



INSTITUT SUPÉRIEUR DES ÉTUDES TECHNOLOGIQUES DE NABEUL

المعهد العالي للدراسات التكنولوجique بنابل

Département Technologies de l'Informatique

Code :16/25

PARCOURS

DEVELOPPEMENT DES SYSTEMES D'INFORMATION

PROJET DE FIN DE PARCOURS

INTITULE

CONCEPTION ET DEVELOPPEMENT D'UNE APPLICATION DE TOURISME MEDICAL « KLINIQA »

ORGANISME D'ACCUEIL

AFTERCODE

REALISE PAR

MALAK BEN OTHMANE

ISLEM ZIADI

ENCADRANT ACADEMIQUE

SALIMA MNIF

ENCADRANT PROFESSIONNEL

WAEL MAJDOUN

Fiche synthétique

Nom de l'entreprise	AFTERCODE
Ville	Nabeul
Nom de l'encadrant industriel	Wael Majdoub
Date début et date fin du stage	De 05 février 2025 jusqu'à 28 mai 2025
Nombre de jours effectifs passés dans les locaux de l'entreprise	Deux fois par semaine
Les principales tâches réalisées durant le stage (maximum 3)	Auto-formation, Conception et Développement
Le stage contient-il une période d'observation ? Si oui combien de jours ça a duré.	Non
Le stage contient-il une partie conception ? Si oui combien de jours ça a pris	Oui, 15 jours
Le stage contient-il une partie développement ? Si oui combien de jours ça a pris	Oui, 4 mois
Quels sont les prérequis pour ce stage en termes de compétences techniques ?	Avoir une connaissance de base de Flutter et Next.js
Les outils et logiciels utilisés durant le stage	Slack, Plane, Visual Studio Code, Android Studio, GitHub, Postman, Draw.io
Le matériel et la machinerie (autre que les PCs) utilisés durant le stage	Aucune
Méthodologies utilisées durant le stage	Approche Agile : SCRUM
Les deux principales qualifications compétences comportementales acquises à travers ce stage	<ul style="list-style-type: none"> - Travail en équipe - Gestion de temps - Gestion du stress
Les deux principales qualifications compétences techniques acquises à travers ce stage	<ul style="list-style-type: none"> - Utilisation d'un stack moderne : Flutter, Next.js, Hono, Bun, Drizzle ORM avec PostgreSQL (Neon). - Mise en place d'une architecture monolithique modulaire basée sur le Domain-Driven Design

Dedicaces

A dieu le tout puissant

Je remercie celui qui m'a accordé la force et le courage nécessaires à la réalisation de ce travail.

Je dédie ce travail a toutes les personnes qui ont été là, pendant ce long chemin, et vous y avez tous contribue d'une manière ou d'une autre.

À ma famille,

Merci. Pour tout. Pour l'amour, la patience, les encouragements. Je vous remercie pour tous vos sacrifices incommensurables pour mon bonheur et mon confort.

À Ma Chère Binôme Islam,

On en a traversé, des hauts, des bas, des nuits blanches, des rushs, des moments de doute et aussi des bons fous rires ! Tu as été plus qu'une partenaire. Merci pour ta patience, ton sérieux et ta capacite à toujours garder le cap. On a fait une belle équipe.

À Mes Amis,

Que ce soit de l'université ou hors de l'université. Vous ne m'avez jamais laisse tomber. Même à distance, même occupés, vous étiez là. Merci pour tous les partages, les fous rires, les idées jetées à la va-vite qui se sont parfois transformées en bonnes solutions.

À mes professeurs universitaires,

Je remercie tous mes professeurs pour leur soutien durant mes études. Un merci tout particulier à Mme Salima Mnif, pour sa patience, ses conseils et son aide précieuse qui nous ont vraiment aidé à avancer

À l'équipe de l'entreprise d'accueil,

Un grand merci à toute, votre soutien et votre disponibilité. Votre expertise et vos conseils ont été précieux, et votre engagement a vraiment fait la différence dans la réussite de ce projet

Malak Ben Othmane

Dédicaces

Alhamdulillah, je loue et remercie Allah du fond du cœur pour m'avoir guidé, soutenu tout au long de ce parcours.

Je dédie ce succès à toutes les personnes dont la bienveillance et le soutien ont façonné mon parcours universitaire.

À ma famille,

Votre amour inconditionnel, vos nombreux sacrifices et votre présence quotidienne ont toujours été ma force. Dans les moments difficiles comme dans les instants de joie, vous m'avez aidé à croire en moi et à persévérer.

À Ma Chère binôme Malak,

Merci pour cette collaboration sincère et enrichissante. Ton engagement, ta rigueur et notre complémentarité ont été essentiels pour mener à bien ce projet. Ensemble, nous avons surmonté les défis et partagé nos idées dans un véritable esprit d'équipe.

À mes professeurs universitaires,

Je tiens à exprimer ma profonde gratitude, en particulier envers Mme Salima Mnif. Son expertise, son dévouement et son soutien ont été d'une valeur inestimable tout au long de mon parcours universitaire.

À l'équipe de l'entreprise d'accueil,

Merci pour votre accueil, votre soutien, votre disponibilité et le partage de votre expertise.

Merci à toutes et à tous pour votre soutien, vos conseils et votre engagement, qui ont rendu cette réussite possible.

Islem Ziadi

Remerciements

Nous exprimons nos sincères remerciements à toutes les personnes qui ont facilité notre stage.

À l'équipe de l'organisme d'accueil, nous souhaitons remercier toute l'équipe pour son accueil chaleureux. Votre soutien et votre confiance ont contribué à créer un environnement favorable à notre apprentissage.

À notre encadrant externe nous sommes reconnaissants à monsieur Wael Majdoub pour son aide précieuse et sa totale confiance. Nous remercions également monsieur Laith mahdi pour ses encouragements, ses conseils techniques et son implication, qui ont joué un rôle essentiel dans l'avancement de notre travail.

À notre encadrante au sein de l'ISET Nabeul nous tenons à exprimer notre profonde gratitude à madame Salima Mnif pour son accompagnement attentif, son soutien inestimable et ses conseils avisés tout au long de notre stage. Sa disponibilité et son aide lors des moments les plus difficiles méritent notre respect et notre reconnaissance.

Aux enseignants de l'ISET Nabeul nous remercions également tous les enseignants qui ont contribué à notre formation. Leur soutien, leur partage de connaissances et leur engagement ont été déterminants dans notre parcours académique.

Aux membres du jury enfin, nous exprimons notre sincère reconnaissance aux membres du jury qui ont consacré leur temps et leur expertise à l'évaluation de notre travail.

Merci à toutes et à tous pour votre soutien et votre engagement.

Sommaire

INTRODUCTION GENERALE.....	1
1. CHAPITRE 1 : CADRE GENERAL DU PROJET.....	2
1.1 Présentation de l'organisme d'accueil	3
1.1.1 Identification de la société	3
1.1.2 Fiche d'identité de société	3
1.1.3 Services	3
1.2 Cadre du projet	4
1.2.1 Problématique.....	4
1.2.2 Analyse de l'existant.....	5
1.2.3 Solution proposée	6
1.3 Méthodologie de travail.....	7
1.3.1 Les méthodologies agiles.....	7
1.3.2 La méthodologie SCRUM.....	8
2. CHAPITRE 2 :PREPARATION DU PROJET	10
2.1 Capture des besoins	11
2.1.1 Définition des acteurs	11
2.1.2 Spécification des besoins	11
2.1.2.1 Besoins fonctionnels	11
2.1.2.2 Besoins non fonctionnels.....	14
2.1.3 Modélisation des besoins	15
2.2 Pilotage du projet avec SCRUM	16
2.2.1 Equipe et rôles.....	16
2.2.2 Le Backlog du produit	16
2.2.3 Planification des releases / sprints	24
2.3 Environnement de travail.....	25
2.3.1 Environnement matériel	25
2.3.2 Environnement technologies	26
2.3.2.1 Outils logiciels	26
2.3.2.2 Technologies et frameworks	28
2.3.2.3 Protocoles.....	29

2.3.2.4	Bibliothèques	30
2.4	Architecture proposée.....	31
2.4.1	Présentation de l'architecture « Modular Monolith with Domain-	31
Driven Design »	31	
2.4.2	Avantage de cette architecture.....	32
2.4.3	Comparaison et évaluation des architectures	32
3.	CHAPITRE 3 : RELEASE 1 « MODULE MEDICAL »	34
3.1	Développement du Sprint 1 « Fondations du module médical »	35
3.1.1	Backlog du Sprint 1	35
3.1.2	Analyse.....	40
3.1.2.1	Diagramme de cas d'utilisation globale de sprint 1.....	40
3.1.2.2	Diagrammes de cas d'utilisation détaillés	41
3.1.3.1	Description textuelle des cas d'utilisation	44
3.1.3	Conception	47
3.1.3.2	Diagramme de classe de sprint 1	47
3.1.3.3	Diagrammes de séquence de sprint 1.....	48
3.1.4	Réalisation.....	51
3.1.4.1	Développement de Coté Web.....	51
3.1.4.2	Développement de Coté Mobile	53
3.1.4.1	SCRUM Board.....	54
3.2	Développement du Sprint 2 « Services patients et rendez- vous ».....	55
3.2.1	Backlog du Sprint 2	55
3.2.2	Analyse.....	59
3.2.2.1	Diagramme de cas d'utilisation globale de sprint 2.....	59
3.2.2.2	Diagramme de cas d'utilisation détaillés.....	60
3.2.2.3	Description textuelle des cas d'utilisation	63
3.2.3	Conception	66
3.2.3.1	Diagramme de classe de sprint 2	66
3.2.3.2	Diagrammes de séquences de sprint 2	67
3.2.4	Réalisation.....	70
4.	CHAPITRE 4 : RELEASE 2 « MODULE DE COMMUNICATION ETAI ».....	74
4.1	Développement du Sprint 3 « Services de communication».....	75
4.1.1	Backlog du Sprint 3	75
4.1.1	Analyse	77

4.1.2.1	Diagramme de cas d'utilisation globale de sprint 3.....	77
4.1.2.2	Description textuelle des cas d'utilisation	78
4.1.2	Conception	80
4.1.3.1	Diagramme de classe de sprint 3	80
4.1.3.2	Digrammes de séquences de sprint 3.....	81
4.1.3.3	Diagramme d'activité de sprint 3	83
4.1.3	Réalisation	84
4.1.4.1	Espace logement	84
5.	CHAPITRE 5 : RELEASE 3 «MODULETOURISME»	87
5.1	Développement du Sprint 4 « Services de bien-être et d'hébergement »	88
5.1.1	Backlog du sprint 4.....	88
5.1.2	Analyse	92
5.1.2.1	Diagramme de cas d'utilisation global du sprint 4	92
5.1.2.2	Diagrammes de cas d'utilisation détaillés	93
5.1.2.3	Description textuelle des cas d'utilisation	96
5.1.3	Conception	103
5.1.3.1	Diagrammes des séquences	103
5.1.3.2	Diagramme de classe de sprint 4	106
5.1.4	Réalisation	107
5.1.4.1	Espace logement.....	107
5.1.4.2	Espace endroits de bien-être	110
	CONCLUSION GENERALE	113
	WEBOGRAPHIE.....	114

Liste des figures

FIGURE 1.1 : LOGO DE L'ENTREPRISE D'ACCUEIL.....	3
FIGURE 1.2 : CHART DE LA SOLUTION PROPOSEE	7
FIGURE 1.3 : ARCHITECTURE DE METHODOLOGIE AGILE : SCRUM	9
FIGURE 2.1 : DIAGRAMME DE CAS D'UTILISATION GLOBALE.....	15
FIGURE 2.2 : DIAGRAMME DE GANT PLANIFICATION DES SPRINTS	25
FIGURE 2.3 : LOGO DRAW.IO	26
FIGURE 2.4 : LOGO FIGMA	26
FIGURE 2.5 : LOGO VS CODE	26
FIGURE 2.6 : LOGO ANDROID STUDIO	26
FIGURE 2.7 : LOGO POSTMAN	27
FIGURE 2.8 : LOGO GITHUB	27
FIGURE 2.9 : LOGO SLACK	27
FIGURE 2.10 : LOGO PLANE	27
FIGURE 2.11 : LOGO FLUTTER	28
FIGURE 2.12 : LOGO NEXT.JS	28
FIGURE 2.13 : LOGO HONO	28
FIGURE 2.14 : LOGO BUN	28
FIGURE 2.15 : LOGO DRIZZLE-ORM	29
FIGURE 2.16 : LOGO NEON POSTGRESQL	29
FIGURE 2.17 : LOGO SUPABASE	29
FIGURE 2.18 : HTTP PROTOCOLE.....	29
FIGURE 2.19 : LOGO ZODVALIDATOR	30
FIGURE 2.20 : LOGO SHADCN/UI	30
FIGURE 2.21 : LOGO TANSTACK QUERY	30
FIGURE 2.22 : DIO PACKAGE.....	30
FIGURE 2.23 : LIVEKIT CLIENT	30
FIGURE 2.24 : ARCHITECTURE MODULAR MONOLITH WITH DD	33
FIGURE 3.1 : DIAGRAMME DE CAS D'UTILISATION DE SPRINT 1	40
FIGURE 3.2 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE FILTRAGE DOCTEUR.....	41
FIGURE 3.3 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE FILTRAGE ETABLISSEMENT....	41
FIGURE 3.4 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION ETABLISSEMENT.....	42

FIGURE 3.5 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION DOCTEUR	42
FIGURE 3.6 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION D'INFORMATION PROFESSIONNELLE.....	43
FIGURE 3.7: DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION DE SPECIALITES.....	43
FIGURE 3.8 : DIAGRAMME DE CLASSE SPRINT 1	47
FIGURE 3.9 : DIAGRAMME DE SEQUENCE « AJOUT UN ETABLISSEMENT »	48
FIGURE 3.10 : DIAGRAMME DE SEQUENCE DE MODIFIER INFORMATION PROFESSIONNELLE.....	49
FIGURE 3.11 : DIAGRAMME DE SEQUENCE D'INSCRIPTION DOCTEUR	50
FIGURE 3.12 : INTERFACE WEB DE LISTE DES DOCTEURS.....	51
FIGURE 3.13 : INTERFACE WEB DE LISTE DES ETABLISSEMENTS	52
FIGURE 3.14 : INTERFACE D'AJOUT D'UNE AFFILIATION	52
FIGURE 3.15 : INTERFACE DE DETAILS D'UN DOCTEUR.....	53
FIGURE 3.16 : INTERFACE DETAILS D'UNE AFFILIATION	53
FIGURE 3.17 : SCRUM BOARD DE GESTION SPECIALITE SPRINT 1	54
FIGURE 3.18 : DIAGRAMME DE CAS D'UTILISATION GLOBALE DE SPRINT 2.....	59
FIGURE 3.19 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION EMPLOI DU TEMPS .	60
FIGURE 3.20 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION DISPONIBILITE.....	60
FIGURE 3.21 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION DES ORDONNANCES	61
FIGURE 3.22 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION PATIENTS	61
FIGURE 3.23 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION FICHIER EXTERNES	62
FIGURE 3.24 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION DES RENDEZ VOUS.	62
FIGURE 3.25 : DIAGRAMME DE CLASSE DE SPRINT 2	66
FIGURE 3.26 : DIAGRAMME DE SEQUENCE D'AJOUTER CRENEAU HORAIRE	67
FIGURE 3.27 : DIAGRAMME DE SEQUENCE DE PARTAGER D'UN DOCUMENT	68
FIGURE 3.28 : DIAGRAMME DE SEQUENCE DE PASSER RENDEZ VOUS.....	69
FIGURE 3.29 : INTERFACE MOBILE D'EMPLOI DU TEMPS DE DOCTEUR	70
FIGURE 3.30 : INTERFACE DE CALENDRIER DE DOCTEUR	71
FIGURE 3.31 : INTERFACE DE LA LISTE DES RENDEZ-VOUS	72
FIGURE 3.32 : INTERFACE DETAILS D'UN PATIENT	73
FIGURE 4.1 : DIAGRAMME DE CAS D'UTILISATION DE SPRINT 3	77
FIGURE 4.2 : DIAGRAMME DE CLASSE DE SPRINT 3	80
FIGURE 4.3 : DIAGRAMME DE SEQUENCE DE ENVOYER MESSAGE	81

FIGURE 4.4 : DIAGRAMME DE SEQUENCE DE REJOINDRE VIDEO CONFERENCE	82
FIGURE 4.5 : DIAGRAMME D'ACTIVITE D'ASSISTANT AI.....	83
FIGURE 4.6 : INTERFACE DE MESSAGERIE ENTRE DOCTEUR ET PATIENT	84
FIGURE 4.7 INTERFACE DE DISCUSSION AVEC ASSISTANT AI	85
FIGURE 4.8 : INTERFACE DE VISIO CONFERENCE	86
FIGURE 5.1 DIAGRAMME DE CAS D'UTILISATION GLOBALE DE SPRINT 4	92
FIGURE 5.2 DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE CONSULTER DETAILS LOGEMENT	93
FIGURE 5.3 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION FAVORIS LOGEMENT	93
FIGURE 5.4 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE FILTRAGE LOGEMENT	94
FIGURE 5.5 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE RESERVATION D'UN LOGEMENT	94
FIGURE 5.6 : DIAGRAMME DE CAS D'UTILISATION DE CONSULTER LES ENDROITS DE BIEN ETRE	95
FIGURE 5.7 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE FILTRAGE DES ENDROITS DE BIEN ETRE.....	95
FIGURE 5.8 : DIAGRAMME DE CAS D'UTILISATION DETAILLEE DE GESTION DES LIEUX DE BIEN ETRE FAVORIS	96
FIGURE 5.9 : DIAGRAMME DE SEQUENCE DE CONSULTER HEBERGEMENT	103
FIGURE 5.10 : DIAGRAMME DE SEQUENCE DE RESERVER UN HEBERGEMENT	104
FIGURE 5.11 : DIAGRAMME DE SEQUENCE DE FILTRAGE D'ENDROITS DE BIEN ETRE	105
FIGURE 5.12 : DIAGRAMME DE CLASSE SPRINT 5	106
FIGURE 5.13 : INTERFACE DE RESULTAT FILTRAGE LOGEMENT.....	107
FIGURE 5.14 : INTERFACE DETAILS LOGEMENT.....	108
FIGURE 5.15 : INTERFACE DISPONIBILITES CHAMBRES.....	109
FIGURE 5.16 : INTERFACE DETAILS CHAMBRE	109
FIGURE 5.17 : INTERFACE D'ACCUEIL DE SERVICE DE BIEN ETRE	110
FIGURE 5.18 : INTERFACE DE FILTRAGE D'ENDROIT DE BIEN ETRE	111
FIGURE 5.19 : INTERFACE DETAILS D'UN ENDROIT DE BIEN ETRE	111
FIGURE 5.20 : INTERFACE DETAILS D'UNE OFFRE	112

Liste des tableaux

TABLEAU 1.1 : FICHE D'IDENTITE DE SOCIETE	3
TABLEAU 1.2 : ANALYSE D'EXISTANT.....	5
TABLEAU 2.1 : EQUIPE SCRUM.....	16
TABLEAU 2.2 : ENVIRONNEMENT MATERIEL	25
TABLEAU 2.3 : COMPARAISON ENTRE ARCHITECTURES	31
TABLEAU 2.4 : AVANTAGES TECHNIQUES DE L'ARCHITECTURE UTILISEE	32
TABLEAU 2.5 : AVANTAGES ARCHITECTURAUX DE L'ARCHITECTURE UTILISEE	32
TABLEAU 2.6 : EVALUATION DES ARCHITECTURES.....	33
TABLEAU 3.1 : BACKLOG DU SPRINT 1	35
TABLEAU 3.2 : DESCRIPTION TEXTUELLE DU CAS D'UTILISATION « AJOUTER UNE AFFILIATION »	44
TABLEAU 3.3 : DESCRIPTION TEXTUELLE DE CAS D'UTILISATION « FILTRER UN DOCTEUR »	45
TABLEAU 3.4 : BACKLOG DU SPRINT 2	55
TABLEAU 3.5 : DESCRIPTION TEXTUELLE DE CAS D'UTILISATION « PASSER UN RENDEZ-VOUS »	63
TABLEAU 3.6 : DESCRIPTION TEXTUELLE DE CAS D'UTILISATION « CREER UNE ORDONNANCE ».	64
TABLEAU 4.1 : BACKLOG DU SPRINT 3	75
TABLEAU 4.2 : DESCRIPTION TEXTUELLE DE CAS D'UTILISATION « ENVOYER UN MESSAGE » ...	78
TABLEAU 4.3 : DESCRIPTION TEXTUELLE DE CAS D'UTILISATION « CREER UNE VIDEOCONFERENCE »	79
TABLEAU 5.1 : BACKLOG DU SPRINT 4	88
TABLEAU 5.2 : DESCRIPTION TEXTUELLE DE CAS D'UTILISATION "FILTRER UN LOGEMENT"	96
TABLEAU 5.3 : DESCRIPTION TEXTUELLE DE CAS D'UTILISATION « RESERVER UN LOGEMENT »	98
TABLEAU 5.4 : DESCRIPTION TEXTUELLE DE CAS D'UTILISATION « CONSULTER LES LOGEMENTS »	100
TABLEAU 5.5 : DESCRIPTION TEXTUELLE DE CAS D'UTILISATION « FILTRER LES ENDROITS » .	101

Introduction Générale

Afin de faciliter l'accès aux soins médicaux et de répondre aux problématiques liées aux délais d'attente prolongés ainsi qu'aux coûts élevés des consultations, de nombreuses initiatives ont vu le jour dans le secteur de la santé. Toutefois, ces solutions restent souvent limitées à certaines prestations spécifiques ou à des zones géographiques restreintes, ce qui ne permet pas toujours de satisfaire les besoins des patients en quête de services médicaux rapides et abordables.

Le développement des technologies numériques et des plateformes de télémédecine a ouvert de nouvelles perspectives, permettant aux patients d'accéder à des soins spécialisés à distance tout en optimisant le processus de prise de rendez-vous et de consultation. Dans cette optique, le tourisme médical international se présente comme une solution innovante, offrant aux patients la possibilité de consulter des médecins de différents pays, tout en bénéficiant d'une prise en charge complète, incluant l'hébergement, les services de bien-être et de soins, ainsi que l'accès aux laboratoires et aux pharmacies.

Dans le cadre de ce projet, nous avons eu l'opportunité de concevoir une application mobile dédiée au tourisme médical international, avec un premier déploiement centré en Tunisie. Ce pays, réputé pour l'expertise de ses médecins et la qualité de ses infrastructures médicales, constitue un point de départ idéal pour mettre en valeur les compétences des professionnels de santé tunisiens et offrir une alternative accessible aux patients confrontés à des difficultés d'accès aux soins dans leur pays d'origine.

Ce rapport résume l'essentiel de notre travail. Il est structuré de la manière suivante :

Le premier chapitre présente le contexte général du projet, en mettant en lumière les enjeux du tourisme médical et les avantages de cette solution numérique. Dans le deuxième chapitre, nous détaillons les fonctionnalités de l'application, sa conception et les technologies utilisées. Les derniers chapitres abordent les différentes étapes de développement du projet, les défis rencontrés et les perspectives d'amélioration future.

1. Chapitre 1 :

Cadre général du projet

1.1 Présentation de l'organisme d'accueil

1.1.1 Identification de la société

AFTERCODE, est une entreprise tunisienne de développement de logiciels, offre depuis 2019 des services de premier ordre grâce à des talents d'exception et des outils de pointe. Elle est spécialisée dans le développement sur mesure des applications web et mobile, la création des sites web, des solutions personnalisées en AI, ainsi que des services de design. De plus, elle propose des services de maintenance fluides pour les projets existants.



Figure 1.1 : Logo de l'entreprise d'accueil

1.1.2 Fiche d'identité de société

Tableau 1.1 : Fiche d'identité de société

Nom de l'entreprise	AFTERCODE
Secteur d'activité	Développement de logiciels
Adresse	نهج الجامعة العربية Nabeul, Dar Chaabane Al Fehri 8011
Adresse électronique	contact@aftercode.tn
Téléphone	+216 24 201 201
Site Web	https://aftercode.tn/fr/

1.1.3 Services

Les services offerts aux clients de AFTERCODE sont les suivants :

- **Développement Mobile** : Créer des applications mobiles compatible avec Android et IOS, optimisées pour la performance et l'interactivité.
- **Développement Web** : Développer des logiciels sur mesure qui résoudront les problèmes quotidiens du client.
- **Design** : Conçoit, développer et optimiser des interfaces numériques et des visuels impactantes afin d'offrir une expérience utilisateur fluide.

- **Community management** : Propose d'accompagner le client dans sa stratégie de communication digitale pour une valeur ajoutée.
- **Référencement web** : Permet de fortement améliorer le positionnement d'un site sur les moteurs de recherche.
- **Hébergement web** : Offre des services d'hébergement du site web, et la création du nom de domaine.
- **Formation** : Offre aux entreprises et particuliers des formations en développement web et mobile.

1.2 Cadre du projet

Entre le 10 février et le 28 mai, nous avons eu l'opportunité de réaliser un stage pratique chez AFTERCODE. Durant cette période, nous avons été confrontés à divers défis techniques, ce qui a contribué à approfondir notre compréhension de l'informatique. Plus qu'une simple acquisition de connaissances, ce stage nous a offert un aperçu précieux du fonctionnement interne d'une entreprise et des stratégies pour répondre efficacement aux attentes des clients.

1.2.1 Problématique

Dans de nombreux pays, les patients sont confrontés à des délais d'attente prolongés pour obtenir un rendez-vous médical, qui peuvent aller de plusieurs semaines à plusieurs mois. Cette situation met en péril leur santé et leur qualité de vie, en particulier lorsqu'il s'agit de traitements ou diagnostics nécessitant un second avis médical ou une consultation spécialisée. Par ailleurs, les solutions locales sont souvent coûteuses, ou les patients n'ont pas accès à des spécialistes qualifiés. D'un autre côté, les opportunités de consultation online ou dans d'autres pays sont souvent méconnues, alors qu'elles peuvent être plus rapides et parfois moins chères, même en tenant compte des coûts de déplacement et de séjour.

De plus, organiser un déplacement médical à l'étranger est complexe :

- Trouver un médecin ou un établissement médical de confiance
- Trouver un logement adapté à ses besoins

Cette complexité décourage de nombreux patients qui pourraient autrement bénéficier d'un traitement rapide et efficace

1.2.2 Analyse de l'existant

Tableau 1.2 : Analyse d'existant

Plateforme	Points forts	Points faibles
Med.tn	<ul style="list-style-type: none"> • Couverture nationale en Tunisie • Annuaire complet (médecins, labo, pharmacie) • Appli mobile native 	<ul style="list-style-type: none"> • Pas de téléconsultation vidéo • Pas de messagerie sécurisée interne • Pas de services bien-être/tourisme
Vezeeta	<ul style="list-style-type: none"> • Téléconsultation audio/vidéo • Prise de RDV (cabinet, domicile) • Pharmacie en ligne avec livraison • UX moderne et transparente sur les prix 	<ul style="list-style-type: none"> • Couverture géographique limitée à MENA/Afrique • Pas de volet tourisme médical • Pas de messagerie interne dédiée
Bookimed	<ul style="list-style-type: none"> • Tourisme médical international (50+ pays) • Hébergement et logistique pris en charge • Offres bien-être à l'étranger 	<ul style="list-style-type: none"> • Pas de vraie téléconsultation vidéo • Pas de messagerie sécurisée patient-médecin • Pas d'appli mobile
Vaidam	<ul style="list-style-type: none"> • Couverture très large (125+ pays) • Coordination complète séjour (hôtel, transferts) • Avis initial gratuit et devis personnalisé 	<ul style="list-style-type: none"> • Pas de téléconsultation vidéo intégrée • Pas de messagerie interne • Pas de services pharmacie/labos indépendants

- Actuellement, aucune application existante n'offre et ne s'occupe de toutes les fonctionnalités nécessaires pour une solution de tourisme médical efficace. C'est pour cela qu'il est nécessaire de concevoir une plateforme tout-en-un, capable de réunir l'ensemble des services

1.2.3 Solution proposée

Après une recherche élargie en termes de technologie, fonctionnalités et compatibilité aux besoins, notre solution « Kliniqa », est une application tout-en-un conçue pour offrir une prise en charge complète du tourisme médical. Elle intègre l'ensemble des fonctionnalités nécessaires, de la prise de contact avec des spécialistes hautement qualifiés à la gestion de logement et du bien-être du patient tout au long de son expérience d'utilisation de l'application. Notre solution offre à l'utilisateur la possibilité de bénéficier de :

- **Une expertise médicale** : L'application met en avant la compétence et l'excellence des médecins tunisiens/ syrien / soudaniens. Chaque praticien dispose d'un profil détaillé (qualifications, expériences, avis) renforçant ainsi la confiance des futurs patients.
- **Des services de pharmacie et de laboratoire** : L'application permet de passer une commande d'une manière légale de médicaments en pharmacie et réservation de tests dans les laboratoires à proximité.
- **Une gestion complète du parcours patient** : L'application assure un accompagnement personnalisé et sécurisé au patient en intégrant la planification de rendez-vous médicaux, les consultations à distance (télémedecine) et le suivi post-opératoire.
- **Une prise en charge logistique** : Outre les services médicaux, on a la possibilité de réserver un hébergement. Ce service intégré permet d'alléger le stress lié aux déplacements et de se concentrer sur le traitement et la récupération.
- **Un accompagnement bien-être** : L'application propose des programmes dédiés au bien-être et à la réhabilitation, incluant des options de spa, de retraites bien-être et de remise en forme, afin d'optimiser le séjour et favoriser une convalescence harmonieuse.
- **Une assistance en continu** : L'application offre un assistant AI, garantissant une communication fluide et un service de qualité.

➤ Après avoir détaillé la solution proposée, le schéma ci-dessous permet de visualiser l'ensemble des grands composants clés de l'application ainsi que leur répartition selon les responsabilités de développement :

- Les parties à développer côté Web
- Les parties à développer côté mobile
- Les parties à développer par une autre équipe

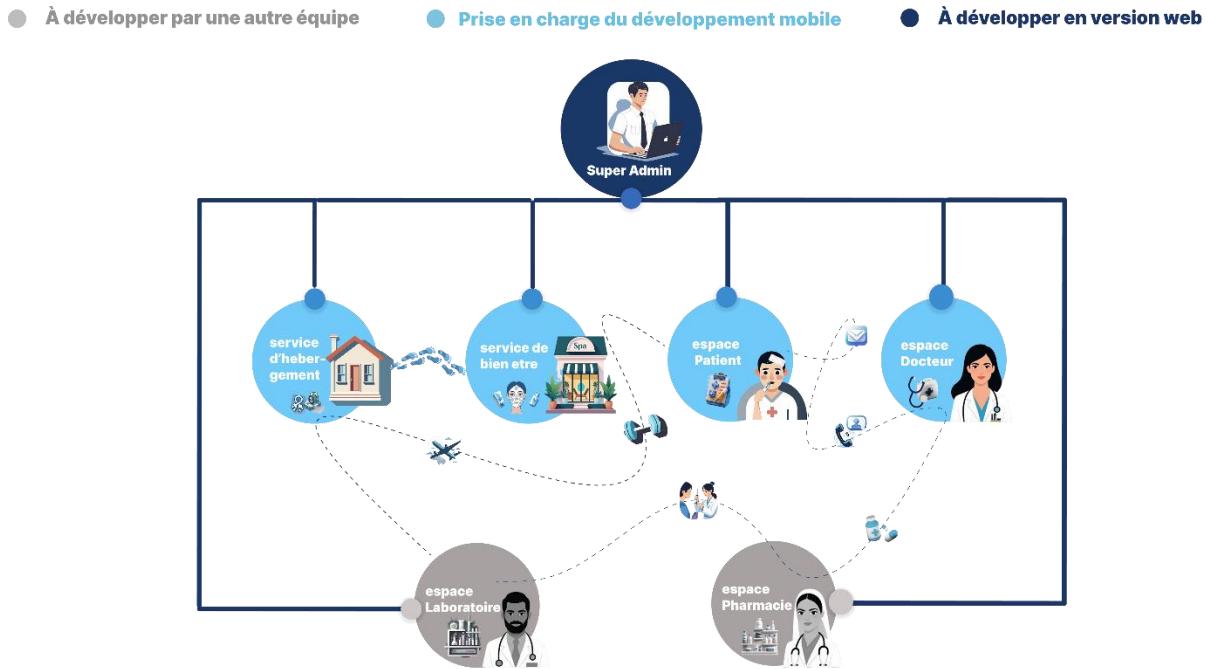


Figure 1.2 : Chart de la solution proposée

1.3 Méthodologie de travail

La réalisation du projet dans les délais de livraison est le souci majeur de chaque équipe de développement d'un logiciel. L'un des problèmes les plus fréquemment affrontés lors de la construction du logiciel est la mauvaise spécification et le changement brusque des besoins. Cela peut influencer non seulement l'équipe de développement en créant un environnement de stress mais aussi le temps consacré pour la réalisation du projet et donc des délais de livraison dépassés. Afin d'éviter ces situations critiques, nous adoptons la méthodologie SCRUM pour la gestion de notre projet.

1.3.1 Les méthodologies agiles

Une méthode agile est une approche itérative et incrémentale qui est menée dans un esprit collaboratif avec juste ce qu'il faut de formalisme, elle génère un produit de haute qualité tout en prenant en compte l'évolution des besoins des clients. Cette méthode est une réponse aux méthodes de gestion des projets dites prédictives comme le cycle en V ou en cascade.

Les 4 principes essentiels du Manifeste Agile :

- **L'équipe** : Personnes et interactions *plutôt que* processus et outils.
- **L'application** : Logiciel fonctionnel *plutôt que* documentation complète.
- **La collaboration** : Collaboration avec le client plutôt que négociation de contrat.
- **L'acceptation du changement** : Réagir au changement plutôt que suivre un plan.

Nous avons proposé d'explorer l'une des méthodes Agile existant qu'on a choisi de s'adapter.

1.3.2 La méthodologie SCRUM

Scrum est une méthodologie ou cadre de gestion de projet. Elle est construite autour d'une équipe Scrum, d'artéfacts et de cérémonies. C'est un cadre méthodologie incrémental, organisé en sprint pendant lesquels l'équipe Scrum va produire un incrément potentiellement livrable du produit. Le but est de livrer un incrément qui a une valeur ajoutée pour le produit final.

- **Les Piliers de Scrum :**

Scrum est fondé sur la théorie du contrôle empirique de processus et soutenu par 3 piliers fondamentaux :

- **La transparence** : le fait d'être honnête, de ne rien avoir à cacher, de travailler ensemble au succès du produit/projet en rendant les aspects importants du processus visibles à tous ceux qui sont responsables des résultats.
- **L'inspection** : le fait de pouvoir s'entraider et inspecter les artefacts Scrum et l'état d'avancement par rapport à un Objectif de Sprint afin de détecter les écarts indésirables.
- **L'adaptation** : le fait de s'adapter aux changements en général, changements de produit.

