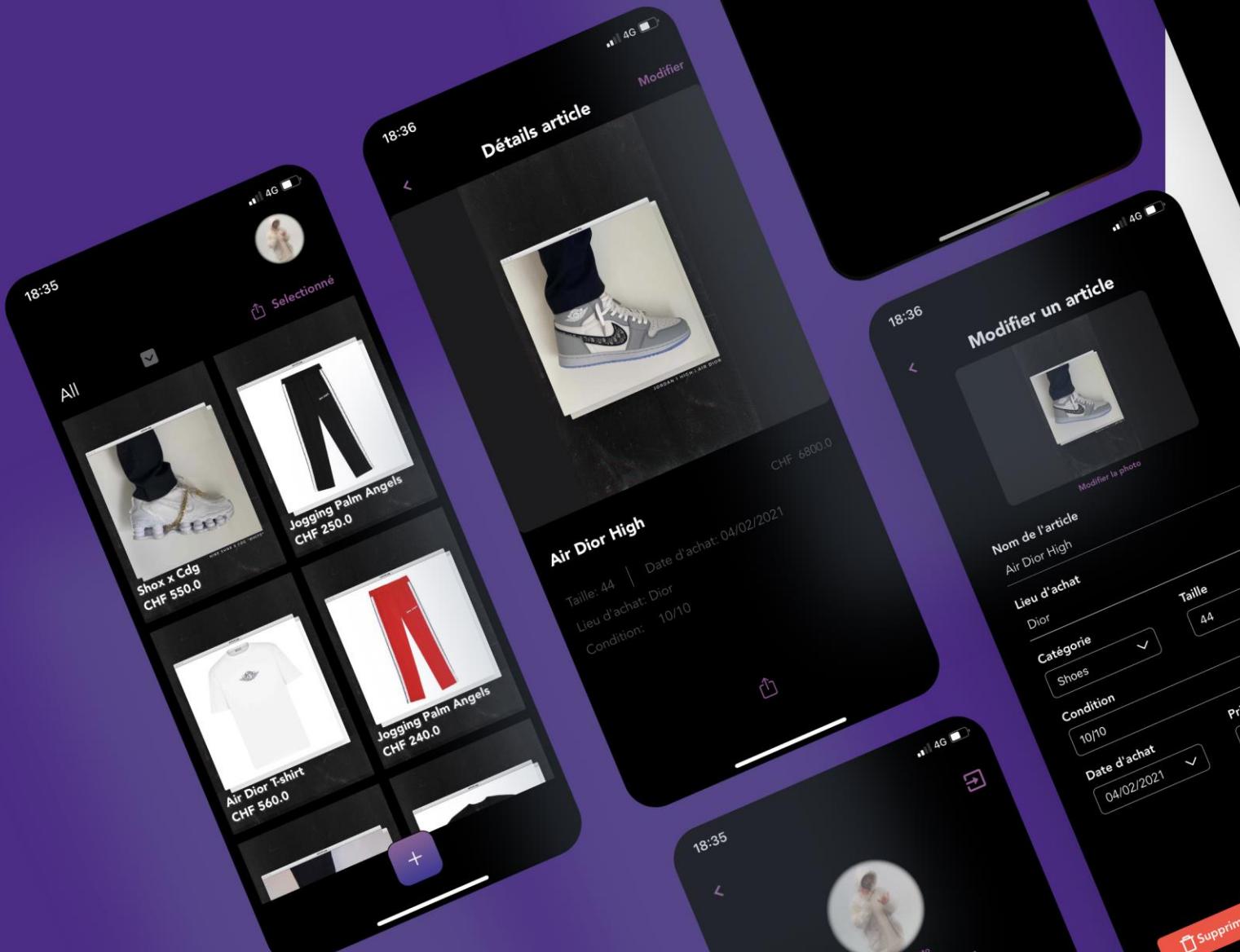


inventaireHabits

Projet TPI du 05.05.21 au 04.06.21



Chef de projet : Gruaz Gilbert

Experts : Carrel Xavier
Perez Carlos

Table des matières

1 GLOSSAIRE	4
2 ANALYSE PRÉLIMINAIRE.....	7
2.1 INTRODUCTION	7
2.1.1 Titre	7
2.1.2 Description	7
2.1.3 Contexte.....	7
2.2 OBJECTIFS	8
2.3 PLANIFICATION INITIALE ET DÉTAILLÉE	10
3 ANALYSE / CONCEPTION.....	11
3.1 CONCEPT	11
3.1.1 Méthodologie de gestion de projet : En cascade	11
3.1.2 Maquettes graphiques / Liaison des pages :	12
3.1.3 Modèle Conceptuel des Données	21
3.1.4 Besoins des utilisateurs.....	21
3.2 STRATÉGIE DE TEST	22
3.3 RISQUES TECHNIQUES	22
3.4 PLANIFICATION	23
3.5 DOSSIER DE CONCEPTION	24
3.5.1 Matériel Hardware / Système d'exploitation.....	24
3.5.2 Logiciels	24
3.5.3 Suivi de projet.....	25
3.5.4 Base de données : Firebase	25
3.5.5 Architecture du programme : Modèle-Vue-Contrôleur.....	27
3.5.6 CocoaPods – Gestionnaire de dépendances	28
3.5.7 Cas d'utilisation	29
3.5.8 Modèle Logique de Données.....	30
4 RÉALISATION	31
4.1 LISTE DES FICHIERS : RÉPERTOIRE	31
4.2 LISTE DES FICHIERS : CODE SOURCE.....	32
4.3 FIREBASE : AUTHENTIFICATION.....	33
4.4 FIREBASE : FIRESTORE.....	34
4.5 CRÉATIONS DES CELLULES D'ARTICLE.....	35
4.6 GESTION DES CATÉGORIES.....	36
4.7 VERSIONS DU PROJET	37
4.8 BASE DE DONNÉES : FIREBASE	37
4.9 CHARTE GRAPHIQUES : APPLICATION.....	37
4.10 DESCRIPTION DES TESTS EFFECTUÉS	38
4.11 ERREURS RESTANTES	42
4.12 LISTE DES DOCUMENTS FOURNIS.....	43
5 CONCLUSION.....	43
5.1 BILAN DES FONCTIONNALITÉS DEMANDÉES	43
5.2 BILAN PERSONNEL	44
5.1 BILAN DE LA PLANIFICATION.....	45
6 ANNEXES	46
6.1 RÉSUMÉ DU PROJET INVENTAIRE HABITS.....	46
6.2 WEBOGRAPHIE	47
6.3 TABLE DES ILLUSTRATIONS	48

6.4	DOSSIER DES TESTS.....	48
6.5	ARCHIVES DU PROJET.....	48
6.6	PLANIFICATION – JOURNAL DE TRAVAIL	49
6.7	JOURNAL DE TRAVAIL	56
6.8	MANUEL D'UTILISATION	69
6.1	MANUEL D'INSTALLATION	74

1 GLOSSAIRE

B

Base de données de type « NoSQL »

NoSQL signifiant « not only SQL » désigne une famille de systèmes de gestion de bases de données (SGBD) non tabulaires et stockent les données différemment des tables classiques relationnelles.

Les principaux types sont « Document », « Key-value », « wide-column » et « Graph ». Ils fournissent des schémas flexibles et évoluent facilement avec de grandes quantités de données et des charges utilisateurs élevées.

Firebase est une base de données NoSQL mais il en existe d'autres comme MongoDB ou Cassandra.

Da

C

Cocoa

Cocoa est une API d'Apple pour le développement orienté objet sur système d'exploitation macOS. Les applications Cocoa sont construites en utilisant l'ensemble des outils de développement d'Apple, comme Xcode, Interface Builder, leurs langage de programmation Objective-C et Swift mais il peut également être utilisé via des outils comme Ruby et C#.

D

Dropdown Menu

Le dropdown est le terme anglais pour parler d'une liste déroulante. Il s'agit d'un champ contenant une liste prédefinie de valeurs dans laquelle on peut sélectionner une des valeurs, celle-ci sera prise en compte, c'est un objet très utile pour la complétion de formulaire ou lorsqu'il y a des choix multiples.

F

Framework

Un Framework est une collection de code tout fait et prêt à l'usage pour les développeurs.
Il leur permet donc de réutiliser du code existant, non seulement pour gagner du temps mais aussi pour intégrer une sorte de standardisation des applications.
Ils ont été conçus et sont utilisés pour modeler l'architecture des logiciels applicatifs.
Cocoa est un framework.

I

Interface Builder

C'est un outil de développements graphique, permettant de gérer la partie interface d'une application, il fait partie du programme Xcode, il a été développé en 1988, il était à l'origine utilisé pour NeXTSTEP.
Il fournit une multitude d'éléments graphique et d'objets, utilisable au sein de l'interface graphique d'une application, comme des champs de texte, des objets permettant de Scroll une page, des boutons etc.
C'est un outil pratique pour la communauté car il est intuitif et l'on peut y implémenter de nouveaux objets via des bibliothèques ou autres.

J

JSON – JavaScript Object Notation

C'est un format de données textuelles, il permet de représenter de l'information structurée comme le permet XML par exemple.

Un document JSON comprend deux types d'éléments structurels :

- des ensembles de paires « Nom » (Alias « clé ») / Valeur
- des listes ordonnées de valeurs

Pour la réalisation de ce projet, j'ai utilisé ce format de données pour stocker mes données d'application.

L

Langage machine

Le langage Machine ou Code machine, est ce qui sera interprété par le processeur de l'ordinateur qui exécutera l'application ou un programme.

C'est le seul langage que puisse interpréter un processeur, il est composé d'instructions et d'éléments écrit en code binaire suite à l'interprétation du code grâce au compilateur du langage de programmation de l'application exécuté.

P

Pods

Dans l'environnement de développement IOS, le gestionnaire de dépendances utilisé est CocoaPods, un pod est le nom donné aux bibliothèques externes utilisé sous IOS.

S

Software Development Kit ou Kit de développement¹

C'est ce qui constitue l'ensemble d'outils logiciels destinés aux développeurs, son objectif est de faciliter le développement d'un logiciel.

Il comporte en général :

- Un compilateur (traduit le fichier source en langage de programmation vers un fichier destination en langage machine)
- Un éditeur de liens (lie ensemble les différents modules et les bibliothèques de routines en un seul fichier exécutable « programme »)
- Des bibliothèques contenant des routines prêtes à l'utilisation.

¹ https://fr.wikipedia.org/wiki/Kit_de_d%C3%A9veloppement

2 ANALYSE PRÉLIMINAIRE

2.1 Introduction

2.1.1 Titre

Inventaire Habits

2.1.2 Description

Cette application mobile sera réalisée en Swift, un langage de programmation pour les mobiles sous IOS.

Il s'agit d'une application qui fonctionnera sur un iPhone, qui permettra à un utilisateur d'inventorier et de gérer sa garde-robe d'habits/chaussures.

Les fonctionnalités attendues sont :

- Saisie des informations concernant un article, tel que, par exemples, le prix d'achat, date d'achat, taille, lieu d'achat, condition (usure), nom de l'article etc.
- Mise à jour des informations et suppression d'articles.
- Recherches d'articles à l'aide de critères, avec un résultat qui donne une liste dans laquelle on peut sélectionner un article en particulier et obtenir ainsi les informations le concernant.
- Production d'une liste d'habits selon des critères qui peut être envoyée par la suite à l'aide de l'e-mail ou sms à un magasin de ventes d'habits, ou à la Chaîne du Bonheur

L'application servira de gestionnaires pour l'utilisateur, il aura la possibilité d'ajouter de nouveaux articles, en supprimer et en modifier, cela s'affichera sous la forme d'une collection que l'on pourra parcourir à l'aide de différentes catégories (T-shirt, Hoodie) etc.

2.1.3 Contexte

Ce projet est réalisé dans le cadre de mon TPI afin de valider mon CFC, il me permettra de démontrer mes capacités acquises lors de mon apprentissage.

La thématique du projet a été choisie par moi-même avec la validation de mon chef de projet Gruaz Gilbert ayant pris en compte ma passion autour de l'univers de la mode, des vêtements et des chaussures.

2.2 Objectifs

Le candidat est responsable de livrer, à son chef de projet et aux deux experts, les éléments de suivi de projet suivants :

- Une planification initiale (jour 1) et une planification aussi détaillée que possible (jour 3)
- Un tableau des besoins utilisateurs (clients et/ou vendeurs)
- Un rapport de projet
- Un journal de travail
- Les instructions pour la mise en œuvre de l'application, ou à une simulation de celle-ci

En plus de cela, l'application mobile sera évaluée sur les 7 points spécifiques suivants (Point A14 à A20):

1. Production d'un modèle de données pertinent pour les informations concernant les habits/Article :

- Création d'une base de données
- Implémenter les données suivantes dans la base de données :
 - Image de l'article*
 - Nom *
 - Taille *
 - Prix *
 - Lieu d'achat *
 - Condition *
 - Date d'achat

2. Mise en œuvre d'une interface pour la saisie de ces informations :

- Page ajout d'un article :
 - Toutes les informations concernant l'article
 - Liaison avec la base de données (écriture DB)
 - Bouton de validation
 - Bouton retour

3. Mise en œuvre d'une interface de modifications d'informations :

- Page modification d'un article :
 - Toutes les informations concernant l'article
 - Liaison avec la base de données (lecture et écriture DB)
 - Bouton de validation
 - Bouton de suppression de l'article
 - Bouton retour

4. Mise en œuvre d'une interface de suppressions d'articles :

- Bouton disponible sur la page de modification, permettant la suppression d'un article.

5. Mise en œuvre d'une interface de recherches d'habits selon des critères :

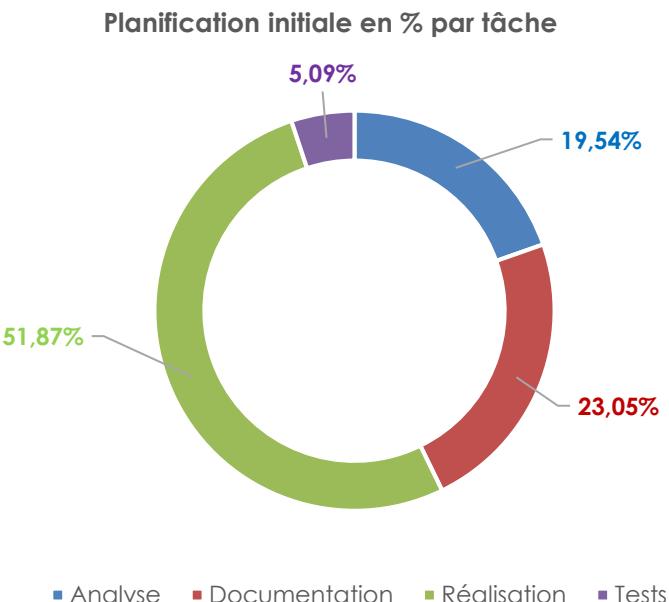
- Page principale de l'application :
 - Affichage des articles ajoutés sous forme de collection
 - Bouton ou Menu déroulant pour sélectionner la catégorie d'articles souhaité : « T-shirt, Pull, Pantalon, Chaussures, etc. »

6. Mise en œuvre d'une liste d'habits selon des critères, qui pourra être communiquée par e-mail, sms ou autres :

- Système permettant de sélectionner plusieurs articles pour les partager par e-mail, What's App, SMS, etc.

7. Les commentaires dans le code sont clairement rédigés avant de coder, de façon à permettre une réécriture dans un autre langage comme pour Android par exemple.

2.3 Planification Initiale et détaillée



La planification initiale a été réalisée le premier jour du projet et elle permet de répartir les tâches à effectuer lors des 5 semaines du projet TPI.

Pendant ce projet, chaque journée est divisée en quart d'heure et nous avons en tout pendant ce projet 3 journées entière par semaine et 1 demi-journée.

La planification initiale est consultable en annexes.

Annexe : [Planification](#)

3 ANALYSE / CONCEPTION

3.1 Concept

3.1.1 Méthodologie de gestion de projet : En cascade

Lors de la réalisation de ce projet, je vais utiliser la méthode de travail dit « **en cascade** ».

Je vais donc organiser mon projet de manière séquentielle, lorsque je finirais une étape, je passerai donc à la suivante en vérifiant la précédente, c'est une méthode très adaptée au développement d'applications.

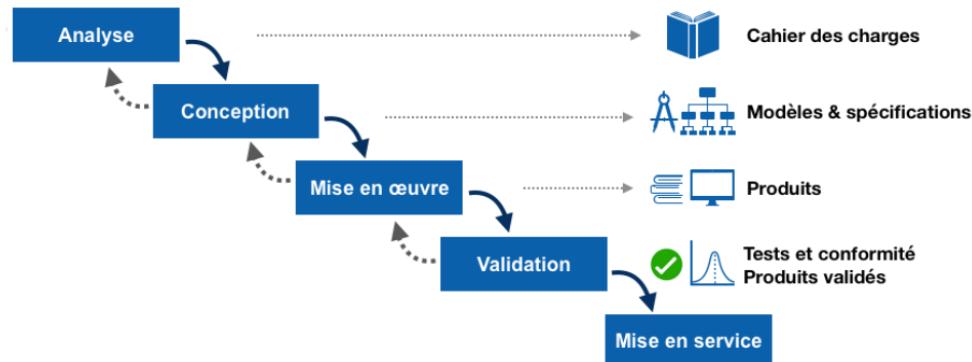


Figure 1 [Modèle en cascade - Page wikipedia](#)

Analyse :

L'analyse du projet comprend, la réflexion concernant les besoins de l'utilisateur selon le cahier des charges, ainsi que la réalisation de la planification du projet.

Conception :

La conception du projet comprend la réalisation des maquettes graphiques de l'application, du tableau des besoins de l'utilisateur, du MCD et du MLD de la base de données.

Ensuite, la description en détail des fonctionnalités qui devront apparaître sur les différentes pages de l'application.

Mise en œuvre :

La mise en œuvre comprend le développement de l'application selon le cahier des charges ainsi que la retranscription du développement de l'application sur le rapport du projet.

Validation :

La validation du projet consistera à effectuer les tests attestant que les besoins de l'utilisateur sont respectés et satisfaits.

Mais en plus que l'application fonctionne correctement et comprend toutes les technologies décrites dans le cahier des charges.

Mise en service :

Finalement, la mise en service du projet dans mon contexte, consistera à la remise du projet final à mon chef de projet ainsi qu'aux experts.

Et comprendra également la présentation du projet.

