**Gestion des dossiers Médicaux D’un Cabinet Dentaire**

#### Rapport de projet J A V A - EE

**Filiére:** Ingénierie Informatique et Réseaux

**Classe : 4iir10**

**Encadré par :** M. El MIDAOUI Omar

**Réalisé par : -** BENZZOUBEIR Ouiam

* HASSANI Doha

Année universitaire: 2023/2024

# Résumé

Cette application de gestion de dossiers médicaux est une solution complète destinée aux cabinets médicaux pour gérer efficacement les informations des patients, les dossiers médicaux, les consultations, les factures, et les paiements. Conçue avec une architecture Spring, elle offre une interface utilisateur intuitive et ergonomique pour les professionnels de santé. Le backend utilise le framework Spring Boot pour assurer la robustesse et la scalabilité, tandis que le frontend est construit avec des technologies modernes pour une expérience utilisateur fluide.

L'application permet aux utilisateurs de :

* **Gérer les patients** : Ajouter, modifier et supprimer les informations des patients.
* **Gérer les dossiers médicaux** : Accéder aux antécédents médicaux, consultations, et diagnostics.
* **Gérer les consultations** : Planifier et suivre les consultations médicales.
* **Gérer les finances** : Créer et suivre les factures et les paiements.

Ce rapport couvre et explique toutes les étapes suivies du début jusqu’à la fin pour mener cette réalisation, que nous vous laissons le plaisir de le découvrir.

Mots clés

Spring Boot, Cabinet médical, conception, Dossiers médicaux.

# Abstract

This medical records management application is designed to provide a comprehensive solution for medical clinics to efficiently manage patient information, medical records, consultations, invoices, and payments. Built using the Spring Boot framework, the backend ensures robustness and scalability, while the frontend leverages modern technologies for a seamless user experience. The application offers an intuitive and ergonomic user interface for healthcare professionals, enabling them to perform various tasks such as adding, modifying, and deleting patient information, accessing and updating medical records, scheduling and tracking consultations, and managing financial operations including invoicing and payments.

Keywords

Spring Boot, Consultations.

# Table des matières

Introduction générale ................................................................................................................ 5

**Chapitre 1 : contexte générale du projet**

1.1. Présentation du projet.........................................................................................6

1.2. Cahier des charges ......................................................................................................6

1.2.1. Présentation du projet .........................................................................................6

1.2.1.1. Objectifs ..................................................................................................6

1.2.2.1. Méthodologie ..........................................................................................7

**Chapitre 2 : Analyse et conception**

2.1. Introduction .................................................................................................................8

2.2. Présentation du langage UML .....................................................................................8

2.3. Identification des acteurs ............................................................................................8

2.4. Diagrammes de cas d’utilisation .................................................................................9

2.4.1. Définition ...........................................................................................................9

2.4.2. Diagramme de cas d’utilisation global ...............................................................9

2.5. Diagramme des classes ...............................................................................................10

2.5.1. Définition ..........................................................................................................10

2.6. Conclusion ..................................................................................................................11

**Chapitre 3 : Outils et environnement**

3.1. Introduction ...............................................................................................................12

3.2. Outils et technologie utilisés ......................................................................................13

3.3. Conclusion ..................................................................................................................15

**Chapitre 4 : Mise en œuvre du projet**

4.1. Introduction ...............................................................................................................17

4.2. Problèmes rencontrés au cours de la réalisation du projet .......................................17

4.3. Solutions ....................................................................................................................17

Les principales interfaces de l’application .......................................................................18

Conclusion ...................................................................................................................23

**Chapitre 5 : Conclusion**

Conclusion générale ............................................................................................................24

# Introduction générale

La gestion efficace des dossiers médicaux est essentielle pour le bon fonctionnement des cabinets médicaux. Avec l'évolution rapide des technologies et l'augmentation des attentes en matière de soins de santé, les professionnels de santé doivent adopter des solutions numériques pour optimiser leurs opérations quotidiennes. Les méthodes traditionnelles de gestion documentaire, basées sur le papier, deviennent de plus en plus obsolètes face à la complexité croissante et au volume des données à gérer.

Ce projet a été une occasion précieuse de mettre en pratique nos compétences en développement informatique tout en répondant à une problématique concrète de gestion d’un cabinet. Le présent rapport détaillera ainsi les étapes du développement, les technologies utilisées, les résultats obtenus et les perspectives d'évolution de cette application de gestion des dossiers médicaux.

Les différentes phases proposées lors du traitement de notre sujet sont les suivantes :

* Présentation du projet
* Analyse et conception
* Technologies et outils utilisés
* Développement et implémentation
* Tests et validation
* Conclusion et perspectives d'évolution

**Chapitre 1 :**

## Contexte général du projet



#### 1.1 Problématique du projet

#### La gestion des dossiers médicaux dans un cabinet de santé est une tâche complexe et critique qui exige précision, rapidité et confidentialité. Traditionnellement, les cabinets médicaux ont utilisé des méthodes papier pour gérer les dossiers des patients, ce qui s'avère de plus en plus inadéquat face à l'augmentation du volume d'informations et aux exigences croissantes de rapidité et d'efficacité dans les soins de santé.

#### 1.2 Planification du projet

La planification a pour objectif d’organiser le déroulement des étapes du projet dans le temps. Une tâche fondamentale pour la maîtrise des délais.

Planifier un projet consiste à découper en plusieurs étapes (identification de l’ensemble des tâches à réaliser), d’en estimer la durée, d’identifier l’enchaînement des étapes (dont celles qui peuvent être conduites en parallèle ordonnancement des tâches, chemin critique), affecter des ressources (humaines et financières) et enfin modéliser cette organisation sur un document opérationnel partagé entre tous les acteurs concernés pour optimiser le déroulement et le suivi de la réalisation.

#### 1.3Cahier des charges

#### Objectifs

1. **Gestion des Patients** : Ajouter, modifier, et supprimer les informations des patients.
2. **Gestion des Dossiers Médicaux** : Accéder et mettre à jour les antécédents médicaux et les consultations des patients.
3. **Planification des Consultations** : Planifier, modifier et suivre les consultations des patients.
4. **Gestion des Finances** : Créer et suivre les factures et les paiements des patients.
5. **Interface Utilisateur Intuitive** : Fournir une interface facile à utiliser pour les professionnels de santé.

