



Ministère de l'Enseignement Supérieur, de la Recherche
Scientifique

Direction Générale des Études Technologiques

Institut Supérieur des Études Technologiques de Djerba
Département Technologies de l'informatique



Code projet
----------------	-------

Rapport de

STAGE DE FIN D'ETUDES

Présenté en vue de l'obtention du diplôme de

**Licence en Technologies de
l'Informatique**

Parcours: MDW

Gestion des rendez-vous médicaux

Elaboré par :

**Alia Azzabi
Nesrine Boukraa**

Encadré par :

M. Rabii MEJAI

Effectué à :

**Entreprise : AZ DEV
Encadreur : M. Maher El Hor**

Année universitaire : 2023/2024

SOMMAIRE

Introduction générale	1
Chapitre 1 : Contexte Générale	2
Introduction.....	2
1. Présentation de l'organisme d'accueil	2
1.1. La localisation de l'organisme	2
1.2. Les services proposés	3
1.3. Les Logiciels utilisés par AZ Dev	3
1.4. Organisation de l'organisme d'accueil	4
1.4.1. Fiche technique	4
2.2 Structure de la société	4
2. Contexte du projet	5
2.1. Cadre du projet.....	5
2.2 Etude de l'existant.....	5
2.3. Solution proposée.....	6
3. Présentation du projet	7
3.1. Objectifs du projet.....	7
3.2. Méthodologie du projet	7
3.3. Diagramme de GANTT	10
Conclusion	10
Chapitre 2 : Analyse et spécification des besoins.....	11
Introduction.....	11
1. Identification des acteurs.....	11
2. Identification des besoins	13
2.1. Besoins fonctionnels.....	13
2.2. Besoins non fonctionnels	14
3. Modélisation de la solution	15
3.1 Product backlog	15

3.2. Planification du release.....	17
Conclusion :	19
Chapitre 3 : Etude Conceptuel.....	20
Introduction.....	20
1. Description détaillée des cas d'utilisations	20
1.2. Diagramme de cas d'utilisation Global	28
2. Diagramme des classes	29
3. Diagramme des séquences	31
3.1. Diagramme de séquence « Authentification »	31
3.2. Diagramme de séquence Secrétaire et Médecin	32
A)Diagramme de séquence « Création d'un rendez-vous ».....	32
B) Diagramme de séquence « Modification d'un rendez-vous »	33
C) Diagramme de séquence « Supprimer un rendez-vous »	34
Conclusion	34
Chapitre 4 : Réalisation et mise en œuvre	35
Introduction.....	35
1. Environnement de développement et outils de développement	35
1.1. Environnement matériels	35
1.2. Environnement logiciels	36
1.3. Technologie utilisé	38
1.3.1. Base de donné :.....	38
1.3.2. Framework et Langages	38
2. Architecture	43
3. Présentation des interfaces de l'application web	44
3.1. Interface d'authentification.....	44
3.2. Espace de secrétaire.....	45
3.2.1. Interface pour gérer les rendez-vous.....	45
3.2.2. Interface pour gérer les patients	48

3.2.3 Historique d'appel.....	49
3.3. Espace Admin	49
3.3.1. Interface pour gérer les médecins	49
3.3.2. Interface pour gérer les assistants	50
3.3.3. Interface pour gérer les spécialités	51
3.4. Espace médecin.....	51
4. Présentation des interfaces de l'application mobile.....	52
Conclusion	55
Conclusion générale.....	56
WEBOGRAPHIE	56

Liste des figures

Figure 1: La localisation de l'organisme	2
Figure 2: Organigramme de la société	4
Figure 3: Méthode Agile	8
Figure 4: Méthode Scrum.....	8
Figure 5 : Processus de Suivi Scrum.....	9
Figure 6: Diagramme de GANTT.....	10
Figure 7: Fonctionnement de JWT	14
Figure 8 : Diagramme Use Case Assistant & Médecin	21
Figure 9 : Diagramme de cas d'utilisation d'Administrateur	24
Figure 10: diagramme de use case	28
Figure 11: Diagramme des classes.....	30
Figure 12 : Diagramme de séquence "Authentification"	31
Figure 13 : Diagramme de séquence « Ajouter un rendez-vous ».....	32
Figure 14: diagramme de séquence « Modifier rendez-vous »	33
Figure 15 : diagramme de séquence « Supprimer rendez-vous ».....	34
Figure 16: Fiche technique du pc d'Alia	35
Figure 17: fiche technique du pc de Nesrine	36
Figure 18: logo POSTMAN	36
Figure 19: logo Visual Studio Code.....	37
Figure 20: Logo Lucidchart.....	37
Figure 21: MongoDB.....	38
Figure 22: Express.js/ Node.js.....	38
Figure 23: React.js	39
Figure 24: React Native	39
Figure 25: TailwindCSS.....	40
Figure 26: CSS.....	40
Figure 27: Nodemailer	41
Figure 28: Infobip	41
Figure 29: Socket.IO	42
Figure 30 : GitHub.....	42
Figure 31 : Architecture MVC.....	43
Figure 32: interface de login.....	44

Figure 33: Dashboard.....	44
Figure 34 : Interface user Profile	45
Figure 35: Visualisation des rendez-vous par jour	45
Figure 36: Visualisation des rendez-vous par mois	46
Figure 37: Ajouter un rendez-vous	46
Figure 38: Ajouter un patient	47
Figure 39: Modifier un rendez-vous	47
Figure 40: Interface liste des patients	48
Figure 41 : Gérer les demande Rendez-vous.....	48
Figure 42 : Historique d'appel	49
Figure 43: liste des médecins	49
Figure 44: formulaire ajouter un médecin.....	50
Figure 45: interface liste des assistants	50
Figure 46: Interface pour gérer les spécialités.....	51
Figure 47:Interface pour gérer les assistants d'un médecin.....	51
Figure 48: page d'accueil Figure 49: home page	52
Figure 50: interface ajouter rendez-vous.....	53
Figure 51: Consulter l'état des rendez-vous	54

Liste des tableaux

Tableau 1 : Acteurs et leurs rôles	12
Tableau 2 : Backlog Product	17
Tableau 3: Sprint 1	17
Tableau 4 : Sprint 2.....	18
Tableau 5 : Sprint 3.....	18
Tableau 6 : Sprint 4.....	19
Tableau 7 : Descriptif de cas d'utilisation Authentifier	20
Tableau 8 : Descriptif de cas d'utilisation Ajouter patient.....	22
Tableau 9 : Descriptif de cas d'utilisation Ajouter rendez-vous	23
Tableau 10 : Descriptif de cas d'utilisation Ajouter Médecin.....	25
Tableau 11 : Descriptif de cas d'utilisation Ajouter Assistant	26
Tableau 12 : Descriptif de cas Ajouter Spécialité	27

Dédicace

Je dédie ce projet de fin d'études à toutes les personnes qui ont soutenu, encouragé et aidé tout au long de ce parcours.

À mon père, pour sa sagesse et ses conseils précieux. Il a toujours montré l'importance de travailler dur et de rester honnête. Sa foi en moi et sa force m'ont donné la motivation de surmonter les obstacles et de ne jamais abandonner.

À ma mère, pour son amour inconditionnel et sa patience infinie. Elle a été présente dans les moments difficiles, offrant réconfort et encouragement à chaque étape. Son soutien constant et son dévouement m'ont donné le courage de persévérer et de croire en moi-même.

À mes sœurs et à mon frère, pour leur compréhension et leur soutien moral. Leur présence et leurs encouragements ont été une source de motivation et de joie. Ils ont partagé les moments de doute et de célébration, et leur amour et leurs conseils ont aidé à rester concentré et déterminé. Ils ont été des piliers dans les moments difficiles et des compagnons dans les moments de joie.

À mon binôme, Alia Azzabi, Je tiens à exprimer ma profonde gratitude pour sa collaboration et son amitié précieuse tout au long de ce projet. Grâce à notre travail d'équipe, nous avons surmonté de nombreux défis et avons beaucoup appris ensemble.

Merci à chacun pour sa présence et son soutien tout au long de ce parcours. Ce projet est le fruit de nos efforts collectifs autant que du mien. Ensemble, nous avons réalisé quelque chose de significatif, et je suis profondément reconnaissant pour tout ce qui a été fait.

Nesrine Boukraa

Dédicace

Je tiens à exprimer ma profonde gratitude à tous ceux qui ont soutenu et encouragé mes efforts tout au long de ce projet.

À mon père, pilier indéfectible de ma vie, dont la sagesse éclaire chacun de mes pas. Tes conseils avisés ont été ma boussole dans les moments d'incertitude, et ta présence rassurante a toujours été mon refuge.

À ma mère, source intarissable d'amour et de douceur, dont le sourire illumine mes journées les plus sombres. Ta patience infinie et ton soutien inconditionnel ont été mes piliers dans les moments de doute et d'épreuve.

À mes sœurs, compagnes de vie et de rires, dont la présence joyeuse a coloré chacun de mes jours. Votre soutien inébranlable et vos encouragements chaleureux m'ont porté à chaque étape de ce voyage.

À mon frère, complice de mes souvenirs les plus précieux, dont la force et la bravoure ont été mon inspiration. Ta camaraderie indéfectible et ton soutien inépuisable ont été mes forces motrices dans les moments de challenge.

À mon binôme Nesrine Boukraa, complice de mes réussites et de mes défis, dont la collaboration fructueuse a été le socle de notre réussite commune. Ta créativité, ton dévouement et ton esprit d'équipe ont été des atouts inestimables qui ont enrichi notre parcours.

À tous ceux qui ont croisé ma route et ont contribué, de près ou de loin, à cette étape cruciale de mon parcours académique. Votre soutien indéfectible et vos encouragements constants ont été d'une valeur inestimable. Je vous remercie sincèrement pour cette expérience enrichissante et mémorable, qui marque la fin d'une étape et le début d'une nouvelle aventure.

Alia Azzabi

Remerciements

Au terme de ce stage de fin d'études, nous tenons à remercier infiniment toutes personnes ayant contribué à ce travail, tous les enseignants et toute l'équipe pédagogique de l'ISSET Djerba.

Nous tenons à remercier dans un premier temps, toute l'équipe pédagogique de l'Institut Supérieur des Etudes Technologiques de Djerba qui ont contribué à nous donner une formation solide tout au long de nos années d'études.

Un remerciement particulier est adressé à notre encadrant académique, M. **Rabii MEJAI**, pour ses conseils éclairés, son encadrement attentif et son soutien constant durant ce projet.

Nos remerciements s'étendent également aux membres du jury, pour avoir accepté d'évaluer notre travail.

Enfin, nous témoignons notre gratitude à toutes les personnes ayant contribué de près ou de loin à la réalisation de ce travail et au bon déroulement de notre stage, même si leurs noms ne sont pas mentionnés ici.

Introduction générale

Dans le cadre de notre cursus universitaire à l'Institut Supérieur des Etudes Technologiques de Djerba, nous sommes tenus d'effectuer un stage de fin d'études de quatre mois afin de valider nos acquis des six semestres précédents et d'obtenir notre diplôme de licence en sciences et techniques, spécialité MDW. Notre stage s'est déroulé au sein de l'entreprise AZ DEV, du 05février2024 au 24mai2024.

Au cours de cette période, nous avons conçu une application de gestion des rendez-vous pour un cabinet médical, visant à faciliter diverses tâches pour le personnel médical. Nous avons utilisé les technologies suivantes : React.js, Node.js, Express.js, MongoDB TailwindCSS. En parallèle, nous avons également travaillé sur une application mobile destinée aux patients, leur permettant de prendre des rendez-vous en ligne de manière simple et efficace. Pour cette application mobile, nous avons utilisé React Native, CSS, MongoDB, Node.js et Express.js. Ces deux projets complémentaires améliorent significativement l'expérience utilisateur tant pour le personnel médical que pour les patients.

La simplification, la numérisation et la traçabilité de la prise de rendez-vous sont essentielles pour un regroupement de médecins. En effet, ces aspects permettent de réduire les erreurs humaines, d'optimiser la gestion du temps, de faciliter la coordination entre les différents intervenants médicaux et d'améliorer la satisfaction des patients. Une solution numérique intégrée assure une meilleure organisation des plannings, une diminution des délais d'attente et une traçabilité accrue des rendez-vous, ce qui est crucial pour un suivi médical de qualité.

Ce rapport est structuré de la manière suivante : Le premier chapitre aborde le contexte général du projet, présentant l'entreprise d'accueil, l'étude de l'existant et la solution envisagée pour y remédier. Le deuxième chapitre est consacré à l'analyse et à la spécification des besoins, tandis que le troisième chapitre porte sur l'étude conceptuelle de notre projet. Enfin, le troisième chapitre se concentre sur les aspects techniques, incluant les langages de programmation utilisés, les technologies adoptées et l'environnement de développement, ainsi qu'une description des interfaces principales de l'application.

Chapitre 1 : Contexte Générale

Introduction

L'étude d'un projet est une démarche stratégique qui va nous permettre d'avoir une vision globale sur ce dernier visant ainsi à bien organiser le bon déroulement du projet. Cette étude fera donc l'objet du premier chapitre qui sera consacré à la présentation de l'organisme d'accueil, Etude et critique de l'existant et la présentation du projet.

1. Présentation de l'organisme d'accueil

Créée en mars 2019, AZ Dev se présente comme une entreprise unipersonnelle spécialisée dans le développement informatique. Axée sur la satisfaction de ses clients, qu'ils soient des particuliers ou des entreprises, cette entité s'engage à répondre de manière efficiente aux objectifs définis par ces derniers. Grâce à son professionnalisme avéré, AZ Dev bénéficie d'une solide expérience et de compétences avancées, lui permettant de concevoir des solutions informatiques sur mesure. Son savoir-faire s'étend à la création d'outils numériques essentiels, tels que des applications mobiles, des logiciels, etc., adaptés aux diverses activités professionnelles.

1.1. La localisation de l'organisme

Le bureau de la société AZDEV est situé au sein de la pépinière de l'institut supérieur des études technologiques de Djerba.

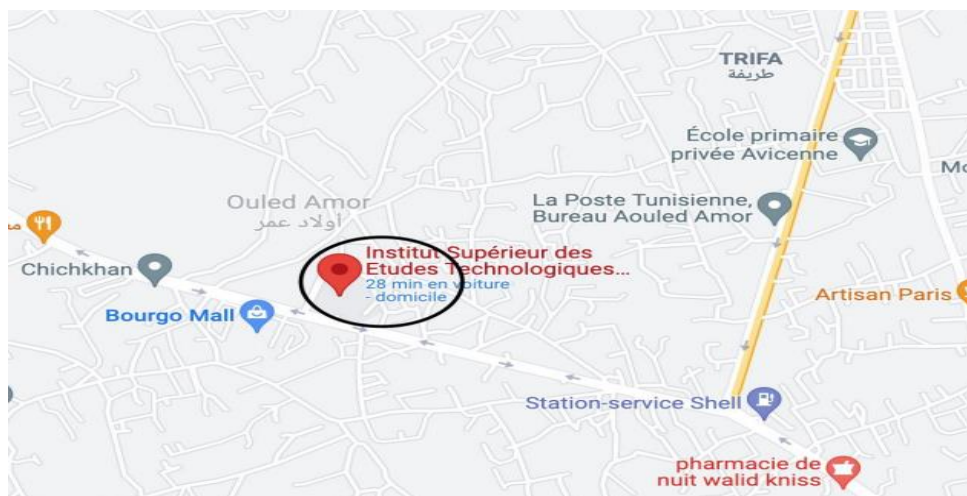


Figure 1: La localisation de l'organisme

1.2. Les services proposés

AZ Dev propose une gamme variée de services professionnels pour répondre aux besoins de ses clients.

-Développement web : AZ Dev dispose d'une équipe expérimentée qui excelle dans le domaine du développement web, créant des sites de qualité grâce à une expertise solide en conception et en développement que ce soit au niveau du front-end ou le back-end.

-Mobile : En matière d'application mobile, l'équipe assure la conception et le développement d'applications robustes pour diverses plateformes, garantissant une expérience utilisateur optimale.

-Développement de logiciels : AZ Dev propose des services de développement logiciel, concevant des solutions personnalisées en étroite collaboration avec les clients, que ce soit pour des projets d'envergure ou des besoins spécifiques d'entreprise.

1.3. Les Logiciels utilisés par AZ Dev

Visual Studio Code : Cet IDE polyvalent a été fréquemment utilisé pour le développement et la programmation, offrant une interface conviviale et des fonctionnalités d'édition avancées.

Visual Studio : Employé pour des projets plus complexes, Visual Studio a permis une approche plus intégrée du développement, offrant des fonctionnalités étendues pour les applications diverses.

Git : Fondamental pour le contrôle de version, Git a été employé pour le suivi des modifications du code source, la gestion des branches, et la collaboration efficace au sein de l'équipe de développement.

Adobe Créative Cloud (Illustration, Photoshop) : Ces outils ont été utilisés pour la conception graphique, la création de maquettes et l'élaboration d'éléments visuels pour les projets web.

Figma : Utilisé pour la conception collaborative et la création d'interfaces utilisateur, Figma a facilité le prototypage et la conception centrée sur l'utilisateur.

