|  |  |  |
| --- | --- | --- |
| IsetFinal_mai_2005%20copier2 | **Ministère de l'Enseignement Supérieur, de la Recherche Scientifique**  **Direction Générale des Études Technologiques** | Logo  Entreprise |
| **Institut Supérieur des Études Technologiques de Djerba**  **Département Technologies de l’informatique** | | |

|  |  |
| --- | --- |
| **Code projet** | **…………** |

**Rapport de**

**STAGE DE FIN D’ETUDES**

**Présenté en vue de l’obtention du diplôme de**

**Licence Appliquée en Technologies de**

**l’Informatique**

**Parcours: DSI**

**Conception et développement d’applications web et Mobile pour la gestion des cabinets médicaux**

**Elaboré par :**

**Sarah BEN DAOUED**

**Malek KHANNOUSSI**

**Encadré par :**

**M. Aymen BOUREGBA**

**Effectué à :**

**Entreprise : VINTONET**

**Encadreur : M. Mohamed Thabet JMAEL**

**Année universitaire : 2020/2021**

**Sommaire**

# Liste des figures

**Remerciements**

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

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

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

**Dédicaces**

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

**À Mes chers parents**

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

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

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

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

**À tous mes ami(e)s**

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

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

**À M. Aymen BOUREGBA**

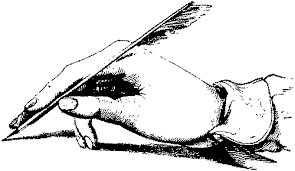
Merci pour votre bonne volonté d'accepter de nous encadrer, pour tout le temps qu’il nous a octroyé et pour tous les conseils qu’il nous a prodigué.

Vous êtes l’enseignant qui a réussi à m’inspirer, à me donner confiance en moi et en l’avenir, mais aussi qui a réussi à me donner l’envie d’apprendre.

**À mes enseignants**

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

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

****

***Sarah BEN DAOUED***

**Dédicaces**

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

**À Ma Chère Mère**

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

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

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

Que Dieu, accorde la santé et le bonheur.

**À l’âme de mon père**

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

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

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

**À Mes Frères et Ma Sœur**

Vous étiez toujours présents pour m’aider et m’encourager.

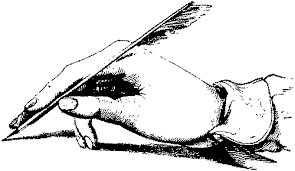
Sachiez que vous serez toujours dans mon cœur.

**À M. Aymen BOUREGBA**

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

**À mes amis**

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

****

***Malek KHANNOUSSI***

**Introduction Générale**

De nos jours, la digitalisation est devenue indispensable dans la vie professionnelle et la vie quotidienne.

La digitalisation est devenue une solution de publicité et de communication très rapide et efficace. C’est pour raison que la majorité des différentes organisations ont recours surtout à des sites web pour mieux diriger leurs affaires.

Une telle technologie, peut être utilisée pour mettre en place un système de **gestion des cabinets médicaux**.

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

En effet, nos applications permettront de gérer les patients, les rendez-vous, les dossiers médicaux... Afin de faciliter la réservation des rendez-vous et l’échange des données entre les médecins et les patients.

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

* Dans le premier chapitre, nous présenterons d’une manière générale l’université, l’organisme d’accueil, le cadre général de travail, l’étude de l’existant, le projet réalisé en expliquant la méthodologie adoptée, l’architecture, l’étude technologique, et en fin l’environnement logiciel et matériel.
* Dans le deuxième chapitre, nous présentons les acteurs, l’analyse des besoins, la planification des releases, le diagramme de cas d’utilisation et le diagramme de classe.
* Le troisième chapitre est dédié au release 1 pour la gestion des comptes dont on présente l’organisation des sprints, le backlog des sprints, la conception, et le travail dans chaque sprint.
* Le quatrième chapitre est dédié au release 2 pour la relation médecin/patient dont on présente l’organisation des sprints, le backlog des sprints, la conception, et le travail dans chaque sprint.
* Le cinquième chapitre est dédié au release 3 pour la gestion des rendez-vous dont on présente l’organisation des sprints, le backlog des sprints, la conception, et le travail dans chaque sprint.
* Le sixième chapitre est dédié au release 4 pour la gestion des consultations dont on présente l’organisation des sprints, le backlog des sprints, la conception, et le travail dans chaque sprint.

**Chapitre 1: Etude préalable**

1. **Introduction**

Dans ce chapitre, nous présentons l’université, l’organisme d’accueil, ainsi que le projet en le plaçant dans son cadre général.

Ensuite, nous recommandons d’introduire l’environnement de projet comme suit:

Présenter l’étude de l’existant, les méthodologies adoptées, l’architecture ainsi et l’étude technologique.

Enfin nous présentons l’environnement logiciel et matériel pour mener à bien ce projet.

1. **Présentation de l’université**

L'institut supérieur des études technologiques de Djerba (ISET Djerba), a été créé le 11 mai 2000, fait partie d'un réseau de 25 instituts supérieurs des études technologiques. Dans le cadre du LMD, il dispense une formation supérieure technologique dans les spécialités suivantes : Technologies de l'Informatique Génie électrique, Génie mécanique, et Sciences économique et gestion.

