

RAPPORT D'AVANCEMENT DE PROJET PERSONNEL ENCADRE **4^{ème} ANNEE INGENIEURS EN INFORMATIQUE**

Option : génie logiciel

Intitulé du stage
Conception et implémentation d'une application de gestion d'un cabinet de médecin pédiatre et d'analyse des données des patients.

Réalisé par
Mariem Taieb
Eya Guerfel
Hichem Ben Hamed
Encadré par
Abir Gorrab

Remerciements

Au terme de ce projet nous tenons à remercier infiniment tous les enseignants et administrateurs de l'École Supérieure privée des Sciences Appliquées et de Management.

Nos vifs remerciements à Madame Abir GORRAB, qui a gardé un œil attentif sur le déroulement du projet en donnant des remarques constructives. Nous le remercions pour sa disponibilité et son aide précieuse et cela a été un plaisir de travailler sous sa directive.

Nous exprimons également notre reconnaissance à tous les membres du jury pour avoir accepté de juger ce travail.

Table des matières

Introduction	1
Chapitre 1 Cadre général du projet	3
1.1 Présentation de cadre de projet	3
1.1.1 Etude de l'existant	3
1.1.2 Critique de l'existant :	6
1.1.3 Solution proposée :	7
1.2 Choix de méthodologie de conception	7
1.3 Methodologie Agile :	8
1.3.1 Méthodologie adoptée :	8
Chapitre 2 Sprint 0 : Analyse et spécifications des besoins	12
2.1 Spécification des besoins	12
2.1.1 Identification des acteurs	13
2.1.2 Besoins fonctionnels	13
2.1.3 Besoins non fonctionnels	13
2.1.4 Diagramme de cas d'utilisation	14
2.2 Pilotage du projet par Scrum	15
2.2.1 Equipe Scrum	15
2.2.2 Backlog du produit	15
2.3 Planification des sprints	18
2.4 Environnement de travail	19
2.4.1 Choix technologiques	19
2.5 Environnement logiciel	21
2.6 Méthodes et outils de modélisation	21
2.6.1 L'outil de modélisation	22
2.7 Architecture logicielle	22

Chapitre 3 Chapitre 3 : Réalisation du sprint 1	24
3.1 Spécification fonctionnelle	24
3.1.1 Diagramme de cas d'utilisation du premier sprint	26
3.1.2 Description des cas d'utilisation	27
3.2 Conception détaillée	30
3.2.1 Diagramme de séquence objet du cas d'utilisation « S'authentifier»	30
3.2.2 Diagramme de séquence objet du cas d'utilisation «Ajouter patient»	31
3.3 Réalisation	32
Chapitre 4 Chapitre 4 : Réalisation du sprint 2	39
4.1 Spécification fonctionnelle	39
4.1.1 Diagramme de cas d'utilisation du deuxième sprint	42
4.1.2 Description des cas d'utilisation	42
4.2 Conception détaillée	44
4.2.1 Diagramme de séquence objet du cas d'utilisation «Ajouter un rendez-vous»	44
4.2.2 Diagramme de séquence objet du cas d'utilisation «Annuler RDV»	44
4.3 Réalisation	45
Chapitre 5 Chapitre 5 : Réalisation du sprint 3	49
5.1 Spécification fonctionnelle	49
5.1.1 Diagramme de cas d'utilisation du troisième sprint	51
5.1.2 Description des cas d'utilisation	52
5.2 Réalisation	54
Chapitre 6 Chapitre 6 : Réalisation du sprint 4	58
6.1 Spécification fonctionnelle	58
6.2 Diagramme de cas d'utilisation du quatrième sprint	59
6.3 Réalisation :	60
6.3.1 Analyser les données :	60
6.3.2 Gestion et suivi du dossier médical	62
Conclusion générale	64
Netographie	65

Introduction

Durant ces dernières années l'informatique s'est imposée d'une manière impressionnante dans tous les domaines , cela est du à son apport extraordinaire dans le domaine de gestion des bases de données.

Au cours de ce projet, nous nous intéressons au domaine médicale. La démarche médicale est fondée sur l'observation du malade, la mémoire du médecin était autrefois suffisante pour enregistrer les données relatives aux patients et servir l'exercice médical, les données médicales étaient rassemblées sous forme d'articles médicaux, de registres à visée épidémiologique, nosologique et administrative, avec la multiplication des effets de l'environnement, de nos jours la bonne tenue d'un dossier exige des moyens informatiques.

L'automatisation du système d'information consiste à structurer et gérer un ensemble de données dont le but est de les organiser et d'avoir des résultats rapides.

Dans ce cadre, nous sommes appelés à concevoir, développer et mettre en place une application de gestion d'un cabinet de médecin pédiatre et d'analyse des données des patients.

Le logiciel devrait mettre l'organisation et l'automatisation de la gestion d'un cabinet médical, afin d'augmenter la fiabilité, l'efficacité de l'effort humain et faciliter les tâches pénibles au sein du cabinet.

Ainsi, ce présent document est préparé, afin de vous donner une synthèse de nos travaux. Les résultats de nos travaux sont présentés dans ce rapport qui comportera quatre chapitres.

Le premier chapitre dédié au cadre général qui comporte une présentation de notre projet avec une présentation de la méthodologie de travail "Scrum" utilisée pour la réalisation de ce projet.

Dans un deuxième chapitre, nous nous sommes aussi intéressées à présenter le sprint zéro qui présente la phase initiale pour le démarrage d'un projet Scrum. Dans cette étape nous partageons une vision claire du projet en présentant la définition des besoins fonc-

tionnels et non fonctionnels, suivie par le backlog produit et la planification des sprints.

Le troisième chapitre, qui est consacré à la réalisation de la première partie de notre étude, s'intéresse à l'authentification, Gérer les patients, Gérer les comptes.

Dans le quatrième chapitre, nous intéressons à la réalisation du deuxième sprint qui est consacré à la deuxième partie de notre étude qui est la gestion des rendez-vous, des couloirs et des chefs couloirs.

1

Cadre général du projet

Introduction

A travers ce premier chapitre, nous allons présenter une étude de l'existant ainsi que la problématique qui a donné naissance à notre projet. Nous allons indiquer aussi la méthodologie de travail utilisée pour aboutir à la réalisation d'un projet qui répond aux besoins demandées.

1.1 Présentation de cadre de projet

L'étude des applications existantes nous permettra de spécifier notre projet et identifier quelques critiques à partir de lesquelles nous présenterons notre solution proposée.

1.1.1 Etude de l'existant

Dans cette partie, nous présentons l'étude de l'existant en domaine d'applications médicales, nous allons essayer de faire une étude comparative des logiciels qui existent au marché. Cette étude permettra de :

- S'inspirer des expériences similaires à notre projet et qui ont déjà réalisées. Cette inspiration sera consacrée dans l'outil qu'on devra réaliser.
- Localiser notre système par rapport aux systèmes déjà existants
- Préciser ce que notre application doit faire.

Plusieurs applications existent aujourd’hui dans le domaine médical pour faciliter par exemple la tâche de prise des rendez-vous . Nous avons choisi les applications les plus connus et utilisées en Tunisie et en France.

Med.tn :

Med.tn est une plateforme innovante qui permet de trouver rapidement un médecin le plus proche de chez vous et de prendre rendez-vous en ligne gratuitement.

Comme elle permet aux utilisateurs de poser leurs questions en ligne directement aux médecins en respectant les termes de confidentialité. [URL1]



FIGURE 1.1 – Interface de l’application Med.tn

Doctolib :

Doctolib est une entreprise franco-allemande qui propose un service de gestion en ligne des consultations pour les professionnels de santé et un service de prise de rendez-vous

en ligne pour les patients. Elle a comme objectif aussi de permettre aux utilisateurs de trouver un centre de vaccination proche de chez eux et prendre un rendez-vous de manière simple en ligne ou par téléphone. [URL2]



FIGURE 1.2 – Interface de l'application Doctolib

Rdvmédicaux

La mission de cette application consiste à offrir aux patients une solution simplissime pour prendre rendez-vous avec leurs médecins depuis un ordinateur ou un smartphone. RDVmedicaux bénéficie d'un large savoir-faire en terme de prise de rendez-vous qui lui permet de gérer tous les cas particuliers, notamment grâce à une stratégie de développement étroitement liée à celle des permanences téléphoniques. [URL3]



FIGURE 1.3 – Interface de l’application Rdvmédicaux

En se basant sur les fonctionnalités décrites dans notre cahier des charges, nous avons réalisé une comparaison entre ces produits qui a pour but de mieux comprendre l’existant.

Fonctionnalités	Med.tn	Doctolib	RDVmédicaux
Gestion et Suivi du Dossier Médical :	Non	Non	Non
Gestion des Rendez-vous :	Oui	Oui	Oui
Gestion de la Fiche Patient :	Non	Non	Non
Suivre le planning de vaccination :	Non	Non	Non
Gestion des Ordonnances :	Non	Non	Non
Analyse des données des patients :	Non	Non	Non

1.1.2 Critique de l’existant :

L’analyse de l’existant nous a permis de savoir les insuffisances des solutions existantes. Nous prenons des exemples des problèmes rencontrés :

- Les applications existantes se limite seulement au prise des rendez-vous et la recherche des médecin.

- il existe des applications desktop spécifique pour un cabinet d'un pédiatre mais avec des fonctionnalités basiques.

1.1.3 Solution proposée :

Dans le secteur médical, où l'encombrement des patients ainsi que leurs dossiers poussent plusieurs médecins à mal gérer leurs activités dans leurs cabinets, alors nous proposons de développer une application d'un cabinet pédiatre qui est destinées à la gestion d'un cabinet médical et au suivi des patients. Ainsi nous allons essayer de répondre à la problématique suivante : "Quel est l'impact d'une mise en place d'une application web sur la gestion d'un cabinet médical et sur le suivi des patients, au niveau de la performance et de la sécurité des informations ?".

L'application a un impact positif sur la performance du cabinet. Elle a comme objectif l'amélioration de la sécurité des informations ainsi que la communication interne et externe du cabinet. La mise en place de cette application améliore aussi l'image marketing du cabinet.

1.2 Choix de méthodologie de conception

Le terme "agile" définit une approche de gestion de projet qui prend le contre-pied des approches traditionnelles prédictives et séquentielles de type cycle en V ou waterfall (en cascade). La notion même de "gestion de projet" est remise en question au profit de "gestion de produit". De façon à raisonner davantage "produit" que "projet". Après tout l'objectif d'un projet consiste bien à donner naissance à un produit. [1]

Une approche dite "traditionnelle" attend généralement du client une expression détaillée et validée du besoin en entrée de réalisation, laissant peu de place au changement. La réalisation dure le temps qu'il faut et le rendez vous est repris avec le client pour la recette. Cet effet tunnel peut être très néfaste et conflictuel, on constate souvent un déphasage entre le besoin initial et l'application réalisée. On se rapporte alors aux spécifications validées et au contrat. Certains projets se terminent dans la douleur (surtout dans le cadre d'un contrat au forfait classique) au risque de compromettre la relation client. De plus il n'est pas rare que certaines fonctionnalités demandées se révèlent finalement inutiles à l'usage alors que d'autres, découvertes en cours de route, auraient pu donner plus de valeur au produit.[1]

Pour la performance de notre application nous avons choisi la méthode agile pour le déroulement du projet.

1.3 **Methodologie Agile :**

La méthodologie Agile se base sur ce principe simple : planifier la totalité de votre projet dans les moindres détails avant de le développer est contre-productif.[2]

La méthode Agile recommande de se fixer des objectifs à court terme. Le projet est donc divisé en plusieurs sous-projets. Une fois l'objectif atteint, on passe au suivant jusqu'à l'accomplissement de l'objectif final. Cette approche est plus flexible. Puisqu'il est impossible de tout prévoir et de tout anticiper, elle laisse la place aux imprévus et aux changements. Elle repose sur une relation privilégiée entre le client et l'équipe projet. La satisfaction du client étant la priorité, l'implication totale de l'équipe et sa réactivité face aux changements du client comme aux imprévus sont nécessaires. Le dialogue avec le client est privilégié. C'est lui qui valide chaque étape du projet. L'évolution de ses besoins est prise en compte et les ajustements sont effectués en temps réel afin de répondre à ses attentes. [1]

1.3.1 **Méthodologie adoptée :**

Pour le bon déroulement des différentes phases de notre projet nous avons opté pour la méthode agile Scrum.

Le principe de base étant que l'équipe avance ensemble et soit toujours prête à réorienter le projet au fur-et-à-mesure de sa progression. Les concepts de la méthode Scrum sont les suivants : [3]

Sprint :

un sprint désigne le délai réel dont l'équipe Scrum a besoin pour terminer un incrément. La longueur classique d'un sprint varie entre deux semaines quatre semaines au maximum.

Daily Scrum :

C'est une mini-réunion quotidienne qui a lieu à la même heure (généralement, le matin) et au même endroit pour simplifier les choses. Cette réunion est également appelée « stand-

up quotidien » pour souligner le fait qu'elle doit être rapide. L'objectif du « Daily Scrum » est de mettre tous les membres de l'équipe en phase avec l'objectif du sprint et de définir un plan pour les prochaines 24 heures.

Revue de sprint :

à la fin du sprint, l'équipe se rassemble pour une session informelle afin d'assister à une démo de l'incrément ou de l'inspecter. Cette revue est également l'occasion pour le Product Owner d'apporter des modifications au backlog produit sur la base du sprint actuel.

Rétrospective de sprint :

est l'occasion pour l'équipe de se rassembler afin de documenter ce qui a fonctionné ou non dans un sprint, un projet, des relations, des outils, chez des personnes, voire dans certaines cérémonies et d'en discuter.

Trois rôles essentiels pour la réussite de Scrum :

Les Product Owners sont les responsables de leur produit. Leur priorité est de comprendre les exigences du business, des clients et du marché, puis de prioriser le travail de l'équipe d'ingénierie en conséquence.

- créent et gèrent le backlog produit.
- travaillent en étroite collaboration avec le business et l'équipe pour s'assurer que tous savent en quoi consistent les tâches du backlog produit.
- fournissent à l'équipe des orientations claires sur les prochaines fonctionnalités à livrer.
- décident du moment où le produit doit être livré, avec une prédisposition pour des livraisons plus fréquentes.

Le Scrum Master Il coache l'équipe, les Product Owners et le business sur le processusScrum et cherchent les façons d'affiner leur pratique en la matière.

- Un Scrum Master efficace comprend parfaitement le travail que l'équipe doit réaliser et peut aider celle-ci à optimiser la transparence et le flux de livraison. En tant que chef d'orchestre, il prévoit les ressources nécessaires (humaines et logistiques) pour planifier le sprint, le stand-up, la revue et la rétrospective de sprint.

L'équipe de développement Scrum : Constituée de 3 à 9 personnes, elle permet de privilégier un rythme de livraison significatif tout en évitant une organisation trop complexe. De plus, elle contribue à produire un résultat à chaque sprint : un incrément de produit.

Cycle de vie de la méthode Scrum

[5]

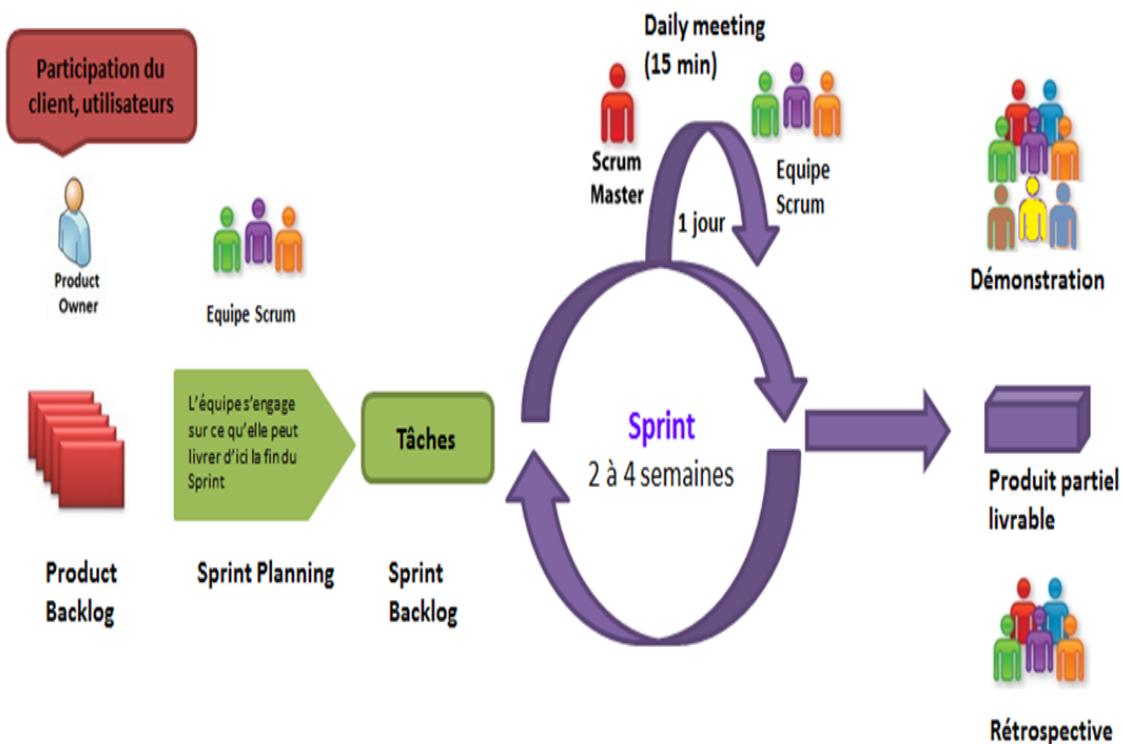


FIGURE 1.4 – Processus Scrum

La méthode Scrum suit un processus itératif permettant d'obtenir un produit proche des besoins client en prenant en compte l'évolution de ces derniers et ainsi maximiser la valeur du produit livré. Si au cours d'un Sprint, un problème survient, la responsabilité n'incombe pas à une seule personne mais elle est partagée entre le Product Owner, le Scrum Master et l'équipe Scrum. [4]

Conclusion

Dans ce premier chapitre introductif, nous avons présenté le cadre général du projet ainsi qu'une étude de l'existant. Nous avons aussi proposé une solution envisagée pour faire face à la situation courante. Ensuite, on a indiqué la méthodologie de conception qui sera utilisée dans notre projet. Dans le chapitre suivant nous détaillerons le sprint 0, qui s'intéresse à l'analyse et la spécification des besoins.

