

Módulo: Programación Multimedia y dispositivos móviles.
Unidad de Trabajo Nº1: Introducción al desarrollo de aplicaciones móviles.
Actividad de Clase 3: Lanzar una segunda Activity.

1. Objetivo general.

Comenzar a manejar diferentes componentes en Android Studio y lanzar nuevas actividades (Activity).

2. Metodología.

Cada alumno realizará la app.

3. Descripción.

Realiza lo siguiente:

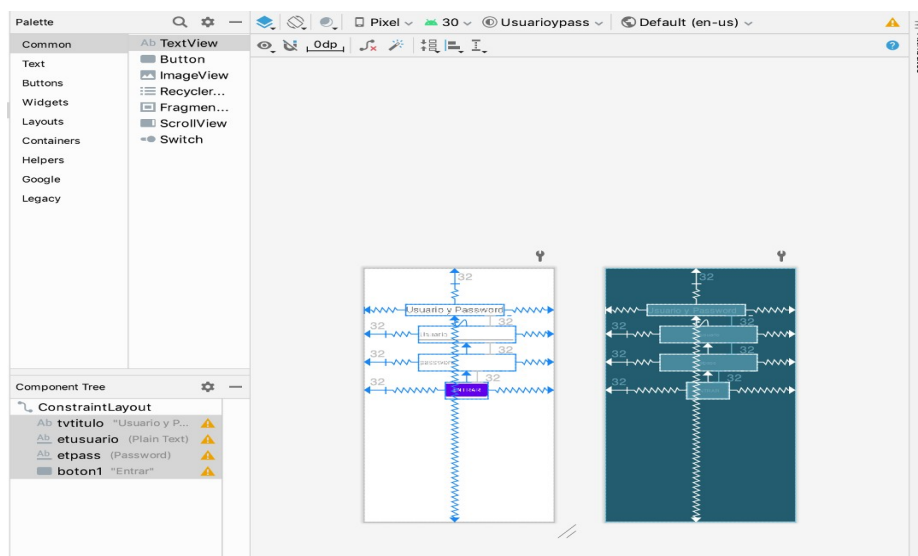
Lanzar Activity

Vamos a crear una app que va a simular un login. El usuario pondrá su nombre y password y hará click en el botón acceder. El programa comprobará si el Usuario es: **Admin** y la password (**1906Red**) se abrirá un segunda actividad donde se recoge el nombre de usuario y se le saluda en un **TextView**. En caso de que el nombre de usuario y la contraseña no coincidan debe aparecer un mensaje en **Toast** en el que diga usuario o clave incorrecta. Si quieres, puedes especificar cuál de los dos (usuario o contraseña) no son correctos.

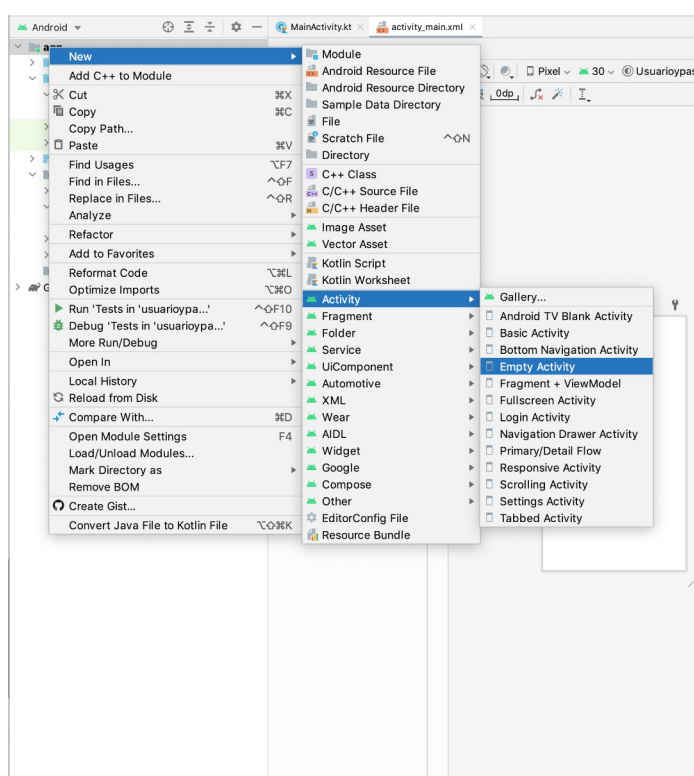
1. La **Activity Principal** contendrá los siguientes controles:

- 1.1. Un **TextView** para el Título de la App.
- 1.2. Dos **EditText** (Uno para el nombre de usuario y otro para la contraseña).
- 1.3. Un **botón** que le de a entrar.

