



La Clase Fragment.

CONCEPTOS.

Un fragmento es una parte aislada o incompleta de la sección modular de una actividad, que tiene su propio ciclo de vida, recibe sus propios eventos de entrada, y que se pueden agregar o quitar mientras que la actividad está en marcha; es decir, es una sub-actividad que se puede reutilizar en diferentes actividades. Ver la figura 1.

Fragment estático o final: Se crea en el archivo XML de la carpeta **Layout** directamente. No se elimina o sustituye.
Fragment dinámico: Se crea desde el código Java y se asocia a un **ViewGroup**, o un **FrameLayout** ya que éste sí se puede eliminar o sustituir por otro **Fragment** u otro contenido. Ver la figura 3.

Características de los fragmentos.

Modularidad: Dividir en fragmentos el código mejora la organización y el mantenimiento de una actividad compleja,

Reusabilidad: Fragmentando el comportamiento, o las interface gráficas, permite compartirlas entre varias actividades.

Adaptabilidad: Representar en fragmentos las diferentes secciones de una interface gráfica y reutilizar las diferentes plantillas que dependen del tamaño y la orientación de la pantalla. Ver la figura 2.

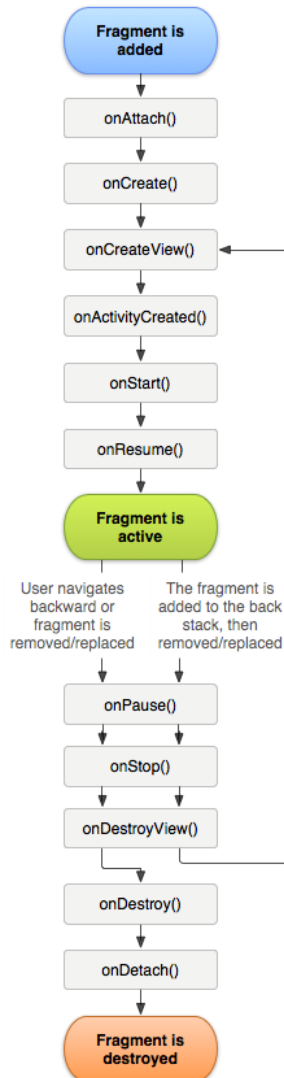


Figura 2. Ciclo de vida de un fragmento mientras se ejecuta su actividad.

Modularidad:



Reusabilidad:



Adaptabilidad:



Figura 1. El aspecto de diseño utilizando los fragmentos.

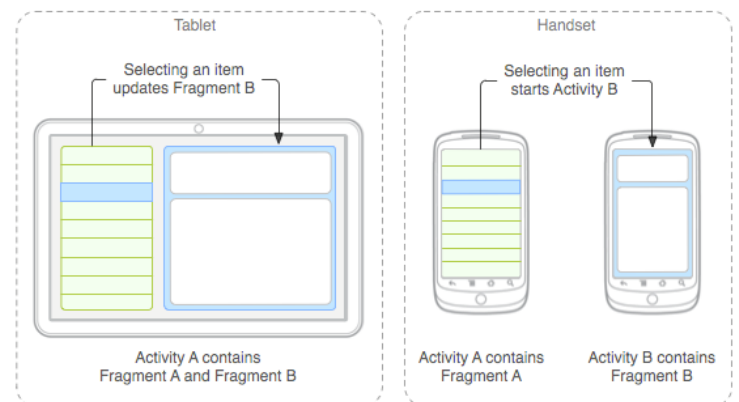


Figura 3. Dos módulos de una UI, definidos con fragmentos se combinan en la actividad de la tableta y se separan en un celular.



Cuando se agrega un fragmento, como parte del diseño de la actividad, se agrega en un `ViewGroup` dentro de la jerarquía de la vista de la actividad y el fragmento define su propia vista de diseño. Se puede insertar un fragmento en la plantilla de la actividad declarando el fragmento en el archivo de la plantilla de la actividad, como un elemento `<fragment>`, o desde el código de aplicación añadiéndolo a un `ViewGroup` existente. Sin embargo, un fragmento no está obligado a ser una parte de la plantilla de la actividad; también se puede utilizar un fragmento sin su propia interfaz de usuario como un participante invisible de la actividad.

