

# Programación multimedia y dispositivos móviles

Actividad 2.4.2

Francisco José García Cutillas | 2FPGS\_DAM

---

## Índice

Ejercicio 1 .....	3
-------------------	---

## Ejercicio 1

Siguiendo con el caso práctico anterior, vamos a añadir funcionalidad al menú que hemos generado para nuestra aplicación.

En primer lugar, tenéis que reestructurar la aplicación: ahora crear una nueva MainActivity, a la que darás el contenido más adelante, y la actividad con la que estamos trabajando pasará a llamarse EditParkActivity.

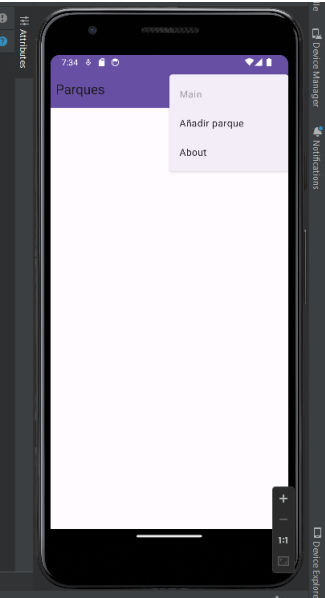
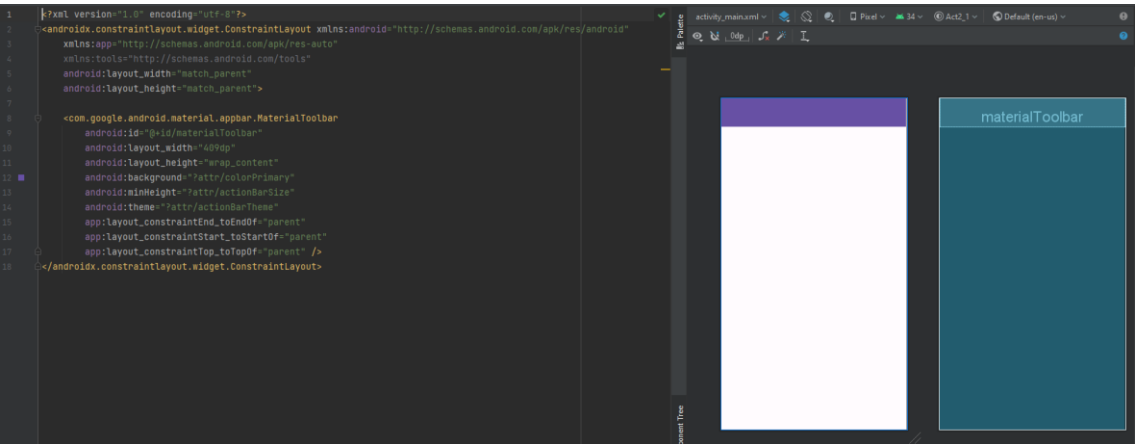
Una vez hecho esto, se pide incorporar la funcionalidad siguiente:

- El menú deberá aparecer tanto en la actividad principal nueva MainActivity como en EditParkActivity.
- Cuando se seleccione la opción main, se mostrará la interfaz de la actividad principal (de momento está vacía).
- Cuando se seleccione la opción AddPark, se mostrará la interfaz de la actividad EditParkActivity con el contenido que tenía hasta el momento, incluidos los diálogos.
- Cuando nos encontremos en una actividad, esta aparecerá desactivada en el menú.
- Cuando se abra una actividad, si esta ya está en la pila de actividades, solamente deberemos reordenarla para ponerla en primer plano.

```

1 package com.example.act2_1
2
3 import ...
4
5
6
7 class MainActivity : AppCompatActivity() {
8
9     private lateinit var binding : ActivityMainBinding
10
11     override fun onCreate(savedInstanceState: Bundle?) {
12         super.onCreate(savedInstanceState)
13         binding = ActivityMainBinding.inflate(layoutInflater)
14         val view = binding.root
15         setContentView(view)
16         setSupportActionBar(findViewById(R.id.materialToolbar))
17
18         val intent = Intent(packageContext, MainActivity::class.java)
19         intent.addFlags(Intent.FLAG_ACTIVITY_REORDER_TO_FRONT)
20         startActivity(intent)
21     }
22 }
23
24
25
26
27

