

**UNIVERSIDAD
DE ANTIOQUIA**
1 8 0 3

Sesión 4. Fragments

Ing. Edwin Andrés Cubillos Vega Msc.





Introducción

Fragments Estáticos

Fragments Dinámicos

Swipe Tabs



Introducción

Fragments Estáticos

Fragments Dinámicos

Swipe Tabs

- ❖ A partir de Android 3.0 (API 11)

```
import android.app.Fragment;
```

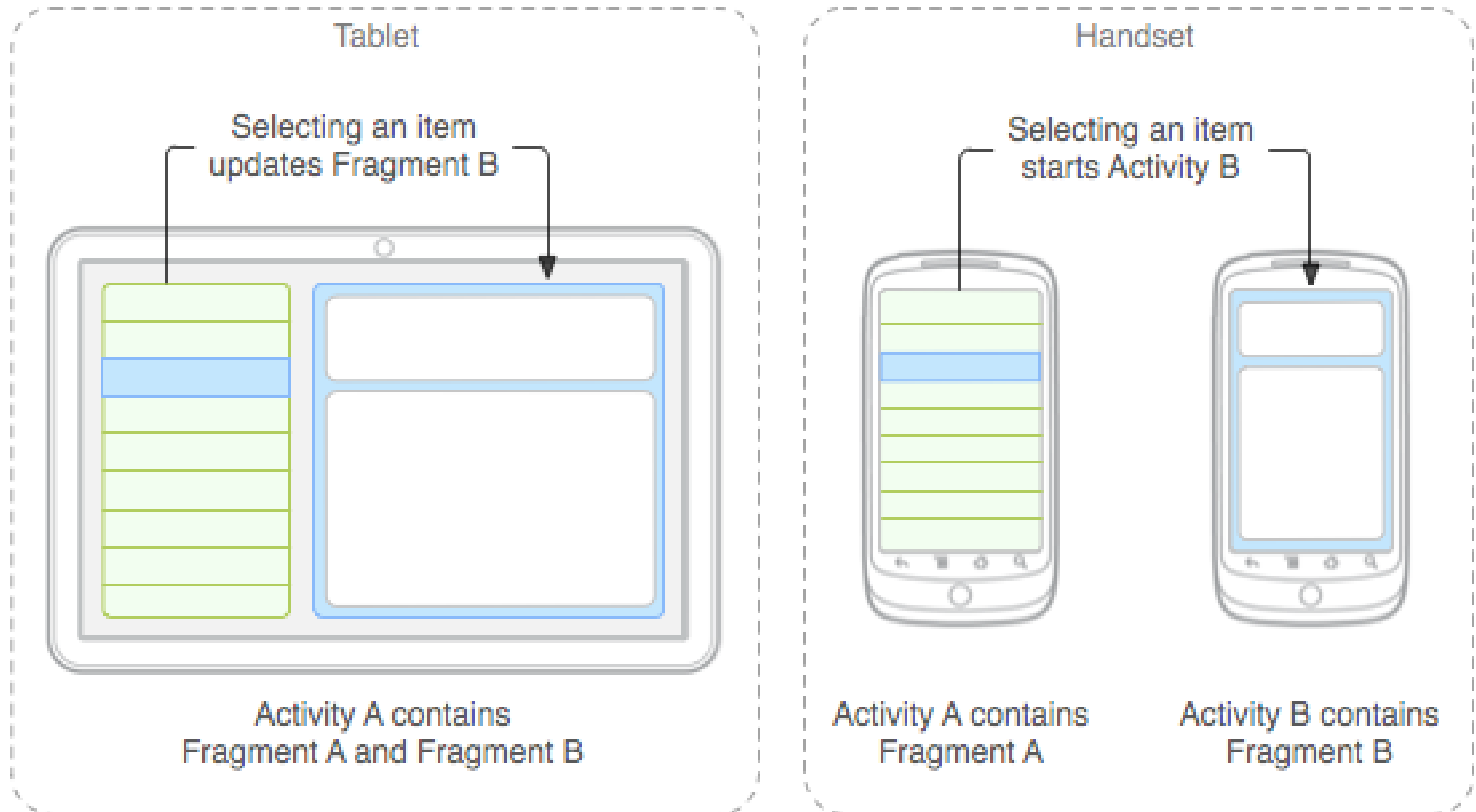
- ❖ Se puede utilizar desde la versión 1.6 (API 4)

```
import android.support.v4.app.Fragment;
```

- ❖ Surge debido al crecimiento de tablets de gran tamaño la interfaz gráfica de una app no se adaptaba a un dispositivo con 4 o 5 pulgadas más de tamaño
- ❖ Un **fragment** es una porción de la interfaz de usuario que puede añadirse o eliminarse de forma independiente al resto de los elementos de la actividad.
- ❖ Esto permite dividir la interfaz en varias porciones de forma que podamos diseñar diversas configuraciones de pantalla: tamaño y orientación.
- ❖ Puede ser añadido o eliminado mientras la actividad esta en ejecución

Fragments

❖ Ejemplo



- ❖ Antes de los fragments era necesario implementar diferentes actividades con diferentes layouts.
- ❖ Con los fragments, se coloca el listado de correos en un fragment y la vista de detalle en otro, cada uno con su lógica.
- ❖ Solo faltaría definir los layouts para cada configuración de pantalla.

Fragments

Ciclo de Vida ([Info](#))

Añadimos el Fragment a la Activity

onAttach()

Se adjunta el Fragment a la Activity

onCreate()

Creación inicial del Fragment
Uso Recomendado

onCreateView()

Donde se asocia el Layout al Fragment
Uso Recomendado

onActivityCreated()

Se ha completado en onCreate de la Activity

onViewStateRestored()

El estado guardado de la jerarquía de la vista ha sido restaurado

onStart()

Se hace el Fragment visible al usuario

onResume()

El Fragment permite la interacción con el usuario

Fragment está activo

El usuario navega hacia atrás o el Fragment es eliminado/remplazado

El Fragment es añadido a la parte inferior de la pila, entonces es eliminado/remplazado

onPause()

El Fragment ya no interactúa con el usuario
Uso Recomendado

onStop()

El Fragment ya no es visible para el usuario

onDestroyView()

Limpieza de las vistas asociadas al Fragment

El Fragment vuelve al Layout desde la parte inferior de la pila

onDestroy()

Limpieza del estado del Fragment

onDetach()

