

## Ministère de l'Enseignement Supérieur, de la Recherche Scientifique

Logo Entreprise

**Direction Générale des Études Technologiques** 

Institut Supérieur des Études Technologiques de Djerba Département Technologies de l'informatique

Code	
projet	•••••

## Rapport de

## STAGE DE FIN D'ETUDES

Présenté en vue de l'obtention du diplôme de

## Licence Appliquée en Technologies de L'informatique

Parcours: DSI

Conception et développement d'une application web et Mobile pour la gestion des cabinets médicaux

#### Elaboré par :

Sarah BEN DAOUED Malek KHANNOUSSI

#### **Encadré par:**

M. Aymen BOUREGBA

#### Effectué à :

**Entreprise: VINTONET** 

**Encadreur: M. Mohamed Thabet JMAEL** 

Année universitaire : 2020/2021

## Remerciements

Tout d'abord, nous tenons remercions notre encadreur de stage M. Mohamed Thabet JAMEL qui a accepté notre présence comme stagiaires dans son entreprise et qui nous a accompagnés avec beaucoup de patience et de pédagogie tout au long du processus.

Mes remerciements les plus sincères à notre encadreur M. Aymen BOUREGBA pour son encadrement rigoureux, ses remarques, ses idées directives, ses méthodes pédagogie, son soutien tout au long de notre projet.

Nous tenons également à exprimer notre sincère gratitude à tous les enseignants qui ont partagé leurs connaissances avec nous et tous les membres du département informatique de l'ISET.

## **Dédicaces**

Du profond de mon cœur, Je dédie ce modeste travail à tous ceux qui me sont chers :

#### À Mes chers parents

Je vous dédie ce travail pour démontrer ma gratitude et mon profond amour, votre gentillesse et votre tendresse ont toujours rempli mon cœur.

Vous avez utilisé votre santé pour me faire fleurir, et je suis heureuse de vous voir cueillir vos fruits aujourd'hui.

#### À mon cher frère et mes chères sœurs

Avec mes sentiments éternels, je vous souhaite le bonheur au plus profond de votre cœur, et que la vie nous ne sépare jamais.

#### À tous mes ami(e)s

Merci pour votre gentillesse et votre patience pour les bons moments qu'on a partagé ensemble.

Je vous dédie ce travail tout en espérant que notre amitié dépassera les limites du temps et de l'espace.

### À M. Aymen BOUREGBA

Merci pour votre bonne volonté d'accepter de nous encadrer, pour tout le temps qu'il nous a octroyé et pour tous les conseils qu'il nous a prodigué.

#### À mes enseignants

Qui m'ont apporté leurs connaissances, leur soutien et leur gentillesse tout au long de l'année.

Mon expérience d'apprentissage à l'ISET m'amené des souvenirs agréables, riches en connaissances.



Sarah BEN DAOUED

## **Dédicaces**

Au fond de mon cœur, je dédie ce travail à :

#### À Ma Chère Mère

Aucune dédicace ne peut exprimer mon respect, mon amour éternel et mon respect sacrifie.

Vous avez consenti à mon éducation et à mon bien-être.

J'espère que ce travail peut atteindre les souhaits que vous exprimés et le fruit de vos innombrables sacrifies.

Que Dieu, accorde la santé et le bonheur.

#### À l'âme de mon père

Ce travail est dédié à mon père, décédé trop tôt, qui m'a toujours poussé et motivé dans mes études.

J'espère que, du monde qui est sein maintenant, il apprécie cet humble geste comme preuve de reconnaissance de la part d'un fils qui a toujours prié pour le salut de son âme.

Puisse Dieu, le tout puissant, l'avoir en sa sainte miséricorde!

#### À Mes Frères et Ma Sœur

Vous étiez toujours présents pour m'aider et m'encourager. Sachiez que vous serez toujours dans mon cœur.

## À M. Aymen BOUREGBA

Merci d'avoir pris le temps de nous aider au cours de cette année et de nous avoir accompagné dans la maitrise de nos connaissances.

#### À mes amis

Pour ceux qui ont été à mes côtés et qui m'ont accompagné dans l'enseignement supérieur, chers amis, merci pour tout le bon temps ensemble.



Malek KHANNOUSSI

## Sommaire

Introd	uction Générale	1
Chapi	tre 1 : Etude préalable	2
I.	Introduction	2
II.	Cadre du projet	2
III.	Présentation de projet	6
IV.	Méthodologies adoptées	7
V.	Environnement techniques	9
VI.	Conclusion	14
Chapi	tre 2 : Mise en œuvre de projet	15
I.	Introduction	15
II.	Identification des acteurs	15
III.	Analyse des besoins	15
IV.	Planification des releases	20
V.	Diagramme de cas d'utilisation	20
VII.	Diagramme de classe	22
VI.	Conclusion	23
Releas	se 1 : Gestion de compte	24
I.	Introduction	24
II.	Organisation des sprints	24
III.	Product Backlog Gestion de compte (release 1)	24
IV.	Conception de gestion de compte (release1)	25
V.	Sprint 1: Inscription	26
VI.	Sprint 2: Authentification	29
VII.	. Sprint 3 : Gestion de compte	31
VIII	I. Conclusion	34
Releas	se 2 : Relation entre patient / médecin	35
I.	Introduction	35
II.	Organisation des sprints	35
III.	Backlog Relation patient/ médecin (release2)	35
IV.	Conception de Relation patient/ médecin (release2)	36
V.	Sprint 1 : Ajout médecin/patient	37
VI.	Sprint 2 : Consultation liste médecin/patient	41
VII.	. Sprint 3 : Chercher liste médecin/patient	44
VIII	I. Conclusion	47

Release 3 : Gestion des rendez-vous	48
I. Introduction	48
II. Organisation des sprints	48
III. Backlog relation patient/ médecin (release3)	48
IV. Conception de Gestion des rendez-vous (release3)	49
V. Sprint 1 : Réservation rendez-vous médecin pour un patient	50
VI. Sprint 2: Réservation rendez-vous patient chez un médecin	54
VII. Sprint 3 : Consultation rendez-vous	57
VIII. Conclusion	59
Release 4 : Gestion des consultations	60
I. Introduction	60
II. Organisations des sprints	60
III. Backlog Gestion consultation (release4)	60
IV. Conception Gestion de consultation (release4)	61
V. Sprint 1 : Ajout consultation	63
VI. Sprint 2 : Consultation historique	65
VII. Conclusion	66
Release 5 : Partie administrative & Partie mobile	67
I. Introduction	67
II. Organisation des sprints	67
III. Product Backlog	67
IV. Conception	68
V. Sprint 1 : Partie administrative	68
VI. Sprint 2 : Partie mobile	70
VII. Conclusion	74
Conclusion Générale et perspectives	75
Nétographie	76
Annexes	77

## Liste des figures

Figure 1: iset jb	
Figure 2: Emplacement organisme d'accueil	3
Figure 3: Cyber-Parc	3
Figure 4: Equipe	4
Figure 5: Réalisation d'un projet	5
Figure 6 : Gestion de projet	8
Figure 7 : Scrum	9
Figure 8 : Architecture 3-tiers	10
Figure 11: API REST	
Figure 9 : Laravel	
Figure 10: React	12
Figure 12: Drawio	
Figure 13: Visual Code	12
Figure 14: Xampp	12
Figure 15: Discord	
Figure 16: GitHub	13
Figure 17: Word	
Figure 18: Planification des releases	20
Figure 19 : Diagramme de cas d'utilisation Er	reur! Signet non défini.
Figure 20 : Diagramme de classe	
Figure 21 : Sprints de gestion de compte (release1)	24
Figure 22 : Diagramme de cas d'utilisation Gestion de compte	
Figure 23 : Diagramme de classe Gestion de compte	26
Figure 24 : Diagramme de séquence d'inscription	27
Figure 25: Interface d'inscription	28
Figure 26: Erreur email	28
Figure 27: Erreur CIN	28
Figure 28 : Erreur téléphone	28
Figure 29 : Diagramme de séquence d'authentification	30
Figure 30: Interface d'authentification	30
Figure 31 : Diagramme de séquence gestion de compte	32
Figure 32 : Interface Gestion de compte d'un médecin	
Figure 33: Interface Gestion de compte d'un patient	34
Figure 34 : sprints de relation patient / médecin	35
Figure 35 : Diagramme de cas d'utilisation relation patient/ médec in .	36
Figure 36 : Diagramme de classe relation patient/ médecin	
Figure 37 : Diagramme de séquence d'ajout médecin	38
Figure 38 : Interface d'ajout médecin	38
Figure 39 : Diagramme de séquence d'ajout patient	
Figure 40: Interface d'ajout patient	
Figure 41 : Interface consultation liste médecin	
Figure 42 : Diagramme de séquence consultation liste patient	43
Figure 43: Interface consultation liste médecin	44
Figure 44 : Diagramme de séquence chercher médecin	
Figure 45 : Interface chercher médecin	
Figure 46: Interface chercher patient	
Figure 47: Interface chercher patient	47

Figure 48 : Sprints de gestion des rendez-vous	48
Figure 49 : Diagramme de cas d'utilisation Gestion des rendez-vous	49
Figure 50 : Diagramme de classe Gestion des rendez-vous	50
Figure 51 : Diagramme de séquence réservation d'un rendez-vous pour un patient	51
Figure 52 : Interface Dashboard médecin	52
Figure 53 : Interface sélectionner patient	52
Figure 54: Ajout patient	53
Figure 55 : Créer compte patient	53
Figure 56 : Diagramme de séquence réservation rendez-vous chez un médecin	55
Figure 57: Interface réservation d'un rendez-vous chez un médecin	55
Figure 58 : Interface salle d'attente	56
Figure 59: Interface confirmation rendez-vous	56
Figure 60 : Diagramme de séquence consultation rendez-vous patient	57
Figure 61: Interface consultation rendez-vous patient	58
Figure 62 : Diagramme de séquence consultation rendez-vous médecin	59
Figure 63: Interface consultation rendez-vous médecin	59
Figure 64 : Sprints gestion de consultation	60
Figure 65 : Diagramme de cas d'utilisation gestion de consultation	61
Figure 66 : Diagramme de classe gestion de consultation	62
Figure 67 : Diagramme de séquence ajout consultation	64
Figure 68 : Diagramme de séquence consultation historique patient	
Figure 69 : Diagramme de séquence consultation historique	66

## **Tableaux**

Tableau 1 : Inconvènients et solutions proposé	7
Tableau 2: Environnement matériel	14
Tableau 3: Product Backlog	19
Tableau 4 : Product Backlog Gestion de compte	25
Tableau 5 : Cas d'utilisation d'inscription	
Tableau 6 : Cas d'utilisation d'authentification	29
Tableau 7 : Cas d'utilisation gestion de compte	31
Tableau 8: Product Backlog relation patient/ médecin	36
Tableau 9: Cas d'utilisation d'ajout médecin	
Tableau 10: Cas d'utilisation d'ajout patient	
Tableau 11 : Cas d'utilisation consultation liste médecin	
Tableau 12 : Diagramme de séquence consultation liste médecin	42
Tableau 13: Cas d'utilisation consultation liste patient	43
Tableau 14 : Cas d'utilisation chercher médecin	44
Tableau 15: Cas d'utilisation chercher patient	46
Tableau 16 : Product Backlog de gestion des rendez-vous	49
Tableau 17 : Cas d'utilisation réservation des rendez-vous pour un patient	
Tableau 18: Cas d'utilisation réservation rendez-vous chez un médecin	54
Tableau 19: Cas d'utilisation consultation rendez-vous patient	57
Tableau 20 : Cas d'utilisation consultation rendez-vous médecin	58
Tableau 21: Product backlog gestion consultation	60
Tableau 22: Cas d'utilisation ajout consultation	63
Tableau 23: Cas d'utilisation consultation historique patient	65
Tableau 24: Cas d'utilisation consultation historique	66

## **Introduction Générale**

De nos jours, la digitalisation est indispensable dans la vie professionnelle et la vie quotidienne. La digitalisation est devenue une solution de publicité et de communication très rapide et efficace. C'est pour ces raisons que la majorité des différentes organisations ont recours surtout à des sites web pour mieux diriger leurs affaires.

Dans cette optique s'introduit notre projet de fin d'étude effectué au sein de la société **VINTONET** qui consiste à développer une plateforme web et une application mobile.

En effet, notre application permet de gérer les patients, les rendez-vous, les dossiers médicaux... Afin de faciliter la réservation des rendez-vous et l'échange des données entre les médecins et les patients.

Notre travail se traduit dans ce rapport qui développe les différentes phases par lesquelles nous somme passé qui sont organisées en quatre chapitres comme suit :

- Dans le premier chapitre, nous présentons d'une manière générale l'université, l'organisme d'accueil, le cadre général de travail, l'étude de l'existant, le projet réalisé en expliquant la méthodologie adoptée, l'architecture, l'étude technologique, et en fin l'environnement logiciel et matériel.
- Dans le deuxième chapitre, nous présentons les acteurs, l'analyse des besoins, la planification des releases, le diagramme de cas d'utilisation et le diagramme de classe.
- Pour les quatre derniers chapitres, nous présentons chacun d'eux de la manière suivante : l'organisation des sprints, le backlog des sprints, la conception, et le travail de chaque sprint.

## Chapitre 1 : Etude préalable

#### I. Introduction

Dans ce chapitre, nous présentons l'université, l'organisme d'accueil, ainsi que le projet en le plaçant dans son cadre général. Ensuite, nous recommandons d'introduire l'environnement de projet comme suit : Présenter l'étude de l'existant, les méthodologies adoptées, l'architecture ainsi que l'étude technologique. Enfin nous présentons l'environnement logiciel et matériel pour mener à bien ce projet.

### II. Cadre du projet

#### II.1 Présentation de l'université

L'institut supérieur des études technologiques de Djerba (ISET Djerba), a été créé le 11 mai 2000, fait partie d'un réseau de 26 instituts supérieurs des études technologiques. Dans le cadre du LMD, il dispense une formation supérieure technologique dans les spécialités suivantes : Technologies de l'Informatique Génie électrique, Génie mécanique, et Sciences économique et gestion.

Au terme de trois années d'études, l'étudiant obtient un diplôme national d'une licence appliquée.

Depuis 2003, l'ISET de Djerba a proposé des programmes de master professionnels dans le secteur de l'hôtellerie et du tourisme.

