

Día 4 (Diálogos, Lectura de QR y Volley)

1. Agregar nueva UI y menú a CreateTaskFragment

a. Agregar nueva UI

- i. Modificar el **layout (xml)** del fragment **CreateTaskFragment** para que la jerarquía quede:

1. **LinearLayout (Vertical)**

- a. **LinearLayout (Horizontal)**

- i. **EditText**

- ii. **Button**

- b. **LinearLayout (Horizontal)**

- i. **EditText**

- ii. **Button**

- c. **LinearLayout (Horizontal)**

- i. **ImageView**

- ii. **Button**

b. Agregar menú

- i. Crear un nuevo **xml** dentro de **res/menu** llamado *fragment_create_task.xml* y agregar un ítem para guardar el task.
- ii. Dentro del *onCreateView()* de **CreateTaskFragment** se debe llamar el método *Fragment#setHasOptionsMenu(true)*. Esto provoca que el Fragment llame el método *Fragment#onCreateOptionsMenu()* durante su ciclo de vida.
- iii. En **CreateTaskFragment** se debe sobrescribir el método *Fragment#onCreateOptionsMenu()* para agregar el nuevo menú al action bar.
- iv. En **CreateTaskFragment** se debe sobrescribir el método *Fragment#onOptionsItemSelected()* para agregar reaccionar al evento de click de los nuevos ítems del menú.

2. Diálogos

a. Crear TitleSelectDialogFragment

- i. Crear la clase **TitleSelectDialogFragment** que hereda de [android.support.v4.app.DialogFragment].

- ii. Crear un **layout (xml)** que se muestre en el *onCreateView()* de **TitleSelectDialogFragment**. Este layout se va a ver así:

1. LinearLayout
 - a. TextView
 - b. RadioGroup
 - i. RadioButton
 - ii. RadioButton
 - iii. RadioButton
 - c. Button

b. Mostrar TitleSelectDialogFragment

- i. Se debe crear una instancia de **TitleSelectDialogFragment** y se muestra mediante el método *DialogFragment#show()*.
- ii. Agregar un listener para notificar cuando se selecciona un título desde **TitleSelectDialogFragment**. El fragment que muestre el diálogo debe implementar este listener.
- iii. El fragment que muestra el diálogo (en este caso **CreateTaskFragment**) se pasa al diálogo usando el método *Fragment#setTargetFragment()* el cual se accede desde el diálogo mediante el método *Fragment#getTargetFragment()*.

3. Lectura de código QR

a. Agregar dependencias al proyecto

- i. En el archivo build.gradle del módulo app se debe agregar la dependencia hacia la biblioteca de soporte de diseño:

1. `compile 'me.dm7.barcodescanner:zxing:1.9'`

b. Crear QRScannerActivity

- i. Crear la clase **QRScannerActivity** que hereda de [android.support.v7.app.AppCompatActivity].
- ii. **QRScannerActivity** debe estar declarada dentro del **AndroidManifest.xml**.
- iii. En este caso, esta Activity no necesita un **layout (xml)**. El view se va a definir completamente vía código.
- iv. Seguir los pasos para integración de la biblioteca mostrados en <https://github.com/dm77/barcodescanner>

- v. Devolver el resultado luego de iniciar la activity utilizando *startActivityForResult()*.

4. Introducción a Volley

a. Agregar dependencias al proyecto

- i. En el archivo build.gradle del módulo app se debe agregar la dependencia hacia la biblioteca de soporte de diseño:

1. `compile 'com.android.volley:volley:1.0.0'`

b. Agregar permisos para Internet

- i. Solo se requiere agregar el permiso en el **AndroidManifest.xml**.

c. Crear VolleyRequestSingleton

- i. Crear una clase **VolleyRequestSingleton**. Esta clase debe ser singleton. No debe heredar de nada en específico.
- ii. Agregar el Queue de volley.
- iii. Agregar método genérico para agregar un request al Queue.

d. Crear modelo de Usuario

- i. Crear clase **User**. No debe heredar de nada en específico. Agregar campos para:
 - 1. AuthToken : String
 - 2. Email : String
 - 3. FirstName : String
 - 4. LastName : String
 - 5. Password : String
 - 6. PasswordConfirmation : String

e. Crear BackendAccess

- i. Crear clase **BackendAccess**. No debe heredar de nada en específico. Esta es la clase que se va a encargar de agregar requests a volley para login/register/logout etc.
- ii. Esta clase se encarga de generar los URL y las representaciones en JSON de los modelos (User, Task, etc...)
- iii. Agregar un **interface** para notificar cuando se finaliza un request.

f. Agregar llamadas al Backend

- i. Dentro de los fragment **LoginFragment** y **RegisterFragment** se debe agregar una instancia de **BackendAccess**.
- ii. Cuando se intenta hacer login o crear un usuario se hacen las llamadas correspondientes de **BackendAccess** y se inician las nuevas ventanas o se muestra el error dentro del **callback**.