DEPARTMENT OF COMPUTER SCIENCE

Android Application Development

Lab Manual

Celebrate life

RATHINAM

RATHINAM COLLEGE OF ARTS & SCIENCE (AUTONOMOUS)

Prepared by:	Approved & Reviewed by:	Issued by:	W.e.f Date:
S.KARTHIKEYAN			JUNE-2019

In-charge **HOD** **Principal**

Celebrate life

RATHINAM

RATHINAM COLLEGE OF ARTS & SCIENCE

Rathinam Techzone Campus, Pollachi Road, Eachanari,
Coimbatore - 641021, Tamil Nadu.
http://www.rathinamcollege.com

Department of Computer Science

Lab Manual for the Academic Year 2019-20

(in accordance with Computer Science syllabus)

SUBJECT : Android Application Development

STREAM : BCA/B.Sc [CS/CT/IT]

H.O.D DEAN

List of Lab Exercises

1. Getting Started with Mobile App Development

2. Create and validate a login application using username as Email ID else login button must remain disabled

3. Creating a Hello World program Experiment with the most basic features and mobile application interaction concepts lists, text boxes, buttons, graphics, etc)

4. Create a following menu items in mobile application program a. cut b. copy c. pasted. delete e. select all f. unselect all

5. Create an application to change screen color as per the user choice from a menu

6. Create an Android application with a combo box, spinner, toast message get selected item

7. Create an application to call a phone number entered by the user and edit Text

8. Develop a native application that uses GPS location information.

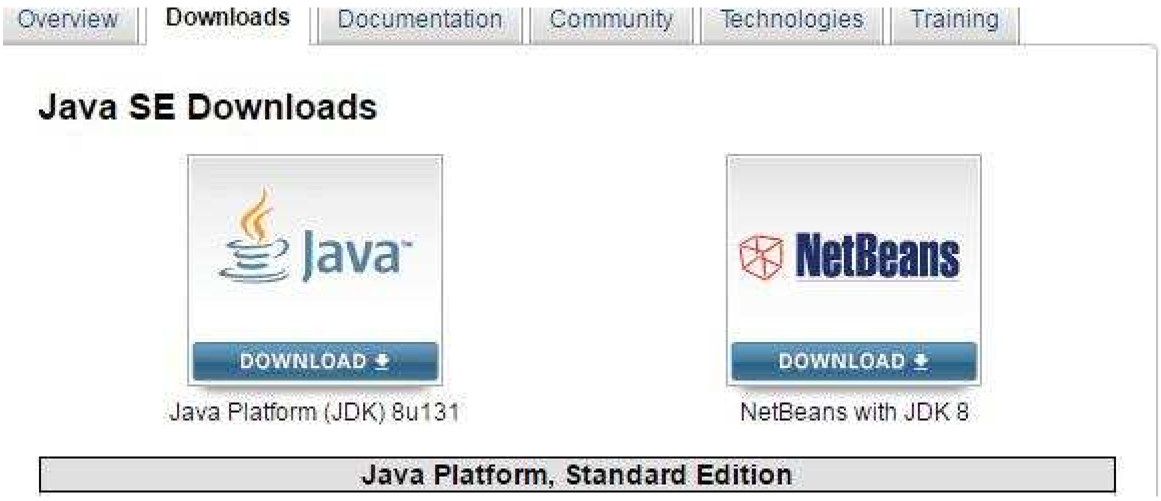
1. GETTING STARTED WITH ANDROID

Aim:

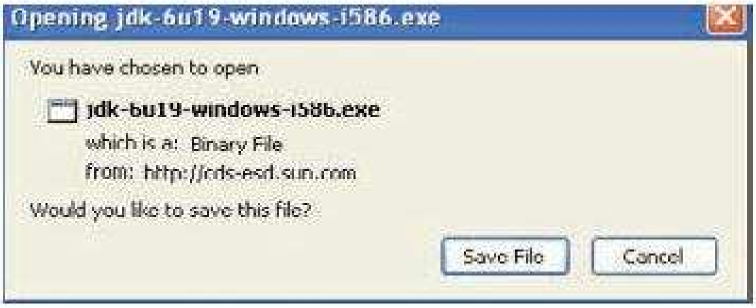
To know how to get started with Android Development Environment .

Procedure:

Step 1: Download the **Java Development Kit(jdk)** from the oracle Website**.**



Step 2: Downloadable **jdk** will be shown.Click **Save file.**



Opening jdk-6u19-windows-i586.exe

You have chosen to open

jdk-6u19-windows-i586.exe

which is a: Binary File

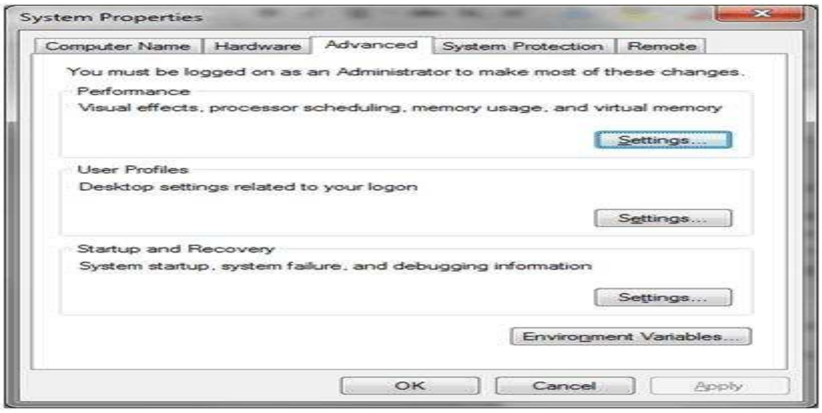
from: http://cds-esd.sun.com

Would you like to save this file?

Step 3: Install the downloaded **Java Development Kit.**

Step 4: It will install all the required **Java Runtime Environment.**

Step 5: Select **Start** -> **Control Panel -> System** and select the **Advanced** tab and then click on **Environment Variables.**



System Properties

You must be logged on as an Administrator to make most of these changes.

Performance

Visual effects, processor scheduling, memory usage, and virtual memory

User Profiles

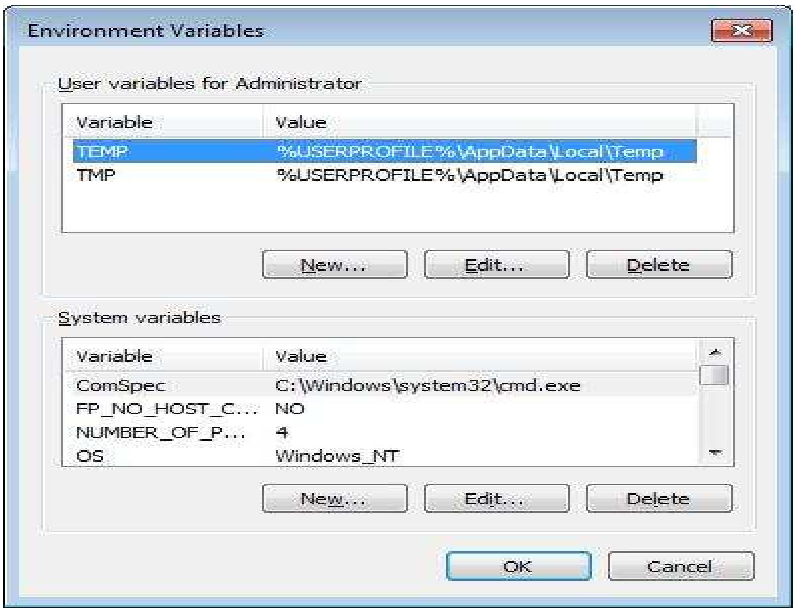
Desktop settings related to your logon

Startup and Recovery

System startup, system failure, and debugging information

Step 6: Select **Start** -> **Control Panel -> System -> Advanced System Settings.**In the **Advanced** tab and then click on **Environment Variables.**

Step 7 : On the **Environment Variables,**In the **System Variables,**Select **Path.**



Environment Variables

User variables for Administrator

Variable	Value
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

System variables

Variable	Value
ComSpec	C:\Windows\system32\cmd.exe
FP_NO_HOST_C...	NO
NUMBER_OF_P...	4
OS	Windows_NT

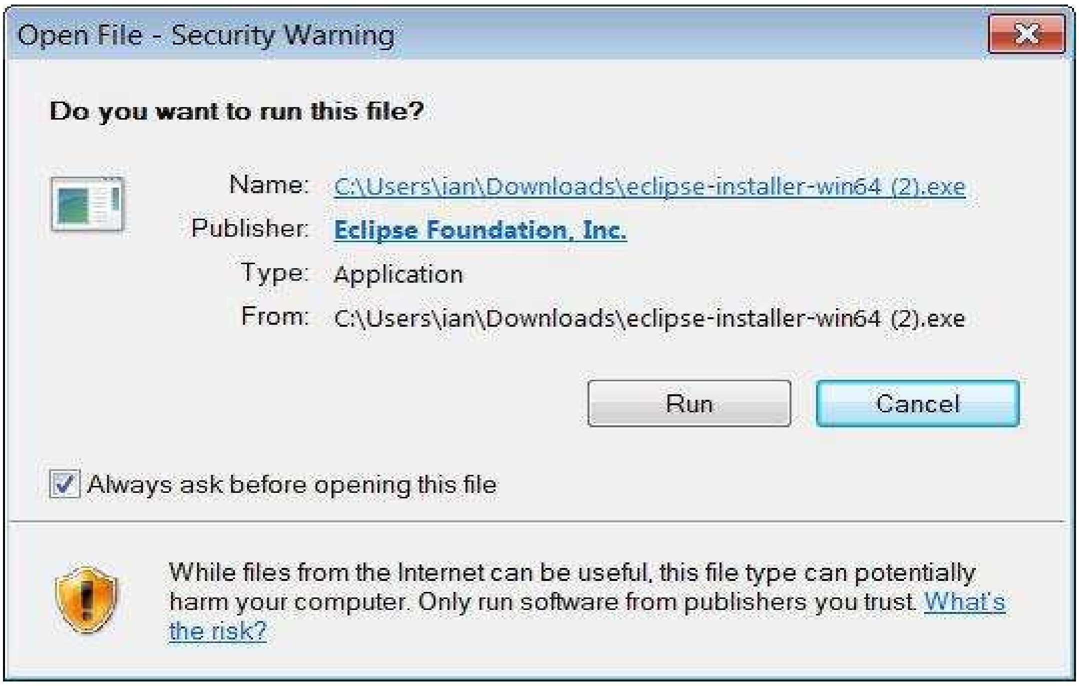
Step 8 : On the **Edit System Variables** window, Specify the following Path(Example)**.**

set PATH=C:\jdk1.8.0_102\bin;%PATH%

set JAVA_HOME=C:\jdk1.8.0_102

Step 9 : Download **Android SDK** from the website.

Step 10 : Download **Android SDK** from the website.



Open File - Security Warning

Do you want to run this file?

Name: C:\Users\ian\Downloads\eclipse-installer-win64 (2).exe

Publisher: Eclipse Foundation, Inc.

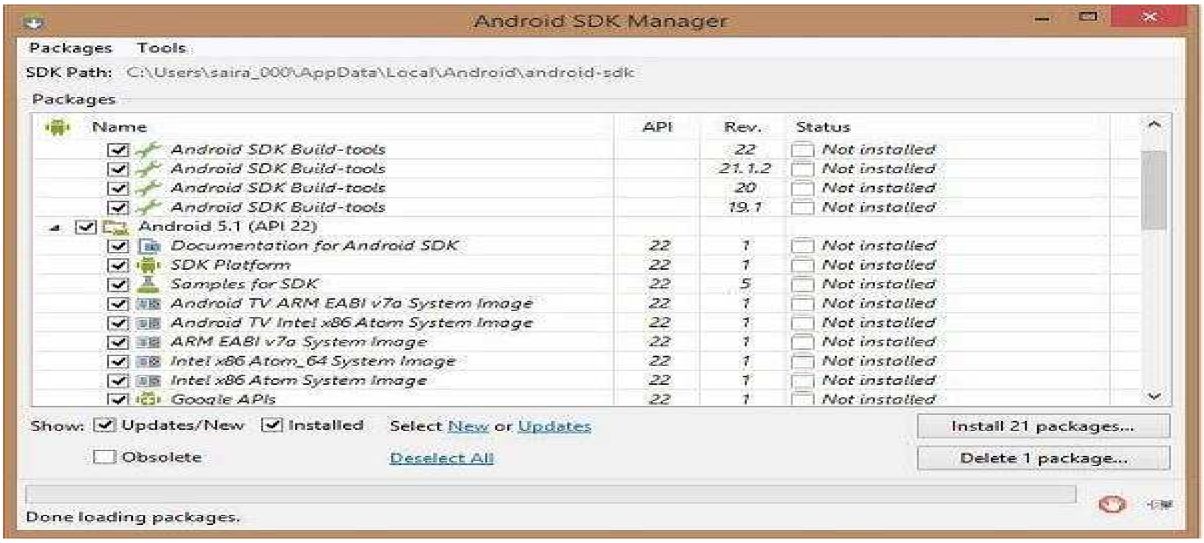
Type: Application

From: C:\Users\ian\Downloads\eclipse-installer-win64 (2).exe

☑ Always ask before opening this file

While files from the Internet can be useful, this file type can potentially harm your computer. Only run software from publishers you trust. What's the risk?

Step 11 : Select **All Programs > Android SDK Tools > SDK Manager**, this will give you following window.



Android SDK Manager

Packages Tools

SDK Path: C:\Users\saira_000\AppData\Local\Android\android-sdk

Packages

Name	API	Rev.	Status
☑ Android SDK Build-tools		22	☐ Not installed
☑ Android SDK Build-tools		21.1.2	☐ Not installed
☑ Android SDK Build-tools		20	☐ Not installed
☑ Android SDK Build-tools		19.1	☐ Not installed
☑ Android 5.1 (API 22)			
☑ Documentation for Android SDK	22	1	☐ Not installed
☑ SDK Platform	22	1	☐ Not installed
☑ Samples for SDK	22	5	☐ Not installed
☑ Android TV ARM EABI v7a System Image	22	1	☐ Not installed
☑ Android TV Intel x86 Atom System Image	22	1	☐ Not installed
☑ ARM EABI v7a System Image	22	1	☐ Not installed
☑ Intel x86 Atom_64 System Image	22	1	☐ Not installed
☑ Intel x86 Atom System Image	22	1	☐ Not installed
☑ Google APIs	22	1	☐ Not installed

Show: ☑ Updates/New ☑ Installed Select New or Updates

☐ Obsolete Deselect All

Install 21 packages...

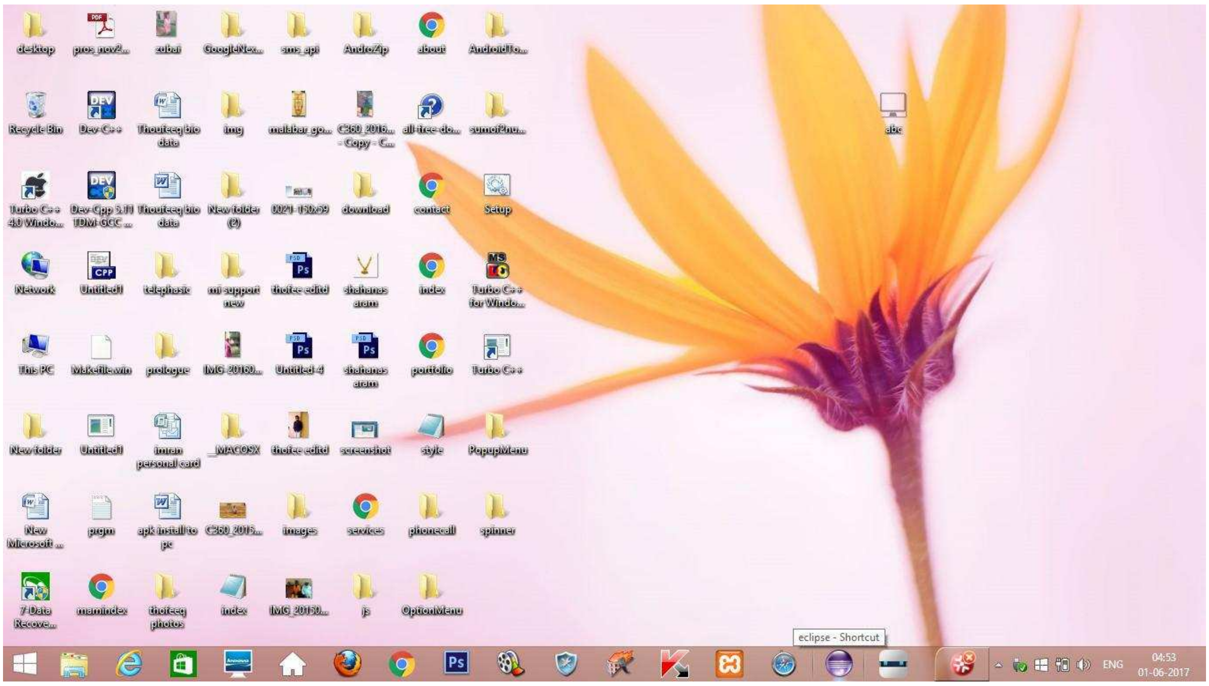
Delete 1 package...

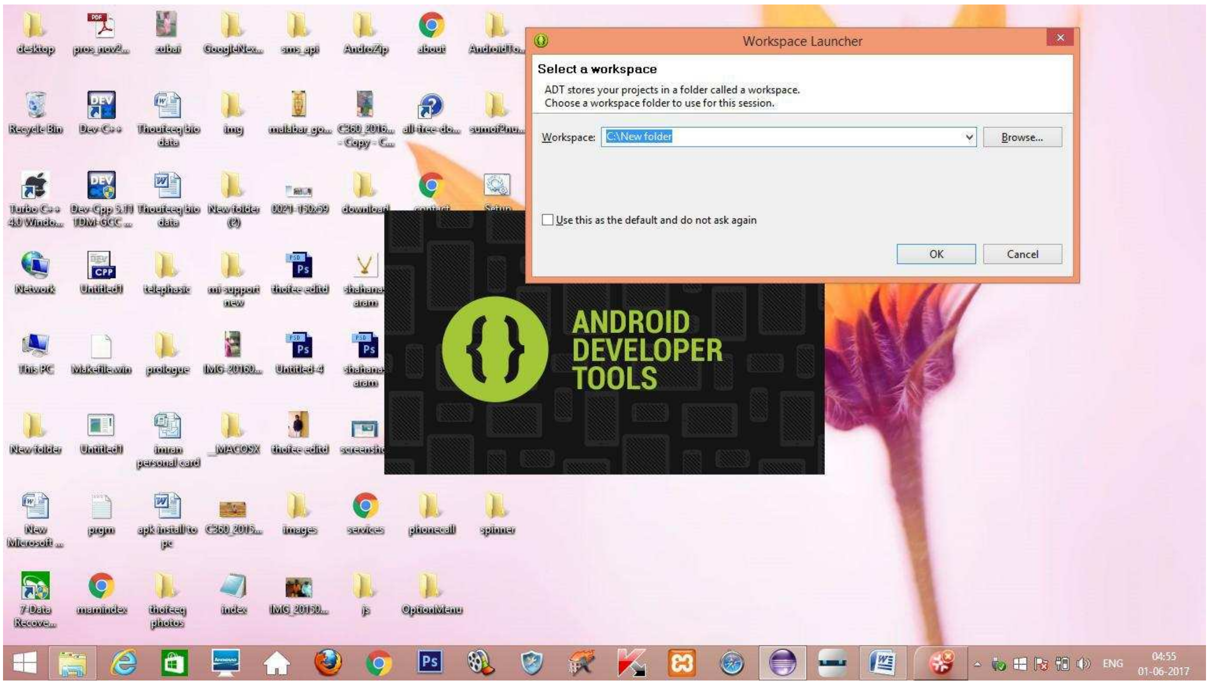
Done loading packages.

Step 12 : On the **Android SDK Manager** Window, Select **Install Packages**. **Step 13 :** On the **Choose Packages to Install** Window, Check **Accept,**and Click **Install**.

Step 14 : Android SDK Will be installed.

Output:





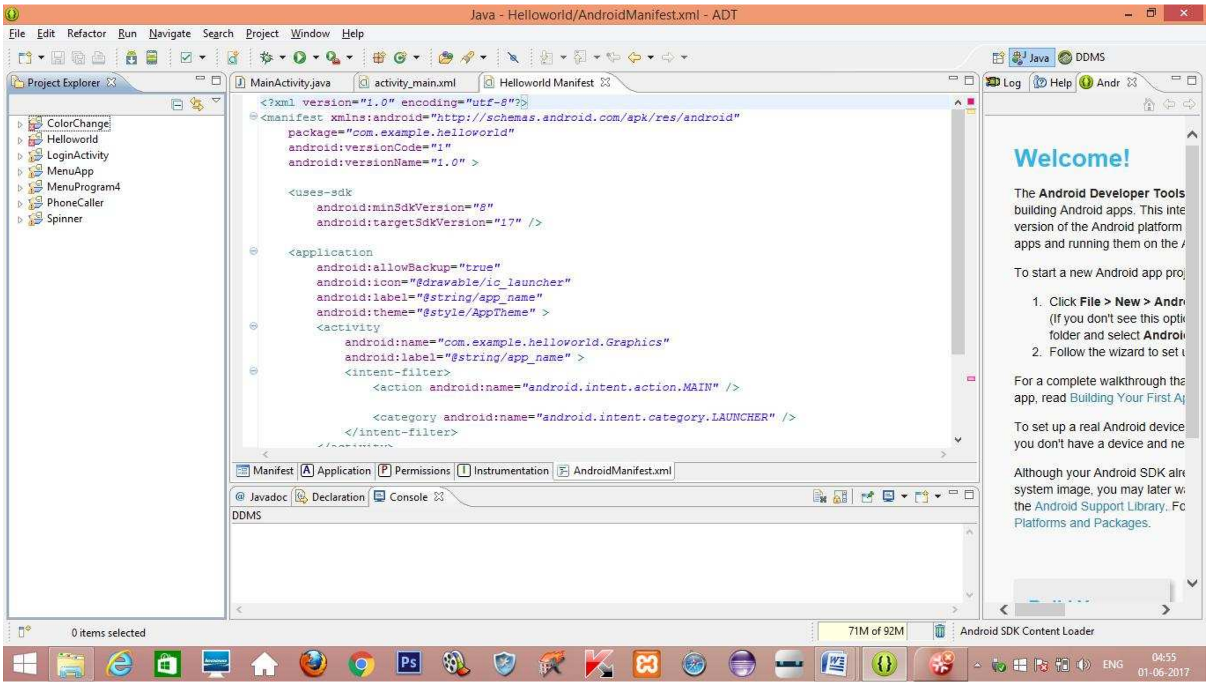
Workspace Launcher

Select a workspace

ADT stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

Workspace: C:\New folder

□ Use this as the default and do not ask again



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.helloworld"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="17" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name="com.example.helloworld.Graphics"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

The **Android Developer Tools** building Android apps. This inte version of the Android platform apps and running them on the

To start a new Android app proj

1. Click **File > New > Andr** (If you don't see this opti folder and select **Androi**
2. Follow the wizard to set u

For a complete walkthrough tha app, read Building Your First Ap

To set up a real Android device you don't have a device and ne

Although your Android SDK alre system image, you may later w the Android Support Library. Fo Platforms and Packages.

Result:

The above aim of the program has been achieved successfully.

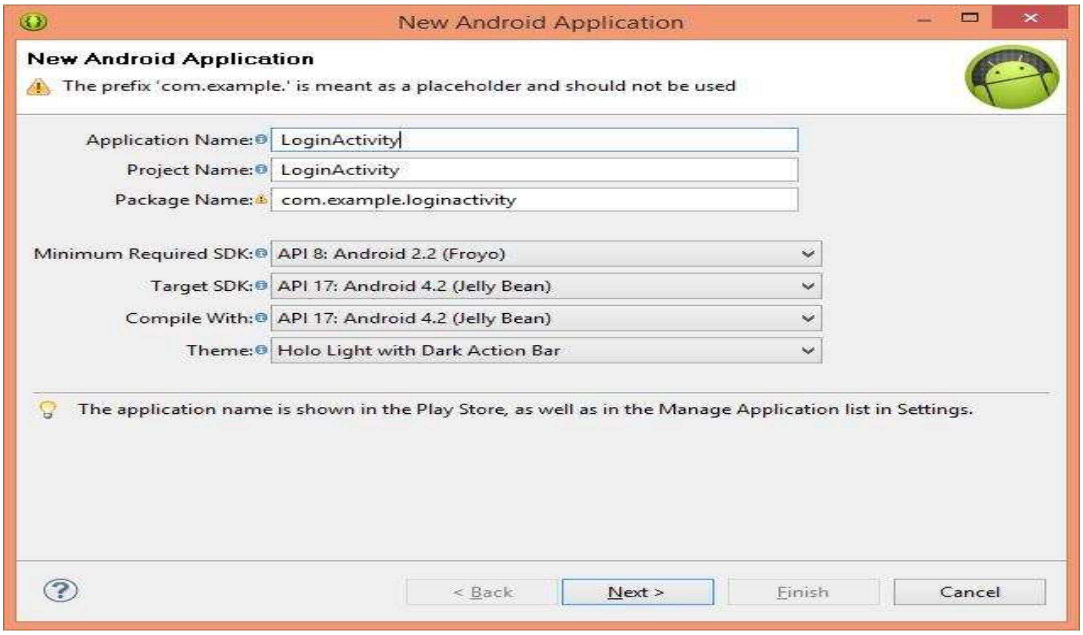
2.LOGIN ACTIVITY

Aim:

To create an android application with login operations.

Procedure:

Step 1: Open Eclipse IDE and go to **File -> New** -> **Project** -> **Android** -> **Android Application Project**. You have to specify the **Application Name**, the **Project Name** and the **Package name** in the appropriate text fields and then click **Next**.



New Android Application

The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: LoginActivity

Project Name: LoginActivity

Package Name: com.example.loginactivity

Minimum Required SDK: API 8: Android 2.2 (Froyo)

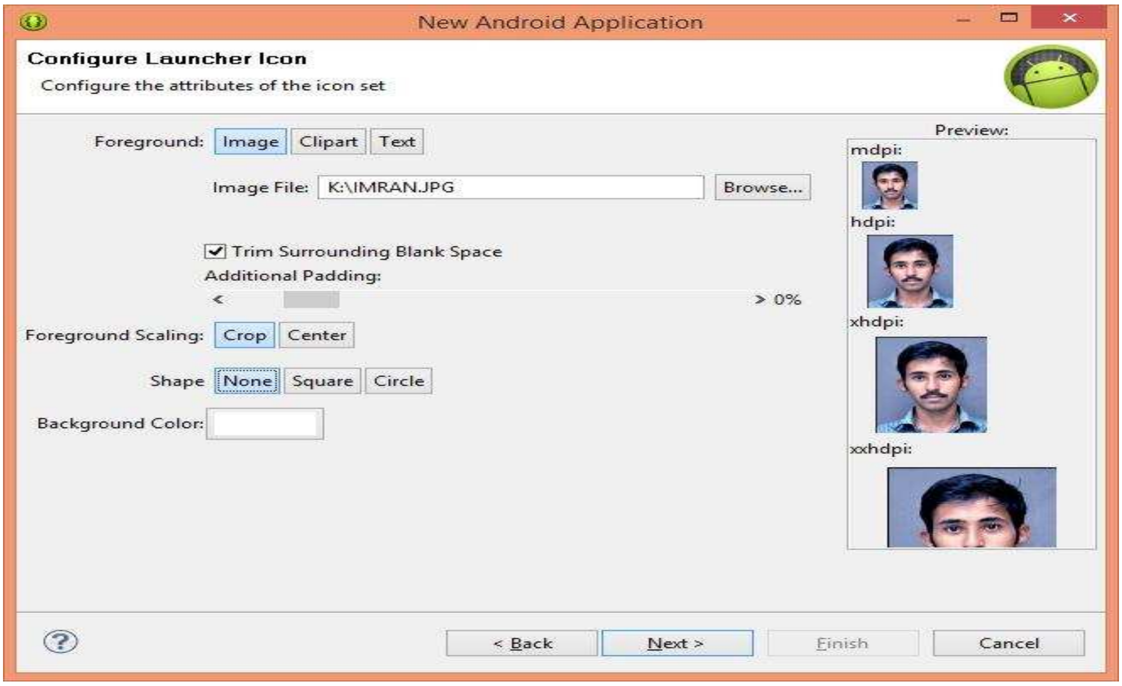
Target SDK: API 17: Android 4.2 (Jelly Bean)

Compile With: API 17: Android 4.2 (Jelly Bean)

Theme: Holo Light with Dark Action Bar

The application name is shown in the Play Store, as well as in the Manage Application list in Settings.

Step 2: In the next window make sure the select the **Application Launcher Icon** option is selected in order to create a new activity for your project, and click **Next**.



Configure Launcher Icon

Configure the attributes of the icon set

Foreground: Image Clipart Text

Image File: K:\IMRAN.JPG

☑ Trim Surrounding Blank Space

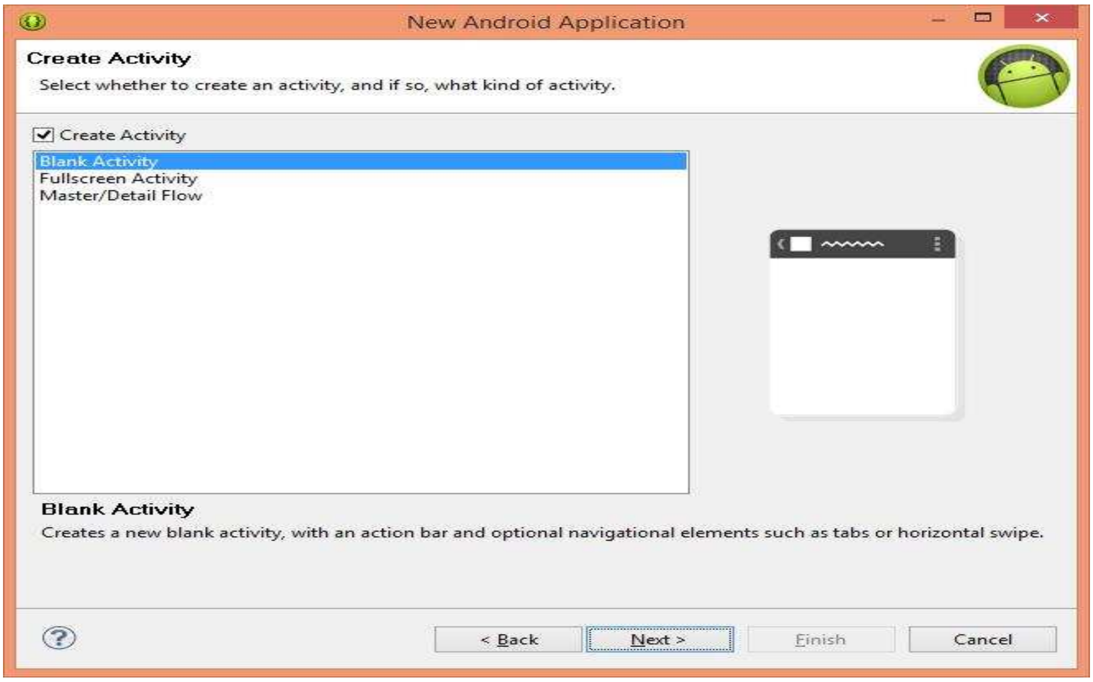
Additional Padding: 0%

Foreground Scaling: Crop Center

Shape None Square Circle

Background Color:

Step 3: On the next window click to select the **Blank Activity** and click **Next**.