3.1.2 Maquettes graphiques / Liason des pages :

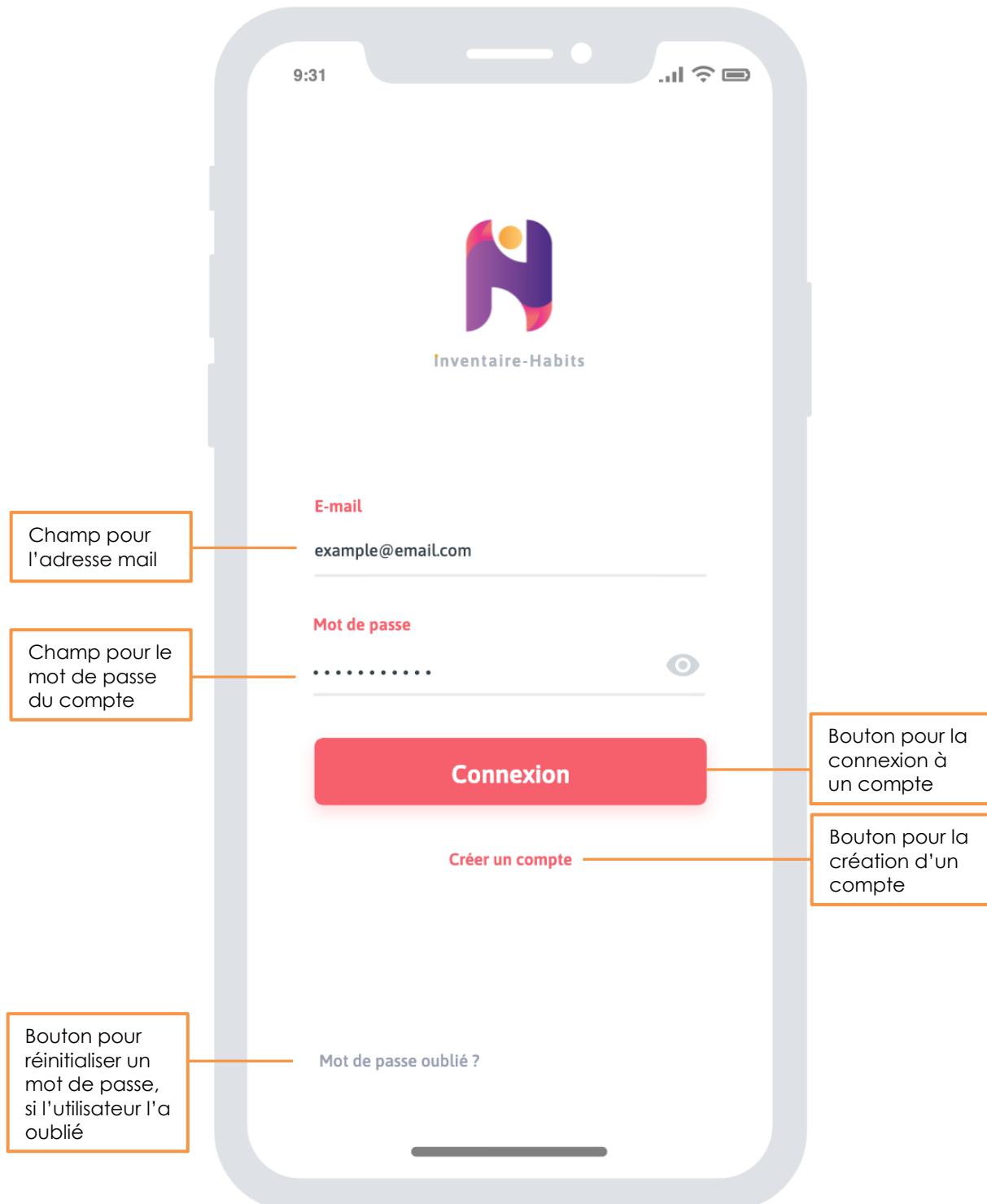


Figure 2 Maquette : Page de connexion

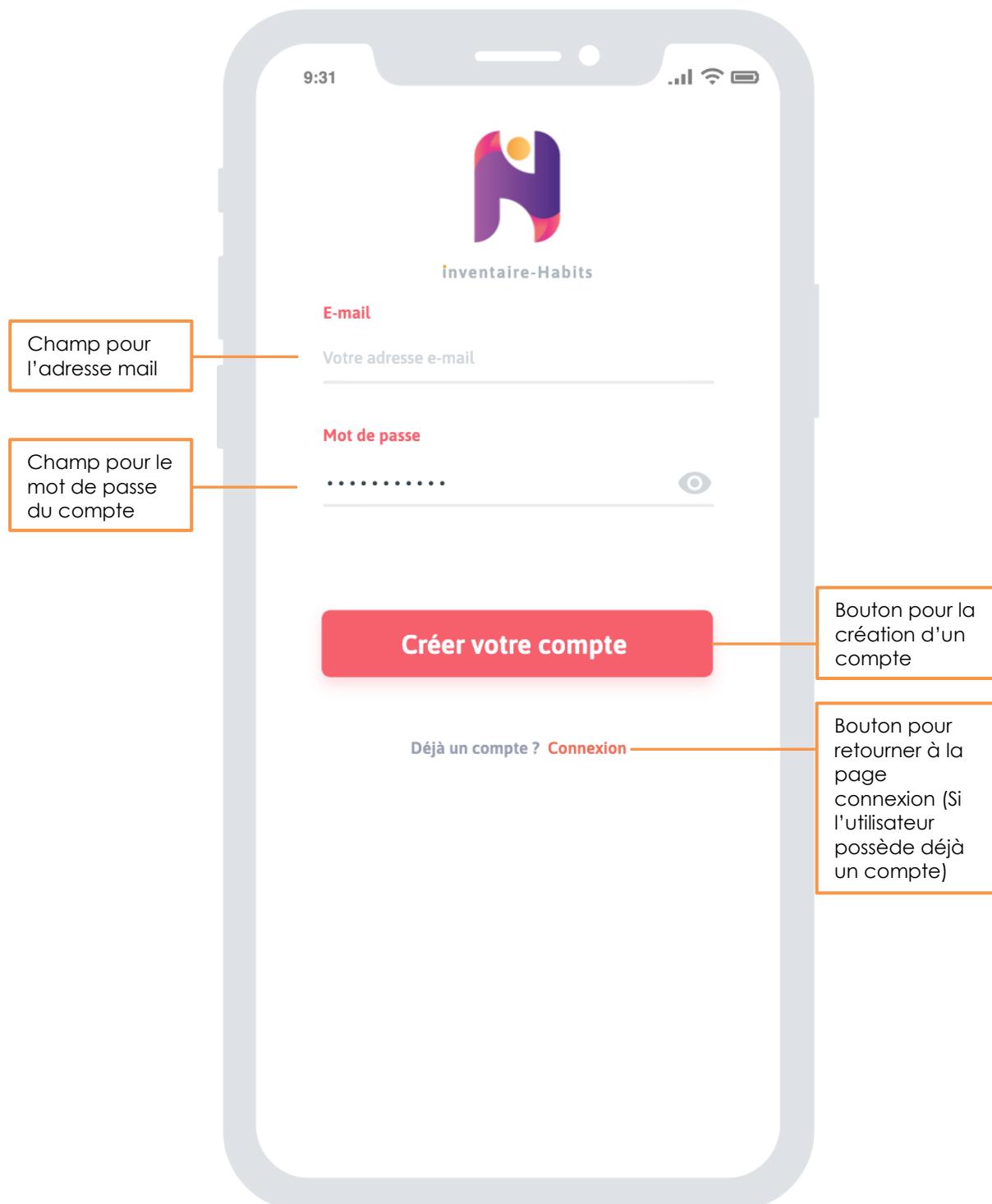


Figure 3 Maquette : Page de création de compte

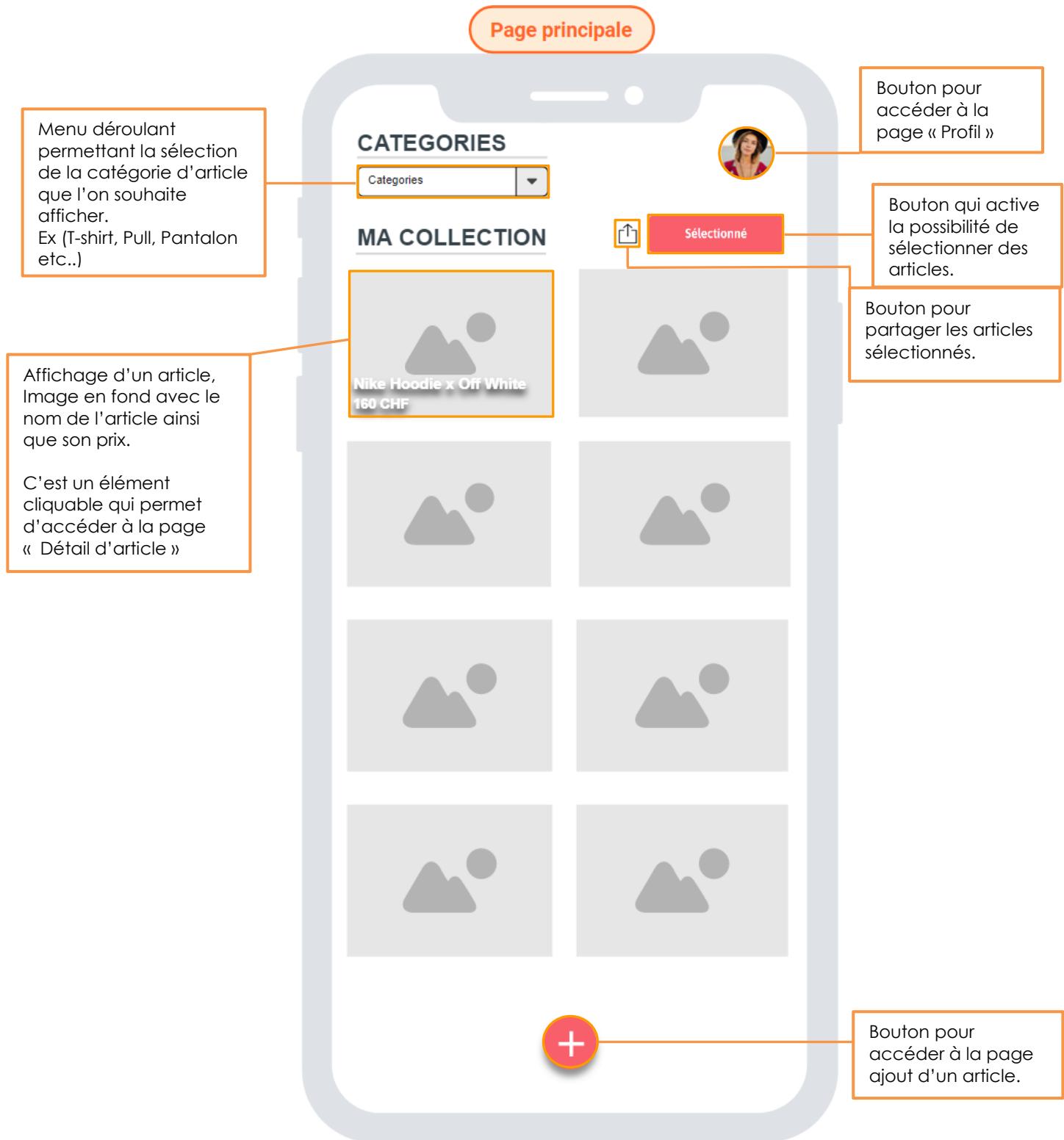


Figure 4 Maquette : Page principale de l'application

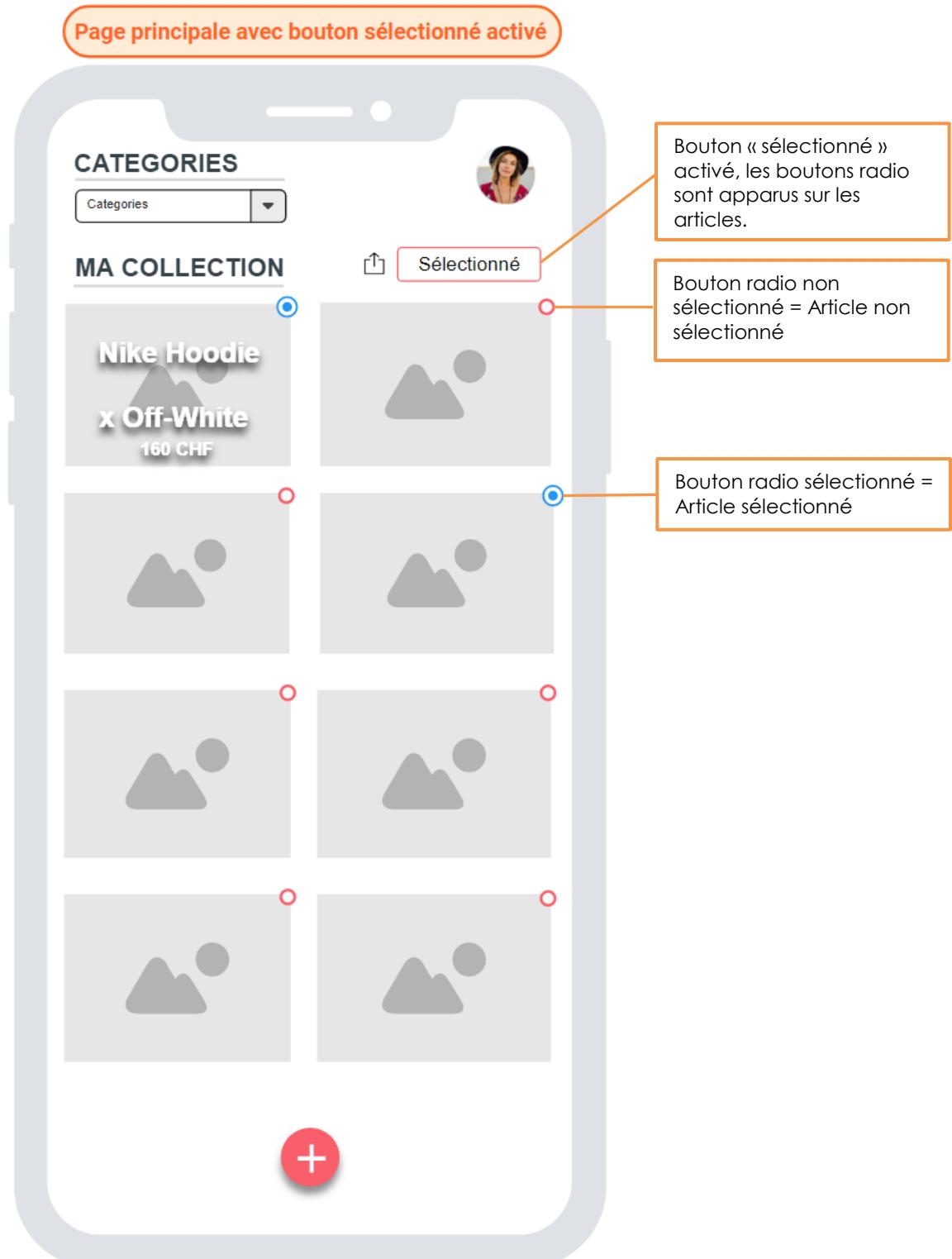


Figure 5 Maquette : Page principale de l'application (Bouton sélection activé)

