



**Ministère de l'Enseignement Supérieur, de la Recherche
Scientifique
Direction Générale des Études Technologiques**

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

Code projet	DS106
------------------------	--------------

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)

(Application web pour la gestion d'un cabinet médical)

Elaboré par :

Aya Haddad

Encadré par :

Anis Assas

Effectué à :

Entreprise : Cabinet Hanen

Encadreur : Hanen Tamzini

Année universitaire : 2022/2023

Dédicaces

Avec l'expression de ma reconnaissance, je dédie ce travail à

Mes chers parents,

Qui m'ont soutenu et encouragé durant ces années d'études, qui ne m'ont jamais arrêté, qui ont toujours été là pour moi, me soutenant et me poussant à atteindre mes objectifs, j'espère que ce travail sera l'expression de vos efforts.

Que dieu vous préserve en bonne santé et vous accorde une longue vie.

Mes frères et mes amis,

Vous étiez toujours présents pour m'aider et m'encourager, Sachez que vous serez toujours dans mon cœur, je vous souhaite beaucoup de succès dans votre vie, A tous mes amis qui m'ont aidée et supporté dans les moments difficiles.

Remerciements

C'est avec un immense plaisir que je réserve ces quelques lignes en signe de gratitude et de reconnaissance à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail.

Je voudrais tout d'abord exprimer mes sincères remerciements aux enseignants et administrateurs de l'Institut Supérieur des Etudes Technologiques de Djerba.

Mes remerciements d'adressent autant à **M. Anis Assas**, mon encadrant pédagogique, pour son amabilité, son temps précieux, ses remarques pertinentes et son encouragement continu qui m'a permis de mener à bien ce projet.

Mes remerciements d'adressent autant à **Mme Hanen Tamzini**, mon encadrante professionnelle, pour sa disponibilité et ses conseils précieux qui m'ont guidé dans la réalisation de ce travail.

Je ne saurais oublier de remercier **M. Soufien Ben jema** pour ses conseils et ses compétences.

Sommaire

Introduction Générale	1
Chapitre 1 : Etude Préalable	2
Introduction.....	2
1 Présentation de l'organisme d'accueil	2
2 Etude de l'existant.....	2
2.1 Présentation de l'existant.....	2
2.2 Critique de l'existant	2
2.3 Solution proposée	3
3 Méthodologie de travail	3
3.1 La méthodologie Agile.....	3
3.2 La méthodologie Agile SCRUM.....	3
3.2.1 SCRUM.....	3
4 Environnement technique	5
4.1 Langage de modélisation UML.....	5
4.2 L'outils de modélisation.....	5
4.3 Outils logiciels.....	5
4.4 Langages de développement	6
4.5 Les Framework.....	7
4.6 Architecture adoptée.....	7
Conclusion	8
Chapitre 2 : Analyse et spécification des besoins	9
Introduction.....	9
1 Analyse des besoins	9
1.1 Identification des acteurs.....	9
1.2 Les besoins fonctionnels	9
1.3 Les besoins non fonctionnels	10
2 Diagramme de cas d'utilisation global.....	11
3 Backlog des produits.....	12
3.1 Les histoires du backlog du produit	12
3.2 Diagramme de classes	13
3.3 Planification des sprints.....	14
Conclusion	15
Chapitre 3 : Sprint 1 : Gestion des comptes utilisateurs	16
Introduction.....	16
1 Backlog de sprint 1	16
2 Description fonctionnelle des histoires du sprint 1	16
2.1 Diagramme de classe.....	17
2.2 Diagramme de cas d'utilisation.....	17
2.2.1 Diagramme de cas d'utilisation global du sprint1	17
2.2.2 Diagrammes de cas d'utilisation détaillé du sprint 1	18

2.3	Description textuelle des cas d'utilisations	19
3	Description comportementale des histoires du sprint	22
4	Réalisation du sprint 1.....	26
	Conclusion	31
	Chapitre 4 : Sprint 2 : Gestion des	32
	rendez-vous	32
	Introduction.....	32
1	Backlog de sprint 2	32
2	Description fonctionnelle des histoires du sprint 2.....	33
2.1	Diagramme de classe.....	33
2.2	Diagramme de cas d'utilisation.....	33
2.2.1	Diagramme de cas d'utilisation global du sprint 2	33
2.2.2	Diagramme de cas d'utilisation détaillé de sprint 2	34
2.3	Description textuelle des cas d'utilisations	35
3	Description comportementale des histoires du sprint 2	38
4	Réalisation du sprint 2.....	42
4.1	Espace médecin	42
	Conclusion	45
	Chapitre 5 : Sprint 3 : Gestion des dossiers médicaux.....	46
	Introduction.....	46
1	Backlog du sprint 3	46
2	Description fonctionnelle des histoires du sprint 3	46
2.1	Diagramme de classes	47
2.2	Diagramme de cas d'utilisation.....	47
2.2.1	Diagramme de cas d'utilisation global du sprint 3	47
2.2.2	Diagramme de cas d'utilisation détaillé du sprint 3.....	47
2.3	Description textuelle des cas d'utilisations	48
3	Description comportementale des histoires du sprint	50
4	Réalisation du sprint 3.....	50
	Conclusion	54
	Conclusion générale.....	55
	Webographie	56

Liste des figures

Figure 1: La méthode agile scrum.....	5
Figure 2 : Architecture MVC	8
Figure 3 : JSON Web Token.....	10
Figure 4 : Diagramme de cas d'utilisation globale	11
Figure 5 : Diagramme des classes.....	14
Figure 6:Diagramme de classe du sprint 1	17
Figure 7 : Diagramme de cas d'utilisation global de sprint 1	17
Figure 8 : Diagrammes de cas d'utilisation « Gérer les patients »	18
Figure 9 : Diagrammes de cas d'utilisation détaillé du patient « S'inscrire »	18
Figure 10 : Réinitialiser mot de passe.....	19
Figure 11 : Diagramme de séquence d'authentification	23
Figure 12 : Diagramme de séquence de réinitialiser mot de passe	23
Figure 13 : Diagramme de séquence Gérer les patients : ajouter patient.....	24
Figure 14 : Diagramme de séquence Gérer les patients : modifier patient	24
Figure 15 : Diagramme de séquence Gérer les patients : archiver patient.....	25
Figure 16 : Diagramme de séquence Gérer les patients : chercher patient	25
Figure 17 : Diagramme de séquence s'inscrire	26
Figure 18 : Interface d'accueil	26
Figure 19 : Interface d'authentification	27
Figure 20 : Interface mot de passe oubliée	27
Figure 21 : Interface pour annoncer l'envoi d'e-mail	28
Figure 22 : Interface de la vérification du code	28
Figure 23 : Interface pour la modification de mot de passe.....	28
Figure 24 : Interface pour consulter, archiver et chercher patient	29
Figure 25 : Interface ajouter patient	29
Figure 26 : Interface modifier patient	30
Figure 27 : Interface archive patient	30
Figure 28 : Interface d'inscription patient.....	31
Figure 29 : Diagramme de classe du sprint 2.....	33
Figure 30 : Diagramme de cas d'utilisation global du sprint 2	33
Figure 31 : Diagramme de cas d'utilisation détaillé du médecin : « Gérer l'horaire de travail ».....	34
Figure 32 : Diagramme de cas d'utilisation détaillé du médecin : « Envoyer des e-mails »	34
Figure 33 : Diagramme de cas d'utilisation détaillé du médecin et patient : « Gérer les rendez-vous »	34
Figure 34 : Diagramme de séquence Gérer l'horaire de travail : Ajouter horaire	39
Figure 35 : Diagramme de séquence Gérer l'horaire de travail : Modifier horaire	39
Figure 36 : Diagramme de séquence Gérer les rendez-vous (Ajouter rendez-vous) : médecin.....	40
Figure 37 : Diagramme de séquence Gérer les rendez-vous (Modifier rendez-vous) : médecin.....	40
Figure 38 : Diagramme de séquence Gérer les rendez-vous (Annuler rendez-vous) : médecin.....	41
Figure 39 : Diagramme de séquence « Gérer les rendez-vous (Ajouter rendez-vous) : patient »	41
Figure 40 : Diagramme de séquence « Gérer les rendez-vous (Modifier rendez-vous) : patient »	42
Figure 41 : Diagramme de séquence Gérer les rendez-vous (Annuler rendez-vous) : patient	42
Figure 42 : Interface « Profil médecin » pour gérer l'horaire de travail	43
Figure 43 : Formulaire ajouter horaire de travail	43
Figure 44 : Formulaire modifier horaire de travail	43
Figure 45 : Formulaire ajouter un rendez-vous	44
Figure 46 : Interface des rendez-vous	44
Figure 47 : Interface pour consulter la disponibilité du cabinet et les informations de médecin	45
Figure 48 : Interface des rendez-vous d'un patient	45
Figure 49 : diagramme de classe du sprint 3.....	47
Figure 50 : Diagramme de cas d'utilisation global du sprint 3	47
Figure 51 : Diagramme de cas d'utilisation détaillé du médecin « Gérer les consultations »	48
Figure 52 : Diagramme de cas d'utilisation détaillé du patient « Consulter son dossier médical »	48
Figure 53 : Diagramme de séquence « Gérer les consultations »	50

Figure 54 : Formulaire consultation.....	51
Figure 55 : Formulaire traitements.....	51
Figure 56 : Formulaire fichiers	52
Figure 57 : Ordonnance en HTML	52
Figure 58 : Ordonnance en PDF	52
Figure 59 : Interface pour gérer les consultations.....	53
Figure 60 : Interface de dossier médical d'un patient	53
Figure 61 : Fichier PDF de dossier médical.....	54

Liste des tableaux

Tableau 1: Les acteurs dans SCRUM	4
Tableau 2 : Les histoires du backlog du produit	13
Tableau 3 : Planification des sprints	14
Tableau 4 : Backlog de sprint 1	16
Tableau 5 : Description textuelle du cas d'utilisation « Authentification ».....	19
Tableau 6 : Description textuelle du cas d'utilisation « Réinitialiser mot de passe ».....	20
Tableau 7 : Description textuelle du cas d'utilisation « Gérer les patients ».....	21
Tableau 8 : Description textuelle du cas d'utilisation « S'inscrire »	22
Tableau 9 : Backlog de sprint 2	32
Tableau 10 : Description textuelle de cas d'utilisation détaillé du médecin : « Gérer l'horaire de travail ».....	35
Tableau 11 : Description textuelle du cas d'utilisation « Gérer les rendez-vous » de médecin	37
Tableau 12 : Description textuelle du cas d'utilisation « Gérer les rendez-vous » de patient.....	38
Tableau 13 : Backlog du sprint 3	46
Tableau 14 : Description textuelle du cas d'utilisation « Gérer les consultations ».....	49
Tableau 15 : Description textuelle du cas d'utilisation « Gérer son dossier médical ».....	50

Introduction Générale

Le monde autour de nous évolue d'une manière très rapide et vaste, et en fait, cette évolution croît d'une manière exponentielle et simultanément à la croissance de l'information et de la technologie.

Dans notre vie quotidienne, tout est devenue lié à la technologie car elle nous procure un confort dans divers domaines, particulièrement le domaine de la santé.

L'approche médicale est basée sur l'observation du patient. Les données médicales ont été recueillies sous forme d'articles médicaux et de dossiers épidémiologiques et administratifs.

La mémoire du médecin était insuffisante pour enregistrer les données du patient et servir la pratique médicale. Avec la prévalence des influences environnementales, de nos jours, une bonne tenue des dossiers nécessite des ressources informatiques. L'automatisation du système d'information consiste à structurer et gérer un ensemble de données qui vise à l'organiser et à obtenir des résultats rapides. Cependant, elle suscite de nombreuses interrogations de la part des usagers du système et des professionnels de santé, qu'il s'agisse du respect de l'éthique médicale et des droits humains, du transfert des pratiques professionnelles et de la confiance entre patient et médecin.

Pour surmonter les obstacles rencontrés par les médecins et les patients, nous nous proposons dans ce projet de développer une application web qui peut gérer le cabinet des médecins afin d'améliorer la qualité des soins de santé délivrés aux patients, augmenter le rendement du personnel médical et faciliter les tâches administratives au sein du cabinet.

Les différentes étapes de réalisation de ce projet se résument dans le présent rapport comme suit :

- Un premier chapitre : Etude préalable.
- Un deuxième chapitre : Analyse et spécification des besoins.
- Un troisième chapitre : Sprint 1 : Gestion des comptes utilisateurs.
- Un quatrième chapitre : Sprint 2 : Gestion des rendez-vous.
- Un cinquième chapitre : Sprint 3 : Gestion des dossier médicaux.

Chapitre 1 : Etude Préalable

Introduction

Ce premier chapitre sera consacré à une présentation du projet effectué durant mon stage de fin d'étude, une présentation de l'organisme et l'analyse de l'existant afin de présenter notre solution proposée.

1 Présentation de l'organisme d'accueil

Mon stage s'est déroulé au sein de la cabinet médical du Docteur Hanen Tamzini situé à Sedouikech Djerba. C'est un lieu qui pratique la médecine générale en examinant les patients et en établissant un diagnostic en fonction de ses observations médicales et de ses symptômes.

2 Etude de l'existant

L'étude de l'existant nous permet de mettre le projet dans ce contexte tout en envoyant de décrire la procédure du travail actuelle au niveau du cabinet médical.

2.1 Présentation de l'existant

Le secteur de la santé est très sensible, le médecin porte donc une grande responsabilité vis-à-vis de son patient et de ses informations personnelles. Un médecin doit toujours être disponible pour les urgences, pour gérer les statistiques de son cabinet, les rendez-vous, etc.... et pour cela il est obligé de passer toute sa journée dans son cabinet.

Le cabinet de Hanen est un lieu qui pratique la médecine générale ou Mme. Hanen gère manuellement les dossiers médicaux de ses patients en les archivant dans des dossiers, organise aussi les rendez-vous de ses patients et élabore les statistiques sur les rémunérations des visites et des dépenses.

Nous trouvons une secrétaire dans la salle d'attente, son rôle consiste à recevoir les patients et à consigner leurs noms dans un carnet.

2.2 Critique de l'existant

Le domaine des technologies de l'information impose sa présence dans tous les secteurs afin que toutes les données soient conservées au même endroit et aussi pour ajouter le sceau de la protection et de la confidentialité. Mais malheureusement pas dans le secteur de la santé où le médecin utilise toujours des fichiers pour conserver les informations personnelles de ses patients et cela peut poser des problèmes de protection des informations car elles peuvent être volées, perdues, voir même les dossiers des patients mélangés, etc.

Les informations médicales sont très sensibles et personne ne peut les voir sauf le médecin, c'est pourquoi le cabinet médical doit disposer d'un système de haute sécurité pour protéger ces informations.

Le médecin doit toujours être au courant des statistiques de son cabinet mais peut ne pas avoir assez de temps pour calculer manuellement le coût des visites et des dépenses.

2.3 Solution proposée

Afin de gérer le cabinet médical de manière sécurisée et fiable, nous devons intégrer le domaine informatique dans le secteur de la santé, la création d'une application Web est donc la meilleure solution.

L'application est un vrai plus pour les médecins car elle va organiser et faciliter leur travail, le médecin ou son assistant n'a qu'à saisir les informations.

3 Méthodologie de travail

Le choix d'une méthodologie est important pour parvenir à une bonne réalisation du projet en passant par plusieurs étapes, de la planification à l'exécution. Y en a plusieurs méthodologies telle que la méthode agile, DevOps, Lean, etc... et dans mon cas j'ai choisi agile comme méthodologie de développement.

3.1 La méthodologie Agile

La méthode agile est une approche du développement logiciel dont l'objectif est de distribuer en continu des logiciels opérationnels créés sur la base d'itérations rapides.

Concrètement, les méthodes de développement de logiciels agile sont axées sur la distribution rapide de petites parties de logiciels opérationnels pour améliorer la satisfaction client. Elles se basent sur une approche qui encourage l'adaptation et le travail d'équipe pour favoriser l'amélioration continue. En général, le développement logiciel agile implique la réunion régulière et en personne de petites équipes autonomes de développeurs logiciels et de représentants métier tout au long du cycle de vie. La méthode agile privilégie une approche simplifiée plutôt que la documentation logicielle et favorise les changements à toutes les étapes du cycle de vie, au lieu d'y faire obstacle.[1]

3.2 La méthodologie Agile SCRUM

Après la sélection de la méthodologie Agile comme étant la méthodologie de conduite du projet nous avons choisi SCRUM comme méthode agile.

3.2.1 SCRUM

➤ Définition

Scrum est une des méthodes de gestion de projet Agile. En tant que telle, son objectif est d'améliorer la productivité des équipes agiles même à distance, tout en permettant une optimisation du produit grâce à des feedbacks réguliers avec les utilisateurs finaux.

La méthode Scrum tire son nom du monde du rugby (scrum = mêlée), car les équipes agiles qui utilisent Scrum se réunissent le plus souvent possible afin de vérifier que le projet avance correctement, toujours

prêts à réorienter ce dernier au fil de son avancement. C'est donc une approche dynamique et participative de la conduite du projet, qui garantit pour le client le juste équilibre entre l'investissement prévu et le produit finalement livré.[2]

➤ Les principes de SCRUM

La méthodologie SCRUM est basée sur 3 principes qui sont au même temps les principes de la culture agile :

- **Transparence** : Scrum met l'accent sur le fait d'avoir un langage commun.
- **Inspection** : À intervalle régulier, Scrum propose de faire le point sur les différents artefacts produits, afin de détecter toute variation indésirable.
- **Adaptation** : Si une dérive est constatée pendant l'inspection, le processus doit alors être adapté.

➤ Les acteurs dans SCRUM

Scrum se compose de trois rôles avec des responsabilités différentes afin de bien organiser le travail.

Les trois rôles sont :

- **Product Owner** : Le Product Owner est responsable de maximiser la valeur du produit résultant du travail de l'équipe de développement. La façon de jouer ce rôle peut varier grandement selon les organisations, les équipes Scrum et les individus.
- **L'équipe de développement** : L'équipe de développement se compose de professionnels qui fournissent un incrément « Fini » potentiellement publiable (Releasable) à la fin de chaque Sprint. Un incrément « Fini » est requis à la revue de sprint. Seuls les membres de l'équipe de développement créent l'incrément.
- **Le Scrum master** : Le Scrum Master est chargé de promouvoir et supporter Scrum tel que défini dans le Guide Scrum. Les Scrum Masters remplissent leur rôle en aidant tout le monde à comprendre la théorie, les pratiques, les règles et les valeurs de Scrum.

Rôle Scrum	Responsable(s)
Product Owner	Mme. Hanen Tamzini
Développeur	Aya Haddad
Scrum master	M. Anis Assas

Tableau 1: Les acteurs dans SCRUM

➤ Le modèle Scrum

C'est une figure qui peut donner une description générale sur la méthode agile Scrum.

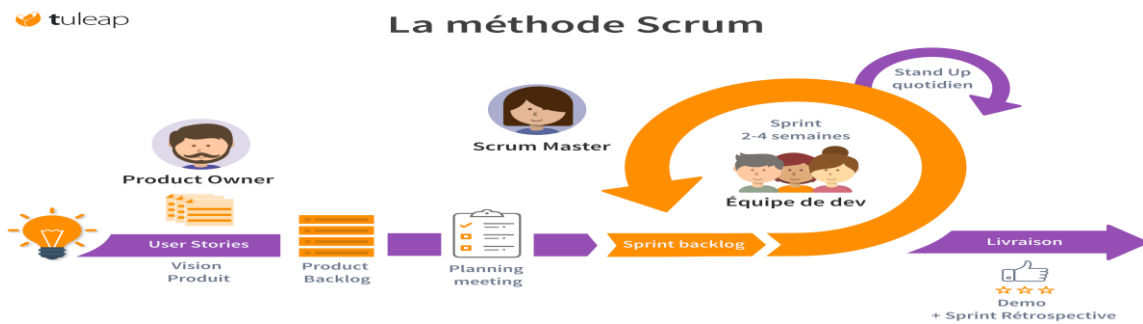


Figure 1: La méthode agile scrum

4 Environnement technique

4.1 Langage de modélisation UML

Le Langage de Modélisation Unifiée, de l'anglais Unified *Modeling* Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet.[18]



4.2 L'outils de modélisation

Star UML : StarUML est un logiciel de modélisation UML, qui a été « cédé comme open source » par son éditeur, en fait tous les diagrammes UML peuvent être générés facilement. [8]



4.3 Outils logiciels

Visual Studio Code : est un éditeur de code open-source développé par Microsoft supportant un très grand nombre de langages grâce à des extensions. Il supporte l'autocomplétions, la coloration syntaxique, le débogage, et les commandes git. [3]



Spring tools 4 for eclipse : STS est un environnement de développement basé sur Eclipse qui est personnalisé pour le développement d'applications Spring.

Il fournit un environnement prêt à l'emploi pour implémenter, déboguer, exécuter et déployer vos applications. Il inclut également l'intégration pour Pivotal tc Server, Pivotal Cloud Foundry, Git, Maven et AspectJ. STS est construit comme un ajout aux dernières versions d'Eclipse.[4]



Laragon : Laragon est un environnement de développement web qui regroupe les serveurs suivants :

- Apache HTTP Server : Serveur HTTP, tournant par défaut sur le port 80



- **MySQL** : Serveur de base de données relationnelles SQL, tournant par défaut sur le port 3306.

