

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Abderrahmane Mira - Béjaïa



Faculté de Science exacte  
Département Informatique

**PROJET FIN DE CYCLE**

En vue de l'obtention du diplôme de  
**MASTER PROFESSIONNEL**

En Génie Logiciel

**Thème**

---

**Application web de vente de vêtements d'occasion  
avec *recommandation, chatbot et recherche  
visuelle.***

---

***Réalisé par :***

IDIRI THINHINANE  
BENADJAOUD LYDIA

***Encadrés par :***

DR. BENNAI Yani-Athmane

Soutenu le 01 Juillet 2025, devant le jury composé de :

Président :	M. BOUZIDI Zair,	Université de Béjaïa
Examineur :	M. MIR Foudil,	Université de Béjaïa
Examineur :	M. SLIMANI Hachem,	Université de Béjaïa
Examineur :	M. GOUDJIL Slimane,	Université de Béjaïa

Année universitaire 2024/2025



---

◇ REMERCIEMENTS ◇

---

« *La gratitude transforme ce que nous avons en suffisance* »

Nous souhaitons remercier chaleureusement toutes les personnes qui ont contribué à la réalisation de ce mémoire.

**Tout d’abord**, nos remerciements les plus sincères vont à ALLAH pour nous avoir guidés et donné la force nécessaire pour mener à bien ce projet.

Nous exprimons également notre **profonde gratitude** à Monsieur **BENNAI Yani-Athmane**, notre encadrant, pour sa bienveillance, son expertise et son accompagnement précieux tout au long de ce travail.

Un grand **merci** à nos **familles**, qui ont été un soutien sans faille, remplis de patience, d’amour et de sacrifices. Cette réussite leur est dédiée, avec toute notre reconnaissance.

Nous remercions aussi les membres du jury pour leurs observations pertinentes et leur expertise enrichissante.

Enfin, une **reconnaissance particulière** à nos chers **Ilyes SEDKAOUI**, **Naouel IDIRI** et **Djamel IDIRI**, qui ont été là à chaque étape, offrant leur soutien, leur amitié et leurs précieux conseils.

---

◇ *À tous ceux qui ont cru en nous, merci du fond du cœur* ◇

---

---

## ◇ DÉDICACES ◇

---

Je dédie ce projet à **mes parents**, pour leur amour sans fin, leur soutien Éternel et leur patience infinie. Vous êtes mes racines et ma lumière, toujours présents dans les moments de doute comme dans ceux de joie. Sans vous, je n'aurais jamais eu le courage de croire en moi ni l'envie d'avancer. Vous êtes mon guide, mon port d'attache, et ce projet est le reflet de tout ce que vous m'avez offert.

À **mon frère Yudas**, pour ton soutien constant, ta force tranquille et ton affection. Tu es une présence rassurante, et ta confiance en moi m'a donné l'énergie nécessaire pour aller de l'avant. Je suis fier de t'avoir à mes côtés à chaque étape de ce chemin.

À **Maylis**, tu es bien plus qu'une petite cousine pour moi. Ton amour pur et ta lumière m'ont apporté une joie infinie, et chaque instant passé à tes côtés est un véritable cadeau. Tu occupes une place spéciale dans mon cœur.

À **mes grands-parents** maternels, partis trop tôt, mais toujours présents dans mes pensées. À mes grands-parents paternels, qui me soutiennent chaque jour avec leur amour et leur sagesse.

À **ma meilleure amie, ma sœur et ma binôme, Thinhinane**, pour ta fidélité sans faille, ton soutien indéfectible et ton énergie incroyable. Tu es bien plus qu'une amie, tu es une sœur de cœur. Ce projet n'aurait pas pris la même forme sans toi. Ton rire, ton amour et ton implication ont été des moteurs essentiels dans cette aventure. Merci d'être là, toujours.

À **mes meilleures amies, Kenza et Zouza**, sans vous, mon parcours aurait été bien moins agréable. Vous avez transformé chaque moment difficile en un souvenir précieux grâce à votre soutien et vos rires. Les instants passés ensemble resteront à jamais gravés dans mon cœur, et je suis profondément reconnaissante de vous avoir à mes côtés. Vous avez rendu ce chemin inoubliable.

À **mes cousines et cousins, en particulier Tissa, Tia, Sophie et manel**, pour l'amour sincère et les moments de partage. Votre soutien a toujours été une source de réconfort pour moi. Je vous en suis profondément reconnaissante.

À **mes oncles, et en particulier Farid et Abedelhak**, pour vos conseils avisés, votre sagesse et votre bienveillance. Vous avez toujours été là pour me guider, et votre soutien m'a permis d'avancer dans les moments difficiles.

À **toutes mes tantes**, pour votre amour, votre bienveillance et votre tendresse. Vous êtes des modèles pour moi, et votre exemple me pousse chaque jour à devenir meilleure.

À **Amel**, ma source d'inspiration depuis mon plus jeune âge. Ta force, ta sagesse et ta détermination m'ont toujours guidée. Grâce à toi, j'ai appris à ne jamais abandonner, à croire en mes rêves et à avancer avec courage même dans les moments les plus difficiles.

À **Yacine**, mon guide dans la vie, pour ta sagesse et ton soutien inébranlable. Tu as toujours su me montrer la voie, même dans les périodes de doute. Je te serai éternellement reconnaissante pour tout ce que tu m'as appris.

À **mes rencontres de l'université**, Hadjila, Thassadit, Céline, Mariya, Sofiane et Sabiha pour votre amitié précieuse, vos échanges enrichissants et votre soutien inestimable tout au long de mon parcours universitaire.

◇ *À toutes les personnes qui occupent une place précieuse dans mon cœur* ◇

Lydia Benadjaoud

---

## ◇ DÉDICACES ◇

---

Je dédie ce travail à **mes parents**, mes repères, mes racines, mes étoiles. Votre amour inconditionnel est la lumière qui éclaire mon chemin, votre confiance une aile invisible qui m'a permis de m'élever. Vous êtes ma force tranquille, mon éternelle source de courage et de tendresse.

À **ma grand-mère maternelle Nouara**, celle qui a toujours été à mes côtés, celle qui m'a élevée avec tant d'amour et de douceur, tu étais bien plus qu'une grand-mère, tu étais ma deuxième mère. Partie trop tôt, tu laisses un vide immense, mais ton amour et ta sagesse restent à jamais gravés dans mon cœur. Tu es toujours présente dans mes pensées, et ton souvenir m'accompagne chaque jour. Tu seras à jamais une source de chaleur et d'inspiration.

À **Naouel**, ma sœur, mon guide, mon ancrage. Par ton amour profond et ta tendresse infinie, tu as été ce cœur maternel qui m'a portée, rassurée et élevée. Tu es ma lumière dans l'ombre, ma boussole du cœur.

À **ma fratrie de cœur, Djamel & Katia**, deux âmes unies, ainsi qu'à mon cher neveu. Votre bienveillance, votre stabilité et votre amour ont profondément marqué mon chemin. **Djamel, mon cher frère**, tu es ma force tranquille, ce pilier sûr dans mes silences et mes doutes. **Katia**, ta douceur et ta lumière ont réchauffé tant d'instant.

À **mon frère Nabil**, tu es bien plus qu'un beau-frère pour moi. Tu es un véritable frère, toujours présent avec des conseils précieux et un soutien constant.

À **ma meilleure amie, ma sœur et ma binôme, Lydia**, ta fidélité, ton soutien sans condition, et ton énergie. Tu es bien plus qu'une amie, tu es une sœur de cœur, et ce projet n'aurait pas été ce qu'il est sans toi. Ton implication, ton sourire et ton amour m'ont poussée à avancer chaque jour.

À **Céline et Inès**, mes sœurs, mes alliées fidèles, mes perles rares, mes âmes de lumière. Votre bonté, votre énergie et votre loyauté m'ont portée plus d'une fois. Vous avez su être là, sans condition, avec vos bras ouverts et vos cœurs grands.

À **Kenza, Manel & Tissa, Zouza et kenz** mes éclats de bonheur. Chaque instant partagé avec vous a réchauffé mon âme, votre présence un cadeau sans pareil.

À **mes cousins et mes cousines : Amine, Wahid, Maya, Amir, Farida, Amira**, les branches fortes de mon arbre de vie. Merci pour votre amour brut, vrai, sincère. Vous incarnez la chaleur de la famille, celle qui embrasse sans poser de questions.

À **mes amies de la fac : Mariya, Amine, Hadjila, Thassadit et Sabiha**, mes complices de savoir et d'aventure. Votre bienveillance, vos éclats de rire entre deux cours, vos mots doux dans les moments difficiles... Vous avez coloré ce chapitre universitaire avec humanité et douceur.

À **mon oncle Smail et sa femme Messad**, pour leur amour sincère, leurs précieux conseils et leur soutien constant. Je suis fière de vous avoir à mes côtés.

Et à toi, **ma chère tante Ghaniya**, ton amour discret a été un socle silencieux mais indestructible. Ta présence feutrée et rassurante m'a portée dans l'ombre, sans jamais réclamer de lumière. Merci pour ta constance, ton écoute, ton amour si délicat.

◇ *À toutes les personnes qui occupent une place précieuse dans mon cœur* ◇

**Thinhinane Idiri**

# Liste des abréviations

<b>API</b>	Application Programming Interface (Interface de Programmation d'Application)
<b>CDN</b>	Content Delivery Network (Réseau de diffusion de contenu)
<b>CNN</b>	Convolutional Neural Network
<b>CSS</b>	Cascading Style Sheets (Feuilles de style en cascade)
<b>FastAPI</b>	Fast Application Programming Interface
<b>GELU</b>	Gaussian Error Linear Unit
<b>HTML</b>	HyperText Markup Language (Langage de balisage hypertexte)
<b>HTTP</b>	HyperText Transfer Protocol (Protocole de transfert hypertexte)
<b>IA</b>	Intelligence Artificielle
<b>JSON</b>	JavaScript Object Notation
<b>JWT</b>	JSON Web Token
<b>ML</b>	Machine Learning (Apprentissage automatique)
<b>NCS</b>	Natural Color System
<b>NoSQL</b>	Not Only SQL
<b>PIL</b>	Python Imaging Library
<b>RAL</b>	Reichs-Ausschuss für Lieferbedingungen
<b>SMTP</b>	Simple Mail Transfer Protocol (Protocole Simple de Transfert de Courrier)
<b>UI</b>	User Interface (Interface Utilisateur)
<b>UML</b>	Unified Modeling Language (Langage de modélisation unifié)
<b>UX</b>	User Experience (Expérience Utilisateur)
<b>ViT</b>	Vision Transformer
<b>XP</b>	Extreme Programming (Programmation Extrême)

# Table des matières

## Liste des abréviations

<b>1</b>	<b>Présentation du projet</b>	<b>2</b>
1	Introduction . . . . .	2
2	Définition du projet . . . . .	2
2.1	Présentation du projet de vente de vêtements de second main "ReWear"	2
2.2	Contexte et motivation . . . . .	2
2.3	Problématique . . . . .	3
2.4	Etude de l'existant . . . . .	4
2.5	Solution . . . . .	7
2.6	Méthodologie de développement . . . . .	8
3	Conclusion . . . . .	10
<b>2</b>	<b>Sprint de Planification du projet</b>	<b>11</b>
1	Introduction . . . . .	11
2	Répartition des rôles . . . . .	11
3	Identification des acteurs et cas d'utilisation . . . . .	12
3.1	Identification des acteurs . . . . .	12
3.2	Diagramme de cas d'utilisation global . . . . .	12
4	Product Backlog . . . . .	14
5	Décomposition des sprints . . . . .	17
5.1	Charte graphique . . . . .	17
5.2	Le logo . . . . .	17
5.2.1	Signification du logo . . . . .	18
5.3	Palette de couleurs utilisée . . . . .	18
5.4	Outils et langages utilisés . . . . .	18
5.4.1	Niveau conceptuel . . . . .	18
5.4.2	Niveau développement . . . . .	19
6	Conclusion . . . . .	20
<b>3</b>	<b>Sprint 1,2</b>	
	<b>Mise en place des fonctionnalités de base</b>	<b>21</b>
1	Introduction . . . . .	21
2	<b>Sprint 1</b>	
	<b>Gestion des Comptes utilisateurs</b> . . . . .	<b>21</b>
2.1	Diagramme de cas d'utilisation . . . . .	21
2.2	Product Backlog . . . . .	22
2.3	Décomposition du sprint . . . . .	23
2.4	présentation des diagrammes de séquences . . . . .	24
2.4.1	Fonctionnalité « Inscription » . . . . .	24
2.4.2	Fonctionnalité « Authentification » . . . . .	26

2.4.3	Fonctionnalité « Modification du profil » . . . . .	27
2.5	diagramme de classes . . . . .	29
2.6	Vue des Interfaces Utilisateur . . . . .	29
2.6.1	Interface d'inscription . . . . .	29
2.6.2	Interface d'authentification . . . . .	30
2.6.3	Interface de Gestion du profil . . . . .	30
3	<b>Sprint 2</b>	
	<b>Gestion de la vente et recherche d'articles</b> . . . . .	31
3.1	Diagramme de cas d'utilisation . . . . .	31
3.2	Product Backlog . . . . .	32
3.3	Décomposition du sprint . . . . .	34
3.4	Présentation des diagrammes de séquence . . . . .	34
3.5	Fonctionnalité « Gestion de mise en vente » . . . . .	34
3.5.1	Fonctionnalité ajouter article . . . . .	34
3.5.2	Fonctionnalité Modifier et supprimer un article . . . . .	35
3.5.3	Modifier et supprimer un article . . . . .	36
3.6	Fonctionnalité « Consultation du catalogue » . . . . .	37
3.7	Scénarios alternatifs (erreurs) . . . . .	37
3.8	Fonctionnalité « Affichage des détails d'article » . . . . .	39
3.9	Fonctionnalité « Recherche des articles » . . . . .	39
3.10	Scénarios alternatifs (erreurs) . . . . .	39
3.11	Diagramme de classes . . . . .	40
3.12	Vue des Interfaces Utilisateur . . . . .	40
3.12.1	Interface de gestion de mise en vente . . . . .	40
3.12.2	Interface du catalogue . . . . .	41
3.12.3	Interface fiche détaillée d'article . . . . .	41
4	Conclusion . . . . .	42
4	<b>Sprint 3, 4</b>	
	<b>Développement des fonctionnalités commerciales essentielles</b>	<b>43</b>
1	Introduction . . . . .	43
2	<b>Sprint 3</b>	
	<b>Gestion des favoris, du panier et des commandes</b> . . . . .	43
2.1	Diagramme de cas d'utilisation . . . . .	43
2.2	Product Backlog . . . . .	44
2.3	Décomposition du sprint . . . . .	46
2.4	Présentation des diagrammes de séquence . . . . .	47
2.4.1	Fonctionnalité « Gestion des favoris » . . . . .	47
2.4.2	Fonctionnalité « Gestion du panier » . . . . .	48
2.4.3	Fonctionnalité « Gestion des commandes » . . . . .	49
2.4.4	Fonctionnalité « Paiement » . . . . .	51
2.5	Diagramme de classes . . . . .	54
2.6	Vue des Interfaces Utilisateur . . . . .	54
2.6.1	Interface de gestion des favoris . . . . .	55
2.6.2	Interface de gestion du panier . . . . .	55
2.6.3	Interface de passage de la commande . . . . .	55
2.6.4	Interface de paiement via Chargily . . . . .	56
3	<b>Sprint 4</b>	
	<b>Gestion de la messagerie, évaluation, notifications et du tableau de bord admin</b> . . . . .	56

3.1	Diagramme de cas d'utilisation . . . . .	56
3.2	Product Backlog . . . . .	57
3.3	Décomposition du sprint . . . . .	61
3.4	Présentation des diagrammes de séquence . . . . .	61
3.4.1	Fonctionnalité « Messagerie » . . . . .	61
3.4.2	Fonctionnalité « Évaluation des articles » . . . . .	64
3.4.3	Fonctionnalité « Tableau de bord pour l'administrateur » . . . . .	65
3.4.4	Fonctionnalité « Notifications » . . . . .	70
3.5	Diagramme de classes . . . . .	71
3.6	Vue des Interfaces Utilisateur . . . . .	72
3.6.1	Interface de messagerie . . . . .	72
3.6.2	Interface d'évaluation des articles . . . . .	72
3.6.3	Interface du tableau de bord administrateur . . . . .	73
3.6.4	Interface de notifications . . . . .	73
4	Conclusion . . . . .	74

## 5 Sprint 5

	<b>Implémentation d'un système recommandation avec l'API Recombee</b>	<b>75</b>
1	Introduction . . . . .	75
1.1	Présentation de l'API de recommandation Recombee . . . . .	75
1.1.1	Définition de l'API Recombee : . . . . .	75
1.1.2	Principe de fonctionnement de Recombee . . . . .	75
1.2	Diagramme de cas d'utilisation . . . . .	76
1.3	Product Backlog . . . . .	76
1.4	Décomposition du sprint . . . . .	77
1.5	Présentation des diagrammes de séquence . . . . .	78
1.5.1	Fonctionnalité « voir recommandations » . . . . .	78
2	Diagramme de classes . . . . .	80
3	Vue des Interfaces Utilisateur . . . . .	81
3.0.1	Interface d'affichage des recommandations . . . . .	82
4	Conclusion . . . . .	82

## 6 Sprint 6 Intégration d'un Chatbot Intelligent avec Rasa

1	Introduction . . . . .	83
2	Présentation de Rasa . . . . .	83
2.1	Définition et Architecture . . . . .	83
2.2	Concepts Fondamentaux . . . . .	84
2.3	Avantages de Rasa pour notre Projet . . . . .	84
2.4	Diagramme de cas d'utilisation . . . . .	84
2.5	Product Backlog . . . . .	86
2.6	Décomposition du sprint . . . . .	87
2.7	Présentation des diagrammes de séquence . . . . .	87
3	Vue des Interfaces Utilisateur . . . . .	89
3.1	Interface de recherche d'article via le chatbot . . . . .	89
3.2	Interface de navigation entre pages via le chatbot . . . . .	90
4	Conclusion . . . . .	90

## 7 Sprint 7

	<b>Intégration de la Recherche visuelle avec l'Intelligence Artificielle</b>	<b>91</b>
1	Introduction . . . . .	91
1.1	Outils utilisés pour l'implémentation de la recherche visuelle . . . . .	91

1.1.1	API Hugging Face . . . . .	91
1.1.2	Techniques d'analyse d'image . . . . .	92
2	Diagramme de cas d'utilisation . . . . .	92
3	Product Backlog . . . . .	94
4	Décomposition du sprint . . . . .	94
5	Présentation du diagramme de séquence . . . . .	95
5.0.1	Fonctionnalité « Rechercher un article par une image » . . . . .	95
6	Vue des Interfaces Utilisateur . . . . .	96
6.1	Interface de recherche par image . . . . .	96
6.2	Interface des résultats de la recherche . . . . .	97
7	Conclusion . . . . .	97

# Table des figures

1.1	L'interface de l'application mobile <b>Vandista</b> . . . . .	4
1.2	Autre interface de l'application mobile <b>Vandista</b> . . . . .	4
1.3	L'interface de l'application web <b>OuedKniss</b> . . . . .	5
1.4	L'interface de l'application <b>Marketplace Facebook</b> . . . . .	6
1.5	Méthode <b>SCRUM</b> . . . . .	10
2.1	Diagramme de cas d'utilisation . . . . .	13
2.2	Sprint planning . . . . .	17
2.3	Logo de l'application . . . . .	17
2.4	La palette de couleurs . . . . .	18
2.5	Draw.io . . . . .	19
2.6	Figma . . . . .	19
3.1	Diagramme de cas d'utilisation pour <b>Sprint 1</b> . . . . .	22
3.2	Décomposition du sprint 1 . . . . .	24
3.3	Diagramme de séquence pour la fonctionnalité <b>Inscription</b> . . . . .	25
3.4	Diagramme de séquence pour la fonctionnalité <b>Authentification</b> . . . . .	26
3.5	Diagramme de séquence pour la fonctionnalité <b>Gestion de profil</b> . . . . .	28
3.6	diagramme de classes <b>sprint 1</b> . . . . .	29
3.7	Interface d'inscription . . . . .	30
3.8	Interface d'authentification . . . . .	30
3.9	Interface du profil . . . . .	31
3.10	Interface de modification du profil (avatar,numéro de téléphone) . . . . .	31
3.11	Interface de modification du profil (Adresse) . . . . .	31
3.12	Ensemble des interfaces de gestion du profil utilisateur . . . . .	31
3.13	Diagramme de cas d'utilisation pour <b>Sprint 2</b> . . . . .	32
3.14	Décomposition du Sprint 2 . . . . .	34
3.15	Diagramme de séquence pour la fonctionnalité <b>Ajouter article</b> . . . . .	35
3.16	Diagramme de séquence pour la fonctionnalité <b>Modifier ou supprimer article</b> . . . . .	36
3.17	Diagramme de séquence pour la fonctionnalité <b>parcourir le catalogue en fil- trants les articles</b> . . . . .	38
3.18	Diagramme de séquence pour la fonctionnalité <b>Recherche des articles</b> . . . . .	39
3.19	Diagramme de classes <b>Sprint 2</b> . . . . .	40
3.20	Mise en vente des articles . . . . .	41
3.21	Interface de consultation du catalogue . . . . .	41
3.22	Interface fiche détaillée d'article . . . . .	42
4.1	Diagramme de cas d'utilisation pour <b>Sprint 3</b> . . . . .	44
4.2	Décomposition du Sprint 3 . . . . .	47
4.3	Diagramme de séquence pour la fonctionnalité <b>gestion des favoris</b> . . . . .	48
4.4	Diagramme de séquence pour la fonctionnalité <b>gestion du panier</b> . . . . .	49
4.5	Diagramme de séquence pour la fonctionnalité <b>passer commande</b> . . . . .	51

4.6	Diagramme de séquence pour la fonctionnalité <b>paiement avec Chargily</b> . . . . .	53
4.7	Diagramme de classes <b>Sprint 3</b> . . . . .	54
4.8	Interfaces des favoris . . . . .	55
4.9	Interfaces du panier . . . . .	55
4.10	Interface de passage de commande . . . . .	56
4.11	Interfaces de l'API Charegily . . . . .	56
4.12	Diagramme de cas d'utilisation pour <b>Sprint 4</b> . . . . .	57
4.13	Décomposition du Sprint 4 . . . . .	61
4.14	Diagramme de séquence pour la fonctionnalité <b>Envoyer un message</b> . . . . .	63
4.15	Diagramme de séquence pour fonctionnalité <b>laisser un avis sur un article</b> . . . . .	65
4.16	Diagramme de séquence pour fonctionnalité <b>ajouter une sous-catégorie</b> . . . . .	67
4.17	Diagramme de séquence pour fonctionnalité <b>modifier ou supprimer une sous-catégorie</b> . . . . .	69
4.18	Diagramme de séquence pour fonctionnalité <b>Envoi des notifications</b> . . . . .	70
4.19	Diagramme de classes <b>Sprint 4</b> . . . . .	71
4.20	Messagerie . . . . .	72
4.21	Interface d'évaluation des articles . . . . .	72
4.22	Interfaces des fonctionnalités de l'administrateur . . . . .	73
4.23	Notifications . . . . .	73
5.1	Interface de l'API Recombee . . . . .	76
5.2	Diagramme de cas d'utilisation pour <b>Sprint 5</b> . . . . .	76
5.3	Décomposition du <b>sprint 5</b> . . . . .	78
5.4	Diagramme de séquence pour la fonctionnalité <b>Recommandation avec l'API Recombee</b> . . . . .	79
5.5	Diagramme de classes <b>Sprint 5</b> . . . . .	81
5.6	Interface des recommandations . . . . .	82
6.1	Diagramme de cas d'utilisation pour <b>Sprint 6</b> . . . . .	85
6.2	Décomposition du Sprint 6 . . . . .	87
6.3	Diagramme de séquence - Fonctionnalité "Rechercher un produit" . . . . .	88
6.4	Diagramme de séquence - Fonctionnalité "Naviguer vers la page de vente" . . . . .	89
6.5	Interface de recherche d'article via chatbot . . . . .	90
6.6	Interface de navigation entre pages via un chatbot . . . . .	90
7.1	Illustration de l'algorithme <b>K-means</b> . . . . .	92
7.2	Diagramme de cas d'utilisation du <b>sprint 7</b> . . . . .	93
7.3	Décomposition du <b>Sprint 7</b> . . . . .	95
7.4	Diagramme de séquence pour la fonctionnalité <b>recherche par image</b> . . . . .	96
7.5	Recherche d'un article par image . . . . .	96
7.6	Résultats de la recherche par image . . . . .	97

# Liste des tableaux

1.1	Avantages et Limites de Vandista . . . . .	5
1.2	Avantages et Limites de Ouedkniss . . . . .	6
1.3	Avantages et Limites de Facebook Marketplace . . . . .	7
2.1	Répartition des rôles SCRUM . . . . .	11
2.2	Product Backlog . . . . .	16
3.1	Product Backlog du sprint 1 . . . . .	23
3.2	Product Backlog du sprint 2 . . . . .	33
4.1	Product Backlog du sprint 3 . . . . .	46
4.2	Product Backlog du sprint 4 . . . . .	61
5.1	Product Backlog du sprint 5 . . . . .	77
6.1	Product Backlog du sprint 6 . . . . .	87
7.1	Product Backlog du sprint 7 . . . . .	94

# Introduction Générale

Face à une prise de conscience écologique croissante et à un contexte économique difficile, le marché des vêtements de seconde main connaît un essor notable en Algérie. Dans une conjoncture où le coût de la vie devient de plus en plus cher, de nombreux Algériens explorent de nouvelles alternatives pour bien s'habiller sans compromettre leur budget.

Cependant, ce secteur reste largement informel, avec environ 90% des transactions réalisées via des canaux non structurés, tels que les réseaux sociaux [1]. Cette situation engendre des défis majeurs en termes de confiance des consommateurs, de sécurité des transactions et d'efficacité des processus de recherche.

L'Algérie, malgré l'ampleur de son marché de la friperie, ne dispose pas encore d'une solution numérique structurée, contrairement aux plateformes européennes et nord-américaines comme Vinted, Vestiaire Collective ou ThredUp. Les quelques initiatives existantes restent limitées aux réseaux sociaux, sans garanties de sécurité ou système de paiement intégré.

Le projet ReWear vise à structurer ce marché informel en proposant une plateforme web innovante dédiée à l'achat et à la vente de vêtements d'occasion. La solution intègre des technologies avancées telles qu'un chatbot intelligent, système de recommandation personnalisé, recherche par image et paiement sécurisé via Chargly. L'objectif est double : digitaliser et sécuriser les échanges tout en contribuant à une économie circulaire responsable, créant ainsi de la valeur économique et sociale pour l'écosystème local.

Notre étude suit une méthodologie agile structurée en sept chapitres :

- Chapitre 1 – Présentation du projet : Contexte, objectifs et motivations du projet ReWear, ainsi que la problématique liée au marché des vêtements de seconde main en Algérie.
- Chapitre 2 – Sprint de planification : Organisation de l'équipe Scrum, présentation de la méthodologie adoptée, définition des rôles et responsabilités, ainsi que le planning des différentes phases de développement.
- Chapitre 3 – Sprint 1 et 2 : Mise en place des fonctionnalités de base, telles que l'inscription des utilisateurs, l'authentification et la gestion des profils personnels.
- Chapitre 4 – Sprint 3 et 4 : Développement des fonctionnalités commerciales essentielles, incluant la gestion des favoris, le panier d'achat, l'intégration d'un paiement sécurisé via Chargly, ainsi que des outils de communication comme la messagerie, la notation des articles et le tableau de bord administrateur.
- Chapitre 5 – Sprint 5 : Intégration d'un système de recommandation personnalisé, utilisant l'intelligence artificielle pour analyser les préférences des utilisateurs et leur proposer des articles adaptés à leurs goûts.
- Chapitre 6 – Sprint 6 : Déploiement d'un chatbot intelligent, basé sur le traitement du langage naturel, pour guider les utilisateurs tout au long de leur expérience sur la plateforme.
- Chapitre 7 – Sprint 7 : Mise en place de la fonctionnalité de recherche par image, pour permettre aux utilisateurs de trouver des articles similaires à partir de photos téléchargées.

# Chapitre 1

## Présentation du projet

### 1 Introduction

Dans ce premier chapitre , nous présentons le projet ReWear, une plateforme en ligne innovante qui facilite l'achat et la vente d'articles d'occasion entre particuliers. L'objectif principal de ce projet est de promouvoir une consommation responsable, inspirée par les principes de l'économie circulaire et du recyclage. Nous analysons également la problématique ainsi que la solution que ce projet vise à apporter. Enfin , nous décrivons l'approche Scrum Agile adoptée pour la gestion du projet et les principaux choix techniques mis en œuvre.

### 2 Définition du projet

#### 2.1 Présentation du projet de vente de vêtements de second main "ReWear"

ReWear est une application web intuitive dédiée à l'achat et à la vente de vêtements d'occasion pour hommes, femmes et enfants en Algérie. Elle offre aux utilisateurs un espace sécurisé et facile à utiliser, où ils peuvent publier des annonces, gérer leurs transactions et communiquer via une messagerie intégrée. Afin d'améliorer l'expérience utilisateur, ReWear intègre des technologies d'intelligence artificielle, notamment un système de recommandation personnalisé basé sur les préférences individuelles, ainsi qu'un chatbot assistant pour faciliter la navigation et l'aide à la recherche. Grâce à des filtres avancés, la recherche de vêtements devient simple et rapide, permettant ainsi à chaque utilisateur de trouver facilement des articles adaptés à ses besoins. Cette approche innovante favorise une consommation responsable en donnant une nouvelle vie aux vêtements tout en rendant la mode plus accessible.

#### 2.2 Contexte et motivation

- **Environnemental**

En Algérie, des milliers de tonnes de vêtements usagés sont jetées chaque année dans des décharges non contrôlées, contribuant ainsi à la pollution de l'environnement. En favorisant la réutilisation des vêtements, on peut réduire ces déchets et protéger l'environnement. ReWear veut encourager cette consommation responsable en facilitant la vente de vêtements d'occasion. [2]

- **Économique**

Le commerce en ligne se développe rapidement en Algérie, avec une croissance de 25% en 2023. Les moyens de paiement électroniques, comme les cartes bancaires ou les paiements

mobiles, deviennent plus populaires, ce qui rend les achats en ligne plus faciles et sûrs. Ce contexte est une bonne occasion pour créer des plateformes comme ReWear, qui facilitent les échanges de vêtements d'occasion grâce aux nouvelles technologies. [3]

- **Social et culturel**

La connectivité en Algérie dépasse aujourd'hui les 100% grâce à l'utilisation massive des téléphones mobiles. Les jeunes consommateurs, en particulier, font de plus en plus attention à l'origine, à la qualité et à l'impact écologique de ce qu'ils achètent. Ils préfèrent de plus en plus acheter de manière responsable et privilégier les circuits courts. De plus, le marché des vêtements d'occasion est en forte croissance, offrant une alternative économique tout en aidant à réduire les déchets textiles. [4]

ReWear s'inscrit dans cette tendance en proposant une solution moderne, facile d'accès et sécurisée, qui répond aux attentes des consommateurs algériens.

## 2.3 Problématique

Le marché des vêtements d'occasion connaît un essor grâce à une prise de conscience écologique et économique. En Algérie, cette tendance commence à émerger, mais elle rencontre encore plusieurs obstacles qui ralentissent son expansion. Cette situation soulève une problématique essentielle : **comment structurer et dynamiser le marché des vêtements de seconde main en Algérie afin de le rendre accessible, sécurisé et attractif?**

Malgré un intérêt croissant pour la consommation responsable, la vente et l'achat de vêtements d'occasion restent désorganisés, ce qui s'explique par l'existence de multiples obstacles majeurs :

- **Confiance limitée des consommateurs**

constitue un frein majeur. En l'absence de garanties claires et d'informations précises, les acheteurs hésitent à se lancer dans ce type d'achat, craignant de recevoir des produits endommagés ou non conformes à la description. [2, 5].

- **L'absence d'une plateforme sécurisée et facile à utiliser pour acheter et vendre des vêtements d'occasion**

complique la gestion des transactions et limite la visibilité des offres disponibles. Les utilisateurs manquent d'un espace centralisé leur permettant de publier, consulter et négocier leurs articles en toute sérénité. [5].

- **La difficulté à trouver des articles adaptés aux goûts et besoins personnels**

freine l'adoption de la seconde main. Les recherches sont souvent longues et inefficaces faute de filtres pertinents ou de recommandations personnalisées. [2, 6].

- **Fragmentation du marché**

Le marché de la fripe est encore très informel et peu organisé, ce qui rend difficile la visibilité des offres et la gestion des transactions. L'absence d'une plateforme intégrée et conviviale nuit à l'expérience utilisateur et limite la croissance du marché. [4, 7].

- **Le manque de communication fluide entre acheteurs et vendeurs**

entraîne souvent des malentendus ou des frustrations, nuisant à l'expérience utilisateur globale. [4].

Ces défis rendent nécessaire la conception d'une solution digitale innovante, capable de faciliter l'accès au marché de la seconde main tout en renforçant la confiance et l'engagement des utilisateurs. Avant de proposer une telle solution, il est primordial de réaliser une étude approfondie de l'existant et d'analyser précisément ces problématiques afin de construire une plateforme adaptée et efficace pour le contexte algérien.

## 2.4 Etude de l'existant

En analysant le marché algérien, nous avons constaté que le secteur de la vente de vêtements d'occasion reste encore peu développé. Nous avons donc choisi d'étudier trois plateformes principales : Vandista, qui est la seule application véritablement spécialisée dans la vente de vêtements de seconde main, ainsi que Ouedkniss et Facebook Marketplace, qui sont spécialisées dans la vente de multiples articles d'occasion. Cette analyse a pour objectif de comprendre comment ces plateformes fonctionnent, d'identifier leurs points forts et leurs limites, afin de déterminer ce qu'il faut ajouter et améliorer dans notre application ReWear pour qu'elle trouve sa place sur le marché algérien. Pour débiter cette analyse, nous examinerons successivement les fonctionnalités, les avantages et les limitations de chaque plateforme :

### Application Vandista

- **Présentation de l'application**

Vandista est une application mobile algérienne de vide-dressing, spécialisée dans la vente et l'achat de tous types de vêtements d'occasion. Elle permet de créer des comptes, de publier des annonces de vente et de rechercher et filtrer les articles selon différentes caractéristiques telles que la taille, la marque et le prix. Elle est disponible en Algérie via le Play Store et l'App Store.[8]

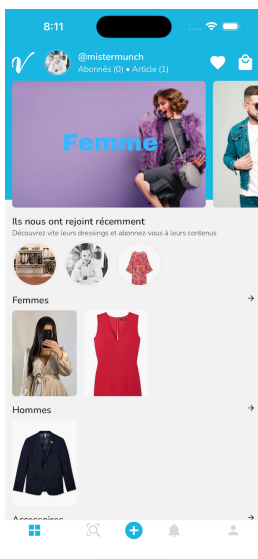


FIGURE 1.1 – L'interface de l'application mobile **Vandista**



FIGURE 1.2 – Autre interface de l'application mobile **Vandista**

- **Avantages et limites de l'application Vandista**

Avantages	Limites
- Une interface simple et conviviale ce qui facilite la mise en vente des articles.	- Absence de la messagerie et de la communication en temps réel entre les utilisateurs, limitant ainsi la possibilité de négociation pour les acheteurs.
- Livraison rapide et sécurisée via des partenaires de livraison.	- L'absence de l'intelligence artificielle comme un système de recommandations et un chatbot intelligent limite l'expérience utilisateur et réduit le taux de ventes des produits dans l'application.
- Protection des transactions en permettant de demander des retours en cas de défaut ou de fraude.	- Limitation géographique, restreignant l'accès à la plateforme uniquement en Algérie ce qui limite les autres utilisateurs d'y accéder.
- Disponibilité sur le Play Store et l'App Store, rendant l'application accessible aux utilisateurs d'Android et d'iOS.	
- Partage des annonces sur les réseaux sociaux.	

