



El futuro digital
es de todos

MinTIC



Universidad
Pontificia
Bolivariana

Vigilada Mineducación

«Misión
TIC 2022»

Introducción a las aplicaciones móviles con Android

» www.upb.edu.co/es/mision-tic





El futuro digital
es de todos

MinTIC



Universidad
Pontificia
Bolivariana

Vigilada Mineducación

‘Misión
TIC2022’

Agenda

Actividades

Layouts

Datos globales

Navegación



www.upb.edu.co/es/mision-tic



El futuro digital
es de todos

MinTIC



Actividades

Una actividad es cada vista que muestra una aplicación en Android, por lo general la transición e interacción entre actividades es administrada por la aplicación.

El sistema operativo se encarga de controlar la ejecución y optimización de las vistas de tal forma que cuando se navega en una aplicación el cambio entre actividades se realiza de una forma rápida y sencilla.

Por lo general, no existe persistencia de datos entre actividades, pero es posible enviar datos de una actividad a otra.



El futuro digital
es de todos

MinTIC



Layout

Es la estructura de la actividad, es un archivo que se encarga de definir el conjunto de contenedores y los elementos que definen el contenido de cada sección de la actividad.

Es un archivo XML, que permite definir la interfaz gráfica de una actividad, por lo general se debe crear un layout por actividad.

Este archivo tiene dos vistas, la vista de texto y la vista de diseño, se recomienda comenzar por la vista de texto e intercambiar entre vistas para construir la interfaz final.

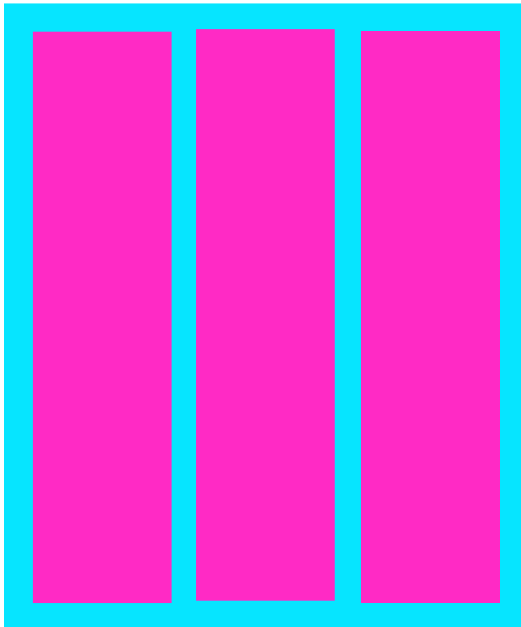
Recuerde que Android Studio ha mejorado bastante, pero todavía tiene algunos bugs, que se solucionan por lo general realizando un rebuild del proyecto.



LinearLayout - Horizontal

Es el más sencillo, de frecuente uso, ordena los elementos uno al lado del otro, teniendo en cuenta principalmente el ancho del elemento.

Gráficamente



XML

```
<LinearLayout  
  android:orientation="horizontal"  
  android:layout_width="match_parent"  
  android:layout_height="wrap_content">
```

<!-- elementos acá -->

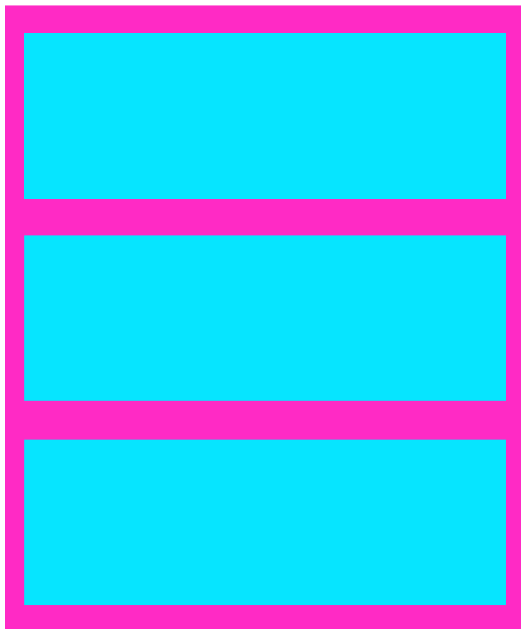
```
</LinearLayout>
```



LinearLayout - Vertical

Es el más sencillo, de frecuente uso, ordena los elementos uno al lado del otro, teniendo en cuenta principalmente el ancho del elemento.

