

## Guías Prácticas Nº 2

## Asignatura

# DESARROLLO DE APLICACIONES MÓVILES

Docente : Ing. Yan Casayco Contreras

Facultad : Ingeniería

Sección: \_\_\_\_\_ Fecha: \_\_\_\_\_

# 1. Introducción al IDE y estructura de un proyecto Explicación:

- Android Studio es el IDE oficial para desarrollo Android.
- Estructura de carpetas principales:
  - app/java/ → Código fuente Java (clases, activities, lógica).
  - o app/res/layout/ → Archivos XML para interfaces gráficas.
  - o app/res/values/ → Strings, colores, estilos.
  - AndroidManifest.xml → Configuración de actividades, permisos.
  - Gradle Scripts → Configuración del proyecto.

#### Actividad:

- 1. Crear un nuevo proyecto Empty Activity (nombre: MiPrimerApp).
- 2. Explorar cada carpeta y archivo inicial.
- Identificar dónde está la clase MainActivity. java y el archivo activity\_main.xml.

# 2. Clases Java y elementos XML

## Explicación:

- Una Activity es una pantalla de la aplicación. Ejemplo: MainActivity.
- Cada Activity se asocia a un layout XML que define su interfaz.
- XML → Define vistas (botones, textos, imágenes).
- Java → Controla la lógica (qué pasa cuando tocas un botón).

#### Ejemplo:



#### activity\_main.xml

MainActivity.java



```
package com.example.miapp;
import android.os.Bundle;
import android.widget.Button;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        TextView txtMensaje = findViewById(R.id.txtMensaje);
        Button btnSaludar = findViewById(R.id.btnSaludar);

        btnSaludar.setOnClickListener(v -> txtMensaje.setText("¡Bienvenido a Android Studio!"));
    }
}
```

# 3. Intents explícitos, implícitos y paso de parámetros

## Explicación:

- Intent explícito: Abre una Activity propia del proyecto.
- Intent implícito: Abre una app externa (ej. navegador, teléfono).
- Paso de parámetros: Enviar datos de una Activity a otra.

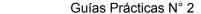
### **Ejemplo:**

- 1. Crear nueva Activity → SegundaActivity.
- 2. Layout con un TextView.

#### MainActivity.java

```
Intent intent = new Intent(MainActivity.this, SegundaActivity.class);
intent.putExtra("usuario", "Juan Pérez");
startActivity(intent);
```

#### SegundaActivity.java





```
Bundle extras = getIntent().getExtras();
if (extras != null) {
    String nombre = extras.getString("usuario");
    TextView txt = findViewById(R.id.txtSaludo);
    txt.setText("Hola " + nombre);
}
```

Intent implícito (abrir navegador):

```
Intent intent = new Intent(Intent.ACTION_VIEW);
intent.setData(Uri.parse("https://www.google.com"));
startActivity(intent);
```

# 4. Layouts, vistas y widgets

## Explicación:

- Layouts comunes:
  - $\circ$  LinearLayout  $\rightarrow$  Elementos en fila o columna.
  - $\hspace{1cm} \circ \hspace{1cm} \textbf{RelativeLayout} \rightarrow \textbf{Elementos relativos entre s\'i}. \\$
  - $\circ$  ConstraintLayout  $\rightarrow$  Más flexible, recomendado.
- Widgets comunes: TextView, EditText, Button, ImageView, CheckBox.

#### **Actividad:**

- En activity\_main.xml agregar:
  - EditText para nombre.
  - o Button que salude al nombre ingresado.
  - ImageView con un ícono.

Ejemplo completo en activity\_main.xml (mezcla de vistas y widgets):





Código en MainActivity.java:



```
package com.example.miapp;
import android.os.Bundle;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        EditText edtNombre = findViewById(R.id.edtNombre);
        Button btnSaludar = findViewById(R.id.btnSaludar);
        TextView txtSaludo = findViewById(R.id.txtSaludo);
        ImageView imgLogo = findViewById(R.id.imgLogo);
        CheckBox chkRecordar = findViewById(R.id.chkRecordar);
```

Con esto, ya usan: EditText, Button, TextView, ImageView, CheckBox.



# 5. Diseño de interfaces

## **Explicación:**

- Uso de paleta de componentes en Android Studio.
- Buenas prácticas:
  - Usar dp para tamaños, sp para texto.
  - Definir colores en colors.xml.
  - Definir textos en strings.xml.

Aquí se busca que los estudiantes mejoren la presentación usando buenas prácticas:

## Colores y textos en recursos

En res/values/colors.xml:

En res/values/strings.xml:

```
<resources>
     <string name="app_name">Mi Primer App</string>
          <string name="hint_nombre">Escribe tu nombre</string>
          <string name="btn_saludar">Saludar</string>
          <string name="msg_saludo">Aquí aparecerá el saludo</string>
          </resources>
```

Mejora de activity\_main.xml



```
<Button
   android:id="@+id/btnSaludar"
   android:layout width="match parent"
   android:layout_height="wrap_content"
   android:text="@string/btn_saludar"
   android:backgroundTint="@color/colorBoton"
   android:textColor="@android:color/white"
   android:layout_marginTop="12dp"/>
<TextView
   android:id="@+id/txtSaludo"
   android:layout width="match parent"
   android:layout height="wrap content"
   android:text="@string/msg_saludo"
   android:gravity="center"
   android:layout_marginTop="20dp"
   android:textColor="@color/colorPrimario"/>
```

