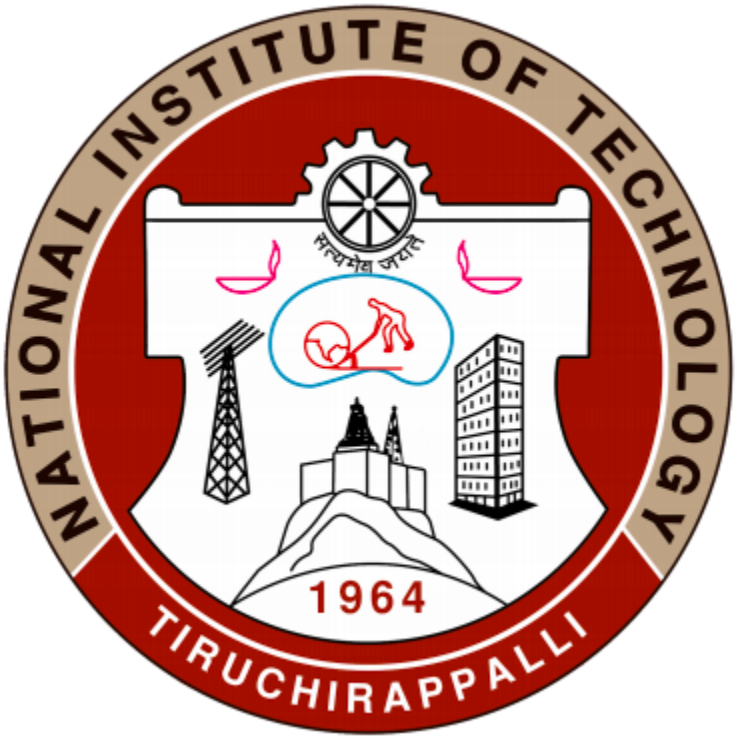
Mobile Application Development Laboratory
Model Exam



Application Apk:
https://drive.google.com/file/d/1v7Ss4fzq7teHpr_FopG8whpUuWqEW5gn/view?usp=sharing

RVS SATYANAND
106118083
CSE A

Mobile Application Development Laboratory – Model Exam

Experiment Name: To develop single player and multiplayer dice game.

Date: 26-04-2021

Aim: Design Dice game application using Android Studio with following functionalities.

A. When the user opens app, have two buttons – one for player vs player; another for player vs computer. Upon selecting one button, ask for player name/names based upon the mode selected. Upon submission, redirect to an activity that has two dices and two buttons (one button in case of player vs computer). When a user presses the button means he/she is rolling the dice. Visualize the rolling of dice using series of images. Each user gets 3 chances in total. Chances are given one after another(alternatively). The player who is getting the highest sum after all 3 chances is the winner. Display the score of both players while playing and Display the winner with player name in both the modes once the game is over.

B. Have menu option on top with menu items
    1. Exit – exit the application
    2. Help – Redirect to google search application with search input “Dice games”
    3. Refresh – Reset Scores to zero and start

C. Use toasts atleast twice wherever necessary.

Technical Concepts Used :

➢ **XML Designing ->** Using constraint layout and material.io library

➢ **Intents and Intent Transition/ Implicit and Explicit Intents ->** To move from one activity to another activity like if student select singleMode then redirecting to activity where single player is handled else redirecting to activity where multiplayer is accessed, along with that passed data like player1 name and player2 name through intent (by using intent.putExtra)

➢ **Menu in action bar ->** To let users to select operation like home(Which will redirect them to mainActivity), refresh(Which will reset the game), Help(Which will open google search activity with “Dice Game” as query) and Exit (Which will quit the game).

➢ **MenuInflator ->** Used it to Inflate the xml designing of menu onto to java code. This is very important without this I cannot access methods and properties in java code.

➢ **Toasts ->** Showed messages like, player name should be at least 4 chars long and both players name should not match.

➢ **AnimationUtils, Animation ->** Used to Visualize the rolling of dice.

➢ `handler.postDelayed(new Runnable(){})`

1. `Handler class->` This class's object is used to postDelay() a part of code, that is run a code after some time lag. I used it for **computer** chance so that users can get a feel of the game.

Source Code: The link of my app zip

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity"
   android:layout_margin="20dp"
   android:orientation="vertical">

   <TextView
       android:id="@+id/textGreet"
       android:layout_width="match_parent"
       android:layout_height="wrap_content"
       android:gravity="center"
       android:padding="10dp"
       android:text="Game Mode!"
       android:textColor="@color/black"
       android:textSize="32sp"
       android:textStyle="bold|italic" />

   <LinearLayout
       android:layout_width="match_parent"
       android:layout_height="match_parent"
       android:gravity="center_vertical"
       android:orientation="vertical">
```

```
        <androidx.appcompat.widget.AppCompatButton
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginTop="40dp"
            android:layout_marginBottom="10dp"
            android:background="@drawable/btn_state_green_gray"
            android:paddingStart="50dp"
            android:paddingTop="10dp"
            android:paddingEnd="50dp"
            android:onClick="singlePlayer"
            android:paddingBottom="10dp"
            android:text="Single Player"
            android:textColor="@color/white"
            android:textSize="18sp"
            android:textStyle="bold" />

        <androidx.appcompat.widget.AppCompatButton
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:onClick="multiPlayer"
            android:layout_marginTop="40dp"
            android:layout_marginBottom="10dp"
            android:background="@drawable/btn_state_green_gray"
            android:paddingStart="50dp"
            android:paddingTop="10dp"
            android:paddingEnd="50dp"
            android:paddingBottom="10dp"
            android:text="Multi Player"
            android:textColor="@color/white"
            android:textSize="18sp"
            android:textStyle="bold" />

    </LinearLayout>
</LinearLayout>
```

activity_name.xml

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

```
    android:layout_height="match_parent"
    android:layout_margin="20dp"
    android:orientation="vertical"
    tools:context=".NameActivity">

    <TextView
        android:id="@+id/textGreet"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:padding="10dp"
        android:text="Enter Player Name"
        android:textColor="@color/black"
        android:textSize="24sp"
        android:textStyle="bold|italic" />


