

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira de Béjaïa

Faculté des Sciences Exactes
Département Informatique



Mémoire fin de cycle

En vue de l'obtention du diplôme de Master Professionnel en Informatique

Option : Génie Logiciel

Thème

Conception et réalisation d'une application
informatique dédiée aux activités philanthropiques

Réalisé par :

HAMIS Sonia
BRAHAM Zakaria

Soutenu le **02/07/2023** devant le jury composé de :

Président	M. AKILAL	Abdellah
Examinatrice	Mme. AIT HACENE	Souhila
Encadreur	M. ACHROUFENE	Achour

Année universitaire 2022/2023

Remerciements

À la clôture de ce travail, nous souhaitons exprimer notre profonde gratitude envers le Dieu clément qui nous a accordé la force et le courage nécessaires pour mener à bien ce projet.

Nous tenons à remercier chaleureusement nos chers parents pour leur soutien constant et leur encouragement tout au long de notre formation académique. Leur présence bienveillante et leurs sacrifices ont été la raison de notre réussite.

Nous exprimons également notre profonde reconnaissance envers notre encadrant, Monsieur A. ACHROUFENE, pour sa disponibilité, sa patience et son précieux accompagnement tout au long de la réalisation de ce travail. Ses conseils avisés et son suivi attentif ont grandement contribué à notre progression.

De même, nous souhaitons exprimer notre gratitude envers les membres du jury pour avoir consacré une partie de leur temps à examiner ce mémoire, pour l'intérêt qu'ils ont porté à notre travail et pour leurs contributions à l'enrichir.

Nos remerciements s'adressent également à tous les enseignants qui ont participé à notre formation tout le long de notre cursus.

Enfin, nous souhaitons adresser nos sincères remerciements à toutes les personnes qui ont contribué à l'élaboration de ce mémoire.

Nous sommes reconnaissants envers chacune de ces personnes pour leur soutien et leur aide précieuse tout au long de ce parcours universitaire.

Dédicaces

À mon binôme, qui a partagé ce voyage avec moi, je souhaite exprimer ma profonde gratitude. Ensemble, nous avons surmonté les défis, échangé des idées et soutenu mutuellement. Ta persévérance, ta collaboration et ton engagement ont été des atouts précieux tout au long de cette expérience.

À mes parents, qui ont été ma source inépuisable d'amour, de soutien et d'encouragement tout au long de ce parcours. Votre foi en moi m'a donné la force de persévérer et de réaliser mes rêves. Vous êtes mes piliers, ma motivation et mon inspiration constants.

À mon frère Ali, qui a toujours été là pour me tendre la main lorsque j'en avais besoin.

À ma sœur Maya, ma complice et mon amie, merci d'avoir partagé les hauts et les bas de ce chemin avec moi.

À Massiva, ma petite sœur bien-aimée, j'espère pouvoir dresser un exemple inspirant pour toi. Ton regard innocent et ta présence lumineuse dans ma vie sont une source de joie et de motivation inépuisables. Ce travail t'est dédié avec tout mon amour et dans l'espoir que tu puisses trouver en moi un modèle inspirant pour ton propre parcours.

À ma tante GALI Lynda, tu as toujours été une source d'inspiration et un modèle pour moi. Ta détermination, ton intelligence et ta bienveillance ont été des guides précieux dans ma vie.

À mes amies et amis, vous êtes la preuve vivante que le soutien et l'amitié sont des éléments essentiels pour surmonter les défis. Vos encouragements et votre présence bienveillante ont rendu ce voyage plus riche et agréable. Je vous remercie sincèrement d'avoir été là pour moi.

Enfin, à tous ceux qui ont contribué de près ou de loin à ce travail, que ce soit par vos conseils, vos relectures, vos encouragements ou vos discussions inspirantes, je vous suis infiniment reconnaissante.

HAMIS Sonia

À mon binôme, je suis très reconnaissant d'avoir eu la chance de travailler avec une personne aussi dévouée, talentueuse et engagée. Ta passion et ton professionnalisme ont inspiré en moi une volonté d'excellence et ont contribué à notre succès commun. Notre partenariat a été une véritable source de synergie et de complémentarité, et je suis reconnaissant pour les échanges constructifs, les idées partagées et les efforts conjoints pour mener à bien notre travail.

À mes parents bien-aimés, je dédie ces mots remplis d'amour et de reconnaissance, vous êtes ma source de force et d'inspiration. Votre dévouement, votre bienveillance et votre amour inconditionnel ont été les piliers de ma vie. Vous avez sacrifié tant de choses pour moi, et je serai éternellement reconnaissant, car vous êtes mon plus grand soutien et mon plus grand trésor. Je vous aime.

À mes grands-parents bien-aimés, et en particulier à mon cher grand-père AMRAH Mohand Rabah, un homme d'une bravoure exemplaire, tu es mon guide et mon modèle dans cette vie. Grand-mère, ta douceur et ta bienveillance m'inspirent chaque jour. Je suis béni d'avoir eu la chance d'avoir des grands-parents aussi merveilleux que vous.

Je n'oublie pas mes frères et sœurs ainsi que tous mes cousins, en particulier Bali (Walid), qui ont été là à chaque moment de ma vie. Leur soutien, leurs encouragements et leur volonté de partager leurs connaissances ont été d'une importance capitale.

Un immense merci à mes amis, en particulier Karim, Said, Khaled, Nabil, Sofiane, Lamine, Amine, qui ont été présents pour m'écouter, me soutenir et m'encourager tout au long de ce projet. Leur amitié et leur soutien moral ont été une source d'inspiration inestimable.

Enfin, je tiens à remercier toutes les personnes qui ont contribué de près ou de loin à notre réussite. Vos encouragements, vos conseils et votre aide ont joué un rôle crucial dans la réalisation de notre travail.

Merci du fond du cœur pour votre générosité, votre amour et votre soutien.

BRAHAM Zakaria

Table des matières

LISTE DES FIGURES.....	VII
LISTE DES TABLEAUX.....	IX
INTRODUCTION GENERALE.....	1
I. PRESENTATION DU PROJET.....	3
I.1 Introduction.....	3
I.2 Description du projet.....	3
I.2.1 Définitions.....	3
I.2.2 Problématique.....	5
I.2.3 Etude de l'existant.....	6
I.2.4 Solution proposée.....	9
I.2.5 Méthodologie de développement.....	10
I.2.6 Langage de modélisation unifié.....	14
I.2.7 Le design pattern Modèle-Vue-Contrôleur.....	15
I.3 Conclusion.....	17
II. SPRINT DE PLANIFICATION.....	18
II.1 Introduction.....	18
II.2 Répartition des rôles.....	18
II.3 Identification des acteurs et les cas d'utilisations.....	19
II.3.1 Identification des acteurs.....	19
II.3.2 Diagramme de cas d'utilisation global.....	19
II.4 Product Backlog.....	20
II.5 Décomposition des sprints.....	21
II.6 Charte graphique.....	22
II.7 Outils et langages utilisés.....	24
II.7.1 Niveau Conceptuel.....	24
II.7.2 Niveau Développement.....	25
II.8 Conclusion.....	27
III. SPRINT 1.....	28
III.1 Introduction.....	28
III.1.1 Diagramme de cas d'utilisation.....	28
III.1.2 Sprint Backlog.....	29
III.1.3 Décomposition du sprint 1.....	31
III.1.4 Présentations des diagrammes de séquence.....	31
III.1.5 Diagramme de classe.....	41
III.1.6 Vue des interfaces utilisateur.....	42
III.2 Conclusion.....	48
IV. SPRINT 2, 3.....	49
IV.1 Introduction.....	49
IV.2 Sprint 2.....	49
IV.2.1 Diagramme de cas d'utilisation Sprint 2.....	49
IV.2.2 Sprint Backlog.....	50
IV.2.3 Décomposition du sprint 2.....	52

IV.2.4	Présentation des diagrammes de séquence.....	52
IV.2.5	Diagramme de classe du sprint 2.....	54
IV.2.6	Vue des interfaces utilisateur.....	55
IV.3	Sprint 3.....	58
IV.3.1	Diagramme de cas d'utilisation Sprint 3.....	58
IV.3.2	Sprint Backlog.....	58
IV.3.3	Décomposition du sprint 3.....	59
IV.3.4	Présentation des diagrammes de séquence.....	60
IV.3.5	Diagramme de classe du sprint 3.....	61
IV.3.6	Vue des interfaces utilisateur des cas d'utilisations.....	62
IV.4	Conclusion.....	65
V.	SPRINT 4,5,6.....	66
V.1	Introduction.....	66
V.2	Sprint 4.....	66
V.2.1	Diagramme de cas d'utilisation Sprint 4.....	66
V.2.2	Sprint Backlog.....	67
V.2.3	Décomposition du sprint 4.....	68
V.2.4	Présentation des diagrammes de séquence.....	69
V.2.5	Diagramme de classe du sprint 4.....	72
V.2.6	Vue des interfaces utilisateur.....	73
V.3	Sprint 5.....	77
V.3.1	Diagramme de cas d'utilisation Sprint 5.....	77
V.3.2	Sprint Backlog.....	78
V.3.3	Décomposition du sprint 5.....	78
V.3.4	Présentation des diagrammes de séquence.....	79
V.3.5	Diagramme de classe du sprint 5.....	81
V.3.6	Vue des interfaces utilisateur.....	82
V.4	Sprint 6.....	83
V.4.1	Sprint Backlog.....	83
V.4.2	Décomposition du sprint 6.....	83
V.4.3	Présentation des interfaces utilisateur.....	83
V.5	Conclusion.....	87
	CONCLUSION GENERALE ET PERSPECTIVES.....	88
	BIBLIOGRAPHIE.....	89
	ANNEXES.....	90

Liste des figures

Figure I.1 : Cycle de développement de Scrum.....	14
Figure I.2 : Les composants du MVC	16
Figure I.3 : Le client et l'architecture MVC	16
Figure II.1 : Diagramme de cas d'utilisation globale.....	20
Figure II.2 : Planification des sprints	21
Figure II.3 : Logo de l'application	22
Figure II.4 : Signification du logo	22
Figure II.5 : La palette des couleurs et leurs significations	23
Figure II.6 : Aperçu de la typographie "Space Grotesk".....	24
Figure III.1 : Diagramme de cas d'utilisation du sprint 1.....	29
Figure III.2 : Décomposition du sprint 1.....	31
Figure III.3 : Diagramme de séquence de la fonctionnalité « Inscription »	33
Figure III.4 : Diagramme de séquence de la fonctionnalité « Authentification Administrateur »	35
Figure III.5 : Diagramme de séquence de la fonctionnalité « Authentification ».....	36
Figure III.6 : Diagramme de séquence de la fonctionnalité « Créer un article ».....	37
Figure III.7 : Diagramme de séquence de la fonctionnalité « Modifier/Supprimer un article».....	39
Figure III.8 : Diagramme de séquence du cas d'utilisation « Créer une demande »	40
Figure III.9 : Diagramme de séquence de la fonctionnalité «Refuser,Accepter,Annuler demande»	41
Figure III.10 : Diagramme de classe du sprint 1	42
Figure III.11 : Interface D'accueil	42
Figure III.12 : Interface d'inscription	43
Figure III.13 : Interface d'authentification.....	43
Figure III.14 : Interface de profil utilisateur	44
Figure III.15 : Interface Contact Administrateur.....	44
Figure III.16 : Interface Tous les articles (Objets).....	45
Figure III.17 : Interface Détail article	46
Figure III.18 : Interface « Mes demandes Envoyées »	46
Figure III.19 : Interface "Mes demandes Reçues".....	47
Figure III.20 : Interface de détails des demandes reçues.....	47
Figure III.21 : Interface de gestion des comptes par l'administrateur.....	48
Figure IV.1 : Diagramme de cas d'utilisation du sprint 2	50
Figure IV.2 : Décomposition du sprint 2.....	52
Figure IV.3 : Diagramme de séquence de la fonctionnalité « Mettre à niveau Utilisateur »	53
Figure IV.4 : Gestion de profil organisation	54
Figure IV.5 : Diagramme de classe du sprint 2.....	55
Figure IV.6 : Formulaire de Demande de mise à niveau coté Utilisateur	55
Figure IV.7 : Gestion des demandes mise à niveau coté Administrateur	56
Figure IV.8 : Espace et profile d'une organisation	57
Figure IV.9 : Interface Modifier le profil d'une organisation.....	57

Figure IV.10 : Diagramme de cas d'utilisation du sprint 3.....	58
Figure IV.11 : Décomposition du sprint 3.....	59
Figure IV.12 : Diagramme de séquence du cas d'utilisation « Inscription évènement ».....	60
Figure IV.13 : Diagramme de séquence du cas d'utilisation « Réservation ticket ».....	61
Figure IV.14 : Diagramme de classe du sprint 3.....	62
Figure IV.15 : Interface Inscription évènement	62
Figure IV.16 : Interface Réservation Ticket (événement gratuit).....	63
Figure IV.17 : Interface Réservation Ticket (événement Payant).....	63
Figure IV.18 : Interface Ajouter un événement Payant.....	64
Figure IV.19 : Interface détail d'évènement	64
Figure IV.20 : Interface Tous les événements	65
Figure V.1 : Diagramme de cas d'utilisation du sprint 4	67
Figure V.2 : Décomposition du sprint 4	69
Figure V.3 : Diagramme de séquence de la fonctionnalité « Contribuer aux cagnottes ».....	70
Figure V.4 : Diagramme de séquence de la fonctionnalité « Contact entre utilisateurs »	71
Figure V.5 : Diagramme de séquence de la fonctionnalité « Rechercher sur le site ».....	72
Figure V.6 : Diagramme de classe du sprint 4.....	73
Figure V.7 : Page Gestion de cagnottes.....	74
Figure V.8 : Interface de contribution aux cagnottes	74
Figure V.9 : Interface Vue profile d'un utilisateur par d'autres utilisateurs.....	75
Figure V.10 : Interfaces de contact entre utilisateurs « messagerie »	75
Figure V.11 : Interface de gestion d'avis	76
Figure V.12 : Interface de recherche.....	77
Figure V.13 : Diagramme de cas d'utilisation sprint 5.....	77
Figure V.14 : Décomposition du sprint 5	79
Figure V.15 : Diagramme de séquence de la fonctionnalité participer aux volontariats	80
Figure V.16 : Diagramme de séquence de la fonctionnalité « Demander un échange ».....	81
Figure V.17 : Diagramme de classe sprint 5.....	82
Figure V.18 : Interface demande d'un échange.....	82
Figure V.19 : Décomposition du sprint 6	83
Figure V.20 : Interface « Accueil & Inscription & Authentification »	84
Figure V.21 : Interfaces « Utilisateur Accueil & Articles & Details article »	84
Figure V.22 : Interfaces « Gestion de demandes Profile Organisation et profil utilisateur »	85
Figure V.23 : Interfaces « Gestion des Evènements, Annonces, Cagnottes »	85
Figure V.24 : Interfaces Contacte utilisateur (Messagerie)	86
Figure V.25 : Interface d'ajout « Article, Evènement »	86
Figure V.26 : Diagramme de classe globale.....	90
Figure V.27 : Schéma de la base de données.....	92

Liste des tableaux

Tableau I.1: Résultats de l'analyse de Nmedlek.com.....	6
Tableau I.2 : Résultats de l'analyse de Geev.com	7
Tableau I.3 : Résultat de l'analyse de GoFundMe.com.....	8
Tableau I.4 : Résultats de l'analyse de Volunteermatch.org	9
Tableau II.1 : Les membres de l'équipe Scrum pour notre projet.....	18
Tableau II.2 : Product Backlog.....	21
Tableau III.1 : Backlog du sprint 1	30
Tableau IV.1 : Backlog du sprint 2.....	51
Tableau IV.2 : Backlog du sprint 3.....	59
Tableau V.1 : Backlog du sprint 4	68
Tableau V.2 : Backlog du sprint 5.....	78
Tableau V.3 : Backlog du sprint 6.....	83

Introduction générale

La pandémie de COVID-19 a engendré des défis et des perturbations sans précédent à travers le monde, mais elle a également suscité une vague de solidarité et de mobilisation remarquable. En Algérie, différentes initiatives ont émergé, telles que la distribution de nourriture et de produits essentiels, le soutien aux personnels de santé, les campagnes de sensibilisation, les initiatives de solidarité, ainsi que l'aide apportée lors des incendies de forêts. Ces actions ont souligné la ténacité et l'esprit de communauté du peuple algérien.

Notre engagement dans diverses mobilisations organisées par nos villages pendant ces crises nous a incité à entreprendre ce projet. Ces expériences nous ont permis de constater les difficultés rencontrées et les lacunes des méthodes traditionnelles dans le domaine de la philanthropie¹, telles que la collecte de fonds, la complexité des processus de don, le manque d'informations fiables sur l'utilisation des fonds et le suivi limité de l'impact des aides.

Afin d'apporter notre aide et remédier à certaines difficultés recensées, nous allons concevoir et réaliser une application informatique dédiée à la philanthropie, avec pour objectif de favoriser une culture de solidarité continue en rassemblant les individus qui souhaitent aider les autres en leur offrant des opportunités concrètes.

Pour accomplir ce travail, nous allons structurer ce mémoire en cinq (05) chapitres :

Le premier chapitre, intitulé "Présentation du projet", mettra en lumière les objectifs et le fonctionnement détaillés de notre projet. Nous aborderons également les problèmes rencontrés par les organismes caritatifs, puis nous réaliserons une étude de l'existant à l'échelle locale et mondiale afin de mettre en évidence les limites des systèmes actuels et de justifier la nécessité d'une solution informatique adaptée. En outre, nous détaillerons la méthodologie de développement que nous adopterons, en mettant en

¹ La philanthropie : est acte de générosité désintéressée avec pour but l'amélioration de la vie d'autrui.

avant l'utilisation des design patterns MVC et du langage de modélisation UML qui joueront un rôle clé dans la conception et la mise en œuvre de notre application.

Le deuxième chapitre, intitulé "Planification des sprints", se concentrera sur la structuration de notre travail à travers la planification des sprints. Nous commencerons par identifier les besoins du système et déterminer les acteurs impliqués ainsi que les cas d'utilisation associés. Par la suite, nous présenterons les outils de développement que nous utiliserons pour mener à bien le projet. Enfin, nous aborderons la charte graphique qui guidera la conception visuelle de notre application.

Dans le troisième chapitre intitulé "Sprint 1", nous mettrons l'accent sur des fonctionnalités clés telles que l'inscription des utilisateurs et l'authentification. Ce sprint initial visera à établir les bases de notre application en assurant un accès sécurisé et en permettant aux utilisateurs de créer des comptes et de se connecter. Nous accorderons également une attention particulière à la gestion des articles, en développant les fonctionnalités nécessaires à leur création, modification et suppression.

Le quatrième chapitre, "Sprint 2 et 3", décrira les deux sprints suivants, qui seront dédiés à l'ajout de fonctionnalités essentielles répondant aux besoins des utilisateurs. Parmi ces fonctionnalités, nous inclurons la gestion de l'espace des organisations et la gestion des événements au sein de l'application.

Le cinquième chapitre, intitulé "Sprint 4, 5 et 6", sera consacré aux trois derniers sprints de développement de notre application. Cette phase finale du projet se concentrera sur l'ajout de fonctionnalités avancées et l'amélioration de l'expérience utilisateur, telles qu'un système de messagerie pour faciliter la communication entre les utilisateurs, la fonctionnalité d'échange d'objets et le développement d'une version mobile de l'application.

En conclusion, nous dresserons une synthèse générale des résultats obtenus et des perspectives futures pour notre application, mettant en évidence les avantages qu'elle apporte aux utilisateurs et les améliorations potentielles pour répondre à leurs besoins en constante évolution.

I. Présentation du projet

I.1 Introduction

La charité est une valeur fondamentale de la société humaine, incarnant notre volonté d'assister ceux dans le besoin. Au fil du temps, elle s'est développée pour englober diverses formes d'assistance, allant des besoins essentiels tels que la nourriture et le logement dans l'Antiquité, jusqu'aux collectes de fonds, au bénévolat et à l'engagement communautaire actuels.

Ce premier chapitre vise à fournir une vue d'ensemble sur notre projet, nous définirons certains termes couramment utilisés, nous présenterons la problématique et les obstacles rencontrés par les organisations caritatives, par la suite nous effectuerons une étude de l'existant portant sur les applications en place, aussi bien au niveau local qu'à l'échelle internationale, en outre, nous proposerons notre solution pour cette problématique spécifique.

Ensuite, nous aborderons la méthode agile Scrum adoptée pour notre projet. De plus, nous introduirons le langage de modélisation UML et le design pattern Modèle-Vue-Contrôleur employés lors de la réalisation de ce projet.

I.2 Description du projet

I.2.1 Définitions

Dans cette section, nous définissons et clarifions les termes fréquemment utilisés dans le contexte de la philanthropie et le bien social, notamment le troc, les dons, le crowdfunding et le bénévolat.

a. Le Troc (Echange)

Le troc est l'échange direct de biens sans utiliser de monnaie, il permet de se détacher de la valeur indicative de l'argent et de créer des échanges dans une économie alternative basée sur le lien et la confiance.

L'apparition d'Internet a considérablement transformé les méthodes de troc. De nos jours, il existe des plateformes spécialisées dans cette pratique, certaines étant dédiées

à des échanges spécifiques tels que le troc d'objets ou de services particuliers. Des sites de troc de livres, de vêtements, de cours et de services sont ainsi accessibles [1].

b. Le don

Le don exprime notre engagement envers les autres et se fait par compassion, solidarité ou pour une cause particulière. Descartes disait que « la générosité est la clé de toutes les vertus »². En donnant, nous acceptons de partager notre argent ou nos possessions avec les autres. Le don peut également être sous forme de temps. Dans tous les cas, il s'agit d'agir sans attendre de contrepartie en échange.

Avec l'avènement de la technologie, le don est devenu plus facile et plus accessible. Les plateformes en ligne et les réseaux sociaux ont permis de collecter des dons de manière plus efficace et de toucher un public plus large [2].

c. Le crowdfunding

Le financement participatif, également appelé crowdfunding, est une méthode de financement alternative qui a émergé au début des années 2000 grâce à la popularisation d'internet. Contrairement aux circuits traditionnels de financement tels que les banques, le crowdfunding sollicite des contributions financières auprès d'un grand nombre d'internautes pour financer des projets de différentes natures tels que des projets culturels, artistiques, entrepreneuriaux ou caritatives, via des plateformes dédiées à cet effet [3].

d. Les associations

La législation Algérienne définit une association comme : « *L'association est le regroupement de personnes physiques et/ou de personnes morales sur une base contractuelle à durée déterminée ou à durée indéterminée. Ces personnes mettent en commun, bénévolement et dans un but non lucratif, leurs connaissances et leurs moyens pour promouvoir et encourager les activités dans les domaines notamment, professionnel, social, scientifique, religieux, éducatif, culturel, sportif, environnemental, caritatif et humanitaire* »³.

Selon les statistiques annoncées par le ministère de l'Intérieur et des Collectivités Locales sur son site officiel⁴, l'Algérie compte 4304 associations de solidarité et de

² Art. 161. « Comment la générosité peut être acquise », **Passions de l'âme de DESCARTES**, Paris, Gallimard, Bibliothèque de la Pléiade, 1953, p. 695-796

³ Article 2 de Loi n° 12-06 du 18 Safar 1433 correspondant au 12 janvier 2012 relative aux associations. <https://www.joradp.dz/ftp/jo-francais/2012/f2012002.pdf>

⁴ <https://www.interieur.gov.dz/images/pdf/Thematiquedesassociations.pdf>

bienfaisance. Ces organisations travaillent sans relâche pour fournir une assistance aux personnes dans le besoin à travers diverses activités⁵ :

- Distribution de nourriture et de produits de première nécessité.
- Soutien aux enfants orphelins et défavorisés.
- Accompagnement des personnes en situation de handicap.
- Sensibilisation et éducation.

Cependant, elles sont confrontées à des défis qui impactent leurs activités et compromettent efficacité dans la réalisation de leur mission.

I.2.2 Problématique

Selon une étude publiée par El Hadi Makboul⁶ (expert international et ex-DG du Centre National d'Etudes et d'Analyses pour la Population) dans le journal Liberté, la pauvreté en Algérie connaît une montée inquiétante. Les données recueillies au cours des dernières décennies révèlent une augmentation significative du taux de prévalence de la précarité, passant de 3,6% en 1988 à 8,1% en 1995, puis à 12% en 2000, 18% en 2012 et 19% en 2020. Ces chiffres continuent de s'aggraver en raison de l'effondrement économique, de la crise sanitaire et l'augmentation du chômage. Face à cette précarité grandissante, plusieurs associations se sont engagées pour aider les populations vulnérables en Algérie. Cependant, ces organisations caritatives et humanitaires rencontrent elles-mêmes des problèmes majeurs qui entravent leurs actions.

- L'accès limité au financement constitue un autre problème préoccupant. Sans un soutien financier adéquat, il devient ardu de fournir les ressources et le soutien nécessaires aux personnes dans le besoin.
- La gestion des données est également complexe pour ces organisations. Avec des ressources et une technologie limitée, il est difficile de collecter et d'analyser les données de manière efficace, rendant ainsi difficile la mesure de l'impact de leurs programmes et la prise de décisions éclairées sur l'allocation des ressources.
- La mobilisation des bénévoles peut représenter un défi. Bien que les bénévoles soient essentiels au succès de ces organisations, leur gestion peut prendre beaucoup de temps et peut être aussi exigeante.

⁵ <https://sharek-algerie.com/repertoire-des-associations-de-jeunesse-algeriennes/>

⁶ <https://www.liberte-algerie.com/contribution/montee-inquietante-de-la-pauvrete-en-algerie-358198>(publié le 03 Mai 2021)

- L'organisation et la gestion efficace des événements constituent un autre enjeu crucial. La planification, la coordination et la logistique liées à la tenue d'événements nécessitent une attention particulière pour garantir leur réussite. Avant de proposer une solution, il est primordial de réaliser une étude de l'existant afin de mettre en œuvre une solution adaptée à nos objectifs.

I.2.3 Etude de l'existant

L'essor des applications en ligne a révolutionné la manière dont nous pouvons contribuer à notre communauté et apporter un soutien aux causes qui nous tiennent à cœur. Ces applications offrent des opportunités de bénévolat, de collecte de fonds et dons, permettant ainsi à un plus grand nombre de personnes de s'impliquer et de faire une différence dans le monde qui les entoure.

Cette étude a pour but d'explorer et d'évaluer les applications les plus utilisées et les plus connues en Algérie et à l'étranger. Nous analyserons trois types d'applications :

- **Applications de dons d'objets**

Une plateforme de don d'objets permet aux individus de donner leurs objets inutilisés à d'autres personnes dans le besoin. Cela favorise la redistribution des biens, réduit le gaspillage et contribue à la durabilité environnementale. Un exemple notable dans cette catégorie en Algérie est nmedlek.com⁷, la seule application algérienne qui facilite le don d'objets et l'échange de services entre particuliers et organisations caritatives. Le tableau [I.1] représente un résumé de notre analyse.

Fonctionnalités	- Publication d'annonces décrivant les dons ou les services offerts par l'utilisateur. - Spécification des catégories d'articles ou de services, inclusion d'images et détails sur les conditions d'échange. -Disponibilité en 3 langues : Arabe, Français et Anglais.
Modèle de fonctionnement	Les utilisateurs peuvent proposer des dons ou des services qu'ils sont prêts à offrir, et d'autres utilisateurs intéressés peuvent les contacter pour organiser les détails de l'échange.
Portée géographique	Algérie.
Interface utilisateur	Conviviale et intuitive.

Tableau I.1: Résultats de l'analyse de Nmedlek.com

⁷ <https://nmedlek.com/fr>

Toutefois, il est intéressant de prendre en compte des applications étrangères qui ont connu un succès significatif dans d'autres pays tels que Geev⁸, une application en ligne basée en France qui permet aux utilisateurs de donner les objets dont ils n'ont plus besoin. Le tableau [I.2] résume les résultats de notre analyse.

Fonctionnalités	<ul style="list-style-type: none"> - Publication d'annonces décrivant les objets que les utilisateurs souhaitent donner gratuitement. -Les utilisateurs peuvent inclure des images, des descriptions et des informations sur l'emplacement de l'objet -Fonctionnalité de partage sur les réseaux sociaux. -Disponibilité sous forme d'application mobile. -Disponibilité en 2 langues : Français et Anglais
Modèle de fonctionnement	Les utilisateurs peuvent proposer des objets dont ils n'ont plus besoin et les offrir gratuitement à d'autres utilisateurs intéressés.
Portée géographique	France
Interface utilisateur	Conviviale et intuitive.

Tableau I.2 : Résultats de l'analyse de Geev.com

En plus des plateformes de dons d'objets, il existe également des plateformes de crowdfunding qui se concentrent sur le financement participatif de projets spécifiques.

• Applications de Crowdfunding

Ces applications offrent aux organisations à but non lucratif la possibilité de lancer des campagnes de financement et de solliciter des contributions de la part de nombreux donateurs, en Algérie, ce type d'applications fonctionnelles est rare et les tentatives antérieures n'ont pas connu de continuité à citer « chriky.com »⁹ et « Ninvesti »¹⁰.

Cependant, une des plateformes internationales bien connues dans ce domaine est GoFundMe. C'est une application de financement participatif en ligne qui permet aux individus, aux organisations et aux causes de collecter des fonds auprès de donateurs du monde entier. Le tableau [I.3] résume les résultats de notre analyse.

⁸ <https://www.geev.com/fr>

⁹ <https://www.chriky.com/>

¹⁰ SANSRI Sara, CHEURFA Hakima, Les plateformes du Crowdfunding, alternatives numériques pour les PME en temps du COVID-19. Cas : Plateforme Algérienne NINVESTI, Revue des Sciences Economiques, de Gestion et Sciences Commerciales Volume : 13/ N° : 02 (2020), p 294- 308

Fonctionnalités	<ul style="list-style-type: none"> - Création de campagnes de collecte de fonds. - Partage sur les réseaux sociaux. - Donations avec plusieurs options de paiement. - Outils de suivi et de gestion des campagnes.
Modèle de fonctionnement	Les utilisateurs créent des campagnes de collecte de fonds et reçoivent des dons de la part de la communauté en ligne. La plateforme prélève des frais de traitement des transactions et de services pour couvrir les coûts opérationnels. Les frais varient en fonction de la région et du type de campagne, et ils ont généralement déduits des fonds collectés.
Portée géographique	Internationale (disponible dans 19 pays)
Interface utilisateur	Conviviale et intuitive.

Tableau I.3 : Résultat de l'analyse de GoFundMe.com

D'un autre côté, les plateformes de volontariat jouent un rôle essentiel dans la facilitation du recrutement et de la gestion des bénévoles au sein des organisations à but non lucratif.

- **Applications de volontariat**

Elles offrent des fonctionnalités telles que la création d'opportunités de bénévolat, la gestion des horaires, la communication avec les bénévoles et le suivi des heures de bénévolat.

Malheureusement, en Algérie ces applications sont également peu répandues et les initiatives antérieures, à citer « Warda »¹¹, n'ont pas connu une continuité significative en raison de problèmes tels que le manque de financement, la sensibilisation limitée, les défis techniques et le manque d'engagement des utilisateurs. Les adaptations locales et le soutien institutionnel peuvent être cruciaux pour améliorer leur succès à long terme. Malgré ces défis, l'étude de modèles réussis peut servir d'inspiration pour développer des solutions adaptées à la réalité locale.

VolunteerMatch¹², une plateforme en ligne qui facilite la mise en relation entre les individus et les organisations à but non lucratif à la recherche de volontaires est un exemple dans cette catégorie, grâce à sa vaste base de données de bénévoles et ses outils de qualification des bénévoles. Le tableau [I.4] résume les résultats de notre analyse.

¹¹ <https://www.esi.dz/warda-dz-un-projet-ambitieux-realise-par-5-etudiants/>

¹² <https://www.volunteermatch.org/>

Fonctionnalités	- Recherche d'opportunités de volontariat. - Profils détaillés des organisations. - Inscriptions et gestion de comptes. - Évaluations et commentaires.
Modèle de fonctionnement	Les organisations à but non lucratif créent des profils et publient des offres de bénévolat, détaillant les tâches et les compétences requises. Les bénévoles peuvent rechercher et filtrer les opportunités selon leurs préférences. Une fois qu'un bénévole trouve une opportunité qui lui convient, il postule via la plateforme. Les organisations examinent les candidatures et coordonnent les détails du bénévolat avec les volontaires potentiels.
Portée géographique	États-Unis
Interface utilisateur	Conviviale et intuitive.

Tableau I.4 : Résultats de l'analyse de Volunteermatch.org

En examinant les solutions existantes, nous avons constaté qu'elles sont insuffisantes pour répondre à certains problèmes spécifiques, tels que le crowdfunding, qui n'est pas pris en considération. De plus, certaines applications traitant ces problèmes sont destinées à d'autres pays et ne sont pas adaptées à l'environnement algérien.

Face à ces lacunes, nous allons proposer une nouvelle application qui comblera ces manques. Pour nous démarquer efficacement, il est primordial d'intégrer des fonctionnalités uniques et d'apporter des améliorations supplémentaires.

I.2.4 Solution proposée

Notre projet vise à relever les défis auxquels sont confrontées les institutions caritatives en développant une application informatique qui offre des fonctionnalités telles que :

- La création de campagnes de crowdfunding : Les institutions caritatives pourront créer des campagnes de financement participatif pour collecter des fonds et soutenir leurs initiatives.
- L'échange et don d'objets physiques : Nous offrirons une section dédiée à l'échange et au don d'objets physiques, permettant aux utilisateurs de donner et recevoir des objets encore en bonne état.
- Une communication améliorée : L'application facilitera la communication entre les organisations caritatives et les donateurs en intégrant un système de messagerie en temps réel, favorisant ainsi une interaction fluide et transparente.

- La gestion des bénévoles : Les institutions caritatives pourront gérer les bénévoles et coordonner leurs actions au sein de l'application.
- La promotion des événements caritatifs : L'application permettra de promouvoir les événements caritatifs et d'encourager la participation de la communauté.

Afin de garantir le succès de n'importe quel projet, il est essentiel de sélectionner soigneusement la méthodologie de développement appropriée. Ce choix revêt une importance capitale, car il a un impact significatif sur plusieurs aspects clés tels que les coûts, les délais, la satisfaction du client, la qualité du logiciel et la réduction des dépenses liées aux échecs. Il est donc primordial de choisir une approche de gestion qui correspond parfaitement aux objectifs et aux exigences spécifiques afin d'assurer sa réussite [4].

I.2.5 Méthodologie de développement

Les approches classiques de gestion de projet, telles que la méthode en cascade ou en V, sont basées sur des activités séquentielles et une planification rigide. Cependant, ces méthodologies se sont révélées trop rigides pour faire face aux besoins changeants du marché. Dans les années 1990, les méthodes agiles ont émergé comme des approches moins prédictives et plus souples, permettant aux organisations de s'adapter plus facilement aux contraintes du marché [5].

