

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira - Béjaia -
Faculté des Sciences Exactes
Département d'Informatique



Mémoire de fin de cycle en vue de l'obtention d'un Master Professionnel
En Informatique

Options : Génie Logiciel - Intelligence Artificielle

Thème

**Conception et Réalisation d'une plateforme e-santé.
Au profit de l'entreprise « Tech-Instinct - Béjaïa »**

Réalisé par :

M^{lle} BEN BOUDAOU D Anissa

M^r BENNIA Anis

Soutenu le 22/09/2022 devant le jury composé de :

Présidente D^r KHOULALENE Nadjette

M.C.A

U.A/Mira Béjaïa,Algérie.

Examineur D^r MOKTEFI Mohand

M.C.A

U.A/Mira Béjaïa,Algérie.

Encadrant D^r KHANOUCHE E.Mohamed

M.C.A

U.A/Mira Béjaïa,Algérie.

Promotion 2022-2023.

- Remerciements -

Nous remercions ALLAH tout Puissant de nous avoir permis de mener à terme ce projet qui est pour nous le point de départ d'une merveilleuse aventure, celle du développement, source de remise en cause permanente et de perfectionnement perpétuelle.

*Qu'il nous soit permis de rendre un vibrant hommage à notre encadrant **Mr Mohammed Essaid Khanouche** et notre responsable de stage **Mr BELLATAF Youcef** pour avoir bien voulu superviser ce modeste travail et donner de leurs temps et de leurs intelligences à la réussite de ce projet qui pour nous représente un modèle de réussite et une source de motivation permanente, pour leurs disponibilités, et leurs sens aigüés de l'humanisme pédagogique.*

Nous profitons de cette tribune pour remercier les personnes qui de passage, ont pu nous apporter leurs contributions, que ce soit au niveau des idées qu'à celui des conceptions. Qu'elles trouvent ici l'expression de nos sincères reconnaissances.

Enfin nous remercions les membres du jury qui ont bien voulu accepter, et ce nonobstant, leur lourdes et exaltantes responsabilités pour procéder à l'évaluation de ce modeste travail.

Là où il y'a une Volonté, il y'a un chemin

- Dédicaces -



....On dédie ce travail :

A

nos chers parents, pour tous leurs sacrifices, leurs amours, leurs tendresses, leurs soutiens et leurs prières tout au long de nos études.

A

nos chères frères, nos chères sœurs pour leurs encouragements permanents, et leurs soutiens moral. A toutes nos familles pour leurs soutiens tout au long de notre parcours universitaire.

A

tous nos amis et tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Que Dieu les protège

★ ★ ★

Anis et Anissa

Table des matières

Table des matières	i
Table des figures	iv
Liste des Tableaux	vii
Liste des abréviations	viii
Introduction générale	1
I Contexte d'étude et Recueil des besoins	3
I.1 Introduction	3
I.2 État de l'art de la e-santé	3
I.2.1 Présentation des solutions existantes	4
I.3 Problématique	6
I.4 Objectifs	7
I.5 Intérêt de la solution par rapport aux existantes	9
I.6 Conclusion	9
II Méthodologie de conception et spécification des besoins	10
II.1 Introduction	10
II.2 Présentation de la méthode SCRUM	10
II.3 Langage de modélisation	11
II.3.1 le formalisme UML (Unified Modeling Language)	11
II.3.2 Différent types de diagrammes UML	12
II.4 Modélisation des besoins	12
II.4.1 Identification des acteurs	12
II.4.2 Diagramme de cas d'utilisation global	17
II.5 Pilotage d'un projet avec Scrum	18
II.5.1 Rôle et user stories	18
II.5.2 Planification des livrables (Releases)	19
II.5.3 Backlog-Product (Carnet de produit)	21
II.6 Conclusion	23

III RELEASE 1 : Étude des sprints du Livrable 1	24
III.1 Introduction	24
III.2 Étude du premier Sprint	24
III.2.1 Description textuelle du cas d'utilisation « Rechercher un Médecin »	25
III.2.2 Diagramme d'interaction du cas d'utilisation « Rechercher Médecin »	26
III.2.3 Description textuelle du cas d'utilisation « localiser un Médecin »	27
III.2.4 Diagramme d'interaction du cas d'utilisation « Localiser Médecin »	28
III.2.5 Diagramme de classes participantes du cas d'utilisation « Localiser Médecin »	29
III.2.6 Fiche Descriptive du cas d'utilisation « Inscription du Patient »	30
III.2.7 Diagramme d'interaction du cas d'utilisation « Inscription Patient »	31
III.2.8 Diagramme de classes participantes du cas d'utilisation Inscription Patient	32
III.2.9 Fiche Descriptive du cas d'utilisation « Authentification Patient »	32
III.2.10 Diagramme de classes participantes du cas d'utilisation « Authentification du Patient »	34
III.2.11 Diagramme d'interaction du cas d'utilisation « S'authentifier Patient »	35
III.3 Étude du deuxième Sprint : Espace Administrateur	36
III.3.1 Fiche Descriptive du cas d'utilisation « Gérer les Médecins »	36
III.3.2 Diagramme de classes participantes du cas d'utilisation « Ajouter un Médecin »	37
III.3.3 Diagramme d'interaction du cas d'utilisation « Gérer les Patients "	38
III.3.4 Diagramme de classes participantes du cas d'utilisation « Ajouter un Patient "	39
III.3.5 Fiche Descriptive du cas d'utilisation « Gérer les Médicaments »	40
III.3.6 Diagramme d'interaction du cas d'utilisation « Gérer les Médicaments »	41
III.3.7 Diagramme de classes participantes du cas d'utilisation « Rechercher et Modifier un Médicament "	42
III.3.8 Fiche Descriptive du cas d'utilisation « Gérer les Spécialités »	43
III.3.9 Diagramme d'interaction du cas d'utilisation « Gérer les Spécialités »	44
III.3.10 Diagramme de classes participantes du cas d'utilisation « Ajouter une spécialité "	45
III.4 Conclusion	45
IV RELEASE 2 : Étude des sprints du Livrable 2	46
IV.1 Introduction	46
IV.2 Étude du troisième Sprint	46
IV.2.1 Fiche Descriptive du cas d'utilisation « Prendre Rendez-vous Patient »	47
IV.2.2 Diagramme d'interaction du cas d'utilisation « Prendre Rendez-vous »	48
IV.2.3 Diagramme de classes participantes du cas d'utilisation « Prendre Rendez-vous par le patient "	49
IV.2.4 Fiche Descriptive du cas d'utilisation « Annuler Rendez-vous du patient »	50
IV.2.5 Diagramme d'interaction du cas d'utilisation « Annuler Rendez-vous »	51
IV.2.6 Fiche Descriptive du cas d'utilisation « Consulter son Agenda de Rendez-vous »	52
IV.2.7 Diagramme d'interaction du cas d'utilisation « Consulter Mon Agenda de Rendez-vous	53

IV.2.8	Fiche Descriptive du cas d'utilisation « Ajouter prochain Rendez-vous »	54
IV.2.9	Fiche Descriptive des cas d'utilisations « Annuler Rendez-vous et supprimer un rendez-vous »	55
IV.2.10	Diagramme de classes participantes du cas d'utilisation « Annuler Rendez-vous du Patient par le praticien »	56
IV.2.11	Fiche Descriptive de cas d'utilisation « Envoyer un email de Confirmation d'un Rendez-vous ou d'annulation »	57
IV.2.12	Fiche Descriptive du cas d'utilisations « Configurer son planning »	58
IV.2.13	Diagramme d'interaction du cas d'utilisation « Configurer son Planning"	58
IV.2.14	Fiche Descriptive du cas d'utilisation « Contacter l'administrateur »	59
IV.2.15	Diagramme d'interaction du cas d'utilisation "Prendre Rendez-vous"	60
IV.2.16	Diagramme d'interaction du cas d'utilisation « Gérer Rendez-vous »	61
IV.3	Étude du quatrième sprint	62
IV.3.1	Fiche Descriptive du cas d'utilisation « Ajouter une nouvelle consultation »	62
IV.3.2	Diagramme de classes participante pour le cas d'utilisation « Ajouter une consultation »	63
IV.3.3	Fiche Descriptive du cas d'utilisation « Supprimer une consultation »	64
IV.3.4	Fiche Descriptive du cas d'utilisation « Modifier une nouvelle consultation »	65
IV.4	Étude du Sprint 5	65
IV.4.1	Fiche Descriptive du cas d'utilisation « Consulter la liste des ordonnances »	66
IV.4.2	Fiche Descriptive du cas d'utilisation « Éditer Automatiquement l'ordonnance médicale »	67
IV.4.3	Diagramme d'interaction du cas d'utilisation « Éditer automatiquement l'ordonnance »	68
IV.4.4	Diagramme de classes participantes du cas d'utilisation « Imprimer Ordonnance »	69
IV.5	Étude du Sprint 6 : Dossier Médical	70
IV.5.1	Fiche Descriptive du cas d'utilisation « Consulter le dossier Médical du Patient »	70
IV.5.2	Fiche Descriptive du cas d'utilisation « Consulter le dossier Médical du Patient »	70
IV.5.3	Diagramme d'interaction du cas d'utilisation « Consulter le dossier Médical du patient	72
IV.5.4	Fiche Descriptive du cas d'utilisation « Configurer les droits D'accès au Dossier médical »	73
IV.5.5	Diagramme d'interaction du cas d'utilisation « Configurer les droits d'accès au dossier Médical	74
IV.6	Conclusion	74
V	Réalisation	75
V.1	Introduction	75
V.2	Diagramme de classes	75
V.3	Élaboration du Dictionnaire de Données	77
V.4	JPA (Java Persistence API)	79

V.4.1	Les relations avec JPA	79
V.4.2	Schéma Relationnel	0
V.5	Langages et environnement de développement	84
V.6	Les MicroServices	88
V.7	Architecture des Micro-Services	89
V.8	Sécurisation de l'accès à l'application grâce a l'authentification et l'autorisation	90
V.8.1	Jwt Json Web Token	91
V.8.1.1	La structure d'un JSON Web Token	91
V.8.1.2	Fonctionnement du JWT	94
V.8.2	L'Algorithme HS256	94
V.9	Présentation des interfaces	95
V.9.1	Interface d'accueil DAWINI	95
V.9.2	Interface Résultat de recherche de médecin	97
V.9.3	Interface de localisation du praticien	98
V.9.4	Interface d'inscription du médecin	98
V.9.5	Interface Inscription du patient	99
V.9.6	Interface d'accueil de l'administrateur	100
V.9.7	Interface gérer les médecins	101
V.9.8	Interface ajouter un médecin	101
V.9.9	Interface Rechercher médecin	102
V.9.10	Interface gérer les patients	102
V.9.11	Interface modifier un patient	103
V.9.12	Interface ajouter un patient	103
V.9.13	Interface gérer les médicaments	104
V.9.14	Interface Ajouter un médicament	104
V.9.15	Interface modifier un médicament	105
V.9.16	Interface gérer les spécialités	105
V.9.17	Interface ajouter une spécialité	106
V.9.18	Interface contactez nous	106
V.9.19	Interface Agenda du médecin	107
V.9.20	Interface confirmer un rendez-vous	107
V.9.21	Notification par email/rendez-vous confirmé par le médecin	108
V.9.22	Notification par email/rendez-vous annulé par le médecin	108
V.9.23	Interface de prise de rendez-vous du médecin	109
V.10	Conclusion	109
	Conclusion générale	110
	Bibliographie	113

Table des figures

II.1	Les rôles de la méthode SCRUM	11
II.2	Type de diagrammes UML version 2.5.1	12
II.3	Diagramme de cas d'utilisation du Visiteur de la plateforme	13
II.4	Diagramme de cas d'utilisation du patient	14
II.5	Diagramme de cas d'utilisation du médecin	15
II.6	Diagramme de cas d'utilisation de l'Administrateur	16
II.7	Diagramme de cas d'utilisation global.	17
II.8	Schéma des livrables 1 et 2	20
II.9	Logo de la plate-forme DAWINI	23
III.1	Diagramme d'interaction de cas d'utilisation « Rechercher Médecin »	26
III.2	Diagramme d'interaction de cas d'utilisation « Localiser Médecin »	28
III.3	Diagramme d'interaction de cas d'utilisation « Rechercher et Localiser un médecin »	29
III.4	Diagramme d'interaction de cas d'utilisation « Authentification Patient »	31
III.5	Diagramme d'interaction de cas d'utilisation « Inscrire Patient »	32
III.6	Diagramme d'interaction de classes participantes « Authentification du Patient »	34
III.7	Diagramme d'interaction de cas d'utilisation « Authentification Patient »	35
III.8	Diagramme de classes participantes du cas d'utilisation « Ajouter un Médecin »	37
III.9	Diagramme d'interaction « Gérer les Patients »	38
III.10	Diagramme de classes participantes « Ajouter un patient »	39
III.11	Diagramme d'interaction « Gérer les Médicaments »	41
III.12	Diagramme de classes participantes « Rechercher et modifier un médicament »	42
III.13	Diagramme d'interaction « Gérer les Spécialités »	44
III.14	Diagramme de classes participantes « Ajouter une spécialité »	45
IV.1	Diagramme d'interaction de cas d'utilisation « Prendre Rendez-vous Patient »	48
IV.2	Diagramme de classes participantes « Prendre Rendez-vous par le patient »	49
IV.3	Diagramme d'interaction de cas d'utilisation « Annuler Rendez-vous »	51
IV.4	Diagramme d'interaction du cas d'utilisation « Consulter mon agenda de Rendez-vous »	53
IV.5	Diagramme de classes participantes « Annuler Rendez-vous du Patient par le praticien »	56
IV.6	Diagramme d'interaction « Configurer son planning »	58

IV.7	Diagramme d'interaction de cas d'utilisation "Prendre Rendez-vous"	60
IV.8	Diagramme d'interaction de cas d'utilisation « Gérer Rendez-vous »	61
IV.9	Diagramme de classes participantes de cas d'utilisation « Ajouter une consultation »	63
IV.10	Diagramme d'interaction du cas d'utilisation « Éditer automatiquement l'ordonnance »	68
IV.11	Diagramme de classes participantes de cas d'utilisation « Imprimer Ordonnance »	69
IV.12	Diagramme d'interaction du cas d'utilisation « Consulter le dossier Médical du patient »	72
IV.13	Diagramme d'interaction du cas d'utilisation « Configurer les droits d'accès au dossier médical »	74
V.1	Diagramme de classes	76
V.2	Architecture de notre application	84
V.3	Représentation des couches de spring boot	85
V.4	API REST Representational State Transfer	89
V.5	Architecture Micro-Service	90
V.6	Les fonctionnalités de sécurisation	90
V.7	La structure d'un Json web token	91
V.8	Fonctionnement du HS256	94
V.9	Page d'accueil- Rechercher praticiens	95
V.10	Page d'accueil- Section 3	96
V.11	Page d'accueil- Footer	97
V.12	Page Résultat de recherche de médecin	97
V.13	Page Localisation du médecin	98
V.14	Inscription médecin (1)	98
V.15	Inscription médecin (2)	99
V.16	Inscription patient (1)	99
V.17	Inscription patient (2)	100
V.18	Page d'accueil- Espace Admin	100
V.19	Gestion des médecins- Espace Admin	101
V.20	Ajouter un médecin- Espace Admin	101
V.21	Rechercher un médecin par nom- Espace Admin	102
V.22	Gestion des patients- Espace Admin	102
V.23	Modifier un Patient- Espace Admin	103
V.24	Ajouter un patient- Espace Admin	103
V.25	Consulter la liste des médicaments- Espace Admin	104
V.26	Ajouter un médicament- Espace Admin	104
V.27	Modifier un Médicament- Espace Admin	105
V.28	Liste de spécialité- Espace Admin	105
V.29	Ajouté une spécialité- Espace Admin	106
V.30	Page Contactez nous	106

V.31 Page Agenda du Médecin - Annuler Rendez-vous	107
V.32 Page Agenda du Médecin - Confirmer Rendez-vous	107
V.33 Notification par email de la confirmation du Rendez-vous	108
V.34 Notification par email de l'annulation du Rendez-vous	108
V.35 Page de Prise de Rendez-vous pour Médecin	109
V.36 Diagramme d'interaction de cas d'utilisation « Inscription Médecin »	111

Liste des tableaux

I.1	Présentation des applications Doctolib et DzDoc.	6
II.1	Répartition des Rôles.	18
II.2	Quelques Exemples de User Stories.	19
II.3	Le Carnet de Produit.	22
III.1	Description textuelle du cas d'utilisation « Rechercher Médecin ».	25
III.2	Description textuelle du cas d'utilisation « Localiser Médecin »	27
III.3	Description textuelle du cas d'utilisation « S'inscrire Patient »	30
III.4	Description textuelle du cas d'utilisation « S'authentifier Patient »	33
III.5	Description textuelle du cas d'utilisation « Gérer les Médecins »	37
III.6	Description textuelle du cas d'utilisation « Gérer les Médicaments »	40
III.7	Description textuelle du cas d'utilisation « Gérer les Spécialités »	43
IV.1	Description textuelle du cas d'utilisation « Prendre Rendez-vous Patient »	47
IV.2	Description textuelle du cas d'utilisation « Annuler Rendez-vous Patient »	50
IV.3	Description textuelle du cas d'utilisation « Consulter son Agenda de Rendez-vous »	52
IV.4	Description textuelle du cas d'utilisation « Ajouter un prochain Rendez-vous »	54
IV.5	Description textuelle des cas d'utilisation « supprimer un Rendez-vous de la part du praticien »	55
IV.9	Description textuelle du cas d'utilisation « Ajouter une nouvelle consultation ».	63
IV.10	Description textuelle du cas d'utilisation « Supprimer une consultation ».	64
IV.11	Description textuelle du cas d'utilisation « Modifier une consultation ».	65
IV.12	Description textuelle du cas d'utilisation « Consulter la liste des ordonnances. ».	66
IV.13	Description textuelle du cas d'utilisation « Éditer une Ordonnance Médicale »	67
IV.14	Description textuelle du cas d'utilisation « Consulter le dossier Médical du Patient »	71
IV.15	Description textuelle du cas d'utilisation « Configurer les droits d'accès au dossier Médical »	73
V.1	Dictionnaire de données.	79
V.2	Description textuelle du cas d'utilisation « Inscription Médecin »	112

Liste des abréviations

UML	Unified Modeling Language
ORM	Object Relational Mapping
JPA	Java Persistence Api
API	Application Programming Interface
JWT	Json Web Token
DMP	Dossier Medical Personnel
RSA	Rivest Shamir Adleman
HTTPS	Hypertext Transfert Protocol Secure
IOS	Interneetwork Operating System
REST	Representational State Transfer
HTTP	Hypertext Transfert Protocol
OMT	Object Modeling Technique
OOSE	Object Oriented Software Engineering
SGBDOO	Système de Gestion de Bases de Données Orienté Objet
SHA	Secure Hash Algorithm
ECDSA	Elliptic Curve Digital Signature Algorithm
JWE	Json Web Encryption
JWS	Json Web Signature
OTP	One Time Password

Introduction générale

De nos jours, le digital s'insère progressivement dans les pratiques de l'ensemble des industries, notamment celles des industries de santé. La e-santé englobe l'ensemble des innovations s'appuyant sur les technologies de l'information et de la communication pour la santé afin de faciliter le transfert et l'échange à distance de données en matière de santé, que ce soit à des fins d'information, de formation, de diagnostic, de traitement, de recherche ou de gestion. Elle impacte le monde médical à plusieurs niveaux allant de la prévention jusqu'à la guérison des patients.

L'utilisation de l'e-santé en Algérie peut apporter énormément au secteur de la santé, en effet avec le manque avéré de médecins dans la plupart des régions des hauts plateaux ainsi que les régions du sud, utiliser l'e-santé peut apporter des bénéfices non négligeables non seulement aux patients mais également à leurs familles pour éviter les déplacements pour avis, ou expertise, ou en évitant d'éventuelles erreurs de diagnostic, ou encore de médication.

Par exemple, en assistant le médecin lors de la prescription de médicaments, afin d'éviter d'éventuelles erreurs ou des interactions dangereuses. Les données à caractère médical sont hautement sensibles, et sujettes au secret médical. La protection de ces données doit être garantie, afin de garantir le respect des droits des personnes à la vie privée. Les trois critères à prendre en considération dans la sécurisation des données médicales sont la confidentialité, l'intégrité et la disponibilité.

Avec la renommée grandissante des agendas médicaux en ligne, de plus en plus de médecins se tournent vers ce type de logiciel, dont ils ont tant entendu parler. Parmi eux, certains utilisent encore l'agenda papier et souhaitent se moderniser. D'autres recherchent activement une solution pour eux-mêmes, mais aussi pour échanger plus facilement avec leurs collaborateurs.

Pour plus de souplesse le médecin généraliste par exemple peut consulter et modifier son agenda, où qu'il soit (en visite, chez lui, et même en vacances), à n'importe quelle heure. Cela lui apporte une grande souplesse dans son organisation de travail. Il peut également consulter le programme de sa journée le matin avant de l'entamer : il a une visibilité sur son organisation quotidienne. Il peut se mettre en condition ou se préparer pour certaines consultations plus complexes.

L'agenda médical en ligne permet de faire baisser l'absentéisme, par l'envoi de messages automatisés soit de confirmation ou d'annulation de rendez-vous (sms, par email et même par téléphone). Il permet aussi de proposer à ses patients de pouvoir réserver, par eux-même, un créneau disponible. Cette réservation peut avoir lieu par internet, sur une plateforme e-santé.

De même, La version papier des ordonnances va bientôt céder sa place à l'ordonnance en ligne et électronique pour obtenir automatiquement et facilement les prescriptions médicamenteuses.

Vu la difficulté et la délicatesse de telles tâches, l'objectif de ce travail est de concevoir et réaliser une

plateforme e-santé qui englobe toutes ces fonctionnalités.

La plate-forme DAWINI a pour but de répondre aux objectifs suivants :

Pour le praticien :

- La Réalisation de l’agenda en-ligne destiné aux médecins de différentes spécialités ;
- Permettre au médecin de configurer son planning selon ses horaires d’ouvertures et de fermetures ;
- La Facilité de prise de prochains Rendez-vous, ainsi que la suppression et l’annulation de RDV avec envoi automatique d’email, de confirmation ou d’annulation ;
- La création automatique du dossier médical sécurisé du Patient et la possibilité de consultation et de partage entre médecins avec une configuration de droit d’accès ;
- L’automatisation de l’édition de l’ordonnance ;
- La consultation de l’historique des patients (Consultations, Ordonnances, prochains rendez-vous...etc).

Pour le patient :

- La Facilité de prise de Rendez-vous, ainsi que de recherche et de localisation de médecins de différentes spécialités ;
- Consulter la liste des prochains RDVs ;
- Consulter historique de ses ordonnances ;
- Consulter son dossier Médical.

Pour cela nous avons organisé notre rapport comme suit :

Dans le premier chapitre « Contexte d’étude et Recueil des besoins » dont nous allons donner une description détaillés de l’étude de l’existant du domaine de la e-santé afin d’automatiser quelques aspects et présenter quelques plateformes existantes au niveau européen et algérien.

Puisque nous avons suivi la méthode SCRUM, Le deuxième chapitre « Méthodologie de conception et spécification des besoins » sera consacré pour présenter cette méthode ainsi que le langage de modélisation utilisé et les spécifications des besoins, nous allons identifier les utilisateurs et les acteurs ainsi d’établir le diagramme de cas d’utilisation pour chaque acteur puis le diagramme de cas d’utilisation global.

Nous présenterons dans le troisième et le quatrième chapitre l’étude des livrables selon les sprints planifié, les descriptions textuelle, la réalisation des diagrammes de classes participantes, et d’interaction pour chaque sprint.

Dans Le cinquième chapitre nous allons présenter le diagramme classes, dictionnaire de données et passage au relationnel, l’implémentation de cette plateforme, nous présentons les outils et les langages utilisés ainsi que des captures pour les différentes pages de nos applications. Nous terminerons par une conclusion et quelques perspectives.

Chapitre I

Contexte d'étude et Recueil des besoins

I.1 Introduction

Le secteur de la e-santé, produit de la digitalisation active de notre société, elle se présente comme la rencontre des secteurs de la santé et du numérique. Porté par l'usage des nouvelles technologies de l'information et de la communication. Concrètement, il apporte de la valeur ajoutée aux métiers et aux acteurs de la santé, implémente des solutions qui améliorent nos quotidiens et fait ainsi progresser notre système de santé comme notre qualité de vie à tous.

Durant notre étude, on essaye d'analyser le domaine de l'e-santé afin de recueillir le maximum d'informations sur ce dernier.

Ce chapitre décrit : l'étude de l'existant de notre travail, qui va nous servir à poser les bases des besoins du système à réaliser, il consiste à effectuer un premier repérage des besoins fonctionnels et opérationnels.



I.2 État de l'art de la e-santé

Notre Mission dans le cadre de ce projet est de créer une plateforme e-santé permettant de gérer le secteur de la santé qui propose des services aux médecins et à leurs patients, il s'agit de définir les responsabilités de la gestion, mettre à jour les données, organiser des données collectées, d'automatiser des fonctionnalités, de renforcer la sécurité, le contrôle et la confrontation, assurer une meilleure gestion médicale et une cohérence de l'information et enfin faciliter le travail des responsables de santé de différentes spécialités. Notre application aura comme principale fonctionnalités :

- ✓ Rechercher des médecins par nom, spécialité et par API Google maps et permettre aux patients et aux visiteurs de l'application de choisir un praticien, en fonction de sa localisation ;
- ✓ La Prise de rendez-vous en ligne ;
- ✓ Agenda en ligne du médecin ;
- ✓ Automatiser l'édition de l'ordonnance ;
- ✓ Informatiser le dossier médical du patient et renforcer sa sécurité ainsi que son partage entre praticiens.

I.2.1 Présentation des solutions existantes

Après des recherches sur les plateformes e-santé existantes, les plateformes intéressantes qui nous inspire sont Doctolib (Plateforme européenne) et DzDoc (Plateforme algérienne) :

Applications		
Description	<p>Doctolib permet de prendre des RDVs directement en ligne, avec des professionnels de santé 24h/24 et 7j/7. Elle offre aussi la possibilité de réserver une consultation vidéo (téléconsultation) [10].</p>	<p>DZDOC est une application web/mobile qui facilite la recherche d'un médecin ou clinique et aussi la prise de RDV gratuitement et en ligne [1]. Nous trouvons dans l'espace conseils santé des articles relatifs aux maladies, nutritions, forme / bien être et conseils des docteurs. DZDOC assiste ses utilisateurs avec des vidéos explicatives sur comment créer un compte, comment prendre un RDV et comment réinitialiser le mot de passe.</p>
Fonctionnalités	<ul style="list-style-type: none"> — Recherche de médecin, cliniques... etc — Réservation des RDVs en ligne — Téléconsultation (réserver une consultation en vidéo) — Convertir un RDV présentiel en consultation vidéo — Notification (SMS/ e-mail) — Géolocalisation 	<p>Cette application possède plusieurs fonctionnalités notamment :</p> <ul style="list-style-type: none"> — Recherche (médecin ou clinique) — Réservation (prise de RDV en ligne) — Géolocalisation — Notification (recevoir un rappel de RDV par email ou SMS contenant toutes les informations nécessaires la veille de votre RDV)

	<ul style="list-style-type: none"> — Agenda en ligne (permet de remettre un RDV annulé à la dernière minute disponible sur internet, ce qui permet à de nouveaux patients de le sélectionner et aussi elle s'occupe de leur confirmer et de leur rappeler les RDV). — Consultation et gestion des RDVs (Vérifier/ déplacer/ annuler) — Ajout des proches (les patients peuvent prendre RDV pour un proche, par exemple un enfant ou un parent plus âgé). 	<ul style="list-style-type: none"> — Annulation ou report du RDV — Ajout des proches (5 max) : les patients peuvent prendre RDV pour un proche, par exemple un enfant ou un parent plus âgé
Points forts	<ul style="list-style-type: none"> — Service simple ; — Sécurité et confidentialité ; — Recherche rapide ; — Choix de la date et l'heure qui convient l'utilisateur ; — Blocage automatique de la prise de RDV en ligne pour les patients ayant manqué 3 fois leur RDV ; — Réception des rappels ; — Application responsive (le service est disponible sur ordinateur, tablette ou mobile) 	<ul style="list-style-type: none"> — Service simple et gratuit ; — Recherche rapide ; — Choix de la date et l'heure qui vous convient (en consultant l'agenda du médecin) ; — Réception des rappels ; — Application responsive (le service est disponible sur ordinateur, tablette ou mobile).

<p>Points Faible</p>	<ul style="list-style-type: none"> — Service Doctolib est payant pour les médecins ; — Ne pas pouvoir communiquer avec le médecin via l'application, il y a juste le numéro ; — Si un médecin s'absente pour un imprévu de dernière minute, il se contentait de seulement supprimer les RDV donc les patients seront avertis de l'annulation sans avoir un mot d'excuse ce qui conduit à une perte de confiance. 	<ul style="list-style-type: none"> — La plupart des médecins n'offrent pas la prise de RDV en ligne par exemple, à Bejaïa, on ne trouve que les informations relatives aux médecins (numéro, adresse) mais pas la possibilité de prendre un RDV ; — La géolocalisation n'est pas disponible pour tous les médecins, elle l'est uniquement pour les médecins ayant accepté les prises de RDV en ligne ; — Manque de sécurité, aucune vérification de compte utilisateur par un code d'authentification via numéro ou e-mail lors de l'inscription, d'où risque élevé d'avoir des RDV fictifs (personnes non réelles) ; — Impossible de communiquer avec le médecin via l'application, il y a juste le numéro. Liste non exhaustive des médecins ; — Réception d'un rappel la veille de votre RDV, malgré son annulation ou son report.
----------------------	---	--

TABLE I.1: Présentation des applications Doctolib et Dz-Doc.

I.3 Problématique

De nos jours, le secteur médical Algérien ne dispose malheureusement pas d'une plateforme e-santé efficace concernant la recherche des praticiens et la gestion de prises de rendez-vous et le partage du dossier médical, des consultations entraînant ainsi le non respect et suivi réel des horaires prises par chaque patient. L'impact de cette mauvaise gestion se traduit par le fait que le patient n'a aucune visibilité sur

son heure de consultation, et ainsi ne peut organiser efficacement les autres engagements de sa journée : notion de perte de temps.

En effet, dans la télémedecine, il est possible de prendre rendez-vous pour une consultation aussi bien avec un médecin généraliste qu'avec un spécialiste.

La prise de rendez-vous se fait actuellement soit en appelant soit en se déplaçant vers l'établissement. Les appels téléphoniques ne sont pas toujours concluants parce qu'il n'est pas aisé d'avoir le contact d'un centre de santé ou d'un médecin et les appels ne sont reçus qu'aux horaires d'ouverture du secrétariat de l'établissement de santé .

De plus, il n'est pas rare de constater que le patient ou le médecin ne soit pas en mesure d'honorer un rendez-vous préalablement validé. Dans ces cas de figure, la seconde partie n'est généralement pas tenue informée à temps de l'annulation du rendez-vous.

Il arrive aussi que l'une des deux personnes en particulier le patient, oublie sa date de consultation. Une seconde remarque, toujours dans le domaine sanitaire, est qu'il est difficile pour les agents sanitaires de réunir à temps les dossiers médicaux des patients surtout quand ceux-ci possèdent des dossiers sur plusieurs années.

Faute d'organisation, on note également un nombre important de cas de carnets médicaux et de dossiers médicaux égarés. Ces pertes d'archives entraînent une perte de continuité dans les dossiers médicaux des patients.

De plus, chaque médecin dispose de sa gestion de dossier. Ceci amène un éparpillement des informations des patients qui doivent aussi en cas de besoin se déplacer avec leur dossier.

On demande au patient la date ou la période de sa dernière consultation ; il ne s'en rappelle pas toujours ; il lui est alors créé un nouveau dossier médical. Ceci implique l'impossibilité d'un suivi efficace du patient sur une longue durée.

Dans le cas où il se rappelle vaguement, les infirmiers cherchent dans les archives de dossiers médicaux le dossier médical du patient.

Étant manuel, ce dossier s'il est retrouvé peut être dans un mauvais état, déchiré ou l'écriture effacée lorsque la dernière consultation remonte à longtemps. Le risque de perte des informations relatives aux antécédents du patient est ainsi élevé.

Les médecins suivent toujours les méthodes de rédaction d'ordonnances manuscrite ce qui leurs entraîne une perte de temps par rapport au progrès informatique actuelle.

Nous proposons pour cela une application web destinée aux professionnels de santé et aux patients.

I.4 Objectifs

Ce projet vise l'amélioration des applications e-santé algériennes grâce à une gestion efficace des connaissances en matière de santé . Cela permettra la prise d'un rendez-vous, son report ou son annulation tant par le patient que par le médecin 24h/24 et 7jr/7. Il sera ainsi question de réduire le flux de coups de fil dans les établissements de santé ainsi que le nombre de rendez-vous non honorés ; situation qui pénalise aussi bien le patient que le médecin.

Ce projet se veut également d'aider dans la gestion des dossiers médicaux. Il s'agira de permettre au patient à travers son compte privé de prise de rendez-vous.

Il pourra ainsi accéder à son dossier médical à tout instant et trouver ou localiser un médecin.

Au médecin, sera permis de gérer son agenda, configurer son planning, automatiser l'édition d'ordonnances de retrouver facilement le dossier médical d'un patient ; ce qui lui permet de suivre l'évolution de la santé du patient en ayant accès à ses antécédents et d'y ajouter des consultations

Nous opérons pour le développement d'une health-care plateforme afin d'améliorer la gestion opérationnelle des praticiens de santé et des patients. Notre solution informatique pourra résoudre les problématiques suivantes :

Pour les patients

- Permettre la création d'un compte utilisateur ;
- Permettre la recherche (par nom, spécialité, et ville) et la localisation des médecins ;
- Permettre la prise d'un rendez-vous en-ligne ;
- Permettre l'annulation d'un rendez-vous ;
- Donner au patient la possibilité d'avoir sur lui en tout temps son dossier médical ;
- Consulter sa liste d'ordonnances ;
- Consulter la liste de ses prochains rendez-vous ;
- L'envoi d'un email d'annulation, de confirmation (permettre de signaler au patient et l'informer de l'annulation d'un rendez-vous par un médecin, si ce dernier est indisponible ;
- De modifier ses paramètres de compte.

Pour les praticiens

- Permettre la création de son propre compte ;
- Consulter Agenda calendrier en ligne ;
- L'envoi d'email de confirmation et d'annulation ;
- Consulter les dossiers médicaux de ses malades avec possibilité de partage avec d'autres médecins en cas de besoin en d'autre terme configurer les droits d'accès de ce dossier médical ;
- de Rechercher un patient et ajouter un nouveau patient ;
- de Gérer les rendez-vous (Prendre des Rdvs aux patients) et consulter la liste des prochains RDVs ;
- Automatisation de l'édition d'ordonnance ;
- Configurer son planning (Configuration des plages horaires d'ouvertures, de congé légaux et de ses propres congés).

Pour les Administrateurs

- De Gérer la liste des médecins, des médicaments, des patients, des spécialités.

I.5 Intérêt de la solution par rapport aux existantes

Les solutions existantes au niveau européen offrent des plateformes performantes et sécurisées permettant aux patients de prendre rapidement des rendez-vous en ligne pour une consultation et d'annuler leurs rendez-vous ainsi de rechercher et localiser les médecins de différentes spécialités, la consultation de leurs dossiers médicaux, l'historique de leurs RDVs et ordonnances et aux médecins la facilité de la recherche d'un dossier médical d'un patient, le partager avec d'autres médecins en cas de nécessité ainsi que le remplissage du dossier au cas ou c'est son premier médecin traitant avec les données de la consultation, l'édition automatique d'ordonnances...etc

Mais malheureusement ces solutions ne sont pas conforme au marché algérien et il n'y'a pas une plateforme qui réunit toutes les problématiques cités en haut, pour cela on essayera de s'en inspirer de doctolib pour mettre en place une solution efficace conforme au marché algérien qui va réunir toutes les fonctionnalités (prise de rendez-vous, recherche et localisation, protection et partage des dossiers médicaux ...etc) pour s'assurer du suivi effectif à long terme du patient.

I.6 Conclusion

Dans ce chapitre, nous avons entamé notre travail par la présentation de l'état de l'art ainsi que l'étude des solutions existantes. Ensuite on a soulevé quelques problématiques concernant la gestion du secteur de santé et la prise de rendez vous entre patients et médecins. Enfin nous avons décrit le contenu des objectifs qui devra répondre aux besoins des patients et médecins.

Chapitre II

Méthodologie de conception et spécification des besoins

II.1 Introduction

Ce chapitre est consacré à la présentation de la méthodologie de conception. Nous présenterons tout d'abord les différentes méthodes agiles en se focalisant sur SCRUM la méthode la plus appliquée dans notre cas d'étude ensuite nous introduisons le langage de modélisation UML puis dans la partie qui suit sera consacré à la spécification des besoins de notre système qui met l'accent sur les avantages de la méthode scrum, surtout le plan de la productivité et de l'efficacité.

Commençant par l'interviewer de l'acteur principal pour générer les user-stories ensuite l'élaboration du carnet de produit " Backlog Product " basé sur les user-stories classifiés par priorité.

Au niveau de la troisième étape, le backlog-product validé sera détaillé sous forme de backlog-sprint afin de produire les différents releases. Au sein de chaque release, des mêlées quotidiennes seront planifiées et des révisions des sprints seront réalisées pour faire un suivi des objectifs à réaliser.