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

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

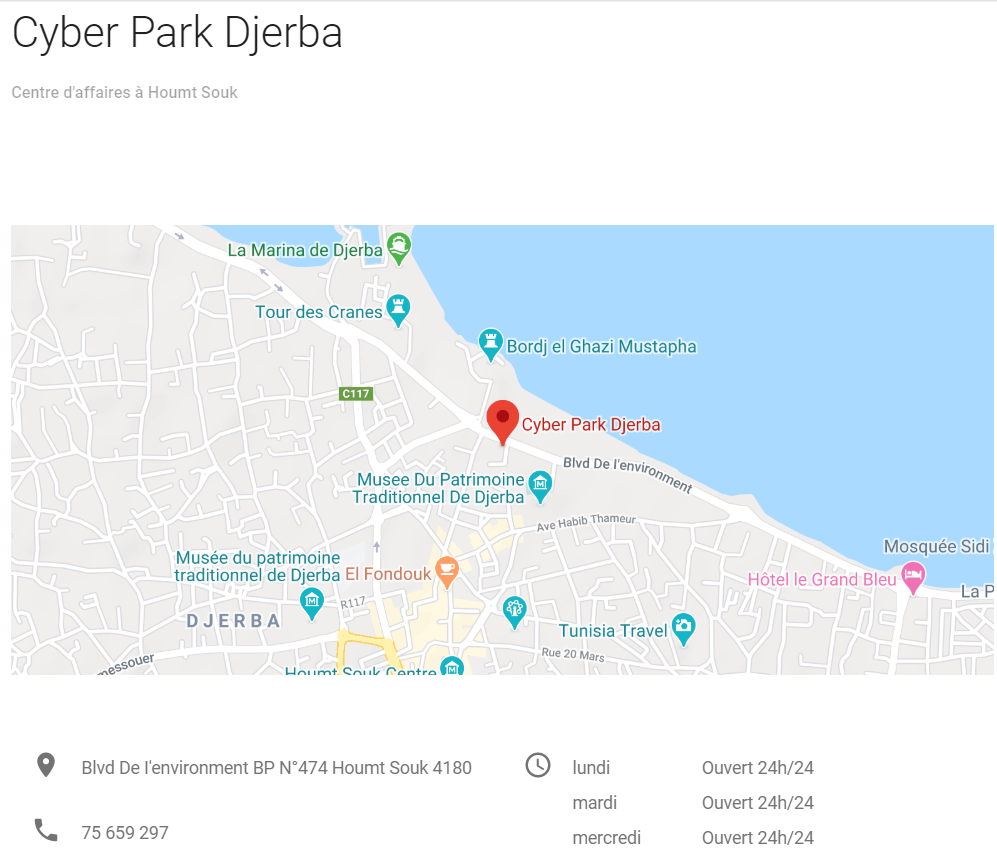
L’iset jb situé à Midoun, Djerba Midun 4116.

1. **Présentation de l’organisme d’accueil**

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

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



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



* **Organisation de l’organisme d’accueil**
* **L’équipe**

Dans cette société il y a plusieurs organismes:

* Directeur générale
* Département commercial et contact client.
* Département développement
* **Réalisation d’un projet**

La réalisation d’un projet se fait comme suit :

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

1. **Cadre générale de travail**

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

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

1. **Présentation du projet**

* **Contexte**

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

* **Objectif**

Regrouper les médecins et leurs patients afin de leur faciliter l’échange des données de façon, l’interaction peut devenir instantanée afin d’éviter par exemple le risque de prise des médicaments qui ne peuvent pas être utilisés ensemble.

D’autre part, l'application offre la possibilité de réserver un rendez-vous en ligne chez un médecin.

* **Les apports de projets**

Les apports de projets se divisent en 2 parties :

* **Pour les médecins :** la suivie des dossiers médicaux de ses patients.
* **Pour les secrétaires :** l’organisation des dossiers médicaux.
* **Pour les patients :** la prise des rendez-vous et la suivie de son dossier médicale.
* **Cible de projet**

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

Les cibles de projets sont principalement les médecins et leurs patients.

1. **Etude de l’existant**

Nous avons beaucoup de personnes qui consultent des médecins cependant il est impossible d’échanger instantanément des données entre eux.

Cependant, les médecins suivre les informations de leur patient sur la fiche médicale.

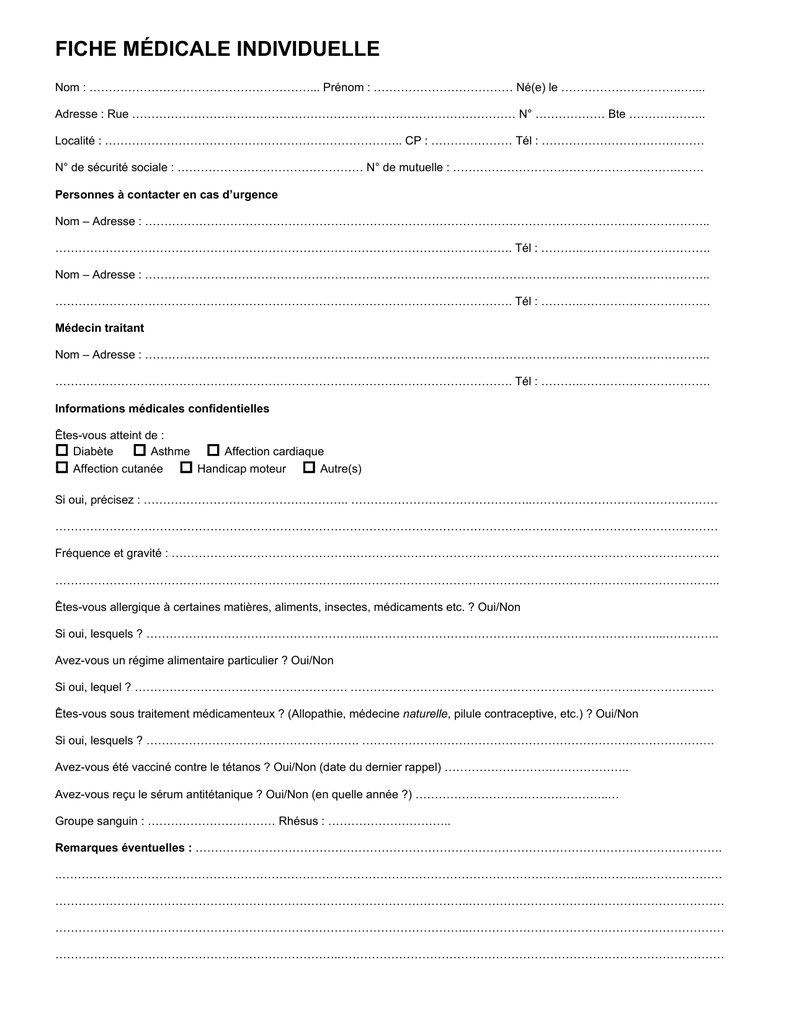
En effet, les fiches médicales ne sont pas suffisantes pour consulter tous les informations de patient.

Par exemple, le médecin ne peut pas consulter les antécédents médicaux du patient et les médicaments qu’il utilise chez d’autres médecins.

D’où proviennent les risques liés à la prise des médicaments qui ne peuvent pas être utilisés ensemble.

De plus, la fiche peut être facilement perdue et la perte d’une fiche implique la perte du patient.

En fin, l’organisation des rendez-vous et l’organisation de fiches médicales sont les soucis les plus difficiles à gérer dans les cabinets médicaux.



