



Projet de Fin d'Etudes

Licence Sciences et Techniques Génie Informatique

Département Informatique

Application web pour la gestion de cabinet médical



Lieu de stage : ScreenDay

Réalisé par :

EL MOKADEM Mohammed
EL KHAYARI Mohamed

Encadré par :

Pr. Adil Ben Abbou
Mr. Amine El Houti

Soutenu le **04/07/2022** devant le jury composé de :

Pr. Youness OUBENAALLA
Pr. Fatiha MRABTI
Pr. Adil Ben Abbou

REMERCIEMENT

Avant tout développement sur cette expérience professionnelle, il apparaît opportun de commencer ce rapport de stage par des remerciements.

Nous tenons à remercier dans un premier temps, toute l'équipe pédagogique de la faculté des sciences et techniques de Fès et les intervenants professionnels responsables de la LST génie informatique qui ont contribué à nous donner une formation solide tout au long de nos années d'études.

Nous remercions tout particulièrement notre encadrant académique Mr. Ben Abbou Adil, pour son encadrement, tous ses conseils, le soutien dont il nous a fait bénéficier, de ses encouragements, les aides précieuses qu'il n'a pas cessés de nous apporter tout au long de la période de ce projet et pour ses efforts afin d'assurer le bon déroulement de ce stage.

Nous tenons également à remercier et à témoigner toute notre reconnaissance, à Mr. Amine El Houti, Directeur général de la société ScreenDay, pour son accueil et la confiance qu'il nous a accordée dès notre arrivée dans la société. Pour le temps qu'il nous a consacré tout au long de cette période, sachant répondre à toutes nos interrogations. Sans oublier sa participation au cheminement de ce rapport.

Nous remercions également Pr. Fatiha MRABTI, Pr. Youness OUBENAALLA et Pr. Adil Ben Abbou les membres du jury pour nous avoir honorés en acceptant d'évaluer et de juger ce travail.

Enfin nous remercions toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce travail, ainsi qu'au bon déroulement du stage, et dont les noms ne figurent pas dans ce document.

Résumé

L'objectif de notre projet de fin d'études, présenté dans ce rapport, est la réalisation d'une application web de gestion de cabinet médical. Cette application permettra la gestion des rendez-vous, la gestion des patients, la comptabilité et statistiques, la gestion des documents (établir et imprimer), la gestion de salle d'attente et la gestion des médecins et secrétaires.

Abstract

The purpose of our end of studies' project, presented in this report, is building a medical-office management web application. This application's goal is managing appointments, patients, accounting and statistics, managing documents (establishing and printing), managing the awaiting room and managing doctors and secretaries.

Sommaire

Introduction.....	8
Chapitre 1 : Contexte Général Du Projet	9
1. Présentation du lieu de stage.....	9
2. Présentation du projet	10
2.1. Etude de l'existant	10
2.2. Cahier des charges	11
a) Objectif du projet	11
b) Travail demandé.....	11
2.3. Solution proposée	12
2.4. Diagramme de Gantt.....	12
Chapitre 2 : Analyse et Conception	13
1. Besoins fonctionnels et techniques	13
1.1. Besoins fonctionnels.....	13
1.2. Besoins techniques	14
2. Identification des acteurs	14
3. Diagrammes de cas d'utilisation.....	15
3.1. Diagramme de cas d'utilisations de la secrétaire.....	15
3.2. Diagramme de cas d'utilisations de médecin	16
3.3. Diagramme de cas d'utilisations de médecin chef	17
3.4. Diagramme de cas d'utilisations de patient.....	17
3.5. Description détaillée des cas d'utilisations.....	18
4. Diagramme de classes.....	25
5. Diagrammes de séquence.....	27
5.1. Diagramme de séquence Authentification.....	27
5.2. Diagramme de séquence Ajouter rendez-vous	28
5.3. Diagramme de séquence Etablir facture	29
5.4. Diagramme de séquence Ajouter patient à la salle d'attente.....	30
5.5. Diagramme de séquence Etablir document	30
Chapitre 3 : Réalisation de l'application.....	31
1. Architecture applicative	31
2. Outils de développement.....	32
3. Présentation de l'application.....	33

3.1. Authentification	33
3.2. Espace de secrétaires	34
a) Menu Comptabilité	34
b) Menu Rendez-vous	36
c) Menu Patients.....	37
d) Menu Documents	38
e) Menu Salle d’attente	39
3.3. Espace de médecins	39
a) Menu Salle d’attente	40
b) Menu Consultation.....	40
c) Menu Historique visites	41
3.4. Espace de médecin chef.....	42
a) Menu Médecins.....	42
b) Menu Secrétaires.....	43
3.5. Prendre ou annuler un rendez-vous en ligne	44
Conclusion	45
Références.....	46
Bibliographie.....	46
Webographie	46

Liste des figures

Figure 1 : Diagramme de Gantt	12
Figure 2 : Diagramme de cas d'utilisations de secrétaire	16
Figure 3 : Diagramme de cas d'utilisations de médecin	16
Figure 4 : Diagramme de cas d'utilisations de médecin chef	17
Figure 5 : Diagramme de cas d'utilisations de patient	17
Figure 6 : Diagramme de classes	26
Figure 7 : Diagramme de séquence de cas d'utilisation s'authentifier	27
Figure 8 : Diagramme de séquence de cas d'utilisation ajouter rendez-vous	28
Figure 9 : Diagramme de séquence de cas d'utilisation établir une facture	29
Figure 10 : Diagramme de séquence de cas d'utilisation ajouter patient à la salle d'attente	30
Figure 11 : Diagramme de séquence de cas d'utilisation établir document	30
Figure 12 : Architecture du modèle MVC	31
Figure 13 : Interface d'authentification	33
Figure 14 : Interface d'espace de secrétaires	34
Figure 15 : Menu Statistiques	34
Figure 16 : Menu Historiques de paiements	35
Figure 17 : Menu Ajouter une Facture	35
Figure 18 : Menu Rendez-vous	36
Figure 19 : Ajouter rendez-vous	36
Figure 20 : Menu Patient	37
Figure 21 : Ajouter Patient	37
Figure 22 : Menu Documents	38
Figure 23 : Interface imprimer document	38
Figure 24 : Menu Salle d'attente	39
Figure 25 : Interface d'espace de médecins	39
Figure 26 : Menu Salle d'attente	40
Figure 27 : Menu Dossier médical	40
Figure 28 : Menu Etablir document	41
Figure 29 : Menu Historique visites	41
Figure 30 : Interface d'espace de médecin chef	42
Figure 31 : Menu Médecins	42
Figure 32 : Ajouter un médecin	43
Figure 33 : Menu Secrétaires	43
Figure 34 : Ajouter une secrétaire	44
Figure 35 : Prendre ou annuler un rendez-vous en ligne	44

Liste des tables

Tableau 1 : Information de la société ScreenDay	9
Tableau 2 : Acteurs et leurs rôles	15
Tableau 3 : Descriptif de cas d'utilisation Authentification	18
Tableau 4 : Descriptif de cas d'utilisation Ajouter patient	19
Tableau 5 : Descriptif de cas d'utilisation Ajouter rendez-vous	20
Tableau 6 : Descriptif de cas d'utilisation Imprimer document.....	21
Tableau 7 : Descriptif de cas d'utilisation Etablir documents.....	22
Tableau 8 : Descriptif de cas d'utilisation Ajouter facture.....	22
Tableau 9 : Descriptif de cas d'utilisation Ajouter patient à la salle d'attente	23
Tableau 10 : Descriptif de cas d'utilisation Ajouter médecin.....	24
Tableau 11 : Descriptif de cas d'utilisation Ajouter secrétaire	25

Liste des acronymes

AJAX : Asynchronous JavaScript And Xml

CSS : Cascading Style Sheets

HTML : HyperText Markup Language

MVC : Model-View-Controller

PHP : HyperText Preprocessor

SQL : Structured Query Language

UML : Unified Modeling Language

XAMPP : Cross Apache MariaDB Perl PHP

Introduction

Dans le but de valider nos études acquises durant les 6 semestres au sein de la Faculté des sciences et techniques de Fès et dans l'intention d'obtenir le diplôme de la licence en sciences et techniques, nous sommes amenés à effectuer un stage de fin d'études de 2 mois.

Relativement à notre troisième année de licence à la Faculté, spécialité génie informatique, nous avons eu l'avantage d'effectuer un stage au sein de l'entreprise **ScreenDay** qui a duré du 25-04-2022 jusqu'à 25-06-2022.

Pendant cette période de stage, nous avons réalisé une application de gestion du cabinet médical pour faciliter plusieurs tâches aux employés d'un cabinet médical, en utilisant le Framework LARAVEL qui est basé sur PHP, et aussi JQuery, AJAX, HTML 5, CSS3, et MYSQL.

Dans la suite on va adopter la planification suivante :

Le premier chapitre présente le cadre général du projet, en décrivant la société d'accueil, la problématique de projet, et la solution proposée pour pallier aux problèmes survenus.

Le deuxième chapitre concerne l'analyse fonctionnelle du projet, et la conception de la solution finale.

Le troisième chapitre se concentre sur la description des outils techniques utilisés, y compris les langages de programmation, les technologies adaptées, et l'environnement de développement. Ainsi que la description des principales interfaces de l'application.

Chapitre 1 : Contexte Général Du Projet

1. Présentation du lieu de stage

La société **ScreenDay** est une agence web qui développe des solutions spécifiques complexes sur-mesure afin de répondre aux exigences des entreprises de différents horizons, qu'il s'agisse d'un site web, d'une application à haute valeur ajoutée, d'une extension dédiée à accomplir une tâche, ou d'une fonctionnalité plus modeste mais indispensable aux yeux de client tel que :

- Marketing digital
- Social media management
- Production and design
- Digital business solutions

Secteurs	Services d'information
Taille de l'entreprise	2-10 employés
Site web	https://www.screenday.ma
Numéro de téléphone	06 90 40 20 20
Siège social	Fès, FES MEKNES
Fondée en	2016
Location	ALLAL BEN ABDELLAH RÉSIDENCE LYOUBIA. 6EME ÉTAGE, BUREAU N 35, Fès 30000
Domaines	Création site web, Référencement SEO, Création site e-commerce, Communication production audiovisuelle, Communication digitale Experte web marketing

Tableau 1 : Information de la société ScreenDay

2. *Présentation du projet*

2.1. *Etude de l'existant*

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

La gestion d'un cabinet médical est un emploi complexe. Cependant, depuis les premières années de construction des cabinets médicaux, les médecins s'orientaient à un objectif bien déterminé, celui-ci se résumait à comment travailler avec les patients à l'aide de moyens faciles et rapides. Mais malgré tous les efforts des médecins ils ont trouvé des difficultés, à savoir :

- L'archivage des dossiers des patients sur des feuilles peut engendrer leur perte et demande beaucoup de temps pour trouver un dossier, ou pour modifier un élément dans un dossier. Cela peut obliger le médecin à refaire un nouveau dossier. Ceci pose un véritable problème de gestion d'organisation et de consultations. Une manière quasi-manuelle rend le travail lourd et compliqué ce qui pose un nombre important des problèmes tels que :

- Recherche difficile sur les registres.
- Détérioration des archives à cause de leur utilisation trop fréquente.
- Difficulté de stockage.
- Risque de perte des archives.
- Problèmes de calcul des statistiques.

- La gestion de rendez-vous peut être complexe, car les rendez-vous s'enregistrent juste dans les papiers, donc la perte de ces papiers causera la perte des rendez-vous.

- Et plusieurs autres difficultés qu'ils ont été rencontré par les employés de cabinet médical, ce qui nous a amené à créer cette application.

2.2. *Cahier des charges*

a) Objectif du projet

C'est la gestion d'un cabinet médical qui consiste à gérer les patients, c'est à dire la gestion des dossiers, la gestion des rendez-vous, la gestion de salle d'attente, etc....

b) Travail demandé

Une secrétaire peut :

- Générer une facture de consultation au patient.
- Ajouter ou annuler un rendez-vous au patient.
- Gérer la salle d'attente par ajouter ou supprimer un patient après une recherche.
- Ajouter, modifier ou supprimer un patient.
- Consulter l'historique des paiements, et les statistiques (nombre patients, rendez-vous, nombre patients à la salle d'attente et les revenus).

Un médecin peut :

- Effectuer des consultations aux patients.
- Savoir les patients dans la salle d'attente, il sélectionne un patient puis peut consulter le dossier médical du patient, ou rédiger un document (ordonnance, certificat médicale, compte rendu) à ce dernier.
- Consulter l'historique de ses visites.

Le médecin chef peut :

- Gérer les médecins et les secrétaires, il peut les consulter, ajouter, modifier ou supprimer.
- Accéder à toutes les tâches du médecin.

Toute utilisation de l'application nécessite une authentification par un login et un mot de passe.

2.3. *Solution proposée*

Nous avons proposé **une application web** pour mieux organiser les consultations et la gestion des rendez-vous, de factures, ... avec des interfaces simples et faciles à utiliser selon l'architecture MVC.

Cette application a pour buts de :

- Faciliter la gestion des patients.
- Faciliter la gestion des rendez-vous.
- Faciliter la gestion de comptabilité.
- Garantir une authentification sécurisée.
- Faciliter la consultation des rapports de patients et établir les différents types de documents.
- Faciliter la gestion de la salle d'attente

2.4. *Diagramme de Gantt*

Pour représenter visuellement l'avancement des différentes activités durant notre stage, nous avons établie le diagramme de Gantt suivant :

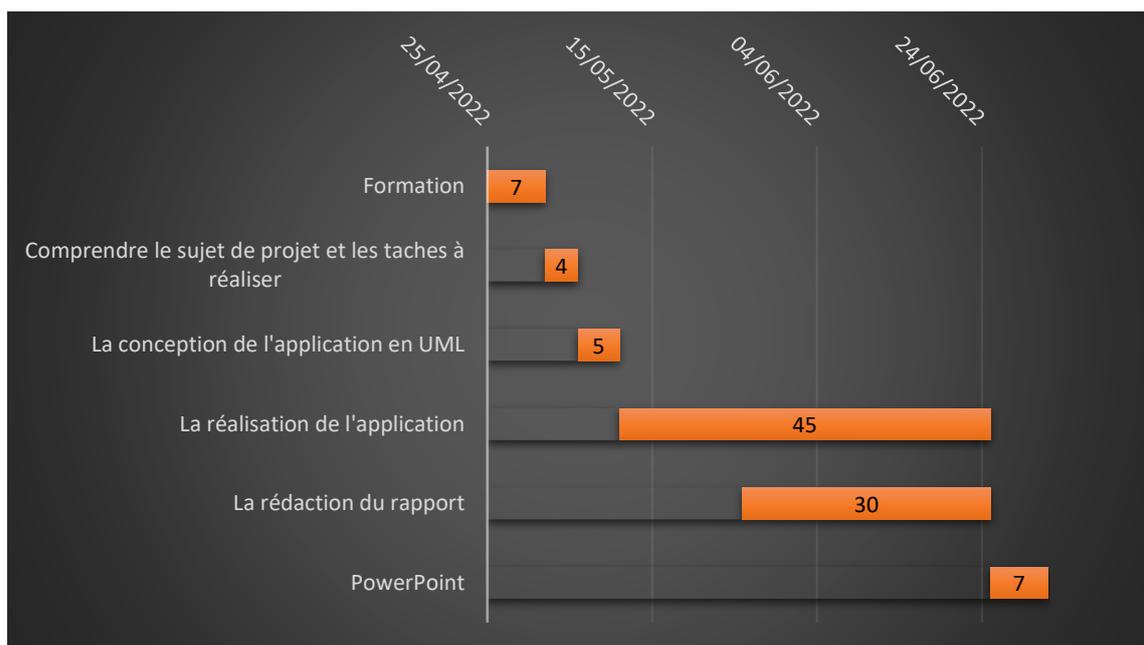


Figure 1 : Diagramme de Gantt

Chapitre 2 : Analyse et Conception

1. *Besoins fonctionnels et techniques*

1.1. *Besoins fonctionnels*

Notre application doit satisfaire les besoins fonctionnels suivants :

Patient :

- **Prendre un rendez-vous :** les patients peuvent prendre des rendez-vous à distance.
- **Annuler un rendez-vous :** les patients peuvent annuler leurs rendez-vous.

