

8a, avenue V. Maistriau B-7000 Mons

Tél : +32 (0)65 33 81 54

Fax : +32 (0)65 31 30 51

E-mail : tech-mons@heh.be

[www.heh.be](http://www.heh.be)

## Projet développement mobile

## Table des matières

1	Introduction.....	3
2	Présentation de l'application .....	3
3	Développement de l'application .....	4
4	Problème .....	5
4.1	L'affichage du matériel.....	5
4.2	Barre de navigation actualisé .....	5
4.3	Orientation de l'écran .....	6
5	Amélioration.....	6
5.1	Partage du QRCode .....	6
5.2	Historique du stockage.....	6
6	Conclusion .....	6
7	Table des figures.....	7
8	Bibliographie.....	7

## 1 Introduction

Dans le cadre du cours de « Développement mobile », il a été demandé de réaliser une application mobile avec Android Studio, pour la gestion d'un stockage.

## 2 Présentation de l'application

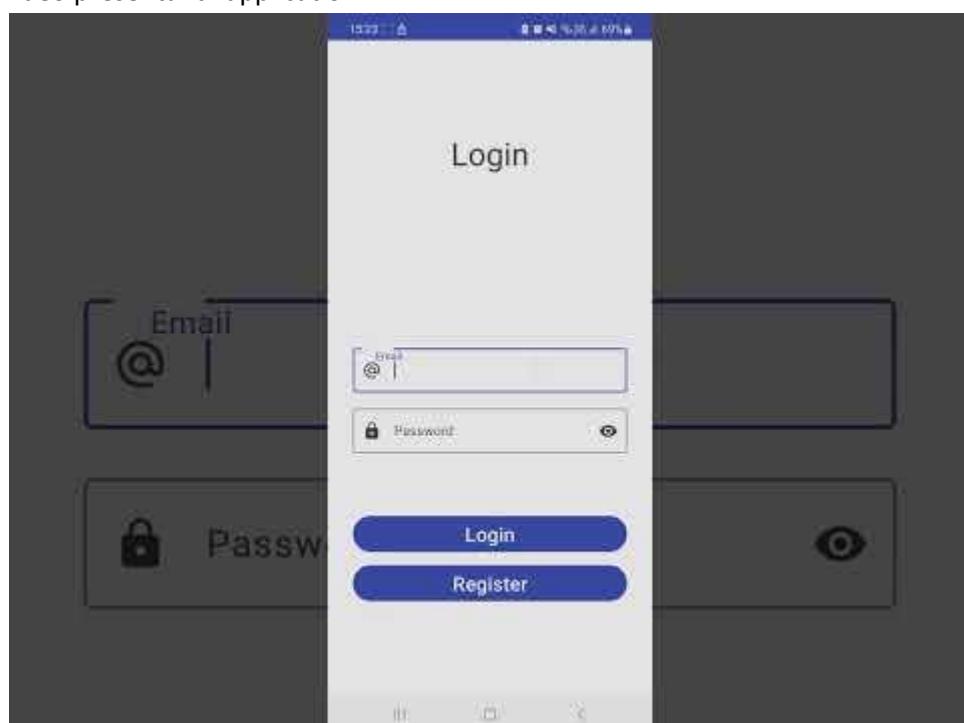
Cette application permet de consulter l'état d'un stockage de matériel informatique (smartphone et tablette) mais il faut être connecté avec un compte pour pouvoir l'utilisé.

En scannant le QR code d'un matériel, on peut soit accéder à l'ensemble de ces informations soit définir son entré ou sa sortie du stockage.

Seul l'administrateur et les utilisateurs autorisés ont la possibilité d'édition ou d'ajouter un matériel.

Et c'est lors du 1<sup>er</sup> démarrage, que le compte administrateur est défini, permettant la suppression des autres comptes ou de les autoriser à cet ajout/édition.

Voici une vidéo présentant l'application :



### 3 Développement de l'application

Pour réaliser ce projet, j'ai d'abord commencé par réaliser la maquette de l'application :