TABLE 1.1 – Avantages et Limites de Vandista

## Plateforme d'annonce de vente OuedKniss

### • Présentation de l'application

Ouedkniss est la première plateforme d'annonces de vente d'articles d'occasion en ligne en Algérie. Le site a vu le jour en 2006 par cinq lycéens souhaitant redonner vie au marché de la rue de Oued Kniss. Aujourd'hui, le site héberge plus d'un million d'annonces couvrant de nombreux domaines : voyage, voitures, électroménager, immobilier, vêtements, etc.[9] La plateforme possède un site web et une application mobile, avec une interface simple et conviviale qui facilite son utilisation. Elle permet de publier des annonces, de les filtrer et de les rechercher selon des caractéristiques spécifiques.[10]

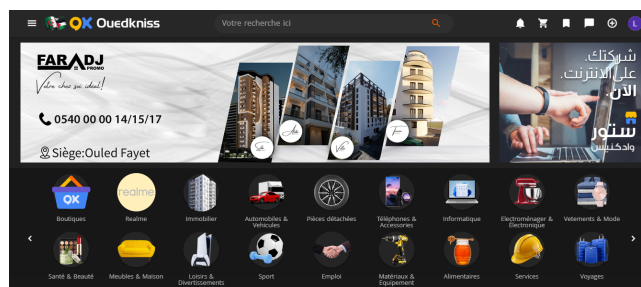


FIGURE 1.3 – L'interface de l'application webOuedKniss

### • Avantages et limites de la plateforme OuedKniss

Avantages	Limites
- Accessibilité web et mobile, permettant aux utilisateurs de consulter la plateforme à tout moment et n'importe où.	-L'absence de l'intelligence artificielle comme un système de recommandations et un chatbot intelligent limite l'expérience utilisateur et réduit le taux de ventes des produits dans l'application.
- Un système de communication varié entre les utilisateurs : messagerie interne et externe (téléphone et e-mail), facilitant la prise de rendez-vous et la conclusion des transactions.	La disponibilité de plusieurs services et produits dans Ouedkniss rend la recherche de vêtements moins intuitive et encombré par des milliers d'annonces non pertinentes.
-Un service payant de création de boutique professionnelles en ligne, appelé Store Ouedkniss, qui permet aux vendeurs d'avoir une vitrine personnalisée et mise en avant sur la plateforme pour attirer plus de clients.	-Manque des services de paiement en ligne et de livraison.

TABLE 1.2 – Avantages et Limites de Ouedkniss

## Plateforme de vente Facebook Marketplace

### • Présentation de l'application

Marketplace Facebook a été créé en 2017 en France. Il s'agit d'une plateforme intégrée à Facebook, dont le but est de mettre en relation des vendeurs et des acheteurs. Tous les utilisateurs de Facebook peuvent donc acheter et vendre des articles neufs et d'occasion à d'autres utilisateurs.

[11]

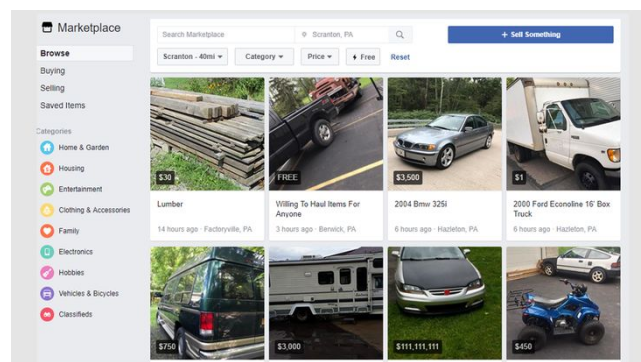


FIGURE 1.4 – L'interface de l'application Marketplace Facebook

- **Avantages et limites de Facebook Marketplace**

Avantages	Limites
- Accessibilité web et mobile, permettant aux utilisateurs de consulter la plateforme à tout moment et n'importe où.	-La disponibilité de plusieurs services et produits dans Facebook Marketplace rend la recherche de vêtements moins intuitive et encombré par des milliers d'annonces non pertinentes.
- L'immense audience de Facebook augmente fortement le nombre de vues potentielles par annonce et diversifie l'offre disponible.	- Manque des services de paiement en ligne et de livraison.
- Système de filtrage avancé selon plusieurs critères : prix, catégorie, état et localisation, ce qui facilite le tri parmi les millions d'annonces.	
- La transparence du profil de chaque utilisateur, les outils de signalement et de modération permettent de renforcer la confiance et la sécurité.[11][12]	
- Un système de recommandation avancé basé sur l'intelligence artificielle enrichit l'expérience utilisateur et augmente le taux de vente.[13]	

TABLE 1.3 – Avantages et Limites de Facebook Marketplace

En conclusion, l'analyse de Vandista, Ouedkniss et Facebook Marketplace met en évidence un double constat : d'un côté, les plateformes généralistes (Ouedkniss, Marketplace) offrent un vaste catalogue mais manquent de spécialisation (filtres précis, recommandations IA, paiement et livraison intégrés) ; de l'autre, la solution exclusivement mode (Vandista) propose des outils dédiés (guide des tailles, messagerie sécurisée, expéditions organisées) mais souffre de limites techniques (absence d'IA).

Pour combler les lacunes actuelles, nous proposons une solution complète et spécialisée qui offrira aux utilisateurs algériens une expérience de second main à la fois intuitive, fiable et efficace.

## 2.5 Solution

La solution proposée est ReWear, une application web spécialisée et complète, qui combine les meilleures fonctionnalités des plateformes existantes tout en comblant leurs lacunes, conçue pour rendre le marché algérien des vêtements de seconde main accessible, sécurisé et attrayant.

### 1. Spécialisation et centralisation

- Un catalogue organisé et facile à explorer via un système de filtrage avancé (taille, style, état, prix, ...).

- La recherche visuelle d'articles par reconnaissance d'image pour une découverte intuitive des articles.

## 2. Système de recommandation avancé

- Un système de recommandation avancé basé sur l'intelligence artificielle, conçu pour analyser les actions et les interactions des utilisateurs (achats, favoris) pour les satisfaire, en leur proposant des articles adaptés à leurs préférences.

## 3. Interaction sécurisée et fluide

- Une messagerie interne et sécurisée pour garantir une communication directe et transparente entre vendeurs et acheteurs.
- Un chatbot intelligent qui sera comme un guide pour faciliter l'utilisation des fonctionnalités de l'application.
- Une solution de paiement sécurisée via l'API Chargily offrant aux utilisateurs des transactions rapides et protégées.

## 4. La confiance

- Un système d'évaluations et de commentaires qui garantira la fiabilité des articles.
- Images obligatoires et description détaillée pour chaque article, limitant ainsi la fraude.

## 5. Interface simple et responsive

- Une interface moderne, attractive, facile à utiliser et accessible depuis tout type d'appareil afin de garantir une expérience utilisateur performante et agréable.

## 2.6 Méthodologie de développement

Dans le développement logiciel, on distingue généralement deux approches majeures : l'approche traditionnelle et l'approche agile.

L'approche traditionnelle de gestion de projet, souvent appelée « cascade » ou « cycle en V », est un modèle linéaire qui repose sur l'utilisation d'un processus strict permettant de définir l'ensemble des fonctionnalités du projet, de les spécifier de façon détaillée, de les développer puis de les tester avant la validation. Cette approche, fortement contraignante, affaiblit la flexibilité et rend les ajustements difficiles lorsque les exigences évoluent en cours de développement. À la fin des années 1990, les méthodes agiles sont apparues pour remédier à ce problème et assouplir le processus de développement.[14]

L'approche Agile est une méthode de développement itérative et incrémentale, caractérisée par sa simplicité et sa flexibilité, en privilégiant la communication et la collaboration entre l'équipe et les clients.[14, 15]

La méthodologie agile a développé plusieurs frameworks tels que Extreme Programming (XP), Kanban, et Scrum, chacun avec ses propres principes et caractéristiques [slack2024]. Parmi ces méthodes, Scrum a su se démarquer par son cadre structuré, ses rôles clairement définis et sa capacité à gérer des projets par itérations courtes appelées sprints.[asana2025]

Scrum est composé de plusieurs éléments clés qui travaillent ensemble pour faciliter la gestion de projet agile.

- **Sprint** : Est une unité fondamentale dans la méthode Scrum, par laquelle les projets sont décomposés en cycles courts et répétitifs, généralement compris entre 1 et 4 semaines.[qrpinternational2021]

- **Les artefacts** : Sont des outils ou des documents qui aident l'équipe à organiser, planifier, suivre et livrer le projet.[16] Les principaux artefacts Agiles Scrum sont : le backlog produit, le backlog de sprint et l'incrément.
  - **Le backlog produit** : C'est une liste structurée de toutes les fonctionnalités, exigences, améliorations, corrections et tâches nécessaires pour développer un produit.[asana2025]
  - **Le backlog de sprint** : C'est une liste détaillée des fonctionnalités sélectionnées à partir du backlog produit lors de la séance de planification du sprint pour guider l'équipe dans le travail.[asana2025]
  - **L'incrément** : Appelé aussi produit partiel, est un livrable obtenu à la fin de chaque sprint regroupant tous les éléments du backlog terminés pendant les sprints précédant.[asana2025]
  
- **Les membres d'une équipe Scrum** : Dans la méthode Scrum, l'organisation de l'équipe de travail est primordiale. Chaque membre joue un rôle essentiel pour assurer la bonne réalisation du projet. On distingue trois rôles principaux dans une équipe Scrum :
  - **Product Owner** : C'est le responsable du projet et le chargé de la tenue du backlog produit. Il agit comme porte-parole entre l'Équipe de développement et les clients.[17]
  - **Scrum Master** : C'est le responsable du bon déroulement de la méthode Scrum. Il veille à ce que tout soit mis en œuvre correctement.[18]
  - **Équipe de développement** : Elle est composée de développeurs, de UX/UI designers, de testeurs, etc. Les membres de l'équipe doivent collaborer ensemble afin de livrer un produit final adapté aux besoins de l'utilisateur.[19]
  
- **Les cérémonies Scrum** : Les membres de l'équipe Scrum se réunissent lors des cérémonies Scrum pour planifier et organiser les sprints, discuter des tâches en cours et recueillir du feedback.[16]
 

Il y a principalement quatre réunions Scrum Agile qui permettent de renforcer les équipes de toutes sortes.

  - **Planification du Sprint** : C'est une réunion faite au début de chaque sprint pour définir l'objectif du sprint et les tâches à accomplir.[16]
  - **Le stand-up quotidien** : C'est une réunion quotidienne de 15 min pour discuter de l'état d'avancement du sprint.[16]
  - **La revue du sprint** : C'est une réunion accompli à chaque fin du sprint pour montrer le travail de chaque membre.[16]
  - **rétrospective de sprint** : C'est une réunion à la fin de chaque sprint pour évaluer le travail et identifier des améliorations pour le prochain sprint.

Grâce à cette méthode, nous pourrions ajuster les fonctionnalités et exigences de notre projet au fil de son évolution, en nous adaptant rapidement aux changements et pour assurer une bonne compréhension globale du système, il est essentiel d'utiliser la modélisation.

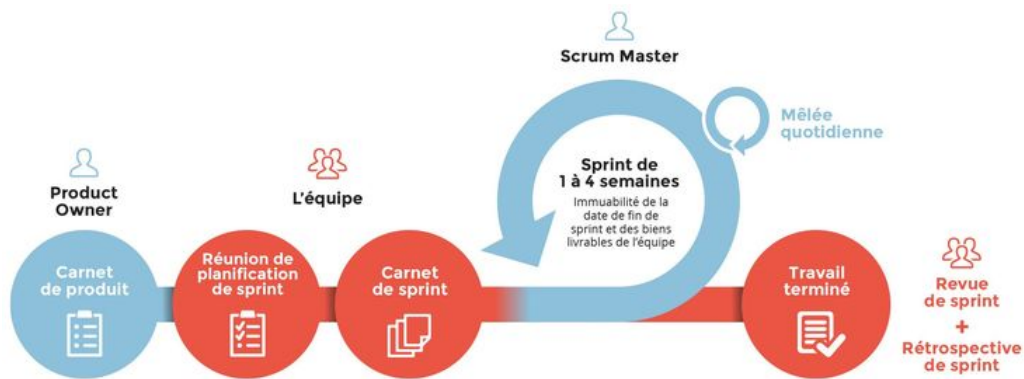


FIGURE 1.5 – Méthode **SCRUM**

### 3 Conclusion

En conclusion, dans ce chapitre, nous avons posé les bases essentielles à la compréhension du projet ReWear en mettant en lumière le contexte global ainsi que les motivations qui ont conduit à la création d'une plateforme dédiée à l'achat et à la vente de vêtements de seconde main en Algérie. Puis, nous avons formulé la problématique qui guide notre démarche en effectuant une étude comparative sur des solutions existantes sur le marché, mettant en lumière leurs forces et leurs limites. Ensuite, nous avons défini une vision claire et concise de ce que notre solution apportera aux utilisateurs. Par ailleurs, ce chapitre explique la méthodologie de développement choisie Scrum, qui organise le travail pour une meilleure efficacité. Dans le chapitre suivant, nous aborderons la phase de planification ainsi que le développement technique et fonctionnel.

# Chapitre 2

## Sprint de Planification du projet

### 1 Introduction

Dans ce chapitre, nous présentons l'organisation de notre équipe Scrum en détaillant les rôles et responsabilités de chacun, afin d'optimiser la collaboration et garantir une gestion efficace du projet. Nous identifions ensuite les acteurs clés du système ainsi que les principaux cas d'utilisation, en lien avec les besoins et priorités définis dans le Product Backlog. Nous exposons la méthodologie Scrum adoptée, en précisant les objectifs fixés et en établissant un planning réaliste pour les différentes phases de développement. Enfin, nous abordons la charte graphique de l'application ainsi que les outils et technologies choisis pour la modélisation et le développement de la solution.

### 2 Répartition des rôles

Pour assurer une organisation efficace et une collaboration optimale dans notre projet, il est primordial de définir clairement les rôles de chaque membre de notre équipe selon la méthode de développement SCRUM.

Rôle SCRUM	Personne affectée
Product Owner	— Monsieur BENNAI Yani-Athmane — Mlle IDIRI Tinhianane — Mlle BENADJAOUD Lydia
Scrum Master	Monsieur BENNAI Yani-Athmane
Équipe de développement	— Mlle IDIRI Tinhianane — Mlle BENADJAOUD Lydia

TABLE 2.1 – Répartition des rôles SCRUM

## 3 Identification des acteurs et cas d'utilisation

Pour assurer un développement correct et efficace de notre application, il est essentiel de définir clairement les acteurs impliqués, leurs rôles respectifs, ainsi que leurs interactions avec le système.

### 3.1 Identification des acteurs

Un acteur est une entité externe qui interagit avec le système ; il peut être représenté par une personne, un périphérique ou un autre système. Dans notre application, nous avons identifié quatre acteurs qui interagissent avec le système.

- **Visiteur** : C'est une personne qui accède à l'application sans être inscrite. Elle peut parcourir le catalogue et visualiser les produits.
- **Vendeur** : C'est un utilisateur inscrit dans le système qui assure la mise en vente de ses produits et bénéficie d'un accès à toutes les fonctionnalités offertes par l'application.
- **Acheteur** : C'est un utilisateur inscrit dans le système qui parcourt le catalogue, sélectionne et achète des produits, bénéficiant d'un accès à toutes les fonctionnalités offertes par l'application.
- **Administrateur** : C'est un utilisateur qui possède des droits élevés lui permettant de veiller au bon fonctionnement de l'application et de contrôler les comptes utilisateurs.

### 3.2 Diagramme de cas d'utilisation global

L'ensemble des cas d'utilisation de notre application est représenté dans le diagramme dans la figure 2.1

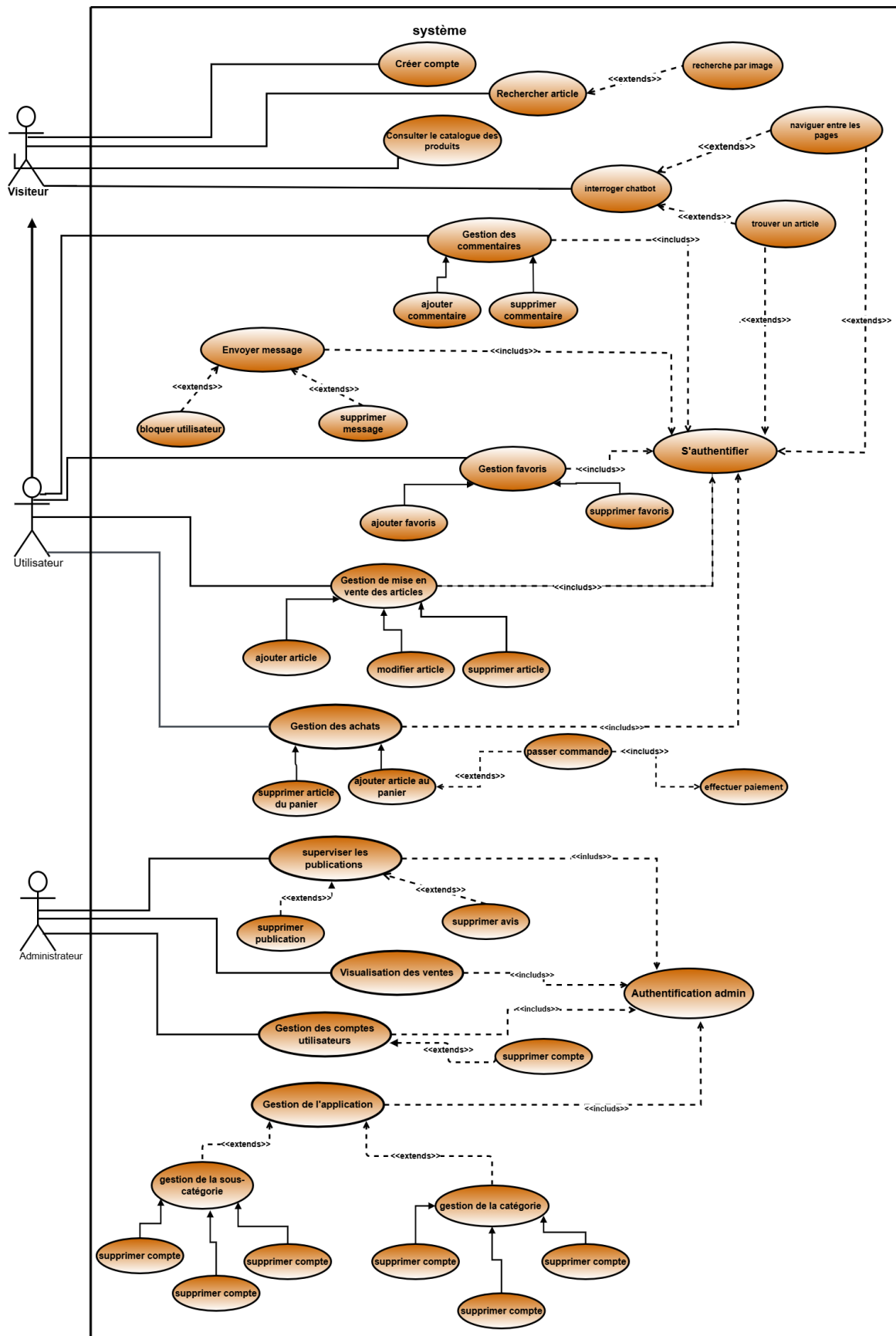


FIGURE 2.1 – Diagramme de cas d'utilisation

## 4 Product Backlog

Le "Product Owner" organise chaque étape en évaluant la priorité de chaque fonctionnalité listée dans le Product Backlog présenté ci-dessous.

<b>Sprint</b>	<b>Items</b>	<b>Acteur</b>	<b>Description du cas d'utilisation</b>	<b>Priorité</b>
Sprint 1	Inscription	Visiteur	En tant que visiteur, je veux pouvoir créer un compte.	1
	Authentification	Utilisateur et Administrateur	En tant qu'utilisateur, je veux avoir la possibilité de me connecter à mon compte pour accéder à toutes les fonctionnalités de l'application	1
	Gestion du profil	Utilisateur	En tant qu'utilisateur, je veux pouvoir récupérer et modifier les informations de mon profil	1
Sprint 2	Gestion de mise en vente des articles	Utilisateur	En tant qu'utilisateur, je veux publier un article en spécifiant tous les détails (prix, marque, matière, catégorie, couleur, taille, état, description), le modifier ou le supprimer	1
	Interface pour afficher le catalogue des articles	Utilisateur, Visiteur	En tant qu'utilisateur ou visiteur, je parcourir le catalogue et voir tous les articles disponibles	1
	Interface pour afficher les détails de chaque article	Utilisateur, Visiteur	En tant qu'utilisateur ou visiteur, je veux avoir la possibilité d'accéder à une fiche détaillée pour chaque article, incluant toutes ses caractéristiques.	2
	Recherche des articles	Utilisateur, Visiteur	En tant qu'utilisateur ou visiteur, je veux rechercher n'importe quel article par son nom	2

Sprint 3	Gestion favoris	Utilisateur	En tant qu'utilisateur, je veux avoir la possibilité d'ajouter ou de supprimer un article dans les favoris	2
	Gestion panier	Utilisateur	En tant qu'utilisateur, je veux avoir la possibilité d'ajouter ou de supprimer un article dans mon panier	2
	Passer la commande	Utilisateur	En tant qu'utilisateur, je dois pouvoir valider la commande et passer au paiement	2
Sprint 4	Messagerie	Utilisateur	En tant qu'utilisateur, je veux avoir la possibilité de communiquer avec n'importe quel autre utilisateur.	3
	Évaluation des articles	Utilisateur	En tant qu'utilisateur, je veux évaluer et commenter tous les articles disponibles sur l'application.	3
	Tableau de bord pour l'administrateur	Administrateur	En tant qu'administrateur, je veux avoir la possibilité de superviser et de contrôler le bon fonctionnement de l'application.	3
	Notifications	Utilisateur	En tant qu'utilisateur, je veux être notifié en temps réel des actions importantes liées à mon activité sur l'application.	3

Sprint 5	Recommandation d'articles personnalisés via l'IA.	Utilisateur	En tant qu'utilisateur, je veux avoir la possibilité de parcourir mes recommandations personnalisées, générées automatiquement par le système en fonction de mes interactions précédentes.	4
Sprint 6	Mise en place d'un chatbot intelligent via l'IA.	Utilisateur	En tant qu'utilisateur, je veux en mesure d'interroger le chatbot pour obtenir des informations ou de l'assistance concernant les fonctionnalités et services proposés par l'application.	4
Sprint 7	Recherche des articles par image via l'IA	Utilisateur	En tant qu'utilisateur, je veux pouvoir effectuer une recherche d'articles en insérant une image; le système devra alors analyser visuellement l'image et me proposer les articles similaires disponibles sur la plateforme.	4

TABLE 2.2 – Product Backlog

## 5 Décomposition des sprints

La durée de chaque sprint est spécifiquement définie afin de garantir une progression régulière du projet. Cette décomposition est illustrée dans la figure 2.2.

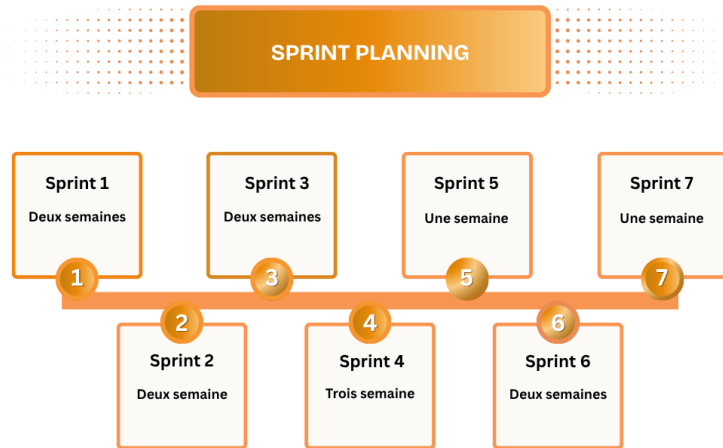


FIGURE 2.2 – Sprint planning

### 5.1 Charte graphique

La charte graphique est un document fondamental pour toute application souhaitant affirmer son identité visuelle. Elle englobe tout ce qui concerne l'esthétique de l'application, tels que : logo, couleurs, symboles, etc.[20]

### 5.2 Le logo

Le logo constitue le repère visuel central de la marque. Il doit se distinguer par son unicité, sa lisibilité et sa capacité à marquer les esprits. Notre logo est représenté dans la figure 2.3



FIGURE 2.3 – Logo de l'application

### 5.2.1 Signification du logo

- **Le cintre** : Il représente directement l'univers de l'habillement et de la mode, et évoque la seconde main, car il est associé à des vêtements suspendus, souvent vus dans des friperies, boutiques ou dressings.
- **La flèche** : Symbolise la réutilisation, la durabilité et la transformation, et renforce l'idée que ReWear donne une seconde vie aux vêtements.
- **La feuille et la branche** : Évoquent la nature, l'écologie et la soutenabilité, symbolisant à la fois la fraîcheur, le renouveau, le respect de l'environnement, ainsi que la croissance et le potentiel de transformation dans la consommation textile.
- **Le texte ReWear** : Jeu de mots entre "re-wear" (remettre, porter à nouveau) et "wear" (vêtement).
- **Le slogan "SECONDE MAIN, PREMIER STYLE"** : L'expression "Seconde Main" met en avant la seconde main, tandis que "Premier Style" affirme que même la seconde main peut être tendance, élégante et haut de gamme.

### 5.3 Palette de couleurs utilisée

La palette de couleurs que nous avons choisie joue un rôle essentiel dans l'identité visuelle de notre marque. Elle reflète nos valeurs, crée une ambiance unique et renforce l'expérience utilisateur. Chaleureuse, naturelle et équilibrée, elle incarne notre engagement pour une mode seconde main à la fois éthique, élégante et accessible. La palette de couleurs est présentée dans la figure 2.4.



FIGURE 2.4 – La palette de couleurs

### 5.4 Outils et langages utilisés

Pour développer notre application, nous avons choisi des technologies modernes, fiables et adaptées à nos besoins. Chaque outil a été sélectionné pour garantir une expérience fluide, sécurisée et performante, aussi bien pour les utilisateurs que pour les administrateurs.

#### 5.4.1 Niveau conceptuel

- **Draw.io** : Est un outil qui permet de concevoir des diagrammes UML. Il nous a aidés à structurer efficacement l'architecture de l'application, ce qui a contribué à la réalisation

d'une solution complète et cohérente.

- **Figma** : Est un outil de conception d'interface utilisateur basé sur le cloud. Il nous a permis de structurer efficacement notre travail tout au long du processus de développement.[21]



**draw.io**

FIGURE 2.5 – Draw.io



FIGURE 2.6 – Figma

### 5.4.2 Niveau développement

La pile MERN regroupe les technologies MongoDB, Express, React et Node.js, toutes basées sur JavaScript, pour créer des applications web modernes. Elle offre une base de données flexible, une interface utilisateur réactive et une fondation solide pour des projets performants.[22, 23]

#### a) Développement Frontend

- **React.Js** : Est une bibliothèque JavaScript utilisée pour construire des interfaces utilisateur de manière efficace. Elle repose sur des composants réutilisables, ce qui facilite le développement et évite de répéter du code. Grâce à ses caractéristiques, nous avons pu créer une application fluide, dynamique et facile à maintenir.[24]
- **Tailwind CSS** : Est un framework CSS basé sur une approche utilitaire, qui propose des classes prédéfinies à utiliser directement dans le code HTML pour concevoir et styler rapidement des interfaces. Nous avons utilisé Tailwind CSS dans notre application pour créer une interface moderne, responsive et facile à maintenir.[25]

#### b) Développement Backend

- **Node.Js** : Node.js est un environnement d'exécution open-source qui permet d'exécuter du JavaScript côté serveur. Rapide et léger, il est idéal pour les applications en temps réel et évolutives. Nous l'avons utilisé pour gérer efficacement le développement de la partie serveur de notre application, assurant des échanges de données rapides et une architecture performante et fiable. [26]
- **Express.Js** : Est un framework backend léger pour Node.js, conçu pour créer des applications web rapides et évolutives. Il offre une base solide pour structurer le code serveur, avec une gestion simplifiée des routes, des requêtes HTTP et des intergiciels (middleware). Il a été intégré dans notre projet pour développer l'architecture backend de l'application.[27]
- **MongoDB** : Est une base de données NoSQL orientée document, qui se distingue des bases de données relationnelles par sa flexibilité et sa capacité à gérer des données non structurées. Elle stocke les informations sous forme de documents JSON, ce qui facilite leur manipulation côté serveur.[28]

Nous avons utilisé MongoDB dans notre application pour gérer les données de manière souple, évolutive et adaptée aux besoins de notre architecture.

- **Cloudinary** : Est une plateforme cloud qui permet de gérer facilement les images et les vidéos dans une application. Elle offre des fonctionnalités comme le stockage, la transformation, l'optimisation et la diffusion rapide des fichiers via un CDN. Nous l'avons utilisé pour gérer et optimiser les contenus visuels de notre application de manière fluide et performante. [29]
- **Socket.io** : Est une bibliothèque JavaScript qui facilite la communication en temps réel entre un client (comme un navigateur) et un serveur. Elle est idéale pour des fonctionnalités comme le chat en direct et les notifications instantanées. Dans notre application, nous l'avons utilisé pour assurer des interactions fluides et instantanées entre les utilisateurs et le serveur. [socketio\_docs, 30]
- **JWT (JSON Web Token)** : Est une méthode sécurisée pour transmettre des informations entre un client et un serveur sous forme d'objet JSON signé. Généralement utilisé pour l'authentification et l'autorisation dans les applications web, JWT permet de vérifier l'identité d'un utilisateur sans stockage de session côté serveur. Dans notre application, nous l'avons intégré pour gérer l'authentification des utilisateurs, assurant ainsi des échanges sécurisés et une gestion efficace des sessions. [31]
- **Python** : Python est un langage de programmation largement utilisé dans les applications Web, le développement de logiciels, la science des données et le machine learning (ML)[aws\_python]. Nous avons choisi ce langage pour l'intégration de l'intelligence artificielle dans notre applications, notamment les recommandations, le chatbot et la recherche par image.

## 6 Conclusion

En conclusion, ce chapitre a permis de clarifier l'organisation de l'équipe Scrum ainsi que les rôles et responsabilités de chacun de ses membres, tout en identifiant les acteurs clés et en définissant les cas d'utilisation qui composent notre Product Backlog. Nous avons ensuite découpé le travail en plusieurs sprints, permettant une réponse itérative et progressive aux besoins des utilisateurs. Enfin, nous avons présenté la charte graphique de l'application ainsi que les outils qui soutiendront son développement. Le chapitre suivant sera consacré au Sprint 1, au cours duquel nous débuterons le développement des fonctionnalités prioritaires pour concrétiser notre application.

# Chapitre 3

## Sprint 1,2

# Mise en place des fonctionnalités de base

## 1 Introduction

Ce chapitre présente la mise en œuvre des deux premiers sprints de notre projet, répartis sur quatre semaines.

Le Sprint 1 s'est focalisé sur l'inscription des visiteurs, l'authentification des utilisateurs et la gestion du profil utilisateur, formant ainsi le socle sécuritaire et personnel de l'application. Le Sprint 2 a poursuivi cette dynamique en introduisant la gestion de la mise en vente, la consultation du catalogue d'articles, ainsi que la recherche et la navigation dans les produits.

Dans les sections suivantes, nous détaillerons les fonctionnalités développées lors de ces deux sprints, en présentant les diagrammes de séquence et de classes associés pour chaque étape, afin d'illustrer clairement l'architecture et les interactions clés de notre système.

## 2 Sprint 1

### Gestion des Comptes utilisateurs

#### 2.1 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation illustré dans la figure 3.1 montre les interactions entre les différents acteurs (Visiteur, Utilisateur) et les trois fonctionnalités développées lors du sprint 1.

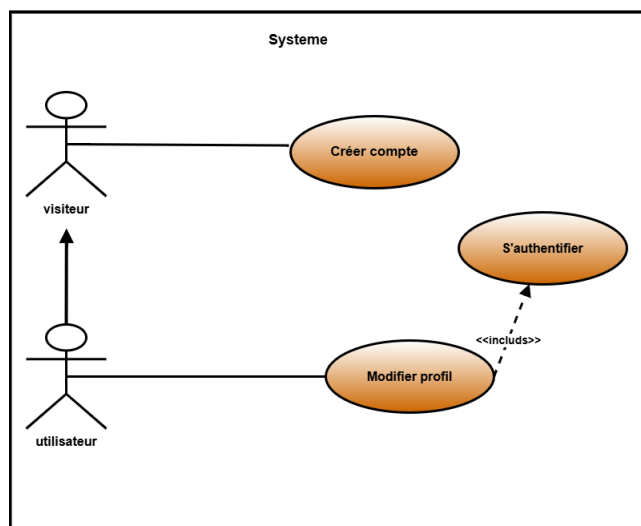


FIGURE 3.1 – Diagramme de cas d’utilisation pour **Sprint 1**

## 2.2 Product Backlog

Dans un premier temps, nous présenterons le backlog du sprint 1, listant l’ensemble des tâches à réaliser ainsi que leurs estimations, comme détaillé dans le tableau ci-dessous.

Fonctionnalité	Tâche	Responsable	Priorité	Statut
<b>Inscription</b>	Créer la page d’inscription avec formulaire (nom, prénom, email, mot de passe, photo)	Thinhinane	Haute	À faire
	Implémenter l’upload et la validation de la photo de profil	Lydia	Moyenne	À faire
	Mettre en place la validation des données d’inscription côté client	Thinhinane	Haute	À faire
	Implémenter la logique d’inscription côté serveur et base de données	Lydia	Haute	À faire
	Configurer le système d’envoi d’e-mail d’activation	Thinhinane	Haute	À faire

<b>Fonctionnalité</b>	<b>Tâche</b>	<b>Responsable</b>	<b>Priorité</b>	<b>Statut</b>
	Créer la page et logique de validation du token d'activation	Thinhinane	Haute	À faire
<b>Authentification</b>	Concevoir la page de connexion (email, mot de passe)	Lydia	Moyenne	À faire
	Implémenter la logique d'authentification côté serveur	Lydia	Haute	À faire
	Gérer les sessions utilisateur et redirection post-connexion	Thinhinane	Moyenne	À faire
<b>Gestion du profil</b>	Créer la page d'affichage du profil utilisateur	Lydia	Moyenne	À faire
	Implémenter la modification des informations du profil	Thinhinane	Moyenne	À faire
	Permettre la modification de la photo de profil	Lydia	Moyenne	À faire

TABLE 3.1 – Product Backlog du sprint 1

### 2.3 Décomposition du sprint

Pendant ce sprint, nous développons les fonctionnalités pour livrer un incrément. La figure 3.2 présente la répartition du temps du sprint 1.

