



Unidad Profesional Interdisciplinaria en Ingeniería y
Tecnologías Avanzadas

DISPOSITIVOS MÓVILES

Práctica 21

Grupo: 2TM19

Gallegos Ruiz Diana Abigail

1 XML

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   android:orientation="vertical"
8   android:background="@color/cardview_dark_background"
9   android:padding="30dp"
10  >
11
12  <TextView
13    android:gravity="center"
14    android:layout_width="match_parent"
15    android:layout_height="wrap_content"
16    android:text="Animaci n con Animation Listener"
17    android:textColor="@color/white"
18    android:textSize="16pt"/>
19
20
21  <Button
22    android:layout_marginTop="50dp"
23    android:text="Animaci n"
24    android:id="@+id/boton1"
25    android:layout_width="match_parent"
26    android:layout_height="wrap_content" />
27
28  <TextView
29    android:id="@+id/texto1"
30    android:layout_marginTop="80dp"
31    android:gravity="center"
32    android:layout_width="match_parent"
33    android:layout_height="wrap_content"
34    android:text="Contador = 10 "
35    android:textColor="@color/white"
36    android:textSize="10pt"/>
37
38 </LinearLayout>
```

2 JAVA

```
1 package com.example.listener_animation;
2 import androidx.appcompat.app.AppCompatActivity;
3
4 import android.os.Bundle;
5 import android.view.View;
```

```
6  import android.view.animation.AlphaAnimation;
7  import android.view.animation.Animation;
8  import android.view.animation.AnimationSet;
9  import android.view.animation.BounceInterpolator;
10 import android.view.animation.LinearInterpolator;
11 import android.view.animation.ScaleAnimation;
12 import android.view.animation.TranslateAnimation;
13 import android.widget.Button;
14 import android.widget.TextView;
15
16 public class MainActivity extends AppCompatActivity implements
    Animation.AnimationListener{
17
18     TextView txt1;
19     Button boton1;
20     Animation escalamiento,transparencia,traslacion,animboton;
21     int i=10;
22
23     @Override
24     protected void onCreate(Bundle savedInstanceState) {
25         super.onCreate(savedInstanceState);
26         setContentView(R.layout.activity_main);
27         txt1 = (TextView) findViewById(R.id.texto1);
28         boton1= (Button) findViewById(R.id.boton1);
29         txt1.setText( "Contador = " + i);
30
31
32         //Animaci n de transparencia (Aparici n)
33         transparencia = new AlphaAnimation(0,1);
34         transparencia.setInterpolator( new LinearInterpolator());
35         transparencia.setDuration(2000);
36         transparencia.setRepeatMode(Animation.RESTART);
37         transparencia.setRepeatCount(10);
38         txt1.startAnimation(transparencia);
39
40         //Animacion de escalamiento
41         int rs= ScaleAnimation.RELATIVE_TO_SELF;
42         escalamiento = new ScaleAnimation(1,2,1,5,rs,0.5f,rs,0.5f);
43         escalamiento.setDuration(5000);
44         escalamiento.setInterpolator(new LinearInterpolator());
45         escalamiento.setFillAfter(true);
46         //txt1.startAnimation(escalamiento);
47
48         //Animaci n de traslaci n
49         int rs2 = TranslateAnimation.RELATIVE_TO_SELF;
50         traslacion = new TranslateAnimation(rs2, 0, rs2, 0, rs2, 0, rs2, 0.3
51 f);
52         traslacion.setDuration(10000);
53         traslacion.setInterpolator(new BounceInterpolator());
54         traslacion.setFillAfter(true);
55         //txt1.startAnimation(traslacion);
```