2. La **Activity** podría quedar de forma similar a la que aparece en la siguiente imagen:



3. Recordamos que al hacer click sobre el botón se abrirá una segunda actividad (caso de que coincidan el nombre de usuario y la contraseña) donde se recoge el nombre de usuario y se le saluda en un TextView. Para el paso de mensajes entre Actividades utilizaremos objetos **Intent**.
4. Vamos a crear la segunda Activity. Para ello hacemos clic con el botón derecho del mouse en la ventana de Project donde dice "app" y seleccionamos New -> Activity -> Empty Activity.



5. Aparece un diálogo donde debemos definir el nombre del Activity "Activity Name" y le he asignado como nombre "Saludo" (puedes usar el que creas conveniente (se crearán dos archivos **Saludo.kt** y **activity_saludo.xml**). Si has utilizado otro nombre el nombre será que el hayas elegido.

Empty Activity
Creates a new empty activity

Activity Name

☒ Generate a Layout File

Layout Name

☐ Launcher Activity

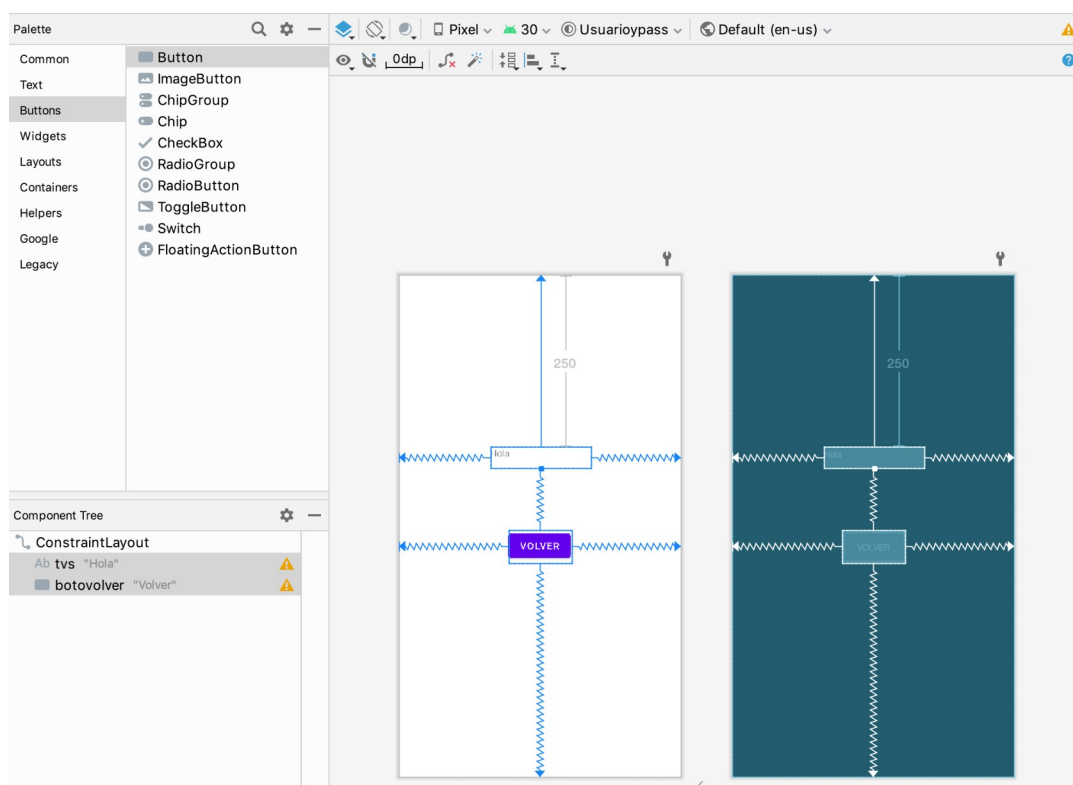
Package name

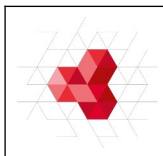
Source Language

Target Source Set

Cancel Previous Next Finish

6. Una vez creada la segunda activity tienes que añadir un TextView donde se mostrará el nombre de usuario y un botón que ponga volver y que al pulsarlo volverá a la primera Activity (MainActivity). Debería quedar algo similar a la siguiente imagen.





