

**Informatique Génie Logiciel**

**Développement d’une application full Stack de gestion des rendez-vous médicaux en ligne**

## Abdeljelile Cyrine

### 10-06-2024

Walid Chafroud

Hatem Sandid

Hatem Jarboua

# Dédicaces

Je tiens à exprimer ma profonde gratitude et mon affection à toutes les personnes qui ont contribué à la réalisation de ce projet.

À mes parents, pour leur soutien inconditionnel et leurs encouragements constants, merci de m'avoir toujours montré le chemin de la persévérance et de l'excellence.

À mon petit enfant, dont la joie de vivre et l'innocence m'ont apporté la motivation et le courage pour être une meilleure personne chaque jour.

À toute ma famille, pour leur amour et leur soutien indéfectibles. Vous avez tous contribué à mon succès de manière significative.

À mes amies, pour leur soutien, leurs encouragements et leur amitié précieuse. Merci d'avoir toujours cru en moi.

# Cyrine

Remerciements

À l'issue de ce travail, je tiens à exprimer ma gratitude :

Avant tout, je remercie Dieu pour m'avoir donné le courage, la force et la volonté nécessaires à la réalisation de ce projet.

Je suis profondément reconnaissant à mon encadrant, Monsieur Hatem Jarboua, enseignant à l'École Supérieure Privée Polytechnique de Monastir, pour son soutien et ses commentaires constructifs qui ont grandement enrichi ce travail.

Je souhaite également remercier les membres du jury pour l'honneur qu'ils m'ont fait en prenant le temps de lire et d'évaluer ce travail. Votre attention et votre expertise sont très appréciées.

Ma gratitude va également aux équipes pédagogiques et administratives de Polytech pour leur dévouement et leurs efforts constants afin de nous offrir une formation de qualité.

Enfin, je remercie chaleureusement toutes les personnes qui, de près ou de loin, ont contribué à la réalisation de ce projet.

Contents

[Introduction générale 1](#_bookmark0)

[Chapitre 1 3](#_bookmark1)

[Cadre général du projet 3](#_bookmark2)

[Introduction 3](#_bookmark3)

[Cadre de Projet 3](#_bookmark4)

[Présentation du projet 3](#_bookmark5)

[Contexte et problématique du projet 3](#_bookmark6)

[Etude de l’existant 4](#_bookmark7)

[Analyse de l’existant 4](#_bookmark8)

[Solution proposée 5](#_bookmark12)

[Conclusion 5](#_bookmark13)

[Chapitre 2 6](#_bookmark14)

[Spécification, analyse des besoins et conception 6](#_bookmark15)

[Introduction 6](#_bookmark16)

[Analyse des besoins de mon projet 6](#_bookmark17)

[Identification des acteurs 6](#_bookmark18)

[Besoins Fonctionnels 7](#_bookmark20)

[Besoins non Fonctionnels 7](#_bookmark21)

[Choix du langage de modélisation UML 8](#_bookmark22)

[Présentation d’UML 8](#_bookmark23)

[Caractéristiques 8](#_bookmark24)

[Diagramme de cas d’utilisation 8](#_bookmark25)

[Diagramme des cas d’utilisation global 9](#_bookmark26)

[Diagramme des cas d’utilisation étudiant 9](#_bookmark28)

[Diagramme du cas d’utilisation « gestion cours » 9](#_bookmark29)

[Diagramme détaillé du cas d’utilisation « gestion QCM étudiant » 11](#_bookmark32)

[Diagramme de cas d’utilisation enseignant 12](#_bookmark35)

[Diagramme détaillé du cas d’utilisation « gestion cours » 12](#_bookmark36)

[Diagramme détaillé du cas d’utilisation « gestion QCM enseignant » 13](#_bookmark39)

[Diagramme des cas d’utilisation admin 14](#_bookmark42)

[Diagramme du cas d’utilisation « gestion utilisateurs » 14](#_bookmark43)

[Diagramme détaillé du cas d’utilisation « Gestion cours » 16](#_bookmark46)

[Diagramme détaillé du cas d’utilisation « Gestion catégories » 17](#_bookmark49)

[Diagramme détaillé du cas d’utilisation « Gestion résultats test QCM » 18](#_bookmark52)

[Diagramme des séquences 19](#_bookmark55)

[Inscription et Login 20](#_bookmark56)

[Conclusion 20](#_bookmark57)

[Chapitre 3 Réalisation 21](#_bookmark59)

[Introduction 21](#_bookmark60)

[Choix Technologiques 21](#_bookmark61)

[Langage de développement 21](#_bookmark62)

[Environnement logiciel 23](#_bookmark69)

[Présentation et description de l’application « plateforme e-learning» 25](#_bookmark71)

[Interface de lancement de l’application 25](#_bookmark72)

[Interfaces d’authentification et d’inscription 25](#_bookmark74)

[Interface administrateur 26](#_bookmark76)

[Interface Gestion des enseignants 28](#_bookmark82)

[Interface Gestion des étudiants 30](#_bookmark87)

[Conclusion 32](#_bookmark92)

[Conclusion générale 33](#_bookmark93)

## Table des Figures

[Figure 1 Interface Moodle 4](#_bookmark9)

[Figure 2 l’outil Canvas 5](#_bookmark10)

[Figure 3 l’outil Canvas 5](#_bookmark11)

[Figure 4 Les acteurs de l'application 6](#_bookmark19)

[Figure 5 Diagramme de cas d'utilisation global de l’application << e-learning >> 9](#_bookmark27)

[Figure 6 Diagramme du cas d’utilisation « gestion cours » 10](#_bookmark30)

[Figure 7 Diagramme détaillé du cas d’utilisation « gestion QCM » 11](#_bookmark33)

[Figure 8 Diagramme détaillé du cas d’utilisation « gestion cours » 12](#_bookmark37)

[Figure 9 Diagramme détaillé du cas d’utilisation « gestion QCM » 13](#_bookmark40)

[Figure 10 Diagramme du cas d’utilisation « gestion utilisateurs » 15](#_bookmark44)

[Figure 11 Diagramme détaillé du cas d’utilisation « Gestion des cours » 16](#_bookmark47)

[Figure 12 Diagramme détaillé du cas d’utilisation « Gestion des catégories » 17](#_bookmark50)

[Figure 13 Diagramme détaillé du cas d’utilisation « Gestion résultats test QCM » 19](#_bookmark53)

[Figure 14 diagramme de séquence Inscription et login 20](#_bookmark58)

[Figure 15 Logo MERN 21](#_bookmark63)

[Figure 16 Logo mongodb 21](#_bookmark64)

[Figure 17 Logo Express 22](#_bookmark65)

[Figure 18 Logo Reactjs 22](#_bookmark66)

[Figure 19 Logo nodejs 22](#_bookmark67)

[Figure 20 Logo Docker 23](#_bookmark68)

[Figure 21 Interface docker compose 23](#_bookmark70)

[Figure 22 Interface d'accueil 25](#_bookmark73)

[Figure 23 Interface de connexion 26](#_bookmark75)

[Figure 24 Interface d'inscription 26](#_bookmark77)

[Figure 25 Interface liste des utilisateurs 27](#_bookmark78)

[Figure 26 Interface liste des cours 27](#_bookmark79)

[Figure 27 Interface liste des catégories 27](#_bookmark80)

[Figure 28 Interface liste des résultats pour les différents tests QCM 28](#_bookmark81)

[Figure 29 Interface liste des cours 28](#_bookmark83)

[Figure 30 Interface ajouter cours 29](#_bookmark84)

[Figure 31 Interface ajouter lecture 29](#_bookmark85)

[Figure 32 Interface ajouter test QCM 30](#_bookmark86)

[Figure 33 Interface liste des cours 30](#_bookmark88)

[Figure 34 Interface cours 31](#_bookmark89)

[Figure 35 Interface test QCM 31](#_bookmark90)

[Figure 36 Interface résultats des tests QCM 32](#_bookmark91)

# Liste des tableaux

[Tableau 1 Tableau descriptif du cas d’utilisation de gestion cours 10](#_bookmark31)

[Tableau 2 Tableau descriptif du cas d’utilisation de gestion QCM 11](#_bookmark34)

[Tableau 3 Tableau descriptif du cas d’utilisation de gestion cours 13](#_bookmark38)

[Tableau 4 Tableau descriptif du cas d’utilisation de gestion QCM 14](#_bookmark41)

[Tableau 5 Tableau descriptif du cas d’utilisation de gestion utilisateurs 15](#_bookmark45)

[Tableau 6 Tableau descriptif du cas d’utilisation de gestion cours 16](#_bookmark48)

[Tableau 7 Tableau descriptif du cas d’utilisation de gestion catégories 18](#_bookmark51)

[Tableau 8 Tableau descriptif du cas d’utilisation de gestion résultats test QCM 19](#_bookmark54)

#### Introduction générale

Dans le monde actuel, la digitalisation des services a transformé de nombreux secteurs, et le domaine médical n'y fait pas exception. La gestion des rendez-vous médicaux est une tâche cruciale qui influence directement l'efficacité et la satisfaction des patients ainsi que celle des professionnels de la santé. Souvent, se rendre chez le médecin peut être un exercice fastidieux : il faut d'abord trouver un praticien à proximité, puis patienter dans une salle d'attente, ce qui entraîne une perte de temps significative pour le patient.

De plus, il arrive fréquemment que les patients cherchent un médecin pendant longtemps, alors qu’un professionnel se trouve à quelques pas de chez eux, ou que les cabinets médicaux changent d'adresse sans que l'information soit facilement accessible. Ce manque de coordination et d'information peut engendrer des frustrations et des inefficacités notables.

Face à ces défis, notre projet vise à développer un site web dynamique de gestion des rendez-vous médicaux en ligne. Ce système permettra aux patients de réserver leurs consultations selon des critères spécifiques (localisation, disponibilité et spécialité), tout en offrant aux médecins une interface efficace pour gérer leurs rendez-vous. L'objectif principal est de simplifier le processus de prise de rendez-vous, de réduire les temps d'attente et d'améliorer l'accessibilité aux soins.

En utilisant notre plateforme, les patients pourront facilement trouver des médecins disponibles à proximité, tandis que les médecins pourront optimiser leur emploi du temps et mieux répondre aux besoins de leurs patients. Ce projet représente une avancée significative vers une gestion plus fluide et plus moderne des rendez-vous médicaux, contribuant ainsi à améliorer la qualité des soins et l'expérience des patients.