L'iset jb situé à Midoun, Djerba Midoun 4116. [1]





Figure 1 : iset jb

### II.2 Présentation de l'organisme d'accueil

Cyber Park Djerba

Vintonet a vu jour en 2014, c'est une institution spécialisée dans l'édition et l'intégration de solutions des applications web et mobile.

Vintonet située à Cyber Parc\* Djerba Houmt Souk Blvd De L'environnent BP N°474 Houmt Souk 4180.

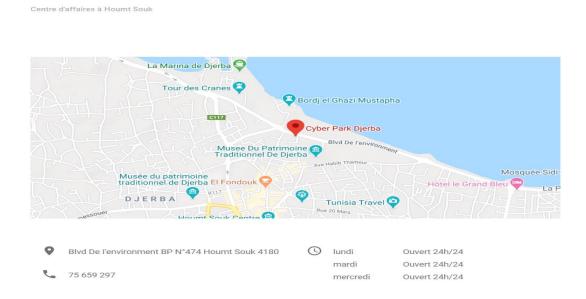


Figure 2: Emplacement organisme d'accueil

\*Le Cyber-Parc est un espace aménagé, équipé de réseaux modernes d'information et de communication.



Figure 3: Cyber-Parc

#### II.2.1 Organigramme

Dans cette société il y a plusieurs organismes :

- Directeur générale
- Département commercial et contact client.
- Département développement

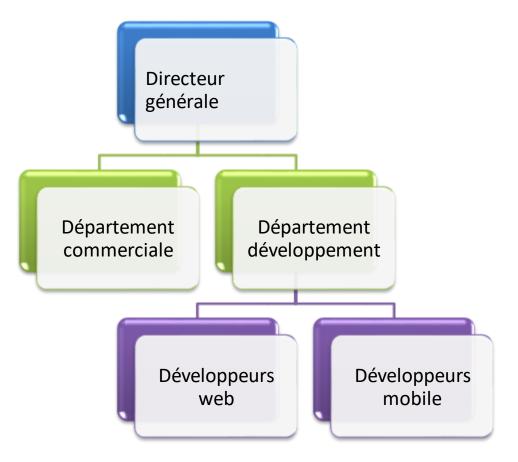


Figure 4: Equipe

#### II.2.2 Réalisation d'un projet

La réalisation d'un projet se fait comme suit :

- Département Commerciale : Communication avec le client sur le projet
- Réunir toutes les équipes
- Analyser les besoins du client
- Conception de projet
- Département commerciale : validation avec le client
- Département Développement : réalisation de projet

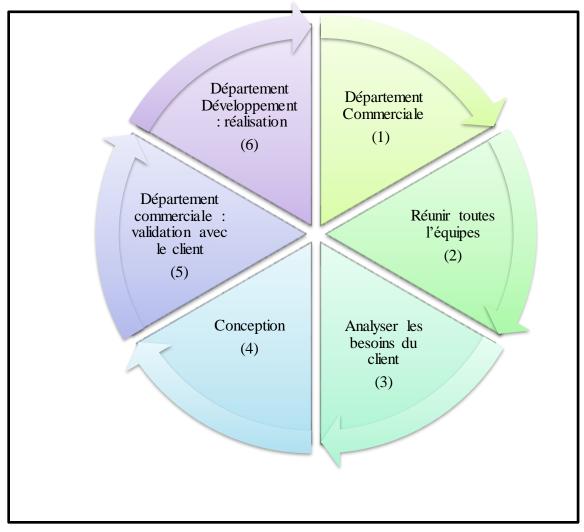


Figure 5: Réalisation d'un projet

## II.3 Cadre générale de travail

Ce stage est réalisé dans le cadre d'un projet de fin d'études dans le but d'obtenir un diplôme « licence appliquée en technologie d'informatique spécialité développement système d'information» à l'institut supérieur des études technologiques de Djerba.

Ce stage a été effectué au sein de la société **VINTONET** dans le but d'atteindre un objectif final qui consiste à concevoir et développer une application web et mobile pour la **gestion des cabinets médicaux**.

## III. Présentation de projet

#### III.1 Etude de l'existant

Chaque jour, il y a des personnes qui consultent des médecins cependant il est impossible d'échanger instantanément des données entre eux. Cependant, les médecins suivent les informations de leur patient sur la fiche médicale. En effet, les fiches médicales ne sont pas suffisantes pour consulter toutes les informations de patient. Par exemple, le médecin ne peut pas consulter les antécédents médicaux du patient et les médicaments qu'il utilise chez d'autres médecins. D'où proviennent les risques liés à la prise des médicaments qui ne peuvent pas être utilisés ensemble. De plus, la fiche peut être facilement perdue et la perte d'une fiche implique la perte du patient. Enfin, l'organisation des rendez-vous et l'organisation de fiches médicales sont les soucis les plus difficiles à gérer dans les cabinets médicaux.

### III.2 Solution proposé

#### III.2.1 Contexte

Développement d'une plateforme web et d'une application mobile pour faciliter la prise de rendez-vous et l'échange des données entre les médecins et les patients.

#### III.2.2 Objectif

Regrouper les médecins et leurs patients afin de leur faciliter l'échange des données de façon, l'interaction peut devenir instantanée afin :

- ❖ D'offrir au médecin, la suivie les antécédents de son patient.
- ❖ D'offrir au patient, la possibilité de demander une réservation d'un rendezvous en ligne chez un médecin.
- ❖ De faciliter la gestion du cabinet médical.
- ❖ De permet aux utilisateurs de suivre les dossiers médicaux à tout moment on donnant un privilège.

#### III.2.3 Les apports de projets

Les apports de projets se divisent en 2 parties :

- ❖ Pour les médecins : la suivie des dossiers médicaux de ses patients.
- ❖ Pour les secrétaires : l'organisation des dossiers médicaux.
- ❖ Pour les patients : la prise des rendez-vous et la suivie de son dossier médical.

#### III.2.4 Cible de projet

La cible est la population que l'on souhaite toucher lors d'une action.

Les cibles de ce projet sont principalement : les médecins, les secrétaires et les patients.

## III.3 Comparaison entre l'étude de l'existant et notre solutions

Le tableau ci-dessous présente la comparaison entre l'étude de l'existant et notre projet.

Les inconvénients Nos solutions proposées			
La fiche médicale ne contient pas	L'application contient toutes les		
toutes les informations nécessaires du	informations du patient.		
le patient.			
Le médecin ne peut pas suivre le	Le médecin suivre le dossier médical de son		
dossier de patient chez les autres	patient sur l'application.		
médecins (les antécédents, les			
médicaments utilisés)			
Les informations de patient se trouvent	Les informations de patient se trouvent dans		
dans une fiche qui peut être perdu.	la plateforme		
Le patient ne peut pas proposer un	Le patient peut proposer un rendez-vous		
rendez-vous.	chez un médecin.		
Le patient ne peut pas consulter l'état	Le patient peut suivre instantanément l'état		
de la salle d'attente.	de la salle d'attente.		

Tableau 1 : Inconvènients et solutions proposé

## IV. Méthodologies adoptées

## IV.1 Gestion de projet

La gestion de projet est l'ensemble des activités visant à organiser le bon déroulement d'un projet et à en atteindre les objectifs, incluant des contraintes de couts de délais et de ressources.

Par conséquent, nous devrons avoir plusieurs compétences dont la maitrise des techniques de la gestion de projet, la gestion d'équipe, et essentiellement bien comprendre les besoins du projet. L'objectif de la gestion de projet est de pouvoir le terminer dans les délais. [2]

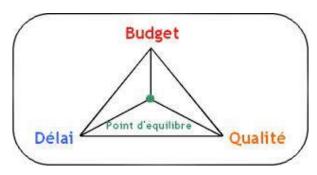


Figure 6: Gestion de projet

#### IV.2 Scrum

Pour mener à bien ce travail, nous avons choisi **Scrum**. C'est une méthodologie agile pour les projets informatiques, dont les ressources seront mises à jour régulièrement. Le principe est basé sur le fait d'être prêt à rediriger le projet au fur et à mesure de son avancement. Il s'agit d'une approche dynamique et participative du projet. [3]

#### > Les avantages

#### ✓ Comprendre le travail et les tâches à effectuer

Appliquer **Scrum**, c'est subdiviser votre projet en plusieurs petites parties réalisables. Cette fragmentation vous oblige à vous demander si toutes les tâches doivent vraiment être effectuées pour mener à bien votre projet, et vous permet d'examiner d'un œil critique leur exécution.

#### **✓** Transparence et respect

**Scrum** exige de la transparence. Les membres de l'équipe doivent savoir ce que les autres accomplissent et le résultat qu'ils peuvent en attendre. Mais chacun peut déterminer comment il accomplit sa tâche.

#### ✓ Visibilité continue

Travailler de manière efficace et maligne n'est possible que si vous conservez une vue d'ensemble et restez organisé. Pour tout tenir à jour, il faut communiquer ouvertement. C'est vraiment le cœur du processus de travail : pour assurer une bonne réalisation.

#### ✓ Focus et flexibilité

**Scrum** a été conçu non seulement pour améliorer les projets mais aussi pour en accélérer la réalisation. [4]

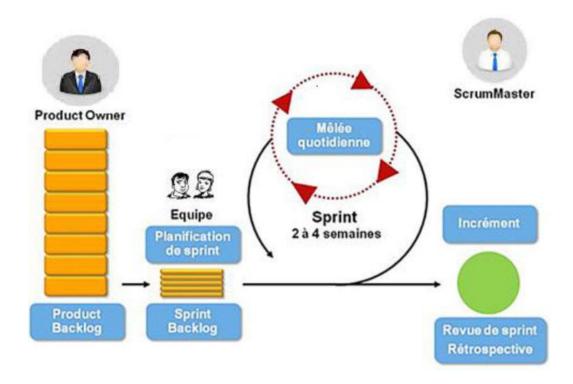


Figure 7 : Scrum

#### IV.3 UML

UML, On le traduit par « Langage de modélisation unifié ». La notation UML est un langage visuel constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, etc. [5]

## V. Environnement techniques

## V.1 Architecture de l'application (3-tiers)

On a proposé dans ce projet de suivre l'architecture 3-tiers.

L'architecture 3-tiers est composée de trois éléments, ou plus précisément dans ce cadrelà de trois couches. En effet dans ce contexte, et dans la philosophie qui a guidé l'élaboration de cette architecture, il est plus adéquat de parler de couche fonctionnelle où à chacune d'elle est attachée un élément/entité logique. [6]

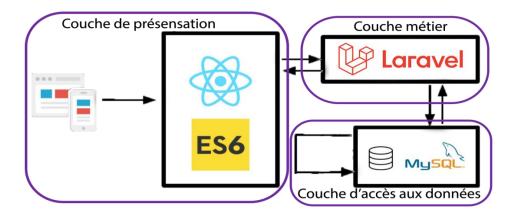


Figure 8: Architecture 3-tiers

#### V.2 API REST

La partie front n'est pas directement connectée à la base de données, mais via l'API REST qui joue le rôle d'interfaçage. En effet, une API compatible REST, ou « RESTful », est une interface de programmation d'application qui fait appel à des requêtes HTTP pour obtenir (GET), placer (PUT), publier (POST) et supprimer (DELETE) des données. [7]

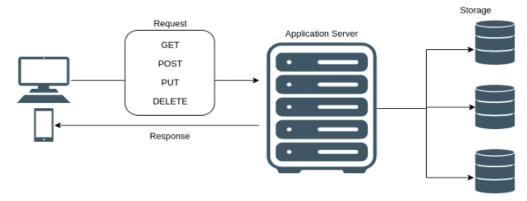


Figure 9 : API REST

### V.3 Technologies adoptées

#### V.3.1 Laravel

L'outil qu'on a utilisé pour développer l'application back end est le Framework laravel. Laravel est un Framework open source écrit en PHP.

Laravel est aujourd'hui le produit PHP le mieux noté sur le site de Git Hub. [8] En effet, laravel présente plusieurs avantages c'est pour cette raison on a le choisis pour développer notre application comme avantages on peut citer :

#### > Les avantages

- ✓ Laravel est l'un des meilleurs Framework **PHP** car il possède des bibliothèques orientées objet et d'autres prés installés, qui ne se trouvent dans aucun autre **Framework** PHP. L'une des bibliothèques pré installées est la bibliothèque d'authentification. Ces bibliothèques regorgent d'excellentes fonctionnalités faciles à utiliser et à implémenter pour chaque développeur.
- ✓ Mise à jour de la base de données : La mise à jour de base de données est un processus automatisé.
- ✓ Une documentation de haute qualité en cas de besoin : Chaque développeur web utilisant Laravel peut accéder à une documentation particulièrement facile à comprendre, soignée et beaucoup plus organisée.



Figure 10 : Laravel

#### V.3.2 React

L'outil qu'on a utilisé pour développer l'application front end est la bibliothèque **React**. En effet, **React** est une bibliothèque **JavaScript** libre développée par Facebook depuis 2003. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage.

**React** est une bibliothèque qui ne gère que l'interface de l'application, considéré comme la vue dans le modèle MVC. [9]

#### > Les avantages

#### ✓ Très rapide

**ReactJS** crée son propre **DOM** virtuel. Cette approche vous donne énormément de flexibilité et des performances exceptionnelles, car **ReactJS** calcule quel changement dans le **DOM** a besoin d'être fait, et change juste la partie qui a besoin d'être mise à jour.

#### ✓ L'intelligibilité

**ReactJS** produit du code « propre » (simple à lire), sa lecture permet de déterminer immédiatement quelles sont les fonctionnalités de votre application. Ce qui est essentiel pour la maintenance et l'expansion de votre projet dans le temps.

#### ✓ Le contenu est référençable

C'est la fonctionnalité qui fait la différence par rapport aux autres Framework. Grâce à l'utilisation d'un serveur **Node**, le code va pouvoir être généré côté client ET côté serveur.



Figure 11: React

#### V.4 Environnement logiciels et matériels

#### V.4.1 Environnement logiciels

#### Drawio

**Drawio** est une application de création de diagrammes et schéma. [10]



Figure 12 : Drawio

#### Visual studio code

**Visual studio code** est un éditeur de code extensible développé par Microsoft.

- **Première version :** 14 novembre 2015.
- **Dernière version :** 1.55.2 en 13 avril 2021.
- **Version utilisée :** 1.55.2.



Figure 13: Visual Code

#### > Xampp

C'est un ensemble de logiciels permettant de mettre en place facilement un serveur web.