New Android Application

Create Activity

Select whether to create an activity, and if so, what kind of activity.

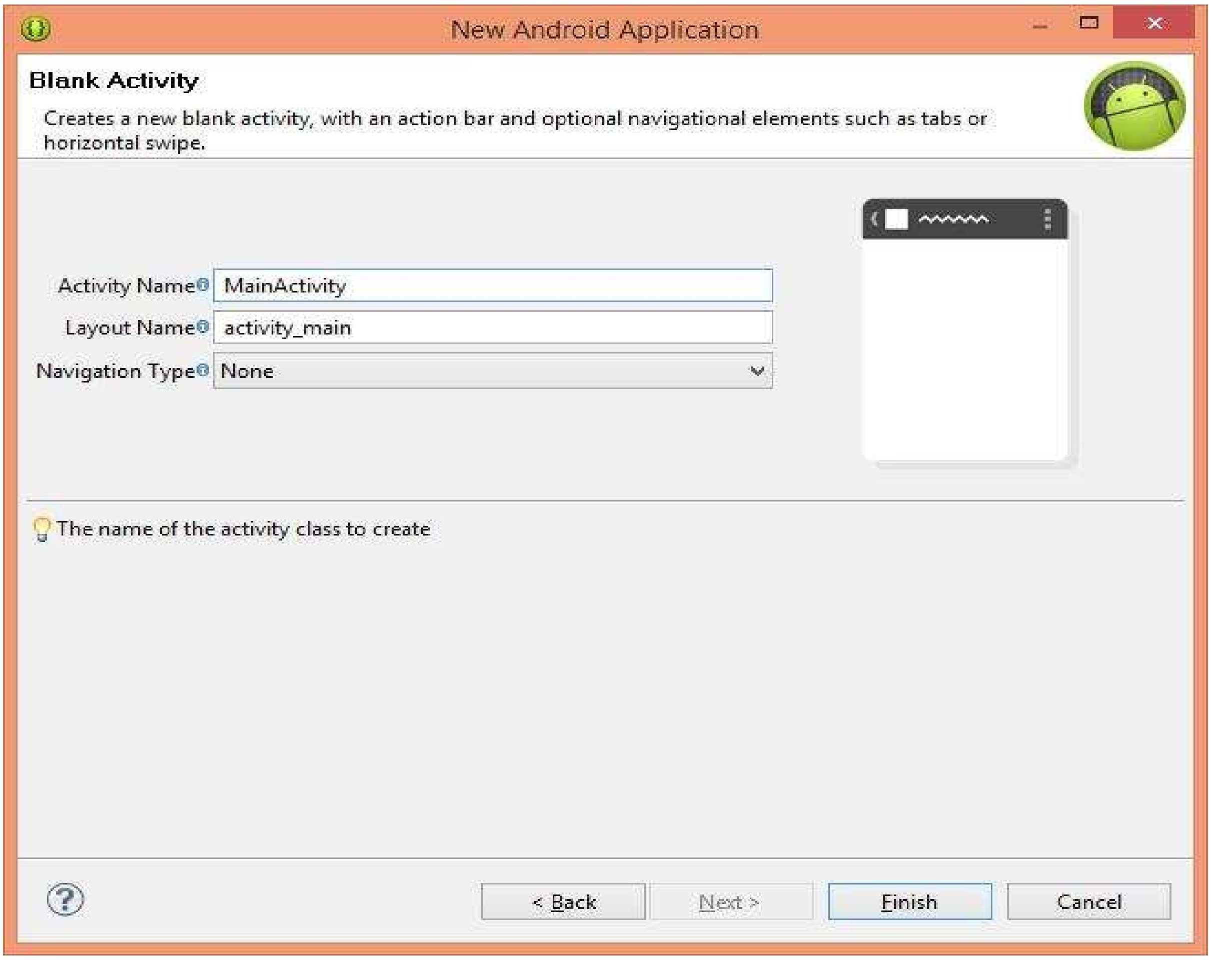
☑ Create Activity

- Blank Activity
- Fullscreen Activity
- Master/Detail Flow

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Step 4: On the next window, name the specify **Activity Name, Layout Name and Navigation Type,** click **Next**.



New Android Application

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Activity Name: MainActivity

Layout Name: activity_main

Navigation Type: None

The name of the activity class to create

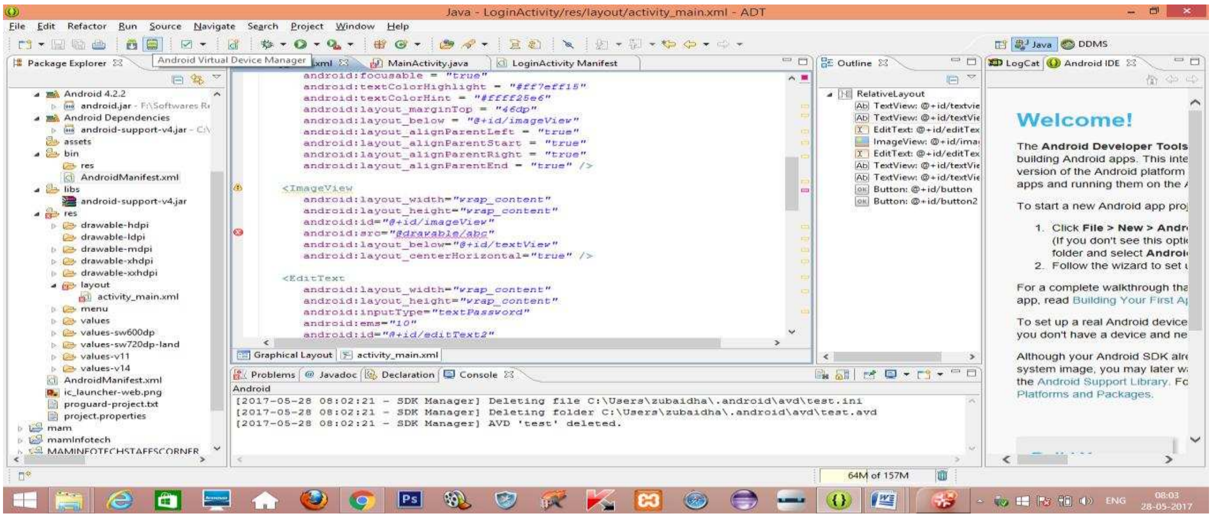
Step 5: Drag and Drop All the needed Components from the Palette Window to the **Design View of** the **Activity main.xml.**

Step 6: Create a new activity named **Success.xml,** and add some content.

Step 7: Code the operations that are want to perform in the **MainActivity.java, of** the preferred **.Xml** files.

Step 8: Create a new activity named **Success.xml,** and add some content for mobility.

Step 8: Create an **AVD, by** Clicking the **Android Virtual Device Icon, on** the **Toolbar.**

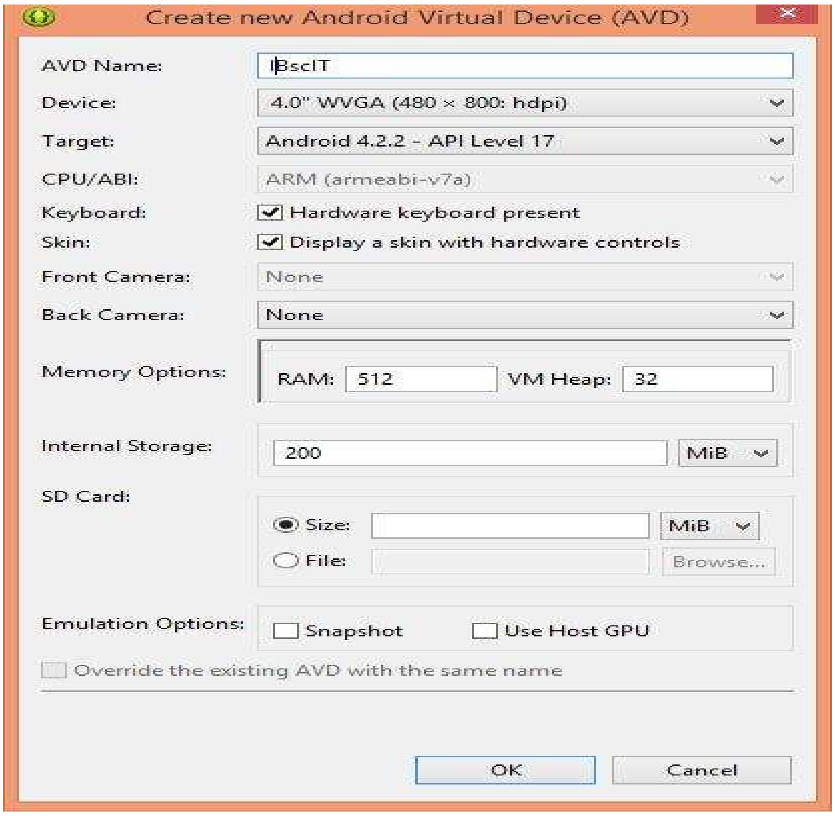


```
    android:focusable = "true"
    android:textColorHighlight = "#ff7eff15"
    android:textColorHint = "#ffff25e6"
    android:layout_marginTop = "46dp"
    android:layout_below = "@+id/imageView"
    android:layout_alignParentLeft = "true"
    android:layout_alignParentStart = "true"
    android:layout_alignParentRight = "true"
    android:layout_alignParentEnd = "true" />

<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/imageView"
    android:src="@drawable/abc"
    android:layout_below="@+id/textView"
    android:layout_centerHorizontal="true" />

<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    android:ems="10"
    android:id="@+id/editText2"
```

```
Android
[2017-05-28 08:02:21 - SDK Manager] Deleting file C:\Users\zubaidha\.android\avd\test.ini
[2017-05-28 08:02:21 - SDK Manager] Deleting folder C:\Users\zubaidha\.android\avd\test.avd
[2017-05-28 08:02:21 - SDK Manager] AVD 'test' deleted.
```

Step 8: On the window, select **New.**

Step 9: On the next window, specify **AVD Name, Device, Target** and **Other Details** and Click **OK.**



AVD Name: BscIT

Device: 4.0" WVGA (480 × 800: hdpi)

Target: Android 4.2.2 - API Level 17

CPU/ABI: ARM (armeabi-v7a)

Keyboard: ☑ Hardware keyboard present

Skin: ☑ Display a skin with hardware controls

Front Camera: None

Back Camera: None

Memory Options: RAM: 512 VM Heap: 32

Internal Storage: 200 MiB

SD Card: ● Size: MiB ○ File: Browse...

Emulation Options: □ Snapshot □ Use Host GPU

□ Override the existing AVD with the same name

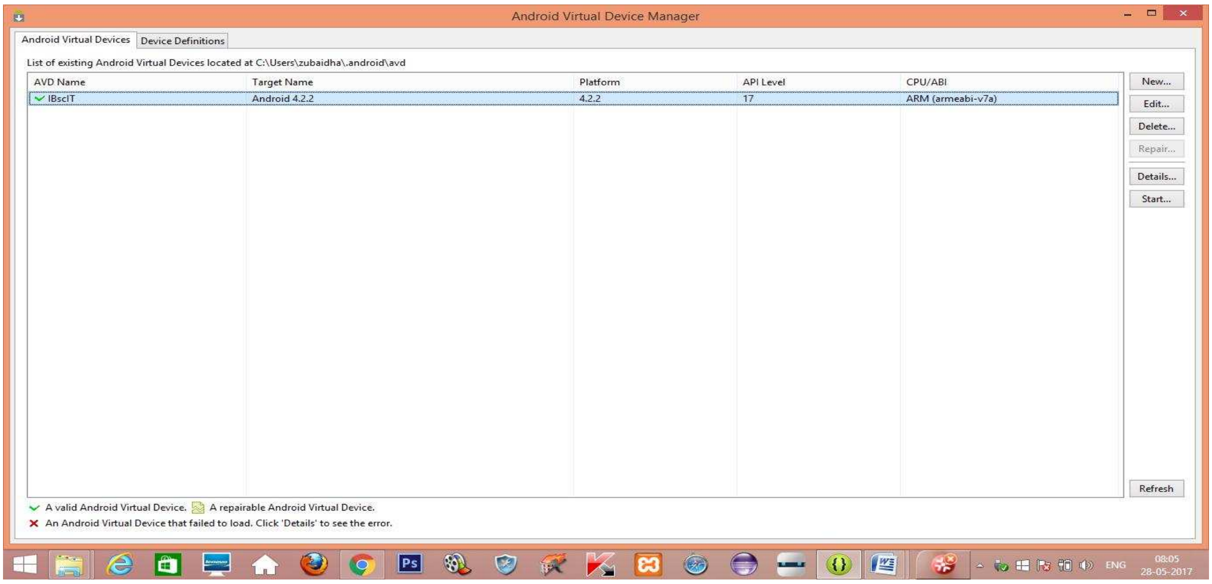
Step 10: Now a new **AVD** is created**.**

Step 11: On Completion of the **Project,** select **File ->Save All.**

Step 12: Select **Debug Icon** on the **Toolbar,** to debug the application.

Step 13: To run the application Select, **Run Icon** on the **Toolbar.**

Step 14: On the Android **Virtual Device Manager** window, select **AVD Name and**, Click **Start.**



Android Virtual Device Manager

List of existing Android Virtual Devices located at C:\Users\zubaidha\.android\avd

AVD Name	Target Name	Platform	API Level	CPU/ABI
✓ IBscIT	Android 4.2.2	4.2.2	17	ARM (armeabi-v7a)

✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✕ An Android Virtual Device that failed to load. Click 'Details' to see the error.

Step 15: On the Launch **Options** window, Check Wipe **User Data and**, Click **Launch.**

Program:

Activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

       <TextView
           android:id="@+id/textview"
           android:layout_width="wrap_content"
           android:layout_height="wrap_content"
           android:layout_alignParentTop="true"
           android:layout_centerHorizontal="true"
           android:text="Login Screen"
           android:textSize="35dp" />

   <TextView
       android:id="@+id/textView"
       android:layout_width="wrap_content"
       android:layout_height="wrap_content"
       android:layout_below="@+id/textview"
       android:layout_centerHorizontal="true"
       android:text="II B.sc IT"
       android:textColor="#ff7aff24"
       android:textSize="35dp" />

   <EditText
      android:layout_width = "wrap_content"
      android:layout_height = "wrap_content"
      android:id = "@+id/editText"
      android:hint = "Enter Name"
      android:focusable = "true"
      android:textColorHighlight = "#ff7eff15"
      android:textColorHint = "#ffff25e6"
      android:layout_marginTop = "46dp"
      android:layout_below = "@+id/imageView"
      android:layout_alignParentLeft = "true"
      android:layout_alignParentStart = "true"
      android:layout_alignParentRight = "true"
      android:layout_alignParentEnd = "true" />

   <ImageView
       android:id="@+id/imageView"
       android:layout_width="wrap_content"
       android:layout_height="wrap_content"
       android:layout_below="@+id/textView"
```

```
        android:layout_centerHorizontal="true"
        android:src="@drawable/ic_launcher" />

    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:ems="10"
        android:id="@+id/editText2"
        android:layout_below="@+id/editText"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_alignRight="@+id/editText"
        android:layout_alignEnd="@+id/editText"
        android:textColorHint="#ffff299f"
        android:hint="Password" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Attempts Left:"
        android:id="@+id/textView2"
        android:layout_below="@+id/editText2"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:textSize="25dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="New Text"
        android:id="@+id/textView3"
        android:layout_alignTop="@+id/textView2"
        android:layout_alignParentRight="true"
        android:layout_alignParentEnd="true"
        android:layout_alignBottom="@+id/textView2"
        android:layout_toEndOf="@+id/textview"
        android:textSize="25dp"
        android:layout_toRightOf="@+id/textview" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="login"
        android:id="@+id/button"
        android:layout_alignParentBottom="true"
        android:layout_toLeftOf="@+id/textview"
        android:layout_toStartOf="@+id/textview" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Cancel"
        android:id="@+id/button2"
        android:layout_alignParentBottom="true"
        android:layout_toRightOf="@+id/textview"
        android:layout_toEndOf="@+id/textview" />

    </RelativeLayout>
```

MainActivity.java

```
package com.example.loginactivity;

import android.app.Activity;
import android.graphics.Color;
import android.os.Bundle;
import android.view.View;

import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;


public class MainActivity extends Activity {
  Button b1,b2;
  EditText ed1,ed2;

  TextView tx1;
  int counter = 3;

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

    b1 = (Button)findViewById(R.id.button);
    ed1 = (EditText)findViewById(R.id.editText);
    ed2 = (EditText)findViewById(R.id.editText2);

    b2 = (Button)findViewById(R.id.button2);
    tx1 = (TextView)findViewById(R.id.textView3);
    tx1.setVisibility(View.GONE);

    b1.setOnClickListener(new View.OnClickListener()
      { @Override
      public void onClick(View v) {
        if(ed1.getText().toString().equals("bscit") &&
          ed2.getText().toString().equals("bscit")) {
            Toast.makeText(getApplicationContext(),
              "Redirecting...",Toast.LENGTH_SHORT).show();
          }else{
Toast.makeText(getApplicationContext(), "Wrong
Credentials",Toast.LENGTH_SHORT).show();
```

```
            tx1.setVisibility(View.VISIBLE);
            tx1.setBackgroundColor(Color.RED);
            counter--;
            tx1.setText(Integer.toString(counter));

            if (counter == 0) {
              b1.setEnabled(false);
            }
          }
      }
    });

    b2.setOnClickListener(new View.OnClickListener()
      { @Override
      public void onClick(View v) {
        finish();
      }
    });
  }
}
```

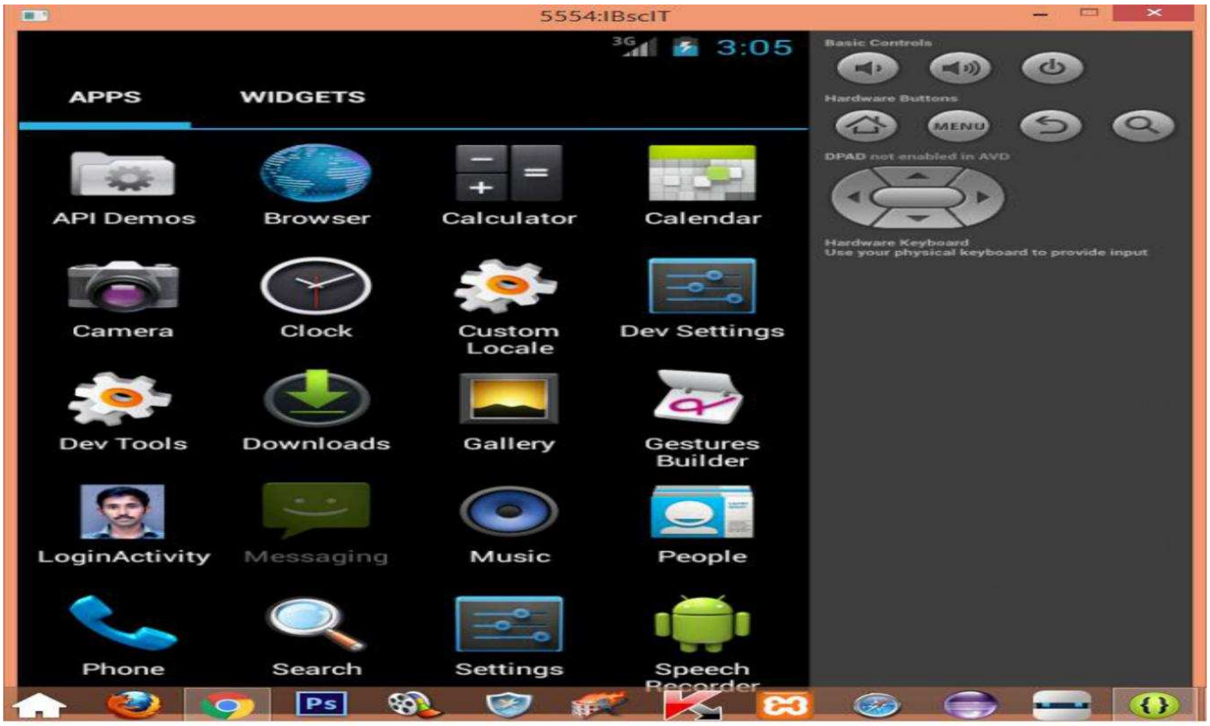
Android manifest.xml

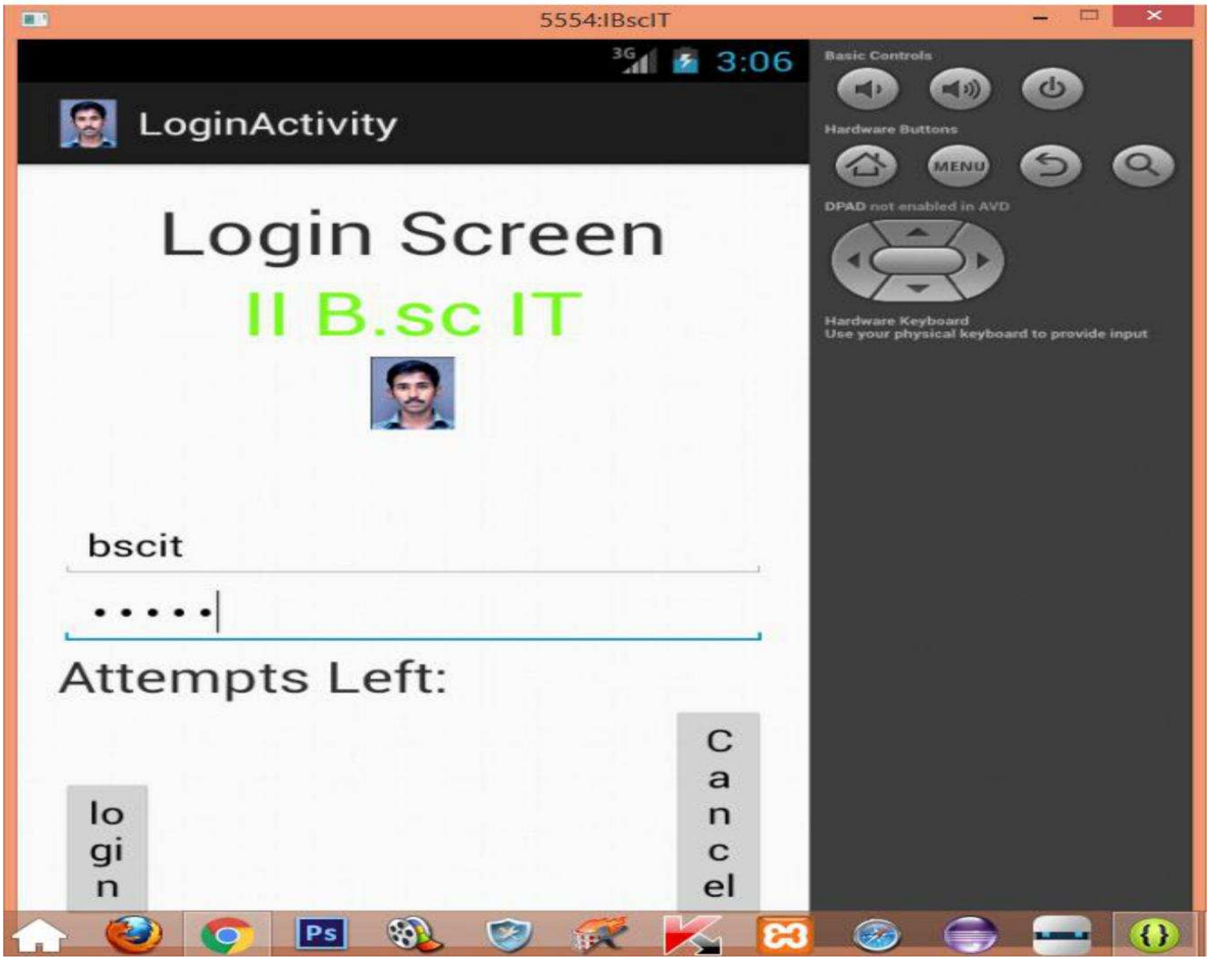
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.loginactivity"
   android:versionCode="1"
   android:versionName="1.0" >

   <uses-sdk
      android:minSdkVersion="8"
      android:targetSdkVersion="17" />

   <application
      android:allowBackup="true"
      android:icon="@drawable/ic_launcher"
      android:label="@string/app_name"
      android:theme="@style/AppTheme" >
      <activity
         android:name="com.example.loginactivity.MainActivity"
         android:label="@string/app_name" >
         <intent-filter>
            <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

Strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">LoginActivity</string>
    <string name="action_settings">Settings</string>
    <string name="hello_world">Hello world!</string>
</resources>
```

Output:





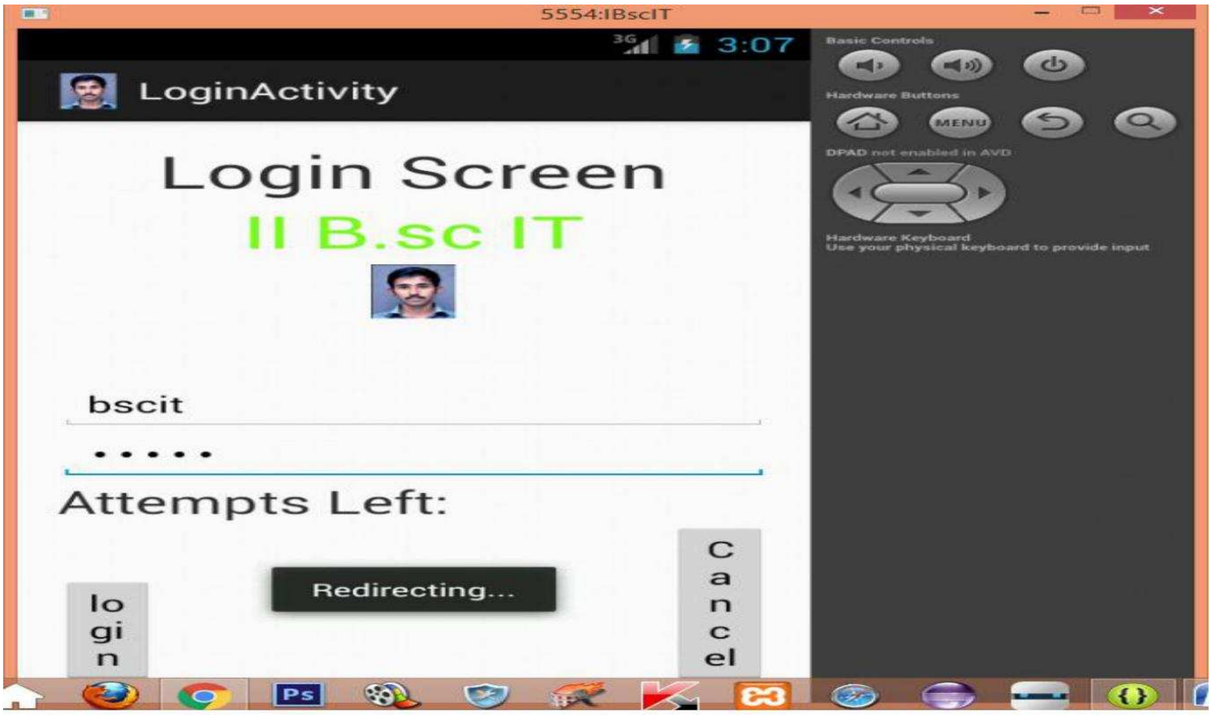
Login Screen

II B.sc IT

bscit

•••••

Attempts Left:



Login Screen

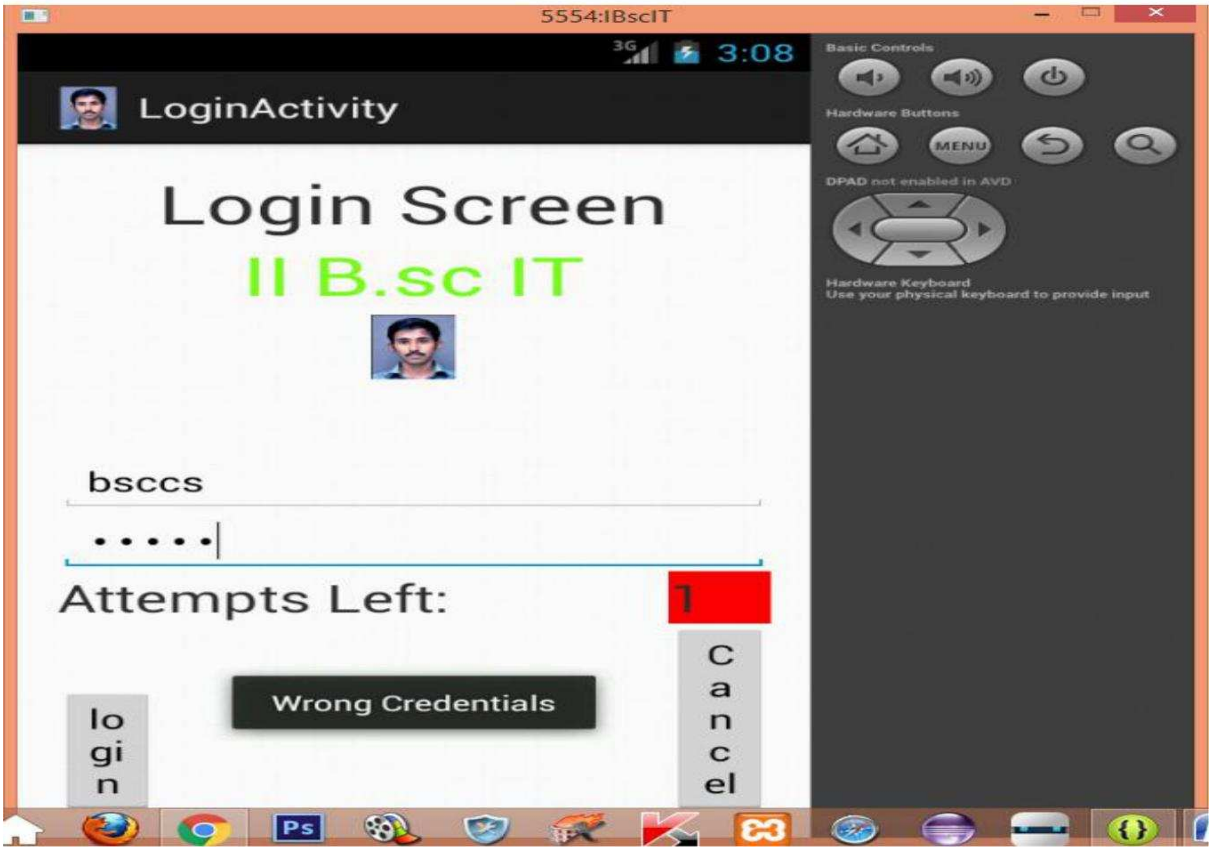
II B.sc IT

bscit

•••••

Attempts Left:

Redirecting...



