##### 

Rapport de mini-projet

Site-web

Thème: Conception et développement d’un site Prise de rendez-vous médical en ligne

Elabore par : Moulazem Omayma Encadre par : MME Rahali Hajer

Département : Informatique

Spécialité : Business intelligence

Niveau : 2 Cycle d'ingénieur

Année universitaire 2023/2024

**Sommaire**

Introduction generale .................................................................................................... ...............1

chapitre 1 : Analyse et spécification des besoins...................................................................2

1. Introduction ..................................................................................................................................... 3
2. Problématique .................................................................................................................................. 5
3. Solution Proposé .......................................................................................................................... 5
4. Les Besoins non fonctionnels et besoins fonctionnels ................................................................ 6
   1. Les besoins non fonctionnels....................................................................................................... 7
   2. Besoins fonctionnels .................................................................................................................... 6
5. Conclusion ............................................................................................................................... 7

Chapitre 2 : conception............................................ ..........................................................8

1. Introduction ................................................................................................................................... 10
2. Définition langage UML et ses avantages ..................................................................................... 10
   1. Définition UML : ....................................................................................................................... 10
   2. Ses avantages ............................................................................................................................. 10
3. Diagramme en cas d’utilisation ................................................................................................. 11
   1. Cas d’utilisation ......................................................................................................................... 11
   2. Identification des acteurs ........................................................................................................... 11
   3. Diagramme en cas d’utilisation : (l’administrateur) .................................................................. 12
   4. Diagramme en cas d’utilisation : (Médecin) ............................................................................. 14
   5. Diagramme en cas d’utilisation : (patient) .................................................................................. 1
4. Diagramme de séquence .............................................................................................................. 4
   1. Diagramme de séquence d’authentification : .............................................................................. 4
   2. Diagramme de séquence d’ajout patient :.................................................................................... 5
   3. Diagramme de séquence de mettre à jour patient : ...................................................................... 6

[4) Diagramme de séquence de supprime patient : 1](#_Toc48320)

[5) Diagramme de séquence de lire patient : 2](#_Toc48321)

[V. Diagramme de classe 3](#_Toc48322)

[VI. Conclusion 4](#_Toc48323)

[Chapitre 3 : Environnement Matériel et logiciels 5](#_Toc48324)

[1) Introduction 5](#_Toc48325)

[2) Environnement matériel 5](#_Toc48326)

[3) Environnement logiciel 5](#_Toc48327)

[4) Conclusion 10](#_Toc48328)

[Chapitre 4 : Réalisation 11](#_Toc48329)

[1) Introduction 11](#_Toc48330)

[2) Création des tables de la base de données 11](#_Toc48331)

[3) Conclusion 18](#_Toc48332)

[Conclusion générale 18](#_Toc48333)

[I. Bibliographie 19](#_Toc48334)

**Introduction générale**

Le monde connaît actuellement des progrès technologiques significatifs dans tous les domaines grâce à l’informatique, qui étudie les techniques de traitement automatique de l’information. Dans le domaine de la santé, l’informatique n’est utilisée que pour enregistrer les informations du patient lors d’un rendez-vous.

Pour prendre rendez-vous, vous devez aller chez le médecin qui est un exercice fastidieux, tout d’abord, vous devez trouver le médecin pas trop loin de la maison ou du travail, et si vous êtes chanceux, vous devez encore être patient dans la salle d’attente avant de pouvoir avoir une consultation.

La gestion des rendez-vous médicaux est une composante essentielle et indispensable des activités régulières de la pratique médicale. Notre mission est donc de concevoir et de mettre en œuvre un site Web dynamique pour gérer les horaires en tenant compte des limites qui peuvent survenir lorsqu’un patient fait une réservation.

Grâce à notre site, il est possible de fixer un rendez-vous selon les critères souhaités par le patient (lieux, temps et spécialisation) et pour les médecins de gérer leurs rendez-vous.

Le but de notre projet est de faciliter la tâche au patient et au médecin, ainsi que d’économiser du temps et de la distance.

Dans ce contexte, le projet réalisé sur quatre chapitres dont on parle :

* Le première chapitre intitulé « **Analyse et spécification des besoins »** est présente la problématique, la solution proposée, les besoins non fonctionnels et besoin fonctionnels.
* Le deuxième chapitre intitulé « **Conception Merise** » est destiné la modélisation des différents diagrammes UML (cas d’utilisation, diagramme des classes, diagrammes de séquences...).
* Le troisième chapitre intitulé « **Environnement Matériel et logiciels** » vise la description de l’environnement matériel et logiciel que nous avons utilisé pour développer notre site web.
* La dernière chapitre intitulé « **Réalisation** » où nous présenterons les tables de base de données.
* Nous terminons par une conclusion générale qui synthétisera le travail effectué et les perspectives futures à envisager.

Chapitre 1 :

Analyse et spécification des besoins

#### I. Introduction

Pour organiser correctement notre projet, la phase de spécification est une partie très importante avant de développer notre site. Dans ce chapitre, on entame l'analyse et la spécification des besoins. On commencer par le problématique. Ensuite, on propose solution de ce problème. Ce dernier on donne les besoins non fonctionnels et besoin fonctionnels pour les patients.

**II. Etude de l’existant :**

Dans cette partie, nous présentons une étude de la situation existante en décrivant

Les premières applications actuelles disponible sur l’internet pour le rendez-vous médical.

Puis nous décrivons la solution proposée et la méthodologie adoptée.

Analyse de l’existant :

1. **Doctolib** : Doctolib est une plateforme de prise de rendez-vous médicaux en ligne très répandue en Europe. Elle permet aux patients de rechercher des médecins et de prendre des rendez-vous directement sur le site.



Figure 1 : Page d’accueil Doctolib

1. **Zocdoc** : Zocdoc est une plateforme similaire qui facilite la recherche de professionnels de la santé et la prise de rendez-vous aux États-Unis. Elle propose également des fonctionnalités de filtrage par spécialité, assurance, emplacement, etc.

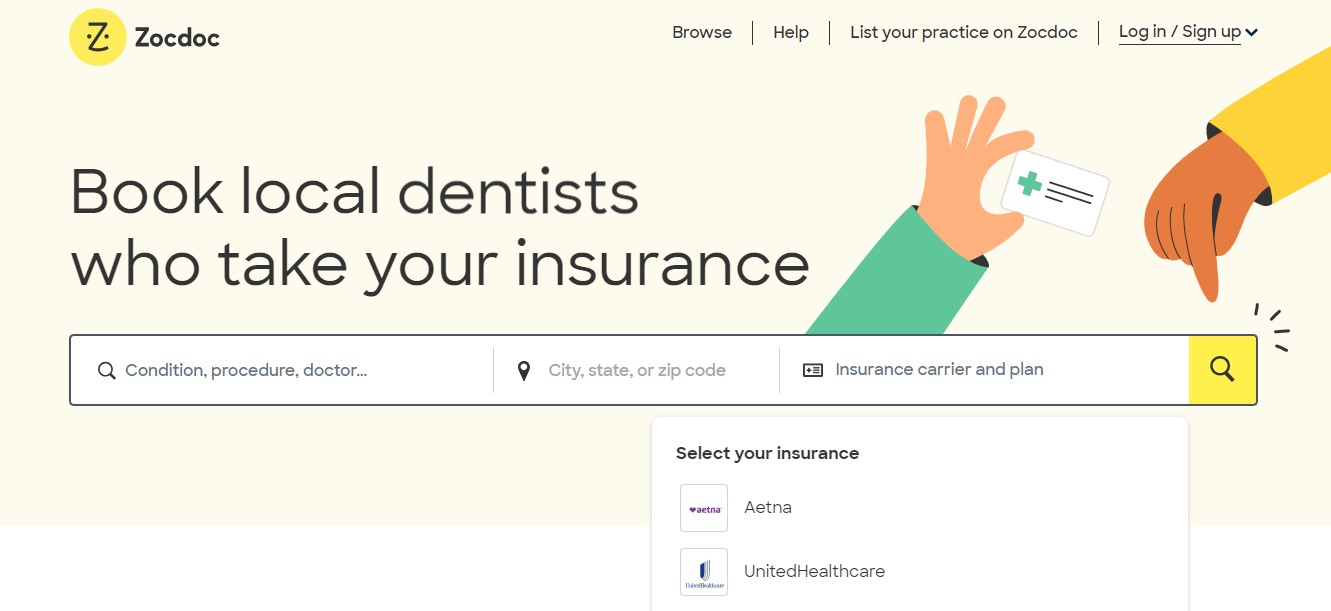


Figure 2 : Page d’accueil ZocDoc

1. **Practo** : Practo est une plateforme mondiale qui offre des services de recherche de professionnels de la santé et de réservation de rendez-vous en ligne. Elle est utilisée dans plusieurs pays.

