

Remerciements

C'est avec un grand honneur et un grand plaisir que j'écris ces lignes afin de remercier toute personne ayant contribué de près ou de loin à l'accomplissement de mon projet de fin d'études. Je tiens tout d'abord à exprimer mes sincères respects envers mon encadrante, Madame Olfa Fakhfakh qui a accepté de me guider tout au long du projet. Grâce à ses conseils constructifs, sa disponibilité et ses encouragements, j'ai pu mener à bien mon travail.

Je tiens également à exprimer ma reconnaissance envers toute l'équipe de 3D Wave pour leur accueil chaleureux et les excellentes conditions de travail pendant mon stage.

Je tiens à exprimer ma profonde gratitude et ma reconnaissance respectueuse envers mon encadrant professionnel, Monsieur Houssem Eddine Mbarek pour le temps qu'il m'a accordé et la confiance qu'il m'a témoignée tout au long de mon stage.

Je saisirai cette occasion pour remercier l'ensemble du corps professoral de l'École Supérieure des Sciences et de la Technologie de Hammam Sousse, qui nous a formés avec diligence en partageant leurs connaissances et leur expertise.

Je remercie également les membres du jury d'avoir accepté d'évaluer mon travail, en espérant qu'ils trouveront dans ce rapport la motivation et la clarté qu'ils attendent.

Enfin, je tiens à exprimer ma reconnaissance envers toutes les autres personnes qui ont contribué de différentes manières au bon déroulement de mon projet de stage.

Dédicaces

Je dédie ce modeste travail en signe de reconnaissance et d'amour à :

À mes parents, pour leur soutien inconditionnel et leurs sacrifices consentis tout au long de mon parcours académique. Votre amour, vos encouragements et votre confiance en moi ont été les piliers de ma réussite.

À mes sœurs, pour leur soutien constant et leurs encouragements. Votre présence bienveillante dans ma vie a été une source de force et de motivation.

À mes amis proches, qui ont été là pour moi à chaque étape de ce parcours. Votre amitié précieuse et votre soutien moral ont illuminé mes journées et m'ont donné la force de persévérer.

À mes enseignants, dont les connaissances et les conseils ont façonné ma compréhension dans ce domaine d'étude. Votre dévouement à l'éducation et votre passion pour l'apprentissage m'ont inspiré à donner le meilleur de moi-même.

À toutes les personnes qui ont croisé mon chemin et m'ont apporté leur aide, leur encouragement et leur inspiration. Votre impact sur ma vie ne sera jamais oublié et je vous suis reconnaissant pour toutes les leçons que j'ai apprises à travers nos interactions.

Que cette dédicace soit un témoignage de ma gratitude envers chacune de ces personnes. Votre soutien et votre amour ont été inestimables, et je vous en serai éternellement reconnaissant.

Bien à vous,

AZIZ.

Table des matières

Introduction générale	1
1. Introduction	4
2. Présentation de l'organisme d'accueil.....	4
2.1. 3D Wave	4
2.2. Fiche descriptive	4
3. Présentation du projet.....	5
3.1. Problématique	5
3.2. Objectif du projet	6
3.3. Etude de l'existant.....	7
3.3.1. KelDoc.....	7
3.3.2. Dossier Médical Partagé	8
3.3.3. MEDASENSE	8
3.4. Critique de l'existant.....	9
3.5. Solution proposée.....	10
4. Le processus de développement adopté	11
4.1. Les méthodes traditionnelles.....	11
4.2. Les méthodes agiles	11
4.3. Choix de la méthodologie de travail	12
4.3.1. Présentation de Scrum	13
4.3.2. Les rôles Scrum	13
4.3.3. Les éléments Scrum.....	13

5.	Conclusion.....	15
6.	Introduction	16
7.	Spécification des besoins	16
7.1.	Identification des acteurs	16
7.2.	Besoins fonctionnels	17
7.3.	Les besoins non fonctionnels	19
7.4.	Le Backlog produit	19
8.	Analyse des besoins et conception	25
8.1.	Les diagrammes de cas d'utilisation	25
8.1.1.	Diagramme de cas d'utilisation « Administrateur ».....	25
8.1.2.	Diagramme de cas d'utilisation « Patient »	30
8.1.3.	Diagramme de cas d'utilisation « Médecin »	34
8.1.4.	Diagramme de cas d'utilisation « Agent de laboratoire »	37
8.1.5.	Diagramme de cas d'utilisation « Agent du centre d'imagerie médical ».....	40
9.	Conclusion.....	42
10.	Introduction	43
11.	Architecture de l'application	43
11.1.	Architecture MVC.....	43
12.	Conception détaillée	45
12.1.	Choix du langage de conception	45
12.2.	Diagramme de classes de l'application	46
12.3.	Les diagrammes de séquence	48

12.3.1.	Diagramme de séquence « authentification »	48
12.3.2.	Diagramme de séquence « gérer la liste de demande des fournisseurs de soins »	49
12.3.3.	Diagramme de séquence « gérer les demandes de suivi »	50
12.3.4.	Diagramme de séquence « ajouter d'un fichier médical ».....	51
12.3.5.	Diagramme de séquence « gérer un dossier médical »	52
12.3.6.	Diagramme de séquence « Messagerie »	53
13.	Conclusion	54
14.	Introduction	55
15.	Environnement de travail.....	55
15.1.	Technologies utilisées	55
15.1.1.	Node.js	55
15.1.2.	Flutter.....	56
15.1.3.	MongoDB	56
15.1.4.	Firebase	56
15.1.5.	Socket.IO	57
15.2.	Les outils utilisés :	57
15.2.1.	Visual Studio Code	57
15.2.2.	Postman.....	58
15.2.3.	GitHub.....	58
15.2.4.	StarUml :	58
15.2.5.	Git	59
15.2.6.	Ngrok	59

15.3.	Les langages utilisés.....	60
15.3.1.	HTML	60
15.3.2.	CSS	60
15.3.3.	JavaScript.....	60
15.3.4.	Dart	61
16.	Test et validation	61
16.1.	Sprint 1 : Authentification.....	62
16.1.1.	Interface de la page d'authentification (Patient, Médecin, agent de laboratoire, agent du centre médicale)	62
16.1.2.	Interface d'inscription (Patient, Fournisseurs de soins)	63
16.1.	Sprint 2 : Gestion des demandes d'inscription.....	67
16.2.	Sprint 3 : Gestion des demandes de suivi.....	69
16.2.1.	Gestion des demandes de suivi :	69
16.3.	Sprint 4 : Création des conversations en temps réel.....	72
16.4.	Sprint 5 : Gestion des dossiers médicaux.....	74
16.5.	Sprint 6 : Paiement en ligne d'un rendez-vous.....	77
	Après que le patient consulte le profil de son médecin, il peut prendre un rendez-vous et le payer en ligne voir figure 48 :	77
17.	Conclusion	81
	Conclusion et perspectives	83
	Références bibliographiques.....	85

Liste des tableaux

Table 1:cordonnées de l'organisme d'accueil	5
Table 2:Tableau comparatif des solutions existantes	9
Table 3:Tableau récapitulatif des sprints.....	15
Table 4:Le Backlog de l'application	24
Table 5:Tableau descriptif du cas d'utilisation « Administrateur »	30
Table 6:bleau descriptif du cas d'utilisation «Patient».....	34
Table 7:Tableau descriptif du cas d'utilisation « Médecin ».....	37
Table 8:Tableau descriptif du cas d'utilisation « Agent du laboratoire »	40
Table 9:Tableau descriptif du cas d'utilisation « Agent du centre d'imagerie médical » ...	42

Liste des figures

Figure 1: Logo 3D Wave	5
Figure 2: Logo de KelDoc	7
Figure 3: Logo de DMP	8
Figure 4: LOGO de MEDASENSE	9
Figure 5: Comparaison entre la méthode agile et la méthode classique.....	12
Figure 6: Le processus Scrum	14
Figure 7: Diagramme de cas d'utilisation de l'acteur « Administrateur »	26
Figure 8: Diagramme de cas d'utilisation de l'acteur « Patient ».	31
Figure 9: Diagramme de cas d'utilisation de l'acteur « Médecin ».....	35
Figure 10: Diagramme de cas d'utilisation de l'acteur «Agent de laboratoire ».....	38
Figure 11: Diagramme de cas d'utilisation de l'acteur «Agent du centre d'imagerie médical»	41
Figure 12 : Les composants MVC.....	43
Figure 13: Schéma de l'architecture MVC.....	44
Figure 14: Les différentes vues de UML.....	45
Figure 15: Diagramme de classes de l'application	48
Figure 16: Diagramme de séquence d'authentification	49
Figure 17: Diagramme de séquence de gérer la liste de demande des fournisseurs de soin	50
Figure 18: Diagramme de séquence gérer les demandes de suivi	51
Figure 19: Diagramme de séquence ajouter d'un fichier médical	52
Figure 20: Diagramme de séquence de gérer un dossier médical	53

Figure 21: Diagramme de séquence de Messagrie	54
Figure 22: Logo de Node js	55
Figure 23: Logo de Flutter	56
Figure 24: Logo de Monogodb	56
Figure 25: Logo Firebase.....	57
Figure 26: Logo Socket.IO	57
Figure 27: Logo Visual Studio Code	57
Figure 28: Logo de Postman.....	58
Figure 29: Logo de GitHub	58
Figure 30: logo de StarUML	59
Figure 31: Logo de git	59
Figure 32;logo de ngrok	59
Figure 33: Logo HTML.....	60
Figure 34: Logo CSS	60
Figure 35: Logo JavaScript.....	61
Figure 38: Interface d'inscription.....	65
Figure 39: Interface d'inscription.....	66
Figure 40: Interface d'inscription.....	67
Figure 41: Interface d'inscription.....	68
Figure 42: Interface de modification d'un utilisateur.....	69
Figure 43: Interface de chercher un fournisseur de soin	70
Figure 44: Interface de gestion des invitations du patient	71

Figure 45: Interface de gestion des demandes de suivi coté fournisseur	72
Figure 46: Interface de création des conversations en temps réel	74
Figure 47: Gestion des dossiers médicales	77
Figure 48: Gestion de paiement en ligne	81

Introduction générale

Le suivi régulier de l'état de santé est essentiel pour détecter les problèmes précocement, prévenir les maladies, améliorer la qualité de vie et garantir une prise en charge proactive. Cependant, malgré les avantages de la digitalisation dans notre monde moderne, le suivi régulier de l'état de santé rencontre plusieurs défis.

En effet, la perte de documents médicaux constitue un problème majeur, entraînant des retards et des erreurs dans l'évaluation de l'état de santé des patients. De plus, le manque d'automatisation des processus médicaux quotidiens provoque des inefficacités, des retards et des erreurs dans les opérations médicales, compromettant ainsi la qualité des soins. Un autre défi important est le coût élevé des analyses médicales et la redondance des examens déjà effectués lorsqu'un patient change de médecin, ce qui entraîne des dépenses supplémentaires et un gaspillage de ressources. Il est essentiel de mettre en place des systèmes de gestion efficaces et d'accéder facilement aux dossiers médicaux afin de résoudre ces problèmes et d'améliorer la qualité et l'efficacité des soins de santé.

Dans ce contexte, notre projet de fin d'études, réalisé au sein de l'entreprise 3D WAVE, s'inscrit dans le cadre de l'obtention du diplôme de la Licence Nationale en Sciences de l'informatique, spécialité Génie logiciel et systèmes d'information à l'École Supérieure des Sciences et de la Technologie de Hammam Sousse. L'objectif de notre projet est de concevoir et mettre en œuvre une application mobile médicale pour suivre l'état de santé des patients qui permettra de surmonter ces difficultés et de faciliter les échanges entre les patients et leurs fournisseurs de soin en toute sécurité.

Notre application vise à offrir une solution de suivi de l'état médical des patients, conçue pour répondre aux défis de l'automatisation des processus médicaux quotidiens. Son objectif est de faciliter le suivi et l'échange des documents médicaux entre les patients et les professionnels de la santé, afin de prévenir la perte de ces documents et d'optimiser les soins

médicaux. Les utilisateurs pourront accéder facilement aux informations médicales actualisées et prendre des décisions éclairées.

Dans ce rapport, nous examinerons les étapes de conception et de mise en œuvre de notre application, en mettant l'accent sur les décisions technologiques prises. Nous mettrons en évidence les fonctionnalités principales de l'application ainsi que les avantages qu'elle apporte aux utilisateurs. En outre, nous discuterons des résultats obtenus, des difficultés rencontrées et des possibilités d'amélioration à l'avenir.

Ce rapport a pour but de fournir une vue claire du travail réalisé lors de la réalisation du projet. Il est structuré en quatre chapitres.

Le premier chapitre de ce rapport offre une présentation de l'organisation hôte ainsi que du contexte global du sujet, en mettant en évidence la problématique à résoudre. De plus, il examine l'étude préalable de l'existant et expose la méthodologie de travail qui a été adoptée.

Le deuxième chapitre se focalise sur l'analyse et la spécification des besoins. Nous examinons de manière détaillée les différents cas d'utilisation propres à chaque acteur, ce qui nous permet d'acquérir une meilleure compréhension des interactions entre les utilisateurs et le système.

Dans le troisième chapitre, nous abordons l'architecture du projet, ainsi que les aspects conceptuels à travers les diagrammes de classes et les diagrammes de séquence. Nous détaillons la manière dont les différentes composantes du système sont organisées et interagissent entre elles.

Enfin, en ce qui concerne le quatrième chapitre, il traite des outils et des technologies employés tout au long du projet, ainsi que de la phase de test et de validation. De plus, ce chapitre offre un aperçu des résultats obtenus à la fin de chaque itération de développement.

En guise de conclusion, ce rapport récapitule l'ensemble des étapes suivies pour mener à bien ce projet. Il offre un résumé complet du processus de développement, mettant en évidence les

différentes phases telles que la conception, l'implémentation et les tests. De plus, il souligne les perspectives d'amélioration envisagées pour les futures versions de l'application, afin de répondre aux besoins croissants des utilisateurs et de fournir une expérience utilisateur encore plus satisfaisante.

Cadre général du stage

1. Introduction

Dans ce premier chapitre, nous présentons le contexte général de notre projet, y compris le cadre et l'organisation d'accueil. Nous exposons les objectifs de notre projet, ainsi qu'analysons de manière critique les solutions existantes pour répondre à la problématique. Enfin, nous proposons notre propre solution, en nous basant sur les résultats de notre recherche et en prenant en compte les critiques formulées à l'encontre des solutions existantes.

2. Présentation de l'organisme d'accueil

Dans cette partie, nous présentons 3D Wave, l'organisme qui nous a accueillis, et donnons un aperçu de son secteur d'activité.

2.1. 3D Wave

3D Wave est une entreprise tunisienne basée à Sousse et fondée en 2018. Elle se compose d'une équipe jeune, dynamique et expérimentée combinant différentes compétences. Elle se spécialise de la technologie de la numérisation 3D et le développement de réalité augmentée. 3D wave offre une large gamme de services pour l'industrie de la numérisation 3D, du contrôle dimensionnel à l'impression 3D de prototypes et de petites séries. Elle propose également des applications industrielles de réalité augmentée personnalisées [1].

2.2. Fiche descriptive

La fiche descriptive du 3D Wave se présente comme suit :



Figure 1: Logo 3D Wave

Adresse	Khezama, Sousse
Téléphone	+216 28899806
E-mail	contact@3dwave.tech
Site Web	https://www.3dwave.tech/
Secteur d'activité	Informatique / télécoms

Table 1:cordonnées de l'organisme d'accueil

3. Présentation du projet

3.1. Problématique

Dans le domaine de la santé, les patients et les médecins font face à divers problèmes et difficultés. Afin d'améliorer la qualité et l'efficacité des soins de santé, il est essentiel de considérer la nécessité d'automatiser les processus médicaux quotidiens. Cette automatisation permettrait de réduire les erreurs humaines et d'optimiser le temps consacré aux traitements et aux évaluations médicales.

De plus, la perte de documents médicaux tels que les résultats d'examens sanguins, les IRM ou les scanners, est une préoccupation majeure dans le domaine de la santé, car cela peut entraîner des retards et des erreurs dans l'évaluation de l'état de santé des patients. Il est donc primordial de mettre en place des systèmes de suivi efficaces afin de prévenir ces perturbations.

En outre, un autre problème important dans le domaine de la santé est le coût élevé des analyses médicales. De plus, lorsque les patients changent de médecin, le nouveau praticien demande souvent les mêmes analyses ou bilans déjà réalisés, ce qui oblige le patient à les refaire. Cela entraîne des dépenses supplémentaires et un gaspillage de ressources. Afin de résoudre ce problème, il est essentiel de garantir que le patient ait toujours accès à son dossier médical, ce qui lui permettrait de partager facilement les résultats d'analyses précédentes avec les professionnels de la santé, réduisant ainsi les redondances et les coûts inutiles.

Enfin, il est essentiel d'avoir un suivi en temps réel de l'état de santé des patients afin de garantir des prises de décision éclairées basées sur des informations médicales à jour. En tenant compte de ces exigences, il est possible d'améliorer l'efficacité et la productivité des prestataires de soins de santé tout en minimisant les risques associés aux problèmes courants de santé.

3.2. Objectif du projet

3D Wave nous a confié la mission de concevoir et développer une application mobile médicale pour suivre l'état de santé des patients. Cette application agira comme une interface efficace entre les patients, les médecins, les laboratoires et les centres d'imagerie médicale, dans le but d'améliorer leur expérience et de simplifier les processus pour toutes les parties concernées. Notre objectif est de résoudre les problèmes identifiés en créant une solution qui offrira une expérience utilisateur satisfaisante pour tous les acteurs impliqués.

3.3. Etude de l'existant

La phase d'étude de l'existant est une étape fondamentale dans la réalisation d'un projet, car elle permet de comprendre et d'analyser les solutions existantes afin d'identifier leurs forces et leurs faiblesses, et de déterminer les objectifs et le contexte d'application du projet. Cette phase se compose généralement de trois parties : l'analyse de l'existant, la critique de l'existant et la proposition de solutions. Dans ce cadre, nous avons opté pour prendre en exemple les solutions existantes les plus répandues au niveau international telles que "KelDoc", "DMP" et "Medasense".

3.3.1. KelDoc

KelDoc est une plateforme en ligne qui permet aux patients de prendre rendez-vous, offre aux patients plusieurs fonctionnalités pour faciliter la prise de rendez-vous. Voici un aperçu des principales :

- **Prise de rendez-vous en ligne :** Les patients peuvent prendre rendez-vous en ligne avec le professionnel de santé de leur choix en quelques clics.

- **Gestion des rendez-vous :** Les patients peuvent consulter et gérer leurs rendez-vous à tout moment depuis leur compte Keldoc.

La figure 3 ci-dessous montre le logo de Keldoc :



Figure 2: Logo de KelDoc

3.3.2. Dossier Médical Partagé

- Dossier Médical Partagé (DMP) est une plateforme qui permet aux patients de stocker et de partager leur dossier médical avec les professionnels de santé de leur choix. Voici un aperçu des principales :
- Stockage de données de santé : Le DMP permet de stocker toutes les informations de santé d'un patient, telles que les résultats d'examens médicaux, les antécédents médicaux.
- Accès aux données de santé : Les patients peuvent accéder à leurs données de santé à tout moment via l'interface en ligne du DMP.

La figure ci-dessous montre le logo de DMP :



Figure 3: Logo de DMP

3.3.3. MEDASENSE

MEDASENSE est une plateforme qui permet aux patients de consulter leurs résultats de laboratoire en temps réel. Voici un aperçu des principales :

- Accès en temps réel : Les patients peuvent accéder à leurs résultats d'examens médicaux en temps réel via l'interface en ligne de MEDASENSE.
- Stockage de résultats d'examens médicaux : Medasense permet de stocker les résultats d'examens médicaux de chaque patient de manière sécurisée sur ses serveurs.

La figure ci-dessous montre le logo de MEDASENSE :



Figure 4: LOGO de MEDASENSE

3.4. Critique de l'existant

Bien que ces solutions offrent des services aux utilisateurs et présentent des caractéristiques avantageuses, elles demeurent sujettes à certaines vulnérabilités.

Le tableau 2 suivant présente une comparaison entre les solutions mentionnées précédemment en fonction de différents critères :

Application	Keldoc	Dossier Médical Partagé	Medasense	Notre solution
Prise de rendez-vous en ligne	+	-	-	+
Stockage de données de santé	-	+	-	+
Accès aux données de santé	+	+	+	+
Envoie des documents médical en temps réel	-	-	+	+
Messagerie sécurisée en temps réel	+	+	+	+

Table 2:Tableau comparatif des solutions existantes

Outre les critères énumérés dans le tableau n°2 ci-dessus, la plupart des solutions actuellement disponibles présentent des lacunes.

En effet, il n'existe pas de logiciel qui offre l'ensemble des fonctionnalités en même temps, et aucun logiciel développé à l'échelle nationale n'interagit avec tous les professionnels de santé, les laboratoires et les centres d'imagerie médicale à la fois.

Ainsi, pour répondre aux besoins des utilisateurs de manière optimale, notre solution devra prendre en compte tous ces critères tout en combinant les points forts des différentes solutions existantes.

3.5. Solution proposée

Après avoir étudié les solutions existantes, nous avons eu l'idée de créer une application mobile médicale de suivi de l'état de santé des patients dédiée aux patients et à leurs médecins. Notre objectif est de faciliter le traitement et l'échange des documents médicaux entre les deux parties, afin d'éviter la perte de ces documents. De plus, cette application permettra aux patients de consulter leurs dossiers médicaux même s'ils changent de médecin. Ainsi, ils pourront continuer à accéder à leurs historiques médicaux, incluant les traitements antérieurs, les résultats d'examens et les diagnostics posés par leurs anciens médecins.

En outre, les patients pourront recevoir en temps réel leurs documents médicaux provenant des laboratoires ou des centres d'imagerie médicale. Cela leur permettra d'avoir un accès immédiat aux résultats de leurs analyses, ce qui favorisera une gestion plus rapide et efficace de leurs informations médicales. Par ailleurs, les patients auront la possibilité d'envoyer en temps réel leurs propres documents médicaux à leur médecin, que ce soit des résultats d'analyses, d'imagerie ou des informations quotidiennes telles que la mesure de la pression artérielle. Cette fonctionnalité facilitera la communication fluide et la coordination des soins entre les patients et leurs médecins.

