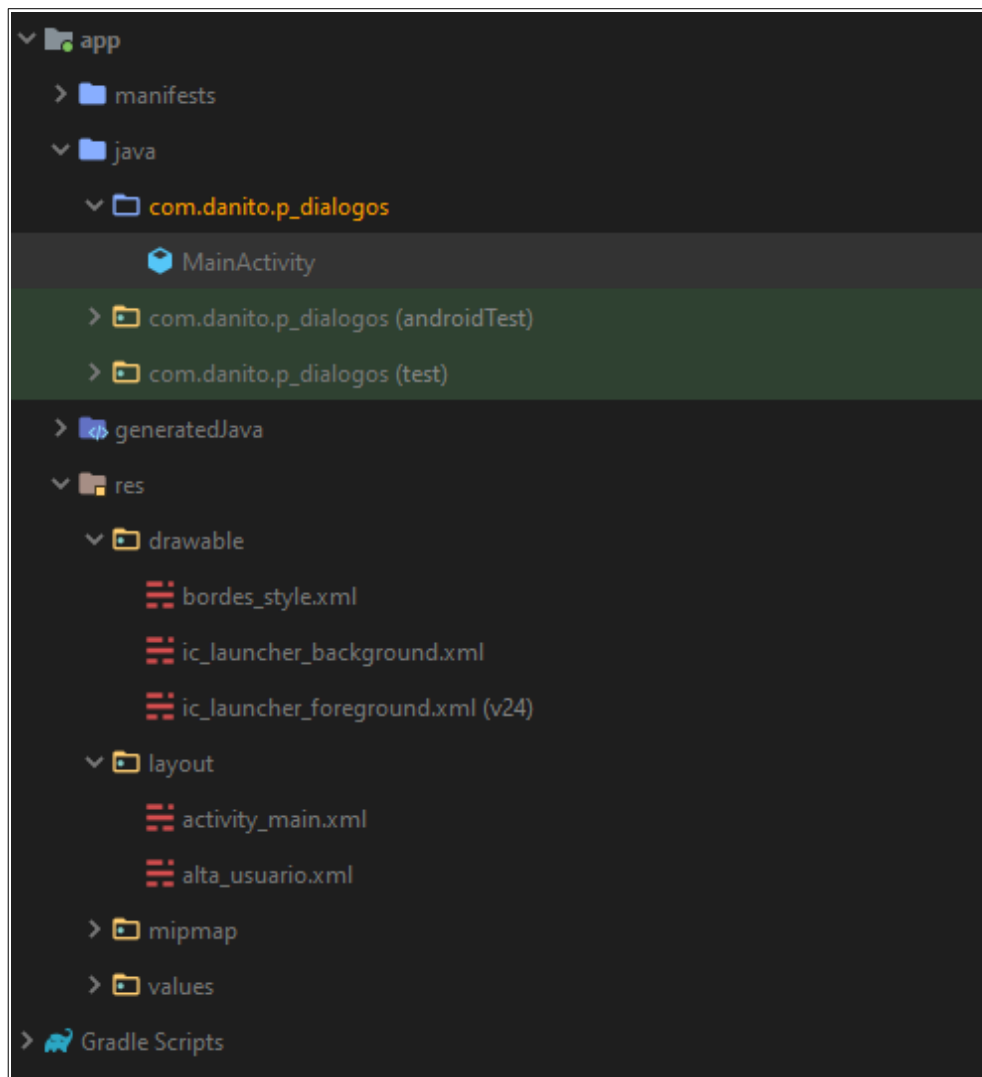


Documentación

Proyecto Diálogos

1) Estructura del proyecto y archivos.



2) Desarrollo: activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    tools:context=".MainActivity"
    android:orientation="vertical"
    android:padding="32dp"
    android:gravity="center_vertical">
```

```
<Button
    android:id="@+id/botonIniciar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Iniciar aplicación"
    style="@android:style/Widget.Material.Button.Colored"/>