Secrétaire :

- **Gestion des patients :** les patients sont ajoutés au système par une secrétaire, qui peut aussi modifier et supprimer un patient.
- **Gestion des rendez-vous :** une secrétaire peut ajouter ou annuler un rendez-vous.
- **Gestion de la salle d'attente :** une secrétaire peut ajouter un patient à la salle d'attente ou le supprimer.
- **Gestion de la comptabilité et statistiques :** une secrétaire peut consulter les statistiques et les historiques du paiement et peut aussi gère les factures (établir facture, supprimer facture, modifier facture.)
- **Gestion du document :** une secrétaire peut imprimer les documents d'un patient.

Médecin :

- **Gestion de la salle d'attente :** Un médecin peut consulter la liste des patients dans la salle d'attente et valider la visite d'un patient.
- **Gestion des documents :** le médecin peut soit consulter le dossier médical du patient ou bien établir les documents (Compte rendu, ordonnance, certificat médicale) pour ce dernier.

- **Historique des visites** : Chaque médecin peut consulter l'historique de ses visites.

Médecin chef :

- **Gestion des médecins et des secrétaires** : les secrétaires et les médecins sont ajoutés par le médecin chef qui peut aussi les modifier ou les supprimer.
- **Peut aussi accéder à tous les menus du médecin**

1.2. Besoins techniques

- **Validité** : L'application doit être capable de remplir exactement les fonctions définies par le cahier des charges.
- **Efficacité et Fiabilité** : L'application doit être capable de fonctionner correctement dans des conditions anormales.
- **Maniabilité et Réutilisabilité** : Le code de l'application doit être lisible et compréhensible afin de faciliter la réutilisation, d'apprentissage, d'interprétation des erreurs et de rattrapage en cas d'erreur d'utilisation.
- **Sécurité** : L'application doit sécuriser les informations qui ne doivent pas être accessibles à tout le monde.

2. Identification des acteurs

L'acteur représente une entité externe qui interagit directement avec le système étudié, en échangeant de l'information (en entrée et en sortie). On trouve les acteurs en observant les utilisateurs directs du système, les responsables de la maintenance, ainsi que les autres systèmes qui interagissent avec le système.

Dans notre projet, on peut distinguer les quatre acteurs suivants :

- **Secrétaire.**
- **Médecin.**
- **Médecin chef.**
- **Patient.**

Utilisateur	Leur Rôle
<i>Secrétaire</i>	<ul style="list-style-type: none"> ▪ Gestion des rendez-vous. ▪ Gestion des patients. ▪ Gestion de la comptabilité. ▪ Gestion de salle d'attente. ▪ Impression des documents
<i>Médecin</i>	<ul style="list-style-type: none"> ▪ Gestion des documents (Etablir les différents documents de patients (Compte rendu, ordonnance, certificat médical), consultation de dossier médical de patient). ▪ Consulter salle d'attente.
<i>Médecin chef</i>	<ul style="list-style-type: none"> ▪ Gestion des secrétaires. ▪ Gestion des médecins. ▪ Plus les rôles de médecin.
<i>Patient</i>	<ul style="list-style-type: none"> ▪ Prendre rendez-vous. ▪ Annuler rendez-vous.

Tableau 2 : Acteurs et leurs rôles

3. Diagrammes de cas d'utilisation

Un cas d'utilisation est utilisé pour définir le comportement d'un système ou la sémantique de toute autre entité sans révéler sa structure interne.

Chaque cas d'utilisation spécifie une séquence d'action, y compris des variantes, que l'entité réalise, en interagissant avec les acteurs de l'entité. La responsabilité d'un cas d'utilisation est de spécifier un ensemble d'instances, où une instance de cas d'utilisation représente une séquence d'actions que le système réalise et qui fournit un résultat observable par l'acteur.

3.1. Diagramme de cas d'utilisations de la secrétaire

Une secrétaire peut ajouter des rendez-vous ou bien les annuler, ajouter des patients, établir les factures, imprimer les documents, etc...

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

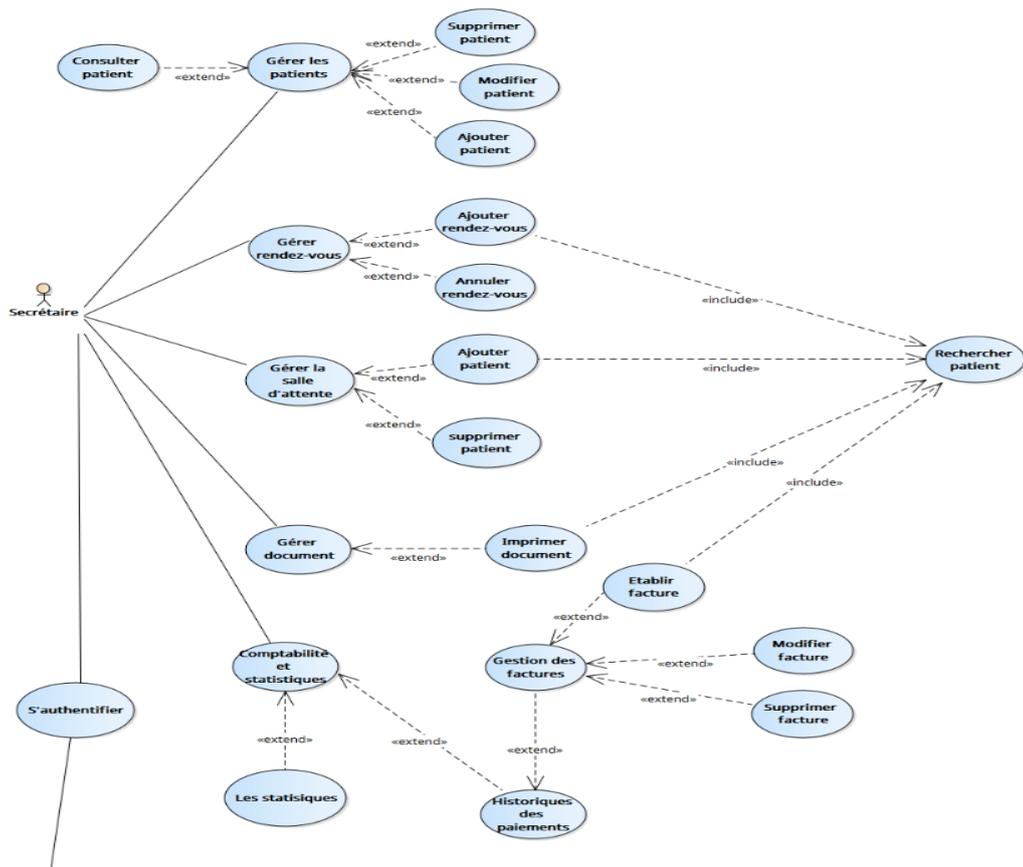


Figure 2 : Diagramme de cas d'utilisations de secrétaire

3.2. Diagramme de cas d'utilisations de médecin

Un médecin peut établir les différents documents (Ordonnance, certificat médicale, compte rendu), consulter la liste de patients dans la salle d'attente ...

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

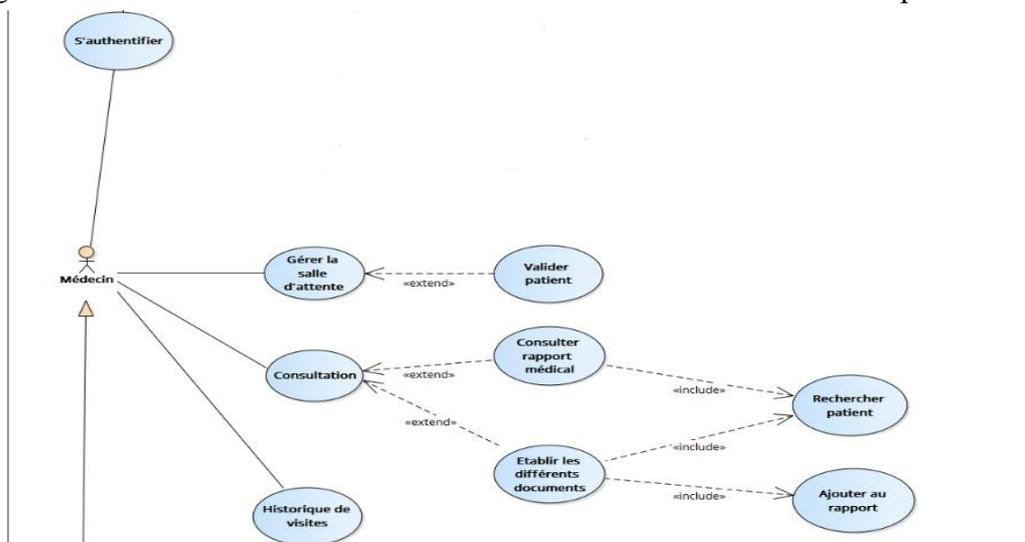


Figure 3 : Diagramme de cas d'utilisations de médecin

3.3. Diagramme de cas d'utilisations de médecin chef

En plus des rôles de médecin, le médecin chef peut gérer les médecins et les secrétaires.

Le diagramme de cas d'utilisation suivant illustre en détails les rôles assurés par le médecin chef :

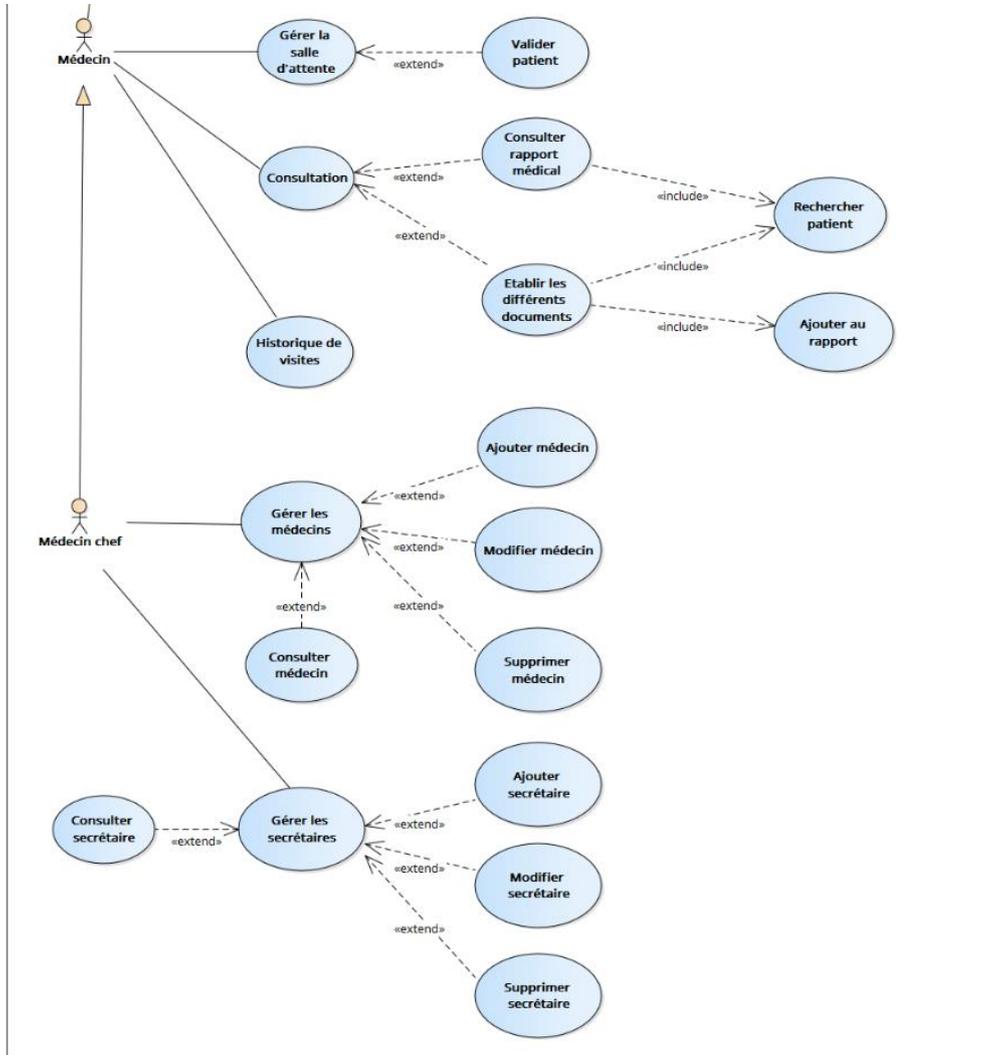


Figure 4 : Diagramme de cas d'utilisations de médecin chef

3.4. Diagramme de cas d'utilisations de patient

Un patient peut prendre un rendez-vous ou annuler un rendez-vous en ligne.

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

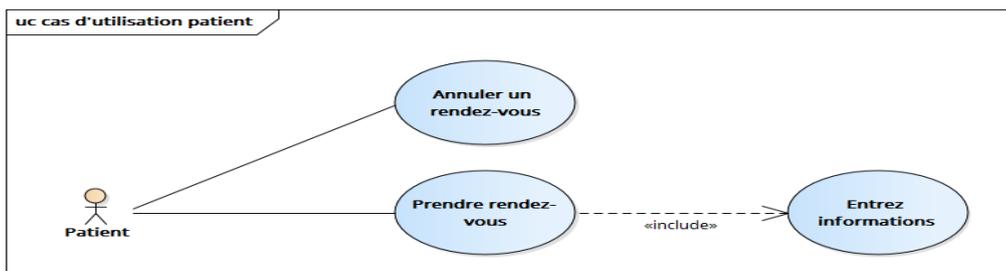


Figure 5 : Diagramme de cas d'utilisations de patient

3.5. Description détaillée des cas d'utilisations

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

- **Authentification**

Nom de cas d'utilisation	Authentification
Acteur	Secrétaire, médecin, médecin chef
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	<p style="text-align: center;">Scenario 1</p> <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. <p style="text-align: center;">Scenario 2</p> <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 sont bien remplis. ▪ Les informations entrées sont erronées. ▪ Un message d'erreur s'affiche.

