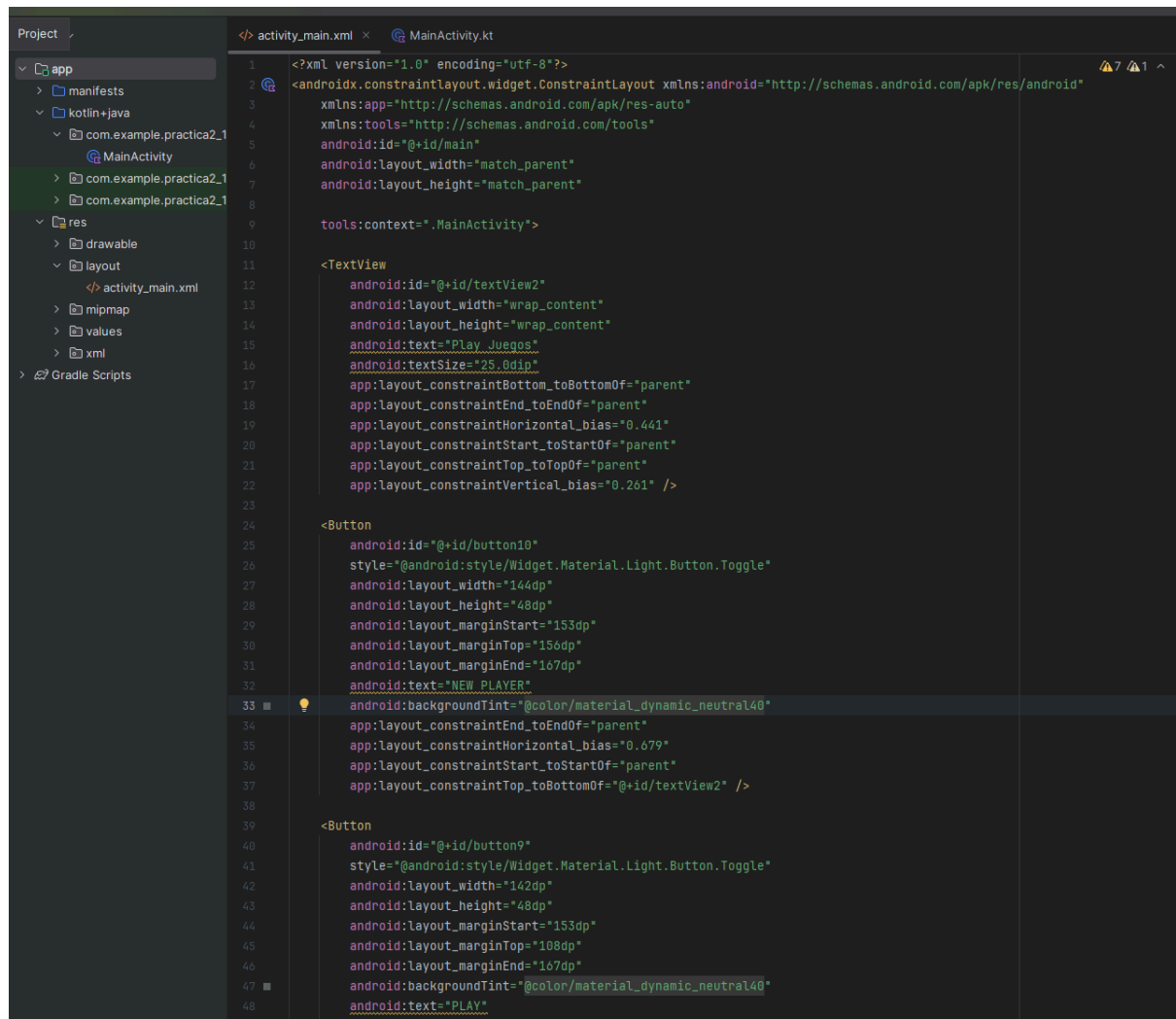


Enlace al github

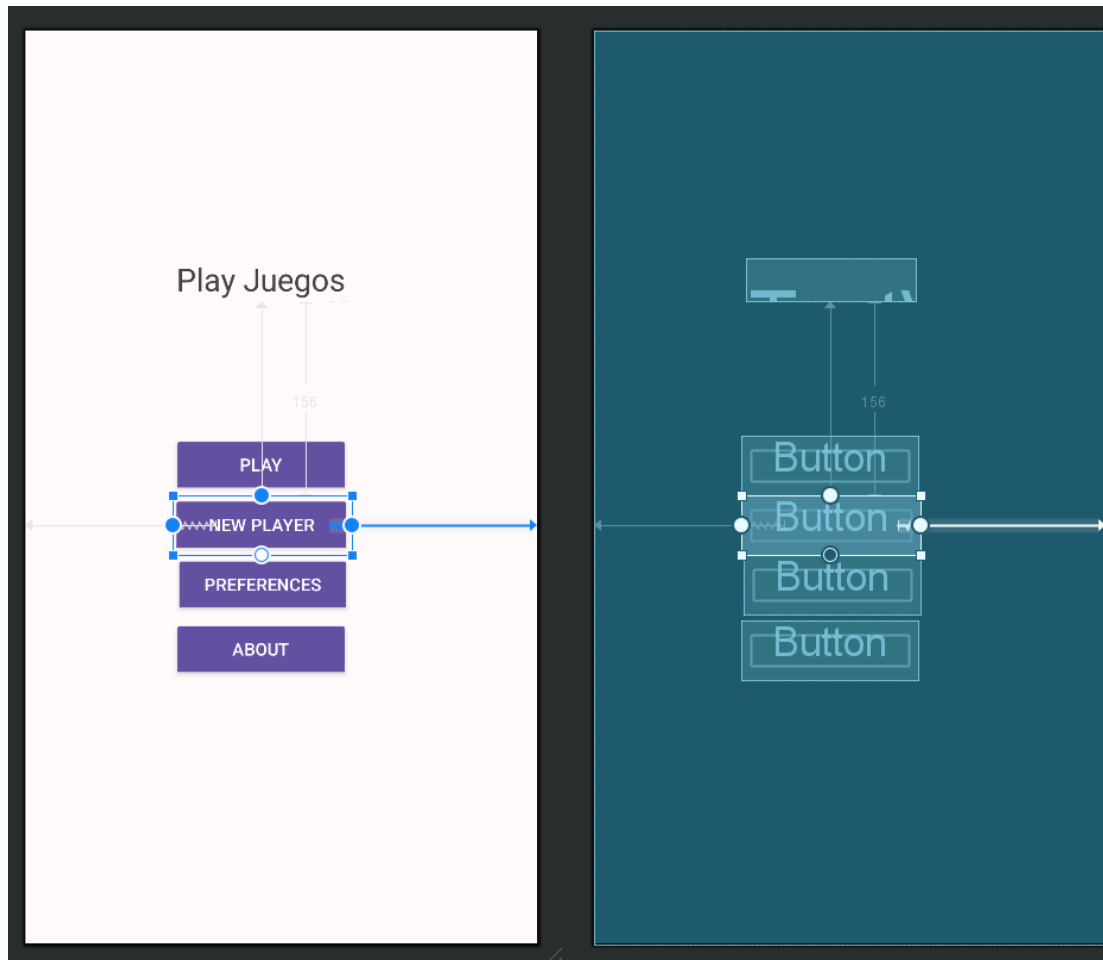
<https://github.com/K-Purpl3/PracticasLayout-Practica2.1>

PRÀCTICA 2.1. Panels (layouts)

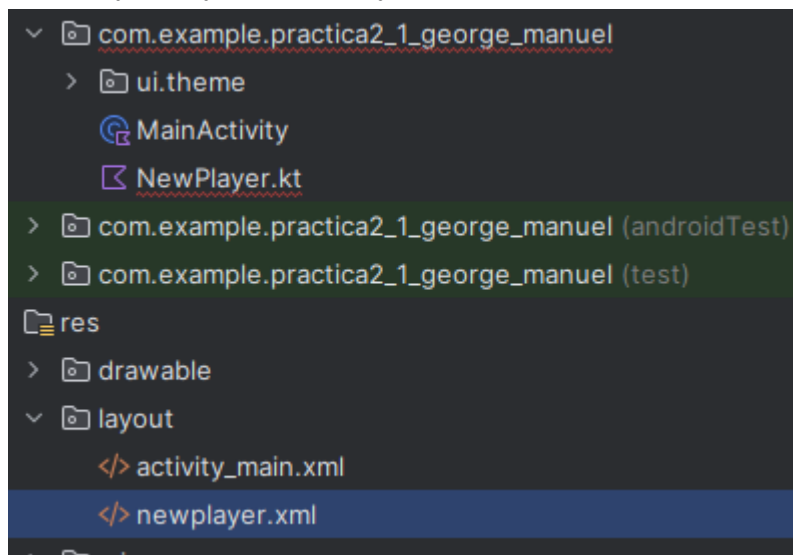
Ejercicio 1. LinearLayout. Cread un proyecto vacío (Play juegos) y elaborad en él un layout de nombre activity_main.xml. Este layout estará en la carpeta res/layout y hará referencia a la orientación portrait. Los componentes o vistas de este panel serán las siguientes: un TextView (id:textView) para el título, y cuatro Buttons (con id's: button, button1, button2 y button3). El resultado quedará como se indica