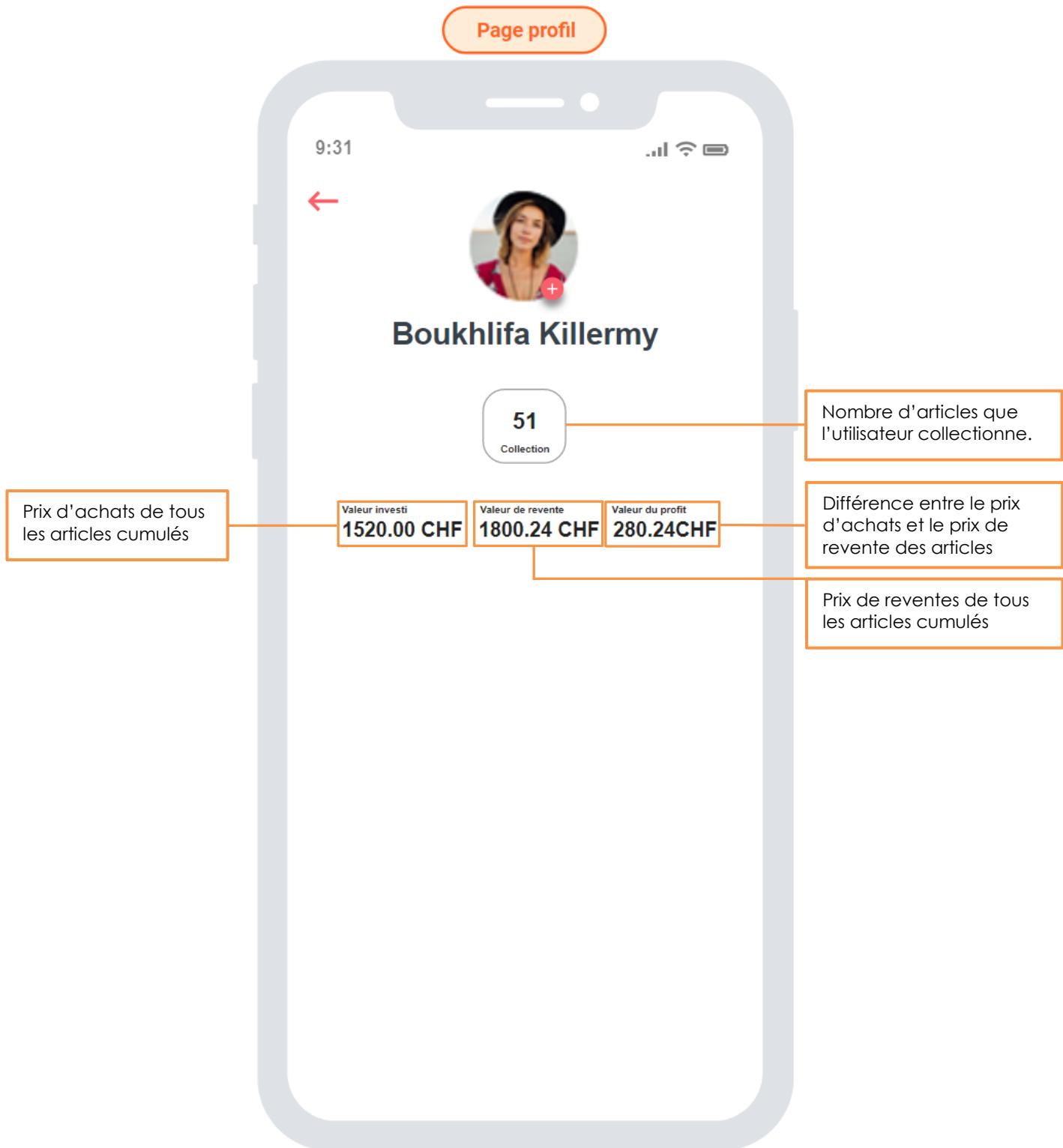


Figure 6 Maquette : Page du profil utilisateur



Figure 7 Maquette : Page ajout d'un article

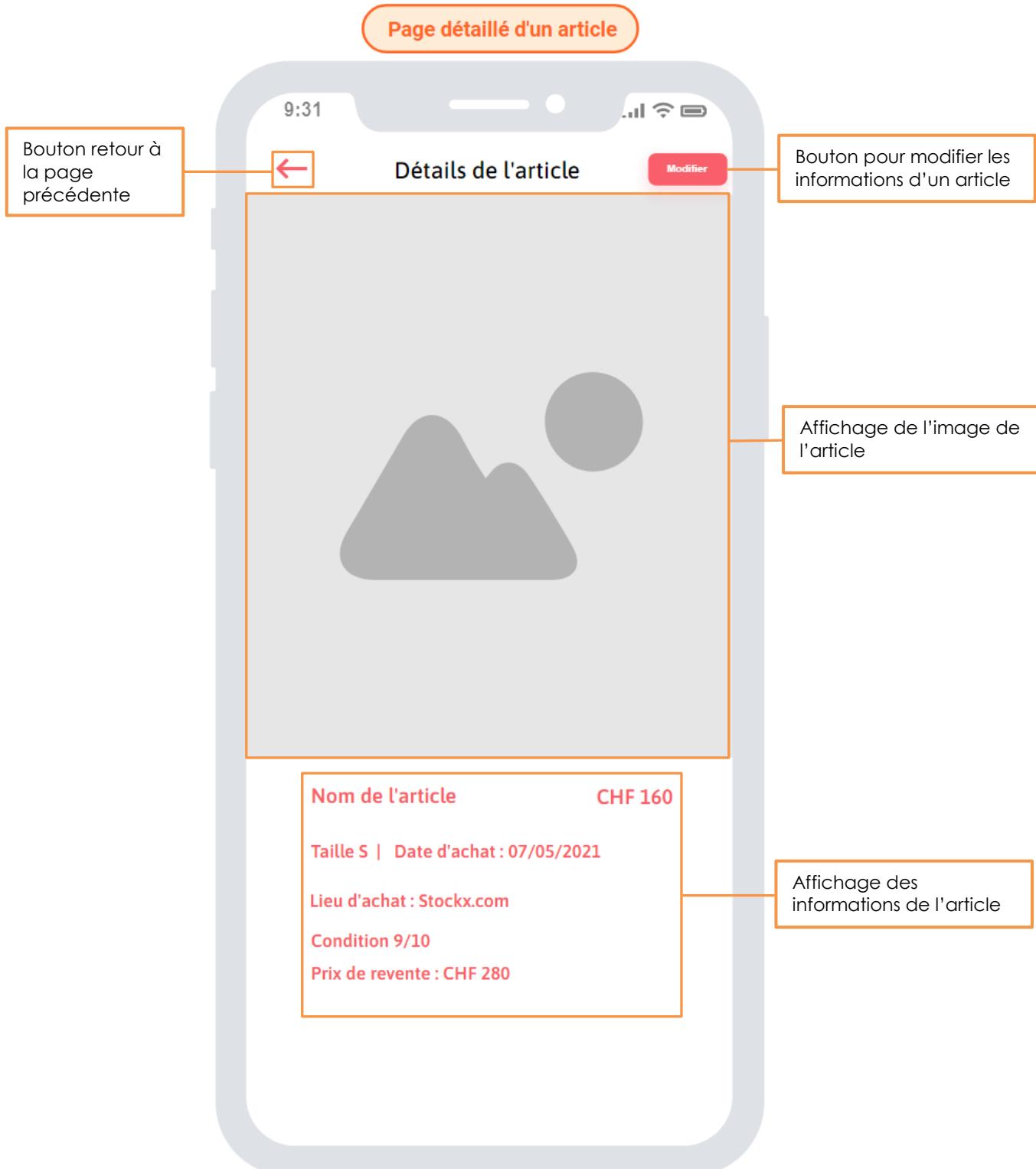


Figure 8 Maquette : Page détail d'un article

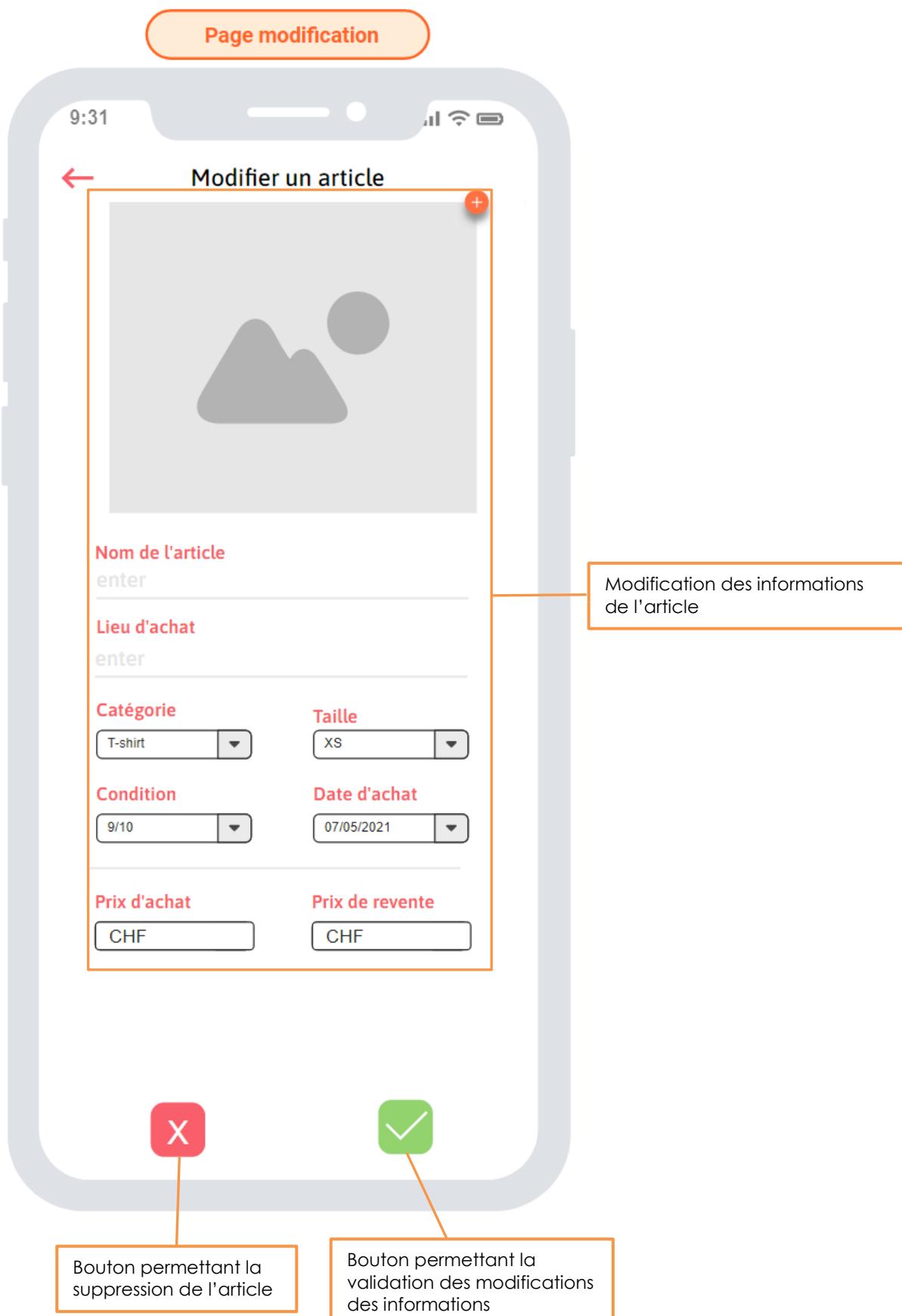


Figure 9 Maquette : Page modification d'un article

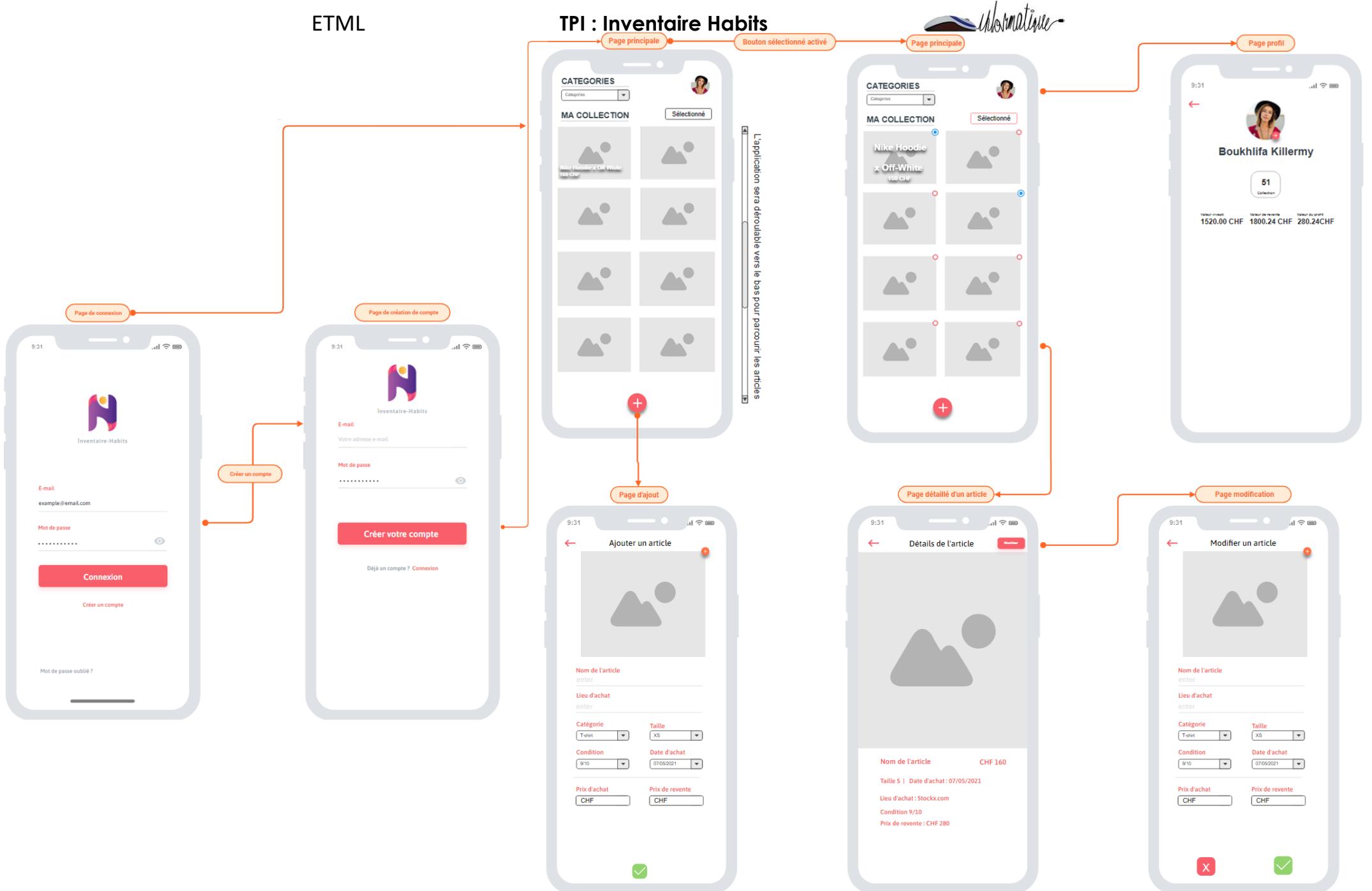


Figure 10 Maquette : Navigation de l'application

3.1.3 Modèle Conceptuel des Données

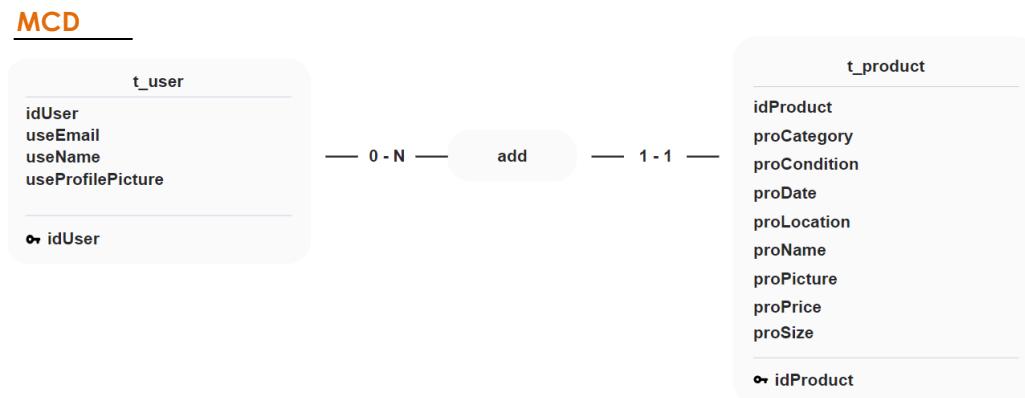


Figure 11 Maquette : Modèle conceptuel des données

3.1.4 Besoins des utilisateurs

Besoins des utilisateurs	Page répondant à ces besoins.
Enregistrement de ses informations.	Page création d'un compte/Base de donnée Firestore
Ajouts d'articles dans la collection de l'utilisateur	Page principale « Gallerie » pour parcourir et trier tous les articles de l'utilisateur à l'aide du bouton se trouvant sur la page principale. Page détail d'un article, pour consulter en détail les informations concernant un article.
Modifier les informations d'un article	Page « Modification » permettant de modifier les informations d'un article, changer le nom, description, l'image etc. D'un article précédemment ajouté.
Supprimer un article de sa collection	Page « Modification » bouton permettant de supprimer un article.
Obtenir un bilan de la collection de l'utilisateur	Page « Profil » permettant à l'utilisateur d'obtenir des informations concernant sa collection complète.
Partager les articles ce trouvant dans la collection de l'utilisateur	Page « Principale – Galerie » Bouton sélection pour sélectionner plusieurs articles et partager leurs informations via E-mail, SMS, What's App etc...

3.2 Stratégie de test

Les tests me permettront de savoir si mon projet se déroule correctement et de détecter d'éventuels problèmes survenus lors de la réalisation du projet.

J'effectuerai vers la fin de mon projet des tests afin de corriger de potentiels problèmes.

Les tests seront notés de la manière suivante :

- Test validé



- Test partiellement validé



- Test non validé



Date	Description	Conséquence	Résultat
La date de réalisation du test	Description de l'élément testé sur le projet.	Que s'est-il passé durant le test.	Est-ce que le test est validé ou non.

Les tests seront effectués manuellement sur l'application, ils seront réalisés dans l'ordre d'apparition des pages.

3.3 Risques techniques

Concernant la complexité de ce projet, je risque d'avoir des difficultés quant aux gestions de données étant donné que c'est un élément que je n'ai jamais exploré dans une application IOS.

Lors de ce projet, je vais également devoir me documenter durant mes périodes de projet, mais aussi lors de mon temps libre afin de gagner du temps et me consacrer uniquement à la réalisation lorsque je me trouverais en période de projet.

Afin de faciliter le stockage des données, je compte réaliser une gestion des utilisateurs avec un système d'authentification à l'aide de Firebase, ce qui me permettrait d'utiliser Firebase pour stocker/modifier/supprimer mes données.

Le système d'authentification était le sujet sur lequel je me suis formé lors du pré-TPI, ce qui me faciliterait sa mise en place et me permettrait un gain de temps considérable.

3.4 Planification

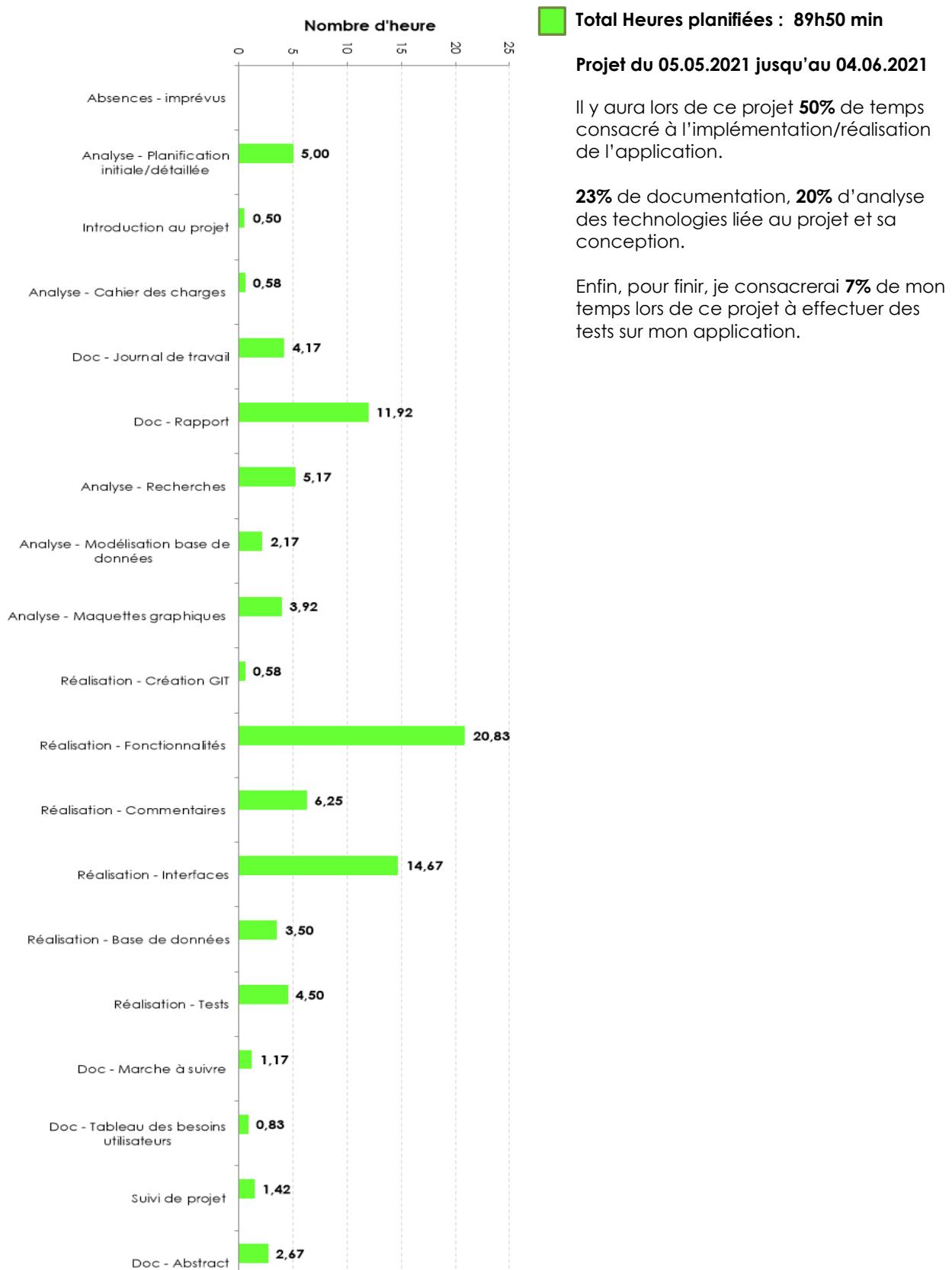


Figure 12 Graphique : Bilan de la planification initiale

3.5 Dossier de conception

3.5.1 Matériel Hardware / Système d'exploitation

1. Ordinateur ETML - Windows 10 Education – Version 1909

3,40 GHz Intel Core i7-6700 CPU

32 Go de mémoire RAM

2. MacBook Pro - MacOS Big Sur – Version 11.3

2,2 GHz Intel Core i7 quatre cœurs

16 Go de mémoire RAM / 1600 MHZ DDR3

Intel Iris Pro 1536 Mo

3. Iphone 11 - IOS – Version 14.5.1

Téléphone sur lequel l'application sera testée et devra être fonctionnelle.

3.5.2 Logiciels

1. Moqups.com / Overflow.io / Adobe Photoshop 2020/ Lucidchart.com

Outils utilisés pour la réalisation des maquettes graphiques de l'application, des modèles relationnels ainsi que les éléments visuels qui composeront l'application (Logo, Icône, Bouton etc.)

2. Xcode – Version 12.5

Outil utilisé pour le développement de mon application IOS

3. Swift 5

Langage de programmation avec lequel a été développé mon application.

4. CocoaPods – Version 1.10.7

Gestionnaire de dépendances, il m'a permis d'installer des API ou des éléments de différentes librairies pour réaliser mon application.

5. Firebase – SDK (Software Development Kit) 8.0.0 IOS

Service utilisé pour l'hébergement et l'authentification au sein de mon application.

6. GitHub / GitHub Desktop – Version 2.8.1

Service d'hébergement de mon projet en ligne

(Répertoire : https://github.com/KillermyB/TPI_inventaireHabits_BoukhliFAK)

Outil permettant d'effectuer mes commits directement depuis mon bureau.

7. FireBase – Firestore

Service d'hébergement Cloud des données de mon application.

8. Suite Office 365 – Version 2016

Suites de logiciels utilisés pour la réalisation de ma documentation.(Microsoft Word, Excel, Outlook et Powerpoint)

9. Scribens – Reverso Correcteur

Outils de correction orthographique utilisés lors de la rédaction de la documentation de mon projet.

3.5.3 Suivi de projet

Lors de ce projet, je vais permettre à mon chef de projet ainsi qu'aux experts de suivre mon projet en temps réel via un [répertoire Git](#)² mis à jour quotidiennement.

Je remplirai également de manière quotidienne un journal de travail attestant de mon avancement dans le projet qui permettra de suivre les différentes tâches effectuées et finalement comparer ma planification de projet avec sa réalisation.

Je notifierai également mon chef de projet et mes experts chaque mercredi et vendredi en fin de journée afin de leur mettre à disposition la dernière version de mon rapport et de mon journal de travail quotidien.

3.5.4 Base de données : Firebase

1. Firebase Authentication

Pour réaliser ce projet, j'ai utilisé le service d'authentification de Google : Firebase Authentication

Il fournit des services de backend, des SDK simples d'implémentation et une multitude de bibliothèques d'interface d'authentification. (Google, Facebook, Twitter, etc.)

J'ai implémenté pour mon projet l'authentification via une adresse-mail et un mot de passe, l'utilisateur peut donc créer son compte, il obtient alors un User-ID défini par Firebase.

Cet User-ID sera ensuite vérifié avec les informations de connexion par Firebase et elles seront renvoyées une fois vérifiées vers mon application pour confirmer la connexion ou refuser une connexion, ce qui me permet d'avoir une application sécurisée et fiable.

Identifiant	Fournisseurs	Date de création	Dernière connexion	UID utilisateur
killermy1@hotmail.fr		13 mai 2021	29 mai 2021	c8d000UW70NzpcEGNXLuyV6Z3...

Figure 13 Firebase Authentication : Page contenant les Users

² https://github.com/KillermyB/TPI_inventaireHabits_BoukhlikaK.git

2. Firebase – Firestore

Concernant le stockage des données de mon application, j'ai utilisé un autre service de Firebase : Firestore Database

C'est une base de données cloud NoSQL qui a la particularité d'être flexible et évolutive.

J'ai utilisé un système de stockage orienté documents qui consistent à identifier un élément via une clé liée à une valeur, qui dans ce cas, est un document, dans le document, nous y stockons une liste de champs qui a une valeur associée.

Ces documents sont de type JSON.

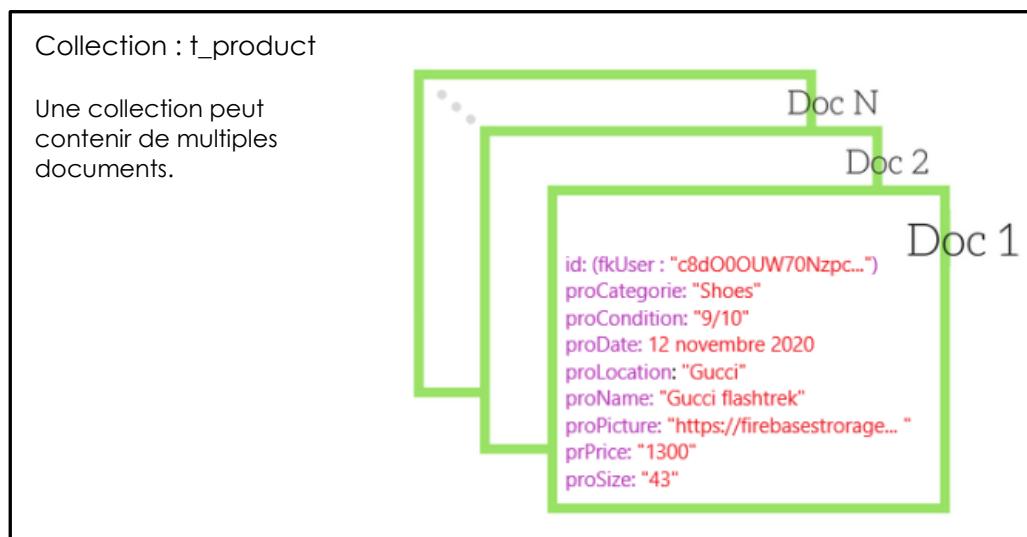


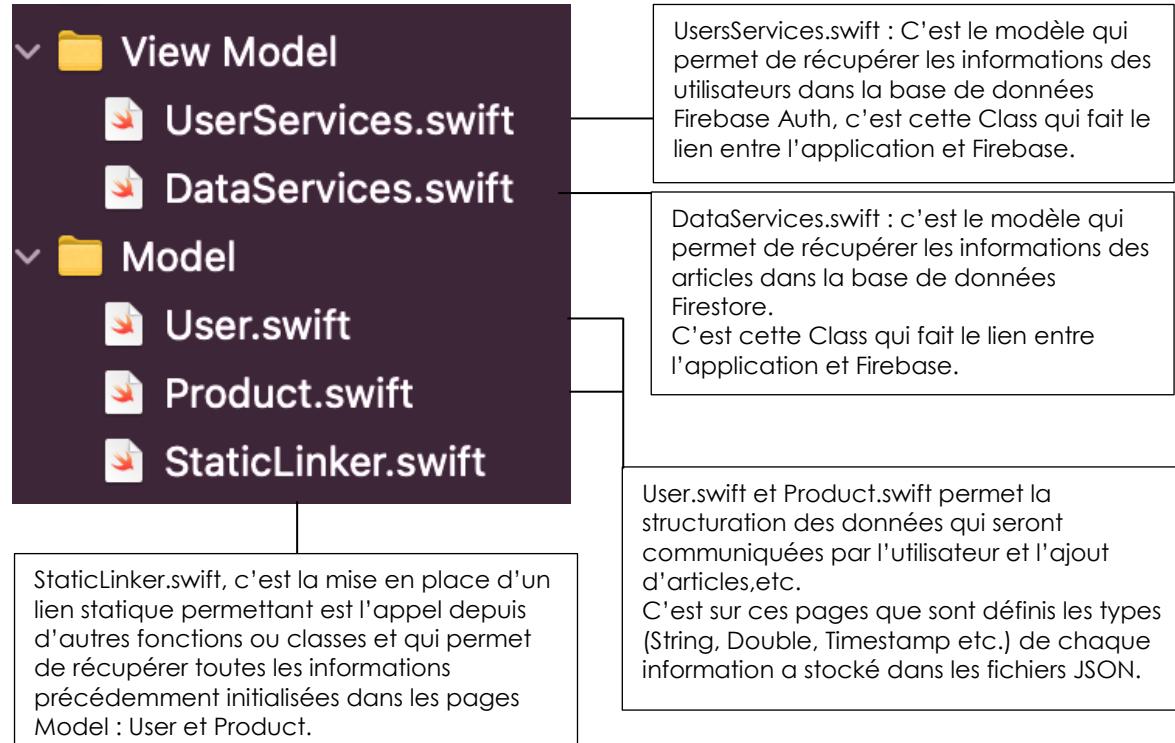
Figure 14 Firebase Firestore : Explication du stockage orienté "documents"

3.5.5 Architecture du programme : Modèle-Vue-Contrôleur

Pour la réalisation de ce projet, j'ai utilisé l'architecture MVC qui consiste à diviser le développement d'une application en trois types de modules différents.

Les **Modèles** contiennent les données à afficher :

Figure 15 Architecture - Application : Modèle



Les **Vues** contiennent la présentation de l'interface graphique, c'est ce que l'utilisateur verra lors de l'utilisation de l'application.