La méthodologie agile peut être vue comme un ensemble de techniques, de valeurs et de principes conçus pour guider et améliorer la manière dont les équipes de développement logiciel travaillent ensemble pour livrer de nouveaux produits ou services, elle décompose un projet en plusieurs phases et vise à fournir de la valeur aux clients plus rapidement et avec moins de complications [6].

En effet, le mouvement Agile a engendré de nombreux frameworks, chacun avec ses propres principes et pratiques uniques, notamment le cadre Scrum.

Selon le rapport "The 16th State of Agile"¹³ publié par digital.ai, Scrum se démarque comme l'un des Framework agile le plus adopté dans la gestion de projets. Cette conclusion repose sur une enquête approfondie menée auprès d'un large éventail d'organisations et de praticiens agiles.

De la même manière qu'une équipe de rugby (d'où le nom de cette méthode) se prépare pour un match important, Scrum encourage les équipes à apprendre par l'expérience,

¹³ <https://digital.ai/resource-center/analyst-reports/state-of-agile-report>

à s'auto-organiser tout en travaillant sur un problème et à réfléchir à leurs succès et échecs pour s'améliorer continuellement [7].

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

a. Sprint

La première unité élémentaire du processus Scrum est un sprint, autrement appelé itération ou chronologie. Essentiellement, cela fonctionne comme un effort dans les délais, où la durée est préalablement fixée et convenue, et elle se situe généralement entre une et quatre semaines. Le plus souvent, un sprint dure deux semaines [8].

b. Artefacts

Les artefacts [9] sont des informations utilisées par les équipes Scrum et les parties prenantes pour documenter le produit, les actions entreprises pour le développer et le progrès du projet.

Les principaux artefacts Agiles Scrum sont le carnet de produit (Product Backlog), le carnet de sprint (sprint backlog) et les incréments.

- a. **Product backlog** : C'est un artefact dynamique qui est constamment mis à jour en fonction des nouvelles informations disponibles, il contient une liste d'éléments tels que des fonctionnalités, des améliorations, des corrections de bugs, des tâches ou des exigences nécessaires pour développer un produit.
- b. **Sprint Backlog** : C'est une liste des tâches spécifiques sélectionnées à partir du Product backlog pour être réalisées lors d'un sprint donné. Il est créé par l'équipe de développement dans le but de planifier et de suivre le travail à effectuer pendant la durée du sprint.
- c. **Incrément** : Un incrément de produit est un livrable qui est créé en intégrant tous les éléments terminés du Backlog produit lors d'un sprint. L'ensemble de l'équipe Scrum est responsable de créer un incrément de valeur et d'utilité pour chaque sprint

Ces artefacts offrent des métadonnées sur les performances de chaque sprint, favorisant ainsi les attributs fondamentaux de transparence, d'inspection et d'adaptation propres à Scrum.

c. Membres d'une équipe Scrum (Rôles)

Une équipe Scrum [8] est considérée comme une pierre angulaire, composée d'un Scrum Master, d'un Product Owner et d'une équipe de développement. Cette équipe travaille de manière unie, se concentrant sur un objectif commun. Les équipes Scrum sont caractérisées par leur pluridisciplinarité, avec des membres possédant toutes les compétences requises pour créer de la valeur à chaque Sprint. Elles sont également autonomes, ce qui signifie qu'elles décident en interne qui fait quoi, quand et comment.

- a. **Product Owner** : le Product Owner (PO) est la personne responsable de la vision du produit, de la détermination des caractéristiques et des fonctionnalités, et de l'assurance que l'équipe de développement comprend ce qui est nécessaire.
- b. **Scrum Master** : Le Scrum Master (SM) est chargé de veiller à ce que Scrum soit mis en œuvre correctement.
- c. **Équipe de développement** : l'équipe de développement est responsable de la conception, du développement, des tests et de la livraison des incréments de produit.

d. Cérémonies

Les cérémonies Scrum sont un ensemble d'événements qui se déroulent au cours de chaque sprint. Les quatre cérémonies Scrum sont la planification du sprint, la réunion quotidienne, la revue du sprint et la rétrospective du sprint.

Ces cérémonies sont conçues pour aider l'équipe Scrum à planifier, exécuter et revoir son travail de manière itérative et collaborative.

- a. **Planification du sprint** : avant chaque sprint, le Product Owner présente les principaux éléments du Product backlog lors d'une réunion de planification du sprint. L'équipe détermine le travail qu'elle peut accomplir pendant le sprint et transforme le backlog de produit en backlog de sprint.
- b. **Réunion quotidienne** : la réunion quotidienne est une réunion debout de 15 minutes, qui a lieu à la même heure et au même endroit chaque jour pendant le sprint. Au cours de cette réunion, chaque membre de l'équipe parle de ce qu'il a fait la veille et sur quoi il va travailler aujourd'hui et les éventuels obstacles rencontrés.

- c. **Revue de Sprint** : à la fin de chaque sprint, l'équipe présente le travail qu'elle a accompli sous la forme d'une démonstration directe plutôt que d'une présentation.
- d. **Rétrospective du sprint** : à la fin de chaque sprint, l'équipe réfléchit à la manière dont Scrum fonctionne et discute des changements qui doivent être apportés au prochain sprint.

e. Cycle de développement

Le cycle de vie Scrum [figure I.1]¹⁴ est basé sur des itérations (Sprints), avant le début de chaque sprint, lors d'une réunion de planification appelée « Sprint planning meeting », l'équipe sélectionne les exigences les plus prioritaires du « Product backlog » qui seront développées, testées et livrées au client. Ces exigences forment le « Sprint backlog ».

Pendant le sprint, une réunion quotidienne d'avancement, connue sous le nom de « Daily Scrum », est organisée avec tous les membres de l'équipe. Cette réunion permet de suivre l'évolution du travail et de s'assurer que les objectifs du sprint seront atteints.

À la fin du sprint, une démonstration des développements réalisés est présentée au client lors de la réunion de revue du sprint, appelée « Sprint Review Meeting ». Ensuite, une réunion de rétrospective de fin d'itération est organisée pour évaluer qualitativement le fonctionnement de l'équipe et permettre un bilan sur la façon d'améliorer les processus pour les itérations suivantes [5].

¹⁴ jordanjob.me/blog/scrums-diagram/

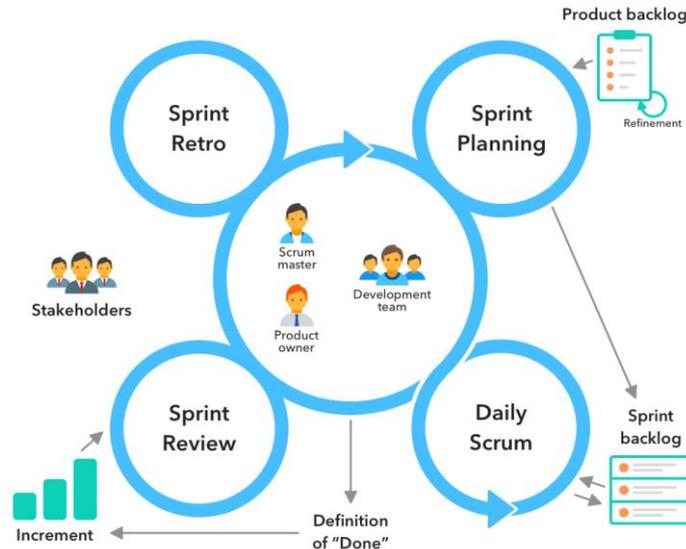


Figure I.1 : Cycle de développement de Scrum

La capacité de Scrum à gérer des projets logiciels dynamiques grâce à des retours continus et une adaptabilité, ainsi que son accent sur les équipes collaboratives et la prise de décisions en temps réel, en font un choix parfaitement adapté à la nature en constante évolution des technologies web et mobile.

Grâce à cette approche, nous pourrions ajuster les fonctionnalités et les exigences de notre projet au fur et à mesure de leur développement, en nous adaptant rapidement aux changements et aux nouvelles idées.

Cependant, afin de garantir une compréhension globale claire de notre système en constante évolution, il est essentiel d'utiliser la modélisation de système. Cette approche repose généralement sur l'utilisation de notations graphiques, qui sont aujourd'hui presque toujours basées sur les notations du langage de modélisation unifié (UML).

I.2.6 Langage de modélisation unifié

Le langage de modélisation unifié (UML) [10] est un langage utilisé pour spécifier, visualiser, construire et documenter les éléments des systèmes logiciels, ainsi que pour la modélisation des processus métier et d'autres systèmes non logiciels.

UML comprend un ensemble de notations graphiques et de diagrammes qui représentent différents aspects d'un système logiciel.

Parmi les principaux diagrammes d'UML [11] et souvent utilisés figurent le diagramme de classes, le diagramme de cas d'utilisation et le diagramme de séquence.

Le diagramme de classes : Le diagramme de classe est largement employé en programmation orientée objet (POO) comme principal outil de conception. Il offre une représentation visuelle des classes, de leurs attributs et méthodes, ainsi que des relations qui les lient (telles que l'héritage, l'implémentation d'interfaces ou les associations)

Le diagramme de cas d'utilisation : Le diagramme de cas d'utilisation représente les interactions entre les utilisateurs et les différents cas d'utilisation du système. Il permet d'identifier les utilisateurs potentiels du système et offre une vue claire et simplifiée des fonctionnalités essentielles. Il est particulièrement utile lors des échanges avec un client, car il simplifie la communication en mettant l'accent sur les interactions utilisateur-système, sans se perdre dans les détails techniques.

Le diagramme de séquence : Ce type de diagramme met en évidence les échanges de messages entre les objets, illustrant ainsi le flux d'exécution dans un scénario donné. Il offre une vue chronologique des interactions et facilite la compréhension des échanges entre les différents éléments du système.

L'UML a été initialement conçu pour la description des systèmes orientés objet. Toutefois, lors de la phase de conception architecturale, il est courant de vouloir décrire les systèmes à un niveau d'abstraction supérieur. Les classes d'objets, étant plus proches de l'implémentation, peuvent s'avérer moins appropriées pour une description architecturale précise. Il devient donc nécessaire d'utiliser des techniques et des notations plus adaptées pour représenter les aspects plus généraux et conceptuels de l'architecture d'un système. C'est dans ce contexte que le design pattern Modèle-Vue-Contrôleur (MVC) entre en jeu.

I.2.7 Le design pattern Modèle-Vue-Contrôleur

Le Modèle-Vue-Contrôleur (MVC) [13] est un modèle de conception largement utilisé qui permet de structurer une application en trois composants clés : le modèle, la vue et le contrôleur. Chacun de ces composants joue un rôle spécifique dans le développement de l'application, offrant ainsi plusieurs avantages en termes de gestion et de maintenabilité du code.

Le pattern MVC permet de bien organiser son code source. Il permet de savoir quels fichiers créer, mais surtout à définir leur rôle. Le but de MVC est justement de séparer la logique du code en trois parties que l'on retrouve dans des fichiers distincts.

1. **Modèle** (logique métier) : Il représente la couche de données de l'application et gère la logique qui lui est associée. Le modèle contient les informations nécessaires, telles que les données des utilisateurs.
2. **Vue** (l'affichage) : Elle est responsable de l'interface utilisateur et de l'affichage des données à l'écran. La vue présente les informations de manière claire et attrayante pour les utilisateurs, tout en gérant les interactions avec ces derniers.
3. **Contrôleur** (échanges) : Il assure la gestion des actions de l'utilisateur et fait le lien entre le modèle et la vue. Le contrôleur reçoit les commandes de l'utilisateur, les traite et les achemine vers les parties appropriées de l'application.

La figure [I.2]¹⁵ représente les composants du MVC



Figure I.2 : Les composants du MVC

L'architecture MVC (représentée par la figure [I.3]¹⁶) offre une séparation claire entre la logique métier, l'interface utilisateur et la gestion des entrées. Elle permet également un contrôle complet sur le code HTML et les URL, facilitant ainsi la conception de l'architecture d'une application web.

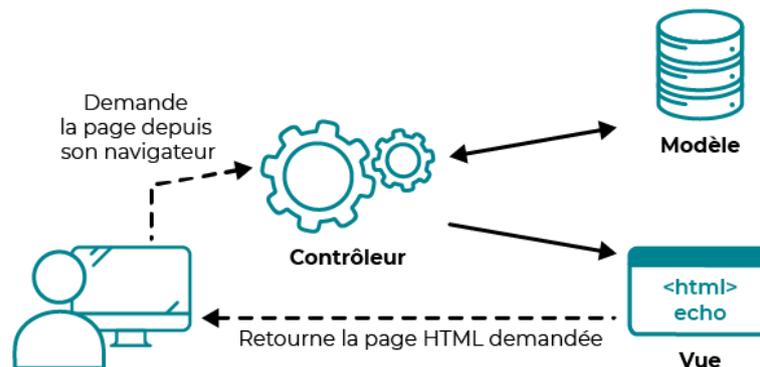


Figure I.3 : Le client et l'architecture MVC

¹⁵ <https://openclassrooms.com/fr/courses/4670706-adoptez-une-architecture-mvc-en-php/7847928-decouvrez-comment-fonctionne-une-architecture-mvc>

¹⁶ <https://openclassrooms.com/fr/courses/4670706-adoptez-une-architecture-mvc-en-php/7847928-decouvrez-comment-fonctionne-une-architecture-mvc>

De plus, la structure MVC favorise le développement axé sur les tests, car chaque composant peut être conçu avec des interfaces claires et testé indépendamment à l'aide d'objets simulés. Il convient de noter que l'architecture MVC est largement adoptée pour développer des interfaces utilisateur, tant pour les applications de bureau que pour les applications web, dans différents langages de programmation. Son utilisation permet de simplifier le code, de favoriser la maintenabilité et de faciliter l'évolution de l'application.

I.3 Conclusion

En conclusion, ce chapitre a permis de jeter les bases de notre projet en présentant une vue d'ensemble de son contexte. Nous avons défini les notions liées au projet et examiné les différentes activités des organisations caritatives en Algérie ainsi que les défis auxquels elles font face, ensuite, nous avons effectué une étude comparative des applications existantes en Algérie qu'à l'étranger, afin de mieux comprendre les fonctionnalités attendues et d'identifier les objectifs spécifiques de notre système après nous avons proposé notre solution qui consiste en la conception et réalisation d'une application informatique dédiée aux actions philanthropiques. Nous avons également exploré le processus de développement Scrum, souligné l'importance de la modélisation UML et examiné l'architecture MVC. Il s'agit maintenant de passer à la phase de planification, qui vise à établir une vision claire et à définir les étapes concrètes pour la réalisation de notre projet.

II. Sprint de planification

II.1 Introduction

Dans ce chapitre, nous allons explorer en détail le fonctionnement de notre équipe Scrum et clarifier les rôles et les responsabilités de chaque membre. Cela nous permettra d'assurer une meilleure collaboration et une répartition efficace des tâches.

Nous identifierons les acteurs de notre système, en analysant les parties prenantes impliquées. Ensuite, nous aborderons l'identification des cas d'utilisation en regroupant les fonctionnalités selon leur importance dans le Product backlog. Nous organiserons notre travail en utilisant la méthode Scrum, en établissant des objectifs clairs et un calendrier réaliste nos livraisons. Enfin, nous définirons la charte graphique de notre application et discuterons des outils et langages utilisés pour la modélisation et l'implémentation.

II.2 Répartition des rôles

Le succès d'un projet Scrum repose en grande partie sur la collaboration efficace des membres de l'équipe. Chaque individu doit comprendre pleinement son rôle et ses responsabilités afin de garantir un processus sans faille. Pour faciliter cette compréhension, il est utile de créer un tableau [II.1] de répartition des rôles, qui indique clairement qui est responsable de quoi.

Rôles Scrum	La personne affectée
Product Owner	L'association qui agit en tant que Product Owner a choisi de rester anonyme. Cette décision est respectée dans le cadre du projet
Scrum Master 1	Monsieur GHILES HAMZA
Scrum Master 2	Monsieur ACHROUFENE Achour
Equipe de développement	Mlle HAMIS Sonia et Mr BRAHAM Zakaria

Tableau II.1 : Les membres de l'équipe Scrum pour notre projet.

II.3 Identification des acteurs et les cas d'utilisations

Cette étape permet de définir les rôles et les interactions nécessaires pour le développement de l'application.

II.3.1 Identification des acteurs

Un acteur [14] représente un rôle joué par une entité externe qui interagit directement avec le système étudié. L'acteur peut accéder au système et modifier son état en envoyant et en recevant des messages qui peuvent inclure des données.

Dans le cadre de notre projet, nous avons identifié quatre acteurs qui interagissent directement avec le système étudié. Ces acteurs peuvent être des visiteurs ou des utilisateurs lorsqu'ils sont inscrits, des dirigeants des organisations caritatives, ou des administrateurs.

- **Visiteur** : c'est une personne qui accède à l'application sans être inscrite. Il peut naviguer sur le site pour découvrir ses fonctionnalités.
- **Utilisateur** : c'est une personne qui a créé un compte sur l'application. Il peut accéder à toutes les fonctionnalités disponibles pour les utilisateurs enregistrés.
- **Organisateur** : c'est une personne qui dirige une organisation caritative.
- **Administrateur** : c'est la personne chargée de la maintenance de l'application et de la gestion des comptes des utilisateurs. Il veille au bon fonctionnement du serveur de données et à sa sécurité.

II.3.2 Diagramme de cas d'utilisation global

Les cas d'utilisation recensés sont présentés dans le diagramme de cas d'utilisation global de la figure [II.1], qui sera détaillé par la suite.

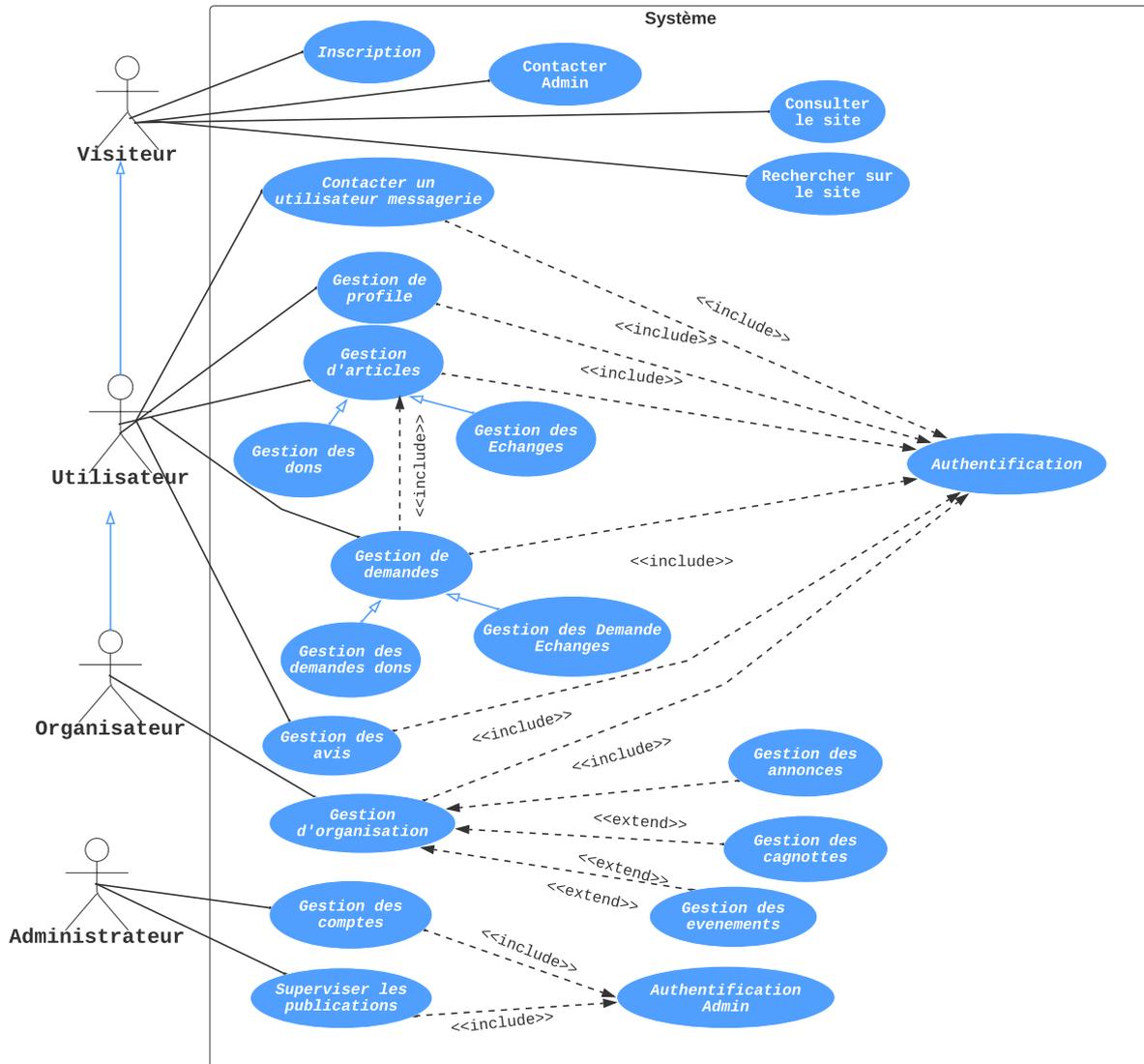


Figure II.1 : Diagramme de cas d'utilisation globale

II.4 Product Backlog

Le "Product Owner" évalue chaque fonctionnalité en termes de priorité pour les organiser dans le Product backlog donné par le tableau [II.2].

Sprint	Items	En tant que	Je veux ...	Priorité
Sprint 1	Inscription	Visiteur	S'inscrire	1
	S'authentifier	Utilisateur, Organisateur et Administrateur	S'authentifier	1
	Consulter le site	Visiteur, Utilisateur et Organisateur	Consulter le contenu de site	1

	Contactez l'administrateur	Visiteur, Utilisateur et Organisateur	Contactez l'administrateur	1
	Gestion des comptes	Administrateur	Ajouter supprimer récupérer modifier	2
	Gestion d'articles	Utilisateur Organisateur	Ajouter supprimer récupérer modifier	2
	Gestion de demandes	Utilisateur Organisateur	Ajouter annuler accepter, refuser finaliser	2
	Gestion de profile	Utilisateur Organisateur	Récupérer modifier	3
Sprint 2	Gestion d'organisation	Organisateur	Ajouter récupérer modifier	3
	Gestion des annonces	Organisateur	Ajouter supprimer récupérer modifier	3
Sprint 3	Gestion des événements	Organisateur	Ajouter supprimer récupérer modifier	4
	Superviser les publications	Administrateur	Superviser les évènements	4
Sprint 4	Gestion des cagnottes	Organisateur	Ajouter supprimer récupérer modifier	5
	Gestion des avis	Utilisateur	Ajouter supprimer récupérer modifier	6
	Rechercher sur le site	Visiteur	Rechercher les articles les évènements cagnottes annonces	7
	Contact entre utilisateurs messagerie	Utilisateur	Envoyer des messages et contactes en temps réel avec les utilisateurs	7
Sprint 5	Gestion des échanges et des dons	Utilisateur	Ajouter supprimer récupérer modifier	8
	Gestion des demandes des échanges et des dons	Utilisateur	Ajouter annuler accepter, refuser finaliser	8
Sprint 6	Réaliser une version mobile de l'application	Organisateur Utilisateur	Une version mobile de l'application	9

Tableau II.2 : Product Backlog

II.5 Décomposition des sprints

La figure [II.2] représente une décomposition de notre projet selon une estimation pour chaque sprint.

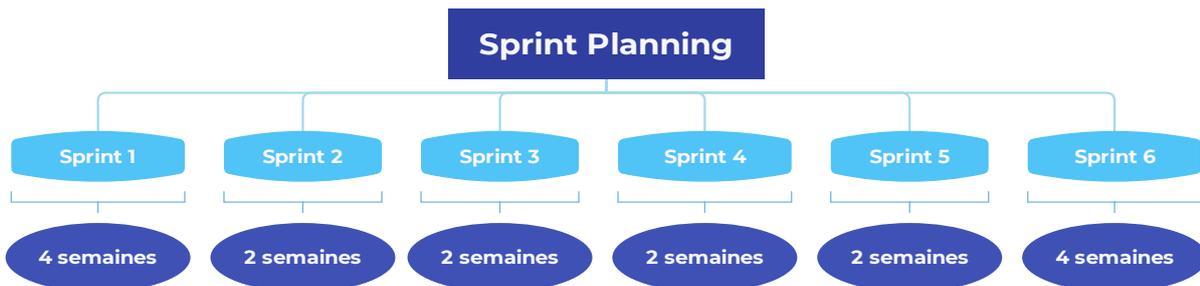


Figure II.2 : Planification des sprints

II.6 Charte graphique

La charte graphique¹⁷ joue un rôle essentiel dans la création de l'identité visuelle d'une application. Elle englobe tous les éléments graphiques qui composent cette identité visuelle, tels que les couleurs, les logos, les polices de caractères, les symboles, et bien d'autres éléments. Nous passons maintenant à la présentation de la charte graphique de notre projet.

a. Logotype

Un logo est une représentation graphique ou symbolique qui représente une entreprise, une marque, une organisation ou un produit. Il s'agit d'un élément visuel distinctif qui permet d'identifier et de différencier une entité d'une autre. Le logo conçu pour notre application est représenté par la figure [II.3].



Figure II.3 : Logo de l'application

b. Signification du logo

La figure [II.4] représente une illustration explicative du concept de notre logo, mettant en évidence les éléments visuels et symboliques qui reflètent l'identité et les valeurs de notre projet.

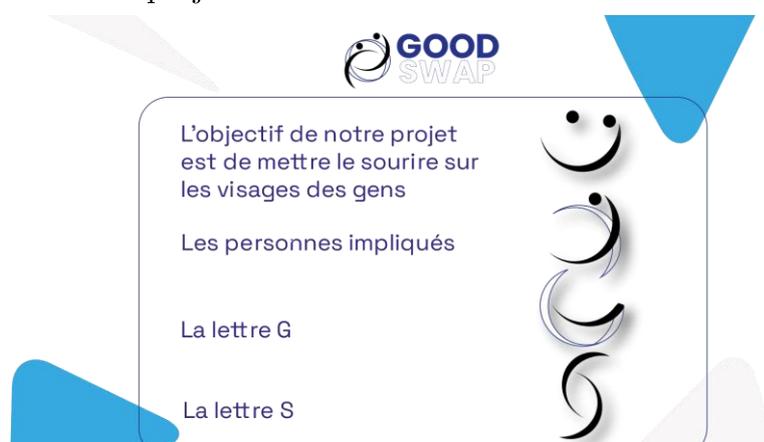


Figure II.4 : Signification du logo

¹⁷ <https://sokeo.fr/charte-graphique-application-web/>

c. **Palette des couleurs :** La palette de couleurs définit l'identité visuelle de l'application. Elle est basée sur le nuancier Pantone, attribuant à chaque couleur un nom et un numéro. Les valeurs numériques RVB (Rouge-Vert-Bleu) et HEX (Hexadécimal), ainsi que les couleurs d'impression CMJN (Cyan-Magenta-Jaune-Noir). La figure [II.5] représente notre palette de couleurs soigneusement sélectionnée, qui incarne l'essence de notre projet et crée une expérience visuelle cohérente et engageante.

Palette des couleurs et leurs significations

La spiritualité, la créativité et l'expression individuelle	
	#44337A
Évoque la chaleur, l'énergie et la vitalité.	
	#f8b038
Symbolise le douceur, personnalisation et le partage	
	#851a50
Apporte une neutralité, un équilibre et une sobriété.	
	#DFE4E8
La profondeur, la stabilité et la confiance.	
	#161853
Évoque la clarté, la fraîcheur et la confiance.	
	#2590EB

Figure II.5 : La palette des couleurs et leurs significations

d. **Typographie :** La typographie¹⁸ est l'art d'arranger les lettres et le texte de manière lisible et attrayante. Elle joue un rôle essentiel dans le design d'interface, renforçant la reconnaissance d'une marque, influençant les décisions des utilisateurs et captant leur attention. Pour notre projet, nous avons choisi d'intégrer Space Grotesk¹⁹ (représentée par la figure [II.6]) en tant que typographie privilégiée car elle correspond parfaitement à notre vision pour notre application. Sa lisibilité, son esthétique moderne et sa polyvalence en font un choix idéal pour créer une expérience utilisateur engageante.

¹⁸ <https://careerfoundry.com/en/blog/ui-design/beginners-guide-to-typography/>

¹⁹ <https://fonts.google.com/specimen/Space+Grotesk>

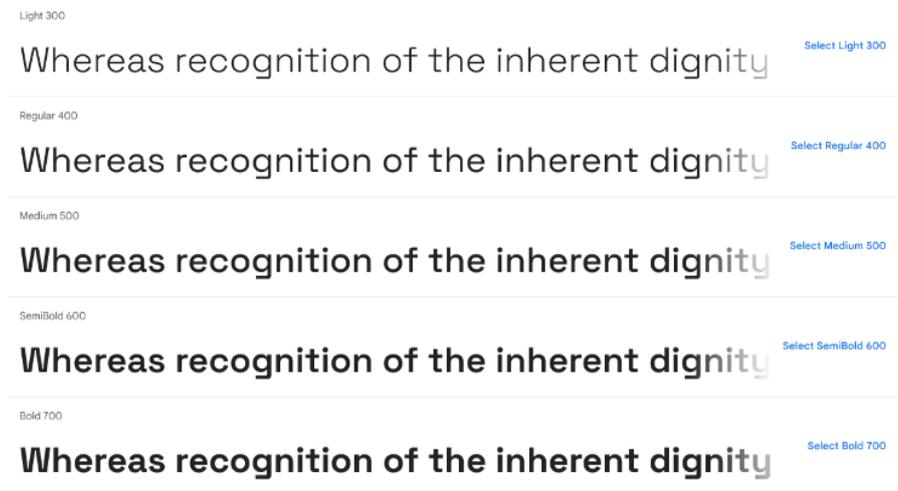


Figure II.6 : Aperçu de la typographie "Space Grotesk".

II.7 Outils et langages utilisés

Dans cette section nous allons présenter les différentes technologies utilisées lors de la réalisation de notre projet et on va les présenter en deux niveaux principaux : conceptuel et développement.

II.7.1 Niveau Conceptuel

- a. **Clickup** ²⁰ : Nous avons utilisé Clickup comme outil de gestion de projet personnalisable pour organiser et suivre les tâches liées au développement de notre application. Il nous a permis de collaborer entre nous, d'assigner des responsabilités et de respecter les délais
- b. **LucidChart** ²¹ : Cet outil nous a permis de créer des diagrammes de cas d'utilisations, séquence, classe et diagrammes d'activité pour mieux modéliser les fonctionnalités de l'application.
- c. **Figma** ²² : Nous avons opté pour Figma qui est un outil de conception basé sur le Cloud afin créer des prototypes interactifs et de partager facilement les designs de l'interface utilisateur.

²⁰ <https://clickup.com/blog/asana-alternatives/#1-1-clickup>

²¹ <https://www.lucidchart.com/pages/>

²² <https://www.figma.com/>

- d. **Illustrator**²³ : Nous avons intégré Illustrator dans notre processus de conception de l'identité visuelle de notre projet. Cet outil de conception de graphiques nous a permis de créer des graphiques vectoriels de haute qualité.
- e. **XMind**²⁴ : Nous avons utilisé XMind comme logiciel de cartographie mentale pour organiser et analyser visuellement les informations liées à notre projet. Cet outil polyvalent nous a permis de créer des cartes mentales, d'importer et d'exporter des cartes dans différents formats.

II.7.2 Niveau Développement

Le stack MERN (MongoDB, Express, React, Node.js) [15] est un ensemble de technologies utilisé dans le développement d'applications web. Il combine MongoDB, une base de données NoSQL offrant une flexibilité dans le stockage des données, avec Express, un framework web basé sur Node.js qui facilite la création de l'infrastructure serveur. React, développé par Facebook, permet quant à lui la construction d'interfaces utilisateur réactives, tandis que Node.js exécute du code JavaScript côté serveur.

En utilisant le stack MERN, les développeurs peuvent créer des applications web modernes, performantes et évolutives. Ce stack favorise particulièrement le développement d'applications basées sur des Single Page Applications (SPA), offrant une expérience utilisateur fluide et réactive. Nous allons détailler chaque technologie et expliquer son intégration dans notre application.

a. Développement Front-End

- ✓ **React.Js**²⁵ : Pour le développement de l'interface utilisateur, nous avons utilisé React.js en tant que bibliothèque JavaScript. Grâce à sa flexibilité et à sa capacité à créer des composants réutilisables, nous avons pu construire une interface utilisateur dynamique et réactive pour notre application.
- ✓ **TailwindCss**²⁶ : Pour la gestion des styles et de la mise en page de notre application nous avons intégré TailwindCss. Ce framework CSS personnalisable nous a permis de créer rapidement et facilement des interfaces web élégantes, en utilisant ses classes prédéfinies et ses utilitaires.

²³ https://www.adobe.com/mena_en/products/illustrator.html

²⁴ <https://xmind.app/>

²⁵ <https://react.dev/>

²⁶ <https://tailwindcss.com/docs/installation/framework-guides>

- ✓ **React Native**²⁷ : React Native est un framework open-source qui nous a permis de créer des applications mobiles pour iOS et Android en utilisant JavaScript et la bibliothèque React.

b. Développement Back-End

- ✓ **Node.js**²⁸ : Pour la partie serveur de notre application web, nous avons utilisé Node.js comme environnement d'exécution côté serveur. Sa performance et sa scalabilité nous ont aidés à développer les fonctionnalités serveur nécessaires pour répondre aux demandes des utilisateurs.
- ✓ **Express.js**²⁹ : Express.js : Nous avons choisi Express.js comme framework d'application web pour Node.js. Ses fonctionnalités, telles que les fonctions middleware et le routage, nous ont facilité la création d'API et le traitement des requêtes et réponses HTTP.
- ✓ **MongoDB**³⁰ : Comme système de gestion de base de données, nous avons utilisé MongoDB pour stocker et gérer les données de notre application web. Sa flexibilité et ses performances nous ont permis de manipuler efficacement les données non structurées de notre application.
- ✓ **Mongoose**³¹ : Mongoose est une bibliothèque ODM (Object Data Modeling) pour Node.js et MongoDB qui permet de travailler de manière plus organisée avec les données en créant des schémas et en validant les données. Elle simplifie le travail avec MongoDB et fournit des fonctionnalités qui facilitent le travail avec les données.
- ✓ **Cloudinary**³² : Cloudinary est une solution logicielle basée sur le modèle Software-as-a-Service (SaaS). Elle est principalement utilisée pour gérer tous les actifs multimédias présents dans le cloud. Les actifs peuvent être de tout type, qu'il s'agisse de ressources pour des applications web ou mobiles. Cloudinary propose une solution complète pour tous les besoins en matière de médias, notamment l'importation, l'administration, la diffusion optimisée, le stockage et la manipulation. Tout cela est géré sur les serveurs de Cloudinary, qui sont automatiquement dimensionnés pour faire face à des charges élevées soudaines et à des pics de trafic fréquents.

