



**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**  
**Université Mohamed Khider – BISKRA**

**Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie**

**Département d'informatique**

**N° d'ordre :SIOD8 /M2/2021**

## **Mémoire**

Présenté pour obtenir le diplôme de master académique en

# **Informatique**

Parcours : Système d'Information Optimisation et Décision (SIOD)

---

## **Titre**

**Réalisation et évolutions sur une application web  
pour la prise de rdv médicaux ligne**

---

**Par : kaddour Aouicha**

**Encadrer par : Dr.Guemieda Abdelbasset**

:

Année universitaire 2020-2021

## Résumé

Pour améliorer la prise en charge médicale des patients, leur relation avec les professionnels de santé doit être mieux organisée pour réduire l'écart entre l'offre et la demande des soins. En réponse aux difficultés rencontrées par les patients en Algérie relatives aux méthodes de prise de rendez-vous médicaux, ce projet vise à analyser, concevoir et développer des évolutions pour un système d'information dédié à la prise de rendez-vous médicaux en ligne. Il s'agit d'un site web dynamique, simple et facile à utiliser par les patients en Algérie pour fixer des rendez-vous médicaux en ligne et éviter ainsi les déplacements et le temps d'attente inutiles chez les professionnels de santé.

**Mots clés :** Système d'information, site web dynamique, Santé, Rendez-vous médical, patient et professionnel de santé

## Abstract

To improve the medical care of patients, their relationship with health professionals must be better organized to reduce the gap between the supply and demand for care. In response to the difficulties encountered by patients in Algeria relating to the methods of making medical appointments, this project aims to analyze, design and develop evolutions for an information system, dedicated to making medical appointments online. It is a dynamic, simple and easy-to-use website for patients in Algeria to schedule medical appointments online and thus avoid unnecessary travel and waiting time in professionals' healthcare offices.

**Keywords:** Information system, dynamic website, Health, Medical appointment, patient and health professional

## ملخص

لتحسين الرعاية الطبية للمرضى، يجب أن تكون علاقتهم بالمهنيين الصحيين منظمة بشكل أفضل لتقليل الفجوة بين العرض والطلب على الرعاية. استجابة للصعوبات التي يواجهها المرضى في الجزائر فيما يتعلق بأساليب تحديد المواعيد

الطبية، يهدف هذا المشروع إلى تحليل وتصميم وتطوير نظام معلومات يمكن الوصول إليه عبر الإنترنت لتحديد المواعيد الطبية عبر الإنترنت. إنه موقع ديناميكي وبسيط وسهل الاستخدام للمرضى في الجزائر لحجز المواعيد الطبية عبر الإنترنت وبالتالي تجنب السفر غير الضروري ووقت الانتظار لأخصائيي الرعاية الصحية.

الكلمات الدلالية: نظام معلومات ، موقع ديناميكي ، صحة،موعد طبي ، مريض ، المهنيين الصحيين

## *Remerciement*

*Aux termes de ce mémoire, nous tenons à exprimer mes*

*Remerciements au bon « DIEU » qui m'a entouré de sa bienveillance et m'a renforcé avec le courage et la force pour pouvoir mener à bien ce travail.*

*Que mes sincères et profonds remerciements aillent à Monsieur **Goumaïda abd elbasset** pour l'honneur qu'il m'a fait en acceptant d'être mon encadrant, pour l'intérêt qu'il a porté à ce travail et pour ses précieux conseils tout au long de l'élaboration de ce projet ; qu'il soit rassuré de mes reconnaissances et mon respect indéfectible.*

*A tous mes enseignants.*

*Au président et aux membres du jury qui ont accepté*

*De juger mon travail.*

## *Dédicace*

*Je dédie ce mémoire d'u master :*

*A*

*Ma très chère mère et mon très cher père.*

*En témoignage de ma reconnaissance envers le soutien,  
les sacrifices et tous les efforts qu'ils ont fait pour  
mon éducation ainsi que ma formation*

*A*

*Mes chères frères "Mohammed" et "Oussama"*

*Mon fiancé « Younes »*

*Mes chères sœurs "Naffissa" Hana" "Imen" "Anfel" « Hadjer »  
Salsabîle"*

*et*

*"Rayan Ait Elrahman"*

*A*

*Mes chers neveux et nièces 'Syrine' Arwa" "Mohammed" "Amîna "  
'Ibrahîm "*

*Pour leur affection, compréhension et patience*

*A*

*Mes chères et fidèles amies*

*A*

*Tous ceux qui ont une relation de proche ou de loin  
avec la réalisation du présent rapport*

# Sommaire

## Introduction générale

Introduction générale .....	1
-----------------------------	---

## Chapitre I: Etat de l'Art

I . Introduction .....	3
II . Problématique .....	4
III . Vue du projet .....	4
IV . Objectifs du projet.....	5
V .1 Rendez-vous téléphoniques.....	7
V.2 Rendez-vous de visite physique .....	7
VI . Conclusion.....	8

## Chapitre II: Etudes analytiques

I . Introduction .....	9
II . Présentation des solutions existantes.....	9
II.1 Intérêt de la solution par rapport aux existantes .....	12
III . Analyse des besoins .....	13
III .1 Les acteurs de ce système .....	13
III .2 Besoins fonctionnels .....	13
III .3 Besoin non fonctionnelle .....	15
IV . Limites de l'application .....	16
V. Conclusion .....	16

## Chapitre I: Modélisation et Conceptions

I . Introduction .....	9
II . Méthodes et outils de modélisation .....	9
II .1 Langage de modélisation (UML).....	9
II.2. L'outil de modélisation .....	10
II.2.1 Diagrams.net (anciennement Draw.io).....	10
III . Représentation des diagrammes.....	10
III .1. Identification des acteurs.....	10
III .2. Description des cas d'utilisation.....	10
III .3. Diagrammes de cas d'utilisation.....	12

IV . Description et représentation des scénarios.....	14
IV . 1 Définition du mot scénario.....	15
V . Représentation des diagrammes d'activité.....	20
VI . Élaboration de diagramme de séquence.....	24
VII . Diagramme de classe.....	30
VIII . Conclusion.....	30

## **Chapitre IV: Implémentations**

I . Introduction.....	32
II . Architecture du système et environnement de test.....	32
II . 1 Définition d'une application web.....	32
II . 3 Les différentes architectures client/serveur.....	33
II . 3.1 Architecture client/serveur à deux niveaux.....	33
II . 3.2 Architecture client/serveur à trois niveaux.....	34
II . 4 Architecture model-vue-contrôleur (MVC).....	35
II . 5 Framework.....	36
a/ Définition Framework.....	37
b/ Pourquoi utiliser un Framework ?.....	37
II . 6 Architecture générale du système.....	38
III . Environnement de développement.....	39
III . 1. Environnement matériel.....	39
III . 2 Environnement logiciel.....	39
III . 2.1 Le Framework Laravel.....	39
III.2.2 Framework Bootstrap :.....	41
III.3 Langages de développement :.....	42
III . 4 Outils de développement :.....	43
V . Conclusion.....	50

## **Conclusion générale**

Conclusion générale.....	52
--------------------------	----

## **Références Bibliographiques**

Références bibliographiques.....	<b>Error! Bookmark not defined.</b>
----------------------------------	-------------------------------------

## Liste des figures

Figure 1 : Interface de "esiha" .....	10
Figure 2: Interface d'accueil de DzDoc .....	10
Figure 3:Interface d'accueil RDVToubib. ....	11
Figure 4:Quellques interfaces de Doctlib.....	11
Figure 5:Quellques interfaces de Maiaa. ....	12
Figure 6:Logo de diagrams.net.....	10
Figure 7 : Diagramme de cas d'utilisation de patient .....	12
Figure 8:Diagramme de cas d'utilisation de médecin.....	13
Figure 9:Diagramme de cas d'utilisation d'administrateur .....	14
Figure 10: Diagramme d'activité d'inscription .....	20
Figure 11:Diagramme d'activité d'authentification .....	21
Figure 12:Diagramme d'activité de chercher un médecin.....	22
Figure 13:Diagramme d'activité de consulter rendez-vous .....	22
Figure 14: Diagramme d'activité de prendre rendez-vous .....	23
Figure 15:diagramme d'activité pour gérer rendez-vous.....	24
Figure 16: diagramme de séquence d'authentification de patient .....	25
Figure 17:Diagramme de séquence d'authentification de médecin.....	26
Figure 18: diagramme de séquence de rechercher médecin.....	27
Figure 19: Diagramme de séquence pour prendre rendez-vous .....	28
Figure 20:Diagramme de séquence de gérer rendez-vous .....	29
Figure 21: Diagramme de classe.....	30
Figure 22: Architecture d'application web .....	33
Figure 23:Figure : Présentation de l'architecture à 2 niveaux .....	33
Figure 24:Architecture client /serveur à deux eaux .....	34
Figure 25: Architecture de MVS .....	36
Figure 26: jjjjjj .....	38

## Listes des Tableaux

Tableau 2:Besoin non fonctionnelle .....	15
Tableau 3:Description des cas d'utilisation.....	11
Tableau 4:Scénario descriptive de cas d'utilisation « Inscription » .....	15
Tableau 5:Scénario descriptive de cas d'utilisation « prendre RDV » .....	15
Tableau 6:Scénario descriptive de cas d'utilisation « chercher médecin.....	16
Tableau 7:Scénario descriptive de cas d'utilisation «prendre rendez-vous» .....	17
Tableau 8: Scénario descriptive de cas d'utilisation « Gérer rendez-vous » .....	19
Tableau 9 : Comparaison entre CMS et Framework .....	38



# Introduction générale

---

## Introduction générale

Si l'émergence de technologies nouvelles améliore considérablement notre confort de vie, elle se rend également propice à l'apparition de nouvelles maladies et de ce fait l'augmentation de la demande de soins.

Les conditions de vie moderne nous exposent par l'augmentation de la pollution, par la mauvaise alimentation, par l'exposition excessive au stress et par le choix d'une vie sédentaire principalement.

L'accroissement de la population en souffrance mobilise excessivement les soins de santé.

Avec le développement des nouvelles technologies la population a pu acquérir un nouveau savoir et de nouvelles connaissances cela a mené au fait que chacun devient un acteur de sa propre santé en s'inscrivant dans des prises en charges médicales personnalisées.

Dans plusieurs pays il est apparu depuis ces cinq dernières années une nouvelle méthodologie de demande de soins de santé empruntant internet. Ces applications portent le nom de doctolib en France, en Allemagne et au Canada ou encore Doctena en suisse pour ne citer que cela.

Par ce biais l'informatique devient un outil facilitant et améliorant considérablement l'accès aux soins et permet la prise de rendez-vous plus rapide et plus efficace que les précédentes méthodes qui empruntent essentiellement la téléphonie.

Sans aller jusqu'à la téléconsultation qui commence à peine à être utilisée dans certains pays comme les Etats Unis ou l'Europe, ce projet de fin d'études de Master vise à démontrer comment améliorer l'accès aux soins en Algérie.

Ce travail permet une approche critique qui démontre l'intérêt qu'apporte l'informatique dans les échanges entre professionnels de santé et leurs patients, il se compose de quatre parties.

La première décrit le projet et le contexte de l'application en partant des méthodes existantes de prise de rendez-vous. La seconde présente une étude préliminaire détaillée des systèmes d'information web existants de nominations. La troisième partie décrit de façon détaillée le concept de cette application. et enfin la quatrième partie présente les avantages de ce nouvel outil de prise de rendez-vous.

# Chapitre I : Etat de l'Art

---

## I . Introduction

La prise de rendez-vous médicaux en ligne est une vitalité indispensable Pour organiser la relation du patient aux soins de santé, notre travail consiste donc à la conception et l'implémentation d'une application web de prise de rendez-vous médicaux prendra en compte toutes les contraintes qui peuvent survenir lorsqu'un patient établi des réservations.

Le système de rendez-vous médical en ligne relie les professionnels de la santé et les patients, à l'aide d'un téléphone ou d'un ordinateur et d'un navigateur où que les utilisateurs se trouvent, L'utilisation de données pertinentes permet à ces types d'applications d'offrir de nombreux services à leurs utilisateurs, comme des recherches spécifiques pour tout médecin, ou la gestion d'un rendez-vous, ainsi ces services leur facilitent la vie et gagnent leur temps. Surtout pour les grands pays, les rendez-vous médicaux en ligne empêchent les patients de parcourir de longues distances juste pour prendre rendez-vous.

Cette application a pour objectif de réduire la fatiguée des patients en diminuant le temps d'attente lors de la prise de rendez-vous. Ainsi « Rendez-vous Médical en Ligne » s'adresse aux professionnels de santé disponibles 24h/24 et 7j/7, permettant aux médecins et aux patients de choisir le moment le plus approprié pour leurs examens médicaux.

Ce chapitre traite du problème de rendez-vous médical qui nous motive à créer cette application et des objectifs que nous voulons atteindre.

# Chapitre I : Etat de l'Art

---

## II . Problématique

La prise de rendez-vous médicaux avec des professionnels de santé en Algérie est en fait associée à plusieurs problèmes :

Souvent aller chez le médecin est un exercice fastidieux ; il faut d'abord trouver le médecin pas trop loin de chez soi ou de son travail et si on a de la chance il faut encore faire preuve de patience dans la salle d'attente avant de pouvoir faire sa consultation.

Il arrive parfois qu'on cherche longtemps alors qu'un médecin se trouve à quelques pas de soi, dans d'autres cas, le médecin est absent ou le cabinet a changé d'adresse sans qu'on le sache.

## III . Vue du projet

Afin de présenter le projet de fin d'études en Informatique et le second Master en Systèmes d'Information, Amélioration et Prise de Décision (ISOD), Nous présentons ce projet qui vise à fournir un système d'information simple et fiable (application web dynamique) pour soutenir la communication. Répondre de manière appropriée aux besoins des patients en leur offrant des services qui facilitent leur prise en charge médicale, pour réduire le taux de non-présentation, réduire le travail du personnel et le temps d'attente, améliorer la satisfaction, et permettre aux professionnels de mieux gérer leurs horaires de rendez-vous pour être plus reconnaissables, car la flexibilité, la sécurité et l'intégrité sont les principales raisons qui découragent les patients. À propos du passage à la planification basée sur le Web ,développé pour gérer le processus de prise de rendez-vous pour les professionnels de la santé et les organisations, quel que soit le type de service qu'ils choisissent, Les cabinets devront s'inscrire eux-mêmes sur le portail de prise de rendez-vous en ligne et pourront consulter les rendez-vous pris par les utilisateurs et les patients. Il aidera les utilisateurs à prendre leurs rendez-vous à l'aide de l'application Rendez-vous médicaux. Cela réduira le nombre d'appels pour prendre rendez-vous et évitera la cohue matinale pour prendre un rendez-vous urgent. En outre, cela réduira probablement le besoin de personnel d'accueil supplémentaire, ce qui réduit considérablement la main-d'œuvre

De plus, cela permet aux utilisateurs de gagner du temps et d'éviter d'avoir à négocier avec la réceptionniste pour une date appropriée. Cette plate-forme peut transformer le processus de recrutement laborieux existant et lui permettre de fonctionner de manière plus efficace, efficiente et rentable. Non seulement pour prendre rendez-vous, cette application II

## Chapitre I : Etat de l'Art

---

peut également être dédié à des études analytiques, car les données collectées par ce système peuvent constituer une base d'étude pour comprendre les besoins de santé de la population

Suggère aux décideurs les emplacements géographiques appropriés des professionnels de la santé en créant une liste des besoins de la population actuelle et des professionnels de la santé. Détection précoce de la maladie dans certaines régions grâce à la recherche de listes de patients dans la même région pour la même spécialité de médecins.

### IV . Objectifs du projet

Afin de contribuer à la démarche d'amélioration des conditions de prise en charge des patients, nous proposons ce travail consistant en la conceptualisation et la réalisation d'un outil dédié de prise de rendez-vous médicaux en ligne. Cet outil est destiné au patient pour lui permettre de mieux gérer ses rendez-vous médicaux à travers une gamme de services conçus dans le cadre de son parcours médical. A travers ces services, cet outil vise à contribuer à :

Pour le patient, à :

- ✓ Eviter le déplacement pour prendre un rendez-vous, se renseigner sur les médecins, ou pour obtenir leurs numéros de téléphone.
- ✓ Gagnez du temps : les patients n'auront plus à s'absenter du travail, à perdre une journée entière sur un simple rendez-vous de 15 minutes, ou à se réveiller à 4h du matin pour être en tête de la liste des malades du jour.
- ✓ Gagnez en confort : En réduisant le temps perdu dans les salles d'attente à un quart d'heure avant la consultation, le patient ne se fatiguera pas et affrontera à nouveau l'attente ennuyeuse.
- ✓ Gain de vision : Avec le système actuel, un nouveau médecin qui s'installe dans un nouveau lieu n'a pas forcément une bonne vision à proximité des patients pour débiter son activité médicale. Cependant, avec l'outil proposé, il aura une vision similaire des cliniciens formés et pourra profiter pleinement d'un véritable support de communication avec le patient.
- ✓ Faites des économies : en évitant les déplacements inappropriés, les allers-retours inutiles, et les dépenses de sommes inutiles.

**Pour le médecin, à :**

## Chapitre I : Etat de l'Art

---

- ✓ Gain de vision : Avec le système actuel, un nouveau médecin qui s'installe dans un nouveau lieu n'a pas forcément une bonne vision à proximité des patients pour débiter son activité médicale. Cependant, avec l'outil proposé, il aura une vision similaire aux médecins précédents et pourra profiter pleinement d'un véritable support de communication patient.
- ✓ Meilleure gestion des agendas : sélectionner des cases d'accueil, se déplacer si besoin en accord avec les patients sur des rendez-vous contrairement à ses autres restrictions
- ✓ Maîtrise des coûts : plus besoin d'embaucher une équipe d'hôtes pour gérer les annonces,
- ✓ Améliorer l'image du médecin vis-à-vis de son patient : Avec les conditions d'accueil souvent dures et les longues attentes dans les salles d'attente dépourvues parfois des moyens nécessaires pour reconforter les patients, les patients sont pris en otage avec des prestations qui ne sont pas toujours à la hauteur de leurs attentes, notamment pour leur état de santé, et le caractère humain de la profession santé, la distance parcourue et les sommes importantes dépensées. Avec ce type d'outil, un professionnel peut réserver un accueil plus humain et personnalisé à ses patients. Cela aidera les médecins à redonner confiance à leurs patients et à se sentir entre de bonnes mains.

