

2º DAM

*Programación Multimedia y
Dispositivos Móviles*

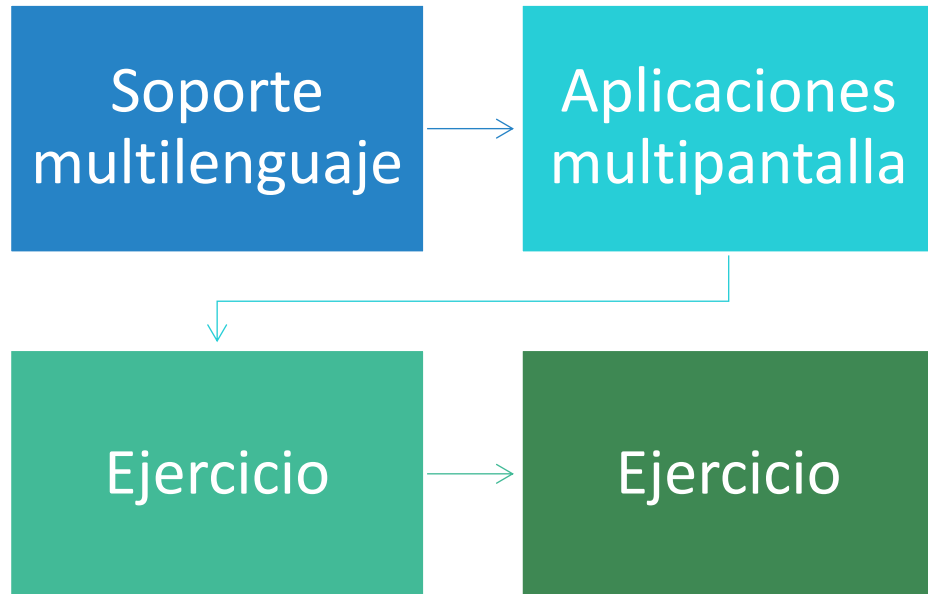
Programación Android



android

José A. Lara

Contenido



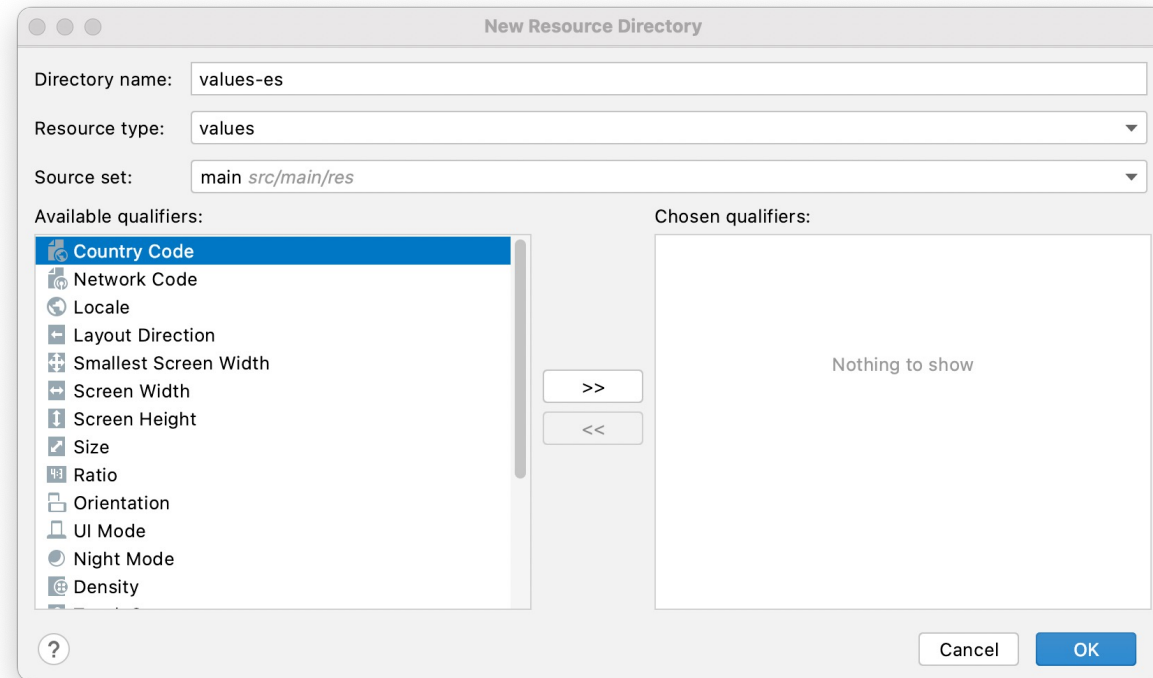
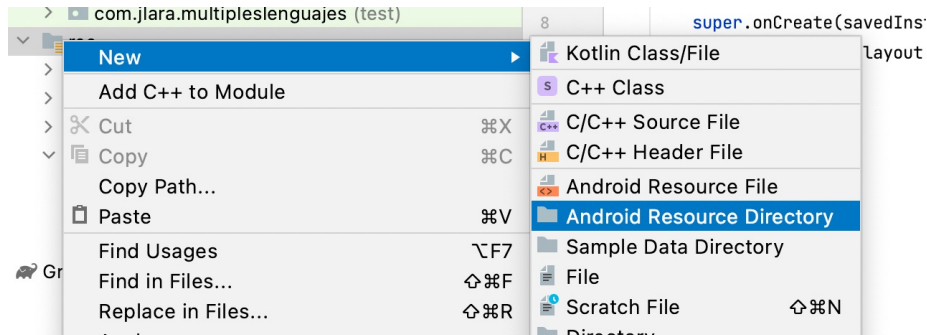
Soporte multilinguaje