²⁷ <https://reactnative.dev/>

²⁸ <https://nodejs.org/en/docs>

²⁹ <https://expressjs.com/en/5x/api.html>

³⁰ <https://www.mongodb.com/docs/>

³¹ <https://mongoosejs.com/docs/api/mongoose.html>

³² <https://cloudinary.com/>

- ✓ **Socket.io**³³: Est une bibliothèque JavaScript qui permet de créer des applications web asynchrones (applications en temps réel) qui obtiennent et envoient des données depuis et vers le serveur sans avoir à l'appeler explicitement. Par exemple, des applications de chat en temps réel ou des applications de notifications comme Twitter.

c. Environnement de développement

- ✓ **Visual Code**³⁴ : Visual Studio Code est un éditeur de code source léger mais puissant qui s'exécute sur votre ordinateur de bureau et est disponible pour Windows, macOS et Linux. Il offre une prise en charge intégrée de JavaScript, TypeScript et Node.js, et dispose d'un riche écosystème d'extensions pour d'autres langages et environnements d'exécution (tels que C++, C#, Java, Python, PHP, Go, .NET).

II.8 Conclusion

Au cours de ce chapitre, nous avons procédé à la planification de notre projet en définissant les rôles de l'équipe, en identifiant les acteurs et en établissant les cas d'utilisation qui constituent notre Product Backlog. Nous avons ensuite découpé notre travail en plusieurs sprints afin de répondre aux besoins des utilisateurs de manière itérative. Enfin, nous avons présenté l'identité visuelle de notre application et décrit les outils que nous utiliserons tout au long du processus de développement.

Dans le prochain chapitre, nous aborderons le sprint 1, où nous commencerons à concrétiser notre application en mettant en œuvre les fonctionnalités prioritaires du Product Backlog.

³³ <https://socket.io/docs/v4/>

³⁴ <https://code.visualstudio.com/docs>

III. Sprint 1

III.1 Introduction

Le présent chapitre vise à détailler le premier lancement du projet, qui comprend le premier sprint d'une durée de quatre semaines, qui est composé d'un backlog spécifique et d'un ensemble de cas d'utilisation qui représentent les fonctionnalités du système. De plus, nous présenterons des diagrammes de séquence et le diagramme de classe correspondants à ce sprint, ce qui nous permettra de mieux visualiser le fonctionnement et la structure de notre application, après on montrera quelques interfaces du résultat obtenu dans ce sprint.

III.1.1 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation de la figure [III.1] décrit d'une façon globale les cas d'utilisations composant les items du sprint 1

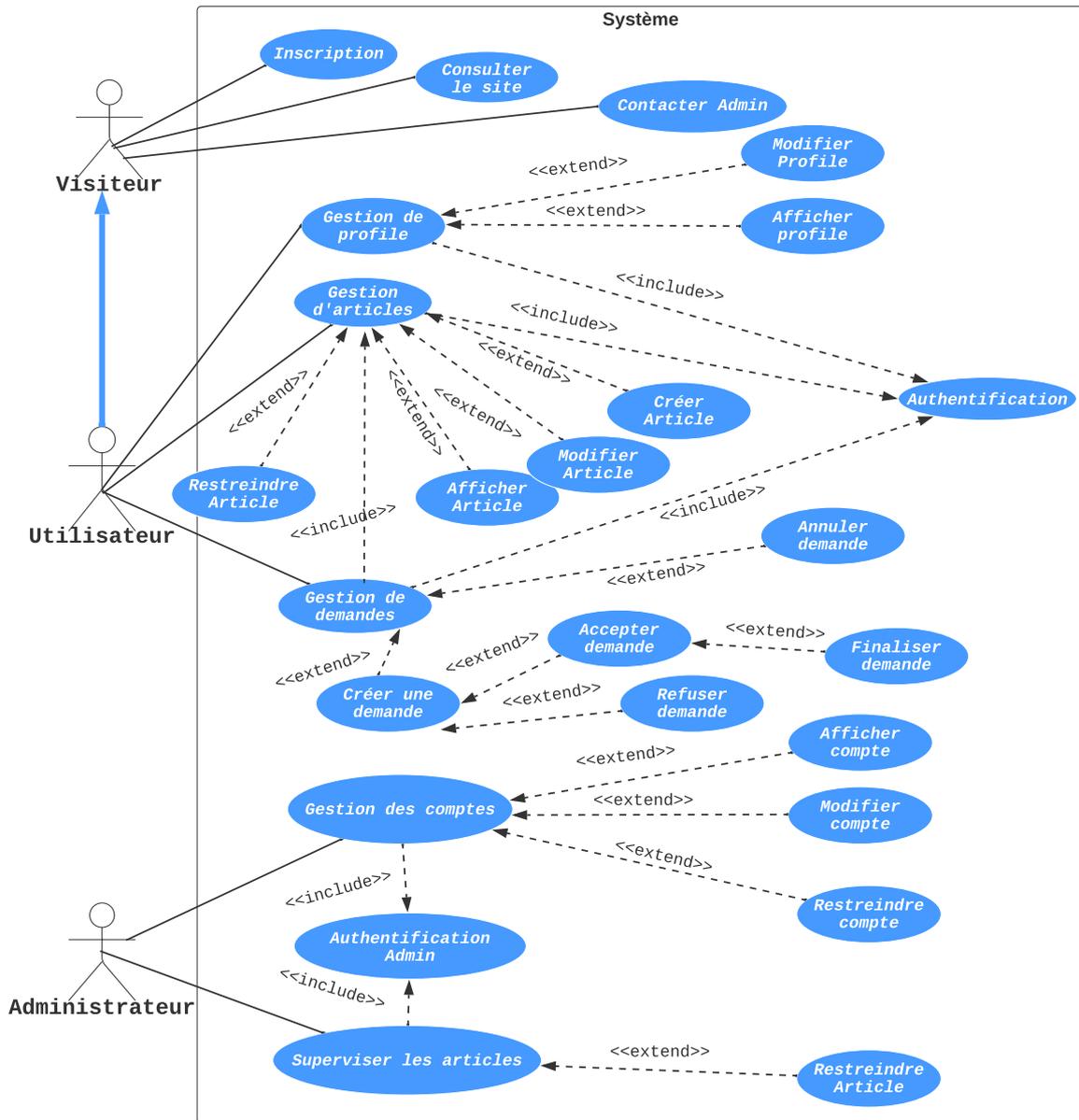


Figure III.1 : Diagramme de cas d'utilisation du sprint 1

III.1.2 Sprint Backlog

Tout d’abord on va présenter le backlog du sprint 1 qui contient l’ensemble des tâches à réaliser et leurs estimations qui figurent dans le tableau [III.1] :

Fonctionnalité	Tache	Responsable	Estimation d'efforts (par heure)	Priorité	Statut
Inscription	Créer le formulaire d'inscription	Hamis	6	Moyenne	A faire
	Mettre en place la validation des données d'inscription	Hamis	4	Haute	A faire
	Implémenter la logique d'inscription côté serveur	Braham	6	Moyenne	A faire

S'authentifier	Concevoir la page de connexion pour les utilisateurs	Hamis	4	Moyenne	A faire
	Implémenter la logique d'authentification côté serveur	Braham	6	Moyenne	A faire
Consulter le site	Concevoir la page principale du site	Hamis	6	Moyenne	A faire
	Mettre en place la navigation entre les différentes pages du site	Braham	6	Moyenne	A faire
Contactez l'admin	Concevoir la page de contact pour les utilisateurs	Hamis	8	Haute	A faire
	Mettre en place la logique d'envoi de message à l'administrateur	Braham	6	Moyenne	A faire
Gestion Profile	Concevoir la page d'affichage d'un profil	Hamis	6	Moyenne	A faire
	Implémenter la fonctionnalité de modification des profils	Braham	10	Haute	A faire
Gestion d'articles	Créer le formulaire de création d'article et mettre à jour	Hamis	6	Moyenne	A faire
	Concevoir la page d'affichage des détails d'article	Hamis	6	Moyenne	A faire
	Mettre en place la logique côté serveur	Braham	15	Haute	A faire
Gestion de demandes	Créer le formulaire de création de demande	Hamis	6	Moyenne	A faire
	Concevoir la page d'affichage des détails de demande	Hamis	8	Moyenne	A faire
	Mettre en place la logique côté serveur	Braham	6	Moyenne	A faire
Gestion Comptes	Créer le formulaire de modifier de compte	Hamis	8	Haute	A faire
	Mettre en place la logique de modification de compte	Braham	6	Moyenne	A faire
	Concevoir la page d'affichage des détails de compte	Hamis	6	Moyenne	A faire
Superviser Articles	Mettre en place la fonctionnalité de restriction d'article	Braham	10	Haute	A faire
	Implémenter la fonctionnalité de suppression d'article	Braham	6	Moyenne	A faire

Tableau III.1 : Backlog du sprint 1

III.1.3 Décomposition du sprint 1

Tout au long de cette période, nous traitons les fonctionnalités de ce sprint pour produire un incrément potentiellement livrable. La figure [III.2] représente la décomposition du sprint 1 en termes de temps.

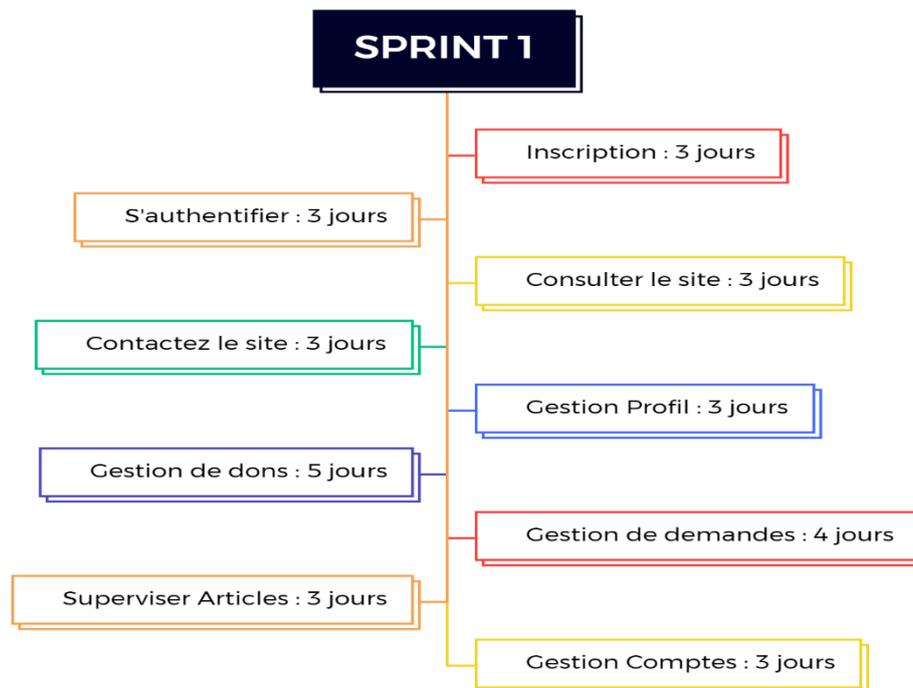


Figure III.2 : Décomposition du sprint 1

III.1.4 Présentations des diagrammes de séquence

Nous allons décrire d'une manière détaillée les fonctionnalités :

- « Inscription »
- « Authentification »
- « Gestion d'articles »
- « Gestions des demandes ».

a. Fonctionnalité « Inscription »

Lorsqu'un visiteur arrive sur la page d'inscription, il est invité à remplir ses informations personnelles telles que son nom, prénom, adresse e-mail et mot de passe. Une fois qu'il a soumis le formulaire, les données saisies sont vérifiées.

Si certaines informations sont incorrectes ou si des champs obligatoires sont laissés vides, le visiteur est redirigé vers la page d'inscription avec un message d'erreur lui

demandant de corriger les erreurs. Il lui est alors demandé de ressaisir les informations incorrectes.

Si toutes les informations saisies sont correctes, le système procède à une vérification supplémentaire. Il vérifie si l'adresse e-mail fournie par le visiteur n'est pas déjà enregistrée dans la base de données. Si l'adresse e-mail existe déjà, le visiteur est informé que cette adresse est déjà utilisée et est invité à en choisir une autre.

Si l'adresse e-mail n'existe pas déjà, le système génère un code OTP (One-Time Password) unique et l'envoie au visiteur via l'adresse e-mail qu'il a fournie. Le visiteur reçoit alors ce code OTP dans sa boîte de réception.

Le visiteur est invité à saisir ce code OTP sur la page d'inscription pour vérification. S'il saisit le code correctement, son inscription est considérée comme réussie et son compte est activé. Il reçoit alors une confirmation d'inscription et peut accéder à son compte nouvellement créé.

Cependant, si le visiteur saisit un code OTP incorrect, le système affiche un message d'erreur et lui demande de le ressaisir. Cette étape vise à s'assurer que le visiteur est bien en possession du code de vérification envoyé à son adresse e-mail.

La figure [III.3] représente le diagramme de séquence de la fonctionnalité « Inscription »

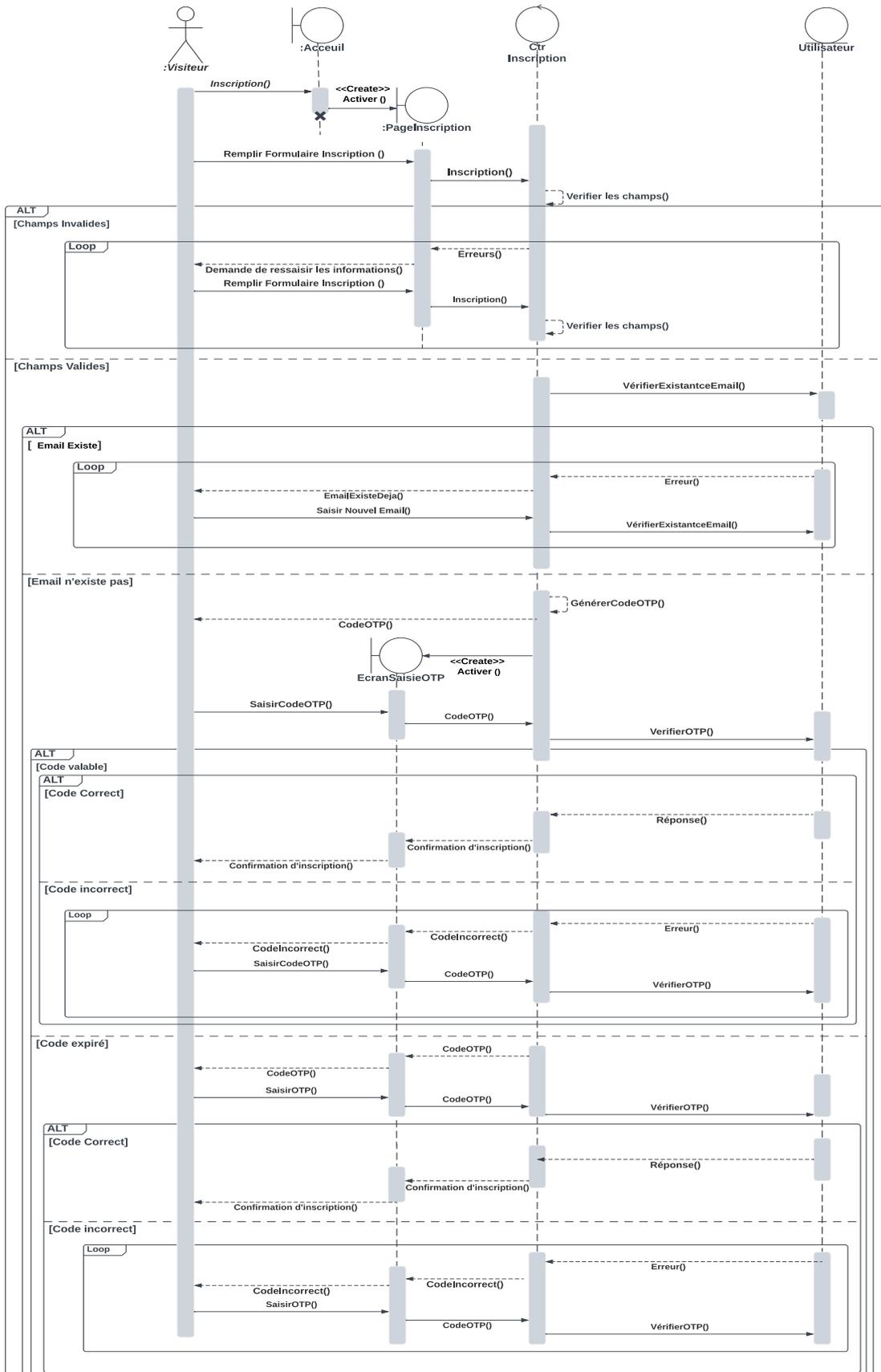


Figure III.3 : Diagramme de séquence de la fonctionnalité « Inscription »

b. Fonctionnalité « Authentification Administrateur »

Lorsqu'un administrateur souhaite se connecter, il accède à la page de connexion réservée aux administrateurs. Il saisit son identifiant (adresse e-mail) ainsi que son mot de passe dans les champs correspondants du formulaire de connexion.

Une fois que l'administrateur a soumis le formulaire, le système vérifie les informations de connexion fournies. Si les informations saisies sont incorrectes, le système affiche un message d'erreur indiquant que l'authentification a échoué et demande à l'administrateur de les saisir à nouveau.

Si les informations de connexion sont correctes, le système génère un code OTP qui est envoyé à l'administrateur via e-mail. Sur la page de connexion, l'administrateur est invité à saisir le code OTP qu'il a reçu.

Si le code OTP saisi correspond exactement au code OTP généré par le système, l'authentification de l'administrateur est considérée comme réussie. L'administrateur est alors redirigé vers le tableau de bord de l'administration et le système enregistre l'appareil et les coordonnées de la session pour se souvenir de l'admin.

Cependant, si l'administrateur saisit un code OTP incorrect, le système affiche un message d'erreur et lui demande de le saisir à nouveau. Cette mesure de sécurité empêche toute tentative d'usurpation d'identité par une personne non autorisée.

La figure [III.4] représente le diagramme de séquence de la fonctionnalité « Authentification Administrateur »

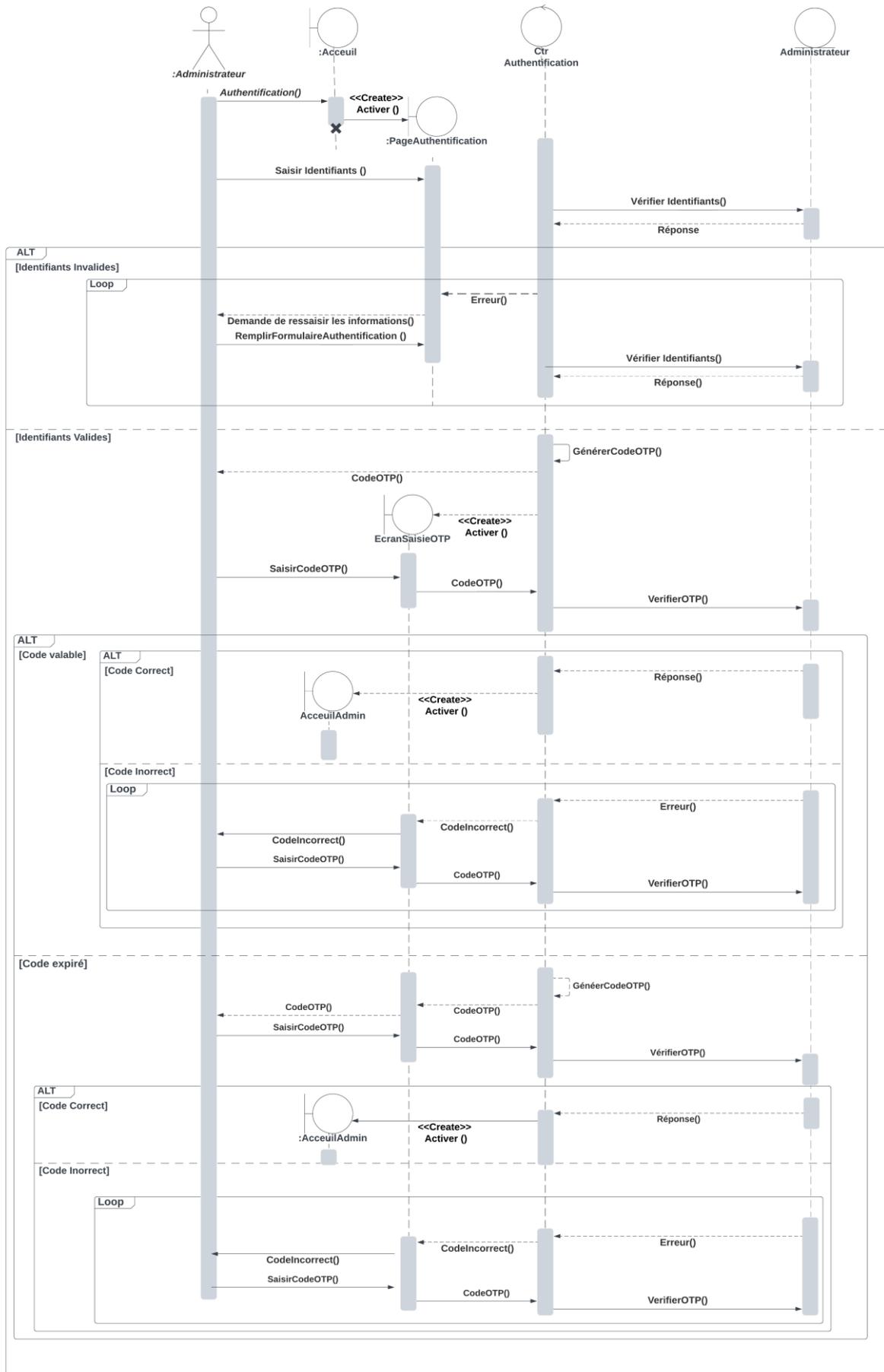


Figure III.4 : Diagramme de séquence de la fonctionnalité « Authentication Administateur »

c. Fonctionnalité « Authentification Utilisateur »

Lorsqu'un utilisateur souhaite accéder à son compte, il se rend sur la page de connexion dédiée aux utilisateurs. Sur cette page, il saisit son identifiant ainsi que son mot de passe dans les champs appropriés du formulaire de connexion.

Une fois que l'utilisateur a soumis le formulaire, le système procède à une vérification des informations de connexion fournies. Si les informations saisies sont incorrectes, le système affiche un message d'erreur indiquant que l'authentification a échoué et invite l'utilisateur à les saisir à nouveau, si les informations de connexion sont correctes, l'utilisateur est authentifié avec succès. Il est alors redirigé vers la page d'accueil de son espace. La figure [III.5] représente le diagramme de séquence du cas d'utilisation « Authentification »

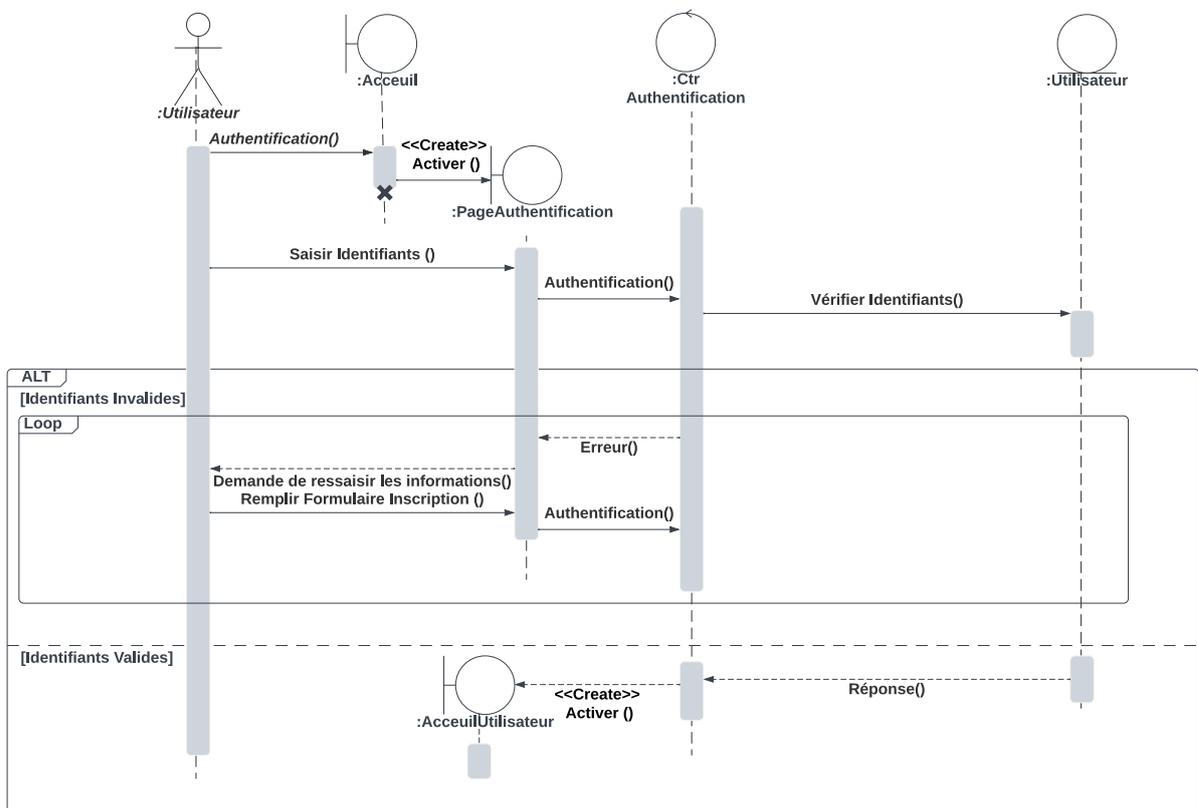


Figure III.5 : Diagramme de séquence de la fonctionnalité « Authentification »

d. Fonctionnalités de « Gestion d'articles »

- **Fonctionnalité « Créer un article »**

Lorsqu'un utilisateur souhaite créer un nouvel article, il accède à la page dédiée à la création d'articles. Sur cette page, il trouve un formulaire comprenant différents

champs à remplir pour créer son article. L'utilisateur commence à remplir le formulaire et le soumet. Si toutes les informations sont correctes, l'article est enregistré dans la base de données et publié sur l'application. L'utilisateur reçoit alors une notification ou un message confirmant la publication réussie de son article. En cas d'erreurs ou d'informations manquantes, le système affiche un message d'erreur indiquant les corrections à apporter. L'utilisateur est invité à corriger les erreurs et à soumettre à nouveau le formulaire. La figure [III.6] représente le diagramme de séquence de la fonctionnalité « créer un article »

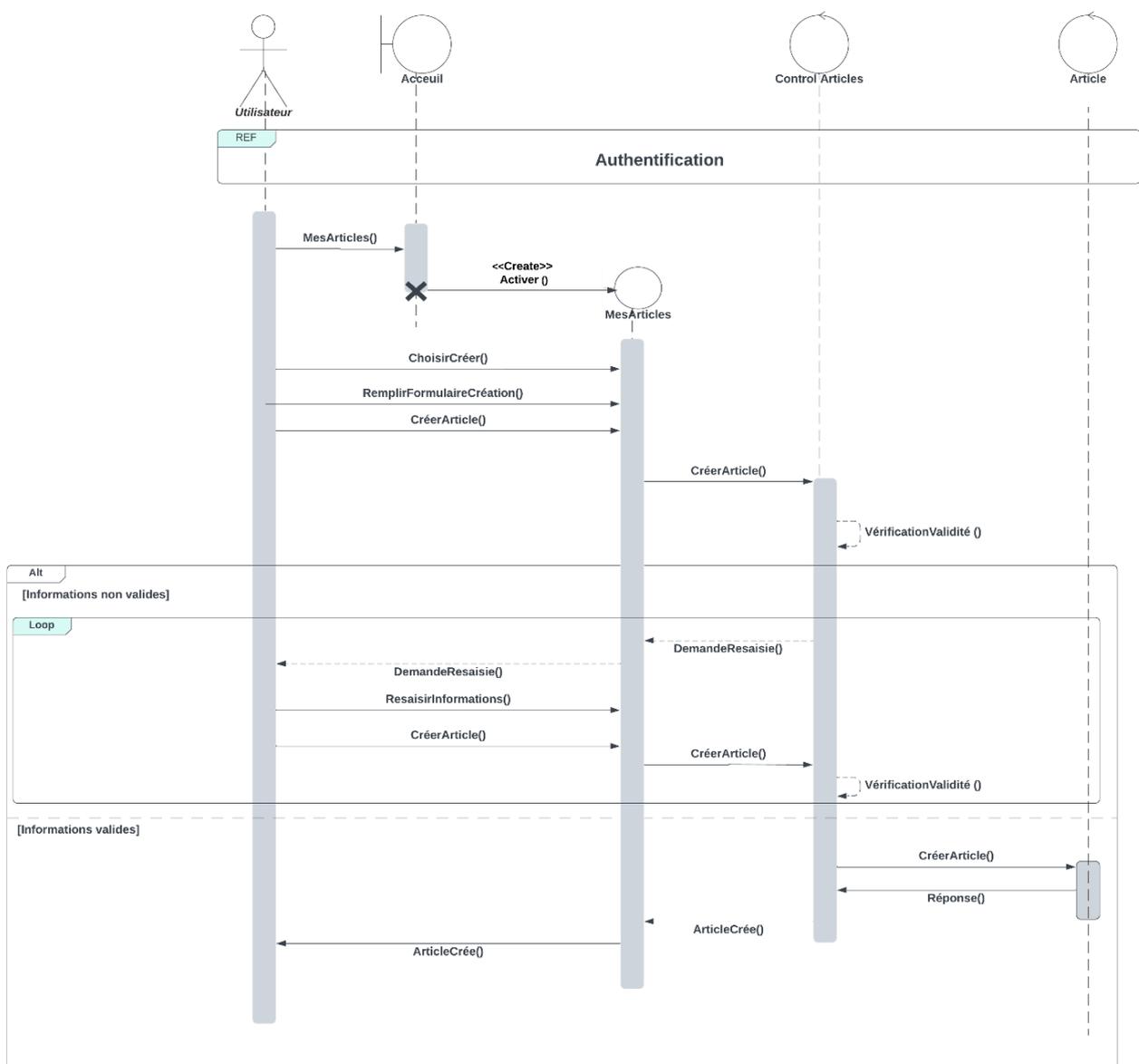


Figure III.6 : Diagramme de séquence de la fonctionnalité « Créer un article »

- **Fonctionnalité Modification d'un article**

L'utilisateur accède à sa liste d'articles, il identifie l'article qu'il souhaite modifier et sélectionne l'option de modification associée à cet article. Le système affiche un formulaire de modification.

Une fois les modifications apportées, l'utilisateur soumet le formulaire de modification. Le système procède à une vérification des données saisies pour s'assurer de leur validité. Si toutes les données sont valides, l'article est mis à jour dans la base de données avec les nouvelles informations. L'utilisateur reçoit une notification ou un message confirmant que les modifications de l'article ont été enregistrées avec succès. En cas d'erreurs de saisie ou de données non valides, le système affiche un message d'erreur spécifiant les corrections à apporter. L'utilisateur est invité à corriger les informations incorrectes et à soumettre à nouveau le formulaire de modification.

- **Fonctionnalité Suppression d'un article**

L'utilisateur normal accède à sa liste d'articles, il identifie l'article qu'il souhaite supprimer et sélectionne l'option de suppression associée à cet article. Le système affiche une fenêtre de confirmation pour demander la confirmation de la suppression. L'utilisateur doit confirmer sa décision en cliquant sur un bouton de confirmation. Une fois la suppression confirmée, le système désactive l'article et il ne sera plus affiché. L'utilisateur reçoit une notification ou un message confirmant que l'article a été

supprimé avec succès. La figure [III.7] représente le diagramme de séquence de la fonctionnalité « Modifier et supprimer un article »

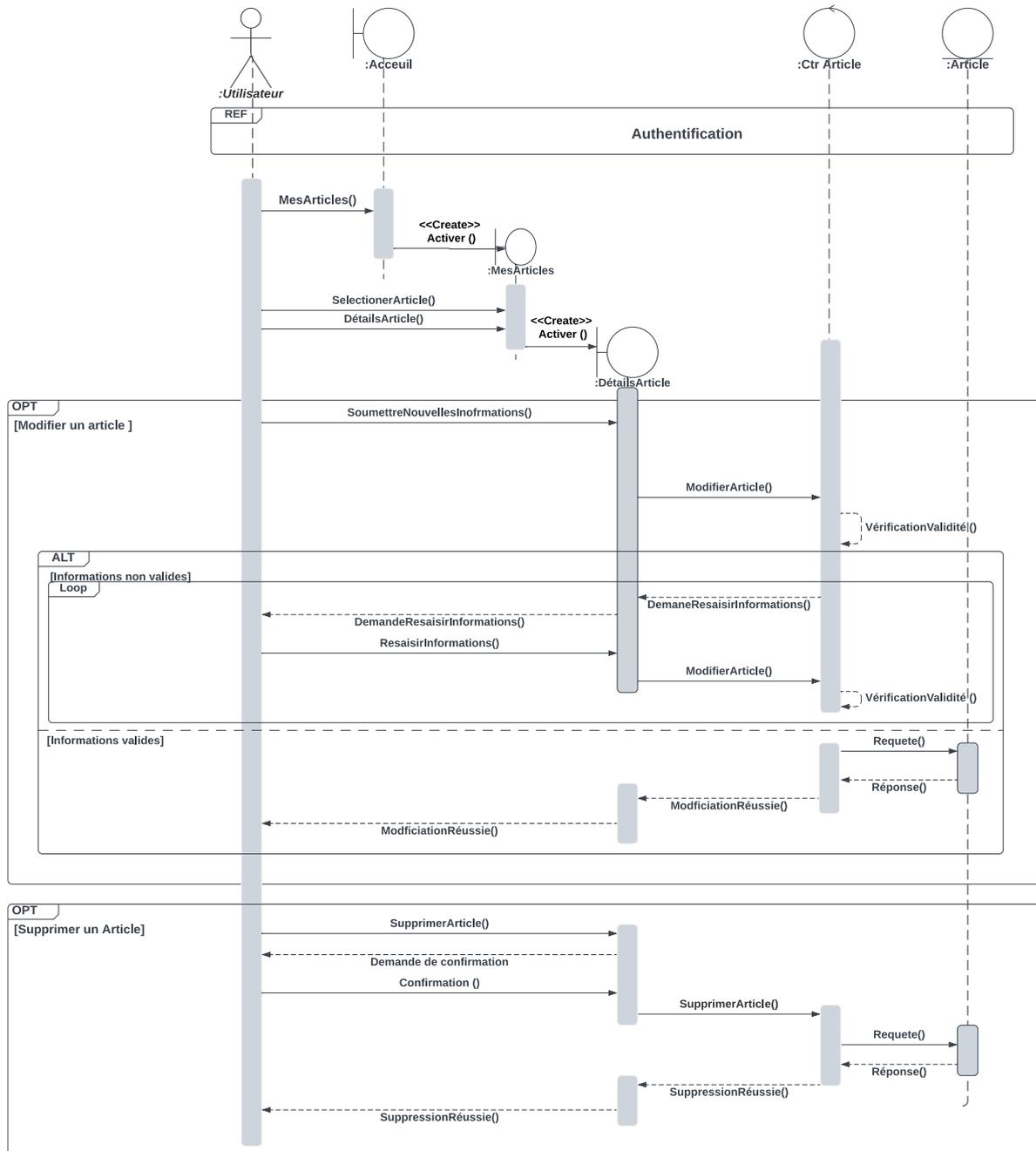


Figure III.7 : Diagramme de séquence de la fonctionnalité « Modifier/Supprimer un article»

e. Fonctionnalités de « Gestion de demandes »

- Fonctionnalité « Créer une demande pour un article »

L'utilisateur accède à la page des dons, il choisit le don qu'il veut demander. Le système affiche les détails du don choisi, l'utilisateur sélectionne l'option de demande associée à ce don, le système vérifie si le don est disponible.

