# SENZORI

*Rješenje zadaće -*

<**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=".MainActivity"**>  
  
 <**TextView  
 android:id="@+id/scTry"  
 android:layout\_width="22dp"  
 android:layout\_height="48dp"  
 android:layout\_marginStart="208dp"  
 android:layout\_marginTop="16dp"  
 android:text="0"  
 android:textSize="34sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView3"** />  
  
 <**TextView  
 android:id="@+id/score2"  
 android:layout\_width="22dp"  
 android:layout\_height="48dp"  
 android:layout\_marginStart="176dp"  
 android:layout\_marginTop="16dp"  
 android:text="/"  
 android:textSize="34sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView3"** />  
  
 <**ImageView  
 android:id="@+id/imageView5"  
 android:layout\_width="123dp"  
 android:layout\_height="110dp"  
 android:layout\_marginStart="144dp"  
 android:onClick="increaseScore"  
 android:layout\_marginTop="132dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/timer"  
 app:srcCompat="@drawable/ball"** />  
  
 <**ImageView  
 android:id="@+id/imageView1"  
 android:layout\_width="123dp"  
 android:onClick="increaseScore"  
 android:layout\_height="110dp"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginTop="24dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/timer"  
 app:srcCompat="@drawable/ball"** />  
  
 <**ImageView  
 android:id="@+id/imageView2"  
 android:layout\_width="123dp"  
 android:layout\_height="110dp"  
 android:layout\_marginStart="144dp"  
 android:onClick="increaseScore"  
 android:layout\_marginTop="24dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/timer"  
 app:srcCompat="@drawable/ball"** />  
  
 <**ImageView  
 android:id="@+id/imageView3"  
 android:layout\_width="123dp"  
 android:layout\_height="110dp"  
 android:layout\_marginStart="272dp"  
 android:onClick="increaseScore"  
 android:layout\_marginTop="24dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/timer"  
 app:srcCompat="@drawable/ball"** />  
  
 <**ImageView  
 android:id="@+id/imageView4"  
 android:layout\_width="123dp"  
 android:layout\_height="110dp"  
 android:layout\_marginStart="16dp"  
 android:onClick="increaseScore"  
 android:layout\_marginTop="132dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/timer"  
 app:srcCompat="@drawable/ball"** />  
  
 <**ImageView  
 android:id="@+id/imageView7"  
 android:layout\_width="123dp"  
 android:layout\_height="110dp"  
 android:layout\_marginStart="20dp"  
 android:onClick="increaseScore"  
 android:layout\_marginTop="240dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/timer"  
 app:srcCompat="@drawable/ball"** />  
  
 <**ImageView  
 android:id="@+id/imageView8"  
 android:layout\_width="123dp"  
 android:layout\_height="110dp"  
 android:layout\_marginStart="148dp"  
 android:onClick="increaseScore"  
 android:layout\_marginTop="240dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/timer"  
 app:srcCompat="@drawable/ball"** />  
  
 <**ImageView  
 android:id="@+id/imageView9"  
 android:layout\_width="123dp"  
 android:layout\_height="110dp"  
 android:layout\_marginStart="272dp"  
 android:onClick="increaseScore"  
 android:layout\_marginTop="240dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/timer"  
 app:srcCompat="@drawable/ball"** />  
  
 <**ImageView  
 android:id="@+id/imageView6"  
 android:layout\_width="123dp"  
 android:layout\_height="110dp"  
 android:layout\_marginStart="272dp"  
 android:onClick="increaseScore"  
 android:layout\_marginTop="132dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/timer"  
 app:srcCompat="@drawable/ball"** />  
  
 <**TextView  
 android:id="@+id/timer"  
 android:layout\_width="141dp"  
 android:layout\_height="49dp"  
 android:layout\_marginStart="144dp"  
 android:onClick="increaseScore"  
 android:layout\_marginTop="80dp"  
 android:textSize="24sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
 <**TextView  
 android:id="@+id/textView"  
 android:layout\_width="94dp"  
 android:layout\_height="46dp"  
 android:layout\_marginStart="168dp"  
 android:layout\_marginTop="36dp"  
 android:text="TIMER"  
 android:textSize="24sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
 <**TextView  
 android:id="@+id/textView3"  
 android:layout\_width="112dp"  
 android:layout\_height="73dp"  
 android:layout\_marginStart="132dp"  
 android:layout\_marginTop="44dp"  
 android:gravity="center"  
 android:text="SCORE (HIT/TRY)"  
 android:textSize="24sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/imageView8"** />  
  
