**Шагомер**

**activityMain:**

<?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=".MainActivity">  
  
 <LinearLayout  
 android:id="@+id/linearLayout"  
 android:layout\_width="270dp"  
 android:layout\_height="270dp"  
 android:layout\_marginStart="50dp"  
 android:layout\_marginTop="100dp"  
 android:layout\_marginEnd="50dp"  
 android:background="@drawable/radiuss"  
 android:orientation="vertical"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.487"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.132">  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="75dp"  
 android:gravity="center"  
 android:text="50"  
 android:textColor="#00B7FF"  
 android:textSize="64sp" />  
  
 <TextView  
 android:id="@+id/textView3"  
 android:layout\_width="match\_parent"  
 android:layout\_height="45dp"  
 android:layout\_weight="1"  
 android:gravity="center"  
 android:text="ШАГИ"  
 android:textColor="#000000" />  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="270dp"  
 android:layout\_height="60dp"  
 android:alayout\_width="match\_parent"  
 android:background="@drawable/radius\_button"  
 android:text="ПАУЗА"  
 android:textColor="#00B7FF"  
 app:toggleCheckedStateOnClick="false" />  
  
 </LinearLayout>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**Radiuss:**

<?xml version="1.0" encoding="utf-8"?>  
<shape xmlns:android="http://schemas.android.com/apk/res/android">  
 android:shape="rectangle">  
 <solid android:color= "#FFFFFF"/>  
  
 <stroke android:widht="0dp"  
 android:color="#ff000000"/>  
  
 <padding android:left ="2dp"  
 android:top= "2dp"  
 android:right= "2dp"  
 android:bottom="2dp" />  
  