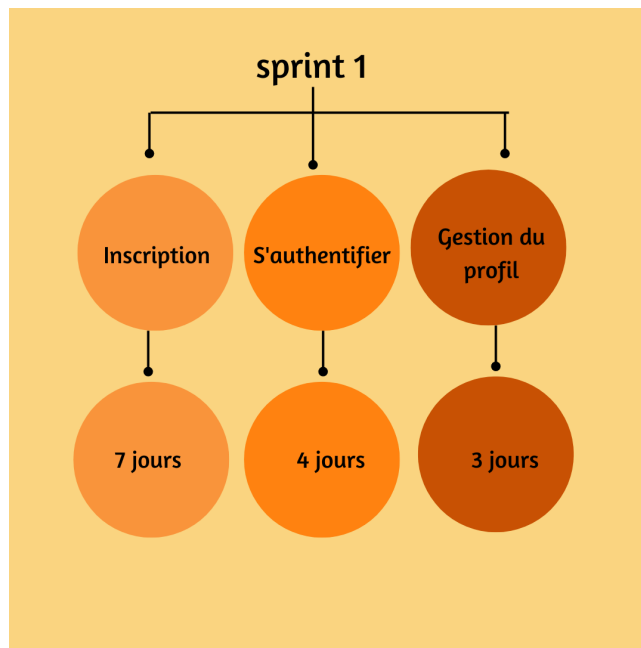


FIGURE 3.2 – Décomposition du sprint 1

## 2.4 présentation des diagrammes de séquences

Nous allons présenter une description détaillée des fonctionnalités :

- Inscription.
- Authentification.
- Gestion du Profil.

### 2.4.1 Fonctionnalité « Inscription »

#### Scénario principal (cas normal)

1. L'utilisateur visite la page d'inscription.
2. Il remplit le formulaire par son nom complet, son adresse e-mail, son mot de passe, et télécharge une photo .
3. Il soumet le formulaire.
4. Le système valide la conformité des données (format e-mail, mots de passe conformes, photo acceptable).
5. Le système crée un compte avec statut « en attente d'activation ».
6. Un e-mail d'activation est envoyé à l'adresse fournie via le protocole **SMTP** (Simple Mail Transfer Protocol).
7. L'utilisateur consulte sa boîte mail et clique le lien d'activation.
8. Le compte est activé et l'utilisateur est redirigé vers la page de connexion.

#### Scénarios alternatifs (erreurs)

- Données manquantes ou mal formées (ex : e-mail invalide, mot de passe trop court)  
→ Le système affiche un message d'erreur et invite à corriger les champs.
- Photo invalide ou non conforme(format non supporté,taille excessive)  
→ Le système affiche un message d'erreur et invite à télécharger une nouvelle photo respectant les critères.

- Adresse e-mail déjà utilisée  
→ Le système affiche un message indiquant que l'adresse est déjà enregistrée.
- Problème lors de l'envoi de l'e-mail d'activation  
→ Le système affiche un message d'erreur et propose de renvoyer l'e-mail.
- Lien d'activation expiré ou invalide  
→ Le système affiche un message d'erreur et propose une nouvelle demande d'activation.
- Problème lors de l'enregistrement en base  
→ Le système informe l'utilisateur et propose de réessayer.

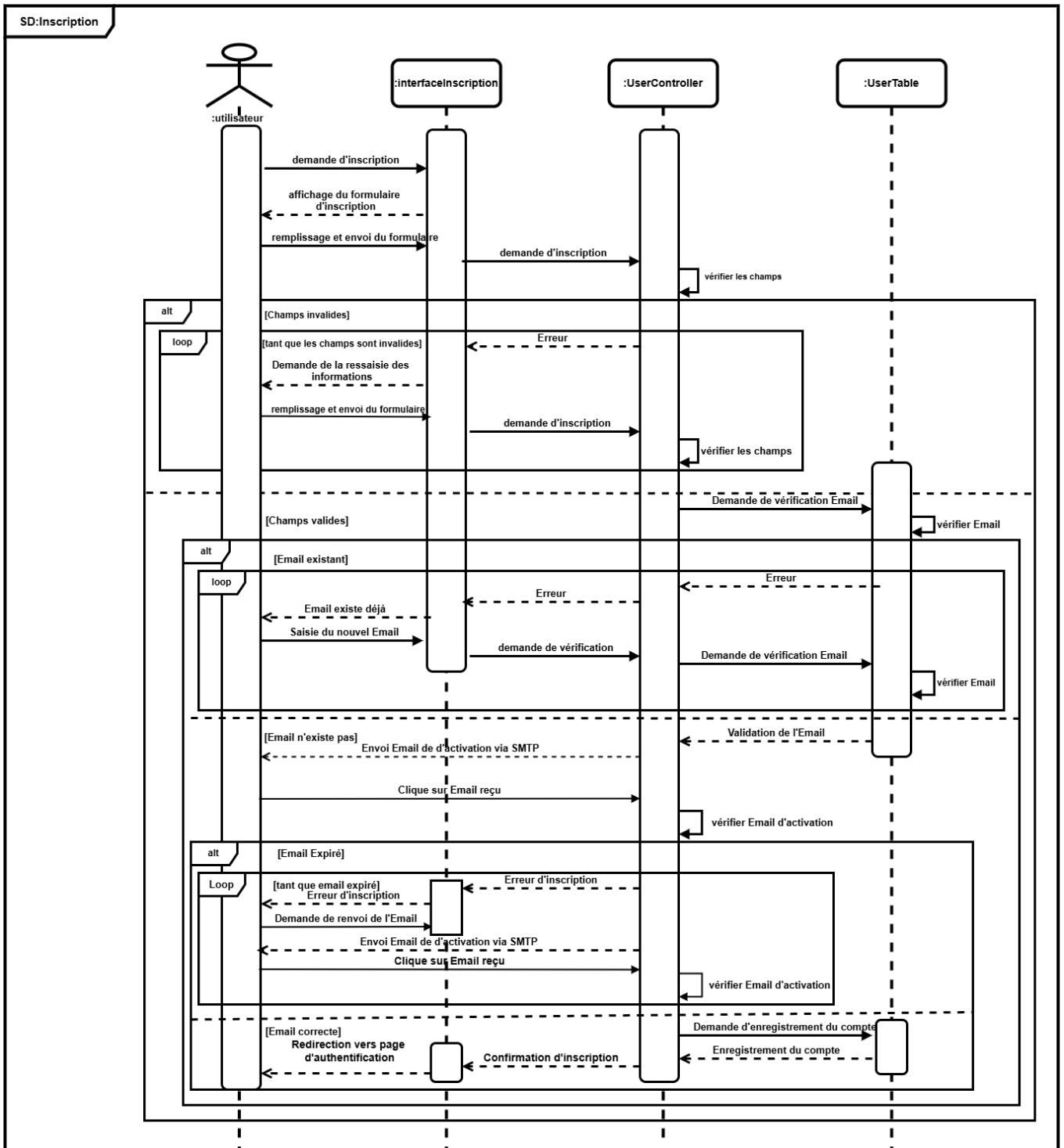


FIGURE 3.3 – Diagramme de séquence pour la fonctionnalité **Inscription**

## 2.4.2 Fonctionnalité « Authentification »

### Scénario principal (cas normal)

1. L'utilisateur accède à la page de connexion.
2. Il saisit son adresse e-mail et son mot de passe.
3. Le système vérifie la validité des informations.
4. Si les informations sont correctes et que le compte est activé, l'utilisateur est connecté.
5. Il est redirigé vers la page d'accueil avec accès aux fonctionnalités privées.

### Scénarios alternatifs (erreurs)

- Identifiants incorrects (e-mail ou mot de passe)  
→ Message d'erreur : « Identifiants invalides », invitation à ressaisir.
- compte non activé(email non confirmé)  
→ Message d'errur :« Compte bloqué ou suspendu».
- Champs vides ou mal formatés  
→ Message d'errur :«Champs manquant ou mal formatés»,invitation à remplir tous les champs requis.

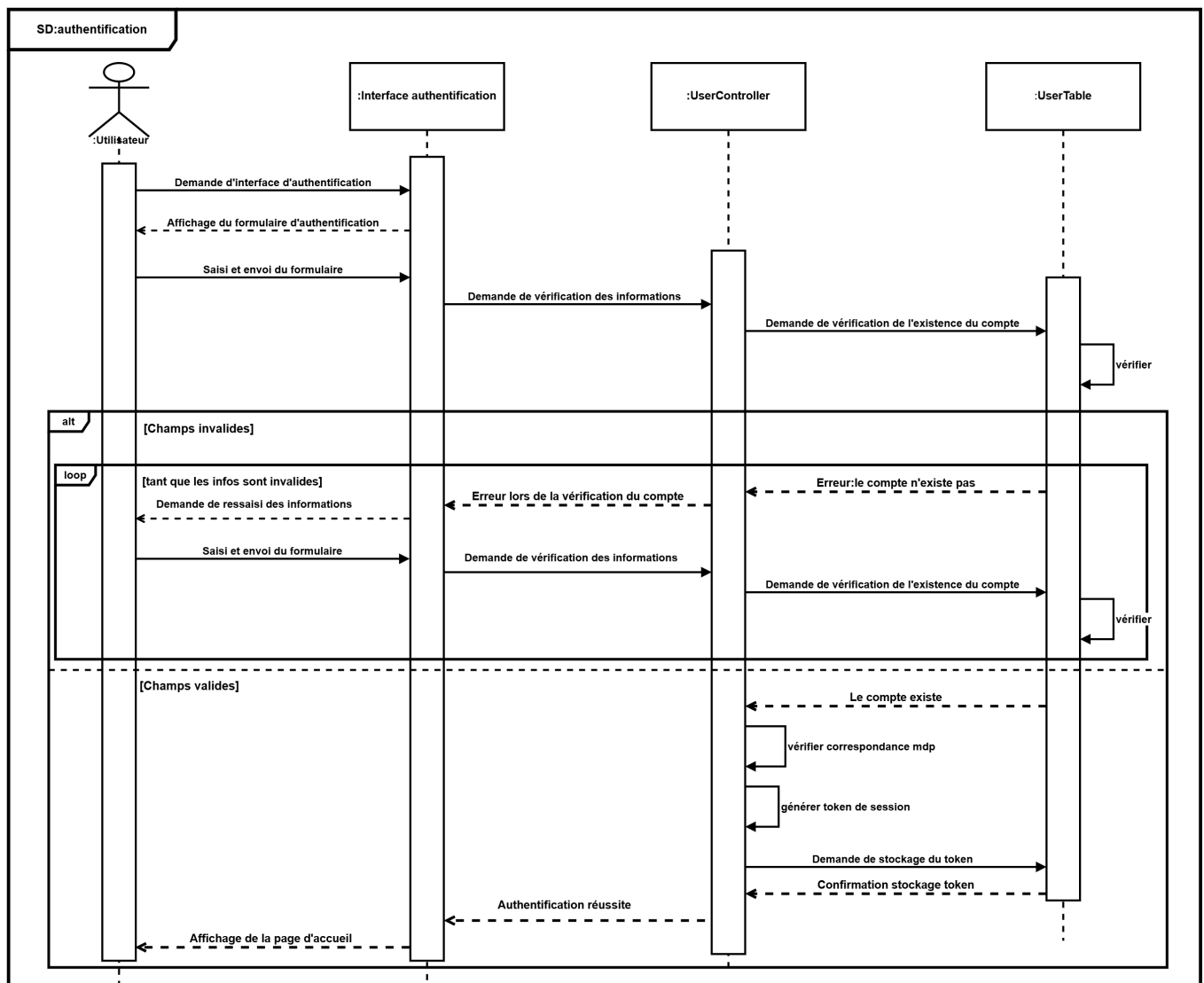


FIGURE 3.4 – Diagramme de séquence pour la fonctionnalité **Authentification**

### 2.4.3 Fonctionnalité « Modification du profil »

#### Scénario principal (cas normal)

1. L'utilisateur connecté accède à sa page de profil.
2. Il clique sur le bouton « Modifier profil ».
3. Un formulaire apparaît avec ses informations actuelles (nom, téléphone, Email (non modifiable) ,localisation,photo).
4. Il modifie les champs souhaités.
5. Il soumet le formulaire.
6. Le système valide les nouvelles données (ex : format téléphone, localisation valide).
7. Les informations sont mises à jour dans la base de données.
8. Le profil est affiché avec les données modifiées.

#### Scénarios alternatifs (erreurs)

- Données invalides dans le formulaire (ex : téléphone mal formaté)  
→ Message d'erreur, correction demandée.
- Photo invalide ou non conforme(format non supporté,taille excessive)  
→ Le système affiche un message d'erreur et invite à télécharger une nouvelle photo respectant les critères.
- Problème lors de l'enregistrement en base  
→ Le système informe l'utilisateur et propose de réessayer.

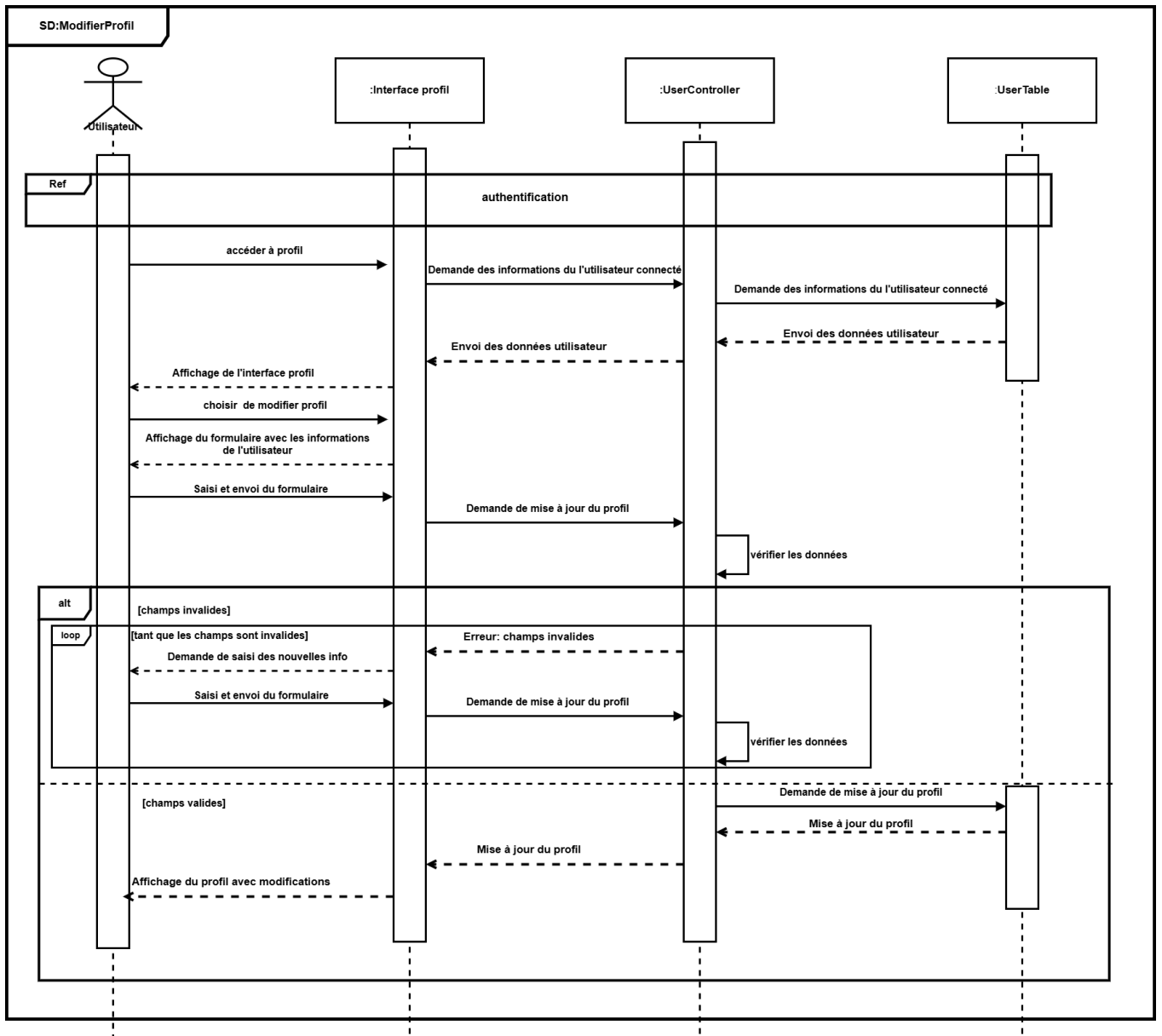


FIGURE 3.5 – Diagramme de séquence pour la fonctionnalité **Gestion de profil**

## 2.5 diagramme de classes

Le diagramme de classes présenté dans la figure 3.6 offre une vue d'ensemble des classes constituant les éléments du sprint 1.

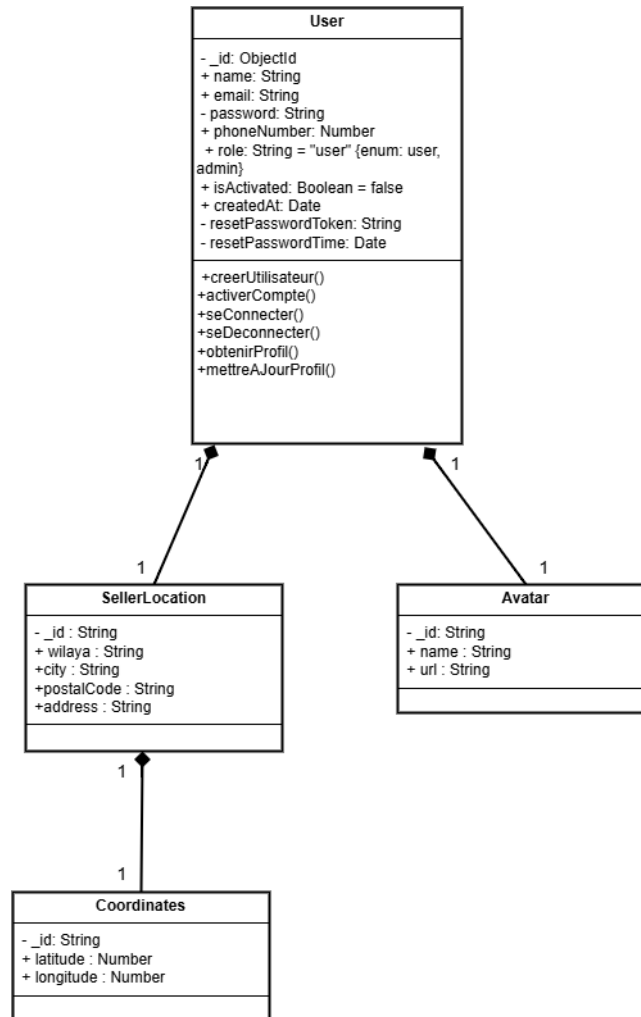


FIGURE 3.6 – diagramme de classes **sprint 1**

## 2.6 Vue des Interfaces Utilisateur

Dans cette section, nous présentons les interfaces graphiques essentielles qui composent le sprint 1

### 2.6.1 Interface d'inscription

La figure 3.7 illustre l'interface d'inscription qui permet à un nouvel utilisateur de créer un compte en fournissant les informations nécessaires, telles que le nom, l'adresse email, un mot de passe sécurisé et une photo.

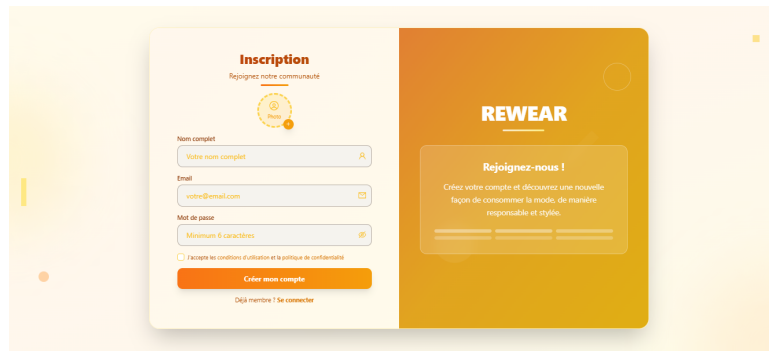


FIGURE 3.7 – Interface d’inscription

### 2.6.2 Interface d’authentification

L'utilisateur accède aux services de l'application via une interface d'authentification qui est Représenté dans la figure 3.8 en renseignant son adresse email ainsi que son mot de passe.

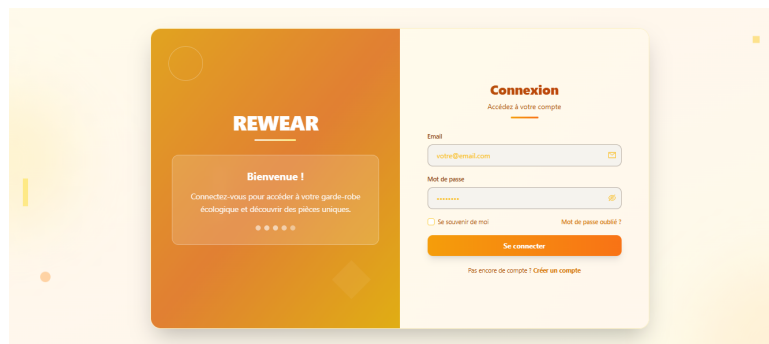


FIGURE 3.8 – Interface d’authentification

### 2.6.3 Interface de Gestion du profil

La figure 3.12 met en évidence l'interface offre à l'utilisateur la possibilité de mettre à jour ses informations personnelles, comme le nom, la localisation, la photo et le numéro de téléphone, afin de maintenir son profil à jour.

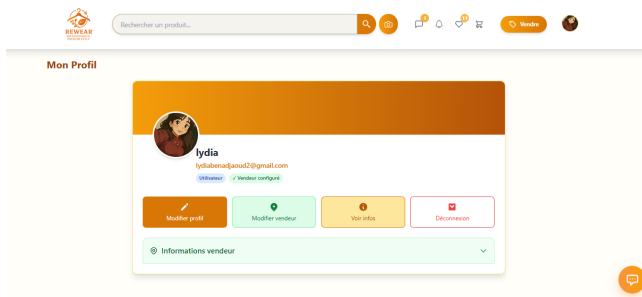


FIGURE 3.9 – Interface du profil

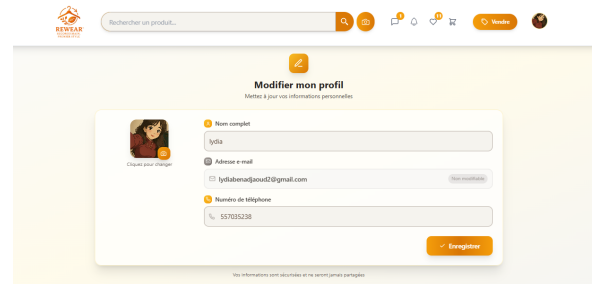


FIGURE 3.10 – Interface de modification du profil (avatar,numéro de téléphone)

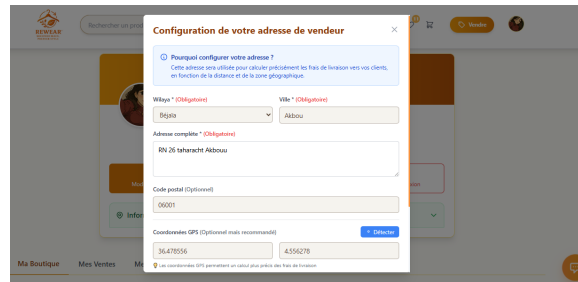


FIGURE 3.11 – Interface de modification du profil (Adresse )

FIGURE 3.12 – Ensemble des interfaces de gestion du profil utilisateur

## 3 Sprint 2

### Gestion de la vente et recherche d'articles

#### 3.1 Diagramme de cas d'utilisation

La figure 3.13 expose le diagramme de cas d'utilisation du Sprint 2 illustrant les interactions entre les différents acteurs (Utilisateur, Visiteur) et les fonctionnalités développées lors de ce sprint.

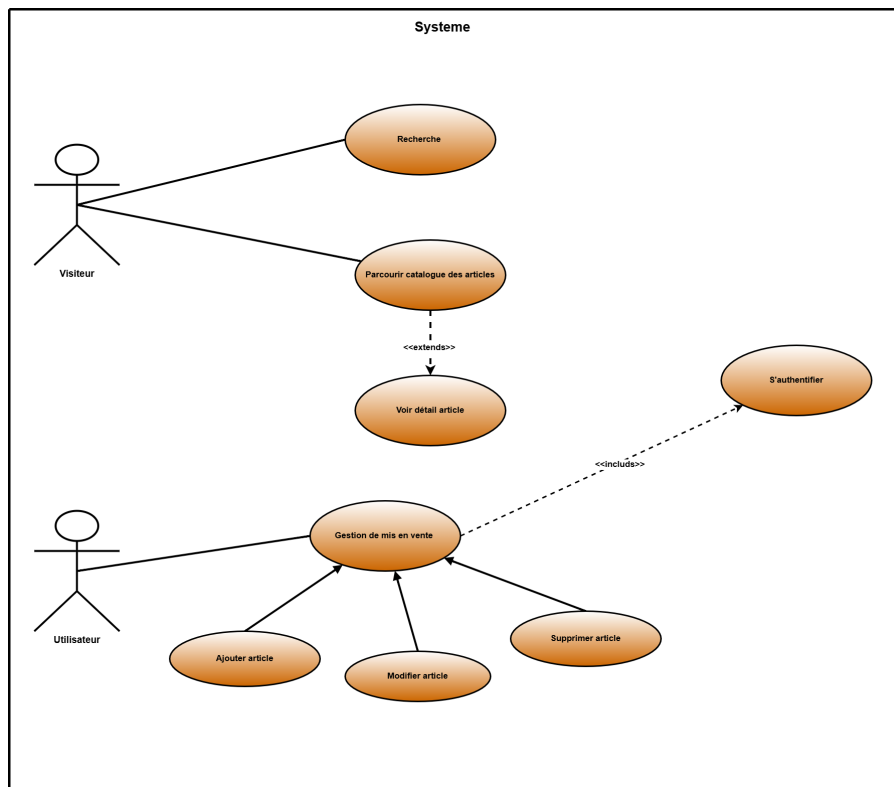


FIGURE 3.13 – Diagramme de cas d'utilisation pour **Sprint 2**

## 3.2 Product Backlog

Le tableau ci-dessous présente le backlog détaillé du Sprint 2, listant les fonctionnalités à développer, les tâches associées, les responsables, les priorités et le statut.

Fonctionnalité	Tâche	Responsable	Priorité	Statut
Gestion de mise en vente	Concevoir le formulaire de mise en vente (prix, marque, catégorie, couleur, taille, état, description)	Thinhinane	Haute	À faire
	Gérer la création, modification et suppression d'articles en base	Lydia	Haute	À faire
	Mettre à jour l'interface utilisateur après ajout d'article	Thinhinane	Moyenne	À faire
<i>Suite à la page suivante...</i>				

<b>Fonctionnalité</b>	<b>Tâche</b>	<b>Responsable</b>	<b>Priorité</b>	<b>Statut</b>
Catalogue des articles	Développer la Home Page affichant tous les articles	Lydia	Haute	À faire
	Implémenter la navigation par catégories et sous-catégories	Thinhinane	Haute	À faire
	Développer les endpoints backend pour récupérer les articles selon catégories, sous-catégories et filtres	Lydia	Haute	À faire
	Traiter les filtres simples et multi-critères dans les contrôleurs backend	Lydia	Haute	À faire
	Gérer l’affichage dynamique des résultats filtrés côté frontend	Thinhinane	Moyenne	À faire
Détail d’article	Concevoir la page React pour afficher les détails d’un article	Lydia	Haute	À faire
	Permettre la navigation depuis le catalogue vers la fiche détaillée	Lydia	Haute	À faire
	Développer les endpoints backend pour fournir les détails d’un article	Thinhinane	Haute	À faire
Recherche des articles	Développer la recherche par nom d’article	Lydia	Haute	À faire
	Optimiser la gestion des erreurs de saisie dans la recherche	Thinhinane	Moyenne	À faire

TABLE 3.2 – Product Backlog du sprint 2

### 3.3 Décomposition du sprint

Ce sprint est réparti sur une durée de deux semaines, avec une allocation du temps dédiée à la conception, au développement et aux tests des différentes fonctionnalités listées précédemment. Ceci est représenté dans la figure 3.14

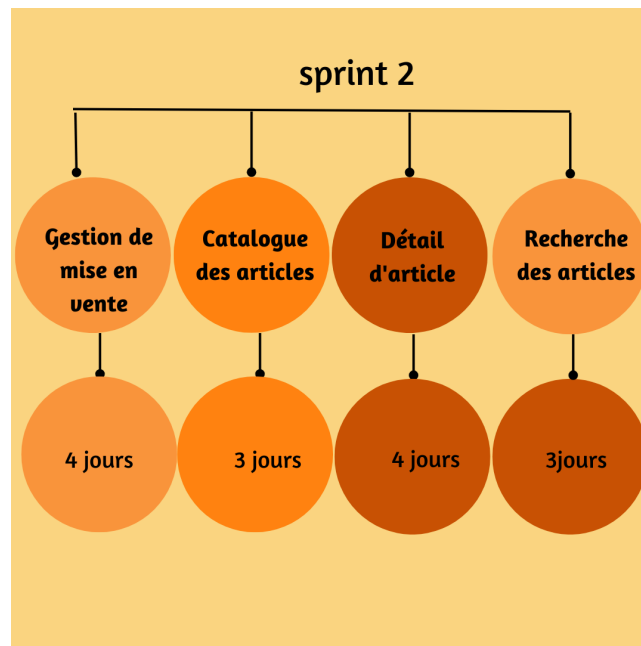


FIGURE 3.14 – Décomposition du Sprint 2

### 3.4 Présentation des diagrammes de séquence

Nous présentons ci-dessous la description détaillée des fonctionnalités majeures de ce sprint, accompagnée des diagrammes de séquence illustrant le déroulement des interactions système-utilisateur.

### 3.5 Fonctionnalité « Gestion de mise en vente »

#### 3.5.1 Fonctionnalité ajouter article

- **Scénario principal (cas normal)**

1. L'utilisateur connecté accède au formulaire de mise en vente.
2. Il renseigne toutes les informations nécessaires sur l'article (prix, marque, matière, catégorie, couleur, taille, état, description, photos).
3. Il soumet le formulaire.
4. Le système valide les données saisies.
5. L'article est enregistré dans la base de données.
6. Le système confirme la mise en vente de l'article à l'utilisateur.

- **Scénarios alternatifs (erreurs)**

- Données manquantes ou mal formatées  
→ Le système affiche un message d'erreur demandant la correction.
- Photo invalide ou non conforme (format non supporté, taille excessive)  
→ Le système affiche un message d'erreur et invite à télécharger une nouvelle photo respectant les critères.

- Problème lors de l'enregistrement en base  
→ Le système informe l'utilisateur et propose de réessayer.

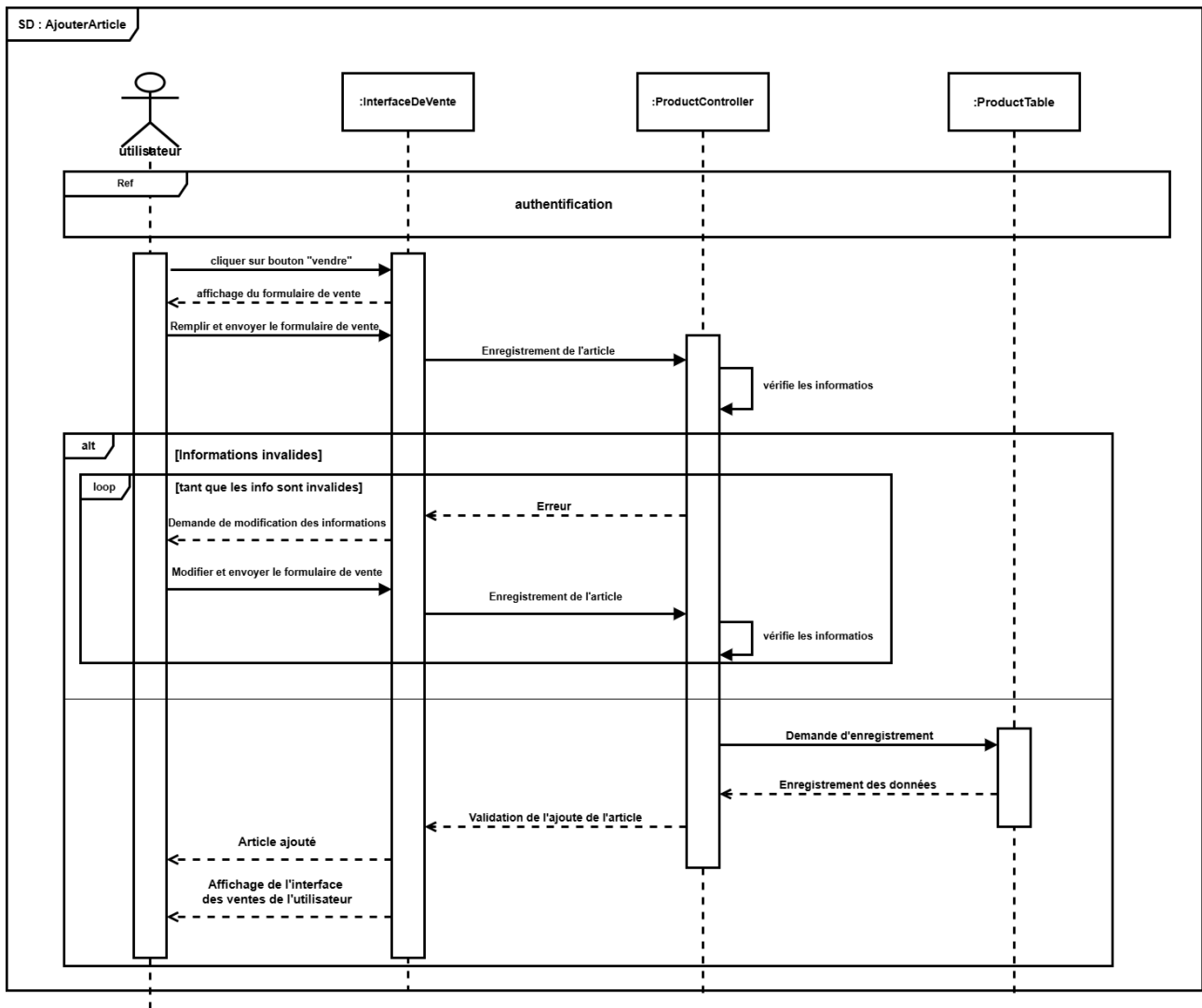


FIGURE 3.15 – Diagramme de séquence pour la fonctionnalité **Ajouter article**

### 3.5.2 Fonctionnalité Modifier et supprimer un article

- **Scénario principal (cas normal)**

1. L'utilisateur accède à sa boutique personnelle listant tous ses articles.
2. Il sélectionne un article pour modification ou suppression.
3. Pour modification, il accède à un formulaire pré-rempli, modifie les données et soumet.
4. Pour suppression, il confirme la suppression de l'article.
5. Le système valide et applique la modification ou suppression en base de données.
6. Le système confirme l'opération et met à jour l'affichage de la boutique personnelle.

- **Scénarios alternatifs (erreurs)**

- Données invalides lors de la modification  
→ Le système affiche un message d'erreur et invite à corriger.
- Problème lors de la mise à jour ou suppression en base  
→ Le système informe l'utilisateur et propose de réessayer.