2

Sprint 0 : Analyse et spécifications des besoins

Introduction

Dans le chapitre précédent, nous avons identifié les problématiques du projet et nous avons présenté notre choix de méthodologie de gestion de travail. Ce chapitre présente l'étude des besoins qui constitue une phase d'analyse du projet. Nous allons présenter tout d'abord, les acteurs principaux de l'application ainsi qu'identifier les besoins fonctionnels et non fonctionnels de l'application pour bien analyser les problèmes à résoudre. Puis, nous allons proposer une modélisation du diagramme de cas d'utilisation global du projet. Ensuite, nous allons exposer l'équipe Scrum ainsi que le Backlog de produit et la planification des sprints. Enfin, nous allons clôturer le chapitre par présentation des choix technologiques et l'architecture adopté.

2.1 Spécification des besoins

La partie d'identification des besoins consiste à traduire les objectifs du projet en un ensemble de fonctionnalités ciblées par l'outil à réaliser. Ceci procurera une compréhension plus approfondie des tâches à mettre en œuvre. La spécification des besoins s'agit d'identifier les acteurs, définir les besoins fonctionnels et non fonctionnels et de déterminer les cas d'utilisation initiaux.

2.1.1 Identification des acteurs

Un acteur est une entité externe au système. Il représente une personne ou un autre système informatique qui attend un ou plusieurs services offertes par une interface d'accès. Il interagit avec le système par envoi ou réception des messages. Par ailleurs, notre application va intervenir les différents acteurs suivants :

- Médecin : C'est le responsable de l'application. Il a pour tâche de gérer tout le système. Il a une visibilité totale sur la base de données.
- Secrétaire : c'est la responsable de gérer les rendez-vous, les fiches patients et suivre le planning des vaccinations
- Parent : c'est le responsable de consulter les dates de vaccinations de ces enfants.

2.1.2 Besoins fonctionnels

Les besoins fonctionnels sont l'ensemble des besoins que le système doit effectuer. Les exigences fonctionnelles décrites dans cette partie correspondent aux fonctions les plus couramment exigées par la pédiatrie. Cette application doit permettre à tous les utilisateurs de réaliser à distance les opérations suivantes :

- Authentification : Les parents peuvent créer un compte, le médecin peut ajouter un compte pour la secrétaire. Chaque utilisateur (médecin, secrétaire, parent), possède un login et un mot de passe spécifique qui lui permet de vérifier son identité, afin d'autoriser l'accès de cette entité à des ressources en toute sécurité.

Il permet ainsi d'offrir au médecin et au secrétaire la possibilité de :

- Gérer un rendez-vous : Le médecin et la secrétaire peuvent supprimer ou modifier les rendez-vous.

les Parents qui sont les propriétaires des patients, le système doit les offrir la possibilité :

- Gérer un patient : Le parent peut ajouter, modifier, supprimer les coordonnées de ses enfants.
- Ajouter un rendez-vous : Le parent peut prendre un rendez-vous pour ses enfants.

2.1.3 Besoins non fonctionnels

Les besoins non fonctionnels présentent les exigences internes pour le projet et sont cachées vis-à-vis des clients. La solution doit prendre en considération certaines caractéristiques afin de satisfaire l'utilisateur. Par conséquent, la demande sera conçue autour

des qualités suivantes :

- Disponibilité : L'application doit être disponible à tout moment.
- Rapidité : L'application ne doit pas être lente. En Utilisant le Framework react nous allons limiter les appels serveurs coûteux en termes de mémoire et de temps de réponse, pour assurer la rapidité de l'application.
- L'ergonomie : En termes d'ergonomie, l'interface utilisateur doit être Simple et facile à manipuler

2.1.4 Diagramme de cas d'utilisation

Nous donnons ci-dessous une vue globale concernant le comportement fonctionnel du système. Ce diagramme permet de représenter les interactions entre les acteurs et les cas d'utilisation du système. La figure 2.1 montre le diagramme de cas d'utilisation global.

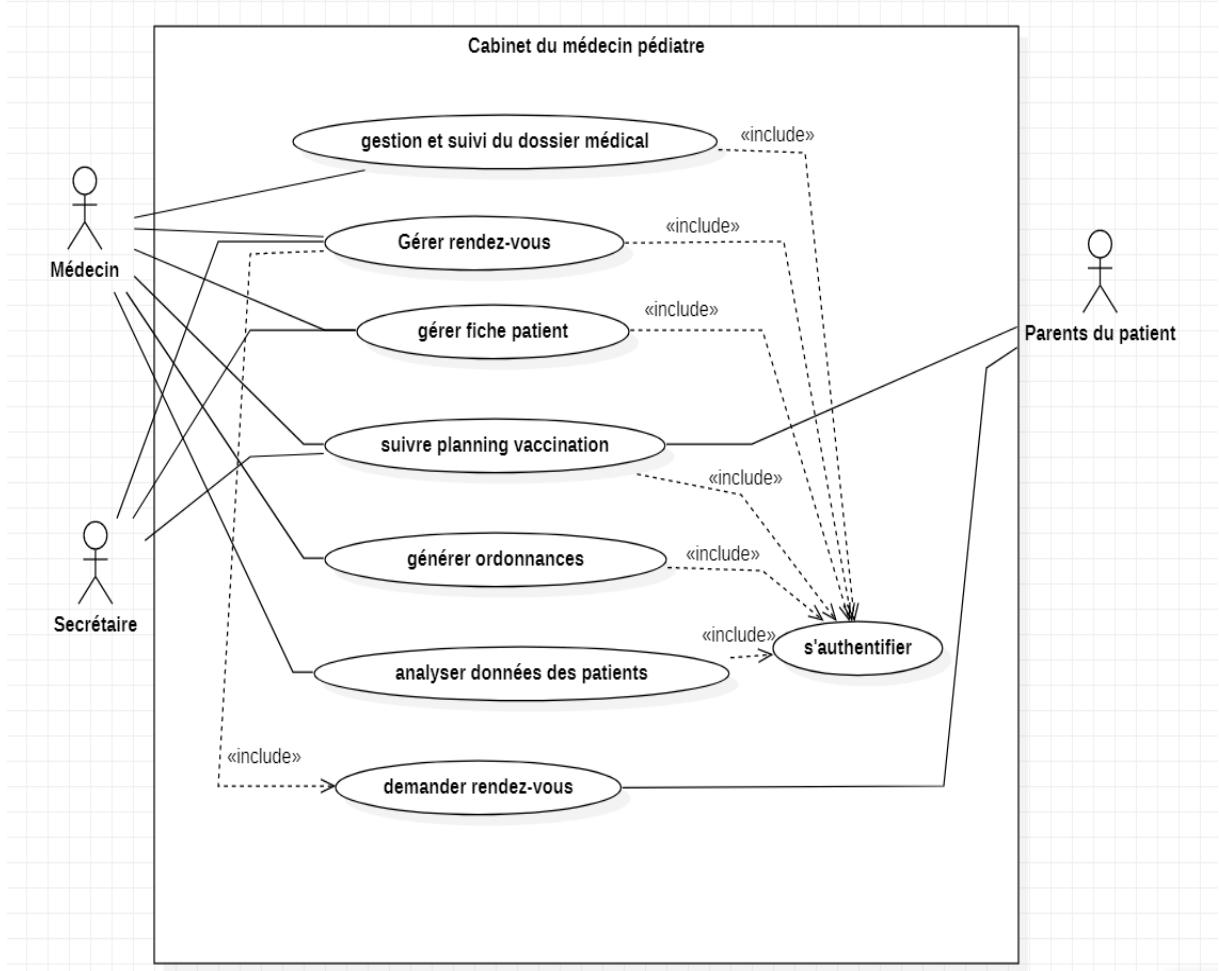


FIGURE 2.1 – le diagramme de cas d'utilisation globale

2.2 Pilotage du projet par Scrum

Une bonne planification du travail est très importante pour aboutir à un bon projet. Dans cette partie, nous allons identifier les membres de l'équipe, élaborer le backlog du produit et présenter le découpage de notre projet.

2.2.1 Équipe Scrum

La méthodologie agile de Scrum nécessite la présence de plusieurs intervenants pour une bonne gestion du projet. Une hiérarchie de l'équipe est présentée dans la figure 6.

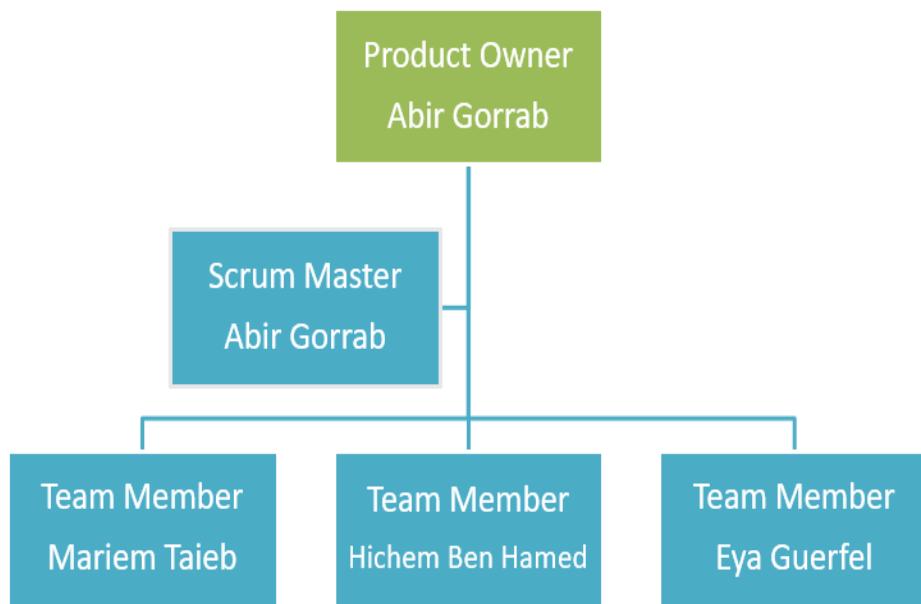


FIGURE 2.2 – Hiérarchie de l'équipe Scrum

2.2.2 Backlog du produit

La méthodologie Scrum utilise un outil sous la forme d'un tableau pour collecter les spécifications fonctionnelles du produit : Cet outil est appelé "Product Backlog". C'est à travers le Product Backlog que s'établit une vue générale des fonctions à réaliser. À chaque sprint, l'équipe de développement part d'une partie de fonctionnalités extraites du Product Backlog afin de les réaliser et livrer à sa fin un produit contenant ces fonctionnalités. Par la suite une description des termes utilisés dans le Product Backlog.

- Fonctionnalité : Cette colonne contient les besoins fonctionnels initiaux fixés auparavant.

- ID Story : C'est la numérotation des fonctionnalités raffinées.
- User Story : Cette colonne décrit le QUI et le QUOI. Elle présente la fonctionnalité et qui va l'exécuter.
- Priorité : L'équipe accorde des priorités aux différentes fonctionnalités.

Le tableau 2.1 représente le Product Backlog de notre projet.

ID	Fonctionnalité	User story	User Story	priorité
1	Authentification	1.1	En tant qu'utilisateur (Médecin, secrétaire, parent) je souhaite m'authentifier pour accéder à l'application.	1
2	Gérer les patients	2.1	En tant que parent je souhaite ajouter un patient	1
		2.2	En tant que parent je souhaite Supprimer un patient	1
		2.3	En tant que Parent je souhaite modifier un patient	1
		2.4	En tant que parent je souhaite consulter un patient	1
		2.5	En tant que utilisateur je souhaite Rechercher un patient par leur nom	1
3	Gérer les comptes	3.1	En tant que utilisateur je souhaite ajouter un compte.	1
		3.2	En tant que medecin je souhaite Supprimer un patient	1
		3.3	En tant que médecin je souhaite modifier un patient	1
		3.4	En tant que médecin je souhaite gérer les rôles	1

ID	Fonctionnalité	User story	User Story	priorité
4	Consulter le calendrier des rendez-vous	4.1	En tant que Secrétaire, Médecin je souhaite Confirmer les rendez-vous.	2
		4.2	En tant que Secrétaire, Médecin je souhaite Intégrer FullCalendar	2
5	Gérer les rendez-vous	5.1	En tant que secrétaire je souhaite Confirmer les rendez-vous.	2
		5.2	En tant que parent je souhaite ajouter un rendez-vous	2
		5.3	En tant que secrétaire et parent je souhaite Supprimer un rendez-vous	2
		5.4	En tant secrétaire et parent je souhaite modifier les rendez-vous	2
6	Générer fiche patient	6.1	En tant que médecin ou secrétaire je souhaite ajouter les coordonnées d'un patient.	2
		6.2	En tant que medecin, secrétaire je souhaite Modifier les coordonnées d'un patient	2
		6.3	En tant que medecin ou secrétaire je souhaite Supprimer les coordonnées d'un patient	2
		6.4	En tant que medecin, secrétaire ou parent je souhaite Consulter les coordonnées d'un patient	2
7	Générer les vaccins	7.1	En tant que médecin je souhaite modifier un vaccin.	3
		7.2	En tant que médecin je souhaite consulter les vaccins pour chaque patient.	3

8	Gérer planning vaccinations	8.1	En tant que parent je souhaite recevoir une notification par email.	3
		8.2	En tant que parent je souhaite recevoir une notification à travers un message téléphonique.	3
9	Gérer les ordonnances	9.1	En tant que médecin je souhaite ajouter une ordonnance.	3
		9.2	En tant que médecin je souhaite supprimer une ordonnance.	3
		9.3	En tant que médecin je souhaite Imprimer une ordonnance sous la forme PDF.	3
10	Analyser les données	10.1	En tant que médecin je souhaite consulter les statistiques des maladies reconnues	4
11	Gestion et suivi du dossier médical	10.2	En tant que médecin je fais le suivi le dossier médical d'un patient	4

TABLE 2.1 : Product Backlog

2.3 Planification des sprints

Nous illustrons par la figure 2.2 le cycle de gestion du projet suivi tout au long de la réalisation de notre projet.

Sprint 1 : (DE 24/03/2021 à 15/04/2021)	Sprint 2 : (DE 16/04/2021 à 08/05/2021)	Sprint 3 : (DE 10/05/2021 à 31/05/2021)	Sprint 4 : (DE 01/06/2021 à 12/06/2021)
Authentification Gérer les patients Gérer les comptes demander rendez-vous	Consulter le calendrier des rendez-vous Gérer les rendez-vous Générer fiche patient	Générer les vaccins Gérer planning vaccinations Gérer les ordonnances	Analyser les données gestion et suivi du dossier médical

Figure 2.2 planification des sprints

2.4 Environnement de travail

Dans cette partie, nous allons préciser les technologies utilisées au cours de ce projet.

2.4.1 Choix technologiques

Cette section est divisée sur deux parties, une première partie qui se focalise sur les technologies utilisées pendant le développement du Back-End et une deuxième partie qui est consacré à la présentation des différentes technologies utiliser pour crée les interfaces utilisateurs.

Les choix des frameworks Back-end

— Node.js

Description : est une plateforme de développement JavaScript intégrant un serveur HTTP. Son fonctionnement se base sur une boucle événementielle qui lui permet le support de fortes montées en charge. Caractérisée comme étant une bibliothèque de ce langage, elle permet la réalisation d'actions comme créer un fichier ou bien ouvrir et fermer des connections réseau. [6]

— Mongodb

Description : Est un système de gestion de base de données orienté documents, répartissables sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini de données. Il est écrit en C++. Le serveur et les outils sont distribués sous licence SSPL(Server Side Public License), les pilotes sous licence Apache et la documentation sous licence Créative. [7]

— **Express**

Description : est une infrastructure de serveur d'applications Web Node js, spécialement conçue pour la création d'applications Web à une page, à plusieurs pages et hybrides (Les applications hybrides sont des applications disponibles sur les App stores et qui s'installent sur un device, comme n'importe quelle autre application). Il est devenu le cadre de serveur standard pour node.js. Express est la partie backend de quelque chose connu sous le nom de pile MERN. [8]

- MongoDb => La base.
- Express.js => Le cadre d'applications Web par défaut.
- React js => Le framework JavaScript MVC utilisé pour les applications Web.