 <**TextView  
 android:id="@+id/score"  
 android:layout\_width="22dp"  
 android:layout\_height="48dp"  
 android:layout\_marginStart="148dp"  
 android:layout\_marginTop="16dp"  
 android:text="0"  
 android:textSize="34sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView3"** />  
  
</**androidx.constraintlayout.widget.ConstraintLayout**>

**package** com.example.myapplication16;  
**import** android.os.Bundle;  
**import** androidx.appcompat.app.AppCompatActivity;  
**import** androidx.appcompat.app.AlertDialog;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.os.CountDownTimer;  
**import** android.os.Handler;  
**import** android.os.Looper;  
**import** android.util.AttributeSet;  
**import** android.view.View;  
**import** android.view.Window;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
**import** java.util.Random;  
  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 TextView **timer**, **scoring**, **scTry**;  
 ImageView **imageView1**,**imageView2**,**imageView3**,**imageView4**,**imageView5**,**imageView6**,**imageView7**,**imageView8**,**imageView9**;  
 Handler **handler**;  
 Runnable **runnable**;  
  
 ImageView[] **imageList**;  
 **int score**;  
 **int s**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 requestWindowFeature(Window.***FEATURE\_NO\_TITLE***);  
 getSupportActionBar().hide();  
 setContentView(R.layout.***activity\_main***);  
  
 **timer** = findViewById(R.id.***timer***);  
 **scoring** = findViewById(R.id.***score***);  
 **scTry** = findViewById(R.id.***scTry***);  
  
 **imageView1** = findViewById(R.id.***imageView1***);  
 **imageView2** = findViewById(R.id.***imageView2***);  
 **imageView3** = findViewById(R.id.***imageView3***);  
 **imageView4** = findViewById(R.id.***imageView4***);  
 **imageView5** = findViewById(R.id.***imageView5***);  
 **imageView6** = findViewById(R.id.***imageView6***);  
 **imageView7** = findViewById(R.id.***imageView7***);  
 **imageView8** = findViewById(R.id.***imageView8***);  
 **imageView9** = findViewById(R.id.***imageView9***);  
  
 **imageList**=**new** ImageView[]{**imageView1**,**imageView2**,**imageView3**,**imageView4**,**imageView5**,**imageView6**,**imageView7**,**imageView8**,**imageView9**};  
 makeitgone();  
  
  
  
 *// setting timer to play game* **new** CountDownTimer(10000, 1000)  
 {  
  
 *// increasing time* @Override  
 **public void** onTick(**long** l) {  
 **timer**.setText(**"Time : "**+l/1000);  
 }  
  
 *// When time is finished* @Override  
 **public void** onFinish() {  
 **timer**.setText(**"Time Over"**);  
 **handler**.removeCallbacks(**runnable**);  
  
 *// using for loop* **for** (ImageView image:**imageList**)  
 {  
 image.setVisibility(View.***INVISIBLE***);  
 }  
  
 *// dialog box to ask user's input* AlertDialog.Builder alert=**new** AlertDialog.Builder(MainActivity.**this**);  
 alert.setTitle(**"Try Again!"**);  
 alert.setMessage(**"Do you want to restart?"**);  
  
 *// if user want to restart game* alert.setPositiveButton(**"Yes"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialogInterface, **int** i) {  
 Intent intent=getIntent();  
 finish();  
 startActivity(intent);  
 }  
 });  
  
 *// When user not want to play again* alert.setNegativeButton(**"No"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialogInterface, **int** i) {  
 Toast.*makeText*(MainActivity.**this**, **"Game Over!!!"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 });  
 alert.show();  
 }  
 }.start();  
 }  
  
 **private void** makeitgone() {  
 **handler**=**new** Handler();  
 **runnable**=**new** Runnable() {  
 @Override  
 **public void** run() {  
 brPok();  
 **for**(ImageView image:**imageList**)  
 {  
 image.setImageResource(R.drawable.***ball***);  
 **final** Handler handler=**new** Handler(Looper.*getMainLooper*());  
 handler.postDelayed(**new** Runnable() {  
 @Override  
 **public void** run() {  
 image.setImageResource(R.drawable.***ball***);  
 }  
 },900);  
 image.setVisibility(View.***INVISIBLE***);  
 }  
  
