

JAVA EN ANDROID

2

Oscar Chávez-Bosquez

<http://chavezbosquez.github.io/diplomado/android.html>

Hoy veremos

1. Desarrollo de Apps
2. Hola, mundo!
3. Login
4. Calculadora

Desarrollo de apps

¿Que es una app?

- Una o más pantallas interactivas
- Escritas en **Java** y **XML**
- Usa el *Android Software Development Kit* (SDK)
- Usa bibliotecas Android y el *Android Application Framework*
- Se ejecuta en la *Android Runtime Virtual machine* (ART)

Retos a considerar

- Múltiples tamaños y resoluciones de pantalla
- **Rendimiento:** apps *responsivas* y fluidas
- **Seguridad:** código fuente y datos del usuario
- **Compatibilidad:** funciona en plataformas anteriores
- **Marketing:** considerar el mercado y a los usuarios

Bloques de construcción de apps

- **Recursos:** *layouts*, imágenes, texto y colores en archivos XML y archivos de medios
- **Componentes:** *activities*, servicios y clases en código Java
- **Manifest:** información acerca de la app para el runtime
- **Build configuration:** versiones de APK en archivos de configuración Gradle

Documentos para desarrolladores



- Introducción:

<https://developer.android.com/guide/index.html>


- Introducción a las *Activities*:

<https://developer.android.com/guide/components/activities/intro-activities>

- Diseño de UIs:

<https://developer.android.com/guide/topics/ui/declaring-layout>

Conceptos preliminares



A Java Swing window titled "Inicio de sesión". It features a dark header bar with the title and standard window controls. The main area has a light gray background. It contains two text input fields: "Usuario" and "Contraseña". Below the fields are two buttons: "Aceptar" and "Cancelar".

JFrame



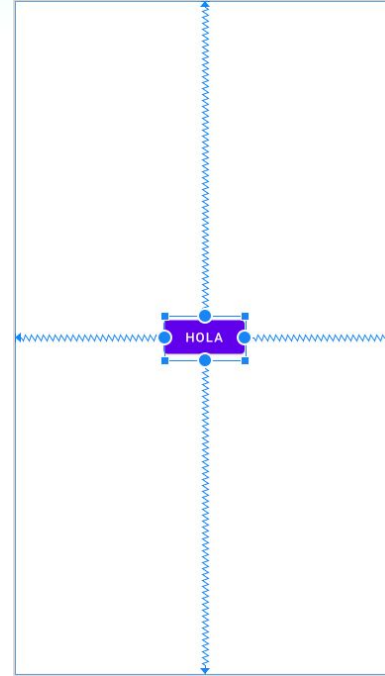
A smartphone screen displaying a login interface. The status bar at the top shows a warning icon, 4G signal, battery level, and time (12:32). The app title "Inicio de sesión" is in a purple header. The main content area is white and contains two text input fields: "Introduce tu correo" and "Contraseña". Below the fields is a large purple button labeled "ACEPTAR". The bottom of the screen shows the standard Android navigation bar with back, home, and search icons.

Activity

Conceptos preliminares



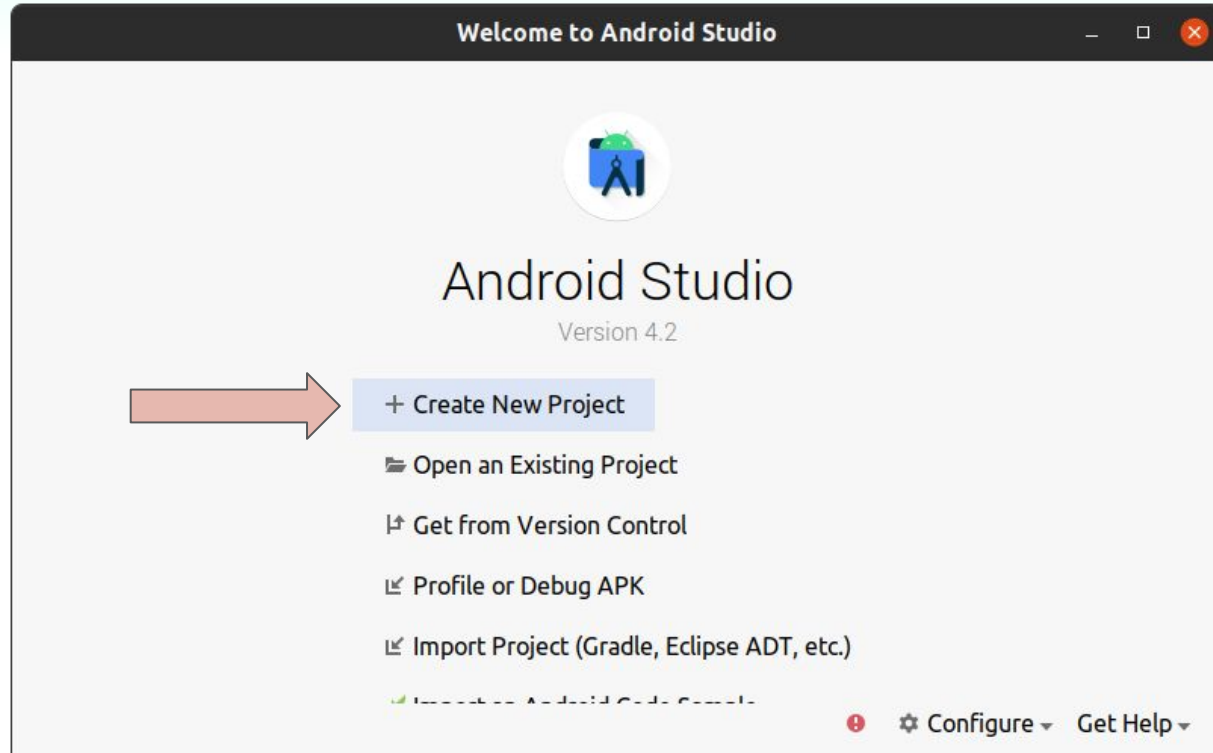
Component



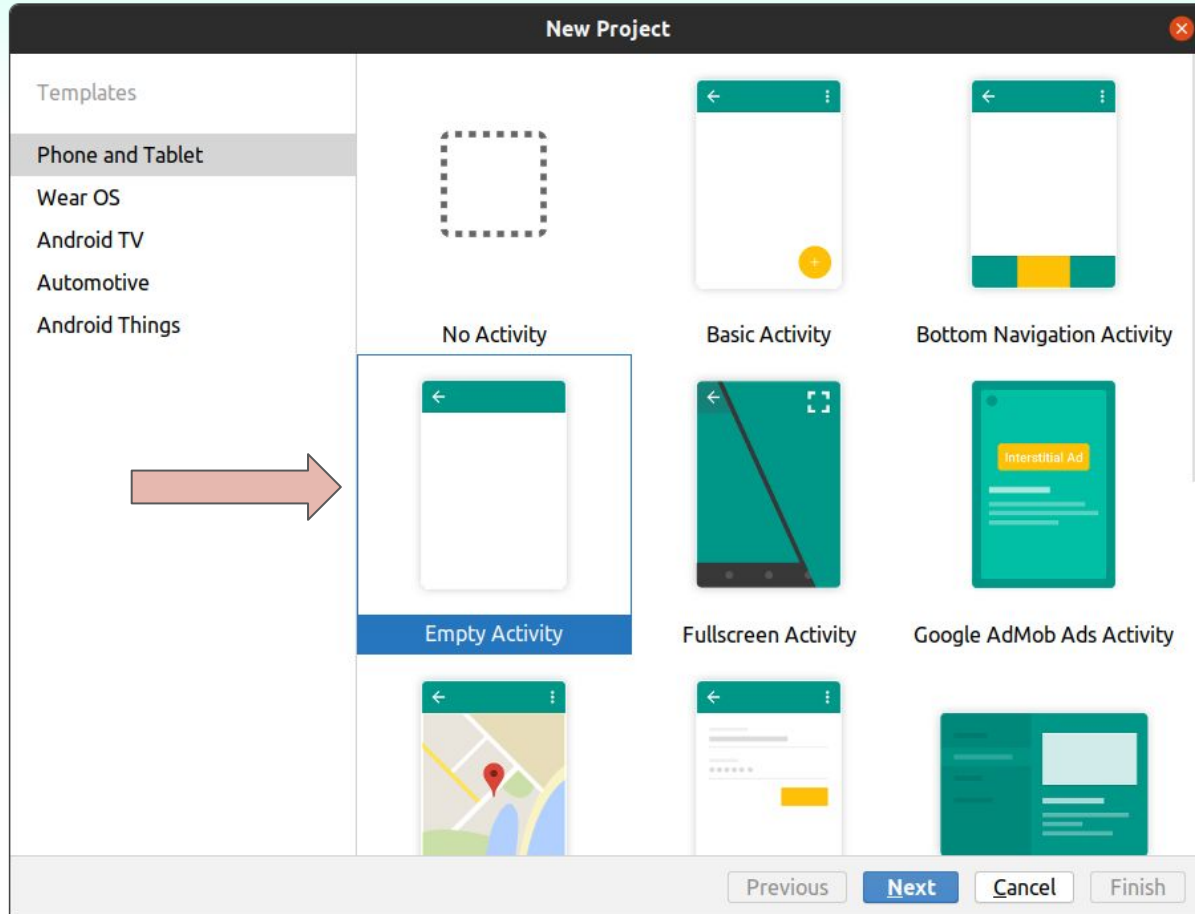
View

Hola, mundo!

Nuevo proyecto



Activity vacía



HolaMundo

New Project

Empty Activity

Creates a new empty activity

Name

HolaMundo

Package name

com.example.holamundo

Save location

/home/chavez/AndroidStudioProjects/HolaMundo

Language

Java

Minimum SDK

API 16: Android 4.1 (Jelly Bean)

Your app will run on approximately **99.8%** of devices.
[Help me choose](#)

☐

Use legacy android.support libraries ?


Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries

Previous

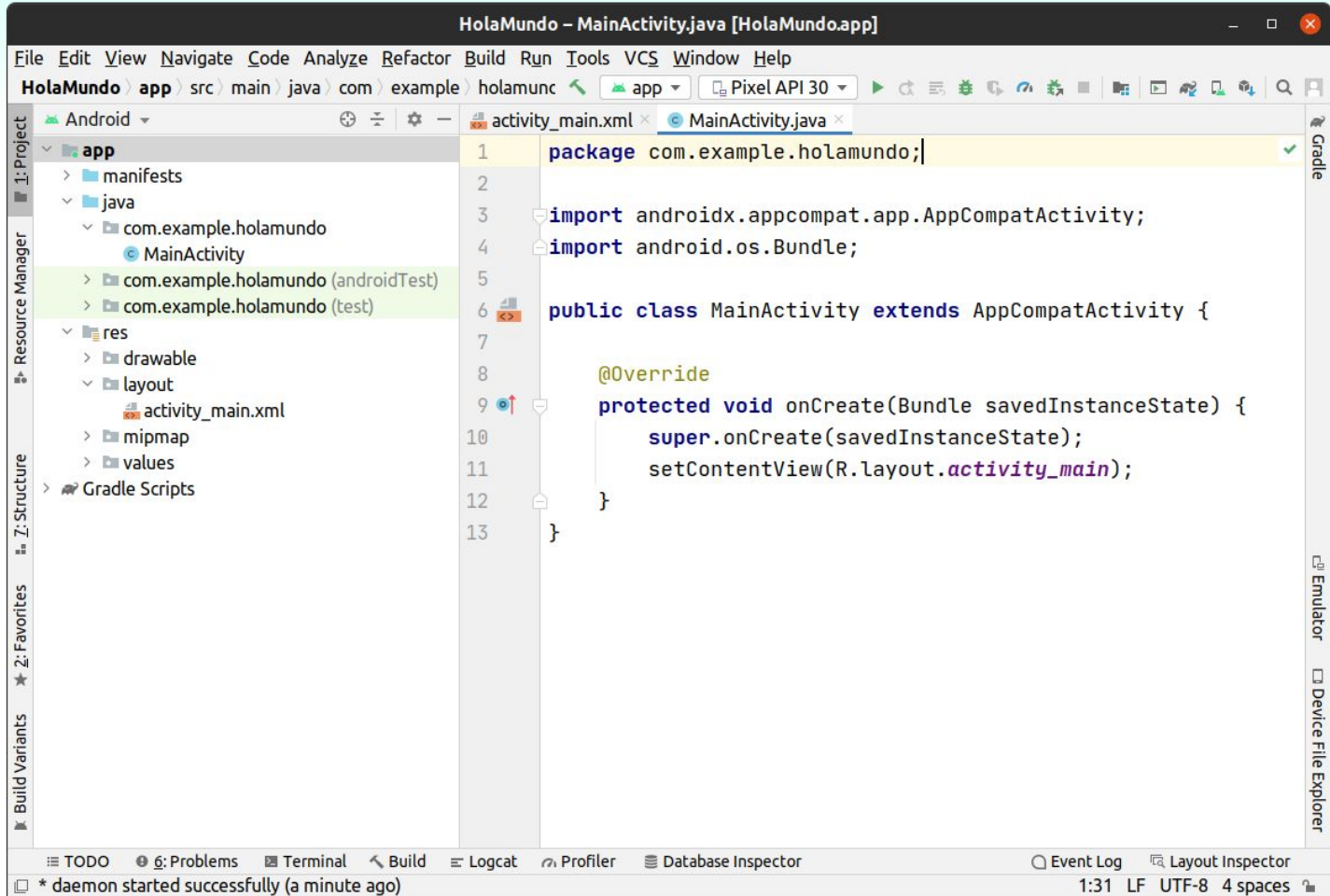
Next

Cancel

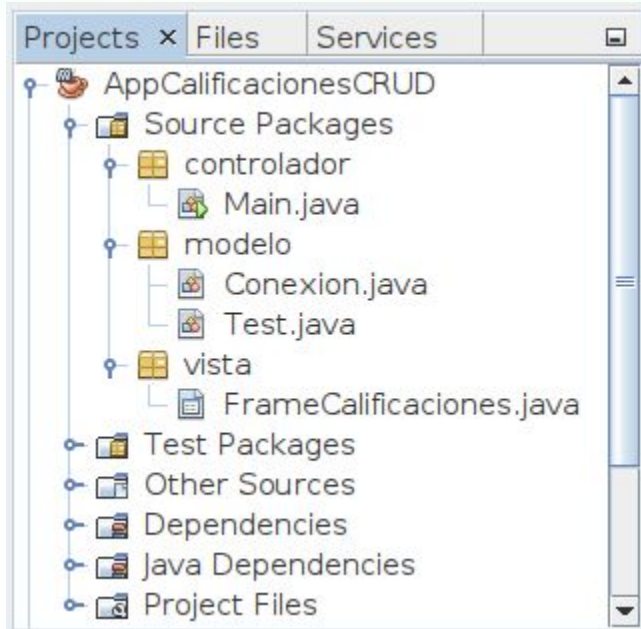
Finish



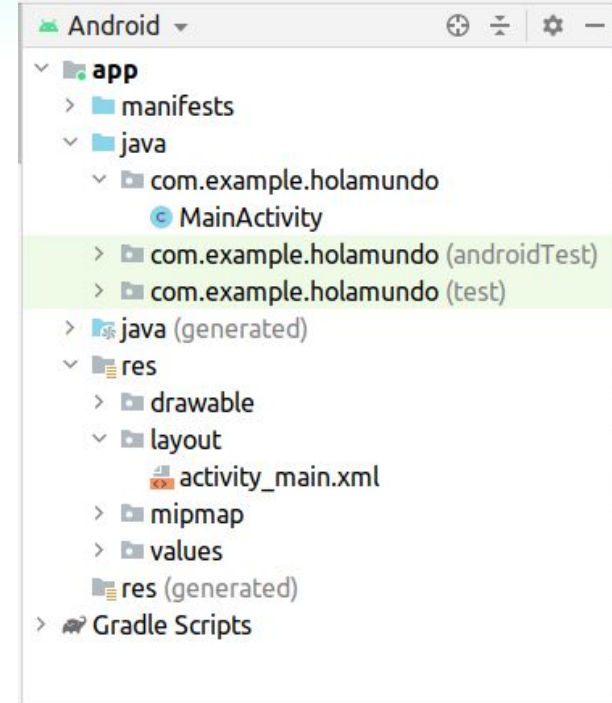
Estructura del proyecto



Analogías con NetBeans

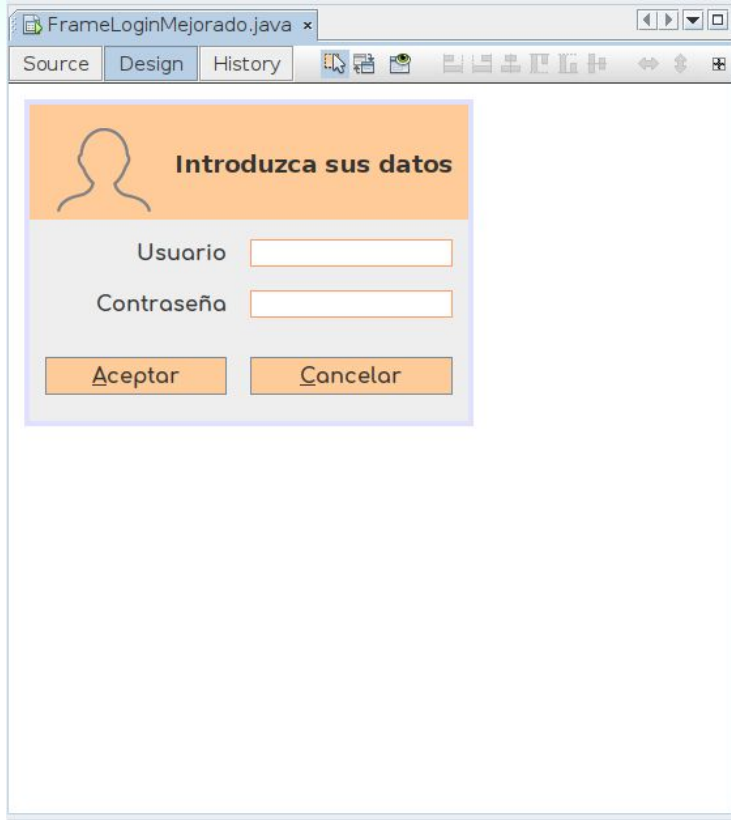


Panel de proyecto

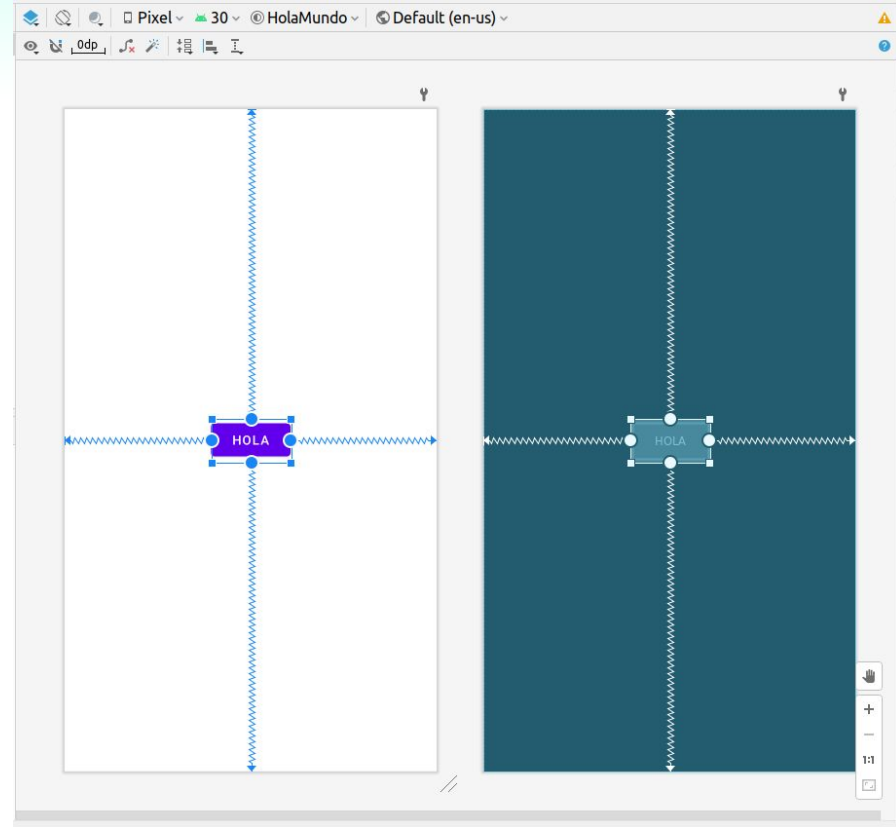


Panel de proyecto

Analogías con NetBeans

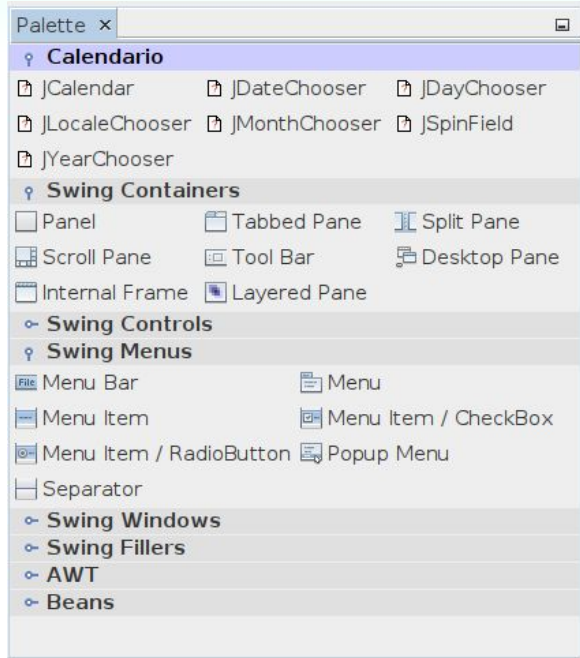


Diseñador de UIs

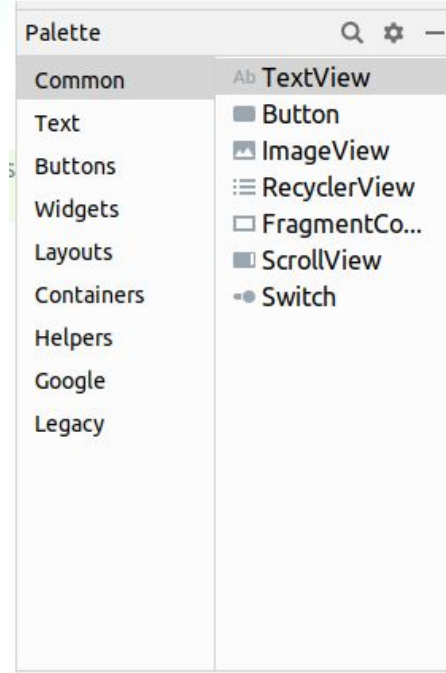


Diseñador de UIs

Analogías con NetBeans

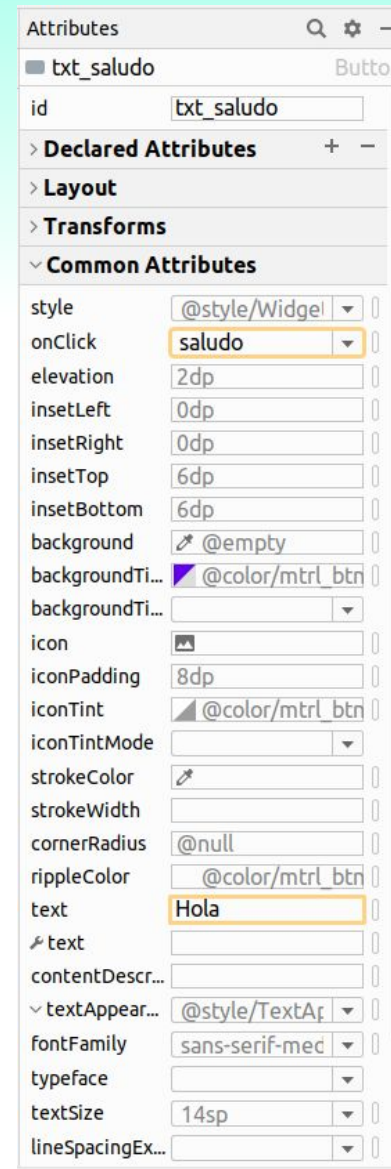
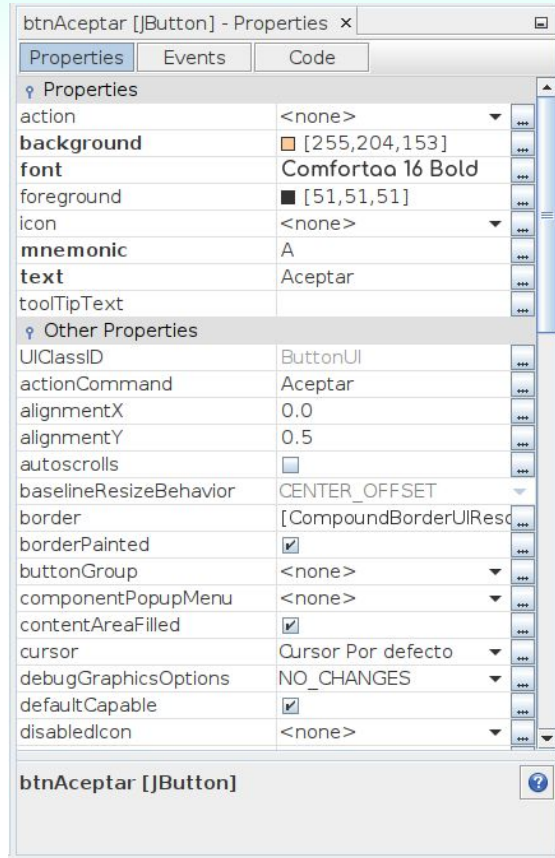


Paleta de
componentes



Paleta de
Vistas

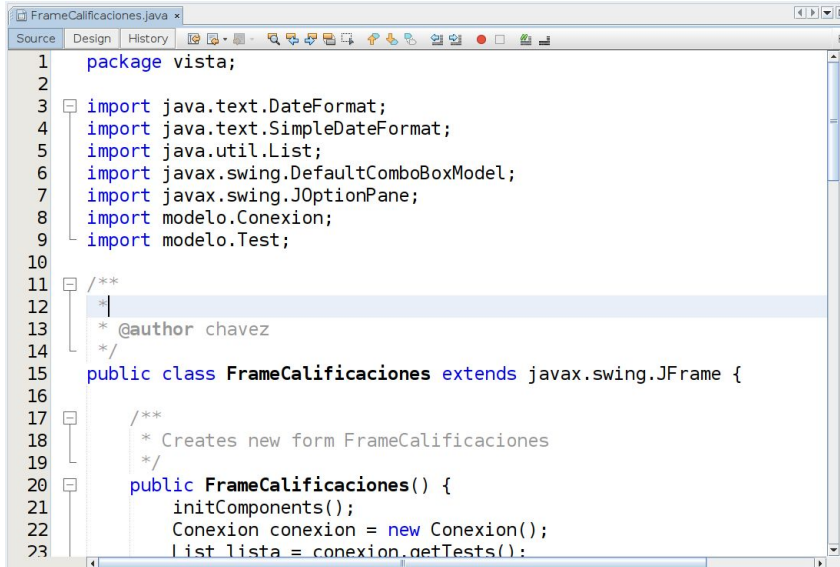
Analogías con NetBeans