#### Fonctionnalités

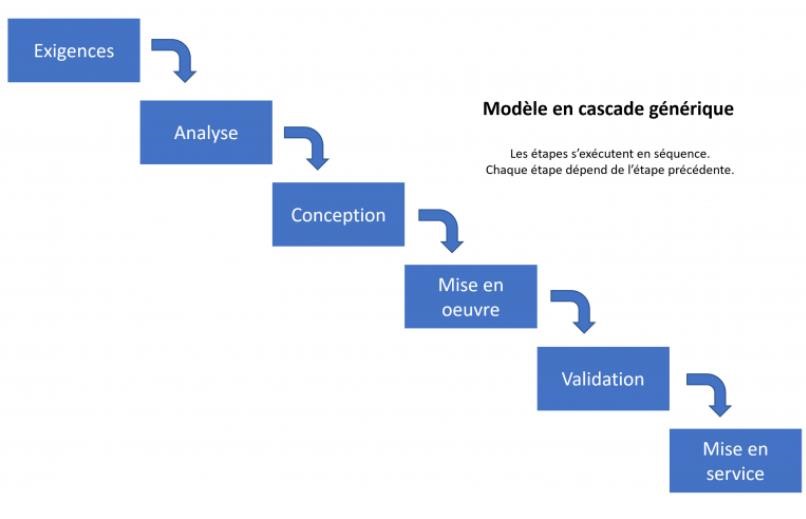
* **Authentification et Autorisation** : Gestion des rôles et des permissions des utilisateurs.
* **Tableau de Bord** : Vue d'ensemble des activités et des statistiques clés.
* **Gestion des Patients** : Formulaires pour ajouter et modifier les informations des patients.
* **Dossiers Médicaux** : Accès détaillé aux antécédents médicaux, consultations, et diagnostics.
* **Consultations** : Planification et suivi des consultations avec des rappels et notifications.
* **Facturation et Paiements** : Création, modification, et suivi des factures et paiements.

1.4 Méthodologie :

La méthode qu’on a adopté pour la réalisation de notre projet est la méthode cascade.

Le modèle en cascade, comme son nom l’indique, suit la logique d’une chute d’eau. Une fois que l’eau a dévalé le flanc de la montagne, elle ne peut plus remonter, mais seulement continuer son chemin. Ainsi, **dès qu’une étape du projet est terminée, l’équipe passe à l’étape suivante** ; il n’y a pas (ou peu) de retour en arrière. L’idée est d’avancer naturellement, étape par étape, jusqu’à atteindre l’objectif final en suivant une direction claire et précise. De cette façon, on **limite le risque d’erreur** et l’incertitude.

En appliquant cette méthode, je suis le cahier des charges à la lettre et travaille sur la totalité du projet jusqu’à sa livraison. Il n’y a pas d’interaction avec le client qui recevra son projet une fois que celui-ci est terminé.



*Figure 1 : modèle en cascade générique*

**Chapitre 2 :**

## Analyse et conception



### 2.1 Introduction

A travers ce chapitre, nous entamons l’analyse et la conception qui présente une étape fondamentale qui précède la réalisation. Il existe plusieurs méthodes de modélisation, la plus répandue étant le langage UML, qui a été choisi.

### 2.2 Présentation du langage UML

Le langage UML (Unified Modeling Language, ou langage de modélisation unifié) a été pensé pour être un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement.

Il est destiné à l’architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leurs structures aussi bien que leur comportement. L’UML a des applications qui vont au-delà du développement logiciel, notamment pour les flux de processus dans l’industrie.

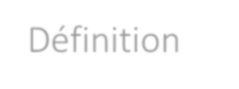
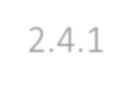
Nous avons utilisé plusieurs diagrammes afin de représenter notre système d’information, notamment les diagrammes de cas d’utilisation, de séquence et des classes.

### 2.3 Identification des acteurs

Un projet informatique ne se réalise pas seul et nécessite la contribution de plusieurs acteurs effectuant des rôles bien précis.

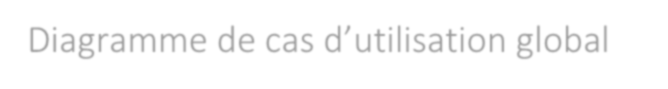
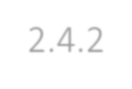
2.4 Diagrammes de cas d’utilisation

#### 2.4.1 Définition

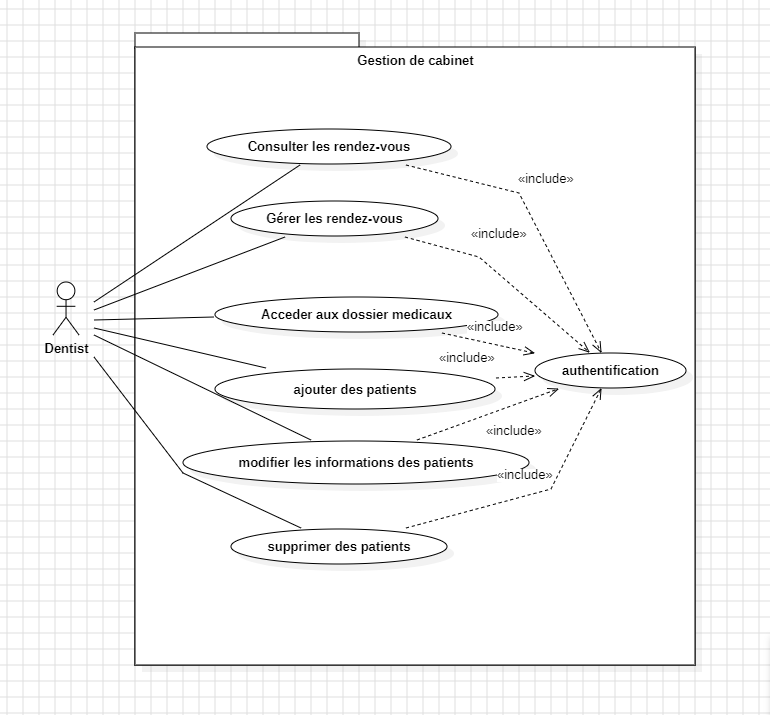


Un cas d’utilisation est la description d’un ensemble de séquence d’actions qu’un système effectue pour produire un résultat observable par un acteur. Un cas d’utilisation représente une exigence fonctionnelle de votre système dans son ensemble. Les diagrammes de cas d’utilisation décrivent ce qu’un système fait du point de vue d'un observateur externe. L'accent est mis sur ce qu'un système fait, plutôt que sur la façon dont il le fait.

#### 2.4.2 Diagramme de cas d’utilisation global



Ce diagramme illustre globalement les différentes exigences fonctionnelles de notre système.