```



```
1 package com.example.act2_1
2
3 import ...
4
15 class EditParkActivity : AppCompatActivity() {
16
17     private lateinit var editTextName: EditText
18     private lateinit var editTextDescription: EditText
19     private lateinit var editTextPhone: EditText
20     private lateinit var editTextWebsite: EditText
21     private lateinit var spinnerOpening: Spinner
22     private lateinit var spinnerClosing: Spinner
23     private lateinit var checkBoxSports: CheckBox
24     private lateinit var checkBoxChildrenPark: CheckBox
25     private lateinit var checkBoxBar: CheckBox
26     private lateinit var checkBoxMascotas: CheckBox
27     private lateinit var buttonSave: Button
28     private lateinit var binding: EditParkActivityBinding
29
30     override fun onCreate(savedInstanceState: Bundle?) {
31         super.onCreate(savedInstanceState)
32         binding = EditParkActivityBinding.inflate(layoutInflater)
33         val view = binding.root
34         setContentView(view)
35         setSupportActionBar(findViewById(R.id.materialToolbar))
36
37         val intent = Intent(packageContext, this, EditParkActivity::class.java)
38         intent.addFlags(Intent.FLAG_ACTIVITY_REORDER_TO_FRONT)
39         startActivity(intent)
40
41         spinner()
42     }
```

```

43     editTextName = binding.Name
44     editTextDescription = binding.Description
45     editTextPhone = binding.Phone
46     editTextWebsite = binding.webSite
47     spinnerOpening = binding.SpinnerOpeningTime
48     spinnerClosing = binding.SpinnerClosingTime
49     checkBoxSports = binding.cbSport
50     checkBoxChildrenPark = binding.cbChildren
51     checkBoxBar = binding.cbBar
52     checkBoxMascotas = binding.cbMascotas
53     buttonSave = binding.btSave
54
55     buttonSave.setOnClickListener { it: View!
56
57         val name = editTextName.text.toString()
58         val description = editTextDescription.text.toString()
59         val phone = editTextPhone.text.toString()
60         val website = editTextWebsite.text.toString()
61         val opening = spinnerOpening.selectedItem.toString()
62         val closing = spinnerClosing.selectedItem.toString()
63         var sports = ""
64         var childrenPark = ""
65         var bar = ""
66         var mascotas = ""
67
68         if (checkBoxSports.isChecked){
69             sports = "Con instalaciones deportivas"
70
71         } else {
72
73             sports = "Sin instalaciones deportivas"
74
75         }
76

```

```

78         if (checkBoxChildrenPark.isChecked){
79             childrenPark = "Con instalaciones para niños"
80
81         } else {
82
83             childrenPark = "Sin instalaciones para niños"
84
85         }
86
87         if (checkBoxBar.isChecked){
88             bar = "Con zona de restauración"
89
90         } else {
91
92             bar = "Sin zona de restauración"
93
94         }
95
96         if (checkBoxMascotas.isChecked){
97             mascotas = "Con zona para mascotas"
98
99         } else {
100
101             mascotas = "Sin zona para mascotas"
102
103         }
104
105         mostrarDialogoConfirmacion(name, description, phone, website, opening, closing, sports, childrenPark,
106                                     bar, mascotas)
107