 <corners android:radius= "135dp"/>  
</shape>

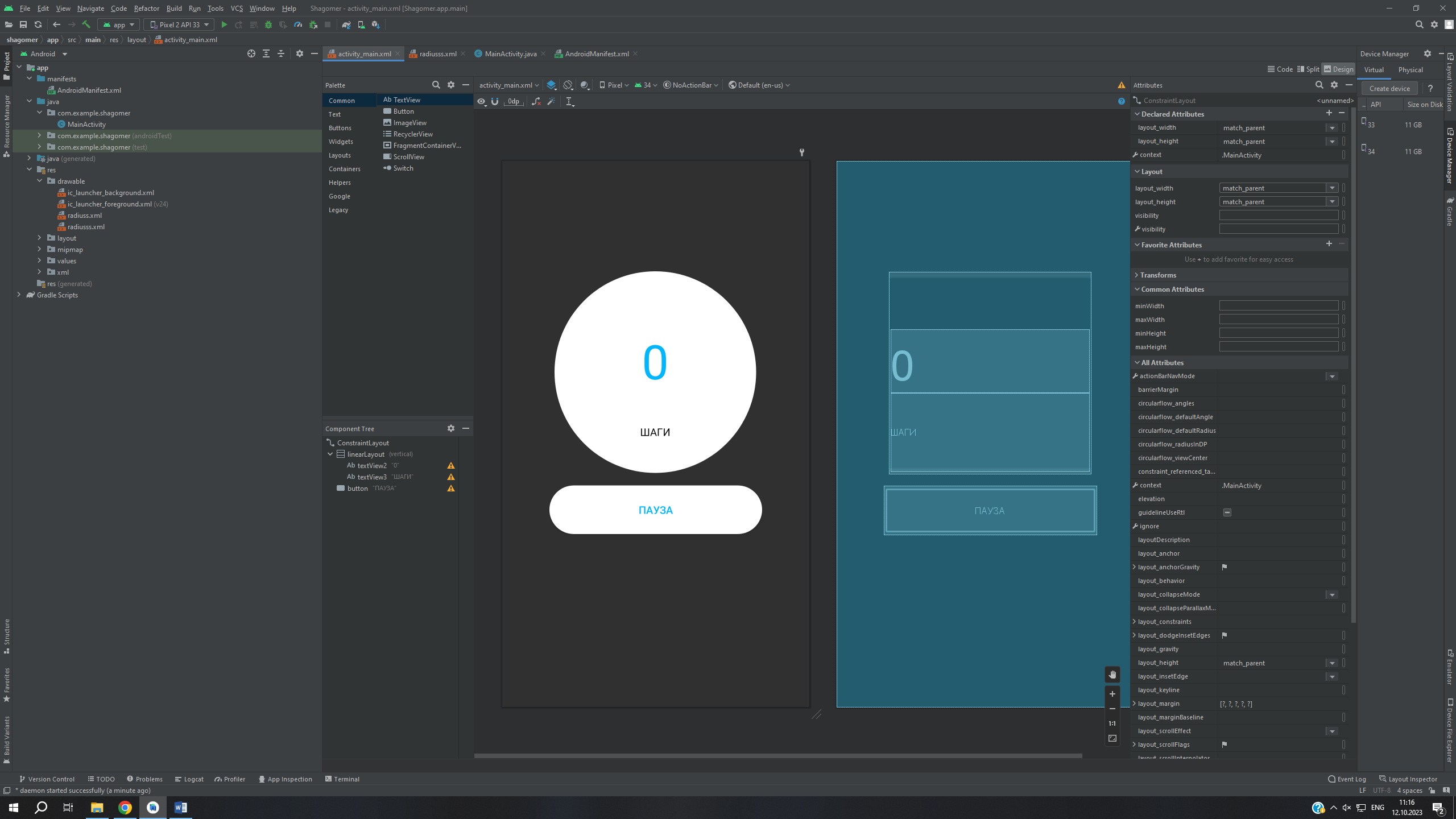
**Radius\_button:**

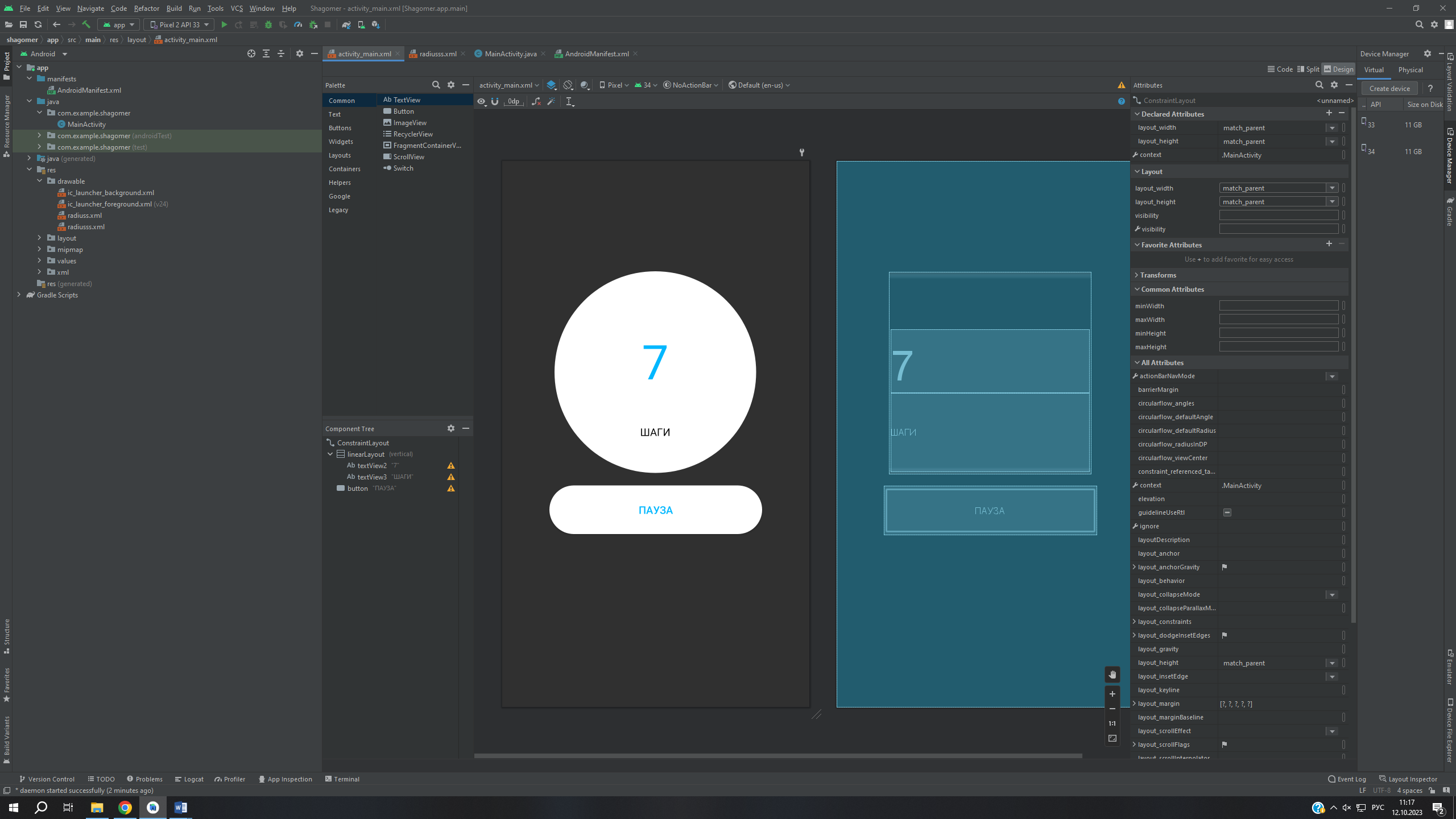
<?xml version="1.0" encoding="utf-8"?>  
<shape xmlns:android="http://schemas.android.com/apk/res/android">  
 android:shape="rectangle">  
 <solid android:color= "#FFFFFF"/>  
  