- **❖ Première version :** 5.6.14 en 15 octobre 2015.
- **Dernière version :** 7.4.13 en 11 janvier 2021.
- **❖** Version utilisée : 7.3.22



Figure 14: Xampp

#### Discord

**Discord** est un logiciel gratuit de messagerie utilisé pour discuter et obtenir l'aide ou des conseils.

**❖** Première version : 13 mai 2015.

**❖ Dernière version :** 76.611en 11 février 2021.

**❖** Version utilisée : 78.12



Figure 15 : Discord

#### **➢** GitHub

**GitHub** est une plateforme open source de gestion de versions Et de collaboration destinée aux développeurs de logiciels.

**❖ Première version :** 7 avril 2005.

**Dernière version :** 2.31.1en 16 juin 2021.

**❖** Version utilisée : 2.32.0



Figure 16 : GitHub

#### > Word

Word est un outil de traitement de texte pour créer le rapport de notre projet, le traiter et le mise en page.

**Première version :** 1989.

**Dernière version :** Word2019 en 24 septembre 2018.

**❖ Version utilisée : Word** 2010 et Word 2019.



Figure 17: Word

#### > Android Studio

Android Studio est un environnement de développement pour développer des applications mobiles Android.

**❖** Première version : 1<sup>er</sup> décembre 2014.

**❖ Dernière version :** 13 mai 2021.

**Version utilisée :** 4.0.1



Figure 18 : Android studio

#### V.4.2 Environnement matériel

Marque	DELL	HP pavillon 15
Processeur	I7-6700HQ	I5-9300H
RAM	16 GO	16GO
Disque Dur	1 TO	700GO + 500GO
Carte Graphique	GTX 960M 4 GB	GTX 1650 4Go
Système d'exploitation	Windows 10	Windows 10 Pro (Workstation)
Image de pc	PRIL	

Tableau 2 : Environnement matériel

## VI. Conclusion

Dans ce chapitre, on a présenté le cadre du projet et la méthode utiliser pour sa réalisation. Ensuite, on a présenté l'analyse des besoins dont on cite les besoins fonctionnels, non fonctionnels, ainsi que le backlog. Enfin, on a présenté la planification des releases et la conception. Le prochain chapitre, ce concentra sur le release 1 spécialisé pour la gestion de compte.

## Chapitre 2 : Mise en œuvre de projet

### I. Introduction

La phase de mise en œuvre du projet est l'une des phases les plus importantes en matière de développement des sites web. Dans ce chapitre, nous commençons par l'identification des acteurs. Ensuite, nous présentons l'analyse des besoins dont on cite les besoins fonctionnels, non fonctionnels, ainsi que le backlog. Enfin, la planification des releases et la conception.

#### II. Identification des acteurs

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

Dans notre système, on peut extraire quatre acteurs principaux : Un administrateur, les médecins, les secrétaires, les patients.

### III. Analyse des besoins

#### III.1 Fonctionnels

Cette partie vise à compter les besoins des utilisateurs de notre application.

#### • L'application offre au patient :

- ✓ Gérer son compte.
- ✓ Parcourir la liste de ses médecins.
- ✓ Chercher un médecin.
- ✓ Ajouter un médecin à sa propre liste des médecins.
- ✓ Consulter les informations à propos des médecins.
- ✓ Prendre un rendez-vous.
- ✓ Consulter ces rendez-vous.
- ✓ Consulter l'état de la salle d'attente et
- ✓ Être notifier en cas de prise, de modification ou de suppression d'un rendezvous.

#### • L'application offre au secrétaire :

- ✓ Gérer son compte.
- ✓ Ajouter un nouveau patient (déjà existant dans l'application) dans la liste des patients.
- ✓ Créer un nouveau compte pour un nouveau patient.
- ✓ Consulter la liste des patients.
- ✓ Chercher un patient dans la liste de ses patients.
- ✓ Consulter des informations sur les patients.
- ✓ Réserver un rendez-vous pour un patient.
- ✓ Consulter l'état de la salle d'attente.
- ✓ Gérer l'état du patient (l'ajouter à la salle d'attente, à la salle de la consultation).
- ✓ Gérer les certificats.

#### • L'application offre au médecin :

- ✓ Gérer son compte.
- ✓ Consulter l'espace de ses patients (Consulter ses antécédents, ses consultations, ses ordonnances, ses certificats, ses examens, ...).
- ✓ Consulter les dossiers médicaux des patients. (Antécédents, médicament).
- ✓ Faire une consultation.
- ✓ Ajouter des actions (consultations, ordonnances, certificats, examens, ...)
- ✓ Générer la facture du patient.

#### • L'application offre à l'administrateur :

- ✓ Gérer les comptes des utilisateurs.
- ✓ Consulter la liste des patients.
- ✓ Consulter la liste des médecins.
- ✓ Consulter la liste des secrétaires.

#### III.2 Non fonctionnels

Ce sont les besoins qui permettraient d'améliorer la qualité des services du site comme la convivialité et l'ergonomie des interfaces et l'amélioration du temps de réponse. Parmi ces besoins on cite :

La sécurité: mettre en place un mécanisme d'authentification qui permet à chaque utilisateur de se connecter pour consulter en sécurité son profil.

La sécurité de l'api avec l'authentification token méthode md5 (screetkey+time+user\_agent). Dans l'api on envoi le temps, l'user\_agent et 1 md5 (une chaine de caractère qui contient qui contient le screetkey+time+user\_agent). La partie back end contient une liste des md5 (contient une chaine de caractère pour le temps et l'user agent). Elle compare le nouveau md5 avec la liste. S'il y a un rassemblement il envoie les données correspondantes.

#### **\Lambda** La convivialit\(\text{i} :

L'application doit être facile à utiliser. Il doit présenter un enchainement logique entre les interfaces et un ensemble de liens suffisants pour assurer une navigation rapide et un texte compréhensible, lisible.

Cette application répond facilement aux demandes des utilisateurs.

- « L'utilisateur trouve toutes les données dont il a besoin sans se soucier de les trouver ».
- La disponibilité: Lorsque n'importe quel utilisateur désire consulter l'application, elle doit être disponible.
- \* Temps de réponse : Le temps de réponse doit être le plus court possible.
- \* Refresh : Lorsque l'utilisateur quitte puis réutilise l'application. Les données se rafraîchie automatiquement.

## III.3 Backlog de projet

Le **backlog** de projet est réalisé avant le lancement du développement des **releases** en fonction des besoins des utilisateurs. Son rôle se résume à planifier des **releases** et **sprints** et à clarifier le contenu des sprints pour lancer le travail.

- ❖ Un backlog : est une liste de tâches priorisées définissant les caractéristiques d'un produit. Il est un des éléments fondamentaux de la méthodologie Scrum. [10]
- \* Release: est une nouvelle version du produit, livrée aux utilisateurs. Elle est le fruit de plusieurs Sprints. [11]

Le tableau ci-dessous présente le Product backlog de projet.

	Product Backlog				
ID	User Story	Thème	Priorité	Release	
1.1	En tant qu'utilisateur je veux s'inscrire.		1		
1.2	En tant qu'utilisateur je veux se connecter.	Gérer les	1		
1.3	En tant qu'utilisateur je veux gérer mon compte.		1	1	
1.4	En tant que médecin ou secrétaire je peux créer un compte pour un patient.	comptes.	1		
2.1	En tant que patient je peux ajouter un médecin à ma liste des médecins.		2		
2.2	En tant que médecin je peux ajouter un patient à ma liste des patients.	<b>D</b> 1 ()	2		
2.3	En tant que patient je peux consulter la liste des médecins.	Relation médecins/	2	2	
2.4	En tant que médecin je peux consulter la liste des patients.	patients	2		
2.5	En tant que patient je peux chercher un médecin.	Patients	2		
2.6	En tant que médecin je peux chercher un patient dans ma liste de patient.		2		
3.1	En tant que patient je peux réserver un rendez-vous chez un médecin.		3		
3.2	En tant que médecin/ secrétaire je peux réserver un rendez- vous pour un patient.	Gérer rendez-vous	3	3	
3.3	En tant que patient je peux consulter mes rendez-vous chez mes médecins.		3		
3.4	En tant que médecin ou secrétaire je peux consulter mes rendez-vous.		3		
3.5	En tant que médecin ou secrétaire je peux modifier un rendez-vous.		3		
3.6 4.1	En tant que médecin ou secrétaire je peux supprimer un rendez-vous.		3		
4.1	En tant que secrétaire ou médecin je peux envoyer un patient à la salle d'examen.  En tant que médecin je peux faire une consultation pour un		4		
	patient.	G 1/	4		
4.3	En cours de la consultation, en tant que médecin je peux consulter toutes les informations sur mon patient.	Consulter un patient	4	4	
4.4	En tant que médecin je peux ajouter une nouvelle information (ordonnance, examen, certificat,) dans une consultation.		4		
4.5	En tant que patient je peux consulter mon historique.		4		
5.1	En tant que administrateur, je peux gérer les comptes des utilisateurs.		5		
5.2	En tant que administrateur, je peux la liste des patients.	Partie admin	5	5	
5.3	En tant que administrateur, je peux la liste des médecins.	aumm	5		
5.4	En tant que administrateur, je peux la liste des secrétaires.		5		

Tableau 3 : Product Backlog

#### IV. Planification des releases

Release / Version: Une release est une nouvelle version du produit, livrée aux utilisateurs. Elle est le fruit de plusieurs Sprints.

La figure ci-dessous présente la planification des releases de notre projet.

## Inscription et authentification

## Relation patient médecin

## Gestion des rendez-vous

## Gestion des consultation

Figure 19 : Planification des releases

## V. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation illustre d'une manière globale les différentes actions que peut faire l'utilisateur c'est-à-dire l'interaction qui existe entre l'utilisateur et l'application.

La figure ci-dessous présente le diagramme de cas d'utilisation de notre application pour les trois (3) acteurs : patient, secrétaire et médecin.

- ➤ Patient : Après avoir authentifié, il peut gérer son compte, gérer les médecins, gérer ses rendez-vous et consulter son historique.
- > Secrétaire : Après avoir authentifié, elle peut gérer les patients, gérer les rendez-vous, gérer les actes (certificat, actes).
- ➤ **Médecin :** Après avoir authentifié, il peut consulter un patient et consulte son historique.

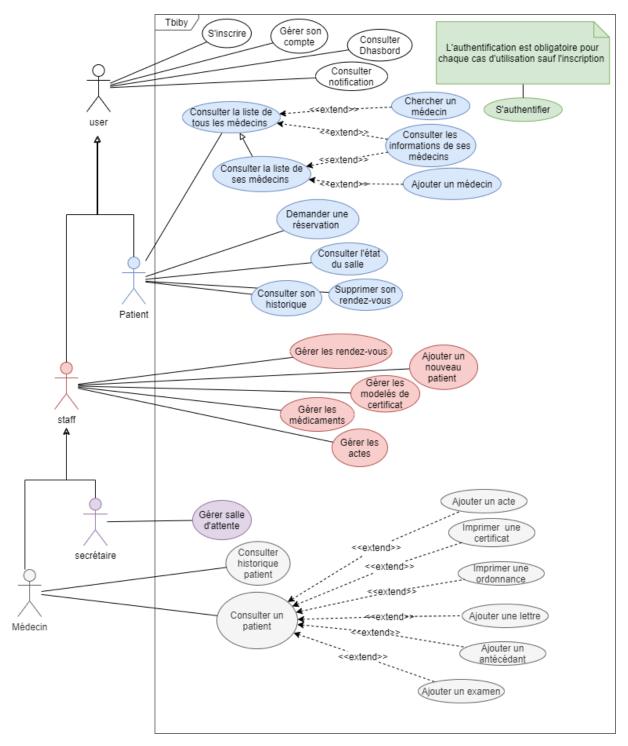


Figure 20 : Diagramme de cas d'utilisation

## VII. Diagramme de classe

- ➤ Le diagramme de classes représente les classes intervenant dans le système. Le diagramme de classe est une représentation statique des éléments qui composent un système et leurs relations.
- > Une classe est composée :
  - Attributs : représentant des données dont les valeurs représentent l'état de l'objet.
  - La méthode : il s'agit des opérations applicables aux objets.

La figure ci-dessous consiste à représenter le diagramme de classe générale de notre application Tbiby.

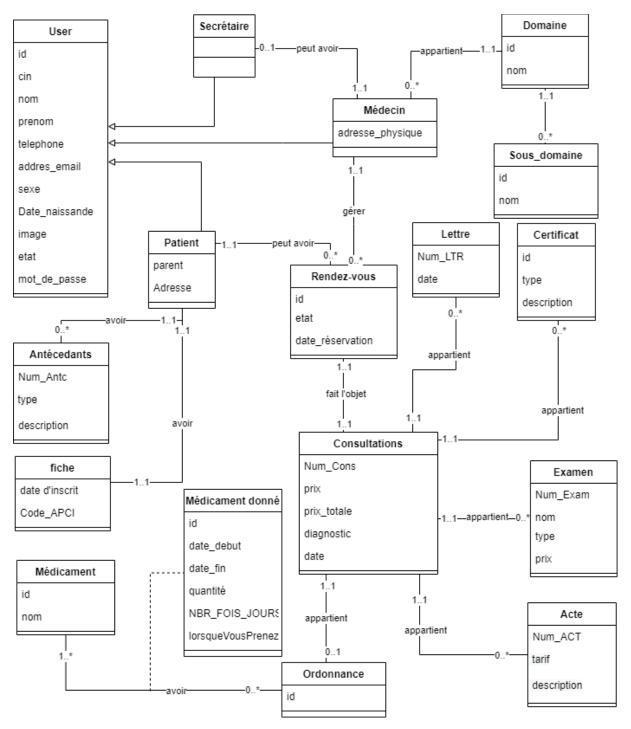


Figure 21 : Diagramme de classe

## VI. Conclusion

Dans ce chapitre, on a commencé par l'identification des acteurs.

Ensuite, on a présenté l'analyse des besoins dont on cite les besoins fonctionnels, non fonctionnels, ainsi que le backlog. Enfin, on a présenté la planification des releases et la conception. Le prochain chapitre, ce concentra sur le release 1 spécialisé pour la gestion de compte.

## Release 1 : Gestion des comptes

#### **I.Introduction**

Dans ce release on va présenter le déroulement pour développer la version de gestion de compte. On a commencé par organiser les sprints. Ensuite, on a défini le backlog des sprints pour la première version ainsi que la conception. En fin, on a présenté chaque sprint.