1.4. Organisation de l'organisme d'accueil

1.4.1. Fiche technique

- Nom : « AZ DEV »
- Domaine d'activité : Développement informatique
- Adresse : Pépinière Iset Djerba
- Marché local : Développement et maintenance service informatique / Développement des systèmes d'information / Développement des solutions de gestion
- E-mail : maherelhor@gmail.com

2.2 Structure de la société

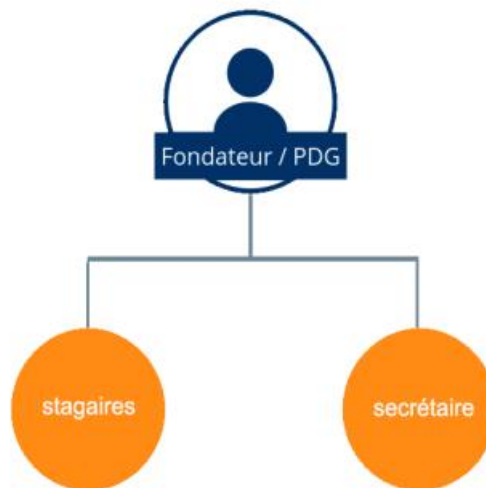


Figure 2: Organigramme de la société

Comme mentionné précédemment, l'organisation du travail repose sur la répartition des tâches entre l'équipe qui effectuée en fonction des différentes phases de production d'un projet, Cette répartition des tâches est réalisée sous la supervision d'un Fondateur pour assurer une coordination efficace entre les équipes et garantir la qualité de chaque phase de production.

2. Contexte du projet

2.1. Cadre du projet

Dans le cadre de notre stage au sein de la société **AZ DEV**, l'équipe nous a confié un projet de développement web. Il s'agit de réaliser une application web pour la gestion de cabinet médical.

Notre application vise à simplifier la gestion des rendez-vous et l'expérience des patients dans le cadre d'une consultation médicale. Elle permet aux patients de planifier leurs rendez-vous en ligne, aux assistants médicaux de valider et de gérer efficacement les plannings et la salle d'attente, et aux médecins de consulter et d'organiser leurs rendez-vous de manière optimale.

2.2. Etude de l'existant

Dans le contexte actuel du cabinet médical, la gestion des rendez-vous repose principalement sur des processus manuels et des outils traditionnels tels que les appels téléphoniques et les calendriers papier. Les secrétaires sont chargées de prendre les appels des patients, de noter les rendez-vous et de les enregistrer dans un calendrier. Cependant, ce mode de fonctionnement présente plusieurs limitations et défis :

Risque d'erreurs humaines : La saisie manuelle des rendez-vous peut entraîner des erreurs de transcription, des doublons de rendez-vous ou des oublis.

Difficulté à fournir des rappels aux patients : Les secrétaires doivent se rappeler de fournir des rappels aux patients par téléphone, ce qui peut être fastidieux et sujet à des oublis.

Manque de visibilité pour les médecins : Les médecins peuvent ne pas avoir accès en temps réel aux informations sur les rendez-vous à venir, ce qui peut entraîner des retards ou des confusions.

2.3. Solution proposée

Pour surmonter ces défis et améliorer la gestion des rendez-vous, nous proposons le développement d'une application de gestion des rendez-vous pour le cabinet médical. Cette application offrira une solution intégrée et automatisée pour les patients, les secrétaires, les médecins et les administrateurs. Voici les principales fonctionnalités de cette solution :

Interface conviviale pour les patients : Les patients pourront facilement prendre rendez-vous en ligne via un portail sécurisé accessible depuis un appareil mobile. Ils pourront consulter les disponibilités en temps réel et choisir un créneau qui leur convient.

Gestion centralisée des rendez-vous : Les secrétaires auront accès à un tableau de bord intuitif où ils pourront voir tous les rendez-vous planifiés. Ils pourront ajouter, modifier ou annuler des rendez-vous en quelques clics, avec la possibilité de fournir des notes ou des instructions spécifiques.

Notification automatisée : L'application enverra automatiquement des rappels par e-mail ou SMS aux patients pour leurs rendez-vous, ainsi que des notifications en cas de changement ou de retard. Cela réduira le nombre de rendez-vous manqués et améliorera la ponctualité des patients.

Gestion des utilisateurs et des autorisations : Les administrateurs auront un contrôle total sur les comptes des utilisateurs, avec la possibilité de créer, modifier ou supprimer des comptes, ainsi que de définir des autorisations d'accès spécifiques en fonction des rôles et des responsabilités.

Analyse et suivi des performances : L'application fournira des outils d'analyse avancés permettant aux administrateurs de suivre les performances globales du cabinet médical, de générer des rapports sur les rendez-vous pris, annulés ou reportés, et d'identifier les tendances pour une prise de décision éclairée

3. Présentation du projet

3.1. Objectifs du projet

Le projet vise à créer une application innovante de gestion des rendez-vous pour un cabinet médical, destinée à améliorer l'expérience des patients et à optimiser les opérations internes. Cette application permettra aux patients de prendre des rendez-vous en ligne de manière autonome, tandis que les assistants médicaux pourront valider les demandes et gérer efficacement les plannings et la salle d'attente. Les médecins, de leur côté, auront une vue claire et organisée de leurs consultations à venir, facilitant ainsi la planification et la gestion de leur emploi du temps. Grâce à des notifications par e-mail et SMS, les patients seront informés en temps réel de tout changement ou retard.

3.2. Méthodologie du projet

Les méthodes Agile :

Dans le domaine du développement, les méthodologies agiles se sont imposées comme une solution efficace pour gérer les projets de manière flexible et itérative. Les méthodes agiles, telles que Kanban, Extrême Programming (XP), Lean Development et bien d'autres, mettent l'accent sur la collaboration étroite avec les parties prenantes, la livraison fréquente de versions fonctionnelles du logiciel, et l'adaptation rapide aux changements. Ces méthodologies favorisent également l'amélioration continue grâce à des cycles de développement courts et à des feedbacks réguliers des utilisateurs.

Parmi ces méthodologies, Scrum est l'une des plus populaires et largement adoptées dans l'industrie du logiciel. Scrum propose une approche structurée et itérative qui divise le projet en sprints, des périodes de travail courtes généralement de deux à quatre semaines. Cette méthodologie facilite la planification, l'exécution et le suivi des tâches tout en assurant une communication fluide au sein de l'équipe.

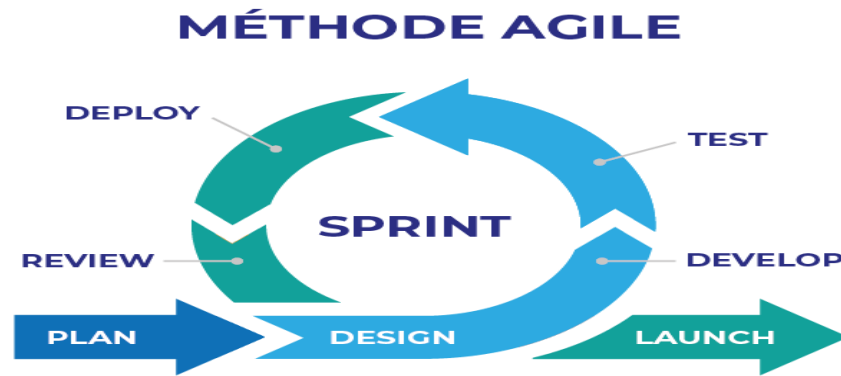


Figure 3: Méthode Agile

La méthodologie adoptée SCRUM :

Nous avons choisi de travailler avec la méthodologie Scrum pour notre projet de gestion des rendez-vous du cabinet médical. Scrum nous offre une approche structurée et itérative qui favorise la flexibilité et la collaboration entre les membres de l'équipe. En divisant le projet en sprints, nous pouvons développer et livrer des fonctionnalités de manière progressive, tout en recevant et intégrant régulièrement les retours des utilisateurs. Les réunions quotidiennes et les revues de sprint permettent de maintenir une communication fluide et d'assurer une amélioration continue, ce qui est essentiel pour répondre aux exigences changeantes et garantir la réussite du projet.

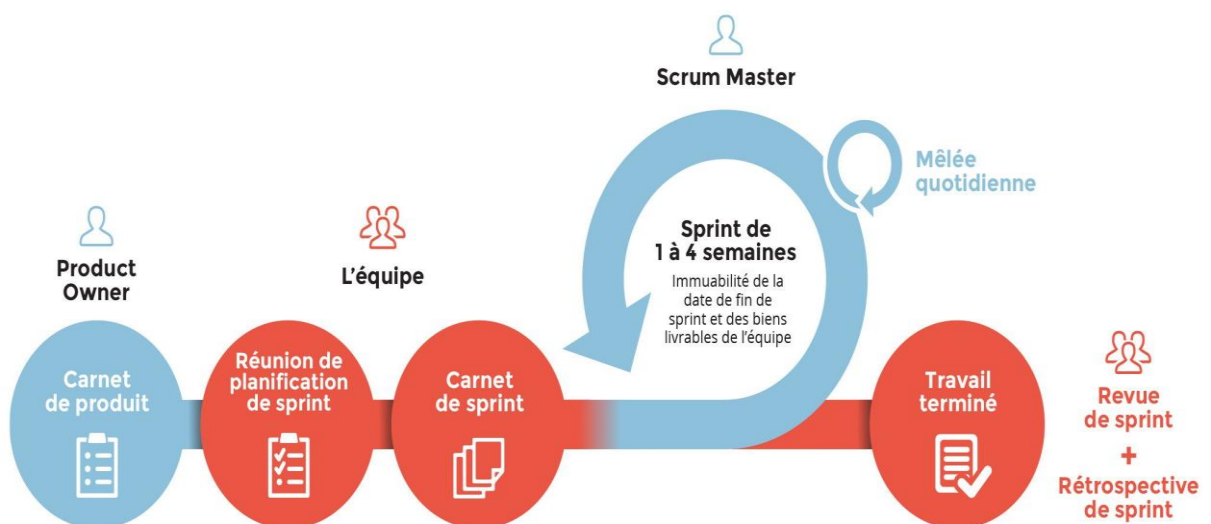


Figure 4: Méthode Scrum

Processus de Suivi de méthode SCRUM :

Pendant chaque Sprint, les membres de l'équipe se réunissent quotidiennement pour discuter de l'avancement des tâches en cours, des défis rencontrés et des tâches restantes à accomplir. Une fois les livrables partiels prêts, nous passons en revue ce qui a été réalisé durant le Sprint afin de vérifier sa conformité et d'identifier des opportunités d'amélioration lors de la rétrospective.

- **Product Owner** : Il est chargé de définir les caractéristiques et les exigences du produit, de prioriser le backlog des fonctionnalités et de prendre les décisions relatives au produit.
- **Scrum Master** : Son rôle est de guider l'équipe Scrum, de veiller à ce que les principes et les processus Scrum soient respectés, et de supprimer les obstacles pouvant entraver la progression du projet.
- **Équipe de développement** : Composée des membres responsables de la réalisation du produit, cette équipe est autonome et se charge d'organiser son travail pour accomplir les tâches assignées lors de chaque sprint.

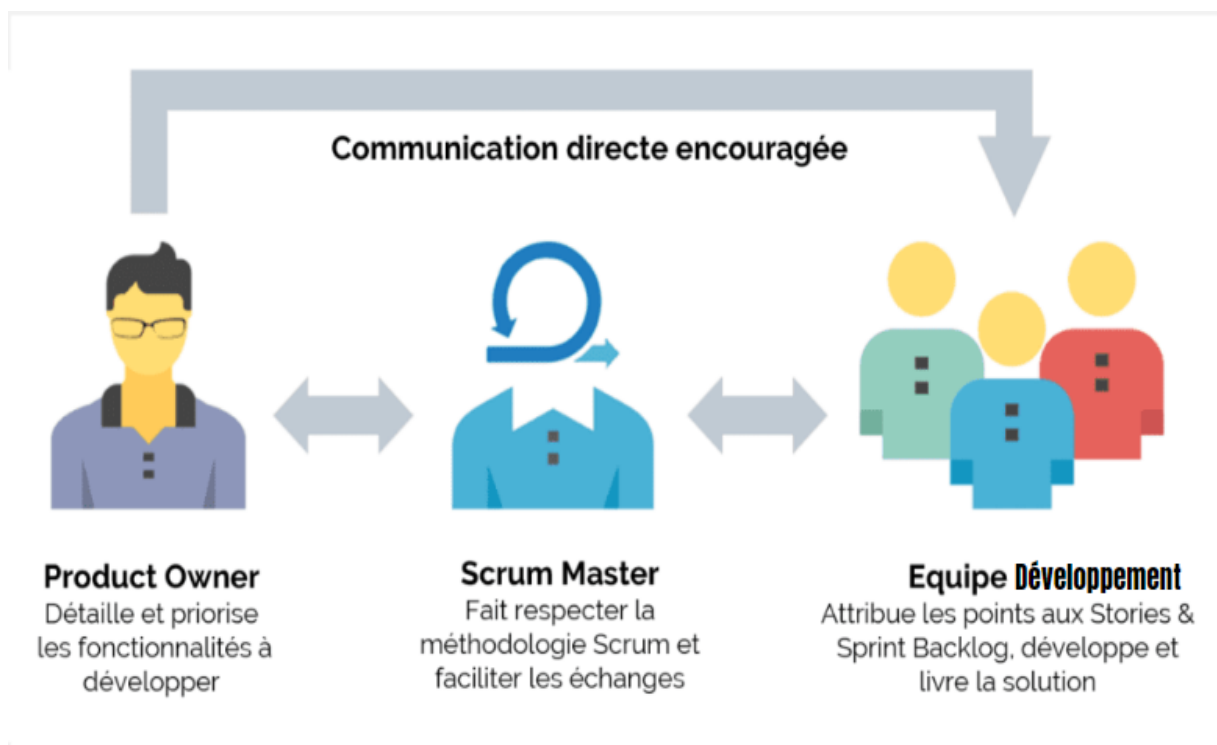


Figure 5 : Processus de Suivi Scrum

3.3. Diagramme de GANTT

Le diagramme de Gantt est un outil de gestion de projet visuel qui permet de planifier, suivre et gérer les différentes tâches et phases d'un projet. Dans le cadre de notre projet de développement d'une application web pour la gestion d'un cabinet médical, le diagramme de Gantt détaillera les étapes clés, les activités à réaliser, les délais associés et les dépendances entre les tâches.



Figure 6: Diagramme de GANTT

Conclusion

En conclusion, ce premier chapitre nous a permis de poser les bases du projet en présentant l'organisme d'accueil, AZ Dev, ainsi que les défis actuels rencontrés dans la gestion des rendez-vous au sein du cabinet médical. Nous avons examiné les limites des processus existants et proposé une solution intégrée et automatisée sous forme d'une application de gestion des rendez-vous. La méthodologie Scrum, choisie pour mener à bien ce projet, offre une approche structurée et flexible, facilitant une collaboration efficace et une adaptation rapide aux besoins changeants.

Chapitre 2 : Analyse et spécification des besoins

Introduction

Dans ce chapitre dédié à l'analyse et à la spécification des besoins pour notre projet de gestion des rendez-vous du cabinet médical, nous avons entrepris une démarche méthodique visant à comprendre les exigences fonctionnelles et non fonctionnelles de notre système, ainsi que les acteurs qui interagiront avec lui

1. Identification des acteurs

Dans cette section, nous allons identifier et décrire les différents acteurs impliqués dans le projet de gestion des rendez-vous pour le cabinet médical. Ces acteurs jouent des rôles essentiels et variés, chacun contribuant à la réalisation et au bon fonctionnement de l'application. Une compréhension claire des rôles et des responsabilités de chaque acteur est cruciale pour répondre aux besoins de tous les utilisateurs et pour assurer le succès du projet.

Les patients :

Les patients interagissent avec l'application principalement pour demander des rendez-vous via une application mobile. Les patients peuvent consulter les créneaux horaires disponibles, choisir un créneau qui leur convient et soumettre leur demande de rendez-vous. Leur expérience utilisateur doit être intuitive et fluide pour faciliter la prise de rendez-vous en ligne.

Les assistants médicaux :

Les assistants médicaux jouent un rôle central dans la gestion des rendez-vous. Ils sont responsables de valider les demandes de rendez-vous soumises par les patients et de gérer les plannings des médecins. Ils doivent s'assurer que les rendez-vous sont correctement enregistrés et que toute modification ou annulation est bien communiquée aux patients.

Les médecins :

Les médecins consultent et organisent leurs rendez-vous à l'aide de l'application. Ils ont besoin d'une vue claire et organisée de leur emploi du temps pour planifier leurs consultations de manière optimale. Ils doivent également pouvoir accéder aux informations des patients et effectuer des tâches administratives si nécessaire.

Les administrateurs :

Les administrateurs sont chargés de la gestion globale de l'application et des utilisateurs. Ils créent, modifient et suppriment les comptes des médecins et des assistants. Ils assignent des rôles et des autorisations spécifiques à chaque utilisateur en fonction de leurs responsabilités et suivent l'activité globale du centre médical grâce à des outils de reporting et de statistiques.

→ Conclusion

Utilisateur	Leur Rôle
Assistants médicaux	<ul style="list-style-type: none">♣ Gestion des rendez-vous.♣ Gestion des patients.♣ Gestion de la comptabilité.♣ Gestion de salle d'attente.
Médecin	<ul style="list-style-type: none">♣ Gestion des rendez-vous.♣ Gestion des patients.♣ Gestion de la comptabilité.
Admin	<ul style="list-style-type: none">♣ Gestion des secrétaires.♣ Gestion des médecins.♣ Plus les rôles de médecin♣ Gérer les spécialités
Patient	<ul style="list-style-type: none">♣ Prendre rendez-vous.♣ Consulter état rendez-vous.

Tableau 1 : Acteurs et leurs rôles

2. Identification des besoins

Pour développer une application de gestion des rendez-vous du cabinet médical qui réponde aux attentes de tous les acteurs impliqués, il est crucial de définir les besoins fonctionnels et non fonctionnels. Les besoins fonctionnels décrivent ce que l'application doit accomplir, tandis que les besoins non fonctionnels se concentrent sur la qualité et les contraintes du système.

2.1. Besoins fonctionnels

Gestion des rendez-vous :

- Les patients doivent pouvoir prendre rendez-vous en ligne.
- Les secrétaires doivent pouvoir valider, modifier ou annuler les rendez-vous, y compris ceux pris par téléphone.
- Les médecins doivent pouvoir consulter leur planning et organiser les rendez-vous.