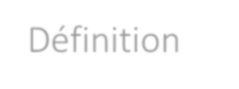
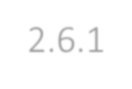


*Figure 2 : Diagramme de cas d’utilisation*

Dans ce diagramme de cas d'utilisation, nous avons l'acteur "Utilisateur" qui interagit avec l'application. L'utilisateur peut se connecter en utilisant le formulaire de login ("Login"). Une fois authentifié, l'utilisateur accède à la page d'accueil ("Accueil") où il peut gérer les documents.

### 2.5 Diagramme des classes

#### 2.5.1 Définition

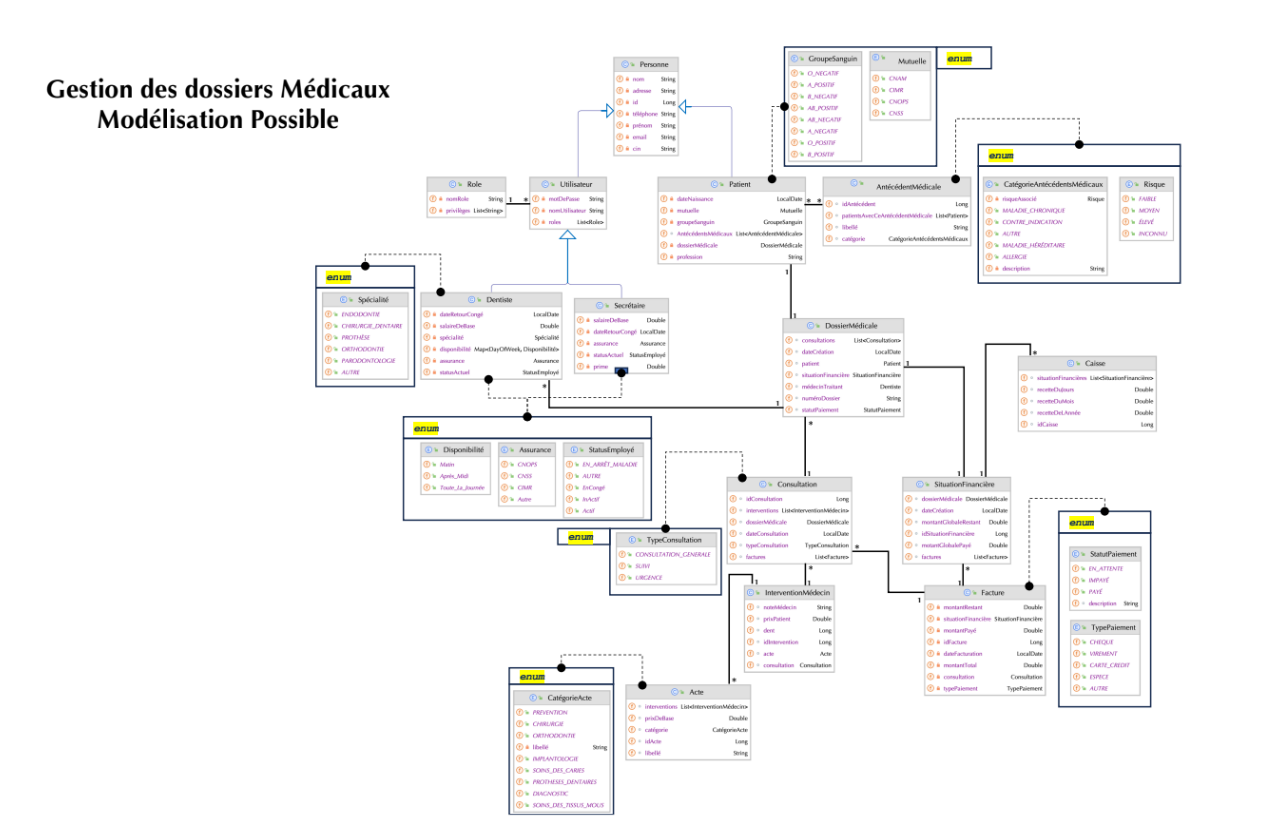


Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation.

Alors que le diagramme de cas d'utilisation montre un système du point de vue des acteurs, le diagramme de classes en montre la structure interne. Il permet de fournir une représentation abstraite des objets du système qui vont interagir pour réaliser les cas d'utilisation. Il est important de noter qu'un même objet peut très bien intervenir dans la réalisation de plusieurs cas d'utilisation. Les cas d'utilisation ne réalisent donc pas une partition des classes du diagramme de classes. Un diagramme de classes n'est donc pas adapté (sauf cas particulier) pour détailler, décomposer, ou illustrer la réalisation d'un cas d'utilisation particulier.

Il s'agit d'une vue statique, car on ne tient pas compte du facteur temporel dans le comportement du système. Le diagramme de classes modélise les concepts du domaine d'application ainsi que les concepts internes créés de toutes pièces dans le cadre de l'implémentation d'une application. Chaque langage de programmation orienté objet donne un moyen spécifique d'implémenter le paradigme objet (pointeurs ou pas, héritage multiple ou pas, etc.), mais le diagramme de classes permet de modéliser les classes du système et leurs relations indépendamment d'un langage de programmation particulier.

Les principaux éléments de cette vue statique sont les classes et leurs relations : association, généralisation et plusieurs types de dépendances, telles que la réalisation et l'utilisation.



*Figure 3 : Diagramme de classe*

### 2.5 Conclusion

Ce chapitre a été consacré à la partie analyse, conception et modélisation du projet, composée de plusieurs diagrammes UML et qui est nécessaire afin de garantir une conduite cohérente du projet.

Afin de satisfaire le client d’un côté et faciliter les tâches de l’administrateur, nous avons choisi de structurer les composants de site web de la manière mentionnée ci-dessus pour garantir la fiabilité et de donner le client la meilleure expérience durant sa navigation dans notre site web.

**Chapitre 3 :**

## Outils et environnement



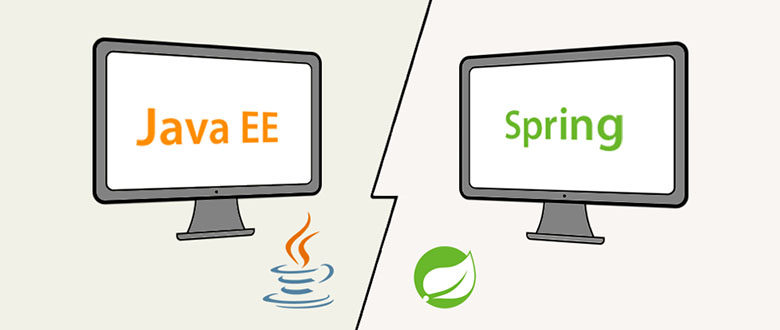
### 3.1 Introduction

Après avoir suivi la méthode et la structure choisie, en passant par les outils et nouvelles technologies, nous avons pu réaliser un site web complet : front-end et back-end. Le chapitre présent a pour but de présenter et de décrire les différents outils et technologies utilisés durant le développement.

