



MaterialToolBar

Crear un proyecto nuevo, en el que para usar MaterialToolBar necesitas tener la dependencia de Material Components en tu build.gradle:

```
dependencies {  
    implementation("com.google.android.material:material:1.12.0")  
}
```

Sobre este proyecto:

- Añadir un MaterialToolBar al layout y configurar en el Activity, el MaterialToolBar como actionBar

```
//configurar MaterialToolbar como actionBar  
val toolbar: MaterialToolbar = binding.toolbar  
setSupportActionBar(toolbar)
```

- Crear 3 logos vectorizados, uno para el menú y los otros para las opciones de búsqueda y de información.
- Personalizar MaterialToolbar usando varios atributos XML y métodos en tu código Kotlin:
 - **Título:** app:title en XML o toolbar.title = "Nuevo título" en Kotlin.
 - **Subtítulo:** app:subtitle en XML o toolbar.subtitle = "Subtítulo" en Kotlin.
 - **Color de fondo:** android:background en XML o toolbar.setBackgroundColor(Color.RED) en Kotlin.
 - **Icono de navegación:** app:navigationIcon en XML o toolbar.navigationIcon = ContextCompat.getDrawable(this, R.drawable.ic_menu) en Kotlin.
- Crear fichero menú.xml: Vamos a new Android resources File, para hacer el menu. File name: mi_menu; tipo menu;

New Resource File

File name:

Resource type:

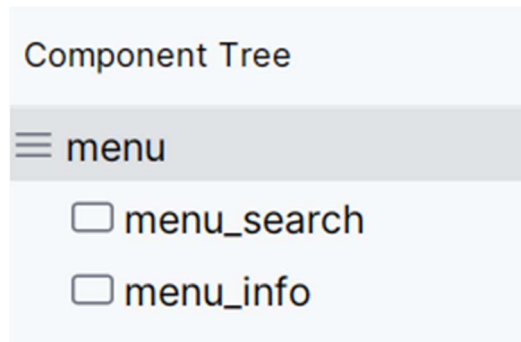
Root element:

Source set:

Directory name:

Al pulsar OK aparece un nuevo fichero xml en res/menú/mi_menu.xml.

Si abrimos este archivo, se puede editar con una vista diseño y añadir 2 ítems al menú. A los ítems, en vista diseño, se les puede asignar los iconos que creamos anteriormente.



En los ítems de menú, usa el atributo `showAsAction` que te permite controlar cómo se muestran los elementos en la Toolbar. Las opciones disponibles son:

- `never`: El elemento de menú siempre se mostrará en el menú desplegable (al pulsar los tres puntos verticales).
- `ifRoom`: El elemento de menú se mostrará en la Toolbar si hay suficiente espacio disponible. De lo contrario, se mostrará en el menú desplegable.
- `withText`: El elemento de menú se mostrará en la Toolbar junto con su texto, si hay suficiente espacio disponible. De lo contrario, se mostrará en el menú desplegable.
- `always`: El elemento de menú siempre se mostrará en la Toolbar, incluso si no hay suficiente espacio disponible. Esto puede hacer que otros elementos de menú se muevan al menú desplegable.
- `collapseActionView`: Esta opción se utiliza para elementos de menú que tienen una vista de acción colapsable, como un campo de búsqueda. Cuando se selecciona el elemento de menú, la vista de acción se expande en la Toolbar. Cuando se colapsa, el elemento de menú vuelve a su tamaño normal.

Puedes combinar estas opciones usando el carácter `|` (OR). El fichero `mi_menu.xml` queda como se muestra a continuación:

```
<item
    android:id="@+id/menu_search"
    android:icon="@drawable/icono_busqueda"
    android:title="@string/menu_busqueda"
    app:showAsAction="ifRoom|withText" />
<item
    android:id="@+id/menu_info"
    android:icon="@drawable/icono_info"
    android:title="@string/menu_info"
    app:showAsAction="never" />
```

- En la pantalla, asignar fichero (mi_menu.xml) al menú, para lo que se debe sobrescribir la función onCreateOptionsMenu

```
override fun onCreateOptionsMenu(menu: Menu): Boolean {
    menuInflater.inflate(R.menu.mi_menu, menu)
    return true
}
```

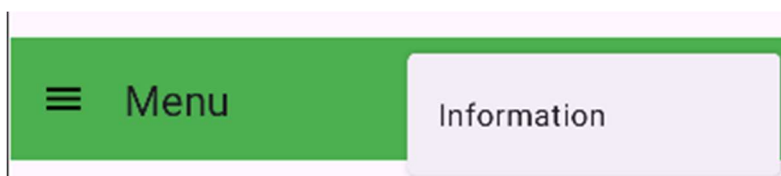
- En la pantalla, debemos manejar los clics en los elementos del menú en onOptionsItemSelected. En cada evento de ítem se creará el código necesario (ir a otro ítem, volver etc)

```
override fun onOptionsItemSelected(item: MenuItem): Boolean {
    return when (item.itemId) {
        R.id.menu_search -> {
            // Acción para el elemento de menú "Búsqueda"
            true
        }
        R.id.menu_info -> {
            // Acción para el elemento de menú "Info"
            true
        }
        else -> false
    }
}
```

La barra de herramientas creada tendrá la siguiente apariencia:



Si se pulsan los 3 puntos, aparece el elemento del menú con la opción showAsAction="never"



Más información en: <https://m2.material.io/components/app-bars-top/android>