II.2 Présentation de la méthode SCRUM

Pour conduire ce projet, nous avons opté pour la Méthode **Scrum** qui est un schéma d'organisation de développement de produits complexes considérée comme une pratique agile; pratique qui implique au maximum le demandeur (client) et permet une grande réactivité à ses demandes.

Scrum est défini comme un cadre de travail suivant un cycle de développement itératif, incrémental qui nous permet une Réalisation d'un ensemble de fonctionnalités par itération (Itération d'une durée fixe (de 2 à 4 semaines) => Sprint) qui intègre la participation du client et adaptatif. S'appuyant sur le découpage d'un projet en boîtes de temps, nommées « sprints » pouvant durer entre quelques heures et un mois (avec une préférence pour deux semaines)[Schwaber and Sutherland].

Chaque sprint commence par une estimation suivie d'une planification opérationnelle et se termine par une démonstration de ce qui a été achevé.

Scrum est une méthode agile qui permet de produire la plus grande valeur métier dans la durée la plus

courte.

Elle se caractérise par [22] :

- La définition des fonctionnalités prioritaires ;
- l'ajout de fonctionnalités en cours de projet (pas pendant un sprint).

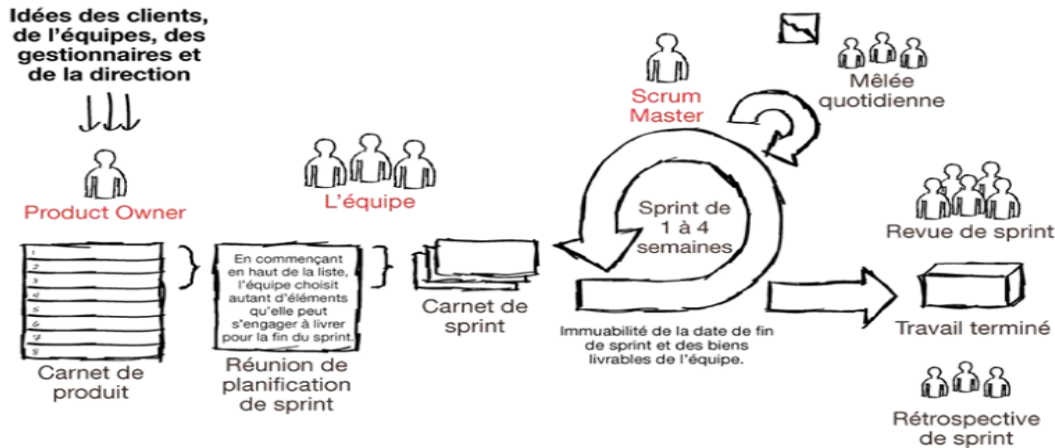


FIGURE II.1 – Les rôles de la méthode SCRUM

- **Le Product Owner** : Il représente les utilisateurs, définit et priorise les demandes produit. Il est intégré à l'équipe et doit savoir être disponible.
- **Le Scrum Master** : Ce n'est pas le chef de projet. Il a un rôle de facilitateur. Sa mission est de tout mettre en œuvre pour que l'équipe travaille dans de bonnes conditions et se concentre sur l'objectif du projet
- **L'équipe SCRUM** : Une équipe regroupant tous les rôles traditionnels : architecte, développeur, testeur, administrateur, etc. Cette équipe développe le produit et se gère en toute autonomie.
- **Coach agile** : Il intervient de manière ponctuelle pour aider à mettre en place les outils Agile dans un domaine méthodologique.

II.3 Langage de modélisation

Pour faciliter notre tâche nous avons fait recours au langage de modélisation unifié UML.

II.3.1 le formalisme UML (Unified Modeling Language)

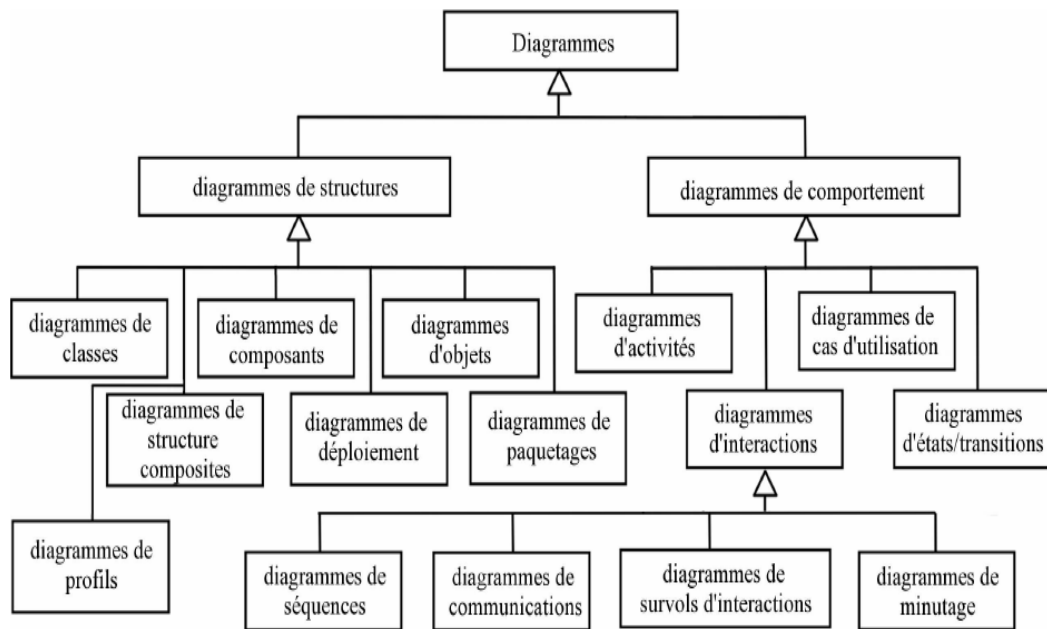
Par rapport à ma cinquantaine de méthode d'analyse et de conception objet qui existaient au début des années 90 [18], seulement trois d'entre elles se sont détachées nettement au bout de quelques années. En effet, la volonté de converger vers une méthode unifiée était déjà bien réelle ; c'est pour cette raison que les méthodes OMT(Object Modeling Technique), BOOCH et OOSE(Object Oriented Software Engineering) se sont démarquées des autres.

UML (Unified Modeling Language) est le résultat de la fusion de trois méthodes d'analyse qui, en quelques années s'est imposé comme standard à utiliser en tant que langage de modélisation.

II.3.2 Différent types de diagrammes UML

Le langage UML s'articule maintenant autour de 14 diagrammes. Chacun d'entre eux est dédié à la représentation d'un système global suivant un point de vue particulier. Par ailleurs le langage UML modélise le système suivant 2 modes de représentation : l'un concerne la structure du système pris au "repos" alors que, l'autre concerne sa dynamique de fonctionnement. Les deux représentations sont nécessaire et complémentaires pour schématiser la façon dont est composé le système et comment ses composants fonctionnent entre eux.

La figure suivante présente les différents types de diagrammes UML.



Les différents types de diagrammes UML

FIGURE II.2 – Type de diagrammes UML version 2.5.1

II.4 Modélisation des besoins

II.4.1 Identification des acteurs

Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel, un autre système) qui interagit directement avec le système étudié.

Dans notre projet on a 6 acteurs qui sont :

- Le Médecin ;
- Le Patient ;
- Le visiteur du site ;

- L'administrateur c'est la personne va gérer les Médecins, Les Patients ainsi que les Médicaments et les spécialités ;
- L'API de Google Maps est un service payant (ou gratuit selon le nombre d'utilisateurs) de cartographie en ligne pour localiser l'adresse des médecins ;
- Le Service Mailtrap est un service de messagerie gmail, permet de transmettre de courts messages textuels de la part des Médecins aux patients pour les informer de la confirmation ou de l'annulation d'un Rendez-vous.

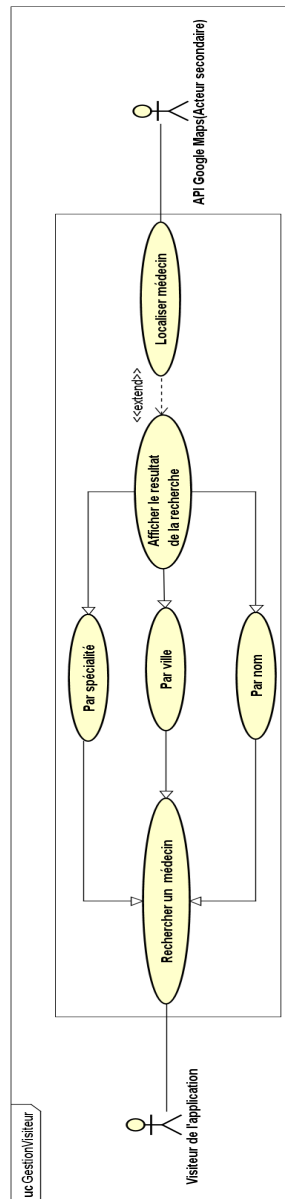


FIGURE II.3 – Diagramme de cas d'utilisation du Visiteur de la plateforme

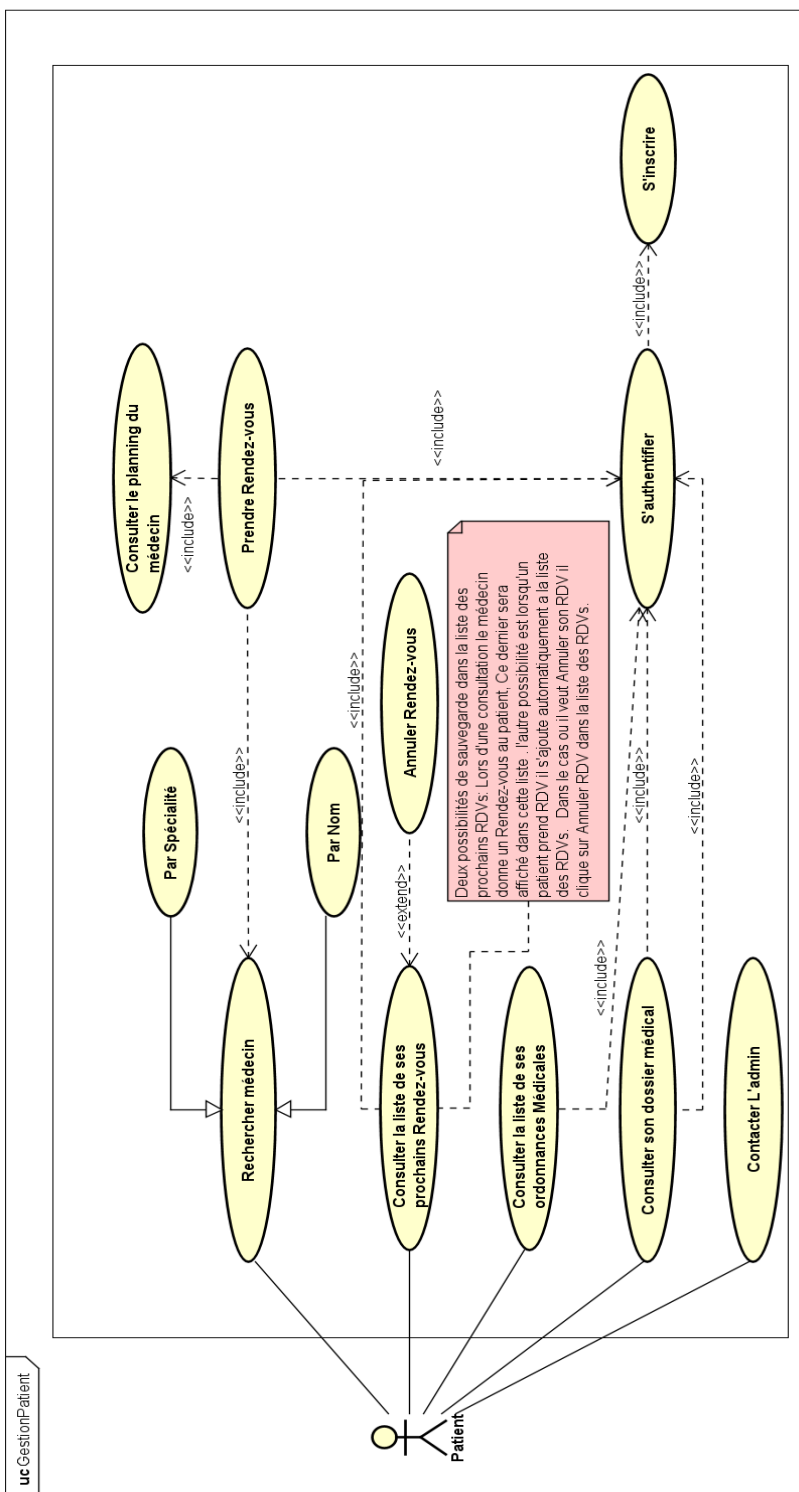


FIGURE II.4 – Diagramme de cas d'utilisation du patient

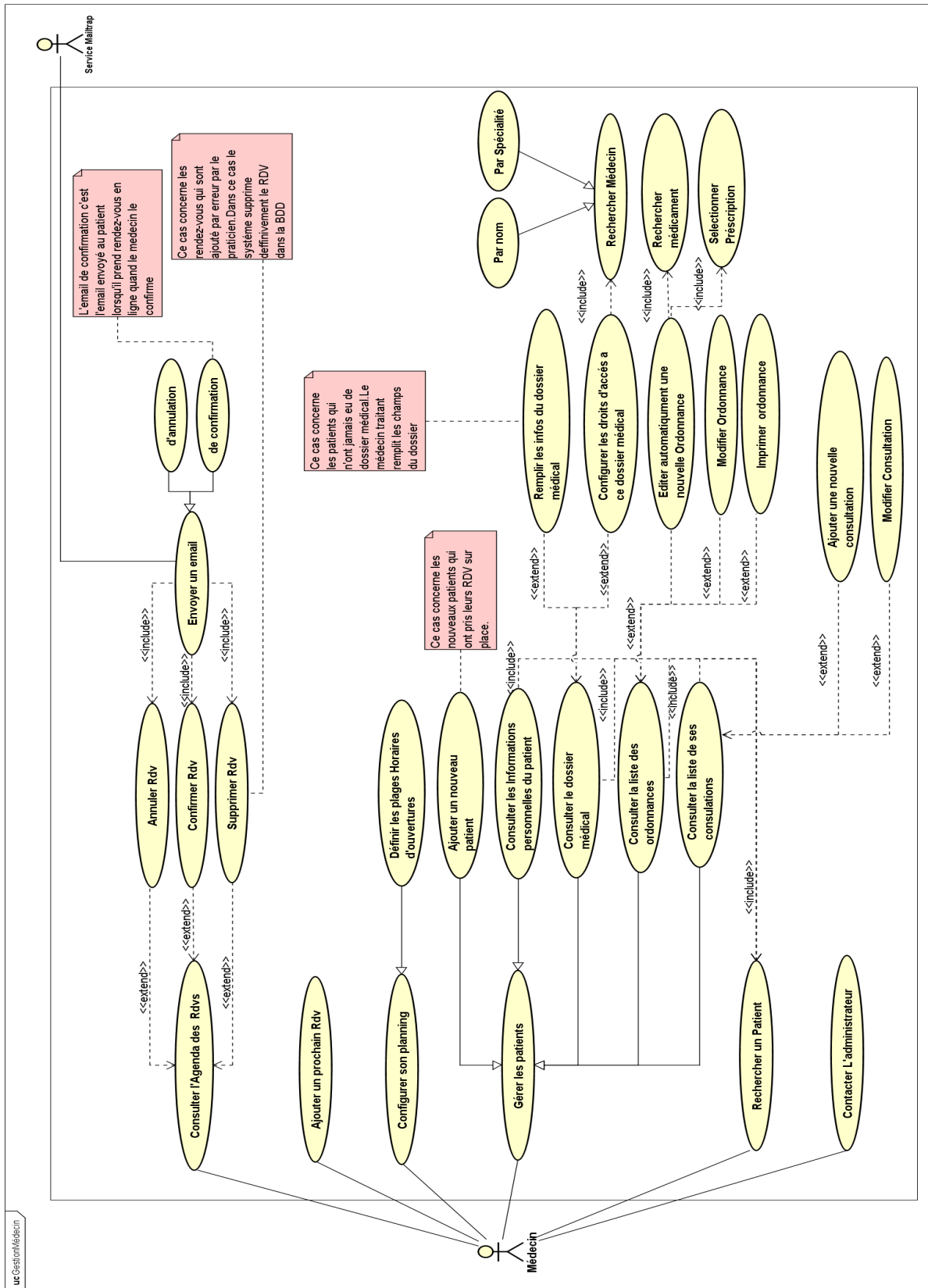


FIGURE II.5 – Diagramme de cas d'utilisation du médecin

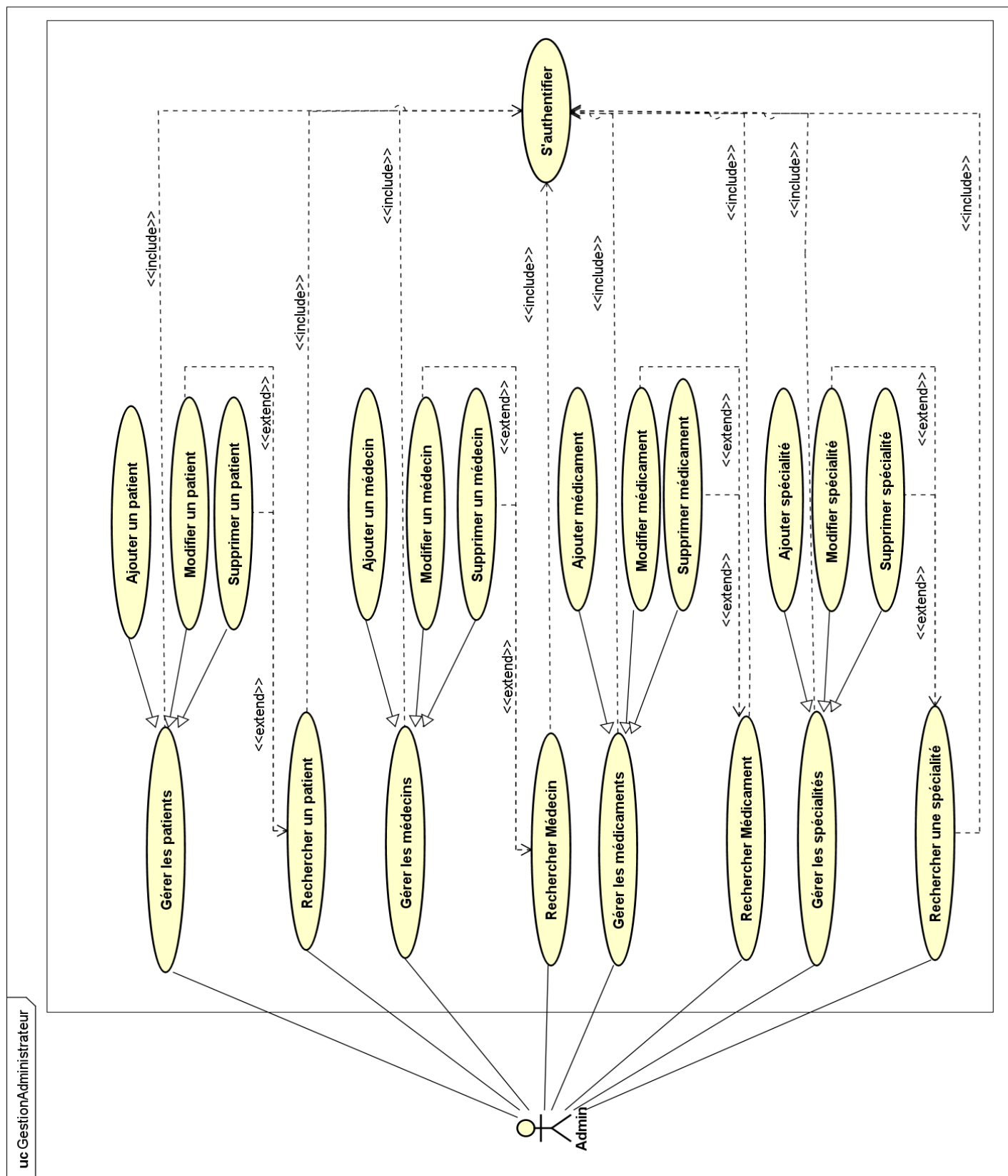


FIGURE II.6 – Diagramme de cas d'utilisation de l'Administrateur

II.4.2 Diagramme de cas d'utilisation global

Ce diagramme de cas d'utilisation globale représente tous les sprints constituant le backlog produit (carnet de produit) qui seront détaillé par la suite.

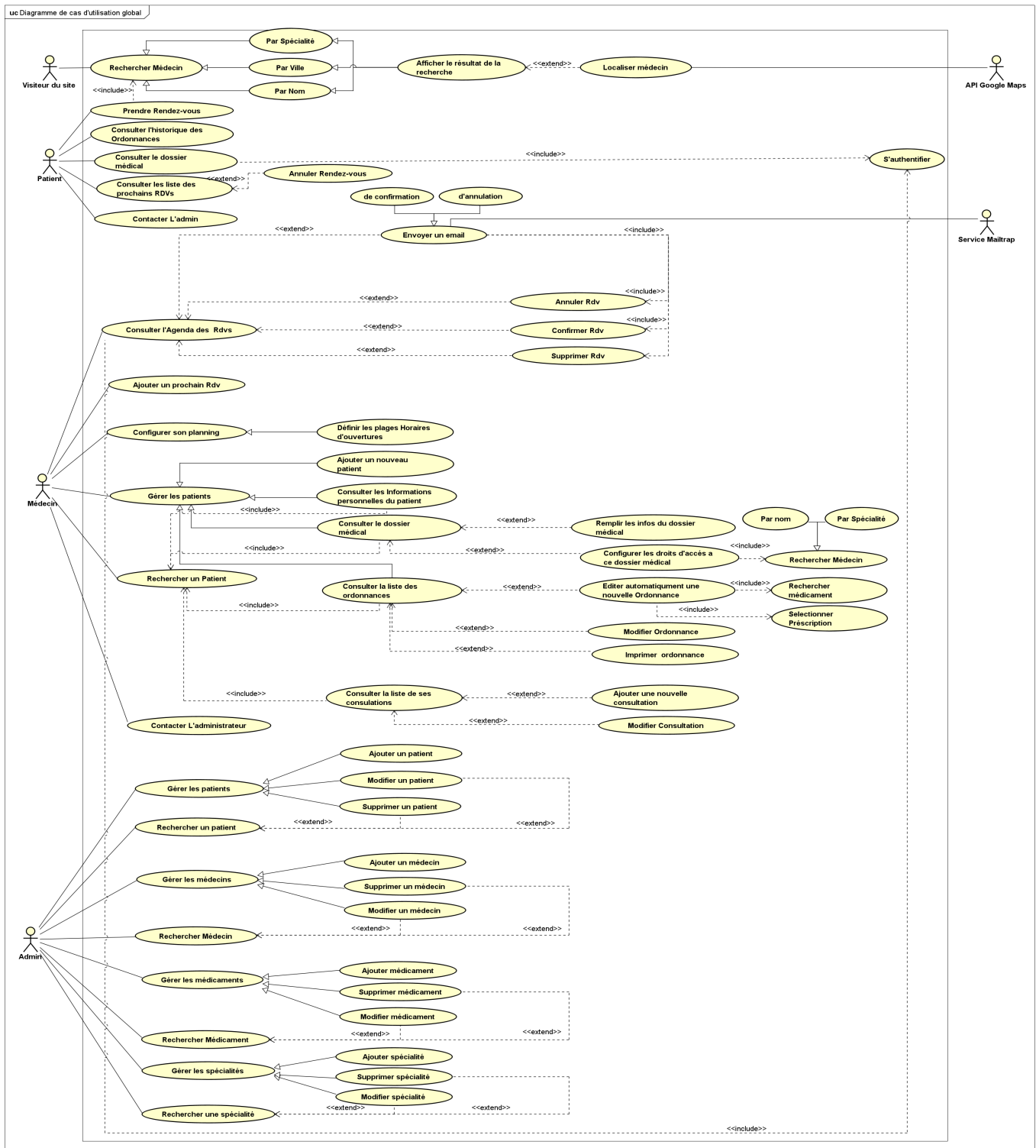


FIGURE II.7 – Diagramme de cas d'utilisation global.

II.5 Pilotage d'un projet avec Scrum

II.5.1 Rôle et user stories

Pour notre projet, les rôles sont répartis comme suit :

Rôles SCRUM	Personnes Affectés
Product Owner	Tech-Instinct
Scrum Master	Mr.KHANOUCHE Mohamed Essaid / Tech-Instinct
Équipe de développement	Mlle BEN BOUDAUD Anissa (Back-End) Mr BENNIA Anis (Front-End)

TABLE II.1: Répartition des Rôles.

On présente sur la figure suivante, les fonctionnalités à développer sous forme de cartes qui définissent des user stories. Ces dernières sont des phases simple dans le langage de tous les jours permettant de décrire avec suffisamment de précision le contenu d'une fonctionnalité à développer.

En tant que médecin , Je veux Consulter mon Agenda en ligne, ainsi que les prochains RDVs, Ajouter un RDV, confirmer un rendez-vous, l'annuler ou le supprimer définitivement et envoyé des emails aux patients (de confirmation, d'annulation	En tant que visiteur de la plate-forme , Je veux trouver et localiser une liste de praticiens par ville et nom ou par spécialité.
En tant que médecin , Je veux Configurer mon planning, définir les plages horaires d'ouvertures et de fermetures	En tant que patient , Je veux prendre un rendez-vous en ligne et consulter mon historique de rendez-vous, et l'annuler en cas d'indisponibilité.
En tant que Médecin , Je veux rechercher un patient, Consulter son dossier médical et configurer les droits d'accès en cas ou je veux le partager avec un autre médecin	En tant que patient , Je veux consulter mon dossier médical.
En tant que Visiteur , Je veux contacter l'administrateur	En tant que Médecin Je veux contacter l'administrateur en cas de besoin.
En tant que Médecin , Je veux Automatiser l'édition de l'ordonnance, Consulter historique des ordonnances, Modifier, Rechercher un médicament et sélectionner prescription	En tant que patient Je veux consulter mon historique d'ordonnances

En tant qu'Administrateur , Je veux Gérer la liste des patients, Rechercher, Ajouter, Modifier, Supprimer	En tant qu'administrateur , Je veux consulter la liste des spécialités, ajouter, rechercher, modifier
En tant qu'Administrateur , Je veux Gérer la liste des médicaments, Rechercher, Ajouter, Modifier, Supprimer	En tant qu'administrateur , Je veux modifier les paramètres de l'application

TABLE II.2: Quelques Exemples de User Stories.

II.5.2 Planification des livrables (Releases)

Un plan de très haut niveau pour plusieurs sprints est créé pendant la planification des releases comme le présente les schémas ci-dessous avec ordre de priorité.

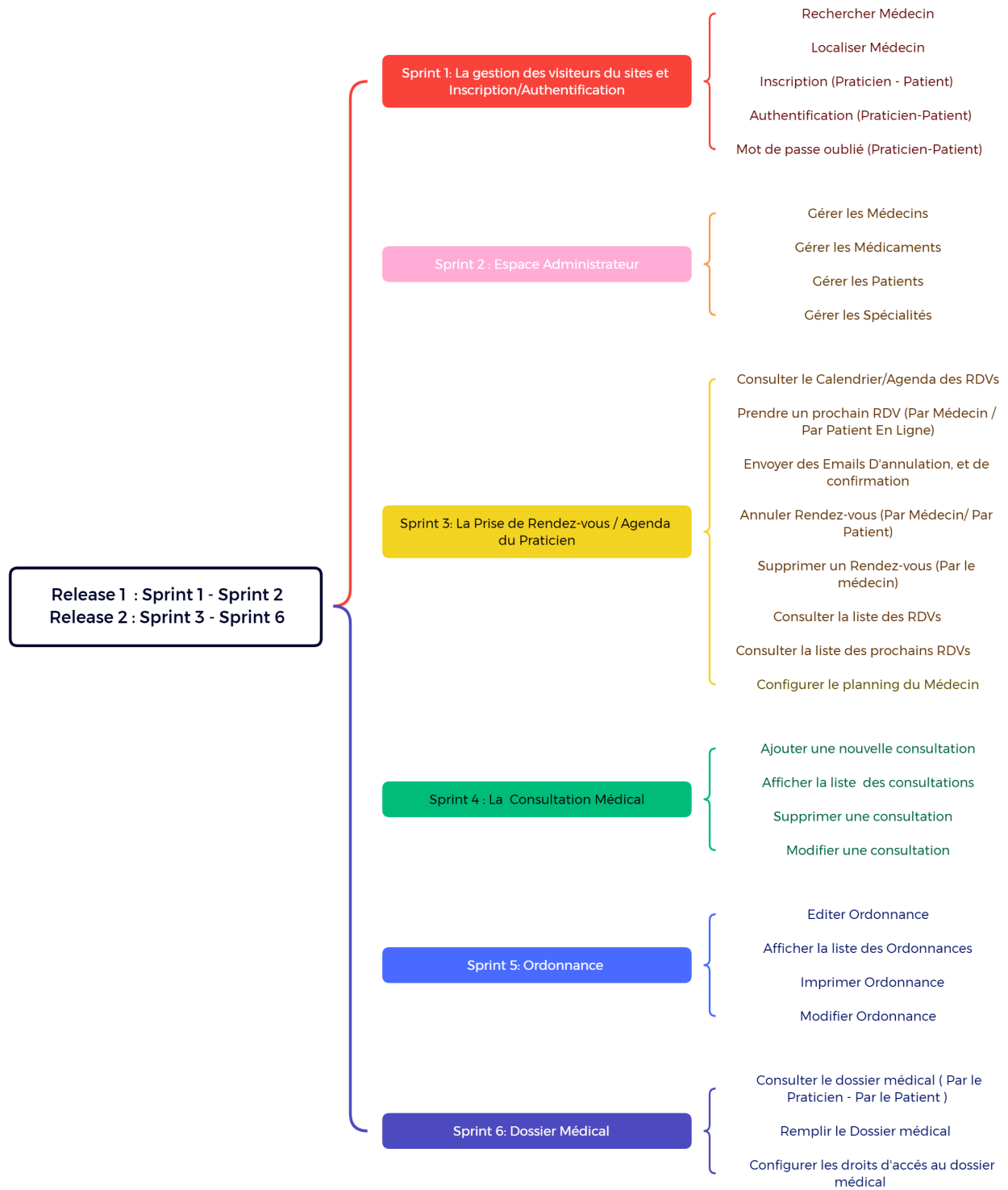


FIGURE II.8 – Schéma des livrables 1 et 2

II.5.3 Backlog-Product (Carnet de produit)

Chaque histoire utilisateur est caractérisée par une priorité dénie par le **Product owner**.

Le backlog-product de notre future système est le suivant :

Sprint	Items	En tant que	Je veux ...	Priorité
Sprint 1	Rechercher médecin	Visiteur	Rechercher un médecin	1
	Localiser médecin	Visiteur	Localiser un médecin	1
	Inscription	Patient	M'inscrire	1
	Inscription	Médecin	M'inscrire	1
	Authentification	Patient	M'authentifier	1
	Authentification	Médecin	M'authentifier	1
	Mot de passe oublié	Médecin	Récupérer mon mot de passe en cas d'oublis	1
	Mot de passe oublié	Patient	Récupérer mon mot de passe en cas d'oublis	1
Sprint 2	Gérer les Médecins	Admin	Ajouter, supprimer, modifier, rechercher un médecin	2
	Gérer les spécialités	Admin	Ajouter, modifier, supprimer, rechercher une spécialité	2
	Gérer les Patients	Admin	Ajouter, modifier, rechercher, supprimer un patient	2
	Gérer les médicaments	Admin	Ajouter, modifier, rechercher, supprimer un médicament	2
Sprint 3	Consulter Agenda/Calendrier du Médecin	Médecin	Consulter Mon Agenda/ Calendrier des Rendez-vous	3
	Prendre un prochain rendez-vous	Patient	Prendre un prochain rendez-vous	3
	Prendre un prochain rendez-vous	Médecin	Prendre un prochain rendez-vous	3
	Envoyer des Émail d'annulation et de confirmation d'un Rendez-vous	Médecin	Envoyer des emails de confirmations aux patients ou d'annulation de rendez-vous	3
	Annuler rendez-vous	Patient	annuler mon rendez-vous	3
	Annuler rendez-vous	Médecin	annuler un rendez-vous en cas d'indisponibilité	3
	Supprimer rendez-vous	Médecin	Supprimer un rendez-vous en cas d'erreur de saisie	3

	Consulter la liste des rendez-vous	Médecin	Consulter la liste de rendez-vous	3
	Consulter la liste des prochains rendez-vous	Médecin	Consulter la liste des prochains rendez-vous	3
	Consulter la liste des patients	Médecin	Ajouter, modifier, supprimer et rechercher	3
	Configurer le Planning du Médecin	Médecin	Ajouter mes jours et heures d'ouvertures.	3
Sprint 4	Ajouter une nouvelle consultation	Médecin	Ajouter une nouvelle consultation pour le patient.	4
	Afficher la liste des consultations	Médecin	Afficher la liste des consultations	4
	Supprimer une consultation	Médecin	Supprimer une consultation	4
	Modifier une consultation	Médecin	Modifier une consultation	4
Sprint 5	Éditer ordonnance	Médecin	Éditer automatiquement l'ordonnance	5
	Afficher la liste des ordonnances	Médecin	Afficher la liste des ordonnances.	5
	Modifier Ordonnance	Médecin	Modifier ordonnance d'un patient en cas d'erreur	5
	Imprimer Ordonnance	Médecin	Imprimer Ordonnance.	5
Sprint 6	Consulter le dossier médical	Médecin	Consulter Le dossier Médical d'un patient	6
	Remplir le dossier médical	Médecin	Remplir les informations du dossier médical d'un patient si c'est premier dossier	6
	Consulter dossier médical	Patient	Consulter mon dossier médical	6
	Configurer les droits d'accès au dossier médical (Le partage)	Médecin	Donner des droits d'accès a un autre médecin de consulter le dossier en cas de besoin	6

TABLE II.3: Le Carnet de Produit.

Signification des couleurs du Logo de la Plate-forme DAWINI

- La Couleur Noir symbole de l'expertise et de la sobriété.
- Le vert évoque la confiance, la sécurité et la proximité. C'est une couleur phare dans le secteur médical.
- Le Blanc c'est une couleur qui sied parfaitement à l'univers médical, surtout dans le monde de médicaments



FIGURE II.9 – Logo de la plate-forme DAWINI

II.6 Conclusion

Dans ce chapitre, nous avons commencé par un aperçu de la méthode SCRUM et ses composants. Ensuite, on a présenté quelques notions de modélisation avec le langage UML. Nous avons spécifié les besoins de notre application via un diagramme de cas d'utilisation globale. Cette phase nous a permis de nous familiariser avec la méthode SCRUM afin de décrire de manière globale et d'exprimer les besoins de l'utilisateur dans un backlog-product pour bien décrire le fonctionnement désiré du système afin d'en faciliter la production.

Dans le chapitre suivant, qui concerne le release 1 (Version de livrable 1), nous entamons une phase très importante dans laquelle on décrira de manière détaillée comment ces besoins seront réalisés dans notre application.

Chapitre III

RELEASE 1 : Étude des sprints du Livrable 1

III.1 Introduction

Dans ce chapitre nous détaillons le travail réalisé pendant les premiers sprint. En effet, chaque release, qui est l'ensemble des sprints, représente une vision distribuée de la période de la production du livrable. Ce premier livrable se compose de deux Sprints tel que :

- **Sprint 1** : La recherche et la localisation des médecins ainsi que l'authentification et l'inscription des patients et médecins ;
- **Sprint 2** : La Réalisation de l'espace de l'administrateur de l'application Web.

Nous présentons par la suite les descriptions textuelles pour chaque cas, ainsi que les diagrammes d'interaction et les diagrammes de classes participantes pour chaque sprint préconisés.

III.2 Étude du premier Sprint

Chaque sprint s'étale sur une semaine et se compose de 8 items :

- Rechercher Médecin
- Localiser Médecin
- Inscription Patient
- Inscription Médecin
- Authentification Patient
- Authentification Médecin
- Mot de passe oublié Patient
- Mot de passe oublié Médecin

Tout au long de cette partie nous traitons les histoires utilisateurs de ce sprint pour produire un incrément potentiellement livrable.

III.2.1 Description textuelle du cas d'utilisation « Rechercher un Médecin »

Sommaire d'identification	
Nom du CU :	Rechercher un médecin.
Acteur :	Visiteur de l'application web.
Type :	Primaire.
Objectif :	Permet aux visiteurs de l'application de rechercher et de localiser les médecins soit par nom, par ville ou par spécialité.
Description des scénarios	
Pré condition :	/
Scénario nominal :	<ol style="list-style-type: none"> 1. Le visiteur de site recherche un médecin (par nom, par spécialité ou par ville). 2. Le système affiche la liste des médecins. 3. Le visiteur consulte les informations du médecin
Scénario Alternatif	6.b. si Aucun médecin ne correspond au nom saisi, le système renvoie un message d'indisponibilité.
Post condition :	Le visiteur consulte les informations et du médecin recherché ou la liste des médecins.

TABLE III.1: Description textuelle du cas d'utilisation « Rechercher Médecin ».