Para crear un fragmento, se crea una subclase de `Fragment` (o una subclase existente de la misma). La clase `Fragment` se parece mucho a una actividad. Contiene métodos de devolución de llamada como `onCreate()`, `onStart()`, `onPause()`, y `onStop()`. Si se fragmenta una aplicación de actividades, se mueve el código de los métodos de retorno de llamada de su actividad en los respectivos métodos de retorno de llamada de su fragmento. Se deben aplicar al menos los métodos `onCreate()`, `onCreateView()`, `onPause()` del ciclo de vida. Ver la figura 2.

Otros tipos de fragmentos son `DialogFragment`, `ListFragment` y `PreferenceFragment`.

Agregando una interface de usuario.

Con una clase que hereda de `Fragment` para cargar una plantilla de un archivo XML:

```
public static class EjemploFragmento extends Fragment {
    public View onCreateView(LayoutInflater li, ViewGroup vg, Bundle b) {
        // Inflar la plantilla de este fragmento
        return li.inflate(R.layout.ejemplo_fragmento, vg, false);
    }
}
```

Agregando un fragmento a una actividad.

Hay dos formas de agregar un fragmento a la plantilla de la actividad:

- Declarar el fragmento en el interior del archivo del `layout` de la actividad.

En este caso, se especifican las propiedades de diseño del fragmento como si se tratara de una vista. El siguiente es un ejemplo del archivo de la plantilla de una actividad con dos fragmentos:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <fragment android:name="com.example.escom.fragmentos"
        android:id="@+id/list"
        android:layout_weight="1"
        android:layout_width="0dp"
        android:layout_height="match_parent" />
    <fragment android:name="com.example.escom.fragmentos"
        android:id="@+id/viewer"
        android:layout_weight="2"
        android:layout_width="0dp"
        android:layout_height="match_parent" />
</LinearLayout>
```

Hay tres formas de asignar un ID a un fragmento:

- Suministrar el atributo `android:id` con un identificador único.
- Suministrar el atributo `android:tag` con una cadena única.
- Si no se proporciona alguno de los dos anteriores, el sistema utiliza el ID de la vista del contenedor.

- O, mediante programación, añadiendo el fragmento a un `ViewGroup` existente.



En cualquier momento, mientras que la actividad se está ejecutando, se pueden añadir fragmentos al diseño de la actividad. Sólo se tendrá que especificar un ViewGroup en el cual colocar el fragmento.

Para realizar transacciones de fragmentos en la actividad (añadir, eliminar o reemplazar un fragmento), se deben utilizar las APIs de `FragmentManager`. Se puede obtener una instancia de `FragmentManager` de la actividad, así:

```
FragmentManager fm = getSupportFragmentManager();
FragmentManager ft = fm.beginTransaction();
```

Enseguida, se puede añadir un fragmento utilizando el método `add()`, especificando el fragmento de añadir y la vista en la cual insertar la misma. Por ejemplo:

```
EjemploFragmento ef = new EjemploFragmento();
ft.add(R.id.xfragment_container, ef);
ft.commit();
```

Para añadir un fragmento sin una UI, se añade el fragmento de la actividad utilizando `add(Fragment, String)` (suministrando una única cadena "tag" al fragmento, en lugar del ID del view). Debido a que el fragmento no está asociado con una vista en el diseño de la actividad, no recibe una invocación a `onCreateView()`, por lo que no es necesario implantar ese método.

DESARROLLO

EJEMPLO 1.

Paso 1. Crear un nuevo proyecto **Fragmentos**. En la actividad principal **MainActivity.java**, capturar el siguiente código:

```
import com.example.escom.fragmentos.MiFragmento.FragmentoListener;
import android.app.*;
import android.os.*;
import android.view.*;
import android.view.View.*;
import android.widget.*;

public class MainActivity extends Activity implements FragmentoListener {
    TextView    xtv;
    Button      jbn;
    @Override
    public void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
        xtv = (TextView) findViewById(R.id.xtv);
        jbn = (Button) findViewById(R.id.xbn);
        jbn.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                FragmentManager fm = getSupportFragmentManager();
                Fragment f = fm.findFragmentByTag("editor");
                if (null == f) {
                    FragmentTransaction ft = fm.beginTransaction();
                    ft.add(R.id.xfl, new MiFragmento(), "editor");
                    ft.commit();
                }
                xtv.setText("");
                Toast.makeText(MainActivity.this, "Utilizando Fragment",
                    Toast.LENGTH_LONG).show();
            }
        });
    }
    @Override
    public void digitado(int r, String s) {
        TextView jtv = (TextView) findViewById(R.id.xtv);
        if (r == MiFragmento.OK) {
```