### 3.2 Outils et technologies utilisés

Dans cette section, nous vous présentons l’environnement logiciel composé par un ensemble des logiciels et programmes utilisés lors de la création du site web.

### Langages de programmation utilisés



*Figure 9 : logo java ee et spring*



*Figure 10 : logo de html*

Un [langage de balisage](https://fr.wikipedia.org/wiki/Langage_de_balisage) conçu pour représenter les [pages web.](https://fr.wikipedia.org/wiki/Page_web) C’est un langage permettant d’écrire de l’[hypertexte,](https://fr.wikipedia.org/wiki/Hypertexte) d’où son nom. HTML permet également de structurer sémantiquement et logiquement et de mettre en forme le contenu des pages, d’inclure des [ressources](https://fr.wikipedia.org/wiki/Ressource_du_World_Wide_Web) [multimédias](https://fr.wikipedia.org/wiki/Multim%C3%A9dia) dont des [images,](https://fr.wikipedia.org/wiki/Image_num%C3%A9rique) des formulaires de saisie et des programmes informatiques.



*Figure 11 : logo de CSS*

Un [langage informatique](https://fr.wikipedia.org/wiki/Langage_informatique) qui décrit la présentation des documents [HTML](https://fr.wikipedia.org/wiki/Hypertext_markup_language) et [XML.](https://fr.wikipedia.org/wiki/Extensible_markup_language)

Les [standards](https://fr.wikipedia.org/wiki/Standard_technique) définissant CSS sont publiés par le [World Wide Web Consortium](https://fr.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C) Le Framework utilisés



*Figure 12 : logo de Bootstrap*

**Bootstrap** est une [collection d'outils](https://fr.wikipedia.org/wiki/Framework) utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de [sites](https://fr.wikipedia.org/wiki/Site_web) et d['applications web.](https://fr.wikipedia.org/wiki/Application_web) C'est un ensemble qui contient des codes [HTML](https://fr.wikipedia.org/wiki/HTML) et [CSS,](https://fr.wikipedia.org/wiki/Cascading_Style_Sheet) des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions [JavaScript](https://fr.wikipedia.org/wiki/JavaScript) en option.

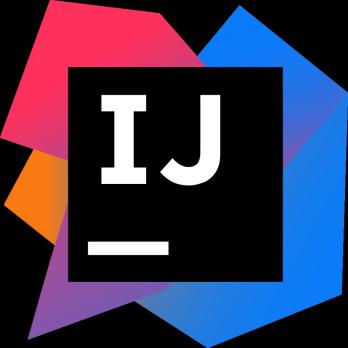
#### Navigateurs de test



*Figure 13 : logo de google chrome*

Un navigateur web gratuit développé par Google. Il a été publié pour la première fois en septembre 2008, pour Microsoft Windows.

#### L’environnement de développement



*Figure 14 : logo de intellij*

* **Spring Boot :** Le framework principal pour développer l'application.



* **Spring Data JPA :** simplifie l'interaction avec la base de données en fournissant une abstraction de haut niveau sur JPA. Il permet de réaliser des opérations CRUD sur les entités sans écrire de code SQL, facilitant ainsi la gestion des données des dossiers médicaux.



* **Spring MVC :** est un module de Spring utilisé pour construire les couches de présentation et de contrôleur de l'application. Il gère les requêtes HTTP, les mappe aux méthodes de contrôleur appropriées et détermine les vues à retourner, facilitant ainsi la création de pages web dynamiques et interactives.



* **Thymeleaf :** est un moteur de template utilisé pour générer des vues HTML. Il permet de créer des interfaces utilisateur dynamiques et interactives, en s'intégrant parfaitement avec Spring MVC. Les templates Thymeleaf sont facilement maintenables et lisibles, ce qui facilite le développement front-end de l'application.



* **Maven :** Maven est un outil de gestion de projet et de construction utilisé pour gérer les dépendances et automatiser le processus de build de l'application. Il facilite l'intégration des bibliothèques tierces et assure la cohérence du projet en gérant les versions des dépendances.



* **Hibernate Validator :** Hibernate Validator est utilisé pour valider les données des formulaires. Il permet de définir des règles de validation directement sur les entités, assurant que les données saisies par les utilisateurs respectent les contraintes définies