[5]

GitHub : est un site web et un service de cloud qui aide les développeurs à stocker et à gérer leur code, ainsi qu'à suivre et contrôler les modifications qui lui sont apportées. [6]



Git : Git est de loin le système de contrôle de version le plus largement utilisé aujourd'hui. Git est un projet open source avancé, qui est activement maintenu.[7]



Postman : Postman est une application permettant de tester des API, elle permet de lancer une série de requêtes http et de tester chacune d'entre elles. [9]



4.4 Langages de développement

HTML5 : le HTML5, pour HyperText Markup Language 5, est une version du célèbre format HTML utilisé pour concevoir les sites Internet. Celui-ci se résume à un langage de balisage qui sert à l'écriture de l'hypertexte indispensable à la mise en forme d'une page Web.[10]



CSS3 : Les feuilles de style (en anglais "Cascading Style Sheets", abrégé CSS) sont un langage qui permet de gérer la présentation d'une page Web. Le langage CSS est une recommandation du World Wide Web Consortium (W3C), au même titre que HTML ou XML.[11]



TypeScript : TypeScript est un langage pour JavaScript à l'échelle de l'application. TypeScript ajoute des types facultatifs à JavaScript qui prennent en charge des outils pour les applications JavaScript à grande échelle pour n'importe quel navigateur, pour n'importe quel hôte, sur n'importe quel système d'exploitation.[12]



Java : Java est une technique informatique développée initialement par Sun Microsystems puis acquise par Oracle à la suite du rachat de l'entreprise. Défini à l'origine comme un langage de programmation, Java a évolué pour devenir un ensemble cohérent d'éléments techniques et non techniques.[13]



MySQL : MySQL un Système de Gestion de Bases de Données Relationnelles (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde.[19]



JSON : Le JavaScript Object Notation (JSON) est le format d'échange de données qui répond à ce besoin. Le format de données JSON est devenu populaire auprès des développeurs, car il se présente sous la forme de texte léger et lisible en plus de nécessiter moins de codage et de rendre les processus plus rapides.[14]



4.5 Les Framework

Bootstrap : Bootstrap est un Framework développé par l'équipe du réseau social Twitter. Proposé en open source (sous licence MIT), ce Framework utilisant les langages HTML, CSS et JavaScript fournit aux développeurs des outils pour créer un site facilement. Ce Framework est pensé pour développer des sites avec un design responsif, qui s'adapte à tout type d'écran.[15]



Spring Boot : Le Spring Framework est très largement utilisé dans la communauté Java. Il permet d'accélérer le développement d'applications d'entreprise (notamment le développement d'applications Web et d'API Web).[16]



Angular : Angular est un Framework open source JavaScript développé par Google. Ce framework est utilisé pour développer des applications web et mobile. Avec cette technologie, on réalise des interfaces de type monopage ou "one page" qui fonctionnent sans rechargement de la page web.[17]



4.6 Architecture adoptée

✱ **Architecture MVC :** Modèle - Vue - Contrôleur, en gros quand on développe avec le MVC on segmente son code en trois parties ou couches, chaque couche ayant une fonction bien précise.

- **La couche vue :** C'est la partie de mon code qui s'occupera de la présentation des données à l'utilisateur, elle retrouve une vue des données venant du modèle, en d'autres termes c'est elle qui est responsable de produire les interfaces de présentation de votre application à partir des informations qu'elle dispose (page HTML par exemple). Cependant, elle n'est pas seulement limitée au HTML ou à la représentation en texte des données, elle peut aussi être utilisée pour offrir une grande variété de formats en fonction de vos besoins. [21]
- **La couche Contrôleur :** C'est la couche chargée de router les informations, elle va décider qui va récupérer l'information et la traiter. Elle gère les requêtes des utilisateurs et retourne une réponse avec l'aide de la couche Modèle et Vue. [21]
- **La couche Modèle :** C'est la partie de votre code qui exécute la logique métier de votre application. Ceci signifie qu'elle est responsable de récupérer les données, de les

convertir selon les concepts de la logique de votre application tels que le traitement, la validation, l'association et tout autre tâche concernant la manipulation des données. Elle est également responsable de l'interaction avec la base de données, elle sait en quelque sorte comment se connecter à une base de données et d'exécuter les requêtes (CREATE, READ, UPDATE, DELETE) sur une base de données. [21]

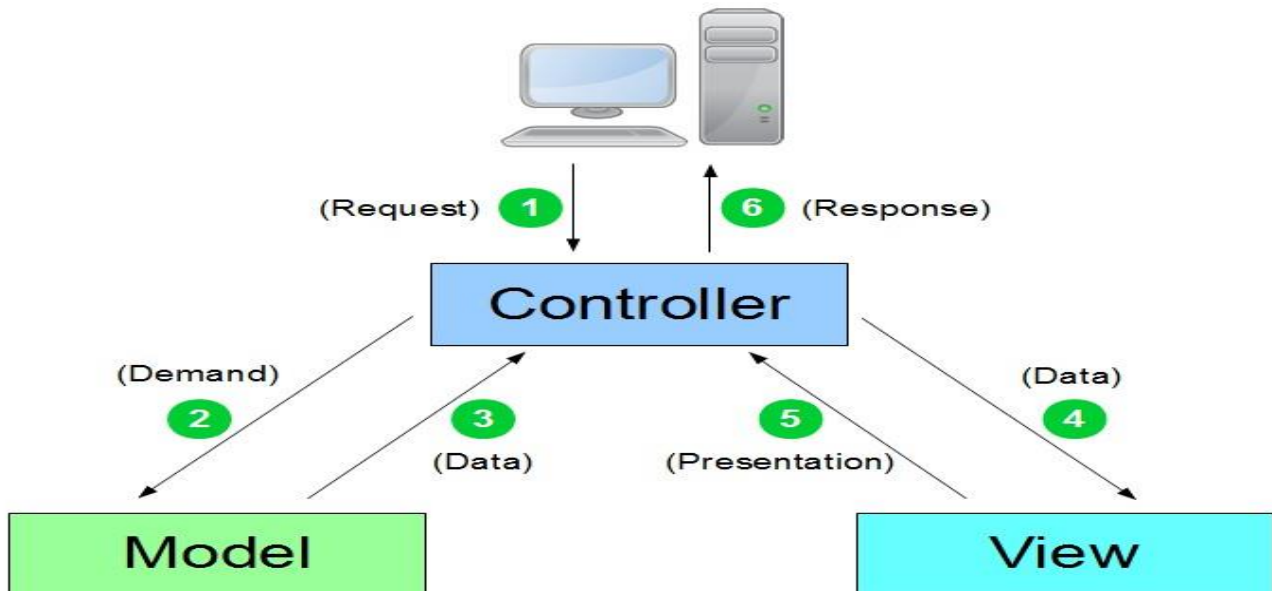


Figure 2 : Architecture MVC

Conclusion

Dans ce chapitre, nous avons progressivement présenté le cadre général de travail. Après avoir fixé l'idée générale de notre projet en faisant une étude de l'existant et en choisissant la méthodologie de gestion de projet et en finissant par la présentation des choix techniques. Le prochain chapitre sera consacré à une analyse et spécification des besoins du projet aussi qu'une conception détaillée.

Chapitre 2 : Analyse et spécification des besoins

Introduction

Ce chapitre vise à donner une idée bien détaillée de notre projet afin de nous mettre dans le cadre. En d'autres termes, ce chapitre devrait répondre à la question comment faire et puisqu'on a utilisé scrum comme méthodologie, ce chapitre sera notre premier sprint « sprint zéro ». Au niveau de ce chapitre nous commençons, d'abord par définir l'analyse des besoins. Puis, nous allons présenter le backlog du produit en mentionnant ses éléments et la planification des sprints.

1 Analyse des besoins

1.1 Identification des acteurs

L'identification des acteurs permet en premier lieu de délimiter le système et en second lieu de comprendre le rôle de chaque acteur. Dans notre application nous disposons de deux acteurs qui sont :

- **Médecin** : Qui joue le rôle d'admin. Il gère son cabinet.
- **Patient** : Son rôle est de consulter son dossier médical

1.2 Les besoins fonctionnels

Avant la création de l'application il faut définir les besoins fonctionnels

❖ L'application devrait offrir au médecin :

- **Inscription** : Un médecin doit être inscrit pour avoir le droit d'utiliser notre application. Notre plateforme met à disposition un formulaire pour saisir ses informations avec un mot de passe pour sécuriser son compte, et après son inscription il peut accéder à son espace administratif.
- **Gestion des fiches des patients** : Le médecin pourra créer des fiches numériques contenant des informations sur le patient. A travers le formulaire de la fiche, le compte du patient est valide afin qu'il puisse consulter son dossier médical quand et où il le souhaite.
- **Editer les ordonnances complètes** : Le médecin doit éditer des ordonnances numériques.
- **Gestion complète des dossiers patients** : Le dossier médical contient les informations personnelles d'un patient tel que ses rendez-vous, ses consultations, ses traitements
- **Gestion des e-mails** : Le médecin peut Rappel des rendez-vous par des e-mails.

- **Gestion des facturations :** Toutes les factures des dépenses du cabinet seront stockées dans l'espace du médecin.
 - **Gestion des statistiques du cabinet :** Elaboration des statistiques liées aux rémunérations des visites et des dépenses.
- ❖ L'application offre au patient :
- **Inscription :** Un patient doit être inscrit pour avoir le droit d'utiliser notre application. Notre plateforme met à disposition un formulaire pour saisir ses informations avec un mot de passe pour sécuriser son compte, et après son inscription il peut accéder à son espace administratif.
 - **Prise de rendez- vous :** : Un patient peut prendre des rendez-vous.
 - **Consultation des dossiers médicaux :** Un patient peut consulter son dossier médical.

1.3 Les besoins non fonctionnels

Les besoins non fonctionnels décrivent toutes les contraintes auxquelles est soumis le système pour sa réalisation et son bon fonctionnement. Parmi les besoins non fonctionnels :

- **La rapidité :** La manipulation et le traitement des données doit être rapide afin d'augmenter le rendement du travail
- **La sécurité :** Étant donné que les informations personnelles du patient sont très sensibles, personne ne peut les voir à l'exception du médecin. L'application doit donc garantir une confidentialité totale des données et ce grâce à un mécanisme d'authentification par token. Les « JSON Web Token » ou JWT sont des jetons générés par un serveur lors de l'authentification d'un utilisateur sur une application Web, et qui sont ensuite transmis au client. Ils seront renvoyés avec chaque requête HTTP au serveur, ce qui lui permettra d'identifier l'utilisateur.[20]

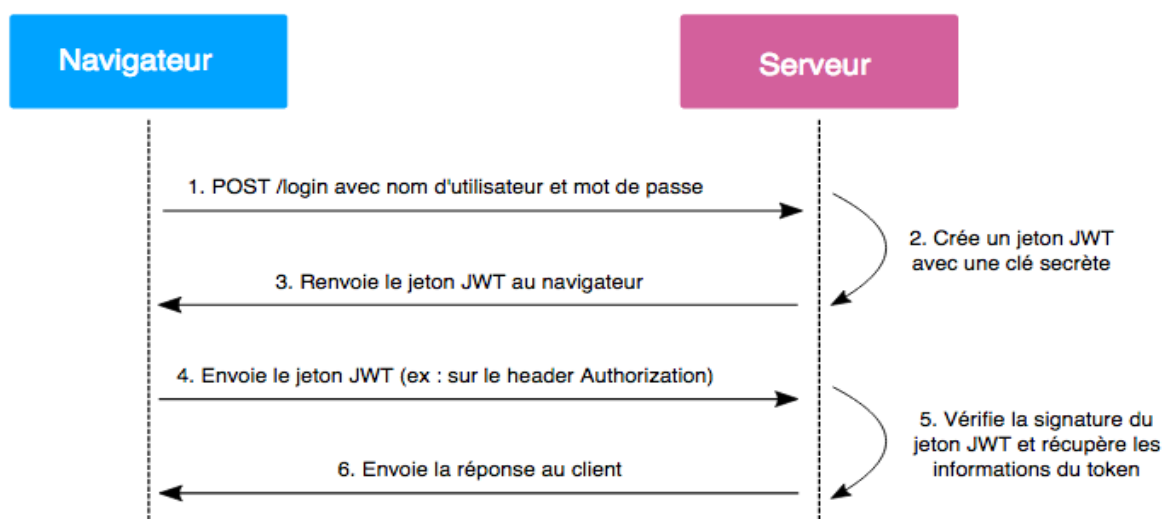


Figure 3 : JSON Web Token

- **La confidentialité** : La confidentialité des applications est importante pour empêcher tout accès non autorisé à des informations sensibles.
- **La disponibilité** : Lorsque n'importe quel utilisateur désire consulter l'application, elle doit être disponible.
- **L'organisation** : L'application doit être bien organisée afin qu'il soit facile de travailler avec.

2 Diagramme de cas d'utilisation global

Les diagrammes de cas d'utilisation globaux dans UML sont utilisés pour donner une vision du comportement fonctionnel d'une application représentée dans les cas d'utilisation et décrire les relations entre les acteurs (Médecin, Patient).

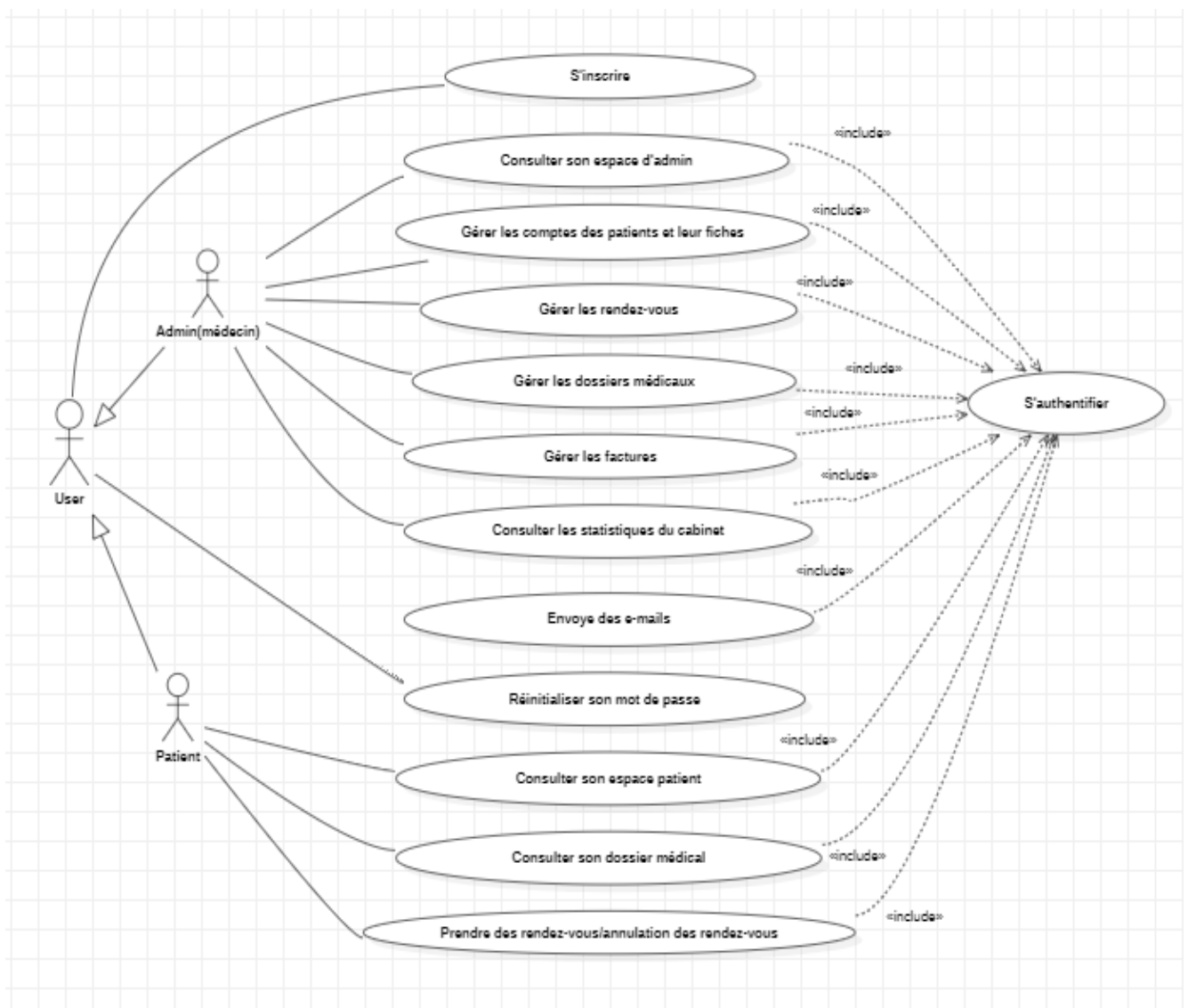


Figure 4 : Diagramme de cas d'utilisation globale

3 Backlog des produits

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.

3.1 Les histoires du backlog du produit

Dans cette partie, nous présenterons chaque « sprint » avec ses différentes « fonctionnalités » en indiquant les « user story » chacune avec son « ID » et son « Estimation ».

Sprint	Fonctionnalités	ID	User story	Estimation
Gestion des comptes	S'inscrire	1	En tant qu'utilisateur je veux s'inscrire.	2 jours
	S'authentifier	1	En tant qu'utilisateur je veux se connecter.	1 semaine
	Gérer les patients	1	En tant que médecin je peux créer, modifier, consulter et archiver un compte pour mon patient.	5 jours
	Modifier mot de passe	1	En tant qu'utilisateur je peux réinitialiser mon mot de passe.	1 semaine
Gestion des rendez-vous	Consulter disponibilité	1	En tant que médecin je peux gérer les horaires de la disponibilité de mon cabinet.	2 jours
		2	En tant que patient je peux voir les coordonnées de mon médecin et la disponibilité de son cabinet.	1 jour
	Réservation	1	En tant qu'un patient je peux réserver un rendez-vous chez un médecin.	4 jours
		2	En tant qu'un médecin je peux réserver un rendez-vous pour mon patient.	
		3	En tant que patient je peux consulter mes rendez-vous.	2 semaines

		4	En tant que médecin je peux consulter les rendez-vous.	
		5	En tant que patient je peux modifier/annuler un rendez-vous.	
		6	En tant que patient je peux supprimer un rendez-vous.	
	Mailing	1	En tant que Médecin je peux rappeler mes patients de leur rendez-vous par des e-mails.	3 jours
Gestion des dossiers médicaux	Gérer les consultations	1	En tant que médecin je peux ajouter/modifier les consultations au dossier médical de mon patient.	2 semaines
		2	En tant que médecin je peux ajouter/modifier des traitements dans la consultation de mon patient.	
		3	En tant que médecin je peux ajouter/modifier des fichier pendant la consultation de mon patient.	
	Gérer les dossiers médicaux	1	En tant que médecin je peux consulter les dossiers médicaux de mes patients et les générer en fichier PDF.	5 jours
		2	En tant que médecin je peux archiver un dossier médical et le patient.	1 semaine
		3	En tant que patient je peux consulter mon dossier médical et le générer en fichier PDF.	2 jours

Tableau 2 : Les histoires du backlog du produit

3.2 Diagramme de classes

Le diagramme de classes permet de visualiser la structure statique d'un système orienté objet, en montrant les classes, leurs attributs, leurs méthodes et les relations entre elles. Cela facilite la

compréhension du système, l'identification des classes nécessaires et les interactions entre les différentes parties du système.

La figure ci-dessous représente le diagramme de classes de notre projet.

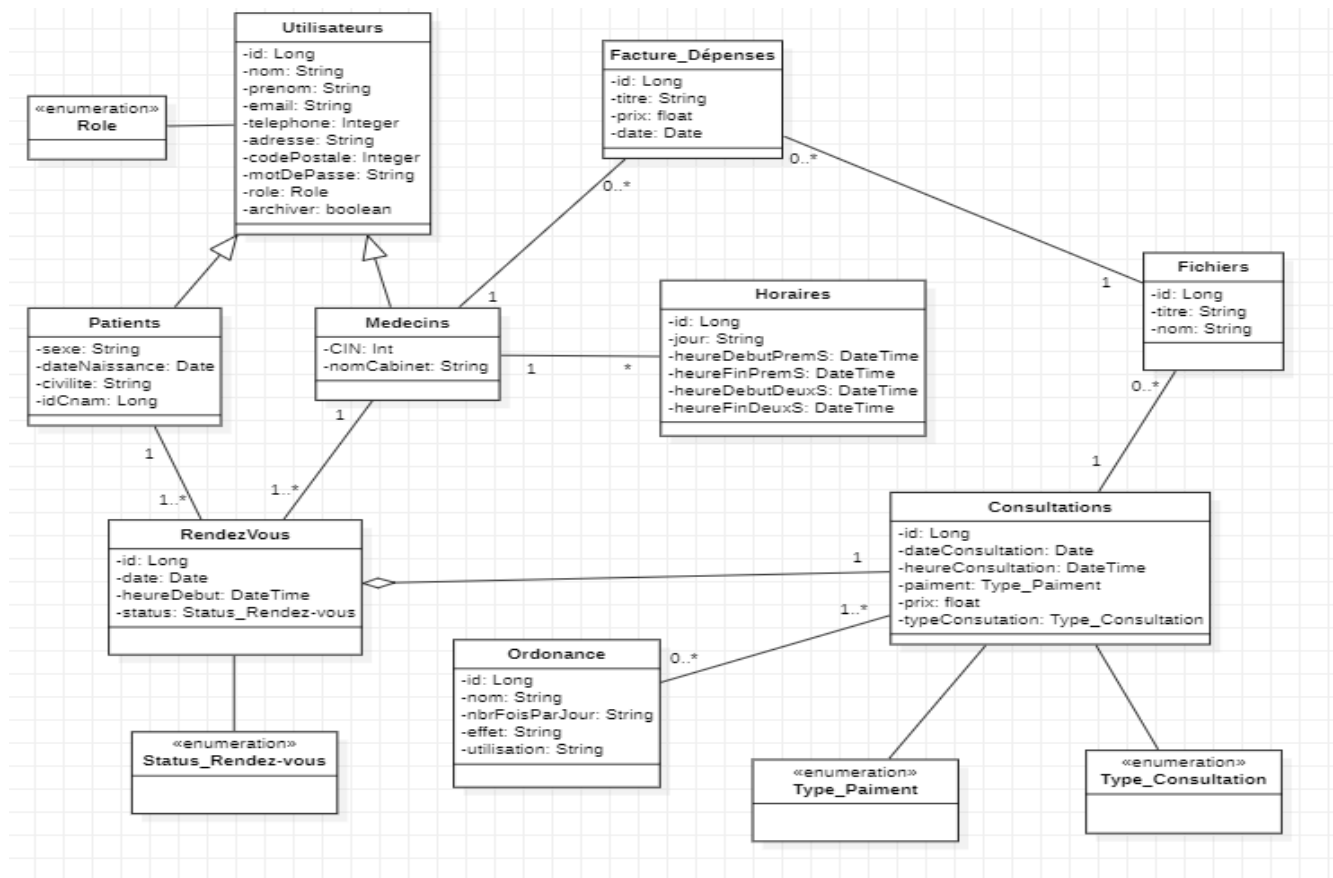


Figure 5 : Diagramme des classes

3.3 Planification des sprints

Nous avons choisi de diviser le projet en 4 sprints. Dans chaque sprint sera développé un module de l'application. La durée et l'intitulé du module de chaque sprint sont indiqués dans le tableau :

Sprint	Module	Durée
Sprint 1	Gestion des comptes utilisateurs	3 semaines
Sprint 2	Gestion des rendez-vous	4 semaines
Sprint 3	Gestion des dossiers médicaux	4 semaines

Tableau 3 : Planification des sprints

Conclusion

Dans ce chapitre nous avons défini les acteurs et listé les besoins fonctionnels et non fonctionnels de ce projet. Ensuite, nous avons présenté les éléments du backlog du produit, puis, la planification des sprints. Dans ce qui suit, nous détaillerons les étapes chaque sprint en commençant par le sprint 1.