```

```

111     }
112
113 }
114
115 fun spinner(){
116     //Spinner del layout
117     spinnerClosing = binding.spinnerClosingTime
118
119     val adaptador = ArrayAdapter.createFromResource( context this, R.array.horas, android.R.layout.simple_dropdown_item_1line)
120
121     spinnerClosing.adapter = adaptador
122
123 }
124
125
126 fun mostrarDialogoConfirmacion(nombre :String, descripcion : String, telefono : String, web : String,
127     horaApertura :String, horaCierre :String, instDep : String,
128     instNin :String, rest :String, masc :String){
129
130     var dialogoConfirmacion = AlertDialog.Builder( context this)
131     dialogoConfirmacion.setTitle("Confirmación")
132     dialogoConfirmacion.setMessage(mensajeMostrar(nombre, descripcion, telefono, web, horaApertura, horaCierre,
133         instDep, instNin, rest, masc))
134     dialogoConfirmacion.setPositiveButton(android.R.string.ok, DialogInterface.OnClickListener {dialog, which ->
135
136         toast( mensajeToast: "Se han guardado los datos")
137     })
138     dialogoConfirmacion.setNegativeButton(android.R.string.cancel, DialogInterface.OnClickListener {dialog, which ->
139
140         toast( mensajeToast: "Acción cancelada")
141     })
142
143 }
144
145 val dialogo = dialogoConfirmacion.create()
146 dialogo.show()
147
148 }
149
150 fun toast(mensajeToast: String){
151
152     Toast.makeText(applicationContext, mensajeToast, Toast.LENGTH_SHORT).show()
153
154 }
155
156 fun mensajeMostrar (nombre :String, descripcion : String, telefono : String, web : String,
157     horaApertura :String, horaCierre :String, instDep : String,
158     instNin :String, rest :String, masc :String) :String{
159
160     return ("Nombre: " + nombre + "\n" +
161         "Descripción: " + descripcion + "\n" +
162         "Teléfono: " + telefono + "\n" +
163         "Web: " + web + "\n" +
164         "Hora de apertura: " + horaApertura + "\n" +
165         "Hora de cierre: " + horaCierre + "\n" +
166         instDep + "\n" +
167         instNin + "\n" +
168         rest + "\n" +
169         masc + "\n\n" +
170         "¿Está seguro que desea guardar los datos?")
171
172 }
173
174 }

```

```

1  <?xml version="1.0" encoding="utf-8"?>
2
3  <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      android:fillViewport="true">
9
10
11     <androidx.constraintlayout.widget.ConstraintLayout
12         android:layout_width="match_parent"
13         android:layout_height="wrap_content"
14         android:backgroundTintMode="screen"
15         android:isScrollContainer="true"
16         tools:context=".EditParkActivity">
17
18         <ImageView
19             android:id="@+id/Image"
20             android:layout_width="0dp"
21             android:layout_height="149dp"
22             android:layout_marginStart="16dp"
23             android:layout_marginTop="31dp"
24             android:layout_marginEnd="16dp"
25             android:background="@drawable/appimg"
26             android:contentDescription="ImagenPark"
27             android:scaleType="fitCenter"
28             app:layout_constraintEnd_toEndOf="parent"
29             app:layout_constraintHorizontal_bias="1.0"
30             app:layout_constraintStart_toStartOf="parent"
31             app:layout_constraintTop_toBottomOf="@+id/materialToolbar"
32             tools:ignore="ImageContrastCheck" />

```

```

34     <EditText
35         android:id="@+id/Name"
36         android:layout_width="0dp"
37         android:layout_height="wrap_content"
38         android:layout_marginTop="16dp"
39         android:ems="10"
40         android:inputType="textPersonName"
41         android:minHeight="48dp"
42         android:text="Name"
43         app:layout_constraintEnd_toEndOf="@+id/Image"
44         app:layout_constraintStart_toStartOf="@+id/Image"
45         app:layout_constraintTop_toBottomOf="@+id/Image" />
46
47     <EditText
48         android:id="@+id/Description"
49         android:layout_width="0dp"
50         android:layout_height="wrap_content"
51         android:ems="10"
52         android:gravity="start|top"
53         android:inputType="textMultiLine"
54         android:minHeight="48dp"
55         android:text="Description"
56         app:layout_constraintEnd_toEndOf="@+id/Name"
57         app:layout_constraintStart_toStartOf="@+id/Name"
58         app:layout_constraintTop_toBottomOf="@+id/Name"
59         tools:ignore="LabelFor" />
60