III.2.2 Diagramme d'interaction du cas d'utilisation « Rechercher Médecin »

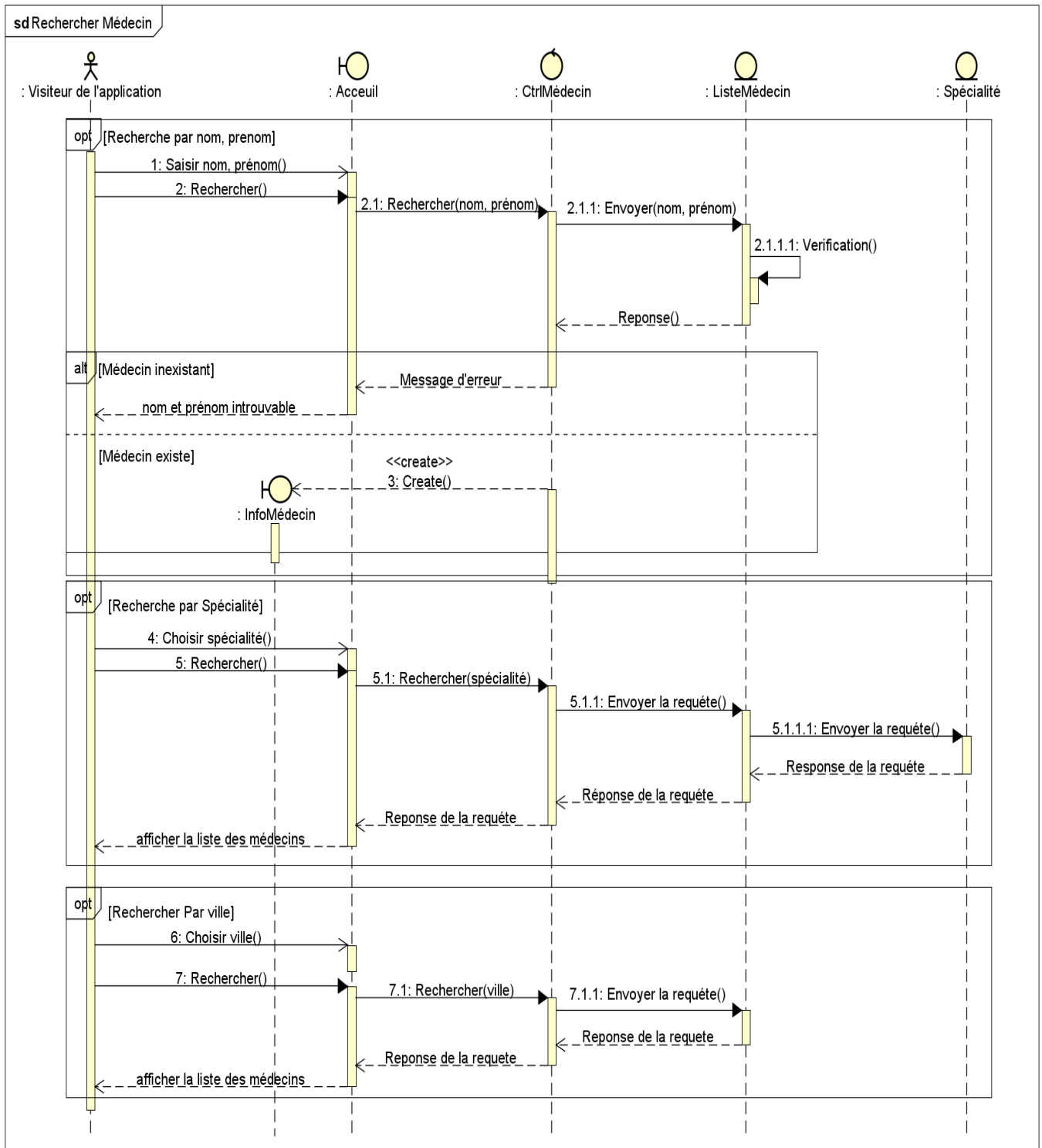


FIGURE III.1 – Diagramme d'interaction de cas d'utilisation « Rechercher Médecin »

III.2.3 Description textuelle du cas d'utilisation « localiser un Médecin »

Sommaire d'identification	
Nom du CU :	localiser un médecin.
Acteur :	Visiteur de l'application, API GOOGLE MAPS.
Type :	Primaire.
Objectif :	Permet aux visiteurs de l'application de localiser l'adresse de l'établissement du médecin sur Google Maps
Description des scénarios	
Pré condition :	/
Scénario nominal :	<ol style="list-style-type: none"> 1. Le visiteur de site recherche un médecin (par nom, par spécialité ou par ville). 2. Le système affiche la liste des médecins. 3. Le visiteur consulte les informations du médecin, il clique sur localisation 4. L'Api google maps lui retourne la localisation du médecin sur la maps.
Post condition :	Le visiteur de l'application localise la l'adresse de l'établissement du médecin recherché.

TABLE III.2: Description textuelle du cas d'utilisation « Localiser Médecin »

III.2.4 Diagramme d'interaction du cas d'utilisation « Localiser Médecin »

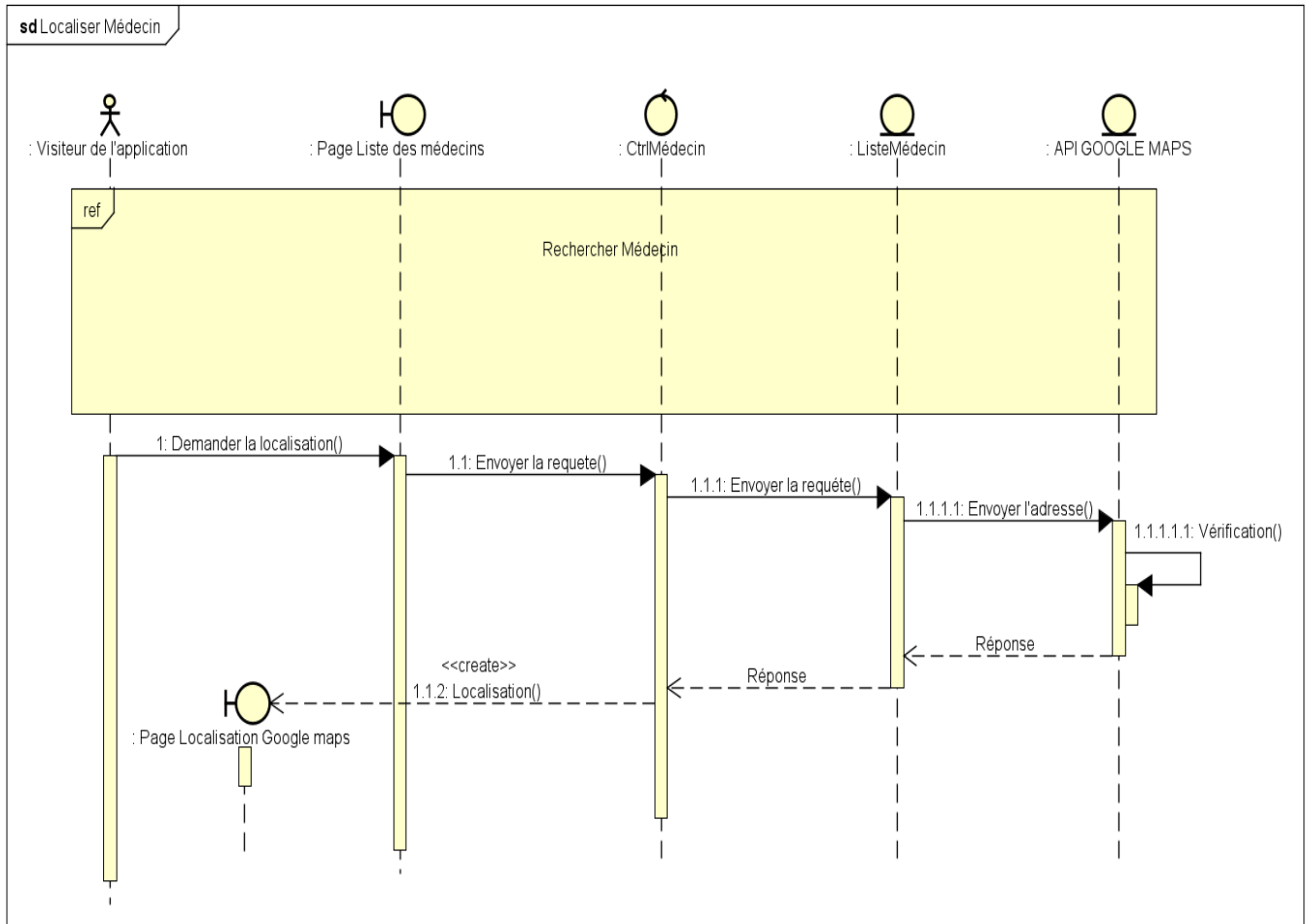


FIGURE III.2 – Diagramme d'interaction de cas d'utilisation « Localiser Médecin »

III.2.5 Diagramme de classes participantes du cas d'utilisation « Localiser Médecin »

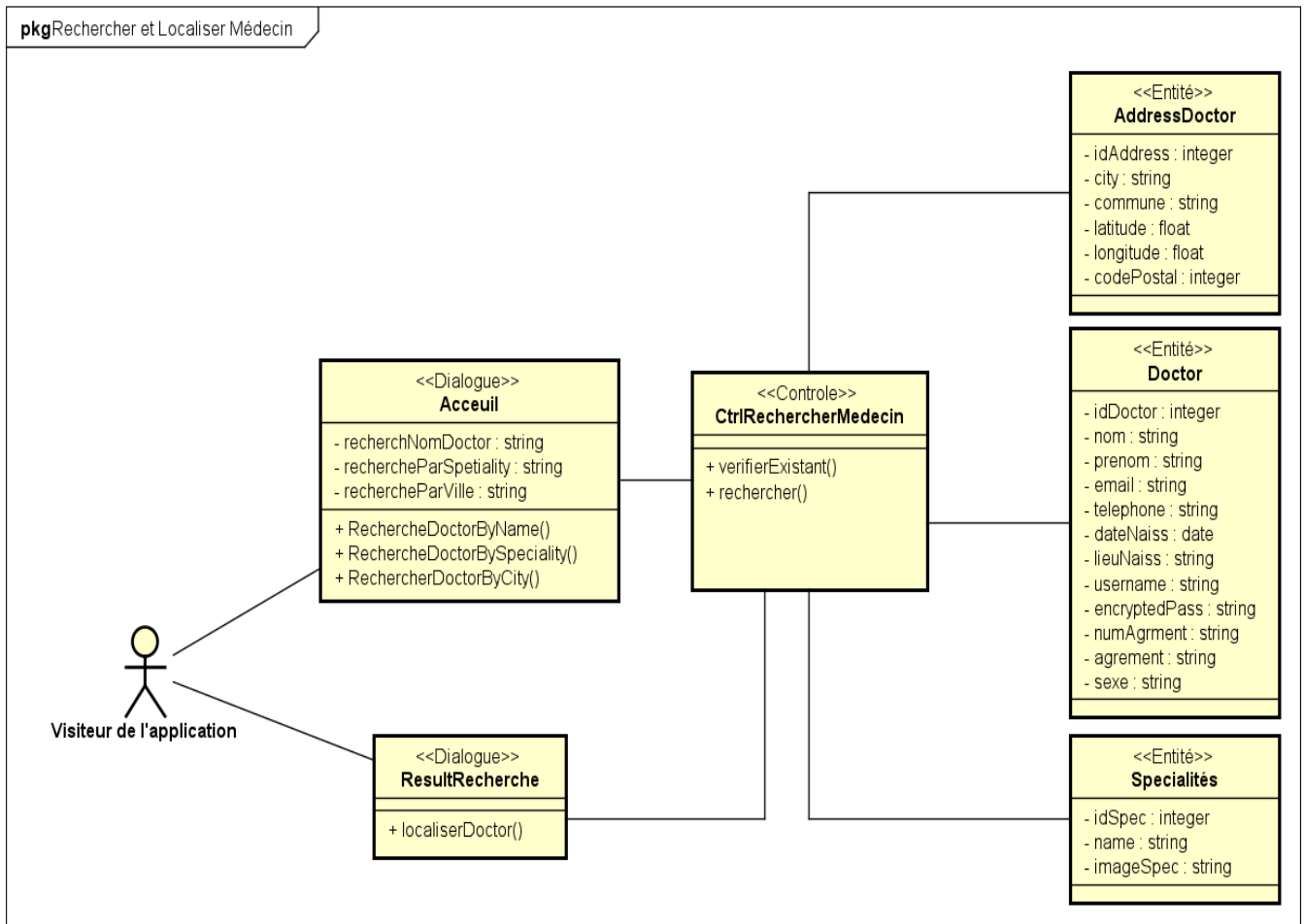


FIGURE III.3 – Diagramme d’interaction de cas d’utilisation « Rechercher et Localiser un médecin »

III.2.6 Fiche Descriptive du cas d'utilisation « Inscription du Patient »

Sommaire d'identification	
Nom du CU :	S'inscrire Patient
Acteur :	Patient.
Type :	Primaire.
Objectif :	Permet au patient de créer un compte au système
Description des scénarios	
Pré condition :	/
Scénario nominal :	<ol style="list-style-type: none"> 1. Le système affiche le formulaire de l'inscription. 2. Le patient remplit le formulaire. 3. Le patient valide le formulaire. 4. Le système vérifie les champs du formulaire. 5. Le système envoie un lien de confirmation à la boîte mail du patient 6. le patient clique sur le lien de confirmation envoyé à sa boîte mail. 8. Le système active le compte.
Scénario Alternatif	<ol style="list-style-type: none"> 2.a. si formulaire mal rempli, afficher des messages d'erreurs. 6.a. si le patient ne clique pas sur le lien de confirmation, son compte reste inactive.
Post condition :	Le patient S'inscrit avec succès.

TABLE III.3: Description textuelle du cas d'utilisation « S'inscrire Patient »

III.2.8 Diagramme de classes participantes du cas d'utilisation Inscription Patient

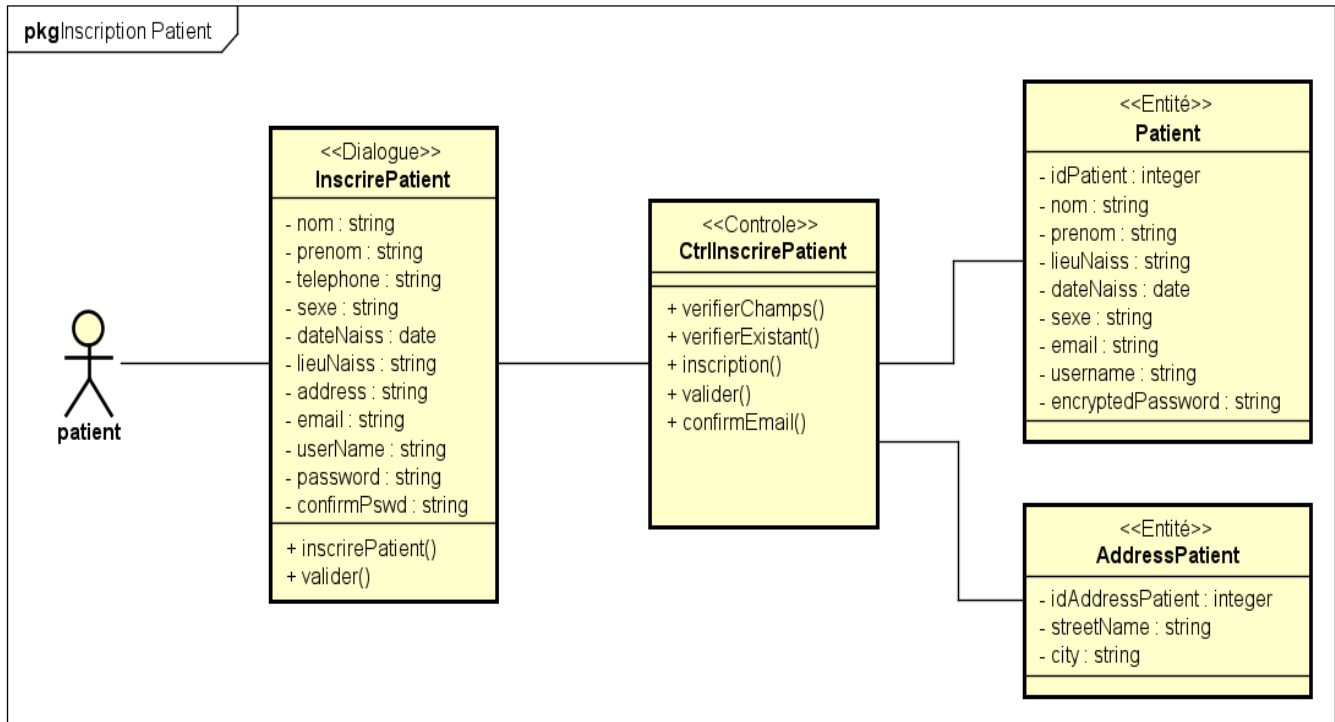


FIGURE III.5 – Diagramme d'interaction de cas d'utilisation « Inscire Patient »

III.2.9 Fiche Descriptive du cas d'utilisation « Authentification Patient »

Sommaire d'identification	
Nom du CU :	S'authentifier patient
Acteur :	Patient.
Type :	Primaire.
Objectif :	Permet au patient d'accéder a son compte.
Description des scénarios	
Pré condition :	/

Scénario nominal :	<ol style="list-style-type: none"> 1. Le système affiche le formulaire de connexion. 2. Le patient saisit son username et mot de passe. 3. Le patient valide la connexion. 4. le système vérifie les champs du formulaire. 5. Le système lui affiche la page d'accueil de son compte.
Scénario Alternatif	<ol style="list-style-type: none"> 2.a. si formulaire mal rempli, afficher des messages d'erreurs (Veuillez vérifier votre username et password S'il vous plaît) et rester dans 1. 2.b. si le patient oublie son mot de passe le patient clique sur mot de passe oublié. <ul style="list-style-type: none"> — le système affiche un formulaire saisir e-mail. — le patient saisit son e-mail et valide. — le système envoie un lien de récupération a sa boite mail et lui affiche sur le formulaire veuillez vérifier votre boite e-mail SVP! — le patient clique sur le lien de récupération — le système affiche un formulaire (nouveau mot de passe, confirmer mot de passe). — le patient remplit le formulaire. — Le système affiche le message Mot de passe renouveler avec succès puis revenir a la page de connexion pour S'authentifier.
Post condition :	Le patient accède a son compte avec succès.

TABLE III.4: Description textuelle du cas d'utilisation
« S'authentifier Patient »

III.2.10 Diagramme de classes participantes du cas d'utilisation « Authentification du Patient »

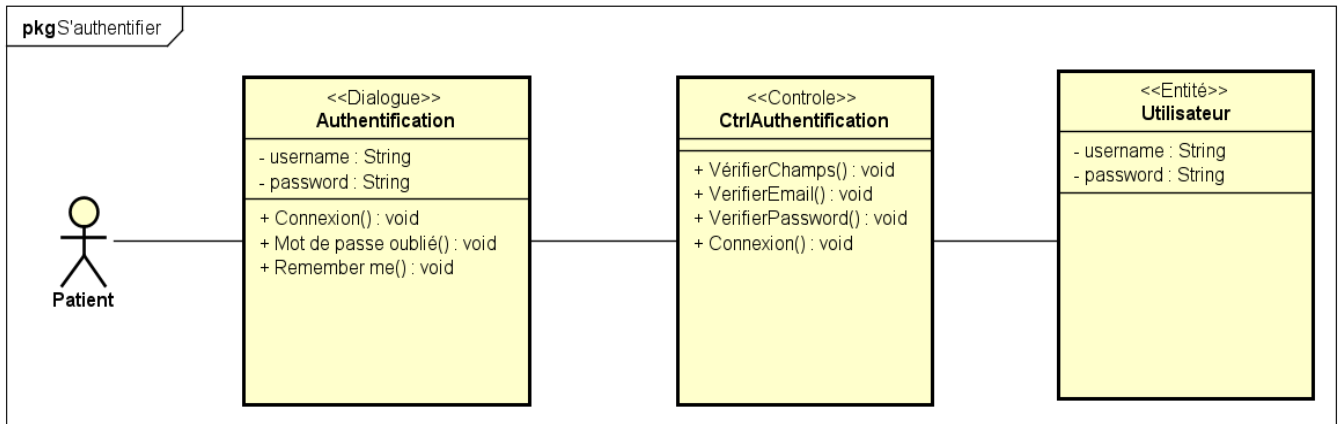


FIGURE III.6 – Diagramme d’interaction de classes participantes « Authentification du Patient »

III.2.11 Diagramme d'interaction du cas d'utilisation « S'authentifier Patient »

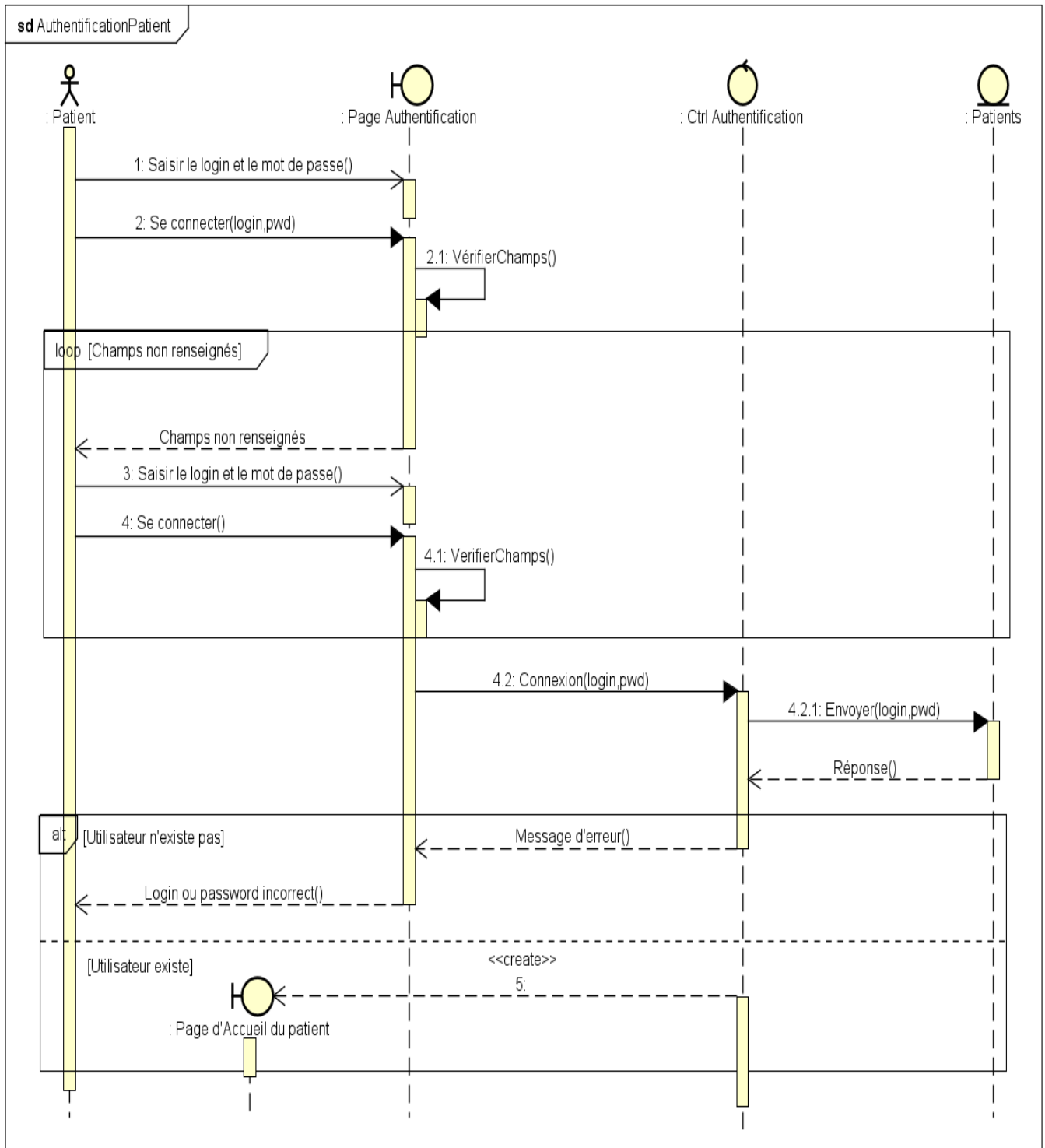


FIGURE III.7 – Diagramme d'interaction de cas d'utilisation « Authentification Patient »

III.3 Étude du deuxième Sprint : Espace Administrateur

Ce sprint s'étale sur deux semaines et se compose de 4 items :

- Gérer les Médecins ;
- Gérer les Patients ;
- Gérer les Spécialités ;
- Gérer les Médicaments.

Tout au long de cette partie nous traitons les histoires utilisateurs de ce sprint pour produire un incrément potentiellement livrable.

III.3.1 Fiche Descriptive du cas d'utilisation « Gérer les Médecins »

Sommaire d'identification	
Nom du CU :	Gérer la Liste des Médecins
Acteur :	Administrateur.
Type :	Primaire.
Objectif :	Permet a l'administrateur de gérer les médecins (d'ajouter des médecins, de Rechercher par nom et spécialité, supprimer, et de modifier ses informations
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. L'administrateur consulte la liste des médecins. 2. Le système lui affiche les informations personnelles. 3. L'administrateur aura la possibilité de Rechercher un médecin, d'ajouter et de modifier ses informations et de le supprimer.
Scénario Alternatif	<ol style="list-style-type: none"> 3.a si le médecin recherché n'existe pas alors le système ne retourne aucun résultat
Post condition :	L'administrateur gère la liste des médecins de la plate-forme avec succès

TABLE III.5: Description textuelle du cas d'utilisation « Gérer les Médecins »

III.3.2 Diagramme de classes participantes du cas d'utilisation « Ajouter un Médecin »

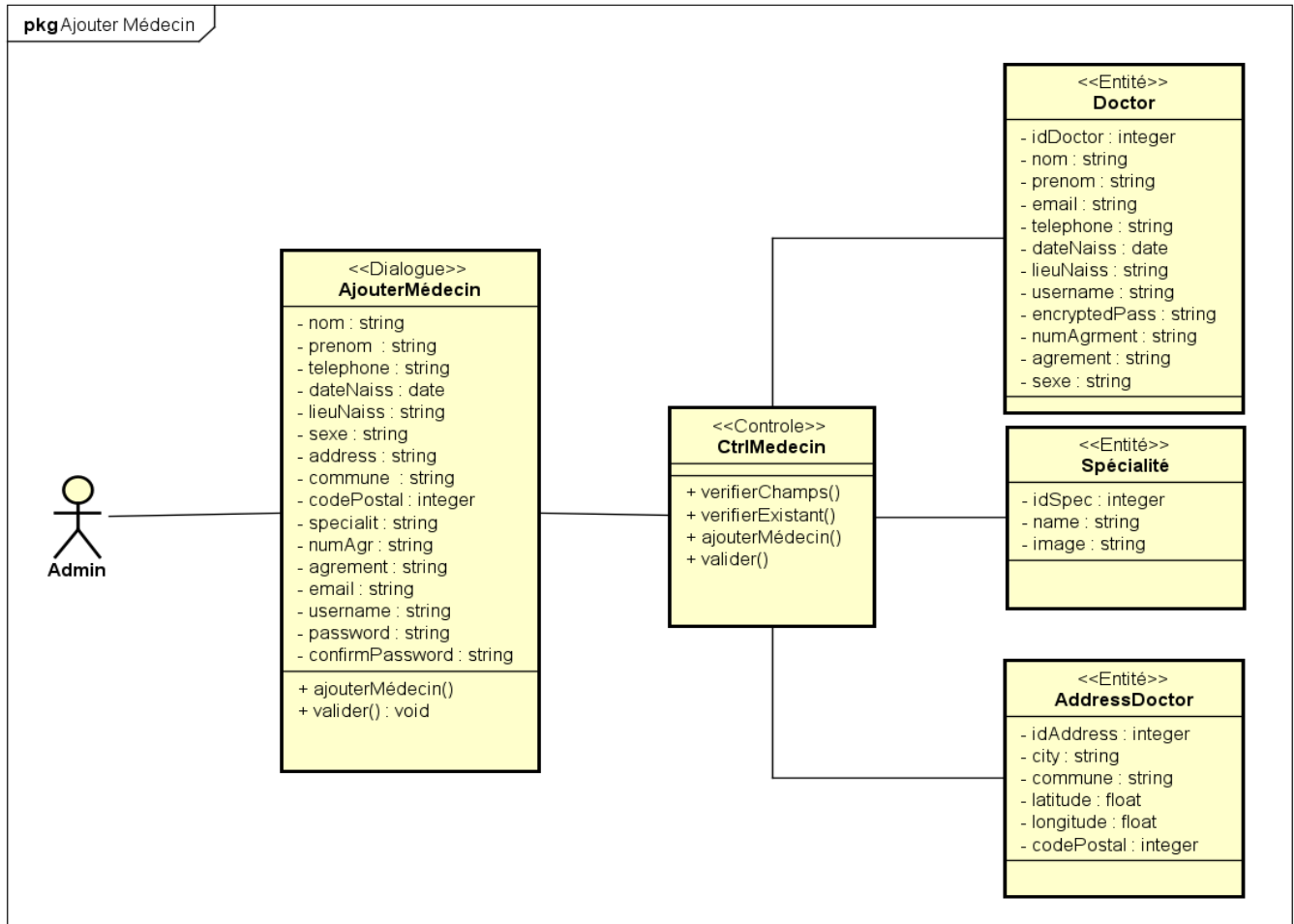


FIGURE III.8 – Diagramme de classes participantes du cas d'utilisation « Ajouter un Médecin »

III.3.3 Diagramme d'interaction du cas d'utilisation « Gérer les Patients »

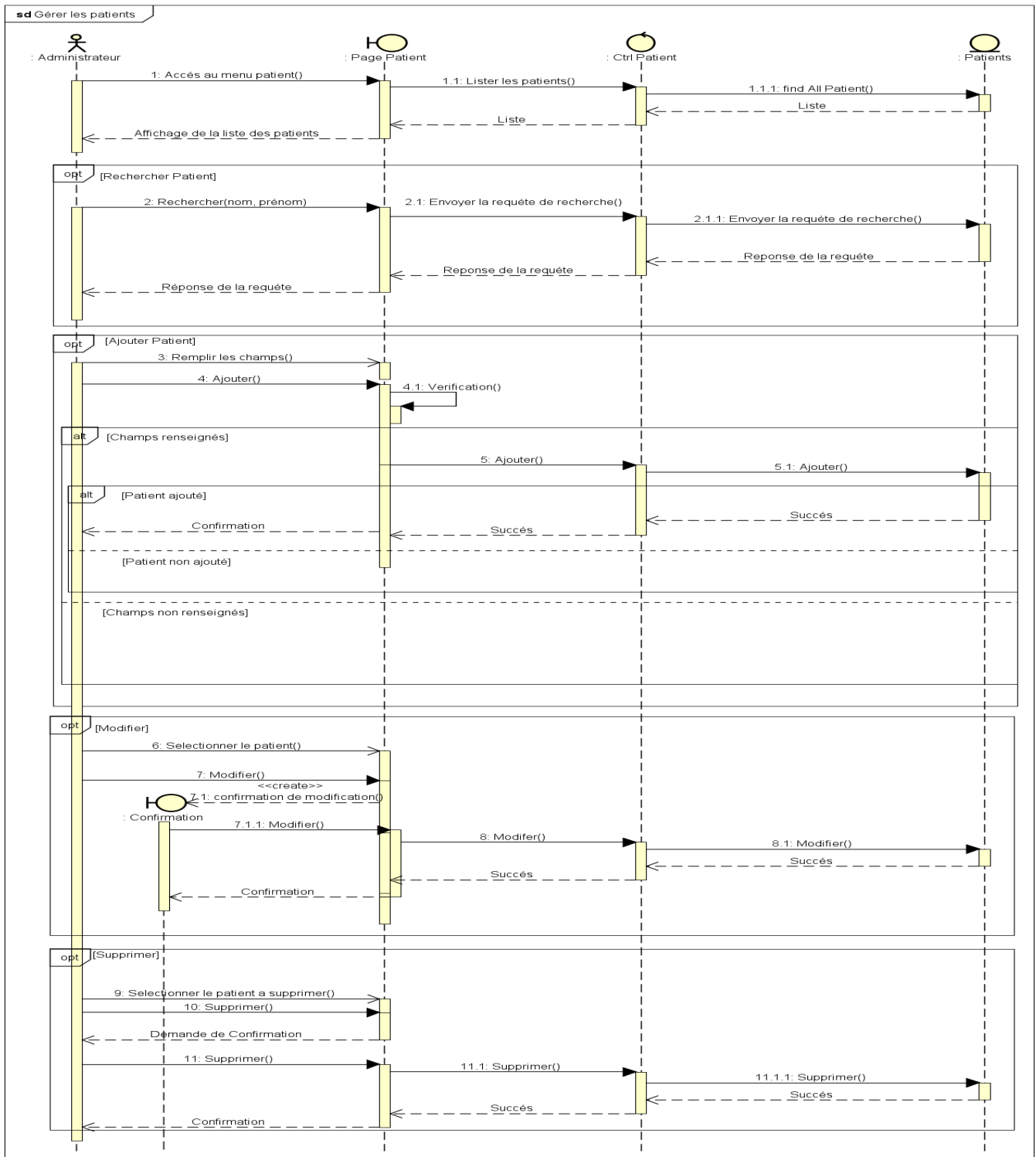


FIGURE III.9 – Diagramme d'interaction « Gérer les Patients »

III.3.4 Diagramme de classes participantes du cas d'utilisation « Ajouter un Patient »

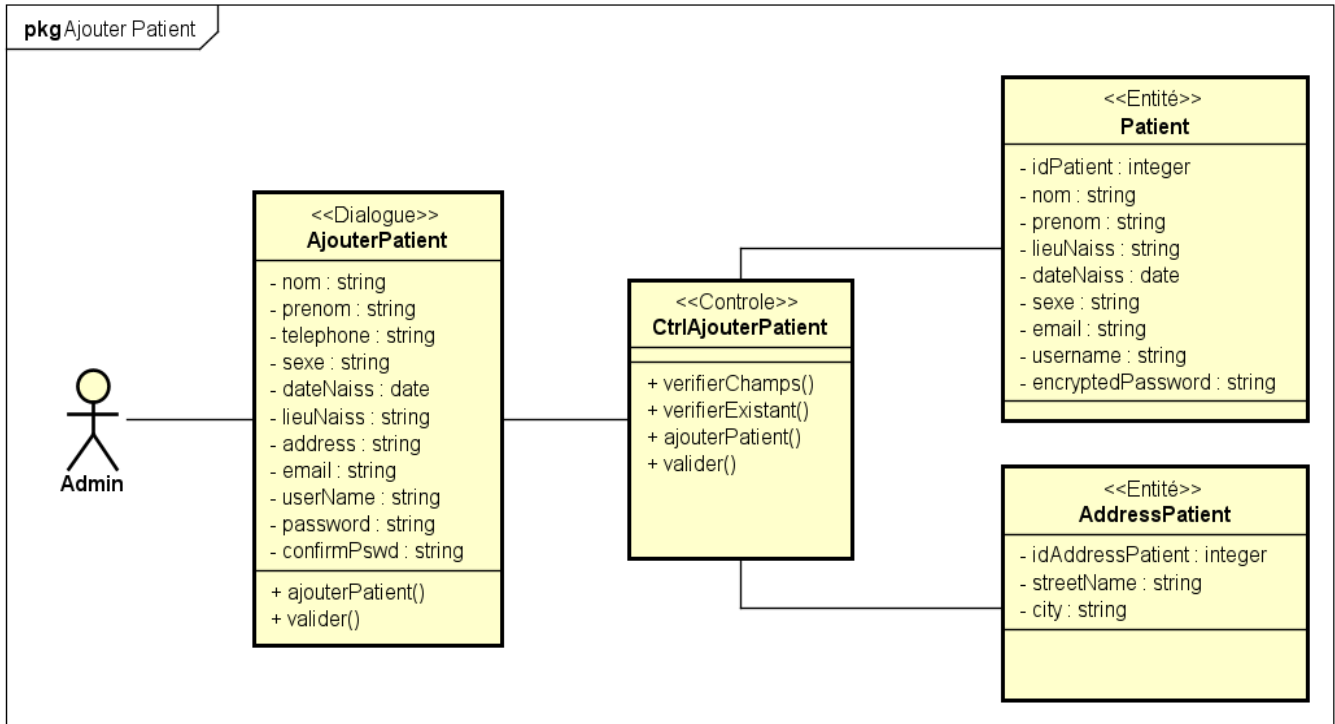


FIGURE III.10 – Diagramme de classes participantes « Ajouter un patient »

III.3.5 Fiche Descriptive du cas d'utilisation « Gérer les Médicaments »

Sommaire d'identification	
Nom du CU :	Gérer les Médicaments
Acteur :	Administrateur.
Type :	Primaire.
Objectif :	Permet a l'administrateur de gérer les Médicaments et de les mettre a jour (Rechercher Médicament par nom, Consulter la liste des Médicaments, Ajouter un Médicament, Modifier un médicament, Supprimer un médicament.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. L'administrateur consulte la liste des médicaments. 2. Le système lui affiche les informations de chaque médicament. 3. L'administrateur aura la possibilité de Rechercher un médicament, d'ajouter et de modifier ses informations ou de le supprimer.
Scénario Alternatif	3.a si le médicament recherché n'existe pas alors le système ne retourne aucun résultat
Post condition :	L'administrateur gère la liste des médicaments de la plate-forme avec succès