Gráficamente



XML

```
<LinearLayout  
  android:layout_width="match_parent"  
  android:layout_height="match_parent"  
  android:orientation="vertical">
```

<!-- Elementos acá -->

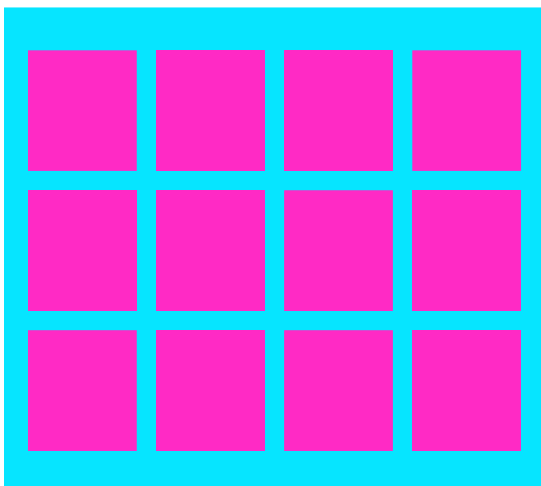
```
</LinearLayout>
```



GridLayout

Es el más sencillo, de frecuente uso, ordena los elementos uno al lado del otro, teniendo en cuenta principalmente el ancho del elemento.

Gráficamente



4 Columnas

3 Filas

XML

```
<GridLayout  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:columnCount="4"  
    android:rowCount="3"  
    android:orientation="horizontal">
```

<!-- Elementos acá -->

```
</GridLayout>
```



El futuro digital
es de todos

MinTIC



Universidad
Pontificia
Bolivariana

Vigilada Mineducación



Datos globales

La clave es extender de [Application](#) y luego crear los miembros datos necesarios para representar los datos que se quieren mantener entre actividades.

```
public class Datos extends Application {  
  
    private String dato;  
    public String getDato() {  
        return dato;  
    }  
  
    public void setDato(String dato) {  
        this.dato= dato;  
    }  
  
}
```



www.upb.edu.co/es/mision-tic



El futuro digital
es de todos

MinTIC



Datos globales

Finalmente en el manifiesto de la aplicación, se debe referenciar la clase que contiene los datos, como se muestra a continuación.

```
<application
    android:name=".util.Datos"
    android:theme="@android:style/Theme.Black.NoTitleBar.Fullscreen">
```

```
<!-- Actividades acá -->
```

```
</application>
```



Obtiene el dato desde cualquier actividad

```
((Datos)this.getApplication()).getDato()
```

Adicionalmente se incluye una línea que permite remover la barra de título de una aplicación.



Navegación - automática

La navegación automática, se usa cuando se requiere mostrar una actividad por un tiempo limitado y que luego se produzca un cambio automático sin necesidad de la interacción del usuario.

```
Timer t = new Timer();
t.schedule(new TimerTask() {
    public void run() {
        /*
        Código que se ejecuta al pasar el tiempo
        que se define en milisegundos
        */
        Clase.this.finish();//Termina una actividad
    }
}, tiempo_milisegundos);
```



Navegación - manual

Requiere de interacción del usuario, permite cambiar entre actividades y enviar datos, es posible cambiar a una nueva actividad sin destruir la actividad actual, de tal forma que sea posible volver a una actividad anterior.

ClaseActual

```
Intent intent;  
intent = new Intent(ClaseActual.this, SiguienteClase.class);  
intent.putExtra("dato1", "Prueba");  
Toast.makeText(getBaseContext(), "Prueba esto",  
Toast.LENGTH_LONG).show();  
startActivity(intent);
```

SiguienteClase

```
getIntent().getStringExtra("dato1");
```



El futuro digital
es de todos

MinTIC



Universidad
Pontificia
Bolivariana

Vigilada Mineducación

‘Misión
TIC2022’

¡Muchas gracias!