Experiment Name: Image Matching game with Notification
106118106 V Venkataraman CSE B

Date: 30-04-2021

Aim:
To create a mobile application game that a user can play by dragging and dropping image names on corresponding images. Press check button to get his score displayed as notification.

Description of App:
The application has a main activity xml file consisting of 6 images and corresponding names. The user can drag and drop the names onto the images to make the images invisible. After all images done, he can press check button to get score displayed as notification. Press the notification to get previous 3 scores of users. Also there is an action bar with menu items to refresh and exit the game.

Device Specifications:

App runs on the minSDK version of 16 and Android version 11 using Pixel 3A API emulator and samsung device. Only default libraries were used.

Technical Concepts Learnt:
- Image drag and drop
- Event listeners
- Pending intent
- Notification
- Action Bar

SOURCE CODE:
MainActivity.java:
```java
package com.example.nametheimage;

import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Build;
import android.os.Bundle;
import android.annotation.SuppressLint;
import android.annotation.TargetApi;
import android.content.ClipData;
import android.graphics.Typeface;
import android.util.Pair;
import android.view.DragEvent;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.MotionEvent;
import android.view.View;
import android.view.View.DragShadowBuilder;
import android.view.View.OnDragListener;
import android.view.View.OnTouchListener;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.NotificationCompat;
import androidx.core.app.NotificationManagerCompat;

import com.google.gson.Gson;
import com.google.gson.reflect.TypeToken;

import java.lang.reflect.Type;
import java.util.ArrayList;
import java.util.List;
import java.util.Queue;


public class MainActivity extends AppCompatActivity {

    private TextView textYellow,textGreen,textCyan,textPurple,textGrey,textPink;
    private ImageView yellow,green,cyan,purple,grey,pink;
    public CharSequence dragData;
    private int score=0;


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

        textCyan=findViewById(R.id.textCyan);
        textGreen=findViewById(R.id.textGreen);
        textGrey=findViewById(R.id.textGrey);
        textPink=findViewById(R.id.textPink);
        textPurple=findViewById(R.id.textPurple);
        textYellow=findViewById(R.id.textYellow);

        yellow=findViewById(R.id.yellow);
        cyan=findViewById(R.id.cyan);
        green=findViewById(R.id.green);
        pink=findViewById(R.id.pink);
        purple=findViewById(R.id.purple);
        grey=findViewById(R.id.grey);


        //set touch listeners
        textCyan.setOnTouchListener(new ChoiceTouchListener());
        textYellow.setOnTouchListener(new ChoiceTouchListener());
        textPurple.setOnTouchListener(new ChoiceTouchListener());
        textPink.setOnTouchListener(new ChoiceTouchListener());
        textGreen.setOnTouchListener(new ChoiceTouchListener());
        textGrey.setOnTouchListener(new ChoiceTouchListener());


        //set drag listeners
        yellow.setOnDragListener(new ChoiceDragListener());
        cyan.setOnDragListener(new ChoiceDragListener());
        green.setOnDragListener(new ChoiceDragListener());
        grey.setOnDragListener(new ChoiceDragListener());
        purple.setOnDragListener(new ChoiceDragListener());
        pink.setOnDragListener(new ChoiceDragListener());


        NotificationChannel channel= new NotificationChannel("My
Notification","My Notification",NotificationManager.IMPORTANCE_DEFAULT);
        NotificationManager manager=getSystemService(NotificationManager.class);
        manager.createNotificationChannel(channel);


    }



    private final class ChoiceTouchListener implements OnTouchListener {
        @SuppressLint("NewApi")
        @Override
        public boolean onTouch(View view, MotionEvent motionEvent) {
            if (motionEvent.getAction() == MotionEvent.ACTION_DOWN) {
                /*
                 * Drag details: we only need default behavior
                 * - clip data could be set to pass data as part of drag
                 * - shadow can be tailored
                 */
                ClipData data = ClipData.newPlainText("", "");
                DragShadowBuilder shadowBuilder = new
View.DragShadowBuilder(view);
                //start dragging the item touched
                view.startDrag(data, shadowBuilder, view, 0);
                return true;
            } else {
                return false;
            }
        }
    }

    @SuppressLint("NewApi")
    private class ChoiceDragListener implements OnDragListener {

        @Override
        public boolean onDrag(View v, DragEvent event) {
            switch (event.getAction()) {
                case DragEvent.ACTION_DRAG_STARTED:
                    //no action necessary
                    break;
                case DragEvent.ACTION_DRAG_ENTERED:
                    //no action necessary
                    break;
                case DragEvent.ACTION_DRAG_EXITED:
                    //no action necessary
                    break;
                case DragEvent.ACTION_DROP:

                    View view = (View) event.getLocalState();
                    TextView dropped = (TextView) view;

                    ImageView dropTarget=(ImageView) v;


                        view.setVisibility(View.INVISIBLE);


                        String imageName=String.valueOf(dropTarget.getTag());
                        if(imageName.equals(dropped.getText().toString())){
