

DISEÑO Y DESARROLLO DE GUI (ACTIVITY)

LAYOUTS - LIENZOS

CONTENEDORES DE COMPONENTES

**CREAR UN NUEVO PROYECTO ANDROID**

**CAMBIAR EL CONTENEDOR**

**SELECCIONAR LAYOUT LINEARLAYOUT**

# DISEÑO DE GUI EN ANDROID

## LAYOUT Y CONTENEDORES

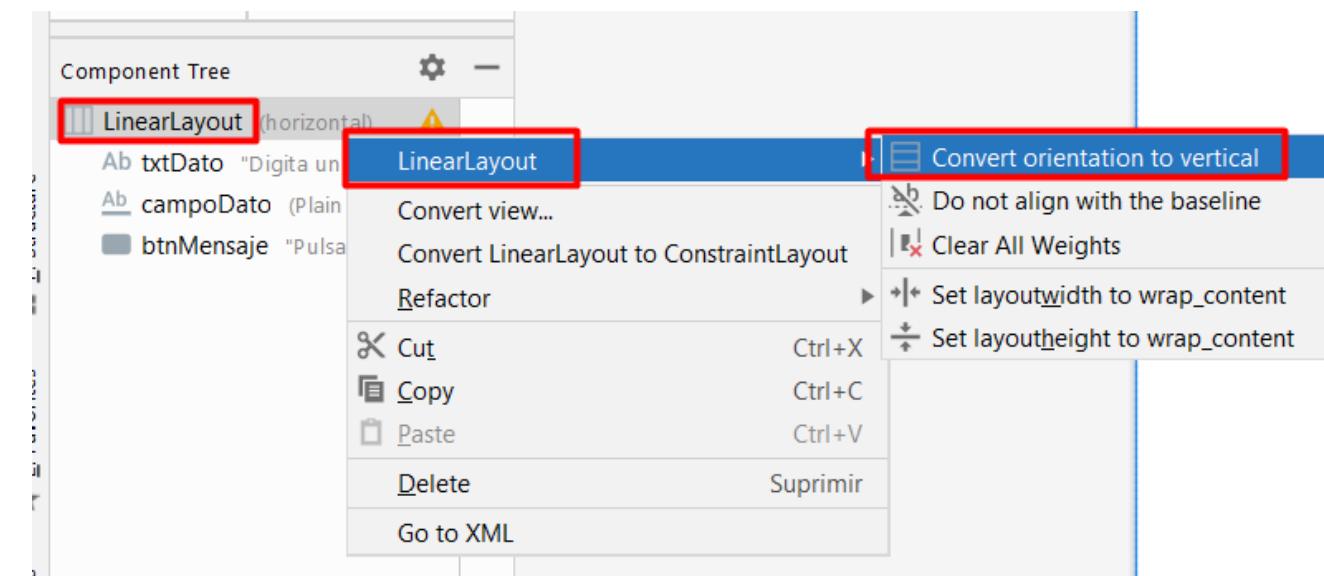
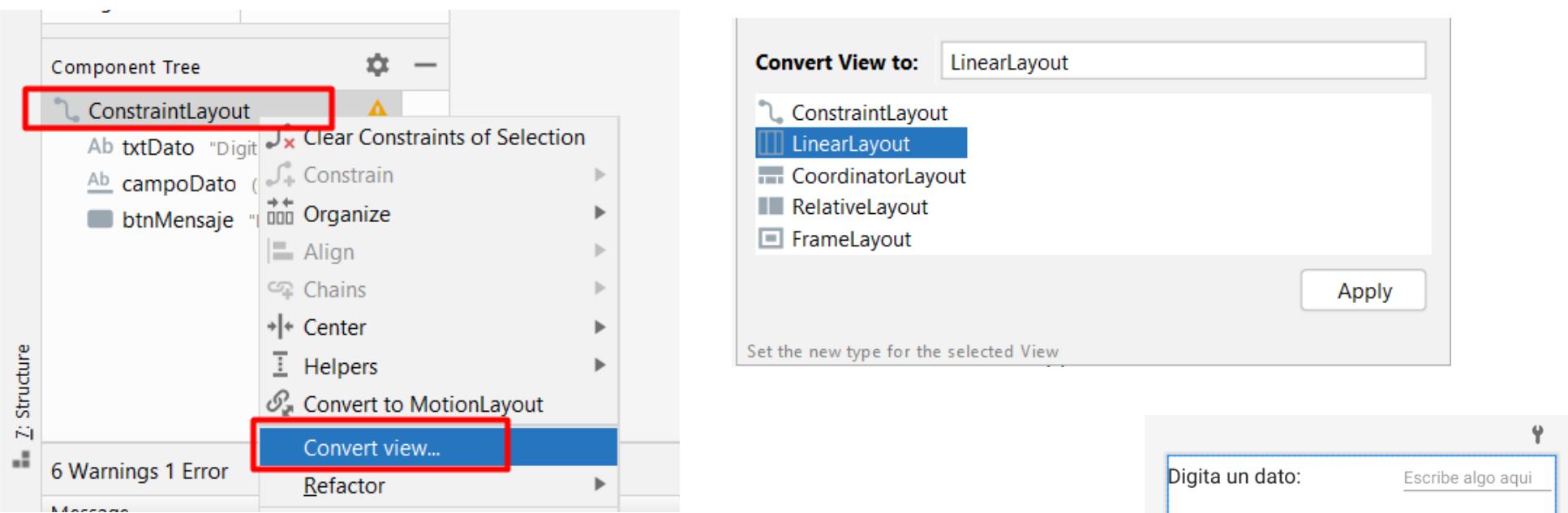
### CONVERTIR EL CONTENDOR

ConstraintLayout

EN UN CONTENDOR

LinearLayout

vertical



# DISEÑO DE GUI EN ANDROID

## LAYOUT Y CONTENEDORES

PANEL DE EDITOR DE PROPIEDADES O ATRIBUTOS PARA CADA UNO DE LOS COMPONENTES QUE AGREGAMOS A LA GUI

The screenshot shows two separate Attribute panels from the Android Studio Properties editor.

**Left Panel (TextView):**

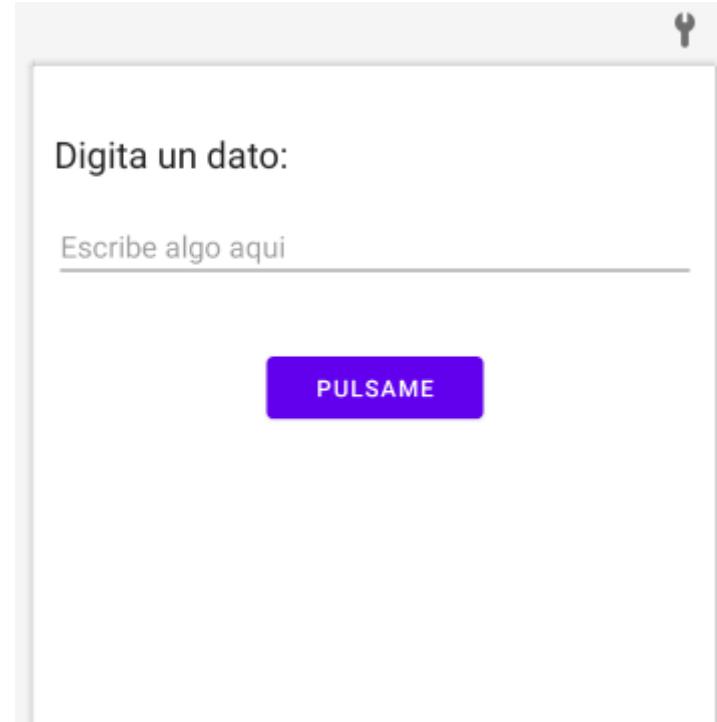
- id:** txtData (highlighted with a red arrow)
- Declared Attributes:**
  - layout\_width: match\_parent
  - layout\_height: wrap\_content
  - layout\_marginLeft: 30px (highlighted with a red arrow)
  - layout\_marginTop: 100px (highlighted with a red arrow)
  - layout\_marginRight: 30px (highlighted with a red arrow)
  - id: txtData
  - text: Digita un dato: (highlighted with a red arrow)
  - textAppearance: @style/TextAppearance.AppCompat.Large

**Right Panel (Button):**

- id:** btnMensaje (highlighted with a red arrow)
- Declared Attributes:**
  - layout\_width: 131dp
  - layout\_height: 49dp
  - id: btnMensaje (highlighted with a red arrow)
  - text: Pulsame (highlighted with a red arrow)

The screenshot shows the Attribute panel for an EditText component.

- id:** campoData (highlighted with a red arrow)
- Declared Attributes:**
  - layout\_width: match\_parent
  - layout\_height: wrap\_content
  - layout\_marginLeft: 30px (highlighted with a red arrow)
  - layout\_marginTop: 50px (highlighted with a red arrow)
  - layout\_marginRight: 30px (highlighted with a red arrow)
  - layout\_marginBottom: 100px (highlighted with a red arrow)
  - ems: 10
  - hint: Escribe algo aqui (highlighted with a red arrow)
  - id: campoData
  - inputType:** textPersonName (highlighted with a red arrow)



# DISEÑO DE GUI EN ANDROID

## LAYOUT Y

## CONTENEDORES

LA ESTRUCTURA DE LA ACTIVIDAD PRINCIPAL, COMPUESTA POR UN CONTENEDOR

LinearLayout

VERTICAL, EL

CUAL CONTINE 3 COMPONENTES,

UN TextView, UN

EditText Y UN

Button

actividad\_principal\_view.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:gravity="center_horizontal"
8      android:orientation="vertical"
9      tools:context=".ActividadPrincipal">
10
11     <TextView ...>
12
13     <EditText ...>
14
15     <Button ...>
16
17 </LinearLayout>
```

# DISEÑO DE GUI EN ANDROID

## LAYOUT Y

## CONTENEDORES

XML QUE DEFINE LOS COMPONENTES AGREGADOS A LOS CONTENEDORES O LAYUOT QUE CONFORMAN CADA ACTIVIDAD (GUI)

### <TextView

```
    android:id="@+id/txtDato"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="30px"
    android:layout_marginTop="100px"
    android:layout_marginRight="30px"
    android:text="Digita un dato:"
    android:textAppearance="@style/TextAppearance.AppCompat.Large" />
```

### <EditText

```
    android:id="@+id/campoDato"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="30px"
    android:layout_marginTop="50px"
    android:layout_marginRight="30px"
    android:layout_marginBottom="100px"
    android:ems="10"
    android:hint="Escribe algo aqui"
    android:inputType="textPersonName" />
```

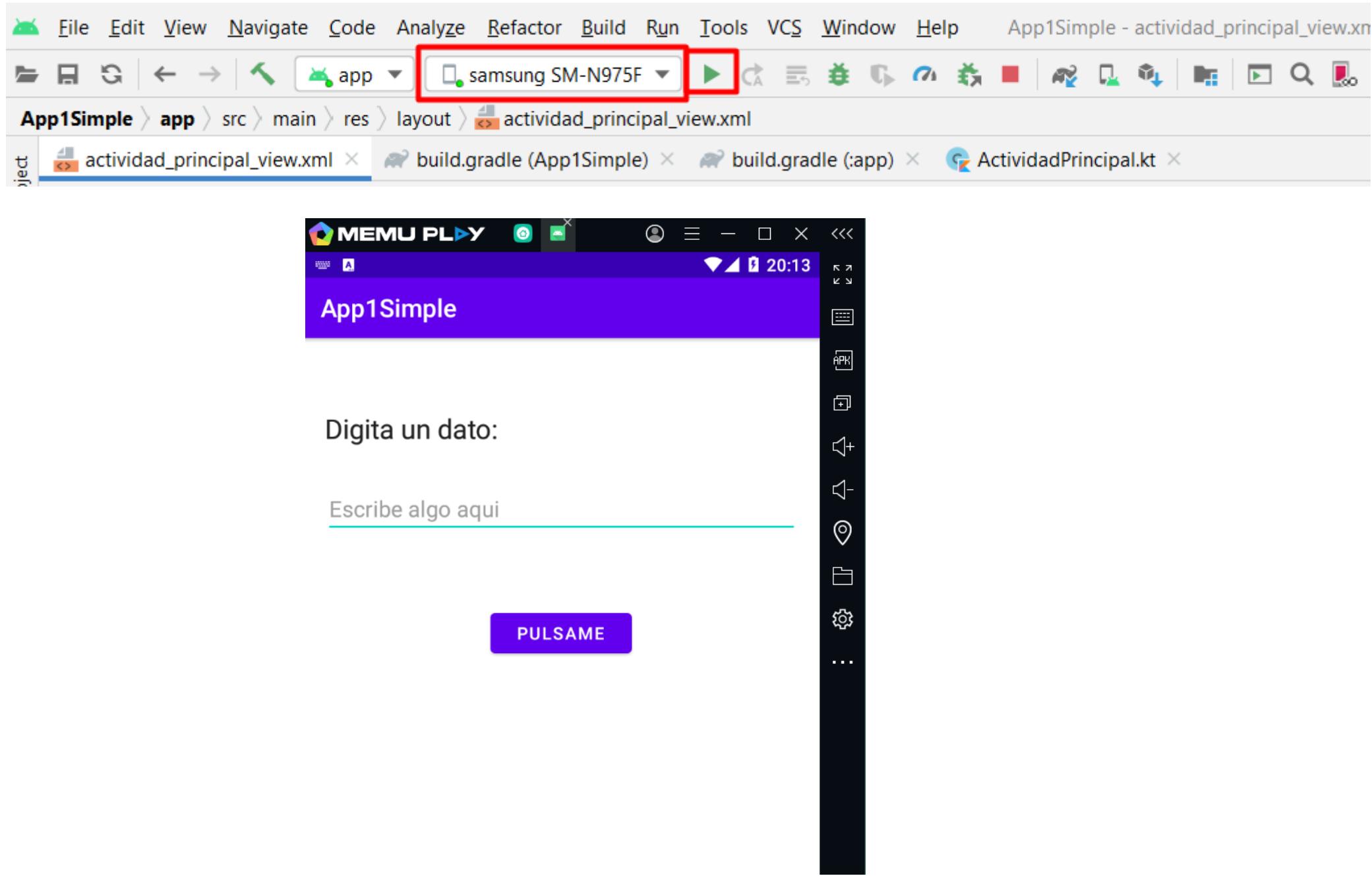
### <Button

```
    android:id="@+id/btnMensaje"
    android:layout_width="131dp"
    android:layout_height="49dp"
    android:text="Pulsame" />
```

# DISEÑO DE GUI EN ANDROID

## LAYOUT Y CONTENEDORES

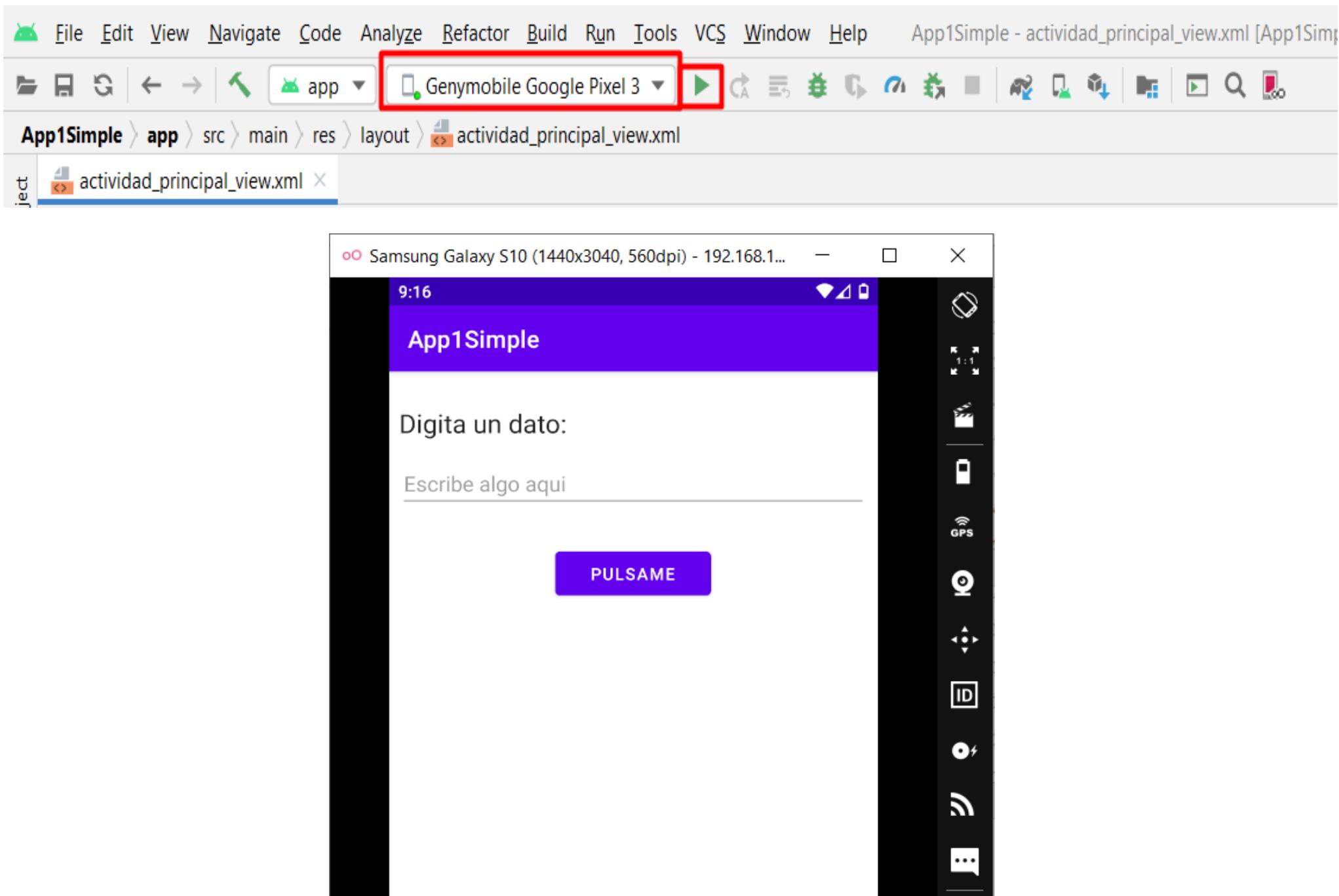
### EMULADOR MEMU PLAY



# DISEÑO DE GUI EN ANDROID

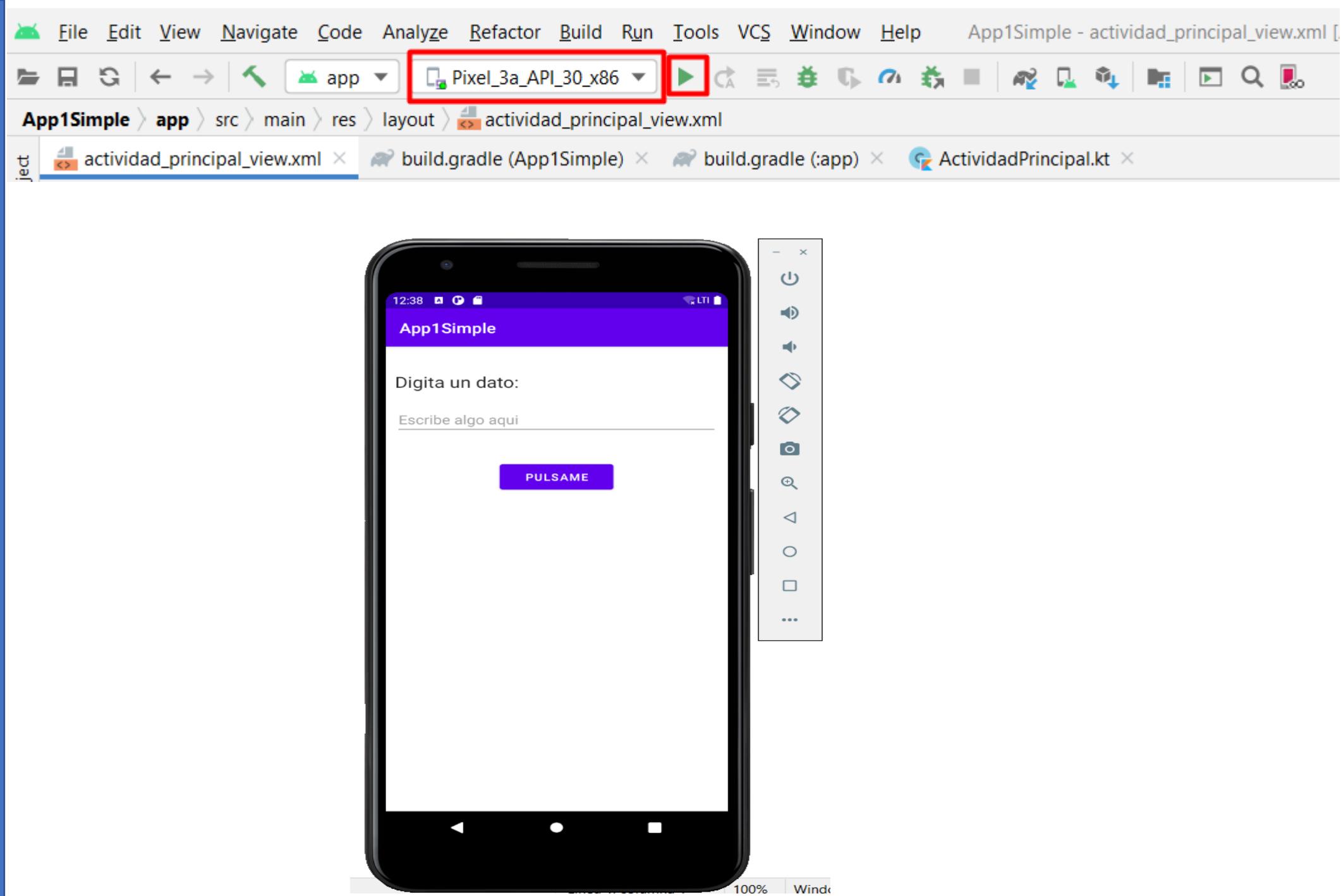
## LAYOUT Y CONTENEDORES

## EMULADOR GENYMOTION



# DISEÑO DE GUI EN ANDROID

## EMULADOR DE ANDROID SDK



# DISEÑO DE GUI EN ANDROID

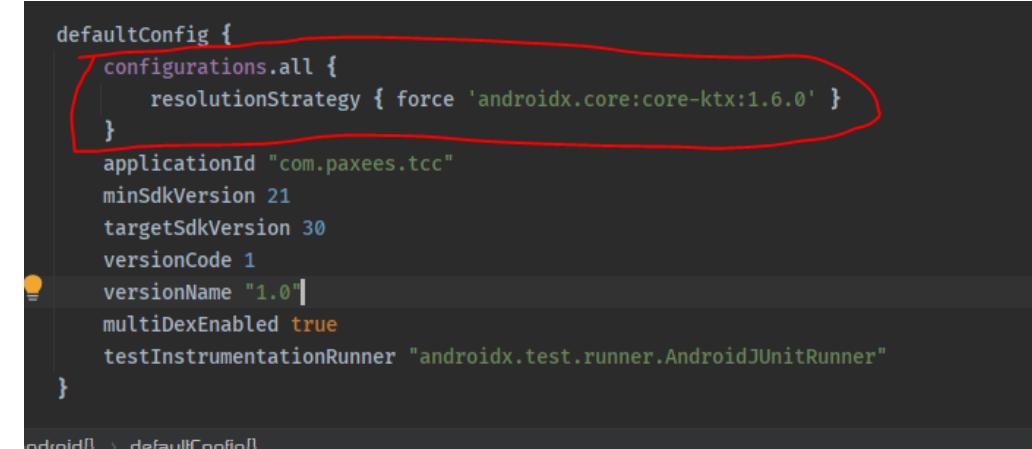
## PROBLEMAS

No puede ejecutar la App en el emulador

- ERROR DE minCompileSdk (31)

```
configurations.all {
    resolutionStrategy { force 'androidx.core:core-ktx:1.6.0' }
}

7   android {
8       compileSdkVersion 30
9       buildToolsVersion "30.0.3"
10
11      defaultConfig {
12          configurations.all {
13              resolutionStrategy { force 'androidx.core:core-ktx:1.6.0' }
14          }
15          applicationId "co.gov.misiontic.ciclo4"
16          minSdkVersion 17
17          targetSdkVersion 30
18          versionCode 1
19          versionName "1.0"
20
21          testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
22      }
23  }
```