Login Screen

II B.sc IT

bsccs

•••••

Attempts Left: 1

Wrong Credentials

Result:

The above aim of the program has been achieved successfully.

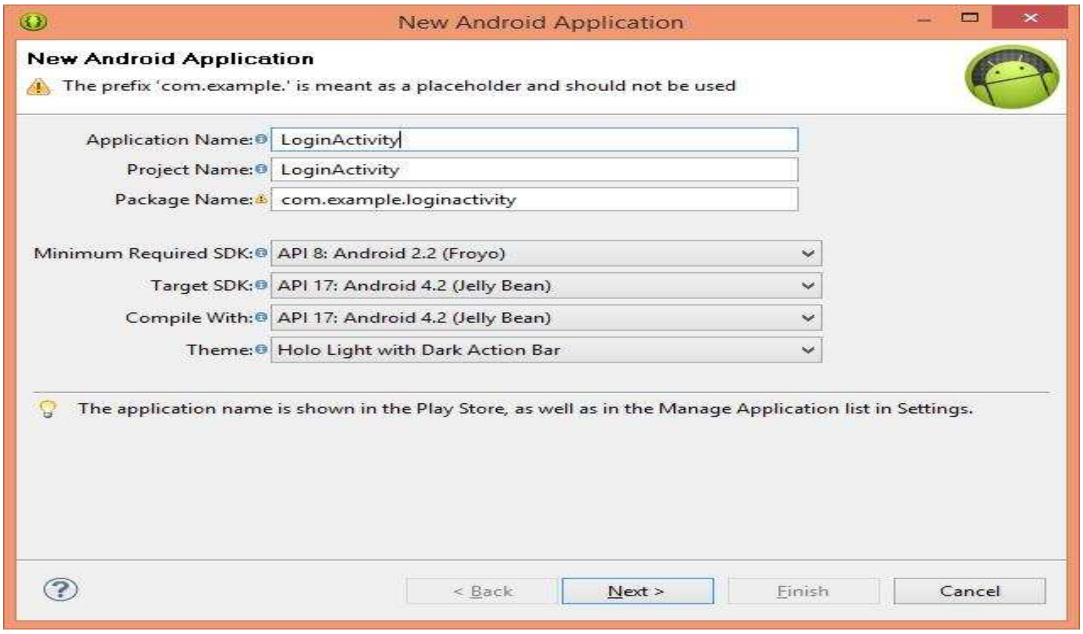
3.HELLO WORLD WITH BASIC INTERACTIONS

Aim:

To create an android application with login operations.

Procedure:

Step 1: Open Eclipse IDE and go to **File -> New** -> **Project** -> **Android** -> **Android Application Project**. You have to specify the **Application Name**, the **Project Name** and the **Package name** in the appropriate text fields and then click **Next**.



New Android Application

New Android Application

The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: LoginActivity

Project Name: LoginActivity

Package Name: com.example.loginactivity

Minimum Required SDK: API 8: Android 2.2 (Froyo)

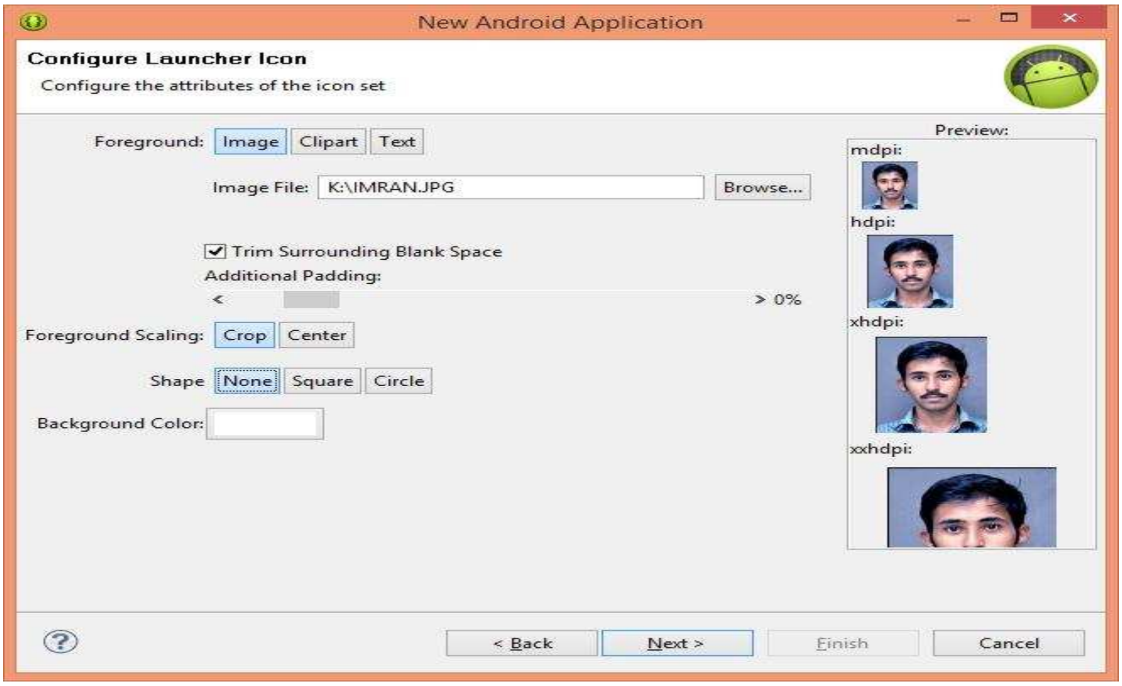
Target SDK: API 17: Android 4.2 (Jelly Bean)

Compile With: API 17: Android 4.2 (Jelly Bean)

Theme: Holo Light with Dark Action Bar

The application name is shown in the Play Store, as well as in the Manage Application list in Settings.

Step 2: In the next window make sure the select the **Application Launcher Icon** option is selected in order to create a new activity for your project, and click **Next**.



Configure Launcher Icon

Configure the attributes of the icon set

Foreground: Image | Clipart | Text

Image File: K:\IMRAN.JPG

☑ Trim Surrounding Blank Space

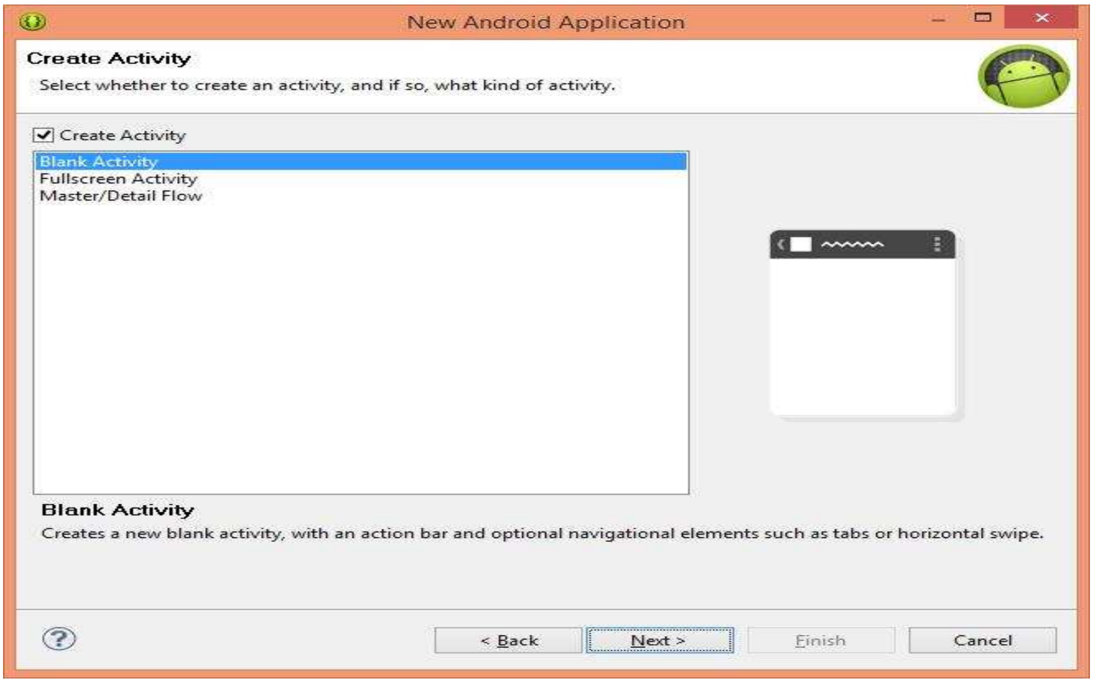
Additional Padding: 0%

Foreground Scaling: Crop | Center

Shape: None | Square | Circle

Background Color:

Step 3: On the next window click to select the **Blank Activity** and click **Next**.



Create Activity

Select whether to create an activity, and if so, what kind of activity.

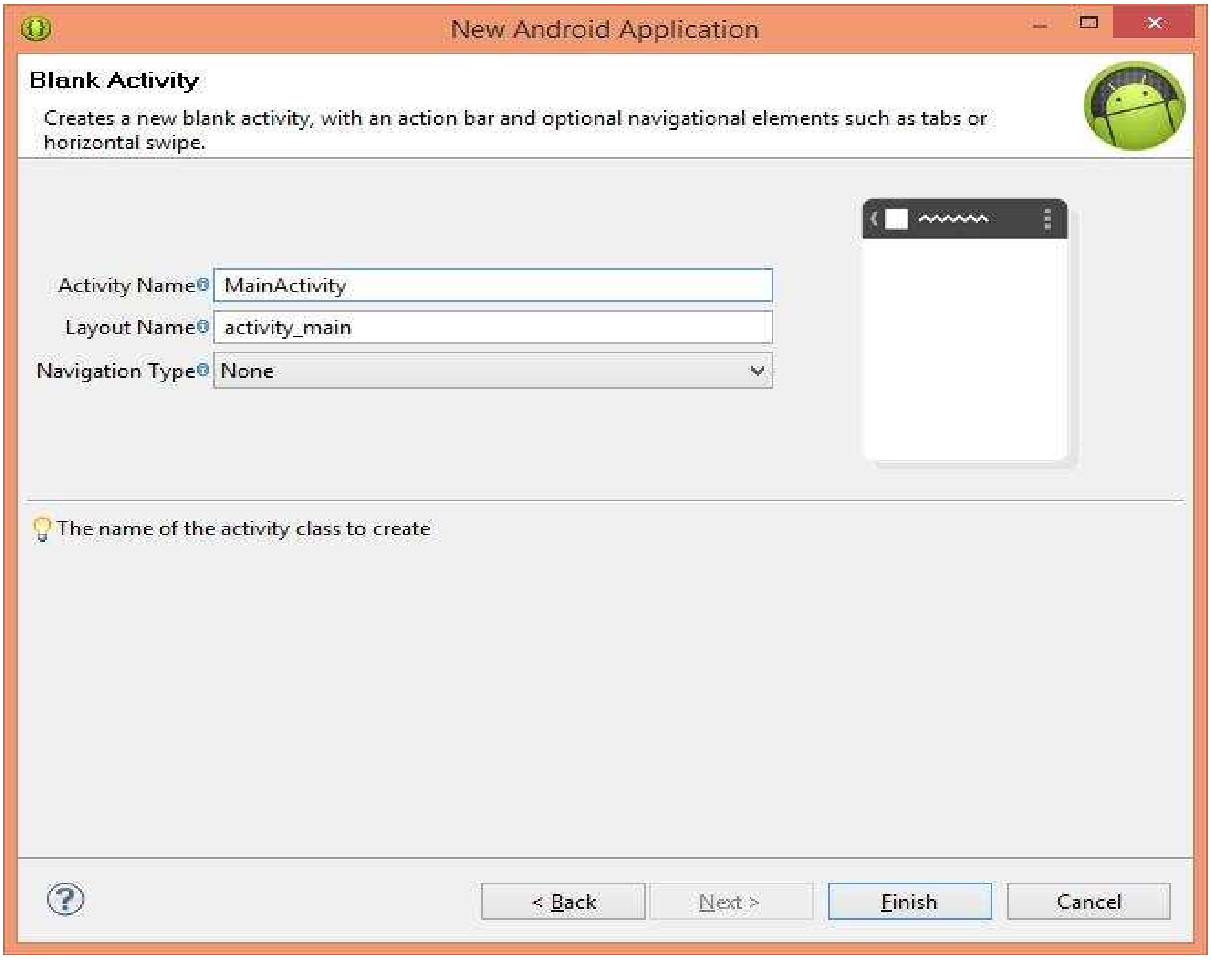
☑ Create Activity

- Blank Activity
- Fullscreen Activity
- Master/Detail Flow

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Step 4: On the next window, name the specify **Activity Name, Layout Name and Navigation Type,** click **Next**.



Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Activity Name: MainActivity

Layout Name: activity_main

Navigation Type: None

The name of the activity class to create

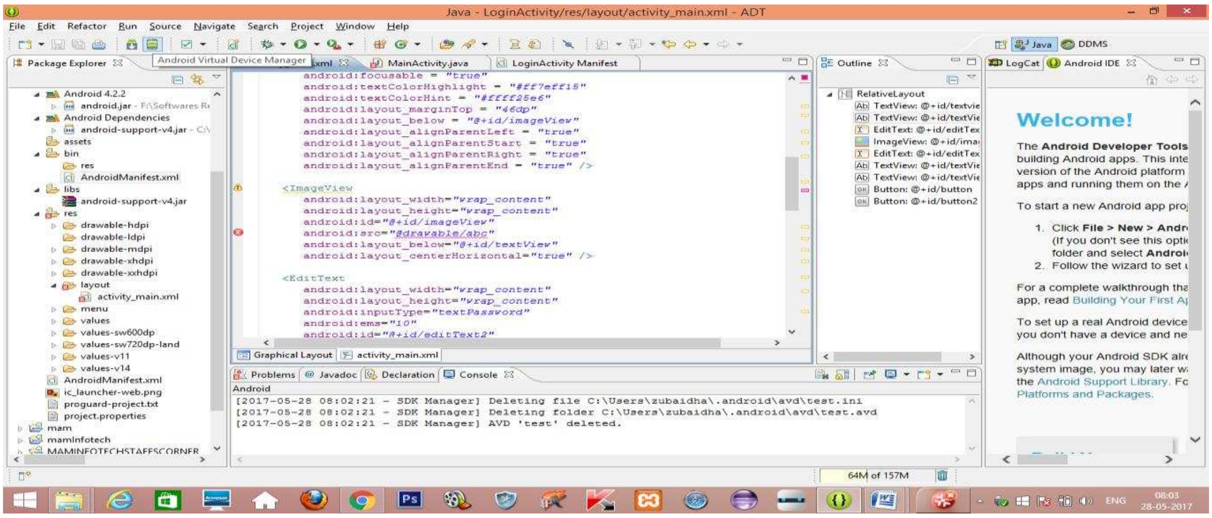
Step 5: Drag and Drop All the needed Components from the Palette Window to the **Design View of** the **Activity main.xml.**

Step 6: Create a new activity named **Success.xml,** and add some content.

Step 7: Code the operations that are want to perform in the **MainActivity.java, of** the preferred **.Xml** files.

Step 8: Create a new activity named **Success.xml,** and add some content for mobility.

Step 8: Create an **AVD, by** Clicking the **Android Virtual Device Icon, on** the **Toolbar.**

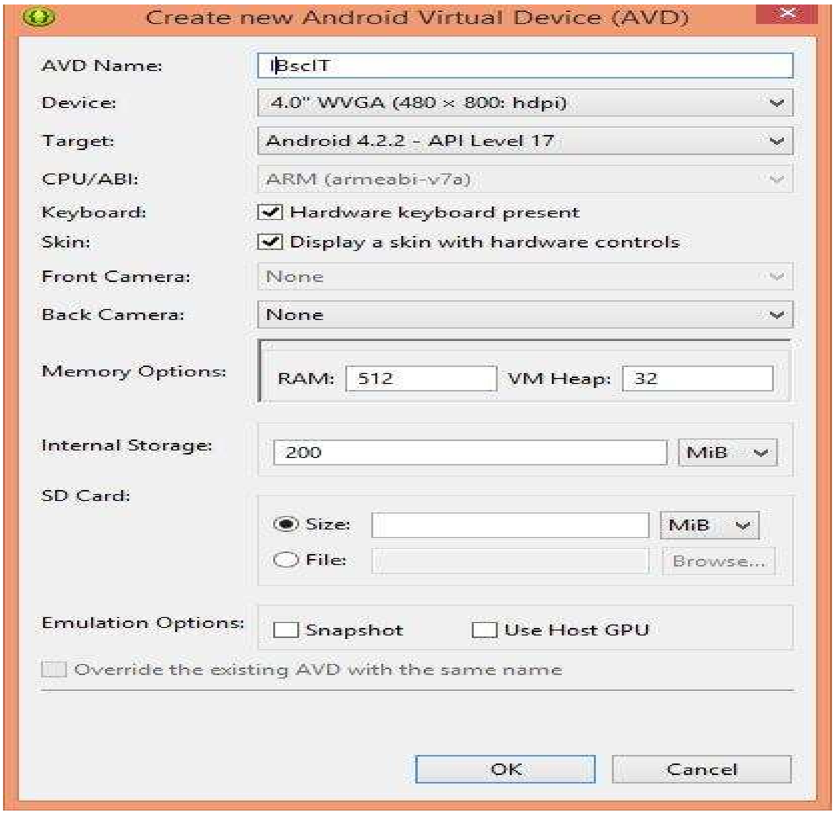


```
        android:focusable = "true"
        android:textColorHighlight = "#ff7eff15"
        android:textColorHint = "#ffff25e6"
        android:layout_marginTop = "46dp"
        android:layout_below = "@+id/imageView"
        android:layout_alignParentLeft = "true"
        android:layout_alignParentStart = "true"
        android:layout_alignParentRight = "true"
        android:layout_alignParentEnd = "true" />

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imageView"
        android:src="@drawable/abc"
        android:layout_below="@+id/textView"
        android:layout_centerHorizontal="true" />

    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:ems="10"
        android:id="@+id/editText2"
```

```
Android
[2017-05-28 08:02:21 - SDK Manager] Deleting file C:\Users\zubaidha\.android\avd\test.ini
[2017-05-28 08:02:21 - SDK Manager] Deleting folder C:\Users\zubaidha\.android\avd\test.avd
[2017-05-28 08:02:21 - SDK Manager] AVD 'test' deleted.
```

Step 8: On the window, select **New.**

Step 9: On the next window, specify **AVD Name, Device, Target** and **Other Details** and Click **OK.**



AVD Name: BscIT

Device: 4.0" WVGA (480 × 800: hdpi)

Target: Android 4.2.2 - API Level 17

CPU/ABI: ARM (armeabi-v7a)

Keyboard: ☑ Hardware keyboard present

Skin: ☑ Display a skin with hardware controls

Front Camera: None

Back Camera: None

Memory Options: RAM: 512 VM Heap: 32

Internal Storage: 200 MiB

SD Card: ● Size: MiB ○ File: Browse...

Emulation Options: □ Snapshot □ Use Host GPU

□ Override the existing AVD with the same name

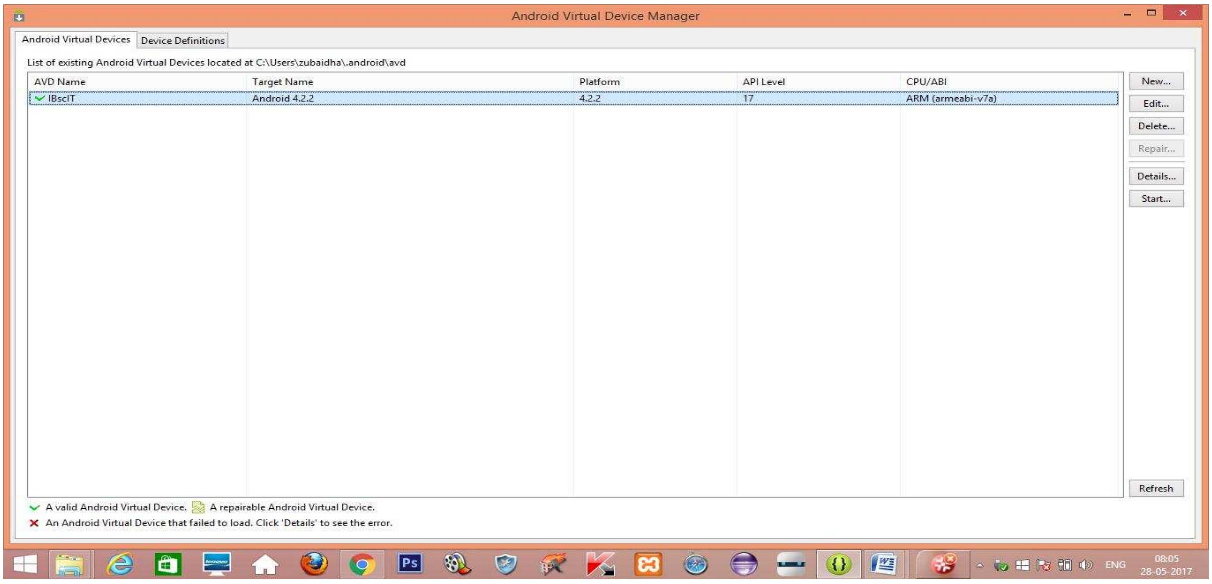
Step 10: Now a new **AVD** is created**.**

Step 11: On Completion of the **Project,** select **File ->Save All.**

Step 12: Select **Debug Icon** on the **Toolbar,** to debug the application.

Step 13: To run the application Select, **Run Icon** on the **Toolbar.**

Step 14: On the Android **Virtual Device Manager** window, select **AVD Name and**, Click **Start.**



Android Virtual Device Manager

List of existing Android Virtual Devices located at C:\Users\zubaidha\.android\avd

AVD Name	Target Name	Platform	API Level	CPU/ABI
✓ IBscIT	Android 4.2.2	4.2.2	17	ARM (armeabi-v7a)

✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

Step 15: On the Launch **Options** window, Check Wipe **User Data and**, Click **Launch.**

Program:

Activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/bu"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

     <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world" />

    <ListView
        android:id="@+id/mobile_list"
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_above="@+id/b1"
        android:layout_alignLeft="@+id/textView1"
        android:layout_marginBottom="26dp" >
    </ListView>

    <Button
        android:id="@+id/b1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_marginBottom="68dp"
        android:layout_toRightOf="@+id/textView"
        android:text="Next" />

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/mobile_list"
        android:layout_alignRight="@+id/textView"
        android:layout_marginRight="29dp"
        android:src="@drawable/ic_launcher" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/imageView"
        android:layout_centerHorizontal="true"
        android:text="II B.sc IT"
        android:textColor="#ff7aff24"
        android:textSize="35dp" />

    <TextView
        android:id="@+id/textview2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/textView"
        android:layout_alignRight="@+id/mobile_list"
        android:layout_marginBottom="14dp"
        android:text="Basic Interactions"
        android:textSize="35dp" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.helloworld;


import android.os.Bundle;

import android.app.Activity;

import android.view.Menu;

import android.widget.ArrayAdapter;

import android.widget.ListView;


public class MainActivity extends Activity {


            private Button next;

             String[] mobileArray = {"Bsc IT","BSc CS"};


             @Override

               protected void onCreate(Bundle savedInstanceState)

                  { super.onCreate(savedInstanceState);

                  setContentView(R.layout.activity_main);

                  //next = (Button)findViewById(R.id.b1);

                  //addListenerOnButton();



                  ArrayAdapter adapter = new ArrayAdapter<String>(this,
               R.layout.activity_listview, mobileArray);

             ListView listView = (ListView) findViewById(R.id.mobile_list);
             listView.setAdapter(adapter);

           }

         }
```

Activity_main2.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".Main2Activity" >

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world" />

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textView2"
        android:layout_marginLeft="58dp"
        android:layout_marginTop="202dp"
        android:layout_toRightOf="@+id/textView2"
        android:hint="Text Box"
        android:text="II BSC IT" />

</RelativeLayout>
```

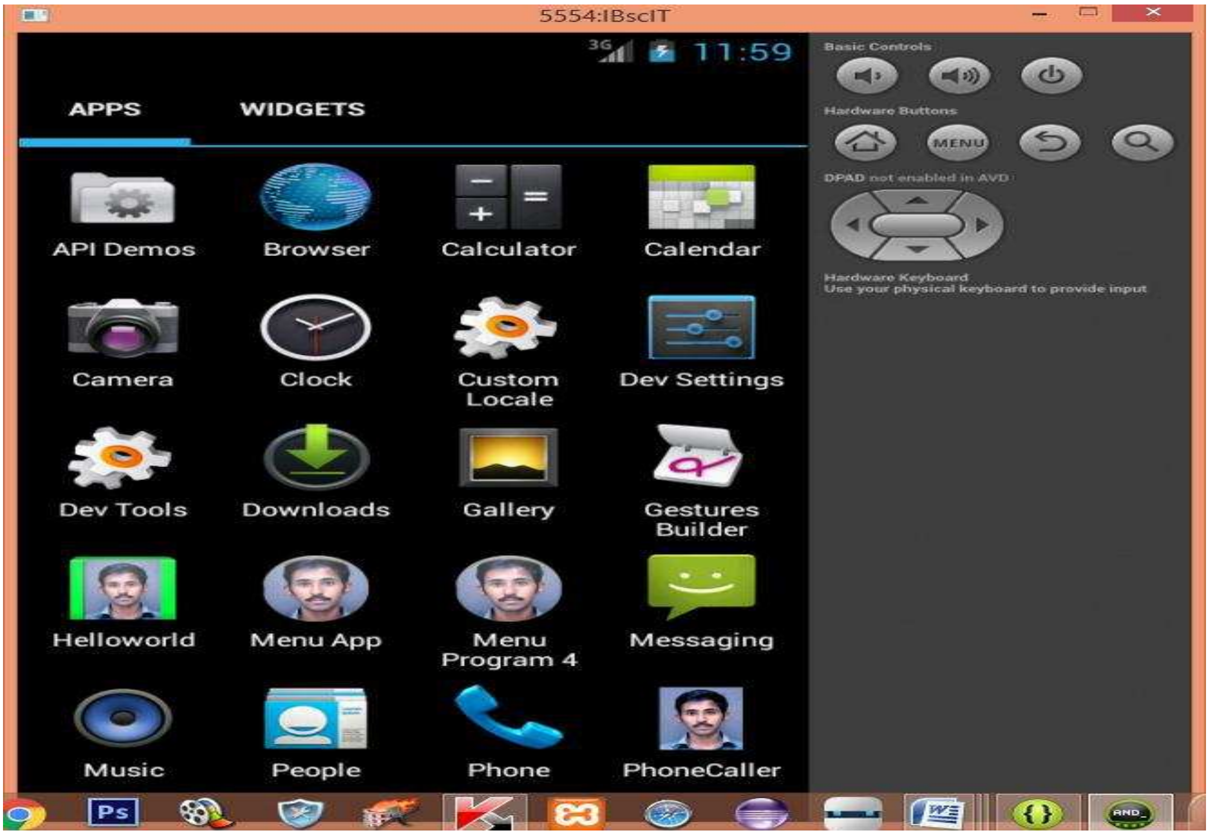
Main2Activity.java

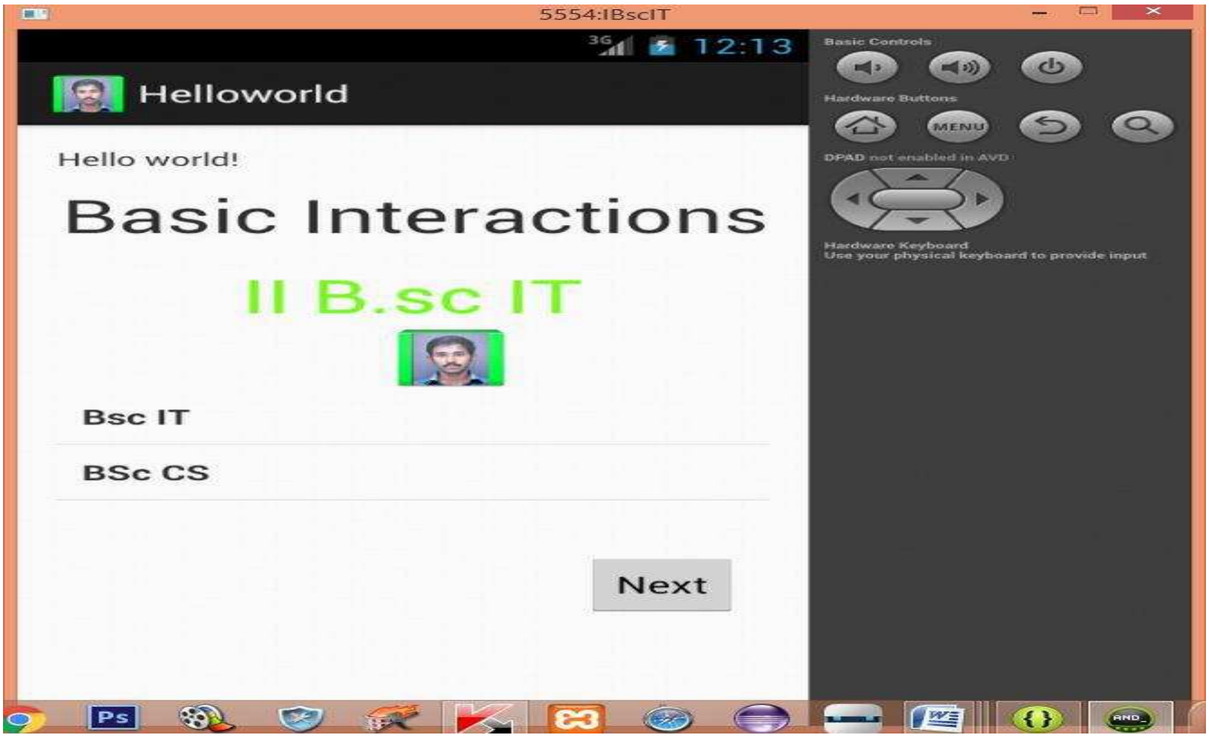
```
package com.example.helloworld;

import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.view.View;
import android.app.Activity;
import android.view.Menu;

public class Main2Activity extends Activity {

             DemoView demoview;

            @Override
    public void onCreate(Bundle savedInstanceState)
        { super.onCreate(savedInstanceState);
        demoview = new DemoView(this);
        setContentView(demoview);
    }

              private class DemoView extends View{
                    public DemoView(Context context){
                        super(context);
```

```
}

@Override protected void onDraw(Canvas canvas)
   { super.onDraw(canvas);

   // custom drawing code here
   Paint paint = new Paint();
   paint.setStyle(Paint.Style.FILL);

   // make the entire canvas white
   paint.setColor(Color.WHITE);
   canvas.drawPaint(paint);

   // draw blue circle with anti aliasing turned
   off paint.setAntiAlias(false);
   paint.setColor(Color.BLUE);
   canvas.drawCircle(20, 20, 15, paint);

   // draw green circle with anti aliasing turned
   on paint.setAntiAlias(true);
   paint.setColor(Color.GREEN);
   canvas.drawCircle(60, 20, 15, paint);

   // draw red rectangle with anti aliasing turned
   off paint.setAntiAlias(false);
   paint.setColor(Color.RED);
   canvas.drawRect(100, 5, 200, 30, paint);

   // draw the rotated text
   canvas.rotate(-45);

   paint.setStyle(Paint.Style.FILL);
   canvas.drawText("Graphics Rotation", 40, 180, paint);

   //undo the rotate
```

```
            canvas.restore();
        }
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
            // Inflate the menu; this adds items to the action bar if it is
            present. getMenuInflater().inflate(R.menu.main2, menu);
            return true;
    }

}
```

Output:





Helloworld

Hello world!