Chapitre 3 : Sprint 1 : Gestion des comptes utilisateurs

Introduction

Dans ce chapitre nous allons présenter le déroulement pour développer le module des comptes. Nous allons commencer par la présentation du backlog du notre premier sprint. Ensuite nous allons passer à la description fonctionnelle en présentant les diagrammes de cas d'utilisation et le diagramme des classes ainsi que la description comportementale à l'aide des diagrammes de séquences. Enfin nous finirons par la présentation des interfaces de réalisation.

1 Backlog de sprint 1

Le tableau ci-dessous consiste à présenter le Product backlog pour le sprint 1 « Gestion des comptes utilisateurs »

Product Backlog (Gestion des comptes utilisateurs)				
Fonctionnalité	ID	User story	Tâches	Estimation
S'inscrire	1	En tant qu'utilisateur, je souhaite m'inscrire pour avoir un compte.	- Conception. - Implémentation. - Test.	2 jours
S'authentifier	2	En tant qu'utilisateur je peux m'authentifier.	- Conception. - Implémentation. - Test.	1 semaine
Gérer les patients	3	En tant que médecin je veux ajouter, consulter, modifier et archiver un patient.	- Conception. - Implémentation. - Test.	1 jour
Modifier mot de passe	4	En tant qu'utilisateur je peux réinitialiser mon mot de passe.	- Conception. - Implémentation. - Test.	1 semaine

Tableau 4 : Backlog de sprint 1

2 Description fonctionnelle des histoires du sprint 1

Pour donner une vision globale du comportement fonctionnel du système, on utilise les diagrammes de cas d'utilisation. Dans cette partie, nous commençons par présenter le diagramme de classes et le diagramme de cas d'utilisation global de notre premier sprint, par la suite, nous le détaillons selon nos acteurs.

2.1 Diagramme de classe

La figure ci-dessous représente le diagramme de classe du sprint 1.

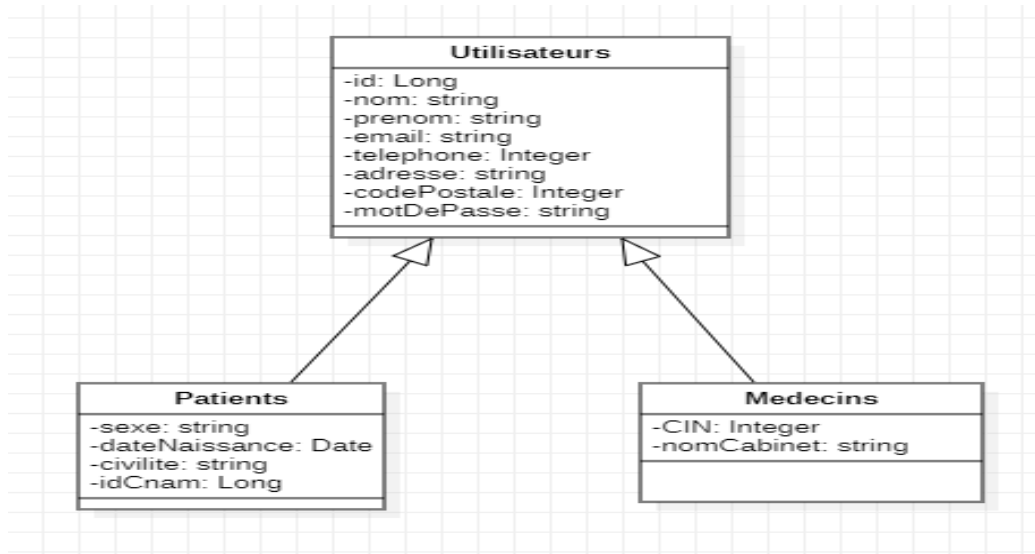


Figure 6: Diagramme de classe du sprint 1

2.2 Diagramme de cas d'utilisation

2.2.1 Diagramme de cas d'utilisation global du sprint1

La figure ci-dessous représente les 4 cas d'utilisations qui sont notre centre d'intérêt tout au long de notre premier sprint.

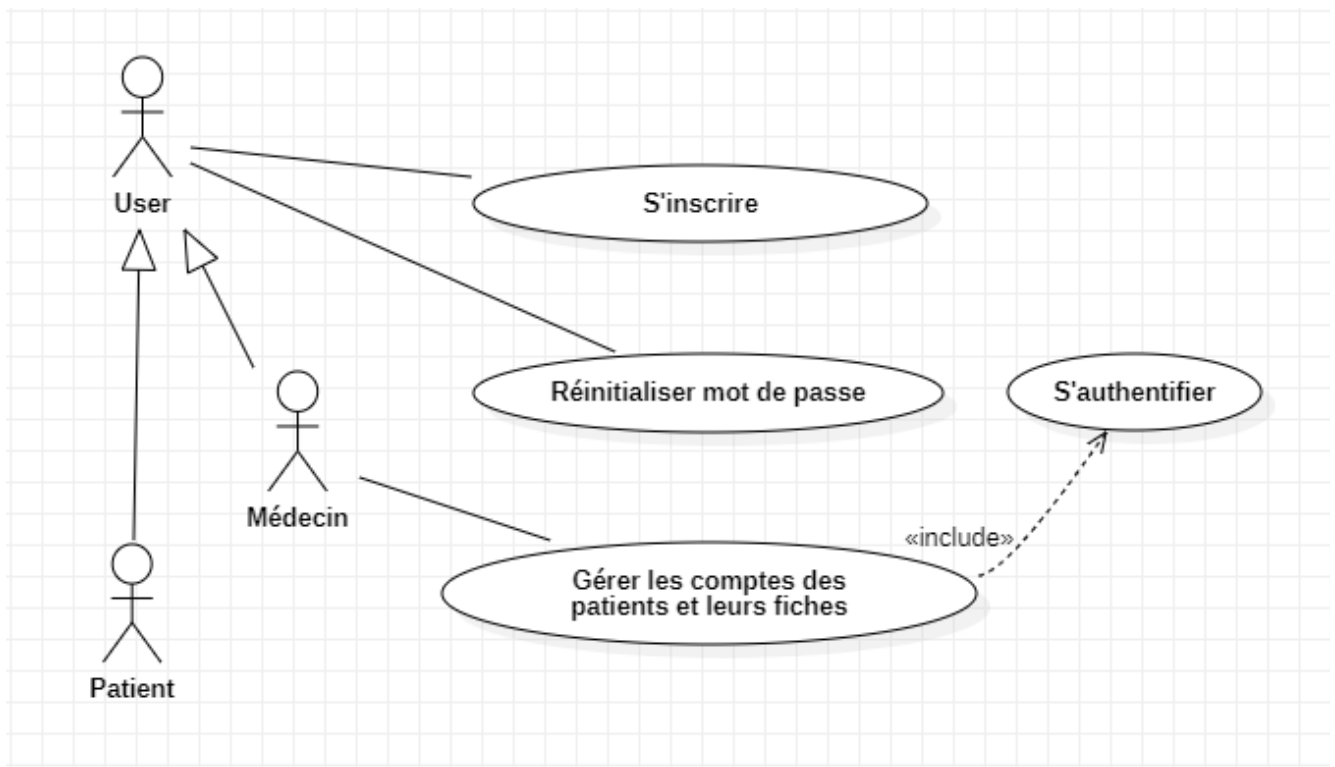


Figure 7 : Diagramme de cas d'utilisation global de sprint 1

2.2.2 Diagrammes de cas d'utilisation détaillé du sprint 1

Après avoir présenté le diagramme de cas d'utilisation global de notre premier sprint, on va expliquer les fonctionnalités de chaque cas d'utilisation.

○ Diagrammes de cas d'utilisation détaillé du médecin « Gérer les patients » :

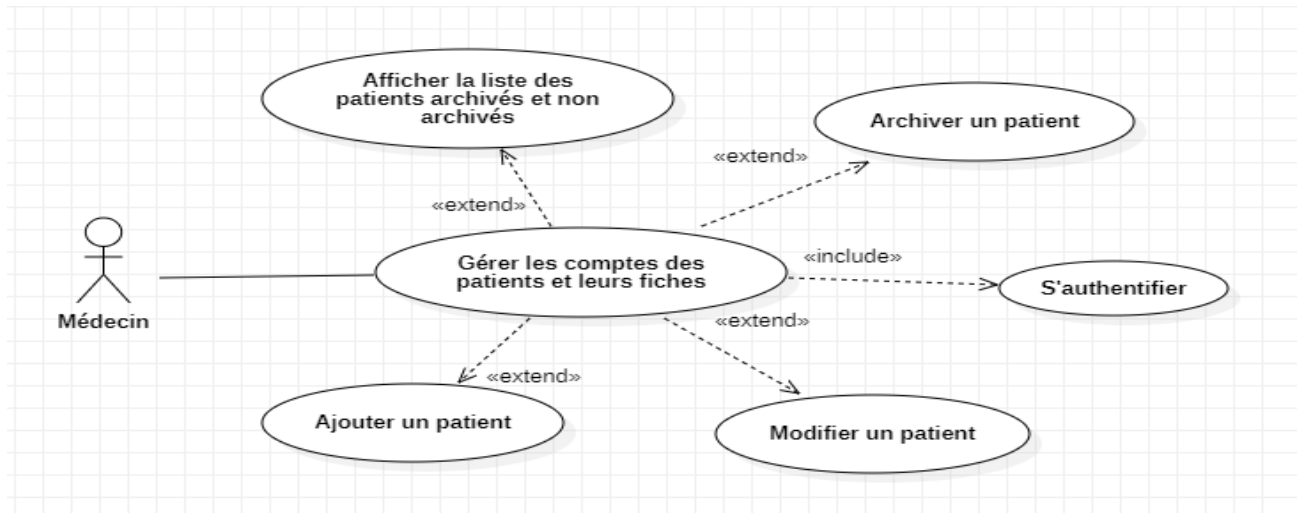


Figure 8 : Diagrammes de cas d'utilisation « Gérer les patients »

○ Diagrammes de cas d'utilisation détaillé du patient « S'inscrire » :

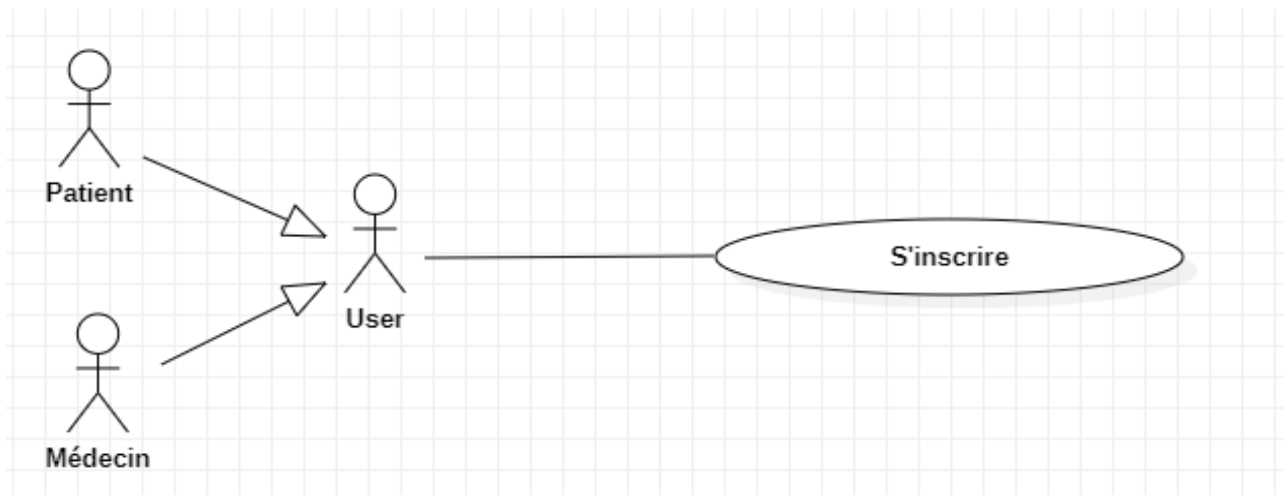


Figure 9 : Diagrammes de cas d'utilisation détaillé du patient « S'inscrire »

○ Diagrammes de cas d'utilisation détaillé de médecin et patient

« Réinitialiser mot de passe » :

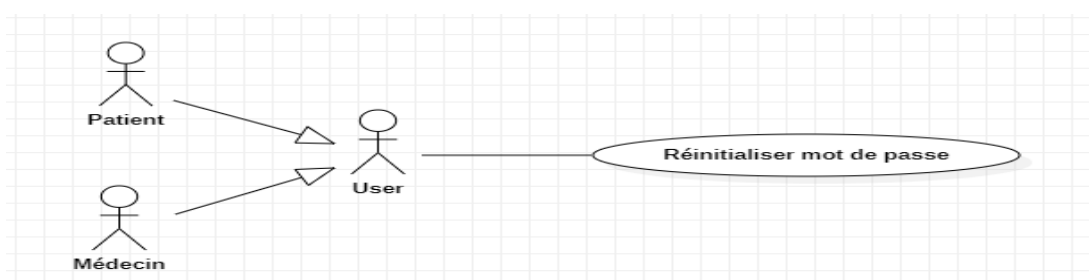


Figure 10 : Réinitialiser mot de passe

2.3 Description textuelle des cas d'utilisations

Dans cette partie nous décrivons textuellement les cas d'utilisation principaux qui apparaissent dans le diagramme de cas d'utilisation global.

➤ Le tableau suivant décrit textuellement le cas d'utilisation S'authentifier

Cas d'utilisation : S'authentifier	
Acteur(s)	User
Objectif	Permettre aux utilisateurs de se connecter afin de pouvoir utiliser les fonctionnalités de l'application.
Pré condition	<ul style="list-style-type: none"> • L'utilisateur doit avoir un compte. • L'email et le mot de passe saisis doivent être corrects.
Post condition	Utilisateur authentifié.
Scénario nominal	<ol style="list-style-type: none"> 1. Afficher l'interface de l'authentification. 2. Saisir le login et le mot de passe. 3. Valider. 4. Le Système vérifie les coordonnées. 5. Le système affiche l'interface propre à l'utilisateur.
Scénario alternatif	<ol style="list-style-type: none"> 1. Saisir le login et le mot de passe. 2. Valider 3. Affichage d'un message d'erreur et retour à l'étape 2 du scénario nominal. 4. Accès interdit.

Tableau 5 : Description textuelle du cas d'utilisation « Authentification »

- Le tableau suivant décrit textuellement le cas d'utilisation Réinitialiser mot de passe.

Cas d'utilisation : Réinitialiser mot de passe	
Acteur(s)	User
Objectif	Permettre aux utilisateurs de réinitialiser son mot de passe afin de pouvoir utiliser les fonctionnalités de l'application.
Pré condition	L'utilisateur doit avoir un compte.
Post condition	L'utilisateur a un nouveau mot de passe.
Scénario nominal	<ol style="list-style-type: none"> 1. L'utilisateur clique sur le lien « Mot de passe oublié ». 2. Le système le redirige vers une autre interface avec un formulaire de l'email. 3. L'utilisateur remplit son email et clique sur le bouton « réinitialiser mot de passe ». 4. Un code aléatoire sera envoyé vers la boîte mail d'utilisateur. 5. Le système le redirige vers une autre interface avec une zone de texte pour saisir le code envoyé par mail. 6. L'utilisateur recopie le code envoyé par mail et clique sur bouton « réinitialiser mot de passe ». 7. L'utilisateur clique sur le lien lui permettant de rediriger vers une interface contenant un formulaire pour le nouveau mot de passe.
Scénario alternatif	L'email saisi est invalide.

Tableau 6 : Description textuelle du cas d'utilisation « Réinitialiser mot de passe »

- Le tableau suivant décrit textuellement le cas d'utilisation « Gérer les patients ».

Cas d'utilisation : Gérer les patients	
Acteur(s)	Médecin
Objectif	Permettre médecin de gérer les patients.
Pré condition	Le médecin doit être authentifié.

Post condition	Gérer les patients.
Scénario nominal	<ol style="list-style-type: none"> 1. Le système affiche le tableau de tous les patients 2. Le médecin affiche la vue « liste détaillé » incluant tous les patients. 3. Le médecin clique sur l'icône (+) <ul style="list-style-type: none"> • Le système affiche un formulaire. • Le médecin remplit le formulaire avec les informations du patient. • Le médecin clique sur le bouton de validation • Le système crée un nouveau compte patient avec un message de succès. 4. Le médecin clique sur l'icône Editer <ul style="list-style-type: none"> • Le système affiche un formulaire. • Le médecin remplit le formulaire avec les nouvelles informations patient. • Le médecin clique sur le bouton de validation. • Le système enregistre le compte patient avec un message de succès. 5. Le médecin clique sur l'icône Archiver <ul style="list-style-type: none"> • Le système archive le patient avec ces historiques.
Scénario alternatif	<ol style="list-style-type: none"> 1. Le médecin saisit des informations invalides. Le système affiche un message d'erreur et retour à l'étape 2 du scénario nominal. 2. Le médecin saisit des informations déjà existantes dans la base de données, Le système recharge le formulaire de création d'un compte patient avec un message d'erreur.

Tableau 7 : Description textuelle du cas d'utilisation « Gérer les patients »

➤ Description textuelle du cas d'utilisation « S'inscrire »

Cas d'utilisation : S'inscrire	
Acteur(s)	Patient
Objectif	Permettre aux patients de créer un compte.
Pré condition	Le patient ne doit pas avoir un compte.
Post condition	Création d'un compte de patient
Scénario nominal	<ol style="list-style-type: none">1. Le système affiche le formulaire d'inscription patient.2. Le patient remplit le formulaire avec ses informations.3. Le patient clique sur le bouton de validation.4. Le système vérifie les données.5. Le système crée un nouveau compte patient avec un message de succès.6. Le système redirige vers la page d'authentification.
Scénario alternatif	<ol style="list-style-type: none">1. Le médecin saisit des informations invalides. Le système affiche un message d'erreur et retour à l'étape 2 du scénario nominal.2. Le médecin saisit des informations déjà existantes dans la base de données, Le système recharge le formulaire de création d'un compte patient avec un message d'erreur.

Tableau 8 : Description textuelle du cas d'utilisation « S'inscrire »

3 Description comportementale des histoires du sprint

Nous présentons dans cette partie les diagrammes de séquences de notre application qui sert à modéliser les interactions entre les objets.

○ *Diagramme de séquences « Authentification »*

Ce scénario présente la procédure d'authentification. L'utilisateur saisit les informations requises et valide le formulaire.

Si les données saisies sont incorrectes, un message d'erreur s'affiche pour refaire la saisie.

La figure dans la page suivante montre l'interaction entre le système et l'utilisateur pour la réalisation du cas d'utilisation « **Authentification** ».

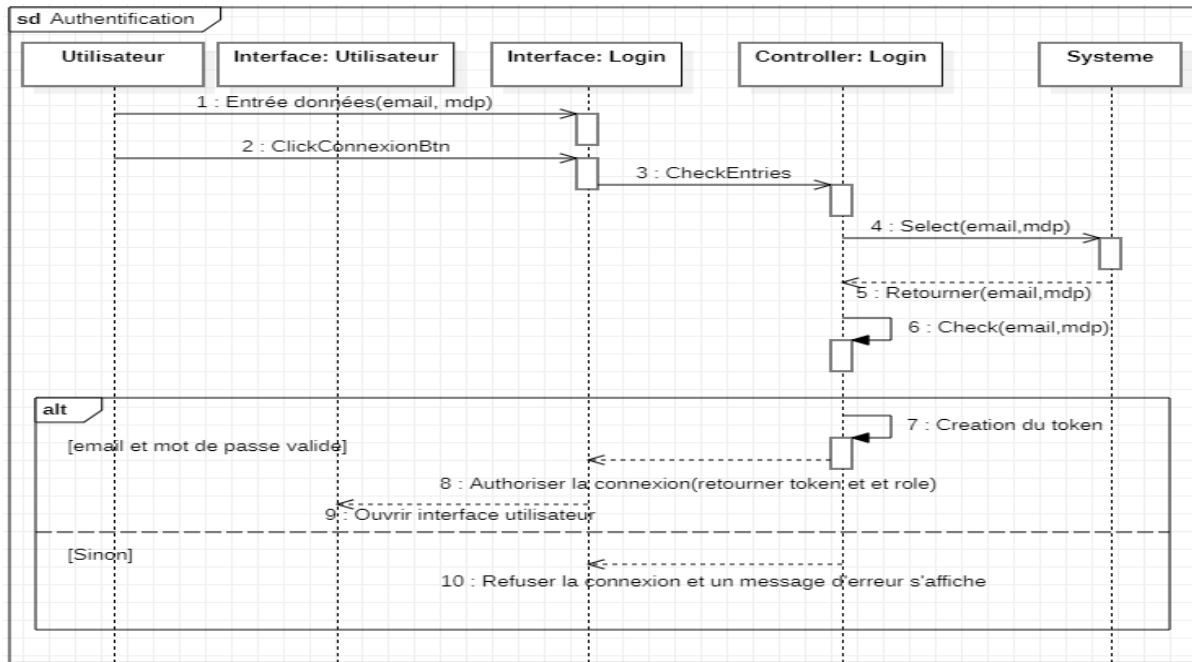


Figure 11 : Diagramme de séquence d'authentification

○ **Diagramme de séquences « Réinitialiser mot de passe »**

Ce scénario présente la procédure de réinitialiser mot de passe. Dans le cas où l'utilisateur oublie son mot de passe, il aura la possibilité de la renouveler à travers son email.