Les choix des frameworks front-end

— **React JS**

Description : React est une bibliothèque qui ne gère que l'interface de l'application, considéré comme la vue dans le modèle MVC (Modèle-vue-contrôleur). Elle peut ainsi être utilisée avec une autre bibliothèque ou un framework MVC comme AngularJS. La bibliothèque se démarque de ses concurrents par sa flexibilité et ses performances, en travaillant avec un DOM (Document Object Model) virtuel et en ne mettant à jour le rendu dans le navigateur qu'en cas de nécessité. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage. [9]

— **Material Design**

Description : Le Material Design est un ensemble de règles de design proposées par Google et qui s'appliquent à l'interface graphique des logiciels et applications. Il est utilisé notamment à partir de la version 5.0 du système d'exploitation Android. [10]

2.5 Environnement logicielle

Dans cette partie, nous allons définir les outils matériels exigés pour le développement ainsi que les outils logiciels utilisés Afin de développer notre plateforme nous avons eu recours à :

— Visual Studio Code

Est un éditeur de code source léger mais puissant qui fonctionne sur votre bureau et est disponible pour Windows, MacOs et Linux. Il est livré avec un support intégré pour JavaScript, Type Script et Node.js et dispose d'un riche écosystème d'extensions pour d'autres langues (C++, C#, Python PHP, Go) et les temps d'exécution (tels que .NET et Unity). [11]

— Git

Est un système de contrôle de versions pour le développement de logiciels qui a été initialement conçu et développé par Linus Torvalds, l'auteur du noyau Linux. Il permet et encourage à avoir plusieurs dépôts locaux pouvant être entièrement indépendantes les unes des autres. La création, la fusion et la suppression de ces lignes de développement se fait en quelques secondes. Le Git facilite le travail en groupe grâce aux pull requests qui simplifient les revues de code, ce qui garantit un code de meilleure qualité et permet de partager des connaissances dans l'équipe.[12]

2.6 Méthodes et outils de modélisation

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

2.6.1 L'outil de modélisation

StarUML est un logiciel de modélisation UML, cédé comme open source par son éditeur.

StarUML gère la plupart des diagrammes spécifiés dans la norme UML 2.0.

2.7 Architecture logicielle

Dans cette partie, nous présentons l'architecture logicielle de notre application qui s'intéresse au découpage logique de l'application et la façon de regrouper les composants selon le type de fonction et traitements qu'ils effectuent. Pour notre solution nous avons opté pour une architecture MVC illustrée par la figure 2.5.

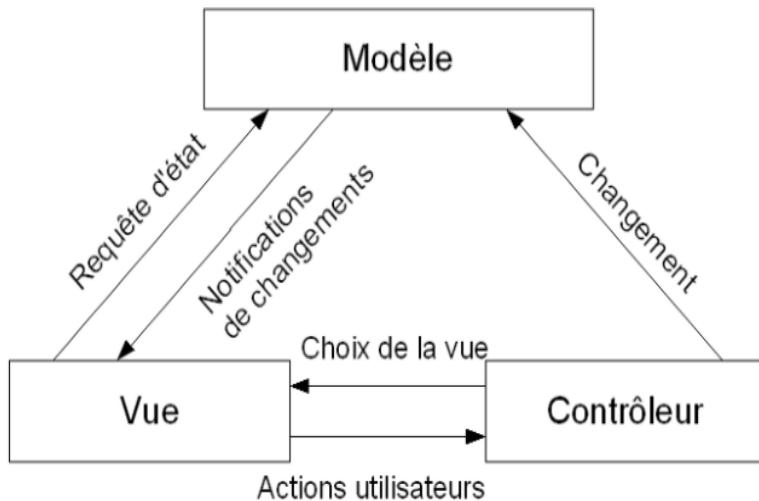


FIGURE 2.3 – L'architecture MVC

Dans notre application nous avons choisi d'utiliser une architecture MVC dans le but d'attribuer des responsabilités. Définir des couches est un moyen de séparer les responsabilités et ainsi, de minimiser l'impact du changement.

- Modèle : noyau de l'application qui gère les données, permet de récupérer les informations dans la base de données, de les organiser pour qu'elles puissent ensuite être traitées par le contrôleur.
- Vue : composant graphique de l'interface qui permet de présenter les données du modèle à l'utilisateur.

- Contrôleur : composant responsable des prises de décision, gère la logique du code qui prend des décisions, il est l'intermédiaire entre le modèle et la vue.

Conclusion

Dans ce chapitre, nous avons passé en revue par les différentes notions nécessaires à la compréhension de notre sujet. Nous avons identifié les acteurs de notre système ainsi que les besoins fonctionnels et non fonctionnels. Puis, nous avons détaillé la première étape de la méthodologie que nous avons choisi à savoir l'identification de l'équipe de travail et la réalisation de backlog du produit. Ainsi, nous avons illustré la planification des sprints pour passer à la troisième étape qui est la mise en œuvre des sprints. Ensuite nous avons présenté le choix des technologies et l'architecture logicielle adoptée.

Dans le chapitre suivant, nous entamerons le premier sprint

3

Chapitre 3 : Réalisation du sprint 1

Introduction

Dans le dernier chapitre, nous avons fixé les spécifications du projet et nous avons établi le Product Backlog qui présente les différents user stories. Nous avons aussi planifié le projet en le décomposant en quatre sprints. Nous pouvons maintenant commencer les travaux nécessaires pour produire la première itération. Ce chapitre traitera le premier sprint qui est consacré à la réalisation de la première partie de notre étude qui est l'authentification, gestion des comptes et la gestion des patients

3.1 Spécification fonctionnelle

Le Sprint Backlog présente les différentes fonctionnalités que l'équipe Scrum doit réaliser durant le Sprint. Pour chaque fonctionnalité, nous allons passer par la spécification fonctionnelle, la conception, le développement et les tests. Le tableau 3.1 présente les différentes fonctionnalités qui seront développées durant ce sprint.

User story ID	User story	ID tache	Tache
1.1	En tant qu'utilisateur (Médecin, secrétaire, parent) je souhaite m'authentifier.	1.1.1	Réalisation des diagrammes de cas d'utilisation, et séquences de la fonctionnalité "S'authentifier".
		1.1.2	Développement du cas "S'authentifier" (avec mot de passe crypté)
		1.1.3	Test du cas "S'authentifier"
2.1	En tant que parent je souhaite ajouter les coordonnées d'un enfant.	2.1.1	éalisation des diagrammes de cas d'utilisation, et séquences de la fonctionnalité "Ajouter enfant".
		2.1.2	Développement du cas "Ajouter enfant".
		2.1.3	Test du cas "Ajouter enfant".
2.2	En tant que parent je souhaite modifier les coordonnées d'un enfant.	2.2.1	éalisation des diagrammes de cas d'utilisation, et séquences de la fonctionnalité "Modifier enfant".
		2.2.2	Développement du cas "Modifier enfant".
		2.2.3	Test du cas "Modifier enfant".
2.3	En tant que parent je souhaite supprimer les coordonnées d'un enfant.	2.3.1	éalisation des diagrammes de cas d'utilisation, et séquences de la fonctionnalité "Supprimer enfant".
		2.3.2	Développement du cas "Supprimer enfant".
		2.3.3	Test du cas "Supprimer enfant".
2.4	En tant que parent je souhaite consulter les coordonnées d'un enfant.	2.4.1	éalisation des diagrammes de cas d'utilisation, et séquences de la fonctionnalité "consulter enfant".
		2.4.2	Développement du cas "consulter enfant".
		2.4.3	Test du cas "Consulter enfant".
3.1	En tant qu'utilisateur je souhaite ajouter un compte.	3.1.1	éalisation des diagrammes de cas d'utilisation, et séquences de la fonctionnalité "Ajouter compte".
		3.1.2	Développement du cas "Ajouter compte".
		3.1.3	Test du cas "Ajouter compte".

3.2	En tant que utilisateur je souhaite modifier un compte.	3.2.1	éalisation des diagrammes de cas d'utilisation,et séquences de la fonctionnalité “ Modifier compte”.
		3.2.2	Développement du cas “Modifier compte ”.
		3.2.3	Test du cas ”Modifier compte”.
3.3	En tant que médecin je souhaite supprimer un compte.	3.3.1	éalisation des diagrammes de cas d'utilisation,et séquences de la fonctionnalité “ Supprimer compte”.
		3.3.2	Développement du cas “Supprimer compte ”.
		3.3.3	Test du cas « Supprimer compte”.
3.4	En tant que médecin je souhaite consulter les comptes.	3.4.1	éalisation des diagrammes de cas d'utilisation,et séquences de la fonctionnalité “Consulter compte”.
		3.4.2	Développement du cas “Consulter compte ”.
		3.4.3	Test du cas « Consulter compte”..
3.5	En tant que médecin j'affecte un rôle spécifique pour chaque utilisateur..	3.5.1	Développement du cas “Affecter rôle ”.

TABLE 3.1 :Le backlog Sprint 1

3.1.1 Diagramme de cas d'utilisation du premier sprint

La figure 3.1 illustre le diagramme de cas d'utilisation initiale du premier sprint. Le parent prendra en charge la gestion des patients. Alors que le médecin a pour rôle de gérer les comptes.

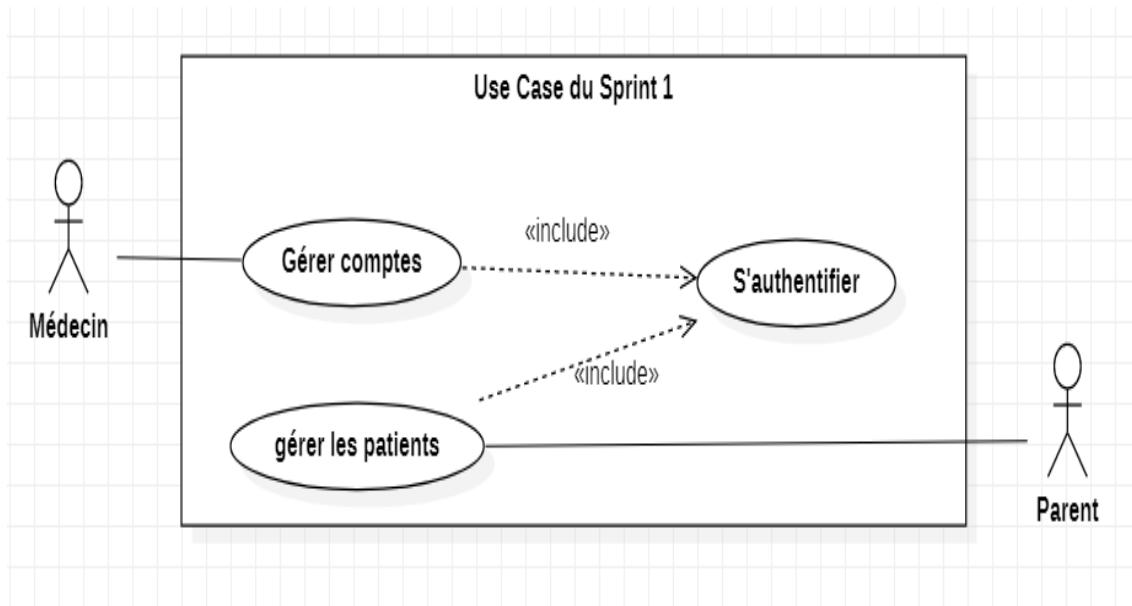


FIGURE 3.1 – Cas d'utilisation Sprint 1

3.1.2 Description des cas d'utilisation

Description du cas d'utilisation « S'authentifier »

Cas d'utilisation	S'authentifier
Acteur	Utilisateur (Médecin, secrétaire, et parent)
Précondition	L'identifiant et le mot de passe saisis doivent être corrects.
Post-condition	Utilisateur authentifié.
Scénario principal	1- L'utilisateur saisit son login et son mot de passe. 2- L'utilisateur clique sur le bouton "S'identifier" 3- Le système affiche l'interface d'accueil propre à l'utilisateur selon son rôle
Scénario alternatif	Un message d'erreur s'affiche pour informer l'utilisateur que l'identifiant ou le mot de passe sont erronés.

TABLE 3.2 : Description textuelle du cas d'utilisation “ S'authentifier ”

Description du cas d'utilisation «Ajouter patient»

Cas d'utilisation	Ajouter un patient
Acteur	Parent
Précondition	Parent authentifié.
Post-condition	Patient ajouté.
Scénario principal	1- Le client clique sur le bouton "Ajouter un enfant". 2- Le système affiche le formulaire d'ajout. 3- Le parent remplit les informations nécessaires. 4- Le parent clique sur le bouton "Enregister". 5- Le système enregistre le nouvel enfant. 6- Le parent reçoit un message de succès.

TABLE 3.3 : Description textuelle du cas d'utilisation “ Ajouter patient”

Description du cas d'utilisation «Supprimer patient»

Cas d'utilisation	Supprimer un patient
Acteur	Parent
Précondition	Parent authentifié.
Post-condition	Patient supprimé.
Scénario principal	1- Le parent choisi l'enfant à supprimer. 2- Le système affiche un message de validation . 3- Le parent confirme son choix .

TABLE 3.4 : Description textuelle du cas d'utilisation “ Supprimer patient”

Raffinement de cas d'utilisation « Gérer patients»

Le parent a pour rôle d'ajouter, supprimer, modifier et consulter la liste de ses enfants. La figure 3.2 illustre le raffinement de cas d'utilisation «Gérer les patients».

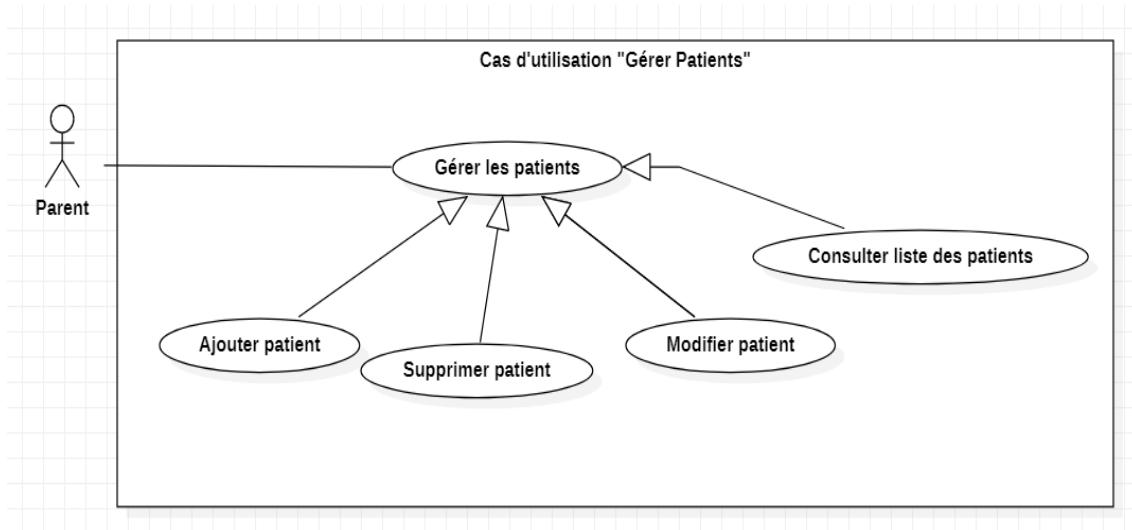


FIGURE 3.2 – Cas d'utilisation Gérer patient ”

Raffinement de cas d'utilisation « Gérer compte»

Le médecin a pour rôle d'ajouter, supprimer, modifier et gérer les rôles. La figure 3.3 illustre le raffinement de cas d'utilisation «Gérer compte»

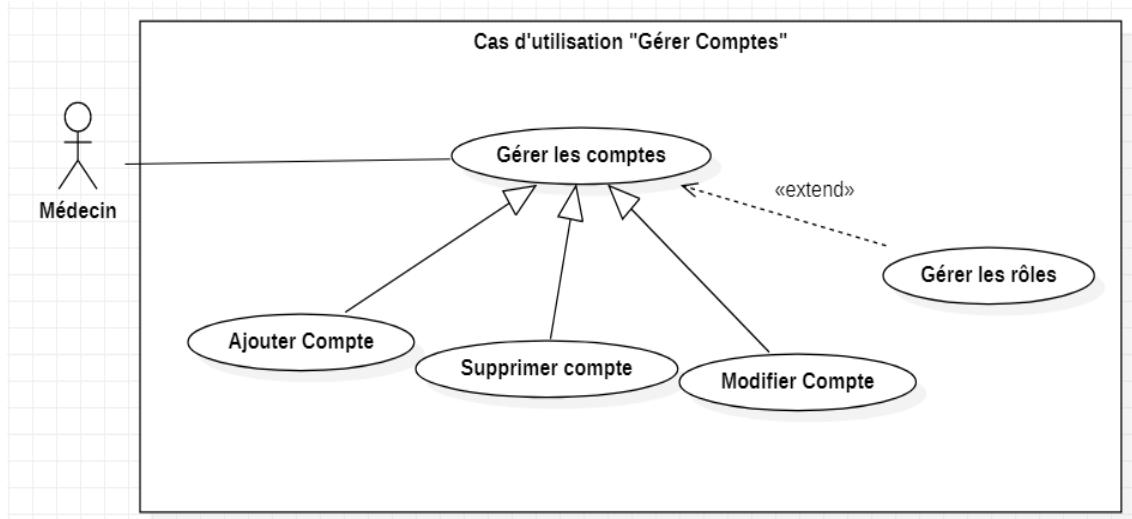


FIGURE 3.3 – Cas d'utilisation Gérer Compte ”

3.2 Conception détaillée

Dans cette partie, nous allons représenter les interactions de ce sprint sous forme de diagrammes de séquence.

3.2.1 Diagramme de séquence objet du cas d'utilisation « S'authentifier »

La figure 3.4 illustre le diagramme de séquence relatif au cas d'utilisation « S'authentifier ».