Enfin, notre application offrira la possibilité aux patients et à leurs médecins d'avoir des conversations en temps réel via un chat. Cette fonctionnalité sera particulièrement utile en cas d'urgence ou lorsque des renseignements supplémentaires seront nécessaires. Elle permettra une communication instantanée et directe, offrant ainsi une assistance rapide et des échanges d'informations précieuses.

En résumé, notre application représente une avancée majeure dans le suivi et l'échange des dossiers et des informations médicaux. Elle offre une solution complète et automatisée pour faciliter le traitement, l'échange et le suivi en temps réel de ces documents, favorisant ainsi une meilleure coordination des soins médicaux.

4. Le processus de développement adopté

Afin de favoriser une collaboration efficace entre les différents acteurs d'un projet, il est essentiel d'opter pour une méthode de gestion de projet qui offre des règles claires et définies. Plusieurs modèles peuvent être employés dans le processus de développement d'un logiciel. Les deux grandes familles de méthodes de gestion de projet à prendre en considération sont :

4.1. Les méthodes traditionnelles

Les méthodes traditionnelles de gestion de projet suivent un processus séquentiel et linéaire, où chaque étape doit être validée avant de passer à la suivante. Ces approches utilisent une documentation détaillée approuvée dès le début du projet, sans interaction directe avec le client. Cependant, elles ne prennent pas en compte les changements imprévus qui peuvent survenir, entraînant des retards, des coûts supplémentaires et la déception du client. Dans ces approches, la réussite dès la première tentative est essentielle, laissant peu de place pour les ajustements.

4.2. Les méthodes agiles

Les méthodes agiles de gestion de projet sont plus souples et efficaces que les méthodes traditionnelles, car elles utilisent un principe de développement itératif. Ce principe implique de découper le projet en plusieurs étapes appelées itérations, où chaque itération représente un mini-projet avec des fonctionnalités définies en collaboration avec le client en fonction de leur priorité. Le chef de projet établit ensuite un macro-planning correspondant aux tâches nécessaires pour développer ces fonctionnalités. Cette approche

reconnaît que, malgré notre expérience, nous ne pouvons pas tout anticiper et prévoir. Ainsi, plutôt que de tout planifier dès le début, le projet est découpé en itérations pour permettre une plus grande flexibilité face aux imprévus susceptibles de survenir en cours de route [2].

La figure 5 présente une comparaison récapitulative entre les approches traditionnelles et agiles utilisées en gestion de projet :

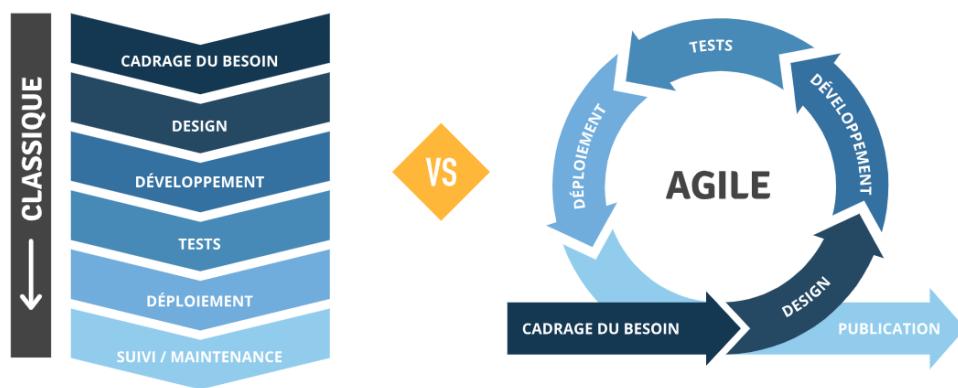


Figure 5: Comparaison entre la méthode agile et la méthode classique

4.3. Choix de la méthodologie de travail

Après avoir effectué une comparaison entre plusieurs méthodologies, nous avons opté pour Scrum en tant que méthode de gestion de projet en raison de ses avantages suivants:

- Sa grande flexibilité.
- Sa capacité d'adaptation au changement grâce à des itérations courtes.
- Le fait que Scrum combine les aspects théoriques et pratiques, ce qui le rend plus proche de la réalité.
- Sa capacité à aligner les objectifs individuels avec ceux de l'entreprise.

4.3.1. Présentation de Scrum

Scrum est une méthode de gestion de projet Agile qui permet de produire des résultats de haute qualité tout en livrant de manière productive et créative des produits de la plus grande valeur possible.

4.3.2. Les rôles Scrum

Les rôles SCRUM définit uniquement trois rôles, qui sont comme suit :

- **Le Product Owner :** C'est le propriétaire du produit gère le Product Backlog (carnet du produit), représente les besoins des utilisateurs et prend la décision de déploiement.
- **Le Scrum Master :** C'est le responsable des événements Scrum anime les réunions quotidiennes, organise la planification, l'analyse et la rétrospective de sprint, jouant ainsi un rôle de chef de projet Scrum.
- **L'équipe Scrum :** C'est L'équipe de développement, composée de 2 à 10 personnes, est autoorganisée et a pour rôle principal de réaliser les user stories du sprint Backlog afin d'atteindre les objectifs définis lors de la planification des sprints.

4.3.3. Les éléments Scrum

- **Backlog produit (Catalogue des besoins) :** c'est la liste principale des tâches à réaliser, elle décrit tout le projet, gérée par le Product Owner ou le Scrum Master. C'est une liste dynamique de fonctionnalités, d'exigences, d'améliorations et de corrections qui sert de point de départ pour le Backlog de sprint (défini ci-dessous). C'est la « to-do list » de l'équipe.
- **Backlog de sprint :** c'est une sélection de tâches retenues du « Product Backlog » pour construire l'objectif du sprint. (C'est un fragment du Backlog Produit).
- **Sprint (itération) :** c'est le cœur de Scrum, une durée de 2 à 4 semaines au cours de laquelle des fonctionnalités du Backlog de sprint sont développées et un incrément

produit fonctionnel est créé. Un nouveau Sprint commence immédiatement après la conclusion du Sprint précédent.

- **User story** : c'est la plus petite unité de travail. C'est un objectif final, pas une fonctionnalité, exprimé du point de vue de l'utilisateur du logiciel. Une histoire utilisateur est une explication générale et informelle d'une fonctionnalité logicielle écrite du point de vue de l'utilisateur final ou du client.
- **Daily Meeting** : c'est un point quotidien qui permet de mettre le point sur ce qui a été réalisé, les problèmes rencontrés et les objectifs de la journée [3].

Afin de mieux comprendre le déroulement d'un projet avec SCRUM, nous avons choisi La figure ci-dessous :

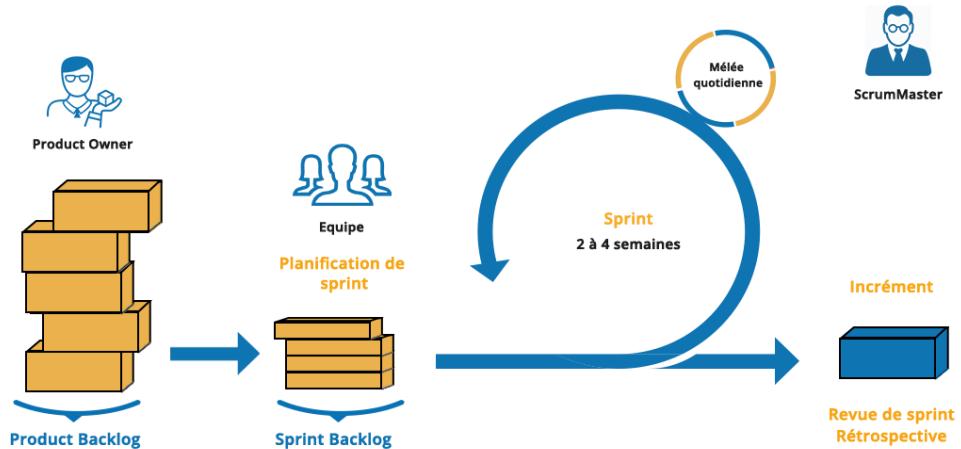


Figure 6: Le processus Scrum

Conformément à ce qui a été mentionné précédemment et pour le bon déroulement du projet, le travail est divisé en sprints qui seront mieux expliqués au niveau du chapitre 4 dans la partie "tests et validation". Les sprints de notre projet ont été divisés et planifiés comme le montre le tableau 2 ci-dessous :

Sprint	Description
--------	-------------

Sprint 0	Spécification des besoins et conception.
Sprint 1	Authentification
Sprint 2	Gestion des demandes d'inscription
Sprint 3	Gestion des demandes de suivi
Sprint 4	Création des conversations en temps réel
Sprint 5	Gestion des dossiers médicaux
Sprint 6	Gestion des rendez-vous
Sprint 7	Test et validation

Table 3:Tableau récapitulatif des sprints

5. Conclusion

Dans le premier chapitre, nous avons introduit notre étude en décrivant le contexte global de notre application et en soulignant le problème à résoudre. Ensuite, nous avons réalisé une recherche approfondie sur les solutions existantes. Par la suite, nous avons présenté notre solution et expliqué la méthodologie que nous avons choisie.

Enfin, nous avons détaillé les différentes étapes de développement regroupées sous forme de "sprints". Dans le prochain chapitre, nous aborderons l'analyse et la spécification des besoins, où nous examinerons en détail les exigences fonctionnelles et non fonctionnelles de notre projet, ainsi que le Backlog produit.

Analyse et spécification des besoins

6. Introduction

Dans cette section, nous abordons la première phase du projet réalisée pendant le stage. Nous détaillons les exigences fonctionnelles et non fonctionnelles de notre application, ainsi que le Backlog produit. Nous identifions également tous les acteurs du système. Ensuite, nous présentons les diagrammes des cas d'utilisation pour chaque acteur afin d'illustrer leur interaction avec le système.

7. Spécification des besoins

Dans cette partie, nous déterminons les acteurs et les besoins de notre application.

7.1. Identification des acteurs

Les utilisateurs suivants seront impliqués dans l'utilisation de notre application :

- **Administrateur** : L'administrateur est responsable de la gestion de toutes les entités du système. Par ailleurs, il doit s'assurer du bon fonctionnement de l'application.
- **Patient** : Le patient c'est celui qui insère et reçoit ses fichiers médicaux sur la plateforme afin de les stocker et les envoyer à son médecin. De plus, il peut ajouter des informations et des valeurs à des métriques médicales disponibles en temps réel. Il peut aussi contacter en temps réel ses médecins, les laboratoires, les centres d'imagerie médicale à travers un chat.
- **Médecin** : le médecin a comme rôle principal de consulter, suivre, modifier et gérer les dossiers médicaux de ses différents patients.
- **Agent de Laboratoire** : le laboratoire c'est celui qui envoie les résultats des analyses à ses patients en temps réel.
- **Agent de Centre d'imagerie médicale** : le centre d'imagerie médicale c'est celui qui envoie les résultats des images médicales à ses patients en temps réel.

7.2. Besoins fonctionnels

Les besoins fonctionnels de notre projet reflètent les attentes de chaque acteur de l'application.

- **Administrateur**

- L'administrateur a le pouvoir d'accepter ou de refuser l'inscription des médecins, des laboratoires ou même des centres d'imagerie médicale. Pour ce faire, il peut vérifier les licences médicales afin de s'assurer de l'authenticité des laboratoires, des médecins et des centres d'imagerie qui souhaitent s'inscrire sur l'application.
- Consulter la liste de tous les utilisateurs.
- Consulter la liste des spécialités des médecins et avoir la possibilité d'ajouter, modifier ou supprimer une spécialité.
- Consulter la liste des métriques de santé et avoir la possibilité d'ajouter, modifier ou supprimer une métrique.
- S'authentifier.

- **Patient**

- S'inscrire.
- S'authentifier.
- Chercher des fournisseurs de soins (médecins, laboratoires et centres d'imagerie médicale) et avoir la possibilité d'envoyer une demande de suivi à un fournisseur choisi.
- Ajouter des fichiers médicaux.
- Ajouter des informations et des valeurs à ses métriques médicales tels que la pression artérielle, ...
- Consulter la liste de ses fournisseurs de soins : la liste des médecins, la liste des laboratoires et la liste des centres d'imagerie médicale.

- Prise d'un rendez-vous avec le médecin en ligne et avoir la possibilité de le payer en ligne.
- Contacter en temps réel tous ses fournisseurs de soin.

- **Médecin**

- S'inscrire
- S'authentifier
- Consulter la liste de ses patients et avoir l'option de consulter les détails d'un patient choisi.
- Consulter rendez-vous prises par ses patients et avoir la possibilité d'accepter, de refuser ou même de changer la date du rendez-vous.
- Consulter le dossier médical d'un patient.
- Contacter ses patients en temps réel.
- Consulter la liste des métriques médicales et avoir la possibilité d'ajouter une métrique.

- **Agent de Laboratoire**

- S'inscrire.
- S'authentifier.
- Consulter la liste de ses patients.
- Ajouter un résultat d'analyse à un patient choisi.
- Contacter les patients en temps réel.

- **Agent de Centre d'imagerie médicale**

- S'inscrire
- S'authentifier
- Consulter la liste des patients.

- Ajouter un résultat d'image médicale à un patient choisi.
- Contacter les patients en temps réel.

7.3. Les besoins non fonctionnels

Pour améliorer la qualité de service de notre application, il est crucial de répondre aux exigences non fonctionnelles suivantes :

- **Convivialité** : Il est important que l'application soit conçue de manière à être facilement utilisable. Cela implique que les interfaces utilisateur doivent être conviviales, c'est-à-dire simples, ergonomiques et adaptées aux utilisateurs.
- **Sécurité** : L'application doit s'assurer que les données de l'utilisateur sont maintenues en toute sécurité, en préservant leur intégrité et leur confidentialité.
- **Fiabilité** : Notre application doit fonctionner de façon cohérente sans erreurs.

7.4. Le Backlog produit

Le tableau ci-dessous présente le Backlog produit. Il présente un ensemble de «user stories» qui décrivent les fonctionnalités du système.

ID	Fonctionnalité	Acteur	Scénario ou user story	Priorité
1	Inscription	Patient /Médecin / Agent de centre d'imagerie médicale / Agent de laboratoire	En tant que non abonné, je veux m'inscrire.	Must
2	Authentification	Patient /Médecin /	En tant qu'abonné, je veux m'authentifier	Must

		Agent de centre d'imagerie médicale / Agent de laboratoire Administrateur		
3	Gérer les demandes d'inscription	Administrateur	En tant qu'administrateur, je veux gérer les demandes d'inscription en les confirmant ou en les refusant, dans le but de garantir la qualité et la sécurité du service proposé.	Must
4	Chercher fournisseur de soin	Administrateur/ Patient	En tant qu'administrateur, je veux chercher un fournisseur de soin pour accéder à son profil et obtenir des informations telles que son nom, sa spécialité, ses coordonnées de contact.	Must
			En tant que patient, je veux chercher un fournisseur de soin afin d'avoir accès à son profil complet, y compris ses informations	
5	Chercher patient	Médecin/ Agent de laboratoire/ Agent de centre d'imagerie médicale	En tant que médecin, je veux chercher un patient afin de consulter son profil et prendre des informations.	Must
			En tant que laboratoire, je veux chercher un patient afin de consulter son profil et ajouter des informations.	

			En tant que centre d'imagerie médicale, je veux chercher un patient afin de consulter son profil et ajouter des informations.	
6	Répondre aux demandes de suivi	Médecin/Agent de laboratoire/ Agent de centre d'imagerie médicale	En tant que médecin, je veux être en mesure de refuser ou d'accepter une demande de suivi de la part d'un patient	Must
			En tant que laboratoire, je veux être en mesure de refuser ou d'accepter une demande de suivi de la part d'un patient	
			En tant que centre d'imagerie médicale, je veux être en mesure de refuser ou d'accepter une demande de suivi de la part d'un patient	
7	Consulter dossier médicale du patient	Médecin	En tant que médecin, je veux être en mesure de consulter le dossier médical de mes patients pour prendre plus d'information médical	Must
8	Ajouter des médicaments et des chirurgies	Médecin	En tant que médecin, je veux être en mesure d'ajouter des médicaments et chirurgie à mes patients.	Should
9	Ajouter des analyses	Agent de laboratoire	En tant que laboratoire, je veux être en mesure d'ajouter des analyses à mes patients.	Should

10	Ajouter des radiographies	Agent de centre d'imagerie médicale	En tant que Centre d'imagerie médicale, je veux être en mesure d'ajouter des radiographies à mes patients.	Should
11	Gérer les informations médicales	Patient	En tant que patient, je veux gérer mes informations médicales pour construire et maintenir mon dossier médical à jour. Je veux être en mesure d'ajouter et de supprimer mes analyses médicales, mes allergies et tout autre détail pertinent pour mon suivi médical	Must
12	Mettre à jour les informations du compte	Patient /Médecin / Agent de centre d'imagerie médical / Agent de laboratoire	En tant que médecin, je veux mettre à jour mes informations (comme nom, prénom, adresse, numéro de téléphone, adresse email, spécialité, etc.).	Must
			En tant que patient, je veux mettre à jour mes informations personnelles (telles que nom, prénom, adresse, numéro de téléphone, adresse email, etc.).	
			En tant que laboratoire, je veux mettre à jour mes informations personnelles (telles que nom, adresse, numéro de téléphone, adresse email, etc.).	

			En tant que laboratoire, je veux mettre à jour mes informations personnelles (telles que nom, adresse, numéro de téléphone, adresse email, etc.).	
13	Gérer les demandes de suivi	Patient	En tant que patient, je veux gérer demandes de suivi afin d'obtenir un suivi adéquat et personnalisé.	Must
14	Gérer la Messagerie instantanée	Patient /Médecin / Agent de centre d'imagerie médical / Agent de laboratoire	En tant que patient, je veux communiquer avec mes fournisseurs de soin en temps réel à travers des messages texte, afin de poser des questions, signaler des changements dans ma santé ou simplement avoir des informations complémentaires sur mon traitement.	Would
			En tant que médecin, je veux communiquer en temps réel avec mon patient à travers des messages textuels, afin de pouvoir répondre rapidement à ses questions.	
			En tant que Centre d'imagerie médical, je veux communiquer en temps réel avec mon patient à travers des messages	

			textuels, afin de pouvoir répondre rapidement à ses questions.	
			En tant que laboratoire, je veux communiquer en temps réel avec mon patient à travers des messages textuels, afin de pouvoir répondre rapidement à ses questions.	
15	Prendre un rendez-vous	Patient	En tant que patient, je veux prendre des rendez-vous.	Would
16	Gérer les utilisateurs	Administrateur	En tant qu'administrateur, je veux gérer les utilisateurs inscrits dans la plateforme	Would

Table 4:Le Backlog de l'application

Avec :

- ✓ **ID** : représente l'identifiant de la "user story".
- ✓ **Scenario ou User Story** : comporte la description des user story suivant la forme "**En tant que ... Je veux ...**".
- ✓ **La priorité** de la "user story" est définie selon l'importance.

8. Analyse des besoins et conception

Après avoir précisé les acteurs et les exigences requises pour cette application, l'étape suivante consiste à modéliser les différentes fonctions à l'aide des diagrammes de cas d'utilisation.

8.1. Les diagrammes de cas d'utilisation

Un diagramme de cas d'utilisation représente une fonctionnalité spécifique dans un système et est créé pour illustrer comment différentes fonctionnalités sont interconnectées et montrer leurs contrôleurs (ou acteurs) internes et externes.

Il modélise le comportement d'un système, permette de capturer les exigences du système et décrit les fonctions générales et la portée d'un système.

Pour assurer une représentation complète de toutes les fonctionnalités requises, nous allons commencer par présenter le diagramme de cas d'utilisation global qui inclut tous les acteurs, avant de passer aux cas d'utilisation détaillés [4].