Panel de
atributos

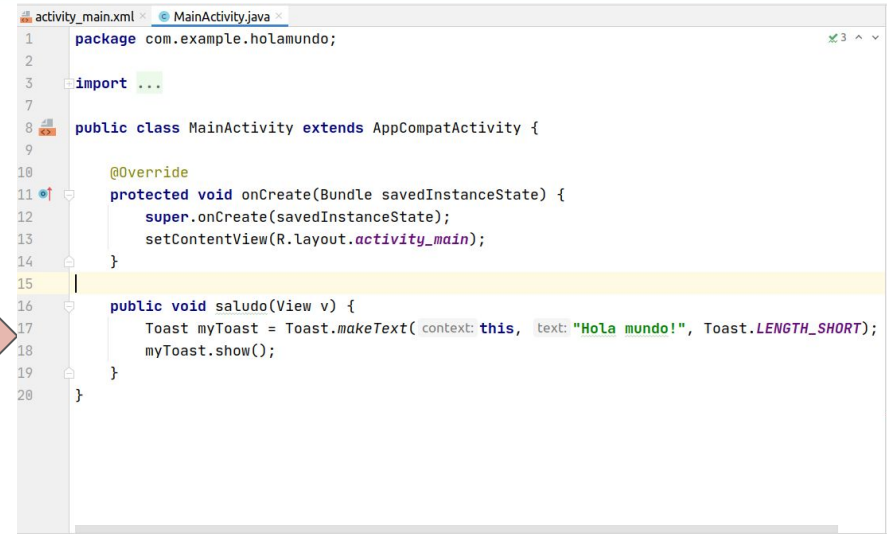
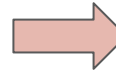
Panel de propiedades

Analogías con NetBeans



```
1 package vista;
2
3 import java.text.DateFormat;
4 import java.text.SimpleDateFormat;
5 import java.util.List;
6 import javax.swing.DefaultComboBoxModel;
7 import javax.swing.JOptionPane;
8 import modelo.Conexion;
9 import modelo.Test;
10
11 /**
12  *
13  * @author chavez
14  */
15 public class FrameCalificaciones extends javax.swing.JFrame {
16
17     /**
18      * Creates new form FrameCalificaciones
19      */
20     public FrameCalificaciones() {
21         initComponents();
22         Conexion conexion = new Conexion();
23         List lista = conexion.getTests();
```