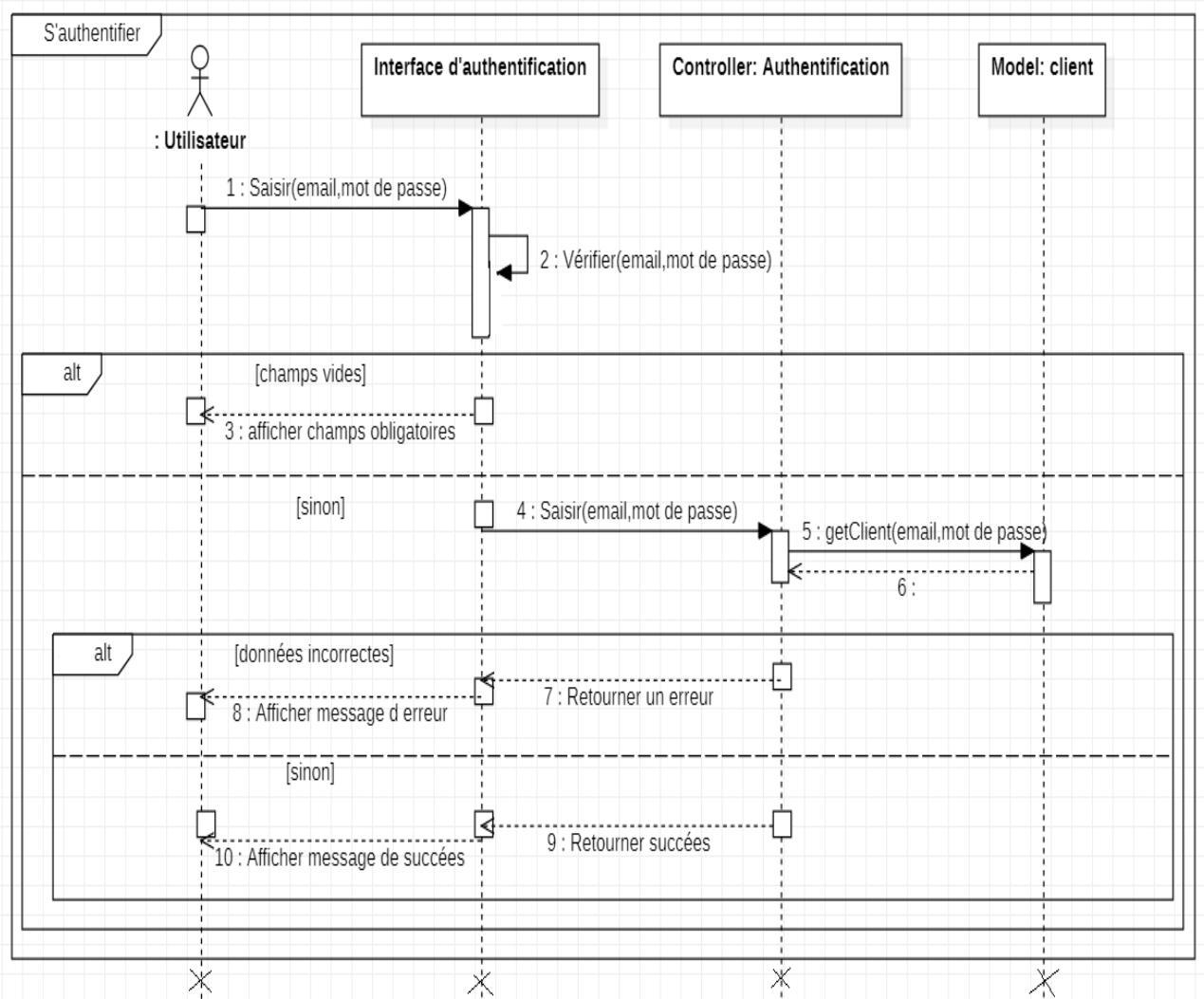


FIGURE 3.4 – Diagramme de séquence objet du cas d'utilisation "S'authentifier"

3.2.2 Diagramme de séquence objet du cas d'utilisation «Ajouter patient»

La figure 3.5 illustre le diagramme de séquence relatif au cas d'utilisation « Ajouter enfant ».

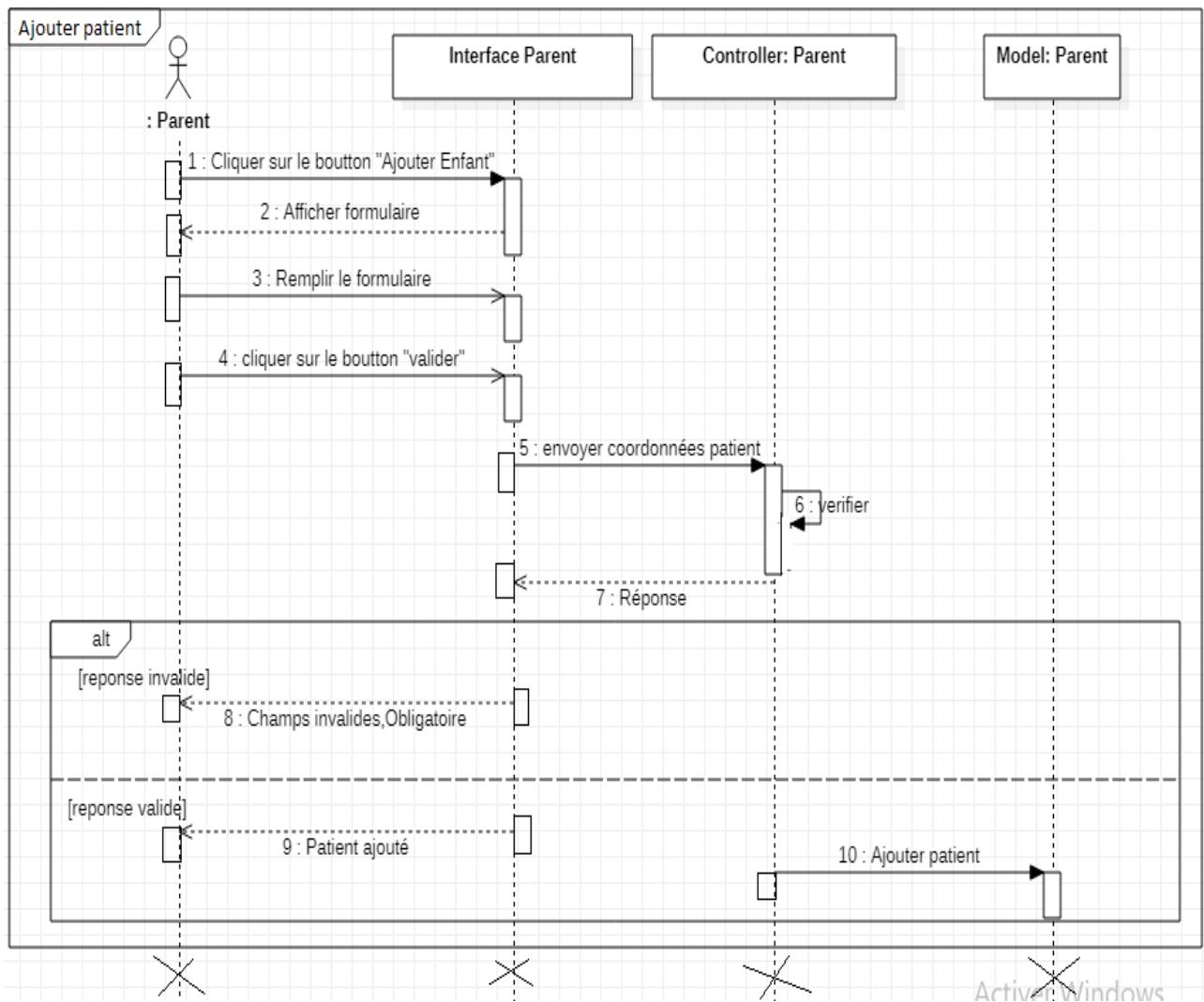


FIGURE 3.5 – : Diagramme de séquence objet du cas d'utilisation "Ajouter enfant"

3.3 Réalisation

Dans cette partie, nous allons exposer quelques scénarios d'exécution à travers des captures d'écran.

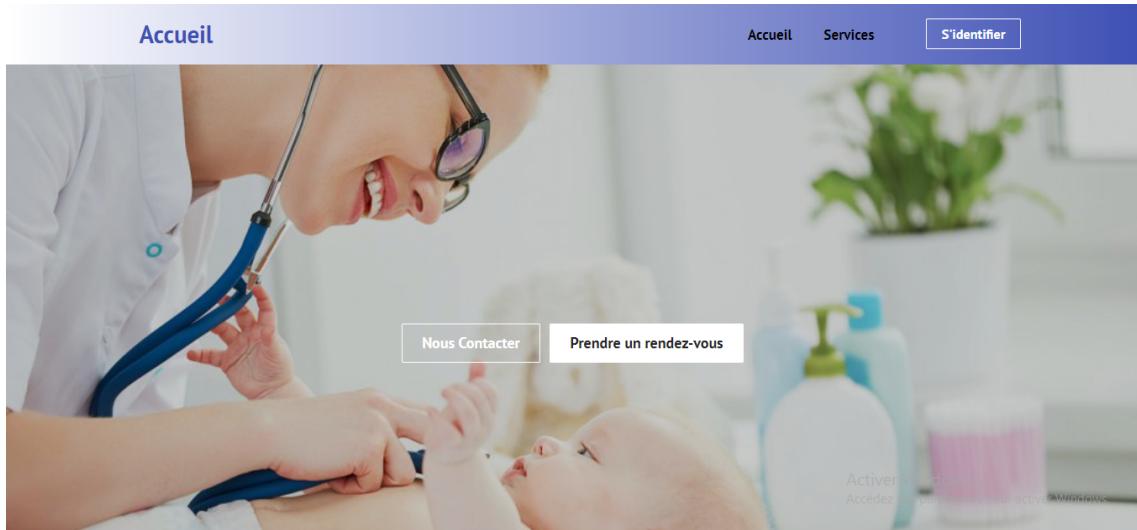
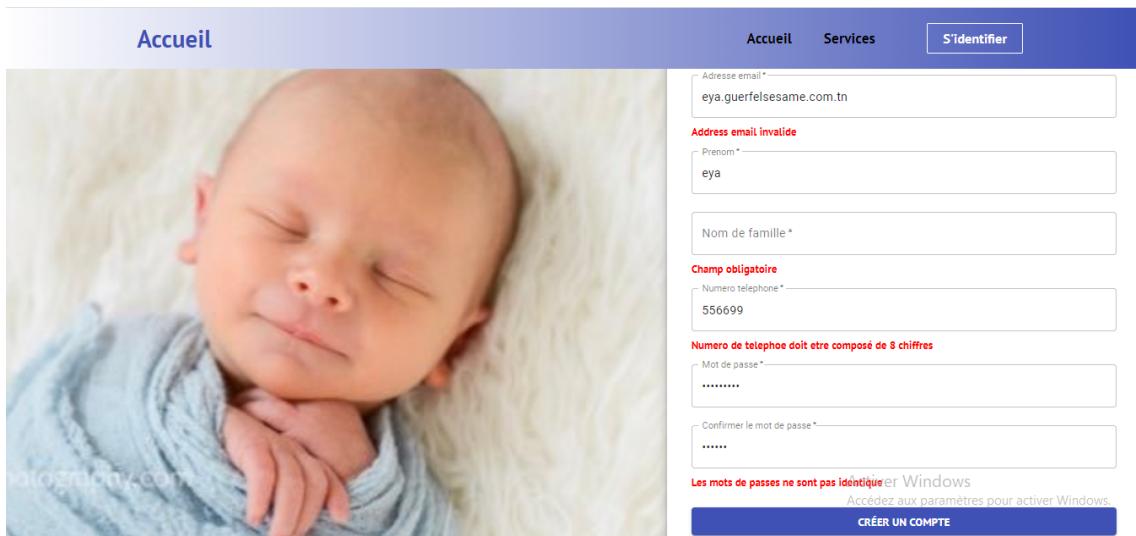


FIGURE 3.6 – Interface d'accueil de l'application

Les figures 3.7 et 3.8 représentent l'interface de création d'un nouveau compte pour les utilisateurs pour pouvoir s'authentifier. Lors de la création d'un nouveau compte, une vérification et validation des saisies de l'utilisateur sera au fur et à mesure des saisies dans les champs de formulaire avec des tests sur les données.

FIGURE 3.7 – Interface de création d'un nouveau compte



Accueil

Services

S'identifier

Adresse email *
eya.guerfelsesame.com.tn

Address email invalide

Prenom *
eya

Nom de famille *

Champ obligatoire

Numero telephone *
556699

Numero de telephone doit etre compose de 8 chiffres

Mot de passe *

Confrimer le mot de passe *

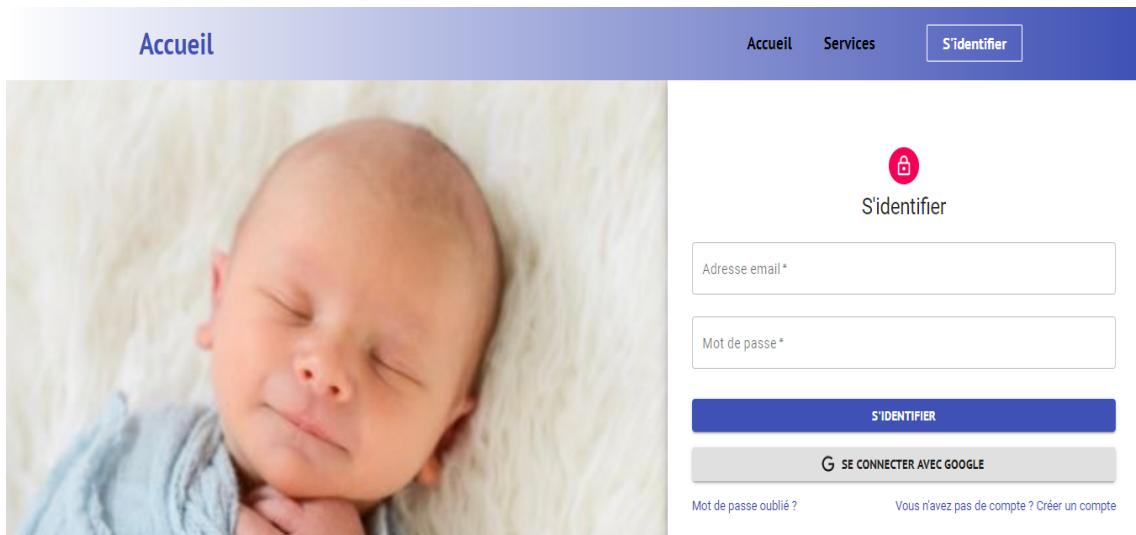
Les mots de passes ne sont pas identiques

Accédez aux paramètres pour activer Windows

CRÉER UN COMPTE

FIGURE 3.8 – Interface de création d'un nouveau compte

Les figures 3.9 et 3.10 représentent l'interface d'authentification pour l'utilisateur afin de manipuler les fonctionnalités de l'application. Nous proposons de contrôler les informations entrées par l'utilisateur lors de son inscription. Un message d'erreur sera affiché si le nom d'utilisateur ou le mot de passe est incorrect.



Accueil

Services

S'identifier

Adresse email *

Mot de passe *

S'IDENTIFIER

SE CONNECTER AVEC GOOGLE

Mot de passe oublié ?

You n'avez pas de compte ? Créez un compte

FIGURE 3.9 – Interface d'authentification

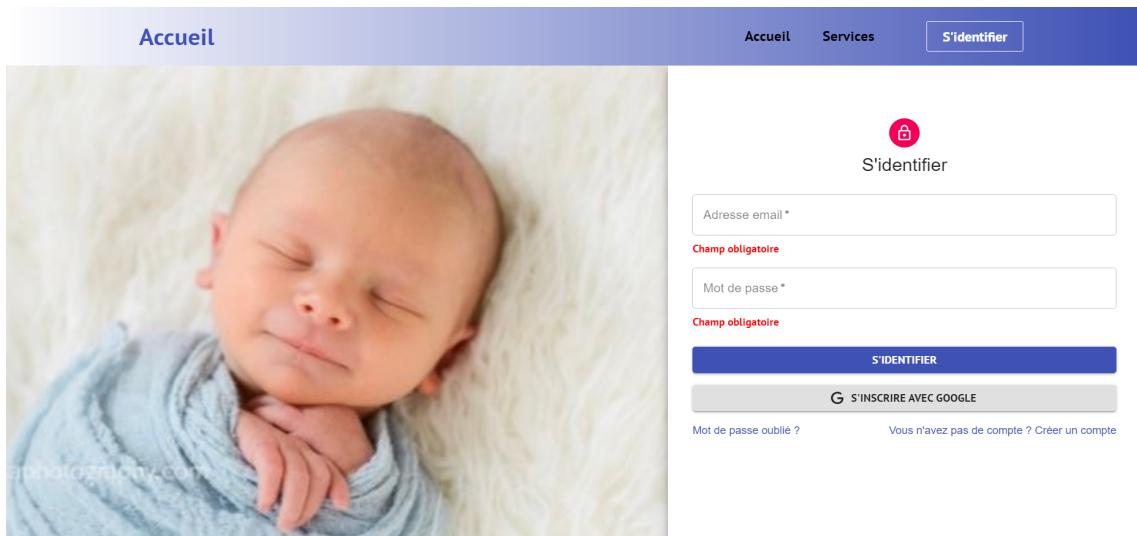


FIGURE 3.10 – Interface d'authentification

Chaque parent inscrit dans l'application est responsable de gérer les patients. Pour ajouter un nouvel enfant, le parent doit remplir le formulaire présenté par la figure 3.11. Les informations doivent être correctes pour qu'elles soient validées.

