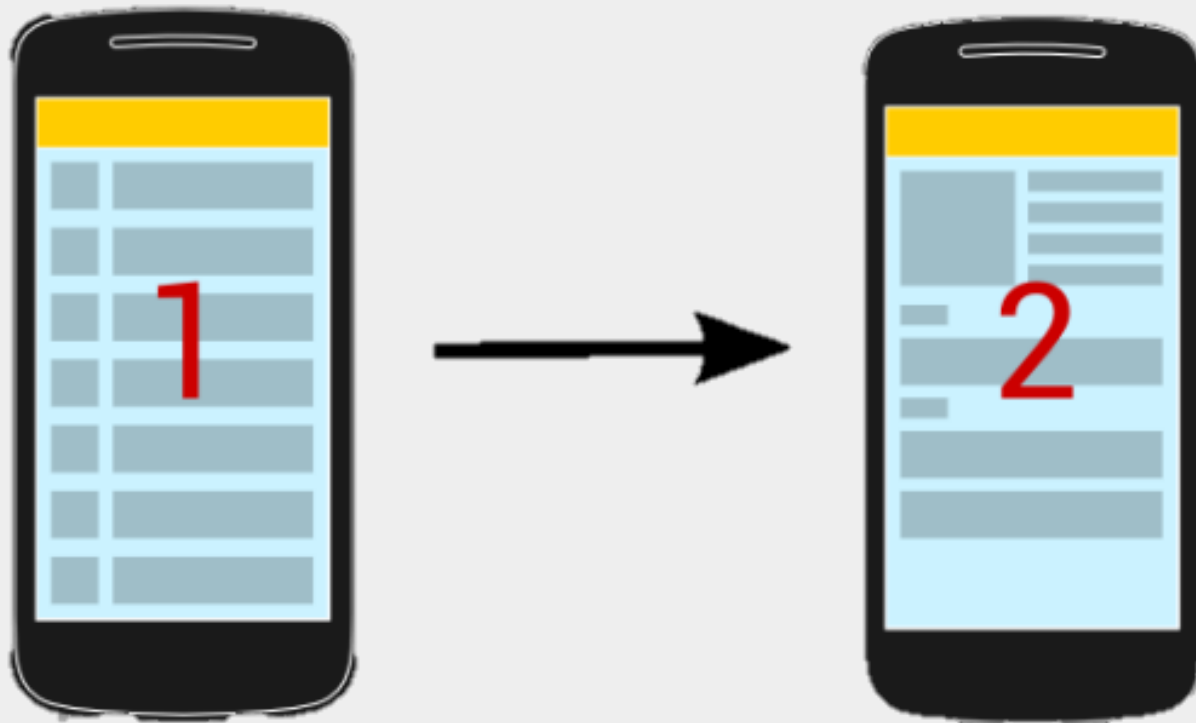




Fragments

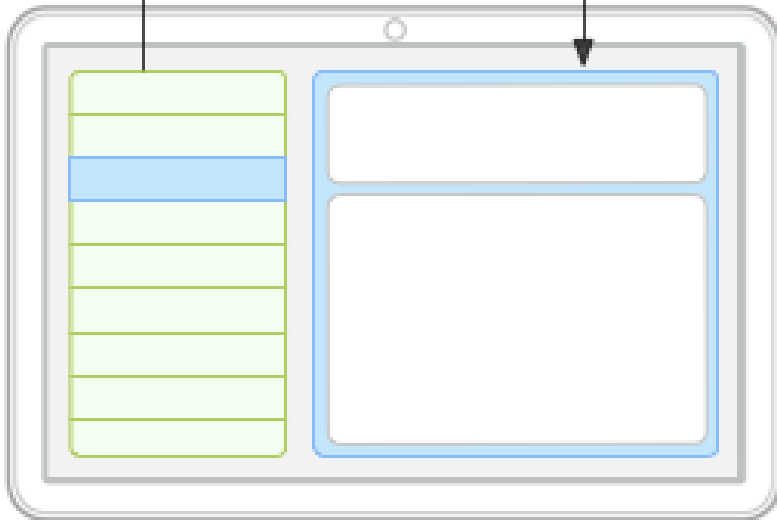
App con Barra



App Master detail

Tablet

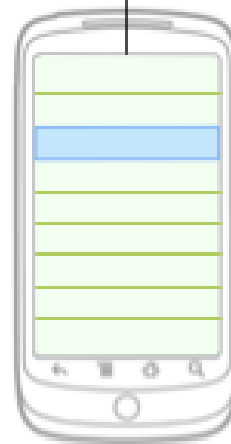
Selecting an item
updates Fragment B