```



```

62 <EditText
63     android:id="@+id/Phone"
64     android:layout_width="0dp"
65     android:layout_height="wrap_content"
66     android:ems="10"
67     android:inputType="text|numberPassword"
68     android:minHeight="48dp"
69     android:singleLine="false"
70     android:text="PhoneNumber"
71     app:layout_constraintEnd_toEndOf="@+id/Description"
72     app:layout_constraintStart_toStartOf="@+id/Description"
73     app:layout_constraintTop_toBottomOf="@+id/Description" />
74
75 <EditText
76     android:id="@+id/webSite"
77     android:layout_width="0dp"
78     android:layout_height="wrap_content"
79     android:ems="10"
80     android:gravity="start|top"
81     android:inputType="textMultiLine"
82     android:minHeight="48dp"
83     android:text="Web"
84     app:layout_constraintEnd_toEndOf="@+id/Phone"
85     app:layout_constraintStart_toStartOf="@+id/Phone"
86     app:layout_constraintTop_toBottomOf="@+id/Phone" />
87
88 <TextView
89     android:id="@+id/textViewOpening"
90     android:layout_width="0dp"
91     android:layout_height="wrap_content"
92     android:text="OpeningTime"
93     app:layout_constraintEnd_toStartOf="@+id/textViewClose"
94     app:layout_constraintStart_toStartOf="@+id/webSite"
95     app:layout_constraintTop_toBottomOf="@+id/webSite" />
96
97 <Spinner
98     android:id="@+id/SpinnerOpeningTime"
99     android:layout_width="0dp"
100    android:layout_height="wrap_content"
101    android:contentDescription="OpeningTime"
102    android:entries="@array/horas"
103    android:minHeight="48dp"
104    android:spinnerMode="dialog"
105    app:layout_constraintEnd_toEndOf="@+id/textViewOpening"
106    app:layout_constraintStart_toStartOf="@+id/textViewOpening"
107    app:layout_constraintTop_toBottomOf="@+id/textViewOpening" />
108
109 <TextView
110     android:id="@+id/textViewClose"
111     android:layout_width="0dp"
112     android:layout_height="wrap_content"
113     android:text="ClosingTime"
114     app:layout_constraintEnd_toEndOf="@+id/webSite"
115     app:layout_constraintStart_toEndOf="@+id/textViewOpening"
116     app:layout_constraintTop_toBottomOf="@+id/webSite" />