 *// making image visible at random positions* Random random=**new** Random();  
 **int** i=random.nextInt(9);  
 **imageList**[i].setVisibility(View.***VISIBLE***);  
 **handler**.postDelayed(**this**,1000);  
 **s**++;  
  
 }  
 };  
 **handler**.post(**runnable**);  
 }  
  
  
 *// increasing score* **public void** increaseScore(View view) {  
 **score**=**score**+1;  
 **scoring**.setText(**""**+**score**);  
 }  
 **public void** brPok()  
 {  
 **scTry**.setText(**""**+**s**);  
 }  
  
}

<https://www.mobileprocessing.org/>

<https://developer.android.com/guide/topics/sensors/sensors_overview>

# Korištenje akcelerometra na Androidu

1. Kreiranje novog projekta

IDE koji podržava razvoj Androida, stvorit će Main klasu.  IDE kreira datoteku glavnog izgleda koju Main klasa koristi za stvaranje korisničkog sučelja aplikacije.

Budući da ćemo koristiti gestu potresanja, dobro je zaključati orijentaciju uređaja. To će osigurati da se korisničko sučelje aplikacije ne prebacuje stalno između portretnog i pejzažnog. Otvorite datoteku **manifesta** projekta i postavite **screenOrientation** opciju na **portrait**.

<**activity android:name="MainActivity"  
android:screenOrientation="portrait"  
android:label="@string/app\_name"**>  
<**intent-filter**>  
 <**action android:name="android.intent.action.MAIN"** />  
 <**category android:name="android.intent.category.LAUNCHER"** />  
</**intent-filter**>  
</**activity**>

1. Postavljanje senzora

**public class** MainActivity **extends** AppCompatActivity {

@Override  
 **public void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.main);  
 }  
}

Uvoz klasa

**import** android.os.Bundle;  
**import** androidx.appcompat.app.AppCompatActivity;

SensorEventListener sučelje

Za korištenje SensorEventListener sučelja, Main klasa aktivnosti treba ga implementirati.

**import** android.hardware.SensorEventListener; //uvoz klase

**public class** MainActivity **extends** AppCompatActivity **implements** SensorEventListener {  
  
 */\*\* Called when the activity is first created. \*/* @Override  
 **public void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.main);  
 }  
}

Dodati dvije potrebne metode u Main klasi i izvan onCreate metode.

@Override  
**public void** onSensorChanged(SensorEvent event) {  
  
}  
  
@Override  
**public void** onAccuracyChanged(Sensor sensor, **int** accuracy) {  
  
}

Koristit ćemo metodu za otkrivanje geste potresanja. Metoda onSensorChanged se poziva svaki put kada ugrađeni senzor otkrije promjenu.  Da bismo koristili Sensori SensorEvent klase, dodajemo dva uvoza.

}  
**import** android.hardware.Sensor;  
**import** android.hardware.SensorEvent;

Prije implementacije onSensorChanged, moramo deklarirati dvije privatne varijable u Main klasi,

senSensorManager tipa SensorManager i senAccelerometer tipa Sensor.

**private** SensorManager **senSensorManager**;  
**private** Sensor **senAccelerometer**;

**import** android.hardware.SensorManager; //uvoz klase

metoda onCreate

@Override  
**protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
  
 senSensorManager = (SensorManager) getSystemService(Context.SENSOR\_SERVICE);  
 senAccelerometer = senSensorManager.getDefaultSensor(Sensor.TYPE\_ACCELEROMETER);  
 senSensorManager.registerListener(**this**, senAccelerometer , SensorManager.SENSOR\_DELAY\_NORMAL);  
}

Kako bi inicijalizirali SensorManager instancu, pozivamo  getSystemService da dohvatimo SensorManager instancu sustava, koju zauzvrat koristimo za pristup senzorima sustava. Metoda getSystemServicese koristi za dobivanje reference na uslugu sustava prosljeđivanjem naziva usluge. S upraviteljem senzora koji nam je na raspolaganju, dobivamo referencu na akcelerometar sustava pozivanjem getDefaultSensorna upravitelja senzora i prosljeđivanjem tipa senzora koji nas zanima. Zatim registriramo senzor pomoću jedne od SensorManager javnih metoda, registerListener. Ova metoda prihvaća tri argumenta, kontekst aktivnosti, senzor i brzinu kojom nam se dostavljaju senzorski događaji.

**private** SensorManager **senSensorManager**;  
 **private** Sensor **senAccelerometer**;  
  
 **public class** Main **extends** Activity **implements** SensorEventListener {  
@Override

**public void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.main);  
 **senSensorManager** = (SensorManager) getSystemService(Context.SENSOR\_SERVICE);  
 **senAccelerometer** = **senSensorManager**.getDefaultSensor(Sensor.TYPE\_ACCELEROMETER);  
 **senSensorManager**.registerListener(**this**, **senAccelerometer** , SensorManager.SENSOR\_DELAY\_NORMAL);  
 }  
  
 @Override  
 **public void** onSensorChanged(SensorEvent sensorEvent) {  
  
 }  
  
 @Override  
 **public void** onAccuracyChanged(Sensor sensor, **int** accuracy) {  
  
 }  
}

Postoje još dvije metode koje moramo nadjačati,  onPause i onResume.

Ovo su metode Main klase. Dobra je praksa poništiti registraciju senzora kada aplikacija u sleep modu i ponovno registrirati senzor kada se aplikacija nastavi.

**protected void** onPause() {  
 **super**.onPause();  
 senSensorManager.unregisterListener(**this**);  
}

**protected void** onResume() {  
 **super**.onResume();  
 senSensorManager.registerListener(**this**, senAccelerometer, SensorManager.SENSOR\_DELAY\_NORMAL);  
}

kada se pokreće uređaj

**private long lastUpdate** = 0;  
**private float last\_x**, **last\_y**, **last\_z**;  
**private static final int *SHAKE\_THRESHOLD*** = 600;

**public void** onSensorChange(SensorEvent sensorEvent) {  
 Sensor mySensor = sensorEvent.sensor;  
  
 **if** (mySensor.getType() == Sensor.TYPE\_ACCELEROMETER) {  
  
 }  
}

**public void** onSensorChange(SensorEvent sensorEvent) {  
 Sensor mySensor = sensorEvent.sensor;  
  
 **if** (mySensor.getType() == Sensor.TYPE\_ACCELEROMETER) {  
 **float** x = sensorEvent.values[0];  
 **float** y = sensorEvent.values[1];  
 **float** z = sensorEvent.values[2];  
  
 **long** curTime = System.*currentTimeMillis*();  
  
 **if** ((curTime - lastUpdate) > 100) {  
 **long** diffTime = (curTime - lastUpdate);  
 lastUpdate = curTime;  
  
 **float** speed = Math.abs(x + y + z - last\_x - last\_y - last\_z)/ diffTime \* 10000;  
  
 **if** (speed > SHAKE\_THRESHOLD) {  
  
 }  
  
 last\_x = x;  
 last\_y = y;  
 last\_z = z;  
 }  
 }  
}

Aplikacija - protresi me

**package** com.example.myapplication16;  
  
**import** android.app.Activity;  
**import** android.graphics.Color;  
**import** android.hardware.Sensor;  
**import** android.hardware.SensorEvent;  
**import** android.hardware.SensorEventListener;  
**import** android.hardware.SensorManager;  
**import** android.os.Bundle;  
  
**import** android.widget.Toast;  
**import** android.view.View;  
**public class** MainActivity **extends** Activity **implements** SensorEventListener {  
 **private** SensorManager **sensorManager**;  
 **private boolean isColor** = **false**;  
 **private** View **view**;  
 **private long lastUpdate**;  
  
 @Override  
 **public void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 **view** = findViewById(R.id.***txtvw***);  
 **view**.setBackgroundColor(Color.***GREEN***);  
  
 **sensorManager** = (SensorManager) getSystemService(***SENSOR\_SERVICE***);  
 **lastUpdate** = System.*currentTimeMillis*();  
 }  
  
 @Override  
 **public void** onAccuracyChanged(Sensor sensor, **int** accuracy) {  
 }  
  
 @Override  
 **public void** onSensorChanged(SensorEvent event) {  
 **if** (event.**sensor**.getType() == Sensor.***TYPE\_ACCELEROMETER***) {  
 getAccelerometer(event);  
 }  
  
 }  
  
 **private void** getAccelerometer(SensorEvent event) {  
 **float**[] values = event.**values**;  
 *// Movement* **float** x = values[0];  
 **float** y = values[1];  
 **float** z = values[2];  
  
 **float** aSquareRoot = (x \* x + y \* y + z \* z)  
 / (SensorManager.***GRAVITY\_EARTH*** \* SensorManager.***GRAVITY\_EARTH***);  
  