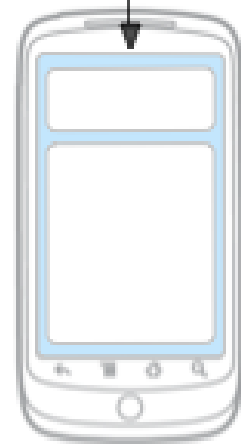
Activity A contains
Fragment A and Fragment B

Handset

Selecting an item
starts Activity B

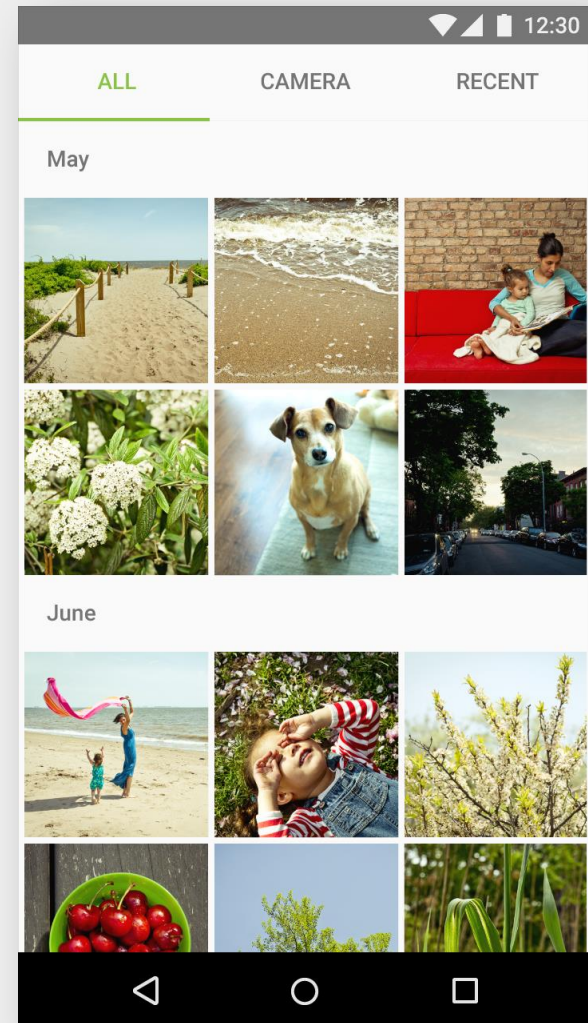
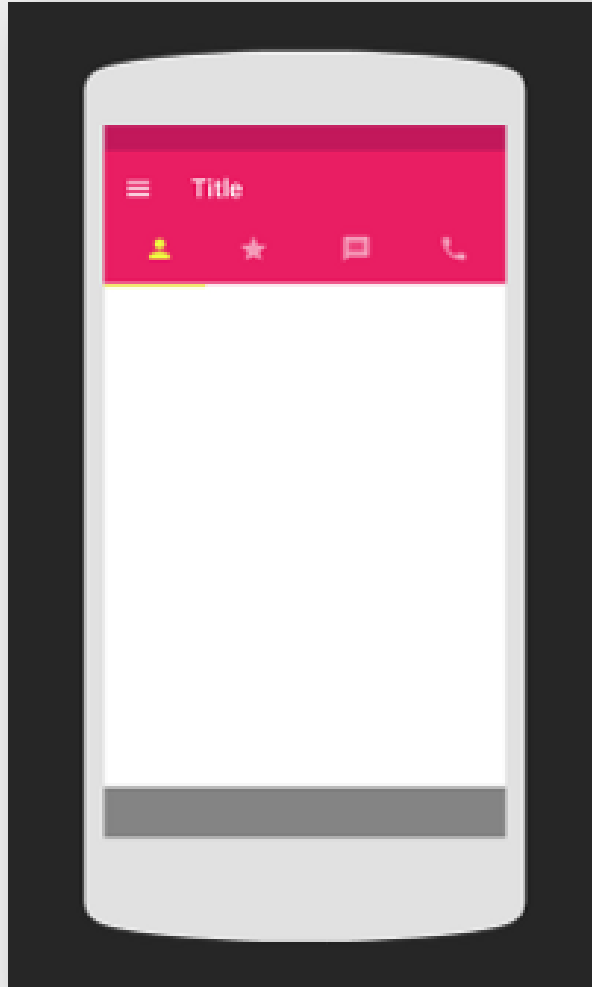