* **Inconvénients et solutions proposé**

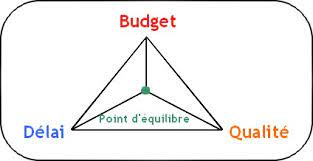
|  |  |
| --- | --- |
| Les inconvénients | Nos solutions proposés |
| La fiche médicale ne contient pas toutes les informations nécessaires sur le patient. | Dans l’application, on trouve toutes les informations de patient. |
| Le médecin ne peut pas suivre le dossier de patient chez les autres médecins(les antécédents, les médicaments utilisés,..) | Le médecin suivre le dossier médical de son patient sur l’application. |
| Les informations de patient se trouvent dans une fiche qui peut être perdu. | Les informations de patient se trouvent dans la plateforme |
| Le médecin ne peut pas consulter les antécédents médicaux du patient et les médicaments qu’il utilise chez d’autres médecins. | Le médecin peut suivre tous le dossier médical de son patient même chez les autres médecins. |
| Le patient ne peut pas proposer un rendez-vous. | Le patient peut proposer un rendez-vous chez un médecin. |
| Le patient ne peut pas consulter l’état de la salle d’attente. | Le patient peut suivre instantanément l’état de la salle d’attente. |
| Le patient ne peut pas suivre son dossier médical. | Le patient peut suivre son dossier médical à tout moment. |

1. **Méthodologies adoptés**

* **Gestion de projet :**

La gestion de projet sert à évoluer les environnements qui amène à une évolution constante.

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



* **Scrum :**

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

* **Les Avantages :**
* **Comprendre le travail et les tâches à effectuer :**

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

* **Transparence et respect** **:**

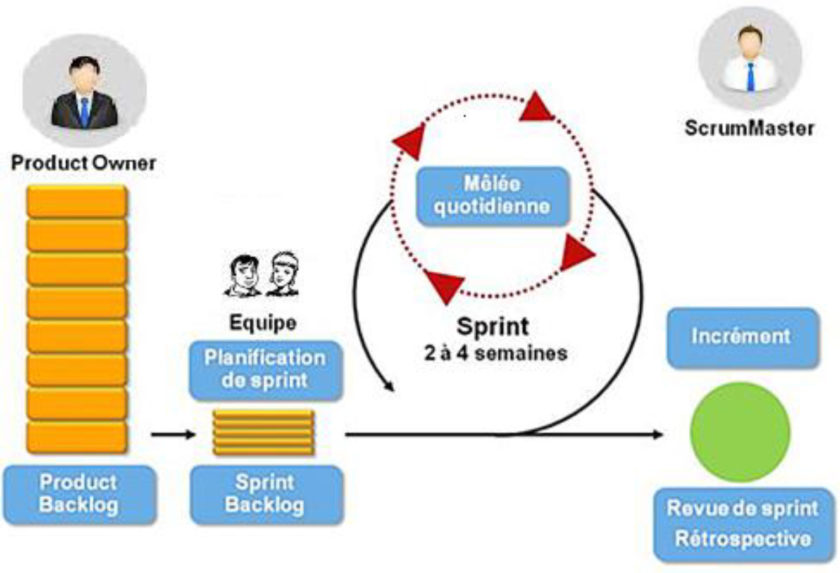
**Scrum** exige de la transparence. Les membres de l’équipe doivent savoir ce que les autres accomplissent et le résultat qu’ils peuvent en attendre. Mais chacun peut déterminer comment il accomplit sa tâche.

* **Visibilité continue :**

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

* **Focus et flexibilité**

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



* **UML :**

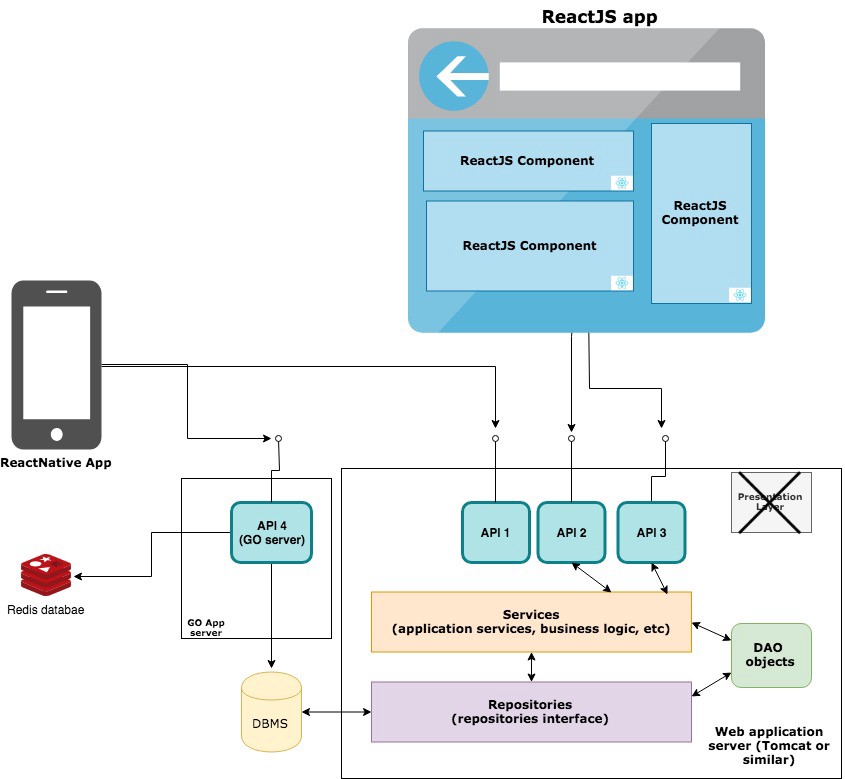
**UML**, On le traduit par « Langage de modélisation unifié ». La notation **UML** est un langage visuel constitué d’un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. **UML** nous fournit donc des diagrammes pour représenter le logiciel à développer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

1. **Architecture de l’application**

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

* **Architecture 3-tiers :**

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

****

1. **Etude technologique**

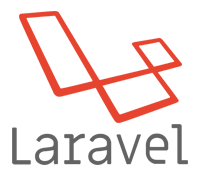
* **Outils de développement** **:**
* **L’application back end :**

L’outil qu’on a utilisé pour développer l’application back end est le Framework **laravel**. **Laravel** est un Framework open source écrit en **PhP** !

**Laravel** est aujourd’hui le produit **PhP** le mieux noté sur le site de **Github**.

En effet, **laravel** présente plusieurs avantages c’est pour cette raison on a le choisis pour développer notre applications comme avantages on peut citer :

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

****

* **L’application front end :**

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

**React** est une bibliothèque qui ne gère que l'interface de l'application, considéré comme la vue dans le modèle **MVC**. Elle peut ainsi être utilisée avec une autre bibliothèque ou un Framework **MVC** comme **AngularJS**. Cette bibliothèque se démarque de ses concurrents par sa flexibilité et ses performances travaillant avec un **DOM** virtuel et en ne mettant à jour le rendu dans le navigateur qu'en cas de nécessité.