8.1.1. Diagramme de cas d'utilisation « Administrateur »

La figure 7 représente le diagramme de cas d'utilisation de l'acteur "Administrateur" :

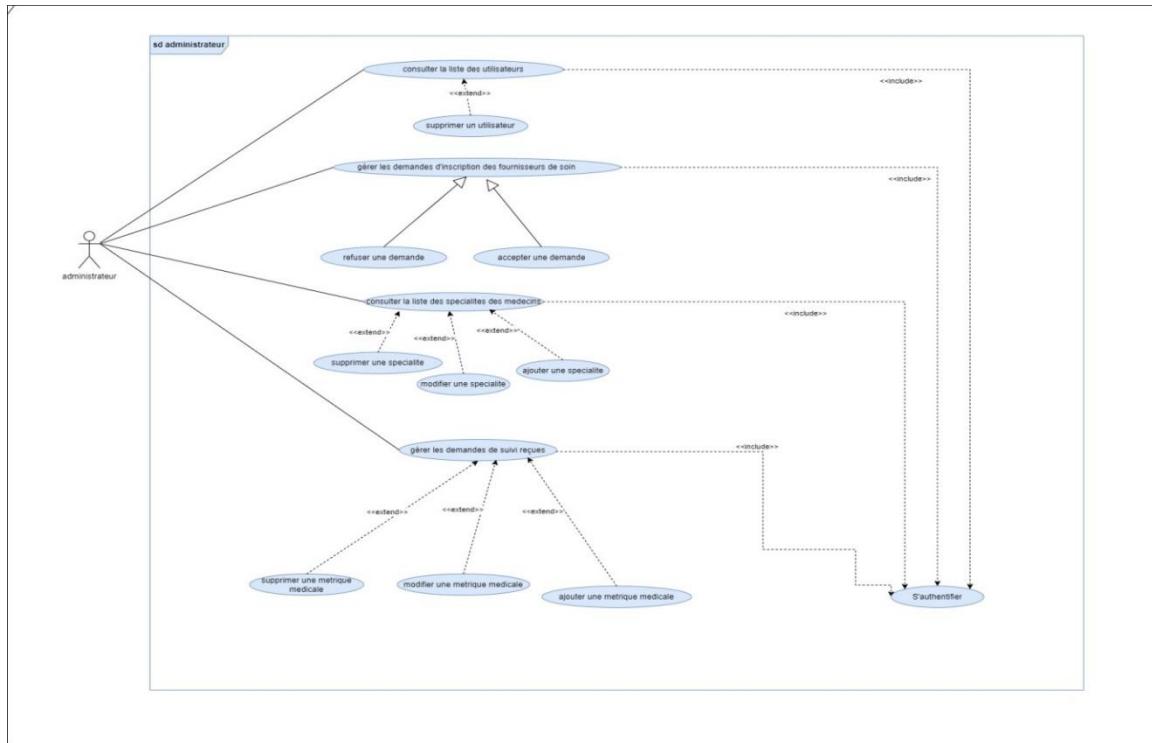


Figure 7: Diagramme de cas d'utilisation de l'acteur « Administrateur »

Le tableau 5 ci-dessous, décrit en détails les différents cas d'utilisation de la figure 7 de l'administrateur :

Acteur	Administrateur
Cas d'utilisation 1	Consulter la liste des utilisateurs
Préconditions	<ul style="list-style-type: none"> L'administrateur doit être authentifié. L'administrateur doit consulter la liste des comptes utilisateurs (Accéder à la page où un tableau de données est affiché, ayant toutes les coordonnées des utilisateurs existants, à partir de son menu).
Postconditions	<ul style="list-style-type: none"> Utilisateur supprimé

Scenario principal	Après l'authentification et la consultation de la liste des utilisateurs, l'administrateur aura le choix de :
	<ul style="list-style-type: none"> • Supprimer un compte utilisateur : l'administrateur clique sur le bouton de suppression ; le système affiche utilisateur supprimé
Scenario d'exception	
Cas d'utilisation 2 :	Gérer les demandes d'inscription des fournisseurs de soin
Préconditions	<ul style="list-style-type: none"> • L'administrateur doit être authentifié. • L'administrateur doit consulter la liste des comptes des fournisseurs de soin en attente (Accéder à la page où un tableau de données est affiché, ayant toutes les coordonnées des fournisseurs en attentes, à partir de son menu)
Postconditions	<ul style="list-style-type: none"> • Fournisseur de soin refusé • Fournisseur de soin approuvé
Scenario principal	Après l'authentification et la consultation de la liste des comptes des fournisseurs de soin en attente, l'administrateur aura le choix de :
	<ul style="list-style-type: none"> • Approuver un compte fournisseur de soin : l'administrateur vérifie la clé de licence médicale du fournisseur après il approuve le fournisseur ; le système affiche fournisseur de soin ajoutée, et le fournisseur de soin recevoir une notification d'acceptation qu'il puisse se connecter à son compte • Refuser un compte fournisseur de soin : après la vérification de la clé de licence médicale du fournisseur, l'administrateur refuse le fournisseur ; le système affiche fournisseur de soin refusé,

Cas d'utilisation 3 :	Consulter la liste des spécialités des médecins
Préconditions	<ul style="list-style-type: none">• L'administrateur doit être authentifié.• L'administrateur doit consulter la liste des spécialités des médecins (Accéder à la page où un tableau de données est affiché, ayant toutes les spécialités des médecins existants, à partir de son menu).
Postconditions	<ul style="list-style-type: none">• Spécialité ajoutée• Spécialité modifiée• Spécialité supprimée
Scenario principal	<p>Après l'authentification et la consultation de la liste des spécialités des médecins, l'administrateur aura le choix de :</p> <ul style="list-style-type: none">• Ajouter une spécialité : L'admin clique sur le bouton ajouter, la nouvelle liste des spécialités sera ajoutés• Supprimer une spécialité : L'admin clique sur le bouton supprimer, la nouvelle liste des spécialités sera ajoutés.• Modifier une spécialité : L'admin choisit une spécialité pour la modifier, l'admin remplit le champ à modifier, la spécialité est modifiée avec succès.

Scenario d'exception	Contrôle de saisie sur les champs obligatoires : <ul style="list-style-type: none"> • Un message d'erreur sera affiché pour les champs non validés • Le système reproduit le message d'erreur jusqu'à ce que les champs soient validés
Cas d'utilisation 4 :	Consulter la liste des métriques médicales
Préconditions	<ul style="list-style-type: none"> • L'administrateur doit être authentifié. • L'administrateur doit consulter la liste des métriques médicales (Accéder à la page où un tableau de données est affiché, ayant toutes les métriques médicales existants, à partir de son menu).
Postconditions	<ul style="list-style-type: none"> • Métrique ajoutée • Métrique modifiée • Métrique supprimée
Scenario principal	<p>Après l'authentification et la consultation de la liste des spécialités des médecins, l'administrateur aura le choix de :</p> <ul style="list-style-type: none"> • Ajouter une métrique : L'admin clique sur le bouton ajouter, la nouvelle liste des métriques sera ajoutés • Supprimer une métrique : L'admin clique sur le bouton supprimer, la nouvelle liste des métriques sera ajoutés. • Modifier une métrique : L'admin choisit une métrique pour la modifier, l'admin remplit le champ à modifier, la métrique est modifiée avec succès.

Scenario	Contrôle de saisie sur les champs obligatoires :
D'exception	<ul style="list-style-type: none">• Un message d'erreur sera affiché pour les champs non validés• Le système reproduit le message d'erreur jusqu'à ce que les champs soient validés

Table 5:Tableau descriptif du cas d'utilisation « Administrateur »

8.1.2. Diagramme de cas d'utilisation « Patient »

La figure 8 représente le diagramme de cas d'utilisation de l'acteur "Patient" :

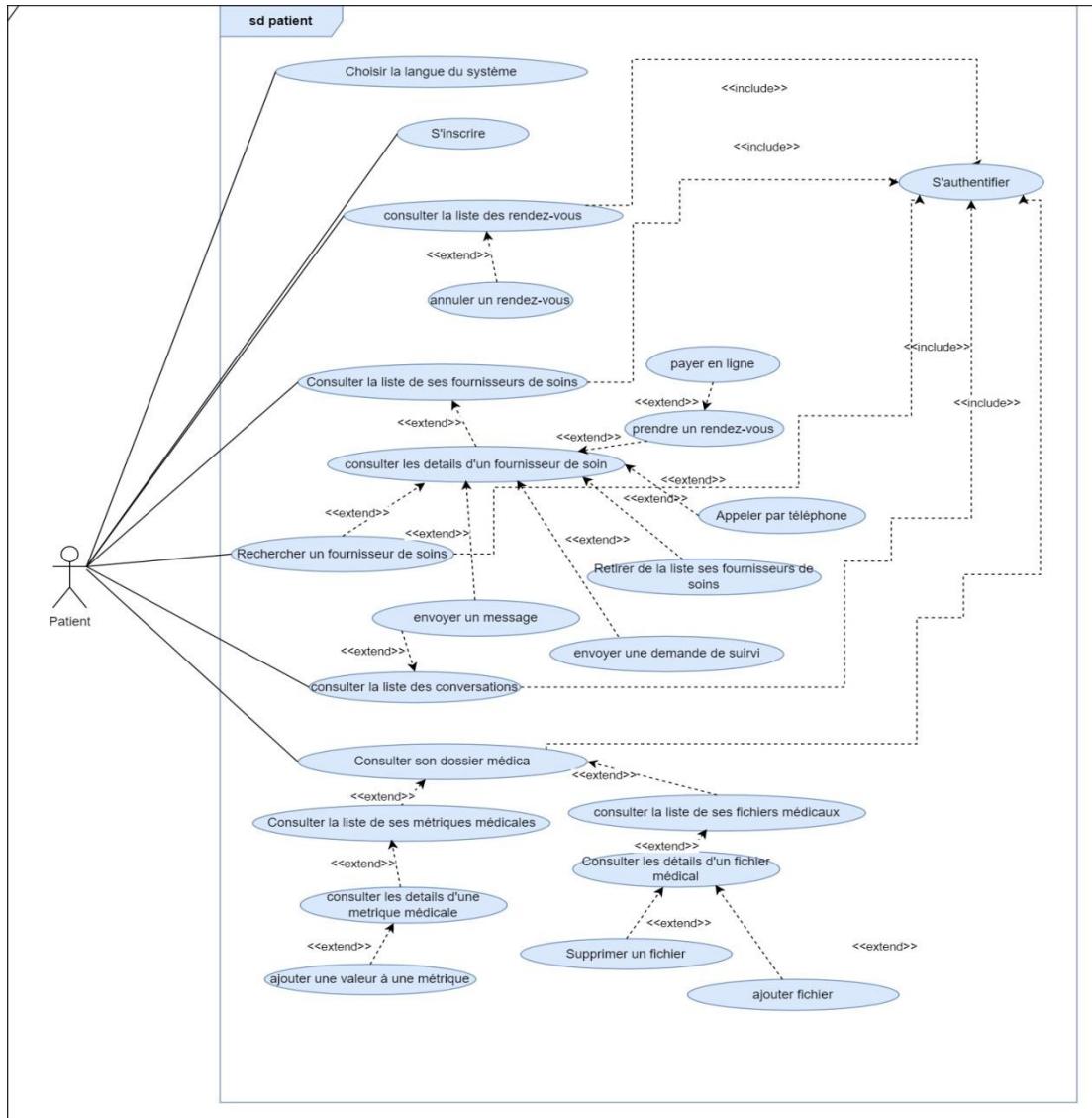


Figure 8: Diagramme de cas d'utilisation de l'acteur « Patient ».

Le tableau 6 ci-dessous, décrit en détails les différents cas d'utilisation du "patient " présentés dans la figure 8 :

Acteurs	Patient
----------------	---------

Cas d'utilisation 1	Choisir la langue de système
Postconditions	<ul style="list-style-type: none"> • La langue Anglais • La langue Français • La langue Arabe
Scénario principal	<ul style="list-style-type: none"> • Une fois que l'utilisateur ouvre l'application, une liste de langues sera affichée parmi lesquelles il pourra choisir la langue du système : en cliquant sur la langue souhaitée, toute l'application se mettra dans cette langue.
Cas d'utilisation 2	Authentification
Préconditions	<ul style="list-style-type: none"> • Avoir déjà un compte
Postconditions	<ul style="list-style-type: none"> • Accéder à son espace personnel.
Scénario principal	<ul style="list-style-type: none"> • Pour accéder à leur compte, les utilisateurs doivent se connecter en saisissant correctement leur adresse e-mail et leur mot de passe. Avant de pouvoir accéder à leur espace personnel
Scenario D'exception	<ul style="list-style-type: none"> • Si l'adresse e-mail ou le mot de passe saisi est incorrect, un message d'erreur s'affichera. Dans le cas où l'utilisateur aurait besoin de le faire, il lui sera possible de réinitialiser ses informations. • Si l'un des champs est laissé vide, l'utilisateur recevra un message d'erreur l'informant de la nécessité de remplir tous les champs obligatoires.
Cas d'utilisation 3	Envoyer une demande de suivi
Préconditions	<ul style="list-style-type: none"> • Le patient doit être authentifié. • Le patient doit consulter l'interface de recherche

	<ul style="list-style-type: none"> Le patient doit consulter le profil du fournisseur de soins
Postconditions	<ul style="list-style-type: none"> Invitation envoyée
Scénario principal	<p>Après l'authentification et la consultation de l'interface de recherche et la sélection du fournisseur désiré, le patient va :</p> <ul style="list-style-type: none"> Envoyer une demande de suivi : le patient clique sur le bouton "suivre", le système affiche un message de succès et une invitation sera envoyée au destinataire Annuler la demande de suivi déjà envoyé : après que le patient clique sur le bouton "annuler", le système affiche un message indiquant que l'invitation est annulée
Cas d'utilisation 4	Ajouter un fichier médical
Préconditions	<ul style="list-style-type: none"> Le client doit être authentifié. Le client doit consulter l'interface de son dossier médical
Postconditions	<ul style="list-style-type: none"> Fichier ajouté
Scénario principal	<p>Après l'authentification et la consultation de son dossier médical, le patient aura la possibilité de :</p> <ul style="list-style-type: none"> Ajouter une fiche d'analyse : le patient accède à l'interface dédiée à l'ajout d'une fiche d'une analyse ; le système affiche le formulaire nécessaire ; le patient insère les données et enregistre ; le système vérifie et si patient a partagé cette fiche avec un médecin ce dernier va recevoir une notification
Scenario D'exception	<ul style="list-style-type: none"> Si l'un des champs obligatoires est laissé vide, un message d'erreur s'affiche indiquant la nécessité de remplir tous les champs.
Cas d'utilisation 5	Prendre un rendez-vous

Préconditions	<ul style="list-style-type: none"> • Le patient doit être authentifié. • Le patient doit consulter le profil d'un de ses fournisseurs de soins (un médecin)
Postconditions	<ul style="list-style-type: none"> • Rendez-vous réservés et payés • Rendez-vous réservés
Scénario principal	<ul style="list-style-type: none"> • Le patient clique sur le bouton « prendre un rendez-vous ». • Le système affiche la page correspondante. • Le patient remplit le formulaire correspondant, • Le système affiche une interface afin que le patient puisse choisir entre : <ol style="list-style-type: none"> 1. Payée en ligne : le patient clique sur le bouton 'payée en ligne' le système affiche la page de paiement, le patient peut choisir une méthode de paiement, 2. Ignore le paiement en ligne : le patient clique sur le bouton 'annuler le paiement en ligne', le système retourne à l'interface du médecin
Scenario D'exception	<ul style="list-style-type: none"> • Si l'un des champs obligatoires est laissé vide, un message d'erreur s'affiche indiquant la nécessité de remplir tous les champs.

Table 6:bleau descriptif du cas d'utilisation «Patient»

8.1.3. Diagramme de cas d'utilisation « Médecin »

La figure 9 représente le diagramme de cas d'utilisation de l'acteur "Médecin"

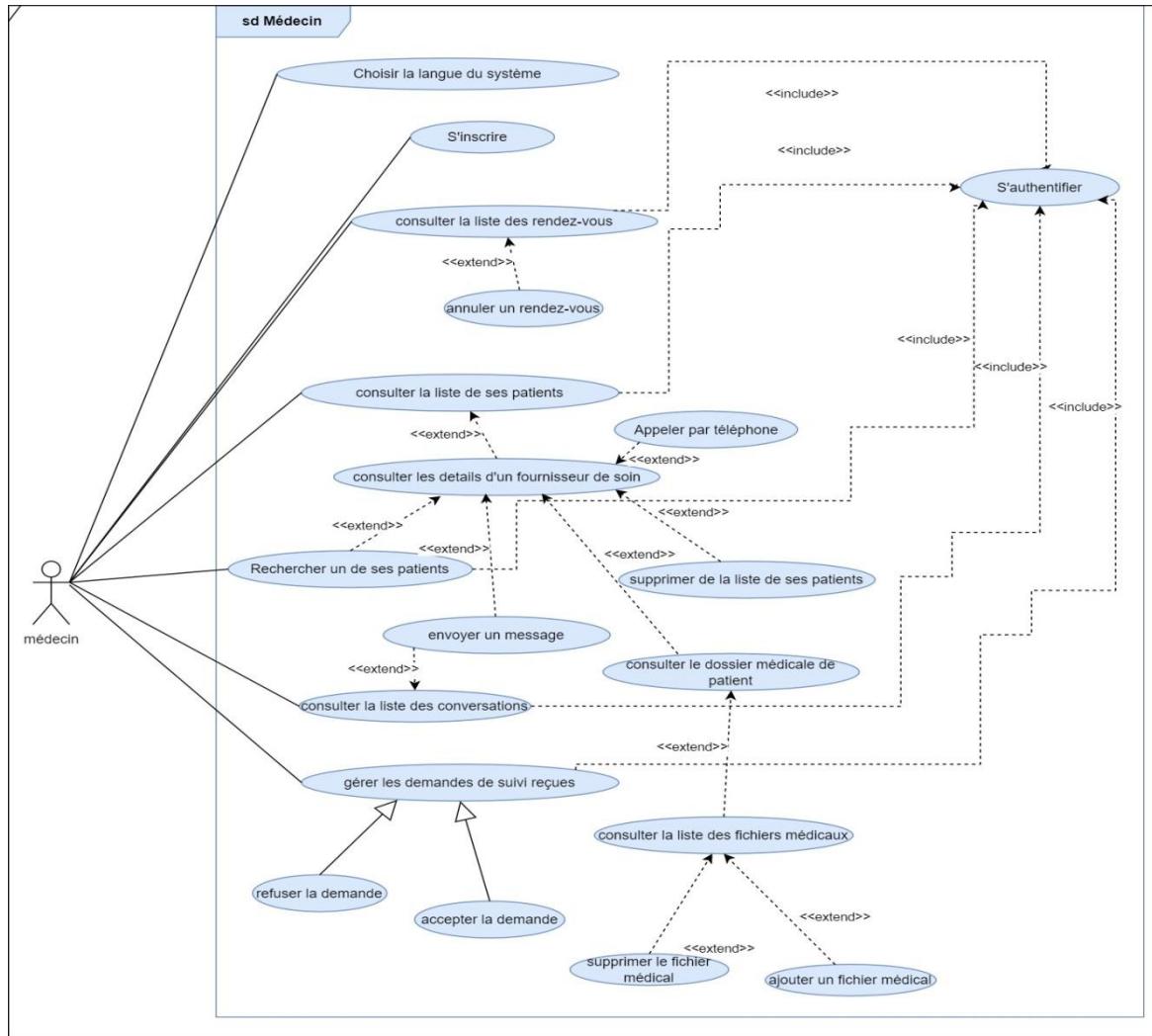


Figure 9: Diagramme de cas d'utilisation de l'acteur « Médecin ».

Le tableau n°7 ci-dessous, décrit en détails les différents cas d'utilisation du « Médecin » présentés dans la figure 9 :

Acteur	Médecin
Cas d'utilisation 1	S'authentifier

Préconditions	<ul style="list-style-type: none"> • Avoir déjà un compte
Postconditions	<ul style="list-style-type: none"> • Accéder à son espace personnel.
Scénario principal	<ul style="list-style-type: none"> • Pour accéder à leurs comptes, les Médecins doivent se connecter en entrant leurs adresses e-mail et leurs mots de passe correctement. Ces Médecins doivent s'inscrire au préalable et être approuvés par l'administrateur avant de pouvoir accéder à leur espace personnel. • Le système vérifie les données entrées par l'utilisateur.
Scénario D'exception	<ul style="list-style-type: none"> • Si l'adresse e-mail et le mot de passe entrés sont incorrects, un message d'erreur s'affiche. Sinon il est possible pour l'utilisateur de réinitialiser ses données si nécessaire. • Si l'un des champs est laissé vide, l'utilisateur recevra un message d'erreur l'informant de la nécessité de remplir tous les champs. • Si l'administrateur n'a pas encore approuvé la demande d'inscription, un message va s'afficher indiquant que le compte n'a pas été encore vérifié et une notification sera envoyée à l'administrateur pour l'informer de vérifier la liste des demandes
Cas d'utilisation 2	Gérer les invitations reçues
Préconditions	<ul style="list-style-type: none"> • Le médecin doit être authentifié. • Le médecin doit consulter la liste des invitations.
Postconditions	<ul style="list-style-type: none"> • Invitation acceptée. • Invitation rejetée.
Scénario principal	<p>Après l'authentification et l'accès à la liste des invitations, le médecin aura le choix entre :</p> <ul style="list-style-type: none"> • Accepter une invitation : le médecin clique sur le bouton « Accepter » ; le comptable confirme l'acceptation ; le

	<p>système change le statut de la demande de « en attente » à « acceptée ».</p> <ul style="list-style-type: none"> • Rejeter une invitation : le médecin clique sur le bouton « Rejeter » ; le système change le statut de la facture de « en attente » à « rejetée ».
Cas d'utilisation 3	Consulter le dossier médical de patient
Préconditions	<ul style="list-style-type: none"> • Le médecin doit être authentifié. • Le médecin doit consulter le profil d'un patient de ses patients.
Postconditions	<ul style="list-style-type: none"> • Courbe d'évaluation d'une métrique médicale
Scénario principal	<p>Après l'authentification, le médecin consulte le profil du patient pour :</p> <ul style="list-style-type: none"> • Consulter l'évaluation d'une métrique médicale : le médecin clique sur le bouton “Métriques” ; le système affiche l'interface de toutes les métriques de patient ; lorsque le médecin clique sur l'une de ces métriques, une courbe associée à l'évaluation des valeurs de la métrique sélectionnée s'affiche.

Table 7:Tableau descriptif du cas d'utilisation « Médecin »

8.1.4. Diagramme de cas d'utilisation « Agent de laboratoire »

La figure 10 représente le diagramme de cas d'utilisation de l'acteur "Agent de laboratoire" :

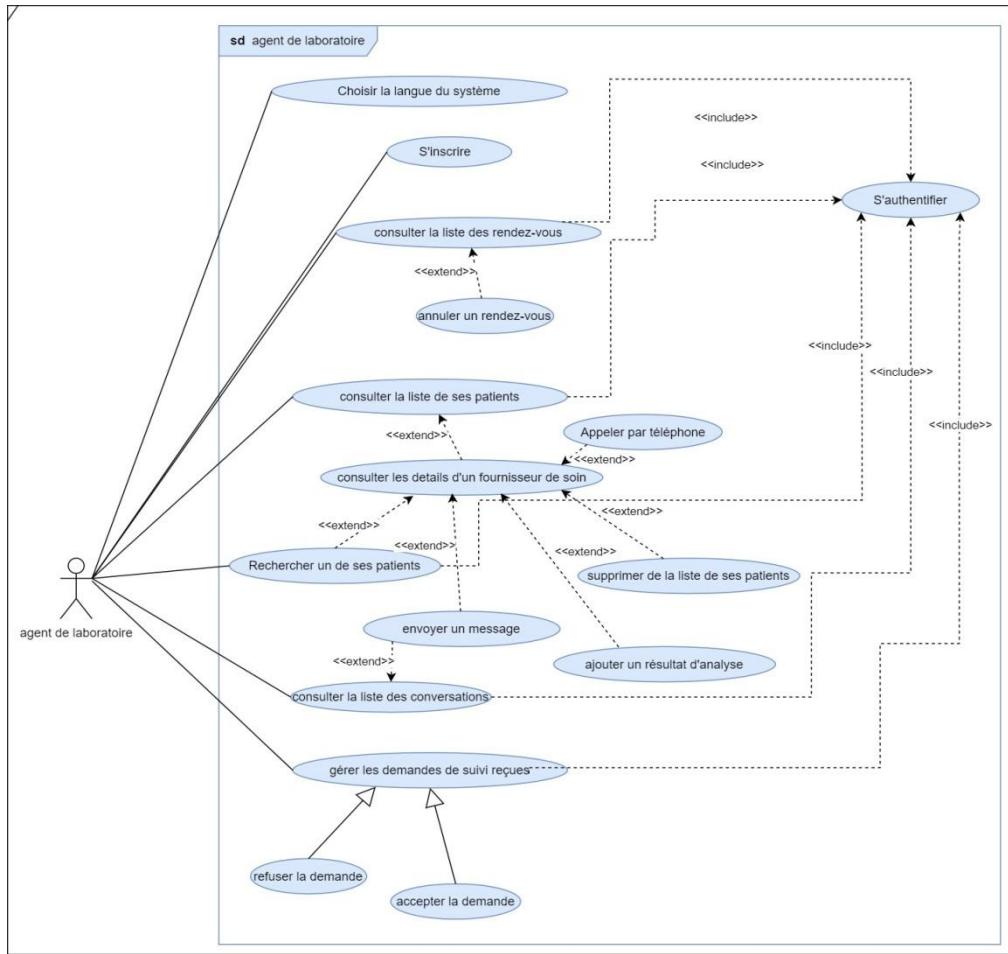


Figure 10: Diagramme de cas d'utilisation de l'acteur «Agent de laboratoire».

