

Intents I. Lanzar unha Activity

Sumario

- 1 Introducción
 - ◆ 1.1 Caso práctico
 - ◆ 1.2 Creación dunha segunda Activity
 - ◆ 1.3 O XML do layout da Activity principal
 - ◆ 1.4 A clase Java a activity principal
 - ◆ 1.5 O XML da Activity que recibe a chamada
 - ◆ 1.6 A clase Java da Activity que recibe a chamada

Introdución

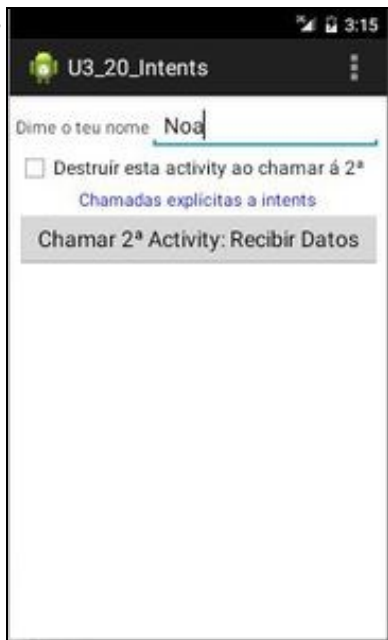
- Un **Intent** é o que lle permite a unha aplicación manifestar a "intención" de que desexa facer algo. Por exemplo:
 - ◆ Abrir unha nova actividade (pantalla), da mesma ou distinta aplicación.
 - ◆ Pasar uns datos dunha activity a outra, da mesma ou distinta aplicación.
 - ◆ Interconectar compoñentes da mesma ou distinta aplicación a través de mensaxes.
- Para poder usar un Intent os sistema debe saber que

Las Intenciones (Intents) permiten a las aplicaciones de Android expresar la intención de que se desea ejecutar una acción sobre unos datos usando algún componente de ésta o de otra aplicación. Las intenciones permiten interconectar componentes de la misma o de distintas aplicaciones mediante mensajes.

- Cando se crea unha aplicación créase a Activity *principal* (A pantalla *principal*).
- Dende esta Activity pódese chamar a outras Activities (pantallas) da propia aplicación ou doutras.
- Para lanzar unha unha activity dende outra úsase a clase **Intent**.
- Un Intent permite chamar a outros Activities, pasarlles e recoller información, interactuar con servizos do sistema, etc.
- Na unidade 3 afondarase sobre o ciclo de vida dunha Activity.
- **Referencias:**
 - ◆ <http://developer.android.com/training/basics/firstapp/starting-activity.html>
 - ◆ <http://developer.android.com/reference/android/content/Intent.html>
 - ◆ <http://developer.android.com/guide/components/intents-filters.html>

Caso práctico

- Crear o proxecto: **U3_20_Intents**.
- Chamadas entre Activities



Dende a pantalla principal vaise introducir un nome. Cando se prema o botón chamarase á segunda activity que recibira o valor do cadro de texto. Ademais nesta primeira pantalla temos a opción de destruír esa activity no momento de chamar á segunda.

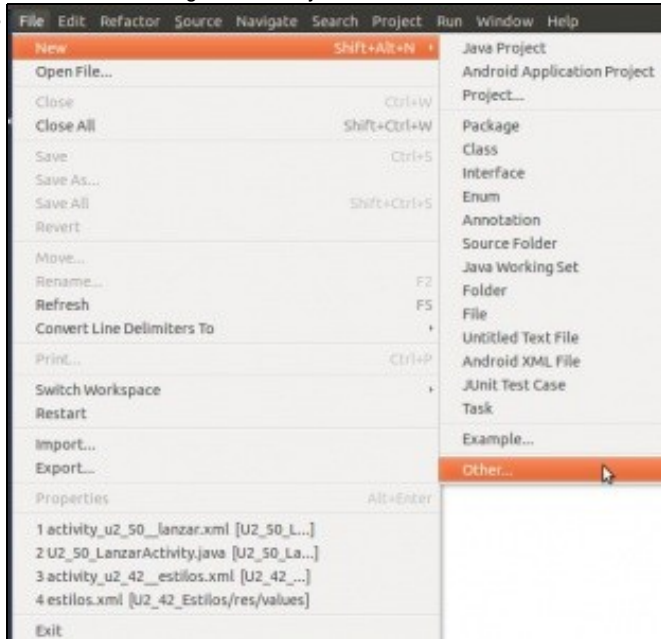
Aconséllase experimentar con esa opción e usar o botón **Back** unha vez que se está na segunda Activity.



