

---

# LSINF1225 Rapport final 2019-2020

## Groupe L11

---

Réalisé par le groupe L11 :  
Nathan Nepper (14281800)  
Corentin Lingier (23431700)  
Corentin Lengele (14751700)  
Gwendal Laurent (20931800)  
Jules Lesuisse (17051800)  
William Ruan (31891400)

Travail fait sous tutorat de Liliya Semerikova  
Soumission le 08/05/2020

Code source du projet :  
<https://nnepper.github.io/LSINF1225-2020-GroupeL1.1-WishList>

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Diagrammes et récits utilisateurs</b>	<b>1</b>
2.1	Schéma ORM . . . . .	1
2.2	Schéma relationnel . . . . .	2
2.3	Récits utilisateurs . . . . .	2
2.3.1	Gestion de compte . . . . .	2
2.3.1.1	Création d'un compte . . . . .	2
2.3.1.2	Modifier son profil utilisateur . . . . .	2
2.3.2	Gestion de Wishlist . . . . .	2
2.3.2.1	Ajouter une WishList . . . . .	2
2.3.2.2	Ajout d'un article à une WishList . . . . .	2
2.3.2.3	Modifier un de mes articles . . . . .	3
2.3.2.4	Modifier mes préférences dans une wishlist . . . . .	3
2.3.2.5	Supprimer un article d'une WishList . . . . .	3
2.3.2.6	Modifier l'état d'un article d'un follower . . . . .	3
2.3.3	Gestion des follows . . . . .	3
2.3.3.1	Follow un utilisateur . . . . .	3
2.3.3.2	Accepter une demande de follow . . . . .	3
2.3.3.3	Consulter le profil d'un ami suivi . . . . .	3
2.3.3.4	Consulter la wishlist d'une personne suivie . . . . .	3
2.3.3.5	Arrêter de suivre un utilisateur . . . . .	3
2.4	Diagramme de classe UML . . . . .	4
2.5	Diagrammes de séquence UML . . . . .	5
2.5.1	askFollow . . . . .	5
2.5.2	editArticle . . . . .	6
2.5.3	showFollowingProfil . . . . .	7
2.5.4	showWishList . . . . .	8
<b>3</b>	<b>Manuel d'utilisation</b>	<b>8</b>
3.1	Se connecter ou s'enregistrer . . . . .	8
3.2	Gestion de mon profil . . . . .	8
3.3	Gestion des wishlists . . . . .	9
3.4	Gestion des produits . . . . .	9
3.5	Gestion des follows . . . . .	9
3.6	Vue d'un profil suivi . . . . .	9
<b>4</b>	<b>Choix d'implémentation</b>	<b>10</b>
4.1	Utilisation de DAO . . . . .	10
4.2	Les comptes privés . . . . .	10
4.3	Les intérêts . . . . .	10
4.4	Pourquoi des follows ? . . . . .	10
<b>5</b>	<b>Organisation au sein du groupe</b>	<b>10</b>
<b>6</b>	<b>Informations utiles et problèmes rencontrés</b>	<b>11</b>
<b>7</b>	<b>Pistes d'amélioration</b>	<b>11</b>
<b>8</b>	<b>Conclusion</b>	<b>12</b>

# 1 Introduction

Voici notre rapport qui représente l'aboutissement des mois de travail en groupe afin de réaliser l'application "Wishlist". On retrouvera dans ce rapport, différents schémas conceptuels qui ont été modifiés par rapport à la première version et permettent de mieux visualiser notre application et sa structure. On y retrouve aussi nos récits utilisateurs, le manuel d'utilisation, une explication de nos choix d'implémentation, un bilan de la répartition des tâches dans le groupe et quelques informations utiles, par exemple les difficultés rencontrées.

Avant de se plonger dans ce rapport, il est important de donner une spécification du terme Follow qui est différent "d'amitié", l'amitié est mutuelle. Or, le fait de "Follow" n'est pas nécessairement mutuel et même une relation à sens unique, une personne peut follow un autre utilisateur sans que ce compte ne le suive en retour.

## 2 Diagrammes et récits utilisateurs

Voici une version corrigée et adaptée à l'implémentation de notre application des diagrammes : schéma ORM, schéma relationnel, diagramme de classes et de séquences. Ils sont tous accompagnés d'une explication pour les changements que nous avons apportés.

### 2.1 Schéma ORM

Sur cette version finale, on a ajouté l'attribut "Privacy" qui dicte si le statut de l'utilisateur est privé dans la table User, cette modification entre en jeu pour notre système de Follow. En effet, un utilisateur qui a pour son attribut "Privacy", la valeur "True", reçoit des demandes de "Follow". Ce dernier peut les refuser, ainsi, les autres utilisateurs n'ont pas accès à son statut. On a aussi ajouté l'attribut "interest rating" à la table Interests pour leur donner une valeur significative.

De plus, nous avons enlevé la relation ternaire entre Wishlist, Wishlist has products et product. Nous avons remplacé cette relation, par une relation binaire qui permet d'assurer la privatisation des products dans les wishlists de chaque utilisateur. Nous avons modifié l'attribut "shipping" de la table product, en l'attribut "purchased", car le terme est mieux adapté.