- **Equipe Scrum :**

Les équipes Scrum sont auto-organisées et pluridisciplinaires. Elles choisissent la meilleure façon d'accomplir leur travail, au lieu d'être dirigées par des personnes externes à l'équipe, ce qui favorise la flexibilité, la créativité et la productivité.

- Le scrum master : responsable de faire respecter le cadre scrum
- Le product owner : responsable du produit, le quoi
- L'équipe de développement : responsable de la production, le comment

- **Les 5 événements Scrum**

- Revue du backlog : permet d'avoir la vision produit
- Planification du Sprint : permet de déterminer l'objectif de sprint
- Daily meeting : permet de synchroniser l'équipe de développement
- Revue du sprint : permet de collecter les retours/feedbacks du client
- Rétrospective de Sprint : permet d'améliorer les pratiques

- **Les 3 Artefacts Scrum**

- Le backlog produit : liste ordonnée de tout ce qui pourrait être requis dans le produit.
- Le backlog de Sprint : prévision faite des fonctionnalités qui seront présentes dans le prochain incrément
- L'incrément visible : éléments du Product Backlog terminés pendant le sprint ainsi que de la valeur cumulative des incréments livrés dans les sprints précédents.

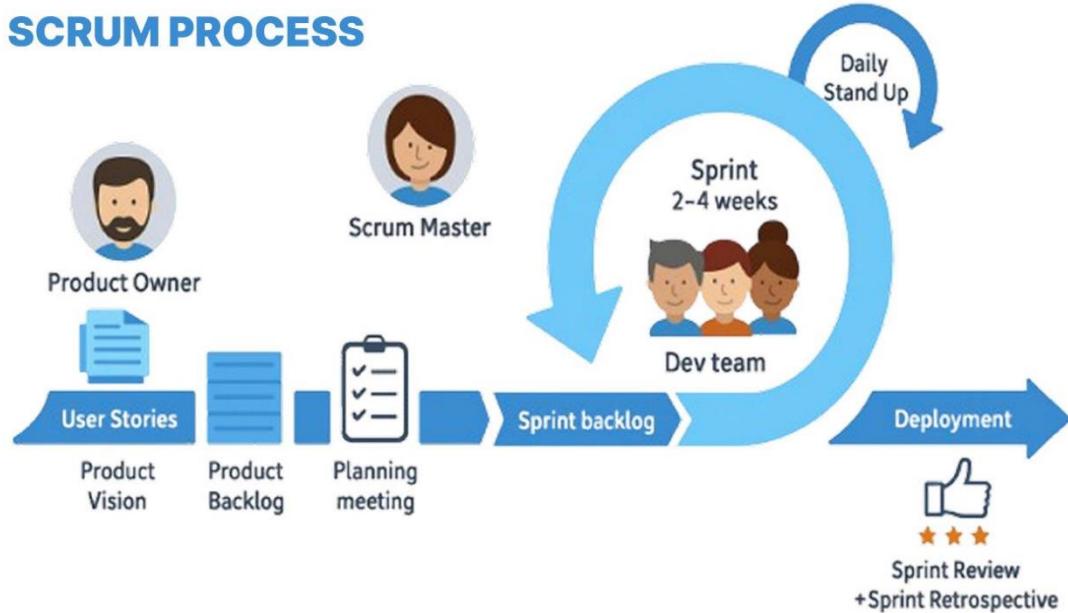


Figure 1.3 : Architecture de méthodologie Agile : SCRUM

2. Chapitre 2 : Préparation du projet

2.1 Capture des besoins

2.1.1 Définition des acteurs

Un acteur est une entité externe qui agit sur le système. Le terme acteur ne désigne pas seulement les utilisateurs humains mais également les autres systèmes.

En effet, notre système sera utilisé par :

- **Super Administrateur** : détient l'accès complet à l'ensemble des fonctionnalités de la plateforme. Il est responsable de la gestion des comptes utilisateurs. Sa mission consiste à superviser et garantir la cohérence, la sécurité et la performance globale du système.
- **Docteur** : gère ses informations personnelles et organise l'ensemble de ses activités professionnelles. Il peut planifier et suivre ses rendez-vous, gérer sa disponibilité et assurer le suivi personnalisé de ses patients
- **Patient** : représente l'utilisateur central de la plateforme. Il bénéficie d'un accès aux principaux modules de l'application (médical, hébergement, service de bien-être) et peut administrer les dossiers médicaux de ses proches. Il est en mesure de planifier des rendez-vous pour lui-même ou pour les membres de sa famille

2.1.2 Spécification des besoins

Les besoins fonctionnels décrivent les différentes fonctionnalités que l'application doit offrir à ses utilisateurs pour répondre à leurs attentes spécifiques

- Les fonctionnalités **administratives** sont disponibles via une **interface web** dédiée au super administrateur
- Les fonctionnalités destinées aux **utilisateurs finaux** (patients et médecins) sont accessibles depuis **l'application mobile**

2.1.2.1 Besoins fonctionnels

- **Inscription**

Les utilisateurs peuvent créer un compte sur l'application pour accéder aux différents services.

Pour les médecins, l'inscription inclut la saisie d'informations de vérification (identité, vérification de diplôme) ainsi que la description de leurs compétences

- **Authentification**

Les utilisateurs se connectent à leur compte de manière sécurisée afin d'accéder à leur espace dédié (patient, médecin, administrateur)

- **Gestion du profil utilisateur**

Chaque utilisateur peut, après connexion, modifier ses informations personnelles nom, adresse, mot de passe, etc.

- **Gestion des informations professionnelles (Médecins)**

Les docteurs ont accès à un formulaire dédié pour mettre à jour leurs données professionnelles : langues parlées, lieux d'exercice (hôpitaux, cliniques), qualifications (diplômes, certifications), sous-spécialités, etc.

- **Gestion des dossiers patients (Patients & Famille)**

Un patient peut créer et gérer des fiches médicales pour lui-même ou pour un proche :

- Ajout et mise à jour des informations médicales (maladies chroniques, allergies, groupe sanguin, etc.)
- Modification des fiches patients existantes

- **Consultation de la liste des docteurs**

Un utilisateur peut consulter la liste des médecins et leurs profils détaillés afin de sélectionner celui qui correspond le mieux à ses besoins.

- **Consultation de la liste des établissements de santé**

Un utilisateur peut consulter la liste des établissements et leurs détails afin de sélectionner celui qui correspond le mieux à ses besoins.

- **Filtrage des établissements de santé**

Un utilisateur peut filtrer les affiliations par pays, les spécialités qu'elle offre.

- **Filtrage des docteurs**

Un utilisateur peut filtrer les docteurs par pays, les spécialités et les sous spécialités qu'il pratique, les langages parlés, l'intervalle des notes des avis et de frais de consultation.

- **Prise de rendez-vous**

Le patient peut réserver un créneau en ligne ou en présentiel, en fonction de la disponibilité du médecin.

- **Gestion des articles partagées**

Les articles informatifs sont publiés et gérés par les médecins, soumis à la modération par l'administrateur et accessibles aux patients dans « Waiting Room »

- **Téléconsultation**

Consultation en ligne via une vidéoconférence pour un deuxième avis ou un suivi personnalisé en temps réel, assurant une prise en charge rapide et efficace des patients

- **Gestion de l'emploi du temps d'un médecin**

Les médecins peuvent définir et actualiser leurs créneaux de consultation (en ligne ou en présentiel) afin d'optimiser la planification des rendez-vous.

- **Gestion de disponibilités docteurs**

Les médecins peuvent bloquer des créneaux horaires et des dates pour indiquer leur indisponibilité (par exemple : opération, réunions à l'hôpital, congés, etc..).

- **Messagerie sécurisée**

Les médecins peuvent échanger avec leurs patients via un chat intégré, permettant à ces derniers d'envoyer des messages textuels.

- **Consultation des hébergements**

Les patients peuvent parcourir la liste des logements disponibles et consulter, pour chaque option, les informations détaillées (localisation, équipements, tarifs) afin de faire un choix éclairé.

- **Filtrage de logements**

Les utilisateurs peuvent filtrer les logements par localisation, coût par mois/jour, services disponibles, commodités.

- **Réservation d'hébergement**

Les patients prévoyant des soins à l'étranger peuvent réserver en ligne un logement entier ou une chambre individuelle, selon leurs besoins et préférences.

- **Gestion des logements favoris**

Les patients peuvent ajouter ou supprimer des hébergements de leur liste de favoris pour un accès rapide et personnalisé

- **Consultation des lieux de santé et bien-être**

Les patients peuvent consulter la liste des endroits de bien être, visualiser pour chacun les services proposés ainsi que leurs tarifs détaillés.

- **Filtrage des lieux de santé et bien-être**

Les utilisateurs peuvent filtrer les lieux par type de soins, coût par séance, services disponibles, commodités

- **Gestion des endroits de bien être favoris**

Les patients peuvent ajouter ou supprimer des lieux de bien être de leur liste de favoris pour un accès rapide et personnalisé

- **Assistant IA « Nassouh »**

Un assistant IA Aide les patients à identifier ses symptômes pour orienter vers la bonne spécialité, avoir des conseils de prévention personnalisés

- ❖ **Administration (Web)**

- **Gestion d'état de compte docteur**

Seul administrateur peut accepter demande d'inscription d'un docteur, le rejeter, l'activer, le bloquer en cas des soucis.

- **Gestion des spécialités et sous spécialités**

Seul le super admin peut ajouter, modifier les spécialités et l'affecté ses sous spécialités

- **Gestion d'établissements de santé**

Le super admin peut ajouter un établissement de santé et modifier son statut.

- **Suivi des activités**

Tableaux de bord KPI (médical, taux de réservation, satisfaction, etc.)

2.1.2.2 Besoins non fonctionnels

- **Documentation**

Utilisation d'Open API pour la description et la documentation des API, facilitant ainsi l'intégration des nouveaux développeurs et la maintenance du système. Fournir des guides d'utilisation les modules

- **Ergonomie et Convivialité**

Les interfaces de notre application doivent être modernes, compréhensibles, simples avec un design attrayant et facile à utiliser

- **Sécurité**

Doit fournir un accès personnalisé et une connexion sécurisée afin de garantir la confidentialité des données

- **Performance**

Favoriser une prise en main rapide des fonctionnalités et répondre à toutes les exigences des usagers d'une manière optimale

2.1.3 Modélisation des besoins

Notre projet offre diverses fonctionnalités. La figure suivante représente le diagramme du cas d'utilisation global, qui modélise les fonctionnalités des modules que nous avons développés.

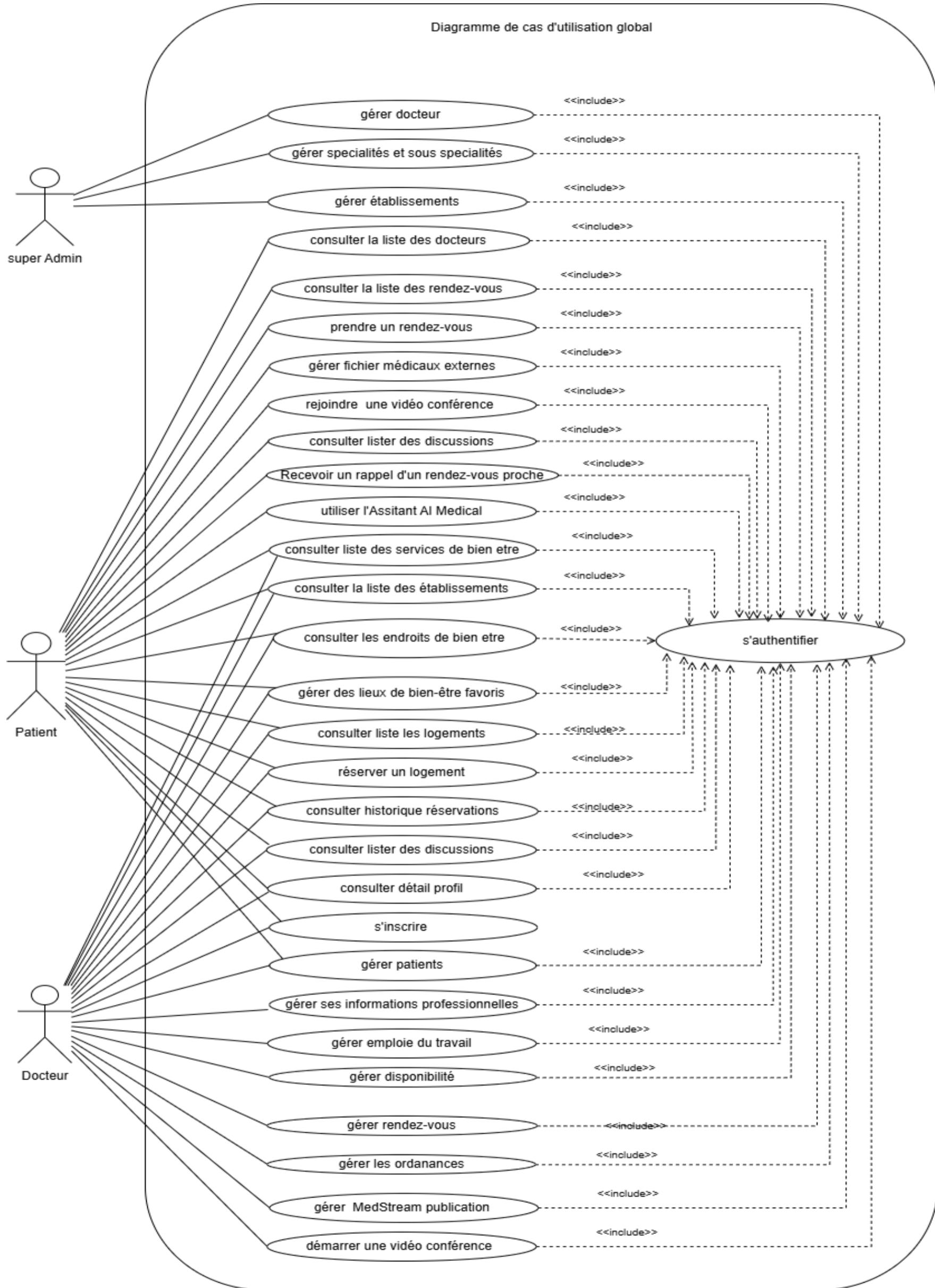


Figure 2.1 : Diagramme de cas d'utilisation globale

2.2 Pilotage du projet avec SCRUM

2.2.1 Équipe et rôles

La méthode SCRUM définit 3 rôles :

Tableau 2.1 : Équipe SCRUM

Rôle	Membre	Mission
Product Owner	Wael Majdoub	C'est le représentant du client, ayant une connaissance complète sur le produit à réaliser. Dans notre cas c'est le client lui-même.
Scrum Master	Laith mahdi	C'est le responsable de l'équipe, de la méthodologie SCRUM. Il gère l'organisation de l'équipe et assure le bon avancement du côté
Scrum Team	Malak Ben Othmane	C'est l'équipe de développement qui est responsable de la conception des interfaces intuitives et développement de l'application
	Isslem Ziedi	

2.2.2 Le Backlog du produit

Le backlog du produit est destiné à recueillir tous les besoins du client que l'équipe projet doit réaliser. Il contient donc la liste des fonctionnalités intervenant dans la constitution d'un produit, ainsi que tous les éléments nécessitant l'intervention de l'équipe projet.

Tous les éléments inclus dans le backlog sont classés par priorité indiquant l'ordre de leur réalisation. Et ce dernier a été classifier selon Moscow (Must, Should, Could, Won't) Les lettres majuscules de l'acronyme Moscow signifient (en anglais) :

- ✓ M (Must have) → Indispensable, doit être fait
- ✓ S (Should have) → Important mais pas critique
- ✓ C (Could have) → Optional, amélioration
- ✓ W (Won't have for now) → Pas prioritaire pour cette version

Tableau 2.2 : Product Backlog

Module	Fonctionnalité	ID	User Story	Priorité
Module Médical	Authentification et gestion des accès	1	En tant que docteur/patient , je peux créer un compte afin de bénéficier des services de l'application	M
		2	En tant qu' utilisateur , je veux me connecter ou déconnecter	M
		3	En tant qu' utilisateur , je peux récupérer mon mot de passe	M
	Gestion des spécialités et sous-spécialités	4	En tant que super admin , je peux gérer une spécialité et ses sous spécialités	M
		5	En tant que super-admin , je peux recevoir une notification indiquant qu'un docteur a créé un compte	M
		6	En tant que super admin , je peux modifier statut de docteur (active, banned, featured, rejected)	M
		7	En tant que docteur , je peux recevoir une notification indiquant que le super-admin a modifié mon statut	M
	Gestion des docteurs	8	En tant qu' utilisateur , je peux lister et consulter les détails des docteurs afin de voir ses qualifications et expériences et ses contacts	M
		9	En tant que docteur , je peux modifier les détails personnels et	M

			professionnelles de mon compte	
	10		En tant que super admin , je peux consulter les détails de vérification d'un compte docteur	M
	11		En tant que super admin , je peux créer un docteur	C
Gestion des établissements de santé	12		En tant que super admin/docteur , je peux ajouter un nouvel établissement de santé	M
	13		En tant que super admin , je peux activer ou désactiver l'apparence de l'établissement	S
	14		En tant que patient/super admin , je peux lister, rechercher et consulter les détails de l'établissement	M
	15		En tant qu' utilisateur , je peux filtrer les docteurs par plusieurs critères selon mon choix	M
Filtrage des affiliations	16		En tant qu' utilisateur , je peux filtrer les affiliations per pays, ville, spécialité disponibles, type	M
Gestion des patients	17		En tant que patient , je peux créer une fiche d'un patient pour chacun de mes proches	M
	18		En tant que patient , je peux créer le dossier médical (moi-même ou un proche), en y ajoutant des informations comme les antécédents médicaux (allergies, paramètres corporels)	M
	19		En tant que docteur/patient , je peux afficher la liste de mes patients/ de	M

			mes relations	
		20	En tant que docteur/patient , je peux consulter les détails d'un patient/l'un de mes relations, afin d'accéder à ses informations médicales,informations personnelles , rapport médical, etc.	M
	Gestion des ordonnances	21	En tant que docteur , je peux créer une ordonnance de tests ou de médicaments et l'affecter à l'un de mes patients	M
		22	En tant que docteur , je peux consulter la liste des ordonnances émises et les filtrer par type et statut	M
		23	En tant que docteur/patient , je peux consulter les détails d'une ordonnance	M
	Gestions des documents externes	24	En tant que patient , je peux gérer les documents de moi-même ou l'un de mes relations	M
		25	En tant que patient , je peux partager un fichier avec un docteur consulté qui va recevoir une notification	M
	Gestion emploie de temps docteur	26	En tant que docteur , je peux gérer mes heures de travail par semaine	M
	Gestion disponibilités de docteur	27	En tant que docteur , je peux modifier mes disponibilités	M
	Gestion des rendez vous	28	En tant que docteur/patient , je peux prendre un rendez-vous en ligne ou hors ligne pour un patient	M
		29	En tant que docteur , je peux accepter, reprogrammer et annuler	M

			des rendez-vous	
		30	En tant que patient , je peux recevoir une notification indiquant l'état de mon rendez-vous	M
		31	En tant que docteur/patient , je peux consulter la liste de mes rendez-vous, ainsi ses détails	M
	Rappel d'un rendez-vous	32	En tant que patient , je peux recevoir une notification d'un rappel de mes rendez-vous les plus proches	M
Module de communication et AI	Gestion des articles partagée « MedStream »	33	En tant que docteur , je peux lister mes articles par type (vidéo, audio, e-article), modifier et supprimer un article, ainsi que créer un nouveau	M
		34	En tant que patient , je peux consulter les articles partagés d'un docteur lors de « waiting room » afin d'attendre un docteur sans s'ennuyer	M
	Messagerie sécurisée	35	En tant que patient/docteur , je souhaite pouvoir communiquer afin d'échanger des informations et des conseils médicaux	M
	Démarrer une Vidéoconférence	36	En tant que docteur , je peux lancer une vidéoconférence pour commencer une consultation en ligne	M
	Rejoindre une Vidéoconférence	37	En tant que patient , je peux rejoindre une vidéoconférence pour avoir une consultation en ligne	M
	Contrôler Audio/Vidéo	38	En tant que participant à la vidéoconférence, je peux couper ou	M

			réactiver mon micro et activer ou désactiver ma caméra	
	Évaluer Consultation	39	En tant que patient , je peux donner mon avis concernant la consultation en ligne après avoir terminé la vidéoconférence	M
	Assistant AI « Nassouh »	40	En tant que patient , je peux communiquer avec un chat bot AI afin de m'aider à identifier la propre spécialité médicale que je dois visiter basée sur mes symptômes	M
		41	En tant que patient , je peux obtenir des conseils de bien-être sûrs pour adopter un mode de vie sain et équilibré	M
Module Tourisme	Consultation des logements	42	En tant qu' utilisateur , je souhaite consulter les logements les plus proches de ma localisation et celles les plus évaluées	M
	Consultation des détails d'un logement	43	En tant qu' utilisateur , je peux consulter les détails d'un logement, y compris les services plus proches, afin de m'aider à faire le meilleur choix en fonction de mes besoins	M
	Contact avec propriétaire	44	En tant qu' utilisateur , je souhaite contacter le propriétaire par message ou par téléphone	M
	Consultation disponibilité d'un logement	45	En tant qu' utilisateur , je souhaite connaître la disponibilité d'un logement	M
	Consultation chambres	46	En tant qu' utilisateur , je souhaite voir quelles chambres sont disponibles dans un logement	M

			partagé	
	Filtrage des logements	47	En tant qu'utilisateur, je souhaite filtrer les logements afin d'affiner ma recherche selon mes critères	M
	Visualisation cartographique	48	En tant qu' utilisateur , je souhaite visualiser les résultats du filtrage sur une carte interactive, avec les prix des logements affichés directement sur celle-ci	M
	Gestion des logements favoris	49	En tant qu' utilisateur , je souhaite ajouter/ supprimer un logement à ma liste de favoris	M
	Réservation d'un hébergement	50	En tant qu' utilisateur , je souhaite pouvoir réserver un logement entier ou une chambre d'un logement partagé pour mon séjour	M
	Consultation d'Historique des réservations	51	En tant qu' utilisateur , je souhaite consulter mes réservations passées	M
	Consultation de la liste des endroits du bien-être	52	En tant que patient , je peux consulter une liste des endroits de bien-être les plus évalués et celles les plus proches de moi	M
	Filtrage des endroits du bien-être	53	En tant que patient , je peux filtrer les lieux par type, activités, services et localisation, et visualiser les résultats directement sur une carte géographique	M
	Consultation des détails d'un endroit	54	En tant que patient , je veux consulter les détails d'un lieu (service proposés, offres, etc....)	M
	Consultation des détails d'une	55	En tant que patient , je veux consulter les détails d'une offre	M

	offre d'un endroit		spécifique	
	Gestion des endroits de bien-être favoris	56	En tant que patient , je peux ajouter un endroit à ma liste de favoris ou le retirer, afin de sauvegarder mes préférences et faciliter mes choix futurs	M

2.2.3 Planification des releases / sprints

Tableau 2.3 : Planification des sprints

Release	Sprints	Fonctionnalités	Durée
R1 : Module Médical	Sprint 1 : Fondations du module médical	<ul style="list-style-type: none"> • Authentification • Gestion des spécialités et sous spécialités • Gestion des docteurs • Gestion des établissements • Filtrage des docteurs • Filtrage des établissements 	4 semaines
	Sprint 2 : Services patients et rendez-vous	<ul style="list-style-type: none"> • Gestion des patients • Gestion des documents médicaux • Gestions des disponibilités des docteurs • Gestion des rendez-vous 	4 semaines
R2 : Module de communication et AI	Sprint 3 : Services de communication	<ul style="list-style-type: none"> • Gestion des articles partagée • Messagerie sécurisée • Vidéoconférence • Assistant AI 	4 semaines
R3 : Module Tourisme	Sprint 4 : Services de bien-être et d'hébergement	<ul style="list-style-type: none"> • Gestion des services de tourisme • Filtrer les activités • Gestion des logements • Gestion des réservations en ligne des logements • Filtrer logements 	4 semaines

Nous avons choisi de diviser le projet en 4 sprints, tel que chaque sprint présente un module de l'application

La planification des sprints est ainsi schématisée dans le graphique suivant :

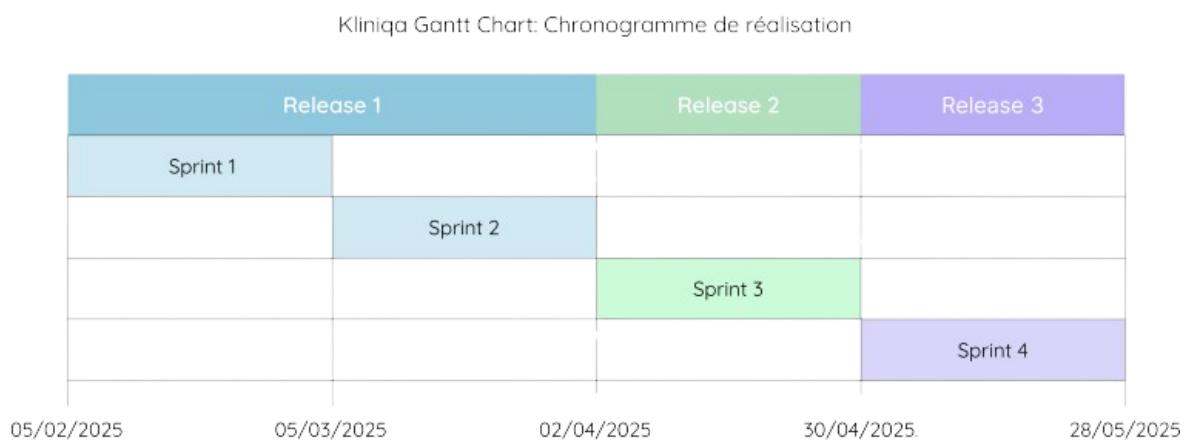


Figure 2.2 : Diagramme de Gant planification des sprints

2.3 Environnement de travail

Cette partie décrit l'environnement matériel et logiciels en précisant les outils logiciels, Framework et langages de programmation utilisés.

2.3.1 Environnement matériel

Nous commençons par lister les caractéristiques minimales de l'environnement matériel de réalisation du projet dans le tableau ci-après. (Tableau)

Tableau 2.2 : Environnement matériel

Ordinateur		
Propriétaire	Isslem Ziedi	Malak Ben Othmane
Modèle	Dell VOSTRO 3400	ASSUS TUF GAMING F15
Processeur	Core i5	Core i5
Ram	16 Go	32 Go
Disque Dur	1256 Go	SSD 550 GO
Système D'exploitation	Windows 11 pro	Windows 11 pro

2.3.2 Environnement technologies

Cette partie décrit l'environnement technologique en précisant les outils logiciels, les langages de programmation et les bibliothèques utilisés

2.3.2.1 Outils logiciels

➤ Outils logiciels de conception

- **Draw.io** : un outil de création des diagrammes tel que les diagramme UML, des organigrammes et des autres structures personnalisées



Figure 2.3 : Logo draw.io

➤ Outils logiciels de design

- **Figma** : un éditeur de graphiques vectoriels et un outil de prototypage.



Figure 2.4 : Logo figma

➤ Outils logiciels de développement

- **Visual Studio Code** : un éditeur de code extensible développé par Microsoft pour Windows, Linux et MacOs. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe



Figure 2.5 : Logo VS Code

- **Android Studio** : un environnement de développement pour développer des applications mobiles Android. Il est basé sur IntelliJ IDEA et utilise le moteur de production Gradle



Figure 2.6 : Logo Android Studio

- **Postman** : permet de construire et d'exécuter des requêtes HTTP, de les stocker dans un historique afin de pouvoir les rejouer, mais surtout de les organiser en collections



Figure 2.7 : Logo Postman

- **Github** : un logiciel libre de forge basé sur git, il est un système de suivi des bugs, d'intégration et de livraison continue



Figure 2.8 : Logo github

➤ Outils logiciels de communication

- **Slack** : une plateforme de communication collaborative propriétaire (SaaS) ainsi qu'un logiciel de gestion de projets



Figure 2.9 : Logo slack

- **Plane** : un logiciel de gestion de projets et de connaissances qui constitue un multiplicateur de force pour vos flux de travail, méthodologies et meilleures pratiques existants

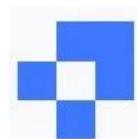


Figure 2.10 : logo Plane

2.3.2.2 Technologies et frameworks

- **Mobile :** **Flutter** est un kit de développement logiciel d'interface utilisateur open-source créé par Google. Il est utilisé pour développer des applications pour Android, iOS, Linux, Mac, Windows, Google Fuchsia et le web à partir d'une seule base de code.



Figure 2.11 : Logo Flutter

- **Web :** **NextJs** est un framework gratuit et open source s'appuyant sur la bibliothèque JavaScript, React et sur la technologie NodeJs permettant de créer des applications web full-stack



Figure 2.12 : logo Next.js

➤ **Backend**

- **Hono** : est un framework léger et rapide pour créer des API web et des applications backend en JavaScript/TypeScript. Il se distingue par sa gestion efficace des middlewares, offrant un traitement fluide des requêtes HTTP. Conçu pour les Edge Functions et les plateformes serverless, il s'intègre parfaitement aux architectures modernes, comme les apps mobiles avec Flutter



Figure 2.13 : Logo Hono

- **Bun** : un runtime JavaScript rapide pour exécuter du code, souvent utilisé à la place de Node.js. Il améliore les performances des applications backend, comme avec Hono, et fonctionne bien avec Flutter pour les applications mobiles



Figure 2.14: logo Bun

- **Drizzle ORM** : est un ORM (Object-Relational Mapping) léger et moderne pour TypeScript/JavaScript. Il prend en charge PostgreSQL, MySQL, SQLite et les solutions serverless. Il offre un typage fort, de hautes performances et une gestion simple des migrations



Figure 2.15 : logo Drizzle-ORM

➤ Base de données

- **PostgreSQL (Neon)** : est un service de base de données PostgreSQL sans serveur qui offre une mise à l'échelle automatique et une gestion efficace. Il simplifie le déploiement des bases de données PostgreSQL avec un minimum de maintenance



Figure 2.16 : Logo Neon PostgreSQL

- **Supabase** : est une plateforme open source qui offre aux développeurs des outils comme l'authentification et le stockage en temps réel. Contrairement à Firebase, Supabase donne plus de liberté, un contrôle total des données et une interopérabilité accrue, évitant les limitations des solutions propriétaires



Figure 2.17 : Logo supabase

2.3.2.3 Protocoles

- **HTTP (HyperText Transfer Protocol)** : permet d'échanger des requêtes et des réponses entre le client et le serveur, en utilisant des méthodes RESTful (GET, POST, PUT, DELETE) pour manipuler les ressources



Figure 2.18 : Http protocole

2.3.2.4 Bibliothèques

➤ Backend :

- **Zod (ZodValidator)** : une bibliothèque TypeScript permettant de valider des schémas de données de manière strictement typée. Elle est Très utile pour assurer la sécurité des entrées utilisateur et prévenir les erreurs



Figure 2.19 : logo ZodValidator

➤ Web :

- **shadcn/ui** : permet d'importer des composants React prêts à l'emploi (cards, modals, boutons, formulaires) basés sur Tailwind CSS et Radix, pour accélérer la création d'interfaces cohérentes et accessibles



Figure 2.20 : logo shadcn/ui

- **TanStack Query** : permet de gérer la récupération, la mise en cache et la synchronisation des données asynchrones dans les applications React, en simplifiant la logique liée aux requêtes HTTP et aux états de chargement



Figure 2.21 : logo tanstack Query

➤ Mobile :

- **Dio** : est une puissante bibliothèque HTTP pour effectuer des requêtes réseau avec des fonctionnalités comme l'annulation de requêtes, l'interception et la mise en cache



Figure 2.22 : Dio package

- **LiveKit SDK** : une plateforme de communication en temps réel pour les applications de streaming vidéo/audio qui supporte WebRTC, le multi-participant et une faible latence, idéal pour des appels en direct ou des conférences



Figure 2.23 : Livekit Client

2.4 Architecture proposée

2.4.1 Présentation de l'architecture « Modular Monolith with Domain-Driven Design »

- **Domain-Driven Design (DDD):**

C'est une approche de conception logicielle qui met l'accent sur la compréhension du domaine métier et son intégration dans le développement logiciel. L'idée principale est que le logiciel doit être conçu en fonction des besoins et des règles du métier, plutôt que d'être dicté uniquement par des considérations techniques

- **Modular Monolith**

C'est une approche hybride qui combine les avantages des architectures monolithiques et modulaires. Contrairement à un monolithe classique, où tout le code est fortement couplé, un monolithe modulaire divise l'application en modules indépendants tout en restant une seule unité déployable

- **Modular Monolith with Domain-Driven**

Cette architecture combine les principes du DDD avec la modularité d'un monolithe modulaire. Cela signifie que chaque module est conçu autour d'un domaine métier bien défini, ce qui permet une meilleure séparation des responsabilités et une évolutivité plus efficace

Le tableau suivant présente les principales différences entre ces trois architectures :

Tableau 2.3 : Comparaison entre architectures

Critère	Monolith Modulaire	Domain-Driven	Architecture utilisé
Organisation	Par couches techniques	Par domaines métier	Par domaines métier
Séparation	Controllers/Services /Models	Bounded contexts	Medical/Tourism /Communication
Base de données	Une seule DB	DB par domaine	DB spécialisées par domaine
Logique métier	Dispersée	Centralisée par domaine	Encapsulée par domaine
Déploiement	Unique	Séparé	Unique
Communication	Directe	API/Messages	Directe

2.4.2 Avantage de cette architecture

❖ Avantages Techniques

Tableau 2.4 : Avantages techniques de l'architecture utilisée

Avantage	Bénéfice	Impact Business
Développement rapide	Un seul serveur à gérer	Time-to-market réduit
Débogage simplifié	Logs centralisés	Résolution bugs rapide
Coûts minimaux	Infrastructure simple	Budget optimisé
Déploiement simple	Une seule commande	Mise en prod facile
Équipe unifiée	Code base partagée	Collaboration fluide

❖ Avantages Architecturaux

Tableau 2.5 : Avantages architecturaux de l'architecture utilisée

Avantage	Description	Valeur Ajoutée
Modularité	Domaines bien séparés	Maintenance facilitée
Sécurité Centralisée	Middleware unifié	Sécurité cohérente
Données Spécialisées	DB par domaine	Performance optimisée
Évolutivité	Prêt pour microservices	Croissance préparée

2.4.3 Comparaison et évaluation des architectures

Pour refléter l'importance stratégique de Product Owner, Nous allons évaluer chaque critère sur une échelle de 1 à 5 :