### 3.5.3 Modifier et supprimer un article

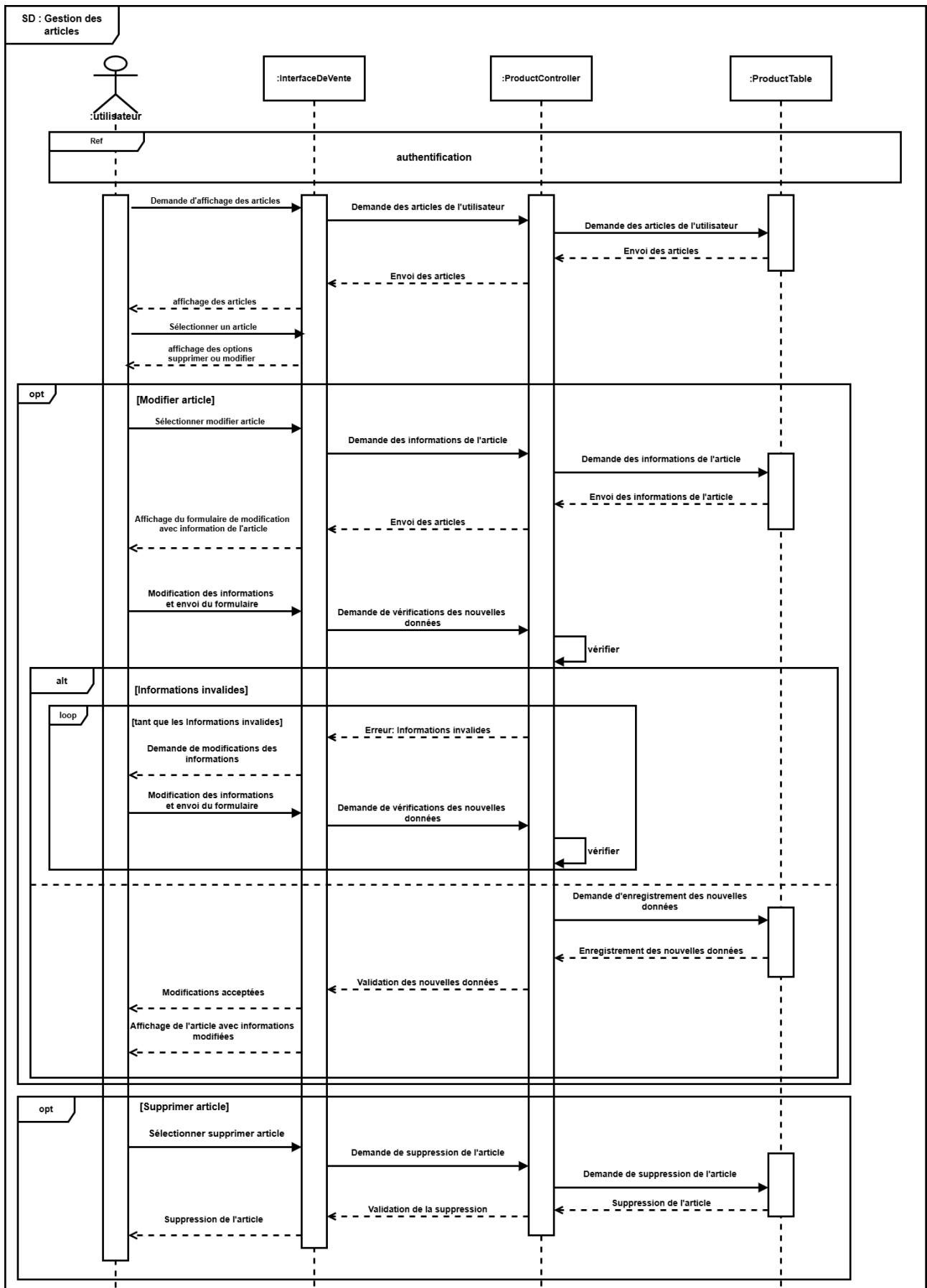


FIGURE 3.16 – Diagramme de séquence pour la fonctionnalité **Modifier ou supprimer article**

### **3.6 Fonctionnalité « Consultation du catalogue »**

#### **Scénario principal (cas normal)**

1. L'utilisateur ou visiteur accède à la page catalogue(HomePage).
2. Le système affiche la liste paginée des articles disponibles.
3. L'utilisateur peut filtrer ou trier les articles selon certains critères.

#### **3.7 Scénarios alternatifs (erreurs)**

- Les filtres ne retournent aucun résultat  
→ Le système affiche un message d'erreur «Aucun article trouvé» .

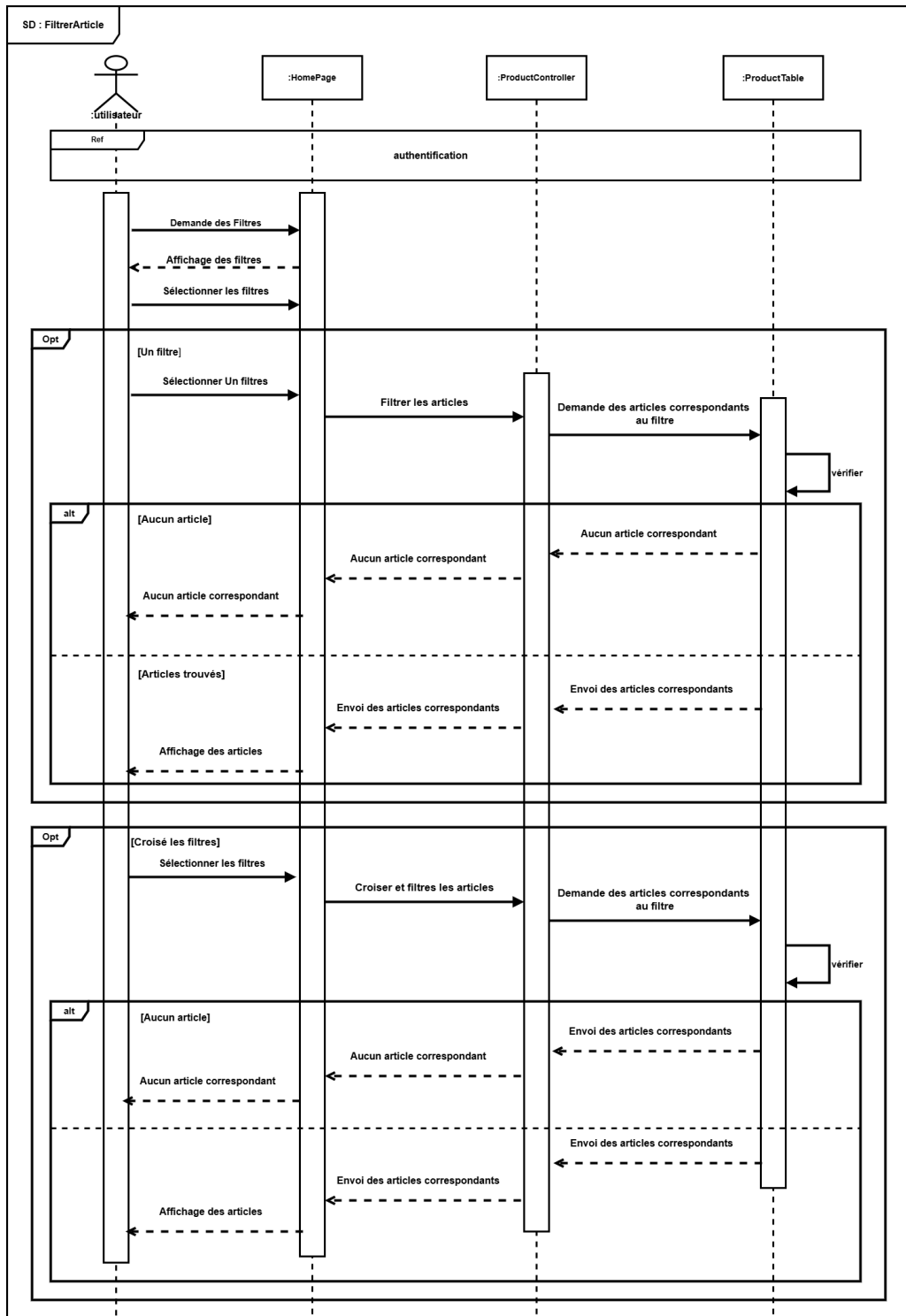


FIGURE 3.17 – Diagramme de séquence pour la fonctionnalité **parcourir le catalogue en filtrants les articles**

### 3.8 Fonctionnalité « Affichage des détails d'article »

#### Scénario principal (cas normal)

1. L'utilisateur clique sur un article dans le catalogue.
2. Le système récupère les détails complets de l'article.
3. Le système affiche la fiche détaillée avec toutes ses caractéristiques.

### 3.9 Fonctionnalité « Recherche des articles »

#### Scénario principal (cas normal)

1. L'utilisateur saisit un nom ou un mot-clé dans le champ de recherche.
2. Le système lance une requête pour retrouver les articles correspondants.
3. Le système affiche la liste des résultats pertinents.

### 3.10 Scénarios alternatifs (erreurs)

- La recherche ne retourne aucun résultat  
→ Le système affiche un message d'erreur «Aucun article trouvé» .

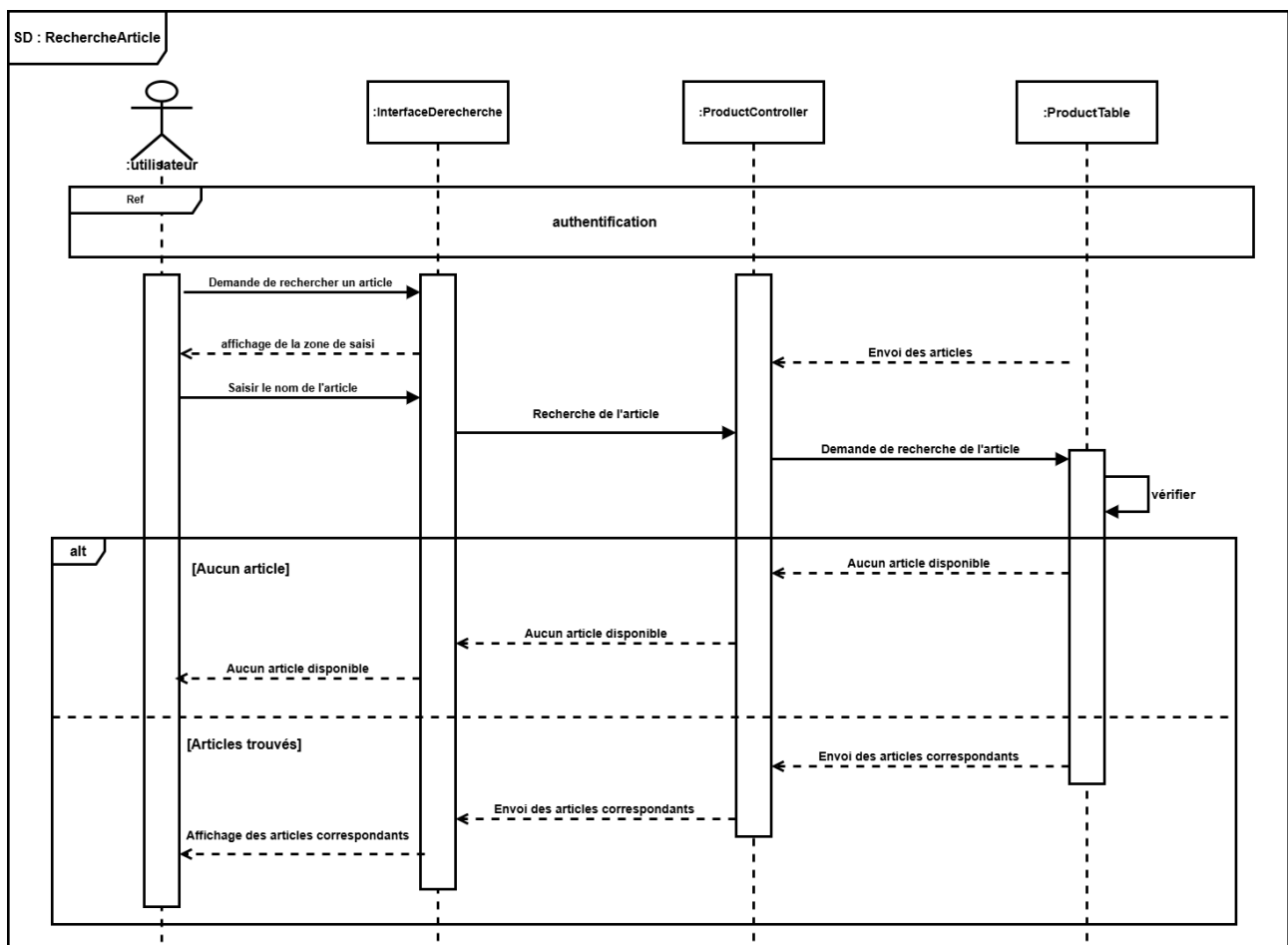


FIGURE 3.18 – Diagramme de séquence pour la fonctionnalité Recherche des articles

### 3.11 Diagramme de classes

Le diagramme de classes illustré dans la figure 3.19 représente les principales classes impliquées dans le Sprint 2, notamment les classes relatives aux articles, catégories et utilisateurs.

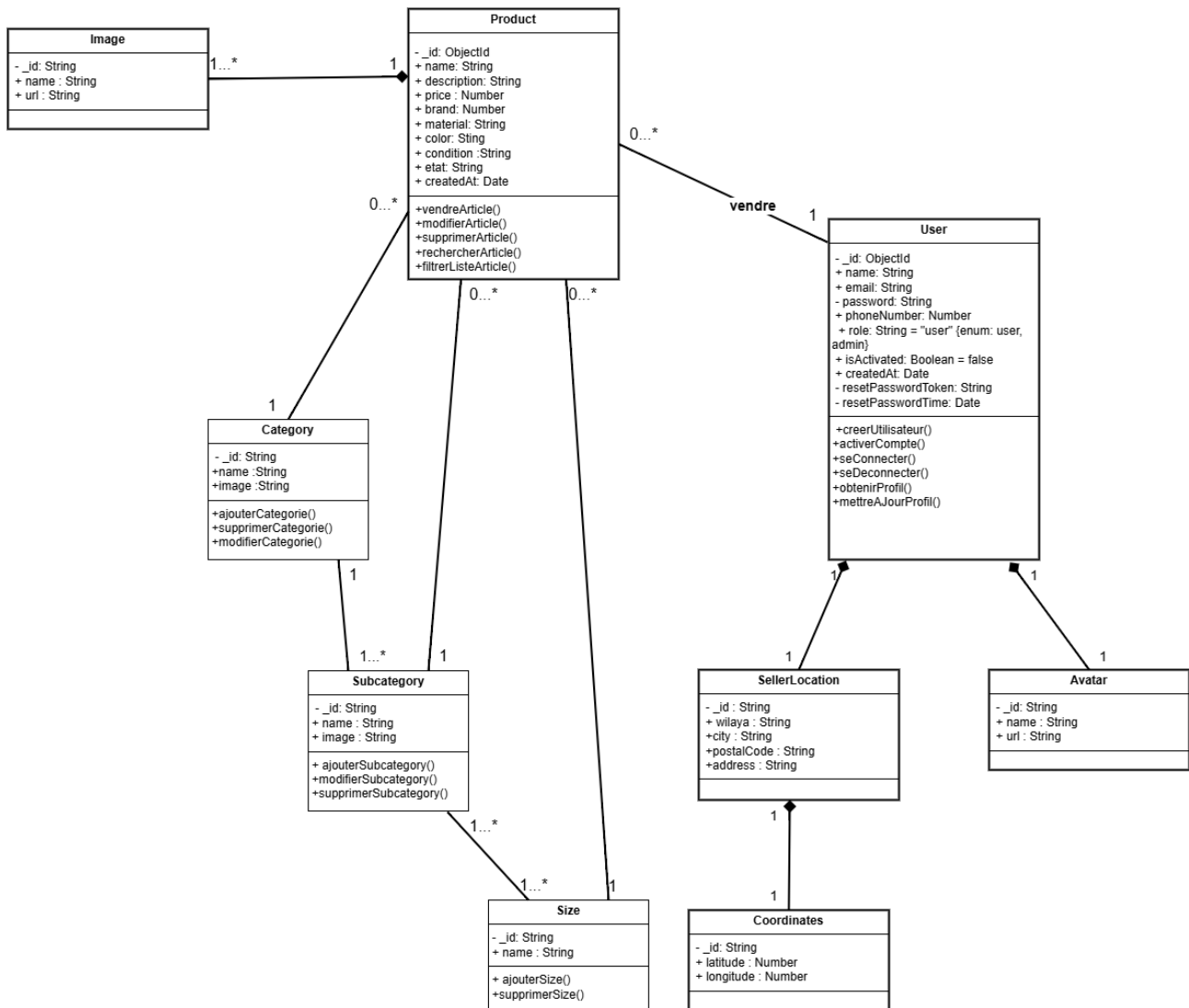


FIGURE 3.19 – Diagramme de classes **Sprint 2**

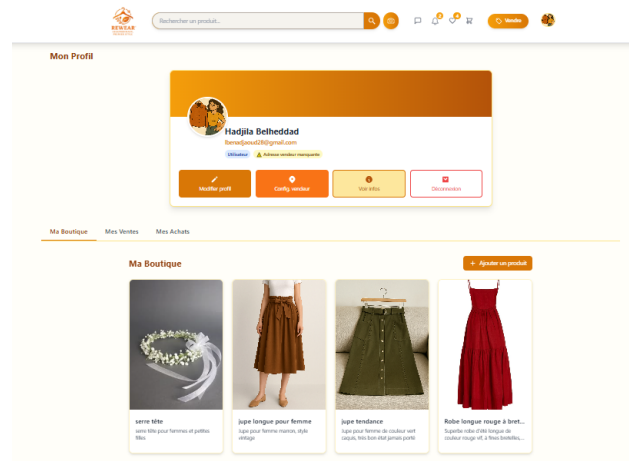
### 3.12 Vue des Interfaces Utilisateur

Dans cette section, nous présentons les interfaces graphiques principales développées lors du Sprint 2.

#### 3.12.1 Interface de gestion de mise en vente

L'interface représentée dans la figure 3.20 permet à l'utilisateur de créer, modifier ou supprimer un article à vendre en saisissant toutes ses caractéristiques.

(a) formulaire de mise en vente des articles



(b) interface d'affichage de la boutique de chaque utilisateur

FIGURE 3.20 – Mise en vente des articles

### 3.12.2 Interface du catalogue

L'interface illustrée dans la figure 3.21 permet de parcourir tous les articles disponibles, avec possibilité de filtrage et de recherche.

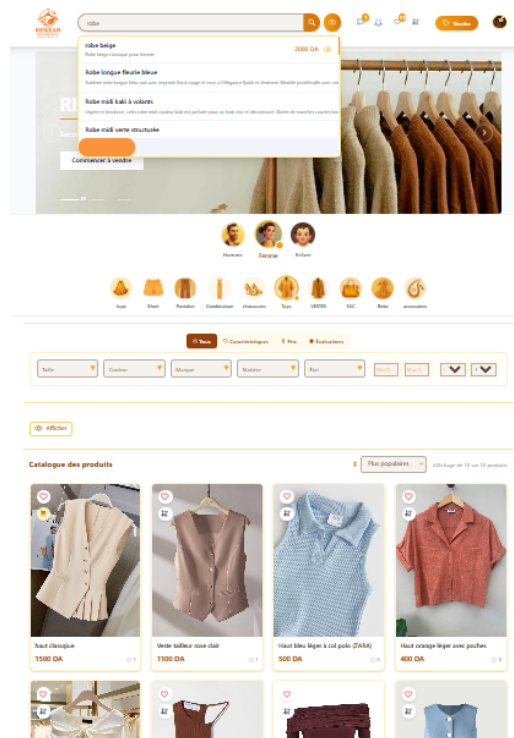


FIGURE 3.21 – Interface de consultation du catalogue

### 3.12.3 Interface fiche détaillée d'article

L'interface illustrée dans la figure 3.22 affiche toutes les informations détaillées d'un article sélectionné.

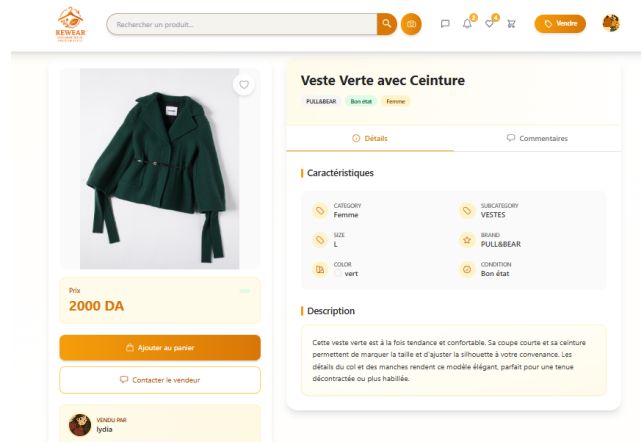


FIGURE 3.22 – Interface fiche détaillée d'article

## 4 Conclusion

En résumé, ce chapitre pose les fondations techniques et fonctionnelles du projet à travers le Sprint 1, axé sur l'inscription, l'authentification et la gestion du profil utilisateur, qui constituent le socle indispensable au bon fonctionnement de l'application. Le Sprint 2 a ensuite enrichi ce socle en intégrant la gestion de mise en vente, la consultation du catalogue et la recherche d'articles, apportant des fonctionnalités essentielles à l'expérience utilisateur. La présentation détaillée des implémentations, soutenue par les diagrammes associés, facilite la compréhension de la structure et du fonctionnement des composants clés. Le chapitre suivant abordera les Sprints 3 et 4, durant lesquels de nouvelles fonctionnalités seront développées pour approfondir et améliorer davantage le système.

# Chapitre 4

## Sprint 3, 4

# Développement des fonctionnalités commerciales essentielles

## 1 Introduction

Ce chapitre présente la réalisation des Sprints 3 et 4 de notre plateforme Rewear, qui ajoutent des fonctionnalités clés pour l'interaction utilisateur et la gestion administrative. Le Sprint 3 inclut la gestion des favoris, un panier d'achat avec calcul des frais de livraison, et la validation des commandes. Le Sprint 4 ajoute un système de messagerie, un module de notation des articles, un tableau de bord administrateur et des notifications en temps réel. Ces améliorations rendent l'application plus fonctionnelle et professionnelle. Dans les sections suivantes, nous détaillerons les fonctionnalités développées au cours de ces deux sprints, en présentant les diagrammes de séquence et de classes associés à chaque fonctionnalité, pour mieux illustrer l'architecture et les interactions clés du système.

## 2 Sprint 3

### Gestion des favoris, du panier et des commandes

#### 2.1 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation du Sprint 3 illustré dans la figure 4.1 montre les interactions entre les différents acteurs (Utilisateur) et les fonctionnalités développées lors de ce sprint.

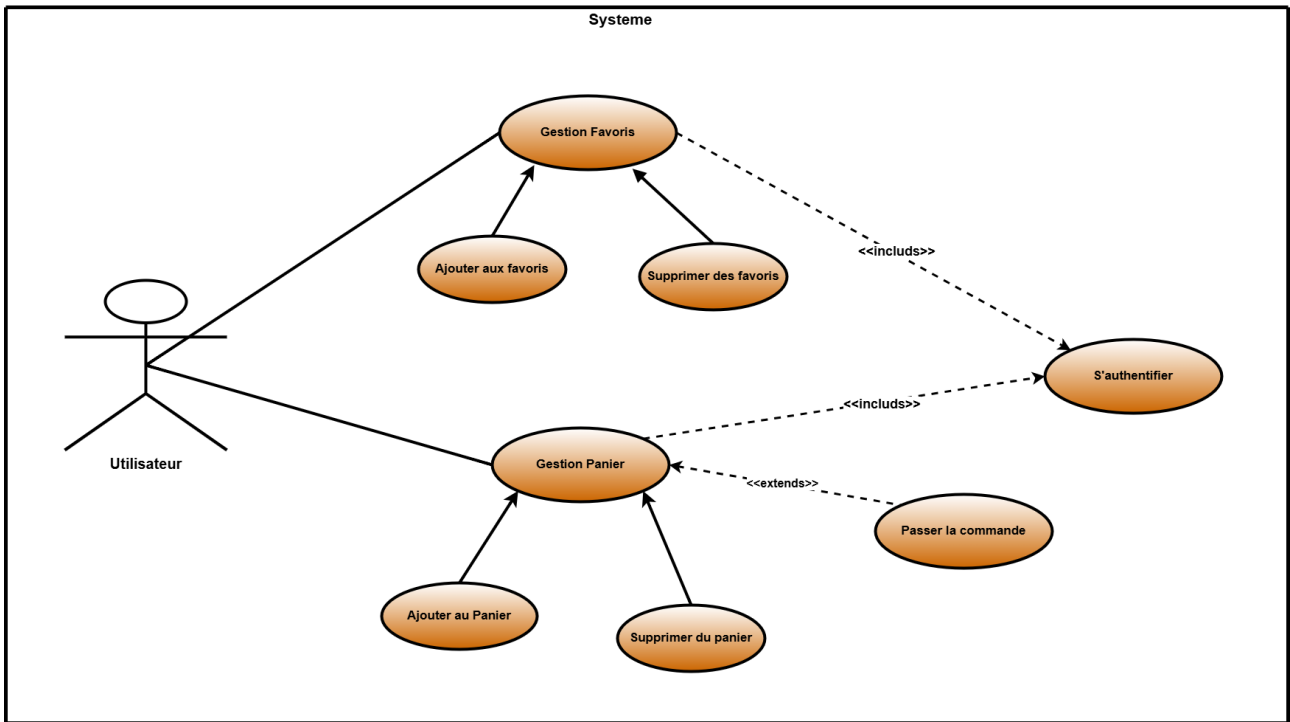


FIGURE 4.1 – Diagramme de cas d'utilisation pour **Sprint 3**

## 2.2 Product Backlog

Le tableau ci-dessous présente le backlog détaillé du Sprint 3, listant les fonctionnalités à développer, les tâches associées, les responsables, les priorités, l'estimation d'effort et le statut.

Fonctionnalité	Tâche	Responsable	Priorité	Estimation (heures)	Statut
Gestion des favoris	Développer la page favoris avec Intégration du bouton favoris dans l'interface du catalogue et page de détails du produits	Thinhinane	Haute	3h	À faire
	Développer la logique de vérification d'état des articles (dans/hors favoris)	Thinhinane	Haute	3h	À faire
	Implémenter la logique toggle : ajouter au premier clic, supprimer au second clic et	Lydia	Haute	2h	À faire

*Suite à la page suivante...*

Fonctionnalité	Tâche	Responsable	Priorité	Estimation (h)	Statut
	Créer le contrôleur de favoris (FavorisController) pour gérer les actions toggle	Lydia	Haute	3h	À faire
	Développer la table de favoris (FavorisTable) pour la persistance des données	Lydia	Haute	2h	À faire
Gestion du panier	Développer la page panier avec intégration du bouton panier dans l'interface du catalogue et page de détails du produits	Thinhinane	Haute	3h	À faire
	Développer la logique backend pour ajouter/retirer des articles au panier	Thinhinane	Haute	3h	À faire
	Implémenter l'affichage du panier avec regroupement par vendeur	Thinhinane	Haute	3h	À faire
	Développer la base de données pour la persistance du panier	Lydia	Haute	2h	À faire
Passer la commande	Créer la page avec le formulaire de commande ( informations personnelles de l'utilisateur, la méthode de paiement et le mode de livraison )	Lydia	Haute	5h	À faire
	Développer la logique de validation et confirmation de commande	Lydia	Haute	5h	À faire
<i>Suite à la page suivante...</i>					

Fonctionnalité	Tâche	Responsable	Priorité	Estimation (h)	Statut
	Calculer la distance vendeur-acheteur pour déterminer les frais de livraison	Thinhinane	Haute	5h	À faire
Paiement avec l'API Chargely	Configurer Chargely en récupérant les clés API en mode test et implémenter la connexion et les test de Chargely	Lydia	Haute	4h	À faire
	Améliorer le formulaire checkout en incluant le paiement avec Chargely et création des page de retour(succès et échec)	Thinhinane	Haute	7h	À faire
	Synchroniser les statuts de commandes avec les paiements	Lydia	Haute	3h	À faire

TABLE 4.1 – Product Backlog du sprint 3

## 2.3 Décomposition du sprint

La figure 4.2 illustre la répartition du sprint 3 sur deux semaines avec une allocation du temps dédiée à la conception, au développement et aux tests des différentes fonctionnalités listées précédemment.

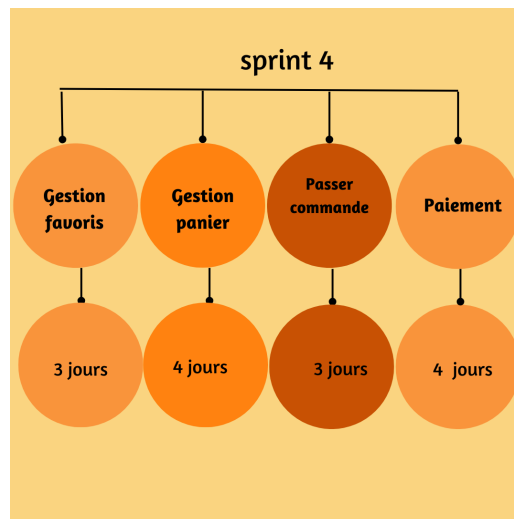


FIGURE 4.2 – Décomposition du Sprint 3

## 2.4 Présentation des diagrammes de séquence

Nous présentons ci-dessous la description détaillée des fonctionnalités majeures de ce sprint, accompagnée des diagrammes de séquence.

### 2.4.1 Fonctionnalité « Gestion des favoris »

#### Scénario principal (cas normal)

1. **Authentification** : L'utilisateur doit être authentifié pour accéder aux fonctionnalités de favoris.
2. **Sélection du bouton favoris** : L'utilisateur clique sur le bouton favoris d'un article.
3. **Vérification d'état** : Le système vérifie via le FavorisController si l'article est déjà dans les favoris de l'utilisateur.
4. **Action conditionnelle** :
  - **Si l'article est déjà dans les favoris** : Le système supprime l'article des favoris dans la base de données, le bouton change d'état (de sélectionné à non-sélectionné), et un message "Article retiré des favoris" s'affiche.
  - **Si l'article n'est pas encore dans les favoris** : Le système ajoute l'article aux favoris dans la base de données, le bouton change d'état (de non-sélectionné à sélectionné), et un message "Article ajouté aux favoris" s'affiche.
5. **Suppression de l'article des favoris** → L'utilisateur accède à la page des favoris, clique sur le bouton "Supprimer l'article des favoris". L'article est supprimé de la base de données, et un message "Article retiré des favoris" s'affiche.
6. **Mise à jour de l'interface** : L'interface utilisateur se met à jour en temps réel pour refléter le nouvel état.

#### Scénarios alternatifs (erreurs)

- **Échec d'authentification** → Si l'utilisateur n'est pas authentifié, il est redirigé vers la page de connexion avec le message "Connectez-vous pour gérer vos favoris".
- **Erreur lors de la suppression ou ajout des favoris** → Si une erreur se produit lors de la mise à jour des favoris dans la base de données, un message d'erreur s'affiche : "Une erreur est survenue, veuillez réessayer plus tard."

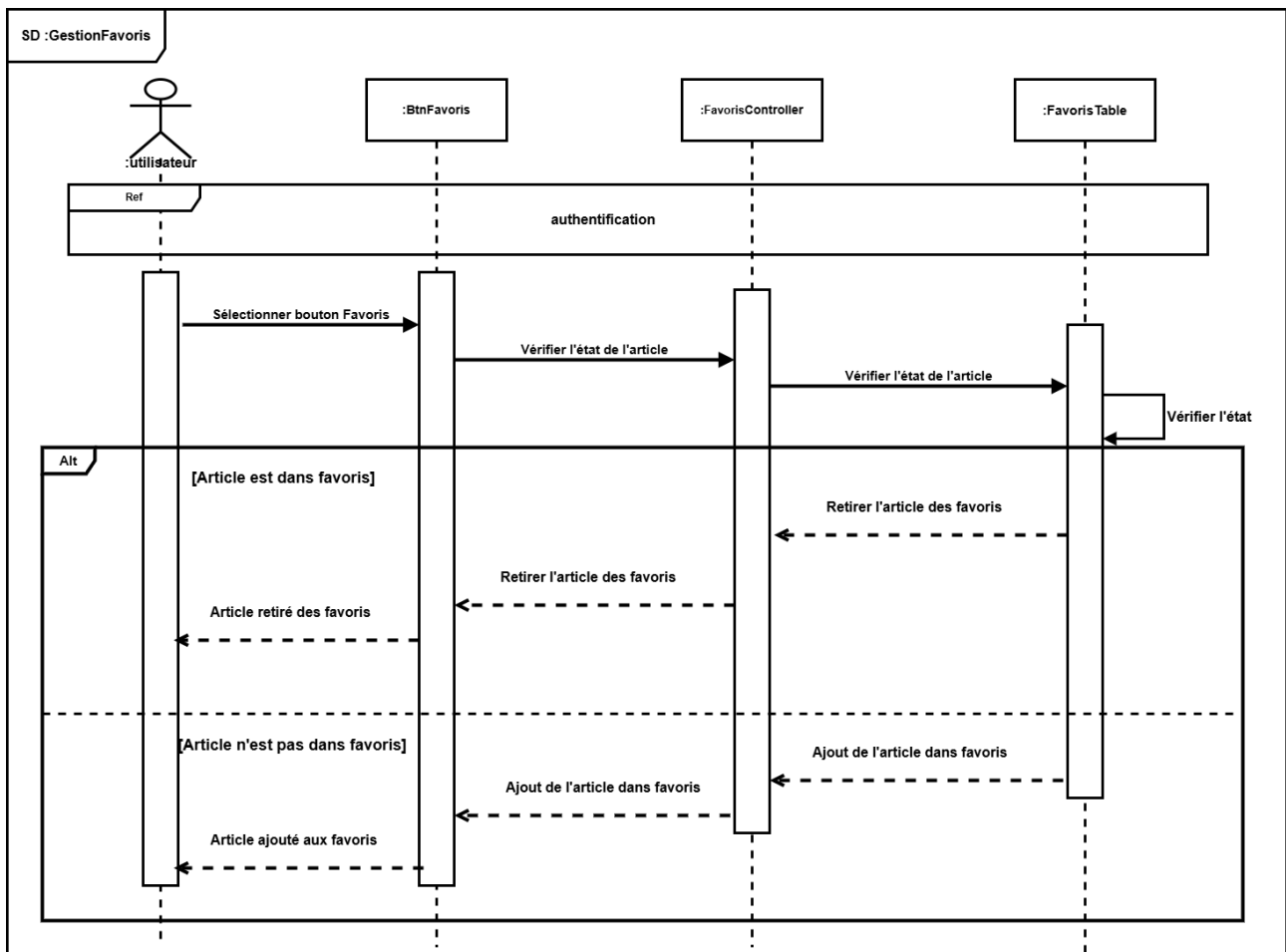


FIGURE 4.3 – Diagramme de séquence pour la fonctionnalité **gestion des favoris**

## 2.4.2 Fonctionnalité « Gestion du panier »

### Scénario principal (cas normal)

1. **Authentification** : L'utilisateur doit être authentifié pour accéder aux fonctionnalités de panier.
2. **Sélection du bouton panier** : L'utilisateur clique sur le bouton "panier" d'un article via l'interface GestionPanier.
3. **Vérification d'état** : Le système vérifie via le PanierController si l'article est déjà dans le panier en interrogeant la base de donnée.
4. **Action conditionnelle** :
  - **Si l'article est dans le panier** : Lorsque l'utilisateur clique sur le bouton d'ajout au panier, le système supprime l'article du panier, le bouton change d'état, et le message "Article retiré du panier" s'affiche.
  - **Si l'article n'est pas dans le panier** : Lorsque l'utilisateur clique sur le bouton d'ajout au panier, le système ajoute l'article au panier, le bouton change d'état, et le message "Article ajouté au panier" s'affiche.
  - **Suppression de l'article dans le panier** → L'utilisateur accède à la page panier, clique sur le bouton "Supprimer". L'article est supprimé de la base de données, et un message "Article retiré du panier" s'affiche.
  - **Mise à jour de l'interface** : L'interface utilisateur se met à jour en temps réel pour refléter le nouvel état.