```

```

118 <Spinner
119     android:id="@+id/SpinnerClosingTime"
120     android:layout_width="0dp"
121     android:layout_height="wrap_content"
122     android:contentDescription="ClosingTime"
123     android:minHeight="48dp"
124     android:spinnerMode="dialog"
125     app:layout_constraintEnd_toEndOf="@+id/webSite"
126     app:layout_constraintStart_toStartOf="@+id/textViewClose"
127     app:layout_constraintTop_toTopOf="@+id/SpinnerOpeningTime" />
128
129 <TextView
130     android:id="@+id/textViewActivities"
131     android:layout_width="wrap_content"
132     android:layout_height="wrap_content"
133     android:layout_marginTop="16dp"
134     android:text="Activities"
135     app:layout_constraintStart_toStartOf="@+id/SpinnerOpeningTime"
136     app:layout_constraintTop_toBottomOf="@+id/SpinnerOpeningTime" />
137
138 <CheckBox
139     android:id="@+id/cb_sport"
140     android:layout_width="0dp"
141     android:layout_height="wrap_content"
142     android:minHeight="48dp"
143     android:text="Sports"
144     app:layout_constraintStart_toStartOf="@+id/textViewActivities"
145     app:layout_constraintTop_toBottomOf="@+id/textViewActivities" />

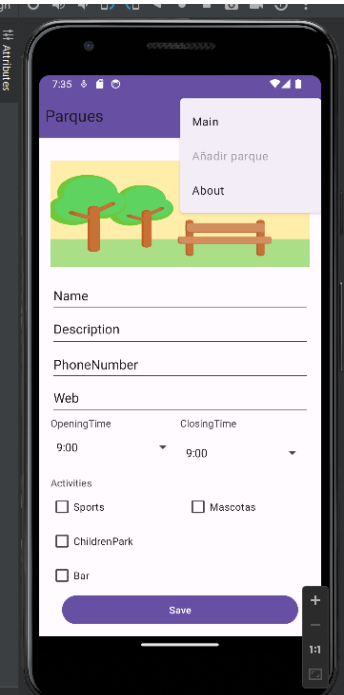
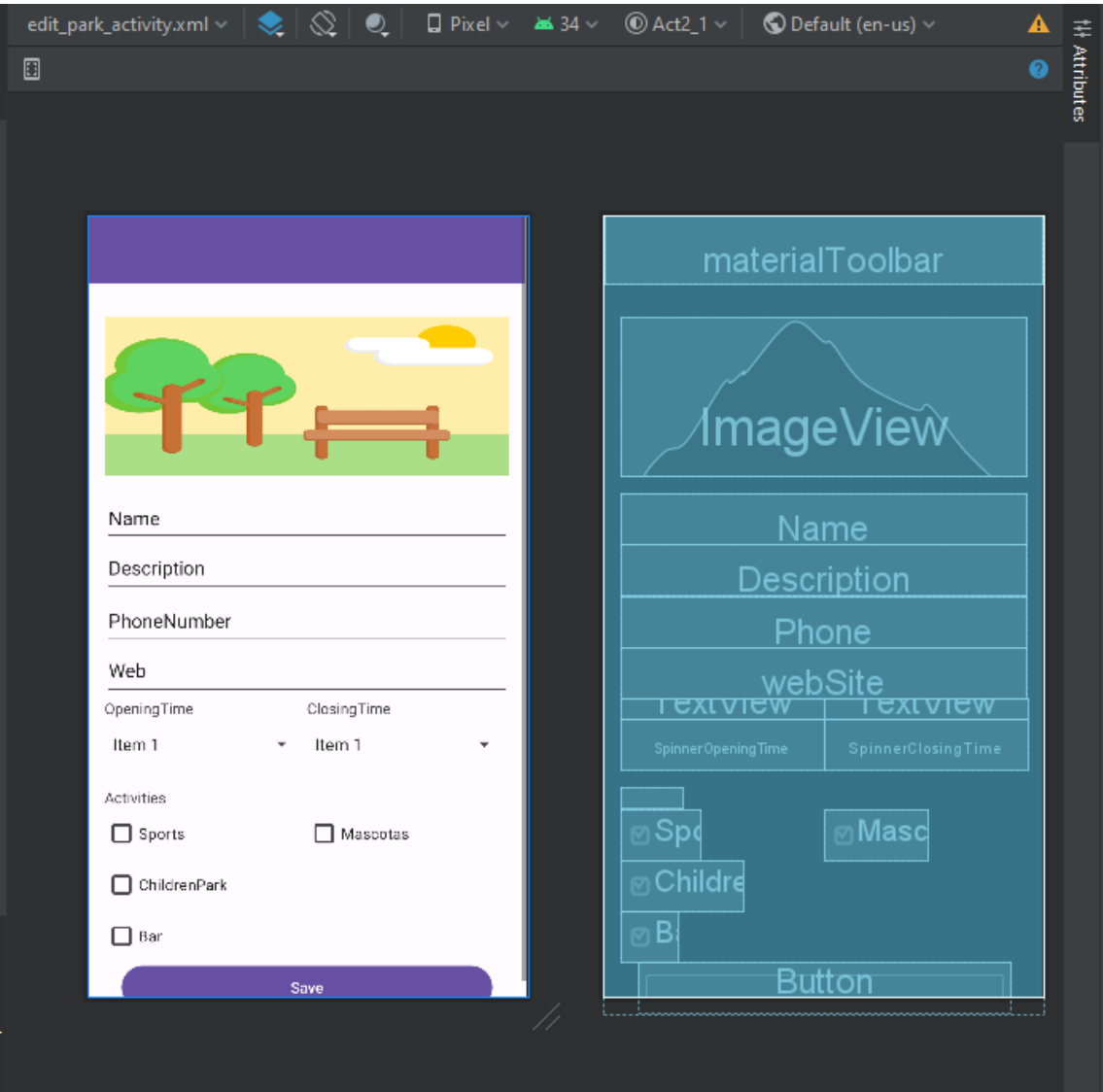
```

```

147 <CheckBox
148     android:id="@+id/cb_children"
149     android:layout_width="0dp"
150     android:layout_height="wrap_content"
151     android:minHeight="48dp"
152     android:text="ChildrenPark"
153     app:layout_constraintStart_toStartOf="@+id/cb_sport"
154     app:layout_constraintTop_toBottomOf="@+id/cb_sport" />
155
156 <CheckBox
157     android:id="@+id/cb_bar"
158     android:layout_width="0dp"
159     android:layout_height="wrap_content"
160     android:minHeight="48dp"
161     android:text="Bar"
162     app:layout_constraintStart_toStartOf="@+id/cb_children"
163     app:layout_constraintTop_toBottomOf="@+id/cb_children" />
164
165 <CheckBox
166     android:id="@+id/cb_mascotas"
167     android:layout_width="0dp"
168     android:layout_height="wrap_content"
169     android:layout_marginStart="132dp"
170     android:minHeight="48dp"
171     android:text="Mascotas"
172     app:layout_constraintStart_toEndOf="@+id/textViewActivities"
173     app:layout_constraintTop_toTopOf="@+id/cb_sport" />
174