Notification :

- Mise en place d'un système de notification par e-mail et SMS pour informer les patients des changements ou des retards de leur rendez-vous.

Administration des utilisateurs :

- Création, modification et suppression des comptes pour les médecins, les assistants et les administrateurs.
- Attribution des rôles et des autorisations en fonction des responsabilités de chaque utilisateur.

Gestion des spécialités médicales :

- Possibilité pour l'administrateur de gérer les différentes spécialités médicales proposées par le cabinet.

Statistiques et suivi :

- Suivi de l'activité globale du centre médical à travers des statistiques (nombre de rendez-vous, taux de présence, etc.)

2.2. Besoins non fonctionnels

Sécurité :

Le système doit être sécurisé au niveau des données : authentification et contrôle des droits d'accès. Pour cela nous utiliserons le jeton de sécurité web (Json Web Token ou tout simplement JWT) qui est un standard pour échanger de l'information de manière sécurisée via un jeton signé. Par exemple un serveur pourrait émettre un jeton possédant l'affirmation "utilisateur identifié en tant qu'administrateur" et le fournir au client. Le client pourrait alors vérifier le jeton pour prouver que l'utilisateur est identifié en tant qu'administrateur. Le jeton JWT est composé de trois parties :

- Un entête : indique quel algorithme a été utilisé pour générer la signature
- L'information : variable en fonction de l'application
- Une signature : comprend une clé, un jeton et la signature effective

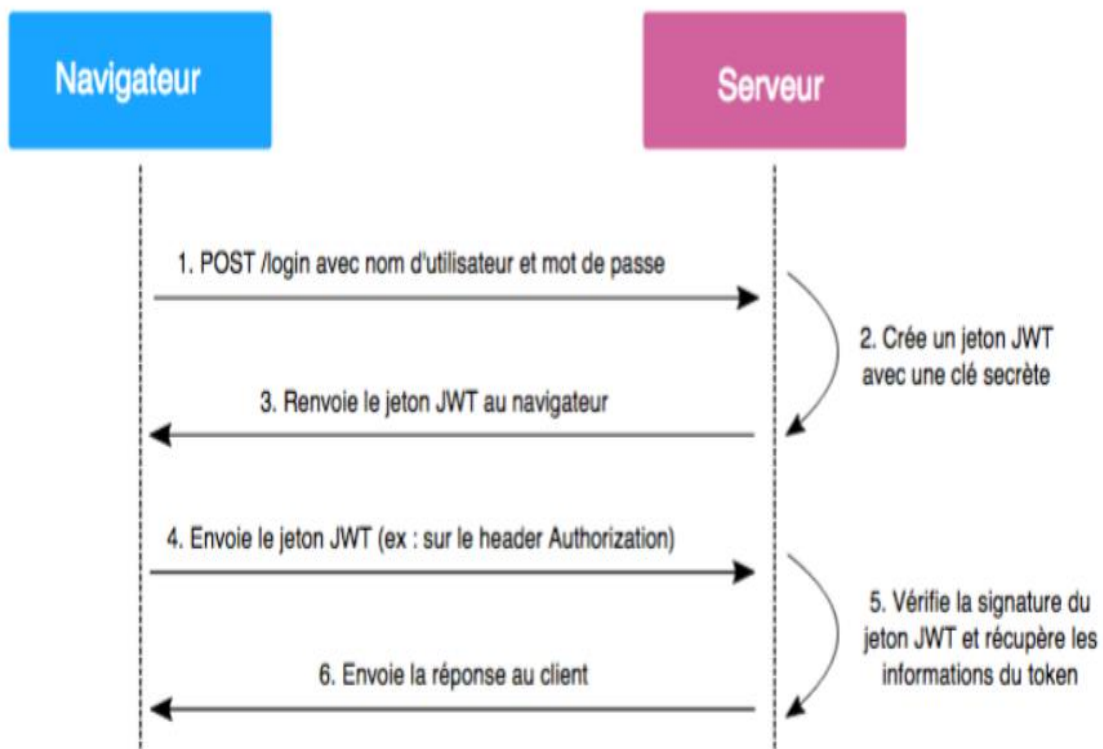


Figure 7: Fonctionnement de JWT

Compatibilité multiplateforme :

- Développement d'applications web et mobiles pour permettre l'accès depuis différents appareils (ordinateurs, Smartphones, tablettes).

Convivialité :

- Interface utilisateur intuitive et conviviale pour faciliter son utilisation par les administrateurs, les assistants médicaux et les médecins.

Maintenabilité :

- Code source bien documenté et structuré pour faciliter la maintenance et les mises à jour futures.
- Utilisation de technologies et de Framework standards pour assurer la pérennité de l'application.

3. Modélisation de la solution

3.1 Product backlog

Le Product Backlog est une liste priorisée et évolutive de toutes les fonctionnalités, améliorations et corrections nécessaires pour le développement de notre application de gestion des rendez-vous pour un cabinet médical. Ce projet vise à simplifier et automatiser les tâches administratives, tout en améliorant l'expérience utilisateur pour le personnel médical et les patients. Le Product Backlog est l'outil central de gestion des exigences du projet, offrant une vision claire et structurée des éléments à développer, leur priorité et leur état d'avancement.

Id	User Story	Description
1	En tant qu'utilisateur, je souhaite me connecter à la plateforme	En tant qu'utilisateur, je veux pouvoir me connecter à la plateforme en utilisant mes identifiants afin d'accéder à mes dispositifs et gérer mes paramètres.
2	En tant que administrateur, je souhaite enregistrer un médecin dans la plateforme	En tant que administrateur, je veux pouvoir enregistrer un médecin dans la plateforme en fournissant les informations requises telles que le nom, l'adresse e-mail et le mot de passe.

3	En tant que administrateur, je souhaite enregistrer un assistant dans la plateforme	En tant que administrateur, je veux pouvoir enregistrer un assistant dans la plateforme en fournissant les informations requises telles que le nom, l'adresse e-mail et le mot de passe.
4	En tant qu'administrateur, je souhaite gérer les profils utilisateur	En tant qu'administrateur, je veux pouvoir gérer les profils utilisateur en ajoutant, modifiant ou supprimant des informations telles que les noms, les adresses, les numéros de téléphone, etc., afin de maintenir des profils précis et à jour.
5	En tant qu'administrateur, je souhaite gérer les spécialités	En tant qu'administrateur, je veux pouvoir gérer les spécialités en ajoutant, modifiant ou supprimant une spécialité.
6	En tant qu'assistant médical, je souhaite gérer les patients	En tant qu'assistant, je veux pouvoir d'ajouter un patient ou Accéder aux patients pour effectuer des tâches administratives telles que la mise à jour des informations personnelles (par exemple : modifier l'adresse ou le numéro de téléphone d'un patient).
7	En tant qu'assistant médical, je souhaite gérer les rendez-vous.	En tant qu'assistant, je veux Aider à la planification des rendez-vous en fonction des disponibilités des médecins (par exemple : ajouter un rendez-vous en spécifiant la date, l'heure et le médecin concerné, Modifier ou annuler les rendez-vous si nécessaire, en coordination avec le médecin.)
8	En tant qu'assistant médical, je souhaite envoyer des notifications aux patients.	En tant qu'assistant, je veux notifier les patients des retards ou des changements éventuels dans leur rendez-vous.
9	En tant que médecin, je veux gérer mon équipe d'assistants	En tant qu'médecin, je veux gérer mon équipe d'assistants médicaux, y compris l'ajout, la suppression et la modification des assistants.

10	En tant que médecin médical, je souhaite gérer les rendez-vous.	En tant qu'médecin, je veux Planifier des rendez-vous pour les patients directement (par exemple : programmer un rendez-vous de suivi après une consultation).
11	En tant que médecin médical, je souhaite gérer les patients.	En tant que médecin, je veux pouvoir d'ajouter un patient ou Accéder aux patients pour effectuer des tâches administratives telles que la mise à jour des informations personnelles (par exemple : modifier l'adresse ou le numéro de téléphone d'un patient).

Tableau 2 : Backlog Product

3.2. Planification du release

Dans notre approche de planification, chaque sprint est fixé à une durée de 4 semaines.

Sprint 1 : du 05/02/2024 au 29/02/2024

Elément du backlog	Taches
Analyse	Etude de l'existant
	Solution proposé
	Analyse des besoins
Préparation du cahier de charge	Définition des objectifs
Apprentissage des technologies (React et Node.js)	Acquérir des connaissances de base à travers des tutoriels et des cours en ligne.
Mise en place de l'environnement de suivi de taches (Jira)	Définir les types de tâches, les statuts de workflow et les tableaux de bord nécessaires pour suivre le progrès du projet.

Tableau 3: Sprint 1

Sprint 2 : du 04/03/2024 au 29/03/2024

Elément du backlog	Taches
Apprentissage de React Native	Créer une application « Hello world » simple pour comprendre les principes de base.
Conception	Faire le diagramme de cas d'utilisation
	Faire le diagramme de classe
	Faire le diagramme de séquences
Choix du Template et Configuration Initiale	Recherche et sélection d'un Template adapté pour le site web.
	Mise en place de la structure de base du projet (pages principales, Dashboard..).

Tableau 4 : Sprint 2

Sprint 3 : du 01/04/2024 au 26/04/2024

Elément du backlog	Taches
S'authentifier	Vérification de l'authentification et de l'autorisation.
Gérer les rendez-vous	Visualiser le calendrier des rendez-vous.
	Créer un rendez-vous.
	Annuler un rendez-vous.
	Modifier un rendez-vous.
Réaliser les opérations CRUD (Créer, Lire, modifier, Supprimer)	Pour les médecins.
	Pour les patients.
	Pour les aides.
	Pour les spécialités.
Fonctionnalité de notification	Mettre en place une fonctionnalité de notification par e-mail.
	Mettre en place une fonctionnalité de notification par SMS.

Tableau 5 : Sprint 3

Sprint 4 : du 29/04/2024 au 24/05/2024

Elément du backlog	Taches
Suivi des statistiques et de l'activité globale	Implémentation des fonctionnalités pour suivre le nombre total de médecins, les patients et les aides dans le centre médical.
Fonctionnalité de modification de profil utilisateur	Conception de l'interface utilisateur permettant aux utilisateurs de modifier leur profil.
Application mobile.	Mise en place du projet de base
Rapport du stage	Rédaction du rapport.

Tableau 6 : Sprint 4

Conclusion :

En conclusion de ce chapitre d'analyse et de spécification des besoins pour notre projet de gestion des rendez-vous du cabinet médical, nous avons identifié les acteurs clés ainsi que leurs rôles et responsabilités dans le système. Nous avons également défini les besoins fonctionnels et non fonctionnels de l'application, garantissant ainsi une compréhension claire des objectifs à atteindre. La planification des sprints a été établie pour une période de quatre semaines, avec des objectifs spécifiques pour chaque itération. Cette approche itérative nous permettra de développer et d'itérer progressivement notre solution tout en restant alignés sur les besoins des utilisateurs.

Chapitre 3 : Etude Conceptuel

Introduction

La phase de conception joue un rôle essentiel dans le cycle de vie d'un logiciel, servant à élaborer une description détaillée du système requis avant sa mise en œuvre. Les développeurs sont souvent confrontés aux risques d'erreurs et de pertes de temps, rendant indispensable l'adoption d'une méthodologie de développement structurée. Ce chapitre se concentrera sur l'étude conceptuelle, visant à poser les bases solides pour la réalisation de notre projet.

1 Diagrammes de cas d'utilisation

1.1 Description détaillée des cas d'utilisations

Dans cette section, nous présentons une description détaillée de principaux cas d'utilisations :

- **Authentification**

Nom du cas d'utilisation	Authentification
Acteur	Assistant, Médecin, Administrateur
Pré condition	L'utilisateur veut s'authentifier pour accéder à son propre espace.
Post condition	L'utilisateur trouve un formulaire d'authentification.
Scenario normal	<ul style="list-style-type: none">♣ Un formulaire s'affiche.♣ L'utilisateur remplit les champs du formulaire.♣ L'utilisateur appuie sur bouton Se Connecter.♣ Validation des champs♣ Les champs sont bien remplis.♣ Les informations entrées sont correctes.♣ Accéder au menu.
Scenario alternative	<ul style="list-style-type: none">♣ Un formulaire s'affiche.♣ L'utilisateur remplit les champs du formulaire.♣ L'utilisateur appuie sur le bouton Se Connecter.♣ Validation des champs♣ Les champs ne sont pas bien remplis.♣ Un message d'erreur s'affiche.

Tableau 7 : Descriptif de cas d'utilisation Authentifier

Diagramme de cas d'utilisations d'assistant et médecin

Le diagramme de cas d'utilisation pour l'assistant et le médecin illustre les interactions essentielles avec le système de gestion des patients et des rendez-vous. Il montre comment les acteurs peuvent ajouter de nouveaux patients et planifier des rendez-vous en assurant la vérification et la validation des informations. Ce système permet d'améliorer l'efficacité et l'organisation des tâches quotidiennes dans un environnement médical.

Le diagramme de cas d'utilisation suivant illustre en détails les rôles assurés par ces acteurs:

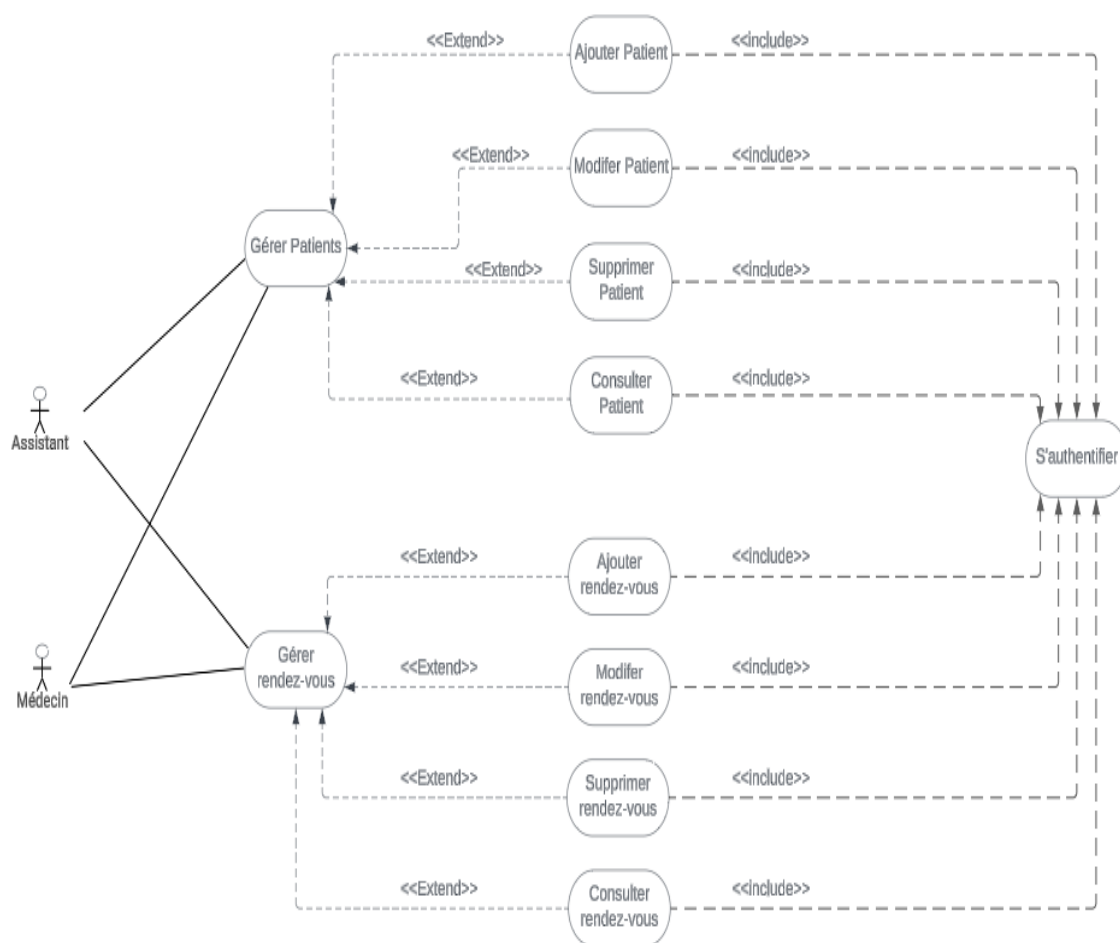


Figure 8 : Diagramme Use Case Assistant & Médecin

- **Ajouter Patient**

Nom du cas d'utilisation	Ajouter patient
Acteur	Assistant, Médecin
Pré condition	L'acteur veut ajouter un patient
Post condition	Un formulaire d'informations du patient à remplir s'affiche.
Scenario normal	<ul style="list-style-type: none">♣ Le formulaire s'affiche.♣ L'acteur remplit les champs du formulaire.♣ L'acteur appuie sur le bouton ajouter.♣ Validation des champs.♣ Les champs sont bien remplis.♣ Le patient n'existe pas déjà.♣ Enregistre le patient dans la base des données.
Scenario alternative	<p>Scenario 1</p> <ul style="list-style-type: none">♣ Le formulaire s'affiche.♣ L'acteur remplit les champs du formulaire.♣ L'acteur appuie sur le bouton ajouter.♣ Validation des champs.♣ Les champs ne sont bien remplis.♣ Un message d'erreur s'affiche pour informer la secrétaire. <p>Scenario 2</p> <ul style="list-style-type: none">♣ Le formulaire s'affiche.♣ L'acteur remplit les champs du formulaire.♣ L'acteur appuie sur le bouton ajouter.♣ Validation des champs♣ Les champs sont bien remplis.♣ Le patient existe déjà.♣ Un message d'erreur s'affiche pour informer la secrétaire.

