and there is no facility available for preparation while they are on move.

The major problem is that most of the Apps give only details about how to prepare, rather than helping the students to prepare. Numbers of Questions available are also limited. There is no provision to store the results of students, in case, the Apps provides facility of tests. Also the interface of the Existing Apps is not interesting. The limitation of the existing apps is that they are not according to the requirement of students. The idea is to make a general app having questions for all the subjects in a hierarchical manner and to include tutorials for all those subjects to help them learn and then they can test themselves by giving the tests. So an App iquiz is designed and developed.

## DESIGN OF PROPOSED SYSTEM

The Proposed App aims to be User centric. After registration, user can log in and can start the test upon selecting the subject. Each question will be given one minute. User has to answer the test within the time and after submission, correct answers will be shown. If the user doesn’t give the answer within specified time then next question will appear. Also user can view his/her previous results but administrator can view the results of all the users.

Actual Users are the students, by logging in they can play quiz. Admin task is to add more questions and to keep track of previous as well as current results of quiz. Fig. 3 and 4 depict use case diagrams for Admin and users of the app.

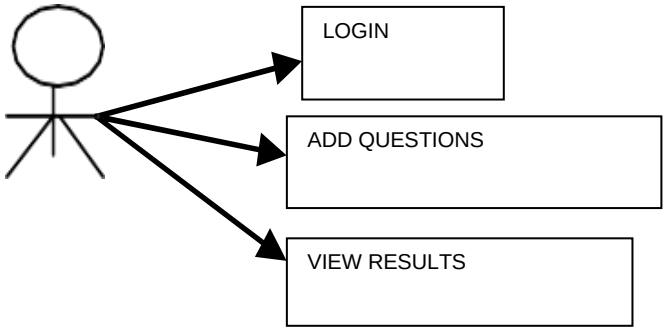



**Fig. 3 Use Case Diagram of Admin**

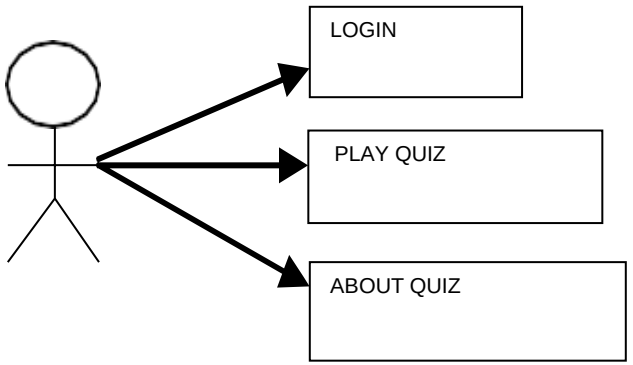
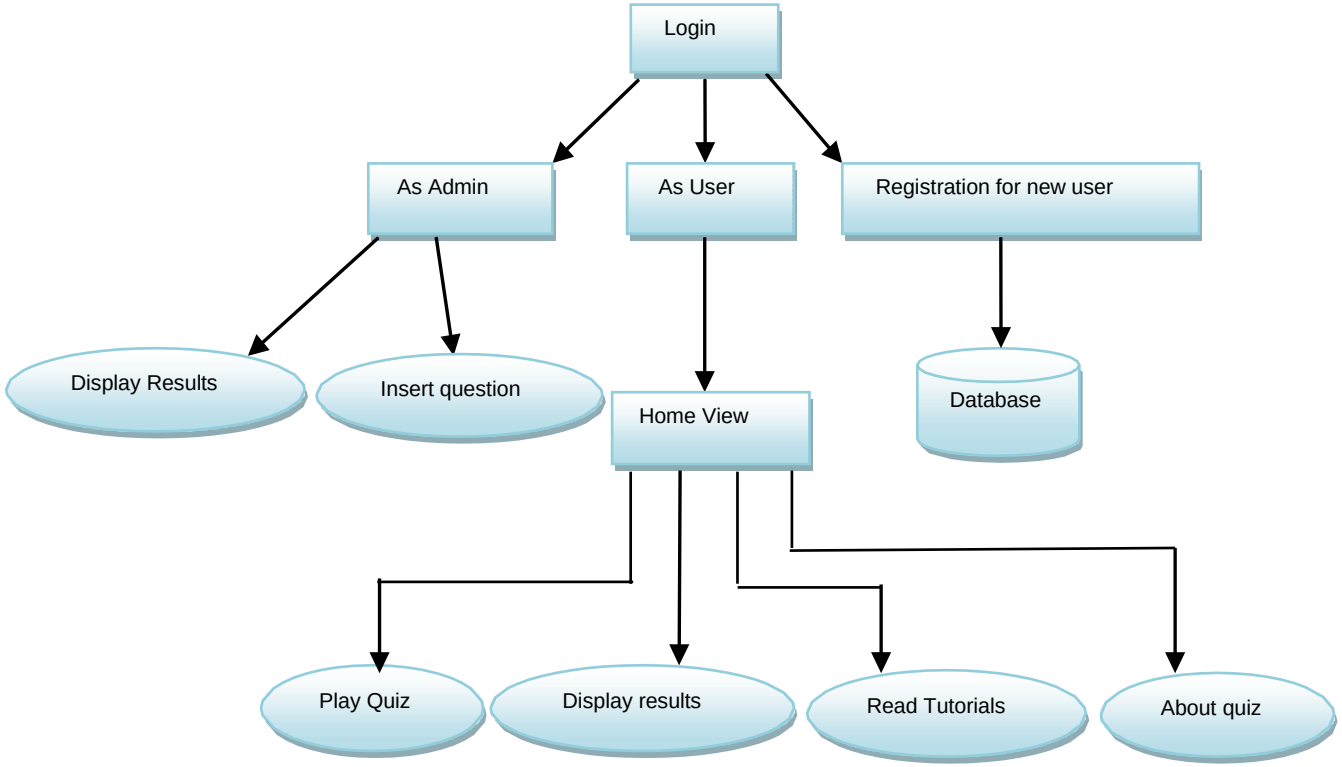