Tableau 3 : Descriptif de cas d'utilisation Authentification

- **Ajouter patient**

Nom de cas d'utilisation	Ajouter patient
Acteur	Secrétaire
Pré condition	La secrétaire 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. ▪ La secrétaire remplit les champs du formulaire. ▪ La secrétaire 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 style="text-align: center;">Scenario 1</p> <ul style="list-style-type: none"> ▪ Le formulaire s'affiche. ▪ La secrétaire remplit les champs du formulaire. ▪ La secrétaire 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 style="text-align: center;">Scenario 2</p> <ul style="list-style-type: none"> ▪ Le formulaire s'affiche. ▪ La secrétaire remplit les champs du formulaire. ▪ La secrétaire 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 4 : Descriptif de cas d'utilisation Ajouter patient

- **Ajouter rendez-vous.**

Nom de cas d'utilisation	Ajouter rendez-vous
Acteur	Secrétaire
Pré condition	Un patient veut prendre un rendez-vous
Post condition	La secrétaire veut ajouter un rendez-vous.
Scenario normal	<ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient existe. ▪ Le patient n'a aucune facture à payer. ▪ 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 style="text-align: center;">Scenario 1</p> <ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient n'existe pas. ▪ Le système affiche un message d'erreur pour informer la secrétaire. <p style="text-align: center;">Scenario 2</p> <ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient existe. ▪ La secrétaire choisit la date du rendez-vous et le médecin. ▪ La disponibilité de date et de médecin est vérifiée. ▪ Soit la date ou le médecin ne sont pas disponibles. ▪ Le système affiche un message d'erreur pour informer la secrétaire. <p style="text-align: center;">Scenario 3</p> <ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient existe. ▪ Le patient a une facture qui n'est pas payée. ▪ Le système affiche un message d'erreur pour informer la secrétaire.

Tableau 5 : Descriptif de cas d'utilisation Ajouter rendez-vous

- **Imprimer documents.**

Nom de cas d'utilisation	Imprimer documents
Acteur	Secrétaire
Pré condition	Après une recherche sur le patient par son nom la secrétaire veut imprimer un document pour ce dernier.
Post condition	La secrétaire veut imprimer un document.
Scenario normal	<ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient existe. ▪ Afficher tous les documents relatifs au patient. ▪ Imprimer le document sélectionné.
Scenario alternative	<p style="text-align: center;">Scenario 1</p> <ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient n'existe pas. ▪ Le système affiche un message d'erreur pour informer la secrétaire. <p style="text-align: center;">Scenario 2</p> <ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient existe ▪ Le patient n'a aucun document. ▪ Un message est affiché pour informer la secrétaire.

Tableau 6 : Descriptif de cas d'utilisation Imprimer document

- **Etablir documents**

Nom de cas d'utilisation	Etablir documents
Acteur	Médecin, médecin chef
Pré condition	Après une recherche sur le patient par son nom le médecin peut établir un document pour ce dernier.
Post condition	Le médecin veut établir un document.
Scenario normal	<ul style="list-style-type: none"> ▪ Le médecin entre le nom de patient ▪ Le patient existe. ▪ Afficher les types de documents à établir. ▪ Le médecin choisit le type de document à établir et remplit les champs de formulaire affiché et valide. ▪ Le document est établi.

Scenario alternative	<p style="text-align: center;">Scenario</p> <ul style="list-style-type: none"> ▪ Le médecin entre le nom de patient. ▪ Le patient n'existe pas. ▪ Le système affiche un message d'erreur pour informer la secrétaire.
----------------------	---

Tableau 7 : Descriptif de cas d'utilisation Etablir documents

- **Ajouter facture.**

Nom de cas d'utilisation	Ajouter facture
Acteur	Secrétaire
Pré condition	La secrétaire fait une recherche sur le patient qu'elle veut ajouter une facture à.
Post condition	La secrétaire veut ajouter une facture.
Scenario normal	<ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient existe. ▪ Le patient a une facture à payer. ▪ Un formulaire s'affiche. ▪ La secrétaire le remplit et valide. ▪ La facture est ajoutée.
Scenario alternative	<p style="text-align: center;">Scenario 1</p> <ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient n'existe pas. ▪ Le système affiche un message d'erreur pour informer la secrétaire. <p style="text-align: center;">Scenario 2</p> <ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient existe. ▪ Le patient n'a aucune facture à payer. ▪ Le système affiche un message d'erreur pour informer la secrétaire.

Tableau 8 : Descriptif de cas d'utilisation Ajouter facture

- **Ajouter patient à la salle d'attente.**

Nom de cas d'utilisation	Ajouter patient à la salle d'attente
Acteur	Secrétaire
Pré condition	Un patient arrive à la salle d'attente du cabinet et la secrétaire veut l'ajouter à la liste de salle d'attente.
Post condition	La secrétaire veut ajouter un patient à la salle d'attente.
Scenario normal	<ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient existe. ▪ Le patient est ajouté.
Scenario alternative	<p style="text-align: center;">Scenario 1</p> <ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient n'a pas de rendez-vous aujourd'hui ou bien est déjà dans la salle d'attente. ▪ Le système affiche un message d'erreur pour informer la secrétaire. <p style="text-align: center;">Scenario 2</p> <ul style="list-style-type: none"> ▪ La secrétaire entre le nom de patient. ▪ Le patient n'existe pas. ▪ Le système affiche un message d'erreur pour informer la secrétaire.

Tableau 9 : Descriptif de cas d'utilisation Ajouter patient à la salle d'attente

- **Ajouter médecin.**

Nom de cas d'utilisation	Ajouter médecin
Acteur	Médecin chef
Pré condition	Le médecin chef veut ajouter un nouveau médecin
Post condition	Un formulaire s'affiche au médecin chef à remplir
Scenario normal	<ul style="list-style-type: none"> ▪ Le formulaire s'affiche. ▪ Le médecin chef remplit les champs du formulaire. ▪ Le médecin chef 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 style="text-align: center;">Scenario 1</p> <ul style="list-style-type: none"> ▪ Le formulaire s’affiche. ▪ Le médecin chef remplit les champs du formulaire. ▪ Le médecin chef appuie sur le bouton ajouter. ▪ Validation des champs. ▪ Les champs ne sont bien remplis. ▪ Le système affiche un message d’erreur pour informer Le médecin chef. <p style="text-align: center;">Scenario 2</p> <ul style="list-style-type: none"> ▪ Le formulaire s’affiche. ▪ Le médecin chef remplit les champs du formulaire. ▪ Le médecin chef 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 Le médecin chef.
----------------------	---

Tableau 10 : Descriptif de cas d’utilisation Ajouter médecin

- **Ajouter secrétaire.**

Nom de cas d’utilisation	Ajouter secrétaire
Acteur	Médecin chef
Pré condition	Le médecin chef veut ajouter une nouvelle secrétaire
Post condition	Un formulaire s’affiche au médecin chef à remplir
Scenario normal	<ul style="list-style-type: none"> ▪ Le formulaire s’affiche. ▪ Le médecin chef remplit les champs du formulaire. ▪ Le médecin chef appuie sur le bouton ajouter. ▪ Validation des champs. ▪ Les champs sont bien remplis. ▪ La secrétaire n’existe pas. ▪ Enregistre la secrétaire dans la base des données.
Scenario alternative	<p style="text-align: center;">Scenario 1</p> <ul style="list-style-type: none"> ▪ Le formulaire s’affiche. ▪ Le médecin chef remplit les champs du formulaire. ▪ Le médecin chef appuie sur le bouton ajouter.

	<ul style="list-style-type: none"> ▪ Validation des champs. ▪ Les champs ne sont bien remplis. ▪ Le système affiche un message d’erreur pour informer Le médecin chef. <p style="text-align: center;">Scenario 2</p> <ul style="list-style-type: none"> ▪ Le formulaire s’affiche. ▪ Le médecin chef remplit les champs du formulaire. ▪ Le médecin chef appuie sur le bouton ajouter. ▪ Validation des champs. ▪ Les champs sont bien remplis. ▪ La secrétaire existe déjà. ▪ Le système affiche un message d’erreur pour informer Le médecin chef.
--	---

Tableau 11 : Descriptif de cas d’utilisation Ajouter secrétaire

4. Diagramme de classes

Définition :

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 :

- CompteRendu
- Certificat
- Ordonnance
- Document
- DossierMédical
- Employé
- Secrétaire
- Médecin
- Médecin chef
- Patient
- SalleAttente
- Consultation
- Facture
- Rendez-vous

Le diagramme de classes suivant montre les associations entre les différentes classes :

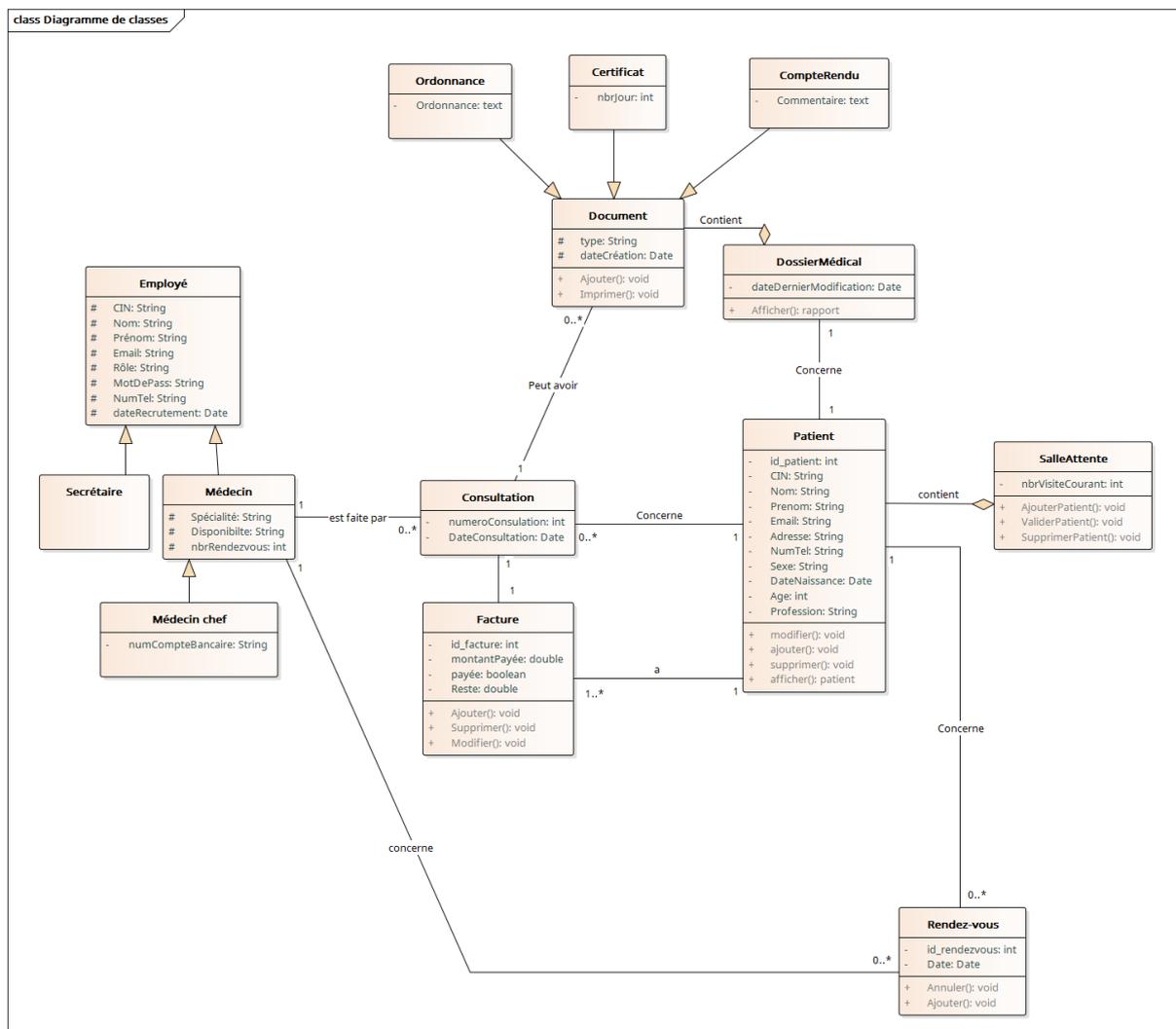


Figure 6 : Diagramme de classes

5. Diagrammes de séquence

Les diagrammes des séquences permettent de représenter les interactions entre objet selon un point de vue temporel .l'accent est mis sur la chronologie des envois de messages.

- **Scénario** : une liste d'actions qui décrivent une interaction entre un acteur et le système.
- **Interaction** : Un comportement qui comprend un ensemble de messages échangés par un ensemble d'objets dans un certain contexte pour accomplir une certaine tâche.
- **Message** : Un message est une transmission d'information unidirectionnelle entre deux objets, l'objet émetteur et l'objet récepteur.

Par la suite nous allons présenter les principaux diagrammes de séquence.