Justo antes de que el Fragment deje de estar asociado a su actividad

El Fragment es destruido

Fragments

- ❖ Para crear un fragment se debe crear una subclase de `Fragment`.
- ❖ Se implementa en el método `onCreateView()`
- ❖ Retorna un `View`

Contenedor de otras vistas

```
public static class ExampleFragment extends Fragment {  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
                             Bundle savedInstanceState) {  
        // Inflate the layout for this fragment  
        return inflater.inflate(R.layout.example_fragment, container, false);  
    }  
}
```

ID del recurso a inflar Contenedor

indicar si el inflado del layout debe ser insertado en el `ViewGroup` (`false` porque directamente se esta insertando el layout en el `ViewGroup`)

Agregar un fragment a una actividad

Se debe agregar el fragment en el xml del layout de la actividad principal

```
<?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.news.ArticleListFragment"
        android:id="@+id/list"
        android:layout_weight="1"
        android:layout_width="0dp"
        android:layout_height="match_parent" />
    <fragment android:name="com.example.news.ArticleReaderFragment"
        android:id="@+id/viewer"
        android:layout_weight="2"
        android:layout_width="0dp"
        android:layout_height="match_parent" />
</LinearLayout>
```

- ❖ Para una API inferior a la 11 se utiliza una librería de compatibilidad.
- ❖ La actividad debe heredar de Fragment Activity

```
import android.os.Bundle;
import android.support.v4.app.FragmentActivity;

public class MainActivity extends FragmentActivity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.news_articles);
    }
}
```

- ❖ Pero si se utiliza la librería **v7 appcompat library** se puede dejar la herencia de AppCompatActivity

❖ Tipos de Fragments:

1. **Fragment Estáticos:** los declarados en en layout mediante la etiqueta <fragment>, como si fueran un botón. Se utilizan principalmente para reutilizar código en actividades.
2. **Fragment Dinámicos:** Los más interesantes, podemos quitarlos, ponerlos y reemplazarlos.



Introduccion



Fragments Estáticos



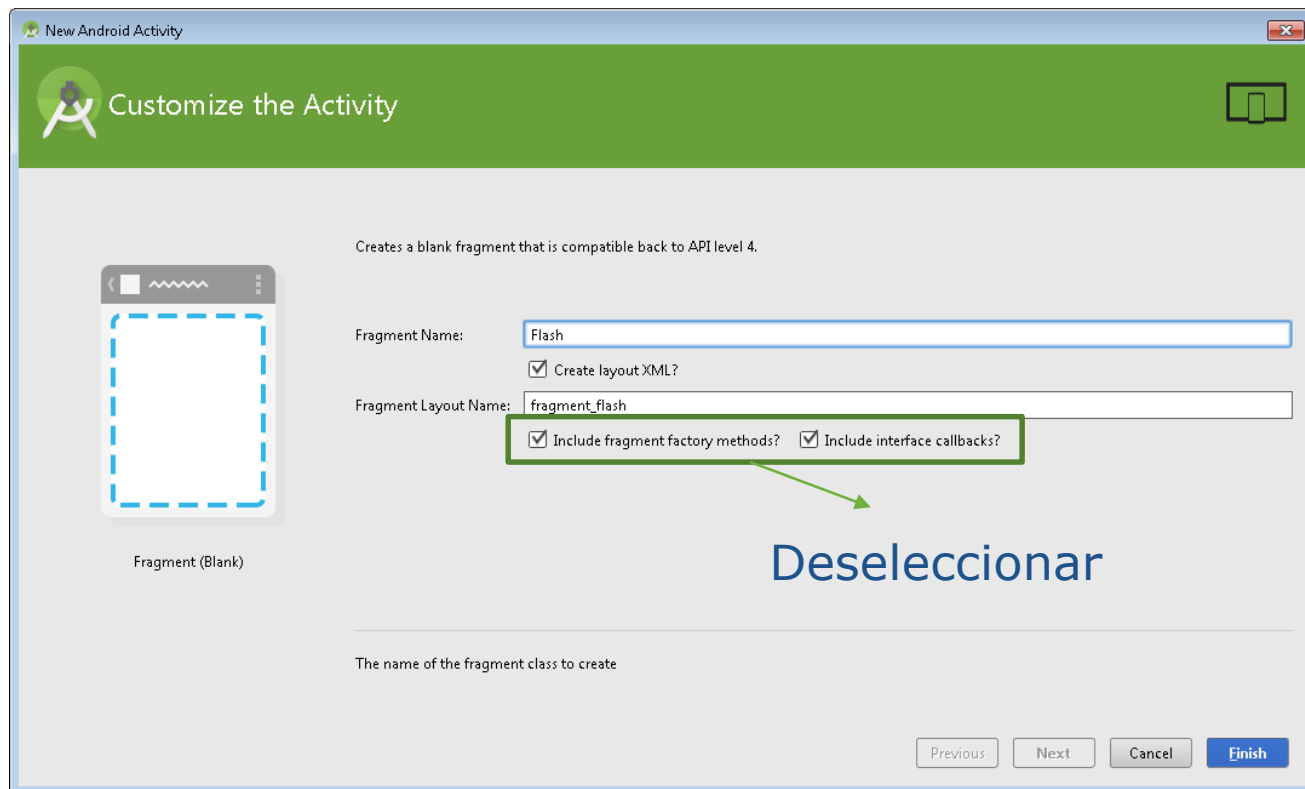
Fragments Dinámicos



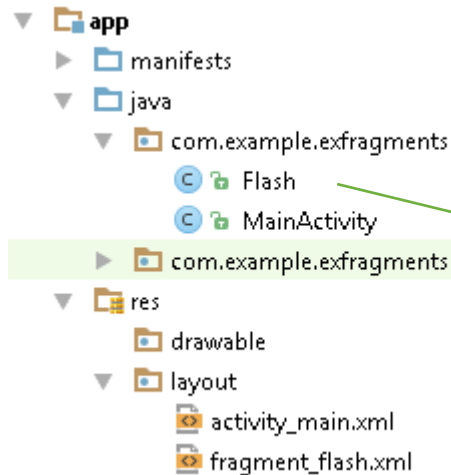
Swipe Tabs

Fragments Estáticos

- ❖ Crear nuevo proyecto (Sesion6) con una actividad en Blanco
- ❖ Click derecho en res -> New Fragment -> Fragment Blank