    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:orientation="vertical">

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="10dp"
            android:paddingLeft="3dp"
            android:paddingRight="3dp"
            android:textColor="@color/white"
            android:textSize="18sp"
            android:textStyle="bold|italic">


<com.google.android.material.textfield.TextInputEditText
                android:id="@+id/player1"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:autofillHints="Name"
                android:background="@color/very_light_gray"
```

```
                android:hint="Player-1 Name"
                android:imeOptions="actionNext"
                android:inputType="text"
                android:lines="1"
                android:singleLine="true"
                android:textColor="@color/black"
                android:textSize="20sp"
                android:textStyle="bold|italic" />


</com.google.android.material.textfield.TextInputLayout>


        <LinearLayout
            android:id="@+id/parentPlayer2"
            android:layout_width="match_parent"
            android:layout_height="match_parent"

            android:orientation="vertical">


<com.google.android.material.textfield.TextInputLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_margin="10dp"

                android:paddingLeft="3dp"
                android:paddingRight="3dp"
                android:textColor="@color/white"
                android:textSize="18sp"
                android:textStyle="bold|italic">


<com.google.android.material.textfield.TextInputEditText
                    android:id="@+id/player2"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:autofillHints="Name"
                    android:background="@color/very_light_gray"
                    android:hint="Player-2 Name"
```

```
                    android:imeOptions="actionNext"
                    android:inputType="text"
                    android:lines="1"
                    android:singleLine="true"
                    android:text=""
                    android:textColor="@color/black"
                    android:textSize="20sp"
                    android:textStyle="bold|italic" />


</com.google.android.material.textfield.TextInputLayout>
        </LinearLayout>


    </LinearLayout>

    <androidx.appcompat.widget.AppCompatButton
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="40dp"
        android:layout_marginBottom="10dp"
        android:background="@drawable/btn_state_green_gray"
        android:onClick="Submit"
        android:paddingStart="50dp"
        android:paddingTop="10dp"
        android:paddingEnd="50dp"
        android:paddingBottom="10dp"
        android:text="Submit"
        android:textColor="@color/white"
        android:textSize="18sp"
        android:textStyle="bold" />
</LinearLayout>
```

activity_game.xml

```
<?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"

    tools:context=".GameActivity">


    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center_horizontal"
        android:orientation="vertical">

        <TextView
            android:id="@+id/turns"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:layout_marginTop="25dp"
            android:gravity="center_horizontal"
            android:text="Turns Left = 3"
            android:textColor="#48810A"
            android:textSize="32dp"
            android:textStyle="bold" />
    </LinearLayout>


    <TextView
        android:id="@+id/tvp1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="P1 = 0"
        android:textColor="#48810A"
        android:textSize="32dp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@id/tvp2"
```

```
        app:layout_constraintHorizontal_bias="0.1"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.1" />

    <TextView
        android:id="@+id/tvp2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="P2 = 0"
        android:textColor="#48810A"
        android:textSize="32dp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.9"
        app:layout_constraintStart_toEndOf="@id/tvp1"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.1" />

    <ImageView
        android:id="@+id/imgDice"
        android:layout_width="230dp"
        android:layout_height="230dp"
        android:scaleType="fitXY"
        app:layout_constraintBottom_toTopOf="@id/btnRoll"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tvp1"
        app:srcCompat="@drawable/fulldice" />

    <Button
        android:id="@+id/btnRoll"
        android:layout_width="300dp"
        android:layout_height="wrap_content"
        android:backgroundTint="#200E2C"
        android:text="P1 Turn"
        android:textColor="@color/white"
        android:textSize="24dp"
        android:textStyle="bold"
```

```
        app:layout_constraintBottom_toTopOf="@id/btnReset"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/imgDice"
        app:layout_constraintVertical_bias="0.9" />

    <Button
        android:id="@+id/btnReset"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:backgroundTint="#200E2C"
        android:text="@string/reset"
        android:textColor="@color/white"
        android:textSize="24dp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/btnRoll"
        app:layout_constraintVertical_bias="0.9" />

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

activity_multi.xml

```
<?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"

    tools:context=".MultiActivity">


    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center_horizontal"
```

```
        android:orientation="vertical">

        <TextView
            android:id="@+id/turns"
            android:layout_width="match_parent"
            android:layout_height="461dp"
            android:layout_gravity="center_horizontal"
            android:layout_marginTop="25dp"
            android:gravity="center_horizontal"
            android:text="Turns Left = 3"
            android:textColor="#48810A"
            android:textSize="32dp"
            android:textStyle="bold" />

    </LinearLayout>


    <TextView
        android:id="@+id/tvp1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="P1 = 0"
        android:textColor="#48810A"
        android:textSize="32dp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@id/tvp2"
        app:layout_constraintHorizontal_bias="0.1"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.1" />

    <TextView
        android:id="@+id/tvp2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="P2 = 0"
        android:textColor="#48810A"
        android:textSize="32dp"
        android:textStyle="bold"
```

```
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.9"
        app:layout_constraintStart_toEndOf="@id/tvp1"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.1" />

    <ImageView
        android:id="@+id/imgDice"
        android:layout_width="180dp"
        android:layout_height="184dp"
        android:scaleType="fitXY"
        app:layout_constraintBottom_toTopOf="@id/btnRoll"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tvp1"
        app:srcCompat="@drawable/fulldice" />


    <ImageView
        android:id="@+id/img2Dice"
        android:layout_width="170dp"
        android:layout_height="173dp"
        android:scaleType="fitXY"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toEndOf="@+id/imgDice"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/fulldice" />


    <Button
        android:id="@+id/btnRoll"
        android:layout_width="300dp"
        android:layout_height="wrap_content"
        android:backgroundTint="#200E2C"
        android:onClick="btnOnlick"
        android:text="P1 Turn"
        android:textColor="@color/white"
        android:textSize="24dp"
        android:textStyle="bold"
```

```
        app:layout_constraintBottom_toTopOf="@id/btnReset"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/imgDice"
        app:layout_constraintVertical_bias="0.9" />


    <Button
        android:id="@+id/btnRoll2"
        android:layout_width="300dp"
        android:layout_height="wrap_content"
        android:backgroundTint="#200E2C"
        android:onClick="btnOnlick2"
        android:text="P2 Turn"

        android:textColor="@color/white"
        android:textSize="24dp"
        android:textStyle="bold"
        app:layout_constraintBottom_toTopOf="@id/btnReset"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/imgDice"
        app:layout_constraintVertical_bias="0.9" />

    <Button
        android:id="@+id/btnReset"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:backgroundTint="#200E2C"
        android:text="@string/reset"
        android:textColor="@color/white"
        android:textSize="24dp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/btnRoll"
        app:layout_constraintVertical_bias="0.9" />

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

menu.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<menu
xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/home"
        android:title="Home" />

    <item
        android:id="@+id/action_help"
        android:title="Help" />
    <item
        android:id="@+id/action_refresh"
        android:title="Refresh" />
    <item
        android:id="@+id/action_exit"
        android:title="Exit" />
</menu>
```