5.1. Diagramme de séquence Authentification

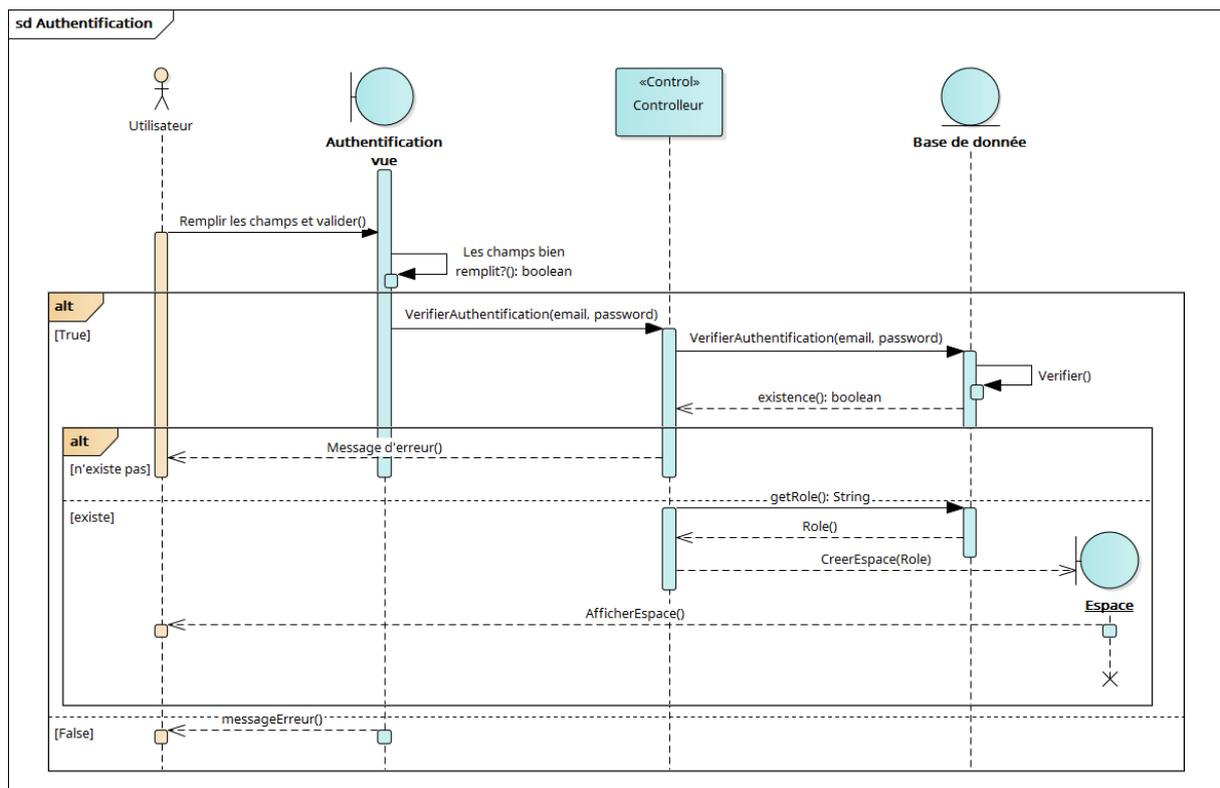


Figure 7 : Diagramme de séquence de cas d'utilisation s'authentifier

5.2. Diagramme de séquence Ajouter rendez-vous

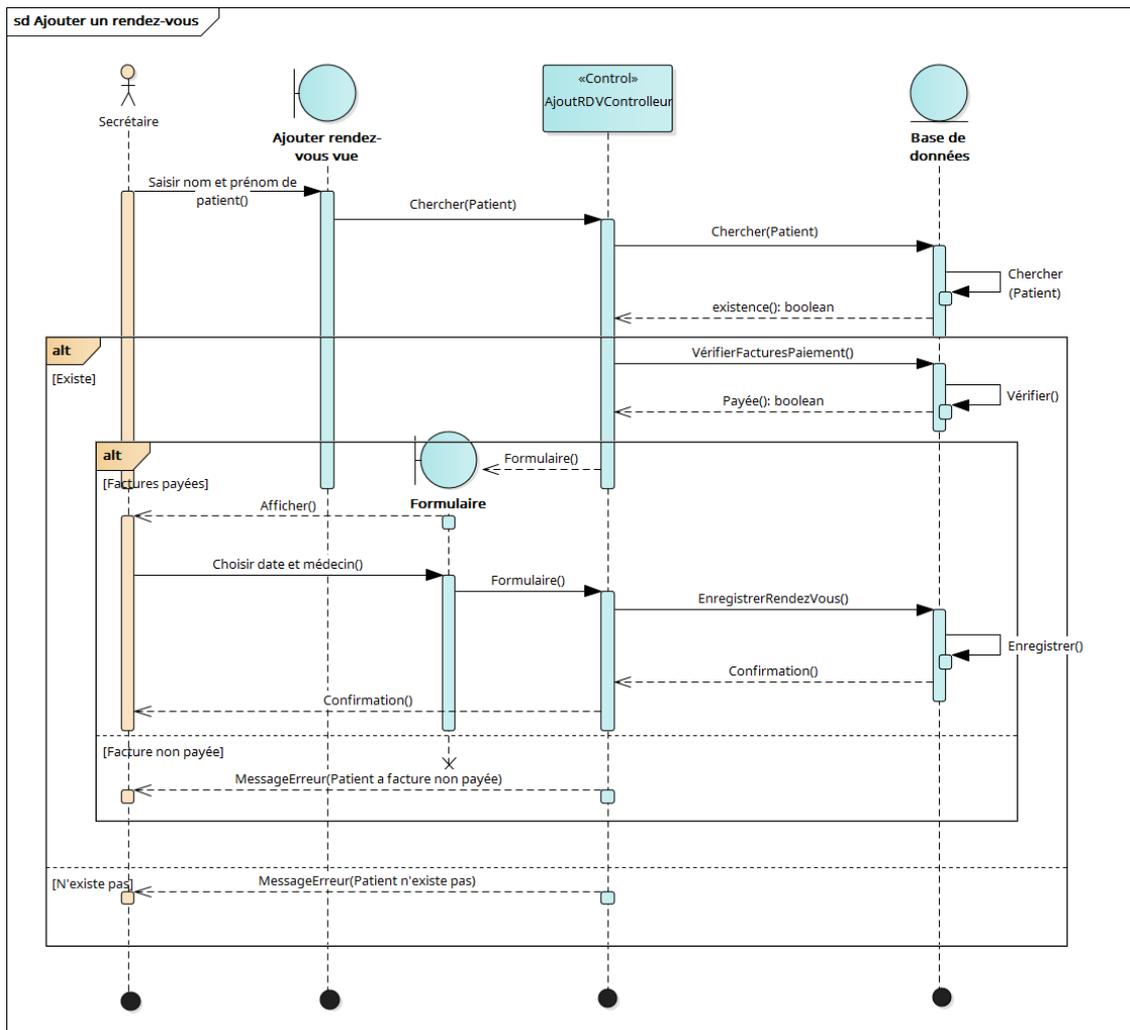


Figure 8 : Diagramme de séquence de cas d'utilisation ajouter rendez-vous

5.3. Diagramme de séquence Etablir facture

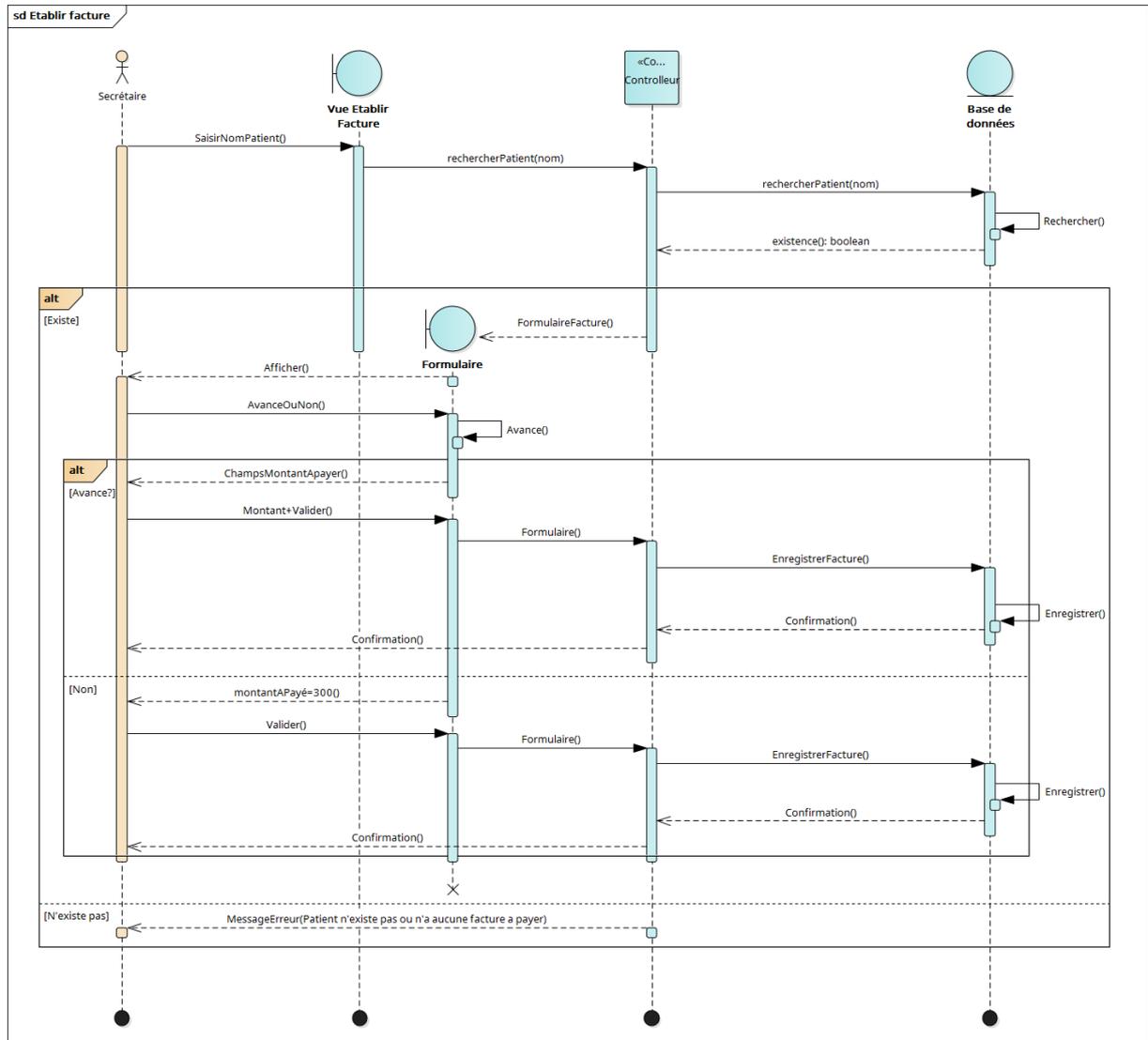


Figure 9 : Diagramme de séquence de cas d'utilisation établir une facture

5.4. Diagramme de séquence Ajouter patient à la salle d'attente

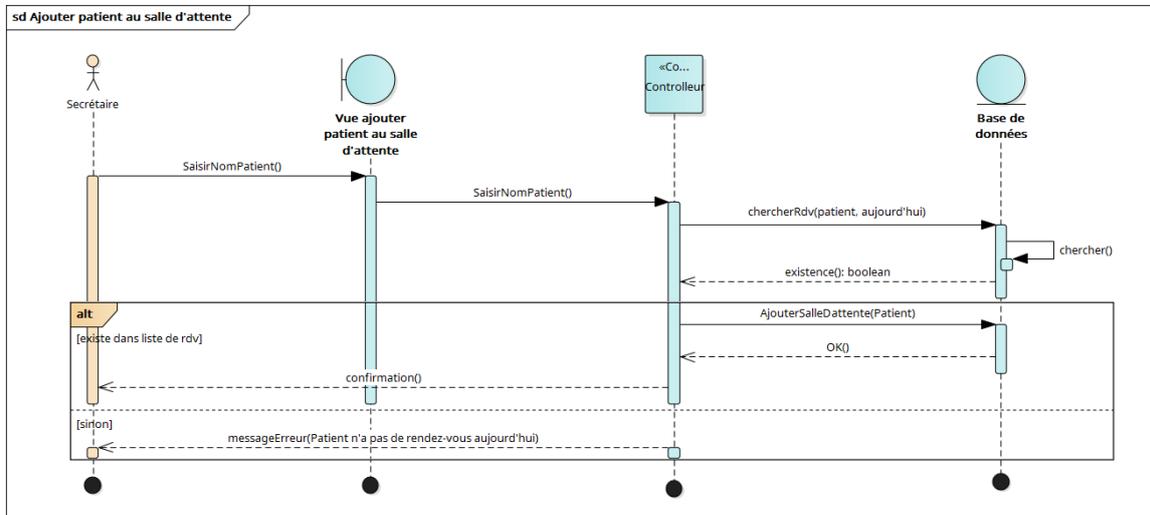


Figure 10 : Diagramme de séquence de cas d'utilisation ajouter patient à la salle d'attente

5.5. Diagramme de séquence Etablir document

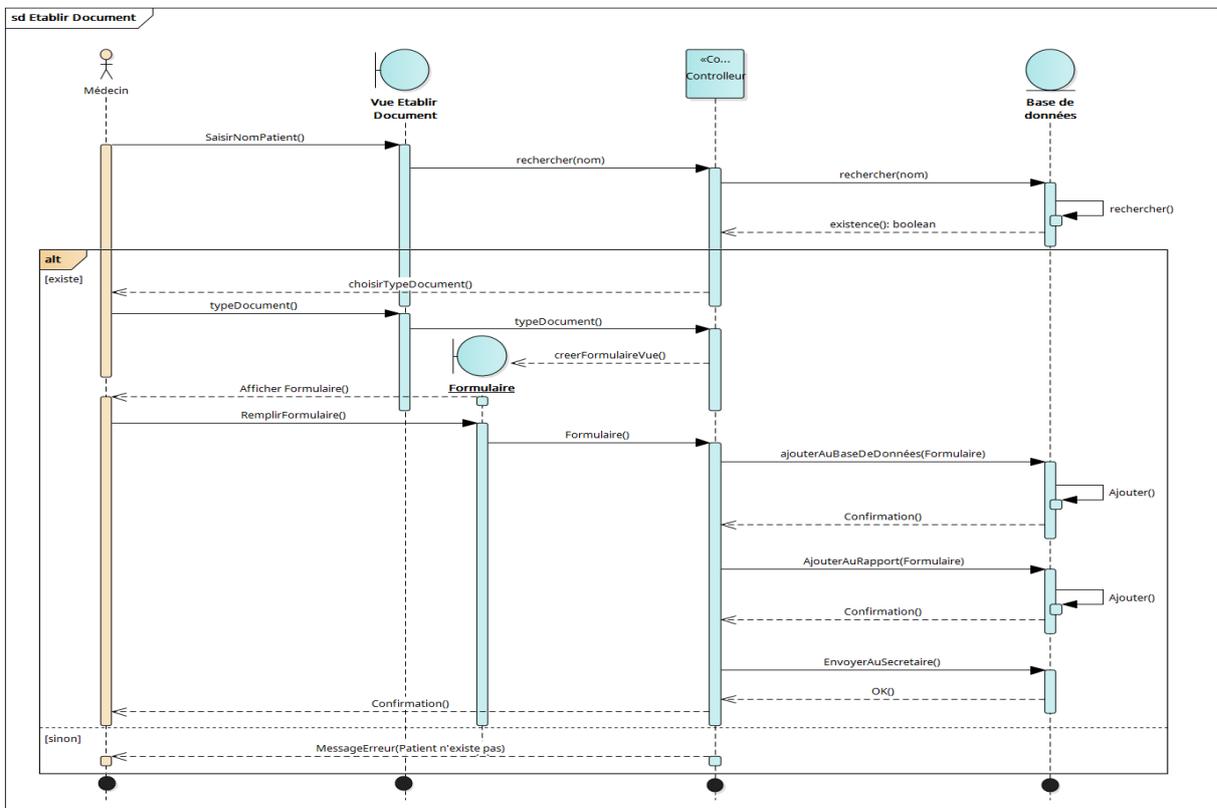


Figure 11 : Diagramme de séquence de cas d'utilisation établir document

Chapitre 3 : Réalisation de l'application

1. Architecture applicative

Dans notre projet, nous avons suivi l'architecture MVC. Cette architecture est un concept très puissant qui intervient dans la réalisation d'une application. Son principal intérêt est la séparation des données (modèle), de l'affichage (vue) et des actions (contrôleur), ce qui assure la clarté de l'architecture et simplifie la tâche du développeur responsable de la maintenance et de l'amélioration du projet. Les différentes interactions entre le modèle, la vue et le contrôleur sont résumées par le schéma de la figure suivante