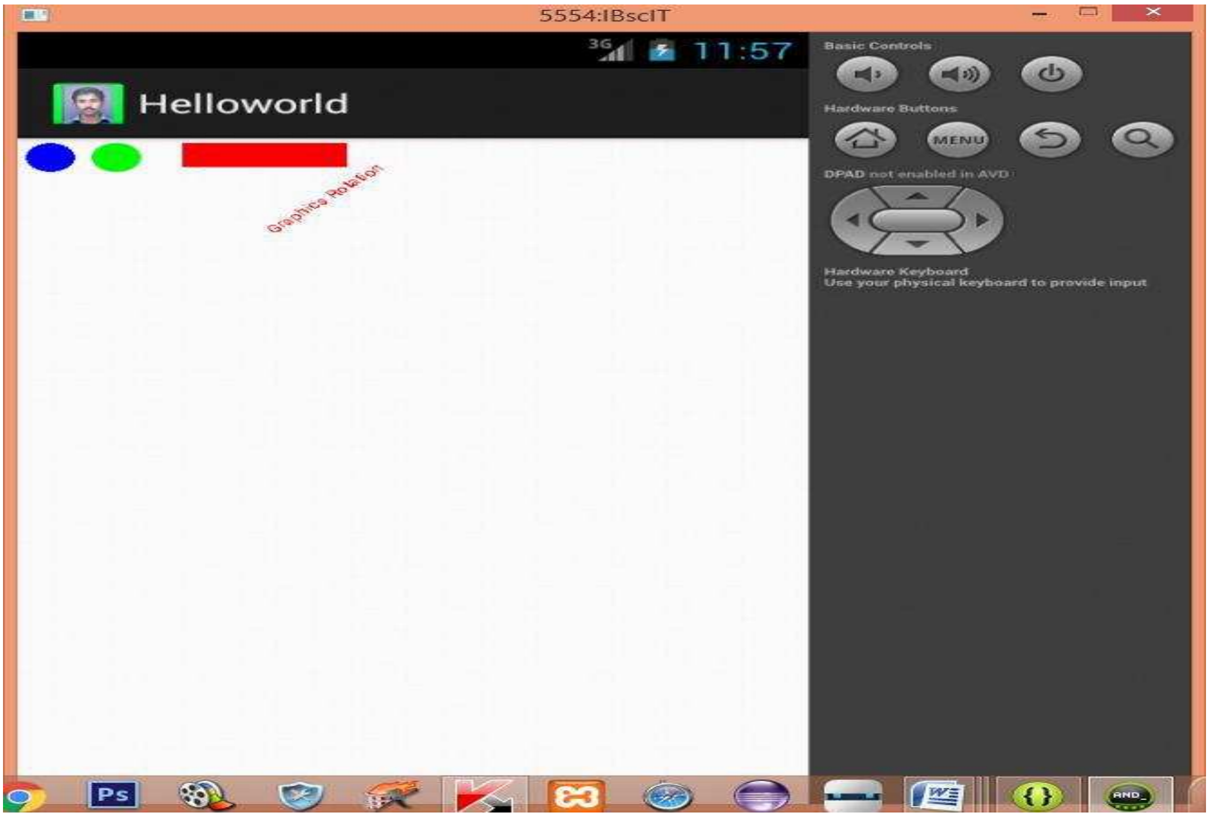
Basic Interactions

II B.sc IT

Bsc IT

BSc CS

Next



Result:

The above aim of the program has been achieved successfully.

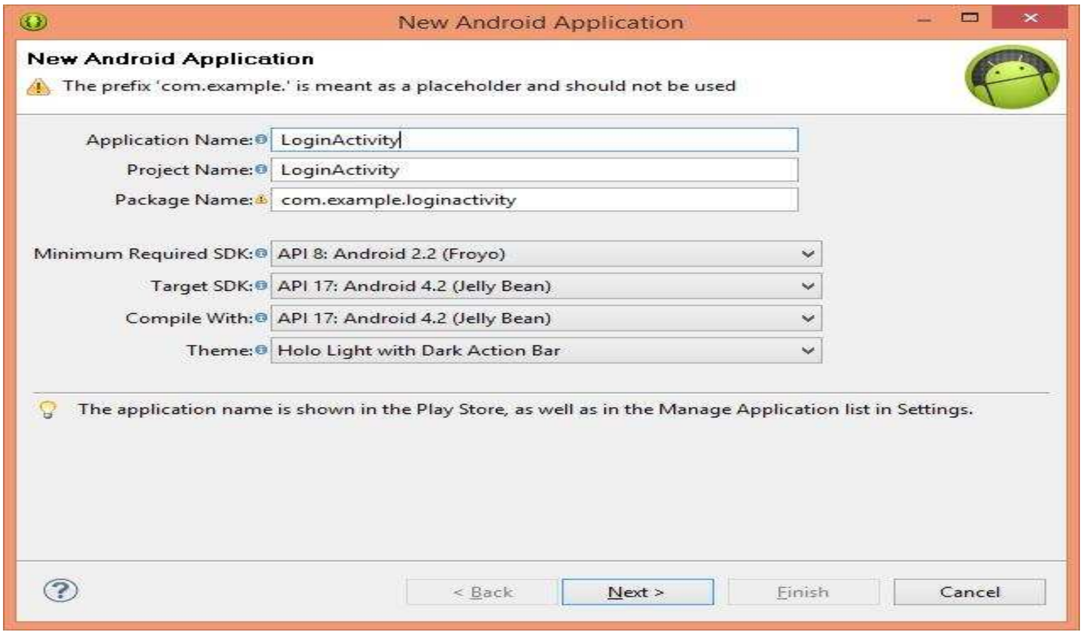
4.MENU ACTIVITY

Aim:

To create an android application with menu functions.

Procedure:

Step 1: Open Eclipse IDE and go to **File -> New** -> **Project** -> **Android** -> **Android Application Project**. You have to specify the **Application Name**, the **Project Name** and the **Package name** in the appropriate text fields and then click **Next**.



New Android Application

The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: LoginActivity

Project Name: LoginActivity

Package Name: com.example.loginactivity

Minimum Required SDK: API 8: Android 2.2 (Froyo)

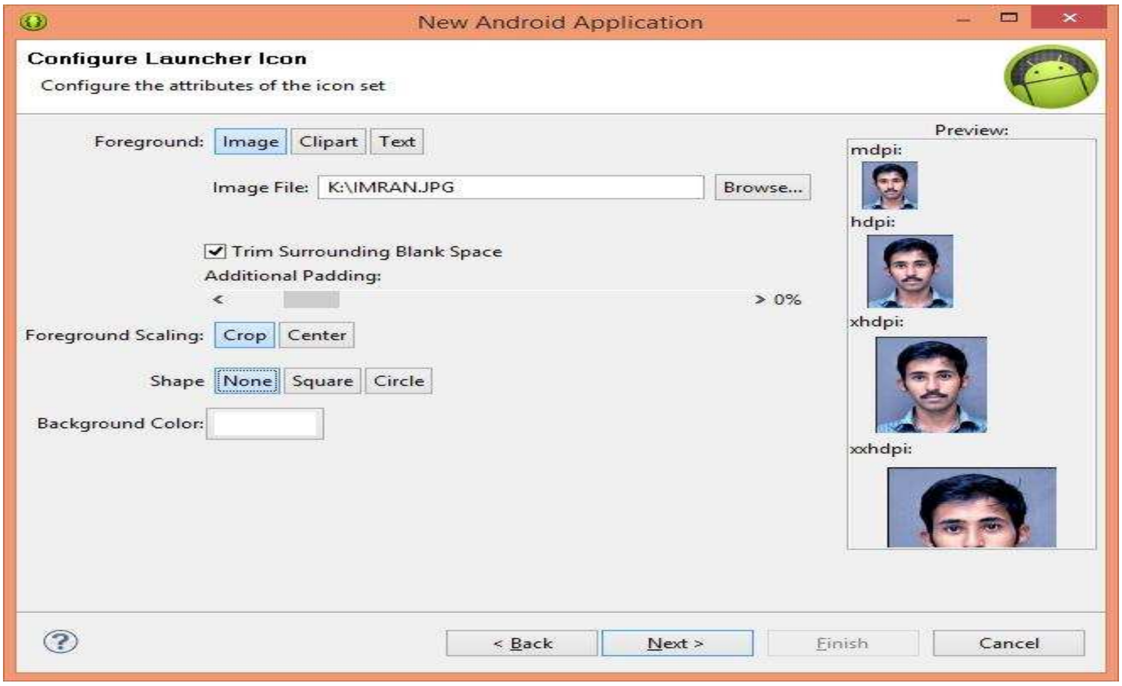
Target SDK: API 17: Android 4.2 (Jelly Bean)

Compile With: API 17: Android 4.2 (Jelly Bean)

Theme: Holo Light with Dark Action Bar

The application name is shown in the Play Store, as well as in the Manage Application list in Settings.

Step 2: In the next window make sure the select the **Application Launcher Icon** option is selected in order to create a new activity for your project, and click **Next**.



Configure Launcher Icon

Configure the attributes of the icon set

Foreground: Image | Clipart | Text

Image File: K:\IMRAN.JPG

☑ Trim Surrounding Blank Space

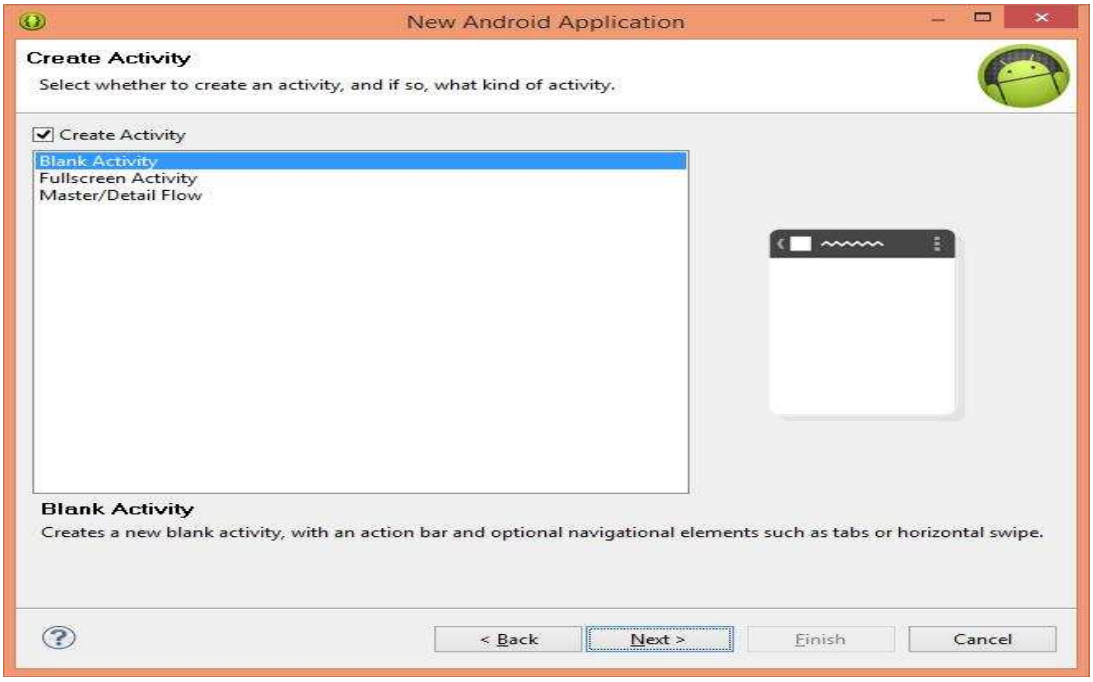
Additional Padding: 0%

Foreground Scaling: Crop | Center

Shape: None | Square | Circle

Background Color:

Step 3: On the next window click to select the **Blank Activity** and click **Next**.



New Android Application

Create Activity

Select whether to create an activity, and if so, what kind of activity.

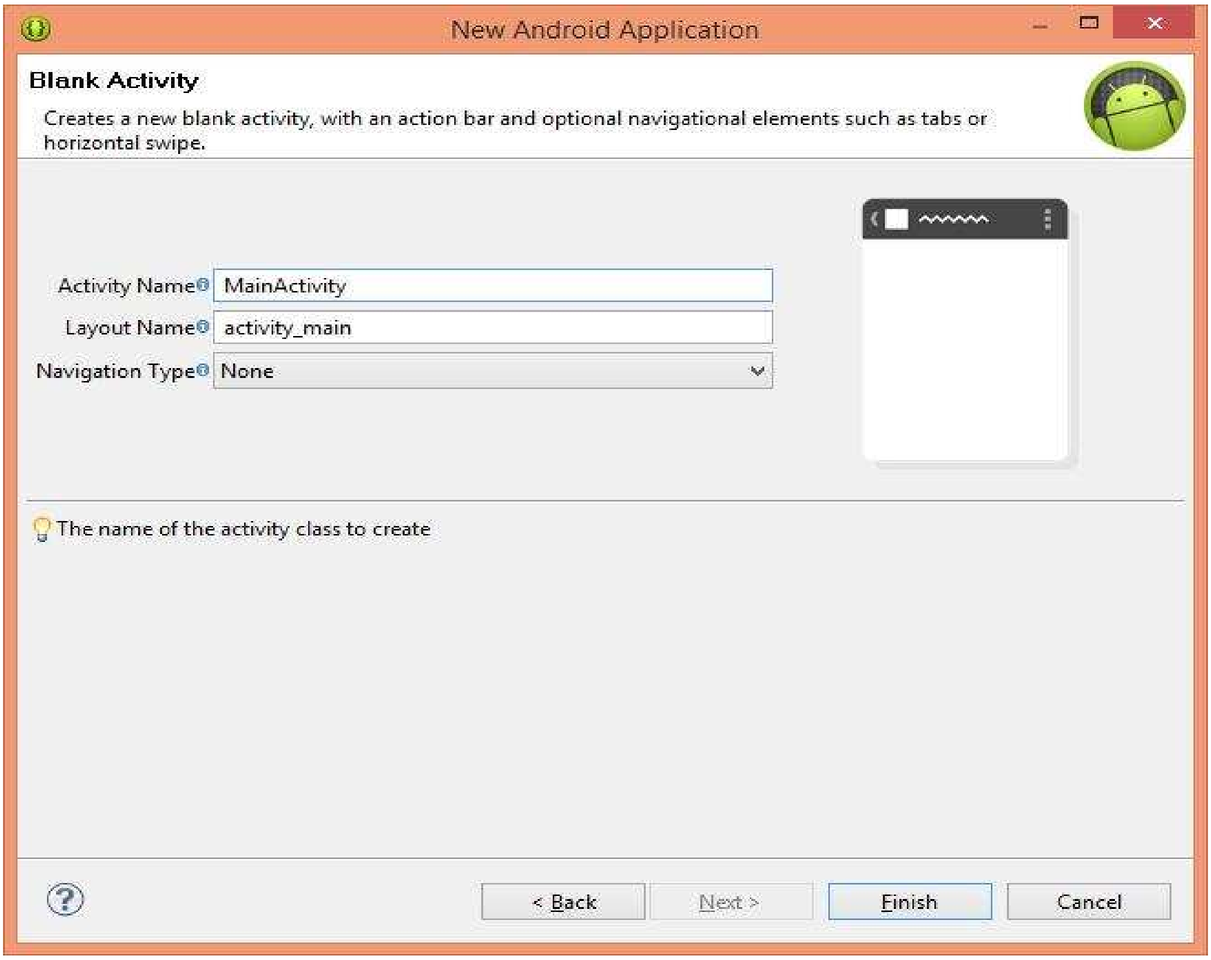
☑ Create Activity

- Blank Activity
- Fullscreen Activity
- Master/Detail Flow

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Step 4: On the next window, name the specify **Activity Name, Layout Name and Navigation Type,** click **Next**.



New Android Application

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Activity Name: MainActivity

Layout Name: activity_main

Navigation Type: None

The name of the activity class to create

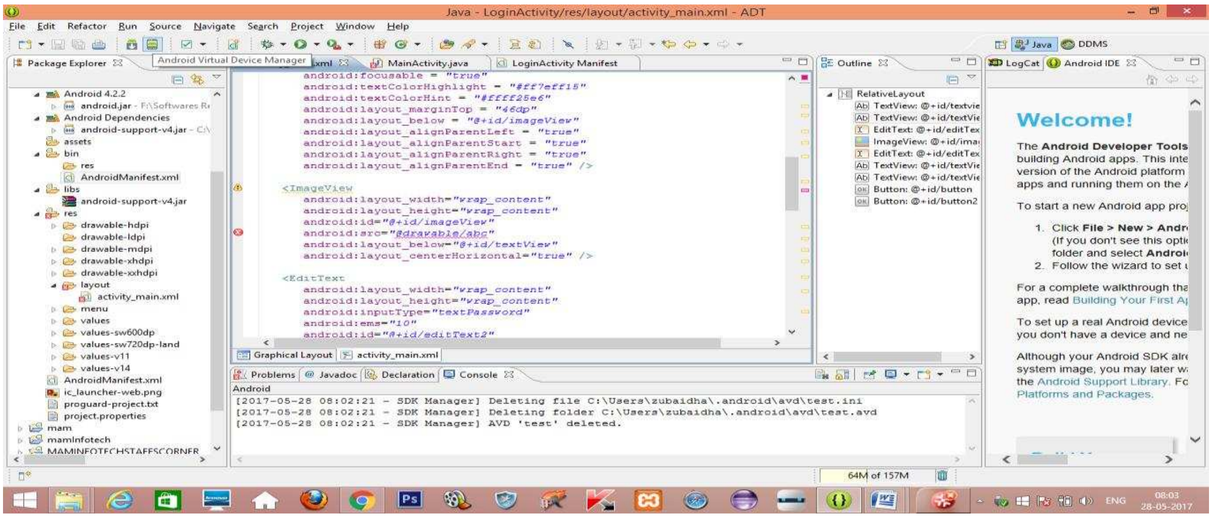
Step 5: Drag and Drop All the needed Components from the Palette Window to the **Design View of** the **Activity main.xml.**

Step 6: Create a new activity named **Success.xml,** and add some content.

Step 7: Code the operations that are want to perform in the **MainActivity.java, of** the preferred **.Xml** files.

Step 8: Create a new activity named **Success.xml,** and add some content for mobility.

Step 8: Create an **AVD, by** Clicking the **Android Virtual Device Icon, on** the **Toolbar.**



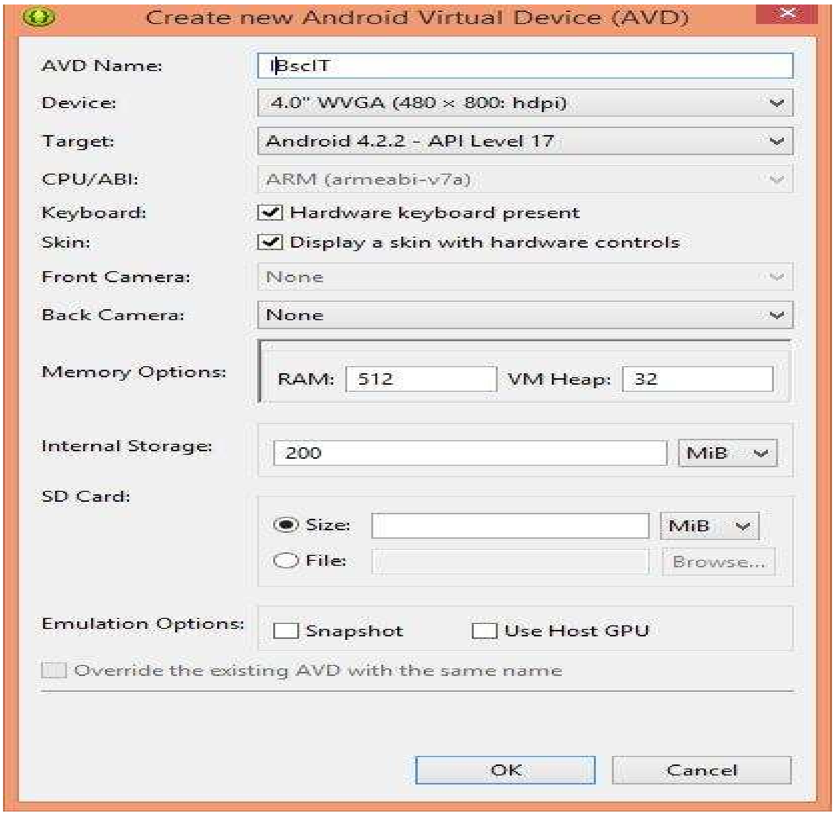
Java - LoginActivity/res/layout/activity_main.xml - ADT

```
    android:focusable = "true"
    android:textColorHighlight = "#ff7eff15"
    android:textColorHint = "#ffff25e6"
    android:layout_marginTop = "46dp"
    android:layout_below = "@+id/imageView"
    android:layout_alignParentLeft = "true"
    android:layout_alignParentStart = "true"
    android:layout_alignParentRight = "true"
    android:layout_alignParentEnd = "true" />

<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/imageView"
    android:src="@drawable/abc"
    android:layout_below="@+id/textView"
    android:layout_centerHorizontal="true" />

<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    android:ems="10"
    android:id="@+id/editText2"
```

```
[2017-05-28 08:02:21 - SDK Manager] Deleting file C:\Users\zubaidha\.android\avd\test.ini
[2017-05-28 08:02:21 - SDK Manager] Deleting folder C:\Users\zubaidha\.android\avd\test.avd
[2017-05-28 08:02:21 - SDK Manager] AVD 'test' deleted.
```

Step 8: On the window, select **New.**

Step 9: On the next window, specify **AVD Name, Device, Target** and **Other Details** and Click **OK.**



Create new Android Virtual Device (AVD)

AVD Name: BscIT

Device: 4.0" WVGA (480 × 800: hdpi)

Target: Android 4.2.2 - API Level 17

CPU/ABI: ARM (armeabi-v7a)

Keyboard: ☑ Hardware keyboard present

Skin: ☑ Display a skin with hardware controls

Front Camera: None

Back Camera: None

Memory Options: RAM: 512 VM Heap: 32

Internal Storage: 200 MiB

SD Card: ● Size: MiB ○ File: Browse...

Emulation Options: □ Snapshot □ Use Host GPU

□ Override the existing AVD with the same name

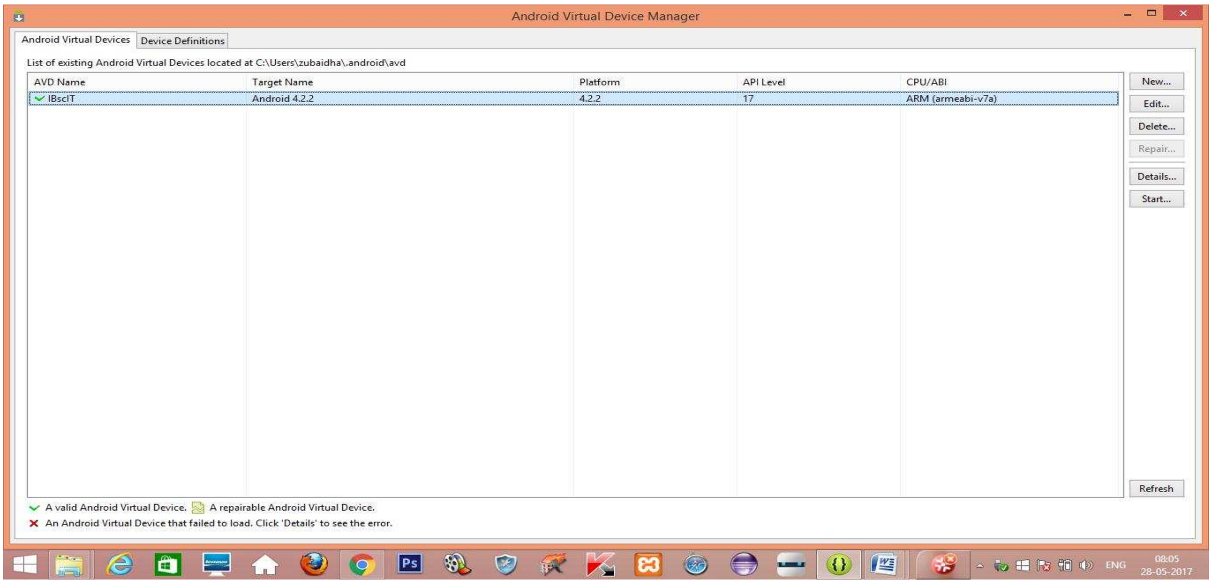
Step 10: Now a new **AVD** is created**.**

Step 11: On Completion of the **Project,** select **File ->Save All.**

Step 12: Select **Debug Icon** on the **Toolbar,** to debug the application.

Step 13: To run the application Select, **Run Icon** on the **Toolbar.**

Step 14: On the Android **Virtual Device Manager** window, select **AVD Name and**, Click **Start.**



Android Virtual Device Manager

Android Virtual Devices | Device Definitions

List of existing Android Virtual Devices located at C:\Users\zubaidha\.android\avd

AVD Name	Target Name	Platform	API Level	CPU/ABI
✓ IBscIT	Android 4.2.2	4.2.2	17	ARM (armeabi-v7a)

✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

Step 15: On the Launch **Options** window, Check Wipe **User Data and**, Click **Launch.**

Program:

Activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

        <TextView
           android:id="@+id/textview"
           android:layout_width="wrap_content"
           android:layout_height="wrap_content"
           android:layout_alignParentTop="true"
           android:layout_centerHorizontal="true"
           android:text="Menu Application"
           android:textSize="35dp" />

   <TextView
       android:id="@+id/textView"
       android:layout_width="wrap_content"
       android:layout_height="wrap_content"
       android:layout_below="@+id/textview"
       android:layout_centerHorizontal="true"
       android:text="II B.sc IT"
       android:textColor="#ff7aff24"
       android:textSize="35dp" />

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/textView"
        android:layout_below="@+id/textView"
        android:layout_marginLeft="28dp"
        android:src="@drawable/ic_launcher" />

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/imageView"
        android:layout_below="@+id/imageView"
        android:layout_marginTop="40dp"
        android:text="Menus" />

</RelativeLayout>
```

Popup_menu File

```
<menu xmlns:android="http://schemas.android.com/apk/res/android" >

    <item
        android:id="@+id/one"
        android:title="Cut"/>

            <item
        android:id="@+id/two"
        android:title="Copy"/>

    <item
        android:id="@+id/three"
        android:title="Paste"/>

        <item
        android:id="@+id/four"
        android:title="Delete"/>

            <item
        android:id="@+id/five"
        android:title="Select All"/>

    <item
        android:id="@+id/six"
        android:title="Unselect All"/>
</menu>
```

MainActivity.java

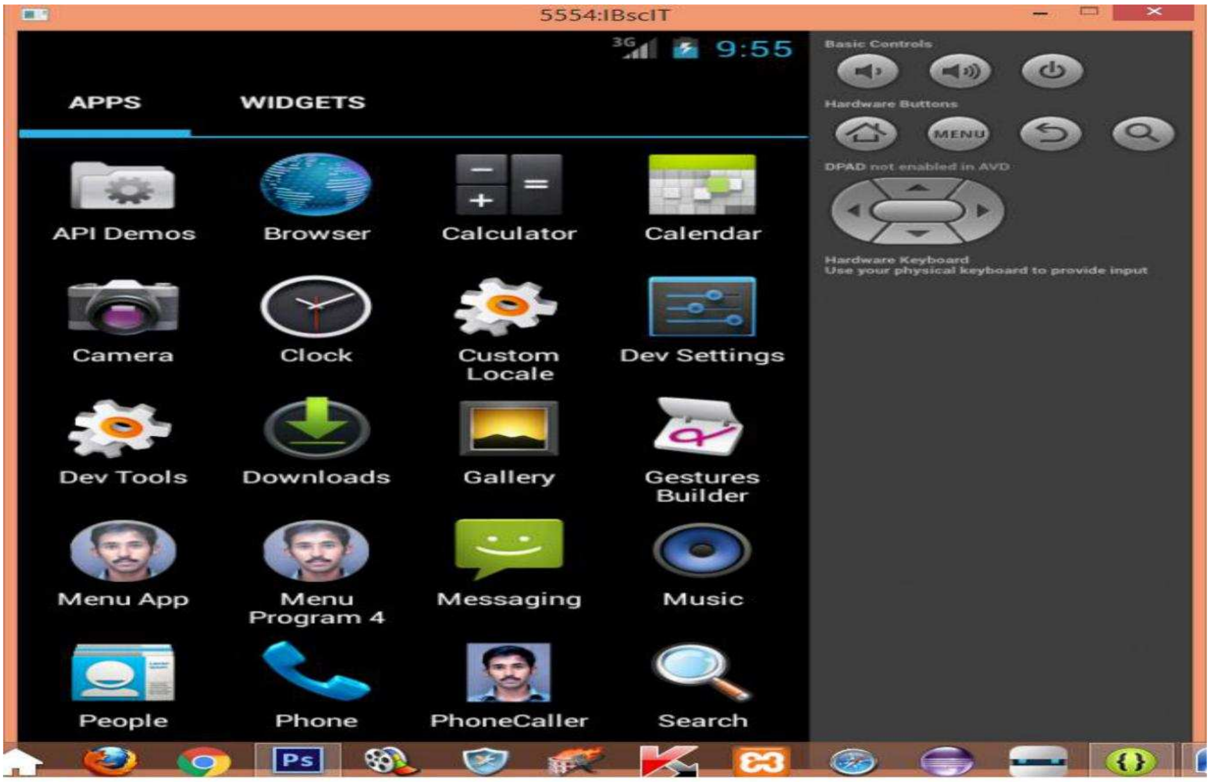
```
package com.example.menuapp;

import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.PopupMenu;
import android.widget.Toast;

public class MainActivity extends Activity {

            Button button1;
                        @Override
                        protected void onCreate(Bundle savedInstanceState)
                         { super.onCreate(savedInstanceState);
                         setContentView(R.layout.activity_main);