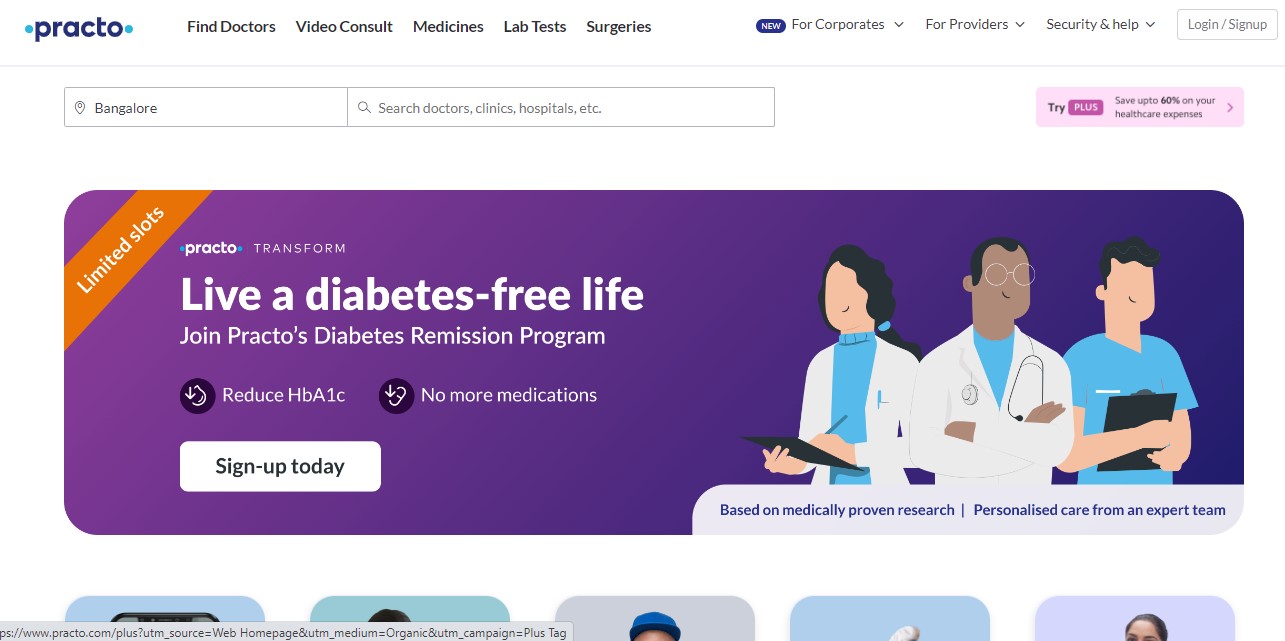


Figure 3 : Page d’accueil Practo

1. **Healthgrades** : Healthgrades permet aux patients de rechercher des médecins, de consulter des évaluations et des avis, et de prendre des rendez-vous en ligne.

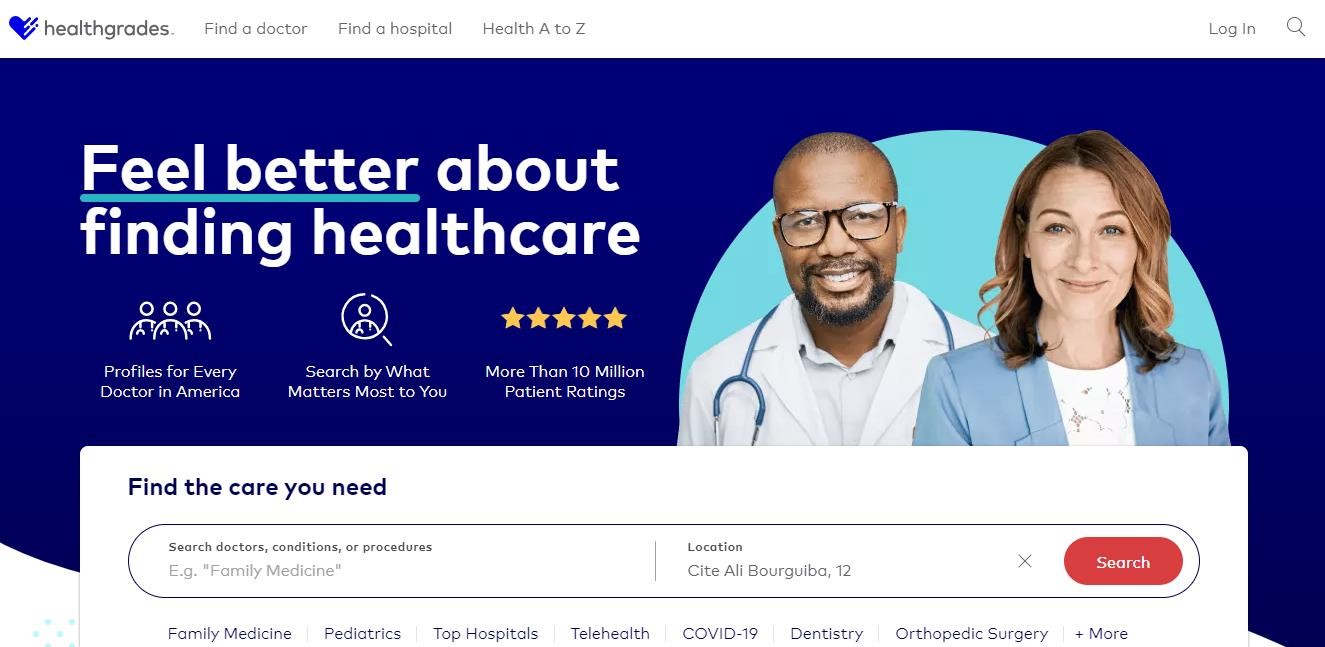


Figure 4 : Page d’accueil Healthgrades

#### III. Problématique

Le secteur médical d’aujourd’hui ne dispose pas de systèmes efficaces pour gérer la planification des rendez-vous, ce qui entraîne un manque de respect et une surveillance efficace des horaires de chaque patient.

De plus, les appels téléphoniques ne sont pas toujours critiques parce qu’ils ne sont pas facilement accessibles à un centre de santé ou à un médecin, et les appels ne sont reçus que pendant les heures de travail.

Aussi, dans le domaine de la santé également, il est difficile pour les travailleurs de la santé de recueillir les dossiers médicaux des patients à temps, surtout lorsqu’ils ont des dossiers depuis plusieurs années à la clinique.

1. **Solution Proposé**

Le projet vise à améliorer la gestion des rendez-vous dans les établissements de santé. Cela permettra la réservation, le report ou l’annulation d’un rendez-vous par chaque patient 24/24 7/7. Il s’agira de limiter le flux d’appels dans les établissements de santé ainsi que le nombre de rendez-vous non comblés.

Ce site permet également au patient de choisir le jour, le médecin et l’heure appropriée, en cas d’urgence, il peut contacter une ambulance à tout moment que disponible.

Le projet vise également à faciliter la gestion des dossiers médicaux. Cela permettra au médecin par l’intermédiaire du site Web de surveiller les rendez-vous et de consulter les dossiers médicaux de chaque patient.

1. **Les Besoins non fonctionnels et besoins fonctionnels**

#### 1) Besoins fonctionnels

Les besoins fonctionnels représentent les actions que le système doit exécuter, Notre site web doit satisfaire les besoins fonctionnels suivantes :

* **Patient :**
  + - Inscription : dans cette partie permet le patient que créé un compte dans notre site.
    - S’authentifier : le patient possède un login et un mot de passe spécifique qui lui permet de vérifier son identité.
    - Gérer les informations de compte : le patient peut modifier ses informations de son compte.
    - Accès à ambulance : dans notre site, le patient peut vérifier la disponibilité d’une ambulance en cas urgent.
    - Consulter la liste des Médecins : Le patient peut voir la liste de médecins et leurs horaires de travail.
    - Prendre rendez-vous

* **Médecin :**
  + S’authentifier : possède un login et un mot de passe spécifique qui lui permet de vérifier son identité.
  + Gérer les informations de compte : le médecin peut modifier ses informations de son compte.
  + Consulter l’historique de rendez-vous : le médecin peut voir les heures et l’historique de rendez-vous.
  + Gérer Fiche patient : dans cette partie permet le médecin qui créé, lire, mettre à Jour et supprimer la fiche patiente.