## II. Organisation des sprints

➤ Sprint : est un rassemblement de personnes impliquées dans un projet afin de se concentrer sur le développement de ce projet. Un sprint dure en général de deux à sept jours.

La figure ci-dessous consiste à présenter l'organisation des sprints pour le release1 correspond à la gestion de compte.



Figure 22 : Sprints de gestion de compte (release1)

## III. Product Backlog Gestion de compte (release 1)

Le tableau ci-dessous consiste à présenter le Product backlog pour le release1 correspond à la gestion de compte.

	Product Backlog (Gestion de compte)				
ID	User Story	Thème	Priorité	Sprint	Release
1.1	En tant qu'utilisateur je veux s'inscrire.		1	1	1
1.2	En tant qu'utilisateur je veux se connecter.	Gérer	2	2	1
1.3	En tant qu'utilisateur je veux gérer mon compte.	les	3	3	
1.4	En tant que médecin ou secrétaire je peux créer un	compte	4	1	4
1.4	compte pour un patient.	S	7	1	7

### IV. Conception de gestion des comptes (release1)

# IV.1 Diagramme de cas d'utilisation de gestion des comptes (release1)

La figure ci-dessous présente le diagramme de cas d'utilisation de gestion de compte.

Le patient et le staff sont hérités des utilisateurs qui peuvent s'inscrire et doivent être authentifiés pour gérer leurs comptes.

Le médecin et la secrétaire sont hérités d'un staff qui peut créer un compte patient après avoir s'authentifier.

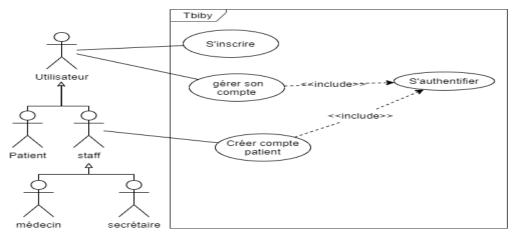


Figure 23 : Diagramme de cas d'utilisation Gestion de compte

## IV.2 Diagramme de classe pour le release 1 (Gestion de compte)

La figure ci-dessous présente le diagramme de classe de gestion de compte.

Les classes médecin, patient et secrétaire sont hérités de la classe User.

Une secrétaire appartient à un et un seul médecin.

Un médecin peut avoir plusieurs patients, et un patient peut avoir plusieurs médecins.

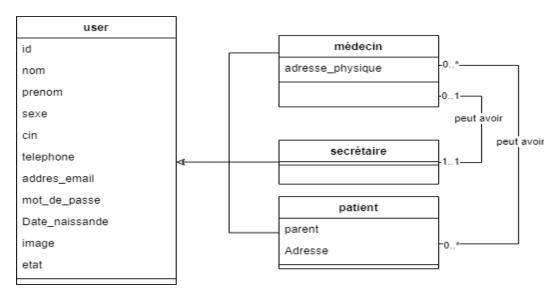


Figure 24 : Diagramme de classe Gestion de compte

## V.Sprint 1: Inscription

## V.1 Cas d'utilisation d'inscription (sprint1 release1)

Le tableau ci-dessous présente le diagramme de cas d'utilisation d'inscription.

	Inscription	
Titre	S'inscrire	
But	Permet de s'inscrire à l'application	
Acteurs	Utilisateur (médecin, patient, secrétaire)	
Pré-cor	ndition Connexion	
	Scenario nominal	
1.	L'utilisateur demande page d'inscription	
	Le système affiche page d'inscription	
	3. L'utilisateur rempli les champs nécessaires	
4. Le système vérifie les données		
5. Le système enregistre les données		
6.	Le système redirige vers la page d'authentification	
	Scenario alternatifs	
4.	<b>,</b>	
existe déjà »		
A chaque étape s'il y a un problème de connexion le système affiche « problème		
	de connexion »	

Tableau 5 : Cas d'utilisation d'inscription

## V.2 Diagramme de séquence d'inscription (sprint1 release1)

La figure ci-dessous présente le diagramme de séquence d'inscription.

L'utilisateur doit remplir les champs nécessaires. Une fois vérifié le système charge l'interface d'authentification.

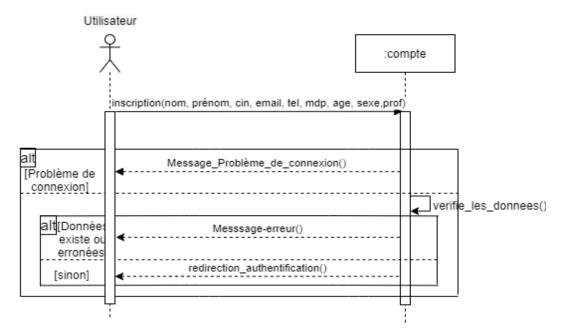


Figure 25 : Diagramme de séquence d'inscription

## V.3Interfaces d'inscription (sprint1)

Cette interface décrit l'inscription. L'utilisateur doit s'inscrire pour qu'il puisse se connecter. Il doit remplir les champs nécessaires. Une fois vérifiée l'interface d'authentification sera affichée.

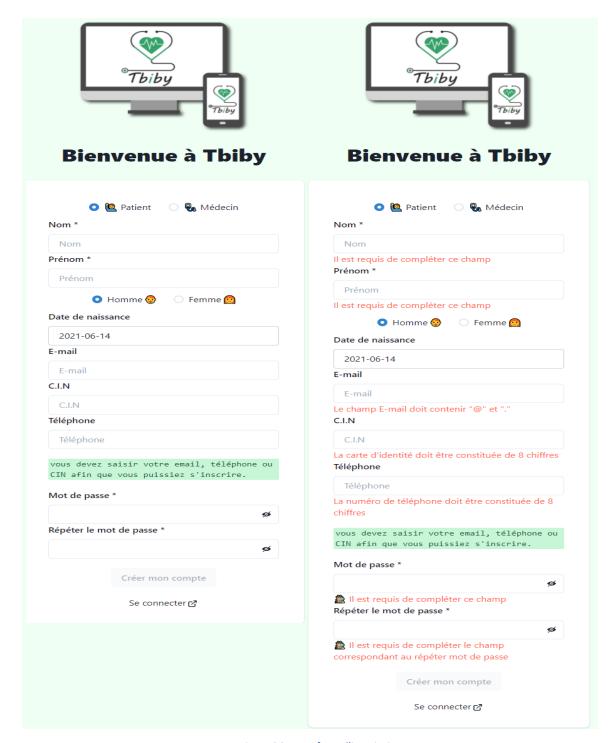


Figure 26: Interface d'inscription



Figure 27 : Erreur email

Figure 28 : Erreur CIN

Figure 29 : Erreur téléphone

### VI. Sprint 2: Authentification

### VI.1 Cas d'utilisation d'authentification (sprint2 release1)

Le tableau ci-dessous présente le diagramme de cas d'utilisation d'authentification.

Authentification		
Titre	S'authentifier	
But	Permet d'authentifier à l'application	
Acteur	Utilisateur (médecin, patient, secrétaire)	
Pré-co	ndition Connexion	
	Scenario nominal	
1.	L'utilisateur demande page d'authentification	
2.	Le système affiche page d'authentification	
	2.1 Si l'utilisateur clique sur créer compte le système affiche la page d'inscription	
3.	L'utilisateur rempli les champs nécessaires	
4.	Le système affiche le Dashboard	
Scenario alternatifs		
3.1 Si l'utilisateur saisie des données incorrectes le système affiche « Entrez votre mail,		
téléphone ou CIN et votre mot de passe »		
A chaque étape s'il y a un problème de connexion le système affiche « problème de		
connexion »		

Tableau 6 : Cas d'utilisation d'authentification

#### VI.2 Diagramme de séquence d'authentification (sprint2 release1)

La figure ci-dessous présente le diagramme de séquence d'authentification.

L'utilisateur doit saisir son login et mot de passe. Une fois vérifié le système charge le Dashboard.

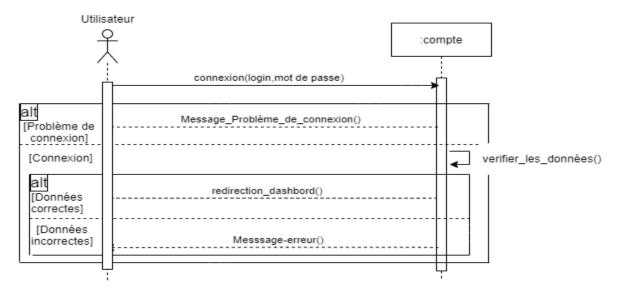


Figure 30 : Diagramme de séquence d'authentification

#### VI.3 Interfaces d'authentification (sprint2 release1)

Cette interface décrit le cas de l'authentification, l'utilisateur doit indiquer son email et mot de passe. Une fois vérifié le système charge le Dashboard si non un message d'erreur survenue sera affiché.



Figure 31: Interface d'authentification

### VII.Sprint 3: Gestion des comptes

#### VII.1 Cas d'utilisation de gestion de compte (sprint3 release1)

Le tableau ci-dessous présente le diagramme de cas d'utilisation de gestion de compte.

Gestion de compte			
Titre	Gestion de compte		
But	Permet de gérer un compte		
Acteurs	Utilisateur (médecin, patient, secrétaire)		
Pré-co	ndition Authentification, connexion		
	Scenario nominal		
1.	L'utilisateur s'authentifier		
2.	Le système affiche Dashboard		
3.	3. L'utilisateur demande page gestion de compte		
4.	4. Le système affiche page gestion de compte		
5.	5. L'utilisateur rempli les champs nécessaire		
6. Le système enregistre les modifications			
Scenario alternatifs			
A chaque étape s'il y a un problème de connexion le système affiche « problème de connexion »			
	COHECAOH //		

Tableau 7 : Cas d'utilisation gestion de compte

#### VII.2 Diagramme de séquence « modifier compte » (sprint3 release1)

La figure ci-dessous présente le diagramme de séquence de gestion de compte.

L'utilisateur doit remplir les champs nécessaires. Une fois vérifié le système synchronise avec la base de données.

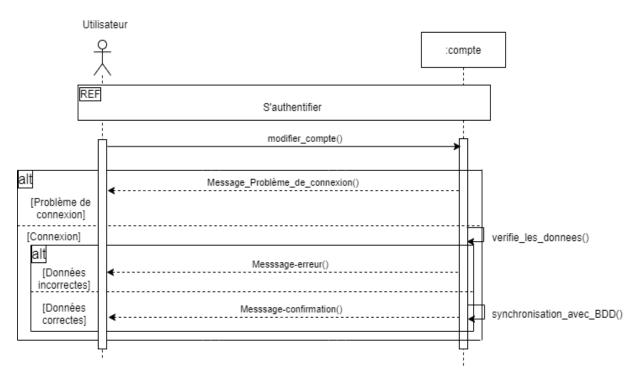


Figure 32 : Diagramme de séquence gestion de compte

### VII.3 Interfaces de gestion de compte (sprint3 release1)

Cette interface décrit le cas de gestion de compte pour le médecin. Il doit remplir les champs nécessaires.

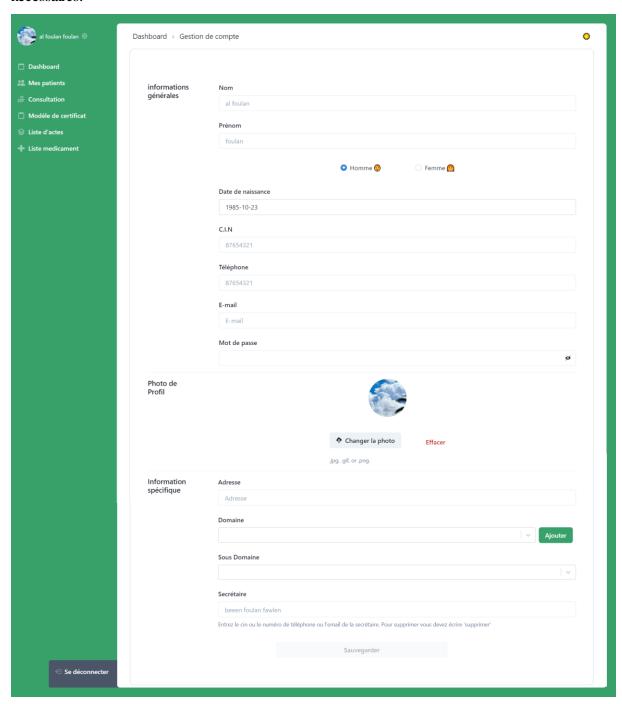


Figure 33 : Interface Gestion de compte d'un médecin

L'interface de gestion de compte pour le patient est la même interface que celle de médecin seulement la partie correspondante aux informations spécifique sera modifié.

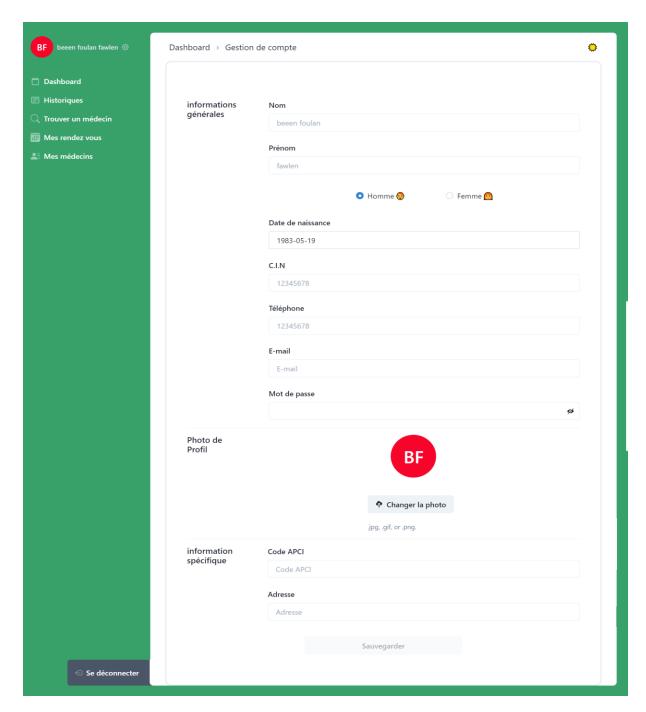


Figure 34 : Interface Gestion de compte d'un patient

#### VIII. Conclusion

Dans ce chapitre, on a présenté le déroulement de travail pour développer la version de gestion de compte.