Plus précisément, parmi les services de base que cette application doit fournir, on peut Citation :

- ✓ Chercher des médecins par leurs coordonnées, leur spécialité, leur emplacement, leur prochaine disponibilité et l'établissement de leur établissement (pour les hôpitaux et les cliniques).
- ✓ Obtenir des informations sur les médecins : Consultez leurs informations générales (adresse, localisation, numéros de téléphone, horaires d'ouverture) et des informations détaillées (spécialité, points d'expérience, diplômes).
- ✓ Gérer les rendez-vous : prendre rendez-vous, déplacer un rendez-vous, annuler un rendez-vous, approuver un rendez-vous.

Avant de commencer à analyser le domaine de la prise de rendez-vous en ligne, nous devons définir les processus des méthodes primitives de prise de rendez-vous.

### V . Méthodes de rendez-vous médical hors ligne

Il existe deux méthodes pour planifier un rendez-vous hors ligne pour un examen médical en Algérie, soit un appel téléphonique, soit une visite physique chez le professionnel de santé (rendez-vous physique). Ci-dessous la description de chaque méthode.

#### V .1 Rendez-vous téléphoniques

La méthode de planification d'un appel nécessite une connaissance préalable des numéros de téléphone des professionnels , Ce problème amène de nombreuses personnes à demander et à rechercher les numéros de téléphone des médecins pour planifier leurs rendez-vous médicaux, et ce processus est répété chaque fois que les patients doivent consulter un nouveau médecin et prendre rendez-vous.

#### V.2 Rendez-vous de visite physique

Physical visit appointment method, the patients should wake up early and go to the professional, to register their names in the list of patients' appointments. This process is very tiring, especially for the patients who come from far distances, which disturbs them from their labors and tasks.

### VI . Conclusion

Les soins de santé de la population sont un domaine humain d'une importance cruciale, et avec chaque innovation ou développement lié à la santé des patients, qui répond à leurs besoins, cela leur donne une nouvelle occasion de profiter d'une bonne santé. Faciliter les rendez-vous médicaux fait partie des besoins des patients. Par conséquent, développer une application de prise de rendez-vous en ligne est très nécessaire pour faciliter la tâche des patients et des professionnels et leur éviter des voies difficiles (appel téléphonique et visite physique) pour gérer les rendez-vous.

Dans ce chapitre, nous avons présenté le cadre général de notre projet pour développer un système qui facilitera la gestion des rendez-vous médicaux, Après avoir fixé nos objectifs, l'étape suivante sera consacrée à une analyse et la spécification des besoins

## Chapitre II : Etudes analytiques

---

### I . Introduction

La création d'une application de prise de rendez-vous médical en ligne nécessite une bonne analyse approfondie pour connaître les imperfections des applications existantes dans ce domaine et les besoins des patients aussi.

Par conséquent, nous nous concentrons sur ces besoins pour améliorer les services de ce domaine pour obtenir la satisfaction des patients et des professionnels.

Ainsi, dans ce chapitre et avant de commencer la partie conception, il est nécessaire de comprendre et d'analyser le domaine de la prise de rendez-vous médical en ligne, en analysant le mode de prise de rendez-vous en ligne existant, et les données nécessaires au bon fonctionnement de cette application.

Nous allons jeter un œil et analyser les applications déjà développées dans ce domaine pour définir les besoins des utilisateurs (patients et professionnels de santé), et définir les fonctionnalités nécessaires de ce système pour les atteindre.

### II . Présentation des solutions existantes

Concernant la prise de rendez-vous médical en ligne en Algérie, il existe actuellement quelques applications web, Il s'agit notamment de :

***eSiha*** : eSiha est une application qui vous permet de trouver un médecin près de chez vous, de voir ses disponibilités ou même de prendre un rendez-vous sans besoin de vous déplacer. Elle vous permet également d'avoir une téléconsultation, et peut aussi permettre au médecin de rédiger une ordonnance numérique qui sera envoyée au pharmacien. [1]

Cette application dédiée aux médecins et non seulement les patients, vous permet de gérer vos patients et leurs dossiers médicaux grâce à la fonctionnalité d'enregistrement d'historique, ce qui rend le dossier médical mobile et peut être envoyé vers un autre médecin si le patient change son médecin.

Non seulement ça, mais elle est dédiée aussi aux centres médicaux et permet de gérer chaque département ou spécialité



## Chapitre II : Etudes analytiques



Figure 1 : Interface de "esiha"

**DzDoc** : Plateforme médicale Multi-services en Algérie, qui a pour but de digitaliser le secteur de la santé en et offrir des solutions pour prise des rendez-vous médicaux en ligne .[2]

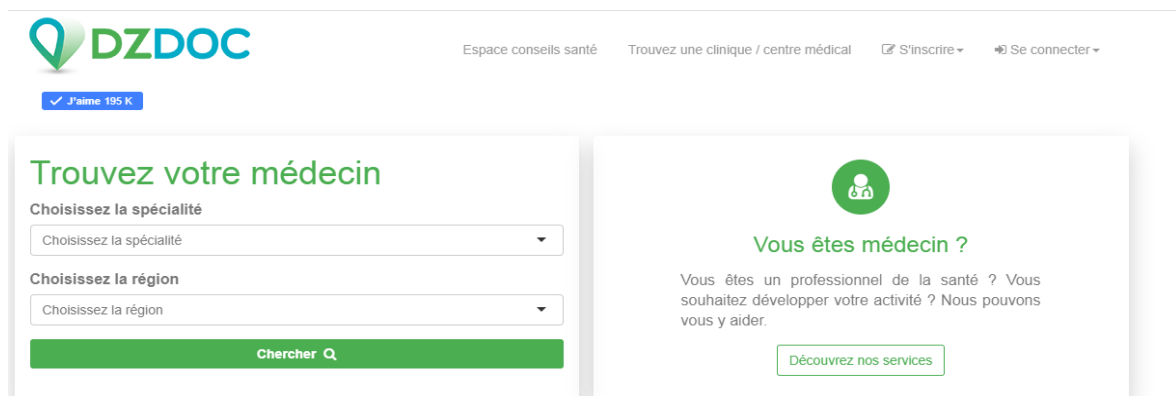


Figure 2: Interface d'accueil de DzDoc

**RdvToubib** : Plateforme qui permet à tout un chacun de prendre rendez-vous chez le médecin directement en ligne, c'est une solution qui ambitionne de simplifier la vie des patients algériens. [3]

## Chapitre II : Etudes analytiques

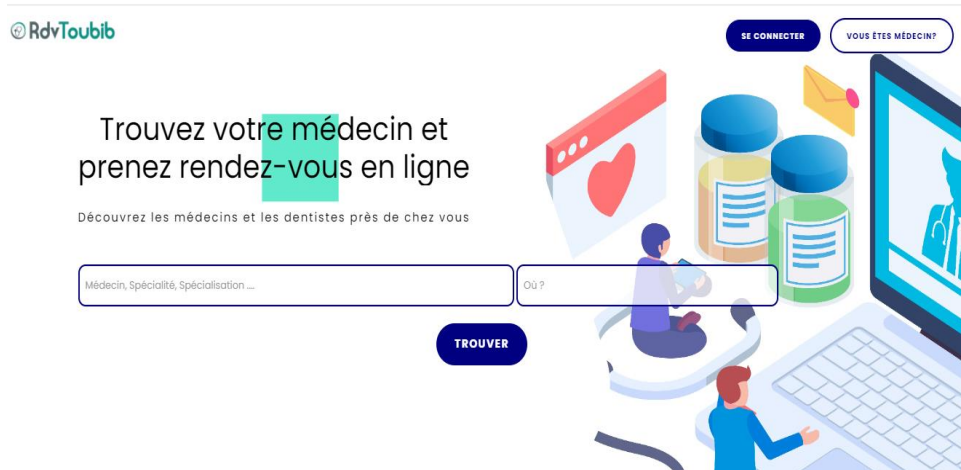


Figure 3:Interface d'accueil RDVToubib.

Il existe également d'autres sites étrangers tels que :

**Doctolib :** Doctolib est une entreprise franco-allemande fondée en 2013 par Stanislas Niox-Château qui propose un service de gestion en ligne des consultations pour les professionnels de santé et un service de prise de rendez-vous en ligne pour Les patients. [5]

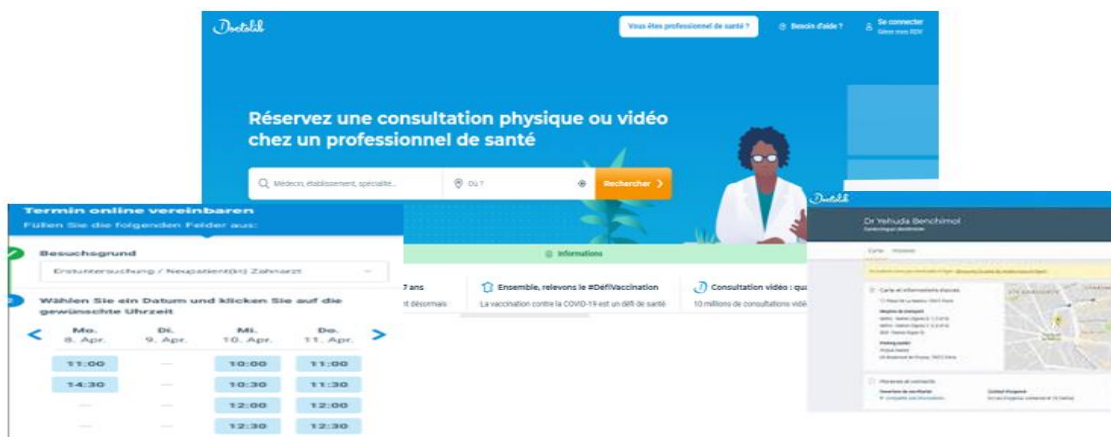


Figure 4:Quelques interfaces de Doctolib

## Chapitre II : Etudes analytiques

**Maiia** : Maiia appartient au groupe Cogedim, fondé en 1969 et spécialiste de la conception de logiciels métiers et solutions destinés aux professionnels de santé. C'est une application aide les professionnels de santé à se concentrer sur le soin de leurs patients, via une offre complète de services (agenda, téléconsultation) Suite au lancement réussi de sa solution de téléconsultation auprès des praticiens en novembre 2018, elle accompagne chaque mois des centaines de nouveaux médecins. [5 ]



Figure 5:Quelques interfaces de Maiia.

### II.1 Intérêt de la solution par rapport aux existantes

Selon nos recherches sur ces applications Web pour prise de rendez-vous médicale en ligne, nous pouvons recueillir des informations sur ces applications pour certains critères. Toutes ces applications utilisent presque la même idée et la même méthode de prise de rendez-vous ; Mais il y a peu de différences entre eux.

D'après l'analyse de celle-ci, on peut dire que toutes ces applications sont modernes, ce qui explique la couverture limitée ou très limitée des professionnels de santé. Pour pallier ce problème de couverture géographique, nous constatons que l'utilisation de la plupart de ces applications est gratuite,

Les applications DZDOC et esiha restent plus avancées que les autres applications en termes de services fournis en Algérie

### III . Analyse des besoins

L'analyse consiste à l'aboutissement de l'élaboration d'une solution technique à partir de l'étude des besoins.

C'est la première phase du cycle de développement d'un logiciel. Elle sert à identifier les acteurs du système et leur associer chacun l'ensemble des actions avec lesquelles il intervient dans l'objectif de donner un résultat optimal et satisfaisant au client.

#### III .1 Les acteurs de ce système

- Patient : Contact pour choisir le bon moment et médecin pour prendre rendez-vous
- Professionnelle de sante : se connectent pour gérer leurs agendas (Rendez-vous) selon les établissements de travail (Hôpital, Clinique, Cabinet Médical, dentiste, pharmacie, laboratoire, centre de radiologie, laboratoire d'analyse).
- Administrateur : il est responsable de la gestion de toutes les fonctionnalités principales du système.

#### III .2 Besoins fonctionnels

Les besoins fonctionnels répondent aux points précis du cahier de charges et sont donc requis par le client. Ils constituent le besoin primaire du client et définissent une fois résolu l'opérationnalité du système. Ainsi l'application de gestion de prise de rendez-vous santé doit permettre :

## Chapitre II : Etudes analytiques

---

Utilisateur	Les besoins fonctionnels
Patient	<ul style="list-style-type: none"><li>✓ Authentifier</li><li>✓ Créer compte personnel</li><li>✓ Gérer son compte</li><li>✓ Chercher du professionnel de santé</li><li>✓ Prendre un rendez-vous</li><li>✓ Consulter rendez-vous</li><li>✓ Contacter utilisateur (patient, médecin, administrateur)</li></ul>
Professionnel De santé	<ul style="list-style-type: none"><li>✓ Authentifier.</li><li>✓ Créer compte personnel</li><li>✓ Gérer son compte</li><li>✓ Géré son agenda</li><li>✓ Gérer les rendez-vous.</li><li>✓ Prendre rendez-vous.</li><li>✓ Consulter rendez-vous</li><li>✓ Contacter utilisateur</li></ul>
Administrateur	<ul style="list-style-type: none"><li>✓ Authentifier</li><li>✓ Gérer des professionnels de santé et patients.</li><li>✓ Gérer les spécialités</li><li>✓ Gérer les établissements.</li><li>✓ Contacter utilisateur</li></ul>

## Chapitre II : Etudes analytiques

---

### III .3 Besoin non fonctionnelle

Les besoins non fonctionnels sont importants car ils agissent de façon indirecte sur le résultat et sur la performance du système, ce qui fait qu'ils ne doivent pas être négligés, pour cela il faut répondre aux exigences comme dans ce tableau suivant :

**Tableau 1:**Besoin non fonctionnelle

<b>Sécurisé</b>	<ul style="list-style-type: none"><li>✓ Le système doit être sécurisé au niveau des données pour authentification et contrôle des droits d'accès.</li></ul>
<b>Performante</b>	<ul style="list-style-type: none"><li>✓ Le Temps de réponse dans navigateur (Temps de chargement d'une page web) ne devrait pas prendre plus de 15 secondes en condition normale.</li><li>✓ Compatible à la plupart des navigateurs</li></ul>
<b>Utilisable</b>	Font référence aux aspects généraux de l'interface utilisateur <ul style="list-style-type: none"><li>✓ Facilité de compréhension.</li><li>✓ Facilité d'apprentissage.</li><li>✓ Facilité exploitation/d'utilisation.</li><li>✓ Attractivité.</li></ul>
<b>Maintenabilité</b>	<ul style="list-style-type: none"><li>✓ Facilité d'analyse.</li><li>✓ Facilité de modification.</li><li>✓ Stabilité.</li><li>✓ Testabilité.</li></ul>
<b>Matériels</b>	<ul style="list-style-type: none"><li>✓ Définissent les configurations matérielles minimales nécessaires au fonctionnement du système (Pentium 4, 2G...etc.)</li></ul>

### IV . Limites de l'application

En général, lorsque les gens entendent parler d'une nouvelle application qui répond à l'un de leurs besoins, ils s'attendent à d'autres services que cette application peut également fournir, donc l'utilisation de ce qui suit est destinée à montrer les limites de cette application :

- Ce système n'est utilisé que lorsque l'utilisateur est en ligne, sinon il ne fonctionne pas.
- Cette plateforme permet de prendre rendez-vous sur le lieu de travail des médecins
- Cette application ne propose pas de consultations en ligne, elle limite les dates de réservation,
- Ce système ne gère pas les documents médicaux (ordonnance).

### V. Conclusion

Dans ce chapitre, nous avons présenté certaines solutions préexistantes, et nous avons exposé l'analyse et la spécification des besoins fonctionnel, et non fonctionnel pour permettant de concevoir et de développer un système qui facilitera la prise de rendez-vous médicale en ligne.

Après avoir fixé les besoins système, l'étape suivante sera consacrée à une conception détaillée.

# Chapitre III : Modélisation et conception

---

## I . Introduction

Le processus de conception de notre application repose sur l'organisation conceptuelle, logique et physique des données collectées durant la phase analyse. En effet, elle s'appuie essentiellement sur quelques diagrammes du langage de modélisation.

Dans ce chapitre nous commençons par une présentation des différents outils logiciels et les langages de modélisations utilisés. Ensuite nous détaillons les diagrammes des cas d'utilisation, les diagrammes des classes et les diagrammes de séquences.

## II . Méthodes et outils de modélisation

### II .1 Langage de modélisation (UML)

Pour la conception de notre système nous avons adopté une méthode orientée objet. En effet cette dernière est une approche incontournable dans le cadre du développement des applications.

Pour mieux présenter l'architecture de notre système, on va choisir le langage de modélisation le plus adopté UML : C'est un langage de modélisation, défini comme une norme de modélisation objet qui sert à décrire et à documenter un système d'information.[6]

En utilisant ce langage, les objectifs de la modélisation objet suivant sont assurés :

- ✓ Formaliser la conception d'application.
- ✓ Faciliter la communication entre les différents intervenants au sein d'un projet informatique.
- ✓ Coordonner les activités entre les différents intervenants.
- ✓ Gérer l'évolution d'un projet informatique.
- ✓ Proposer des outils standardisés prenant en compte de nombreux aspects de la conception.



