

# Ma liste de course

## **1) Présentation générale de l'application**

Cette application a été créée dans le but de remplacer le traditionnel post-it ou morceau de papier sur lequel on griffonne la liste de courses à faire au supermarché du coin.

L'application va reproduire la gestuelle de la création de cette liste et le fait de raturer un élément lorsqu'un produit est déposé dans le caddie.

C'est aussi une idée écologiquement responsable !

## **2) Caractéristiques techniques**

Le projet comporte trois classes pour les règles métiers :

- La classe produit (Product) comporte trois attributs : un champ image, un champ nom et une « checkbox » booléenne qui simulera l'action raturer/déposer dans le caddie.
- La classe d'accès aux données ou DAO pour la persistance locale du dernier produit produit ajouté à la liste qui sera affiché à chaque ouverture de l'application.
- La classe Storage qui utilise la dépendance get\_storage pour persister une liste complète de course localement et la gestion de cette liste (CRUD).

La base de données à distance utilisée est le cloud Firebase, elle permet de stocker, éditer une collection de produits (CRUD) et fournit une banque d'images pour hydrater le champ image lors de la création de la liste de courses.

Des requêtes de tri permettront de créer un filtre de recherche de produit ou de sélectionner la bonne image à afficher par rapport à la saisie du nom de produit.

La partie vue a été réalisée avec Flutter et a été découpée en Widgets pour les différents éléments comme les boutons, barres de menu ou de navigations, containers et modal pour la création de produits.

Remarque : de nombreuses dépendances ont été utilisées pour ce projet notamment pour la gestion de Firebase, des images et de la persistance locale.

Concernant les images, le document « file » est converti en Uint8List base64 puis encodé avec un header dans la base de données afin de pouvoir être lu dans un navigateur, puis elles seront décodées dans l'application pour être affichées sous la forme :

"Image.memory(base64Decode(element.image))".

L'utilisateur n'a donc pas besoin de copier/coller des url dans ce champ, ce qui est peu pratique sur un mobile et cette possibilité a été supprimée mais rien n'empêche ce dernier de sauvegarder sur le téléphone une image trouvée sur le net et l'utiliser ensuite.

Des messages contextuels sous forme de Widget de dialogue, de type « snackbar » ou « alert », préviennent l'utilisateur en cas de saisie invalide ou d'action réussie.

Enfin une icône dédiée s'affiche sur le téléphone à la place de celle par défaut.

### **3) Manuel d'utilisation**

Le projet comporte trois pages en mode portrait, il n'a pas de gestion du mode paysage dans le cadre de cette application dans sa première version.

Ce mode pourra être envisagé dans une mise à jour future pour une gestion de listes à thèmes nécessitant un espace d'affichage plus important et plus de temps pour être implémenté.

#### **a) La page principale**

C'est le cœur de l'application. L'entête comporte la navigation principale sous forme d'icônes explicites.

L'icône du caddie renvoie vers la seconde page et l'icône + dirige vers la page de création de produit. Chaque page possède une flèche de retour à la page principale. La page principale est communicante avec la page du caddie pour y transférer ses données.

Un champ de saisie permet de créer un élément de la liste de courses. Si le produit possède un visuel associé au texte saisi dans la base de données, l'image correspondante sera affichée, dans le cas contraire, un visuel par défaut sera affiché. A noter que la suggestion en auto-complétions du clavier software peut être utilisée également avec succès.

La liste est ordonnable par « drag and drop » et le « swipe » permet de supprimer un élément de cette liste.

Si l'utilisateur veut sauvegarder sa liste, il suffit de taper l'action éponyme du bouton dans le menu de gestion de la liste du pied de page. Il suffira de faire de même pour charger cette liste ou la supprimer complètement.

Cependant si la liste est déjà chargée à l'écran, la suppression ne sera effective qu'a posteriori.

Enfin il suffit de cocher la case (checkbox) pour ajouter le produit au caddie, le produit est transféré en seconde page et disparaît de la liste initiale.

#### **b) La page caddie**

La seconde page est très simple, elle ne fait que recevoir les produits transférés de la page principale et ajoutés dans le caddie.

Aucune action n'est possible sur cette page excepté la suppression totale du caddie grâce à l'icône dans l'entête afin de recevoir une nouvelle sélection.

### **c) La page gestion des produits**

Cette page permet la création de contenu dynamique dans la base de données, chaque produit crée est éditable ou peut être supprimé.

Pour ce faire il suffit de taper sur le « + » en bas de page pour afficher le modal de création/édition.

Les images peuvent être sélectionnées soit à partir des données stockées sur le téléphone ou soit par capture directe avec l'appareil photo avec les icônes idoines.

Si la caméra est sélectionnée, la photo prise n'est pas sauvegardée sur le téléphone et évite à l'utilisateur de devoir les supprimer ultérieurement.

Un container affiche par défaut une image d'attente afin d'éviter la valeur nulle pour l'image tant qu'elle n'a pas été sélectionnée.

Deux champs de type contrôleur texte permettent de saisir le nom du produit et récupérer l'image sélectionnée. A noter la présence d'un petit décalage temporel sur téléphone entre la sélection et l'affichage.

Pour obtenir l'affichage final de l'image dans son champ de saisie, il suffit de faire un « tap » en premier sur le champ de saisie du nom du produit pour activer le clavier software et ensuite répéter l'action sur le champ image afin d'hydrater ce dernier.

La dernière image choisie reste persistante en mode création en attendant une nouvelle sélection, seul le champ texte est vidé.

Enfin, cette page permet aussi de rechercher un produit existant en base de données afin de ne pas créer de doublons mais aussi de connaître les homonymes potentiels puisque par défaut je permets l'affichage de cinq produits issus de la base de données sur cette page à cet effet.

Par exemple, si le mot « pain » est sélectionné, on pourrait avoir aussi à l'écran « pain complet » ou « pain de mie » en sus de la valeur recherchée s'ils existent.

### **Conclusion :**

J'ai testé l'application intensivement sur mon propre téléphone afin de détecter les soucis d'expérience utilisateur et les bugs éventuels afin d'y apporter des correctifs.