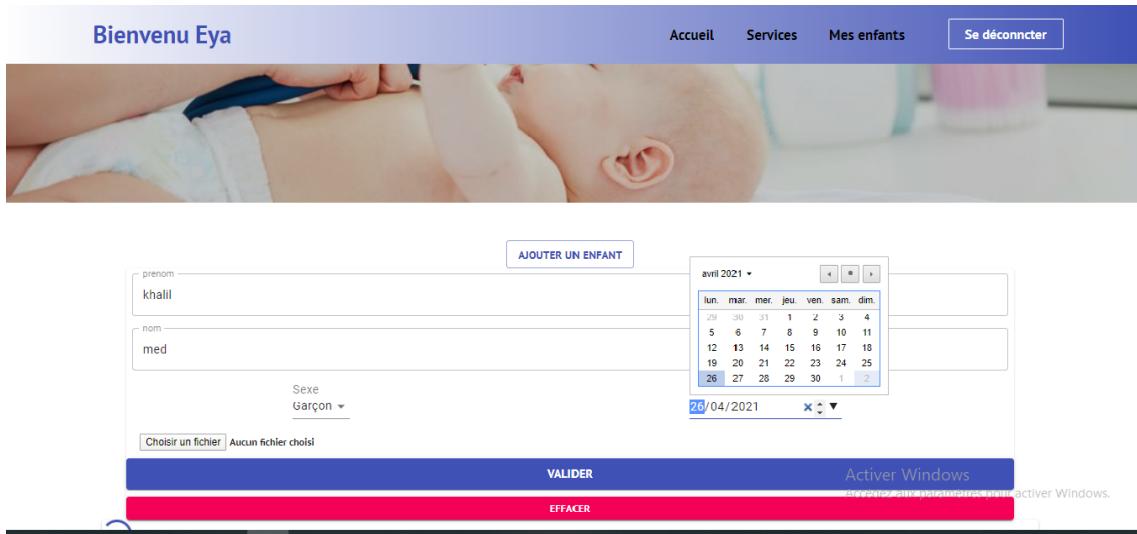


FIGURE 3.11 – Interface d'ajout d'un nouvel enfant

Lorsque le parent veut effectuer des changements pour ses enfants, il clique sur le bouton "modifier", un formulaire rempli avec des données déjà enregistrées, il fait les changements nécessaires et il sauvegarde de nouveau.

La figure 3.12 montre l'interface de modification d'un enfant.



Bienvenu hichem

Accueil Services Mes enfants Se déconnecter

AJOUTER UN ENFANT

Modifier "karim"

nom: karim

nom: med

Sexe: Garçon

Date de naissance: 12/05/2021

Choisir un fichier Aucun fichier choisi

VALIDER EFFACER

FIGURE 3.12 – : Interface de modification d'un enfant

Lorsque le parent veut supprimer un enfant, il clique sur le bouton "Supprimer", un popup sera affiché pour faire la confirmation.

La figure 3.13 l'interface de suppression d'un enfant.

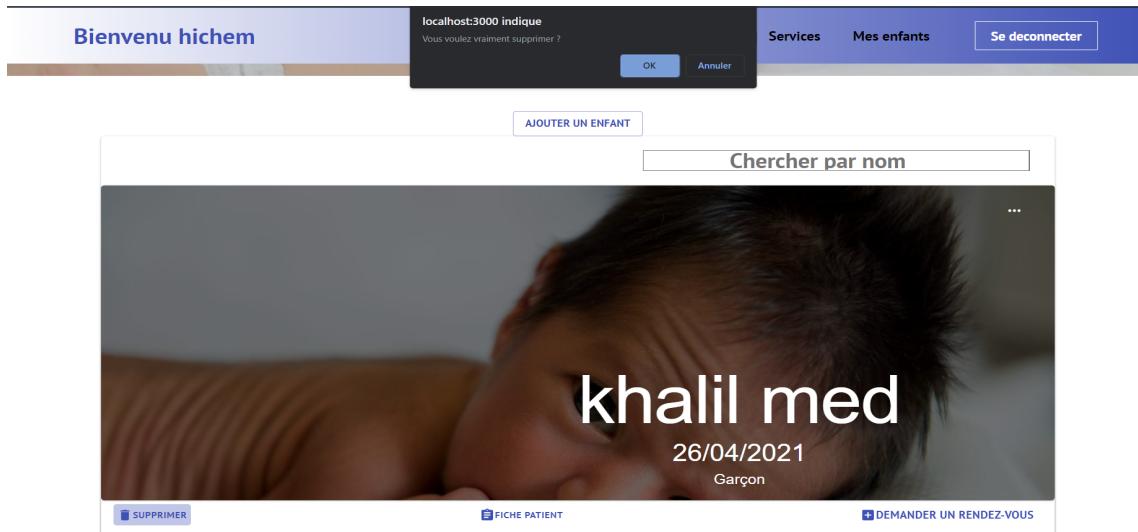


FIGURE 3.13 – : Interface de suppression d'un enfant

La gestion des rendez-vous est une tâche essentielle, celle-ci vérifie la disponibilité de la date demandée et par la suite le parent ajoute un rendez-vous en saisissant les renseignements nécessaires. On a deux modes de recherche sur une date réservée : par

l'heure et la date. Une exception sera déclenchée suite à une opération d'ajout d'un rendez-vous et un message d'erreur sera affiché pour la 'Date et l'heure' qui sont déjà réservées par un parent. La figure 3.14 l'interface de réservation d'un RDV avec vérification du date.

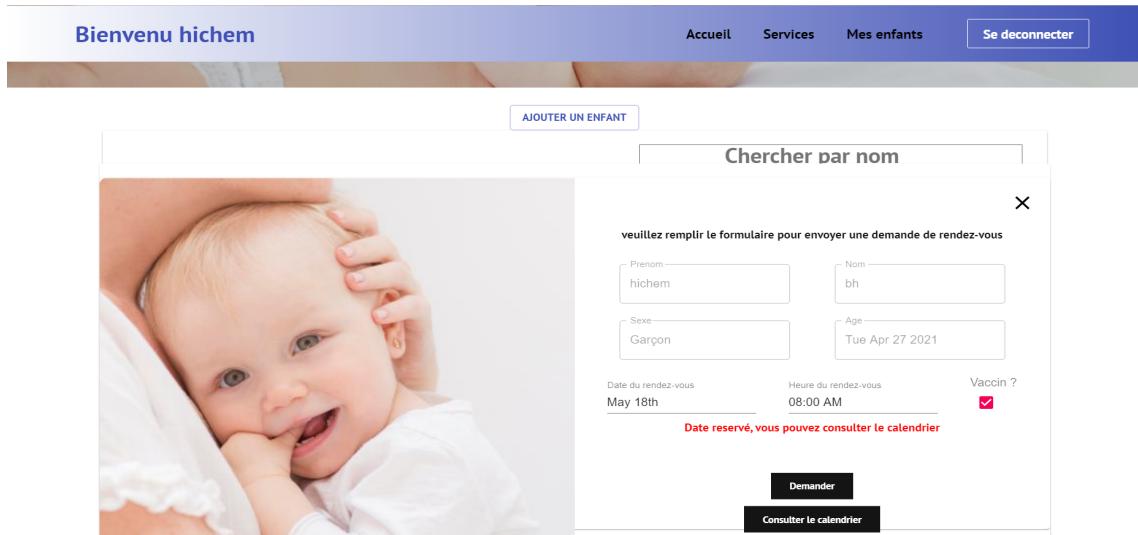


FIGURE 3.14 – : Interface de reservation d'un RDV avec vérification du date

Lorsque le parent veut annuler un RDV, il clique sur le bouton "Supprimer", un popup sera affiché pour faire la confirmation.

La figure 3.15 l'interface d'annulation d'un RDV.

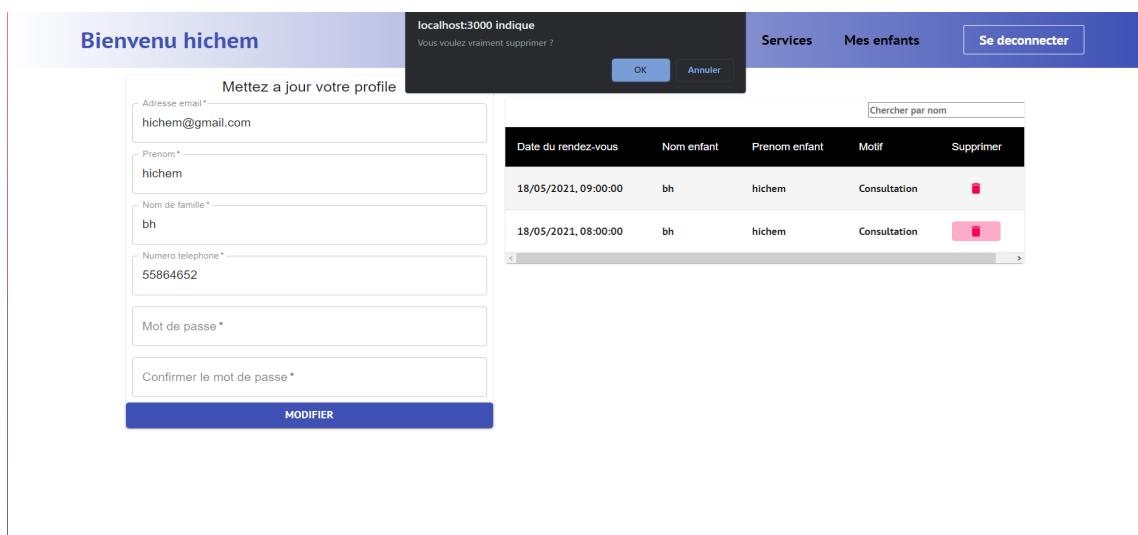


FIGURE 3.15 – : Interface d'annulation d'un RDV.

Le médecin inscrit dans l'application est responsable de gérer les comptes, là où il peut

modifier, supprimer, et rechercher avec nom du patient. il a aussi la possibilité de gérer les rôles.

La figure 3.16 l'interface de gérer un compte.

Nom de l'utilisateur	Prenom de l'utilisateur	Nombre enfant	Role	Supprimer	Modifier
mohamed	med	1	Parent		
sonia	bh	0	Secrétaire		
admin	admin	0	Admin		
hichem	bh	2	Parent		

FIGURE 3.16 – : Interface de gérer compte

Lorsque le médecin veut supprimer un compte, il clique sur le bouton "Supprimer", un popup sera affiché pour faire la confirmation.

La figure 3.17 l'interface de suppression d'un compte.

localhost:3000 indique

Voulez vous vraiment supprimer cet utilisateur ?

Nom de l'utilisateur	Prenom de l'utilisateur	Nombre enfant	Role	Supprimer	Modifier
mohamed	med	1	Parent		
sonia	bh	0	Secrétaire		
admin	admin	0	Admin		
hichem	bh	2	Parent		

FIGURE 3.17 – : Interface de suppression d'un compte

Lorsque l'utilisateur veut effectuer des changements pour son compte, il clique sur le bouton modifier, un formulaire rempli avec des données déjà enregistrées, il fait les chan-

gements nécessaires et il sauvegarde de nouveau.

La figure 3.18 l'interface de modification d'un compte uilisateur.

Mettez à jour votre profile

Adresse email *
hichem@gmail.com

Prenom *
hichem

Nom de famille *
bh

Numero telephone *
55555645

Mot de passe *

Confirmer le mot de passe *

MODIFIER



FIGURE 3.18 – : Interface de modification d'un compte utilisateur

Conclusion

Au cours de ce chapitre, nous avons présenté le premier sprint. Pour ce faire, nous avons passé par la spécification, la conception, le codage et les tests. Dans le chapitre suivant nous entamons le deuxième sprint.

4

Chapitre 4 : Réalisation du sprint 2

Introduction

Dans le chapitre précédent, dédié au premier sprint, nous avons présenté la conception, le développement, la réalisation et le test de notre première partie du projet associé à l'authentification, gérer les patients, et gérer les comptes. Après la réalisation de ce dernier nous avons donc obtenu une version initiale de notre application. Dans ce deuxième incrément, nous allons nous concentrer sur gérer les rendez-vous,gérer fiche patient et consulter le calendrier des rendez-vous

4.1 Spécification fonctionnelle

Le tableau 4.1 représente le Sprint Backlog du deuxième sprint.

User story ID	User story	ID tache	Tache
4.1	En tant que Secretaire ou Medecin je souhaite intégrer FullCalendar.	4.1.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ Intégrer FullCalendar”.
		4.1.2	Développement du cas “ Intégrer FullCalendar”
		4.1.3	Test du cas “Intégrer FullCalenda ”
5.1	En tant que secretaire je souhaite confirmer les rendez-vous.	5.1.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ confirmer les rendez-vous”
		5.1.2	Développement du cas “ confirmer les rendez-vous”.
		5.1.3	Test du cas “confirmer les rendez-vous ”
5.2	En tant que parent je souhaite ajouter un rendez-vous.	5.2.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ ajouter les rendez-vous”
		5.2.2	Développement du cas “ ajouter les rendez-vous”.
		5.2.3	Test du cas “ajouter un rendez-vous ”
5.3	En tant que utilisateur je souhaite consulter un rendez-vous.	5.3.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ supprimer un rendez-vous”
		5.3.2	Développement du cas “ supprimer un rendez-vous”.
		5.3.3	Test du cas “ supprimer un rendez-vous ”
5.4	En tant que secretaire je souhaite supprimer un rendez-vous.	5.4.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ supprimer un rendez-vous”
		5.4.2	Développement du cas “ supprimer un rendez-vous”.
		5.4.3	Test du cas “ supprimer un rendez-vous ”

6.1	En tant que medecin ou secretaire je souhaite ajouter les coordonnees d'un patient.	6.1.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ ajouter les coordonnees d'un patient”
		6.1.2	Développement du cas “ ajouter les coordonnees d'un patient”.
		6.1.3	Test du cas “ajouter les coordonnees d'un patient”
6.2	En tant que medecin ou secretaire je souhaite modifier les coordonnees d'un patient.	6.2.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ modifier les coordonnees d'un patient”
		6.2.2	Développement du cas “ modifier les coordonnees d'un patient”.
		6.2.3	Test du cas “modifier les coordonnees d'un patient”
6.3	En tant que medecin ou secretaire je souhaite supprimer les coordonnees d'un patient.	6.3.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ supprimer les coordonnees d'un patient”
		6.3.2	Développement du cas “ supprimer les coordonnees d'un patient”.
		6.3.3	Test du cas “supprimer les coordonnees d'un patient”
6.4	En tant que medecin ou secretaire je souhaite consulter les coordonnees d'un patient.	6.4.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ consulter les coordonnees d'un patient”
		6.4.2	Développement du cas “ consulter les coordonnees d'un patient”.
		6.4.3	Test du cas “consulter les coordonnees d'un patient”

TABLE 4.1 : Backlog Sprint 2

4.1.1 Diagramme de cas d'utilisation du deuxième sprint

La figure 4.1 illustre le diagramme de cas d'utilisation globale du deuxième sprint. Le médecin prendra en charge la gestion des fiches patients, et des rendez-vous, ainsi il peut consulter le calendrier. La secrétaire de l'application peuvent consulter le calendrier, modifier, supprimer et consulter la liste des rendez-vous et générer fiche patient. Le parent a la possibilité d'ajouter et consulter un rendez-vous.

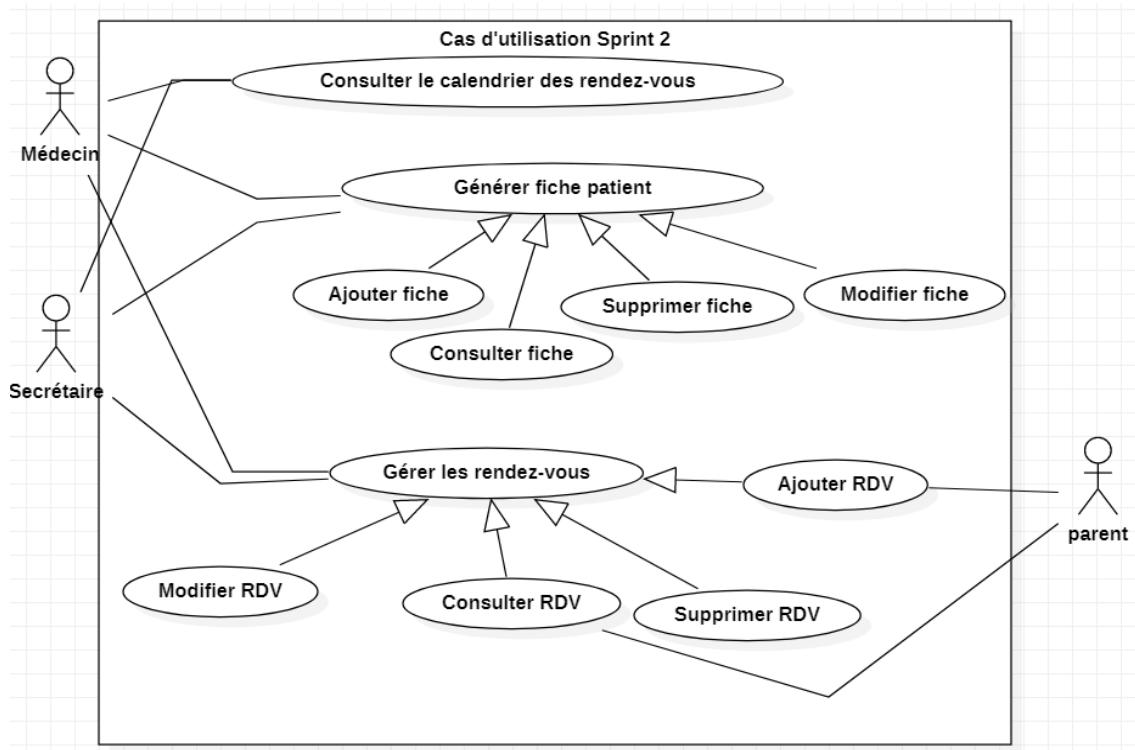


FIGURE 4.1 – Diagramme de cas d'utilisation du deuxième sprint

4.1.2 Description des cas d'utilisation

Au cours de cette partie nous allons expliquer les cas d'utilisation suivantes.