Le tableau n°8 ci-dessous, décrit en détails les différents cas d'utilisation du « Agent de laboratoire » présentés dans la figure 10 :

Acteur	Agent de laboratoire
Cas d'utilisation 1	Annuler un rendez-vous

Préconditions	<ul style="list-style-type: none"> • L'agent du laboratoire doit être authentifié. • L'agent du laboratoire doit consulter la liste des rendez-vous
Postconditions	<ul style="list-style-type: none"> • Rendez-vous annulés
Scénario principal	<p>Après l'authentification, L'agent du laboratoire consulte la liste des rendez-vous :</p> <ul style="list-style-type: none"> • Annuler un rendez-vous : l'agent clique sur le bouton “annuler” ; le système affiche “rendez-vous annulés”, et retourne la nouvelle liste des rendez-vous,
Cas d'utilisation 2	Ajouter un résultat d'analyse
Préconditions	<ul style="list-style-type: none"> • L'agent doit être authentifié. • L'agent doit consulter le profil d'un patient de ses patients.
Postconditions	<ul style="list-style-type: none"> • Résultat d'analyse ajoutée
Scénario principal	<p>Après l'authentification, l'agent de laboratoire consulte le profil du patient pour :</p> <ul style="list-style-type: none"> • Ajouter un résultat d'analyse : l'agent clique sur le bouton “ajouter une analyse” ; le système affiche le formulaire nécessaire ; l'agent insère les données et enregistre ; le système vérifie et le patient reçoit une notification

Scénario D'exception	<ul style="list-style-type: none"> Si l'un des champs obligatoires est laissé vide, un message d'erreur s'affiche indiquant la nécessité de remplir tous les champs.
---------------------------------	---

Table 8:Tableau descriptif du cas d'utilisation « Agent du laboratoire »

8.1.5. Diagramme de cas d'utilisation « Agent du centre d'imagerie médical »

La figure 11 représente le diagramme de cas d'utilisation de l'acteur " Agent du centre d'imagerie médicale " :

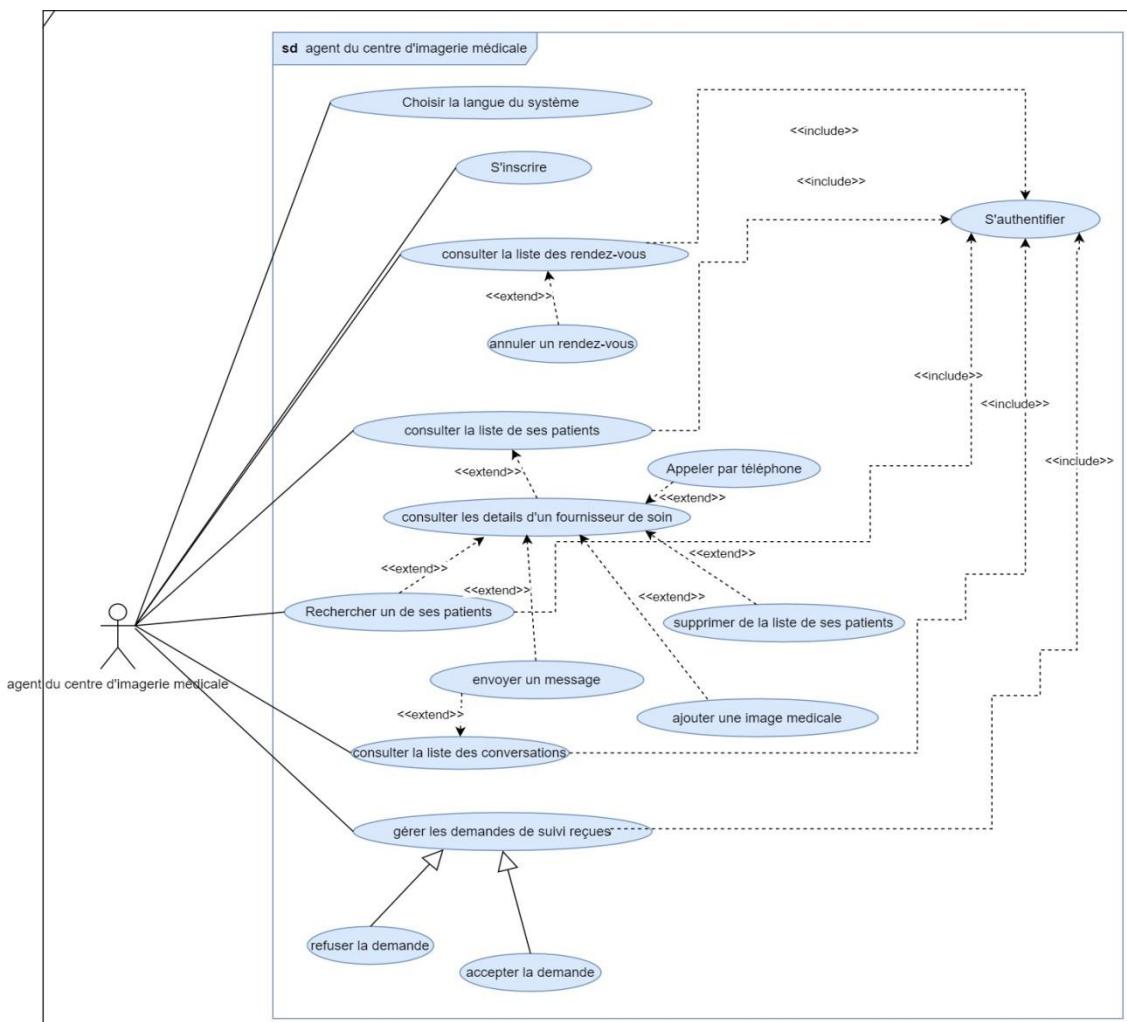


Figure 11: Diagramme de cas d'utilisation de l'acteur «Agent du centre d'imagerie médical»

Le tableau n°9 ci-dessous, décrit en détails les différents cas d'utilisation du « Agent du centre d'imagerie médical » présentés dans la figure 11 :

Acteur :	Agent du centre d'imagerie médical
Cas d'utilisation 1	Supprimer de la liste de ses patients
Préconditions	<ul style="list-style-type: none">• L'agent doit être authentifié.• L'agent doit consulter le profil d'un patient de ses patients.
Postconditions	<ul style="list-style-type: none">• Patient retirée de la liste
Scénario principal	Après l'authentification, l'agent du centre médical consulte le profil du patient pour : <ul style="list-style-type: none">• Retirer de la liste des patients : l'agent clique sur le bouton “retirer” ; le système affiche un message de “patient retirer” ; et le patient aura notifié de cette retraitation
Cas d'utilisation 2	<ul style="list-style-type: none">• Ajouter une image médicale

Préconditions	<ul style="list-style-type: none"> • L'agent doit être authentifié. • L'agent doit consulter le profil d'un patient de ses patients.
Postconditions	<ul style="list-style-type: none"> • Radiographie ajoutée
Scénario principal	<p>Après l'authentification, l'agent du centre médical consulte le profil du patient pour :</p> <ul style="list-style-type: none"> • Ajouter une fiche d'une radiographie : l'agent clique sur le bouton “ajouter une radiographie” ; le système affiche le formulaire nécessaire ; l'agent insère les données et enregistre ; le système vérifie et le patient reçoit une notification
Scénario D'exception	<ul style="list-style-type: none"> • Si l'un des champs obligatoires est laissé vide, un message d'erreur s'affiche indiquant la nécessité de remplir tous les champs.

Table 9:Tableau descriptif du cas d'utilisation « Agent du centre d'imagerie médical »

9. Conclusion

Au cours de ce chapitre, nous avons identifié les acteurs impliqués dans notre application, détaillé les exigences fonctionnelles et non fonctionnelles, et présenté le tableau du backlog produit. Le prochain chapitre se concentrera sur une étude conceptuelle approfondie de l'application.

Etude conceptuelle

10. Introduction

Dans ce chapitre, nous adopterons une approche structurée de modélisation pour mieux comprendre le système que nous allons développer. Tout d'abord, nous présenterons la conception générale de notre projet en détaillant son architecture basée sur l'architecture n-tiers et le modèle MVC. Ensuite, nous nous pencherons sur une conception approfondie de notre application, en utilisant un diagramme de classes ainsi que quelques diagrammes de séquence pour clarifier les différentes étapes de fonctionnement.

11. Architecture de l'application

Afin d'avoir une visibilité parfaite sur l'architecture de l'application, nous allons nous concentrer dans cette partie et examiner ses différentes composantes.

11.1. Architecture MVC

L'architecture de notre application repose sur le modèle MVC (Modèle-Vue-Contrôleur). C'est un modèle architectural qui sépare une application en trois composants logiques principaux comme le montre la figure suivante :



Figure 12 : Les composants MVC

- **Modèle :** noyau de l'application qui gère les données, et permet de récupérer les informations dans la base de données, de les organiser pour qu'elles puissent ensuite être traitées par le contrôleur.
- **Vue :** cette partie se concentre sur l'affichage. Elle ne fait presque aucun calcul et se contente de récupérer des variables pour savoir ce qu'elle doit afficher.
- **Contrôleur :** cette partie gère la logique du code qui prend des décisions. C'est en quelque sorte l'intermédiaire entre le modèle et la vue : le contrôleur va demander au modèle les données, les analyser, prendre des décisions et renvoyer le texte à afficher à la vue [5].

Le schéma de la figure 13 suivante présente un schéma descriptif de cette architecture :

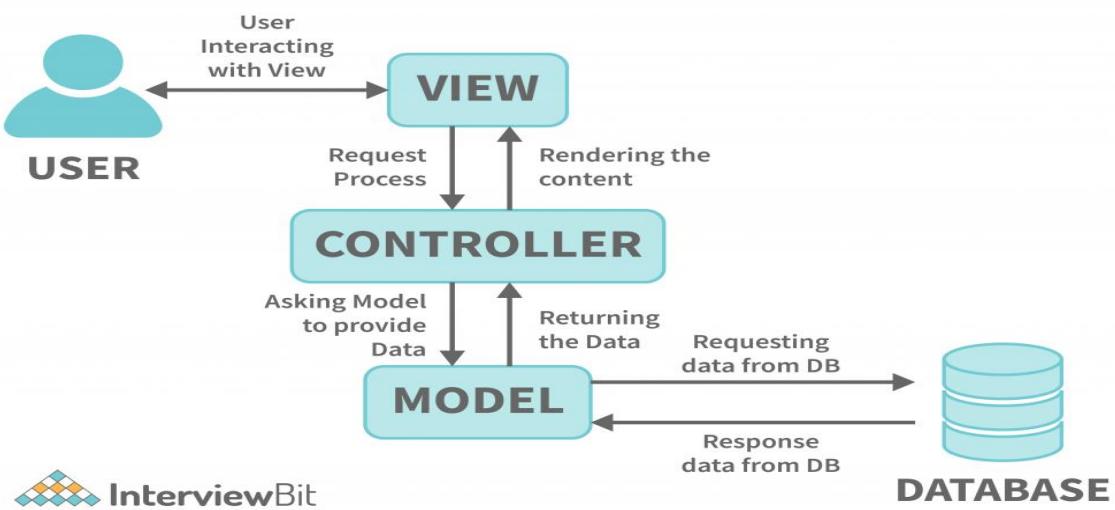


Figure 13: Schéma de l'architecture MVC

12. Conception détaillée

Dans cette étape, nous allons présenter en premier lieu le modèle statique qui décrit la structure et le comportement des objets constituant notre application grâce au diagramme de classe, puis le modèle dynamique avec les diagrammes de séquences.

12.1. Choix du langage de conception

UML, ou Unified Modeling Language, est un langage universel de modélisation objet utilisé comme une notation visuelle pour représenter les différents aspects d'un système logiciel. Il facilite la communication entre les acteurs du développement sans être un langage de programmation ou un processus de développement spécifique. UML est indépendant d'un langage de programmation particulier, ce qui en fait un outil flexible et puissant pour concevoir et communiquer des modèles logiciels [6].

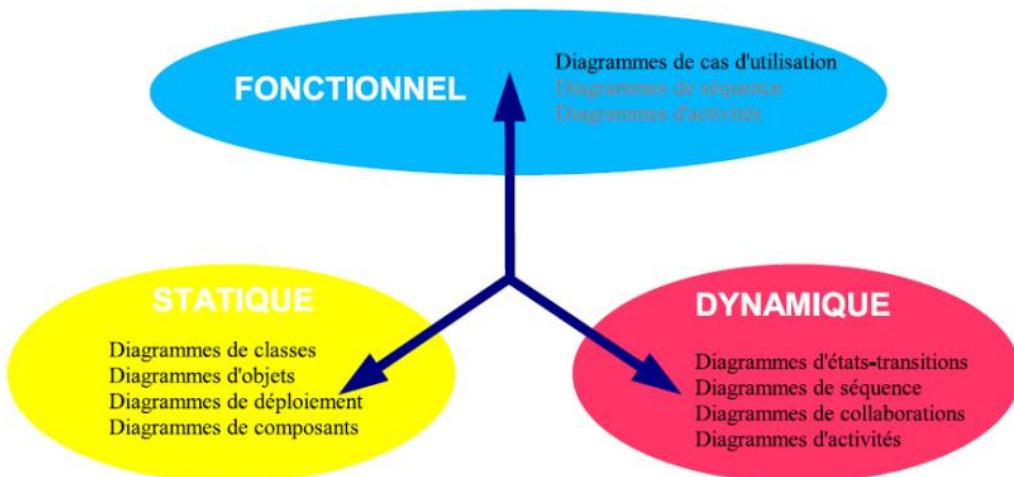


Figure 14: Les différentes vues de UML

12.2. Diagramme de classes de l'application

Le diagramme de classes est un schéma utilisé en génie logiciel pour représenter les classes et les interfaces d'un système ainsi que les relations entre elles. Chaque classe est composée d'un ensemble d'attributs liés par un champ sémantique commune.

En utilisant le diagramme présenté dans la figure 15 ci-dessous, nous pouvons identifier les entités de la base de données qui correspondent à cette application.

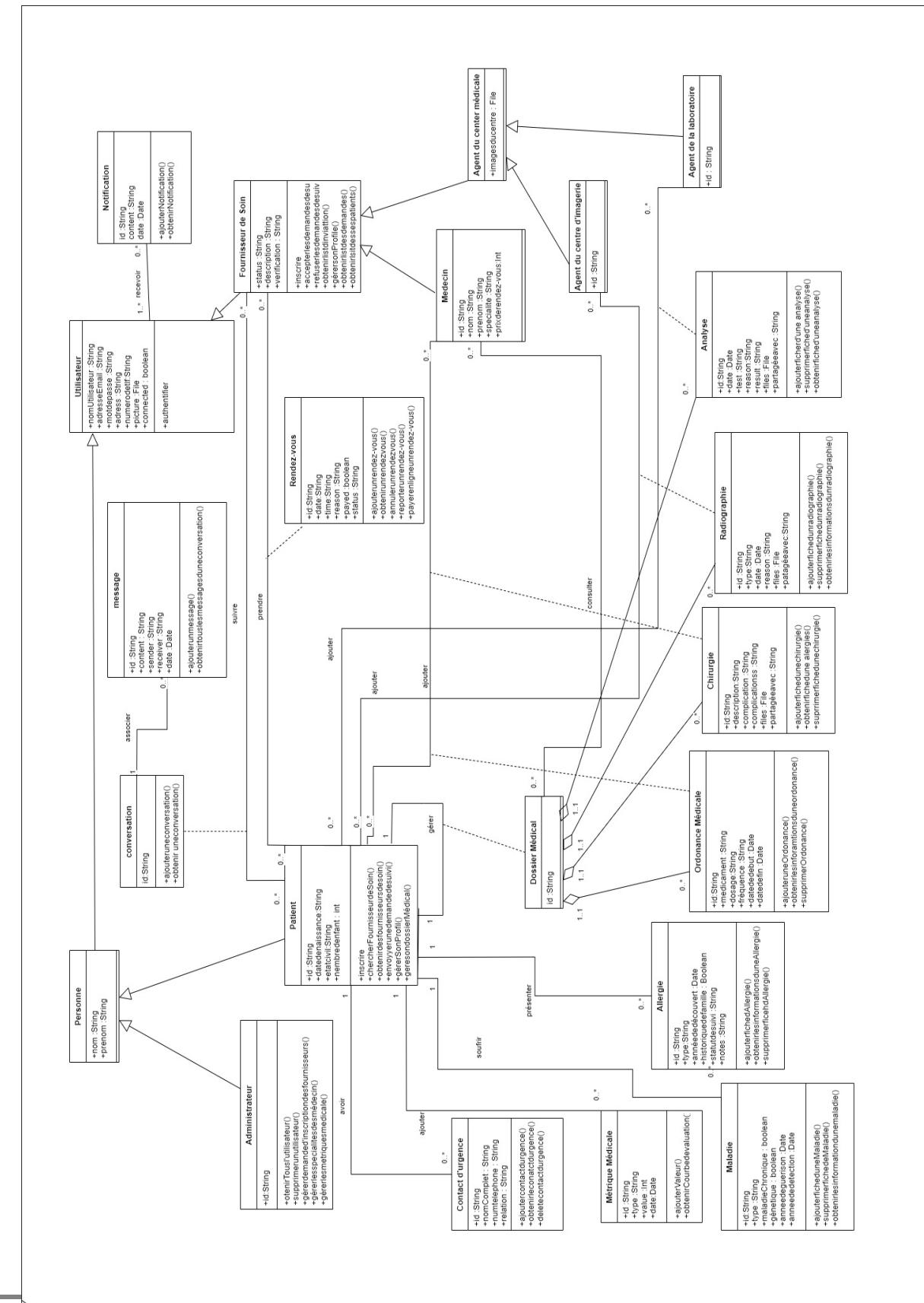


Figure 15: Diagramme de classes de l'application

12.3. Les diagrammes de séquence

Un diagramme de séquence est un diagramme UML qui représente la séquence de messages entre les objets au cours d'une interaction. Un diagramme de séquence comprend un groupe d'objets, représentés par des lignes de vie, et les messages que ces objets échangent lors de l'interaction. Les diagrammes de séquence représentent la séquence de messages transmis entre des objets. Ils peuvent également représenter les structures de contrôle entre des objets [7].

12.3.1. Diagramme de séquence « authentification »

Avant de pouvoir accéder aux fonctionnalités de l'application, tous les utilisateurs doivent s'authentifier en utilisant leurs adresses e-mail et leurs mots de passe. Une fois ces informations saisies, le système les vérifie pour s'assurer de leur validité. Si les données fournies ne sont pas correctes, un message d'erreur est affiché. En cas de succès de la connexion, une interface personnalisée selon son rôle sera affichée.