* **Avantages :**
* **Très rapide :**

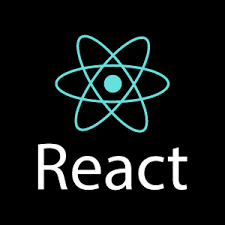
**ReactJS** crée son propre **DOM** virtuel. Cette approche vous donne énormément de flexibilité et des performances exceptionnelles, car **ReactJS** calcule quel changement dans le **DOM** a besoin d’être fait, et change juste la partie qui a besoin d’être mise à jour. De cette façon, **ReactJS** évite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

* **L’intelligibilité :**

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

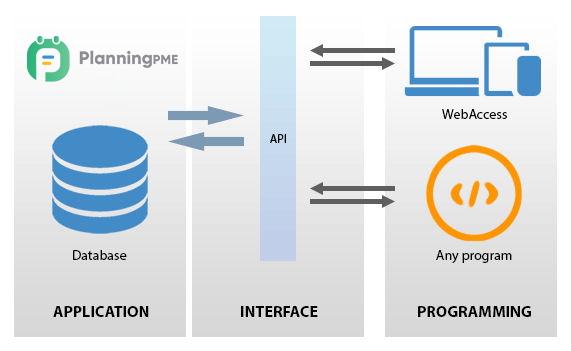
* **Le contenu est référençable :**

C’est la fonctionnalité qui fait la différence par rapport aux autres frameworks. Grâce à l’utilisation d’un serveur **Node**, le code va pouvoir être généré côté client ET côté serveur (un peu comme du PHP donc… tout en gardant les avantages du JavaScript !) à la différence des autres frameworks JS traditionnels (Backbone.js, AngularJS, Ember.js, etc.) qui de manière native exécutent le code seulement côté client (dans le navigateur).

****

* **API REST:**

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

****

1. **Environnements logiciels**

* **Drawio**

**Drawio** est une application de création de diagrammes et schéma.

****

* **Visual studio code**

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

****

* **Xampp**

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

****

* **Discord**

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

****

* **Github**

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

****

* **Word 2010**

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

****

1. **Environnement matériel**

|  |  |  |
| --- | --- | --- |
| Marque | Dell | HP pavillon |
| Processeur | I7-6700HQ |  |
| RAM | 16 GO |  |
| Disque Dur | 1 TO |  |
| Carte Graphique | 960M 4 GB |  |
| Système d’exploitation | Windows 10 |  |

1. **Conclusion**

Dans ce chapitre, on a présenté le cadre du projet et la méthode utiliser pour sa réalisation. Le prochain chapitre se concentrera sur la mise en œuvre du projet.

**Chapitre 2: Mise en œuvre du projet**

1. **Introduction**

La phase de mise en œuvre du projet est l’une des phases les plus importantes en matière de développement des sites web.

Dans ce chapitre, nous commençons par l’identification des acteurs.

Ensuite, nous présentons l’analyse des besoins dont on cite les besoins fonctionnels, non fonctionnels, ainsi que le backlog.

Enfin, la planification des releases et la conception.

1. **Identification des acteurs**

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

Dans notre système, on peut extraire trois acteurs principaux :

* Le patient
* Le médecin
* La secrétaire

1. **Analyse des besoins**

* **Fonctionnels :**

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

* Pour le patient :
* **L’application offre au patient :**
* Gérer son compte.
* Consulter le journal d’accès (l’accès d’un médecin à ses informations).
* Parcourir la liste de ses médecins.
* Parcourir la liste de ses rappels.
* Chercher un médecin.
* Consulter l’état de la salle d’attente (Combien de patients y a-t-il ?) et prendre un rendez-vous.
* Pour la secrétaire :
* **L’application offre au secrétaire :**
* Gérer son compte.
* Ajouter un nouveau patient.
* Réserver un rendez-vous pour un patient.
* Consulter l’état de la salle d’attente (Combien de patients y a-t-il ?).
* Gérer l’état du patient (l’ajouter à la salle d’attente, à la salle de la consultation, à la salle d’opération…).
* Consulter les dépenses du patient.
* Pour le médecin:
* **L’application offre au médecin :**
* Gérer son compte.
* Faire tout ce que peut faire la secrétaire.
* Il peut consulter l’espace de ses patients (Consulter ses antécédents, ses consultations, ses ordonnances, ses certificats, ses examens radiologiques ou biologiques,..).
* En envoyant une notification au patient, il peut avoir accès à ses informations médicales issues d’un autre médecin. (Consulter antécédents, consultations, ordonnances, certificats, examens radiologiques ou biologiques, …).
* Choisir la secrétaire.
* Faire une consultation. Ajouter des actions (consultations, ordonnances, certificats, examens radiologiques ou biologiques, ...).
* Préciser la facture du patient.
* **Non-fonctionnels :**

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

* **La sécurité :** mettre en place un mécanisme d’authentification qui permet à chaque utilisateur de se connecter pour consulter en sécurité son profil.
* **La convivialité :** L’application doit être facile à utiliser. Il doit présenter un enchainement logique entre les interfaces et un ensemble de liens suffisants pour assurer une navigation rapide et un texte compréhensible, lisible.
* **La disponibilité :** Lorsque n’importe quel utilisateur désire consulter l’application, elle doit être disponible.
* **Temps de réponse :** Le temps de réponse doit être le plus court possible.
* **Backlog de projet :**