                         button1 = (Button) findViewById(R.id.button1);
```

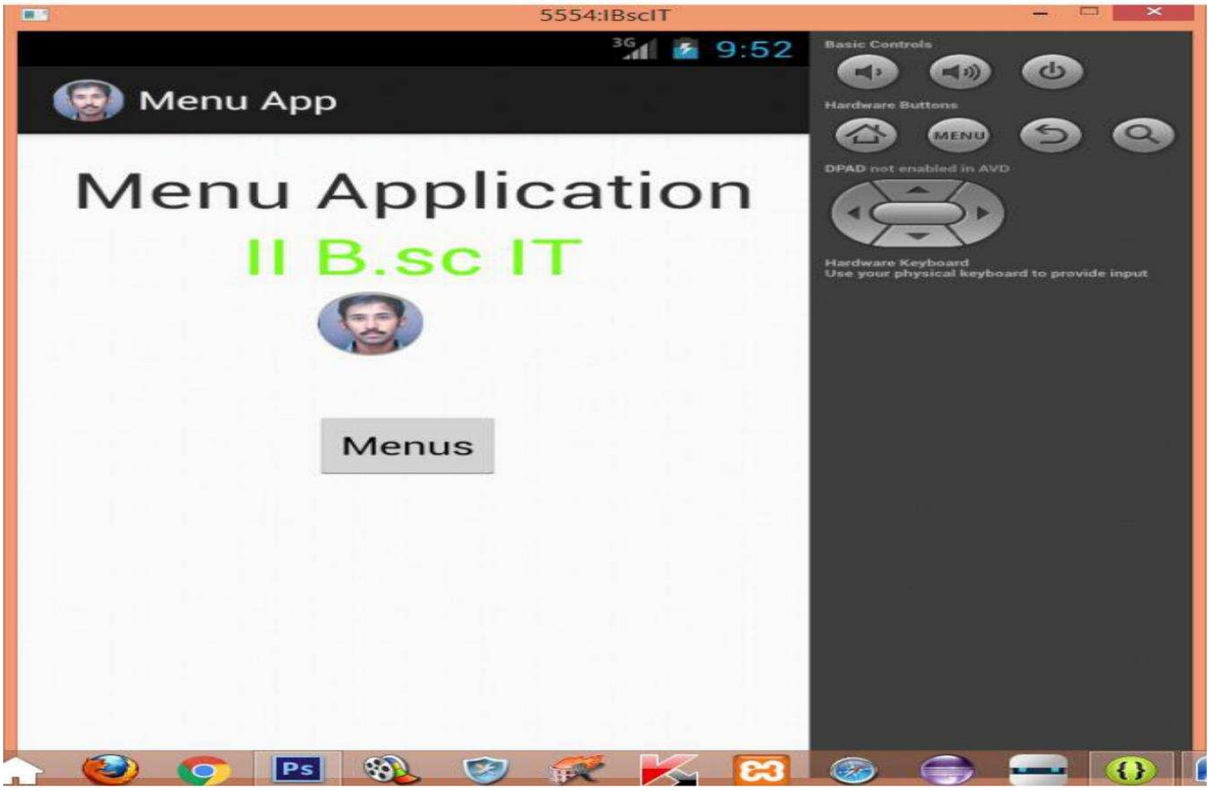
```
                    button1.setOnClickListener(new OnClickListener() {

                     @Override
                     public void onClick(View v) {
                            //Creating the instance of PopupMenu
                      PopupMenu popup = new PopupMenu(MainActivity.this, button1);
                      //Inflating the Popup using xml file
                      popup.getMenuInflater().inflate(R.menu.popup_menu, popup.getMenu());

                                         popup.setOnMenuItemClickListener(new
PopupMenu.OnMenuItemClickListener() {
                       public boolean onMenuItemClick(MenuItem item) {
                        Toast.makeText(MainActivity.this,"You Clicked : " +
 item.getTitle(),Toast.LENGTH_SHORT).show();
                        return true;
                       }
                      });

                      popup.show();//showing popup menu
                     }
                    });

                   }}
```

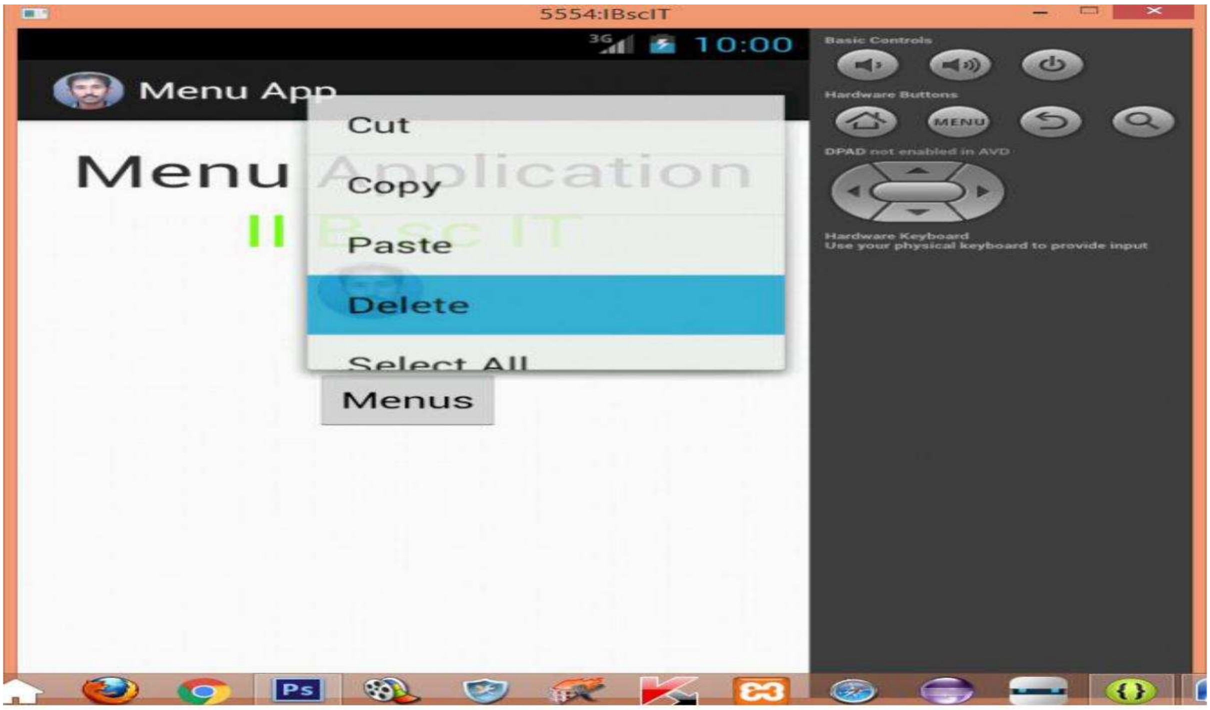
Android Manifest.xml

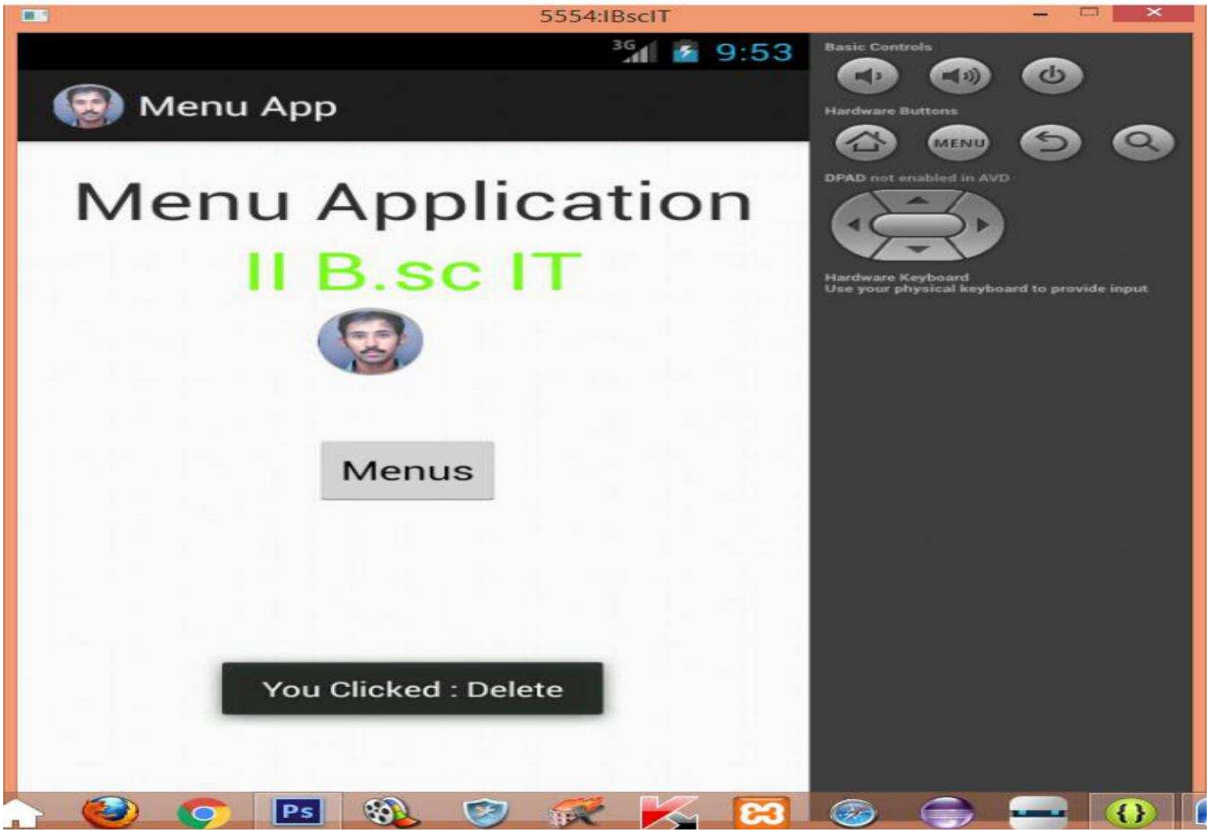
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.menuapp"
   android:versionCode="1"
   android:versionName="1.0" >

   <uses-sdk
      android:minSdkVersion="12"
      android:targetSdkVersion="17" />

   <application
      android:allowBackup="true"
      android:icon="@drawable/ic_launcher"
      android:label="@string/app_name"
      android:theme="@style/AppTheme" >
      <activity
         android:name="com.example.menuapp.MainActivity"

         android:label="@string/app_name" > <intent-filter>

            <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

Output:


5554:IBscIT
9:55
APPS
WIDGETS
API Demos
Browser
Calculator
Calendar
Camera
Clock
Custom Locale
Dev Settings
Dev Tools
Downloads
Gallery
Gestures Builder
Menu App
Menu Program 4
Messaging
Music
People
Phone
PhoneCaller
Search
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input


5554:IBscIT
9:52
Menu App
Menu Application
II B.sc IT
Menus
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input





You Clicked : Delete

Result:

The above aim of the program has been achieved successfully.

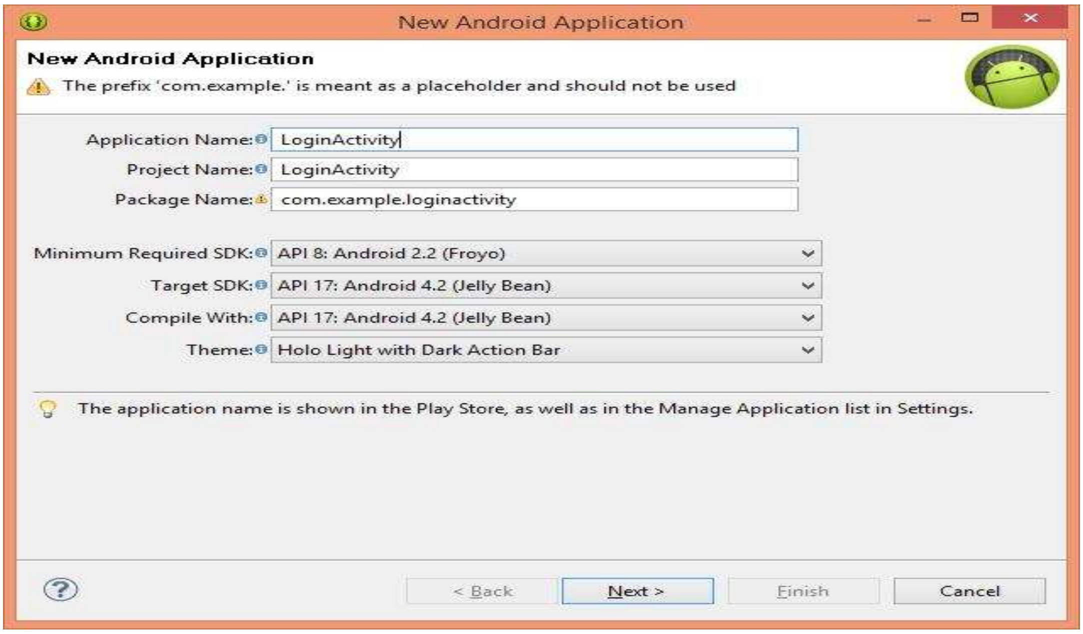
5.SCREEN COLOR CHANGE

Aim:

To create an android application with login operations.

Procedure:

Step 1: Open Eclipse IDE and go to **File -> New** -> **Project** -> **Android** -> **Android Application Project**. You have to specify the **Application Name**, the **Project Name** and the **Package name** in the appropriate text fields and then click **Next**.



New Android Application

New Android Application

The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: LoginActivity

Project Name: LoginActivity

Package Name: com.example.loginactivity

Minimum Required SDK: API 8: Android 2.2 (Froyo)

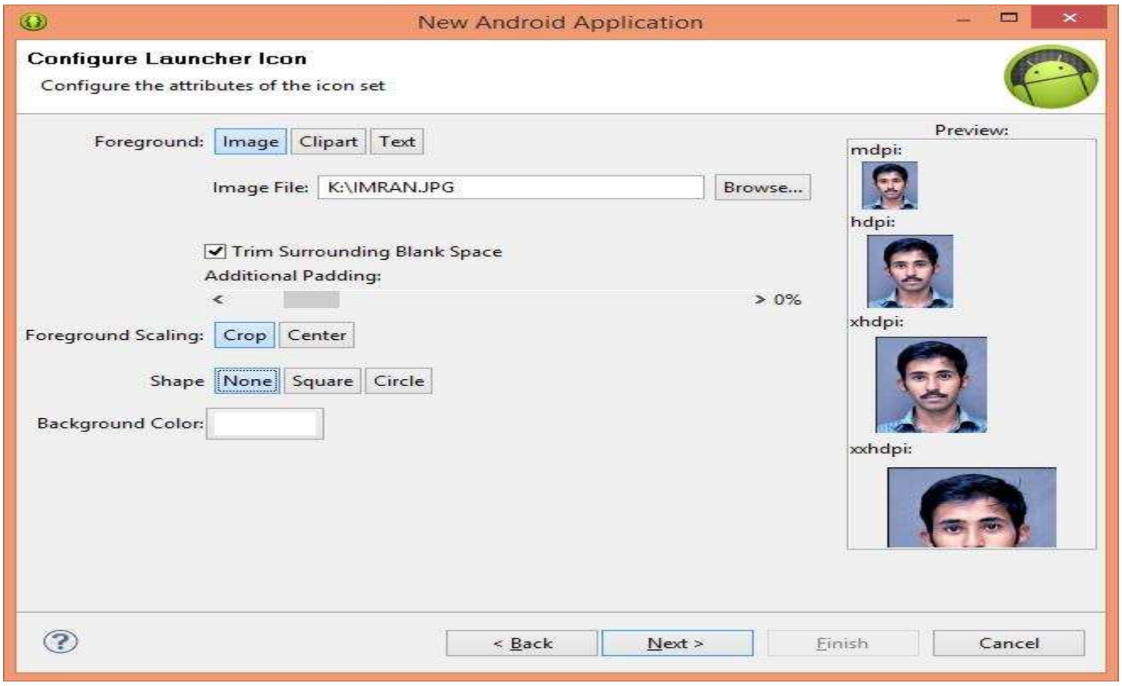
Target SDK: API 17: Android 4.2 (Jelly Bean)

Compile With: API 17: Android 4.2 (Jelly Bean)

Theme: Holo Light with Dark Action Bar

The application name is shown in the Play Store, as well as in the Manage Application list in Settings.

Step 2: In the next window make sure the select the **Application Launcher Icon** option is selected in order to create a new activity for your project, and click **Next**.



New Android Application

Configure Launcher Icon

Configure the attributes of the icon set

Foreground: Image

Image File: K:\IMRAN.JPG

☑ Trim Surrounding Blank Space

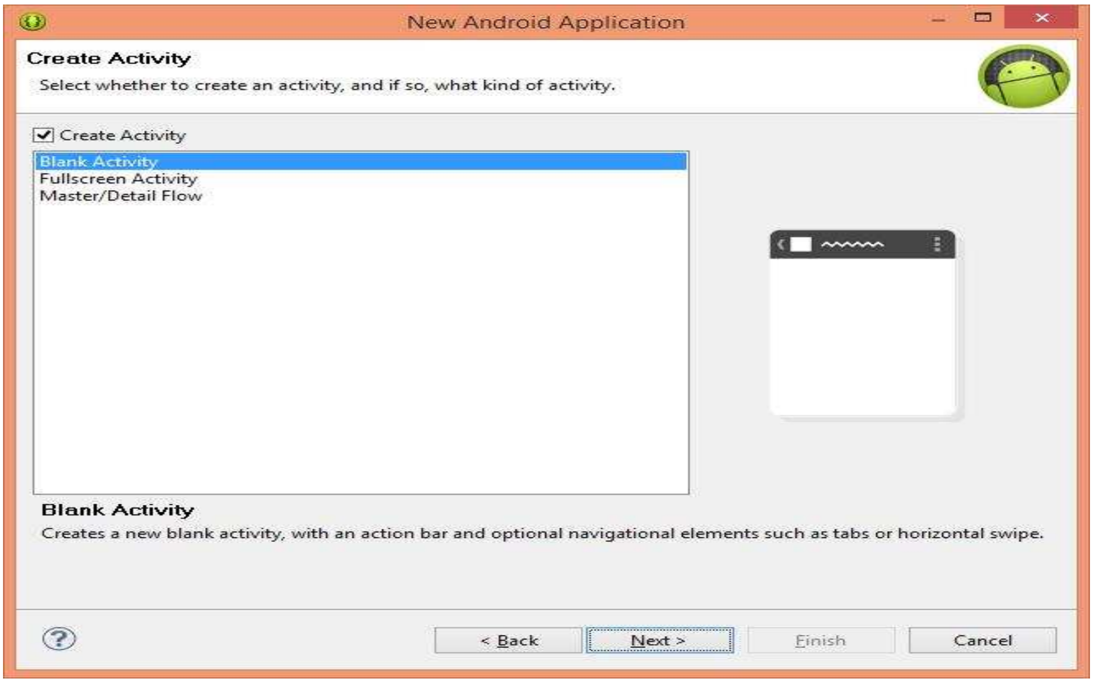
Additional Padding: 0%

Foreground Scaling: Crop

Shape None

Background Color:

Step 3: On the next window click to select the **Blank Activity** and click **Next**.



New Android Application

Create Activity

Select whether to create an activity, and if so, what kind of activity.

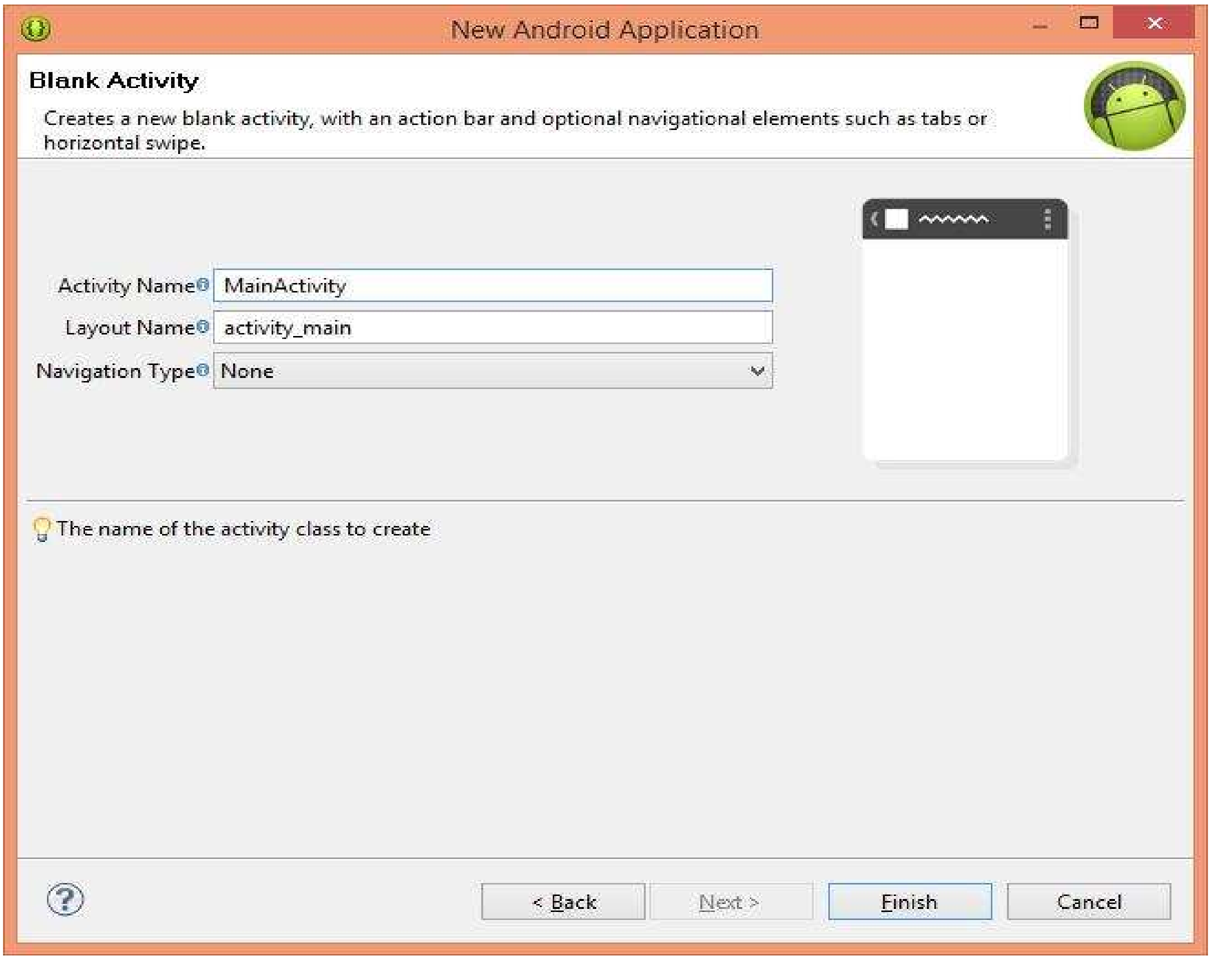
☑ Create Activity

Blank Activity
Fullscreen Activity
Master/Detail Flow

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Step 4: On the next window, name the specify **Activity Name, Layout Name and Navigation Type,** click **Next.**



New Android Application

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Activity Name: MainActivity

Layout Name: activity_main

Navigation Type: None

The name of the activity class to create

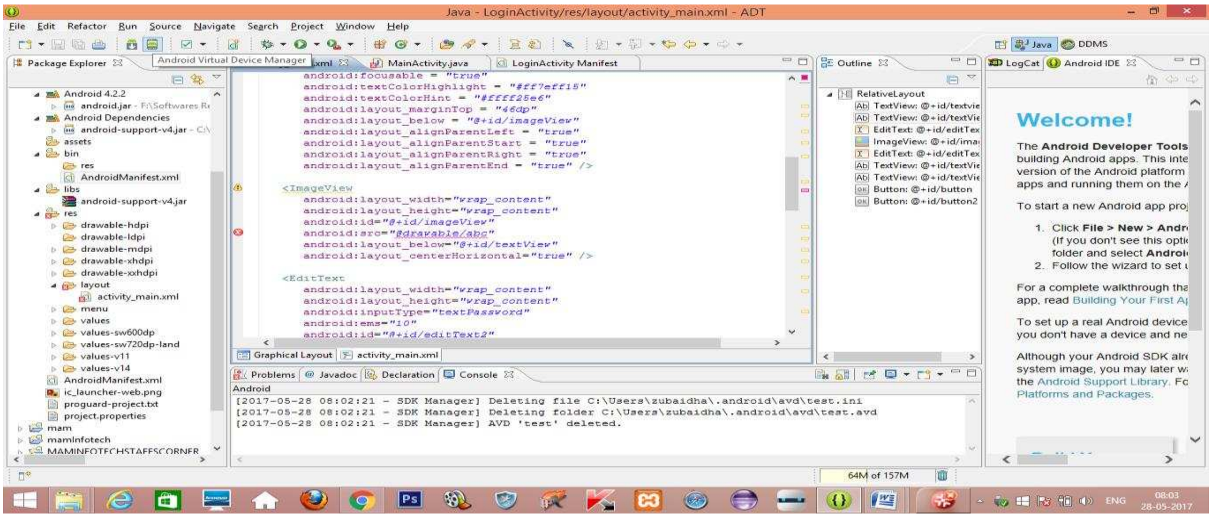
Step 5: Drag and Drop All the needed Components from the Palette Window to the **Design View of** the **Activity main.xml.**

Step 6: Create a new activity named **Success.xml,** and add some content.

Step 7: Code the operations that are want to perform in the **MainActivity.java, of** the preferred **.Xml** files.

Step 8: Create a new activity named **Success.xml,** and add some content for mobility.

Step 8: Create an **AVD, by** Clicking the **Android Virtual Device Icon, on** the **Toolbar.**



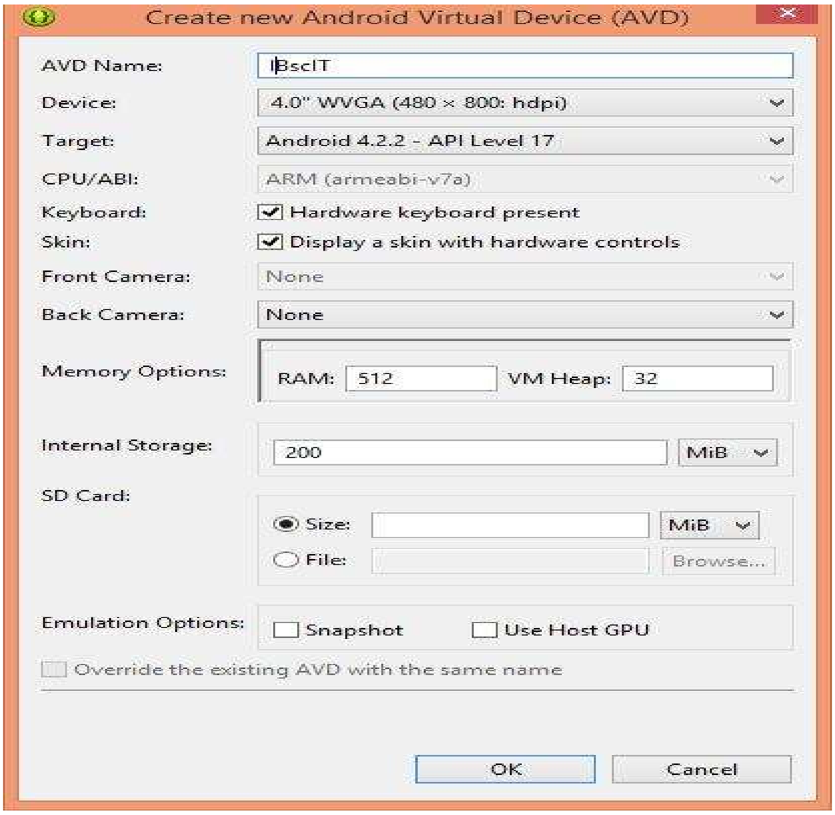
Java - LoginActivity/res/layout/activity_main.xml - ADT

```
        android:focusable = "true"
        android:textColorHighlight = "#ff7eff15"
        android:textColorHint = "#ffff25e6"
        android:layout_marginTop = "46dp"
        android:layout_below = "@+id/imageView"
        android:layout_alignParentLeft = "true"
        android:layout_alignParentStart = "true"
        android:layout_alignParentRight = "true"
        android:layout_alignParentEnd = "true" />

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imageView"
        android:src="@drawable/abc"
        android:layout_below="@+id/textView"
        android:layout_centerHorizontal="true" />

    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:ems="10"
        android:id="@+id/editText2"
```

