

Presentazione TrackingMyPantry

Filippo Perrina

Giugno 2021

Grafo di navigazione

L'applicazione è composta da 8 **activities** e 3 **fragments**. Il seguente grafo di navigazione mostra le possibili direzioni che l'utente può seguire durante l'uso dell'applicazione.

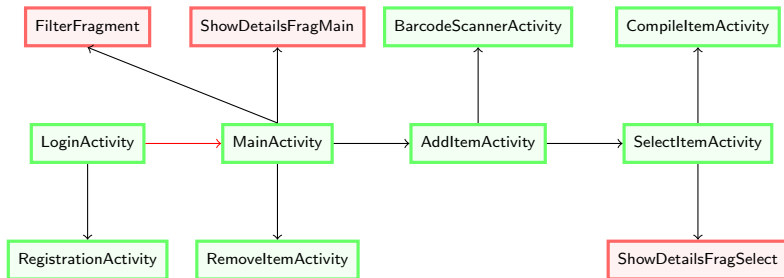


Figure: Grafo di navigazione dell'applicazione

Struttura

L'applicazione è sviluppata seguendo il modello ModelViewViewModel (MVVM), con un ulteriore livello di astrazione fornito dalla repository.

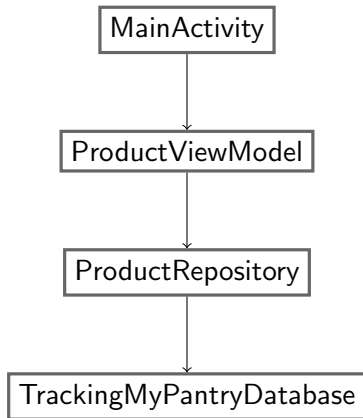


Figure: Modello ModelViewViewModel

La dispensa

La `recyclerViewListProducts` è una `recyclerView` che mostra gli elementi che sono presenti nella dispensa locale. E' contenuta nella `MainActivity` ed è l'unico elemento nell'intera applicazione che viene controllato dal `ViewModel`.

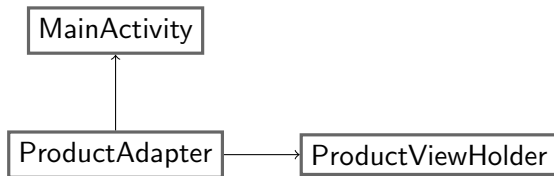


Figure: Struttura della `recyclerViewListProducts`

Database locale

Tre classi si occupano della gestione del database interno:

- ▶ `TrackingMyPantryDatabase`: istanzia il database, implementato usando Room.
- ▶ `ProductDAO`: interfaccia contenente sette operazioni CRUD sul database.
- ▶ `ProductEntity`: unica tabella del database. Tutti i dati sono memorizzati come `String`, tranne `quantity`.

PRODUCT	id	barcode	description	category	endDate	quantity
---------	----	---------	-------------	----------	---------	----------

Table: Schema del database

Componenti applicazione

Ognuno dei componenti con un lifecycle ha associato un layout, l'associazione è: `NomeComponenteTipoComponente.java`
→ `tipoComponente_nomeComponente.xml` (e.g. `MainActivity.java`
→ `activity_main.xml`).

- ▶ `LoginActivity`: schermata iniziale, gestisce il login.
- ▶ `RegistrationActivity`: realizza la registrazione di un nuovo utente al server.

Componenti applicazione (MainActivity)

La MainActivity è la schermata principale dell'applicazione.

- ▶ mostra la dispensa.
- ▶ realizza il filtraggio della lista sia per stringa che per caratteristiche del prodotto. (filterFragement)
- ▶ permette all'utente di vedere e modificare i dettagli dei prodotti nella dispensa interna. (showDetailsFragmentMain)

Componenti applicazione

- ▶ `RemoveItemActivity`: passa il prodotto da rimuovere dalla dispensa locale alla `MainActivity`, no dispensa globale.
- ▶ `AddItemActivity`: cerca i prodotti corrispondenti al codice a barre inserito sul server remoto, passa il risultato a `SelectItemActivity`.
- ▶ `BarcodeScannerActivity`: scannerizza attraverso la fotocamera il codice a barre di un prodotto.

Componenti applicazione

- ▶ `SelectItemActivity`: mostra i risultati e i dettagli (`showDetailsFragmentSelect`) associati al codice a barre ricercati e passa il prodotto selezionato ad `addItemActivity`. I risultati sono mostrati in una `recyclerView` gestita da `SelectAdapter`.
- ▶ `CompileItemActivity`: permette all'utente di inserire un nuovo prodotto sul server remoto, passa il prodotto inserito a `SelectItemActivity`.