Le but de ce rapport est de bien comprendre les travaux réalisés et les étapes de leur mise en œuvre. Il est divisé en trois chapitres. Le contenu est le suivant :

* Le premier chapitre présentera le cadre général du projet.
* Le deuxième chapitre présentera spécifiquement les différents besoins de l'application. Puis, je donnerai les étapes de conception de l’application à l’aide de la méthodologie de conception « UML ».
* Dans le troisième chapitre, je présenterai l'environnement d'implémentation et je donnerai les descriptions de certaines interfaces de l'application. Il s’agira également de comprendre ma méthode d'implémentation.

#### Chapitre 1

#### Cadre général du projet

#### Introduction

Dans ce chapitre, je vais commencer par une brève présentation du cadre de mon projet, avant d’étudier et critiquer ce qui existe actuellement dans le but de dégager leurs insuffisances et proposer les orientations de notre solution future.

#### Cadre de Projet

Le projet s'intitule « Conception et production d'applications web pour la gestion des réservations de consultations médicaux » et est réalisé au sein de la Ecole Polytechnique Privée de Monastir. Ce travail entre dans le cadre du projet de fin d’année.

#### Présentation du projet

L'objectif principal de ce projet est de développer une plateforme web dynamique permettant la réservation de consultations médicales en ligne. Cette application vise à simplifier et optimiser le processus de prise de rendez-vous pour les patients et les professionnels de santé.

Les objectifs spécifiques incluent :

**Faciliter l'Accès aux Soins :** Permettre aux patients de trouver et réserver des consultations médicales facilement en fonction de critères tels que la localisation, la disponibilité et la spécialité du médecin.

**Réduire les Temps d'Attente :** Minimiser le temps passé en salle d'attente en permettant aux patients de prendre rendez-vous à l'avance et en informant les médecins de leurs horaires de consultation.

**Améliorer la Gestion des Rendez-Vous :** Offrir aux médecins un outil efficace pour gérer leur emploi du temps, organiser les rendez-vous et optimiser leur pratique médicale.

**Offrir des Services Innovants :** Intégrer des fonctionnalités supplémentaires comme les consultations à distance, le support multilingue et la connexion avec les assurances, pour répondre aux besoins diversifiés des utilisateurs.

#### Contexte et problématique du projet

La prise de rendez-vous médicaux traditionnelle pose des défis majeurs. Tout d'abord, les patients font face à des difficultés pour trouver des médecins disponibles à proximité et pour planifier des consultations à des moments qui leur conviennent. Ensuite, les longues files d'attente en salle d'attente sont une source de frustration pour les patients, et elles peuvent également entraîner des retards dans la prestation des soins. Enfin, du côté des professionnels de la santé, la gestion inefficace des rendez-vous peut entraîner des consultations surchargées ou des périodes de sous-utilisation de leur emploi du temps, compromettant ainsi la qualité des soins prodigués.

#### Etude de l’existant

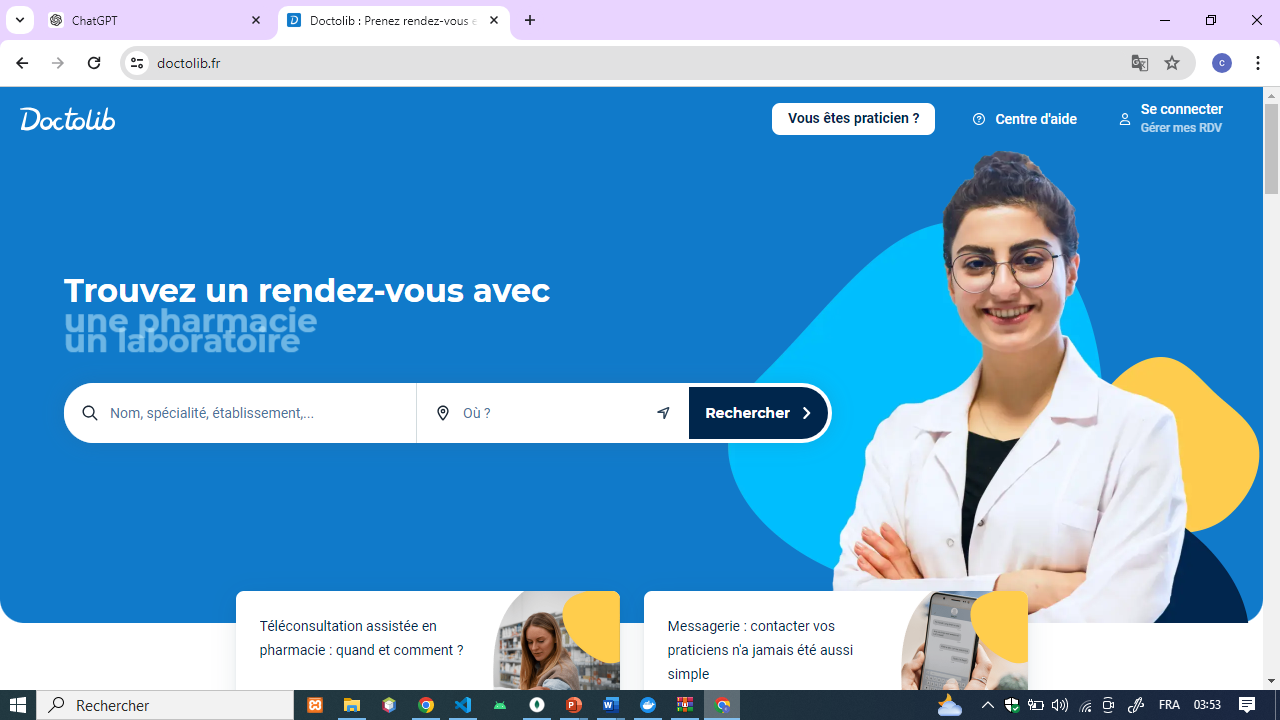
#### L'étude préalable ou de l'existant consiste à analyser les solutions actuellement disponibles sur le marché ainsi que les pratiques courantes dans le domaine de la réservation de consultations médicales. Cette étape permet de mieux comprendre les forces et les faiblesses des systèmes existants, ainsi que les attentes et les besoins des utilisateurs.

#### Analyse de l’existant

##### En examinant les systèmes actuels, nous pourrons identifier les lacunes à combler et les opportunités à saisir pour concevoir une solution innovante et adaptée aux besoins du marché. Cette analyse servira de base solide pour la définition des spécifications fonctionnelles et techniques de notre application de réservation de consultations médicales en ligne.

Un exemple d'application de réservation de rendez-vous médicaux en ligne est "Doctolib". Cette plateforme permet aux patients de rechercher des professionnels de santé dans leur région, de consulter leurs disponibilités en temps réel et de prendre rendez-vous en quelques clics.

**Doctolib :**



**Limites de la plateforme doctolib:**

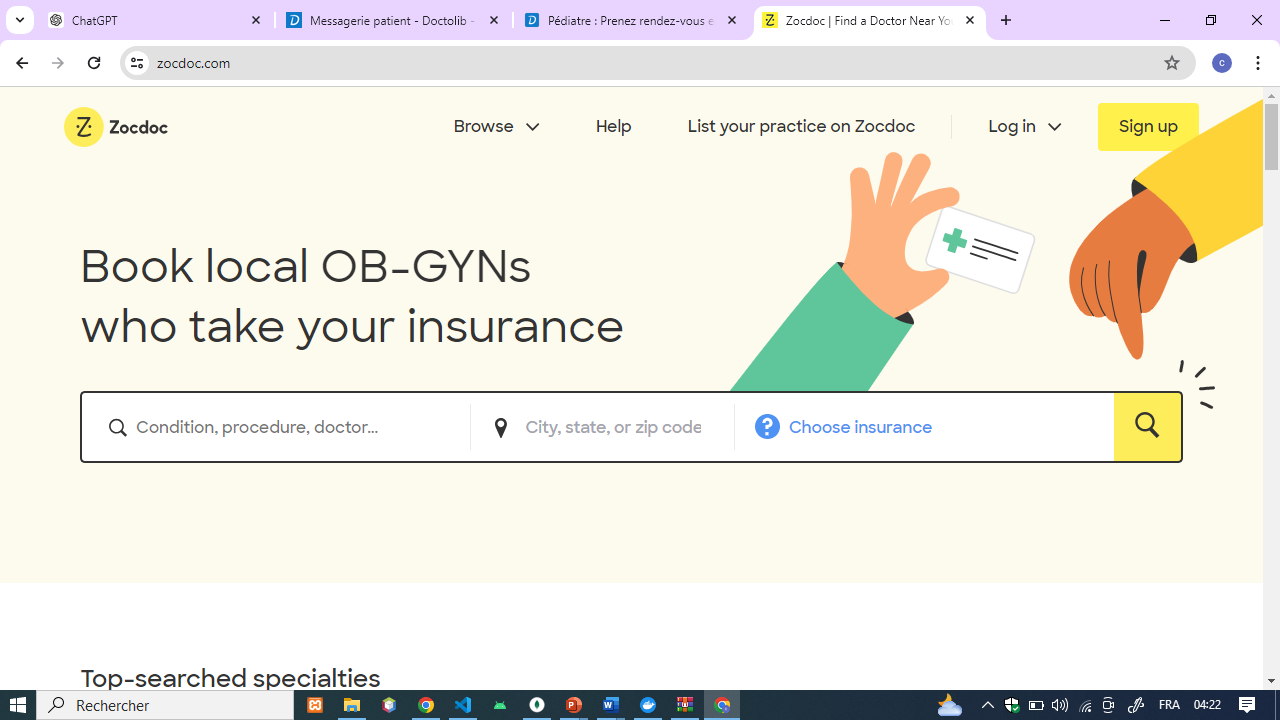
*Figure 1 Interface doctolib*

Cependant, malgré ses avantages, Doctolib présente certaines limitations :

