

10. Toolbar

Alberto Luengo



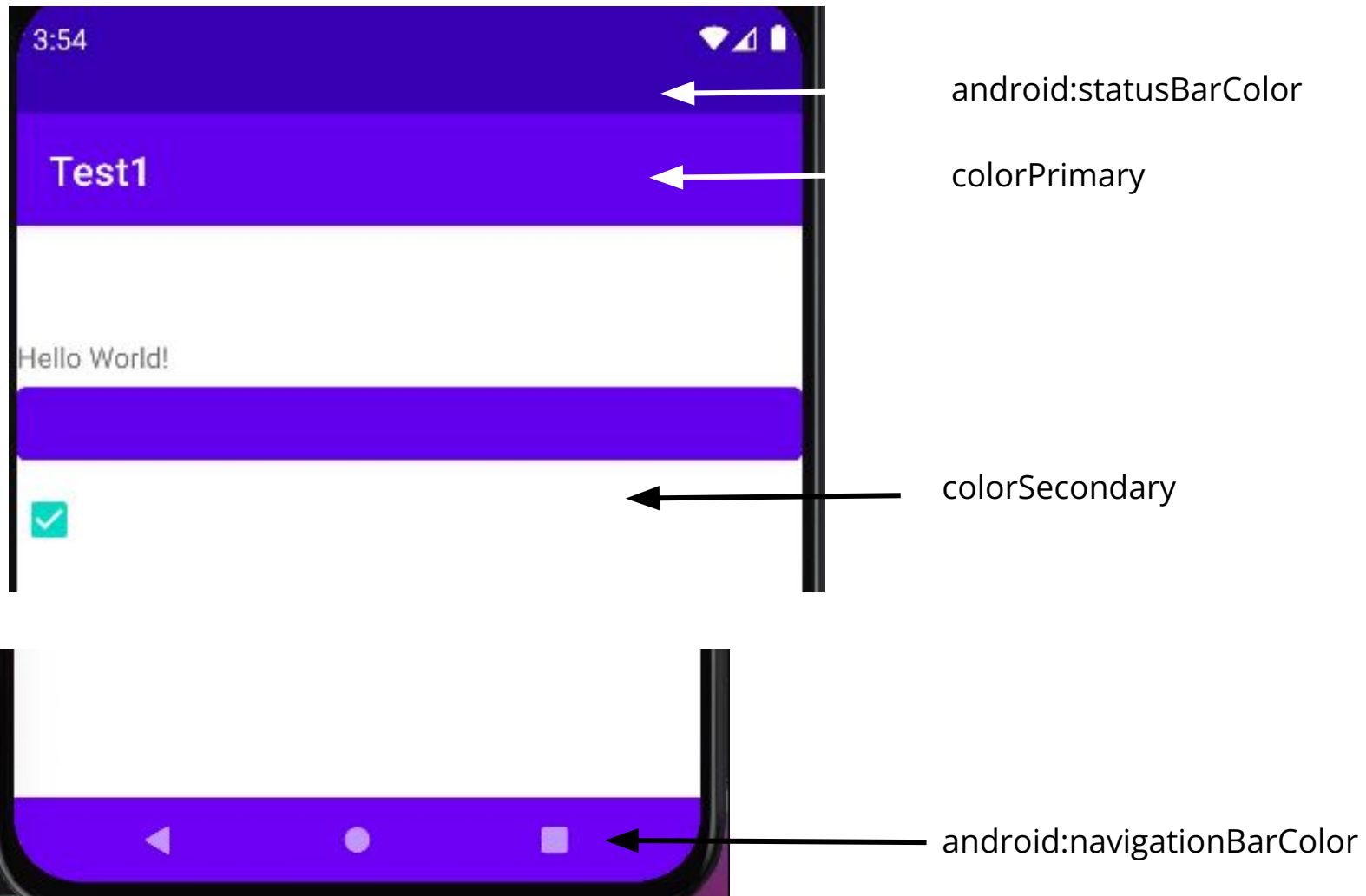


Toolbar

Menu Items

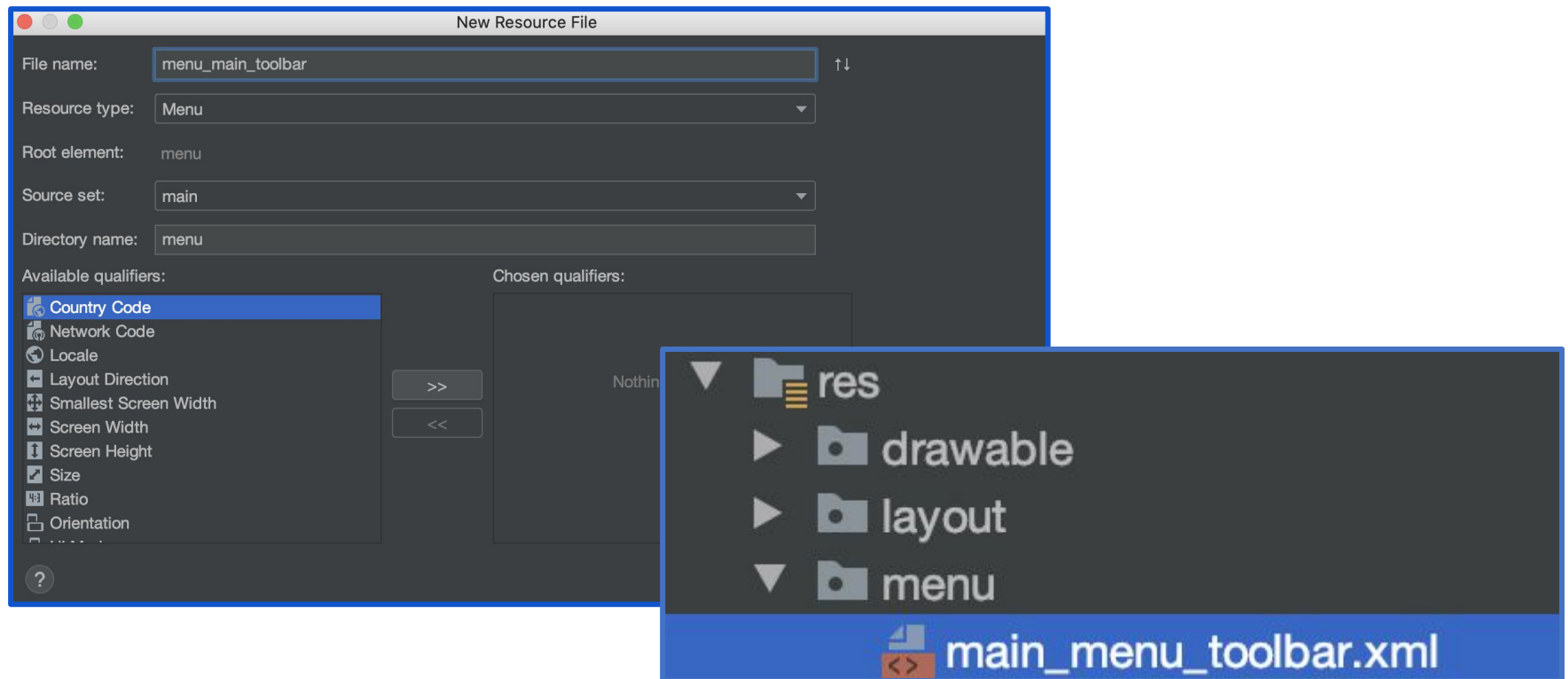
Menu Items
(ocultos)

Equivalencias del archivo **themes.xml**



Cómo Añadir Menultems

Ahora es momento de asignar los elementos del toolbar, para ello creamos un **menu**. En la carpeta **res**, añadimos un nuevo recurso de tipo **menu**.



Cómo Añadir Menultems

En el menú xml, tendremos “items” que serán los elementos del menú con las siguientes opciones ([más info](#)):

`android:id`

El ID de un recurso que es exclusivo del elemento y permite que la aplicación lo reconozca cuando el usuario lo selecciona.

`android:icon`

Referencia a un elemento de diseño para usar como el ícono del elemento.

`android:title`

Referencia a una string para usar como el título del elemento.

`android:showAsAction`

Especifica cuándo y cómo el elemento debe aparecer como un elemento de acción en la barra de app.

Cómo Añadir MenúItems

Nosotros crearemos el siguiente menú, con otro menú anidado:

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto">
  <item
    android:id="@+id/action_chat"
    android:icon="@drawable/ic_chat_black_24dp"
    android:title="Chat"
    app:showAsAction="ifRoom" />

  <item
    android:id="@+id/action_info"
    android:icon="@drawable/ic_chat_black_24dp"
    android:title="Más información"
    app:showAsAction="collapseActionView" />