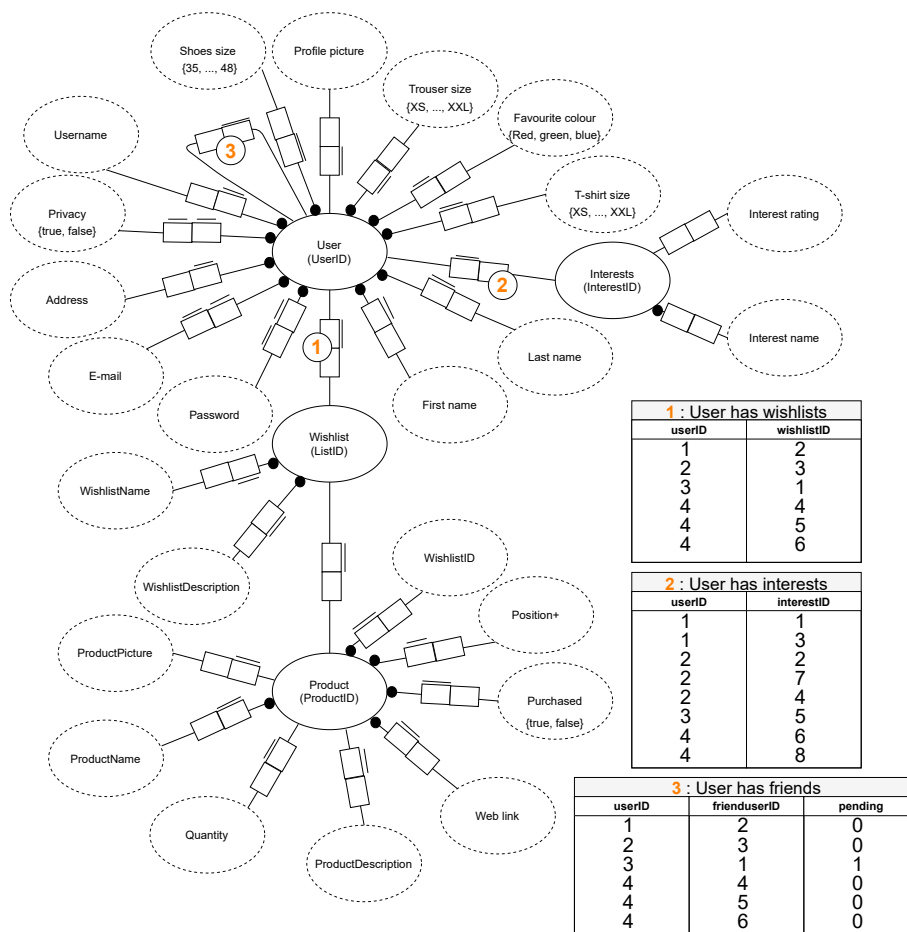


FIGURE 1 – Schéma ORM

## 2.2 Schéma relationnel

Voici notre schéma relationnel. Nous utilisons les ID de chaque élément dans la base de données pour pouvoir faire le lien d'appartenance et des liaisons. L'userID permet : de faire des tables de follows pour enregistrer quel utilisateur suit un autre, de mémoriser les intérêts qui sont les siens, et quelles wishlists lui appartiennent. L'ID des wishlist permet de mémoriser à laquelle un produit est lié.

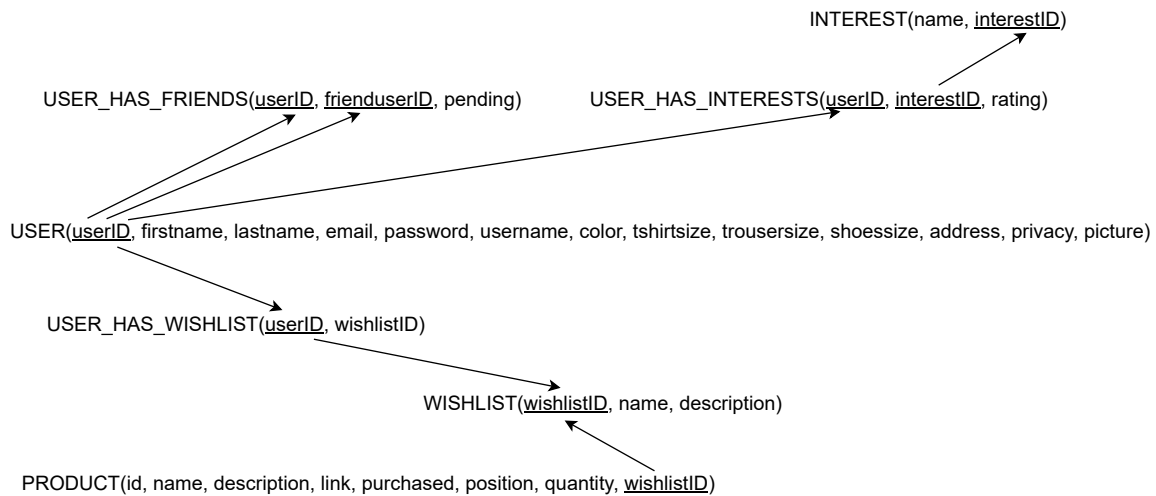


FIGURE 2 – Schéma relationnel

## 2.3 Récits utilisateurs

### 2.3.1 Gestion de compte

Ces récits d'utilisateurs sont sur la gestion du compte : créer le compte, en modifier les données, voire réinitialiser son mot de passe.

#### 2.3.1.1 Création d'un compte

EN TANT que nouvel utilisateur, AFIN d'adapter l'information fournie par l'application à mon nom, mes tailles et mes centres d'intérêt, JE VEUX encoder un profil personnalisé lors de ma première connexion à l'application.

Temps prévu : 4h.

#### 2.3.1.2 Modifier son profil utilisateur

AFIN de pouvoir modifier mon profil personnalisé, EN TANT qu'utilisateur actif de l'application, JE VEUX pouvoir changer via un menu dédié mes informations personnelles et mes centres d'intérêt, qui seront automatiquement prises en compte par l'application.

Temps prévu : 3h.

### 2.3.2 Gestion de Wishlist

Ces récits d'utilisateurs sont sur la gestion de ses wishlists : création et gestion des produits comme la modification des données de ceux-ci ou du lien qu'ils ont avec leur wishlist.

#### 2.3.2.1 Ajouter une WishList

EN TANT qu'utilisateur de l'application, JE VEUX pouvoir ajouter une WishList composée d'articles que j'enregistrerai via un menu dédié. Cette WishList sera caractérisée par un nom et une description, AFIN de partager à mes followers les articles qui m'intéressent et ce de manière organisée.

Temps prévu : 8h.

#### 2.3.2.2 Ajout d'un article à une WishList

EN TANT qu'utilisateur de l'application, JE VEUX pouvoir ajouter un article à une WishList existante, AFIN DE remplir ma WishList de mes derniers coups de coeur.

