

REPORT PROGETTO LABORATORIO DI APPLICAZIONI MOBILI 2020/2021

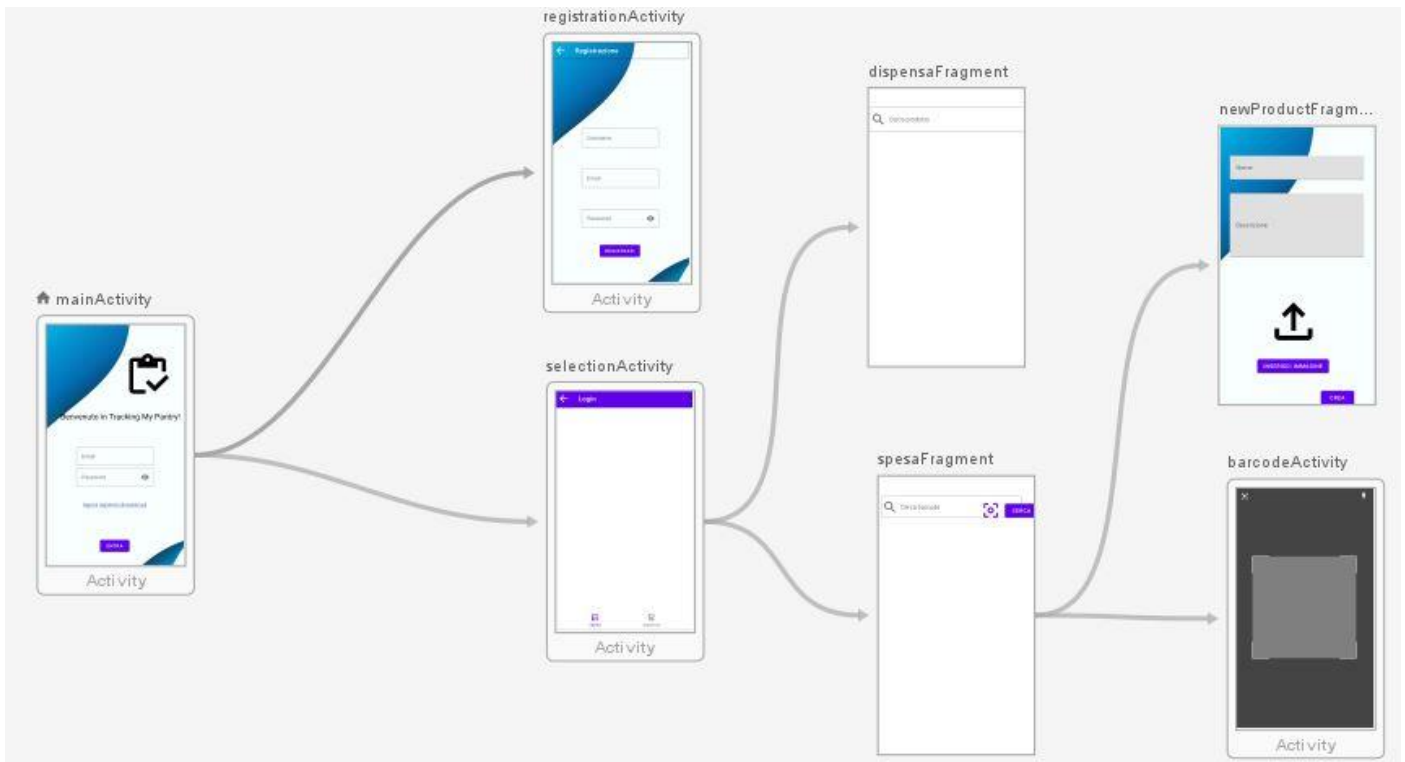
TRACKING MY PANTRY

Bianca Raimondi

873520

[bianca.raimondi2@studio.unibo.it](mailto:bianca.raimondi2@studio.unibo.it)