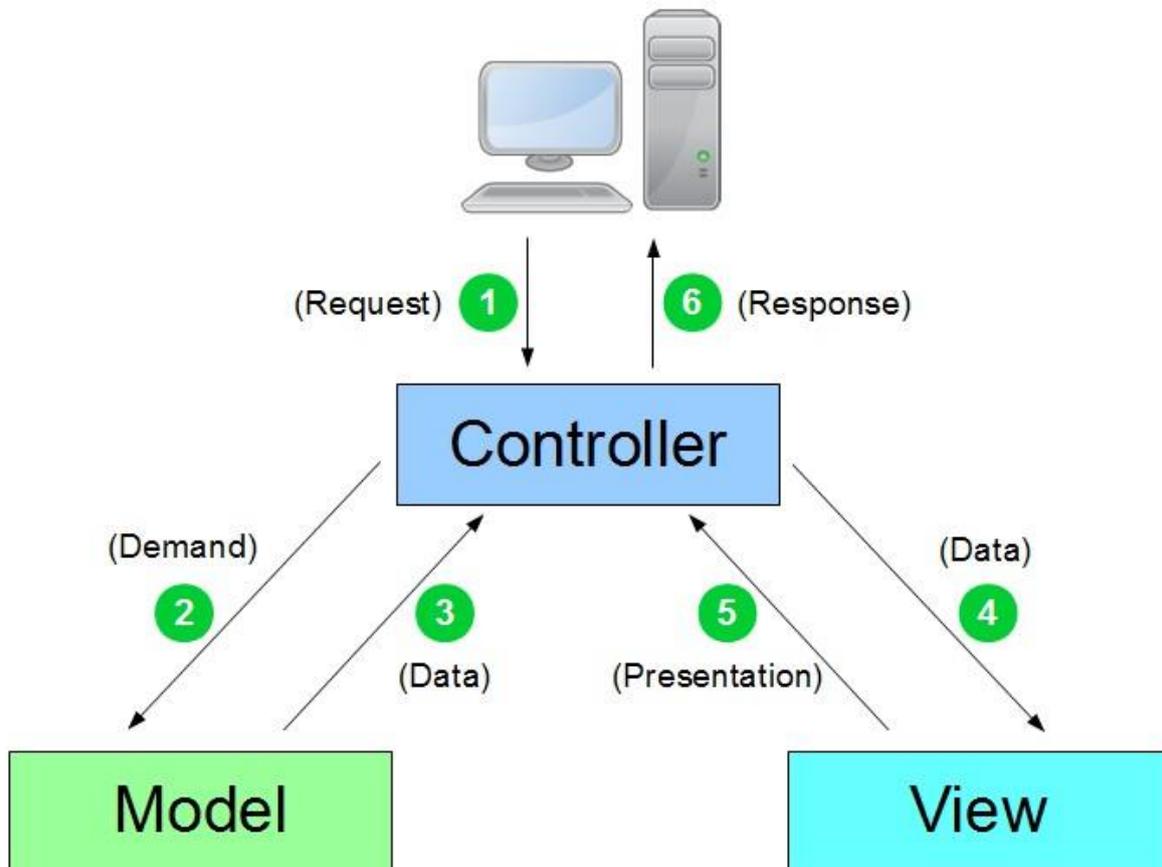


Figure 12 : Architecture du modèle MVC

2. Outils de développement



Enterprise Architect est un logiciel de modélisation et de conception UML, édité par la société australienne [Sparx Systems](#). Couvrant, par ses fonctionnalités, l'ensemble des étapes du cycle de conception d'application, en passant par les étapes d'analyse, les modèles de conception et les étapes de test et d'entretien. Cet outil permet de bien schématiser notre application, pour passer de la conception vers la réalisation. Il facilite la représentation des diagrammes UML.



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'auto complétion, la coloration syntaxique, le débogage, et les commandes git.



phpMyAdmin est une application Web pour les systèmes de gestion de base de données MySQL réalisée en PHP.



LARAVEL est un Framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet.



HyperText Preprocessor, plus connu sous son sigle **PHP** est un langage de programmation libre principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale.



XAMPP est un ensemble de logiciels permettant de mettre en place un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres (**X** (cross) **A**pache **M**aria **D**B **P**erl **P**HP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide.

3. *Présentation de l'application*

Dans cette section nous allons présenter les principales interfaces de notre application.

3.1 *Authentification*

Se Connecter

Connectez vous à votre compte pour continuer

EMAIL

example@example.com

MOT DE PASSE [Mot de passe oublié?](#)

Mot de passe

Se Connecter

Figure 13 : Interface d'authentification

Il est important de noter que chaque opération ne peut être réalisée qu'après authentification. Seul le médecin chef qui se charge de définir le login et le mot de passe de chaque employé. Cette page permet au médecin chef, médecin ou secrétaire de se connecter si les données entrées sont valides.

Commençant par l'espace de secrétaires :

3.2. Espace de secrétaires

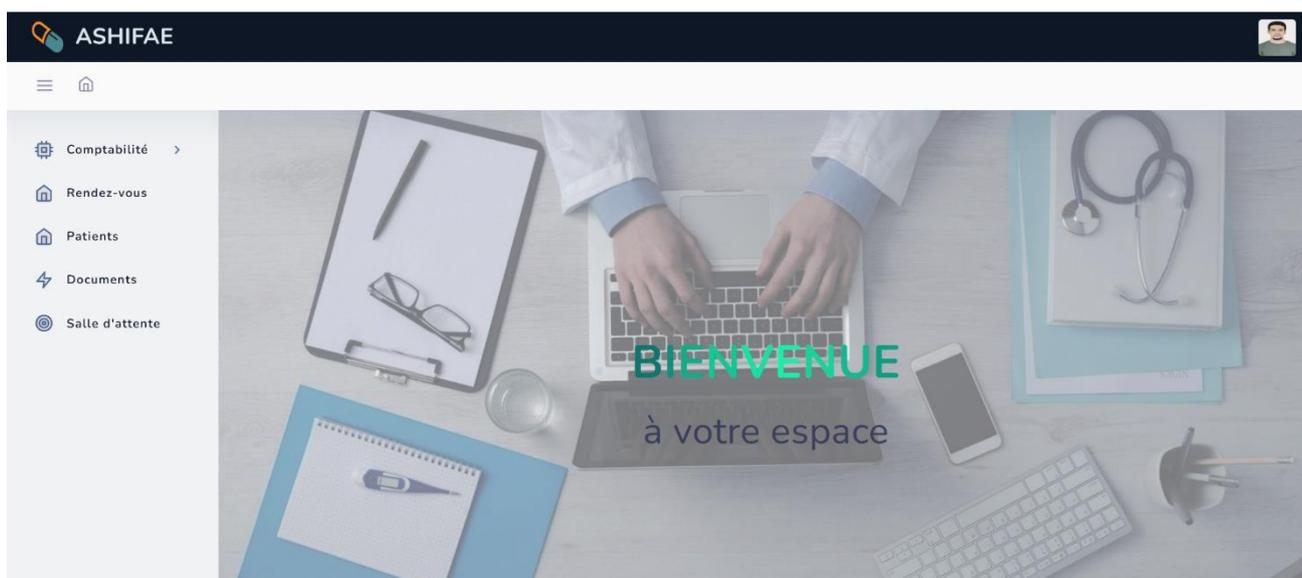


Figure 14 : Interface d'espace de secrétaires

Dans l'espace de secrétaire on a 5 menus principaux :

- a) **Menu Comptabilité**
 - **Menu Statistiques**

Dans ce menu la secrétaire peut consulter les différentes statistiques du cabinet.

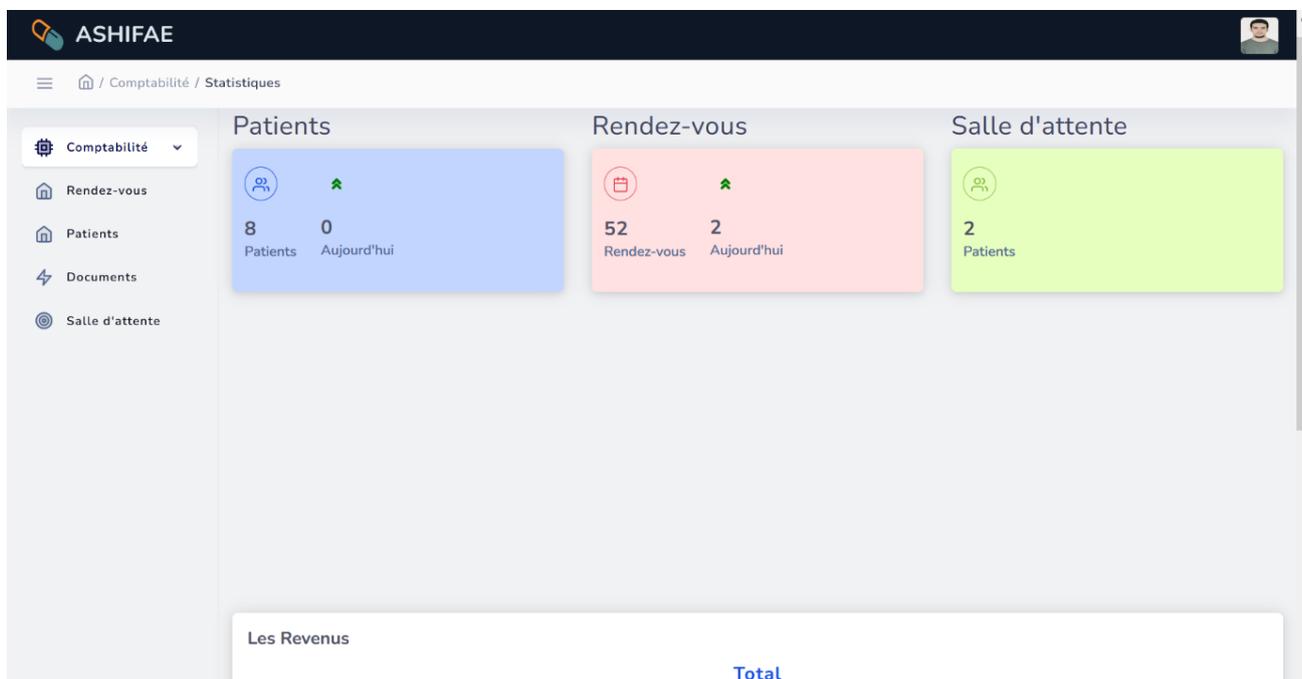


Figure 15 : Menu Statistiques

- **Menu Historiques de paiements**

Dans ce menu la secrétaire peut soit ajouter, supprimer ou modifier une facture.

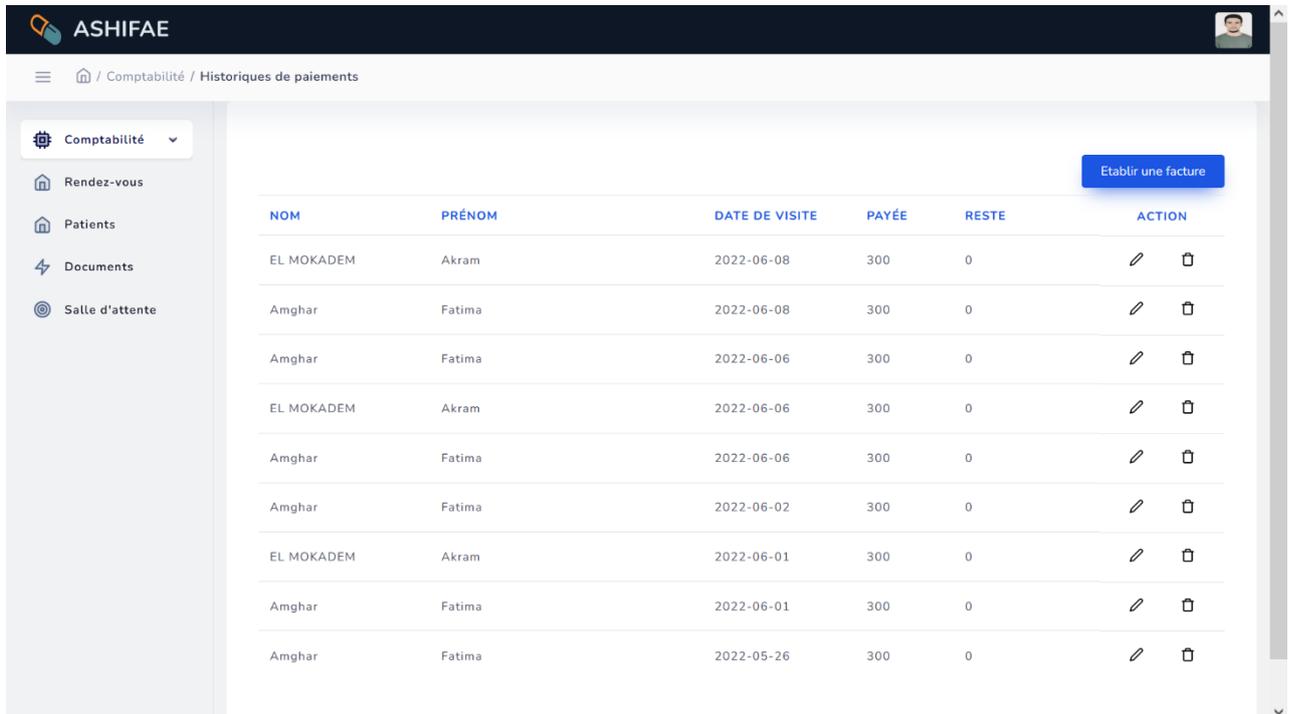


Figure 16 : Menu Historiques de paiements

- **Menu Ajouter une facture**

Dans cette interface un formulaire pour ajouter facture s'affiche.

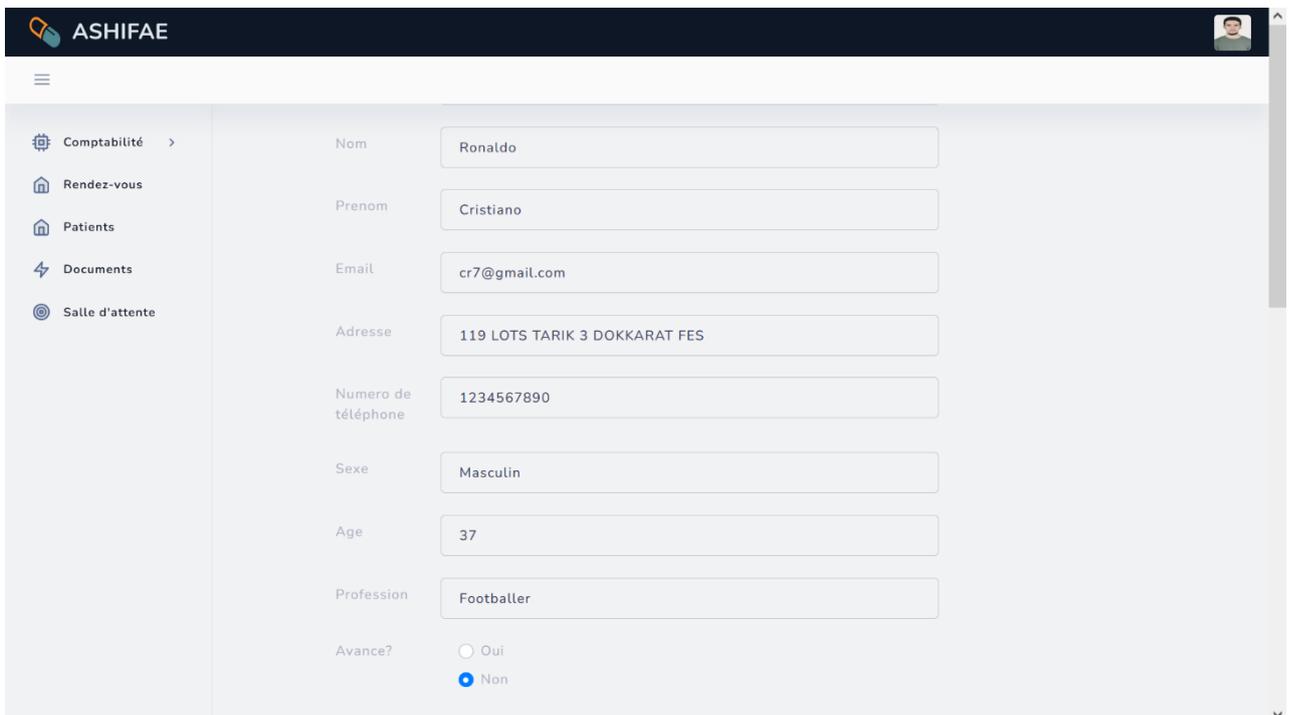


Figure 17 : Menu Ajouter une Facture

b) Menu Rendez-vous

Dans ce menu la secrétaire peut consulter les rendez-vous d'un jour choisi, ajouter ou annuler un rendez-vous.