TABLE III.6: Description textuelle du cas d'utilisation « Gérer les Médicaments »

III.3.6 Diagramme d'interaction du cas d'utilisation « Gérer les Médicaments »

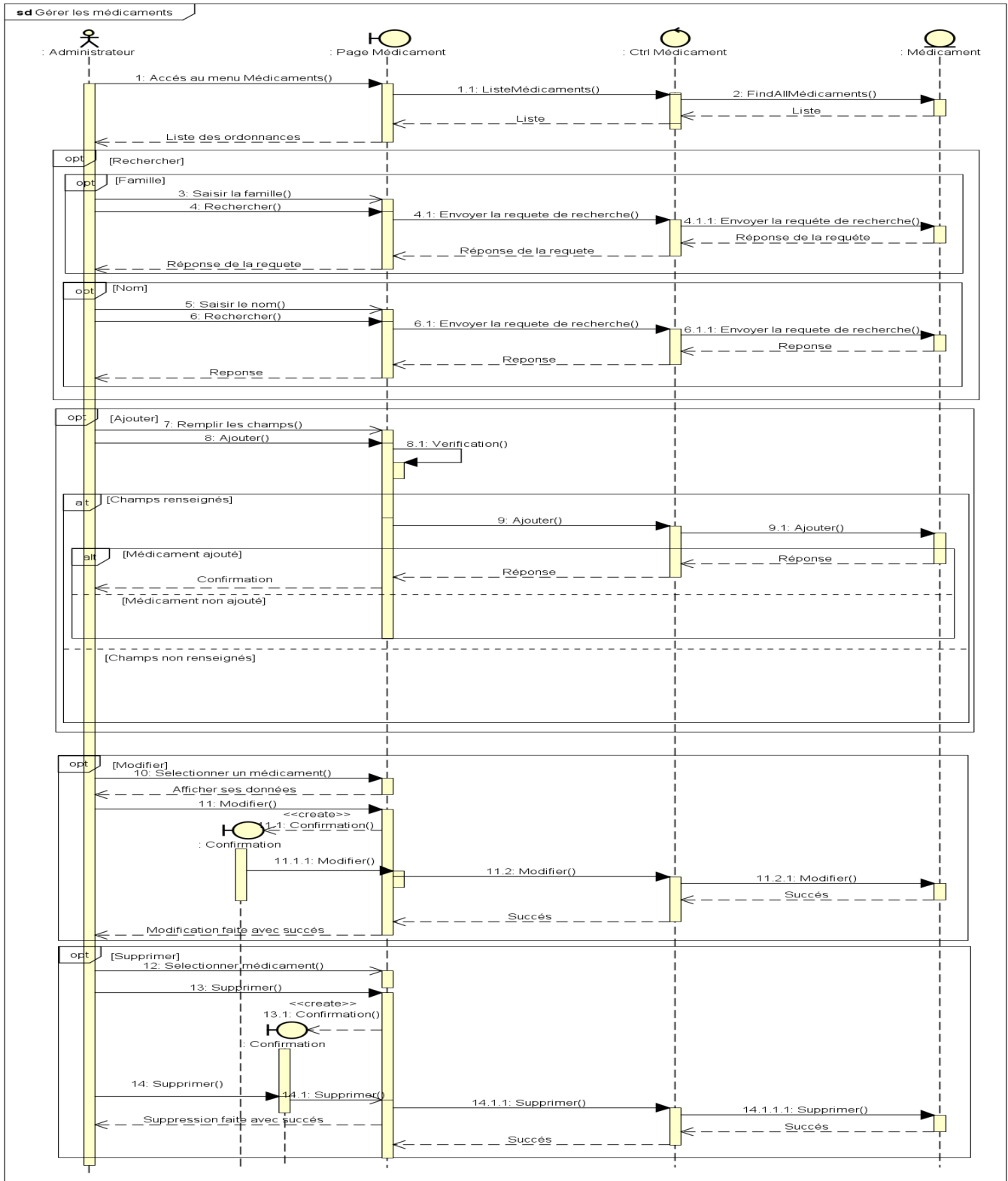


FIGURE III.11 – Diagramme d'interaction « Gérer les Médicaments »

III.3.7 Diagramme de classes participantes du cas d'utilisation « Rechercher et Modifier un Médicament »

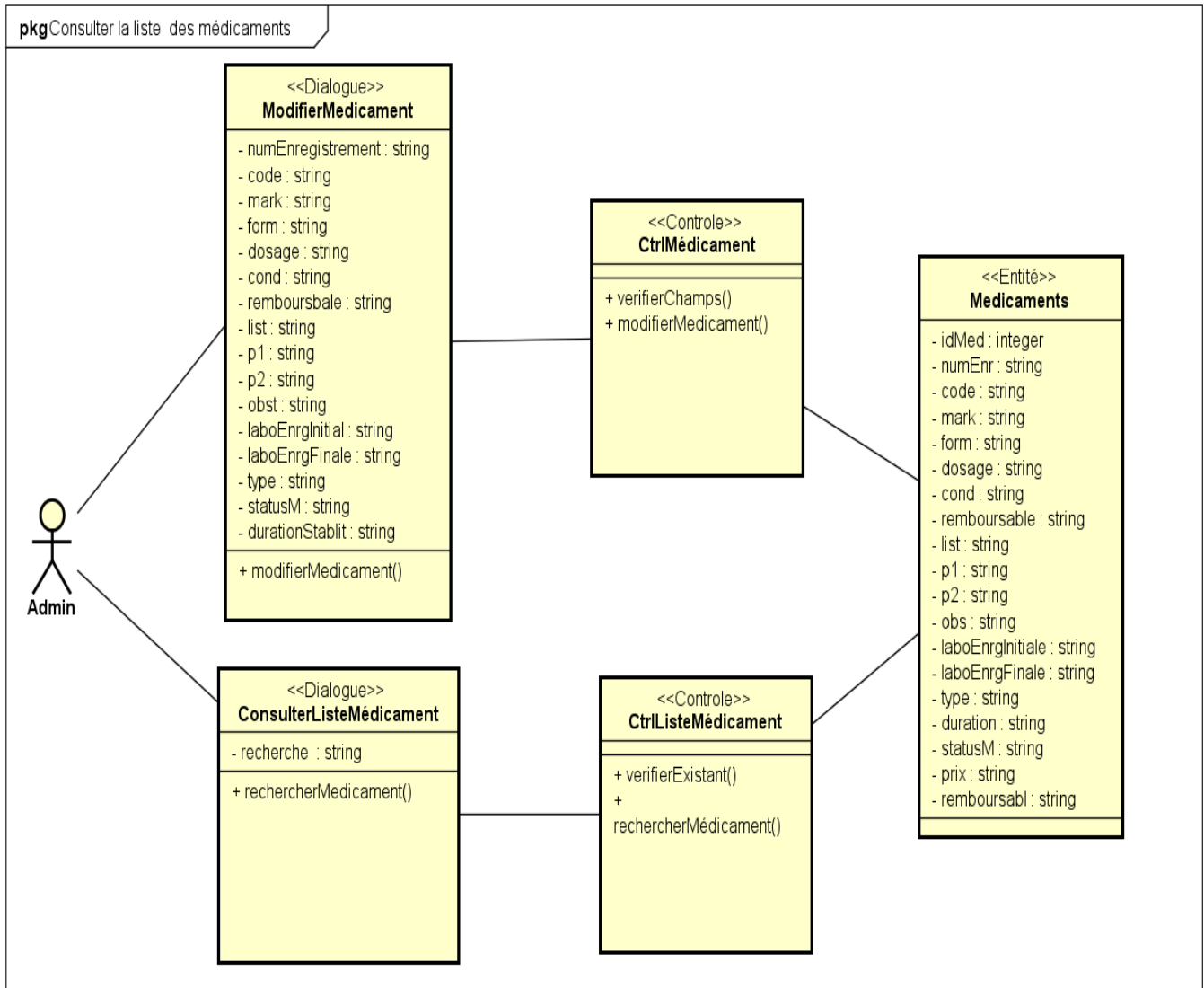


FIGURE III.12 – Diagramme de classes participantes « Rechercher et modifier un médicament »

III.3.8 Fiche Descriptive du cas d'utilisation « Gérer les Spécialités »

Sommaire d'identification	
Nom du CU :	Gérer les Spécialités
Acteur :	Administrateur.
Type :	Primaire.
Objectif :	Permet a l'administrateur de gérer les Spécialités et de les mettre a jour (Rechercher Spécialité par nom, Consulter la liste des Spécialités, Ajouter une Spécialité, Modifier une Spécialité, Supprimer une Spécialité.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. L'administrateur consulte la liste des Spécialités. 2. Le système lui affiche la liste des Spécialités. 3. L'administrateur aura la possibilité de Rechercher une Spécialité, d'ajouter et de modifier ses informations ou de la supprimer.
Scénario Alternatif	<ol style="list-style-type: none"> 3.a si la Spécialité recherché n'existe pas alors le système ne retourne aucun résultat
Post condition :	L'administrateur gère la liste des spécialités de la plate-forme avec succès

TABLE III.7: Description textuelle du cas d'utilisation « Gérer les Spécialités »

III.3.9 Diagramme d'interaction du cas d'utilisation « Gérer les Spécialités »

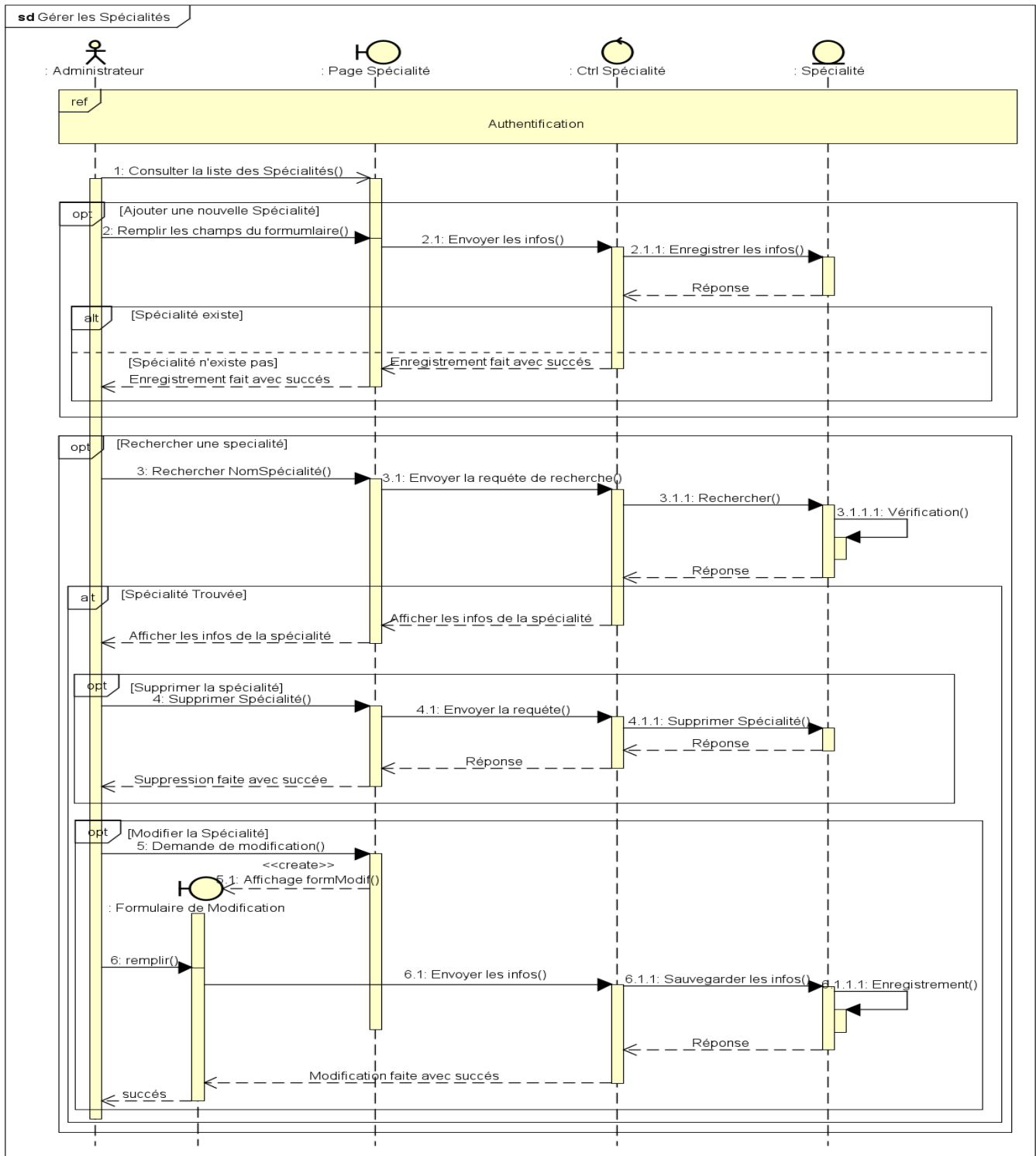


FIGURE III.13 – Diagramme d'interaction « Gérer les Spécialités »

III.3.10 Diagramme de classes participantes du cas d'utilisation « Ajouter une spécialité »

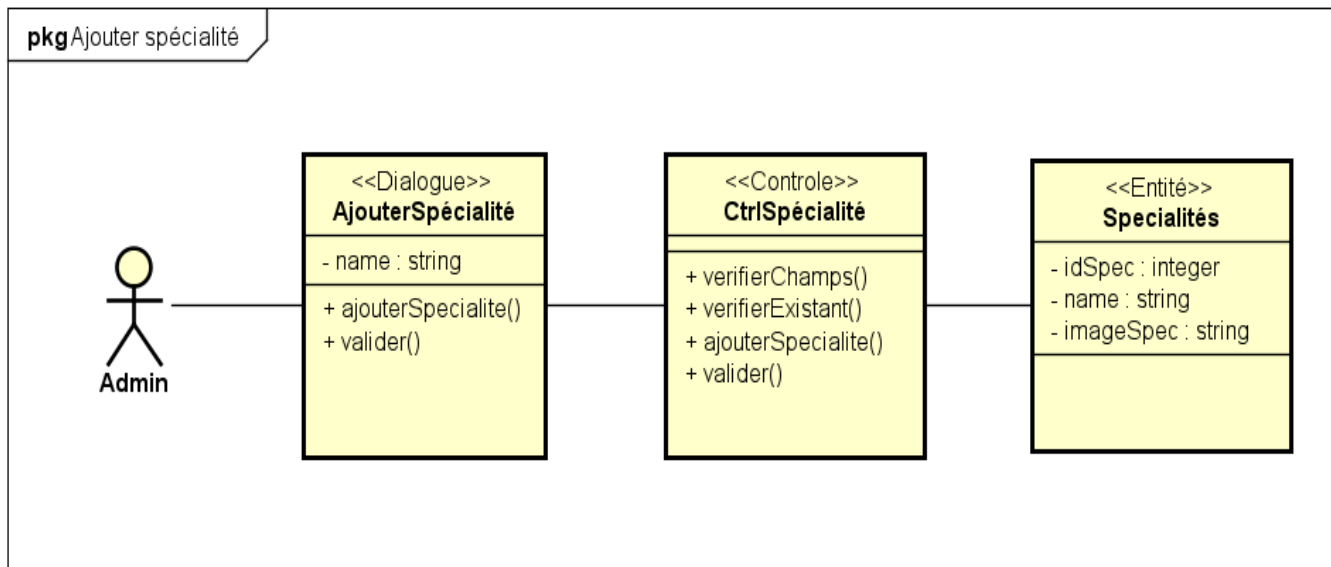


FIGURE III.14 – Diagramme de classes participantes « Ajouter une spécialité »

III.4 Conclusion

Dans ce chapitre, nous avons produit un incrément de valeur pour le client et utile dans un environnement de production et nous avons effectué la conception du livrable 1. Ce dernier a été présenté dans le cadre d'une réunion de fin de livrable et une préparation au prochain.

Dans le chapitre qui suit, nous produisons le second livrable couvrant les fonctionnalités suivantes : la prise de rendez-vous, la consultation médical, l'ordonnance médical et le dossier médical.

Chapitre IV

RELEASE 2 : Étude des sprints du Livrable 2

IV.1 Introduction

Dans ce chapitre nous détaillons le travail réalisé pendant les sprints 3, 4, 5 et 6 .Ce livrable se compose de Sprints tel que :

- **sprint 3** : La Réalisation de l' Agenda du médecin ainsi que la prise de rendez-vous par le patient et par le praticien ;
- **Sprint 4** : Gérer les consultations médicales ;
- **Sprint 5** : Automatisation de l'édition de l'ordonnance ;
- **Sprint 6** : Le Dossier Médical.

Nous présentons par la suite, les description textuelle pour chaque cas d'utilisation et des diagrammes d'interaction ainsi que quelques diagrammes de classes participantes.

IV.2 Étude du troisième Sprint

Ce sprint se compose de 7 Items et s'étale sur 2 semaines qui sont :

- Consulter l'Agenda du Médecin ;
- configurer son planning ;
- Prendre Rendez-vous ;
- Envoyer des email de confirmation ou d'annulation de Rendez-vous ;
- Annuler un Rendez-vous. ;
- Consulter la liste des Rendez-vous.
- Consulter la liste des prochains Rendez-vous.

IV.2.1 Fiche Descriptive du cas d'utilisation « Prendre Rendez-vous Patient »

Sommaire d'identification	
Nom du CU :	Prendre RDV.
Acteur :	Patient.
Type :	Primaire.
Objectif :	Permet au patient de prendre Rendez-vous.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. Le patient sélectionne un médecin après avoir fait sa recherche et sélectionne une date, remplit ses coordonnées. 2. Le système affiche la liste de disponibilité pour le medecin (ses créneaux horaire) pour la date sélectionnée. 3. Le patient sélectionne un créneau horaire et valide. 6. le Système valide le rendez-vous et le rajoute a la liste des prochains Rendez-vous du patient.
Post condition :	Le patient prend son Rendez-vous avec succès.

TABLE IV.1: Description textuelle du cas d'utilisation
« Prendre Rendez-vous Patient »

IV.2.2 Diagramme d'interaction du cas d'utilisation « Prendre Rendez-vous »

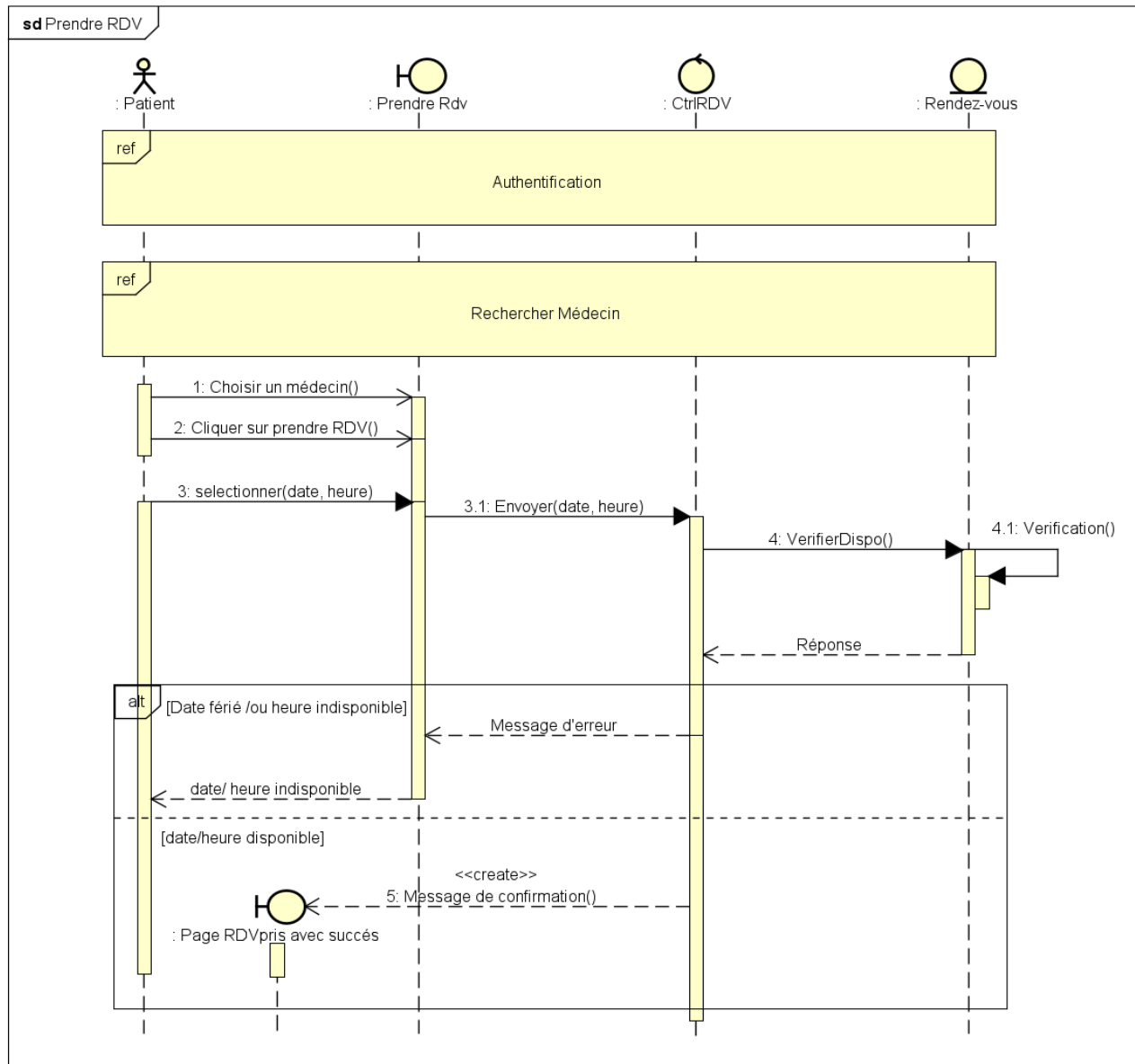


FIGURE IV.1 – Diagramme d'interaction de cas d'utilisation « Prendre Rendez-vous Patient »

IV.2.3 Diagramme de classes participantes du cas d'utilisation « Prendre Rendez-vous par le patient »

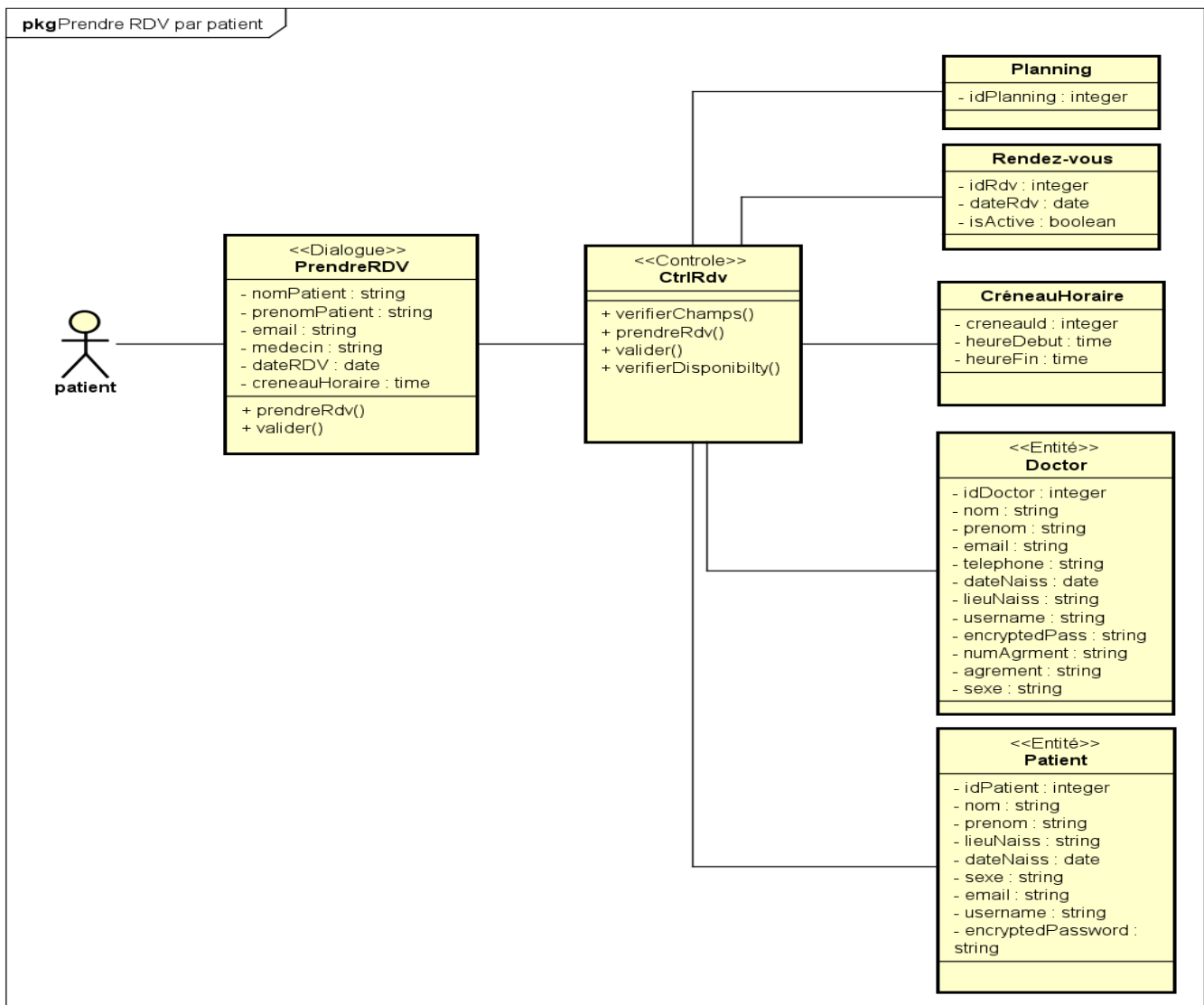


FIGURE IV.2 – Diagramme de classes participantes « Prendre Rendez-vous par le patient »

IV.2.4 Fiche Descriptive du cas d'utilisation « Annuler Rendez-vous du patient »

Sommaire d'identification	
Nom du CU :	Annuler RDV.
Acteur :	Patient.
Type :	Primaire.
Objectif :	Permet au patient d'annuler son Rendez-vous pris par lui même ou par un médecin.
Description des scénarios	
Pré condition :	/
Scénario nominal :	<ol style="list-style-type: none"> 1. Lorsqu'un patient prend un RDV en ligne ou par un médecin le Système envoie un email de confirmation en lui informant qu'il est préférable d'annuler son Rendez-vous (par exemple 48h avant le Rendez-vous). 2. Le patient accède a son compte personnel et consulte la liste des prochains Rendez-vous dans sa page d'accueil. 3. Le Patient Annule le rendez-vous. 4. Le système valide.
Post condition :	Le patient Annule son Rendez-vous avec succès.

TABLE IV.2: Description textuelle du cas d'utilisation « Annuler Rendez-vous Patient »

IV.2.5 Diagramme d'interaction du cas d'utilisation « Annuler Rendez-vous »

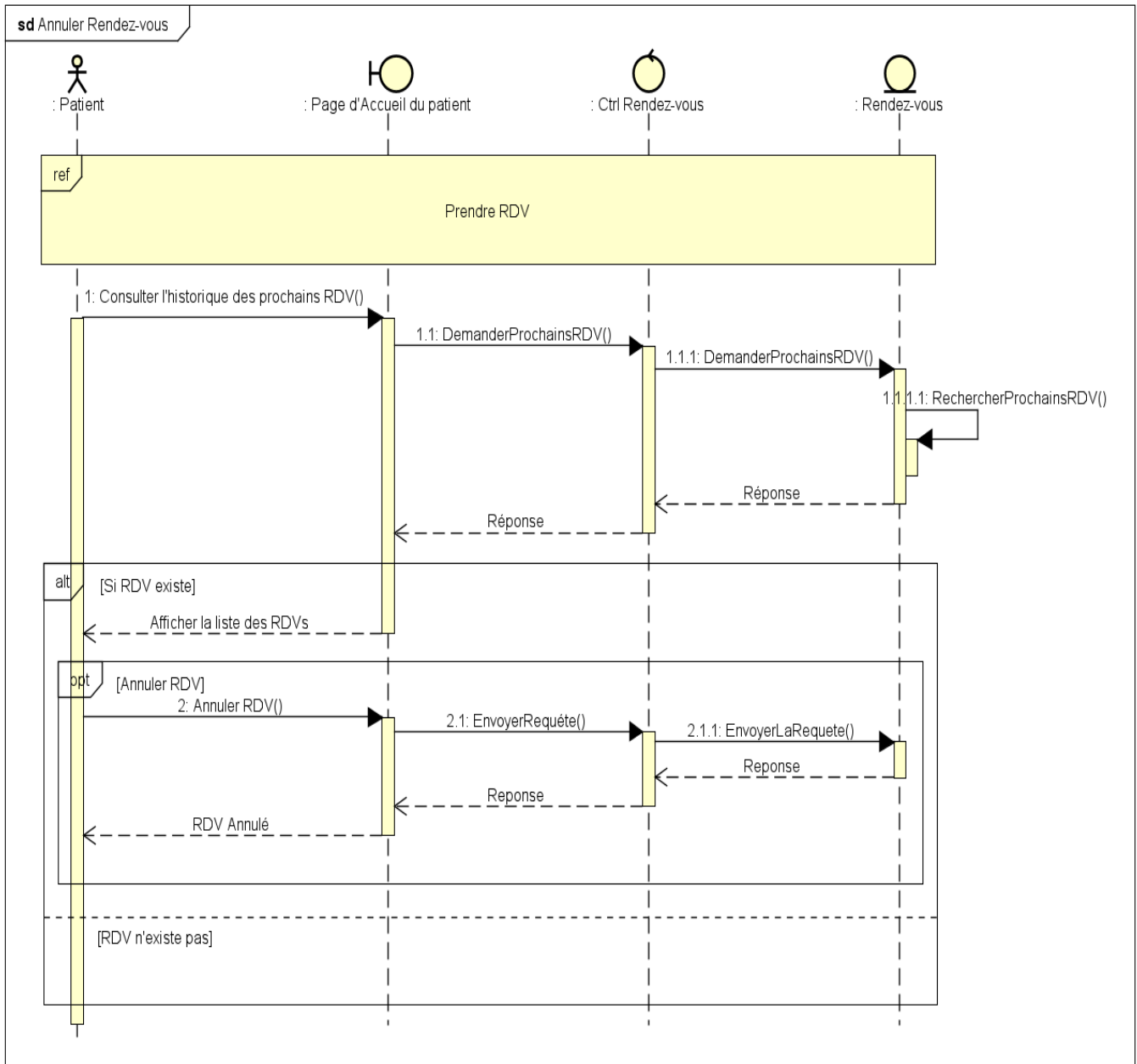


FIGURE IV.3 – Diagramme d'interaction de cas d'utilisation « Annuler Rendez-vous »

IV.2.6 Fiche Descriptive du cas d'utilisation « Consulter son Agenda de Rendez-vous »

Sommaire d'identification	
Nom du CU :	Consulter Son Agenda de RDV
Acteur :	Médecin
Type :	Primaire.
Objectif :	Permet au praticien de visualiser ses rendez-vous sur un mois, une semaine ou un jour. Les rendez-vous peuvent avoir été créés par le praticien ou par le patient depuis la prise de rendez-vous en ligne.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	1. Le système affiche le l'agenda avec qui contient Les Rendez-vous en attente du jour.
Post condition :	Le Médecin consulte son Agenda.

TABLE IV.3: Description textuelle du cas d'utilisation
« Consulter son Agenda de Rendez-vous »

IV.2.7 Diagramme d'interaction du cas d'utilisation « Consulter Mon Agenda de Rendez-vous »

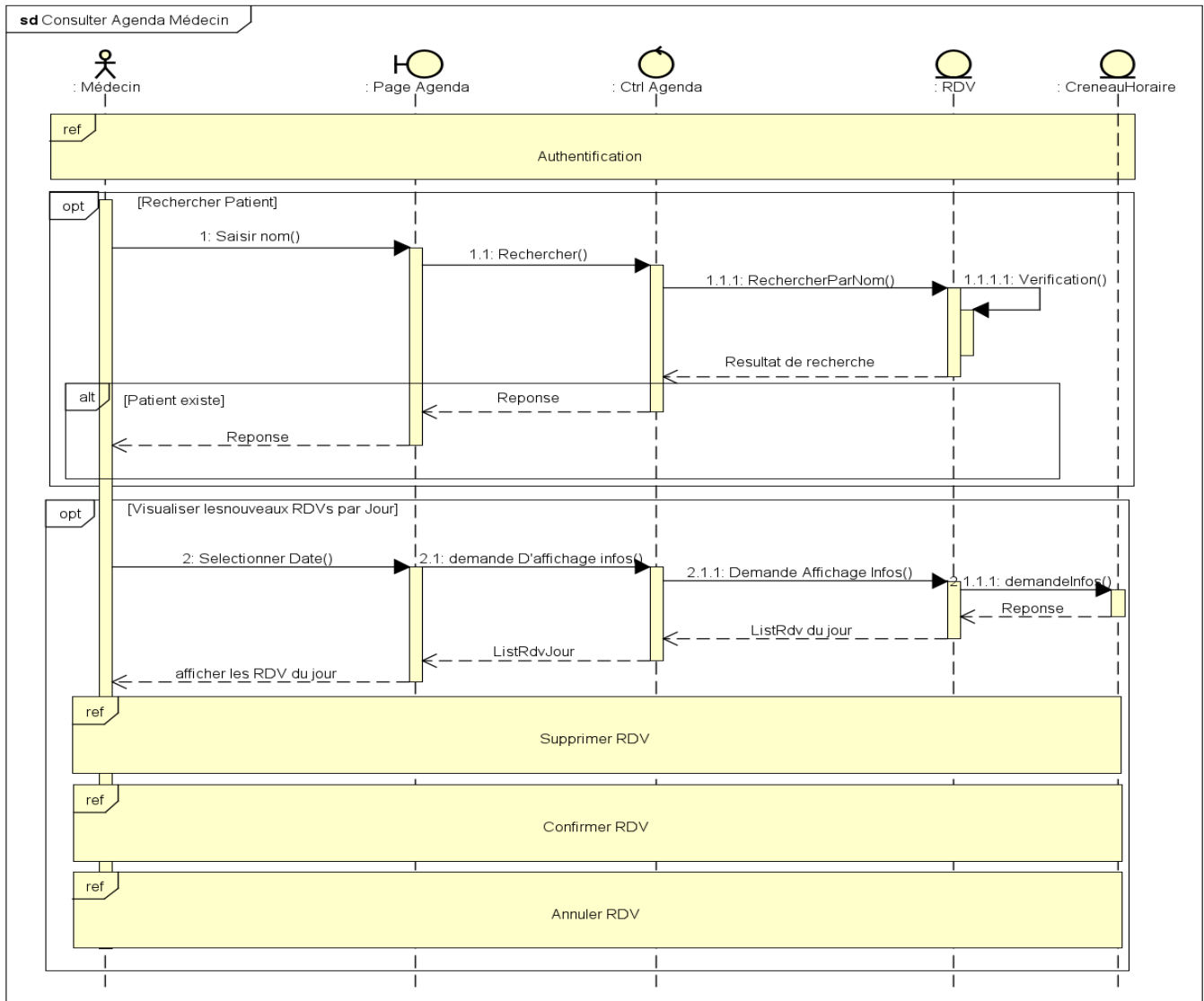


FIGURE IV.4 – Diagramme d'interaction du cas d'utilisation « Consulter mon agenda de Rendez-vous »

IV.2.8 Fiche Descriptive du cas d'utilisation « Ajouter prochain Rendez-vous »

Sommaire d'identification	
Nom du CU :	Ajouter un prochain Rendez-vous
Acteur :	Médecin
Type :	Primaire.
Objectif :	Les prochains Rendez-vous peuvent être pris par le praticien ou par le patient depuis la prise de rendez-vous en ligne.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. Le médecin sélectionne la date et l'heure du rendez-vous ainsi que les coordonnées du patient email, nom et prénom. 6. A la fin de la création d'un rendez-vous, le praticien clique sur le bouton valider pour l'enregistrement du rendez-vous. 7. ce rendez-vous créé sera visible sur l'agenda au bon créneau horaire. 8. Le Praticien ensuite peut envoyer un email de confirmation au patient ou bien d'annulation en cas d'indisponibilité.
Scénario Alternatif	2.a. si aucun créneau est disponible, impossible de prendre rdv pour la date souhaitée.
Post condition :	Le médecin crée un nouveau rendez-vous avec succès.

TABLE IV.4: Description textuelle du cas d'utilisation
« Ajouter un prochain Rendez-vous »

IV.2.9 Fiche Descriptive des cas d'utilisations « Annuler Rendez-vous et supprimer un rendez-vous »

Sommaire d'identification	
Nom du CU :	Annuler Rendez-vous ou le supprimer
Acteur :	Médecin, Système
Type :	Primaire.
Objectif :	Permet au médecin d'annuler un rendez-vous ou le supprimer définitivement.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. Le Médecin clique sur un rendez-vous dans l'agenda. 3. le praticien clique sur supprimer.le système Supprime définitivement le rendez-vous de la base de données.