Fragments Estaticos



```
public class FlashFragment extends Fragment {  
  
    public FlashFragment() {  
        // Required empty public constructor  
    }  
  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
        Bundle savedInstanceState) {  
        // Inflate the layout for this fragment  
        final View view = inflater.inflate(R.layout.fragment_flash, container, false);  
        return view;  
    }  
}
```

```
<FrameLayout 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" tools:context="com.example.exfragments.Flash">  
  
    <!-- TODO: Update blank fragment layout -->  
    <TextView android:layout_width="match_parent" android:layout_height="match_parent"  
        android:text="Hello blank fragment" />  
  
</FrameLayout>
```

Fragment Estáticos

- ❖ Editar el Fragment con información a su gusto
- ❖ Primero se definen los estilos

```
<style name="linearLayoutStyle" parent="@android:style/Theme">
    <item name="android:paddingBottom">@dimen/activity_vertical_margin</item>
    <item name="android:paddingLeft">@dimen/activity_horizontal_margin</item>
    <item name="android:paddingRight">@dimen/activity_horizontal_margin</item>
    <item name="android:paddingTop">@dimen/activity_vertical_margin</item>
    <item name="android:layout_width">match_parent</item>
    <item name="android:layout_height">match_parent</item>
    <item name="android:orientation">vertical</item>
    <item name="android:background">@color/white</item>
</style>

<style name="textViewStyle" parent="@android:style/TextAppearance">
    <item name="android:layout_width">match_parent</item>
    <item name="android:layout_height">wrap_content</item>
    <item name="android:textColor">@color/black</item>
    <item name="android:gravity">center</item>
</style>

<style name="textViewStyle.Title">
    <item name="android:textSize">30sp</item>
</style>

<style name="textViewStyle.Content">
    <item name="android:textSize">20sp</item>
</style>

<style name="imageViewStyle" parent="@android:style/Theme">
    <item name="android:layout_gravity">center</item>
    <item name="android:layout_width">200dp</item>
    <item name="android:layout_height">200dp</item>
</style>
```



Fragment Estáticos

❖ Luego se edita el fragment


fragment_flash.xml ×

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    tools:context="com.edwinacubillos.fragmentsex.Flash">


    <LinearLayout
        style="@style/linearLayoutStyle">
        <TextView
            style="@style/textViewStyle.Title"
            android:text="FLASH"/>
        <ImageView
            style="@style/imageViewStyle"
            android:src="@drawable/flash"/>
        <TextView
            style="@style/textViewStyle.Content"
            android:text="Super speed, intangibility, superhuman agility"/>
        </LinearLayout>

    </FrameLayout>
```

Preview

 Nexus 4

 AppTheme

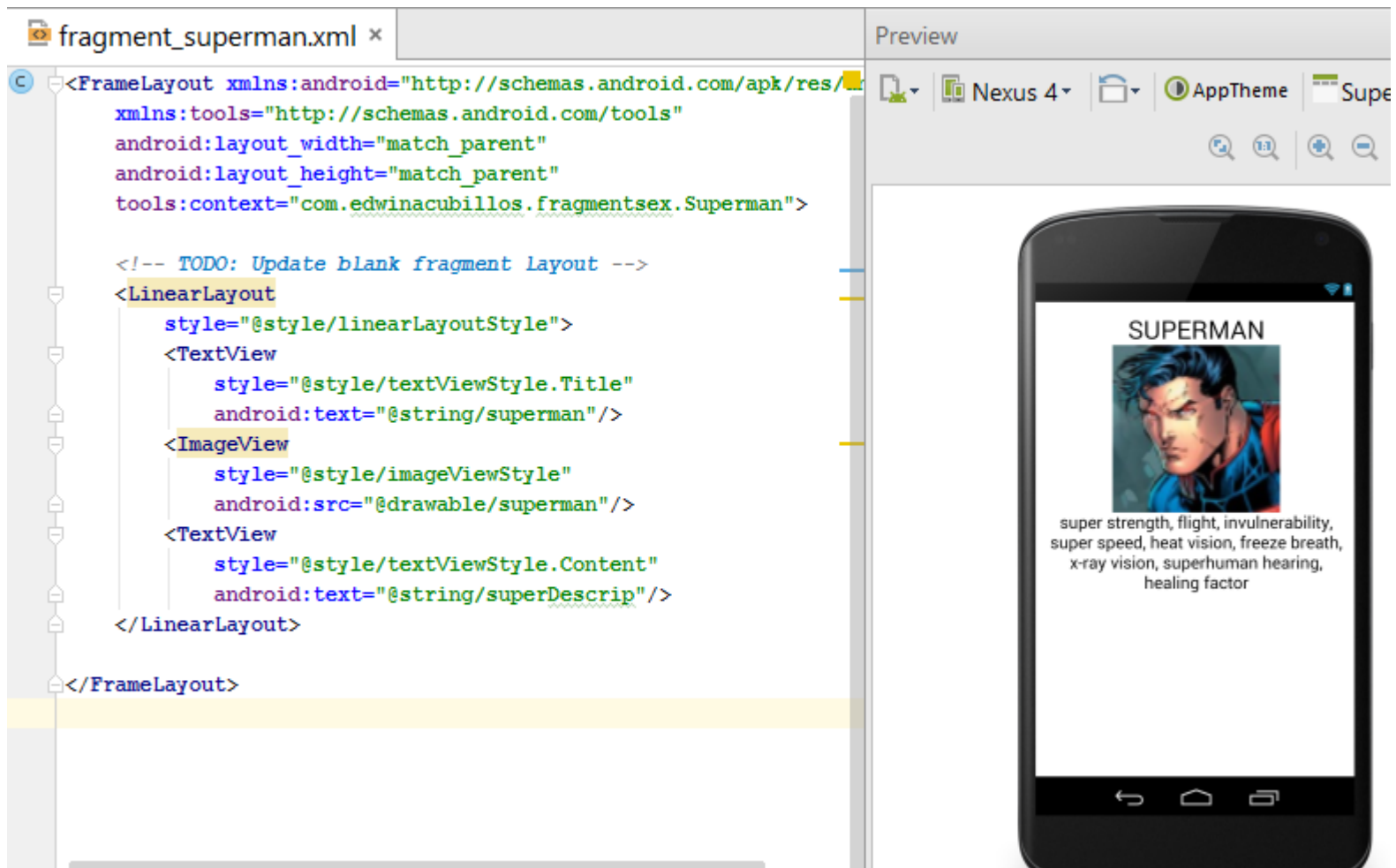
 Flash





Fragment Estáticos

❖ Ahora creamos otro Fragment para Superman



The screenshot displays the Android Studio interface with the `fragment_superman.xml` file open. The XML code defines a `LinearLayout` containing a title `TextView`, a `ImageView` of Superman, and a description `TextView`.