  <item
    android:id="@+id/text_color"
    android:title="Ajustes"
    app:showAsAction="ifRoom"
    android:icon="@drawable/ic_library_add_black_24dp"
    >
    <menu>
      <item
        android:id="@+id/other"
        android:title="Otro"
        android:icon="@drawable/ic_library_add_black_24dp"
        />
      <item
        android:id="@+id/number"
        android:title="Numero"
        android:icon="@drawable/ic_chat_black_24dp"
        />
      <item
        android:id="@+id/camera"
        android:title="Camara"
        android:icon="@drawable/ic_chat_black_24dp"
        />
    </menu>
  </item>
</menu>
```

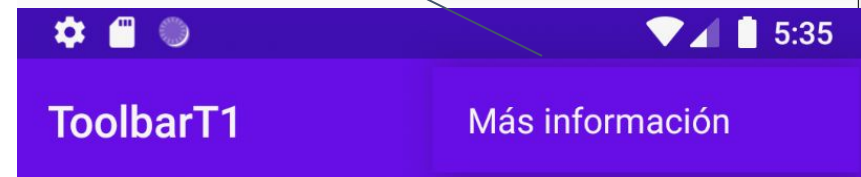
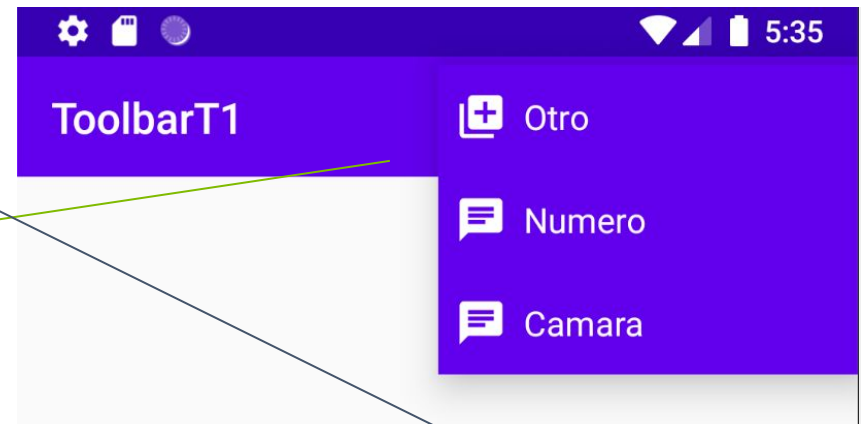

Cómo Añadir MenúItems

Nosotros crearemos el siguiente menú, con otro menú anidado:

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item
        android:id="@+id/action_chat"
        android:icon="@drawable/ic_chat_black_24dp"
        android:title="Chat"
        app:showAsAction="ifRoom" />

    <item
        android:id="@+id/action_info"
        android:icon="@drawable/ic_chat_black_24dp"
        android:title="Más información"
        app:showAsAction="collapseActionView" />

    <item
        android:id="@+id/text_color"
        android:title="Ajustes"
        app:showAsAction="ifRoom"
        android:icon="@drawable/ic_library_add_black_24dp"
        >
        <menu>
            <item
                android:id="@+id/other"
                android:title="Otro"
                android:icon="@drawable/ic_library_add_black_24dp"
                />
            <item
                android:id="@+id/number"
                android:title="Numero"
                android:icon="@drawable/ic_chat_black_24dp"
                />
            <item
                android:id="@+id/camera"
                android:title="Camara"
                android:icon="@drawable/ic_chat_black_24dp"
                />
        </menu>
    </item>
</menu>
```



Cómo Añadir MenuItems

Ahora hay que instanciar por programación nuestro toolbar. Para ello, en el activity hacemos lo siguiente:

```
class MainActivity : AppCompatActivity() {  
    val tag = "MainActivity"  
  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_main)  
    }  
  
    override fun onCreateOptionsMenu(menu: Menu?): Boolean {  
        menuInflater.inflate(R.menu.main_menu_toolbar, menu)  
        return super.onCreateOptionsMenu(menu)  
    }  
  
    override fun onOptionsItemSelected(item: MenuItem): Boolean {  
        when (item.itemId) {  
            R.id.action_chat -> chat()  
            R.id.action_info -> info()  
        }  
        return super.onOptionsItemSelected(item)  
    }  
  
    private fun info() {  
        Log.d(tag, "Info Clicked")  
    }  
  
    private fun chat() {  
        Log.d(tag, "chat Clicked")  
    }  
}
```


Custom Toolbar

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <com.google.android.material.appbar.MaterialToolbar
        android:id="@+id/toolbar"
        style="@style/Toolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/purple_500"
        app:theme="@style/Toolbar" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:layout_gravity="center"
        android:text="Hello World!" />

</RelativeLayout>
```

themes.xml

```
<resources>

  <style name="Theme.Test1" parent="Theme.MaterialComponents.DayNight.NoActionBar">
    ...
  </style>

  <style name="Toolbar">
    <item name="android:textColorPrimary">@color/white</item>
    <item name="actionMenuTextColor">@color/white</item>
    <item name="android:textColorSecondary">@color/white</item>
    <item name="colorSurface">@color/purple_500</item>
  </style>

</resources>
```

cambiar a NoActionBar

Añadir este style

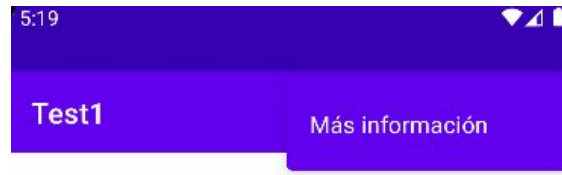
MainActivity

```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
    setSupportActionBar(findViewById(R.id.toolbar))
}
```

Importante el
setSupportActionBar



Hello World!



Hello World!

**¿Y si queremos
cambiar el
fondo de los
menus?**

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <com.google.android.material.appbar.MaterialToolbar
        android:id="@+id/toolbar"
        app:theme="@style/Toolbar"
        style="@style/Toolbar"
        android:background="@color/purple_500"
        app:popupTheme="@style/Theme.AppCompat.Light"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:layout_gravity="center"
        android:text="Hello World!" />

</RelativeLayout>
```

popupTheme



Gracias.