# NAVIGAZIONE



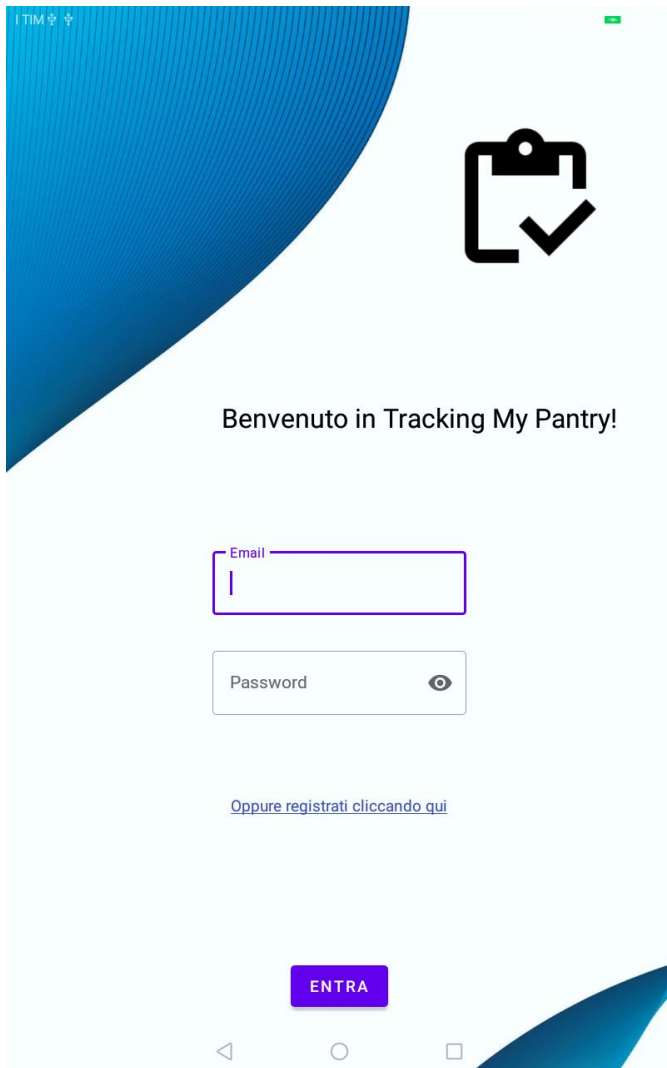
L'applicazione è composta da 4 activity e 3 fragment.

La MainActivity è l'activity principale, dove l'utente può fare login oppure essere reindirizzato all'activity per la registrazione (RegistrationActivity).

Quando un utente ha effettuato il login correttamente si passa alla visualizzazione della SelectionActivity, che, in base alla scelta dell'utente potrà mostrare

- SpesaFragment, dove i prodotti vengono cercati utilizzando il campo di testo per inserire il barcode oppure utilizzando BarcodeActivity che scannerizza il barcode in modo automatico. Sempre da SpesaFragment è possibile creare un nuovo prodotto aprendo NewProductFragment
- DispensaFragment, dove vengono visualizzati i prodotti attualmente nel database dell'utente ed è possibile modificarne le quantità oppure eliminarli