Description du cas d'utilisation « Prendre rendez-vous »

Pour prendre un rendez-vous, le parent sélectionne la date et l'heure souhaité. Après la vérification de la disponibilité, le rendez-vous est enregistré dans la base de données et un message de confirmation s'affiche au parent. Si la date et/ou l'heure sélectionné ne correspondent pas à la disponibilité, le système envoie un message d'erreur au parent pour

lui informer que son rendez-vous n'a pas été enregistré et lui demande de choisir une autre date. Ci-dessous un tableau descriptif de sous cas d'utilisation.

Description du cas d'utilisation «Ajouter RDV»

Le tableau 4.2 représente une description du sous cas d'utilisation « Ajouter rendez-vous »

Cas d'utilisation	Ajouter un RDV
Acteur	Parent
Précondition	Date de rendez-vous disponible et non prise par un autre parent.
Post-condition	Date de rendez-vous fixée, Un nouveau rendez-vous sera ajouté
Scénario principal	1- Le parent clique sur le bouton "demander RDV". 2- une fenêtre de création d'un nouveau rendez-vous s'affiche . 3- Le parent clique sur la date et l'heure désirées . 4- Le parent clique sur le bouton "Demander" .

TABLE 4.2 : Description textuelle du cas d'utilisation “ Ajouter RDV”

Description du cas d'utilisation «Annuler RDV»

Le tableau 4.3 représente une description du sous cas d'utilisation « Annuler rendez-vous »

Cas d'utilisation	Annuler un RDV
Acteur	Parent
Précondition	Rendez-vous déjà existant.
Post-condition	Rendez-vous annulé
Scénario principal	1- Le parent accède au calendrier . 2- Le parent clique sur le rendez-vous déjà réservé. 3- une fenêtre s'affiche pour l'annulation de rendez-vous. 4- le client clique sur “Supprimer rendez-vous” . 5-le rendez-vous est annulé

TABLE 4.3 : Description textuelle du cas d'utilisation “ Annuler RDV”

4.2 Conception détaillée

Dans cette partie, nous allons représenter les interactions de ce sprint sous forme de diagrammes de séquence.

4.2.1 Diagramme de séquence objet du cas d'utilisation «Ajouter un rendez-vous»

La figure 4.2 illustre le diagramme de séquence relatif au cas d'utilisation « Ajouter un rendez-vous».

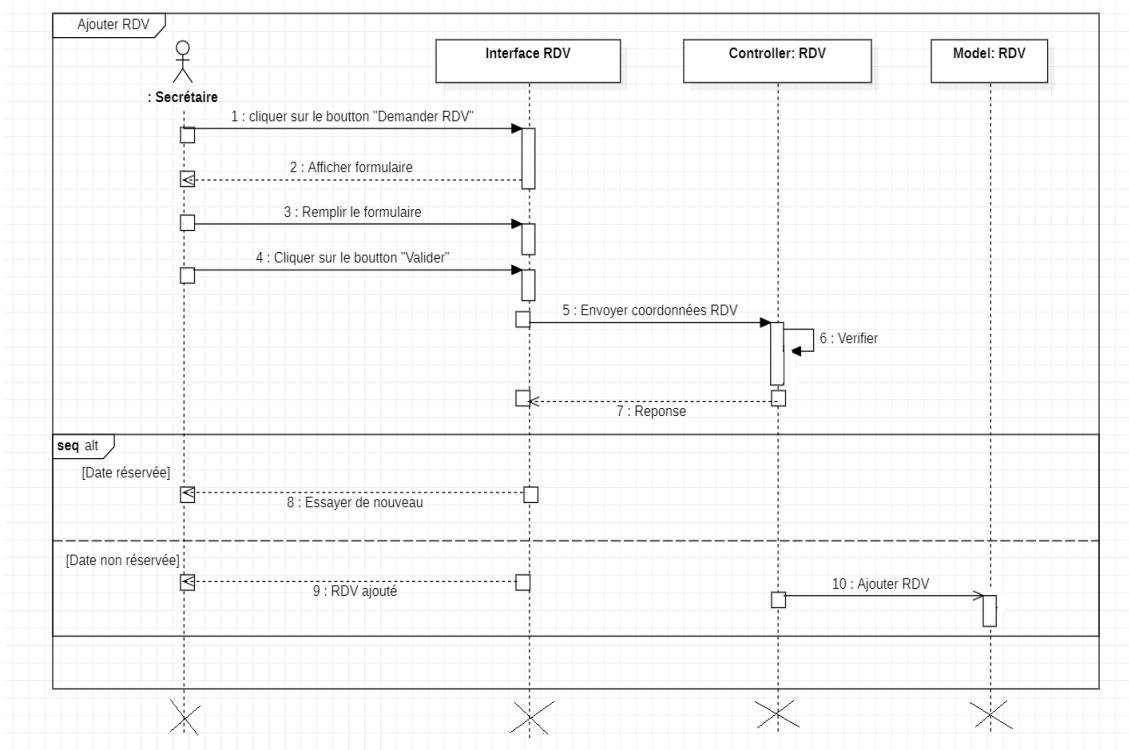


FIGURE 4.2 – Diagramme de cas d'utilisation de gérer RDV

4.2.2 Diagramme de séquence objet du cas d'utilisation «Annuler RDV»

La figure 4.3 illustre le diagramme de séquence relatif au cas d'utilisation «Annuler RDV».

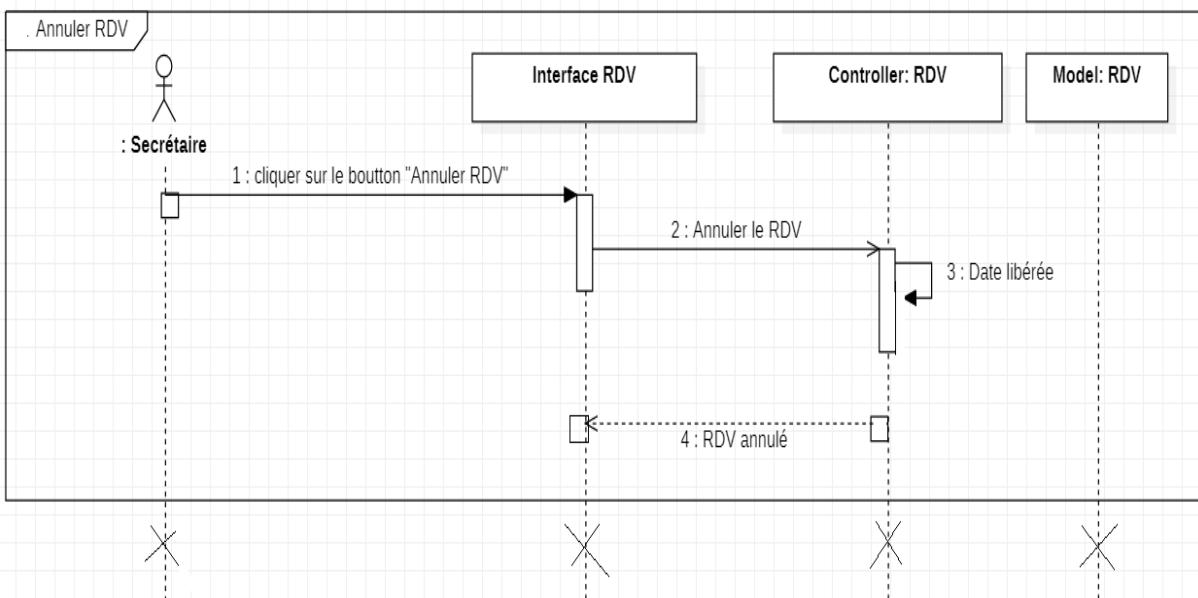


FIGURE 4.3 – : Diagramme de séquence objet du cas d'utilisation ”Annuler RDV”

4.3 Réalisation

Dans cette partie, nous allons exposer quelques scénarios d'exécution à travers des captures d'écran.

Le parent inscrit a le droit de consulter le calendrier des rendez-vous choisi comme il est illustré dans la figures 4.4.

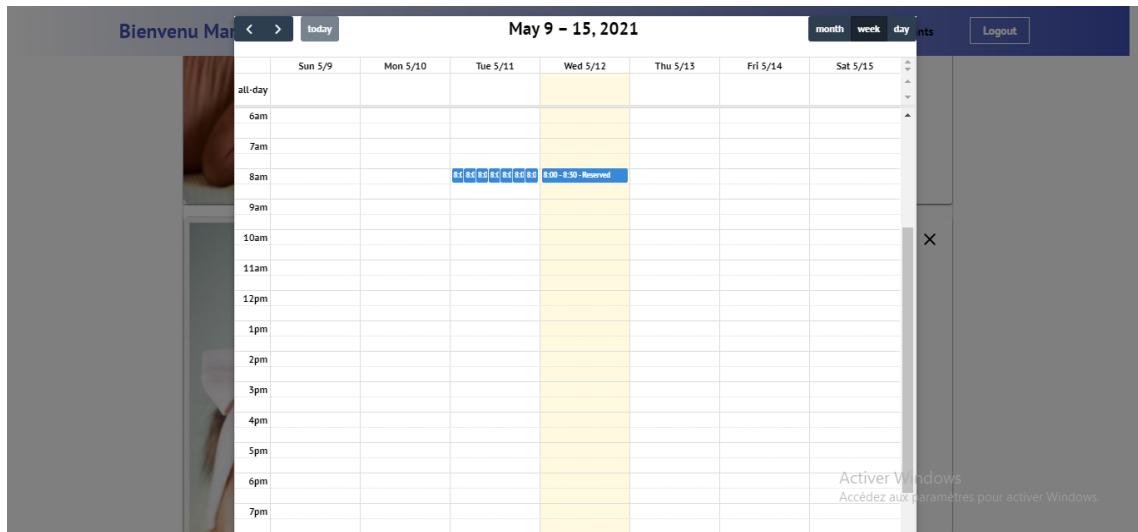


FIGURE 4.4 – : Interface de consulter le calendrier des RDV

A l'arrivée d'un nouveau patient le médecin remplit les informations relatives du pa-

tient sur une fiche. La figure 4.5 l'interface de fiche patient.

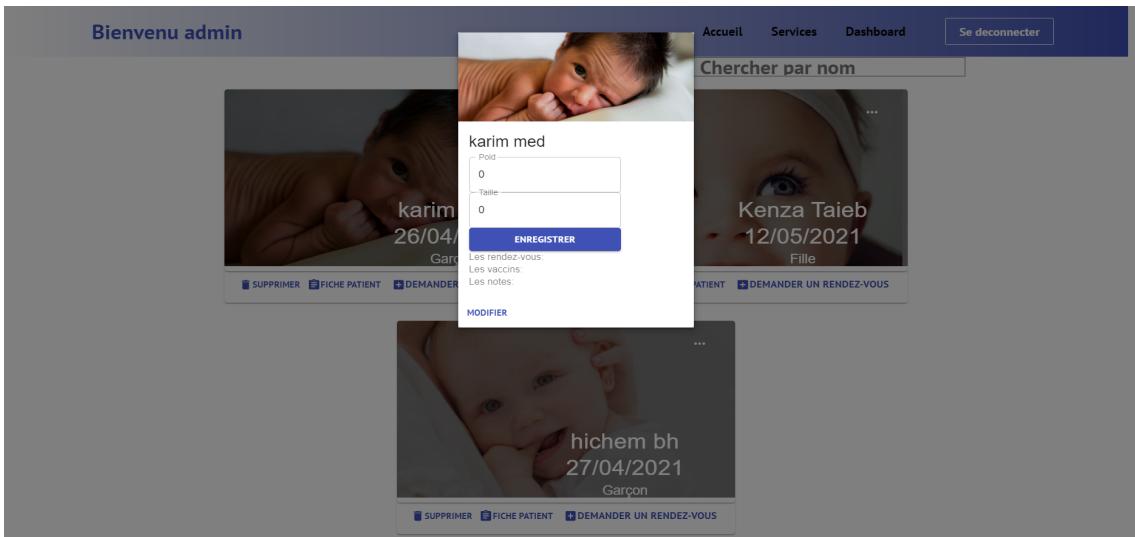


FIGURE 4.5 – : Interface de fiche patient

S'il s'agit d'un ancien patient, le médecin demande son nom et prénom pour chercher sa fiche médicale pour passer à la consultation

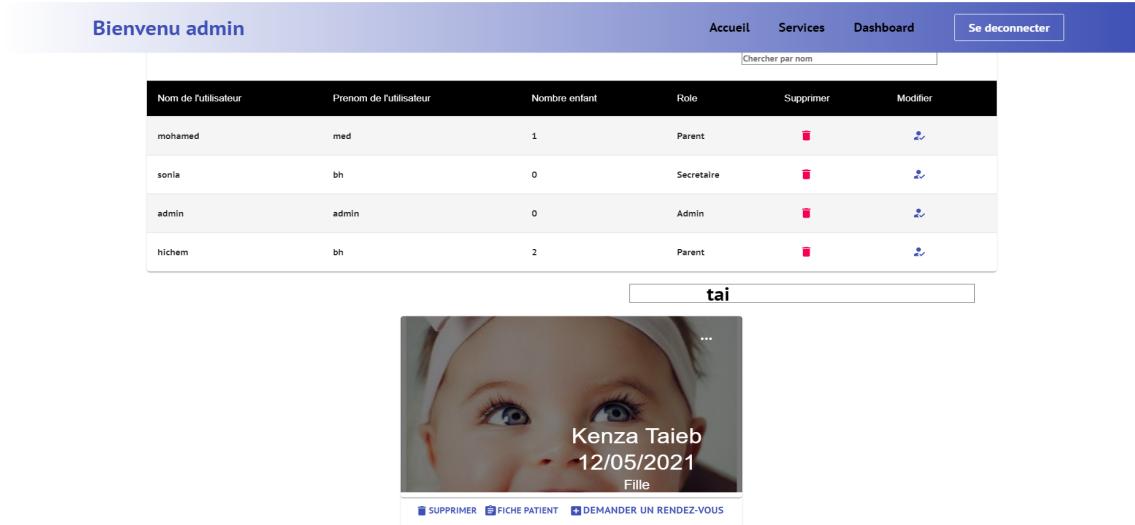


FIGURE 4.6 – : Interface de fiche d'un patient

La secrétaire a le droit de consulter les rendez-vous et de les supprimer comme il est illustré dans la figure 4.7.

The screenshot shows a web application interface for managing patient appointments. At the top, there's a purple header bar with the text "Bienvenu sonia". To the right of the header are links for "Accueil", "Services", "Dashboard", and a "Se deconnecter" button. Below the header is a navigation bar with tabs: "CALENDRIER" (selected), "Table des rendez-vous", and a search bar labeled "Chercher par nom".

The main content area displays a table titled "Table des rendez-vous" showing three scheduled appointments:

Date du rendez-vous	Nom enfant	Prenom enfant	Prenom parent	Motif	Supprimer
14/05/2021, 08:00:00	med	karim	mohamed	Consultation	
18/05/2021, 09:00:00	bh	hichem	hichem	Consultation	
18/05/2021, 08:00:00	bh	hichem	hichem	Consultation	

Below the table, there are two close-up images of babies with their names and birth dates overlaid: "karim med 26/04/2021" and "Kenza Taieb 12/05/2021".

FIGURE 4.7 – : Interface de consultation et suppression des RDV

le medecin et la secrétaire peuvent faire la recherche des patients par leur nom.

The screenshot shows a web application interface for searching patients. At the top, there's a purple header bar with the text "Bienvenu admin". To the right of the header are links for "Accueil", "Services", "Dashboard", and a "Se deconnecter" button. Below the header is a search bar labeled "Chercher par nom".

The main content area displays a table showing users and their children:

Nom de l'utilisateur	Prenom de l'utilisateur	Nombre enfant	Role	Supprimer	Modifier
mohamed	med	1	Parent		
sonia	bh	0	Secrétaire		
admin	admin	0	Admin		
hichem	bh	2	Parent		

Below the table, there is a search bar containing the letters "tai". An image of a baby's face with the name "Kenza Taieb 12/05/2021 Fille" is displayed. At the bottom of the interface are three buttons: "SUPPRIMER", "FICHE PATIENT", and "DEMANDE UN RDV".

FIGURE 4.8 – : Interface de rechercher les patients par leur noms

Conclusion

Au cours de ce chapitre, nous avons présenté le deuxième sprint. Pour ce faire, nous avons passé par la spécification, la conception, le codage et les tests. Dans le chapitre suivant nous entamons le troisième sprint.

5

Chapitre 5 : Réalisation du sprint 3

Introduction

Dans le chapitre précédent, dédié au deuxième sprint, nous avons présenté la conception, le développement, la réalisation et le test de notre deuxième partie de projet associé à la gestion des rendez-vous, générer fiche patient et consulter le calendrier des rendez-vous. Dans ce troisième incrément, nous allons nous concentrer sur la partie de générer les vaccins, et gérer planning vaccinations et les ordonances.

5.1 Spécification fonctionnelle

Le tableau 5.1 représente le Sprint Backlog du troisième sprint.