Activity A contains
Fragment A

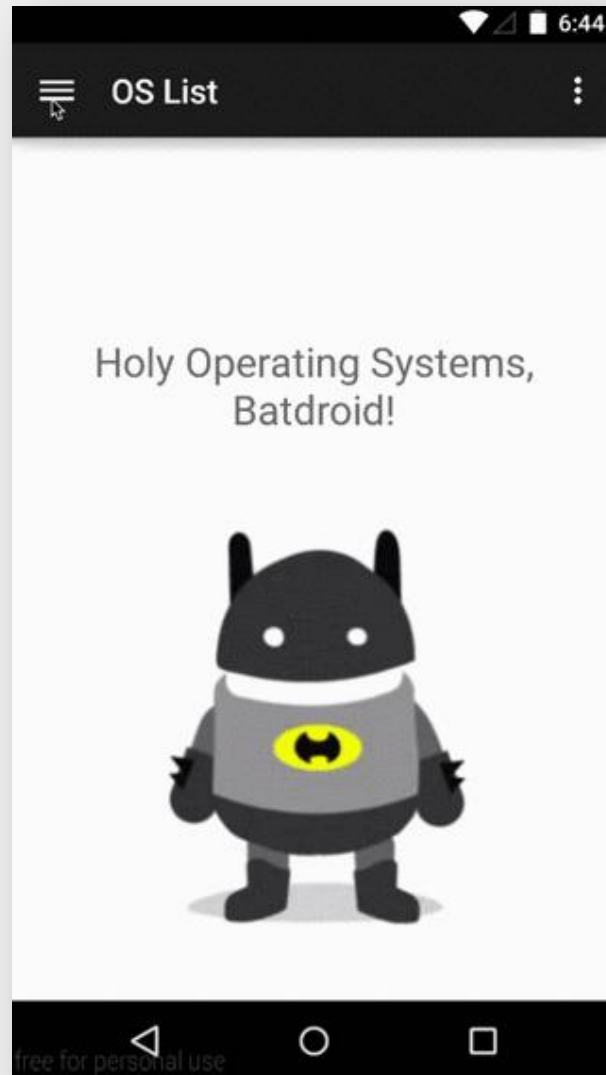


Activity B contains
Fragment B

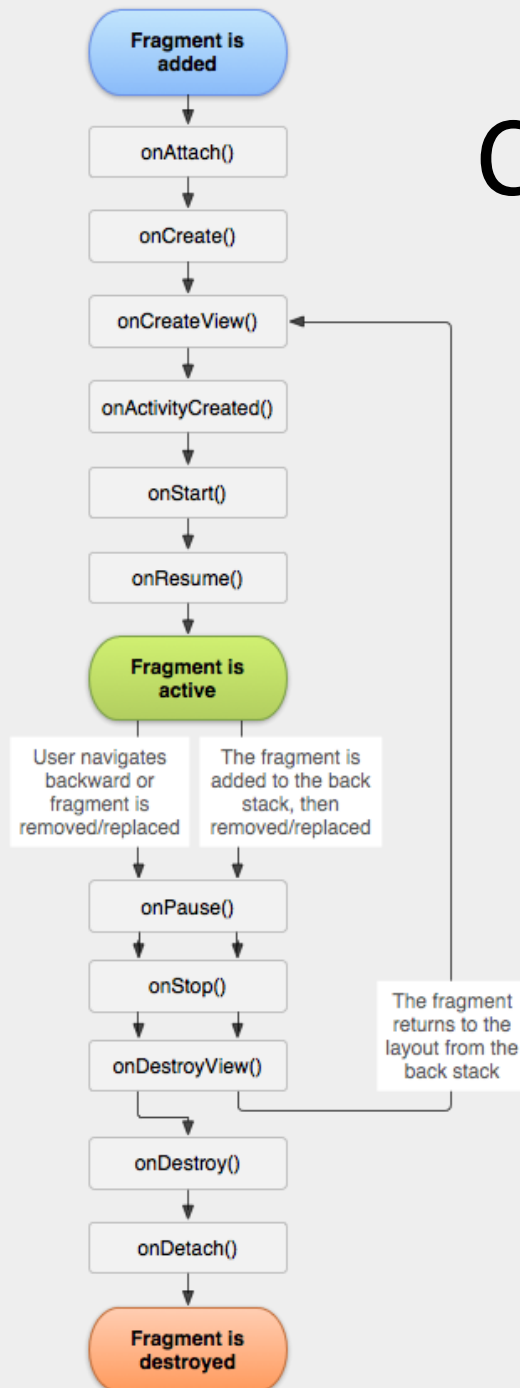
Tabs



Navigation drawer



Ciclo de vida



En **onAttach** los fragments reciben una referencia a la actividad padre.

En un los fragments los accesos a las vistas se hacen en **onCreateView**, no en **onCreate** como en las actividades.

En **onActivityCreated** los fragment saben que la actividad padre ha sido creada.

Crear un Fragment

1. Construir un **layout** para el Fragment:

R.layout.*mi_fragment*

2. Crear clase **Java** heredando de Fragment:

```
public class MiFragment extends Fragment {  
    @Nullable  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
                             Bundle savedInstanceState) {  
        return inflater.inflate(R.layout.mi_fragment, container, false);  
    }  
}
```

3. Incluir el Fragment en la Activity:

- a) Fragment **estático**: Añadir al **xml** de la Activity
- b) Fragment **dinámico**: Generar con código Java

Fragment estático

- En el **layout** de la actividad:

```
<fragment  
    android:id="@+id/fragment"  
    android:name="com.profe.appConFragments.MiFragment"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content" />
```


Fragment dinámico

- Añadir fragment en la actividad:

```
MiFragment fragment = new MiFragment();  
FragmentManager manager = getSupportFragmentManager();  
FragmentTransaction transaction = manager.beginTransaction();  