Tableau 8 : Descriptif de cas d'utilisation Ajouter patient

- **Ajouter Rendez-vous**

Nom du cas d'utilisation	Ajouter rendez-vous
Acteur	Assistant, Médecin
Pré condition	Un patient veut prendre un rendez-vous
Post condition	L'acteur veut ajouter un rendez-vous.
Scenario normal	<p>L'acteur entre le nom de patient.</p> <ul style="list-style-type: none">♣ Le patient existe♣ La secrétaire choisit la date du rendez-vous et le médecin.♣ La disponibilité de date et médecin est vérifiée.♣ Tous les deux sont disponibles.♣ Le rendez-vous est ajouté.
Scenario alternative	<p>Scenario 1</p> <ul style="list-style-type: none">♣ L'acteur entre le nom de patient.♣ Le patient n'existe pas.♣ Le système affiche un message d'erreur pour l'informer. <p>Scenario 2</p> <ul style="list-style-type: none">♣ L'acteur entre le nom de patient.♣ Le patient existe.♣ L'acteur choisit la date du rendez-vous et le médecin.♣ La disponibilité de date et de médecin est vérifiée.♣ Soit la date du le médecin n'est pas disponible.♣ Le système affiche un message d'erreur pour l'informer.

Tableau 9 : Descriptif de cas d'utilisation Ajouter rendez-vous

Diagramme de cas d'utilisations d'administrateur

En plus des rôles de médecin, l'administrateur peut gérer les spécialités les médecins et les secrétaires.

Le diagramme de cas d'utilisation suivant illustre en détails les rôles assurés par l'admin:

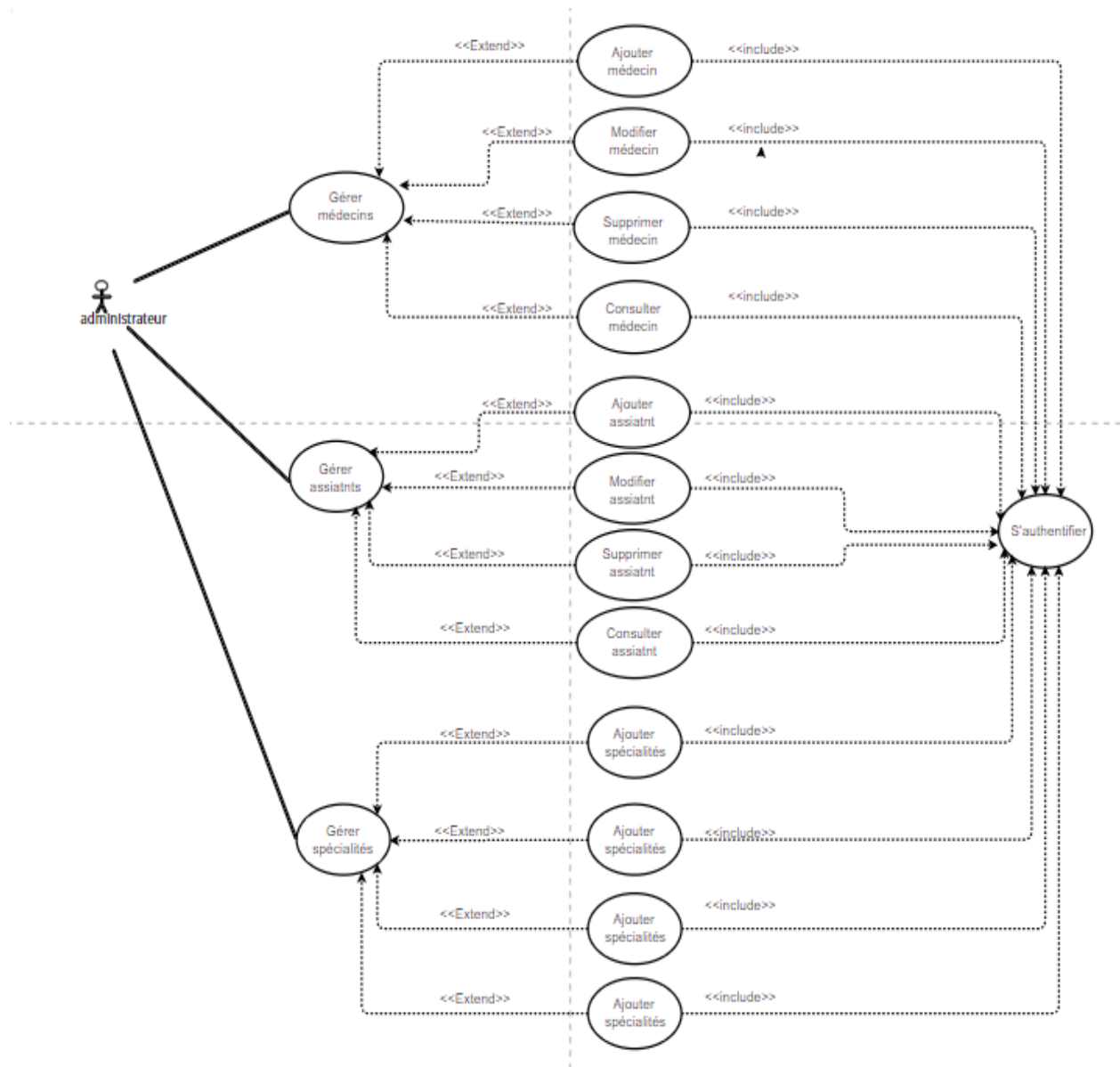


Figure 9 : Diagramme de cas d'utilisation d'Administrateur

- **Ajouter médecin.**

Nom du cas d'utilisation	Ajouter médecin
Acteur	Administrateur
Pré condition	L'administrateur veut ajouter un nouveau médecin
Post condition	Un formulaire s'affiche à l'administrateur à remplir
Scenario normal	<p>Le formulaire s'affiche.</p> <ul style="list-style-type: none">♣ L'administrateur remplit les champs du formulaire.♣ L'administrateur appuie sur le bouton ajouter.♣ Validation des champs.♣ Les champs sont bien remplis.♣ Le médecin n'existe pas déjà.♣ Enregistrer le médecin dans la base des données.
Scenario alternative	<p>Scenario 1</p> <ul style="list-style-type: none">♣ Le formulaire s'affiche.♣ L'administrateur remplit les champs du formulaire.♣ L'administrateur appuie sur le bouton ajouter.♣ Validation des champs.♣ Les champs ne sont bien remplis.♣ Le système affiche un message d'erreur pour L'administrateur. <p>Scenario 2</p> <ul style="list-style-type: none">♣ Le formulaire s'affiche.♣ L'administrateur remplit les champs du formulaire.♣ L'administrateur appuie sur le bouton ajouter.♣ Validation des champs.♣ Les champs sont bien remplis.♣ Le médecin existe déjà.♣ Le système affiche un message d'erreur pour informer L'administrateur.

Tableau 10 : Descriptif de cas d'utilisation Ajouter Médecin

- **Ajouter Assistant**

Nom du cas d'utilisation	Ajouter Assistant
Acteur	Administrateur
Pré condition	L'administrateur veut ajouter un nouvel assistant
Post condition	Un formulaire s'affiche à l'administrateur à remplir
Scenario normal	<p>Le formulaire s'affiche.</p> <p>L'administrateur remplit les champs du formulaire.</p> <p>L'administrateur appuie sur le bouton ajouter.</p> <ul style="list-style-type: none">♣ Validation des champs.♣ Les champs sont bien remplis.♣ L'assistant n'existe pas déjà.♣ Enregistrer l'assistant dans la base des données.
Scenario alternative	<p>Scenario 1</p> <ul style="list-style-type: none">♣ Le formulaire s'affiche.♣ L'administrateur remplit les champs du formulaire.♣ L'administrateur appuie sur le bouton ajouter.♣ Validation des champs.♣ Les champs ne sont bien remplis.♣ Le système affiche un message d'erreur pour L'administrateur. <p>Scenario 2</p> <ul style="list-style-type: none">♣ Le formulaire s'affiche.♣ L'administrateur remplit les champs du formulaire.♣ L'administrateur appuie sur le bouton ajouter.♣ Validation des champs.♣ Les champs sont bien remplis.♣ L'assistant existe déjà.♣ Le système affiche un message d'erreur pour informer L'administrateur.

Tableau 11 : Descriptif de cas d'utilisation Ajouter Assistant

- Ajouter Spécialité

Nom du cas d'utilisation	Ajouter Spécialité
Acteur	Administrateur
Pré condition	L'administrateur veut ajouter une nouvelle spécialité
Post condition	Un formulaire s'affiche à l'administrateur à remplir
Scenario normal	<p>Le formulaire s'affiche.</p> <p>L'administrateur remplit les champs du formulaire.</p> <p>L'administrateur chef appuie sur le bouton ajouter.</p> <ul style="list-style-type: none">♣ Validation des champs.♣ Les champs sont bien remplis.♣ La spécialité n'existe pas déjà.♣ Enregistrer la spécialité dans la base des données.
Scenario alternative	<p>Scenario 1</p> <ul style="list-style-type: none">♣ Le formulaire s'affiche.♣ L'administrateur remplit les champs du formulaire.♣ L'administrateur appuie sur le bouton ajouter.♣ Validation des champs.♣ Les champs ne sont pas bien remplis.♣ Le système affiche un message d'erreur pour L'administrateur. <p>Scenario 2</p> <ul style="list-style-type: none">♣ Le formulaire s'affiche.♣ L'administrateur remplit les champs du formulaire.♣ L'administrateur appuie sur le bouton ajouter.♣ Validation des champs.♣ Les champs sont bien remplis.♣ La spécialité existe déjà.♣ Le système affiche un message d'erreur pour informer L'administrateur.

Tableau 12 : Descriptif de cas Ajouter Spécialité

1.2. Diagramme de cas d'utilisation Global

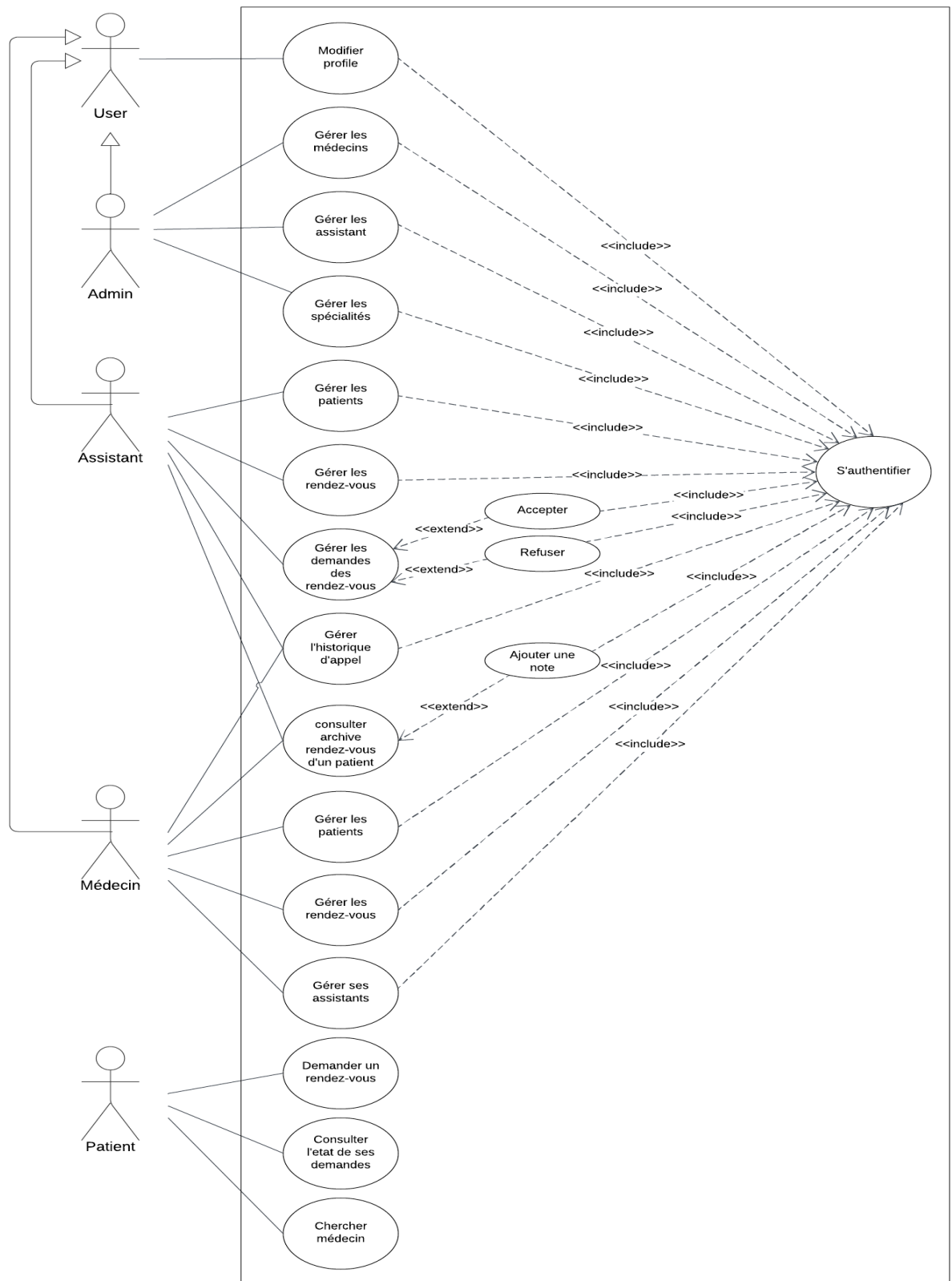


Figure 10: diagramme de use case

2. Diagramme des classes

Les diagrammes de classes sans doute sont les diagrammes les plus utilisés d'UML, ils décrivent les types des objets qui composent un système et les différents types de relation statique qui existent entre eux. Les classes sur lesquelles se porte notre application sont les suivantes :

- User
- Assistant Médicaux
- Médecin
- Admin
- Patient
- Spécialité
- Image
- Historique d'appel
- Rendez-vous
- Demande Rendez-vous

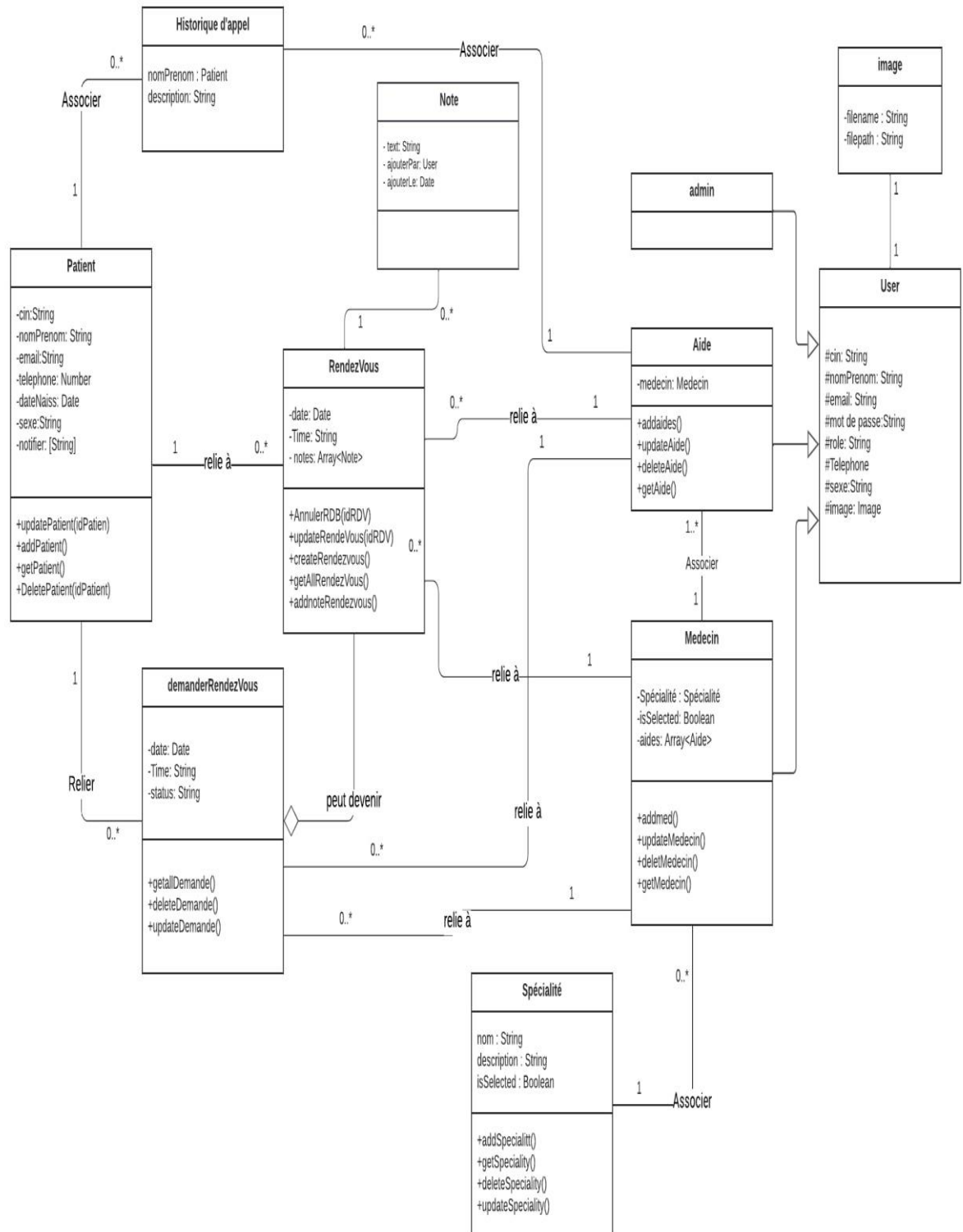


Figure 11: Diagramme des classes

3. Diagramme des séquences

Le diagramme de séquence permet de montrer les interactions d'objets dans le cadre d'un scénario d'un diagramme des cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite du système. Le but est de décrire comment se déroulent les interactions entre les acteurs ou objets.