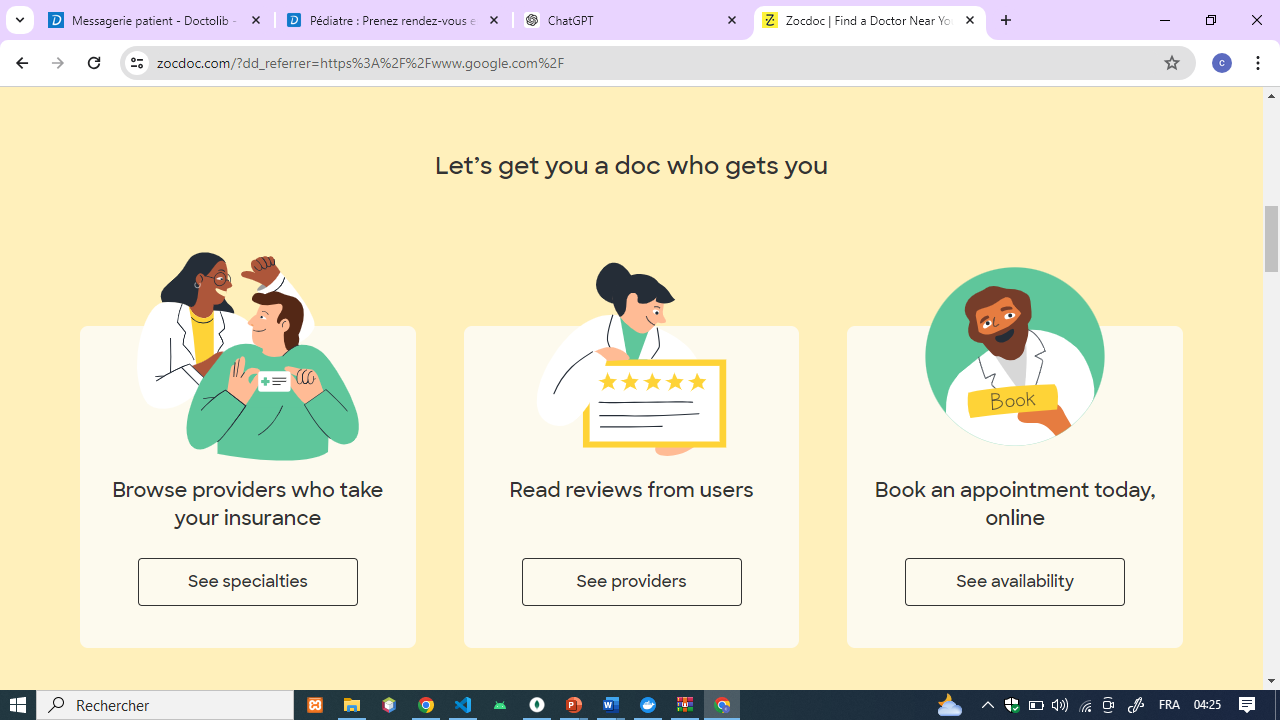
* Manque de disponibilité : Certains médecins ne sont pas inscrits sur la plateforme, limitant ainsi le choix des patients.
* Complexité de navigation : Certains utilisateurs trouvent l'interface de Doctolib complexe à utiliser, en particulier pour les personnes âgées ou peu familières avec la technologie.
* Absence de fonctionnalités avancées : Doctolib offre des fonctionnalités de base pour la réservation de rendez-vous, mais certaines fonctionnalités avancées telles que les consultations vidéo ne sont pas disponibles pour tous les praticiens.

**Zocdoc :**

Un autre exemple d'application de réservation de rendez-vous médicaux en ligne est "Zocdoc". Cette plateforme permet aux utilisateurs de rechercher des professionnels de santé par spécialité, localisation et assurance, et de prendre rendez-vous directement via l'application



*Figure 2 interface Zocdoc*



*Figure 3 interface Zocdoc*

###### Limites de la plateforme Zocdoc :

* Complexité des informations : Certains utilisateurs ont signalé une complexité dans la présentation des informations, rendant parfois difficile la recherche de professionnels de santé spécifiques.
* Restriction géographique : La disponibilité des professionnels de santé peut varier en fonction de la région, limitant ainsi les options pour les patients dans certaines zones.

#### Solution proposée

Face à ces défis, notre application de réservation de rendez-vous médicaux en ligne s'efforce de proposer une solution plus accessible et intuitive. En offrant une interface conviviale, une large gamme de professionnels de santé et des fonctionnalités avancées telles que la géolocalisation et les rappels de rendez-vous, notre application vise à améliorer l'expérience des utilisateurs dans la prise de rendez-vous médicaux.

#### Conclusion

Dans la conclusion de ce premier chapitre, nous avons plongé dans le cœur du projet en exposant le contexte et la problématique qui le sous-tendent. En explorant l'existant, nous avons identifié les forces et les faiblesses des solutions actuellement disponibles sur le marché, telles que Doctolib et Zocdoc, mettant ainsi en lumière les défis persistants dans le processus de réservation de rendez-vous médicaux en ligne.

La phase suivante consistera donc à concevoir une solution qui soit à la fois innovante et fonctionnelle. qui répondra aux attentes des patients et des professionnels de santé, tout en offrant une expérience utilisateur optimale.

#### Chapitre 2

#### Spécification, analyse des besoins et conception

#### Introduction

Dans le deuxième chapitre, nous explorerons en détail les fonctionnalités de notre application et les besoins qu'elle doit satisfaire. Nous utiliserons la méthode UML pour créer des diagrammes de cas d'utilisation, qui décriront les interactions entre les utilisateurs et le système. Ces diagrammes nous aideront à comprendre les actions que les utilisateurs peuvent entreprendre et les résultats attendus. En outre, nous présenterons des diagrammes de séquence pour illustrer comment ces actions sont exécutées. En résumé, ce chapitre nous permettra de clarifier les besoins de notre application et d'établir une base solide pour la conception.Analyse des besoins de mon projet

.

#### Identification des acteurs

Pour mon application « « plateforme d’éducation en ligne » j’ai identifié 3 acteurs :

Administrateur Patient docteur

*Figure 4 Les acteurs de l'application*

**Administrateur**

* **Rôle** : Gérer les informations du système et assurer son bon fonctionnement.
* **Responsabilités** :
  + Gestion des comptes utilisateurs (patients et docteurs)
  + Analyse des besoins en termes de performance et de rapidité
  + Maintenance du système et résolution des problèmes techniques

**Patient**

* **Rôle** : Utiliser l'application pour réserver des consultations médicales.
* **Responsabilités** :
  + Créer et gérer son compte utilisateur.
  + Rechercher des médecins par spécialité .
  + Prendre et gérer des rendez-vous.
  + Consulter l'historique des rendez-vous et des consultations..

**Docteur**

* **Rôle** : Utiliser l'application pour gérer ses consultations et interagir avec les patients.
* **Responsabilités** :
  + Créer et gérer son compte professionnel.
  + Mettre à jour ses disponibilités.
  + Gérer les rendez-vous pris par les patients.

#### Besoins Fonctionnels

Pour concevoir une application de gestion de rendez-vous médicaux en ligne, il est crucial d'identifier à la fois les besoins fonctionnels et non fonctionnels. Voici une liste détaillée pour chaque catégorie.

1. **Administrateur**
   * **Gestion des comptes utilisateurs** :
     + Gérer les comptes patients et docteurs.
     + Gérer les droits d'accès des utilisateurs.
   * **Gestion du système** :
     + Surveiller la performance du système.
     + Mettre à jour les informations du système et résoudre les problèmes techniques.
2. **Patient**
   * **Gestion du compte** :
     + Créer et modifier son profil utilisateur.
     + Changer le mot de passe et récupérer le compte en cas d'oubli.
   * **Recherche et réservation de rendez-vous** :
     + Rechercher des médecins par spécialité et disponibilité.
     + Consulter les profils des médecins, incluant leurs qualifications et avis des patients.
     + Prendre un rendez-vous en ligne.
     + Modifier ou annuler un rendez-vous existant.
   * **Consultation de l'historique** :
     + Consulter l'historique des rendez-vous passés et futurs.
     + Accéder aux notes et prescriptions des consultations précédentes (si disponibles).

* **Avis et commentaires** :
  + - Laisser des avis et des notes sur les docteurs après les consultations.
    - Consulter les avis et les notes laissés par d'autres patients.

1. **Docteur**
   * **Gestion du compte professionnel** :
     + Créer et modifier son profil professionnel.
     + Mettre à jour ses qualifications, spécialités et horaires de disponibilité.
   * **Gestion des rendez-vous** :
     + Voir et gérer les rendez-vous pris par les patients.
     + Confirmer, modifier ou annuler les rendez-vous.
   * **Consultation des avis :**
     + Consulter les avis et les notes laissés par les patients après les consultations.
     + Répondre aux avis des patients pour améliorer la communication et la satisfaction des patients**.**

#### Besoins non Fonctionnels

**Performance**

* L'application doit être capable de gérer un grand nombre de requêtes simultanées sans ralentissement.
* Les temps de réponse pour les actions de l'utilisateur doivent être inférieurs à deux secondes.

**Sécurité**

* Les données des utilisateurs doivent être protégées par des mesures de sécurité robustes, telles que le chiffrement et des politiques de mot de passe fortes.
* L'application doit être conforme aux régulations en vigueur concernant la protection des données personnelles (ex. RGPD).

**Fiabilité**

* L'application doit avoir un temps de disponibilité d'au moins 99,9%.
* Des mécanismes de sauvegarde régulière des données doivent être mis en place.

**Scalabilité**

* L'application doit être conçue pour s'adapter à une augmentation du nombre d'utilisateurs sans dégradation de la performance.

**Maintenabilité**

* Le code de l'application doit être bien documenté pour faciliter les mises à jour et les corrections de bugs.
* L'architecture doit permettre une extension facile des fonctionnalités.

**Utilisabilité**

* L'interface utilisateur doit être intuitive et facile à utiliser pour les patients, les docteurs et les administrateurs.
* Une assistance en ligne et des tutoriels doivent être disponibles pour aider les utilisateurs à naviguer dans l'application.

#### Choix du langage de modélisation UML

Pour la modélisation de notre application de gestion de rendez-vous médicaux en ligne, nous avons choisi d'utiliser le langage UML (Unified Modeling Language). Ce choix se justifie par plusieurs raisons, qui démontrent la pertinence et l'efficacité d'UML dans le cadre de notre projet.

#### Présentation d’UML

UML, ou Unified Modeling Language, est un langage de modélisation graphique utilisé pour visualiser, spécifier, construire et documenter les artefacts d'un système logiciel. Il est largement adopté dans le domaine de l'ingénierie logicielle pour sa capacité à représenter de manière compréhensible et standardisée les différents aspects d'un système.

#### Caractéristiques

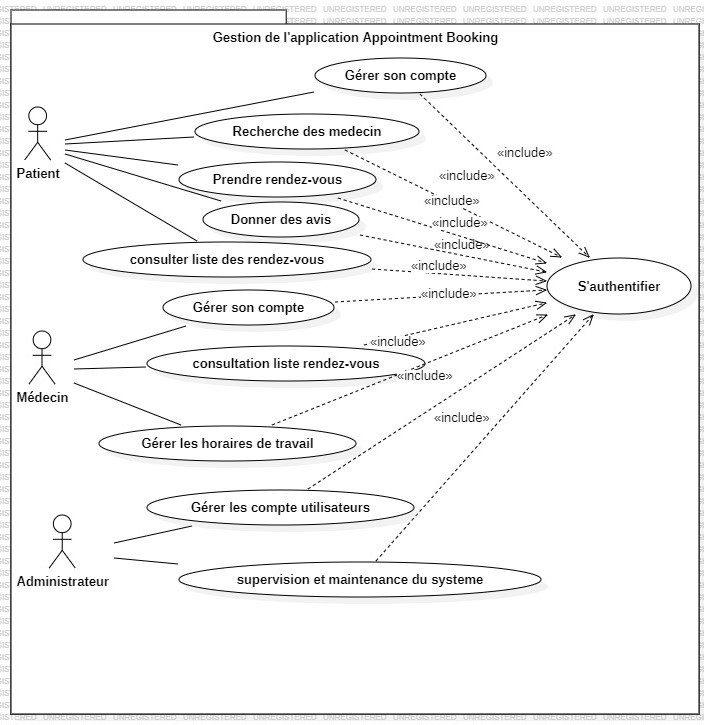
UML est une méthode d'expression du modèle objet qui ne tient pas compte de la réalisation du modèle objet. Le modèle fourni par UML, une fois obtenu, est ainsi valide pour tout langage de programmation.