MainActivity.java

```java
package com.example.modelexam;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {

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

    public void singlePlayer(View view) {
        Intent i = new Intent(this, NameActivity.class);
```

```
        i.putExtra("mode", "single");
        i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);

        startActivity(i);
    }

    public void multiPlayer(View view) {
        Intent i = new Intent(this, NameActivity.class);
        i.putExtra("mode", "multi");
        i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
        startActivity(i);
    }
}
```

NameActivity.java

```
package com.example.modelexam;

import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.Toast;

public class NameActivity extends AppCompatActivity {

    LinearLayout parentId;
    String type, pl1Name, pl2Name;
    EditText player1, player2;

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

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

        Intent intent = getIntent();
        Bundle extras = intent.getExtras();

        type = extras.getString("mode");

        Toast.makeText(this, type + " player mode", 0).show();

        player1 = findViewById(R.id.player1);
        player2 = findViewById(R.id.player2);
        parentId = findViewById(R.id.parentPlayer2);

        if (type.equals("single")) {
            parentId.setVisibility(View.INVISIBLE);
            pl2Name = "Computer";
        }
    }

    public void Submit(View view) {
        pl1Name = player1.getText().toString();
        String temp = player2.getText().toString();

        if (temp.length() > 0) {
            pl2Name = temp;
        }

        if (pl1Name.length() <= 3) {

            Toast.makeText(this, "Player-1 name should be
atleast 4 chars long!", 0).show();
            return;
        }

        if (pl2Name.length() <= 3) {

            Toast.makeText(this, "Player-2 name should be
atleast 4 chars long!", 0).show();
            return;
        }
```

```
        if (pl1Name.equals(pl2Name)) {
            Toast.makeText(this, "Player-1 name should be
differnet than Player-2 name!!!", 0).show();
            return;

        }

        if (!pl2Name.equals("Computer")) {
            Intent i = new Intent(this, MultiActivity.class);
            i.putExtra("pl1Name", pl1Name);
            i.putExtra("pl2Name", pl2Name);
            i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
            startActivity(i);
            return;
        }

        Intent i = new Intent(this, GameActivity.class);
        i.putExtra("pl1Name", pl1Name);
        i.putExtra("pl2Name", pl2Name);
        i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
        startActivity(i);
    }
}
```

GameActivity.java

```
package com.example.modelexam;

import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.os.Handler;
import android.view.Menu;
import android.view.MenuInflater;
```

```
import android.view.MenuItem;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import java.util.Random;

public class GameActivity extends AppCompatActivity {


   Random rnd = new Random();
   TextView p1ScoreText, p2ScoreText, turnsView;
   int p1score = 0, p2score = 0;
   int turn = 1;
   int numOfTurns;
   boolean singlePlayerGame = false;

   Button rollBtn, resetBtn;
   ImageView diePic;

   String pl1Name, pl2Name;

   @SuppressLint("WrongConstant")
   @Override
   protected void onCreate(Bundle savedInstanceState) {
       super.onCreate(savedInstanceState);
       setContentView(R.layout.activity_game);

       numOfTurns = 6;

       Intent intent = getIntent();
       Bundle extras = intent.getExtras();
       pl1Name = extras.getString("pl1Name");
       pl2Name = extras.getString("pl2Name");
```

```
        singlePlayerGame = false;
        singlePlayerGame = pl2Name.equals("Computer");

        Toast.makeText(this, "Game Started " + pl1Name + "
chance", 0).show();

        diePic = (ImageView) findViewById(R.id.imgDice);
        p1ScoreText = (TextView) findViewById(R.id.tvp1);
        p2ScoreText = (TextView) findViewById(R.id.tvp2);
        rollBtn = (Button) findViewById((R.id.btnRoll));
        turnsView = findViewById((R.id.turns));


        rollBtn.setText(pl1Name + " TURN");
        resetAll();


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

        resetBtn = (Button) findViewById(R.id.btnReset);
        resetBtn.setOnClickListener(new View.OnClickListener()
{
            @Override
            public void onClick(View v) {
                resetAll();
            }
        });
    }

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

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

            case R.id.home: {
                Intent intent = new Intent(this,
MainActivity.class);
                intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);

                startActivity(intent);
                return true;
            }

            case R.id.action_help: {
                Intent intent = new Intent(Intent.ACTION_VIEW,
Uri.parse("https://www.google.com/search?q=Dice+Games"));
                startActivity(intent);
                return true;
            }
            case R.id.action_refresh: {
                resetAll();
                return true;
            }
            case R.id.action_exit: {
                finish();
                return true;
            }
            default:
                return super.onOptionsItemSelected(item);
        }
    }

    private void resetAll() {
        p1score = 0;
        p2score = 0;
        numOfTurns = 6;
        turnsView.setText("Turns Left = " + ((numOfTurns + 1) /
2));
```

```
        turn = 1;
        p1ScoreText.setText(pl1Name + " = 0");
        p2ScoreText.setText(pl2Name + " = 0");
        diePic.setImageResource(R.drawable.fulldice);
        rollBtn.setText(pl1Name + " Turn");
        rollBtn.setEnabled(true);
    }

    private void diceRotate() {
        int num = rnd.nextInt(6) + 1;


        Animation rolling = AnimationUtils.loadAnimation(this,
R.anim.rotate);
        diePic.setImageResource(R.drawable.fulldice);


        rolling.setAnimationListener(new
Animation.AnimationListener() {
            @Override
            public void onAnimationStart(Animation animation) {
                diePic.setImageResource(R.drawable.fulldice);
                rollBtn.setEnabled(false);
            }

            @Override
            public void onAnimationEnd(Animation animation) {
                rollBtn.setEnabled(true);
                numOfTurns -= 1;
                turnsView.setText("Turns Left = " +
((numOfTurns + 1) / 2));

                switch (num) {
                    case 1:

diePic.setImageResource(R.drawable.die_1);
                        break;
                    case 2:

diePic.setImageResource(R.drawable.die_2);
```

```
                        break;
                    case 3:

diePic.setImageResource(R.drawable.die_3);
                        break;
                    case 4:

diePic.setImageResource(R.drawable.die_4);
                        break;
                    case 5:

diePic.setImageResource(R.drawable.die_5);
                        break;
                    case 6:

diePic.setImageResource(R.drawable.die_6);
                        break;
                }

                if (turn == 1) {
                    p1score += num;
                    String p1text = p11Name + " = " +
Integer.toString(p1score);
                    p1ScoreText.setText(p1text);
                    turn *= -1;
                    rollBtn.setText(p12Name + " Turn");
                    if (numOfTurns <= 0) {
                        Toast.makeText(GameActivity.this,
p11Name + " Won with score " + p1score,
Toast.LENGTH_SHORT).show();
                        turnsView.setText((p11Name + " Won with
score " + p1score));
                        rollBtn.setText("Press Reset");
                        rollBtn.setEnabled(false);
                    } else if (singlePlayerGame) {
                        rollBtn.setEnabled(false);
                        new Handler().postDelayed(new
Runnable() {
                            @Override
                            public void run() {
```

```
                                rollBtn.setEnabled(true);
                                rollBtn.performClick();
                                rollBtn.setEnabled(false);
                            }
                        }, 1000);
                    }


                } else if (turn == -1) {
                    p2score += num;
                    String p2text = pl2Name + " = " +
Integer.toString(p2score);
                    p2ScoreText.setText(p2text);
                    turn *= -1;
                    rollBtn.setText(pl1Name + " Turn");

                    rollBtn.setEnabled(true);

                    if (numOfTurns <= 0) {
                        Toast.makeText(GameActivity.this,
pl2Name + " Won with score " + p2score,
Toast.LENGTH_SHORT).show();
                        turnsView.setText((pl1Name + " Won with
score " + p1score));
                        rollBtn.setText("Press Reset");
                        rollBtn.setEnabled(false);
                    }
                }
            }

            @Override
            public void onAnimationRepeat(Animation animation)
{

            }
        });

        diePic.startAnimation((rolling));
```

```
    }
}
```