Temps prévu : 4h.

### **2.3.2.3 Modifier un de mes articles**

EN TANT qu'utilisateur de l'application, JE VEUX pouvoir modifier la description ou le lien internet de l'objet, AFIN DE tenir mes articles à jour par rapport à la situation actuelle des prix et des moyens de l'acheter.  
Temps prévu : 4h.

### **2.3.2.4 Modifier mes préférences dans une wishlist**

EN TANT qu'utilisateur de l'application, JE VEUX pouvoir modifier la quantité et l'ordre des produits dans la wishlist, AFIN D'informer mes amis de mon ordre de préférence parmi les produits et la quantité souhaitée pour ce produit dans cette wishlist.  
Temps prévu : 4h.

### **2.3.2.5 Supprimer un article d'une WishList**

EN TANT qu'utilisateur de l'application, JE VEUX pouvoir supprimer un article d'une WishList, AFIN DE ne plus l'afficher dans mes souhaits.  
Temps prévu : 4h.

### **2.3.2.6 Modifier l'état d'un article d'un follower**

EN TANT qu'utilisateur de l'application, JE VEUX pouvoir changer l'état d'achat d'un article dans la WishList d'un utilisateur que je follow, AFIN D'éviter à cet utilisateur de se voir commander plusieurs fois le même article.  
Temps prévu : 3h.

## **2.3.3 Gestion des follows**

Ces récits d'utilisateurs sont sur la gestion des follows : demander de suivre un autre compte, accepter la demande d'un autre utilisateur, pouvoir aller consulter le profil des personnes que l'on suit et voir leurs informations et finalement supprimer des personnes que l'on suit.

### **2.3.3.1 Follow un utilisateur**

EN TANT qu'utilisateur de l'application, AFIN DE suivre des utilisateurs et leurs WishLists, JE VEUX pouvoir suivre un utilisateur afin d'être au courant de ses articles et de ses préférences.  
Temps prévu : 2h.

### **2.3.3.2 Accepter une demande de follow**

EN TANT qu'utilisateur de l'application, AFIN DE pouvoir gérer qui peut voir mon profil et mes WishLists, pour ainsi garder une main sur ma vie privée, JE VEUX pouvoir accepter ou refuser une demande de follow en mettant mon compte privé.  
Temps prévu : 2h.

### **2.3.3.3 Consulter le profil d'un ami suivi**

EN TANT qu'utilisateur de l'application, JE VEUX pouvoir consulter les WishLists et les informations personnelles d'un ami, AFIN D'avoir des informations sur ses intérêts, ses tailles et ses WishLists.  
Temps prévu : 2h.

### **2.3.3.4 Consulter la wishlist d'une personne suivie**

EN TANT qu'utilisateur de l'application, JE VEUX pouvoir consulter en détail les WishLists de des personnes que je suis et les produits qu'elles contiennent avec leurs informations, AFIN D'avoir d'avoir plus de détails sur les produits qui lui feraient plaisir.  
Temps prévu : 2h.

### **2.3.3.5 Arrêter de suivre un utilisateur**

EN TANT qu'utilisateur de l'application, JE VEUX pouvoir unfollow un utilisateur de la liste des personnes que je suis, AFIN de ne plus être informé de ses préférences et wishlists.  
Temps prévu : 2h.

## 2.4 Diagramme de classe UML

Pour cette version finale, nous avons apporté quelques modifications par rapport au diagramme initial. Tout d'abord, nous avons ajouté l'attribut `interestRating` à la classe `Interest`. Ensuite, nous avons ajouté l'attribut `ID` à la classe `User`, pour pouvoir distinguer chaque utilisateur et y accéder facilement dans la database. De plus, nous avons ajouté l'attribut `"picture"` à la classe `User`, permettant à chaque utilisateur de posséder une photo de profil. En outre, pour les mêmes raisons de singularités, les attributs `wishlistID` et `user` ont été ajoutés à la classe `Wishlist`, dans le but de distinguer les `Wishlist` et voir à quel utilisateur ils appartiennent, ainsi que pour y avoir accès facilement dans la base de données. En ce qui concerne les méthodes des classes `Interest`, `User` et `Product`, les méthodes de ces classes ont été remplacés par des setters et getters, car ces derniers sont nécessaires et suffisant au fonctionnement. Nous avons aussi enlevé les méthodes obsolètes de la classe `Wishlist`. Il a aussi fallu ajouter l'attribut `productID` pour les mêmes raisons de singularités dans la classe `Product`. Pour finir, l'attribut `productRating` a été ajouté aussi dans la classe `product` pour que chaque produit puisse avoir une évaluation par l'utilisateur.