Ces différentes vues font partie de la « Main.storyboard » qui correspond à une page contenant tous les écrans de l'application, la gestion de la navigation entre les pages.

Il y a également la « LaunchScreen.storyboard » qui correspond à l'écran de chargement affiché à l'ouverture de l'application.

Et un dossier « Cells » contenant les éléments graphiques contenus dans une case d'article, elle est utilisée comme template d'affichage.

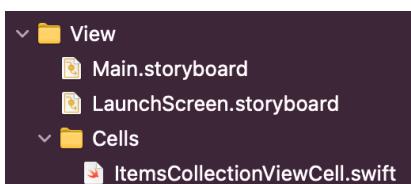
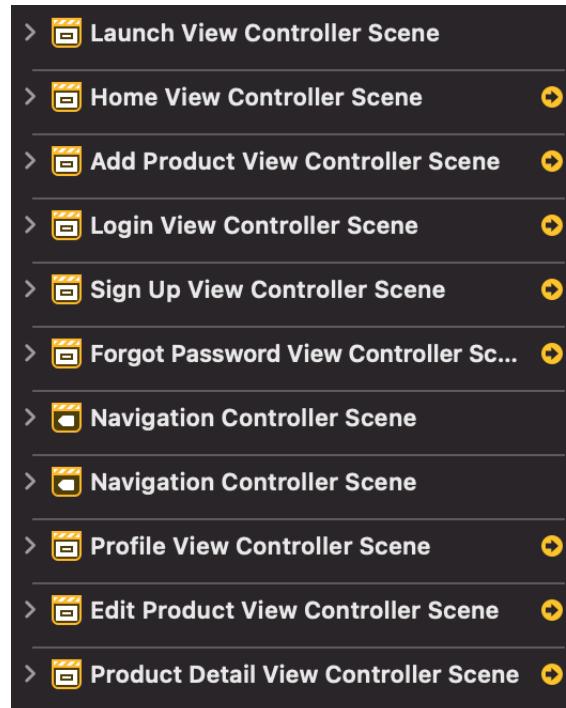


Figure 16 Architecture – Application : Vue - Dossiers

Figure 17 Architecture - Application : Vues de l'application



Les **Contrôleurs** contiennent la logique concernant les actions effectuées par l'utilisateur.

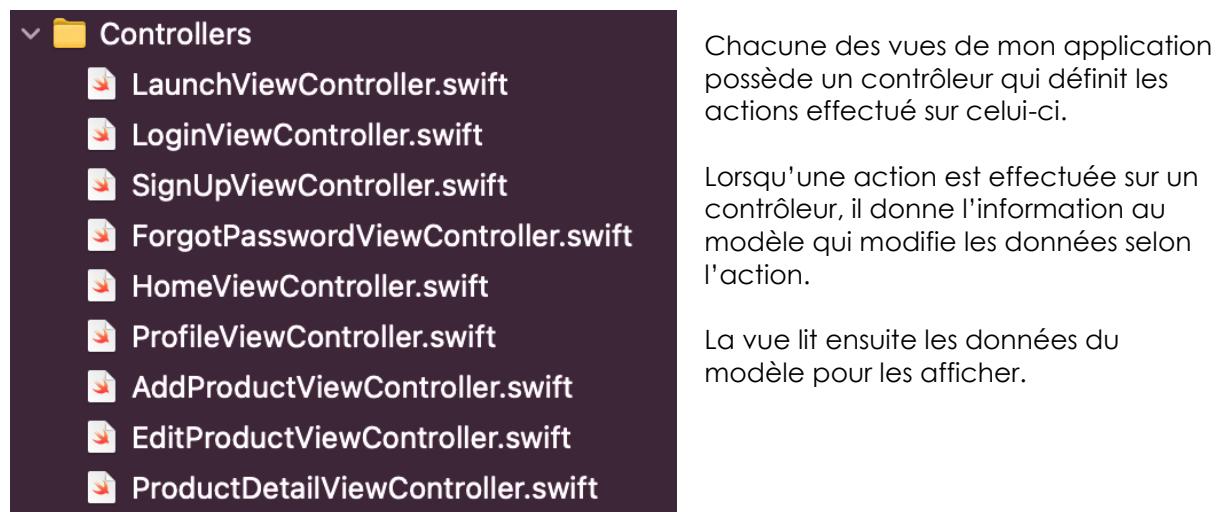


Figure 18 Architecture - Application : Contrôleurs - Dossier

3.5.6 CocoaPods – Gestionnaire de dépendances

C'est un gestionnaire de dépendances au niveau de l'application pour Objective-C, Swift et tout autre langage s'exécutant sur le moteur d'exécution d'Objective-C.

Il permet de simplifier l'importation de différentes bibliothèques externes (frameworks), permettant d'ajouter des fonctionnalités ou des éléments à une application, et permet de mettre à jour simultanément l'ensemble des bibliothèques installés avec la commande suivante : « **pod update** »

Pods installées	Description
Firebase ³	Importation du SDK Firebase
FirebaseAuth	Importation de la librairie concernant l'authentification Firebase
Firebase Core	Importation de la librairie concernant le noyau de Firebase
Firebase Firestore	Importation de la librairie concernant le service Firestore.
Firebase Storage	Importation de la librairie concernant le stockage Firebase.
IQKeyboardManagerSwift ⁴	Importation de la librairie universelle permettant d'éviter les problèmes d'ouverture du clavier, et de superposition avec des éléments de l'application.

³ <https://github.com/firebase/firebase-ios-sdk> - Répertoire git de la librairie Firebase

⁴ <https://github.com/hackiftekhar/IQKeyboardManager> - Répertoire git de la librairie.

MBProgressHUD – 1.2.0 ⁵	Importation de la librairie permettant d'ajouter à une application un indicateur de progression. Une solution native à Swift déjà existante UIProgressHUD, mais elle est mal documentée.
SDWebImage – 5.0 ⁶	Importation de la librairie permettant le téléchargement d'images asynchrone avec une prise en charge du cache en tant que catégorie UIImageView.

3.5.7 Cas d'utilisation

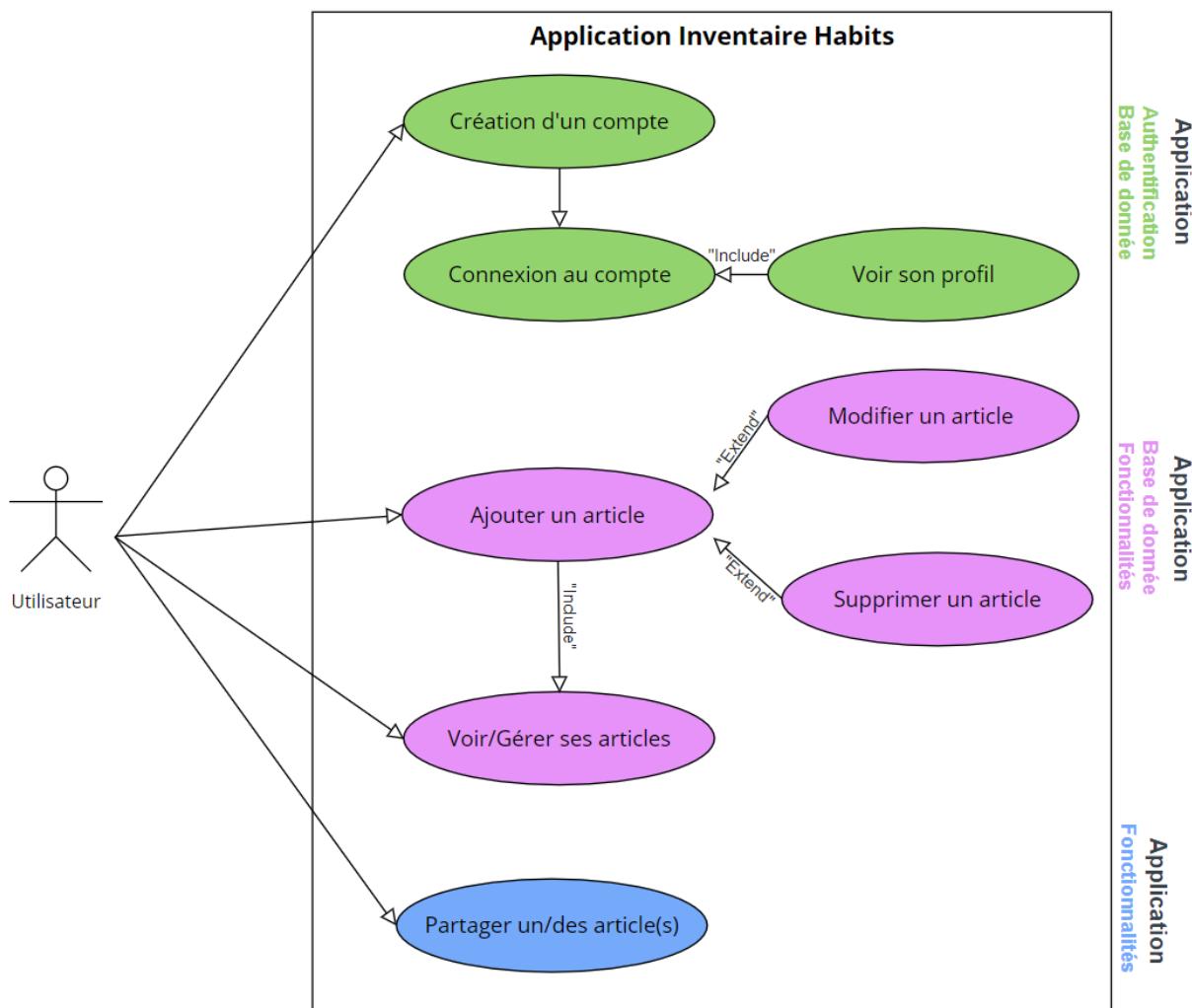


Figure 19 Maquette : Cas d'utilisation de l'application

⁵ <https://github.com/matej/MBProgressHUD> - Répertoire git de la librairie MBProgressHUD

⁶ <https://github.com/SDWebImage/SDWebImage> - Répertoire git de la librairie SDWebImage

3.5.8 Modèle Logique de Données

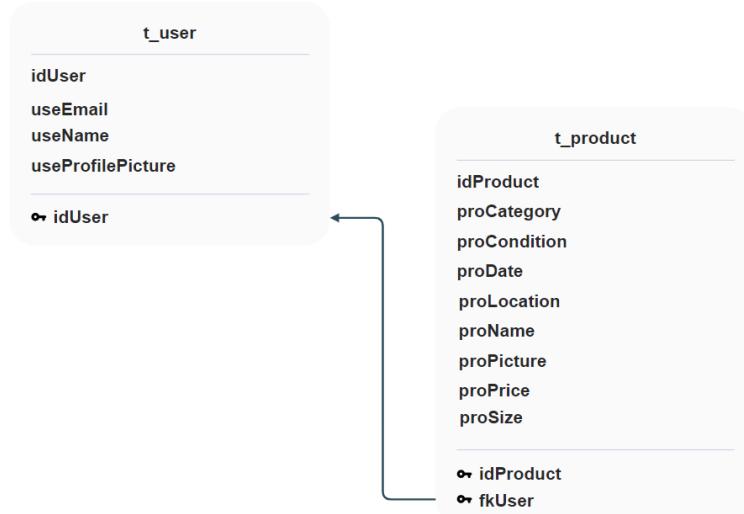


Figure 20 Maquette : Modèle logique des données

<table border="1"> <thead> <tr> <th colspan="2">t_user</th> </tr> </thead> <tbody> <tr> <td>Type String :</td><td>idUser</td></tr> <tr> <td>Type String :</td><td>useEmail</td></tr> <tr> <td>Type String :</td><td>useName</td></tr> <tr> <td>Type String :</td><td>useProfilePicture</td></tr> <tr> <td colspan="2">• idUser</td></tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="2">t_product</th> </tr> </thead> <tbody> <tr> <td>Type String :</td><td>idProduct</td></tr> <tr> <td>Type String :</td><td>proCategory</td></tr> <tr> <td>Type String :</td><td>proCondition</td></tr> <tr> <td>Type timestamp :</td><td>proDate</td></tr> <tr> <td>Type String :</td><td>proLocation</td></tr> <tr> <td>Type String :</td><td>proName</td></tr> <tr> <td>Type String :</td><td>proPicture</td></tr> <tr> <td>Type number :</td><td>proPrice</td></tr> <tr> <td>Type String :</td><td>proSize</td></tr> <tr> <td colspan="2">• idProduct</td></tr> <tr> <td colspan="2">• fkUser</td></tr> </tbody> </table>	t_user		Type String :	idUser	Type String :	useEmail	Type String :	useName	Type String :	useProfilePicture	• idUser		t_product		Type String :	idProduct	Type String :	proCategory	Type String :	proCondition	Type timestamp :	proDate	Type String :	proLocation	Type String :	proName	Type String :	proPicture	Type number :	proPrice	Type String :	proSize	• idProduct		• fkUser		<ul style="list-style-type: none"> — idUser : Identifiant de l'utilisateur dans la table — useEmail : enregistrement du mail de l'utilisateur - — useName : enregistrement du nom de l'utilisateur — useProfilePicture : enregistrement de l'image de profil de l'utilisateur - String — idUser : Clé primaire de la table t_user — idProduct : Identifiant du produit (Article) dans la table — proCategory : enregistrement de la catégorie de l'article (T-shirt, etc.) — proCondition : enregistrement de la condition de l'article (10/10 Neuf, 5/10 Usée etc.) — proDate : enregistrement de la date d'achat de l'article — proLocation: enregistrement du lieu d'achat de l'article — proName: enregistrement du nom de l'article (Hoodie Dior « CD », Vest Louis Vuitton) — proPicture: enregistrement de l'image de l'article — proPrice: enregistrement du prix de l'article — proSize: enregistrement de la taille de l'article (XS, S, M etc.) — idProduct: Clé primaire de la table t_product — fkUser: Clé étrangère de la table t_user
t_user																																					
Type String :	idUser																																				
Type String :	useEmail																																				
Type String :	useName																																				
Type String :	useProfilePicture																																				
• idUser																																					
t_product																																					
Type String :	idProduct																																				
Type String :	proCategory																																				
Type String :	proCondition																																				
Type timestamp :	proDate																																				
Type String :	proLocation																																				
Type String :	proName																																				
Type String :	proPicture																																				
Type number :	proPrice																																				
Type String :	proSize																																				
• idProduct																																					
• fkUser																																					

4 RÉALISATION

4.1 Liste des fichiers : Répertoire

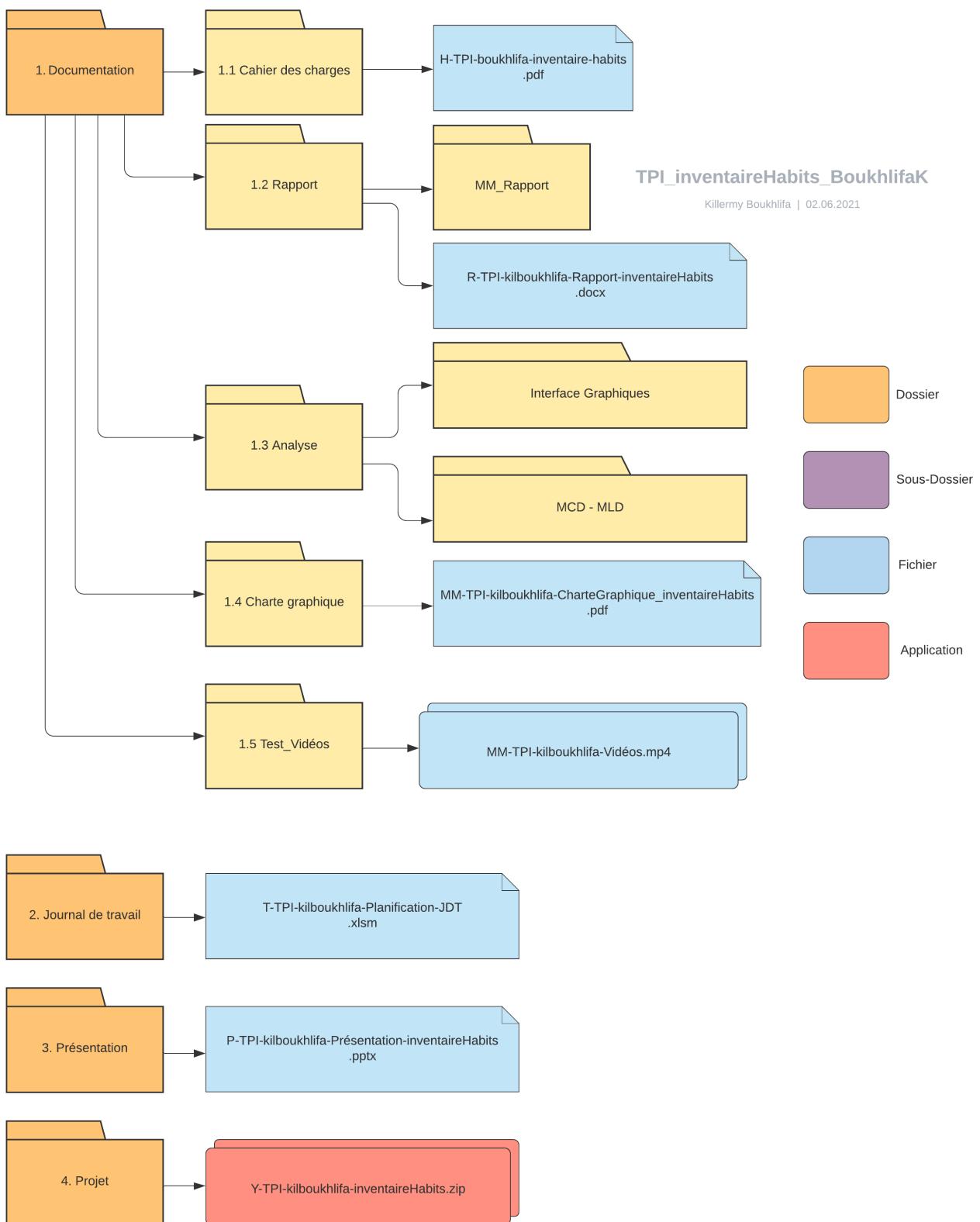


Figure 21 Répertoire : Projet

4.2 Liste des fichiers : Code source

✓  Inventaire Habits	« Inventaire Habits » contenant l'ensemble des fichiers de l'application.
✓  Inventaire Habits	Contient la configuration de l'application.
 AppDelegate.swift	Est responsable de l'affichage de l'application.
 SceneDelegate.swift	« Utilities » contient des utilitaires pour l'application.
✓  Utilities	Contient le design de certains éléments.
 Designables.swift	Gère la configuration du date time picker et des UIViewController.
 Extensions.swift	« ViewModel » contient les liens avec la base de données Firebase.
✓  View Model	Récupère les informations de la table t_user.
 UserServices.swift	Récupère les informations de la table t_product.
 DataServices.swift	« Model » contient les liens avec la base de données Firebase.
✓  Model	Structure les données de l'utilisateur (définit les types etc.)
 User.swift	Structure les données de les articles (définit les types etc.)
 Product.swift	Établie le lien statique entre les classes
 StaticLinker.swift	« Controllers » contient la gestion des actions effectuées par l'utilisateur.
✓  Controllers	Contrôleur de la page de lancement.
 LaunchViewController.swift	Contrôleur de la page de connexion.
 LoginViewController.swift	Contrôleur de la page de création d'un compte.
 SignUpViewController.swift	Contrôleur de la page mot de passe oublié.
 ForgotPasswordViewController.swift	Contrôleur de la page principale.
 HomeViewController.swift	Contrôleur de la page profil de l'utilisateur.
 ProfileViewController.swift	Contrôleur de la page ajout d'un article.
 AddProductViewController.swift	Contrôleur de la page modification d'un article.
 EditProductViewController.swift	Contrôleur de la page détail d'un article.
 ProductDetailViewController.swift	« View » contient les différentes vues de l'application.
✓  View	Contient l'ensemble des pages de l'application.
 Main.storyboard	Contient l'affichage de la page de lancement.
 LaunchScreen.storyboard	« Cells » contient le template des cases d'articles.
>  Cells	Contient les images/icônes de l'application.
 Assets.xcassets	Contient les informations de l'application.
 Info.plist	Contient les propriétés du projet lié à Firebase.
 GoogleService-Info.plist	« Products » contient l'application.
✓  Products	L'application exécutable.
 Inventaire Habits.app	« Pods » contient les pods installés dans le projet.
>  Pods	

Figure 22 Répertoire : Application

4.3 Firebase : Authentification

Pour commencer l'application, j'ai mis en place un système d'authentification, il m'a fallu premièrement lier mon application avec la base de données Firebase et récupérer les informations d'un utilisateur qui se connecte avec la fonction **sign In**.

J'ai donc importé les modules liés à Firebase, ensuite, j'ai défini la fonction **SignIn**, en initialisant que c'était une connexion par e-mail et que les informations saisies doivent être un e-mail et un mot de passe.

Enfin, on lit la collection « t-user » lié à l'authentification, on sélectionne l'user actuel en comparant avec les informations présente dans la base de données.