TABLE IV.5: Description textuelle des cas d'utilisation
« supprimer un Rendez-vous de la part du praticien »

IV.2.10 Diagramme de classes participantes du cas d'utilisation « Annuler Rendez-vous du Patient par le praticien »

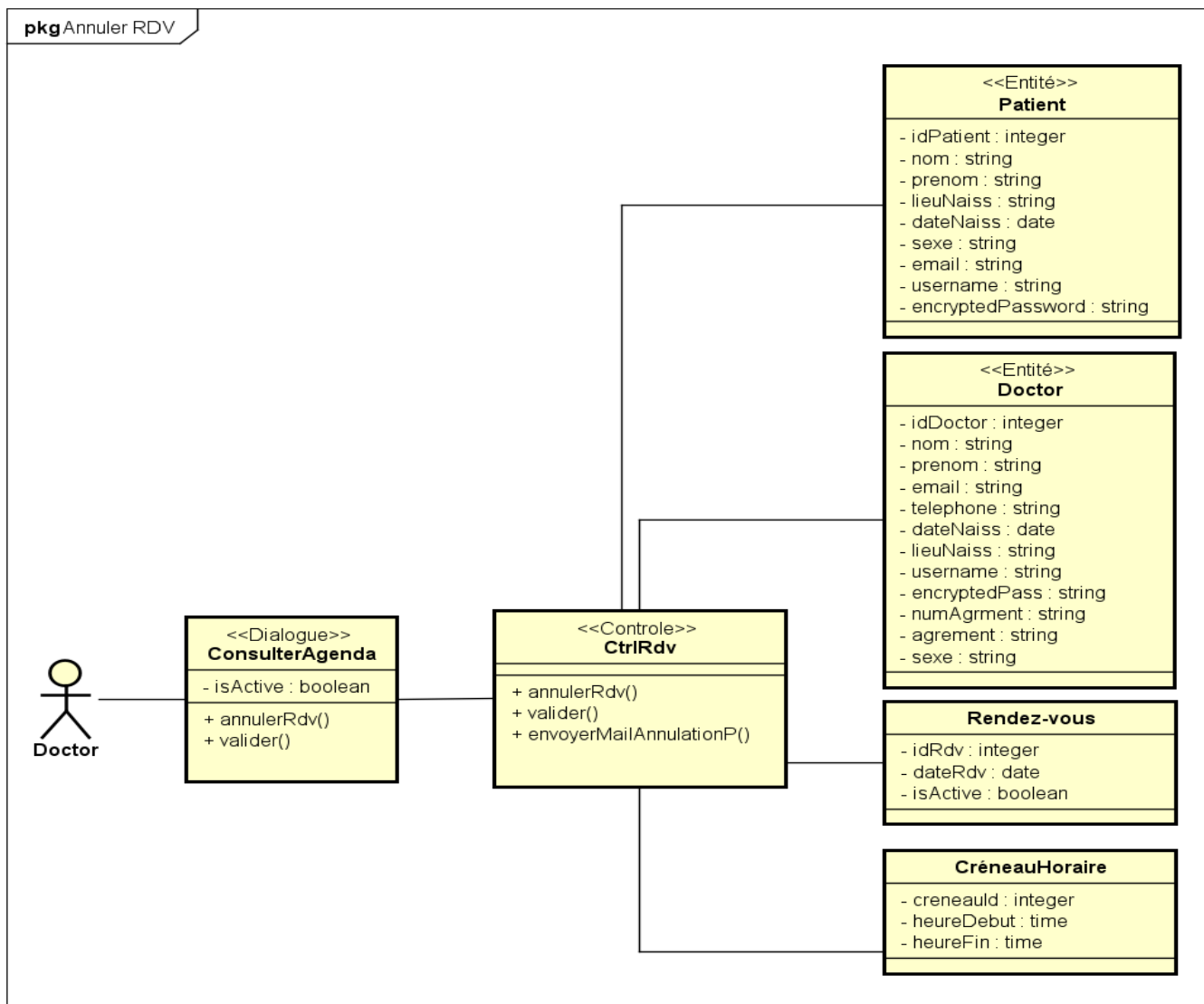


FIGURE IV.5 – Diagramme de classes participantes « Annuler Rendez-vous du Patient par le praticien »

IV.2.11 Fiche Descriptive de cas d'utilisation « Envoyer un email de Confirmation d'un Rendez-vous ou d'annulation »

Sommaire d'identification	
Nom du CU :	Envoyer un email de Confirmation ou d'annulation
Acteur :	Médecin
Type :	Primaire.
Objectif :	Permet au médecin de faire un envoi d'email automatique pour informer les patients de confirmer les rendez-vous pris en ligne ou de les annuler en cas d'indisponibilité.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. Le système affiche le Calendrier/Agenda avec qui contient Les Rendez-vous en attente du jour. 2. Le Médecin clique sur le bouton confirmé pour confirmer un rendez-vous pris pour un créneau horaire. 3. Un email sera envoyé au patient. 4. Dans le cas d'indisponibilité du médecin pour un créneau horaire il clique sur le bouton annulé. 5. Un email d'annulation sera envoyé au patient.
Post condition :	Le médecin confirme ou annule un rendez-vous avec envoi automatique d'email au patient.

IV.2.12 Fiche Descriptive du cas d'utilisations « Configurer son planning »

Sommaire d'identification	
Nom du CU :	Définir ses plages horaires d'ouvertures
Acteur :	Médecin, Système
Type :	Primaire.
Objectif :	Permet au médecin de configurer ses plages horaires d'ouvertures et ses propres congés.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. Le praticien a le choix de modifier son planning en le configurant par ses heures d'ouvertures et de fermetures 2. le praticien valide ses modifications.

IV.2.13 Diagramme d'interaction du cas d'utilisation « Configurer son Planning''

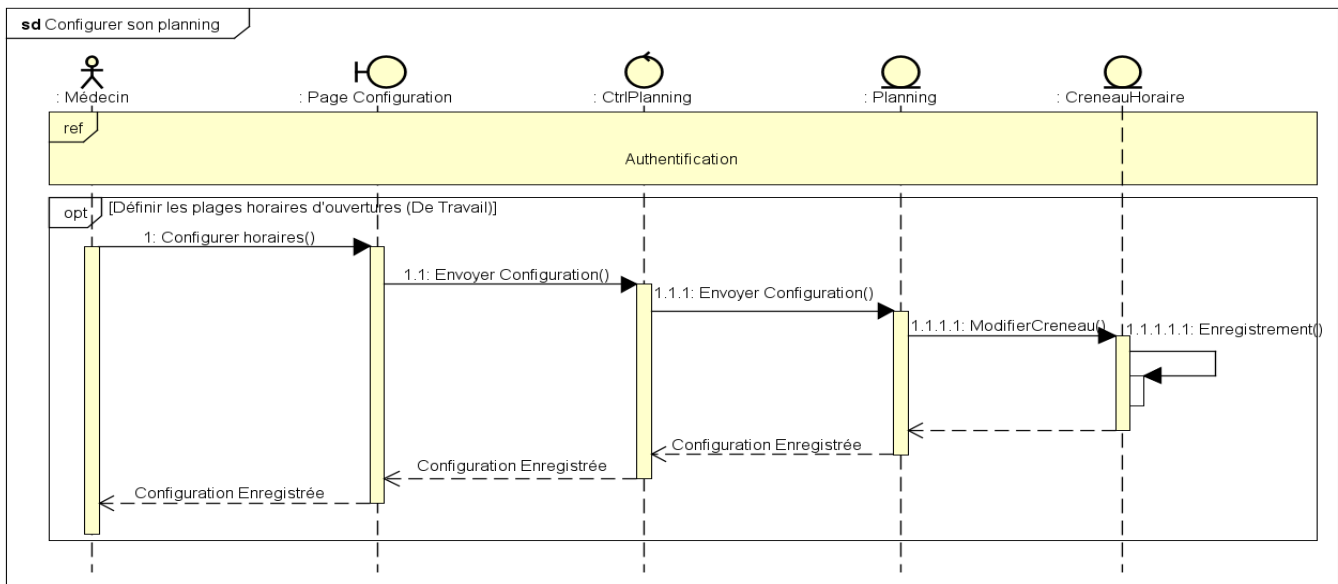


FIGURE IV.6 – Diagramme d'interaction « Configurer son planning »

IV.2.14 Fiche Descriptive du cas d'utilisation « Contacter l'administrateur »

Sommaire d'identification	
Nom du CU :	Contacteur l'administrateur.
Acteur :	Médecin, Système, Mailtrap
Type :	Primaire.
Objectif :	Permet au praticien de contacter l'administrateur en cas de besoin.
Description des scénarios	
Pré condition :	/
Scénario nominal :	<ol style="list-style-type: none"> 1. Le praticien saisit son nom et prénom ainsi que son email et son message, il valide et envois. 2. L'administrateur reçoit le message envoyé par le praticien sur sa boite mail.
Post condition :	Le médecin contacte l'administrateur avec succès.

IV.2.15 Diagramme d'interaction du cas d'utilisation "Prendre Rendez-vous"

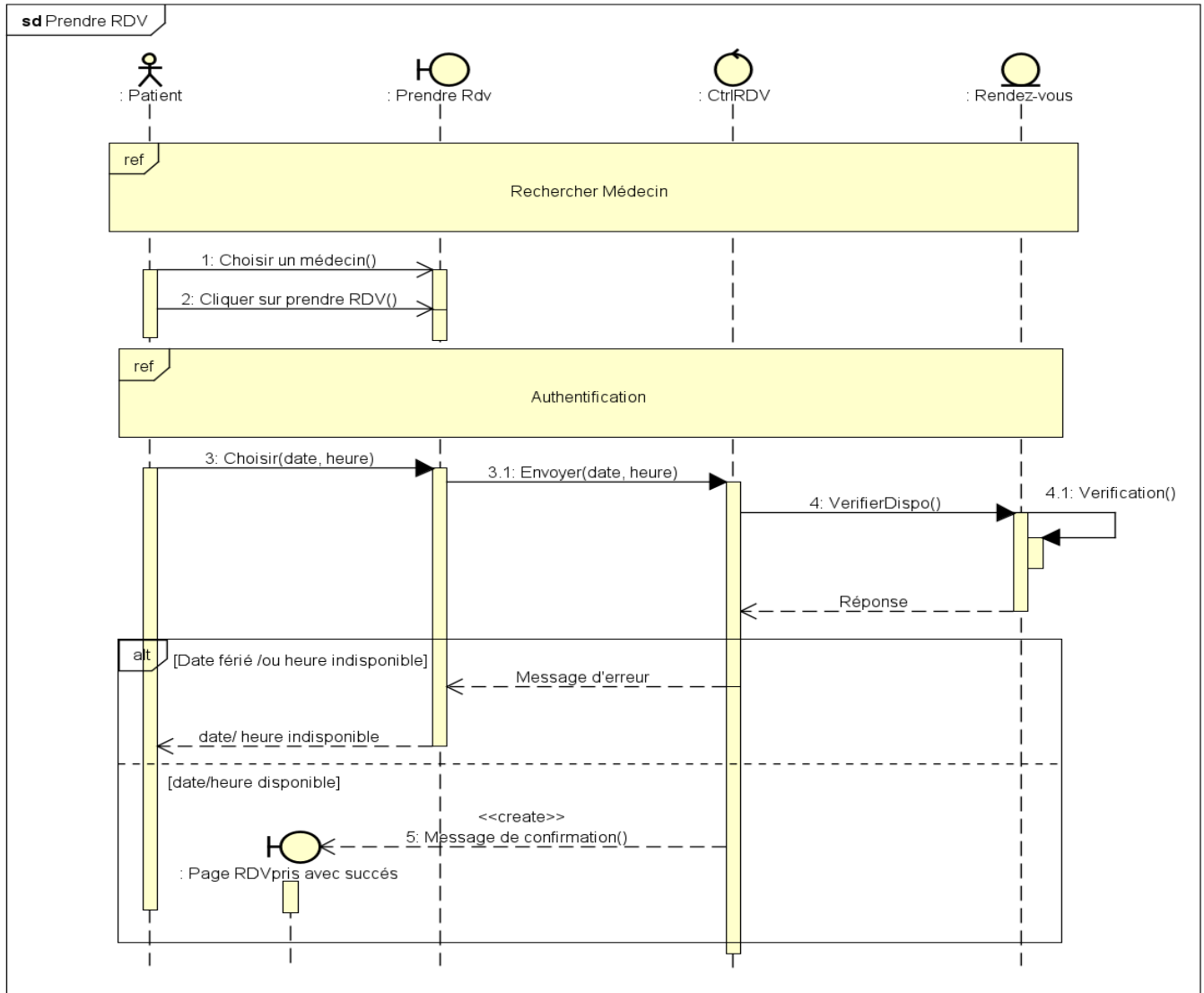


FIGURE IV.7 – Diagramme d'interaction de cas d'utilisation "Prendre Rendez-vous"

IV.2.16 Diagramme d'interaction du cas d'utilisation « Gérer Rendez-vous »

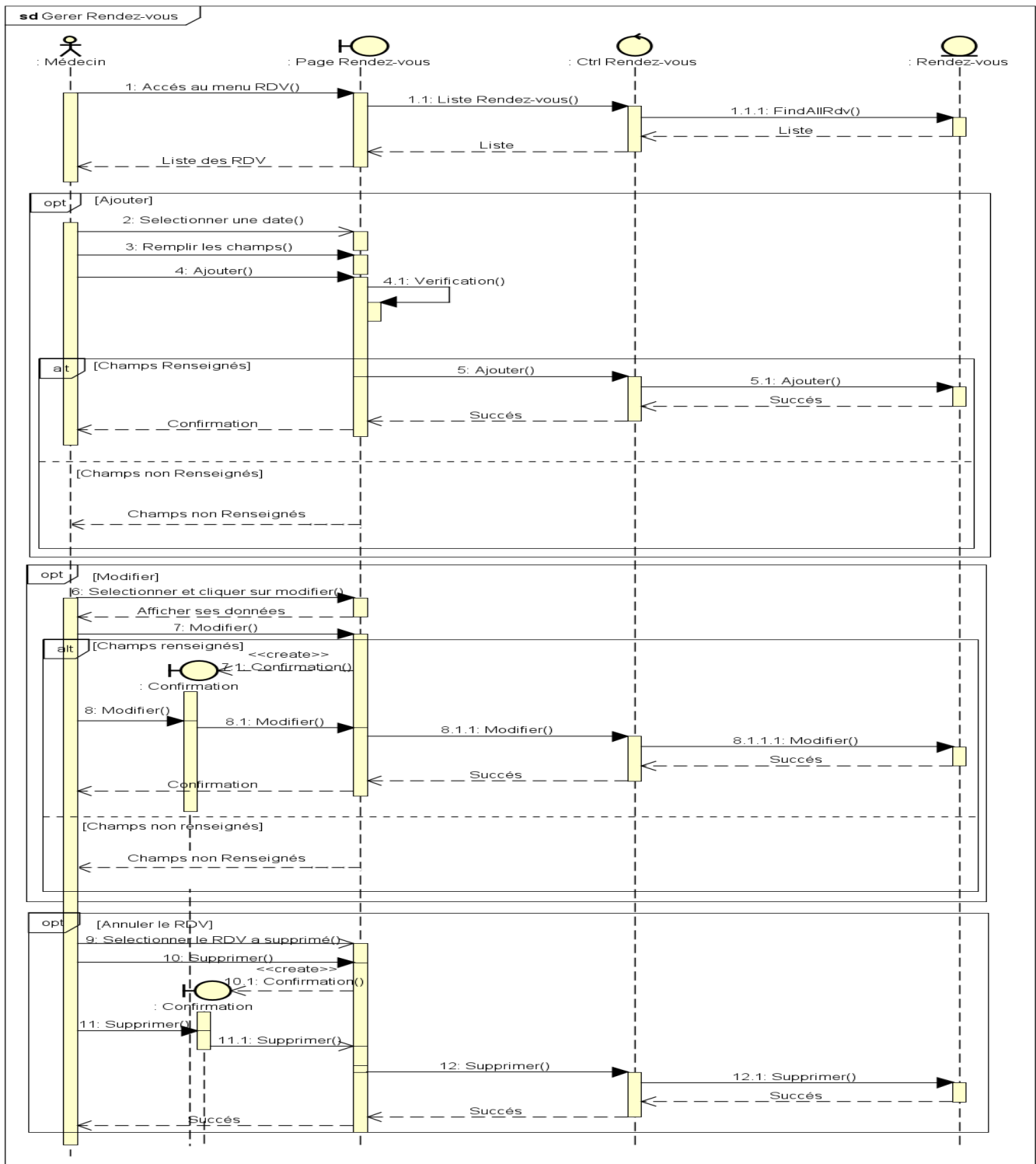


FIGURE IV.8 – Diagramme d'interaction de cas d'utilisation « Gérer Rendez-vous »

IV.3 Étude du quatrième sprint

Ce sprint s'étale sur une semaine et se compose de 4 items :

- Ajouter une nouvelle consultation ;
- Afficher la liste des consultations ;
- Supprimer une consultation ;
- Modifier une consultation.

IV.3.1 Fiche Descriptive du cas d'utilisation « Ajouter une nouvelle consultation »

Sommaire d'identification	
Nom du CU :	Ajouter une nouvelle consultation
Acteur :	Médecin.
Type :	Primaire.
Objectif :	Permet au Médecin d'ajouter une nouvelle consultation pour un patient
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. Le Médecin accède a la session des consultations. 2. Le Système affiche la fenêtre des consultations. 3. Le Médecin Recherche un patient. 4. le Système lui retourne les informations du patients ainsi que sa liste de consultations. 5. le Médecin remplit les champs a ajouté dans une nouvelle consultation et valide. 6. La nouvelle consultation est ajoutée avec succès.
Scénario Alternatif	<ol style="list-style-type: none"> 2.a. si Aucun patient ne convient au nom saisi le Système ne retourne rien.

Post condition :	Le médecin ajoute une nouvelle consultation avec succès.
-------------------------	--

TABLE IV.9: Description textuelle du cas d'utilisation
« Ajouter une nouvelle consultation ».

IV.3.2 Diagramme de classes participante pour le cas d'utilisation « Ajouter une consultation »

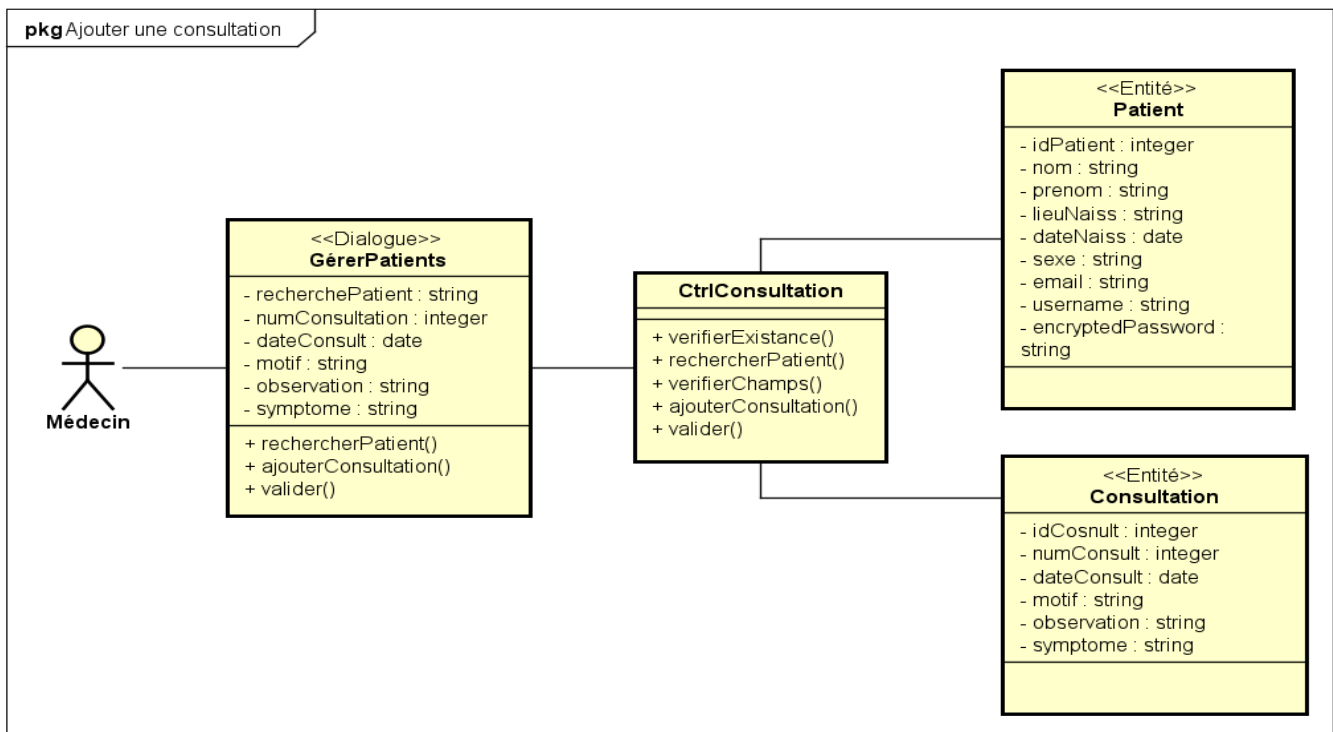


FIGURE IV.9 – Diagramme de classes participantes de cas d'utilisation « Ajouter une consultation »

IV.3.3 Fiche Descriptive du cas d'utilisation « Supprimer une consultation »

Sommaire d'identification	
Nom du CU :	Supprimer une consultation
Acteur :	Médecin.
Type :	Primaire.
Objectif :	Permet au Médecin de supprimer une consultation.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. Le Médecin accède a la session des consultations. 2. Le Système affiche la fenêtre des consultations. 3. Le Médecin Recherche un patient. 4. le Système lui retourne les informations du patients ainsi que sa liste de consultations. 5. le Médecin sélectionne une consultation et la supprime. 6. La consultation est supprimée avec succès.
Scénario Alternatif	2.a. si Aucun patient ne convient au nom saisit le Système ne retourne rien.
Post condition :	Le médecin supprime une consultation avec succès.

TABLE IV.10: Description textuelle du cas d'utilisation « Supprimer une consultation ».

IV.3.4 Fiche Descriptive du cas d'utilisation « Modifier une nouvelle consultation »

Sommaire d'identification	
Nom du CU :	Ajouter une nouvelle consultation
Acteur :	Médecin.
Type :	Primaire.
Objectif :	Permet au Médecin de modifier une consultation.
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. Le Médecin accède a la session des consultations. 2. Le Système affiche la fenêtre des consultations. 3. Le Médecin Recherche un patient. 4. le Système lui retourne les informations du patients ainsi que sa liste de consultations. 5. le Médecin sélectionne une consultation et modifie les informations souhaitées. 6. La consultation est modifiée avec succès.
Scénario Alternatif	2.a. si Aucun patient ne convient au nom saisi le Système ne retourne rien.
Post condition :	Le médecin modifie une consultation avec succès.

TABLE IV.11: Description textuelle du cas d'utilisation « Modifier une consultaiton ».

IV.4 Étude du Sprint 5

Ce sprint s'étale sur une semaine et se compose de 8 items :

- Consulter la liste des ordonnances.

- Supprimer Ordonnance.
- Modifier Ordonnance.
- Ajouter Ordonnance.
- Imprimer Ordonnance.

Tout au long de cette partie nous traitons les histoires utilisateurs de ce sprint pour produire un incrément potentiellement livrable.

IV.4.1 Fiche Descriptive du cas d'utilisation « Consulter la liste des ordonnances »

Sommaire d'identification	
Nom du CU :	Consulter la liste des ordonnances
Acteur :	Médecin.
Type :	Primaire.
Objectif :	Permet au Médecin de consulter la liste des ordonnances pour un patient
Description des scénarios	
Pré condition :	Authentification
Scénario nominal :	<ol style="list-style-type: none"> 1. Le Médecin accède a la session des Patients. 2. Le Système affiche la fenêtre des patients. 3. Le Médecin Recherche un patient. 4. le Système lui retourne les informations du patient ainsi que sa liste des ordonnances.
Scénario Alternatif	<ol style="list-style-type: none"> 2.a. si Aucun patient ne convient au nom saisi le Système ne retourne rien.
Post condition :	Le médecin consulte la liste des ordonnances pour un patient.

TABLE IV.12: Description textuelle du cas d'utilisation
« Consulter la liste des ordonnances. ».

IV.4.2 Fiche Descriptive du cas d'utilisation « Éditer Automatiquement l'ordonnance médicale »

Sommaire d'identification	
Nom du CU :	Éditer automatiquement une nouvelle Ordonnance Médicale
Acteur :	Médecin.
Type :	Primaire.
Objectif :	Permet au Médecin d'éditer automatiquement une ordonnance Médicale
Description des scénarios	
Pré condition :	Authentification et Rechercher Patient
Scénario nominal :	<ol style="list-style-type: none"> 1. Le Médecin Recherche un médicament avec le nom. 2. Le Système retourne le médicament trouvé 3. Le Médecin sélectionne le médicament. 4. Le Médecin sélectionne la prescription et valide. 5. Le Système Ajoute automatiquement les informations sélectionnées dans l'ordonnance médicale et l'enregistre dans l'historique des ordonnances.
Scénario Alternatif	
Post condition :	Le Médecin édite automatiquement une ordonnance Médical

TABLE IV.13: Description textuelle du cas d'utilisation
« Éditer une Ordonnance Médicale »

IV.4.3 Diagramme d'interaction du cas d'utilisation « Éditer automatiquement l'ordonnance »

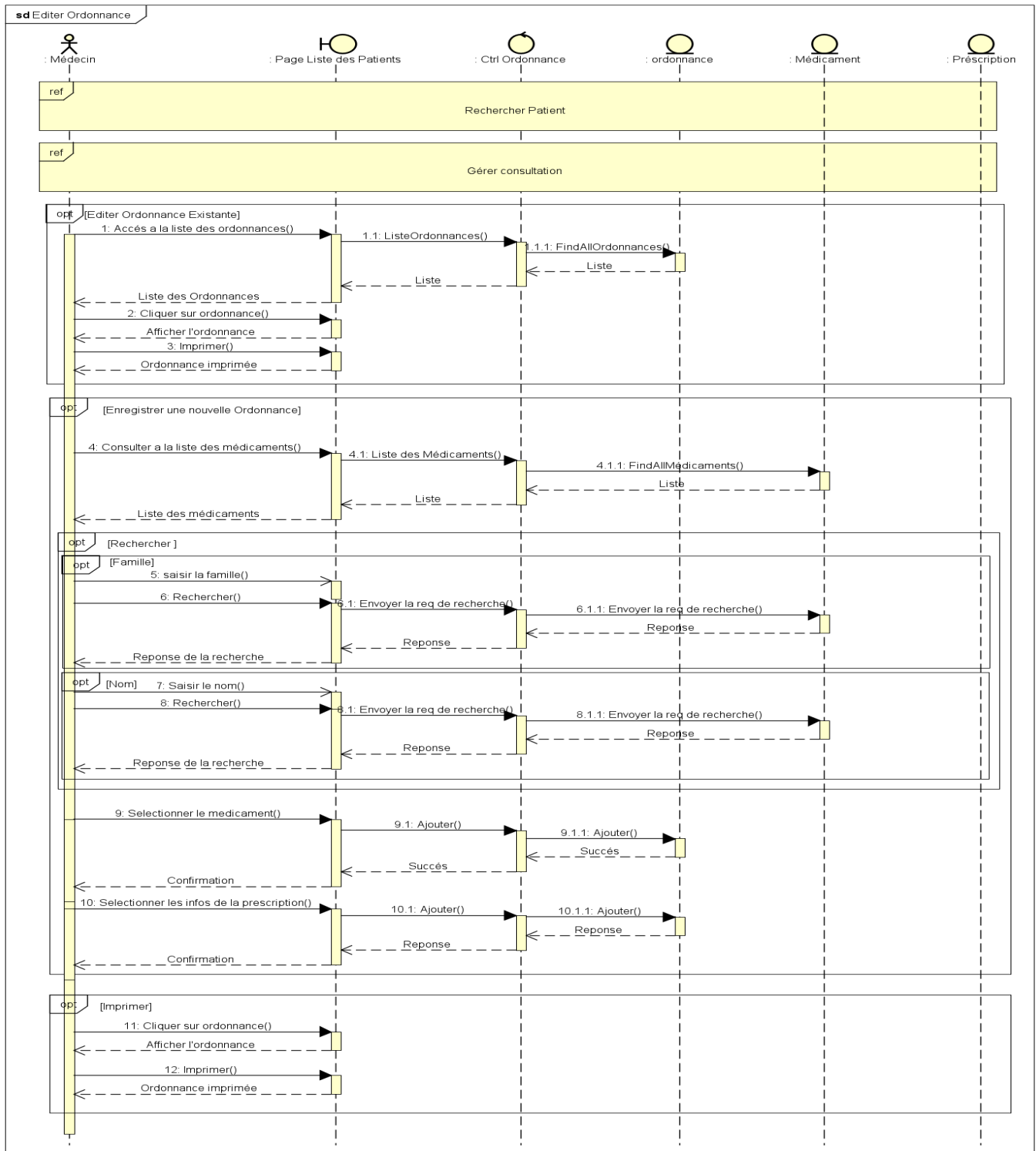


FIGURE IV.10 – Diagramme d'interaction du cas d'utilisation « Éditer automatiquement l'ordonnance »

IV.4.4 Diagramme de classes participantes du cas d'utilisation « Imprimer Ordonnance »

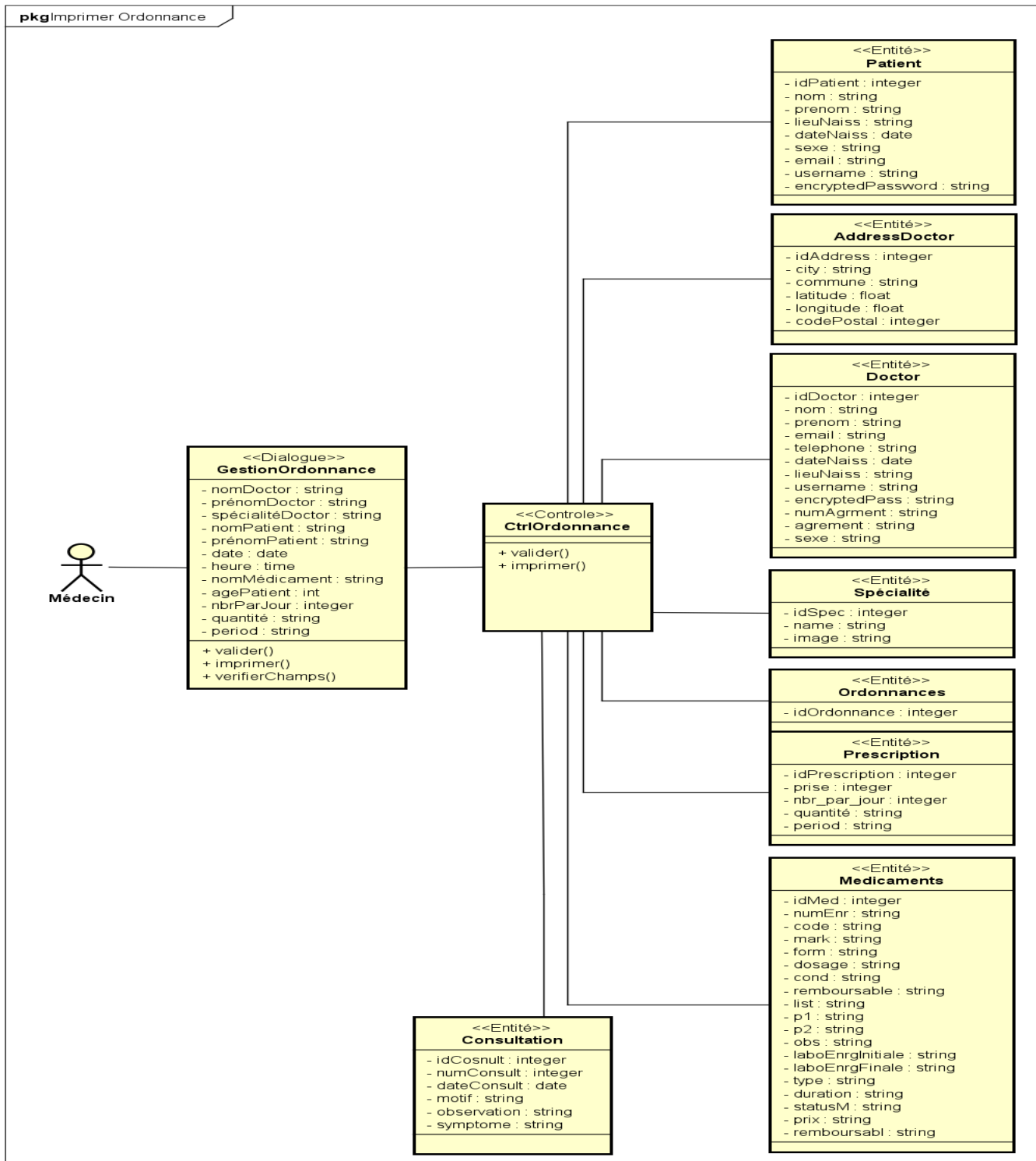


FIGURE IV.11 – Diagramme de classes participantes de cas d'utilisation « Imprimer Ordonnance »

IV.5 Étude du Sprint 6 : Dossier Médical

Ce sprint s'étale sur 4 semaines et qui se compose de 3 items qui sont :

- Consulter le dossier médical (Par le Praticien - Par le Patient) ;
- Remplir le Dossier médical ;
- Configurer les droits d'accès au dossier médical

IV.5.1 Fiche Descriptive du cas d'utilisation « Consulter le dossier Médical du Patient »

Le Médecin Peut consulter le dossier Médical du patient si seulement s'il était son médecin Traitant (Créateur du dossier Médical).

Le Médecin Créateur du dossier Médical est unique alors il possède le droit de partager le dossier avec un autre médecin en cas de nécessité.

Si Le patient ne Possède aucun dossier Médical alors c'est a son Médecin traitant de le remplir.

Sans droit d'accès aucun autre médecin ne pourra lire les informations du dossier, ces dernières seront chiffrés.