Dans le chapitre suivant, on va présenter le release 2 qui concerne la relation entre le patient et le médecin.

## Release 2 : Relation entre patient / médecin

#### **I.Introduction**

Dans ce release on va présenter le déroulement de développement de la version qui exprime la relation entre le patient et le médecin. On a commencé par l'organisation des sprints. Ensuite, on a définir le backlog des sprints pour la version 2 ainsi que la conception. En fin, on a présenté chaque sprint.

#### II. Organisation des sprints

La figure ci-dessous consiste à présenter l'organisation des sprints pour le release2 correspond à la relation entre le patient et le médecin.

Sprint 1: Ajout médecin/ patient

Sprint 2: Consultation liste médecin/ patient

Sprint3: Chercher médecin/ patient

Figure 35 : sprints de relation patient / médecin

#### III. Backlog Relation patient/ médecin (release2)

Le tableau ci-dessous consiste à présenter le Product backlog pour le release2 correspond à la relation entre le patient et le médecin.

	Product Backlog release 2 (Relation patient médecin)				
ID	User Story	Thème	Priorité	Sprint	Release
2.1	En tant que patient je peux ajouter un médecin à ma liste des médecins.		2	1	
2.2	En tant que médecin je peux ajouter un patient à ma liste des patients.		2		
2.3	En tant que patient je peux consulter la liste des médecins.	Gérer	2	2	2
2.4	En tant que médecin je peux consulter la liste des patients.	médecins / patients	2		2
2.5	En tant que patient je peux chercher un médecin.		2	3	
2.6	En tant que médecin je peux chercher un patient dans ma liste des patients.		2	3	

Tableau 8 : Product Backlog relation patient/ médecin

### IV. Conception de Relation patient/ médecin (release2)

#### IV.1 Diagramme de cas d'utilisation relation patient/ médecin (release2)

La figure ci-dessous présente le diagramme de cas d'utilisation des relations entre le patient et le médecin.

Après avoir s'authentifier le médecin peut gérer les patients (ajouter, afficher et chercher). Après avoir s'authentifier le patient peut gérer les médecins (ajouter, afficher et chercher).

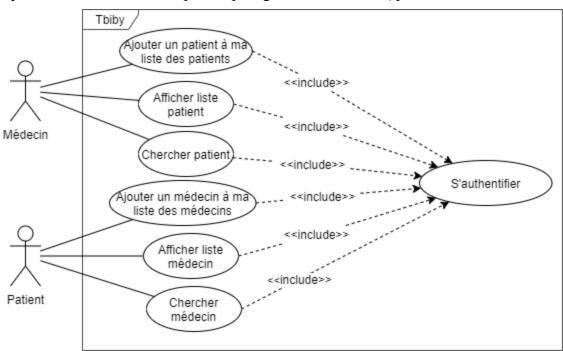


Figure 36 : Diagramme de cas d'utilisation relation patient/ médecin

#### IV.2 Diagramme de classe de relation patient/ médecin (release2)

La figure ci-dessous présente le diagramme de classe de relation entre le patient et le médecin. Les classes médecin, patient et secrétaire sont hérités de la classe User.

Une secrétaire appartient à un et un seul médecin.

Un médecin peut avoir plusieurs patients, et un patient peut avoir plusieurs médecins.

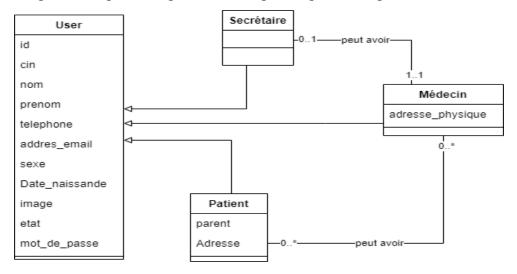


Figure 37 : Diagramme de classe relation patient/ médecin

#### V.Sprint 1 : Ajout médecin/patient

#### V.1 Cas d'utilisation d'ajout médecin (sprint1 release2)

Le tableau ci-dessous présente le diagramme de cas d'utilisation d'ajout médecin.

	Ajout médecin		
Titre	Ajout médecin		
But	Permet au patient d'ajouter un médecin		
Acteur	s Patient		
Précon	ditions Authentification, connexion		
	Scenario nominal		
1.	Le patient s'authentifier		
2.	Le système affiche Dashboard		
3.	Le patient demande page mes médecins		
4.	Le système affiche page liste médecin		
5.	5. Le patient saisie tél/mail pour ajouter un médecin et clique sur ajouter		
6.	Le système ajout le médecin et l'affiche dans la liste		
Scenario alternatifs			
	1. Si les informations saisi sont incorrectes le système redirection à l'étape 5.		
	2. A chaque étape s'il y a un problème de connexion le système affiche « problème de connexion »		

Tableau 9 : Cas d'utilisation d'ajout médecin

#### V.2Diagramme de séquence d'ajout médecin (sprint1 release2)

La figure ci-dessous présente le diagramme de séquence d'ajout médecin.

Le patient doit saisir les informations du médecin. Une fois vérifié le système enregistre cette relation.

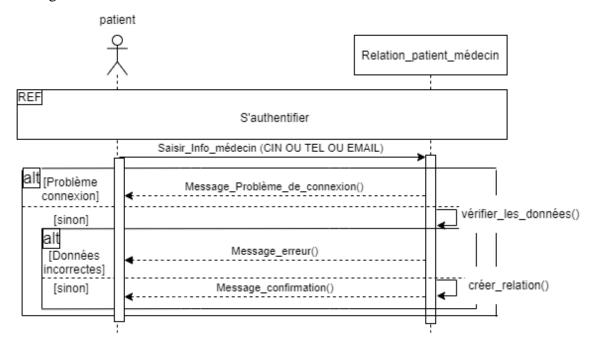


Figure 38 : Diagramme de séquence d'ajout médecin

#### V.3 Interfaces d'ajout médecin (sprint1 release2)



Figure 39 : Interface d'ajout médecin

#### V.4 Cas d'utilisation d'ajout patient (sprint1 release2)

Le tableau ci-dessous présente le diagramme de cas d'utilisation d'ajout patient.

Ajout patient			
Titre	Ajout patient		
But	Permet au médecin d'ajouter un patient		
Acteur	s <b>Médecin</b>		
Pré-co	ndition Authentification, connexion		
	Scenario nominal		
1.	Le patient s'authentifier		
2.	Le système affiche Dashboard		
3.	Le médecin demande page mes patients		
4.	4. Le système affiche page liste patient		
5. Le médecin saisie tél/mail pour ajouter un patient et clique sur ajouter			
6.	6. Le système ajout le patient et l'affiche dans la liste		
Scenario alternatifs			
1. Si les informations saisi sont incorrectes le système redirection à l'étape 5.			
2. A chaque étape s'il y a un problème de connexion le système affiche			
	« problème de connexion »		

Tableau 10 : Cas d'utilisation d'ajout patient

#### V.5 Diagramme de séquence d'ajout patient (sprint1 release2)

La figure ci-dessous présente le diagramme de séquence d'ajout patient.

Le médecin doit saisir les informations du médecin. Une fois vérifié le système enregistre cette relation.

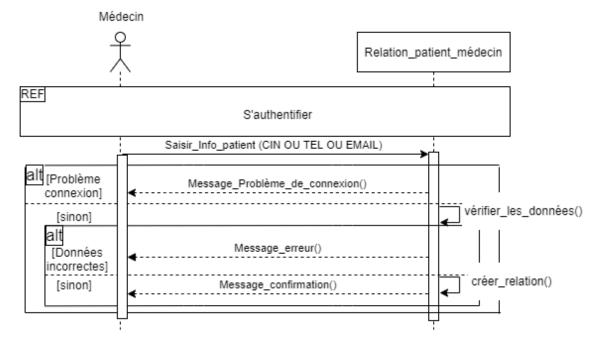


Figure 40 : Diagramme de séquence d'ajout patient

### V.6 Interfaces d'ajout patient (sprint1 release2)

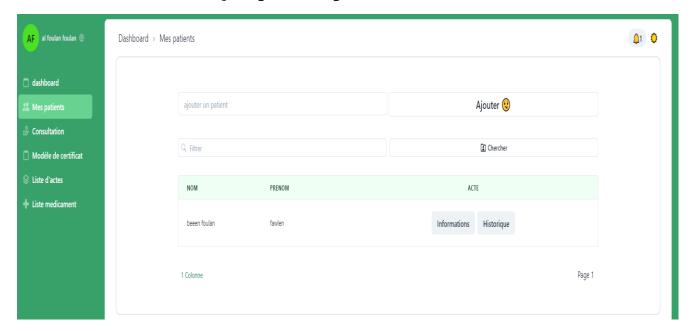


Figure 41 : Interface d'ajout patient

### VI. Sprint 2: Consultation liste médecin/patient

## VI.1 Cas d'utilisation de consultation liste médecin (sprint2 release2)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui permet au médecin de consulter la liste des médecins.

	Consultation liste médecin
Titre	Consultation liste médecin
But	Permet de consulter la liste des médecins
Acteur	S Patient
Pré-co	ndition Authentification, connexion
	Scenario nominal
2.	Le patient s'authentifier  Le système affiche Dashboard  a. Le patient peut choisir « Mes patients » pour afficher la liste de ses médecins  b. Le patient peut choisir « Trouver médecin » pour afficher la liste de tous les médecins  Le système affiche liste des médecins  a. Le patient peut choisir « information» pour consulter les informations de médecin  b. Le patient peut choisir « Rendez-vous » pour consulter le Dashboard de médecin
	Scenario alternatifs
1	. A chaque étape s'il y a un problème de connexion le système affiche « problème de connexion »

Tableau 11 : Cas d'utilisation consultation liste médecin

## VI.2 Diagramme de séquence consultation liste médecin (sprint2 release2)

La figure ci-dessous présente le diagramme de séquence de consultation liste médecin.

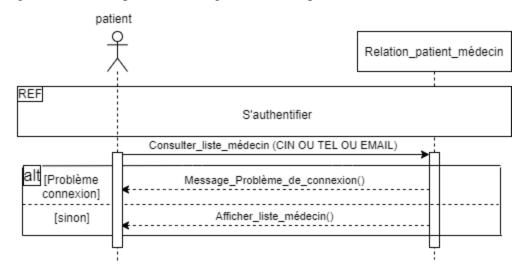


Tableau 12 : Diagramme de séquence consultation liste médecin

#### V.3 Interfaces de Consultation liste médecin (sprint2 release2)

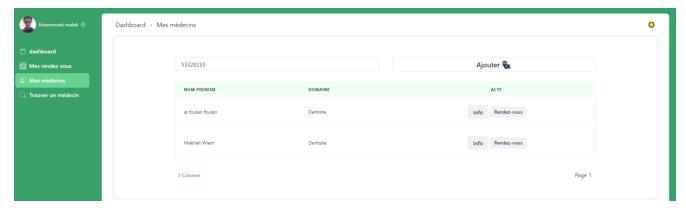


Figure 42 : Interface consultation liste médecin

#### VI.4 Cas d'utilisation consultation liste patient (sprint2 release2)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui permet au médecin de consulter la liste de ses patients.

	Consultation liste patient	
Titre	Consulter liste médecin patient	
But	Permet au médecin de consulter la liste de ses patients	
Acteur	Médecin	
Pré-co	ndition Authentification, connexion	
	Scenario nominal	
1.	Le médecin s'authentifier	
2.	Le système affiche Dashboard	
3.	Le médecin demande page mes patients	
4.	Le système affiche page liste patient	
a. Le médecin clique sur « Information» pour consulter des informations sur le		
	patient	
b. Le médecin clique sur « Historique» pour consulter l'historique de son patient		
5. Le système affiche les informations de patient		
Scenario alternatifs		
1. A chaque étape s'il y a un problème de connexion le système affiche « problème de		
	connexion »	

Tableau 13 : Cas d'utilisation consultation liste patient

# VI.5 Diagramme de séquence consultation liste patient (sprint2 release2)

La figure ci-dessous présente le diagramme de séquence de consultation liste patient.

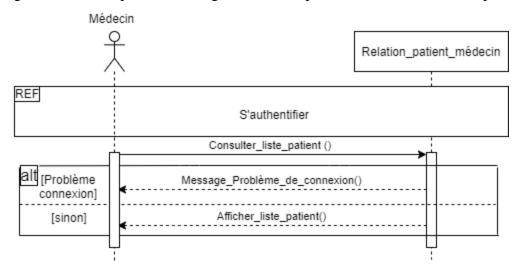


Figure 43 : Diagramme de séquence consultation liste patient

### VI.6 Interfaces de consultation liste patient (sprint2 relase2)

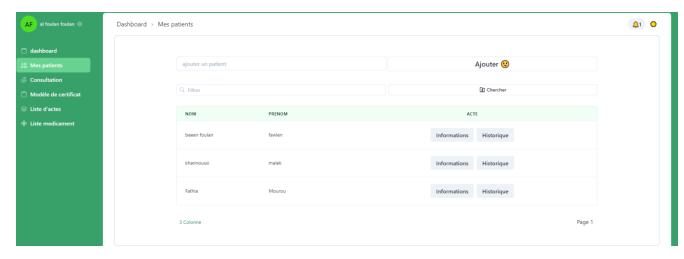


Figure 44 : Interface consultation liste médecin

### VII. Sprint 3: Chercher liste médecin/patient

### VII.1 Cas d'utilisation chercher médecin (sprint3 release3)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui permet au patient la recherche d'un médecin.

Chercher médecin			
Titre	Chercher médecin		
But	Permet de chercher un médecin		
Acteur	S Patient		
Pré-co	ndition Authentification, connexion		
	Scenario nominal		
1.	Le patient s'authentifier		
2.	Le système affiche Dashboard		
3.	3. Le patient demande page «chercher médecins »		
4.	4. Le système affiche page «chercher médecin »		
5.	5. Le patient saisie tél ou email ou choisir un domaine pour chercher un médecin		
6.	Le système affiche le médecin		
Scenario alternatifs			
1.	Si les informations sont incorrectes le système affiche « vous devez saisir le mail, tél		
	ou domaine »		
2.	A chaque étape s'il y a un problème de connexion le système affiche « problème de connexion »		

Tableau 14 : Cas d'utilisation chercher médecin

#### VII.2 Diagramme de séquence chercher médecin (sprint3 release3)

La figure ci-dessous présente le diagramme de séquence de la recherche d'un médecin. Le patient doit saisir les informations du médecin. Une fois vérifié le système affiche le médecin cherché.