3.1. Diagramme de séquence « Authentification »

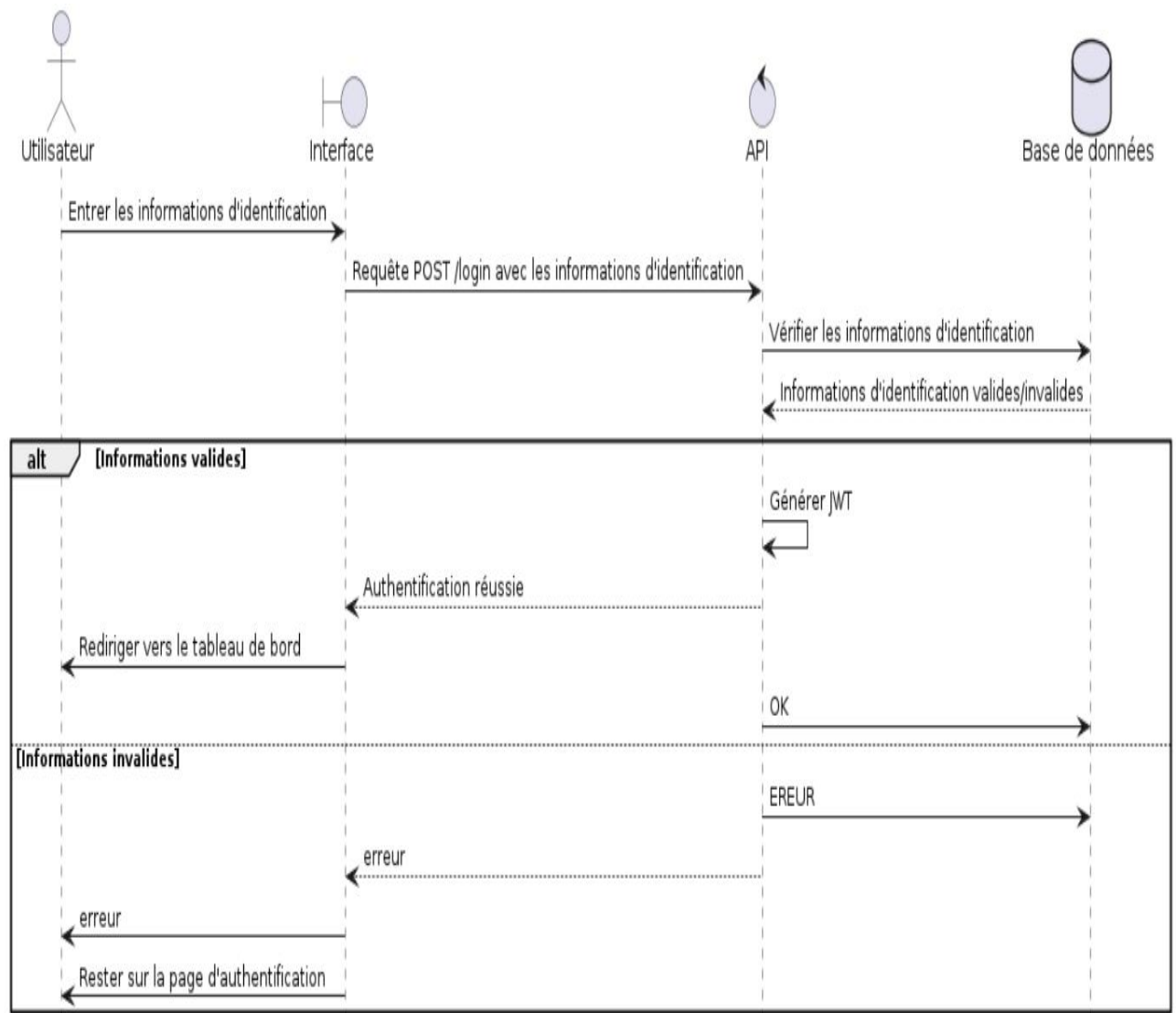


Figure 12 : Diagramme de séquence "Authentification"

3.2. Diagramme de séquence Secrétaire et Médecin

Dans le système de gestion des rendez-vous médicaux, l'acteur 'Utilisateur', tel que représenté dans notre diagramme UML, peut être soit un secrétaire, soit un médecin. Cette représentation souligne la flexibilité du système, permettant de comprendre que des actions telles que la création, la modification ou la suppression de rendez-vous peuvent être effectuées par différents types d'utilisateurs.

A) Diagramme de séquence « Création d'un rendez-vous »

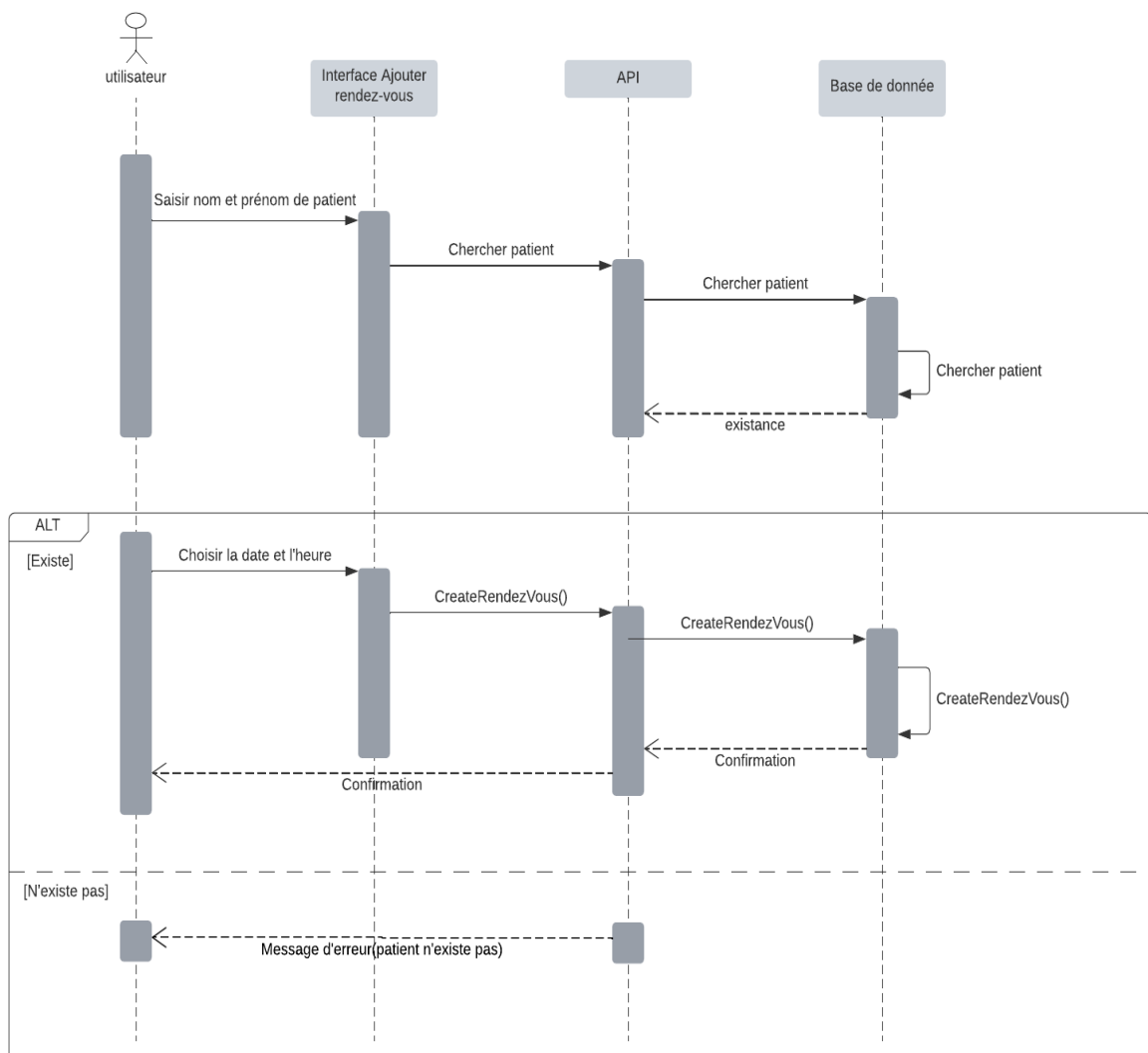


Figure 13 : Diagramme de séquence « Ajouter un rendez-vous »

B) Diagramme de séquence « Modification d'un rendez-vous »

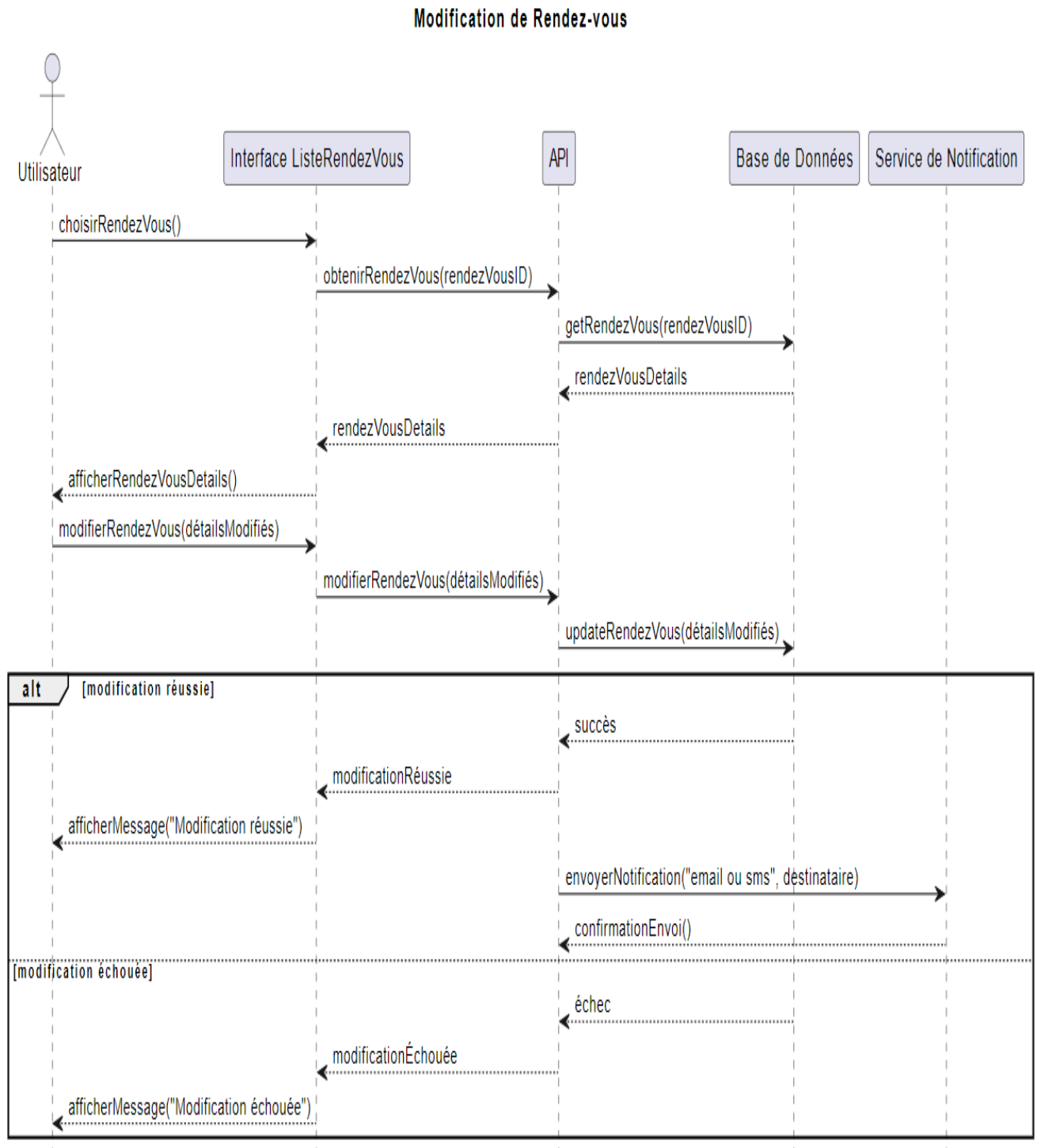


Figure 14: diagramme de séquence « Modifier rendez-vous »

C) Diagramme de séquence « Supprimer un rendez-vous »

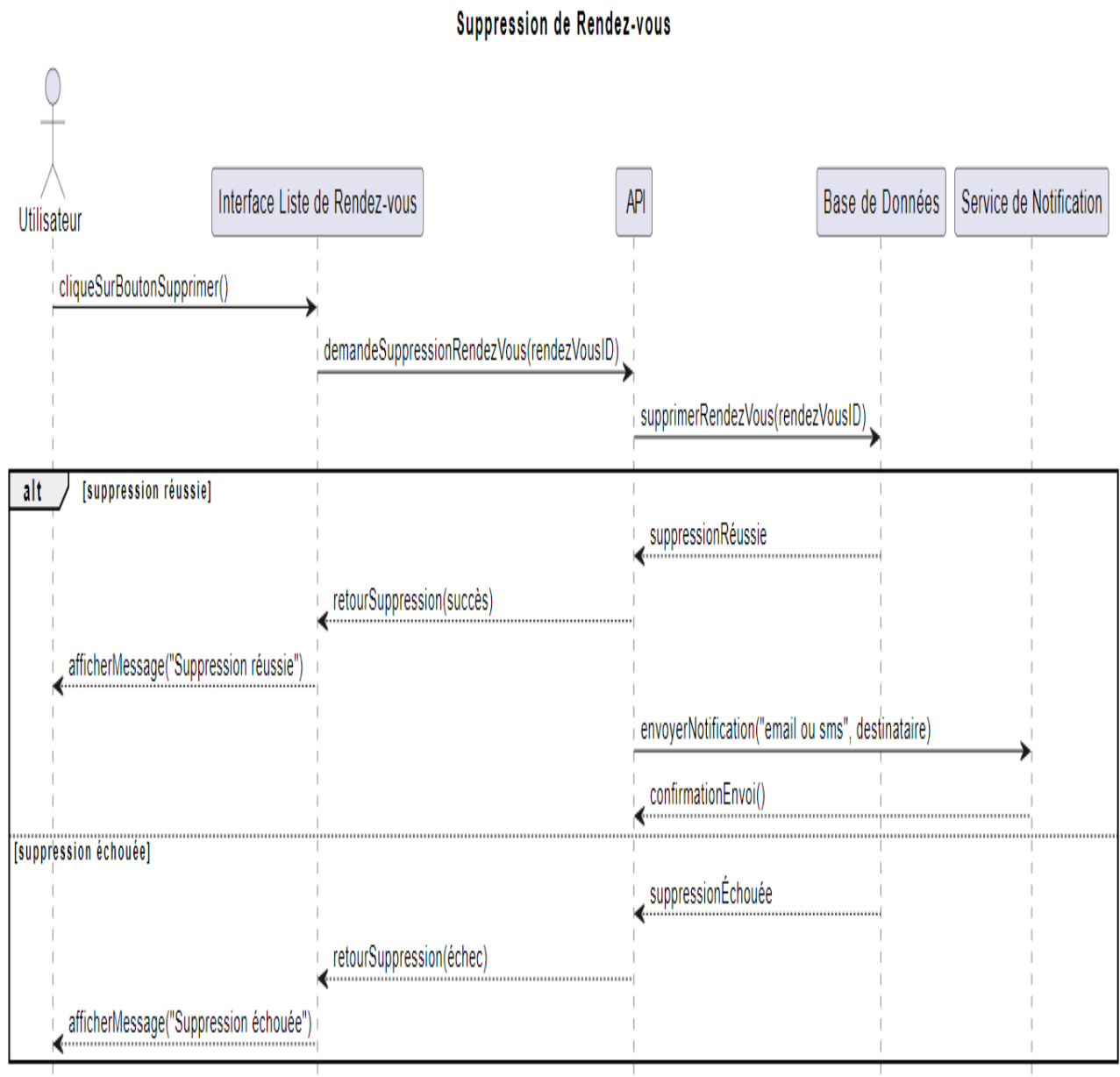


Figure 15 : diagramme de séquence « Supprimer rendez-vous »

Conclusion

Ce chapitre a établi les bases conceptuelles essentielles pour notre projet logiciel. Nous avons détaillé les principales fonctionnalités et interactions du système à travers le diagramme de cas d'utilisation, la structure statique via le diagramme des classes, et les interactions dynamiques avec le diagramme des séquences pour la création d'un rendez-vous. Cette étude conceptuelle vise à minimiser les risques d'erreurs et à optimiser le développement, préparant ainsi une transition fluide vers la phase de mise en œuvre.

Chapitre 4 : Réalisation et mise en œuvre

Introduction

Ce chapitre aborde la phase pratique de notre projet, où nous passons de la conception à la réalisation concrète. Nous commencerons par décrire l'environnement de développement, y compris les outils et les technologies utilisés pour construire notre application. Ensuite, nous explorerons l'architecture choisie pour structurer efficacement notre projet. Enfin, nous présenterons les différentes interfaces utilisateur de l'application, tant pour le web que pour le mobile.