La figure 16 présente le diagramme de séquence pour le cas d'utilisation "S'authentifier" :

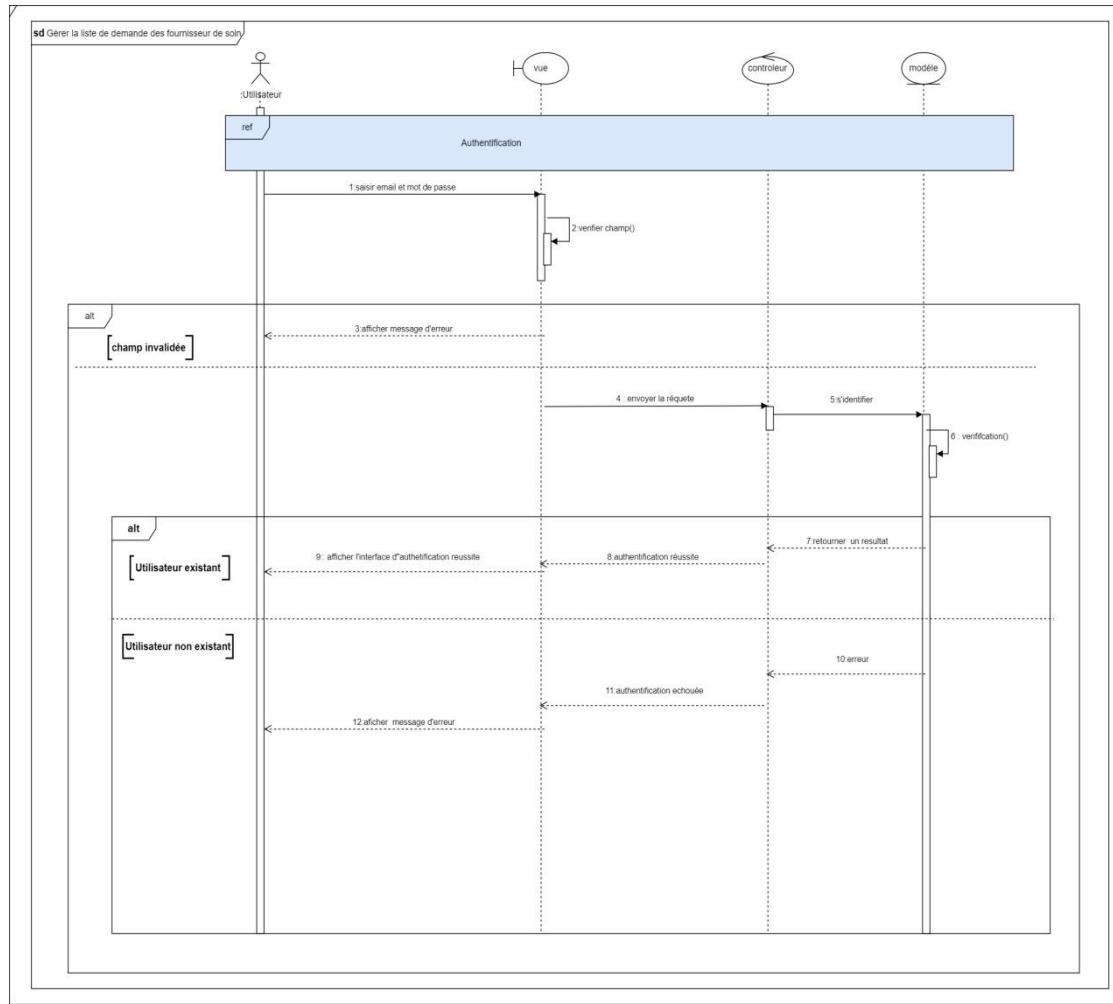


Figure 16: Diagramme de séquence d'authentification

12.3.2. Diagramme de séquence « gérer la liste de demande des fournisseurs de soins »

La figure 17 illustre le diagramme de séquence pour le cas d'utilisation "gérer la liste de demande des fournisseurs de soins"

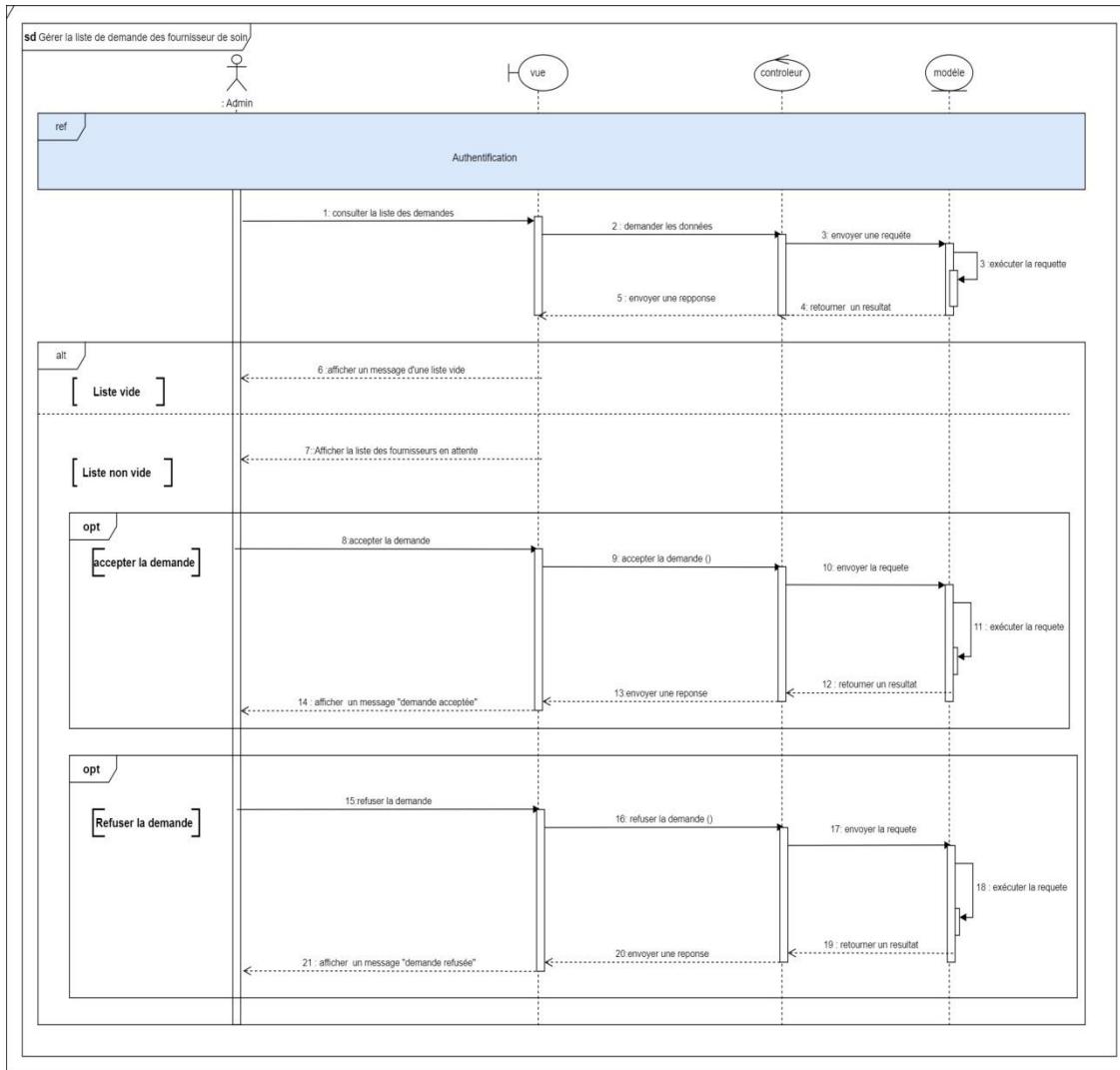


Figure 17: Diagramme de séquence de gérer la liste de demande des fournisseurs de soin

12.3.3. Diagramme de séquence « gérer les demandes de suivi »

Le diagramme de séquence illustré dans la figure 18 présente les différentes étapes que le patient doit suivre pour envoyer ou annuler une demande de suivi :

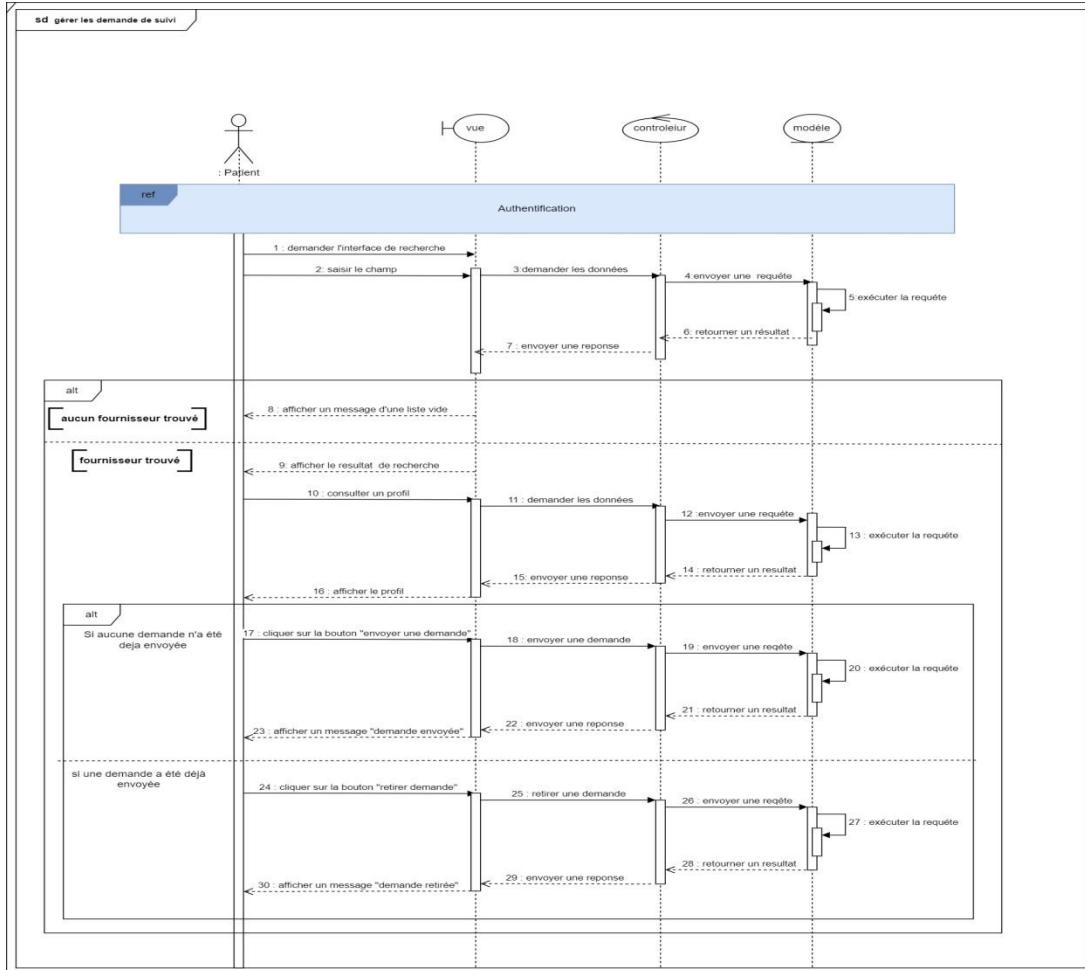


Figure 18: Diagramme de séquence gérer les demandes de suivi

12.3.4. Diagramme de séquence « ajouter d'un fichier médical »

Pour créer un fichier médical, il est essentiel de remplir un formulaire qui sera ensuite vérifié par l'interface et le contrôleur.

Le diagramme de séquence présenté dans la figure 19 décrit les différentes étapes du processus de création d'un fichier médical :

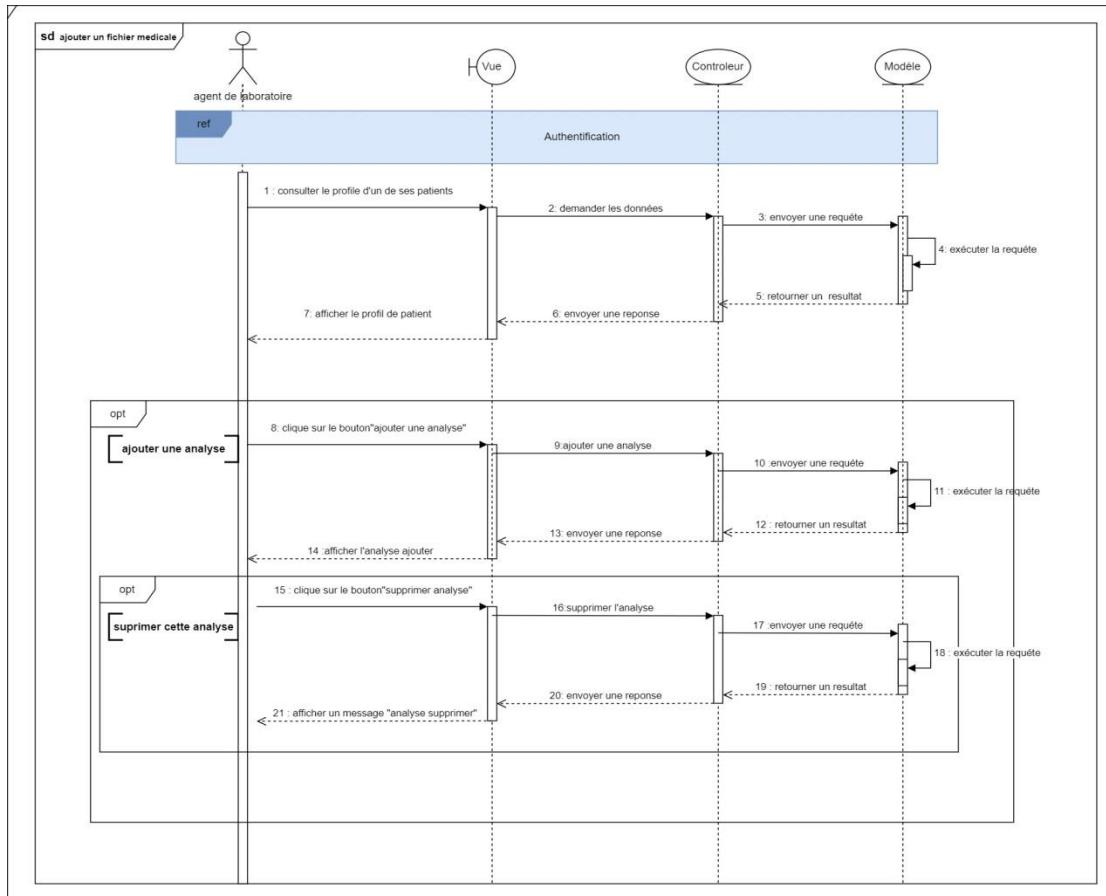


Figure 19: Diagramme de séquence ajouter d'un fichier médical

12.3.5. Diagramme de séquence « gérer un dossier médical »

Le diagramme de séquence de gérer un dossier médical est présenté dans la figure 20 suivante :

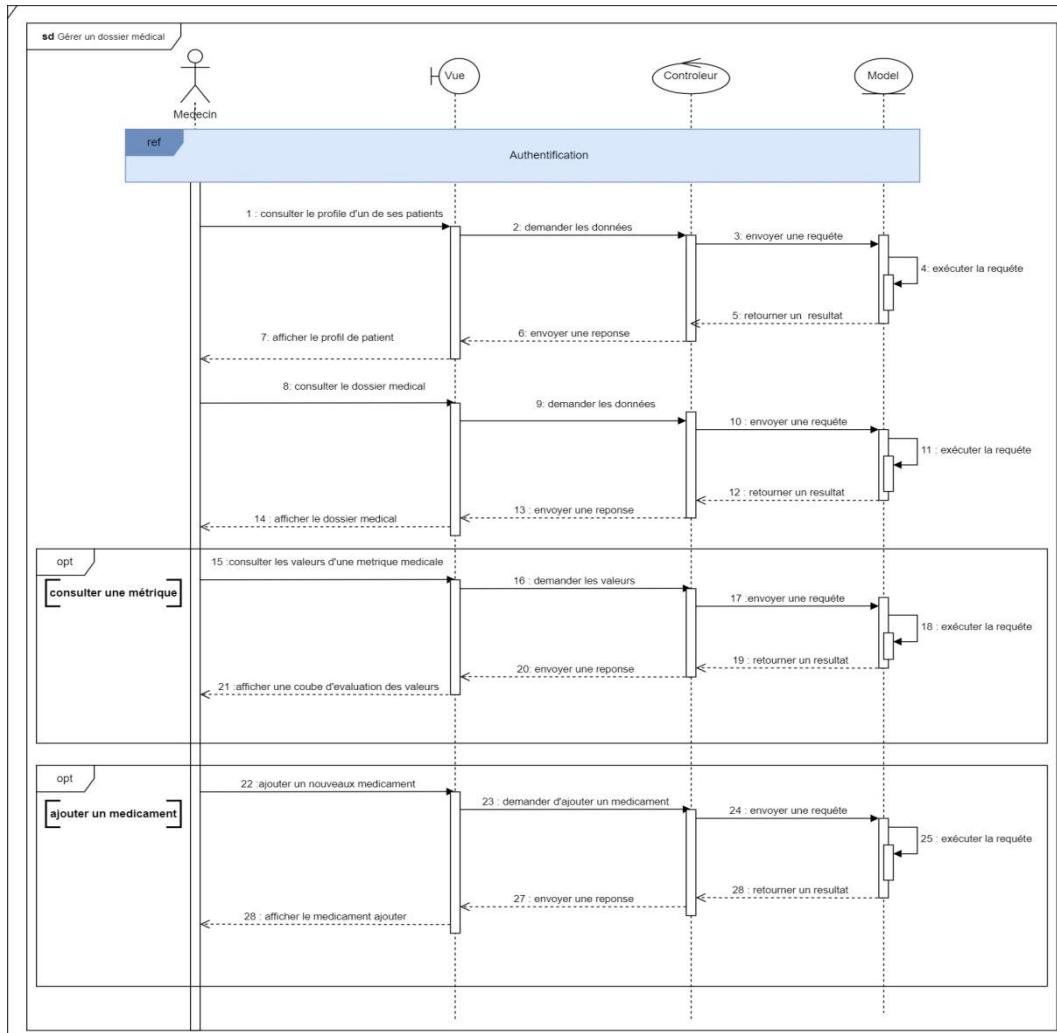


Figure 20: Diagramme de séquence de gérer un dossier médical

12.3.6. Diagramme de séquence « Messagerie »

Le diagramme de séquence de Messagerie est présenté dans la figure 21 suivante :

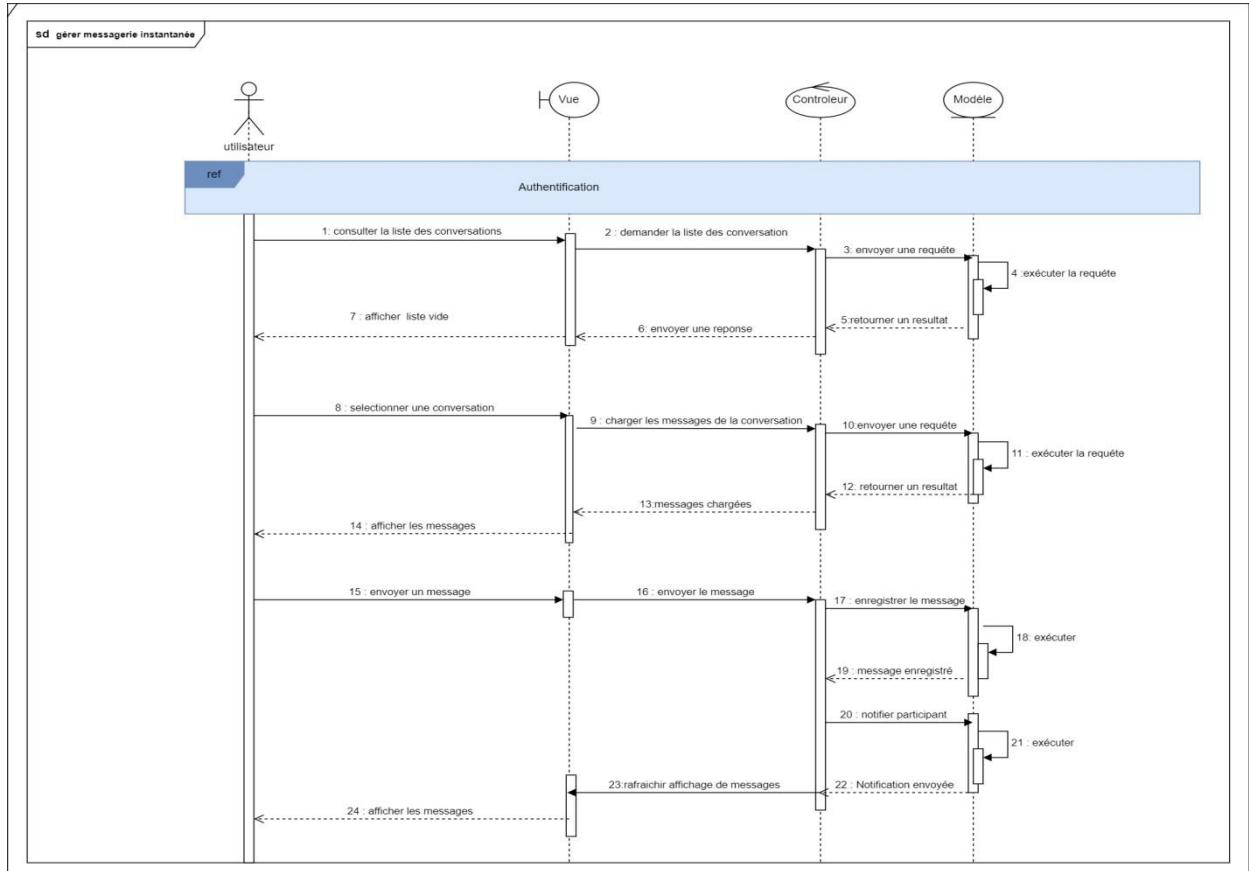


Figure 21: Diagramme de séquence de Messagrie

13. Conclusion

Dans le chapitre précédent, nous avons examiné de manière approfondie les aspects statiques et dynamiques de notre plateforme. Pour ce faire, nous avons utilisé divers outils tels que les diagrammes de classe et les diagrammes de séquence, afin d'obtenir une meilleure compréhension du fonctionnement de notre système. Dans le chapitre suivant, nous nous concentrerons sur la mise en œuvre concrète de notre projet.

Réalisation

14. Introduction

Ce chapitre vise à présenter les différentes technologies et outils utilisés dans le cadre de notre projet, ainsi que les logiciels qui ont été employés. De plus, nous allons aborder la présentation des interfaces conçues et développées pour cette application dans la partie test et validation. Par ailleurs, nous allons présenter les étapes suivies dans le déploiement du projet.

15. Environnement de travail

Dans cette section, nous allons décrire l'environnement matériel et logiciel qui a été utilisé tout au long de la réalisation de ce projet.

15.1. Technologies utilisées

Les technologies utilisées pour la réalisation de ce projet, sont décrites dans ce suit :

15.1.1. Node.js



Figure 22: Logo de Node js

Node.js est une plateforme de développement Javascript. Ce n'est pas un serveur, ce n'est pas un framework, c'est juste le langage Javascript avec des bibliothèques permettant de réaliser des actions comme écrire sur la sortie standard, ouvrir/fermer des connections réseau ou encore créer un fichier [8].

15.1.2. Flutter



Figure 23: Logo de Flutter

Flutter est un Framework open source développé et pris en charge par Google. Les développeurs frontend et full-stack utilisent Flutter pour créer l'interface utilisateur (UI) d'une application pour plusieurs plateformes avec une seule base de code.

Lorsque Flutter a été lancé en 2018, il prenait principalement en charge le développement d'applications mobiles [9].

15.1.3. MongoDB



Figure 24: Logo de Monogodb

MongoDB est une base de données NoSQL open source. Puisqu'il s'agit d'une base de données non relationnelle, elle peut traiter des données structurées, semi-structurées et non structurées. Elle utilise un modèle de données non relationnel, orienté document, et un langage de requête non structuré. MongoDB est très flexible et permet d'associer et de stocker plusieurs types de données. [10].

15.1.4. Firebase



Figure 25: Logo Firebase

Firebase est une plateforme complète de développement d'applications de Google qui a révolutionné la façon dont les développeurs créent, déploient et développent leurs applications. Grâce à son ensemble de fonctionnalités robustes, Firebase aide les développeurs à économiser du temps et des efforts en rationalisant les différents aspects du développement d'applications. [11].