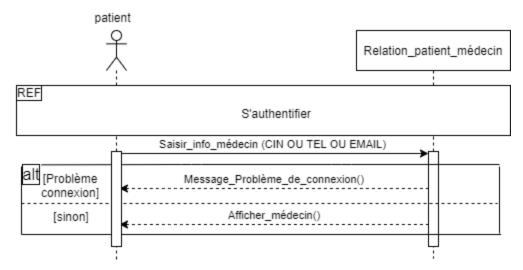


Figure 45 : Diagramme de séquence chercher médecin

#### VII.3 Interfaces de chercher médecin (sprint3 relase3)

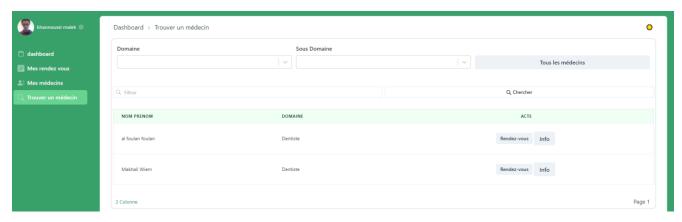


Figure 46 : Interface chercher médecin

#### VII.4 Cas d'utilisation de Chercher patient (sprint3 release3)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui au médecin de chercher un patient.

Chercher Patient			
Titre	Chercher patient		
But	Permet de chercher un patient		
Acteurs	Médecin		
Pré-condition	Authentification, connexion		
	Scenario nominal		
1. Le médecin s	'authentifier		
2. Le système at	ffiche Dashboard		
3. Le médecin d	emande page mes patients		
4. Le système at	ffiche page mes patients		
5. Le médecin saisie cin ou tél ou email pour chercher un patient			
6. Le système at	ffiche le patient		
	Scenario alternatifs		
1. Si données incorrectes le système affiche « Vous n'avez aucun patient avec ses informations »			
2. A chaque étape s'il y a un problème de connexion le système affiche « problème de connexion »			

Tableau 15: Cas d'utilisation chercher patient

#### VII.5 Cas d'utilisation de chercher patient (sprint3 release3)

La figure ci-dessous présente le diagramme de séquence de la recherche d'un patient. Le médecin doit saisir les informations du patient. Une fois vérifié le système affiche le patient cherché.

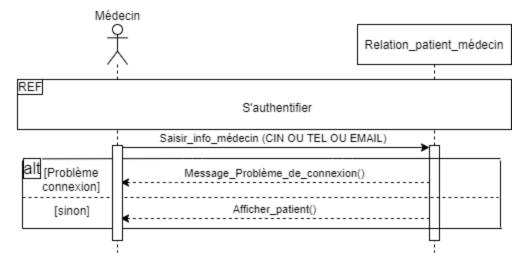


Figure 47: Interface chercher patient

### VII.6 Interfaces de chercher patient (sprint3 release3)



Figure 48: Interface chercher patient

### VIII. Conclusion

Dans ce chapitre, on a présenté le déroulement de travail pour développer la version pour la relation entre le patient et le médecin.

Dans le chapitre suivant, on va présenter le release 3 qui concerne la gestion de rendez-vous.

### Release 3: Gestion des rendez-vous

#### I. Introduction

Dans ce release on va présenter le déroulement pour développer la version qui concerne la gestion de rendez-vous. On a commencé par organiser les sprints. Ensuite, on a définir le backlog des sprints pour la version 3 ainsi que la conception. En fin, on a présenté chaque sprint.

#### II. Organisation des sprints

La figure ci-dessous consiste à présenter l'organisation des sprints pour le release3 correspond à la gestion des rendez-vous.



Figure 49 : Sprints de gestion des rendez-vous

#### III. Backlog relation patient/ médecin (release3)

Le tableau ci-dessous consiste à présenter le Product backlog pour le release3 correspond à la gestion des rendez-vous.

	Product Backlog release 3 (Relation patient médecin)				
ID	<b>User Story</b>	Thème	Priorité	Sprint	Release
3.1	En tant que patient je peux réserver un rendez-vous chez un médecin.		3	1	
3.2	En tant que médecin je peux réserver un rendezvous pour un patient.		3		
3.3	En tant que patient je peux consulter mes rendez- vous chez un médecin.	Gérer	3	2	3
3.4	En tant que médecin ou secrétaire je peux consulter mes rendez-vous.	rendez- vous	3		3
3.5	En tant que médecin je peux modifier la date d'un rendez-vous.		3	3	
3.6	En tant que médecin je peux supprimer un rendezvous.		3	S	

Tableau 16 : Product Backlog de gestion des rendez-vous

### IV. Conception de Gestion des rendez-vous (release3)

## IV.1 Diagramme de cas d'utilisation Gestion des rendez-vous (release3)

La figure ci-dessous présente le diagramme de cas d'utilisation de gestion des rendezvous.

Après avoir s'authentifier le médecin peut gérer les rendez-vous (ajouter, afficher et modifier).

Après avoir s'authentifier le patient peut gérer les médecins (ajouter, afficher).

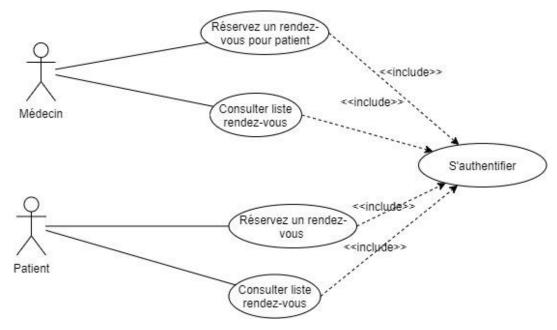


Figure 50 : Diagramme de cas d'utilisation Gestion des rendez-vous

#### IV.2 Diagramme de classe Gestion des rendez-vous (release3)

La figure ci-dessous présente le diagramme de classe de gestion des rendez-vous.

Les classes médecin, patient et secrétaire sont hérités de la classe User.

Un médecin peut avoir plusieurs rendez-vous des plusieurs patients.

Un patient peut avoir plusieurs rendez-vous chez plusieurs médecins.

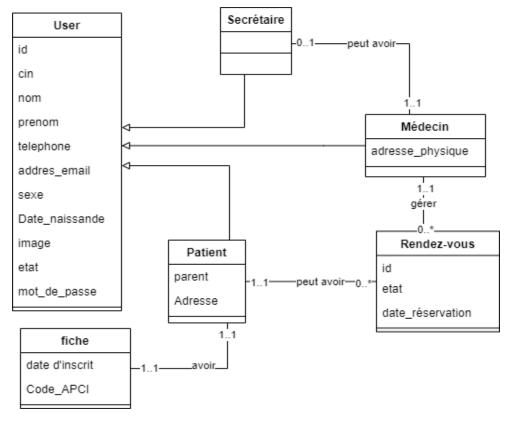


Figure 51 : Diagramme de classe Gestion des rendez-vous

#### V. Sprint 1: Réservation rendez-vous médecin pour un patient

## V.1 Cas d'utilisation Réservation rendez-vous pour un patient (sprint1 release3)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui permet au médecin la réservation d'un rendez-vous pour un patient.

Réservation rendez-vous Patient		
Titre	Réserver rendez-vous	
But	Permet de réserver un rendez-vous	
Acteurs	Médecin, secrétaire	
Pré-condition	Authentification, connexion	
Scenario nominal		

- 1. Le médecin s'authentifier
- 2. Le système affiche Dashboard
- 3. Le médecin choisir date et choisir un patient
  - a. Choisir le patient depuis la liste de ses patients
  - b. Ajouter le patient à sa liste (à travers son email, tél ou CIN)
  - c. Créer un nouveau compte pour le patient
- 4. Le système ajoute le rendez-vous au Dashboard

#### Scenario alternatifs

- 1. Si données incorrectes le système affiche « Vérifier données » sinon affiche « patient ajouté. Vous devez le choisir depuis la liste ».
- 2. A chaque étape s'il y a un problème de connexion le système affiche « problème de connexion »

Tableau 17 : Cas d'utilisation réservation des rendez-vous pour un patient

## V.2 Diagramme de séquence « Réserver rendez-vous pour un patient » (sprint1 release3)

La figure ci-dessous présente le diagramme de séquence de la réservation d'un rendezvous pour un patient.

La secrétaire doit choisir une date et un patient pour lui réserver un rendez-vous. Une fois le choix est terminé le système envoi une notification au patient.

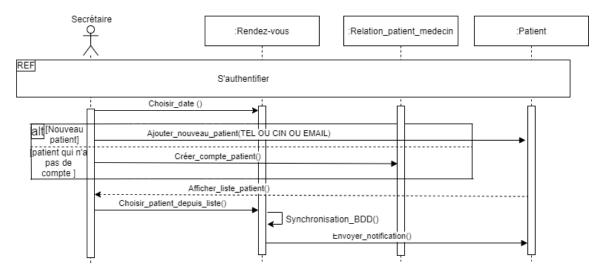


Figure 52 : Diagramme de séquence réservation d'un rendez-vous pour un patient

# V.3 Interfaces de Réservation rendez-vous pour un patient (sprint1 release3)



Figure 53 : Interface Dashboard médecin



Figure 54 : Interface sélectionner patient



Figure 55 : Ajout patient

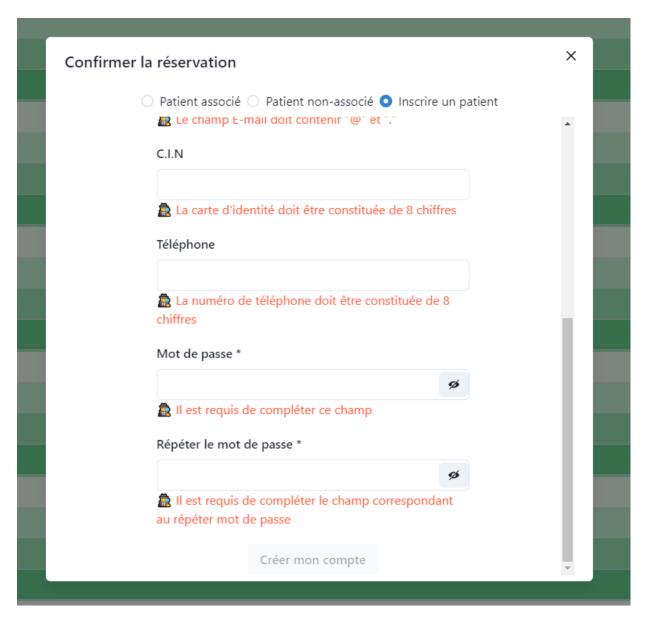


Figure 56 : Créer compte patient

### VI. Sprint 2 : Réservation rendez-vous patient chez un médecin

## VI.1 Cas d'utilisation Réservation rendez-vous chez un médecin (sprint2 release3)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui permet au patient la réservation d'un rendez-vous chez un médecin.

Réservation rendez-vous			
Titre Réserver rendez-vous			
But Permet de réserver un rendez-vous			
Acteurs	Patient		
Pré-condition	Authentification, connexion		
	Scenario nominal		
1. Le patient s'a	nuthentifier		
2. Le système at	ffiche Dashboard		
3. Le patient cli	que sur « mes médecins »		
4. Le système at	4. Le système affiche page « mes médecin »		
5. Le patient che	5. Le patient choisir un médecin et clique sur « rendez-vous »		
6. Le système at	6. Le système affiche Dashboard de ce médecin		
7. Le patient che	oisir une date		
8. Le système affiche alerte de confirmation			
a. Le patient confirme le rendez-vous et le système enregistre le sur le			
Dashboard			
Scenario alternatifs			
1. A chaque	1. A chaque étape s'il y a un problème de connexion le système affiche		

Tableau 18 : Cas d'utilisation réservation rendez-vous chez un médecin

« problème de connexion »

# VI.2 Diagramme de séquence Réservation rendez-vous chez un médecin (sprint2 release3)

La figure ci-dessous présente le diagramme de séquence de la réservation d'un rendezvous chez un médecin.

Le patient doit choisir un médecin et consulter sa disponibilité pour réservez un rendez-vous. Une fois terminé le système envoi une notification au médecin dans laquelle il peut accepter ou refuser ce rendez.

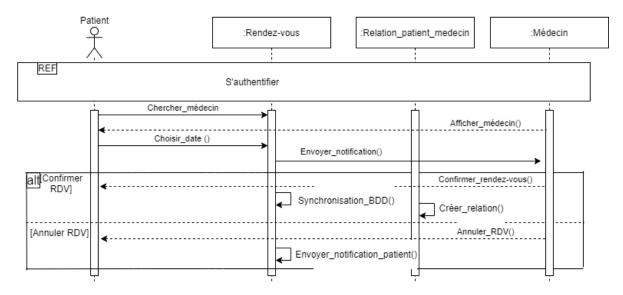


Figure 57 : Diagramme de séquence réservation rendez-vous chez un médecin

## VI.3 Interfaces de Réservation rendez-vous chez un médecin (sprint2 release3)

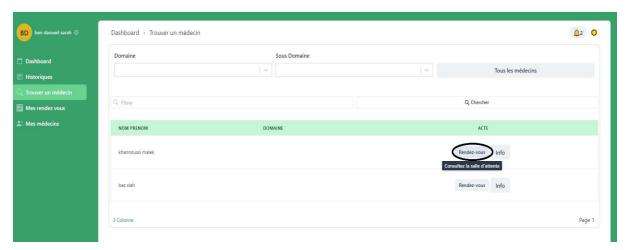


Figure 58 : Interface réservation d'un rendez-vous chez un médecin

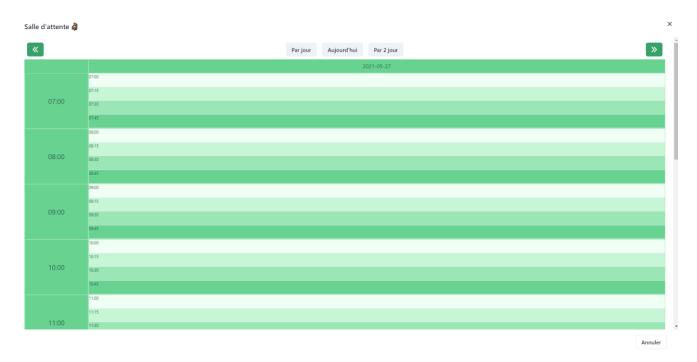


Figure 59 : Interface salle d'attente

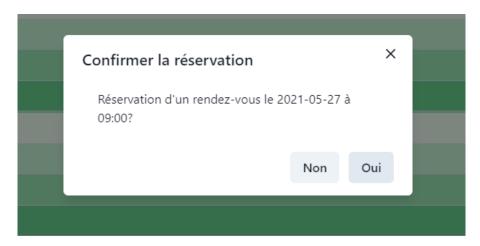


Figure 60 : Interface confirmation rendez-vous

#### VII. Sprint 3: Consultation rendez-vous

## VII.1 Cas d'utilisation Consultation rendez-vous patient (sprint3 release3)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui permet au patient de consulter la liste de ces rendez-vous.