5. **Logique de fonctionnement** : Le système fonctionne selon un principe de basculement (toggle) où le même bouton sert alternativement à ajouter ou retirer l'article selon son état actuel dans le panier. Le libellé du bouton et l'action exécutée dépendent de la présence ou non de l'article dans le panier au moment du clic.

### Scénarios alternatifs (erreurs)

- **Échec d'authentification**  
→ L'utilisateur non authentifié ne peut pas accéder aux fonctionnalités de panier
- **Erreur lors de la suppression ou ajout aux panier** → Si une erreur se produit lors de la mise à jour des favoris dans la base de données, un message d'erreur s'affiche : "Une erreur est survenue, veuillez réessayer plus tard."

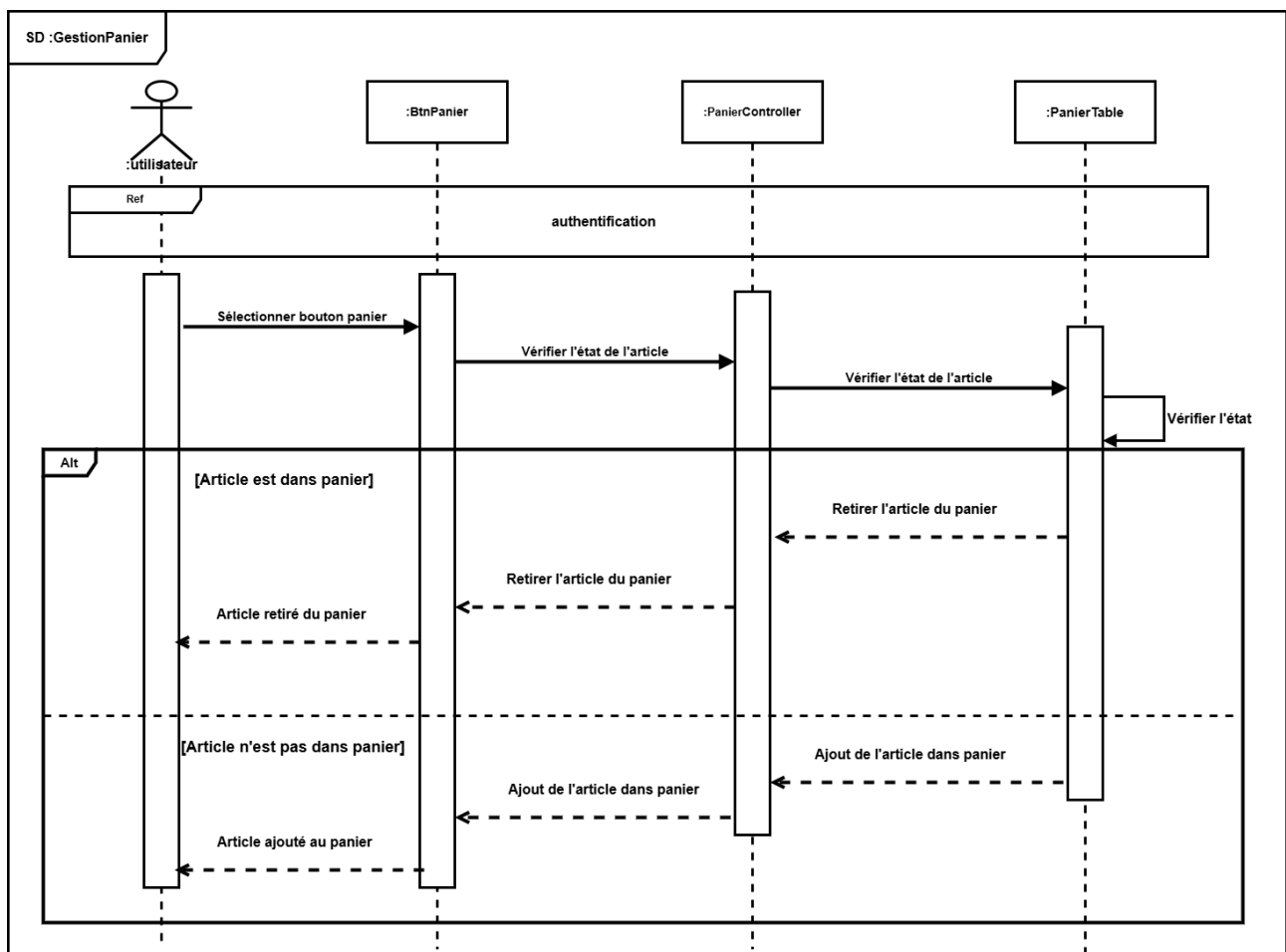


FIGURE 4.4 – Diagramme de séquence pour la fonctionnalité **gestion du panier**

### 2.4.3 Fonctionnalité « Gestion des commandes »

#### Scénario principal (cas normal)

1. **Authentification** : L'utilisateur doit être authentifié pour accéder au système de commande.
2. **Saisie de la commande** : L'utilisateur saisit sa commande via l'Interface Commande et remplit le formulaire d'achat (informations personnelles, informations sur le mode de paiement, et choisit le mode de livraison).
3. **Envoi du formulaire** : Le système transmet les informations de commande au backend.

4. **Vérification des informations** : le système vérifie les informations envoyées par l'utilisateur.
5. **Validation des informations** : Si les informations sont valides, le système procède au calcul frais de livraison et renvoie le montant total.
6. **Envoi du total** : Le système envoie le montant total à payer avec les frais de livraison à l'utilisateur.
7. **Confirmation de commande** : L'utilisateur confirme sa commande (« Passer la commande »).
8. **Enregistrement** : Une fois le paiement validé, la commande est enregistrée dans base de donnée avec le statut "en traitement".
9. **Confirmation finale** : Le système confirme l'enregistrement de la commande à l'utilisateur.

### Scénarios alternatifs (erreurs)

- **Informations invalides**  
→ Le système détecte des informations invalides ou champs manquants, demande des modifications et renvoie le formulaire.
- **Annulation de commande**  
→ L'utilisateur peut annuler sa commande à tout moment, le système annule la commande et l'utilisateur retourne à l'interface panier.
- **Échec du paiement**  
→ Redirection vers une nouvelle tentative de commande
- **Erreur d'enregistrement**  
→ Message d'erreur « Impossible d'enregistrer la commande » et demande de réessayer

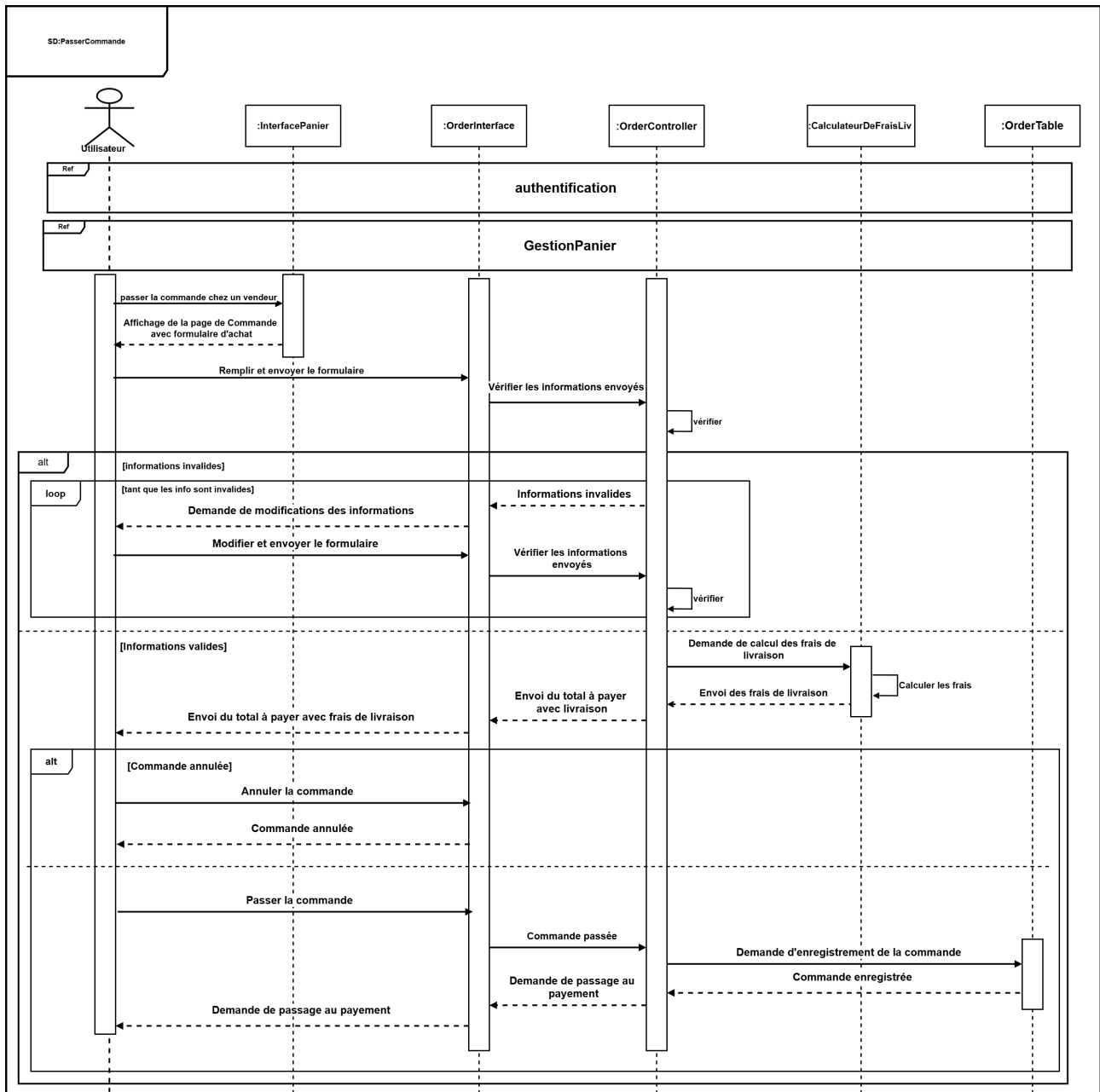


FIGURE 4.5 – Diagramme de séquence pour la fonctionnalité **passer commande**

## 2.4.4 Fonctionnalité « Paiement »

### Scénario principal (cas normal)

1. **Authentification** : L'utilisateur doit être authentifié pour accéder au processus de paiement.
2. **Initiation du paiement** : L'utilisateur finalise la commande en passant par la page de commande et procède au paiement.
3. **Vérification de la commande** : Le controller confirme que la commande est légitime et enregistrée et vérifie la méthode de paiement.
4. **Action conditionnelle** :
  - **Si paiement à la livraison** : Lorsque l'utilisateur choisit cette option, la commande est immédiatement enregistrée et finalisée.
  - **Si paiement par carte** : L'utilisateur passe au paiement via Chargily .

## 5. Traitement du paiement par carte

- Le controller demande au ChargilyService de créer un checkout Chargily.
- Le ChargilyService génère le checkout et retourne les détails de paiement.
- Le système demande la redirection des utilisateurs vers la page Chargily.
- L'utilisateur est dirigé vers l'espace de paiement sécurisé Chargily.

6. **Finalisation sur Chargily** : L'utilisateur saisit ses informations et valide le paiement sur la plateforme Chargily.

## 7. Traitement du retour de paiement :

- **Cas de succès** :Chargily redirige vers la page de succès avec confirmation de paiement.
- **Cas d'échec** : Chargily redirige vers la page d'erreur.

## 8. Gestion de l'échec (si le paiement a échoué) :

- L'utilisateur accède à la page d'erreur.
- Le système annule la commande .
- La base de données supprime la commande.

## 9. Gestion du succès (si paiement validé) :

- L'utilisateur accède à la page de succès.
- Le système finalise le paiement.
- Demande d'enregistrement de la commande avec statut payé dans la base de données.
- Envoi d'une notification au vendeur pour l'informer de la vente.

## Scénarios alternatifs (erreurs)

- **Échec d'authentification**  
→ Redirection vers la page de connexion .
- **Erreur de connexion Chargily**  
→ Erreur de connexion demande de réessayer.
- **Annulation volontaire de l'utilisateur**  
→ Commande annulée.

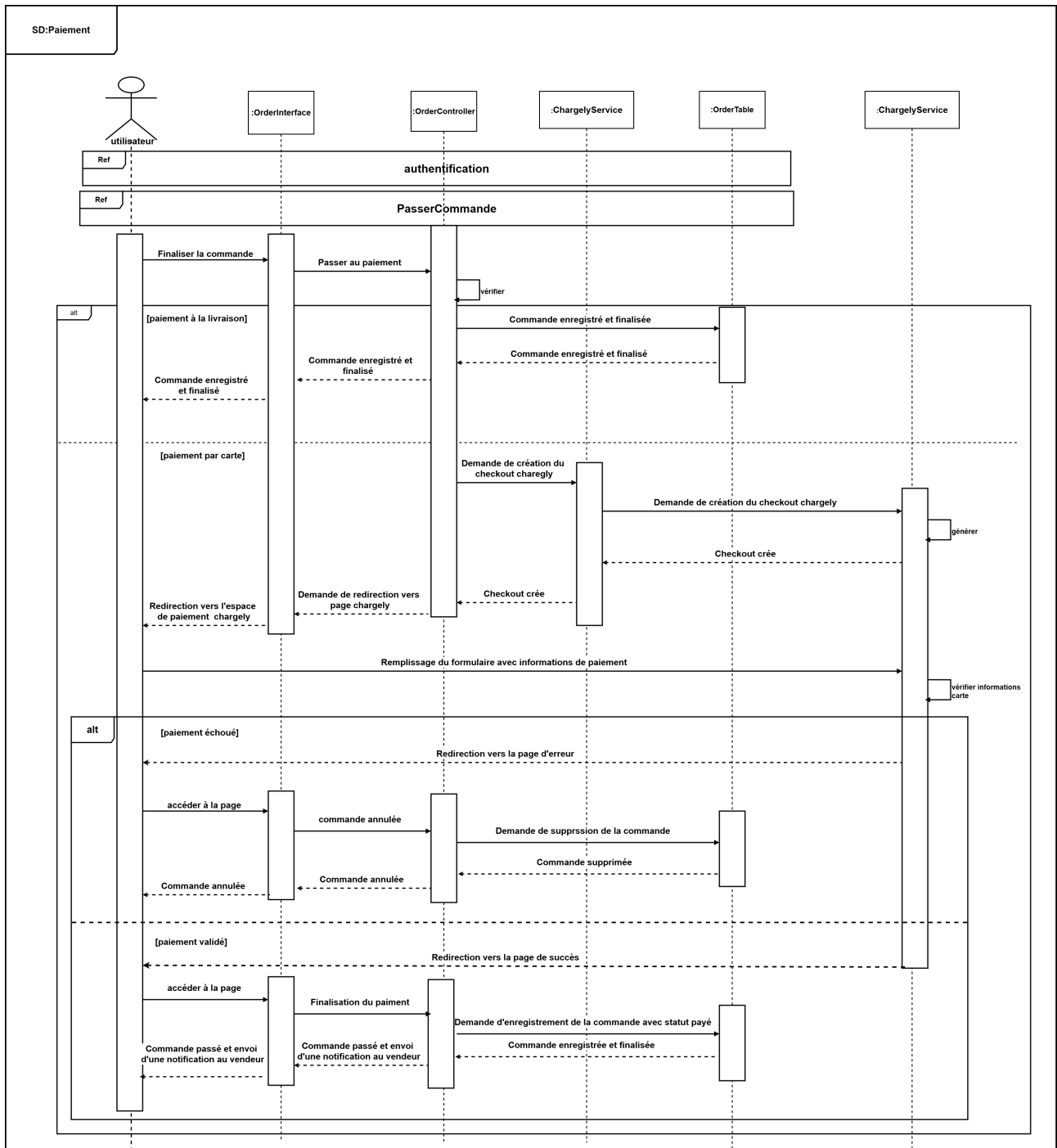


FIGURE 4.6 – Diagramme de séquence pour la fonctionnalité **paiement avec Chargily**

## 2.5 Diagramme de classes

Le diagramme de classes présenté dans la figure 4.7 offre une vue d'ensemble des classes constituant les éléments du Sprint 3, notamment les classes relatives aux articles, favoris, panier et commandes.

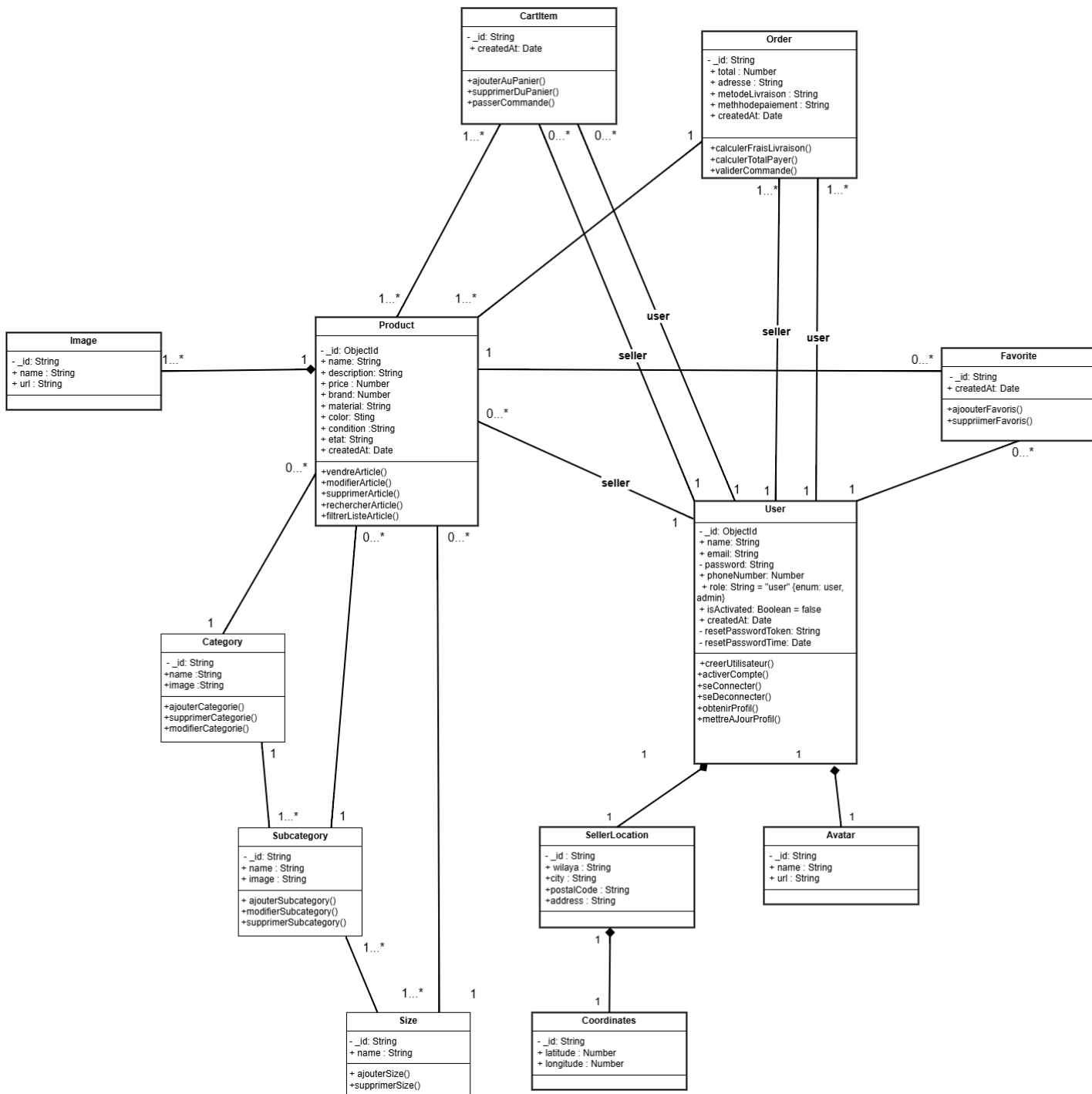


FIGURE 4.7 – Diagramme de classes **Sprint 3**

## 2.6 Vue des Interfaces Utilisateur

Dans cette section, nous présentons les interfaces graphiques essentielles développées lors du Sprint 3.

### 2.6.1 Interface de gestion des favoris

La figure 4.8 montre les interfaces permettant à l'utilisateur de visualiser et gérer ses articles favoris avec un système de bouton 'toggle' intuitif.

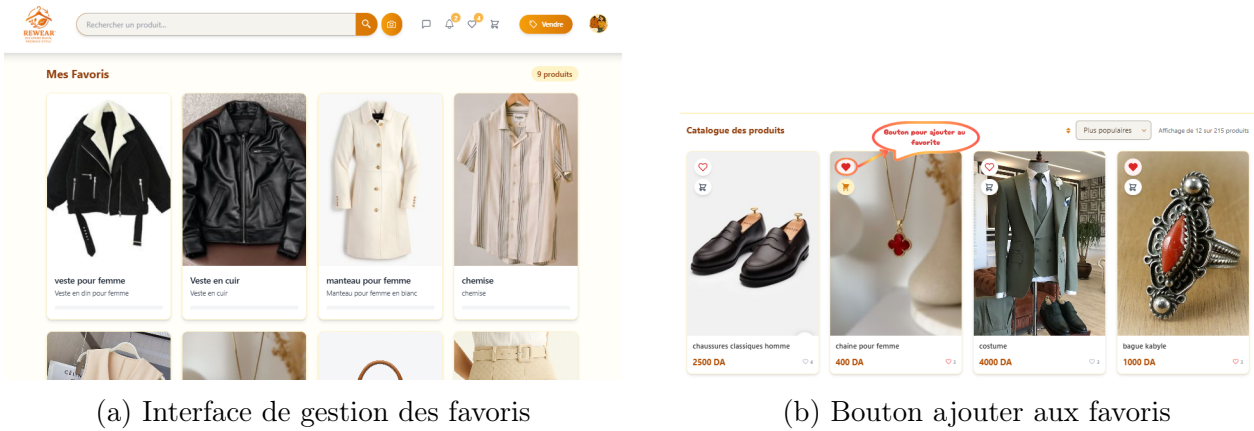


FIGURE 4.8 – Interfaces des favoris

### 2.6.2 Interface de gestion du panier

La figure 4.9 montre les interfaces qui permettent à l'utilisateur de visualiser les articles ajoutés à son panier, supprimer des articles, et passer une commande chez un vendeur précis.

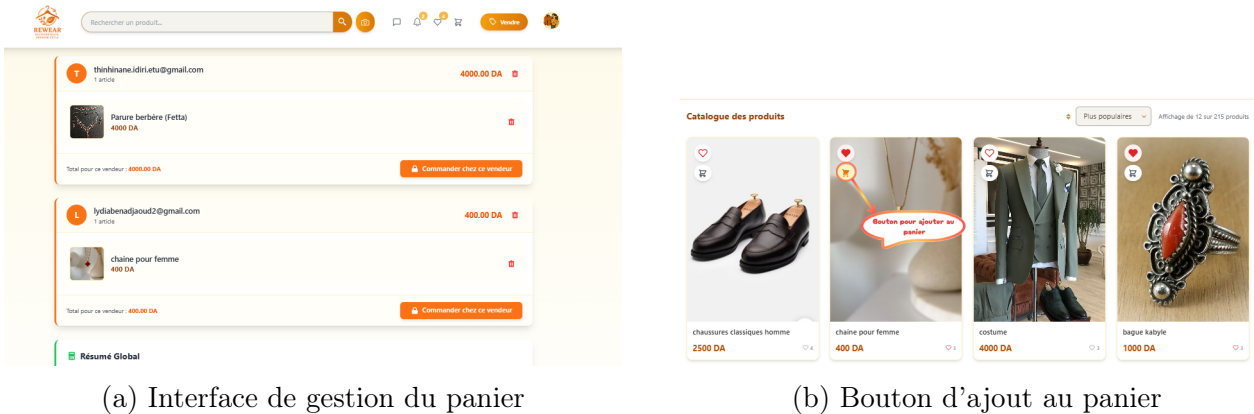


FIGURE 4.9 – Interfaces du panier

### 2.6.3 Interface de passage de la commande

L'interface illustrée dans la figure 4.10 présente le récapitulatif de la commande et permet à l'utilisateur de finaliser son achat et de passer au paiement.

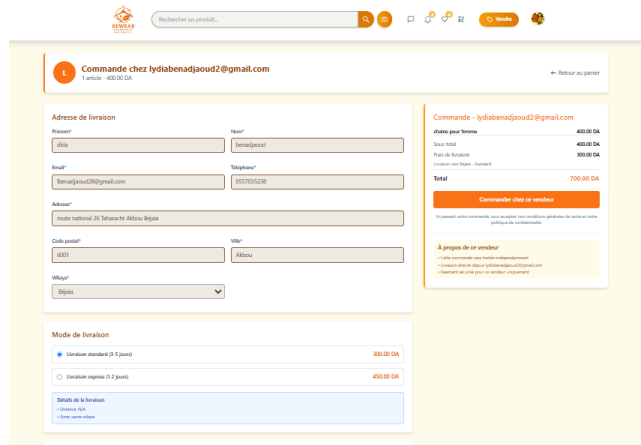
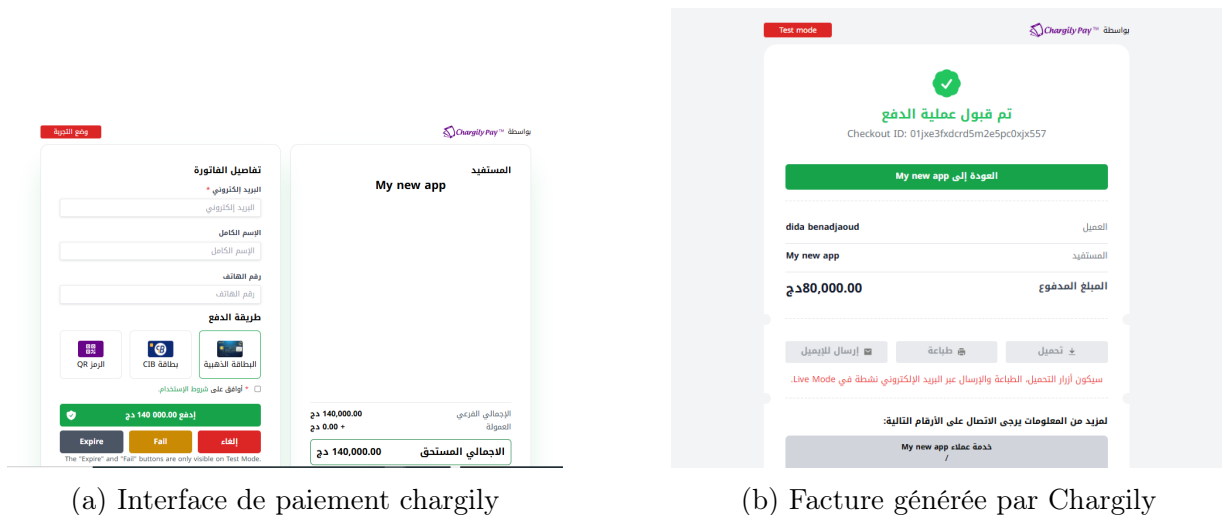


FIGURE 4.10 – Interface de passage de commande

### 2.6.4 Interface de paiement via Chargily

L'interface illustré dans la figure 4.11 présente l'interface de paiement de l'API Chargily en mode test.



(a) Interface de paiement chargily

(b) Facture générée par Chargily

FIGURE 4.11 – Interfaces de l'API Chargily

## 3 Sprint 4

### Gestion de la messagerie, évaluation, notifications et du tableau de bord admin

#### 3.1 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation du Sprint 4 illustré dans la figure 4.12 montre les interactions entre les différents acteurs (Utilisateur, Administrateur) et les fonctionnalités développées lors de ce sprint.

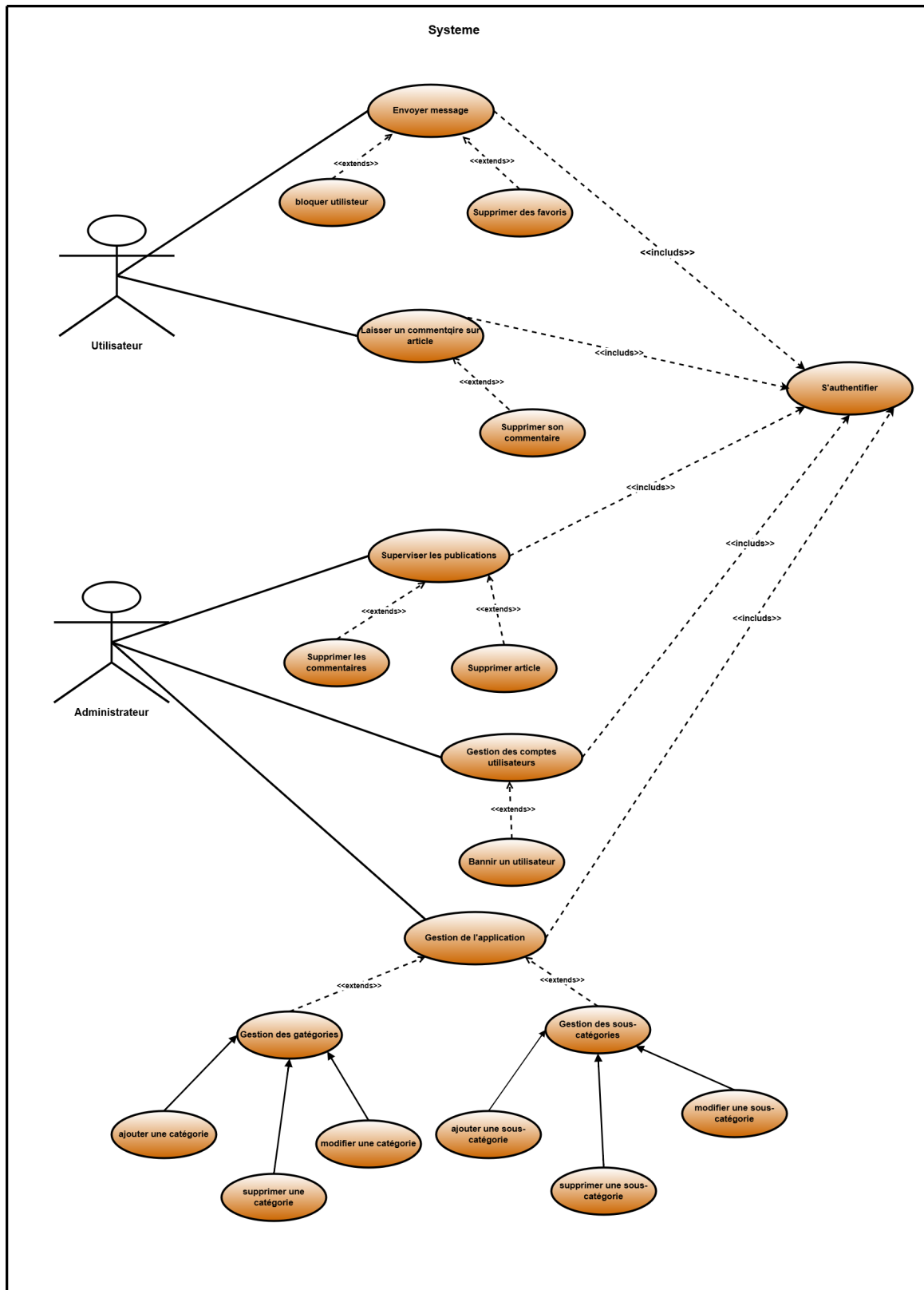


FIGURE 4.12 – Diagramme de cas d'utilisation pour **Sprint 4**

### 3.2 Product Backlog

Le tableau ci-dessous présente le backlog détaillé du Sprint 4, mis à jour selon les diagrammes de séquence, listant les fonctionnalités à développer, les tâches associées, les responsables, priorités, l'estimation d'effort et le statut.

<b>Fonctionnalité</b>	<b>Tâche</b>	<b>Responsable</b>	<b>Priorité</b>	<b>Estimation (heures)</b>	<b>Statut</b>
Messagerie	Créer la page de messagerie (conversations/messages).	Thinhinane	Haute	3h	À faire
	Implémenter la logique backend pour l'envoi et la réception des messages, incluant la gestion des erreurs et la confirmation de réception	Lydia	Haute	12h	À faire
	Mettre en place la recherche d'utilisateurs par nom dans la barre de recherche, en effectuant une requête en base de données	Thinhinane	Haute	5h	À faire
	Enregistrer les conversations et les messages dans la base de données	Lydia	Haute	3h	À faire
	développer et Ajouter la vérification du statut utilisateur (bloqué/non bloqué) lors de l'envoi de messages, avec une gestion des restrictions si l'utilisateur est bloqué	Thinhinane	Haute	12h	À faire
	Implémenter les fonctionnalités de suppression de message et de suppression de conversation, avec une confirmation avant suppression pour éviter toute suppression accidentelle	Lydia	Haute	7h	À faire
<i>Suite à la page suivante...</i>					

<b>Fonctionnalité</b>	<b>Tâche</b>	<b>Responsable</b>	<b>Priorité</b>	<b>Estimation (h)</b>	<b>Statut</b>
Évaluation des articles	Développer le composant commentaires et le système d'étoiles interactif (1 à 5)	Lydia	Haute	8h	À faire
	Intégrer le composant dans la page de détails produit	Lydia	Haute	4h	À faire
	Créer la structure de base de données	Thinhinane	Haute	3h	À faire
	Développer la logique backend pour les commentaires et le système d'étoiles	Thinhinane	Haute	7h	À faire
Tableau de Bord	Créer l'interface de la page administrateur	Lydia	Critique	6h	À faire
	Créer les composants d'affichage des statistiques (nombre d'utilisateurs, articles, commandes avec chiffre d'affaires)	Thinhinane	Critique	12h	À faire
	Créer le menu latéral (la navigation entre les différentes pages) et gestion des autorisations administrateur	Lydia	Haute	8h	À faire
	Récupération et affichage de la liste des utilisateurs existants avec actions de suppression	Lydia	Haute	6h	À faire
	Ajout de la recherche et filtres pour les articles (prix, évaluation)	Thinhinane	Moyenne	5h	À faire
<i>Suite à la page suivante...</i>					

<b>Fonctionnalité</b>	<b>Tâche</b>	<b>Responsable</b>	<b>Priorité</b>	<b>Estimation (h)</b>	<b>Statut</b>
	Récupération et affichage des articles existants avec actions (suppression/détails)	Thinhinane	Haute	7h	À faire
	Récupération et affichage des commandes existantes avec leurs statuts	Lydia	Haute	6h	À faire
	Ajout du tri par (date, montant, article) et recherche pour les commandes par statut	Lydia	Moyenne	4h	À faire
	Récupération et affichage des catégories existantes avec actions (ajout, modification et suppression)	Thinhinane	Moyenne	5h	À faire
	Récupération et affichage des tailles existantes avec actions (ajout et suppression)	Lydia	Basse	3h	À faire
	Récupération et affichage des sous-catégories existantes avec actions (ajout, modification et suppression)	Thinhinane	Basse	4h	À faire
Notifications	Créer l'interface notifications	Thinhinane	Critique	6h	À faire
	Développer l'enregistrement des notifications en base de données	Thinhinane	Haute	5h	À faire
<i>Suite à la page suivante...</i>					

Fonctionnalité	Tâche	Responsable	Priorité	Estimation (h)	Statut
	Implémenter un mécanisme de notifications en temps réel pour alerter les propriétaires d'articles lors d'actions tierces	Lydia	Haute	5h	À faire

TABLE 4.2 – Product Backlog du sprint 4

### 3.3 Décomposition du sprint

Le Sprint 4 se déroule sur une période de deux semaines, avec une planification détaillée des tâches à réaliser en tenant compte des interactions complexes révélées par les diagrammes de séquence, et une attention particulière portée aux tests de validation des scénarios alternatifs et des cas d'erreur. Cette décomposition est illustrée dans la figure 4.13

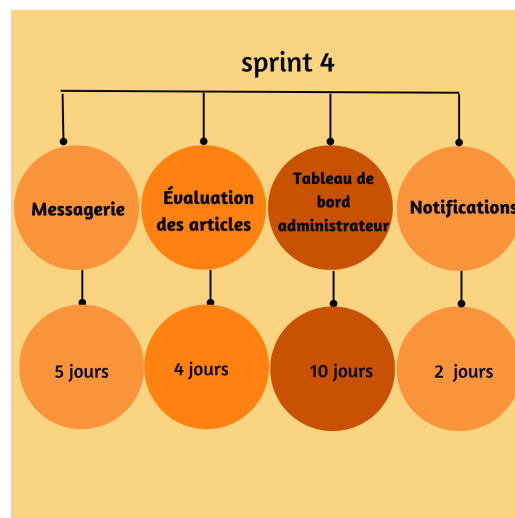


FIGURE 4.13 – Décomposition du Sprint 4

### 3.4 Présentation des diagrammes de séquence

Nous présentons ci-dessous la description détaillée des fonctionnalités majeures de ce sprint, accompagnée des diagrammes de séquence illustrant le déroulement des interactions système-utilisateur.

