

Projet Application Mobile - Android Studio

Lola-Marie Moron (204), Raphaël Gonçalves Catarino (209), William Zhang (204)

Structure globale du projet :

- SplashScreenActivity / activity_splash_screen
- LoginActivity / activity_login
- RegisterActivity / activity_register
- MainActivity / activity_main
 - HomeFragment / fragment_home
 - AllProductsFragment
 - AllProductsRecyclerViewAdapter
 - AddFragment / fragment_add
 - AccountFragment / fragment_account
 - AllProductsFragment
 - AllProductsOwnedRecyclerViewAdapter
 - SettingsFragment / fragment_settings

Base de données :

Nous avons opté pour Firebase Firestore pour la base de données. En effet, son intégration est très complète dans Android Studio et très simple d'utilisation à l'aide de l'ensemble des méthodes fournies.

Explication de chaque activité/fragment :

SplashScreenActivity



S'affiche au lancement de l'application pour accueillir l'utilisateur et laisser le temps à l'application de se charger.

LoginActivity



Permet à l'utilisateur de se connecter en entrant son mail ainsi que son mot de passe. La connexion se fait via Firebase Authentification.

Une fois connecté, on enregistre l'identifiant de l'utilisateur dans les sharedPreferences de l'application (pour des utilisations futures relatives à la BD) puis on redirige l'utilisateur vers la MainActivity.

S'il n'a pas de compte, il peut cliquer en bas pour être redirigé vers RegisterActivity.

RegisterActivity



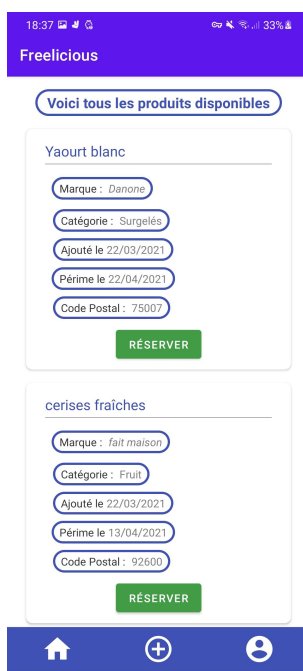
Cette activité permet à un utilisateur de se créer un compte en renseignant son identifiant, son adresse mail et son mot de passe. Une fois la validation faite, on ajoute l'utilisateur via Firebase Authentification et une table lui correspondant dans la BD, puis on le redirige vers MainActivity.

MainActivity

Cette activité se compose de 2 parties : une première en bas de page qui affiche la navbar, et juste au-dessus, sur le reste de l'écran, un layout dans lequel sera intégré l'un des 4 fragments principaux (HomeFragment, AddFragment, AccountFragment, SettingsFragment). Cela permet d'avoir un code épuré et homogène, tout en permettant un chargement rapide de chaque page.

Ainsi selon l'icône sur laquelle l'utilisateur clique, le fragment correspond se charge (sauf SettingsFragment qui est accessible depuis AccountFragment). Par défaut, on affiche HomeFragment.

HomeFragment

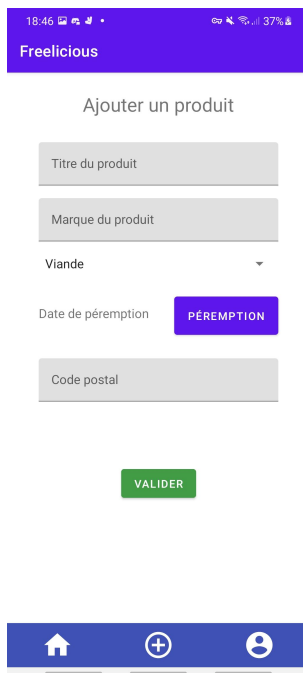


Cette page affiche l'ensemble des produits qui n'appartiennent pas à l'utilisateur et qui ne sont pas périmés.

Pour cela on utilise un RecyclerView que l'on détaillera par la suite.