1. Environnement de développement et outils de développement

1.1. Environnement matériels

Spécifications de l'appareil	
Nom de l'appareil	DESKTOP-JQHSQ79
Processeur	12th Gen Intel(R) Core(TM) i5-1235U 1.30 GHz
Mémoire RAM installée	16,0 Go (15,7 Go utilisable)
ID de périphérique	816B53B4-22AF-49DF-BEE1-3F8BD1724F0D
ID de produit	00330-80000-00000-AA951
Type du système	Système d'exploitation 64 bits, processeur x64
Stylet et fonction tactile	La fonctionnalité d'entrée tactile ou avec un stylet n'est pas disponible sur cet écran

Figure 16: Fiche technique du pc d'Alia

Spécifications de l'appareil

Nom de l'appareil	DESKTOP-R8BGLVG
Processeur	11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz
Mémoire RAM installée	12,0 Go (11,8 Go utilisable)
ID de périphérique	62B2AE50- B898-432D-8AEB-8727A703DECB
ID de produit	00326-30000-00001-AA381
Type du système	Système d'exploitation 64 bits, processeur x64
Styleret et fonction tactile	La fonctionnalité d'entrée tactile ou avec un styleret n'est pas disponible sur cet écran

Figure 17: fiche technique du pc de Nesrine

1.2. Environnement logiciels

- **Postman :**

Postman est un outil de développement d'API collaboratif qui simplifie la conception, les tests, la documentation et le partage des API. Avec une interface conviviale, il permet d'envoyer des requêtes HTTP, d'analyser les réponses, de gérer les environnements et les variables, et de créer des collections pour automatiser les tests. Postman favorise la communication et la collaboration au sein des équipes, offrant des fonctionnalités avancées telles que les tests automatisés et l'intégration CI/CD. Essentiel pour améliorer l'efficacité et la qualité du développement d'API.



Figure 18: logo POSTMAN

- **Visual Studio Code (VSCode) :**

Est un éditeur de code léger, puissant et extensible développé par Microsoft. Il prend en charge de nombreux langages de programmation et offre des fonctionnalités avancées telles que la coloration syntaxique, l'autocomplétion, le débogage intégré et la gestion de versions avec Git. Avec sa flexibilité et sa large gamme d'extensions, VSCode est devenu l'un des éditeurs de code les plus appréciés par les développeurs.

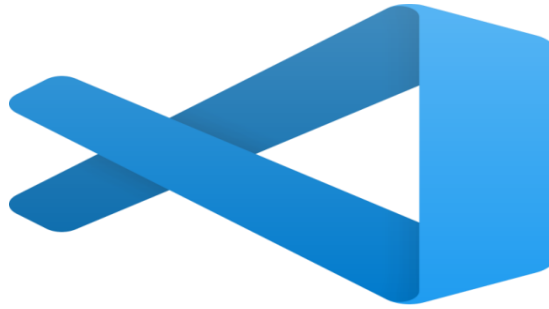


Figure 19: logo Visual Studio Code

- **Lucidchart : UML**

Est un outil en ligne de création de diagrammes et de visualisation de données. Il permet aux utilisateurs de concevoir facilement des diagrammes de flux, des organigrammes, des maquettes, des UML, des diagrammes de réseau et bien plus encore.

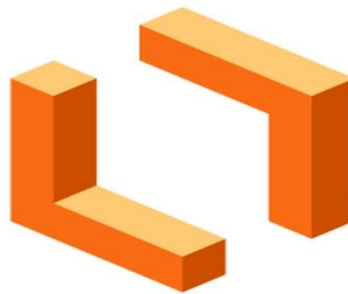


Figure 20: Logo Lucidchart

1.3. Technologie utilisé

1.3.1. Base de donnéé :

- **MongoDB**

MongoDB est une base de données NoSQL orientée documents. Contrairement aux bases de données relationnelles traditionnelles, elle stocke les données sous forme de documents JSON flexibles, ce qui permet une grande souplesse dans la gestion des données.



Figure 21: MongoDB

1.3.2. Framework et Langages

- **Express.js :**

Utilisé comme Framework web pour Node.js, il a simplifié le développement côté serveur en offrant des fonctionnalités robustes.

- **Node.js :**

Choisi comme environnement d'exécution côté serveur pour son exécution rapide et efficace du code JavaScript.



Figure 22: Express.js/ Node.js

- **React :**

React est une bibliothèque JavaScript utilisée pour construire des composants d'interface utilisateur réutilisables. Selon la documentation officielle de React, la définition est la suivante - React est une bibliothèque permettant de construire des interfaces utilisateur composables.



Figure 23: React.js

- **React Native :**

React Native est un Framework open-source développé par Facebook, permettant de créer des applications mobiles natives pour iOS et Android en utilisant JavaScript et React. Il permet aux développeurs de construire des interfaces utilisateur riches et performantes, avec une base de code unique, en combinant la facilité de développement web avec les performances des applications natives.

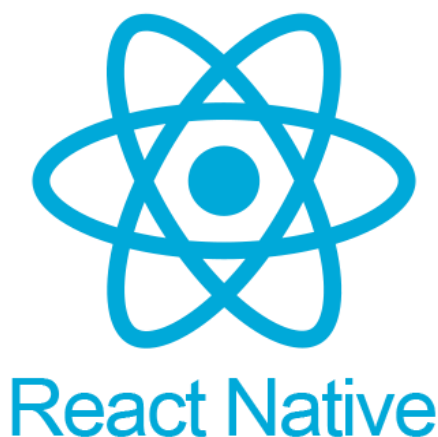


Figure 24: React Native

- **TailwindCSS :**

TailwindCSS CSS est un Framework CSS utilitaire. Au lieu de fournir des composants prêts à l'emploi, il offre une série de classes utilitaires qui permettent de construire des interfaces utilisateur directement dans le balisage HTML. Cette approche permet une grande flexibilité et une personnalisation facile.



Figure 25: TailwindCSS

- **CSS**

CSS, ou Cascading Style Sheets, est un langage de feuilles de style utilisé pour définir l'apparence et la mise en forme des documents HTML. Il permet de contrôler la présentation des éléments sur une page web, tels que les couleurs, les polices, les marges et les espacements. CSS facilite la séparation entre la structure (HTML) et le style (CSS) d'une page, offrant ainsi aux développeurs une grande flexibilité pour créer des designs attrayants et cohérents sur le web.



Figure 26: CSS

- **Nodemailer**

Nodemailer est un module Node.js pour envoyer des e-mails. Il offre une API simple pour envoyer des e-mails via SMTP. Nodemailer est souvent utilisé pour les applications web nécessitant des fonctionnalités d'e-mailing, telles que la validation d'utilisateur et les notifications.



Figure 27: Nodemailer

- **Infobip**

Infobip est une plateforme de communication en nuage permettant aux entreprises d'envoyer des messages sur divers canaux, tels que SMS, email, chat, et notifications push.



Figure 28: Infobip

- **Socket.IO**

Socket.IO est une bibliothèque JavaScript permettant une communication bidirectionnelle en temps réel entre clients et serveurs via WebSockets, facilitant la création d'applications interactives telles que les chats en ligne et les notifications instantanées.

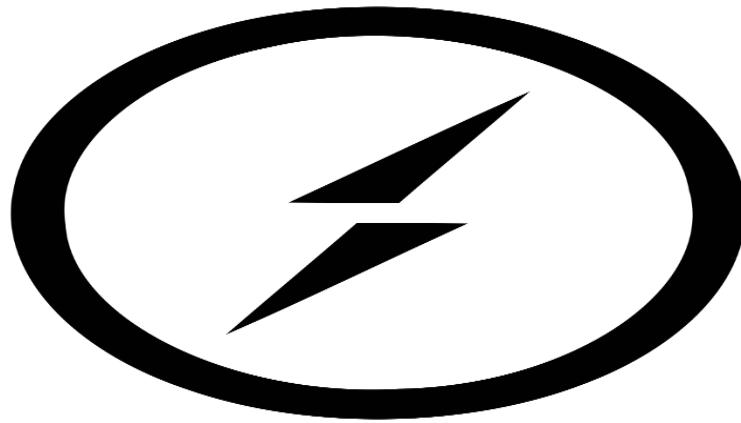


Figure 29: Socket.IO

- **Git Hub**

Git Hub est une plateforme de développement collaboratif où vous pouvez uploader, partager et gérer votre code, permettant de travailler en équipe sur des projets logiciels en utilisant le système de contrôle de version Git.



Figure 30 : Git Hub

2. Architecture

Dans notre projet, nous avons adopté l'architecture MVC, un concept puissant pour le développement d'applications. Cette architecture permet de séparer les données (modèle), l'affichage (vue) et les actions (contrôleur), assurant ainsi une clarté structurelle et facilitant la maintenance et l'amélioration du projet par les développeurs. Le schéma ci-dessous illustre les interactions entre le modèle, la vue et le contrôleur.

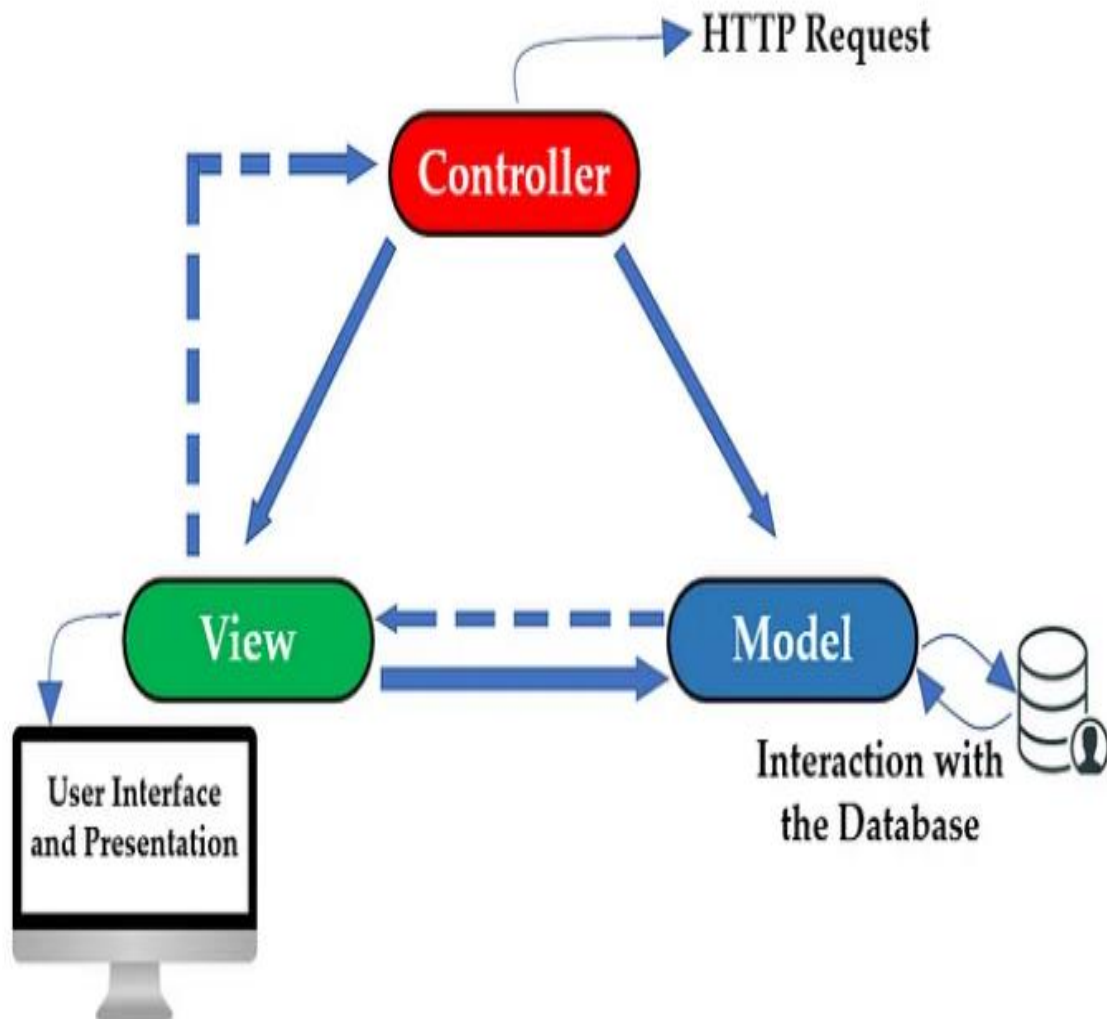


Figure 31 : Architecture MVC

3. Présentation des interfaces de l'application web

3.1. Interface d'authentification

Il est important de noter que chaque opération ne peut être effectuée qu'après authentification. Seul le l'admin est responsable de définir le login et le mot de passe de chaque employé. Cette page permet au admin, aux médecins et aux secrétaires de se connecter si les informations saisies sont valides.

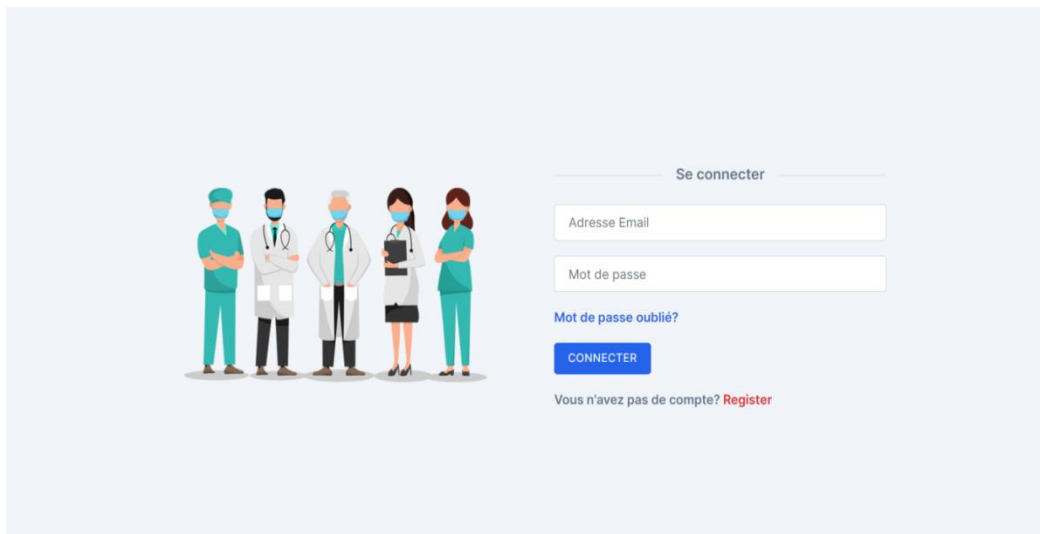


Figure 32: interface de login

Dashboard :

Les utilisateurs peuvent consulter les différentes statistiques du cabinet.

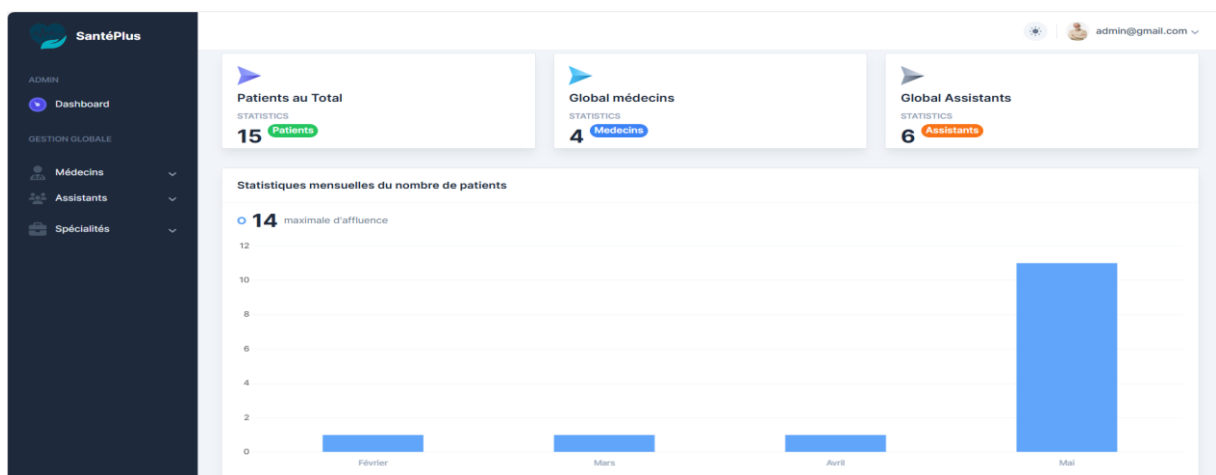


Figure 33: Dashboard

L'interface de profil : permet aux utilisateurs de modifier leurs informations personnelles. Grâce à cette interface, ils peuvent mettre à jour leurs coordonnées, changer leur mot de passe et ajuster d'autres détails pertinents de leur profil.

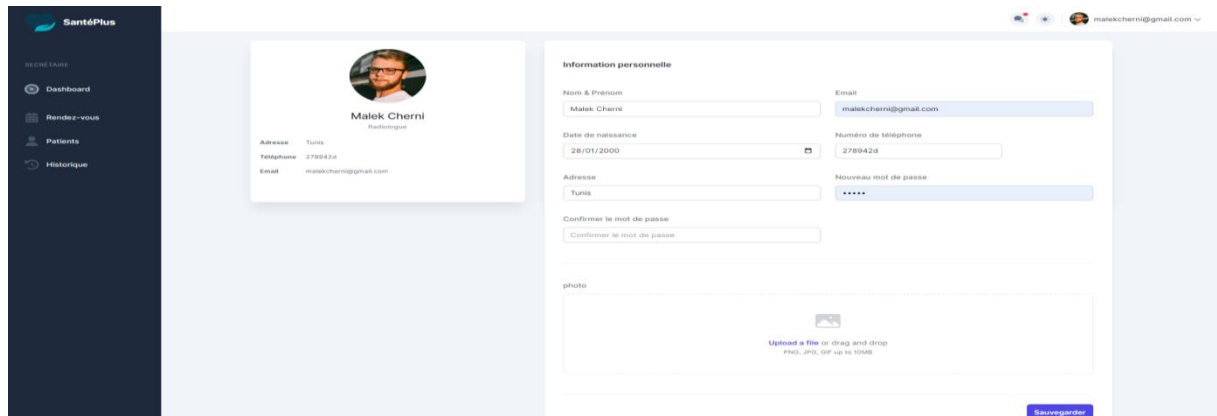


Figure 34 : Interface user Profile

3.2. Espace de secrétaire

3.2.1. Interface pour gérer les rendez-vous

L'interface de calendrier permet aux utilisateurs de visualiser les rendez-vous médicaux de manière claire et organisée. Depuis cette interface, ils peuvent consulter les rendez-vous programmés par mois, par semaine et par jour. Cette fonctionnalité facilite la gestion des emplois du temps, permettant de vérifier rapidement les disponibilités et de planifier efficacement les consultations médicales.