Coût : 4/5 - maîtrise budgétaire essentielle

Vitesse Développement : 5/5 - Priorité absolue, time-to-market critique

Complexité : 3/5 - Impact sur la maintenance long terme

Évolutivité : 4/5 - Capacité d'adaptation aux besoins futurs

Tel que : Score total = Complexité×3 + Coût×4 + Vitesse×5 + Évolutivité×4

Tableau 2.6 : Evaluation des architectures

Architecture	Complexité	Coût	Vitesse Dev	Évolutivité	Score Total
Monolithique	Moyenne (4/5)	Faible (5/5)	Rapide (5/5)	Excellent (5/5)	77/80
Monolithique Classique	Faible (5/5)	Minimal (5/5)	Rapide (4/5)	Limitée (2/5)	63/80
Microservices	Élevée (2/5)	Élevée (2/5)	Lente (2/5)	Maximale (5/5)	44/80

- L'architecture **Modular Monolith with Domain-Driven** est le meilleur choix pour **Kliniqa**, offrant le parfait équilibre entre rapidité de développement, coûts maîtrisés et évolutivité pour répondre aux besoins du Product Owner

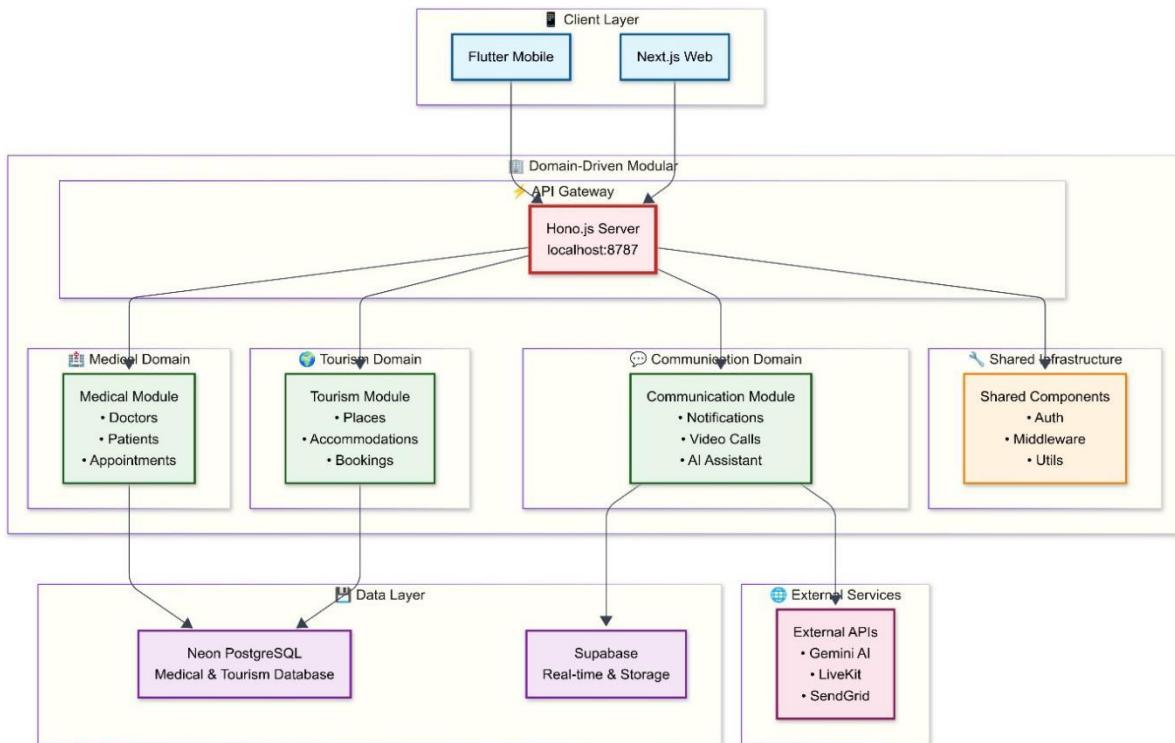


Figure 2.24 : Architecture Modular Monolith with DD

3. Chapitre 3 : Release 1

« Module Médical »

3.1 Développement du Sprint 1 « Fondations du module médical »

3.1.1 Backlog du Sprint 1

Dans ce tableau nous définissons les tâches qui seront effectuées pour chaque user story durant le sprint 1

Tableau 3.1 : Backlog du sprint 1

Fonctionnalité	ID	User Story	Tâches	Story Point
Authentification et gestion des accès	1	En tant que docteur/patient , je peux créer un compte afin de bénéficier des services de l'application	-Créer le/les page(s) d'inscription avec un Formulaire personnalisé, adapté à chaque rôle d'utilisateur (Patient/Docteur)	Difficile
	2	En tant qu' utilisateur , je veux me connecter ou déconnecter	- Crée une page de connexion avec un formulaire de saisie login et de mot de passe	Moyenne
	3	En tant qu' utilisateur , je peux récupérer mon mot de passe	- Mettre en place la logique d'inscription spécifique à chaque rôle (validation des champs, enregistrement en base, envoi d'e-mails) - Mettre en place la logique de connexion (vérification des Identifiants, redirection vers l'espace dédié) - Mettre en place la logique de connexion via services tiers Google - Mettre en place la logique de réinitialisation de mot de passe	Moyenne

Gestion des spécialités et sous spécialités	4	En tant que super admin , je peux gérer une spécialité et ses sous spécialités	<ul style="list-style-type: none"> - Implémenter une page pour afficher la liste des spécialités en Next.js/ Flutter - Implémenter la fonctionnalité d'ajout d'une spécialité avec ses sous spécialités - Implémenter la fonctionnalité de consultation des détails d'une spécialité - Implémenter la fonctionnalité de modification d'une spécialité 	Difficile
Administration des profils de docteurs	5	En tant que super admin, je peux recevoir une notification indiquant qu'un docteur à créer un compte.	<ul style="list-style-type: none"> - Intégrer la logique d'envoi d'une notification dans l'API d'inscription du médecin en cas de succès de l'inscription. 	Facile
	6	En tant que super admin , je peux modifier statut de docteur (active, banned, featured, rejected)	<ul style="list-style-type: none"> - Implémenter la logique de modification statut docteur et l'envoi des emails personnalisé selon statut -Créer le composant de liste déroulante statuts docteur dans l'interface web 	Moyenne
	7	En tant que docteur, je peux recevoir une notification indiquant que le super admin a modifié mon statut	<ul style="list-style-type: none"> - Intégrer la logique d'envoi d'une notification dans l'API modification statut docteur en cas de succès de modification 	Facile

Gestion des établissements	8	En tant que super admin/docteur , je peux ajouter un nouvel établissement de santé	<ul style="list-style-type: none"> - Implémenter la fonctionnalité d'ajouter un nouvel établissement de la santé et la visualiser sur la carte géo . - Implémenter le formulaire de la boite de dialogue d'ajout d'un établissement de santé . 	Difficile
	9	En tant que super admin , je peux activer ou désactiver l'apparence de l'établissement	<ul style="list-style-type: none"> - Implémenter l'API de modification du statut de l'établissement. - Créer le composant de liste déroulante statut d'établissement dans l'interface web 	Facile
	10	En tant que super admin/patient , je peux lister, rechercher et consulter les détails de l'établissement	<ul style="list-style-type: none"> - Implémenter une page pour afficher la liste des Établissements de la santé en Flutter/NextJs - Implémenter l'API de récupérer la liste d'établissement. -Implémenter la fonctionnalité de recherche d'un établissement - Implémenter une page pour afficher les détails d'un Établissement de la santé en Flutter/NextJs - Implémenter l'API de récupérer les détails d'établissement de la santé 	Difficile

Gestion des docteurs	11	<p>En tant qu'utilisateur, je peux lister et consulter les détails des docteurs afin de voir ses qualifications et expériences etc. ...</p>	<ul style="list-style-type: none"> - Implémenter une page pour afficher la liste des docteurs en Flutter/NextJs - Implémenter une page pour afficher les détails d'un docteur en Flutter/NextJs - Implémenter l'API de récupérer les détails d'un docteur - Implémenter la fonctionnalité de recherche d'un docteur 	Difficile
	12	<p>En tant que docteur, je peux modifier les détails personnels et professionnelles de mon compte</p>	<ul style="list-style-type: none"> - Implémenter la fonctionnalité de modifier les langages parlé par un docteur (ajouter, supprimer) - Implémenter la fonctionnalité de modifier les diplômes acquis par un docteur (ajouter, supprimer, modifier) - Implémenter la fonctionnalité de modifier les spécialités et les sous spécialités pratiquées par un docteur (ajouter, supprimer) - Implémenter la fonctionnalité de créer compte d'un docteur par super-admin en web 	Difficile
	13	<p>En tant que super admin, je peux consulter les détails de vérification d'un compte docteur</p>	<ul style="list-style-type: none"> -Implémenter l'API de récupération des fiches de vérification d'un médecin -Développer la page web d'affichage des détails de vérification d'un médecin. 	Facile

Filtrage des docteurs	14	En tant qu' utilisateur , je peux filtrer les docteurs par plusieurs critères selon mon choix.	- Implémenter la page de filtrage des docteurs et celle de résultat de filtrage en NextJs/Flutter -Implémenter l'API de filtrage des docteurs.	Moyenne
Filtrage des affiliations	15	En tant qu' utilisateur , je peux filtrer les affiliations par pays, ville, spécialité disponible, type	- Implémenter la page de filtrage des établissements et celle de résultat de filtrage en Flutter -Implémenter l'API de filtrage d'un établissement	Facile

3.1.2 Analyse

Pour la spécification fonctionnelle du sprint 1, nous présentons les diagrammes de cas d'utilisation global et détaillés

3.1.2.1 Diagramme de cas d'utilisation globale de sprint 1

Ce diagramme de cas d'utilisation présente les différentes fonctionnalités que le docteur et le patient peuvent utiliser durant le sprint 1

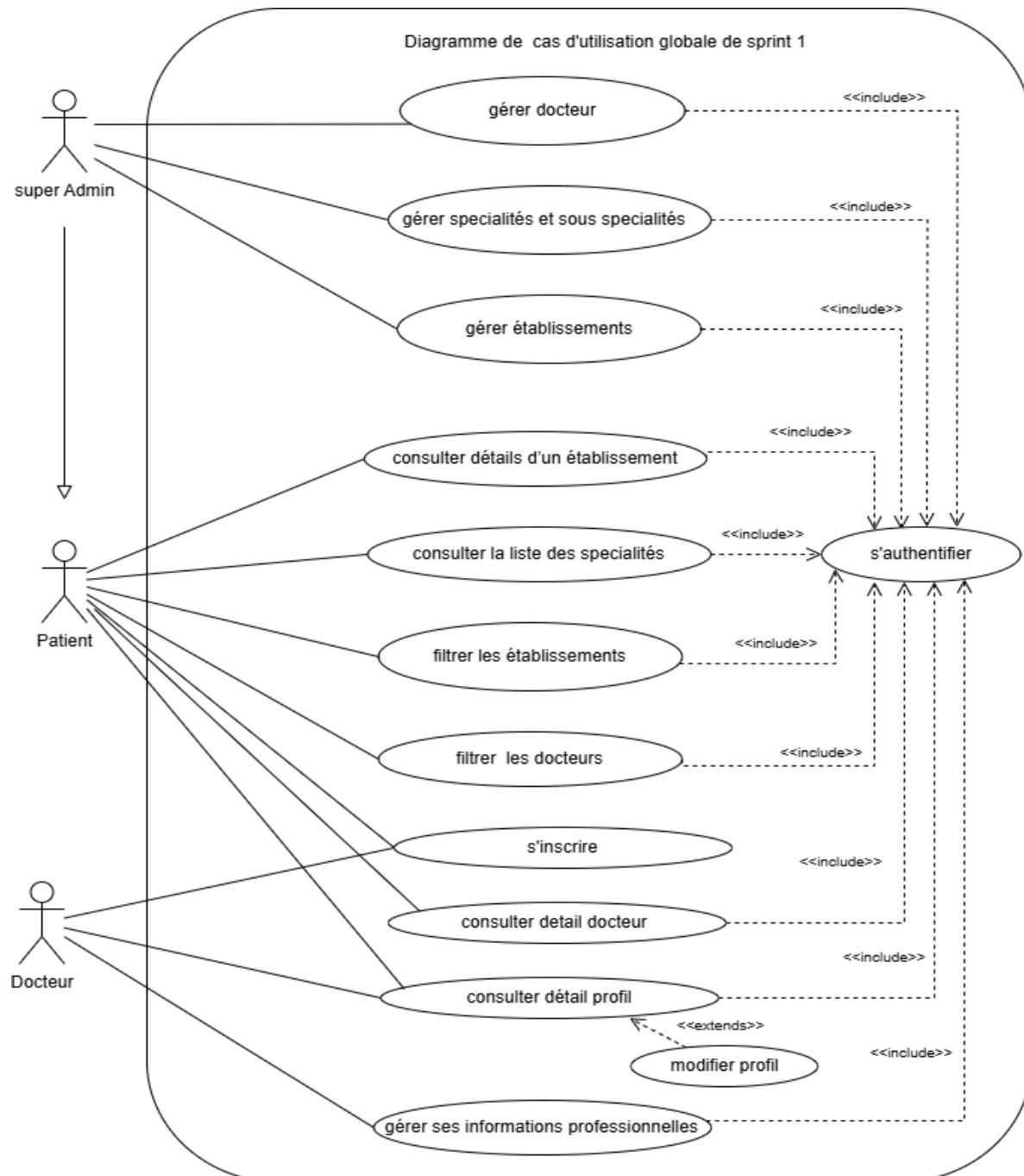


Figure 3.1 : Diagramme de cas d'utilisation de sprint 1

3.1.2.2 Diagrammes de cas d'utilisation détaillés

Dans la suite, nous détaillons les diagrammes de cas d'utilisation pour les principales fonctionnalités

- Diagramme de cas d'utilisation détaillée « filtrer docteur »**

Cette figure présente les détails du cas d'utilisation filtrer docteur

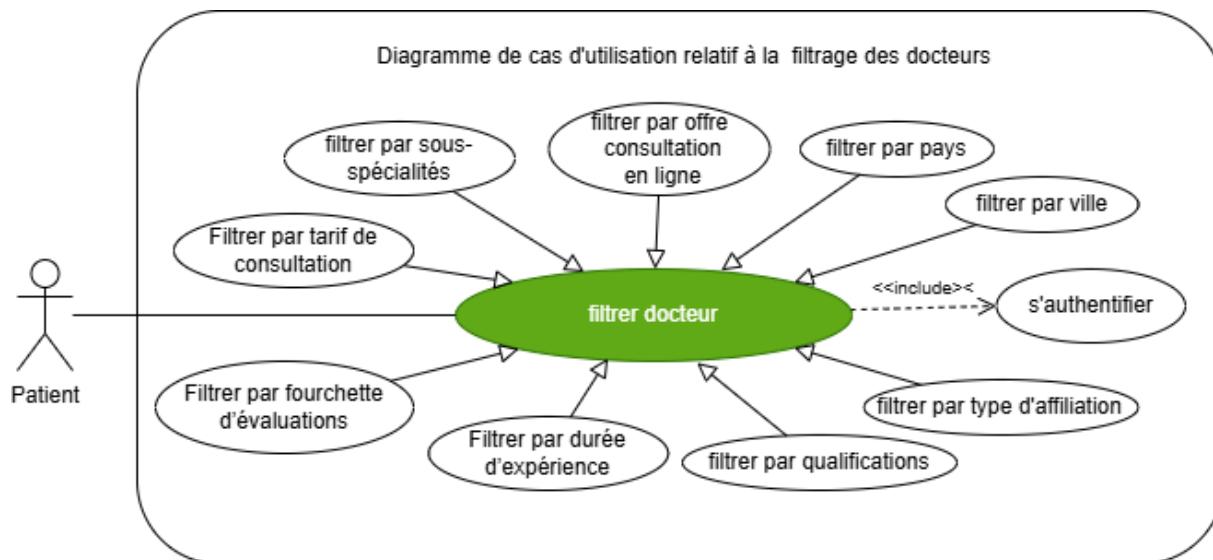


Figure 3.2 : Diagramme de cas d'utilisation détaillée de filtrage docteur

- Diagramme de cas d'utilisation détaillée « filtrer établissement »**

Cette figure présente les détails du cas d'utilisation filtrer établissement

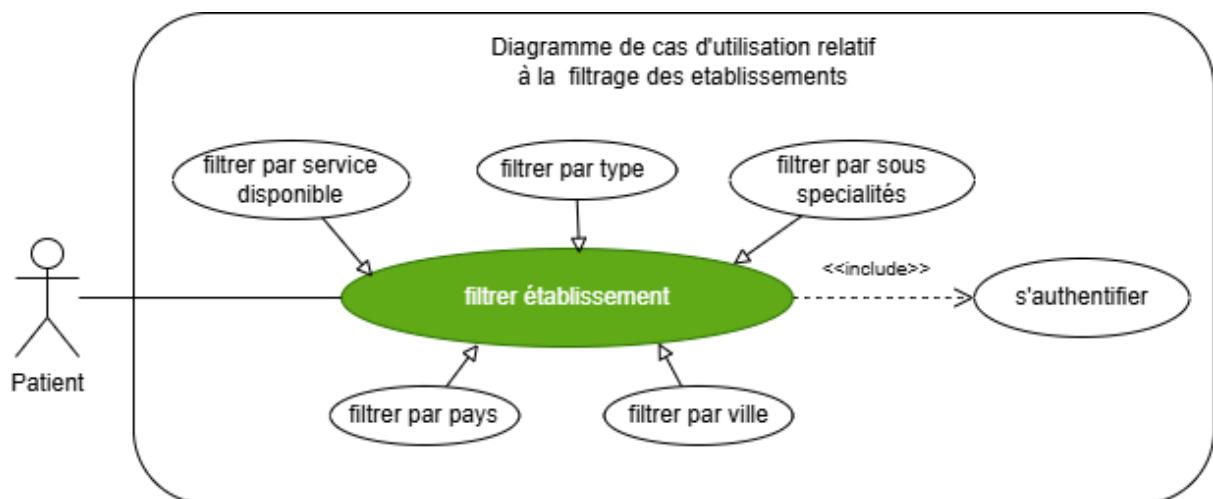


Figure 3.3 : Diagramme de cas d'utilisation détaillée de filtrage établissement

- **Diagramme de cas d'utilisation détaillée « gérer établissement »**

Cette figure présente les détails du cas d'utilisation gérer établissement

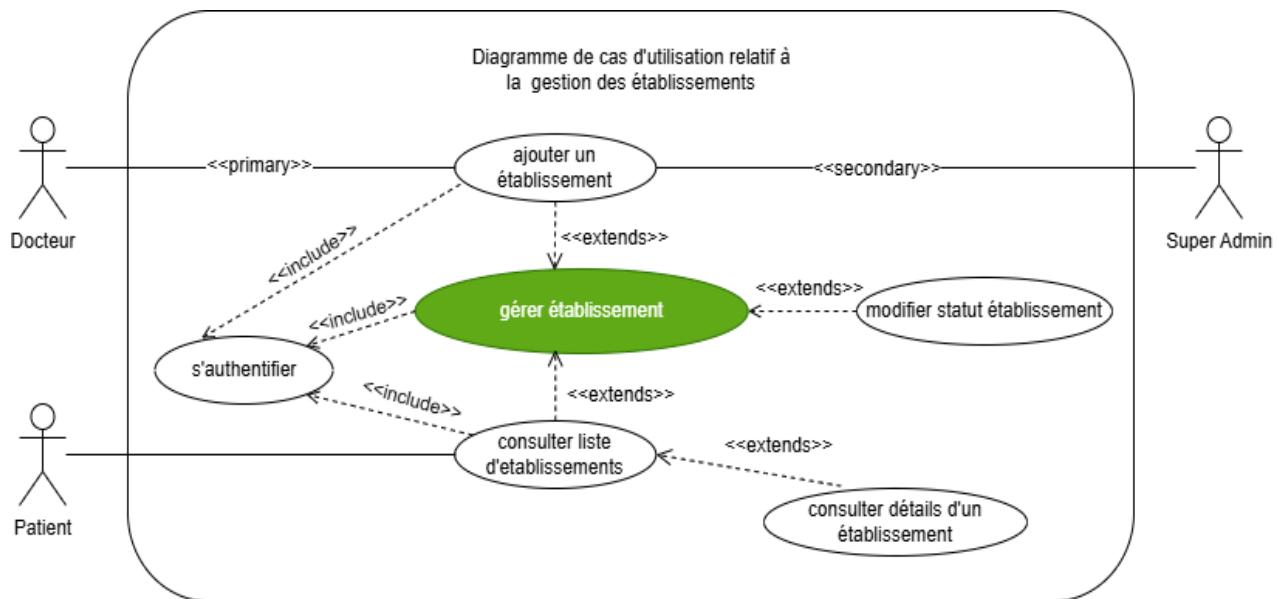


Figure 3.4 : Diagramme de cas d'utilisation détaillée de gestion établissement

- **Diagramme de cas d'utilisation détaillée « gérer docteur »**

Cette figure présente les détails du cas d'utilisation gérer docteur

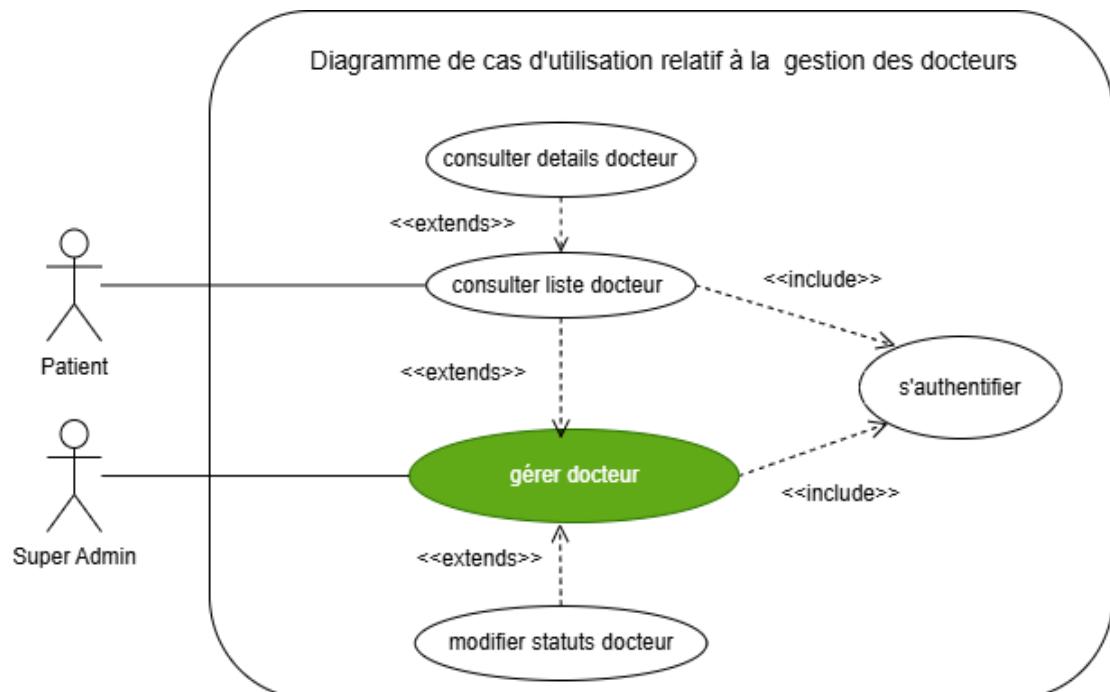


Figure 3.5 : Diagramme de cas d'utilisation détaillée de gestion docteur

- **Diagramme de cas d'utilisation détaillée « gérer information professionnelle »**

Cette figure présente les détails du cas d'utilisation gérer information professionnelle

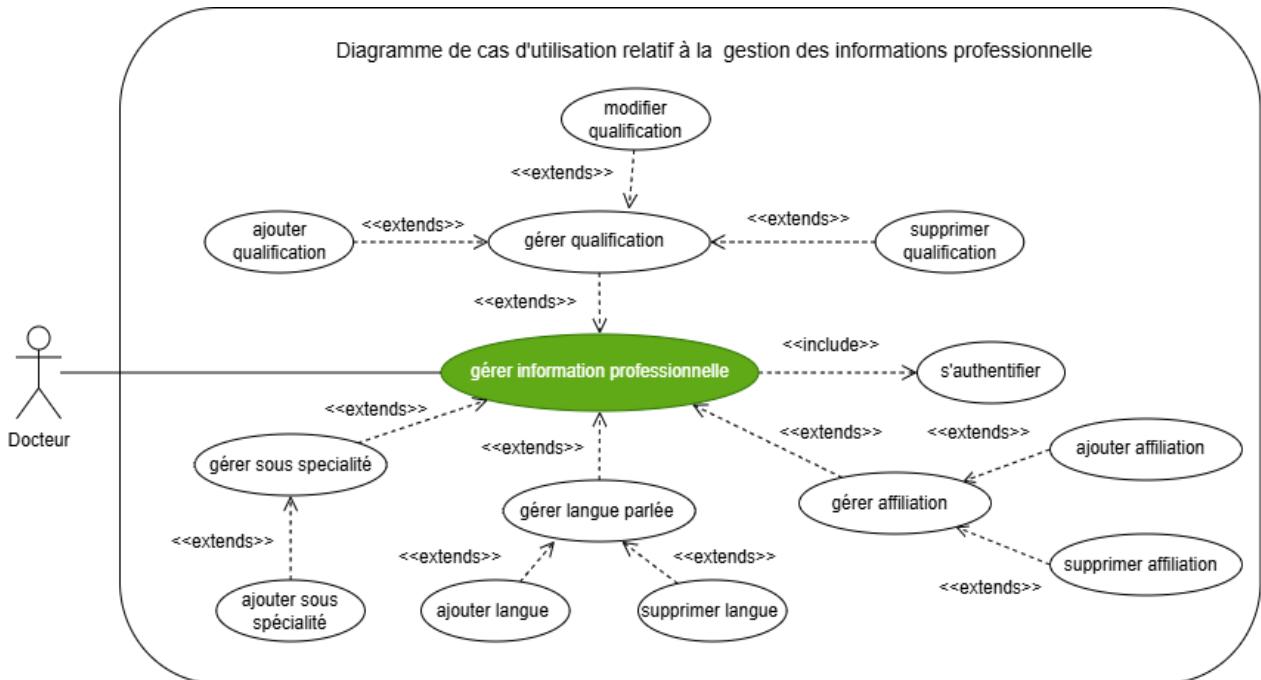


Figure 3.6 : Diagramme de cas d'utilisation détaillée de gestion d'information professionnelle

- **Diagramme de cas d'utilisation détaillée « gérer spécialité et sous spécialité »**

Cette figure présente les détails du cas d'utilisation gérer spécialité et sous spécialité

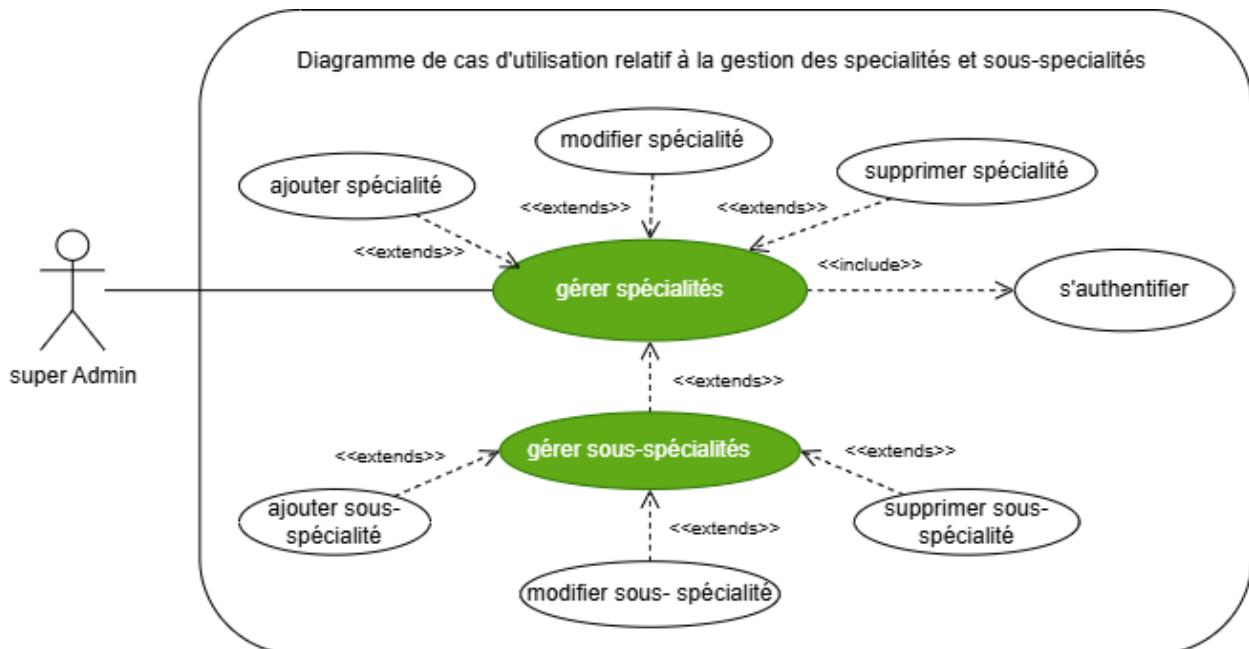


Figure 3.7: Diagramme de cas d'utilisation détaillée de gestion de spécialités

3.1.3.1 Description textuelle des cas d'utilisation

- **Description textuelle de cas d'utilisation « Ajouter une affiliation »**

Tableau 3.2 : Description textuelle du cas d'utilisation « Ajouter une affiliation »

Cas d'utilisation : Ajouter une affiliation
Acteur(s) : Super-Admin
Objectif : Permettre au super admin d'ajouter un nouvel établissement de santé (affiliation) au système et de le visualiser sur la carte
Pré-condition(s) : Le super-admin doit être authentifié
Post-condition(s) : L'affiliation est créée et enregistrée dans le système
Scénario principal : <ol style="list-style-type: none">1. Le super admin accède à l'interface de gestion des affiliations2. Le super admin clique sur le bouton "Ajouter une affiliation"3. Le système affiche un formulaire d'ajout d'affiliation4. Le super admin saisit le nom de l'affiliation dans le champ de recherche5. Le système propose des suggestions d'affiliations via autocomplétions6. Le super admin sélectionne l'affiliation correspondante dans la liste7. Le système affiche automatiquement la localisation de l'affiliation sur la carte8. Le système récupère et affiche les informations disponibles : informations générales, informations de contact (email, téléphone, fax, site web), spécialité(s), informations de localisation (pays, ville, adresse)9. Le super admin sélectionne le type d'affiliation (hôpital, cabinet, clinique)10. Le super admin importe des images de l'affiliation11. Le super admin saisit une description pour l'affiliation12. Le super admin définit le statut de l'affiliation (actif/inactif)13. Le super admin valide et enregistre l'affiliation <p>Le système confirme l'ajout de l'affiliation</p>
Scénario alternatif : <ol style="list-style-type: none">4.1 Aucune information trouvée pour le nom saisi :<ul style="list-style-type: none">○ Le système affiche un message indiquant qu'aucune affiliation n'est disponible avec ce nom8.1 Informations incomplètes :

- Les champs s'affichent vides lorsqu'aucune donnée n'est disponible
- Le super admin doit compléter manuellement les informations manquantes

9.1 Validation en temps réel des champs saisis :

- Le système vérifie la saisie au fur et à mesure
- Les erreurs s'affichent immédiatement

13.1 Informations obligatoires manquantes :

- Le système signale les champs obligatoires non remplis
- Le super admin doit compléter ces champs avant de pouvoir valider

- **Description textuelle de cas d'utilisation « Filtrer un docteur »**

Tableau 3.3 : Description textuelle de cas d'utilisation « Filtrer un docteur »

Cas d'utilisation : Filtrer un docteur
Acteur(s) : Patient
Objectif : Permettre aux patients de filtrer les docteurs d'une spécialité sélectionnée selon différents critères
Pré-condition(s) : <ul style="list-style-type: none">• Le patient doit avoir accès à la page des spécialités• Des docteurs doivent être enregistrés dans le système
Post-condition(s) : <p>Une liste filtrée de docteurs correspondant aux critères de recherche est affichée</p>
Scénario principal : <ol style="list-style-type: none">1. Le patient accède à la page des spécialités2. Le patient sélectionne une spécialité spécifique3. Le système affiche la liste des docteurs exerçant cette spécialité4. Le patient utilise les options de filtre disponibles : sous-spécialité(s), langue(s) parlée(s), type(s) de diplôme(s) acquis, plage des années d'expérience(min-max), plage de prix de consultation (min-max), localisation (pays, ville), plage des avis moyens, lieu d'exercice médical (hôpital, clinique, cabinet privé), offre des consultations en ligne (oui/non)

5. Le patient appuie sur le bouton "Appliquer les filtres"
6. Le système traite les critères de filtrage
7. Le système affiche la liste des docteurs correspondant aux critères sélectionnés

Scénario alternatif :

3.1 Aucun docteur pour la spécialité sélectionnée :

- Le système affiche un message indiquant qu'aucun docteur n'exerce cette spécialité

7.1 Aucun résultat après filtrage :

- Le système affiche un message indiquant qu'aucun docteur ne correspond aux critères de filtrage