```
defaultConfig {
    configurations.all {
        resolutionStrategy { force 'androidx.core:core-ktx:1.6.0' }
    }
    applicationId "com.paxees.tcc"
    minSdkVersion 21
    targetSdkVersion 30
    versionCode 1
    versionName "1.0"
    multiDexEnabled true
    testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
}
```

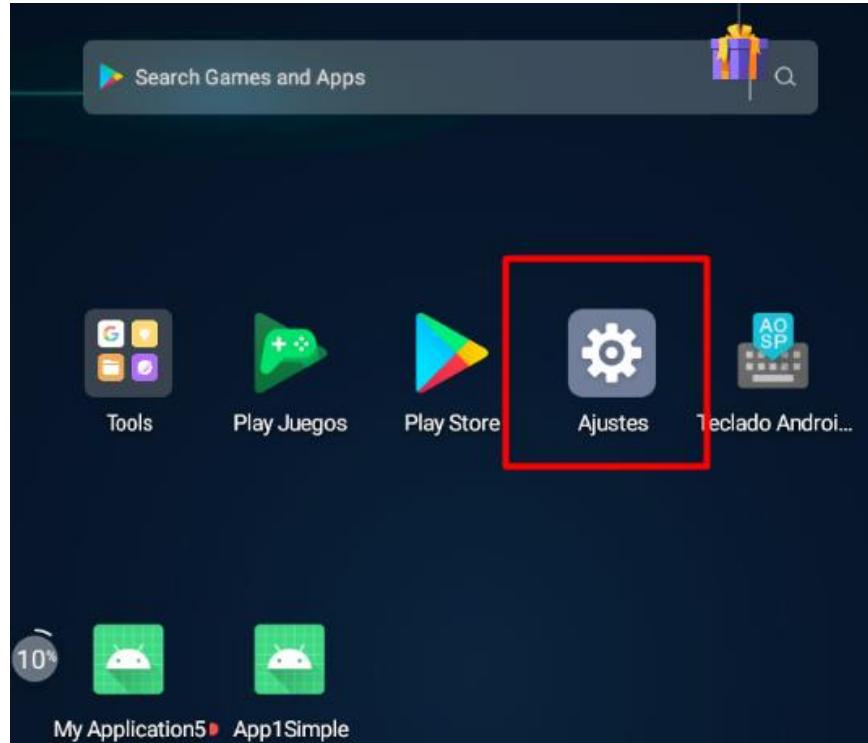
build.gradle

# DISEÑO DE GUI EN ANDROID

## PROBLEMAS

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador



Información del tablet  
Android 7.1.2

Versión del kernel  
4.0.9  
cjw@mv-dev1 #1634  
Tue May 25 11:20:57 CST 2021

Número de compilación  
samsung-user 7.1.2 20171130.276299 release-keys

Opciones de desarrollo

Verificar aplicaciones por USB  
Comprueba las aplicaciones instaladas mediante ADB/ADT para detectar comportamientos dañinos

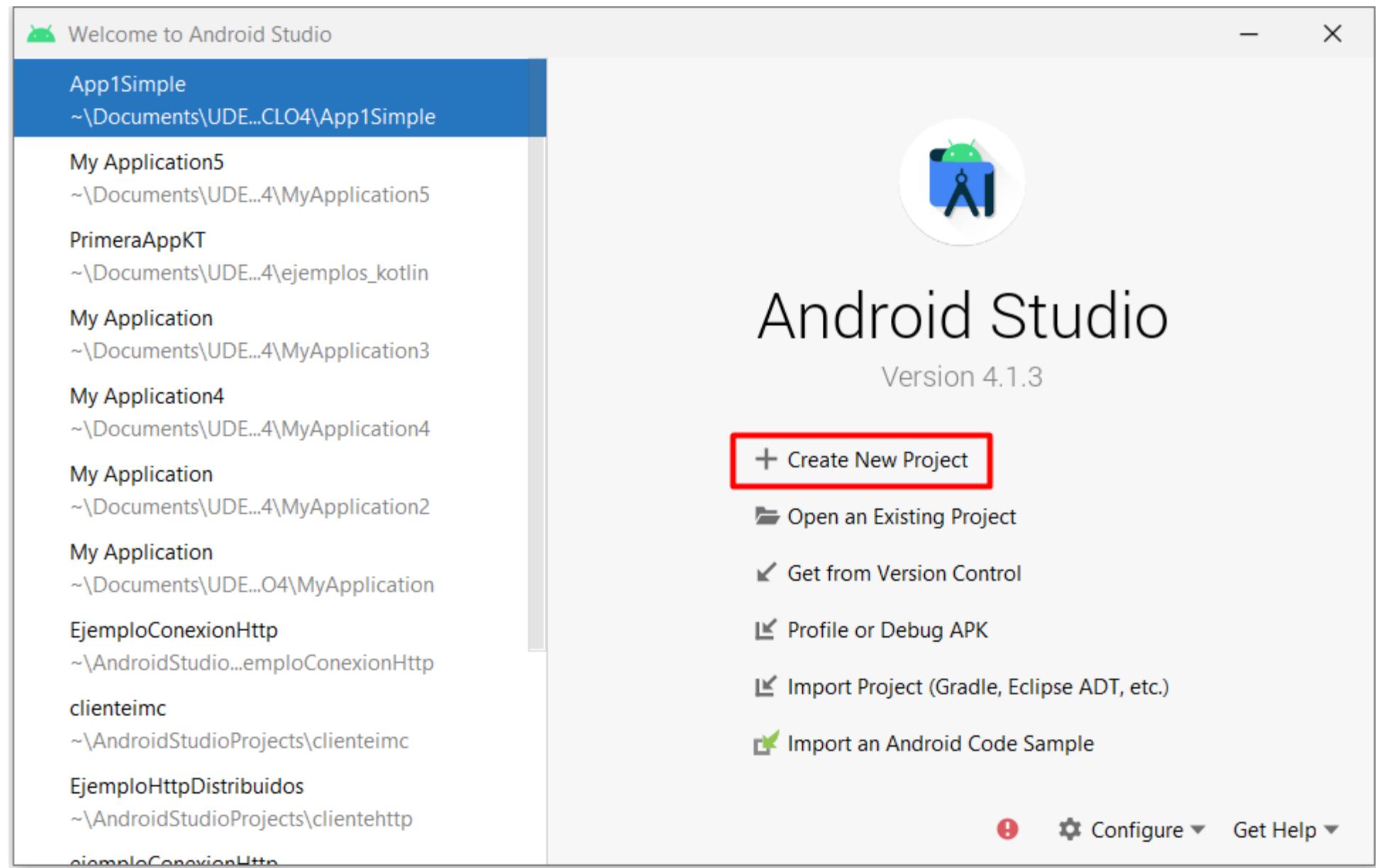
A toggle switch is shown in the 'on' position, indicated by a green circle, and is enclosed in a red box.

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

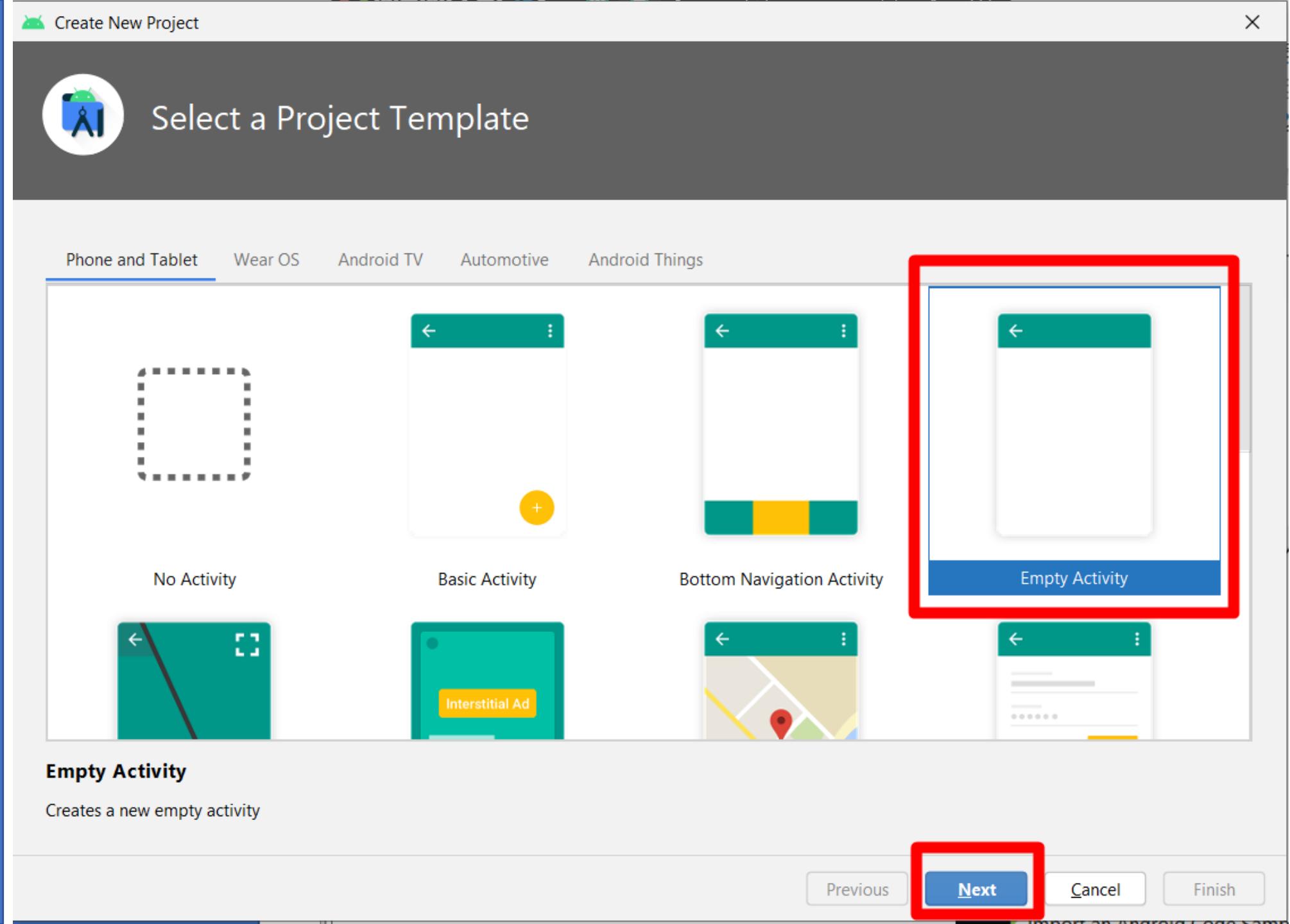


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar  
la App en el  
emulador

Al instalar el APK  
en el Emulador

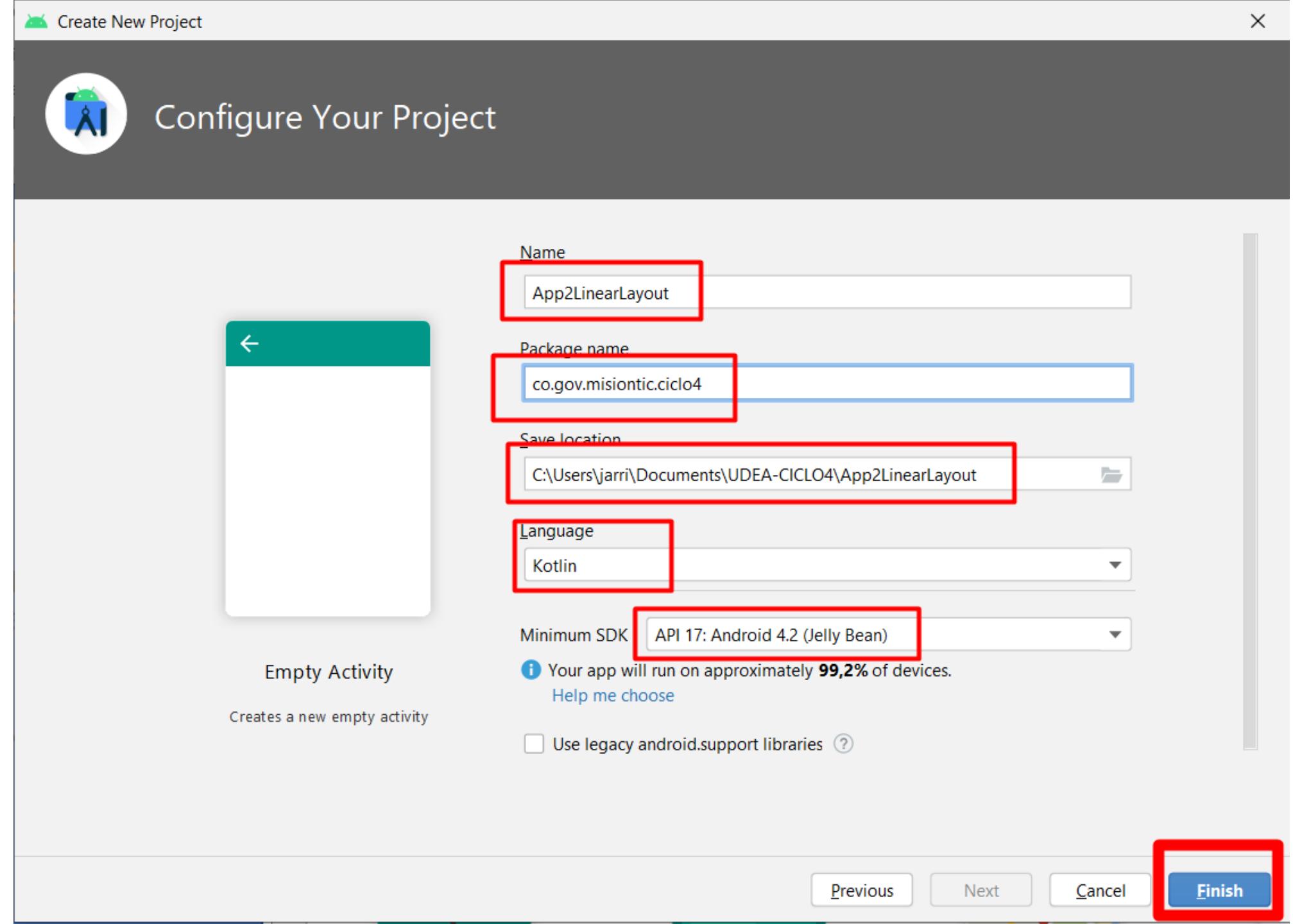


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar  
la App en el  
emulador

Al instalar el APK  
en el Emulador



# DISEÑO DE GUI EN ANDROID

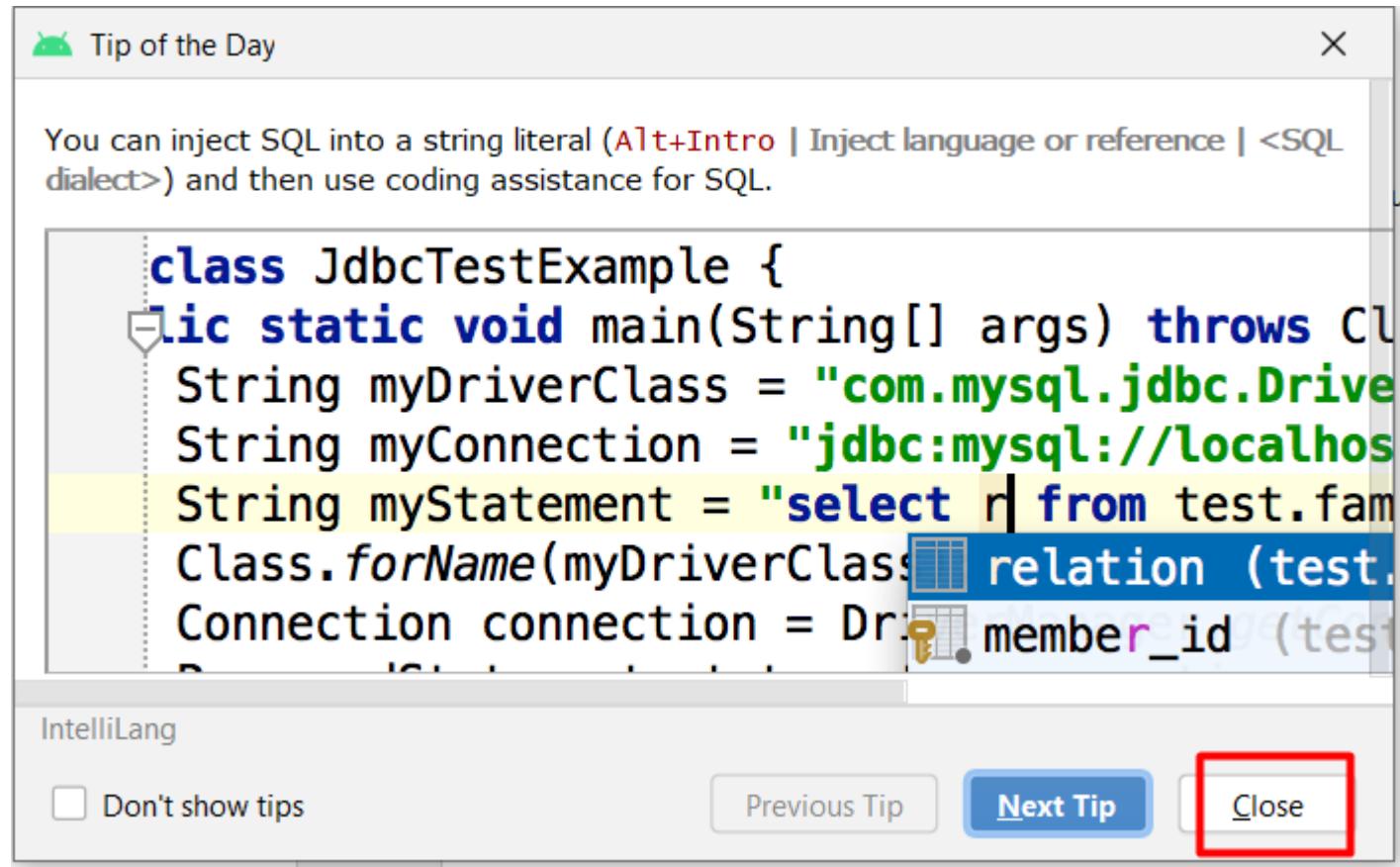
# EJEMPLO 2

## USO DE

### LinearLayout

No puede ejecutar  
la App en el  
emulador

## Al instalar el APK en el Emulador



# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

The screenshot shows the Android Studio interface with the following details:

- File Bar:** File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help, App2LinearLayout.
- Toolbar:** Includes icons for file operations, navigation, and device selection (Genymobile Google Pixel 3).
- Project Structure:** Shows the project tree with app, manifests, Java, and resources.
- Main Activity:** The Java file `MainActivity.kt` is open in the editor. It contains the following code:

```
package co.gov.misiontic.ciclo4

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
}
```

A red box highlights the tab bar for `MainActivity.kt`, and a red arrow points from the error in the code to the `activity_main.xml` file in the structure view.
- Layout XML:** The XML file `activity_main.xml` is also highlighted with a red box and arrow, indicating a dependency issue.
- Bottom Bar:** Includes tabs for TODO, Terminal, Build, Logcat, Profiler, Database Inspector, Event Log, and Layout Inspector. A message at the bottom states "Gradle sync finished in 26 s 420 ms (8 minutes ago)".

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

The screenshot shows the Android Studio interface with the project 'App2LinearLayout' open. The 'AndroidManifest.xml' file is selected in the top navigation bar and the main editor area. The code in the manifest is highlighted with red boxes:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="co.gov.misiontic.ciclo4">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="App2LinearLayout"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.App2LinearLayout">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
```

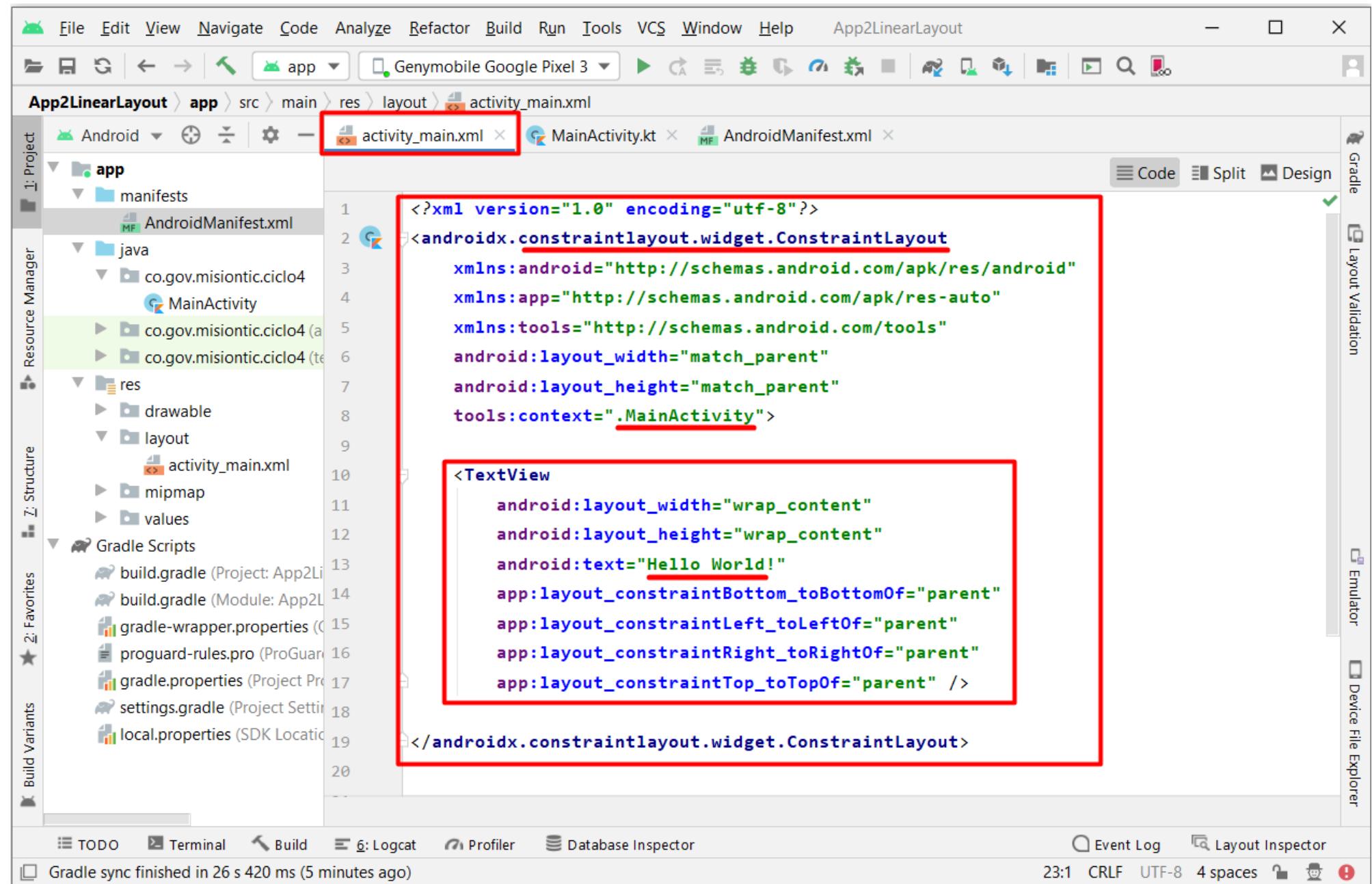
The highlighted sections include the package declaration, the application element with its attributes, and the activity element with its intent filter. The code editor shows syntax highlighting for XML tags and attributes.

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador



The screenshot shows the Android Studio interface with the following details:

- File Bar:** File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help, App2LinearLayout.
- Toolbar:** Includes icons for file operations, navigation, and various developer tools.
- Project Structure:** Shows the project tree under "app".
  - manifests: AndroidManifest.xml
  - java: co.gov.misiontic.ciclo4 (MainActivity)
  - res: layout (activity\_main.xml), drawable, mipmap, values
  - Gradle Scripts: build.gradle, gradle-wrapper.properties, proguard-rules.pro, gradle.properties, settings.gradle, local.properties
- Activity Tab:** activity\_main.xml (selected)
- Code Editor:** Displays the XML code for activity\_main.xml. A red box highlights the entire code block.