## Chapitre III : Modélisation et conception

---

### II.2. L'outil de modélisation

Pour mieux organiser notre travail. Nous avons utilisé cet outil (diagrams.net) parce qu'est très simple d'utilisation et s'avère très intuitif

#### II.2.1 Diagrams.net (anciennement Draw.io)

Diagramme.net est une application création de diagrammes compatible avec Google Drive (TM) et entièrement gratuite vous permettant de dessiner : [7]

- ✓ Des organigrammes.
- ✓ Des diagramme UML.
- ✓ Des diagramme ERD.
- ✓ Des schémas réseaux.
- ✓ Business Process Model.
- ✓ Organigrammes.
- ✓ Des circuits électroniques.
- ✓ Maquettage d'interface.



Figure 6:Logo de diagrams.net

### III . Représentation des diagrammes

Cette partie de notre modélisation consistera à présenter les différents acteurs du système ainsi que les différentes interactions qu'ils pourront effectuer avec celui-ci.

#### III .1. Identification des acteurs

En UML, on entend par acteur toute idéalisation d'un rôle joué par une personne externe, un processus ou une entité qui interagit avec le système.

Dans notre system il

#### III .2. Description des cas d'utilisation

Les acteurs et les cas d'utilisation sont résumés dans le tableau suivant :

## Chapitre III : Modélisation et conception

---

**Tableau 2:Description des cas d'utilisation**

Acteur	Cas d'utilisation
<b>Patient</b>	Chercher médecin (par nom ou par spécialité ou par établissement et localisation sur google maps)
	Prendre un rendez-vous après la recherche du médecin.
	Gérer ses rendez-vous et peut (Annuler ou modifie)
	Gérer son compte personnel et il peut (modifier ou supprimer) son compte
	Doit s'authentifier pour accède son compte
	Peut contacter administrateur ou médecin
<b>Médecin</b>	Doit s'authentifier pour accède son compte
	Gérer les rendez-vous et il peut (Accepter ou Déplacer ou Annuler)
	Gérer s'agenda après la choisir la date et il peut (ajouter ou annuler) une journée ou ajouter une vacance
	Consulter les rendez-vous (rendez-vous demandée, rendez-vous occupée, rendez-vous archivée)
	Contacteur administrateur et il peut proposer d ajoute (médecin ou spécialité ou établissement)
	Gérer son compte personnel et il peut (modifier ou supprimer) sa compte
<b>Administrateur</b>	Doit s'authentifier pour accède son compte
	Gérer médecins et patients
	Gérer les spécialités avec la possibilité de suppression
	Gérer les établissements avec leur localisation.et la possibilité de suppression
	Contacteur un utilisateur (administrateur, patient, médecin)

## III .3. Diagrammes de cas d'utilisation

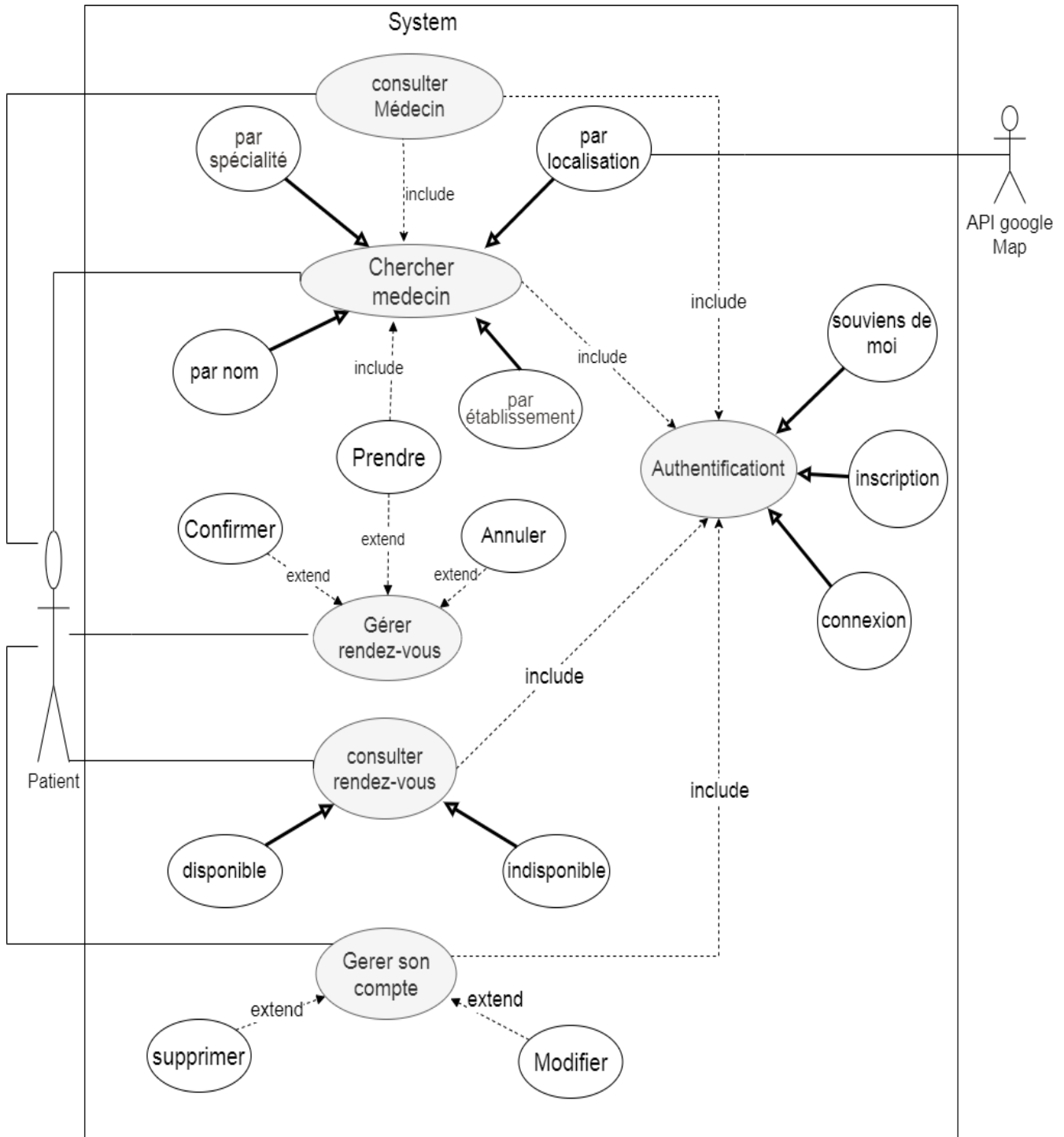


Figure 7 : Diagramme de cas d'utilisation de patient

## Chapitre III : Modélisation et conception

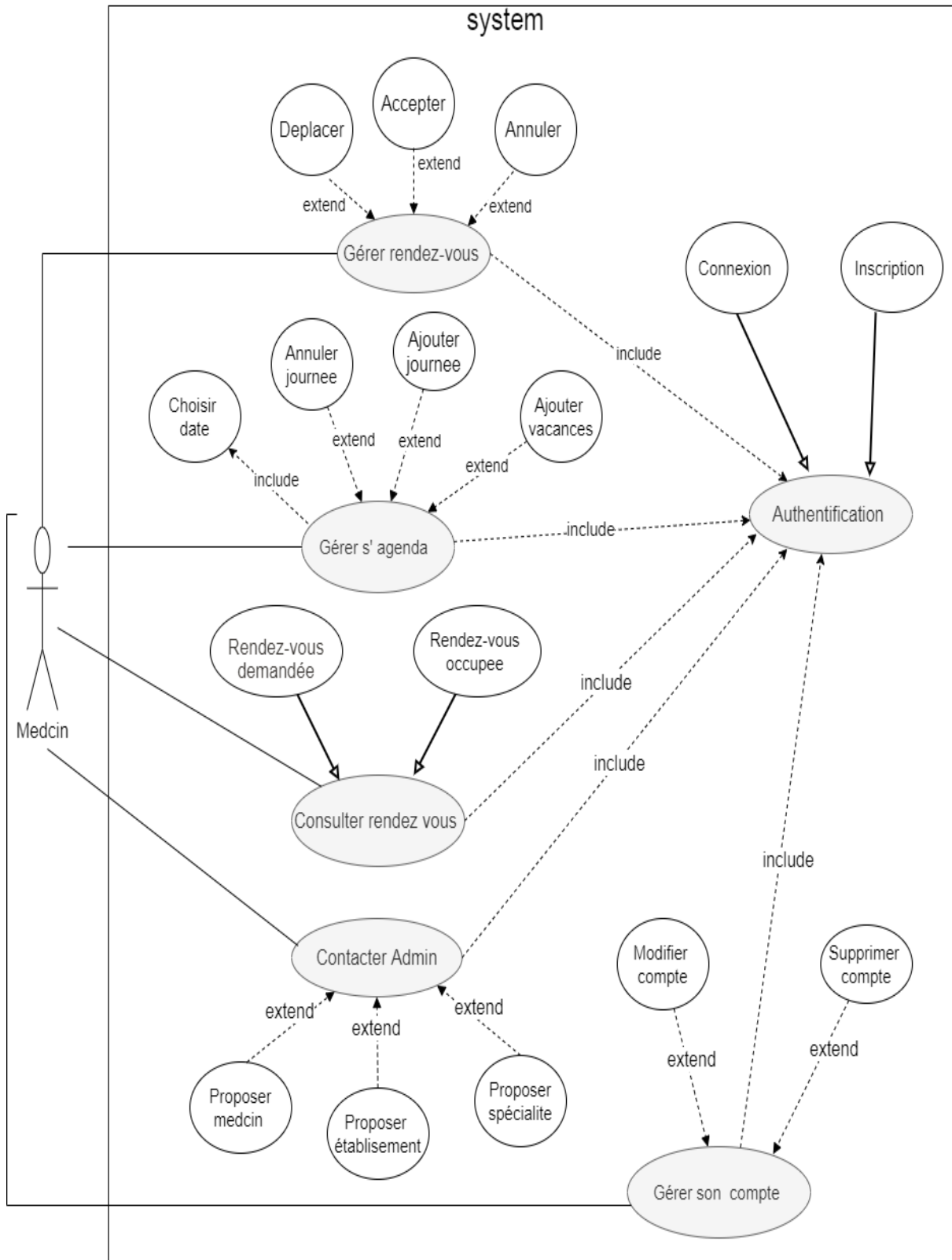


Figure 8: Diagramme de cas d'utilisation de médecin

## Chapitre III : Modélisation et conception

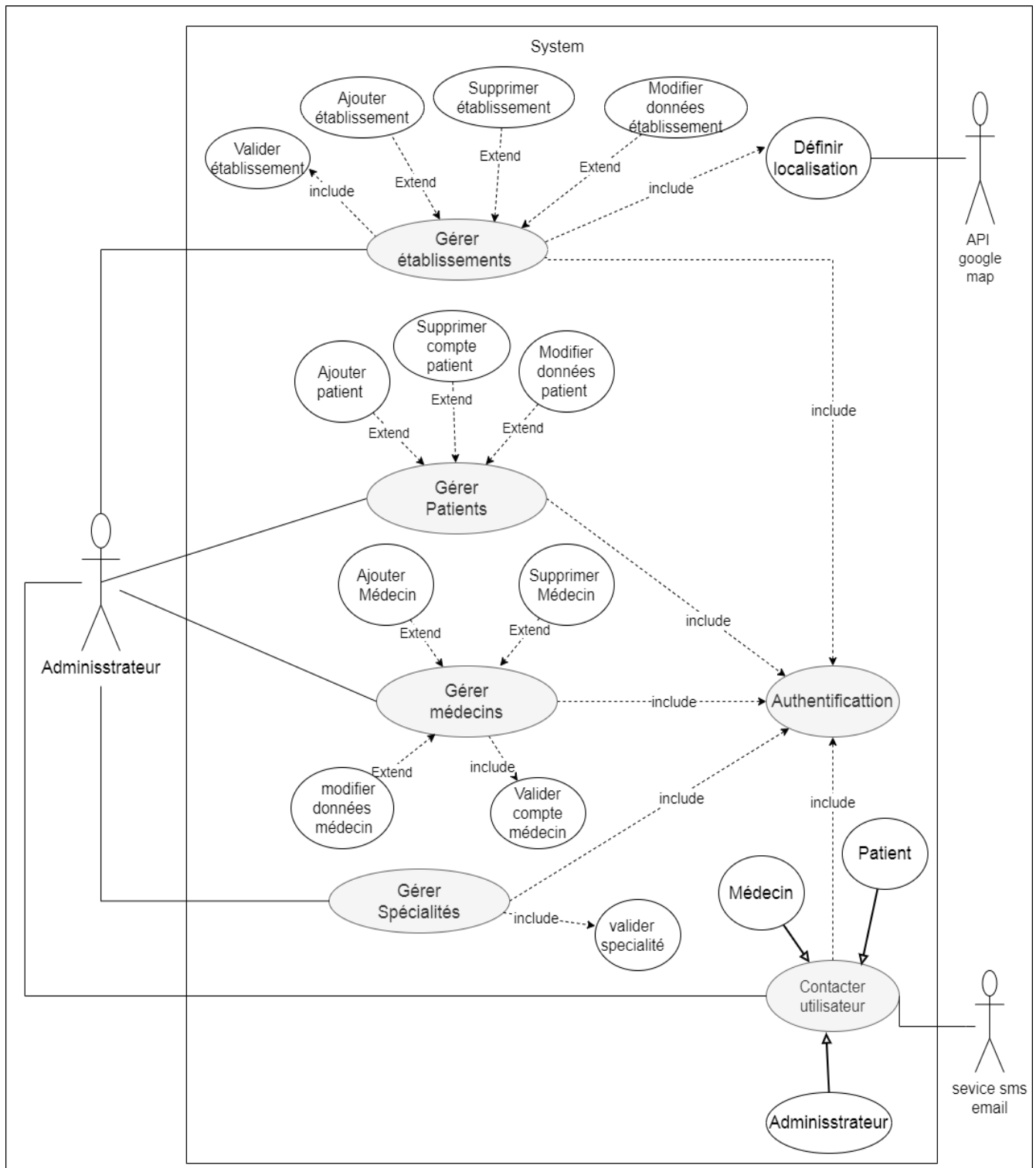


Figure 9:Diagramme de cas d'utilisation d'administrateur

### IV . Description et représentation des scénarios

Dans cette section on va définir le mot scénarios et élaborer les fiches descriptives.

## Chapitre III : Modélisation et conception

---

### IV . 1 Définition du mot scénario

Un scénario décrit une exécution particulière d'un cas d'utilisation du début à la fin. Il correspond à un enchaînement d'actions du cas d'utilisation, se terminant par une fin normale ou non (exception ou erreur).

En général, il n'est pas possible de décrire tous les scénarios d'un cas d'utilisation. Il faudra choisir les principaux scénarios nominaux (ceux qui sont fréquents) et quelques scénarios d'exception.

**Tableau 3:Scénario descriptive de cas d'utilisation « Inscription »**

Cas d'utilisation	Inscrire
Acteur :	Patient, ou médecin
Objectif :	Ce cas permet le patient de créer un compte personnel
Scénario nominal :	<ol style="list-style-type: none"><li>1. Le patient choisit formulaire d'inscription</li><li>2. Le system affiche formulaire d'inscription</li><li>3. Le patient saisir son information (id, nom, email, adresse,)</li><li>4. Le system valider l'information saisies</li><li>5. Le system vérifie formulaire</li><li>6. Le system sauvegarder information dans la base de données</li></ol>
Alternative :	5 : si l'information incorrecte allez à 3.
Scenarion d'erreur :	<ol style="list-style-type: none"><li>1.Le patient saisit des informations erronées</li><li>2.Le system envoyer un message d'erreur au patient</li></ol>

**Tableau 4:Scénario descriptive de cas d'utilisation « prendre RDV »**

Cas d'utilisation	Inscrire
-------------------	----------

## Chapitre III : Modélisation et conception

---

Acteur :	Médecin
Objectif :	Ce cas permet le médecin de créer un compte personnel
Scénario nominal :	<ol style="list-style-type: none"><li>7. Le médecin choisit formulaire d'inscription</li><li>8. Le system affiche formulaire d'inscription</li><li>9. Le médecin saisir son information (id, nom, email, adresse,)</li><li>10. Le médecin sélectionne une spécialité, établissement</li><li>11. Le system valider l'information saisies par médecin</li><li>12. Le system vérifie formulaire</li><li>13. Le system sauvegarder information dans la base de données</li></ol>
Alternative :	5 : si l'information incorrecte allez à 3.
Scenariio d'erreur :	<ol style="list-style-type: none"><li>1.Le patient saisit des informations erronées</li><li>2.Le system envoyer un message d'erreur au patient</li></ol>

**Tableau 5:Scénario descriptive de cas d'utilisation « chercher médecin**

Cas d'utilisation	Chercher médecin
Acteur :	Patient

## Chapitre III : Modélisation et conception

Objectif :	Ce cas permet au patient de chercher un médecin
Condition	Le patient authentifie
Scénario nominal :	<ol style="list-style-type: none"> <li>1. Le patient sélectionne la barre de recherche.</li> <li>2. Le patient saisit le nom du médecin.</li> <li>3. Le système affiche tous les médecins avec ce nom.</li> <li>4. Le patient choisit un médecin de la liste</li> <li>5. Le système affiche le profil de médecin choisie</li>   <li>6. Le patient sélectionne une spécialité.</li> <li>7. Le system affiche tous les médecins avec spécialité choisie.</li> <li>8. Le patient choisit un médecin de la liste.</li> <li>9. Le système affiche le profil de médecin choisie</li>   <li>10. Le patient sélectionne un établissement.</li> <li>11. Le system affiche tous les médecins de l'établissement choisie.</li> <li>12. Le patient choisit un médecin de la liste.</li> <li>13. Le système affiche le profil du médecin sélectionné</li>   <li>14. Le patient chercher médecin par localisation</li> <li>15. Le système afficher tous les médecins et l'établissement localise Dans position choisie</li> <li>17. Le patient choisit et sélectionne un médecin de la liste.</li> <li>18. Le système affiche le profil du médecin sélectionné</li> </ol>
Scenario d'erreur :	<p>Le patient saisie un nom erroné</p> <p>Le patient ne saisit aucun nom</p> <p>Le patient ne choisit aucun (spécialité, établissement, location)</p> <p>Le patient ne choisit aucun médecin</p>