Dans le cas de disponibilité, la demande est enregistrée et l'utilisateur reçoit une notification de succès. Sinon il reçoit une notification d'erreur lui signalant que le don n'est plus disponible.

La figure [III.8] représente le diagramme de séquence de la fonctionnalité « Ajouter une demandes »

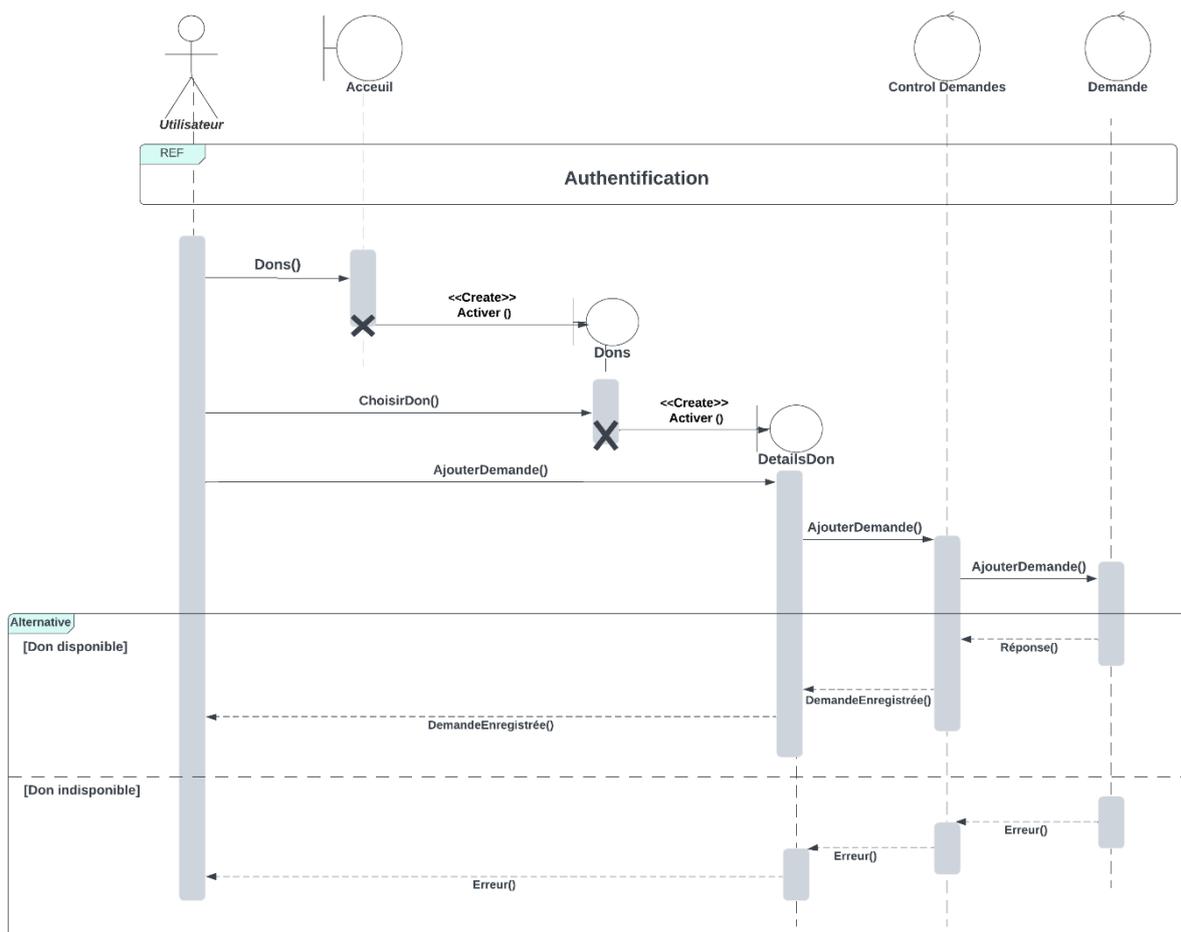


Figure III.8 : Diagramme de séquence du cas d'utilisation « Créer une demande »

- **Fonctionnalité « Accepter, Refuser, Annuler une demande »**

L'utilisateur à la section dédiée à la gestion des demandes. Sur cette page, une liste des demandes existantes est affichée.

En sélectionnant une demande spécifique et en fonction de son état actuel, l'utilisateur peut effectuer différentes actions telles que l'acceptation, l'annulation ou le rejet de la demande.

La figure [III.9] représente le diagramme de séquence de la fonctionnalité « Refuser, Accepter, Annuler une demande »

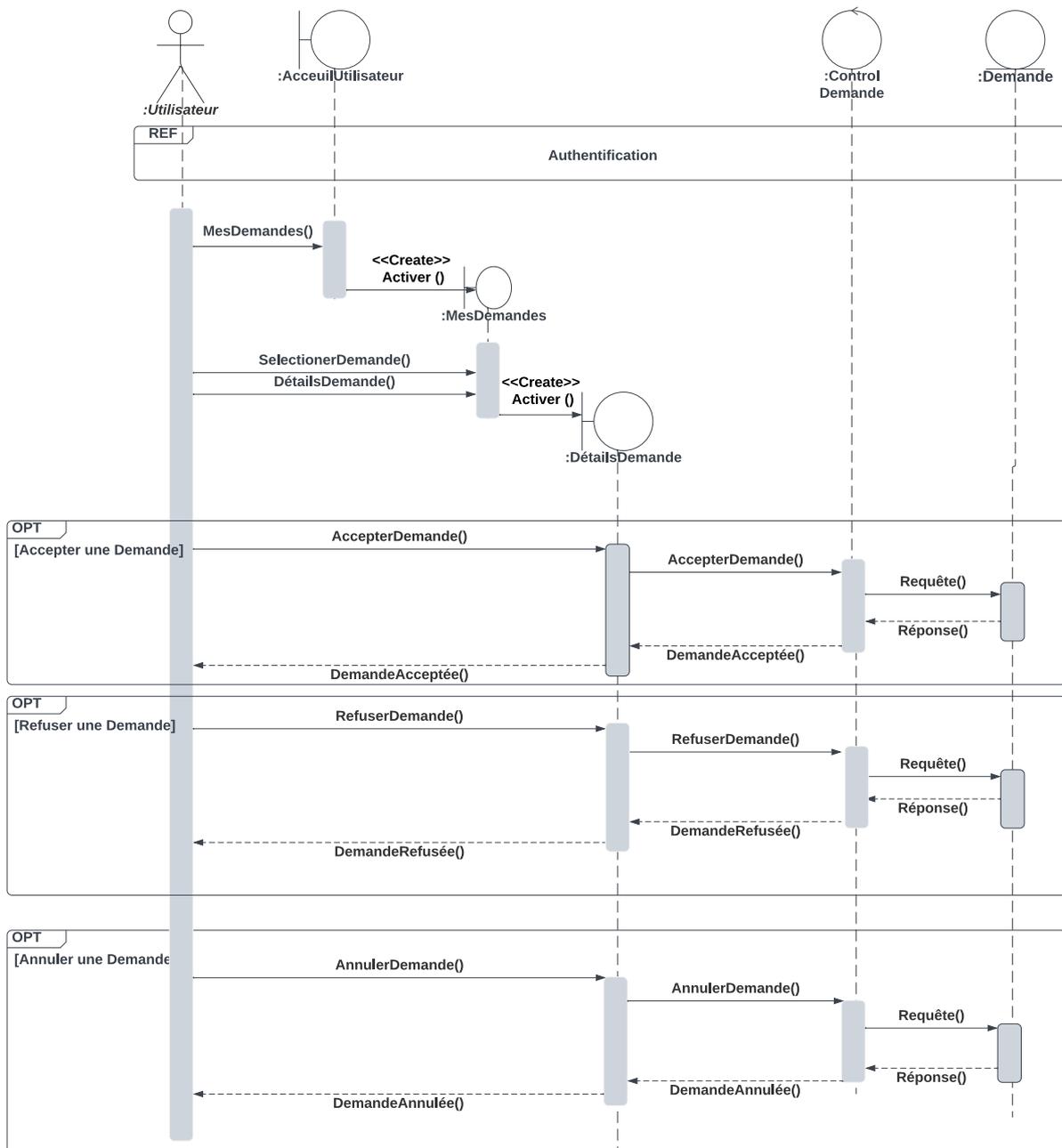


Figure III.9: Diagramme de séquence de la fonctionnalité «Refuser,Accepter,Annuler demande»

III.1.5 Diagramme de classe

Le diagramme de classe de la figure [III.10] décrit d'une façon globale les classes composant les items du sprint 1

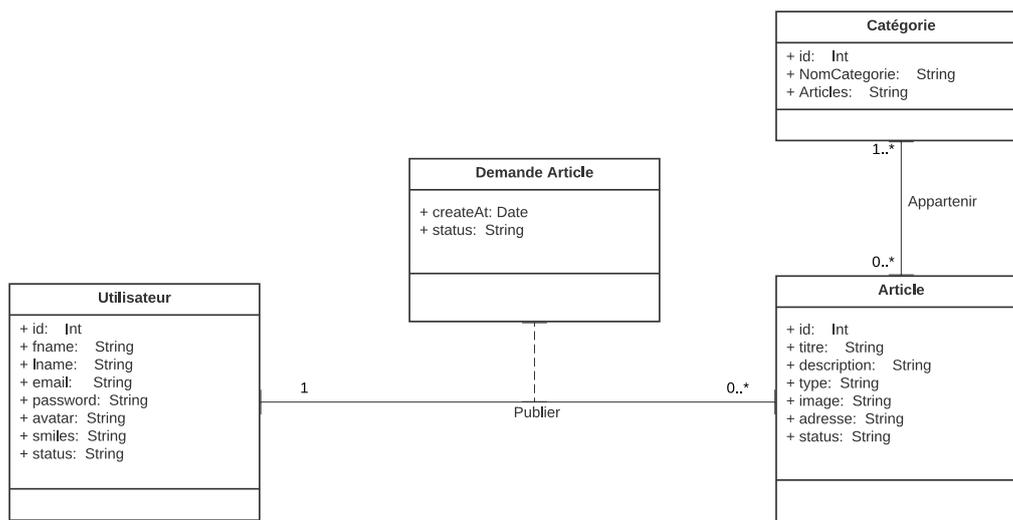


Figure III.10 : Diagramme de classe du sprint 1

III.1.6 Vue des interfaces utilisateur

Dans cette section nous allons présenter quelques interfaces graphiques de notre application de base

a. Interface d'accueil

L'utilisateur de accède à notre application via une interface d'accueil (la figure [III.11]), qui présentera notre plateforme, le contexte, les objectifs ainsi que les services proposés par notre application.



Figure III.11 : Interface D'accueil

b. Interface d'inscription

Pour que l'utilisateur accède à notre application via une interface qui affichera un formulaire d'inscription représentée par la figure [III.12].

Figure III.12 : Interface d'inscription

c. Interface Authentification

L'utilisateur de l'application accède aux services de notre application via une interface d'authentification (figure [III.13]) en saisissant son email et le mot de passe.

Figure III.13 : Interface d'authentification

d. Interface profile Utilisateur

L'utilisateur de l'application accède à son profil (figure [III.14]) de notre application via un bouton de sélection. Dans cette page l'utilisateur peut changer sa photo de profil,

ses données personnels, il peut aussi s'il n'est pas un organisateur il peut faire une demande pour le devenir.

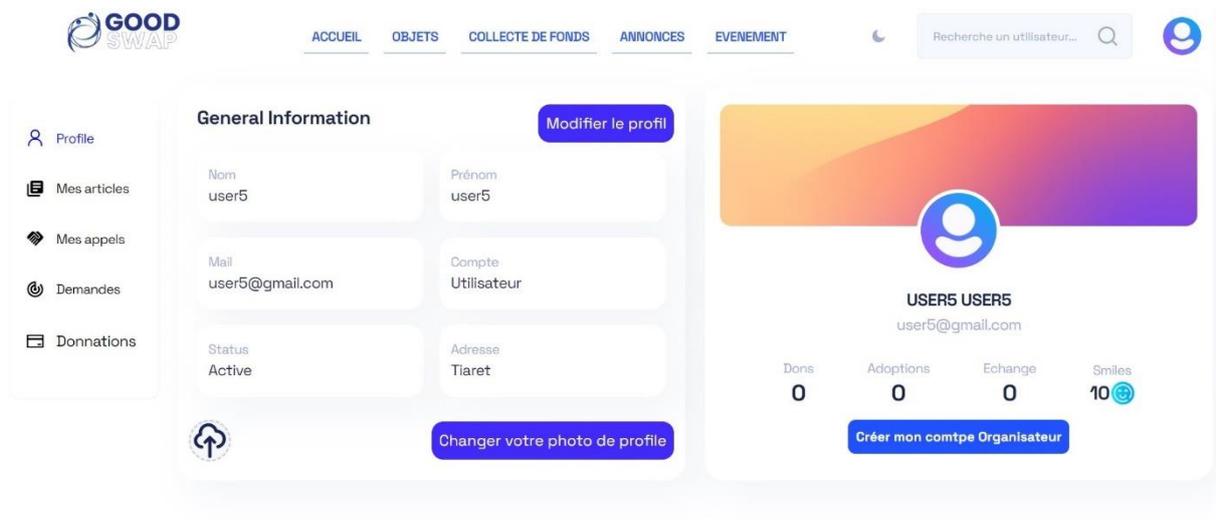


Figure III.14 : Interface de profil utilisateur

e. Interface Contact

La figure [III.15] représente la page contact qui joue un rôle essentiel dans la communication, offrant un moyen facile pour les visiteurs d'entrer en contact avec l'administrateur.

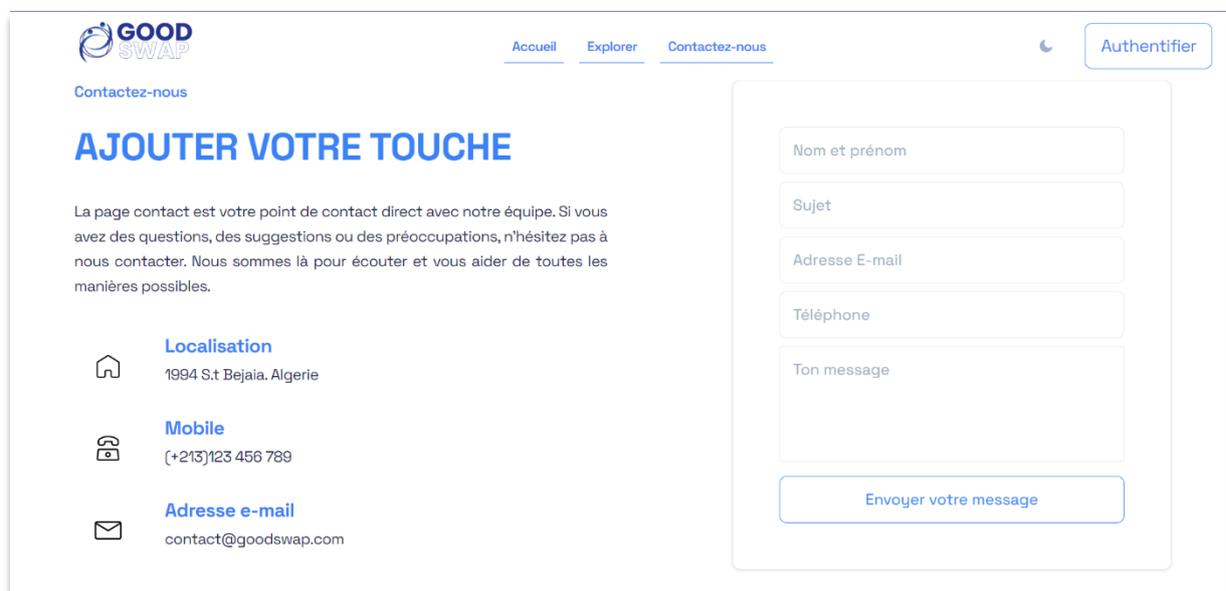


Figure III.15 : Interface Contact Administrateur

f. Interface de Tous les articles

La figure [III.16] représente la page Tous les articles (Objets) qui affiche un espace permet aux utilisateurs de visualiser tous les articles disponibles sur notre application

filtrer par catégorie et ceux qui sont près des utilisateurs et les plus récents, l'utilisateur peut rechercher n'importe quel article dans la barre de recherche.

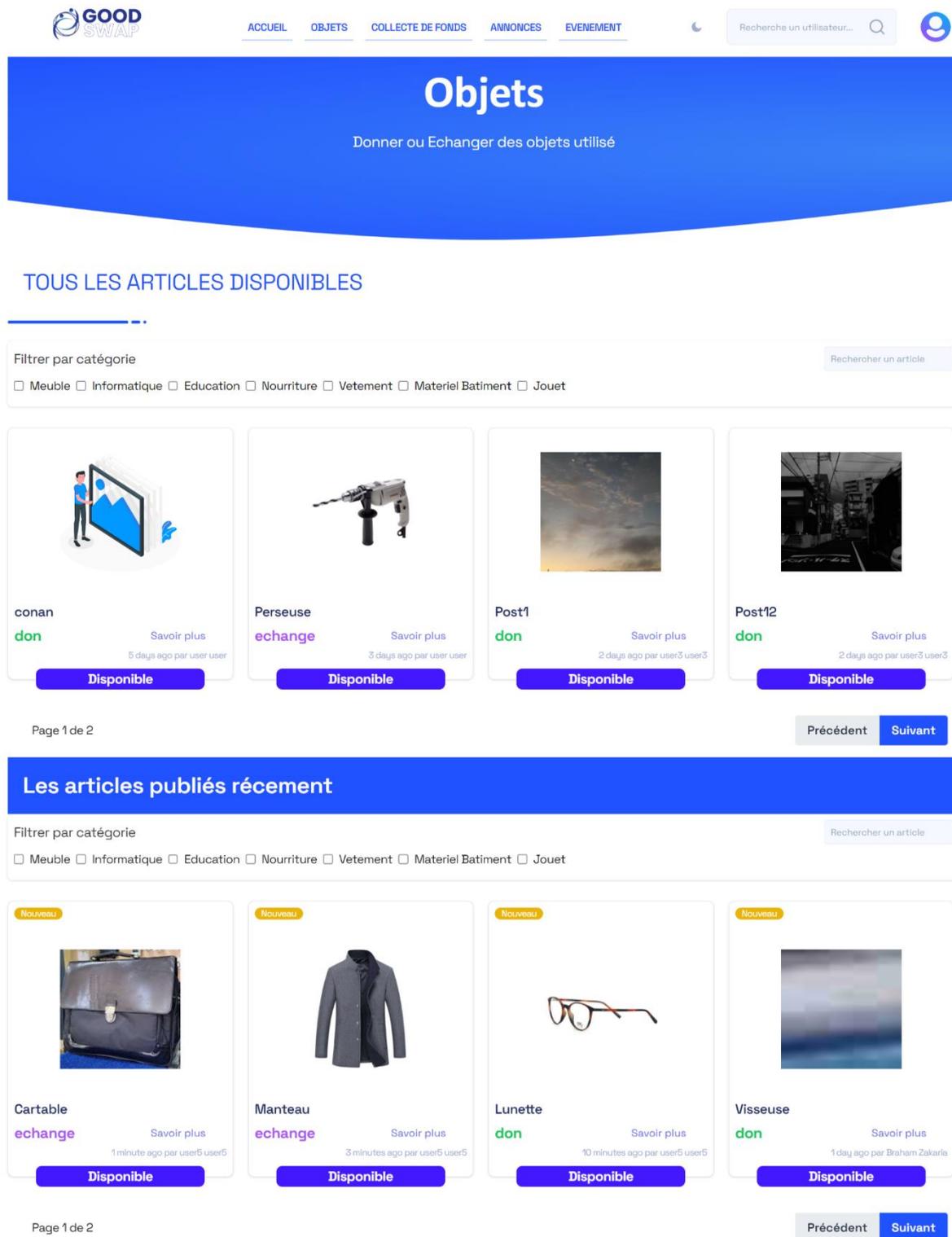


Figure III.16 : Interface Tous les articles (Objets)

e. **Interface Gestion de demandes**

Pour créer une demande pour un article ceci se fera via l'interface détail article en cliquant sur le Button demander (figure [III.17]), les interfaces de gestion des demandes permettent aux utilisateurs via plusieurs interfaces (figures [III.18] et [III.19] et [III.20]) d'approuver ou de rejeter une demande reçue ou bien annuler la réservation si on l'approuve, on peut aussi annuler, ou finaliser une demande envoyée. Ce qui simplifie le processus de gestion des demandes et favorise une communication claire et cohérente avec les demandeurs.

- **Interface Détail Article**

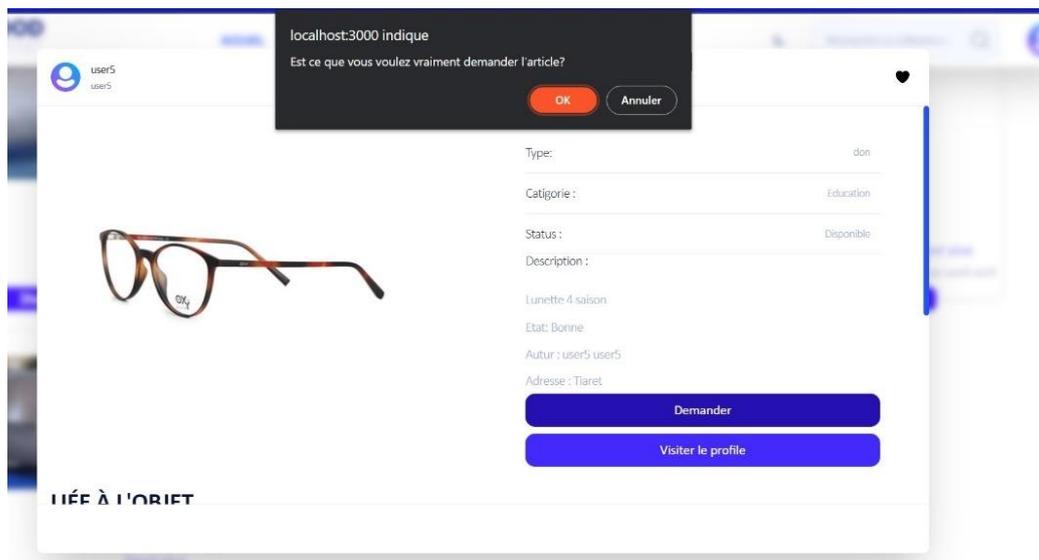


Figure III.17 : Interface Détail article

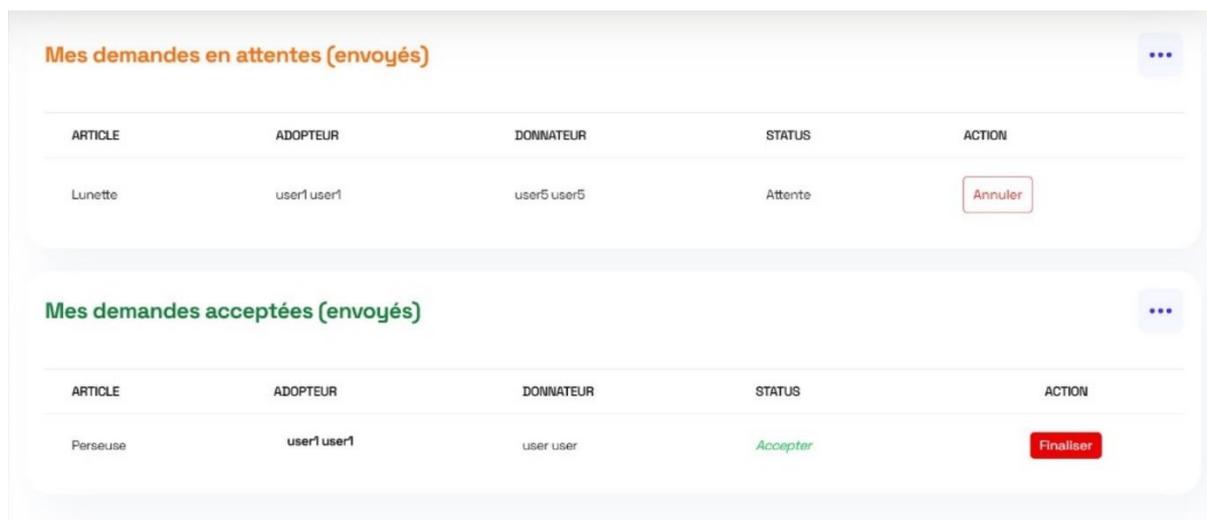


Figure III.18 : Interface « Mes demandes Envoyées »

The screenshot shows two tables of requests. The first table, titled 'Mes demande en attentes (Reçues)', has columns for ARTICLE, ADOPTEUR, DONNATEUR, STATUS, and ACTION. It contains one row for 'Lunette' with adopter 'user1 user1', donor 'user5 user5', and status 'Attente'. The second table, titled 'Mes demandes acceptées (Reçues)', has the same columns and contains one row for 'Perseuse' with adopter 'user1 user1', donor 'user user', and status 'Accepter'.

ARTICLE	ADOPTEUR	DONNATEUR	STATUS	ACTION
Lunette	user1 user1	user5 user5	Attente	Voir details

ARTICLE	ADOPTEUR	DONNATEUR	STATUS	ACTION
Perseuse	user1 user1	user user	Accepter	Annuler Reservation

Figure III.19 : Interface "Mes demandes Reçues"

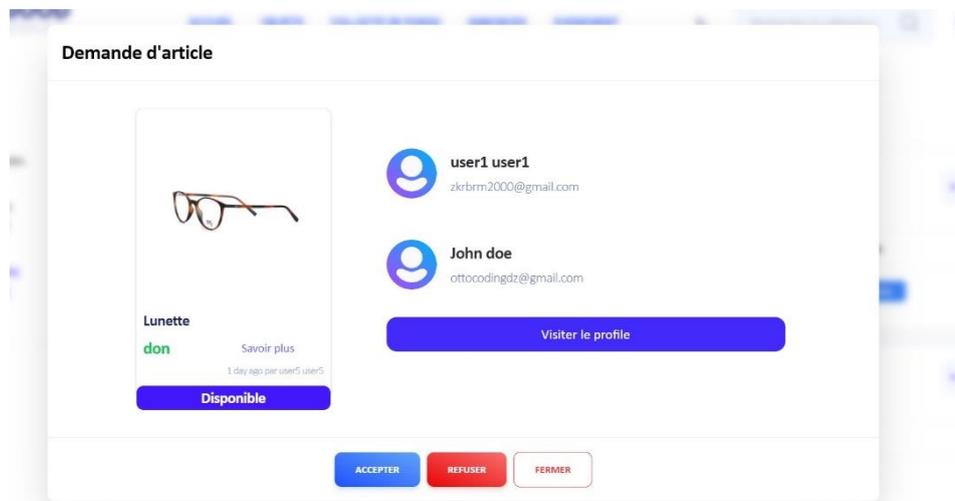


Figure III.20 : Interface de détails des demandes reçues

f. Interface de gestion des comptes par l'administrateur

L'interface de gestion des comptes (figure [III.21]), conçue spécifiquement pour les administrateurs, permet d'effectuer diverses actions, telles que la modification du statut des utilisateurs et la suppression des comptes indésirables.

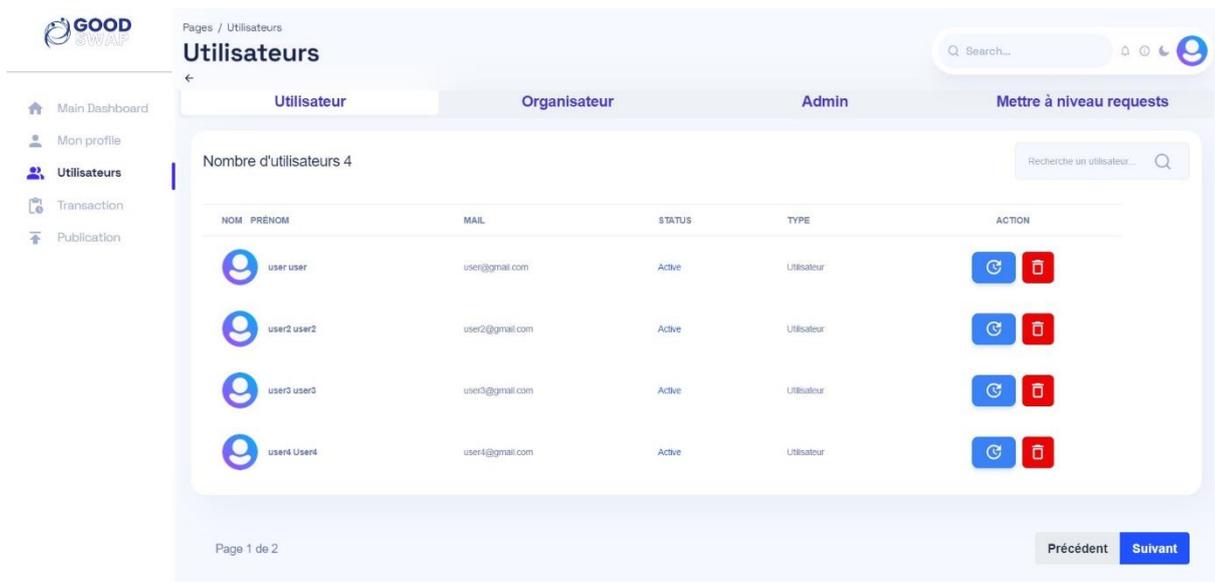


Figure III.21: Interface de gestion des comptes par l'administrateur

III.2 Conclusion

Dans ce chapitre, nous avons présenté le premier sprint de notre projet. Qui a été décomposé en un backlog spécifique contenant les tâches à réaliser et leurs estimations d'efforts. Les fonctionnalités du système ont été représentées à l'aide de diagrammes de cas d'utilisation et de diagrammes de séquence. Puis un diagramme de classes qui décrit de manière générale les classes composantes du sprint et enfin, nous avons donné un aperçu des interfaces utilisateur pour les fonctionnalités principales.

Dans le prochain chapitre, nous aborderons le déroulement des sprints 2 et 3, en poursuivant le développement des fonctionnalités du système et en consolidant progressivement son évolution.

IV. Sprint 2, 3

IV.1 Introduction

Dans ce chapitre, nous allons poursuivre le développement de notre application en examinant les sprints 2 et 3. Nous présenterons le sprint backlog de chaque sprint, décrivant les tâches à réaliser et leurs estimations, ainsi que les diagrammes de cas d'utilisation, les diagrammes de séquence et le diagramme de classe associés à ces fonctionnalités. Enfin, nous fournirons un aperçu des interfaces utilisateur pour les principales fonctionnalités développées.

IV.2 Sprint 2

Le deuxième sprint, d'une durée de deux semaines, est consacré à la réalisation et à l'implémentation de fonctionnalités supplémentaires identifiées précédemment.

IV.2.1 Diagramme de cas d'utilisation Sprint 2

Le diagramme de cas d'utilisation de la figure [IV.1] décrit d'une façon globale les cas d'utilisations composant les items du sprint 2

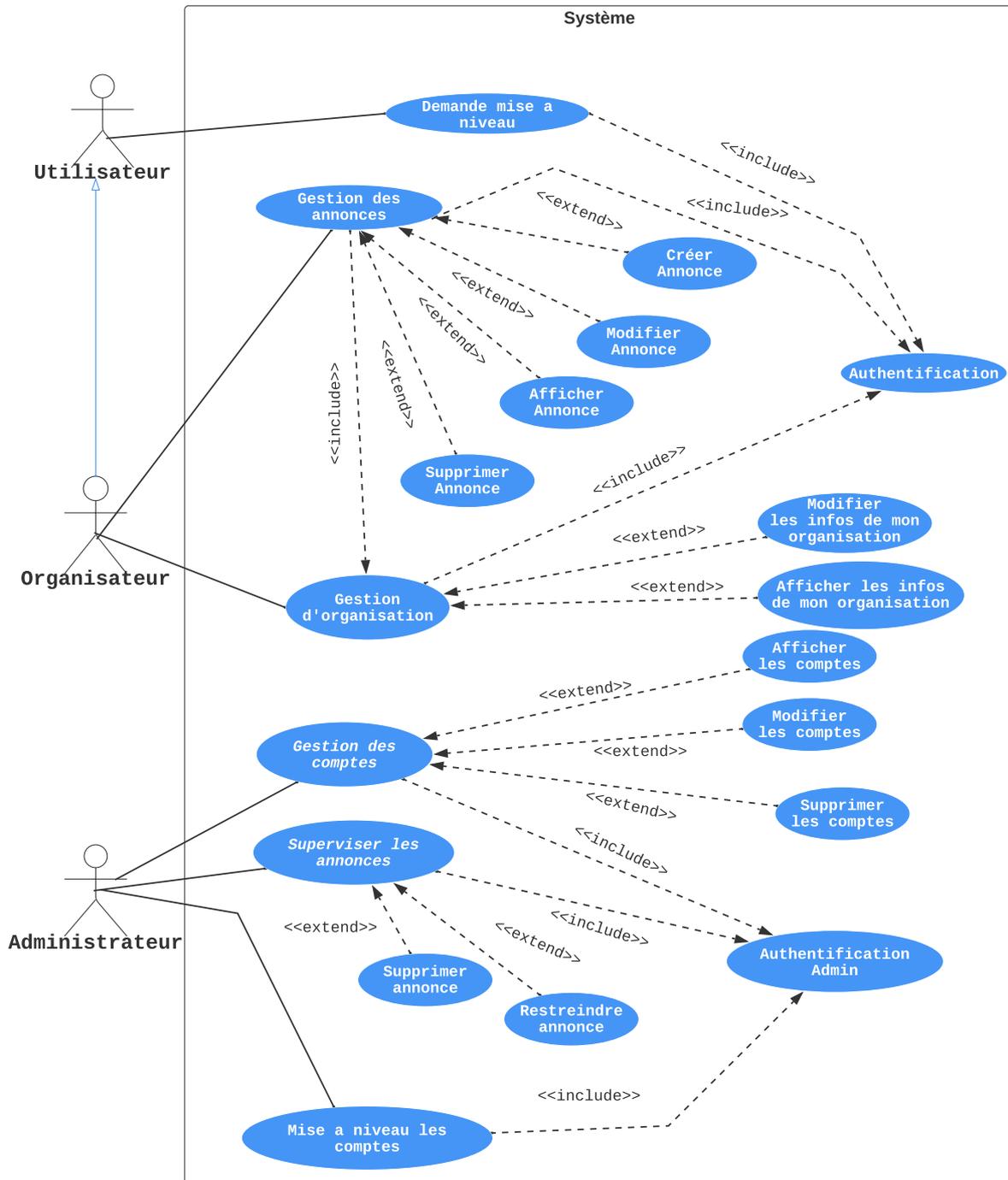


Figure IV.1 : Diagramme de cas d'utilisation du sprint 2

IV.2.2 Sprint Backlog

Tout d'abord on va présenter le backlog du sprint 2 qui contient l'ensemble des tâches à réaliser et leurs estimations qui figurent dans le tableau [IV.1] :

Fonctionnalité	Tache	Responsable	Estimation (Par heure)	Priorité	Statut
Réagir à une annonce	Créer une page pour afficher toutes les annonces	Hamis	8	Haute	En cours
	Créer un bouton de réaction	Hamis	4	Haute	En cours
	Implémenter la logique des réactions	Braham	12	Haute	En cours
Demande de mise à niveau	Créer un formulaire pour la demande de mise à niveau	Braham	6	Haute	En cours
	Implémenter la logique de la demande	Hamis	10	Haute	En cours
Gestion annonces	Créer des formulaires pour la création/modification d'une annonce	Hamis	8	Moyenne	En cours
	Créer une page pour afficher les détails d'une annonce	Braham	6	Moyenne	En cours
	Implémenter la logique de la gestion des annonces	Hamis	14	Moyenne	En cours
Mise à niveau compte utilisateur	Créer une page pour afficher les options de mise à niveau	Hamis	6	Moyenne	En cours
	Implémenter la logique de la mise à niveau	Braham	10	Moyenne	En cours
Gestion Organisations	Créer des formulaires pour la création/modification d'une organisation	Hamis	8	Moyenne	En cours
	Créer une page pour afficher les détails d'une organisation	Hamis	6	Moyenne	En cours
	Implémenter la logique de la gestion des organisations	Braham	12	Moyenne	En cours
Superviser Organisations	Mettre en place la fonctionnalité de restriction d'une organisation	Braham	8	Basse	En attente
	Implémenter la fonctionnalité de suppression d'une organisation	Braham	8	Basse	En attente
Superviser Annonces	Mettre en place la fonctionnalité de restriction d'une annonce	Braham	8	Basse	En attente
	Implémenter la fonctionnalité de suppression d'une annonce	Braham	8	Basse	En attente

Tableau IV.1 : Backlog du sprint 2

IV.2.3 Décomposition du sprint 2

Tout au long de cette période, nous traitons les fonctionnalités de ce sprint pour produire un incrément potentiellement livrable. La figure [IV.2] représente la décomposition du sprint 2 en termes de temps.