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    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">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

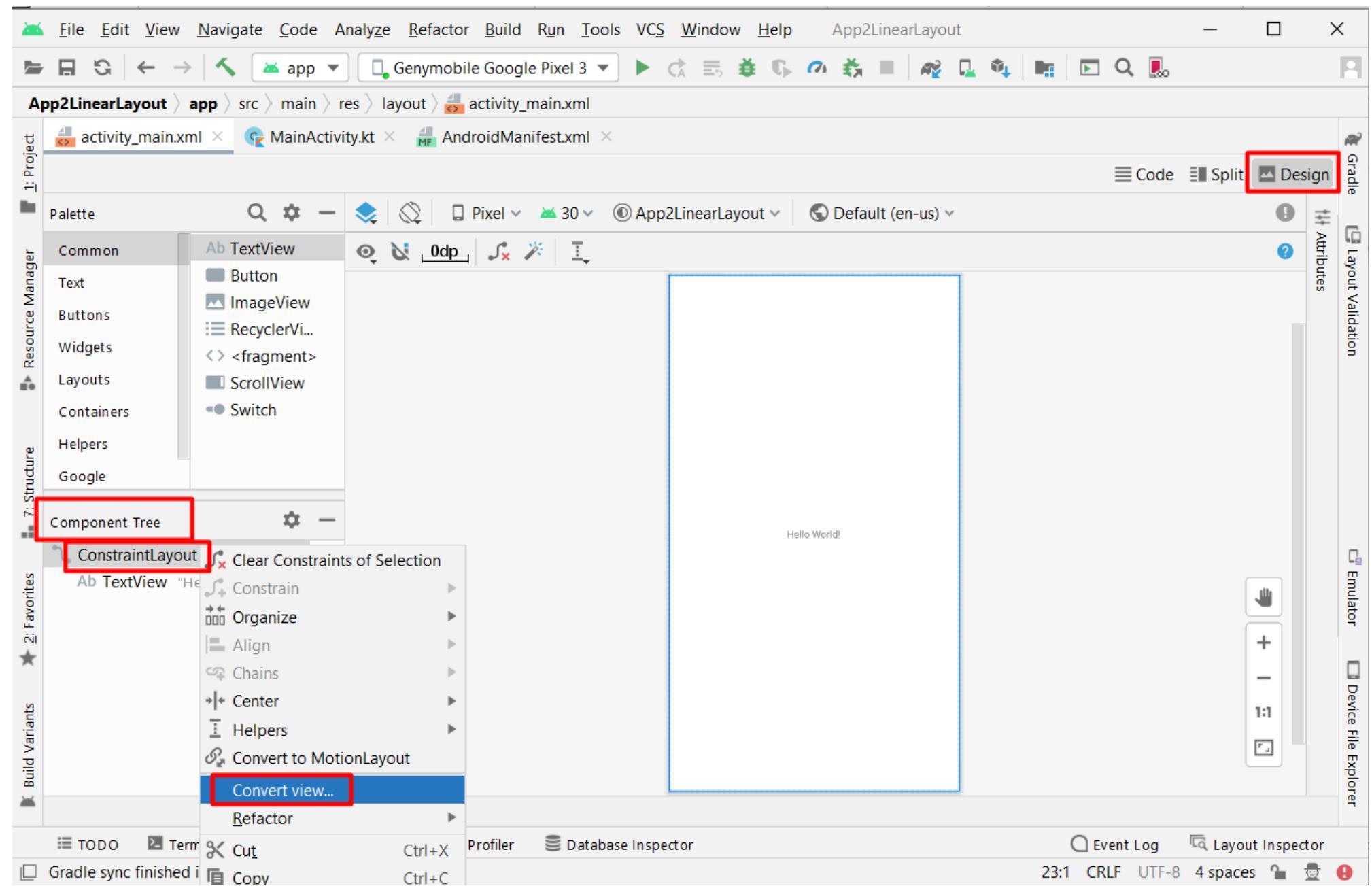
</androidx.constraintlayout.widget.ConstraintLayout>
```
- Toolbars:** Code, Split, Design, Gradle, Layout Validation, Emulator, Device File Explorer.
- Bottom Status Bar:** Event Log, Layout Inspector, Logcat, Profiler, Database Inspector, Gradle sync finished in 26 s 420 ms (5 minutes ago), 23:1 CRLF UTF-8 4 spaces.

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

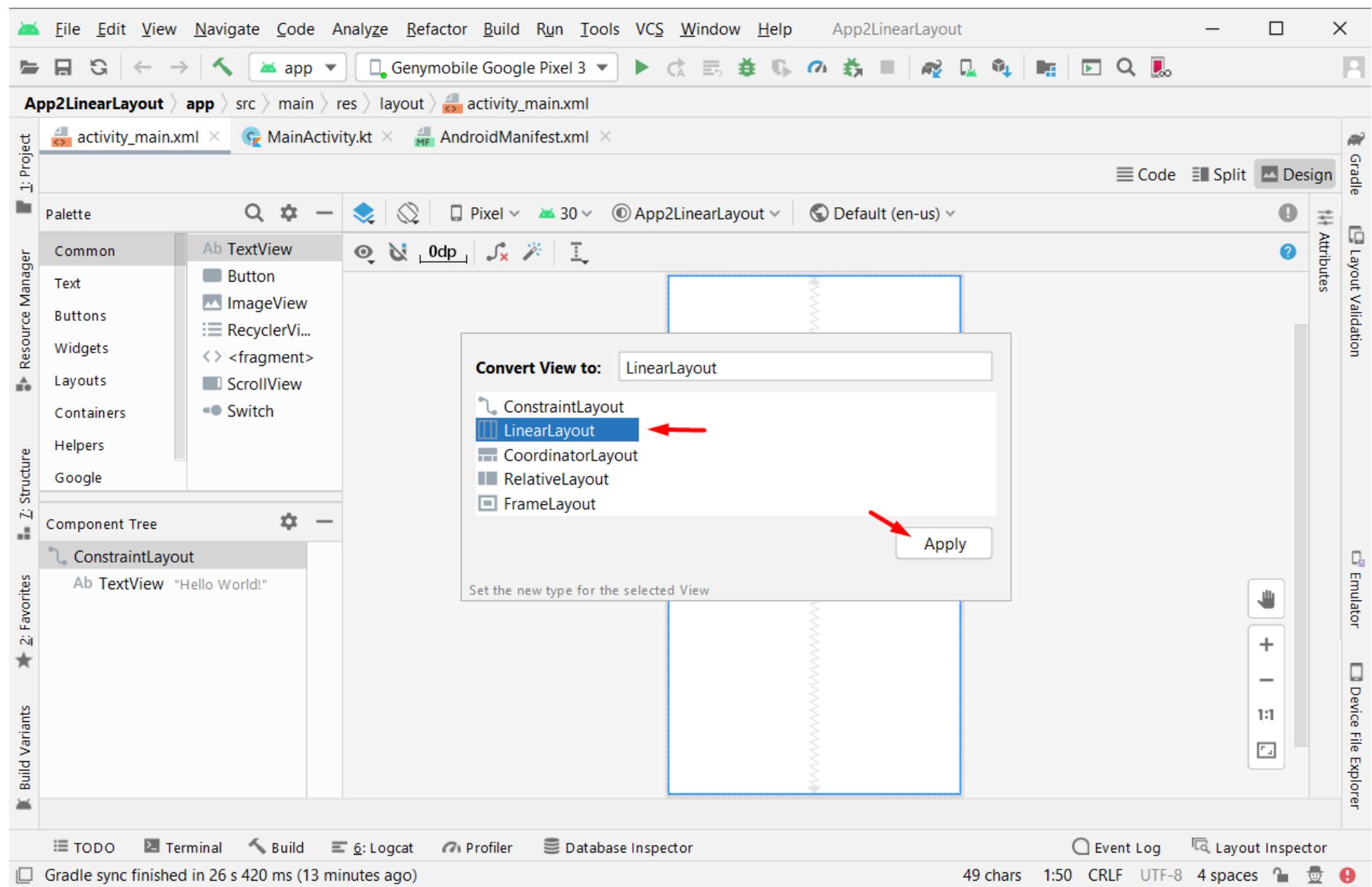


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador



# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

The screenshot shows the Android Studio interface with the following details:

- Top Bar:** File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help, App2LinearLayout.
- Toolbar:** Includes icons for file operations like Open, Save, Find, and Run, along with device and emulator selection.
- Project Tab:** Shows the project structure with files: activity\_main.xml, MainActivity.kt, and AndroidManifest.xml.
- Code Editor:** Displays the XML code for activity\_main.xml. The root element <LinearLayout> and its closing tag </LinearLayout> are highlighted with red boxes.
- Design Tab:** Shows a preview of the layout with a single TextView containing the text "Hello World!". This preview area is also highlighted with a red box.
- Toolbars:** On the right side, there are several toolbars: Palette, Attributes, Layout Validation, Emulator, and Device File Explorer.
- Bottom Navigation:** Includes tabs for TODO, Terminal, Build, Logcat, Profiler, Database Inspector, Event Log, and Layout Inspector.
- Status Bar:** Shows the message "Gradle sync finished in 26 s 420 ms (15 minutes ago)" and system information like "3:5 CRLF UTF-8 4 spaces".

**Annotations:** Several parts of the interface are annotated with red boxes:

- A red box highlights the "Split" tab in the top right toolbar.
- A red box highlights the "Hello World!" text in the Design preview.
- A red box highlights the opening <LinearLayout> tag in the code editor.
- A red box highlights the closing </LinearLayout> tag in the code editor.

**Text in the code editor:**

```
<?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">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!" />