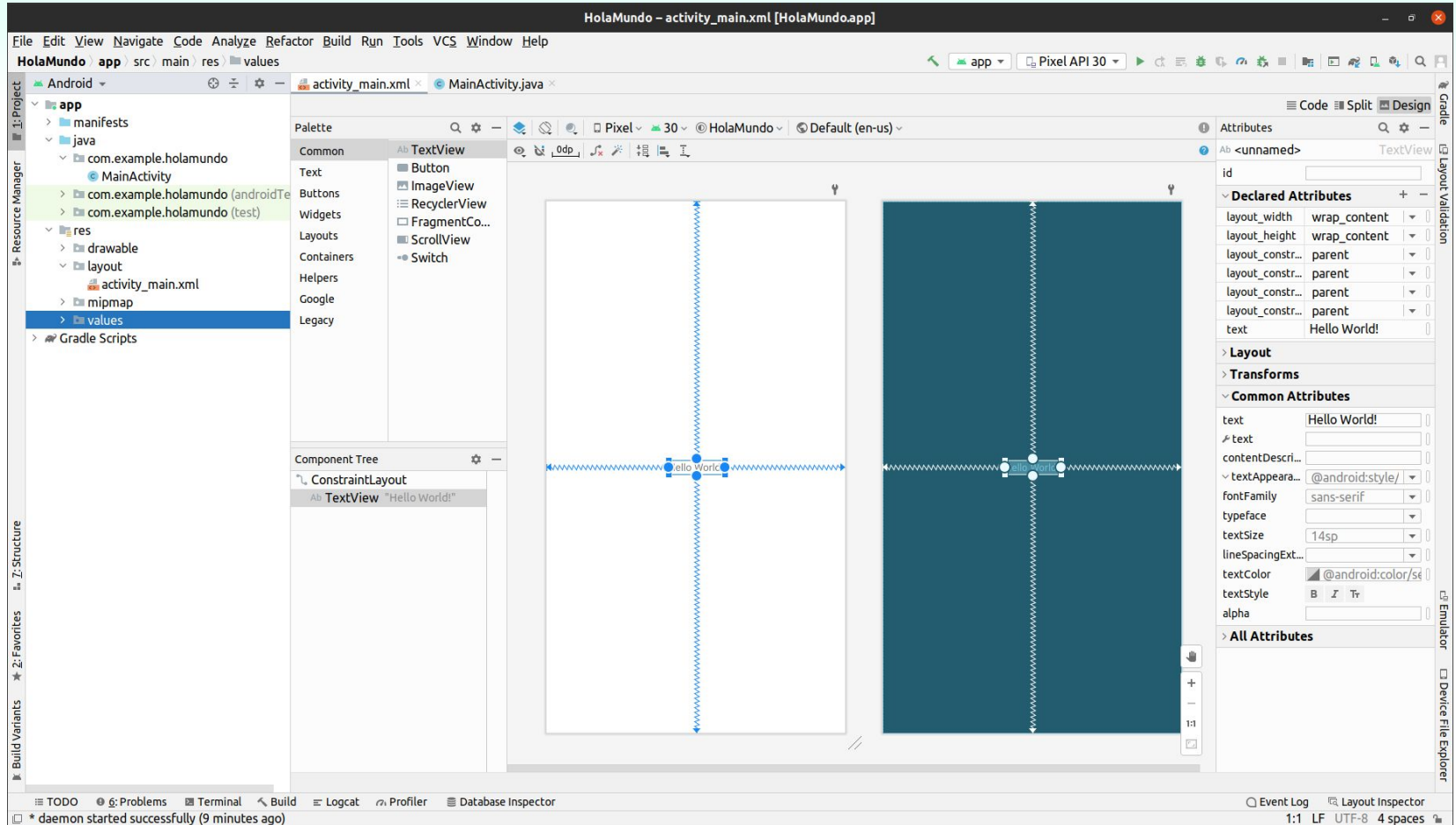
Editor de código



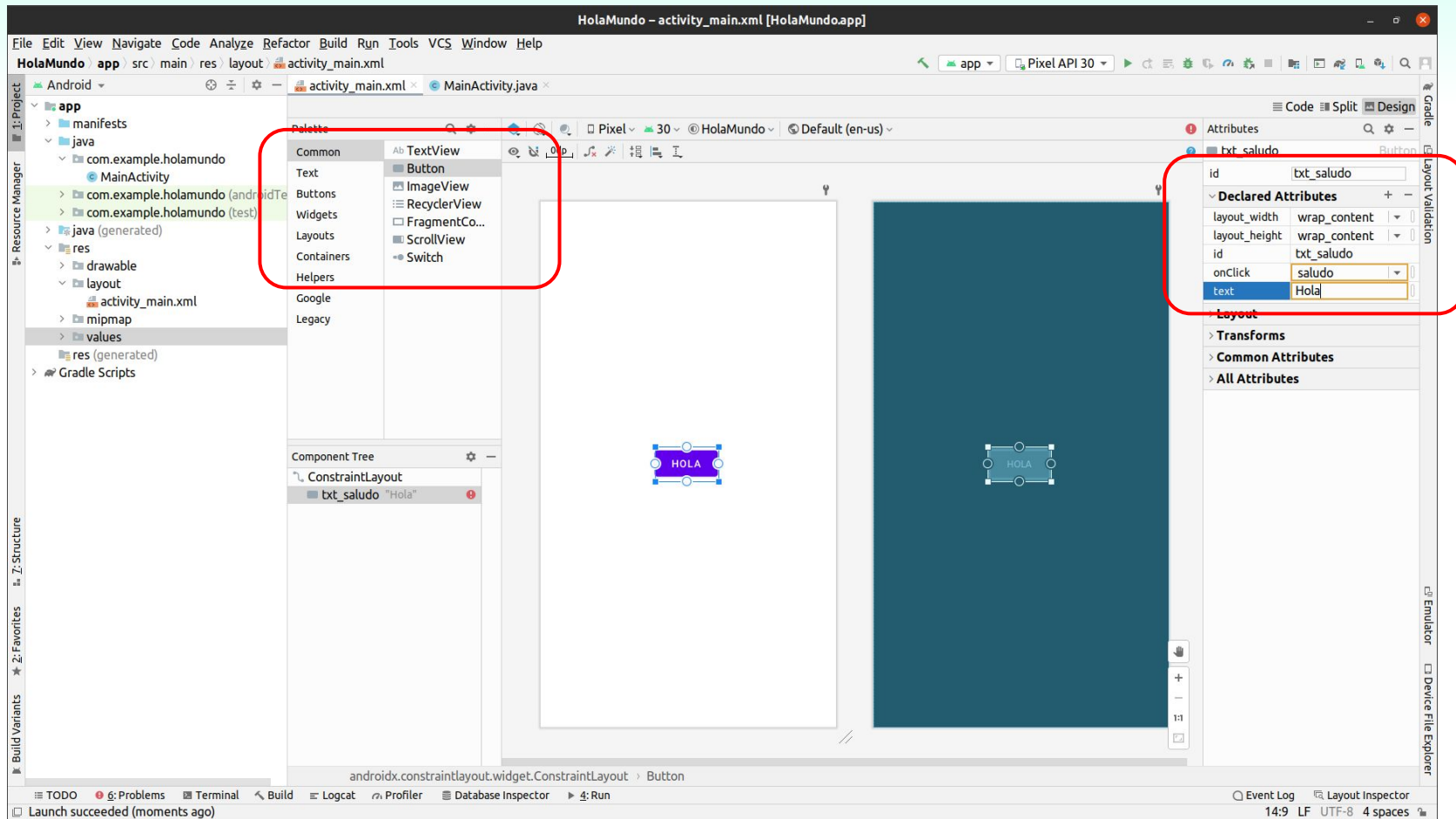
```
1 package com.example.holamundo;
2
3 import ...
4
5 public class MainActivity extends AppCompatActivity {
6
7     @Override
8     protected void onCreate(Bundle savedInstanceState) {
9         super.onCreate(savedInstanceState);
10         setContentView(R.layout.activity_main);
11     }
12
13     public void saludo(View v) {
14         Toast myToast = Toast.makeText(context: this, text: "Hola mundo!", Toast.LENGTH_SHORT);
15         myToast.show();
16     }
17 }
```