**Fig. 4 Use Case Diagram of User**




Technically, when a user starts the App he/she can login as an administrator or a user, in case user is already registered or he/she can register. After registration, a username and password will be assigned to user. If the user login as an administrator, user can view the results of all the other users or can insert the questions into the database by selecting a particular area and then by selecting the subject

related to that area. If the user has login as simple user, then home view of the App will be visible. In the home view, there are five icons for different purposes. By clicking the Quiz icon, he/she can play the quiz, read the tutorials and display the results etc. For each question time of one minute will be given. User has to answer the question within that time. If the user answers the question, correct answer will be shown. User can read the tutorial for learning. User can view his/her own previous results but administrator can view the results of all the users. Diagrammatic View of the App is shown in Fig. 5.

## IMPLEMENTATION DETAILS

For implementation of the App iquiz, the Platform used is Android 4.2 and language used is Java and

XML. SQLite is used at the backend. For setting environment of Android, there is a minimum requirement of JDK i.e java must be installed on the system. JDK
[8] SE development kit 7 is used for this purpose. Code is written by using a editor i.e Eclipse which is a opensource editor available [9]. After Eclipse (Juno, the default version)[9] is installed, there is a need to connect Android SDK with Eclipse,this is done by using ADT Plugin[10]. By using ADT Plugin link, plugin developer tools can be downloaded and installed. SQLite is automatically embedded into the Android device. Using SQLite in Android doesn't require any setup to be followed. Developers only have to define SQL statements for creating and updating database in SQLite. If the App creates and uses a database then the default location of database is DATA/data/APP_NAME/databases/FILENAME. Various modules of the app are as follows:

A. **User login module:** In this module, user has to first register. User will be moved to Register activity after clicking register button on Login screen. User has to fill details like Username, Email-id, Mobile number, Password and Confirm Password Etc. Only registered user can play quiz by logging in.
B. **Admin module:** If the user log in as an Admin then user can add new questions by selecting a particular area and a subject related to that area or make any changes to database. Admin can display result of all users, datewise or name wise as the case may be.
C. **Registration for new User:** If the user wants to play quiz, first user must register by filling details like User name,Email- id,Password,Confirm Password etc. After successful Registration, user can login.

After login, user can Play the quiz.

D. **Home view:** This module shows icons for playing quiz, About, Results, Exit. When user clicks on Quiz icon, the options will be displayed, representing different fields. When user select some specific subject then random questions will be asked. Each Question will be given one minute. User has to answer the question within that time. At last, result will be displayed.
E. **Results view:** This module is used to display previous as well as current scores of users. By viewing the results, user can measure performance compared to last attempts.

The main objective of this project was to test the knowledge of user in various fields and also on the basis of score, students can judge, where they are lacking. It provides help in every field included in iquiz application. It provides tutorials to students so that they can learn along with giving the tests.

## FINDINGS

Based upon personal experience in developing iquiz and available literature, the following are the challenges faced by the Android App developers:

**Multiple Devices:** The most common challenge is to set the properties of app for different devices with different screen sizes, resolution etc. There are many versions of each android device and while releasing the app, version specific details need to be checked this makes the task very critical. As it need to be run in different devices, therefore it also affects the cost and budget implications. For each version of Android, developer has to write code again because there are migration problem available to migrate the existing code to new platform. Sometimes, behavior is also different across new platform.

A. **Testing of Android Application**: Currently, the Development Environment does not have enough tools for testing. There is a need of testing techniques for the Android Platform. Also debugging features must be made available.

B. **Limited Capabilities of Different Devices:** Sometimes different devices have different capabilities in terms of software support like some browsers has poor support for HTML5.
C. **Emulators/Simulators Problem:** Emulators are the devices which provides us hardware environment of Android Devices while Simulators provide us the software environment. Apps are tested on the emulators which are not enough for testing. Emulators are very slow and take a lot of time to start and run.
D. **Data Intensive Apps:** Since Mobile Devices have very limited memory, so it is very difficult to store huge amount of data in it. Offline caching doesn't work well and synchronizing with another data source is a challenging task.
E. **Lack of Software/Hardware Integration:** One Button on Android device works differently on another device. So it is difficult to build an app that relies on a particular hardware to do a thing.
F. **Security Issue:** Since Android is an open Source operating system so there can be malware Apps which users can install by mistake or by any other reason that can lead to loss of data.