```
55
56 //Animaci n del bot n
57 animboton = new ScaleAnimation(1,1,0,1);
58 animboton.setInterpolator(new BounceInterpolator ( ) );
59 animboton.setFillAfter(true);
60 animboton.setDuration(500);
61 animboton.setRepeatMode(AnimationSet.RESTART);
62 animboton.setRepeatCount(1);
63
64 transparencia.setAnimationListener(this);
65 escalamiento.setAnimationListener(this);
66
67 boton1.setOnClickListener(new View.OnClickListener() {
68     @Override
69     public void onClick(View view) {
70         boton1.startAnimation(animboton);
71     }
72 });
73
74 }
75
76 @Override
77 public void onAnimationStart(Animation animation) {
78 }
79
80 @Override
81 public void onAnimationEnd(Animation animation) {
82     if (animation == transparencia){
83         txt1.setText("BOOM!");
84         txt1.startAnimation(escalamiento);
85     }
86     if (animation == escalamiento){
87         txt1.startAnimation(traslacion);
88     }
89 }
90
91 @Override
92 public void onAnimationRepeat(Animation animation) {
93     if (animation == transparencia){
94         i--;
95         txt1.setText("Contador= " + i);
96     }
97 }
98 }
99 }
100
```

3 Resultados

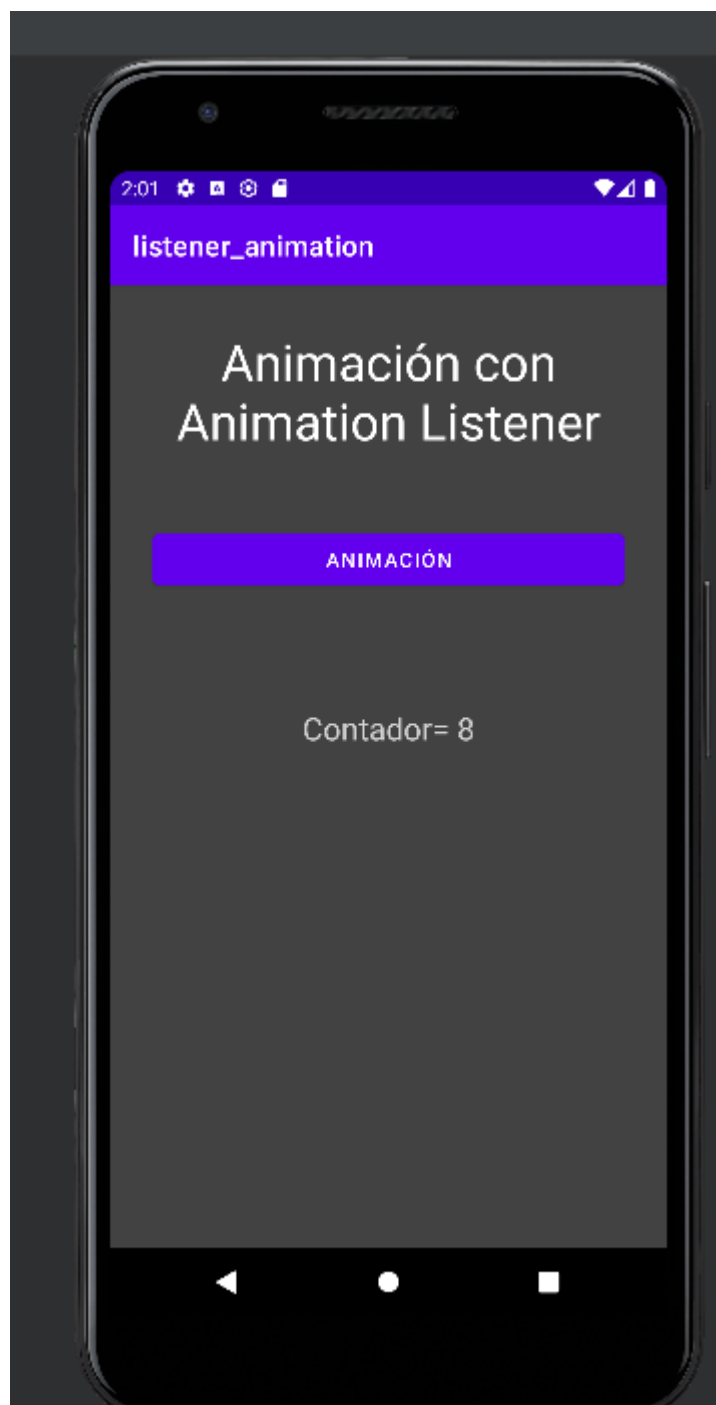


Figure 1: Animacin contador

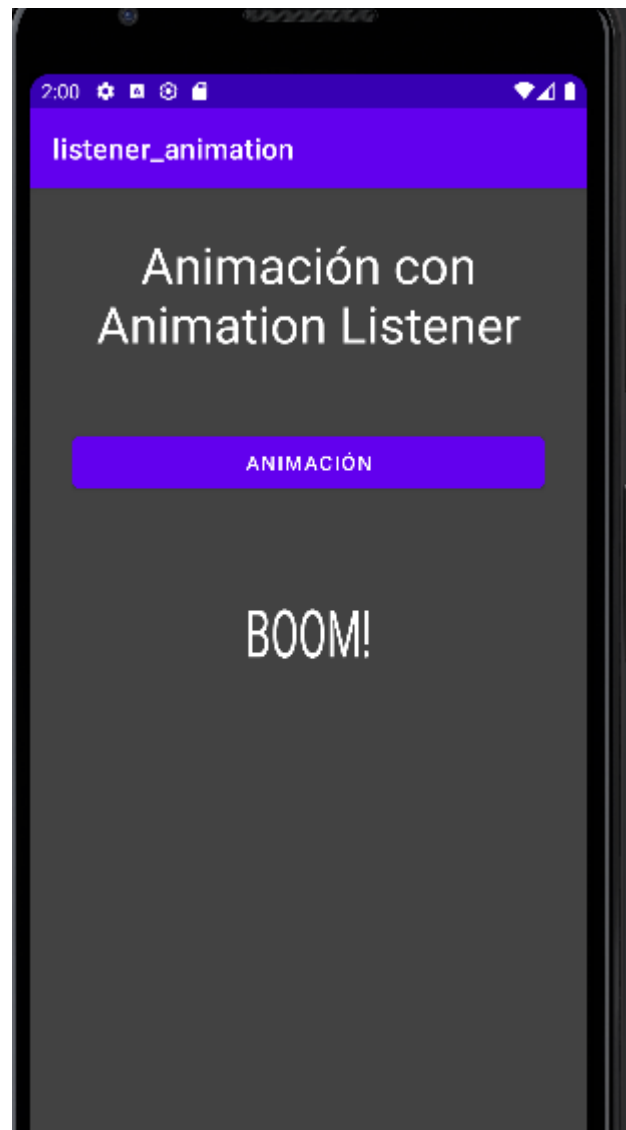


Figure 2: Animación escalamiento