User story ID	User story	ID tache	Tache
5.1	En tant que medecin je souhaite modifier un vaccin.	5.2.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “modifier un vaccin”
		5.1.1	Développement du cas “modifier un vaccin”.
		5.1.2	Test du cas “modifier un vaccin”
5.2	En tant que medecin je souhaite consulter les vaccins pour chaque patient.	5.2.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “consulter les vaccins pour chaque patient”
		5.2.2	Développement du cas “consulter les vaccins pour chaque patient”.
		5.2.3	Test du cas “consulter les vaccins pour chaque patient”
6.1	En tant que parent je souhaite recevoir une notification par email.	6.1.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “recevoir une notification par email”.
		6.1.2	Developpement du cas “recevoir une notification par email”
		6.1.3	Test du cas “recevoir une notification par email”
6.2	En tant que parent je souhaite recevoir une notification à travers un message telephonique.	6.2.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “ recevoir une notification à travers un message telephonique”.
		6.2.2	Developpement du cas “ recevoir une notification à travers un message telephonique”
		6.2.3	Test du cas “ recevoir une notification à travers un message telephonique”

7.1	En tant que medecin je souhaite ajouter une ordonnance.	7.1.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité "ajouter une ordonnance".
		7.1.2	Developpement du cas "ajouter une ordonnance"
		7.1.3	Test du cas "ajouter une ordonance"
7.2	En tant que medecin je souhaite supprimer une ordonnance.	7.2.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité "supprimer une ordonnance".
		7.2.2	Developpement du cas "supprimer une ordonnance"
		7.2.3	Test du cas "supprimer une ordonnance"
7.3	En tant que medecin je souhaite imprimer une ordonnance sous la forme PDF.	7.3.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité "imprimer une ordonnance sous la forme PDF".
		7.3.2	Developpement du cas "imprimer une ordonnance sous la forme PDF"
		7.3.3	Test du cas "imprimer une ordonnance sous la forme PDF"
7.4	En tant que parent je souhaite consulter les ordonnances de mes enfants.	7.4.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité "consulter les ordonnances de mes enfants".
		7.4.2	Developpement du cas "consulter les ordonnances de mes enfants"
		7.4.3	Test du cas "consulter les ordonnances de mes enfants"

TABLE 5.1 : Backlog Sprint 3

5.1.1 Diagramme de cas d'utilisation du troisième sprint

La figure 5.1 illustre le diagramme de cas d'utilisation globale du troisième sprint. Le médecin a pour rôle de générer les vaccins, et gérer les ordonnances. Le parent a la

possibilité de recevoir une notification par email et par un message téléphonique.

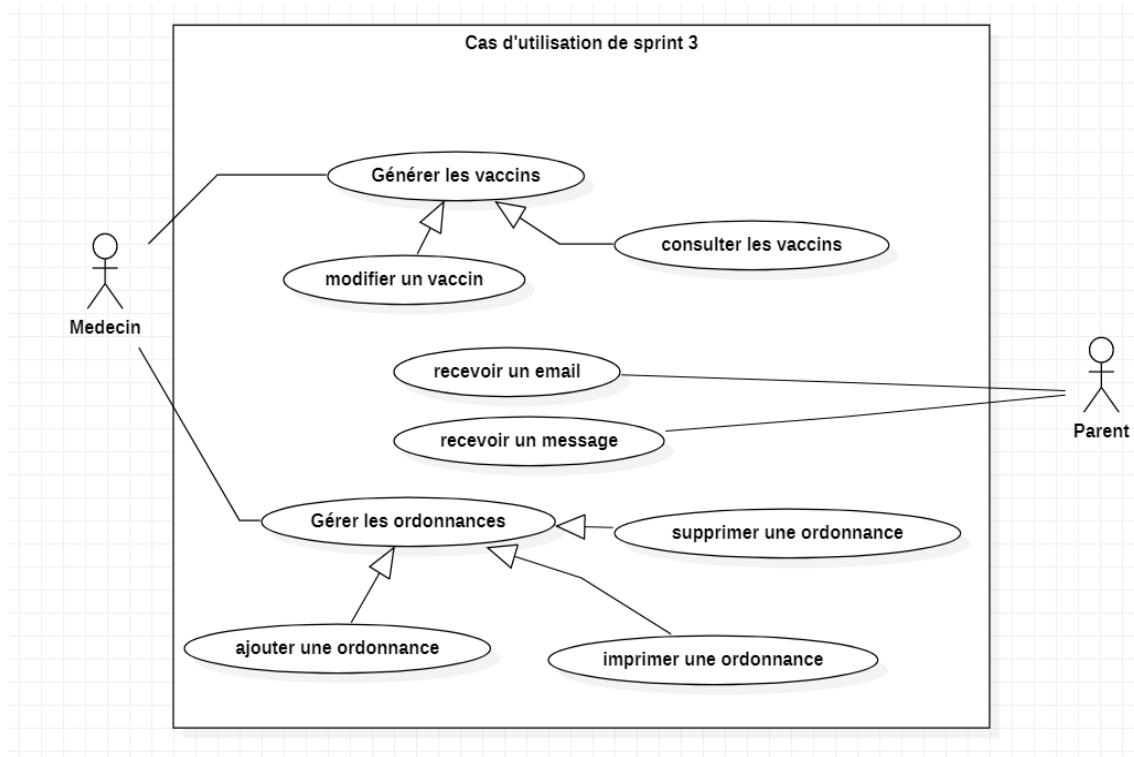


FIGURE 5.1 – Diagramme de cas d'utilisation du troisième sprint

5.1.2 Description des cas d'utilisation

Au cours de cette partie nous allons expliquer les cas d'utilisation suivantes.

Description du cas d'utilisation « Imprimer une ordonnance en forma PDF »

Après la consultation le médecin a le droit de remplir une ordonnance et de l'imprimer sous forme PDF. Le tableau 5.2 représente une description du sous cas d'utilisation « Imprimer une ordonnance »

Cas d'utilisation	Imprimer une ordonnance
Acteur	Medecin
Post-condition	Une nouvelle ordonnance sera ajoutée
Scénario principal	1- Le médecin clique sur le bouton "Imprimer". 2- L'ordonnance sera téléchargée en forme PDF.

TABLE 5.2 : Description textuelle du cas d'utilisation “ Imprimer une ordonnance”

Description du cas d'utilisation « Notification par email »

Lorsque le parent prend un rendez-vous, il reçoit une notification par email et par un message téléphonique pour faire un rappel concernant les vaccins de ses enfants. Le tableau 5.3 représente une description du sous cas d'utilisation « notification par email »

Cas d'utilisation	Notification par email
Acteur	Medecin
Précondition	Le parent prend un rendez-vous.
Post-condition	Le rendez-vous sera ajouté

TABLE 5.3 : Description textuelle du cas d'utilisation “ Notification par email”

Raffinement de cas d'utilisation « Gérer les vaccins»

Le medecin a pour rôle de modifier, et consulter la liste des vaccins. La figure 5.2 illustre le raffinement de cas d'utilisation «Gérer les vaccins».

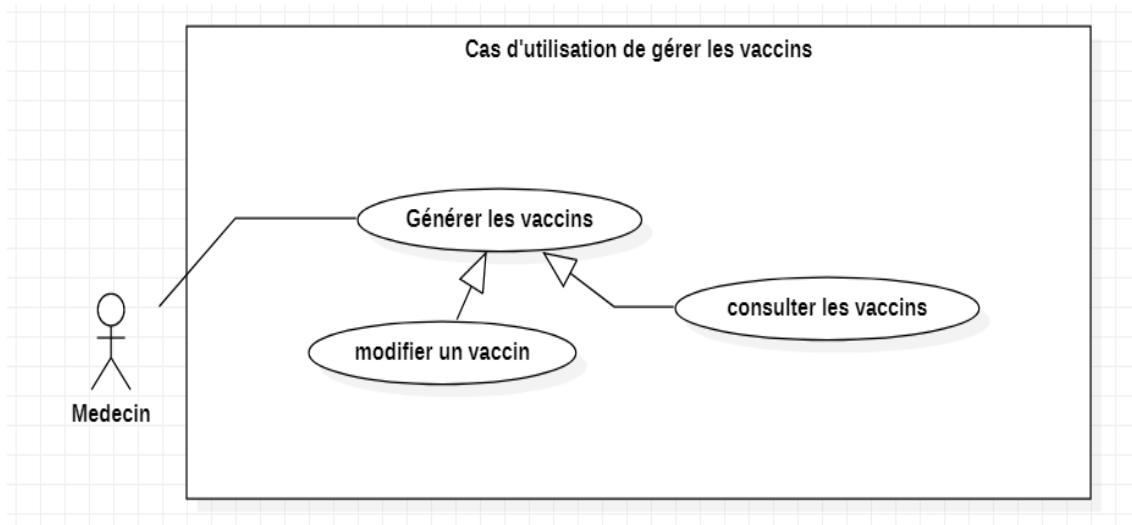


FIGURE 5.2 – Cas d'utilisation ” Gérer les vaccins ”

Raffinement de cas d'utilisation « Gérer les ordonnances»

Le medecin a pour rôle d'ajouter, supprimer, imprimer, et consulter la liste des ordonnances. Ainsi que le parent peut consulter la liste des ordonnances. La figure 5.3 illustre le raffinement de cas d'utilisation «Gérer les ordonnances».

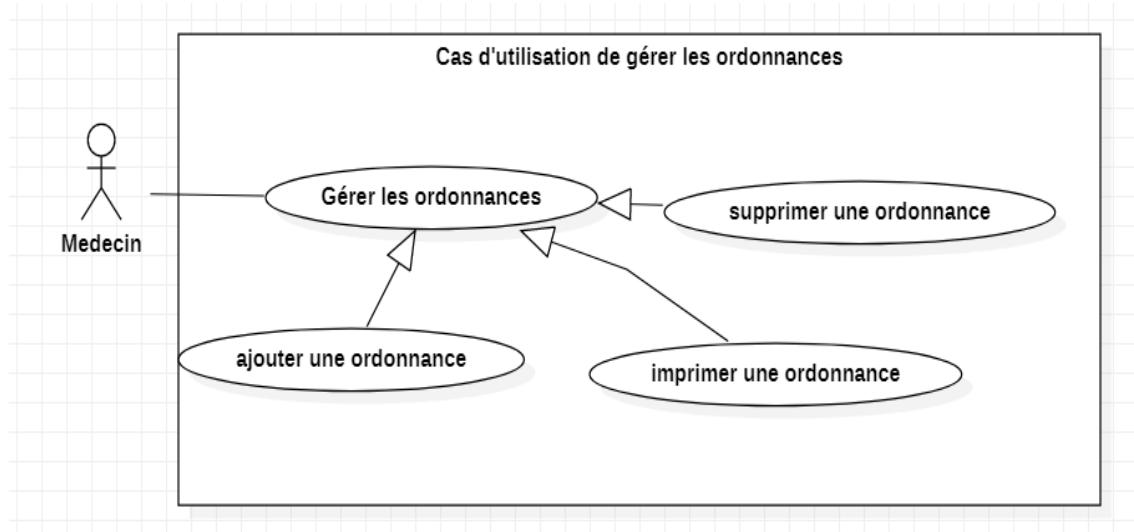


FIGURE 5.3 – Cas d'utilisation "Gérer les ordonnances "

5.2 Réalisation

Dans cette partie, nous allons exposer quelques scénarios d'exécution à travers des captures d'écran.

Le medecin a le droit de modifier et consulter la liste des vaccins choisi comme il est illustré dans la figures 5.4.

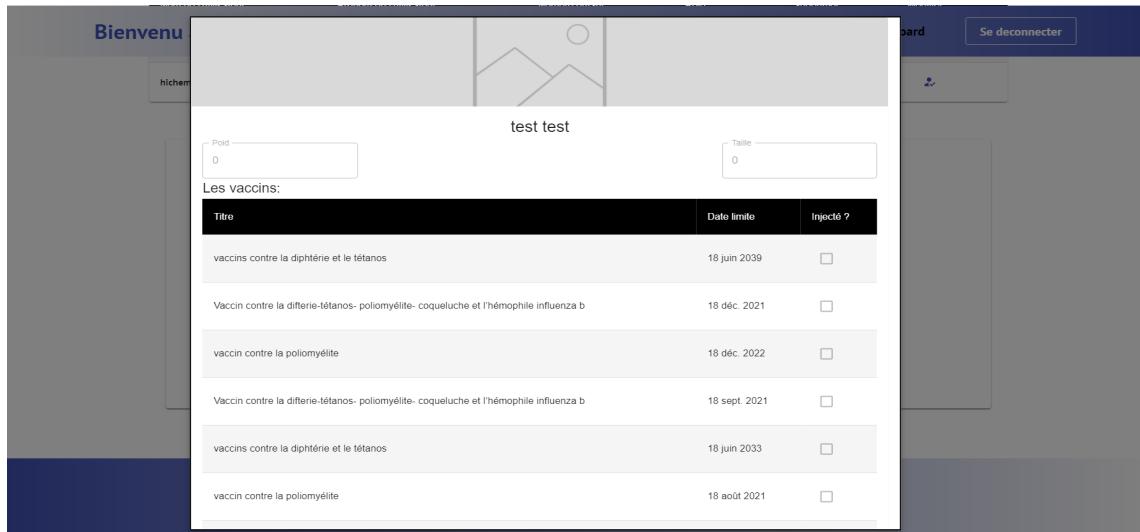


FIGURE 5.4 – : Interface de consulter la liste des vaccins

Le medecin a le droit d'ajouter, et supprimer une ordonnance choisi comme il est illustré dans la figures 5.6 et 5.7.



FIGURE 5.5 – : Interface d'ajouter une ordonnance

Les ordonnances:

Ordonnance créée le 20/06/2021			IMPRIMER	SUPPRIMER
Date du traitement	Medicament	Dosage		
20/06/2021, 18:01:38	doliprane	3 fois par jour		
20/06/2021, 18:01:38	test	12		
Ordonnance créée le 20/06/2021				
Date du traitement	Medicament	Dosage		
20/06/2021, 18:10:32	augmentin	2 fois par jour		
20/06/2021, 18:10:32	ferex	3 fois par jour		
Ordonnance créée le 20/06/2021				
Date du traitement	Medicament	Dosage		
20/06/2021, 18:10:50	clamoxyl 500	une fois par jour		

MODIFIER **AJOUTER ORDONNANCE**

FIGURE 5.6 – : Interface de manipulation d'une ordonnance

Le medecin peut imprimer une ordonnance sous forma PDF. comme il est illustré dans la figure 5.7.

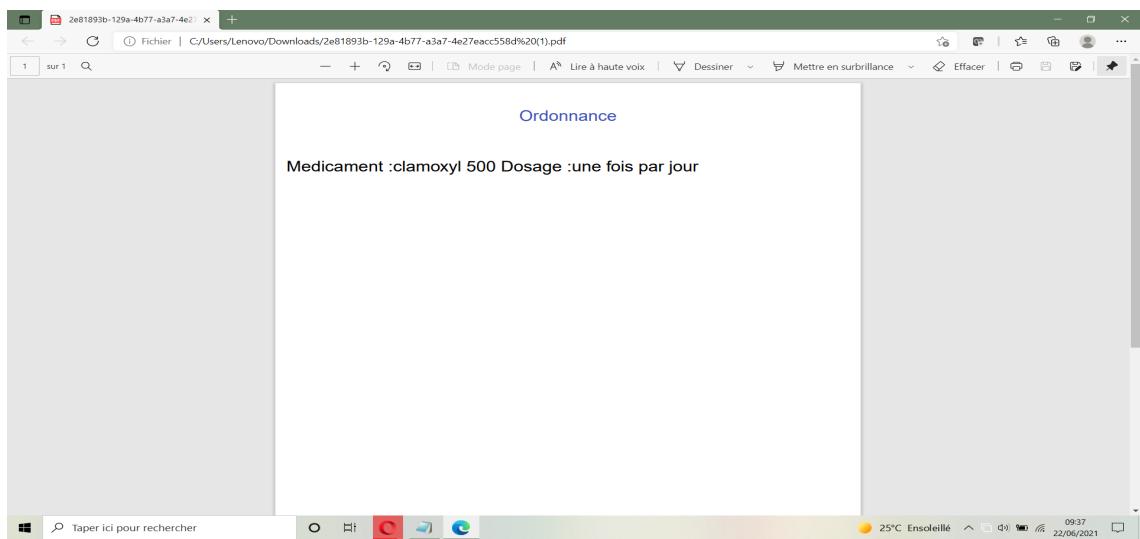


FIGURE 5.7 – : Interface d'imprimer une ordonnance

Lorsque le parent prend un rendez-vous, il reçoit un message sur son téléphone et un email, comme il est illustré dans la figure 5.8 et 5.9.



FIGURE 5.8 – : Interface de notification avec message

Pour la notification par message nous allons utiliser Vonage. Il existe un certain nombre de concepts partagés entre les différentes API Vonage :

- Applications : Informations de sécurité et de configuration dont vous avez besoin pour

vous connecter aux points de terminaison Vonage.