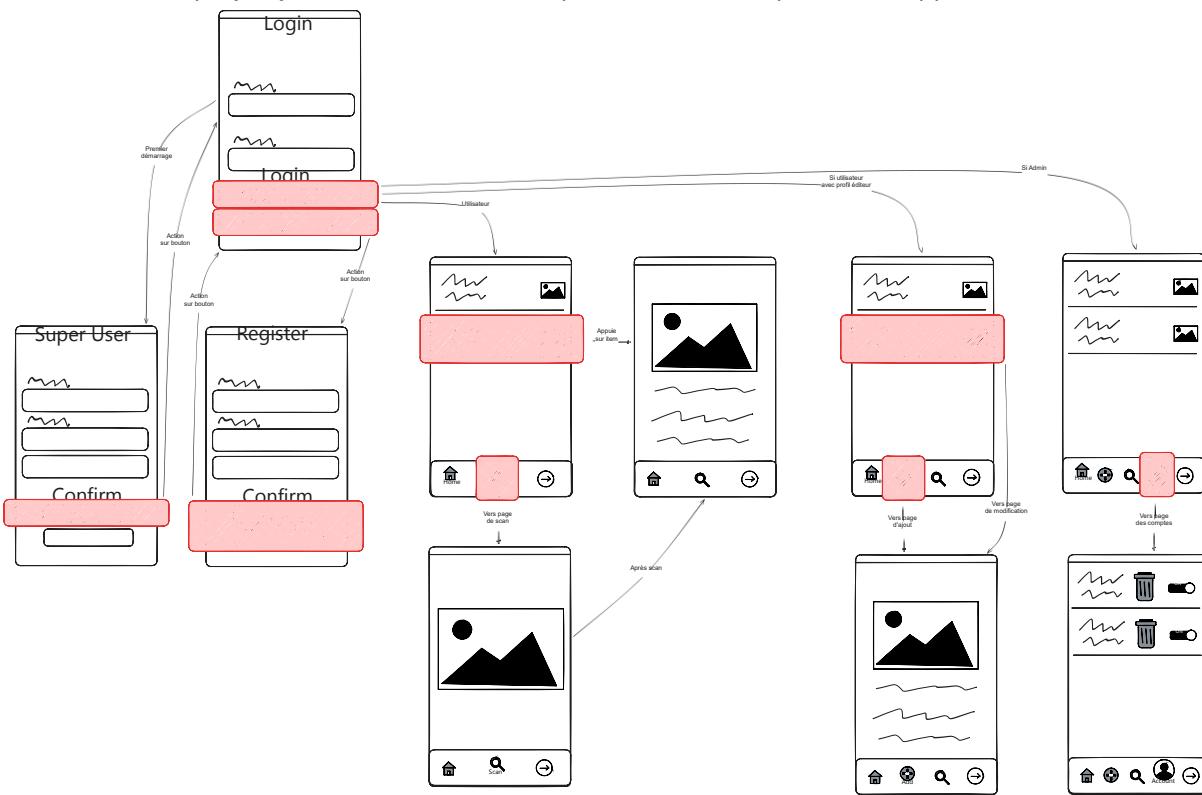


Figure 1: Plan de l'application

De plus, j'ai créé les 3 pages (Login, Register et Super User) avec leur code respectif.

Pour créer un compte, il est impératif que l'email contienne soit « @heh.be », soit « @std.heh.be » et pour le mot de passe, il doit être d'une longueur de minimum 4 caractères<sup>1</sup>.

Ainsi, nous pouvons ajouter des comptes et nous connecter avec.

Ensuite, j'ai réalisé les pages pour la barre de navigation (Home, Add, Scan et Account) et en fonction du compte connecter, la barre de navigation n'affiche que certain élément.

Tous les utilisateurs ont accès à Home, Scan et Logout, les utilisateurs autorisés ont accès à Add et l'admin à accès à Add et Account.

Chaque page à son utilité :

- la page Home est la 1<sup>ère</sup> à s'afficher après s'être connecté, elle permet de visualiser l'ensemble des produits sous forme d'une liste. Cette liste affiche le type (représenté par une image), le model, le numéro de référence, le lien du site du fabricant et une image pour représenter la disponibilité dans le stockage pour chaque produit.  
Si on appui sur le lien du site, nous sommes rediriger vers celui-ci ou si on appui sur la zone du produit nous sommes rediriger vers la page Add ;
- la page Add permet d'ajouter un produit ou après redirection du Home, elle permet de modifier (si on y est autorisé), sinon elle affichera simplement les informations du produit ;

<sup>1</sup> Demandé dans les consignes (Scopel, 2023)

- la page Scan ne contient que 3 boutons, un pour scanner et nous redirigeant vers la page Add du produit si le QR code scanné est correct et les 2 autres boutons permettent de définir la disponibilité du produit (Deposit pour remis ou Borrow pour pris)
- la page Account permet d'afficher, seulement pour l'administrateur, tous les comptes créer et de soit modifier la possibilité d'un compte d'éditer/ajouter un matériel ou de soit supprimer les comptes.

Enfin, j'ai codé le fonctionnement de chaque page de la navigation avec l'ajouter, la lecture, la modification et la suppression des éléments dans la DB (base de données).