 **long** actualTime = System.*currentTimeMillis*();  
 Toast.*makeText*(getApplicationContext(), String.*valueOf*(aSquareRoot) + **" "** +  
 SensorManager.***GRAVITY\_EARTH***, Toast.***LENGTH\_SHORT***).show();  
  
 **if** (aSquareRoot >= 2) *//it will be executed if you shuffle* {  
  
 **if** (actualTime - **lastUpdate** < 200) {  
 **return**;  
 }  
 **lastUpdate** = actualTime;*//updating lastUpdate for next shuffle* **if** (**isColor**) {  
 **view**.setBackgroundColor(Color.***GREEN***);  
  
 } **else** {  
 **view**.setBackgroundColor(Color.***RED***);  
 }  
 **isColor** = !**isColor**;  
 }  
 }  
  
 @Override  
 **protected void** onResume() {  
 **super**.onResume();  
  
 **sensorManager**.registerListener(**this**, **sensorManager**.getDefaultSensor(Sensor.***TYPE\_ACCELEROMETER***),  
 SensorManager.***SENSOR\_DELAY\_NORMAL***);  
 }  
  
 @Override  
 **protected void** onPause() {  
  
 **super**.onPause();  
 **sensorManager**.unregisterListener(**this**);  
 }  
}

*<?***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:screenOrientation="portrait"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"**>  
  
 <**TextView  
 android:id="@+id/txtvw"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text=""  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
</**androidx.constraintlayout.widget.ConstraintLayout**>

Aplikacija – kompas

**package** com.example.myapplication16;  
  
**import** androidx.appcompat.app.AppCompatActivity;  
  
**import** android.content.Context;  
**import** android.hardware.Sensor;  
**import** android.hardware.SensorEvent;  
**import** android.hardware.SensorEventListener;  
**import** android.hardware.SensorManager;  
**import** android.os.Bundle;  
**import** android.widget.TextView;  
  
**public class** MainActivity **extends** AppCompatActivity {  
 **private** SensorManager **sensorManager**;  
 **private** TextView **txtvw**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 **txtvw** = findViewById(R.id.***txtvw***);  
 **sensorManager** = (SensorManager) getSystemService(Context.***SENSOR\_SERVICE***);  
 Sensor sensor = **sensorManager**.getDefaultSensor(Sensor.***TYPE\_MAGNETIC\_FIELD***);  
 **sensorManager**.registerListener(**listener**, sensor, SensorManager.***SENSOR\_DELAY\_NORMAL***);  
 }  
 @Override  
 **protected void** onDestroy() {  
 **super**.onDestroy();  
 **if** (**sensorManager** != **null**) {  
 **sensorManager**.unregisterListener(**listener**);  
 }  
 }  
 **private** SensorEventListener **listener** = **new** SensorEventListener() {  
 @Override  
 **public void** onSensorChanged(SensorEvent event) {  
 **float** xValue = event.**values**[0];  
 **float** yValue = event.**values**[1];  
 **float** zValue = event.**values**[2];  
 **float** rot = (**float**) (180/Math.***PI***\*Math.*copySign*(Math.*acos*(xValue/Math.*sqrt*(Math.*pow*(xValue, 2)+Math.*pow*(yValue, 2))), yValue));  
 *// float rot = (float) (180/Math.PI\*Math.copySign(Math.atan(yValue/xValue), yValue));* **float** sca = (**float**) (Math.*sqrt*(Math.*pow*(xValue, 2)+Math.*pow*(yValue, 2)));  
 *// txtvw.setText("x:"+xValue+"\ny:"+yValue+"\nz:"+zValue+"\nd:"+rot);* **if** (Math.*abs*(yValue)>90) {  
 **txtvw**.setText(**"→"**);  
 **txtvw**.setRotation(-(rot));  
 } **else** {  
 **txtvw**.setText(**"←"**);  
 **txtvw**.setRotation(Math.*copySign*(180-Math.*abs*(rot), rot));  
 }  
 **txtvw**.setScaleX(sca);  
 **txtvw**.setScaleY(sca);  
 }  
 @Override  
 **public void** onAccuracyChanged(Sensor sensor, **int** accuracy) {  
 }  
 };  
}

*<?***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:screenOrientation="portrait"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"**>  
  
 <**TextView  
 android:id="@+id/txtvw"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text=""  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
</**androidx.constraintlayout.widget.ConstraintLayout**>

Zadatak :

Kreirati aplikaciju koja kod svakog pokreta uređaja mijenja sliku kockice (brojevi od 1 do 6).