UML est en fait un langage basé sur un méta-modèle. Ce dernier est un modèle de haut niveau qui définit à la fois les éléments (concepts disponibles) d'UML et leur sémantique (leurs significations et leur utilisation) et ce, à un niveau d'abstraction plus élevé. Celui-ci est dans les faits conçu pour être plus général que le modèle qu'il permet de construire. Le méta-modèle UML est aussi un langage sans ambiguïté qui peut être utilisé comme un langage universel. Cela signifie qu’il peut être pris en charge par n'importe quel langage orienté objet et par une méthode de définition de la structure du programme.

#### Diagramme de cas d’utilisation

Le diagramme des cas d'utilisation représente les opérations effectuées par le système. Les résultats peuvent répondre aux besoins spécifiques des participants. Je vais d'abord montrer le diagramme des cas d'utilisation au complet. Ensuite, je vais continuer à présenter le diagramme des cas d'utilisation détaillés de chaque acteur du système.

#### Diagramme des cas d’utilisation global



*Figure 5 Diagramme de cas d'utilisation global de l’application << Prise de rendez-vous m >>*

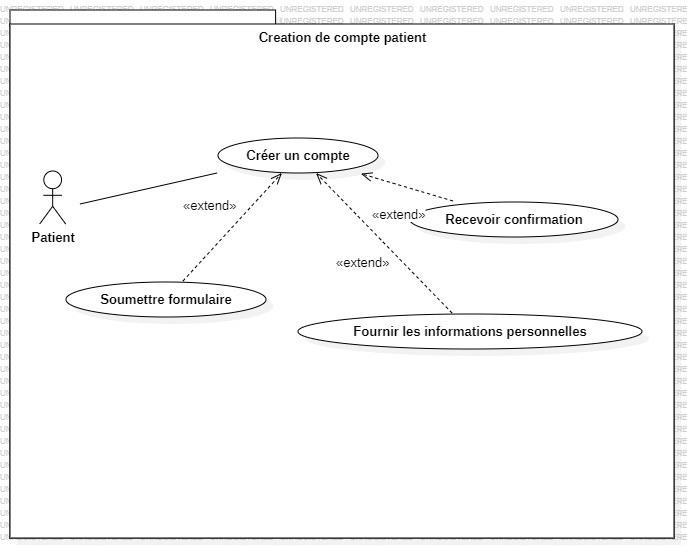
Les cas d'utilisation détaillés ci-dessus montrent les principales fonctionnalités offertes aux administrateurs, patients, et médecins sur la plateforme de gestion de rendez-vous médicaux en ligne. En modélisant ces interactions, nous pouvons mieux comprendre les responsabilités de chaque acteur et garantir que le système répond efficacement à leurs besoins.

Ces cas d'utilisation serviront de base pour la conception et le développement des fonctionnalités associées, assurant ainsi une application complète et conviviale pour tous les utilisateurs.

#### Diagramme des cas d’utilisation Patient :

#### Diagramme du cas d’utilisation « Création du compte » :

La figure 6 représente le diagramme détaillé du cas d’utilisation de création de compte :



*Figure 6 Diagramme du cas d’utilisation « Création du compte »*

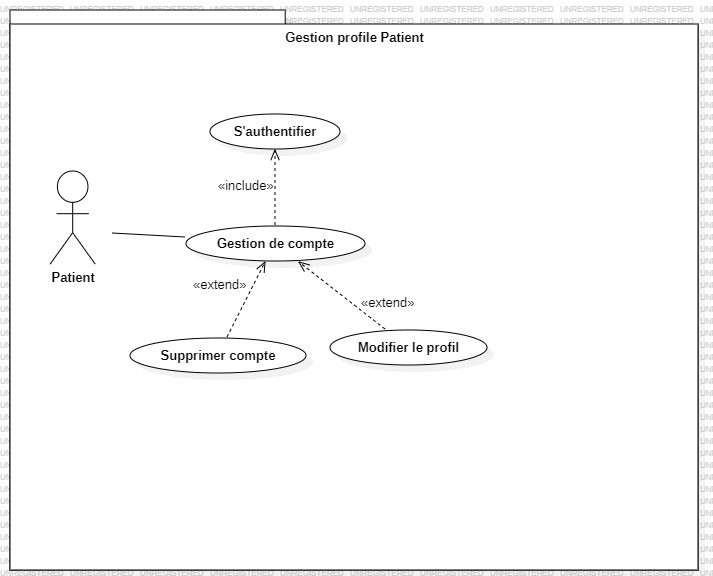
Dans le tableau 1, j’ai donné une description partielle du cas d’utilisation : Création de compte :

|  |  |
| --- | --- |
| **L’action** | Créer un compte |
| **L’acteur** | Patient |
| **Description** | Le patient veut créer un compte sur le système |
| **Précondition** | Aucune |
| **Scénario normale** | |
| 1. Le patient accède à la page d'inscription. 2. Le patient fournit ses informations. 3. Le patient soumet le formulaire d'inscription. 4. Le système vérifie les informations fournies 5. Si les informations sont valides, le système crée un nouveau compte utilisateur. 6. Le patient reçoit une confirmation | |
| **Post conditions** | Un nouveau compte patient est créé et enregistré dans le système |

*Tableau 1 : Tableau descriptif du cas d’utilisation de Création de compte*

#### Diagramme du cas d’utilisation « Gestion Profile »

La figure 7 représente le diagramme détaillé du cas d’utilisation de gestion de profile.



*Figure 7 Diagramme du cas d’utilisation « gestion profile »*

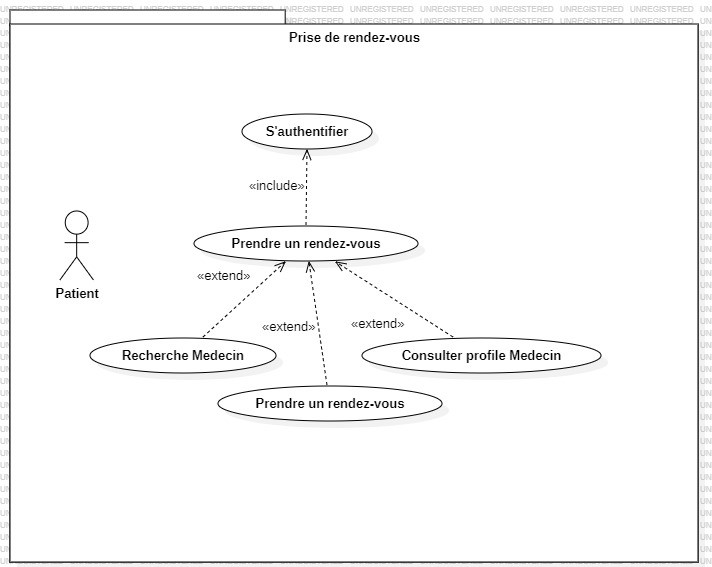
Dans le tableau 2, j’ai donné une description partielle du cas d’utilisation : gestion profile :

|  |  |
| --- | --- |
| **L’action** | Gérer profile |
| **L’acteur** | Patient |
| **Description** | le patient peut mettre à jours ses information personnels |
| **Précondition** | Le patient doit s’authentifier. |
| **Scénario normale** | |
| 1. Consulter profile. 2. Soit :    * Modification des informations.    * Supprimer compte | |
| **Post conditions** | 1. Consulter profile. 2. Modification profile. 3. Suppression compte |

*Tableau 2 Tableau descriptif du cas d’utilisation de gestion profile*

#### Diagramme détaillé du cas d’utilisation « Prise de rendez-vous »

La *figure 8* représente le diagramme détaillé du cas d’utilisation : Prise de rendez-vous



*Figure 8 Diagramme détaillé du cas d’utilisation « Prise de rendez-vous »*

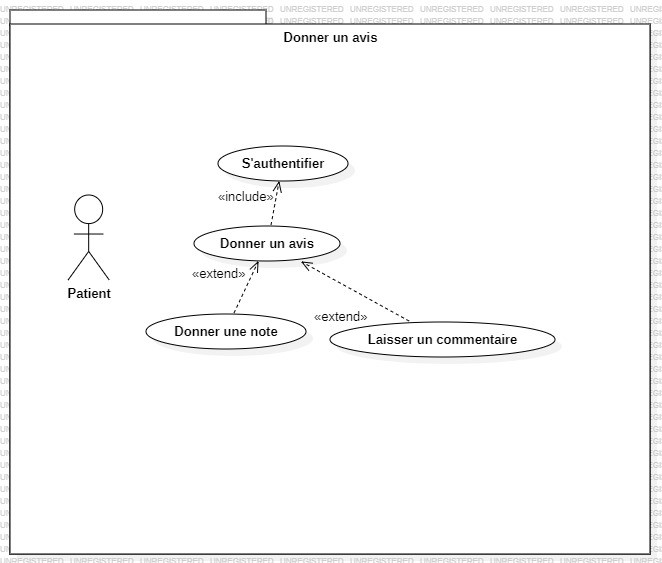
Dans le tableau 3, j’ai donné une description partielle du cas

|  |  |
| --- | --- |
| **L’action** | Prendre rendez-vous |
| **L’acteur** | Patient |
| **Description** | Le patient effectue l'ensemble des actions nécessaires pour prendre un rendez-vous avec un médecin. |
| **Précondition** | Le patient doit être authentifié |
| **Scénario normale** | |
| 1. Le patient consulte le profil des médecins trouvés. 2. Le patient consulte le profil des médecins trouvés. 3. Le patient prend un rendez-vous avec le médecin sélectionné. | |
| **Post conditions** | 1. Une liste de médecins correspondant aux critères de recherche est affichée. 2. Le rendez-vous est confirmé et enregistré dans le système. |

*Tableau 3 Tableau descriptif du cas d’utilisation de Prise de rendez-vous*

#### Diagramme du cas d’utilisation « Donner un avis »

La *figure 9* représente le diagramme détaillé du cas d’utilisation :Donner un avis :