**Chapitre 4 :**

## Mise en Œuvre Du projet



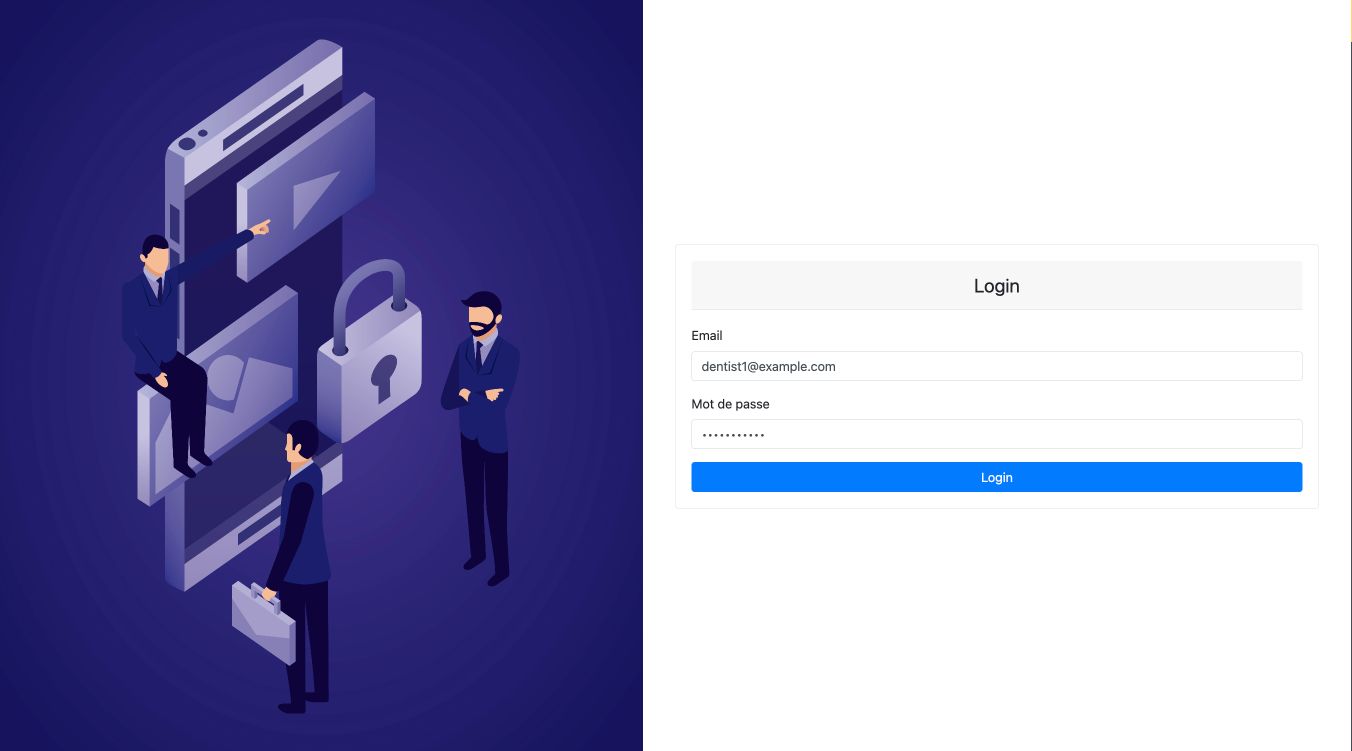
### 4.1. Introduction

Ce chapitre est considéré comme le fruit de tout le travail mené durant notre année universitaire. Afin de mettre en évidence le résultat obtenu, nous présenterons un aperçu sur les différentes solutions de notre projet.

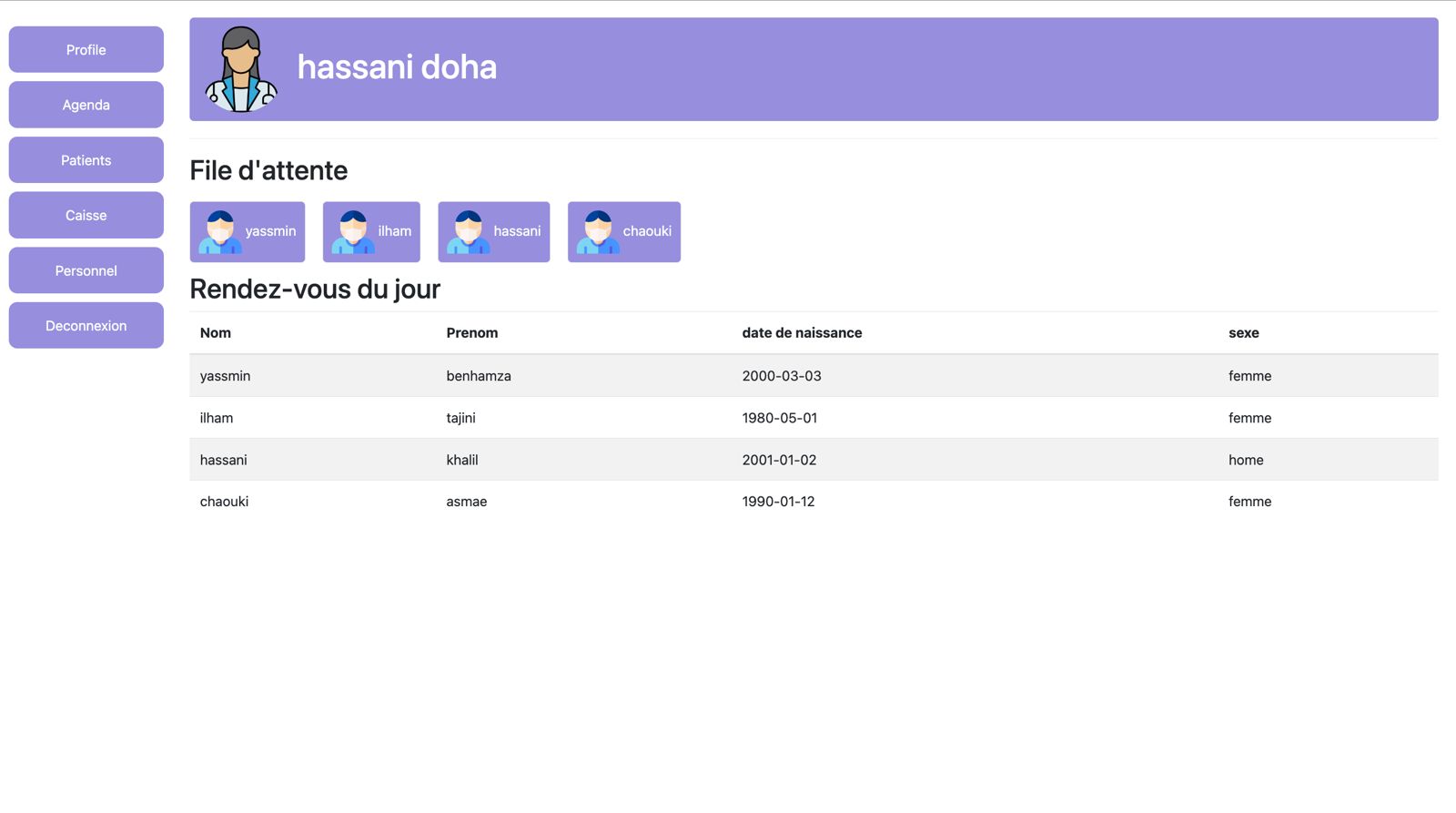
### 4.2. Problèmes rencontrés au cours de la réalisation du projet

➢ Gestion des erreurs et des exceptions ➢ Gestion des versions.

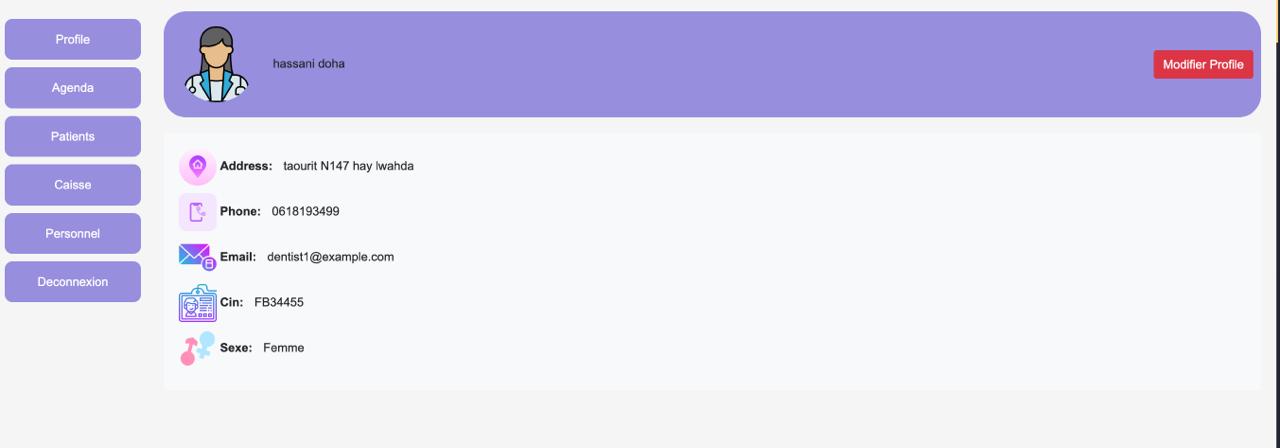
### 4.3. Les principales interfaces du site web