transaction.add(R.id. activityLayout, fragment, "MiFragmentTag");  
transaction.commit();
```

Quitar y reemplazar fragments

- Quitar:

```
FragmentManager manager = getSupportFragmentManager();  
Fragment fragment = manager.findFragmentByTag("MiFragmentTag");  
FragmentTransaction transaction = manager.beginTransaction();  
transaction.remove(fragment);  
transaction.commit();
```

- Reemplazar:

```
Fragment fragment = new MiOtroFragment();  
FragmentManager manager = getSupportFragmentManager();  
FragmentTransaction transaction = manager.beginTransaction();  
transaction.replace(R.id.activityMainLayout, fragment, "MiFragmentTag");  
transaction.commit();
```

Enviar datos de Activity a Fragment

- En **Fragment**:
 - Crear método público para recibir dato
- En **Activity**:
 - Obtener el fragment con Fragment Manager
 - Llamar al método del fragment con los datos

Enviar datos de Fragment a Activity

- En **Activity**:
 - Crear método público para recibir dato
- En **Fragment**:
 - Obtener actividad usando getActivity()
 - Casting la clase de nuestra actividad
 - Llamar al método de la actividad con los datos

Master detail

- **MasterFragment**: Lista de elementos
- **DetailFragment**: Detalle de elemento seleccionado
- **DetailActivity**: Contiene DetailFragment en móvil.
- **MainActivity**:
 - En **móvil**: Contiene MasterFragment y abre DetailActivity al seleccionar elemento.
 - En **tablet**: Contiene MasterFragment y DetailFragment y actualiza DetailFragment al seleccionar elemento.