```

        jtv.setText(s);
    }
    FragmentManager fm = getFragmentManager();
    Fragment fe = fm.findFragmentByTag("editor");
    FragmentTransaction ft = fm.beginTransaction();
    ft.remove(fe);
    ft.commit();
}
}

```

Paso 2. En el archivo `activity_main.xml` predeterminado, capturar el siguiente código:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <TextView
        android:id="@+id/xtv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Hola" />
    <Button
        android:id="@+id/xbn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Digitar" >
    </Button>
    <FrameLayout
        android:id="@+id/xfl"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
    <fragment
        class="com.example.escom.fragmentos.MiFragmento"
        android:id="@+id/fragmento"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</LinearLayout>

```

Paso 3. En la carpeta `java/com.example.escom.fragmentos` del repositorio, crear un nuevo archivo para la clase `MiFragmento.java`, y allí capturar el siguiente código para el fragmento:

```

import android.app.*;
import android.os.*;
import android.view.*;
import android.view.View.OnClickListener;
import android.widget.*;
public class MiFragmento extends Fragment {
    public final static int OK = 0;
    public final static int CANCEL = 1;
    private FragmentoListener fl;
    @Override
    public void onAttach(Activity a) {
        super.onAttach(a);
        if (a instanceof FragmentoListener) {
            fl = (FragmentoListener) a;
        }
    }
    public interface FragmentoListener {

```



```

        public void digitado(int resultado, String texto);
    }
    @Override
    public View onCreateView(LayoutInflater li, ViewGroup vg, Bundle b) {
        View v = li.inflate(R.layout.activity_mifragmento, vg, false);
        ((Button) v.findViewById(R.id.xbnA)).setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                botonDigitado(v);
                Toast.makeText(getActivity(), "Desde MiFragmento",
                    Toast.LENGTH_LONG).show();
            }
        });
        ((Button) v.findViewById(R.id.xbnC)).setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                botonDigitado(v);
            }
        });
        return v;
    }
    public void botonDigitado(View v) {
        if (null == fl)
            return;
        if (((Button) v).getText().equals("Aceptar"))
            fl.digitado(OK, ((EditText)
                getActivity().findViewById(R.id.xet)).getText().toString());
        else
            fl.digitado(CANCEL, "");
    }
}

```

Paso 4. En la carpeta `res/layout` del repositorio, crear un nuevo archivo para el XML del fragmento `activity_mifragmento.xml`, y allí capturar el siguiente código para la plantilla del fragmento:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical" >
    <EditText
        android:id="@+id/xet"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Dato del Fragment: " />
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <Button
            android:id="@+id/xbnA"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Aceptar" >
        </Button>
        <Button
            android:id="@+id/xbnC"

```

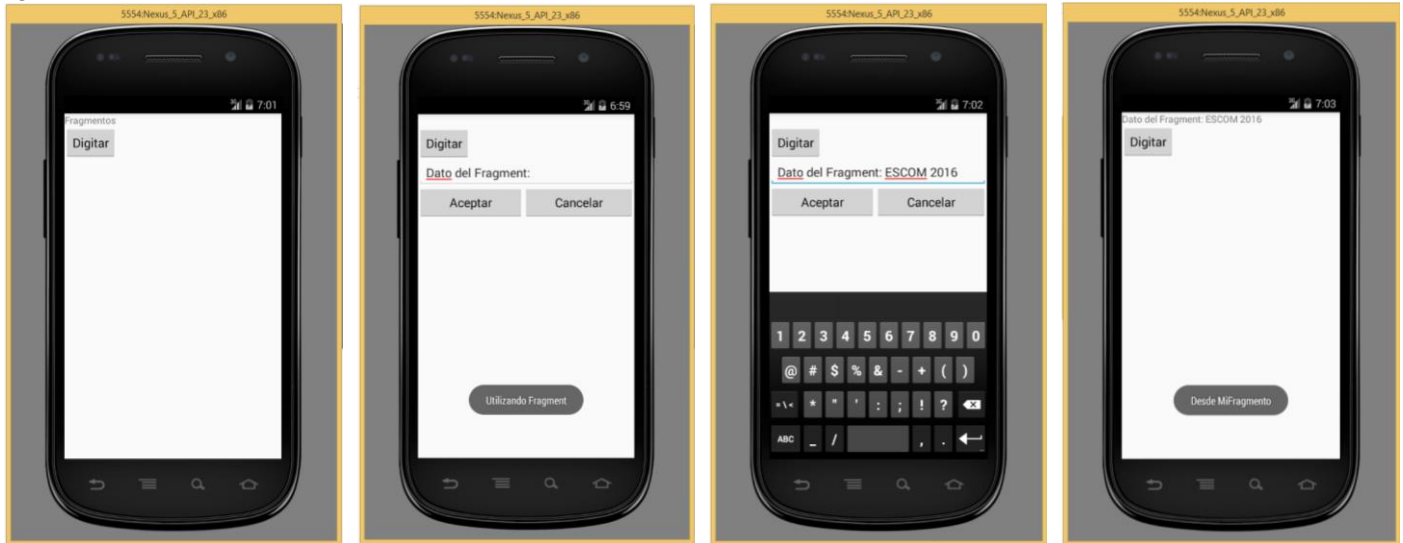