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

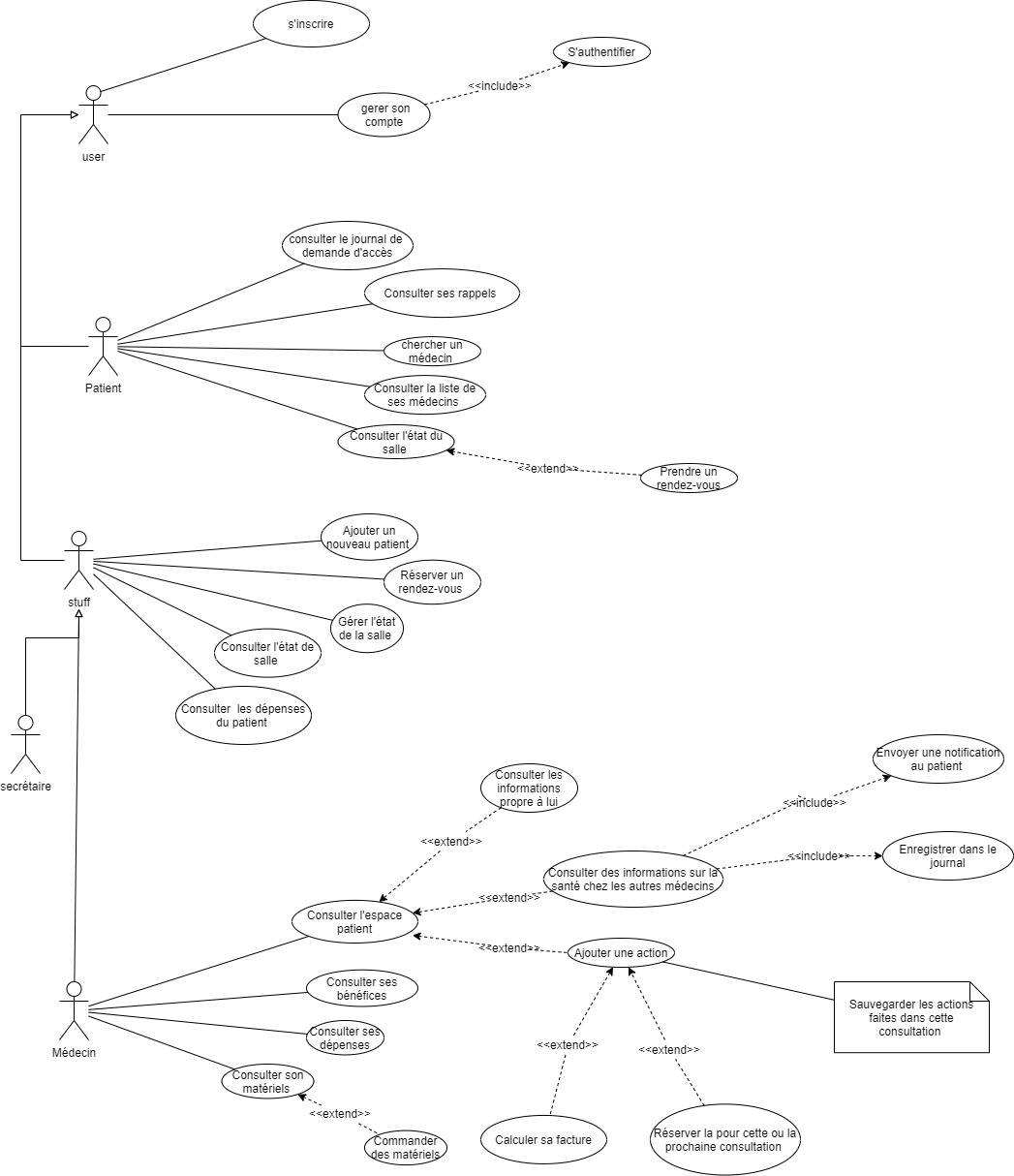
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Product Backlog | | | | |
| ID | **User Story** | **Thème** | **Priorité** | **Sprint** |
| 1.1 | **En tant qu’utilisateur je veux s’inscrire.** | **Gérer les comptes.** | **1** | **1** |
| 1.2 | **En tant qu’utilisateur je veux se connecter.** | **1** |
| 1.3 | **En tant qu’utilisateur je veux gérer mon compte.** | **1** |
| 1.4 | **En tant que médecin ou secrétaire je peux créer un compte pour un patient.** | **1** |
| 2.1 | **En tant que patient je peux ajouter un médecin à ma liste des médecins.** | **Gérer médecins/ patients** | **2** | **2** |
| 2.2 | **En tant que médecin je peux ajouter un patient à ma liste des patients.** | **2** |
| 2.3 | **En tant que patient je peux consulter la liste des médecins.** | **2** |
| 2.4 | **En tant que médecin je peux consulter la liste des patients.** | **2** |
| 2.5 | **En tant que patient je peux chercher un médecin.** | **2** |
| 2.6 | **En tant que médecin je peux chercher un patient dans ma liste de patient.** | **2** |
| 3.1 | **En tant que patient je peux réserver un rendez-vous chez un médecin.** | **Gérer rendez-vous** | **3** | **3** |
| 3.2 | **En tant que médecin je peux réserver un rendez-vous pour un patient.** | **3** |
| 3.3 | **En tant que patient je peux consulter mes rendez-vous chez un médecin.** | **3** |
| 3.4 | **En tant que médecin je peux consulter mes rendez-vous.** | **3** |
| 4.1 | **En tant que médecin je peux examiner un patient.** | **Examiner un patient** | **4** | **4** |
| 4.2 | **En tant que médecin je peux consulter tous les informations sur mon patient.** | **4** |
| 4.3 | **En tant que médecin je peux ajouter une nouvelle information pour mon patient.** | **4** |
| 5.1 | **En tant que patient je peux consulter mes rappels.** | **Consulter rappel** | **5** | **5** |