15.1.5. Socket.IO



Figure 26: Logo Socket.IO

Socket.IO est une bibliothèque qui permet une communication à faible latence, bidirectionnelle et basée sur les événements entre un client et un serveur [12].

15.2. Les outils utilisés :

Les outils utilisés pour la réalisation de ce projet, sont décrits et mentionnés dans ce suit :

15.2.1. Visual Studio Code



Figure 27: Logo Visual Studio Code

Visual Studio Code est un éditeur de code open-source développé par Microsoft supportant un très grand nombre de langages grâce à des extensions. Il supporte l'autocomplétions, la coloration syntaxique, le débogage, et les commandes git. [13].

15.2.2. Postman



Figure 28: Logo de Postman

Postman est une application permettant de tester des API, créée en 2012 à Bangalore pour répondre à une problématique de test d'API partageable. D'abord module complémentaire de Google Chrome, puis client lourd, et finalement client léger, elle est à présent utilisée par plus de 500 000 entreprises dans le monde [14].

15.2.3. GitHub



Figure 29: Logo de GitHub

GitHub est un service d'hébergement Open-Source, permettant aux programmeurs et aux dévelopeurs de partager le code informatique de leurs projets afin de travailler dessus de façon collaborative. Le code source des projets est hébergé dans différents langages de programmation, et les changements apportés à chaque itération sont gardés en mémoire. Les autres utilisateurs de GitHub peuvent passer en revue le code et proposer des modifications ou des améliorations. [15].

15.2.4. StarUML :



Figure 30: logo de StarUML

StarUML est un outil dédié à la modélisation *UML*. Il offre des fonctions de modélisation orientée objet basée. Il gère les modèles dans des fichiers de projets ayant pour extension. mdj [16].

15.2.5. Git



Figure 31: Logo de git

Git est un système de contrôle de version qui a été inventé et développé par Linus Torvalds, également connu pour l'invention du noyau Linux, en 2005. Il s'agit d'un outil de développement qui aide une équipe de développeurs à gérer les changements apportés au code source au fil du temps [17].

15.2.6. Ngrok



Figure 32;logo de ngrok

Ngrok est un outil qui permet de rendre un poste de développement web local accessible depuis l'extérieur, via un tunnel sécurisé, depuis une url du type <https://azerty.ngrok.io>.

Cette url est accessible par n'importe qui et depuis n'importe où. De plus, Ngrok prend en charge les configurations délicates de pare-feu, en s'occupant d'ouvrir les ports nécessaires, en toute sécurité [18].

15.3. Les langages utilisés

Les langages utilisés pour la réalisation de ce projet, sont décrits dans ce suit :

15.3.1. HTML



Figure 33: Logo HTML

HTML signifie « HyperText Markup Language » qu'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure [19].

15.3.2. CSS



Figure 34: Logo CSS

Le CSS pour Cascading Style Sheets, est un langage informatique utilisé sur Internet pour la mise en forme de fichiers et de pages HTML. On le traduit en français par feuilles de style en cascade. Grâce au CSS, vous pouvez en effet appliquer des règles de mise en forme (titrage, alignement, polices, couleurs, bordures, etc.) [20].

15.3.3. JavaScript



Figure 35: Logo JavaScript

JavaScript (souvent abrégé en « JS ») est un langage de script léger, orienté objet, principalement connu comme le langage de script des pages web [21].

15.3.4. Dart



Figure 36: Logo Dart

Dart est un langage de programmation, qui a été et est développé principalement par Google. La programmation dans Dart doit être une alternative attrayante à JavaScript dans des navigateurs Web modernes [22].

16. Test et validation

Nous avons conçu et réalisé une application mobile médicale pour suivre l'état de santé des patients qui facilite les échanges entre les patients et les fournisseurs de soin.

Dans la section précédente, nous avons examiné en détail les divers outils et technologies utilisés tout au long du projet. Nous allons maintenant passer à la section suivante, où nous présenterons quelques interfaces de notre application, organisées selon les différents sprints mentionnés dans le premier chapitre. Cela nous donnera une vision concrète des fonctionnalités de l'application.

Les sprints sont :

- Sprint 1 : Authentification.
- Sprint 2 : Gestion des demandes d'inscription.

- Sprint 3 : Gestion des demandes de suivi.
- Sprint 4 : Création des conversations en temps réel.
- Sprint 5 : Gestion des dossiers médicaux.
- Sprint 6 : Paiement en ligne d'un rendez-vous.

16.1. Sprint 1 : Authentification

16.1.1. Interface de la page d'authentification (Patient, Médecin, agent de laboratoire, agent du centre médicale)

Cette interface offre aux utilisateurs la possibilité de se connecter et d'accéder à l'application. Dans la figure 37 suivante, nous pouvons observer l'interface d'authentification conçue pour tous les utilisateurs sauf l'administrateur. Elle permet une identification sécurisée et un accès personnalisé aux fonctionnalités de l'application, offrant ainsi une utilisation adaptée à chaque type d'utilisateur.

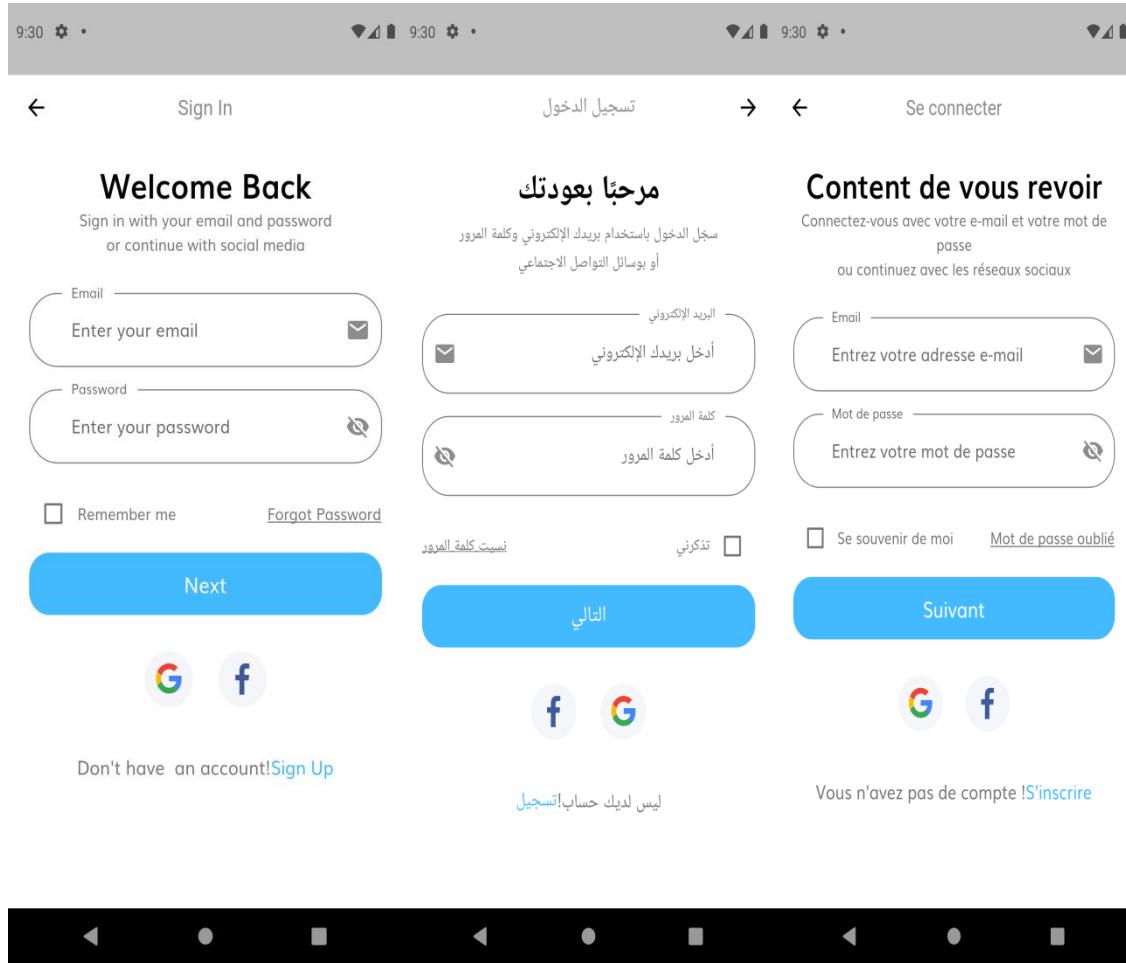


Figure 37: Interface de connexion

16.1.2. Interface d'inscription (Patient, Fournisseurs de soins)

Cette interface est spécifiquement destinée pour tous les types des utilisateurs. Contrairement à eux, l'administrateur n'a pas besoin de passer par un processus d'inscription. Chaque utilisateur doit remplir un formulaire d'inscription en sélectionnant son rôle à partir d'un bouton dédié, puis en fournissant les informations requises.

English Version:

Register Account
Complete your details or continue with social media

Patient Health care provider

Email: Enter your email

Password: Enter your password

Confirm Password: Please enter re-password

Next

French Version:

Créer un compte
Complétez vos informations ou continuez avec les réseaux sociaux

Patient Fournisseur de soin

Email: Entrez votre adresse e-mail

Mot de passe: Entrez votre mot de passe

Confirmer le mot de passe: Veuillez entrer à nouveau ...

Suivant

Arabic Version:

تسجيل حساب
أكمل معلوماتك أو استمر مع وسائل التواصل الاجتماعي

مريض مقدم الرعاية الصحية

البريد الإلكتروني: أدخل بريدك الإلكتروني

كلمة المرور: أدخل كلمة المرور

تأكيد كلمة المرور: تأكيد كلمة المرور

يرجى إعادة إدخال كلمة المرور

التالي

Buttons: Back, Next, Previous, Done.

Logos: Google, Facebook.

Text Labels: By continuing you confirm that you agree with our Term and Condition. En continuant, vous confirmez que vous acceptez nos conditions générales d'utilisation. من خلال المتابعة، فإنك توافق على اتفاقاتنا.

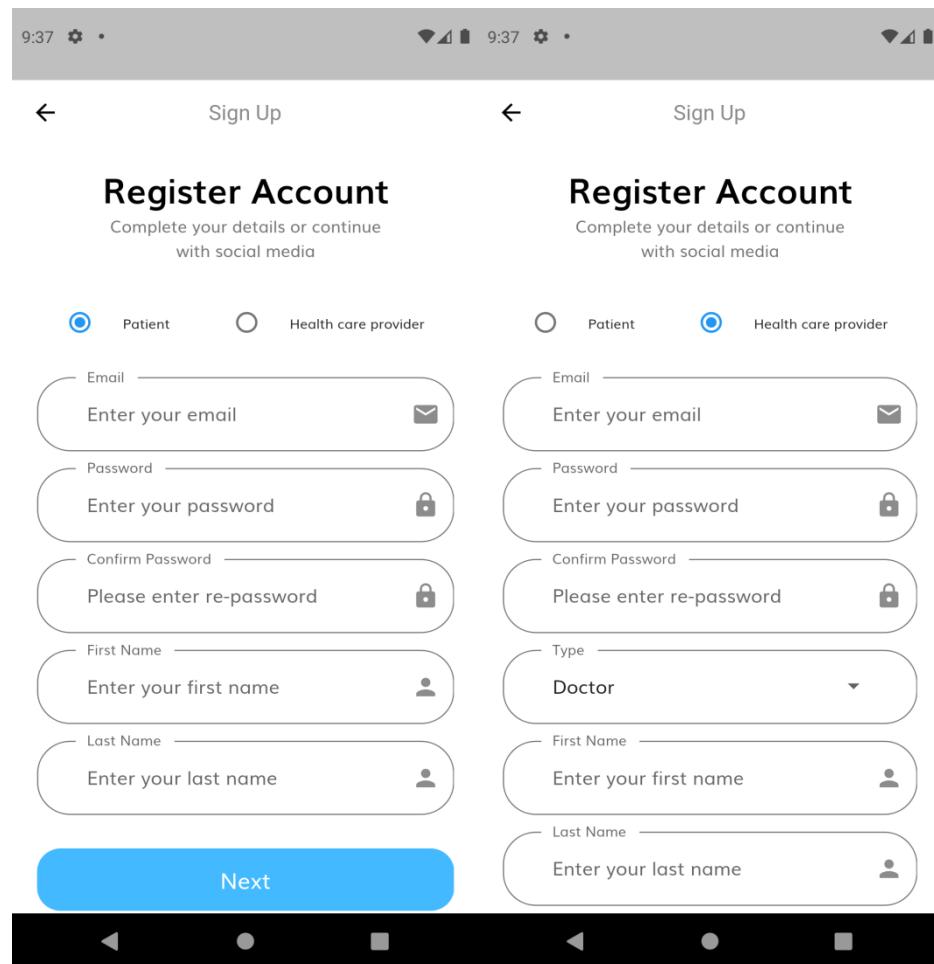


Figure 368: Interface d'inscription

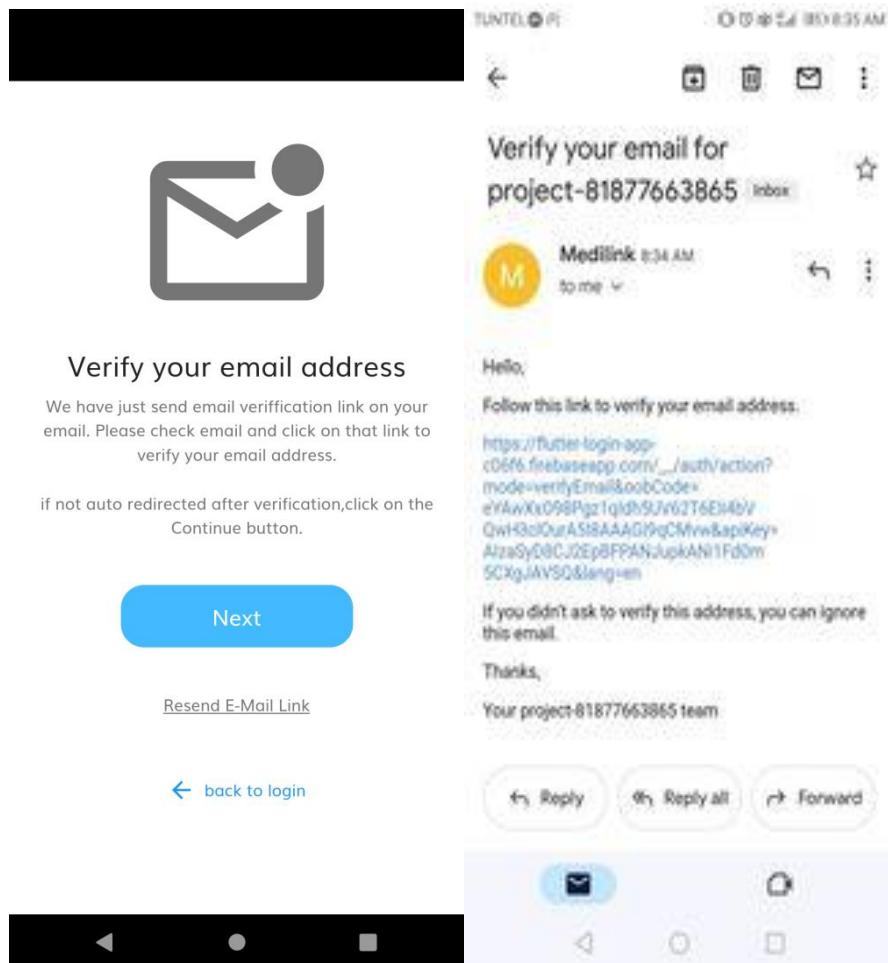


Figure 379: Interface d'inscription

Après la vérification de son mail le fournisseur de soins doit compléter ce formulaire et attendre la vérification de l'administrateur.

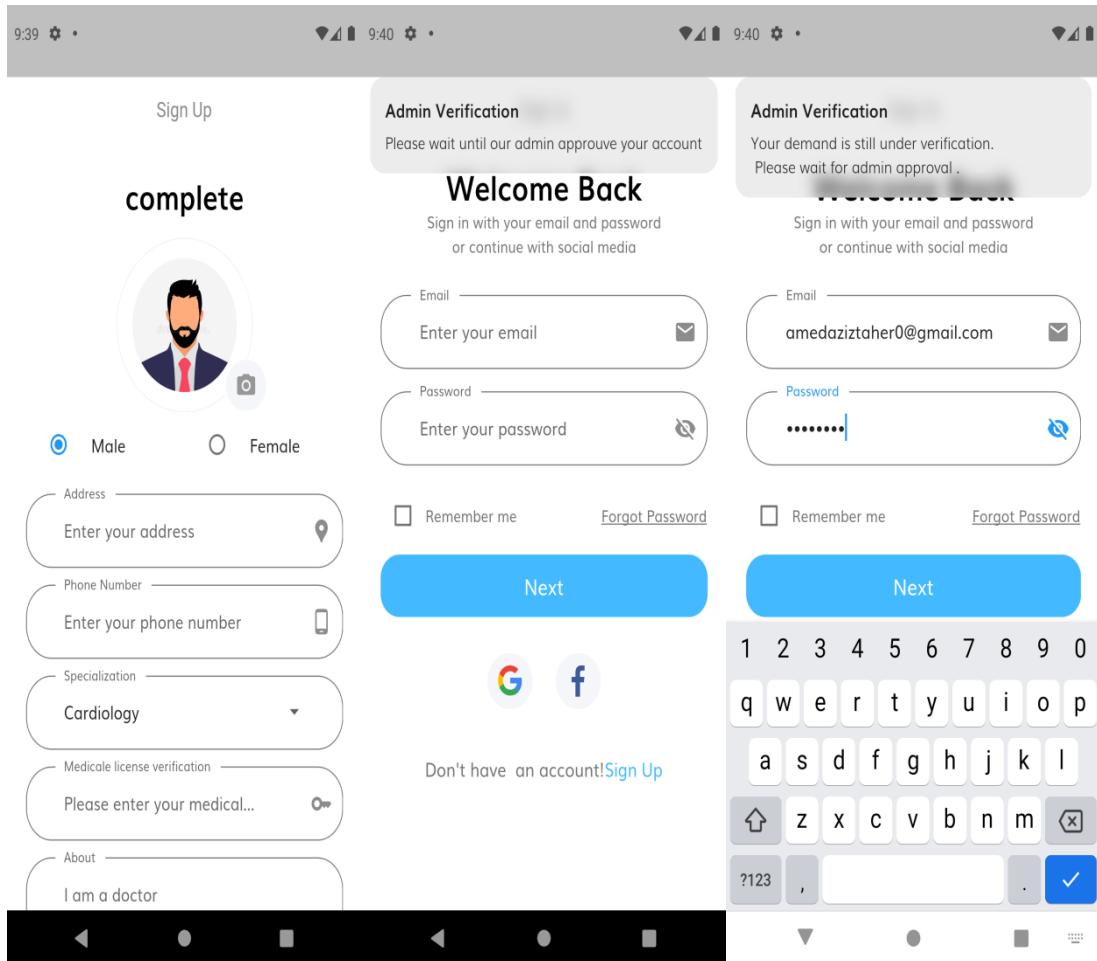


Figure 380: Interface d'inscription

16.1. Sprint 2 : Gestion des demandes d'inscription

Après qu'un fournisseur de soins s'inscrit sur l'application une notification est envoyée vers le compte admin pour valider ou bien supprimer cette inscription :

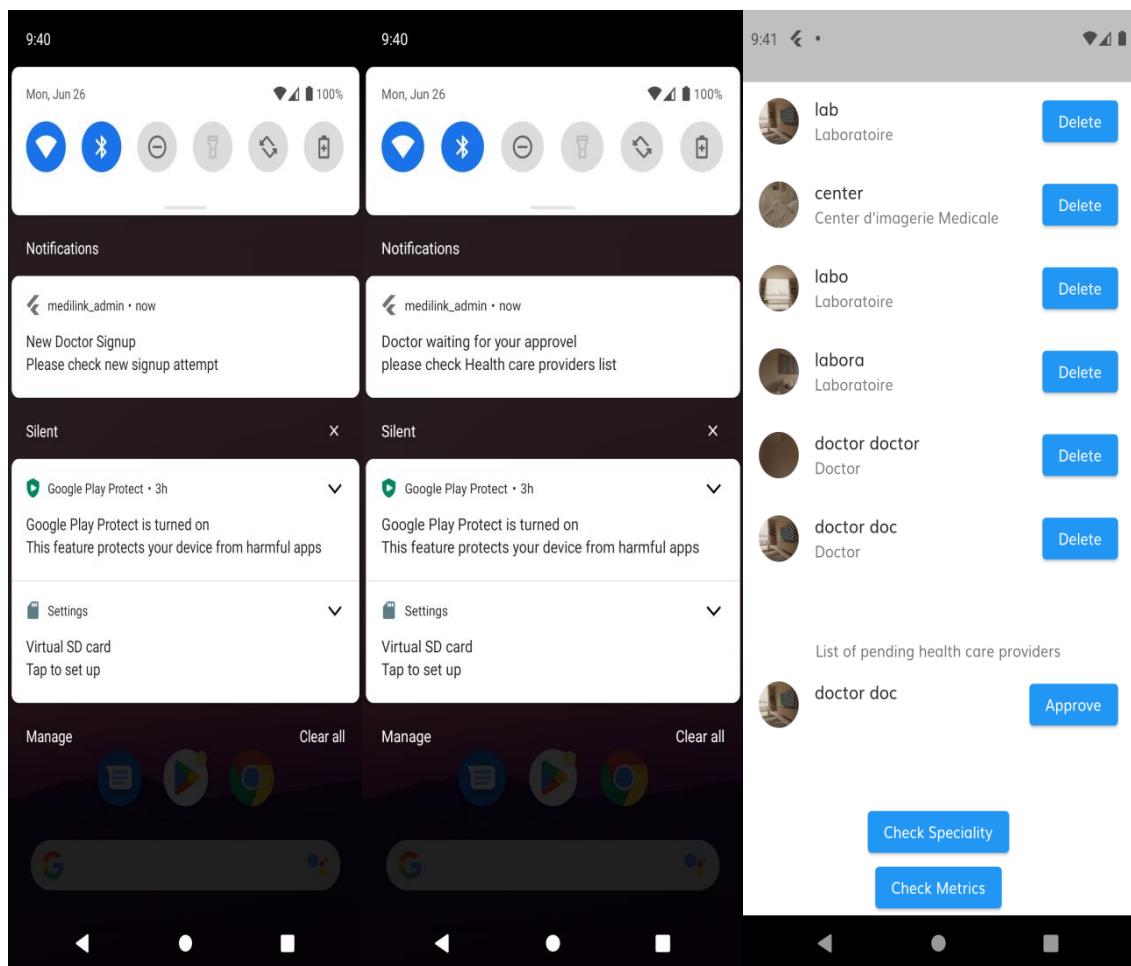


Figure 391: Interface d'inscription

Après la confirmation du fournisseur de soin il va être notifié pour connecter à son compte