Si l'email saisi est incorrect (n'existe pas dans la base de données), un message d'erreur s'affiche pour refaire la saisie.

La figure ci-dessous montre l'interaction entre le système et l'utilisateur pour la réalisation du cas d'utilisation « **Réinitialiser mot de passe** ».

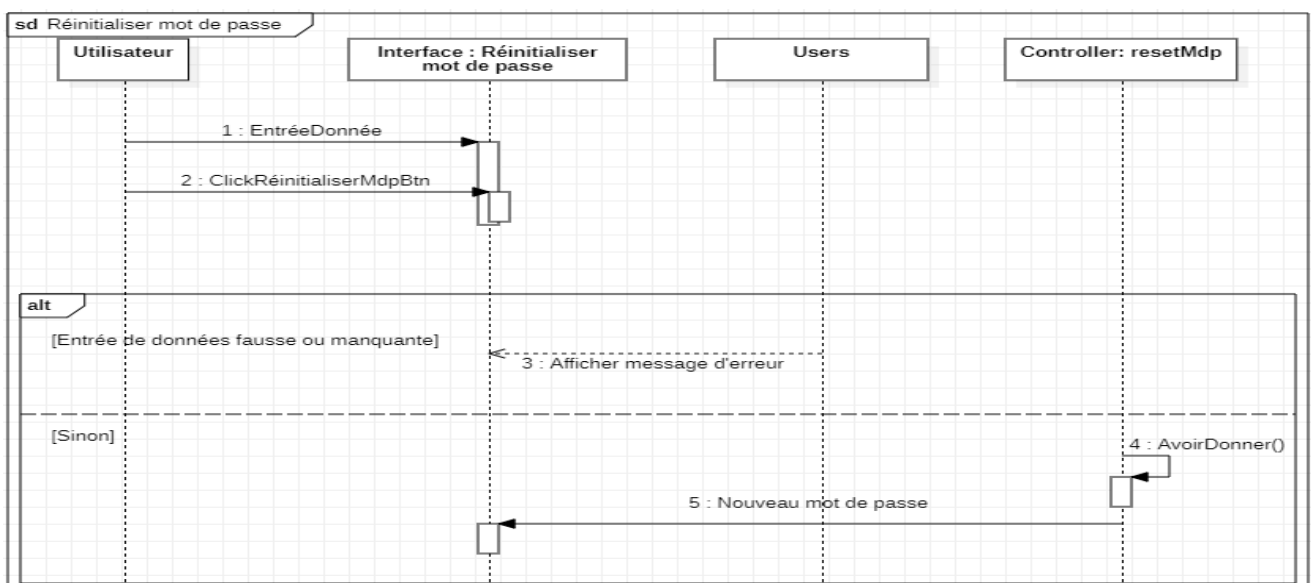


Figure 12 : Diagramme de séquence de réinitialiser mot de passe

○ *Diagramme de séquences « Gérer les patients »*

Ce scénario présente la procédure de gérer les patients (ajouter, modifier, archiver, chercher). Le médecin a le droit de gérer les comptes de ses patients.

Le médecin doit être authentifié afin de gérer ses patients.

Les figures ci-dessous montre l'interaction entre le système et le médecin pour la réalisation du cas d'utilisation « *Gérer les patients* ».

▪ *Diagramme de séquences « Gérer les patients : ajouter patient »*

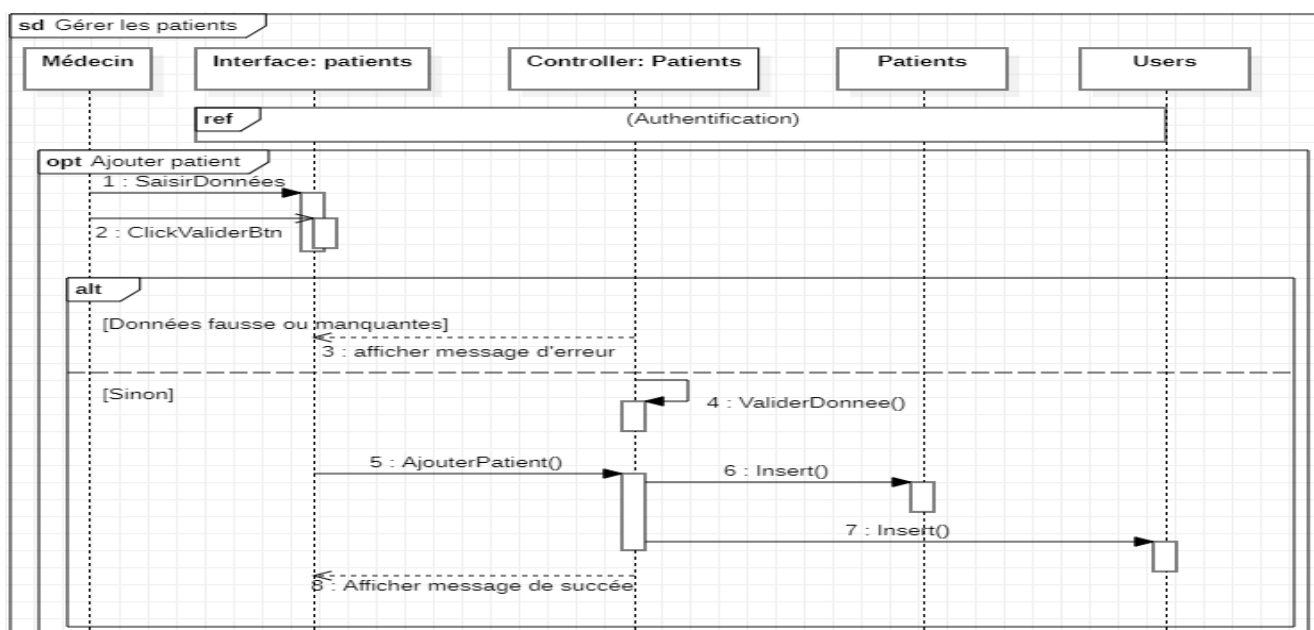


Figure 13 : Diagramme de séquence Gérer les patients : ajouter patient

▪ *Diagramme de séquences « Gérer les patients : modifier patient »*

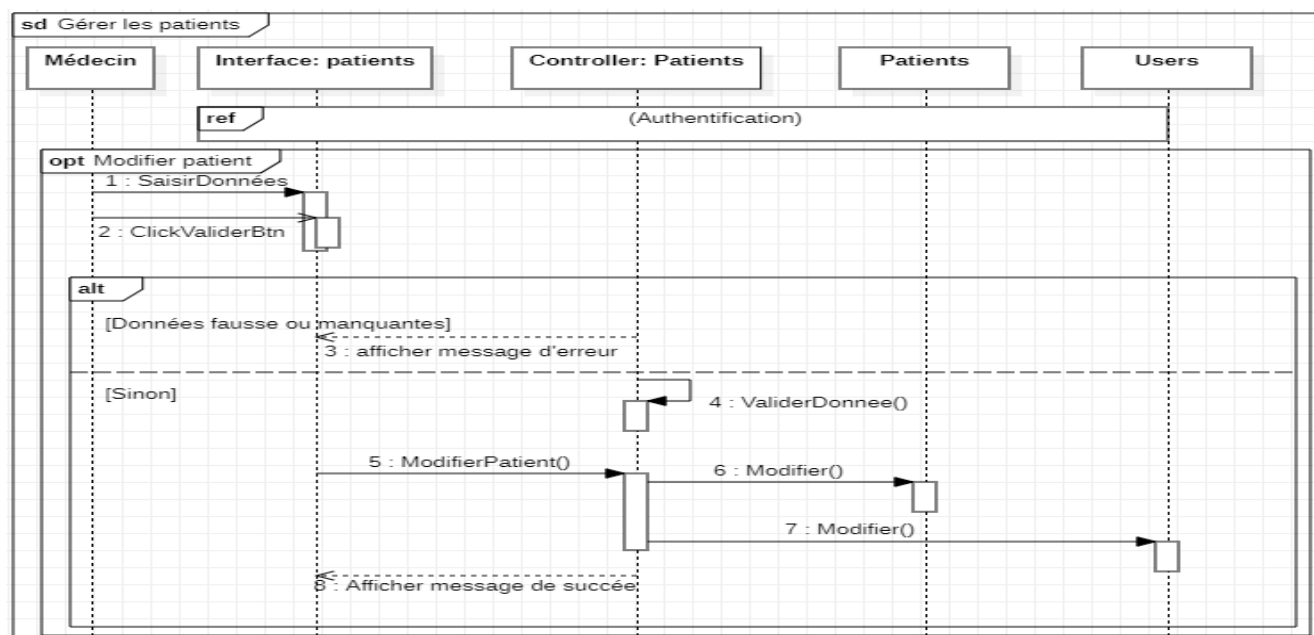


Figure 14 : Diagramme de séquence Gérer les patients : modifier patient

▪ **Diagramme de séquences « Gérer les patients : archiver patient »**

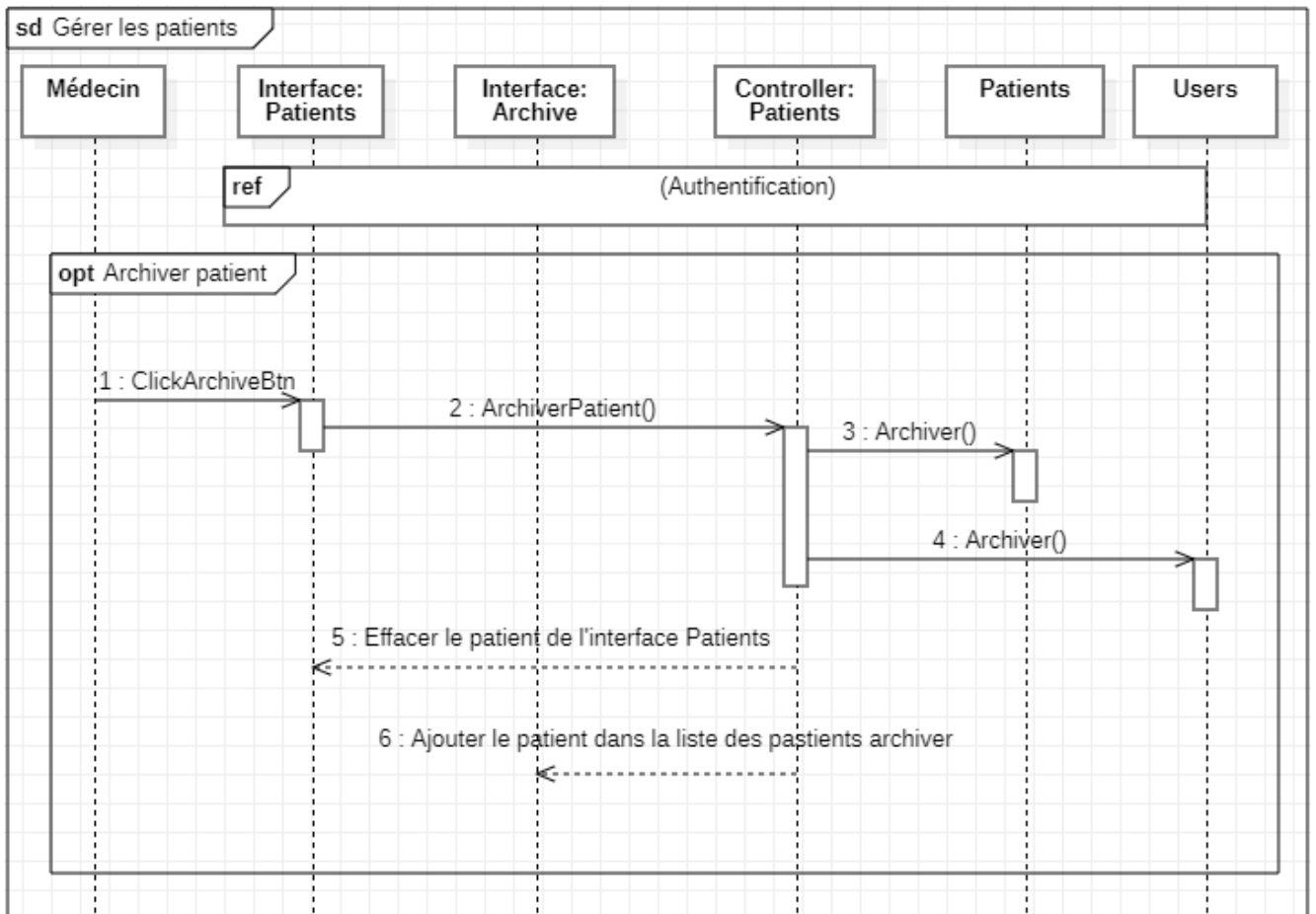


Figure 15 : Diagramme de séquence Gérer les patients : archiver patient

▪ **Diagramme de séquences « Gérer les patients : chercher patient »**

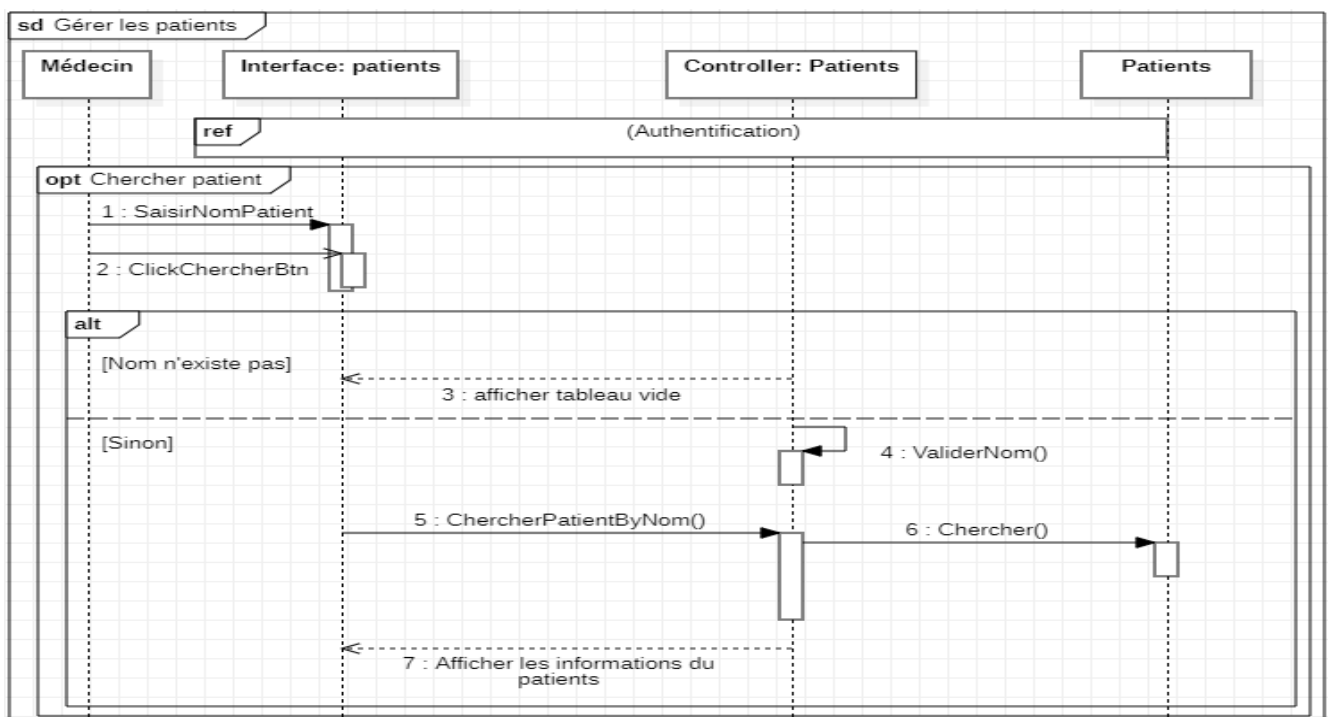


Figure 16 : Diagramme de séquence Gérer les patients : chercher patient

○ Diagramme de séquences « S'inscrire »

Ce scénario présente la procédure de l'inscription des patients. Chaque utilisateur a le droit de créer un compte patient.

La figure ci-dessous montre l'interaction entre le système et le patient pour la réalisation du cas d'utilisation « S'inscrire ».

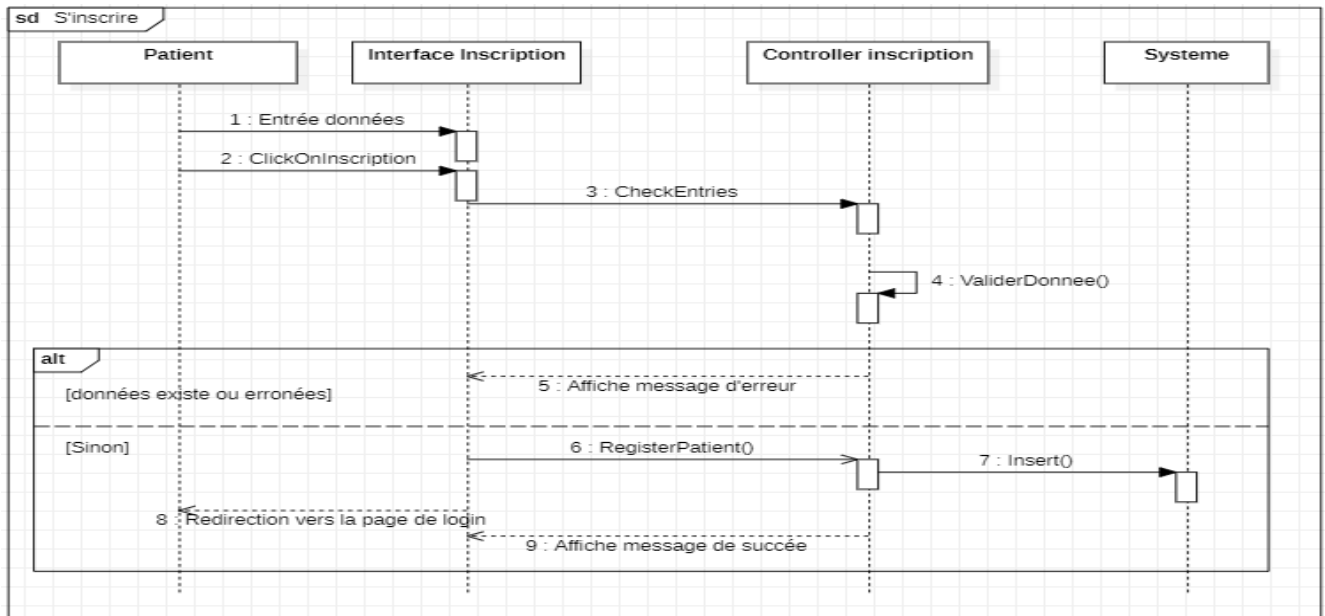


Figure 17 : Diagramme de séquence s'inscrire

4 Réalisation du sprint 1

Pour illustrer la réalisation du premier sprint, nous présenterons les interfaces des principales fonctionnalités offertes par le module utilisateur.

➤ Interface d'accueil :

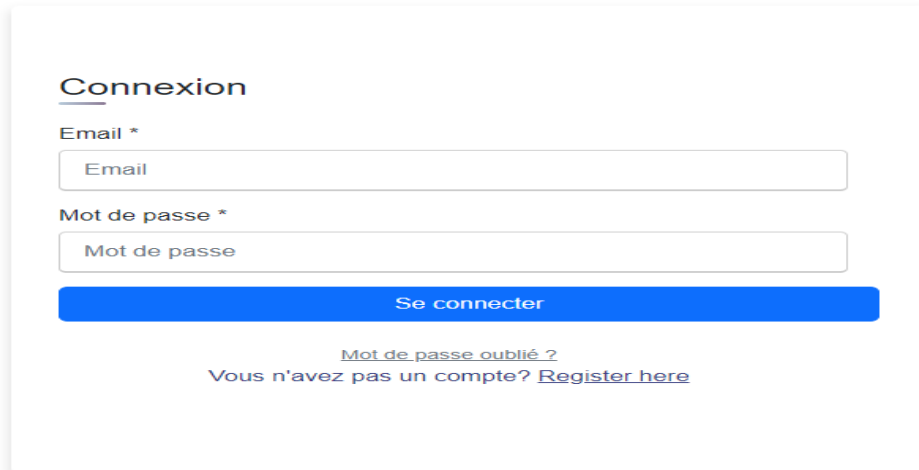
C'est la première interface qui apparaît pour les utilisateurs lorsqu'ils accèdent à l'application. Nous trouvons le logo de notre plateforme et un bouton qui nous redirige vers l'interface d'authentification.



Figure 18 : Interface d'accueil

➤ **Interface d'authentification :**

L'utilisateur saisit son login, qui doit être une adresse email, et son mot de passe (ne contient pas son nom et il contient au moins 8 caractères), puis clique sur le bouton "Connexion". Le système vérifie la validité des informations saisies en dirigeant l'utilisateur vers sa propre interface selon son rôle.