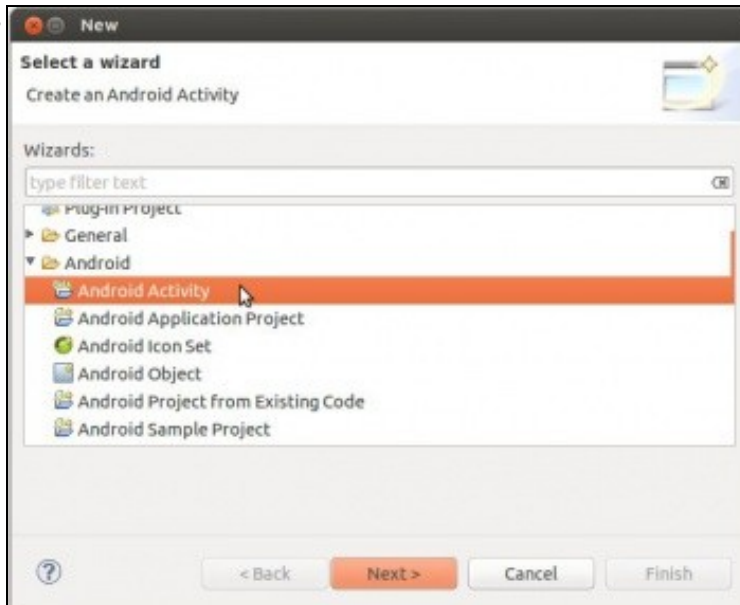
Segunda pantalla que recibe a información da primeira e ademais ten un botón para destruír esta segunda Activity.

Creación dunha segunda Activity

- Pódese facer manualmente ou a través da utilidade que proporciona o IDE para crear unha Activity.
- Creación dunha segunda Activity



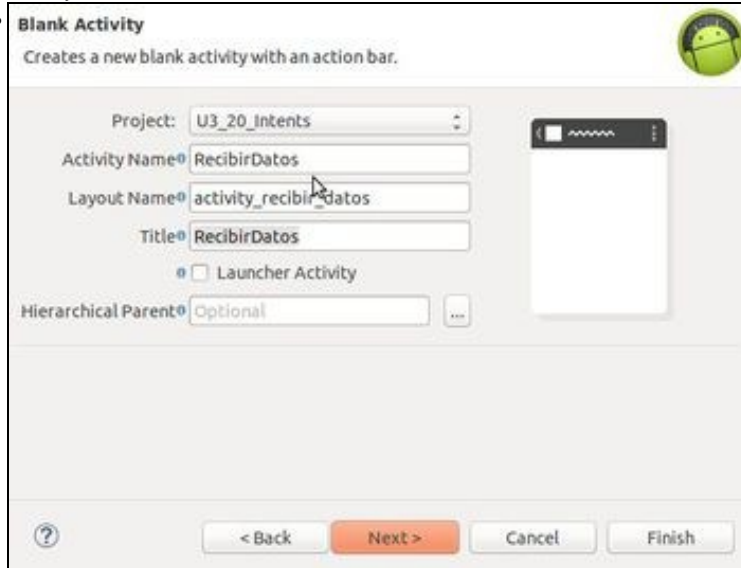
Ir ao menú **File->New->Other**



Escoller Android Activity



Activity en branco.