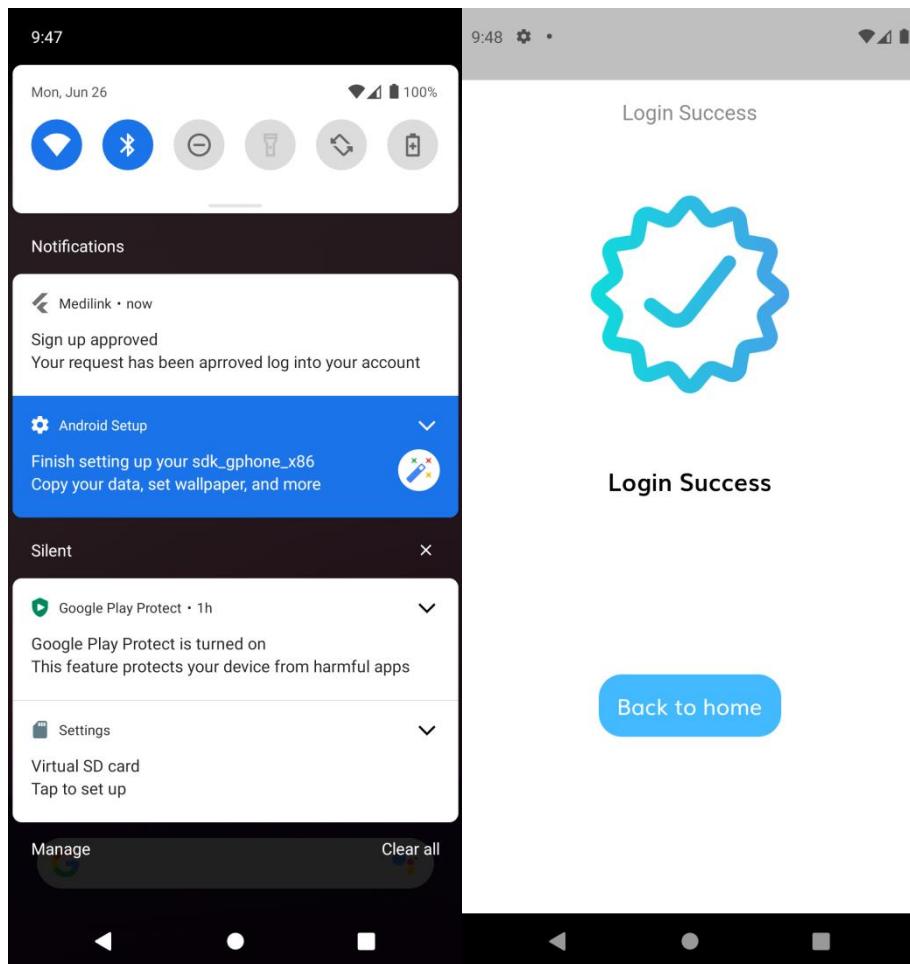


Figure 402: Interface de modification d'un utilisateur

16.2. Sprint 3 : Gestion des demandes de suivi

16.2.1. Gestion des demandes de suivi :

La figure 43 présente la page qui permet au patient de consulter la liste des fournisseurs de soins qui existent sur l'application :

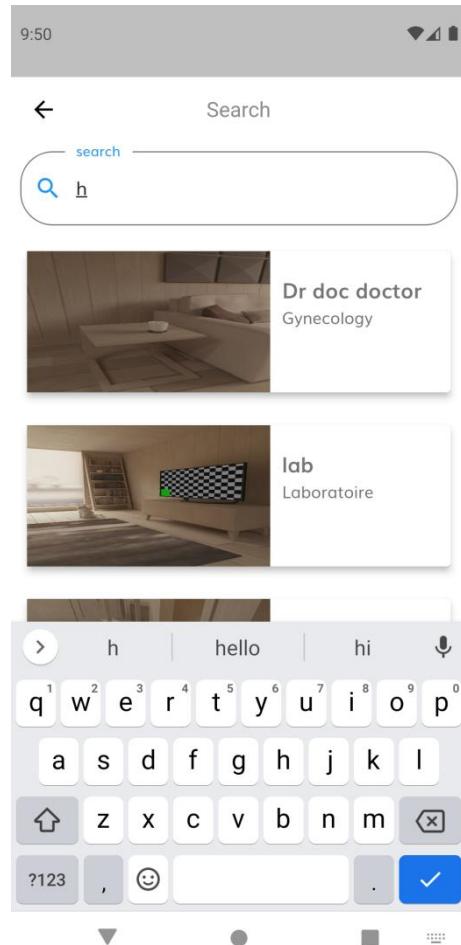


Figure 413: Interface de chercher un fournisseur de soin

Après il consulte le profil du fournisseur désiré comme le montre la figure 44:

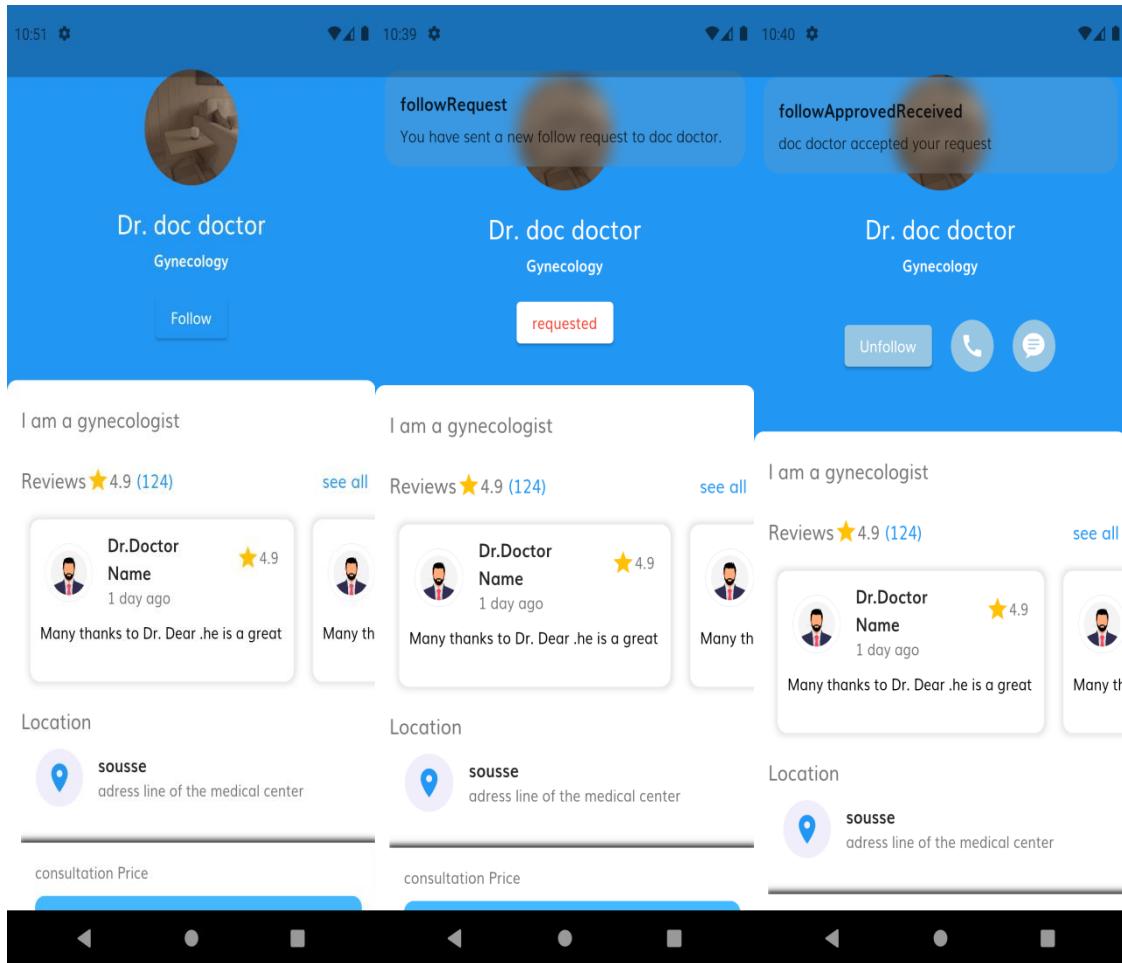


Figure 424: Interface de gestion des invitations du patient

Le fournisseur reçoit une notification d'une nouvelle invitation, il peut consulter cette demande comme montre la figure 45:

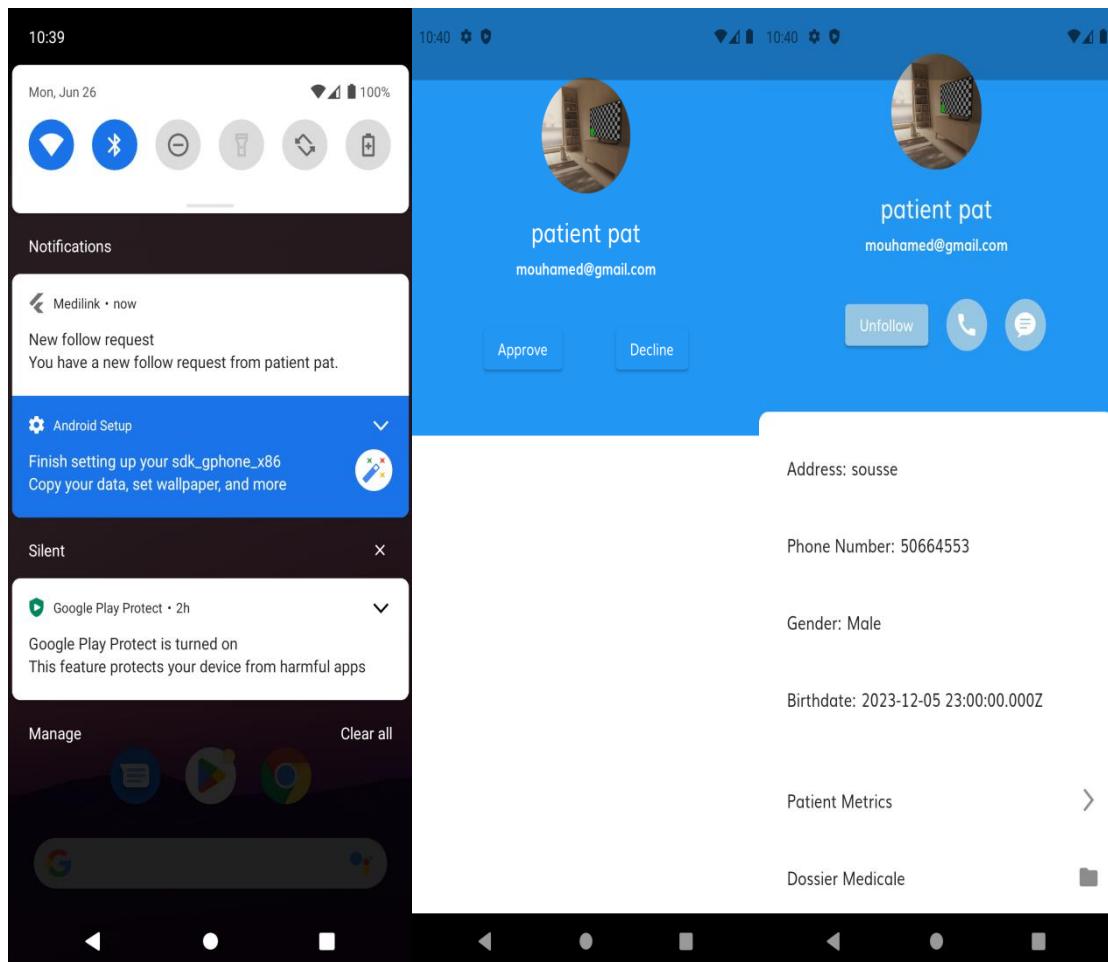
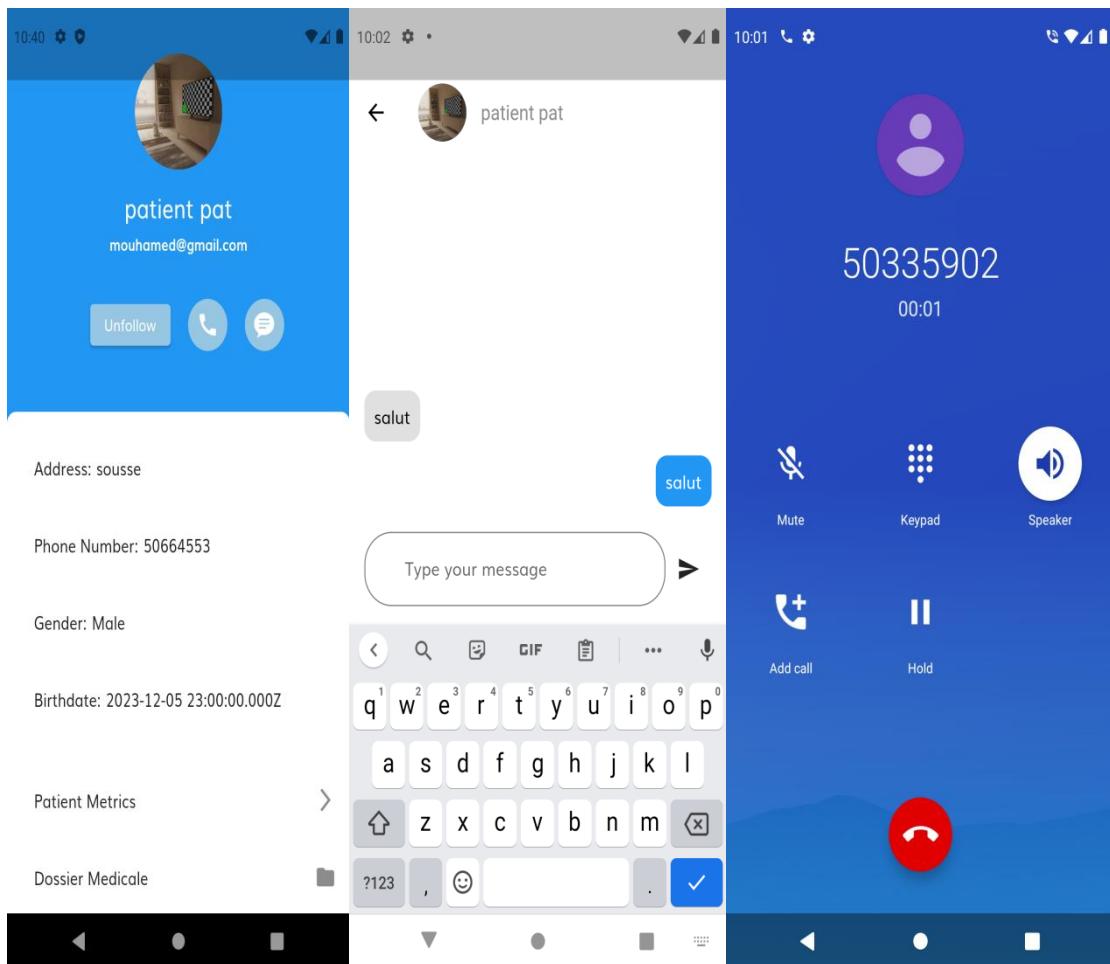


Figure 435: Interface de gestion des demandes de suivi coté fournisseur

16.3. Sprint 4 : Création des conversations en temps réel

Après que le fournisseur accepte la demande de suivi de patient, les deux peuvent commencer à discuter dans une conversation de chat en temps réel ou bien lance un appel téléphonique explique dans la figure 46 :



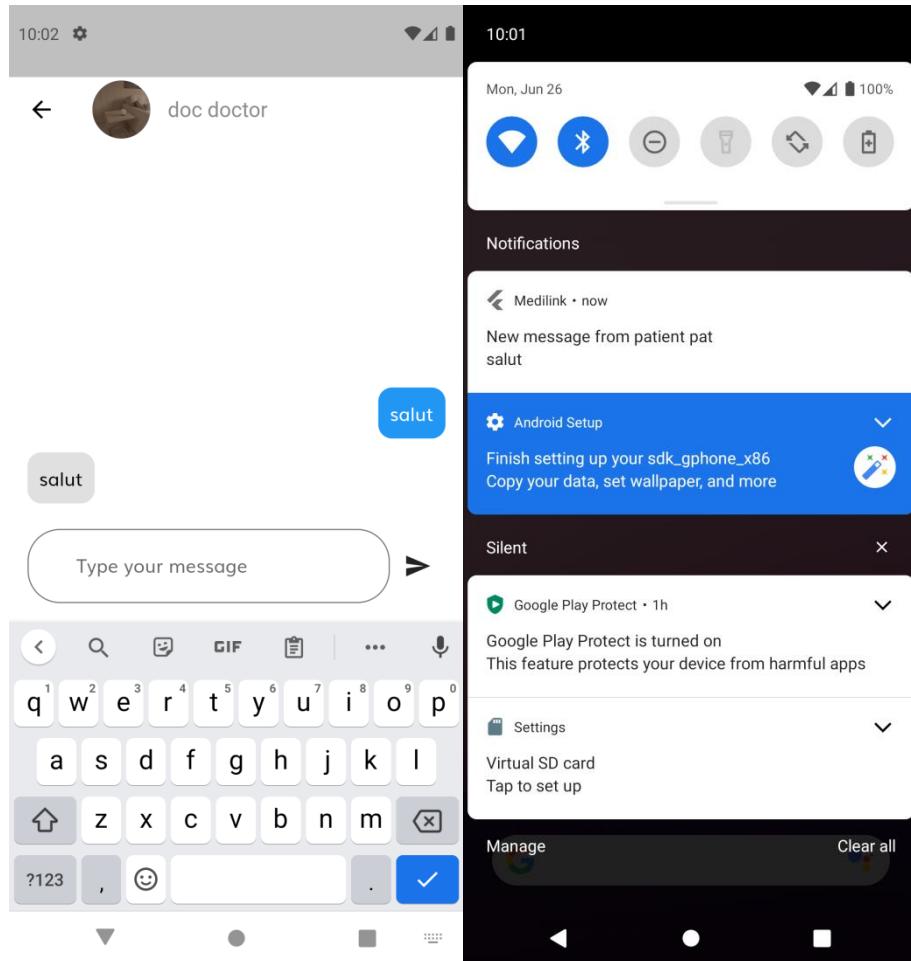
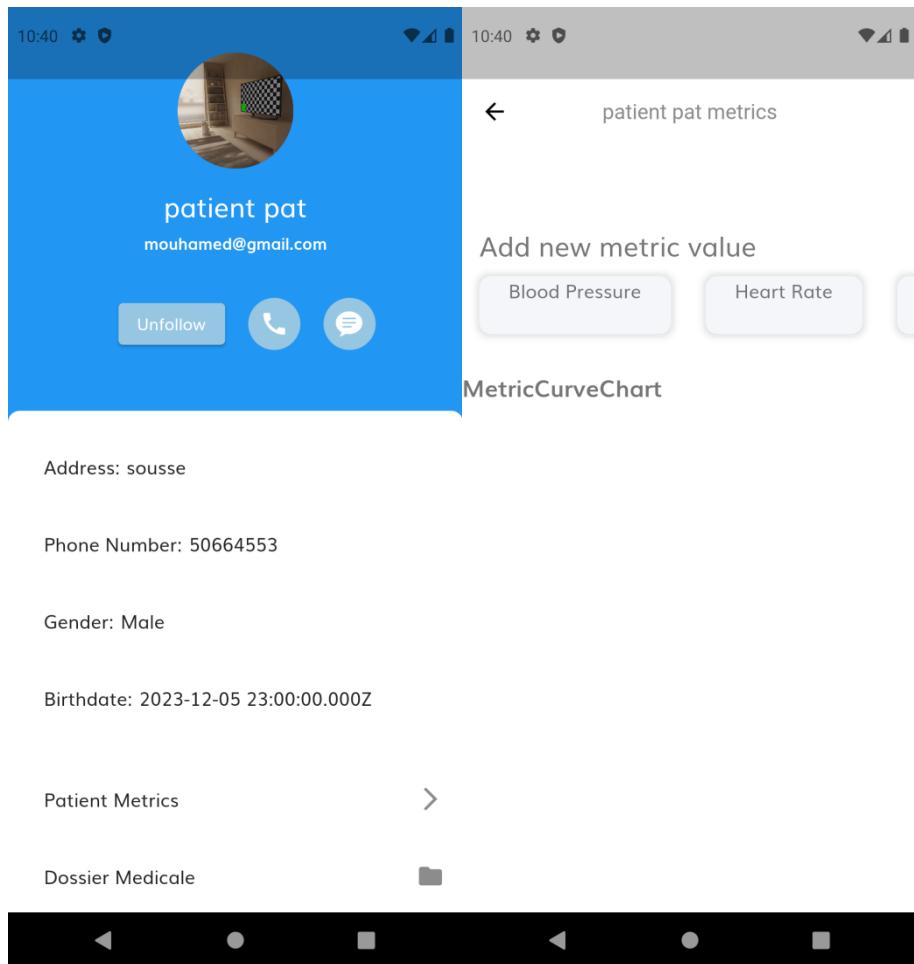
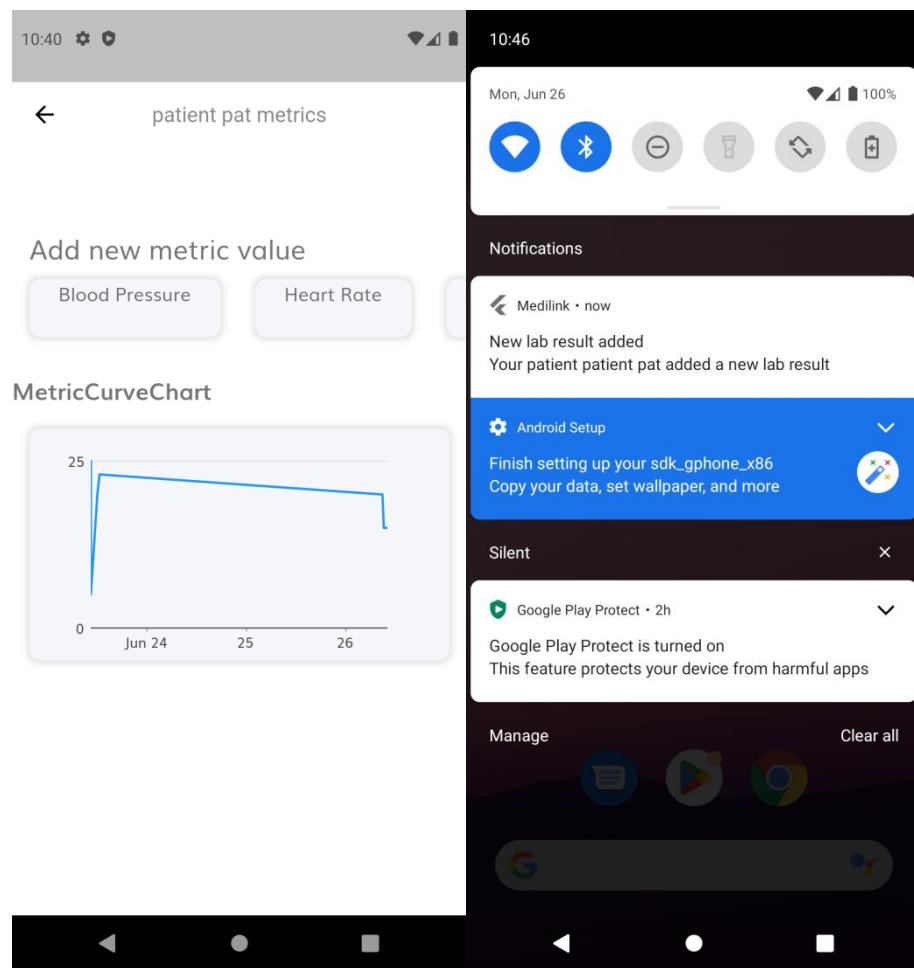