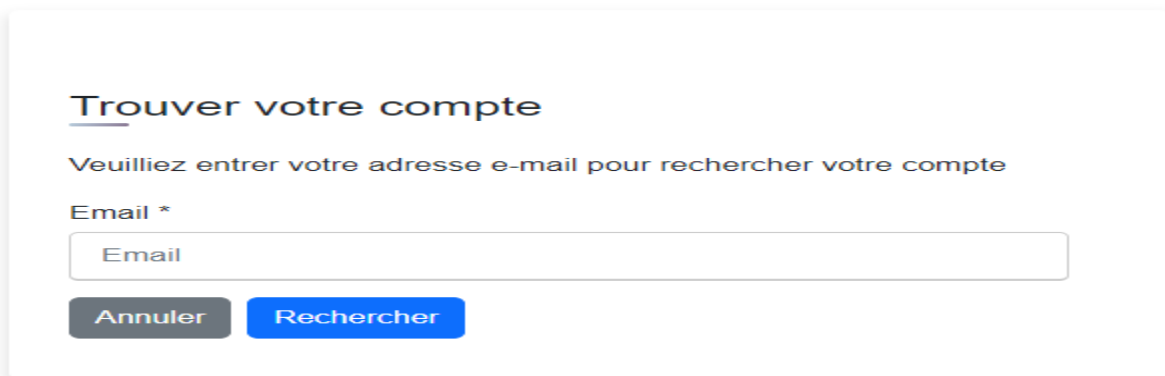


The screenshot shows a login form with the title "Connexion". It contains two input fields: "Email *" and "Mot de passe *". Below the "Mot de passe *" field is a blue button labeled "Se connecter". At the bottom, there is a link "Mot de passe oublié ?" and a text line "Vous n'avez pas un compte? [Register here](#)".

Figure 19 : Interface d'authentification

➤ **Interface mot de passe oubliée :**

L'utilisateur doit saisir son e-mail afin de recevoir un code de vérification pour qu'il puisse modifier son mot de passe.



The screenshot shows a form titled "Trouver votre compte". Below the title is the instruction "Veuillez entrer votre adresse e-mail pour rechercher votre compte". There is an input field labeled "Email *". At the bottom, there are two buttons: "Annuler" (grey) and "Rechercher" (blue).

Figure 20 : Interface mot de passe oubliée

➤ **Interface pour annoncer l'envoi d'e-mail :**

L'utilisateur va recevoir un mail vers l'e-mail qu'il a saisi.

Nous vous enverrons un code à votre adresse e-mail

Nous pouvons envoyer un code de connexion à
aya@gmail.com

[se connecter avec le mot de passe](#) [Continuer](#)

Figure 21 : Interface pour annoncer l'envoi d'e-mail

➤ **Interface de la vérification du code :**

L'utilisateur doit saisir le code qu'il a reçu.

Entrer le code de sécurité

Merci de vérifier dans vos e-mails que vous avez reçu un message avec votre code. Celui-ci est composé de 6 chiffres.

Code *

[Code non reçu ?](#) [Annuler](#) [Continuer](#)

Figure 22 : Interface de la vérification du code

➤ **Interface pour la modification de mot de passe :**

Nouveau mot de passe

Email *

Nouveau mot de passe *

Répéter mot de passe *

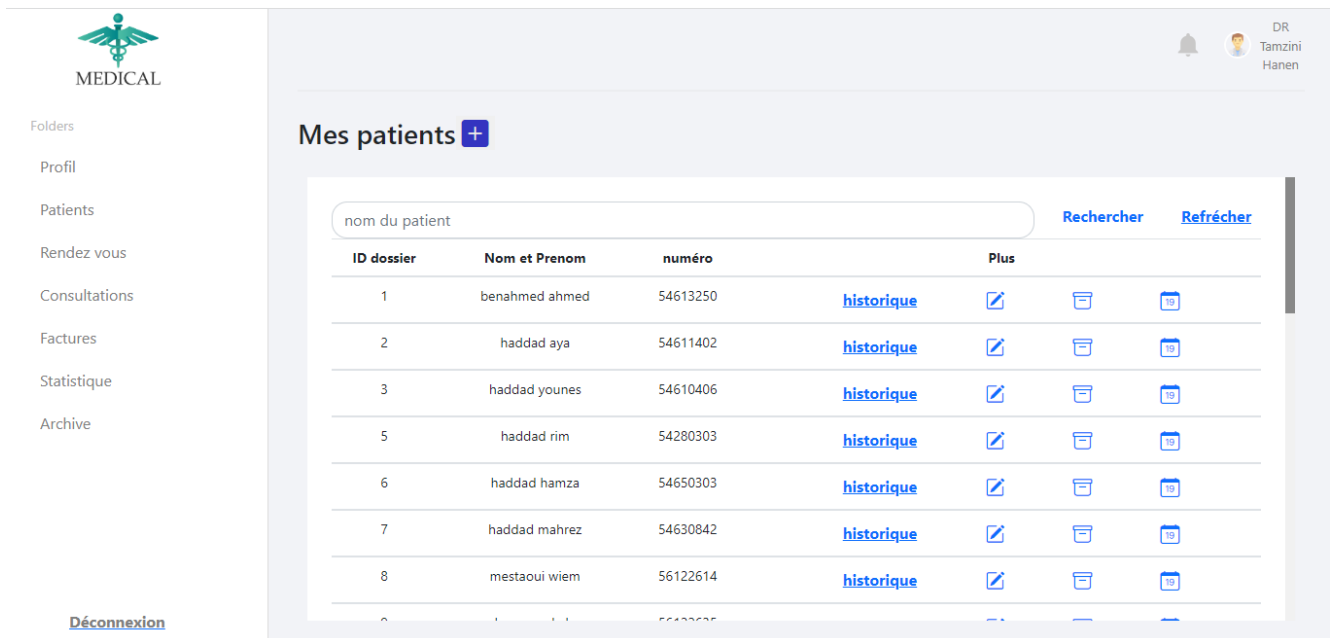
[Se connecter](#)

Figure 23 : Interface pour la modification de mot de passe

➤ Interface pour gérer les patients :

Après l'authentification, le médecin sera capable de consulter son Dashboard afin de gérer ses patients.

Le médecin peut chercher un patient par son nom.

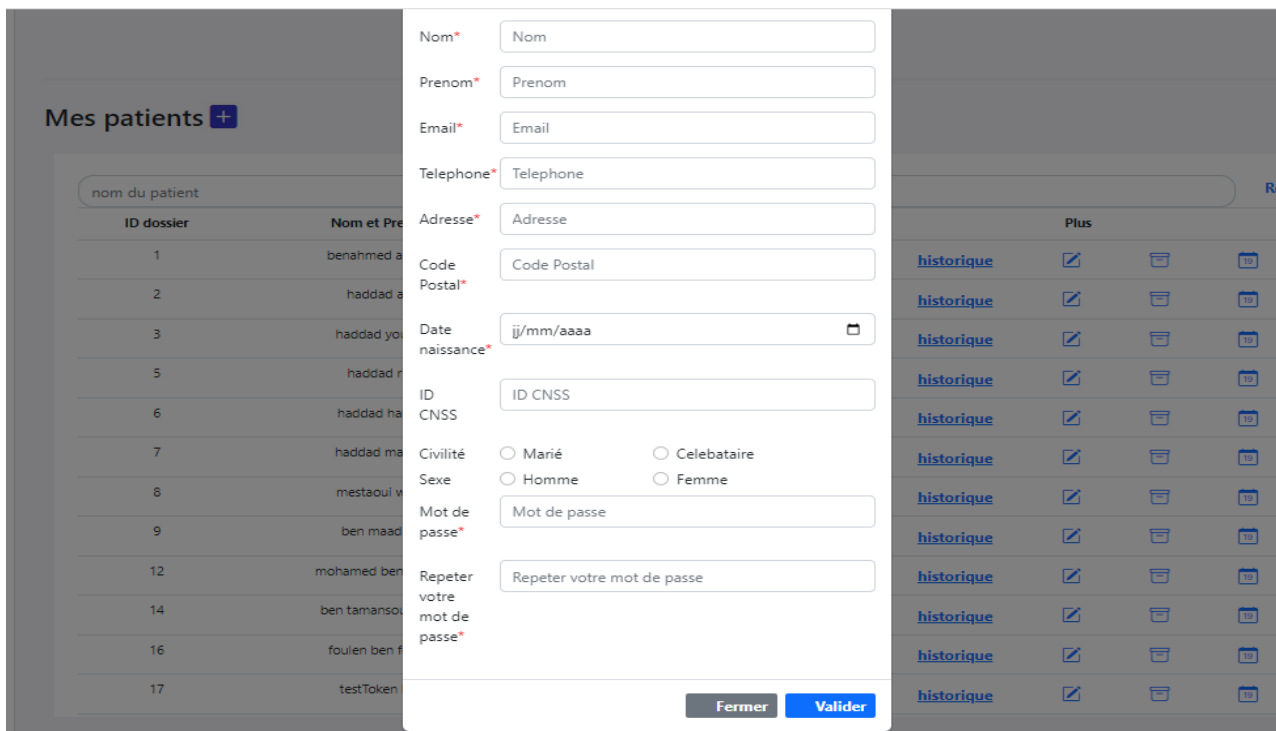


The screenshot shows a web application interface for managing patients. On the left is a sidebar with a 'MEDICAL' logo and a menu: Folders, Profil, Patients, Rendez vous, Consultations, Factures, Statistique, Archive, and Déconnexion. The main area is titled 'Mes patients +'. It features a search bar labeled 'nom du patient' with 'Rechercher' and 'Réfrécher' buttons. Below is a table of patients:

ID dossier	Nom et Prenom	numéro	Plus
1	benahmed ahmed	54613250	historique [edit] [archive] [calendar]
2	haddad aya	54611402	historique [edit] [archive] [calendar]
3	haddad younes	54610406	historique [edit] [archive] [calendar]
5	haddad rim	54280303	historique [edit] [archive] [calendar]
6	haddad hamza	54650303	historique [edit] [archive] [calendar]
7	haddad mahrez	54630842	historique [edit] [archive] [calendar]
8	mestaoui wiem	56122614	historique [edit] [archive] [calendar]

Figure 24 : Interface pour consulter, archiver et chercher patient

Le médecin peut ajouter un compte de patient en cliquant sur le bouton (+) et il doit remplir le formulaire.



The screenshot shows the 'Ajouter patient' form overlaying the 'Mes patients' list. The form fields are:

- Nom* (text input)
- Prenom* (text input)
- Email* (text input)
- Telephone* (text input)
- Adresse* (text input)
- Code Postal* (text input)
- Date naissance* (date picker, format jj/mm/aaaa)
- ID CNSS (text input)
- Civilité (radio buttons: Marié, Célibataire)
- Sexe (radio buttons: Homme, Femme)
- Mot de passe* (password input)
- Repetez votre mot de passe* (password input)

At the bottom of the form are 'Fermer' and 'Valider' buttons.

Figure 25 : Interface ajouter patient

Le médecin peut modifier les informations de son patient en cliquant sur l'icône de modification.

Modification patient

Nom * : benahmed

Prenom : ahmed

Email * : anis@gmail.com

Telephone : 54613250

Adresse : Djerba midoun

Code : 4541

Postal * :

Date de naissance : 03/05/2023

ID : 123659

CNAM :

Sexe : ☐ Homme ☐ Femme

Civilité : ☐ Marié ☐ Célibataire

Fermer Modifier

Figure 26 : Interface modifier patient

Avec un seul clic sur l'icône d'archivage, le patient sera archivé et sera éliminé de la liste des patients (en termes d'affichage).

MEDICAL

DR Tamzini Hanen

Dashboard / Archive

nom du patient [Rechercher](#) [Refresh](#)

ID dossier	Nom et Prenom	numéro	More
12	mohamed ben ahmed	56366035	Historique Désarchiver
14	ben tamansourt ines	50369705	Historique Désarchiver
49	asma rahma	2356811	Historique Désarchiver
58	sami ben	12345670	Historique Désarchiver
60	christine gnama	22336655	Historique Désarchiver

Déconnexion

Figure 27 : Interface archive patient

➤ Interface d'inscription patient :

En cliquant sur le mot "s'inscrire" on se redirige vers une page contenant un formulaire où l'utilisateur doit indiquer les détails de son compte. Si un champ est manquant, une erreur est affichée, sinon l'utilisateur est authentifié et redirigé à l'interface d'authentification.



The image shows a patient registration form titled "Inscription". It is organized into two columns. The left column contains fields for "Nom *", "Email *", "Adresse *", "ID cnam *", "Mot de passe *", "Sexe *", and "Civilité *". The right column contains fields for "Prenom *", "Numéro telephone *", "Code postal *", "Date naissance *", and "Repeter mot de passe". Each text input field has a placeholder text starting with "Taper votre". The "Date naissance *" field has a date format mask "jj/mm/aaaa" and a calendar icon. The "Sexe *" and "Civilité *" fields use radio buttons with options "Homme", "Femme", "Marié", and "Celebataire". A blue "S'inscrire" button is at the bottom center.

Inscription	
Nom *	Prenom *
<input type="text" value="Taper votre nom"/>	<input type="text" value="Taper votre prénom"/>
Email *	Numéro telephone *
<input type="text" value="Taper votre email"/>	<input type="text" value="Taper votre numéro telephone"/>
Adresse *	Code postal *
<input type="text" value="Taper votre adresse"/>	<input type="text" value="Taper votre code postal"/>
ID cnam *	Date naissance *
<input type="text" value="Taper votre ID cnam"/>	<input type="text" value="jj/mm/aaaa"/>
Mot de passe *	Repeter mot de passe
<input type="text" value="Taper votre mot de passe"/>	<input type="text" value="Repeter votre mot de passe"/>
Sexe *	
<input type="radio"/> Homme	<input type="radio"/> Femme
Civilité *	
<input type="radio"/> Marié	<input type="radio"/> Celebataire
<input type="button" value="S'inscrire"/>	

Figure 28 : Interface d'inscription patient

Conclusion

Dans ce chapitre nous avons établi le premier sprint, et nous avons défini les différentes fonctionnalités réalisées à travers ses descriptions fonctionnelles et comportementales des histoires. Finalement nous avons défini son diagramme de classes suivi de ses réalisations en termes d'interface réalisées dans ce sprint. Dans le chapitre suivant nous allons décrire le deuxième sprint.

Chapitre 4 : Sprint 2 : Gestion des rendez-vous

Introduction

Dans ce chapitre nous allons présenter le déroulement pour développer la version qui concerne la gestion de rendez-vous en suivant le même enchaînement que le chapitre précédent. Nous allons commencer par la présentation du backlog du notre premier sprint. Ensuite nous allons passer à la description fonctionnelle en présentant les diagrammes de cas d'utilisation et le diagramme de classes ainsi que la description comportementale à l'aide des diagrammes de séquences. Enfin, nous allons présenter les interfaces de réalisation reliée à ce chapitre.

1 Backlog de sprint 2

Le tableau ci-dessous consiste à présenter le Product backlog pour le sprint 2 correspond au module des rendez-vous.

Product Backlog (Gestion des rendez-vous)				
Fonctionnalité	ID	User story	Tâches	Estimation
Consulter la disponibilité	1	En tant que médecin je peux gérer les horaires de la disponibilité de mon cabinet.	-Conception -Implémentation -Test	2 jours
	2	En tant que patient je peux voir les coordonner et la disponibilité du cabinet de mon médecin.	-Conception -Implémentation -Test	1 jour
Réservation	1	En tant que patient je peux réserver un rendez-vous.	-Conception -Implémentation -Test	4 jours
	2	En tant que patient je peux consulter mes rendez-vous.		
	3	En tant que patient je peux annuler mon rendez-vous.		
	4	En tant que patient je peux modifier la date de mon rendez-vous.		
	5	En tant que médecin je peux réserver un rendez-vous.	-Conception -Implémentation -Test	2 semaines
	6	En tant que médecin je peux consulter ma liste des rendez-vous.		
	7	En tant que médecin je peux annuler un rendez-vous.		
	8	En tant que médecin je peux modifier la date de mon rendez-vous.		
Mailing	1	En tant que médecin je peux envoyer des mails à mes patients pour les rappeler de leur rendez-vous	-Conception -Implémentation -Test	3 jours

Tableau 9 : Backlog de sprint 2

2 Description fonctionnelle des histoires du sprint 2

Dans cette partie nous allons commencer par l'élaboration du diagramme de classe de sprint 2 et pour mieux illustrer les différentes manières dont les utilisateurs peuvent interagir avec le système, les fonctionnalités mentionnées dans le Backlog de sprint 2 sont illustrées dans un diagramme de cas d'utilisation global du notre sprint et nous le détaillons selon nos acteurs.

2.1 Diagramme de classe

La figure ci-dessous représente le diagramme de classe de sprint 2

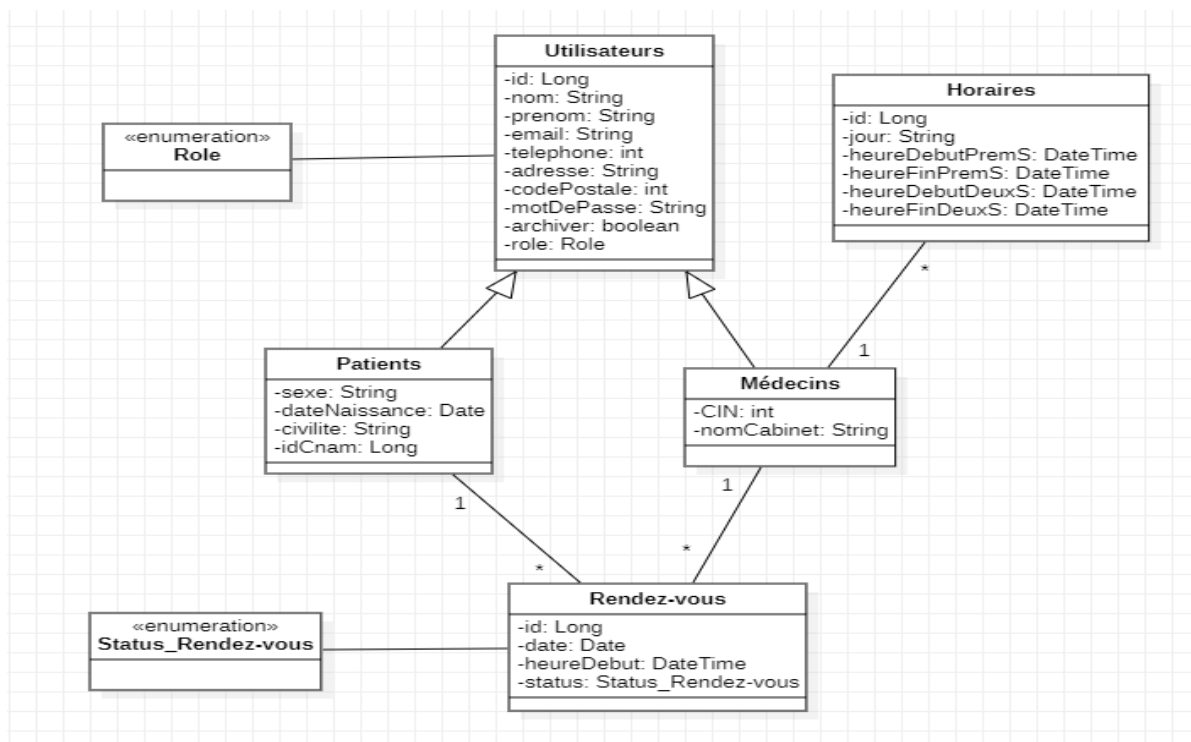


Figure 29 : Diagramme de classe du sprint 2

2.2 Diagramme de cas d'utilisation

2.2.1 Diagramme de cas d'utilisation global du sprint 2

Pour ce second sprint, nous présentons le diagramme de cas d'utilisation global qui illustre les fonctionnalités offertes par le module « Rendez-vous ».

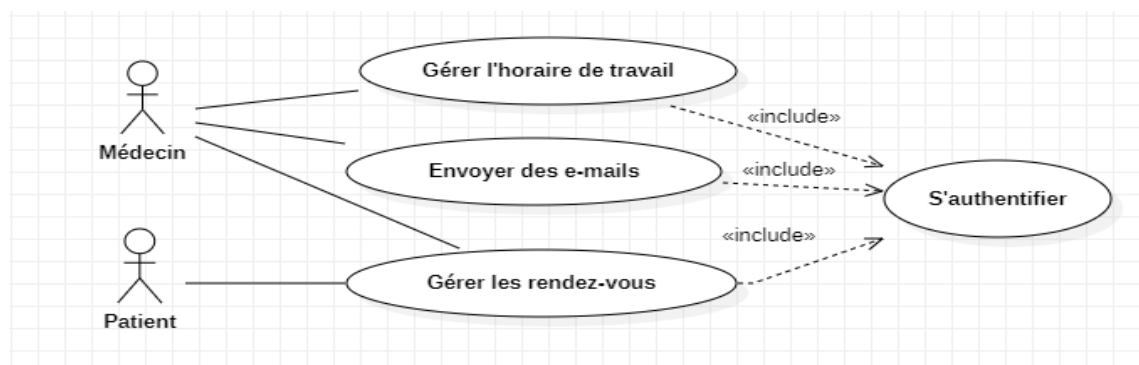


Figure 30 : Diagramme de cas d'utilisation global du sprint 2

2.2.2 Diagramme de cas d'utilisation détaillé de sprint 2

Après avoir présenté le diagramme de cas d'utilisation global de notre deuxième sprint, nous allons expliquer les fonctionnalités de chaque cas d'utilisation.