En cliquant sur le bouton "réserver", la réservation est confirmée (ajout d'un champ "reservePar" dans Firestore), et une snackbar l'indique à l'utilisateur. Si on le souhaite, on peut cliquer sur "annuler" sur la snackbar pour annuler la réservation et rendre de nouveau le produit disponible à tous.

AddFragment



18:46 37%

Freelicious

Ajouter un produit

Titre du produit

Marque du produit

Vlande

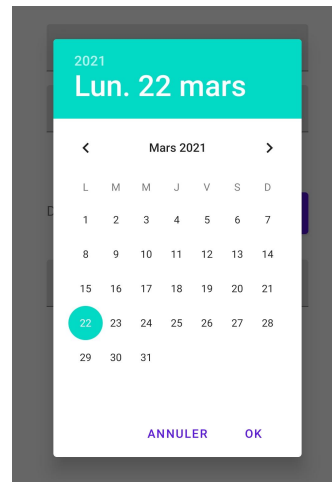
Date de péremption **PÉREMPTION**

Code postal

VALIDER

Home Add Profile

Permet d'ajouter un produit en indiquant un titre, une marque, une catégorie, une date de péremption et un code postal. La catégorie peut être choisie via un Spinner, et la date via un DatePickerDialog ainsi :



Une fois validé, le produit est ajouté à Firestore et visible par l'ensemble des autres utilisateurs.

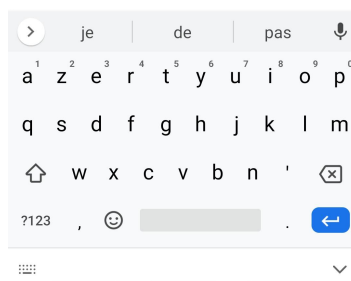
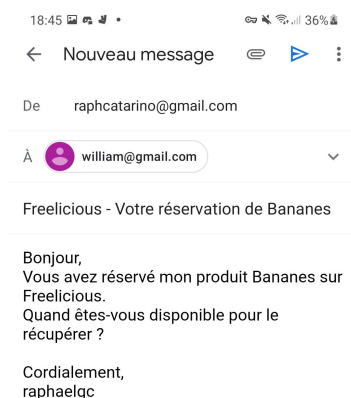
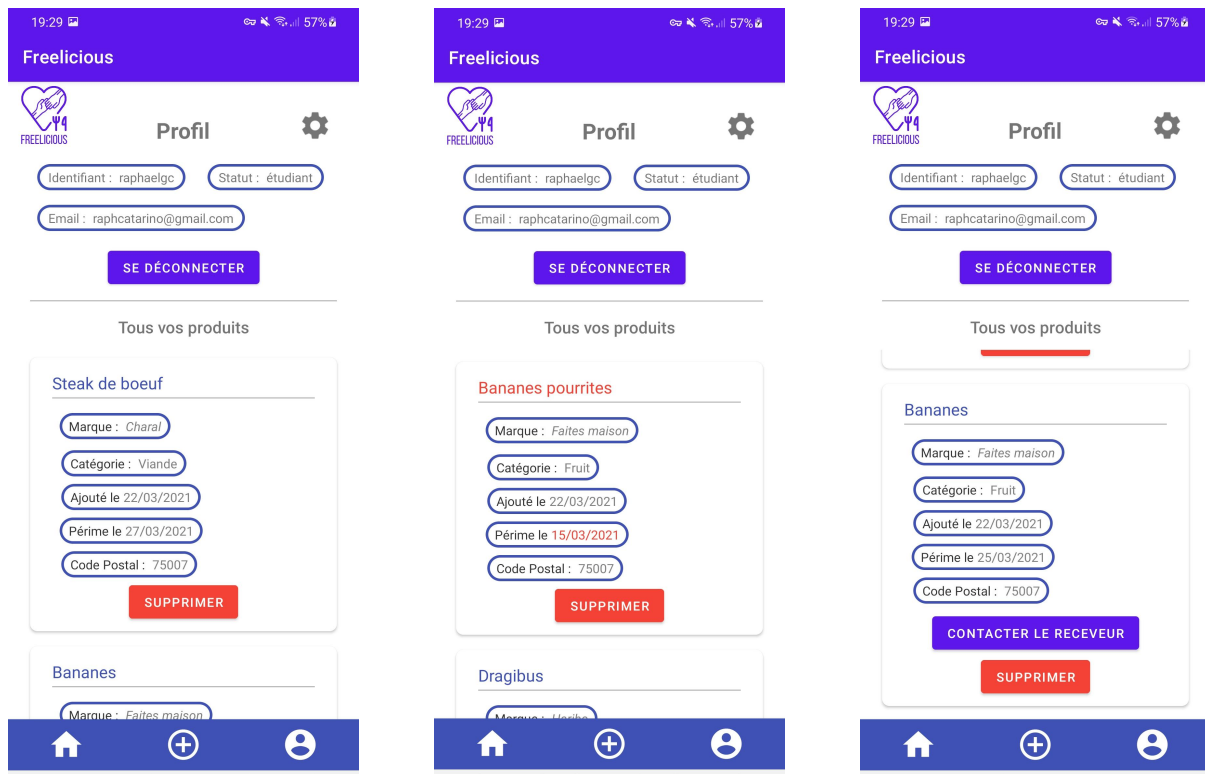
AccountFragment