</LinearLayout>
```

**Text in the bottom right (red box):**

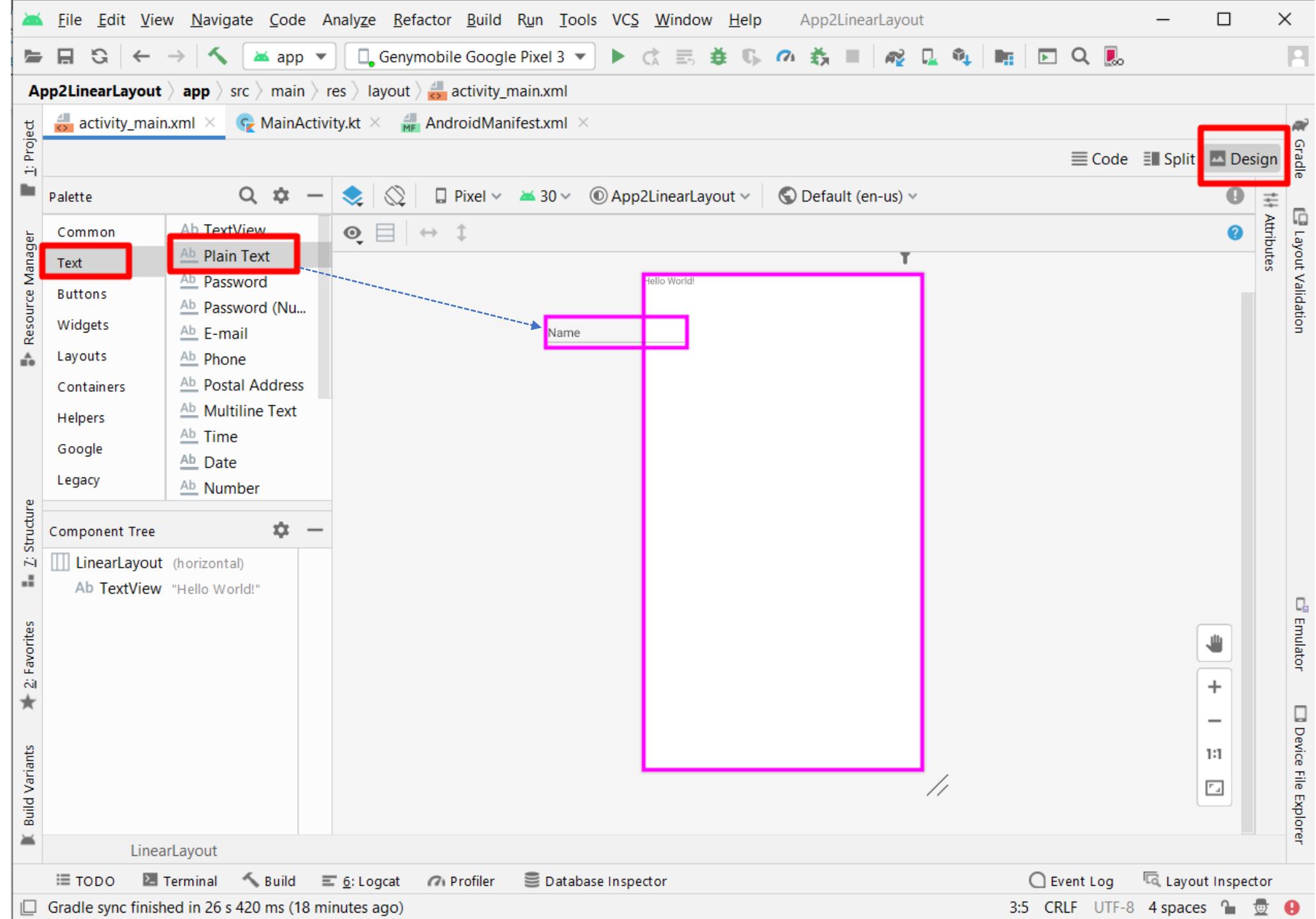
Este Layout tiene un comportamiento horizontal, es decir, cada componente que agregamos a él, este se coloca justo al lado del anterior

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

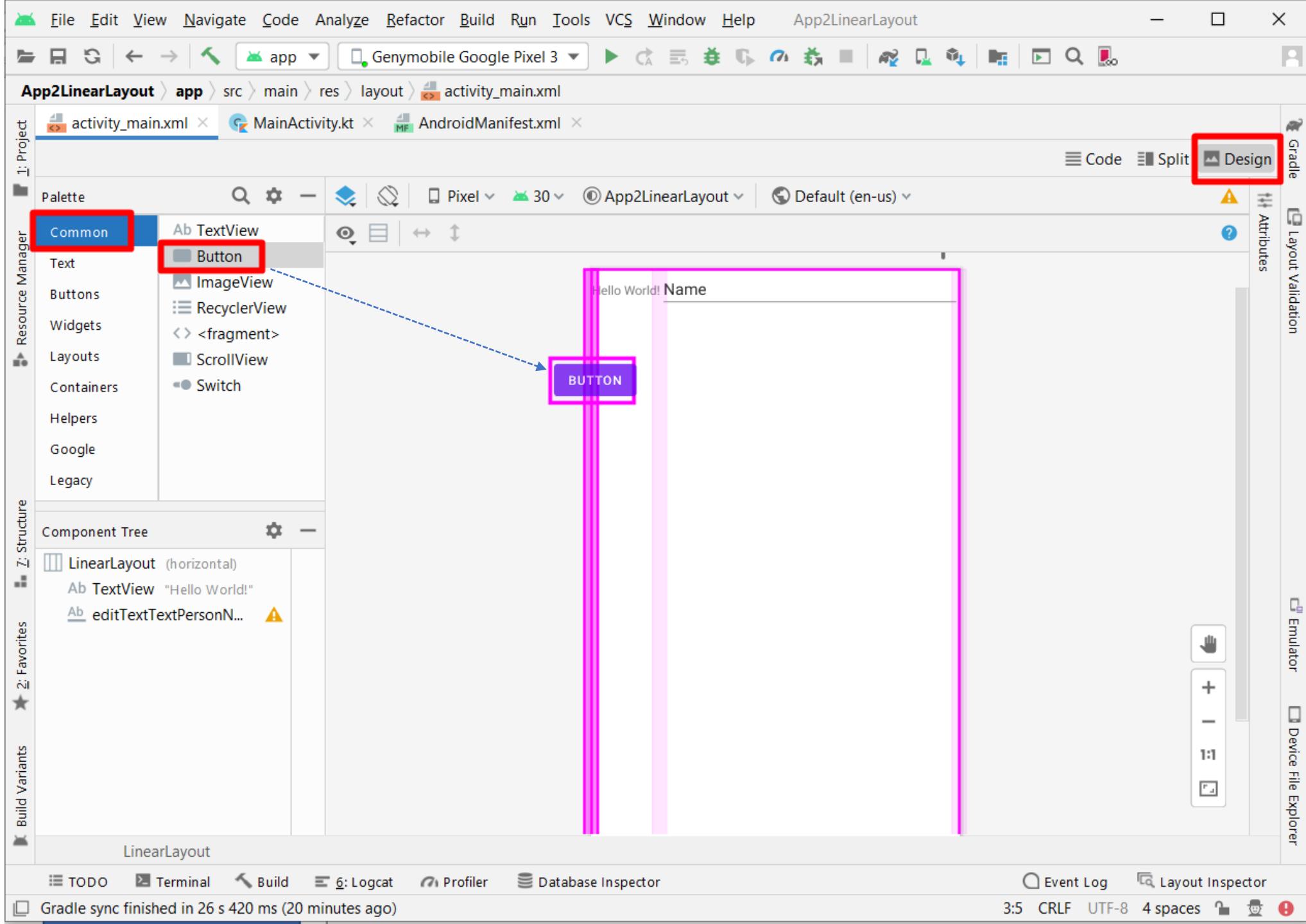


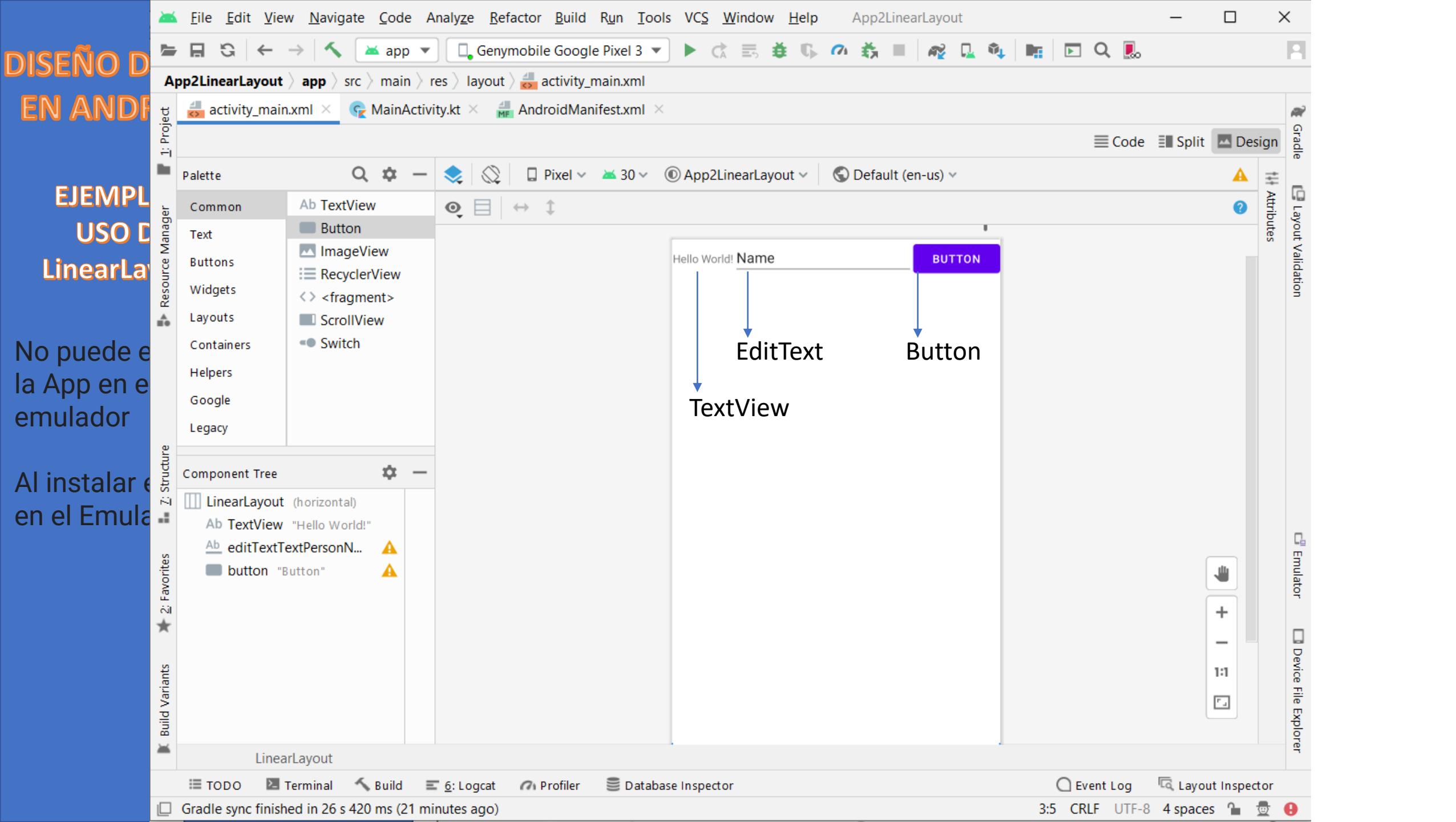
# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar  
la App en el  
emulador

Al instalar el APK  
en el Emulador



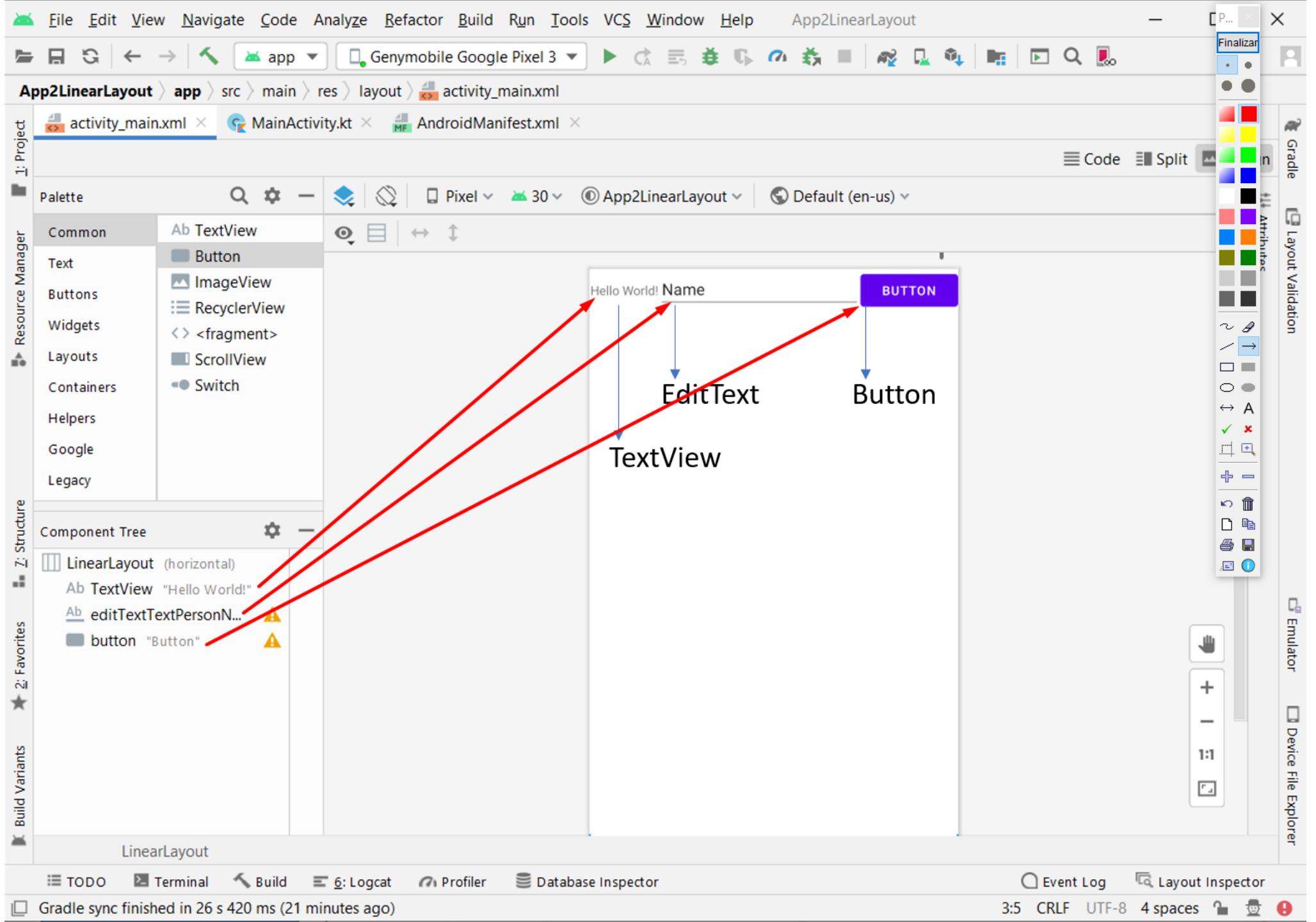


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

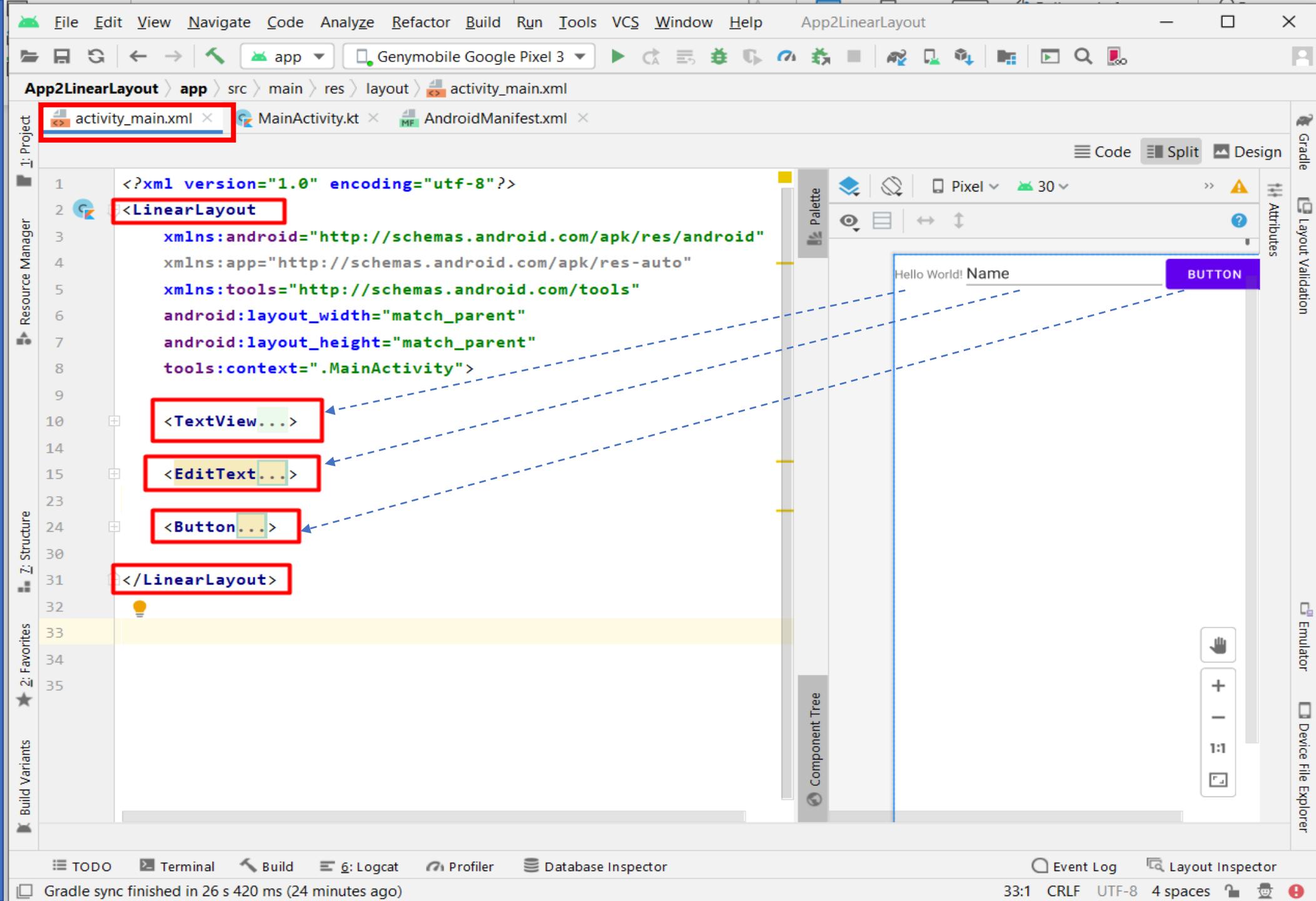


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

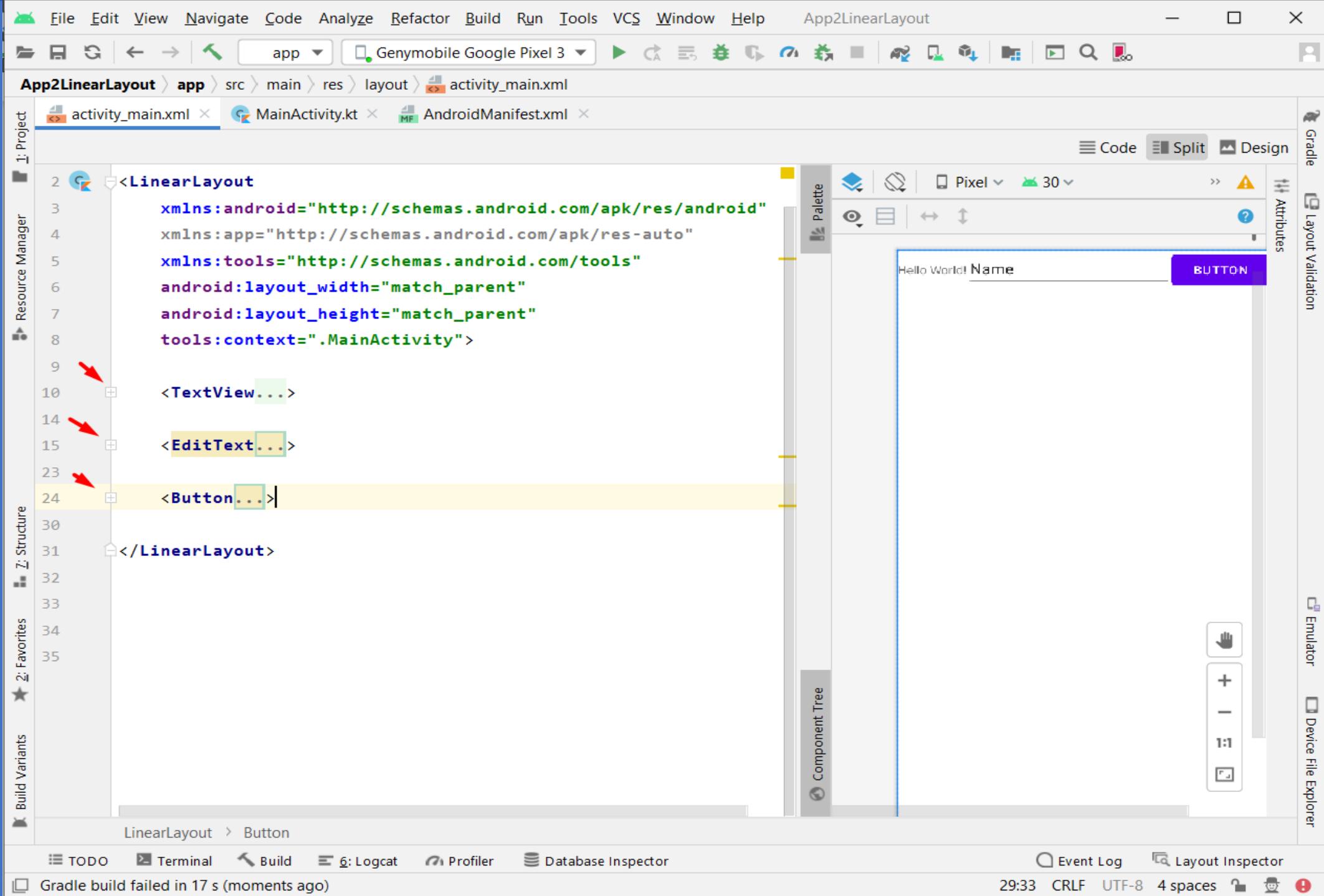


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

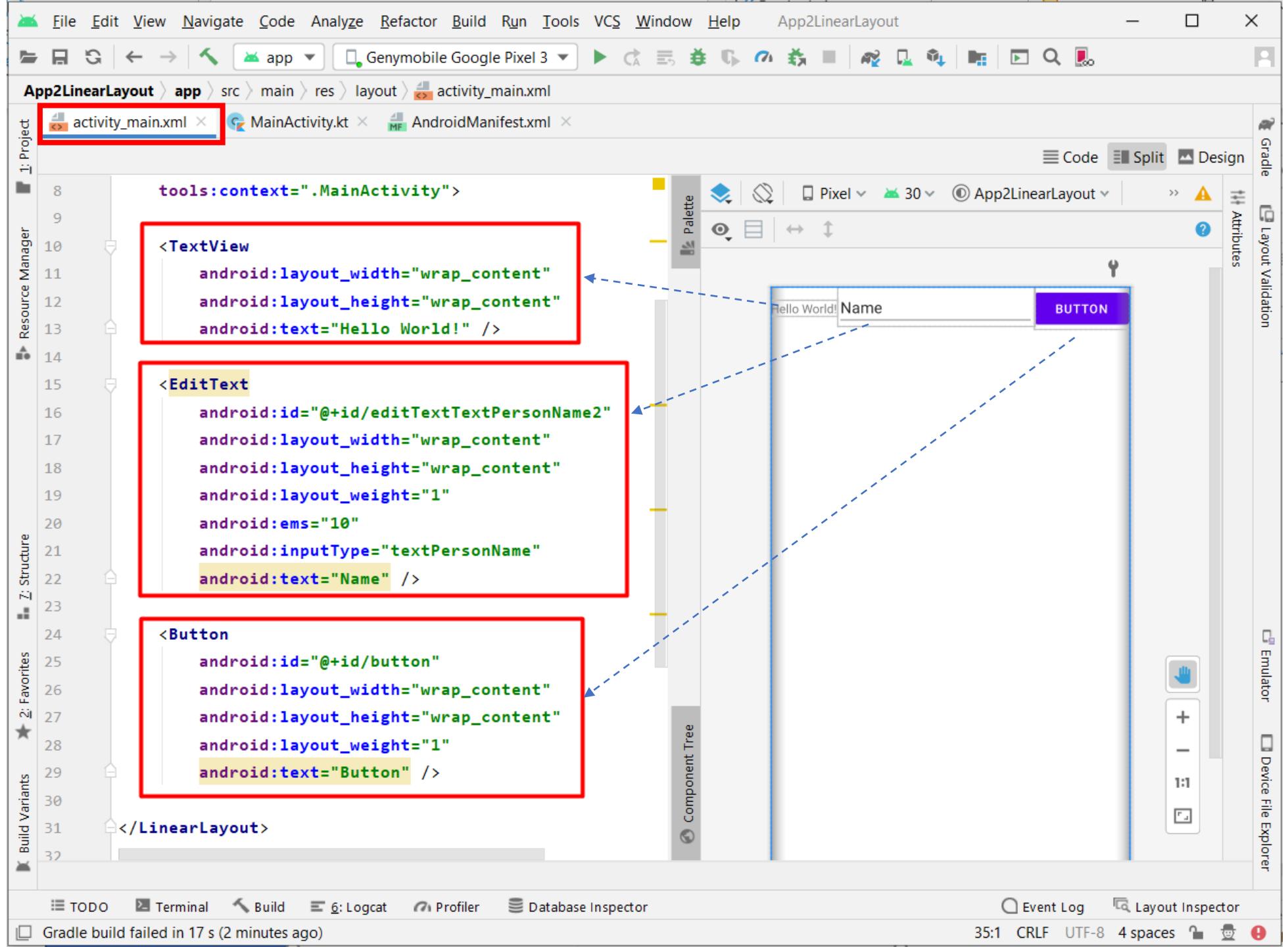


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

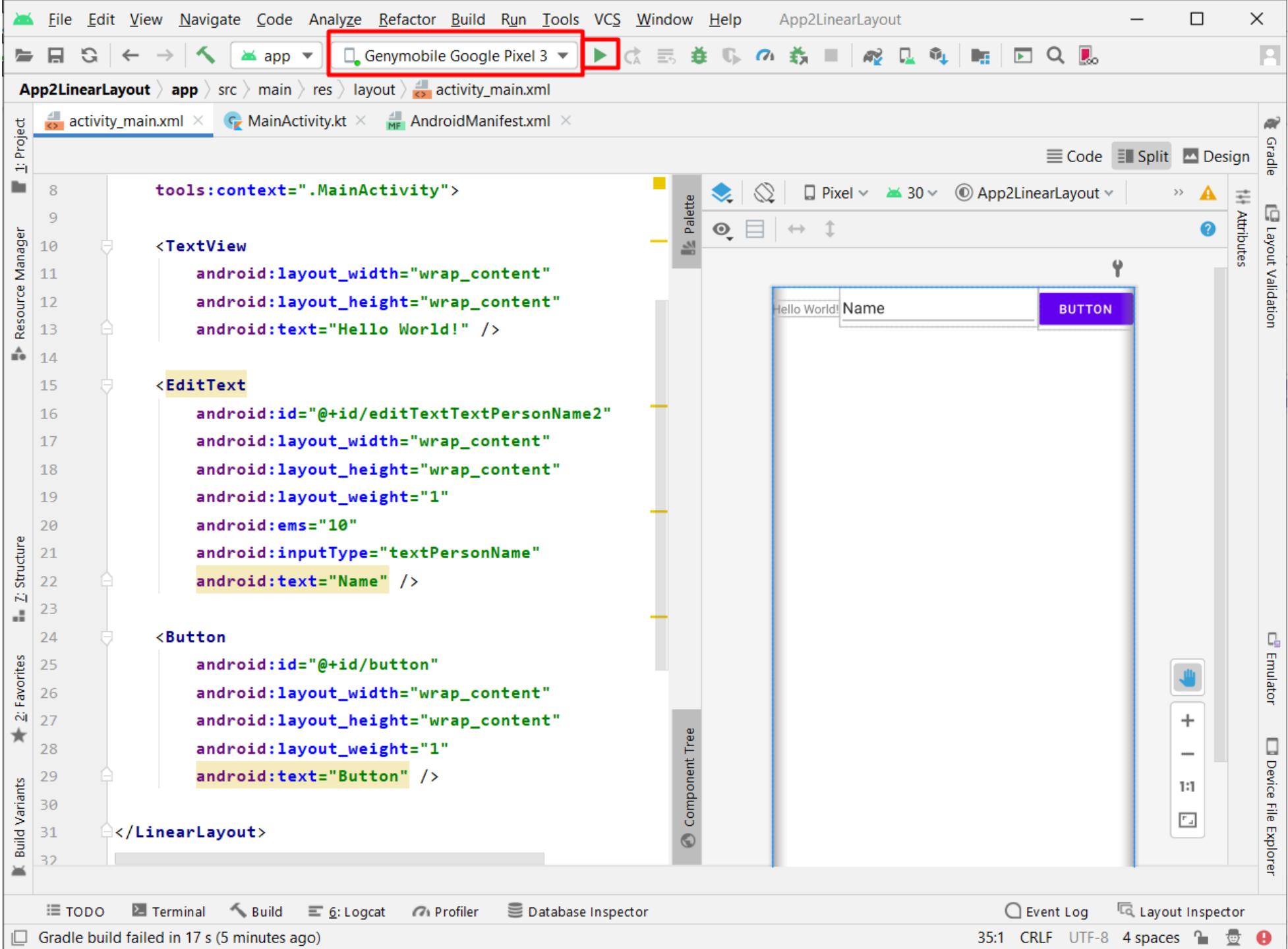


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

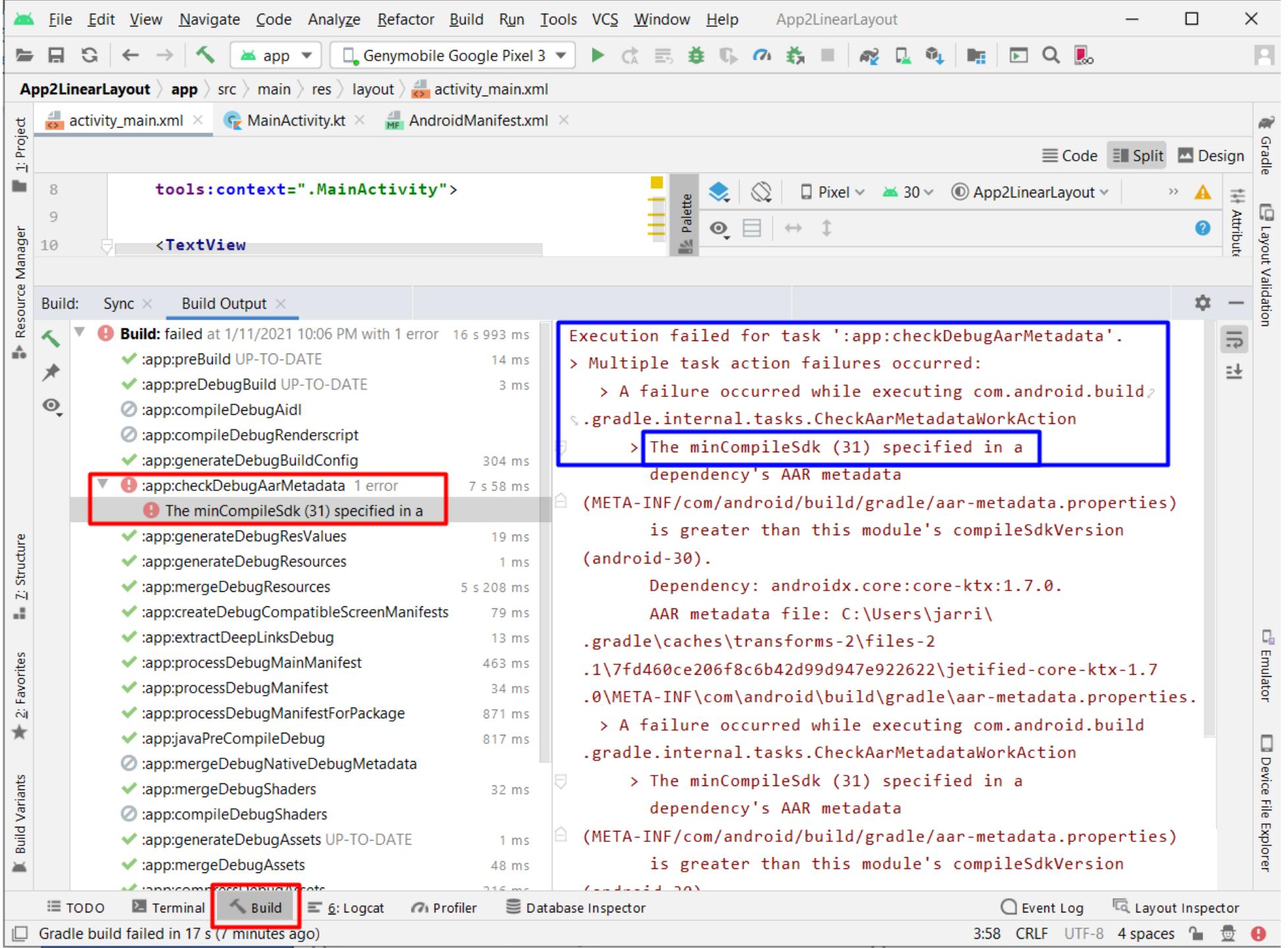


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador



# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

The screenshot shows the Android Studio interface with the following details:

- Project Structure:** The project is named "App2LinearLayout". The "app" module is selected.
- Resource Manager:** Shows the app directory structure, including manifests, Java files (MainActivity), and resources (res).
- Gradle Scripts:** The "build.gradle (Module: App2LinearLayout.app)" file is open.
- Code Editor:** The build.gradle file contains configuration code. A red box highlights the "resolutionStrategy" block:

```
defaultConfig {
    configurations.all {
        resolutionStrategy { force 'androidx.core:core-ktx:1.6.0' }
    }
}
```
- Toolbars and Status:** The top bar shows the menu (File, Edit, View, etc.) and the current file (build.gradle). The bottom bar includes tabs for TODO, Terminal, Build, Logcat, Profiler, Database Inspector, Event Log, and Layout Inspector. A status message at the bottom left says "Gradle build failed in 17 s (12 minutes ago)".
- Right Panel:** Shows a warning about Gradle files changing since last sync, with options to "Sync Now", "Ok, apply suggestion!", or "Ignore these changes".

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

The screenshot shows the Android Studio interface with the following details:

- File Bar:** File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help.
- Toolbar:** Includes icons for file operations like Open, Save, and Build, along with navigation and search tools.
- Project Structure:** Shows the project tree under "app".
- Build Variants:** Shows "Favorites" selected.
- Code Editor:** Displays the `build.gradle` file for the app module. A message at the top says: "Gradle files have changed since last project sync. A project sync may be necessary for Sync Now or Ignore these changes". The code includes:

```
plugins {
    id 'com.android.application'
    id 'kotlin-android'

}

android {
    compileSdkVersion 30
    buildToolsVersion "30.0.3"
}

applicationId "co.gov.misiontic.ciclo4"
minSdkVersion 17
targetSdkVersion 30
versionCode 1
versionName "1.0"

testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"

buildTypes {
    release {

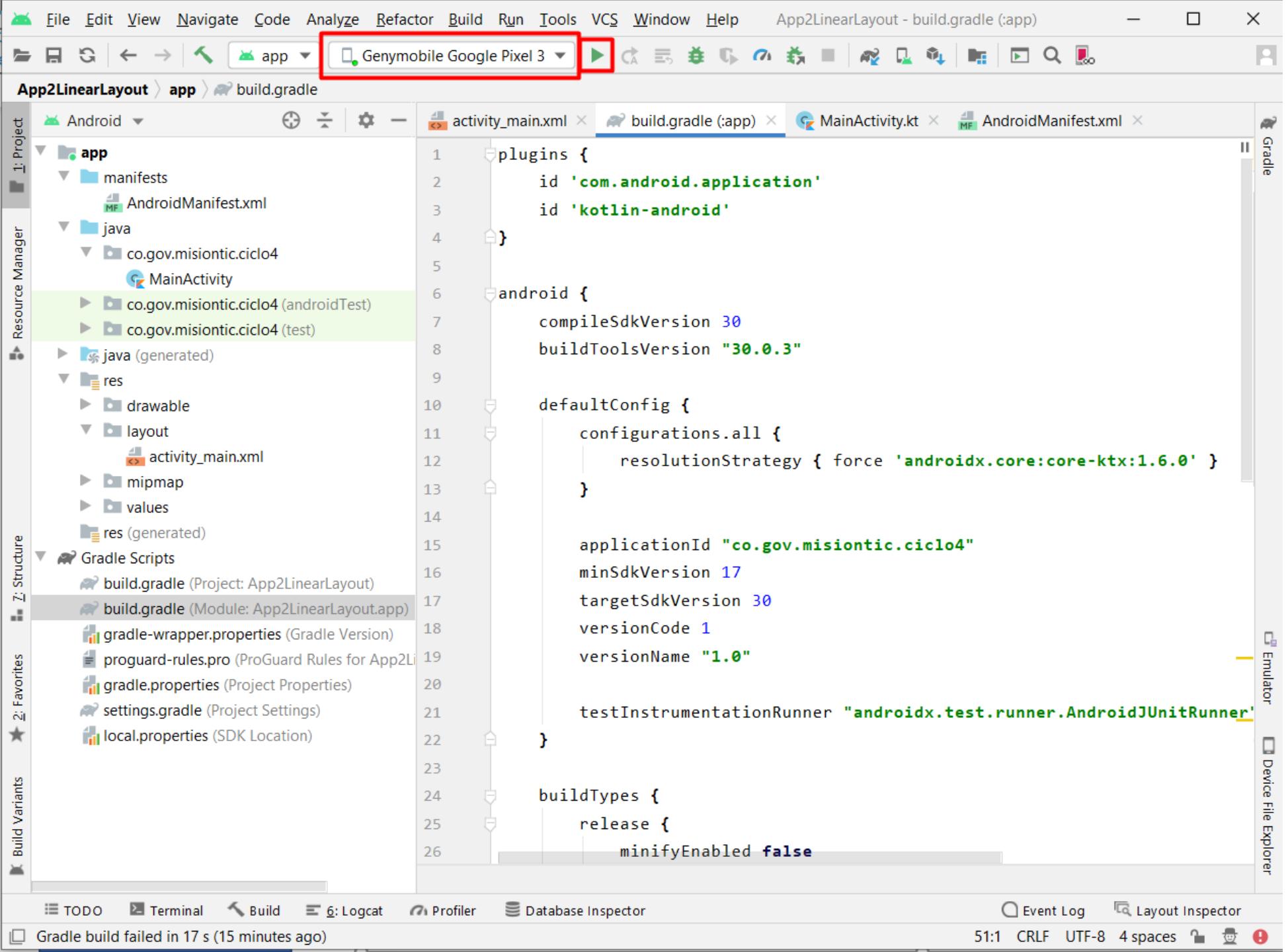
```
- Progress Dialog:** A modal dialog titled "Importing 'App2LinearLayout' Gradle project" with "Gradle: Build model..." and "Background" buttons, highlighted with a red box.
- Bottom Navigation:** TODO, Terminal, Build, Logcat, Profiler, Database Inspector.
- Bottom Status:** Event Log, Layout Inspector.
- Bottom Footer:** Gradle build failed in 17 s (14 minutes ago), 51:1 CRLF, UTF-8, 4 spaces.

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador



# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

The screenshot shows the Android Studio interface with the following details:

- File Bar:** File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help, App2LinearLayout.
- Toolbar:** Includes icons for file operations, navigation, and device selection (Genymobile Google Pixel 3).
- Project Structure:** Shows the project tree under "app": manifests (AndroidManifest.xml), java (co.gov.misiontic.ciclo4, MainActivity), and resources (activity\_main.xml).
- Build Output:** Displays the build log with the message "BUILD SUCCESSFUL in 1m 26s".
- Bottom Status:** Success: Operation succeeded (moments ago).
- Bottom Right:** Event Log and Layout Inspector tabs.