- **Diagramme de cas d'utilisation détaillé du médecin :**
« Gérer l'horaire de travail »



Figure 31 : Diagramme de cas d'utilisation détaillé du médecin : « Gérer l'horaire de travail »

- **Diagramme de cas d'utilisation détaillé du médecin :**
« Envoyer des e-mails »



Figure 32 : Diagramme de cas d'utilisation détaillé du médecin : « Envoyer des e-mails »

- **Diagramme de cas d'utilisation détaillé du médecin et patient :**
« Gérer les rendez-vous »

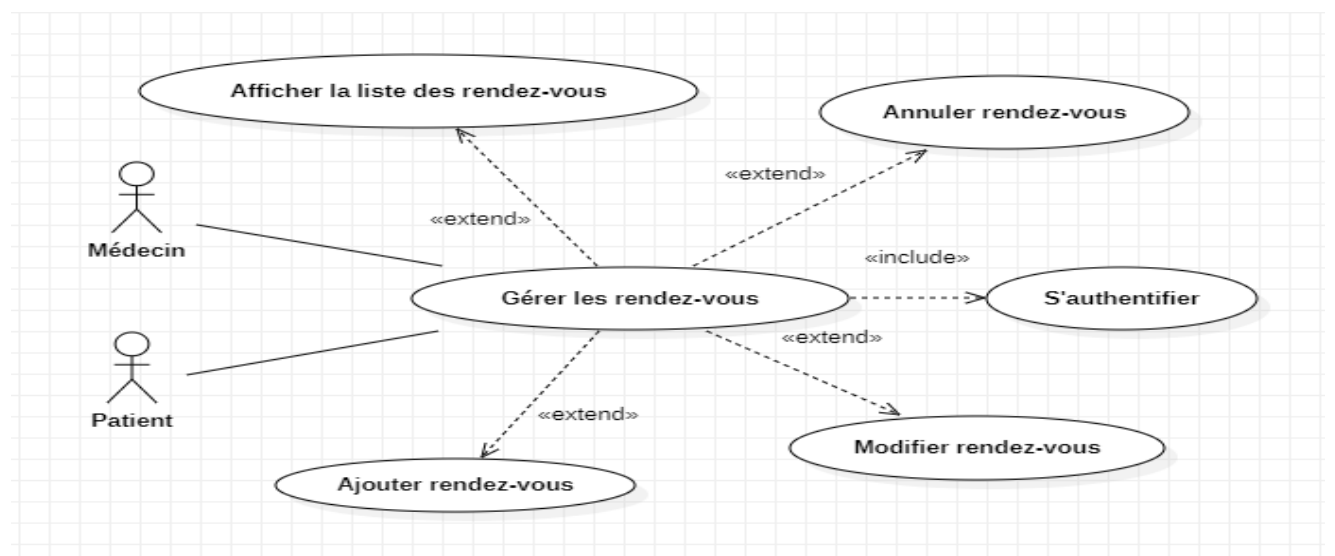


Figure 33 : Diagramme de cas d'utilisation détaillé du médecin et patient : « Gérer les rendez-vous »

2.3 Description textuelle des cas d'utilisations

Dans cette partie nous décrivons textuellement les cas d'utilisation principale qui apparaissent dans le digramme de cas d'utilisation global du sprint 2.

➤ Le tableau suivant décrit textuellement le cas d'utilisation Gérer l'horaire de travail.

Cas d'utilisation : Gérer l'horaire de travail	
Acteur(s)	Médecin
Objectif	Permettre au médecin de « Gérer ses horaires de travail (ajouter, modifier, consulter) ».
Pré condition	Le médecin doit être authentifié.
Post condition	Gérer les horaires de travail.
Scénario nominal	<ol style="list-style-type: none"> 1. Le système affiche le profil de médecin. 2. Le médecin clique sur l'icône (+) : <ul style="list-style-type: none"> • Le système affiche un formulaire. • Le médecin remplit le formulaire avec les horaires de travail. • Le médecin clique sur le bouton de validation. • L'horaire sera enregistré dans la base de données et un message de succès sera affiché. 3. Le médecin clique sur l'icône « Editer » : <ul style="list-style-type: none"> • Le système affiche un formulaire. • Le médecin remplit le formulaire avec les nouveaux horaires de travail. • Le médecin clique sur le bouton de validation. • Le système modifie l'horaire et un message de succès sera affiché. 4. Le médecin clique sur l'icône « Supprimer » : <ul style="list-style-type: none"> • Le système supprime l'horaire et un message de succès sera affiché.
Scénario alternatif	Le médecin saisi des information invalides. Le système affiche un message d'erreur.

Tableau 10 : Description textuelle de cas d'utilisation détaillé du médecin : « Gérer l'horaire de travail »

➤ Le tableau suivant décrit textuellement le cas d'utilisation Gérer les rendez-vous « Médecin ».

Cas d'utilisation : Gérer les rendez-vous « Médecin »	
Acteur(s)	Médecin
Objectif	Permettre au médecin de « Gérer les rendez-vous ».
Pré condition	Le médecin doit être authentifié.
Post condition	Gérer les rendez-vous (consulter, réserver, annuler, modifier).
Scénario nominal	<ul style="list-style-type: none"> • Réserver un rendez-vous : <ol style="list-style-type: none"> 1. Le système affiche la liste des patients. 2. Le médecin recherche le patient par son nom. 3. Le médecin clique sur l'icône de calendrier : <ol style="list-style-type: none"> a. Le système affiche un formulaire. b. Le médecin choisit la date de rendez-vous. c. Le médecin clique sur le bouton de la validation. d. Le système ajoute la date du rendez-vous dans la base de données et affiche un message de succès. • Modifier un rendez-vous : <ol style="list-style-type: none"> 1. Le système affiche la liste des rendez-vous. 2. Le médecin cherche un rendez-vous par le nom du patient. 3. Le médecin clique sur l'icône « Editer » : <ol style="list-style-type: none"> a. Le système affiche un formulaire. b. Le médecin remplit le formulaire avec la nouvelle date. c. Le médecin clique sur le bouton de validation. d. Le système enregistre les changements de la date dans la base de données et affiche un message de succès. • Annuler un rendez-vous : <ol style="list-style-type: none"> 1. Le système affiche la liste des rendez-vous. 2. Le médecin click sur le bouton « les rendez-vous non réaliser ». 3. Le médecin cherche un rendez-vous par le nom du patient. 4. Le médecin clique sur l'icône d'annulation. 5. Le système change le statu du rendez-vous à « ANNULER » et affiche un message du succès.

Scénario alternatif	Le cabinet est indisponible pendant la date sélectionnée.
---------------------	---

Tableau 11 : Description textuelle du cas d'utilisation « Gérer les rendez-vous » de médecin

- Le tableau suivant décrit textuellement le cas d'utilisation Gérer les rendez-vous « Patient »

Cas d'utilisation : Gérer les rendez-vous « Patient »	
Acteur(s)	Patient
Objectif	Permettre au patient de « Gérer leurs rendez-vous ».
Pré condition	Le patient doit être authentifié.
Post condition	Gérer les rendez-vous (consulter, réserver, annuler, modifier).

Scénario nominal	<ul style="list-style-type: none"> • Réserver un rendez-vous : <ol style="list-style-type: none"> 1. Le système affiche le profil du médecin. 2. Le patient clique sur l'icône « Calendrier » : <ol style="list-style-type: none"> a. Le système affiche un formulaire. b. Le patient choisit la date de rendez-vous. c. Le patient clique sur le bouton de validation. d. Le système ajout la date du rendez-vous dans la base de données et affiche un message de succès. • Modifier un rendez-vous : <ol style="list-style-type: none"> 1. Le système affiche la liste des rendez-vous. 2. Le patient clique sur le bouton « Non réalisés ». 3. Le patient cherche son rendez-vous par la date de rendez-vous. 4. Le patient clique sur l'icône « Editer » : <ol style="list-style-type: none"> a. Le système affiche un formulaire. b. Le patient remplit le formulaire avec la nouvelle date. c. Le patient clique sur le bouton de validation. d. Le système enregistre les changements de la date dans la base de données et affiche un message de succès. • Annuler un rendez-vous : <ol style="list-style-type: none"> 1. Le système affiche la liste des rendez-vous. 2. Le patient clique sur le bouton « Non réalisés ». 3. Le patient cherche un rendez-vous par la date de rendez-vous. 4. Le patient clique sur l'icône « Annulation ». 5. Le système change le statu du rendez-vous à « annuler » et affiche un message du succès.
Scénario alternatif	Le cabinet est indisponible pendant la date sélectionnée.

Tableau 12 : Description textuelle du cas d'utilisation « Gérer les rendez-vous » de patient

3 Description comportementale des histoires du sprint 2

Dans la suite, nous présentons les diagrammes de séquences correspondant aux principaux cas d'utilisation de notre sprint2.

- Diagramme de séquence « Gérer l'horaire de travail »

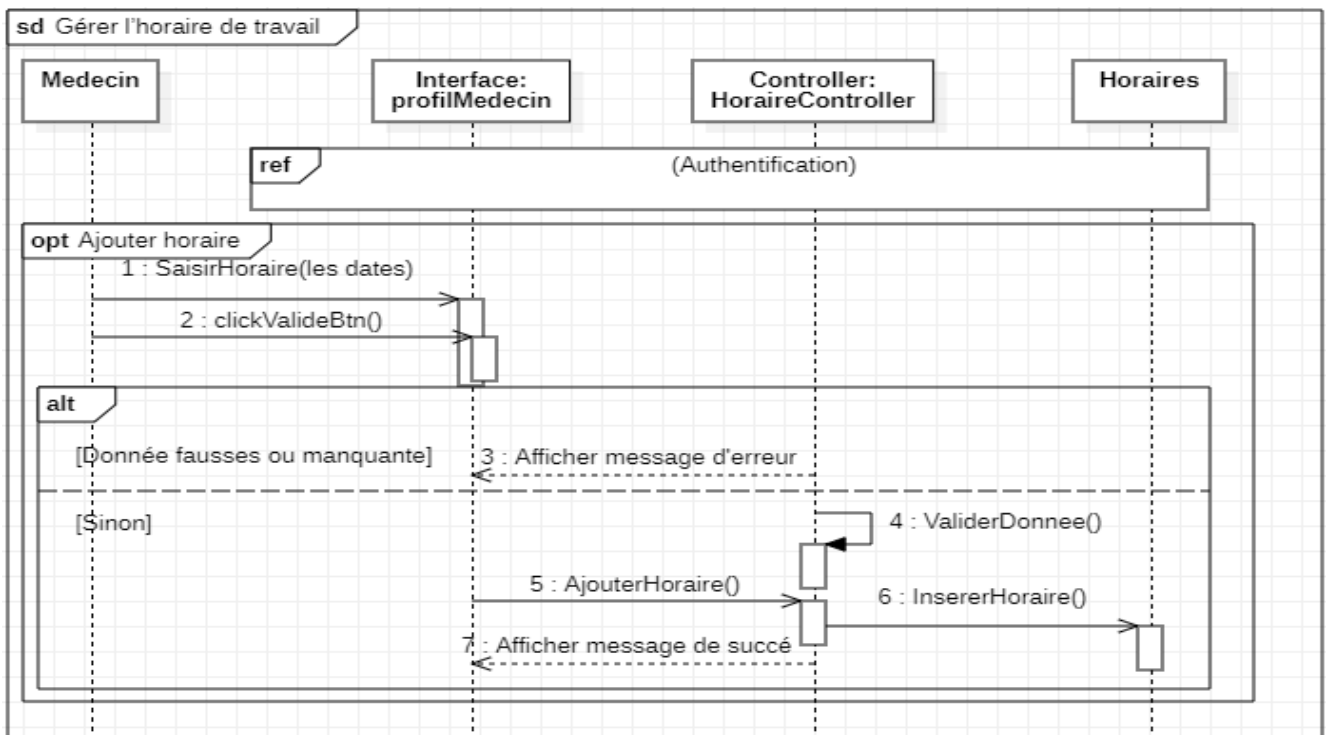


Figure 34 : Diagramme de séquence Gérer l'horaire de travail : Ajouter horaire

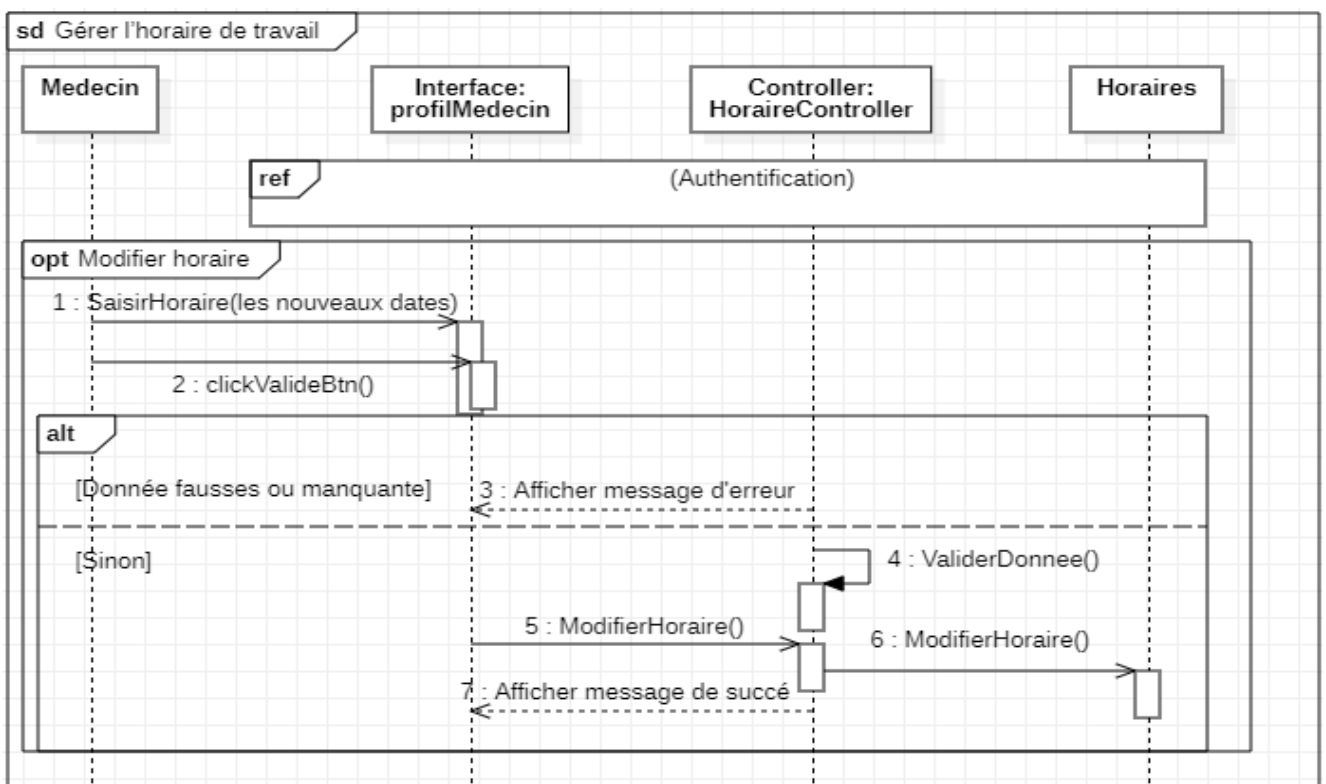


Figure 35 : Diagramme de séquence Gérer l'horaire de travail : Modifier horaire

- Diagramme de séquence « Gérer les rendez-vous : médecin »

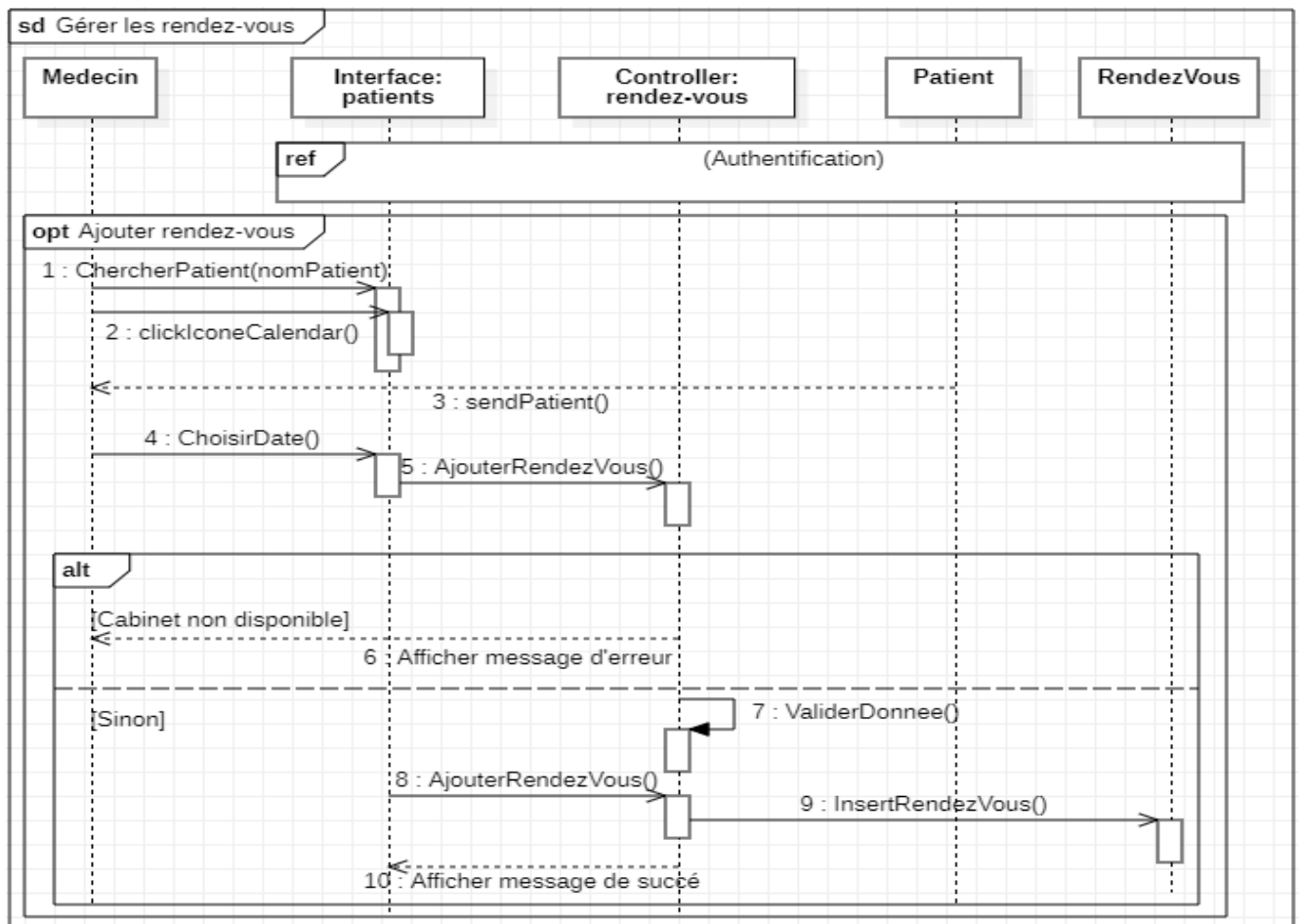


Figure 36 : Diagramme de séquence Gérer les rendez-vous (Ajouter rendez-vous) : médecin

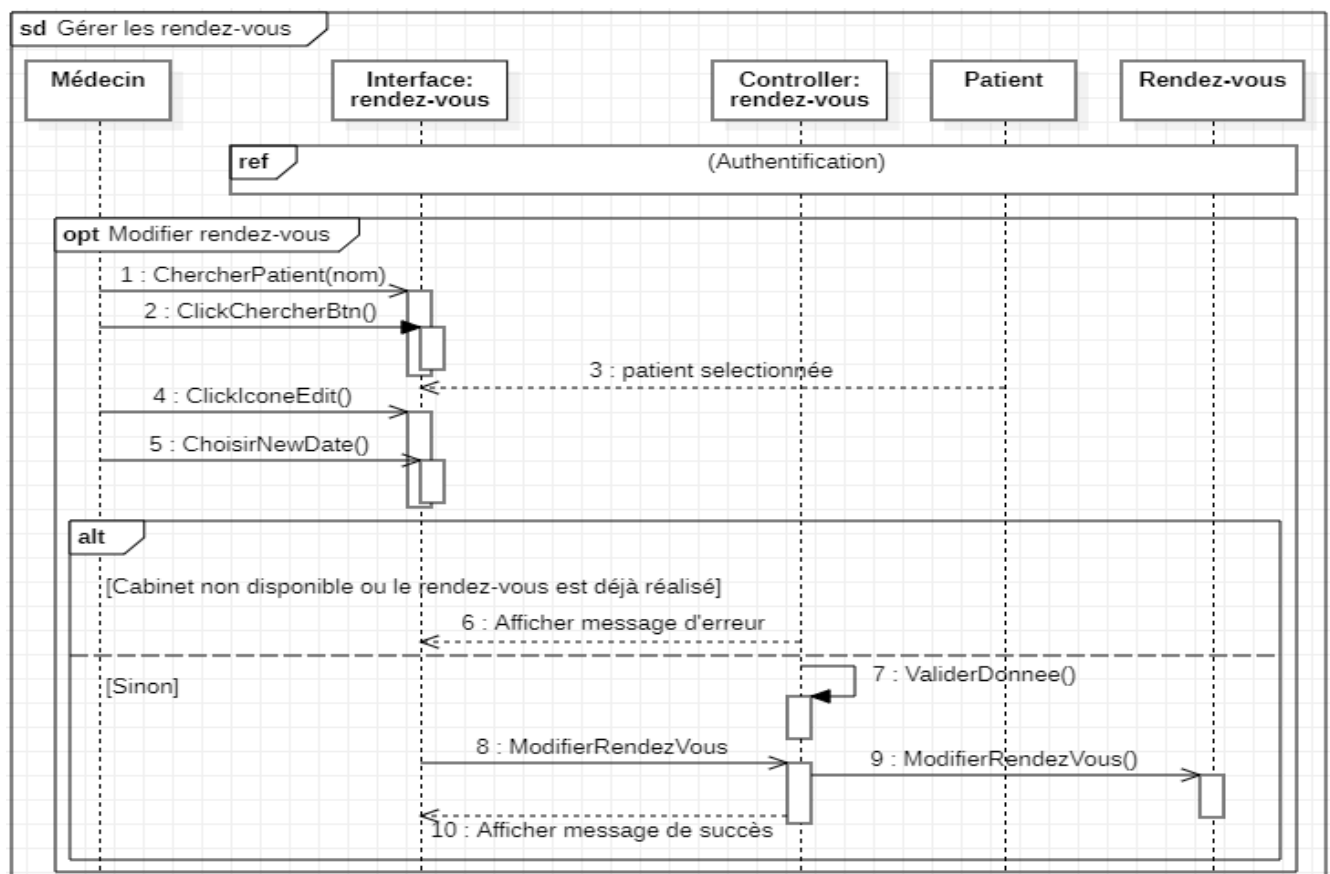


Figure 37 : Diagramme de séquence Gérer les rendez-vous (Modifier rendez-vous) : médecin