**Tableau 6: Scénario descriptive de cas d'utilisation «prendre rendez-vous»**

Cas utilisation :	Prendre rendez-vous
Acteur :	patient



## Chapitre III : Modélisation et conception

---

Objectif :	Permet au patient prendre rendez-vous
condition :	Chercher médecin, authentifier.
Scénario nominal :	<ol style="list-style-type: none"><li>1. Le patient recherche un médecin</li><li>2. Le patient selectionner un médecin.</li><li>3. Le système affiche le calendrier du médecin.</li><li>4. Le patient choisit une date de rendez-vous.</li><li>5. Le système affiche une formulaire de choix authentification (connexion ou s'inscrire )si Le session pas ouvert .</li><li>6. le patient choisit de s'authentifier pour confirmer le rendez-vous</li><li>7. Le patient confirmer la demandes</li><li>8. le système sauvegarder la demandes</li></ol>
Alternative :	6 : si session déjà ouverte. Allez à 8.
Scenariod'erreur :	<ol style="list-style-type: none"><li>1.Le patient ne recherche aucun médecin</li><li>2.Le patient ne choisit aucun date de rendez-vous .</li><li>3.Le patient ne confirme pas la demandes de rendez-vous</li></ol>

## Chapitre III : Modélisation et conception

---

**Tableau 7: Scénario descriptive de cas d'utilisation « Gérer rendez-vous »**

Cas d'utilisation :	Gérer rendez-vous
Acteur :	Médecin
Objectif	Permet au médecin gérer rendez-vous
Précondition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"><li>1. Le médecin choisit liste de rendez-vous</li><li>2. Le system afficher liste rendez-vous</li><li>3. Le médecin sélectionner rendez-vous</li><li>4. Le médecin choisit une option pour (accepter ou déplacer ou rejeter) rendez-vous.</li><li>5. Le médecin sélectionner nouvelle date et temps</li><li>6. le système sauvegarder la demandes</li></ol>
Alternative :	<p>4 : Si le médecin choisit déplacer allez à 5</p> <p>4 : Si le médecin choisit accepter allez à 6.</p>
Scénario d'erreur	<ol style="list-style-type: none"><li>1. Le médecin ne choisit aucun rendez-vous pour la gestion</li><li>2. Le médecin ne choisit aucun date pour deplacer un rendez_vous</li><li>3. Le patient ne confirme pas la demandes de rendez-vous</li></ol>

### V . Représentation des diagrammes d'activité

Dans cette étape nous allons présenter le diagramme d'activité de quelques cas d'utilisation :