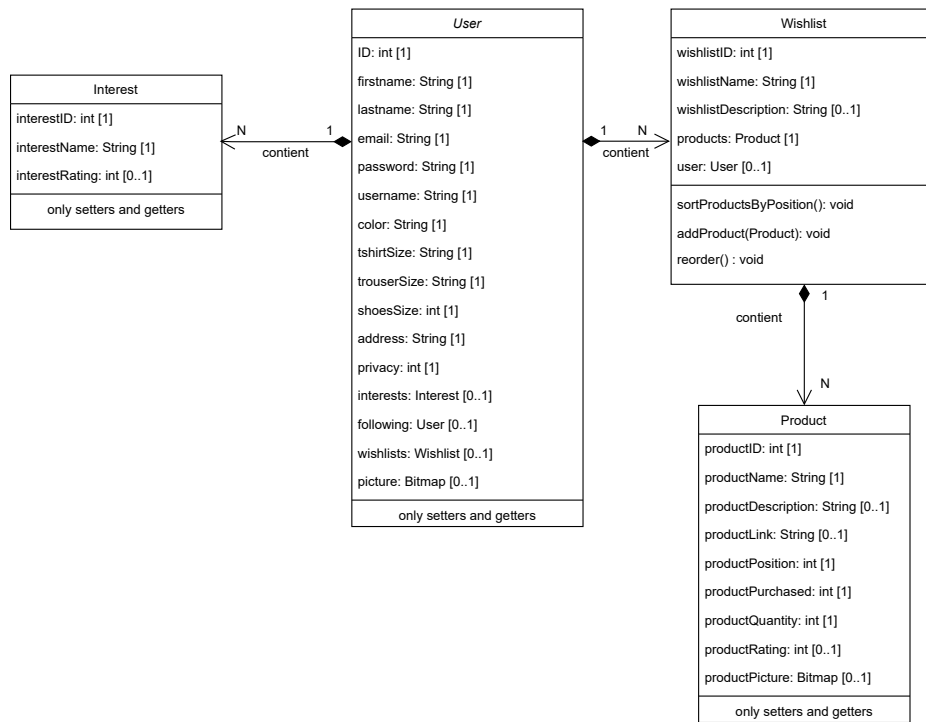


FIGURE 3 – Diagramme de classe UML

## 2.5 Diagrammes de séquence UML

### 2.5.1 askFollow

Nous avons enlevé le participant "DAO" de notre diagramme car celui-ci n'est pas adéquat dans les diagrammes de séquences. De plus suite à l'implémentation que nous avons choisie, le nouveau déroulement des actions est plus correct, nous regardons si l'utilisateur à suivre est un compte privé ou pas, proposons un message de confirmation de follow (ou demande de follow) adéquat à l'utilisateur et selon sa réponse nous confirmons le follow avec le pending correspondant à la privacy du compte concerné.

Ce diagramme de séquence correspond au récit utilisateur Follow un utilisateur.

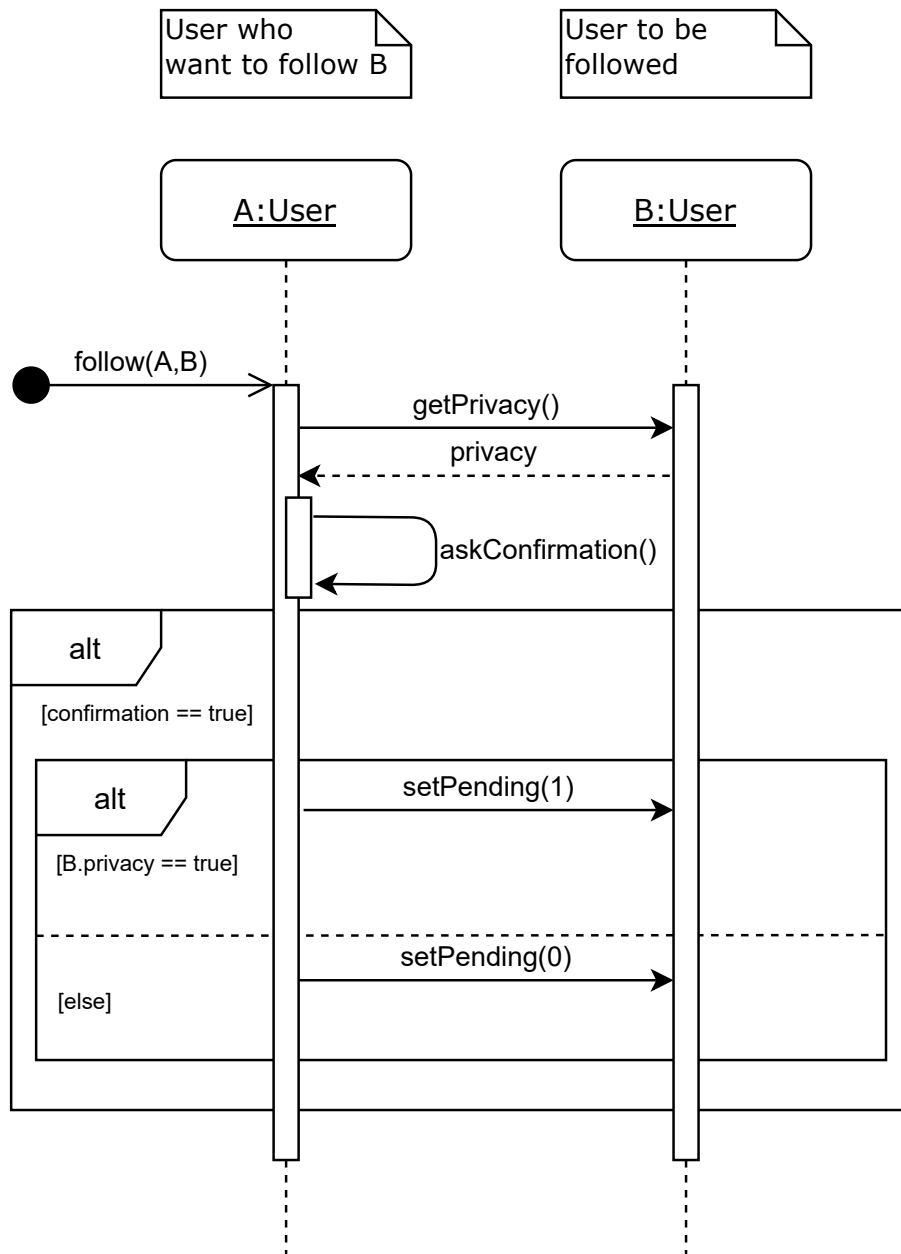


FIGURE 4 – Diagramme de séquence askFollow

### 2.5.2 editArticle

Lors de l'implémentation de l'édition d'un produit, nous nous sommes rendu compte que les choses étaient d'un point de vue diagramme, bien plus simple que ce que nous avons soumis la première fois. En effet nous connaissons déjà la wishlist et n'avons donc pas besoin de passer par l'utilisateur pour trouver le produit, raison pour laquelle celui-ci n'apparaît plus sur notre diagramme. Une fois le produit récupéré, nous laissons l'utilisateur apporter ses changements et puis nous updatons le produit.

Ce diagramme de séquence correspond au récit utilisateur Modifier un de mes articles.

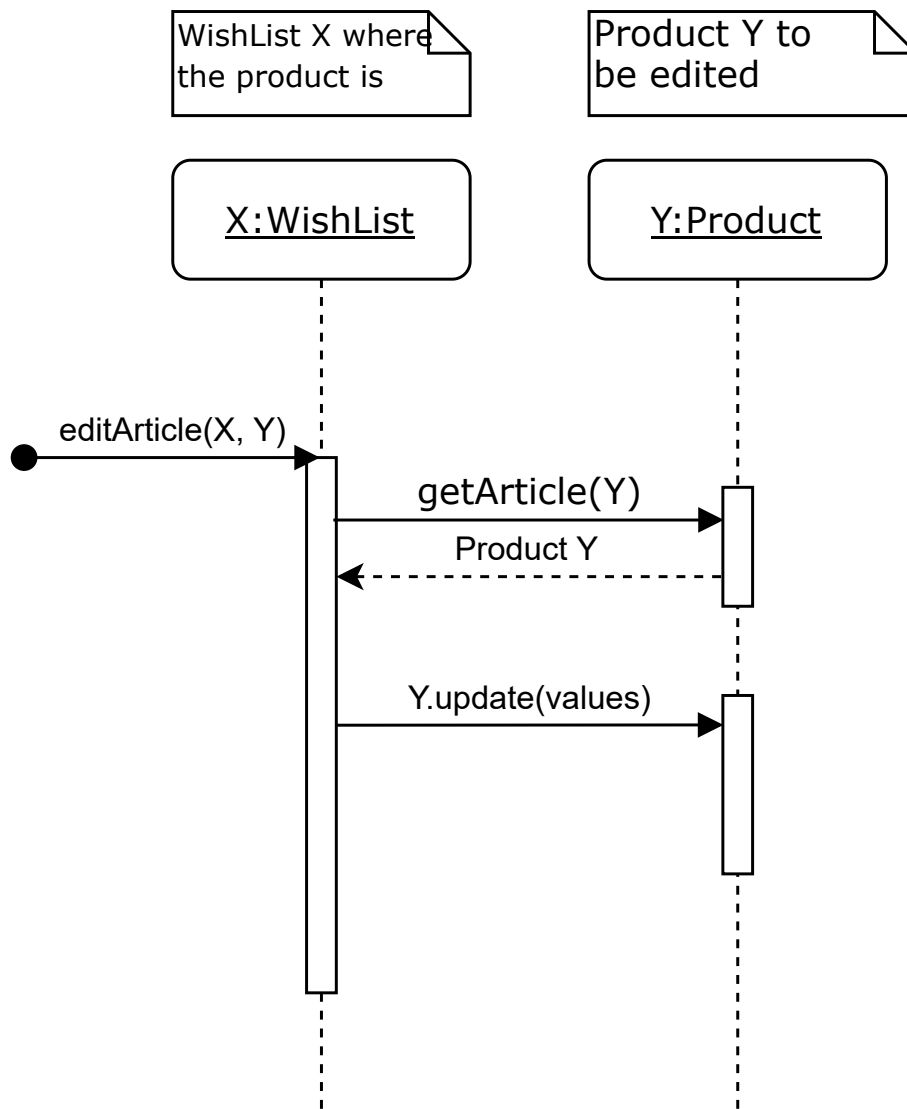


FIGURE 5 – Diagramme de séquence editArticle



### 2.5.3 showFollowingProfil

La consultation du profil d'une compte suivi se fait en deux temps, d'abord on accède à une vue des wishlists de l'utilisateur et puis ensuite on peut avoir une vue des informations détaillées. C'est pourquoi nous avons adapté le schéma en fonction. On récupère les wishlists et on les affiche et si l'utilisateur veut voir plus de détails sur les informations de l'utilisateur, on récupère ces infos et on les affiche. Par rapport à l'ancien diagramme nous pouvons donc constater une vue en deux temps au lieu d'un seul.

Ce diagramme de séquence correspond au récit utilisateur Consulter le profil d'un ami suivi.

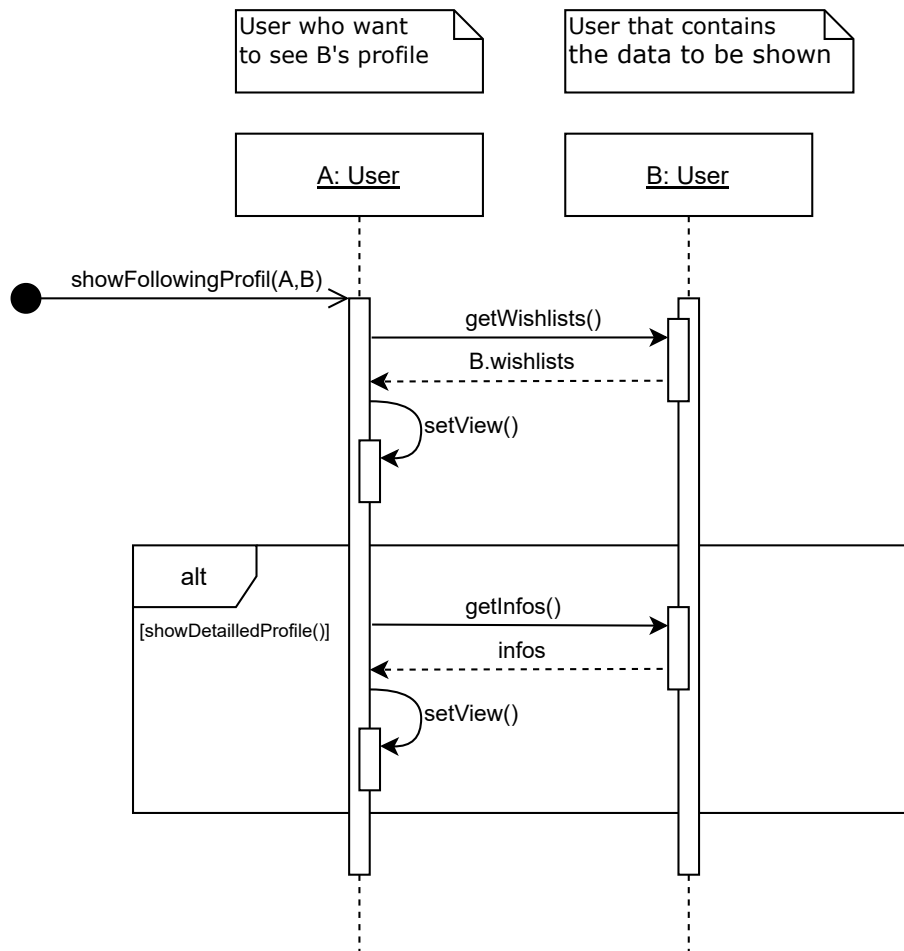


FIGURE 6 – Diagramme de séquence `showFollowingProfil`

#### 2.5.4 showWishList

Ce diagramme n'a pas beaucoup changé. Comme products est une variable appartenant à la wishlist, nous n'avons plus besoin de récupérer la liste et nous pouvons directement itérer sur la liste pour en afficher les produits en détail.

Ce diagramme de séquence correspond au récit utilisateur Consulter la wishlist d'une personne suivie.

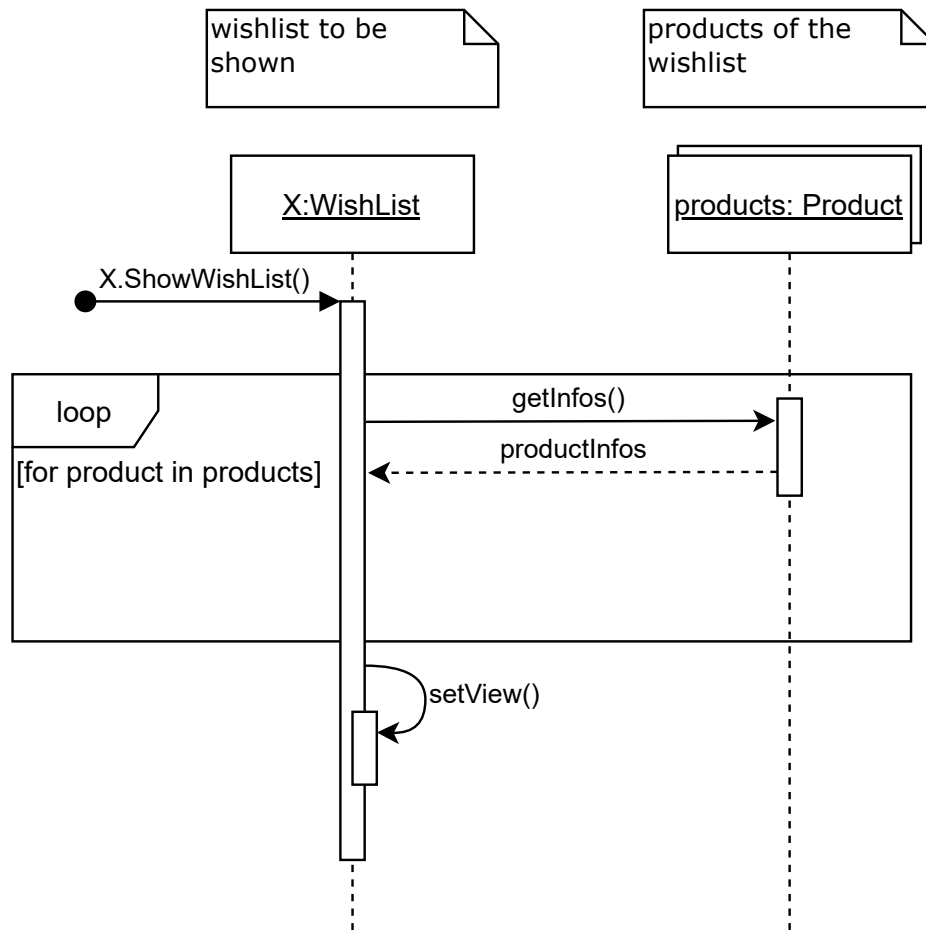


FIGURE 7 – Diagramme de séquence showWishList

### 3 Manuel d'utilisation

Le manuel d'utilisation sera divisé en plusieurs parties, selon les différents buts d'utilisations qu'un utilisateur peut avoir avec notre application. Il est structuré afin d'expliquer à un utilisateur novice, comment configurer son compte et faire ses premiers pas dans notre application.

#### 3.1 Se connecter ou s'enregistrer

Pour se connecter, rien de plus simple ! Sur l'écran de bienvenue sélectionnez le bouton LOG IN et une page de connexion classique s'affichent demandant de spécifier l'adresse mail et le mot de passe de votre compte.

Pour s'enregistrer dans l'application choisissez REGISTER, une première page vous invite à choisir un nom d'utilisateur un mot de passe qui est à confirmer une seconde fois, puis une adresse mail qui sera propre à votre compte et vous servira à vous connecter. Une fois que tout est bien complété, appuyez sur SUBMIT. Une seconde page s'affiche alors demandant plus de détails sur vous, remplissez là et définissez vos tailles, ces informations seront visibles par les gens qui vous suivent afin d'en savoir plus sur vous et vous offrir des cadeaux parfaits ! Quand vous avez fini, cliquez à nouveau SUBMIT et vous arriverez sur votre profil ! Vous pouvez remarquer 3 boutons tout en bas de l'application, ils vont vous permettre de naviguer facilement dans l'app.

#### 3.2 Gestion de mon profil

Votre adresse mail est définitive, mais votre username peut être modifié, il vous sert juste à vous démarquer des autres. Ainsi dans la vue de votre profil, vous pouvez modifier les données à votre guise, mais n'oubliez pas

de confirmer les changements avec le petit symbole de confirmation dans le coin supérieur gauche ! Vous avez aussi la possibilité d'ajouter une photo de profil en cliquant sur l'avatar par défaut.

Quatre autres interactions majeurs sont possibles depuis cet endroit. Commençons par mettre votre compte en privé, ceci est faisable en un tapotement sur le petit cadenas à droite de votre photo de profil, et vous voilà en privé. Ceci implique que vous aurez le contrôle sur toutes les demandes de follow vous concernant. Si un utilisateur vous ajoute, vous en serez notifié à votre arrivée sur l'application. Voilà donc la deuxième interaction, vous pourrez alors voir les demandes de follow en cliquant sur la petite cloche de notification à gauche de la photo de profil et choisir depuis cette vue si vous voulez accepter ou refuser la demande. Ensuite vous avez également la possibilité de définir vos intérêts, cliquer sur le bouton CHOOSE MY INTERESTS et une liste des intérêts vous sera affichée, cochez ceux qui vous correspondent et ils seront mis à jour. La dernière interaction est le bouton tout en haut à gauche, il vous permet de quitter l'application, mais ne nous quittez pas si vite nous avons encore plein de choses à vous apprendre !

### 3.3 Gestion des wishlists

Votre compte est maintenant parfaitement réglé, il est temps de créer votre première wishlist. Allez à la vue dédiée si ce n'est déjà fait et cliquer sur le petit plus pour en ajouter une. Un dialogue s'ouvre à vous et vous invite à choisir un nom et une description pour la liste, complétez et soumettez et voilà votre première wishlist est créée, rien de plus facile n'est-ce pas ? Essayons maintenant de la modifier. Cliquez de dessus, une vue s'ouvre. Sélectionnez le bouton dans le coin inférieur droit et vous aurez trois nouvelles interactions. Le crayon vous permet d'éditer le nom et la description de la wishlist. C'est bien tout ça, mais c'est vide ! Cliquez donc sur le petit plus pour ajouter un premier produit à la wishlist, remplissez le nom, la quantité et un lien internet permettant de trouver votre produit et confirmez. Le tour est joué, après votre première wishlist vous venez de créer votre premier produit ! Vous pouvez le supprimer ou cliquer dessus pour pouvoir en savoir plus. Mais ajoutez quelques produits, vous allez découvrir quelque chose de vraiment sympa. Quand c'est fait et que vous êtes dans la vue vous présentant la liste des produits de votre wishlist, appuyez longtemps sur un produit, vous allez ressentir une vibration puis être libre de déplacer l'objet dans la liste. Ceci vous permet d'organiser l'ordre de votre wishlist comme vous le souhaitez comme par exemple mettre ceux que vous voulez le plus en premier, mais n'oubliez pas de sélectionner l'icône avec les 3 barres et d'appliquer le changement avec le petit V. Une dernière chose, vos amis pourront le faire pour vous une fois qu'ils auront acheté le produit, mais vous pouvez cocher les produits que vous avez reçus afin de le montrer, cochez simplement la case check.

### 3.4 Gestion des produits

Vous pouvez les ajouter depuis la gestion des wishlists, mais si vous voulez apporter un changement à un produit cliquez dessus, tout d'abord il va s'afficher en détail. Depuis ce détail vous pouvez donner un rating au produit pour dire au combien vous l'appréciez par rapport aux autres. Vous pouvez de la même manière que pour mettre une photo de profil, cliquez sur l'image et en choisir une. Pour modifier en détail le produit, préférez le petit crayon, faites vos changements et sauvez. Vous êtes maintenant un pro des wishlists ! Mais attendez... Les gens peuvent vous suivre, pourquoi ne pas aller faire un tour pour trouver des personnes à suivre vous aussi !

### 3.5 Gestion des follows

Commençons par ajouter un ami, tapotez le petit plus, un dialogue s'ouvre et vous permet d'entrer les termes de votre recherche. Comme la recherche se fait via les noms d'utilisateurs, tapez-en un. Si vous ne savez pas qui chercher dans l'application, n'hésitez pas à chercher après le compte de démo, son nom d'utilisateur est usernameDemo ou pour voir tous les utilisateurs que vous ne suivez pas, faites une recherche vide. La liste des usernames correspondant à votre recherche s'affiche, pour ajouter un utilisateur, choisissez le +. Le dialogue qui s'ouvre vous demandera une confirmation de follow et si le compte est privé, vous l'indiquera. Faites votre choix. et revenez à la vue de vos follows, si l'utilisateur que vous avez ajouté est privé, ne vous inquiétez pas de ne rien voir de nouveau, il faudra d'abord qu'il accepte votre demande, sinon, c'est super vous avez votre premier follow dans la liste.

### 3.6 Vue d'un profil suivi

J'imagine que vous trépignez d'impatience à l'idée de découvrir ce que la personne que vous suivez souhaiterait recevoir pour son anniversaire ! Cliquez simplement sur son profil et vous pourrez voir l'entièreté de ses wishlists, choisissez-en une pour voir les produits qu'elle contient, et choisissez enfin un produit pour voir ses détails. C'est le cadeau que vous voulez lui offrir ? Parfait, procédez-en un clic à son achat en cliquant sur le lien web. Une fois que c'est fait n'hésitez pas à le signaler à l'utilisateur et les autres personnes qui le suivent en checkant la box pour dire que vous l'avez acheté !

Vous connaissez maintenant mieux l'application que quiconque ! Merci d'avoir rejoint notre communauté et n'hésitez pas à inviter vos amis pour vous faire de super cadeaux.

## 4 Choix d'implémentation

### 4.1 Utilisation de DAO

Afin de faciliter l'accès et le traitement des données, mais aussi pour conserver une certaine clarté dans notre application, nous avons décidé de découper l'accès aux données SQLite en différentes classe DAO pour chacune des classes Backend qui nous utilisons.

### 4.2 Les comptes privés

L'ajout de comptes privés nous a semblé une évidence afin d'offrir un certain contrôle de l'utilisation de l'application à notre utilisateur et un respect de l'affichage de ses données. S'il active cette option, il peut choisir lui-même qui peut le suivre et ne pas le suivre et ainsi accéder à ses données. En effet le seul moyen d'accéder aux données d'un utilisateur est de le faire depuis la liste des personnes que l'on suit. L'utilisateur garde donc un contrôle total sur ce qu'il ajoute comme informations, souvent très personnelles dans notre app.

### 4.3 Les intérêts

Nous avons implémenté des intérêts dans notre application. Ceux-ci sont actuellement fixes et chaque utilisateur peut choisir ceux qui lui correspondent. Un rating propre à l'utilisateur est implémenté dans toute l'application, mais n'a pas été implémenté dans l'utilisation de l'application. Le projet qui est lié aux intérêts est en fait de grande ampleur et le temps qui nous a été laissé ne permettait pas de le concrétiser pleinement. Ces intérêts pourraient, une fois implémentés, être utilisés afin de suggérer des produits qui sont appréciés par des utilisateurs ayant les mêmes centre d'intérêts que lui. Ceci ouvre beaucoup de portes d'amélioration pour notre application et lui donne un bon potentiel. Grâce à cette implémentation, on peut également facilement suggérer des personnes à suivre ayant les mêmes centres d'intérêts pour, pourquoi pas, partager ses passions avec eux et pouvoir suggérer des ajouts de produits dans ses listes de souhait.

### 4.4 Pourquoi des follows ?

Un de nos choix d'implémentation est le fait de choisir le "Follow" (lien entre utilisateurs à sens unique). Nous avons fait ce choix, car nous jugions plus utile d'utiliser ce type d'amitié pour notre application, afin de pouvoir aller offrir un cadeau. Il est implémenté comme un attribut de la classe User et comme étant une classe à part entière dans la base de données. En effet, cela nous semblait plus simple à gérer du côté de la base de données, ainsi que du côté des classes.

## 5 Organisation au sein du groupe

Voici 3 tableaux qui reprennent la participation active de chaque membre pour chaque partie :

FIGURE 8 – Rapports

Participants	Mockups et dépendance	Lexique	Faits élémentaires	ORM	Schéma relationnel	Users stories
Nathan		X	X			X
Corentin Lingier						
Corentin Lengele	X			X		
Gwendal			X			
Jules		X		X	X	X
William			X	X		

FIGURE 9 – Rapports

Participants	Diag. de classe	Diag. de séquence	Choix conceptions	Correction des schémas	Écriture des rapports
Nathan		X	X		
Corentin Lingier					
Corentin Lengele					
Gwendal		X		X	
Jules	X	X	X	X	X
William					X

FIGURE 10 – Conception de l'application (avec G pour gestion, A pour affichage et BDD pour base de données)

Participants	Login et register	follows	wishlists	produits	profil user	BDD et DAO
Nathan	G.A.	G	G	G	G	G
Corentin Lingier						G
Corentin Lengele	A	A	G.A	G.A		
Gwendal	G.A.	A	A	G.A	G.A	
Jules		G.A			G.A	
William						G

## 6 Informations utiles et problèmes rencontrés

S'il fallait relever les principaux problèmes rencontrés, nous avons tout d'abord eu du mal à nous organiser au niveau de la répartition des tâches ainsi qu'une sous estimation du temps nécessaire au développement du programme, ce qui a engendré des ralentissements à l'écriture conjointe du code et du rapport de manière efficace. De plus, nous avons rencontré un problème avec la DAO, ce qui a arrêté l'avancement du projet, en effet nous avons utilisé une mauvaise librairie, n'exploitant pas les avantages de la programmation Android, une fois cela implémenté correctement, nous avons avancé efficacement construit étape par étape l'application en coopération efficace.

## 7 Pistes d'amélioration

La partie qui va suivre va rassembler l'ensemble des idées qu'on prévoyait d'implémenter mais qui, par manque de temps (pour les raisons citées plus haut) nous n'avons pas eu le temps de réfléchir à leur implémentation.

Pour commencer, nous avons prévu de mettre en place un système de vérification d'adresse mail via un mécanisme de mailto, permettant de demander une confirmation de l'adresse-mail par le nouvel utilisateur. Ce mécanisme se serait basé sur l'envoi d'un code généré par le programme qui serait issu d'une formule prenant l'ID de l'user en paramètre, ceci permettrait donc de vérifier si le code mis dans l'application est bien celui envoyé par mail et ainsi validé l'adresse-mail entrée.

Ce système permettrait également de permettre à l'utilisateur de réinitialiser son mot de passe à l'aide de ce même principe de code issu d'une formule.

Ensuite, une autre fonctionnalité prévue initialement était le fait de permettre à un utilisateur de pouvoir ajouter les intérêts qu'il veut pour ainsi rendre l'application entièrement ouverte à l'utilisateur. Cette fonctionnalité impliquait trop de travail au vu du temps limité qu'il nous restait, nous avons donc décidé de mettre de côté cette fonctionnalité pour privilégier la bonne qualité des autres fonctionnalités de l'application.

Pour continuer, nous avons également imaginé l'implémentation d'un mécanisme de transfert vers Amazon, pour qu'un utilisateur puisse consulter très facilement un objet via le lien ou via une recherche Amazon de l'objet sur base du nom.

Cette interface permettrait donc de rendre la consultation de l'objet d'une personne qu'on suit très ergonomique.

Finalement, par soucis de la protection de la vie privée de nos utilisateurs, nous avons prévu de sécuriser l'entièreté de notre application via un hachage des mots de passe et des userID de la table User\_has\_wishList, afin de garder privée à la fois le mot de passe et les wishList d'un utilisateur.

## 8 Conclusion

Pour conclure, si nous avions commencé le projet par la conception Android sans passer par le travail sur les différents schémas conceptuels, nous n'aurions sûrement pas pu aboutir à un résultat satisfaisant. En effet, la séparation du projet en plusieurs étapes, nous a permis de nous structurer, faire mûrir nos idées et de nous rendre compte de l'importance de ces étapes pour aboutir à une bonne application. Si notre application était comparable à une horloge, alors chacune des étapes du projet est un engrenage qui participe au bon fonctionnement de la machine.