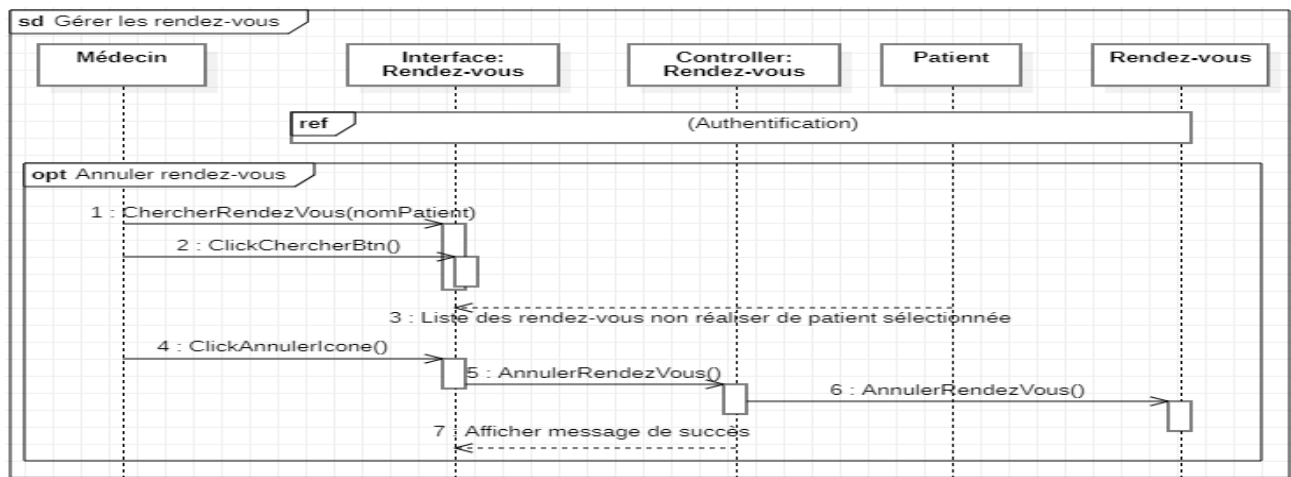


Figure 38 : Diagramme de séquence Gérer les rendez-vous (Annuler rendez-vous) : médecin

- Diagramme de séquence « Gérer les rendez-vous : patient »

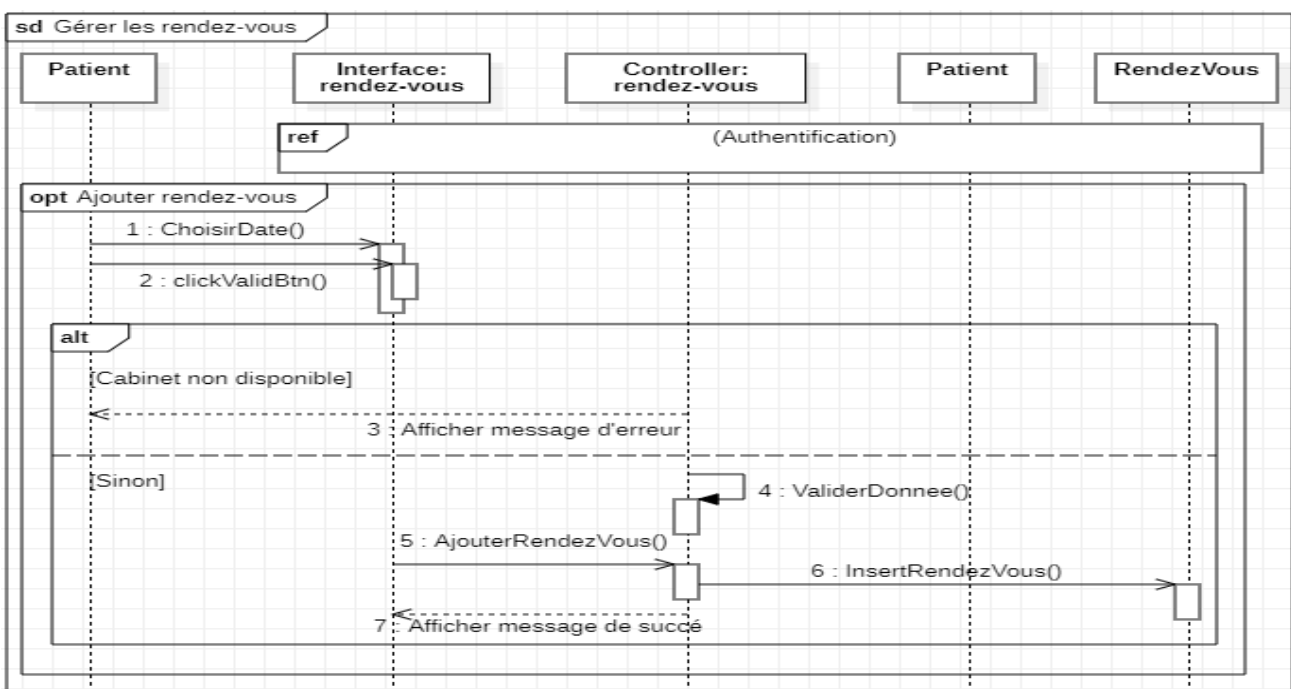


Figure 39 : Diagramme de séquence « Gérer les rendez-vous (Ajouter rendez-vous) : patient »

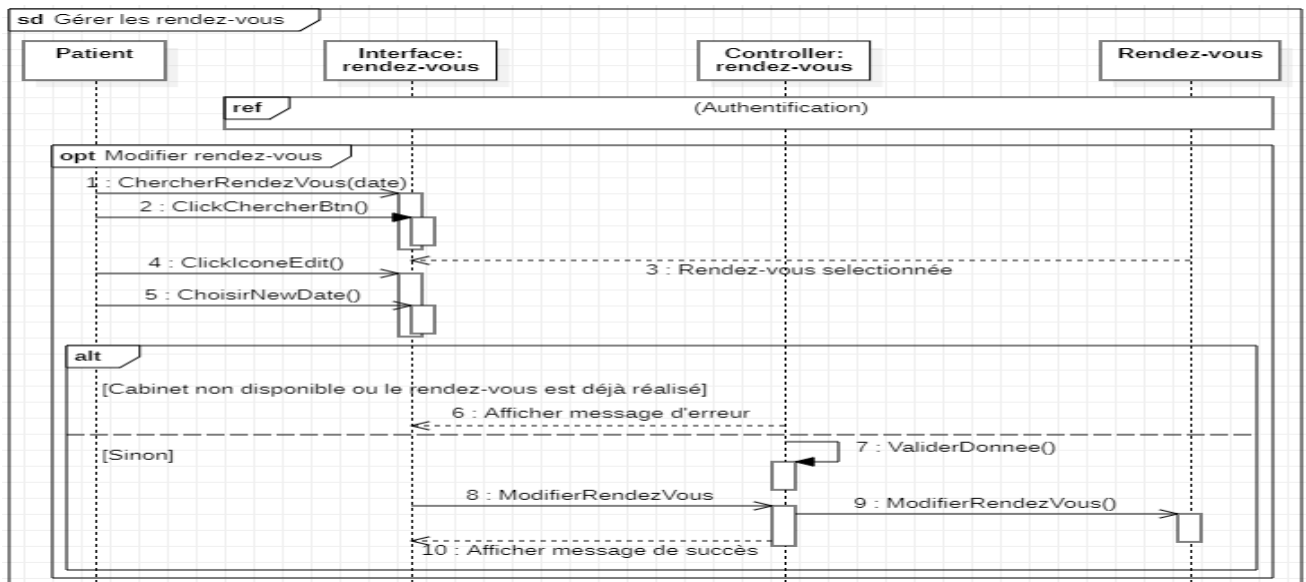


Figure 40 : Diagramme de séquence « Gérer les rendez-vous (Modifier rendez-vous) : patient »

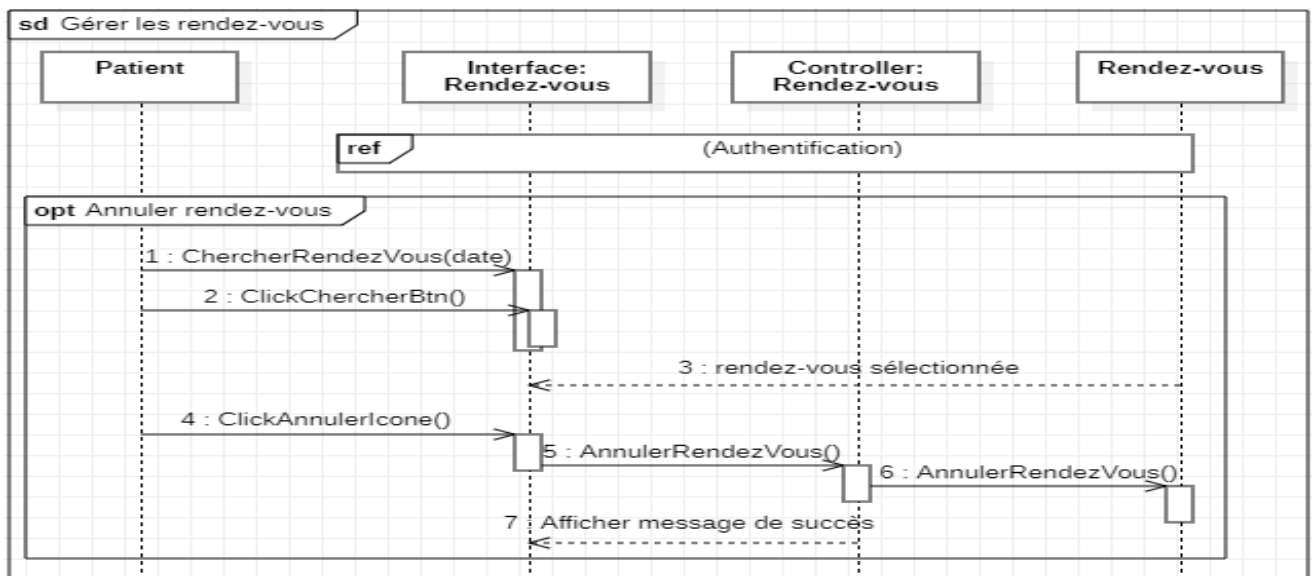


Figure 41 : Diagramme de séquence Gérer les rendez-vous (Annuler rendez-vous) : patient

4 Réalisation du sprint 2

Pour illustrer la réalisation du deuxième sprint, nous allons exposer les interfaces des principales fonctionnalités offertes par la gestion des rendez-vous.

4.1 Espace médecin

➤ Interface « Profil médecin » pour gérer l'horaire de travail

Dans cet espace le médecin peut trouver toutes les informations de son compte avec les horaires de son travail.

En cliquant sur l'icône (+) le médecin peut ajouter l'horaire à travers le remplissage d'un formulaire.

Pour supprimer l'horaire, le médecin peut cliquer sur l'icône de suppression.

Le médecin peut aussi modifier l'horaire en cliquant sur l'icône de modification.

DR Tamzini Hanen

Dashboard / Profil

Horraire

Jour	Heure début premier séance	Heure fin premier séance	Heure début deuxième séance	Heure fin deuxième séance	More
Mardi	08:30	12:30	15:00	17:30	
Mercredi	08:30	12:30	15:00	17:30	
Jeudi	08:30	12:30	15:00	17:30	
Vendredi	08:30	12:30	15:00	17:30	

Déconnexion

Figure 42 : Interface « Profil médecin » pour gérer l'horaire de travail

Horaire de travail

Jour

Heure debut 1er séance

Heure fin 1er séance

Heure debut 2ème séance

Heure fin 2ème séance

Figure 43 : Formulaire ajouter horaire de travail

Modifier l'horaire de travail de Mardi

Heure debut 1er séance

Heure fin 1er séance

Heure debut 2ème séance

Heure fin 2ème séance

Figure 44 : Formulaire modifier horaire de travail

➤ Interface des patients

Pour réserver un rendez-vous le médecin doit accéder à l'interface de la liste de ses patients.

Pour faciliter la recherche du patient, le médecin peut taper le nom de patient dans la zone du texte puis cliquer sur le bouton « Rechercher ».

Le patient clique sur l'icône « Calendrier » pour remplir le formulaire de rendez-vous

Rendez-vous

Jour

jj/mm/aaaa

Heure debut

Choisir

Fermer

Valider

Figure 45 : Formulaire ajouter un rendez-vous

Le médecin peut aussi réserver un rendez-vous à partir de l'espace des rendez-vous.

MEDICAL

Folders

Profil

Patients

Rendez vous

Consultations

Factures

Statistique

Archive

Déconnexion

DR
Tamzini
Hanan

Dashboard / Rendez vous

Mes rendez-vous

nom du patient

Rechercher

Refrécher

Etat rendez-vous :

Tout

Realiser

Annuler

En attente

Numéro dossier	Patient	Date rendez-vous	Heure rendez-vous	Acte
25	testFront testFront	2023-04-10	11:00	
16	foulen ben foulen	2023-04-11	16:30	✓
0	ben moad ali	2023-05-03	09:30	✓

Figure 46 : Interface des rendez-vous

44

2.1. Espace patient

The screenshot shows a web application interface for a patient's space. On the left is a sidebar with a logo and navigation links: Folders, Profil, Médecin, Rendez vous, and Historiques. The main content area displays information for 'haddad aya'. It includes a table with doctor details and another table showing the work schedule.

Nom et prenom	Telephone	Email	Nom cabinet
Tamzini Hanen	55303280	hanen@gmail.com	Hanen cabinet

Horaire de travail				
Jour	Heure debut premier S	Heure fin premier S	Heure debut deuxième S	Heure fin deuxième S
Mardi	08:30	12:30	15:00	17:30
Mercredi	08:30	12:30	15:00	17:30
Jeudi	08:30	12:30	15:00	17:30
Vendredi	08:30	12:30	15:00	17:30
Samedi	08:30	12:30		
Lundi	08:30	12:30	15:00	17:30

Figure 47 : Interface pour consulter la disponibilité du cabinet et les informations de médecin

Le patient a le droit de gérer ses rendez-vous à travers l'interface « Rendez-vous »

The screenshot shows the 'Mes rendez-vous' (My appointments) section of the patient interface. It features a search bar with a date format 'jj/mm/aaaa', a search button, and a refresh button. Below is a table with appointment status filters and a list of appointments.

Mes rendez-vous +

jj/mm/aaaa Rechercher Refrécher

Etat rendez-vous : Tout Realiser Annuler En attente

Numéro dossier	Patient	Date rendez-vous	Heure rendez-vous	Acte
2	haddad aya	2023-04-06	15:00	
2	haddad aya	2023-05-11	16:00	
2	haddad aya	2023-06-15	09:30	<input checked="" type="checkbox"/> <input type="checkbox"/>
2	haddad aya	2023-07-28	12:00	<input checked="" type="checkbox"/> <input type="checkbox"/>
2	haddad aya	2023-09-20	10:00	<input checked="" type="checkbox"/> <input type="checkbox"/>

Figure 48 : Interface des rendez-vous d'un patient

Conclusion

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

Dans notre suivant chapitre, nous allons établir le même travail pour le sprint 3 « Gestion des dossiers médicaux ».

Chapitre 5 : Sprint 3 : Gestion des dossiers médicaux

Introduction

Dans ce chapitre nous présenterons le déroulement pour développer la version qui concerne la gestion de des dossiers médicaux en suivons le même enchaînement que les deux chapitres précédents. Nous allons commencer par la présentation du backlog du notre troisième sprint. Ensuite nous passerons à la description fonctionnelle en présentant les diagrammes de cas d'utilisation et le diagramme des classes ainsi que la description comportementale à l'aide des diagrammes de séquences. Enfin, nous présenterons les interfaces de réalisation reliée à ce chapitre.

1 Backlog du sprint 3

Product Backlog (Gestion des dossier médicaux)				
Fonctionnalité	ID	User story	Tâches	Estimation
Gérer les consultations	1	En tant que médecin je peux ajouter/modifier les consultations au dossier médical de mon patient.	-Conception -Implémentation -Test	2 semaines
	2	En tant que médecin je peux ajouter/modifier des traitements dans la consultation de mon patient et générer son ordonnance en fichier PDF.		
	3	En tant que médecin je peux ajouter/modifier des fichier pendant la consultation de mon patient.		
Gérer les dossiers médicaux	1	En tant que médecin je peux consulter les dossiers médicaux de mes patients et le générer en fichier PDF.	-Conception -Implémentation -Test	5 jours
	2	En tant que médecin je peux archiver un dossier médical à travers le patient.	-Conception -Implémentation -Test	1 semaine
	3	En tant que patient je peux consulter mes dossiers médicaux et le générer en fichier PDF.	-Conception -Implémentation -Test	2 jours

Tableau 13 : Backlog du sprint 3

2 Description fonctionnelle des histoires du sprint 3

Dans cette partie, nous commençons par présenter le diagramme de classes et aussi le diagramme de cas d'utilisation global correspondant au sprint 3, par la suite, nous raffinerons par acteur.

2.1 Diagramme de classes

La figure ci-dessous représente le diagramme de classe du sprint 3.

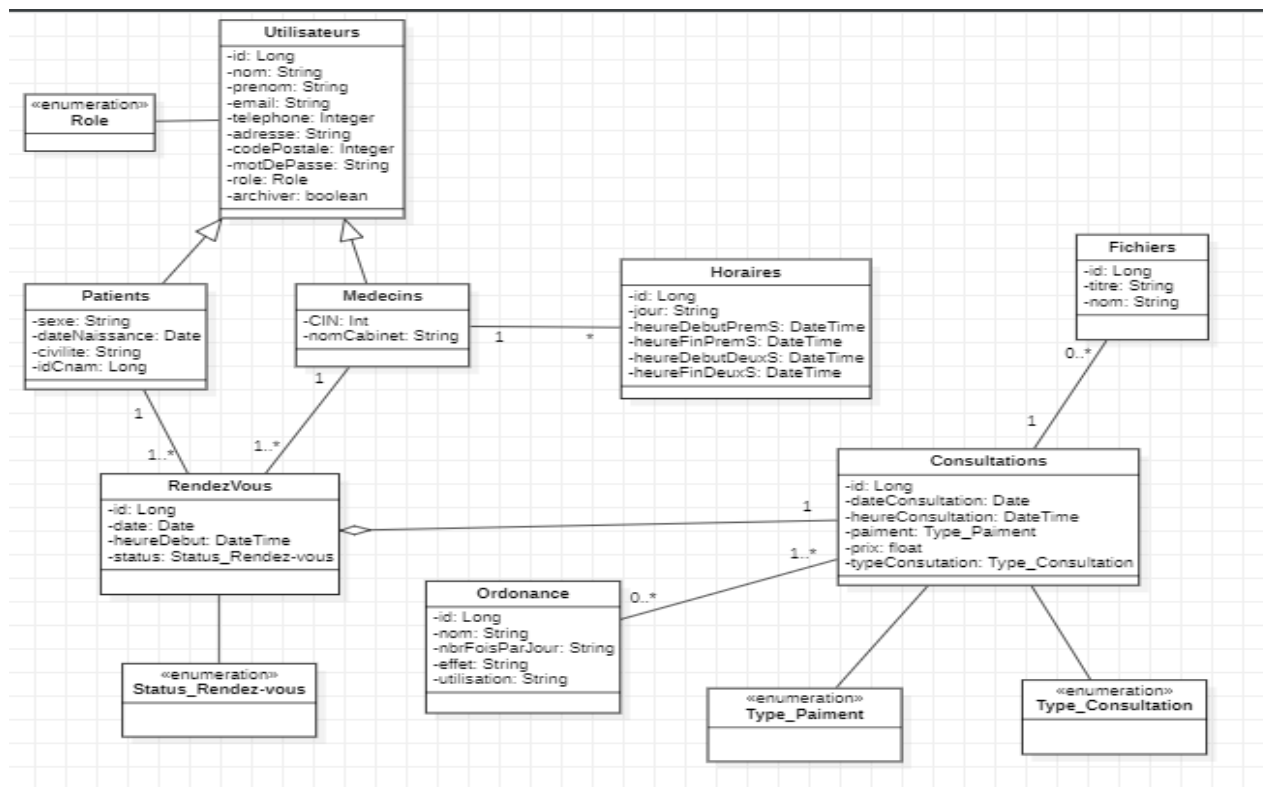


Figure 49 : diagramme de classe du sprint 3

2.2 Diagramme de cas d'utilisation

2.2.1 Diagramme de cas d'utilisation global du sprint 3

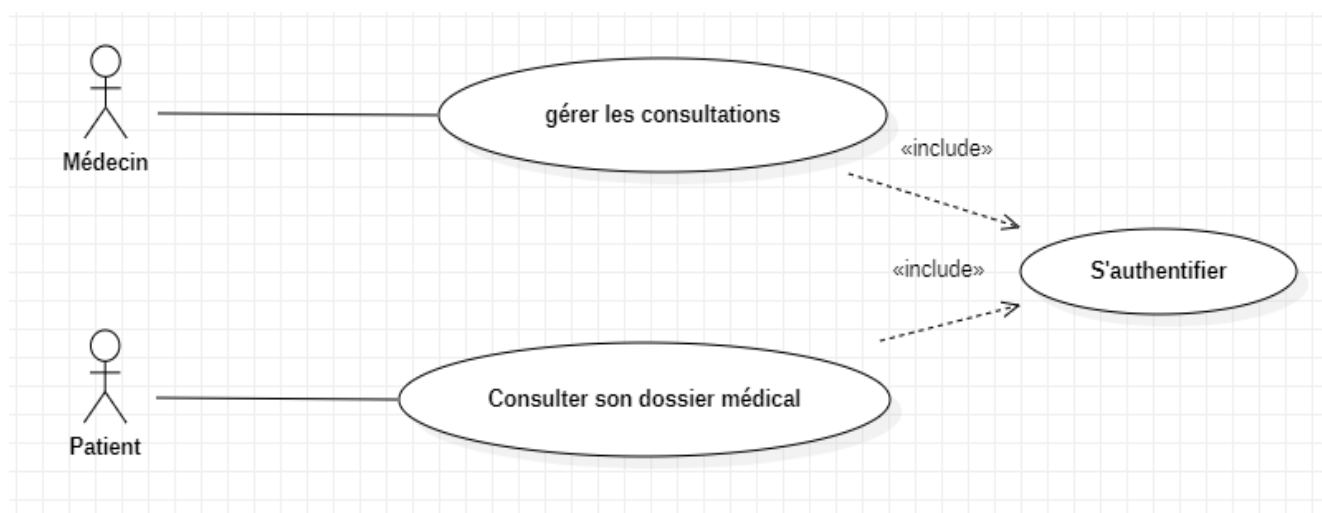


Figure 50 : Diagramme de cas d'utilisation global du sprint 3

2.2.2 Diagramme de cas d'utilisation détaillé du sprint 3

Après avoir présenté le diagramme de cas d'utilisation global de notre deuxième sprint, on va expliquer les fonctionnalités de chaque cas d'utilisation.