#### 3.4.1 Fonctionnalité « Messagerie »

##### Scénarios principale (cas normal)

1. L'utilisateur doit être connecté pour accéder à la messagerie.

2. Une fois connecté, il peut débiter une conversation de deux manières possibles :
  - **Recherche d'un utilisateur**  
→ L'utilisateur peut rechercher une personne par son nom via la barre de recherche, puis la sélectionner pour démarrer une conversation.
  - **Depuis la page de produit**  
→ L'utilisateur peut cliquer sur le bouton "contacter le vendeur" sur la page de détail d'un produit pour être redirigé vers la messagerie et démarrer une conversation avec le vendeur.
3. Le système vérifie si une conversation existe déjà entre les utilisateurs :
  - **Si aucune conversation existante** : Le système crée une nouvelle conversation.
  - **Si une conversation existe** : Le système ouvre la conversation existante.
4. L'utilisateur rédige et envoie son message.
5. Le système vérifie le statut de l'utilisateur destinataire (bloqué ou non). Si l'utilisateur n'est pas bloqué, le message est envoyé et enregistré.
6. L'utilisateur peut supprimer un message déjà envoyé ou supprimer toute la conversation .
7. L'utilisateur peut bloquer une personne.

### Scénarios alternatifs (erreurs)

- **Échec d'authentification**  
→ Si l'utilisateur n'est pas authentifié, il est redirigé vers la page de connexion et ne peut pas accéder à la messagerie.
- **Utilisateur bloqué**  
→ Si l'utilisateur destinataire est bloqué, le message ne peut pas être envoyé et une erreur est affichée.
- **Utilisateur inexistant**  
→ Si la recherche ne trouve aucun utilisateur, un message d'erreur est affiché.

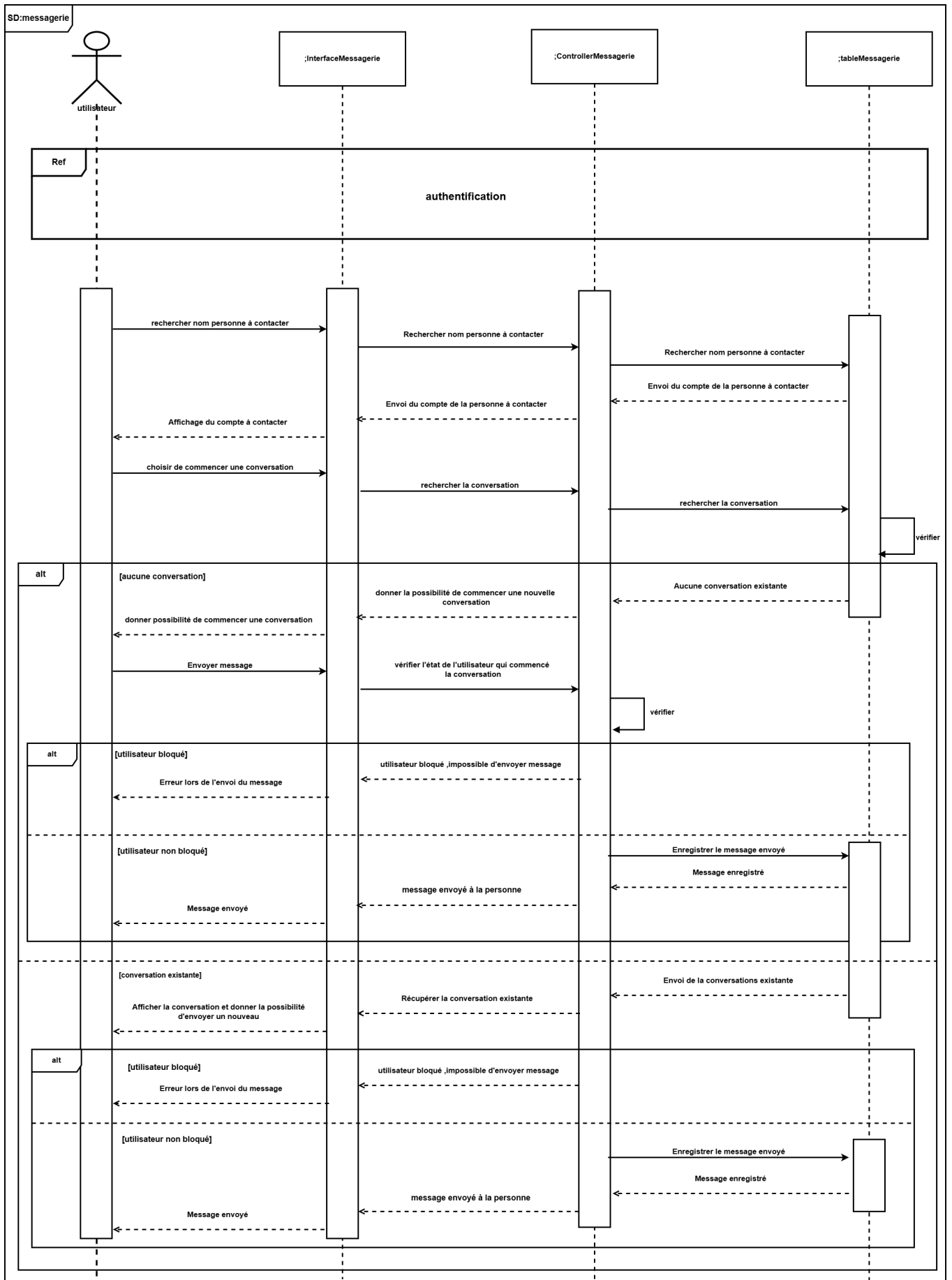


FIGURE 4.14 – Diagramme de séquence pour la fonctionnalité **Envoyer un message**

### 3.4.2 Fonctionnalité « Évaluation des articles »

#### Scénario principal (cas normal)

1. L'utilisateur est connecté au système.
2. L'utilisateur sélectionne un article depuis HomePage.
3. Le système affiche la page de détails de l'article avec ses caractéristiques.
4. Le système affiche les avis existants des autres utilisateurs.
5. L'utilisateur peut rédiger et publier un avis avec commentaire et notation par étoiles.
6. Le système enregistre le commentaire la note.
7. Le système affiche la liste mise à jour des avis incluant les nouveaux commentaires.

#### Scénario alternatif (cas erreur)

- **Échec d'authentification**

→ Si l'utilisateur n'est pas authentifié, il est redirigé vers la page de connexion et ne peut pas accéder à la messagerie.

- **Suppression de commentaire**

→ L'utilisateur sélectionne un avis qu'il souhaite supprimer

→ Le système vérifie s'il est propriétaire de l'avis

→ si il est propriétaire alors il donne la possibilité de le supprimer.

→ si il n'est pas propriétaire impossible de supprimer

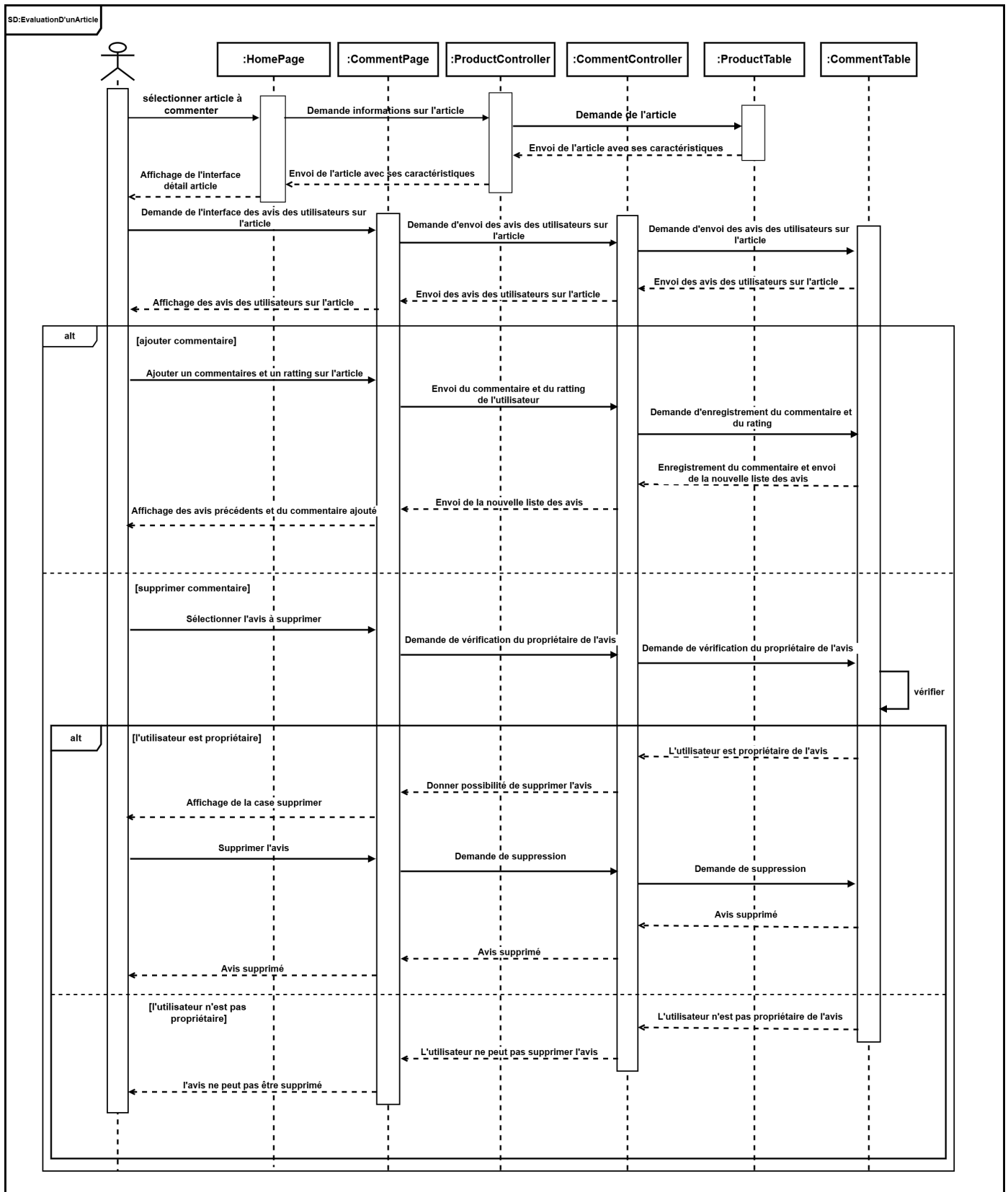


FIGURE 4.15 – Diagramme de séquence pour fonctionnalité laisser un avis sur un article

### 3.4.3 Fonctionnalité « Tableau de bord pour l'administrateur »

En raison de la diversité des tâches liées à la fonctionnalité "admin", nous présentons un exemple détaillé de la gestion des sous-catégories pour illustrer le fonctionnement du système.

#### Fonctionnalité d'ajout des sous-catégories

### Scénario principale (cas normal)

1. L'administrateur accède à l'interface de gestion des sous-catégories.
2. Le système charge et affiche la liste des sous-catégories existantes.
3. L'administrateur remplit le formulaire d'ajout de nouvelle sous-catégorie(nom,catégorie,sous-catégorie,taille,matière,état,description,images).
4. Il soumet le formulaire au système.
5. Le système effectue une vérification de validation du formulaire.
6. Si les champs sont valides, la sous-catégorie est enregistrée.
7. La liste actualisée des sous-catégories est affichée avec la nouvelle entrée.

### Scénario alternatif (cas erreur)

- **Échec d'authentification**

→ Si l'utilisateur n'est pas authentifié, il est redirigé vers la page de connexion et ne peut pas accéder à la messagerie.

- **Validation des champs échouée**

→ Si des champs sont vides ou contiennent des formats invalides, Le système affiche une erreur spécifique

→ L'administrateur corrige les informations dans le formulaire, Il soumet à nouveau jusqu'à validation réussie

- **Validation de l'image**

→ Si l'image ne répond pas aux critères de l'application, le système demande à l'utilisateur de la modifier (taille, format, etc.) .

→ L'administrateur modifie l'image et soumet à nouveau jusqu'à validation réussie .

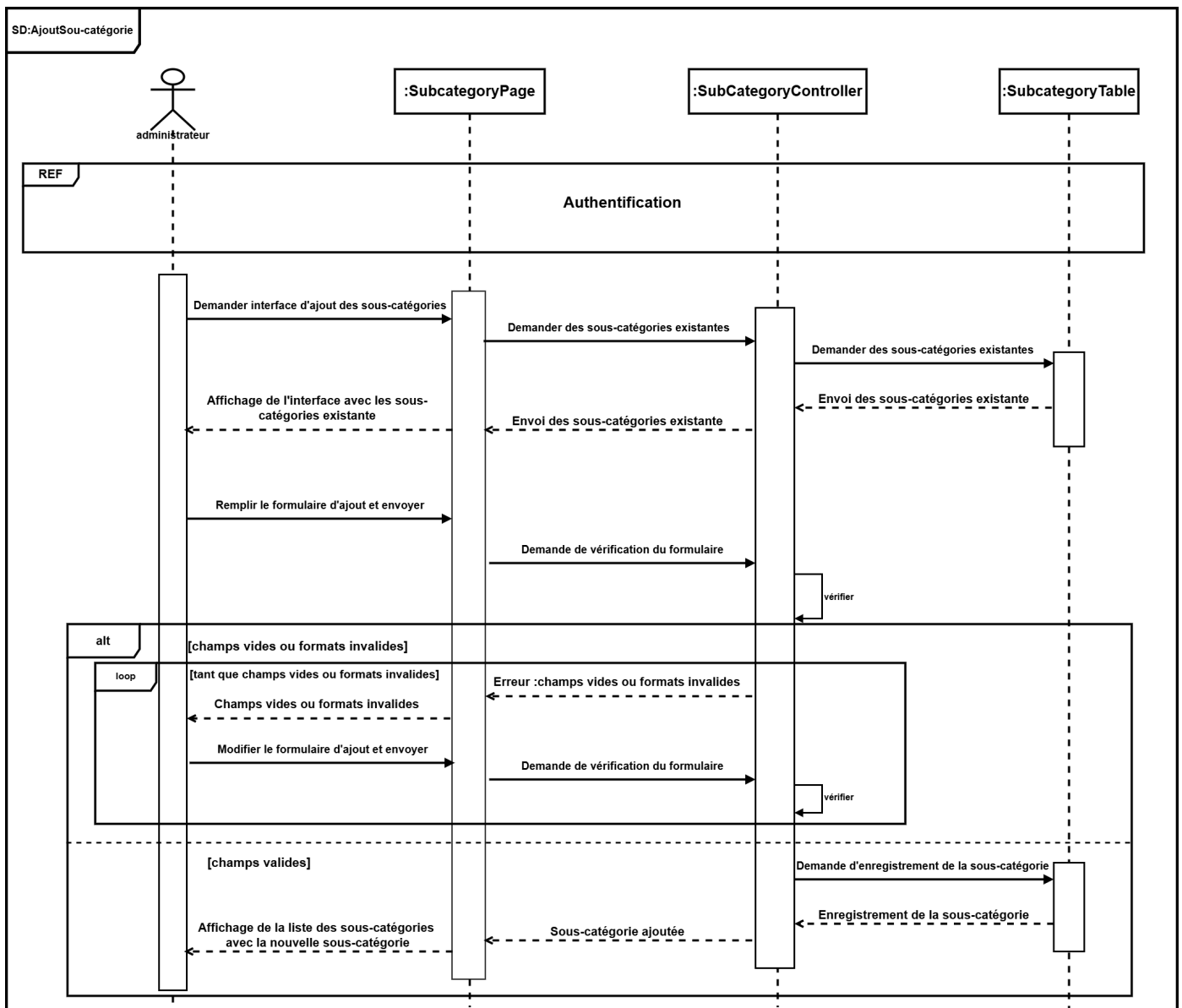


FIGURE 4.16 – Diagramme de séquence pour fonctionnalité **ajouter une sous-catégorie**

### Fonctionnalité de suppression ou modification d'une sous-catégorie

#### Scénario principale (cas normal)

1. L'administrateur accède à l'interface de gestion des sous-catégories.
2. Le système charge et affiche la liste des sous-catégories existantes.
3. L'administrateur choisit une action : consulter, modifier ou supprimer une sous-catégorie.
4. Pour la modification : le système affiche le formulaire pré-rempli, l'administrateur modifie les informations et soumet le formulaire.
5. Pour la suppression : l'administrateur confirme la suppression.
6. Le système effectue une vérification des informations.
7. Si les données sont valides, l'opération est enregistrée.
8. La liste actualisée des sous-catégories est affichée.

## Scénario alternatif (cas erreur)

- **Échec d'authentification**

→ Si l'utilisateur n'est pas authentifié, il est redirigé vers la page de connexion et ne peut pas accéder à la messagerie.

- **Validation de modification échouée**

→ Si des champs sont vides ou contiennent des formats invalides, Le système affiche une erreur spécifique

→ L'administrateur corrige les informations dans le formulaire, Il soumet à nouveau jusqu'à validation réussie

- **Validation de l'image**

→ Si l'image ne répond pas aux critères de l'application, le système demande à l'utilisateur de la modifier (taille, format, etc.) .

→ L'administrateur modifie l'image et soumet à nouveau jusqu'à validation réussie .

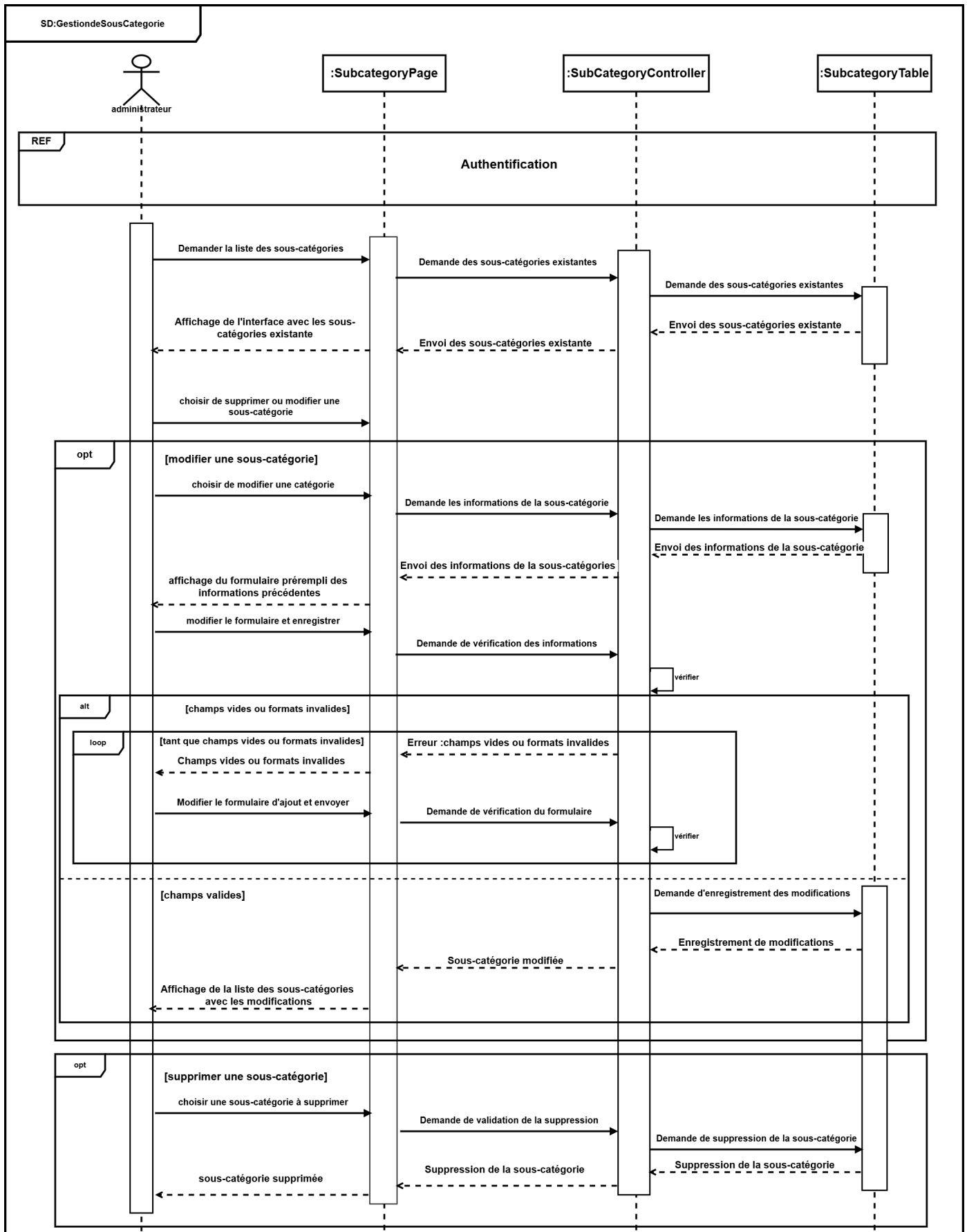


FIGURE 4.17 – Diagramme de séquence pour fonctionnalité **modifier ou supprimer une sous-catégorie**

### 3.4.4 Fonctionnalité « Notifications »

#### Scénario principal (cas normal)

1. L'utilisateur est connecté au système.
2. Un utilisateur émetteur ajoute un commentaire sur un article.
3. Le système détecte automatiquement cette action .
4. Une demande d'envoi de notification est générée.
5. Le système enregistre la notification dans la base de données.
6. Pour tous les utilisateurs connectés concernés, le système envoie les notifications stockées.
7. Les notifications sont affichées dans l'interface de chaque utilisateur récepteur.
8. Les utilisateurs peuvent consulter leurs notifications en temps réel.

#### Scénarios alternatifs

- **Utilisateur déconnecté**  
→ Les notifications sont stockées et seront envoyées lors de la prochaine connexion
- **Notification lue**  
→ Une fois consultée, la notification peut être marquée comme lue
- **Suppression de notification**  
→ L'utilisateur peut supprimer une notification de sa liste

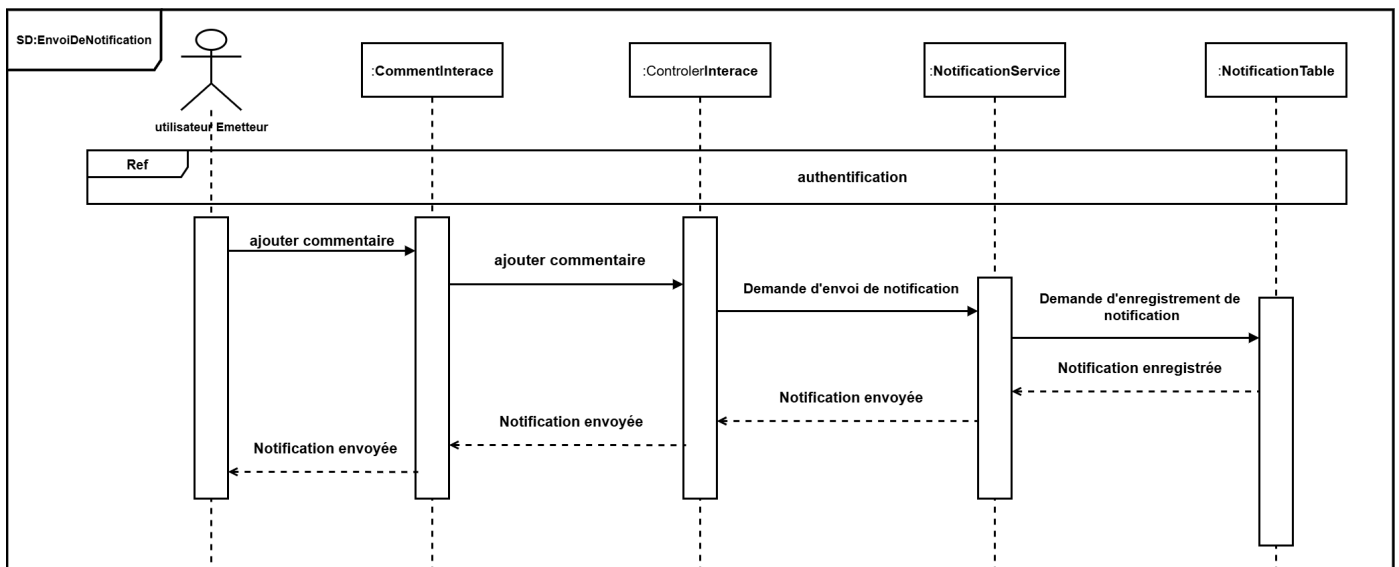


FIGURE 4.18 – Diagramme de séquence pour fonctionnalité **Envoi des notifications**

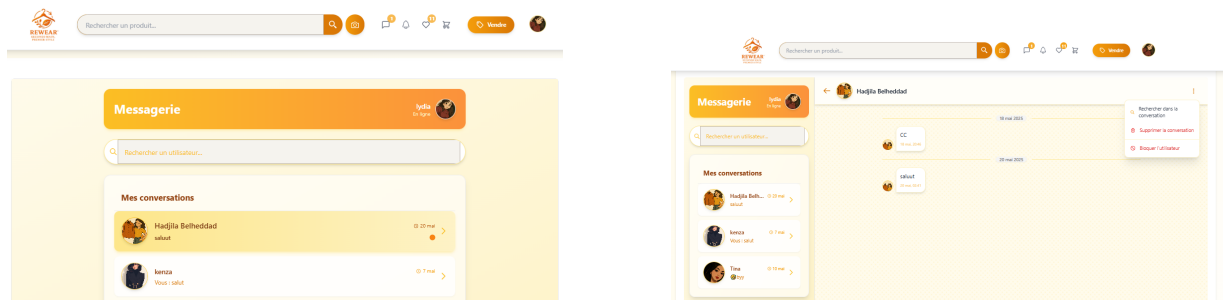


### 3.6 Vue des Interfaces Utilisateur

Dans cette section, nous présentons les interfaces graphiques essentielles développées lors du Sprint 4.

#### 3.6.1 Interface de messagerie

La figure 4.20 montre les fonctionnalités principales de la messagerie (envoyer et recevoir des messages privés avec d'autres utilisateurs).



(a) Interface de la liste des conversations

(b) Interface de la conversation

FIGURE 4.20 – Messagerie

#### 3.6.2 Interface d'évaluation des articles

L'utilisateur peut évaluer les articles et laisser des commentaires. Ceci est illustré dans la figure 4.21.

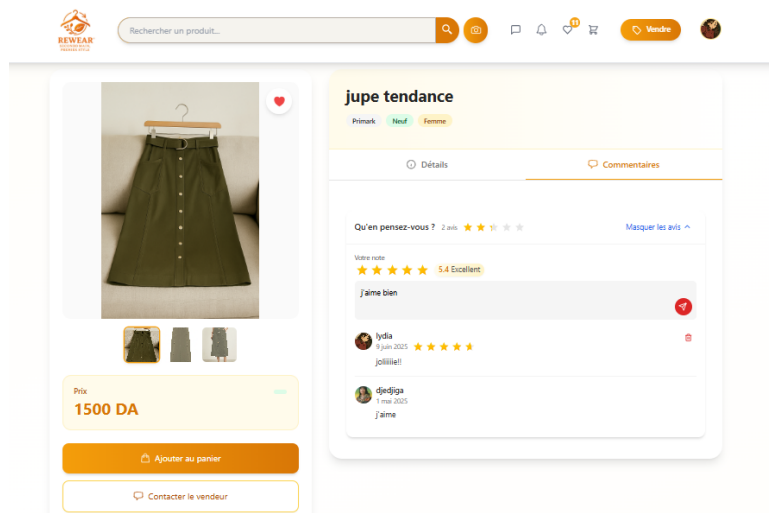


FIGURE 4.21 – Interface d'évaluation des articles

### 3.6.3 Interface du tableau de bord administrateur

L'administrateur peut gérer les utilisateurs et les articles depuis un tableau de bord dédié. Ceci est illustré dans la figure 4.22



FIGURE 4.22 – Interfaces des fonctionnalités de l'administrateur

### 3.6.4 Interface de notifications

Les utilisateurs sont informés des actions importantes via des notifications. Ceci est illustré dans la figure 4.23

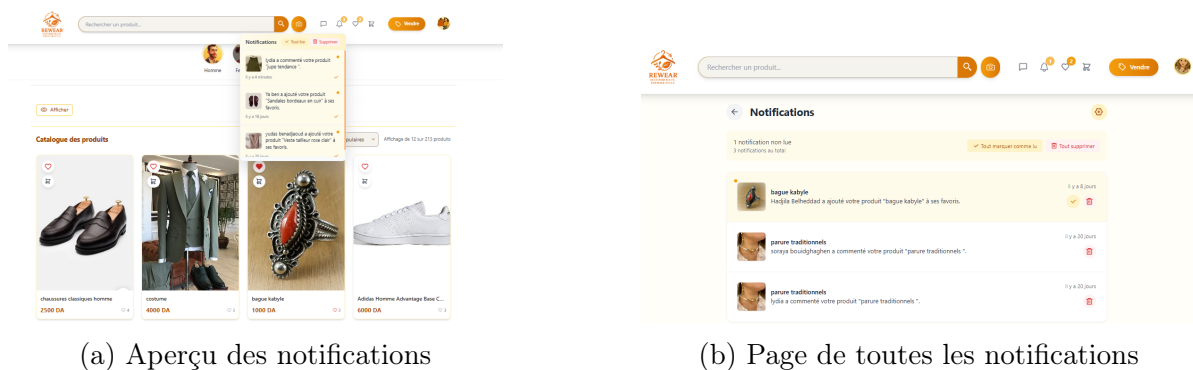


FIGURE 4.23 – Notifications

## 4 Conclusion

En résumé, ce chapitre présente les Sprints 3 et 4, qui ont permis de transformer notre plateforme en un écosystème e-commerce complet. Le Sprint 3 a posé des bases solides pour les transactions avec des fonctionnalités telles que la gestion des favoris, le panier et la validation des commandes. Le Sprint 4 a enrichi l'expérience utilisateur avec l'ajout d'un système de messagerie, d'un module d'évaluation des articles, d'un tableau de bord administrateur et de notifications en temps réel. Ces améliorations ont permis de créer une plateforme fluide pour les utilisateurs et dotée d'outils efficaces pour les administrateurs. Le chapitre suivant présentera le Sprint 5, qui introduira de nouvelles fonctionnalités pour renforcer encore le système.

# Chapitre 5

## Sprint 5

# Implémentation d'un système de recommandation avec l'API Recombee

## 1 Introduction

Avec l'ajout de l'API Recombee, un système de recommandation intelligent basé sur l'IA, ce cinquième sprint marque une étape significative dans le développement de notre plateforme e-commerce. L'objectif est d'utiliser les données comportementales des utilisateurs pour générer des recommandations personnalisées, ce qui améliorera l'expérience utilisateur et augmentera l'engagement sur la plateforme.

### 1.1 Présentation de l'API de recommandation Recombee

#### 1.1.1 Définition de l'API Recombee :

Recombee est une plateforme Software as a Service (SaaS) conçue pour la recommandation de produits et de contenu. Alimentée par l'intelligence artificielle, elle permet aux utilisateurs de recevoir des recommandations personnalisées en fonction de leurs interactions passées avec l'application.[32]

#### 1.1.2 Principe de fonctionnement de Recombee

Recombee s'appuie sur une variété d'algorithmes avancés pour générer des recommandations personnalisées.

- **Filtrage collaboratif** : Son principe repose sur l'idée que les utilisateurs ayant des comportements passés similaires auront probablement des préférences similaires à l'avenir.[33, 32]
- **Filtrage basé sur le contenu** : Son principe est de se concentrer sur les caractéristiques des produits (taille, couleur, matière) pour suggérer des articles similaires à ceux que l'utilisateur a déjà appréciés.[33, 32]
- **Apprentissage profond** : Recombee utilise des techniques avancées d'apprentissage profond pour affiner ses recommandations tel que les réseaux de neurones profonds qui sont utilisés pour prédire les actions futures des utilisateurs en analysant leur comportement passé et en capturant des motifs complexes.[33, 32]
- **Modèles Spécialisés** : Recombee s'appuie sur des algorithmes éprouvés dans de nombreux projets concrets, utilisant des modèles de diversification qui évitent la monotonie

des suggestions, des modèles de popularité qui suivent les tendances du moment, et des modèles de périodicité qui repèrent les habitudes répétitives des utilisateurs, garantissant ainsi des recommandations parfaitement adaptées à chaque situation.[32]

itemid	name	price	role	seller	size	sizes	subCategory
323485453000304173250880	maillie	NULL	NULL	NULL	NULL	NULL	NULL
6710876465030279938286f	Femme	NULL	NULL	NULL	NULL	NULL	NULL
6714878465030279938286f	Enfant	NULL	NULL	NULL	NULL	NULL	NULL
67104493298-84963390ac3	Jupe	NULL	NULL	NULL	NULL	[671b8e01e50b302f9938299f, 671c8e01e50b302f9938289... more	NULL
671495182938-84963390ac3	Short	NULL	NULL	NULL	NULL	[671b8e01e50b302f9938299f, 671c8e01e50b302f99382897... more	NULL
6714954-2938-84963390ac3	Parablon	NULL	NULL	NULL	NULL	[671b8e01e50b302f9938299f, 671c8e01e50b302f99382897... more	NULL
671495879398-84963390ac3	Combinaison	NULL	NULL	NULL	NULL	[671b8e01e50b302f9938299f, 671c8e01e50b302f99382897... more	NULL
671497292938-84963390ac3	Chausure	NULL	NULL	NULL	NULL	[671b8e01e50b302f9938299f, 671c8e01e50b302f9938289... more	NULL
67149729398-84963390ac3	Sac	NULL	NULL	NULL	NULL		NULL

FIGURE 5.1 – Interface de l’API Recombee

## 1.2 Diagramme de cas d’utilisation

Les interactions entre les différents acteurs (utilisateurs) et les fonctionnalités créées lors de ce sprint sont illustrées dans le diagramme de cas d’utilisation représenté dans la figure 5.2

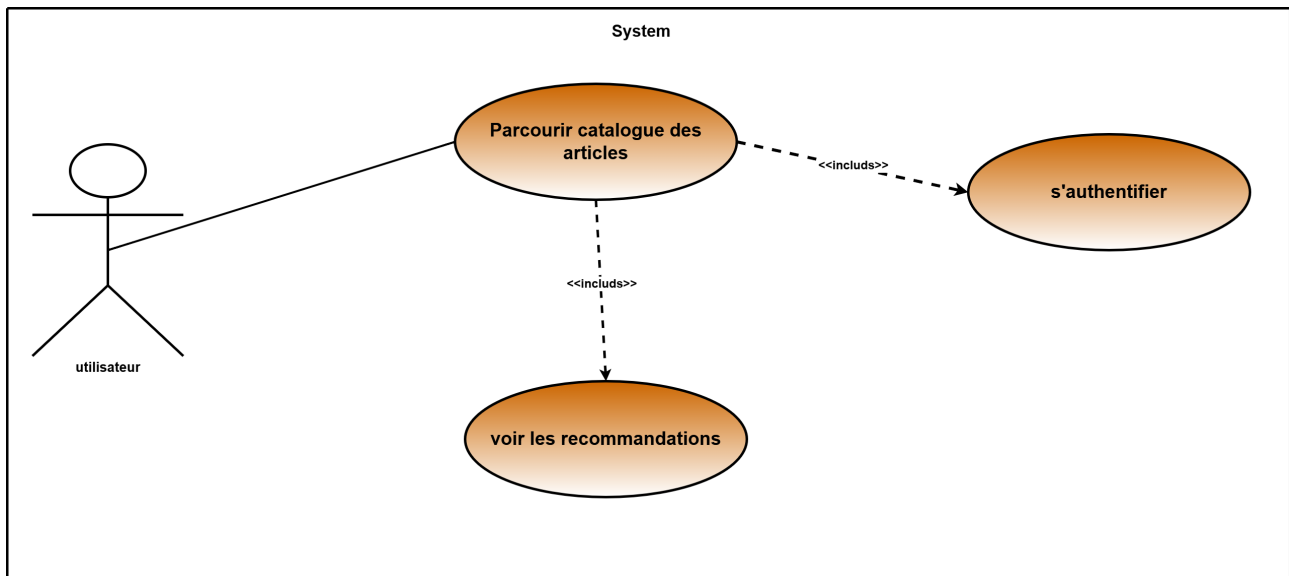


FIGURE 5.2 – Diagramme de cas d’utilisation pour Sprint 5

## 1.3 Product Backlog

Le tableau ci-dessous présente le backlog du Sprint 5 en détail, incluant les tâches et les fonctionnalités à développer.