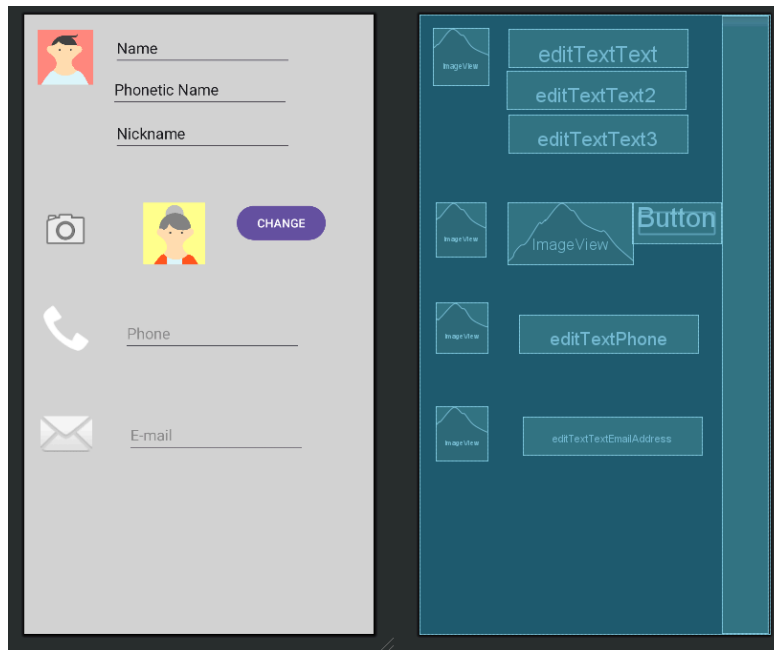


```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:id="@+id/main"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8
9     tools:context=".MainActivity">
10
11     <TextView
12         android:id="@+id/textView2"
13         android:layout_width="wrap_content"
14         android:layout_height="wrap_content"
15         android:text="Play Juegos"
16         android:textSize="25.0dip"
17         app:layout_constraintBottom_toBottomOf="parent"
18         app:layout_constraintEnd_toEndOf="parent"
19         app:layout_constraintHorizontal_bias="0.441"
20         app:layout_constraintStart_toStartOf="parent"
21         app:layout_constraintTop_toTopOf="parent"
22         app:layout_constraintVertical_bias="0.261" />
23
24     <Button
25         android:id="@+id/button10"
26         style="@android:style/Widget.Material.Light.Button.Toggle"
27         android:layout_width="144dp"
28         android:layout_height="48dp"
29         android:layout_marginStart="153dp"
30         android:layout_marginTop="156dp"
31         android:layout_marginEnd="167dp"
32         android:text="NEW PLAYER"
33         android:backgroundTint="@color/material_dynamic_neutral40"
34         app:layout_constraintEnd_toEndOf="parent"
35         app:layout_constraintHorizontal_bias="0.679"
36         app:layout_constraintStart_toStartOf="parent"
37         app:layout_constraintTop_toBottomOf="@+id/textView2" />
38
39     <Button
40         android:id="@+id/button9"
41         style="@android:style/Widget.Material.Light.Button.Toggle"
42         android:layout_width="142dp"
43         android:layout_height="48dp"
44         android:layout_marginStart="153dp"
45         android:layout_marginTop="108dp"
46         android:layout_marginEnd="167dp"
47         android:backgroundTint="@color/material_dynamic_neutral40"
48         android:text="PLAY"
49         app:layout_constraintEnd_toEndOf="parent"
50         app:layout_constraintHorizontal_bias="0.679"
51         app:layout_constraintStart_toStartOf="parent"
52         app:layout_constraintTop_toBottomOf="@+id/button10" />
53 </androidx.constraintlayout.widget.ConstraintLayout>
54 </tools:context>
```



Ejercicio 2. Tablelayout. En el proyecto anterior vamos a crear una nueva actividad (NewPlayer.kt) y un nuevo layout, ponéde de nombre (newplayer.xml)





Android manifest

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

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="Practica2_1_George_Manuel"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Practica2_1_George_Manuel"
        tools:targetApi="31">
        <activity
            android:name=".NewPlayer.kt"
            android:exported="false"
            android:label="NewPlayer.kt"
            android:theme="@style/Theme.Practica2_1_George_Manuel" />
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