Ce fragment représente la page profil de l'utilisateur actuel. La partie supérieure permet d'afficher l'identifiant, le statut et l'email de l'utilisateur ainsi que de se déconnecter (redirigé vers LoginActivity).

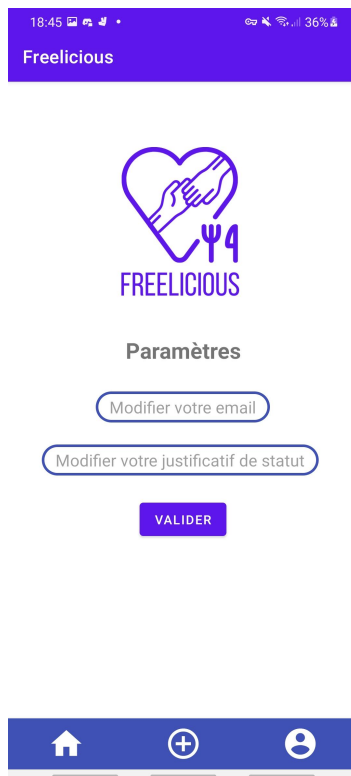
La partie inférieure affiche la liste de tous les produits publiés par l'utilisateur actuel (via un RecyclerView). On peut le supprimer en cliquant sur le bouton correspond, une snackbar apparait pour confirmer la suppression et l'annuler si on le souhaite.

Si le produit est périmé (2), il s'affiche en rouge, tout comme sa date de péremption.

Si le produit est réservé (3) par un autre utilisateur, un bouton apparait alors nous permettant de contacter le receveur par mail (redirection automatique vers une application externe de messagerie mail avec email du destinataire/sujet/objet).



SettingsFragment :



Ce fragment permet à l'utilisateur de changer son adresse email ainsi que d'ajouter un lien vers un justificatif de statut (permettant aux administrateurs de vérifier ce dernier).

Lors de la validation, on ajoute les informations dans le document de Firestore correspondant et via Firebase Authentication (pour l'email).

Utilisation de RecyclerView

Pour l'affichage de la liste des produits, nous avons décidé d'utiliser un Fragment RecyclerView. Ce dernier prend en paramètre un booléen permettant de savoir sur quelle page il se situe et donc effectué la requête Firestore adéquat ainsi que charger l'Adapter correspondant.

- Si ce booléen est à TRUE (page Account) : Le fragment récupère tous les produits de l'utilisateur actuel puis lance l'Adapter AllProductsOwnedRecyclerViewAdapter, qui va donc afficher le bouton supprimer, et si le produit est réservé, le bouton "Contacter le receveur".
- Sinon (page Home) : Le fragment récupère tous les produits des autres utilisateurs (non périmés) puis lance AllProductsRecyclerViewAdapter, qui va alors afficher uniquement le bouton "réserver".