```
<?xml version="1.0" encoding="utf-8" android:namespace="http://schemas.android.com/apk/res/android"
    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"
    tools:context="com.edwinacubillos.fragmentsex.Superman">

    <!-- TODO: Update blank fragment layout -->
    <LinearLayout
        style="@style/linearLayoutStyle">
        <TextView
            style="@style/textViewStyle.Title"
            android:text="@string/superman"/>
        <ImageView
            style="@style/imageViewStyle"
            android:src="@drawable/superman"/>
        <TextView
            style="@style/textViewStyle.Content"
            android:text="@string/superDescrip"/>
        </LinearLayout>
    </FrameLayout>
```

The preview window shows the visual result on a Nexus 4 device. The screen displays the word "SUPERMAN" at the top, followed by a portrait of Superman, and then a list of his powers: "super strength, flight, invulnerability, super speed, heat vision, freeze breath, x-ray vision, superhuman hearing, healing factor".

Fragment Estáticos

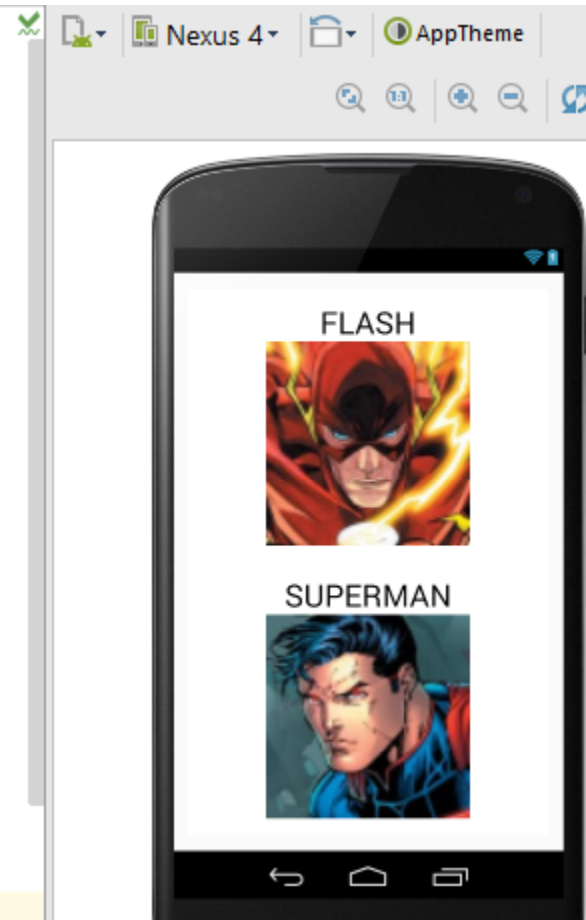
- ❖ Después de crear los dos fragments ponemos los dos en la actividad principal

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:orientation="vertical"
    tools:context="com.edwinacubillos.fragmentsex.MainActivity">

    <fragment android:name="com.edwinacubillos.fragmentsex.Flash"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        tools:layout="@layout/fragment_flash"
        android:id="@+id/fragmentFlash"/>

    <fragment android:name="com.edwinacubillos.fragmentsex.Superman"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        tools:layout="@layout/fragment_superman"
        android:id="@+id/fragmentSuperman" />

</LinearLayout>
```



Fragment Estáticos

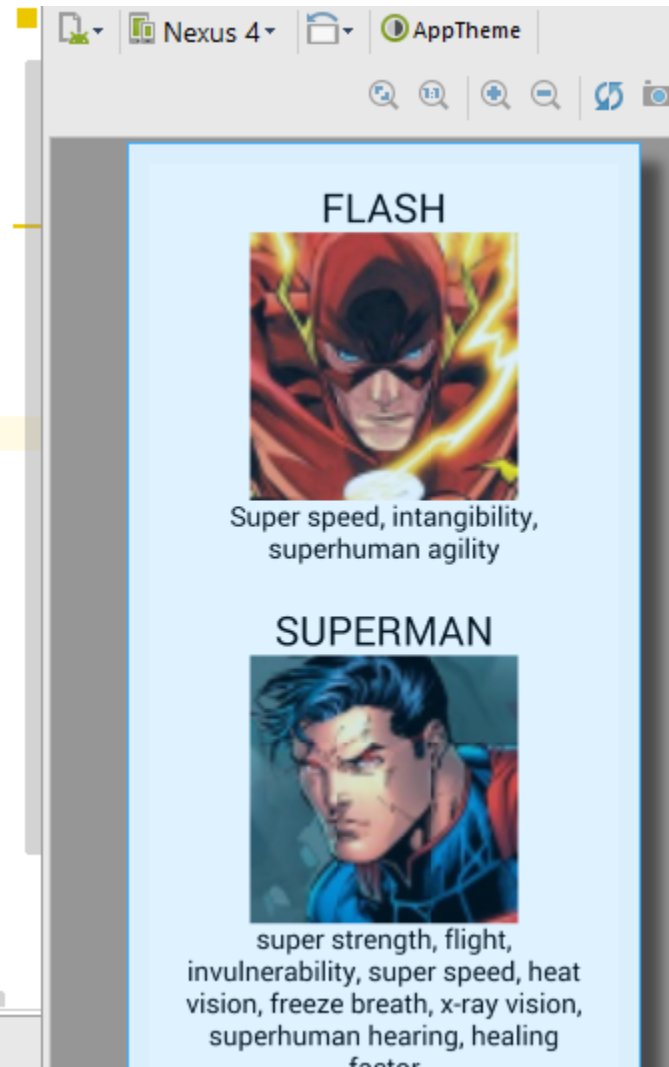
❖ Si queremos ver el fragment completo?

```
<ScrollView 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">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:paddingBottom="@dimen/activity_vertical_margin"
        android:paddingLeft="@dimen/activity_horizontal_margin"
        android:paddingRight="@dimen/activity_horizontal_margin"
        android:paddingTop="@dimen/activity_vertical_margin"
        android:orientation="vertical"
        tools:context="com.edwinacubillos.fragmentsex.MainActivity">

        <fragment android:name="com.edwinacubillos.fragmentsex.Flash"
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            tools:layout="@layout/fragment_flash"
            android:id="@+id/fragmentFlash"/>

        <fragment android:name="com.edwinacubillos.fragmentsex.Superman"
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            tools:layout="@layout/fragment_superman"
            android:id="@+id/fragmentSuperman" />

    </LinearLayout>
</ScrollView>
```