Programación Android

Soporte multilinguaje

Crearemos un archivo strings.xml para que soporte múltiples idiomas. Creamos en primer lugar un nuevo Android Resource Directory llamado values-es (es para español)



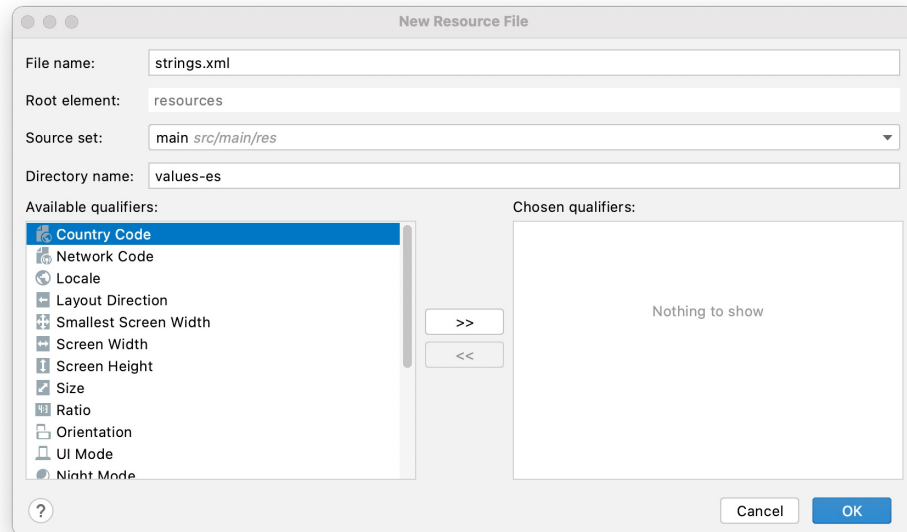
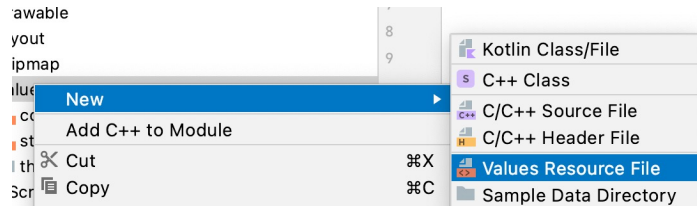
Programación Android

Soporte multilinguaje

Originalmente tenemos en strings.xml lo siguiente:

Creamos un nuevo strings.xml en values-es

```
<resources>
    <string name="app_name">MultiplesLenguajes</string>
    <string name="codigo">Code</string>
    <string name="clave">Password</string>
    <string name="usuario">User</string>
</resources>
```



```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">MultiplesLenguajes</string>
    <string name="codigo">Código</string>
    <string name="clave">Contraseña</string>
    <string name="usuario">Usuario</string>
</resources>
```



Programación Android

Soporte multilinguaje

Cuando se ejecuta la app, si el dispositivo está configurado en inglés, se mostrarán los textos que aparecen en strings.xml que está en values. Si está configurado en español, se mostrarán los textos que aparecen en strings.xml que está en values-es.