S'il y a un problème de connexion, le message d'erreur « Impossible de se connecter » apparaît.

```
import Firebase
import FirebaseAuth
import FirebaseFirestore

class UserServices{

    private let db = Firestore.firestore()
    static let shared = UserServices()

    private init(){}

    func signIn(email:String, password:String, completion: @escaping (_ error: String?) -> ()) {
        Auth.auth().signIn(withEmail: email, password: password, completion: {(user,error) in
            if let error = error{
                completion(error.localizedDescription)
            }else{
                if let error = error{
                    completion(error.localizedDescription)
                }else{
                    //Getting the user object from firebase and saving it to the static global variable in the
                    //static linker class
                    self.db.collection("t_user").document(Auth.auth().currentUser!.uid).getDocument { (snap, error) in
                        if let data = snap?.data(), let uid = snap?.documentID{
                            let user = t_user(data: data, uid: uid)
                            StaticLinker.user = user
                            completion(nil)
                        }else{
                            completion("Une erreur est survenue. Impossible de vous connecter")
                        }
                    }
                }
            }
        })
    }
}
```

Figure 23 View Model : UserServices.swift

Classe objet de l'utilisateur, instancier avec les attributs correspondant à ceux de la base de données propre à l'id User connecté. Cela permet le stockage en format JSON.

```
//Structuring of the information stored for a user in order to be able to directly call t_user thereafter to allow the management of json files,
//Initialization of the types of fields required for storage.
struct t_user{

    var idUser = ""
    var userName: String
    var userEmail: String
    var useProfilePicture: String

    var json: [String: Any] {
        return [
            "userName": self.userName,
            "useEmail": self.userEmail,
            "useProfilePicture": self.useProfilePicture
        ]
    }
    //Initializing a user's type
    init(userName: String, userEmail: String, useProfilePicture: String){
        self.userName = userName
        self.userEmail = userEmail
        self.useProfilePicture = useProfilePicture
    }
    //Initializing a user's data
    init(data: [String: Any], uid: String){
        self.idUser = uid
        self.userName = data["userName"] as! String
        self.userEmail = data["useEmail"] as! String
        self.useProfilePicture = data["useProfilePicture"] as! String
    }
}
```

Figure 24 Model : User.swift

Fonction **signUP** de la page, initialise les champs :**email**, **password** et **name** en texte.

Si l'e-mail et le mot de passe contiennent des informations de connexion correctes, alors j'utilise le framework : **MBProgressHUD** pour indiquer à l'utilisateur que l'application vérifie les informations saisies, le compte est ensuite créé et l'utilisateur est redirigé sur la page principale de l'application en étant connecté automatiquement.

```
//Action when the user presses the login button, displays a loading spinner, retrieves the information entered and registers the user in the database by first checking for potential errors.
@IBAction func signUp(_ sender: Any) {
    if let email = self.email.text, let password = self.password.text, let name = self.name.text{
        if email != "" && password != ""){
            MBProgressHUD.showAdded(to: self.view, animated: true)
            UserServices.shared.register(user: t_user(userName: name, userEmail: email, userProfilePicture: ""), email: email, password: password,
                image: self.image){ (error) in
                    MBProgressHUD.hide(for: self.view, animated: true)
                    if let error = error{
                        self.showAlert(title: "Erreur", message: error) { (_) in }
                    }else{
                        self.showAlert(title: "Succès", message: "Votre compte a été créé") { (_) in
                            self.performSegue(withIdentifier: "home", sender: nil)
                        }
                    }
                }
            //Error message if all fields are not filled
        }else{
            self.showAlert(title: "Erreur", message: "Veuillez remplir tout les champs" ) { (_) in }
        }
    }
}///End of signUp function
```

Figure 25 SignUpViewController.swift

4.4 Firebase : Firestore

```
1 //Initializing a article's type
init(proName: String, proLocation: String, proCategory: String, proSize: String,
    proDate: Date, proPrice: Double, proCondition: String, proPicture: String){
    self.fkUser = StaticLinker.user.idUser
    self.proName = proName
    self.proLocation = proLocation
    self.proCategory = proCategory
    self.proSize = proSize
    self.proDate = Timestamp(date: proDate)
    self.proPrice = proPrice
    self.proCondition = proCondition
    self.proPicture = proPicture
}
//Initializing a article's type for the JSON Files
init(data: [String: Any], id: String){
    self.idProduct = id
    self.fkUser = data["fkUser"] as! String
    self.proName = data["proName"] as! String
    self.proLocation = data["proLocation"] as! String
    self.proCategory = data["proCategory"] as! String
    self.proSize = data["proSize"] as! String
    self.proDate = data["proDate"] as! Timestamp
    self.proPrice = data["proPrice"] as! Double
    self.proCondition = data["proCondition"] as! String
    self.proPicture = data["proPicture"] as! String
}
```

Figure 26 Model : Product.swift

```
2 //Function for deleting an article, Retrieving the article directory
func deleteProduct(pid: String, completion: @escaping (_ error: String?) -> ()) {
    self.db.collection("t_product").document(pid).delete(completion: { (error) in
        if let error = error{
            completion(error.localizedDescription)
        }else{
            completion(nil)
        }
    })
}///End of deleteProduct func

//Function for retrieving information about an article according to user, article id etc.
func getProducts(completion: @escaping (_ error: String?, _ products: [t_product]?) -> ()) {
    self.db.collection("t_product").whereField("fkUser", isEqualTo: StaticLinker.user.idUser).addSnapshotListener { (snap, error) in
        if let error = error{
            completion(error.localizedDescription, nil)
        }else if let snap = snap{
            var products = [t_product]()
            for i in snap.documents{
                products.append(t_product(data: i.data(), id: i.documentID))
            }
            completion(nil, products)
        }else{
            completion(nil, nil)
        }
    }
}///End of getProducts func
```

Figure 27 ViewModel : DataServices.swift

4.5 Créations des cellules d'article

Lors de la conception des cases d'articles, j'ai rencontré quelques difficultés, étant donné que je ne connaissais pas parfaitement les possibilités offertes par l'outil de développement XCode et l'interface Builder, j'ai eu de mal à concevoir comment j'allais mis prendre afin d'afficher des cases, qui s'ordonneront seules lors d'ajouts de nouveaux articles.

C'est à ce moment que j'ai découvert l'objet : UICollectionView⁷ c'est un objet qui permet de gérer une collection ordonnée d'éléments de données et les présente à l'aide de mises en pages personnalisables.

J'ai donc créé une cellule type « cell s » contenant une image, une image d'icône arrondie et un label.

```
class ItemsCollectionViewCell: UICollectionViewCell {
    //Initialization of the elements to display in the boxes containing the articles
    @IBOutlet weak var selectImg: UIImageView!
    @IBOutlet weak var img: RoundableImageView!
    @IBOutlet weak var lbl: UILabel!
}
```

Figure 27 Controller : Cells - Class ItemsCollectionViewCell

Désormais, la cellule type est créée, j'ai ensuite appelé cette cellule dans l'UICollectionView de cette manière :

```
func collectionView(_ collectionView: UICollectionView, numberOfItemsInSection section: Int) -> Int {
    return self.items.count
}
//Setting up the collectionview cells
func collectionView(_ collectionView: UICollectionView, cellForItemAt indexPath: IndexPath) -> UICollectionViewCell {
    let cell = collectionView.dequeueReusableCell(withIdentifier: "itemCell", for: indexPath) as! ItemsCollectionViewCell
    cell.selectImg.isHidden = !collectionView.allowsMultipleSelection
    if self.selectedIndex.contains(indexPath.row){
        cell.selectImg.tintColor = .green
    }else{
        cell.selectImg.tintColor = .purple
    }
    let text = "\((self.items[indexPath.row].proName)\nCHF \((self.items[indexPath.row].proPrice.rounded(toPlaces: 2))\""
    cell.lbl.text = text
    if self.items[indexPath.row].proPicture == ""{
        cell.img.image = UIImage(named: "image")
    }else{
        cell.img.sd_setImage(with: URL(string: self.items[indexPath.row].proPicture), completed: nil)
    }
    return cell
}
```

Figure 28 Controller : HomeViewController.swift - Func collectionView

J'ai défini dans un premier temps une fonction permettant de compter le nombre « d'items » présents dans la collection view, j'ai ensuite initialisé les données à remplir dans les cellules en utilisant l'index des items :
 indexPath.row -> proPicture, proName, proPrice
 (arrondie à deux unités après la virgule.)

Et j'ai géré la gestion de la sélection multiple, lorsqu'un utilisateur appuie sur le bouton « Sélectionné » les cercles de sélections apparaissent sur les cellules, et j'utilise « selectedIndex » afin de savoir si l'article est sélectionné ou non, la couleur du cercle de sélection change également.

Voici une cellule d'exemple contenant un article avec une photo, un nom et un prix. Elle a été sélectionnée pour un partage, c'est indiquer par le cercle vert.



Figure 29 Application : Cellule exemple

⁷ <https://developer.apple.com/documentation/uikit/uicollectionview>

4.6 Gestion des catégories

Ce fut une fonctionnalité relativement simple à implémenter, cependant je l'ai trouvé intéressante.

En premier lieu, il m'a fallu intégrer le framework « DropDownMenu », j'ai ensuite dû initialiser les différents éléments contenus dans le menu déroulant.

dataSource = [« All », « Hoodie » etc.], je définis ensuite un index en fonction de la catégorie que j'attribue à selectedCategory :

```
private func setup(){
    self.colv.delegate = self
    self.colv.dataSource = self
    self.colv.reloadData()
    //Setting up the drop down list
    self.menuBtn.setTitle(" All ", for: .normal)
    self.dropDown.anchorView = self.menuBtn
    self.dropDown.dataSource = ["All", "Hoodie", "Tee", "Pant", "Accesories", "Jacket", "Shoes"]
    self.dropDown.selectionAction = { [unowned self] (index: Int, item: String) in
        self.menuBtn.setTitle(" \(item) ", for: .normal)
        self.selectedCategory = index
        self.sortFilterData(field: self.dropDown.dataSource[self.selectedCategory])
    }
    self.getData()
}
```

Figure 30 Controller : HomeViewController.swift - Func setup

Ensuite dans la fonction sortFilterData, je définis que si la catégorie sélectionnée est « All » alors je récupère tous les articles de l'utilisateur pour les afficher dans l'ordre d'ajout sur l'application en rechargeant les données.

Et si, une autre catégorie est sélectionnée alors je récupère les éléments de la catégorie que je trie également dans l'ordre d'ajout en rechargeant les données.

```
//Sorting data according to the filter value selected
private func sortFilterData(field: String){
    if field == "All"{
        self.items = self.allItems
        self.items.sort(by: {$0.proDate.dateValue() < $1.proDate.dateValue()})
        self.colv.reloadData()
    }else{
        self.items = self.allItems.filter({$0.proCategory == field})
        self.items.sort(by: {$0.proDate.dateValue() < $1.proDate.dateValue()})
        self.colv.reloadData()
    }
}
```

Figure 31 Controller : HomeViewController.swift - Func sortFilterData

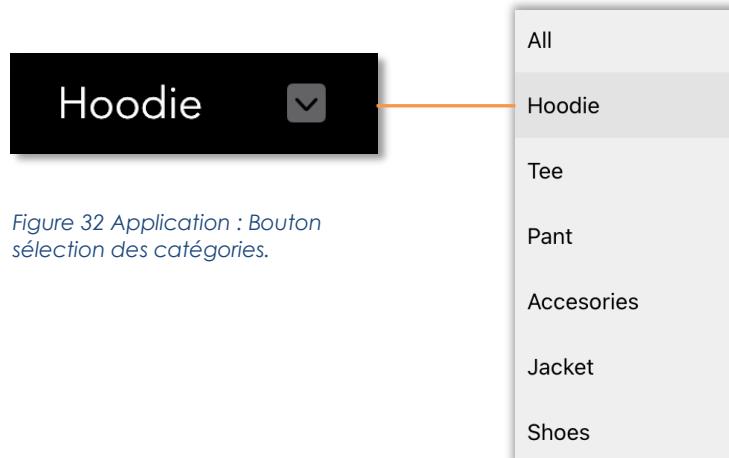


Figure 32 Application : Bouton sélection des catégories.

Figure 33 Application : Menu déroulant des catégories.

4.7 Versions du projet

Les versions du projet sont disponibles sur le Git du projet, en consultant l'historique des push, il est possible de retourner à toutes les versions existantes du projet et consulter les différentes modifications effectuées.

4.8 Base de données : Firebase

Collection t_user :

inventairehabits	t_user	c8d000UW70NzpcEGNXLUyV6Z3UH3
+ Commencer une collection	+ Ajouter un document	+ Commencer une collection
t_product	8kT1AbEly7NU3FPKW5iAX03EEciw2 9pMS0fQ16UcDisxr85ye16jnitu2 c8d000UW70NzpcEGNXLUyV6Z3UH3	+ Ajouter un champ
t_user		useEmail: "Killermym1@hotmail.fr" useName: "Killermym Boukhilifa" useProfilePicture: "https://firebasestorage.googleapis.com/v0/b/inventairehal... alt=media&token=0383be14-71da-4d77-bbfa-aaed10b314c"

Figure 34 Firebase : Collection t_user

Collection t_product :

inventairehabits	t_product		0GpsOuF7jbRhbekT9yYh	
+ Commencer une collection	+ Ajouter un document		+ Commencer une collection	
t_product	0GpsOuF7jbRhbekT9yYh	>	+ Ajouter un champ	
t_user	0aPpbGv6f1nqF8e141yc		fkUser: "c8d00OUW70NzpcEGNXLUyV6Z3UH3"	
	1CVxqjHh2QZRip1KCcrG		proCategory: "Shoes"	
	3sbY5MKRVZQl58gB5rNL		proCondition: "9/10"	
	4rz0lkGlsWhPBjE0pdsb		proDate: 12 novembre 2020 à 20:42:00 UTC+1	
	7EPca9Ez4TAadppxtAVyc		proLocation: "Gucci"	
	DJNHeY1lEkE8t0LiiCp0		proName: "Gucci flashtrek"	
	F5vPASCYS7hwV16kFvg		proPicture: "https://firebasestorage.googleapis.com/v0/b/inventairehabits.appspot.com/.../0GpsOuF7jbRhbekT9yYh.jpg?alt=media&token=23df1025-9892-4d0e-be4a-f9ca14b57dc6"	
	FHm60cn4XrqGrA6xmbXD		proPrice: 1300	
	GCC2gkKy5Rwkm6VCcgNh		proSize: "43"	
	HigKGEIwRHV1myJ0zJK			

Figure 35 Firebase : Collection t product

4.9 Charte graphiques : Application

J'ai réalisé une charte graphique de l'application, elle se trouve dans le dossier : 1.4 Charte_graphique -> [MM-tpi-boukhlifa-CharteGraphique_inventaireHabits.pdf](#)

4.10 Description des tests effectués

Ces tests seront effectués afin de vérifier que le projet fonctionne correctement.
Les tests seront notés de la manière suivante :

- Test validé



- Test partiellement validé



- Test non validé



Description	Conséquence	Résultat
Page connexion - Bouton mot de passe oublié et créer un compte	Les boutons « mot de passe oublié » et « créer un compte » redirige correctement vers leurs pages respectives. Voir : 6.3 Dossier des test	
- Connexion d'un utilisateur	Lorsqu'un utilisateur, saisie les informations de connexion d'un compte existant, il est correctement redirigé vers la page principale de l'application propre à son compte. Voir : 6.3 Dossier des test	
- Erreur indiquée lors de la saisie des informations de connexion.	Si l'utilisateur saisi des informations de connexion erroné, un pop-up lui indiquant apparaît. Voir : 6.3 Dossier des test	
Page création d'un compte - Ajout d'une image de profil	Lorsque l'utilisateur sélectionne une image dans sa pellicule ou prend une photo, elle est ajoutée comme image de profil. Voir : 6.3 Dossier des test	
- Saisies des informations enregistrées et création d'un compte possible.	La création d'un compte est possible, et les informations sont correctement sauvegardées. Voir : 6.3 Dossier des test	
- Bouton retour	Le bouton retour redirige sur la page « Connexion » Voir : 6.3 Dossier des test	

Page mot de passe oublié	<ul style="list-style-type: none"> - Saisi de l'e-mail <p>L'application vérifie qu'un e-mail est saisi et s'il correspond à un e-mail existant alors un mail est envoyé à l'utilisateur.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Possibilité de modifier son mot de passe 	<p>Lorsque l'utilisateur reçoit l'e-mail, il a la possibilité de modifier son mot de passe.</p> <p>Voir : 6.3 Dossier des test</p>	
Page principale	<ul style="list-style-type: none"> - Bouton profil / Affichage de la photo de profil <p>Le bouton profil contient la photo de profil et il redirige l'utilisateur sur la page profil.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Liste des articles affichés 	<p>Sur la page principale, les articles de l'utilisateur sont affichés lors de la connexion.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Articles cliquables pour accéder à la page détaillée de l'article 	<p>Lorsqu'un utilisateur clique sur un article, il est redirigé vers la page détails de l'article correspondant.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Bouton d'ajout d'un article 	<p>Le bouton « + » en bas de l'écran redirige vers la page d'ajout d'un article.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Bouton sélection d'un article - Possibilité de partager un article 	<p>Lorsque l'utilisateur appuie sur le bouton « Sélectionné », la possibilité de sélectionner différents articles (bouton radio sur la case) lui permet ensuite de partager les informations des articles sous forme de textes.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Possibilité de trier les articles par catégorie 	<p>Un menu déroulant est activable sur la page principale, il permet de sélectionner les différentes « catégories » d'articles.</p> <p>Voir : 6.3 Dossier des test</p>	
Page détails d'un article	<ul style="list-style-type: none"> - Bouton retour <p>Le bouton « retour » redirige correctement l'utilisateur sur la page principale.</p> <p>Voir : 6.3 Dossier des test</p>	

<ul style="list-style-type: none"> - Image de l'article - Nom de l'article - Lieu d'achat - Catégorie de l'article - Taille - Condition de l'article - Date d'achat - Prix d'achat 	<p>La page détails d'un article contient toutes les informations prévues.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Bouton modification de l'article 	<p>La page détails d'un article contient un bouton pour modifier un article, et le bouton redirige l'utilisateur sur la page modification d'un article.</p> <p>Voir : 6.3 Dossier des test</p>	
<p>Page ajout d'un article</p> <p>Champs d'informations :</p> <ul style="list-style-type: none"> - Image de l'article - Nom de l'article - Lieu d'achat - Catégorie de l'article - Taille - Condition de l'article - Date d'achat - Prix d'achat 	<p>La page ajouts d'un article permet d'ajouter un article à la base de données comprenant toutes les informations prévues.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Bouton ajout 	<p>Le bouton « Validation » ajoute correctement l'article à la base de données, avec toutes les informations précédemment saisies par l'utilisateur.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Bouton retour 	<p>Le bouton « retour » redirige correctement l'utilisateur sur la page principale.</p> <p>Voir : 6.3 Dossier des test</p>	
<p>Page modification d'un article</p> <p>Champs d'informations modifiables :</p> <ul style="list-style-type: none"> - Image de l'article - Nom de l'article - Lieu d'achat - Catégorie de l'article - Taille - Condition de l'article - Date d'achat - Prix d'achat 	<p>La page de modification d'un article permet de modifier toutes les informations de l'article et les réenregistre sur la base de données.</p> <p>Voir : 6.3 Dossier des test</p>	

<ul style="list-style-type: none"> - Bouton enregistré les modifications 	<p>Le bouton « Enregistrer » enregistre les informations modifiées de l'article et les réenregistre dans la base de données.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Bouton suppression de l'article 	<p>Le bouton « Supprimer » supprime l'article de la base de données via son ID.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Bouton retour 	<p>Le bouton « retour » redirige correctement l'utilisateur sur la page détails d'un article.</p> <p>Voir : 6.3 Dossier des test</p>	
Page profil		
<ul style="list-style-type: none"> - Affichage des informations de l'utilisateur 	<p>La page profil de l'application, affiche correctement la photo de profil correspondante à l'utilisateur, ainsi que son nom et son adresse e-mail.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Modification de l'image de profil de l'utilisateur 	<p>La photo de profil de l'utilisateur est modifiable et s'enregistre dans la base de données.</p> <p>Voir : 6.3 Dossier des test</p>	
<ul style="list-style-type: none"> - Statistiques détaillés des articles disponibles dans la collection de l'utilisateur 	<p>Cette fonctionnalité était une fonctionnalité supplémentaire, Je n'ai pas pu les afficher sous le format que j'aurais souhaiter (Graphiques), cependant j'ai ajouté la valeur totale de la collection de l'utilisateur ainsi que le nombre d'article qu'il possède.</p> <p>Voir : 6.3 Dossier des test</p>	

4.11 Erreurs restantes

De mon point de vue, il existe une erreur qui n'entraîne pas de problème en rapport avec les fonctionnalités, simplement un problème d'affichage, j'ai utilisé un framework : « `DropdownMenu` » cependant je n'ai pas eu l'opportunité de me documenter sur la limitation de taille des menus déroulants.

Le problème se situe sur la page d'ajout d'un article et celle de modification d'un article, ayant des champs avec de nombreuses possibilités comme la taille des chaussures, le menu apparaît et prend une place conséquente sur ces pages :

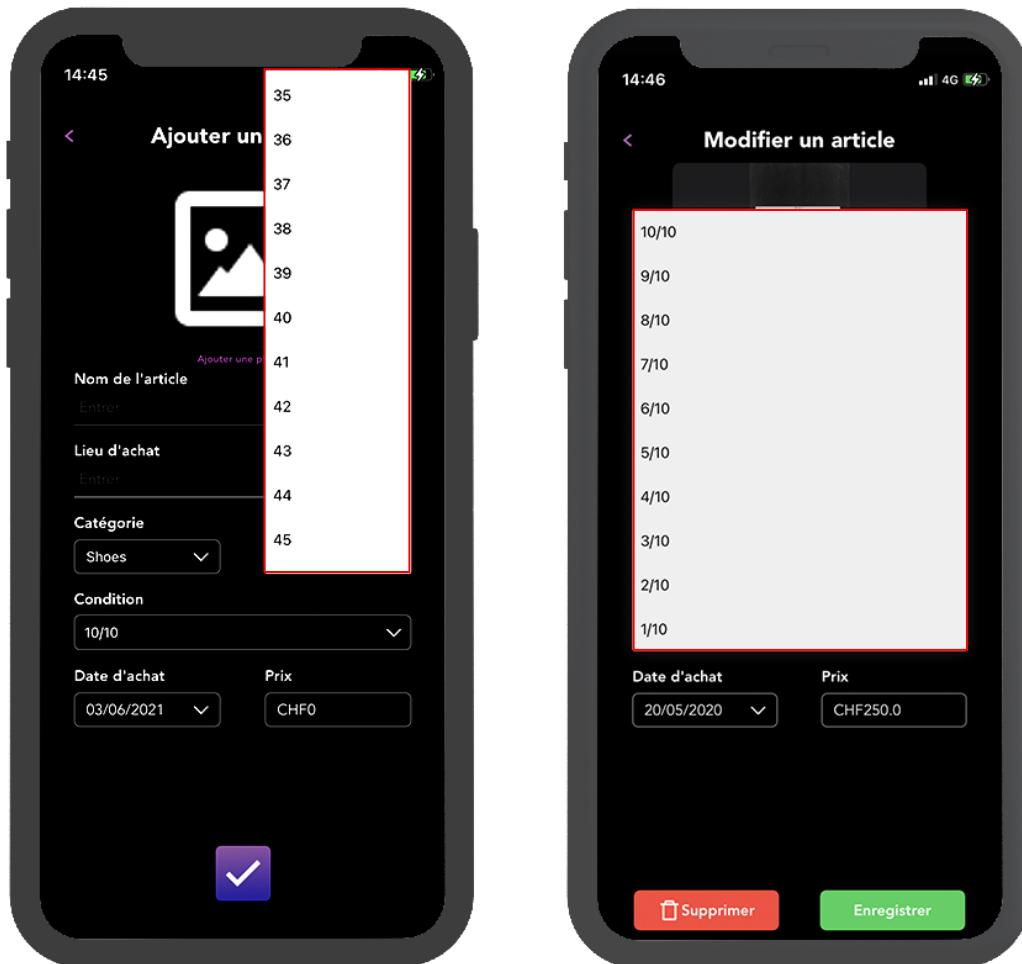


Figure 36 application : Menu déroulant - Erreur d'affichage

Les menus incluent la possibilité d'être défilés de bas en haut, je pense qu'il me faudrait 1 demi-journée pour trouver un moyen de redimensionner les différents menus déroulants de l'application.