7. Ahora vamos a programar la funcionalidad en MainActivity para que cuando se haga clic en el botón se inicie la segunda actividad. Siempre que coincidan el nombre de usuario y la contraseña especificados:

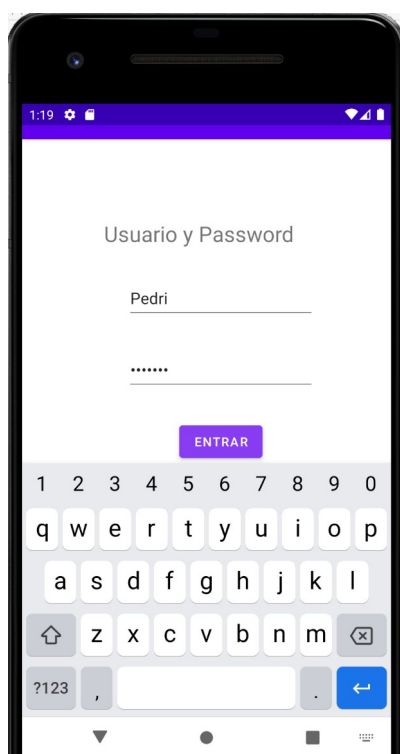
```
} boton.setOnClickListener { it: View!  
    //Creamos el objeto Intent a través de cuál vamos a enviar parámetros a la otra actividad  
    val intento1 = Intent( packageContext: this@MainActivity, Saludo::class.java)  
    //Pasamos el parámetro en este caso el nombre de usuario  
    intento1.putExtra( name: "usuario", etusuario.text.toString())  
    //Iniciamos la actividad  
    startActivity(intento1)  
}
```

8. Lo siguiente que tenemos que hacer es realizar el código de la actividad secundaria donde primero tenemos que recoger el dato que se nos ha pasado de la Activity que ha llamado, mostrarlo en el TextView y luego también implementar el listener para cuando el usuario pulse sobre el botón Volver volvamos a la Activity principal.

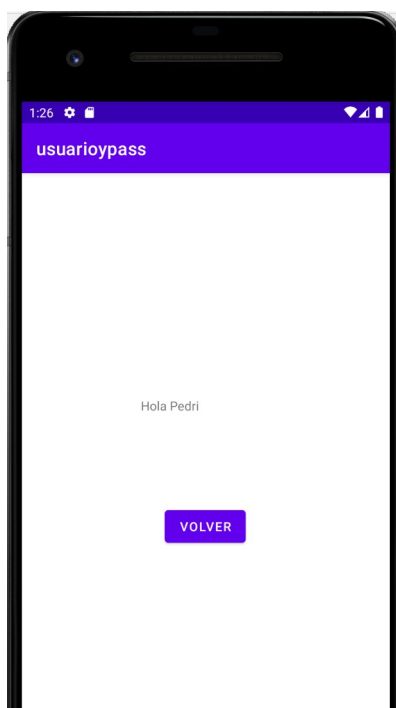
```
1 package com.jacintorm.usuarioypass  
2  
3 import android.content.Intent  
4 import androidx.appcompat.app.AppCompatActivity  
5 import android.os.Bundle  
6 import android.widget.Button  
7 import android.widget.TextView  
8  
9 class Saludo : AppCompatActivity() {  
10     override fun onCreate(savedInstanceState: Bundle?) {  
11         super.onCreate(savedInstanceState)  
12         setContentView(R.layout.activity_saludo)  
13  
14         //Definimos un objeto Bundle para acceder a los datos que nos han enviado.  
15         val bundle = intent.extras  
16         val usuario = bundle?.getString( key: "usuario")  
17         //Definimos los objetos que contiene la actividad Saluda  
18         val tvu = findViewById<TextView>(R.id.tvu)  
19         val botonback = findViewById<Button>(R.id.botonvolver)  
20         tvu.text = "Hola ${usuario}"  
21         //Definimos el listener para el boton volver  
22         botonback.setOnClickListener { it: View!  
23             val intento2 = Intent( packageContext: this@Saludo, MainActivity::class.java)  
24             startActivity(intento2)  
25         }  
26     }  
27 }  
28 }
```

En esta clase definimos una variable de tipo Bundle y la inicializamos con el valor de la propiedad extras de la propiedad intent del Activity (esto lo hacemos para recuperar el o los parámetros que envió la otra actividad (Activity)). También cuando el usuario haga clic en el botón volver crearemos otro Intent para volver a la actividad principal.

9. Ahora ejecuta la app para que veas el funcionamiento de esta. Introduciendo datos y comprobando los resultados. Vamos a ver la primera Activity (MainActivity).



10. Ahora vemos que al hacer clic sobre el botón ENTRAR aparece la segunda actividad con el nombre de usuario que se ha introducido (si coincide usuario y contraseña especificados).



11. Una vez probada la app. ¿Qué cambios tendría que realizar en ella si además del nombre de usuario quisiera pasarle la contraseña a la segunda activity?. Hazlo y comprueba su funcionamiento.

12. Utiliza el Debugger. Para ello coloca un breakpoint justo antes pasar los datos a la segunda actividad y comprueba que el valor que se le va a pasar es el correcto. En la siguiente imagen puedes ver un ejemplo...