```
plugins {
    id 'com.android.application'
    id 'kotlin-android'
```

The build log shows the following tasks:

- > Task :app:mergeDebugJniLibFolders UP-TO-DATE
- > Task :app:checkDebugDuplicateClasses
- > Task :app:validateSigningDebug
- > Task :app:checkDebugAarMetadata
- > Task :app:processDebugResources
- > Task :app:mergeDebugNativeLibs
- > Task :app:mergeExtDexDebug
- > Task :app:compileDebugKotlin
- > Task :app:compileDebugJavaWithJavac
- > Task :app:compileDebugSources
- > Task :app:dexBuilderDebug
- > Task :app:stripDebugDebugSymbols NO-SOURCE
- > Task :app:mergeDebugJavaResource
- > Task :app:mergeDexDebug
- > Task :app:packageDebug
- > Task :app:assembleDebug

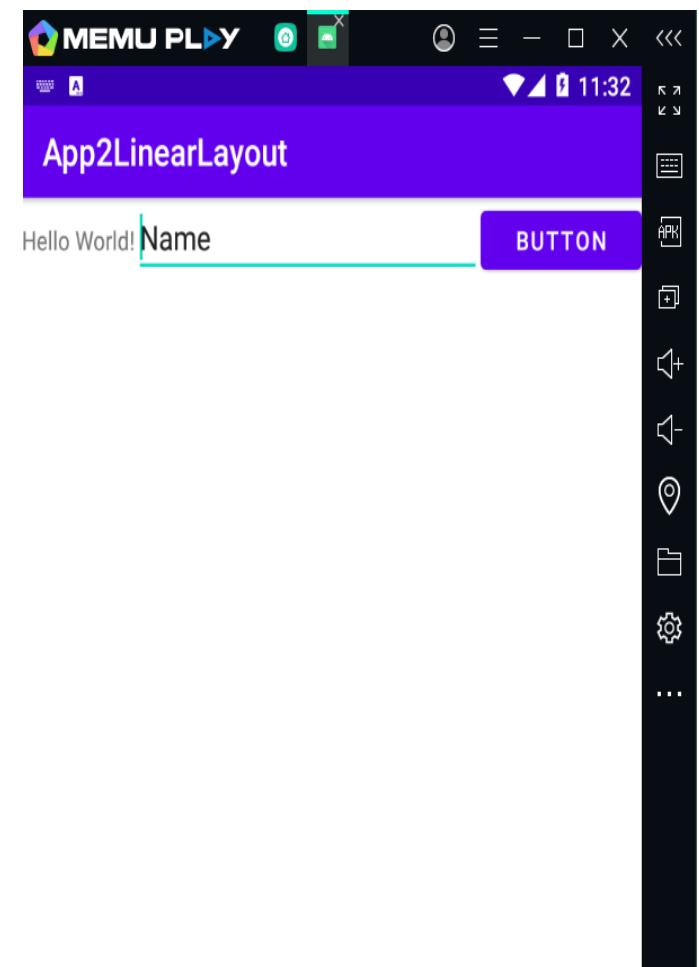
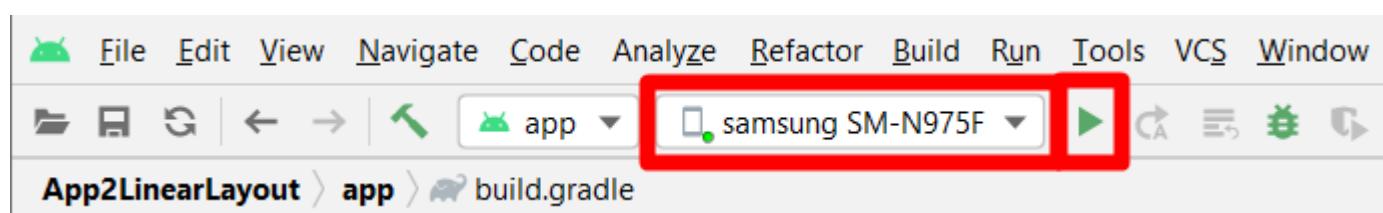
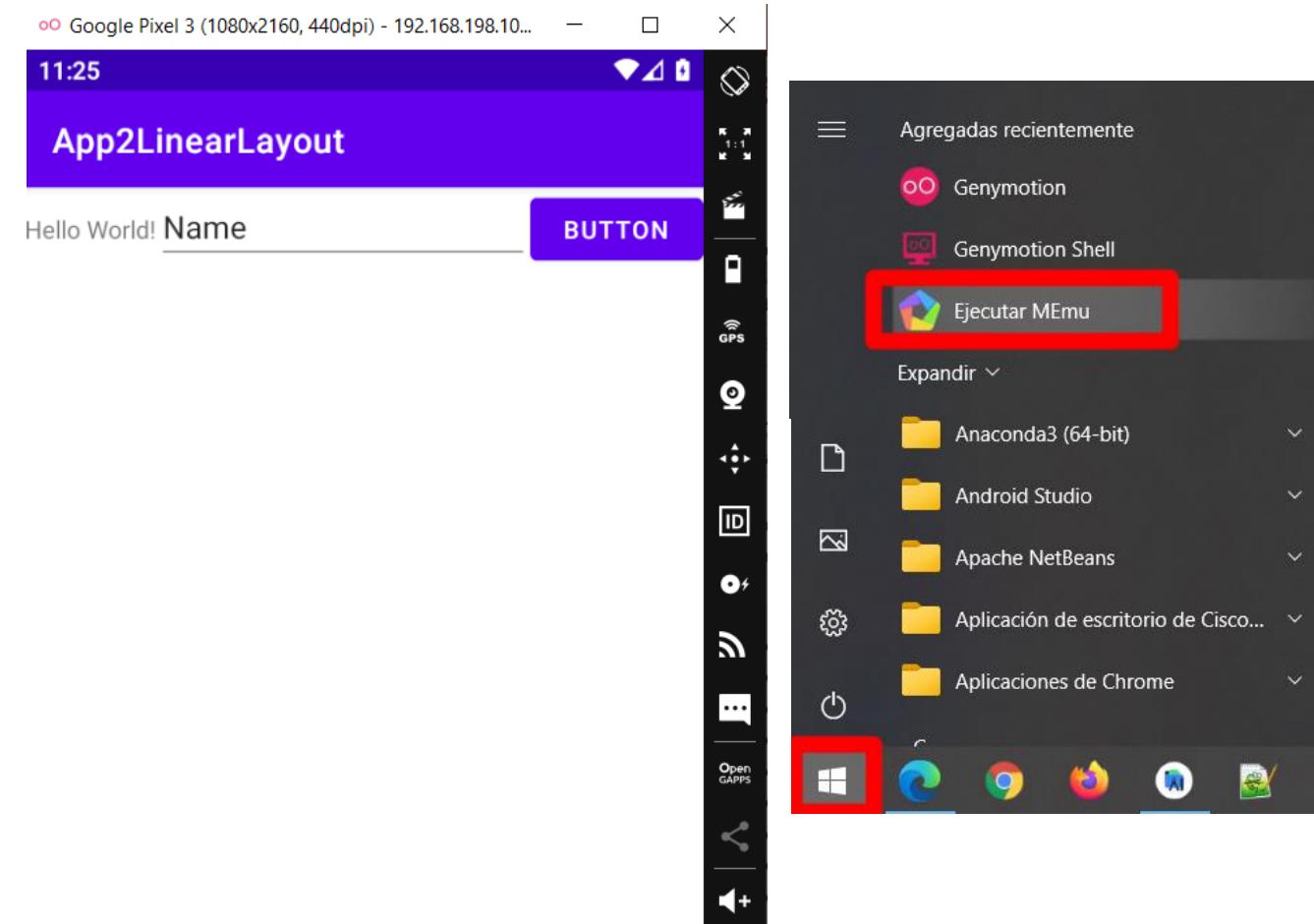
A green box highlights the message "BUILD SUCCESSFUL in 1m 26s". A red box highlights the message "26 actionable tasks: 15 executed, 11 up-to-date". A blue box highlights the message "Gradle build finished in 1 m 28 s 280 ms".

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

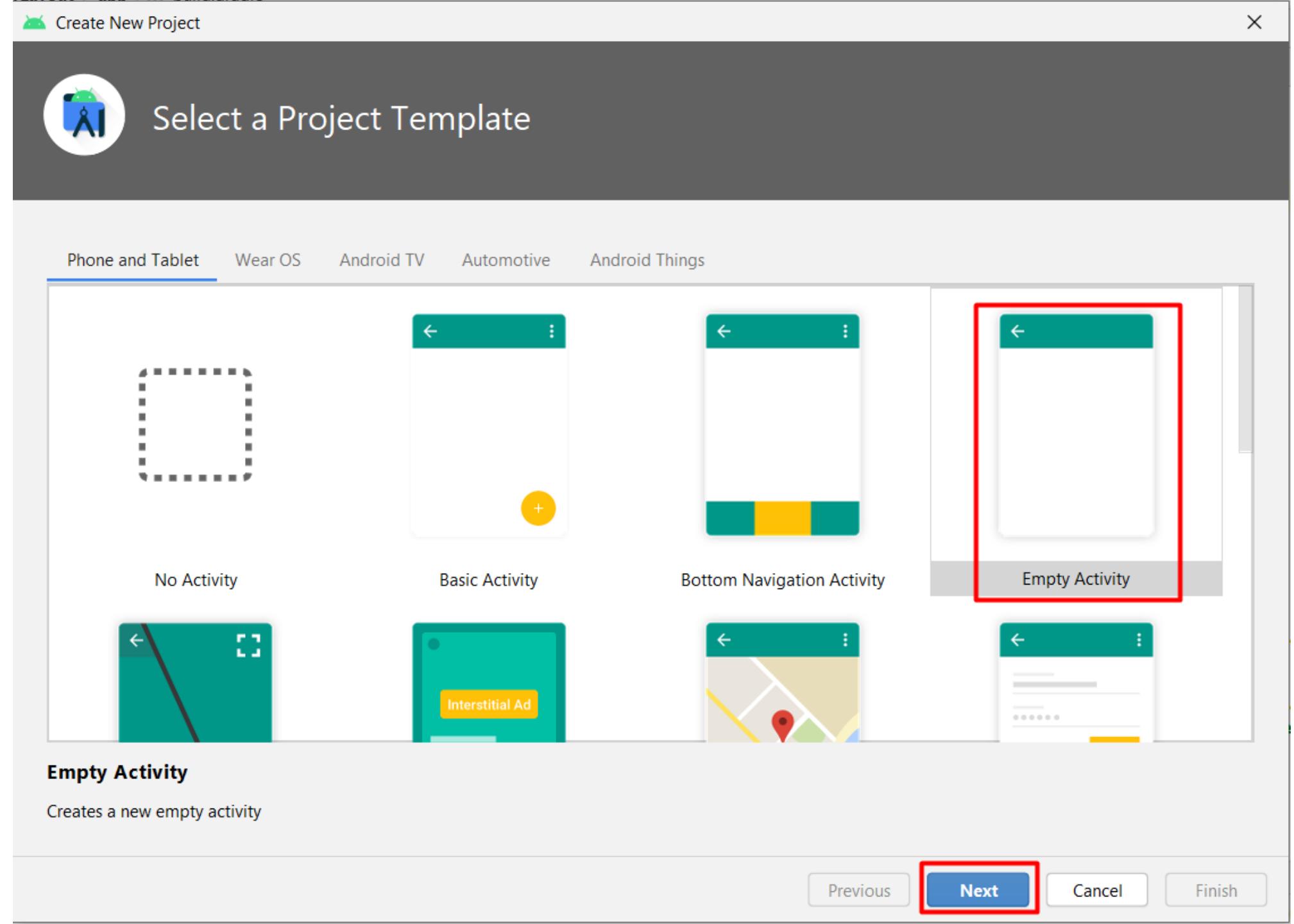


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

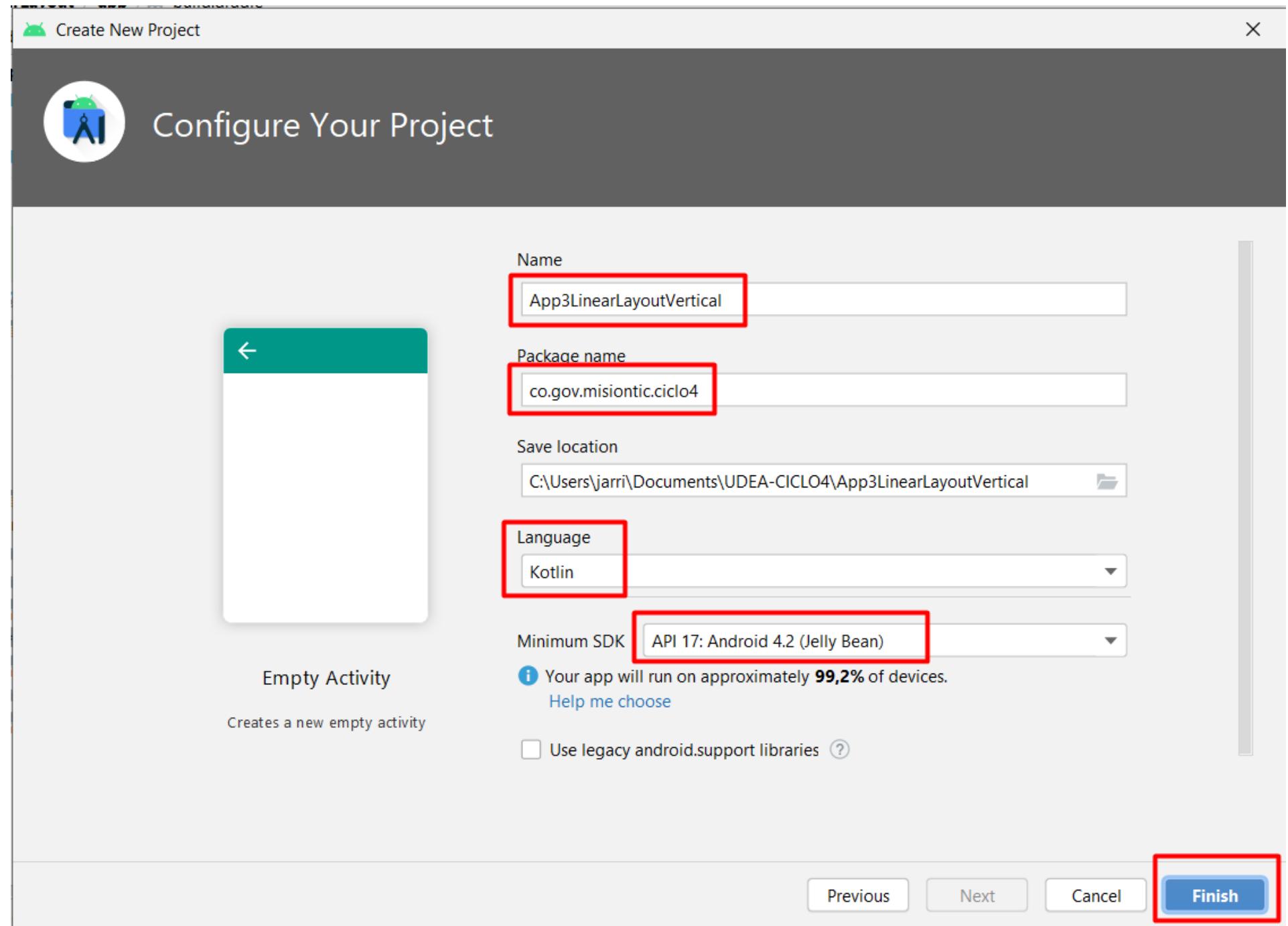
No puede ejecutar  
la App en el  
emulador

Al instalar el APK  
en el Emulador



# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

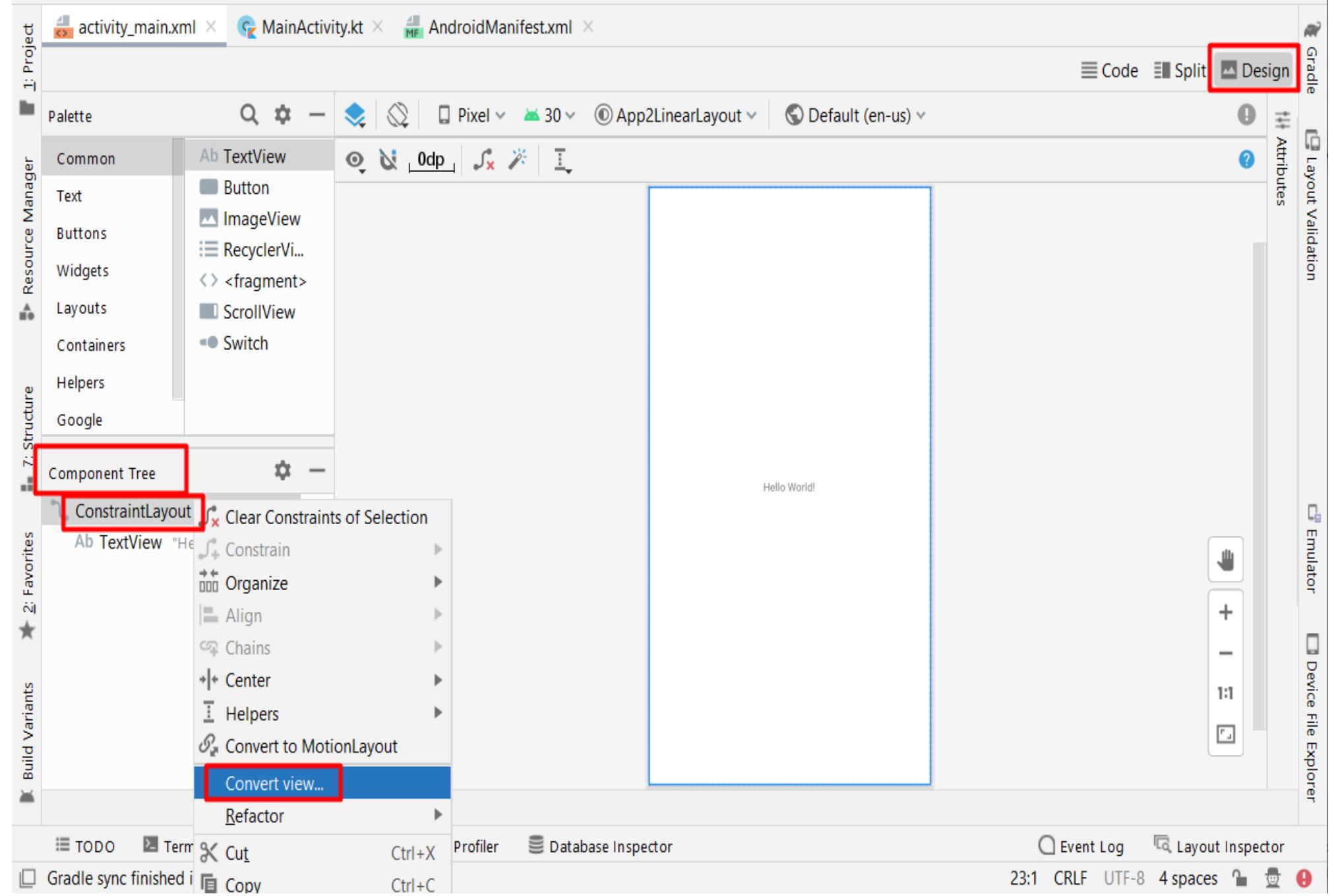


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

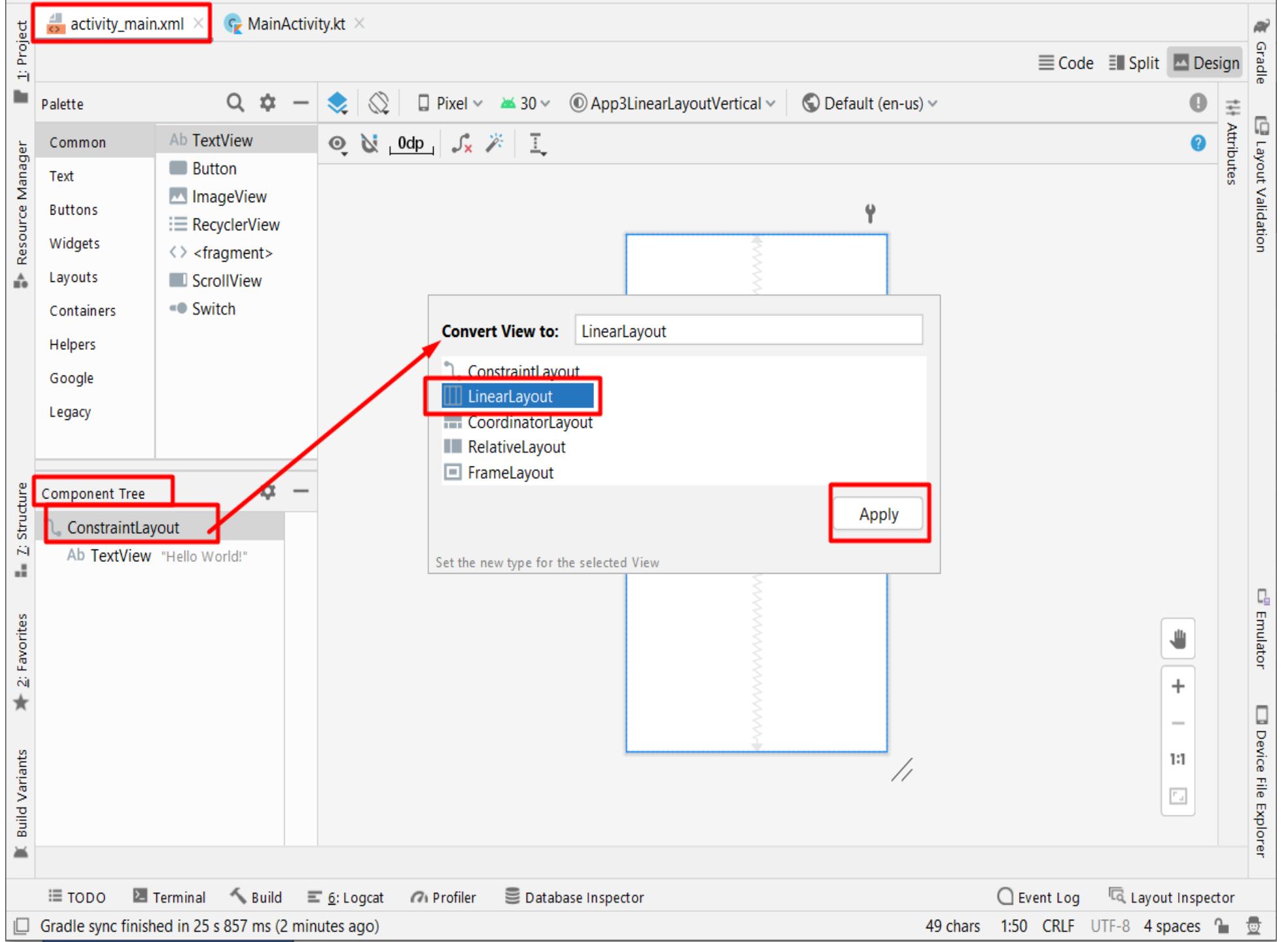
No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador



# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

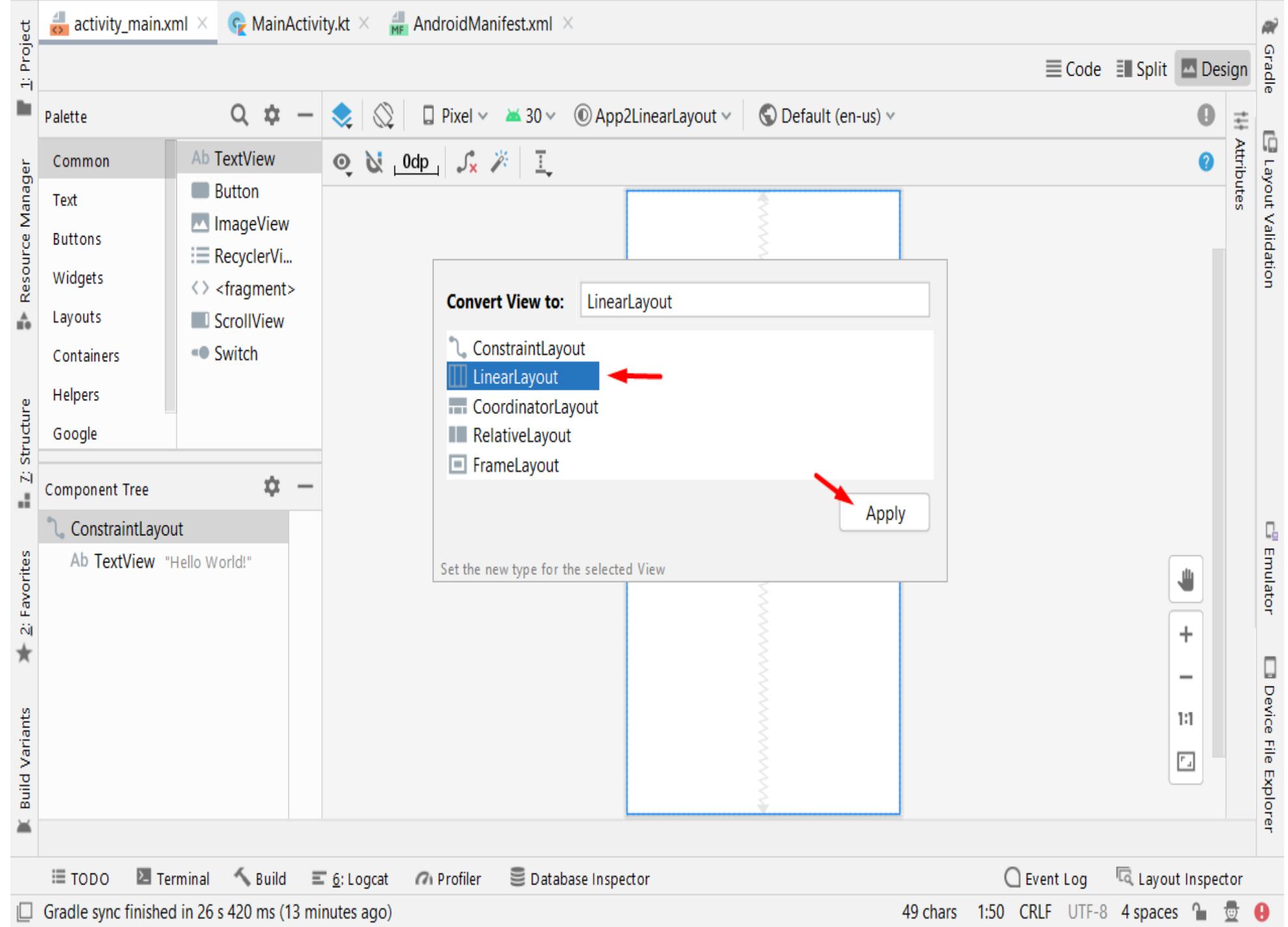


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

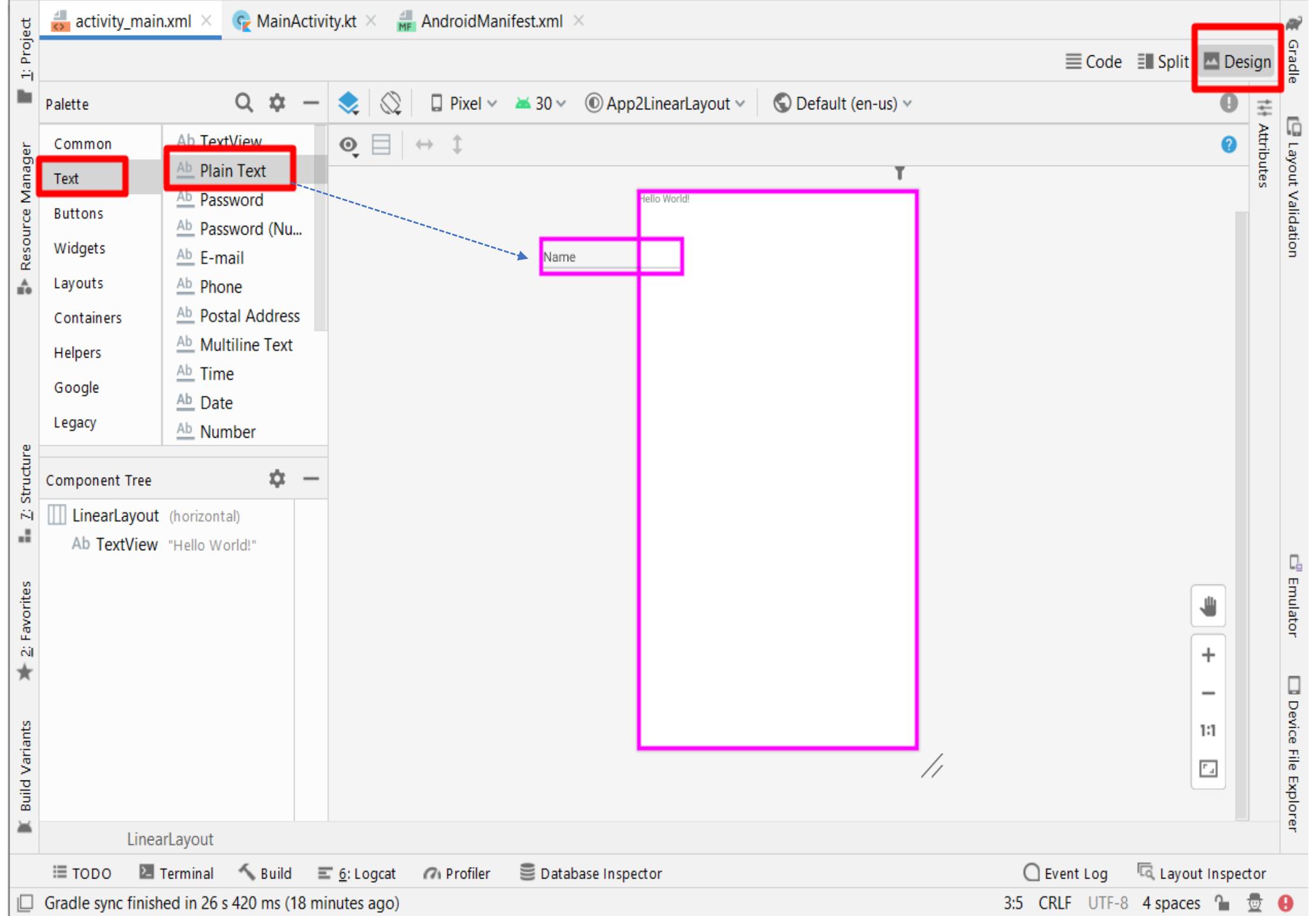


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador

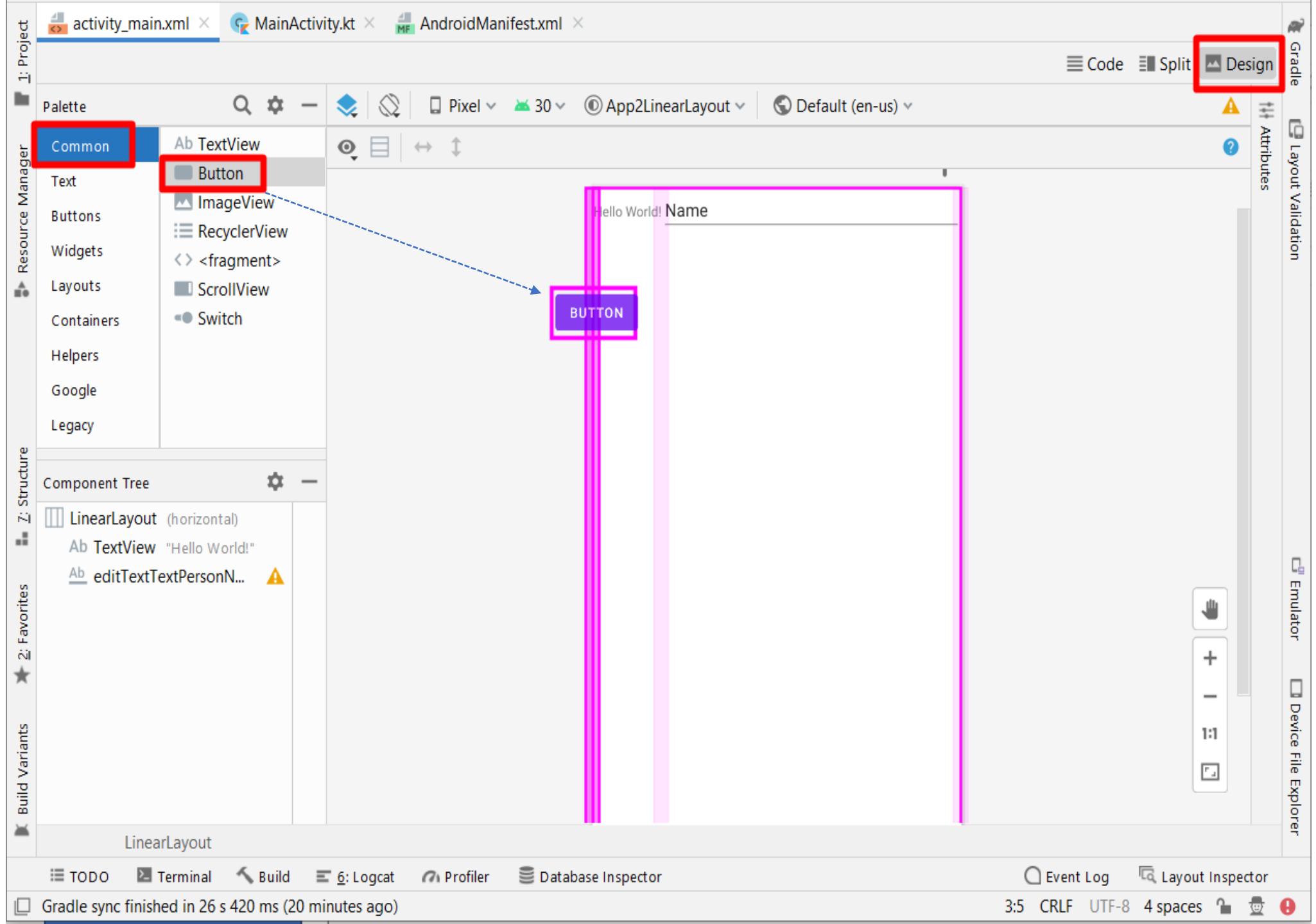


# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout

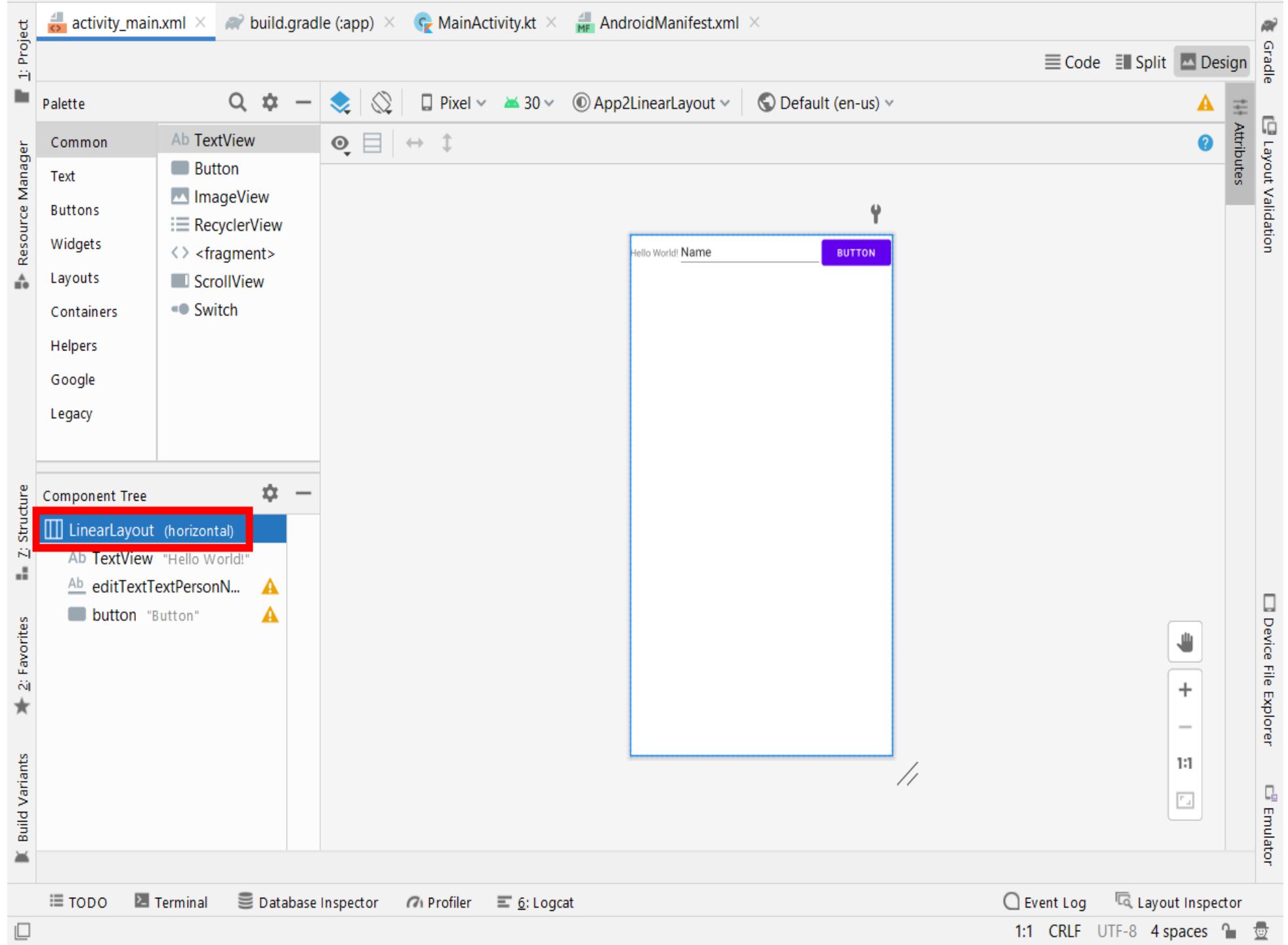
No puede ejecutar la App en el emulador

Al instalar el APK en el Emulador



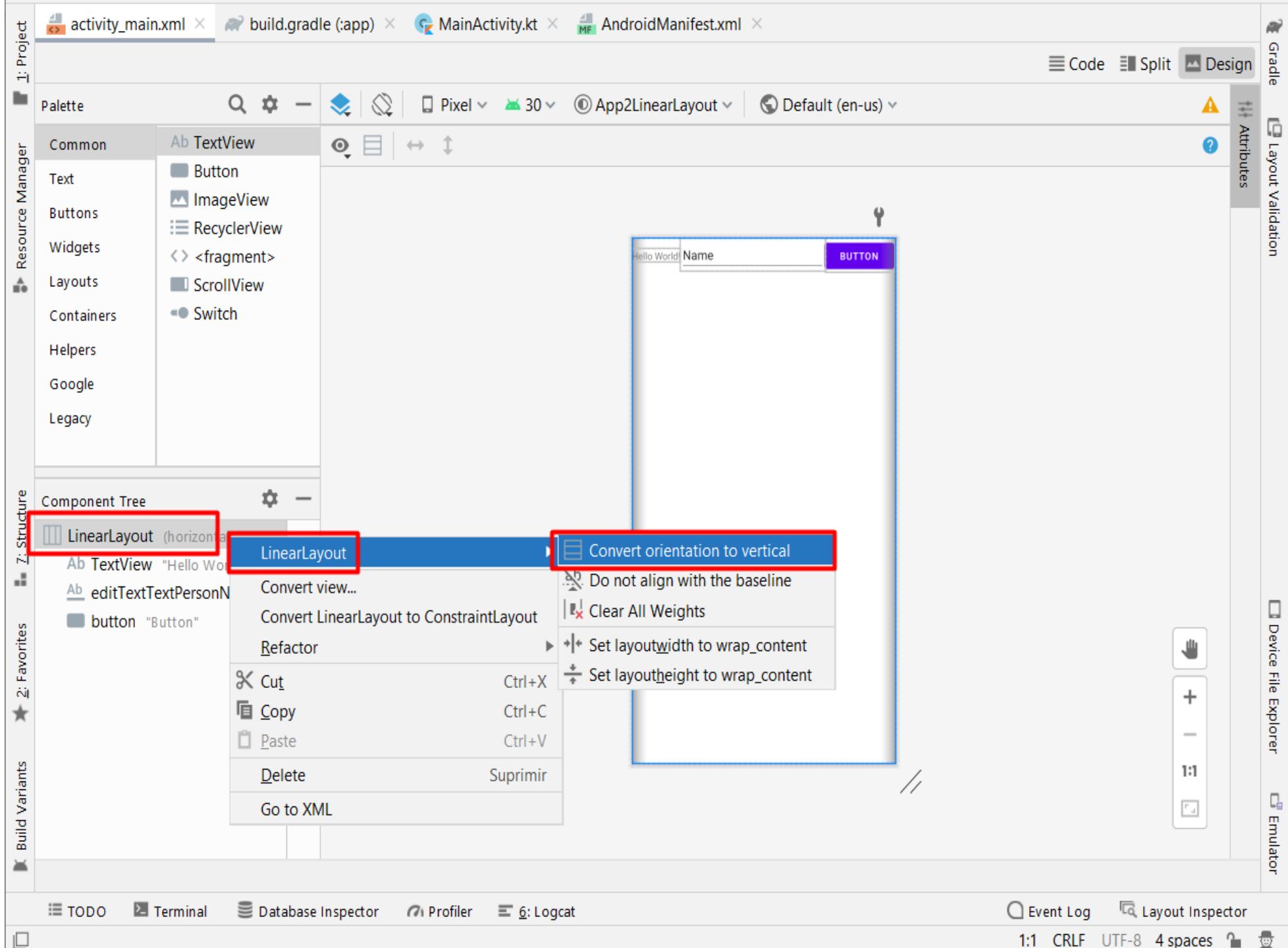
# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical



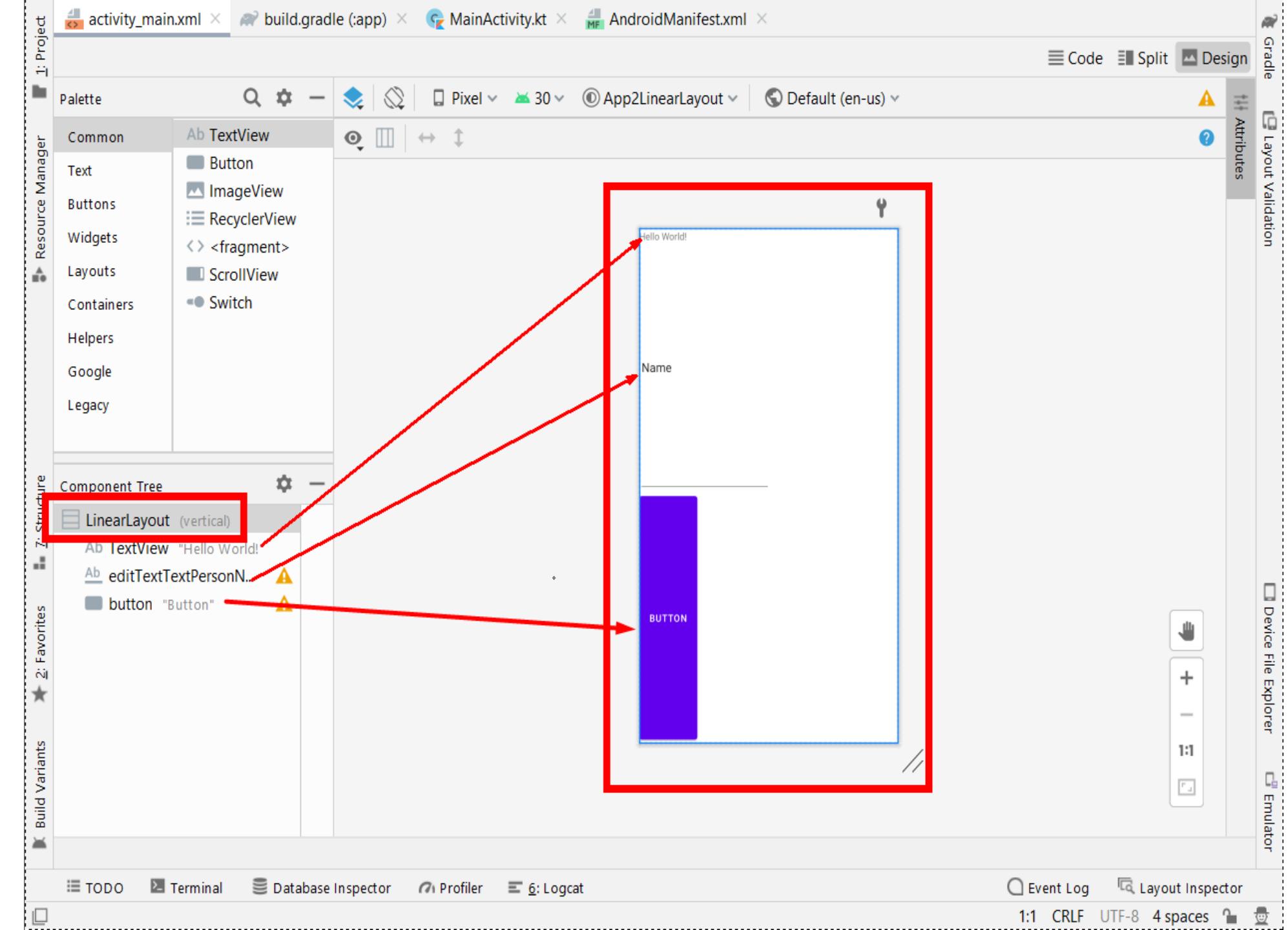
# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical



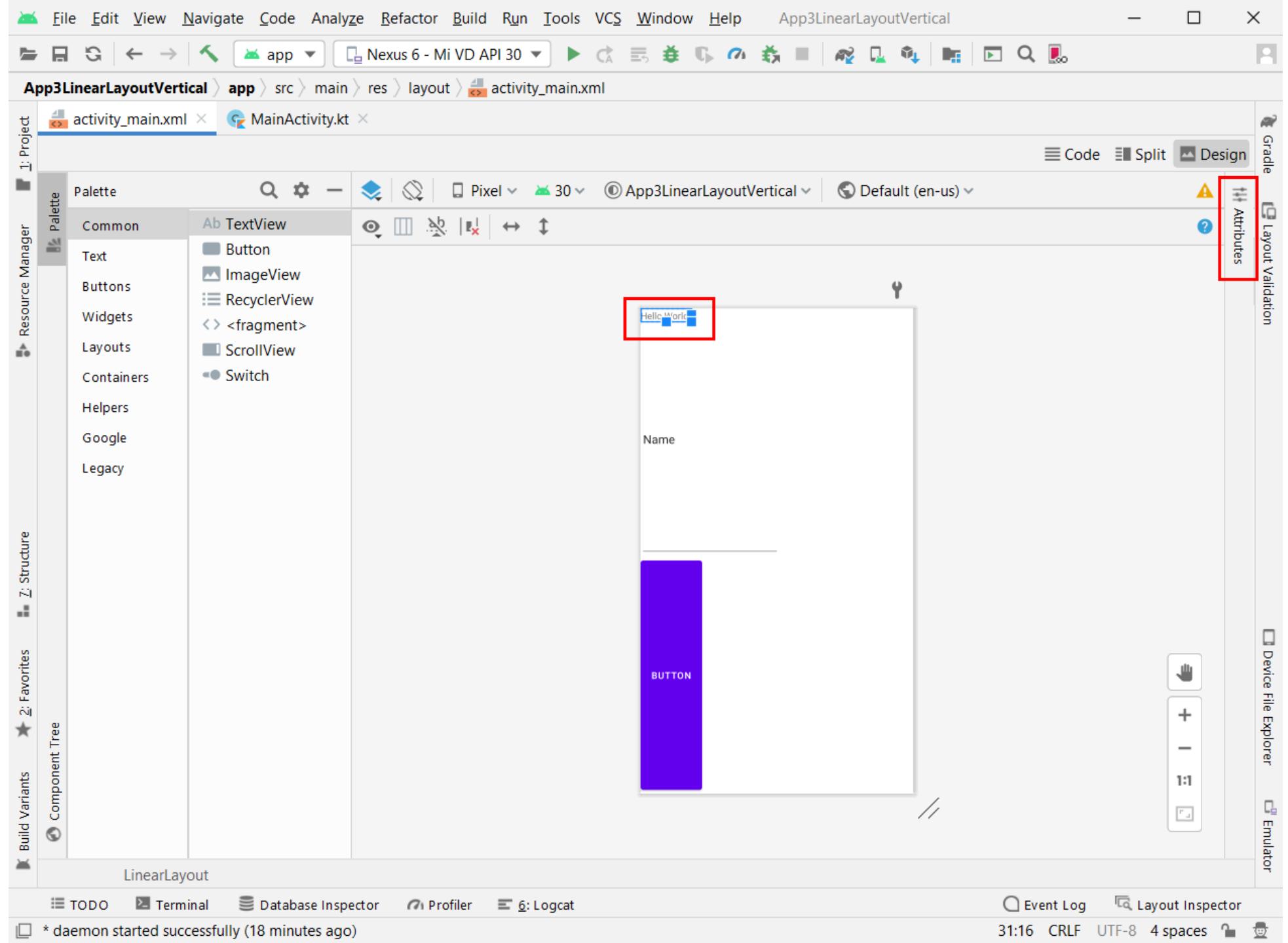
# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical



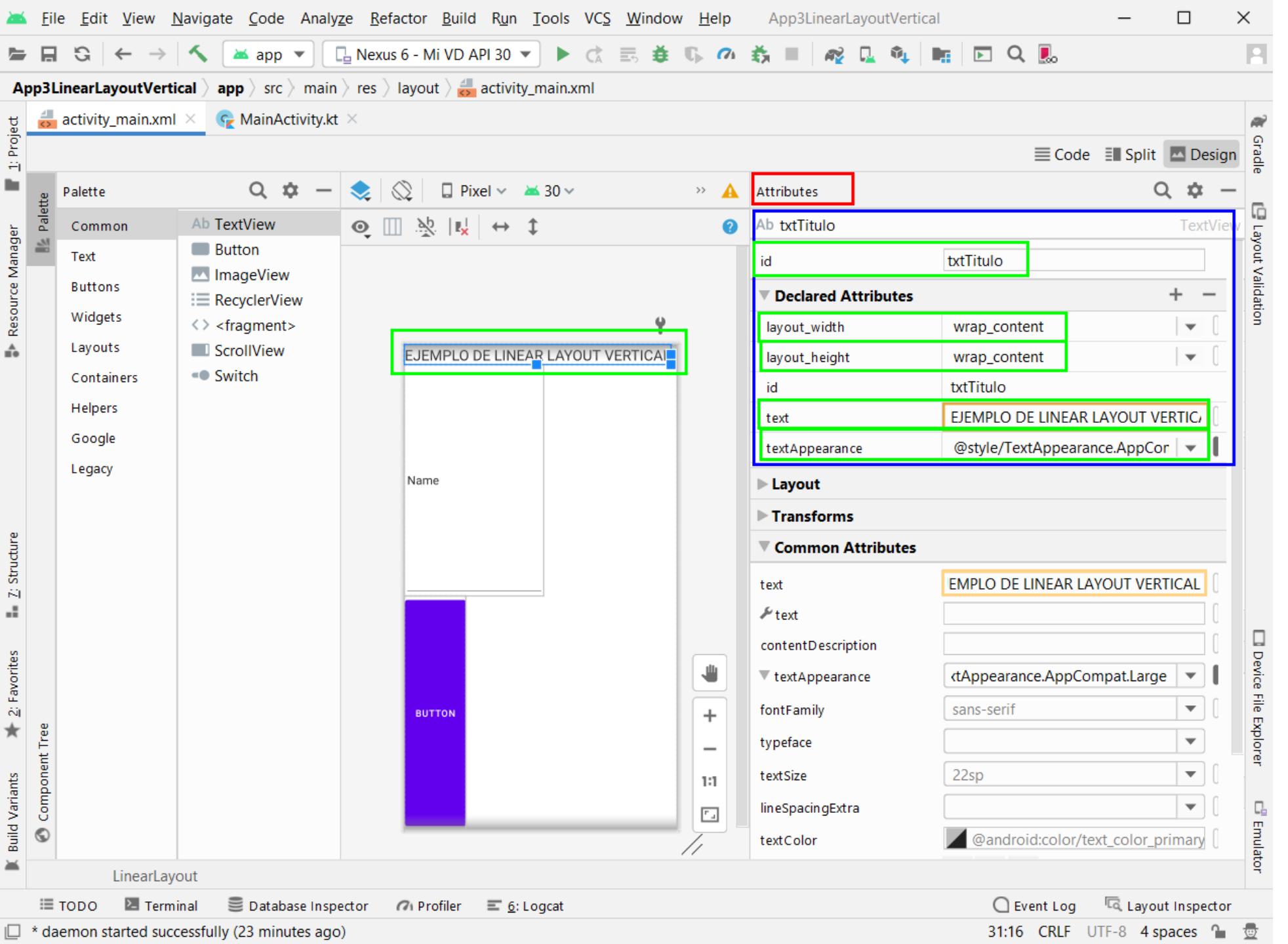
# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical



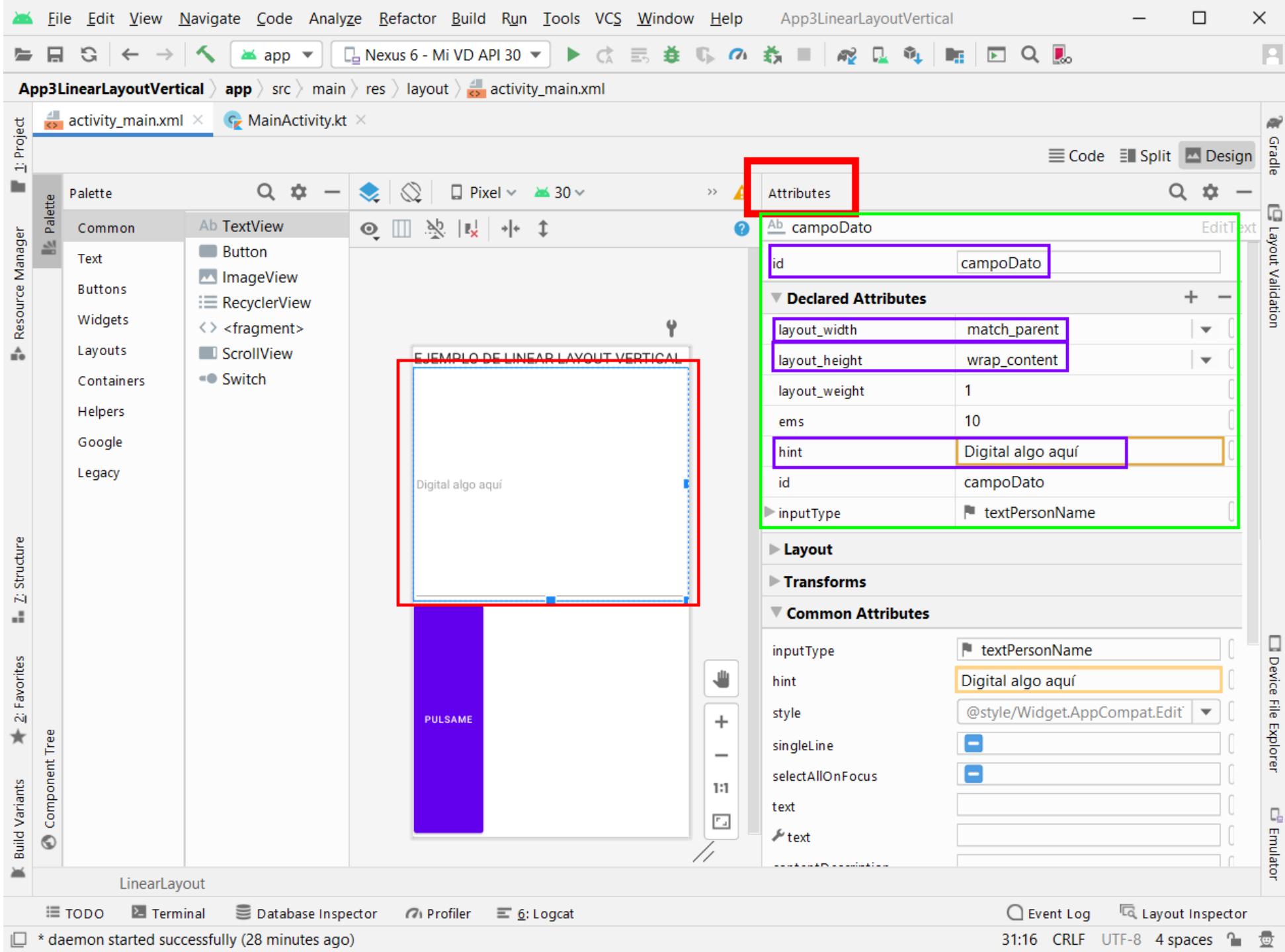
# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical



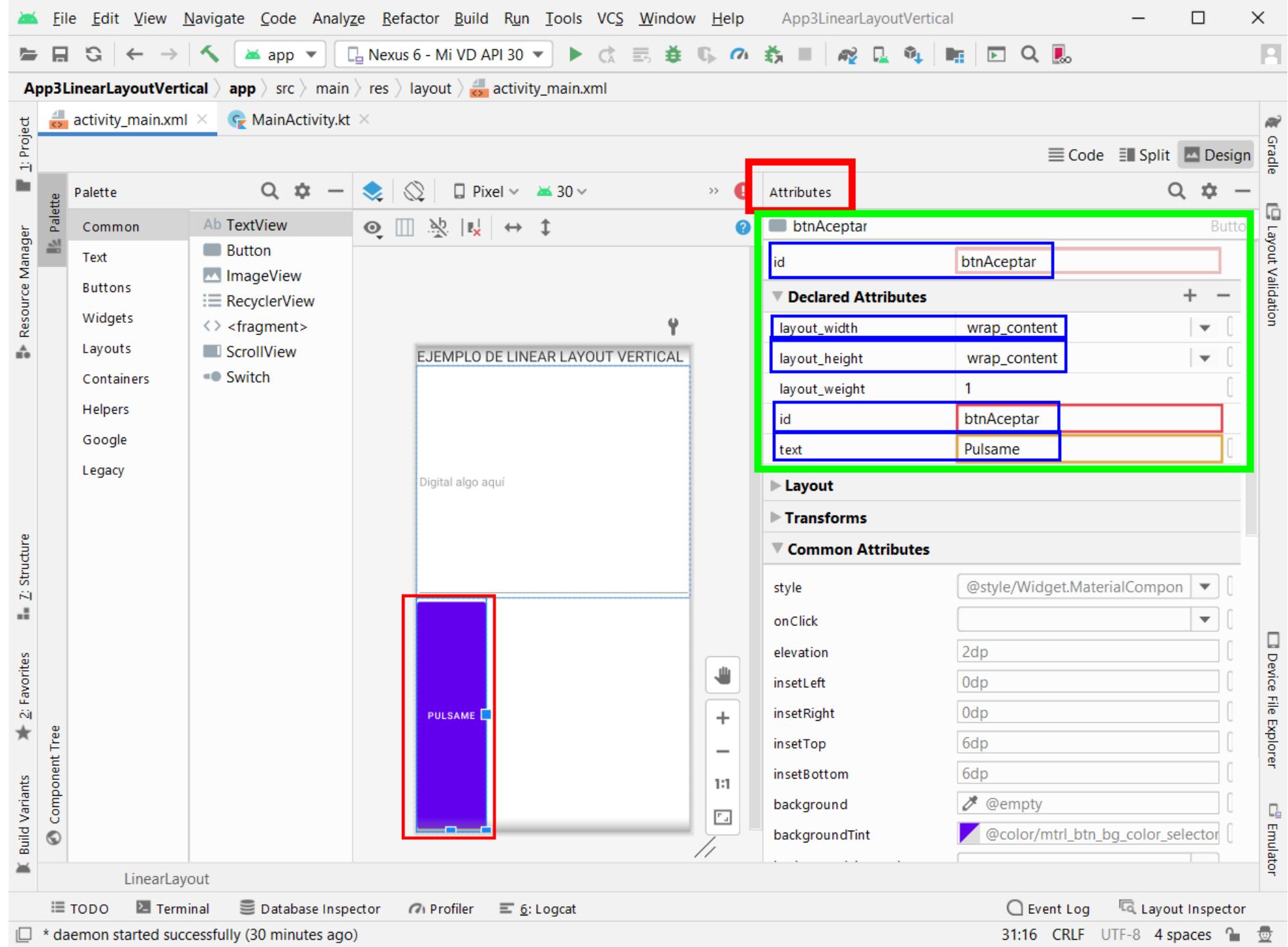
# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical



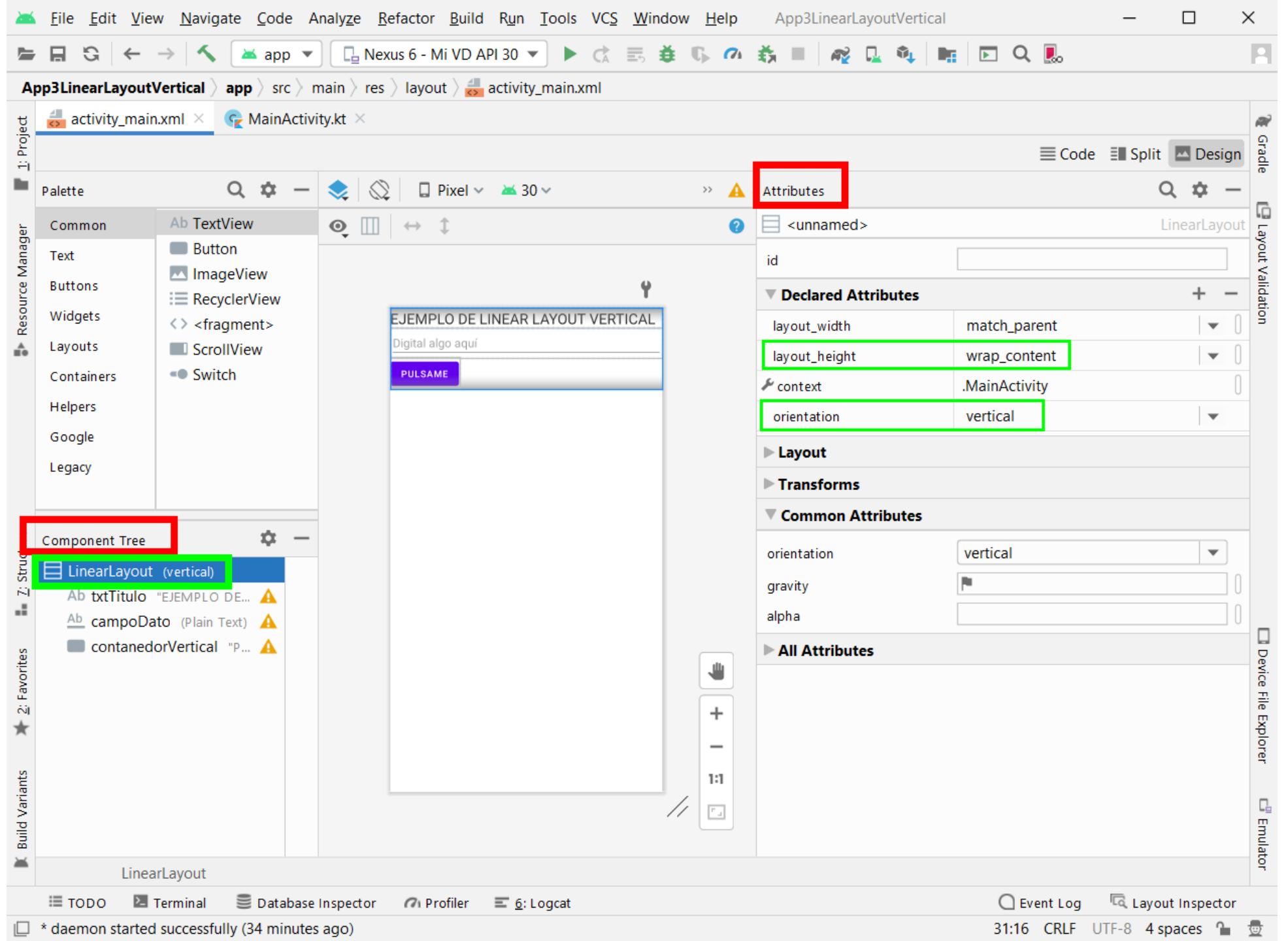
# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical



# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical



# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

The screenshot shows the Android Studio interface with the XML code for an activity. The code defines a vertical linear layout containing a TextView and an EditText, both with specific styling and hints.