```
Android
[2017-05-28 08:02:21 - SDK Manager] Deleting file C:\Users\zubaidha\.android\avd\test.ini
[2017-05-28 08:02:21 - SDK Manager] Deleting folder C:\Users\zubaidha\.android\avd\test.avd
[2017-05-28 08:02:21 - SDK Manager] AVD 'test' deleted.
```

Step 8: On the window, select **New.**

Step 9: On the next window, specify **AVD Name, Device, Target** and **Other Details** and Click **OK.**



AVD Name: BscIT

Device: 4.0" WVGA (480 × 800: hdpi)

Target: Android 4.2.2 - API Level 17

CPU/ABI: ARM (armeabi-v7a)

Keyboard: ☑ Hardware keyboard present

Skin: ☑ Display a skin with hardware controls

Front Camera: None

Back Camera: None

Memory Options: RAM: 512 VM Heap: 32

Internal Storage: 200 MiB

SD Card: ● Size: MiB ○ File: Browse...

Emulation Options: □ Snapshot □ Use Host GPU

□ Override the existing AVD with the same name

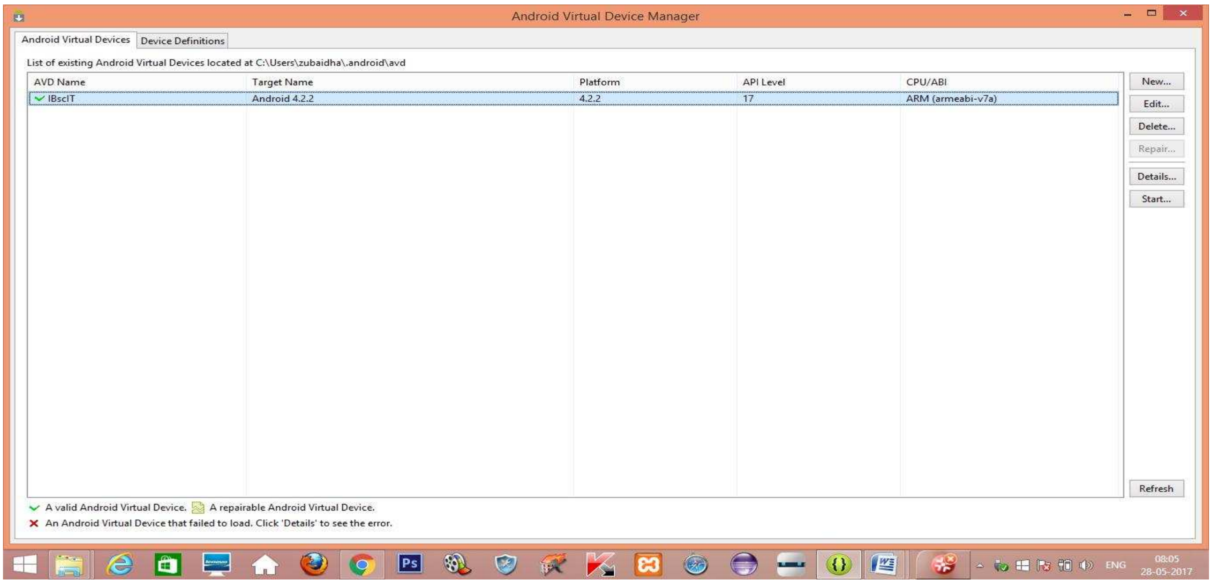
Step 10: Now a new **AVD** is created**.**

Step 11: On Completion of the **Project,** select **File ->Save All.**

Step 12: Select **Debug Icon** on the **Toolbar,** to debug the application.

Step 13: To run the application Select, **Run Icon** on the **Toolbar.**

Step 14: On the Android **Virtual Device Manager** window, select **AVD Name and**, Click **Start.**



List of existing Android Virtual Devices located at C:\Users\zubaidha\.android\avd

AVD Name	Target Name	Platform	API Level	CPU/ABI
✓ IBscIT	Android 4.2.2	4.2.2	17	ARM (armeabi-v7a)

✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

Step 15: On the Launch **Options** window, Check Wipe **User Data and**, Click **Launch.**

Program:

Activity_main.xml

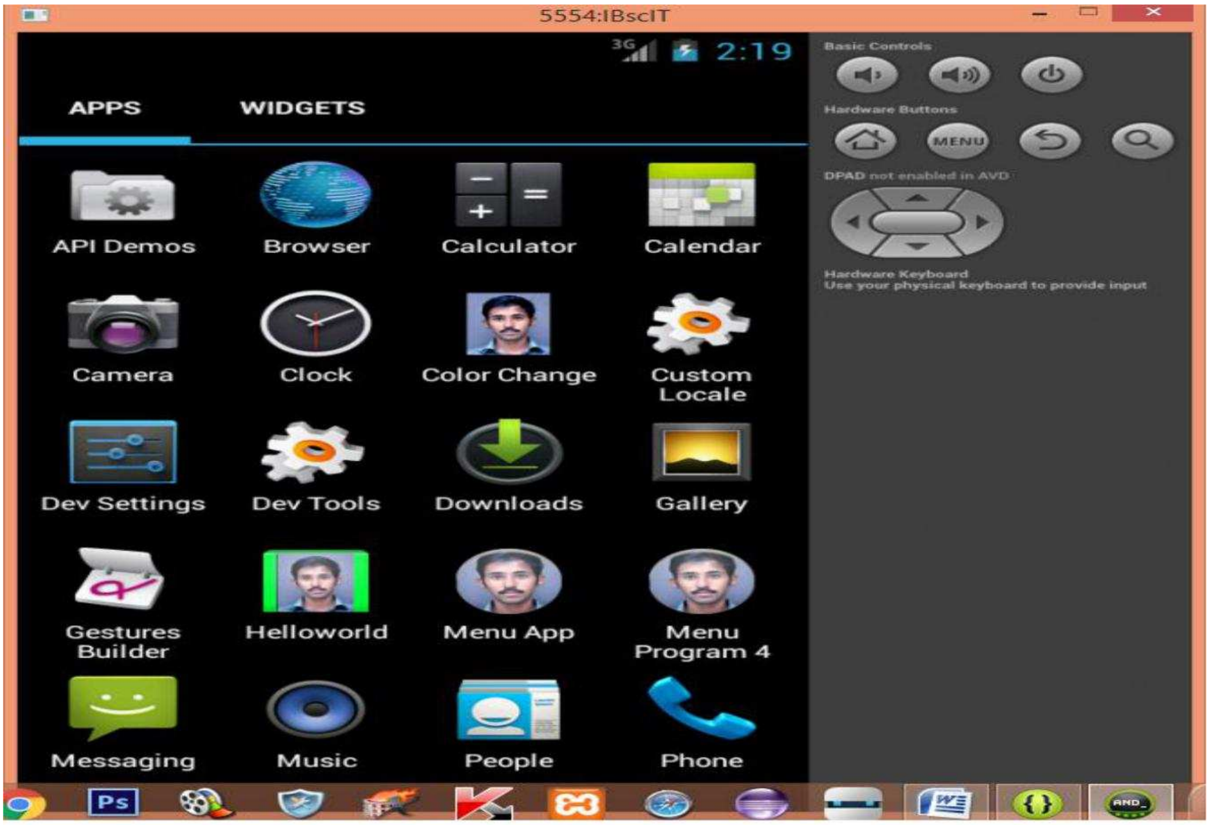
```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >
 <TextView
         android:id="@+id/textview"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:layout_alignParentTop="true"
         android:layout_centerHorizontal="true"
         android:text="Screen Color Change"
         android:textSize="35dp" />
  <TextView
       android:id="@+id/textView"
       android:layout_width="wrap_content"

       android:layout_height="wrap_content"

       android:layout_below="@+id/textview"

       android:layout_centerHorizontal="true"

       android:text="II B.sc IT"

       android:textColor="#ff7aff24"

       android:textSize="35dp" />

  <Button

       android:id="@+id/button2"

       android:layout_width="wrap_content"

       android:layout_height="wrap_content"

       android:layout_alignLeft="@+id/button"

       android:layout_below="@+id/button"

       android:layout_marginTop="52dp"

       android:text="Blue" />

  <Button

       android:id="@+id/button"

       android:layout_width="wrap_content"

       android:layout_height="wrap_content"

       android:layout_alignRight="@+id/textView"

       android:layout_centerVertical="true"
       android:layout_marginRight="23dp"

       android:text="Red" />

  <ImageView

       android:id="@+id/imageView"

       android:layout_width="wrap_content"

       android:layout_height="wrap_content"

       android:layout_below="@+id/textView"

       android:layout_centerHorizontal="true"

       android:src="@drawable/ic_launcher" />

</RelativeLayout>
```

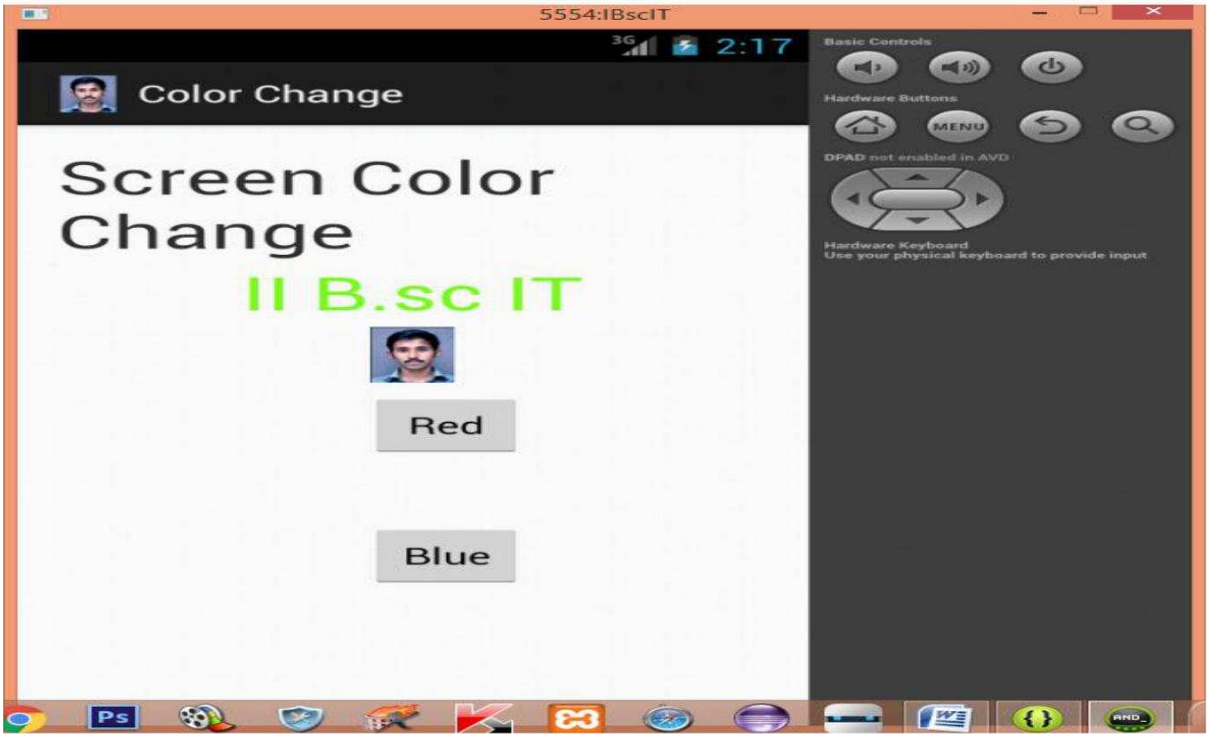
MainActivity.java

```
package com.example.colorchange;

import android.app.Activity;

import android.os.Bundle;

public class MainActivity extends Activity {

        @Override

          protected void onCreate(Bundle savedInstanceState)

            { super.onCreate(savedInstanceState);

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

```
  }

}
```

Output


5554:IBscIT
3G 2:19
APPS
WIDGETS
API Demos
Browser
Calculator
Calendar
Camera
Clock
Color Change
Custom Locale
Dev Settings
Dev Tools
Downloads
Gallery
Gestures Builder
Helloworld
Menu App
Menu Program 4
Messaging
Music
People
Phone
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input


5554:IBscIT
3G 2:17
Color Change
Screen Color Change
II B.sc IT
Red
Blue
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input

Result:

The above aim of the program has been achieved successfully.

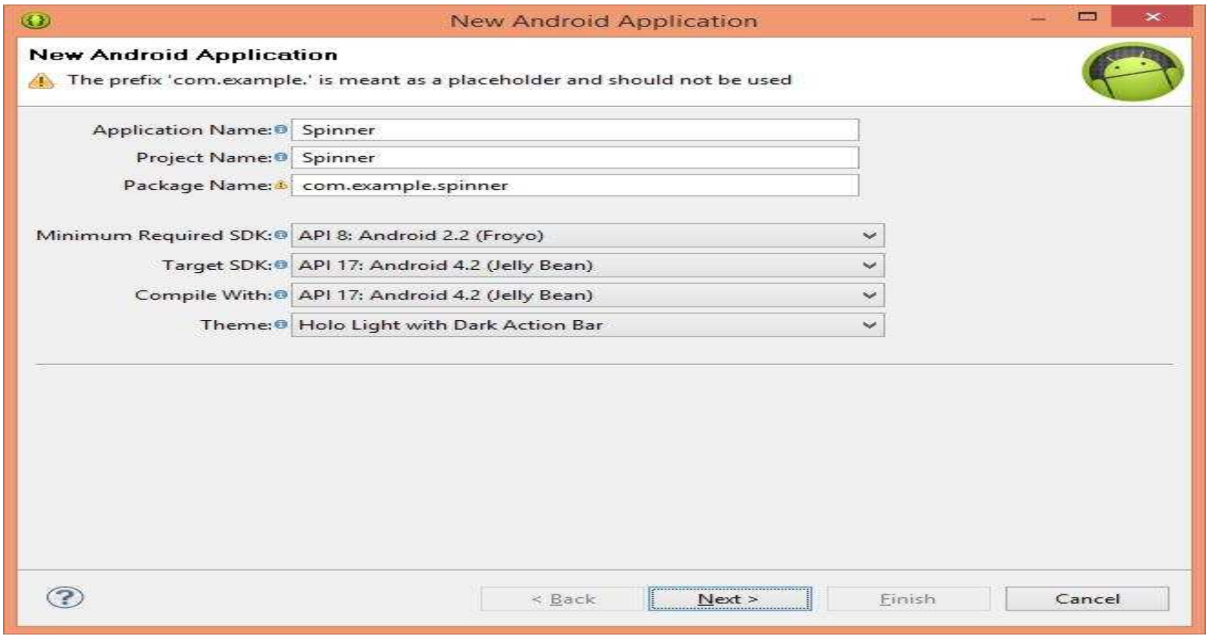
6.COMBO BOX,SPINNER ACTIVITY

Aim:

To create an android application to perform activities using combo box,spinner and Toast Message.

Procedure:

Step 1: Open Eclipse IDE and go to **File -> New** -> **Project** -> **Android** -> **Android Application Project**. You have to specify the **Application Name**, the **Project Name** and the **Package name** in the appropriate text fields and then click **Next**.



New Android Application

New Android Application

The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: Spinner

Project Name: Spinner

Package Name: com.example.spinner

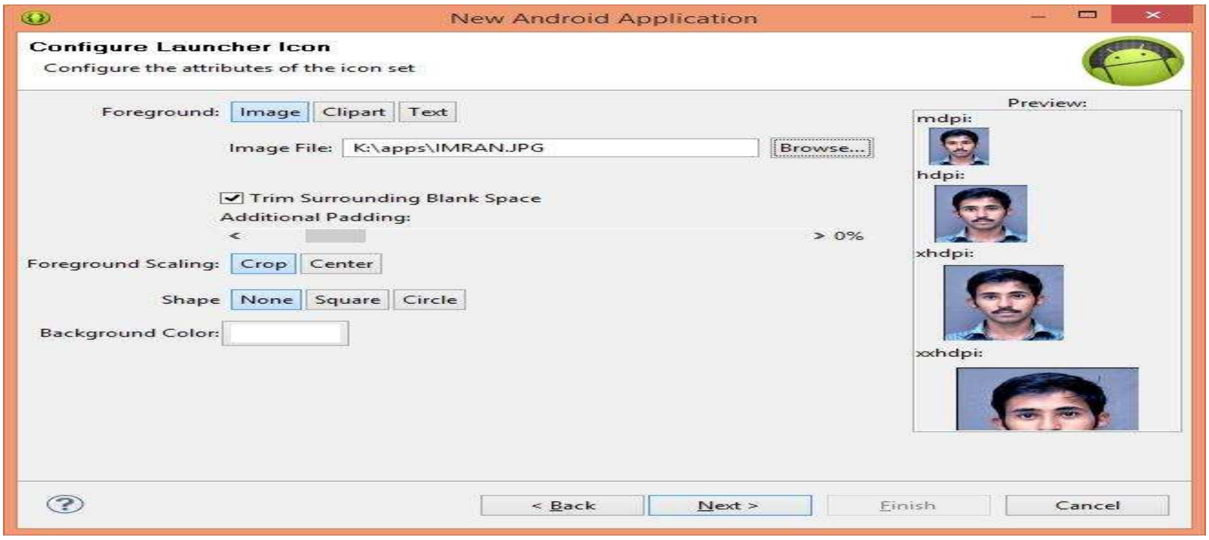
Minimum Required SDK: API 8: Android 2.2 (Froyo)

Target SDK: API 17: Android 4.2 (Jelly Bean)

Compile With: API 17: Android 4.2 (Jelly Bean)

Theme: Holo Light with Dark Action Bar

Step 2: In the next window make sure the select the **Application Launcher Icon** option is selected in order to create a new activity for your project, and click **Next**.



New Android Application

Configure Launcher Icon

Configure the attributes of the icon set

Foreground: Image Clipart Text

Image File: K:\apps\IMRAN.JPG

☑ Trim Surrounding Blank Space

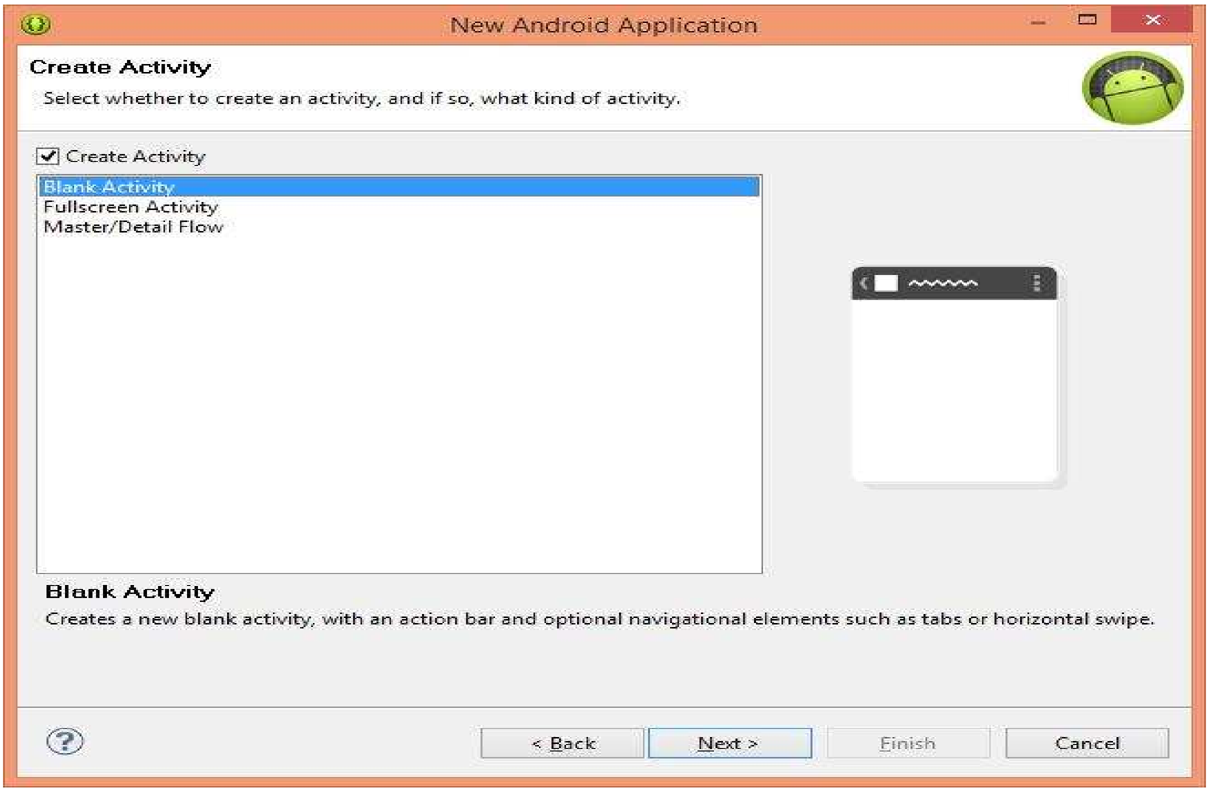
Additional Padding: 0%

Foreground Scaling: Crop Center

Shape: None Square Circle

Background Color:

Step 3: On the next window click to select the **Blank Activity** and click **Next**.



Create Activity

Select whether to create an activity, and if so, what kind of activity.

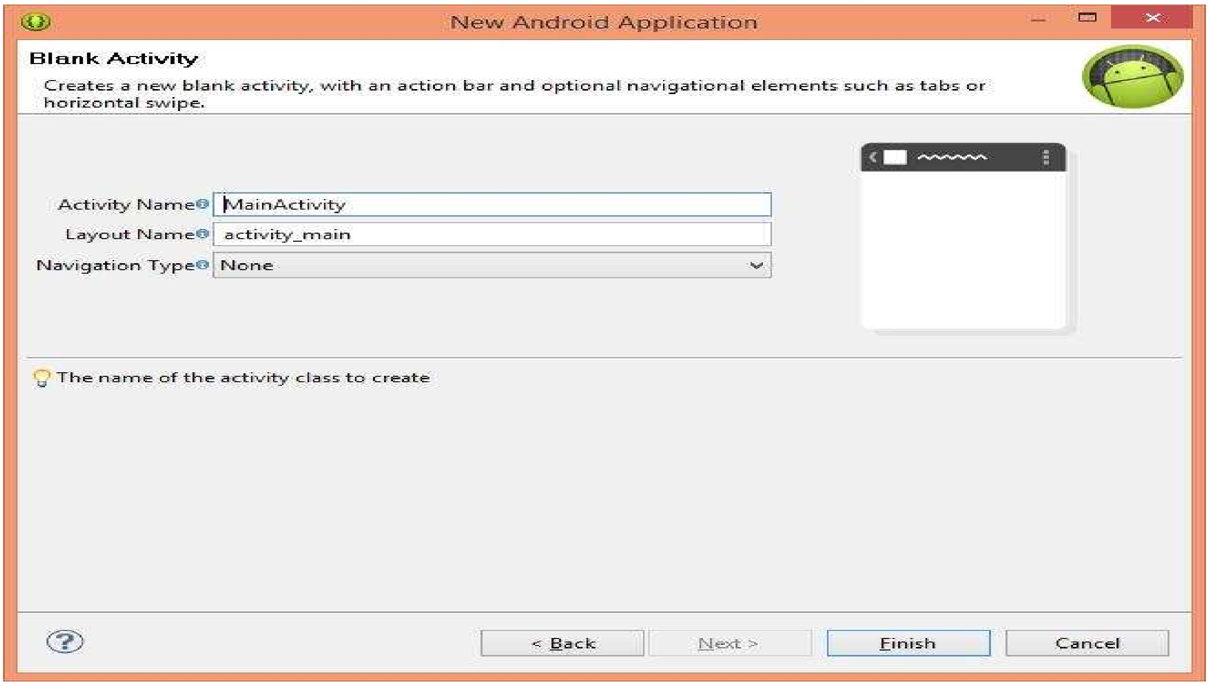
☑ Create Activity

- Blank Activity
- Fullscreen Activity
- Master/Detail Flow

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Step 4: On the next window, name the specify **Activity Name, Layout Name and Navigation Type,** click **Next**.



Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Activity Name: MainActivity

Layout Name: activity_main

Navigation Type: None

The name of the activity class to create

Step 5: Drag and Drop All the needed Components from the Palette Window to the **Design View of** the **Activity main.xml.**

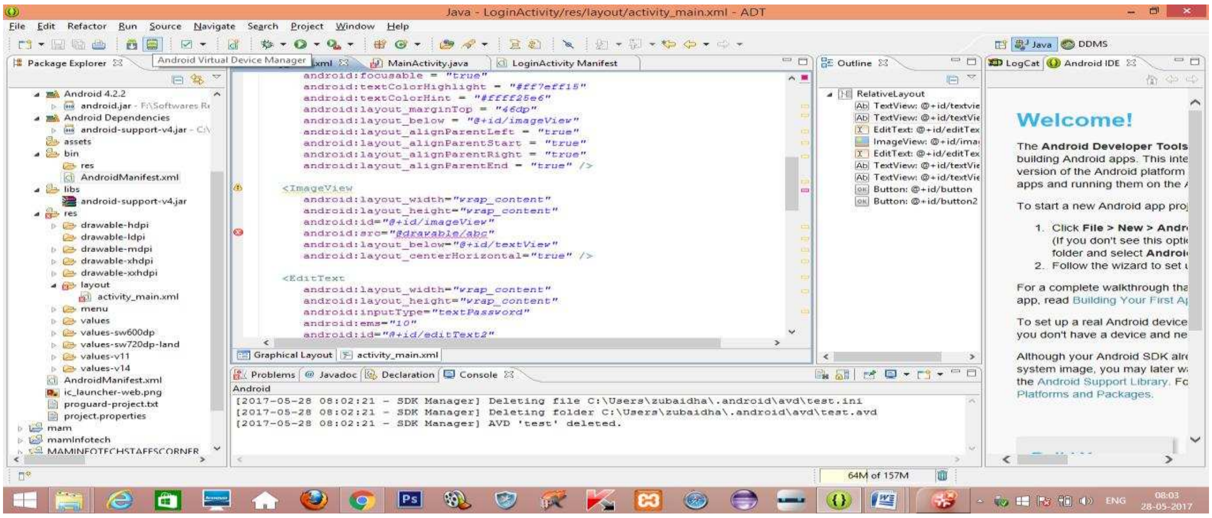
Step 6: Create a new activity named **Success.xml,** and add some content.

Step 7: Code the operations that are want to perform in the **MainActivity.java, of** the preferred **.Xml** files.

Step 8: On the **Mainfest.xml,** file the permission for accessing phone dialer has to obtained.

Step 9: Create a new activity named **Success.xml,** and add some content for mobility.

Step 10: Create an **AVD, by** Clicking the **Android Virtual Device Icon, on** the **Toolbar.**

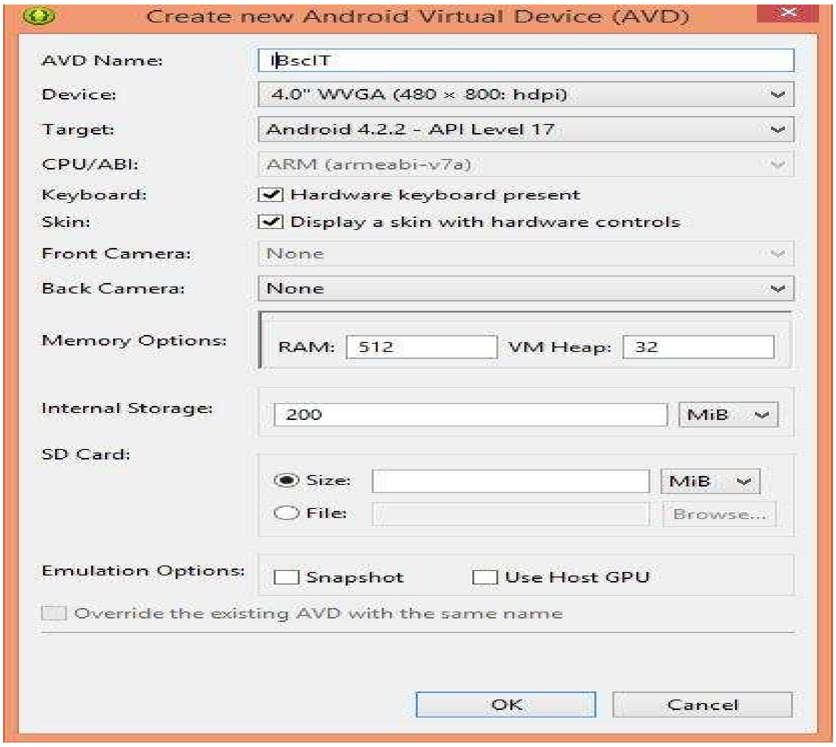


```
    android:focusable = "true"
    android:textColorHighlight = "#ff7eff15"
    android:textColorHint = "#ffff25e6"
    android:layout_marginTop = "46dp"
    android:layout_below = "@+id/imageView"
    android:layout_alignParentLeft = "true"
    android:layout_alignParentStart = "true"
    android:layout_alignParentRight = "true"
    android:layout_alignParentEnd = "true" />

<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/imageView"
    android:src="@dravable/abc"
    android:layout_below="@+id/textView"
    android:layout_centerHorizontal="true" />

<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    android:ems="10"
    android:id="@+id/editText2"
```

```
Android
[2017-05-28 08:02:21 - SDK Manager] Deleting file C:\Users\zubaidha\.android\avd\test.ini
[2017-05-28 08:02:21 - SDK Manager] Deleting folder C:\Users\zubaidha\.android\avd\test.avd
[2017-05-28 08:02:21 - SDK Manager] AVD 'test' deleted.
```

Step 11: On the window, select **New.**

Step 12: On the next window, specify **AVD Name, Device, Target** and **Other Details** and Click **OK.**



Create new Android Virtual Device (AVD)

AVD Name:	BscIT
Device:	4.0" WVGA (480 × 800: hdpi)
Target:	Android 4.2.2 - API Level 17
CPU/ABI:	ARM (armeabi-v7a)
Keyboard:	☑ Hardware keyboard present
Skin:	☑ Display a skin with hardware controls
Front Camera:	None
Back Camera:	None
Memory Options:	RAM: 512 VM Heap: 32
Internal Storage:	200 MiB
SD Card:	● Size: MiB ○ File:
Emulation Options:	□ Snapshot □ Use Host GPU

□ Override the existing AVD with the same name

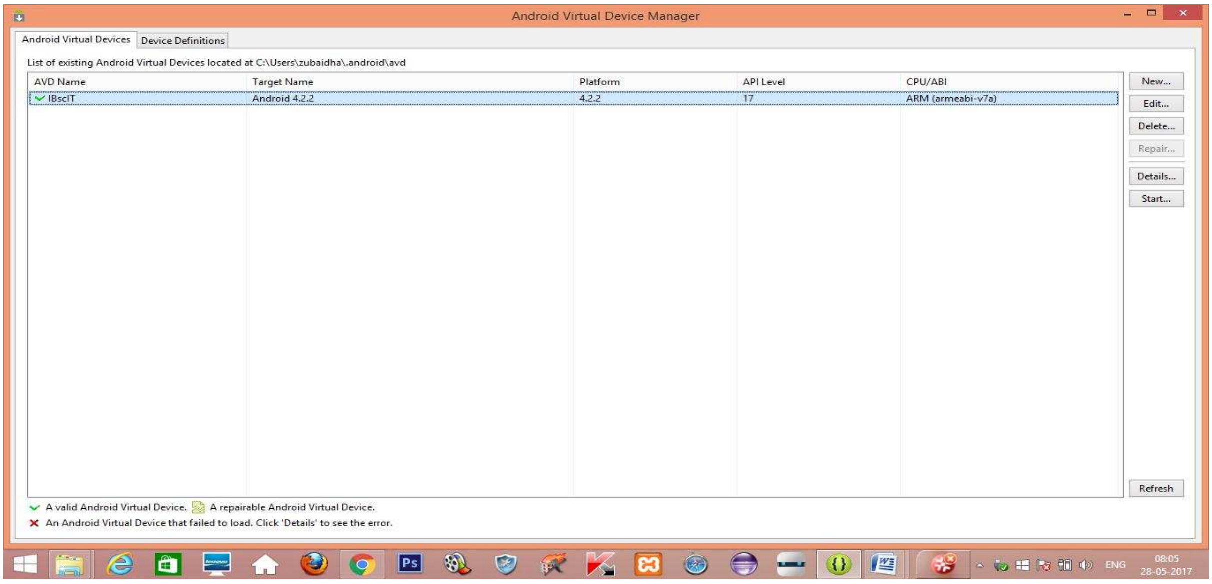
Step 13: Now a new **AVD** is created**.**

Step 14: On Completion of the **Project,** select **File ->Save All.**

Step 15: Select **Debug Icon** on the **Toolbar,** to debug the application.