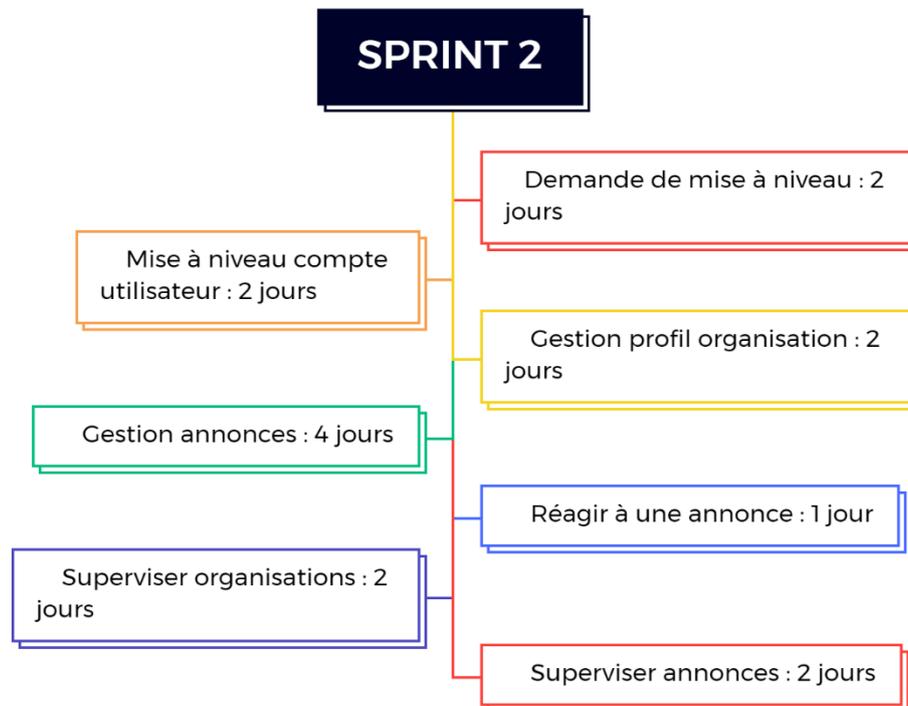


Figure IV.2 : Décomposition du sprint 2

IV.2.4 Présentation des diagrammes de séquence

Nous allons décrire d'une manière détaillée les fonctionnalités :

« Mettre au niveau les comptes utilisateur »

« Gestion d'organisation »

a. Fonctionnalité Mettre à niveau les comptes Utilisateur

L'administrateur accède à la page ou à la section dédiée à la mise à niveau des comptes utilisateur. L'administrateur sélectionne la de mise à niveau appropriée d'un compte utilisateur spécifique.

La figure [IV.3] représente le diagramme de séquence de la fonctionnalité « Mettre à niveau les comptes Utilisateur »

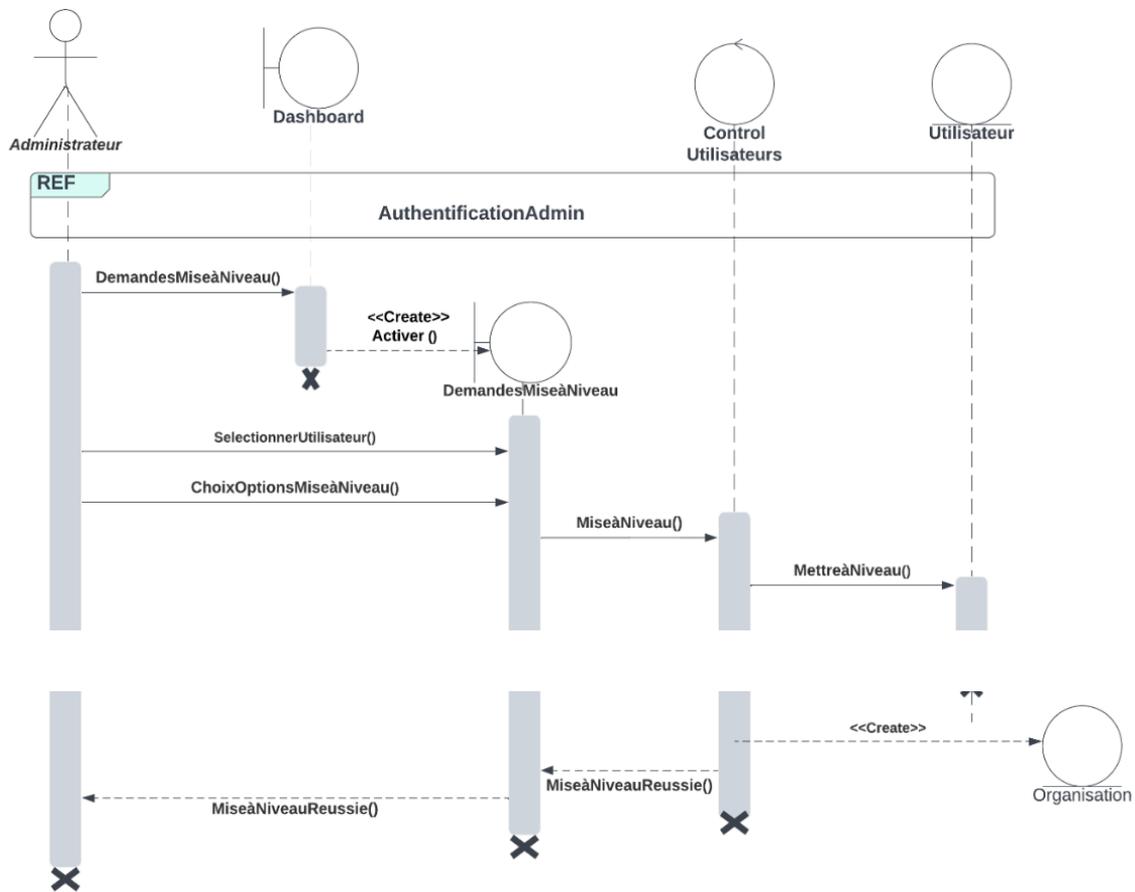


Figure IV.3 : Diagramme de séquence de la fonctionnalité « Mettre à niveau Utilisateur »

b. Fonctionnalité Gestion d'organisation

Une fois qu'un organisateur s'est connecté avec succès, il accède à son espace d'organisation. Sur cette page, il peut consulter les informations de son organisation telles que son nom, son adresse e-mail, son numéro de téléphone, adresse etc.

L'organisateur a la possibilité de modifier ses informations de l'organisation en cliquant sur un bouton, un formulaire prérempli s'affiche, présentant les informations actuelles de l'organisation.

L'organisateur peut alors apporter des modifications aux champs qu'il souhaite mettre à jour, tels que son adresse e-mail, son numéro de téléphone ou toute autre information pertinente. Une fois qu'il a effectué les modifications souhaitées, il soumet le formulaire de mise à jour.

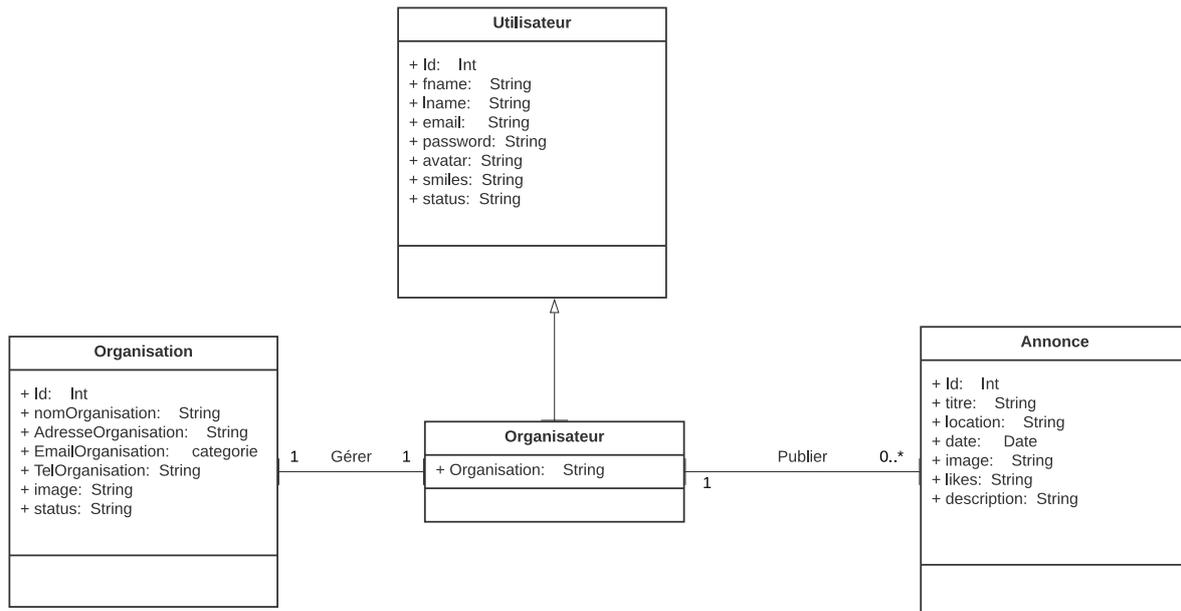


Figure IV.5 : Diagramme de classe du sprint 2

IV.2.6 Vue des interfaces utilisateur

a. Formulaire demande mise à niveau

Le formulaire de demande de mise à niveau (figure [IV.6]) offre aux utilisateurs la possibilité de demander une amélioration ou une mise à niveau de leur compte ou de leurs services existants.

Faire une demande ✕

Nom du Organisation <input type="text" value="Volunteer Plus"/>	Adresse du Organisation <input type="text" value="Bejaia"/>	
Email <input type="text" value="volunteer.plus@outlook.com"/>	Telephone <input type="text" value="1233456789"/>	
Logo de l'Organisation <input type="button" value="Choisir un fichier"/> Blu....png	License/Agrément : <input type="button" value="Choisir un fichier"/> Cap...png	N° Piece d'identité: <input type="text" value="123456789"/>

Figure IV.6 : Formulaire de Demande de mise à niveau coté Utilisateur

b. Page de demandes mise à niveau

La page de demandes de mise à niveau (figure [IV.7]), conçue pour les administrateurs, offre une interface centralisée pour gérer et traiter les demandes de mise à niveau soumises par les utilisateurs.

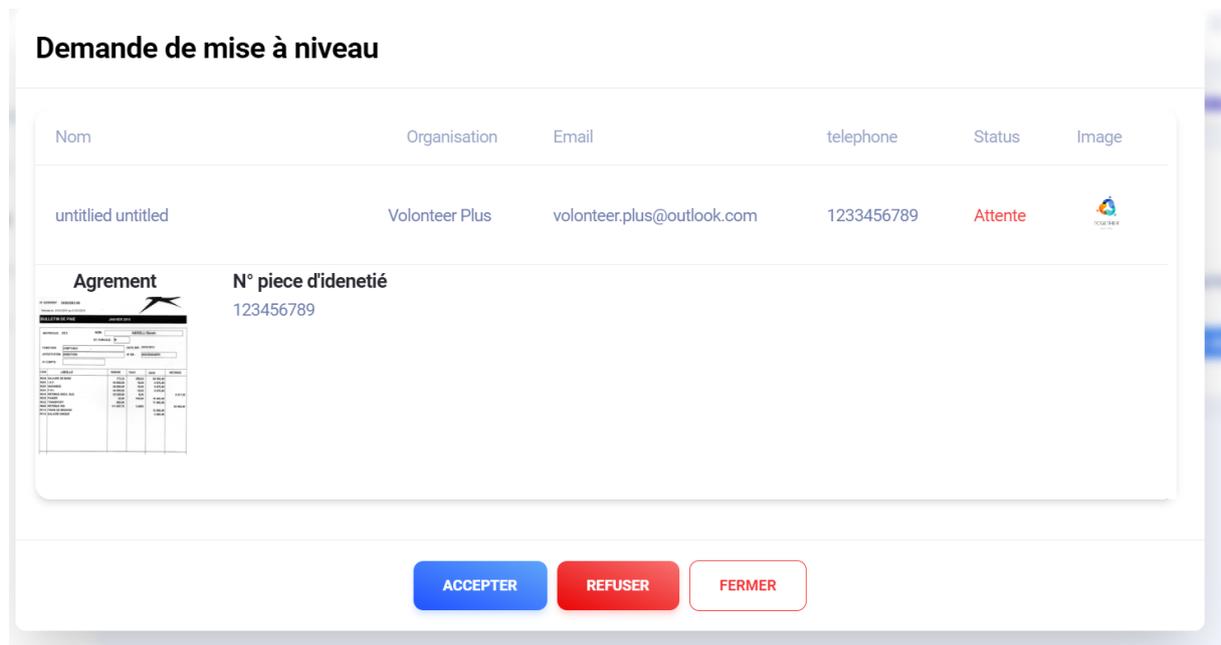


Figure IV.7 : Gestion des demandes mise à niveau coté Administrateur

c. Espace Organisateur

L'Espace Organisateur (figure [IV.8]) proposera une gestion complète du profil, des annonces, des événements, des cagnottes et des statistiques pour permettre aux organisateurs d'optimiser leurs activités

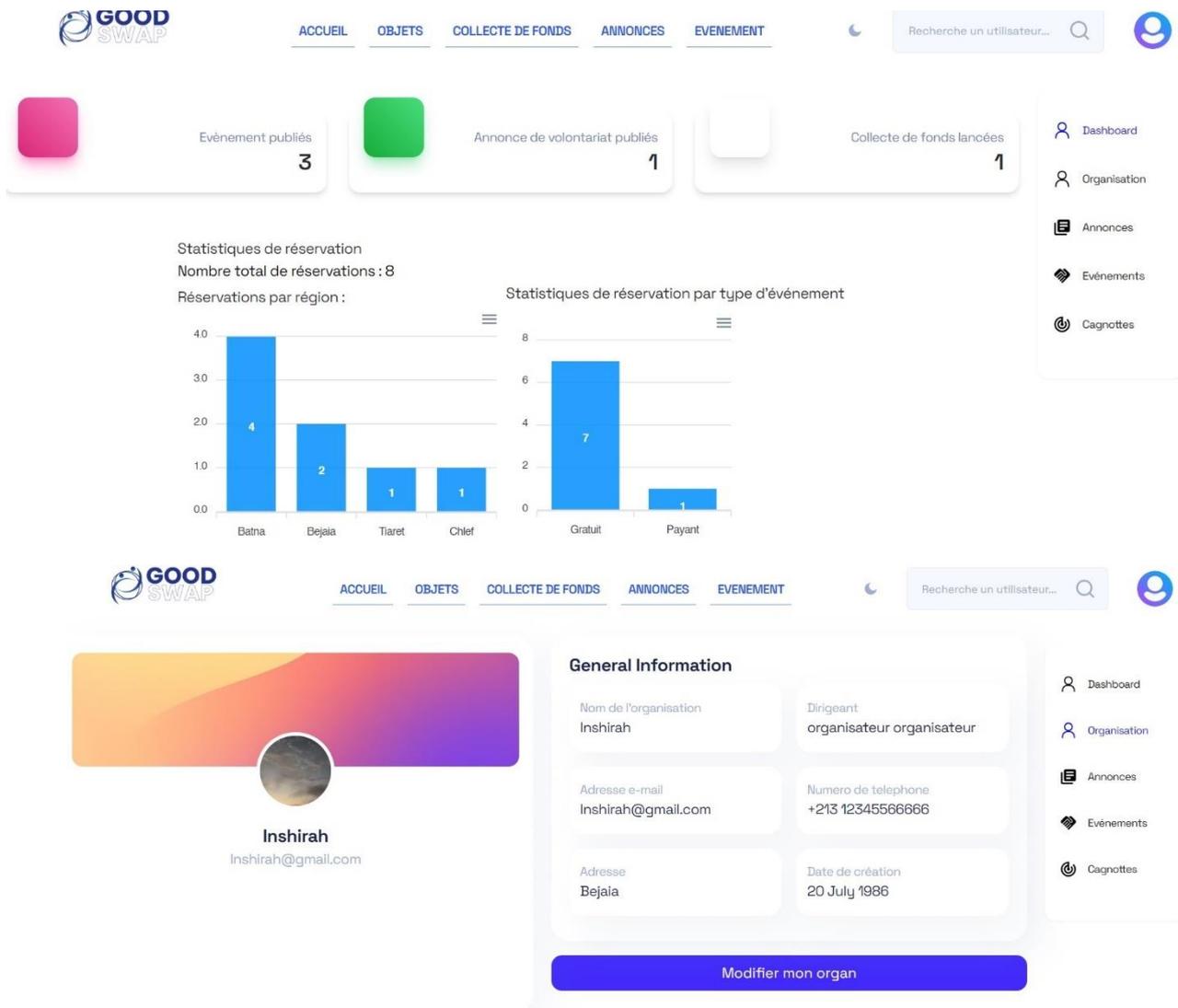


Figure IV.8 : Espace et profile d'une organisation

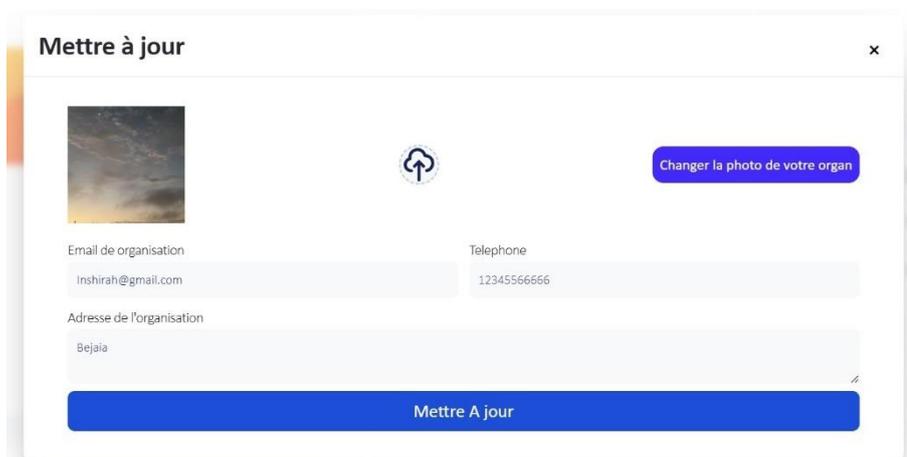


Figure IV.9 : Interface Modifier le profil d'une organisation

IV.3 Sprint 3

Le troisième sprint, d'une durée de deux semaines, est dédié à la réalisation et à l'implémentation de nouvelles fonctionnalités.

IV.3.1 Diagramme de cas d'utilisation Sprint 3

Le diagramme de cas d'utilisation de la figure [IV.10] décrit d'une façon globale les cas d'utilisations composant les fonctionnalités du sprint 3.

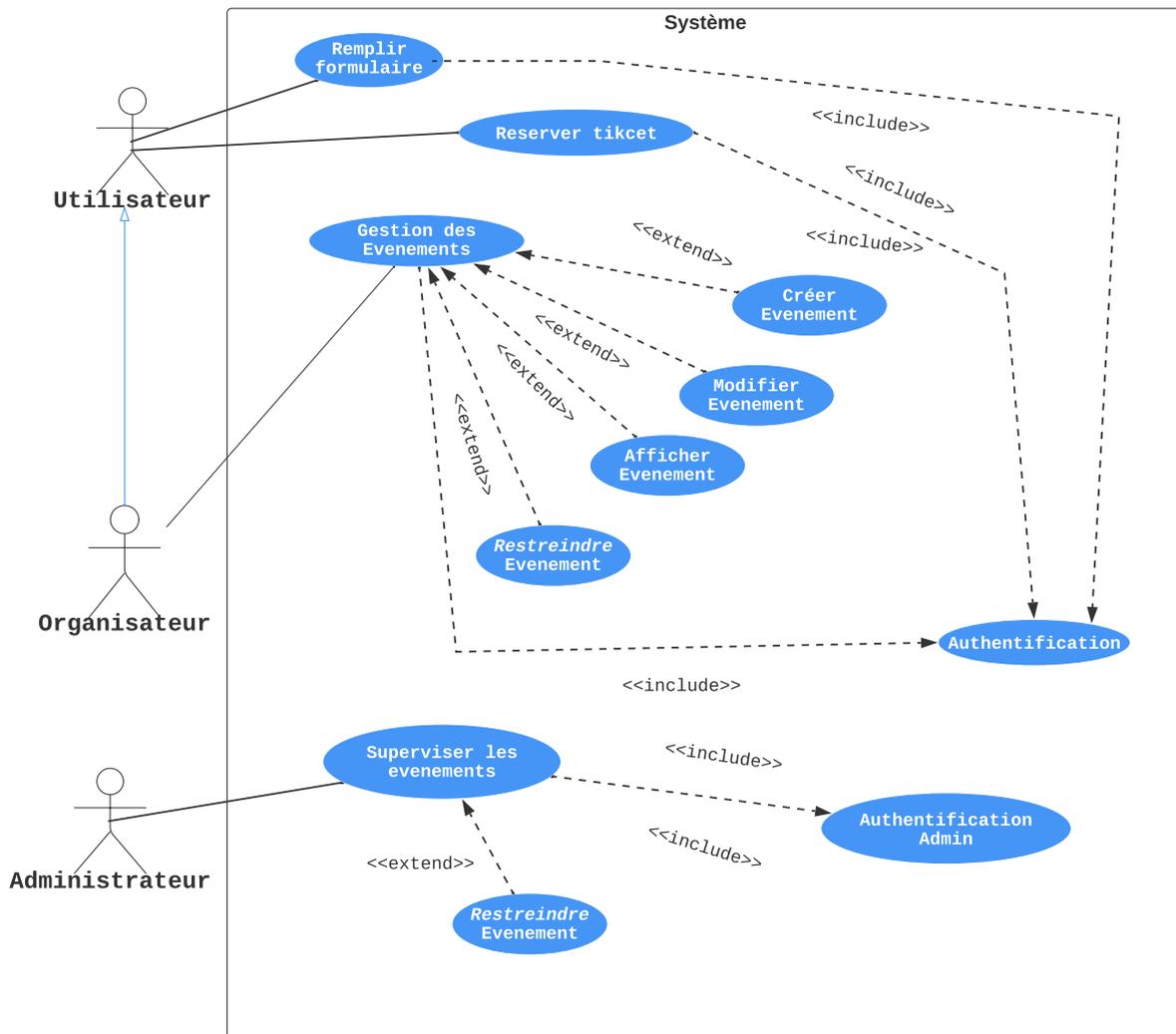


Figure IV.10 : Diagramme de cas d'utilisation du sprint 3

IV.3.2 Sprint Backlog

Tout d'abord on va présenter le backlog du sprint 3 qui contient l'ensemble des tâches à réaliser et leurs estimations qui figurent dans le tableau [IV.2] :

Fonctionnalité	Tache	Responsable	Estimation d'efforts (Par heure)	Priorité	Statut
Réserver un ticket	Créer le formulaire de réservation de ticket	Hamis	4	Haute	En cours
	Implémenter la logique de réservation	Braham	6	Haute	En cours
Gestion événements	Créer le formulaire de création d'événement	Hamis	8	Moyenne	A faire
	Implémenter la logique de création d'événement	Braham	10	Haute	A faire
	Créer la page d'affichage de détails d'événement	Hamis	8	Moyenne	A faire
	Créer le formulaire de modification d'événement	Hamis	8	Moyenne	A faire
	Implémenter la logique de modifier et supprimer événement	Braham	10	Haute	A faire
	Créer le bouton de suppression d'événement	Hamis	8	Moyenne	A faire
Inscription évènement	Implémenter la logique de s'inscrire à l'évènement	Braham	8	Moyenne	A faire
	Créer le formulaire de réservation de ticket	Hamis	8	Moyenne	A faire

Tableau IV.2 : Backlog du sprint 3

IV.3.3 Décomposition du sprint 3

Tout au long de cette période, nous traitons les fonctionnalités de ce sprint pour produire un incrément potentiellement livrable. La figure [IV.11] représente la décomposition du sprint 3 en termes de temps.

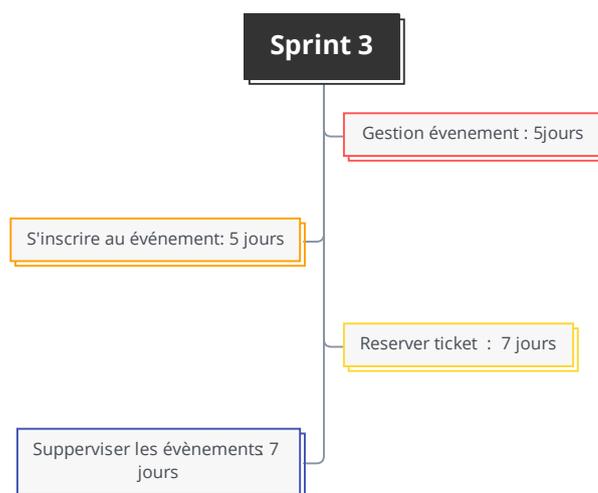


Figure IV.11 : Décomposition du sprint 3

IV.3.4 Présentation des diagrammes de séquence

Nous allons décrire d'une manière détaillée les fonctionnalités :

- « Réservation ticket »
- « Inscription évènement »

a. Fonctionnalité « Inscription évènement »

L'utilisateur accède à la liste des événements disponibles sur l'application. En parcourant la liste, il sélectionne un événement qui l'intéresse et souhaite y participer. Le système affiche les détails de l'événement sélectionné parmi les options disponibles. Si l'évènement est payant, l'utilisateur choisit l'option "S'inscrire" pour participer à l'évènement. Le système affiche un formulaire d'inscription contenant un champ de paiement à remplir. Une fois le paiement effectué, le système notifie l'utilisateur du succès de son inscription. La figure [IV.12] représente le diagramme de séquence de la fonctionnalité « Inscription évènement ».

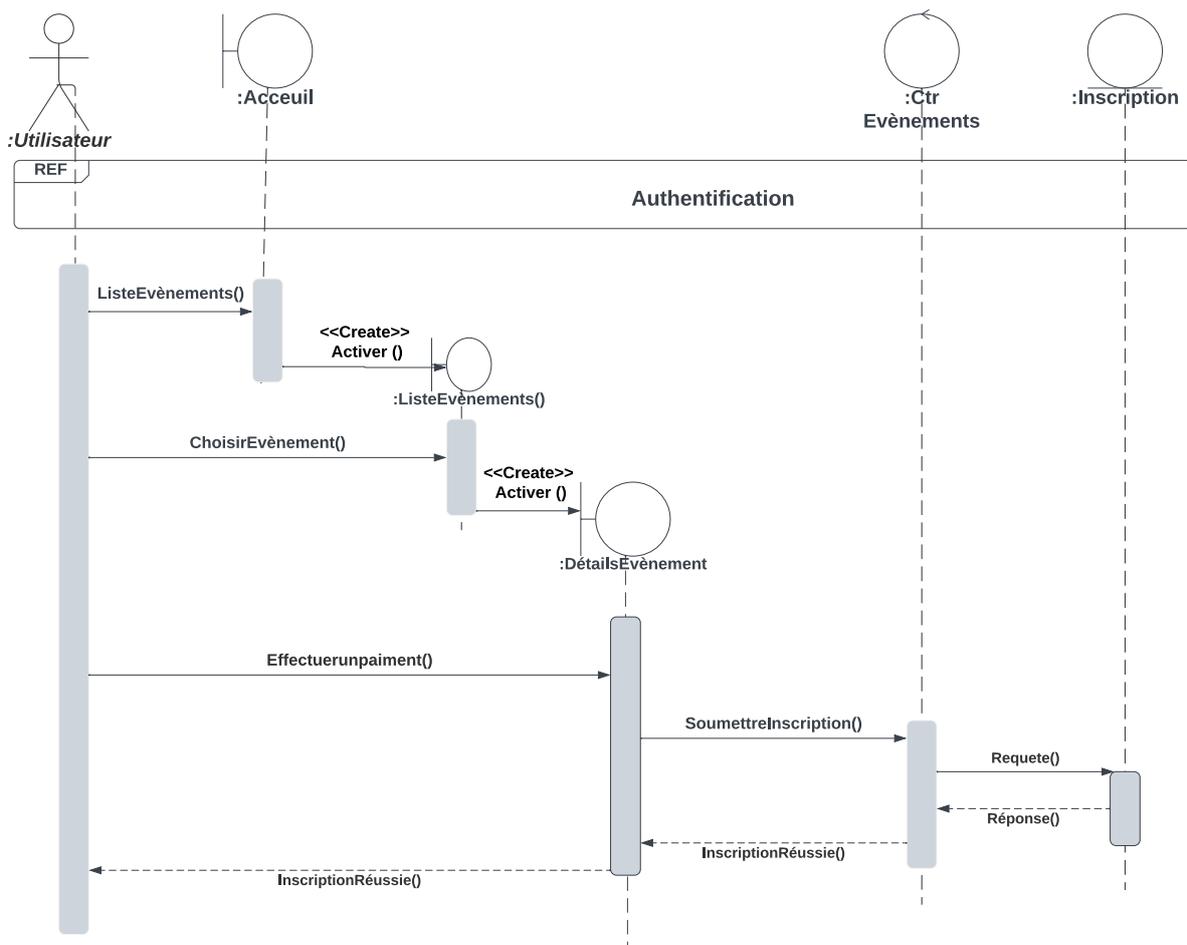


Figure IV.12 : Diagramme de séquence du cas d'utilisation « Inscription évènement »

b. Fonctionnalité « Réservation ticket »

L'utilisateur accède à la liste des événements disponibles sur l'application. En parcourant la liste, l'utilisateur sélectionne un événement pour lequel il souhaite réserver un ticket. Le système affiche les détails de l'événement sélectionné, parmi les options disponibles, l'utilisateur choisit l'option "Réserver un ticket" pour assister à l'événement. Le système affiche un formulaire de réservation contenant un champ de paiement à remplir. Une fois le paiement effectué, le système notifie l'utilisateur du succès de son inscription. La figure [IV.13] représente le diagramme de séquence de la fonctionnalité « Réservation ticket ».

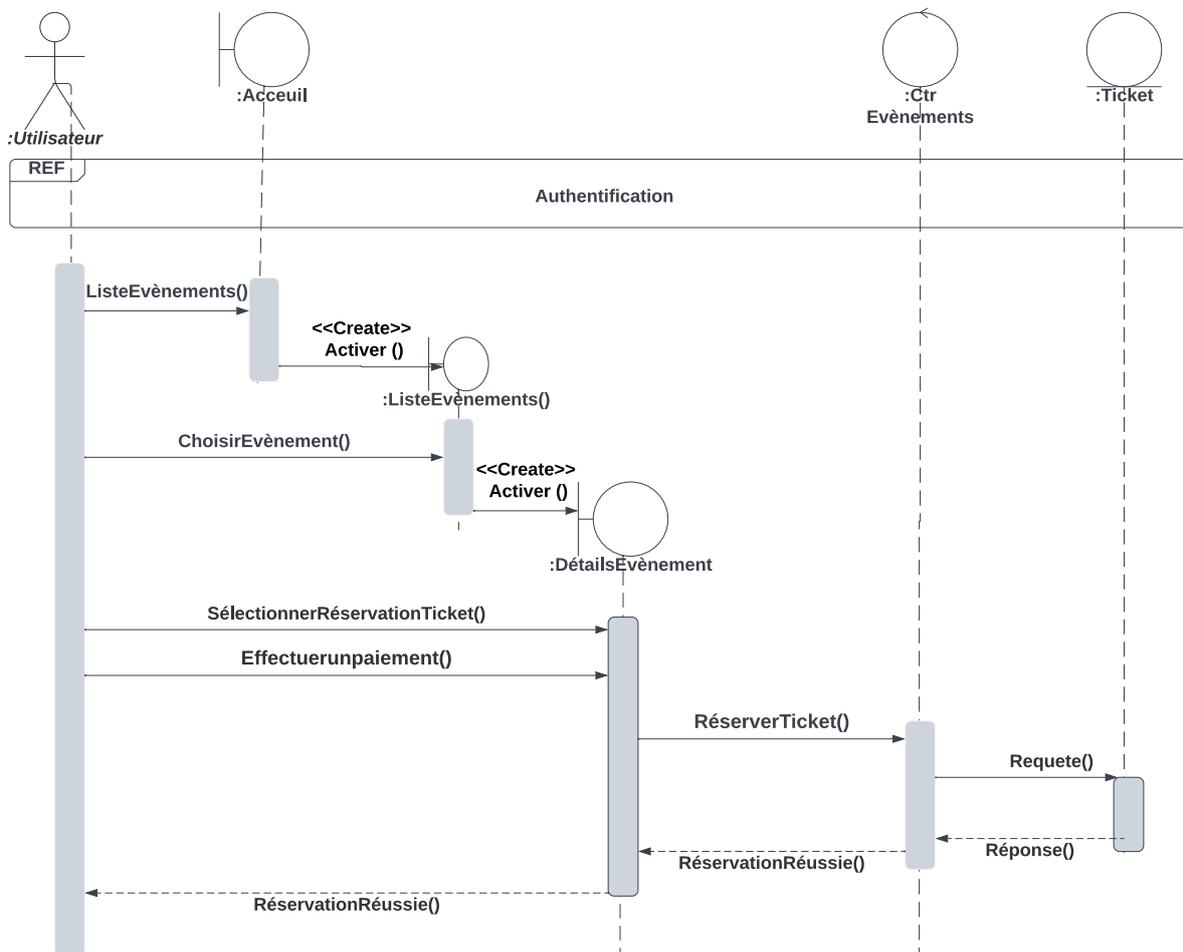


Figure IV.13 : Diagramme de séquence du cas d'utilisation « Réservation ticket »

IV.3.5 Diagramme de classe du sprint 3

Le diagramme de classe de la figure [IV.14] décrit d'une façon globale les classes composant les items du sprint 3.