Vous pouvez voir le schéma de ma DB<sup>2</sup>:

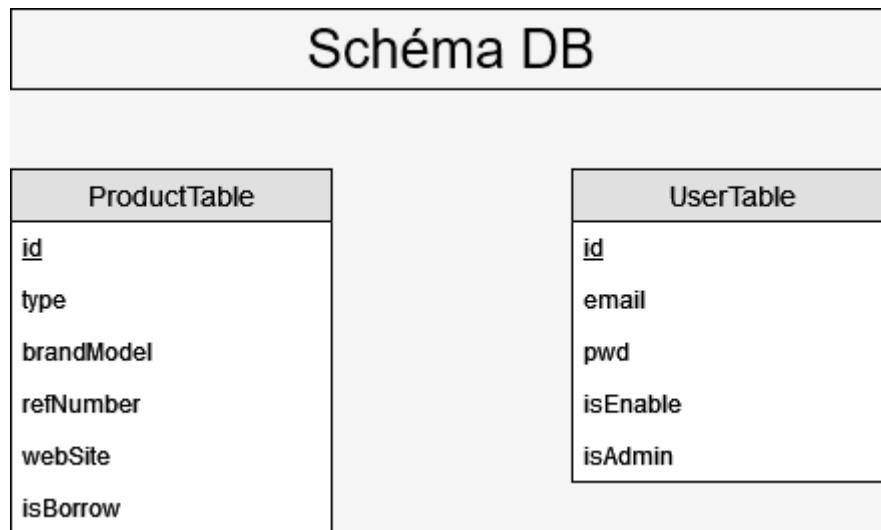


Figure 2: Schéma DB

## 4 Problème

Pour réaliser ce projet, j'ai rencontré plusieurs problèmes :

### 4.1 L'affichage du matériel

L'affichage du matériel de la DB sur la page Home ne se réalise pas car j'utilisé une List et non une MutableList. Il se trouve que lorsqu'on partage un élément de type List, pour l'affichage, une copie de ses éléments est réalisée, ainsi, lorsque le chargement asynchrone des éléments dans la DB se termine, l'affichage ne se mettait pas à jour. La MutableList, permet de corriger cela car lorsqu'on partage un élément de ce type, c'est une référence qui est partagé, ainsi, après le chargement des éléments de la DB, l'affiche se met à jour.

### 4.2 Barre de navigation actualisé

Lorsque l'on clique sur un élément de la liste (sur la page Home), cela nous redirige vers la page Add mais visuellement la barre de navigation ne changeait pas. Ceci à était réglé après de longue

---

<sup>2</sup> Réalisé via le site ([draw.io](https://draw.io), 2023)

recherche sur divers site internet (Google, 2023), vidéo YouTube, post sur StackOverflow (Stack Overflow, 2023) et recommandation de ChatGPT (OpenAI, 2023).

### **4.3     *Orientation de l'écran***

Lorsque nous sommes sur une page de la navigation et que nous changeons l'orientation de l'écran (verticale vers horizontale), nous retournons sur la page Home. Ne trouvant pas d'utilité dans l'application à être en horizontale, j'ai pris la décision de bloqué la rotation de l'écran, ainsi, l'application est exclusivement en verticale.

## **5 Amélioration**

Considèrent avoir remplis l'ensemble des consignes demandé, j'ai décidé d'arrêter le développement de l'application, pourtant plusieurs amélioration pourrait y être apporter :

### **5.1     *Partage du QRCode***

Il est possible d'ajouter un nouveau matériel en scannant un QR code, mais il est aussi possible de le faire en entrant chacune de ces informations dans l'application, hors avec ce fonctionnement, le matériel ne dispose pas de QR code. C'est pourquoi, j'ai décidé qu'il était possible de le générer lorsque nous entrons ces informations et de le partagé pour par exemple l'imprimé.

### **5.2     *Historique du stockage***

Tous les utilisateurs peuvent prendre ou déposé un matériel dans le stockage, mais il serait intéressant de savoir qui l'avait pris avant et qui la actuellement.

## **6 Conclusion**



## 7 Table des figures

Figure 1: Plan de l'application ..... 4

## 8 Bibliographie

- Google. (2023, 12 04). *Enregistrer des données simples avec SharedPreferences*. Récupéré sur Android Developers: <https://developer.android.com/training/data-storage/shared-preferences?hl=fr>
- Google. (2023, 12 04). *Fragments Implementation using Kotlin || Fragments using Kotlin || Android Studio Tutorial || 2021*. Récupéré sur Youtube: <https://youtu.be/jOFLmKMOcK0?si=ce1Pte9FLINDnqlt>
- Material Design. (2023, 12 04). *Selection controls: switches*. Récupéré sur Material Design: <https://m2.material.io/develop/android/components/switches>
- OpenAI. (2023, 12 04). *ChatGPT*. Récupéré sur ChatGPT: <https://chat.openai.com/>
- Scopel, F. (2023, 12 04). *Cours Développement mobile*. Récupéré sur Ecampus HEH: [https://ecampus.heh.be/pluginfile.php/217874/mod\\_resource/content/1/T-PINI-512\\_3IRT\\_Applications\\_sur\\_Android\\_V6\\_2\\_3.pdf](https://ecampus.heh.be/pluginfile.php/217874/mod_resource/content/1/T-PINI-512_3IRT_Applications_sur_Android_V6_2_3.pdf)
- Stack Overflow. (2023, 12 04). *How to update navigation bar selected item when I press back?* Récupéré sur stackoverflow: <https://stackoverflow.com/questions/70822974/how-to-update-navigation-bar-selected-item-when-i-press-back>