IV.5.2 Fiche Descriptive du cas d'utilisation « Consulter le dossier Médical du Patient »

Sommaire d'identification	
Nom du CU :	Consulter le dossier médical du patient
Acteur :	Médecin.
Type :	Primaire.
Objectif :	Permet au Médecin de consulter le dossier médical de son patient
Description des scénarios	
Pré condition :	Authentification et Rechercher Patient

Scénario nominal :	<ol style="list-style-type: none"> 1. Le Médecin traitant accède au volet du Dossier Médical existant dans la session Patients. 2. Le système affiche toutes les informations du dossier médical.
Scénario Alternatif	<ol style="list-style-type: none"> 2.a Si aucune information n'est retournée alors le médecin traitant est invité a remplir en premier le dossier médical du patient.
Post condition :	Le Médecin Consulte le dossier médical du patient

TABLE IV.14: Description textuelle du cas d'utilisation
« Consulter le dossier Médical du Patient »

IV.5.3 Diagramme d'interaction du cas d'utilisation « Consulter le dossier Médical du patient »

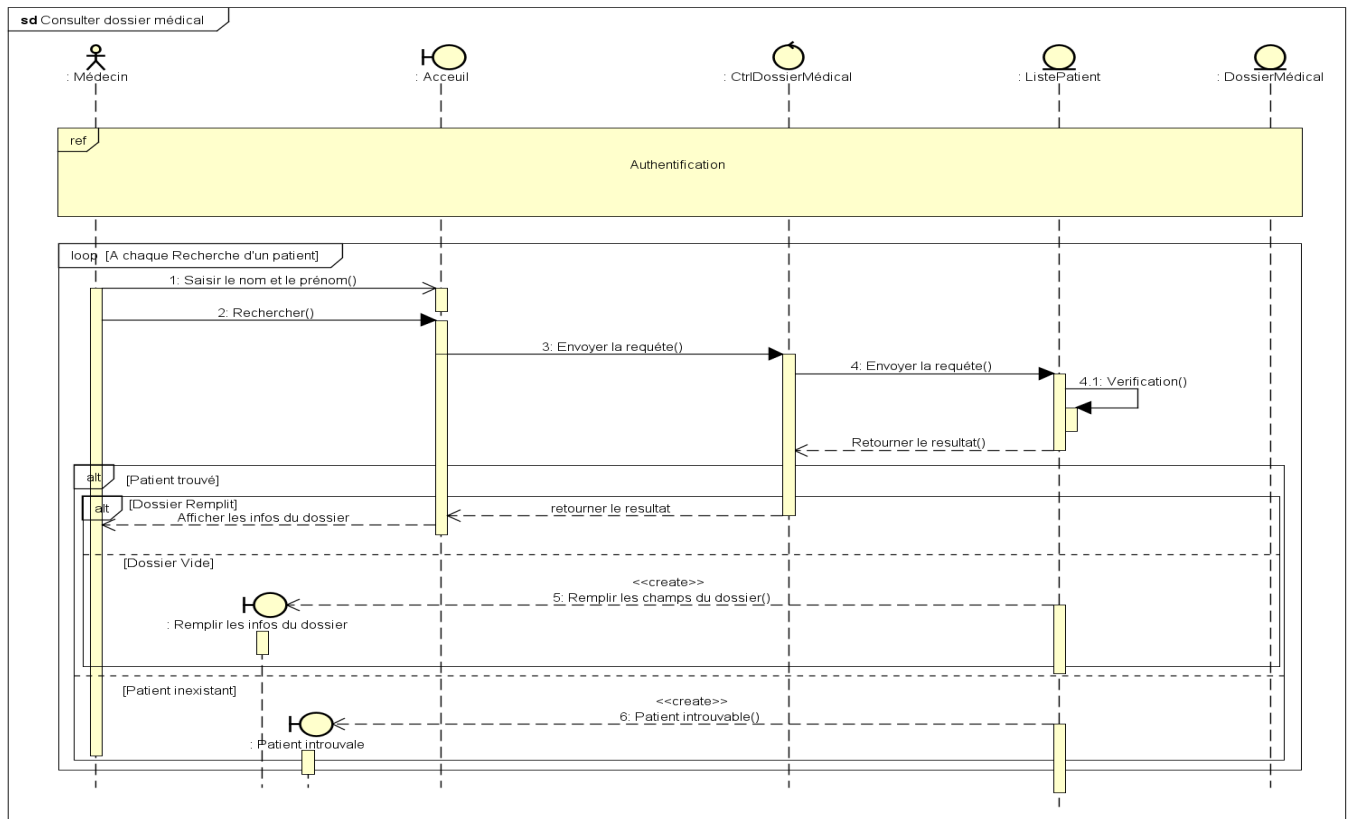


FIGURE IV.12 – Diagramme d'interaction du cas d'utilisation « Consulter le dossier Médical du patient »

IV.5.4 Fiche Descriptive du cas d'utilisation « Configurer les droits D'accès au Dossier médical »

Sommaire d'identification	
Nom du CU :	Configurer les droits d'accès au dossier Médical
Acteur :	Médecin.
Type :	Primaire.
Objectif :	Permet au Médecin Traitant de configurer les droits d'accès au dossier Médical
Description des scénarios	
Pré condition :	Authentification et Rechercher Patient
Scénario nominal :	<ol style="list-style-type: none"> 1. Le Médecin traitant accède au volet Dossier Médical existant dans la session patient. 2. Le système lui affiche les informations du dossier médical ainsi que la configuration des droits d'accès a ce dossier. 3. Le Médecin Traitant recherche un autre médecin par son nom et sa spécialité, et choisit le statut d'accès (il a la possibilité de sélectionner plusieurs médecins avec lesquels il partage le dossier médical du patient et valide la configuration. 4. Le Système donne l'accès aux médecins choisis de consulter les informations Du dossier médical
Scénario Alternatif	3.a si le médecin recherché n'existe pas alors le système ne retourne aucun résultat
Post condition :	Le Médecin configure les droit d'accès au dossier médical

TABLE IV.15: Description textuelle du cas d'utilisation
« Configurer les droits d'accès au dossier Médical »

IV.5.5 Diagramme d'interaction du cas d'utilisation « Configurer les droits d'accès au dossier Médical »

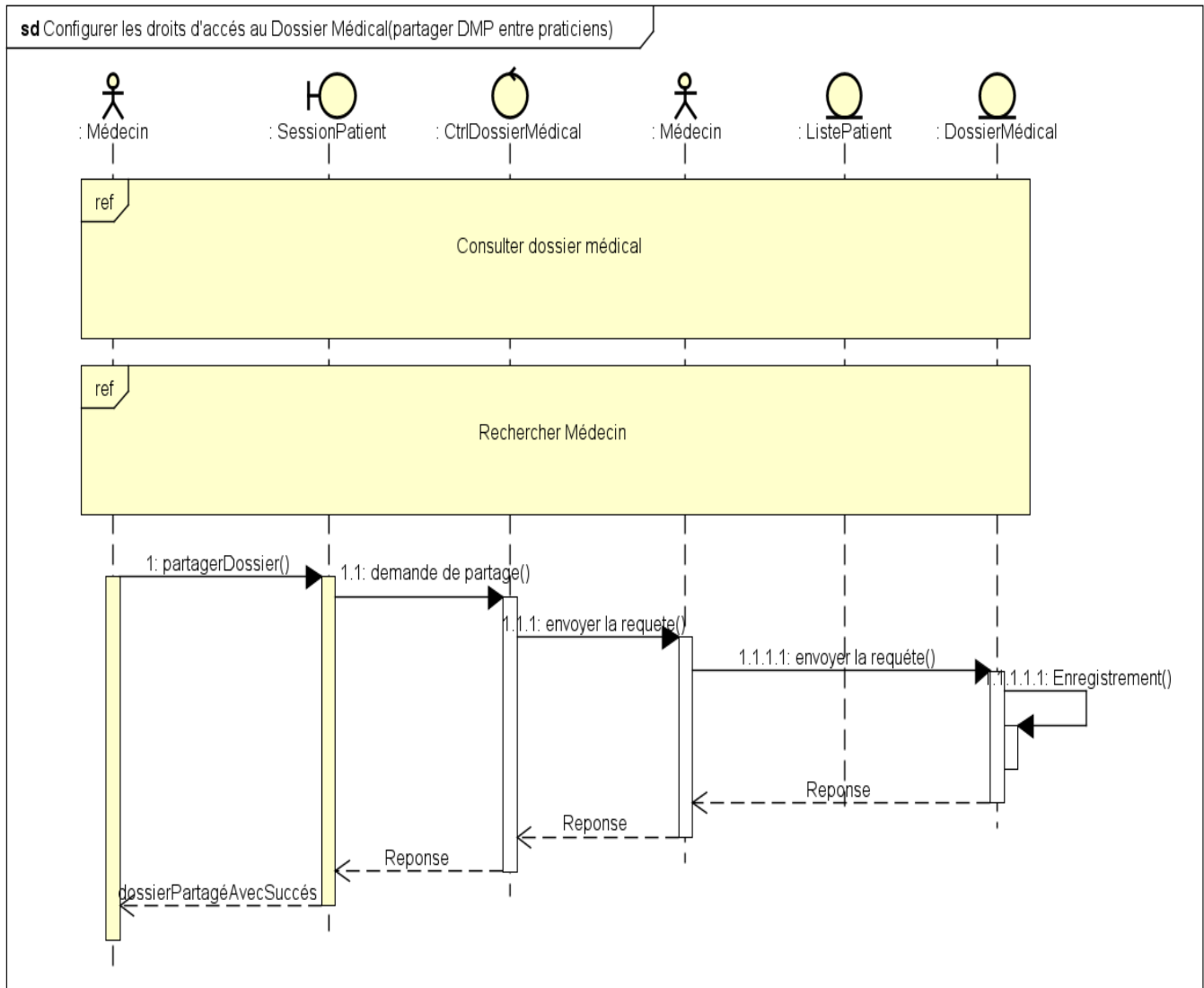


FIGURE IV.13 – Diagramme d'interaction du cas d'utilisation « Configurer les droits d'accès au dossier médical »

IV.6 Conclusion

A la fin de ce chapitre nous avons réussi à produire un incrément répondant aux besoins du client en étant utilisé dans un environnement de production. Dans le chapitre qui suit, nous présentons la partie de réalisation.

Chapitre V

Réalisation

V.1 Introduction

Dans ce chapitre nous entamons la partie pratique, ou nous allons présenter l'environnement et les outils de développement utilisés, l'architecture de l'application et quelques interfaces de celui-ci.

V.2 Diagramme de classes

Le diagramme de classe est une description statique du système focalisé sur le concept de classe et d'association. Une classe représente un ensemble d'objets qui possèdent des propriétés similaires et des comportements communs décrivant en terme d'attributs et d'opérations. Une association consiste à présenter les liens entre les instances de classe.

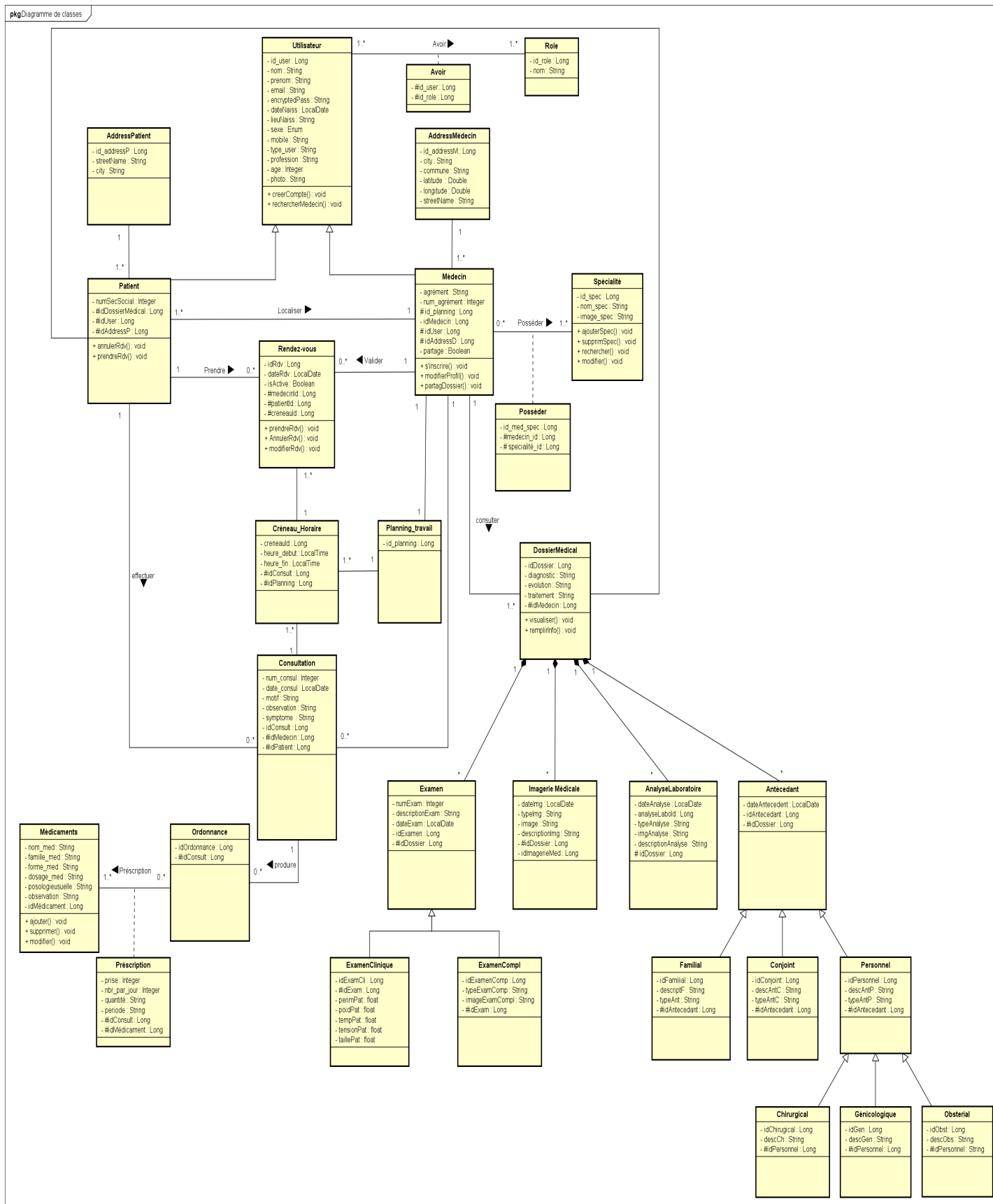


FIGURE V.1 – Diagramme de classes

V.3 Élaboration du Dictionnaire de Données

Classes	Attributs	Désignation	Type	Taille
Utilisateur	id_user	Identifiant de l'utilisateur	integer	
	nom	nom de l'utilisateur	varchar	50
	prenom	Prénom de l'utilisateur	varchar	50
	email	Émail de l'utilisateur	varchar	50
	userName	Nom utilisateur	varchar	50
	enchrptPsd	Mot de passe chiffré	varchar	50
	dateNaiss	date de naissance	date	
	lieuNaiss	lieu de naissance	varchar	50
	sexe	sexe de l'utilisateur (male/female)	varchar	50
	mobile	numéro du téléphone	varchar	50
	profession	profession de l'utilisateur	varchar	50
	age	âge de l'utilisateur	integer	
	photo	photo de l'utilisateur	varchar	50
Patient	IdPatient	Identifiant du Patient	integer	
	numSecSoc	Numéro de sécurité social	String	50
Médecin	IdMed	Identifiant du médecin	integer	
	agrement	image de l'agrément	varchar	100
	numAgrm	numéro de l'agrément	integer	
AddressMedecin	idAdrsMed	Identifiant de l'adresse du médecin	integer	
	city	ville du médecin	varchar	50
	commune	commune du médecin	varchar	50
	latitude	latitude de l'adresse du medecin	float	
	longitude	longitude de l'adresse du médecin	float	
AddressPatient	idAdrsPat	Identifiant de l'adresse du patient	integer	
	streetName	le nom de la rue de l'address du patient	varchar	10
	city	La ville dont l'adresse du patient se situe	integer	
Specialité	idSpec	Identifiant de la spécialité	int	
	nomSpec	Le nom de la spécialité	varchar	50
	imageSpec	Image de la spécialité	varchar	
Planning	idPlanning	Identifiant du planning du médecin	integer	
	nomPlan	Le nom du planning	varchar	50
Rendez-vous	idRdv	Identifiant du Rendez-vous	integer	
	dateRdv	La date du Rendez-vous	date	
	isActive	pour confirmer ou annuler un rendez-vous	varchar	

CréneauHoraire	creneauId	Identifiant du Créneau horaire	int	
	heureDebut	heure de début du créneau horaire	time	
	heureFin	heure de fin du créneau horaire	time	
Consultation	idConsul	Identifiant du Consultation	int	
	numConsul	numéro de la consultation	date	
	dateConsul	date de la consultation	date	
	motif	motif de la consultation	varchar	100
	symptome	les symptômes de la consultation	varchar	100
	observation	observation du patient pendant la consultation	varchar	100
DossierMédical	idDossier	Identifiant du dossier médical	integer	
	diagnostic	diagnostic du patient	varchar	
	évolution	évolution des maladies du patient	varchar	
	traitement	traitement du patient	varchar	100
Examen	IdExamen	Identifiant de l'examen	integer	
	DescExam	La description de l'examen	varchar	100
	numExam	Numero de l'examen	integer	
	dateExam	date de l'examen	date	
ExamClin	IdExClin	Identifiant de l'examen clinique	integer	
	perimPat	le périmètre du patient	float	
	poidPat	le poid du patient	float	
	ttmpPatient	la température du patient	float	
	tensionPat	la tension du patient	float	
	taillePat	la taille du patient	float	
ExamCompl	IdExCom	Identifiant de l'examen complémentaire	integer	
	typeExCom	Type de l'examen complémentaire (Scanner, IRM..)	varchar	100
	imgExCom	Image de l'examen complémentaire	float	
ImagerieMedicale	idImgMed	Identifiant de l'imagerie médicale	int	
	dateImg	Date de l'image	Date	
	titreImg	Titre de l'image	varchar	80
	descrImg	Description du resultat l'image	varchar	
AnalyseLabo	idAnalyse	Identifiant des Analyses du laboratoire	int	
	dateAnalyse	Date des analyses	Date	
	typeAnalyse	Le type des analyses	varchar	80
	imgAnalyse	Image des analyses	varchar	100
	descAnlys	Description des résultats des analyses	varchar	200
Antécédent	idAnt	Identifiant de l'antécédent	integer	
	dateAnt	Date de l'antécédent	Date	

Familial	idFamilial	Identifiant de l'antécédent familial	int	
	descFam	Description de l'antécédent familial	varchar	200
	typeFam	Type de l'antécédent familial	varchar	100
Conjoint	idConjoint	Identifiant de l'antécédent du conjoint	integer	
	descCon	Description de l'antécédent du conjoint	varchar	200
	typeConj	Type de l'antécédent du Conjoint	varchar	100
Personnel	idPersonnel	Identifiant d'antécédent Personnel	integer	
	descPers	Description de l'antécédent personnel	varchar	200
	typePers	Type de l'antécédent personnel	varchar	100
Chirurgical	idChirurg	Identifiant d'antécédent chirurgical	integer	
	descChirg	Description de l'antécédent chirurgical	varchar	200
Gynécologique	idGyn	Identifiant d'antécédent gynécologique	integer	
	descGyn	Description de l'antécédent gynécologique	varchar	200
Obsterical	idObst	Identifiant d'antécédent obstétrical	int	
	descObs	Description de l'antécédent obstétrical	varchar	200
Ordonnance	idOrd	Identifiant de l'ordonnance médical	int	
Médicament	IdMed	Identifiant du médicament	int	
	nomMed	Nom du médicament	varchar	50
	familleMed	Famille du médicament	varchar	50
	formeMed	Forme du médicament	varchar	50
	dosageMed	Dosage du médicament	float	50
	posologie	posologie usuelle du médicament	varchar	50
Prescription	CodeMdt	Code du médicament	integer	
	formMed	Forme d'un médicament	varchar	50
	nbrParJour	Nom du médicament	varchar	100
	qtMed	Quantité a prendre d'un médicament	integer	
	periode	la période (début et fin) de prise de médicament	varchar	80

TABLE V.1: Dictionnaire de données.

V.4 JPA (Java Persistence API)

V.4.1 Les relations avec JPA

Une des fonctionnalités majeures des ORM est de gérer les relations entre objets comme des relations entre tables dans un modèle de base de données relationnelle. JPA définit des modèles de relation qui

peuvent être déclarés par annotation.

Les relations sont spécifiées par les annotations : **@OneToOne**, **@ManyToOne**, **@OneToMany**, **@ManyToMany**[13].

V.4.2 Schéma Relationnel

- **Utilisateur** (idUser, nom ,prénom, email, dateNaiss, lieuNaiss, age, sexe, mobile, type_ pers, login, encryptPassword, profession, photo).
- **Role** (idRole, nom).
- **Avoir** (# idUser,# idRole).
- **Patient** (idPatient, numSecSoc, # idUser, # idDossier, #addressPatientId).
- **AddressPatient** (idAddressP, streetName, city).
- **Médecin** (idMedecin, numAgrement, agrement, # idUser, # idPlanning, # addressDoctorId).
- **AddressMedecin** (idAddressM, streetName, city, commune, codePostal, latitude, longitude).
- **Spécialité** (idialit, nomSpec, imageSpec).
- **MédSpec** (idpe # ideci #idialit).
- **Rendez-vous** (idRdv, dateRdv, isActive, # idMédecin, # idPatient, # creneauId).
- **Planning** (idPlanning, nom).
- **CréneauHoraire** (idCreneau, heureDebut, heureFin, #idPlanning, #idConsult).
- **Consultation** (idConsult, numConsult, motif, observation, symptome, # idMédecin, # idPatient).
- **Ordonnance** (idOrdonnance, # idConsult)
- **Médicament** (idicame, nomMéd, familleMéd, formeMéd, dosageMéd, posologieUsuelle, observation)
- **Prescrire** (idription # idOrdonnance , # idicame, prise, nbrParJour, quantité, periode)
- **DossierMedical** (idDossier, diagnostic, evolution, traitement, # idMedecin)
- **Examen** (idExamen, numExam, descriptionExam, dateExam, #idDossier)
- **ExamenClinique** (idExamCl, perimPat, poidPat, taille, temperaturePat, tensionPat, # idExamen)
- **ExamenComplémentaire** (idExamComp, typeExamCom, descrCom, imageExam, #**idExamen**).
- **Imagerie Médicale** (idImagerieica, dateImg, typeImg, image, descriptionImg, # idDossier).
- **AnalyseLaboratoire** (analyseLaboId, dateAnalyse, typeAnalyse, imgAnalyse, descriptionAnalyse, #idDossier).
- **Antecedant** (idAntecedant,dateAnt, #idDossier).
- **Familial** (idFamilial, descriptionF, typeAntF, #idAntecedant).
- **Conjoint** (idConjoint, descAntC, typeAntC, #idAntecedant).

- **Personnel** (idPersonnel, descAntP, typeAntP, #idAntecedant).
- **Chirurgical** (idChirurgical, desCh, #idPersonnel).
- **Gynécologique** (idGen, descGen, #idPersonnel).
- **Obstercial** (idObst, descObs, #idPersonnel).

V.5 Langages et environnement de développement

• Spring Boot

Spring Boot est un framework de développement JAVA. C'est une déclinaison du framework classique de Spring qui permet essentiellement de réaliser des microservices (ce sont la majeure partie du temps des services web qui sont regroupés en API web) [19].

Chaque projet Spring Boot est composé de bibliothèques dont votre projet a besoin et de rien d'autre ! Ainsi, chaque projet Spring Boot embarque uniquement le strict nécessaire. De manière générale, le framework propose en standard tout ce qui est nécessaire pour :

- Se connecter à toute base de données ;
- Sécuriser son API ;
- Créer des services web performants ;
- Distribuer la configuration entre les microservices.

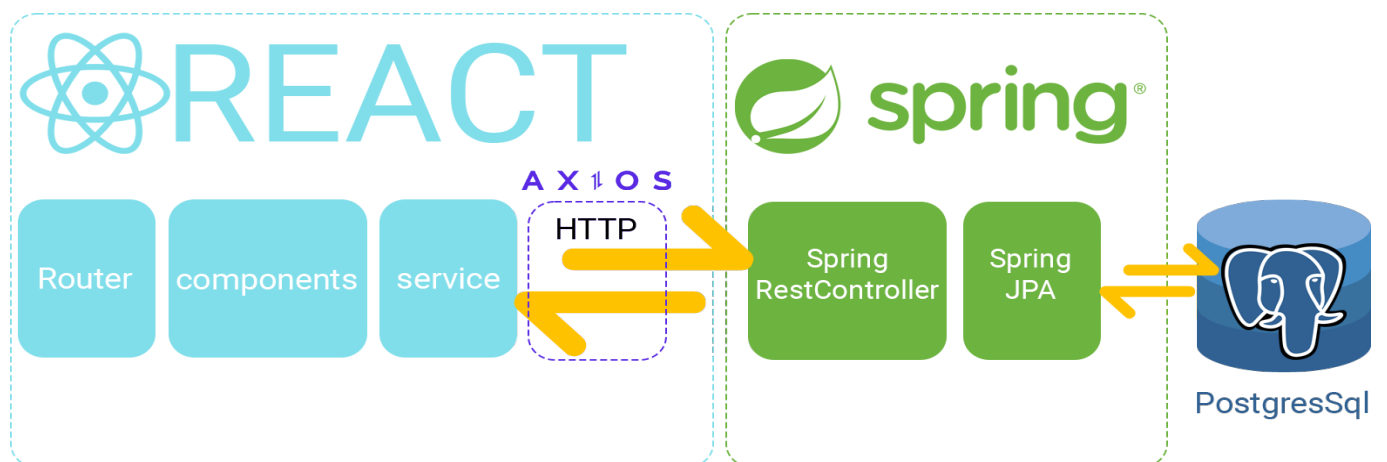


FIGURE V.2 – Architecture de notre application

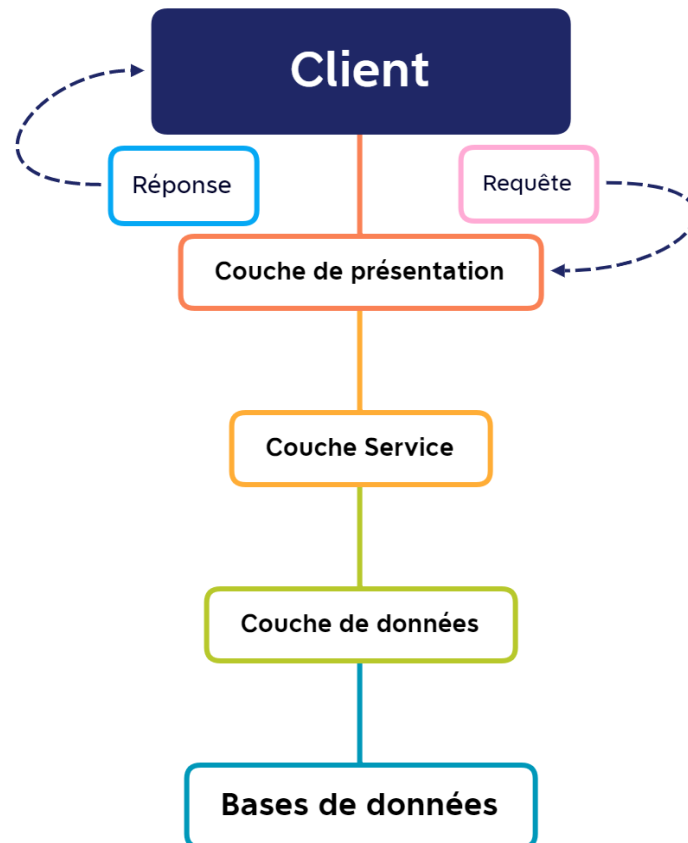


FIGURE V.3 – Représentation des couches de spring boot

Lorsque on utilise Spring ou Spring boot, on utilise une architecture 3 tiers : La couche de présentation, La couche Service et la couche de données.

1. **La couche Présentation (Presentation Layer)** : représenté via le contrôler, c'est elle qui communique avec le client soit réceptionner ou envoyer des données ce qu'on appelle Request/Response.
2. **La couche Service (Service Layer)** : contient le traitement ou le code métier.
3. **La couche de données (Data Layer)** : c'est elle qui communique avec la BDD via le repository.représenté par le JPA ou le Hibernate.

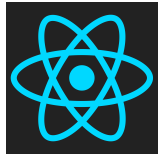
• JHipster

JHipster est un générateur d'application libre et open source utilisé pour développer rapidement des applications Web modernes en utilisant Angular et le framework Spring [5].



• React js

React est une bibliothèque JavaScript déclarative, efficace et flexible pour construire des interfaces utilisateurs (UI). Elle nous permet de composer des UI complexes à partir de petits morceaux de code isolés appelés « composants ».



• Angular js

Angular JS est un framework JavaScript libre et open source2 développé par Google. Il permet de développer des pages web [7].



• Gitkraken

Gitkraken Git GUI est un client graphique multiplateforme puissant et intuitif qui simplifie amplement l'utilisation de Git. De manière très simple, on peut pratiquement tout faire avec GitKraken sans une seule ligne de commande à taper [11].



• Asana

Il s'agit d'un outil qui permet de planifier, d'organiser et de gérer le travail de l'équipe de développement de A à Z. C'est un bon logiciel de gestion de projet qui fait également office d'outil de collaboration : il coordonne les tâches d'équipe pour que tous les membres connaissent les responsabilités de chacun, il facilite aussi le partage des commentaires, des fichiers et des mises à jour de statut et également une vue d'ensemble du travail pour que les équipes s'attèlent aux bonnes tâches au bon moment [8].



• IntelliJ IDEA

IntelliJ IDEA est un IDE intelligent et tenant compte du contexte qui permet de travailler sur toutes sortes d'applications en Java et dans d'autres langages de la JVM tels que Kotlin, Scala et Groovy.

De plus, IntelliJ IDEA Ultimate vous aide à développer des applications web full-stack grâce à ses puissants outils intégrés, à la prise en charge de JavaScript et de ses technologies connexes et à la prise en charge avancée de frameworks populaires tels que Spring, Spring Boot, Jakarta EE, Micronaut, Quarkus et Helidon. IntelliJ IDEA peut être complété par des plugins gratuits développés par JetBrains afin de pouvoir travailler avec d'autres langages de programmation, parmi lesquels Go, Python, SQL, Ruby et PHP [3].



• Astah Uml

Astah Professional est un outil complémentaire pour le langage UML, prêt à être utilisé dès son téléchargement, ce qui permet de de l'améliorer rapidement et facilement, et d'affiner le processus de développement avec l'approche de modélisation innovante d'Astah [9].



• Postman

Postman est un logiciel permettant de créer et de tester des requêtes HTTP [15]. Il vous permet de les personnaliser dans les plus fins détails grâce à une interface ergonomique et intuitive. Vous pouvez choisir la méthode de la requête, entrer l'URL du serveur que vous voulez interroger, et rajouter tous les paramètres possibles pour une requête HTTP. Le logiciel tient un historique de vos requêtes. Il est très utile pour tester une api. Nous l'utiliserons pour s'initier au protocole HTTP en créant différentes requêtes.



• WebStorm

c'est le plus intelligent des IDE JavaScript.

WebStorm est un environnement de développement intégré pour JavaScript et les technologies connexes. Comme les autres IDE de JetBrains, il rend votre expérience de développement plus conviviale, en automatisant les tâches répétitives et en vous aidant à gérer les tâches complexes avec facilité [17].



• PostgreSQL

PostgreSQL est un système de gestion de base de données relationnelle orienté objet puissant et open source qui est capable de prendre en charge en toute sécurité les charges de travail de données les plus complexes. Alors que MySQL donne la priorité à l'évolutivité et aux performances, Postgres donne la priorité à la conformité et à l'extensibilité SQL [14].



• Google meet

Google Meet est le service de visioconférence de Google conçu pour les professionnels qui permet aux entreprises d'organiser facilement des vidéoconférences [12]. La création de réunions est très facile il suffit de transmettre un lien généré automatiquement à vos contacts, les participants ont juste besoin de posséder un compte Google. Il offre une communication facilitée avec la possibilité d'interagir, de partager des documents, présentations, ... Et il est accessible via tous types d'écrans : ordinateurs, tablettes et Smartphones. Grâce à cet outil, nous avons pu avancer dans notre projet tout en restant chez nous, en organisant parfois jusqu'à quatre (04) réunions par semaine.



• Mailtrap

Mailtrap.io est un service qui offre un faux serveur SMTP pour tester les envois d'emails pendant la phase de développement de votre projet [6]. Ceci permet d'éviter d'envoyer des emails à de vrais clients lors des tests mais aussi de tester facilement l'envoi d'email sur un environnement local (qui ne dispose pas de système d'envoi).



• Spring boot security

Spring Security est un Framework de sécurité léger qui fournit une authentification et un support d'autorisation afin de sécuriser les applications Spring. Il est livré avec des implémentations d'algorithmes de sécurité populaires [16].



V.6 Les MicroServices

Les microservices désignent à la fois une architecture et une approche de développement logiciel qui consiste à décomposer les applications en éléments les plus simples, indépendants les uns des autres. [23].

Des API REST sont souvent employées pour relier chaque microservice aux autres. Dans le cas d'un besoin critique de mise à jour d'une ressource, seul le microservice contenant cette ressource sera mise à jour.

L'API REST c'est une architecture et est la partie de communication qui joue le rôle des entrées/sorties soit dans l'envoi des données depuis le backend ou entre microservice vers un autre microservice, interface utilisateur ou réceptionner les données depuis un autre service ou UI.

Dans le cas d'une réception de données depuis un autre service ou UI, cette communication s'effectue via un protocole http ou https (le protocole de communication dans le web)et dans ces protocoles http et https existent des verbes ou des méthodes GET, POST,PUT, DELETE, HEAD.

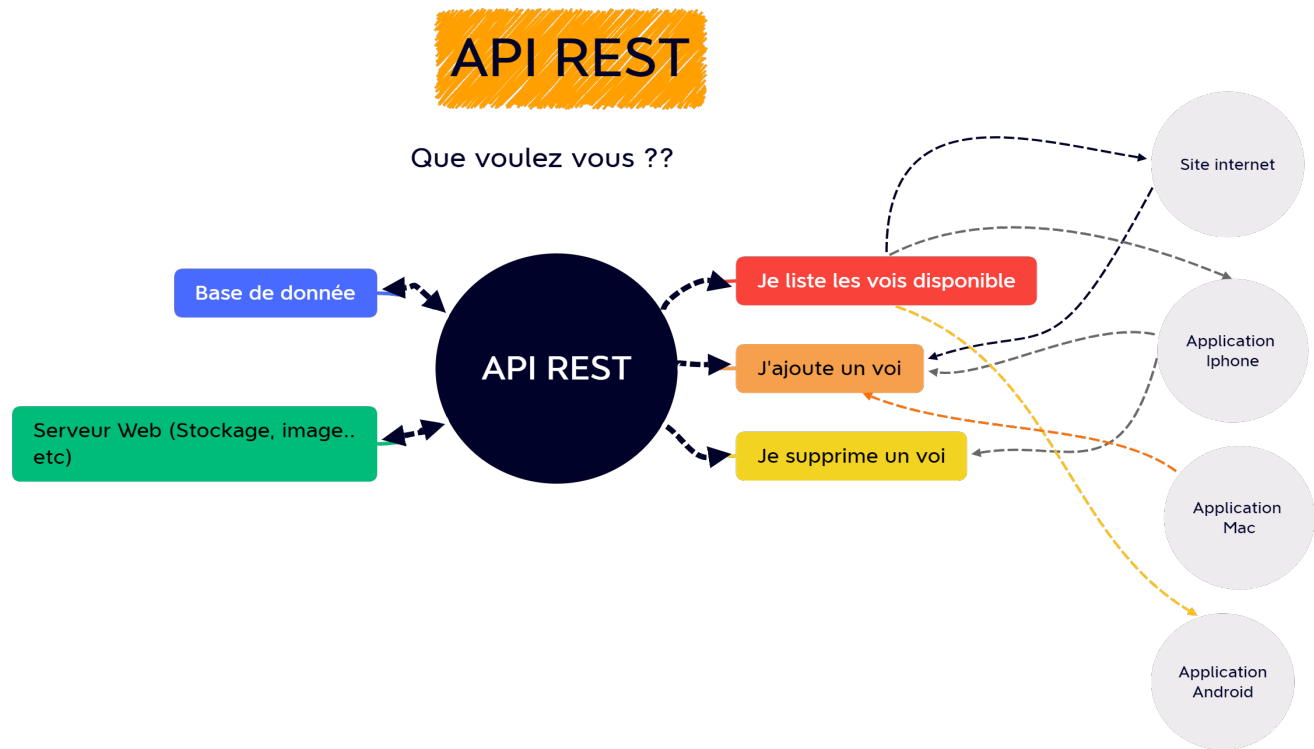


FIGURE V.4 – API REST Representational State Transfer

V.7 Architecture des Micro-Services

La meilleure façon de construire une plate-forme efficace est de la découper en micro-services parfaitement autonomes dans le but de réduire les pannes qui exposent une API REST que les autres Micro-services pourront consommer.

Chaque Micro-service est parfaitement autonome : il a sa propre base de données, son propre serveur d'application (tomcat), ses propres bibliothèques et ainsi de suite. La plupart du temps ces Micro-services sont chacun dans un conteneur Docker, ils sont donc totalement indépendants y compris vis-à-vis de la machine sur laquelle ils tournent.

Voici une vue plus détaillée sur l'architecture Micro-Service.

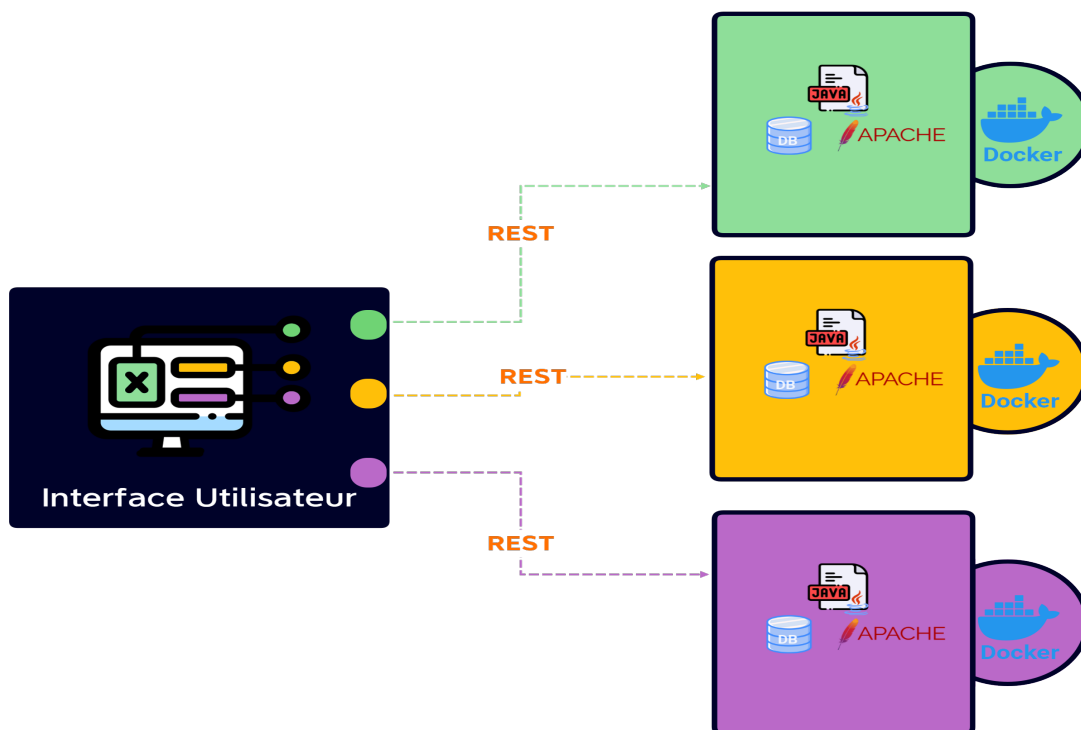


FIGURE V.5 – Architecture Micro-Service

Quand on voudra mettre à jour l'application, il suffira de cibler directement le Micro-service responsable de la fonctionnalité en question.