MultiActivity.java

```
package com.example.modelexam;

import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.os.Handler;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import java.util.Random;

public class MultiActivity extends AppCompatActivity {


    Random rnd = new Random();
    TextView p1ScoreText, p2ScoreText, turnsView;
    int p1score = 0, p2score = 0;
    int turn = 1;
    int numOfTurns;
    boolean singlePlayerGame = false;

    Button rollBtn, rollBtn2, resetBtn;
    ImageView diePic, diePic2;

    String pl1Name, pl2Name;
```

```
    @SuppressLint("WrongConstant")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_multi);

        numOfTurns = 6;

        Intent intent = getIntent();
        Bundle extras = intent.getExtras();
        pl1Name = extras.getString("pl1Name");
        pl2Name = extras.getString("pl2Name");


        singlePlayerGame = false;
        singlePlayerGame = pl2Name.equals("Computer");

        Toast.makeText(this, "Game Started " + pl1Name + "
chance", 0).show();

        diePic = (ImageView) findViewById(R.id.imgDice);
        diePic2 = (ImageView) findViewById(R.id.img2Dice);

        p1ScoreText = (TextView) findViewById(R.id.tvp1);
        p2ScoreText = (TextView) findViewById(R.id.tvp2);


        rollBtn = (Button) findViewById((R.id.btnRoll));
        rollBtn2 = (Button) findViewById((R.id.btnRoll2));

        turnsView = findViewById((R.id.turns));


        rollBtn.setText(pl1Name + " TURN");
        rollBtn2.setText(pl2Name + " TURN");

        resetAll();
```

```
        resetBtn = (Button) findViewById(R.id.btnReset);
        resetBtn.setOnClickListener(new View.OnClickListener()
{
            @Override
            public void onClick(View v) {
                resetAll();
            }
        });
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.menu.menu, menu);
        return true;
    }

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

            case R.id.home: {
                Intent intent = new Intent(this,
MainActivity.class);
                intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);

                startActivity(intent);
                return true;
            }

            case R.id.action_help: {
                Intent intent = new Intent(Intent.ACTION_VIEW,
Uri.parse("https://www.google.com/search?q=Dice+Games"));
                startActivity(intent);
                return true;
            }
            case R.id.action_refresh: {
                resetAll();
                return true;
```

```
            }
            case R.id.action_exit: {
                finish();
                return true;
            }
            default:
                return super.onOptionsItemSelected(item);
        }
    }

    private void resetAll() {
        p1score = 0;
        p2score = 0;
        numOfTurns = 6;
        turnsView.setText("Turns Left = " + ((numOfTurns + 1) /
2));
        turn = 1;
        p1ScoreText.setText(pl1Name + " = 0");
        p2ScoreText.setText(pl2Name + " = 0");
        diePic.setImageResource(R.drawable.fulldice);
        diePic2.setImageResource(R.drawable.fulldice);
        rollBtn.setText(pl1Name + " Turn");
        rollBtn.setEnabled(true);
    }

    private void diceRotate() {
        int num = rnd.nextInt(6) + 1;


        Animation rolling = AnimationUtils.loadAnimation(this,
R.anim.rotate);
        diePic.setImageResource(R.drawable.fulldice);


        rolling.setAnimationListener(new
Animation.AnimationListener() {
            @Override
            public void onAnimationStart(Animation animation) {
                diePic.setImageResource(R.drawable.fulldice);
            }
```

```
            @Override
            public void onAnimationEnd(Animation animation) {
                numOfTurns -= 1;
                turnsView.setText("Turns Left = " +
((numOfTurns + 1) / 2));

                switch (num) {
                    case 1:

diePic.setImageResource(R.drawable.die_1);
                        break;
                    case 2:

diePic.setImageResource(R.drawable.die_2);
                        break;
                    case 3:

diePic.setImageResource(R.drawable.die_3);
                        break;
                    case 4:

diePic.setImageResource(R.drawable.die_4);
                        break;
                    case 5:

diePic.setImageResource(R.drawable.die_5);
                        break;
                    case 6:

diePic.setImageResource(R.drawable.die_6);
                        break;
                }

                if (turn == 1) {
                    p1score += num;
                    String p1text = p11Name + " = " +
Integer.toString(p1score);
                    p1ScoreText.setText(p1text);
                    turn *= -1;
```

```
                    if (numOfTurns <= 0) {
//                        Toast.makeText(GameActivity.this,
pl1Name + " Won with score " + p1score,
Toast.LENGTH_SHORT).show();
                        turnsView.setText((pl1Name + " Won with
score " + p1score));
                        rollBtn.setText("Press Reset");
                        rollBtn.setEnabled(false);
                        rollBtn2.setText("Press Reset");
                        rollBtn2.setEnabled(false);

                    }


                } else if (turn == -1) {
                    p2score += num;
                    String p2text = pl2Name + " = " +
Integer.toString(p2score);
                    p2ScoreText.setText(p2text);
                    turn *= -1;

                    rollBtn.setEnabled(true);

                    if (numOfTurns <= 0) {
//                        Toast.makeText(GameActivity.this,
pl2Name + " Won with score " + p2score,
Toast.LENGTH_SHORT).show();
                        turnsView.setText((pl1Name + " Won with
score " + p1score));
                        rollBtn.setText("Press Reset");
                        rollBtn.setEnabled(false);
                        rollBtn2.setText("Press Reset");
                        rollBtn2.setEnabled(false);

                    }
                }
            }

            @Override
```

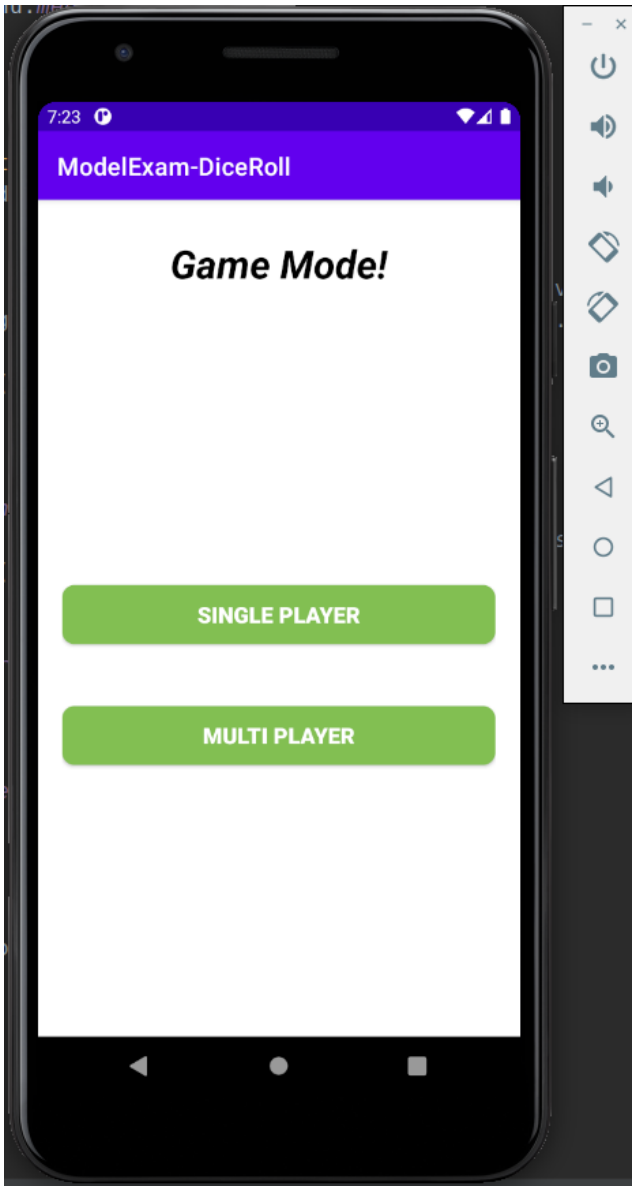
```
            public void onAnimationRepeat(Animation animation)
{

            }
        });

        diePic.startAnimation((rolling));


    }


    private void diceRotate2() {
        int num = rnd.nextInt(6) + 1;


        Animation rolling = AnimationUtils.loadAnimation(this,
R.anim.rotate);
        diePic2.setImageResource(R.drawable.fulldice);


        rolling.setAnimationListener(new
Animation.AnimationListener() {
            @Override
            public void onAnimationStart(Animation animation) {
                diePic2.setImageResource(R.drawable.fulldice);
            }

            @Override
            public void onAnimationEnd(Animation animation) {
                numOfTurns -= 1;
                turnsView.setText("Turns Left = " +
((numOfTurns + 1) / 2));

                switch (num) {
                    case 1:

diePic2.setImageResource(R.drawable.die_1);
                        break;
                    case 2:
```

```
diePic2.setImageResource(R.drawable.die_2);
                        break;
                    case 3:

diePic2.setImageResource(R.drawable.die_3);
                        break;
                    case 4:

diePic2.setImageResource(R.drawable.die_4);
                        break;
                    case 5:

diePic2.setImageResource(R.drawable.die_5);
                        break;
                    case 6:

diePic2.setImageResource(R.drawable.die_6);
                        break;
                }

                if (turn == 1) {
                    p1score += num;
                    String p1text = p11Name + " = " +
Integer.toString(p1score);
                    p1ScoreText.setText(p1text);
                    turn *= -1;
                    if (numOfTurns <= 0) {
//                          Toast.makeText(GameActivity.this,
p11Name + " Won with score " + p1score,
Toast.LENGTH_SHORT).show();
                        turnsView.setText((p11Name + " Won with
score " + p1score));

                        rollBtn.setText("Press Reset");
                        rollBtn.setEnabled(false);
                        rollBtn2.setText("Press Reset");
                        rollBtn2.setEnabled(false);

                    }
```