En attendant, les menus déroulants fonctionnent malgré ce problème d'affichage.

4.12 Liste des documents fournis

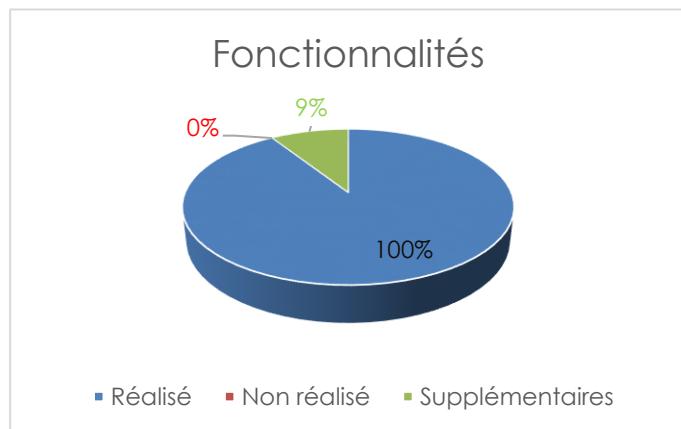
- Cahier des charges – 06.05.2021
- Planification initiale – 06.05.2021
- Journal de travail – 04.06.2021
- Maquettes de l'application – 10.05.2021
- Rapport du projet – 04.06.2021
- Abstract/résumé du projet – 04.06.2021
- Normes de codages ETML – V.3.5.0
- Images du projet – 04.06.2021
- Code source de l'application – V.6 – 04.06.2021
- Charte graphique de l'application – 02.06.2021
- Manuel d'utilisation de l'application – 04.06.2021

5 CONCLUSION

5.1 Bilan des fonctionnalités demandées

Concernant les fonctionnalités demandées, toutes les fonctionnalités présentes dans le cahier des charges ont été implémentées dans l'application.

Cependant dans un souci d'optimisation et d'amélioration, lorsque j'ai commencé le projet, je me suis rendu compte qu'implémenté un système de compte utilisateur rendrait mon travail plus simple et augmenterait la viabilité de l'application alors j'ai ajouté cette fonctionnalité supplémentaire.



J'envisage d'améliorer par la suite cette application, je pense qu'elles porteraient sur :

- L'ajout de différentes animations,lorsque l'utilisateur sélectionne un article, appuie sur un bouton, etc.
- Modélisation de boutons correspondant à la charte graphique.
- Ajouter un lien afin de retrouver un article sur Internet.
- Partage des articles avec leurs photos.

Ensuite, les statistiques, je pense qu'il serait possible de rajouter des informations comme le prix de reventes potentielles d'un article, ce qui permettrait ensuite de comparer l'investissement initial, le gain potentiel et la valeur actuelle de la collection des utilisateurs.

5.2 Bilan personnel

Ce projet m'a permis de visualiser ce que le monde professionnel attend d'un jeune informaticien, que ce soit grâce aux critères TPI ou aux dialogues avec mes experts et mon chef de projet.

J'ai rencontré quelques difficultés lors de la réalisation de ce projet, les cellules d'articles m'ont causé quelques problèmes et j'ai trouvé ça très intéressant d'approfondir jusqu'à réussir finalement à obtenir le rendu désiré.

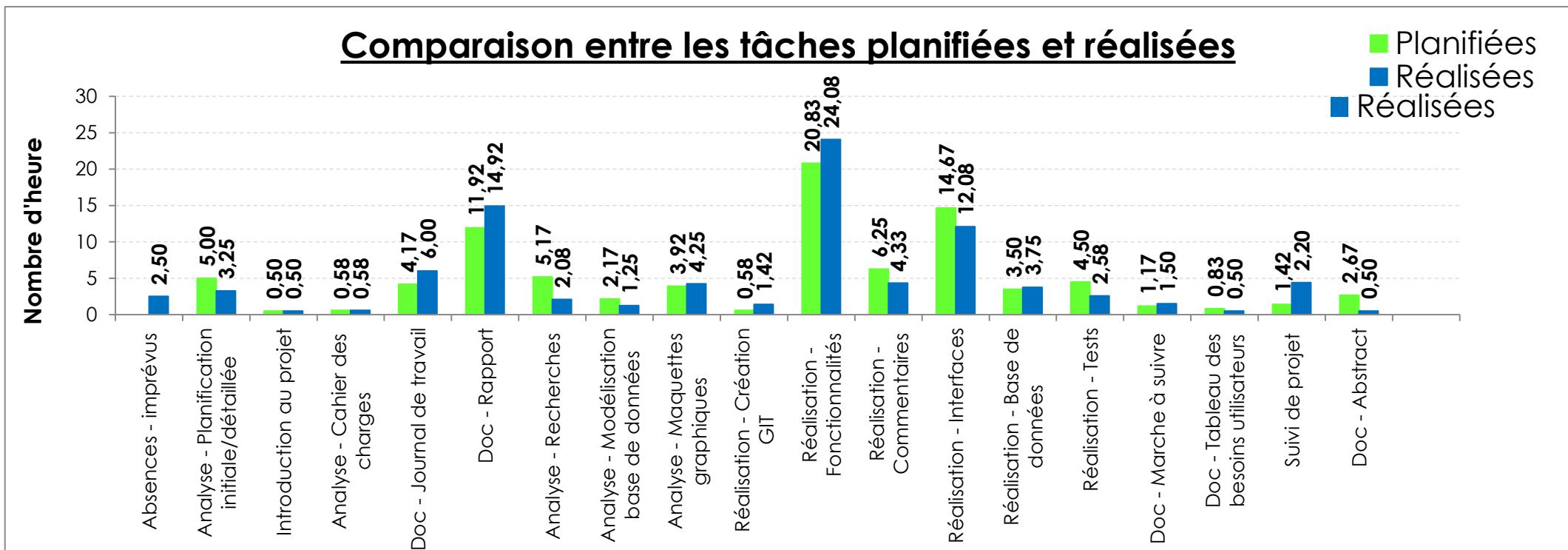
J'ai apprécié réaliser cette application, le sujet concernait l'une de mes passions ce qui constituait donc une motivation supplémentaire. Ce projet m'a permis d'améliorer ma méthode de travail de manière générale, et d'apprendre de nombreuses choses dans le développement d'applications IOS, mais aussi dans l'utilisation d'un répertoire Git.

Finalement, le point négatif que j'ai pu constater me concernant, c'était le remplissage du journal de travail quotidien et structuré, j'ai des difficultés pour retranscrire précisément ce que j'effectue dans la journée, mais ce projet m'a permis de travailler sur ce point.



5.1 Bilan de la planification

Lors de la réalisation de ce projet, il me semble que j'avais relativement bien planifié mon temps, si je devais le refaire, j'essayerai de m'organiser mieux quant à l'ordre de réalisation des tâches et je planifierai plus de temps pour l'interface afin d'améliorer l'intuitivité de l'application ainsi que son aspect visuel global.



6 ANNEXES

6.1 Résumé du projet Inventaire Habits

Situation de départ

Dans le cadre de ma formation d'informaticien, en vue d'obtenir un CFC, j'ai dû réaliser une application IOS en 90 h pour confirmer les compétences acquises durant mes 4 années d'études à ETML.

Ce projet consiste à créer une application mobile permettant à un utilisateur de gérer et d'inventorier sa garde-robe d'habits et chaussures.

L'utilisateur devra pouvoir ajouter/modifier / supprimer/partager un article, les articles seront affichés sous forme de collection triable par catégorie.

Mise en œuvre

Pour réaliser ce projet, j'ai utilisé la méthodologie de travail dit « En cascade ». J'ai commencé par diviser en étape ce projet afin de le planifier de manière optimale : l'analyse du cahier des charges, la planification initiale, la conception de l'application, des maquettes, j'ai choisi d'utiliser le service Firebase pour ma base de données de type No SQL, j'ai réalisé le MCD et le MLD de la base de données, j'ai étudié les besoins des utilisateurs et enfin j'ai établi une stratégie de test.

Concernant le suivi de mon projet, j'ai utilisé un répertoire Git via l'application GitHub Desktop ce qui m'a permis de mettre à disposition des experts des mises à jour régulières attestant de l'avancer de mon projet, mais j'ai aussi tenu à jour un journal de travail.

Finalement, j'ai réalisé mon application selon l'architecture MVC, en utilisant l'outil de développement XCode offrant l'accès à Interface Builder pour l'interface graphique, au fur et à mesure du développement, j'effectuais des tests sur un Iphone 11

Résultat

L'application est fonctionnelle à 100 %, la totalité des fonctionnalités/objectifs du cahier des charges ont été implémentés.

L'aspect graphique de l'application est améliorable, certains objets pourraient être modifiés pour harmoniser l'ensemble de l'application.

Lors de la réalisation, j'ai mis en place un système de compte via une authentification par courriel et mot de passe, ce qui constitue une fonctionnalité supplémentaire importante.

Le répertoire du projet est disponible à cette adresse :

https://github.com/KillermyB/TPI_inventaireHabits_BoukhlifaK

6.2 Webographie

- <https://app.moqups.com/> - Site internet utilisé pour la réalisation des maquettes graphique.
- <https://lucid.app/> - Site internet utilisé pour la réalisation des maquettes graphiques.
- <https://firebase.google.com/> Documentation officiel pour le développement avec Firebase.
- <https://www.apple.com/chfr/swift/> Documentation officiel pour le développement swift
- <https://www.youtube.com/channel/UCnksRRifsSCGUZpQukUKAYa>, Chaîne Youtube : IOS Academy, consultation de différents tutoriels.
- <https://developer.apple.com/> Documentation pour les développeurs Apple.
- https://www.youtube.com/channel/UCLbjUazAH_8INQ9vIStdGdg Chaîne Youtube : Code Pro, consultation de différents tutoriels.
- https://www.youtube.com/channel/UCInZ_vHHL4gfHGiukO1zkg Chaîne Youtube : Replicode, consultation de différents tutoriels.
- <https://www.youtube.com/channel/UCbTw29mcP12YITt1EpUaVJw> Chaîne Youtube : Sean Allen, consultation de différents tutoriels.
- <https://www.youtube.com/user/CodeWithChris> Chaîne Youtube : CodeWithChris, consultation de différents tutoriels.

6.3 Table des illustrations

Figure 1 Modèle en cascade - Page wikipédia.....	11
Figure 2 Maquette : Page de connexion	12
Figure 3 Maquette : Page de création de compte	13
Figure 4 Maquette : Page principale de l'application.....	14
Figure 5 Maquette : Page principale de l'application (Bouton sélection activé)	15
Figure 6 Maquette : Page du profil utilisateur.....	16
Figure 7 Maquette : Page ajout d'un article	17
Figure 8 Maquette : Page détail d'un article	18
Figure 9 Maquette : Page modification d'un article	19
Figure 10 Maquette : Navigation de l'application.....	20
Figure 11 Maquette : Modèle conceptuel des données	21
Figure 12 Graphique : Bilan de la planification initiale	23
Figure 13 Firebase Authentication : Page contenant les Users	25
Figure 14 Firebase Firestore : Explication du stockage orienté "documents"	26
Figure 15 Architecture - Application : Modèle	27
Figure 16 Architecture – Application : Vue - Dossiers	27
Figure 17 Architecture - Application : Vues de l'application	27
Figure 18 Architecture - Application : Contrôleurs - Dossier.....	28
Figure 19 Maquette : Cas d'utilisation de l'application.....	29
Figure 20 Maquette : Modèle logique des données.....	30
Figure 21 Répertoire : Projet.....	31
Figure 22 Répertoire : Application.....	32
Figure 23 View Model : UserServices.swift	33
Figure 24 Model : User.swift	33
Figure 25 SignUpViewController.swift	34
Figure 26 Model : Product.swift	34
Figure 27 Controller : Cells - Class ItemsCollectionViewCell	35
Figure 28 Controller : HomeViewController.swift - Func collectionView.....	35
Figure 29 Application : Cellule exemple	35
Figure 30 Controller : HomeViewController.swift - Func setup	36
Figure 31 Controller : HomeViewController.swift - Func sortFilterData	36
Figure 32 Application : Bouton sélection des catégories.	36
Figure 33 Application : Menu déroulant des catégories.....	36
Figure 34 Firebase : Collection t_user	37
Figure 35 Firebase : Collection t_product	37

6.4 Dossier des tests

Vous trouverez dans le répertoire du projet sous « 1.Documentation » dans le sous-dossier :
 « [Test Vidéos](#) » les tests réalisés sous forme de courtes vidéos ou de captures d'écrans provenant de l'application.

6.5 Archives du projet

Vous pouvez retrouver le dossier d'archivage sur le GitHub du projet en suivant ce lien : https://github.com/KillermyB/TPI_inventaireHabits_BoukhlifaK

Vous pourrez y retrouver toutes les informations importantes du projet.

6.6 Planification – Journal de travail

mercredi, 5 mai 2021
Après-midi

jeudi, 6 mai 2021
Matin

jeudi, 6 mai 2021
Après-midi

vendredi, 7 mai 2021
Matin

Tâches - objectifs	Tranche 5min	Séquence 1								Séquence 2								Séquence 3								Séquence 4							
		0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h
Absences - imprévus	0h	0																															
	2h30min	30																															
Analyse - Planification initiale/détaillée	5h	60																															
	3h35min	43																															
Introduction au projet	0h30min	6																															
	0h30min	6																															
Analyse - Cahier des charges	0h35min	7																															
	0h35min	7																															
Doc - Journal de travail	4h10min	50																															
	5h45min	69																															
Doc - Rapport	11h55min	143																															
	15h25min	185																															
Analyse - Recherches	5h10min	62																															
	2h5min	25																															
Analyse - Modélisation base de données	2h10min	26																															
	1h15min	15																															
Analyse - Maquettes graphiques	3h55min	47																															
	4h45min	57																															
Réalisation - Création GIT	0h35min	7																															
	1h25min	17																															
Réalisation - Fonctionnalités	20h50min	250																															
	24h5min	289																															
Réalisation - Commentaires	6h15min	75																															
	4h20min	52																															
Réalisation - Interfaces	14h40min	176																															
	12h25min	149																															
Réalisation - Base de données	3h30min	42																															
	3h45min	45																															
Réalisation - Tests	4h30min	54																															
	2h35min	31																															
Doc - Marche à suivre	1h10min	14																															
	1h30min	18																															
Doc - Tableau des besoins utilisateurs	0h50min	10																															
	0h30min	6																															
Suivi de projet	1h25min	17																															
	2h20min	28																															
Doc - Abstract	2h40min	32																															
	0h30min	6																															

Tâches - objectifs	vendredi, 7 mai 2021 Après-midi								lundi, 10 mai 2021 Matin								mercredi, 12 mai 2021 Matin								mercredi, 12 mai 2021 Après-midi							
	Séquence 5								Séquence 6								Séquence 7								Séquence 8							
	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h
Absences - imprévus																																
Analyse - Planification initiale/détailée	■								■	■	■	■	■	■	■	■																
Introduction au projet																																
Analyse - Cahier des charges																																
Doc - Journal de travail					■																											
Doc - Rapport	■	■	■						■	■							■	■	■	■	■	■	■	■								
Analyse - Recherches	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■																
Analyse - Modélisation base de données			■	■																												
Analyse - Maquettes graphiques									■	■																						
Réalisation - Création GIT																																
Réalisation - Fonctionnalités																		■	■	■	■	■	■	■								
Réalisation - Commentaires																																
Réalisation - Interfaces																									■	■	■	■	■	■	■	
Réalisation - Base de données									■	■	■	■	■	■	■	■		■	■	■	■	■	■	■								
Réalisation - Tests																																
Doc - Marche à suivre																																
Doc - Tableau des besoins utilisateurs			■	■	■	■	■	■											■	■												
Suivi de projet					■	■	■	■									■	■														
Doc - Abstract																									■	■	■	■	■	■	■	

Tâches - objectifs	lundi, 17 mai 2021 Matin								mercredi, 19 mai 2021 Matin								mercredi, 19 mai 2021 Après-midi								jeudi, 20 mai 2021 Matin										
	Séquence 9								Séquence 10								Séquence 11								Séquence 12										
	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h			
Absences - imprévus																																			
Analyse - Planification initiale/détailée																																			
Introduction au projet																																			
Analyse - Cahier des charges																																			
Doc - Journal de travail																																			
Doc - Rapport																																			
Analyse - Recherches																																			
Analyse - Modélisation base de données																																			
Analyse - Maquettes graphiques																																			
Réalisation - Création GIT																																			
Réalisation - Fonctionnalités																																			
Réalisation - Commentaires																																			
Réalisation - Interfaces																																			
Réalisation - Base de données																																			
Réalisation - Tests																																			
Doc - Marche à suivre																																			
Doc - Tableau des besoins utilisateurs																																			
Suivi de projet																																			
Doc - Abstract																																			

Tâches - objectifs	jeudi, 20 mai 2021 Après-midi								vendredi, 21 mai 2021 Matin								vendredi, 21 mai 2021 Après-midi								mercredi, 26 mai 2021 Matin							
	Séquence 13								Séquence 14								Séquence 15								Séquence 16							
	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h
Absences - imprévus																																
Analyse - Planification initiale/détaillée																																
Introduction au projet																																
Analyse - Cahier des charges																																
Doc - Journal de travail																																
Doc - Rapport																																
Analyse - Recherches																																
Analyse - Modélisation base de données																																
Analyse - Maquettes graphiques																																
Réalisation - Création GIT																																
Réalisation - Fonctionnalités																																
Réalisation - Commentaires																																
Réalisation - Interfaces																																
Réalisation - Base de données																																
Réalisation - Tests																																
Doc - Marche à suivre																																
Doc - Tableau des besoins utilisateurs																																
Suivi de projet																																
Doc - Abstract																																

Tâches - objectifs	mercredi, 26 mai 2021 Après-midi								jeudi, 27 mai 2021 Matin								vendredi, 28 mai 2021 Matin								vendredi, 28 mai 2021 Après-midi							
	Séquence 17								Séquence 18								Séquence 19								Séquence 20							
	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h
Absences - imprévus																																
Analyse - Planification initiale/détaillée																																
Introduction au projet																																
Analyse - Cahier des charges																																
Doc - Journal de travail																																
Doc - Rapport																																
Analyse - Recherches																																
Analyse - Modélisation base de données																																
Analyse - Maquettes graphiques																																
Réalisation - Création GIT																																
Réalisation - Fonctionnalités																																
Réalisation - Commentaires																																
Réalisation - Interfaces																																
Réalisation - Base de données																																
Réalisation - Tests																																
Doc - Marche à suivre																																
Doc - Tableau des besoins utilisateurs																																
Suivi de projet																																
Doc - Abstract																																



Tâches - objectifs	lundi, 31 mai 2021 Matin								mercredi, 2 juin 2021 Matin								mercredi, 2 juin 2021 Après-midi								jeudi, 3 juin 2021 Matin							
	Séquence 21								Séquence 22								Séquence 23								Séquence 24							
	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h	0h30	1h	1h30	2h	2h30	3h	3h30	4h
Absences - imprévus																																
Analyse - Planification initiale/détailée																																
Introduction au projet																																
Analyse - Cahier des charges																																
Doc - Journal de travail								1																								
Doc - Rapport																																
Analyse - Recherches																																
Analyse - Modélisation base de données																																
Analyse - Maquettes graphiques																																
Réalisation - Création GIT																																
Réalisation - Fonctionnalités																																
Réalisation - Commentaires																																
Réalisation - Interfaces																																
Réalisation - Base de données																																
Réalisation - Tests																																
Doc - Marche à suivre																																
Doc - Tableau des besoins utilisateurs																																
Suivi de projet																																
Doc - Abstract																																

Tâches - objectifs	jeudi, 3 juin 2021 Après-midi										vendredi, 4 juin 2021 Matin									
	Séquence 25										Séquence 26									
	0h30	1h	1h30	2h	2h30	3h	3h30	4h	4h30		0h30	1h	1h30	2h	2h30	3h	3h30	4h	4h30	
Absences - imprévus																				
Analyse - Planification initiale/détaillée																				
Introduction au projet																				
Analyse - Cahier des charges																				
Doc - Journal de travail																				
Doc - Rapport																				
Analyse - Recherches																				
Analyse - Modélisation base de données																				
Analyse - Maquettes graphiques																				
Réalisation - Création GIT																				
Réalisation - Fonctionnalités																				
Réalisation - Commentaires																				
Réalisation - Interfaces																				
Réalisation - Base de données																				
Réalisation - Tests																				
Doc - Marche à suivre																				
Doc - Tableau des besoins utilisateurs																				
Suivi de projet																				
Doc - Abstract																				