Escoller en que proxecto crea e o nome, neste caso: RecibirDatos



Observar que a ferramenta xa nos creou

- un XML e
- unha Clase Java para esa Activity.

Agora só resta codificar e deseñar o layout.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.u3_20_intents"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="16"
        android:targetSdkVersion="21" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".U3_20_Intents"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".RecibirDatos"
            android:label="@string/title_activity_recibir_datos" >
        </activity>
    </application>
</manifest>
```

A ferramenta incluso modifica o ficheiro **AndroidManifest.xml**, creando a entrada para a nova Activity.

O XML do layout da Activity principal

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Dime o teu nome" />

    <EditText
        android:id="@+id/et_nome"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

    <CheckBox
        android:id="@+id/chk_destruir"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:checked="false"
        android:text="Destruir esta activity ao chamar á nova" />

    <Button
        android:id="@+id/btn_enviar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="onEnviarClick"
        android:text="Chamar á outra Activity" />

</LinearLayout>
```

A clase Java a activity principal

```
package com.example.u2_50_lanzaractivity;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
```

```

import android.view.View;
import android.widget.CheckBox;
import android.widget.TextView;

public class U2_50_LanzarActivity extends Activity {
    public final static String NOME = "com.example.NOME";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_u2_50__lanzar);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.u2_50__lanzar, menu);
        return true;
    }

    public void onEnviarClick(View view) {
        TextView tvNome = (TextView) findViewById(R.id.et_nome);
        CheckBox chkDestruir = (CheckBox) findViewById(R.id.chk_destruir);

        Intent intent = new Intent(this, RecibirDatos.class);
        intent.putExtra(NOME, tvNome.getText().toString());
        startActivity(intent);

        if (chkDestruir.isChecked())
            finish();
    }
}

```

- **Liña 12:** Definimos unha constante de tipo String, chamada NOME,
- **Liña 31:** Creamos un obxecto de tipo **Intent**. O construtor recibe 2 parámetros:
 - ◆ O primeiro parámetro é unha referencia ao contexto (a clase Activity é unha subclase de Context, por iso poñemos this).
 - ◆ O segundo parámetro é a clase que o sistema ?intentará? cargar (no noso caso o nome da clase asociada á Activity que desexamos cargar).
- **Liña 32:** Pasamos un valor String a través de NOME, definido na liña 12.
- **Liña 33:** Lanzamos a Activity
- **Liña 35-36:** Se o CheckBox do Layout está marcado entón destruímos esta Activity:
 - ◆ Realizar probas marcando e desmarcando esta marca e premendo despois o botón **Back**.

O XML da Activity que recibe a chamada

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >

    <TextView
        android:id="@+id/tv_resultado"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:text="resultado" />

    <Button
        android:id="@+id/btn_pechar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/tv_Resultado"
        android:layout_centerHorizontal="true"
        android:onClick="onPecharActivityClick"
        android:text="Pechar" />

</RelativeLayout>

```

A clase Java da Activity que recibe a chamada

```
package com.example.u2_50_lanzaractivity;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
import android.view.View;
import android.widget.TextView;

public class RecibirDatos extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_recibir_datos);

        Intent intent = getIntent();

        TextView tvResultado = (TextView) findViewById(R.id.tv_resultado);

        tvResultado.setText("Ola "
+ intent.getExtras().getString(U2_50_LanzarActivity.NOME));
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.recibir_datos, menu);
        return true;
    }

    public void onPecharActivityClick(View view) {
        finish();
    }

}
```

- Para recuperar a información na segunda Activity hai que realizalo cando se está creando a Activity, pode ser nun método a parte chamado dende o onCreate() ou no propio onCreate();
- **Liña 17:** Creamos un intent onde recollemos cal foi o intent que iniciou esta Activity.
- **Liña 22:** Collemos a información que viña en NOME e a mostramos nunha etiqueta de texto.
- **Liña 34:** se se preme o botón de pechar destruírase esta Activity.