```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Cancelar" >
    </Button>
</LinearLayout>
</LinearLayout>

```

Paso 5. Por último, ejecutar la aplicación. Seleccionar el botón **Digitar**. En ese momento se invoca al fragmento y se muestra su plantilla, solicitando el ingreso de un dato, por ejemplo ESCOM 2016. Digitar el botón **Aceptar**. El mensaje del fragmento se muestra en el **TextView**, `xtv` de la actividad.



EJEMPLO 2.

En este ejemplo, por sencillez, primero se crean los archivos **Java** y luego los **XML**, pero es indistinto dado que cada actividad o fragmento se puede programar junto con su archivo **XML** correspondiente. .

Crear un nuevo proyecto **Fragmentos2**. En la carpeta `java/com.example.escom.fragmentos2`, capturar los cinco siguientes archivos **Java**.

Paso 1. En la carpeta `java/com.example.escom.fragmentos2`, en la actividad principal `MainActivity.java`, capturar el siguiente código:

```

import com.example.escom.fragmentos.FragmentListado.GruposListener;
import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity implements GruposListener {
    FragmentListado fl;
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
        fl =
(FragmentListado) getSupportFragmentManager().findFragmentById(R.id.FrgListado);
        fl.setGruposListener(this);
    }
    @Override

```



```
public void onGrupoSeleccionado(Grupo c) {
    boolean bo = (getSupportFragmentManager().findFragmentById(R.id.FrgDetalle) !=
null);
    if(bo) {
        ((FragmentDetalle) getSupportFragmentManager().findFragmentById(
R.id.FrgDetalle) ).mostrarDetalle( c.getTexto() );
    }
    else {
        Intent i = new Intent(this, DetalleActivity.class);
        i.putExtra(DetalleActivity.EXTRA_TEXTO, c.getTexto());
        startActivity(i);
    }
}
}
```

Paso 2. Crear una nueva actividad `DetalleActivity.java` y capturar el siguiente código:

```
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
public class DetalleActivity extends AppCompatActivity {
    public static final String EXTRA_TEXTO = "com.example.escom.fragmentos.EXTRA_TEXTO";
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_detalle);
        FragmentDetalle fd = (FragmentDetalle)
getSupportFragmentManager().findFragmentById(R.id.FrgDetalle);
        fd.mostrarDetalle(getIntent().getStringExtra(EXTRA_TEXTO));
    }
}
```

Paso 3. Crear una nueva actividad `FragmentDetalle.java` y capturar el siguiente código:

```
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.*;
import android.widget.TextView;
public class FragmentDetalle extends Fragment {
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle b) {
        return inflater.inflate(R.layout.fragment_detalle, container, false);
    }
    public void mostrarDetalle(String texto) {
        TextView tv = (TextView) getView().findViewById(R.id.xtvDetalle);
        tv.setText(texto);
    }
}
```

Paso 4. Crear una nueva actividad `FragmentListado.java` y capturar el siguiente código:

```
import android.app.Activity;
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.*;
import android.widget.AdapterView.*;
import android.widget.*;
public class FragmentListado extends Fragment {
    private Grupo[] datos = new Grupo[]{
        new Grupo("\nEstudiante 1", "Calificación 1", "\nReporte de aprovechamiento 1"),
        new Grupo("\nEstudiante 2", "Calificación 2", "\nReporte de aprovechamiento 2"),
    }
```



```

        new Grupo("\nEstudiante 3", "Calificación 3", "\nReporte de aprovechamiento 3"),
        new Grupo("\nEstudiante 4", "Calificación 4", "\nReporte de aprovechamiento 4"),
        new Grupo("\nEstudiante 5", "Calificación 5", "\nReporte de aprovechamiento 5")
    };
    private ListView lv;
    private GruposListener cl;
    @Override
    public View onCreateView(LayoutInflater li, ViewGroup vg, Bundle b) {
        return li.inflate(R.layout.fragment_listado, vg, false);
    }
    @Override
    public void onActivityCreated(Bundle b) {
        super.onActivityCreated(b);
        lv = (ListView) getView().findViewById(R.id.xlvListado);
        lv.setAdapter(new AdaptadorGrupos(this));
        lv.setOnItemClickListener(new OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> list, View v, int pos, long id) {
                if (cl!=null) {
                    cl.onGrupoSeleccionado((Grupo) lv.getAdapter().getItem(pos));
                }
            }
        });
    }
    class AdaptadorGrupos extends ArrayAdapter<Grupo> {
        Activity a;
        TextView tv1, tv2;
        AdaptadorGrupos(Fragment f) {
            super(f.getActivity(), R.layout.listitem_grupo, datos);
            this.a = f.getActivity();
        }
        public View getView(int i, View vi, ViewGroup vg) {
            LayoutInflater li = a.getLayoutInflater();
            View v = li.inflate(R.layout.listitem_grupo, null);
            tv1 = (TextView)v.findViewById(R.id.xtvDe);
            tv1.setText(datos[i].getDe());
            tv2 = (TextView)v.findViewById(R.id.xtvAsunto);
            tv2.setText(datos[i].getAsunto());
            return(v);
        }
    }
    public interface GruposListener{
        void onGrupoSeleccionado(Grupo c);
    }
    public void setGruposListener(GruposListener l) {
        this.cl=l;
    }
}

```

Paso 5. Crear una nueva actividad Grupo.java y capturar el siguiente código:

```

public class Grupo{
    private String de, asunto,texto;
    public Grupo(String de, String as, String te){
        this.de = de;
        this.asunto = as;
        this.texto = te;
    }
    public String getDe(){

```




```
        return de;
    }
    public String getAsunto(){
        return asunto;
    }
    public String getTexto(){
        return texto;
    }
}
```

En los siguientes pasos del 6 al 10, en la carpeta `res/layout`, capturar los cinco siguientes archivos XML.

Paso 6. En el archivo `activity_main.xml`, capturar el siguiente código:

```
<?xml version="1.0" encoding="utf-8"?>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
    class="com.example.escom.fragmentos.FragmentListado"
    android:id="@+id/FrgListado"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
```

Paso 7. Crear el archivo `activity_detalle.xml` y capturar el siguiente código:

```
<?xml version="1.0" encoding="utf-8"?>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
    class="com.example.escom.fragmentos.FragmentDetalle"
    android:id="@+id/FrgDetalle"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
```

Paso 8. Crear el archivo `fragment_detalle.xml` y capturar el siguiente código:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#FFBBBBBB" >
    <TextView
        android:id="@+id/xtvDetalle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>
```

Paso 9. Crear el archivo `fragment_listado.xml` y capturar el siguiente código:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Actividad con Fragmentos" />
    <ListView
        android:id="@+id/xlvListado"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
</ListView>
```