6.7 Journal de travail

Séquence 1			Date: mercredi, 5 mai 2021	Après-midi
Tâche	Tranche [5min]	Temps Total	Explications: qu'est-ce qui se fait et comment ?	
Introduction au projet	6	0h30min	Introduction au projet TPI, entrevue avec l'expert 1 : Xavier Carrel, Prise d'informations auprès de M.Gruaz	
Analyse - Cahier des charges	7	0h35min	Prise de connaissance du CDC	
Analyse - Planification initiale/détailée	10	0h50min	Création du planning initiale après la prise de connaissance du CDC	
Réalisation - Création GIT	15	1h15min	Création du répertoire GIT contenant les documents pour mon projet TPI, J'ai rencontré quelques difficultés lors de l'utilisation de GIT sur MAC étant donné que c'était la première fois, j'ai donc suivi un tutoriel Youtube afin d'y parvenir.	
Total tranche	38	3h10min		
Séquence 2			Date: jeudi, 6 mai 2021	Matin
Tâche	Tranche [5min]	Temps Total	Explications: qu'est-ce qui se fait et comment ?	
Réalisation - Création GIT	2	0h10min	J'ai effectuer mon premier push de projet via GitKraken vers Github	
Analyse - Planification initiale/détailée	27	2h15min	Réalisation de la planification initiale complète.	
Doc - Journal de travail	2	0h10min	Rédaction de mon journal de travail pour la première partie de la matinée.	
Analyse - Maquettes graphiques	15	1h15min	Création des maquettes graphiques de l'application : Page principale, Page ajout d'un article, Page Modification d'un article et Page détail d'un article.	
Suivi de projet	1	0h5min	Envoie du mail aux experts ainsi qu'à M.Gruaz contenant le lien du répertoire GIT et la planification initiale.	
Doc - Journal de travail	1	0h5min	Rédaction de mon journal de travail pour la seconde partie de la matinée.	
Total tranche	48	4h		



			Date: jeudi, 6 mai 2021	Après-midi
Séquence	3			
Tâche	Tranche [5min]	38	Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Analyse - Maquettes graphiques	25	2h5min	Création des maquettes graphiques de l'application : Page de connexion, Page principale, Page ajout d'un article, Page Modification d'un article et Page détail d'un article.	Site web pour faire les maquettes graphique : https://app.moqups.com/
Suivi de projet	3	0h15min	Discussion avec Perlez Carlos et Gruaz Gilbert concernant un problème de lisibilité dans le modèle de journal de travail que j'utilise.	
Doc - Rapport	4	0h20min	J'ai commencé la rédaction de mon rapport, j'ai complété l'introduction et ajouté la planification initiale au rapport.	
Doc - Journal de travail	3	0h15min	Ajout d'une page "Tâche" sur le journal de travail afin de stipuler de manière pertinente les tâches "en cours" - "terminé" - "abandonné" - "bloquante" etc.. (Selon un tableau Trello)	
Doc - Journal de travail	2	0h10min	Rédaction de mon journal de travail pour la première partie de la matinée.	
Suivi de projet	1	0h5min	Push du répertoire sur GIT	
Total tranche	38	3h10min		
Séquence	4		Date: vendredi, 7 mai 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Analyse - Maquettes graphiques	12	1h	J'ai amélioré les maquettes graphiques de mon application, en essayant de créer une interface UI accessible et cohérente. Annexe : Maquette - 2.1.1 Rapport	Site web pour faire les maquettes graphique : https://app.moqups.com/
Analyse - Modélisation base de données	10	0h50min	J'ai commencé le modèle conceptuel des données de ma base de données. Annexe : Maquette - 2.1.2 Rapport	https://www.lucidchart.com - Outil pour la création du MCD
Analyse - Recherches	14	1h10min	J'ai effectué des recherches concernant les différentes fonctionnalités à implémenter lors de la réalisation de mon projet, le service que j'utiliserais pour la gestion de la base de données. Pour le moment, j'ai trouvé trois manières de stocker mes données qui me semblent intéressante Firebase avec Firestore (celle qui semble être la plus simple), Core Data semble être une bonne méthode mais probablement trop complexe pour mon projet et pour finir Realm qui est plus simple par rapport à Core Data et est un outil cross-platform (Android et iOS) Mes prochaines recherches : Firebase (Firestore ou Realtime DataBase) et Realm	Dev to - IOS Database comparison. What are the differences ?
Doc - Journal de travail	2	0h10min	Rédaction du journal de travail de la matinée	
Total tranche	38	3h10min		



Séquence 5			Date: vendredi, 7 mai 2021	Après-midi
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Analyse - Planification initiale/détaillée	6	0h30min	J'ai modifié ma planification initiale en y ajoutant des détails sur certaines de tâches afin d'obtenir une planification détaillée.	
Doc - Rapport	14	1h10min	J'ai continué l'analyse préliminaire du rapport, j'ai ajouté les objectifs en les fragments en tâches précises, j'ai également ajouté un graphique pour la planification et commencé à décrire les maquettes.	
Analyse - Modélisation base de données	5	0h25min	J'ai terminé le modèle conceptuel des données de ma base de données. Annexe : Maquette - 2.1.2 Rapport	https://www.lucidchart.com - Outil pour la création du MCD
Analyse - Recherches	5	0h25min	J'ai approfondi mes recherches concernant les différentes bases de données utilisables pour une application iOS, et j'ai finalement fini par choisir Firebase comme service pour effectuer ce projet. Suite à plusieurs tutoriels ainsi que certains articles que j'ai pu parcourir brièvement sur le sujet, j'ai constaté que c'était une bonne solution et facilement adaptable à mes besoins.	
Doc - Tableau des besoins utilisateurs	4	0h20min	J'ai commencé à créer le tableau des besoins utilisateurs, je n'ai pas eu le temps de le finir, je l'implémenterais donc dans le rapport lundi matin.	https://www.lucidchart.com - Outil pour la création du tableau des besoins utilisateurs.
Doc - Journal de travail	2	0h10min	Rédaction du journal de travail de l'après-midi	
Suivi de projet	2	0h10min	Envoie du mail de fin de semaine aux experts ainsi qu'à M.Gruaz, Push du Git.	
Total tranche	38	3h10min		
Séquence 6			Date: lundi, 10 mai 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Base de données	12	1h	J'ai commencé la création de la base de données sur FireBase, J'ai créé les différentes tables avec les différents éléments : User, Size, Color, Condition, Price, Product name etc.	
Analyse - Maquettes graphiques	5	0h25min	J'ai finalisé les différentes maquettes graphiques, et je les ai implémentées dans le rapport.	
Doc - Rapport	3	0h15min	J'ai commencé la partie réalisation du rapport.	
Réalisation - Base de données	15	1h15min	Implémentation du POD FireBase sur Xcode afin de lier ma base de donnée et le service firebase. Puis j'ai commencé à regarder des tutoriels pour gérer la lecture et l'écriture dans une DB à travers Xcode. Tutoriel FireBase - Lecture, écriture etc..	Ajoutez Firebase à un projet iOS
Doc - Journal de travail	2	0h10min	Rédaction du journal de travail de la matinée.	
Suivi de projet	1	0h5min	Push du Git.	
Total tranche	38	3h10min		



Séquence 7			Date: mercredi, 12 mai 2021		Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...	
Doc - Rapport	11	0h55min	Rédaction de la conception de mon TPI : - Maquettes Graphiques, MCD et MLD ajouter au rapport - Dossier de conception complété		
Réalisation - Base de données	6	0h30min	J'ai passé en revue la base de données que j'avais créée lundi matin. Je l'ai comparé avec le MLD afin de vérifier que je n'avais oublié aucun champ lors de sa création.		
Réalisation - Fonctionnalités	10	0h50min	Installation du POD "CocoaPods" qui permet la gestion des dépendances pour Swift et permet de gérer des librairies externes	Tutoriel CocoaPods - Installation pour Xcode	
Doc - Tableau des besoins utilisateurs	2	0h10min	Implémentation du tableau des besoins utilisateurs dans le rapport.		
Réalisation - Interfaces	18	1h30min	Mise en place de l'interface de l'application, mise en place de certains boutons sur la page principale.		What is model view Controller with Example - Swift
Doc - Journal de travail	1	0h5min	Rédaction du journal de travail de la matinée.		
Total tranche	48	4h			
Séquence 8			Date: mercredi, 12 mai 2021		Après-midi
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...	
Réalisation - Interfaces	34	2h50min	Finalisation du MVC de l'application. Ajout des fichiers de configuration permettant la lecture de la base de données, tel que la récupération des différents champs de la table t_users dans le fichier User.swift J'ai également ajouté tous les éléments qui seront présents sur la page principale ainsi que sur la page profil de l'utilisateur. J'ai initialisé le POD de firebase permettant l'accès à FireStore et j'ai ensuite configuré l'écran de chargement de l'application.		Swift 5: Firebase Database in App - Setup/Read/Write Data (Xcode 11) - Beginners
Doc - Journal de travail	2	0h10min	Rédaction du journal de travail de la matinée et de l'après-Midi		
Suivi de projet	2	0h10min	Envoie du mail de fin du mercredi aux experts ainsi qu'à M.Gruaz. Push du Git.		
Total tranche	38	3h10min			



Séquence 9			Date: lundi, 17 mai 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Interfaces	15	1h15min	<p>J'ai créé les différentes vues concernant l'authentification étant donné que pour simplifier le stockage de mes données, j'ai décidé d'utiliser également le service d'authentification de Firebase, ayant travaillé sur son implémentation lors de mon pré-TPI, cela ne devrait pas me prendre trop de temps à réimplémenter sur cette application.</p> <p>Lors de la planification du projet, je n'avais pas prévu de devoir refaire le système d'authentification pour mon application mais pour la suite du projet, le rajouter me permettra de gagner du temps, voilà pourquoi je m'éloigne de ma planification.</p> <p>J'ai créé la page de connexion, avec les différentes redirections disponibles sur la page, l'authentification n'est pas encore fonctionnelle.</p>	swiftludus.org - How to navigate between view controllers in iOS
Réalisation - Commentaires	4	0h20min	J'ai commenté la page connexion de mon application.	
Réalisation - Base de données	12	1h	J'ai lié ma base de données sur Firebase avec les ID des comptes créés à l'aide du service d'authentification de Firebase.	Premiers pas avec l'authentification Firebase sur iOS
Réalisation - Interfaces	6	0h30min	J'ai ajouté la page de création d'un compte mais pour le moment, les boutons sont inutilisables et elle est liée à aucune autre page de l'application.	
Doc - Journal de travail	1	0h5min	Rédaction du journal de travail de la matinée	
Total tranche	38	3h10min		



Date: mercredi, 19 mai 2021

Matin

Séquence 10			Date: mercredi, 19 mai 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Fonctionnalités	35	2h55min	<p>J'ai mis en place le système d'authentification de mon application, j'ai configuré la création des différents utilisateurs via l'adresse mail et un mot de passe stocker via firebase. J'ai géré les erreurs lors de la saisie des informations, la création d'un nouvel utilisateur et la récupération des informations d'un utilisateur précédemment créé.</p> <p>Pour réaliser cela, j'ai dû installer les PODS d'authentification de firebase et également modifier le fichier de configuration concernant mon application, contenant désormais la gestion de l'authentification.</p>	https://firebase.google.com/docs/auth/ios/manage-users
Réalisation - Interfaces	6	0h30min	<p>J'ai terminé la liaison de la page "Connexion" avec la page "Création d'un compte", j'ai mis en place correctement les différents bouton (Bouton retour, bouton "création d'un compte, textbox pour l'email et le mot de passe), je me suis basé sur mes maquettes graphiques, mais j'ai effectuer quelques modifications dû à certaines contraintes imposée par l'interface builder mis à disposition sur xCode qui ne me permettait pas répliquer à l'identique mes maquettes.</p>	https://medium.com/firebase-developers/ios-firebase-authentication-sdk-email-and-password-login-6a3bb27e0536
Analyse - Recherches	6	0h30min	<p>j'ai rencontré un problème d'affichage lors du lancement de l'application suite à des zones de contraintes "AutoLayout" qui empêche de placer certains éléments à certains endroits de l'écran pour permettre la responsivité et l'utilisation de l'application sur d'autres modèles d'iphones.</p> <p>Pour ce projet, j'ai mis de côté la compatibilité d'affichage avec d'autres appareils IOS étant donné que cette application est développée dans l'objectif de fonctionner sur un Iphone 11.</p>	https://firebase.google.com/docs/ios/setup https://developer.apple.com/library/archive/documentation/UserExperience/Conceptual/AutolayoutPG/WorkingwithConstraintsinInterfaceBuilder.html#~text=Interface%20Builder%20Provides%20four%20Auto%20make%20multiple%20constraints%20at%20once.
Doc - Journal de travail	1	0h5min	Rédaction du journal de travail de la journée.	
Total tranche	48	4h		
Séquence 11	Date: mercredi, 19 mai 2021			Après-midi
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Fonctionnalités	13	1h5min	J'ai mis en place la récupération d'un compte précédemment créé et la lecture des différents comptes.J'ai lié les différents comptes avec la base de données, lorsqu'un utilisateur crée son compte, il est directement connecté et s'il ferme l'application, il reste connecté.	https://firebase.google.com/docs/auth/ios/manage-users
Réalisation - Interfaces	15	1h15min	Désormais lorsque l'utilisateur crée son compte, il est redirigé sur la page principale et également lorsqu'il se connecte.	
Absences - imprévus	6	0h30min	J'ai eu une conversation avec mon camarade Luca Böhnen afin de lui expliquer pourquoi il devrait plutôt utiliser firebase à la place de Core Data, en lui montrant l'exemple avec mon application. Je lui ai expliqué brièvement comment mettre en place firebase.	
Suivi de projet	2	0h10min	Push des modifications de la journée sur le Git.	
Doc - Journal de travail	2	0h10min	Rédaction du journal de travail de l'après midi.	



Date: jeudi, 20 mai 2021

Matin

Séquence 12				
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Absences - imprévus	13	1h5min	J'ai continué la conversation avec mon camarade Luca Böhlen, pour lui expliquer de manière approfondie l'utilisation de Firebase ainsi que son fonctionnement, implémentation, et l'utilité qu'aura ce service dans son projet.	
Réalisation - Fonctionnalités	26	2h10min	<p>Je me suis renseigné et j'ai implémenté dans mon projet différents pods que je vais utiliser lors de ce projet :</p> <ul style="list-style-type: none"> - un pods permettant de modifier un bouton pour lui appliquer différentes formes etc. Couleurs de fond directement dans l'interface builder, je vais utiliser ces boutons pour les implémenter sur mes pages de connexion - Un pods pour pouvoir ajouter des menus déroulant lorsque l'utilisateur cliquera sur un bouton, j'ai commencé à essayer de l'implémenter sur la page ajout d'un produit pour sélectionner la taille du produit ainsi que sa catégorie. - Un pods pour afficher à l'utilisateur un état de progression lorsque l'application le connectera avec Firebase authentification ou lorsque l'utilisateur ajoutera un produit. 	https://github.com/IdleHandsApps/DesignableButton
Réalisation - Commentaires	7	0h35min	J'ai commenté mon code concernant l'authentification "LoginController/CreateAccountController" et j'ai commencé à commenter le code concernant la page d'ajout d'un article et la page principale sur laquelle, je n'ai pas encore pu travaillé.	
Doc - Journal de travail	2	0h10min	Rédaction du journal de travail de la matinée.	
Total tranche	48	4h		
Séquence 13			Date: jeudi, 20 mai 2021	Après-midi
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Fonctionnalités	20	1h40min	<p>J'ai commencé à développer la page principale de l'application :</p> <ul style="list-style-type: none"> - j'ai ajouté les différents liens entre les différentes pages, j'ai ajouté le bouton profil, j'ai commencé le développement du bouton de sélection ainsi que le bouton qui permettra de trier par catégorie les articles de l'utilisateur. - J'ai ajouté le bouton pour accéder à la page d'ajout d'un produit, et j'ai commencé à structurer le modèle des cases qui permettra de visionner les produits sur la première page. 	https://www.youtube.com/watch?v=OQYqGM5_wVY&ab_channel=iOSAcademy
Réalisation - Interfaces	12	1h	<p>J'ai commencé à structurer la page d'ajout d'un article avec les différents éléments qui seront sélectionnables pour un article (catégorie, taille, prix, date d'achat etc..)</p> <p>J'ai suivi ma maquette graphique.</p>	
Suivi de projet	5	0h25min	Rendez-vous avec Monsieur Perez pour la seconde visite du TPI, pris de connaissance de différentes informations concernant la suite du déroulement du TPI, pour le moment suite au bilan, je n'ai rien eu à signaler de particulier à Monsieur Perez, le TPI suit son cours. Monsieur Perez veut une version papier du projet.	
Doc - Journal de travail	1	0h5min	Rédaction du journal de travail de la matinée.	
Total tranche	38	3h10min		



Séquence 14			Date: vendredi, 21 mai 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Absences - imprévus	6	0h30min	Je suis arrivé en retard suite à un problème avec mon train, je vais faire en sorte de rattraper cet imprévu sur mon temps libre.	
Réalisation - Commentaires	8	0h40min	J'ai commencé à rédiger les commentaires des pages que j'ai implémentées hier dans l'application.	
Doc - Rapport	14	1h10min	J'ai dû retravailler mon dossier de conception sur mon rapport après une lecture approfondie des critères d'évaluation, il semblait me manquer certaines informations sur mon projet.	
Réalisation - Interfaces	10	0h50min	J'ai continué à travailler sur la conception des cases pour afficher les articles de l'utilisateur, je me heurte à un problème de conception, je ne sais pas encore comment je peux disposer les éléments en forme de liste. J'ai pris le temps de regarder différents tutoriels où guident afin de comprendre plus clairement les différentes possibilités d'affichages qui s'offraient à moi afin que mon application soit le plus proche possible des maquettes graphiques que j'ai réalisées.	https://www.youtube.com/watch?v=WK5vrOD1zCQ&ab_channel=iOSAcademyiOSAcademy
				https://www.youtube.com/watch?v=FtO5QT2D_H8&ab_channel=SeanAllenSeanAllen
				https://www.youtube.com/watch?v=C36sb5sc6IE&ab_channel
Total tranche Séquence	38	3h10min		Date: vendredi, 21 mai 2021
Séquence 15			Après-midi	
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Fonctionnalités	25	2h5min	J'ai continué à travailler sur la conception des cases pour afficher les articles de l'utilisateur, j'ai mis en place les différents éléments et j'ai réussi à mettre en place la possibilité de créer des cases en appuyant sur le bouton ajouté, pour le moment, je n'ai pas réussi à récupérer des informations depuis la base de données pour les afficher dans les cases. Cependant, j'ai eu le temps de me renseigner sur le moyen de réaliser ça, et je pense pouvoir mettre en place la récupération des éléments de la base de données depuis les cases d'ici mercredi prochain.	https://stackoverflow.com/questions/66279978/swift-how-to https://firebase.google.com/docs/database/ios/lists-of-data https://www.youtube.com/watch?v=4RUeW5rUcw&ab_channel=Firebase
Doc - Rapport	5	0h25min	J'ai rédigé la stratégie des tests de mon projet, que je n'ai pas encore retranscrit sur mon rapport.	
Doc - Journal de travail	6	0h30min	Rédaction du journal de travail de la matinée et de l'après-midi.	
Suivi de projet	2	0h10min	Envoie du mail de fin de semaines aux experts M.Carrel et M.Perez ainsi qu'à M.Gruaz. Push du Git.	
Total tranche	38	3h10min		



Date: mercredi, 26 mai 2021 | Après-midi

Séquence 17				
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Doc - Rapport	3	0h15min	J'ai ajouté le chapitre glossaire et une table des illustrations dans mon rapport technique.	
Réalisation - Fonctionnalités	21	1h45min	J'ai travaillé sur la page ProductDetailViewController, j'ai mis en place les différentes informations contenues pour un article, via la fonction "setup" qui initialise les différents champs d'un article. J'ai effectué la récupération des données d'un article depuis la base de données lorsqu'il est sélectionné. Récupération des données d'un article sur la page détaillé, OK	
Réalisation - Interfaces	9	0h45min	J'ai ajusté l'emplacement de certains éléments de l'application, le bouton d'ajout d'un objet, l'icône pour accéder à la page profil, le futur emplacement des catégories des articles. J'ai ajouter un logo à l'application et modifier le design des boutons de l'application via le framework "DesignableButton"	https://github.com/IdleHandsApps/DesignableButton
Suivi de projet	2	0h10min	Envoie du mail du mercredi aux experts M.Carrel et M.Perez ainsi qu'à M.Gruaz. Push du Git.	
Doc - Journal de travail	3	0h15min	Rédaction du journal de travail de l'après-midi	
Total tranche	38	3h10min		
Séquence 18			Date: jeudi, 27 mai 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Doc - Rapport	28	2h20min	Rédaction de la réalisation du projet, (Suite à une arrivée tardive, j'ai commencé la rédaction du rapport dans le train durant 20 minutes) Glossaire de l'application, explication de certains éléments technique en rapport avec mon projet. Mise des légendes sur toutes les illustrations du projet. Ajout du cas d'utilisation de l'application	
Réalisation - Interfaces	17	1h25min	Mise en place de l'interface de la page profil. Image de profil, OK Bouton déconnexion et retour, OK Ajout de deux labels dans lesquels sont récupérer des informations provenant de la base de données, concernant, la collection de l'utilisateur, OK Récupération de la photo de profil/Possibilité de changer la photo, OK Récupération des articles dans la collection, OK Bouton déconnexion, OK	
Doc - Journal de travail	3	0h15min	Rédaction du journal de travail de la matinée	
Total tranche	48	4h		