                            score++;
                        }

                        dropTarget.setVisibility(View.INVISIBLE);

                        dropTarget.setOnDragListener(null);


                    break;
                case DragEvent.ACTION_DRAG_ENDED:
                    //no action necessary
                    break;
                default:
                    break;
            }
            return true;
        }
    }
    public void reset(View view)
    {
       refresh();
    }
    public void displayScore(View view) {

        EditText name=findViewById(R.id.name);
        String username=name.getText().toString();
        if(username.equals("")){
            Toast.makeText(getApplicationContext(),"Type your
name",Toast.LENGTH_SHORT).show();
            return;
        }


        //Toast.makeText(getApplicationContext(),"Your score is: "+
score,Toast.LENGTH_SHORT).show();
        String message="";
        if(score==6)
            message="You Win. Your Score is "+ String.valueOf(score);
        else
            message="You Lose. Your Scores is "+ String.valueOf(score);

        NotificationCompat.Builder builder=new
NotificationCompat.Builder(MainActivity.this,"My
Notification").setSmallIcon(R.drawable.ic_launcher_foreground)
                .setContentTitle("Game
Status").setContentText(message).setAutoCancel(true);


        Intent intent = new Intent(MainActivity.this,Users.class);
        intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);


        PendingIntent pendingIntent =
PendingIntent.getActivity(MainActivity.this,0,intent,PendingIntent.FLAG_UPDATE_CU
RRENT);
        builder.setContentIntent(pendingIntent);

        NotificationManagerCompat notificationManagerCompat=
NotificationManagerCompat.from(MainActivity.this);
        notificationManagerCompat.notify(1,builder.build());

        ArrayList<Integer> Scores;
        ArrayList<String> Users;
        Scores= new ArrayList<Integer>();
        Users= new ArrayList<String>();

        Scores=loadScores();
        Users=loadUsers();
        Scores.add(score);
        Users.add(username);

        while(Scores.size()>3){
            Scores.remove(0);
            Users.remove(0);
        }
        saveSharedPrefs(Scores,Users,this);



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

Toast.makeText(getApplicationContext(),Users.get(i),Toast.LENGTH_SHORT).show();

        reset(view);
    }

    public static void saveSharedPrefs(ArrayList<Integer>
Scores,ArrayList<String> Users,Context context){

        SharedPreferences sharedPreferences= context.getSharedPreferences("My
Prefs",MODE_PRIVATE);
        SharedPreferences.Editor editor= sharedPreferences.edit();

        Gson gson = new Gson();
        String scores= gson.toJson(Scores);
        String users=gson.toJson(Users);
        editor.remove("scores").commit();
        editor.remove("users").commit();
        editor.putString("scores",scores);
        editor.putString("users",users);
        editor.commit();
    }

    public  ArrayList<Integer> loadScores(){

        SharedPreferences sharedPreferences=getSharedPreferences("My
Prefs",MODE_PRIVATE);
        Gson gson=new Gson();
        ArrayList<Integer> scores= new ArrayList<Integer>();
        String json= sharedPreferences.getString("scores","");
        if(json.isEmpty())
            return new ArrayList<Integer>();
        else{
            Type type = new TypeToken<List<Integer>>(){}.getType();
            scores=gson.fromJson(json,type);

            return scores;
        }
    }
    public ArrayList<String> loadUsers(){

        SharedPreferences sharedPreferences=getSharedPreferences("My
Prefs",MODE_PRIVATE);
        Gson gson=new Gson();
        ArrayList<String> user= new ArrayList<String>();
        String json= sharedPreferences.getString("users","");
        if(json.isEmpty())
            return new ArrayList<String>();
        else{
            Type type = new TypeToken<List<String>>(){}.getType();
            user=gson.fromJson(json,type);

            return user;
        }

    }

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

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {

        switch (item.getItemId()){
            case R.id.refresh: refresh();
                break;
            case R.id.Exit:

Toast.makeText(getApplicationContext(),"Exiting",Toast.LENGTH_SHORT).show();
                finish();
                break;
            default:
        }

        return super.onOptionsItemSelected(item);
    }

    public void refresh(){
        textPink.setVisibility(TextView.VISIBLE);
        textPurple.setVisibility(TextView.VISIBLE);
        textGreen.setVisibility(TextView.VISIBLE);
        textGrey.setVisibility(TextView.VISIBLE);
        textCyan.setVisibility(TextView.VISIBLE);
        textYellow.setVisibility(TextView.VISIBLE);
        score=0;

        pink.setVisibility(TextView.VISIBLE);
        purple.setVisibility(TextView.VISIBLE);
        green.setVisibility(TextView.VISIBLE);
        grey.setVisibility(TextView.VISIBLE);
        cyan.setVisibility(TextView.VISIBLE);
        yellow.setVisibility(TextView.VISIBLE);

        yellow.setOnDragListener(new ChoiceDragListener());
        cyan.setOnDragListener(new ChoiceDragListener());
        grey.setOnDragListener(new ChoiceDragListener());
        green.setOnDragListener(new ChoiceDragListener());
        purple.setOnDragListener(new ChoiceDragListener());
        pink.setOnDragListener(new ChoiceDragListener());


Toast.makeText(getApplicationContext(),"Refreshed",Toast.LENGTH_SHORT).show();
    }
}
```

Users.java:

```
package com.example.nametheimage;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.TextView;
import android.widget.Toast;

import com.google.gson.Gson;
import com.google.gson.reflect.TypeToken;

import java.lang.reflect.Type;
import java.util.ArrayList;
import java.util.List;

public class Users extends AppCompatActivity {
    private TextView previousUsers;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_users);

        previousUsers=findViewById(R.id.winners);

        ArrayList<Integer> Scores;
        ArrayList<String> Users;
        Scores= new ArrayList<Integer>();
        Users= new ArrayList<String>();

        Scores=loadScores();
        Users=loadUsers();
        String ans="Username \t Score\n";
        for(int i=0;i<Scores.size();i++){

            ans+=Users.get(i);
            ans+="        "+String.valueOf(Scores.get(i));
            ans+="\n";

        }
        previousUsers.setText(ans);

    }

    public ArrayList<Integer> loadScores(){

        SharedPreferences sharedPreferences=getSharedPreferences("My
Prefs",MODE_PRIVATE);
        Gson gson=new Gson();
        ArrayList<Integer> scores= new ArrayList<Integer>();
        String json= sharedPreferences.getString("scores","");
        if(json.isEmpty())
            return new ArrayList<Integer>();
        else{
            Type type = new TypeToken<List<Integer>>(){}.getType();
            scores=gson.fromJson(json,type);

            return scores;
        }

    }
    public ArrayList<String> loadUsers(){

        SharedPreferences sharedPreferences=getSharedPreferences("My
Prefs",MODE_PRIVATE);
        Gson gson=new Gson();
        ArrayList<String> user= new ArrayList<String>();
        String json= sharedPreferences.getString("users","");
        if(json.isEmpty())
            return new ArrayList<String>();
        else{
            Type type = new TypeToken<List<String>>(){}.getType();
            user=gson.fromJson(json,type);

            return user;
        }

    }

}
```

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"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    android:padding="10dp"
    android:paddingLeft="50dp"
    android:paddingRight="50dp">


    <TextView
        android:id="@+id/textCyan"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:gravity="center"
        android:text="Cyan"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/textGreen"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:gravity="center"
        android:text="Green"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/textPink"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:gravity="center"
        android:text="Pink"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/textYellow"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:gravity="center"
        android:text="Yellow"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/textPurple"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:gravity="center"
        android:text="Purple"
        android:textColor="@color/black" />

    <TextView
        android:id="@+id/textGrey"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:gravity="center"
        android:text="Grey"
        android:textColor="@color/black" />


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

        <ImageView
            android:id="@+id/cyan"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_row="0"
            android:layout_column="0"
            android:layout_marginLeft="10dp"
            android:layout_marginTop="4dp"
            android:layout_marginRight="4dp"
            android:layout_marginBottom="4dp"
            android:tag="Cyan"
            app:srcCompat="@drawable/cyan" />

        <ImageView
            android:id="@+id/green"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_row="0"
            android:layout_column="1"
            android:layout_marginLeft="4dp"
```

```xml
        android:layout_marginTop="4dp"
        android:layout_marginRight="4dp"
        android:layout_marginBottom="4dp"
        android:tag="Green"
        app:srcCompat="@drawable/green" />


    <ImageView
        android:id="@+id/pink"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:layout_row="0"
        android:layout_column="2"
        android:layout_marginLeft="4dp"
        android:layout_marginTop="4dp"
        android:layout_marginRight="4dp"
        android:layout_marginBottom="4dp"
        android:tag="Pink"
        app:srcCompat="@drawable/pink" />

    <ImageView
        android:id="@+id/yellow"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:layout_row="1"
        android:layout_column="0"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="4dp"
        android:layout_marginRight="4dp"
        android:layout_marginBottom="4dp"
        android:tag="Yellow"
        app:srcCompat="@drawable/yellow" />

    <ImageView
        android:id="@+id/purple"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:layout_row="1"
        android:layout_column="1"
        android:layout_marginLeft="4dp"
        android:layout_marginTop="4dp"
        android:layout_marginRight="4dp"
        android:layout_marginBottom="4dp"
        android:tag="Purple"
        app:srcCompat="@drawable/purple" />


    <ImageView
        android:id="@+id/grey"
        android:layout_width="100dp"
        android:layout_height="100dp"
```

```
            android:layout_row="1"
            android:layout_column="2"
            android:layout_marginLeft="4dp"
            android:layout_marginTop="4dp"
            android:layout_marginRight="4dp"
            android:layout_marginBottom="4dp"
            android:tag="Grey"
            app:srcCompat="@drawable/grey" />
    </GridLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <Button
            android:id="@+id/reset"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:backgroundTint="#F40B0B"
            android:onClick="reset"
            android:text="Reset" />

        <Button
            android:id="@+id/check"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="160dp"
            android:backgroundTint="#F40B0B"
            android:onClick="displayScore"
            android:text="Check" />

    </LinearLayout>

    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Name" />


</LinearLayout>
```

Activity_users.xml:

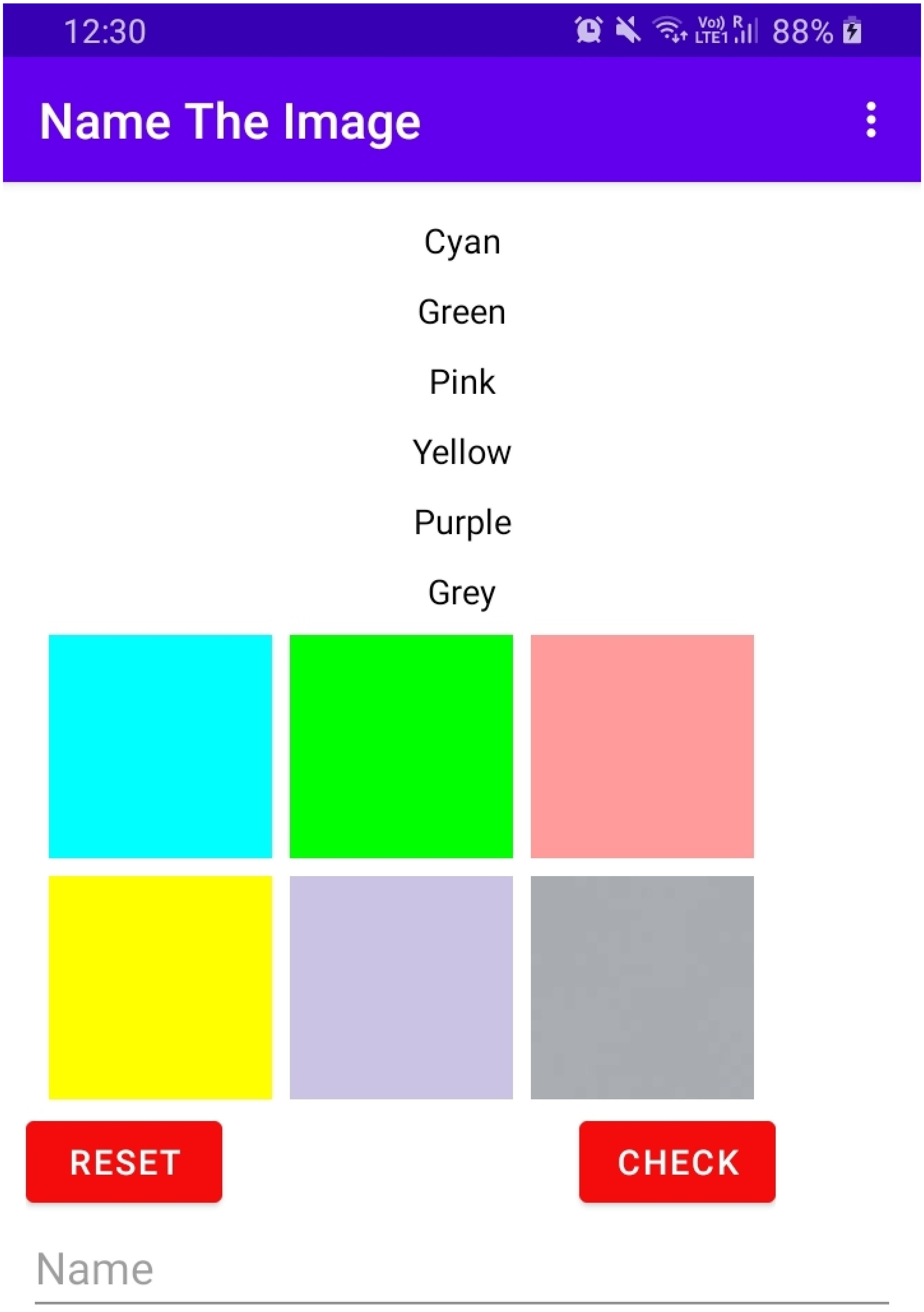
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#1E0202"
    tools:context=".Users">

    <TextView
        android:id="@+id/winners"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="100dp"
        android:gravity="center"
        android:text="TextView"
        android:textColor="@color/white"
        android:textSize="25dp"
        tools:layout_editor_absoluteX="147dp"
        tools:layout_editor_absoluteY="184dp" />
</LinearLayout>
```

Menu.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item
    android:id="@+id/refresh"
    android:title="Refresh"
    app:showAsAction="never"/>
    <item
        android:id="@+id/Exit"
        android:title="Exit"
        app:showAsAction="never"/>
</menu>
```

SCREENSHOTS:



Cyan

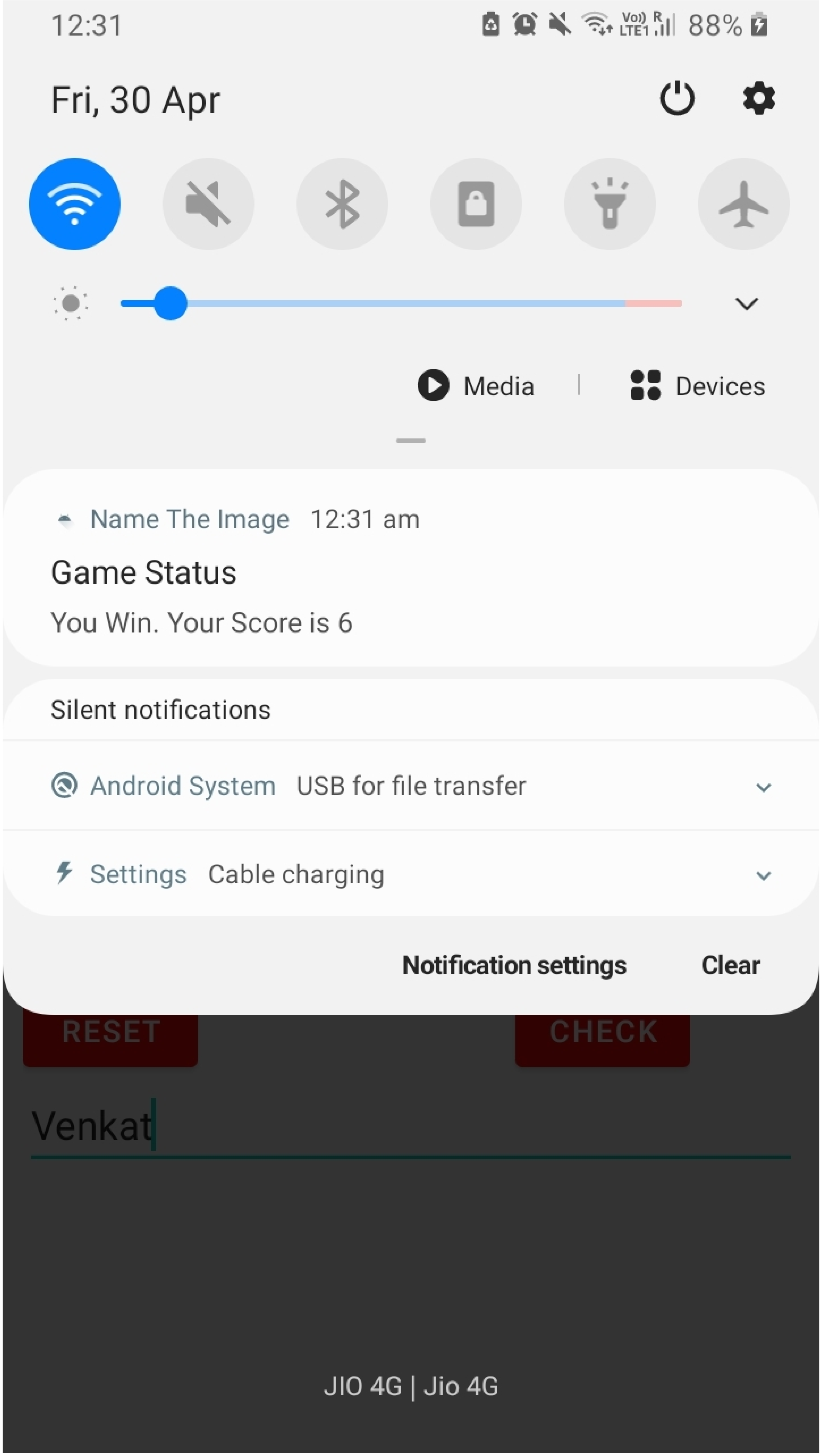
Green

Pink

Yellow

Purple

Grey



Name The Image 12:31 am

Game Status

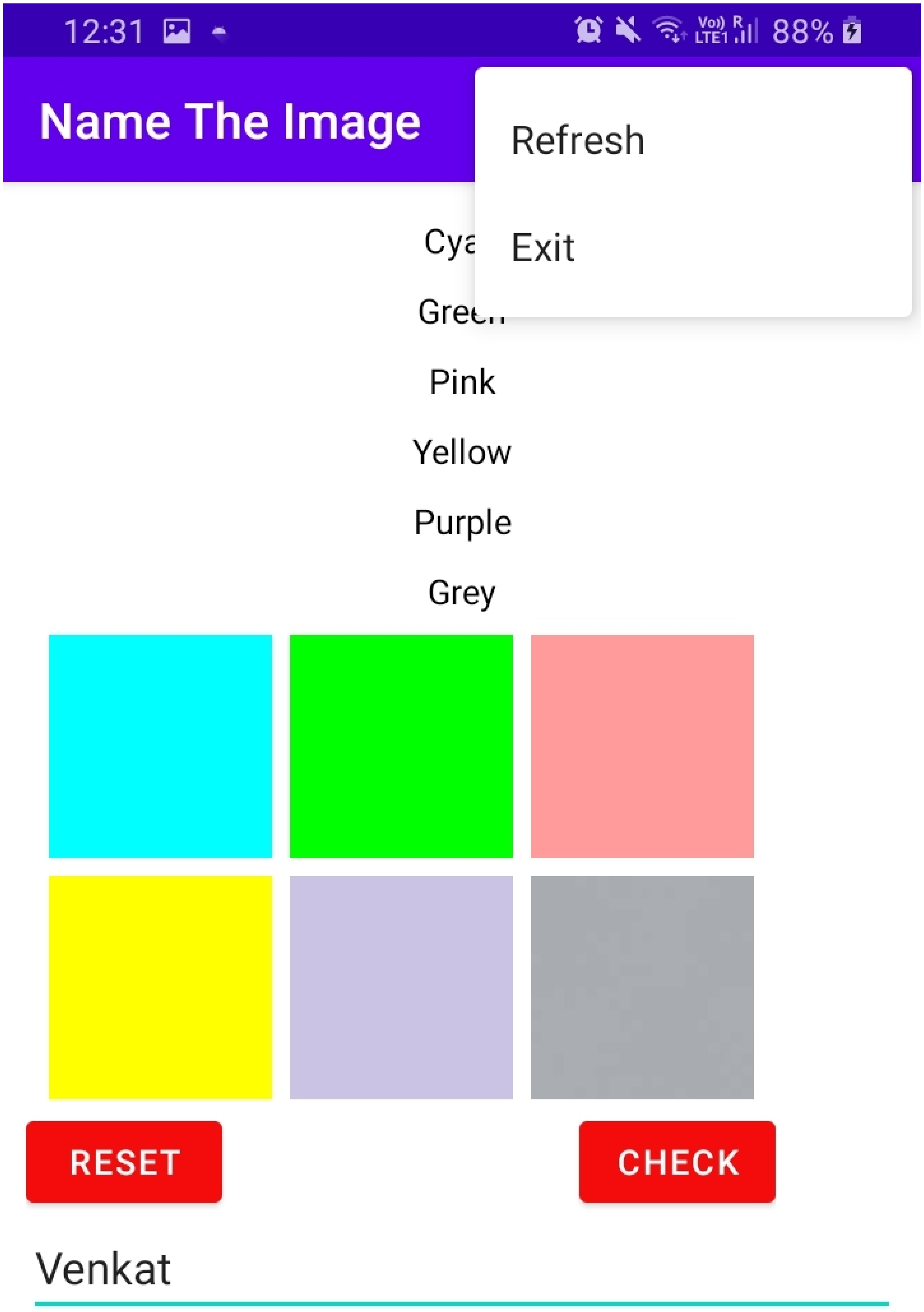
You Win. Your Score is 6

Silent notifications

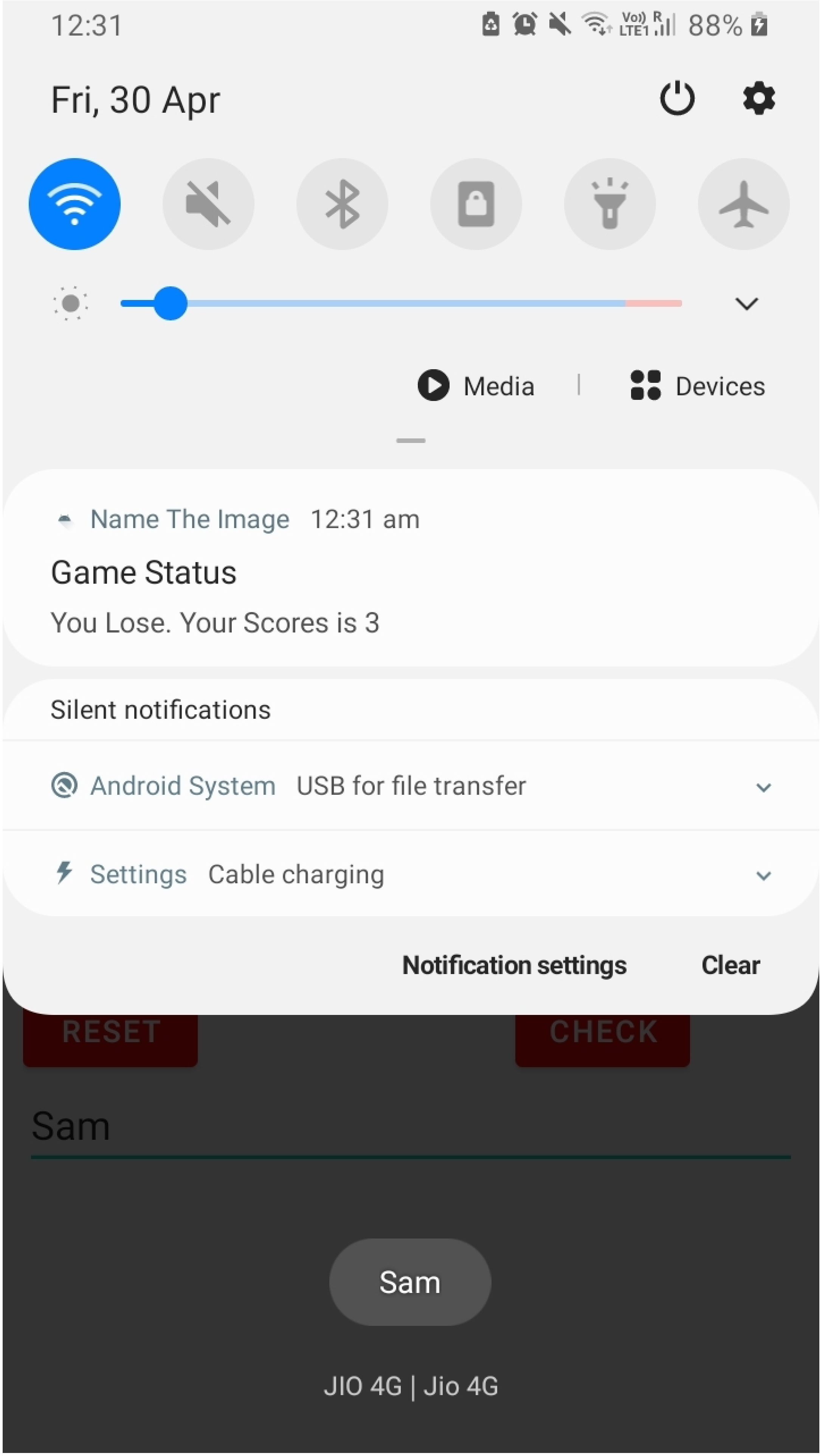