Introducción



Fragments Estáticos



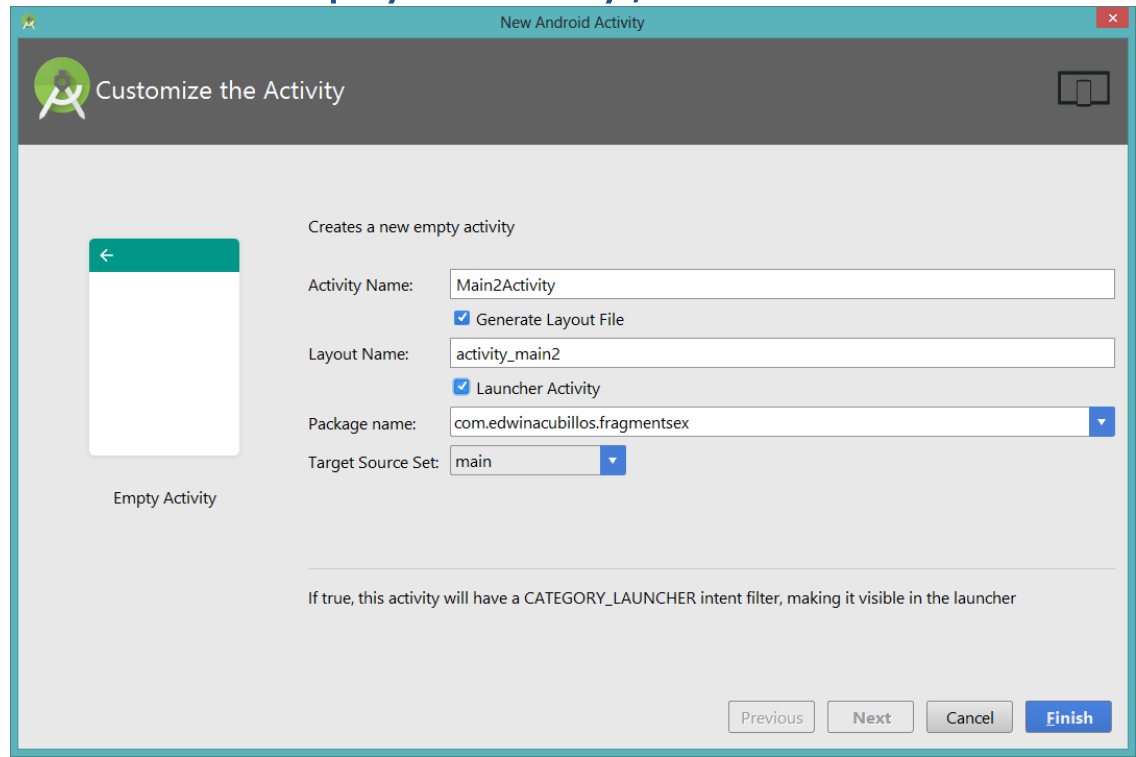
Fragments Dinámicos



Swipe Tabs

Fragments Dinámicos

- ❖ Permite cambiar entre unos y otro fragments:
 - Según la disposición del dispositivo
 - Tamaño de la pantalla.
 - Seleccionando por medio de un menú
- ❖ Crear una nueva actividad – Empty Activity, seleccionar Launcher Activity



Fragments Dinámicos

- ❖ Teniendo los dos fragments vamos a crear un Fragment Dinámico
- ❖ Se implementará en la actividad principal



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context="com.edwinacubillos.fragmentsex.Main2Activity">

</LinearLayout>
```



Fragments Dinámicos

- ❖ Se implementarán por código
- ❖ La Actividad principal sigue heredando de Activity
- ❖ Se crean dos objetos nuevos

```
FragmentManager fm = getFragmentManager();  
FragmentTransaction ft = fm.beginTransaction();
```

1. **FragmentManager**: Permite administrar los fragmentos en una actividad
 2. **FragmentTransaction**: permite añadir: add(), remover: remove(), reemplazar: replace() y realizar otras acciones con los fragments
- ❖ Se crea el objeto tipo fragment y luego se agrega

```
FlashFragment fragment = new FlashFragment();  
ft.add(android.R.id.content, fragment).commit();
```


Fragments Dinámicos

❖ Para visualizar un solo fragment

```
package com.edwinacubillos.fragmentsex;

import android.app.FragmentManager;
import android.app.FragmentTransaction;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class Main2Activity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);

        FragmentManager fm = getFragmentManager();
        FragmentTransaction ft = fm.beginTransaction();

        FlashFragment fragment = new FlashFragment();
        ft.add(android.R.id.content, fragment).commit();
    }
}
```

Se debe llamar este método para que se lleve a cabo

Fragments Dinámicos

FlashFragment.java ×

```
package com.edwinacubillos.fragmentsex;
```

```
import android.os.Bundle;
```

```
import android.support.v4.app.Fragment;
```

```
import android.view.LayoutInflater;
```

```
import android.view.View;
```

```
import android.view.ViewGroup;
```

```
import android.app.Fragment;
```

```
/**
```

```
 * A simple {@link Fragment} subclass.
```

```
 */
```

```
public class FlashFragment extends Fragment {
```

```
    public FlashFragment() {
```

```
        // Required empty public constructor
```

```
    }
```

```
    @Override
```

```
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
                             Bundle savedInstanceState) {
```