* **L’administrateur :** 
  + S’authentifier : possède un login et un mot de passe spécifique qui lui permet de vérifier son identité.
  + Gérer le patient : qui lire, modifier et supprimer l’information de patient. Aussi bien il peut ajouter et supprimer son compte.
  + Gérer le médecin : l’administrateur peut lire, modifier et supprimer l’information de médecin, encore il peut ajouter et supprimer son compte.
  + Gérer le rendez-vous : dans cette partie permet l’administrateur qui lire le rendez-vous.
  + Gérer l’ambulance : l’administrateur peut ajouter et mettre à jour l’ambulance.

#### 2) Les besoins non fonctionnels

Les besoins non fonctionnels sont importants car ils agissent de façon indirecte sur le résultat et sur le rendement de l’utilisateur, ce qui fait qu’ils ne doivent pas être négligés, pour cela il faut répondre aux exigences suivantes :

* **Fiabilité** : Le site doit fonctionner en continu sans erreurs et doit être satisfaisant.
* Erreurs : Les ambiguïtés doivent être signalées par des messages d’erreur bien structurés pour guider et familiariser l’utilisateur avec notre site.
* **Sécurité** : Notre solution doit avant tout respecter la confidentialité des données personnelles des patients, ce qui reste l’une des restrictions les plus importantes sur les sites.
* **Compatibilité et portabilité** : Un site Web peut être invoqué quel que soit son domaine, modifié et langage de programmation seulement avec la compatibilité avec tous les navigateurs Web et tous les moyens, qu’il soit PC, IPAD ou Mobile...
* **Performance** : décrivent les performances d’exécution du site, généralement en termes de temps de réponse.
* **Maintenabilité** : Le site doit assurer une maintenance et une amélioration faciles. Ils doivent être divisés en couches pour assurer la maintenabilité.
* **Convivialité** : le site doit présenter une interface compréhensible, facile à manipuler, ce qui nous permettra d’accroître la rentabilité.
* **Efficacité** : Le site doit permettre une utilisation minimale des ressources : mémoire, processeur, etc.

#### VI. Conclusion

Dans ce premier chapitre, nous avons présenté l’analyse et la spécification des besoins permettant de concevoir et de développer un site web qui facilitera la gestion des rendez-vous médicaux.

Chapitre 2 : conception

#### I. Introduction

Dans ce chapitre, nous commencerons par définir UML (Unified Modeling Language) et ses avantages, puis tout nous examine la partie conception du projet, qui représente une étape importante pour la réflexion dans le cycle de développement du programme, où nous élaborons sur les différents éléments de conception, pour savoir le diagramme de classe, le diagramme de cas utilisateur et diagramme de séquence nous avons choisi d’utiliser Star UML car il est performant et facile à manipuler.

#### II. Définition langage UML et ses avantages

**1) Définition 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 leur structure 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.

Il ressemble aux plans utilisés dans d'autres domaines et se compose de différents types de diagrammes. En générale, les diagrammes UML décrivent la limite, la structure et le comportement du système et des objets qui s'y trouvent.

UML n’est pas un langage de programmation, mais il existe des outils qui peuvent être utilisés pour créer un code de plusieurs langages à partir de diagrammes UML. L'UML a une relation directe avec l’analyse et la conception orientée objet.

Notre choix s’est porté sur le langage UML pour ses nombreux avantages.

**2) Ses avantages**

* Adopté par les grandes entreprises.
* Facile à comprendre.
* Universel.
* Adopté par plusieurs processus de développement.
* Limite les risques d’erreur.
* Non limité au domaine informatique.

#### III. Diagramme en cas d’utilisation

Le Diagramme de cas d’utilisation est le premier diagramme du modèle UML utilisé pour la modélisation des besoins des utilisateurs.

1. **Cas d’utilisation**

Le cas d’utilisation représente une interaction entre le système informatique à développer et l’acteur acteur. Plus précisément, le cas d’utilisation décrit une séquence d’actions que le système réalise par ses fonctions afin de structurer les besoins des utilisateurs produisant des résultats observables par l’acteur.

1. **Identification des acteurs**

Un acteur représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié. Un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages éventuellement porteurs de données. En effet, nous distinguons trois acteurs dans notre système :

|  |  |
| --- | --- |
| **Acteur** | **Rôle** |
| ***L’administrateur*** | Cette personne responsable à gérer toutes les fonctionnalités de notre site. |
| ***Médecin*** | Il gère la fiche de patient et consulter les historiques des rendez-vous. |
| ***Patient*** | Prendre rendez-vous pour une consultation. |

***Tableau 1*** : *Identification des acteurs de Notre site*

1. **Diagramme en cas d’utilisation : (l’administrateur)**

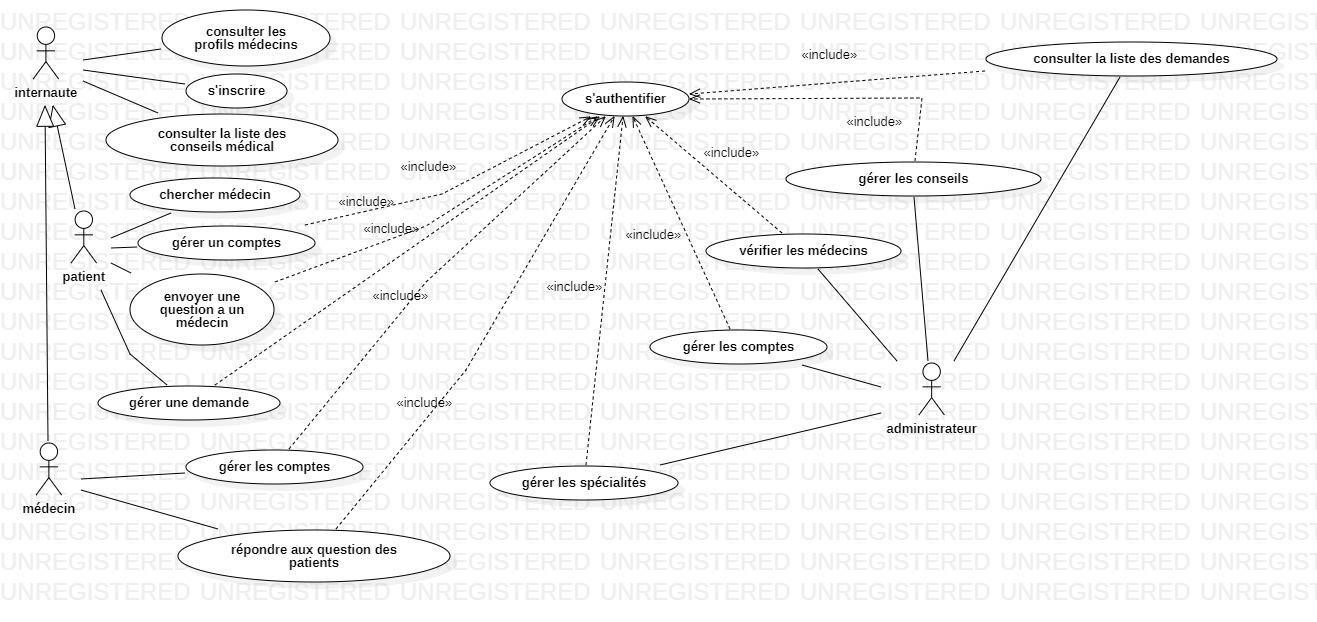
Dans cette partie, nous avons identifié les cas d’utilisation significatifs du Système.

Les cas d’utilisation décrivent les taches fonctionnelles effectuées à travers la plateforme

Le diagramme de cas d’utilisation est un diagramme comportemental qui permet d’identifier les potentialités d’interaction entre le système et les acteurs, c’est-à-dire, toutes les fonctionnalités que doit fournir le système.