Code
User
Password



Código
Usuario
Contraseña



*Aplicaciones
multipantalla*



android

Programación Android

Aplicaciones multipantalla

Cualquier aplicación con cierta entidad dispondrá de varias pantallas para interactuar con el usuario. Una aplicación con tales características en Android se puede desarrollar y gestionar de diversas maneras, pero la forma más correcta es mediante el uso de distintas Activity, que además de permitir un desarrollo más sencillo y más modular, permite la integración con otras Activity y son controladas de forma automática por el ciclo de vida de aplicaciones Android.

Para lanzar una Activity simplemente es necesario obtener una instancia de Intent que describe la Activity a ser lanzada y se utiliza como argumento del método startActivity que permite lanzar tanto una Activity que se encuentre dentro de la misma aplicación como una Activity externa a la misma.



Programación Android

Intent

Una **Intent** es un objeto de mensajería que puedes usar para solicitar una acción de otro componente de una app. Si bien las intents facilitan la comunicación entre componentes de varias formas, existen tres casos de uso principales:

- Iniciar una actividad: Una Activity representa una única pantalla en una aplicación.
- Iniciar un servicio: Un servicio es un componente que realiza operaciones en segundo plano sin una interfaz de usuario.
- Transmitir una emisión: Una emisión es un aviso que cualquier aplicación puede recibir. El sistema transmite varias emisiones de eventos, como cuando se inicia el sistema o comienza a cargarse el dispositivo.



Programación Android

Iniciar una actividad

Una Activity representa una única pantalla en una aplicación.

Para lanzar una Activity de la propia aplicación, es necesario tenerla configurada dentro de AndroidManifest.xml como una Activity adicional:

```
<activity android:name=".MainActivity">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```



Programación Android

Iniciar una actividad

Desde código, simplemente será necesario invocarla empleando `startActivity` y pasando como parámetro un `Intent` que especifica cuál es la Activity actual (`this`) y cuál la Activity que queremos abrir (`ActivityDos`).

```
val intent = Intent( packageContext: this, ActivityDos::class.java)
this.startActivity(intent)
```

```
Intent intent=new Intent( packageContext: this,ActivityDos.class);
this.startActivity(intent);
```



Programación Android

Iniciar una actividad

Para pasar parámetros a la actividad se utilizan los extras del Intent. Se puede pasar cualquier tipo básico, así como un array de éstos o incluso elementos Serializables:

```
val intent = Intent( packageContext: this, ActivityDos::class.java)
intent.putExtra( name: "nombre", nombreEmpleado)
intent.putExtra( name: "sueldo", sueldo)
this.startActivity(intent)
```

```
Intent intent=new Intent( packageContext: this,ActivityDos.class);
intent.putExtra( name: "nombre",nombreEmpleado);
intent.putExtra( name: "sueldo",sueldo);
this.startActivity(intent);
```



Programación Android

Iniciar una actividad

En ActivityDos se podrán “recoger” los valores de estas variables mediante (normalmente en el método onCreate):

```
val nombre = intent.getStringExtra(name: "nombre")  
val sueldo = intent.getDoubleExtra(name: "sueldo", defaultValue: 0.0)
```

```
String nombre = getIntent().getStringExtra(name: "nombre");  
double sueldo = getIntent().getDoubleExtra(name: "sueldo", defaultValue: 0.0);
```



Programación Android

Iniciar una actividad para recibir datos

Con lo comentado hasta el momento solamente se puede invocar a la Activity para ser mostrada, pero no para realizar interacción entre ambas, es decir, para obtener un resultado en la primera Activity en función de lo realizado en la segunda.

Para poder acceder al resultado producido por una Activity es necesario emplear una variable de tipo `ActivityResultLauncher` que va a lanzar un `Intent` y va a recibir mediante `registerForActivityResult` los datos desde esa Activity.



Programación Android

Iniciar una actividad

En la primera Activity tendremos:

```
class MainActivity : AppCompatActivity(), View.OnClickListener {  
    private var intentLaunch: ActivityResultLauncher<Intent>? = null  
    private var nombre:String?="Sin nombre"  
    private var sueldo:Double?=0.0  
  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_main)  
        val btnAbrir = findViewById<Button>(R.id.btnAbrir)  
        btnAbrir.setOnClickListener(this)  
        val tvTexto = findViewById<TextView>(R.id.tvTexto)  
        tvTexto.text="Datos no obtenidos"  
        intentLaunch = registerForActivityResult(  
            StartActivityForResult()  
        ) { result: ActivityResult ->  
            if (result.resultCode == RESULT_OK) {  
                nombre = result.data?.extras?.getString( key: "nombre")  
                sueldo = result.data?.extras?.getDouble( key: "sueldo")  
                tvTexto.text = nombre+" - Sueldo:"+sueldo.toString()  
            }  
        }  
    }  
  
    override fun onClick(p0: View?) {  
        val intent = Intent( packageContext: this, ActivityDos::class.java)  
        intent.putExtra( name: "nombre", nombre)  
        intent.putExtra( name: "sueldo", sueldo)  
        intentLaunch?.launch(intent)  
    }  
}
```



Programación Android

Iniciar una actividad

En la primera Activity tendremos:

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener{
    private ActivityResultLauncher<Intent> intentLaunch;
    private String nombreEmpleado="";
    private double sueldo=0;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button btnAbrir=findViewById(R.id.btnAbrir);
        btnAbrir.setOnClickListener(this);
        TextView tvTexto=findViewById(R.id.tvTexto);
        intentLaunch = registerForActivityResult(
            new ActivityResultContracts.StartActivityForResult(),
            result ->{
                if (result.getResultCode()== Activity.RESULT_OK){
                    nombreEmpleado=result.getData().getStringExtra( name: "nombre");
                    sueldo=result.getData().getDoubleExtra( name: "sueldo", defaultValue: 0);
                    tvTexto.setText(nombreEmpleado+" - Sueldo:"+sueldo);
                }
            }
        );
    }

    @Override
    public void onClick(View view) {
        Intent intent=new Intent( packageContext: this,ActivityDos.class);
        intent.putExtra( name: "nombre",nombreEmpleado);
        intent.putExtra( name: "sueldo",sueldo);
        intentLaunch.launch(intent);
    }
}
```



Programación Android

Iniciar una actividad

Mientras que, en la segunda Activity tendremos:

```
class ActivityDos : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_dos)
        val nombre = intent.getStringExtra( name: "nombre")
        val sueldo = intent.getDoubleExtra( name: "sueldo", defaultValue: 0.0)
        val etNombre = findViewById<TextView>(R.id.etNombre)
        val etSueldo = findViewById<TextView>(R.id.etSueldo)
        etNombre.text = nombre
        etSueldo.text = sueldo.toString()
        val btnDevolver = findViewById<Button>(R.id.btnDevolver)
        btnDevolver.setOnClickListener { it: View!
            val intent = Intent()
            val name=etNombre.text.toString()
            val salary=etSueldo.text.toString().toDouble()
            intent.putExtra( name: "nombre", name)
            intent.putExtra( name: "sueldo", salary)
            setResult(RESULT_OK, intent)
            finish()
        }
    }
}
```