3.1.3 Conception

3.1.3.2 Diagramme de classe de sprint 1

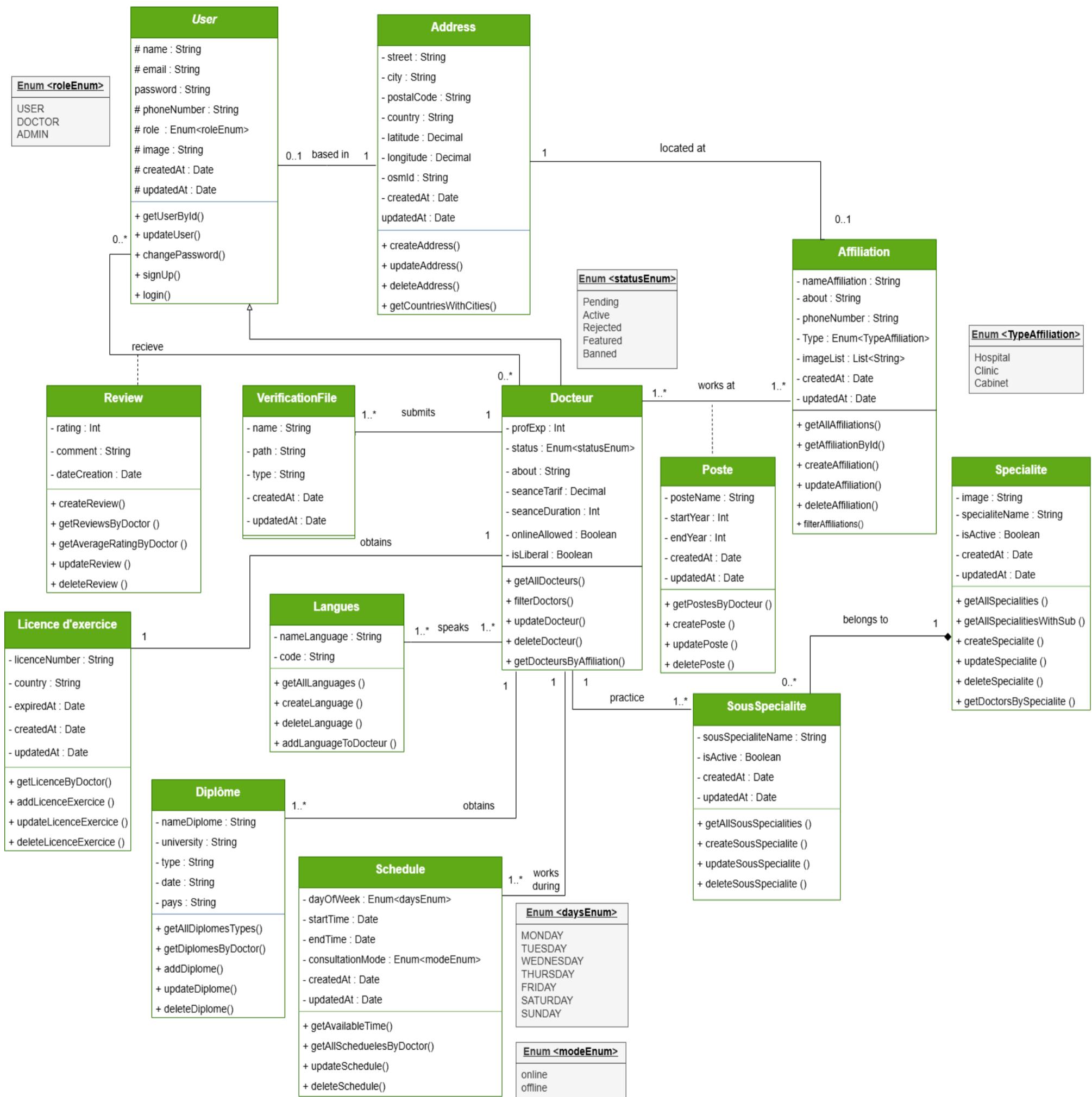


Figure 3.8 : Diagramme de classe sprint 1

3.1.3.3 Diagrammes de séquence de sprint 1

- Diagramme de séquence « ajout un établissement »

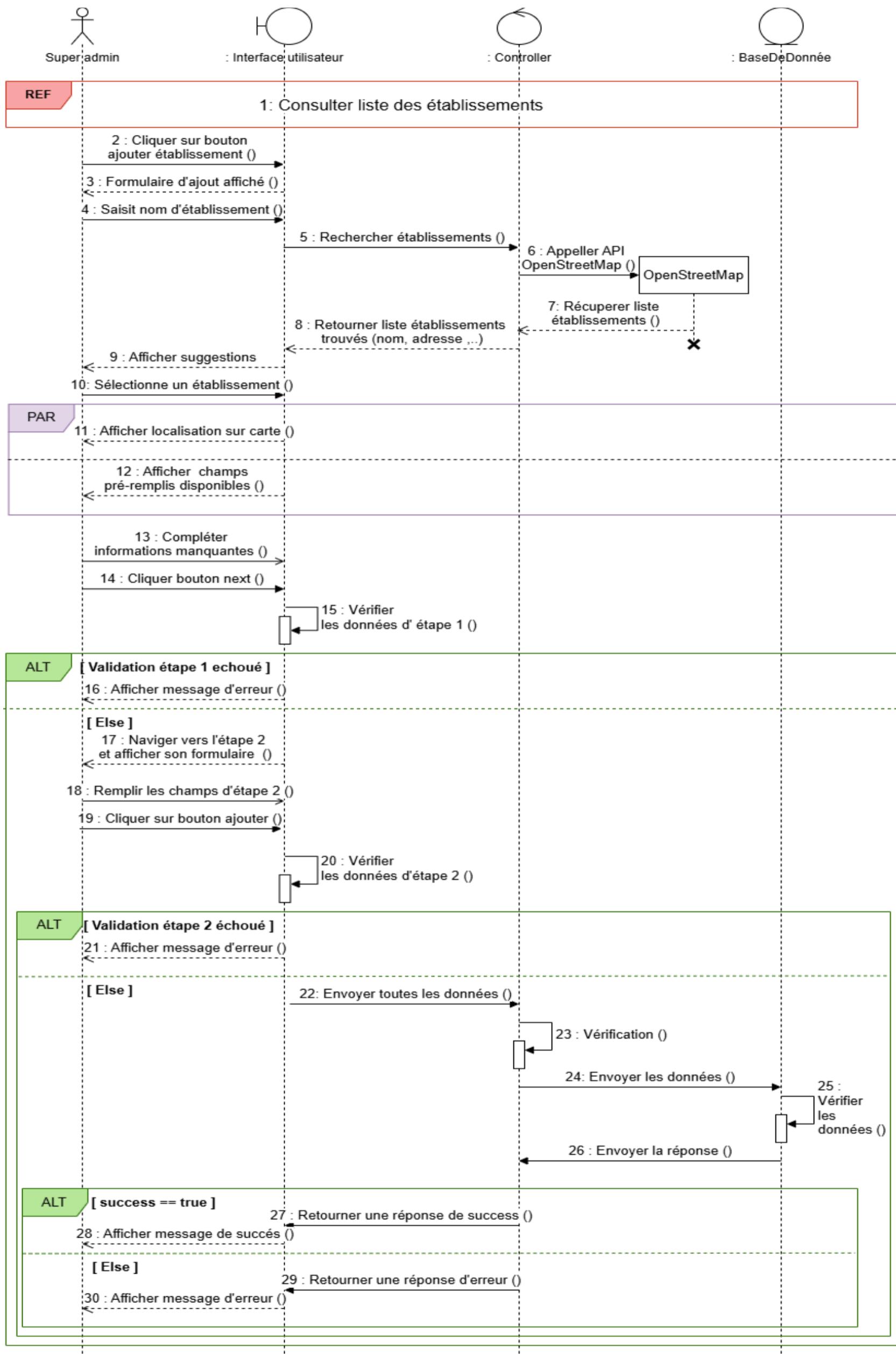


Figure 3.9 : Diagramme de séquence « ajout un établissement »

- Diagramme de séquence modifier info professionnelle d'un docteur

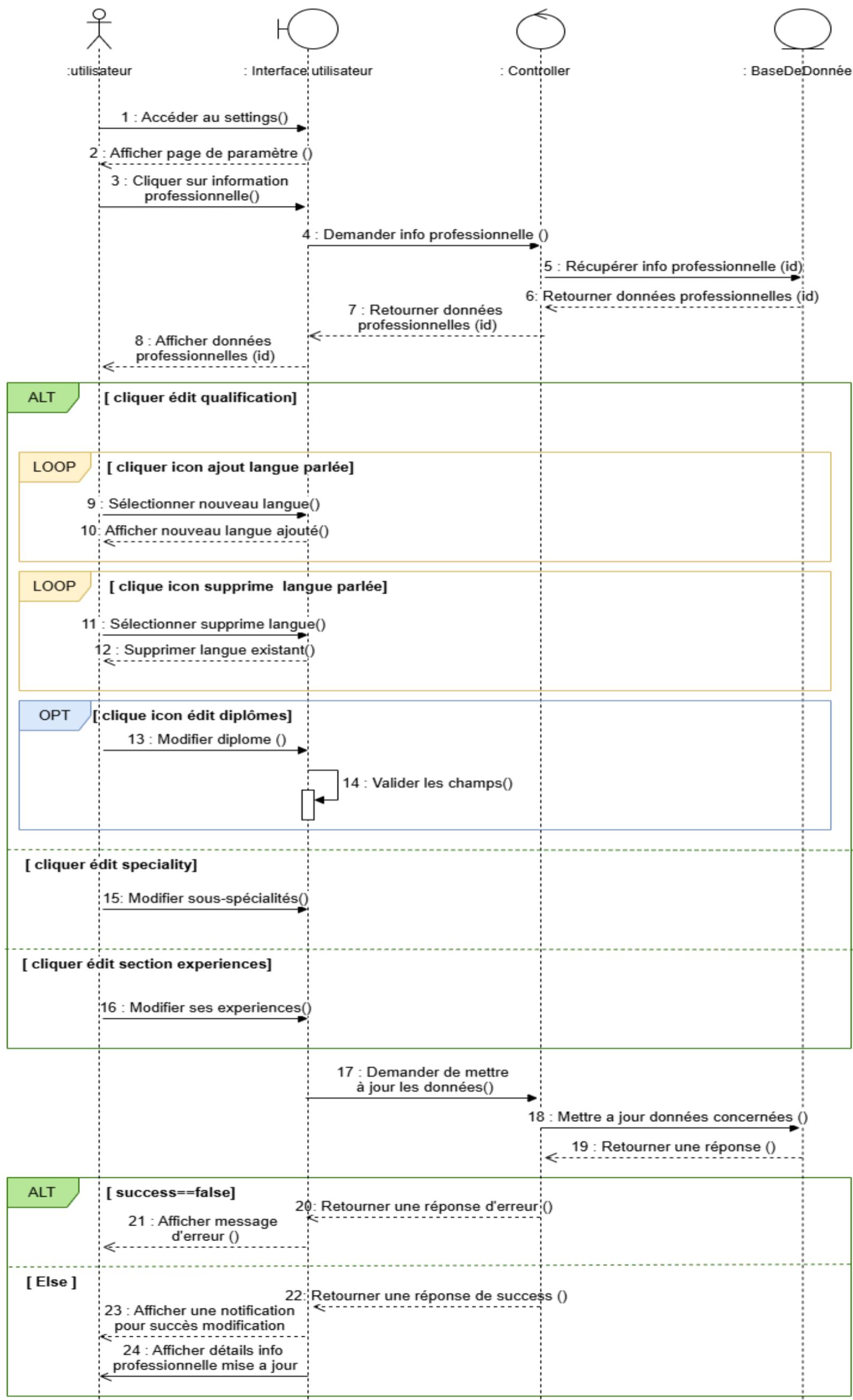


Figure 3.10 : Diagramme de s  quence de modifier information professionnelle

- Diagramme de séquence de « Inscription docteur »

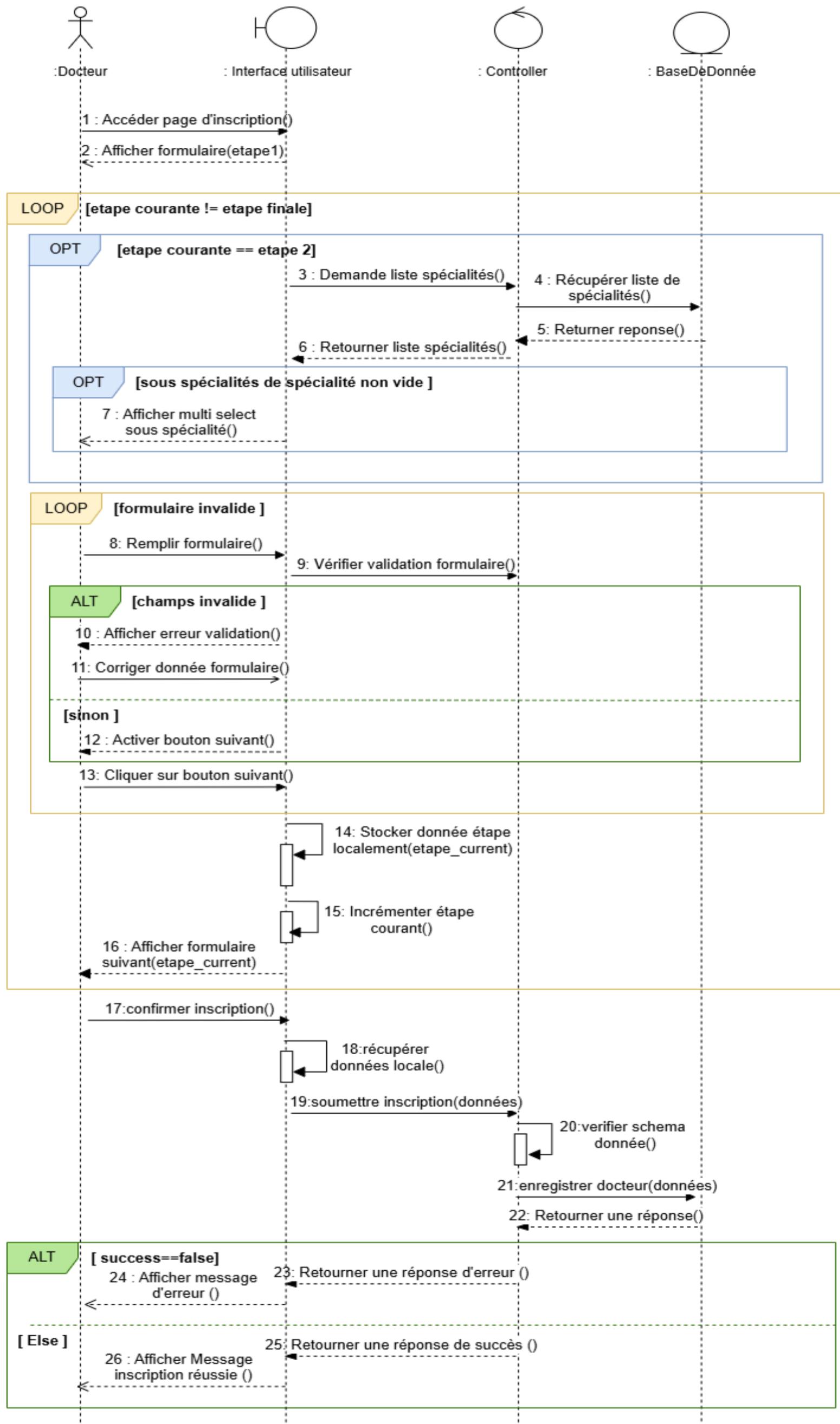


Figure 3.11 : Diagramme de séquence d'inscription docteur

3.1.4 Réalisation

Afin d'illustrer la mise en œuvre du premier sprint, nous décrirons dans cette partie les interfaces graphiques des principales fonctionnalités réalisées

3.1.4.1 Développement de Coté Web

- Interface de la liste de docteurs

Doctor	Specialty	Location	Rating	Experience	Status	Actions
Austin Kuphal	ORL	Tunis, Tunisia	0.0 ★	19 years	Active	...
Dr. Youssef	Generalist	Nabeul, Tunisia	3.0 ★	12 years	Active	...
Elias Ortiz	Nutrition	Roswell, Bonaire	0.0 ★	35 years	Active	...
Karen Watsica	Rheumatology	North Elsa, Eswatini	0.0 ★	3 years	Rejected	...
Kathryn Batz III	Rheumatology	Menzel Bouzefla, Tunisia	0.0 ★	31 years	Featured ★	...
Malcolm Bechtelar	ORL	Reneeview, Sierra Leone	0.0 ★	16 years	Active	...
Miss Joy Stamm	Radiology	UC, Brunel	0.0 ★	38 years	Active	...
Mr.Ernest Barton	ORL	Paris, France	0.0 ★	37 years	Active	...
Ms.FranciscA	Ophthalmology	North Stephania, Latvia	0.0 ★	23 years	Rejected	...

Figure 3.12 : Interface Web de liste des docteurs

Dès que le super-admin authentifié accède à l'interface de la liste des docteurs, il peut consulter tous les docteurs existants dans la base de données. Il peut rechercher un docteur par nom, filtrer la liste par spécialité ou par localisation. Il peut également modifier le statut d'un profil existant ou le supprimer. En cliquant sur l'un des onglets de statut, il peut afficher uniquement les docteurs correspondant au statut sélectionné

- Interface de la liste d'affiliations

Type	Location	Doctors	Services	Phone	Actions
Hôpital	city 1, Tunisia	0	0	11223344	
Clinic	city, Egypt	0	0	11223344	
Clinic	Paris, France	0	0	+33123456789	
Clinic	South Kole, Curacao	1	1	(968) 538-1022 x476	
Clinic	Madison, Nigeria	3	2	874.512.4487 x636	
Clinic	Paris, France	0	0		
Clinic	Jaydenfurt, Republic	2	1	(595) 317-8172	
Clinic	Reneevue, Sierra Leone	1	1	695.452.3857 x81504	
Clinic	Shieldsmouth, Montenegro	4	3	1-468-437-0413 x137	
Hôpital	Nabeul, Tunisia	0	0	(345) 529-8475 x984	

Figure 3.13 : interface Web de liste des établissements

Dès que le super-admin authentifié accède à l'interface de la liste des affiliations, il peut consulter toutes les affiliations existantes dans la base de données. Il peut rechercher une affiliation par nom, filtrer la liste par type ou par localisation. Il peut également ajouter une nouvelle affiliation

- Interface d'ajout une affiliation

The screenshot shows the 'Add New Hospital' modal open over a list of existing hospitals. The modal includes fields for 'Search Hospital' and 'OSM Location Details' which displays a map of 'Clinique Les Violettes' in Tunisie.

Figure 3.14 : Interface d'ajout d'une affiliation

Le super admin remplit le formulaire d'ajout d'une affiliation après vérification des champs non vides celui-ci est ajouté avec succès à la base de données.

3.1.4.2 Développement de Coté Mobile

- Interface des détails d'un docteur

La figure suivante représente l'interface des détails d'un docteur

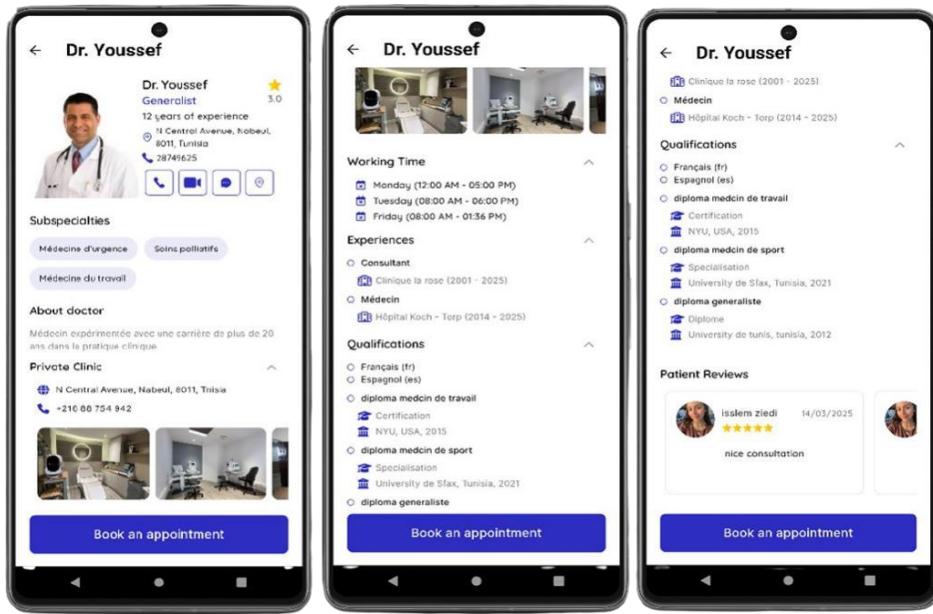


Figure 3.15 : Interface de détails d'un docteur

Dès qu'un patient authentifié sélectionne une carte d'un docteur, l'interface de détail docteur s'affiche. Il y retrouve les informations personnelles, l'expérience professionnelle, les qualifications. Il peut passer un rendez-vous avec ce docteur.

- Interface des détails d'une affiliation

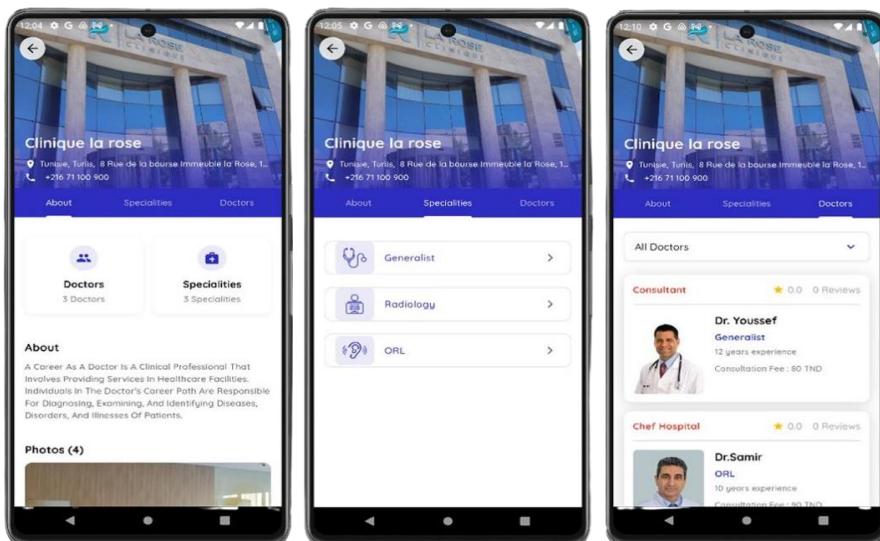


Figure 3.16 : Interface détails d'une affiliation

Dès qu'un patient authentifié sélectionne la carte d'une affiliation, l'interface de détail d'affiliation s'affiche. Il y retrouve les informations générales, la galerie d'images, les spécialités médicales disponibles ainsi que la liste des médecins exerçant au sein de cette affiliation.

3.1.4.1 SCRUM Board

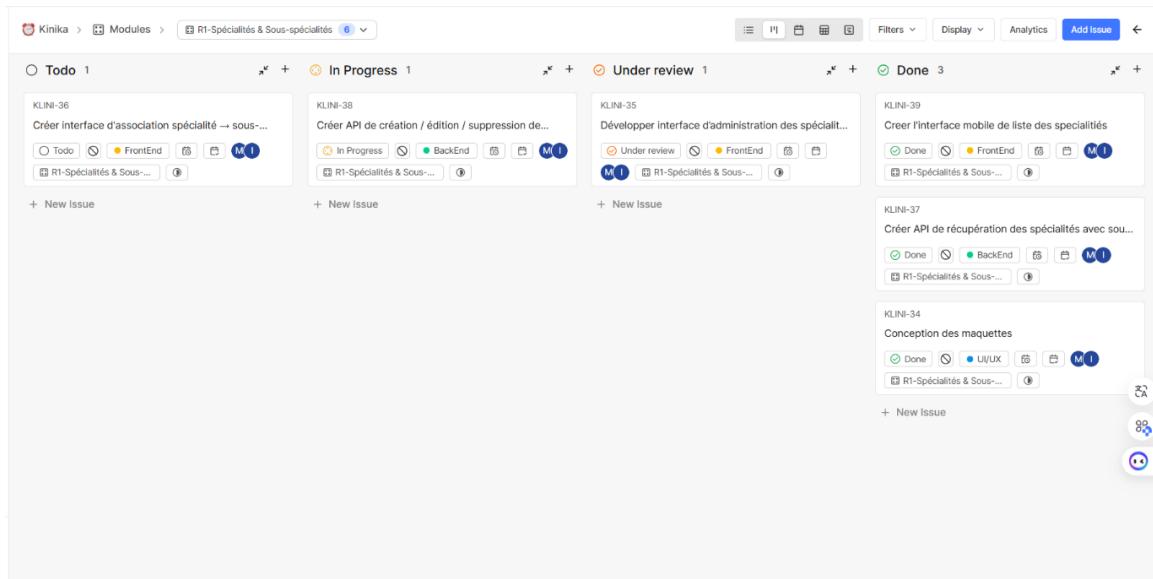


Figure 3.17 : Scrum Board de gestion spécialité sprint 1

Scrum Board intégré à l'application de planification. On y trouve une colonne « À faire » listant les tâches identifiées à réaliser, une colonne « En cours » regroupant les travaux en cours de développement, une colonne « En revue » pour les fonctionnalités soumises à validation, et enfin une colonne « Terminées » pour les tâches achevées et prêtes à être livrées. Cette présentation simple garantit la transparence des progrès et facilite la collaboration au sein de l'équipe.

3.2 Développement du Sprint 2 « Services patients et rendez-vous »

3.2.1 Backlog du Sprint 2

Tableau 3.4 : Backlog du sprint 2

Fonctionnalité	ID	User Story	Tâches	Story Point
Gestion des patients	17	En tant que patient , je peux créer une fiche d'un patient pour chacun de mes proches	- Implémenter le formulaire d'ajout d'un patient - Intégrer l'API d'enregistrement d'un patient	Moyenne
	18	En tant que patient , je peux créer le dossier médical (moi-même ou un proche), en y ajoutant des informations comme les antécédents médicaux (allergies, paramètres corporels)	- Implémenter les formulaires des informations médicales du patient - Implémenter et intégrer l'API liée aux données médicales	Difficile
	19	En tant que docteur/patient , je peux afficher la liste de mes patients / de mes relations	- Implémenter la page de liste de patient - Implémenter l'API de récupérer liste patients/proches relatif selon type utilisateur (docteur ou patient)	Facile
	20	En tant que docteur/patient , je peux consulter les détails d'un patient/ l'un de mes relations, afin d'accéder à ses	- Implémenter la page de détaille d'un patient - Implémenter la page d'historique de consultations	Difficile

		informations médicales, informations personnelles, rapport médical, etc.	<ul style="list-style-type: none"> - Implémenter la page de rapport médical - Implémenter les API de récupérer détails d'un patient ainsi celle de rapport médical. 	
Gestion des ordonnances	21	En tant que docteur , je peux créer une ordonnance de tests ou de médicaments et l'affecter à l'un de mes patients	<ul style="list-style-type: none"> - Implémenter les formulaires spécifiques pour chaque type d'ordonnance - Implémenter et intégrer L'API d'ajout d'ordonnance 	Moyenne
	22	En tant que docteur , je peux consulter la liste des ordonnances émises et les filtrer par type et statut	<ul style="list-style-type: none"> - Implémenter la logique et intégrer les API pour - Récupérer les ordonnances en fonction du type utilisateur (docteur ou patient) 	Facile
	23	En tant que docteur/ patient , je peux consulter les détails d'une ordonnance	<ul style="list-style-type: none"> - Implémenter la page de détails d'une ordonnance - Ajouter les dépendances nécessaires pour la génération de QR codes et l'export au format PDF - Implémenter et intégrer l'API détails d'une ordonnance 	Difficile
Gestion des documents externes	24	En tant que patient , je peux gérer les documents de moi-même ou l'un de	<ul style="list-style-type: none"> - Implémenter page des listes d'ordonnance avec les actions - Implémenter les API de 	Moyenne

médicaux		mes relations	CRUD des documents	
	25	En tant que patient , je peut partager un fichier avec un docteur consulté qui va recevoir une notification	<ul style="list-style-type: none"> - Implémenter page de mettre en collection des fichiers - Implémenter l'API de récupérer docteur consulté par patient - Intégrer la fonction d'envoie automatique de notification dans l'API. - Implémenter la page de partager un document 	Moyenne
Gestion emploie de temps docteur	26	En tant que docteur , je peux gérer mes heures de travail par semaine	<ul style="list-style-type: none"> -Implémenter la page de gestion d'emploie de travail d'un docteur -Implémenter les API CRUD des créneaux horaire par jour. 	Moyenne
Gestion disponibilités de docteur	27	En tant que docteur , je peux modifier mes disponibilités	<ul style="list-style-type: none"> -Implémenter formulaire d'ajout évènement -Implémenter l'API d'ajouter un évènement. 	Moyenne
Gestion des rendez-vous	28	En tant que docteur/patient , je peux créer/prendre un rendez-vous en ligne ou hors ligne	<ul style="list-style-type: none"> - Implémenter page de prendre rendez-vous - Implémenter un formulaire permettant d'associer le rendez-vous à un proche patient existant ou d'en enregistrer un nouveau. - Implémenter la page affichant les détails du 	Moyenne

			rendez-vous avant validation	
	29	En tant que docteur , je peux accepter, reprogrammer et annuler des rendez-vous	<ul style="list-style-type: none"> - Implémenter la page des rendez vous - Développer et intégrer les APIs correspondant à la logique de cette partie 	Difficile
	30	En tant que patient, je peux recevoir une notification indiquant l'état de mes rendez-vous	<ul style="list-style-type: none"> - Intégrer la fonction d'envoie automatique de notification dans l'API de modification de statut rendez-vous 	
	31	En tant que docteur/patient , je peux consulter la liste de mes rendez-vous, ainsi que ses détails	<ul style="list-style-type: none"> - Implémenter la page qui contient composant calender -Implémenter l'API de récupérer rendez -vous -Implémenter Page détails d'un rendez-vous 	Difficile
Rappel d'un rendez-vous	32	En tant que patient , je peux recevoir une notification d'un rappel de mes rendez-vous les plus proches	<ul style="list-style-type: none"> -Implémenter l'API de programmer un rappel de rendez-vous les plus proches 	Moyenne

3.2.2 Analyse

Pour la spécification fonctionnelle du sprint 2, nous présentons les diagrammes de cas d'utilisation global et détaillés

3.2.2.1 Diagramme de cas d'utilisation globale de sprint 2

Ce diagramme de cas d'utilisation présente les différentes fonctionnalités que le docteur et le patient peuvent utiliser durant le sprint 2

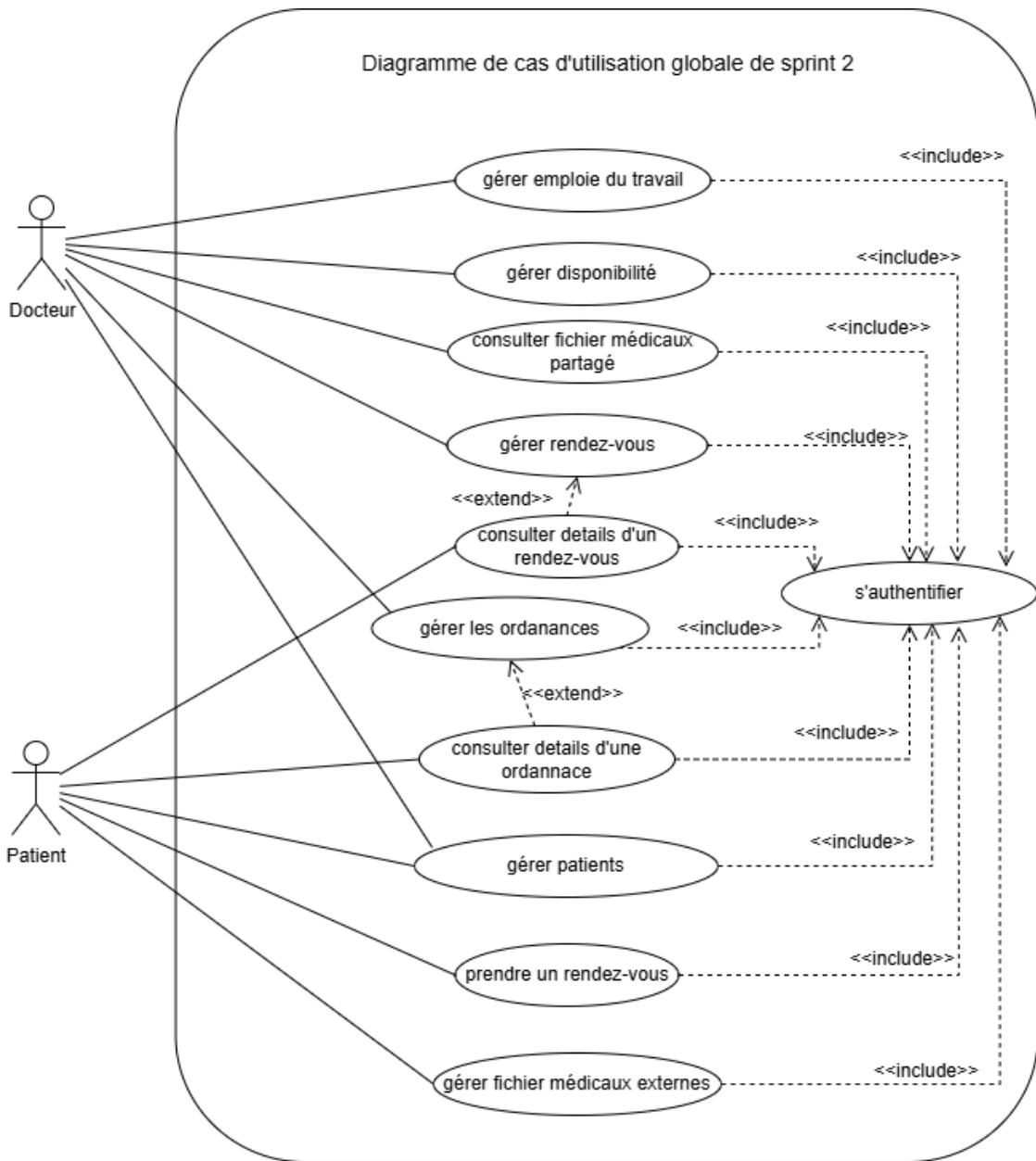


Figure 3.18 : Diagramme de cas d'utilisation globale de sprint 2

3.2.2.2 Diagramme de cas d'utilisation détaillés

Dans la suite, nous détaillons les diagrammes de cas d'utilisation pour les principales fonctionnalités.