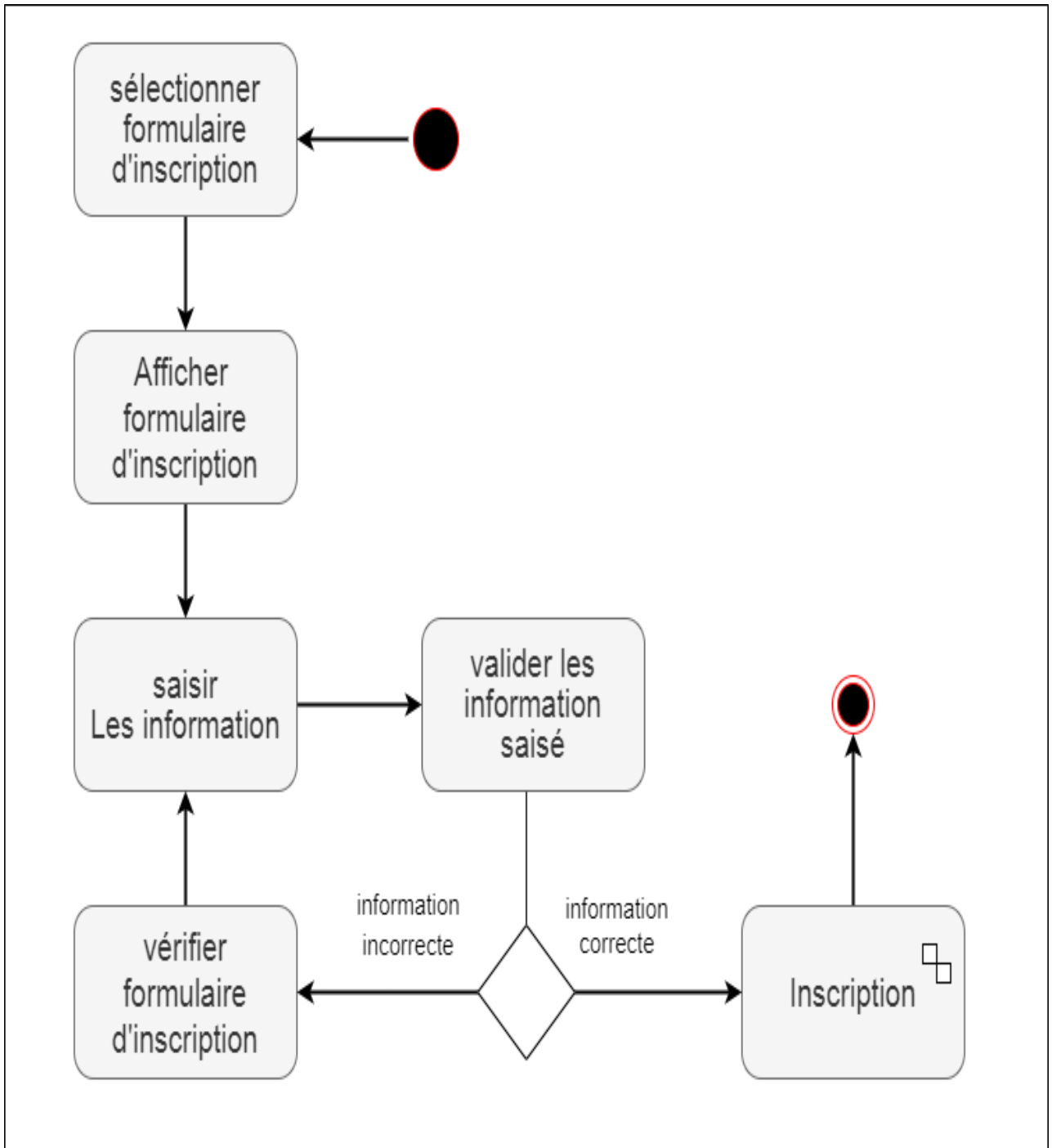


Figure 10: Diagramme d'activité d'inscription

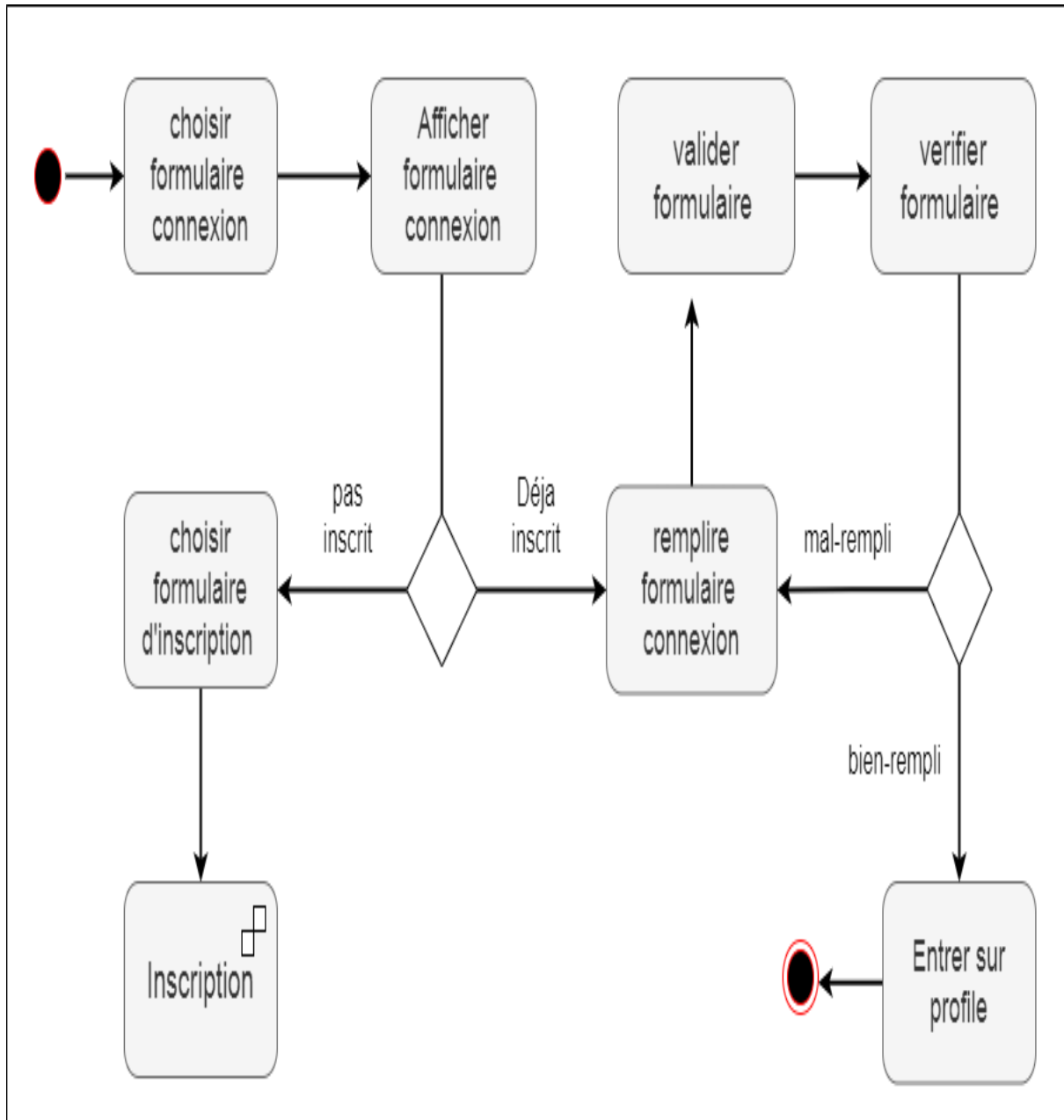


Figure 11:Diagramme d'activité d'authentification

## Chapitre III : Modélisation et conception

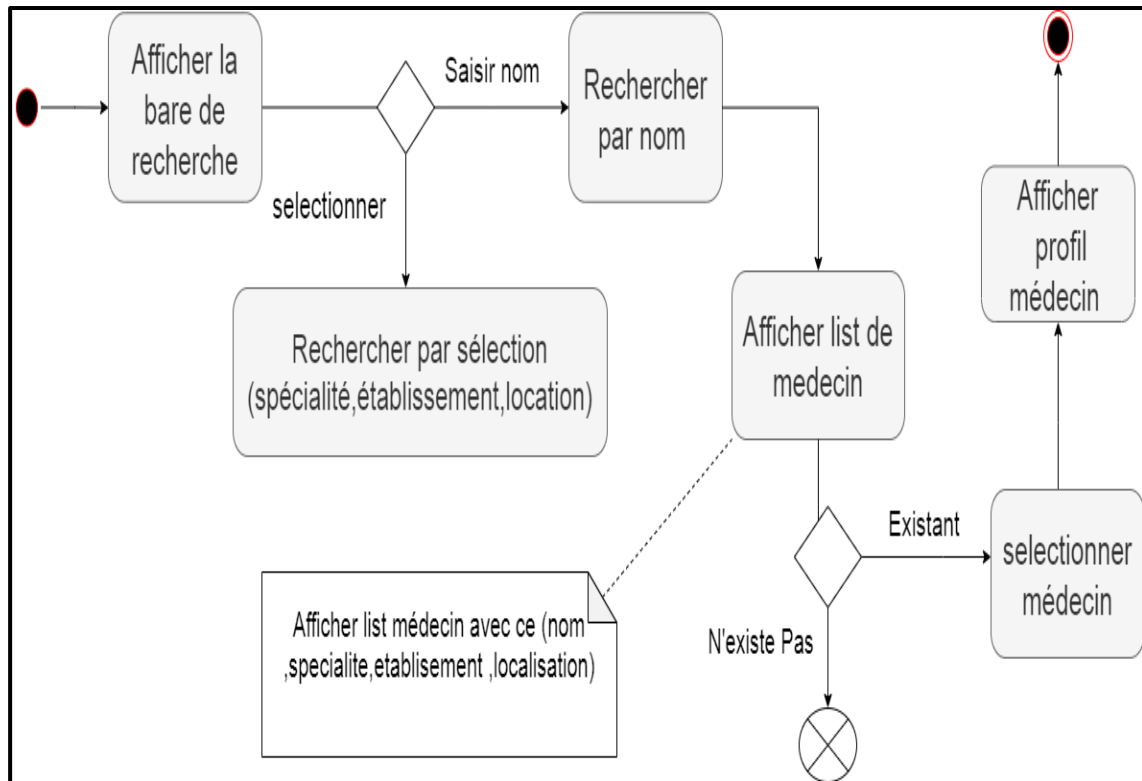


Figure 12:Diagramme d'activité de chercher un médecin

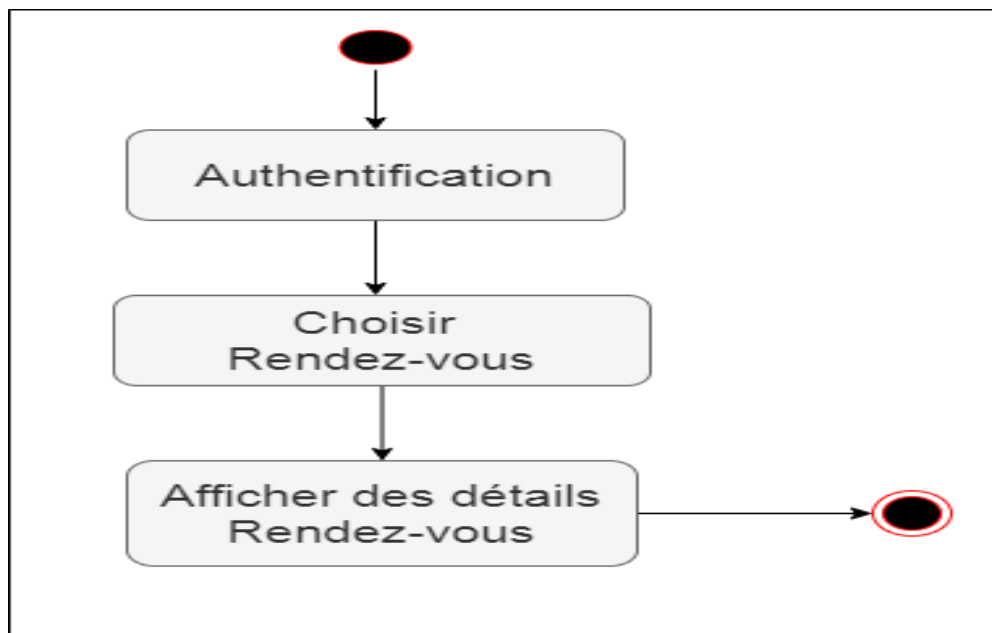


Figure 13:Diagramme d'activité de consulter rendez-vous

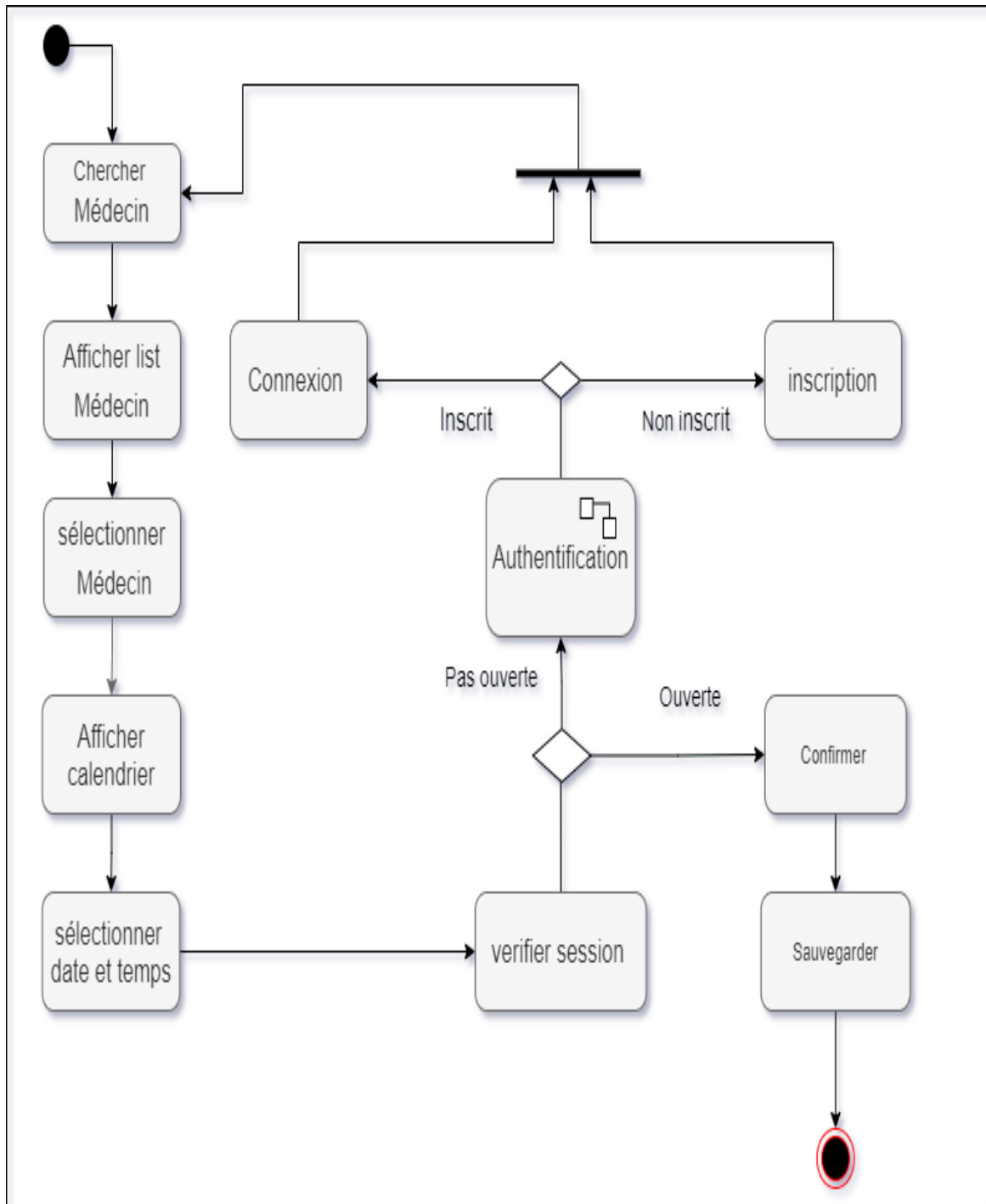


Figure 14: Diagramme d'activité de prendre rendez-vous

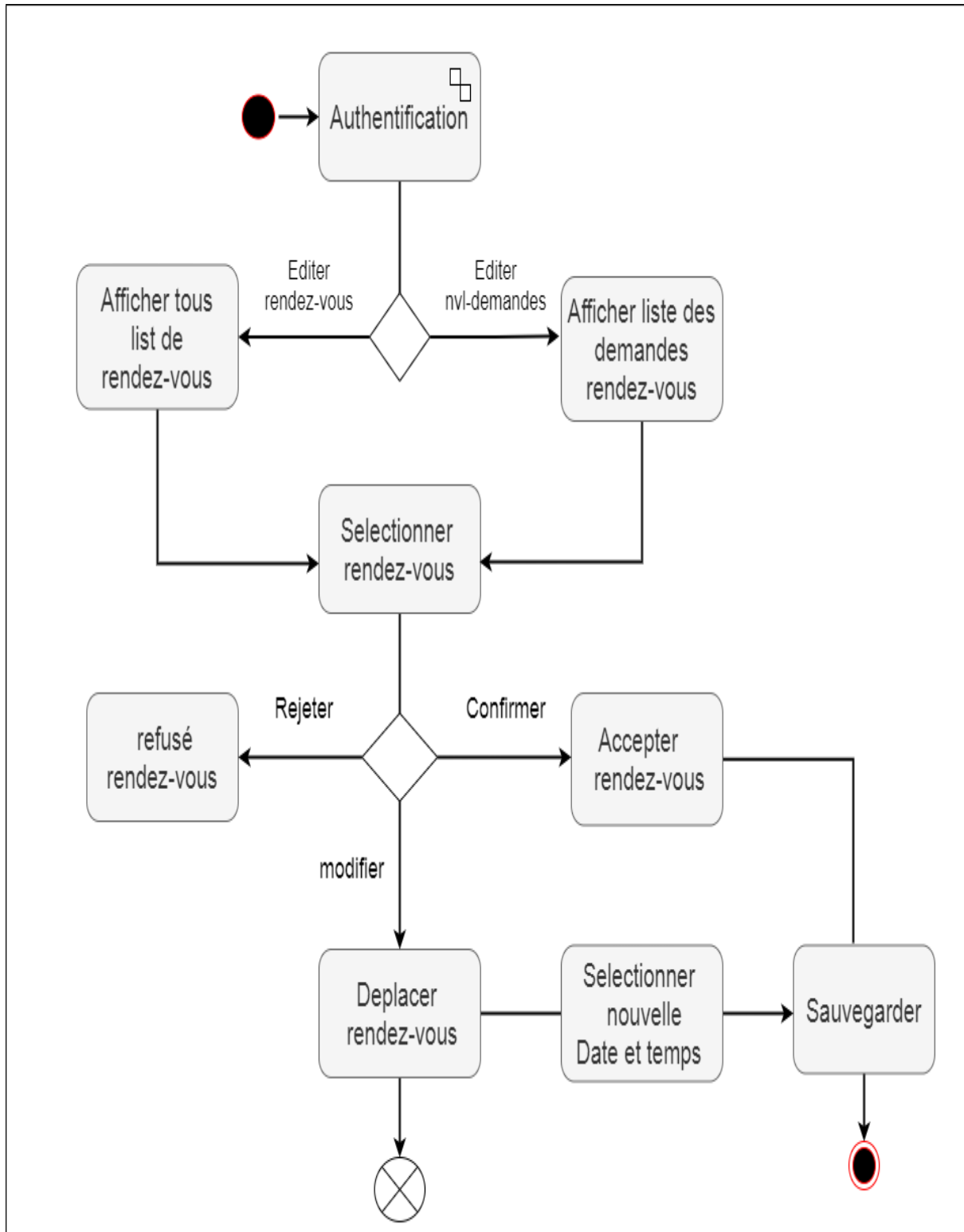


Figure 15:diagramme d'activité pour gérer rendez-vous

### VI . Élaboration de diagramme de séquence

## Chapitre III : Modélisation et conception

Après qu'on a vu le diagramme d'activité nous allons voir maintenant le diagramme de séquence pour quelque cas d'utilisation.