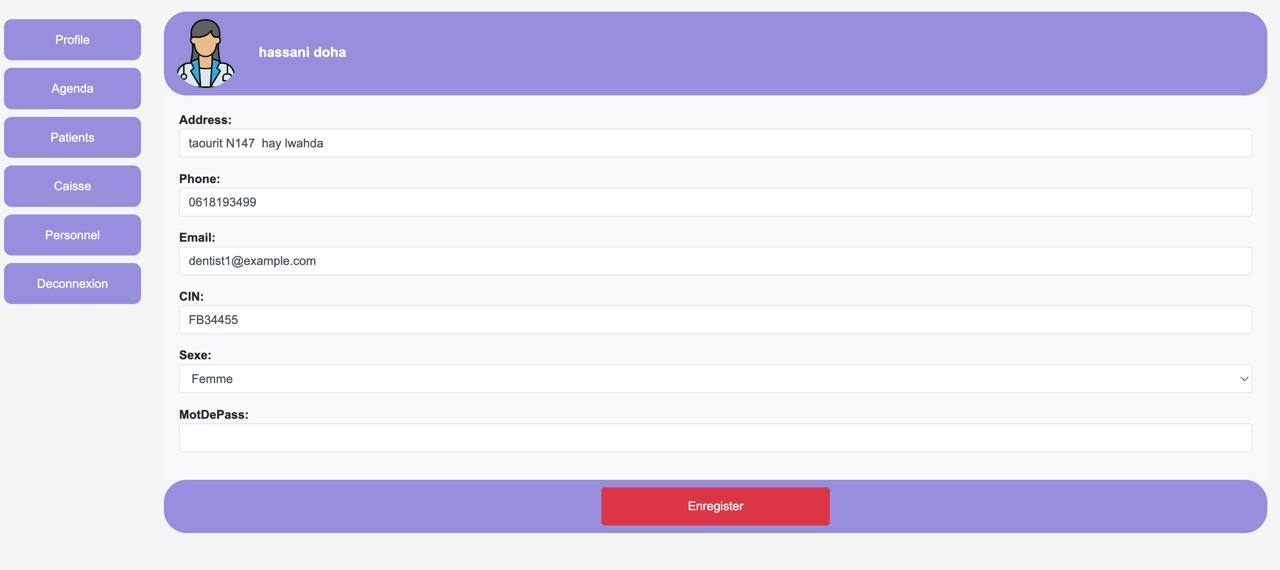
C’est la page du login de l’application .



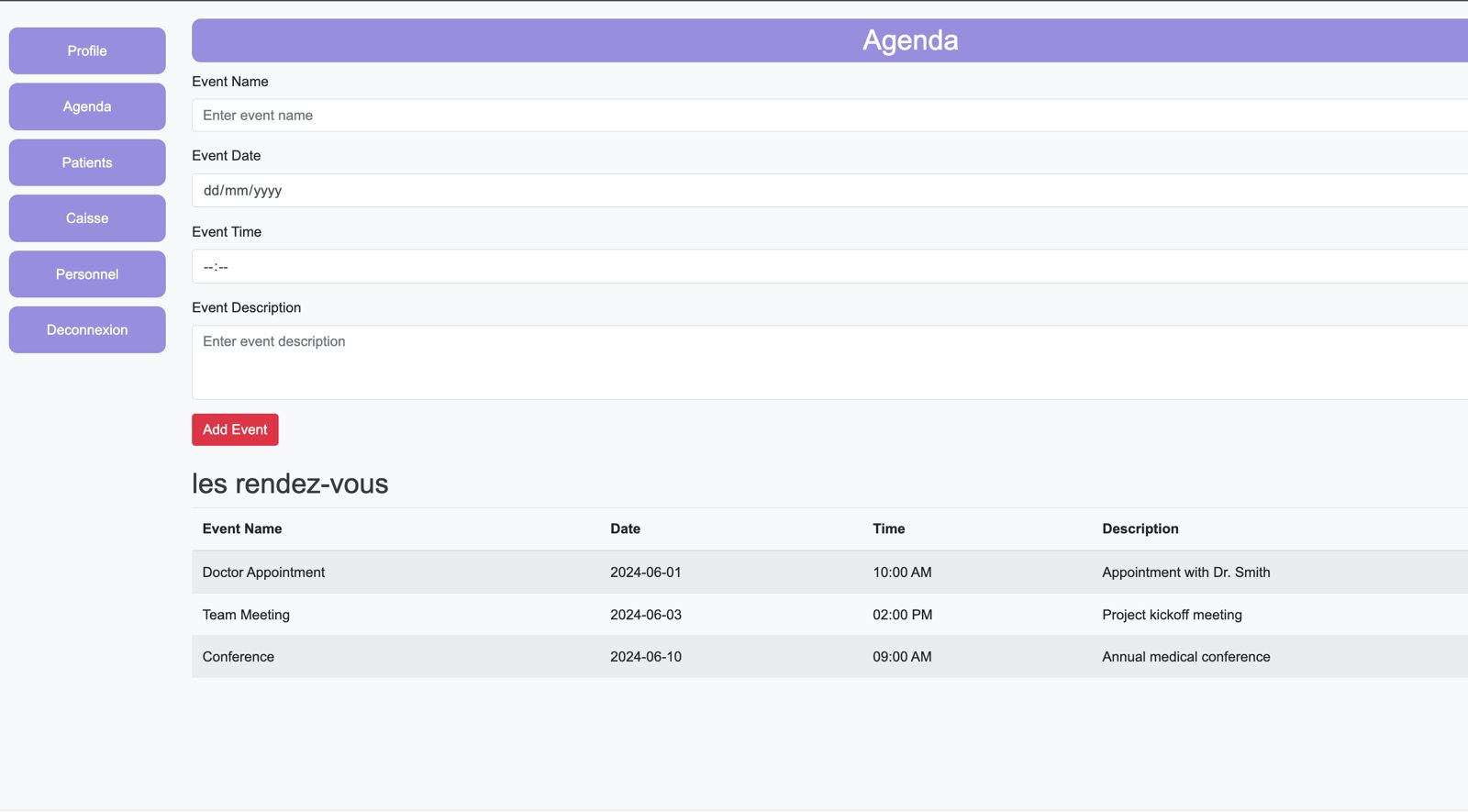
C’est la première page après la connexion il comporte le non du medecin puis la file d’attente et les rendez-vous du jour .