# Abstract


- The main aim for developing this project is to manage Student details, prospective Students, Labs, Course, Course Units, Files, and all the Students related data.
- This system is very useful for schools, colleges and universities.
- This project is a student-level data collection system.
- This application has a good appearance and is very easy to operate.
- This project contains a lot of advance modules which makes the back end system very powerful.

# Present Challenges

- It is very difficult for students to manage all their subjects or courses related work properly.
- Sometimes they forget about doing some assignments.
- It is not easy to remember all the key note points of the day and about any future quiz that you have to attempt after someday on your school/college app.
- It is very important for every student to have their past reports of assignments, quizzes, lab-work so that they can analyse their present performance on the basis of that.

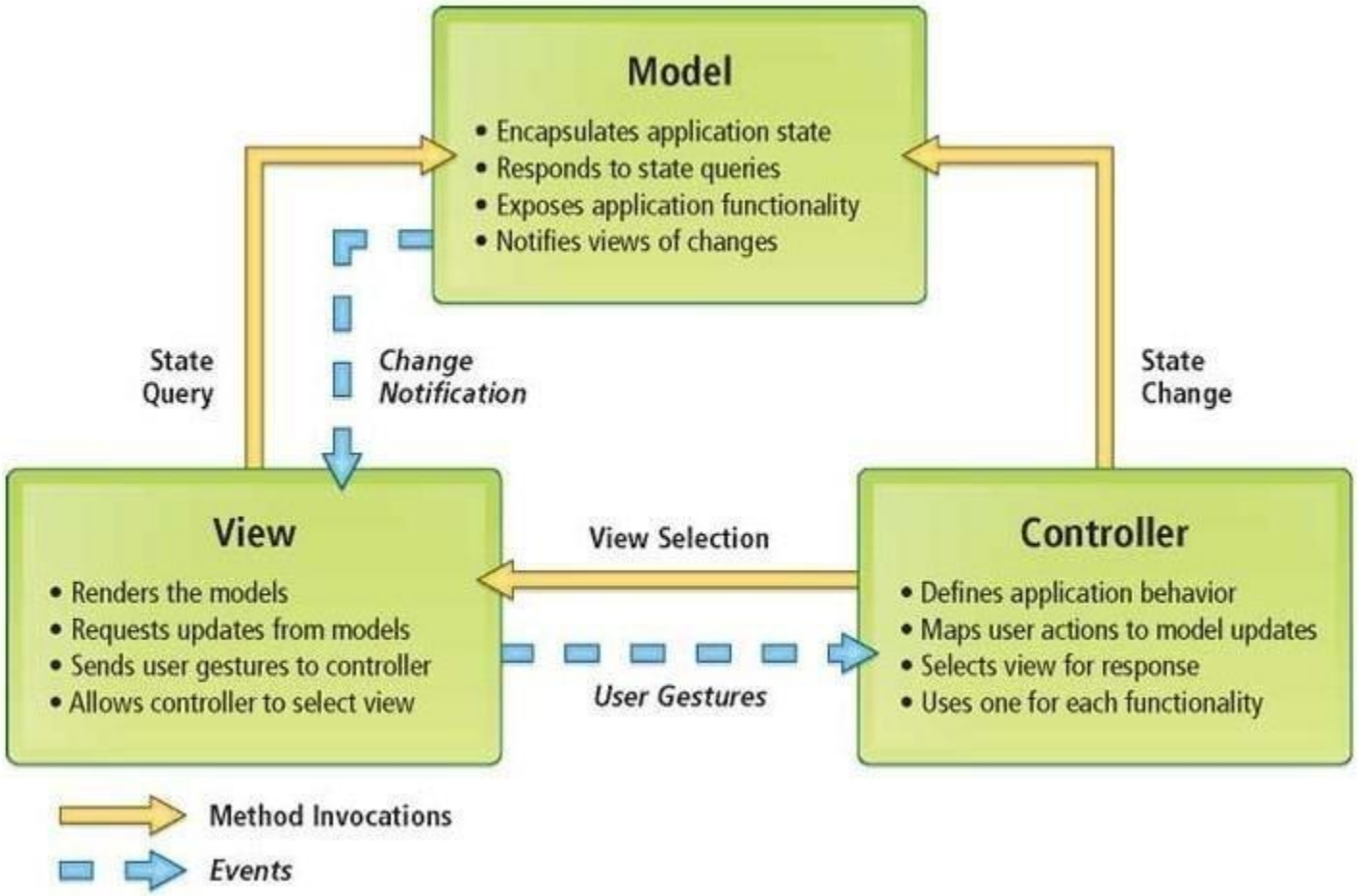
# Features

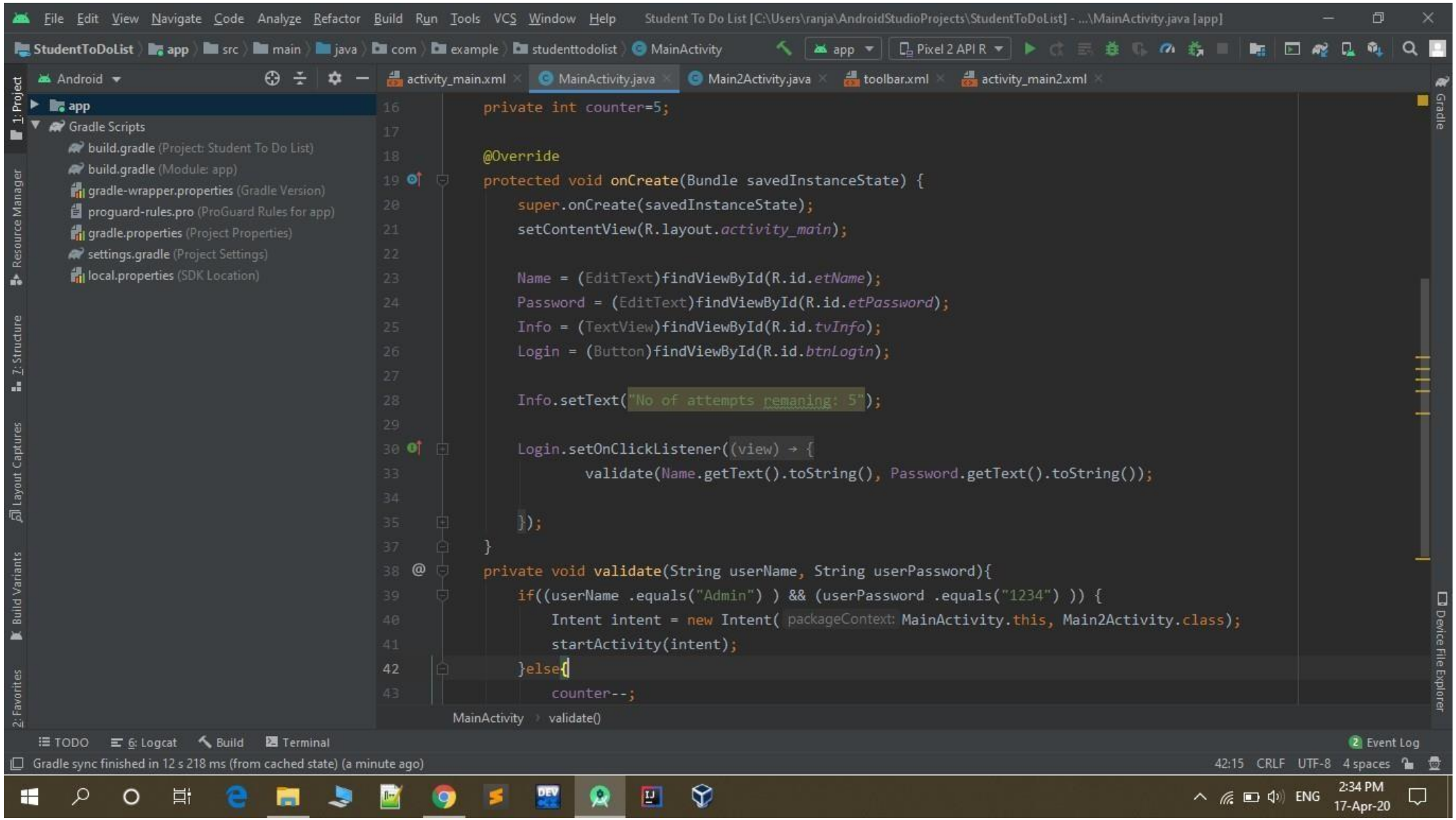
- Student Task Management System provides the searching facilities based on various factors. Such as Student details, prospective Students, Labs, Course, Course Units, Files.
- Manage the information of Students.
- Shows the information and description of the Labs.
- To increase efficiency of managing the Course.
- It deals with monitoring the information and transactions of Course Units.
- Manage the information of Files.
- Editing, adding and updating of Records is improved which results in proper resource management of Student Task Management System data.
- Manage the information of Task
- Integration of all records of Task Management.