```

```
175 <Button
176     android:id="@+id/btSave"
177     android:layout_width="0dp"
178     android:layout_height="wrap_content"
179     android:layout_marginStart="32dp"
180     android:layout_marginEnd="32dp"
181     android:text="Save"
182     app:layout_constraintBottom_toBottomOf="parent"
183     app:layout_constraintEnd_toEndOf="parent"
184     app:layout_constraintHorizontal_bias="0.0"
185     app:layout_constraintStart_toStartOf="parent"
186     app:layout_constraintTop_toBottomOf="@+id/cb_bar"
187     app:layout_constraintVertical_bias="0.0" />
188
189 <com.google.android.material.appbar.MaterialToolbar
190     android:id="@+id/materialToolbar"
191     android:layout_width="409dp"
192     android:layout_height="wrap_content"
193     android:background="?attr/colorPrimary"
194     android:minHeight="?attr/actionBarSize"
195     android:theme="?attr/actionBarTheme"
196     app:layout_constraintBottom_toTopOf="@+id/Image"
197     app:layout_constraintEnd_toEndOf="parent"
198     app:layout_constraintStart_toStartOf="parent"
199     app:layout_constraintTop_toTopOf="parent"
200     app:layout_constraintVertical_bias="0.107" />
201 </androidx.constraintlayout.widget.ConstraintLayout>
202 </ScrollView>
203
```



```

1 package com.example.act2_1
2
3 import ...
4
5 class AboutActivity : AppCompatActivity {
6
7     private lateinit var binding: ActivityAboutBinding
8     private lateinit var textViewVersion: TextView
9     private lateinit var textViewVersion2: TextView
10    override fun onCreate(savedInstanceState: Bundle?) {
11        super.onCreate(savedInstanceState)
12        binding = ActivityAboutBinding.inflate(layoutInflater)
13        val view = binding.root
14        setContentView(view)
15        setSupportActionBar(findViewById(R.id.materialToolbar))
16
17        val intent = Intent(packageContext, AboutActivity::class.java)
18        intent.addFlags(Intent.FLAG_ACTIVITY_REORDER_TO_FRONT)
19        startActivity(intent)
20
21        binding.textViewVersion.text = "Versión 1.0"
22        binding.textViewVersion2.text = "Francisco José García Cutillas"
23    }
24 }