*Figure 9 Diagramme détaillé du cas d’utilisation « Donner un avis »*

Dans le tableau 4, j’ai donné une description partielle du cas d’utilisation : Donner un avis :

|  |  |
| --- | --- |
| **L’action** | Donner un avis |
| **L’acteur** | Patient |
| **Description** | Le patient donne une note au médecin, sous forme de nombre d'étoiles ou d'une échelle de 1 à 5.  Le patient peut écrire un commentaire sur la consultation avec le médecin, décrivant son expérience et ses impressions. |
| **Précondition** | Le patient doit être authentifié et doit avoir eu une consultation avec le médecin. |
| **Scénario normale** | |
| 1. Le patient se connecte au système. 2. Consulte le profil de médecin. 3. Le patient laisse une note et un commentaire sur le médecin. | |

|  |  |
| --- | --- |
|  | |
| **Post conditions** | 1. La note et le commentaire sont enregistrés dans le système et visible par d'autres patients. |

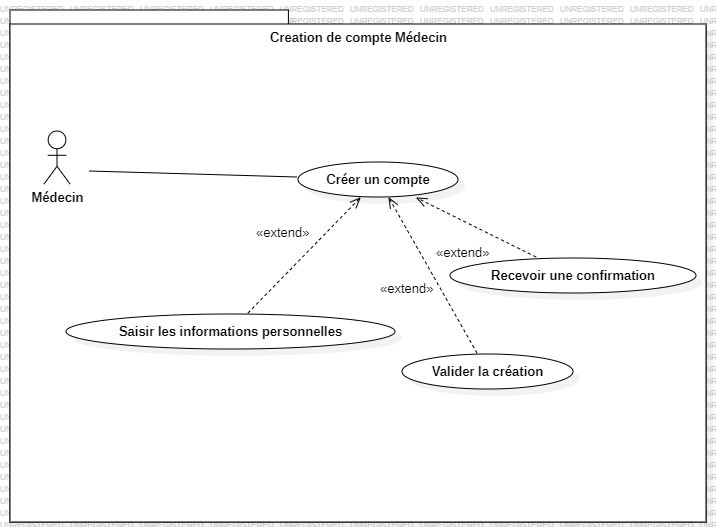
*Tableau 4 Tableau descriptif du cas d’utilisation de donner un avis*

#### 

#### Diagramme de cas d’utilisation Médecin

#### Diagramme détaillé du cas d’utilisation « Création de compte »

La *figure 10* représente le diagramme détaillé du cas d’utilisation de Création de compte :



*Figure 10 Diagramme détaillé du cas d’utilisation « Création de compte »*

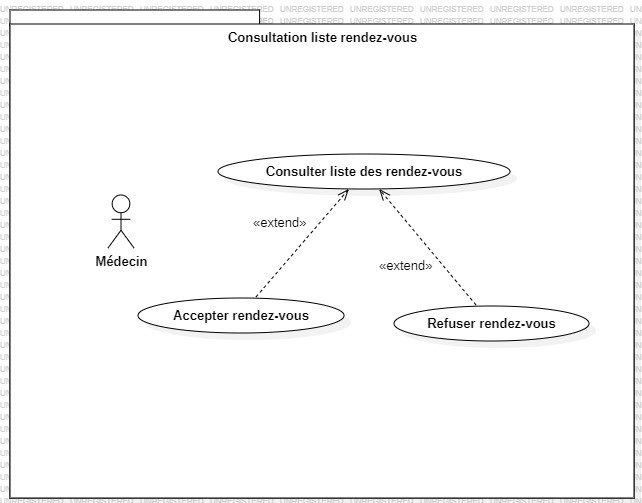
Dans le tableau 5, j’ai donné une description partielle du cas :

|  |  |
| --- | --- |
| **L’action** | Créer un compte |
| **L’acteur** | Médecin |
| **Description** | Le médecin veut créer un compte sur le système |
| **Précondition** | Aucune |
| **Scénario normale** | |
| 1. Le médecin accède à la page d'inscription. 2. Le médecin fournit ses informations . 3. Le médecin soumet le formulaire d'inscription. 4. Le système vérifie les informations fournies 5. Si les informations sont valides, le système crée un nouveau compte utilisateur. 6. Le médecin reçoit une confirmation | |
| **Post conditions** | Un nouveau compte est créé et enregistré dans le système |

*Tableau 5 Tableau descriptif du cas d’utilisation de création de compte*

#### Diagramme détaillé du cas d’utilisation « Consultation Rendez-vous Médecin »

La *figure 11* représente le diagramme détaillé du cas d’utilisation de consultation de rendez-vous :



*Figure 11 Diagramme détaillé du cas d’utilisation « consultation rendez-vous »*

Dans le tableau 6, j’ai donné une description partielle du cas

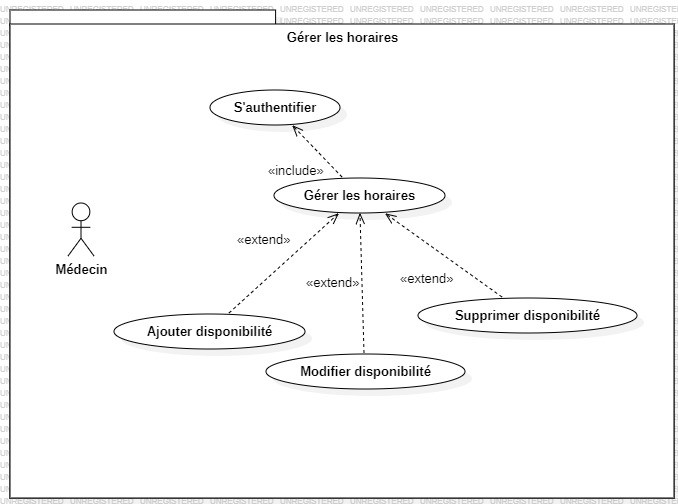
:

|  |  |
| --- | --- |
| **L’action** | Consultation liste de rendez-vous |
| **L’acteur** | Médecin |
| **Description** | Le médecin peut consulter les détails d'un rendez-vous spécifique . |
| **Précondition** | Le médecin doit être authentifié |
| **Scénario normale** | |
| 1. Consulter détails rendez-vous 2. Filtrer rendez-vous : soit accepter ou refuser | |
| **Post conditions** | La liste des rendez-vous est filtrée selon les critères sélectionnés. |

*Tableau 6 Tableau descriptif du cas d’utilisation de gestion QCM*

#### Diagramme détaillé du cas d’utilisation « Gestion des horaires »

La *figure 12* représente le diagramme détaillé du cas d’utilisation de gestion des horaires :



*Figure 12 Diagramme détaillé du cas d’utilisation « Gestion des horaires »*

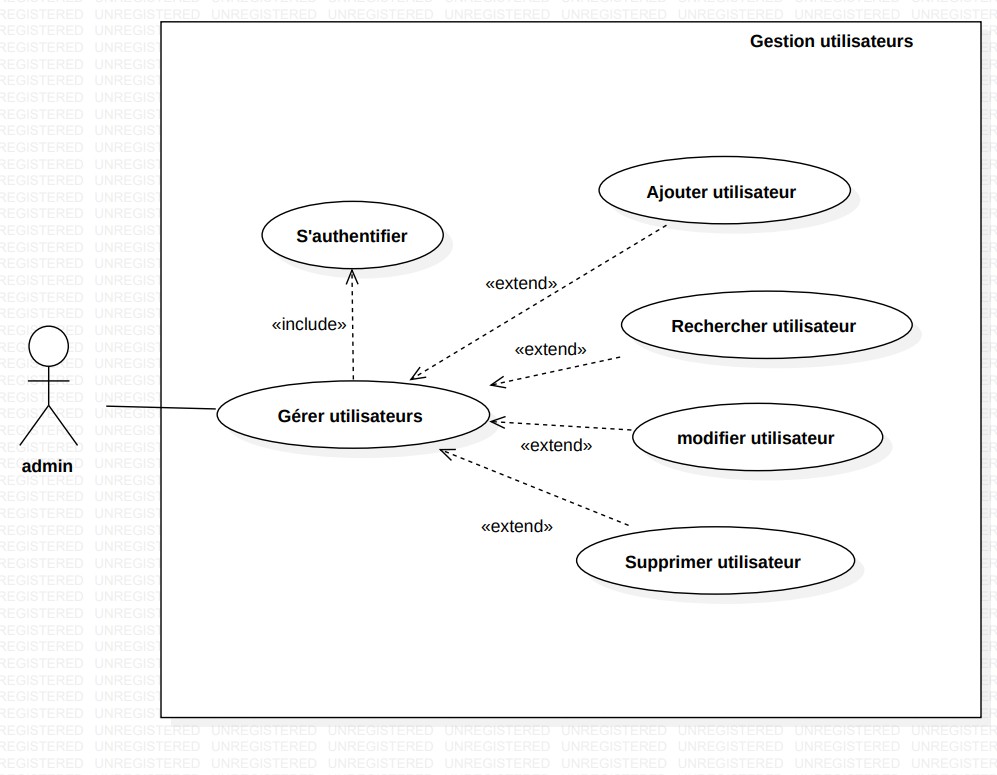
#### Diagramme des cas d’utilisation admin

#### Diagramme du cas d’utilisation « gestion de compte utilisateur»

La *figure 13* représente le diagramme détaillé du cas d’utilisation de gestion des utilisateurs.

L’admin peut facilement mettre à jour les utilisateurs par :

* + - Ajouter utilisateur.
    - Rechercher utilisateur.
    - Modifier utilisateur.
    - Supprimer utilisateur.