The screenshot shows the 'Rendez-vous' menu in the ASHIFAE system. The interface includes a sidebar with navigation options: Comptabilité, Rendez-vous (selected), Patients, Documents, and Salle d'attente. The main content area displays the date '06/22/2022' and a 'Get' button. Below this is a table of appointments for that date.

NUMÉRO DE RENDEZ-VOUS	HEURES	NOM DE PATIENT	NOM DE MÉDECIN	STATUS	ACTION
127	10:30	EL MOKADEM Ismail	EL MOKADEM EL MOKADEM	Annulé	⊗
128	10:30	EL MOKADEM Ismail	EL MOKADEM EL MOKADEM	Annulé	⊗
122	11:20	Amghar Fatima	Imgharene Imgharene	Annulé	⊗
131	12:30	EL MOKADEM Akram	EL MOKADEM EL MOKADEM	à la salle d'attente	⊗
132	13:40	Amghar Fatima	EL MOKADEM EL MOKADEM	à la salle d'attente	⊗

Figure 18 : Menu Rendez-vous

• Ajouter rendez-vous

Dans ce menu la secrétaire peut ajouter un rendez-vous à un patient choisi, pour les heures qui sont en rouges, le rendez-vous ne doit pas être ajouté dans ces heures.

The screenshot shows the 'Ajouter rendez-vous' menu in the ASHIFAE system. The interface includes a sidebar with navigation options: Comptabilité, Rendez-vous (selected), Patients, Documents, and Salle d'attente. The main content area displays the date '06/22/2022' and a 'choisir' button. Below this is a grid of time slots for the day, with some slots highlighted in red to indicate they are unavailable for booking.

Date: 06/22/2022					
Medecins					
EL MOKADEM Mohammed					
9:00	9:10	9:20	9:30	9:40	9:50
10:00	10:10	10:20	10:30	10:40	10:50
11:00	11:10	11:20	11:30	11:40	11:50
12:00	12:10	12:20	12:30	12:40	12:50
13:00	13:10	13:20	13:30	13:40	13:50
15:00	15:10	15:20	15:30	15:40	15:50
16:00	16:10	16:20	16:30	16:40	16:50
17:00	17:10	17:20	17:30	17:40	17:50

Figure 19 : Ajouter rendez-vous

c) Menu Patients

Dans ce menu la secrétaire peut consulter les informations d'un patient, modifier, supprimer ou ajouter un patient.

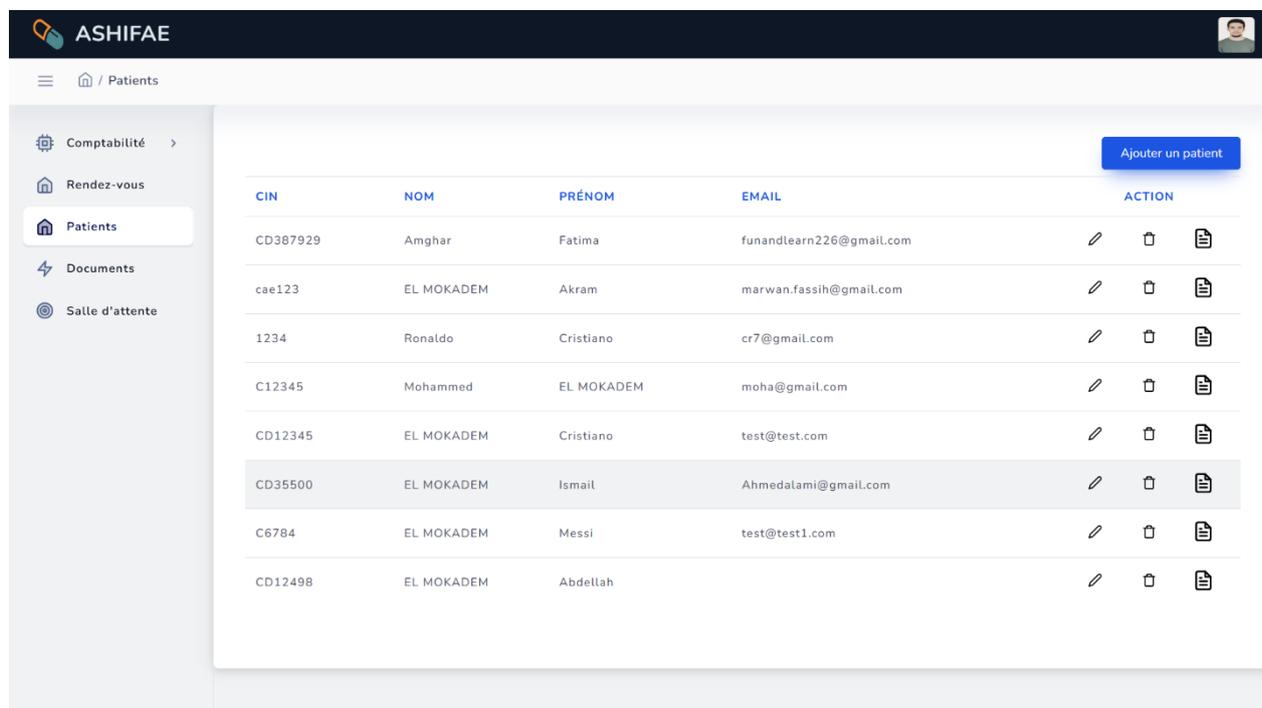


Figure 20 : Menu Patient

- Ajouter Patient

Dans cette interface un formulaire pour ajouter patient s'affiche.

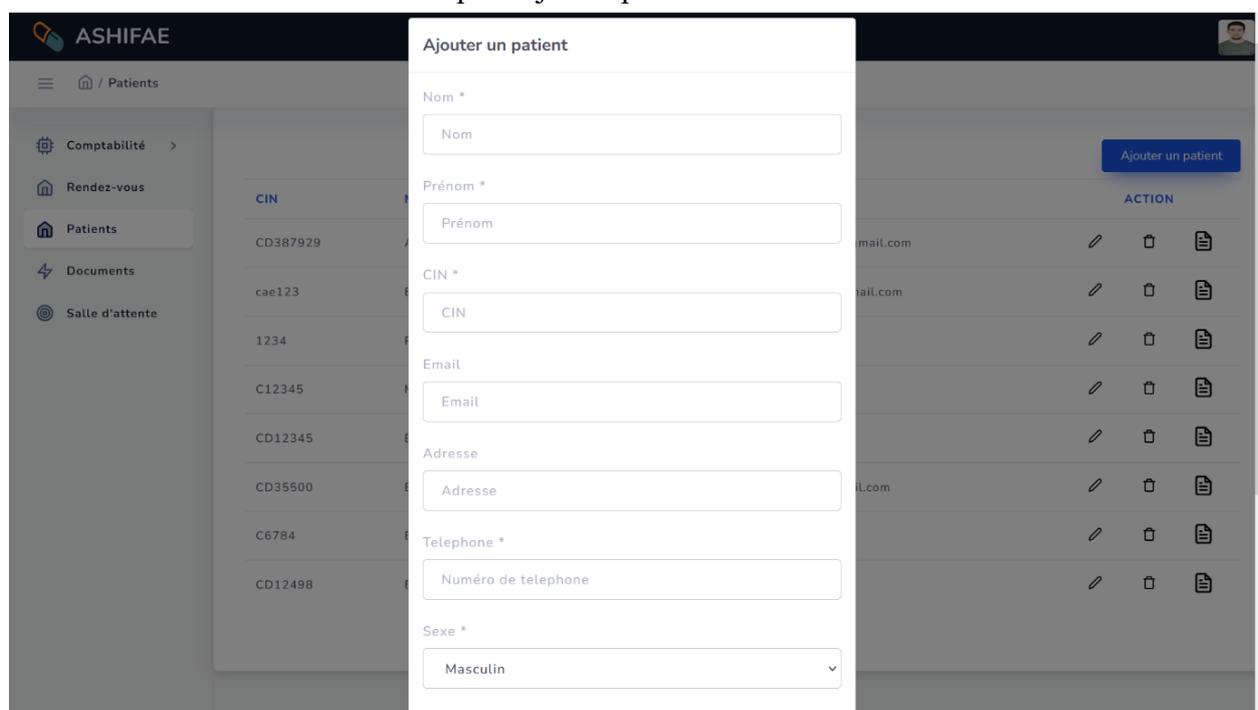


Figure 21 : Ajouter Patient

d) Menu Documents

Après la saisie de nom de patient un tableau de documents de patient s'affiche et qui donne la possibilité d'imprimer un document.

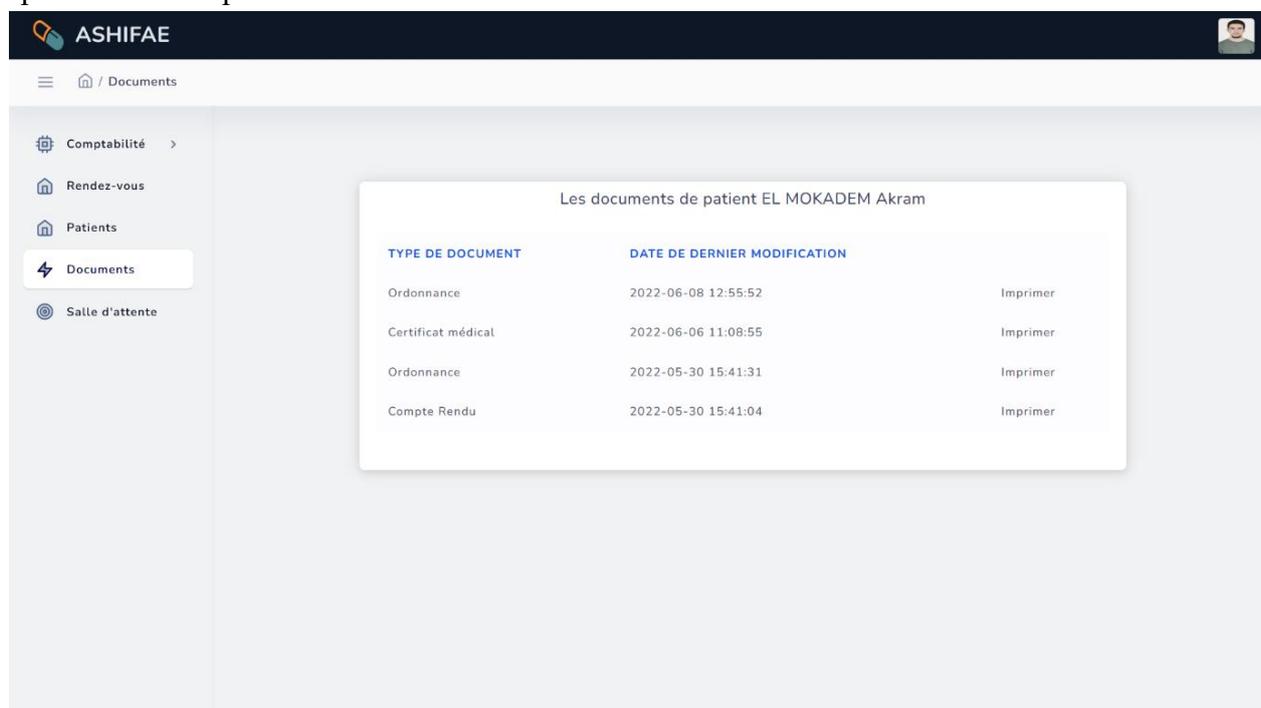


Figure 22 : Menu Documents

- Interface imprimer un document

Quand on clique sur imprimer dans le tableau de documents cette interface s'affiche.

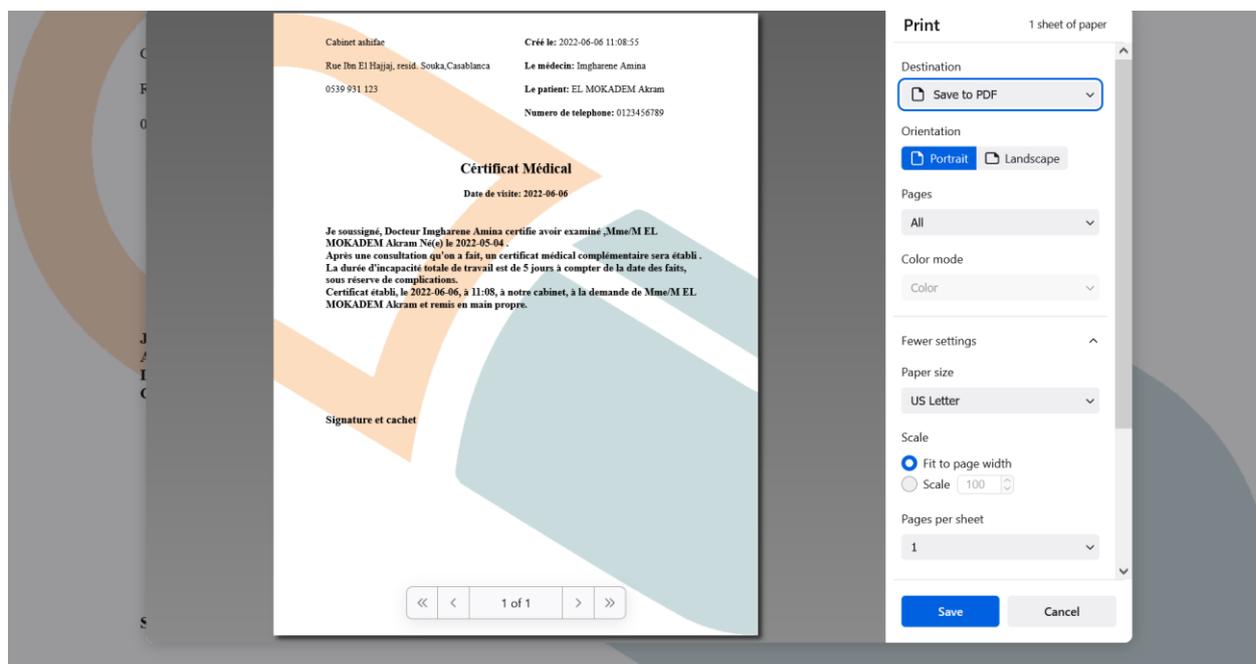


Figure 23 : Interface imprimer document

e) *Menu Salle d'attente*

Dans ce menu la secrétaire peut ajouter ou supprimer un patient à la salle d'attente.