1. **Planification des releases**

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

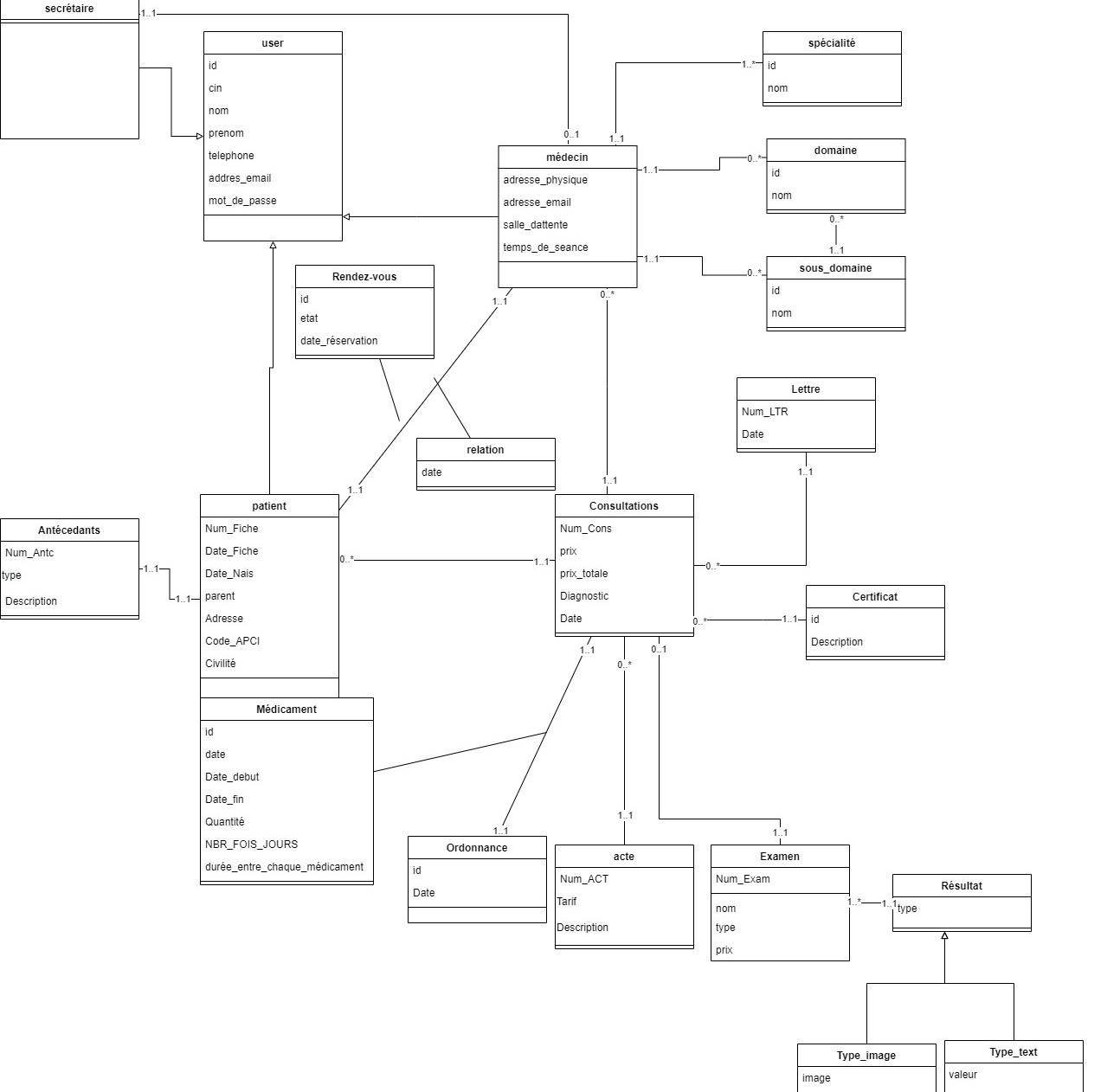
1. **Diagramme de cas d’utilisation**

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



1. **Diagramme de classe**

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

****

1. **Conclusion**

**Release 1 : Gestion de compte**

1. **Introduction**

Dans ce release en va présenter le déroulement pour développer la version de gestion de compte.

On a commencé par organiser les sprints.

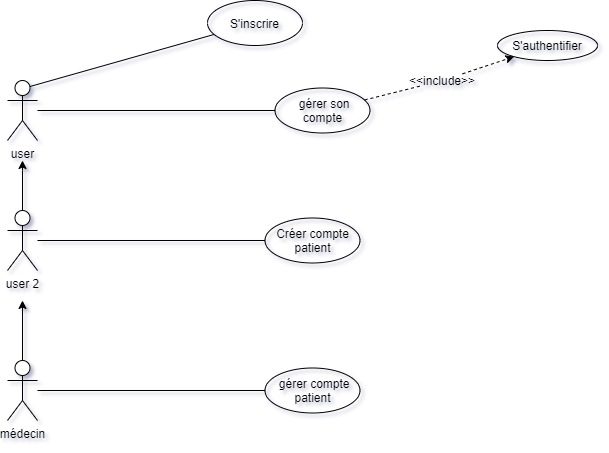
Ensuite, on a définir le backlog des sprints pour la version 1 ainsi que la conception.

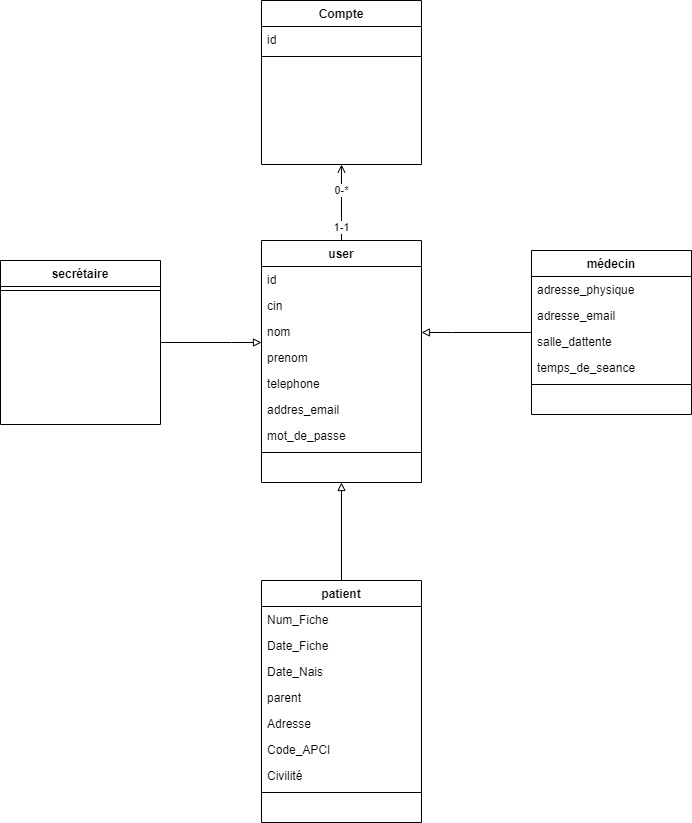
En fin, on a présenté chaque sprint.