Programación Android

Iniciar una actividad

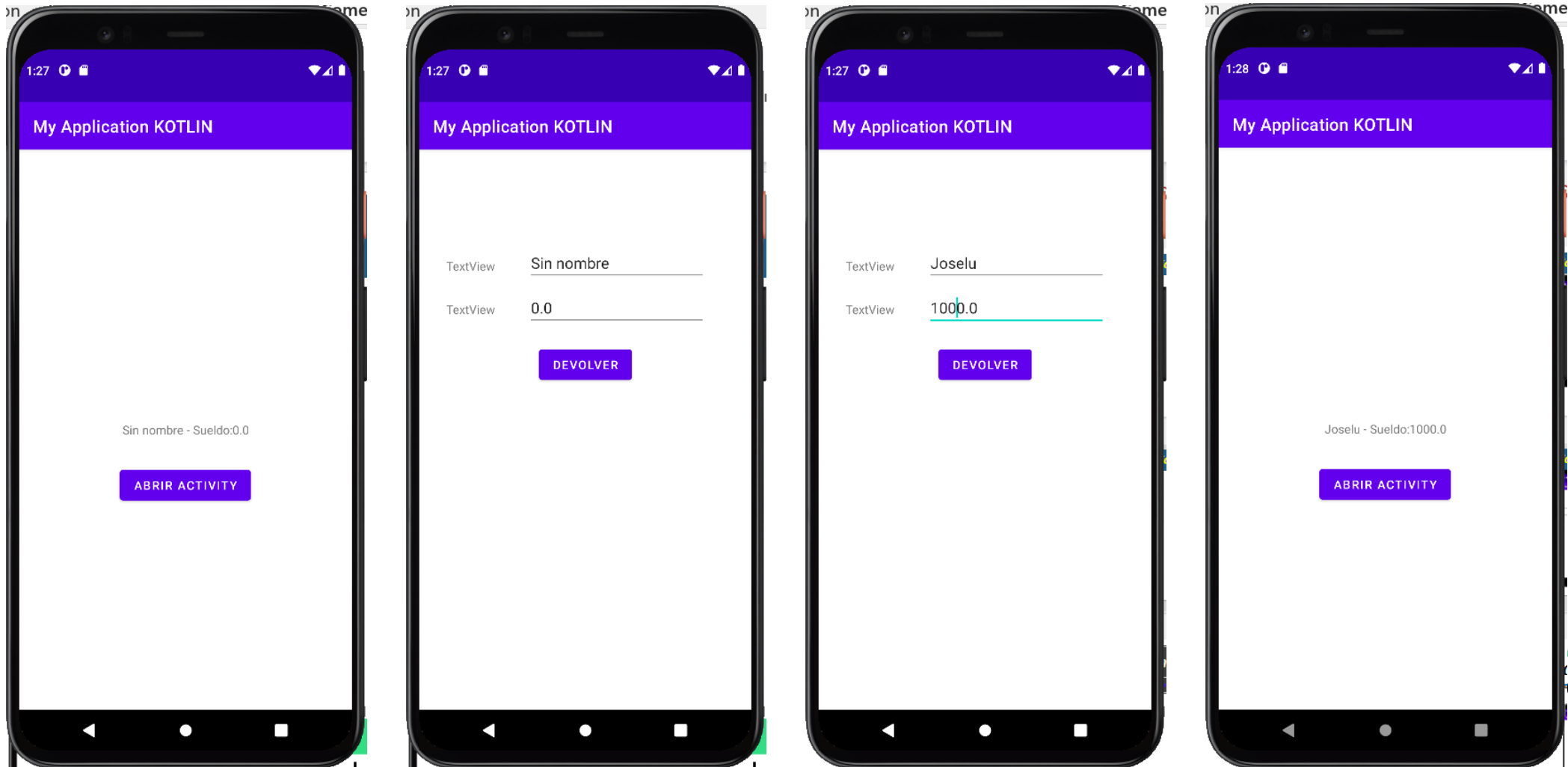
Mientras que, en la segunda Activity tendremos:

```
public class ActivityDos extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_dos);
        String nombre = getIntent().getStringExtra(name: "nombre");
        double sueldo = getIntent().getDoubleExtra(name: "sueldo", defaultValue: 0.0);
        EditText etNombre=findViewById(R.id.etNombre);
        EditText etSueldo=findViewById(R.id.etSueldo);
        etNombre.setText(nombre);
        etSueldo.setText(sueldo+"");
        Button btnDevolver=findViewById(R.id.btnDevolver);
        btnDevolver.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent=new Intent();
                intent.putExtra(name: "nombre",etNombre.getText().toString());
                intent.putExtra(name: "sueldo",Double.parseDouble(etSueldo.getText().toString()));
                setResult(Activity.RESULT_OK,intent);
                finish();
            }
        });
    }
}
```



