

Día 1 (Introducción a Activity y Fragment)

1. Agregar dependencias al proyecto

- a. En el archivo **build.gradle** del módulo app se debe agregar la dependencia hacia la biblioteca de soporte:
 - i. `compile 'com.android.support:appcompat-v7:24.2.1'`

2. Crear LandingActivity

- a. Crear la clase **LandingActivity** que hereda de [android.support.v7.app.AppCompatActivity].
- b. **LandingActivity** debe estar declarada dentro del **AndroidManifest.xml** y contener el intent-filter para que sea esta Activity la que se muestra al inicio de la aplicación.
- c. Crear un **layout (xml)** que se muestre en el onCreate() de LandingActivity. Este layout contiene un view contenedor (FrameLayout) que se utiliza para agregar/remover Fragments. Importante agregar atributo **id** al view.

3. Crear LoginFragment

- a. Crear la clase **LoginFragment** que hereda de [android.support.v4.app.Fragment].
- b. Crear un **layout (xml)** que se muestre en el onCreateView() de LoginFragment. Este layout contiene los EditText necesarios para obtener la información del usuario. Esto se logra mediante el método findViewById(int).

4. Crear RegisterFragment

- a. Crear la clase **RegisterFragment** que hereda de [android.support.v4.app.Fragment].
- b. Crear un **layout (xml)** que se muestre en el onCreateView() de RegisterFragment. Este layout contiene los EditText necesarios para obtener la información del usuario. Esto se logra mediante el método findViewById(int).

5. Mostrar LoginFragment desde LandingActivity

- a. El fragment se debe mostrar **después** de haber establecido el view de la Activity utilizando setContentView(int) dentro del método onCreate().

6. Reemplazar LoginFragment con RegisterFragment

- a. Para esto se utiliza el método *FragmentManager#replace()* en vez de *FragmentManager#add()*.