```
        // Inflate the layout for this fragment
```

```
        final View view = inflater.inflate(R.layout.fragment_flash, container, false);
```

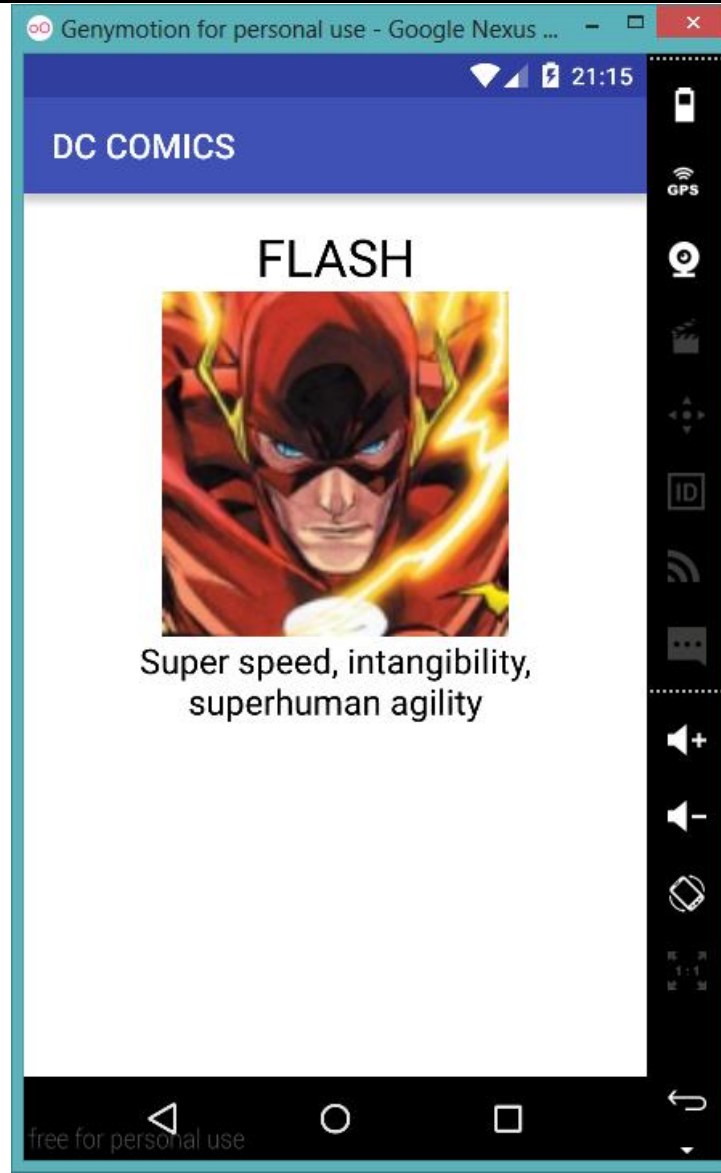
```
        return view;
```

```
    }
```

```
}
```

Se debe cambiar la librería por
La que no tiene compatibilidad

Fragments Dinámicos





Fragments Dinámicos

- ❖ Si se utiliza una API inferior a la 11 (3.0) se debe cambiar el método `getFragmentManager()` por `getSupportFragmentManager()`
- ❖ En este caso la actividad principal debe heredar de `FragmentManager` y no de `AppCompatActivity`

Fragments Dinámicos

- ❖ Para cambiar el fragment utilizando un menú
- ❖ Crear los ítems en el archivo menú.xml

```
<menu 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" tools:context=".MainActivity">

    <item android:id="@+id/menu_flash"
          android:title="@string/flash"/>
    <item android:id="@+id/menu_superman"
          android:title="@string/SuperMan"/>

</menu>
```

Flash

Superman

(Note: Menu preview is only approximate)



Fragments Dinámicos

- ❖ Función onCreate no cambia, solo es configurar los menú

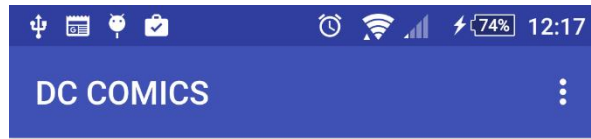
```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();

    fragmentManager fm = getFragmentManager();
    FragmentTransaction ft = fm.beginTransaction();