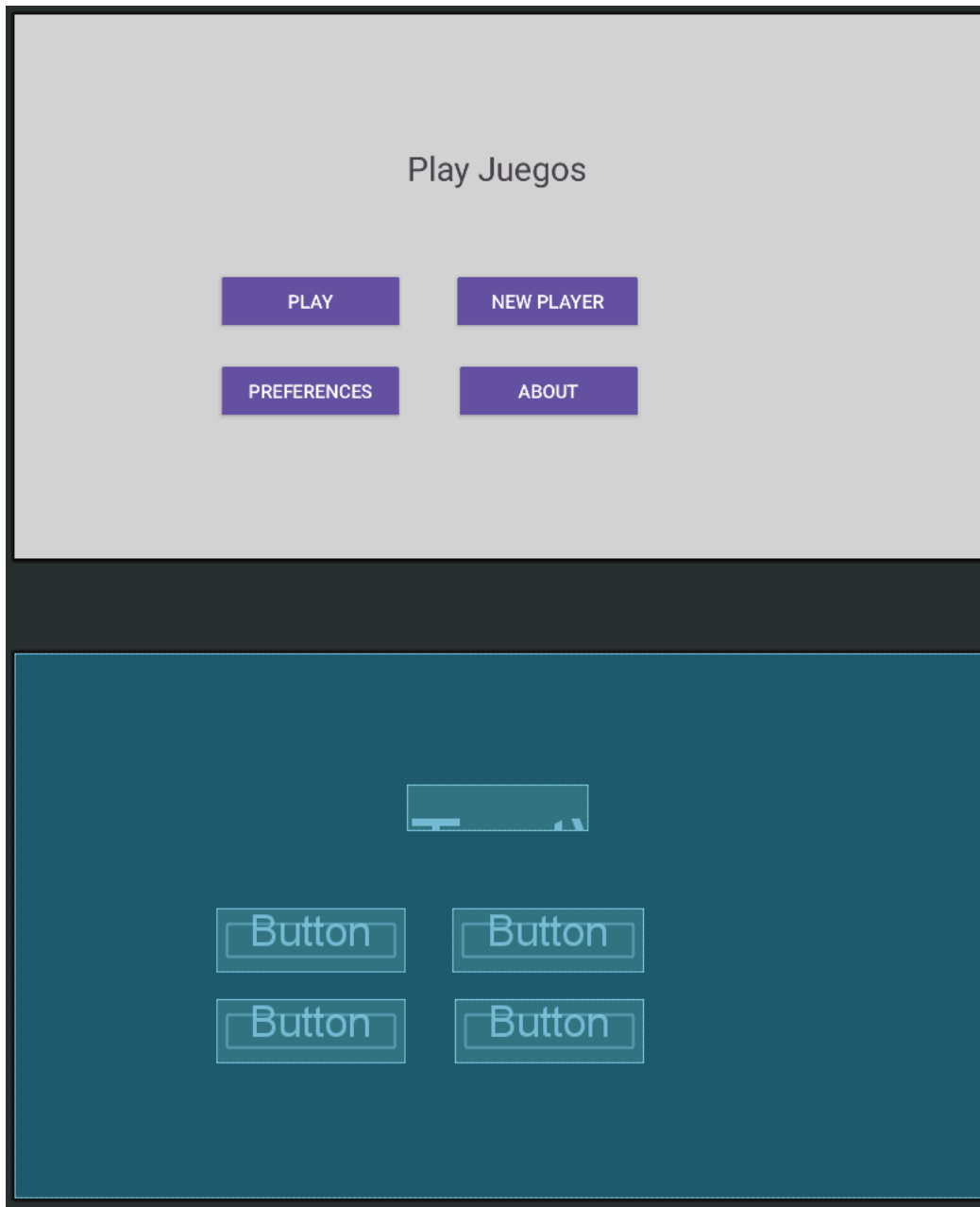
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Ejercicio 3. La aplicación funcionando...Una vez comprobado que funcionan las dos actividades y que los dos layout presentan un aspecto parecido al esperado, pasamos a darle funcionalidad real a la aplicación.