Fonctionnalité	Tâche	Responsable	Priorité	Estimation (h)	Statut
Implémentation du système de recommandation via Recombee	Créer un compte Recombee et configurer la base de données initiale	Thinhinane	Haute	10h	À faire
	Intégrer le SDK Recombee dans l'architecture existante	Lydia	Haute	20h	À faire
	Développer l'API de synchronisation pour envoyer les données produits à Recombee	Lydia	Haute	24h	À faire
	Implémenter le tracking des interactions utilisateurs (favoris, achats, ajouts panier)	Thinhinane	Haute	20h	À faire
	Développer le composant d'affichage des recommandations (frontend)	Lydia	Haute	7h	À faire

TABLE 5.1 – Product Backlog du sprint 5

## 1.4 Décomposition du sprint

Ce sprint est réparti sur deux semaines avec une allocation du temps dédiée à la conception, au développement de la fonctionnalité implémentation du système de recommandation.

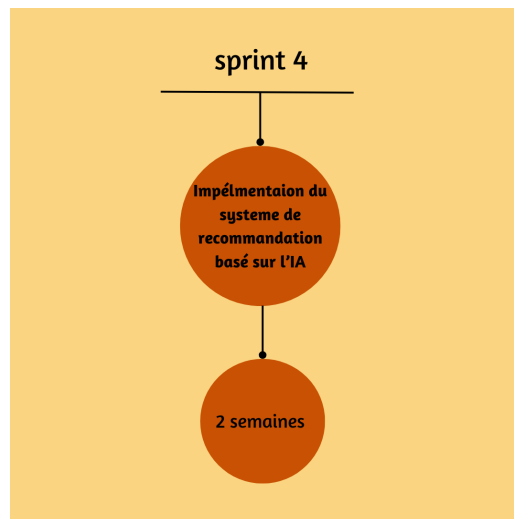


FIGURE 5.3 – Décomposition du sprint 5

## 1.5 Présentation des diagrammes de séquence

Nous présentons ci-dessous la description détaillée de la fonctionnalité Implémentation du système de recommandation via l'API IA Recombee, accompagnée du diagramme de séquence illustrant le déroulement des interactions système-utilisateur.

### 1.5.1 Fonctionnalité « voir recommandations »

#### Scénario principal(cas normal)

1. **Authentification** : L'utilisateur doit être authentifié pour voir les recommandations.
2. **Gestion des favoris** : L'utilisateur doit ajouter des articles aux favoris pour avoir des recommandations personnalisées .
3. **Gestion du panier** : L'utilisateur doit ajouter des articles aux panier pour avoir des recommandations personnalisées .
4. **Passer la commande** : L'utilisateur doit acheter des articles pour avoir des recommandations personnalisées .
5. **L'envoi des données des interactions de Backend Node.js vers l'API Python/FastAPI** : La base de données envoie les données des articles et des utilisateurs avec leurs interactions vers l'API Python/FastAPI.
6. **Synchronisation des données entre l'API Python/FastAPI et Recombee** : l'API Python/FastAPI envoie toutes les données reçues depuis backend Node.js vers Recombee.
7. **Vérification de la connexion Recombee** : Le système vérifie l'état de la connexion avec RecombeeAPI.
8. **Action conditionnelle** :
  - **Si Recombee est connecté et l'utilisateur a des interactions** : : Recombee génère et envoie des recommandations basées sur les favoris et les achats des utilisateurs vers l'API Python/FastAPI
  - **Si Recombee est connecté et l'utilisateur n'a pas encore d'interactions** : Recombee génère et envoie des recommandations selon la popularité des articles vers l'API Python/FastAPI .

9. **Stockage des recommandations dans la base de donnée** : l'API Python/FastAPI envoie les recommandations vers la base de données pour les stocker .
10. **Affichage des recommandations** : la base de donnée envoie les recommandations de l'utilisateur connecté vers le système pour les afficher dans HomePage .

### Scénarios alternatifs (erreurs)

- **Échec d'authentification**  
→ Redirection vers la page de connexion .
- **Erreur de synchronisation**  
→ Message d'erreur "Erreur de synchronisation" et retry automatique du processus de synchronisation des données .

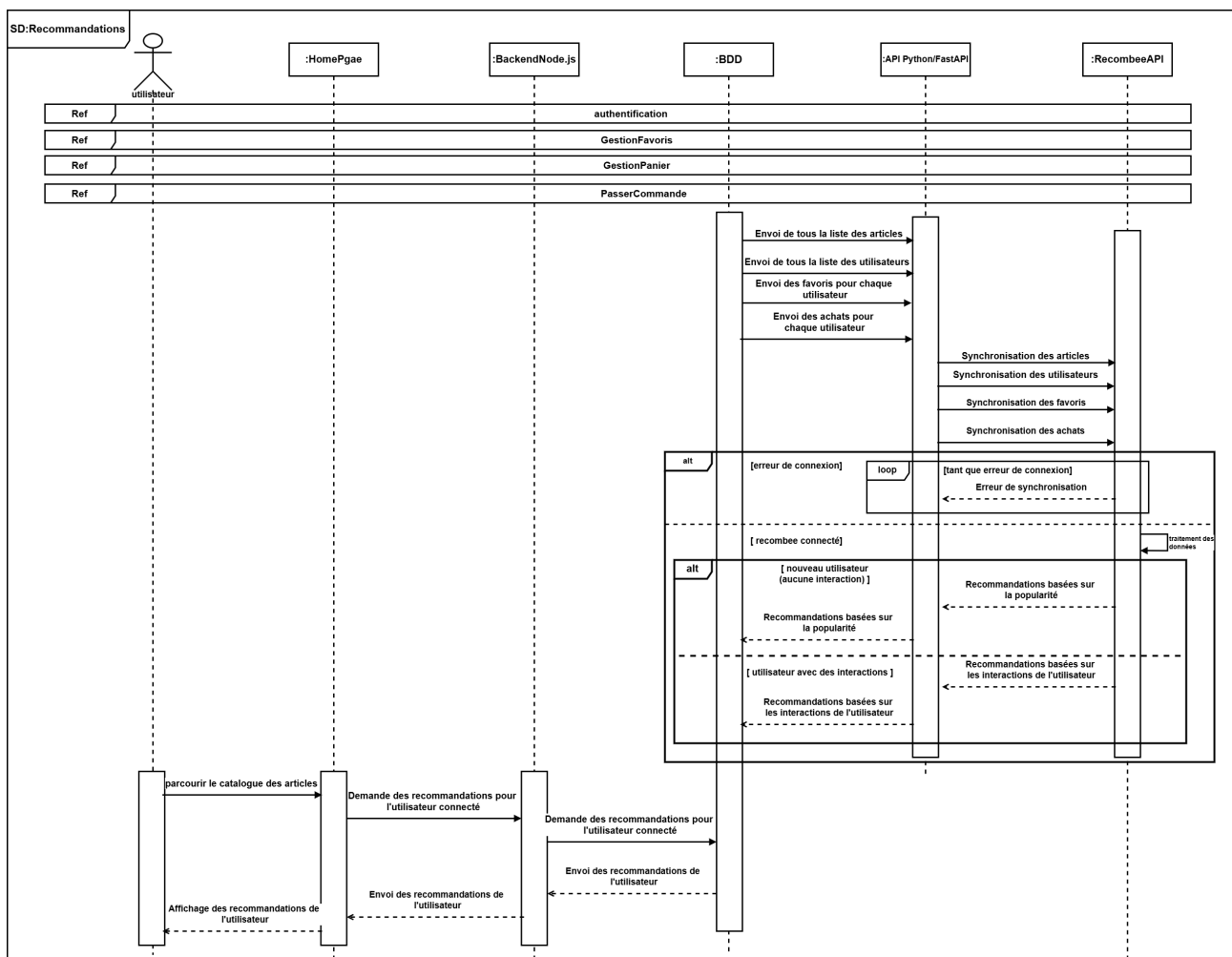


FIGURE 5.4 – Diagramme de séquence pour la fonctionnalité **Recommandation** avec l'API **Recomee**

## 2 Diagramme de classes

Le diagramme de classes présenté dans la figure 5.5 offre une vue d'ensemble des classes constituant les éléments du Sprint 5, notamment la classe relative aux recommandations. Ce diagramme constitue la représentation globale de l'architecture du système, intégrant toutes les entités et leurs associations dans une vision unifiée du projet.



### 3.0.1 Interface d'affichage des recommandations

L'interface illustré dans la figure 5.6 présente la section où les utilisateurs peuvent découvrir leurs recommandations personnalisées.

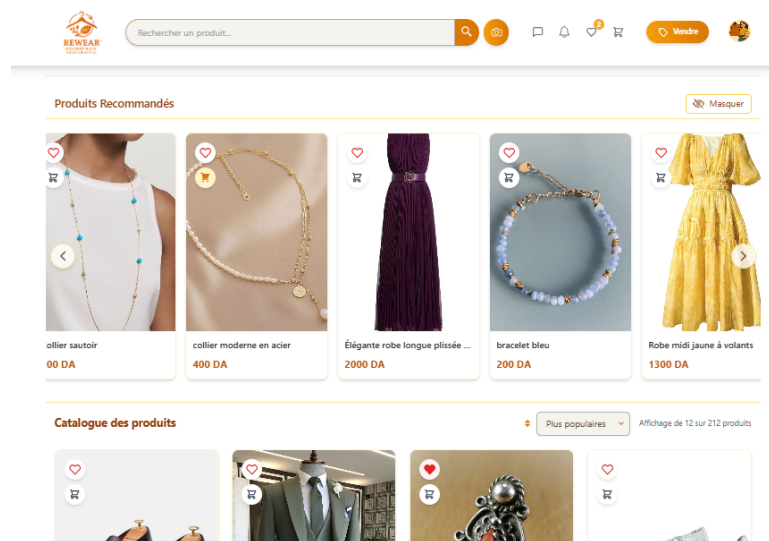


FIGURE 5.6 – Interface des recommandations

## 4 Conclusion

Avec l'intégration réussie du système de recommandation Recombee, le Sprint 5 a marqué une étape importante et offre désormais une expérience utilisateur personnalisée et intelligente. L'architecture qui combine Python/FastAPI, Recombee et Node.js garantit une synchronisation efficace et des recommandations en temps réel. Cette fonctionnalité transforme notre plateforme en un environnement d'achat intelligent, nous donnant un avantage concurrentiel significatif et les fondations pour une relation client à long terme.

# Chapitre 6

## Sprint 6 Intégration d'un Chatbot Intelligent avec Rasa

### 1 Introduction

Ce sprint se concentre sur l'intégration d'un chatbot intelligent au sein de notre plateforme Rewear, en utilisant le framework Rasa . Ce chapitre détaille la conception et le développement de cette solution conversationnelle, qui permet aux utilisateurs d'interagir de manière fluide et naturelle avec la plateforme. Nous y présenterons les outils et technologies utilisés, les diagrammes de séquence illustrant les différents scénarios d'interaction, ainsi que les interfaces utilisateur développées pour offrir une expérience à la fois intuitive et engageante.

### 2 Présentation de Rasa

#### 2.1 Définition et Architecture

Rasa est un framework open-source développé en Python, spécialement conçu pour créer des assistants conversationnels et des chatbots intelligents [34]. Il se distingue par sa capacité à comprendre le langage naturel et à maintenir des conversations contextuelles complexes [35].

##### **Architecture de Rasa :**

Rasa repose sur deux composants principaux qui travaillent en synergie [35] :

**Rasa NLU (Natural Language Understanding) :** Ce module est responsable de l'analyse et de la compréhension du langage naturel. Il effectue plusieurs tâches essentielles :

- **Classification d'intentions (Intent Classification) :** Détermine l'intention de l'utilisateur à partir de son message
- **Extraction d'entités (Entity Extraction) :** Identifie et extrait les informations spécifiques du message
- **Tokenisation :** Décompose le texte en unités plus petites pour l'analyse

**Rasa Core :** Ce composant gère le flux de conversation et détermine les actions à entreprendre. Il utilise des algorithmes d'apprentissage automatique pour :

- Prédire la prochaine action à effectuer
- Maintenir le contexte de la conversation
- Gérer les transitions entre différents états de dialogue

## 2.2 Concepts Fondamentaux

**Intents (Intentions)** : Classification de l'objectif utilisateur. Chaque message est analysé pour déterminer ce que souhaite accomplir l'utilisateur [36].

- **Exemple** : Si un utilisateur écrit "Quel est le prix de ce produit ?", l'intention peut être classifiée comme `prix_produit`.

**Entities (Entités)** : Éléments d'information spécifiques extraits du message (noms, valeurs, dates). Permettent de comprendre *sur quoi* agir en complément de l'intention.

- **Exemple correct** : Dans la phrase "Quel est le prix de la robe rouge ?", l'entité `robe_rouge` serait extraite en tant que produit, et pourrait être utilisée pour rechercher un article spécifique.
- **Exemple incorrect** : Dans la phrase "Combien coûte la robe rouge à pois ?", l'entité `robe_rouge_à_pois` pourrait être mal extraite comme une seule entité, ce qui rendrait la recherche plus complexe si l'article n'est pas précisément enregistré sous ce nom complet.

**Actions** : Réponses du chatbot aux messages des utilisateurs. Il existe deux types :

- **Actions prédéfinies (réponses statiques)** : Des réponses fixes qui ne changent pas selon les situations.
- **Actions personnalisées (opérations complexes, requêtes base de données)** : Ces actions sont dynamiques et exécutent des tâches comme interroger une base de données ou appeler une API.

**Exemple** : Une action personnalisée pourrait être une requête pour afficher les produits disponibles en fonction des critères donnés par l'utilisateur.

**Stories** : Scénarios d'interaction utilisés pour entraîner le modèle de dialogue et définir les flux conversationnels [36].

- **Exemple** : Une story pourrait être la séquence suivante :  
Utilisateur : "Je cherche des chemises."  
Chatbot : "Quel type de chemises cherchez-vous ?"  
Utilisateur : "Une chemise blanche."  
Chatbot : "Voici les chemises blanches disponibles."

**Rules** : Règles définissant des actions spécifiques en fonction de certaines conditions. Contrairement aux stories, les rules sont plus rigides et permettent de spécifier des actions immédiates dans des cas particuliers.

- **Exemple** : Si l'utilisateur écrit "je veux me déconnecter", une règle pourrait être définie pour exécuter immédiatement l'action `logout` sans nécessiter d'interaction supplémentaire [36].

## 2.3 Avantages de Rasa pour notre Projet

**Contrôle des données** : Hébergement local contrairement aux solutions cloud, crucial pour les données sensibles e-commerce [37]. **Personnalisation** : Framework adaptable aux spécificités du domaine métier [35]. **Intégration** : Compatible avec notre stack MERN via APIs REST [34]. **Évolutivité** : Apprentissage continu basé sur les interactions réelles. **Économique** : Solution open-source sans frais par requête [37].

## 2.4 Diagramme de cas d'utilisation

Les interactions entre les différents acteurs (utilisateurs) et les fonctionnalités développées au cours de ce sprint sont représentées dans le diagramme de cas d'utilisation présenté dans la

figure 6.1.

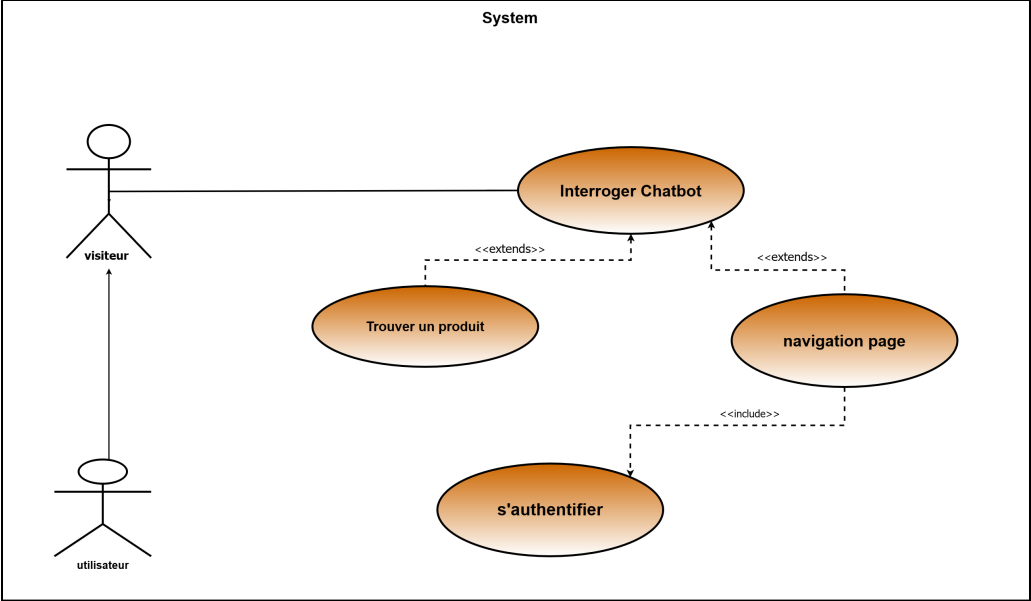


FIGURE 6.1 – Diagramme de cas d'utilisation pour **Sprint 6**

## 2.5 Product Backlog

Le tableau ci-dessous présente le backlog détaillé du projet chatbot Rasa, listant les fonctionnalités développées, les tâches associées, les responsables, les priorités, l'estimation d'effort et le statut.

Fonctionnalité	Tâche	Responsable	Priorité	Estimation (heures)	Statut
Mise en place d'un chatbot intelligent.	Créer le composant chatbot avec interface utilisateur	Lydia	Haute	6h	À faire
	Créer et configurer les intents, stories, domain et rules pour la compréhension des requêtes utilisateurs, la gestion des scénarios de conversation, et les actions conditionnelles (authentification, recherche, navigation, etc.)	Thinhinane	Haute	23h	À faire
	Développer l'action personnalisée pour rechercher un article avec extraction de l'ID de l'article en fonction des critères (nom, catégorie, sous-catégorie, prix.)	Thinhinane	Haute	8h	À faire
	Créer l'action personnalisée de navigation pages pour diriger l'utilisateur vers toutes les pages de l'application (produits, profil, etc.)	Lydia	Haute	7h	À faire
	Implémenter l'action personnalisée pour récupérer les utilisateurs et gérer leur mode (connecté, non connecté)	Thinhinane	Haute	5h	À faire
	Tester l'interface utilisateur complète	Lydia	Moyenne	2h	À faire
	<i>Suite à la page suivante...</i>				

Fonctionnalité	Tâche	Responsable	Priorité	Estimation (h)	Statut
----------------	-------	-------------	----------	----------------	--------

TABLE 6.1 – Product Backlog du sprint 6

## 2.6 Décomposition du sprint

Ce sprint est réparti sur deux semaines avec une allocation du temps dédiée à la conception, au développement et aux tests des différentes fonctionnalités listées précédemment. L'accent est mis sur le développement du chatbot intelligent, comme illustré dans la figure 6.2.

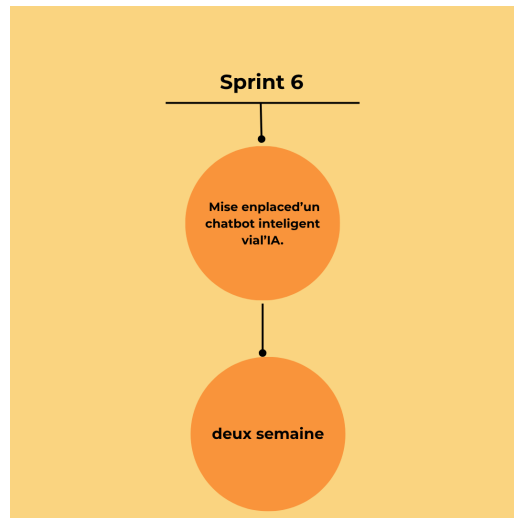


FIGURE 6.2 – Décomposition du Sprint 6

## 2.7 Présentation des diagrammes de séquence

Dans cette section, nous présentons la description détaillée des fonctionnalités majeures de ce sprint, accompagnée des diagrammes de séquence illustrant le flux des actions pour les fonctionnalités "Rechercher un produit", illustrée dans la figure 6.3, et "Naviguer vers une page", présentée dans la figure 6.4.

### Scénario principal (cas normal) : Rechercher un produit

1. L'utilisateur ou le visiteur tape "**Je cherche un produit X**" dans l'interface du chatbot.
2. Le chatbot(frontend) envoie le message à Rasa pour le traitement.
3. Rasa reçoit la requête envoyée par le chatbot, analyse le texte pour en extraire les entités pertinentes du produit recherché (nom, taille, catégorie, etc.).
4. Après l'analyse, Rasa interroge la base de données via son URI pour récupérer les produits correspondant aux entités extraites.
5. La base de données renvoie les produits trouvés à Rasa.
6. Rasa traite les données reçues et les renvoie au chatbot (frontend).
7. Le chatbot (frontend) reçoit les données envoyées par Rasa et affiche la liste des produits trouvés à l'utilisateur.

## Scénario alternatif (cas d'erreur) : Rechercher un produit

1. **Aucun produit trouvé** → Si aucun produit n'est trouvé dans la base de données, le système renvoie une réponse indiquant qu'aucun produit n'a été trouvé.
2. **Chatbot n'a pas compris la question** → Si la question est mal formulée, le chatbot renvoie une réponse demandant à l'utilisateur de reformuler sa question ou de préciser davantage sa recherche.

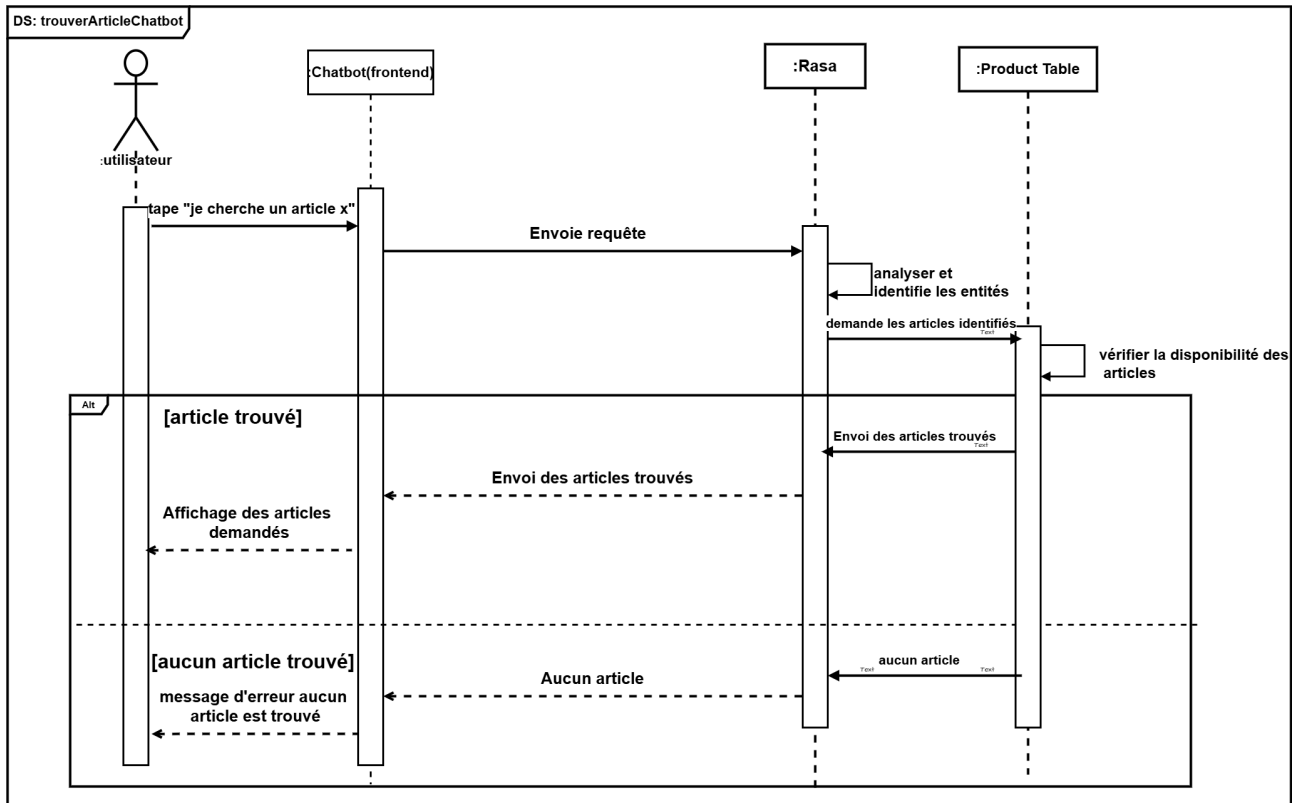


FIGURE 6.3 – Diagramme de séquence - Fonctionnalité "Rechercher un produit"

Dans cette description, nous donnons un exemple de navigation vers la page de vente.

## Scénario principal (cas normal) : Naviguer vers la page de vente

1. L'utilisateur doit s'authentifier et tape "Je veux accéder à la page de vente" dans l'interface du chatbot.
2. le Chatbot (Frontend) Envoie le message à Rasa pour l'analyser .
3. Rasa Analyse l'intention de l'utilisateur et vérifie Si l'utilisateur est authentifié, Ensuite, renvoyer un lien vers la page de vente au frontend.
4. Le chatbot (Frontend) reçoit le lien et l'affiche à l'utilisateur pour l'orienter vers la page de vente.

## Scénario alternatif (cas d'erreur) : Naviguer vers la page de vente

1. **Utilisateur non connecté** → Si l'utilisateur n'est pas connecté, le chatbot renvoie un message indiquant : "Vous devez vous connecter ou vous inscrire pour accéder à la page de vente."

2. **Question mal formulée** → Si la question est mal formulée, le chatbot renvoie une réponse demandant à l'utilisateur de reformuler sa question ou de préciser davantage sa recherche.

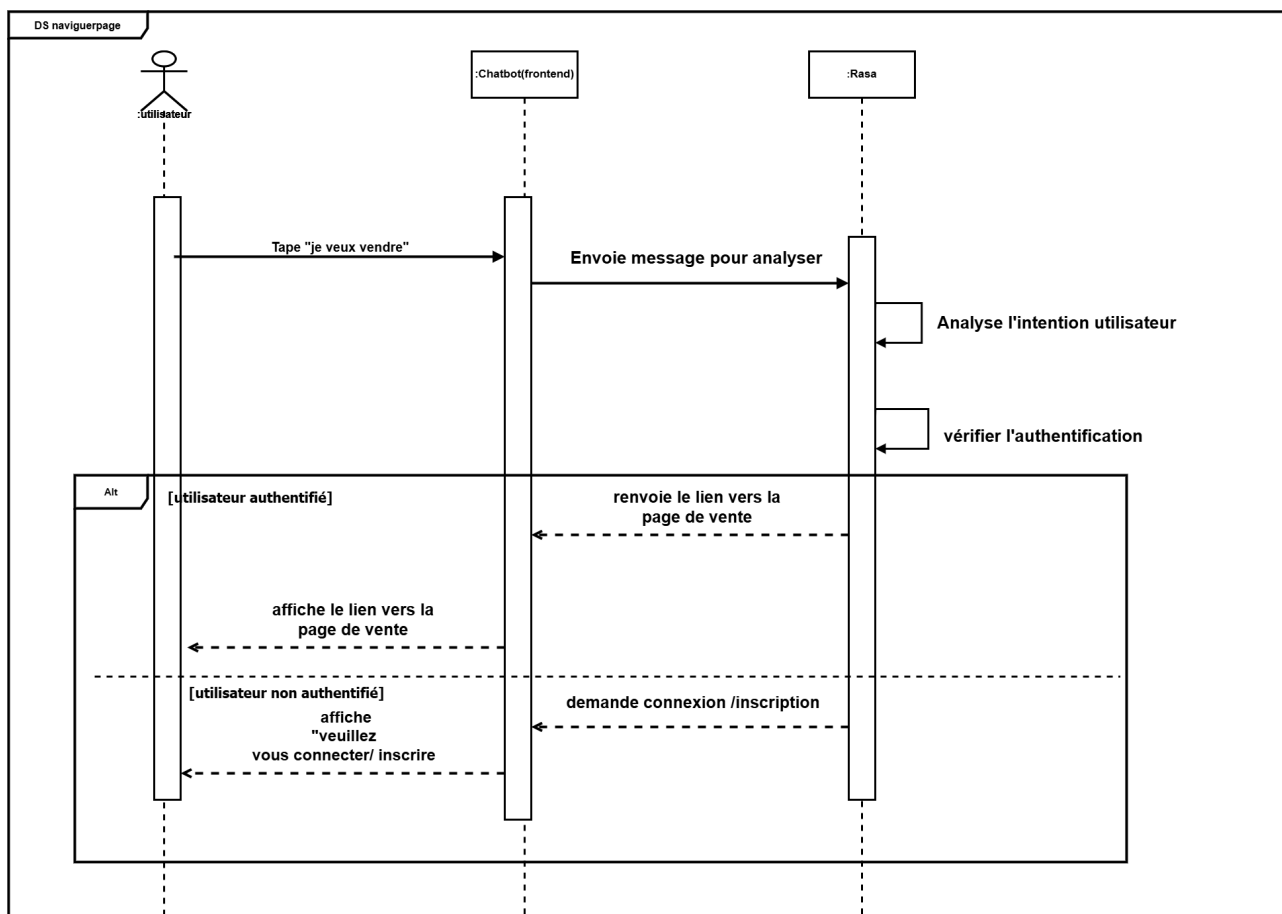


FIGURE 6.4 – Diagramme de séquence - Fonctionnalité "Naviguer vers la page de vente"

### 3 Vue des Interfaces Utilisateur

Dans cette section, nous exposons les interfaces graphiques essentielles développées lors du sprint 6

#### 3.1 Interface de recherche d'article via le chatbot

La figure 6.5 montre l'interface qui permet aux utilisateurs de chercher un article en interrogeant le chatbot.

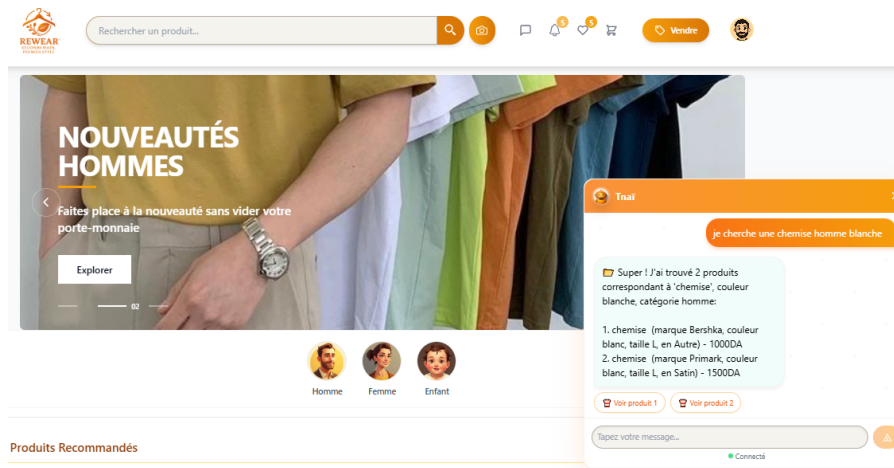
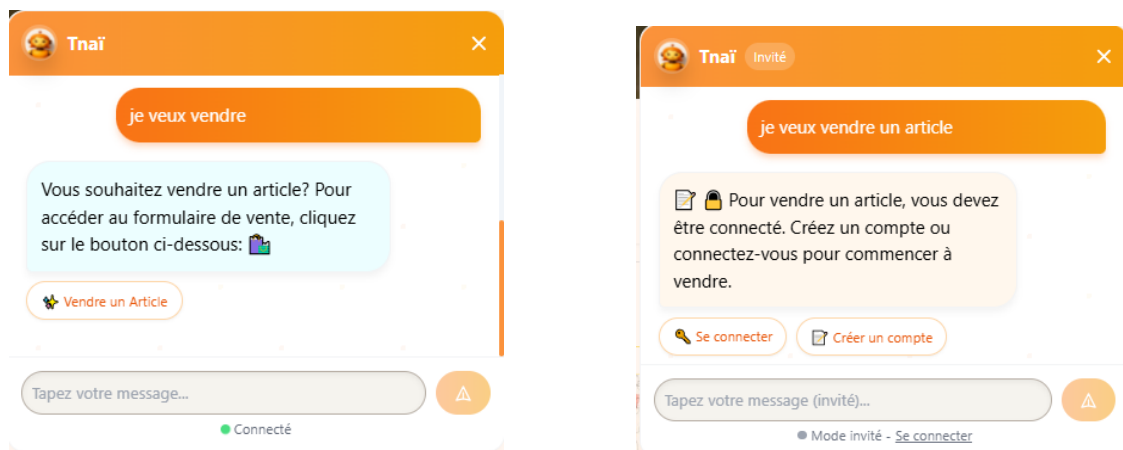


FIGURE 6.5 – Interface de recherche d'article via chatbot

### 3.2 Interface de navigation entre pages via le chatbot

L'interface présentée dans la figure 6.6 illustre la manière dont l'utilisateur peut naviguer entre les différentes pages grâce à l'assistance du chatbot.