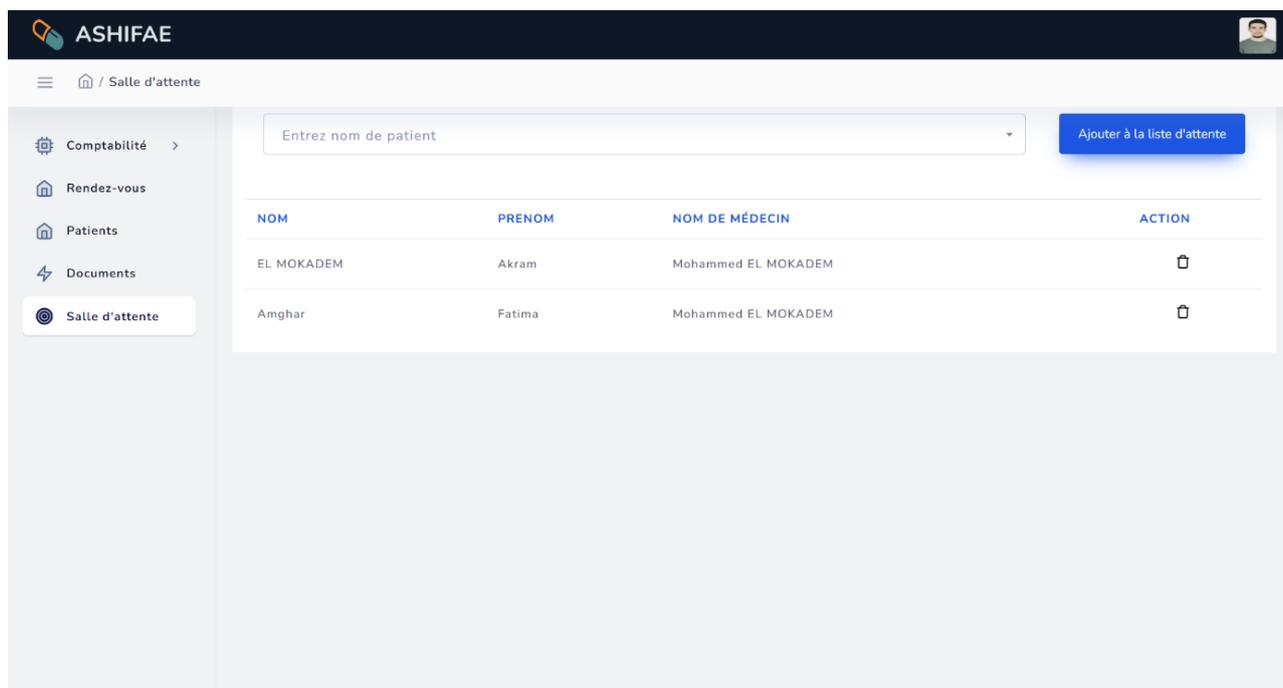


Figure 24 : Menu Salle d'attente

Maintenant on passe à l'espace de médecins :

3.3. *Espace de médecins*

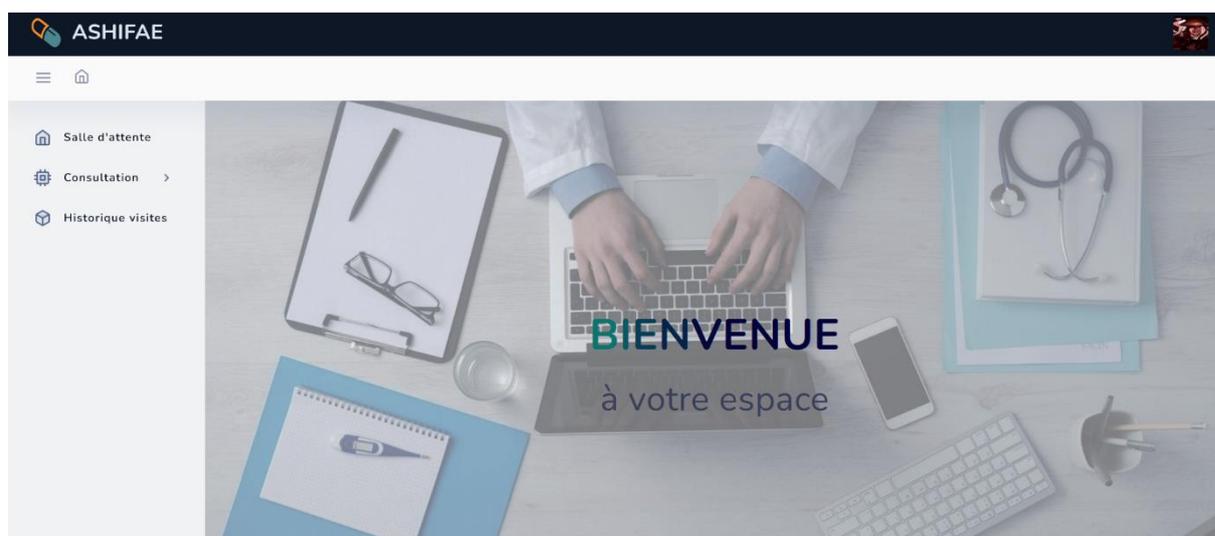


Figure 25 : Interface d'espace de médecins

Dans cette interface on a 5 menus principaux :

a) Menu Salle d'attente

Dans ce menu chaque médecin peut savoir les patients qui ont un rendez-vous avec lui et sont dans la salle d'attente et il peut aussi valider leurs visites.



Figure 26 : Menu Salle d'attente

b) Menu Consultation

- **Menu Dossier médical**

Dans ce menu le médecin peut consulter le dossier médical de chaque patient



Figure 27 : Menu Dossier médical

- **Menu Etablir document**

Dans ce menu le médecin peut établir différents documents (Certificat médicale, ordonnance, Compte rendu) à un patient.

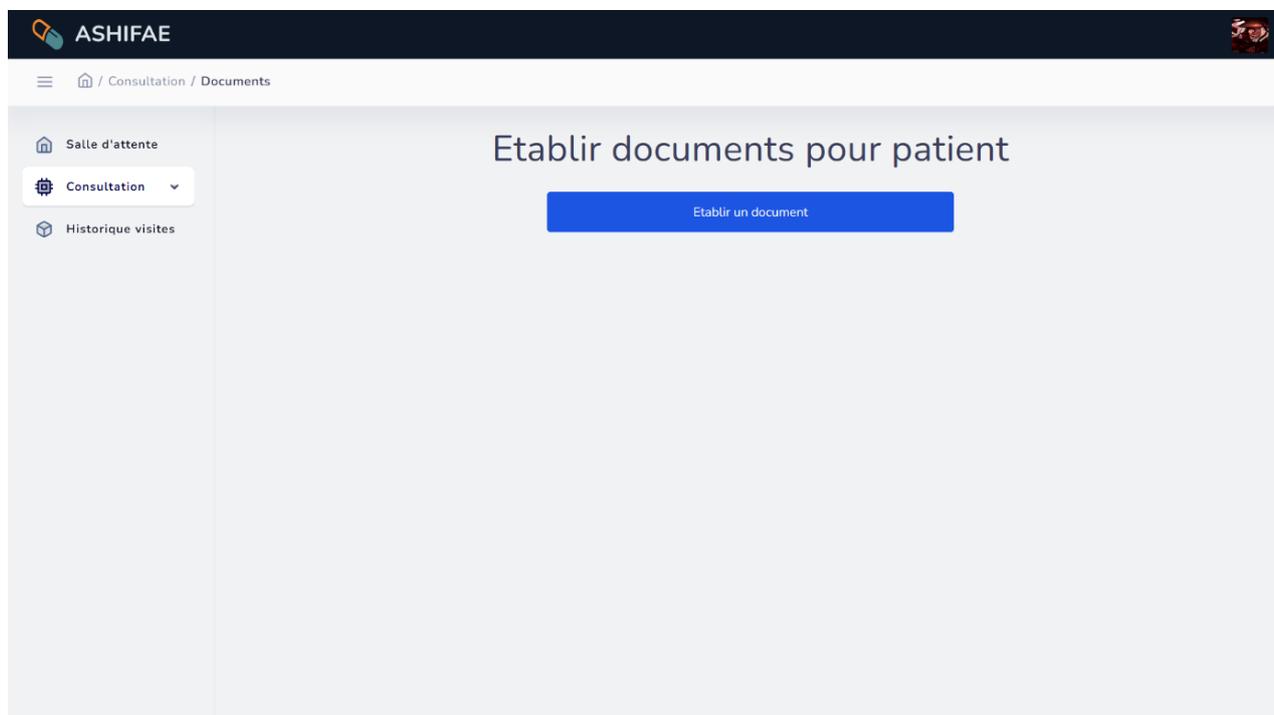


Figure 28 : Menu Etablir document

c) Menu Historique visites

Dans ce menu chaque médecin peut consulter l'historique de ses visites.

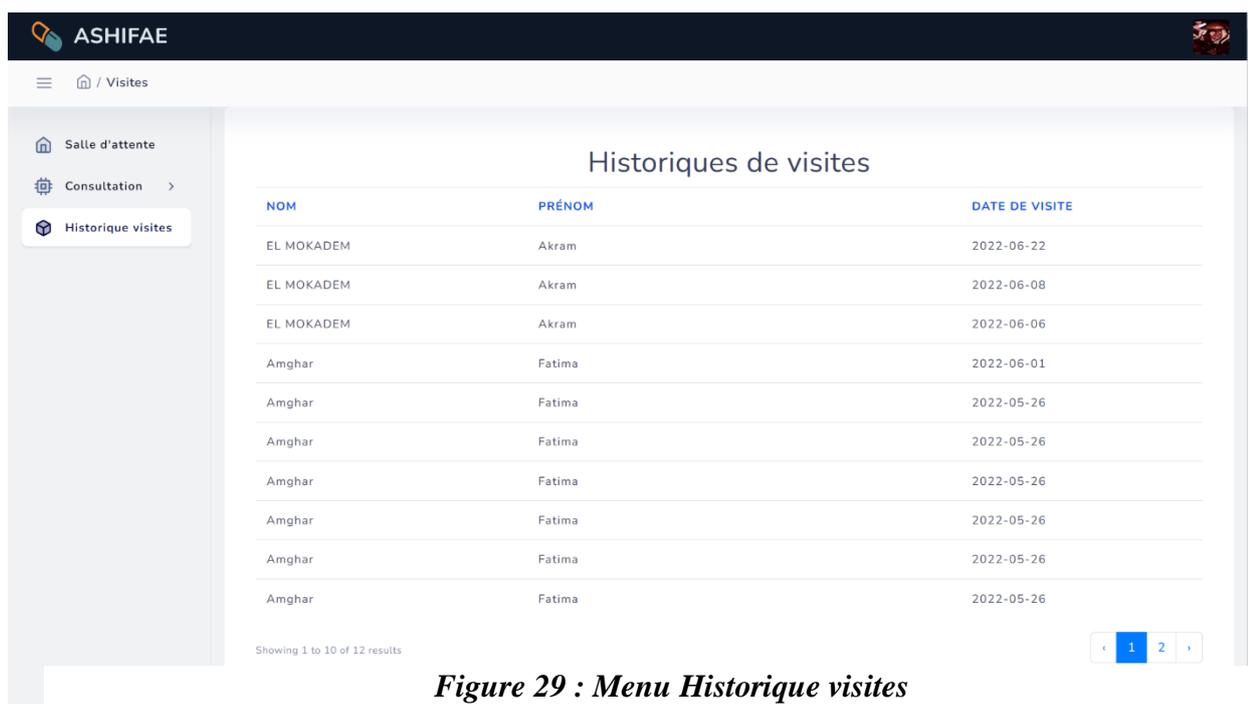


Figure 29 : Menu Historique visites

On passe à l'espace de médecin chef :

3.4. Espace de médecin chef

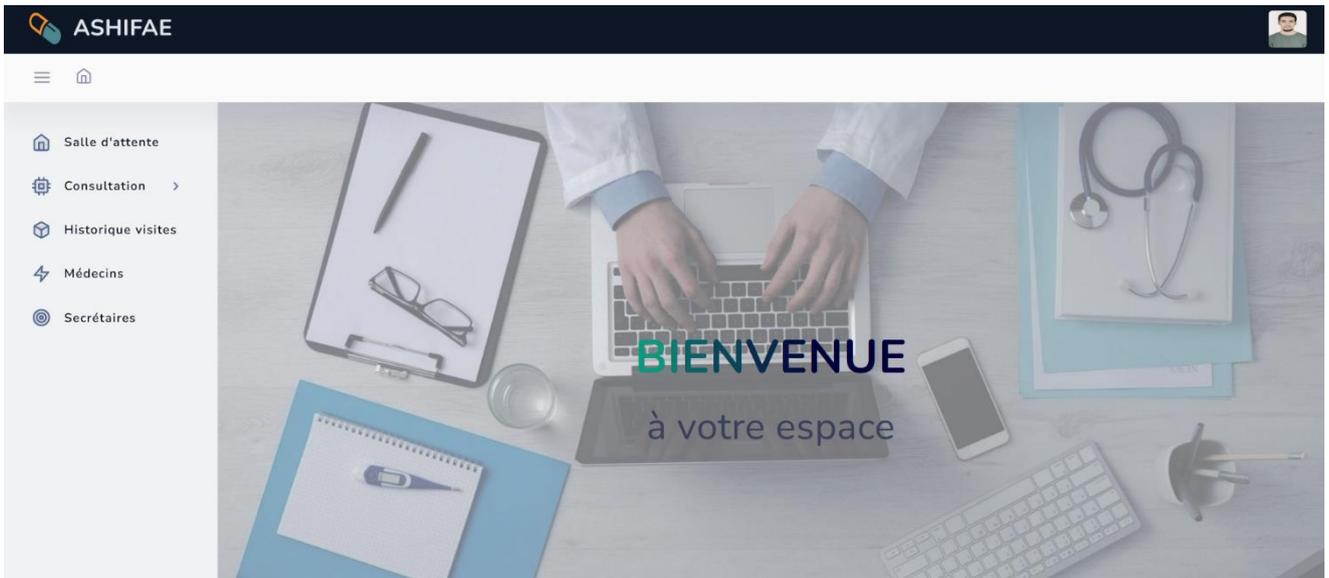


Figure 30 : Interface d'espace de médecin chef

Dans l'interface de médecin chef, on a 5 menus principaux, 3 d'eux sont les mêmes que ceux d'espace de médecin, et 2 sont propre au médecin chef, gestion de médecins et gestion de secrétaires :

a) Menu Médecins

Dans ce menu le médecin chef peut soit consulter, ajouter, modifier ou supprimer un médecin.