* 1. **Diagramme de cas d’utilisation global :**

Le système assure pour les utilisateurs diverses fonctions mises en valeur à travers le diagramme de cas d’utilisation illustré par la figure ci-dessous :

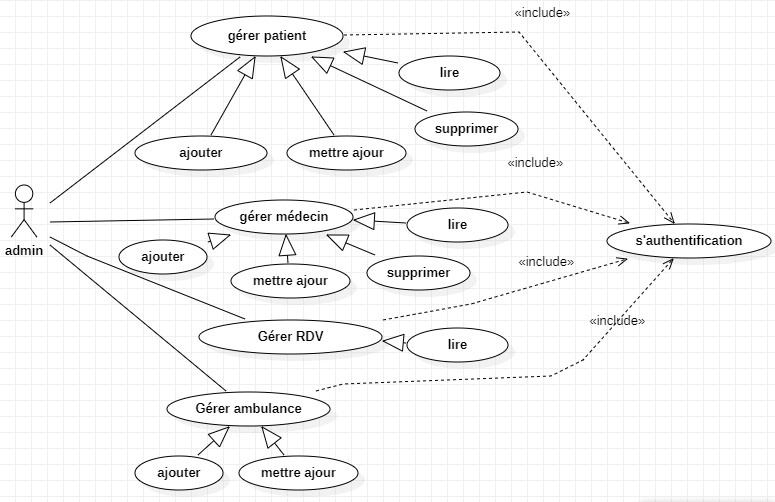


***Figure 5 : diagramme de cas d’utilisation global***

* 1. **Raffinement du diagramme de cas d’utilisation coté administrateur :**

L’administrateur peut gérer le deux acteur (patient et médecin) qu’ajouter, supprimer le patient, lire et modifier ses informations.

Aussi bien l’administrateur peut lire le rendez-vous et gérer l’ambulance.



***Figure 6*** *: diagramme en cas d’utilisation de l’administrateur*

*• Description textuelle du sous cas d’utilisation « Consulter la liste des patients »*

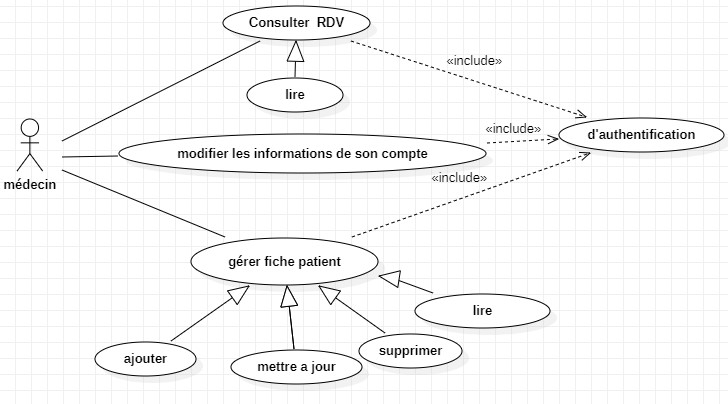
|  |  |
| --- | --- |
| **Cas d’utilisation** | Consulter liste patients |
| **Acteur** | Administrateur |
| **Pré-condition** | Administrateur authentifié |
| **Post-condition** | Liste des comptes affichée avec succès. |
| **Scénario principal** | 1. l’administrateur clique sur consulter comptes. 2. Le système affiche la liste des compte existants. |

Tableau 1 : Description textuelle du sous cas d’utilisation « Consulter la liste des patients »

1. **Diagramme en cas d’utilisation : (Médecin)**

Dans cette cadre de cas d’utilisation de médecin peut lire le rendez-vous et modifier les informations de son compte.

De plus le médecin peut lire et/ou le fiche patient. En outre, il peut changer les informations de patient et cette caractéristique seulement avec un médecin, il a aussi est le seul qui a le droit de supprimer la fiche de patient.



***Figure 7*** *: diagramme en cas d’utilisation de Médecin*

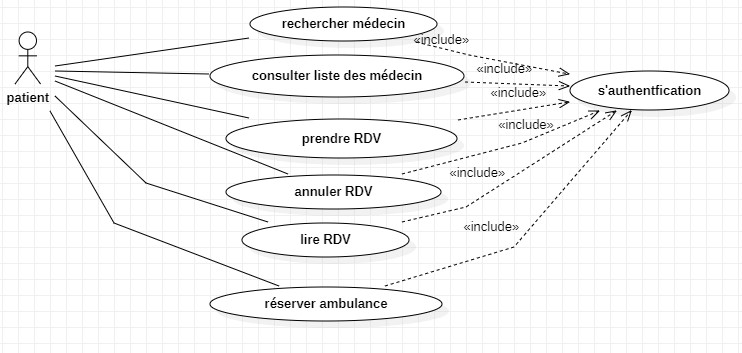
* + - *Description textuelle du sous cas d’utilisation « Modifier profil »*

|  |  |
| --- | --- |
| Titre | Modifier profil |
| Acteur | Médecin |
| But | Cette partie doit offrir aux médecins des interface permettant de modifier leur profil |
| Pré-condition | Médecin authentifié |
| Scénario nominal | 1. Le médecin choisit la rubrique modifié informations 2. Le système retourne la page de modifications 3. Le médecin modifie les informations 4. Le médecin valide 5. Le système affiche le message de succès |
| Enchainementsalternatives | A1 : données incorrectes  A2 : champs vides |

* + - Tableau 2 : *Description textuelle du sous cas d’utilisation « Modifier profil »*

1. **Diagramme en cas d’utilisation : (patient)**

Le patient est un utilisateur du site web, afin d’accéder aux différentes fonctions du site, le patient doit se connecter s’il a déjà un compte, sinon il doit créer un compte. Le patient peut rechercher un médecin directement à partir de la page d’accueil où chaque médecin enregistré dans le système. Une fois le médecin choisi, le patient peut prendre rendez-vous selon sa disponibilité.



***Figure 8*** *: diagramme en cas d’utilisation de patient*

* *Description textuelle du sous cas d’utilisation « demander consultation »*

|  |  |
| --- | --- |
| **Cas d’utilisation** | Demander consultation |
| **Acteur** | Patient |
| **Pré-condition** | Patient authentifié |
| **Post-condition** | Demande envoyée |
| **Scénario principal** | 1. L’utilisateur choisit le médecin à demander. 2. utilisateur clique sur bouton Contacter.   3-le système affiche le formulaire de contact avec le médecin   * 1. patient remplit formulaire de contact.   2. Utilisateur clique sur bouton Envoyer.   3. Le système affiche un message informant l’utilisateur que la demande a été envoyer avec succès. |
| **Scénario alternatif** | Le système affiche une erreur de connexion à la base des données. |

Tableau 3 : Description textuelle du sous cas d’utilisation « demander consultation »

* *Description textuelle du sous cas d’utilisation « consulter détails profil »*

|  |  |
| --- | --- |
| **Cas d’utilisation :** | Consulter les détails des comptes |
| **Acteur :** | Client |
| **Pré Condition :** | Client authentifié |
| **Post Condition :** | Les informations de compte affichée |
| **Scénario nominal :** | 1. Le client demande la consultation de ses informations 2. Le système lui affiche les informations de comptes |

Tableau 4 : Description textuelle du sous cas d’utilisation « consulter détails profil »

* *Description textuelle du sous cas d’utilisation « consulter ses demande »*

|  |  |
| --- | --- |
| **Cas d’utilisation** | Consulter demandes |
| **Acteur** | Patient |
| **Pré-condition** | Patient authentifié |
| **Post-condition** | Liste de ses demandes affichées avec succès |
| **Scénario principal** | 1. le Patient clique sur consulter demandes 2. Le système affiche la liste de ses demandes |
| **Scénario alternatif** | Le système affiche une erreur de connexion à la base des données. |

Tableau 5 : Description textuelle du sous cas d’utilisation « consulter ses demande »

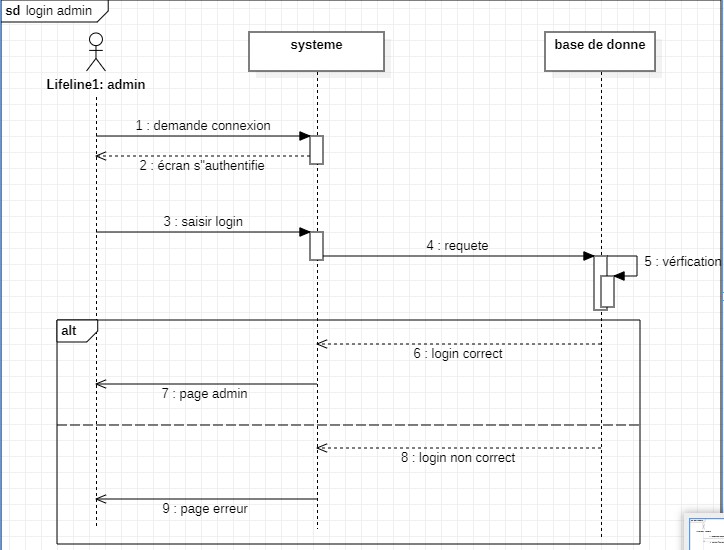
#### IV. Diagramme de séquence

Le diagramme de séquence suit le diagramme de cas d’utilisation car il le complète. Un diagramme de séquence représentant la séquence des messages entre les objets pendant l’interaction.