*Figure 13 Diagramme du cas d’utilisation « gestion utilisateurs »*

Dans le tableau 4, j’ai donné une description partielle du cas

|  |  |
| --- | --- |
| **L’action** | Gérer utilisateurs. |
| **L’acteur** | Administrateur |
| **Description** | L’administrateur peut gérer l’utilisateurs par l’ajout d’utilisateur où modification, où par  suppression, ou par la recherche. |
| **Précondition** | L’administrateur doit s’authentifier. |
| **Scénario normale** | |
| 1. Ajout un utilisateur. 2. Rechercher utilisateur. 3. Modifier utilisateur. 4. Supprimer utilisateur. | |
| **Post conditions** | 1. Ajout un utilisateur. 2. Rechercher utilisateur 3. Modifier utilisateur. 4. Supprimer utilisateur. |

*Tableau 7 Tableau descriptif du cas d’utilisation de gestion utilisateur*

#### Diagramme des séquences

L’objectif du diagramme de séquence est de représenter les interactions entre objets en indiquant la chronologie des échanges. Cette représentation peut se réaliser par cas

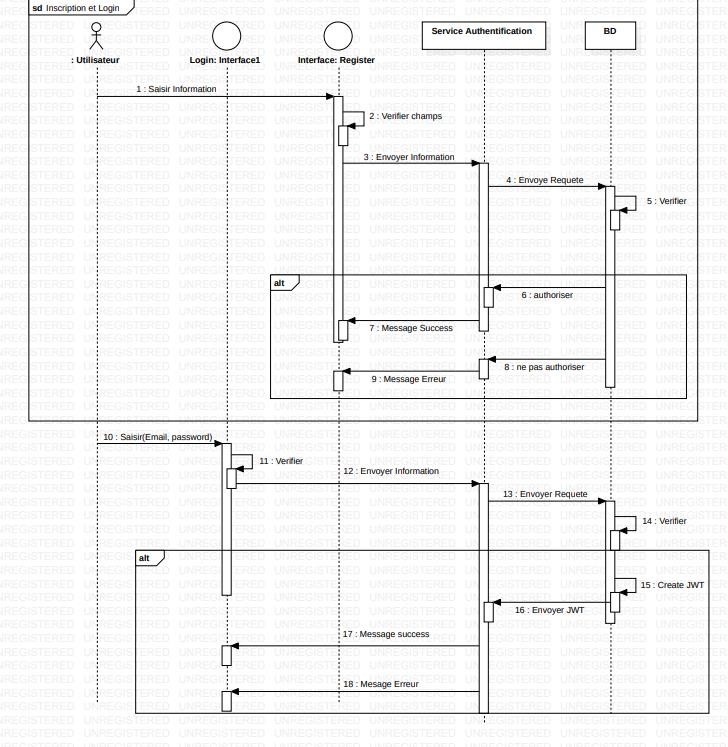
d’utilisation en considérant les différents scénarios associés.

#### Diagramme des séquences

L’objectif du diagramme de séquence est de représenter les interactions entre objets en indiquant la chronologie des échanges. Cette représentation peut se réaliser par cas

d’utilisation en considérant les différents scénarios associés.

#### Inscription et Login



#### Conclusion

*Figure 14 diagramme de séquence Inscription et login*

Dans ce chapitre, j'ai précisé les exigences de mon application afin d'avoir une compréhension plus claire de mes besoins et une compréhension plus approfondie des tâches à effectuer.

Cette recherche me permet de comprendre les fonctionnalités de mon système pour faciliter la mise en place de mon application.

Dans les chapitres suivants, je présenterai l'environnement matériel et le logiciel que j'ai utilisés pour implémenter mon application. Pour terminer, j'aborderai l'étape d'implémentation.

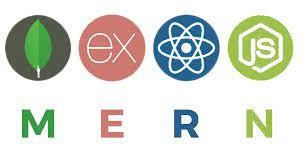
#### Chapitre 3 Réalisation

### Introduction

Dans ce chapitre, je commence par la présentation d'autres choix technologiques. Je vais d'abord décrire mon environnement matériel et logiciel ainsi que les différents outils de développement utilisés lors de la réalisation de l'application. Ensuite, je discuterai de l'architecture et traiterai tous les détails de l’implémentation, en décrivant l’interface d'implémentation.

#### Choix Technologiques Langage de développement

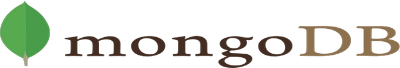
Pour développer mon application, j’ai utilisé :



*Figure 15 Logo MERN*

**MERN** est un acronyme qui désigne un ensemble de technologies populaires utilisées pour le développement d'applications web. Il représente les technologies suivantes :

* **MongoDB :** une base de données NoSQL qui utilise des documents JSON pour stocker les données.



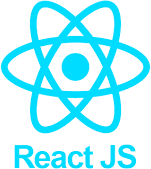
*Figure 16 Logo mongodb*

* **Express.js :** un framework web minimaliste et flexible pour créer des applications web basées sur Node.js.



*Figure 17 Logo Express*

* **React :** une bibliothèque JavaScript pour la création d'interfaces utilisateur réactives et dynamiques.



*Figure 18 Logo Reactjs*

* **Node.js :** un environnement d'exécution JavaScript côté serveur qui permet d'exécuter du code JavaScript en dehors d'un navigateur web.



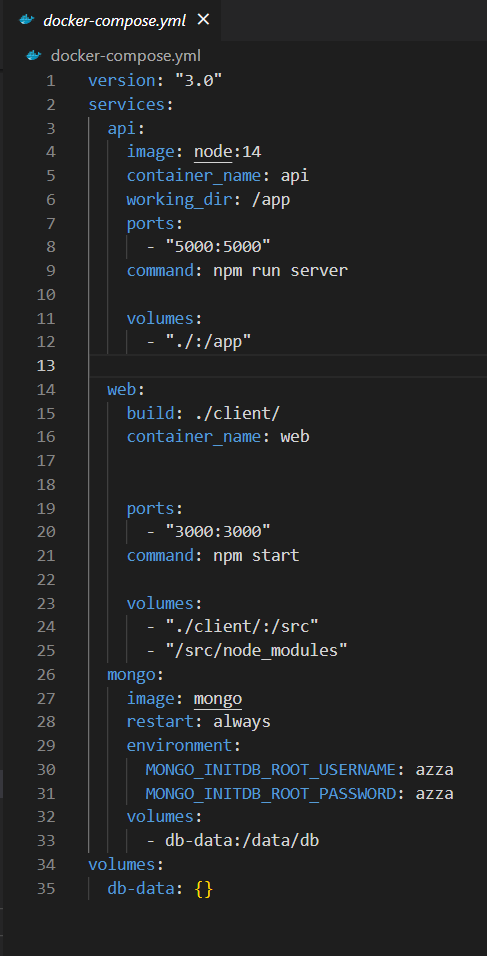
*Figure 19 Logo node.js*

**Docker :** Plateforme open-source pour l'automatisation du déploiement et de la gestion d'applications dans des conteneurs.



*Figure 20 Logo Docker*

**Docker Compose** : Outil pour décrire et gérer des applications multi-conteneurs à l'aide d'un fichier de configuration YAML.



#### Environnement logiciel

*Figure 21 Interface docker compose*

**Visual Studio Code :** est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS.

###### Architecture de l’application

*Figure 22 Logo visual studio code*

En informatique, l'architecture spécifie la structure générale inhérente d'un système informatique, l'organisation de divers éléments du système (logiciels et/ou matériels et/ou personnes et/ou informations) et les relations entre les éléments. Cette structure fait suite à un ensemble de décisions stratégiques prises lors de la conception de la totalité ou de la partie du système informatique. L’architecture de notre projet est en fait une architecture MVC.

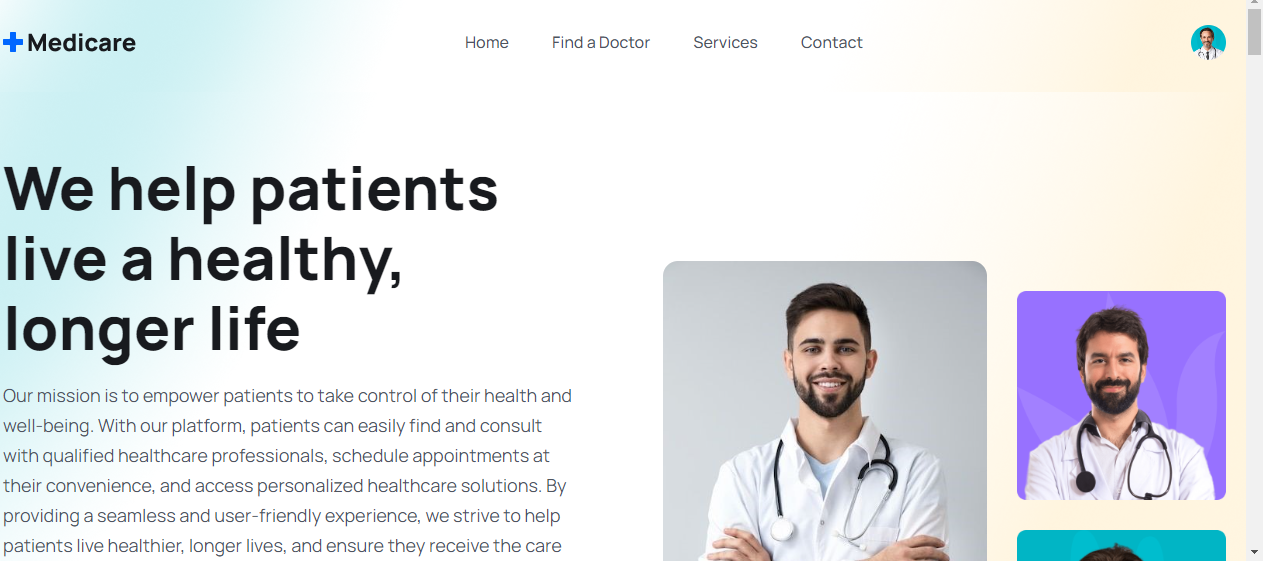
**StarUML :** C’est un outil de modélisation en UML utilisé pendant la phase de conception.

#### Présentation et description de l’application « Medicare »

Dans cette partie, je vais montrer quelques-unes des interfaces de mon application, et je vais essayer de décrire les différents objets interactifs disponibles pour l'utilisateur.

#### Interface de lancement de l’application

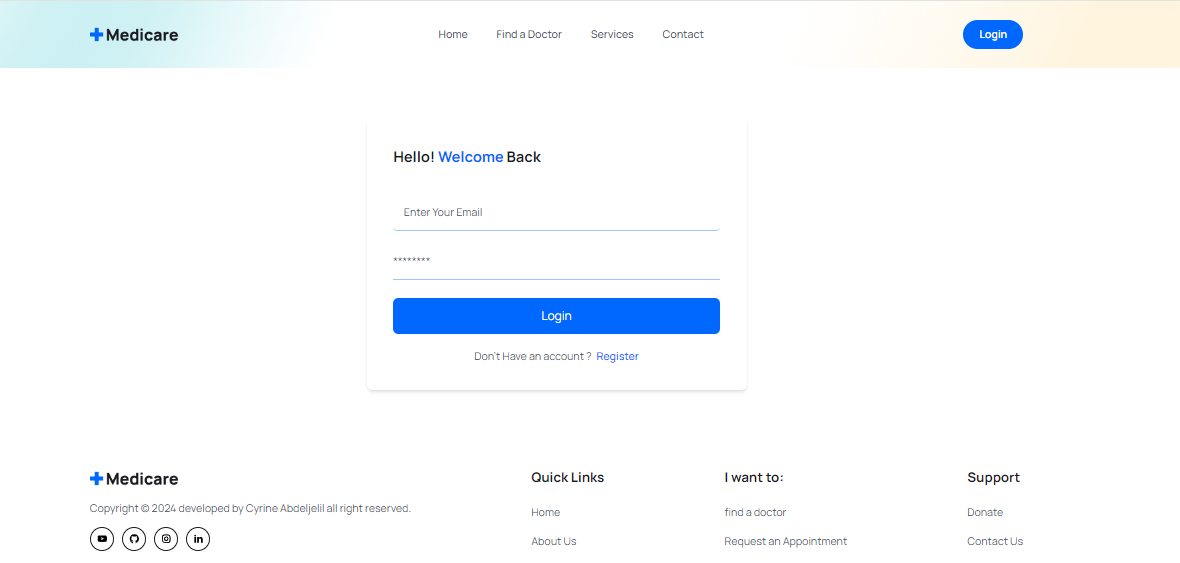
Au démarrage de l'application, l'interface d'accueil est comme indiquée sur la figure 11. Les utilisateurs peuvent « Se Connecter », s'ils ont déjà un compte. Sinon, ils doivent « S’inscrire » et ils peuvent ainsi facilement créer un compte.