```
<?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="wrap_content" android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/txtTitulo"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="EJEMPLO DE LINEAR LAYOUT VERTICAL"
        android:textAppearance="@style/TextAppearance.AppCompat.Large" />

    <EditText
        android:id="@+id/campoData"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:ems="10"
        android:hint="Digital algo aquí"
        android:inputType="textPersonName" />

    <Button
        android:id="@+id/contenedorVertical"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Pulsame" />
</LinearLayout>
```

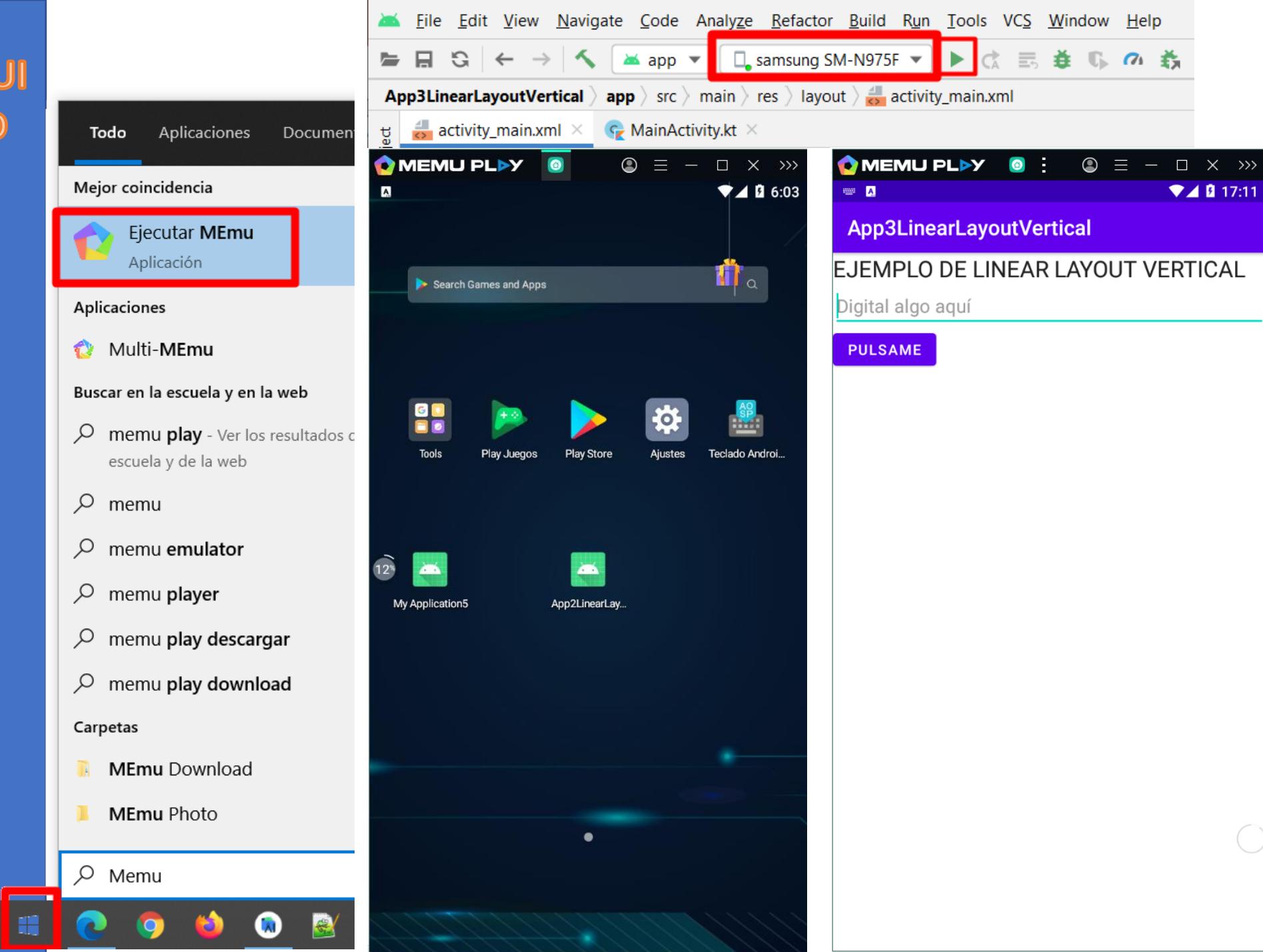
The code is organized into three main sections, each highlighted with a red box:

- Section 1 (Root Layout):** Contains attributes for the root `LinearLayout`, including `xmlns:android`, `xmlns:app`, `xmlns:tools`, `android:layout_width="match_parent"`, `android:layout_height="wrap_content"`, `android:orientation="vertical"`, and `tools:context=".MainActivity"`.
- Section 2 (Text View):** Contains the XML definition for a `TextView` with attributes `android:id="@+id/txtTitulo"`, `android:layout_width="match_parent"`, `android:layout_height="wrap_content"`, `android:text="EJEMPLO DE LINEAR LAYOUT VERTICAL"`, and `android:textAppearance="@style/TextAppearance.AppCompat.Large"`.
- Section 3 (Text Input):** Contains the XML definition for an `EditText` with attributes `android:id="@+id/campoData"`, `android:layout_width="match_parent"`, `android:layout_height="wrap_content"`, `android:layout_weight="1"`, `android:ems="10"`, `android:hint="Digital algo aquí"`, and `android:inputType="textPersonName"`.

The entire code block is enclosed in a large red box.

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical



# DISEÑO DE GUI EN ANDROID

## PROBLEMAS

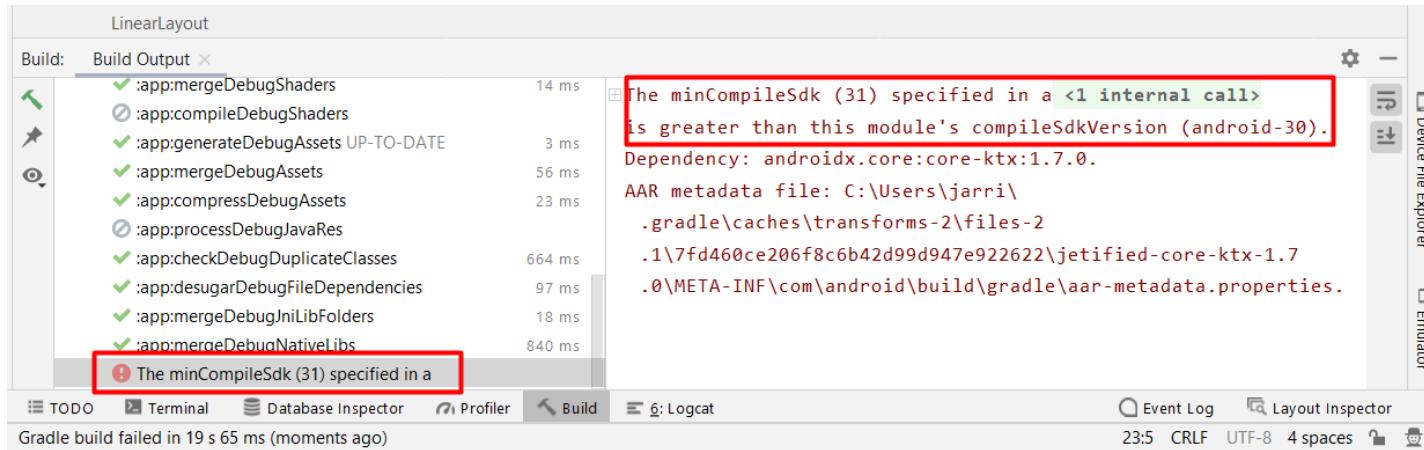
No puede ejecutar la App en el emulador

- ERROR DE minCompileSdk (31)

### build.gradle

```
configurations.all {
    resolutionStrategy { force 'androidx.core:core-ktx:1.6.0' }
}

7   android {
8       compileSdkVersion 30
9       buildToolsVersion "30.0.3"
10
11      defaultConfig {
12          configurations.all {
13              resolutionStrategy { force 'androidx.core:core-ktx:1.6.0' }
14          }
15          applicationId "co.gov.misiontic.ciclo4"
16          minSdkVersion 17
17          targetSdkVersion 30
18          versionCode 1
19          versionName "1.0"
20
21          testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
22      }
23  }
```



```
defaultConfig {
    configurations.all {
        resolutionStrategy { force 'androidx.core:core-ktx:1.6.0' }
    }
    applicationId "com.paxees.tcc"
    minSdkVersion 21
    targetSdkVersion 30
    versionCode 1
    versionName "1.0"
    multiDexEnabled true
    testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
}
```

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

The screenshot shows the Android Studio interface with the following components:

- Code Tab:** Displays the XML code for a linear layout with three buttons.
- Palette:** Shows icons for various UI components like buttons, text views, and images.
- Component Tree:** A tree view of the UI elements in the layout.
- Toolbars:** Includes "Pixel", "30", and other design-related tools.
- Right Panel:** Shows three buttons labeled "BOTON 1", "BOTON 2", and "BOTON 3" arranged vertically.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="horizontal"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <Button
        android:id="@+id/boton1"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:layout_weight="40"
        android:text="Boton 1" />
    <Button
        android:id="@+id/boton2"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_weight="20"
        android:gravity="bottom|right"
        android:text="Boton 2" />
    <Button
        android:id="@+id/boton3"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="top"
        android:layout_weight="40"
        android:text="Boton 3" />
</LinearLayout>
```

The XML code defines a horizontal linear layout with three buttons. The first button has its layout gravity set to "bottom". The second button has its layout gravity set to "center". The third button has its layout gravity set to "top". All buttons have their layout width set to "fill\_parent" and layout height set to "wrap\_content". The layout also has its orientation set to "horizontal" and its layout width and height set to "fill\_parent".