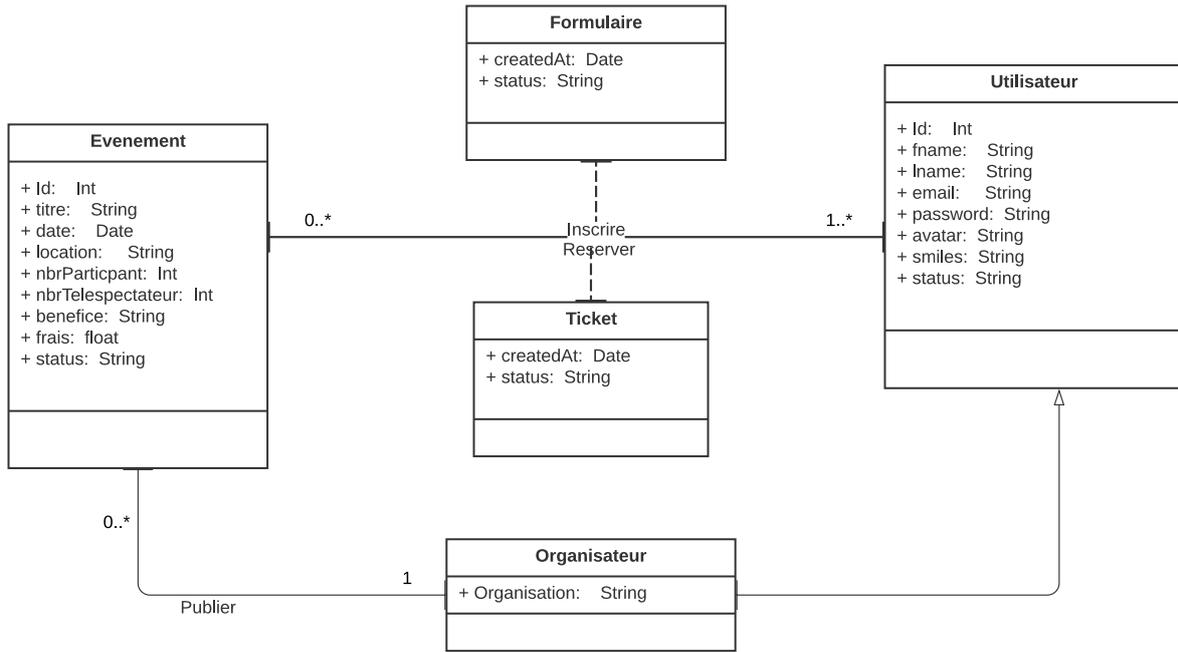


Figure IV.14 : Diagramme de classe du sprint 3

IV.3.6 Vue des interfaces utilisateur des cas d'utilisations

a. Interface Inscription évènement

L'interface "Inscription à l'évènement" (figure [IV.15]) permet aux utilisateurs de s'inscrire facilement à des évènements pertinents.

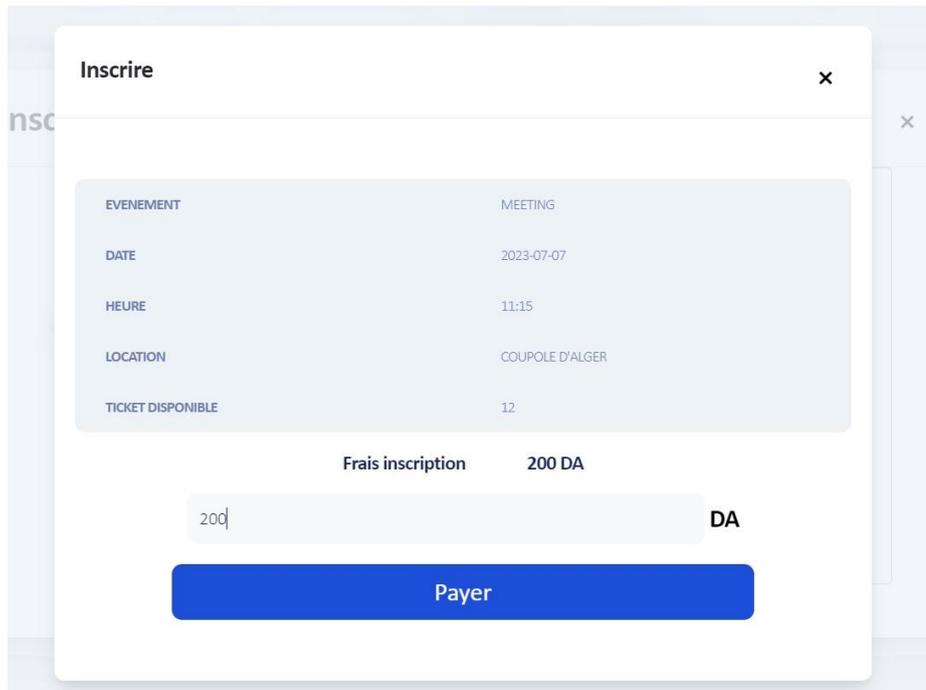


Figure IV.15: Interface Inscription évènement

b. Interface Réservation ticket (événement gratuit)

L'interface "Réservation à l'événement gratuit" (figure [IV.16]) permet aux utilisateurs de réserver des tickets gratuitement pour assister à un événement.

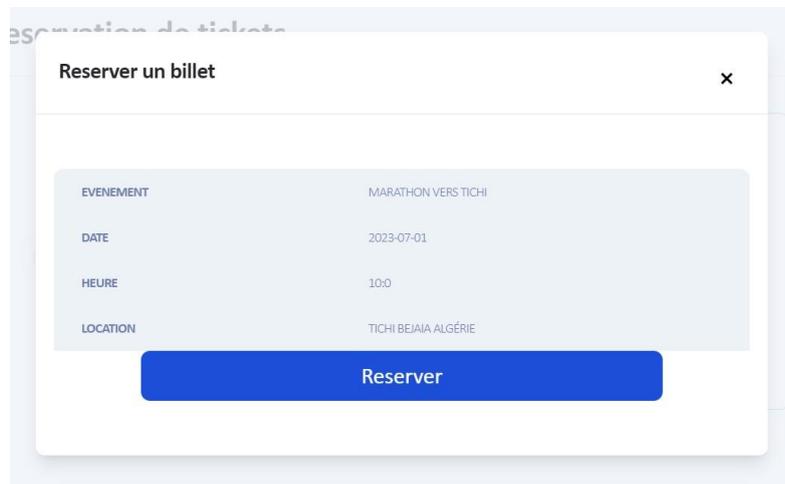


Figure IV.16 : Interface Réservation Ticket (événement gratuit)

c. Interface réservation ticket payant

L'interface "Réservation à l'événement payant" (figure [IV.17]) permet aux utilisateurs de réserver des tickets gratuitement pour assister à un événement

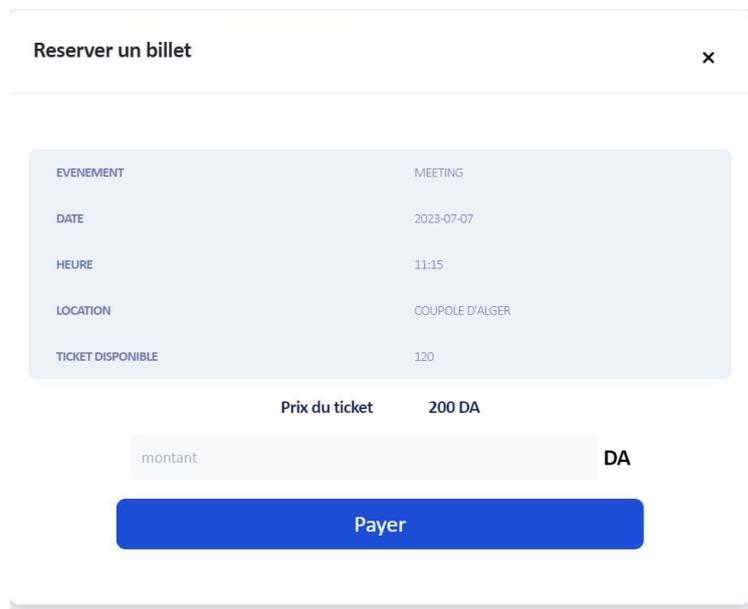


Figure IV.17 : Interface Réservation Ticket (événement Payant)

d. Interfaces gestion événement

Les interfaces de gestion des événements (figure [IV.18], [IV.19] et [IV.20]) offre aux organisateurs la possibilité de gérer leurs événements de manière pratique en leur

permettant d'ajouter de nouveaux événements, de consulter les détails existants, ainsi que d'accéder à d'autres fonctionnalités de gestion essentielles.

- **Interface Ajouter un événement**

Figure IV.18 : Interface Ajouter un événement Payant

- **Interface Détails d'un évènement**

Figure IV.19 : Interface détail d'évènement

- Interface Tous les événements



Figure IV.20 : Interface Tous les événements

IV.4 Conclusion

Dans ce chapitre, nous avons présenté en détail les sprints 2 et 3 de notre projet, au cours desquels nous avons travaillé sur la réalisation et l'implémentation de différentes fonctionnalités. Dans le prochain chapitre, nous aborderons le déroulement des sprints 4, 5 et 6, où nous poursuivrons le développement du système en ajoutant de nouvelles fonctionnalités et en consolidant son évolution progressive pour répondre aux besoins croissants de notre projet.

V. Sprint 4,5,6

V.1 Introduction

Dans ce chapitre, nous exposons les trois derniers sprints du projet, mettant en évidence leurs caractéristiques distinctes. Chaque sprint est accompagné d'un backlog détaillé qui comprend un ensemble de fonctionnalités spécifiques du système. Par la suite, nous présenterons les diagrammes de classes participantes et les diagrammes de séquence correspondants pour chaque sprint, offrant ainsi une vision claire des interactions et des flux de données.

V.2 Sprint 4

Le quatrième sprint, d'une durée de deux semaines, est dédié à la réalisation et à l'implémentation des fonctionnalités essentielles identifiées pour atteindre nos objectifs.

V.2.1 Diagramme de cas d'utilisation Sprint 4

Le diagramme de cas d'utilisation de la figure [V.1] décrit d'une façon globale les cas d'utilisations composant les items du sprint 4

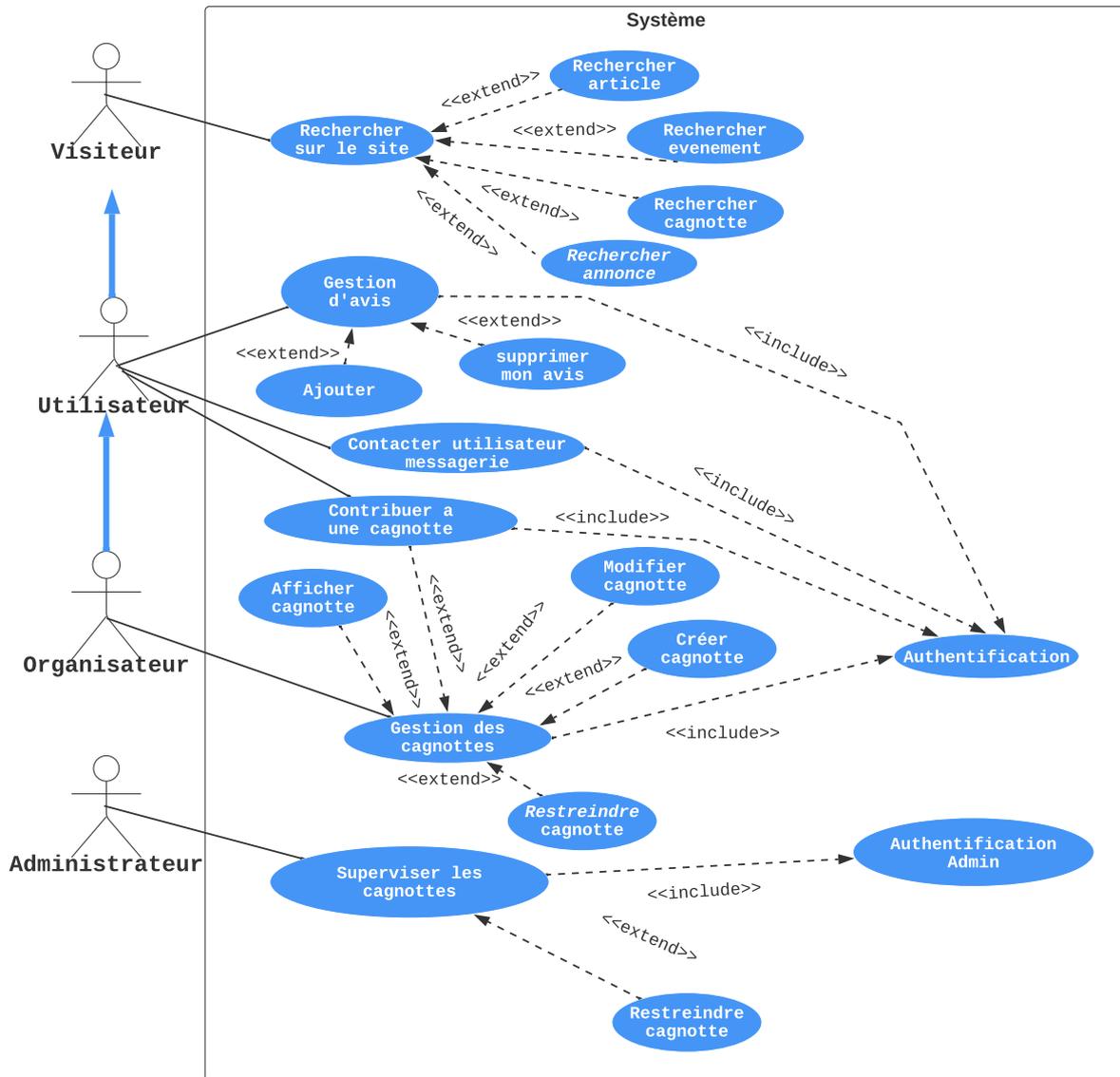


Figure V.1 : Diagramme de cas d'utilisation du sprint 4

V.2.2 Sprint Backlog

Tout d’abord nous allons présenter le backlog du sprint 4 qui contient l’ensemble des taches à réaliser et leurs estimations horaires qui figurent dans le tableau [V.1] :

Fonctionnalité	Tache	Responsable	Estimation d’efforts (Par heure)	Priorité	Statut
Rechercher sur site	Créer le composant de la fonctionnalité de recherche	Hamis	8	Haute	En cours
	Implémenter la logique de recherche	Braham	12	Haute	En cours
	Réaliser les formulaires de création et modification de cagnottes	Hamis	10	Haute	A faire

Gestion cagnottes	Implémenter la logique de création et modification	Braham	12	Haute	A faire
	Créer le bouton de restriction	Hamis	2	Basse	A faire
	Implémenter la logique de restriction	Braham	4	Moyenne	A faire
	Créer la page d'affichage des détails d'une cagnotte	Braham	8	Haute	A faire
Superviser cagnottes	Implémenter la logique de suppression d'une cagnotte	Braham	8	Moyenne	A faire
	Implémenter la fonctionnalité de restriction d'accès à une cagnotte	Braham	12	Moyenne	A faire
Gestion des avis	Créer les composants de l'interface utilisateur, formulaire d'ajout un avis bouton suppression	Hamis	2	Basse	A faire
	Implémenter la logique du coté serveur Ajout, suppression, affichage	Braham	8	Basse	A faire
Messagerie	Créer l'interface de la messagerie	Hamis	10	Haute	A faire
	Implémenter la logique de messagerie	Braham	15	Haute	A faire

Tableau V.1 : Backlog du sprint 4

V.2.3 Décomposition du sprint 4

Tout au long de cette période, nous traitons les fonctionnalités de ce sprint pour produire un incrément potentiellement livrable. La figure [V.2] représente la décomposition du sprint 4 en termes de temps.

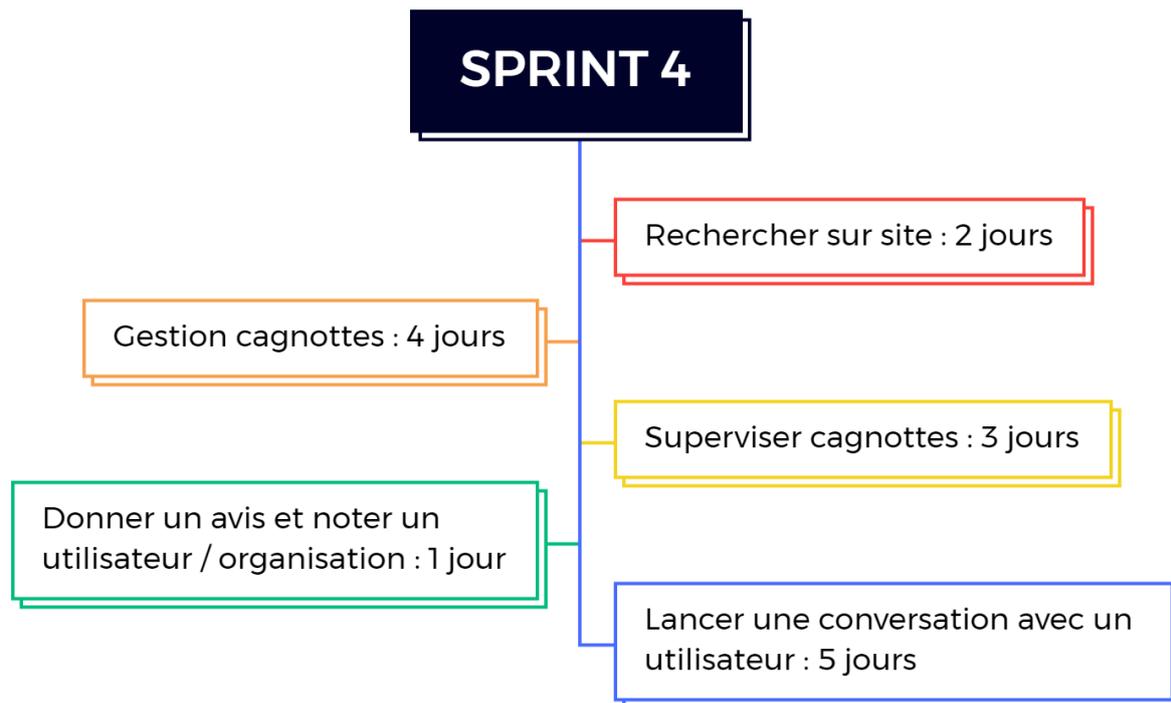


Figure V.2 : Décomposition du sprint 4

V.2.4 Présentation des diagrammes de séquence

Nous allons décrire d'une manière détaillée fonctionnalités :

- « Contribuer aux cagnottes »
- « Contact avec un utilisateur (Messagerie) »
- « Rechercher sur le site »

a. Fonctionnalité « Contribuer aux cagnottes »

L'utilisateur accède à la liste des cagnottes disponibles sur l'application. En parcourant la liste, l'utilisateur sélectionne une cagnotte à laquelle il souhaite contribuer. Le système affiche les détails de la cagnotte sélectionnée, tels que le bénéficiaire, l'objectif financier, la description de la cause, etc. Parmi les options disponibles, l'utilisateur choisit l'option "Fait un don" pour participer à la cagnotte.

Le système affiche un formulaire de paiement, comprenant des champs à remplir, L'utilisateur remplit les champs requis du formulaire de paiement avec les informations demandées.

Une fois que l'utilisateur a rempli tous les champs nécessaires, le système effectue les vérifications nécessaires pour s'assurer de la validité des informations fournies.

Si toutes les informations sont valides, le système enregistre la contribution de l'utilisateur à la cagnotte. Une confirmation de contribution est affichée à l'utilisateur, lui indiquant que sa contribution a été enregistrée avec succès.

Si certaines informations sont invalides ou manquantes, le système affiche un message d'erreur indiquant à l'utilisateur de corriger les champs non conformes. L'utilisateur est invité à remplir à nouveau les informations invalides ou manquantes jusqu'à ce que toutes les données saisies soient valides.

La figure [V.3] représente le diagramme de séquence du cas d'utilisation « Contribuer aux cagnottes »

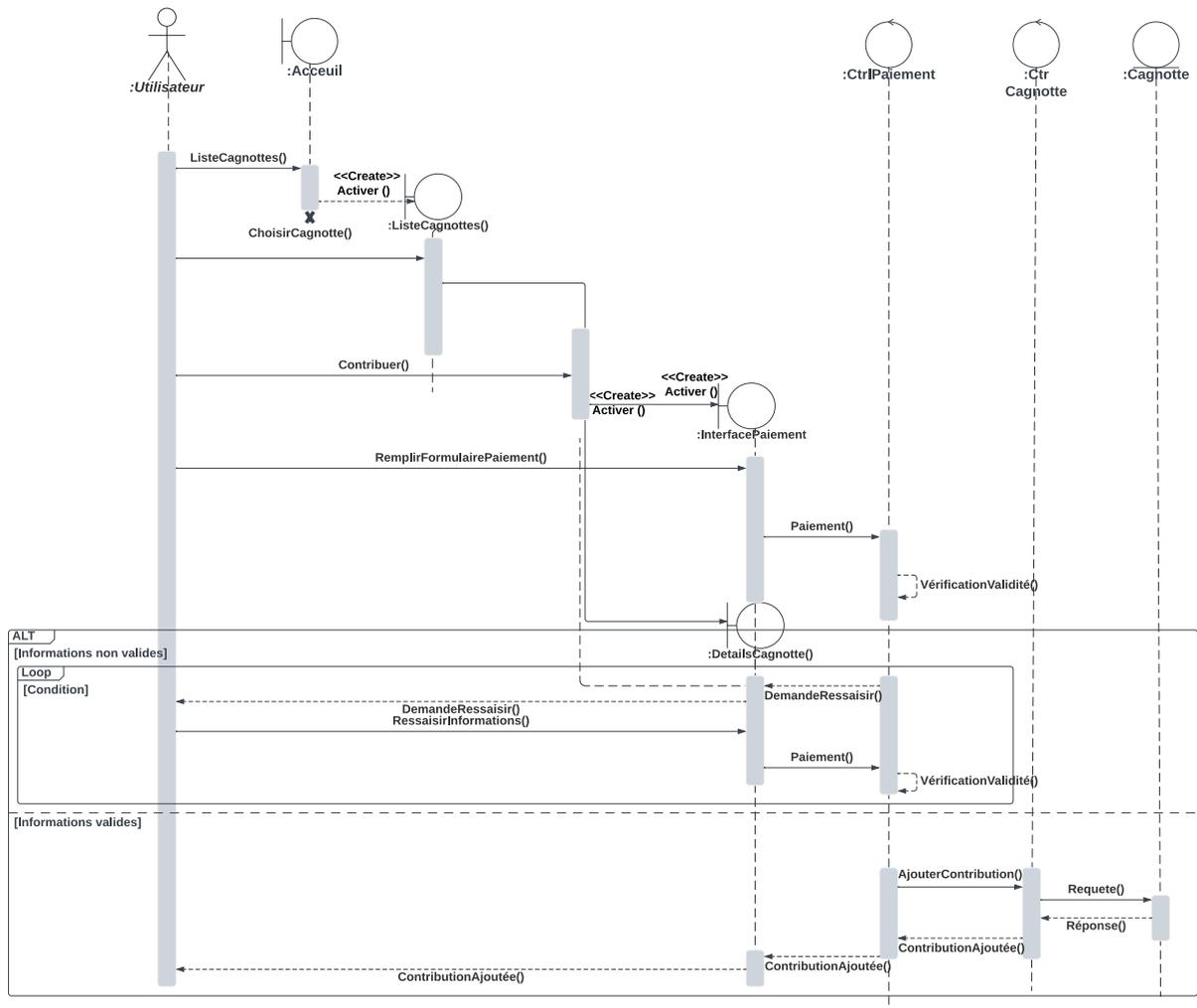


Figure V.3 : Diagramme de séquence de la fonctionnalité « Contribuer aux cagnottes »

b. Fonctionnalité « Contact un utilisateur « Messagerie » »

Dans son compte, l'utilisateur accède à la section "Mes messages", Le système affiche la liste des conversations par l'utilisateur, avec des informations telles que l'expéditeur et l'objet et la date du message. L'utilisateur peut sélectionner une conversation

spécifique pour la consulter en détail. Le contenu de la conversation, y compris les messages, est affiché à l'utilisateur. L'utilisateur a la possibilité de répondre aux messages. Une fois que l'utilisateur a rempli tous les champs nécessaires et rédigé son message, il peut sélectionner l'option "Envoyer" ou un bouton similaire. La figure [V.4] représente le diagramme de séquence de la fonctionnalité « Messagerie »

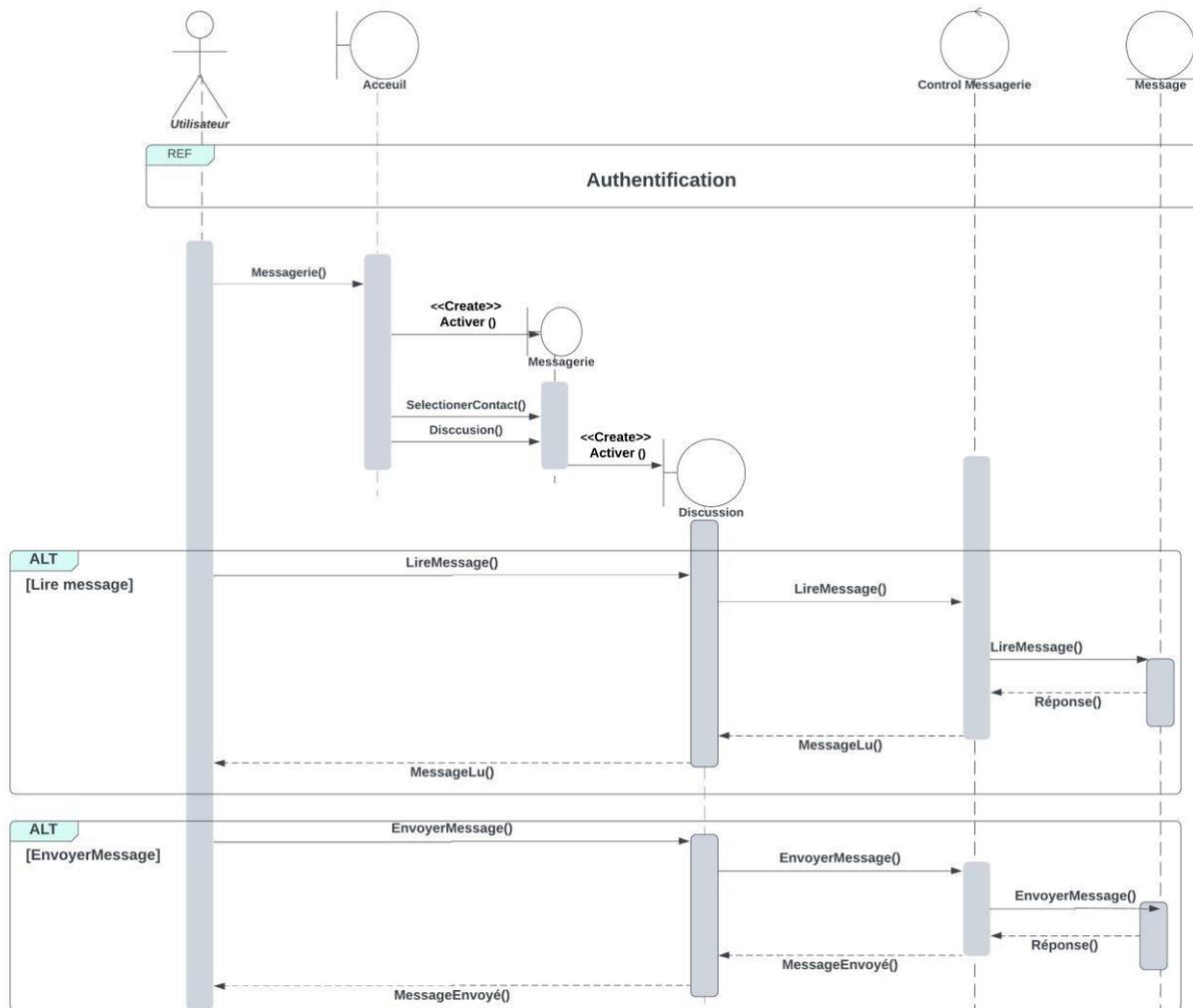


Figure V.4 : Diagramme de séquence de la fonctionnalité « Contact entre utilisateurs »

c. Fonctionnalité « Rechercher sur le site »

L'utilisateur utilise la barre de recherche située sur la plateforme pour effectuer une recherche. Il saisit un ou plusieurs mots-clés ou une requête spécifique dans la barre de recherche.

Une fois que la requête de recherche est saisie, l'utilisateur appuie sur la touche "Entrer" ou sélectionne l'icône de recherche pour lancer la recherche. Le système analyse la requête de recherche et effectue une recherche dans la base de données. Les résultats pertinents de la recherche sont affichés à l'utilisateur, Si aucun résultat ne correspond

à la requête de recherche de l'utilisateur, le système affiche un message indiquant l'absence de résultats pertinents. La figure [V.5] représente le diagramme de séquence de la fonctionnalité « Rechercher sur le site »

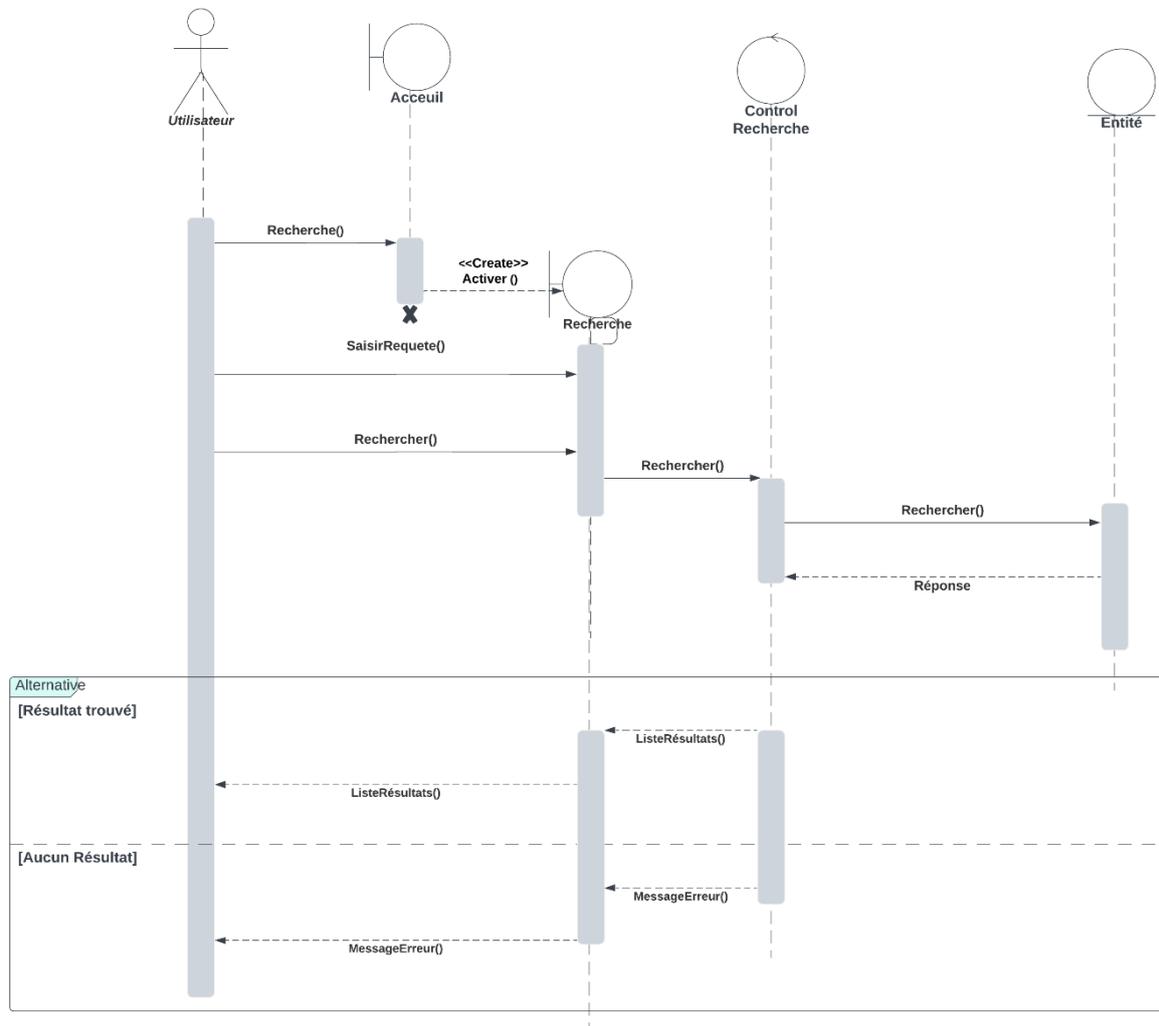


Figure V.5 : Diagramme de séquence de la fonctionnalité « Rechercher sur le site »

V.2.5 Diagramme de classe du sprint 4

Le diagramme de classe de la figure [V.6] décrit d'une façon globale les classes composant les items du sprint 4

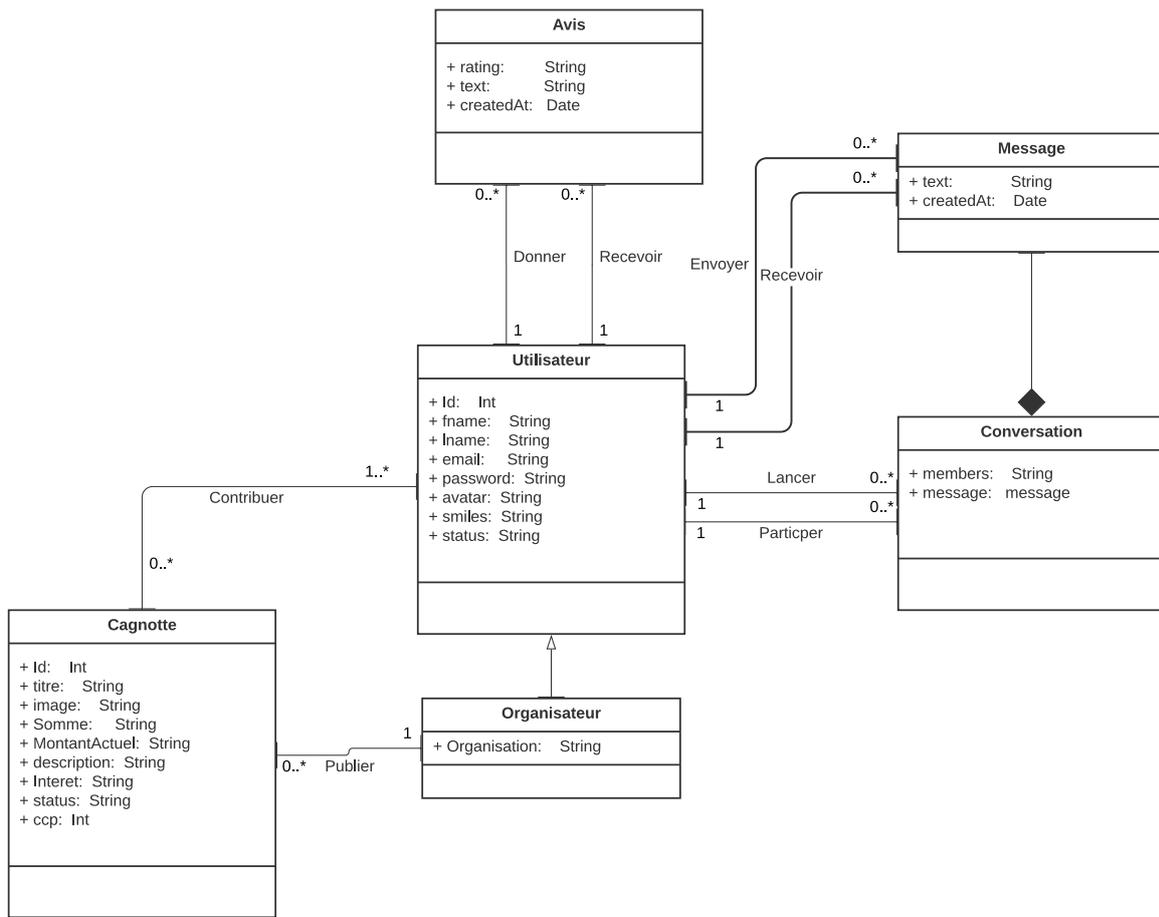


Figure V.6 : Diagramme de classe du sprint 4

V.2.6 Vue des interfaces utilisateur

a. L'interface Gestion de cagnottes

L'interface de gestion de cagnottes (figure [V.7]) permet aux organisateurs de créer, gérer et superviser des cagnottes en ligne.