Editor de código

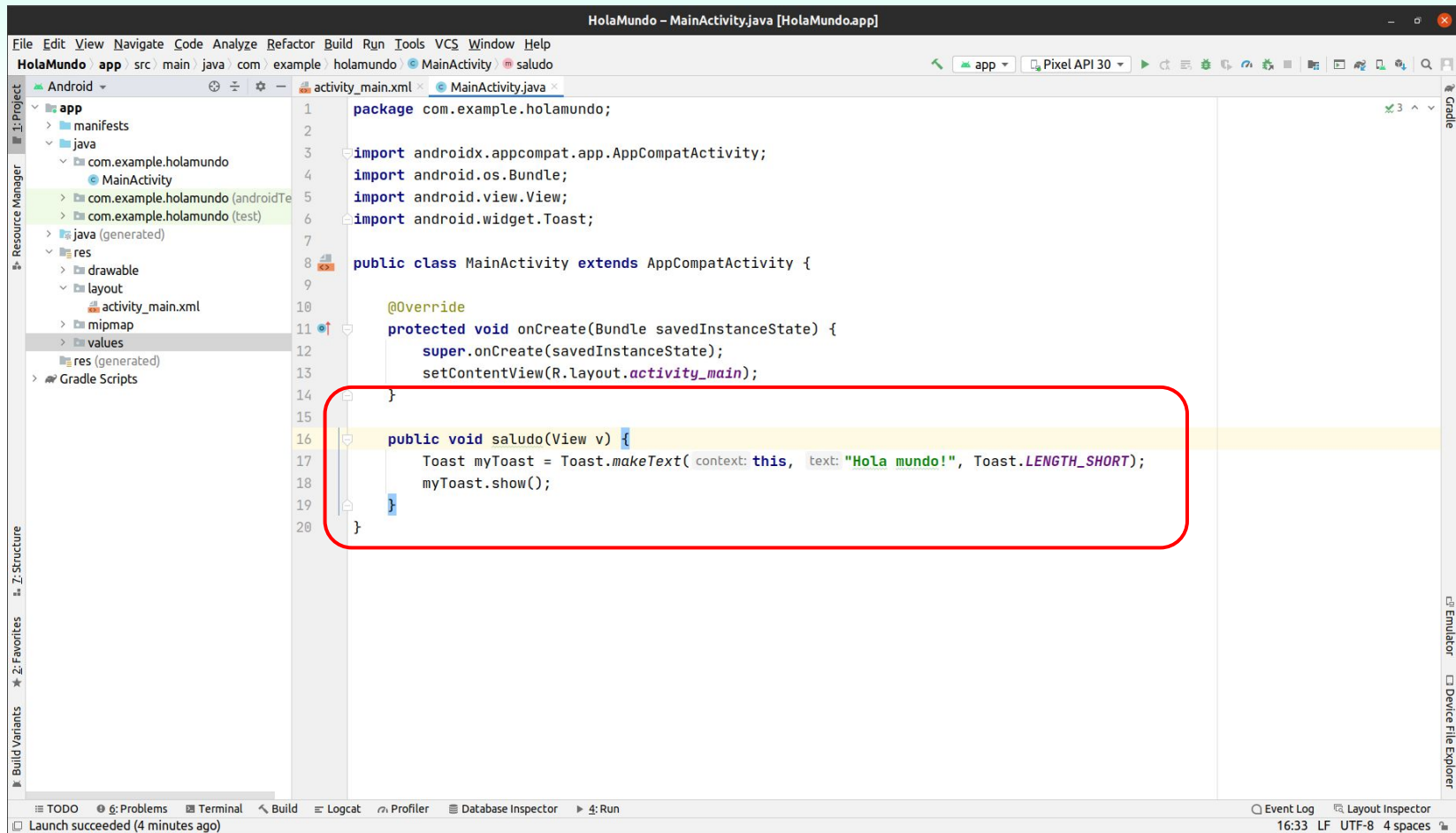
Diseño vs. Código



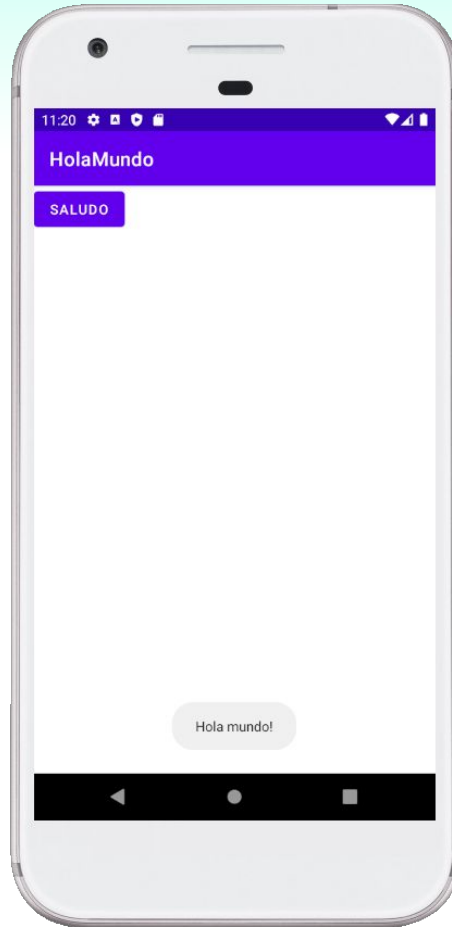
Diseño



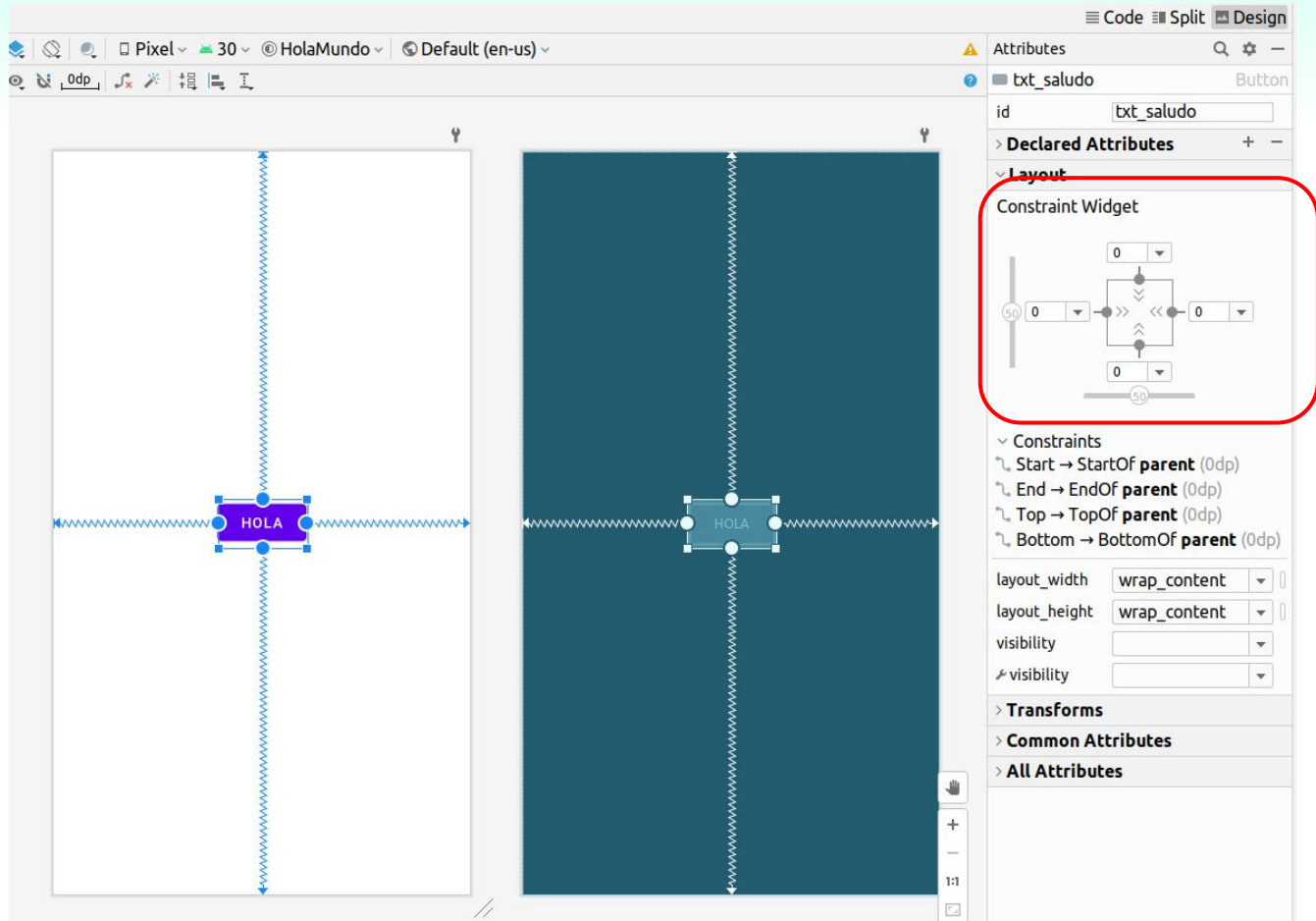
Código



Ejecución



Layout



¿Advertencias?

The screenshot displays the Android Studio IDE with the following components:

- Top Bar:** Tabs for `activity_main.xml` and `MainActivity.java`. A red box highlights the **Attributes** tab.
- Left Panel:**
 - Palette:** Shows widget categories like Common, Text, Buttons, etc. `TextView` is selected.
 - Component Tree:** Lists `ConstraintLayout` and `txt_saludo "Hola"` with a warning icon.
- Design View:** Shows a visual representation of the layout with a blue button.
- Right Panel:** The **Attributes** tab for `txt_saludo` is active, showing declared attributes and layout constraints.
- Bottom Panel:** Displays two warnings:
 - Warning 1:** **Using `android:onClick` on older version of the platform is broken**. It advises using databinding or explicit wiring and notes that old versions of the platform do not properly support resolving `android:onClick`. The issue ID is `UsingOnClickInXml`.
 - Warning 2:** **Hardcoded text**. It states that the hardcoded string `"Hola"` should use an `@string` resource. It explains that hardcoding text is bad for several reasons:
 - When creating configuration variations (for example for landscape or portrait) you have to repeat the actual text (and keep it up to date when making changes).
 - The application cannot be translated to other languages by just adding new translations for existing string resources.It also mentions that there are quickfixes to automatically extract this hardcoded string into a resource lookup.

Advertencia #1

1. activity_main.xml: Eliminar método **onClick** (modo diseño)
2. MainActivity.java:

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener {  
    private Button btnSaludo;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        btnSaludo = findViewById(R.id.btn_saludo);  
        btnSaludo.setOnClickListener(this);  
    }  
  
    @Override  
    public void onClick(View v) {  
        if (v == btnSaludo) {  
            Toast myToast = Toast.makeText(this, "Hola mundo!", Toast.LENGTH_SHORT);  
            myToast.show();  
        }  
    }  
}
```

Advertencia #2

1. activity_main.xml: Eliminar atributo **text** (modo Diseño)

2. activity_main.xml (modo Código):

```
<Button
```

```
    android:id="@+id/btn_saludo"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="@string/btn_saludo_texto"
```

```
    app:layout_constraintBottom_toBottomOf="parent"
```

```
    app:layout_constraintEnd_toEndOf="parent"
```

```
    app:layout_constraintStart_toStartOf="parent"
```

```
    app:layout_constraintTop_toTopOf="parent" />
```

3. strings.xml:

```
<resources>
```

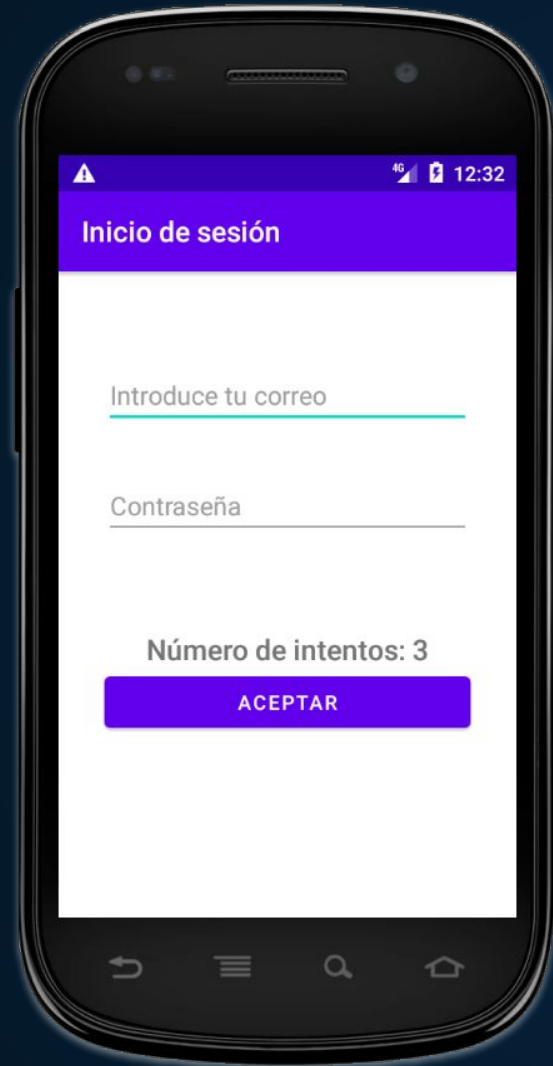
```
    <string name="app_name">HolaMundo</string>
```