V.8 Sécurisation de l'accès à l'application grâce a l'authentification et l'autorisation

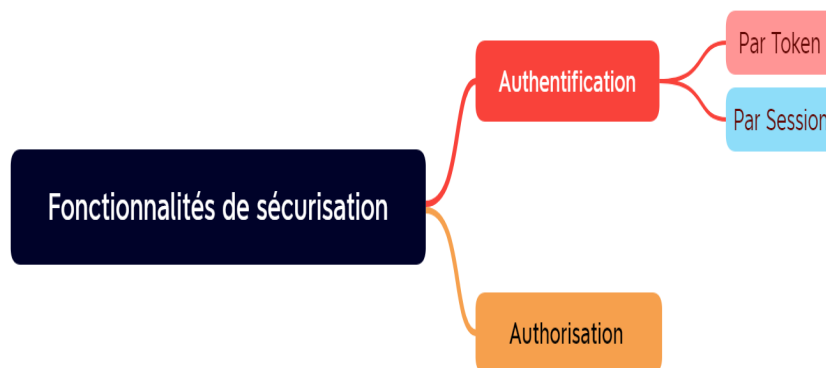


FIGURE V.6 – Les fonctionnalités de sécurisation

- ✓ **L'authentification** permet de s'assurer que l'utilisateur dispose des bonnes informations pour prouver qu'il est bien la personne qu'il prétend être.
- ✓ **L'autorisation** permet de s'assurer que l'utilisateur authentifié se rend uniquement sur les pages qu'il est autorisé à consulter.
- ✓ **L'authentification par session** utilise des cookies pour stocker les informations de l'utilisateur de la session. Ces cookies sont eux-mêmes stockés dans le navigateur de l'utilisateur, et sur le serveur d'autorisation.
- ✓ **L'authentification par token** utilise un JWT transmis par le serveur d'autorisation. Ce JWT est utilisé pour valider la connexion de l'utilisateur, et peut être stocké directement dans le navigateur, mais il est préférable qu'il soit stocké dans un token sécurisé du navigateur.

V.8.1 Jwt Json Web Token

Un Json Web Token [4] est un access token (jeton d'accès) aux normes RFC 7519 qui permet un échange sécurisé de donnée entre deux parties [20]. Il contient toutes les informations importantes sur une entité, ce qui rend la consultation d'une base de données superflue et la session n'a pas besoin d'être stockée sur le serveur (stateless session).

Les JSON Web Token sont particulièrement appréciés pour les opérations d'identification. Les messages courts peuvent être chiffrés et fournissent alors des informations sûres sur l'identité de l'expéditeur et si celui-ci dispose des droits d'accès requis. Les utilisateurs eux-mêmes ne sont qu'indirectement en contact avec les token, par exemple lorsqu'ils entrent un nom d'utilisateur et un mot de passe dans un masque. La véritable communication se fait entre les différentes applications du côté serveur et client.

V.8.1.1 La structure d'un JSON Web Token

Un JWT signé se compose de trois parties codées en base64 et séparées par un point.

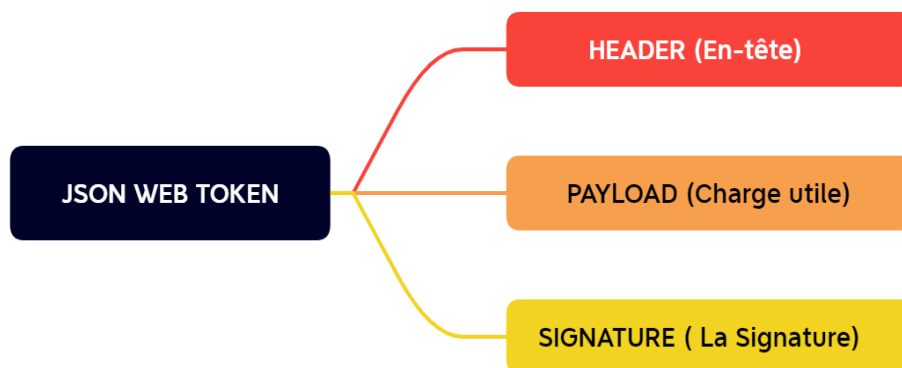


FIGURE V.7 – La structure d'un Json web token

- **L'en-tête, ou header** : est en général composé de deux parties et fournit des informations essentielles sur le token. Il contient le type de token et l'algorithme de signature et/ou de chiffrement

utilisé.

Un exemple de header de JWT :

```
{  
  "alg" : "HS256",  
  "typ" : "JWT"  
}
```

Dans l'exemple ci-dessus, l'en-tête indique que HMAC-SHA256, abrégé « HS256 », est utilisé comme signature du token. Il existe d'autres méthodes courantes de chiffrement, comme RSA avec SHA-256 (« RS256 ») et ECDSA avec SHA-256 (« ES256 »). Il est recommandé de toujours utiliser un chiffrement. Si les données ne requièrent pas un haut niveau de protection, il est possible d'indiquer « none » pour le chiffrement. Les valeurs possibles sont standardisées par la JSON Web Encryption (JWE) selon le RFC 7516.

Le paramètre « cty » pour « Content Type » vient s'ajouter pour les JSON Web Token avec signature ou chiffrement complexe. Il contient également la valeur « JWT ». Dans les autres cas, ce paramètre est laissé de côté.

— **Charge utile (Payload) :** La charge utile du JSON Web Token est la partie qui contient les informations qui doivent être transmises à l'application. C'est là que sont définis certains standards qui déterminent quelles données doivent être transmises. Les informations sont fournies en paire clé/valeur, les clés sont appelées « claims » dans les JWT. Il existe trois types de claims :

1. **Les claims réservées** sont des claims qui sont enregistrées dans le IANA JSON Web Token Claim Register. Leur objet est défini dans une norme. Comme exemple, on peut citer l'émetteur du token (« iss » pour Issuer), le domaine cible (« aud » pour Audience) et la date d'expiration (« exp » pour Expiration Time). On utilise des noms de claims courts afin de limiter au maximum la longueur des token.
2. **Les claims publiques** peuvent être définies librement. Il n'y a aucune limitation. Afin que la sémantique de la clé n'occasionne pas de collision, il est nécessaire d'enregistrer publiquement la claim dans le IANA JSON Web Token Claim Register ou de lui attribuer un nom résistant aux collisions.
3. **Les claims privées** ont conçus pour les informations qui doivent être échangées spécifiquement avec une application donnée. Les claims publiques contiennent des informations comme « Name » ou « E-mail » tandis que les claims privées sont plus spécifiques. Parmi les informations standards pour ce genre de claim, on retrouve par exemple un « identifiant utilisateur » ou un « nom de département » concret. Il est nécessaire de faire attention à éviter les collisions avec les claims réservées ou publiques dans le domaine de nom.

Toutes les claims sont optionnelles. Vous n'avez donc pas à utiliser toutes les claims réservées. En règle général, la charge utile peut contenir autant de claims que nécessaire, il est cependant recommandé de limiter les informations du JWT au strict nécessaire. Plus le JWT est gros, plus il demande de ressources pour être (dé)codé.

La charge utile peut donc être structurée comme suit :

```
{  
  "sub" : "123",  
  "name" : "Alice",  
  "exp" : 30  
}
```

— **La Signature** La signature d'un JSON Web Token est créée grâce au codage base64 de l'en-tête et de la charge utile et la méthode de signature/cryptage spécifiée. La structure est définie par la JSON Web Signature (JWS), une norme standardisée selon le RFC 7515. Pour que la signature fonctionne, il est nécessaire d'utiliser une clé secrète connue uniquement de l'application source. Cette signature vérifie d'une part que le message ne sera pas modifié pendant le transfert. D'autre part, dans le cas d'un jeton signé avec une clé privée, il authentifie également l'expéditeur du JWT. Plusieurs procédés sont disponibles en fonction de la sensibilité des données :

1. **Aucune sécurité** : comme expliqué plus haut, lorsque les données ne nécessitent pas de protection, la valeur « none » peut être indiquée dans l'en-tête. Dans ce cas, aucune signature n'est générée. Le JSON Web Token est alors uniquement composé d'un en-tête et d'une charge utile. Sans sécurité, la charge utile est lisible en texte clair après le déchiffrement de la base64 et il n'est pas vérifié si le message vient du bon expéditeur ou s'il a été modifié lors du transfert.
2. **Signature (JWS)** : en règle générale, il suffit de vérifier si les données viennent du bon expéditeur et si elles ont été modifiées. C'est là qu'intervient le schéma JSON Web Signature (JWS) qui assure que les messages n'ont pas été modifiés pendant le transfert et proviennent du bon expéditeur. Grâce à cette procédure, la charge utile peut également être lue en texte clair après le décryptage base64.
3. **Signature (JWS) et chiffrement (JWE)** : il est possible d'ajouter une JSON Web Encryption (JWE) au JWS. JWE chiffre le contenu de la charge utile qui est ensuite signé par JWS. Un mot de passe commun ou une clé privée est utilisé pour le chiffrement du contenu. L'expéditeur est également vérifié, le message est fiable et authentique et la charge utile n'est plus lisible en texte clair après le décryptage base64. Le cryptage crée une séquence de caractères apparemment aléatoire :

```
{ 7WK5T79u5mIzjIXXi2oI9Fglmgivv7RAJ7izyj9tUyQ }
```

V.8.1.2 Fonctionnement du JWT

Une clé secrète est déterminée avant l'utilisation du JWT. Dès qu'un utilisateur a entré avec succès ses données de connexion, le JWT est renvoyé avec la clé et stocké localement. Le transfert se fait par HTTPS afin de mieux protéger les données.

Lorsque l'utilisateur veut accéder à des ressources protégées comme une API ou un chemin d'accès protégé, le JWT sera envoyé par l'agent utilisateur comme paramètre (par exemple « jwt » pour les GET-Requests) ou comme en-tête d'autorisation (pour POST, PUT, OPTIONS, DELETE). L'interlocuteur peut déchiffrer le JSON Web Token et si le contrôle est réussi, exécuter la demande.

V.8.2 L'Algorithme HS256

Cet algorithme n'est pas utilisé pour chiffrer/déchiffrer des données. Il est plutôt utilisé pour vérifier l'origine ou l'authenticité des données. Lorsque Mary doit envoyer un message ouvert à Jhon et qu'il doit vérifier que le message provient bien de Mary.

Le HS256 peut créer une signature pour un échantillon de données donné en utilisant une clé unique. Lorsque le message est transmis avec la signature, la partie destinataire peut utiliser la même clé pour vérifier que la signature correspond au message. [2].

HS256

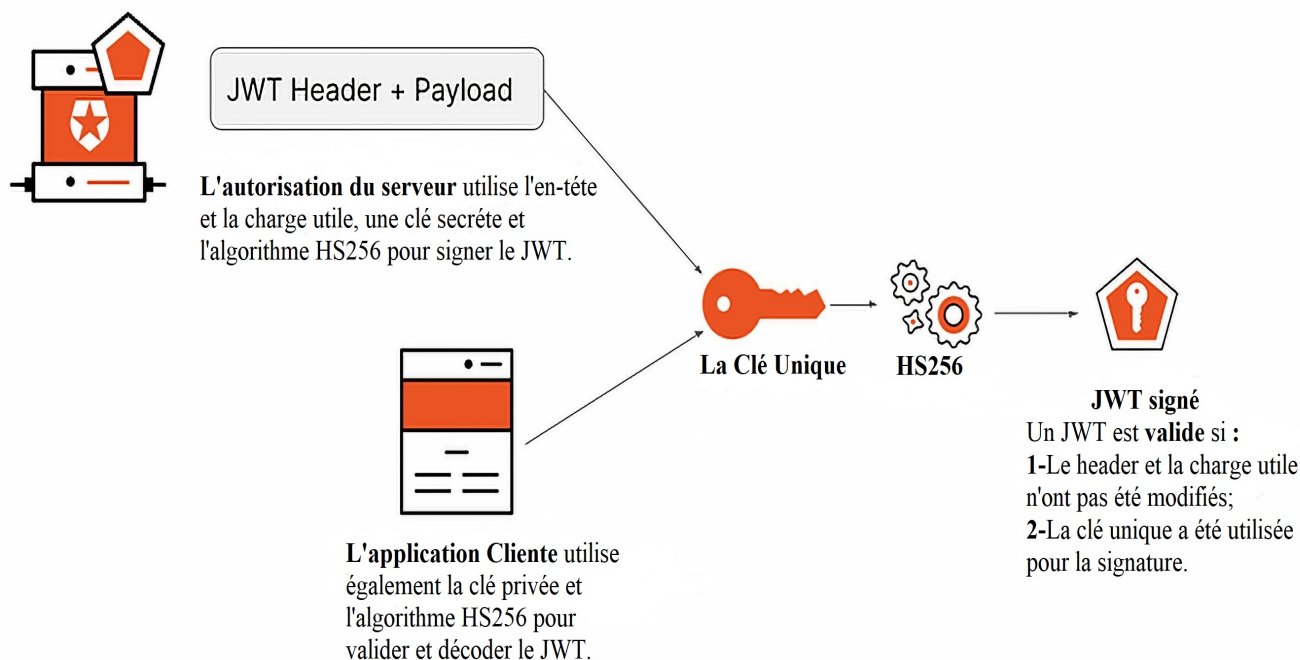


FIGURE V.8 – Fonctionnement du HS256

V.9 Présentation des interfaces

V.9.1 Interface d'accueil DAWINI

Le visiteur de l'application peut rechercher un médecin dans la page d'accueil par nom, spécialité et ville, il peut aussi accéder à son propre compte.

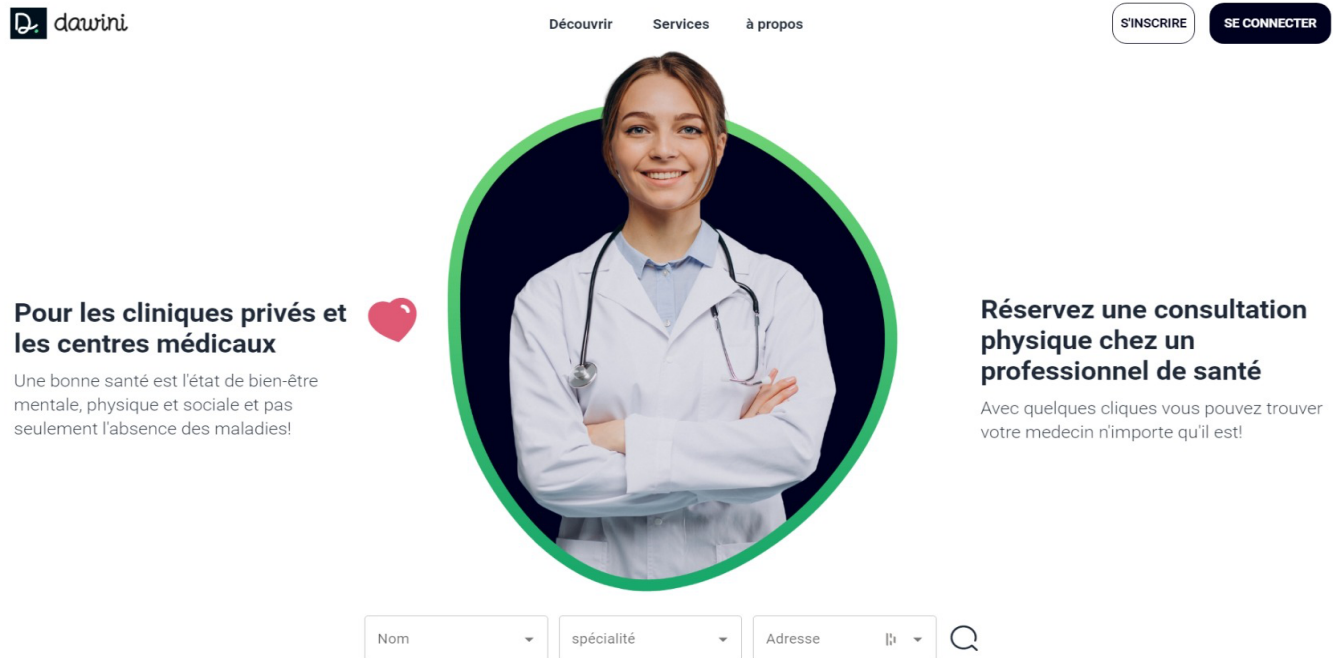
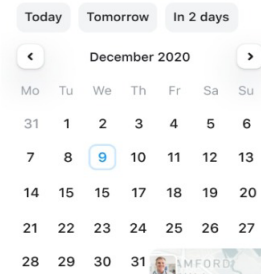
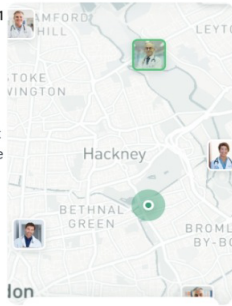


FIGURE V.9 – Page d'accueil- Rechercher praticiens

Vérifiez la disponibilité de votre professionnel de santé, prenez rendez-vous et gagnez du temps.



Trouvez et localisez facilement et rapidement votre professionnel de santé.



Pourquoi rejoindre?

Gérez, consultez et partagez vos documents de santé avec vos professionnels de santé.

Melissa ▾

165 cm 56 kg 3(B)+

Allergy	Intolerance	Diseases
Casein	Lactose	Diabetes (Type 1)
Vitamin C	Gluten	



Vous êtes un professionnel de santé?

Rejoignez notre plateforme pour gagner en confort de travail.

- 👤 Gérez la liste de vos patients et leur suivi de santé.
- 📅 Gérez votre agenda et les heures de disponibilité.
- 📄 Edition automatique des ordonnances avec une base de donnée immenses des médicament utilisé en Algérie.
- 🌟 Améliorez l'accès aux soins pour vos patients en leur proposant la meilleure des expériences.

Rejoindre



FIGURE V.10 – Page d'accueil- Section 3



FIGURE V.11 – Page d'accueil- Footer

V.9.2 Interface Résultat de recherche de médecin

Cette page montre le résultat de la recherche du médecin au visiteur de l'application, ensuite il peut prendre un RDV ou consulter la localisation du médecin.

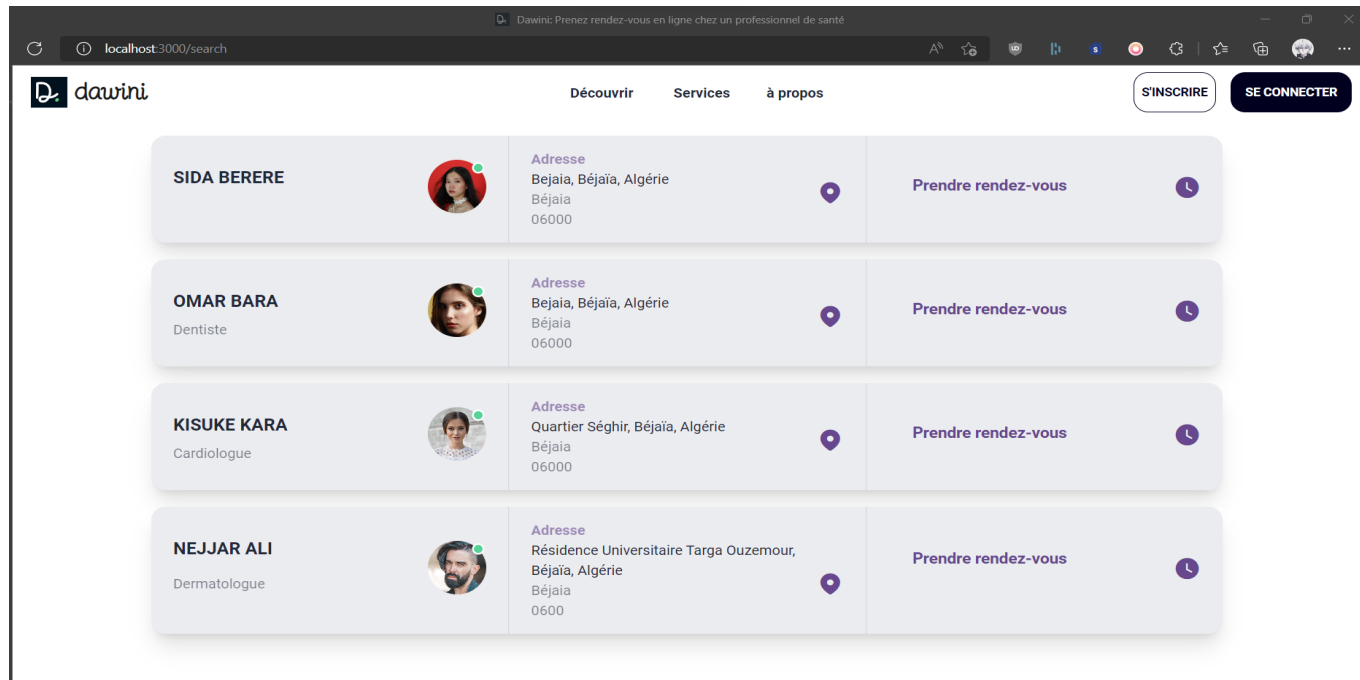


FIGURE V.12 – Page Résultat de recherche de médecin

V.9.3 Interface de localisation du praticien

Cette page montre la localisation du médecin.

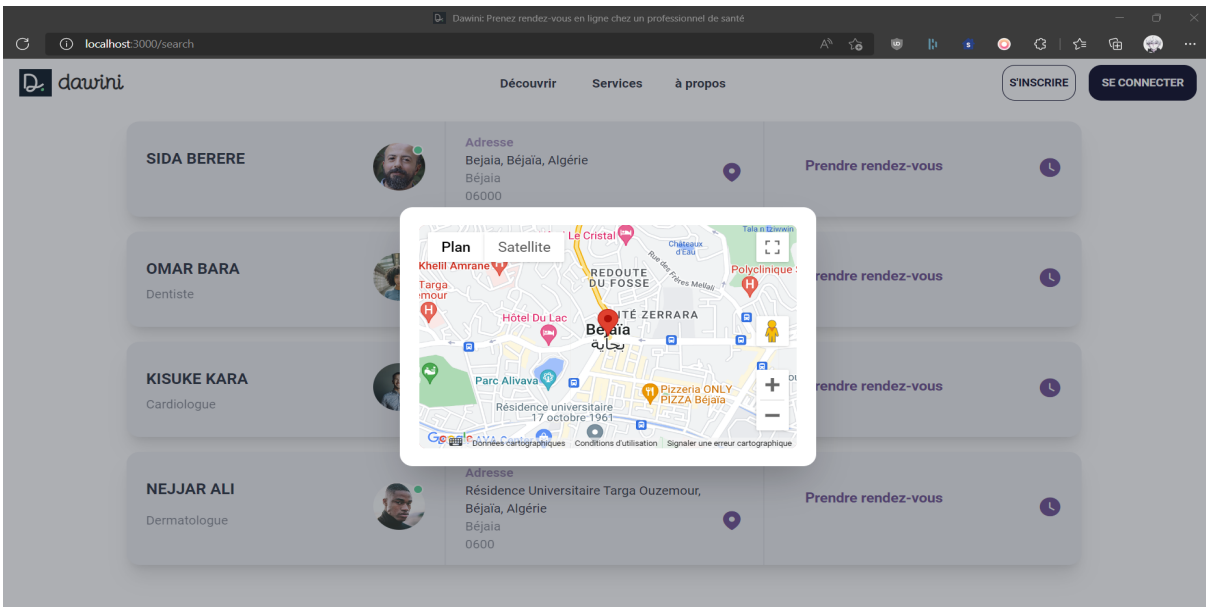


FIGURE V.13 – Page Localisation du médecin

V.9.4 Interface d'inscription du médecin

Dans cette interface le médecin peut s'inscrire.

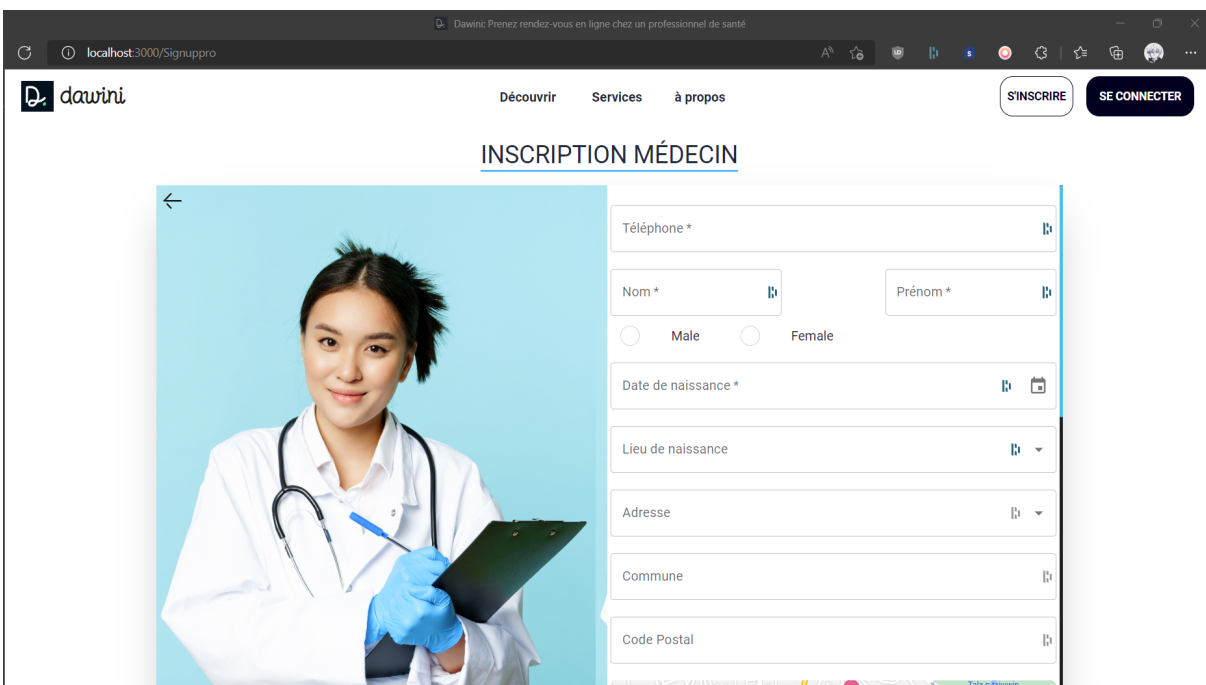


FIGURE V.14 – Inscription médecin (1)

FIGURE V.15 – Inscription médecin (2)

V.9.5 Interface Inscription du patient

Dans cette interface le patient peut s’inscrire.

FIGURE V.16 – Inscription patient (1)

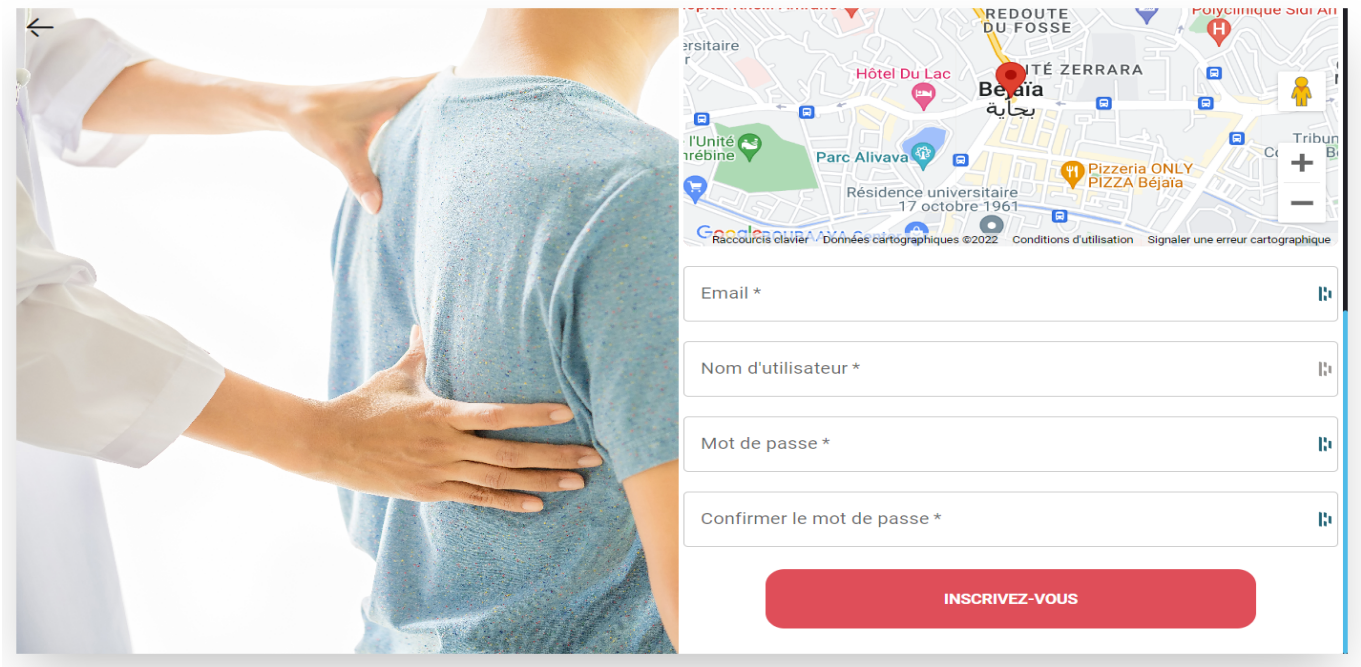


FIGURE V.17 – Inscription patient (2)

V.9.6 Interface d'accueil de l'administrateur

Cette interface représente la page d'accueil de l'administrateur.

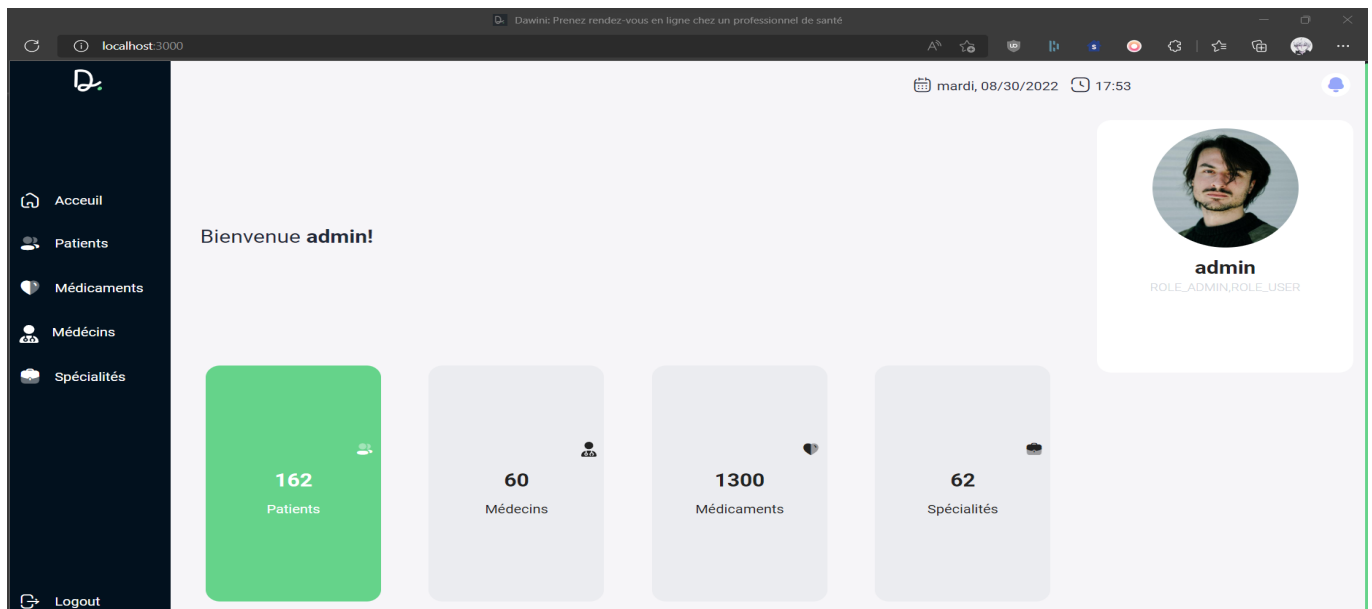


FIGURE V.18 – Page d'accueil- Espace Admin

V.9.7 Interface gérer les médecins

Dans le volet des médecins, l'administrateur peut ajouter, modifier, supprimer ou rechercher un médecin.

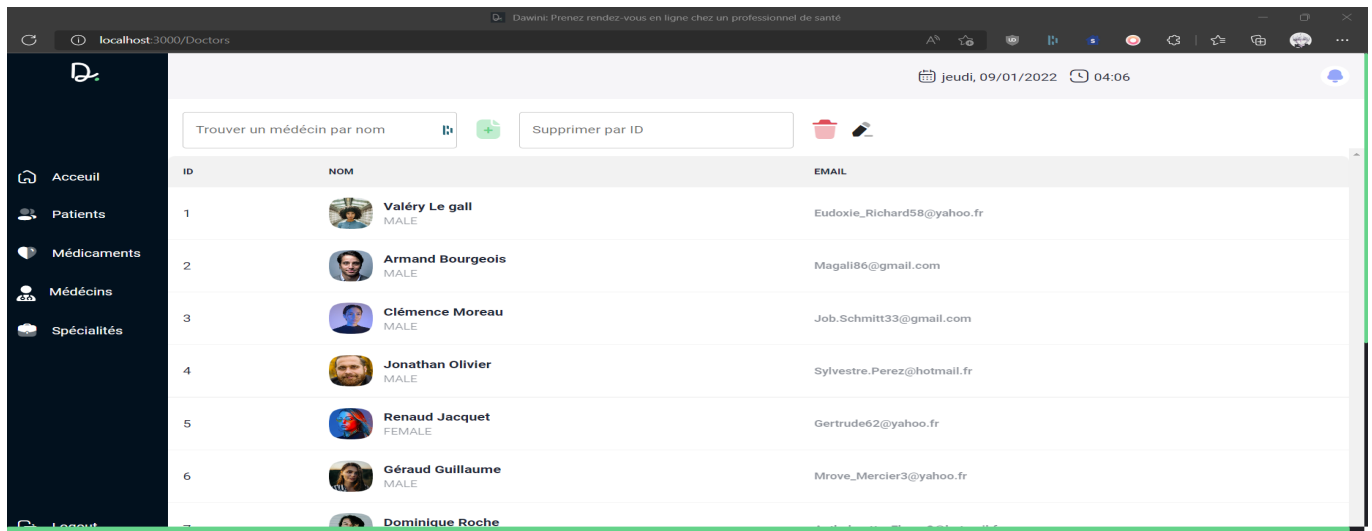


FIGURE V.19 – Gestion des médecins- Espace Admin

V.9.8 Interface ajouter un médecin

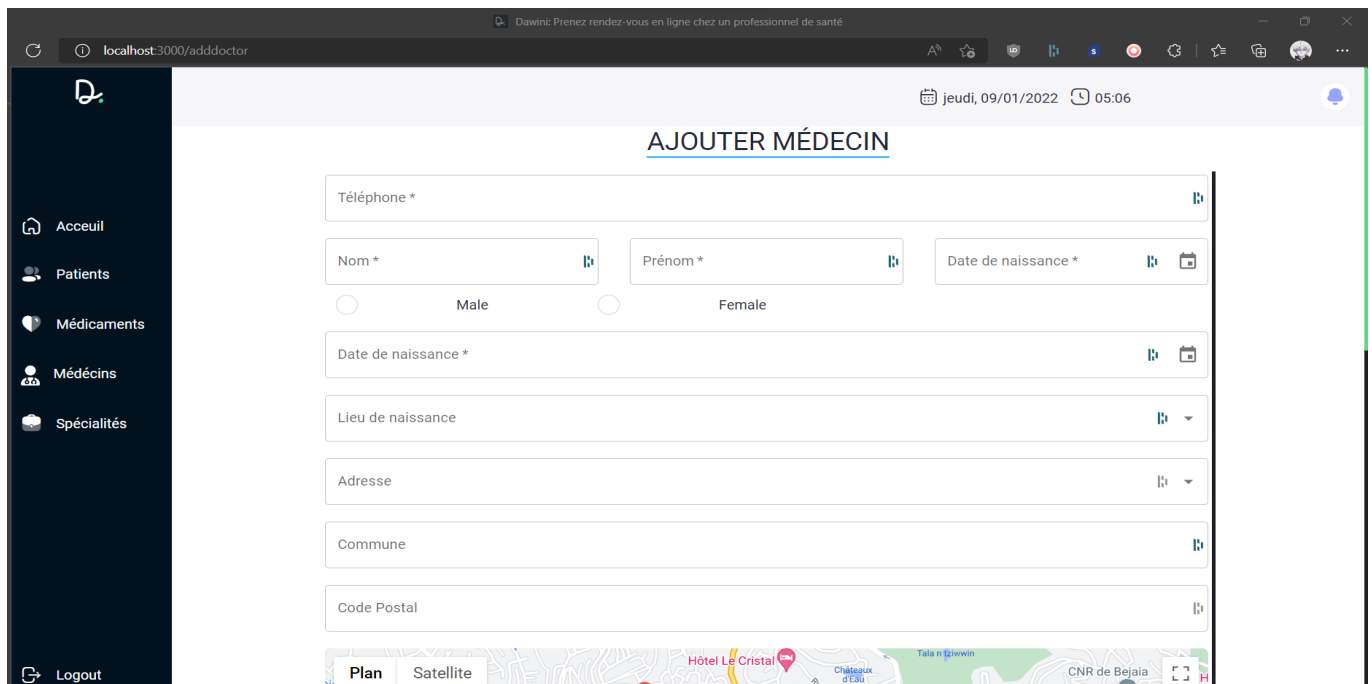


FIGURE V.20 – Ajouter un médecin- Espace Admin

V.9.9 Interface Rechercher médecin

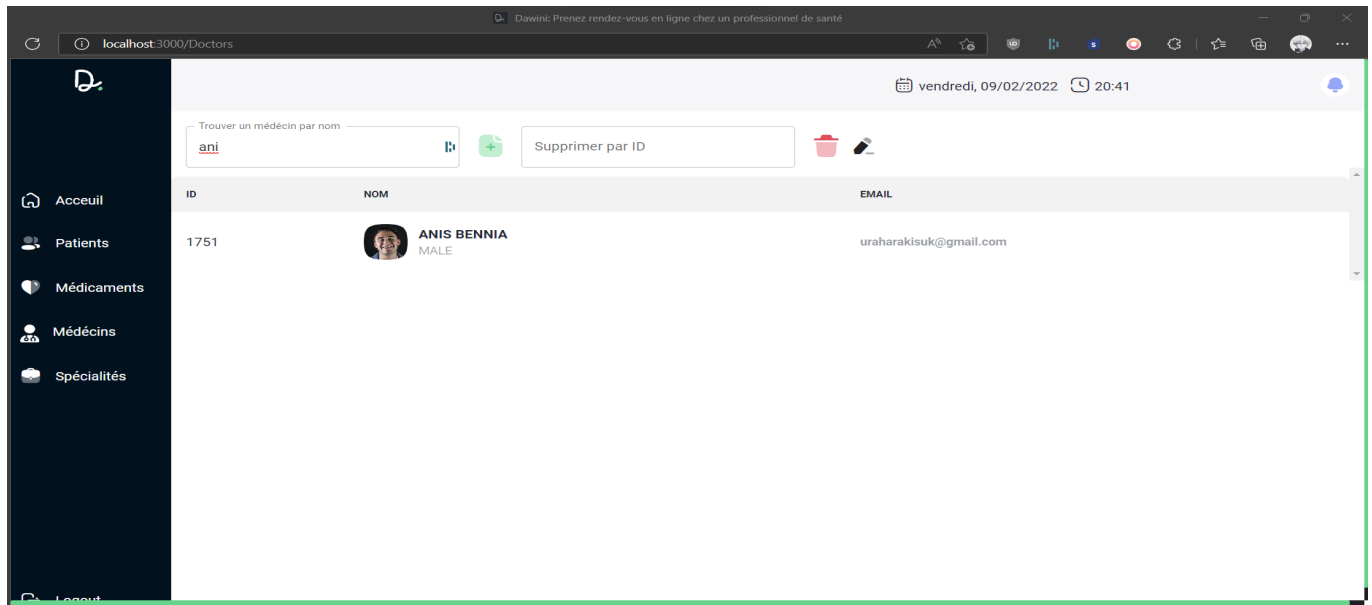


FIGURE V.21 – Rechercher un médecin par nom- Espace Admin

V.9.10 Interface gérer les patients

Dans le volet des patients l'administrateur peut ajouter, modifier, supprimer ou rechercher un patient.