## MAIN ACTIVITY

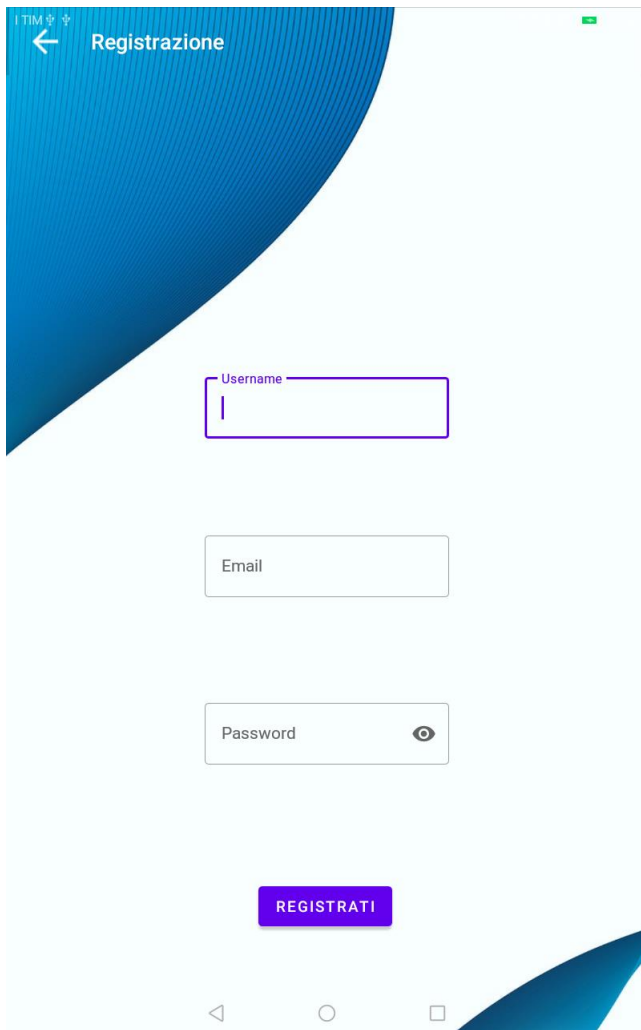


MainActivity si occupa principalmente del login dell'utente richiedendo di inserire le proprie credenziali (Email e Password).

Nel caso in cui l'utente non sia ancora registrato è possibile registrarsi cliccando il link relativo che lo reindirizzerà a RegistrationActivity.

Nel caso in cui l'utente sia già registrato, solo dopo aver inserito le credenziali verrà reindirizzato a SelectionActivity ed iniziare ad utilizzare le funzionalità dell'applicazione.

## REGISTRATION ACTIVITY



The screenshot shows the 'Registrazione' (Registration) screen of an Android application. At the top left, there is a back arrow icon and the title 'Registrazione'. The screen features three input fields: 'Username' (with a purple border), 'Email', and 'Password' (with a toggle eye icon). Below these fields is a purple button labeled 'REGISTRATI'. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.

RegistrationActivity si occupa principalmente della registrazione di un nuovo utente richiedendo di inserire nuove credenziali: Username (il quale non verrà più richiesto dopo la registrazione), Email e Password.

Nel caso in cui l'utente voglia tornare alla schermata di login è possibile tramite l'apposita icona nella AppBar.

## SELECTION ACTIVITY



SelectionActivity si occupa principalmente di gestire la visualizzazione richiesta dall'utente:

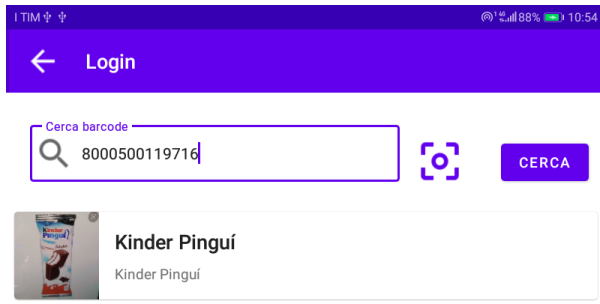
Nel caso in cui un altro utente voglia fare login è possibile tornare alla MainActivity attraverso l'apposita icona nella AppBar.

SelectionActivity contiene un layout per far visualizzare 3 fragment:

- SpesaFragment che si occupa della ricerca di prodotti da inserire nel database (remoto e locale)
- DispensaFragment che si occupa di visualizzare e gestire il database locale
- NewProductFragment che si occupa di inserire un nuovo prodotto all'interno del database globale