    if (id == R.id.mFlash) {
        FlashFragment fragment = new FlashFragment();
        ft.replace(android.R.id.content, fragment).commit();
    }
    if (id == R.id.mSuperman) {
        SupermanFragment fragment = new SupermanFragment();
        ft.replace(android.R.id.content, fragment).commit();
    }
    return super.onOptionsItemSelected(item);
}
```

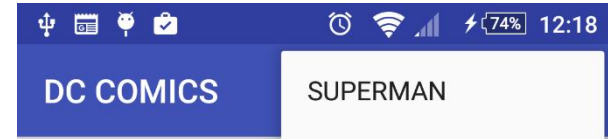
Fragments Dinámicos



FLASH



Super speed, intangibility,
superhuman agility



SUPERMAN



super strength, flight, invulnerability,
super speed, heat vision, freeze
breath, x-ray vision, superhuman
hearing, healing factor





Introduccion



Fragments Estáticos

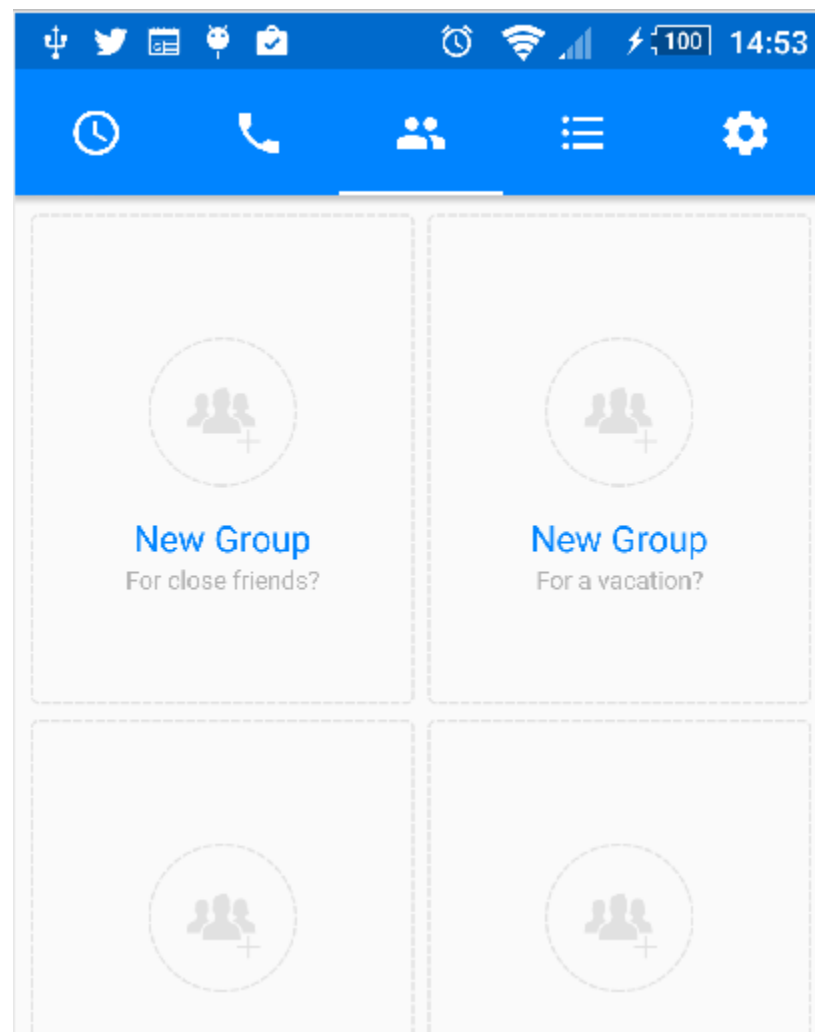
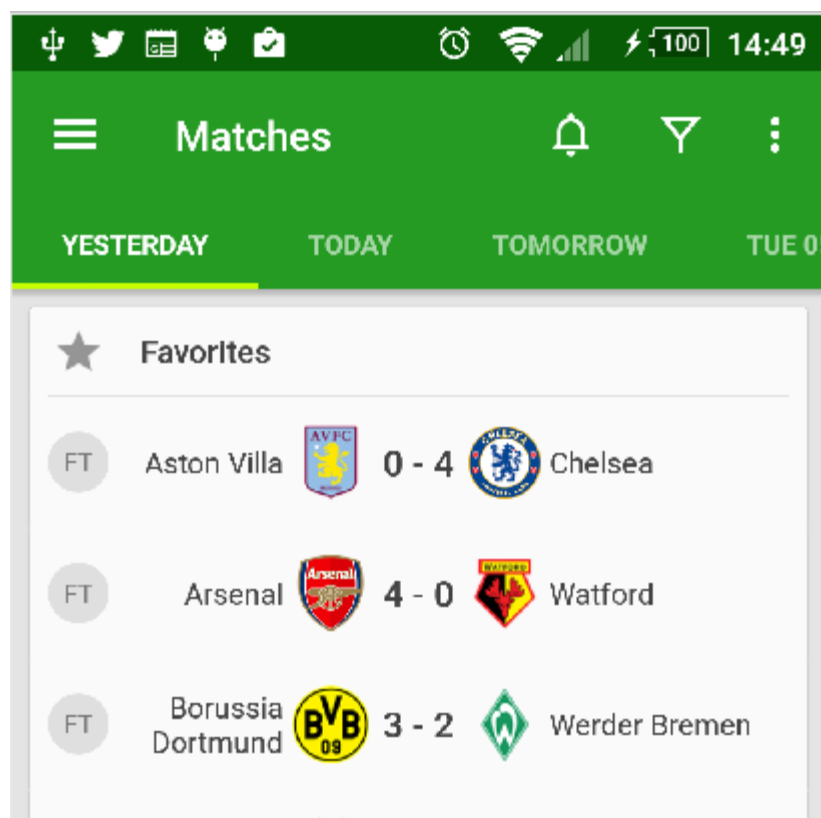


Fragments Dinámicos



Swipe Tabs

Tabs



- ❖ Se implementará la vista tipo Swipe
- ❖ Permite desplazamiento horizontal con el touch
- ❖ Generar un nuevo Módulo
- ❖ Copiar los fragments (layout y clase), strings, styles y colors
- ❖ Se configura en el layout principal un contenedor tipo ViewPager
- ❖ Luego se configura un PageAdapter



Cambiar el layout de base de la actividad principal

- ❖ Permite que los elementos que contiene puedan realizar animaciones coordinadas

```
<?xml version="1.0" encoding="utf-8"?>  
<android.support.design.widget.CoordinatorLayout  
    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:id="@+id/main_content"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:fitsSystemWindows="true"  
    tools:context="com.edwinacubillos.tabbedexample.MainActivity">
```

Permite modificar barra de estado



- ❖ Adicionar un AppBarLayout es un LinearLayout vertical
- ❖ Diseñado para usarlo dentro de un CoordinatorLayout
- ❖ Puede contener un Toolbar y un TabLayout

```
<android.support.design.widget.AppBarLayout  
    android:id="@+id/appbar"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:paddingTop="@dimen/appbar_padding_top"  
    android:theme="@style/AppTheme.AppBarOverlay">
```

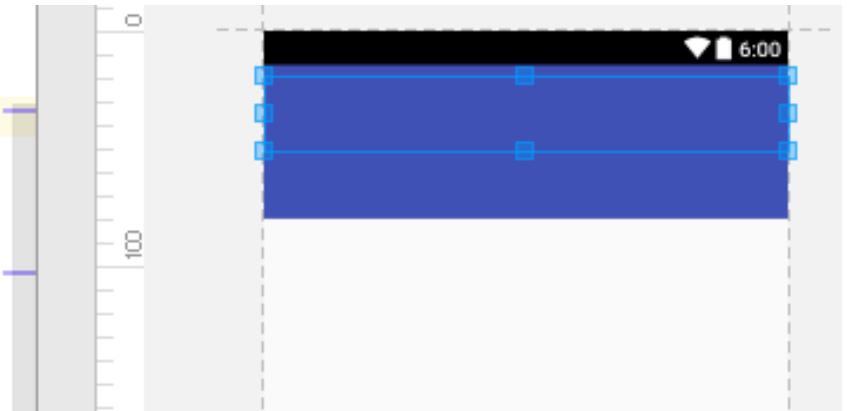


```
<dimen name="appbar_padding_top">8dp</dimen>
```



❖ Adicionar un Toolbar

```
-----  
android:theme="@style/AppTheme.AppBarOverlay">  
  
<android.support.v7.widget.Toolbar  
    android:id="@+id/toolbar"  
    android:layout_width="match_parent"  
    android:layout_height="?attr/actionBarSize"  
    android:background="?attr/colorPrimary"  
    app:layout_scrollFlags="scroll|enterAlways"  
    app:popupTheme="@style/AppTheme.PopupOverlay">
```



❖ y un TabLayout