Figure 446: Interface de création des conversations en temps réel

16.4. Sprint 5 : Gestion des dossiers médicaux

Le patient est le seul qui peut tout faire avec son dossier médical. chaqu'un des fournisseurs de soins à des tâches particulières concernant les dossiers médicaux de leurs patients : lorsqu'un médecin accepte un patient, il aura l'accès de voir l'évaluation de ses métriques médicales au cours d'une période et il va être notifié Lorsqu'un fichier médical sera ajouté et partagé par le patient voire figure 47:





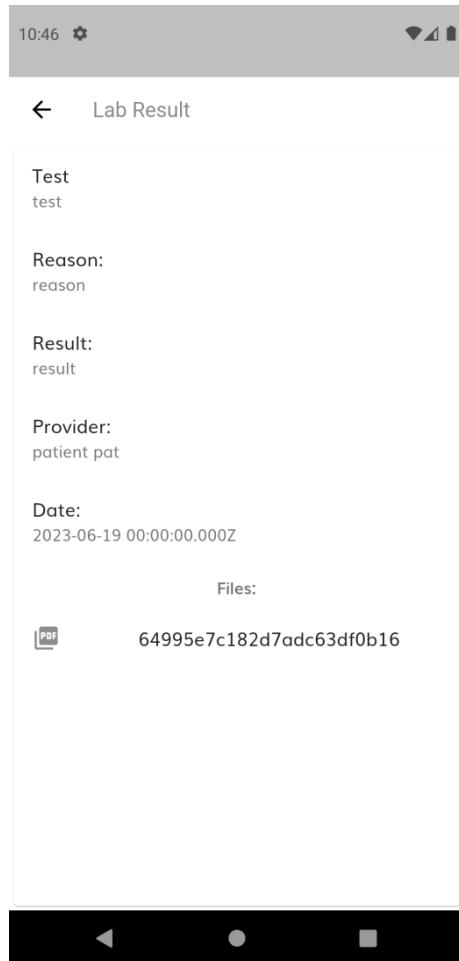
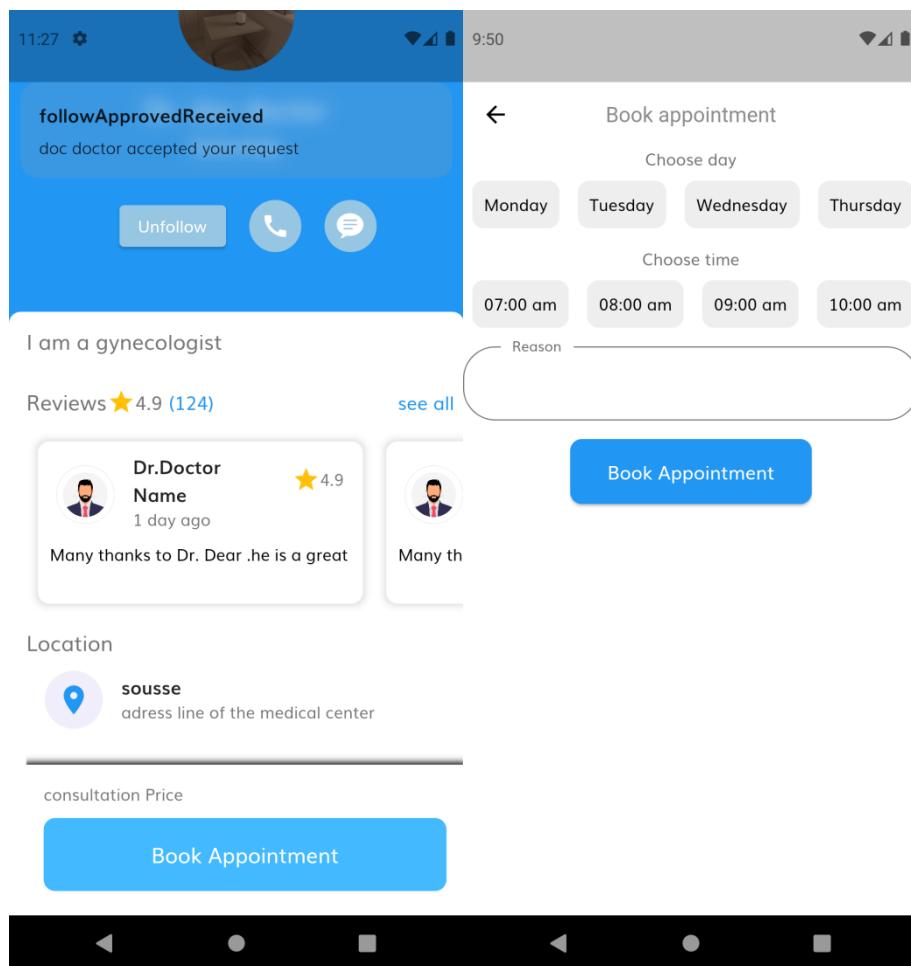
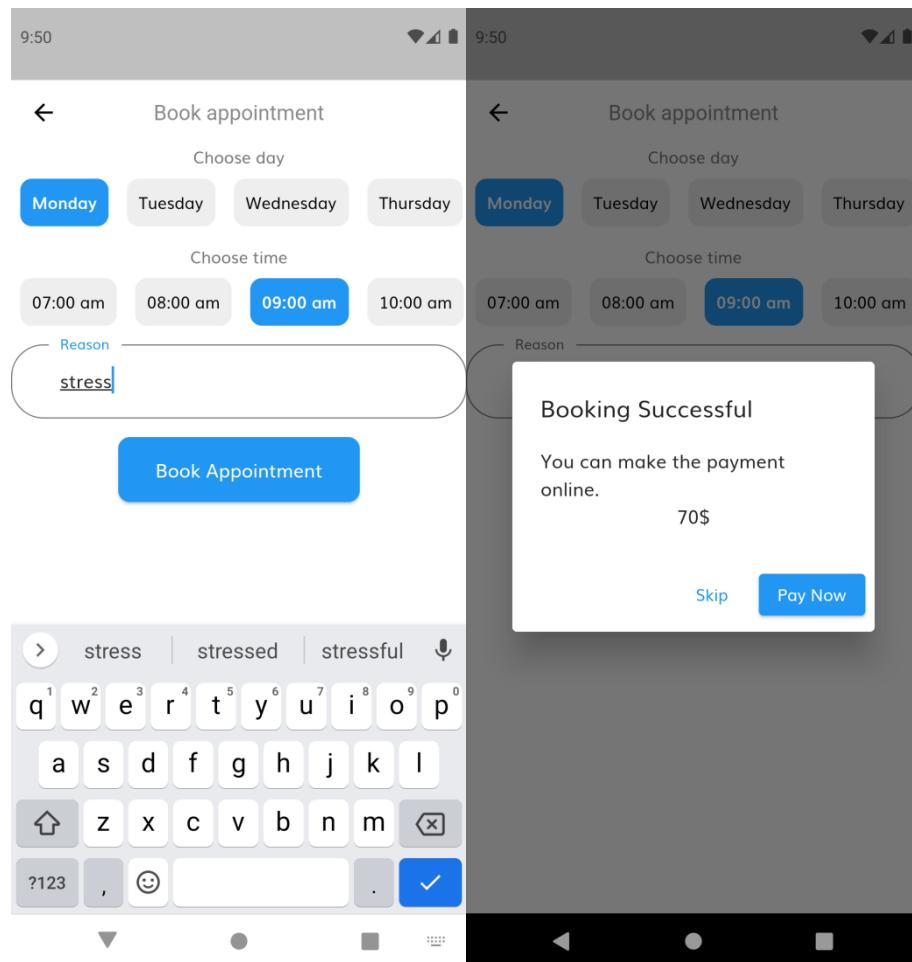


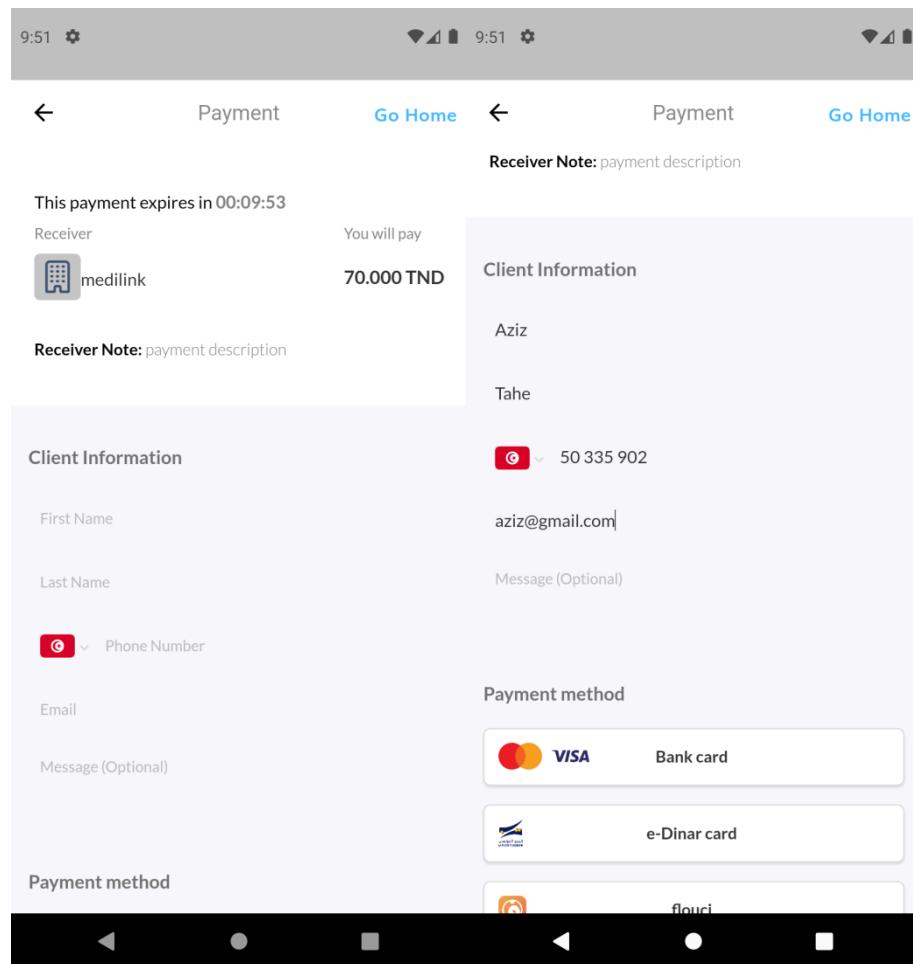
Figure 457: Gestion des dossiers médicales

16.5. Sprint 6 : Paiement en ligne d'un rendez-vous

Après que le patient consulte le profil de son médecin, il peut prendre un rendez-vous et le payer en ligne voir figure 48 :







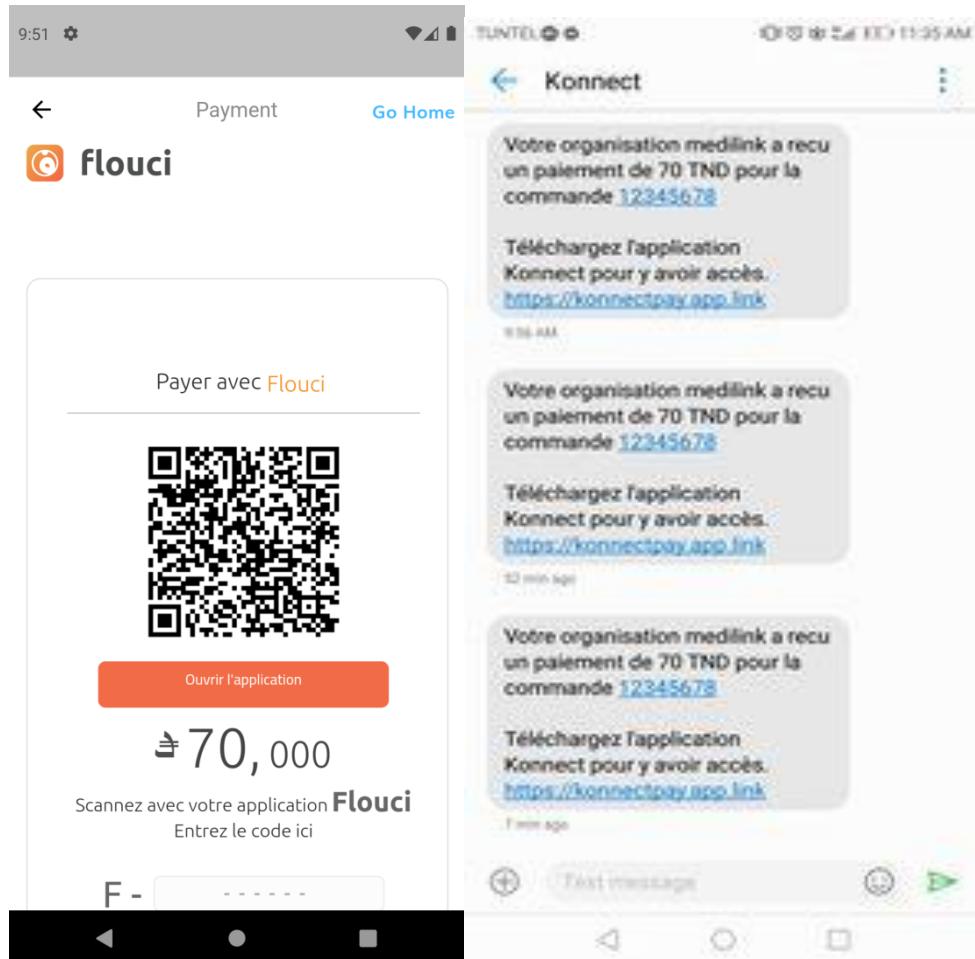


Figure 468: Gestion de paiement en ligne

17. Conclusion

Pour conclure ce dernier chapitre, nous avons présenté l'environnement logiciel de notre application, ainsi que les langages et les technologies utilisés. Ensuite, nous avons décrit en détail les fonctionnalités des interfaces mises en place, en les illustrant avec des captures d'écran et des explications détaillées.

Conclusion et perspectives

Ce projet a été réalisé dans le cadre du stage de fin d'études chez "3D WAVE", une entreprise de développement informatique. Notre objectif principal était de concevoir et mettre en œuvre une application mobile dédiée au suivi de l'état de santé des patients.

Grâce à notre engagement et à l'utilisation de la méthodologie SCRUM, nous avons pu atteindre les objectifs fixés et créer une application fonctionnelle et adaptée aux besoins identifiés.

Notre démarche a débuté par une analyse approfondie des besoins, nous permettant d'identifier les acteurs clés qui interagiront avec l'application. Cette phase nous a permis de préciser les fonctionnalités essentielles et d'établir une vision claire du produit final. Ensuite, nous nous sommes concentrés sur la conception détaillée en utilisant des outils tels que les diagrammes UML. Cela nous a permis de définir l'architecture globale de l'application et de structurer les différents modules et fonctionnalités.

La dernière étape a été consacrée à l'implémentation, au cours de laquelle nous avons utilisé des technologies telles que "Flutter" pour le développement de l'interface utilisateur mobile et "Node.js" pour la partie serveur.

Nous avons travaillé en étroite collaboration avec l'équipe de développement de l'entreprise, en échangeant des idées et en mettant en pratique nos compétences techniques pour réaliser les différentes fonctionnalités de l'application.

Comme perspectives de ce projet, nous avons pour objectif d'intégrer une fonctionnalité d'extraction automatique des données à l'aide de l'intelligence artificielle, ce qui facilitera la collecte et l'organisation des informations médicales des patients. De plus, nous envisageons de mettre en place un chatbot qui fournira des réponses et des conseils aux patients concernant leur état de santé. Cette fonctionnalité permettra aux utilisateurs d'obtenir des informations instantanées et personnalisées. Enfin, nous souhaitons inclure un mécanisme de feedback

permettant à tous les utilisateurs de l'application de communiquer avec l'administrateur en cas de problèmes techniques ou de suggestions d'amélioration. Ces perspectives visent à améliorer l'efficacité et l'accessibilité de l'application tout en renforçant la relation entre les patients et les professionnels de la santé.

Références bibliographiques

- [1]: 3D Wave: <https://www.3dwave.tech>, Date du dernier accès : 25/02 /2023
- [2]:Comparaison deux méthodes:https://www.axiocode.com/methode-agile-vs-classique_quelle-methode-utiliser/,Date du dernier accès : 20/03 /2023
- [3]:Scrum:<https://www.axiocode.com/methode-agile-vs-classique-quelle-methodutiliser/>,Date du dernier accès : 21/03 /2023
- [4]:Use-Case:<https://www.axiocode.com/methode-agile-vs-classique-quelle-methode-utiliser/>, Date du dernier accès: 22/03/2023
- [5]:Architecture-MVC:<https://medium.com/@belcaid.mehdi/larchitecture-logicielle-mvc-1a8bbb5cf6dc>, Date du dernier accès: 23/04/2023
- [6] [7]: Langage UML <https://www.lucidchart.com/pages/fr/langage-uml> , Date du dernier accès: 27/04/2023
- [8]:Node js <https://makina-corpus.com/front-end/introduction-nodejs> ,Date du dernier accès:20/06/2023
- [9]:Flutter <https://aws.amazon.com/fr/what-is/flutter/?fbclid>,Date du dernier accès:20/06/2023
- [10]:MongoDB <https://datascientest.com/mongodb?fbclid> , Date du dernier accès : 25/05/2023
- [11]:Firebase <https://appmaster.io/fr/blog/quest-ce-que-firebase>, Date du dernier accès : 25/05/2023
- [12]:SocketIO <https://socket.io/fr/docs/v4/>, Date du dernier accès : 25/05/2023
- [13]:Visual-studio https://fr.wikipedia.org/wiki/Visual_Studio_Code?fbclid , Date du dernier accès : 25/05/2023
- [14]:Postman [https://fr.wikipedia.org/wiki/Postman_\(logiciel\)?fbclid](https://fr.wikipedia.org/wiki/Postman_(logiciel)?fbclid) , Date du dernier accès : 27/05/2023
- [15]:GitHub <https://datascientest.com/github-tout-savoir> ,Date du dernier accès:27/05/2023
- [16]:StarUML <https://fr.wikipedia.org/wiki/StarUML> , Date du dernier accès: 27/05/2023
- [18]:Ngrok <https://black.bird.eu/fr/blog/ca-roule-entre-magento-et-ngrok>, Date du dernier accès:17/06/2023
- [19]:HTML <https://developer.mozilla.org/fr/docs/Web/HTML> , Date du dernier accès :17/06/2023
- [20]:CSS <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203277-css-cascading-style-sheets-definition-traduction/> , Date du dernier accès:17/06/2023
- [21]:JavaScript <https://developer.mozilla.org/fr/docs/Web/JavaScript> , Date du dernier accès:17/06/2023
- [22]:Dart <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/le-langage-de-programmation-dart/> , Date du dernier accès:17/06/2023

Résumé

Ce travail se situe dans le cadre du projet de fin d'études dans le but d'obtention du Diplôme de la licence nationale en sciences de l'informatique au sein de l'École Supérieure des Sciences et de Technologie Hammam Sousse.

Ce projet consiste à concevoir et réaliser une application mobile médicale pour suivre l'état de santé des patients au sein de la société "3D Wave". Cette application facilite les échanges entre les patients et leurs fournisseurs de soin dans un espace bien sécurisé.

Mots clés : Flutter, Node.Js , MongoDB, JavaScript, Dart, Dossier médical, Patient, Médecin, Santé.

Abstract

This work is part of the Final study project to obtain the Bachelor's Degree in Computer Science within the Higher School of Sciences and Technology Hammam Sousse.

This project involves designing and developing a medical mobile application to monitor the health status of patients within the company "3D Wave." This application facilitates communication between patients and their healthcare providers in a secure environment.

Keywords : Flutter, Node.Js , MongoDB, JavaScript, Dart, Medical record, Patient, Doctor Health.