```

```

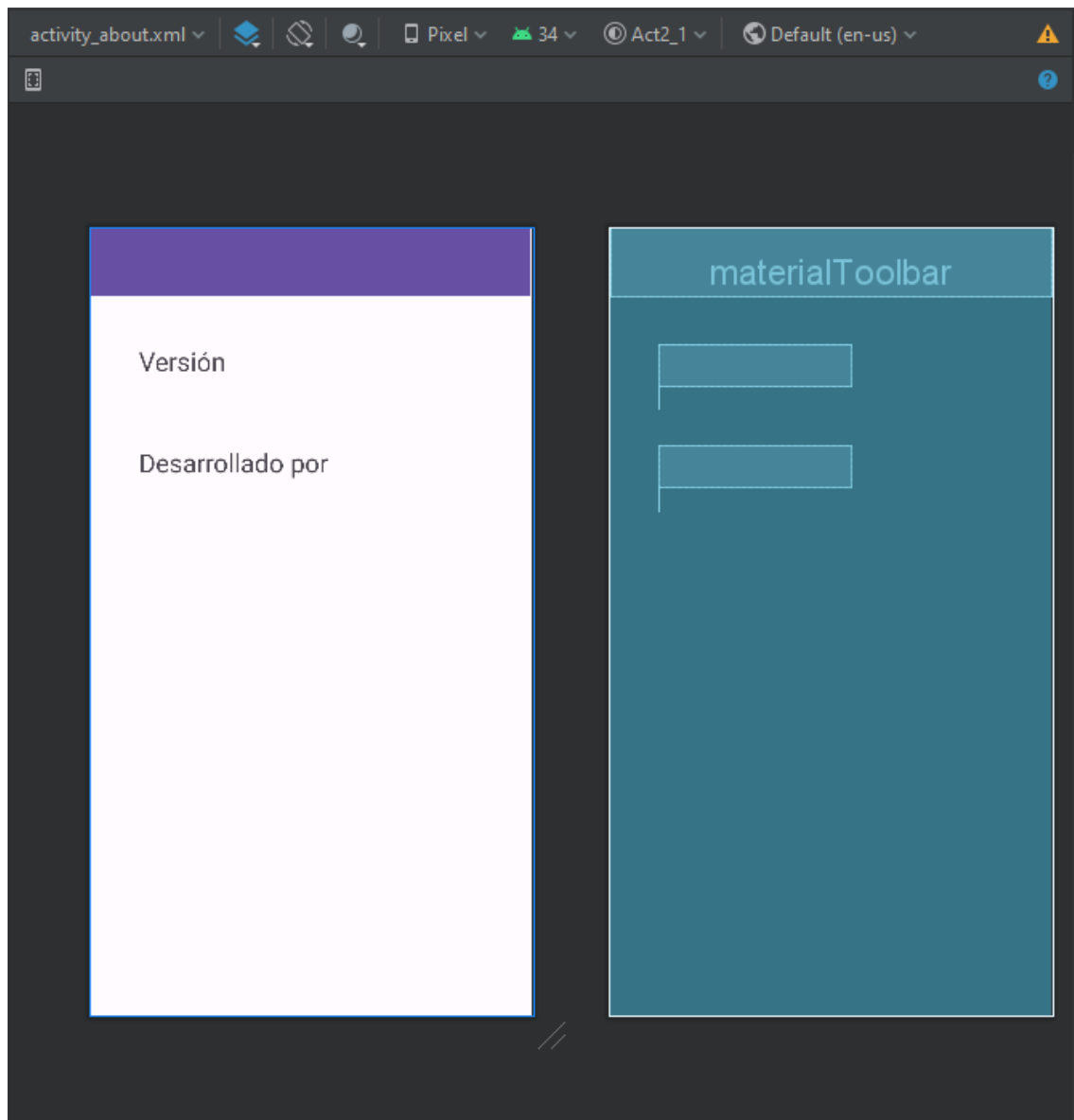
1 <?xml version="1.0" encoding="utf-8"?>
2
3 <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     android:fillViewport="true">
9
10
11     <androidx.constraintlayout.widget.ConstraintLayout
12         android:layout_width="match_parent"
13         android:layout_height="wrap_content"
14         android:backgroundTintMode="screen"
15         android:isScrollContainer="true"
16         tools:context=".EditParkActivity">
17
18
19         <TextView
20             android:id="@+id/viewVersion2"
21             android:layout_width="wrap_content"
22             android:layout_height="wrap_content"
23             android:textSize="16sp"
24             app:layout_constraintStart_toStartOf="@+id/desarrollador"
25             app:layout_constraintTop_toBottomOf="@+id/desarrollador" />
26