- **Diagramme de cas d'utilisation détaillés « Gestion d'emploi du travail »**

La figure ci-dessous présente la réalisation du cas d'utilisation gestion d'emploi du travail

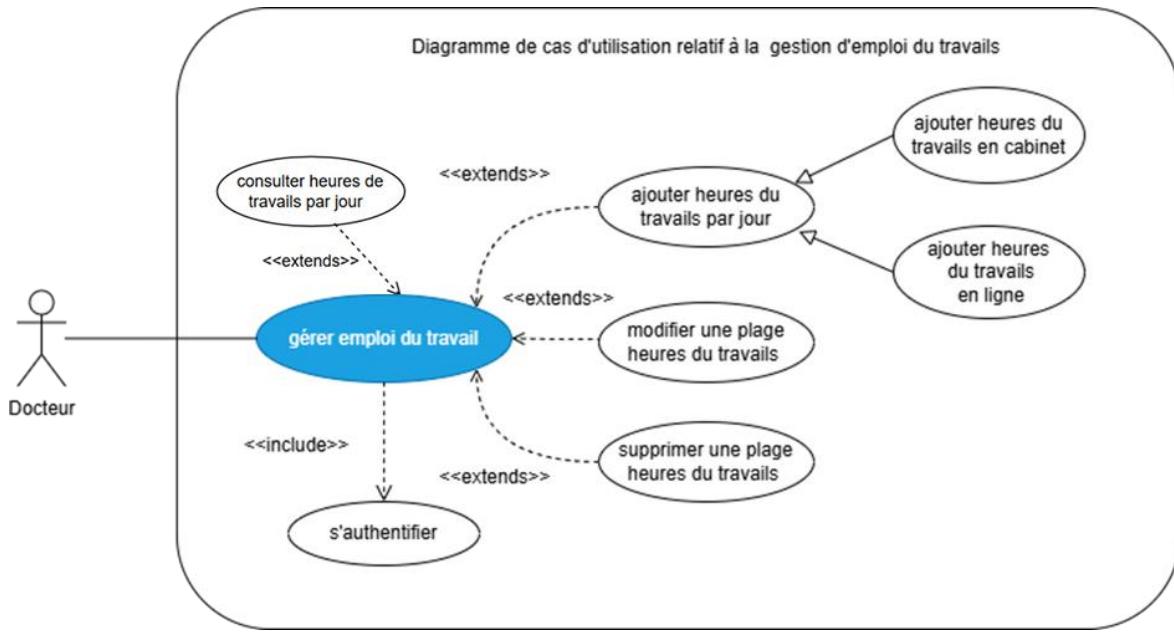


Figure 3.19 : Diagramme de cas d'utilisation détaillée de gestion emploi du temps

- **Diagramme de cas d'utilisation détaillés « Gestion de disponibilité »**

La figure ci-dessous présente la réalisation du cas d'utilisation gestion de disponibilité



Figure 3.20 : Diagramme de cas d'utilisation détaillée de gestion disponibilité

- **Diagrammes de cas d'utilisation détaillés « Gestion des ordonnances »**

La figure ci-dessous présente la réalisation du cas d'utilisation gestion des ordonnances

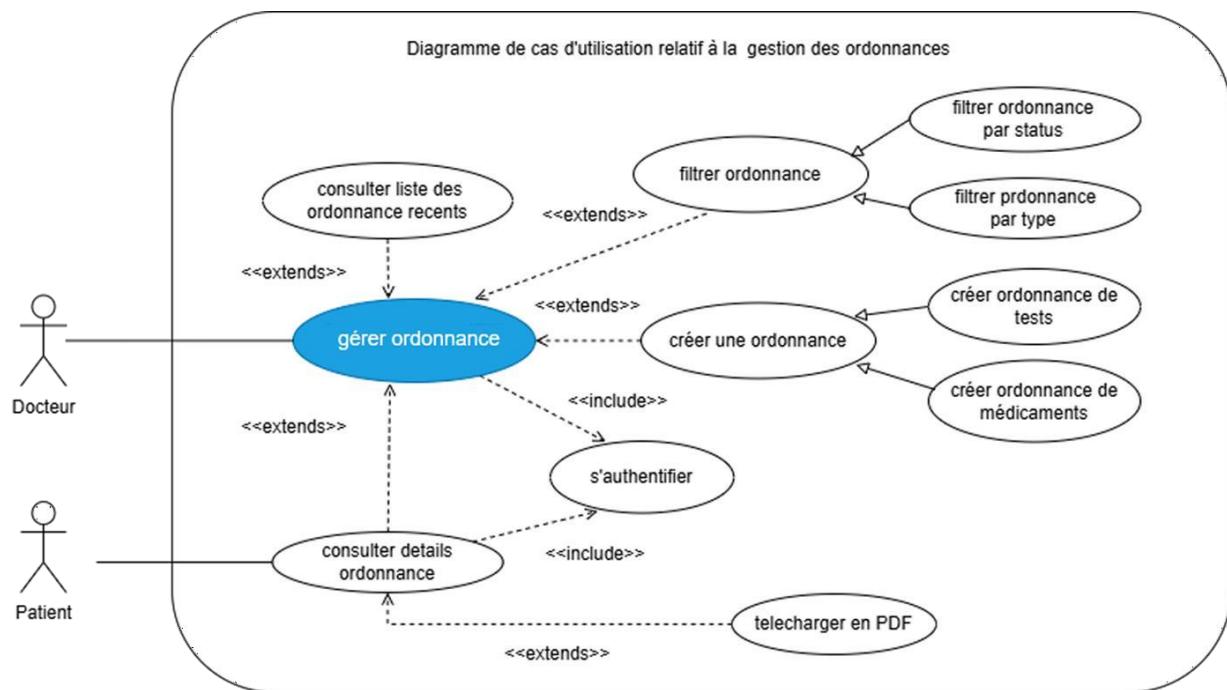


Figure 3.21 : Diagramme de cas d'utilisation détaillé de gestion des ordonnances

- **Diagrammes de cas d'utilisation détaillés « Gestion des patients »**

La figure ci-dessous présente la réalisation du cas d'utilisation gestion des patients

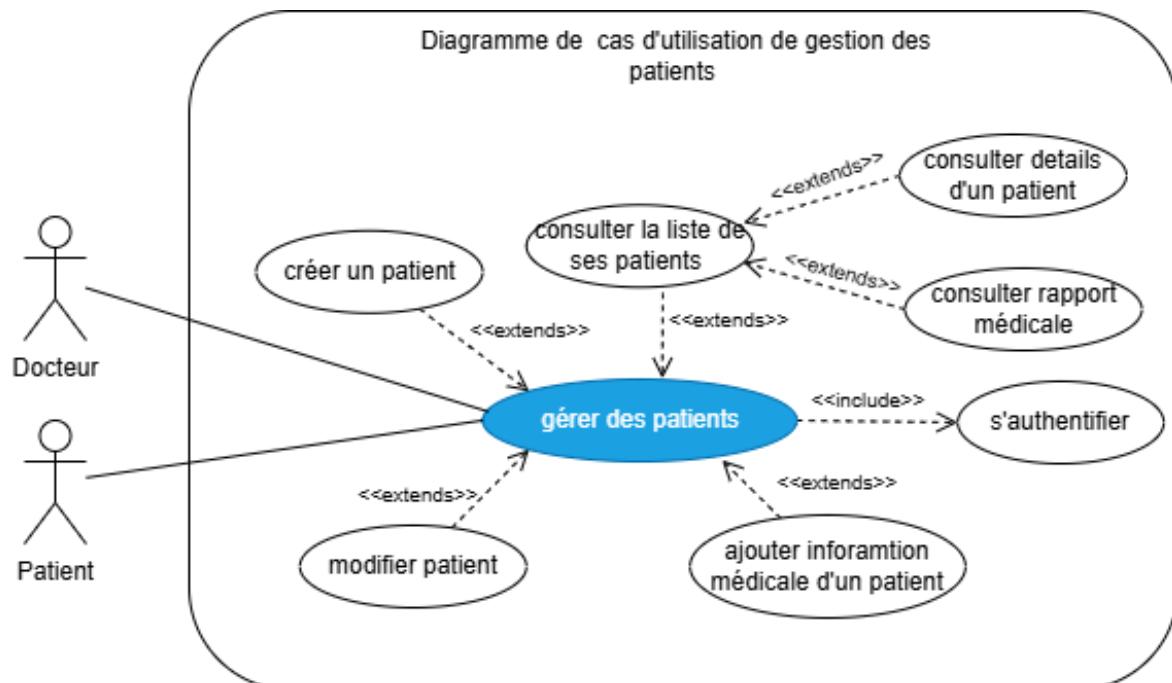


Figure 3.22 : Diagramme de cas d'utilisation détaillé de gestion patients

- **Diagrammes de cas d'utilisation détaillés « Gestion des fichiers médicaux externes »**

La figure ci-dessous présente la réalisation du cas d'utilisation gestion des fichiers médicaux externes

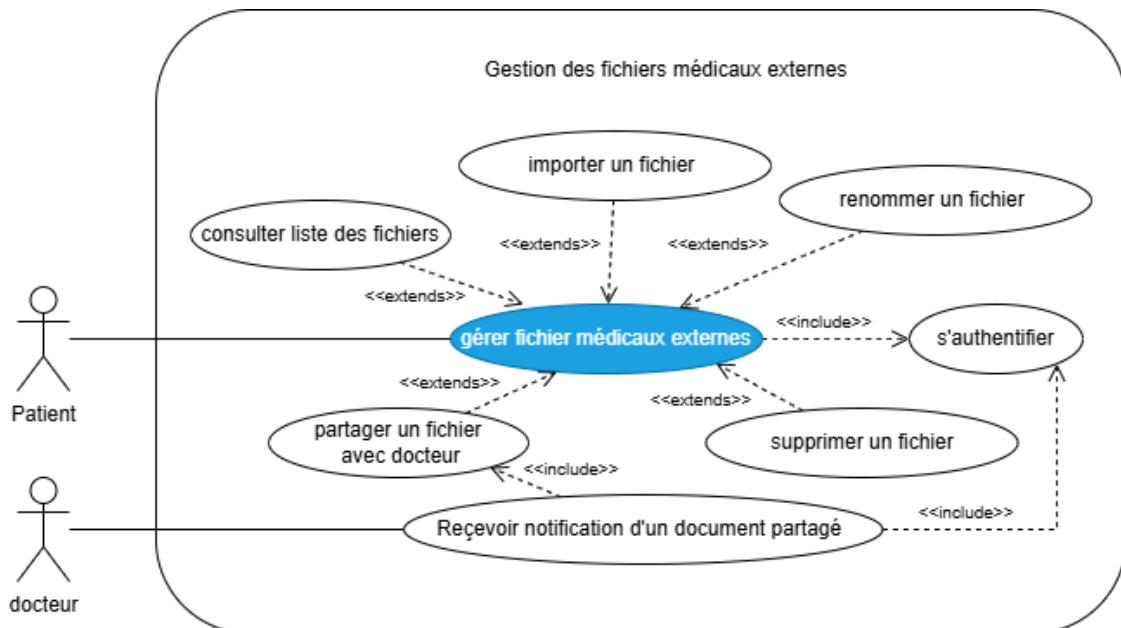


Figure 3.23 : Diagramme de cas d'utilisation détaillée de gestion fichiers externes

- **Diagrammes de cas d'utilisation détaillés « Gestion des rendez-vous »**

La figure ci-dessous présente la réalisation du cas d'utilisation gestion des rendez-vous

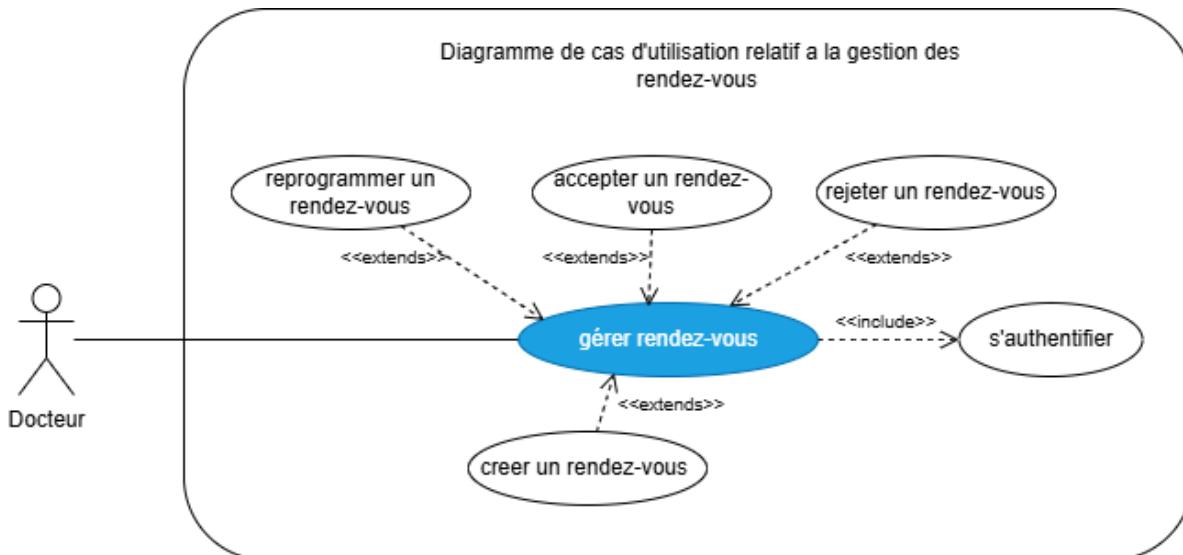


Figure 3.24 : Diagramme de cas d'utilisation détaillée de gestion des rendez vous

3.2.2.3 Description textuelle des cas d'utilisation

- **Description textuelle de cas d'utilisation « Passer un rendez-vous »**

Tableau 3.5 : *Description textuelle de cas d'utilisation « Passer un rendez-vous »*

Cas d'utilisation : Passer un rendez-vous
Acteur(s) : Patient, Docteur
Objectif : Permettre aux patients/docteur de prendre des rendez-vous
Pré-condition(s) : <ul style="list-style-type: none">• Le docteur doit être authentifié• Le patient doit être authentifié
Post-condition(s) : <ul style="list-style-type: none">• Les rendez-vous sont enregistrés dans la base de données• Les rendez-vous apparaissent dans le calendrier du docteur ainsi celle de patient
Scénario principal : <ol style="list-style-type: none">1. Le patient accède à la page de prise de rendez-vous2. Le patient sélectionne le type de rendez-vous (en ligne ou en présentiel)3. Le patient choisit une date4. Le système affiche, pour la date sélectionnée, les créneaux horaires disponibles, groupés par période de la journée5. Le patient choisit un créneau horaire disponible6. Le patient clique sur le bouton « Suivant », une boîte de dialog affiche la liste des patients existants (l'utilisateur lui-même et ses relatives) et un bouton « Nouveau patient »7. Le patient saisit les informations personnelles de patient concerné8. Le patient confirme la demande de rendez-vous9. Le rendez-vous est enregistré avec statut « pending »10. Le docteur est notifié de la nouvelle demande11. Le docteur consulte la demande de rendez-vous12. Le docteur accepte le rendez-vous13. Le système modifie le statut de ce rendez-vous en « confirmed »14. Le système envoie une confirmation au patient via email

Scénario alternatif :

4.1 Aucun créneau horaire disponible pour la date sélectionnée :

- Le système indique qu'aucun créneau n'est disponible pour la date sélectionnée

7.1 Le patient choisit un patient déjà enregistré :

- Le système pré-remplit les informations personnelles

Le patient vérifie et confirme les informations

• **Description textuelle de cas d'utilisation « Créer une ordonnance »**

Tableau 3.6 : Description textuelle de cas d'utilisation « Créer une ordonnance »

Cas d'utilisation : Crée une ordonnance
Acteur(s) : Docteur
Objectif : Permettre aux docteurs de créer des ordonnances de tests ou de médicaments
Pré-condition(s) : <ul style="list-style-type: none">• Le docteur doit être authentifié
Post-condition(s) : <ul style="list-style-type: none">• L'ordonnance est créée et enregistrée dans le système• L'ordonnance est accessible par le docteur et le patient concerné• L'ordonnance est téléchargeable au format PDF
Scénario principal : <ol style="list-style-type: none">1. Le docteur accède à l'interface de création d'ordonnances2. Le docteur choisit le type d'ordonnance (médicaments ou tests)3. Le docteur recherche un patient dans sa liste de patients4. Le docteur sélectionne un patient parmi la liste5. Le docteur sélectionne la date d'expiration de l'ordonnance6. Le docteur choisit les médicaments ou tests et saisit des notes additionnelles7. Le docteur valide et enregistre l'ordonnance8. Le système génère automatiquement l'ordonnance et l'associe au dossier médical du patient, en y intégrant un QR Code pour en garantir l'authenticité.
Scénario alternatif :
3.1 Liste de patients vide : <ul style="list-style-type: none">○ Le système affiche un message indiquant que la liste des patients est vide et le

scenario redémarre de l'étape numéro 4

6.1 Liste de médicaments ou tests vide :

- Le système affiche un message indiquant que la liste est vide

3.2.3 Conception

3.2.3.1 Diagramme de classe de sprint 2

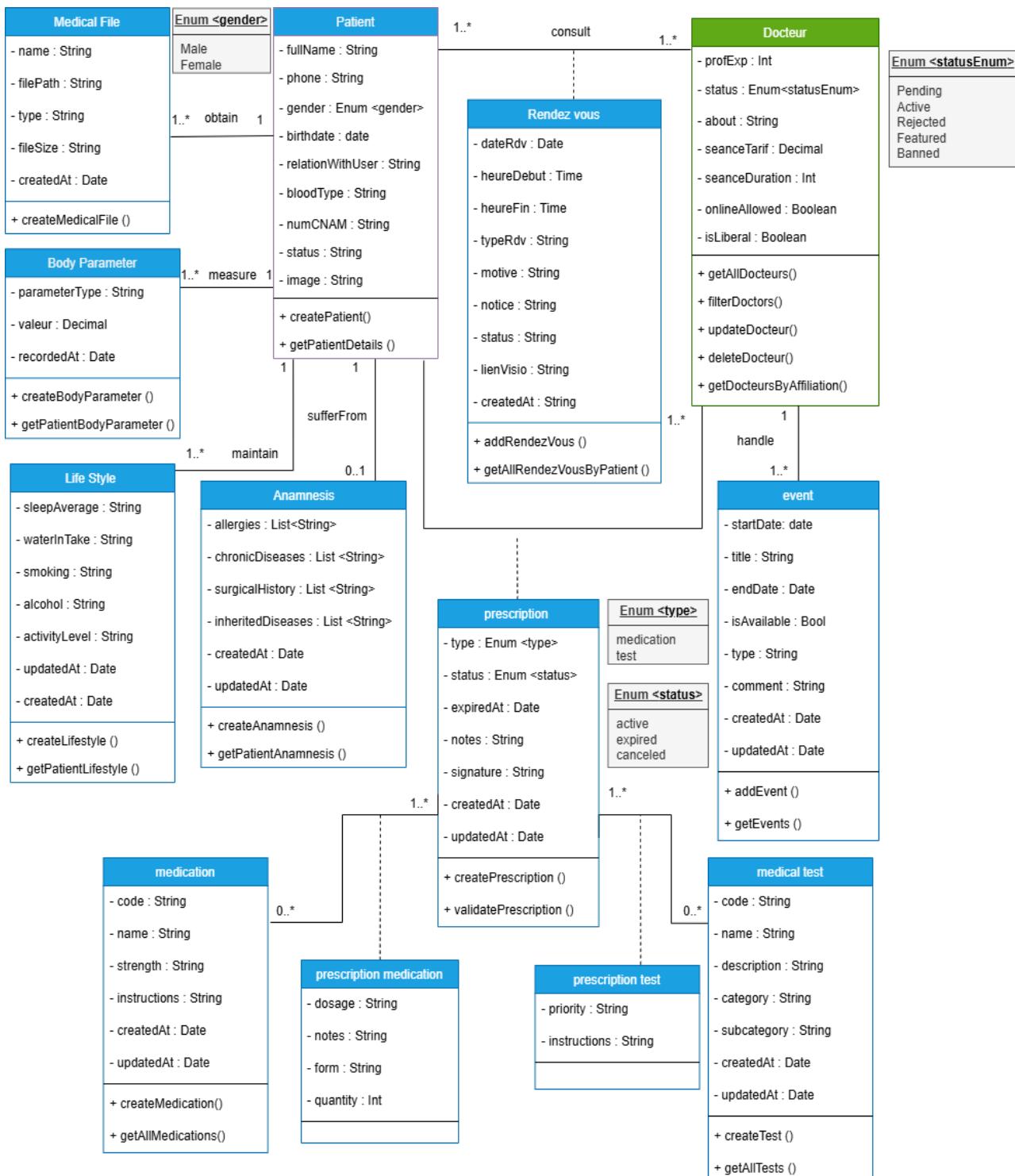


Figure 3.25 : Diagramme de classe de sprint 2

3.2.3.2 Diagrammes de séquences de sprint 2

- Diagramme de séquence de « ajouter un créneau d'horaire en emploi du temps »

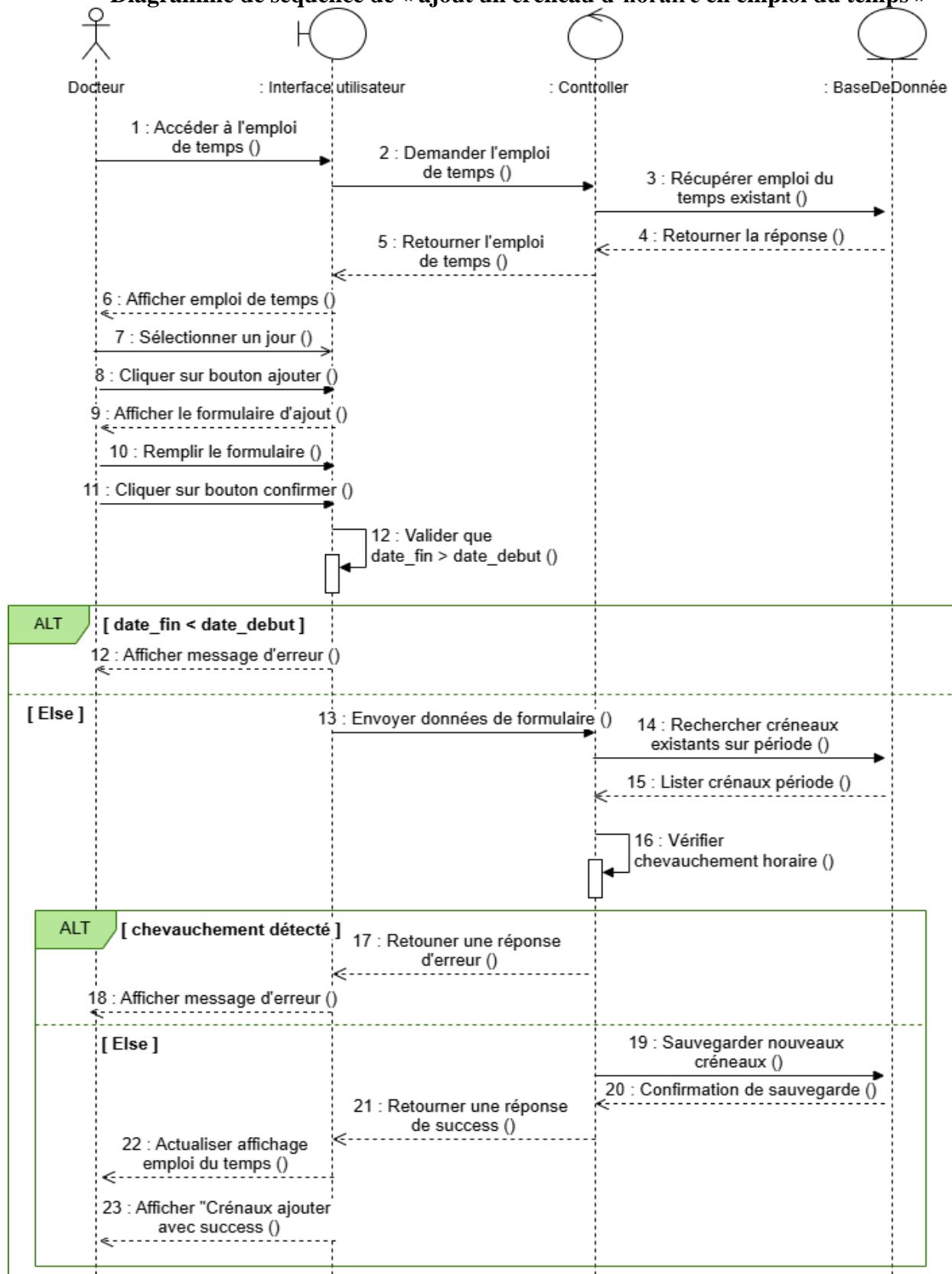


Figure 3.26 : Diagramme de séquencé d'ajouter créneau horaire

- Diagramme de séquence de « partager d'un document »

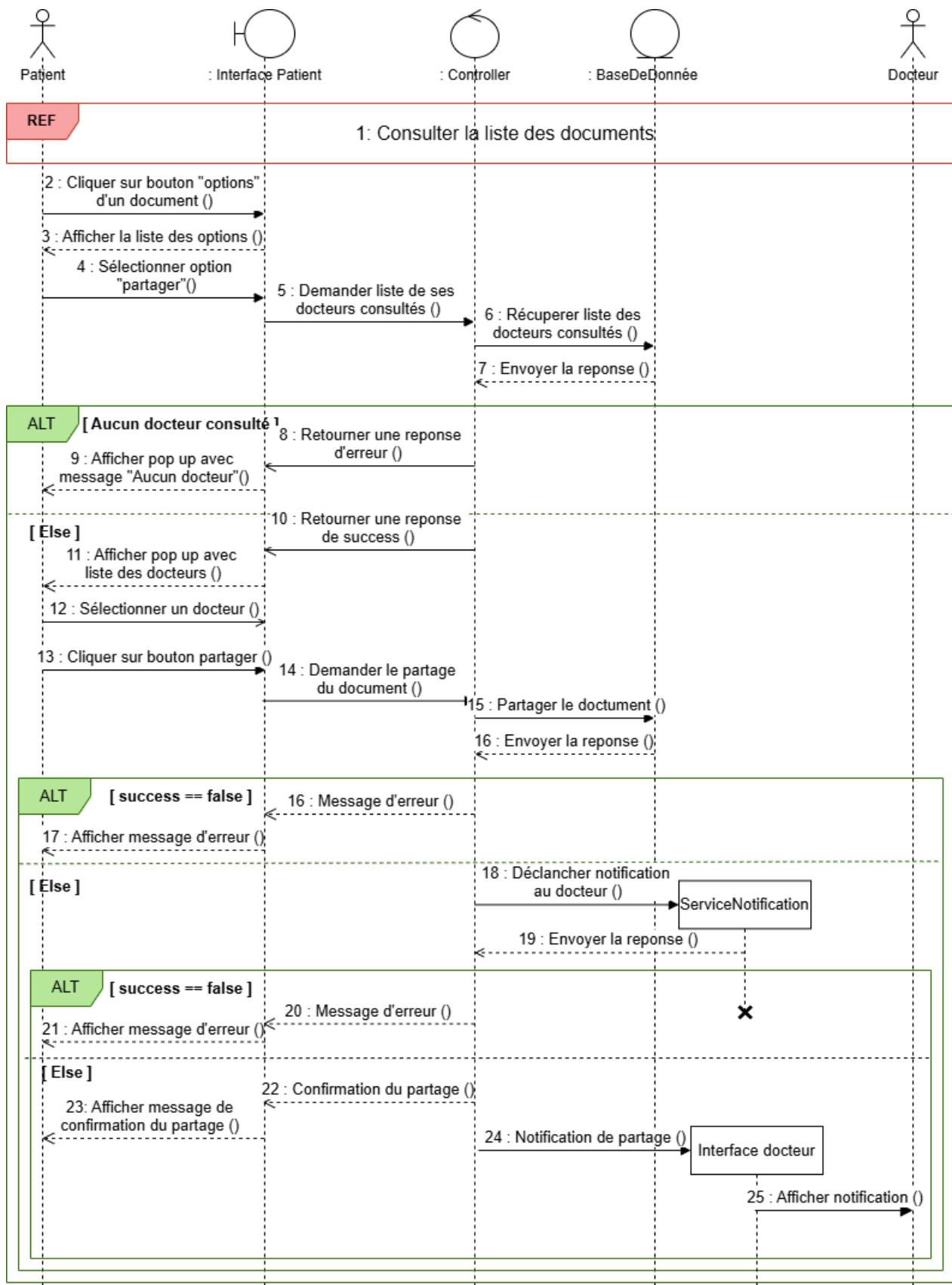


Figure 3.27 : Diagramme de séquencé de partager d'un document

- Diagramme de séquence de « passer un rendez-vous »

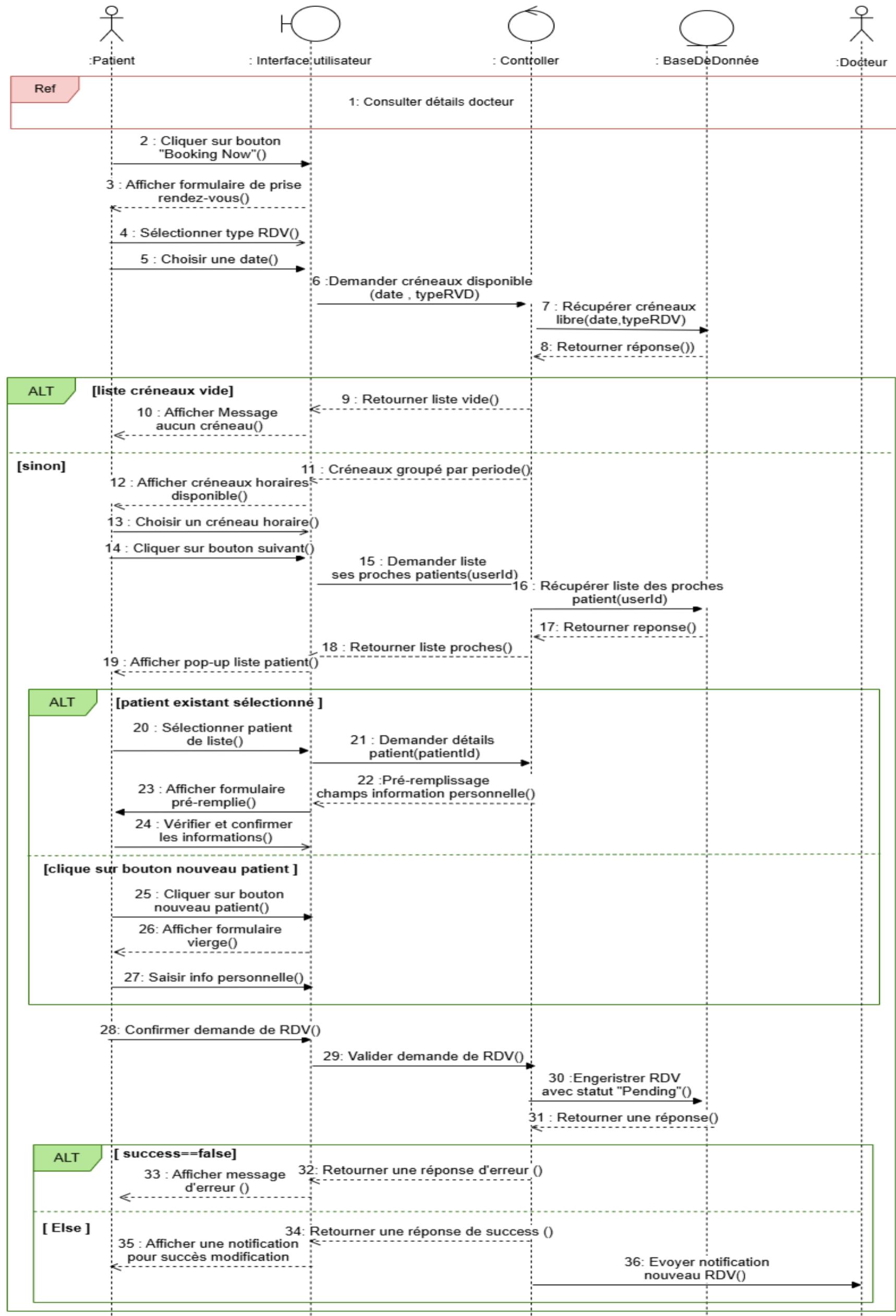


Figure 3.28 : Diagramme de séquencé de passer rendez vous

3.2.4 Réalisation

Afin d'illustrer la mise en œuvre du deuxième sprint, nous décrirons dans cette partie les interfaces graphiques des principales fonctionnalités réalisées

- **Interface d'emploi du temps d'un docteur**

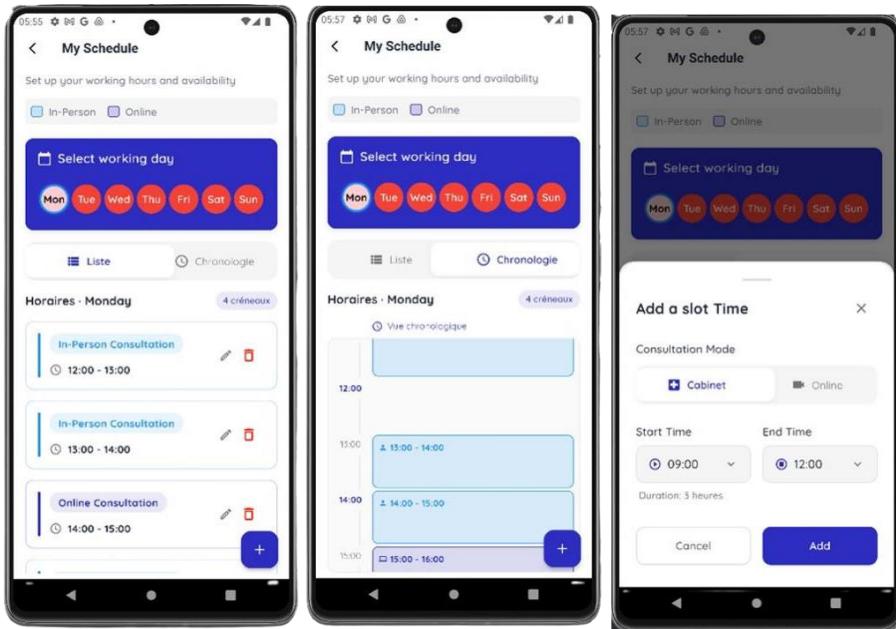


Figure 3.29 : Interface mobile d'emploi du temps de docteur

Dès que le docteur accède à l'interface « emploi du temps », il visualise les jours de la semaine. En sélectionnant un jour, la liste des créneaux de travail définis pour cette journée s'affiche. Il peut choisir d'afficher ces créneaux en vue liste ou en vue chronologique selon sa préférence. Il peut ajouter un nouveau créneau de travail en cliquant sur le bouton d'ajout, puis en remplissant le formulaire et en validant via le bouton « Ajouter ».

Il peut également modifier ou supprimer un créneau existant en cliquant sur les icônes correspondantes disponibles sur chaque carte de créneau horaire

- **Interface de calendrier d'un docteur**

La figure suivante représente l'interface de calendrier d'un docteur

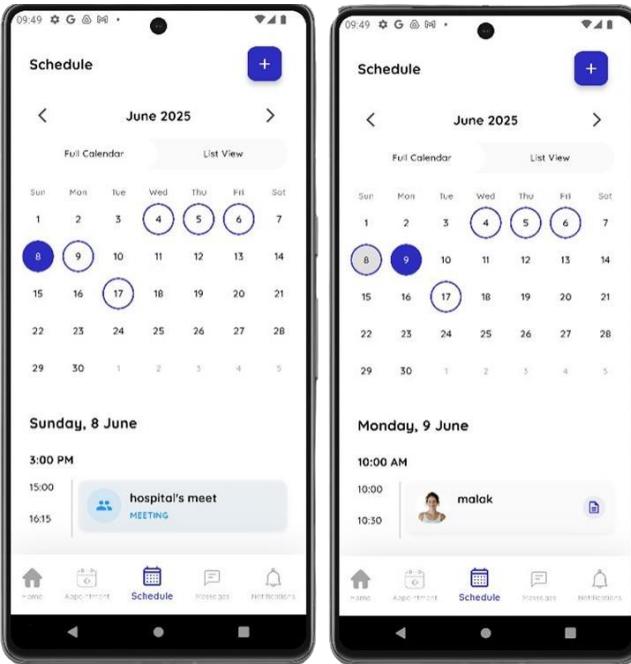


Figure 3.30 : Interface de calendrier de docteur

Dès que le docteur authentifié clique sur l'option « Calendrier » dans la barre de navigation inférieure, il accède à son interface de calendrier personnel. Il peut y consulter tous les événements qui lui sont associés, comme des rendez-vous, des réunions, etc.