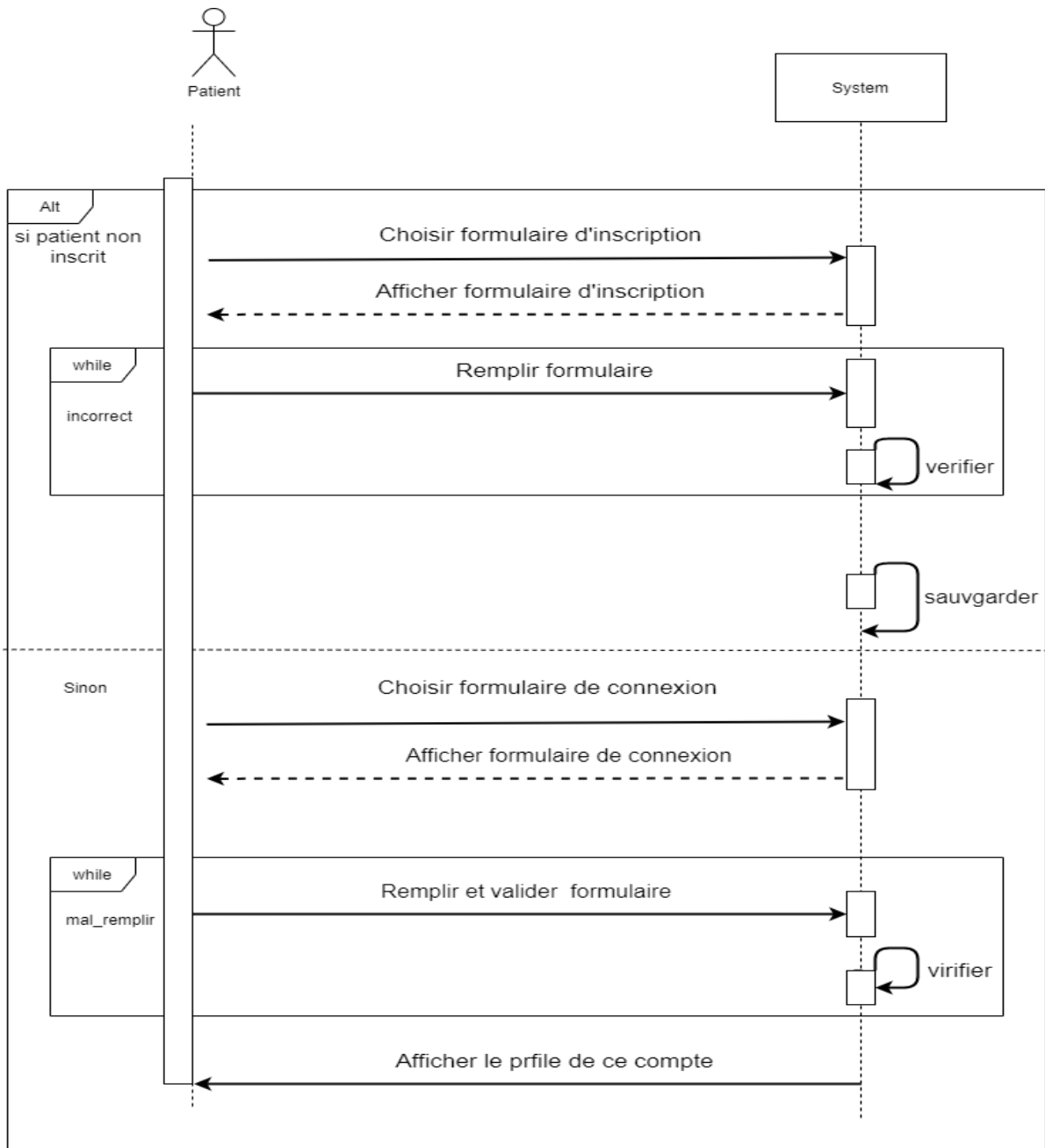


Figure 16: diagramme de séquence d'authentification de patient



## Chapitre III : Modélisation et conception

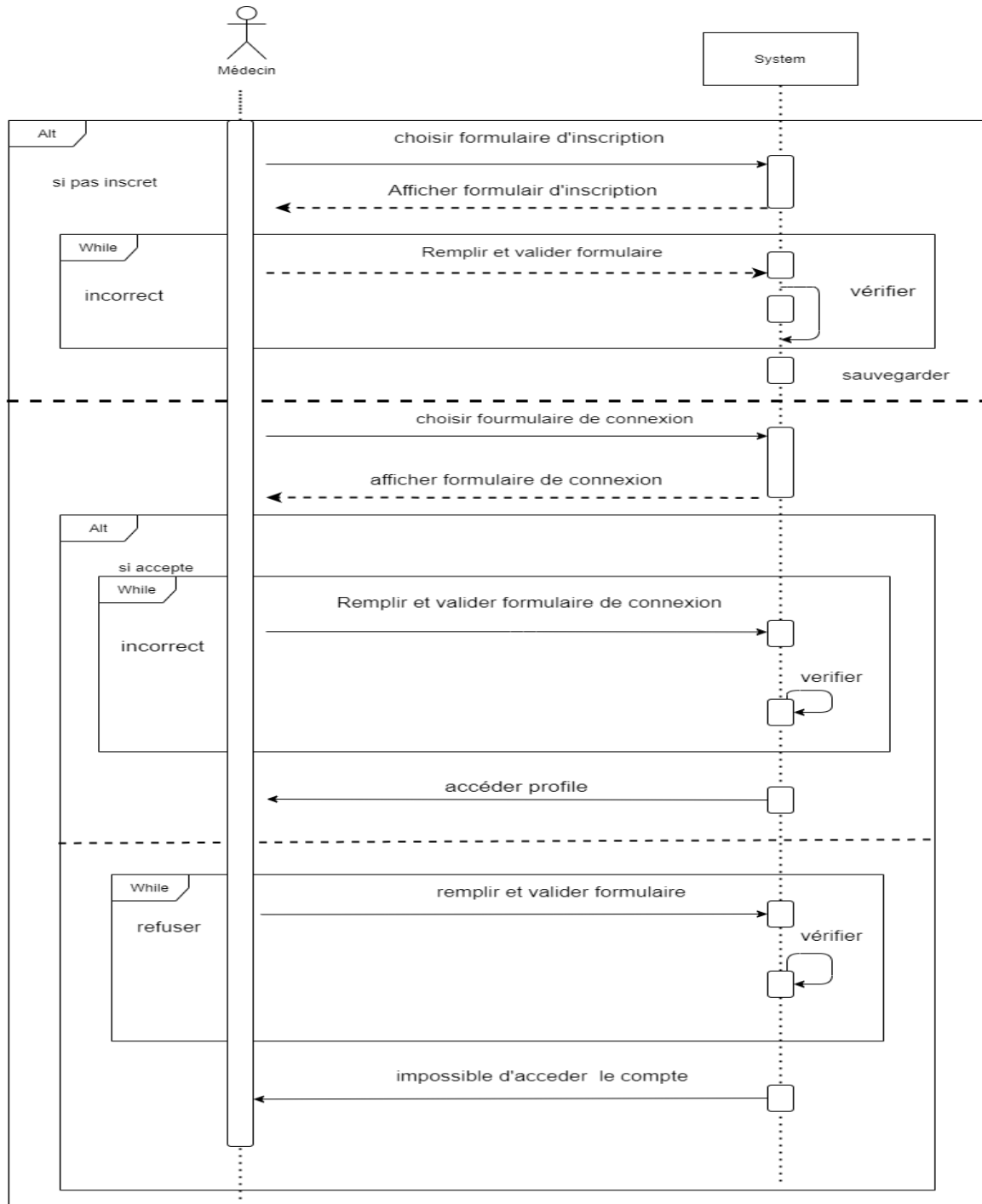


Figure 17: Diagramme de séquence d'authentification de médecin

## Chapitre III : Modélisation et conception

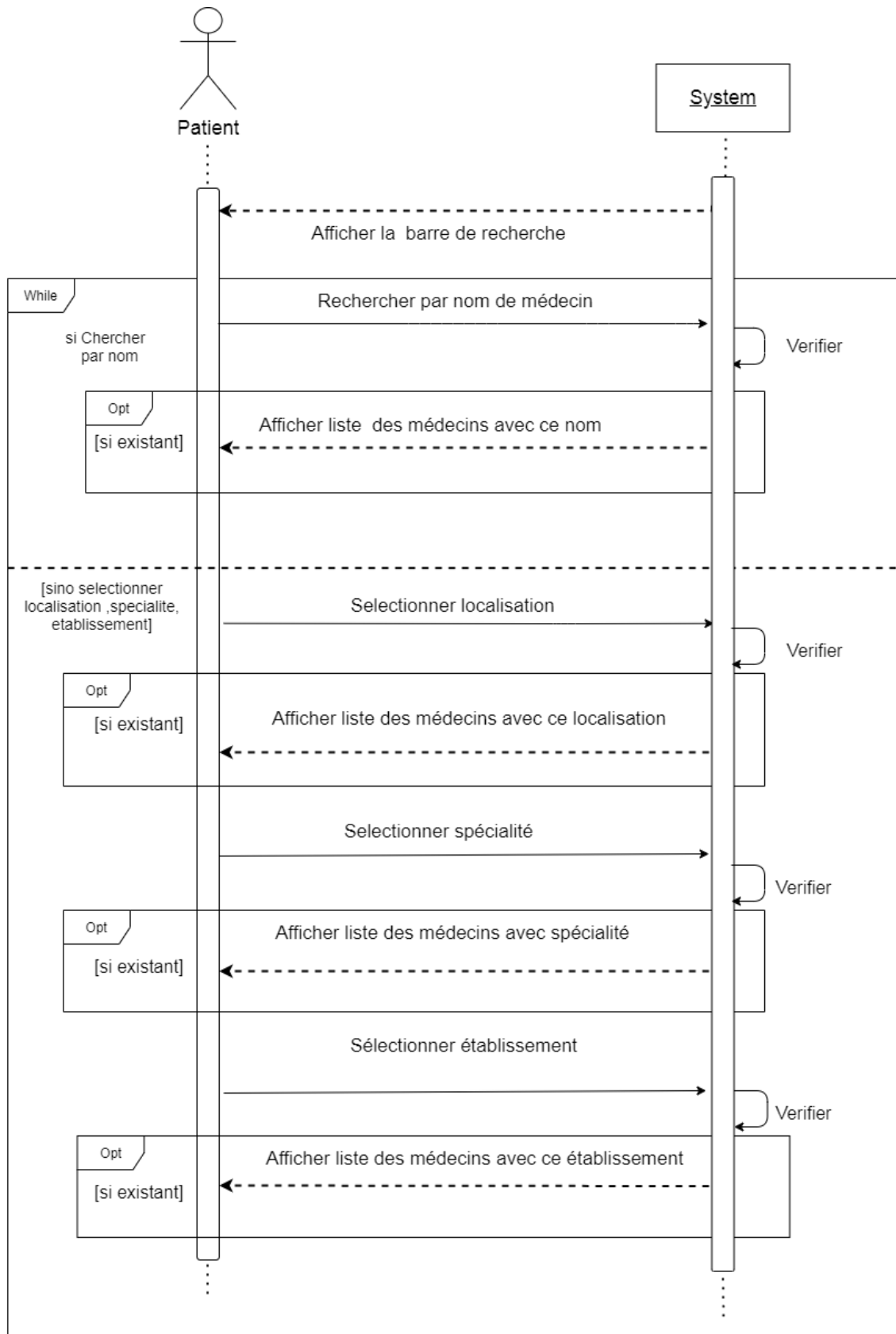


Figure 18: diagramme de séquence de recherche médecin

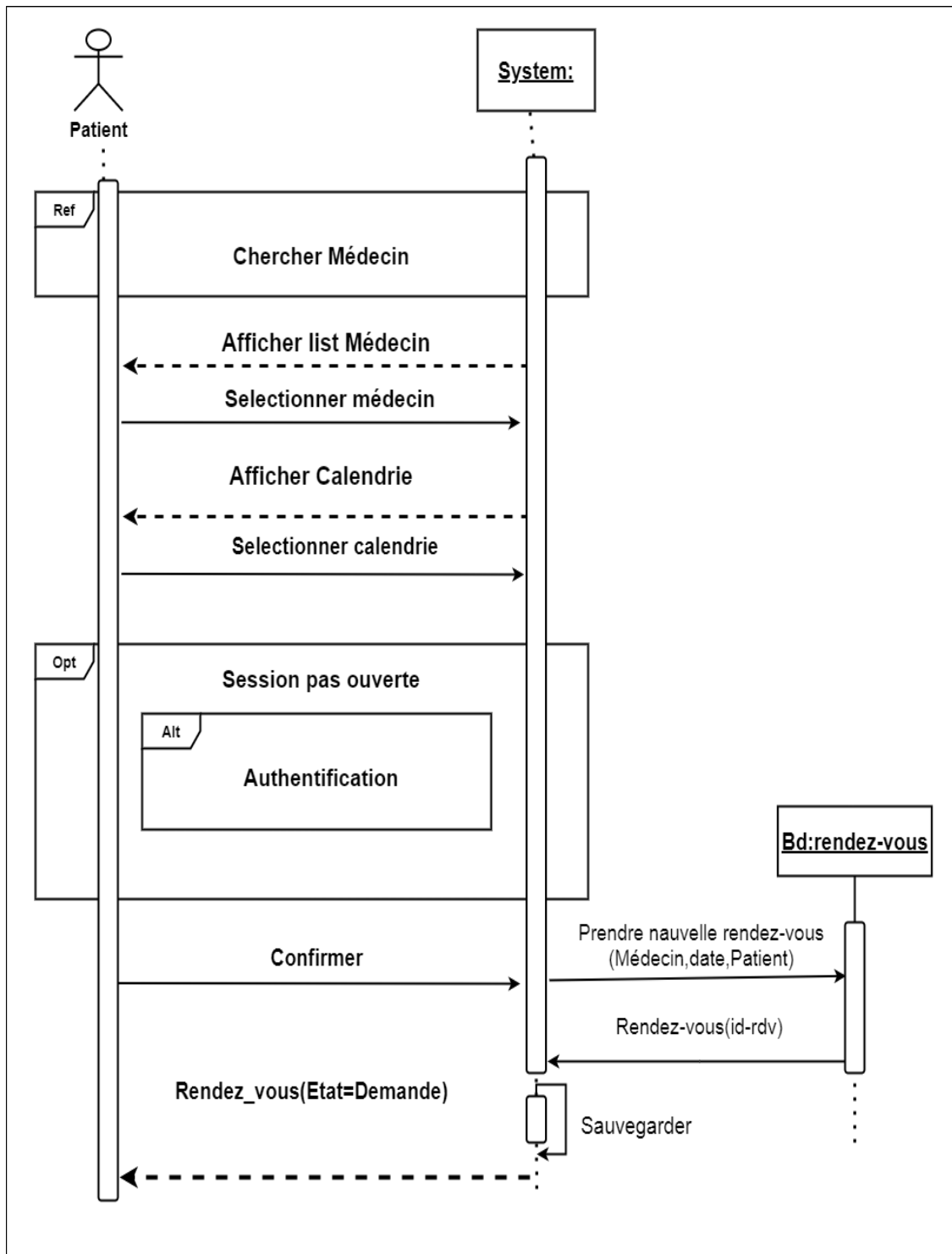


Figure 19: Diagramme de séquence pour prendre rendez-vous

# Chapitre III : Modélisation et conception

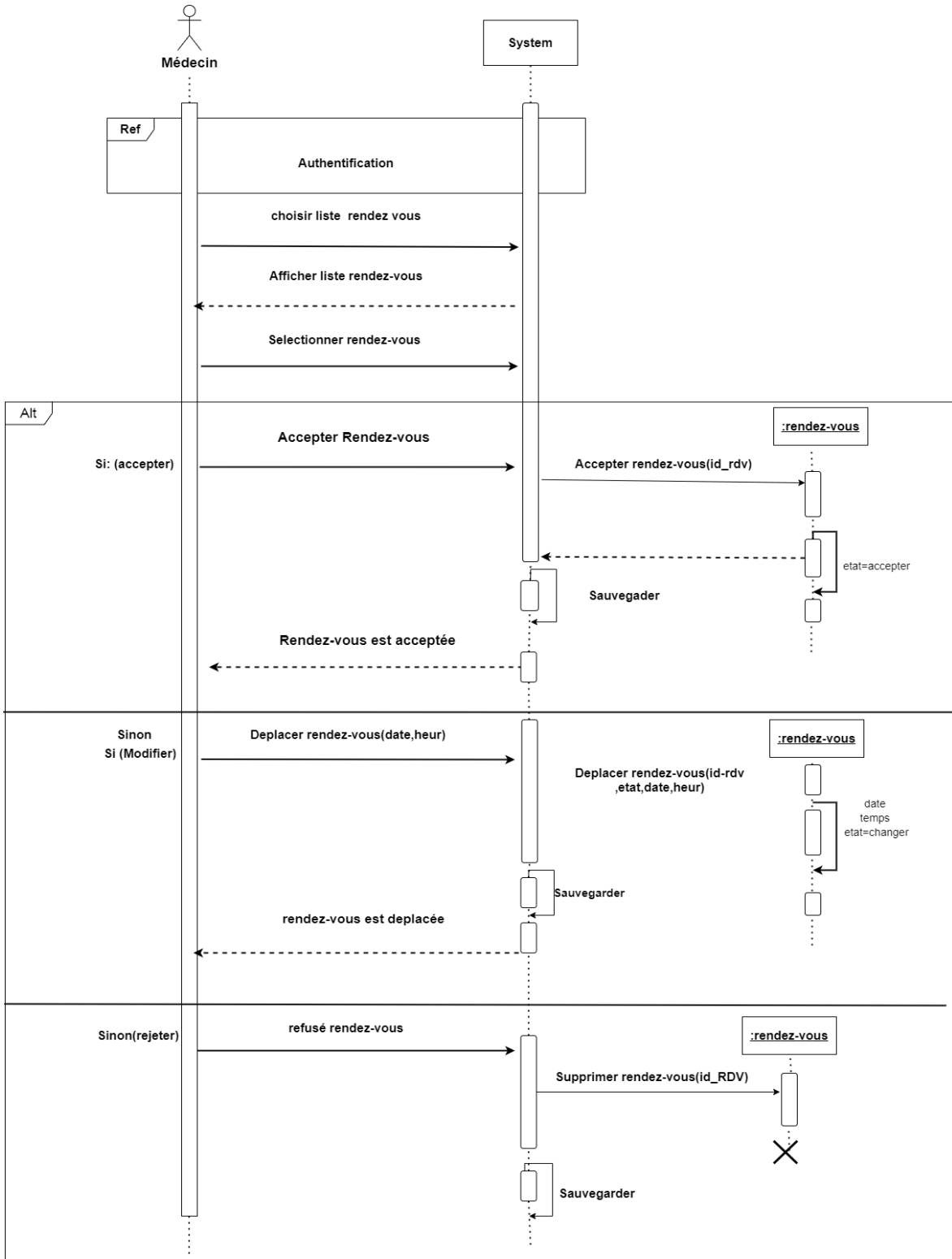
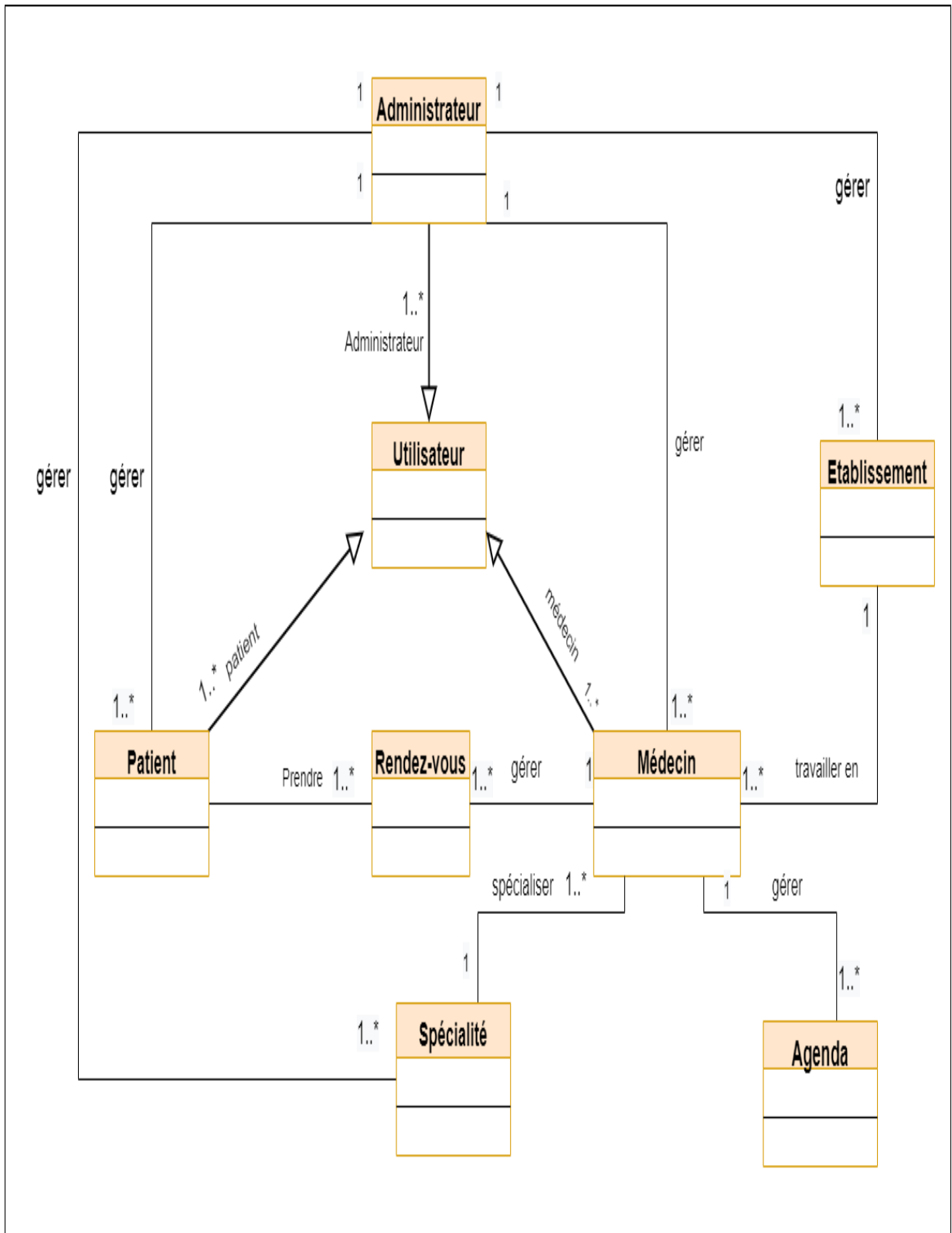


Figure 20:Diagramme de séquence de gérer rendez-vous

## Chapitre III : Modélisation et conception

### VII . Diagramme de classe

La figure suivante représente le diagramme de classes :



VIII .

Figure 21: Diagramme de classe

Conclusion

## **Chapitre III : Modélisation et conception**

---

Ce chapitre nous a permis de présenter notre système proposé, en particulier ses conceptions générales et détaillées. Nous avons choisi de concevoir notre plateforme en utilisant des diagrammes UML. Ce langage de modélisation a joué un rôle important dans la conception d'un programme simple et structuré. L'étape conceptuelle est une étape essentielle à la réalisation de tout projet qui rend la mise en œuvre de cette application plus simple et plus simple. Ainsi, cela nous a permis d'avancer rapidement et de mettre en œuvre la plateforme de manière plus judicieuse. Le chapitre suivant fournira un aperçu de la mise en œuvre de cette plate-forme proposée, décrira les différentes pages Web et le fonctionnement de cette application, et décrira les cadres que nous avons utilisés pour la mettre en œuvre.

# Chapitre IV : Implémentation

---

## I . Introduction

Ce chapitre représente le dernier volet de ce rapport, il sera consacré à l'implémentation de notre système. Nous commençons par définir quelques concepts, puis la présentation des ressources matérielles et logicielles utilisées. Nous passons ensuite à présenter des captures d'écran dans le but de mettre en évidence l'aspect ergonomique et fonctionnel des interfaces développées.

## II . Architecture du système et environnement de test

### II. 1 Définition d'une application web

En informatique, une application Web (aussi appelée Web App) est un logiciel applicatif manipulable grâce à un navigateur Web. De la même manière que les sites Web, une application Web est généralement placée sur un serveur et se manipule en actionnant des widgets à l'aide d'un navigateur Web, via un réseau informatique (Internet, intranet, réseau local, etc.). [8]

### II . 2 Présentation de l'architecture d'un système client/serveur

De nombreuses applications fonctionnent selon un environnement client/serveur, cela signifie que des machines clientes (des machines faisant partie du réseau) contactent un serveur, une machine généralement très puissante en termes de capacités d'entrée-sortie, qui leur fournit des services. Ces services sont des programmes fournissant des données telles que l'heure, des fichiers, une connexion, etc. Les services sont exploités par des programmes, appelés programmes clients, s'exécutant sur les machines clientes. On parle ainsi de client (client FTP, client de messagerie, etc.) lorsque l'on désigne un programme tournant sur une machine cliente, capable de traiter des informations qu'il récupère auprès d'un serveur (dans le cas du client FTP il s'agit de fichiers, tandis que pour le client de messagerie il s'agit de courrier électronique).[9]

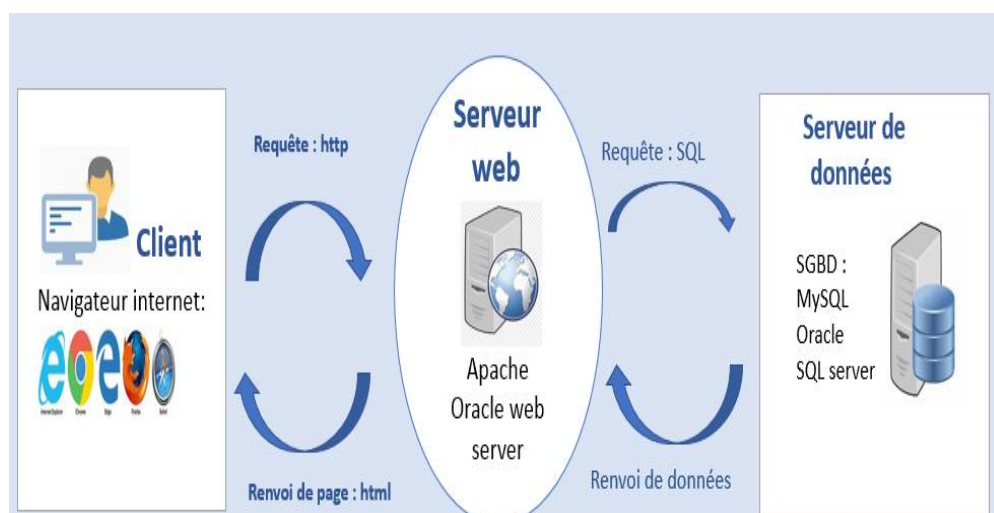


Figure 22: Architecture d'application web

### II . 3 Les différentes architectures client/serveur

#### II. 3.1 Architecture client/serveur à deux niveaux

L'architecture à deux niveaux (aussi appelée architecture 2-tier, tier signifiant rangée en anglais) caractérise les systèmes clients/serveurs pour lesquels le client demande une ressource et le serveur la lui fournit directement, en utilisant ses propres ressources. Cela signifie que le serveur ne fait pas appel à une autre application afin de fournir une partie du service.[10]

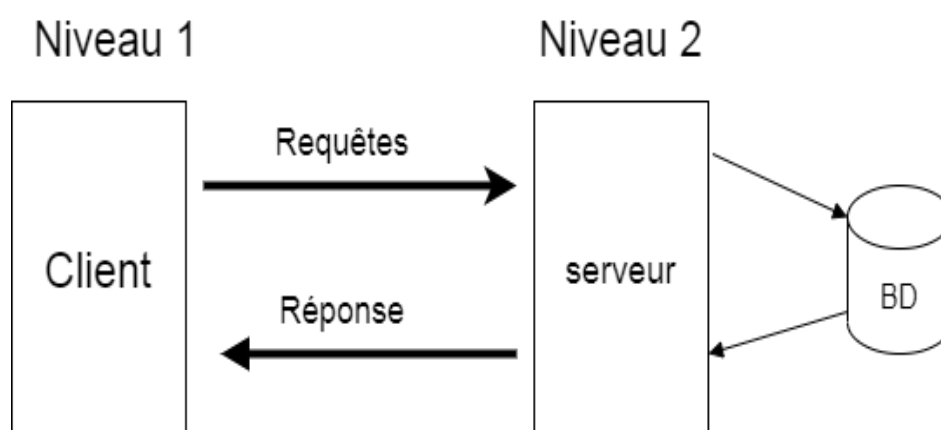


Figure 23:Figure : Présentation de l'architecture à 2 niveaux



## Chapitre IV : Implémentation

### a/ Avantages de l'architecture à deux niveaux :

- Facile à entretenir et la modification est facile.
- La communication est plus rapide.

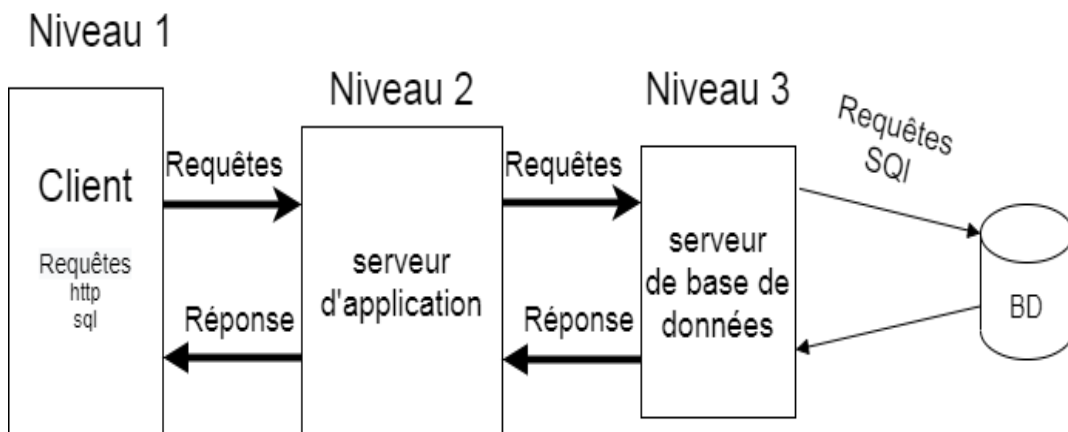
### b/ Inconvénients de l'architecture à deux niveaux

- Les performances des applications seront dégradées lors de l'augmentation du nombre d'utilisateurs.
- Coût inefficace.