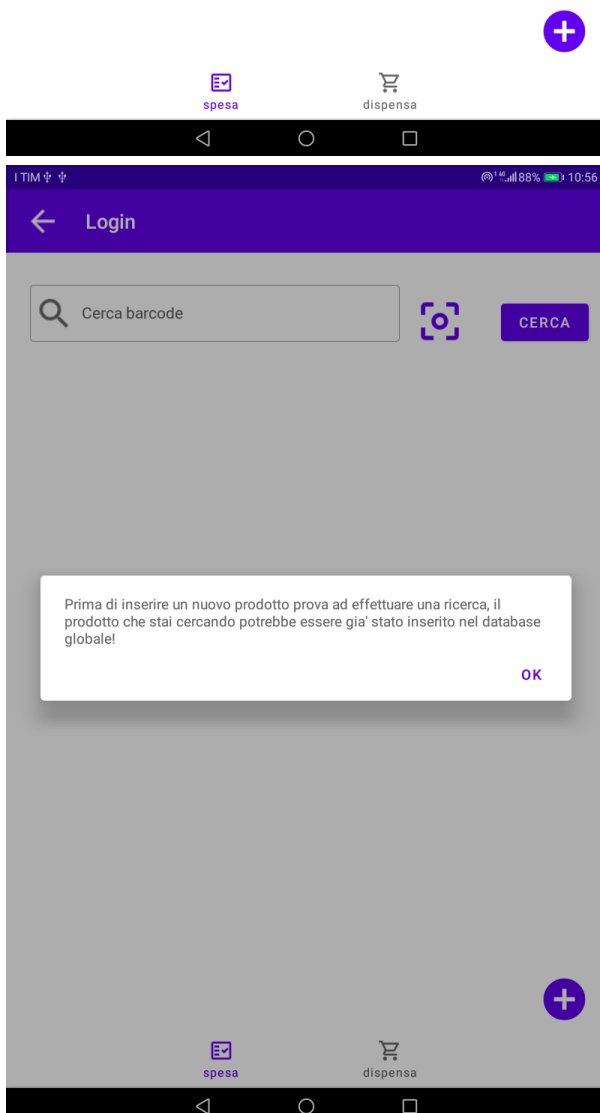


## SPESA FRAGMENT



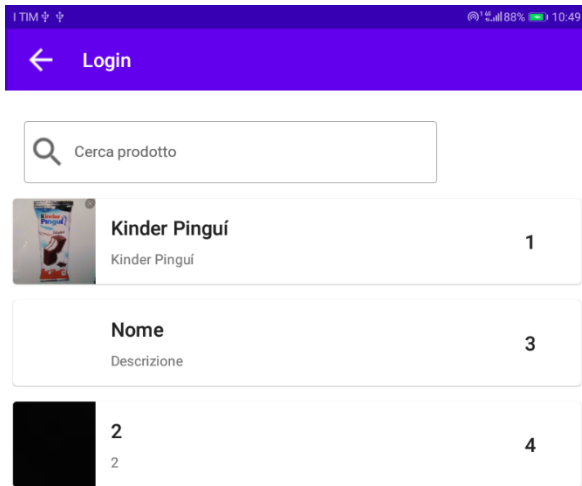
SpesaFragment è il fragment più complesso. Si occupa principalmente di effettuare una nuova ricerca tramite il relativo campo di testo oppure tramite il Button relativo che reindirizza l'utente a BarcodeActivity.

Solo dopo aver inserito un barcode nel campo di testo (che nel caso di BarcodeActivity avviene in modo automatico) è possibile effettuare una ricerca, la quale mostrerà nel ListView sottostante tutti i prodotti trovati attraverso delle CardView.