</LinearLayout>
```

El activity_main.xml consta de un botón en el centro que será el que nos permita “iniciar” la aplicación.

3) Desarrollo: alta_usuario.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="32dp">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:gravity="bottom"
        android:text="Necesitas tener una cuenta creada para iniciar la aplicación."
        android:textAlignment="center"
        android:textAppearance="@style/TextAppearance.AppCompat.Medium" />

    <Button
        android:id="@+id/botonCrear"
        android:layout_marginTop="8dp"
        android:layout_marginBottom="8dp"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:text="Crear cuenta"
        android:textAlignment="center"
        style="@style/Widget.AppCompat.Button.Colored"/>

    <View
        android:id="@+id/divider"
        android:layout_width="match_parent"
        android:layout_height="2dp"
        android:background="@color/colorPrimary" />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_marginTop="8dp"
        android:layout_marginBottom="8dp"
        android:layout_weight="1"
        android:hint="Nombre"
        android:background="@drawable/bordes_style"
        android:paddingLeft="16dp"/>

    <EditText
        style="@style/Widget.AppCompat.AutoCompleteTextView"
        android:layout_width="match_parent"
        android:layout_height="0dp">
```

```
        android:layout_marginTop="8dp"
        android:layout_marginBottom="8dp"
        android:layout_weight="1"
        android:hint="Contraseña"
        android:background="@drawable/bordes_style"
        android:paddingLeft="16dp"/>

<CheckBox
    android:layout_marginTop="8dp"
    android:layout_marginBottom="8dp"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:text="Recordarme"/>

<TextView
    android:layout_marginTop="8dp"
    android:layout_marginBottom="8dp"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:text="¿Olvidaste tu contraseña?"
    android:gravity="center_vertical"/>

<Button
    android:id="@+id/botonEntrar"
    android:layout_marginTop="8dp"
    android:layout_marginBottom="8dp"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1"
    android:text="Entrar"
    android:textAlignment="center"
    style="@style/Widget.AppCompat.Button.Colored"/>

</LinearLayout>
```

El desarrollo de esta activity consta de una serie de controles alineados con un `LinearLayout` en orientación vertical. Cada uno de los controles se reparte el espacio a maneras iguales utilizando la propiedad `android:layout_weight="1"` en cada uno de los controles y a la vez asignándoles 0 en la altura. La única particularidad es el background de los botones.

Asignándole a un `EditText` el background `@drawable/bordes_style` consigo que tenga un borde completo en vez de un borde solo por debajo.

`bordes_style` lo he definido creando un nuevo XML y colocándolo en la carpeta de drawables del proyecto.

4) Desarrollo: `bordes_style.xml`

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <stroke
        android:width="1dp"
        android:color="@android:color/black" />
</shape>
```

Es simplemente una línea de color negro y de grosor 1dp.

El elemento `stroke`, es un elemento de diseño que forma un trazo.

5) Desarrollo: MainActivity.java

```
package com.danito.p_dialogos;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button btniciar = findViewById(R.id.botonIniciar);
        btniciar.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                final AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);
                LayoutInflater inflater = MainActivity.this.getLayoutInflater();
                View dialogView = inflater.inflate(R.layout.alta_usuario, null);
                builder.setView(dialogView);
                Button bCrear = dialogView.findViewById(R.id.botonCrear);
                bCrear.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {
                        Toast.makeText(getApplicationContext(), "Se creará la cuenta", Toast.LENGTH_LONG).show();
                    }
                });

                Button bEntrar = dialogView.findViewById(R.id.botonEntrar);
                bEntrar.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {
                        Toast.makeText(getApplicationContext(), "Se iniciará la aplicación", Toast.LENGTH_LONG).show();
                    }
                });
                builder.show();
            }
        });
    }
}
```

Lo primero que hago es identificar el botón y ponerlo en una variable para usarlo después.

Segundo, le coloco un listener anónimo que es donde se generará el diálogo.

Dentro del listener creamos un AlertDialog.Builder y le indicamos el contexto sobre el que aparecerá un diálogo.

Antes de mostrar el diálogo, creo y utilizo un LayoutInflater que me permitirá superponer un layout sobre el layout principal, el objetivo aquí es que aparezca el layout de alta_usuario sobre el activity_main.

Con el inflater, asigno a una variable View el layout que posteriormente aparecerá en el diálogo.

Este paso podría omitirse, sin embargo, es necesario hacerlo ya que queremos controlar los botones del layout generado.

Ahora asigno a variables los botones del layout accediendo a ellos a través de la variable View que antes declaré y les asigno un listener anónimo, en el que simularemos la acción indicando que es lo que ocurre con un Toast.

Finalmente muestro la alerta con `builder.show()`.