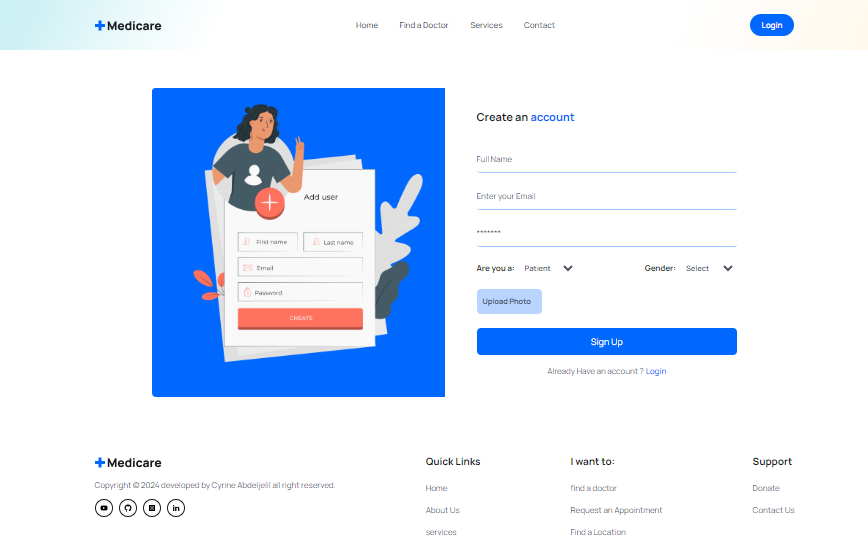
*Figure 23 Interface d'accueil*

#### Interfaces d’authentification et d’inscription

Une interface avec deux champs email et mot-de-passe ainsi qu’un bouton de connexion apparaissent. Il y a également sur l’interface de s’inscrire à l’étudiant et à l’enseignant.



*Figure 24 Interface de connexion*

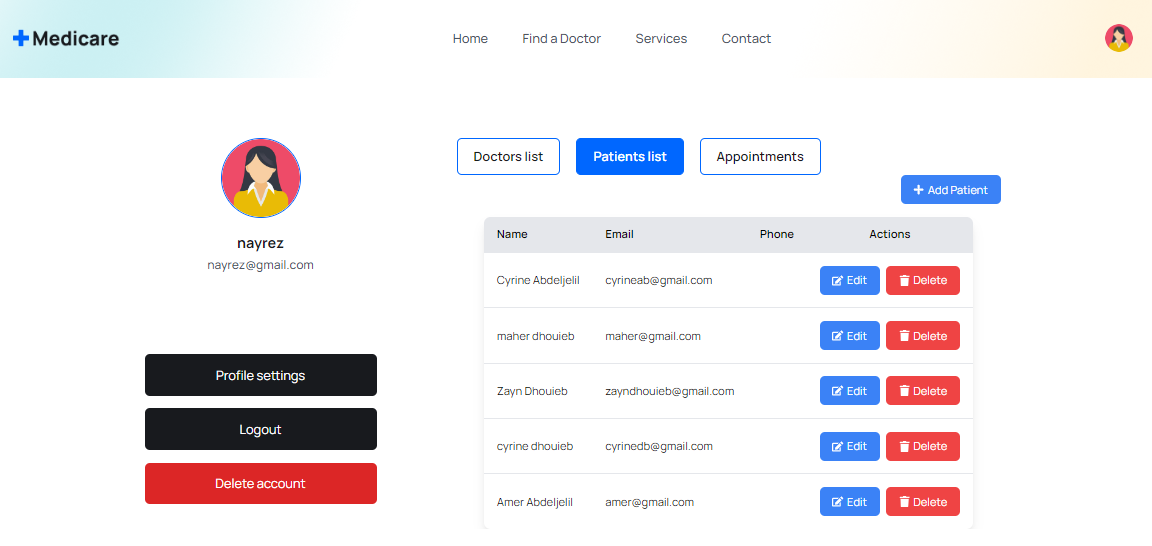


#### Interface administrateur

*Figure 25 Interface d'inscription*

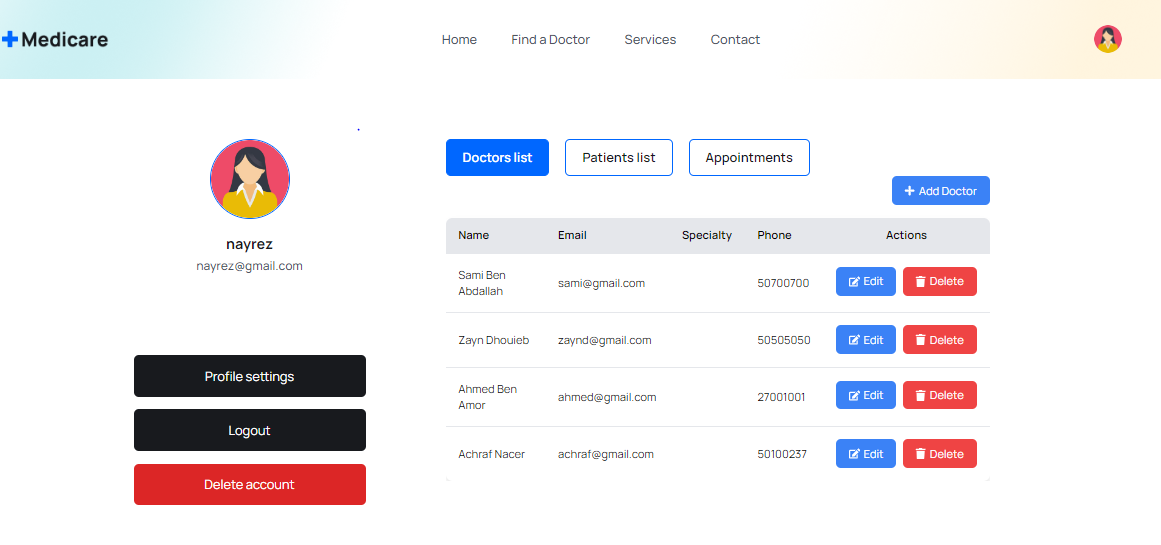
Cette interface qui représente la liste des utilisateurs. L’admin peut créer un nouvel

utilisateur (patient ou médecin) selon le rôle. Ainsi qu’il peut être modifié, supprimer un user.



*Figure 26 Interface liste des patients*

Cette interface représente que la liste des patients. L'admin peut modifier où supprimer un patient.

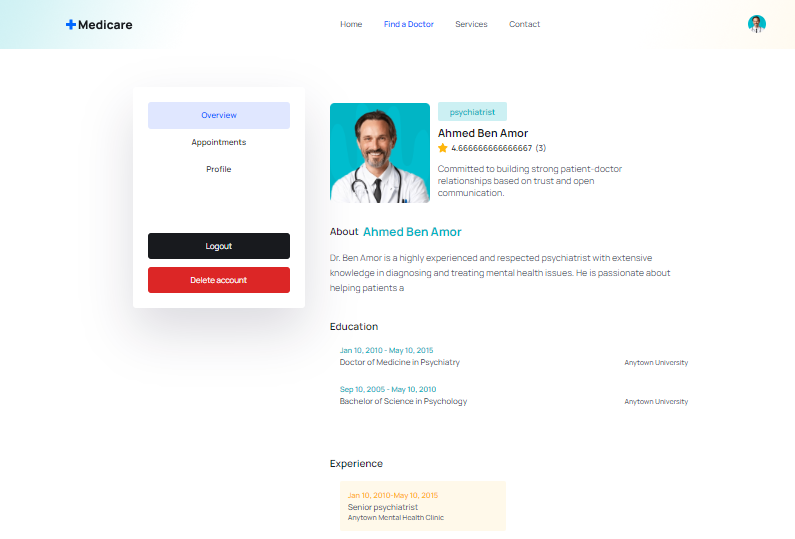


*Figure 27 Interface liste des Médecin*

Cette interface représente que la liste des médecins. L'admin peut modifier où supprimer un médecin.