(a) Interface de navigation entre pages pour un utilisateur authentifié (b) Interface de navigation entre pages pour un utilisateur non authentifié

FIGURE 6.6 – Interface de navigation entre pages via un chatbot

## 4 Conclusion

En conclusion, l'intégration du chatbot intelligent basé sur le framework Rasa a permis d'améliorer considérablement l'expérience utilisateur sur la plateforme Rewear. Grâce à cette solution, les utilisateurs peuvent désormais effectuer des recherches personnalisées pour les articles, avec une réponse précise à leurs requêtes. De plus, le chatbot assure une navigation fluide sur le site en guidant les utilisateurs efficacement à travers les différentes pages. Cette capacité à répondre aux besoins spécifiques des utilisateurs, tout en offrant une interaction naturelle et intuitive, marque une avancée significative dans l'optimisation de l'expérience sur la plateforme. Cependant, certaines limites persistent, notamment dans la gestion des requêtes complexes ou ambiguës, où le chatbot peut encore rencontrer des difficultés à fournir des réponses pertinentes ou à comprendre des contextes spécifiques. Le chapitre suivant présentera le Sprint 8, qui mettra l'accent sur l'intégration de la recherche par image.

# Chapitre 7

## Sprint 7

# Intégration de la Recherche visuelle avec l'Intelligence Artificielle

## 1 Introduction

Ce chapitre présente le Sprint 7 de notre plateforme Rewear, qui introduit une fonctionnalité innovante de recherche par image alimentée par des modèles d'intelligence artificielle. L'objectif principal de ce sprint est d'améliorer l'expérience utilisateur en permettant aux utilisateurs de rechercher des articles similaires simplement en téléchargeant une image. Cette fonctionnalité repose sur l'intégration de technologies avancées, notamment l'API Hugging Face, ainsi que des méthodes d'analyse d'image telles que le K-Means Clustering et l'analyse des couleurs. Nous détaillerons les outils utilisés, les diagrammes de séquence, ainsi que les interfaces utilisateur qui rendent cette recherche intuitive et efficace.

### 1.1 Outils utilisés pour l'implémentation de la recherche visuelle

#### 1.1.1 API Hugging Face

- **Présentation de l'API Hugging Face**

Hugging Face est un hub hébergeant plus de 500 000 modèles pré-entraînés accessible via des API standardisées qui permettent aux développeurs d'intégrer facilement des capacités d'IA (génération de texte, traduction, synthèse vocale, analyse d'images) dans leurs applications.[38]

- **Les modèles Hugging Face utilisé pour l'analyse d'image**

- **Vision Transformer (ViT) - google/vit-base-patch16-224** : Est un modèle d'encodage qui applique l'architecture Transformer aux images en les divisant en patches de 16x16 pixels traités comme une séquence via un mécanisme d'attention.[39]
- **ResNet-50 - microsoft/resnet-50** : Est un réseau de neurones convolutif à 50 couches qui utilise des connexions résiduelles pour résoudre le problème de disparition du gradient et atteindre des performances exceptionnelles en vision par ordinateur.[40]
- **EfficientNet-B0 - google/efficientnet-b0** : Est un CNN développé par Google Research qui optimise simultanément profondeur, largeur et résolution via le com-

pound scaling et utilise des blocs MBConv avec attention par canaux pour exceller en analyse d'images.[41]

- **ConvNeXt-Tiny - facebook/convnext-tiny-224** : Développé par Facebook AI Research, modernise les CNN en intégrant des techniques des Transformers (normalisation par couches, activation GELU) pour exceller en classification, détection et segmentation d'images.[42]

### 1.1.2 Techniques d'analyse d'image

- **K-Means Clustering** : Est un algorithme fondamental en machine learning, il permet de rassembler les pixels aux couleurs similaires en  $K$  groupes pour identifier les teintes dominantes d'une image. Chaque groupe représente une couleur principale, permettant d'extraire les palettes chromatiques et de segmenter l'image par couleur.[43] Ce processus est illustré dans la figure 7.1

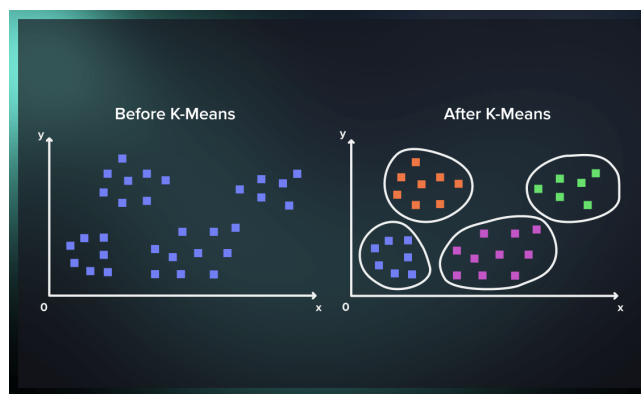


FIGURE 7.1 – Illustration de l'algorithme **K-means**.

- **Color Mapping** : Est une technique de traitement d'images qui associe les couleurs détectées à des noms standardisés (RAL, Pantone, NCS) en calculant la distance colorimétrique avec une base de données de référence. Utilise un algorithme de plus proche voisin pour identifier la correspondance la plus appropriée.[44]
- **Pillow (PIL)** : Est une bibliothèque Python essentielle pour l'analyse d'images, permettant ainsi le prétraitement des données visuelles par redimensionnement, ajustement des couleurs et application de filtres. En combinant **Numpy** et **Pillow** l'analyse de l'images la conversion des images en matrices numériques devient plus facile, ce qui permet l'application d'algorithmes d'analyse avancés pour l'analyse des images.[45]

## 2 Diagramme de cas d'utilisation

Les interactions entre les différents acteurs (utilisateurs) et les fonctionnalités développées au cours de ce sprint sont représentées dans le diagramme de cas d'utilisation présenté dans la figure 7.2

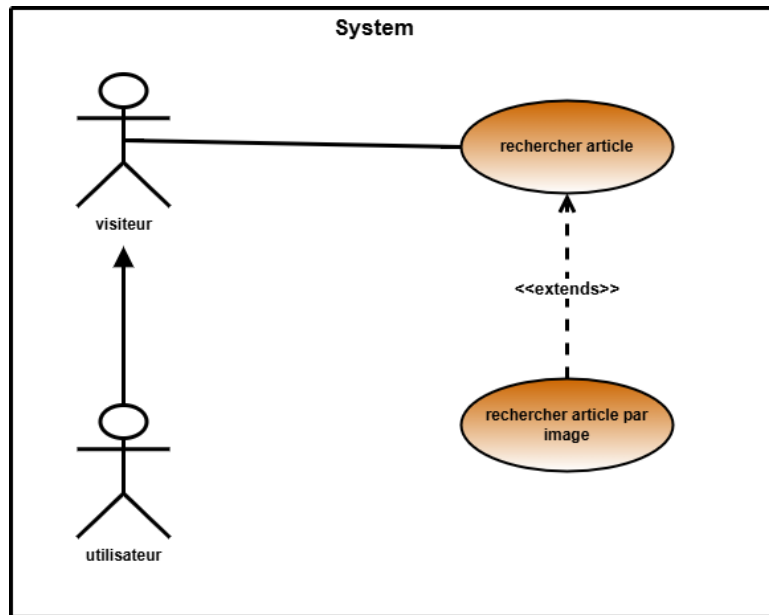


FIGURE 7.2 – Diagramme de cas d'utilisation dusprint 7

### 3 Product Backlog

Le tableau ci-dessous présente le backlog détaillé de la recherche par image, listant les fonctionnalités développées, les tâches associées, les responsables, les priorités, l'estimation d'effort et le statut.

Fonctionnalité	Tâche	Responsable	Priorité	Estimation (heures)	Statut
Implémentation de la recherche visuelle avec intégration de l'IA	Créer le code python pour le redimensionnement, ajustement de la couleur et du contraste des images.	Thinhinane	Haute	20h	À faire
	Configuration Hugging Face, implémentation et intégration de modèles	Lydia	Haute	20h	À faire
	Implémentation des techniques d'analyse de couleurs	Thinhinane	Haute	20h	À faire
	Fusion des analyses IA et des couleurs et mappage aux produits dans la base de données.	Lydia	Haute	13h	À faire
	Développement de la logique de recherche des produits correspondants et de classement des résultats par pertinence.	Thinhinane	Haute	20h	À faire
	Création de l'interface de recherche par image	Lydia	Haute	7h	À faire

TABLE 7.1 – Product Backlog du sprint 7

### 4 Décomposition du sprint

Ce sprint est réparti sur une semaines avec une allocation du temps dédiée à la conception, au développement et aux tests des différentes fonctionnalités listées précédemment. L'accent est mis sur le développement de la recherche par image, comme illustré dans la figure 7.3

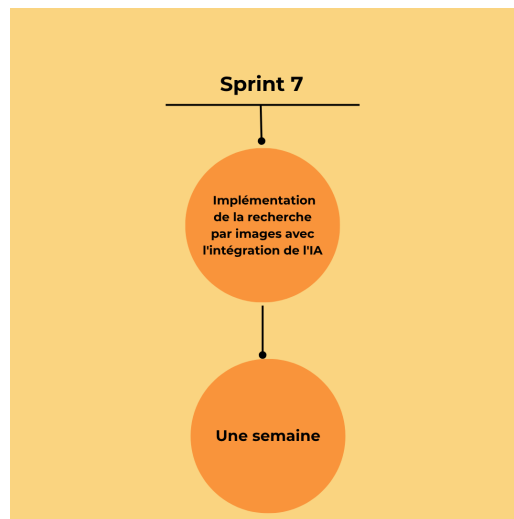


FIGURE 7.3 – Décomposition du Sprint 7

## 5 Présentation du diagramme de séquence

Nous présentons ci-dessous la description détaillée de la fonctionnalité Implémentaion de la recherche par image avec intégration de l'IA, accompagnée du diagramme de séquence illustré dans la figure 7.4.

### 5.0.1 Fonctionnalité « Rechercher un article par une image »

#### Scénario principal(cas normal)

1. **Recherche d'un article** : L'utilisateur ajoute une image pour rechercher un article similaire via l'interface de la recherche
2. **Envoi de l'image au Backend** :L'interface envoie l'image au serveur backend via l'API FastAPI pour traitement.
3. **Prétraitement de l'image** : L'API FastAPI traite l'image envoyé par redimensionnement, conversion, et optimisation de l'image pour l'analyse.
4. **Analyse de l'image** : Les modèles(Vision Transformer et CNN) importé depuis Huggin Face analysent l'image pour identifier les objets et extraire les caractéristiques.
5. **Analyse des couleurs** : KMeans analyse les couleurs dominantes dans l'image pour une recherche plus affinée.
6. **Consolidation des résultats** : le backend consolide les résultat obtenus des analyses précédantes pour correspondre aux catégories et critères de recherche.
7. **Recherche des articles correspondants dans la base de données** : Le système effectue une recherche dans la base de données des articles (TableProduct) en fonction des résultats obtenus pour vérifier la disponibilité des articles.
8. **Affichage des articles correspondants** : Si les articles sont disponible le système affiche les articles similaire via l'interface de recherche.

#### Scénarios alternatifs (erreurs)

- **Erreur de traitement d'image**

→ Si l'image ne correspond pas aux critères nécessaires (taille, format), le système demande à l'utilisateur de télécharger une nouvelle image dans le bon format.

- **Aucun article correspondant :**  
→ Message d'erreur "Aucun article correspondant" est affiché.
- **Problème de connexion à la base de données :**  
→ Message d'erreur "Problème de connexion" et la recherche s'arrête.

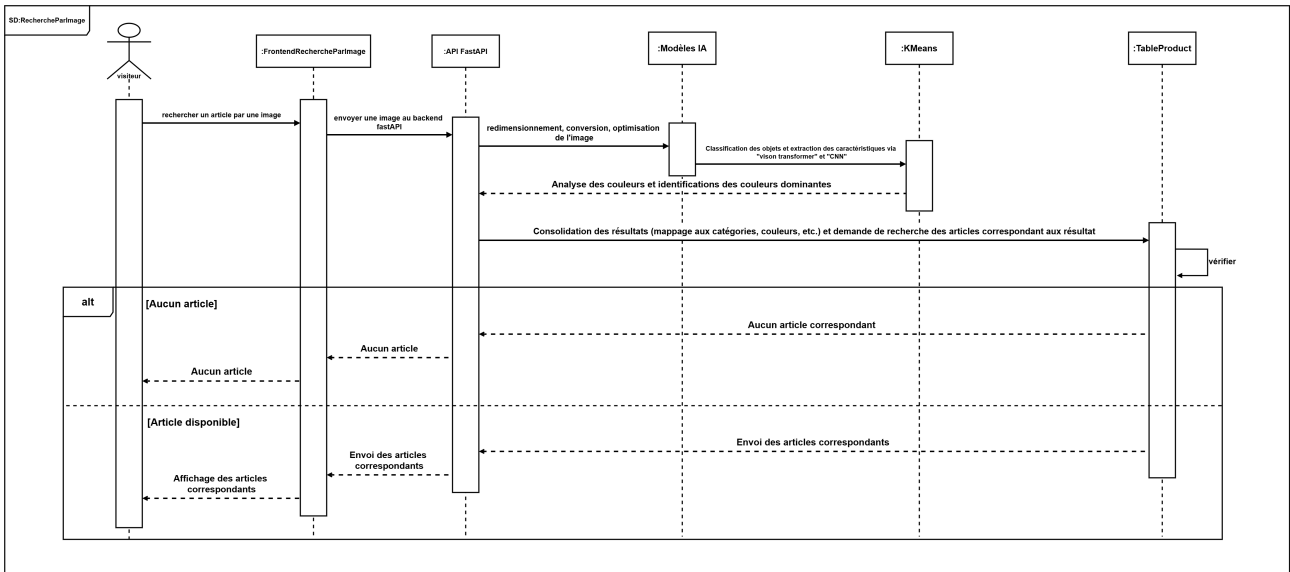


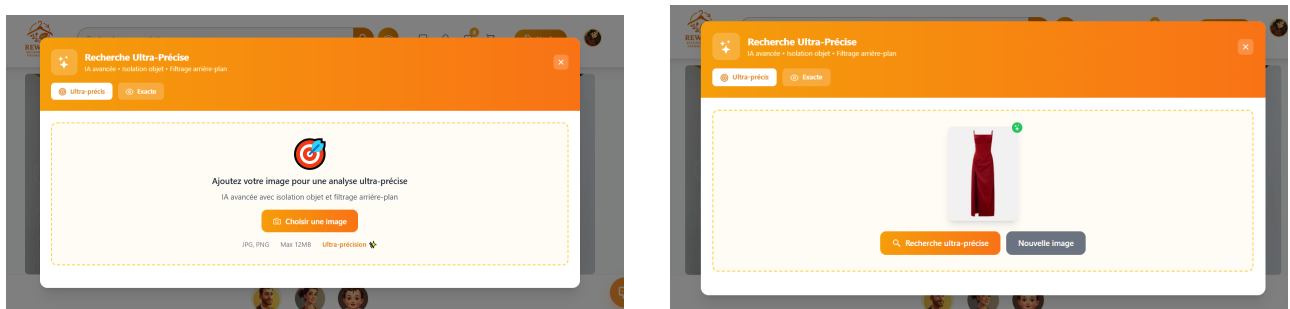
FIGURE 7.4 – Diagramme de séquence pour la fonctionnalité **recherche par image**

## 6 Vue des Interfaces Utilisateur

Dans cette section, nous présentons les interfaces graphiques essentielles développées lors du Sprint 7.

### 6.1 Interface de recherche par image

La figure 7.5 illustre l'interface permettant à l'utilisateur d'ajouter une image pour rechercher un article similaire à celui représenté sur la photo.



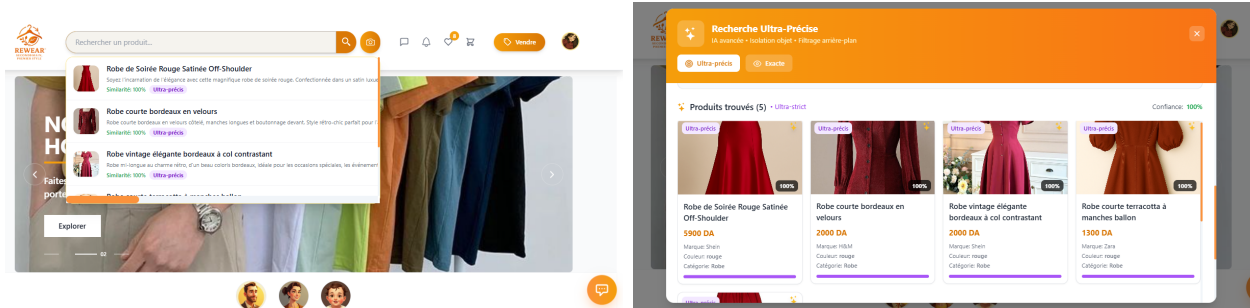
(a) Section d'ajout de l'image

(b) Affichage de l'image ajoutée

FIGURE 7.5 – Recherche d'un article par image

## 6.2 Interface des résultats de la recherche

La figure 7.6 illustre l'interface permettant à l'utilisateur de visualiser les résultats de sa recherche par image. Pour maximiser l'efficacité de cette fonctionnalité, il est conseillé d'utiliser des images respectant certaines bonnes pratiques : éviter les images floues, utiliser un fond neutre et s'assurer d'une résolution suffisante.



(a) Première section d'affichage des résultats de la recherche

(b) Deuxième section d'affichage des résultats de la recherche

FIGURE 7.6 – Résultats de la recherche par image

## 7 Conclusion

En conclusion, ce chapitre met en évidence le Sprint 7, qui a permis l'ajout d'une fonctionnalité de recherche par image à notre plateforme Rewear. En tirant parti des modèles d'intelligence artificielle tels que Vision Transformer et les CNN, combinés à des techniques d'analyse des couleurs, cette nouvelle fonctionnalité enrichit considérablement l'expérience d'achat. Ce sprint a non seulement optimisé la recherche de produits en ligne, mais a également amélioré l'interface utilisateur pour offrir une interaction plus fluide et intuitive.

# Conclusion Générale

Le projet ReWear répond de manière stratégique et innovante aux défis du marché des vêtements de seconde main en Algérie. Dans un secteur encore dominé par l’informalité, cette plateforme numérique offre une alternative complète, sécurisée et structurée, visant à faciliter l’accès à une consommation plus responsable tout en assurant la sécurité des transactions et en modernisant le secteur.

ReWear met l’accent sur l’expérience utilisateur, avec une interface fluide et intuitive. L’application propose des fonctionnalités clés telles que l’authentification des utilisateurs et un système de recommandation intelligent basé sur l’intelligence artificielle. La recherche par image et le chatbot conversationnel enrichissent l’interaction avec les utilisateurs, offrant une assistance personnalisée. De plus, le paiement sécurisé via Chargily garantit des transactions fiables et sûres. La qualité du projet est mesurée par l’efficacité de la plateforme, sa stabilité, et la satisfaction des utilisateurs.

Le développement de l’application a suivi une approche méthodologique Agile, structurée autour de sept sprints successifs, permettant une évolution progressive et cohérente du projet. Chaque sprint a apporté des éléments fonctionnels clés, tout en assurant une gestion optimale des priorités et une résolution rapide des défis techniques.

La plateforme repose sur les technologies MERN (MongoDB, Express.js, React, Node.js), offrant ainsi une infrastructure performante, évolutive et sécurisée. MongoDB assure une gestion flexible des données, tandis qu’Express.js et Node.js garantissent une solide infrastructure backend. React a permis de créer une interface dynamique et réactive, offrant une expérience fluide et rapide.

Pour renforcer la plateforme, plusieurs pistes d’amélioration sont envisagées, comme l’ajout d’une application mobile, un système de détection de fraude par intelligence artificielle et un système de recommandation de prix pour les vendeurs. Ces améliorations viendront enrichir l’impact de ReWear sur l’écosystème local et favoriseront l’économie circulaire en Algérie.

Ce projet a été une expérience formatrice, démontrant comment l’utilisation de technologies avancées peut transformer un secteur traditionnel en un écosystème numérique moderne. ReWear marque le début d’une transformation du commerce en ligne en Algérie, contribuant à structurer le marché des vêtements de seconde main tout en favorisant l’inclusion économique et la responsabilité environnementale. Cette solution répond à des enjeux sociétaux et économiques et participe activement au développement de l’économie numérique du pays.

# Bibliographie

- [1] Nabila AMIR. *Commerce en ligne : 90% du marché reste informel*. [En ligne]. Disponible sur : <https://elwatan-dz.com/commerce-en-ligne-90-du-marche-reste-informel> (consulté le 16 juin 2025). mai 2025.
- [2] Liberté ALGÉRIE. *Le commerce de la fripe en pleine expansion*. [En ligne]. Disponible sur : <https://www.liberte-algerie.com/l-algerie-profonde/le-commerce-de-la-fripe-en-pleine-expansion-376164> (consulté le 15 mai 2025). 2023.
- [3] Tidjara DZ. *Chiffres clés pour l'activité du e-commerce en Algérie pour 2023 et 2024*. [En ligne]. Disponible sur : <https://tidjara.dz/chiffres-cles-pour-lactivite-du-e-commerce-en-algerie-pour-2023-et-2024/> (consulté le 17 mai 2025). Jan. 2025.
- [4] VERSUSDZ. *Consommation responsable en Algérie : pourquoi et comment changer nos habitudes*. [En ligne]. Disponible sur : <https://www.versusdz.com/blog/consommation-responsable-en-algerie-pourquoi-et-comment-changer-nos-habitudes> (consulté le 15 mai 2025). 2023.
- [5] EL MOUDJAHID. *Livre blanc sur l'e-commerce et l'e-paiement en Algérie : état des lieux*. [En ligne]. Disponible sur : <https://www.elmoudjahid.dz/fr/economie/livre-blanc-sur-l-e-commerce-et-l-e-paiement-en-algerie-etat-des-lieux-177544> (consulté le 15 mai 2025). 2023.
- [6] Eugène LAUJOLET. *Le commerce en Algérie : notes sur le peuplement utile de l'Afrique française*. [En ligne]. Disponible sur : <https://www.hachettebnf.fr/livre/le-commerce-en-algerie-9782329838243/> (consulté le 15 mai 2025). Paris : Hachette BNF, 1851.
- [7] Roberto BURDESE. "La consommation responsable". In : *Consommateurs et consommation : XVII<sup>e</sup>-XX<sup>e</sup> siècle, regards franco-espagnols*. Sous la dir. de Nicolas MARTY et Antonio ESCUDERO. [En ligne]. Disponible sur : <https://books.openedition.org/pupvd/4728> (consulté le 15 mai 2025). Perpignan : Presses universitaires de Perpignan, 2015, p. 241-260.
- [8] Events FACTORY. *Vandista : application de vide-dressing*. [En ligne]. Disponible sur : <https://www.vandista.dz/> (consulté le 16 mai 2025). 2023.
- [9] CAWA.FR. *La politique commerciale d'Ouedkniss*. [En ligne]. Disponible sur : <https://www.cawa.fr/la-politique-commerciale-d-ouedkniss/> (consulté le 16 mai 2025). 2020.
- [10] OUEDKNISS. *Ouedkniss – petites annonces en Algérie*. [En ligne]. Disponible sur : <https://www.ouedkniss.com/> (consulté le 16 mai 2025). 2006.
- [11] Marie Seignol de SWARTE. *Facebook Marketplace : guide ultime, astuces et conseils*. [En ligne]. Disponible sur : <https://www.meltwater.com/fr/blog/facebook-marketplace> (consulté le 16 mai 2025). juillet 2024.

- [12] XP NUMÉRIQUE. *Comment fonctionne Facebook Marketplace ?* [En ligne]. Disponible sur : <https://www.xp-numerique.com/comment-fonctionne-facebook-marketplace> (consulté le 16 mai 2025). avril 2025.
- [13] META TRANSPARENCY CENTER. *Système d'IA de Facebook Marketplace : explication du classement.* [En ligne]. Disponible sur : <https://transparency.meta.com/features/explaining-ranking/fb-marketplace/> (consulté le 16 mai 2025). Mars 2025.
- [14] AXIOCODE. *Méthode agile vs classique, quelle méthode utiliser ?* [En ligne]. Disponible sur : <https://www.axiocode.com/methode-agile-vs-classique-quelle-methode-utiliser/> (consulté le 15 juin 2025). 2021.
- [15] Julie GIELEN. *Qu'est-ce que la méthodologie agile ?* [En ligne]. Disponible sur : <https://www.planzone.fr/blog/quest-ce-que-la-methodologie-agile> (consulté le 15 juin 2025). 2017.
- [16] Dan RADIGAN. *Sprint review : A step-by-step guide.* [En ligne]. Par Dan Radigan (Atlassian). Disponible sur : <https://www.atlassian.com/agile/scrum/sprint-reviews> (consulté le 15 juin 2025). juin 2025.
- [17] SCRUM.ORG. *What is a Product Owner ?* [En ligne]. Disponible sur : <https://www.scrum.org/resources/what-is-a-product-owner> (consulté le 16 juin 2025). 2025.
- [18] SCRUM.ORG. *What is a Scrum Master ?* [En ligne]. Disponible sur : <https://www.scrum.org/resources/what-is-a-scrum-master> (consulté le 16 juin 2025). 2025.
- [19] SCRUM.ORG. *What is a Developer in Scrum ?* [En ligne]. Disponible sur : <https://www.scrum.org/resources/what-is-a-developer-in-scrum> (consulté le 16 juin 2025). 2025.
- [20] ADDICTIC. *Charte graphique : le socle de votre identité visuelle et de votre image de marque.* [En ligne]. Disponible sur : <https://addictic.fr/detail-de-l-article/charte-graphique-le-socle-de-votre-identite-visuelle-et-de-votre-image-de-marque> (consulté le 25 mai 2025). avril 2025.
- [21] V-LABS. *Figma : Qu'est-ce que c'est ?* [En ligne]. Disponible sur : <https://www.v-labs.fr/glossaire/figma/> (consulté le 25 mai 2025). avril 2025.
- [22] Jeffrey ERICKSON. *Qu'est-ce que la pile MERN ? Guide & exemples.* [En ligne]. Disponible sur : <https://www.oracle.com/africa-fr/database/mern-stack/> (consulté le 25 mai 2025). Jan. 2024.
- [23] KICKLOX. *Qu'est-ce que la MERN Stack ? Tout ce qu'il faut savoir.* [En ligne]. Disponible sur : <https://www.kicklox.com/blog-talent/mern-stack-definition-enjeux-avantages/> (consulté le 25 mai 2025). février 2024.
- [24] Ihechikara ABBA. *Qu'est-ce que React.js ? Un regard sur la bibliothèque JavaScript populaire.* [En ligne]. Disponible sur : <https://kinsta.com/fr/base-de-connaissances/qu-est-react-js/> (consulté le 25 mai 2025). décembre 2024.
- [25] Ihechikara ABBA. *How to Use Tailwind CSS to Rapidly Develop Snazzy Websites.* [En ligne]. Disponible sur : <https://kinsta.com/blog/tailwind-css/> (consulté le 25 mai 2025). Sept. 2023.
- [26] Aris SENTIKA. *Qu'est-ce que Node.js : Comment fonctionne-t-il et comment démarrer.* [En ligne]. Disponible sur : <https://www.hostinger.com/fr/tutoriels/quest-ce-que-node-js> (consulté le 25 mai 2025). Mars 2025.
- [27] Solomon ESEME. *Qu'est-ce qu'Express.js ? Tout ce que vous devez savoir.* [En ligne]. Disponible sur : <https://kinsta.com/fr/base-de-connaissances/qu-est-express-js/> (consulté le 25 mai 2025). décembre 2024.

- [28] Raphaël KASSEL. *MongoDB : tout savoir sur la base de données NoSQL orientée document*. [En ligne]. Disponible sur : <https://datascientest.com/mongodb> (consulté le 25 mai 2025). février 2021.
- [29] Thomas GROLLEAU. *Qu'est-ce que Cloudinary et comment l'utiliser ?* [En ligne]. Disponible sur : <https://www.journaldufreenaute.fr/quest-ce-que-cloudinary-et-comment-lutiliser/> (consulté le 25 mai 2025). juin 2023.
- [30] SOCKET.IO. *Introduction | Socket.IO*. [En ligne]. Disponible sur : <https://socket.io/fr/docs/v4/> (consulté le 25 mai 2025). avril 2025.
- [31] Lorenzo CARTE et Amin TRAORÉ. *JWT (JSON Web Token) : vulnérabilités, attaques courantes et bonnes pratiques sécurité*. [En ligne]. Disponible sur : <https://www.vaadata.com/blog/fr/jwt-json-web-token-vulnerabilites-attaques-courantes-et-bonnes-pratiques-securite/> (consulté le 25 mai 2025). avril 2025.
- [32] RECOMBEE. *Recombee – Système de recommandation en temps réel alimenté par l'IA*. [En ligne]. Disponible sur : <https://www.recombee.com/> (consulté le 10 juin 2025). 2025.
- [33] Pavel KORDÍK. *Machine Learning for Recommender Systems—Part1 (algorithms, evaluation and cold start)*. [En ligne]. Disponible sur : <https://medium.com/recombee-blog/machine-learning-for-recommender-systems-part-1-algorithms-evaluation-and-cold-start-6f696683d0ed> (consulté le 10 juin 2025). juin 2018.
- [34] RASA TECHNOLOGIES GMBH. *Welcome to the Rasa Docs*. [En ligne]. Documentation officielle. Disponible sur : <https://rasa.com/docs/> (consulté le 10 juin 2025). juillet 2025.
- [35] GROUPE ONEPOINT. *Comment développer son Chatbot avec Rasa ?* [En ligne]. Disponible sur : <https://www.groupeonepoint.com/fr/publications/comment-developper-son-chatbot-avec-rasa/> (consulté le 10 juin 2025). 2025.
- [36] GROUPE ONEPOINT. *Développez votre chatbot avec Rasa (partie 2)*. [En ligne]. Disponible sur : <https://www.groupeonepoint.com/fr/publications/comment-developper-son-chatbot-avec-rasa-entrainement-deploiement/> (consulté le 10 juin 2025). 2025.
- [37] RASA TECHNOLOGIES GMBH. *Conversational AI Platform | Superior Customer Experiences Start Here*. [En ligne]. Disponible sur : <https://rasa.com/> (consulté le 10 juin 2025). 2025.
- [38] HUGGING FACE. *Hugging Face Hub : plateforme de modèles IA*. [En ligne]. Disponible sur : <https://huggingface.co/> (consulté le 14 juin 2025). 2024.
- [39] HUGGING FACE. *Vision Transformer (ViT) – google/vit-base-patch16-224*. [En ligne]. Disponible sur : <https://huggingface.co/google/vit-base-patch16-224> (consulté le 14 juin 2025). 2024.
- [40] HUGGING FACE. *ResNet-50 – microsoft/resnet-50*. [En ligne]. Disponible sur : <https://huggingface.co/microsoft/resnet-50> (consulté le 14 juin 2025). 2024.
- [41] HUGGING FACE. *EfficientNet-B0 – google/efficientnet-b0*. [En ligne]. Modèle pré-entraîné mis à jour le 16 juillet 2024. Disponible sur : <https://huggingface.co/google/efficientnet-b0> (consulté le 14 juin 2025). 2024.
- [42] HUGGING FACE. *ConvNeXt-Tiny – facebook/convnext-tiny-224*. [En ligne]. ConvNeXt-Tiny est un modèle purement convolutionnel introduit dans *\*A ConvNet for the 2020s\**. Disponible sur : <https://huggingface.co/facebook/convnext-tiny-224> (consulté le 14 juin 2025). 2024.

- [43] Sarah LEE. *K-Means for Image Analysis*. [En ligne]. Disponible sur : <https://www.numberanalytics.com/blog/k-means-image-analysis> (consulté le 14 juin 2025). juin 2025.
- [44] H. Sheikh FARIDUL et al. “Colour Mapping : A Review of Recent Methods, Extensions and Applications”. In : *Computer Graphics Forum* 35.1 (2016). [En ligne]. Disponible sur : <https://onlinelibrary.wiley.com/doi/10.1111/cgf.12671> (consulté le 14 juin 2025), p. 1-21. DOI : 10.1111/cgf.12671.
- [45] PILLOW CONTRIBUTORS. *Pillow (PIL Fork) 11.2.1 Documentation*. [En ligne]. Documentation de la version 11.2.1 publiée le 12avril2025. Disponible sur : <https://pillow.readthedocs.io/en/stable/reference/Image.html> (consulté le14juin 2025). avril 2025.
- [46] Sébastien CHOPIN. *Utiliser MongoDB avec Node.js grâce à Mongoose*. [En ligne]. Disponible sur : <https://atinux.developpez.com/tutoriels/javascript/mongodb-nodejs-mongoose/> (consulté le 25 mai 2025). février 2013.
- [47] ÉQUIPE SLACK. *Qu'est-ce qu'une équipe agile et quels sont ses objectifs ?* [En ligne]. Par l'équipe Slack, 8 novembre 2024. Disponible sur : <https://slack.com/intl/fr-fr/blog/collaboration/equipe-agile> (consulté le 15 juin 2025). Nov. 2024.
- [48] QRP INTERNATIONAL. *Scrum, c'est quoi ? Définition du framework*. [En ligne]. Publié le 25 octobre 2022. Disponible sur : <https://www.qrpinternational.fr/blog/glossaire/scrum-cest-quoi-definition-scrum/> (consulté le 15 juin 2025). Oct. 2022.

# Résumé

## **ReWear : Plateforme Numérique pour le Marché des Vêtements de Seconde Main en Algérie**

Le marché des vêtements de seconde main en Algérie connaît une croissance notable, alimentée par une conscience écologique croissante et un contexte économique difficile. Cependant, ce secteur demeure principalement informel, avec environ 90% des transactions effectuées via des canaux non structurés, entraînant des défis en matière de confiance, de sécurité et d'efficacité. Le projet ReWear propose une solution innovante sous forme d'une plateforme web dédiée à l'achat et à la vente de vêtements d'occasion. La plateforme intègre des technologies avancées incluant un chatbot intelligent, un système de recommandation personnalisé basé sur l'IA, une fonctionnalité de recherche par image, et un système de paiement sécurisé via Chargily. Développée selon une méthodologie Agile en sept sprints successifs, ReWear utilise la stack technologique MERN (MongoDB, Express.js, React, Node.js) pour offrir une infrastructure performante et évolutive. L'objectif est de digitaliser et sécuriser les échanges tout en contribuant à une économie circulaire responsable, créant ainsi une valeur économique et sociale pour l'écosystème local algérien. Cette solution répond stratégiquement aux enjeux sociétaux et économiques, participant activement au développement de l'économie numérique du pays tout en favorisant l'inclusion économique et la responsabilité environnementale.

**Mots-clés :** Vente de vêtements d'occasion, Marché de la mode durable, Plateforme de vente en ligne, Économie circulaire, Consommation responsable, Mode de seconde main, Digitalisation du marché, Vente de vêtements d'occasion en ligne, Commerce responsable.

# Abstract

## **ReWear : Digital Platform for the Second-Hand Clothing Market in Algeria**

The pre-owned clothing market in Algeria is growing due to increased environmental awareness and economic challenges. However, it remains largely informal, with about 90% of transactions occurring through unstructured channels, posing challenges in trust, security, and efficiency. The ReWear project offers an innovative solution in the form of a web platform dedicated to buying and selling second-hand clothing. The platform integrates advanced technologies including an intelligent chatbot, AI-based personalized recommendation system, image search functionality, and secure payment system via Chargily. Developed using an Agile methodology across seven successive sprints, ReWear utilizes the MERN technology stack (MongoDB, Express.js, React, Node.js) to provide a performant and scalable infrastructure. The objective is to digitalize and secure exchanges while contributing to a responsible circular economy, thus creating economic and social value for the local Algerian ecosystem. This solution strategically addresses societal and economic challenges, actively participating in the country's digital economy development while promoting economic inclusion and environmental responsibility.

**Keywords :** Second-hand clothing sales, Sustainable fashion market, Online selling platform, Circular economy, Responsible consumption, Pre-owned fashion, Resale of clothing, Digitalization of the market, Second-hand clothing online sales, Responsible commerce.