Il peut sélectionner une date précise (année, mois, jour) afin d'afficher les événements planifiés pour cette journée, triés par heure.

Il peut également ajouter un nouvel événement en cliquant sur le bouton d'ajout.

- **Interface de la liste des rendez-vous**

La figure suivante représente l'interface la liste des rendez-vous

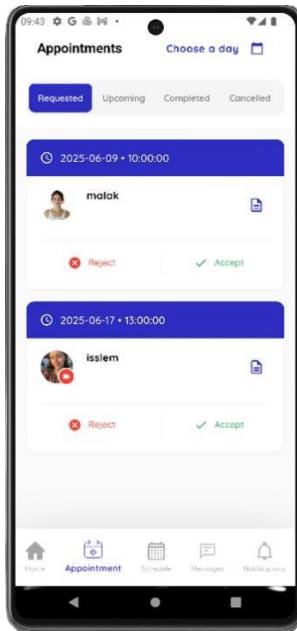


Figure 3.31 : Interface de la liste des rendez-vous

Dès que le docteur authentifié clique sur l'option « Rendez-vous » dans la barre de navigation inférieure, il accède à l'interface de gestion des rendez-vous.

Il peut y consulter ses rendez-vous classés par statut : « Requested », « Upcoming », « Completed » et « Cancelled », en cliquant sur les onglets correspondants.

Dans l'onglet « Requested », il peut accepter ou refuser les demandes. Il peut aussi filtrer les rendez-vous par date en cliquant sur « Choose a day »..

- **Interface des détails d'un patient**

La figure suivante représente l'interface des détails d'un patient

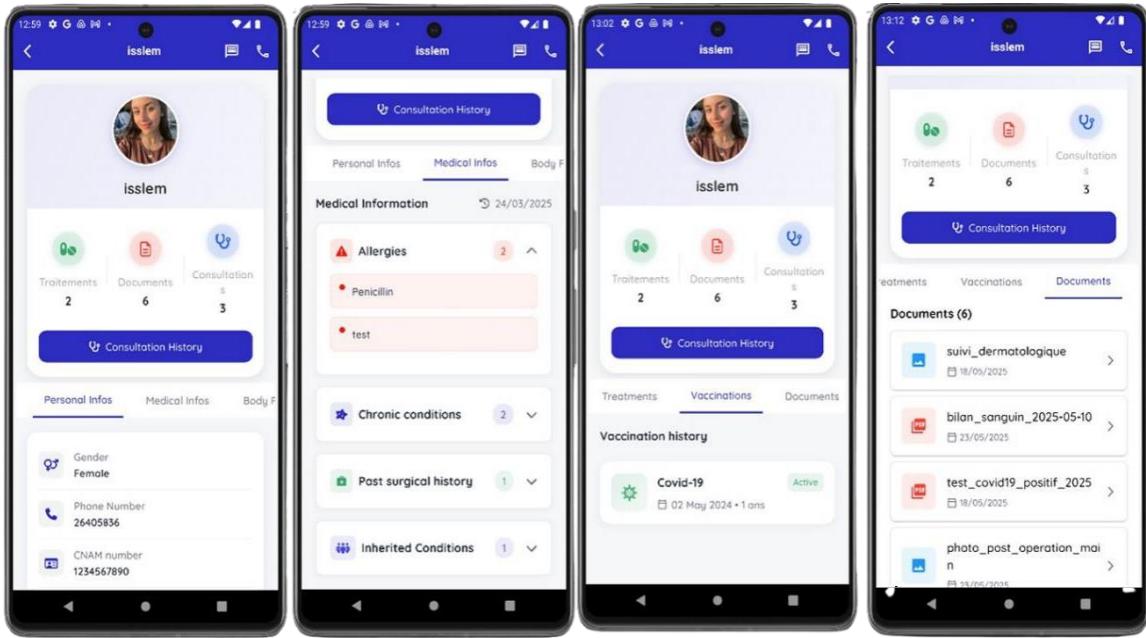


Figure 3.32 : Interface détails d'un patient

Dès que le docteur authentifié sélectionne une carte d'un patient, l'interface de détail patient s'affiche. Il y retrouve les informations personnelles et médicales du patient, ainsi que son historique de consultations en cliquant sur le bouton « Historique des consultations ».

4. CHAPITRE 4 :

Release 2 « Module de communication et AI »

4.1 Développement du Sprint 3 « Services de communication»

4.1.1 Backlog du Sprint 3

Tableau 4.1 : Backlog du sprint 3

Fonctionnalité	ID	User Story	Tâches	Story Point
Gestion des articles partagée « MedStream »	34	En tant que docteur, je peux lister mes articles par type (vidéo, audio, e-article), modifier et supprimer un article, ainsi de créer un nouveau	<ul style="list-style-type: none"> - Implémenter la page d'ajout d'un article avec formulaire adapté à chaque type d'article - Implémenter une page pour afficher la liste des articles - Mettre en place les pop-ups de modification avec formulaires selon le type d'article - Implémenter les fonctionnalités de CRUD d'un article 	Moyenne
	35	En tant que patient, je peux consulter les articles d'un docteur partagés lors de « waiting room » afin d'attendre un docteur sans s'ennuyer	<ul style="list-style-type: none"> - Implémenter la page « waiting room » avec la liste des articles partagés par le docteur - Implémenter l'API de cette logique de cette partie 	Facile
Messagerie sécurisée	36	En tant que patient ou docteur, je souhaite pouvoir communiquer afin d'échanger des informations et des conseils médicaux	<ul style="list-style-type: none"> - Implémenter en flutter les pages et les composants de messagerie (liste des conversations, fil de discussion, champ de saisie) - Mettre en place la logique d'envoi et de réception des messages via supabase realtime 	Difficile

CHAPITRE 4 : Release 2 « Module de communication et AI »

			- Tester l'échange de messages via l'application Flutter	
Démarrer une Vidéoconférence	37	En tant que docteur , je peux lancer une Vidéoconférence pour commencer une consultation en ligne	<ul style="list-style-type: none"> - Recherche et comparaison des différents packages d'implémentation de vidéoconférence - Ajouter le package “livekit client” 	Difficile
Rejoindre une Vidéoconférence	38	En tant que patient , je peux rejoindre une vidéoconférence pour avoir une consultation en ligne	<ul style="list-style-type: none"> - Implémenter la page de demande de participation - Personnaliser la page de vidéoconférence - Implémenter la logique de création et celle de jointure à la vidéoconférence 	Difficile
Contrôler Audio/Vidéo	39	En tant que participant à la vidéoconférence , je peux couper ou réactiver mon micro et activer ou désactiver ma caméra	<ul style="list-style-type: none"> Implémenter la logique de l'acceptation de demande de participation 	Facile
Assistant AI « Nasouh »	40	En tant que patient , je peux communiquer avec un chat bot AI afin de m'aider à identifier la propre spécialité médicale que je dois visiter basée sur mes symptômes	<ul style="list-style-type: none"> - Configurer l'API Gemini - Créer le système de prompt pour limiter les réponses au contexte médical et réglé de sécurité nécessaire - Développer la logique d'identification de spécialités basée sur les symptômes 	Difficile
	41	En tant que patient , je peux obtenir des conseils de bien-être sûrs pour adopter un mode de vie sain et équilibré	<ul style="list-style-type: none"> - Implémenter page chat avec l'assistant en flutter 	Difficile

4.1.1 Analyse

Pour la spécification fonctionnelle du sprint 3, nous présentons les diagrammes de cas d'utilisation global et détaillés

4.1.2.1 Diagramme de cas d'utilisation globale de sprint 3

Ce diagramme de cas d'utilisation présente les différentes fonctionnalités que le docteur et le patient peuvent utiliser durant le sprint 3

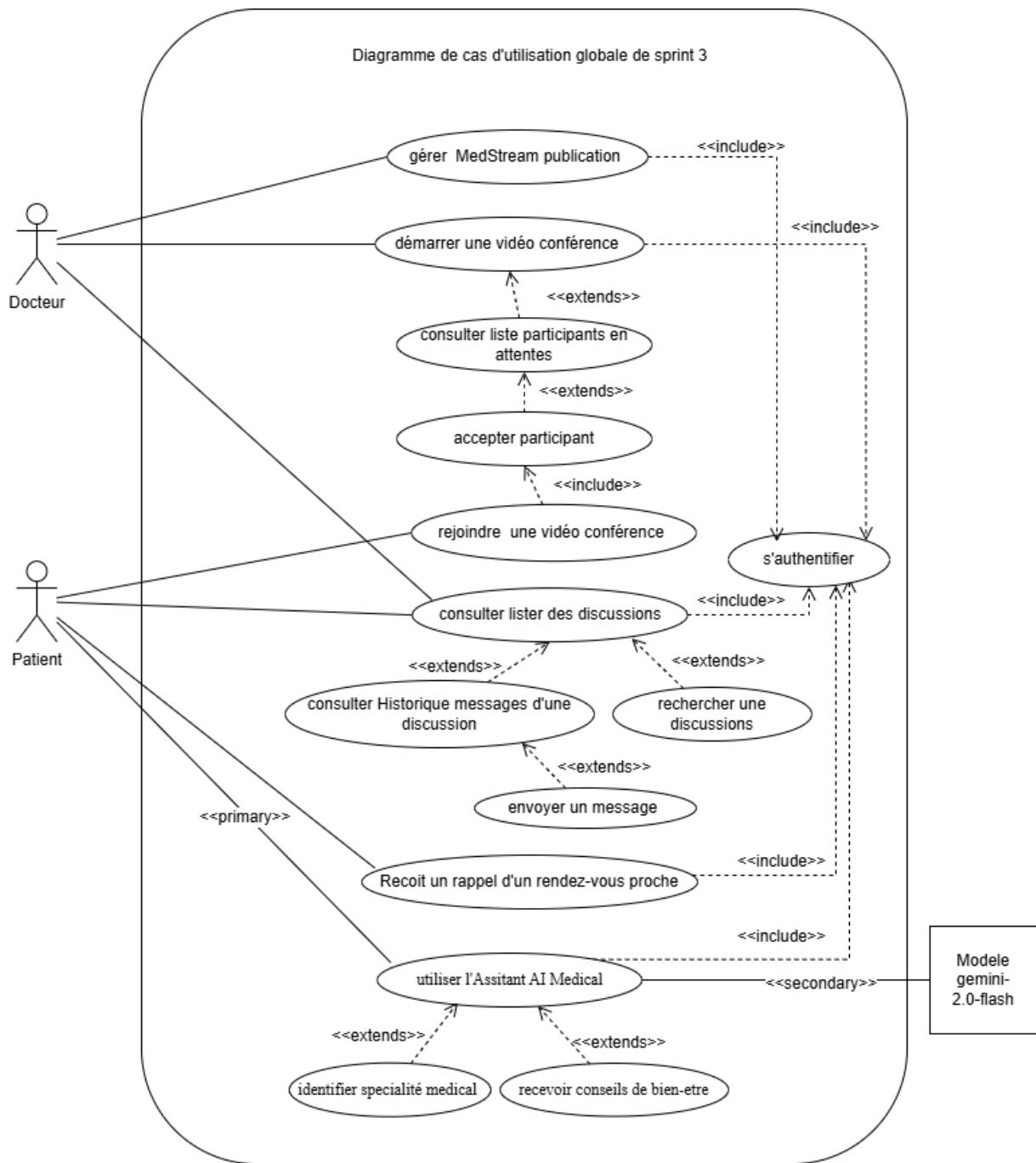


Figure 4.1 : Diagramme de cas d'utilisation de sprint 3

4.1.2.2 Description textuelle des cas d'utilisation

- **Description textuelle de cas d'utilisation « Envoyer un message »**

Tableau 4.2 : Description textuelle de cas d'utilisation « Envoyer un message »

Cas d'utilisation : Envoyer un message
Acteur(s) : Patient, Docteur
Objectif : Permettre aux patients et docteurs d'échanger des messages sécurisés pour communiquer des informations et des conseils médicaux
Pré-condition(s) : <ul style="list-style-type: none"> • Le docteur doit être authentifié • Le patient doit être authentifié • La connexion à Supabase Realtime doit être établie
Post-condition(s) : <ul style="list-style-type: none"> • Le message est envoyé et stocké de manière sécurisée dans le système • Le destinataire reçoit une notification de nouveau message • Le message apparaît dans le fil de discussion des deux utilisateurs
Scénario principal : <ol style="list-style-type: none"> 1. Un patient/docteur accède à la page de la liste des conversations 2. Un patient /docteur sélectionne une conversation existante ou en crée une nouvelle avec un destinataire autorisé (patient pour le docteur, docteur consulté pour le patient) 3. Le système affiche le fil de discussion avec l'historique des messages 4. Un patient /docteur saisit son message dans le champ de texte 5. Un patient/docteur appuie sur le bouton d'envoi 6. Le système enregistre le message dans la base de données Supabase 7. Le système notifie le destinataire via Supabase Realtime 8. Le message apparaît dans le fil de discussion des deux utilisateurs 9. Le système met à jour le champ is _read du message à "TRUE"
Scénario alternatif : <p>2.1 Crédation d'une nouvelle conversation : Liste de destinataires vide</p> <ul style="list-style-type: none"> ○ Le système affiche un message approprié ("Aucun patient enregistré" ou "Aucun docteur consulté")

- **Description textuelle de cas d'utilisation « Démarrer une vidéoconférence »**

Tableau 4.3 : Description textuelle de cas d'utilisation « Créer une vidéoconférence »

Cas d'utilisation : Démarrer une vidéoconférence
Acteur(s) : Docteur
Objectif :
Pré-condition(s) : <ul style="list-style-type: none"> • Le docteur doit être authentifié • L'appareil du docteur doit disposer d'une caméra et d'un microphone fonctionnels
Post-condition(s) : <ul style="list-style-type: none"> • Une salle de vidéoconférence est créée et active • Le docteur est connecté à la salle en tant qu'hôte • Les patients peuvent demander à rejoindre la salle
Scénario principal : <ol style="list-style-type: none"> 1. Le docteur accède à la liste des rendez-vous de téléconsultation d'aujourd'hui restantes 2. Le docteur sélectionne l'option « démarrer » de la première prochaine téléconsultation 3. Le système crée la salle via livekit et établit la connexion 4. Le docteur entre automatiquement dans la salle en tant qu'hôte 5. Le docteur est notifié (toast) lorsqu'un patient demande à rejoindre 6. Le docteur consulte la liste d'attente des patients 7. Le docteur accepte l'accès aux patients en attente 8. Les patients acceptés rejoignent la salle de vidéoconférence
Scénario alternatif : <p>3.1 Problème de connexion :</p> <ul style="list-style-type: none"> ○ Le système affiche message d'erreur <p>4.1 Problème de périphériques :</p> <ul style="list-style-type: none"> ○ Le système détecte que la caméra ou le microphone n'est pas disponible ○ Le système affiche un avertissement ○ Le docteur vérifie ses périphériques et autorisations ○ Le docteur entre dans la salle avec les périphériques disponibles

4.1.2 Conception

4.1.3.1 Diagramme de classe de sprint 3

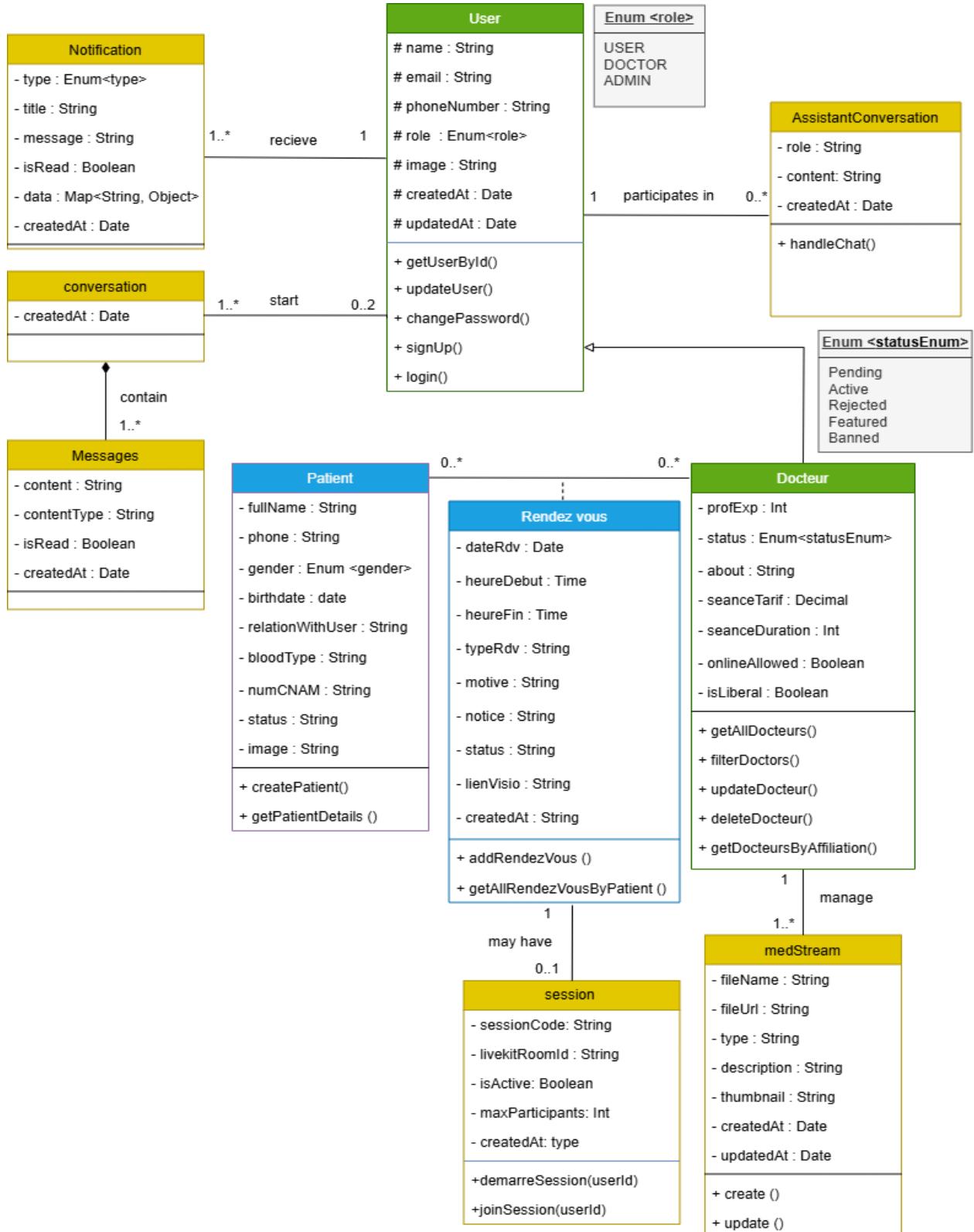


Figure 4.2 : Diagramme de classe de sprint 3

4.1.3.2 Diagrammes de séquences de sprint 3

- Diagramme de séquence de « envoyer message »

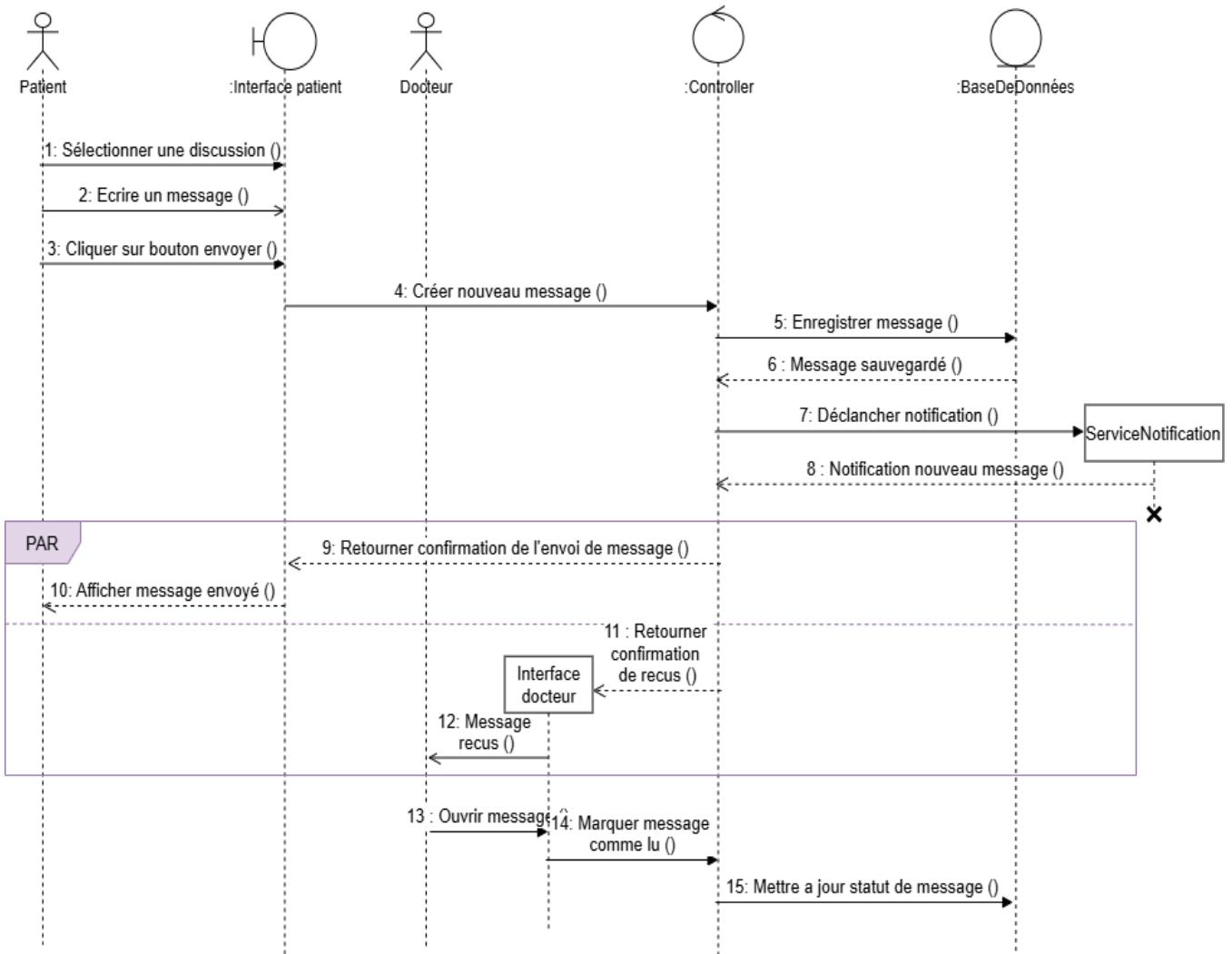


Figure 4.3 : Diagramme de séquencé de envoyer message

- Diagramme de séquence de « Rejoindre vidéoconférence »

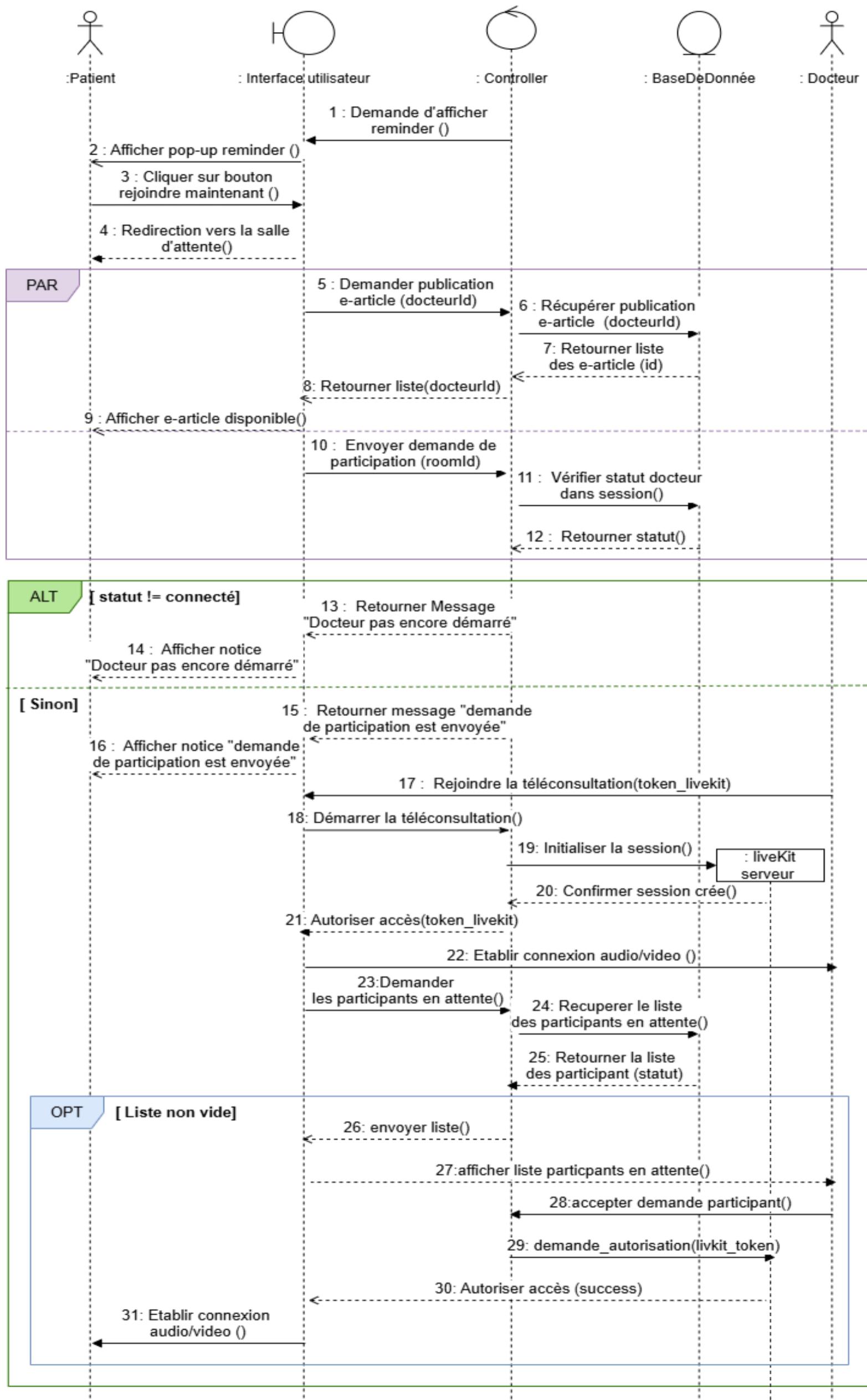


Figure 4.4 : Diagramme de sequence de rejoindre video conference

4.1.3.3 Diagramme d'activité de sprint 3

- Diagramme d'activité d'Assistant IA

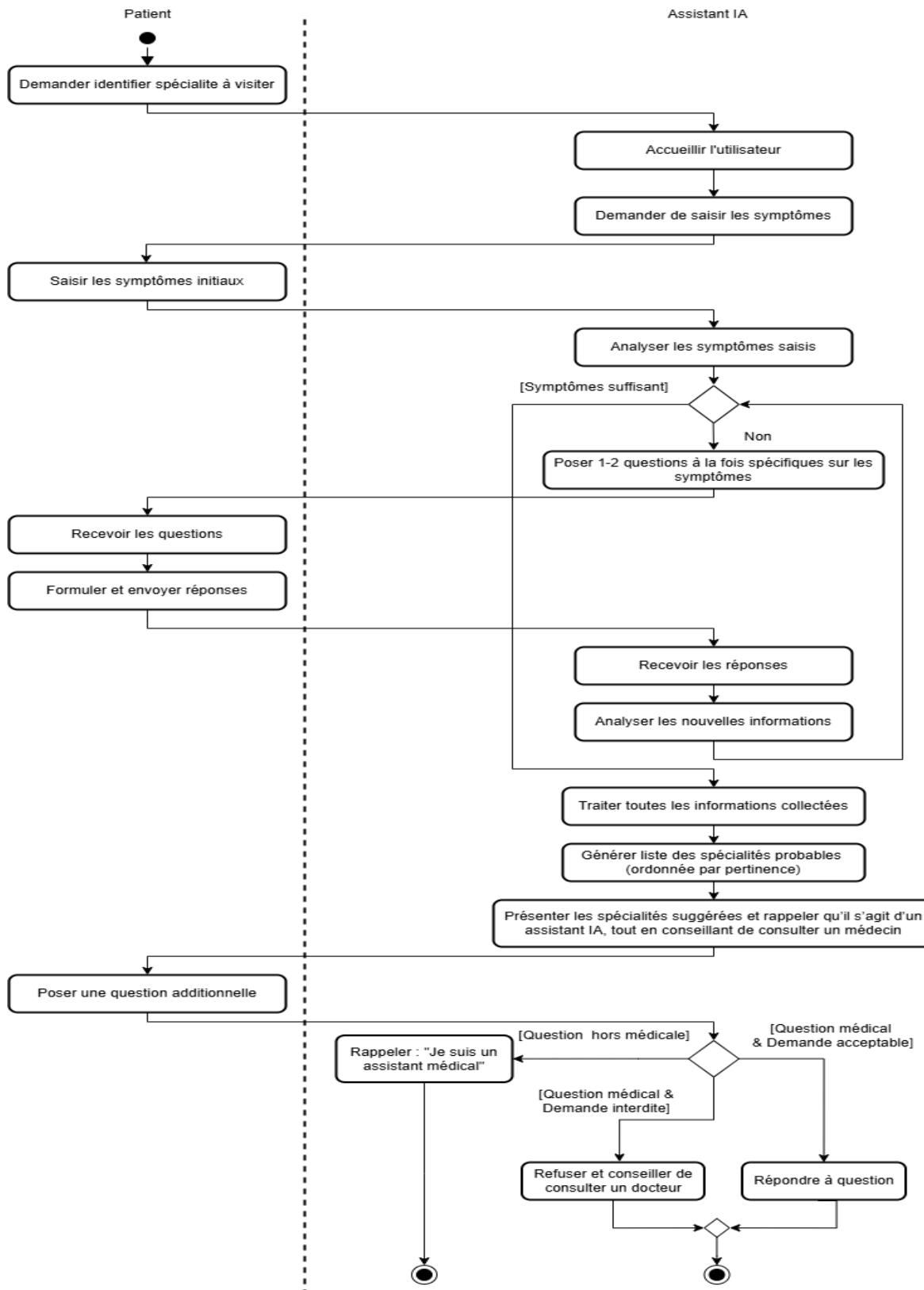


Figure 4.5 : Diagramme d'activité d'assistant AI

4.1.3 Réalisation

Afin d'illustrer la mise en œuvre du la troisième sprint, nous décrirons dans cette partie les interfaces graphiques des principales fonctionnalités réalisées

4.1.4.1 Espace logement

- **Interface de messagerie**

La figure suivante représente l'interface de messagerie

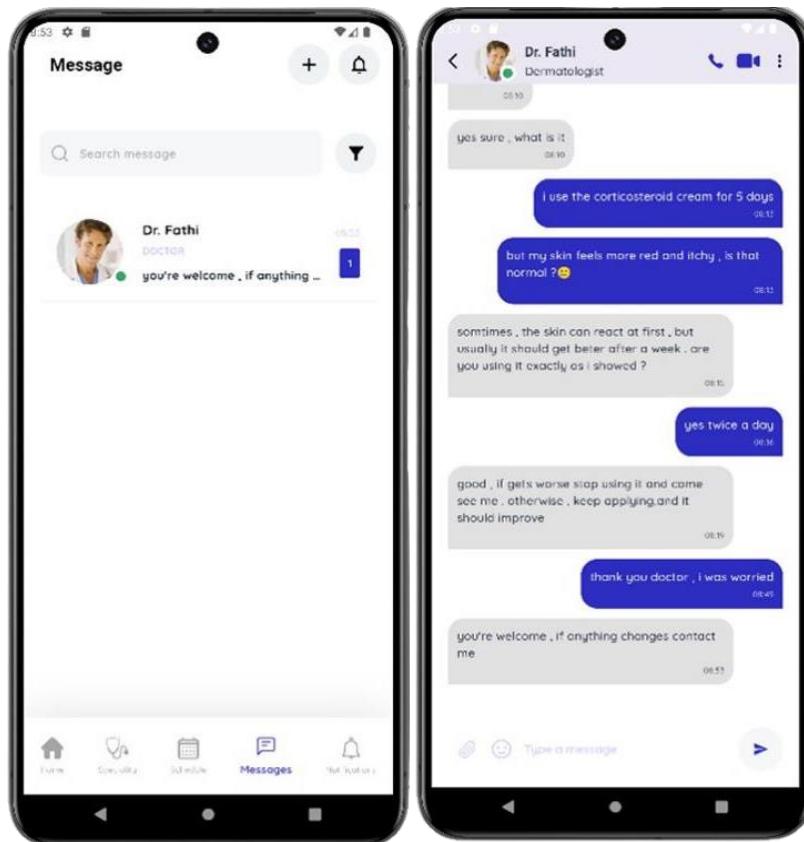


Figure 4.6 : Interface de messagerie entre docteur et patient

Dès que l'utilisateur authentifié clique sur l'option « Messages » dans la barre de navigation inférieure, il accède à l'interface de messagerie. Il peut consulter la liste de toutes ses discussions avec les autres utilisateurs. Il peut sélectionner une conversation pour afficher l'historique complet des échanges. Il peut saisir un nouveau message dans le champ de texte en bas et cliquer sur le bouton d'envoi pour le transmettre.

- **Interface de conversation avec d'Assistant IA**

La figure suivante représente l'interface d'Assistant IA

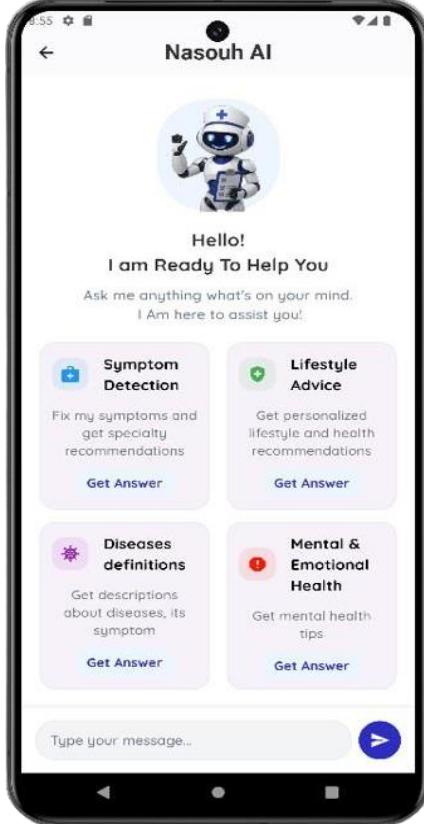


Figure 4.7 Interface de discussion avec Assistant AI

Dès que le patient accède à l'interface de communication avec l'Assistant IA, il découvre en haut une série de cartes présentant les principales fonctionnalités.