	Consultation rendez-vous Patient			
Titre	Consulter rendez-vous			
But	Permet de consulter un rendez-vous			
Acteurs	Patient			
Pré-condition	Authentification, connexion			
	Scenario nominal			
1. Le patient s'authentifier				
2. Le système affiche Dashboard				
3. Le patient clique sur « Mes rendez-vous »				
4. Le système at	ffiche la liste de ses rendez-vous			
a. Le pat médec	ient clique sur « info » pour consulter des informations sur le in			
b. Le patient peut supprimer un rendez-vous on cliquant sur le bouton				
Scenario alternatifs				
A chaque étape s'il	y a un problème de connexion le système affiche « problème			
	de connexion »			

Tableau 19 : Cas d'utilisation consultation rendez-vous patient

## VII.2 Diagramme de séquence Consultation rendez-vous patient (sprint3 release3)

La figure ci-dessous présente le diagramme de séquence qui permet au patient de consulter la liste de ses rendez-vous.

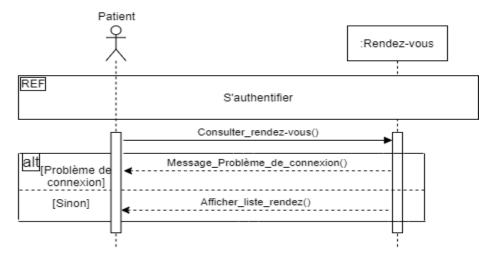


Figure 61 : Diagramme de séquence consultation rendez-vous patient

## VII.3 Interfaces de Consultation rendez-vous patient (sprint3 release3)

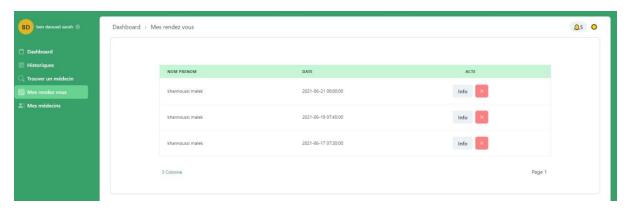


Figure 62: Interface consultation rendez-vous patient

## VII.4 Cas d'utilisation Consultation rendez-vous médecin (sprint3 release3)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui permet au médecin de consulter la liste de ses rendez-vous.

Consultation rendez-vous Médecin				
Titre	Consulter rendez-vous			
But	Permet de consulter un rendez-vous			
Acteurs	Médecin			
Pré-condition Authentification, connexion				
Scenario nominal				
1. Le médecin s'authentifier				
2. Le système affiche Dashboard (calendrier)				
a. On cliquant sur une date le médecin peut ajouter un rendez-vous				
3. Le médecin choisir le format d'affichage par jour, aujourd'hui, par				
semaine ou une date.				
4. Le système répond au choix choisi par le médecin				
Scenario alternatifs				
1. A chaque étape s'il y a un problème de connexion le système affiche				
« problème d	« problème de connexion »			

Tableau 20 : Cas d'utilisation consultation rendez-vous médecin

# VII.5 Diagramme de séquence Consultation rendez-vous médecin (sprint3 release3)

La figure ci-dessous présente le diagramme de séquence qui permet au patient de consulter la liste de ses rendez-vous.

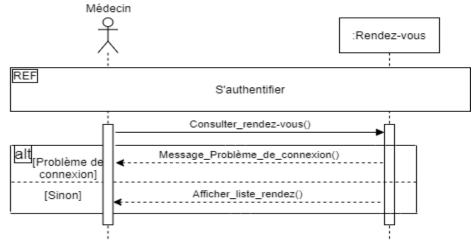


Figure 63 : Diagramme de séquence consultation rendez-vous médecin

#### VII.6 Interfaces Consultation rendez-vous médecin (sprint3 release3)

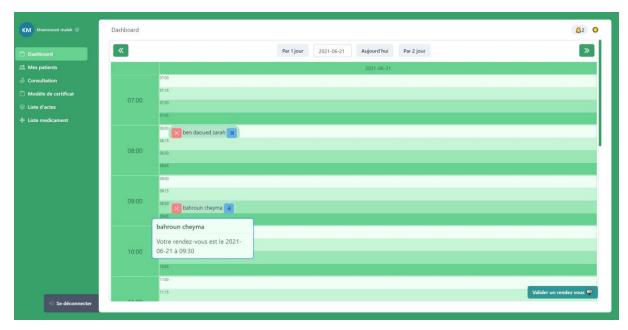


Figure 64 : Interface consultation rendez-vous médecin

#### VIII. Conclusion

Dans ce chapitre, on a présenté le déroulement de travail pour développer la version qui concerne la gestion de rendez-vous.

Dans le chapitre suivant, on va présenter le release 4 qui concerne la gestion de consultation.

### **Release 4 : Gestion des consultations**

#### I. Introduction

Dans ce release en va présenter le déroulement pour développer la version qui concerne la gestion de consultation. On a commencé par organiser les sprints. Ensuite, on a définir le Backlog des sprints pour la version 4 ainsi que la conception. En fin, on a présenté chaque sprint.

#### II. Organisations des sprints

La figure ci-dessous consiste à présenter l'organisation des sprints pour le release4 correspond à la gestion des consultations.

### Sprint 1: Ajout consultation

## Sprint 2: Consultation historique

Figure 65 : Sprints gestion de consultation

### III. Backlog Gestion consultation (release4)

Le tableau ci-dessous consiste à présenter le Product backlog pour le release4 correspond à la gestion des consultations.

Product Backlog release 4 (Gestion de consultation)							
ID	User Story	Thème	Priorité	Sprint	Release		
4.1	En tant que secrétaire ou médecin je peux envoyer un patient à la salle d'examen.		4	1			
4.2	En tant que médecin je peux faire une consultation pour un patient.		4				
4.3	En cours de la consultation, en tant que médecin je peux consulter tous les informations sur mon patient.	Gérer rendez- vous	4	2	4		
4.4	En tant que médecin je peux ajouter une nouvelle information (ordonnance, examen, certificat,) dans une consultation.		4				

Tableau 21 : Product backlog gestion consultation

#### IV. Conception Gestion de consultation (release4)

## IV.1 Diagramme de cas d'utilisation Gestion des consultations (release4)

La figure ci-dessous présente le diagramme de cas d'utilisation de gestion des consultations.

Après avoir authentifié, un patient peut consulter son historique.

Après avoir authentifié, une secrétaire peut gérer les certificats, les actes et les médicaments.

Après avoir s'authentifier le médecin peut faire une consulter pour un patient.

Dans chaque consultations, le médecin peut ajouter plusieurs actes (ordonnance, acte, examen, lettre, antécédent)

Après avoir s'authentifier, Un médecin peut consulter l'historique d'un patient.

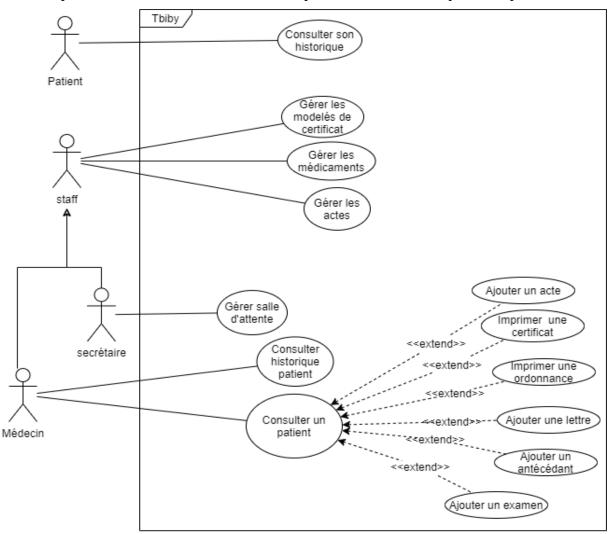


Figure 66 : Diagramme de cas d'utilisation gestion de consultation

#### IV.2 Diagramme de classe Gestion de consultation (release4)

La figure ci-dessous présente le diagramme de classe de gestion des consultations.

Les classes médecin, patient et secrétaire sont hérités de la classe User.

Un médecin peut avoir plusieurs patients, et un patient peut avoir plusieurs médecins.

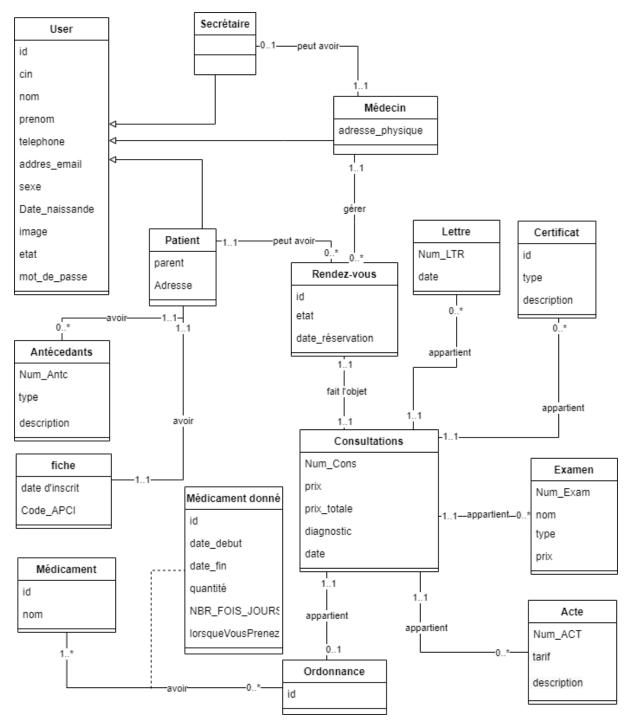


Figure 67 : Diagramme de classe gestion de consultation

#### V. Sprint 1: Ajout consultation

#### V.1 Cas d'utilisation Ajout consultation (sprint 1 release 4)

Le tableau ci-dessous présente le diagramme de cas d'utilisation d'ajout consultation.

Ajout consultation		
Titre	Ajouter consultation	
But	Permet d'ajouter une consultation	
Acteurs	Médecin	
Pré-condition	Authentification, connexion	
Scenario nominal		

- 1. Le médecin s'authentifier
- 2. Le système affiche Dashboard (calendrier)
- 3. On cliquant sur une date le médecin peut ajouter un rendez-vous
- 4. Le système ajoute le rendez-vous sur le Dashboard
- 5. Le médecin envoi le patient à la salle de consultation
- 6. Le système affiche la page de consultation
  - a. Le médecin clique sur le bouton supprimer pour supprimer la réservation du patient
  - b. Le médecin clique sur le bouton de salle d'attente pour renvoyer le médecin à la salle d'attente
  - c. Le médecin clique sur le nom de patient pour commencer la consultation
- 7. Le système affiche la consultation
- 8. Le médecin ajoute des actions (ordonnance, certificat, acte, lettre, examen, antécédents)
  - a. Le médecin clique sur « Informations » pour afficher les informations de ce patient
  - b. Le médecin clique sur « Historique » pour consulter l'historique du patient
  - c. Le médecin clique sur « Antécédent » pour consulter l'historique de patient
- 9. Le système enregistre ces actions.

#### Scenario alternatifs

1. A chaque étape s'il y a un problème de connexion le système affiche « problème de connexion »

Tableau 22 : Cas d'utilisation ajout consultation

#### V.2 Diagramme de séquence Ajout consultation (sprint 2 release 4)

La figure ci-dessous présente le diagramme de séquence d'ajout consultation.

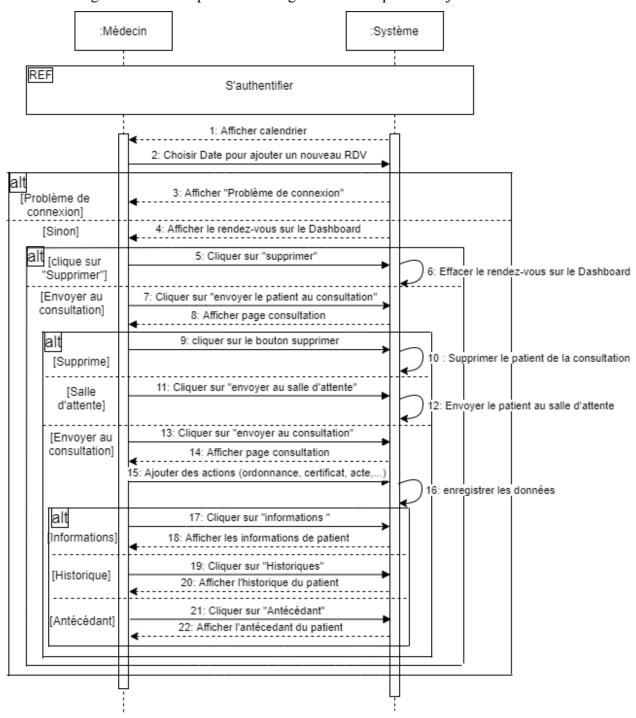


Figure 68 : Diagramme de séquence ajout consultation

#### VI. Sprint 2: Consultation historique

# VI.1 Cas d'utilisation Consultation historique patient (sprint 2 release 4)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui permet au médecin de consulter l'historique de son patient.

Consultation historique patient				
Titre	Consultation historique patient			
But Permet de consulter l'historique de patient				
Acteurs	Médecin			
Pré-condition Authentification, connexion				
Scenario nominal				
1. Le médecin s'authentifier				
2. Le système affiche Dashboard				
3. Le médecin clique sur « Mes patients »				
4. Le système affiche liste patients				
5. Le médecin clique sur « Historique» de l'un des patients				
6. Le système affiche l'historique de ce patient				
Scenario alternatifs				
1. A chaque étape s'il y a un problème de connexion le système affiche « problème de connexion »				

Tableau 23 : Cas d'utilisation consultation historique patient

# VI.2 Diagramme de séquence Consultation historique patient (sprint 2 release 4)

La figure ci-dessous présente le diagramme de séquence qui permet au médecin de consulter l'historique de ses patients.