Android System USB for file transfer

Settings Cable charging

Venkat


12:31
88%
Name The Image
Refresh
Exit
Cya
Gree
Pink
Yellow
Purple
Grey
RESET
CHECK

Venkat



12:31 88%

Fri, 30 Apr

Media | Devices

Name The Image 12:31 am

Game Status

You Lose. Your Scores is 3

Silent notifications

Android System USB for file transfer

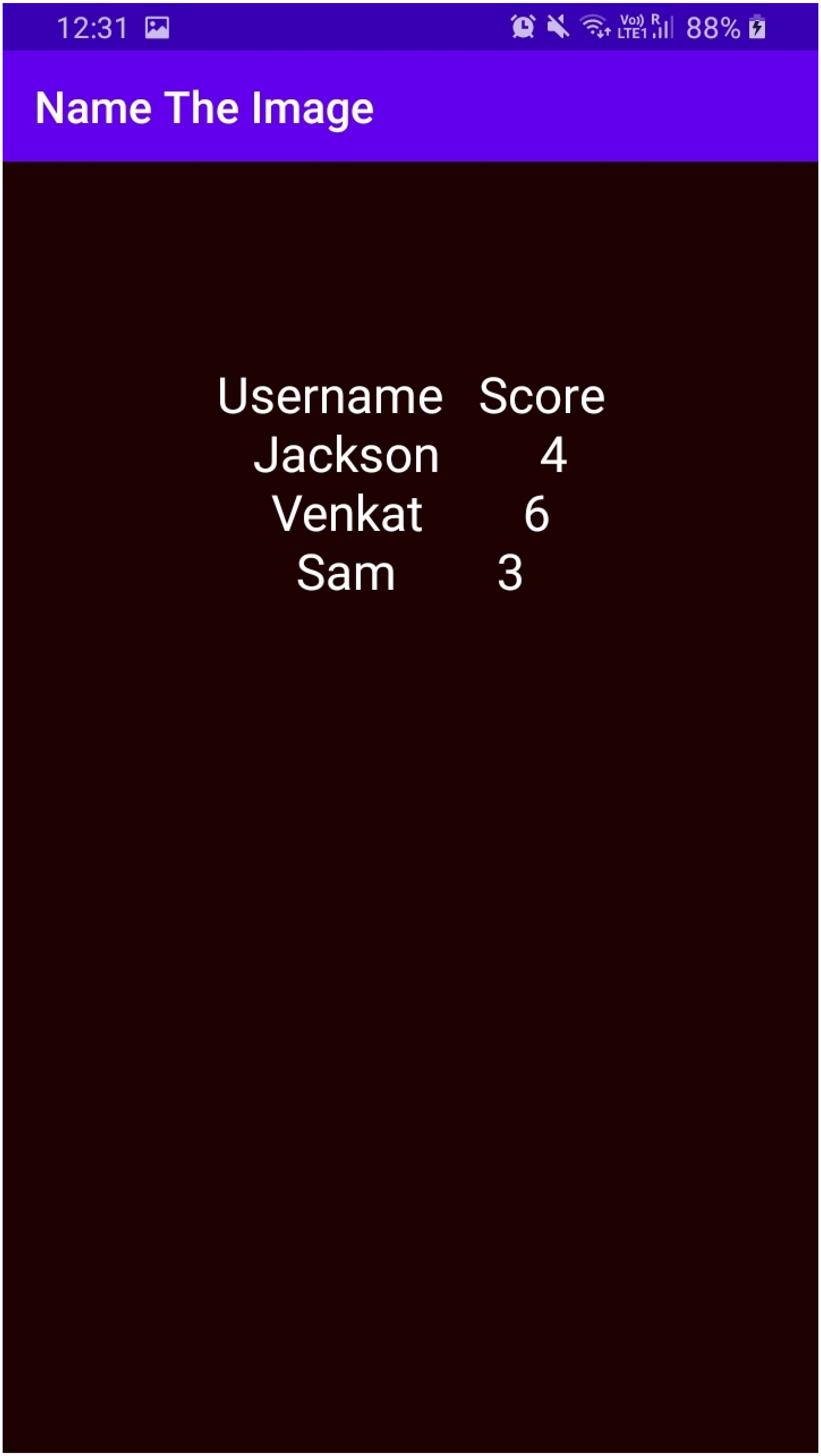
Settings Cable charging

Notification settings Clear

Sam

Sam

JIO 4G | Jio 4G



Username	Score
Jackson	4
Venkat	6
Sam	3

OUTCOME:

App was coded and run successfully without any errors and bugs.