```
    <string name="btn_saludo_texto">Hola</string>
```

```
</resources>
```

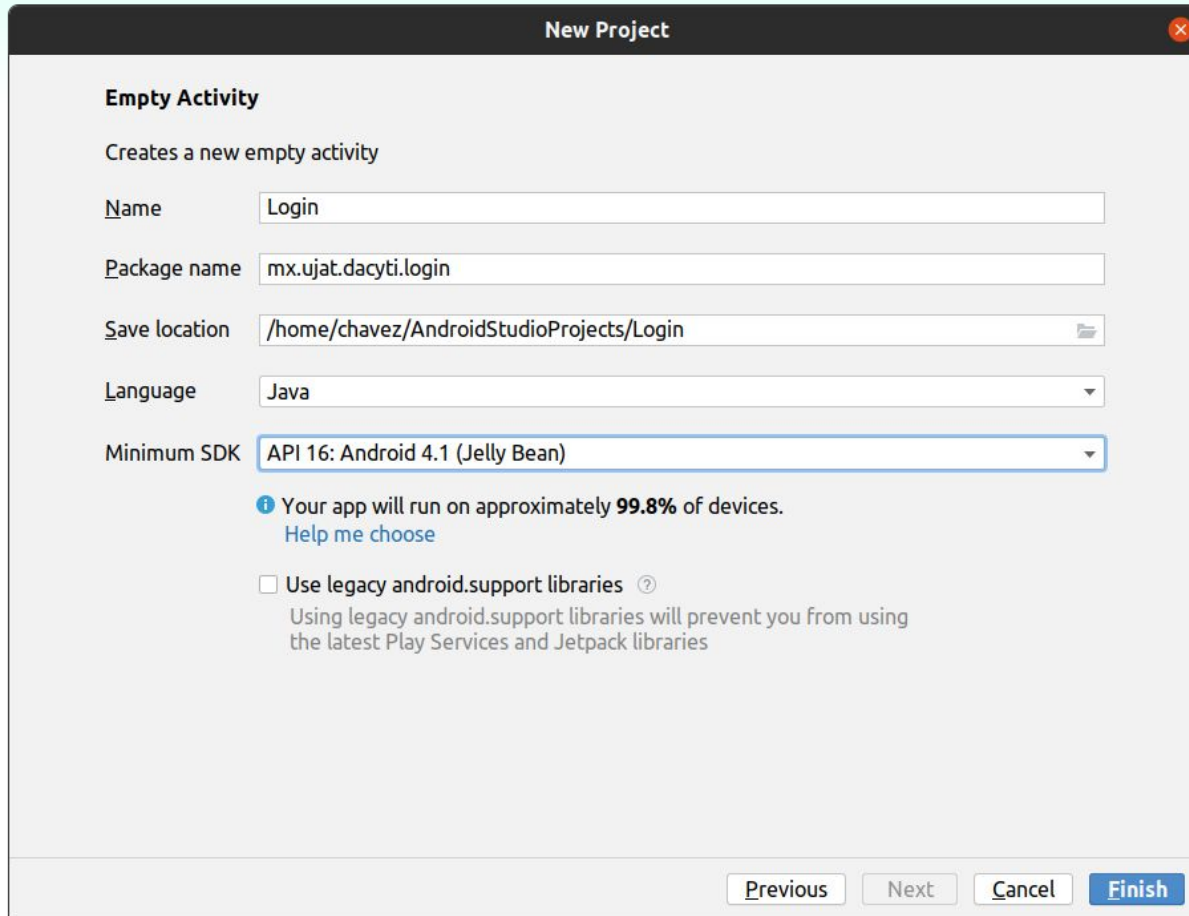
Eliminar una app del emulador

- Settings → Apps & notifications
- Seleccionar la app
- Uninstall



Login app

New project → Empty Activity



New Project

Empty Activity

Creates a new empty activity

Name

Package name

Save location

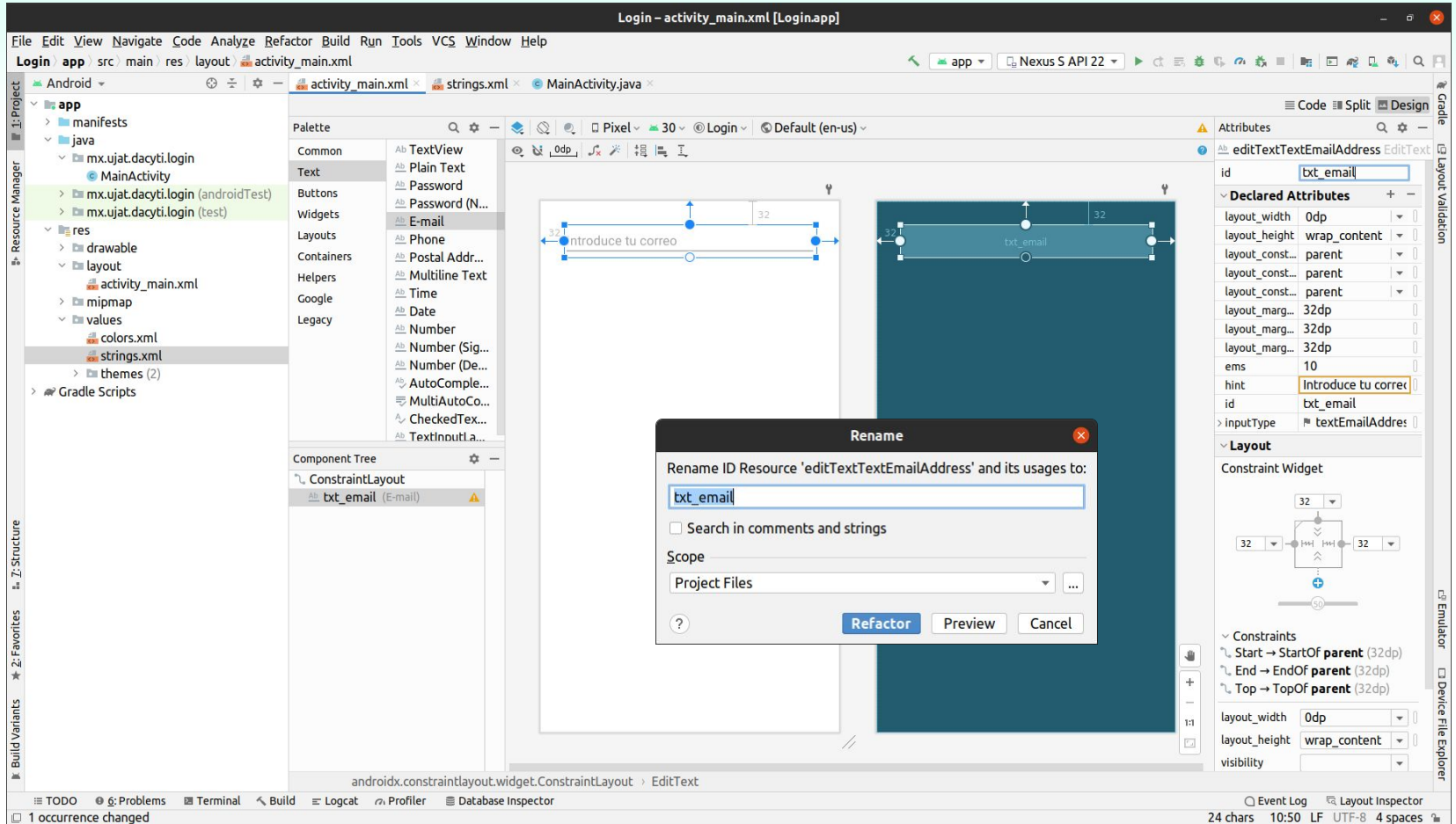
Language

Minimum SDK

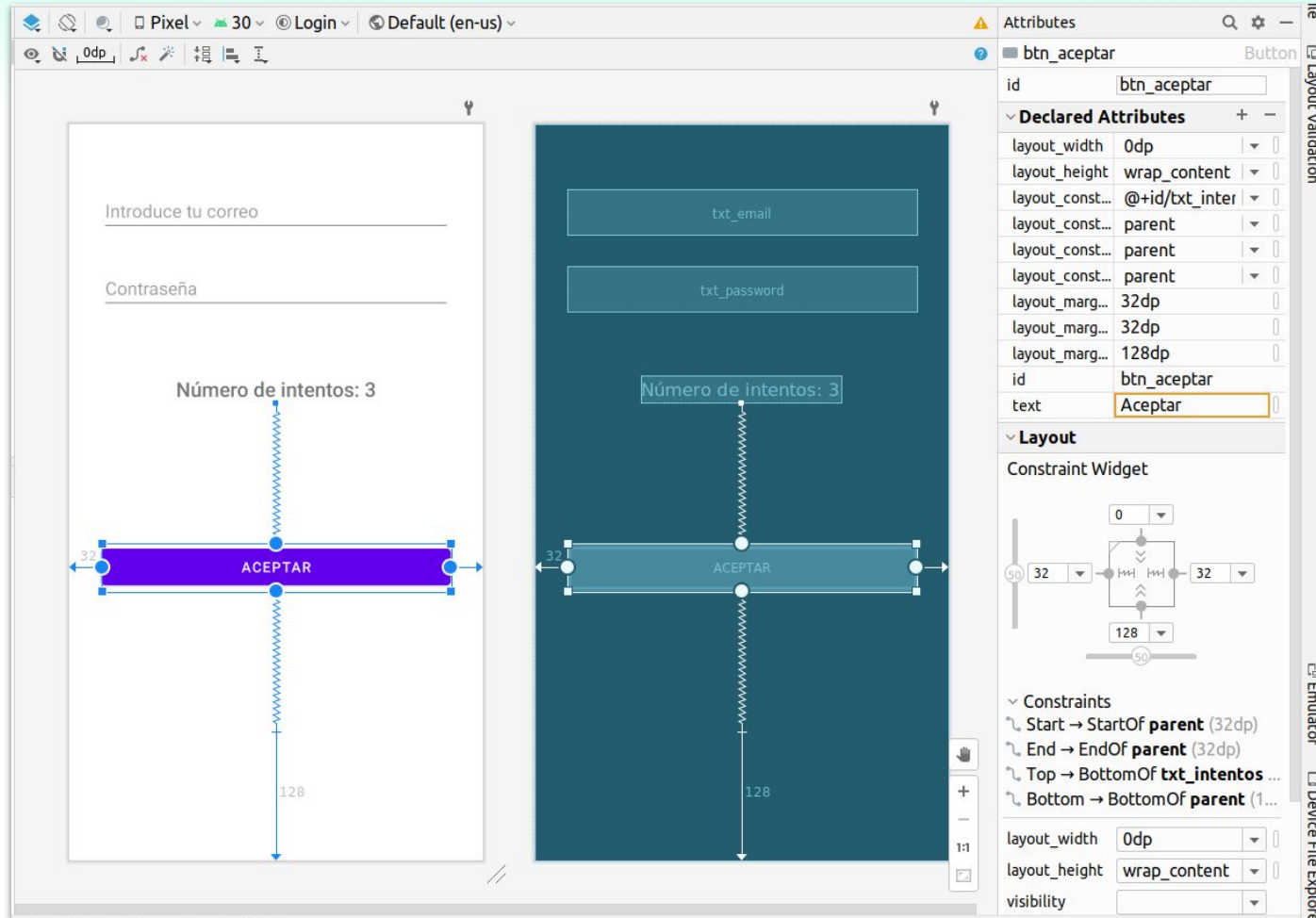
i Your app will run on approximately **99.8%** of devices.
[Help me choose](#)

☐ Use legacy android.support libraries **?**
Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries

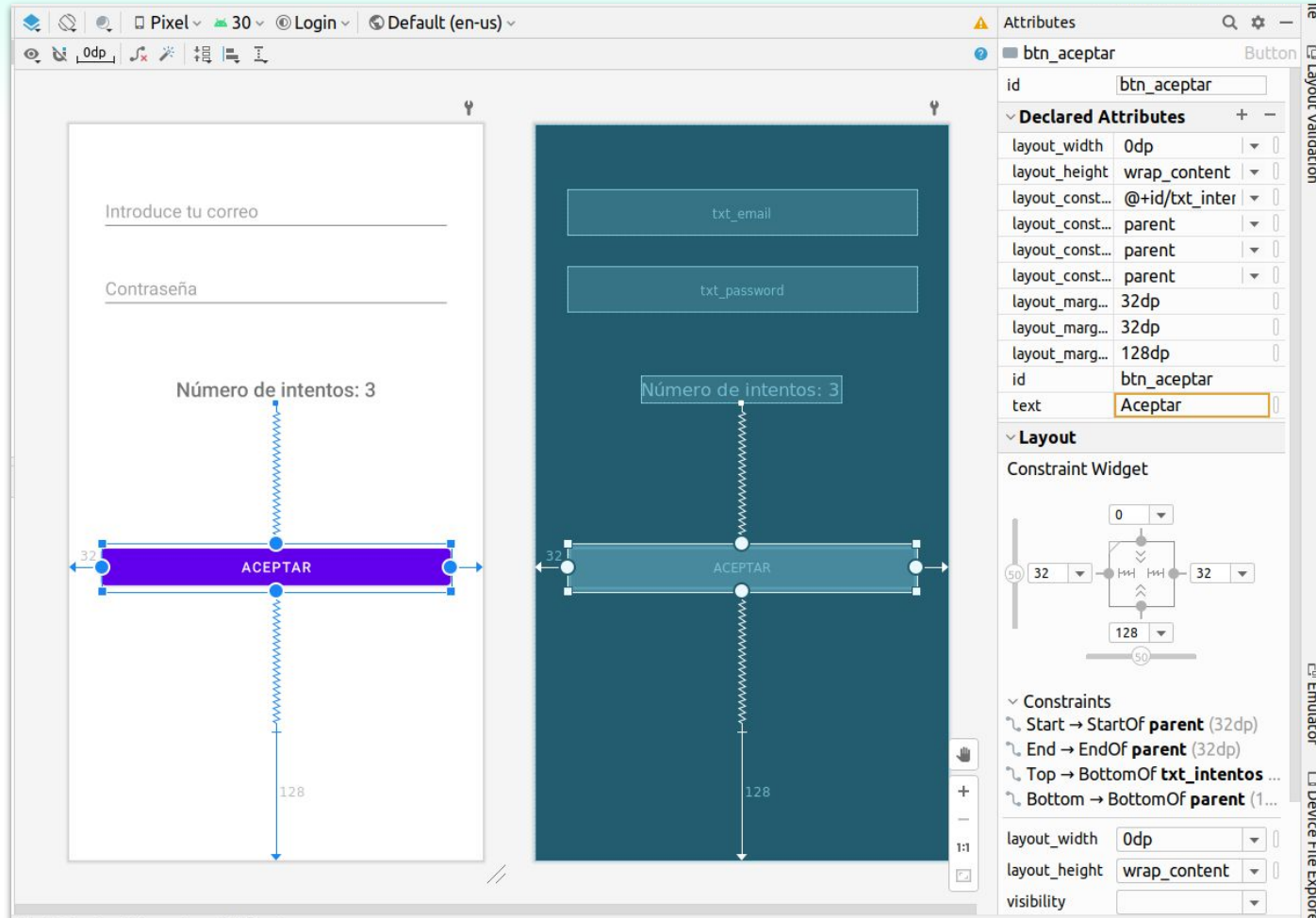
Primero el diseño...



Primero el diseño...



Primero el diseño...



Luego el código: MainActivity.java

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener {  
    Button btnAceptar;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        btnAceptar = findViewById(R.id.btn_aceptar);  
        btnAceptar.setOnClickListener(this);  
    }  
  
    @Override  
    public void onClick(View v) {  
        if (v == btnAceptar) {  
            Toast mensaje = Toast.makeText(this, "Bienvenido a mi app",  
                                           Toast.LENGTH_SHORT);  
            mensaje.show();  
        }  
    }  
}
```