 <stroke android:widht="0dp"  
 android:color="#ff000000"/>  
  
 <padding android:left ="2dp"  
 android:top= "2dp"  
 android:right= "2dp"  
 android:bottom="2dp" />  
  
 <corners android:radius= "30dp"/>  
</shape>

**MainActivity:**

ublic class MainActivity extends AppCompatActivity implements SensorEventListener {  
  
 public boolean active = true;  
 private SensorManager sensorManager;  
 private int count = 0;  
 private TextView text;  
 private long lastUpdate;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 text = findViewById(R.id.*textView2*);  
 text.setText(String.*valueOf*(count));  
 sensorManager=(SensorManager) getSystemService(*SENSOR\_SERVICE*);  
  
 sensorManager.registerListener((SensorEventListener) this,  
 sensorManager.getDefaultSensor(Sensor.*TYPE\_ACCELEROMETER*),  
 SensorManager.*SENSOR\_DELAY\_FASTEST*);  
  
 lastUpdate = System.*currentTimeMillis*();  
 }  
  
 @Override  
 protected void onResume(){  
 super.onResume();  
 sensorManager.registerListener((SensorEventListener) this,  
 sensorManager.getDefaultSensor(Sensor.*TYPE\_ACCELEROMETER*),  
 SensorManager.*SENSOR\_DELAY\_NORMAL*);  
 }  
  
 @Override  
 protected void onPause(){  
 super.onPause();  
 sensorManager.unregisterListener((SensorEventListener) this);  
 }  
  
 public void onStoped(View view){  
 active = !active;  
 if(!active){  
 Button button = findViewById(R.id.*button*);  
 button.setText("ВОЗОБНОВИТЬ");  
 }else {  
 Button button = findViewById(R.id.*button*);  
 button.setText("ПАУЗА");  
 }  
 }  
  
 public void onSensorChanged(SensorEvent event){  
 if(event.sensor.getType() == Sensor.*TYPE\_ACCELEROMETER*){  
 float[] values = event.values;  
 float x= values[0];  
 float y= values[1];  
 float z= values[2];  
  
 float accelationSquareRoot=(x\* x+ y\* y+ z\* z)  
 /(SensorManager.*GRAVITY\_EARTH*\* SensorManager.*GRAVITY\_EARTH*);  
  
 long actualTime=System.*currentTimeMillis*();  
  
 if(accelationSquareRoot>=2)  
 {  
 if(actualTime- lastUpdate<200){  
 return;  
 }  
 lastUpdate=actualTime;  
  
 count ++;  
 text.setText(String.*valueOf*(count));  
 }  
 }  
 }  
  
 @Override  
 public void onAccuracyChanged(Sensor sensor, int accuracy){  
  
 }  
}

**Результат работы:**





**Ссылка на git:**

[GitHub - Sonykapinskya1109/Shagomer](https://github.com/Sonykapinskya1109/Shagomer)