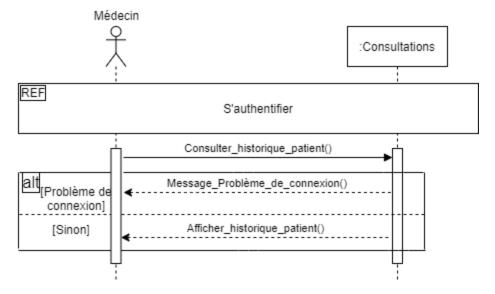


Figure 69 : Diagramme de séquence consultation historique patient

#### VI.3 Cas d'utilisation Consultation historique (sprint 2 release 4)

Le tableau ci-dessous présente le diagramme de cas d'utilisation qui permet au patient de consulter son historique.

Consultation historique				
Titre	Consultation historique			
But	Permet de consulter l'historique			
Acteurs	Patient			
Pré-condition	Authentification, connexion			
Scenario nominal				
1. Le patient s'authentifier				
2. Le système affiche Dashboard				
3. Le patient clique sur « Mon historique »				
4. Le système affiche l'historique				
Scenario alternatifs				
1. A chaque étape s'il y a un problème de connexion le système affiche				
« problème de connexion »				

Tableau 24 : Cas d'utilisation consultation historique

# VI.4 Diagramme de séquence Consultation historique (sprint 2 release 4)

La figure ci-dessous présente le diagramme de séquence qui permet au patient de consulter son historique.

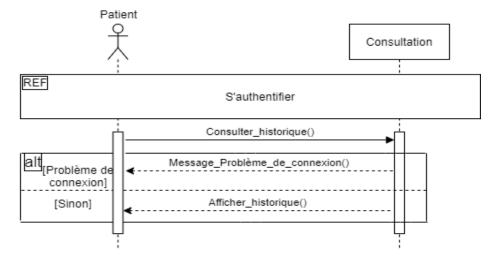


Figure 70: Diagramme de séquence consultation historique

#### VII. Conclusion

Dans ce chapitre, on a présenté le déroulement de travail pour développer la version qui concerne la gestion de consultation.

Dans le chapitre suivant, on va présenter le release 5 qui concerne la partie administrative et la partie mobile.

# Release 5 : Partie d'administration & Partie mobile

#### I. Introduction

Dans ce release on va présenter le déroulement pour développer la version de gestion de compte. On a commencé par organiser les sprints. Ensuite, on a défini le backlog des sprints pour la version 5 ainsi que la conception. En fin, on a présenté chaque sprint.

#### II. Organisation des sprints

La figure ci-dessous consiste à présenter l'organisation des sprints pour le release5 correspond à la partie administrative et la partie mobile.

### Sprint 1: Partie d'administration

## Sprint 2: Partie mobile

Figure 71 : Sprints partie admin & mobile

#### III. Product Backlog

Le tableau ci-dessous consiste à présenter le Product backlog pour le release5 correspond à la partie administrative et la partie mobile.

Product Backlog release 5 (Partie administrative)							
ID	User Story	Thème	Priorité	Sprint	Release		
5.1	En tant que administrateur je peux gérer les comptes des utilisateurs.		5	1	5		
5.2	En tant que administrateur, je peux consulter la liste des patients.	Partie	5				
5.3	En tant que administrateur, je peux consulter la administr liste des médecins. ative		5	1	3		
5.4	En tant que administrateur je peux consulter la liste des secrétaires.		5				

Figure 72: Product Backlog partie admin

#### IV. Conception

#### IV.1 Diagramme de cas d'utilisation de la partie admin (release5)

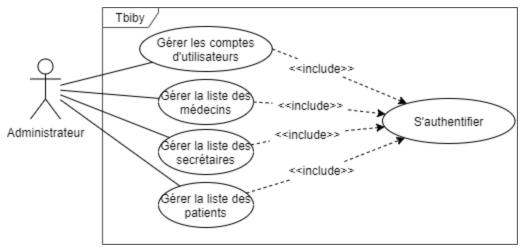


Figure 73 : Diagramme de cas d'utilisation partie admin

#### V. Sprint 1: Partie administrative

#### V.1 Interfaces admin (release5)

Cette interface permet à l'administrateur de consulter la liste des médecins, les patients et la secrétaire de chaque médecin.

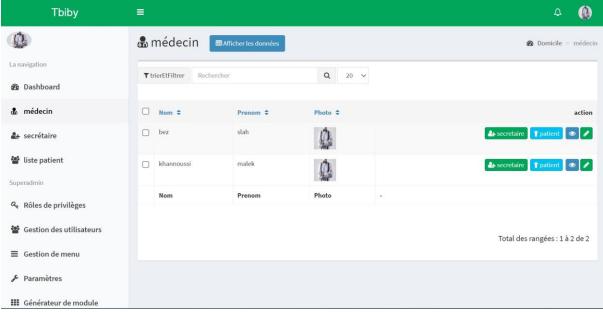


Figure 74 : liste médecin

Cette interface permet à l'administrateur de gérer (afficher et supprimer) la liste des secrétaires.

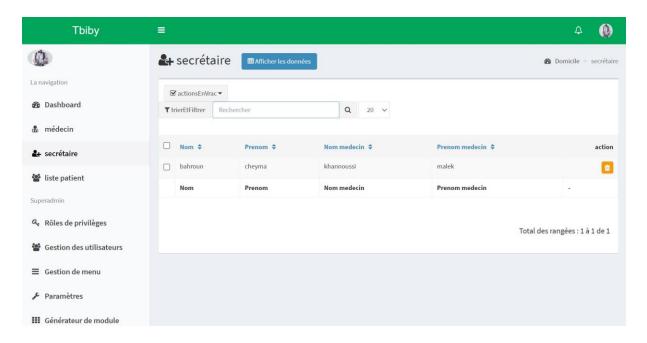


Figure 75 : liste secrétaire

Cette interface permet à l'administrateur de gérer la liste des secrétaires.

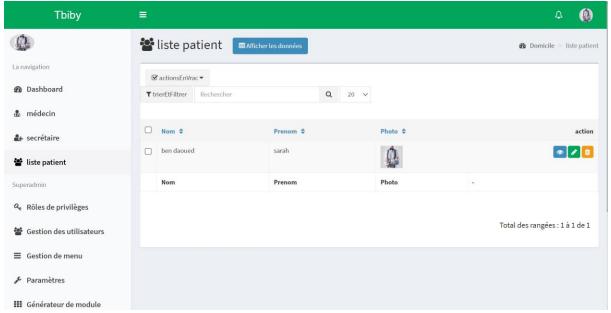


Figure 76 : liste patient

#### VI. Sprint 2: Partie mobile

La partie mobile correspond aux scénarios de l'application web. Pour cette raison on a présenté quelques interfaces.

#### VI.1 Interfaces mobile (release5)

Cette interface décrit la gestion de compte pour la version mobile.

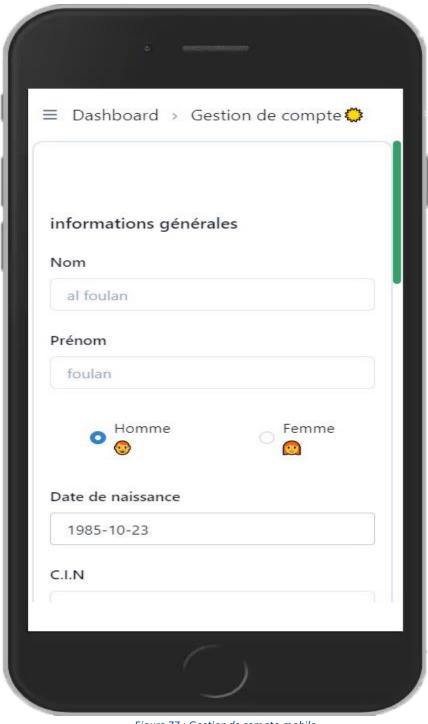


Figure 77 : Gestion de compte mobile

Cette interface présente la liste des médecins pour la version mobile.

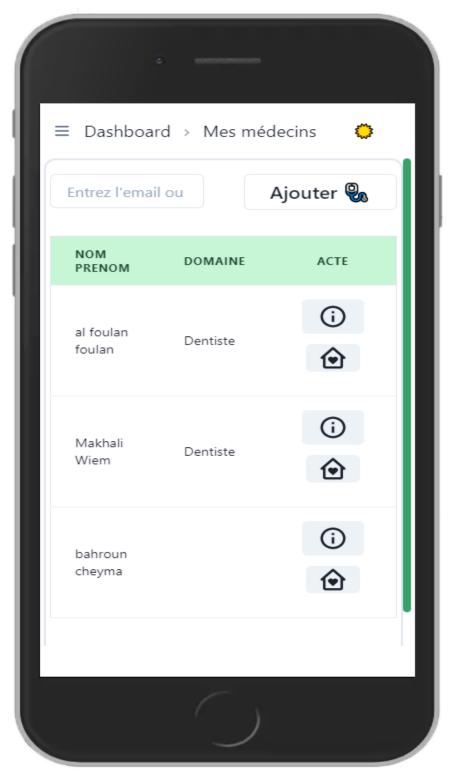


Figure 78 : liste médecin mobile

Cette interface présente la liste des patients pour la version mobile.

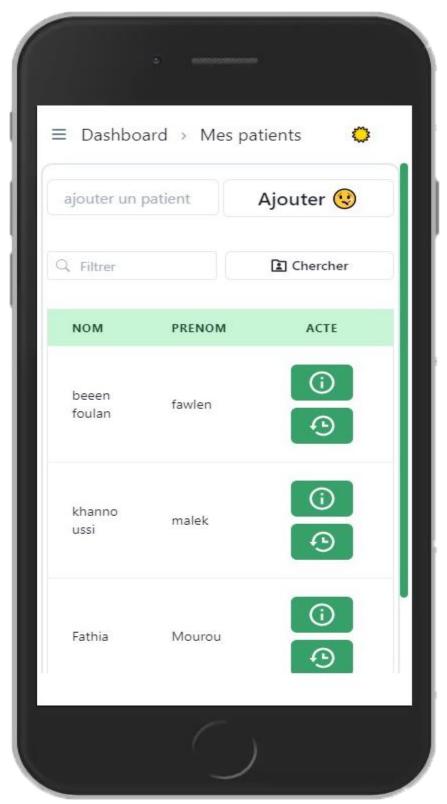


Figure 79 : liste patients mobile

■ Dashboard → Trouver un médecin Domaine Sous Domaine Tous les médecins Q Chercher Q Filtrer NOM DOMAINE ACTE PRENOM al foulan Dentiste foulan Makhali Dentiste Wiem

Cette interface décrit la recherche d'un médecin pour la version mobile.

Figure 80 : recherche médecin mobile

#### VII. Conclusion

Dans ce chapitre, on a présenté le déroulement de travail pour développer la version qui concerne la partie administrative ainsi que la partie mobile.

## Conclusion Générale et perspectives

Ce stage est réalisé dans le cadre d'un projet de fin d'études dans le but d'obtenir un diplôme « licence appliquée en technologie d'informatique spécialité développement système d'information» à l'institut supérieur des études technologiques de Djerba.

Ce stage a été effectué au sein de la société **VINTONET** dans le but d'atteindre un objectif final qui consiste à concevoir et développer une application web et mobile pour la **gestion des cabinets médicaux**.

Dans ce rapport, on a présenté d'une façon cohérente l'ensemble de connaissances obtenues à propos de la société **VINTONET**, son organisation et ses principes de fonctionnement. En plus le reste de ce rapport est consacré pour exposer l'application web qu'on a réalisé en décrivant son concept et les méthodologies adoptés.

Enfin, l'attachement de l'application avec les pharmacies, les laboratoires, les caisses nationale d'assurance-maladie(CNAM), les hôpitaux, était une perspective de ce stage cependant nous n'avons pas eu le temps d'y aboutir.

# Nétographie

http://www.isetjb.rnu.tn/ [1]
https://fr.wikipedia.org/wiki/Gestion_de_projet [2]
https://www.piloter.org/projet/methode/scrum [3]
https://www.manutan.fr/blog/fr-fr/methode-travail/5-avantages-methode-scrum/ [4]
https://openclassrooms.com/fr/courses/2035826-debutez-lanalyse-logicielle-
avecuml/2035851-uml-c-est-quoi [5]
https://www.memoireonline.com/07/09/2392/m Mise-en-place-dune-architecture-3- tiers-
avec-base-de-donnees-centralisee-sous-SQL-SERVER-Cas-du9.html [6]
https://www.lemagit.fr/definition/API-RESTful [7]
https://www.numendo.com/blog/back/laravel-framework-php-complet/ [8]
https://www.nutcache.com/fr/blog/quest-ce-quun-backlog-scrum/ [9]

https://www.abilways-digital.com/magazine/le-petit-lexique-pour-apprendre-parler-agile-et-

<u>scrum</u> [10]

## Annexes

#### FICHE MÉDICALE INDIVIDUELLE

Nom :	Prénom :	Né(e) le	
Adresse : Rue		N°	Bte
Localité :	CP :	Tél :	
N° de sécurité sociale :	N° de mutuelle	:	
Personnes à contacter en cas d'urgence			
Nom – Adresse :			
		Tél :	
Nom – Adresse :			
		Tél :	
Médecin traitant			
Nom – Adresse :			
		Tél :	
Informations médicales confidentielles			
Étes-vous atteint de :  Diabète  Asthme  Affection cardiagu			
	Autre(s)		
Si oui, précisez :			
Fréquence et gravité :			
Êtes-vous allergique à certaines matières, aliments	s, insectes, médicaments	etc. ? Oui/Non	
Si oui, lesquels?			
Avez-vous un régime alimentaire particulier ? Oui/	Non		
Si oui, lequel ?			
Êtes-vous sous traitement médicamenteux ? (Allop	pathie, médecine naturelle	e, pilule contraceptive, etc.) ?	Oui/Non
Si oui, lesquels ?			
Avez-vous été vacciné contre le tétanos ? Oui/Nor	(date du dernier rappel)		
Avez-vous reçu le sérum antitétanique ? Oui/Non (	(en quelle année ?)		
Groupe sanguin :Rhésu			
Remarques éventuelles :			