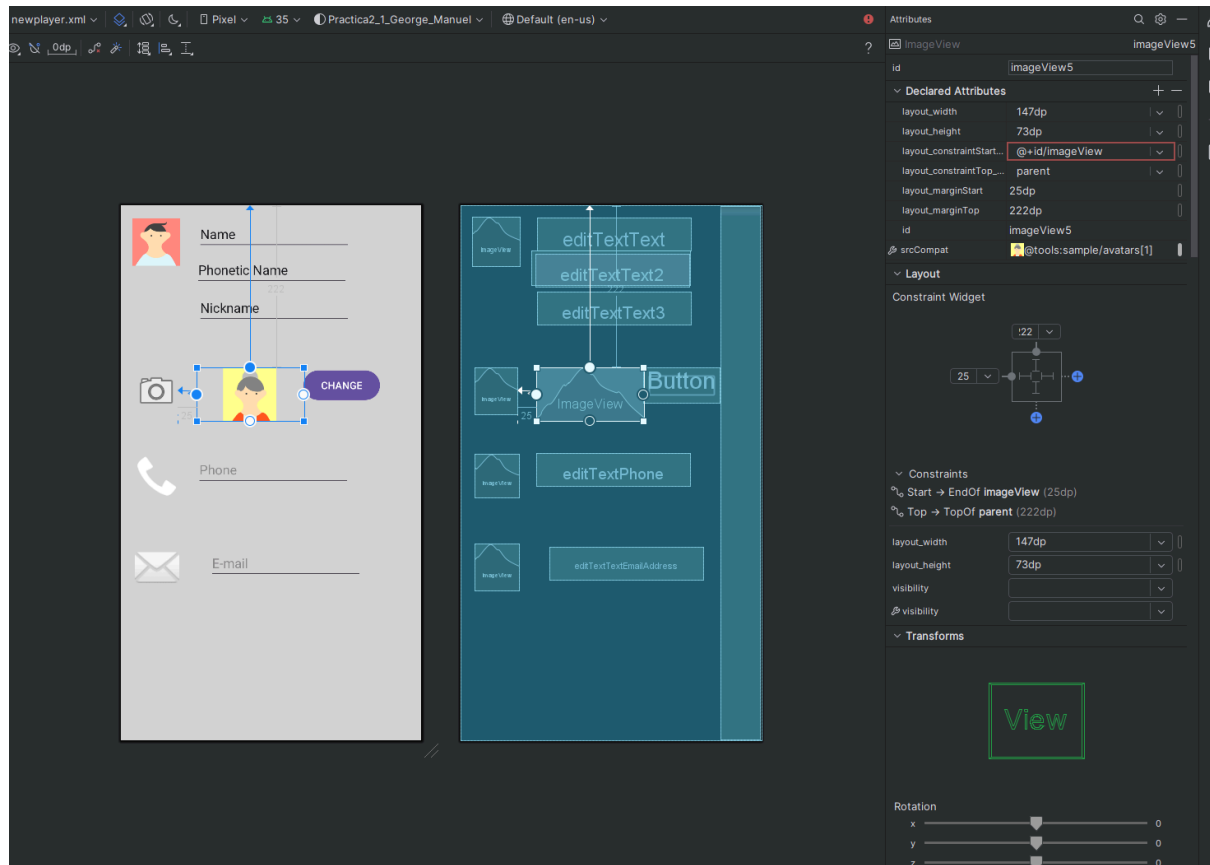
El funcionamiento será el siguiente: Al cargarse la aplicación veremos en pantalla el layout `activity_main.xml`. Al pulsar sobre el botón New Player de dicho layout, se lanzará la actividad (NewPlayer) que mostrará en pantalla el layout `newplayer.xml`. El código en Kotlin para lanzar una actividad después de pulsar un botón (`button10`, en mi caso) es:

```
1 package com.example.practica2_1_george_manuel
2
3 > import ...
4
5
6
7
8
9
10
11 class MainActivity : AppCompatActivity() {
12     /*override fun onCreate(savedInstanceState: Bundle?) {
13         super.onCreate(savedInstanceState)
14         enableEdgeToEdge()
15         setContentView(R.layout.activity_main)
16         ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
17             val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())
18             v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
19             insets
20         }
21     }*/
22
23     override fun onCreate(savedInstanceState: Bundle?, persistentState: PersistableBundle?) {
24         super.onCreate(savedInstanceState, persistentState)
25         setContentView(R.layout.activity_main)
26
27         val jugador = findViewById(R.id.button10) as Button
28         jugador.setOnClickListener { lanzarNewPlayer() }
29
30         fun lanzarNewPlayer() {
31             val i = Intent(this, NewPlayer::class.java)
32             startActivity(i)
33         }
34     }
```

Ejercicio 4. Orientación retrato/paisaje (portrait/landscape). Cuando diseñamos una aplicación, los paneles (layouts) que la conforman no siempre se visualizan de manera estéticamente correcta en ambas orientaciones.



Ejercicio 5



Enlace al github

<https://github.com/K-Purpl3/PracticasLayout-Practica2.1>