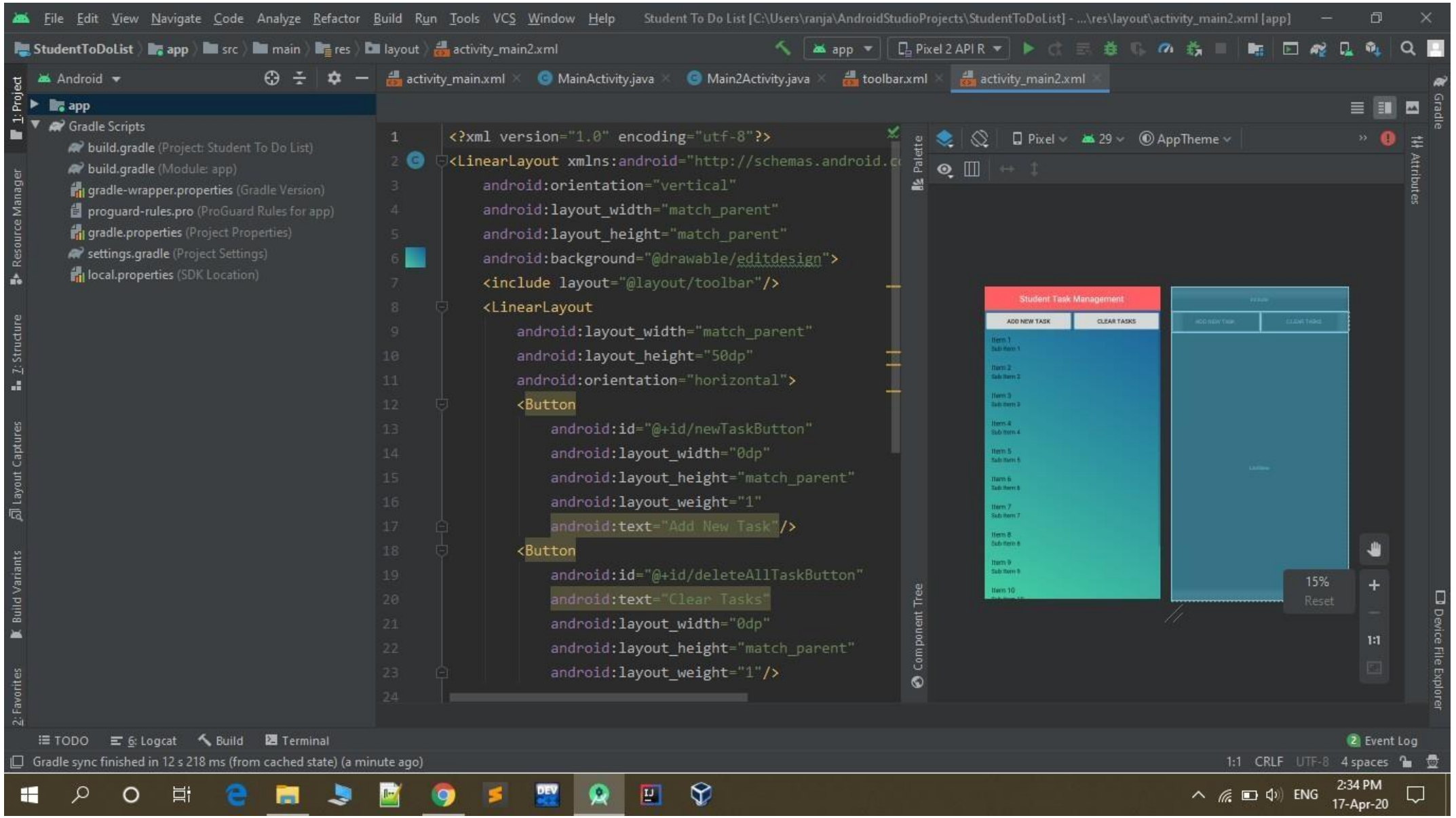
# Implementation Methodology

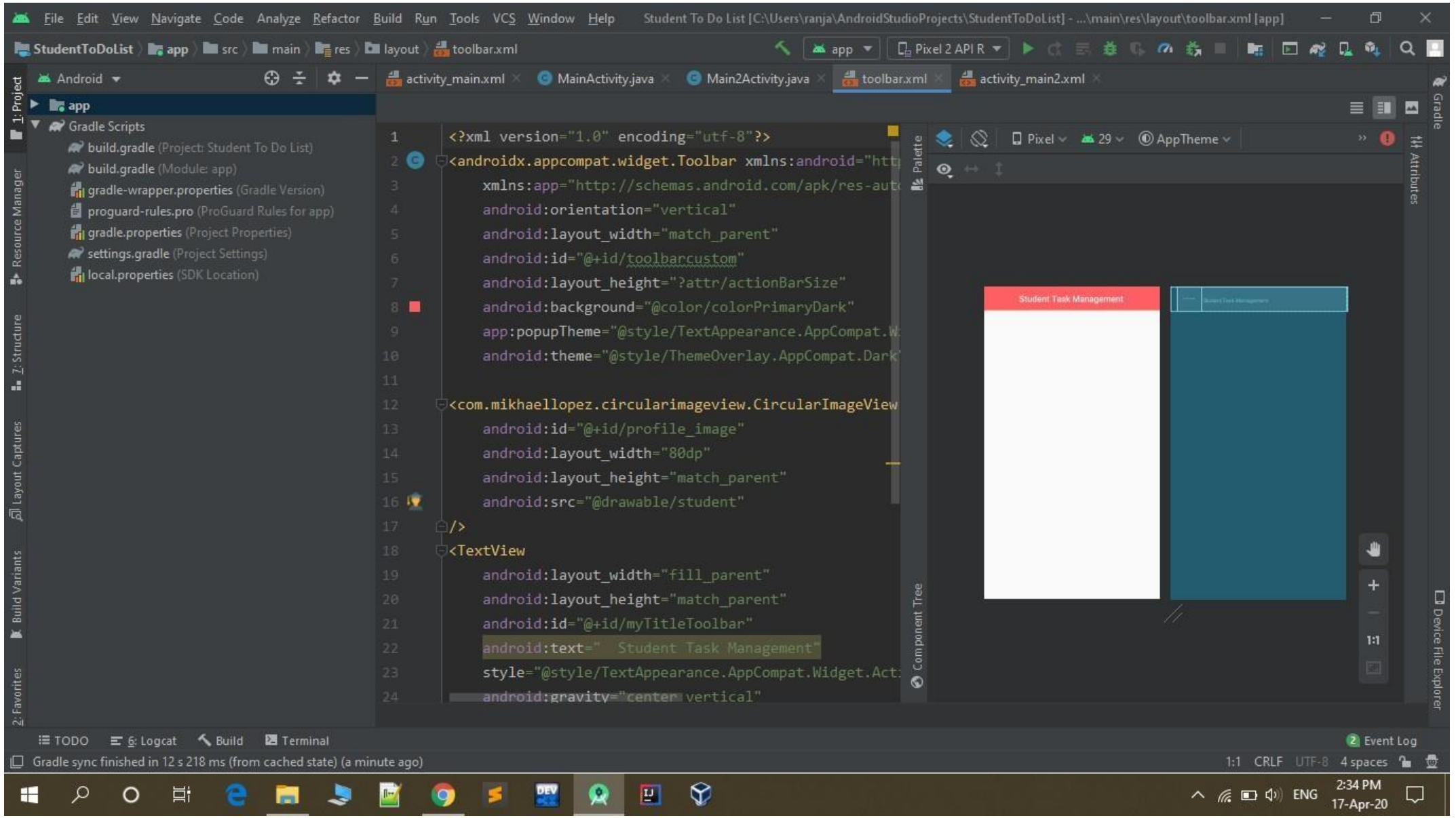
- **Model** - The lowest level of the pattern which is responsible for maintaining data.
- **View** - This is responsible for displaying all or a portion of the data to the user.
- **Controller** - Software Code that controls the interactions between the Model and View.