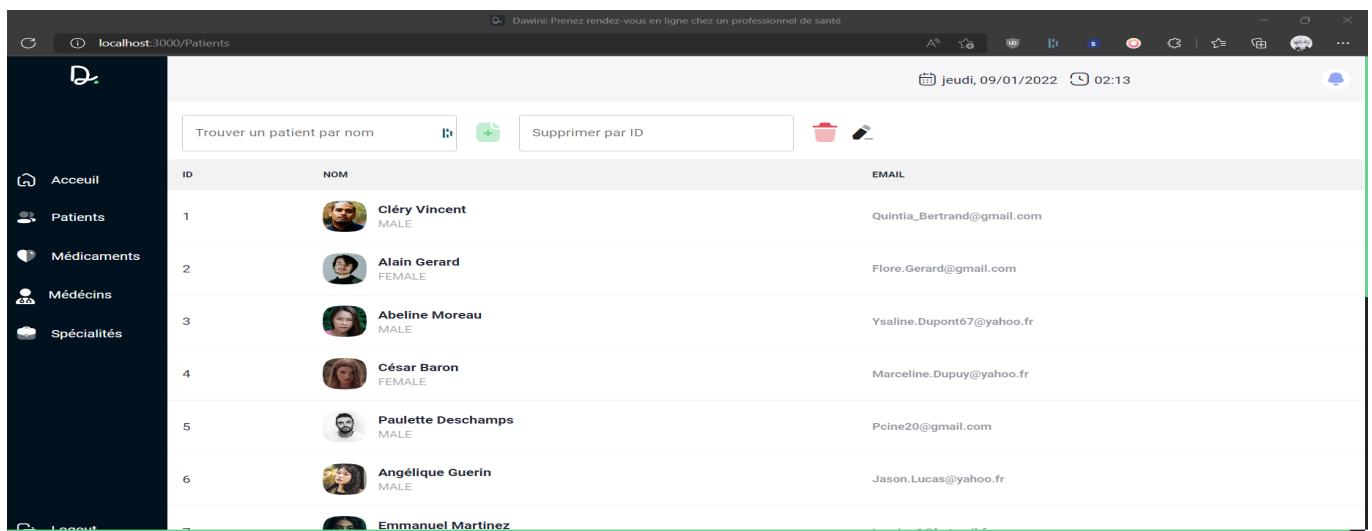


FIGURE V.22 – Gestion des patients- Espace Admin

V.9.11 Interface modifier un patient

localhost:3000/editpatient

jeudi, 09/01/2022 03:49

MODIFIER UN PATIENT

Téléphone *
777898302

Nom *
BENNIA

Prénom *
ANIS

Lieu de naissance
Akbou, Akbou, Algérie

Civilité *
MALE

Date de naissance *
07.07.1996

Email *
uraharakisukee70@gmail.com

MODIFIER

Logout

FIGURE V.23 – Modifier un Patient- Espace Admin

V.9.12 Interface ajouter un patient

localhost:3000/addpatient

jeudi, 09/01/2022 02:42

AJOUTER PATIENT

Téléphone *

Nom * Prénom *

Male Female

Date de naissance *

Lieu de naissance

Adresse

Plan Satellite

Hôpital Kheill Amrano
Maternité Targa Ouzemour
Hôtel Du Lac
Bejaia
CITÉ ZERRARA
REDOUTE DU FOSSE
Château d'eau
Polyclinique Sidi Ahmed
CNR de Bejaia
Hôtel Brahmi HB
CITÉ FRÈRES MEBROUKINE
Tribunalet
CITÉ TOBAL
Université de Béjaia
Stade de l'Unité

Logout

FIGURE V.24 – Ajouter un patient- Espace Admin

V.9.13 Interface gérer les médicaments

Dans cette interface l'administrateur peut ajouter, modifier, supprimer ou rechercher un médicament.

ID	NUMÉRO D'ENREGISTREMENT	DCI	MARK	FORM	DOSAGE	LABORATOIRE D'ENREGISTREMENT
1603	352/15 F 095/17	RIVASTIGMINE TARTRATE EXPRIME EN RIVASTIGMINE	RESTILON	SOLUTION BUV	2MG/ML	EL KENDI INDUSTRIE DES MEDICAM
1604	352/15 F 095/17	RIVASTIGMINE TARTRATE EXPRIME EN RIVASTIGMINE	IBUPROFEN	SOLUTION BUV	2MG/ML	EL KENDI INDUSTRIE DES MEDICAM
1606	243/15 F 095/17	RIVASTIGMINE TARTRATE EXPRIME EN RIVASTIGMINE	DOLIPRANE	SOLUTION BUVE	4MG/M	EL KENDI INDUSTRIE DES MEDICAM
1605	243/15 F 095/17	RIVASTIGMINE TARTRATE EXPRIME EN RIVASTIGMINE	DOLIPRANE	SOLUTION BUV	22MG/ML	EL KENDI INDUSTRIE DES MEDICAM

FIGURE V.25 – Consulter la liste des médicaments- Espace Admin

V.9.14 Interface Ajouter un médicament

AJOUTER UN MÉDICAMENT

Numéro d'enregistrement * Code * dénominations communes internationa...

Mark * Form * Dosage *

Cond * List * P1 *

P2 * Obs * Laboratoire d'enregistrement *

Pays de laboratoire * Date d'enregistrement initial * Date d'enregistrement finale *

Type * StatusM * Duration de Stabilité *

Prix * Remboursable *

AJOUTER

FIGURE V.26 – Ajouter un médicament- Espace Admin

V.9.15 Interface modifier un médicament

localhost:3000/editmed

jeudi, 09/01/2022 01:51

MODIFIER UN MÉDICAMENT

Numéro d'enregistrement * 352/15 F 095/17

Code * 16 E 089

dénominations communes internationales * RIVASTIGMINE TARTRATE EXPRIME EN I

Mark * IBUPROFEN

Form * SOLUTION BUV

Dosage * 2MG/ML

Cond * B/1 FL DE 120ML

List * list I

P1 * HOP

P2 * OFF

Obs * ph

Laboratoire d'enregistrement * EL KENDI INDUSTRIE DES MEDICAMENT

Pays de laboratoire * ALGERIE

Date d'enregistrement initial * 17.11.2021

Date d'enregistrement finale * 26.10.2021

Type * Ge

StatusM * F

Duration de Stabilité * 36MOIS

Prix * PCSU 2907,20 DA

Remboursable * OUI

Logout

MODIFIER

FIGURE V.27 – Modifier un Médicament- Espace Admin

V.9.16 Interface gérer les spécialités

Dans le volet des spécialités l'administrateur peut ajouter, modifier, supprimer ou rechercher une spécialité.

localhost:3000/Specializations

vendredi, 09/02/2022 13:33

Trouver une spécialité par nom

Supprimer par ID

ID	NOM DE SPÉCIALITÉ
2001	Cancerologue
2002	Cardiologue
2003	Chiropracteur
2004	Chirurgien dentiste
2005	Dentiste
2006	Diététicien
2007	Étiopathe
2008	Dermatologue
2009	Chirurgien

Logout

FIGURE V.28 – Liste de spécialité- Espace Admin

V.9.17 Interface ajouter une spécialité

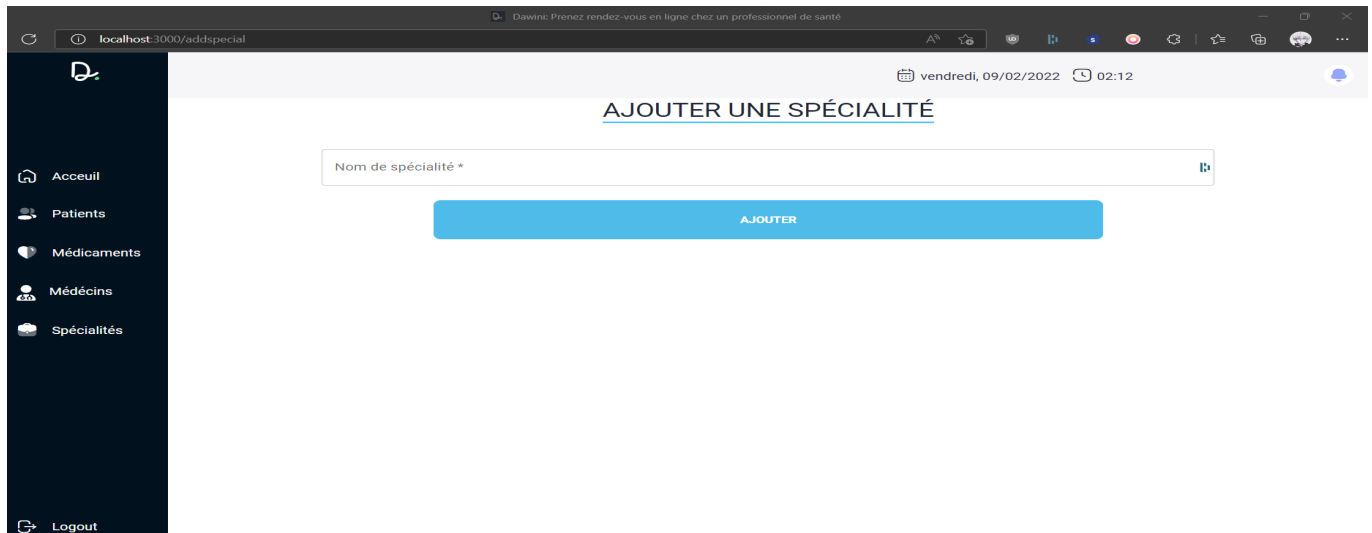


FIGURE V.29 – Ajouté une spécialité- Espace Admin

V.9.18 Interface contactez nous

Dans cette interface le médecin ou le visiteur de l'application peuvent contacter l'administrateur en cas de besoin.

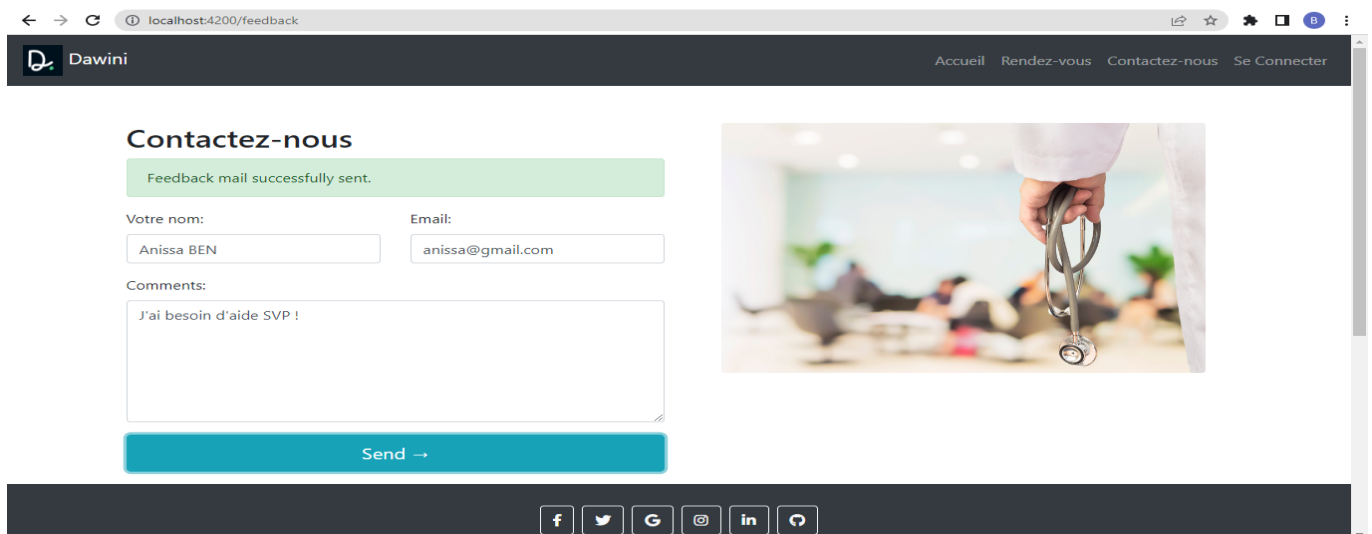


FIGURE V.30 – Page Contactez nous

V.9.19 Interface Agenda du médecin

Le médecin connecté à son propre compte peut consulter son agenda des rendez-vous, confirmer un nouveau rendez-vous, l'annuler ou le supprimer.

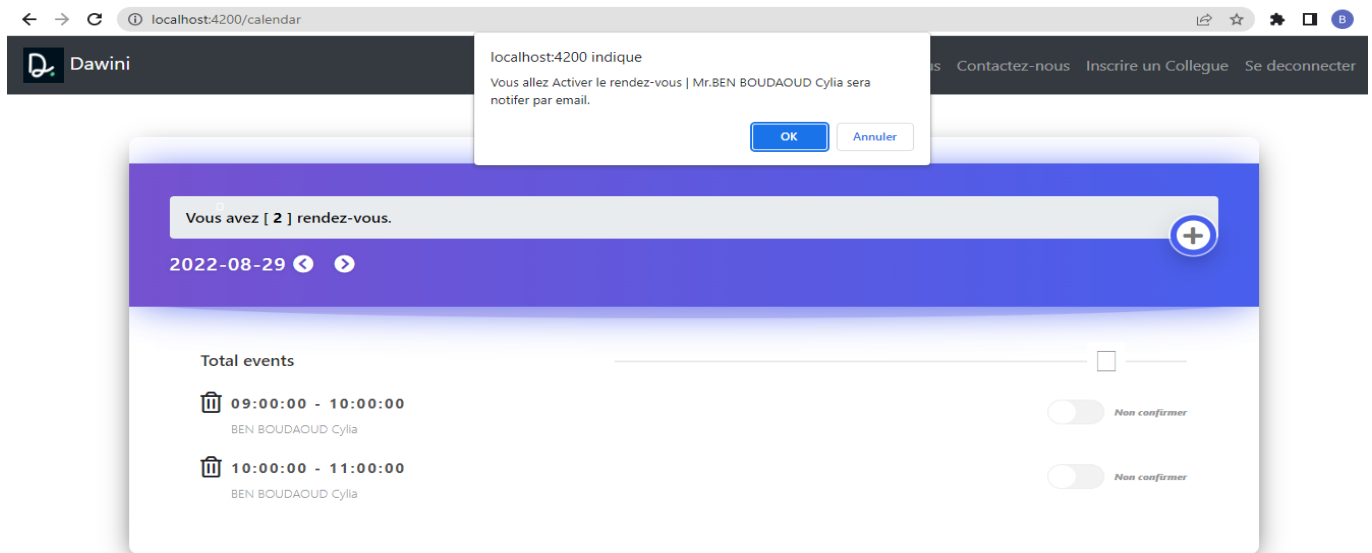


FIGURE V.31 – Page Agenda du Médecin - Annuler Rendez-vous

V.9.20 Interface confirmer un rendez-vous

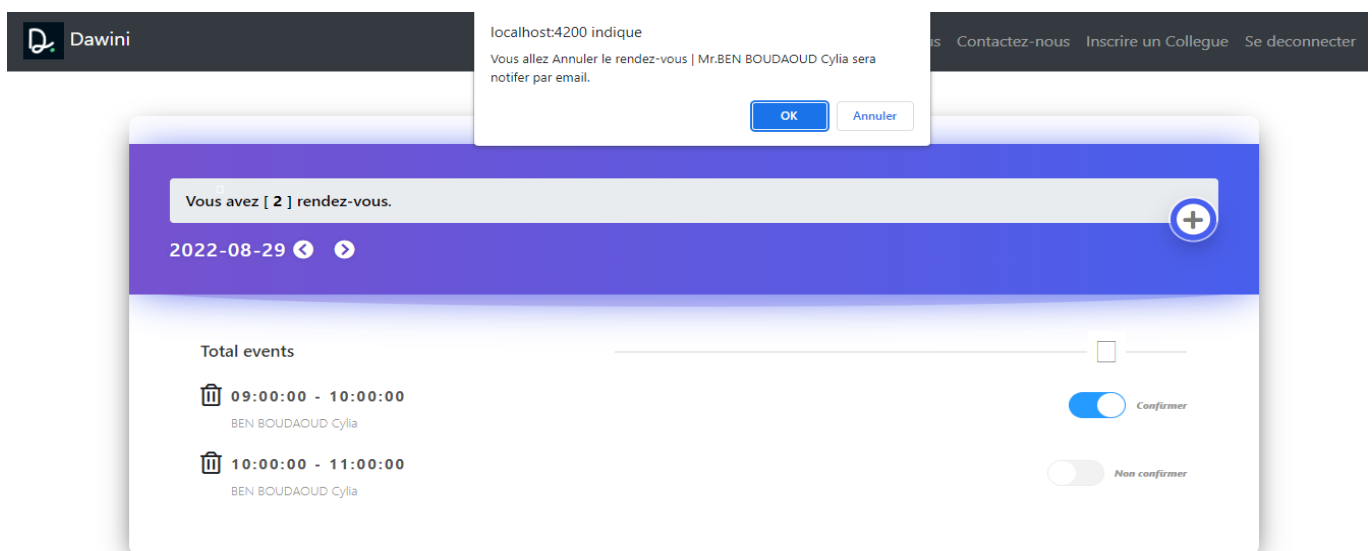


FIGURE V.32 – Page Agenda du Médecin - Confirmer Rendez-vous

V.9.21 Notification par email/rendez-vous confirmé par le médecin

Quand le praticien confirme le nouveau rendez-vous, un email sera envoyé au patient pour l'informer.

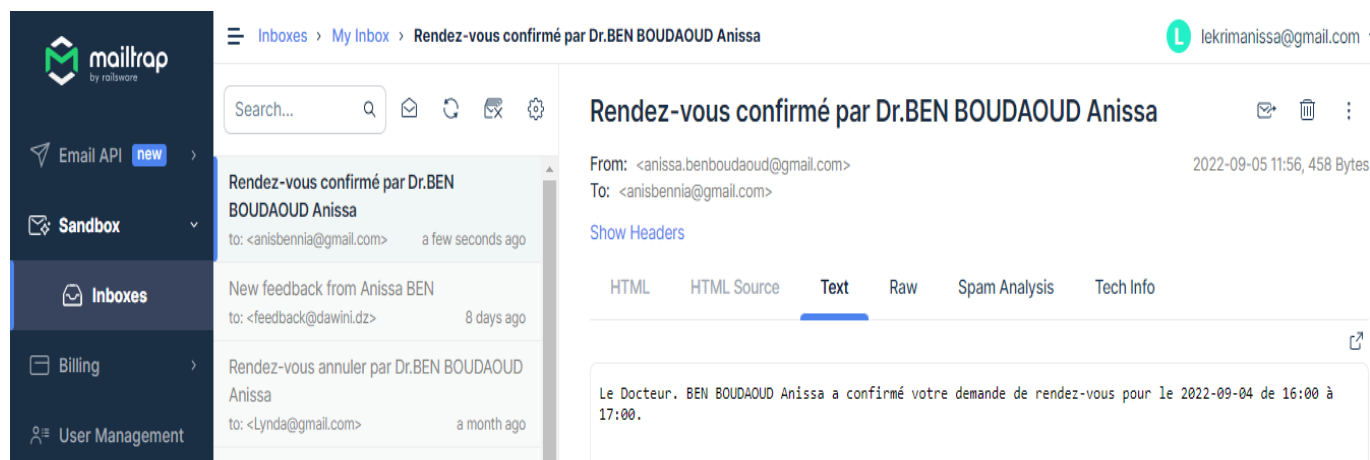


FIGURE V.33 – Notification par email de la confirmation du Rendez-vous

V.9.22 Notification par email/rendez-vous annulé par le médecin

Quand le praticien annule un rendez-vous, un email sera envoyer au patient pour l'informer.

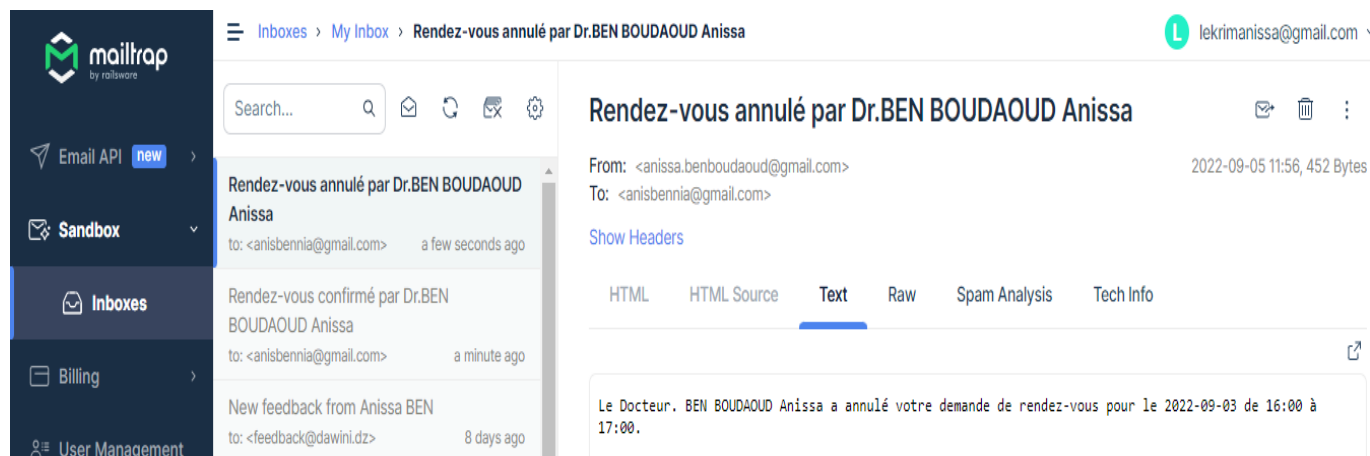


FIGURE V.34 – Notification par email de l'annulation du Rendez-vous

V.9.23 Interface de prise de rendez-vous du médecin

Dans cette page le patient peut prendre un rendez-vous selon la disponibilité des créneaux horaires.

Prenez Rendez-vous

Nom et Prénom du patient Email

Anis BENNNIA anis@gmail.com

Medecin

Dr. BEN BOUDAOU D Anissa

Date souhaitée

< 2022-08-28 >

Prendre rendez-vous →

Liste de disponibilité pour : 2022-08-28

08:00 - 09:00 (Réservé)	<input type="radio"/>
09:00 - 10:00 (Réservé)	<input type="radio"/>
10:00 - 11:00 (Réservé)	<input type="radio"/>
11:00 - 12:00 (Réservé)	<input type="radio"/>
13:00 - 14:00	<input type="radio"/>
14:00 - 15:00	<input type="radio"/>
15:00 - 16:00 (Réservé)	<input type="radio"/>
16:00 - 17:00	<input checked="" type="radio"/>

FIGURE V.35 – Page de Prise de Rendez-vous pour Médecin

V.10 Conclusion

Dans ce chapitre nous avons présenté les aspects pratiques liés à la réalisation de notre application web, à savoir les outils et les langages de développement nécessaires pour le fonctionnement de l'Application, le diagramme de classes, le dictionnaire des données.

En dernier, nous avons illustré quelques interfaces graphiques de notre Application e-santé.

Conclusion générale

Le stage effectué au sein de l'entreprise de développement Tech-Instinct, nous a permis d'acquérir une certaine compétence professionnelle surtout en programmation, on a pu découvrir un vaste domaine de sécurité, et comment combiner des langages de programmations différents pour avoir une application fonctionnelle. En outre, même si le projet n'arrive pas encore à ses fins, nous sommes parvenus à concevoir une plateforme de qualité, grâce à l'emploi de la méthodologie Agile y compris l'utilisation de la méthode SCRUM.

Au cours du développement de notre projet, beaucoup de problèmes qu'on a affronté. Parmi ces problèmes, les grandes lignes sont basées sur la grandeur du projet. En effet un gros projet demande beaucoup de temps nécessaire. En plus, la programmation solitaire nécessite un certain effort surtout au niveau d'analyse et de conception qui demandent beaucoup de réflexion et d'abstraction.

Coté matériel, c'est ce qui est l'un de problèmes qui a retardé le développement coté backend car l'IDE IntelliJ consomme (4Go de RAM) beaucoup de mémoire vive et de processeur.

Les objectifs atteints au cours de ce projet sont :

- ✓ La recherche et la localisation des praticiens.
- ✓ La réalisation de l'espace Administrateur.
- ✓ La prise de Rendez-vous et l'agenda du médecin.
- ✓ L'édition automatique de l'ordonnance.

Cette plateforme est loin d'être terminée car c'est un projet informatique d'entreprise qui peut durer de quatre (4) à (9) mois de développement pour la sensibilité et la grandeur des données du Dossier Médical et nécessite au moins quatre personnes pour le terminer.

Cependant des perspectives restent envisageables, telles que l'amélioration de la qualité des renseignements via une application android et ios qui viendra compléter l'application web ainsi que de consacrer pour chaque spécialiste ses propres pages web qui diffère d'un spécialiste à un autre, rajouter des Rappels automatique par SMS avant 24h du rendez-vous, rajouter le générateur OTP (One-time password) Un mot de passe à usage unique, et un Système reCaptcha pour protéger la plateforme DAWINI contre la fraude et les abus sans créer de friction, et continuer à réaliser toutes les tâches qu'on avaient pas eu le temps de faire.

Annexe

Annexe A.

Diagrammes d'interactions

A.1. Diagramme d'interaction du cas d'utilisation » Inscription Médecin «

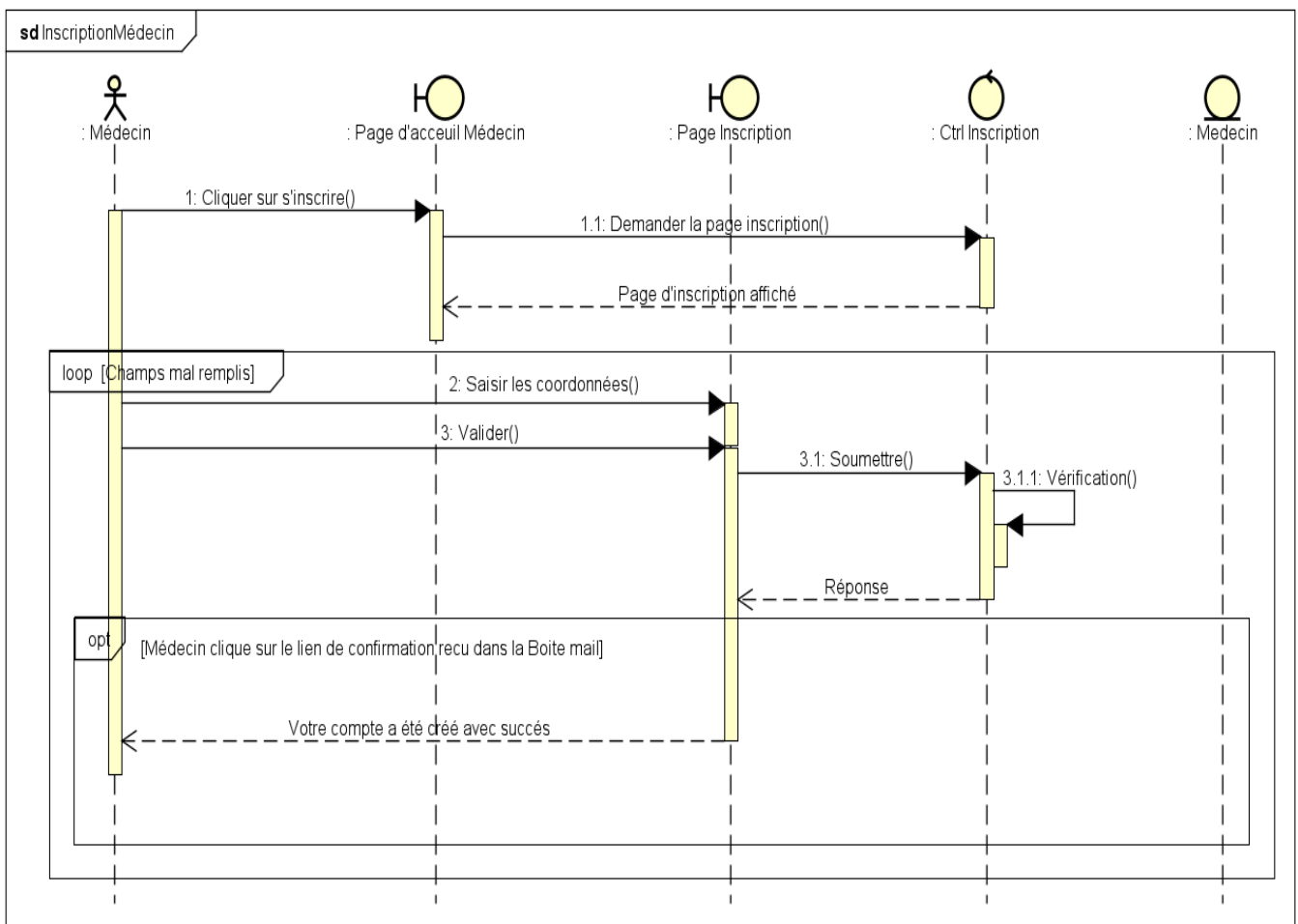


FIGURE V.36 – Diagramme d'interaction de cas d'utilisation « Inscription Médecin »

Annexe B.

Les Descriptions textuelles

B.1.Fiche Descriptive du cas d'utilisation Inscription Médecin

Sommaire d'identification	
Nom du CU :	S'inscrire médecin
Acteur :	Médecin.
Type :	Primaire.
Objectif :	Permet au médecin de créer un compte au système
Description des scénarios	
Pré condition :	/
Scénario nominal :	<ol style="list-style-type: none"> 1. Le système affiche le formulaire de l'inscription. 2. Le médecin remplit le formulaire. 3. Le médecin valide le formulaire. 4. le système vérifie les champs du formulaire. 5. le système envoie un lien de confirmation a la boîte mail du médecin 6. le médecin clique sur le lien de confirmation envoyé a sa boîte mail. 8. Le système active le compte.
Scénario Alternatif	<ol style="list-style-type: none"> 2.a. si formulaire mal rempli, afficher des messages d'erreurs. 6.a. si le médecin ne clique pas sur le lien de confirmation, son compte reste inactive.
Post condition :	Le médecin S'inscrit avec succès.

TABLE V.2: Description textuelle du cas d'utilisation « Inscription Médecin »

Bibliographie

- [1] DZDOC , a propos de dzdoc [en ligne]. [consulté le 07/01/2022]. Consultable à l'adresse : <https://dzdoc.com/apropos.php>.
- [2] RS256 vs HS256 , quelle est la différence? [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://www.web-dev-qa-db-fra.com/fr/jwt/rs256-vs-hs256-quelle-est-la-difference/826436771/>.
- [3] Qu'est-ce qu'IntelliJ IDEA ? [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://www.jetbrains.com/fr-fr/idea/features/>.
- [4] IONOS , jwt [en ligne]. [consulté le 128022]. Consultable à l'adresse : <https://www.ionos.fr/digitalguide/sites-internet/developpement-web/json-web-token-jwt/>.
- [5] Tout savoir sur JHipster! [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://www.processindustries.fr/tout-savoir-sur-jhipster/>.
- [6] Mailtrap.io [en ligne]. [consulté le 128022]. Consultable à l'adresse : <https://grafikart.fr/tutoriels/mailtrap-771>.
- [7] Comprendre AngularJS et se former [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://info-digitale.com/comprendre-angularjs-se-former/>.
- [8] Asana , pourquoi asana? [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://asana.com/fr/guide/resources/info-sheets/why-asana>.
- [9] astah [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://progsoft.net/fr/software/astah-community-edition>.
- [10] WIKIPEDIA , doctolib [en ligne]. [consulté le 07/01/2022]. Consultable à l'adresse : <https://fr.wikipedia.org/wiki/Doctolib>.
- [11] Gitkraken,gitkraken, what is it? and common actions? [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://dev.to/iphytech/gitkraken-what-is-it-and-common-actions-5531>.
- [12] BDM/tools, google meet [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://www.blogdumoderateur.com/tools/google-hangouts-meet/>.

-
- [13] JPA (Java Persistence API) [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://www.jmdoudoux.fr/java/dej/chap-jpa.htm>.
- [14] PostgreSQL [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://www.oracle.com/fr/database/definition-postgresql.html>,.
- [15] Postman c'est quoi? [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://www.eewee.fr/postman-cest-quoi/>,.
- [16] Utilisez Spring Security dans votre application Spring Boot [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://openclassrooms.com/fr/courses/7137776-securisez-votre-application-web-avec-spring-security/7275496-utilisez-spring-security-dans-votre-application-spring-boot>.
- [17] WebStorm le plus intelligent des ide javascript [en ligne]. [consulté le 12/08/2022]. Consultable à l'adresse : <https://www.jetbrains.com/fr-fr/webstorm/#:~:text=WebStorm%20est%20un%20environnement%20de,les%20t%C3%A2ches%20complexes%20avec%20facilit%C3%A9>.
- [18] Grady Booch, Jim Rumbaugh, and I Jakobson. Uml : Unified modeling language. *Versão*, 1997.
- [19] Spring Boot. Spring boot, 2020.
- [20] Michael Jones, John Bradley, and Nat Sakimura. Json web token (jwt). Technical report, 2015.
- [21] Ken Schwaber and Jeff Sutherland. Scrum. *Siehe : http://www.scrum.org/Resources/What-is-Scrum*, 2010.
- [22] Ken Schwaber and Jeff Sutherland. The scrum guide. *Scrum Alliance*, 21(1), 2011.
- [23] Johannes Thönes. Microservices. *IEEE software*, 32(1) :116–116, 2015.

RÉSUMÉ

Ce travail a été réalisé dans le cadre d'un projet de fin de cycle, ayant comme objectif l'obtention du diplôme de Master 2 en Génie Logiciel. Il s'abrege en la conception et réalisation d'une plate-forme e-santé. Ce mémoire se focalise sur l'étude, la conception et la réalisation permettant de faciliter les taches aux utilisateurs ainsi que de gagner du temps en éliminant tous déplacements non nécessaires. Pour schématiser notre solution, nous avons utilisé le processus de développement SCRUM et le langage de modélisation « Unified Modeling Language » (UML). Nous avons choisi de programmer la plate-forme avec le langage React js, Spring Boot et Angular js et Postgresql comme Système de Gestion de Base de Données Orienté objet (SGBDOO).

Mots clés : E-santé, SCRUM, UML, Spring boot, React JS, Angular JS.

ABSTRACT

This work was carried out as part of an end-of-cycle project, with the objective of obtaining the diploma of Master 2 in Software Engineering. It abbreviates in the design and the realization of an e-health platform. This memory focuses on the study, the design and the realization allowing to facilitate the tasks to the users as well as to save time by eliminating all the not necessary displacements. To outline our solution, we used the SCRUM development process and the Unified Modeling Language (UML) modeling language. We chose to program the platform with the React js language, Spring Boot and Angular js and Postgresql as the object-oriented Database Management System (DBMS).

Keywords : E-health, SCRUM, UML, Spring boot, React JS, Angular JS.