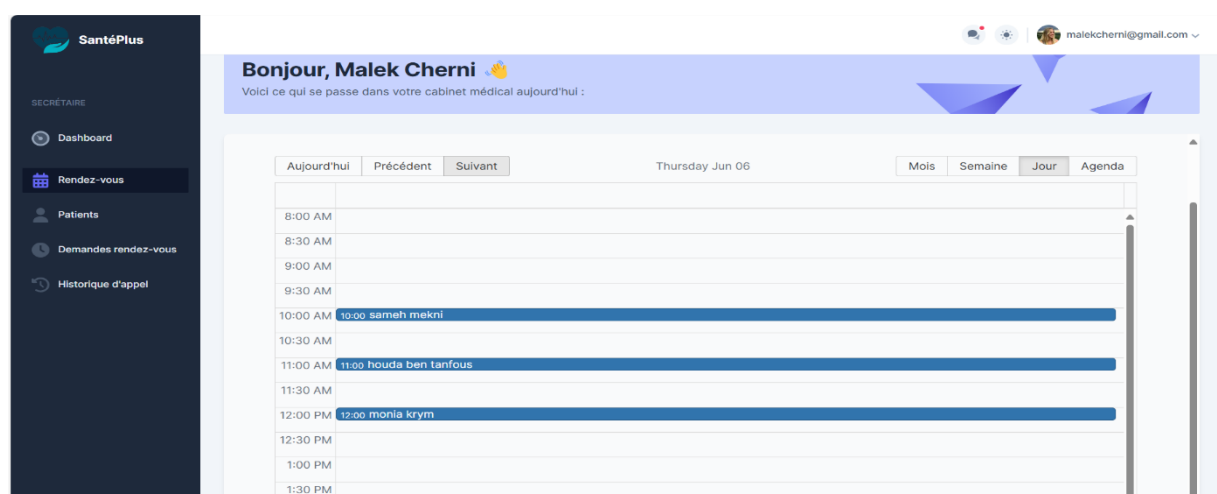


Figure 35: Visualisation des rendez-vous par jour

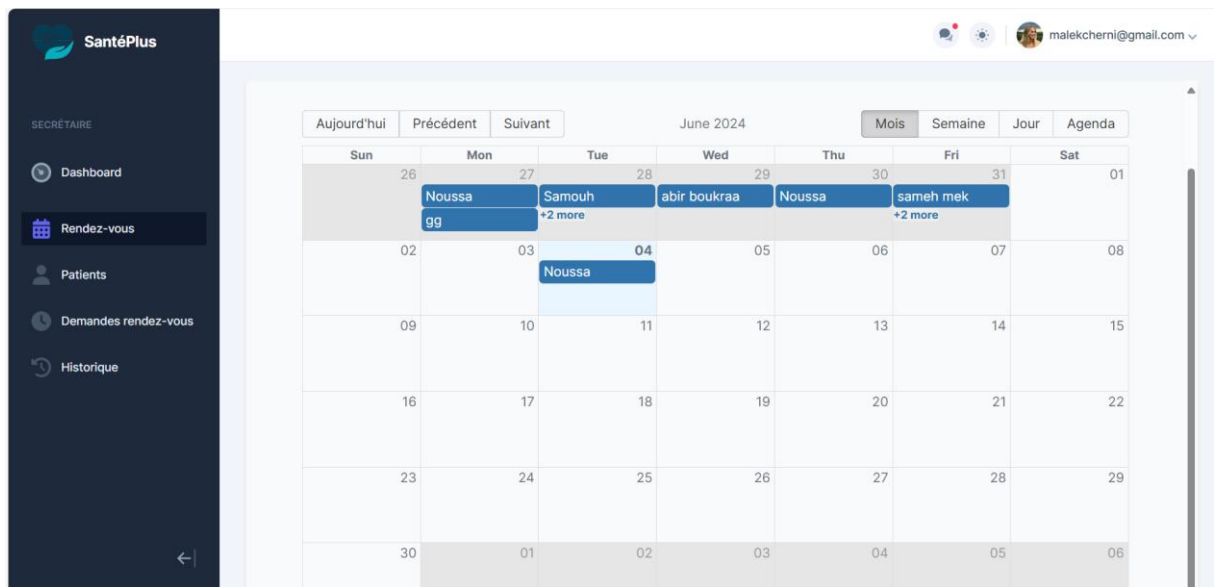


Figure 36: Visualisation des rendez-vous par mois

Ajouter un rendez-vous :

Pour ajouter un rendez-vous, il est d'abord nécessaire de rechercher le nom du patient dans la base de données. Si le patient est déjà enregistré, il suffit de sélectionner son nom, puis de choisir la date et l'heure du rendez-vous souhaité.

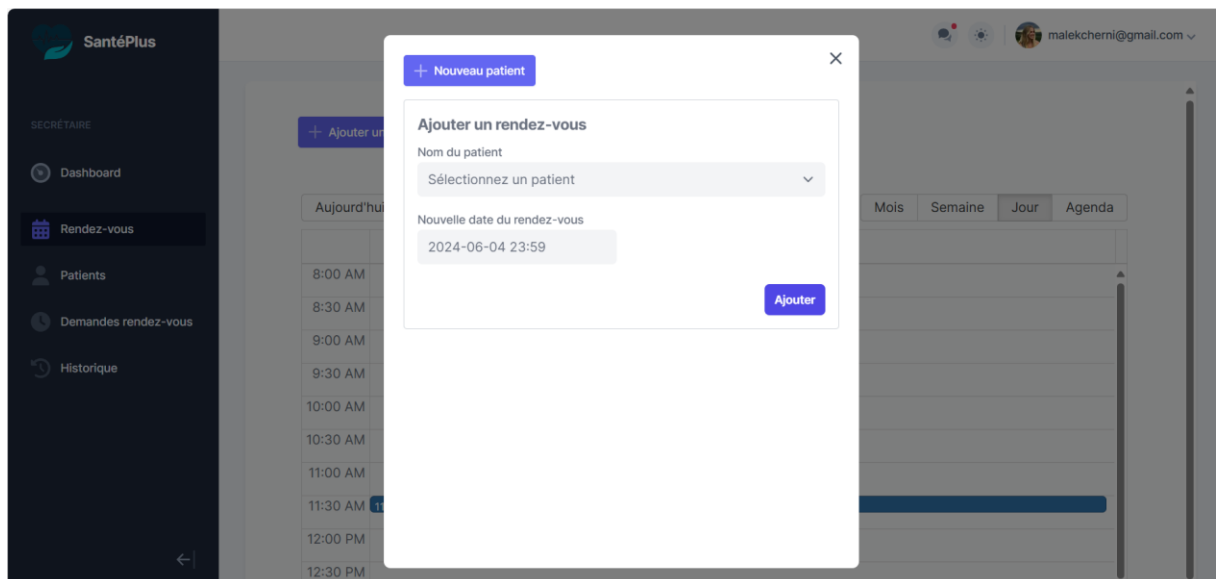


Figure 37: Ajouter un rendez-vous

Si le nom du patient n'existe pas dans la liste, l'utilisateur doit ajouter les informations du nouveau patient avant de procéder à la planification du rendez-vous.

The screenshot shows the 'Ajouter un patient' modal form. The form contains the following fields and options:

- Nom & Prénom ***: Text input field.
- CIN ***: Text input field.
- Email**: Text input field.
- Numéro de téléphone ***: Text input field with a pre-filled value of '+216 25 222 555'.
- Date de naissance ***: Date picker showing '05/06/2024'.
- Sexe ***: Radio buttons for 'Homme' and 'Femme'.
- Notifications**: Checkboxes for 'Par Email', 'Par SMS', and 'Appel téléphonique'.
- Ajouter**: Blue button at the bottom right.

The background shows the main interface with a sidebar menu (Dashboard, Rendez-vous, Patients, Demandes rendez-vous, Historique) and a calendar view for 'Aujourd'hui' and 'Précédent'.

Figure 38: Ajouter un patient

Modifier un rendez-vous :

En cliquant sur un rendez-vous spécifique, une fenêtre modale s'affiche, offrant des options pour modifier la date et l'heure du rendez-vous ou pour l'annuler. Lorsqu'un rendez-vous est modifié ou annulé, le patient reçoit automatiquement une notification par SMS ou par email.

The screenshot shows the 'Modifier le Rendez-vous' modal form. The form contains the following fields and options:

- Nom du patient**: Text input field with the value 'houda ben tanfous'.
- Nouvelle date du rendez-vous**: Date and time picker showing '2024-06-06 11:00'.
- Modifier**: Blue button at the bottom left.
- Annuler le rendez-vous**: Red button at the bottom right.

The background shows the main interface with a sidebar menu and a calendar view for 'Aujourd'hui' and 'Précédent'. The calendar shows a list of appointments with names like 'sameh mekni', 'houda ben tanfous', and 'monia krym'.

Figure 39: Modifier un rendez-vous

3.2.2. Interface pour gérer les patients

L'interface "Gérer patient" offre une gestion complète des dossiers des patients, modifier les informations personnelles des patients, supprimer un patient de la base de données, ou ajouter de nouveaux patients. De plus, cette interface permet de consulter les archives des rendez-vous de chaque patient.

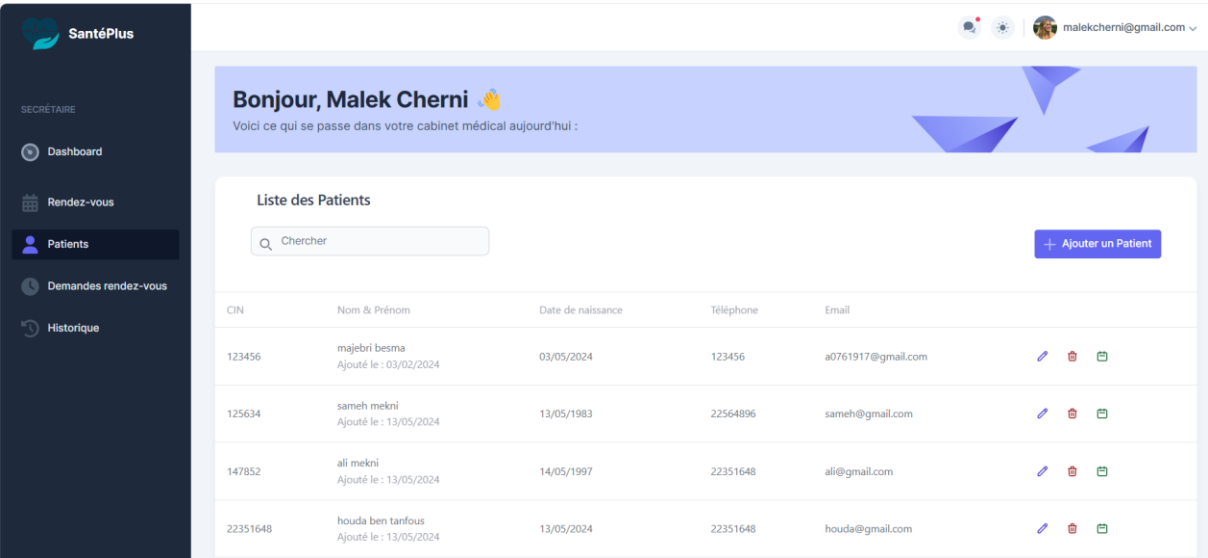


Figure 40: Interface liste des patients

L'interface "Gérer les demandes de rendez-vous" permet à l'assistant d'accepter les rendez-vous des patients fournis par l'application mobile où les patients passent leur réservation.

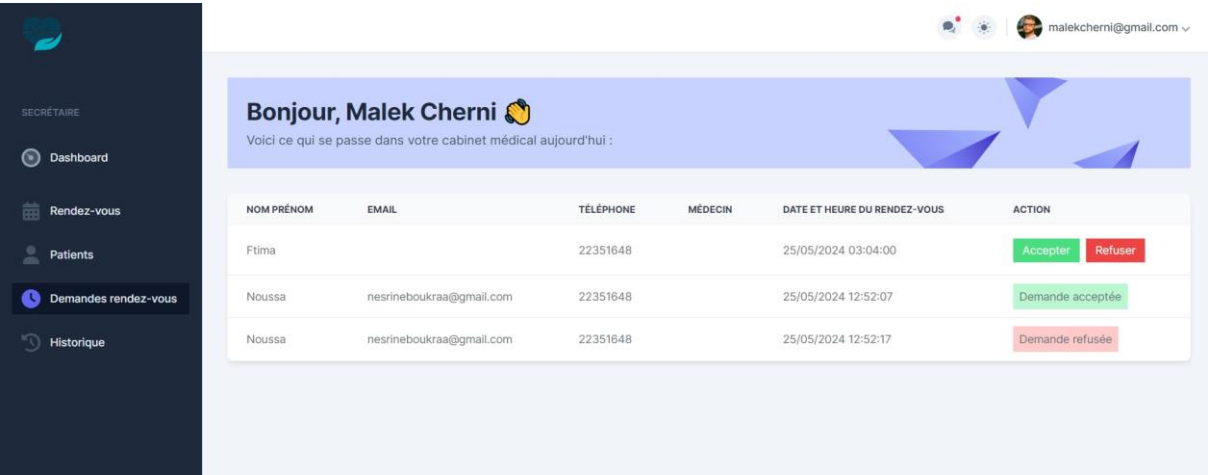


Figure 41 : Gérer les demande Rendez-vous

3.2.3 Historique d'appel

Historique d'appel est un écran qui affiche la liste des patients ayant effectué un appel pour modifier certaines informations.

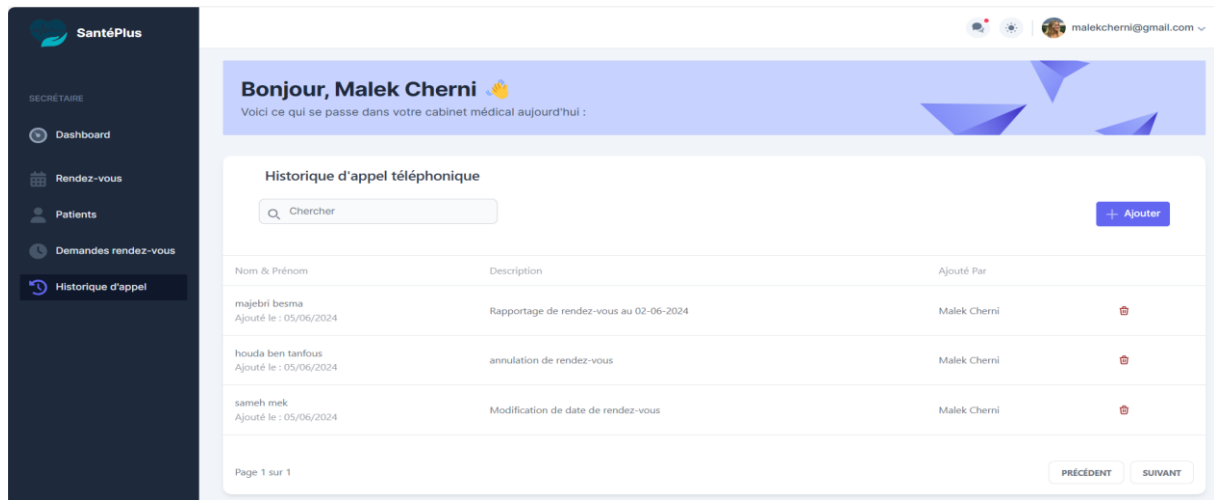


Figure 42 : Historique d'appel

3.3. Espace Admin

3.3.1. Interface pour gérer les médecins

L'admin peut ajouter de nouveaux médecins, modifier leurs informations personnelles ainsi que leurs spécialités. De plus, cette interface permet également de désactiver un médecin si nécessaire.

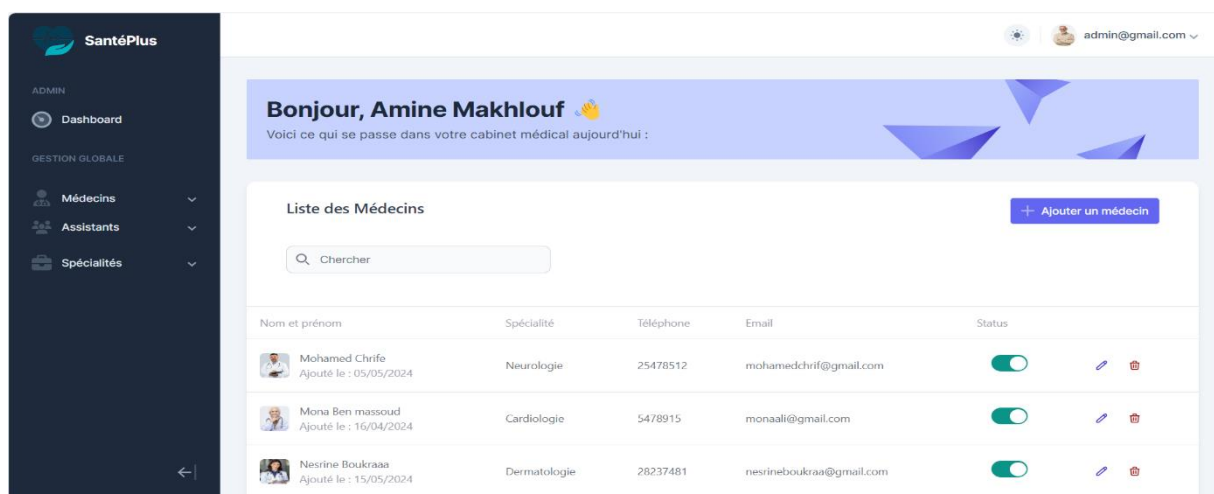


Figure 43: liste des médecins

Le formulaire 'Ajouter un médecin' est divisé en deux colonnes. La colonne de gauche contient les champs : CIN *, Numéro de téléphone *, Mot de passe *, Date d'adhésion *, Adresse *, et une section pour choisir une photo de profil. La colonne de droite contient : Nom & Prénom *, Email *, Sexe *, Date de naissance *, et Spécialité *. Des boutons 'Annuler' et 'Ajouter' sont situés en bas à droite.