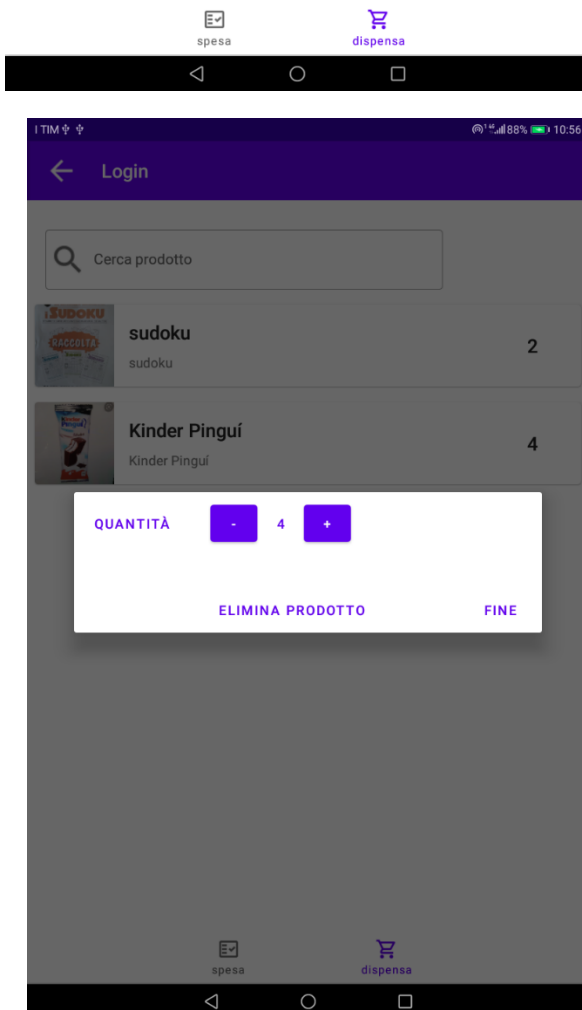
Solo dopo aver effettuato una ricerca sarà possibile inserire un nuovo prodotto utilizzando l'apposita icona; nel caso in cui non è ancora stata effettuata una ricerca verrà visualizzato un AlertDialog con un messaggio di informazione

## DISPENSA FRAGMENT



DispensaFragment si occupa della visualizzazione e della gestione del database locale dell'utente.

Qui è possibile effettuare una ricerca dei prodotti per nome attraverso l'apposito campo di testo. I prodotti verranno poi visualizzati attraverso una CardView che permette di visualizzare anche la quantità del relativo prodotto.



Se l'utente seleziona una CardView è possibile per lui modificarne la quantità, aggiungendo o togliendo unità in base all'utilizzo che l'utente fa del prodotto.

È anche possibile eliminare del tutto il prodotto dal database.

## NEW PRODUCT FRAGMENT

Nome  
Kinder Pinguí

Descrizione  
descrizione Kinder Pinguí

INSERISCI IMMAGINE

CREA

spesa dispensa

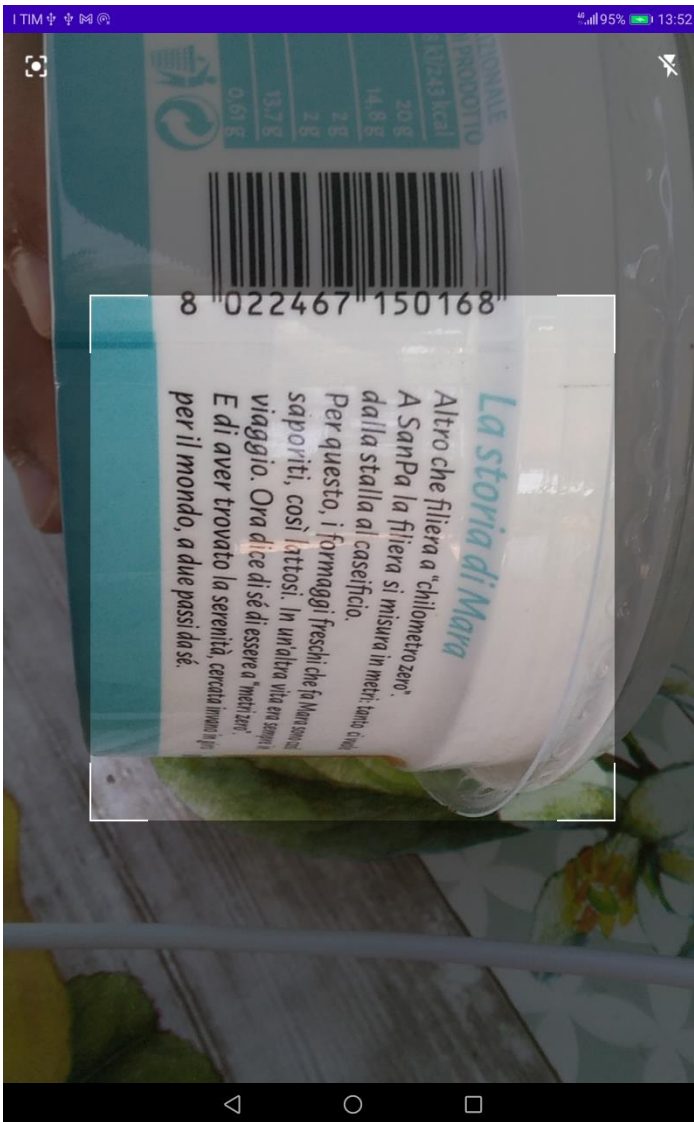
NewProductFragment viene visualizzato nel momento in cui un utente, dopo aver effettuato una ricerca decide di creare un nuovo prodotto.