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

The screenshot shows the Android Studio XML layout editor. The code editor displays an XML file with three TextView components nested within a RelativeLayout. The XML code is as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:text="Centrado en el Padre"
        android:textColor="#4BAE4F" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:text="Centrado Verticalmente"
        android:textColor="#FA291A" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:text="Centrado Horizontalmente"
        android:textColor="#0A73C6" />

</RelativeLayout>
```

The attributes `android:layout_centerInParent="true"`, `android:layout_centerVertical="true"`, and `android:layout_centerHorizontal="true"` are highlighted with red boxes. The text content of each TextView is also highlighted with colored boxes: "Centrado en el Padre" (blue), "Centrado Verticalmente" (red), and "Centrado Horizontalmente" (green). The Android Studio interface includes a palette, attributes panel, component tree, and design view.

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

The screenshot shows the Android Studio interface with the following components:

- Code Tab:** Displays the XML layout file content.
- Split Tab:** Shows the corresponding UI preview.
- Palette:** A sidebar with various UI component icons.
- Component Tree:** A tree view of the UI components.
- Bottom Buttons:** Navigation buttons for the preview area.

**XML Layout Content:**

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <Button
        android:id="@+id/primerBtn"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Primer Boton" />
    <Button
        android:id="@+id/segundoBtn"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Segundo Boton" />
</LinearLayout>
```

**UI Preview:** The preview shows a purple screen with two white buttons. The top button is labeled "PRIMER BOTON" and the bottom button is labeled "SEGUNDO BOTON".

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

The screenshot shows the Android Studio interface with the code editor and design preview. The code editor displays the following XML layout:

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

    <Button
        android:id="@+id/primerBtn"
        android:layout_width="wrap_content"
        android:layout_height="fill_parent"
        android:text="Primer Boton" />

    <Button
        android:id="@+id/segundoBtn"
        android:layout_width="wrap_content"
        android:layout_height="fill_parent"
        android:text="Segundo Boton" />
</LinearLayout>
```

The XML code uses a `LinearLayout` with `android:orientation="vertical"`. It contains two `Button` elements. The first button has `android:layout_height="fill_parent"` and the second has `android:layout_height="fill_parent"`, both highlighted with a red box. The design preview on the right shows a purple rectangular area with the text "PRIMER BOTON" centered inside it.

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

The screenshot shows the Android Studio interface with the following details:

- Activity Layout:** activity\_main.xml
- Kotlin File:** MainActivity.kt
- Code Editor:** Displays the XML code for a vertical LinearLayout with two buttons. The code is as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">
    <Button
        android:id="@+id/primerBtn"
        android:text="Primer Boton"
        android:layout_width="wrap_content"
        android:layout_height="fill_parent"/>
    <Button
        android:id="@+id/segundoBtn"
        android:text="Segundo Boton"
        android:layout_width="wrap_content"
        android:layout_height="fill_parent" />
</LinearLayout>
```

- UI Preview:** Shows a vertical stack of two purple buttons labeled "PRIMER BOTON" and "SEGUNDO BOTON".
- Toolbars and Palettes:** Standard Android Studio toolbars and palettes are visible.

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

The screenshot shows the Android Studio interface with the following components:

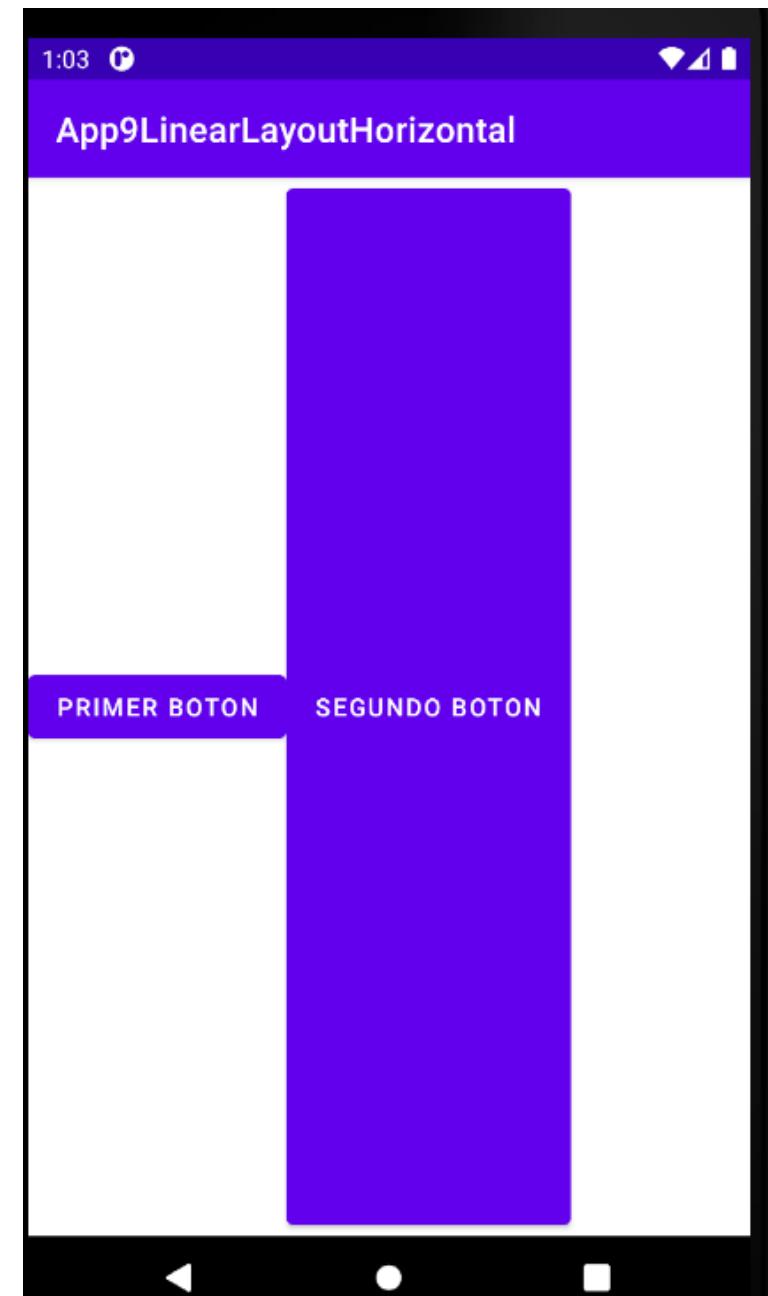
- Code Editor:** Displays the XML code for a layout. The code defines a horizontal `LinearLayout` containing two `Button` elements. The `LinearLayout` has attributes: `android:orientation="horizontal"`, `android:layout_width="fill_parent"`, and `android:layout_height="fill_parent"`. Each `Button` has attributes: `android:id="@+id/primerBtn"`, `android:layout_width="wrap_content"`, `android:layout_height="wrap_content"`, and `android:text="Primer Boton"`. A second `Button` with similar attributes and text "Segundo Boton" is also present.
- Layout Preview:** Shows a visual representation of the layout. It consists of a blue outer container holding two purple rectangular buttons. The left button is labeled "PRIMER BOTON" and the right button is labeled "SEGUNDO BOTON".
- Component Tree:** Located at the bottom right, it shows a tree structure of the layout components.
- Toolbars and Palettes:** Standard Android Studio toolbars and palettes are visible along the top and right edges.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="horizontal"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <Button
        android:id="@+id/primerBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Primer Boton" />
    <Button
        android:id="@+id/segundoBtn"
        android:layout_width="wrap_content"
        android:layout_height="fill_parent"
        android:text="Segundo Boton" />
</LinearLayout>
```

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4          android:orientation="horizontal"
5          android:layout_width="fill_parent"
6          android:layout_height="fill_parent" >
7      <Button
8          android:id="@+id/primerBtn"
9              android:layout_width="wrap_content"
10             android:layout_height="wrap_content"
11             android:text="Primer Boton" />
12     <Button
13         android:id="@+id/segundoBtn"
14             android:layout_width="wrap_content"
15             android:layout_height="fill_parent"
16             android:text="Segundo Boton" />
17  </LinearLayout>
18
19
20
21
22
23
24
25
26
27
28
29
```



# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

The screenshot shows the Android Studio interface with the XML code for a RelativeLayout. The code defines five TextViews with specific layout parameters:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:text="ARRIBA" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="ABAJO"
        android:layout_alignParentBottom="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="A LA IZQUIERDA"
        android:layout_alignParentLeft="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="A LA DERECHA"
        android:layout_alignParentRight="true" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="EN EL CENTRO"
        android:layout_centerInParent="true" />

</RelativeLayout>
```

The XML code is annotated with red boxes highlighting specific attributes:

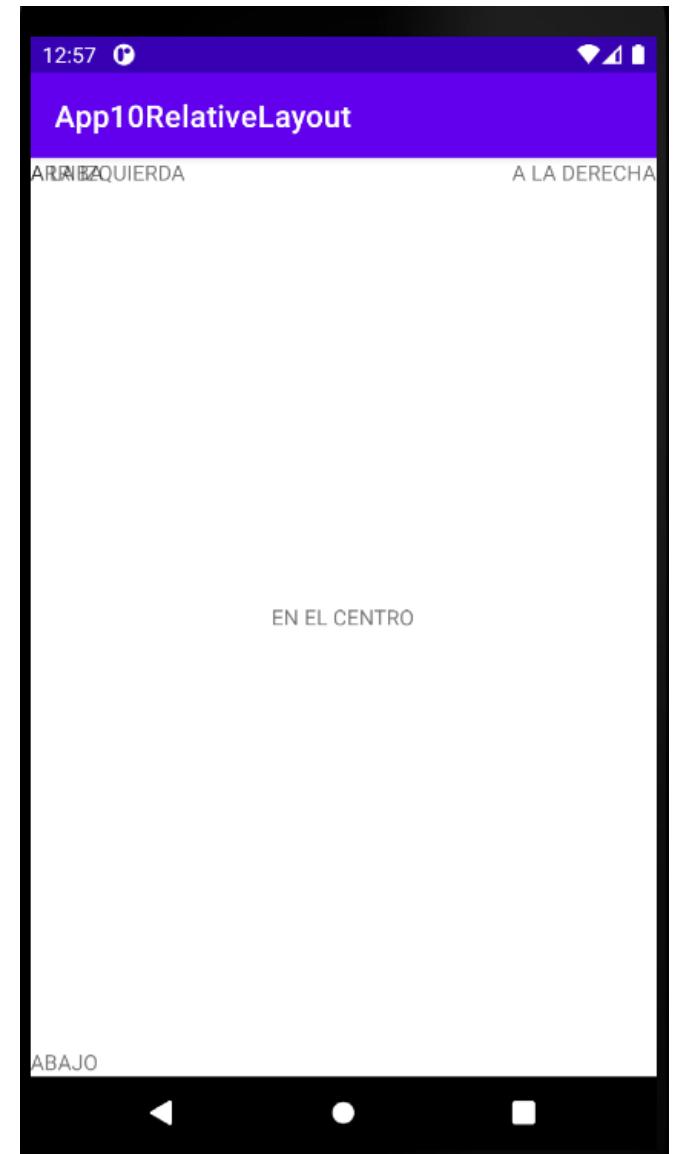
- Line 4:** android:layout\_width="fill\_parent" and android:layout\_height="fill\_parent"
- Line 6:** android:layout\_width="wrap\_content", android:layout\_height="wrap\_content", and android:text="ARRIBA"
- Line 13:** android:layout\_alignParentBottom="true"
- Line 18:** android:layout\_alignParentLeft="true"
- Line 23:** android:layout\_alignParentRight="true"
- Line 28:** android:text="EN EL CENTRO" and android:layout\_centerInParent="true"

The right side of the screen shows the Android Layout Editor with a blue frame representing the layout. Inside the frame, text labels indicate the position of each TextView: "ARRIBA" (top), "ABAJO" (bottom), "A LA IZQUIERDA" (left), "A LA DERECHA" (right), and "EN EL CENTRO" (center). A vertical toolbar on the right contains icons for zooming and orientation.

# DISEÑO DE GUI EN ANDROID

## EJEMPLO 2 USO DE LinearLayout Vertical

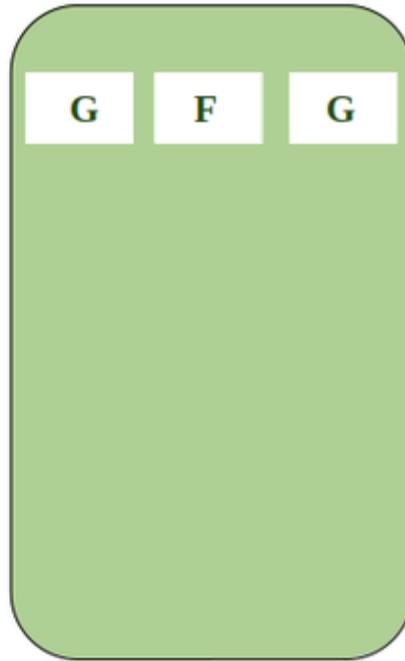
```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     android:layout_width="fill_parent" android:layout_height="fill_parent">
5     <TextView
6         android:layout_width="wrap_content"
7         android:layout_height="wrap_content" android:text="ARRIBA"
8         android:layout_alignParentTop="true" />
9     <TextView
10        android:layout_width="wrap_content"
11        android:layout_height="wrap_content"
12        android:text="ABAJO"
13        android:layout_alignParentBottom="true" />
14     <TextView
15        android:layout_width="wrap_content"
16        android:layout_height="wrap_content"
17        android:text="A LA IZQUIERDA"
18        android:layout_alignParentLeft="true" />
19     <TextView
20        android:layout_width="wrap_content"
21        android:layout_height="wrap_content"
22        android:text="A LA DERECHA"
23        android:layout_alignParentRight="true" />
24     <TextView
25        android:layout_width="wrap_content"
26        android:layout_height="wrap_content"
27        android:text="EN EL CENTRO"
28        android:layout_centerInParent="true" />
29 </RelativeLayout>
```



## VERTICAL VIEW



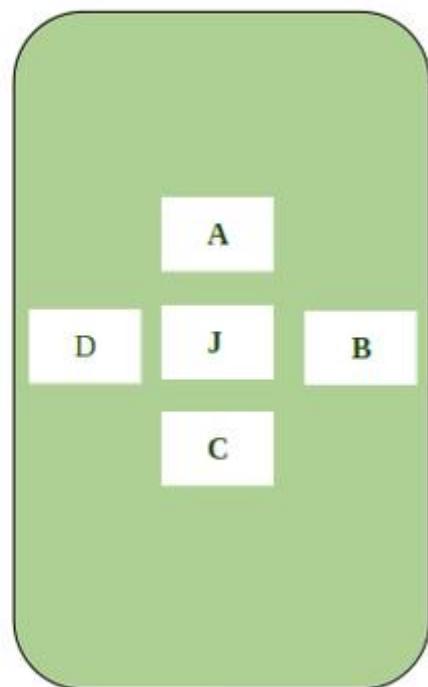
## HORIZONTAL VIEW



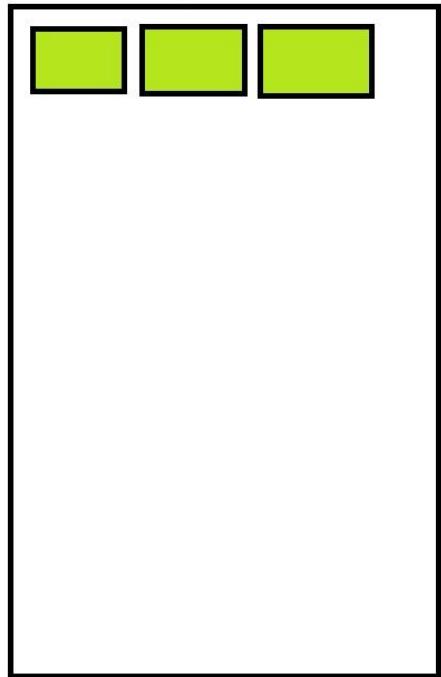
### Relative View 1



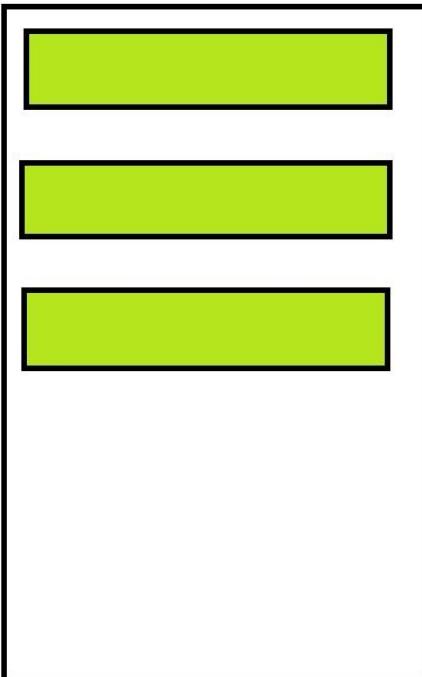
### Relative View 2



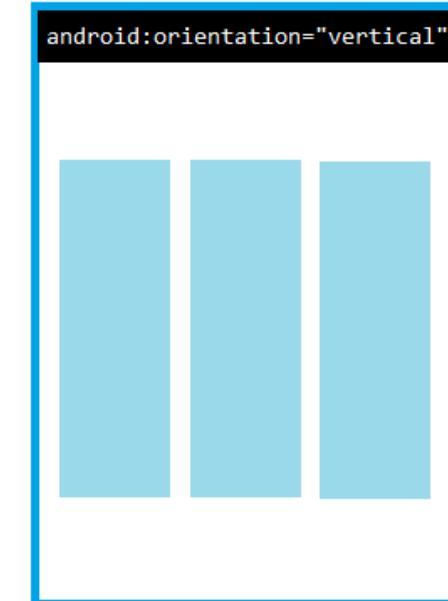
Linear Layout Horizontal



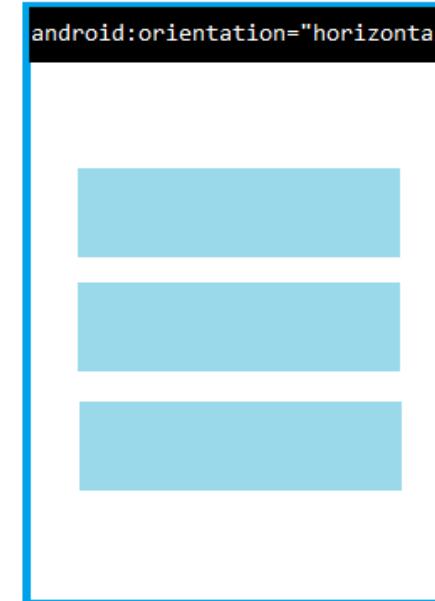
Linear Layout Vertical



android:orientation="vertical"



android:orientation="horizontal"

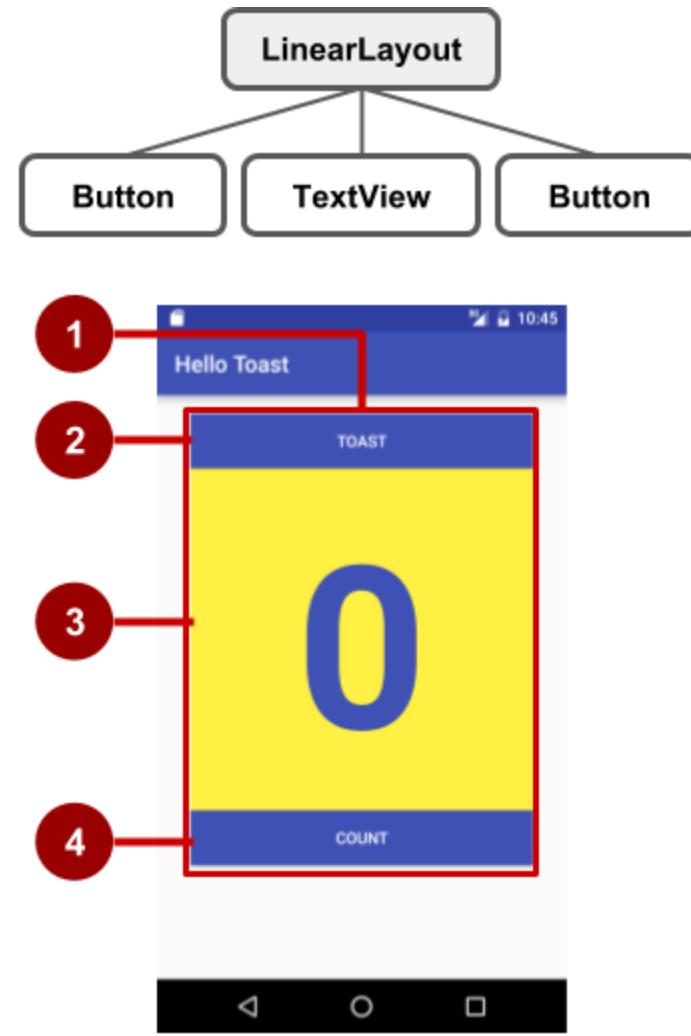
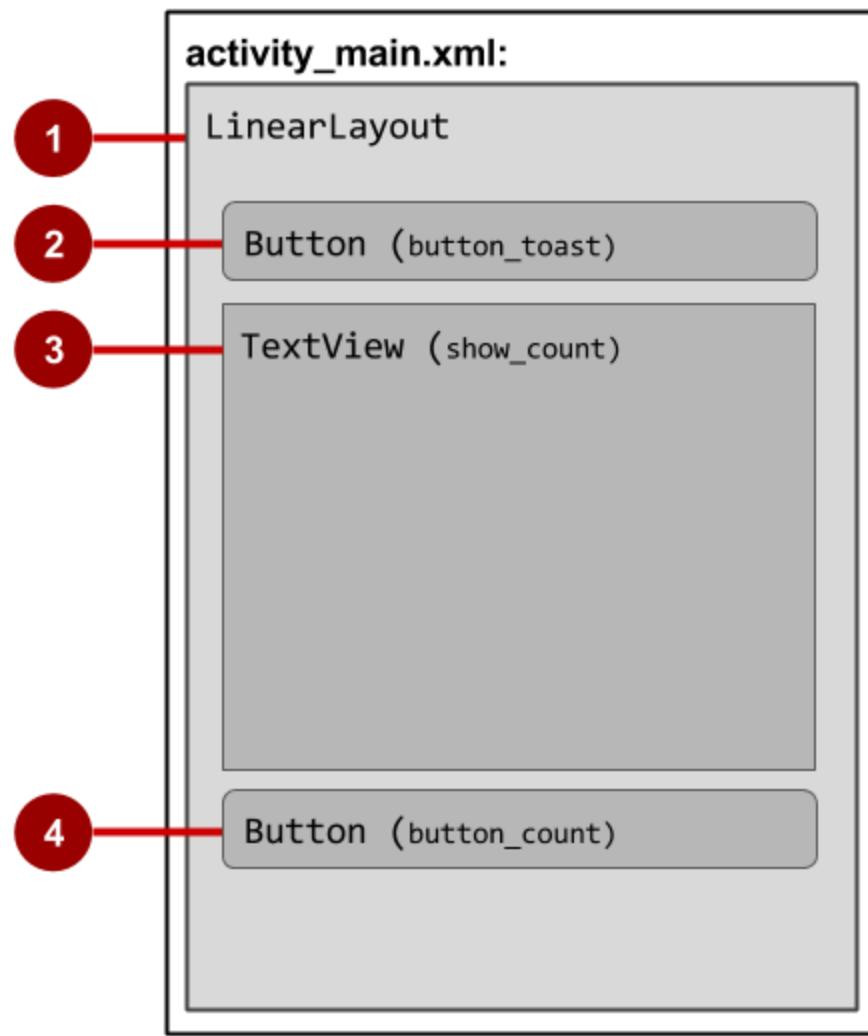


Android:orientation="horizontal"



Android:orientation="vertical"





# Linear Layout

android:orientation="vertical"