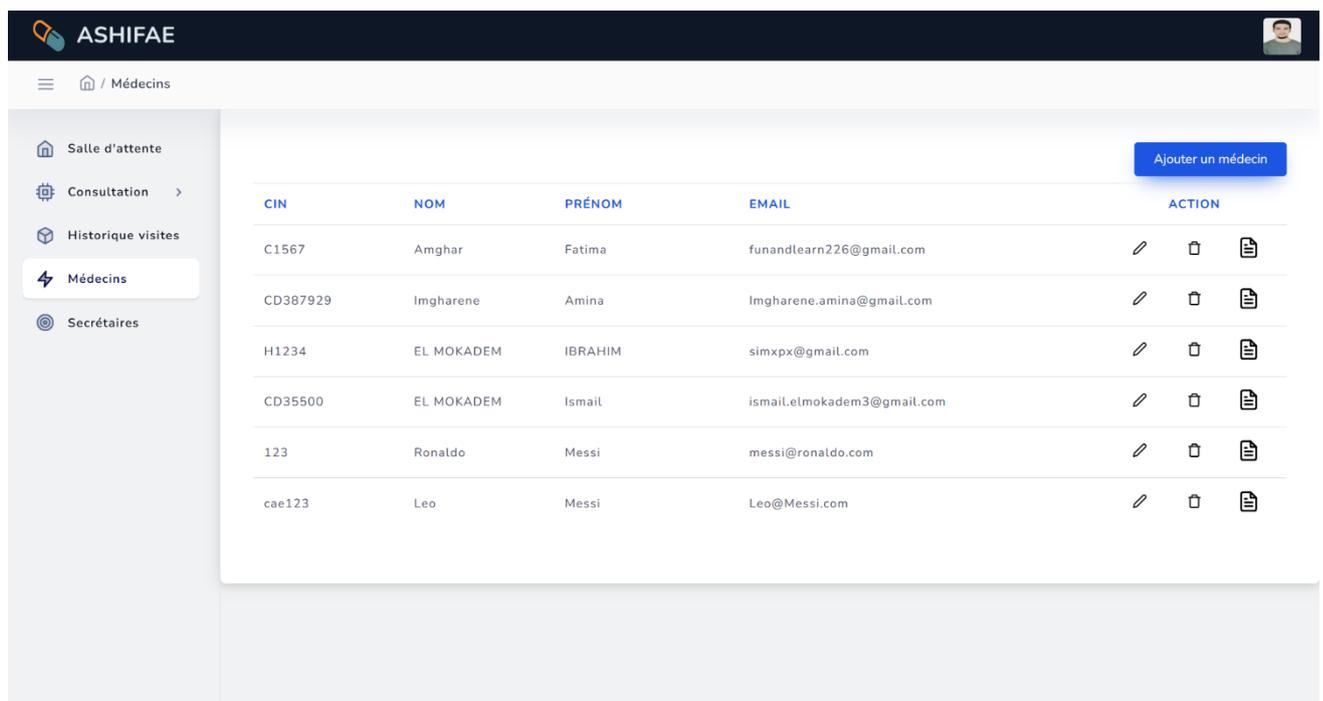


Figure 31 : Menu Médecins

- **Ajouter un médecin**

Dans cette interface un formulaire pour ajouter médecin s'affiche.

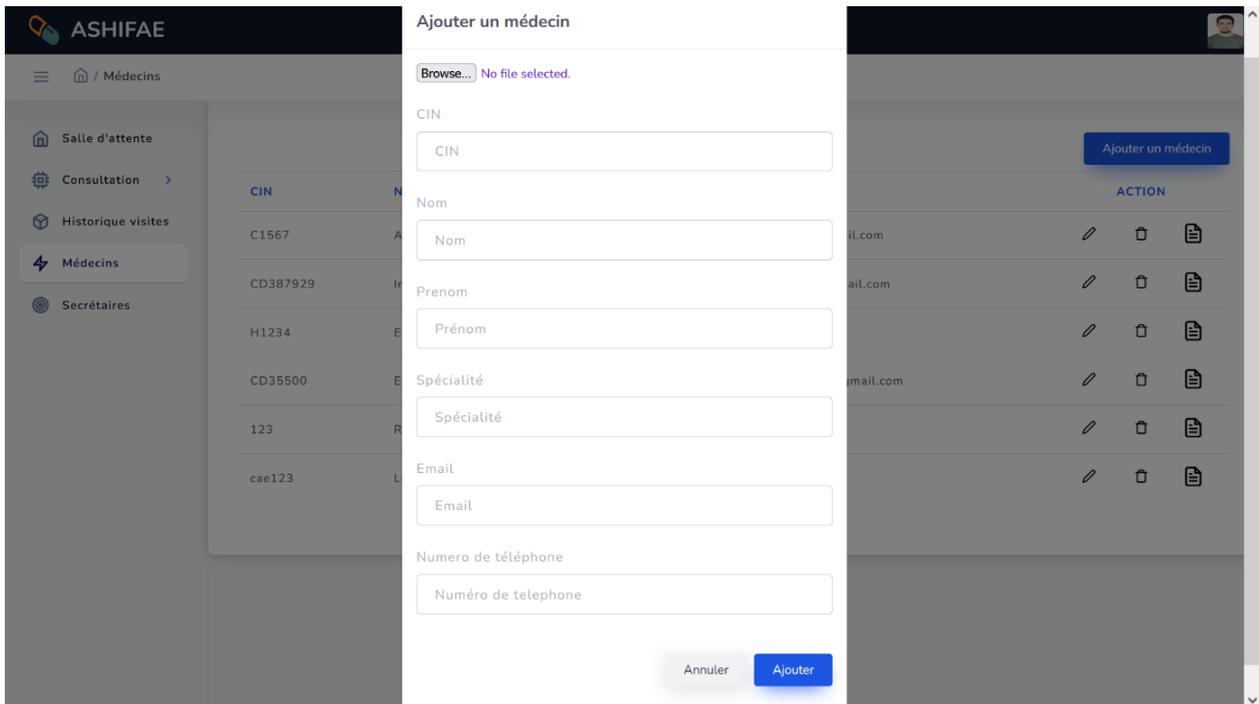


Figure 32 : Ajouter un médecin

b) Menu Secrétaires

Dans ce menu le médecin chef peut soit consulter, ajouter, modifier ou supprimer une secrétaire.

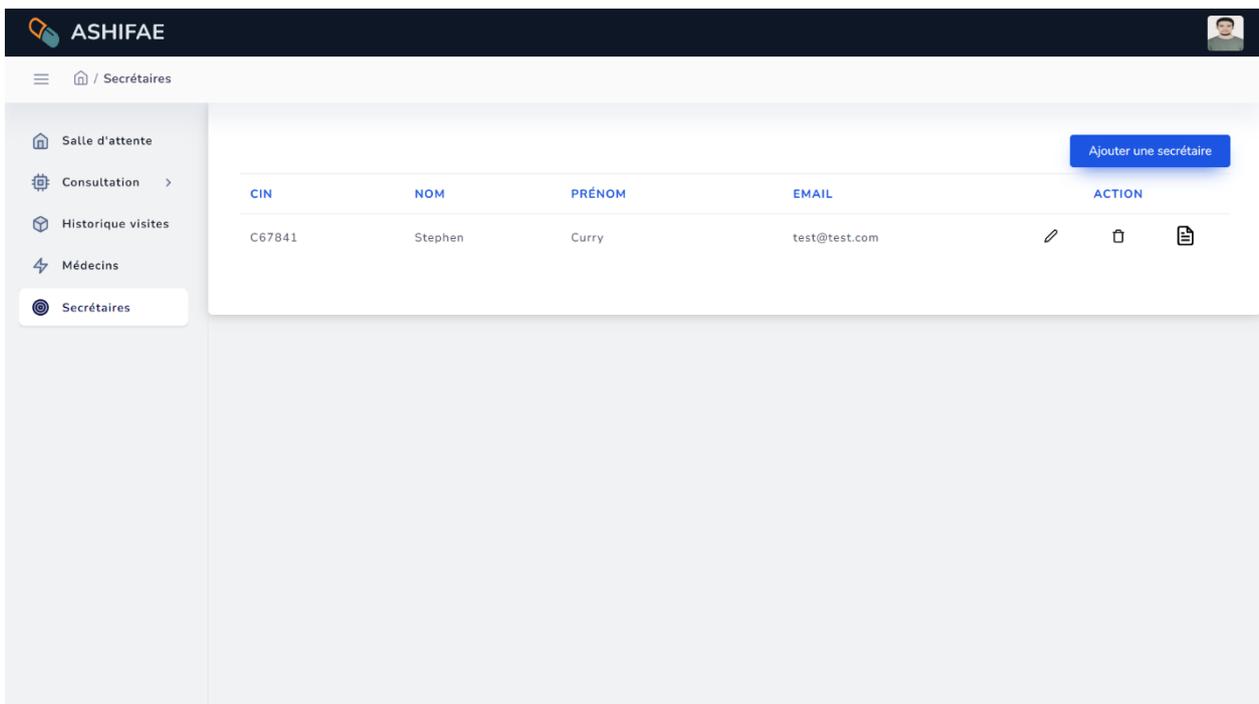


Figure 33 : Menu Secrétaires

- **Ajouter une secrétaire**

Dans cette interface un formulaire pour ajouter secrétaire s'affiche.

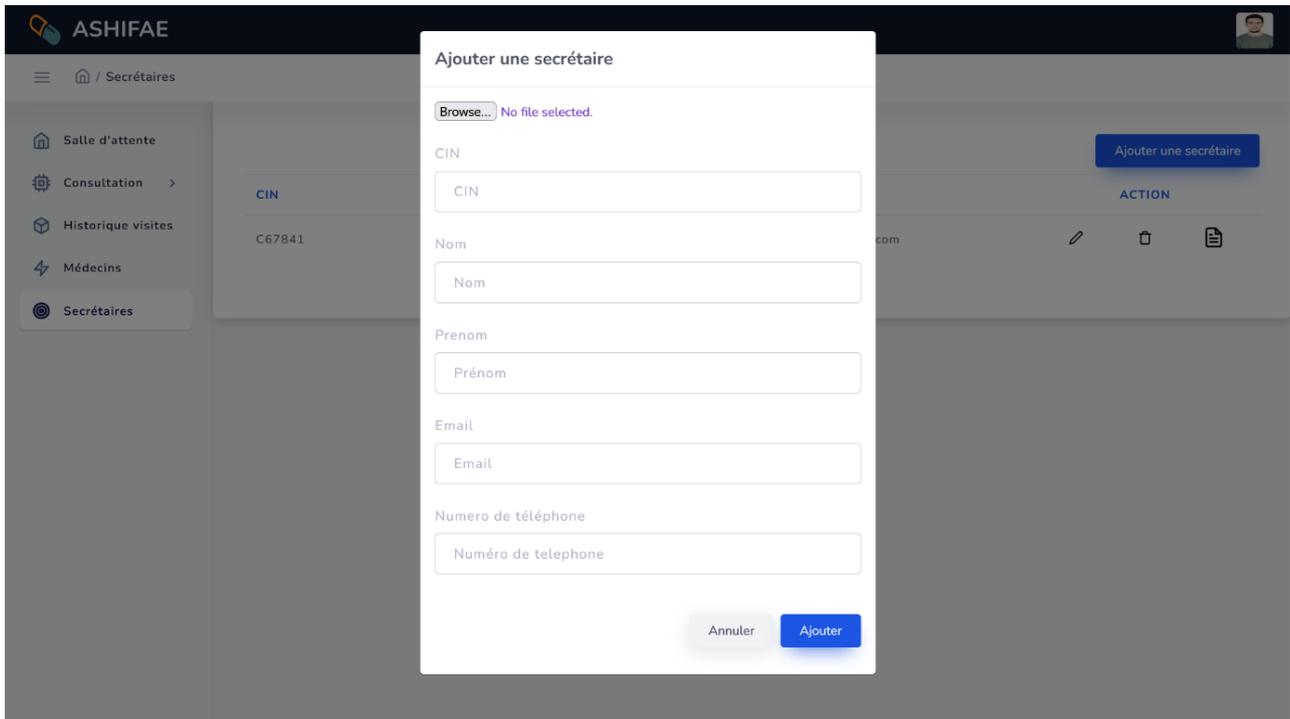


Figure 34 : Ajouter une secrétaire

3.5. Prendre ou annuler un rendez-vous en ligne

Et pour le patient on a une interface dans laquelle il va avoir la possibilité de soit prendre ou annuler un rendez-vous :

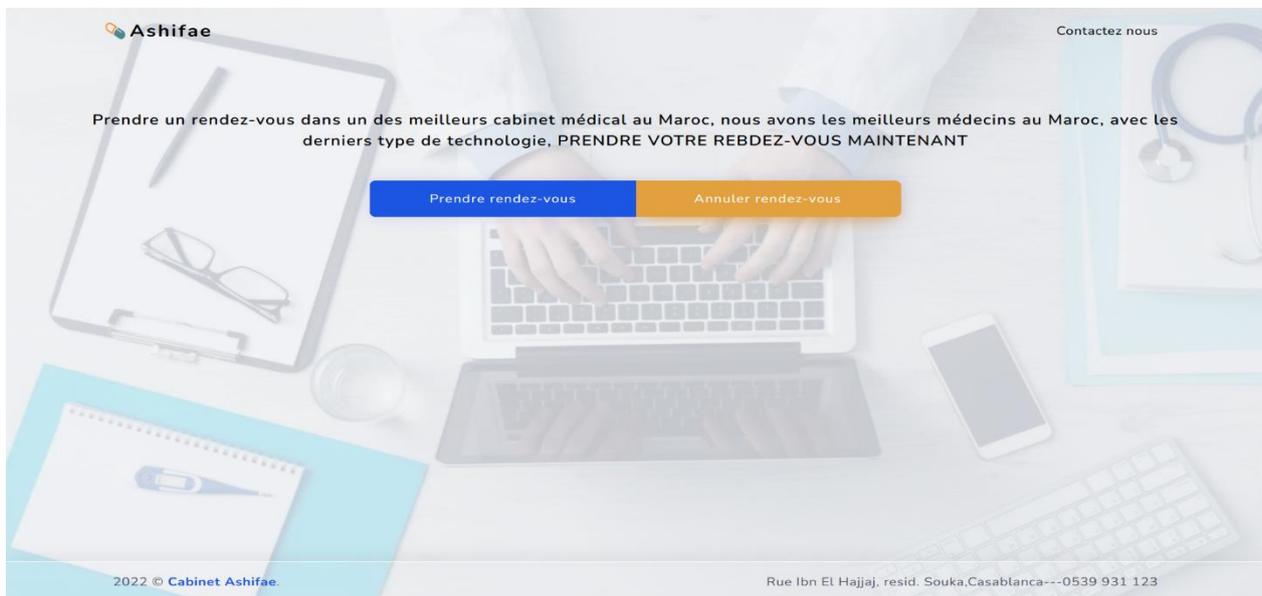


Figure 35 : Prendre ou annuler un rendez-vous en ligne

Conclusion

Ce stage a été pour nous, l'occasion d'approfondir nos connaissances théoriques, acquises durant les 6 semestres de licence, par la pratique des nouvelles technologies.

Au début, nous avons trouvé des difficultés pour bien comprendre le cahier des charges mais après des recherches sur internet, nous avons pu assimiler le cahier des charges et les fonctions que nôtre application doit accomplir.

Pour ce projet nous avons pu développer une application web pour la gestion de cabinet médical qui a comme fonctions d'assurer : la gestion des patients, la gestion des rendez-vous, la gestion des documents, la gestion des factures, la gestion des médecins et secrétaires, la gestion de salle d'attente et la gestion des statistiques.

Vu la taille importante du projet et la limitation du temps, plusieurs améliorations à cette version peuvent être envisagées. En perspective, nous pouvons implémenter d'autres fonctionnalités par exemple : La gestion des médicaments.

Pour conclure, cette expérience en marché de travail nous a offert une bonne préparation à notre insertion professionnelle car elle fut pour nous une expérience enrichissante et complète qui conforte notre désir d'exercer notre futur métier dans le domaine d'informatique.

Références

Bibliographie

- Laravel - Un framework efficace pour développer vos applications PHP (Livre).

Webographie

- <https://stackoverflow.com/questions/26351085/laravel-passing-data-using-ajax-to-controller> (Consulté du 28-05-2022)
- <https://www.w3schools.in/laravel> (Consulté du 25-04-2022)
- <https://laracasts.com/discuss/> (Consulté du 10-05-2022 au 25-06-2022)
- <https://www.codegrepper.com/code-examples/php/frameworks/laravel/> (Consulté du 20-05-2022 au 15-06-2022)
- <https://laravel.io/forum/laravel-email-send-localhost-having-problem> (Consulté le 15-05-2022)
- <https://fr.wikipedia.org> (Consulté le 23-06-2022 pour prendre des images)