```
</LinearLayout>
```

Paso 10. Crear el archivo `listitem_grupo.xml` y capturar el siguiente código:

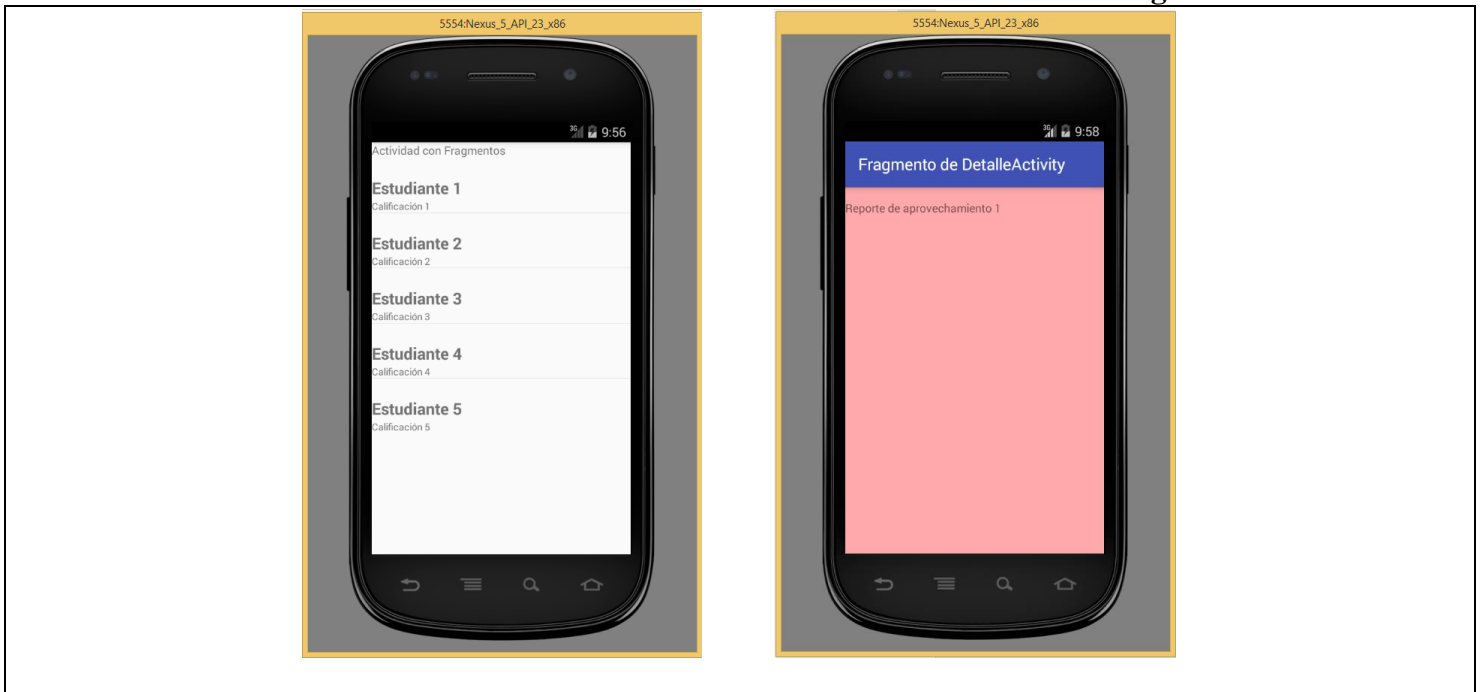
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <TextView android:id="@+id/xtvDe"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textStyle="bold"
        android:textSize="20sp" />
    <TextView android:id="@+id/xtvAsunto"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textStyle="normal"
        android:textSize="12sp" />
</LinearLayout>
```

Paso 11. En la carpeta `app/manifests`, abrir el archivo `AndroidManifest.xml`. Para modificarlo, se inserta la etiqueta de la nueva actividad `DetalleActivity` indicada con letras negritas, entre las etiquetas `<application>` y `</application>`:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.escom.fragmentos">
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name"
            android:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <b>activity</b>
            android:name=".DetalleActivity"
            android:label="Fragmento de DetalleActivity">
        </b>activity</b>
    </application>
</manifest>
```

Paso 12. Por último, ejecutar la aplicación. Seleccionar alguna opción de estudiante. En ese momento se invoca a la plantilla que contiene un fragmento con un mensaje.



NOTA. Generar un reporte y documentarlo con las imágenes obtenidas durante la ejecución de los ejercicios. Guardar el reporte con la sintaxis **NombreTareaGrupo.zip** y enviarlo al sitio indicado por el profesor.