Figure 44: formulaire ajouter un médecin

3.3.2. Interface pour gérer les assistants

L'administrateur peut ajouter de nouveaux assistants, modifier leurs informations et supprimer des profils d'assistants existants selon les besoins de l'établissement médical. De plus, cette interface permet à l'administrateur de lier chaque assistant à un médecin spécifique, facilitant ainsi une répartition efficace des tâches et une coordination optimale au sein de l'équipe médicale.

L'interface 'Liste des Assistants' présente un tableau des assistants avec des boutons d'action. Une barre de recherche est située au-dessus du tableau.










Nom et prénom	Poste	Téléphone	Email	Médecin lié	
 Malek Cherni Ajouté le : 30/04/2024	aide	2789420	malekcherni@gmail.com	Mona Ben massoud	 
 Aymen Moumni Ajouté le : 29/04/2024	aide	54784894	Aymen@gmail.com	Mohamed Chrife	 
 Fatma Akrouf Ajouté le : 08/05/2024	aide	57487125	alyaazzabi@gmail.com	Mona Ben massoud	 

Figure 45: interface liste des assistants

3.3.3. Interface pour gérer les spécialités

L'administrateur peut ajouter de nouvelles spécialités, modifier leurs informations et supprimer des spécialités non associées à des médecins. Cependant, si un médecin est associé à une spécialité, l'administrateur ne peut pas la supprimer. À la place, il peut modifier le statut de la spécialité en "désactivé" si nécessaire.



Figure 46: Interface pour gérer les spécialités

3.4. Espace médecin

Dans l'interface du médecin, il y a quatre pages principales. Trois d'entre elles sont identiques à celles de l'espace de l'assistant, tandis que la quatrième, spécifique au médecin, concerne la gestion de ses assistants.

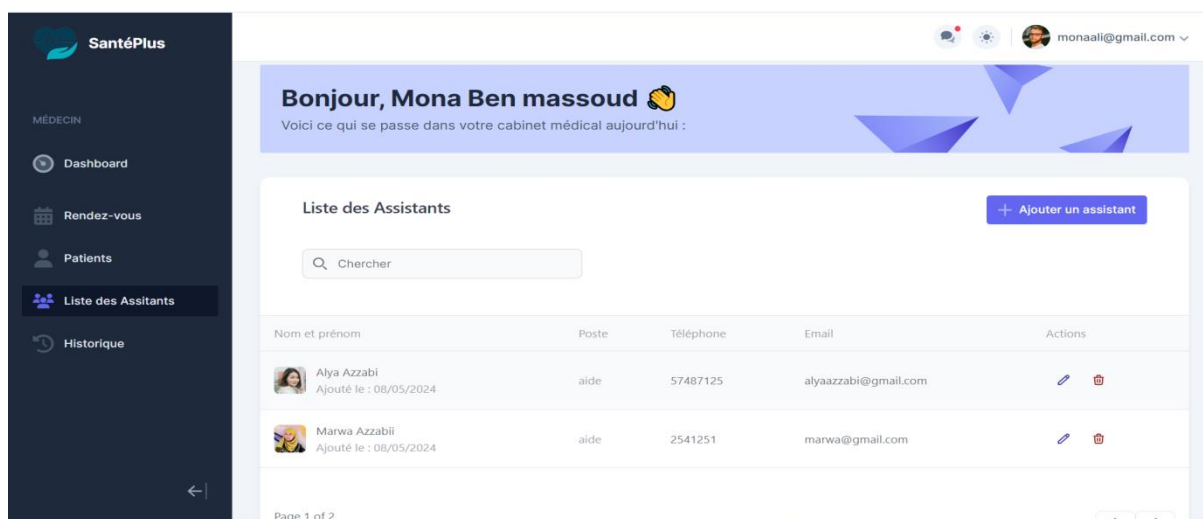


Figure 47: Interface pour gérer les assistants d'un médecin

4. Présentation des interfaces de l'application mobile

- Accueil

L'application "Center santéPlus" vous permet de trouver facilement un médecin selon sa spécialité. La page d'accueil propose une interface conviviale avec des illustrations accueillantes et des boutons clairs pour sélectionner la spécialité médicale souhaitée ou afficher toutes les spécialités. Vous pouvez également prendre rendez-vous directement via l'application. Un message de bienvenue renforce l'accueil chaleureux et l'accès facile aux services médicaux.

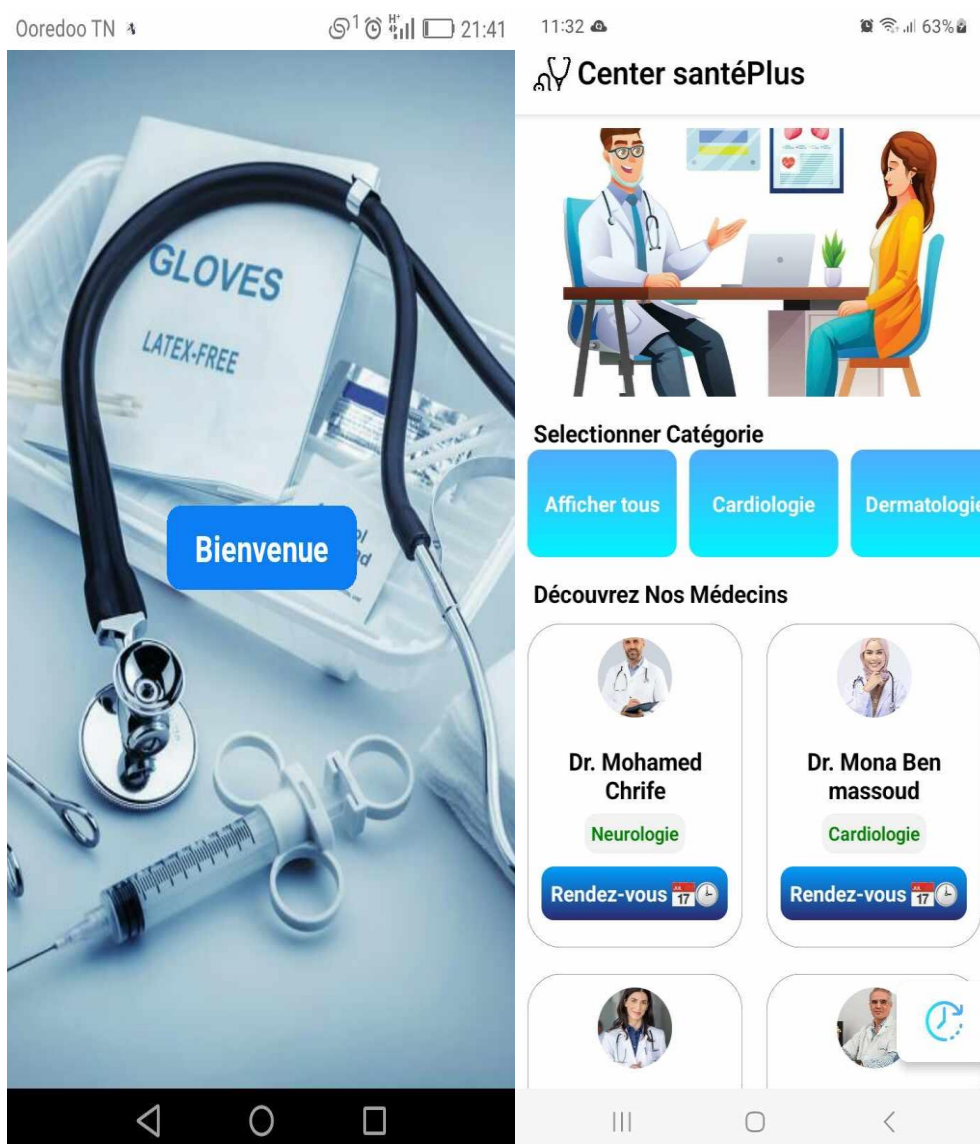


Figure 48: page d'accueil Figure 49: home page

- **Demande de rendez-vous**

Ces deux interfaces sont conçues pour faciliter la prise de rendez-vous. La première interface est destinée aux nouveaux patients. Elle leur permet de choisir la date et l'heure de leur rendez-vous et de remplir un formulaire complet avec leurs informations personnelles.

La deuxième interface est dédiée aux patients existants. Pour ceux-ci, il suffit de saisir leur CIN pour vérifier leur identité et soumettre une demande de rendez-vous. Cela simplifie le processus en évitant de remplir de nouveau toutes les informations personnelles déjà enregistrées.

Les deux interfaces sont intuitives et visent à rendre le processus de prise de rendez-vous aussi fluide et rapide que possible pour tous les patients.

The image displays two side-by-side screenshots of a mobile application interface for booking an appointment with a neurologist.

Left Screenshot (New Patient):

- Header:** Status bar at 11:33, signal strength, and 63% battery. A green pill-shaped button labeled "Docteur de Neurologie".
- Title:** "Planifiez Votre Rendez-vous".
- Selection Buttons:** Two light blue buttons: "Sélectionner la date: 22/05/2024" and "Sélectionner l'heure: 11:32:45".
- Section:** "Coordonné Patient" with a toggle switch set to "On" (green).
- Form:** A series of input fields for "CIN du Patient", "Nom & Prénom", "Téléphone", "Email", "Date de naissance (jj/mm/aaaa)", and "Adresse".
- Gender:** "Sexe:" with "Homme" and "Femme" buttons.
- Notifications:** "SMS", "Email", and "Appel" with toggle switches.
- Action:** A large blue "Réserver" button at the bottom.

Right Screenshot (Existing Patient):

- Header:** Status bar at 11:32, signal strength, and 63% battery. A back arrow and "Rendez-Vous" with a calendar icon showing the 17th.
- Doctor Profile:** A circular profile picture of Dr. Mohamed Chrife, a "Docteur de Neurologie".
- Title:** "Planifiez Votre Rendez-vous".
- Selection Buttons:** Two light blue buttons: "Sélectionner la date: 22/05/2024" and "Sélectionner l'heure: 11:32:45".
- Section:** "Coordonné Patient" with a toggle switch set to "Off" (grey).
- Form:** A single input field labeled "Entrez votre CIN".
- Action:** A large blue "Réserver" button at the bottom.

Figure 50: interface ajouter rendez-vous

- Statuts des demandes des rendez-vous

Cette interface permet aux utilisateurs de vérifier le statut de leurs rendez-vous médicaux. En entrant leur numéro d'identification (CIN) et en cliquant sur "Rechercher", ils peuvent voir une liste de leurs rendez-vous avec les dates, heures, noms des médecins, et le statut de chaque rendez-vous (en attente, accepté, ou refusé). Cette page offre une vue d'ensemble claire et rapide sur les rendez-vous, facilitant ainsi la gestion et le suivi des consultations médicales pour les patients.

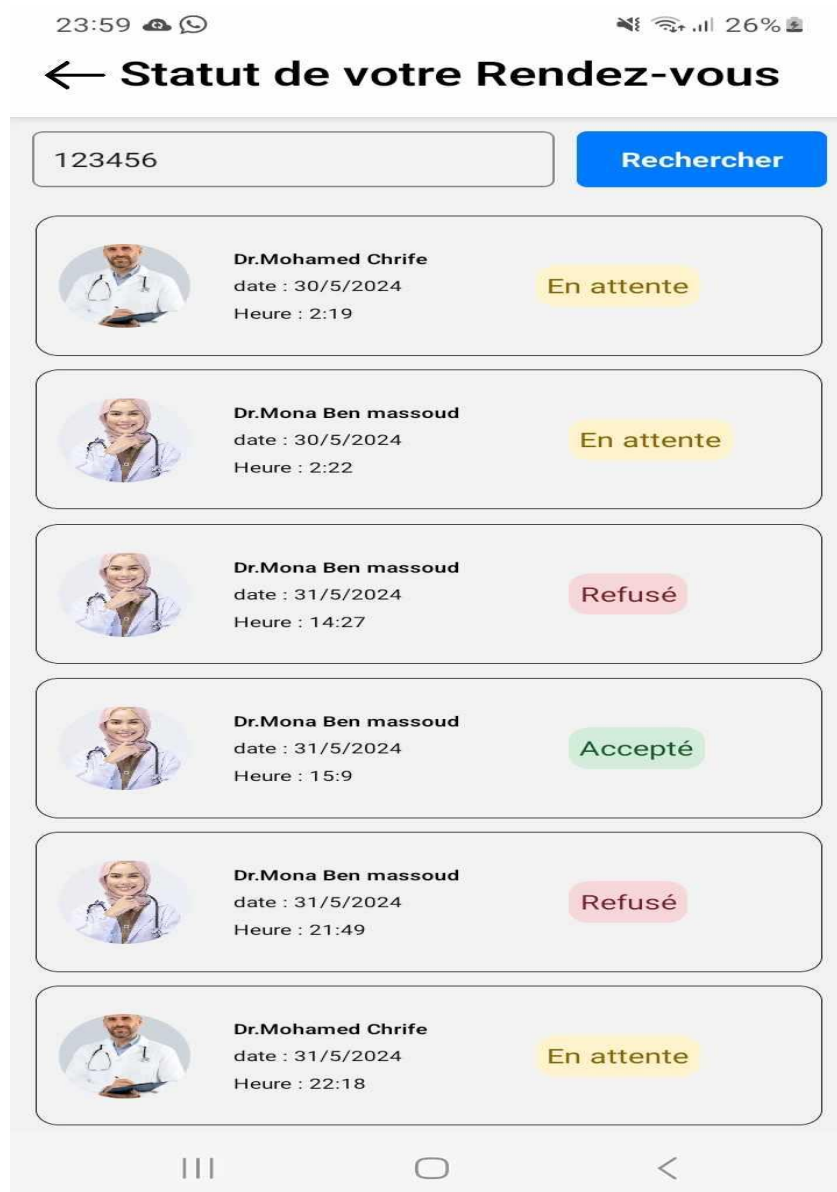


Figure 51: Consulter l'état des rendez-vous

Conclusion

En conclusion, la phase de réalisation et de mise en œuvre de notre projet a permis de concrétiser les concepts et les plans élaborés lors des étapes de conception. Grâce à un environnement de développement bien choisi et à l'utilisation de technologies modernes et robustes, nous avons pu construire une application fonctionnelle et performante. L'architecture MVC adoptée a facilité la gestion structurée du projet, permettant une maintenance et une évolutivité efficaces. Les interfaces utilisateur, tant pour l'application web que mobile, ont été conçues pour offrir une expérience utilisateur intuitive et fluide, répondant aux besoins spécifiques de chaque type d'utilisateur.

Conclusion générale

Ce stage a représenté une opportunité significative pour approfondir nos connaissances théoriques, acquises au cours des six semestres de licence, à travers la pratique des nouvelles technologies. Initialement, nous avons rencontré des difficultés pour appréhender pleinement le cahier des charges. Cependant, grâce à des recherches sur Internet, nous avons pu assimiler ses exigences et les fonctionnalités attendues de notre application.

Dans le cadre de ce projet, nous avons développé une application web pour la gestion des rendez-vous d'un cabinet médical, englobant diverses fonctionnalités telles que la gestion des patients, des rendez-vous, ainsi que la supervision des médecins et des secrétaires, sans oublier la gestion des statistiques.

Parallèlement à l'application web, nous avons également développé une application mobile dédiée aux patients. Cette application permet aux utilisateurs de rechercher des médecins, de prendre des rendez-vous en ligne et de suivre l'état de leurs demandes, offrant ainsi une solution pratique et accessible pour la gestion de leurs soins médicaux.

Malgré l'ampleur du projet et les contraintes temporelles, de nombreuses améliorations pourraient être envisagées pour cette version. Par exemple, l'intégration d'une fonctionnalité de gestion des dossiers médicaux des patients ou la gestion des médicaments. En conclusion, cette immersion sur le marché du travail s'est révélée être une expérience enrichissante et formatrice, renforçant notre désir de poursuivre une carrière professionnelle dans le domaine de l'informatique.

En outre, l'application mobile pourrait être enrichie pour permettre un suivi plus détaillé des antécédents médicaux des patients et une communication directe avec les médecins via des messageries sécurisées. Le déploiement d'algorithmes d'intelligence artificielle pour fournir des recommandations personnalisées basées sur les historiques médicaux des patients est également une piste prometteuse.

En conclusion, cette immersion sur le marché du travail s'est révélée être une expérience enrichissante et formatrice, renforçant notre désir de poursuivre une carrière professionnelle dans le domaine de l'informatique.

WEBOGRAPHIE

Site web visité :

<https://tailwindcss.com/>

<https://react.dev/learn>

<https://www.infobip.com/>

<https://tailwindcss.com/docs/guides/create-react-app>

<https://www.lucidchart.com/pages/fr/diagramme-de-sequence-uml>

<https://www.nodemailer.com/>

https://fr.wikipedia.org/wiki/Diagramme_de_s%C3%A9quence