Le diagramme séquence consiste en un groupe d’objets, représentés par des lignes de vie et les messages que ces objets échangent lors de l'interaction.

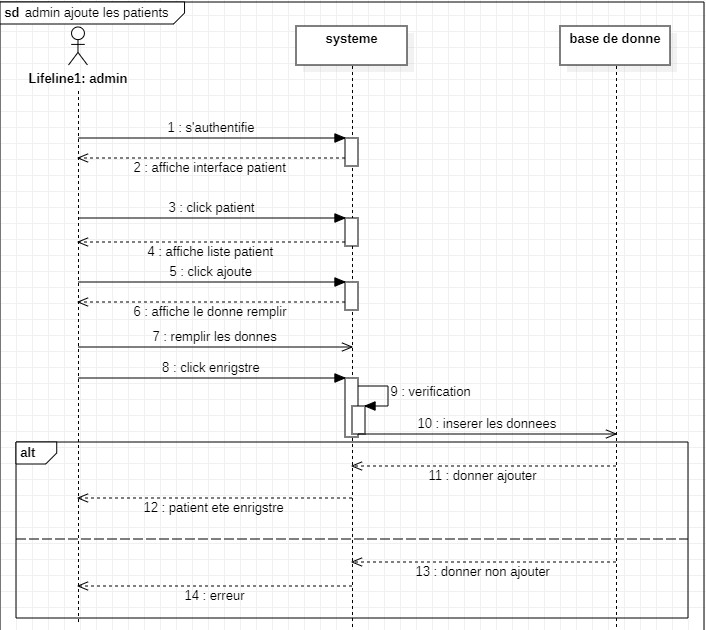
Dans cette partie on va voire quelque diagramme de séquence d’administrateur.

1. **Diagramme de séquence d’authentification :**



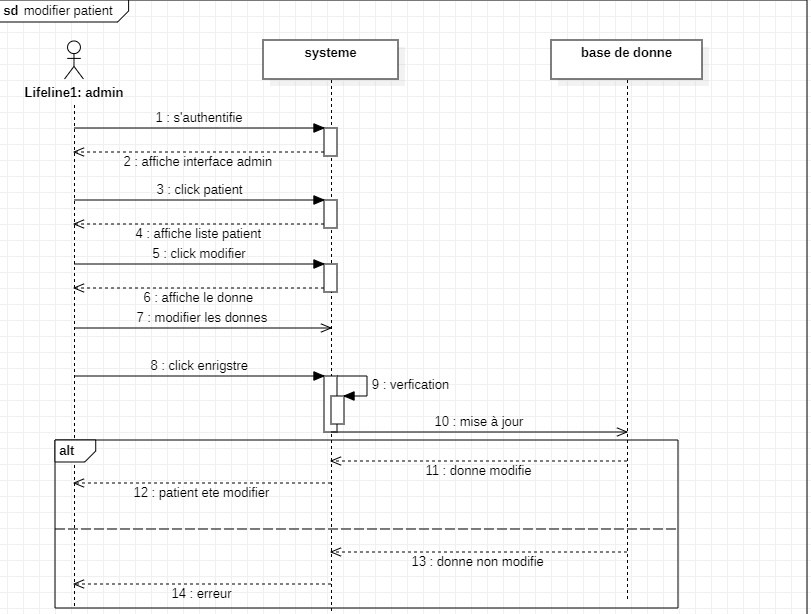
***Figure 9*** *: Diagramme de séquence d’authentification*

1. **Diagramme de séquence d’ajout patient :**



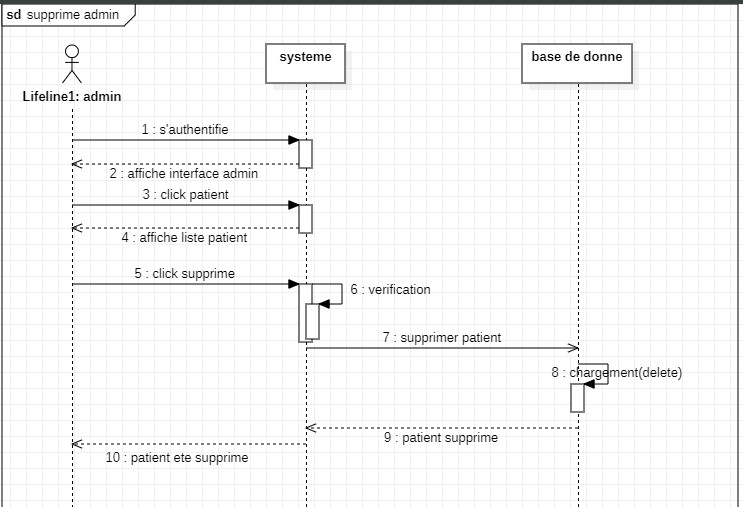
***Figure 10*** *: Diagramme de séquence d’ajout patient*

1. **Diagramme de séquence de mettre à jour patient :**

:

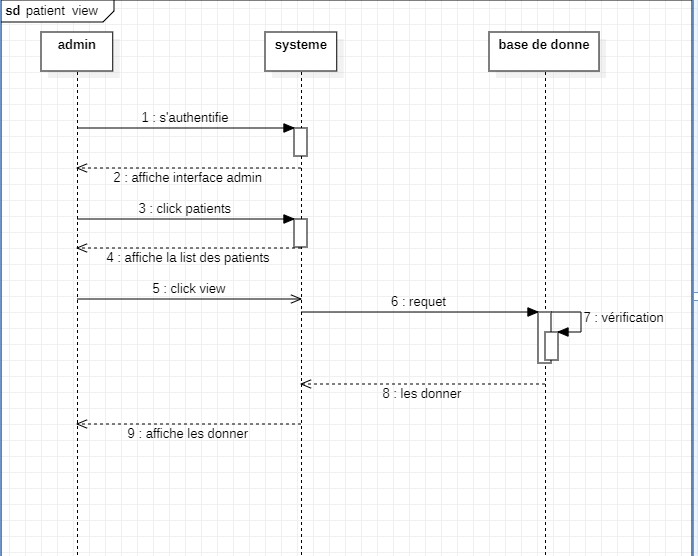
***Figure 11*** *: Diagramme de séquence de mettre à jour patient*

## 4) Diagramme de séquence de supprime patient :



***Figure 12*** *: Diagramme de séquence de supprime patient*

## 5) Diagramme de séquence de lire patient :



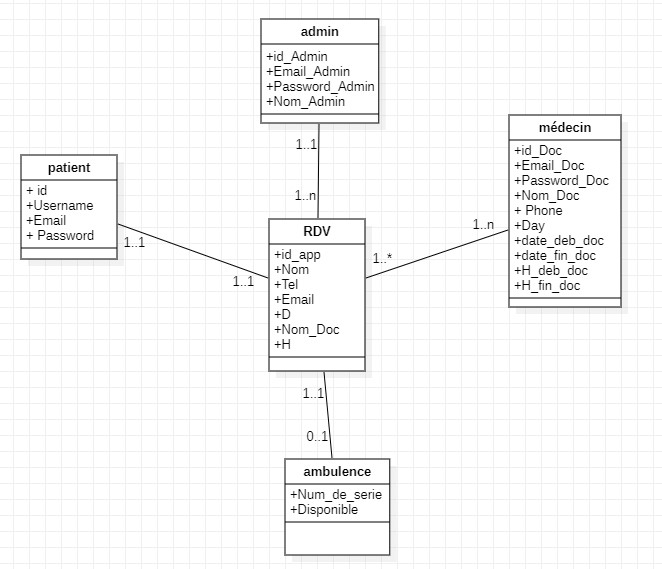
***Figure 13*** *: Diagramme de séquence de lire patient*

# V. Diagramme de classe

Le diagramme de classe montre la structure interne d’un système. 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.