- **Diagramme de cas d'utilisation détaillé du médecin**
« Gérer les consultations »

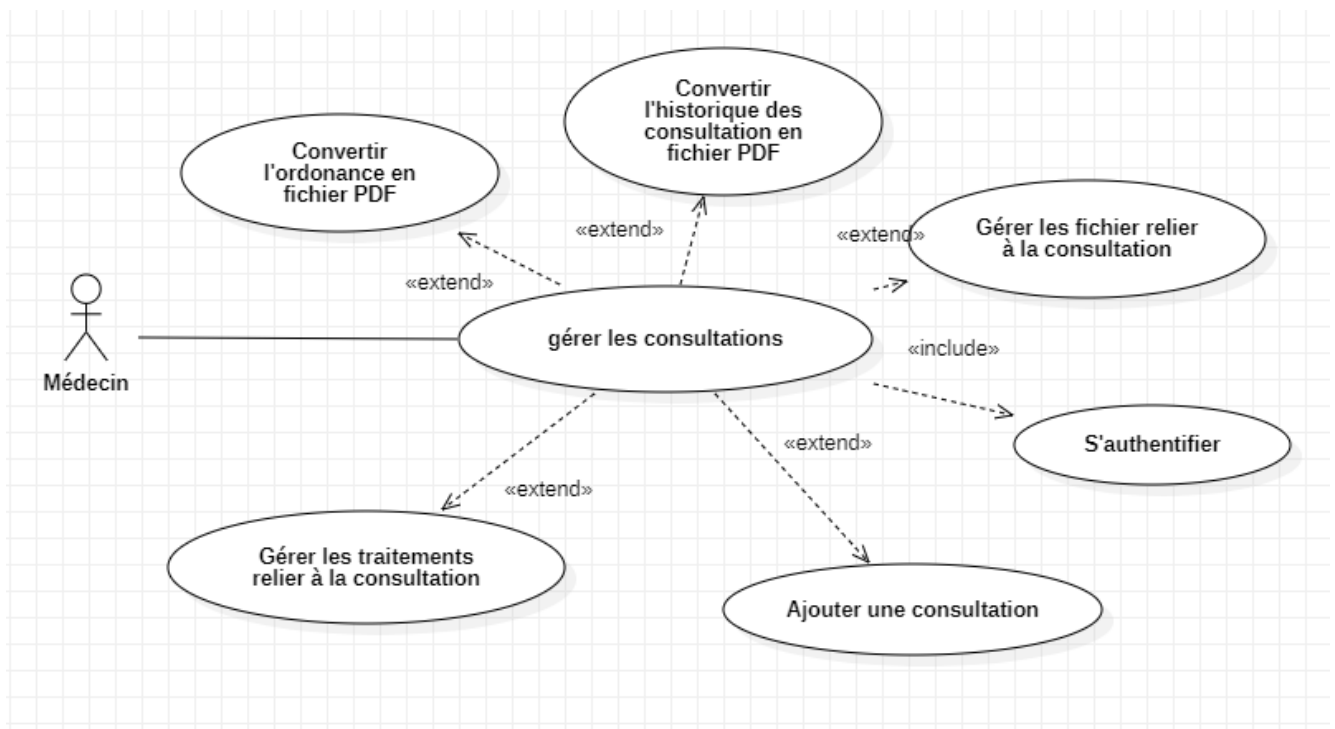


Figure 51 : Diagramme de cas d'utilisation détaillé du médecin « Gérer les consultations »

- **Diagramme de cas d'utilisation détaillé du patient**
« Consulter son dossier médical »

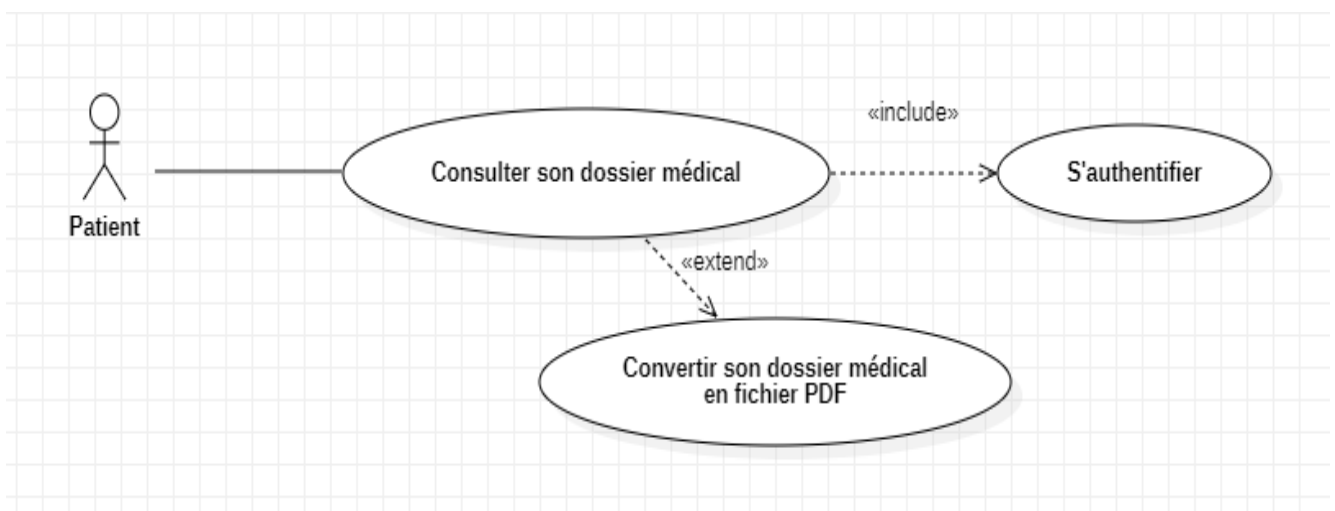


Figure 52 : Diagramme de cas d'utilisation détaillé du patient « Consulter son dossier médical »

2.3 Description textuelle des cas d'utilisations

Dans cette partie nous décrivons textuellement les principaux cas d'utilisation qui apparaissent dans le diagramme de cas d'utilisation global du sprint 3.

- Le tableau suivant décrit textuellement le cas d'utilisation gérer les consultations

Cas d'utilisation : Gérer les consultations	
Acteur(s)	Médecin
Objectif	Permettre au médecin de gérer les consultations.
Pré condition	Le médecin doit être authentifié.
Post condition	Gérer les consultations.
Scénario nominal	<ol style="list-style-type: none"> 1. Le système affiche la liste des rendez-vous. 2. Le médecin sélectionne le rendez-vous à réaliser. 3. Le système affiche l'interface de consultation qui contient un formulaire. 4. Le médecin remplit le formulaire avec les informations de la consultation qui peut contenir des fichiers et des traitements. 5. Le médecin génère l'ordonnance en fichier PDF. 6. Le médecin click sur le bouton de validation. 7. Le système ajout la consultation dans l'historique du patient. 8. Le médecin consulte les historiques de patient avec la possibilité de les télécharger en fichier PDF.
Scénario alternatif	Le médecin saisit des informations invalides. Le système affiche un message d'erreur.

Tableau 14 : Description textuelle du cas d'utilisation « Gérer les consultations »

- Le tableau suivant décrit textuellement le cas d'utilisation consulter son dossier médical

Cas d'utilisation : Gérer les consultations	
Acteur(s)	Médecin
Objectif	Permettre au patient de consulter son dossier médical.
Pré condition	Le patient doit être authentifié.
Post condition	Consulter les dossiers médicaux.

Scénario nominal	<ol style="list-style-type: none"> 1. Le système affiche l'interface des historiques. 2. Le patient télécharge ses historiques en fichier PDF.
Scénario alternatif	Le patient n'a pas encore passé une consultation.

Tableau 15 : Description textuelle du cas d'utilisation « Gérer son dossier médical »

3 Description comportementale des histoires du sprint

- Diagramme de séquence « Gérer les consultations »

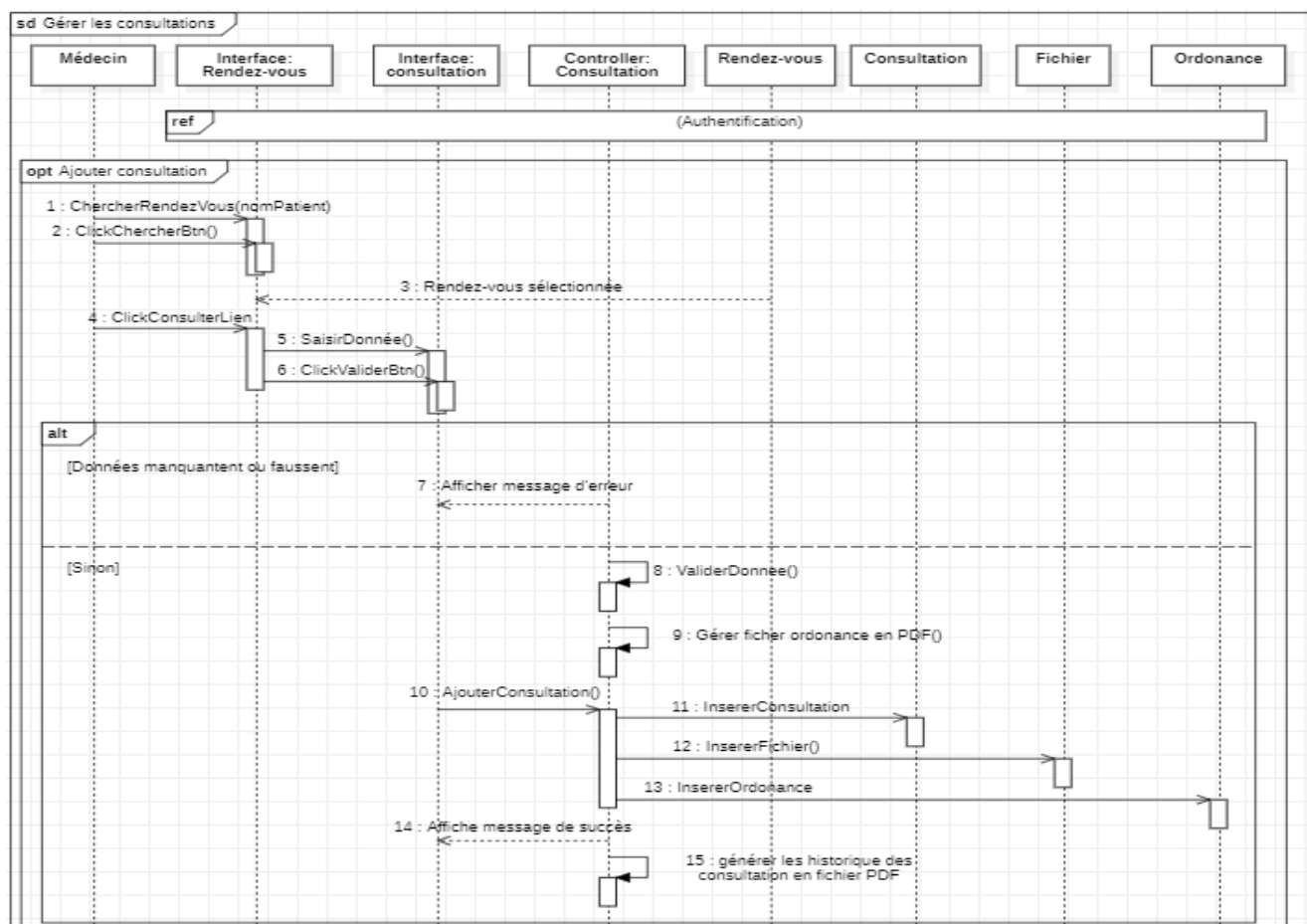


Figure 53 : Diagramme de séquence « Gérer les consultations »

4 Réalisation du sprint 3

Dans cette partie, nous présenterons les différents modules réalisés relatifs au troisième Sprint.

- Pour ajouter une consultation le patient doit avoir un rendez-vous, donc le médecin affichera la liste de ses rendez-vous « En attente » et cliquera sur le lien « Consulter »

Figure 56 : Formulaire fichiers

nom et prenom medecin
DOCTEUR
Hanen Tamzini
DIPLOME DE LA FACULTE DE MEDECINE DE TUNIS
MEDECINE GENERALE
Djerba sedouikech - 4541

Ordonance Médicale

Djerba le : 2023-06-14

Id.Cnam : 009654177530
wejden manita

Prescription :

1 / <u>Doliprane</u>	Qté : 2
4 fois par jour	
Effet : migrène	
2 / <u>Ibucare</u>	Qté : 2
3 fois par jour	
Effet : Douleurs	

E-mail: hanen@gmail.com Tél : 55303280

Fermer **Exporter en pdf**

Figure 57 : Ordonnance en HTML

Tamzini Hanen
DOCTEUR
Hanen Tamzini
DIPLOME DE LA FACULTE DE MEDECINE DE TUNIS
MEDECINE GENERALE
Djerba sedouikech - 4541

Ordonance Médicale

Djerba le : 2023-06-14

Id.Cnam : 009654177530
wejden manita

Prescription :

1 / Doliprane	Qté : 2
4 fois par jour	
Effet : migrène	
2 / Ibucare	Qté : 2
3 fois par jour	
Effet : Douleurs	

E-mail: hanen@gmail.com Tél : 55303280

Figure 58 : Ordonnance en PDF

- Le médecin peut gérer les consultations par l'interface de la liste des consultations.

Folders

Profil

Patients

Rendez vous

Consultations

Factures

Statistique

Archive

Déconnexion

Dashboard / Consultation

Mes consultations +

[Rechercher](#)
[Refrécher](#)

Etat paiement :

Tout
Non payé
Gratuit
Payé





ID dossier	Nom et Prenom	Contact	Plus
16	foulen ben foulen	Tel : 50366705 / Email : foulen@gmail.com	Détail consultation 
9	ben maad ala	Tel : 56122635 / Email : ala@gmail.com	Détail consultation 
1	benahmed ahmed	Tel : 54613250 / Email : anis@gmail.com	Détail consultation 
11	manita wejden	Tel : 56126035 / Email : wejden@gmail.com	Détail consultation 

Figure 59 : Interface pour gérer les consultations

- Pour consulter le dossier médical d'un patient et le générer en fichier PDF, le médecin doit afficher la liste des patients et cliquer sur le lien « Historique ».

Et le patient doit consulter l'interface « Historiques »

Dossier médical de ben foulen foulen

Numéro dossier	Nom et prénom	Date naissance	Email	numéro téléphone	Adresse	Code postal	ID cnss	Sexe	Civilité
16	foulen ben foulen	2005-04-11	foulen@gmail.com	50366705	djerba midoun	4584	9654190990	F	CELEBATAIRE

Numéro consultation	Date	Etat paiement	Prix	Type consultation	Traitements	Fichiers
1	2023-04-11	PAYE	25	CONTROLE	1 Nom traitement : panadole Quantité : 1 effet : migrène Nombre de fois : 2 par jours	1 Titre : scanner
					2 Nom traitement : Ibucare Quantité : 1 effet : douleurs Nombre de fois : 2 par jours	

[Télécharger ↓](#)

Figure 60 : Interface de dossier médical d'un patient

Dossier médical de *ben foulén foulén*

Numéro dossier	Nom et prénom	Date naissance	Email	numéro téléphone	Adresse	Code postal	ID cnss	Sexe	Civilité
16	foulén ben foulén	2005-04-11	foulén@gmail.com	50366705	djerba midoun	4584	9654190990	F	CELEBATAIRE

Numéro consultation	Date	Etat paiement	Prix	Type consultation	Traitements	Fichiers
1	2023-04-11	PAYE	25	CONTROLE	1 Nom traitement : panadole Quantité : 1 effet : migraine Nombre de fois : 2 par jours 2 Nom traitement : Ibucare Quantité : 1 effet : douleurs Nombre de fois : 2 par jours	1 Titre : scanner

Figure 61 : Fichier PDF de dossier médical

Conclusion

Dans ce dernier chapitre, nous nous sommes concentrés sur le développement de notre troisième et dernier sprint. Nous avons donc abouti à un produit fonctionnel, intégrale et livrable.

Conclusion générale

Ce présent rapport expose un projet réalisé durant la période de stage de fin d'études au sein de « cabinet Hanen » visant à mettre fin à la formation d'une licence appliquée en technologie d'informatique spécialité développement système d'information à l'institut supérieur des études technologiques de Djerba. Durant ce stage, nous avons développer un système de gestion d'un cabinet médical d'une plateforme Web.

Tout au long de ce projet, nous avons eu la chance d'élargir nos horizons et d'apprendre de nouveaux domaines et de nouvelles technologies et d'approfondir nos compétences techniques (recherche, codage, connaissances générales...). Nous avons une telle orientation qui nous a poussés à atteindre nos objectifs et nous considérons cette expérience comme une amélioration de nos compétences.

Ce projet nous a appris beaucoup de choses à reporter dans notre vie professionnelle et universitaire et tout cela a été une réalité générée par les efforts fournis.

Comme perspectives nous envisageons de connecter l'application avec les laboratoires, les pharmacies, les caisses nationale d'assurance-maladie (CNAM) et les hôpitaux.

Webographie

- [1] <https://www.redhat.com/fr/devops/what-is-agile-methodology> , Consulter le [28/04/2022]
- [2] <https://www.tuleap.org/fr/agile/comprendre-methode-agile-scrum-10-minutes> , Consulter le [28/04/2023]
- [3] <https://framalibre.org/content/visual-studio-code> , Consulter le [28/04/2022]
- [4] <https://www.baeldung.com/eclipse-sts-spring> , Consulter le [28/04/2023]
- [5] <https://www.akilischool.com/cours/creer-un-projet-laravel-avec-laragon> , Consulter le [28/04/2023]
- [6] <https://github-wiki-see.page/m/ImzagnanIlyas/mailbox/wiki/Langages-ettechnologies> , Consulter le [28/04/2023]
- [7] <https://www.atlassian.com/fr/git/tutorials/what-is-git> , Consulter le [28/04/2023]
- [8] <https://www.wikiwand.com/fr/StarUML> , Consulter le [28/04/2023]
- [9] [https://fr.wikipedia.org/wiki/Postman_\(logiciel\)](https://fr.wikipedia.org/wiki/Postman_(logiciel)) , Consulter le [28/04/2023]
- [10] <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203257- html5-hypertext-markup-langage5-definition-traduction/> , Consulter le [28/04/2023]
- [11] <https://www.futura-sciences.com/tech/definitions/internet-css-4050/> , Consulter le [28/04/2023]
- [12] <https://www.npmjs.com/package/typescript> , Consulter le [28/04/2023]
- [13] [https://fr.wikipedia.org/wiki/Java_\(technique\)](https://fr.wikipedia.org/wiki/Java_(technique)) , Consulter le [28/04/2023]
- [14] <https://www.oracle.com/ca-fr/database/what-is-json/> , Consulter le [28/04/2023]
- [15] <https://www.journaldunet.com/web-tech/developpeur/1159810-bootstrapdefinition-tutoriels-astuces-pratiques/> , Consulter le [28/04/2023]
- [16] <https://gayerie.dev/docs/spring/spring/introduction.html> , Consulter le [28/04/2023]
- [17] <https://www.bocasay.com/fr/developpement-application-angular/> , Consulter le [28/04/2023]
- [18] https://www.memoireonline.com/09/19/10896/m_Systeme-informatise-pour-la-planification-et-le-suivi-de-gestion-des-projets-des-etablissements-s29.html , Consulter le [29/04/2023]
- [19] <https://fr.wikipedia.org/wiki/MySQL> , Consulter le [29/04/2023]

- [20] <https://www.vaadata.com/blog/fr/jetons-jwt-et-securite-principes-et-cas-dutilisation/> , Consulter le [29/04/2023]
- [21] <https://adventy.org/fr/mvc> , [consulter le 05/05/2022]