```

```

27 <TextView
28     android:id="@+id/desarrollador"
29     android:layout_width="179dp"
30     android:layout_height="38dp"
31     android:layout_marginTop="56dp"
32     android:text="Desarrollado por"
33     android:textSize="24sp"
34     app:layout_constraintEnd_toEndOf="@+id/version"
35     app:layout_constraintHorizontal_bias="0.0"
36     app:layout_constraintStart_toStartOf="@+id/version"
37     app:layout_constraintTop_toBottomOf="@+id/version" />
38
39 <com.google.android.material.appbar.MaterialToolbar
40     android:id="@+id/materialToolbar"
41     android:layout_width="409dp"
42     android:layout_height="wrap_content"
43     android:background="?attr/colorPrimary"
44     android:minHeight="?attr/actionBarSize"
45     android:theme="?attr/actionBarTheme"
46     app:layout_constraintEnd_toEndOf="parent"
47     app:layout_constraintStart_toStartOf="parent"
48     app:layout_constraintTop_toTopOf="parent"
49     app:layout_constraintVertical_bias="0.107" />
50
51 <TextView
52     android:id="@+id/version"
53     android:layout_width="179dp"
54     android:layout_height="38dp"
55     android:layout_marginTop="44dp"
56     android:text="Versión"
57     android:textSize="24sp"
58     app:layout_constraintEnd_toEndOf="parent"
59     app:layout_constraintHorizontal_bias="0.198"
60     app:layout_constraintStart_toStartOf="parent"
61     app:layout_constraintTop_toBottomOf="@+id/materialToolbar" />
62
63 <TextView
64     android:id="@+id/viewVersion"
65     android:layout_width="wrap_content"
66     android:layout_height="wrap_content"
67     android:textSize="16sp"
68     app:layout_constraintStart_toStartOf="@+id/version"
69     app:layout_constraintTop_toBottomOf="@+id/version" />
70 </androidx.constraintlayout.widget.ConstraintLayout>
71 </ScrollView>

```





```

1      package com.example.act2_1
2
3      import ...
4
5      open class ActivityWithMenu : AppCompatActivity() {
6
7          companion object {
8              var actividadActual = 0
9          }
10
11          override fun onCreateOptionsMenu(menu: Menu): Boolean {
12
13              val inflater: MenuInflater = menuInflater
14              inflater.inflate(R.menu.menu_principal, menu)
15              for (i in 0 until menu.size()){
16
17                  if (i == actividadActual){
18                      menu.getItem(i).isEnabled = false
19                  } else {
20                      menu.getItem(i).isEnabled = true
21                  }
22              }
23
24              return true
25          }
26      }

```



```

37 override fun onOptionsItemSelected(item: MenuItem): Boolean {
38     return when (item.itemId) {
39         R.id.main -> {
40             if (actividadActual != 0) {
41                 val intent = Intent( packageContext: this, MainActivity::class.java)
42                 actividadActual = 0
43                 startActivity(intent)
44             }
45             true
46         }
47
48         R.id.add_park -> {
49             if (actividadActual != 1) {
50                 val intent = Intent( packageContext: this, EditParkActivity::class.java)
51                 actividadActual = 1
52                 startActivity(intent)
53             }
54             true
55         }
56
57         R.id.about -> {
58             if (actividadActual != 2) {
59                 val intent = Intent( packageContext: this, AboutActivity::class.java)
60                 actividadActual = 2
61                 startActivity(intent)
62             }
63             true
64         }
65
66         else -> super.onOptionsItemSelected(item)
67     }
68 }
69
70

```