Il peut cliquer sur l'une de ces cartes pour envoyer automatiquement un message prérempli à l'assistant.

Il peut également saisir librement son propre message dans le champ de saisie en bas de l'écran et cliquer sur l'icône d'envoi pour démarrer la conversation.

- **Interface de vidéoconférence**

La figure suivante représente l'interface de vidéoconférence

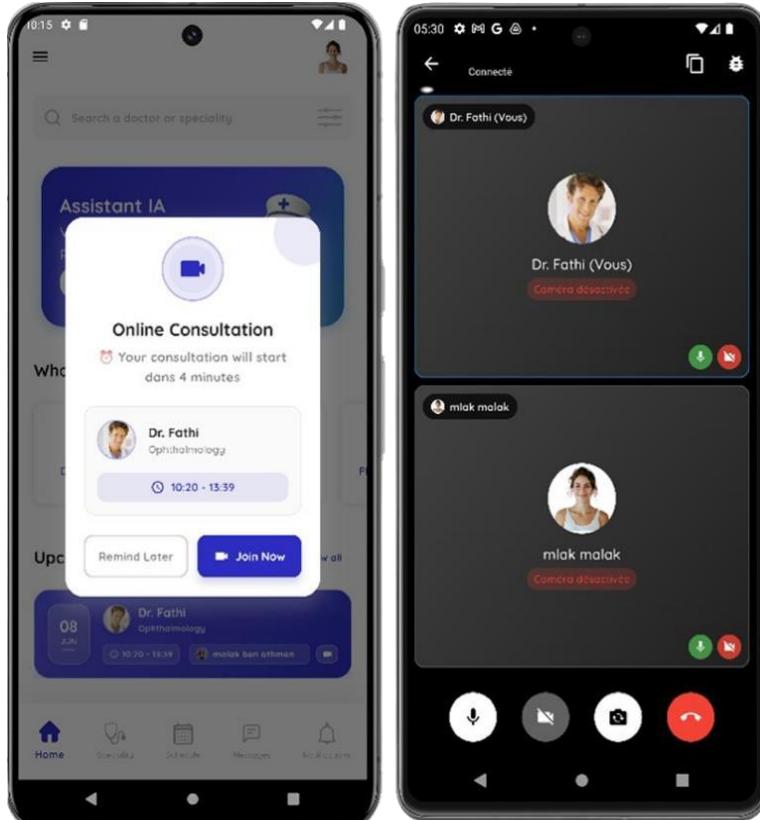


Figure 4.8 : Interface de Visio conférence

Dès qu'un médecin authentifié dispose d'un rendez-vous programmé dans moins de cinq minutes, une boîte de dialogue l'informe que «la consultation en ligne débute dans 5 minutes». Il peut cliquer sur « Me rappeler plus tard » pour fermer la boîte et revenir à l'application.

Il peut cliquer sur « Rejoindre maintenant » pour accéder directement à la salle de visioconférence.

Une fois dans la salle, il reste en attente d'une demande de rejoindre de la part d'un patient : il reçoit une notification et peut accepter la demande pour démarrer la session.

Il peut activer ou désactiver sa caméra en cliquant sur l'icône correspondante, couper ou rétablir son micro via l'icône micro, et terminer la consultation en cliquant sur l'icône de fin d'appel.

5. CHAPITRE 5 :

Release 3 « Module Tourisme »

5.1 Développement du Sprint 4 « Services de bien-être et d'hébergement »

5.1.1 Backlog du sprint 4

Dans ce tableau nous définissons les tâches qui seront effectuées pour chaque user story durant le sprint 4

Tableau 5.1 : Backlog du sprint 4

Fonctionnalité	ID	User Story	Tâches	Story Point
Consultation des logements	42	En tant qu'utilisateur, je souhaite consulter les logements les plus proches de ma localisation et celles les plus évalués	-Implémenter la page de liste des logements en Flutter -Développer des deux l'API de cette partie -Implémenter la pagination des résultats	Difficile
Consultation des détails d'un logement	43	En tant qu'utilisateur, je peux consulter les détails d'un logement, y compris les services plus proches, afin de m'aider à faire le meilleur choix en fonction de mes besoins	-Développer la page des détails d'un logements en Flutter -Implémenter l'API d'afficher détails d'un logement -Implémenter l'API d'afficher les services les plus proches d'un logement	Difficile
Contact avec propriétaire	44	En tant qu'utilisateur, je souhaite contacter le propriétaire par message ou par téléphone	-Développer section de contact propriétaire en Flutter -Gérer les permissions d'accès aux appels et aux SMS sur iOS et Android et implémenter ces fonctionnalités	Facile

Consultation disponibilité d'un logement	45	En tant qu'utilisateur, je souhaite connaître la disponibilité d'un Logement	-Développer la page de Disponibilité d'un logement en Flutter avec le composant calendrier -Développer l'API de vérification de disponibilité	Moyenne
Consultation chambres	46	En tant qu'utilisateur, je souhaite voir quelles chambres sont disponibles dans un logement partagé	-Développer la page des chambres disponibles en Flutter -Développer l'API de disponibilité d'une chambre	Facile
Filtrage des logements	47	En tant qu'utilisateur, je souhaite filtrer les logements afin d'affiner ma recherche selon mes critères	-Développer la « Bottom Sheet » de filtrage d'un logement en Flutter -Implémenter l'API de filtrage d'un logement	Moyenne
Visualisation cartographique	48	En tant qu'utilisateur, je souhaite visualiser les résultats du filtrage sur une carte interactive, avec les prix des logements affichés directement sur celle-ci	-Intégrer «OpenStreetMa» dans Flutter et développer le composant de carte dynamique -Implémenter l'affichage des prix sur la carte -Faire l'intégration avec l'API de -Filtrage pour afficher la liste des logements	Difficile
Gestion des favoris	49	En tant qu'utilisateur, je souhaite ajouter/ supprimer un logement à ma liste de favoris	-Intégrer la logique d'ajouter et de supprimer d'un logement à la liste de favoris	Facile

	50	En tant qu'utilisateur, je souhaite consulter ma liste de logements favoris	-Développer la page de la liste des logements favoris en Flutter -Développer l'API d'afficher les logements favoris	Facile
Réservation d'un Hébergement	51	En tant qu'utilisateur, je souhaite pouvoir réserver un logement entier ou une chambre d'un logement partagé pour mon séjour	-Développer la page de réservation en Flutter, en Indiquant les détails à confirmer selon type de réservation. -Développer l'API de réservation	Moyenne
Consultation d'Historique des réservations	52	En tant qu'utilisateur, je souhaite consulter mes réservations passées.	-Développer la page l'historique de réservation d'un logement en Flutter -Développer cette API et l'intégrer.	Moyenne
Consulter des endroits du bien-être	53	En tant que patient, je peux consulter une liste des services offertes par des endroits liés aux relaxations et au bien-être afin de découvrir des services qui correspondent à mes besoins personnels	-Développer la page en Flutter, y compris la section de : la liste des services, la liste des endroits à proximité/plus évalués -Implémenter les apis de cette partie -Développer la fonctionnalité de tri par popularité	Difficile
Consulter des endroits du bien-être	54	En tant que patient, je peux consulter la liste des endroits à proximité ainsi que les lieux les plus populaires, afin de choisir facilement des		

		options accessibles et bien notées		
Filtrage des endroits du bien-être	55	En tant que patient, je peux filtrer les lieux par type, activités, services et localisation, et visualiser les résultats directement sur une carte géographique, afin d'optimiser ma recherche selon mes préférences	-Implémenter api filter -Développer le « bottom sheet » de filtrage -Développer page de l'affichage des résultats sur une carte géographique	Moyenne
Consulter les détails d'un endroit	56	En tant que patient, je veux consulter les détails d'un lieu afin de prendre une décision claire avant de le visiter	-Implémenter api de détails d'endroit -Implémenter la page de détails d'un endroit	Facile
Consulter les détails d'une offre d'un endroit	57	En tant que patient, je veux consulter les détails d'une offre spécifique, afin de connaître précisément les prestations incluses	-Implémenter api de détails d'une offre -Implémenter la page de détails d'une offre en flutter	Facile
Gestion des endroits de bien-être favoris	58	En tant que patient, je peux ajouter un lieu à ma liste de favoris ou le retirer, afin de sauvegarder mes préférences et faciliter mes choix futurs	<ul style="list-style-type: none"> • Implémenter les deux api d'ajout/supprime aux/de favoris • Créer la vue de liste des favoris 	Facile

5.1.2 Analyse

Pour la spécification fonctionnelle du sprint 4, nous présentons les diagrammes de cas d'utilisation global et détaillés

5.1.2.1 Diagramme de cas d'utilisation global du sprint 4

Ce diagramme de cas d'utilisation présente les différentes fonctionnalités que le docteur et le patient peuvent utiliser durant le sprint 4

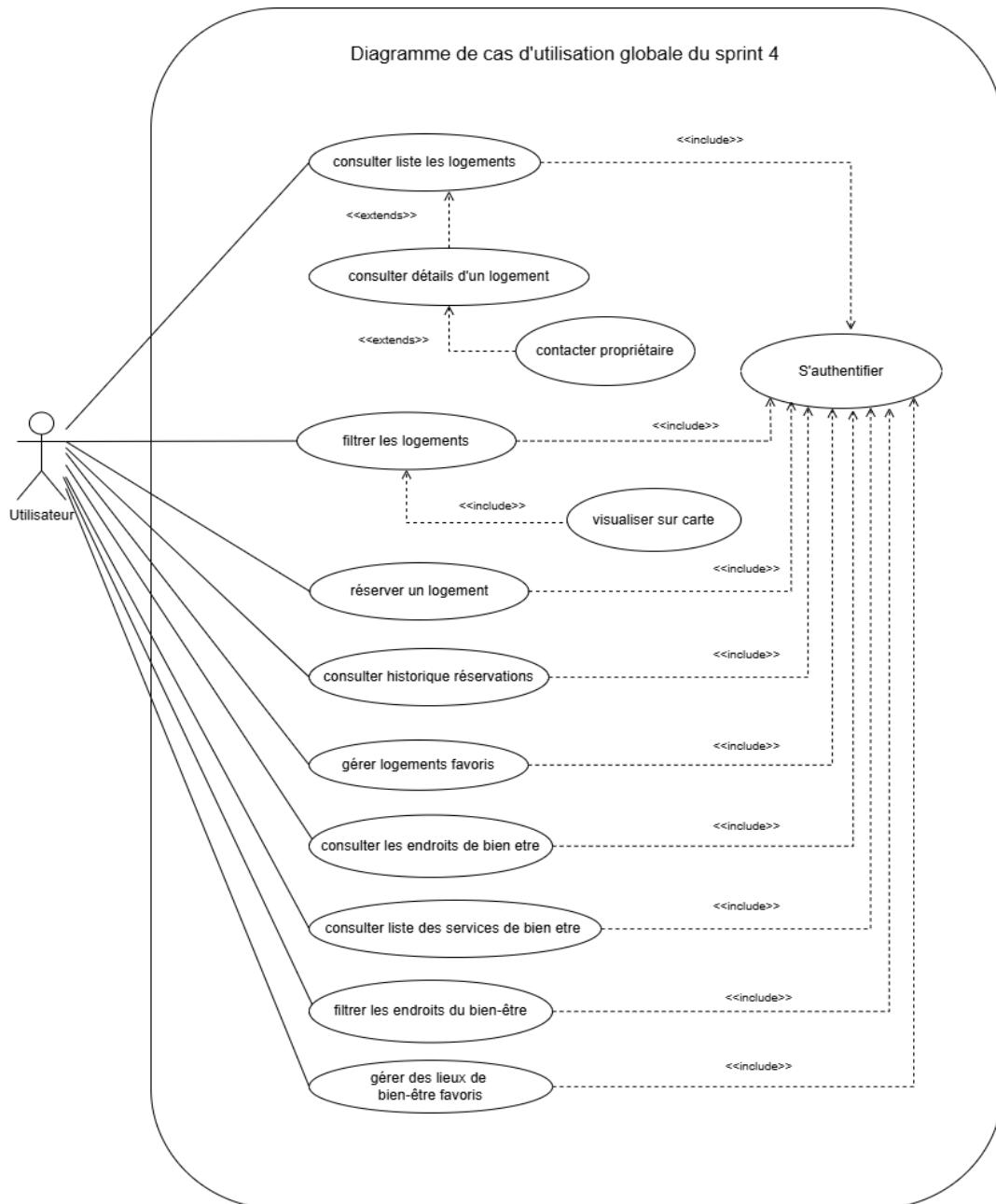


Figure 5.1 Diagramme de cas d'utilisation globale de sprint 4

5.1.2.2 Diagrammes de cas d'utilisation détaillés

Dans la suite, nous détaillons les diagrammes de cas d'utilisation pour les principales fonctionnalités

- **Diagramme de cas d'utilisation détaillée « Consulter les détails d'un logement »**

Cette figure présente les détails du cas d'utilisation consulter les détails d'un logement

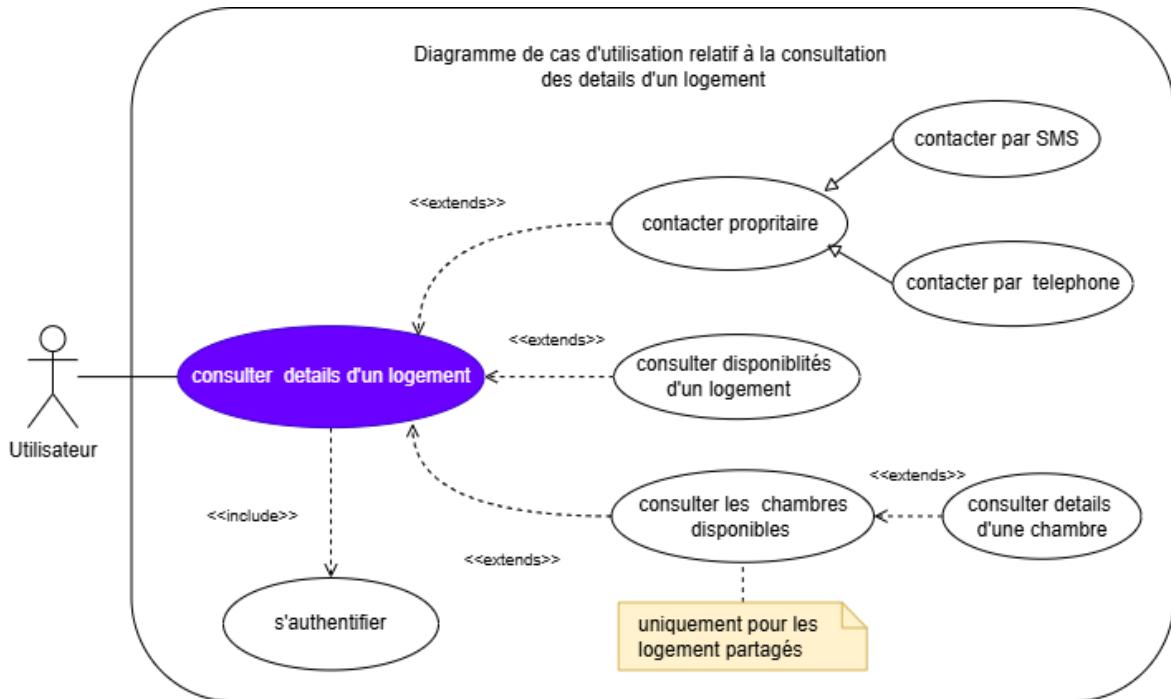


Figure 5.2 Diagramme de cas d'utilisation détaillée de consulter détails logement

- **Diagramme de cas d'utilisation détaillée « Gérer favoris »**

Cette figure présente les détails du cas d'utilisation gérer favoris des logements

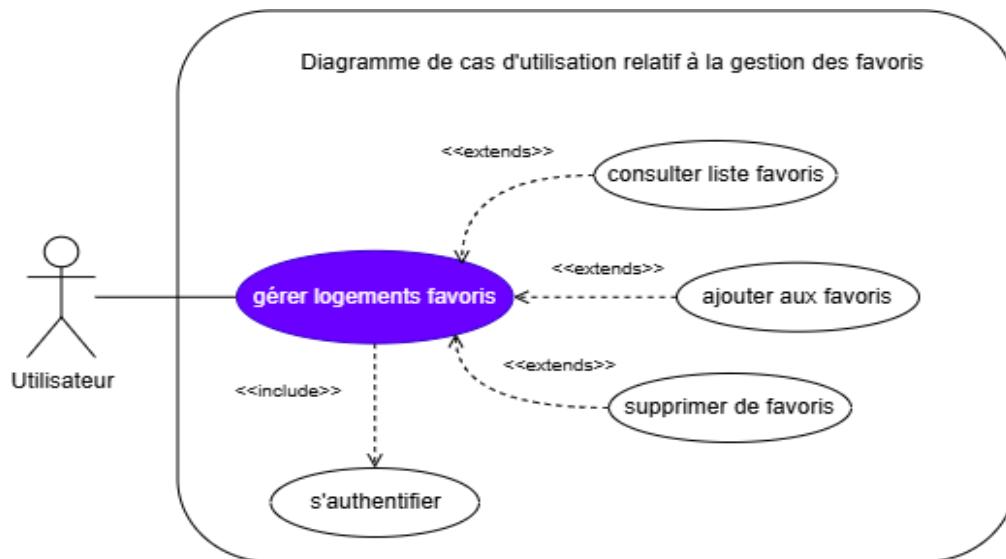


Figure 5.3 : Diagramme de cas d'utilisation détaillée de gestion favoris logement

- **Diagramme de cas d'utilisation détaillée « Filtrer les logements »**

Cette figure présente les détails du cas d'utilisation filtrer les logements

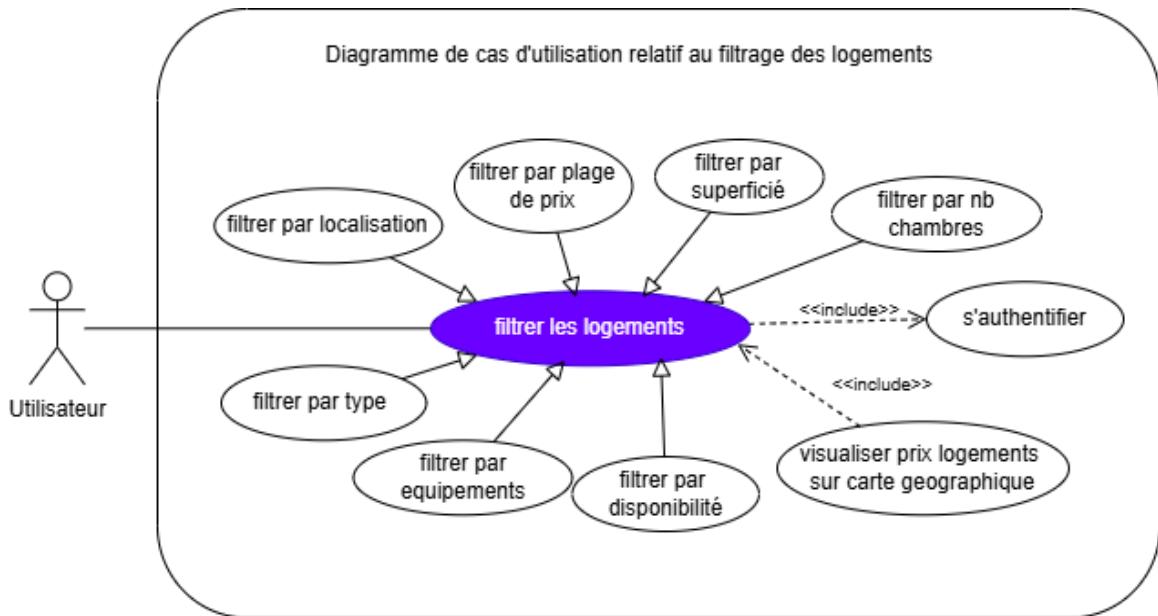


Figure 5.4 : Diagramme de cas d'utilisation détaillée de filtrage logement

- **Diagramme de cas d'utilisation détaillée « Réserver un logement »**

Cette figure présente les détails du cas d'utilisation réservation un logement

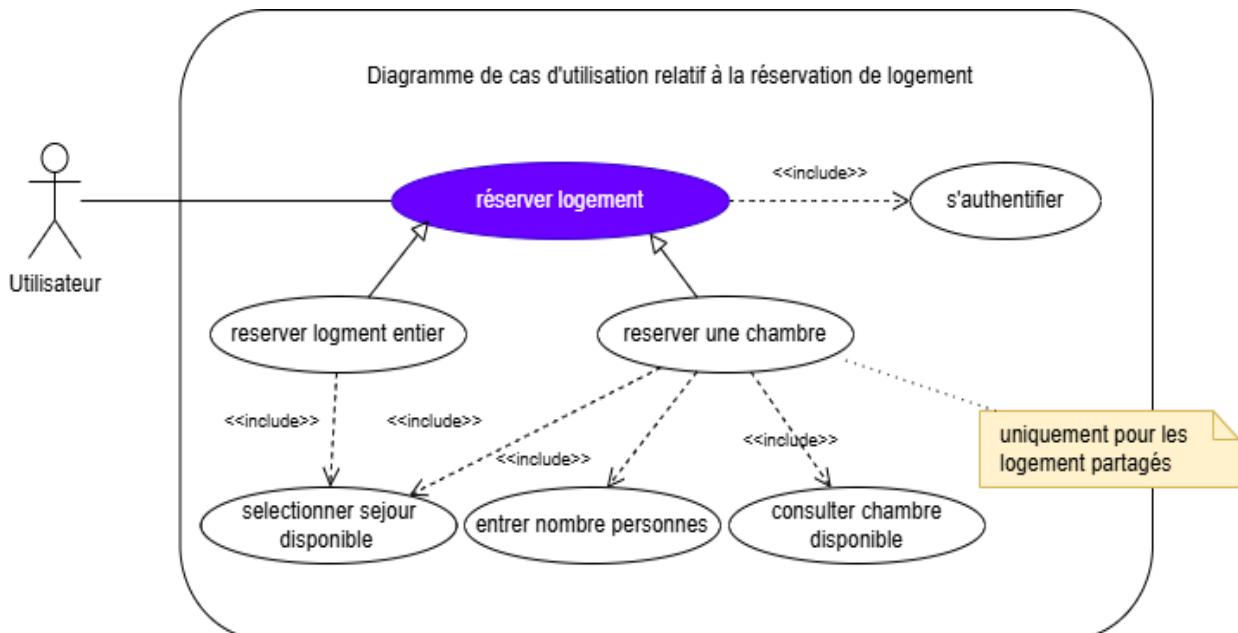


Figure 5.5 : Diagramme de cas d'utilisation détaillée de réservation d'un logement

- **Diagramme de cas d'utilisation détaillée « Consulter les endroits de bien-être »**

Cette figure présente les détails du cas d'utilisation de consultation des endroits de bien-être

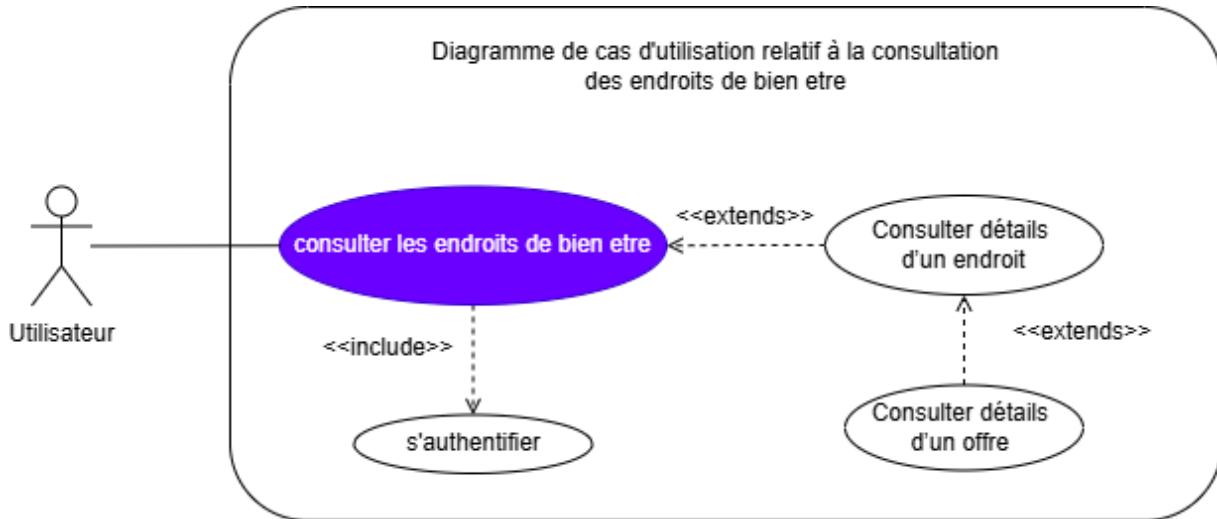


Figure 5.6 : Diagramme de cas d'utilisation de consulter les endroits de bien être

- **Diagramme de cas d'utilisation détaillée « Filtrage des endroits de bien-être »**

Cette figure présente les détails du cas d'utilisation de filtrage des endroits de bien-être

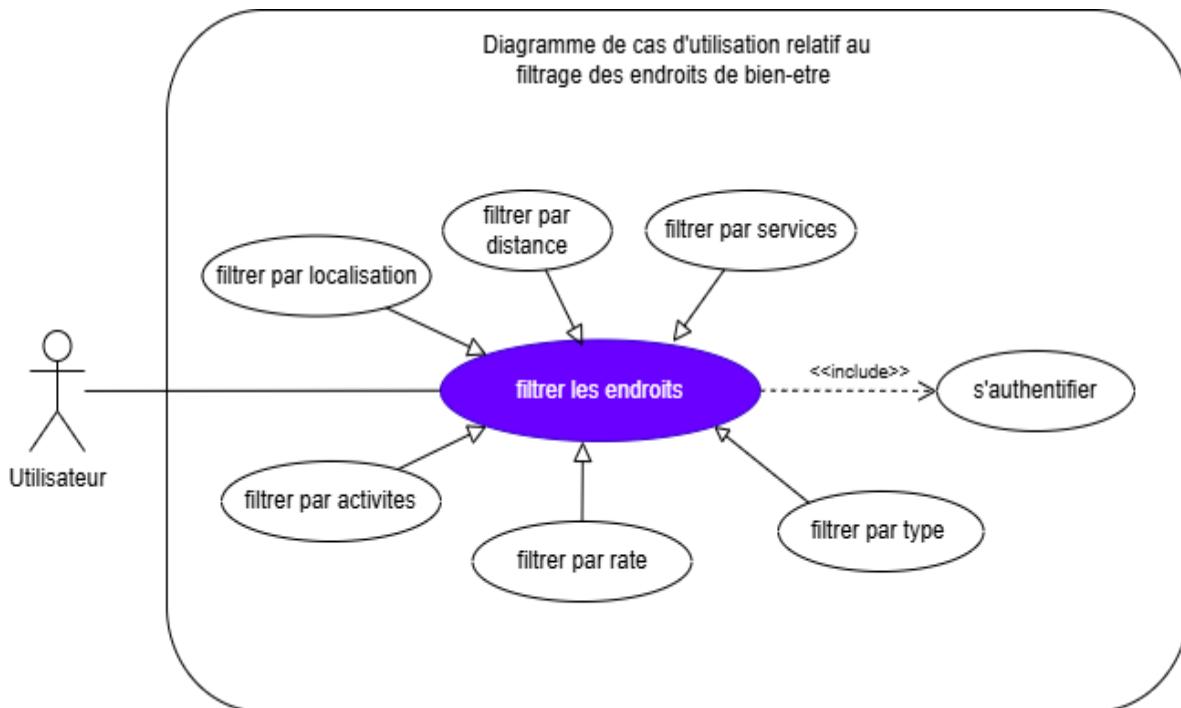


Figure 5.7 : Diagramme de cas d'utilisation détaillée de filtrage des endroits de bien être

- **Diagramme de cas d'utilisation détaillée « Gestion des favoris »**

Cette figure présente les détails du cas d'utilisation de gestion de favoris des endroits de bien-être

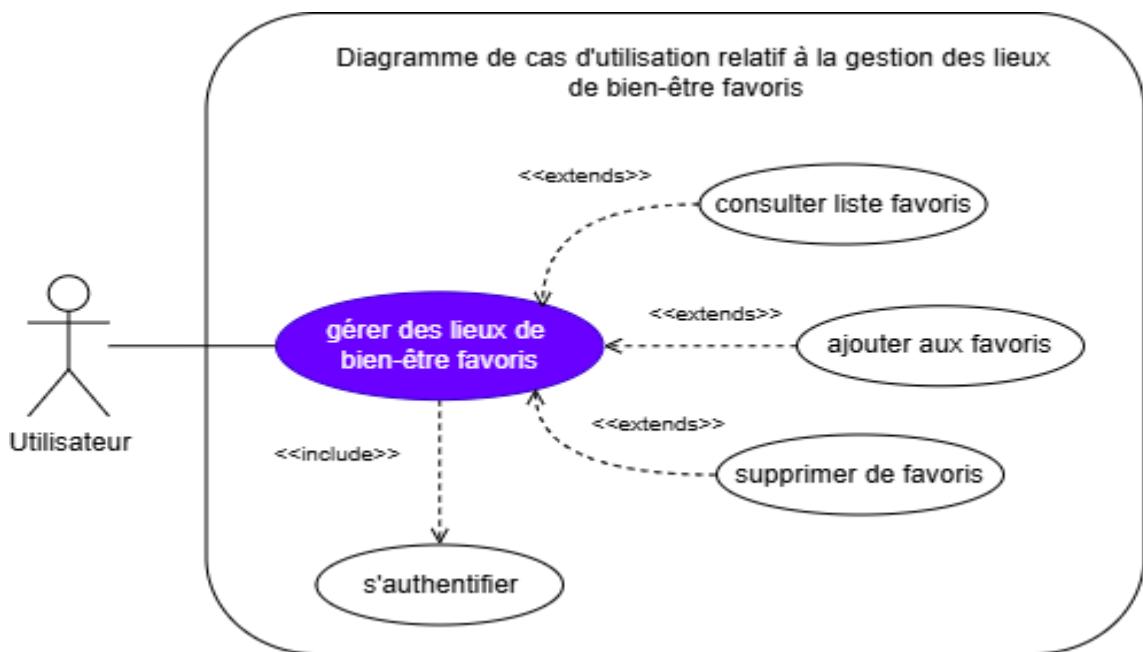


Figure 5.8 : Diagramme de cas d'utilisation détaillée de gestion des lieux de bien être favoris

5.1.2.3 Description textuelle des cas d'utilisation

- **Description textuelle des cas d'utilisation « filtrer un logement »**

Tableau 5.2 : Description textuelle de cas d'utilisation "Filtrer un logement"

Cas d'utilisation : Filter un logement
Acteur(s) : Utilisateur
Objectif : Filtrer un logement correspondant à ses critères
Pré-condition(s) :
<ul style="list-style-type: none"> • L'utilisateur a accès au système de gestion de logements • Des logements sont disponibles dans le système
Post-condition(s) :
<ul style="list-style-type: none"> • L'utilisateur visualise une liste filtrée de logements correspondant à ses critères • L'utilisateur peut voir les logements sur une carte interactive

Scénario principal :

1. L'utilisateur accède à la page de recherche de logements
2. Le système affiche le formulaire de filtrage avec les critères disponibles
3. L'utilisateur sélectionne un ou plusieurs des critères suivants :
 - a. Disponibilité (dates de début et de fin)
 - b. Localisation (pays, ville)
 - c. Type de logement (maison, studio, maison d'hôte, etc.)
 - d. Plage de prix (min - max)
 - e. Plage de superficie (min - max)
 - f. Équipements disponibles (Wi-Fi, climatisation, piscine, etc.)
 - g. Nombre de chambres/cuisines
4. L'utilisateur valide sa sélection de critères
5. Le système recherche dans la base de données les logements correspondant aux critères
6. Le système affiche les résultats sous forme de liste
7. Le système affiche simultanément une carte dynamique avec les logements positionnés et leurs prix affichés
8. L'utilisateur peut sélectionner un logement soit depuis la liste, soit depuis la carte
9. Le système affiche les détails du logement sélectionné

Scénario alternatif :

- 6.1 L'utilisateur peut à tout moment réinitialiser les critères
 - o L'utilisateur clique sur "Réinitialiser"
 - o Le système efface tous les critères sélectionnés
- 5.1 Aucun logement ne correspond aux critères sélectionnés
 - o Le système affiche un message indiquant qu'aucun résultat ne correspond aux critères
 - o Le système suggère à l'utilisateur de modifier ses critères
 - o L'utilisateur peut ajuster les critères et relancer la recherche

- **Description textuelle des cas d'utilisation « Réserver un logement »**

Tableau 5.3 : Description textuelle de cas d'utilisation « Réserver un logement »

Cas d'utilisation : Réserver un logement
Acteur(s) : Utilisateur
Objectif : Réserver un logement
Pré-condition(s) : L'utilisateur est authentifié dans le système L'utilisateur a sélectionné un logement à consulter
Post-condition(s) : <ul style="list-style-type: none"> • La réservation du logement entier ou de la chambre est enregistrée dans le système • L'utilisateur reçoit une confirmation de sa réservation
Scénario principal : <ol style="list-style-type: none"> 1. Le système affiche les détails du logement sélectionné 2. Le système détermine si le logement est de type partagé ou non 3. Si le logement n'est pas partagé : <ol style="list-style-type: none"> a. Le système affiche un bouton "Réserver logement entier" b. L'utilisateur clique sur ce bouton c. Le système affiche un formulaire pour saisir les dates de séjour et le nombre de personnes d. L'utilisateur remplit le formulaire et le valide e. Le système vérifie les informations saisies 4. Si le logement est partagé : <ol style="list-style-type: none"> a. Le système affiche un bouton "voire disponibilité des chambres" b. L'utilisateur clique sur ce bouton c. Le système affiche un formulaire pour saisir les dates de séjour et le nombre de personnes d. L'utilisateur remplit le formulaire et le valide e. Le système vérifie les informations saisies f. Le système recherche les chambres disponibles selon les critères g. Le système affiche la liste des chambres disponibles h. L'utilisateur sélectionne une chambre i. Le système affiche les détails de la chambre et un bouton "Réserver"

- | |
|---|
| <p>j. L'utilisateur clique sur "Réserver"</p> <p>5. Le système affiche un récapitulatif de la réservation (logement entier ou chambre)</p> <p>6. L'utilisateur confirme la réservation</p> <p>7. Le système vérifie la disponibilité finale</p> <p>8. Le système enregistre la réservation</p> <p>9. Le système affiche une confirmation de réservation</p> |
|---|

Scénario alternatif :