Validando campos: MainActivity.java

```
EditText txtEmail;  
EditText txtPassword;  
...  
  
txtEmail = findViewById(R.id.txt_email);  
txtPassword = findViewById(R.id.txt_password);  
...  
  
@Override  
public void onClick(View v) {  
    String email = txtEmail.getText().toString();  
    String password = txtPassword.getText().toString();  
    if (v == btnAceptar) {  
        if (email.equals("") && password.equals("")) {  
            Toast mensaje = Toast.makeText(this, "Introduce tus credenciales de acceso",  
                                           Toast.LENGTH_SHORT);  
  
            View view = mensaje.getView();  
            view.setBackgroundColor(Color.parseColor("red"));  
            mensaje.show();  
            return;  
        }  
    }  
}
```

Controlando el acceso: MainActivity.java

```
TextView lblIntentos;  
int intentos = 3;  
...  
  
lblIntentos = findViewById(R.id.lbl_intentos);  
...  
  
if (email.equals("oscar") && password.equals("chavez")) {  
    Toast mensaje = Toast.makeText(this, "¡Bienvenido!", Toast.LENGTH_SHORT);  
    View view = mensaje.getView();  
    view.setBackgroundColor(Color.parseColor("#038012"));  
    mensaje.show();  
} else {  
    Toast mensaje = Toast.makeText(this, "Credenciales incorrectas", Toast.LENGTH_SHORT);  
    View view = mensaje.getView();  
    view.setBackgroundColor(Color.parseColor("yellow"));  
    mensaje.show();  
    intentos--;  
    lblIntentos.setText("Número de intentos: " + intentos);  
    if (intentos == 0) {  
        btnAceptar.setEnabled(false);  
    }  
}
```

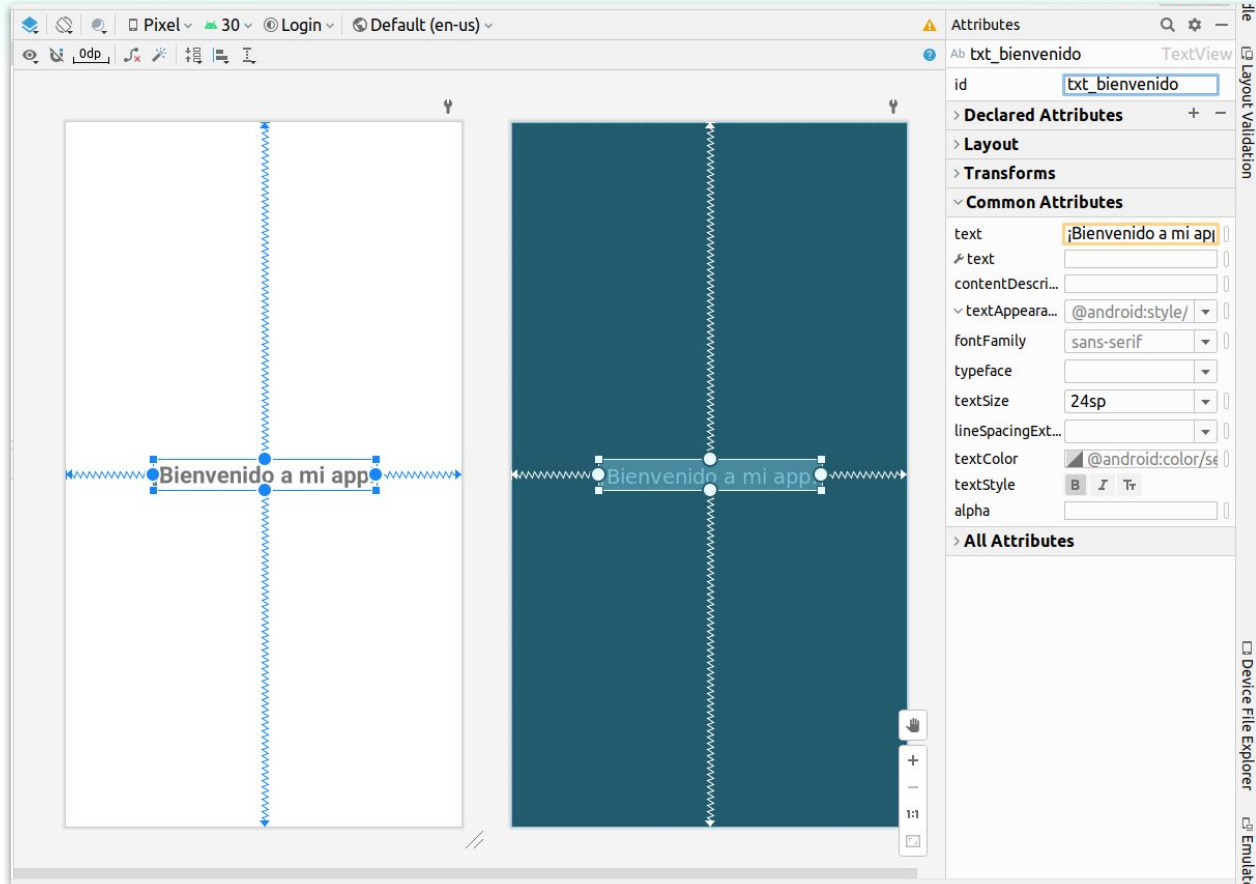
Cambiar título de la app: strings.xml

```
<resources>  
  <string name="app_name">Inicio de sesión</string>  
</resources>
```