- Authentification :Clés API et Web JSON Token (JWT)
- Signer des messages : Comment signer cryptographiquement des messages et vérifier les signatures - Webhooks : L'API de Vonage peut renvoyer des données à votre serveur Web via un webhook (est utilisés pour les notifications en temps réel, de sorte que votre système peut être mis à jour dès que l'événement se produit.)
- OpenAPI :Toutes les API Vonage ont des descriptions OpenAPI dont vous pouvez tirer parti dans votre propre processus de développement

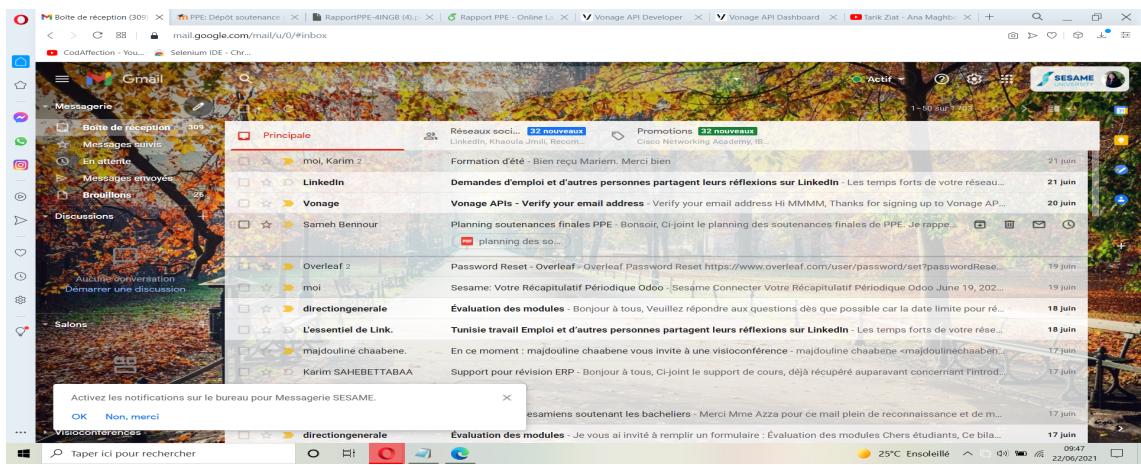


FIGURE 5.9 – : Interface de notification à travers un email

Conclusion

Au cours de ce chapitre, nous avons présenté le troisième sprint. Pour ce faire, nous avons passé par la spécification, la conception, le codage et les tests. Dans le chapitre suivant nous entamons le dernier sprint.

6

Chapitre 6 : Réalisation du sprint 4

Introduction :

Dans le chapitre précédent, dédié au troisième sprint, nous avons présenté la conception, le développement, la réalisation et le test de notre troisième partie de projet associé à la génération des vaccins, gérer planning vaccinations et gérer les ordonnances. Dans ce troisième incrément, nous allons nous concentrer sur la partie de l'analyse des données et la gestion et suivi du dossier médical.

6.1 Spécification fonctionnelle

Le tableau 6.1 représente le Sprint Backlog du quatrième sprint.

User story ID	User story	ID tache	Tache
6.1	En tant que médecin je souhaite analyser des données	6.1.1	En tant que médecin je souhaite consulter les statistiques des maladies reconnues avec google analytics. .
6.2	En tant que medecin je souhaite modifier un dossier médical.	6.2.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “modifier un dossier médical”.
		6.2.2	Developpement du cas “modifier un dossier médical”
		6.2.3	Test du cas “modifier un dossier médical”
6.3	En tant que medecin je souhaite suivre un dossier médical.	6.3.1	Réalisation des diagrammes de cas d'utilisation et séquences de la fonctionnalité “suivre un dossier médical”.
		6.3.2	Developpement du cas “suivre un dossier médical”
		6.3.3	Test du cas “suivre un dossier médical”

TABLE 6.1 : Backlog Sprint 4

6.2 Diagramme de cas d'utilisation du quatrième sprint

La figure 6.1 illustre le diagramme de cas d'utilisation globale du quatrième sprint. Le médecin prendra en charge l'analyse des données et la gestion et suivi du dossier médical.

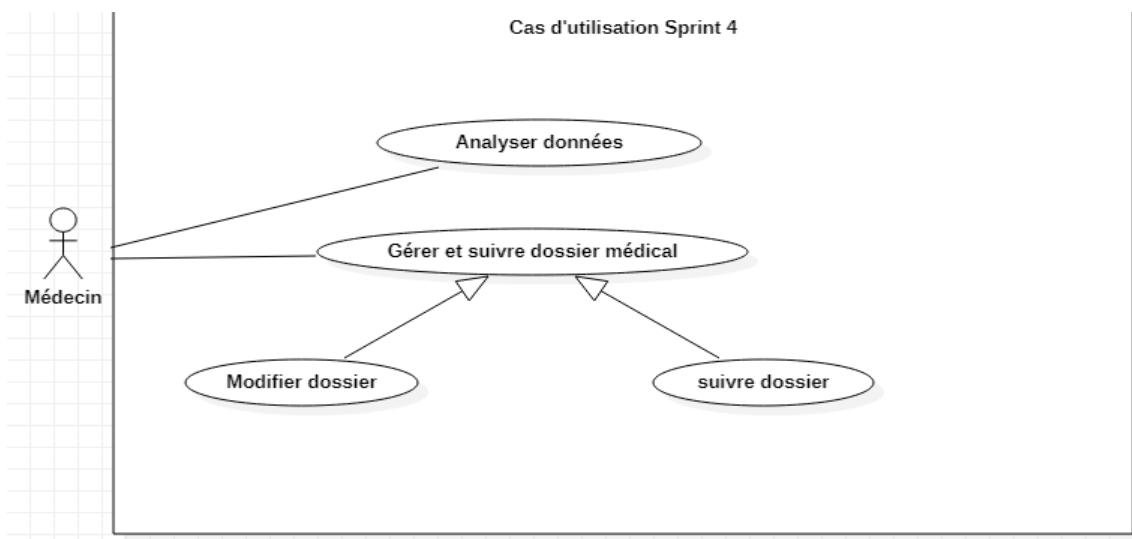


FIGURE 6.1 – Cas d'utilisation Sprint 4

6.3 Réalisation :

Dans cette partie, nous allons exposer quelques scénarios d'execution à travers des captures d'écran.

6.3.1 Analyser les données :

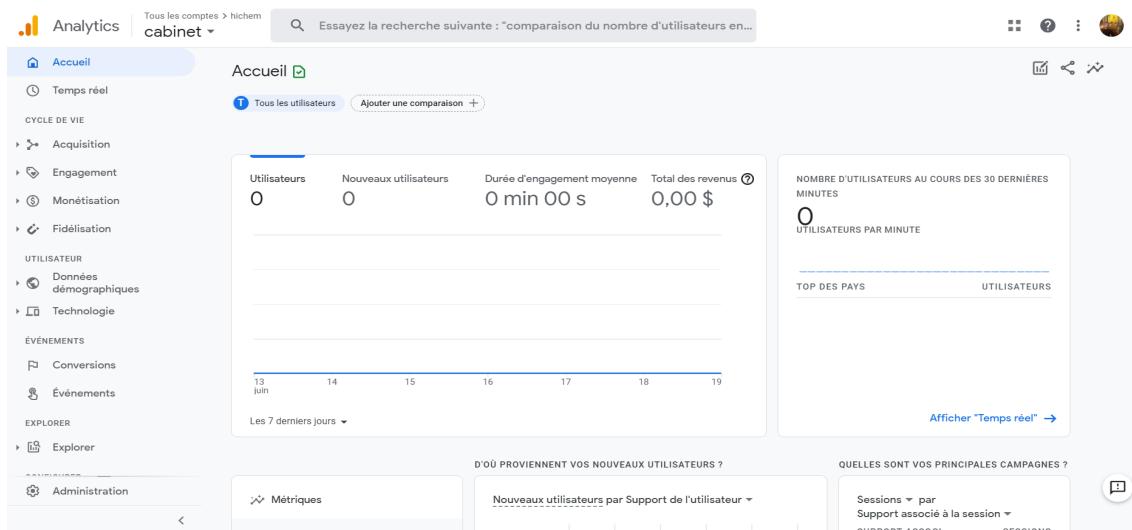


FIGURE 6.2 – Interface Google Analytics

Trois API Analytics sont mises à la disposition de tous les utilisateurs :

- L'API de collecte : personnaliser le code de suivi pour aller au-delà du suivi normal. Collecter les données sur les achats pour notre site, générer des rapports sur deux sous-domaines pour une même vue, créer nos propres définitions du type de visiteurs, etc.
- L'API de gestion : accéder de façon efficace aux données du compte et de la vue Analytics sous forme de flux d'API de données Google. Grâce à l'API de gestion, nous pouvons récupérer un ensemble spécifique de vues pour un utilisateur ou récupérer les données de définition des objectifs associées à une vue spécifique. Enfin, combiner l'API de gestion avec l'API d'exportation pour extraire les données de rapport dont nous avons besoin.
- L'API d'exportation des données : créer des applications à l'aide des données Analytics déjà disponibles dans un compte Analytics. Les applications peuvent exploiter l'API pour demander des données au niveau des rapports à partir d'une vue Analytics existante (si elles sont autorisées à accéder aux données de la vue) et récupérer un rapport de données personnalisé pour une vue donnée.

Les événements

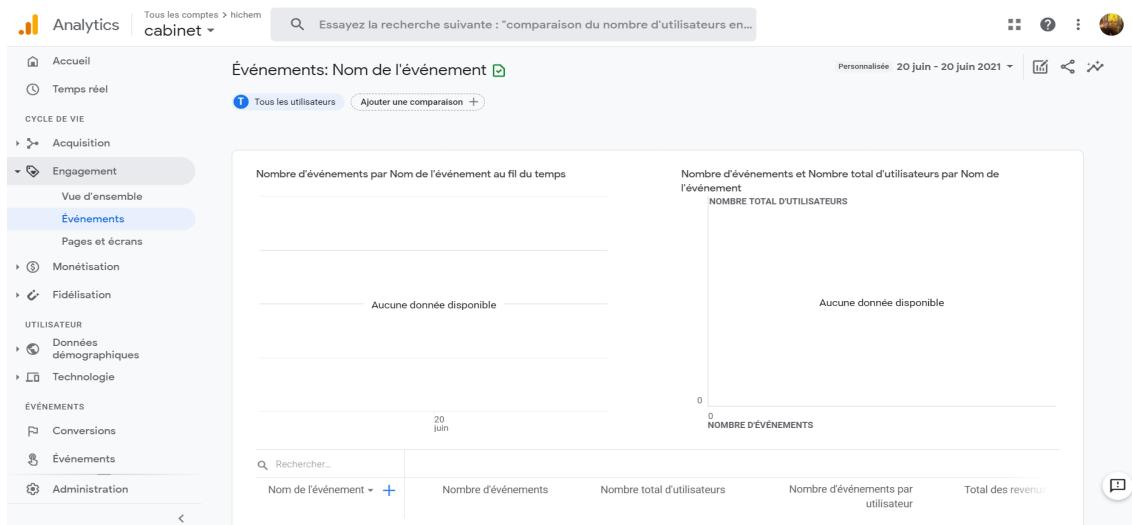


FIGURE 6.3 – Interface Google Analytics event

On entend par "événements" les interactions des utilisateurs avec notre contenu. Ils sont mesurables indépendamment du chargement d'une page Web ou d'un écran. Les

téléchargements, clics sur des liens, envois de formulaires sont autant d'exemples des actions que vous pouvez analyser en tant qu'événements.

6.3.2 Gestion et suivi du dossier médical

Au cours de cette partie, le médecin souhaite faire le suivi et la gestion du dossier médical.

Le médecin peut consulter la liste des vaccins qui peut l'injecter à chaque patient avec la précision de la date limite.

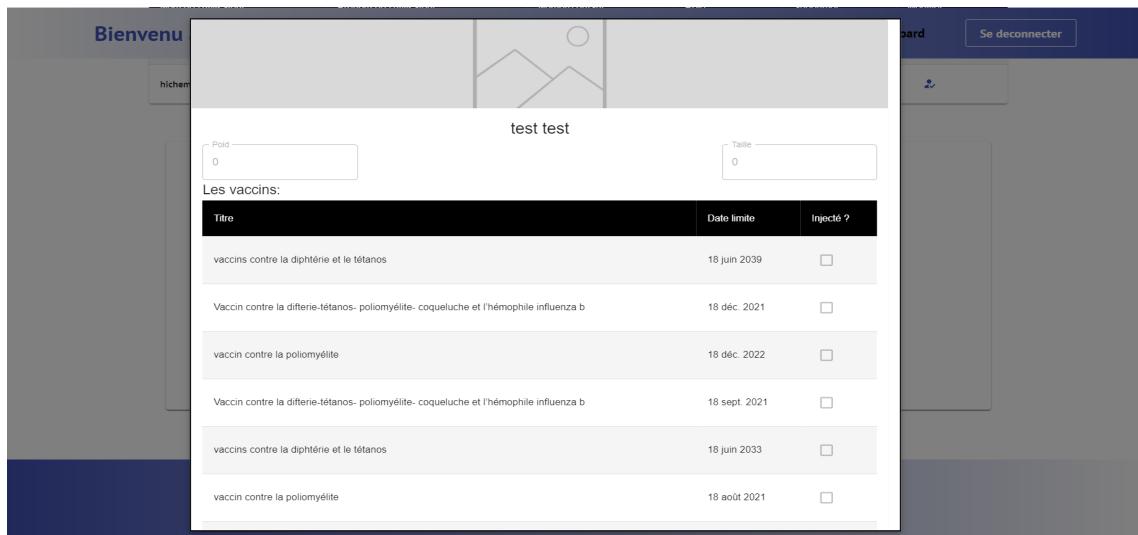


FIGURE 6.4 – Interface Vaccins

A partir du dossier médical, le médecin a la possibilité de savoir les rendez-vous déjà prisent par chaque patient.

Les rendez-vous:

19/06/2021, 08:00:00

FIGURE 6.5 – Interface RDV

Après la précision des médicaments et leurs dosages, le médecin peut consulter la liste des ordonnances créé.

Les ordonnances:		
IMPRIMER SUPPRIMER		
Date du traitement	Medicament	Dosage
20/06/2021, 18:01:38	doliprane	3 fois par jour
20/06/2021, 18:01:38	test	12

Ordonnance créée le 20/06/2021

IMPRIMER SUPPRIMER		
Date du traitement	Medicament	Dosage
20/06/2021, 18:10:32	augmentin	2 fois par jour
20/06/2021, 18:10:32	ferex	3 fois par jour

Ordonnance créée le 20/06/2021

IMPRIMER SUPPRIMER		
Date du traitement	Medicament	Dosage
20/06/2021, 18:10:50	clamoxyl 500	une fois par jour

FIGURE 6.6 – Interface des ordonnances

Conclusion

Au cours de ce chapitre, nous avons présenté le quatrième sprint dans lequel nous avons présenté l'outil utilisé pour l'analyse des données ainsi que la gestion et suivi du dossier médical.

Conclusion générale

Au terme de ce rapport, nous pouvons conclure que ce projet nous a donné une occasion opportune nous permettant de confronter l'acquis théorique à l'environnement pratique.

En effet, ce projet nous a permis de mettre en pratique nos connaissances acquises tout au long de notre formation et de les enrichir grâce à la réalisation d'un projet qui consiste en la conception et la réalisation d'une application web de conception et implémentation d'une application de gestion d'un cabinet de médecin pédiatre et d'analyse des données des patients.

Nous avons essayé tout au long de notre travail de construire notre projet incrément par incrément en utilisant la méthodologie Scrum. Nous avons commencé en premier lieu par comprendre le contexte général de notre projet et pourquoi ce projet doit-il réaliser, ensuite nous avons fait une étude préalable afin de découvrir les technologies et les outils à utiliser et choisir la méthodologie à suivre.

Nous avons passé par la suite à analyser les besoins en déduisant le Product Backlog, la priorisation de ce dernier nous a mené à une première planification. Nous avons, par ailleurs, étudié les différents Sprints de notre backlog, pour obtenir une conception. Ensuite, pour chaque sprint, nous avons abordé l'étape du développement au cours de laquelle nous avons expliqué les principales tâches effectuées.

Comme perspectives de travaux futurs, nous proposons d'enrichir cette application en s'intéressant à certaines fonctionnalités. Pour étendre notre solution, nous proposons d'ajouter quelques fonctionnalités par exemple le paiement en ligne de la consultation.

Netographie

- [1] <https://agiliste.fr/introduction-methodes-agiles/#:~:text=Le%20terme%20%22agile%22%20d%C3%A9finit%20une,de%20%22gestion%20de%20produit%22>. Consulté le 29/04/2021
- [2] <https://www.planzone.fr/blog/quest-ce-que-la-methodologie-agile> Consulté le 29/04/2021
- [3] <https://www.atlassian.com/fr/agile/scrum/sprints> Consulté le 29/04/2021
- [4] <https://blog.dcube.fr/index.php/2014/04/28/scrum-vs-cycle-en-v-2-2/> Consulté le 30/04/2021
- [5] https://www.google.com/url?sa=i&url=https%3A%2F%2Fblog.dcube.fr%2Findex.php%2F2014%2F04%2F28%2Fscrum-vs-cycle-en-v-2-2%2F&psig=A0vVaw0eojnCtjZrf0pJ_ICk292j&ust=1620905052652000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCJCChoKExPACFQAA Consulté le 30/04/2021
- [6] https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm Consulté le 01/05/2021
- [7] <https://www.mongodb.com/fr/nosql-explained> Consulté le 01/05/2021
- [8] <https://www.guru99.com/node-js-express.html#:~:text=js%3F-,Express.,known%20as%20the%20MEAN%20stack> Consulté le 01/05/2021
- [9] <https://www.spi-informatique.fr/2018/05/03/temoignage-1/#:~:text=React%20est%20une%20biblioth%C3%A8que%20qui,un%20framework%20MVC%20comme%20AngularJS.>

&text=js%20et%20React%209 Consulté le 02/05/2021

[10] <https://www.usabilis.com/material-design-lui-selon-google/> Consulté le 02/05/2021

[11] <https://code.visualstudio.com/docs> Consulté le 02/05/2021

[12] <https://www.atlassian.com/fr/git/tutorials/what-is-git> Consulté le 02/05/2021