1. **Organisations des sprints**
2. **Back log des sprints pour le release1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Product Backlog** | | | | |
| **ID** | **User Story** | **Thème** | **Priorité** | **Sprint** |
| **1.1** | **En tant qu’utilisateur je veux s’inscrire.** | **Gérer les comptes** | **1** | **1** |
| **1.2** | **En tant qu’utilisateur je veux se connecter.** | **1** |
| **1.3** | **En tant qu’utilisateur je veux gérer mon compte.** | **1** |
| **1.4** | **En tant que médecin ou secrétaire je peux créer un compte pour un patient.** | **1** |

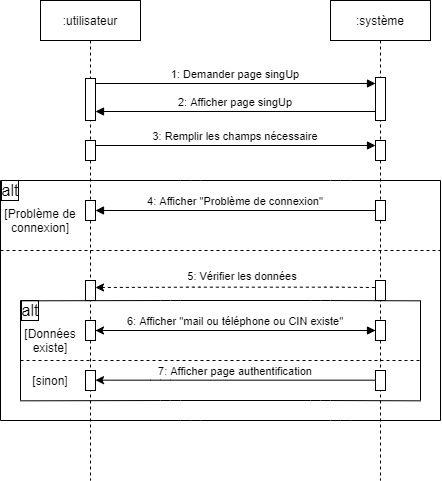
1. **Conception**

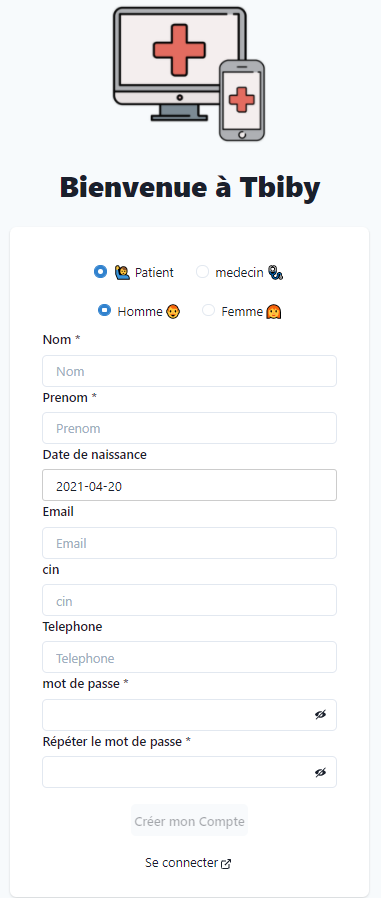
****

****

1. **Sprint1 : Inscription**

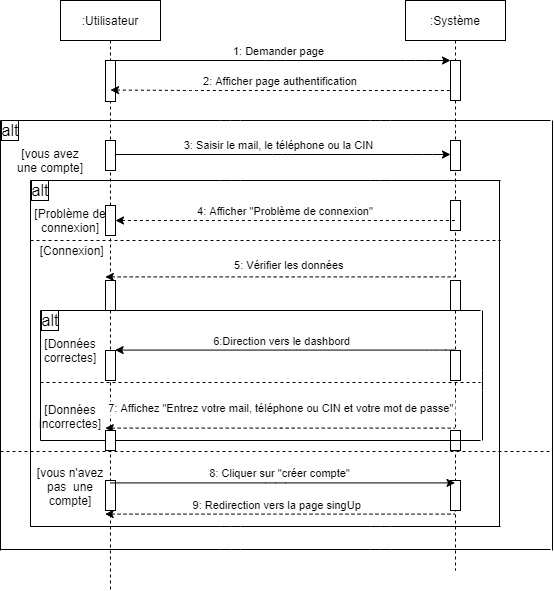
|  |  |
| --- | --- |
| **Inscription** | |
| **Titre** | **S’inscrire** |
| **But** | **Permet de s’inscrire à l’application** |
| **Acteurs** | **Utilisateur (médecin, patient, secrétaire)** |
| **Pré-condition** | **Connexion** |
| **Scenario nominal** | |
| 1. **L’utilisateur demande page d’inscription** 2. **Le système affiche page d’inscription** 3. **L’utilisateur rempli les champs nécessaire** 4. **Le système vérifie les données** | |
| **Scenario alternatifs** | |
| * 1. **Si le numéro de tél ou CIN ou le mail existe le système affiche « tél CIN ou mail existe déjà »**   **A chaque étape s’il y a un problème de connexion le système affiche « problème de connexion »** | |

****

****

1. **Sprint 2 : Authentification**

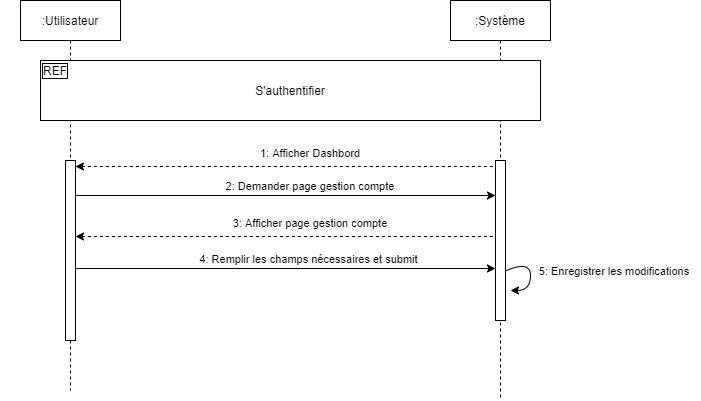
|  |  |
| --- | --- |
| **Authentification** | |
| **Titre** | **S’authentifier** |
| **But** | **Permet d’authentifier à l’application** |
| **Acteurs** | **Utilisateur (médecin, patient, secrétaire)** |
| **Pré-condition** | **Connexion** |
| **Scenario nominal** | |
| 1. **L’utilisateur demande page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

