

**Software para Dispositivos Móviles**  
Grado en Ingeniería Informática del Software  
Escuela de Ingeniería Informática – Universidad de Oviedo

# Diseño interfaz Android Material Design

Juan Ramón Pérez Pérez  
Departamento de Informática  
[jrpp@uniovi.es](mailto:jrpp@uniovi.es)

# Importancia de la interfaz

- Funcionalidad sin que el usuario pueda acceder a ella no sirve de nada (**usabilidad**)
- Interfaz **intuitiva**, que no haya que “estudiar” cuando vayas a utilizar la app
- Aunque haya equipo de diseño, el **ingeniero en informática** es el que integra, prueba y evalúa el resultado final en la app

# No “reinventar la rueda”

- Sólo en casos muy específicos merece la pena crear controles personalizados
- Lo inteligente es utilizar un **marco de diseño** dado
- Esto además ayudará a **unificar estilos y comportamientos** con el resto de apps
- Y la interacción será **más intuitiva** para el usuario

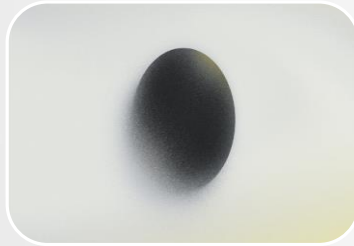
# Material Design

Material 3 is the latest version of Google's open-source design system. Design and build beautiful, usable products with Material 3.

Get started

- Google ha establecido una “filosofía” (sistema) de diseño para Android,
  - que también se puede utilizar en iOS y web
- *Material Design*
  - <https://material.io/>
  - Estas pautas fueron introducidas con Android 5 Lollipop (2014)
  - Android 12 (2022) Material Design 3

# Fundamentos de Material Design



[Accesibilidad](#)



[Diseño de  
contenido](#)



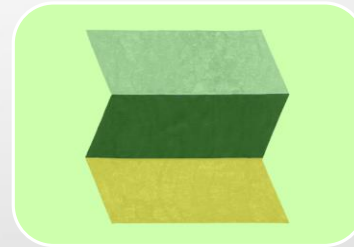
[Personalización](#)



[Design tokens](#)



[Interacción](#)



[Diseño  
adaptativo](#)

<https://material.io/blog/>

# Usando Material Design: Estilos



Color



Elevación



Iconos



Movimiento



Forma



Tipografía

# Usando Material Design: Componentes

## Components

Components are interactive building blocks for creating a user interface. They can be organized into categories based on their purpose: Action, containment, communication, navigation, selection, and text input.

# Diseño de la interfaz con Material Design

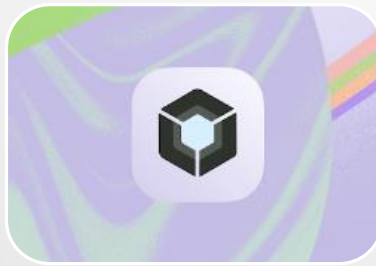
- [Figma](#) es una herramienta muy utilizada por diseñadores web y en general diseñadores de interfaces
  - Herramienta online
  - Edición gráfica y prototipado
  - Colaborativa
- Figma ya tiene una biblioteca con todos los componentes en el [M3 design kit](#)



# Desarrollando con Material Design



[MDC - Android](#)



[Jetpack  
Compose](#)



[Flutter](#)



[Web](#)

# Desarrollo de interfaces mediante layouts vs Jetpack Compose

- XML vs Programación declarativa
- Views vs Composables
  - View & ViewGroup, exponen funciones para cambiar estado → problemas
  - Composables, patrón *observer*, el componente cambia cuando cambia la variable a la que está ligado

# Ejemplo de definición de interfaz simple con Jetpack Compose

```
// example button with text and click Compose

@Composable
fun Button(text: String, onClick: () -> Unit) {
    Button(onClick = onClick) {
        Text(text)
    }
}

fun main() {
    Window(onCloseRequest = ::exitApplication) {
        Button(text = "Pulsame") {
            println("Botón pulsado")
        }
    }
}
```

# Cómo accedo a estas posibilidades de diseño desde Android Studio

- Plantillas
  - Al crear un nuevo proyecto
  - Al crear una nueva activity
- Diseñador de layouts
  - Versión gráfica
  - Versión XML

# Trabajo a realizar sobre la interfaz del proyecto

- Con el planteamiento dado anteriormente y las referencias a recursos en la web,
- Hacer **diseño detallado de la interfaz** de vuestro proyecto
  - **Planteando diseño, componentes y comportamiento general de las distintas activities** necesarias para permitir al usuario utilizar de forma cómoda la funcionalidad planteada en la presentación inicial.
  - Crear bocetos para 3 pantallas y de forma opcional para sus variantes: vertical / apaisado (si hay diferencias) los distintos componentes de cada pestaña en un TabLayout o equivalente).
  - También es importante plantear la navegación entre activities o vistas dentro de una misma activity