- | |
|--|
| <p>3.d / 4.d. Les informations saisies sont incorrectes ou incomplète</p> <ul style="list-style-type: none">○ Le système affiche un message d'erreur spécifique○ L'utilisateur corrige les informations et soumet à nouveau <p>4.g Aucune chambre n'est disponible pour les dates sélectionnées</p> <ul style="list-style-type: none">○ Le système affiche un message indiquant qu'aucune chambre n'est disponible○ Le système propose à l'utilisateur de modifier ses dates de séjour <p>7.1 Le logement/la chambre est devenu(e) indisponible entre-temps</p> <ul style="list-style-type: none">○ Le système affiche un message d'erreur indiquant que le logement/la chambre n'est plus disponible○ Le système propose à l'utilisateur de revenir à la recherche ou de modifier ses critères |
|--|

- **Description textuelle des cas d'utilisation « Consulter les logements »**

Tableau 5.4 : Description textuelle de cas d'utilisation « Consulter les logements »

Cas d'utilisation : Consulter les logements
Acteur(s) : Utilisateur
Objectif : Consultation d'une liste des logements
Pré-condition(s) : <ul style="list-style-type: none">• L'utilisateur a accès au système de gestion de logements
Post-condition(s) : <ul style="list-style-type: none">• L'utilisateur visualise les détails du logement sélectionné
Scénario principal : <ol style="list-style-type: none">1. L'utilisateur accède à la page principale du système2. Le système affiche une liste des logements disponibles3. L'utilisateur sélectionne un logement pour voir ses détails4. Le système affiche les informations détaillées du logement (description, photos, prix, localisation, équipements, disponibilité, évaluations)5. Si le logement est de type partagé, le système affiche également un bouton pour Consulter la disponibilité des chambres
Scénario alternatif : <p>5.1 Le logement déjà favoris alors le système retire le de la liste des favoris</p>

- **Description textuelle de cas d'utilisation « filtrer les endroits »**

Tableau 5.5 : Description textuelle de cas d'utilisation « Filtrer les endroits »

Cas d'utilisation : Filter les endroits
Acteur(s) : Utilisateur
Objectif : Affiner la recherche d'endroits de bien-être selon des critères spécifiques et visualiser les résultats sur une carte
Pré-condition(s) : <ul style="list-style-type: none">• L'utilisateur doit être authentifié
Post-condition(s) : <ul style="list-style-type: none">• Une liste filtrée d'endroits correspondant aux critères est affichée et visualisable sur la carte interactive
Scénario principal : <ol style="list-style-type: none">1. L'utilisateur accède à la liste des endroits de bien-être2. L'utilisateur sélectionne l'option de filtre3. Le système affiche les différentes catégories de filtres disponibles (type d'établissement, activités, services, localisation)4. L'utilisateur sélectionne les critères de filtre souhaités5. Le système applique les filtres en temps réel6. Le système affiche les résultats filtrés sous forme d'une liste7. L'utilisateur demande la visualisation sur carte8. Le système affiche les endroits filtrés sur une carte géographique avec des marqueurs correspondant à chaque établissement

Scénario alternatif :

3. Aucun résultat ne correspond aux critères de filtrage :
 - Le système affiche un message indiquant qu'aucun endroit ne correspond aux critères
 - Le système suggère de modifier ou d'élargir les critères de recherche
7. Problème de chargement des données :
 - Le système affiche un message d'erreur
 - Le système propose à l'utilisateur de réessayer ou de revenir à la liste complète
8. Problème d'affichage de la carte :
 - Le système affiche un message d'erreur
 -

5.1.3 Conception

5.1.3.1 Diagrammes des séquences

- Diagramme de séquence « Consulter un hébergement »

La figure ci-après présente la réalisation du cas d'utilisation consulter un hébergement

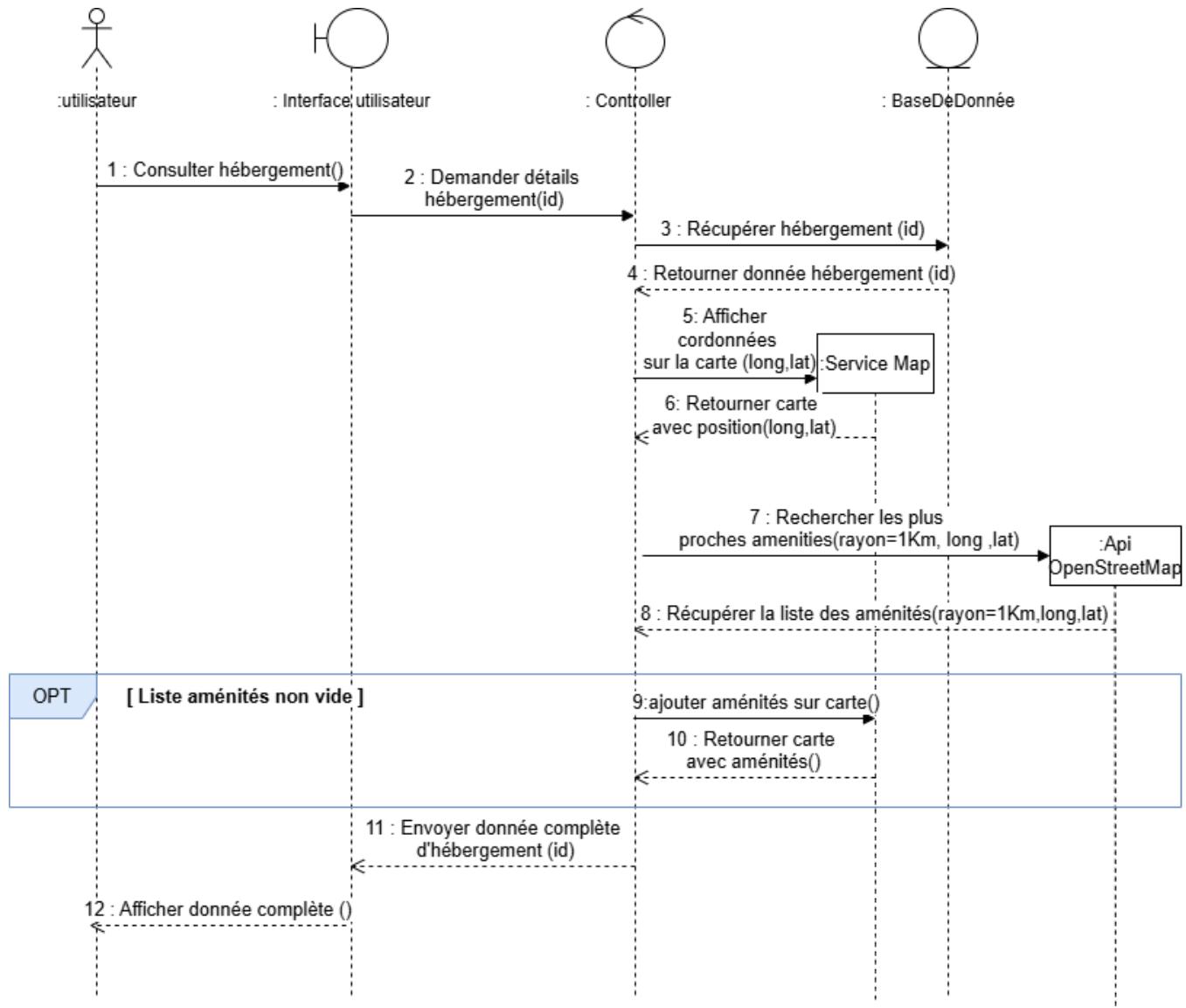


Figure 5.9 : Diagramme de séquence de consulter hébergement

- **Diagramme de séquence « Réserver un logement »**

La figure ci-après présente la réalisation du cas d'utilisation réserver un logement

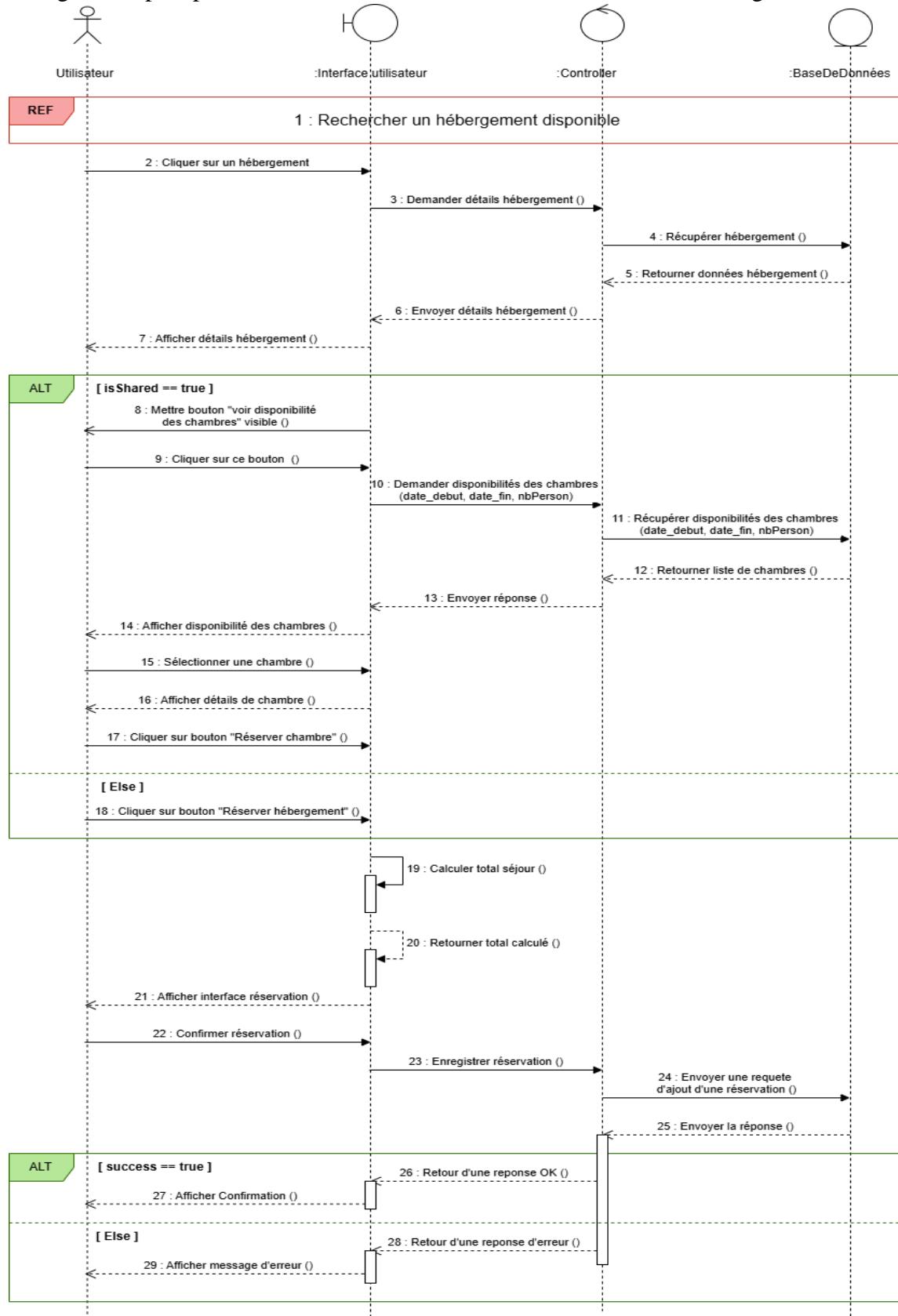


Figure 5.10 : Diagramme de séquence de réserver un hébergement

- **Diagramme de séquence « Filtrer un endroit de bien-être »**

La figure ci-après présente la réalisation du cas d'utilisation filtrer un endroit de bien-être

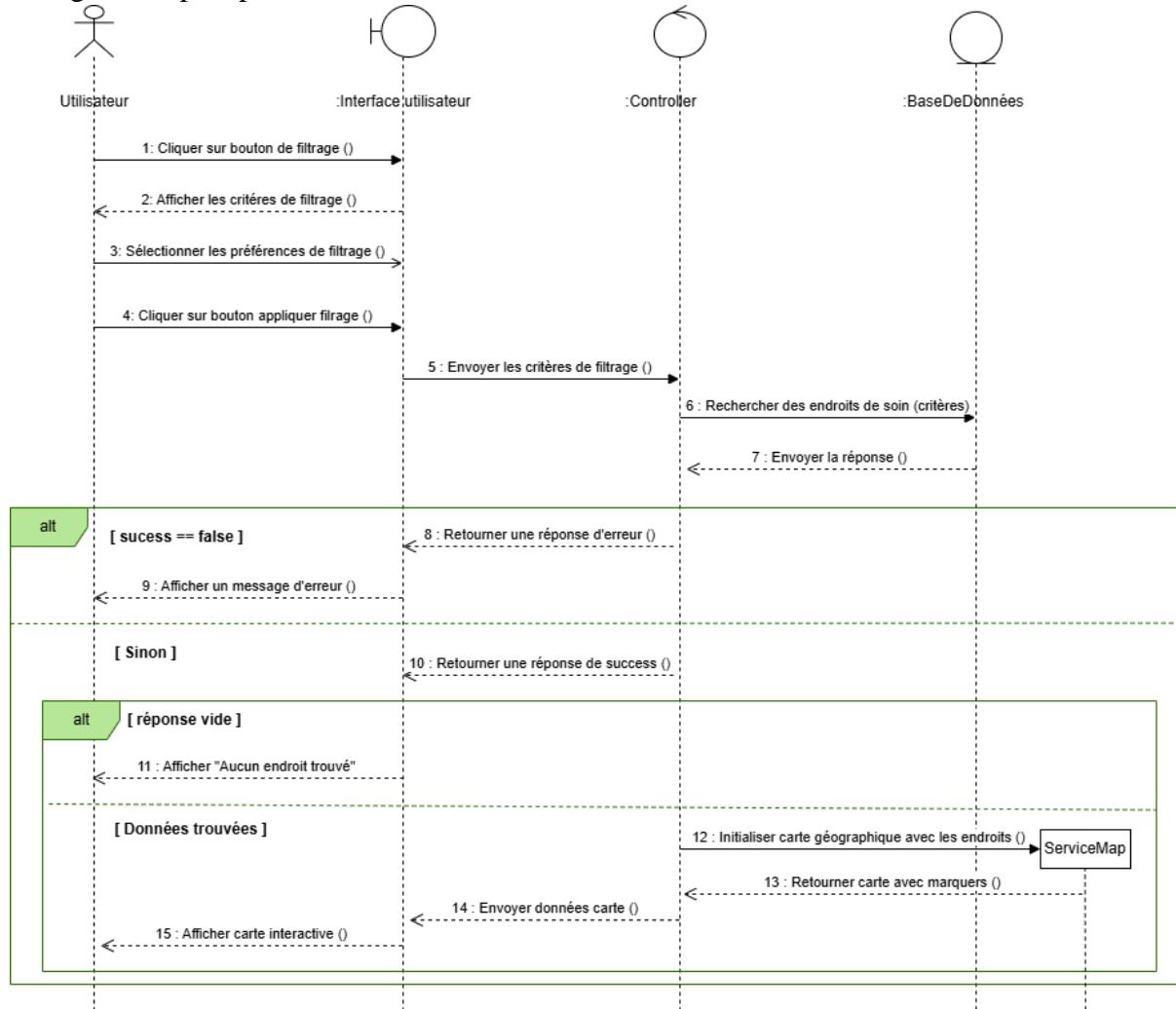


Figure 5.11 : Diagramme de séquence de filtrage d'endroits de bien être

5.1.3.2 Diagramme de classe de sprint 4

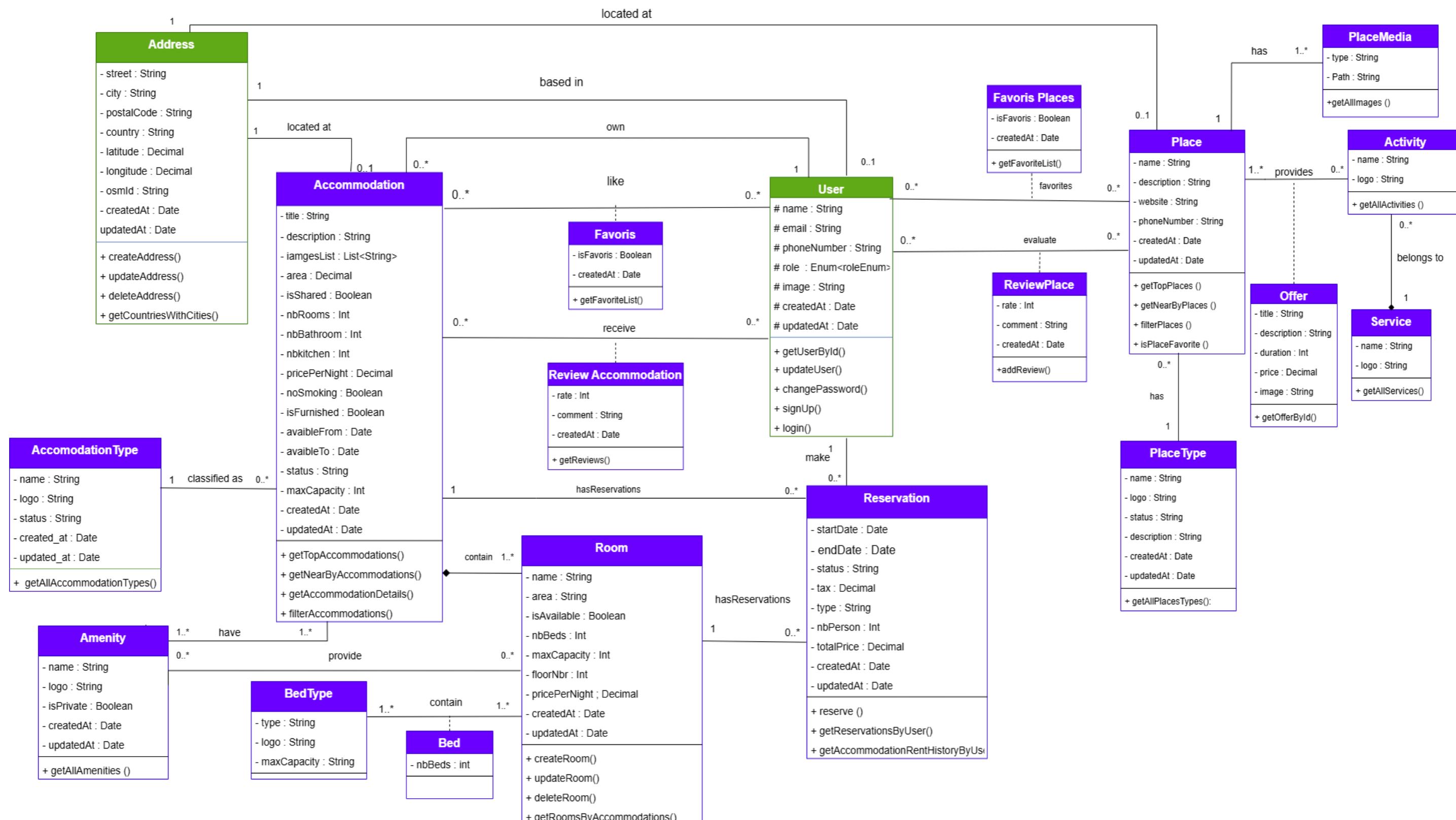


Figure 5.12 : Diagramme de classe sprint 5

5.1.4 Réalisation

Afin d'illustrer la mise en œuvre du dernier sprint, nous décrirons dans cette partie les interfaces graphiques des principales fonctionnalités réalisées

5.1.4.1 Espace logement

• Interface de résultat du filtrage d'un logement

La figure suivante représente l'interface de résultat du filtrage d'un logement

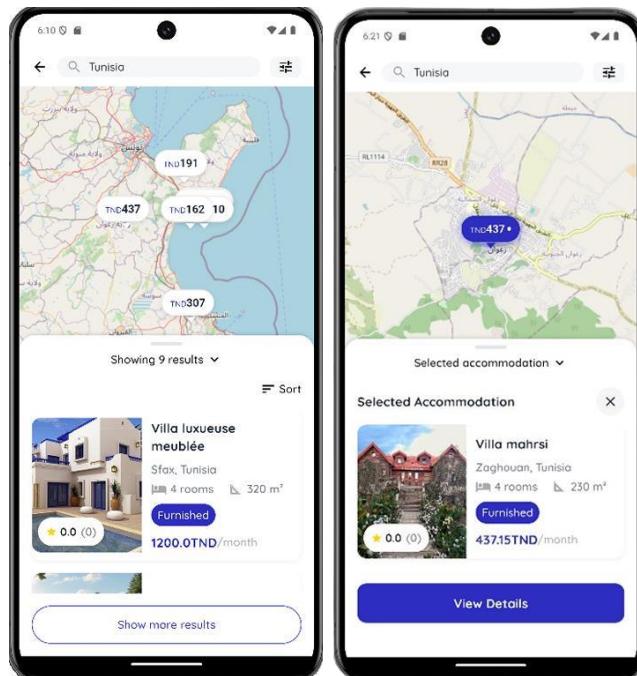


Figure 5.13 : Interface de résultat filtrage logement

Lorsque l'utilisateur clique sur l'icône de filtrage de l'écran d'accueil, saisit ses critères et appuie sur « Filtrer », il voit s'afficher simultanément la liste des logements correspondants et leurs marqueurs de prix sur la carte.

Lorsque l'utilisateur clique sur un logement dans la liste ou sur son marqueur, la carte se centre et zoome sur ce logement, qui devient alors le seul affiché dans la liste, accompagné d'un bouton « Voir détails » pour accéder à son détail

- **Interface de détail d'un logement**

La figure suivante représente l'interface des détails d'un logement

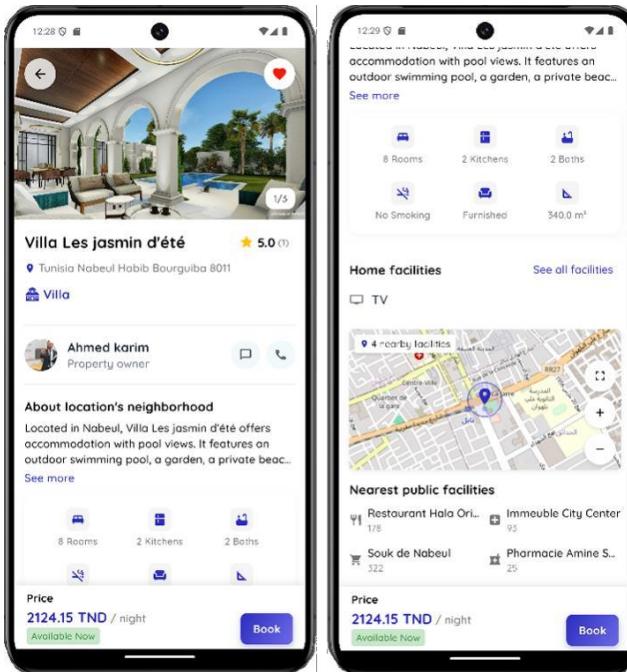


Figure 5.14 : Interface détails logement

Dès que l'utilisateur clique sur un logement, il peut consulter ses détails. Il peut l'ajouter à ses favoris en cliquant sur l'icône en forme de cœur, ou le retirer s'il est déjà dans ses favoris.

Il peut aussi contacter le propriétaire en appelant ou en envoyant un SMS via les icônes prévues.

Il peut également consulter les équipements à proximité sur une carte plein écran, en cliquant sur l'icône « plein écran ».

Il peut réserver le logement en cliquant sur le bouton « Réserver ».

Si le logement est partagé, un bouton « Voir la disponibilité des chambres » s'affiche, lui permettant de consulter les chambres disponibles.

- **Interface de disponibilités des chambres**

La figure suivante représente l'interface de disponibilité des chambres d'un logement

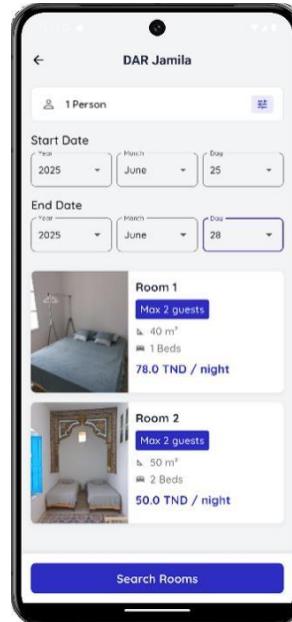


Figure 5.15 : Interface disponibilités chambres

Dès que l'utilisateur clique sur le bouton « Voir la disponibilité des chambres » de l'interface (lorsque le logement est partagé), il est automatiquement redirigé vers la page de disponibilité des chambres. Sur cette page, il doit indiquer : le nombre de personnes, la date de début de réservation, la date de fin de réservation. Il clique ensuite sur le bouton « Rechercher chambres » pour consulter la liste des chambres disponibles correspondant à ses critères.

- **Interface de détail d'une chambre**

La figure suivante représente l'interface de détail d'une chambre

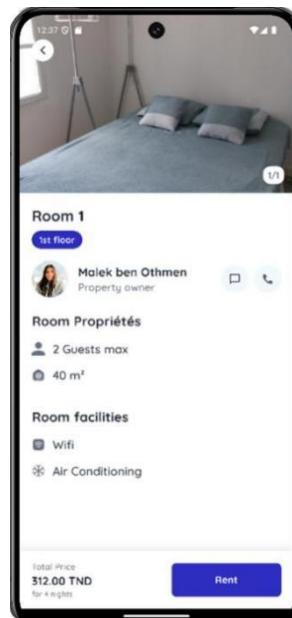


Figure 5.16 : Interface détails chambre

Dès que l'utilisateur clique sur une chambre, il peut consulter ses détails. Il a la possibilité de contacter le propriétaire du logement auquel elle appartient : l'appeler en cliquant sur l'icône téléphone, ou lui envoyer un SMS en cliquant sur l'icône message. Il peut réserver cette chambre en cliquant sur le bouton « Réserver »

5.1.4.2 Espace endroits de bien-être

- **Interface d'accueil**

La figure suivante représente l'interface d'accueil

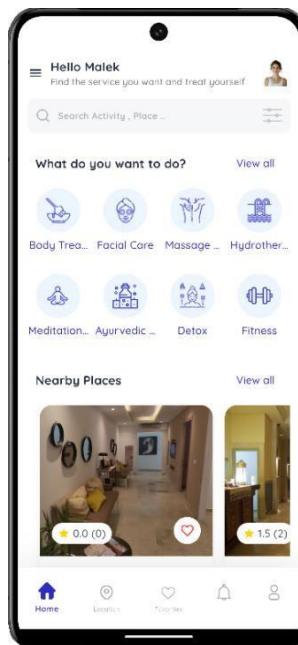


Figure 5.17 : Interface d'accueil de service de bien être

Après s'être connecté, l'utilisateur accède au module « Tourisme » en cliquant sur sa carte. Il est redirigé vers la page d'accueil, où il peut consulter les services de bien-être proposés, la liste des endroits de bien-être les plus proches de sa position ainsi que ceux les mieux notés. Il a la possibilité d'ajouter un endroit à ses favoris en cliquant sur l'icône en forme de cœur, ou de le retirer s'il l'avait déjà ajouté. Il peut également cliquer sur l'icône de filtrage pour afficher un formulaire de filtrage en bas de l'écran .

- **Interface de formulaire de filtrage d'un endroit de bien-être**

La figure suivante représente l'interface de formulaire de filtrage d'un endroit de bien-être

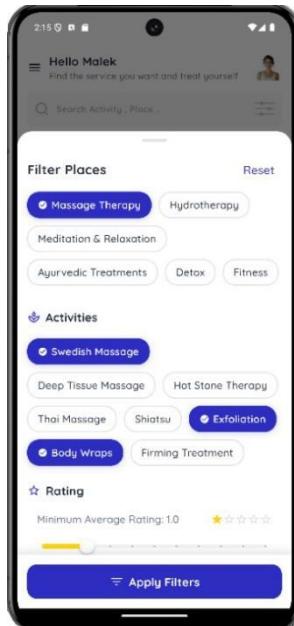


Figure 5.18 : Interface de filtrage d'endroit de bien être

Sur cette interface, l'utilisateur peut définir ses critères de filtrage afin de trouver un endroit de bien-être qui correspond à ses préférences. Il peut filtrer les résultats en fonction des services proposés (spa, massage, etc.), des activités disponibles (yoga, méditation, etc.), de la note moyenne des endroits, ainsi que de la localisation géographique souhaitée. Une fois les critères sélectionnés, il lui suffit de cliquer sur le bouton « Appliquer les filtres » pour afficher les lieux correspondant à sa recherche

- **Interface de détail d'un endroit de bien-être**

La figure suivante représente l'interface de détail d'un endroit de bien-être

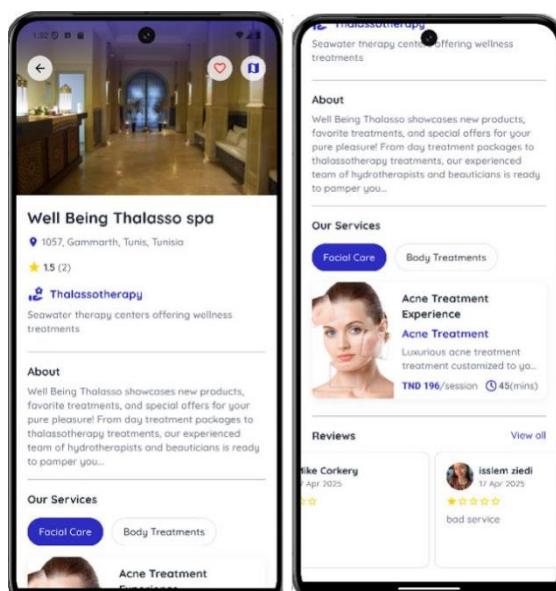


Figure 5.19 : Interface détails d'un endroit de bien être

Dès que l'utilisateur sélectionne un endroit depuis la liste, il accède à sa fiche détaillée. Il a également la possibilité d'ajouter cet endroit à ses favoris en cliquant sur l'icône en forme de cœur, ou de le retirer de ses favoris en recliquant sur cette même icône. De plus, en cliquant sur l'icône de carte, il peut visualiser la localisation exacte de l'endroit sur une carte interactive

- **Interface de détail d'une offre**

La figure suivante représente l'interface de détail d'une offre

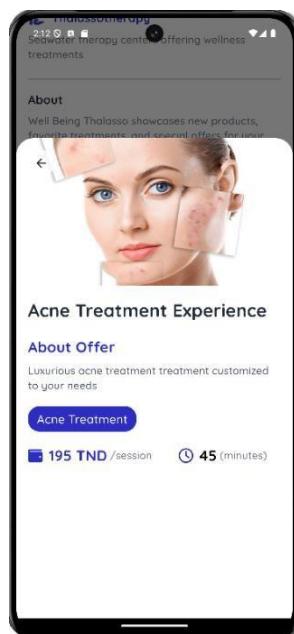


Figure 5.20 : Interface détails d'une offre

Dès que l'utilisateur clique sur la carte d'une offre, l'interface affiche en bas de l'écran les détails de cette offre.

Conclusion générale

Ce rapport retrace en détail le processus de développement de notre plateforme dédiée au tourisme médical, intégrant les modules essentiels tels que la gestion des patients, des médecins, de l'hébergement et des services de bien-être.

L'adoption d'une architecture monolithique modulaire, guidée par une approche **Domain-Driven Design**, a joué un rôle central dans la structuration et l'évolutivité du système, garantissant une meilleure cohérence et une grande flexibilité pour l'intégration de nouvelles fonctionnalités.

Ce projet a été une opportunité précieuse pour développer et consolider des compétences clés, tant sur le plan technique que personnel. Il nous a permis de renforcer nos connaissances en technologies émergentes, en conception logicielle, ainsi qu'en gestion sécurisée des données. Parallèlement, nous avons affiné notre capacité à gérer le temps, à travailler efficacement sous pression et à collaborer en équipe.

Malgré les avancées réalisées, plusieurs axes d'amélioration demeurent. Une meilleure structuration et accessibilité des dossiers médicaux permettrait d'optimiser le travail des professionnels de santé. Par ailleurs, le renforcement de l'assistant intelligent pour générer automatiquement des documents et proposer des traductions adaptées aux besoins des médecins faciliterait la communication avec des patients internationaux. L'extension de la portée et de l'accessibilité de la plateforme permettrait également aux praticiens d'élargir leur champ d'action. Enfin, un renforcement des mécanismes de sécurité garantirait la confidentialité des données et le respect des normes légales en vigueur.

En somme, ce projet nous a offert une immersion concrète dans le monde professionnel, enrichissant notre expertise et nous préparant à relever de nouveaux défis technologiques.

Webographie

- [1] Bun [Consulté le 05/02/2025] [En ligne] <https://bun.sh/docs>
- [2] Hono [Consulté le 05/02/2025] [En ligne] <https://hono.dev/>
- [3] Drizzle ORM [Consulté le 05/02/2025] [En ligne] <https://orm.drizzle.team/docs/get-started>
- [4] Neon [Consulté le 06/02/2025] [En ligne] <https://neon.com/docs/introduction>
- [5] Postgresql [Consulté le 06/02/2025] [En ligne] <https://www.postgresql.org/docs/>
- [6] Supabase [Consulté le 06/02/2025] [En ligne] <https://supabase.com/docs>
- [7] NextJs [Consulté le 07/02/2025] [En ligne] <https://nextjs.org/docs>
- [8] shadcn/ui [Consulté le 07/02/2025] [En ligne] <https://ui.shadcn.com/>
- [9] TanStack Query [Consulté le 07/02/2025] [En ligne] <https://tanstack.com/query/latest>
- [10] ZodValidator [Consulté le 11/02/2025] [En ligne] <https://zod.dev/>
- [11] Flutter [Consulté le 13/02/2025] [En ligne] <https://docs.flutter.dev/>
- [12] Dio [Consulté le 13/02/2025] [En ligne] <https://pub.dev/packages/dio>
- [13] UI Material [Consulté le 13/02/2025] [En ligne] <https://docs.flutter.dev/ui/widgets/material>
- [14] Architecture monolithique [Consulté le 14/02/2025] [En ligne]
<https://www.talend.com/fr/resources/monolithic-architecture/>
- [15] Architecture microservices [Consulté le 14/02/2025] [En ligne]
<https://www.talend.com/fr/resources/guide-microservices/>
- [17] Domain-Driven Design [Consulté le 14/02/2025] [En ligne]
<https://lesdieuxducode.com/blog/2019/7/introduction-au-domain-driven-design>
- [18] Architecture monolithique modulaire [Consulté le 14/02/2025] [En ligne]
<https://www.solutions-numeriques.com/communiques/les-avantages-de-larchitecture-modulaire-informatique/>
- [18] Architecture Modular Monolith with Domain-Driven Design [Consulté le 14/02/2025]
[En ligne] <https://medium.com/@mounir.mouih/unlocking-the-power-of-modular-architecture-and-ddd-2b02a015f4fb>