### II .3.2 Architecture client/serveur à trois niveaux

Dans l'architecture à 3 niveaux (appelée *architecture 3-tier*), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre :

1. Un client, c'est-à-dire l'ordinateur demandeur de ressources, équipée d'une interface utilisateur (généralement un navigateur web) chargée de la présentation ;
2. Le serveur d'application (appelé également middleware), chargé de fournir la ressource mais faisant appel à un autre serveur
3. Le serveur de données, fournissant au serveur d'application les données dont il a besoin.



### a/ Avantages de l'architecture à trois niveaux

- Hautes performances : parce que le niveau Présentation peut mettre en cache les requêtes, l'utilisation est minimisée et la charge est réduite sur les niveaux Application et Données.
- Sécurité : le client n'a pas d'accès direct à la base de données.

Figure 24: Architecture client /serveur à deux eaux

## Chapitre IV : Implémentation

---

- Facile à entretenir : cela n'affectera pas les autres modules.

### **b/ Inconvénients de l'architecture à trois niveaux**

- Complexité et efforts élevés

### **II . 4 Architecture model-vue-contrôleur (MVC)**

**Définition :** modèle-vue-contrôleur (en abrégé MVC, de l'anglais model-view-contrôler), est un modèle destiné à répondre aux besoins des applications interactives en séparant les problématiques liées aux différents composants au sein de leur architecture respective.[11]

- **Ses avantages :**

- Séparation des compétences (design, base de données, application)
- Simplicité de mise à jour
- Vitesse de création de pages

- **Ce paradigme regroupe les fonctions nécessaires en trois catégories :**

- Un modèle (modèle de données),
- Une vue (présentation, interface utilisateur)
- Un contrôleur (logique de contrôle, gestion des événements, Synchronisation).

- **Nous expliquons ces trois parties l'une après l'autre :**

a) **Le modèle (ou Model) :** Le modèle représente les structures de données. Typiquement, les classes modèles contiennent des fonctions qui aident à récupérer, à insérer et à mettre à jour des informations de la base de données.

b) **La vue (ou View) :** La vue correspond à l'interface avec laquelle l'utilisateur interagit. Elle se présente sous la forme d'une Template représentant l'interface. Reprenons l'exemple de l'outil. Ce n'est pas le contrôleur qui affiche le formulaire, il ne fait qu'appeler la bonne vue. Si nous avons une vue formulaire, les balises HTML du formulaire d'édition de l'outil y seront et finalement le contrôleur ne fera qu'afficher cette vue sans savoir vraiment ce qu'il y a dedans. Donc en pratique, c'est le « designer

## Chapitre IV : Implémentation

» d'un projet qui travaille sur les vues. La séparation de vues et contrôleurs permet aux designers et aux développeurs PHP de travailler ensemble sans besoin de contact direct.

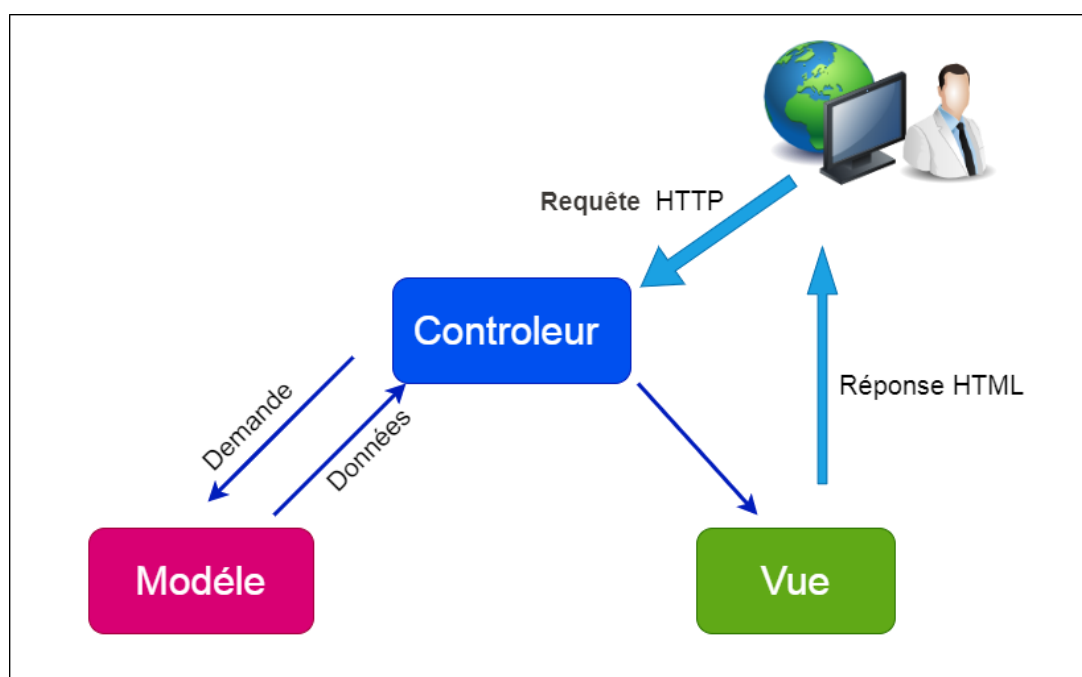


Figure 25: Architecture de MVS

### Le contrôleur

(ou Controller) :

Il gère l'interface entre le modèle et le client. Il va interpréter la requête de ce dernier pour lui envoyer la vue correspondante. Il effectue la synchronisation entre le modèle et les vues.

[11]

Il est la couche qui se charge d'analyser et de traiter la requête de l'utilisateur. Le contrôleur contient la logique de notre application et va se contenter « d'utiliser » les autres composants : les modèles et les vues. Concrètement, un contrôleur va récupérer, par exemple, les informations sur l'utilisateur courant, vérifier qu'il a le droit de modifier un tel outil, récupérer les données de cet outil et demander la page du formulaire d'édition de l'outil.

### II .5 Framework

Il existe de nombreux outils logiciels informatiques permettant de développer rapidement et efficacement des applications web enrichies (RIA, Rich Internet Application).

Parmi eux :

- Les CMS (content management system) : permettant la création de site disposant surtout.

## Chapitre IV : Implémentation

---

- Les Framework web ou plateformes applicatives.

Où nous nous sommes appuyés sur ce dernier pour développer notre projet.

### **a/ Définition Framework**

D'une manière générale, un Framework est un espace de travail modulaire, constitué d'un ensemble de bibliothèques, d'outils et de conventions permettant le développement rapide d'application. Il fournit aussi une structure pour l'application. Par extension, un Framework web est un ensemble d'outils spécialisés pour le développement d'applications web.[12]

### **b/ Pourquoi utiliser un Framework ?**

Les avantages des Framework sont nombreux. En effet, un Framework est portable, de la part de son abstraction de la base de données et de la gestion générique du cache. Les temps de développement avec un Framework sont réellement plus courts. Tous les outils essentiels sont déjà écrits

Le développement des applications sécurisées est facile. Grâce aux systèmes d'authentification, à la gestion des injections SQL ainsi qu'à la protection CSRF (Cross-Site Request Forgery) qui est gérée par la plupart des Framework. Les Framework sont des outils communautaires et ont, par conséquent, des forums, des listes de diffusion et des canaux IRC pour les soutenir.

De plus vu que les Framework sont largement déployés, la chance de trouver les correctifs des problèmes rencontrés est plus grande

Utiliser le langage PHP et passer par le modèle MVC signifie l'utilisation des outils convenables pour bien respecter ce modèle et bien mener nos travaux de développement. Les solutions proposées sur ce domaine sont les Framework et les CMS. Dans cette partie, nous allons mener une étude comparative pour bien choisir la solution qui répondra le plus à nos exigences et nos attentes. C CMS Framework Coût Réduit pour les projets personnels Elevé Compatible avec Les projets personnels essentiellement Tous les projets

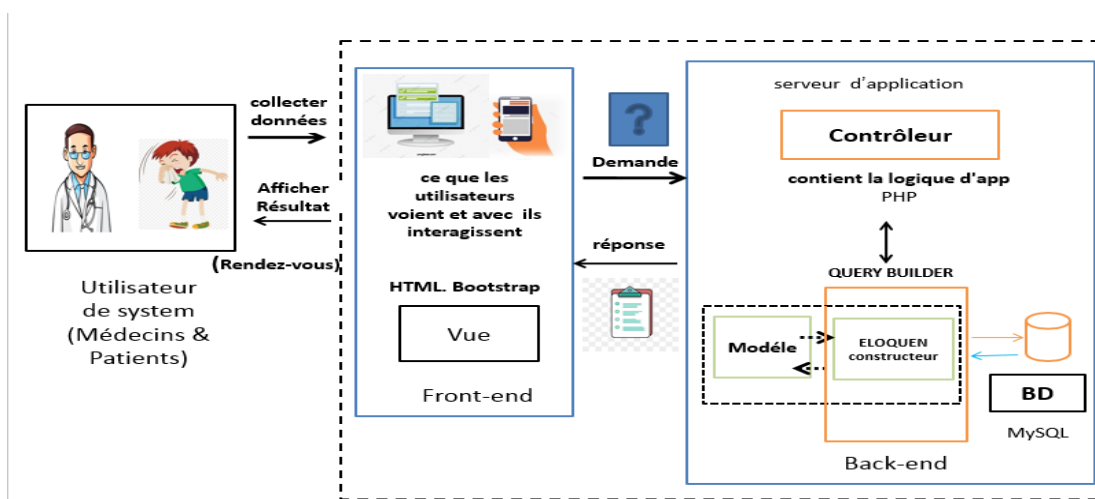
## Chapitre IV : Implémentation

**Tableau 8 : Comparaison entre CMS et Framework**

	CMS	Framework
<b>Coût</b>	Réduit pour les projets personnels	Elevé
<b>Compatible avec</b>	Les projets personnels essentiellement	Tous les projets
<b>Fonctionnalité</b>	Des fonctionnalités superflues pour le projet	Des APIs importantes un peu Compliqués
<b>Communauté</b>	Importante pour quelques CMS	Importante pour la plupart des Framework
<b>Extensibilité</b>	Très difficile de l'étendre	Toujours extensible

### II .6 Architecture générale du système

Notre application utilise une architecture à 2 niveaux ; parce que le serveur Web et le serveur de base de données sont identiques ; voici à quoi ressemble d'application Web comme la Figure suivante :



**Figure 26: Architecture générale de système**

## Chapitre IV : Implémentation

---

- L'utilisateur envoyer une requête à ce serveur d'applications Web (requête au contrôleur) sur Internet, via un navigateur Web ou l'interface utilisateur de l'application.
- Ce serveur d'applications effectue la tâche demandée telles que :
  - \_ l'interrogation de la base de données ou le traitement des données.
  - \_ Génère ensuite les résultats des données demandées.
- Ce serveur d'applications répond aux utilisateurs avec les informations demandées ou les données traitées qui apparaissent ensuite sur l'écran de l'utilisateur.

### III . Environnement de développement

Pour mettre en place notre système, nous avons utilisé un environnement de développement qui a assuré le bon déroulement de la phase implémentation. Cet environnement comporte des outils matériels ainsi que logiciels.

#### III .1. Environnement matériel

Pour le développement de notre application nous avons utilisé un PC portable DELL VOSTRO »

Processeur Intel Core i5-3537U avec fréquence 1.8 GHz.

Quantité de mémoire vive 8 Go (7,8 GB utilisable).

Type de system: Windows 7 Ultimate 64bit.

#### III .2 Environnement logiciel

##### III .2.1 Le Framework Laravel

a/ **Définition** : Laravel est un Framework web gratuit, open-source, écrit en PHP,

Créé par Taylor Otwell et destiné au développement d'applications Web suivant le modèle-vue-contrôleur (MVC). Et entièrement développé en programmation orientée objet. Le code source de Laravel est hébergé sur Git Hub sous licence conformément aux termes de licence MIT. [ 13]



## Chapitre IV : Implémentation

---

### b/ Structure du répertoire :

- **Application app :**

- Contient tout le code généré pour l'application.
- Il contient de nombreux autres répertoires qui sont détaillés après.

- **Ressources :**

- Contient, dans le dossier « views », l'ensemble des fichiers de vues gérés par défaut le moteur de Template de Laravel : [Blade](#).
- Contient également d'autres ressources tels que les fichiers non compilés [LESS](#), [SASS](#) ou [JavaScript](#).
- Contient les fichiers de langues dans le cas d'une application multilingue.

- **database :**

- Contient la migration de la base de données.

- **Routes :**

- Contient des fichiers listant l'ensemble des routes utilisées par l'application.
- Par défaut, plusieurs fichiers d'itinéraire sont inclus avec Laravel ([web.php](#) ; [api.php](#) ; [console.php](#))

### **Bootstrap :**

- **Config :**

- Contient l'ensemble des fichiers de configurations de l'application.

- **Public :**

- Point d'entrée unique de l'application par l'intermédiaire du fichier « index.php ».
- Seul répertoire visible de l'extérieur. Tous les autres sont cachés et protégés des attaques de l'extérieur.
- Contient également vos différents assets tels que les images, les fichiers CSS et JavaScripts compilés

- **Storage :**

- Contient des fichiers compilés automatiquement par le système.

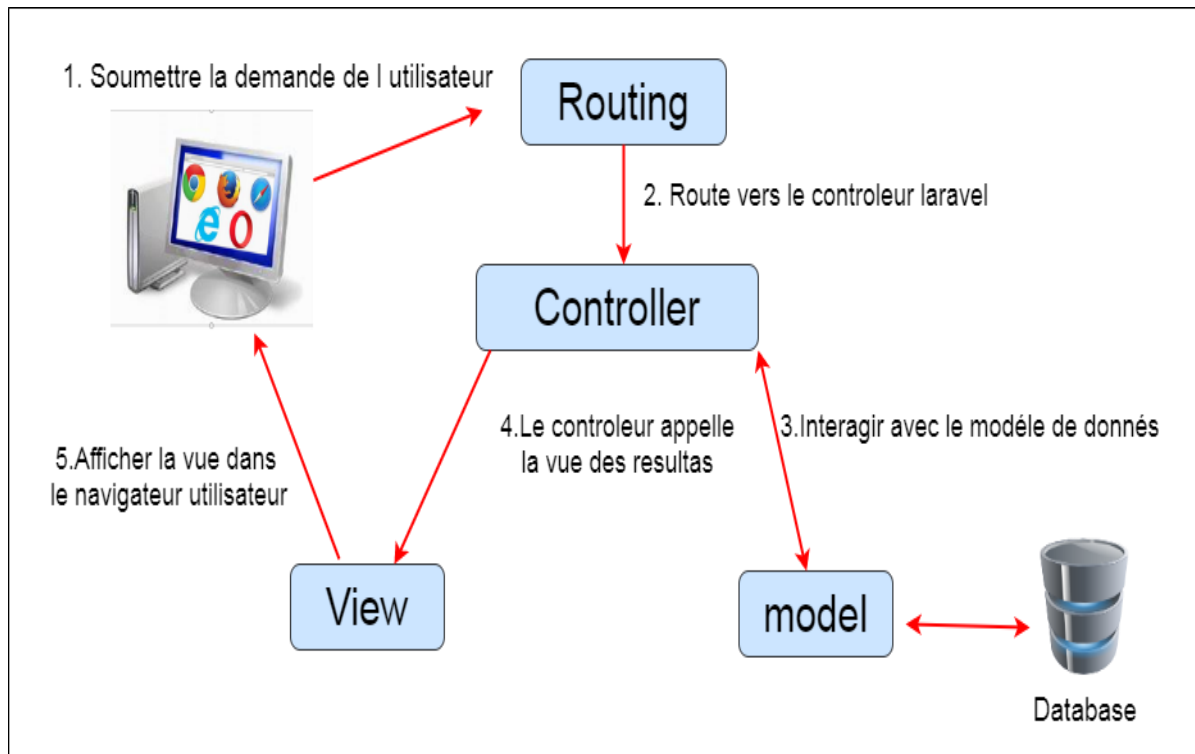
- **Tests :**

- Contient les tests. Laravel utilisé PHPUnit.

## Chapitre IV : Implémentation

- **Vendor :**
  - Contient les dépendances Composer associées à l'application.

### Architecture de Framework laravel



### III.2.2

### Framework

Figure 27 : Architecture de Framework laravel

### Bootstrap



**Définition :** est un Framework qui facilite et accélère Le développement Frontend. Il inclue une base CSS très complète (au format LESS) configurée à partir d'un fichier de variables, un ensemble de conventions de structure HTML et de nommage de classes des bibliothèques JavaScripts simples pour les fonctions les plus courantes. [14]



### III.3 Langages de développement :

#### a. HTML

Le langage HTML permet notamment la lecture de documents sur Internet à partir de machines différentes, grâce au protocole HTTP, permettant d'accéder via le réseau à des documents repérés par une adresse unique, appelée URL.[15]



#### b. PHP :

est un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale.