Une classe décrit les responsabilités, le comportement et le type d’un ensemble d’objets. Les éléments de cet ensemble sont les instances de la classe.



***Figure 14*** *: diagramme de classe*

## VI. Conclusion

Dans ce chapitre, nous avons pu concevoir un site de gestion des rendez-vous médicaux en se basant sur les diagrammes du langage UML à savoir le diagramme de cas d'utilisation, le diagramme de séquences et le diagramme de classes.

# Chapitre 3 : Environnement Matériel et logiciels

### I. Introduction

Ce chapitre c’est la phase de réalisation de ce site web dynamique. Ce chapitre est composé de 2 parties : la première partie présente les ressources matérielles, la deuxième partie présente les logicielles utilisées pour la réalisation du projet.

### II. Environnement matériel

* **LAPTOP :** HP
* **Processeur** : 11th Gen Intel(R) Core (TM) i3-1115G4 @ 3.00GHz 3.00 GHz
* **Mémoire** : 8,00 Go
* **Disque Dur** : 250 Go
* **Système** : Windows 11

### III. Environnement logiciel

Lors du développement de ce site, nous avons utilisé, les outils logiciels et les langages de programmations suivants :

**3.2.2.1 Éditeur de code**

**Visual Studio Code :** C’est un éditeur de code extensible, open source et gratuit, développé par Microsoft pour Windows, Linux et MacOs. Côté fonctionnalités, VS Code se révèle bien pratique sur de nombreux points, incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et l’intégration de Git.



Figure 15 – Logo de Visual Studio Code

**3.2.2.2 Outils de conception**

Le STARUML : StarUML est un outil de génie logiciel Dédié à la modélisation UML et édité de la société coréenne MKLabs. Il est multiplateforme et fonctionne sous Windows, Linux et MacOS. La dernière version gère l'ensemble des diagrammes définis par UML 2, ainsi que plusieurs diagrammes.



Figure 16 – Logo de star uml

**3.2.2.3 Outil de rédaction du rapport**

Microsoft Word : Word est le logiciel phare de la suite Bureautique Microsoft Office. C'est l'un des logiciels les plus utilisés dans le monde et permet de rédiger des lettres, CV, rapports et tous types de documents texte. Word permet de mettre en page vos documents, de générer des sommaires, de numéroter automatiquement des pages, corriger la grammaire et l'orthographe, de créer des schémas, de faire du publipostage



Figure 17 – Logo de Microsoft Word

**3.2.2.4 Outil de création de base de données**

WampServer : WampServer est une plateforme de développement web sous Windows pour les applications web dynamiques à l’aide du serveur Apache2, du langage de scripts PHP et d’une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement vos bases de données.



Figure 18 : Logo de WampServer

**3.2.2.5 Langages utilisées**

JavaScript : JavaScript, souvent abrégé en « JS », est un langage de script léger, orienté objet, principalement connu comme le langage de script des pages web. Mais il est aussi utilisé dans de nombreux environnements extérieurs aux navigateurs web tels que Node js.



Figure 19 – Logo de JAVA Script

HTML : C’est un langage de description (dit de marquage) de pages Web. Il est utilisé afin de créer et de représenter le contenu d’une page web (texte, liens, images, Script.) et sa structure.



Figure 20 – Logo de HTML

CSS : C’est un langage de style dont la syntaxe est extrêmement simple mais son rendement est remarquable. En effet, le CSS (Feuilles de style en cascade) s’intéresse à la mise en forme du contenu intégré avec du HTML.



Figure 21 – Logo de CSS

PHP : PHP (acronyme récursif de PHP : HyperText Preprocessor) est un sous-ensemble de langages de script généraliste et Open Source spécialement conçu pour le développement d'applications web. Il peut être intégré facilement au HTML comme JavaScript et Python. PHP est principalement conçu pour servir de langage de script coté serveur, ce qui fait qu'il est capable de réaliser tout ce qu’un script CGI quelconque peut faire, comme collecter des données de formulaire, générer du contenu dynamique, ou gérer des cookies, mais PHP peut en faire bien plus.



Figure 22 – Logo de php

Canvas :

Canvas est une plateforme australienne, utilisé pour créer des graphiques de réseaux sociaux, présentations, annonce publicitaire, documents, logo, et des autres contenus visuels. Il contient aussi des templates.



Figure 23 – Logo de cana

3.2.2.6 Frameworks et Bibliothèque utilisés :

Bootstrap :

Bootstrap est un framework de développement Web gratuit et open-source. Il est conçu pour faciliter le processus de développement Web de sites Web réactifs et mobiles en fournissant un ensemble de syntaxes pour les conceptions de modèles. En d'autres termes, Bootstrap aide les développeurs Web à créer des sites Web plus rapidement car ils n'ont pas à se soucier des commandes et des fonctions de base. Il se compose de scripts HTML, CSS et JS pour diverses fonctions et composants liés à la conception Web.

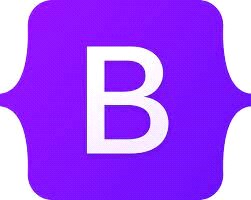


Figure 24 – Logo de BOOTSTRAP

### IV. Conclusion

Au cours de ce chapitre, nous avons décrit les ressources matérielles et logicielles sur lesquelles nous avons développé notre site.

# Chapitre 4 : Réalisation

### I. Introduction

Dans ce dernier chapitre, nous verrons les tables de la base de données qu’utilisé pour stocker et partager les informations de notre site.

### II. Création des tables de la base de données

Les tables sont des fichiers contenant des données relatives à une certaine catégorie d'information.

1. **Table d’administrateur :**

**Table admin :** (id\_Admin ; Email\_Admin ; Password\_Admin ; Nom\_Admin ; rôle)

|  |  |  |
| --- | --- | --- |
|  | **Nom** | **Type** |
| **1** | id\_Admin | Int(10) |
| **2** | Email\_Admin | Varchar (30) |
| **3** | Password\_Admin | Varchar (20) |
| **4** | Nom\_Admin | Varchar (20) |
| **5** | Rôle | Varchar (10) |

***Tableau 2*** *: table d’administrateur*

1. **Table d’appointement :**

**Table d’appointement** : (id\_app ; Nom ; Tel ; Email ; D ; Nom\_Doc ; H)

|  |  |  |
| --- | --- | --- |
|  | **Nom** | **Type** |
| **1** | id\_App | Int (11) |
| **2** | Nom | Varchar (20) |
| **3** | Tel | Int (8) |
| **4** | Email | Varchar (30) |
| **5** | D | Date |
| **6** | Nom\_Doc | Varchar (20) |
| **7** | H | Time |

***Tableau 3*** *: table d’appointement*

1. **Table d’ambulance :**

**Table d’ambulance** : (id\_aumbulance ; Disponible)

|  |  |  |
| --- | --- | --- |
|  | **Nom** | **Type** |
| **1** | id\_aumbulance | int(11) |
| **2** | Disponible | tinyint(1) |

***Tableau 4*** *: table d’ambulance*

1. **Table de médecin :**

**Table de médecin** : (id\_Doc ; Email\_Doc ; Password\_Doc ; Nom\_Doc ; Phone ; date\_deb\_doc ; date\_fin\_doc ; H\_deb\_doc ; H\_fin\_doc ; rôle)

|  |  |  |
| --- | --- | --- |
|  | **Nom** | **Type** |
| **1** | id\_Doc | int(8) |
| **2** | Email\_Doc | varchar(30) |
| **3** | Password\_Doc | varchar(20) |
| **4** | Nom\_Doc | varchar(20) |
| **5** | Phone | int(8) |
| **6** | date\_deb\_doc | Date |
| **7** | date\_fin\_doc | Date |
| **8** | H\_deb\_doc | Time |
| **9** | H\_fin\_doc | Time |
| **10** | rôle | varchar(10) |