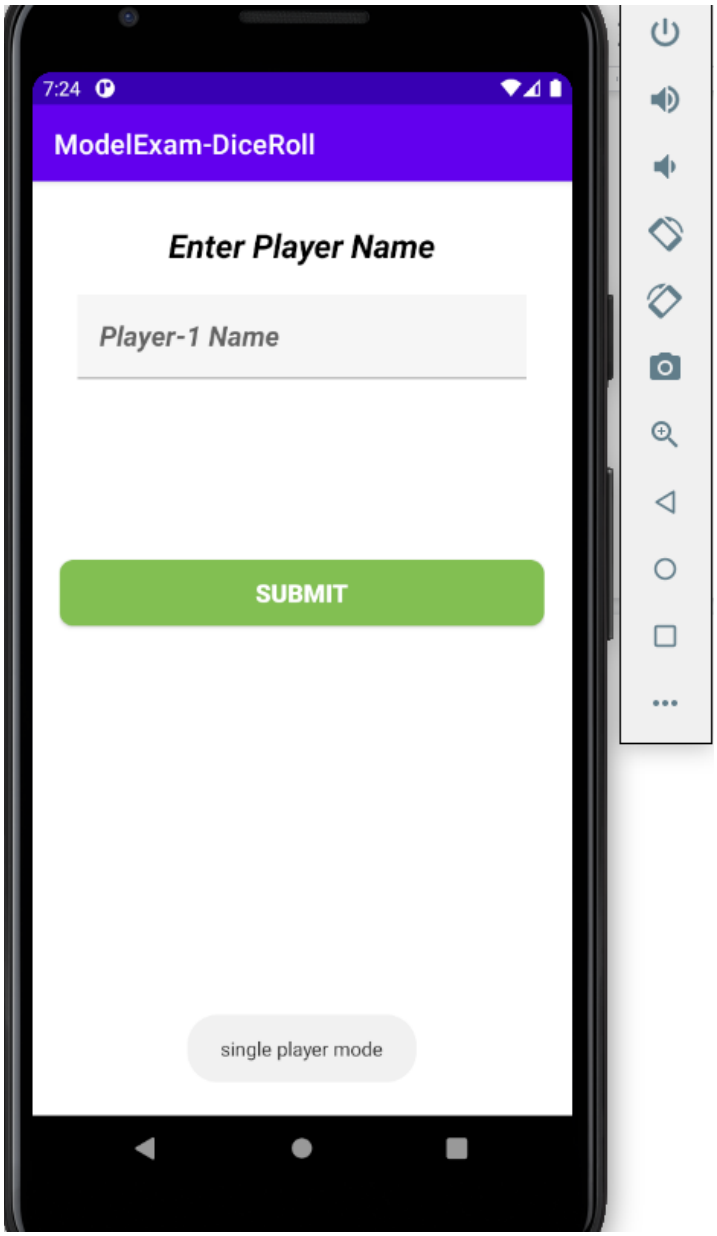
```
                } else if (turn == -1) {
                    p2score += num;
                    String p2text = pl2Name + " = " +
Integer.toString(p2score);
                    p2ScoreText.setText(p2text);
                    turn *= -1;


                    if (numOfTurns <= 0) {
//                          Toast.makeText(GameActivity.this,
pl2Name + " Won with score " + p2score,
Toast.LENGTH_SHORT).show();
                        turnsView.setText((pl1Name + " Won with
score " + p1score));
                        rollBtn.setText("Press Reset");
                        rollBtn.setEnabled(false);
                        rollBtn2.setText("Press Reset");
                        rollBtn2.setEnabled(false);
                    }
                }
            }

            @Override
            public void onAnimationRepeat(Animation animation)
{

            }
        });

        diePic2.startAnimation((rolling));


    }

    public void btnOnlick(View view) {
        diceRotate();
    }
```

```
    public void btnOnlick2(View view) {
        diceRotate2();
    }
}
```

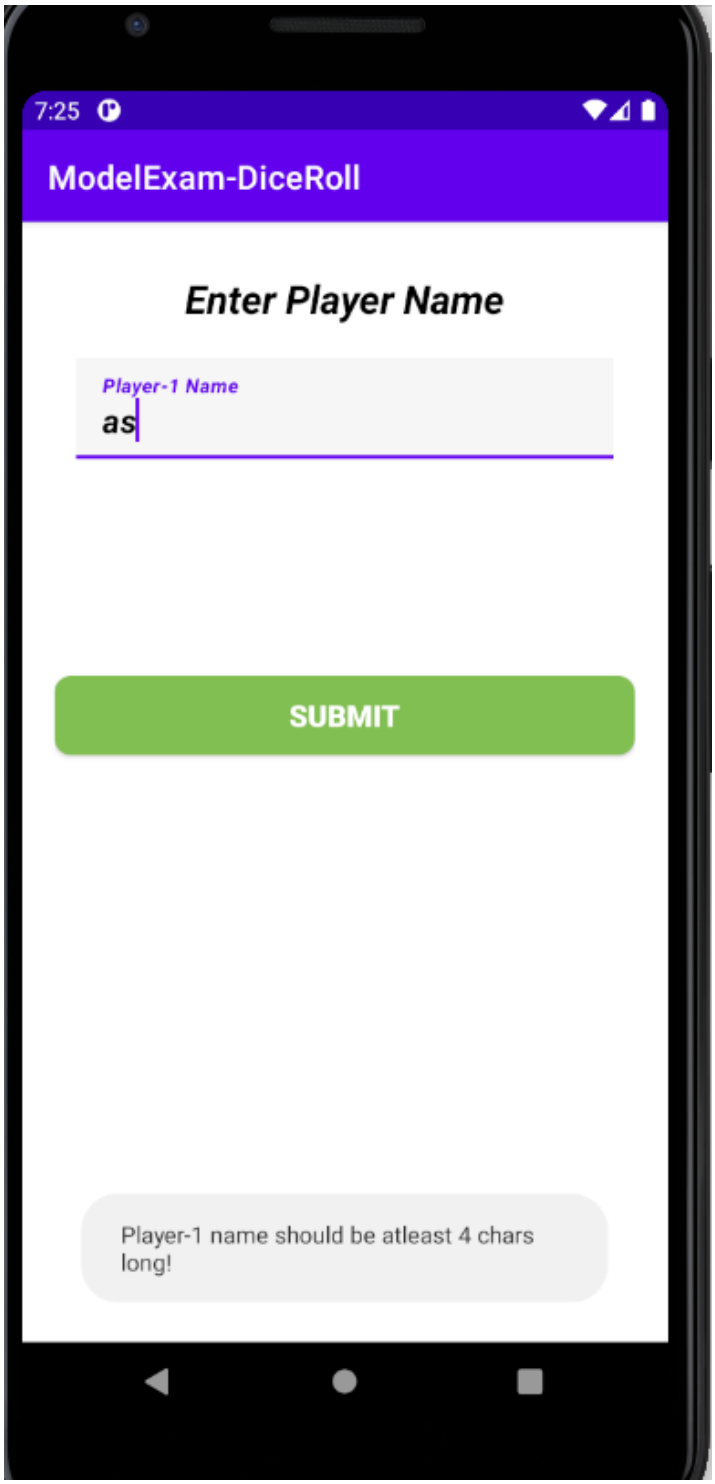
Screenshots: (To see animations in action please install the apk that i've attached in first page of this record)
Start:

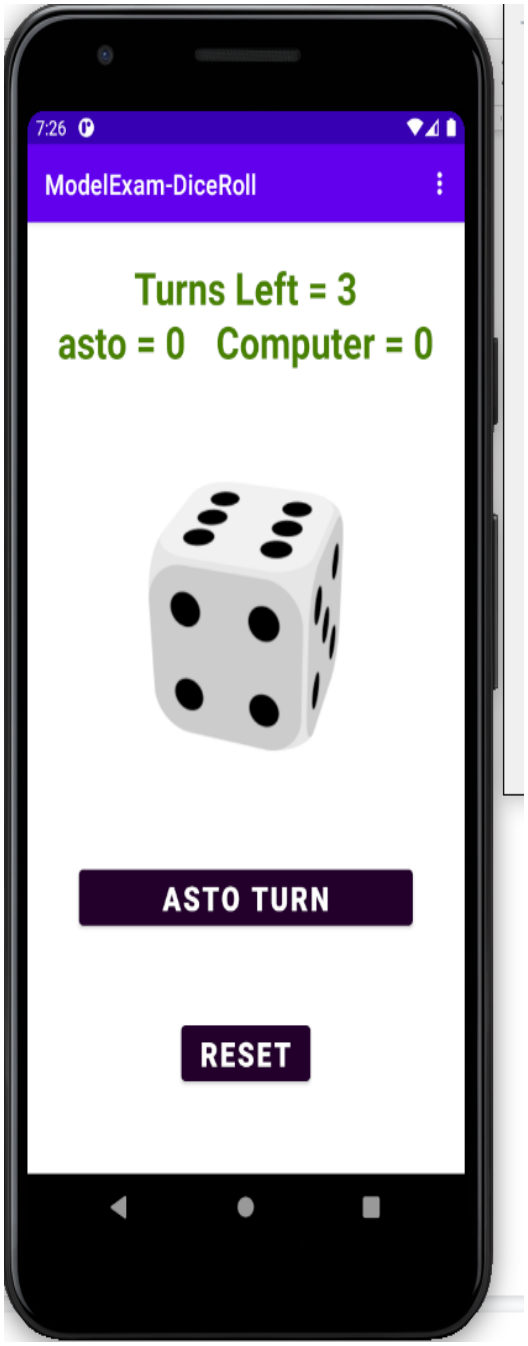


Single Player:

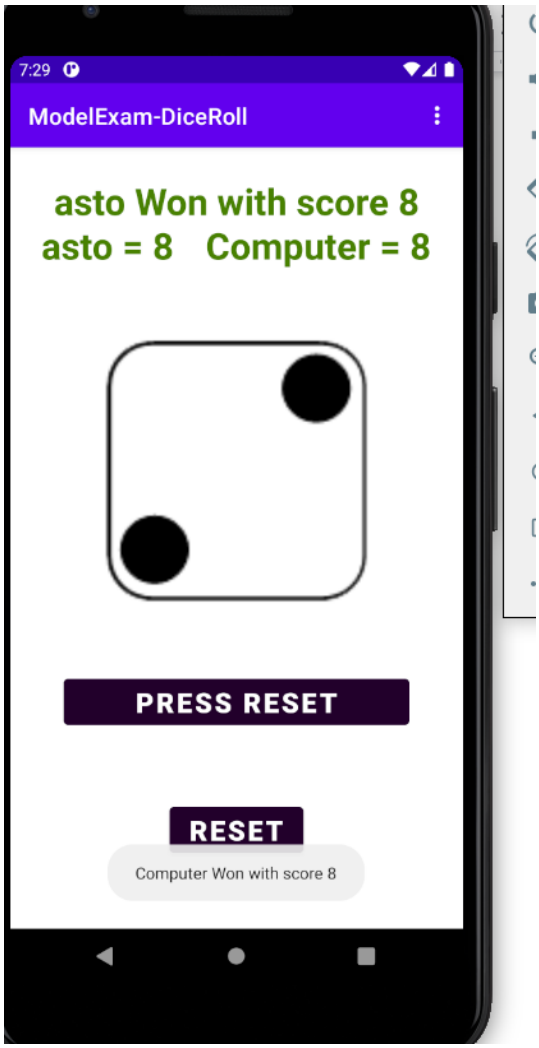


Having Validation of Inputs: Showing error to users using toast:

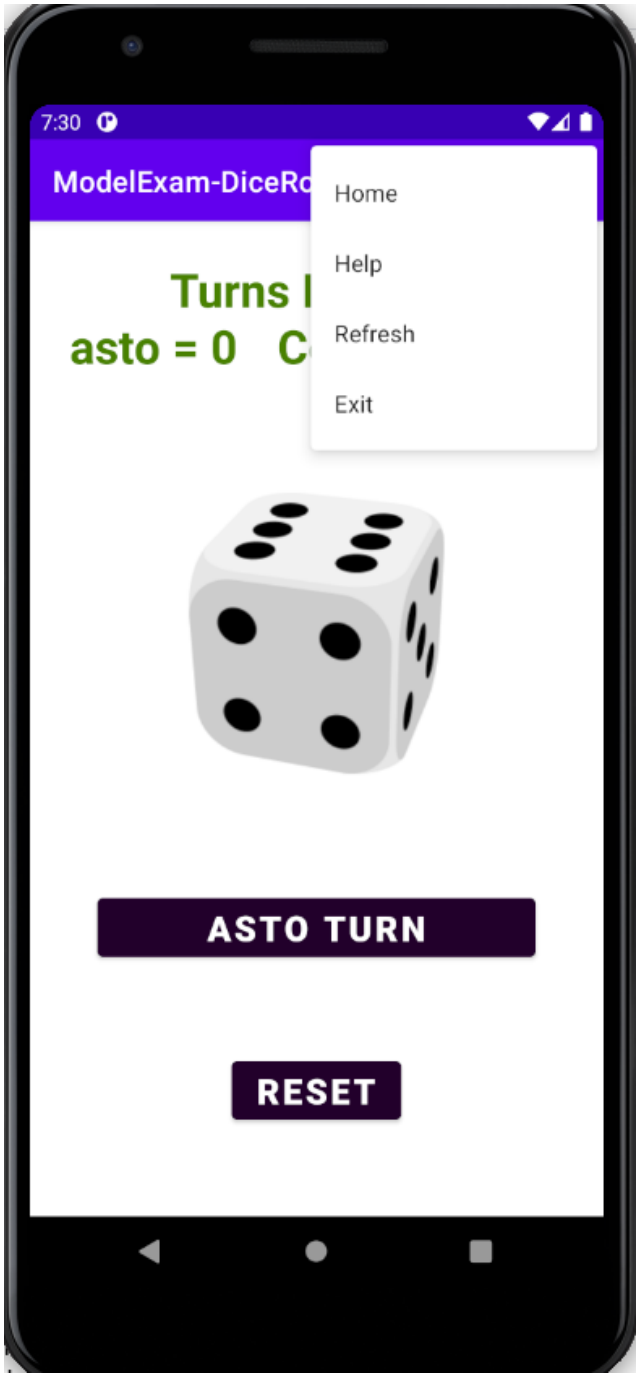



7:26
ModelExam-DiceRoll
Turns Left = 3
asto = 0 Computer = 0
ASTO TURN
RESET

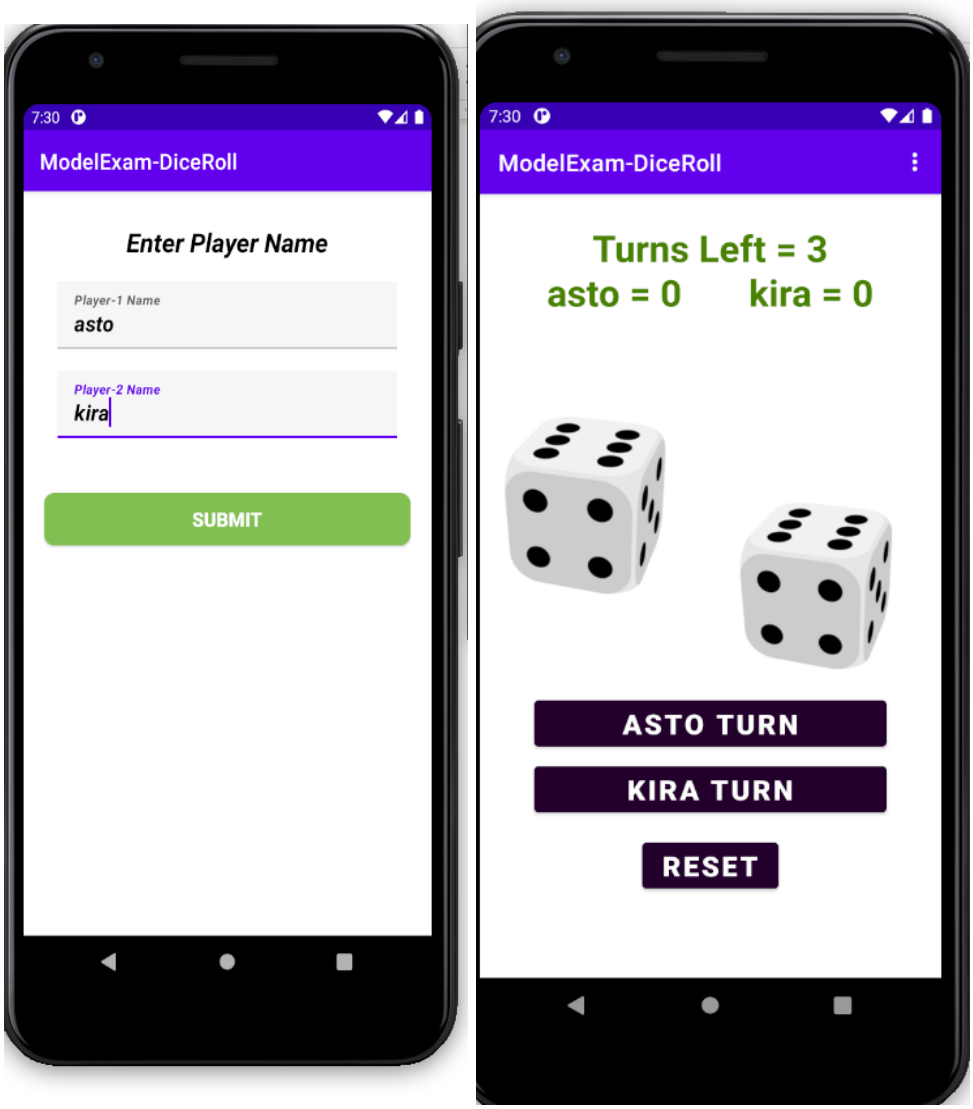
After I finished the game:



Menu Items:


7:30
ModelExam-DiceRo
Home
Help
Refresh
Exit
Turns l
asto = 0 C
ASTO TURN
RESET

Multiplayer:



Conclusion: The dice app is developed successfully and it helps users to play single player or multiplayer game modes. Appropriate checks and toasts ensure proper functioning of the app. Since there is a use of intent, the user is able to open other apps from this app itself and these features are working as expected. All the features were fully functional and were verified by mr kanna sir.