PHP est un langage\_impératif orienté\_objet.[16]

#### c. CSS :

Les CSS (Cascading Style Sheets en anglais, ou « feuilles de style en cascade ») sont le code utilisé pour mettre en forme une page web [17]



#### d. jQuery:

jQuery est une bibliothèque JavaScript libre développée initialement par John Régis et qui est aujourd'hui maintenue et mise à jour par la communauté jQuery Team. Le Framework JavaScript jQuery code rapidement et simplement des traitements à base de code JavaScript pour dynamiser et améliorer l'ergonomie des sites internet. [ 18]



## Chapitre IV : Implémentation

---

### e. MySQL :

MySQL est un système de gestion de base de données relationnelle (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle et Microsoft SQL Server. [ 19]



### III . 4 Outils de développement :

#### a. Visual studio code

Visual Studio Code (VSC par la suite) est un éditeur de code open-source, gratuit et multi-plateforme (Windows, Mac et Linux), développé par Microsoft, à ne pas confondre avec Visual Studio, l'IDE propriétaire de Microsoft. VSC est développé avec Electron et exploite des fonctionnalités d'édition avancées du projet Monaco Editor. Principalement conçu pour le développement d'application avec, PHP, JavaScript, TypeScript et Node.js, l'éditeur peut s'adapter à d'autres types de langages grâce à un système d'extension bien fourni.[20]



b. **Composer** : est un outil primordial pour le travail des programmeurs PHP. Ce gestionnaire de dépendances PHP est capable de vérifier et de signaler les composants et les bibliothèques qui manquent à votre projet web et la version requise, de les télécharger et de les installer de manière à éviter le travail dur.[21]



## Chapitre IV : Implémentation

---

### Xampp :

XAMPP est un ensemble de logiciels permettant de mettre en place un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres (X (cross) Apache Maria Perl PHP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et fonctionne, de plus, sur les systèmes d'exploitation plus répandus. [22]



## Chapitre IV : Implémentation

---

### IV . Présentation d'application web à travers quelques interfaces :

Nous allons faire une présentation de quelques interfaces qui sont intégrés à notre application web.

- **Page d'accueil** : la page d'accueil offre la possibilité de visualiser les différents services de notre site web comme elle est présentée dans la figure si dessous :

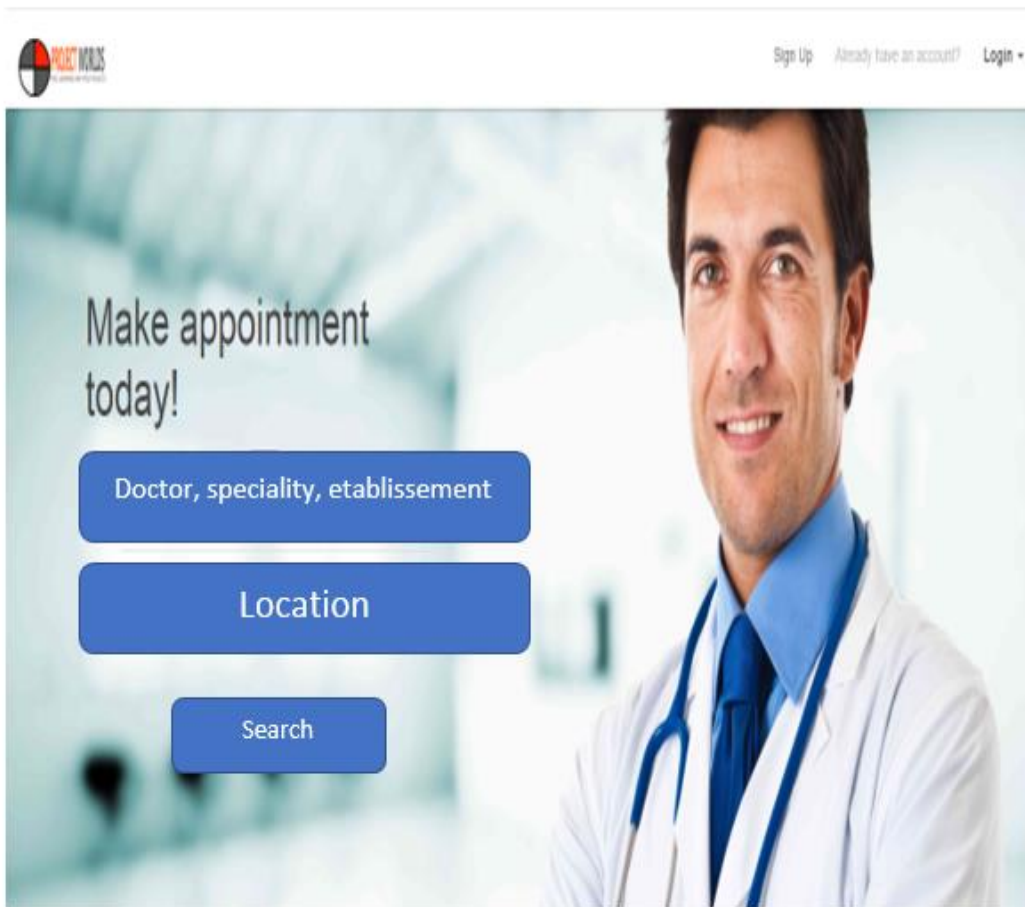
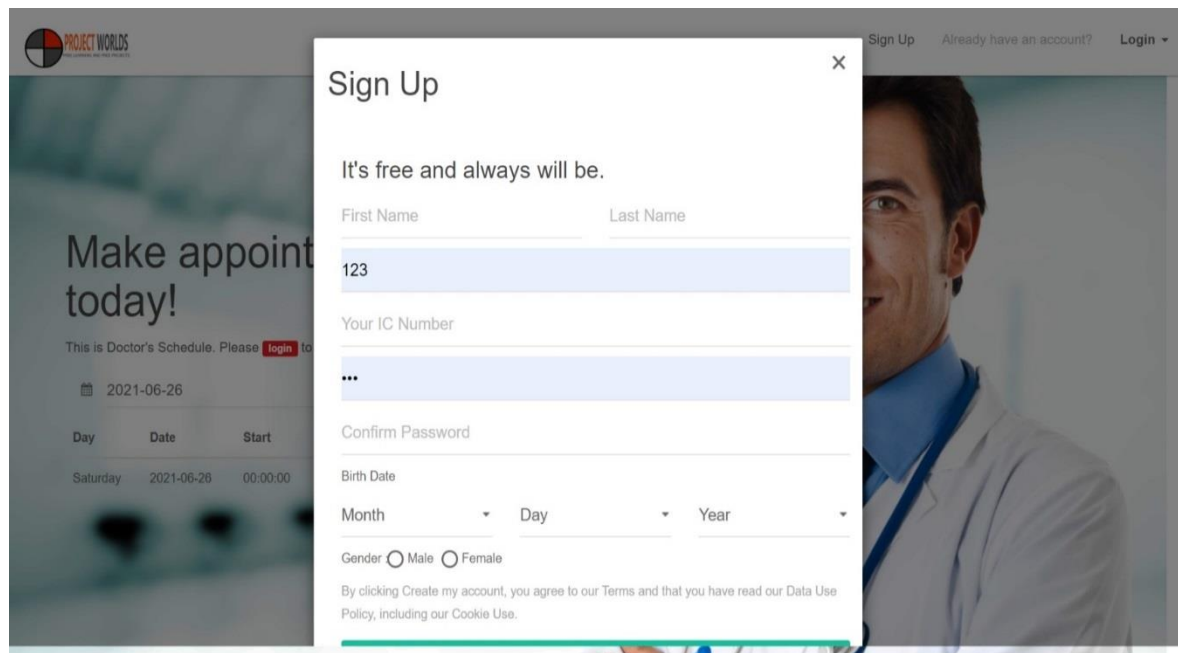


Figure 28:Interface de page d'accueil

**Inscription patient:** permet au patient de s'inscrire à notre site.



## Get in Touch

FEEL FREE TO DROP US A LINE TO CONTACT US



### Branding

Retro chillwave YOLO four loko photo booth. Brooklyn kale chips, seitan hella 3 wolf moon slow-carb paleo.



### Web Design

Retro chillwave YOLO four loko photo booth. Brooklyn kale chips, seitan hella 3 wolf moon slow-carb paleo.



### Social Marketing

Retro chillwave YOLO four loko photo booth. Brooklyn kale chips, seitan hella 3 wolf moon slow-carb paleo.



### SEO

Retro chillwave YOLO four loko photo booth. Brooklyn kale chips, seitan hella 3 wolf moon slow-carb paleo.



### Mobile Apps

Retro chillwave YOLO four loko photo booth. Brooklyn kale chips, seitan hella 3 wolf moon slow-carb paleo.



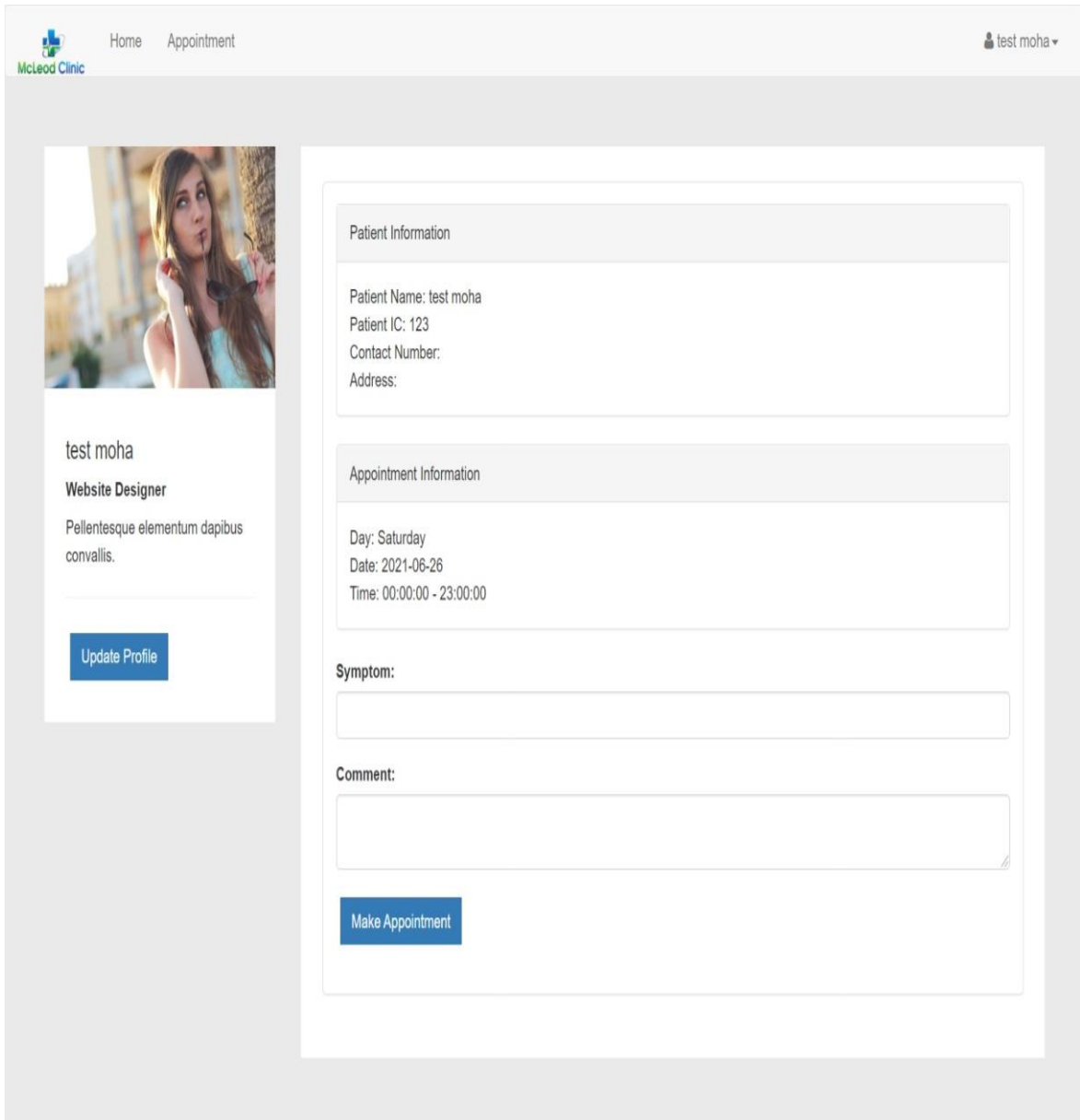
### Corporate Literature

Retro chillwave YOLO four loko photo booth. Brooklyn kale chips, seitan hella 3 wolf moon slow-carb paleo.

Figure 29: Interface d'inscription pour le patient

## Chapitre IV : Implémentation

### Profile de patient : Affiche au patient profile de sa compte



The screenshot displays a patient profile page on the McLeod Clinic website. The page is titled "Profile de patient : Affiche au patient profile de sa compte". The interface includes a navigation bar with "Home" and "Appointment" links, and a user profile dropdown for "test moha".

The main content area is divided into two columns. The left column features a profile picture of a woman, the name "test moha", the title "Website Designer", and a placeholder text "Pellentesque elementum dapibus convallis." Below this is a blue "Update Profile" button.

The right column contains a form with the following sections:

- Patient Information:** Fields for Patient Name (test moha), Patient IC (123), Contact Number, and Address.
- Appointment Information:** Fields for Day (Saturday), Date (2021-06-26), and Time (00:00:00 - 23:00:00).
- Symptom:** A text input field.
- Comment:** A larger text input field.

At the bottom of the form is a blue "Make Appointment" button.

Figure 30: Interface de profile patient

## Chapitre IV : Implémentation

Calendrier du médecin : affiche au patient le calendrier et les détails du médecin :

L

Doctor Schedule

Add Schedule

Date \*

Day \* Monday

Start Time \*

End Time \*

Availability \* available

Submit

List of Patients

scheduleId	scheduleDate	scheduleDay	startTime	endTime	bookAvail	
40	2015-12-13	Sunday	09:00:00	10:00:00	notavail	
41	2015-12-13	Sunday	10:00:00	11:00:00	available	
42	2015-12-13	Sunday	11:00:00	12:00:00	available	
43	2015-12-14	Monday	11:00:00	12:00:00	available	
44	2015-12-13	Sunday	01:00:00	02:00:00	available	
45	2021-06-26	Saturday	00:00:00	23:00:00	notavail	

UPDATE

Figure 31:Interface de Calendrier du médecin

## Chapitre IV : Implémentation

Les rendez-vous: Affiche au medecin et patient la liste de tous les rendez-vous

patient Ic	Name	Contact I	Email	Day	Date	Start	End	Status	Comple	Delete
123	moha		moh@test.com	Saturday	2021-06-26	00:00:00	23:00:00	process	<input type="checkbox"/>	<input type="checkbox"/>
920517105553	Mazlan	173567758	lan.psis@gmail.com	Sunday	2015-12-13	09:00:00	10:00:00	done	<input type="checkbox"/>	<input type="checkbox"/>

Figure 32: interfaces liste des rendez\_vous

Liste du patients: affiche liste de tous les patient qui prendre un rendez\_vous

patient Ic	Name	Password	ContactNo.	Gender	Status	Birthdate	Address	
123	moha	123		male		1994-09-13		<input type="checkbox"/>
920517105553	Mazlan	123	173567758	male	single	1992-05-17	NO 153 BLOK MURNI KOLEJ CANSELOR UNIVERSITI PUTRA MALAYSIA	<input type="checkbox"/>

Figure 33: liste du patients



### V . Conclusion

Au cours de ce chapitre, nous avons décrit les plateformes matérielles et logicielles sur lesquelles nous avons développé notre système. Nous avons ensuite présenté l'applications de notre système de gestion des rendez-vous médicaux à travers quelques interfaces que nous avons développées.

En effet, nous n'avons pas achevé l'implémentation et les tests de tous les cas d'utilisation,

## Conclusion générale

---

### Conclusion générale

Ce projet de fin d'études a pour objectif de réaliser un système de gestion des rendez-vous médicaux permettant d'une part aux patients de chercher un médecin et prendre un rendez-vous en utilisant internet, et d'autre part aux médecins de gérer les rendez-vous. La réalisation de ce travail a passé par trois phases essentielles. En premier lieu, nous avons effectué une étude préalable qui nous a permis d'identifier les besoins de notre application. En deuxième lieu, nous avons entamé la partie conception. Enfin, nous avons présenté les outils de développement adoptés et les résultats obtenus. Finalement, notre application peut être améliorée et enrichie par de nouvelles fonctionnalités comme

- Mètre de disposition des utilisateurs une riche base de données de médecins qu'ils puissent y avoir recours en cas de besoin.
- La téléconsultation
- La gestion des dossiers des patients

# Références

---

## Références bibliographiques

- [1] <https://www.linkedin.com/company/esiha>
- [2] <https://www.linkedin.com/company/dzdoc>
- [3] <https://www.linkedin.com/company/Rdvtoubib->
- [4] <http://wikimonde.com/article/Doctolib>
- [5] <https://www.maiia.fr/qui-sommes-nous>
- [6] <https://apluseduc.com/413-etude-conception>
- [7] <https://chrome.google.com/webstore/detail/diagramsnet/onlkggianjhjenigcpigpjehhpplldkc?hl=fr>
- [8] [ <http://dictionnaire.sensagent.leparisien.fr>
- [9] <https://stph.scenari-community.org/bdd/lap2/co/webUC002archi.html>
- [10] [http://jjpditchanou.free.fr/architecture\\_reseau\\_securite.pdf](http://jjpditchanou.free.fr/architecture_reseau_securite.pdf)
- [11] <https://fruanthony.wordpress.com/veille-technologie/le-modele-mvc/>
- [12] [http://edutechwiki.unige.ch/fr/Framework\\_web](http://edutechwiki.unige.ch/fr/Framework_web)
- [13] <http://bliaudet.free.fr/IMG/pdf/Laravel-Cours-Etape-1.pdf>
- [14] <https://contrib.spip.net/BootStrap-v2-pour-SPIP>
- [15] <https://argan-design.com/2018/11/09/le-langage-html>
- [16] <https://grafikart.fr/tutoriels/php-presentation-1112>
- [17] [https://developer.mozilla.org/fr/docs/Learn/Getting\\_started\\_with\\_the\\_web/CSS\\_basics](https://developer.mozilla.org/fr/docs/Learn/Getting_started_with_the_web/CSS_basics)
- [18] <https://apluseduc.com/414-la-realisation>
- [19] <https://sql.sh/sqbd/mysql>
- [20] [https://edutechwiki.unige.ch/fr/Visual\\_studio\\_code](https://edutechwiki.unige.ch/fr/Visual_studio_code)
- [21] [ <https://composer.fr.malavida.com>]
- [22] [ <https://fr.wikipedia.org/wiki/XAMPP>]

## Références

---

## Références

---