Qui è possibile inserire il nome, la descrizione e, in modo facoltativo anche un'immagine rappresentativa del prodotto. Il barcode viene inserito in automatico dopo la ricerca.

Una volta creato il prodotto, oltre ad essere aggiunto al database globale, ne viene salvata una copia anche nel database locale dell'utente.



## BARCODE ACTIVITY



BarcodeActivity utilizza la libreria CodeScanner per rilevare in modo automatico i barcode tramite l'utilizzo della fotocamera.

Nel caso in cui rileva il barcode di un prodotto l'utente viene reindirizzato a SpesaFragment e il campo del codice a barre viene riempito in automatico. Solo successivamente può essere effettuata una nuova ricerca.

## MODEL VIEW VIEWMODEL

L'applicazione è basata sul modello ModelViewViewModel il quale viene utilizzato principalmente per differenziare logicamente le classi all'interno del progetto.

Per quanto riguarda la View, per ogni activity/fragment sono presenti due diverse visualizzazioni in base all'orientamento del dispositivo e la maggior parte dei dati statici sono stati salvati utilizzando la cartella resources.

Il ViewModel viene utilizzato solamente per SelectionActivity per salvare i dati nel caso in cui l'activity o i fragment vengano ricreati.

Per quanto riguarda il Model viene utilizzato un Repository (ProductRepository) per gestire il database locale.

Il database locale viene gestito utilizzando Room, mentre per quanto riguarda le chiamate al database remoto viene utilizzato Volley.

## DATABASE

Il database locale è composto da un'unica entità Product che gestisce i prodotti nella dispensa dell'utente.

L'entità Product è composta da 7 attributi:

```
@ColumnInfo
@NonNull
public String productId;

@ColumnInfo
@NonNull
public String userId; //email

@ColumnInfo
public String barcode;

@ColumnInfo
public String name;

@ColumnInfo
public String description;

@ColumnInfo
public Integer quantity;

@ColumnInfo
public String image;
```

productId e userId sono la PrimaryKey dell'entità per fare in modo che i prodotti vengano differenziati, oltre che dall'id del prodotto (uguale al productId del database remoto), anche dall'utente che ha inserito il prodotto in dispensa.

A differenza del database remoto è stato aggiunto un campo quantity che indica il numero di unità dello stesso prodotto presenti nel database dell'utente.

## DAO

Per gestire le modifiche al database viene utilizzata una Interface ProductDao composta dai seguenti metodi:

- getProduct(String userId, String productId) che ritorna il prodotto richiesto
- updateProduct(String userId, String productId, **int** productQuantity) che modifica la quantità del prodotto aggiungendo un numero di unità pari a quelle passate per parametro
- modifyQuantity(String userId, String productId) che modifica la quantità del prodotto togliendo un numero di unità pari a quelle passate per parametro
- insertProduct(Product product) che inserisce un nuovo prodotto nel database
- deleteProduct(String userId, String productId) che elimina un prodotto dal database  
deleteAll() che elimina il database
- getListOfProducts(String userId) che ritorna l'intera lista dei prodotti nella dispensa dell'utente passato come parametro