Step 16: To run the application Select, **Run Icon** on the **Toolbar.**

Step 17: On the Android **Virtual Device Manager** window, select **AVD Name and**, Click **Start.**



List of existing Android Virtual Devices located at C:\Users\zubaidha\.android\avd

AVD Name	Target Name	Platform	API Level	CPU/ABI
✓ IBscIT	Android 4.2.2	4.2.2	17	ARM (armeabi-v7a)

✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

Step 18: On the Launch **Options** window, Check Wipe **User Data and**, Click **Launch.**

Program

Activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:tools="http://schemas.android.com/tools"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:paddingBottom="@dimen/activity_vertical_margin"

    android:paddingLeft="@dimen/activity_horizontal_margin"

    android:paddingRight="@dimen/activity_horizontal_margin"

    android:paddingTop="@dimen/activity_vertical_margin"
```

```
 tools:context=".MainActivity" >

 <TextView

        android:id="@+id/textview"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_alignParentTop="true"

        android:layout_centerHorizontal="true"

        android:text="Spinner With Toast"

        android:textSize="35dp" />

<TextView

    android:id="@+id/textView"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_below="@+id/textview"

    android:layout_centerHorizontal="true"

    android:text="II B.sc IT"

    android:textColor="#ff7aff24"

    android:textSize="35dp" />

   <ImageView

    android:id="@+id/imageView"

    android:layout_width="wrap_content"
```

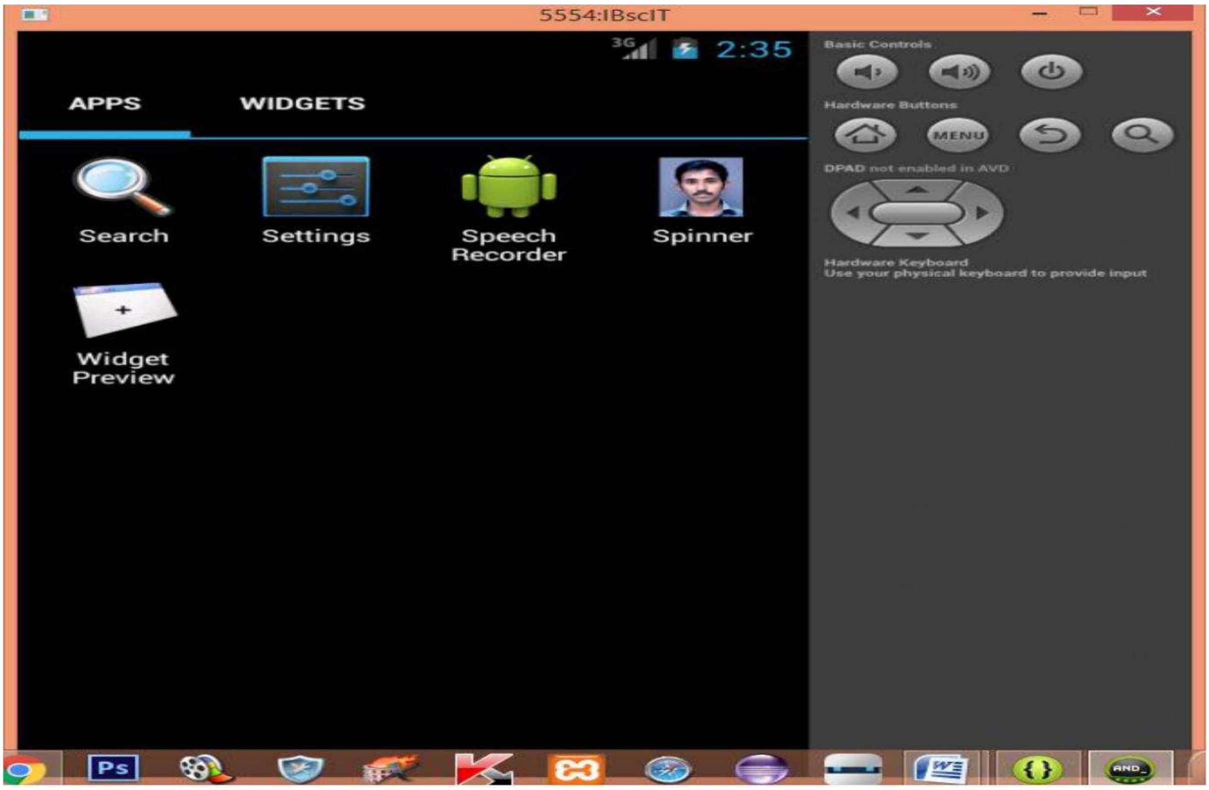
```
       android:layout_height="wrap_content"
       android:layout_below="@+id/textView"
       android:layout_centerHorizontal="true"
       android:src="@drawable/ic_launcher" />
    <Spinner
        android:id="@+id/spinner1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="175dp" />
</RelativeLayout>
```

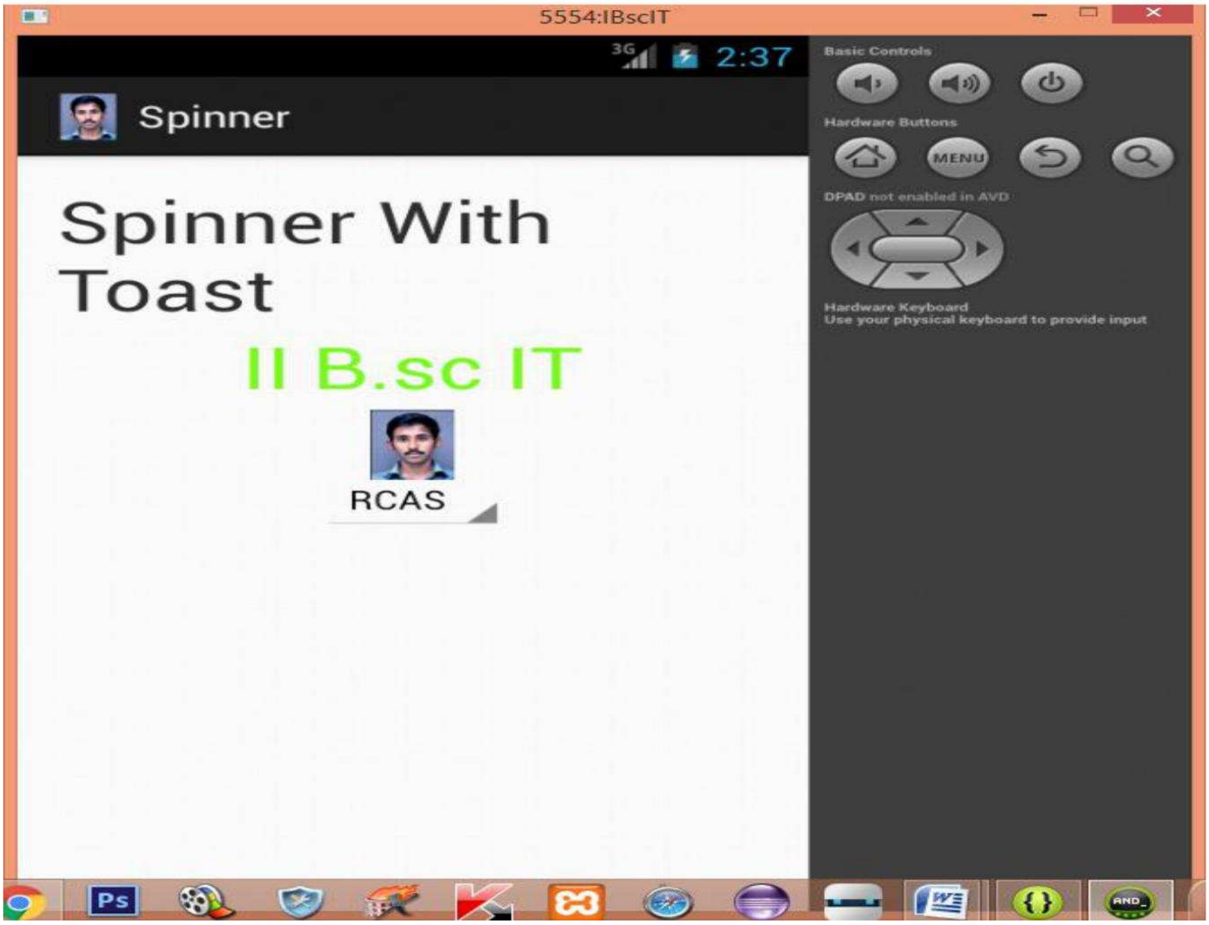
MainActivity.java

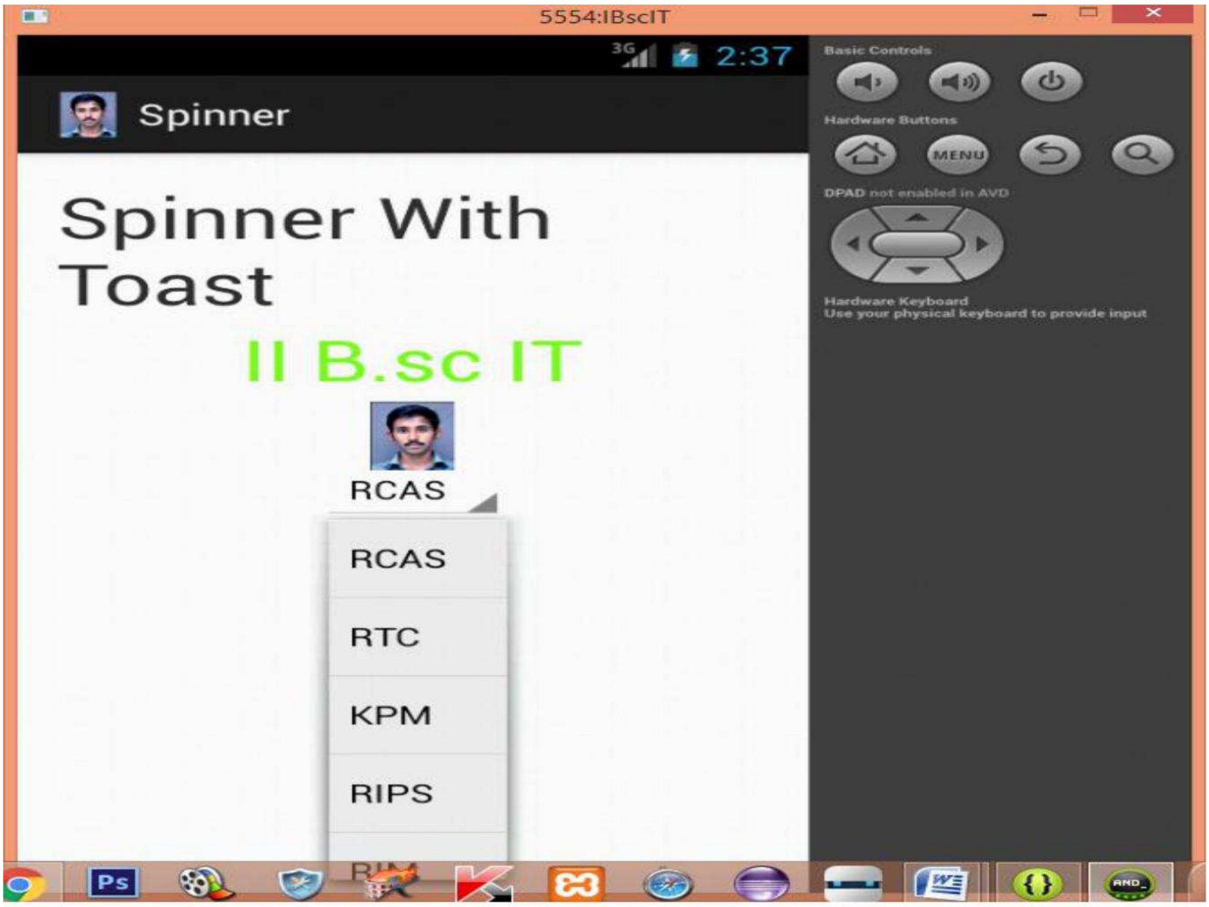
```
package com.example.spinner;
import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Spinner;
import android.widget.TextView;

import android.widget.Toast;

public class MainActivity extends Activity

implements AdapterView.OnItemSelectedListener {

   String[] country = { "RCAS", "RTC", "KPM", "RIPS", "RIM", };

  @Override

   protected void onCreate(Bundle savedInstanceState)

      { super.onCreate(savedInstanceState);

      setContentView(R.layout.activity_main);

      //Getting the instance of Spinner and applying OnItemSelectedListener on

      it Spinner spin = (Spinner) findViewById(R.id.spinner1);

      spin.setOnItemSelectedListener(this);

      //Creating the ArrayAdapter instance having the country list

      ArrayAdapter aa = new
ArrayAdapter(this,android.R.layout.simple_spinner_item,country);

      aa.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);


      //Setting the ArrayAdapter data on the Spinner spin.setAdapter(aa);


   }

   //Performing action onItemSelected and onNothing

   selected @Override
```

```
 public void onItemSelected(AdapterView<?> arg0, View arg1, int position,long id) {

     Toast.makeText(getApplicationContext(),country[position] ,
     Toast.LENGTH_LONG).show();

 }

 @Override

 public void onNothingSelected(AdapterView<?> arg0)

     { // TODO Auto-generated method stub

 }

 @Override

 public boolean onCreateOptionsMenu(Menu menu) {

     // Inflate the menu; this adds items to the action bar if it is

    present. //getMenuInflater().inflate(R.menu.activity_main, menu);

     return true;

 }

}
```

Output:


5554:IBscIT
2:35
APPS
WIDGETS
Search
Settings
Speech Recorder
Spinner
Widget Preview
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input


5554:IBscIT
2:37
Spinner
Spinner With Toast
II B.sc IT
RCAS
Basic Controls
Hardware Buttons
MENU
DPAD not enabled in AVD
Hardware Keyboard
Use your physical keyboard to provide input



Spinner With Toast

II B.sc IT

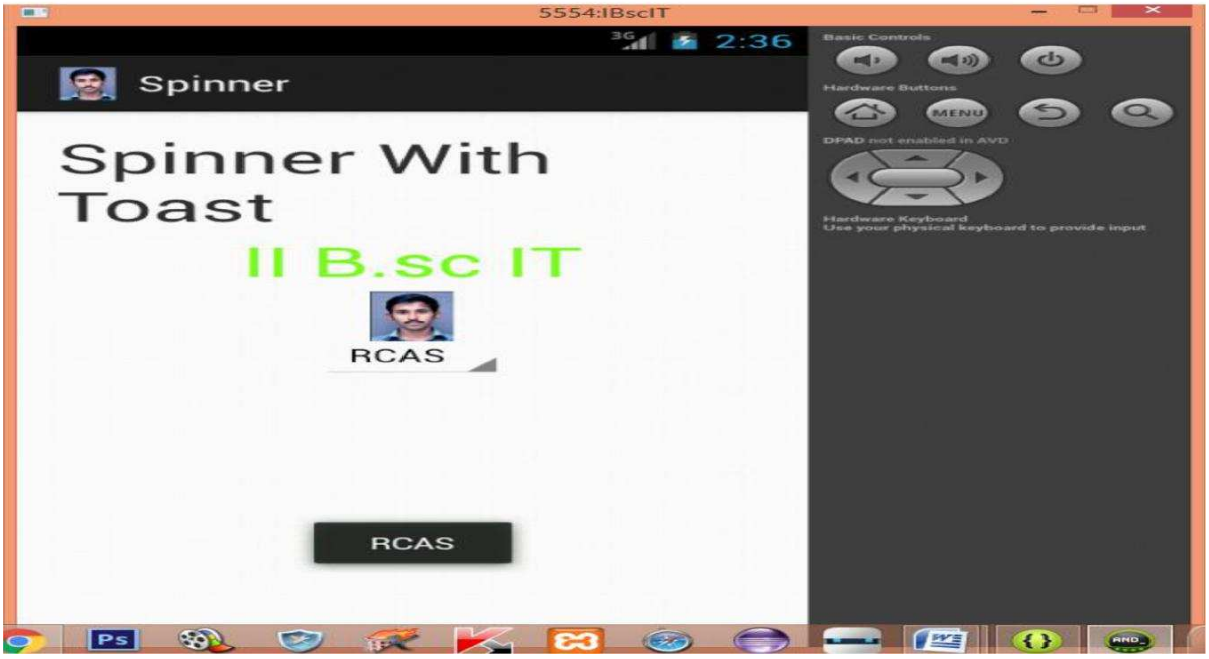
RCAS

RCAS

RTC

KPM

RIPS



Spinner With Toast

II B.sc IT

RCAS

RCAS

Result:

The above aim of the program has been achieved successfully.

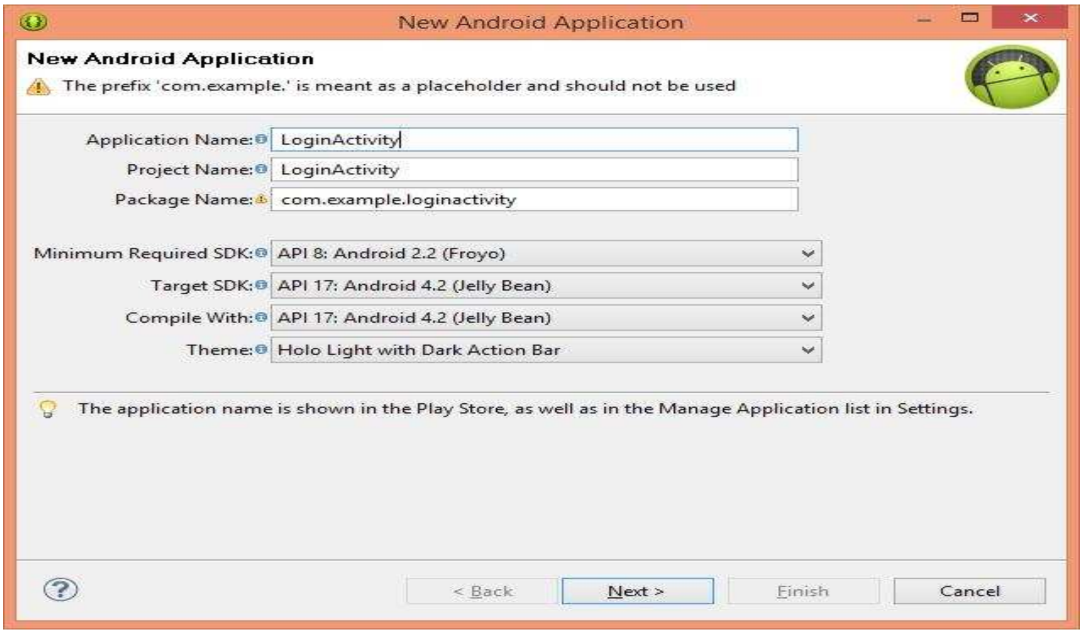
7. PHONE CALL ACTIVITY

Aim:

To create an android application to perform phone call to the user entered number.

Procedure:

Step 1: Open Eclipse IDE and go to File -> New -> Project -> Android -> Android Application Project. You have to specify the Application Name, the Project Name and the Package name in the appropriate text fields and then click Next.



New Android Application

New Android Application

⚠ The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: LoginActivity

Project Name: LoginActivity

Package Name: com.example.loginactivity

Minimum Required SDK: API 8: Android 2.2 (Froyo)

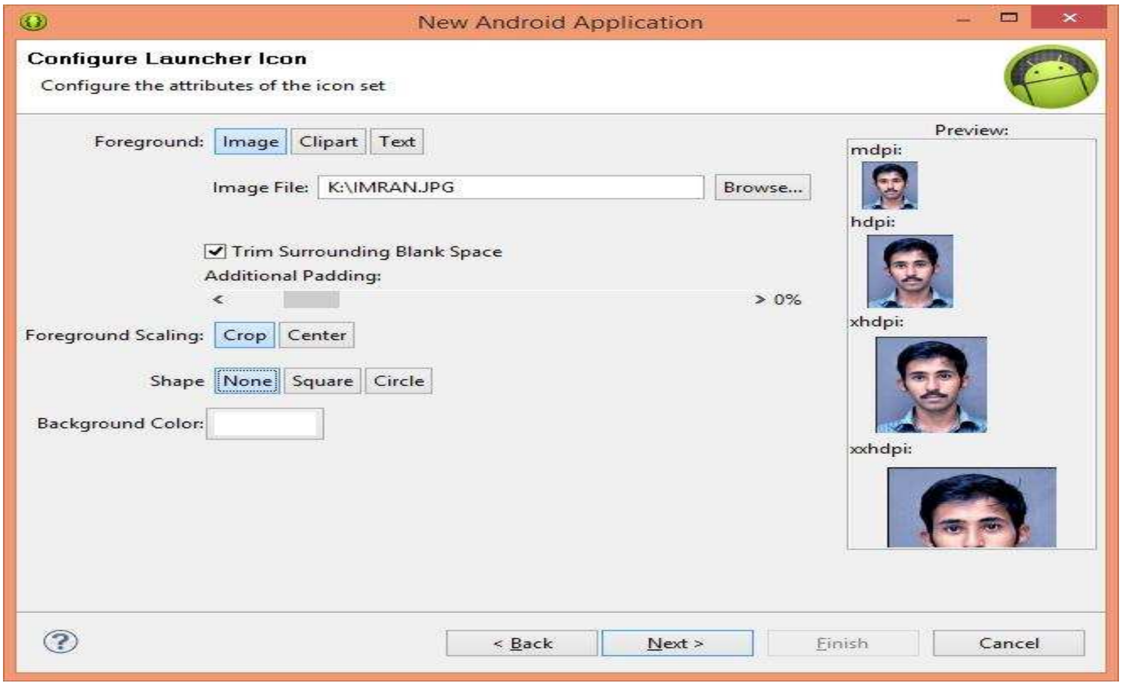
Target SDK: API 17: Android 4.2 (Jelly Bean)

Compile With: API 17: Android 4.2 (Jelly Bean)

Theme: Holo Light with Dark Action Bar

The application name is shown in the Play Store, as well as in the Manage Application list in Settings.

Step 2: In the next window make sure the select the **Application Launcher Icon** option is selected in order to create a new activity for your project, and click **Next**.



New Android Application

Configure Launcher Icon

Configure the attributes of the icon set

Foreground: Image Clipart Text

Image File: K:\IMRAN.JPG

☑ Trim Surrounding Blank Space

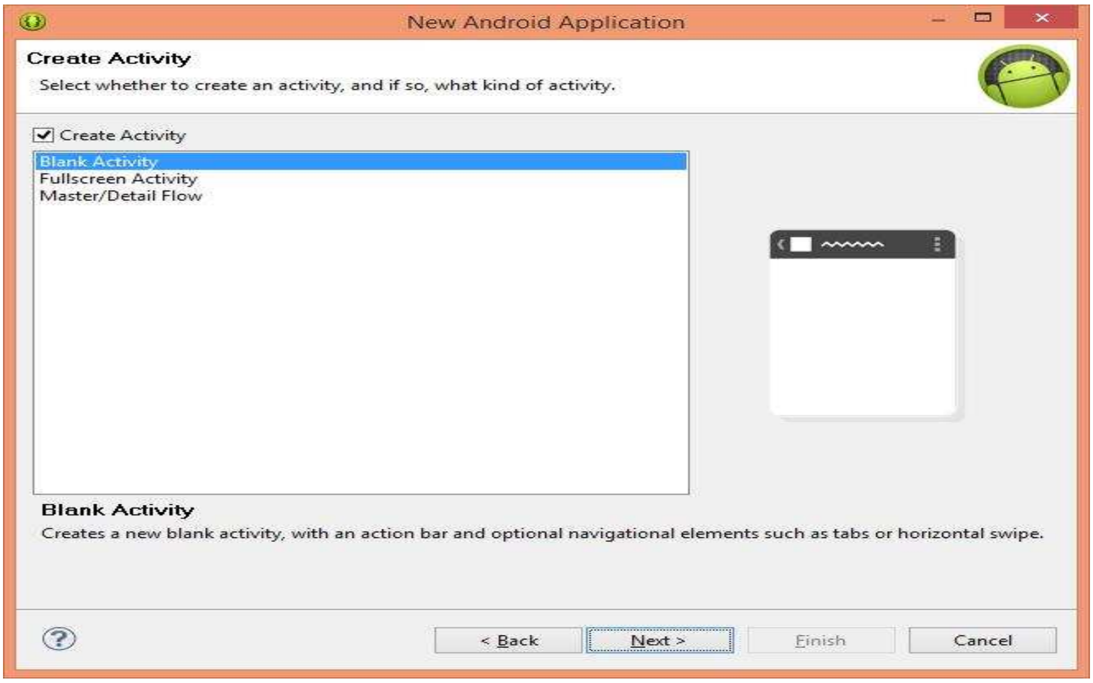
Additional Padding: 0%

Foreground Scaling: Crop Center

Shape None Square Circle

Background Color:

Step 3: On the next window click to select the **Blank Activity** and click **Next**.



New Android Application

Create Activity

Select whether to create an activity, and if so, what kind of activity.

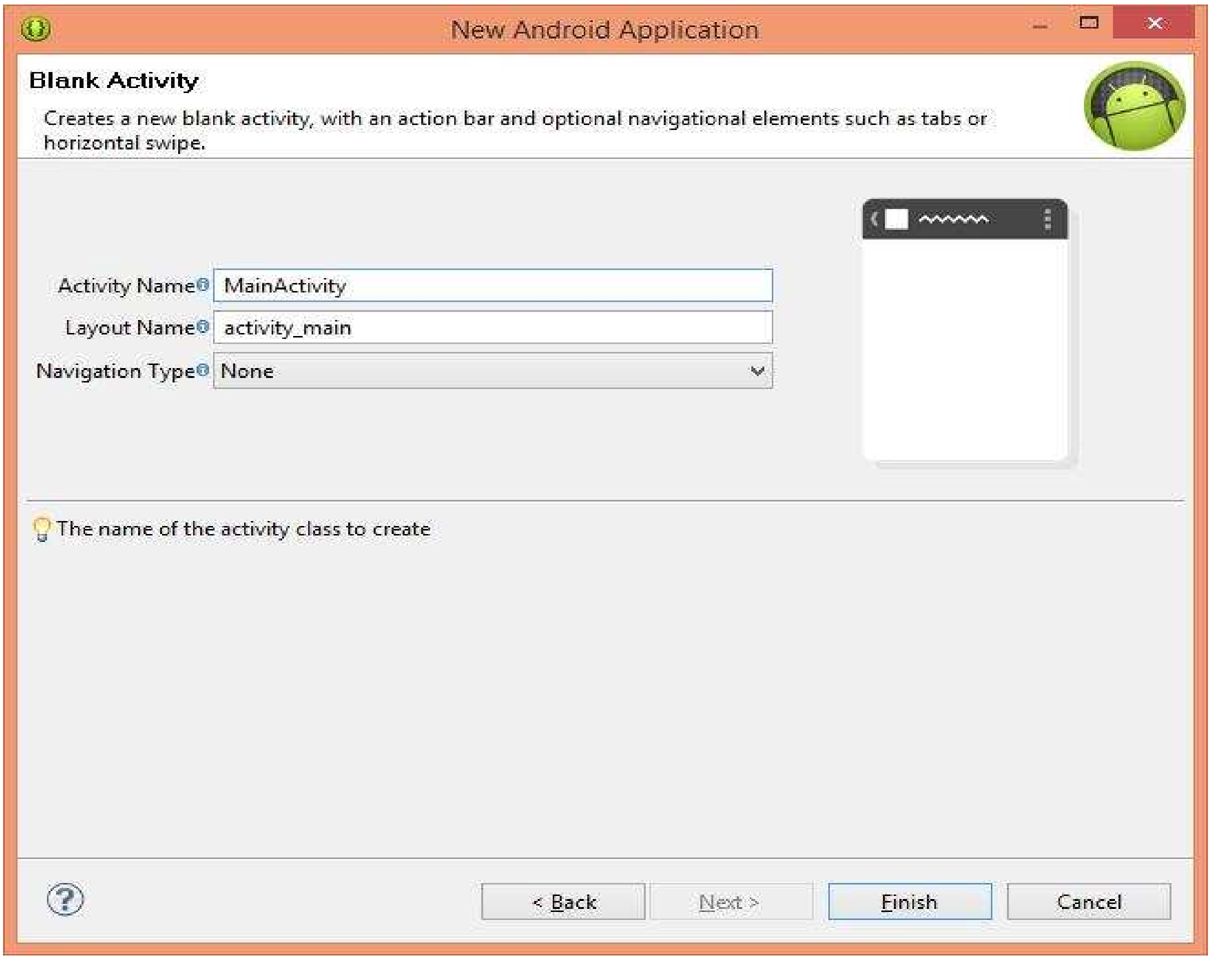
☑ Create Activity

Blank Activity
Fullscreen Activity
Master/Detail Flow

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Step 4: On the next window, name the specify **Activity Name, Layout Name and Navigation Type,** click **Next.**



Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

Activity Name: MainActivity

Layout Name: activity_main

Navigation Type: None

The name of the activity class to create

Step 5: Drag and Drop All the needed Components from the Palette Window to the **Design View of** the **Activity main.xml.**

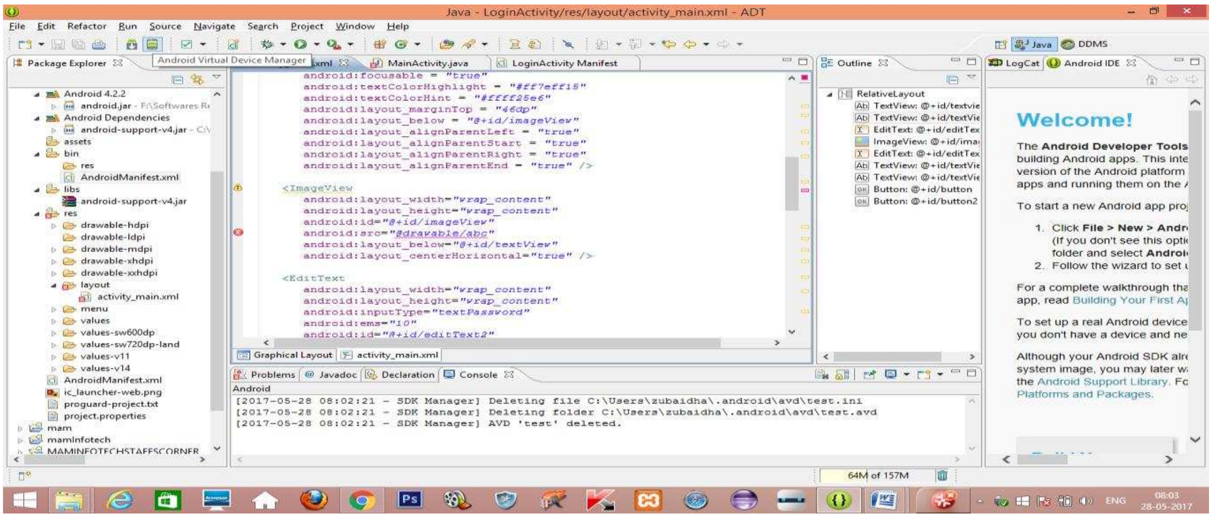
Step 6: Create a new activity named **Success.xml,** and add some content.

Step 7: Code the operations that are want to perform in the **MainActivity.java, of**
the preferred **.Xml** files.

Step 8: On the **Mainfest.xml,** file the permission for accessing phone dialer has to obtained.

Step 9: Create a new activity named **Success.xml,** and add some content for mobility.

Step 10: Create an **AVD, by** Clicking the **Android Virtual Device Icon, on** the **Toolbar.**



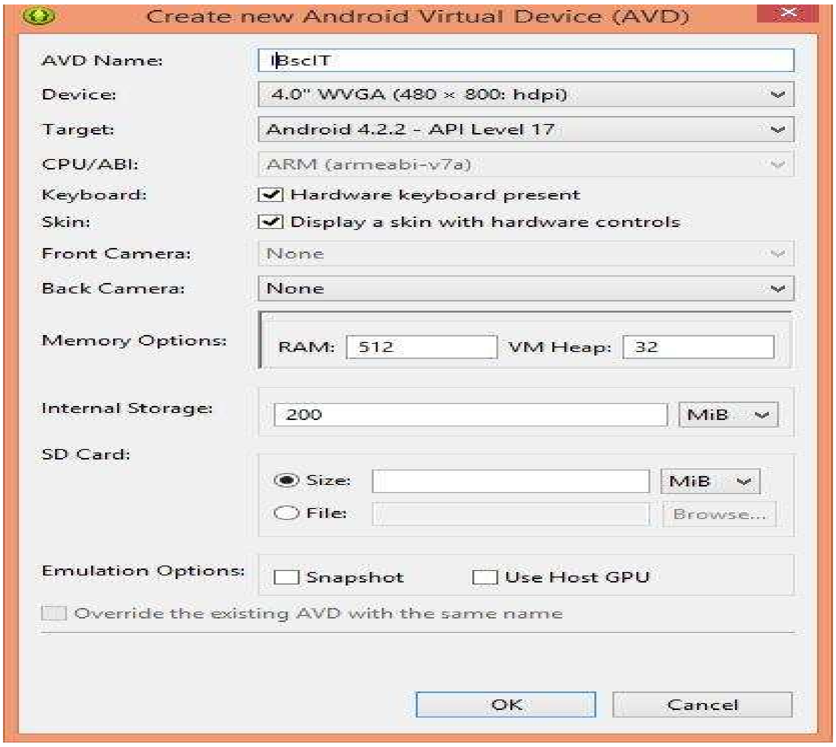
Java - LoginActivity/res/layout/activity_main.xml - ADT