- MVC is popular as it isolates the application logic from the user interface layer and supports separation of concerns.  Here the Controller receives all requests for the application and then works with the Model to prepare any data needed by the View. The View then uses the data prepared by the Controller to generate a final presentable response. The MVC abstraction can be graphically represented as follows.

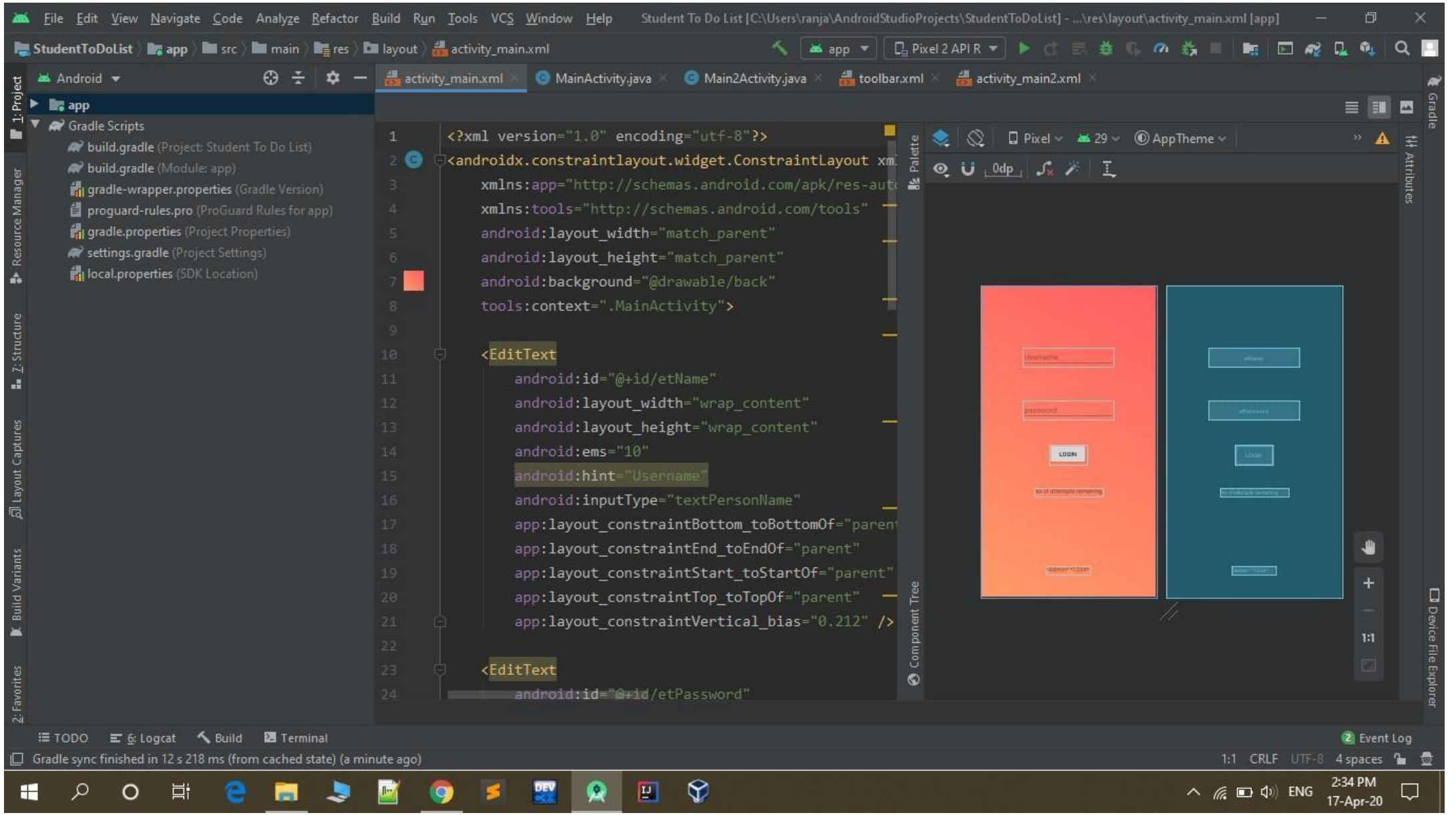
- **MVC (Model View Controller Flow) Diagram**