***Tableau 5*** *: table de médecin*

1. **Table de patient :**

**Table de patient** : (id ; Username ; Email ; Password ; Rôle)

|  |  |  |
| --- | --- | --- |
|  | **Nom** | **Type** |
| **1** | id | int(11) |
| **2** | Username | varchar(20) |
| **3** | Email | varchar(30) |
| **4** | Password | varchar(20) |
| **5** | Rôle | varchar(10) |

***Tableau 6*** *: table de patient*

1. **Table de rendez-vous :**

Table de rendez-vous : (Cin ; nom\_prenom ; sexe ; num\_tel ; date\_de\_naissance ; taille ; poids ; commentaire)

|  |  |  |
| --- | --- | --- |
|  | **Nom** | **Type** |
| **1** | Cin | int(8) |
| **2** | nom\_prenom | varchar(30) |
| **3** | sexe | enum ('H', 'F') |
| **4** | num\_tel | int(8) |
| **5** | date\_de\_naissance | date |
| **6** | taille | float |
| **7** | poids | float |
| **8** | commentaire | varchar(100) |

***Tableau 7*** *: table de rendez-vous*

**3.3 : Aperçu sur le travail réalisé de l’application web**

Cette partie dénombre la présentation des Scénarios applicatifs de l’application. Nous allons présenter dans ce qui suit, les imprimes-écran des principales interfaces réalisées dans notre site web. **Page d’accueil :**

La page d'accueil d'un site web est l'une des pages les plus importantes, car elle crée la première impression pour les visiteurs et les oriente vers le reste du contenu du site.



Figure 25 : Page d’accueil

**Page « Liste des medecins »**

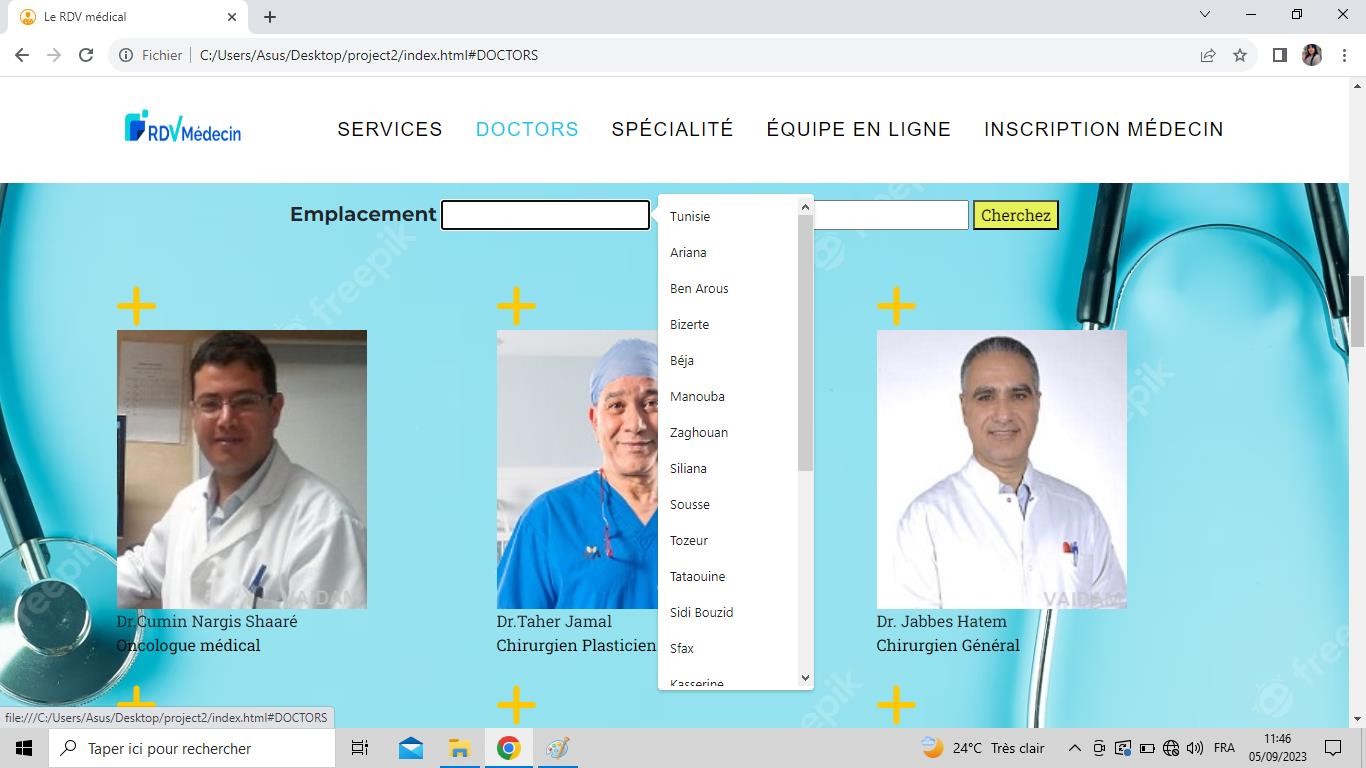


Figure 26 : Liste des médecins

Une page "Liste des médecins" sur un site web peut être une ressource précieuse pour les utilisateurs cherchant à trouver des professionnels de la santé.

Les utilisateurs peuvent utiliser cette page pour rechercher des médecins dans leur région ou dans une spécialité particulière.

**Page d’inscription pour un médecin :**

Une page d'inscription sur un site web est une page essentielle qui permet aux utilisateurs de créer un compte ou de s'inscrire sur la plateforme. Voici ce que vous pourriez inclure sur une page d'inscription typique :

1. Formulaire d'Inscription :
   * Nom complet
   * Adresse e-mail
   * Mot de passe (avec des exigences de sécurité, telles que la longueur minimale et la complexité)
   * Confirmation du mot de passe
2. Options Supplémentaires :
   * Numéro de téléphone (facultatif)
   * Date de naissance (facultatif)
   * Genre (facultatif)
   * Autres informations pertinentes en fonction du type de site (par exemple, adresse, profession, etc.)
3. Bouton d'Inscription :
   * Un bouton "S'Inscrire" ou "Créer un Compte" qui envoie les informations d'inscription au serveur pour traitement.