```
        android:focusable = "true"
        android:textColorHighlight = "#ff7eff15"
        android:textColorHint = "#ffff25e6"
        android:layout_marginTop = "46dp"
        android:layout_below = "@+id/imageView"
        android:layout_alignParentLeft = "true"
        android:layout_alignParentStart = "true"
        android:layout_alignParentRight = "true"
        android:layout_alignParentEnd = "true" />

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/imageView"
        android:src="@drawable/abc"
        android:layout_below="@+id/textView"
        android:layout_centerHorizontal="true" />

    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:ems="10"
        android:id="@+id/editText2"
```

```
[2017-05-28 08:02:21 - SDK Manager] Deleting file C:\Users\zubaidha\.android\avd\test.ini
[2017-05-28 08:02:21 - SDK Manager] Deleting folder C:\Users\zubaidha\.android\avd\test.avd
[2017-05-28 08:02:21 - SDK Manager] AVD 'test' deleted.
```

Step 11: On the window, select **New.**

Step 12: On the next window, specify **AVD Name, Device, Target** and **Other Details** and Click **OK.**



Create new Android Virtual Device (AVD)

AVD Name: BscIT

Device: 4.0" WVGA (480 × 800: hdpi)

Target: Android 4.2.2 - API Level 17

CPU/ABI: ARM (armeabi-v7a)

Keyboard: ☑ Hardware keyboard present

Skin: ☑ Display a skin with hardware controls

Front Camera: None

Back Camera: None

Memory Options: RAM: 512 VM Heap: 32

Internal Storage: 200 MiB

SD Card: ● Size: MiB ○ File: Browse...

Emulation Options: □ Snapshot □ Use Host GPU

□ Override the existing AVD with the same name

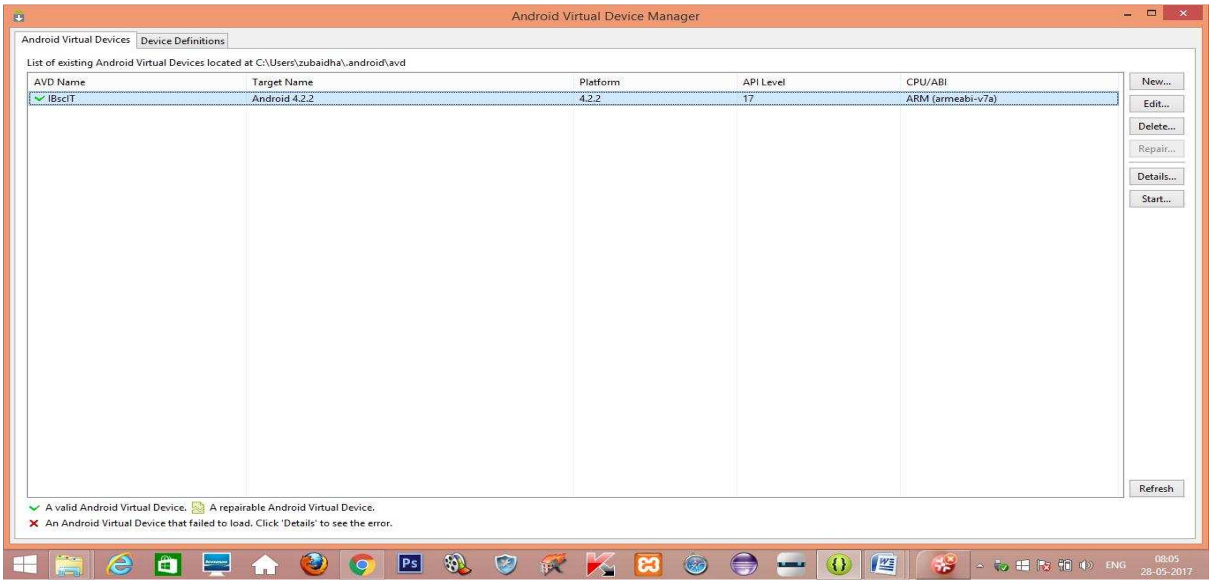
Step 13: Now a new **AVD** is created**.**

Step 14: On Completion of the **Project,** select **File ->Save All.**

Step 15: Select **Debug Icon** on the **Toolbar,** to debug the application.

Step 16: To run the application Select, **Run Icon** on the **Toolbar.**

Step 17: On the Android **Virtual Device Manager** window, select **AVD Name and**, Click **Start.**



List of existing Android Virtual Devices located at C:\Users\zubaidha\.android\avd

AVD Name	Target Name	Platform	API Level	CPU/ABI
✓ IBscIT	Android 4.2.2	4.2.2	17	ARM (armeabi-v7a)

✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

Step 18: On the Launch **Options** window, Check Wipe **User Data and**, Click **Launch.**

Program:

Activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

    <TextView
           android:id="@+id/textview"
           android:layout_width="wrap_content"
           android:layout_height="wrap_content"
           android:layout_alignParentTop="true"
           android:layout_centerHorizontal="true"
           android:text="Phone Dialer"
           android:textSize="35dp" />

   <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textview"
        android:layout_centerHorizontal="true"
        android:text="II B.sc IT"
        android:textColor="#ff7aff24"
        android:textSize="35dp" />

      <ImageView
        android:id="@+id/imageView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textView"
        android:layout_centerHorizontal="true"
        android:src="@drawable/ic_launcher" />

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/imageView"
        android:layout_alignParentBottom="true"
        android:layout_marginBottom="104dp"
        android:text="Call" />

    <EditText
        android:id="@+id/editText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignRight="@+id/textview"
        android:layout_centerVertical="true"
        android:ems="10"
        android:hint="Please Enter 10 Numbers"
        android:inputType="number" >

        <requestFocus />
    </EditText>

  </RelativeLayout>
```

MainActivity.java

```
package com.example.phonecaller;

import android.net.Uri;
import android.os.Bundle;
import android.app.Activity;
import android.content.Intent;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends Activity {
  EditText edittext1;
  Button button1;
  @Override
  protected void onCreate(Bundle savedInstanceState)
    { super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    //Getting the edittext and button instance
    edittext1=(EditText)findViewById(R.id.editText1);
    button1=(Button)findViewById(R.id.button1);

    //Performing action on button click
    button1.setOnClickListener(new OnClickListener(){

      @Override
      public void onClick(View arg0) {
        String number=edittext1.getText().toString();
        Intent callIntent = new Intent(Intent.ACTION_CALL);
        callIntent.setData(Uri.parse("tel:"+number));
        startActivity(callIntent);
      }

    });
  }

  @Override
```

```
   public boolean onCreateOptionsMenu(Menu menu) {
      // Inflate the menu; this adds items to the action bar if it is present.
     // getMenuInflater().inflate(R.menu.activity_main, menu);
      return true;
   }
```

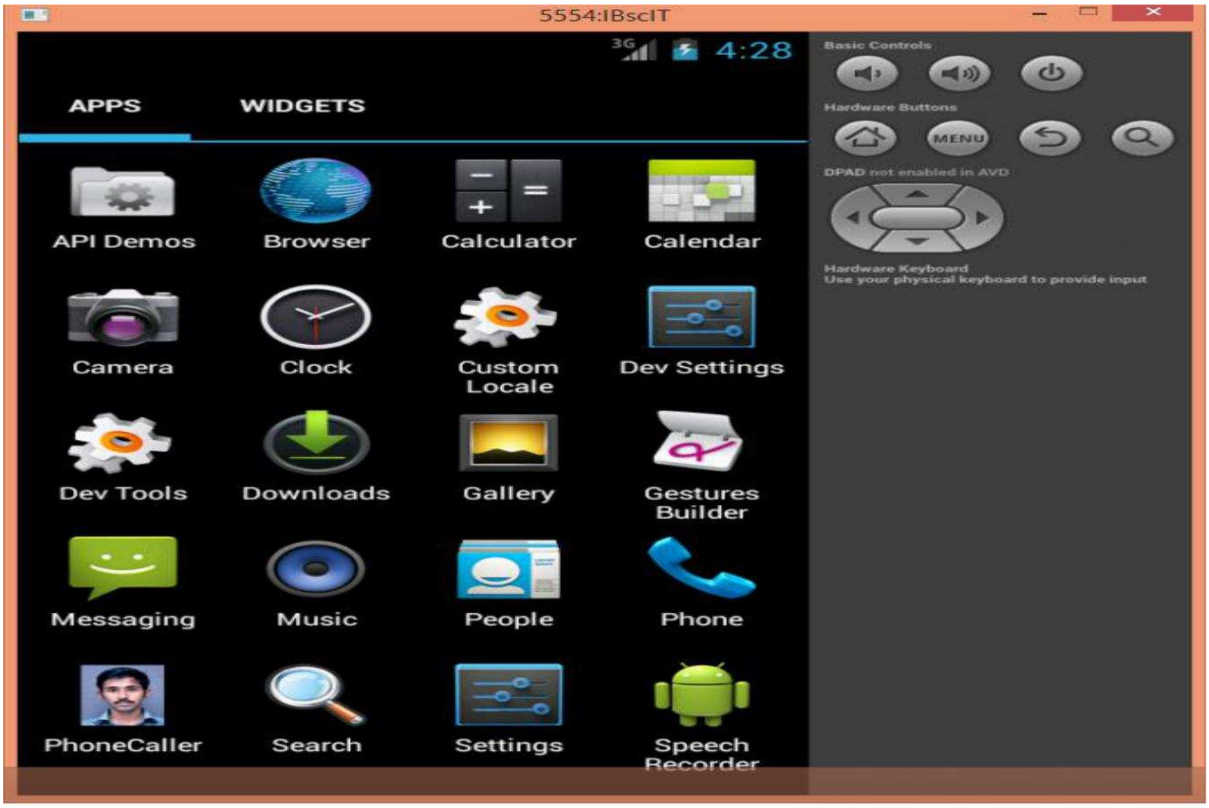
Manifest.xml

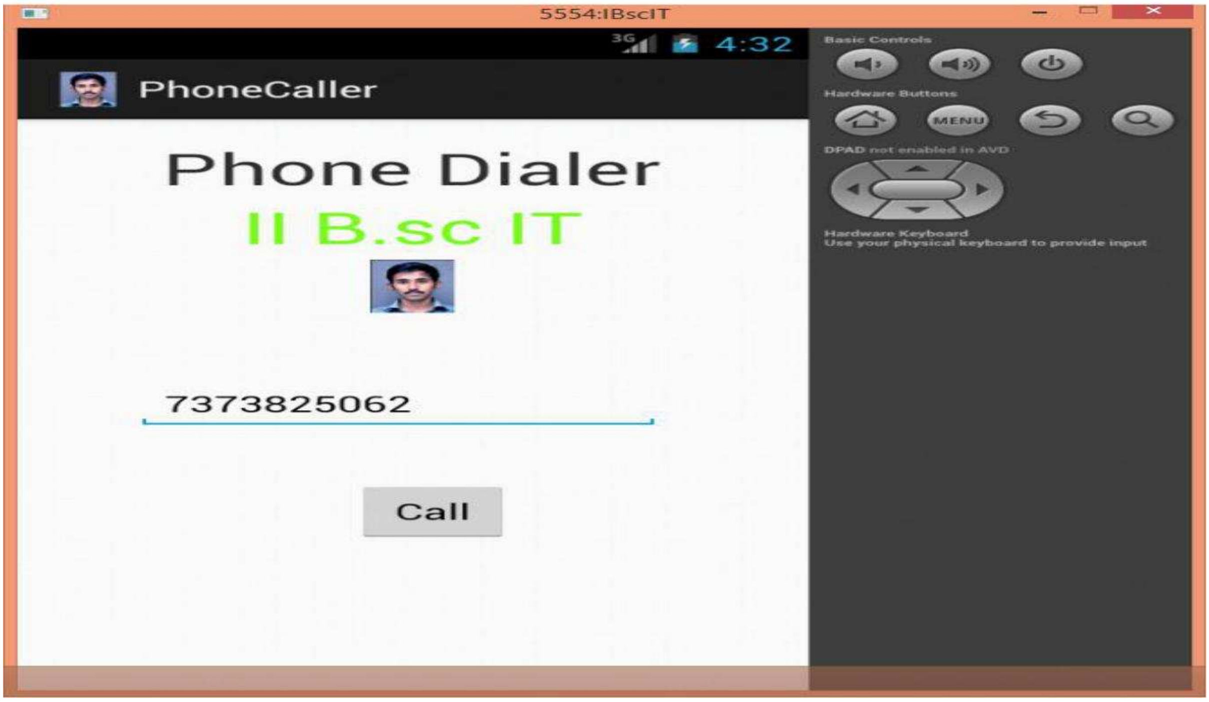
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.phonecaller"
   android:versionCode="1"
   android:versionName="1.0" >

   <uses-sdk
      android:minSdkVersion="8"
      android:targetSdkVersion="17" />

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

   <application
      android:allowBackup="true"
      android:icon="@drawable/ic_launcher"
      android:label="@string/app_name"
      android:theme="@style/AppTheme" >
      <activity
         android:name="com.example.phonecaller.MainActivity"
         android:label="@string/app_name" >
         <intent-filter>
            <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

Strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

   <string name="app_name">PhoneCaller</string>
   <string name="action_settings">Settings</string>
   <string name="hello_world">Hello world!</string>

</resources
```

Output:



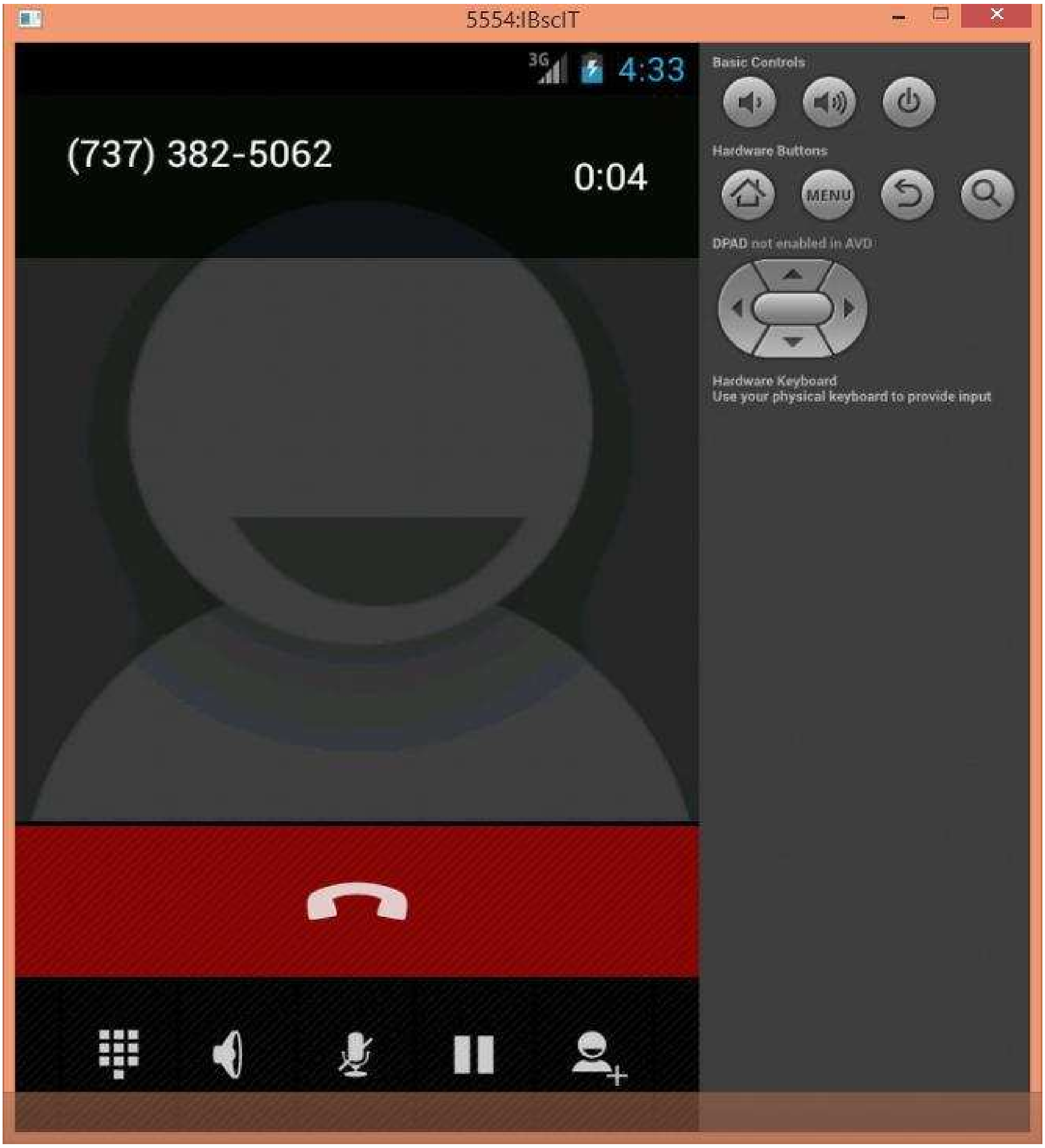


Phone Dialer

II B.sc IT

7373825062

Call



Result:

The above aim of the program has been achieved successfully

8. DEVELOP APPLICATION THAT USES GPS LOCATION INFORMATION.

Aim:

To create an android application to develop an application that uses GPS location information.

Procedure:

Step 1: Open eclipse or android studio and select new android project
Step 2: Give project name and select next
Step 3: Choose the android version. Choose the lowest android version (Android 2.2) and select next
Step 4: Enter the package name. Package name must be two word separated by comma and click finish
Step 5: Go to package explorer in the left hand side. Select our project.
Step 6: Go to res folder and select layout. Double click the main.xml file. Add the code

Source code:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
        android:id="@+id/relativeLayout1"
        android:layout_width="match_parent"
        android:layout_height="match_parent" >
<Button
        android:id="@+id/show_Location"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content
        android:text="Show_Location"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true"
        />
</RelativeLayout>
```

```
package gps.location;
//import android.R;
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
public class GPSlocationActivity extends Activity {

Button btnShowLocation;
GPStrace gps;
```

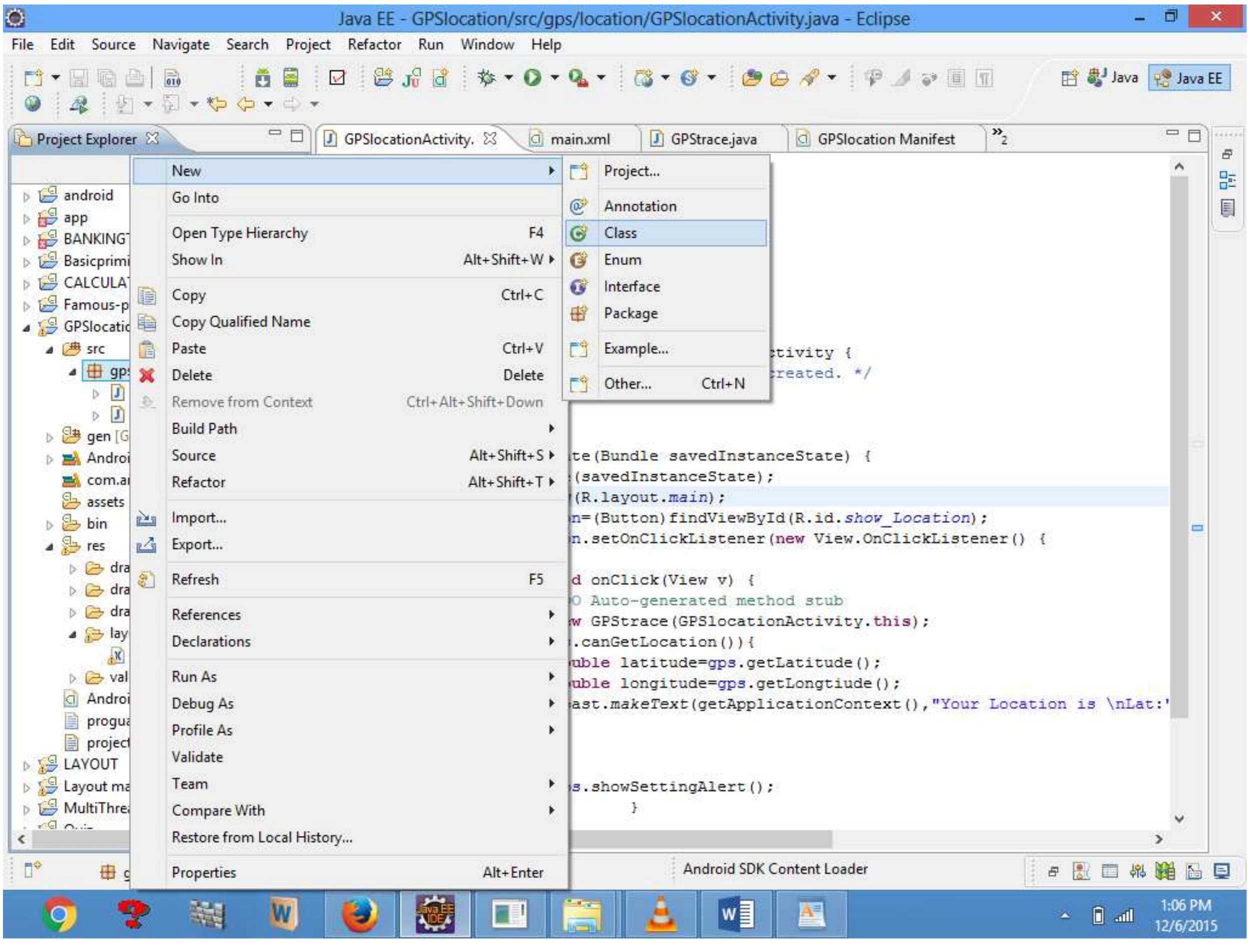
```
@Override
public void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.main);
btnShowLocation=(Button)findViewById(R.id.show_Location);
btnShowLocation.setOnClickListener(new View.OnClickListener() {
```

Android Application Development ***71***

```
@Override

public void onClick(View v) {
// TODO Auto-generated method stub
gps=new GPStrace(GPSlocationActivity.this);
if(gps.canGetLocation()){
double latitude=gps.getLatitude();
double longitude=gps.getLongtiude();
Toast.makeText(getApplicationContext(),"Your Location is

\nLat:"+latitude+"\nLong:"+longitude, Toast.LENGTH_LONG).show();
}
else
{
gps.showSettingAlert();
}
} });  } }
```

)

Go to src folder and Right Click on your package folder and choose new class and give the class nams as GPStrace



```
tivity {
reated. */
te(Bundle savedInstanceState) {
(savedInstanceState);
(R.layout.main);
n=(Button)findViewById(R.id.show_Location);
n.setOnClickListener(new View.OnClickListener() {
d onClick(View v) {
O Auto-generated method stub
w GPStrace(GPSlocationActivity.this);
.canGetLocation()){
uble latitude=gps.getLatitude();
uble longitude=gps.getLongtiude();
ast.makeText(getApplicationContext(),"Your Location is \nLat:'
s.showSettingAlert();
    }
```

Step 9: Select the GPStrace.java file and paste the following code.

```
package gps.location;
import android.app.AlertDialog;
import android.app.Service;
import android.content.Context;
import android.content.DialogInterface;
```

```
import android.content.Intent;

import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.IBinder;
import android.provider.Settings;
public class GPStrace extends Service implements LocationListener{
private final Context context;
boolean isGPSEnabled=false;
boolean canGetLocation=false;
boolean isNetworkEnabled=false;
Location location;
```

```
double latitude;
double longtitude;
private static final long MIN_DISTANCE_CHANGE_FOR_UPDATES=10;
private static final long MIN_TIME_BW_UPDATES=1000*60*1;
protected LocationManager locationManager;
public GPStrace(Context context)
{
        this.context=context;
        getLocation();
}
public Location getLocation()
{
try{
locationManager=(LocationManager) context.getSystemService(LOCATION_SERVICE);
isGPSEnabled=locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER);
isNetworkEnabled=locationManager.isProviderEnabled(LocationManager.NETWORK_P
ROVIDER);
if(!isGPSEnabled && !isNetworkEnabled){
}else{
this.canGetLocation=true;
if(isNetworkEnabled){
locationManager.requestLocationUpdates(
LocationManager.NETWORK_PROVIDER,
MIN_TIME_BW_UPDATES,
MIN_DISTANCE_CHANGE_FOR_UPDATES,this);
}
if(locationManager!=null){
location=locationManager.getLastKnownLocation(LocationManager.NETWORK_PROVI
DER)
;
if(location !=null){
latitude=location.getLatitude();
longtitude=location.getLongitude();
}
}
}
if(isGPSEnabled){
if(location==null){
locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,MIN_TIM
E_B
W_UPDATES, MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
if(locationManager!=null){
location=locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
```

```
if(location!=null){

latitude=location.getLatitude();
longtitude=location.getLongitude();
}
}
}
}
}

catch(Exception e)
{
e.printStackTrace();
}
return location;
}
public void stopUsingGPS(){
if(locationManager!=null){
locationManager.removeUpdates(GPStrace.this);
}
}
public double getLatitude(){
if(location!=null){
latitude=location.getLatitude();
}
return latitude;
}
public double getLongtiude(){
if(location!=null){
longtitude=location.getLatitude();
}
return longtitude;
}
public boolean canGetLocation(){
return this.canGetLocation;
}
public void showSettingAlert(){
AlertDialog.Builder alertDialog=new AlertDialog.Builder(context);
alertDialog.setTitle("GPS is settings");
alertDialog.setMessage("GPS is not enabled.Do you want to go to setting menu?");
alertDialog.setPositiveButton("settings", new DialogInterface.OnClickListener() {
@Override
public void onClick(DialogInterface dialog,int which){
Intent intent=new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
context.startActivity(intent);
}
});
alertDialog.setNegativeButton("cancel", new DialogInterface.OnClickListener() {
@Override
public void onClick(DialogInterface dialog, int which) {
// TODO Auto-generated method stub
dialog.cancel();
}
```

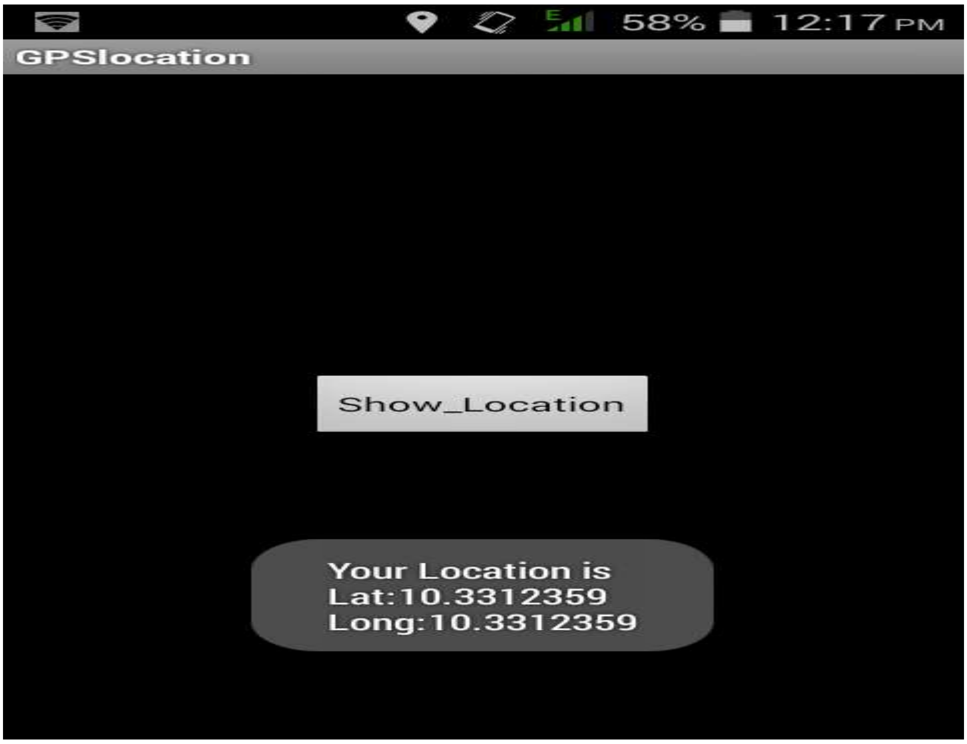
```
});


alertDialog.show();
}
@Override
public void onLocationChanged(Location location) {

// TODO Auto-generated method stub
}

@Override
public void onProviderDisabled(String provider) {

// TODO Auto-generated method stub
}
```

```
@Override
public void onProviderEnabled(String provider) {
// TODO Auto-generated method stub
}
@Override
public void onStatusChanged(String provider, int status, Bundle extras) {
// TODO Auto-generated method stub
}
@Override
public IBinder onBind(Intent intent) {
// TODO Auto-generated method stub
return null;
}
}
```

Step 10: Go to manifest.xml file and add the code below

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

- **Now go to main.xml and right click .select run as option and select run configuration.**
- **Android output is present in the android emulator as shown in below.**



Your Location is
Lat:10.3312359
Long:10.3312359

Result:

The above aim of the program has been achieved successfully.