Programación Android

Iniciar una actividad



Ejercicio



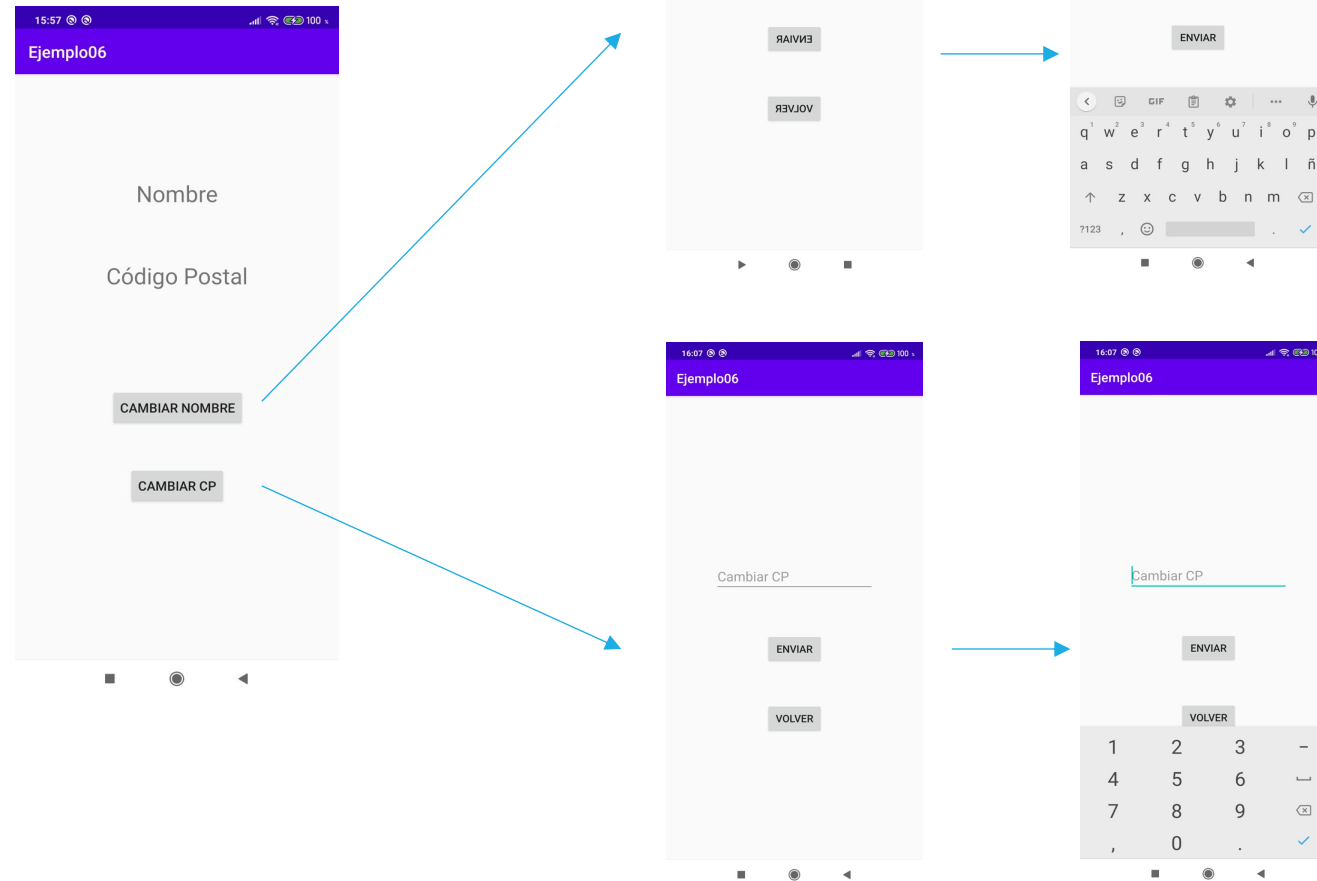
android



Programación Android

Ejercicio

Implementar una aplicación que siga el esquema de pantallas que se presenta a continuación:



Ejercicio



android



Programación Android

Ejercicio

Implementar una aplicación Android que tenga las siguientes pantallas:

- Una lista de comunidades y ciudades autónomas.
- Al dejar un ítem de la lista pulsado (long click) aparecerá el menú contextual con las opciones Eliminar y Editar.
- La opción Eliminar elimina un elemento de la lista.
- Al pulsar sobre Editar se abre una nueva pantalla (activity) con el texto de la comunidad en un campo editable, y con la imagen debajo, de forma que el usuario pueda modificar únicamente el nombre.



Programación Android

Ejercicio