C’est l’interface profile qui affiche les informations du médecin .



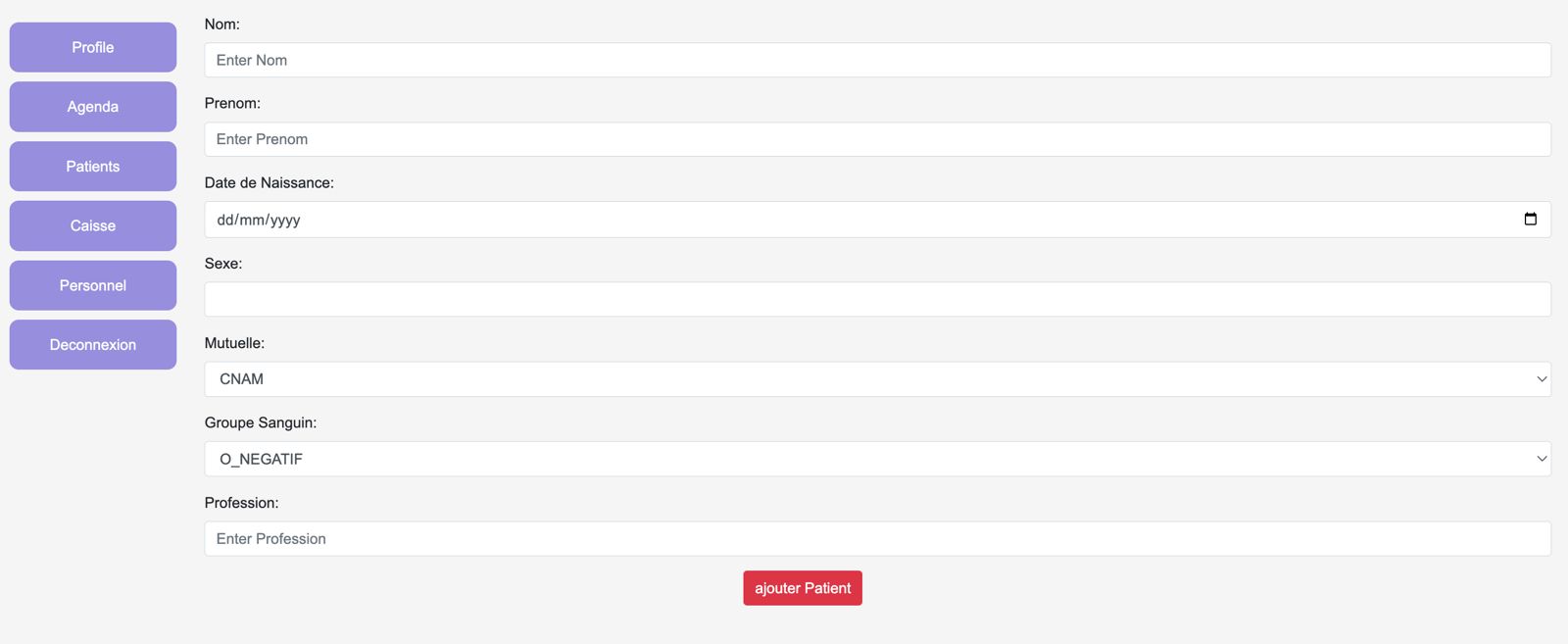
L’interface de modification du profile de médecin .



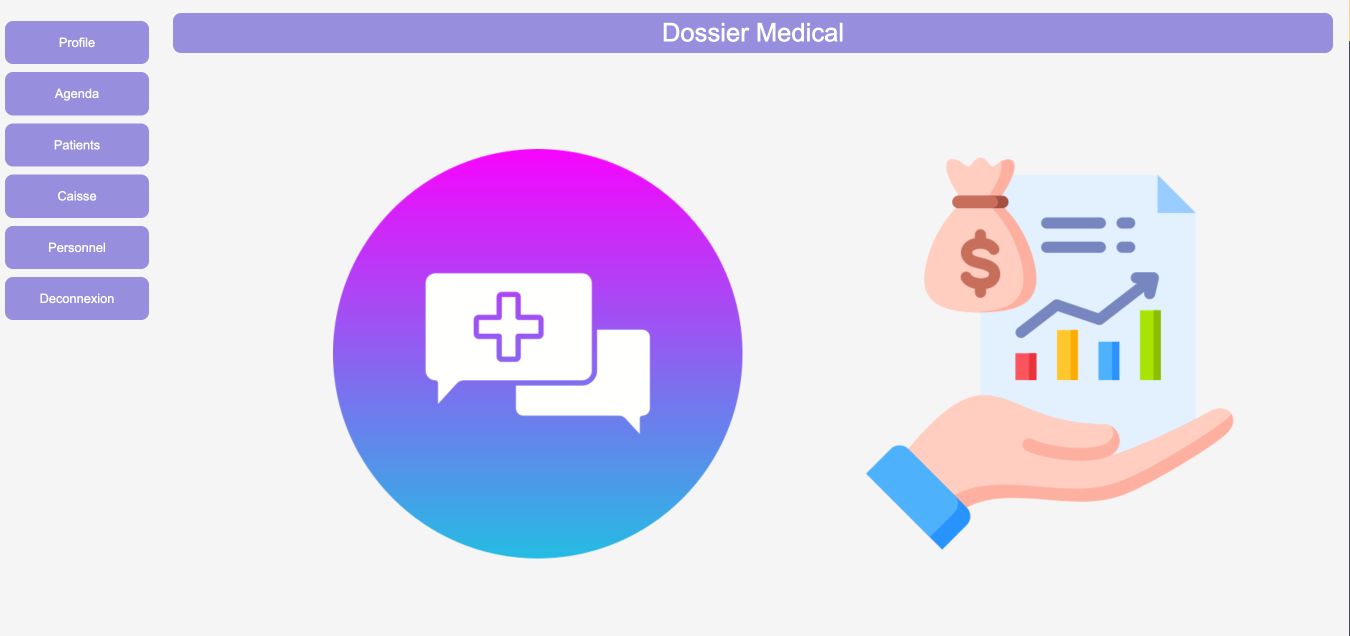
Cette interface agenda affiche les rendez vous et permet d’ajouter un rendez-vous.



Cette interface affiche la liste qui existe avec deux boutons de suppressions et de dossier médicales .