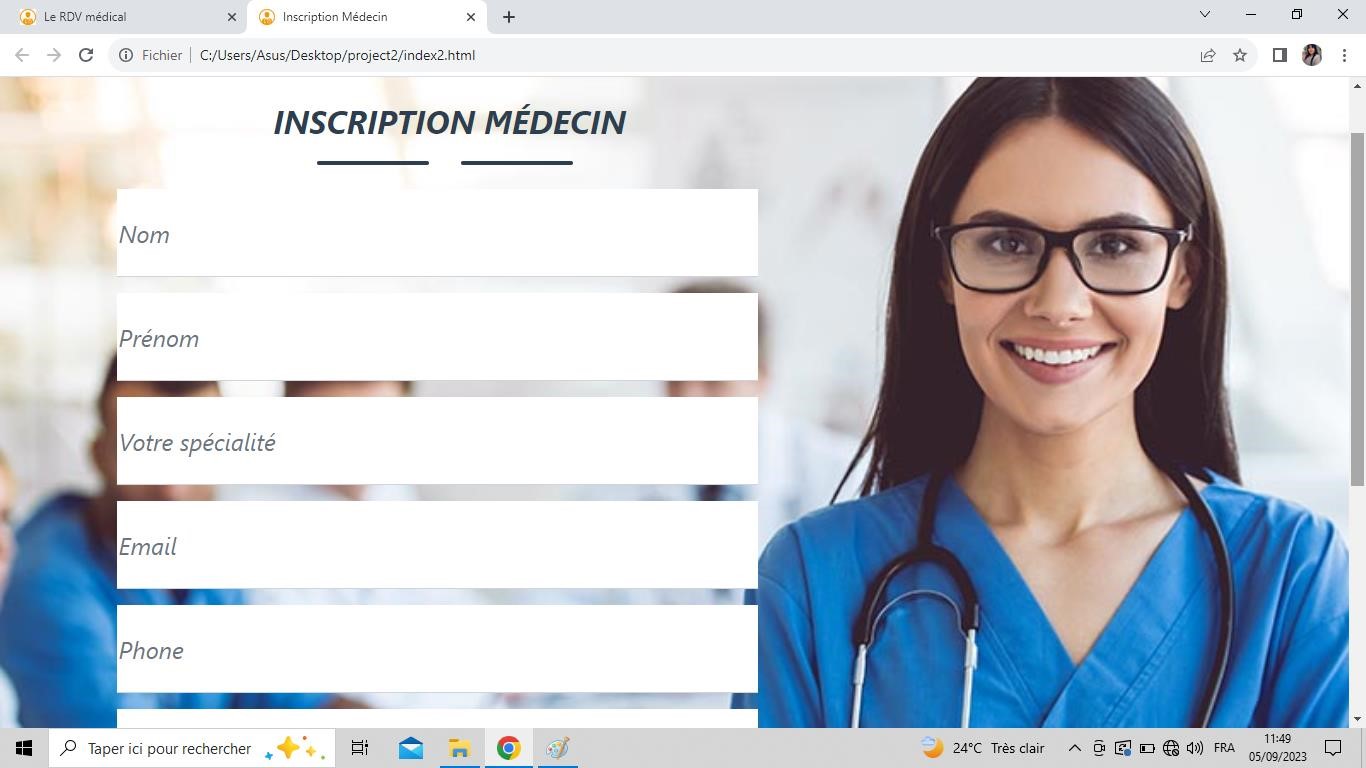


Figure 27 : Formulaire d’inscription **Formulaire de connexion :**



Figure 28 : Formulaire de connexion :

Un formulaire de connexion sur un site web permet aux utilisateurs de se connecter à leur compte existant. Voici les éléments généralement présents dans un formulaire de connexion :

1. Champ d'Adresse E-mail ou de Nom d'Utilisateur :
   * Les utilisateurs doivent entrer leur adresse e-mail ou leur nom d'utilisateur associé à leur compte.
2. Champ de Mot de Passe :
   * Les utilisateurs doivent entrer leur mot de passe associé à leur compte. Le texte du mot de passe doit généralement être masqué pour des raisons de sécurité.
3. Bouton de Connexion :
   * Un bouton "Se Connecter" qui valide les informations d'identification et connecte l'utilisateur à son compte.

**Page des services**

Une page des services sur un site web est un endroit où vous pouvez présenter en détail les services que votre site propose à ses clients ou utilisateurs.



Figure 29 : Page des services

**Page des spécialités des médecins disponible :**

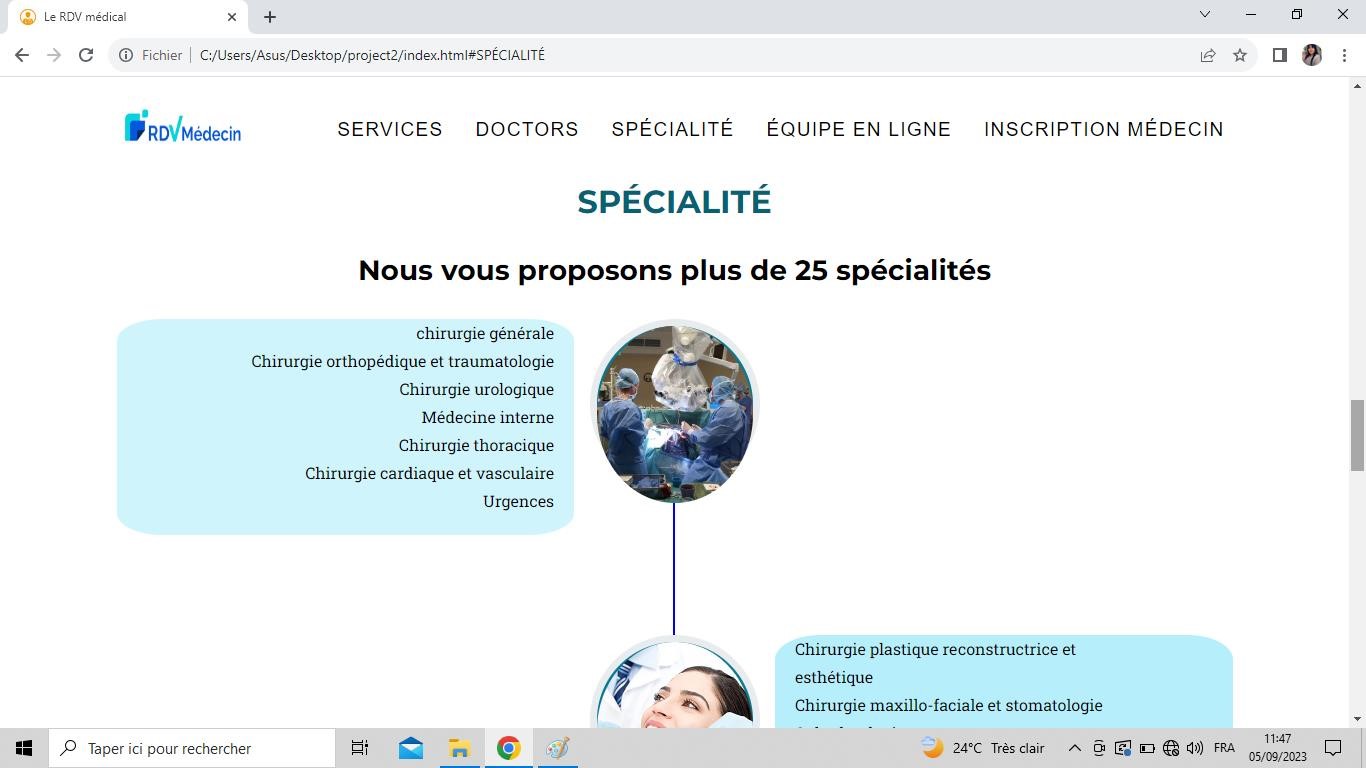


Figure 30 : Liste des spécialités

Une page des spécialités des médecins sur un site web médical permet aux utilisateurs de découvrir les différentes spécialités médicales proposées par les médecins ou les praticiens répertoriés sur le site.

### Conclusion

Dans ce chapitre, nous avons présenté les tables de base de données qui stocker les informations de notre site

# Conclusion générale

Ce rapport a présenté une analyse complète de l'ensemble du processus de développement de notre application de consultation médicale rapide. Les chapitres précédents ont abordé les domaines suivants :

Dans le Chapitre 1 : Étude de l’Existant, nous avons passé en revue les systèmes de santé traditionnels, identifié les lacunes et les défis auxquels sont confrontés les patients et les professionnels de la santé. Nous avons également examiné les précédentes solutions de consultation médicale en ligne pour comprendre les meilleures pratiques et les erreurs à éviter.

Apres au niveau de Chapitre 2 : Spécification des Besoins Nous avons défini les besoins fonctionnels et non fonctionnels de l'application en fonction des exigences des utilisateurs, des objectifs de l'entreprise et des normes de sécurité. Nous avons également identifié les fonctionnalités clés telles que la réservation de consultations, la gestion des dossiers médicaux et la communication sécurisée.

Ensuite Chapitre 3 : Conception Ce chapitre a traité la conception de l'application, y compris l'architecture logicielle, l'interface utilisateur, la sécurité des données et la convivialité. Nous avons élaboré des diagrammes de conception détaillés pour guider le développement.

Et enfin Chapitre 4 : Réalisation Dans ce chapitre, nous avons décrit le processus de développement de l'application, les langages de programmation, les bases de données utilisées et les tests effectués pour garantir le bon fonctionnement de l'application.

En résumé, ce rapport démontre notre engagement envers l'amélioration des soins de santé grâce à l'innovation technologique. L'application de consultation médicale rapide vise à répondre aux besoins des patients tout en garantissant la qualité des soins. Nous recommandons une adoption proactive de cette application, une formation continue des professionnels de la santé et une vigilance constante en matière de sécurité des données pour assurer son succès continu.

# Bibliographie

Lien de site web: <file:///C:/Users/Asus/Desktop/project2/index.html>

----------------------------------

[www.lucidchart.com](http://www.lucidchart.com/) [www.memoireonline.com](http://www.memoireonline.com/)  [www.ibm.com](http://www.ibm.com/)

----------------------------------------