Séquence 19			Date: vendredi, 28 mai 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Interfaces	7	0h35min	Mise en place de l'infrastructure de la page de modification d'un article. J'ai mis en places les différents boutons et champs de texte : 1 bouton pour ajouter une image, un conteneur d'image, 1 bouton retour, 3 textField, 3 dropdownMenu, 1 date timepicker, 1 bouton pour supprimer l'article et 1 pour enregistrer les modifications.	
Réalisation - Fonctionnalités	23	1h55min	Durant cette matinée, j'ai commencé à développer les fonctionnalités suivantes : Récupération des données d'un article dans les champs de la page modification, la suppression d'un article via son id, l'enregistrement des modifications, l'utilisation du framework des dropdown menu, j'ai finalisé la page d'ajout d'un article et j'ai commencé la gestion du tri des articles par catégorie sur la page principale. Page ajout d'un article, OK Récupération des informations de l'article sur la page modification, OK Modifications des données de l'article dans la base de données, OK Suppression d'un article, OK Implémentation des dropdown menu, OK Ajout du date time picker, OK Page modification d'un article, OK	https://www.youtube.com/watch?v=mgVRNrvJtLc&ab_channel=Rebeloper-RebelDeveloper
Réalisation - Commentaires	5	0h25min	J'ai rédigé les commentaires de la page d'ajout d'un article, de la page modification, et j'ai complété ceux de la page principale.	
Doc - Journal de travail	3	0h15min	Rédaction du journal de travail de la matinée	
Total tranche	38	3h10min		
Séquence 20			Date: vendredi, 28 mai 2021	Après-midi
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Fonctionnalités	28	2h20min	J'ai effectué des recherches sur la catégorisation de mes articles, j'ai trouvé un tutoriel qui m'a permis de réaliser pratiquement ce que je voulais et qui m'ont beaucoup aidé pour la réalisation de cette fonctionnalité. DropDown menu comportant les catégories, OK Cases d'articles qui s'affichent en fonction de la catégorie, OK	https://www.youtube.com/watch?v=4y_NLkYT6NU&ab_channel=ReplicodeReplicode
Réalisation - Commentaires	3	0h15min	J'ai rédigé les commentaires de la partie du code concernant la tri par catégorie des différents articles.	
Doc - Journal de travail	4	0h20min	Rédaction du journal de travail de l'après midi.	
Doc - Marche à suivre	3	0h15min	Envoie du mail du mercredi aux experts M.Carrel et M.Perez ainsi qu'à M.Gruaz. Push du Git.	
Total tranche	38	3h10min		



Séquence 21			Date: lundi, 31 mai 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Fonctionnalités	21	1h45min	J'ai réalisé la fonctionnalité de partage d'un article, j'ai géré la récupération des données dans la base de données lorsque l'utilisateur appuie sur le bouton partage de la page détail, ensuite les	
Réalisation - Fonctionnalités	5	0h25min	J'ai ajouté le bouton de sélection sur le menu principal ainsi que sur le menu de détail d'un article.	
Réalisation - Commentaires	2	0h10min	J'ai continué la rédaction des commentaires de la page "HomeViewController" et "ProductDetailViewController" selon la nouvelle fonctionnalité implémenté.	
Doc - Rapport	7	0h35min	Rédaction de la section "Réalisation" du projet	
Doc - Journal de travail	3	0h15min	Rédaction du journal de travail de la matinée	
Total tranche	38	3h10min		
Séquence 22			Date: mercredi, 2 juin 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Tests	10	0h50min	J'ai réalisé des tests sur mon application pour ensuite les retrancrire sur mon rapport.	
Doc - Rapport	25	2h5min	Rédaction de la section "test" de l'application, j'ai également effectué des vidéos afin de démontrer chaque test effectué. J'ai également conçu une maquette graphique comportant quelques captures d'écrans de l'application.	
Réalisation - Fonctionnalités	6	0h30min	J'ai finalisé la fonctionnalité permettant de sélectionné plusieurs articles et d'ensuite partager sous forme de textes leurs informations.	
Réalisation - Commentaires	3	0h15min	J'ai rédigé les commentaires de cette dernière fonctionnalité.	
Doc - Journal de travail	4	0h20min	Rédaction du journal de travail de la matinée	
Total tranche	48	4h		



Date: mercredi, 2 juin 2021

Après-midi

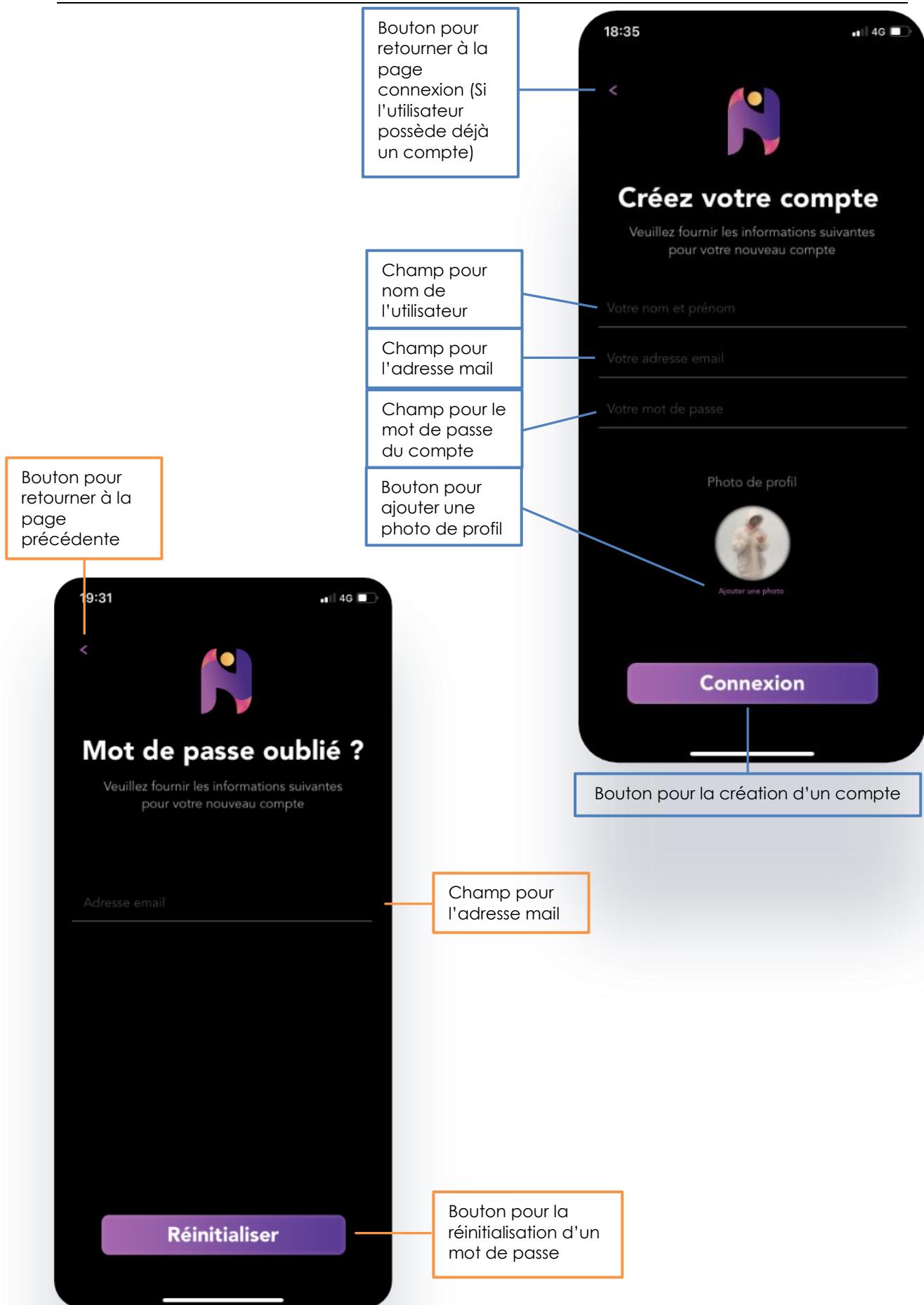
Séquence 23			Date: mercredi, 2 juin 2021	Après-midi
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Fonctionnalités	10	0h50min	Lors de la réalisation des tests de l'application, j'ai constaté une ou deux erreurs : - La récupération des informations sur la page de modification comportait une erreur dans les dates récupérées. - Les labels sur la page profil ne s'actualisaient pas lorsqu'un utilisateur rajoutait un article dans sa collection. J'ai donc corrigé ces deux erreurs que j'ai pu constater, et j'ai recommencé les tests de mon application.	
Réalisation - Tests	15	1h15min	J'ai réalisé des tests sur l'application, après avoir résolu deux problèmes survenus lors de la première séance de test.	
Réalisation - Commentaires	2	0h10min	Je suis retourné sur l'application pour améliorer certaines sections de commentaires où certaines pages sur lesquels d'après moi les explications n'étaient pas complètes.	
Doc - Rapport	7	0h35min	J'ai rédiger les sections suivantes du rapport de projet : -Réalisation, OK -Tests, -Conclusion, -Annexes J'ai également passé en revue mon rapport afin de corriger les possibles erreurs de mise en forme, ou d'orthographes.	
Doc - Journal de travail	2	0h10min	Rédaction du journal de travail de l'après-midi.	
Suivi de projet	2	0h10min	Envie du mail du mercredi aux experts M.Carrel et M.Perez ainsi qu'à M.Gruaz. Push du Git.	
Total tranche	38	3h10min		
Séquence 24			Date: jeudi, 3 juin 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Réalisation - Tests	6	0h30min	J'ai effectué une dernière série de tests afin de m'assurer que les modifications faites à l'application, n'ont pas créé de nouvelles erreurs.	
Doc - Rapport	25	2h5min	J'ai rédigé les sections suivantes du rapport de projet : -Réalisation, OK -Tests, OK -Conclusion -Annexes J'ai également passé en revue mon rapport afin de corriger les possibles erreurs de mise en forme, ou d'orthographes.	
Réalisation - Fonctionnalités	10	0h50min	J'ai ajouté la gestion de certaines erreurs lors de la saisie d'informations. Page Connexion OK Page Création d'un compte OK Page Mot de passe oublié OK Page Ajout d'un article OK Page modification d'un article OK	
Réalisation - Commentaires	3	0h15min	Rédaction des commentaires concernant la gestion des erreurs de saisie.	
Doc - Journal de travail	4	0h20min	Rédaction du journal de travail	
Total tranche	48	4h		

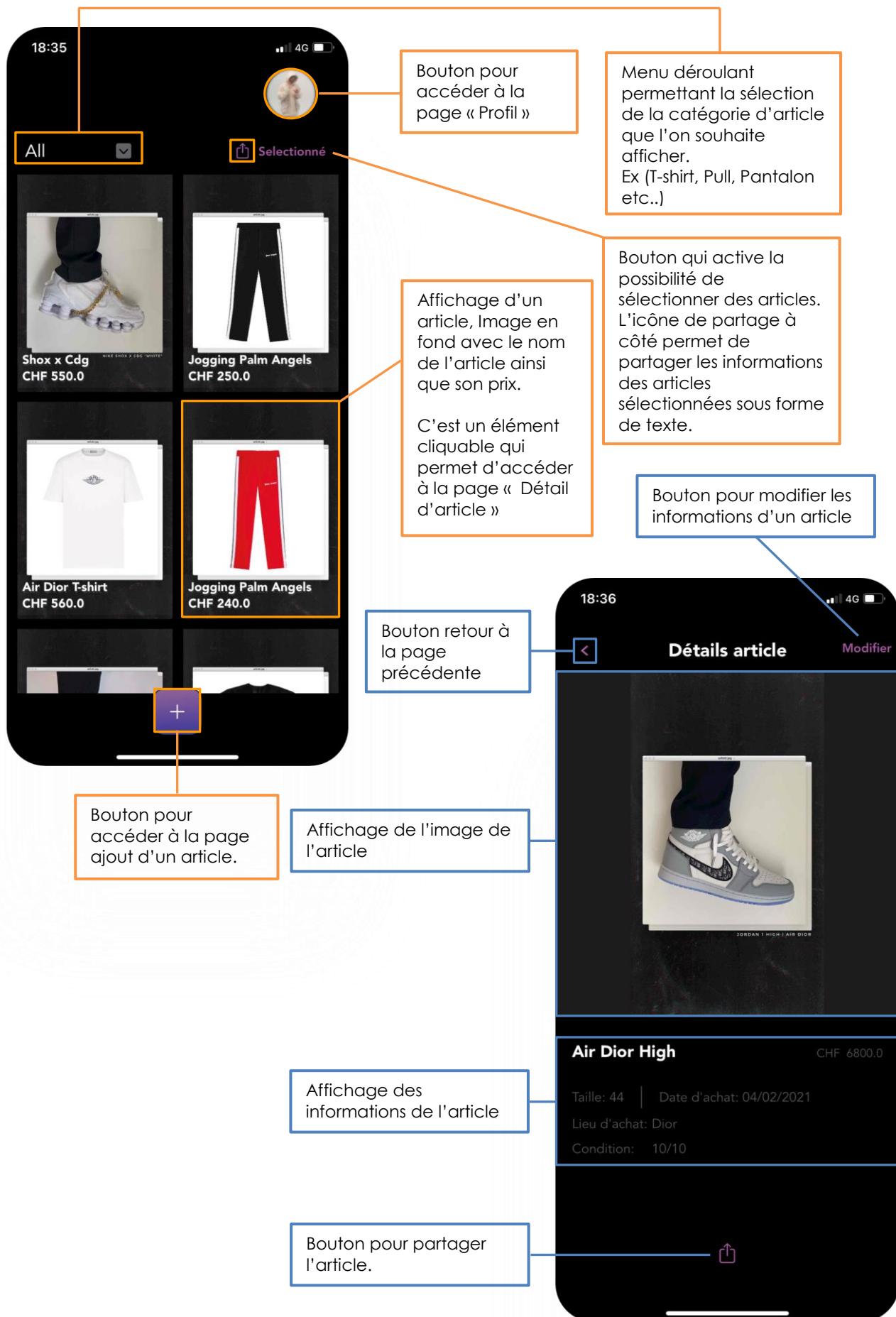


Séquence 25			Date: jeudi, 3 juin 2021	Après-midi
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Doc - Rapport	15	1h15min	J'ai rédigé les sections suivantes du rapport de projet : -Réalisation, OK	
Doc - Marche à suivre	15	1h15min	J'ai rédigé le manuel d'utilisation de l'application.	
Réalisation - Commentaires	6	0h30min	J'ai peaufiné les commentaires de mon application.	
Doc - Journal de travail	2	0h10min	Rédaction du journal de travail de l'après-midi.	
Total tranche	38	3h10min		
Séquence 26			Date: vendredi, 4 juin 2021	Matin
Tâche	Tranche [5min]		Explications: qu'est-ce qui se fait et comment ?	Liens, références, ...
Absences - imprévus	5	0h25min	Je n'ai pas pu prendre le train à l'heure ce matin, mais j'ai travaillé en attendant à la gare sur mon ordinateur portable et dans le train. J'ai ajouté 2 mots techniques dans le glossaire : Interface Builder et Langage Machine.	
Suivi de projet	5	0h25min	Peaufinage final du rapport, relecture finale, correction de certains éléments.	
Doc - Journal de travail	8	0h40min	J'ai corrigé mon journal de travail, et j'ai rédigé la dernière séquence du projet.	
Doc - Rapport	12	1h	Finalisation du manuel d'utilisation ainsi que du manuel d'installation de l'application.	
Doc - Abstract	6	0h30min	J'ai rédigé le résumé du projet et je l'ai incorporé dans le rapport du projet.	
Doc - Rapport	12	1h	J'ai mis en forme la section annexe du rapport, j'ai effectué des captures d'écrans du journal de travail pour les incorporer dans le rapport, j'ai rédigé mon bilan de planification. J'ai également remis en forme la table d'illustration ainsi que le glossaire et la webographie. J'ai fait un schéma du répertoire du projet que j'ai intégré dans la section réalisation du rapport. J'ai ensuite effectué des captures d'écrans du journal de travail pour l'implémenter dans mon rapport et rédiger le bilan de la planification. J'ai finalisé le rapport de projet.	
Total tranche	48	4h		

6.8 Manuel d'utilisation







Bouton retour à la page précédente

Bouton permettant l'ajout d'une **image dans la DB pour un article.**

Champ de texte pour le **nom de l'article et son **lieu d'achat**.**

Menus déroulant permettant de choisir la **catégorie de l'article et sa **taille**.**

Menu déroulant permettant de choisir la **condition de l'article.**

TextBox permettant la saisie du **prix d'achat ainsi qu'un DateTimePicker pour la **date d'achat**.**

Bouton retour à la page précédente

Bouton permettant l'ajout d'une **image dans la DB pour un article.**

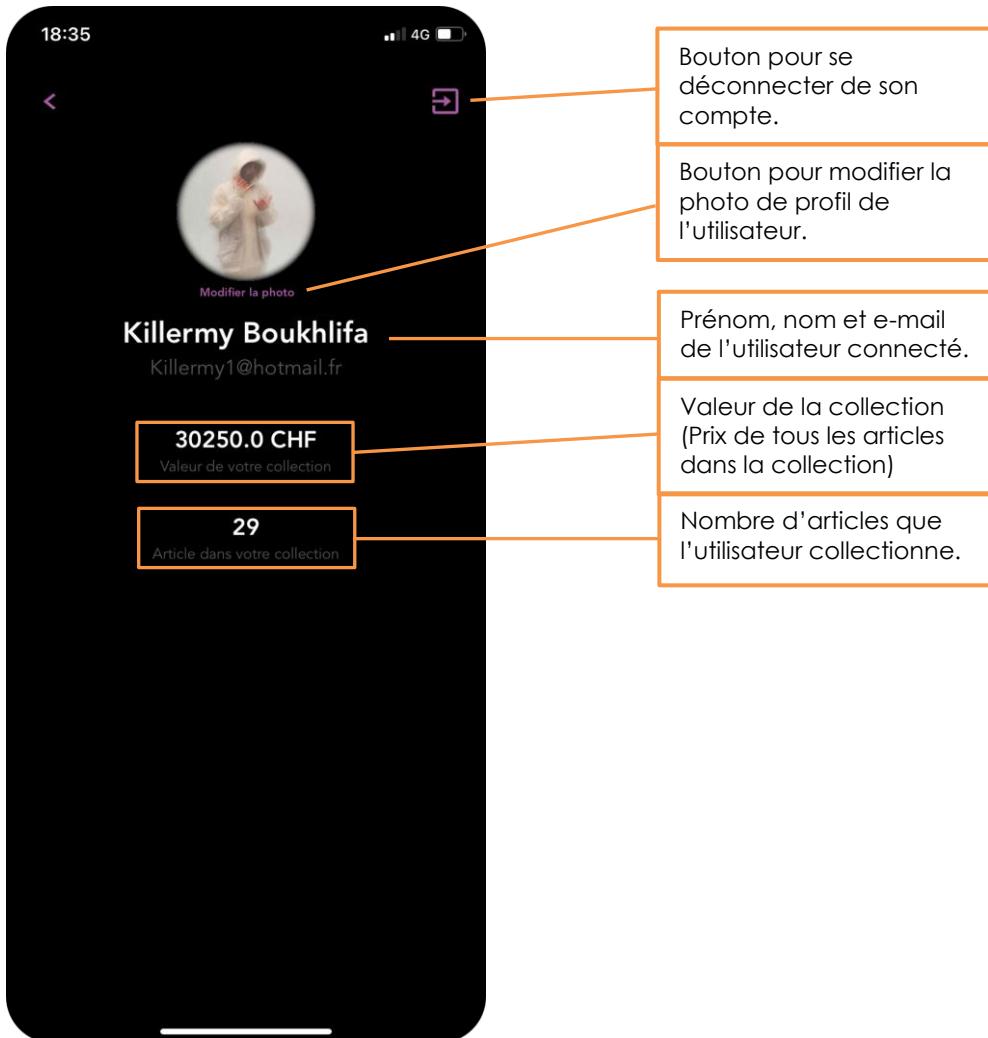
Champ de texte pour le **nom de l'article et son **lieu d'achat****

Menus déroulant permettant de choisir la **catégorie de l'article et sa **taille**.**

Menu déroulant permettant de choisir la **condition de l'article.**

TextBox permettant la saisie du **prix d'achat ainsi qu'un DateTimePicker pour la **date d'achat**.**

Bouton pour valider les informations saisies ou supprimer l'article.



6.1 Manuel d'installation

Pour commencer, il faut installer XCode sur l'apple store :

<https://apps.apple.com/ch/app/xcode/id497799835?l=fr&mt=12>

Ensuite, il faut installer les pods pour faire fonctionner l'application, en suivant ce tutoriel :

https://www.youtube.com/watch?v=7Sp7ojClZPU&ab_channel=Rebeloper-RebelDeveloperRebeloper-RebelDeveloper

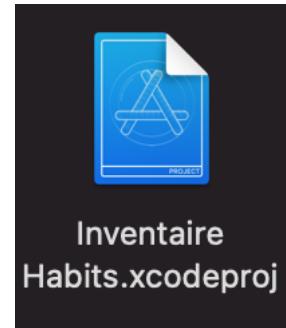
Voici les pods à installer :

```
# Uncomment the next line to define a global platform for your project
platform :ios, '14.5'

target 'Inventaire Habits' do
    # Comment the next line if you don't want to use dynamic frameworks
    use_frameworks!

    # Pods for Inventaire Habits
    pod 'Firebase/Core'
    pod 'Firebase/Auth'
    pod 'Firebase/Storage'
    pod 'Firebase/Firestore'
    pod 'IQKeyboardManagerSwift'
    pod 'MBProgressHUD', '~> 1.2.0'
    pod 'DropDown'
    pod 'SDWebImage', '~> 5.0'
end
```

Ensuite, il faut ouvrir le projet « Inventaire Habits » :



Et pour finir, il faut lancer l'application sur un simulateur ou sur un Iphone en version 14.5 branché sur le MAC, le bouton lancer est en haut de l'écran:

