

RAPPORT 2 : WHISLIST



- Groupe 4.3 -

DE SURAYŸ Philippine(11091800), DEMBLON Antoine(15361800),
DESIDE Guillaume(39731800), GIOT Adrien(23121800),
SWIERZEWSKI Cezary(11141800)



École Polytechnique de Louvain
LSINF1225 : Conception orientée objet & Gestion de
données

Professeur : Kim Mens
Assistant : Raziel Carvajal-Gomez
Année académique 2019-2020

Table des matières

1	Introduction	2
2	Récits utilisateurs	2
2.1	Gestion utilisateur	2
2.2	Gestion wishlist	2
2.3	Gestion groupe	3
2.4	Gestion amitié	4
3	Diagramme de classes UML	4
4	Choix de conception	6
4.1	Récits utilisateurs	6
4.2	Diagramme de classes UML	6
4.3	Diagrammes de séquences UML	6
4.3.1	Obtenir le prix total d'une wishlist	7
4.3.2	Obtenir l'ensemble des listes publiques d'un utilisateur	7
4.3.3	Ajouter un ami	7
4.3.4	Chercher un objet dans une wishlist	7
4.4	Autres remarques	7
5	Diagrammes de séquences UML	8
5.1	Obtenir le prix total d'une wishlist	8
5.2	Obtenir l'ensemble des listes publiques d'un utilisateur	9
5.3	Ajouter un ami	10
5.4	Chercher un objet dans une wishlist	11
6	Conclusion	11

1 Introduction

Après avoir réfléchi à la base de données, il était temps de se concentrer sur ce qui constituera notre travail final, à savoir l'application en elle-même. Mais avant de se lancer dans le code à l'aveugle, une autre étape est nécessaire. Cette étape va nous permettre de conceptualiser l'application sans aborder le code en lui-même.

Dans ce second rapport, nous allons vous présenter successivement nos récits utilisateurs, notre diagramme de classes UML, ainsi que quelques diagrammes de séquences représentatif de notre application pour finir par une conclusion. Nous avons veillé à la cohérence entre ces sections, dont les objets respectifs visent à la conception d'une et une seule application.

2 Récits utilisateurs

Dans cette première partie du rapport, vous trouverez les récits utilisateurs qui caractérisent notre application. Ils se divisent en 4 parties : gestion utilisateur, gestion wishlist, gestion des groupes et gestion amitié. Pour chacun des récits utilisateurs, le nombre de jours estimés a été écrit en dessous.

2.1 Gestion utilisateur

- Afin de pouvoir accéder à l'application, en tant qu'utilisateur, lors du démarrage de l'application, je dois pouvoir m'y connecter ou créer un compte.
nombre de jours estimés :7
- Afin de modifier mes informations personnelles, en tant qu'utilisateur, je dois avoir accès à une interface qui me permet de les modifier.
nombre de jours estimés :3
- Afin de supprimer mon compte, en tant qu'utilisateur, je dois avoir accès à un bouton permettant d'exécuter cette tâche.
nombre de jours estimés :1

2.2 Gestion wishlist

- Afin de créer une liste de souhait, en tant qu'utilisateur, je dois pouvoir créer une liste de souhait.
nombre de jours estimés :5
- En tant qu'utilisateur, afin de mettre à jour les informations d'un objet d'une de mes wishlists, je dois pouvoir avoir accès aux informations de l'objet et

les modifier.

nombre de jours estimés :2

- En tant qu'utilisateur, afin de compléter/modifier ma wishlist, je dois être capable d'ajouter et de supprimer des objets à une liste

nombre de jours estimés :3

- En tant qu'utilisateur, afin de mettre à jour une de mes wishlists, je dois pouvoir modifier les spécifications d'une liste quand j'appuie sur le bouton edit.

nombre de jours estimés :2

- En tant qu'utilisateur, afin de retrouver un objet grâce à son nom, je dois pouvoir entrer son nom dans une barre de recherche.

nombre de jours estimés :2

- En tant qu'utilisateur, afin de savoir toutes les wishlists auquel j'ai accès, je dois pouvoir obtenir une liste de l'ensemble des listes de souhaits auquel j'ai accès en allant dans l'interface wishlist.

nombre de jours estimés :3

- En tant qu'utilisateur, afin de savoir quel objet est le plus/moins cher dans une wishlist, je dois pouvoir trier les objets selon certains critères (prix, note, nom).

nombre de jours estimés :4

- En tant qu'utilisateur, afin de savoir si l'ensemble de la wishlist rentre dans mon budget, je dois pouvoir obtenir le prix total d'une wishlist.

nombre de jours estimés :2

- En tant qu'utilisateur, afin de savoir ce qu'un ami veut, je dois pouvoir consulter ses wishlists publiques.

nombre de jours estimés :2

2.3 Gestion groupe

- En tant qu'utilisateur, afin de regrouper plusieurs amis pour partager des wishlists, je dois pouvoir créer un groupe.

nombre de jours estimés :6

- En tant que membre d'un groupe, afin de savoir ce que désire un autre membre d'un groupe, je dois avoir accès aux wishlists publiques des autres membres.

nombre de jours estimés :3

- En tant qu'utilisateur, afin de savoir dans quels groupes je suis, je dois pouvoir obtenir une liste de tous les groupes dans lesquels je suis membre en allant dans mon onglet groupe.

nombre de jours estimés :3

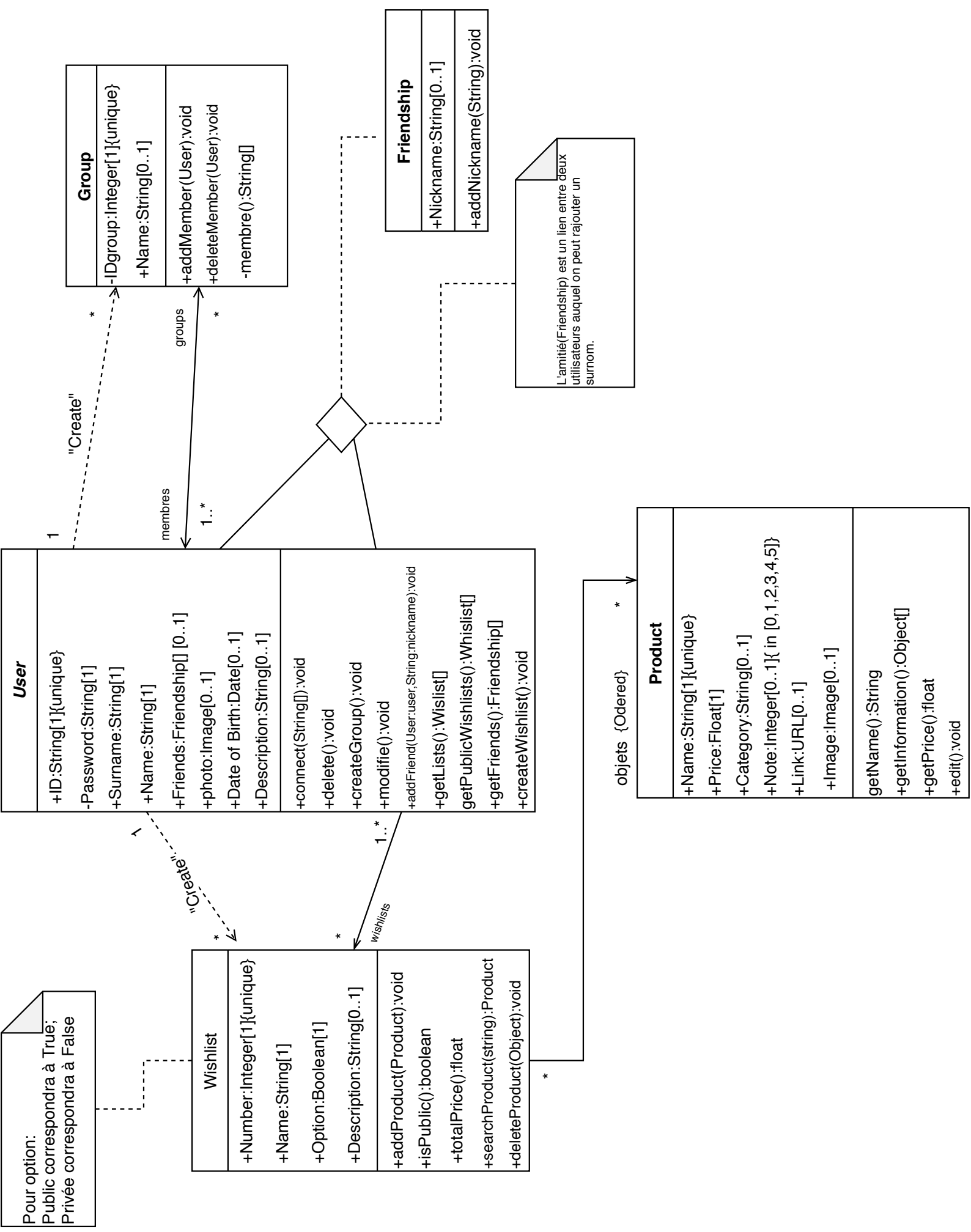
- En tant que membre d'un groupe, afin de supprimer un membre(dont moi-même), je dois pouvoir avoir accès aux membres et choisir lequel ne fait plus partie du groupe.
nombre de jours estimés :2

2.4 Gestion amitié

- En tant qu'utilisateur, afin d'augmenter mon cercle d'amitié ou le diminuer, je dois être capable d'ajouter et supprimer des amis.
nombre de jours estimés :6
- En tant qu'utilisateur, afin d'obtenir des informations sur mon ami, je dois pouvoir consulter son profil avec toutes ses informations.
nombre de jours estimés :2
- En tant qu'utilisateur, afin de savoir avec qui je suis amis, je dois pouvoir obtenir une liste précise de tous mes amis dans l'onglet "amis de l'application".
nombre de jours estimés :2

3 Diagramme de classes UML

Dans cette section, nous allons vous présenter le diagramme de classe de notre application. Celui-ci nous permet de visualiser les différentes classes que nous devons créer et d'observer les liens et contraintes qui les relient.



4 Choix de conception

4.1 Récits utilisateurs

En ce qui concerne les récits utilisateurs, nous avons essayé d'en trouver le maximum. Nous les avons regroupé selon 4 catégories qui nous semblaient essentiels. D'autres récits utilisateurs pourront bien sûr être trouvés lors de la programmation et au fil de la vie de l'application.

Pour les jours estimés, nous avons bien sûr légèrement surestimé pour pallier à certaines erreurs et/ou problèmes lors de l'implémentation. Ainsi nous avons un nombre de jours qui rentre dans les délais en divisant le travail et qui nous permettra d'avoir une marge en cas de soucis.

La division du travail reste à faire. Mais actuellement, on aurait ± 11 jours de travail par personne.

4.2 Diagramme de classes UML

Comme pour notre schéma ORM, l'élément central reste donc l'utilisateur (user). C'est par cette classe que nous créons et stockons les wishlists et les groupes. On n'oublie pas qu'un groupe doit avoir une liste avec tous les identifiants des utilisateurs membres de ce groupe. On garde toujours l'idée qu'une liste peut être utilisée par différents utilisateurs comme dans le cas où elle est liée à un groupe.

Comme on peut le constater dans le diagramme, le seul attribut privé est celui du mot de passe (Password) car il n'est utilisable que par le détenteur du compte associé.

4.3 Diagrammes de séquences UML

Nos diagrammes de classes ne prennent qu'en compte les fonctions du diagramme de classes de la section 3. Nous n'avons pas pris en compte les fonctions qui devront composer notre couche intermédiaire avec la base de données.

Nos diagrammes de séquences présentés dans la section suivante, restent très basiques car nous avons peu d'actions qui demandent une grande complexité. La majorité des autres actions pour lesquels le diagramme de séquences n'a pas été réalisé, seront très largement basées sur les quatre diagrammes qui se trouvent dans la section 4 et qui sont expliquées ci-dessous.

4.3.1 Obtenir le prix total d'une wishlist

Pour une wishlist donnée, on va additionner le prix de chaque objet de la liste afin de pouvoir retourner le prix total. Cette méthode de notre application permettra par la suite de pouvoir lui rajouter d'autres fonctionnalités telles que la possibilité d'imposer un prix maximum à une liste.

4.3.2 Obtenir l'ensemble des listes publiques d'un utilisateur

On retourne une collection d'éléments, chacun d'eux correspondant à une wishlist (ils constituent un identifiant de la liste).

4.3.3 Ajouter un ami

Les autres méthodes qui permettront de créer les différents objets (groupes, wishlists) vont se baser très fortement sur ce diagramme de séquences à un ou deux détails près. Celui montre bien que l'objet créé est stocké dans une liste lors de l'exécution du code.

Il restera à définir les méthodes de la couche intermédiaire qui permettront de stocker et de obtenir les données en ce qui concerne les amis.

4.3.4 Chercher un objet dans une wishlist

En ce qui concerne la recherche d'un objet, cette partie de notre application nous semble essentiel pour avoir une application efficace . Ce diagramme de séquences pourra aussi être transformé pour s'appliquer pour les wishlists, les groupes et les amis.

Le diagramme de séquences présent dans le rapport reprend une procédure de recherche naïve. Lors de la confection de l'application, on pourrait imaginer une recherche dichotomique pour pallier à ce problème. Et dans le meilleur des cas, on pourrait avoir une recherche qui permet d'avoir tous les objets qui ont un nom assez semblable et donc utiliser des comparateurs. Il restera à décider de l'intervalle d'erreur qui nous semble assez correct.

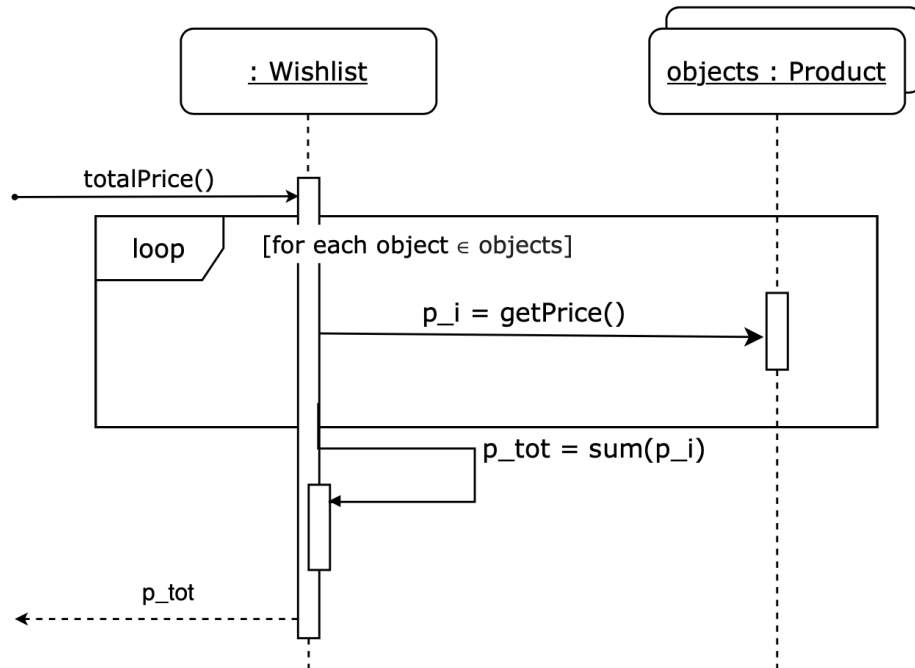
4.4 Autres remarques

La partie, qui sera plus délicate, sera lors de la conception de la couche intermédiaire. Il sera nécessaire d'optimiser le nombre de classes et de préciser clairement l'accès de chacune dans la base de données. Nos diagrammes de séquences restent basiques car cette couche intermédiaire n'est pas visible dans la partie du code.

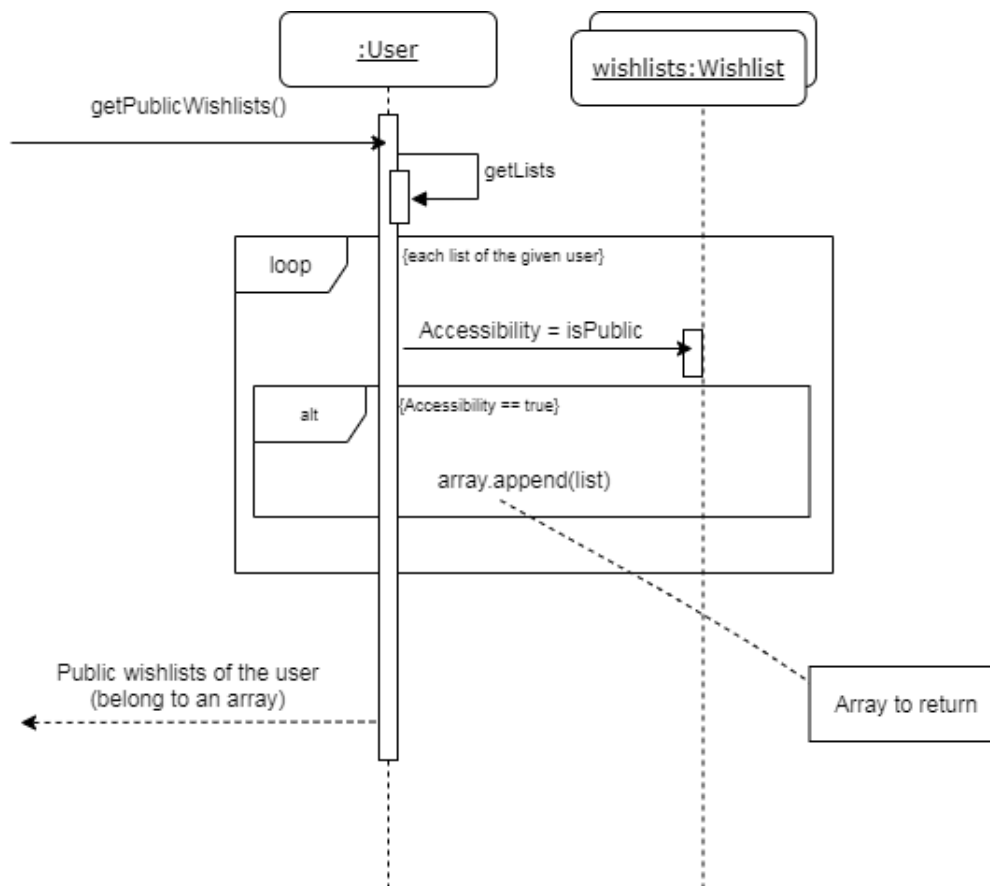
5 Diagrammes de séquences UML

Vous trouverez ci-dessous 4 diagrammes de séquences UML expliqués dans la section précédente. Ceux-ci sont représentatifs de notre applications et des éléments qui la différencient des autres.

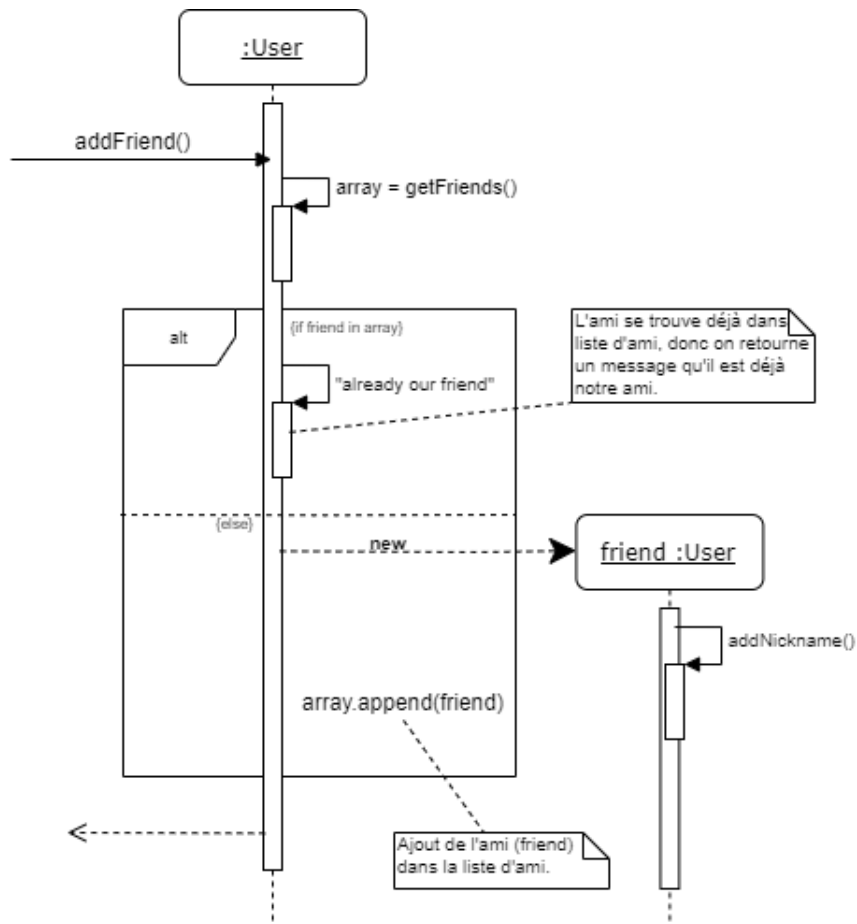
5.1 Obtenir le prix total d'une wishlist



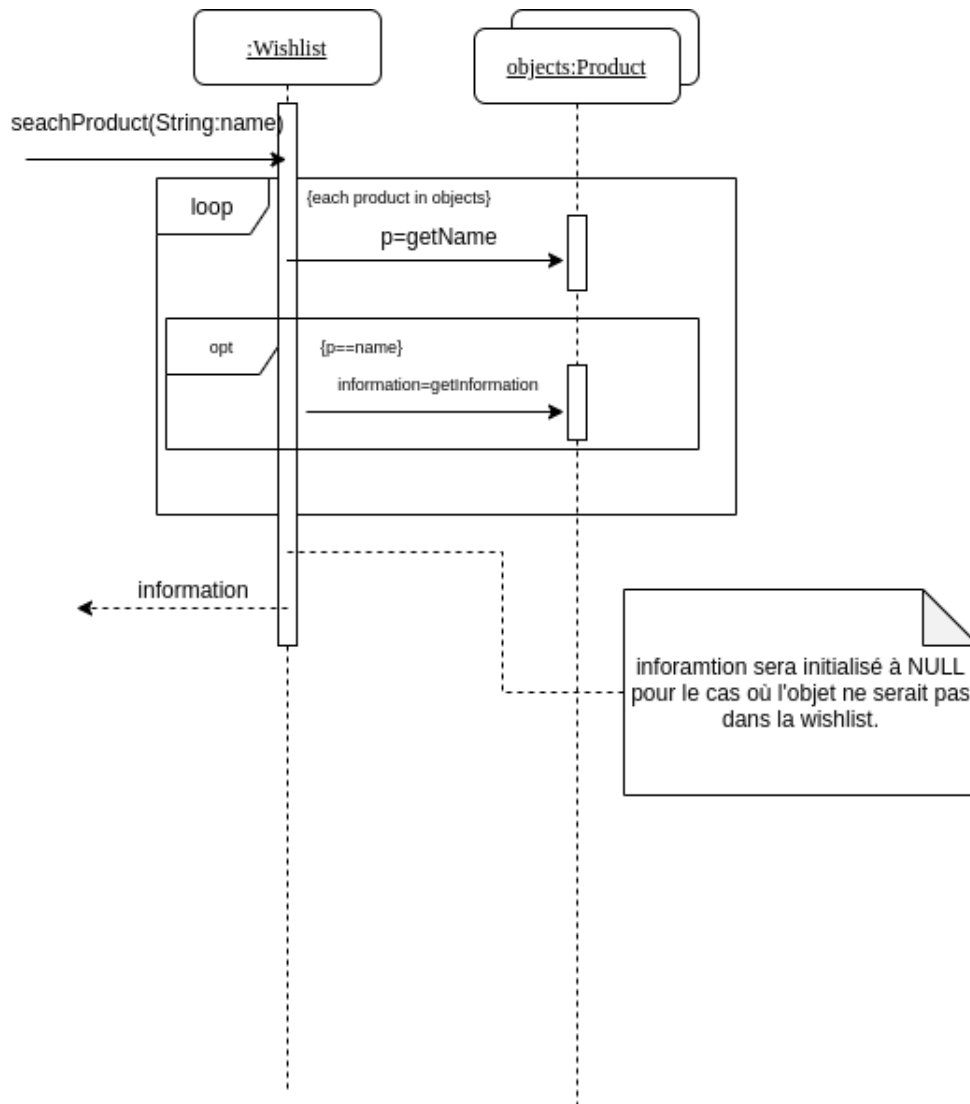
5.2 Obtenir l'ensemble des listes publiques d'un utilisateur



5.3 Ajouter un ami



5.4 Chercher un objet dans une wishlist



6 Conclusion

A l'issue de cette deuxième partie de notre projet, nous avons une idée très claire et précise de notre future application et de ses fonctionnalités. En effet, nous avons pu travailler sur sa conception comme développé dans ce rapport. Nous avons aussi estimé la durée nécessaire pour la développer, ce qui nous permettra de pouvoir mieux nous organiser.

La prochaine partie consistera à la programmer et, pour ce faire, à maîtriser

Android Studio. Nous avons d'ailleurs déjà pu nous pencher dessus et commencer à coder la première interface. Ce sera l'étape cruciale de notre projet, l'aboutissement de notre travail.