File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Student To Do List [C:\Users\ranja\AndroidStudioProjects\StudentToDoList] - ...\MainActivity.java [app]
StudentToDoList app src main java com example studenttodolist MainActivity
app
Pixel 2 API R
Android
app
Gradle Scripts
build.gradle (Project: Student To Do List)
build.gradle (Module: app)
gradle-wrapper.properties (Gradle Version)
proguard-rules.pro (ProGuard Rules for app)
gradle.properties (Project Properties)
settings.gradle (Project Settings)
local.properties (SDK Location)
activity_main.xml
MainActivity.java
Main2Activity.java
toolbar.xml
activity_main2.xml
private int counter=5;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
Name = (EditText)findViewById(R.id.etName);
Password = (EditText)findViewById(R.id.etPassword);
Info = (TextView)findViewById(R.id.tvInfo);
Login = (Button)findViewById(R.id.btnLogin);
Info.setText("No of attempts remaning: 5");
Login.setOnClickListener((view) → {
validate(Name.getText().toString(), Password.getText().toString());
});
}
private void validate(String userName, String userPassword){
if((userName .equals("Admin") ) && (userPassword .equals("1234") )) {
Intent intent = new Intent( packageContext: MainActivity.this, Main2Activity.class);
startActivity(intent);
}else{
counter--;
MainActivity › validate()
TODO
6: Logcat
Build
Terminal
Event Log
Gradle sync finished in 12 s 218 ms (from cached state) (a minute ago)
42:15 CRLF UTF-8 4 spaces
ENG
2:34 PM
17-Apr-20


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Student To Do List [C:\Users\ranja\AndroidStudioProjects\StudentToDoList] - ...\res\layout\activity_main2.xml [app]
StudentToDoList app src main res layout activity_main2.xml
app
Pixel 2 API R
Android
app
Gradle Scripts
build.gradle (Project: Student To Do List)
build.gradle (Module: app)
gradle-wrapper.properties (Gradle Version)
proguard-rules.pro (ProGuard Rules for app)
gradle.properties (Project Properties)
settings.gradle (Project Settings)
local.properties (SDK Location)
activity_main.xml
MainActivity.java
Main2Activity.java
toolbar.xml
activity_main2.xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.c
android:orientation="vertical"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/editdesign">
<include layout="@layout/toolbar"/>
<LinearLayout
android:layout_width="match_parent"
android:layout_height="50dp"
android:orientation="horizontal">
<Button
android:id="@+id/newTaskButton"
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"
android:text="Add New Task"/>
<Button
android:id="@+id/deleteAllTaskButton"
android:text="Clear Tasks"
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="1"/>
Palette
Component Tree
Pixel
29
AppTheme
Attributes
Gradle
Device File Explorer
Student Task Management
ADD NEW TASK
CLEAR TASKS
15%
Reset
1:1
TODO
6: Logcat
Build
Terminal
Event Log
Gradle sync finished in 12 s 218 ms (from cached state) (a minute ago)
1:1 CRLF UTF-8 4 spaces
ENG
2:34 PM
17-Apr-20


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Student To Do List [C:\Users\ranja\AndroidStudioProjects\StudentToDoList] - ...\main\res\layout\toolbar.xml [app]
StudentToDoList app src main res layout toolbar.xml
app
Pixel 2 API R
Android
activity_main.xml
MainActivity.java
Main2Activity.java
toolbar.xml
activity_main2.xml
app
Gradle Scripts
build.gradle (Project: Student To Do List)
build.gradle (Module: app)
gradle-wrapper.properties (Gradle Version)
proguard-rules.pro (ProGuard Rules for app)
gradle.properties (Project Properties)
settings.gradle (Project Settings)
local.properties (SDK Location)
<?xml version="1.0" encoding="utf-8"?>
<androidx.appcompat.widget.Toolbar xmlns:android="http
xmlns:app="http://schemas.android.com/apk/res-aut
android:orientation="vertical"
android:layout_width="match_parent"
android:id="@+id/toolbarcustom"
android:layout_height="?attr/actionBarSize"
android:background="@color/colorPrimaryDark"
app:popupTheme="@style/TextAppearance.AppCompat.W
android:theme="@style/ThemeOverlay.AppCompat.Dark"
<com.mikhaellopez.circularimageview.CircularImageView
android:id="@+id/profile_image"
android:layout_width="80dp"
android:layout_height="match_parent"
android:src="@drawable/student"
/>
<TextView
android:layout_width="fill_parent"
android:layout_height="match_parent"
android:id="@+id/myTitleToolbar"
android:text="  Student Task Management"
style="@style/TextAppearance.AppCompat.Widget.Acti
android:gravity="center vertical"
Pixel
29
AppTheme
Student Task Management
Palette
Component Tree
Attributes
Gradle
Device File Explorer
TODO
6: Logcat
Build
Terminal
Event Log
Gradle sync finished in 12 s 218 ms (from cached state) (a minute ago)
1:1 CRLF UTF-8 4 spaces
ENG
2:34 PM
17-Apr-20


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Student To Do List [C:\Users\ranja\AndroidStudioProjects\StudentToDoList] - ...\res\layout\activity_main.xml [app]
StudentToDoList app src main res layout activity_main.xml
app
Pixel 2 API R
Android
app
Gradle Scripts
build.gradle (Project: Student To Do List)
build.gradle (Module: app)
gradle-wrapper.properties (Gradle Version)
proguard-rules.pro (ProGuard Rules for app)
gradle.properties (Project Properties)
settings.gradle (Project Settings)
local.properties (SDK Location)
activity_main.xml
MainActivity.java
Main2Activity.java
toolbar.xml
activity_main2.xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xm
xmlns:app="http://schemas.android.com/apk/res-aut
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/back"
tools:context=".MainActivity">
<EditText
android:id="@+id/etName"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:ems="10"
android:hint="Username"
android:inputType="textPersonName"
app:layout_constraintBottom_toBottomOf="paren
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.212" />
<EditText
android:id="@+id/etPassword"
Pixel
29
AppTheme
0dp
Palette
Component Tree
Attributes
Gradle
Device File Explorer
TODO
6: Logcat
Build
Terminal
Event Log
Gradle sync finished in 12 s 218 ms (from cached state) (a minute ago)
1:1 CRLF UTF-8 4 spaces
ENG
2:34 PM
17-Apr-20

# Activity_main.xml-Code:-

```
?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    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/back"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/etName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Username"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.212" />

    <EditText
        android:id="@+id/etPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="password"
        android:inputType="textPassword"
        app:layout_constraintBottom_toBottomOf="parent"
```

```
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/etName"
        app:layout_constraintVertical_bias="0.161" />

    <Button
        android:id="@+id/btnLogin"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="login"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/etPassword"
        app:layout_constraintVertical_bias="0.154" />

    <TextView
        android:id="@+id/tvInfo"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="No of attempts remaning"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/btnLogin"
        app:layout_constraintVertical_bias="0.181" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="*Admin* *1234* "
        app:layout_constraintBottom_toBottomOf="parent"
```

```
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/tvInfo"
        app:layout_constraintVertical_bias="0.75" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Activity_main.java-Code:-

```
package com.example.studenttodolist;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    private EditText Name, Password;
    private TextView Info;
    private Button Login;
    private int counter=5;

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

```
        Name = (EditText)findViewById(R.id.etName);
        Password = (EditText)findViewById(R.id.etPassword);
        Info = (TextView)findViewById(R.id.tvInfo);
        Login = (Button)findViewById(R.id.btnLogin);

        Info.setText("No of attempts remaning: 5");

        Login.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                validate(Name.getText().toString(), Password.getText().toString());

            }
        });
    }
    private void validate(String userName, String userPassword){
        if((userName .equals("Admin") ) && (userPassword .equals("1234") )) {
            Intent intent = new Intent(MainActivity.this, Main2Activity.class);
            startActivity(intent);
        }else{
            counter--;
            Info.setText("No of attempts remaning:" +String.valueOf(counter));
            if(counter==0){
                Login.setEnabled(false);
            }
        }
    }
}
```

# Conclusion

- It increases the efficiency of students to perform their tasks.
- The App also acts as a reminder if someone missed to do some assignment, lab-work or attempt any quiz etc.
- It generates the report on Student details, prospective Students, Labs, Course, Course Units, Files.
- Provide filter reports on Student details, prospective Students, Labs, Course, Course Units, Files.
- You can easily export PDF of the daily performance report of the student.