Cette interface d’ajout des patients .

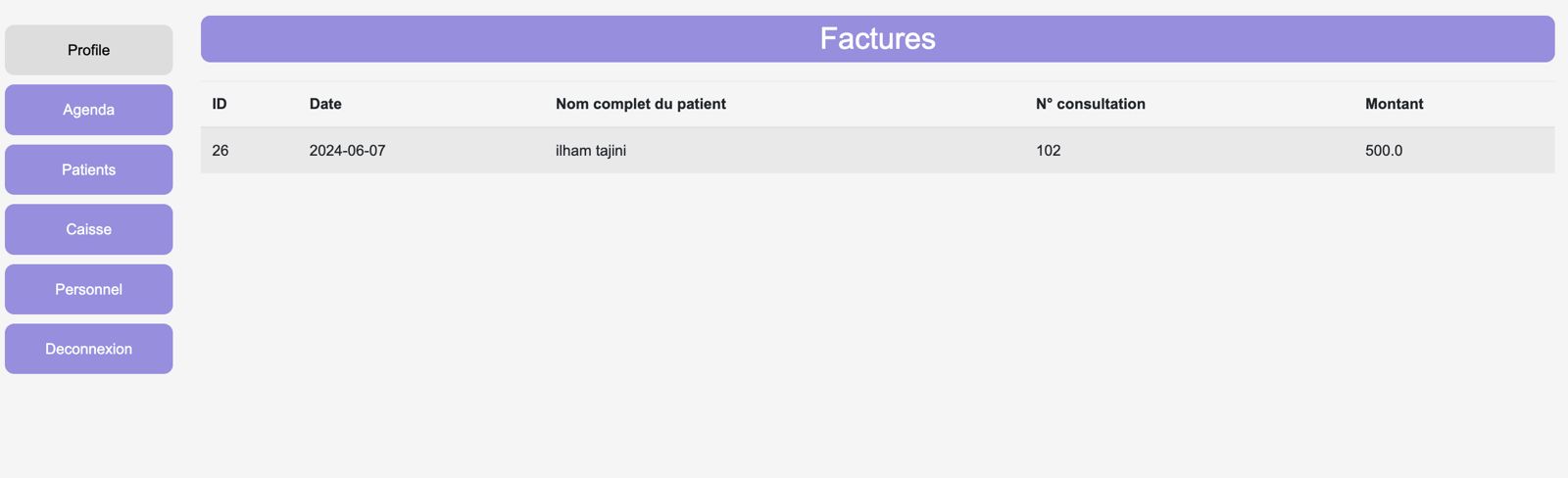


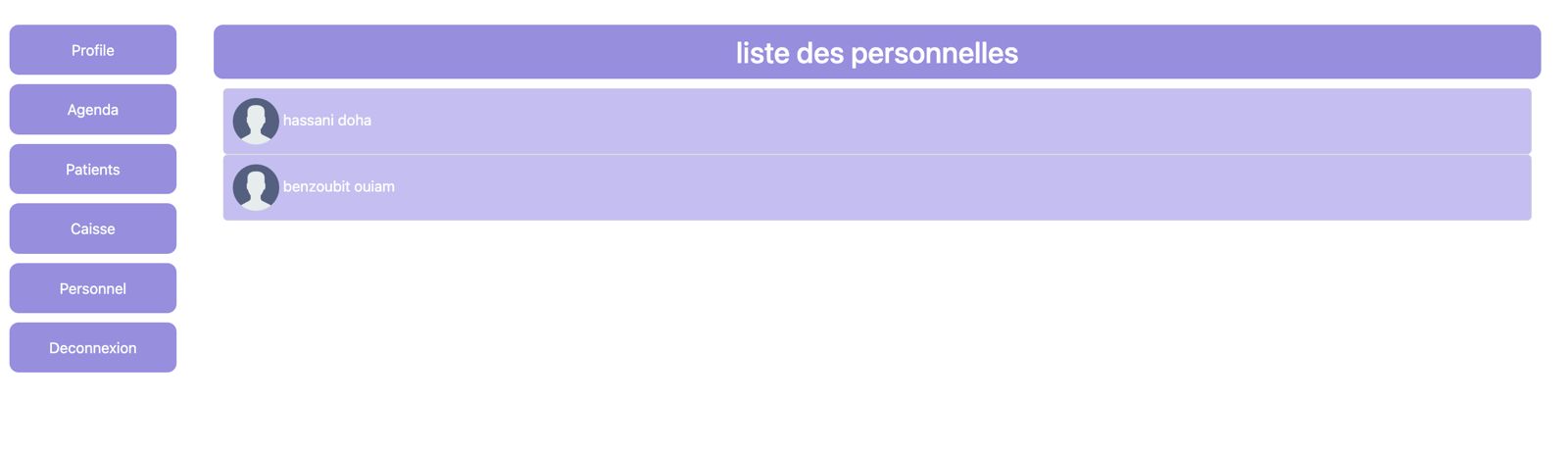


Cette interface affiche les consultations et donne un formulaire d’ajout des consultations .



Cette interface affiche les factures et donne un formulaire d’ajout des factures .





#### 4.4.4.5. Conclusion

Dans cette partie nous avons essayé d’exposer toutes les fonctionnalités de l’application ainsi que les plans futurs programmés pour ce dernier afin de satisfaire la majorité des besoins de la population cible dans le domaine.

**Chapitre 5 :**

## Conclusion



En conclusion, ce stage m'a offert une expérience enrichissante et formatrice dans le domaine du développement d'applications web. J'ai eu l'opportunité de mettre en pratique mes connaissances en programmation, conception de bases de données et gestion de projet.

L'application de gestion des archives que j'ai développées constitue un outil efficace et fonctionnel pour faciliter la gestion des documents au sein de l'entreprise. Elle permet de stocker, organiser et rechercher facilement les documents, ce qui améliore la productivité et la gestion des informations.

Au cours de ce stage, j'ai été confronté à divers défis techniques, notamment la gestion des erreurs, l'optimisation des performances et la sécurité de l'application. J'ai pu les surmonter en utilisant des bonnes pratiques de développement et en faisant preuve de rigueur dans la gestion du code.

En somme, ce stage a été une expérience positive et stimulante qui m'a permis de consolider mes compétences techniques, de découvrir de nouveaux aspects du développement web et de me préparer à une carrière professionnelle dans ce domaine. Je suis reconnaissant envers l'entreprise qui m'a accueilli et envers mon tuteur de stage pour son encadrement et son soutien tout au long de cette expérience. Je suis désormais prêt à relever de nouveaux défis et à contribuer au développement d'applications web innovantes.