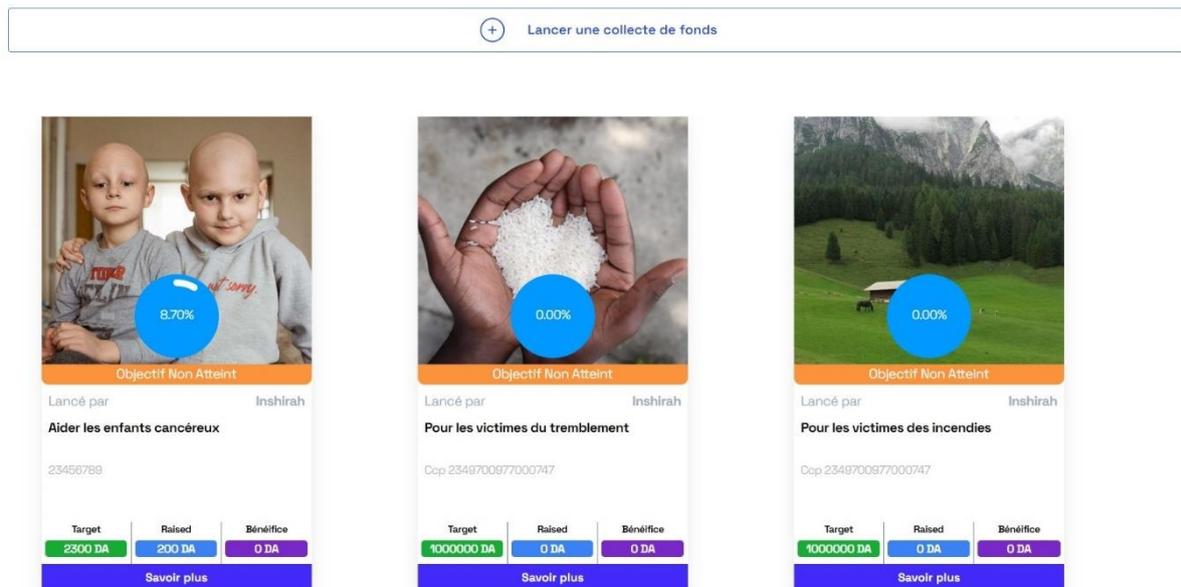


Figure V.7: Page Gestion de cagnottes

b. Interface de contribution aux cagnottes

L'interface de contribution aux cagnottes (figure [V.8]) permet aux utilisateurs de participer activement en faisant des dons à des cagnottes existantes.

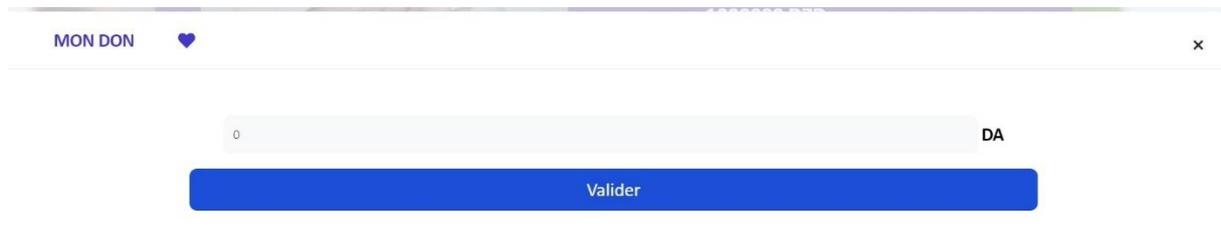


Figure V.8: Interface de contribution aux cagnottes

c. Interfaces de contact entre utilisateurs

Les interfaces de contact entre utilisateurs (figure [V.9]) et [V.10]) facilitent la communication et les interactions entre les utilisateurs de l'application en offrant l'envoi de messages.

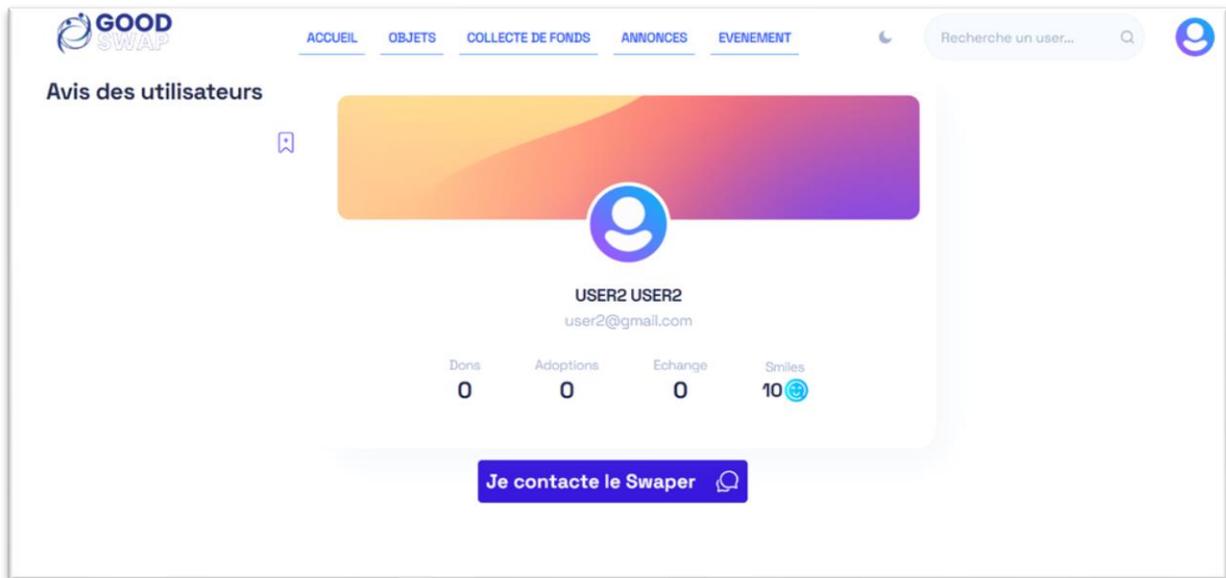


Figure V.9 : Interface Vue profile d'un utilisateur par d'autres utilisateurs

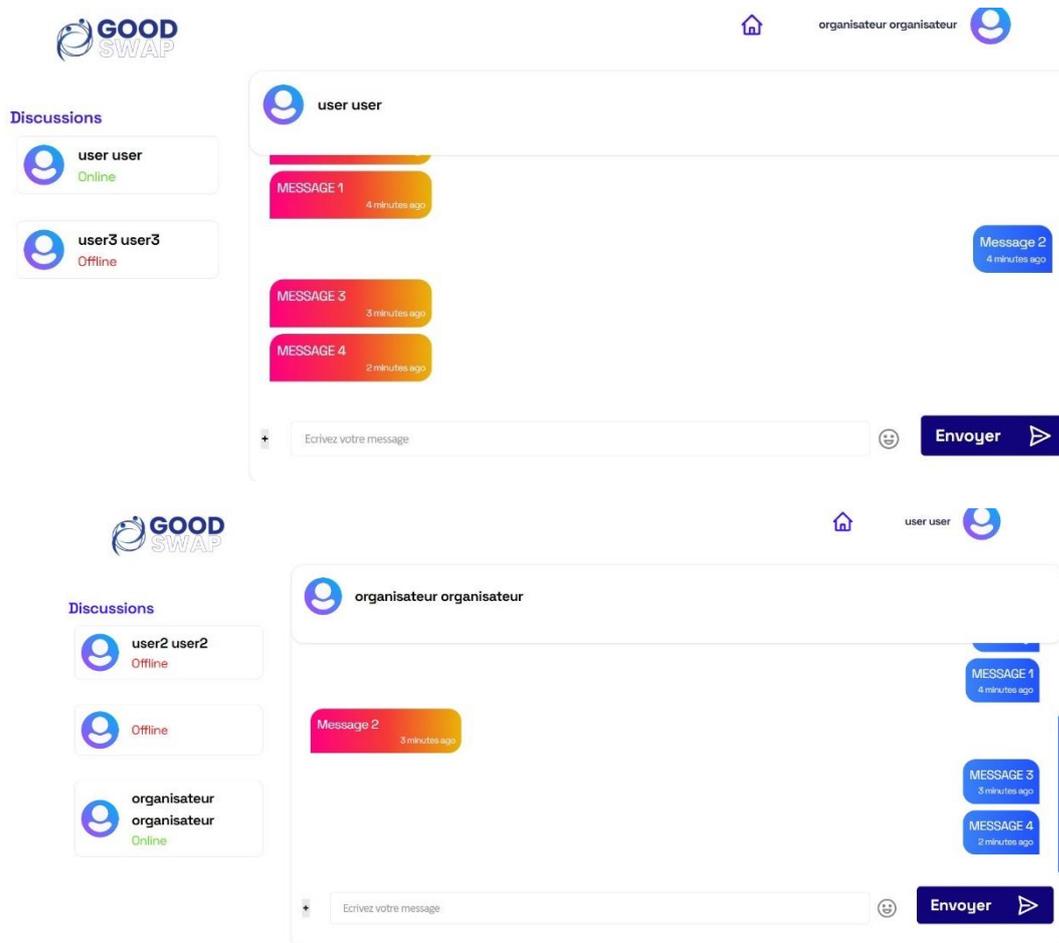


Figure V.10 : Interfaces de contact entre utilisateurs « messagerie »

d. Interface de gestion d'avis

L'interface de gestion d'avis (figure [V.11]) offre aux utilisateurs les outils nécessaires pour gérer et administrer les avis au sein de l'application.

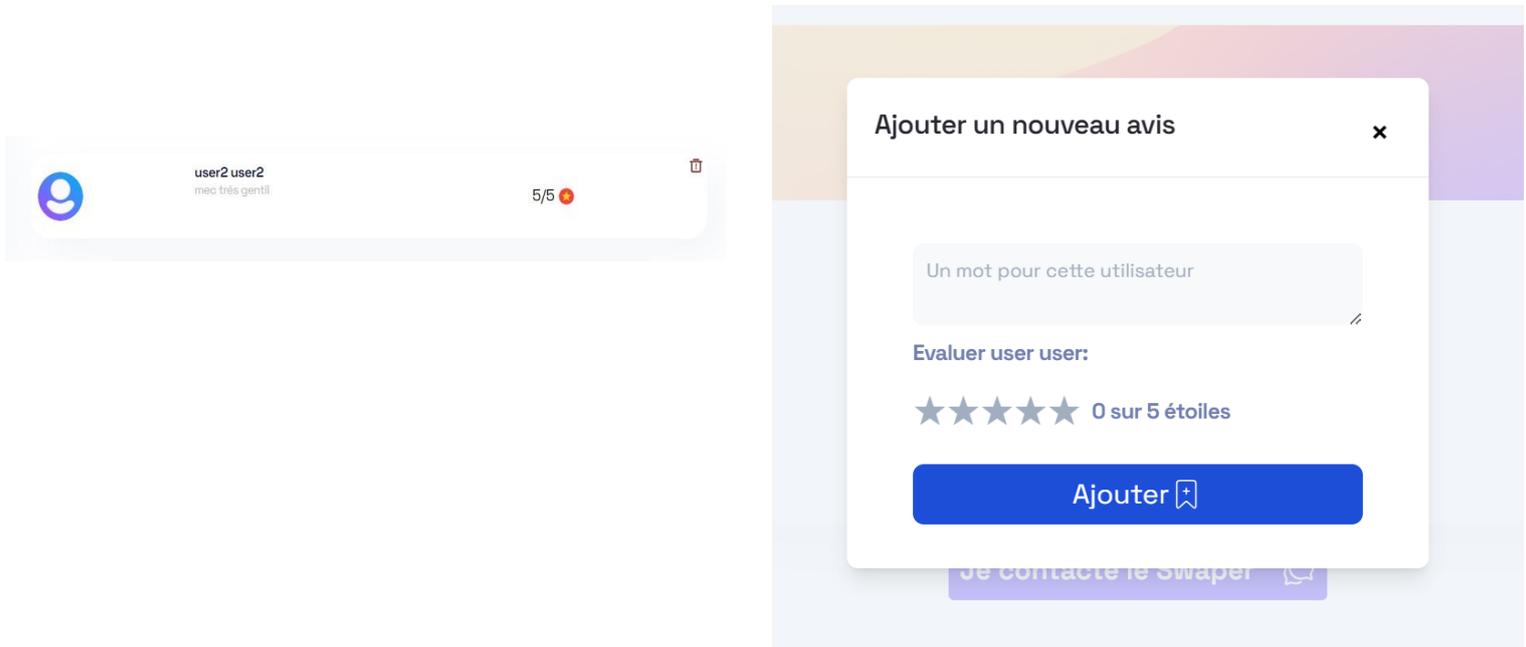


Figure V.11 : Interface de gestion d'avis

e. Interface de recherche

L'interface de recherche (figure [V.12]) offre aux utilisateurs un moyen efficace de trouver rapidement et facilement les publications disponibles sur l'application.

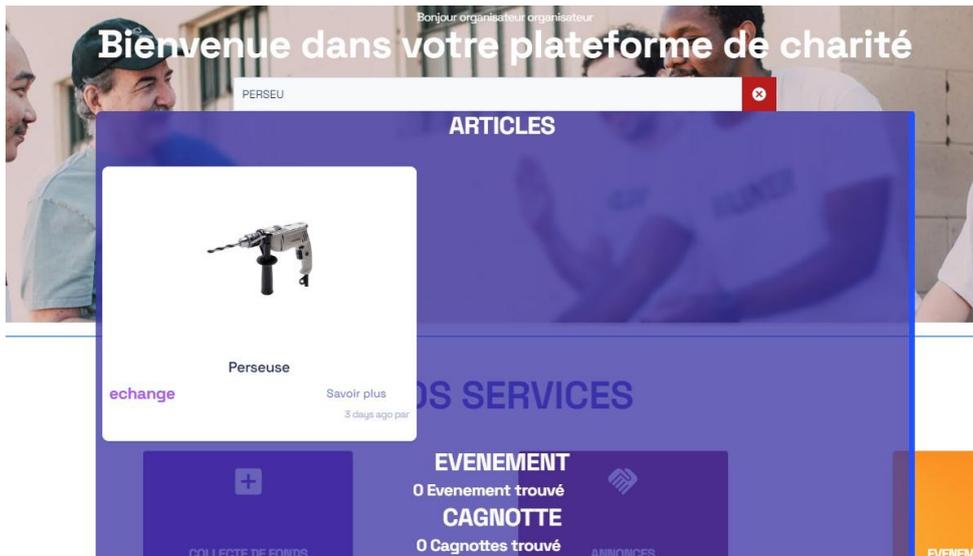


Figure V.12 : Interface de recherche

V.3 Sprint 5

Le cinquième sprint, d'une durée de deux semaines est dédié à la finalisation et à l'optimisation des fonctionnalités existantes, ainsi qu'à l'ajout de nouvelles fonctionnalités pour enrichir davantage notre projet.

V.3.1 Diagramme de cas d'utilisation Sprint 5

Le diagramme de cas d'utilisation de la figure [V.13] décrit d'une façon globale les cas d'utilisations composant les items du sprint 5.

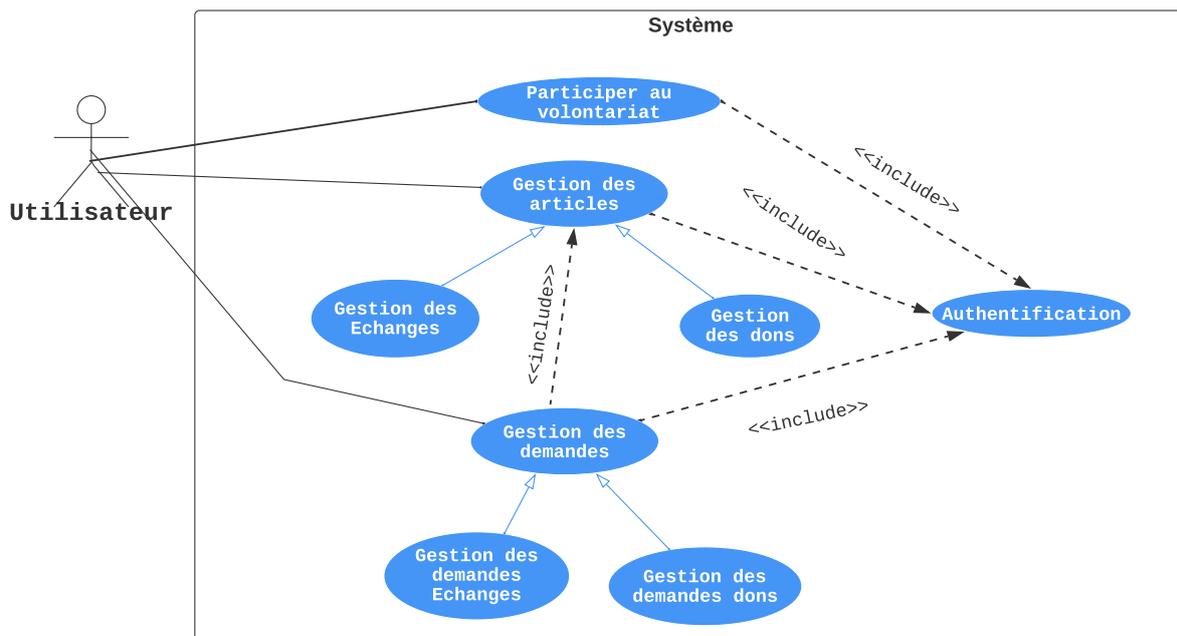


Figure V.13 : Diagramme de cas d'utilisation sprint 5

V.3.2 Sprint Backlog

Tout d’abord on va présenter le backlog du sprint 5 qui contient l’ensemble des tâches à réaliser et leurs estimations qui figurent dans le tableau [V.2] :

Fonctionnalité	Tache	Responsable	Estimation d’efforts (Par heure)	Priorité	Statut
Gestion des échanges	Réaliser les formulaires de création et modification des Echanges	Hamis	8	Haute	A faire
	Implémenter la logique de gestion des échanges	Braham	12	Haute	A faire
Gestion de demandes des échanges	Créer le bouton proposer un échange + Formulaire de proposer un échange	Hamis	8	Haute	A faire
	Implémenter la logique de proposer un échange	Braham	13	Haute	A faire
Participer à un volontariat	Créer le bouton participer au volontariat	Hamis	6	Haute	A faire
	Implémenter la logique de participer au volontariat	Braham	7	Haute	A faire

Tableau V.2 : Backlog du sprint 5

V.3.3 Décomposition du sprint 5

Tout au long de cette période, nous traitons les fonctionnalités ce sprint pour produire un incrément potentiellement livrable. La figure [V.14] représente la décomposition du sprint 5 en termes de temps.

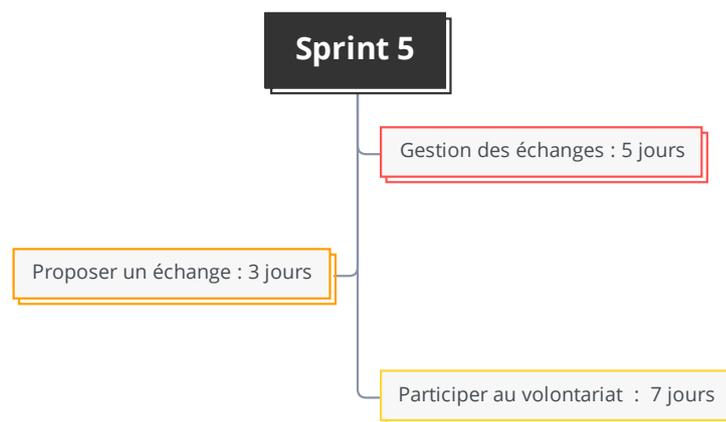


Figure V.14 : Décomposition du sprint 5

V.3.4 Présentation des diagrammes de séquence

Nous allons décrire d'une manière détaillée fonctionnalités :

« Participer à volontariat »

« Gestion des échanges »

« Demander un échange »

a. Fonctionnalité « Participer à un volontariat »

L'utilisateur accède à la liste des annonces de volontariat disponibles sur l'application. En parcourant la liste, il sélectionne une opportunité qui l'intéresse et souhaite y participer, le système affiche les détails de l'annonce sélectionnée, parmi les options disponibles, l'utilisateur choisit l'option "S'inscrire", Le système affiche un bouton d'inscription, l'utilisateur clique sur le bouton d'inscription, le système affiche un message de confirmation une fois que l'utilisateur confirme l'opération le système vérifie si l'utilisateur est déjà inscrit au volontariat.

Le système enregistre l'inscription de l'utilisateur au volontariat. Une confirmation d'inscription est affichée à l'utilisateur, lui indiquant que son inscription a été

enregistrée avec succès. La figure [V.15] représente le diagramme de séquence de la fonctionnalité « participer à un volontariat »

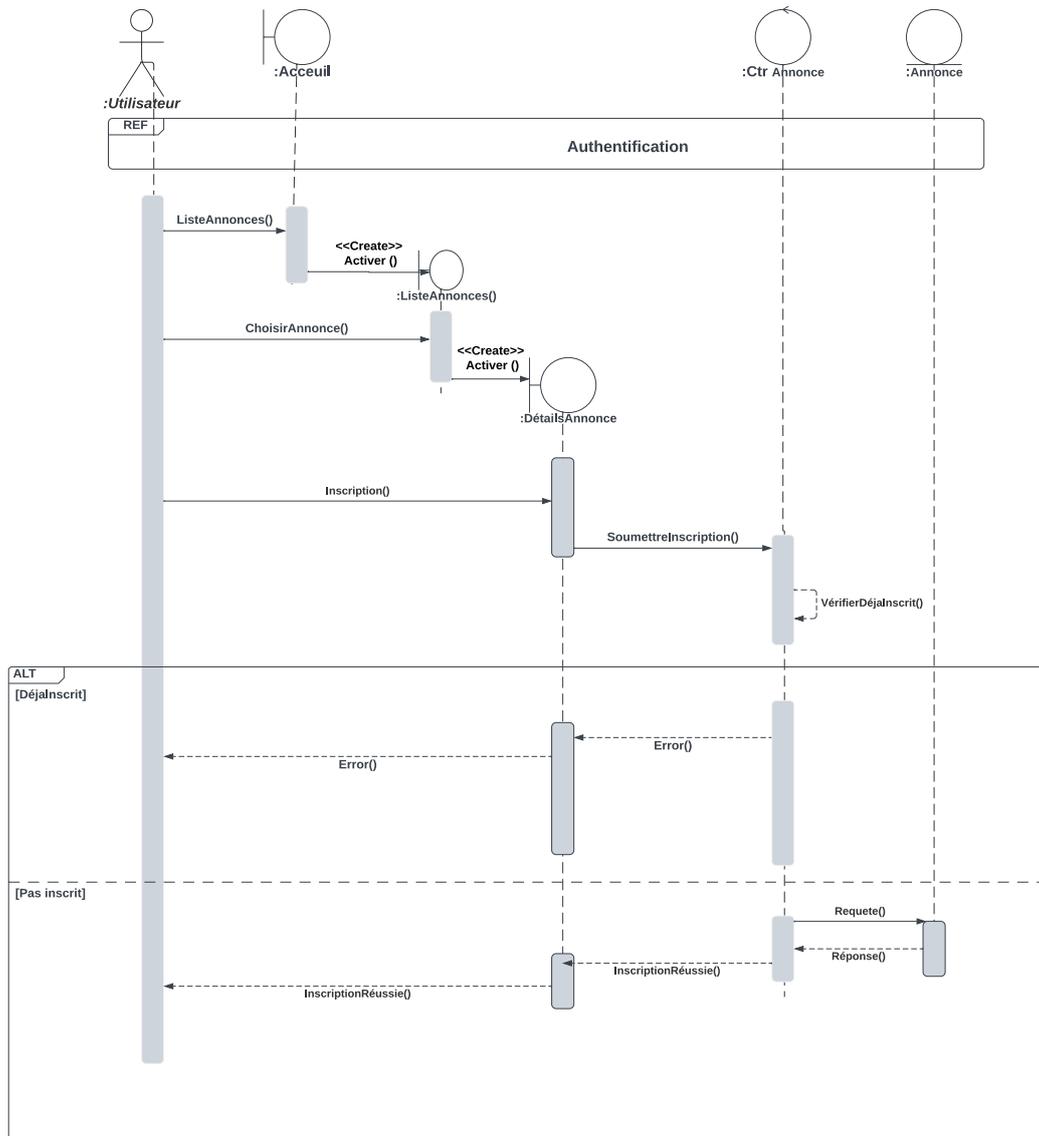


Figure V.15 : Diagramme de séquence de la fonctionnalité participer aux volontariats

b. Fonctionnalité « Demander un échange »

L'utilisateur accède à la page des échanges, il choisit un objet qu'il veut demander d'échanger. Le système affiche les détails du l'objet choisi, l'utilisateur sélectionne l'option de demande associée à cet objet. Le système vérifie si l'utilisateur à des objets disponible pour effectuer un échange avec,

Si oui le système affiche les objets disponibles de l'utilisateur, ce dernier sélectionne un seul objet, puis l'utilisateur valide sa demande

La demande sera enregistrée et l'utilisateur reçoit un message de succès. La figure [V.16] représente le diagramme de séquence de la fonctionnalité « Demander un échange »

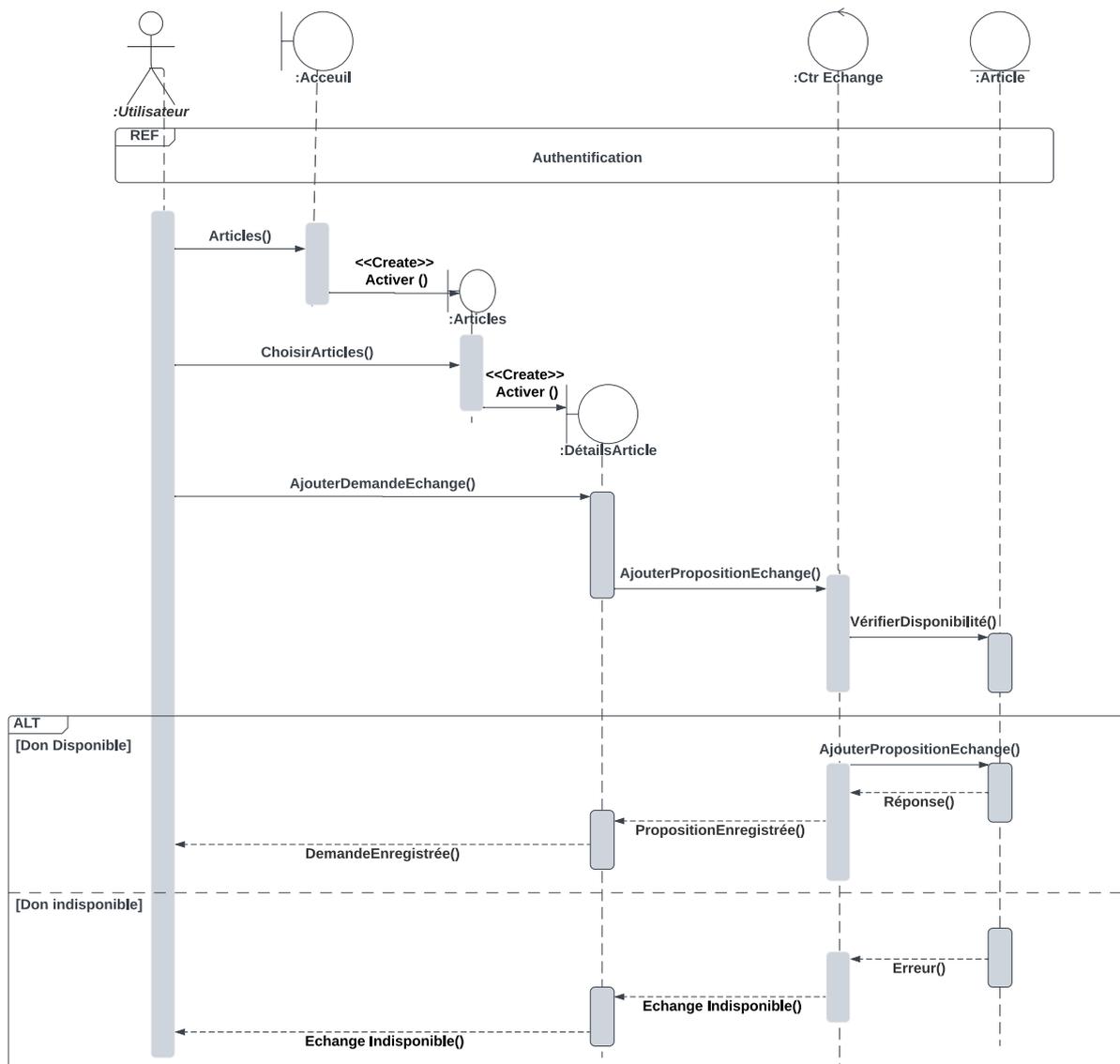


Figure V.16 : Diagramme de séquence de la fonctionnalité « Demander un échange »

V.3.5 Diagramme de classe du sprint 5

Le diagramme de classe de la figure [V.17] décrit d'une façon globale les classes composantes du sprint 5

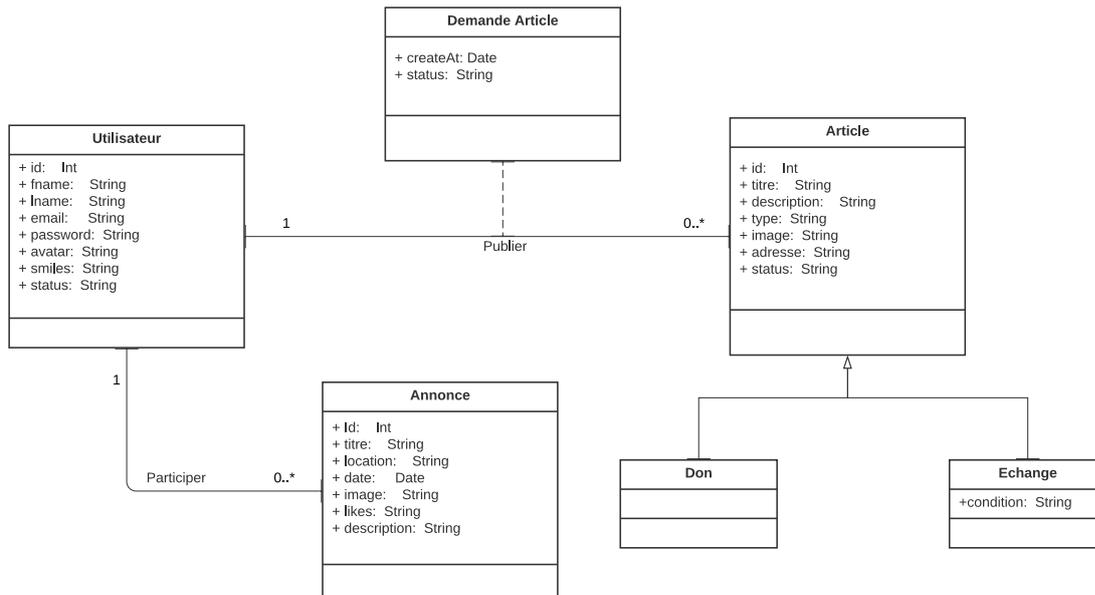


Figure V.17 : Diagramme de classe sprint 5

V.3.6 Vue des interfaces utilisateur

a. Interface demande d'un échange

L'interface de demande d'un échange (figure [V.18]) offre aux utilisateurs la possibilité d'initier des demandes d'échange au sein de l'application.