#### Interface Medecin :

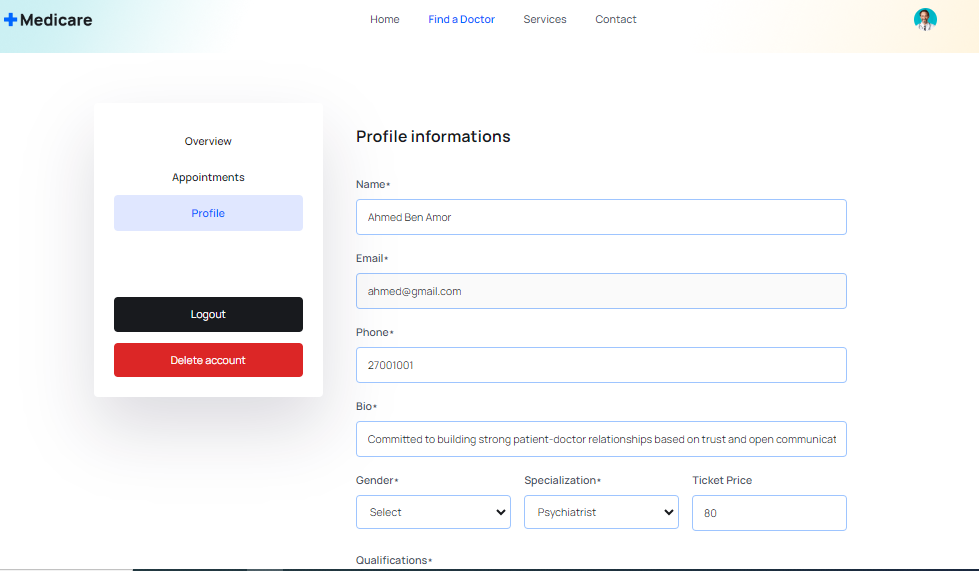
Cette interface réprédente l’espace Médecin :



*Figure 30 Interface Profile Médecin*

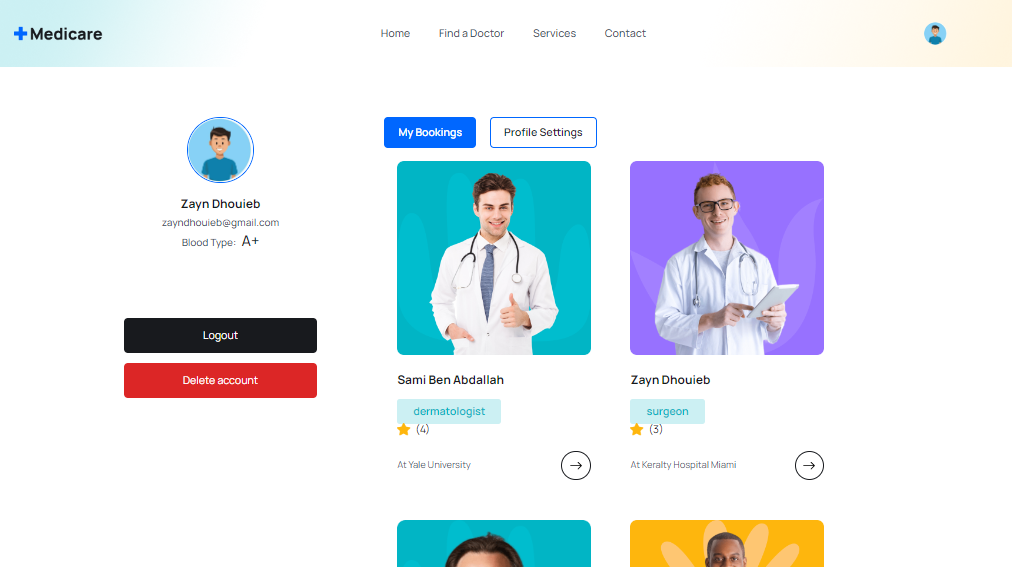
A partir de cette interface le médecin peut :

* Accéceder à la liste des rendez-vous.
* Modifier ses informations
* supprimer son compte .
* se déconnecter



*Figure 31 Interface mise à jour profile médecin*

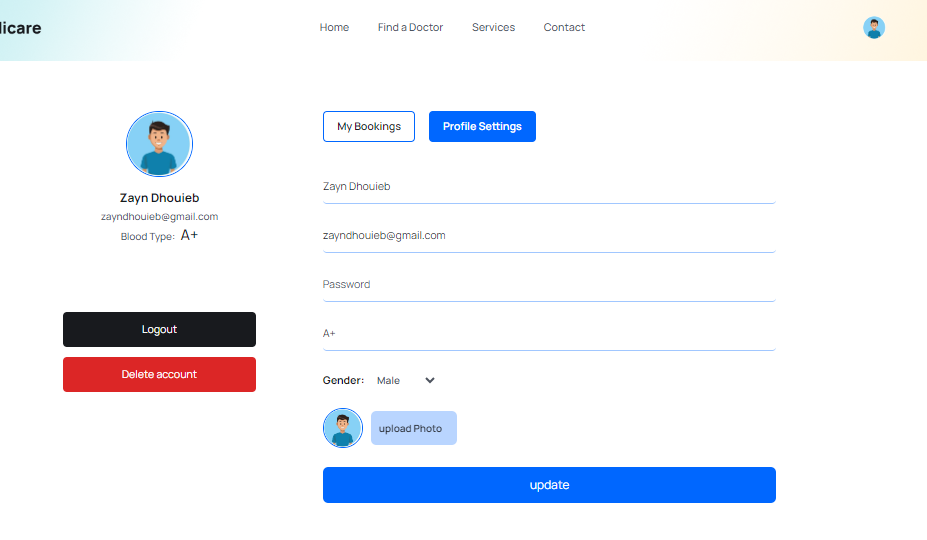
#### Espace Patient :



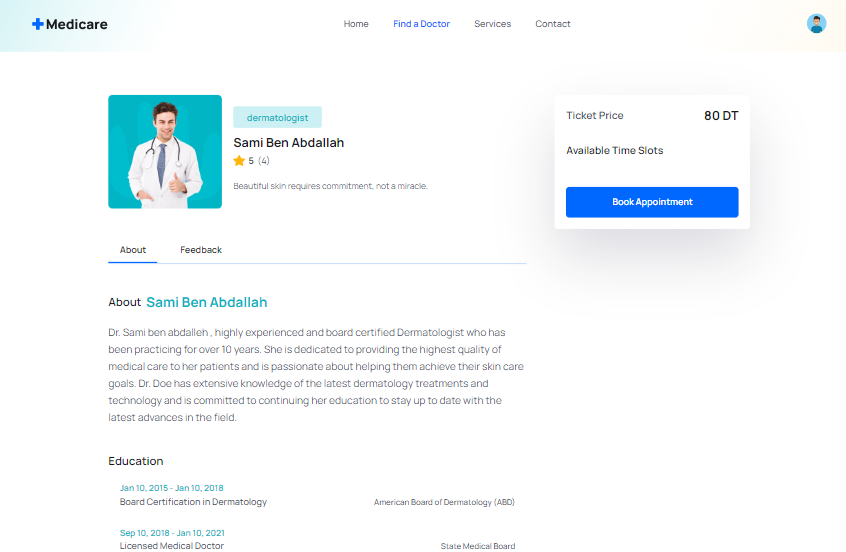
*Figure 32 Interface Patient*

A travers cette interface le patient peut :

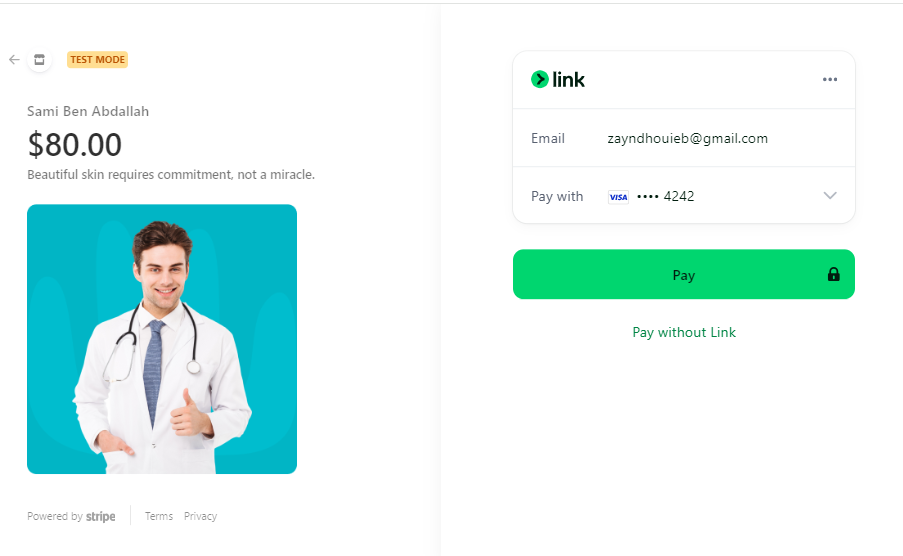
* Consulter les rendez-vous
* Faire le mise à jour de ses informations
* Supprimer son compte
* Se déconnecter



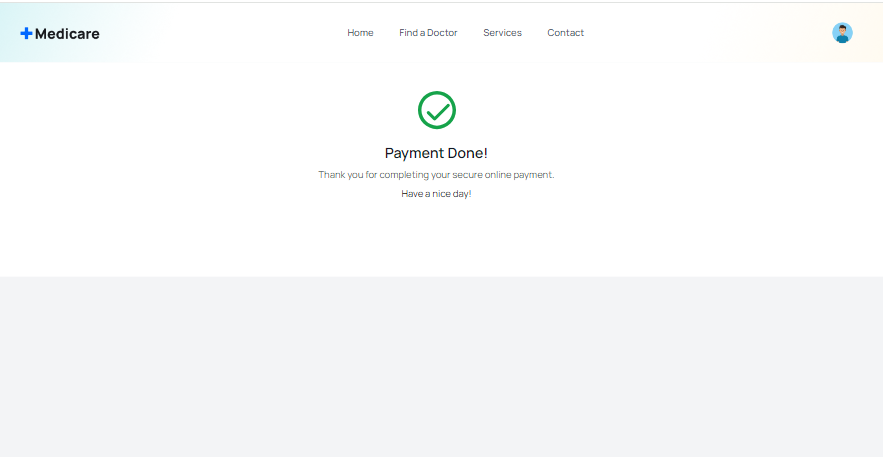
*Figure 35 Interface mise à jours profile client*



*Figure 36 Interface de réservation de rendez-vous*



*Figure 36 Espace paiement*



*Figure 36 Espace paiement validée*

#### Conclusion :

#### Ce chapitre a été dédié à la description de la partie « implémentation de mon application ». J’ai fourni des interfaces pour expliquer le fonctionnement de mon projet

Dans le cadre de mon projet de développement, axé sur la prise de rendez-vous en ligne, j'ai travaillé sur la conception et la réalisation d'une plateforme intuitive et sécurisée. Cette expérience m'a offert l'opportunité d'explorer en profondeur les technologies liées au développement web, notamment en ce qui concerne le front-end et le back-end.

En mettant en place cette solution, j'ai pu répondre aux besoins croissants des utilisateurs en matière de réservation de rendez-vous, en offrant une expérience utilisateur fluide et efficace. De plus, j'ai eu l'occasion d’améliorer mes compétences en matière de gestion de bases de données, de sécurisation des données sensibles et de déploiement d'applications web.

En regardant vers l'avenir, je prévois d'ajouter de nouvelles fonctionnalités à la plateforme, telles que la gestion avancée des rendez-vous, la synchronisation avec les calendriers personnels des utilisateurs, ainsi que des options de communication en temps réel avec les professionnels de santé. Cette évolution permettra de renforcer l'efficacité et la pertinence de la plateforme dans le domaine de la santé et du bien-être.

En conclusion, ce projet m'a permis de mettre en pratique mes connaissances et compétences en développement web tout en contribuant à la résolution d'un problème concret. Il représente une étape importante dans mon parcours professionnel et m'a donné la motivation nécessaire pour continuer à explorer de nouvelles technologies et à relever de nouveaux défis dans le domaine du développement d'applications en ligne.