```
</android.support.v7.widget.Toolbar>  
  
<android.support.design.widget.TabLayout  
    android:id="@+id/tabs"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content" />
```



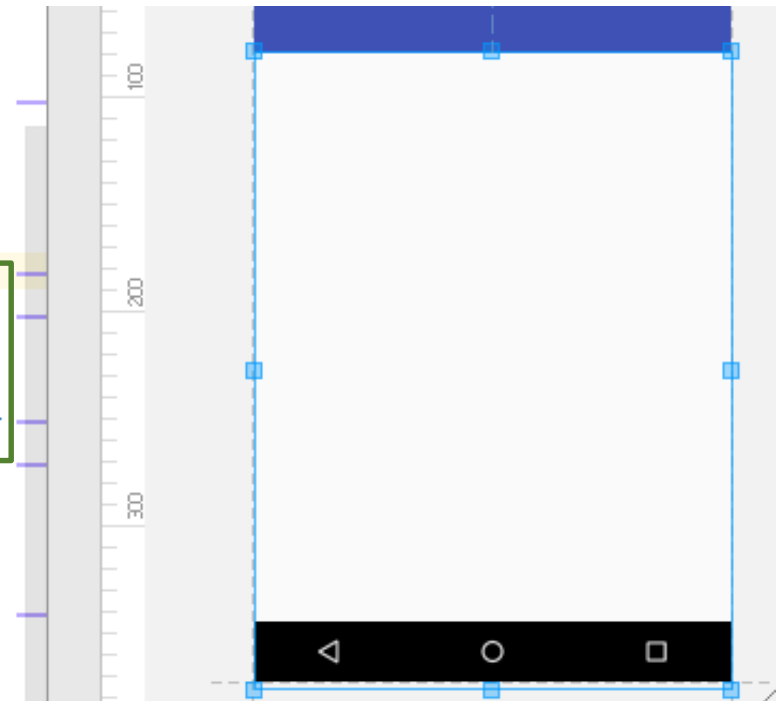
- ❖ Por último se coloca un ViewPager que permitirá visualizar los fragments

```
<android.support.design.widget.TabLayout  
    android:id="@+id/tabs"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content" />
```

```
</android.support.design.widget.AppBarLayout>
```

```
<android.support.v4.view.ViewPager  
    android:id="@+id/container"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    app:layout_behavior="@string/appbar_scrolling_view_behavior" />
```

```
android.support.design.widget.CoordinatorLayout>
```



- ❖ Implementamos un PageAdapter el cual hereda de FragmentPagerAdapter, agregamos el import, los métodos asociados y el constructor

? android.support.v4.app.FragmentPagerAdapter? Alt+Intro

```
public class PagerAdapter extends FragmentPagerAdapter {  
  
}
```

→ **OJO** Se debe cambiar los imports de los fragments

Select Methods to Implement



android.support.v4.app.FragmentPagerAdapter

getItem(i:int):Fragment

android.support.v4.view.PagerAdapter

getCount():int



```
public class PagerAdapter extends FragmentPagerAdapter {  
  
    public PagerAdapter(FragmentManager fm) {  
        super(fm);  
    }  
  
    @Override  
    public Fragment getItem(int i) {  
        return null;  
    }  
  
    @Override  
    public int getCount() {  
        return 0;  
    }  
}
```



```
public class PagerAdapter extends FragmentPagerAdapter {
```

Create constructor matching super

Create Test

Create subclass

```
(int i) {
```

❖ Configuramos los Fragments en la clase PagerAdapter

```
public class PagerAdapter extends FragmentPagerAdapter {  
  
    public PagerAdapter(FragmentManager fm) {  
        super(fm);  
    }  
  
    @Override  
    public Fragment getItem(int position) {  
        // getItem is called to instantiate the fragment for the given page.  
        // Return a PlaceholderFragment (defined as a static inner class below).  
        switch (position) {  
            case 0: BaresFragment tab1 = new BaresFragment();  
                return tab1;  
            case 1: HotelsFragment tab2 = new HotelsFragment();  
                return tab2;  
            case 2: RestaurantesFragment tab3 = new RestaurantesFragment();  
                return tab3;  
            default: return null;  
        }  
    }  
}
```



```
@Override
public int getCount() {
    // Show 3 total pages.
    return 3;
}

@Override
public CharSequence getPageTitle(int position) {
    switch (position) {
        case 0:
            return "Bares";
        case 1:
            return "Hoteles";
        case 2:
            return "Restaurantes";
    }
}
```


- ❖ En el MainActivity crear el ViewPager

```
private ViewPager mViewPager;
```

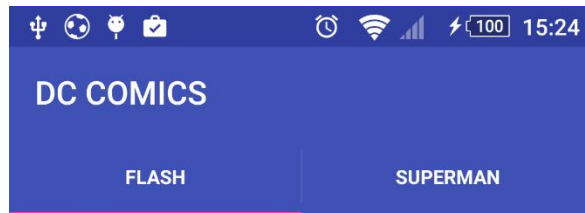
- ❖ Crear un objeto de tipo adapter, crear el ViewPager y adaptarlo.

```
PagerAdapter adapter = new PagerAdapter(getSupportFragmentManager());  
mViewPager = (ViewPager) findViewById(R.id.pager);  
mViewPager.setAdapter(adapter);
```

```
public class MainActivity extends AppCompatActivity {  
  
    /**...*/  
    private PagerAdapter mPagerAdapter;  
  
    /**...*/  
    private ViewPager mViewPager;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  
        setSupportActionBar(toolbar);  
  
        mPagerAdapter = new PagerAdapter(getSupportFragmentManager());  
  
        mViewPager = (ViewPager) findViewById(R.id.container);  
        mViewPager.setAdapter(mPagerAdapter);  
  
        TabLayout tabLayout = (TabLayout) findViewById(R.id.tabs);  
        tabLayout.setupWithViewPager(mViewPager);  
    }  
}
```

Probar la aplicación

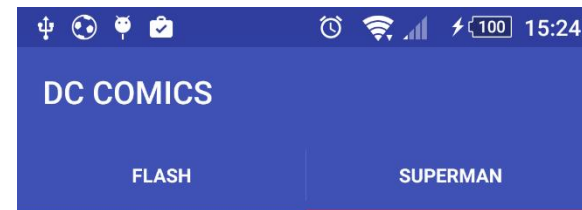
Tabs



FLASH



Super speed, intangibility,
superhuman agility



SUPERMAN



Super strength, flight, invulnerability,
super speed, heat vision, freeze
breath, x-ray vision, superhuman
hearing, healing factor



- ❖ Si se desea personalizar los tabs con imágenes y su texto, scrollable, etc
- ❖ <https://www.google.com/design/spec/components/tabs.html#tabs-specs>