Cambiar de pantalla: `Activity2.java`

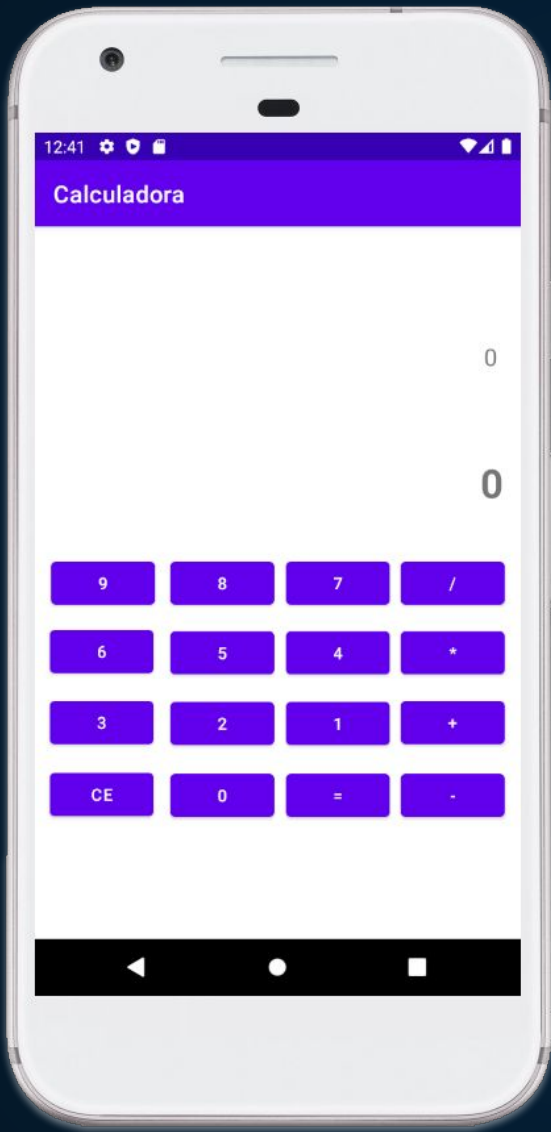
- File → New → Activity → Empty Activity
- `Activity2.java`
- Agregar una etiqueta: `¡Bienvenido a mi app!`

Activity2.java



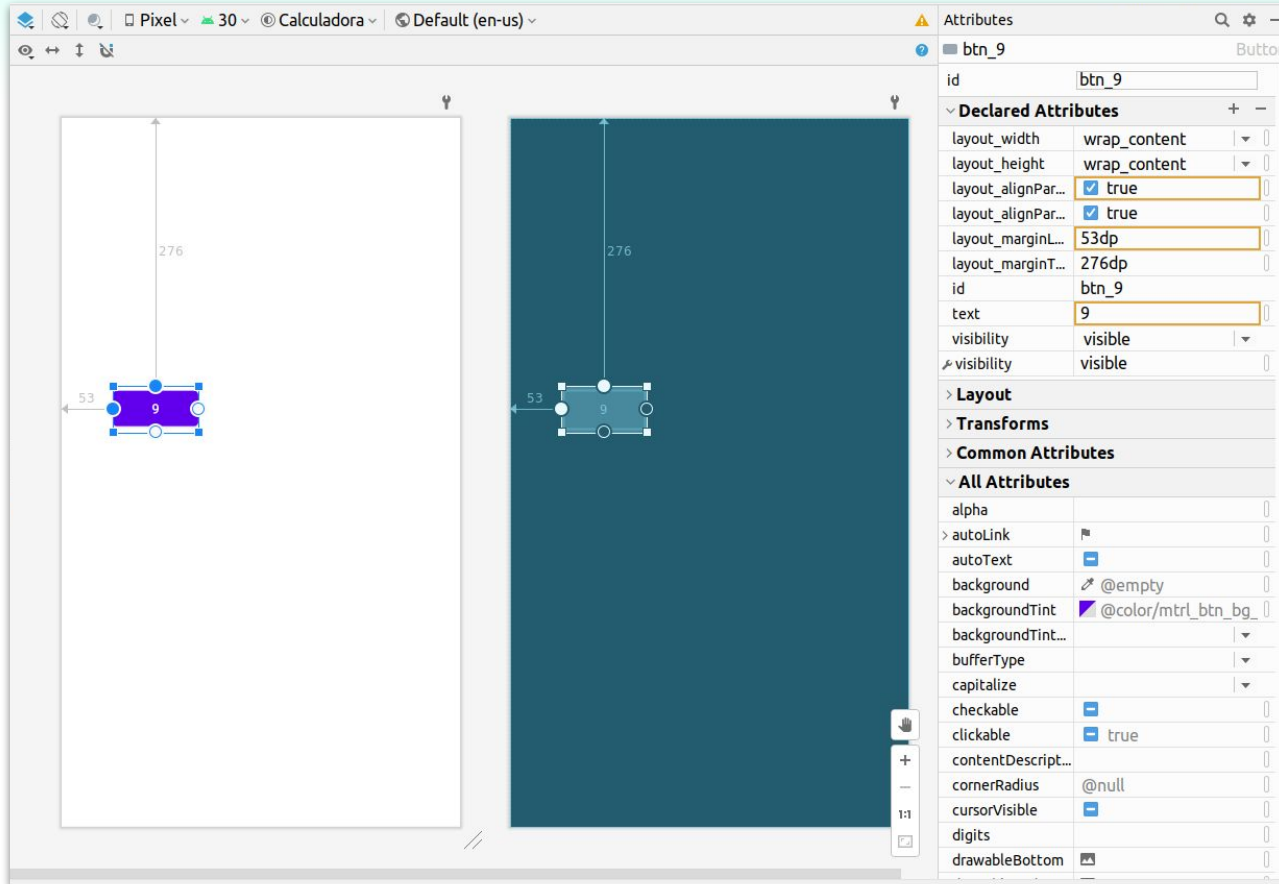
Cambiar de pantalla: MainActivity.java

```
...  
  
if (email.equals("oscar") && password.equals("chavez")) {  
    Intent intent = new Intent(this, Activity2.class);  
    startActivity(intent);  
}  
  
...
```

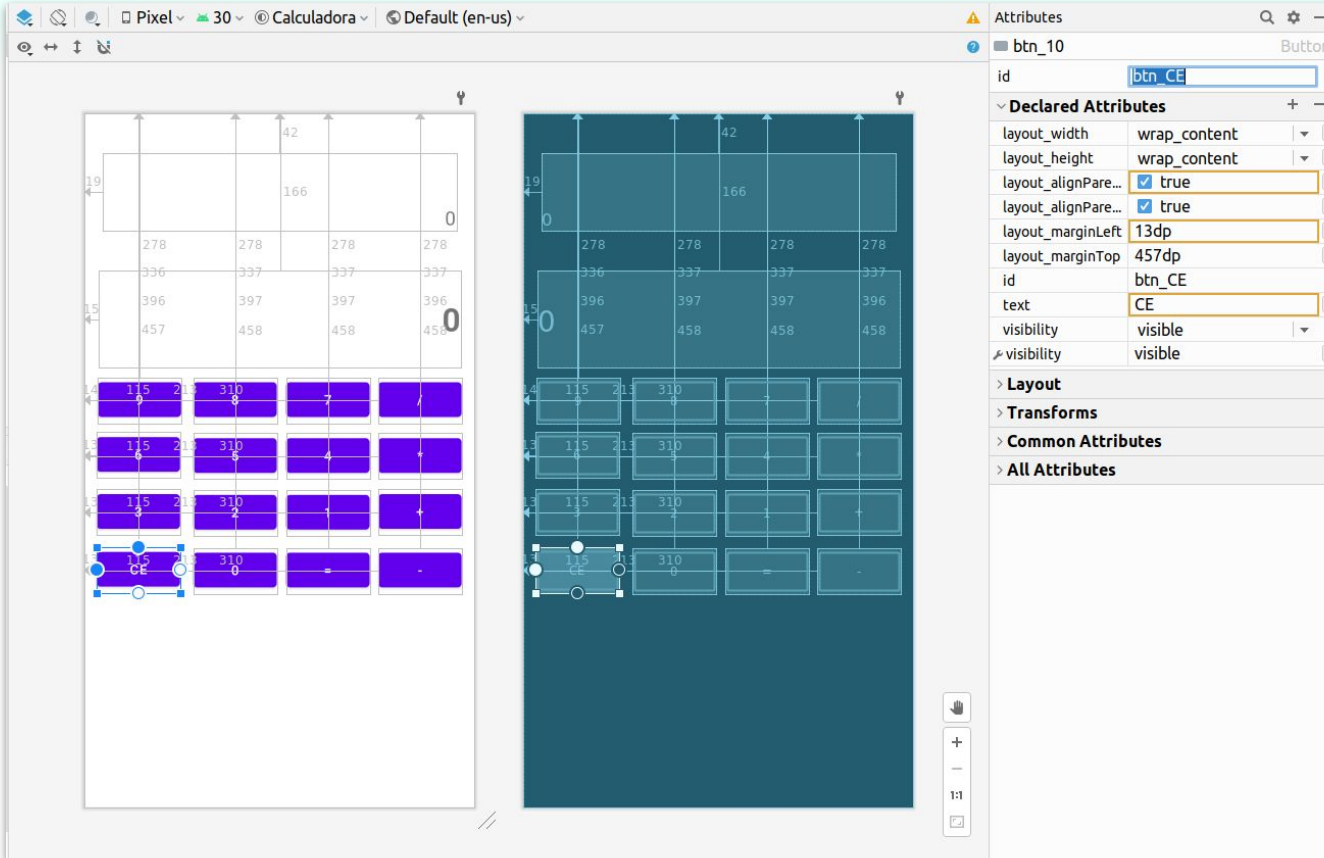



Calculadorapp

MainActivity.xml



MainActivity.xml



Hasta la próxima...