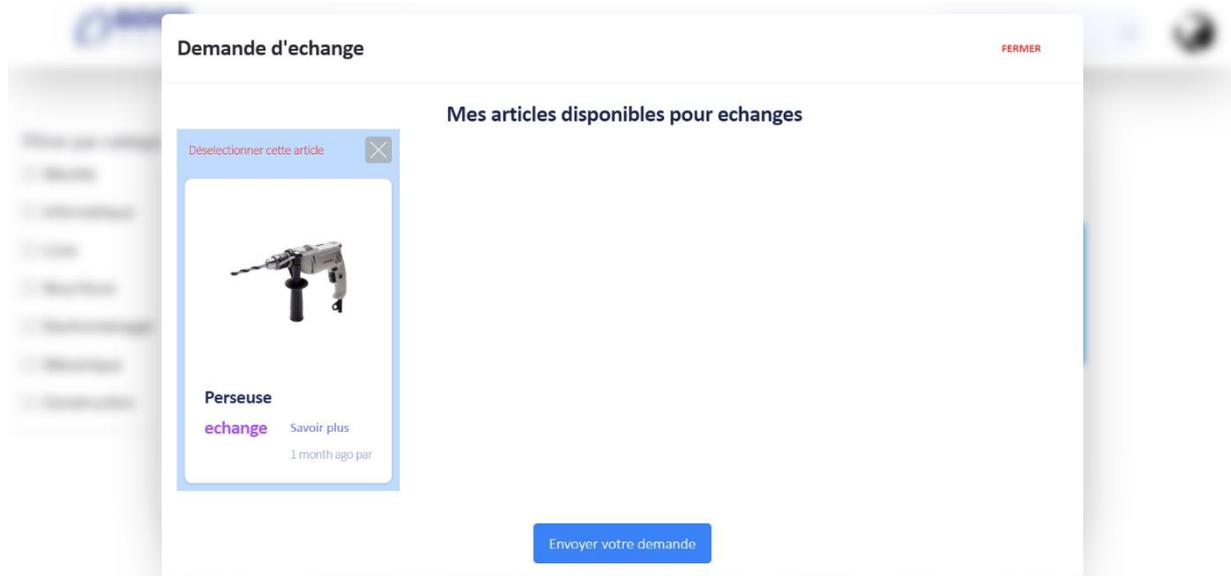


Figure V.18: Interface demande d'un échange

V.4 Sprint 6

Le sixième sprint, d'une durée de quatre semaines, sera entièrement dédié à la version mobile de notre projet.

V.4.1 Sprint Backlog

Tout d'abord on va présenter le backlog du sprint 6 qui contient l'ensemble des tâches à réaliser et leurs estimations qui figurent dans le tableau [V.3] :

Fonctionnalité	Tache	Responsable	Estimation d'efforts (Par heure)	Priorité	Statut
Réaliser une version mobile	Conception des interfaces utilisateur	Hamis	10	Moyenne	A faire
	Développement et intégration des interfaces utilisateur	Hamis	35	Haute	A faire
	Tests et corrections	Hamis	5	Basse	A faire
	Implémentation de la logique coté serveur	Braham	40	Haute	A faire
	Tests et corrections	Braham	10	Basse	A faire

Tableau V.3 : Backlog du sprint 6

V.4.2 Décomposition du sprint 6

Tout au long de cette période, nous traitons les fonctionnalités de ce sprint pour produire un incrément potentiellement livrable. La figure [V.19] représente la décomposition du sprint 6 en termes de temps.

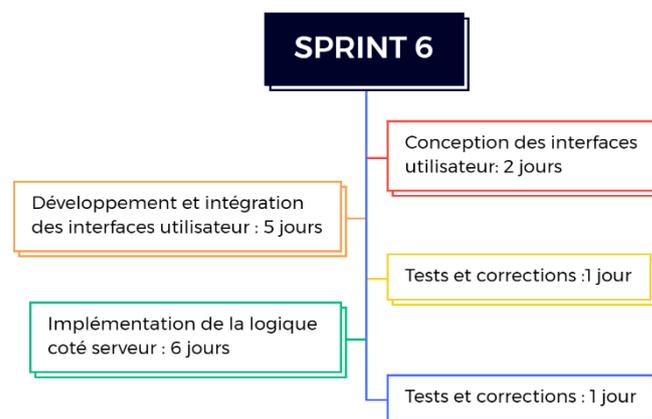


Figure V.19 : Décomposition du sprint 6

V.4.3 Présentation des interfaces utilisateur

Nous allons présenter les interfaces de notre application version mobile qui porte les fonctionnalités et les mêmes utilisations.

a. Interfaces « Accueil & Inscription et Authentification »

La figure [V.20] représente les interfaces « Accueil & Inscription & Authentification »

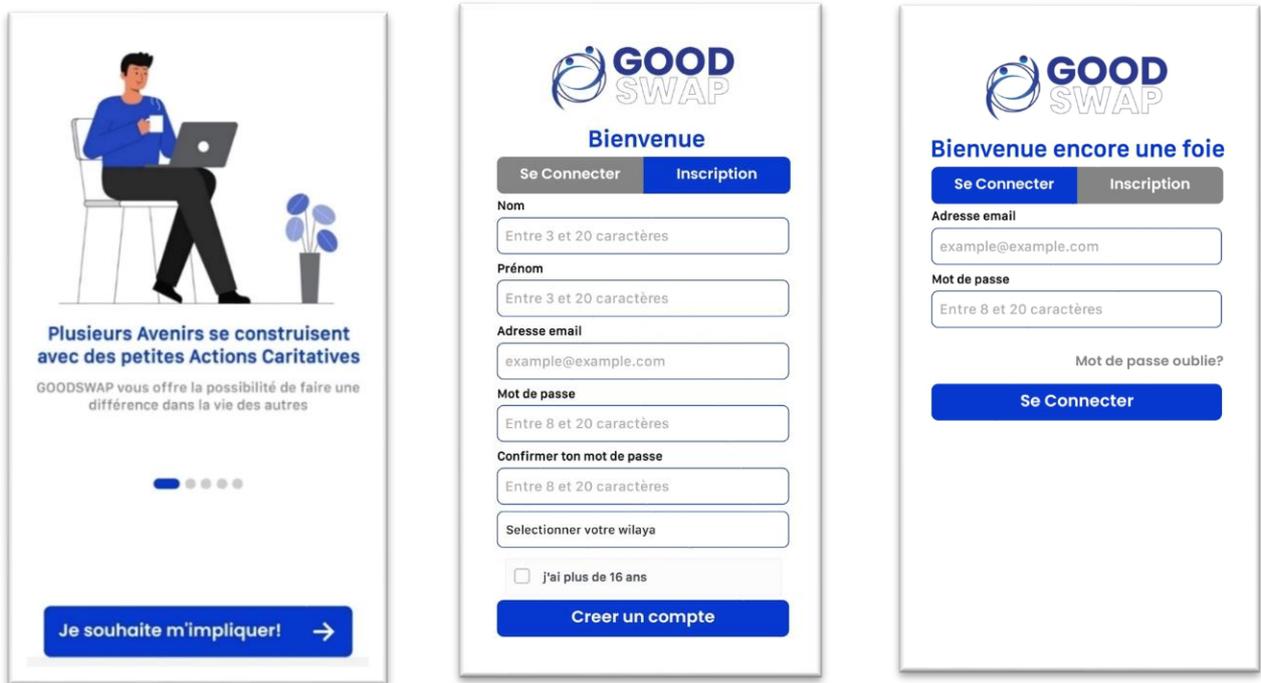


Figure V.20 : Interface « Accueil & Inscription & Authentification »

b. Interfaces « Utilisateur Accueil & articles & Détails d'article »

La figure [V.21] représente les interfaces « Utilisateur Accueil & Articles & Détails d'article »

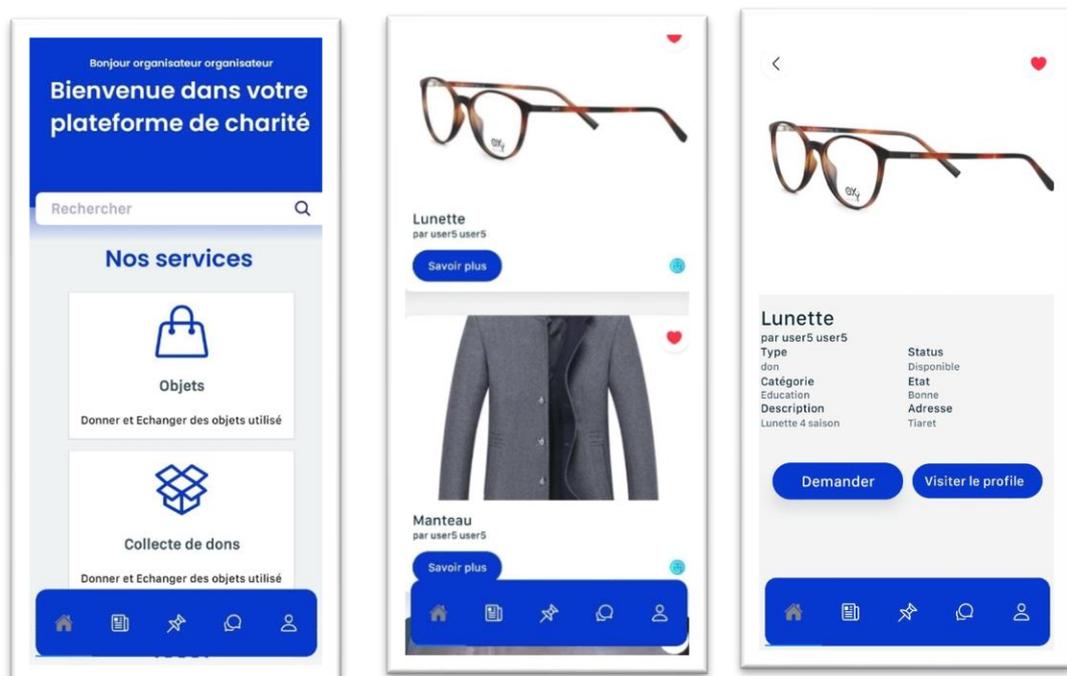


Figure V.21 : Interfaces « Utilisateur Accueil & Articles & Détails article »

c. Interfaces « Gestion des demandes & profile organisation & profile utilisateur »

La figure [V.22] représente les interfaces « Gestion des demandes & profile organisation & profile utilisateur »

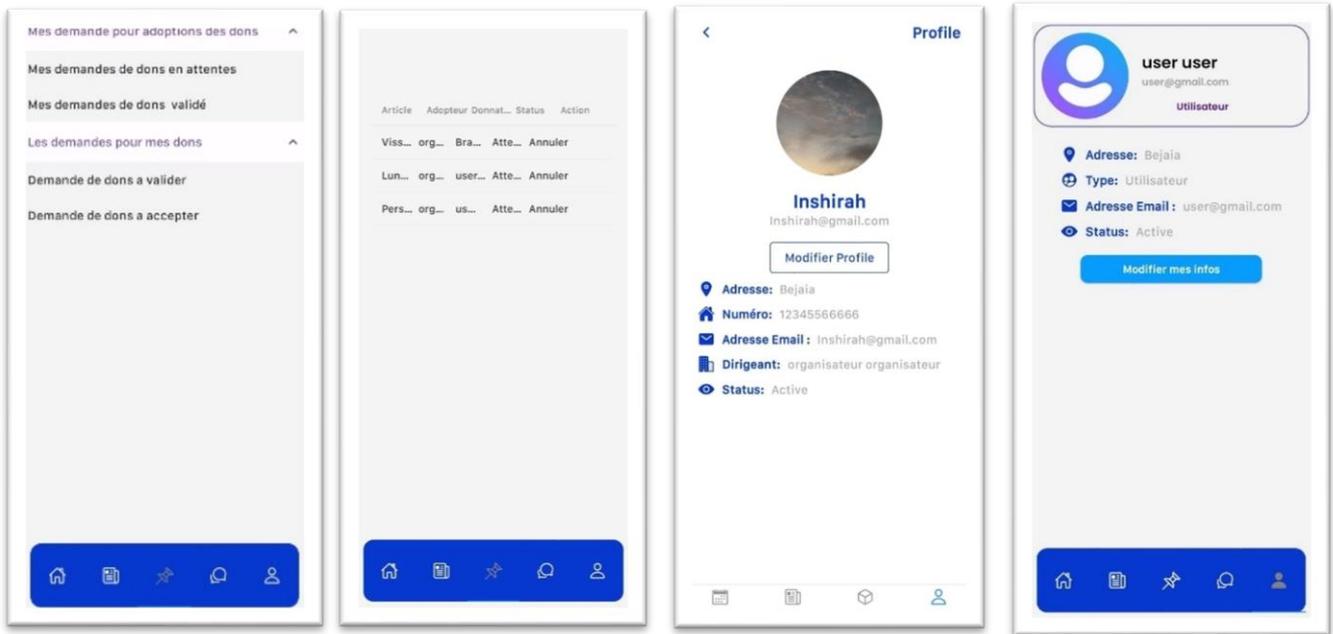


Figure V.22 : Interfaces « Gestion de demandes Profile Organisation et profil utilisateur »

d. Interfaces « Gestion des événements & des annonces »

La figure [V.23] représente les interfaces « Gestion des événements & des annonces & des cagnottes »

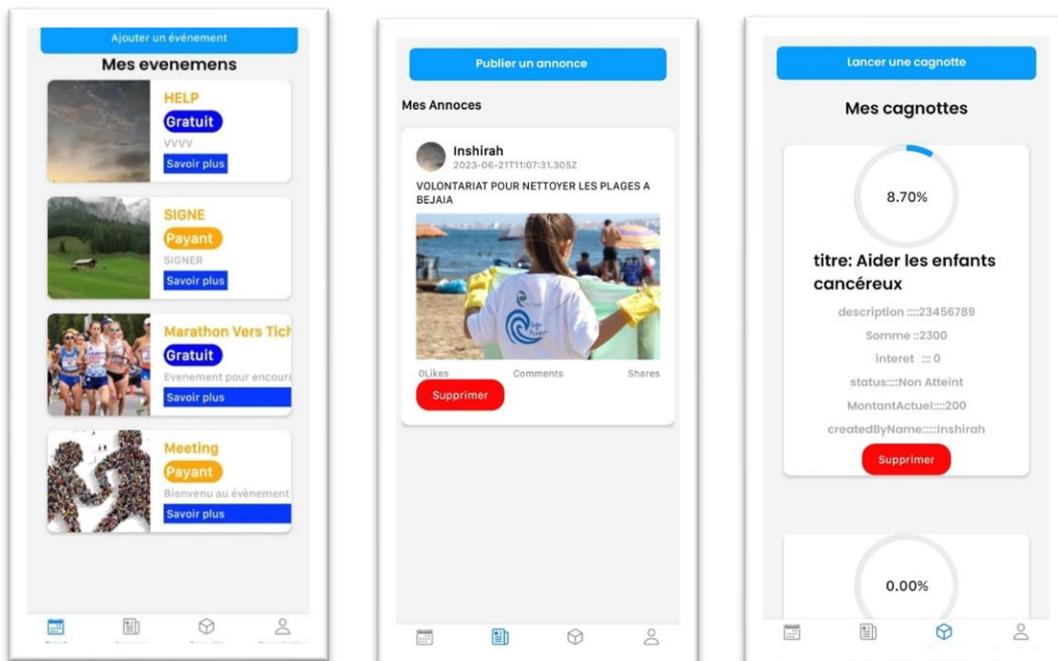


Figure V.23 : Interfaces « Gestion des Evènements, Annonces, Cagnottes »

e. Interfaces « Contact entre utilisateur « Messagerie » »

La figure [V.24] représente les interfaces « Contact entre utilisateur Emetteur & Récepteur & leur conversation »

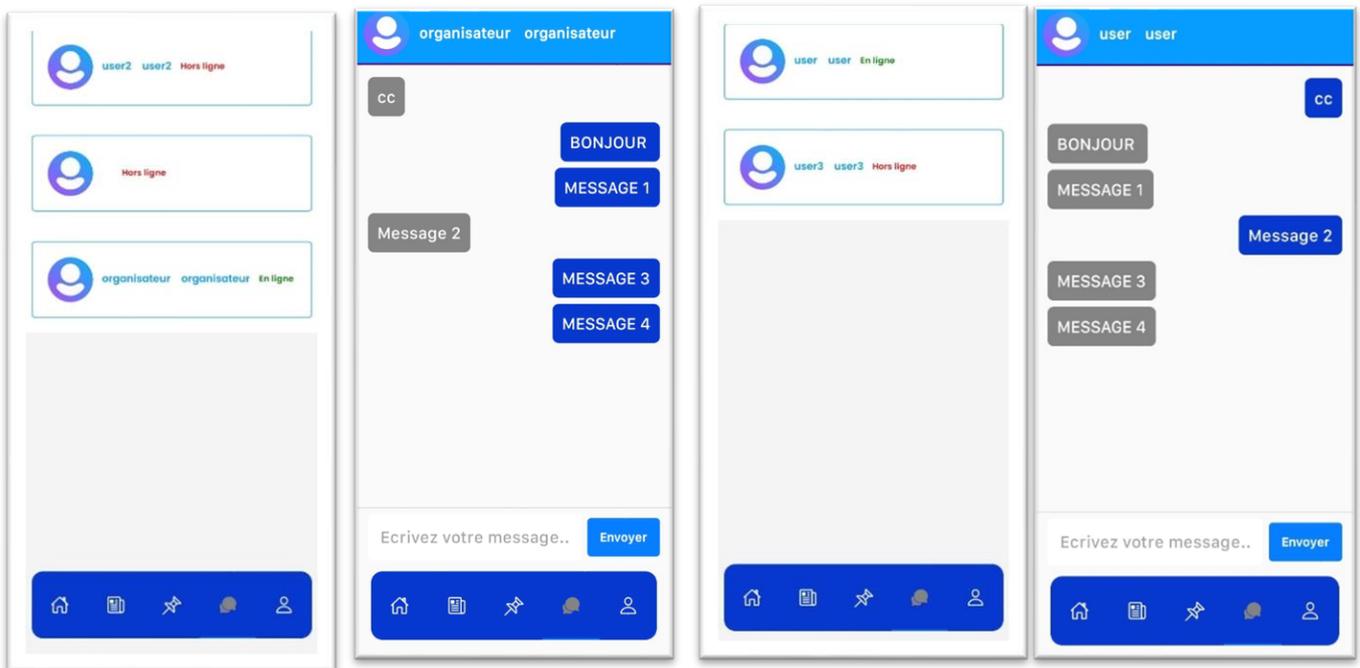


Figure V.24 : Interfaces Contacte utilisateur (Messagerie)

f. Interfaces « Ajouter un article, un événement »

La figure [V.25] représente les interfaces « Contact entre utilisateur Emetteur & Récepteur & leur conversation ».

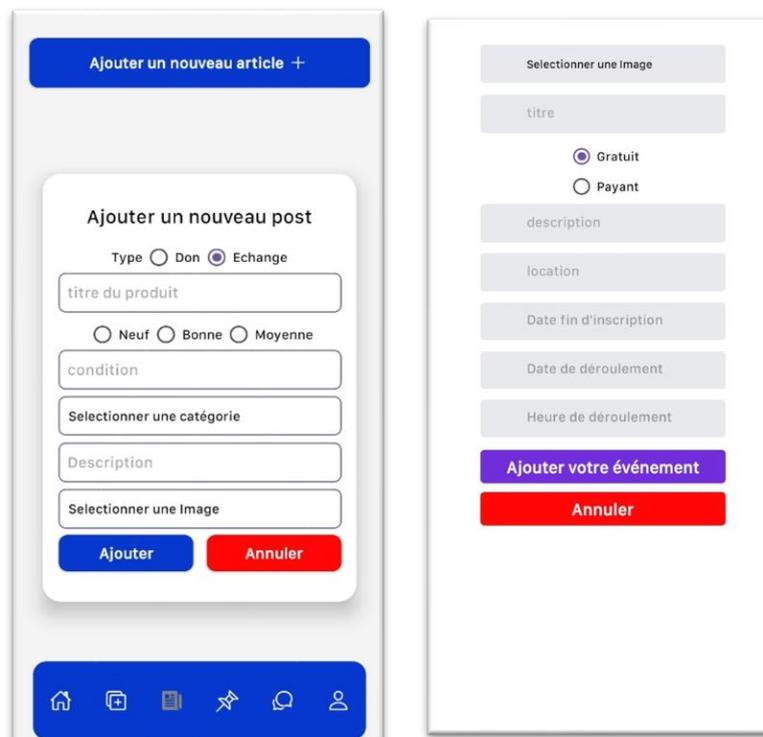


Figure V.25 : Interface d'ajout « Article, Evènement »

V.5 Conclusion

Dans ce chapitre, nous avons présenté en détail les trois sprints : Sprint 4, Sprint 5 et Sprint 6, ainsi que les fonctionnalités développées, les estimations d'efforts et les résultats obtenus jusqu'à présent dans le cadre de notre projet. Les diagrammes de cas d'utilisation, séquence, et de classe, ainsi que les captures d'écran des interfaces utilisateur ont offert une vue claire des différentes fonctionnalités que nous avons implémentées.

Conclusion générale et perspectives

Notre projet de fin d'étude a abouti à la création d'une application dédiée à la philanthropie qui répond aux besoins identifiés et offre des opportunités concrètes d'aide aux personnes dans le besoin.

Pour mener à bien notre projet, nous avons adopté une approche méthodique structurée en différents chapitres. Le premier chapitre a présenté en détail notre projet, ses objectifs et son fonctionnement, ainsi que les méthodologies de développement que nous avons utilisées. Le deuxième chapitre s'est concentré sur la planification des sprints, décrivant comment nous avons organisé notre travail par étapes successives. Les chapitres suivants ont détaillé les différentes phases de développement de notre application, en mettant l'accent sur les résultats obtenus lors des sprints spécifiques. Nous avons suivi une approche itérative, en apportant des améliorations et des ajustements en fonction des retours d'utilisateurs et des besoins identifiés.

Cependant, il reste encore des perspectives d'amélioration et d'expansion pour l'application, notamment en termes de fonctionnalités supplémentaires, d'élargissement de la base d'utilisateurs et de collaboration avec des organisations philanthropiques.

Nous envisageons ainsi d'ajouter de nouvelles fonctionnalités telles que :

- Améliorer l'application mobile en termes d'interface graphique.
- Mettre des plans d'abonnements pour la monnaie de l'application (les smiles).
- Ajouter un module d'aide à la décision pour les organisations pour la bonne gestion des publications et améliorer leur audience.
- Ajouter un système de gestion de logistique.
- L'ajout d'un module de géolocalisation qui va permettre de localiser les utilisateurs à proximité.
- Permettre aux organisations de créer des collectifs.
- Sécuriser et améliorer notre système de messagerie en rajoutant plus de fonctionnalités.

Nous espérons que notre projet contribuera à promouvoir une culture de solidarité continue et à faciliter l'engagement des individus dans des actions philanthropiques, offrant ainsi un soutien précieux aux communautés touchées par les crises et contribuant à un avenir meilleur.

Bibliographie

- [1] O. Amstouts, «le grand retour de troc,» 09 09 2021. [En ligne]. Available: <https://www.lekaba.fr/article/le-grand-retour-du-troc>. [Accès le 12 Avril 2023].
- [2] «Qu'est-ce que le don ? Explications avec infodon,» [En ligne]. Available: <https://infodon.fr/quest-ce-que-le-don/>. [Accès le 18 Mai 2023].
- [3] B. Infos, «Crowdfunding : tout savoir avant de se lancer !,» 23 janvier 2023. [En ligne]. Available: <https://www.economie.gouv.fr/entreprises/crowdfunding-financement-participatif>. [Accès le 15 Avril 2023].
- [4] D. Young, «Software Development Methodologies,» *White paper*, p. 1, 2013.
- [5] M. Véronique, *Gestion de projet Agile avec Scrum , lean , eXtreme programming...*, Eyrolles.
- [6] «What is Agile? | Atlassian,» [En ligne]. Available: <https://www.atlassian.com/agile>. [Accès le 12 Mars 2023].
- [7] C. DRUMOND, «Scrum : de quoi s'agit-il, comment cela fonctionne-t-il et en quoi est-ce génial ?,» [En ligne]. Available: <https://www.atlassian.com/fr/agile/scrum>. [Accès le 12 Avril 2023].
- [8] K. S. & J. Sutherland, *The Scrum Guide*, 2020.
- [9] «Agile Scrum artifacts,» 25 Mai 2023. [En ligne]. Available: <https://www.atlassian.com/agile/scrum/artifacts>. [Accès le 25 Avril 2023].
- [10] O. m. group, *OMG unified modeling language specification*, 2015.
- [11] «Mobile modeling,» 25 Mai 2023. [En ligne]. Available: <https://jmbruel.github.io/MobileModeling2016-1/>. [Accès le 22 Mai 2023].
- [12] «MVC Framework - Introduction,» [En ligne]. Available: https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm. [Accès le 12 Mai 2023].
- [13] P. Roques, *UML 2 , modéliser une application web*, Eyrolles.
- [14] V. Subramanian, *Pro MERN Stack*, 2019.

Schéma de la base de données

Pour migrer vers un modèle non relationnel à partir d'un diagramme de classes, nous avons suivi les étapes suivantes

- Étape 1 : Choix d'un modèle non relationnel

Dans cette étape, nous avons sélectionné le modèle orienté document comme solution appropriée. Ce modèle de base de données non relationnel est spécialement conçu pour stocker des données semi-structurées ou non structurées. Sa flexibilité en termes de schémas, avec chaque document ayant sa propre structure et les collections permettant différents ensembles de champs, correspond parfaitement à la nature de nos données.

- Étape 2 : Conversion des classes en éléments de schéma

Chaque classe du diagramme UML a été transformée en une collection. Les attributs de chaque classe ont été représentés comme des champs à l'intérieur des documents. Ainsi, chaque document correspond à une instance de la classe et contient les données associées à cette instance. Pour gérer l'héritage, chaque classe fille a été représentée par une collection contenant des documents qui héritent des attributs de la classe mère.

- Étape 3 : Gestion des relations

Dans cette étape, nous avons pris en compte les relations entre les classes. Dans les bases de données orientées document, nous avons deux options : incorporer des objets associés à l'intérieur d'un document ou utiliser des références pour lier les documents. Nous avons choisi la méthode la plus appropriée en fonction de la nature des relations et des besoins de notre application.

- Étape 4 : Optimisation du schéma

Nous avons optimisé le schéma en fonction des besoins spécifiques du modèle non relationnel. Cela comprend la dénormalisation des données pour éviter les jointures coûteuses et améliorer les performances des requêtes. De plus, nous avons supprimé les attributs inutiles des documents afin de réduire la taille du schéma et d'optimiser l'espace de stockage.

En suivant ces étapes, nous avons réussi à migrer du modèle relationnel vers un modèle non relationnel en utilisant le modèle orienté document. Cette approche nous offre la flexibilité nécessaire pour gérer nos données de manière efficace et répondre aux besoins spécifiques de notre application.

La figure [V.27] représente le schéma de notre base de données

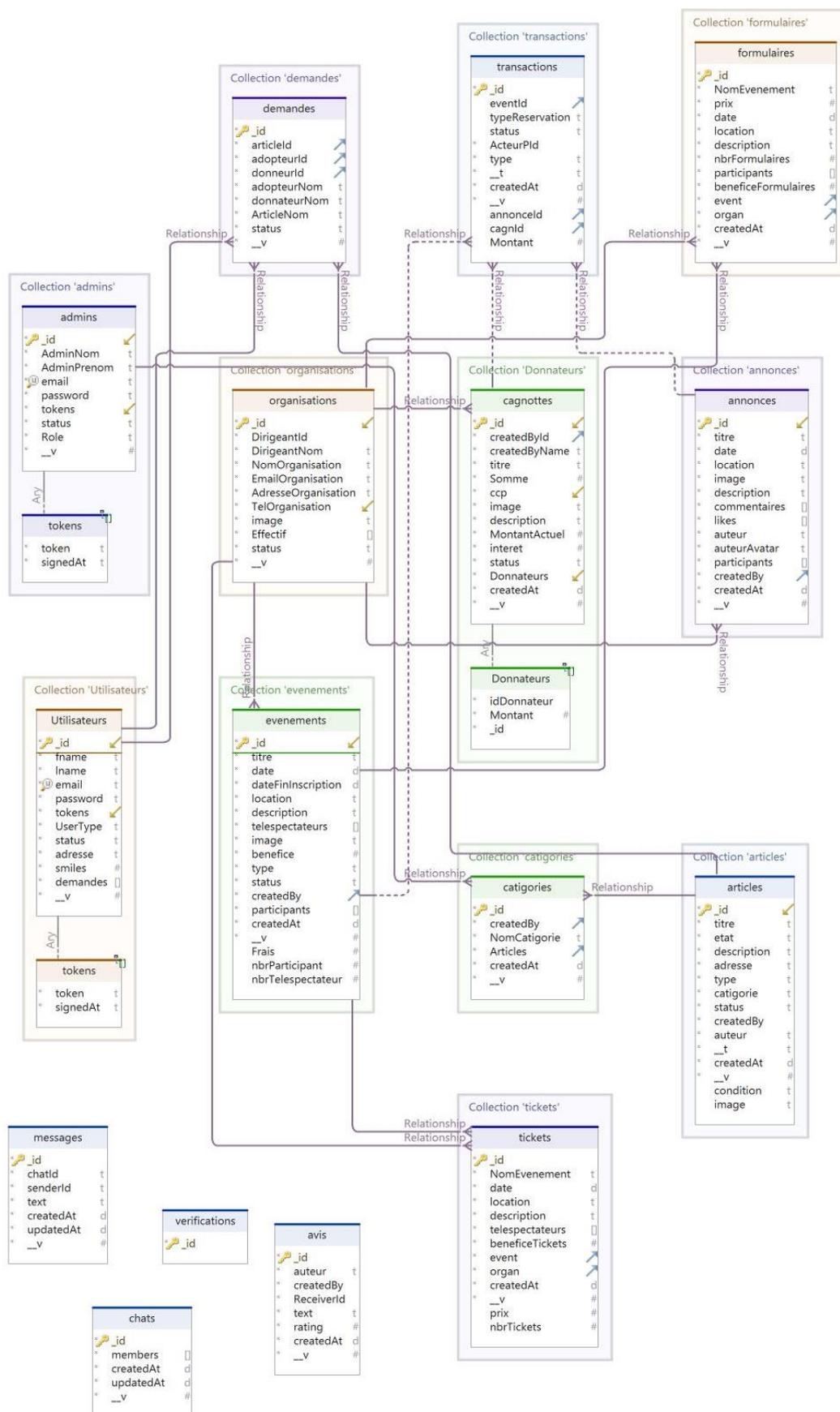


Figure V.27 : Schéma de la base de données

Questionnaire sur les besoins fonctionnels pour l'application

Informations générales sur l'organisation caritative :

1. Nom de l'organisation caritative :
2. Domaine d'action (santé, éducation, environnement, etc.) :
3. Description brève de l'organisation caritative :
4. Adresse et coordonnées :

Objectifs et besoins de l'organisation caritative :

5. Quels sont les principaux objectifs de votre organisation caritative ?
6. Quels types d'activités philanthropiques votre organisation réalise-t-elle actuellement ?
7. Quels sont les besoins fonctionnels que vous souhaitez couvrir avec notre application ?

Gestion des dons :

8. Comment votre organisation caritative gère-t-elle actuellement les dons ?
9. Quelles fonctionnalités de gestion des dons aimeriez-vous voir dans l'application ?
10. Avez-vous besoin de la possibilité d'accepter des dons en ligne ? Si oui, quelles méthodes de paiement préférez-vous utiliser ?

Crowdfunding :

14. Avez-vous déjà utilisé des plateformes de crowdfunding pour financer vos projets ?
15. Quels sont les principaux défis ou besoins que vous avez rencontrés lors de vos campagnes de crowdfunding ?
16. Quelles fonctionnalités liées au crowdfunding aimeriez-vous avoir dans l'application ?

Événements caritatifs :

17. Organisez-vous régulièrement des événements caritatifs ? Si oui, quel type d'événements ?

18. Quelles fonctionnalités relatives aux événements caritatifs aimeriez-vous voir dans l'application ?

Volontariat :

19. Avez-vous des programmes de volontariat ? Si oui, quelles sont les activités de volontariat proposées ?

20. Quelles fonctionnalités relatives au volontariat aimeriez-vous voir dans l'application ?

Intégration avec d'autres systèmes :

21. Utilisez-vous actuellement d'autres systèmes ou outils pour gérer vos activités philanthropiques ?

22. Souhaitez-vous intégrer l'application avec d'autres systèmes existants ? Si oui, lesquels ?

Sécurité et confidentialité :

23. Quels sont vos besoins en matière de sécurité et de confidentialité des données ?

24. Quelles mesures de sécurité doivent être mises en place pour garantir la protection des données des utilisateurs ?

Autres commentaires :

25. Si vous avez d'autres commentaires, suggestions ou besoins spécifiques non mentionnés précédemment, veuillez les indiquer ici.

Merci d'avoir pris le temps de remplir ce questionnaire. Vos réponses nous aideront à développer une application adaptée à vos besoins.

Résumé

La philanthropie est une pratique qui vise à promouvoir le bien-être de l'humanité en soutenant financièrement, en donnant du temps ou en apportant des ressources à des causes sociales, éducatives, environnementales ou humanitaires. Dans le cadre de ce projet, l'objectif principal était de concevoir et de développer une application informatique dédiée aux actions philanthropiques.

Pour garantir le bon déroulement du projet, nous avons choisi de modéliser notre système en utilisant le formalisme UML et le pattern de conception MVC. Pour la méthodologie de développement, nous avons opté pour la méthode SCRUM, qui nous a permis d'organiser notre travail en sprints successifs et de nous adapter aux besoins changeants du projet.

En utilisant la technologie MERN (MongoDB, Express, React, Node.js), nous avons réalisé une application fullstack. Celle-ci comprend des fonctionnalités telles que l'authentification et l'inscription des utilisateurs, la gestion des espaces des organisations, la création et la gestion des événements, ainsi qu'un système de messagerie et d'échange d'objets entre les utilisateurs.

Mots-clés : actions philanthropiques, UML, pattern de conception MVC, méthodologie SCRUM, technologie MERN.

Abstract

Philanthropy is a practice that aims to promote the well-being of humanity by providing financial support, donating time, or contributing resources to social, educational, environmental, or humanitarian causes. In the context of this project, the main objective was to design and develop a computer application dedicated to philanthropic actions.

To ensure the smooth progress of the project, we chose to model our system using the UML formalism and the MVC design pattern. For the development methodology, we opted for the SCRUM method, which allowed us to organize our work in successive sprints and adapt to the changing needs of the project.

Using the MERN technology stack (MongoDB, Express, React, Node.js), we created a full-stack application. This includes features such as user authentication and registration, management of organization spaces, creation and management of events, as well as a messaging system and object exchange between users.

Keywords : philanthropic actions, UML, MVC design pattern, SCRUM methodology, MERN technology.