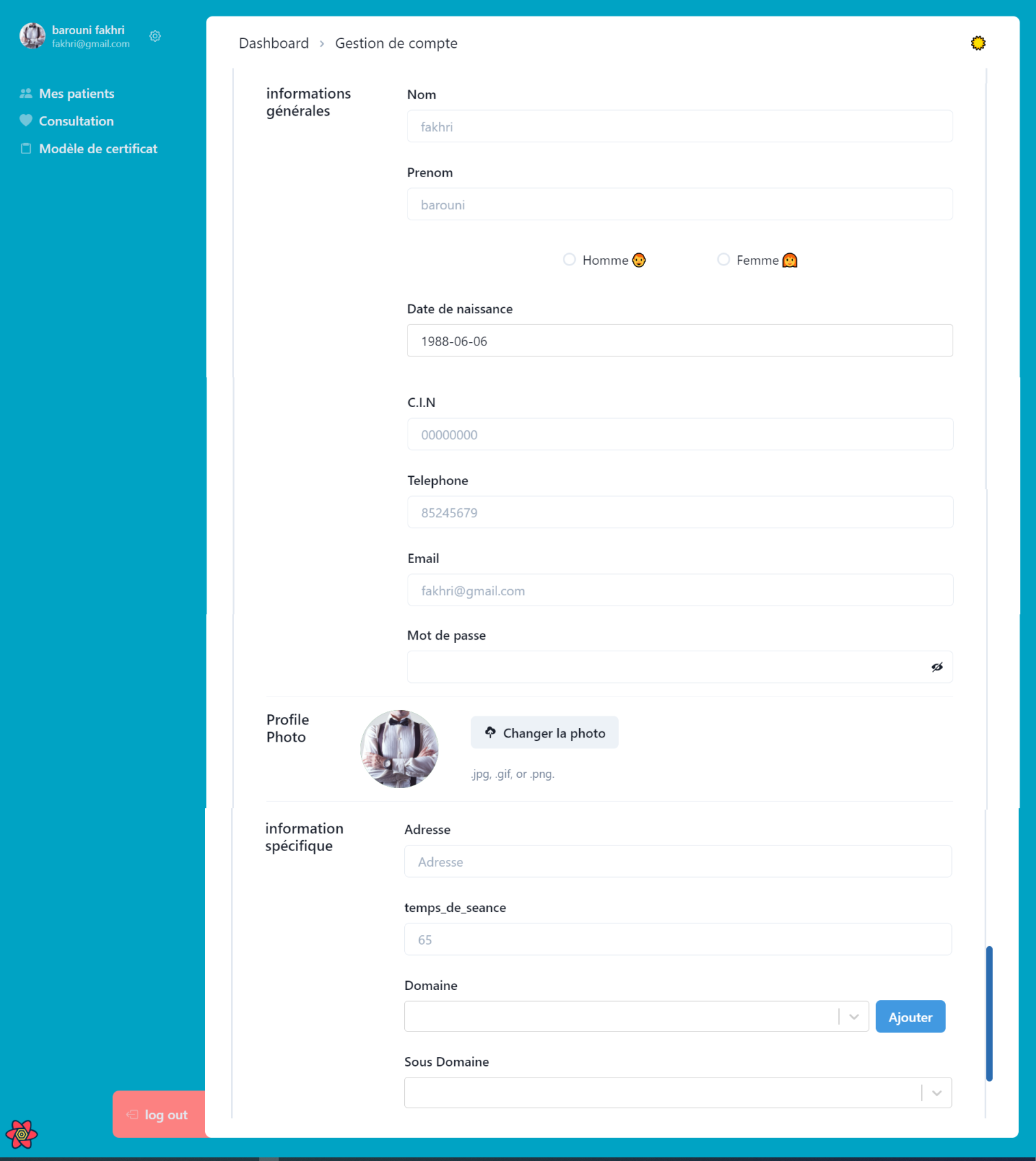
****

****

1. **Sprint 3 : Gestion de compte :**

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

****

****

1. **Conclusion**

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

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

**Release 2 : Relation entre patient et médecin**

1. **Introduction**

Dans ce release en va présenter le déroulement pour développer la version pour la relation entre le patient et le médecin.

On a commencé par organiser les sprints.

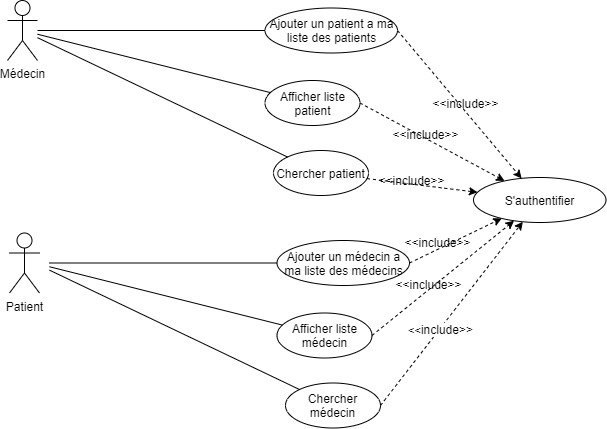
Ensuite, on a définir le backlog des sprints pour la version 2 ainsi que la conception.

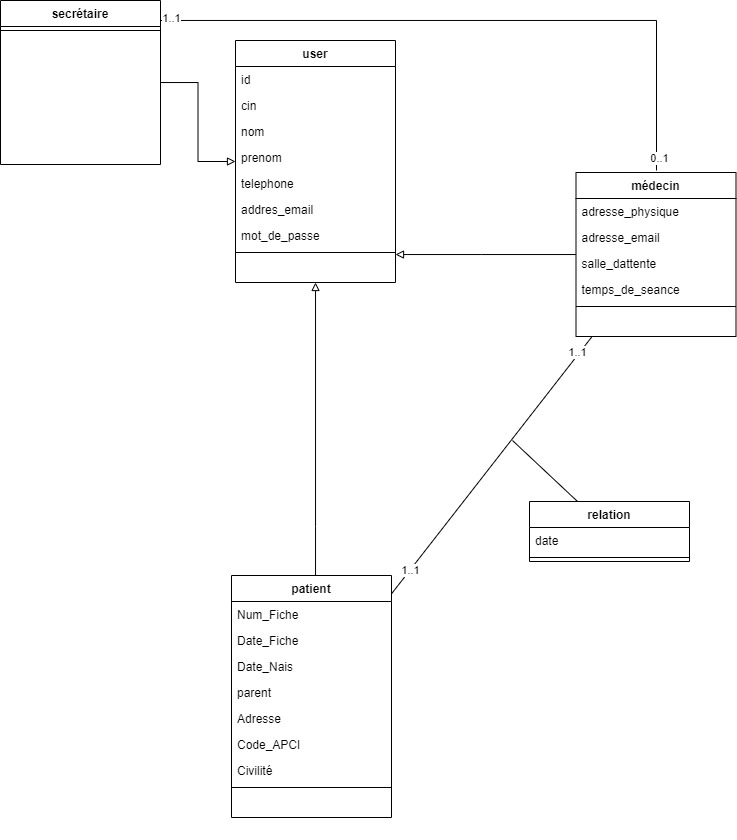
En fin, on a présenté chaque sprint.

1. **Organisations des sprints**
2. **Back log des sprints pour le release2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **2.1** | **En tant que patient je peux ajouter un médecin à ma liste des médecins.** | **Gérer médecins/ patients** | **2** |  |
| **2.2** | **En tant que médecin je peux ajouter un patient à ma liste des patients.** | **2** |
| **2.3** | **En tant que patient je peux consulter la liste des médecins.** | **2** |
| **2.4** | **En tant que médecin je peux consulter la liste des patients.** | **2** |
| **2.5** | **En tant que patient je peux chercher un médecin.** | **2** |
| **2.6** | **En tant que médecin je peux chercher un patient dans ma liste des patients.** | **2** |

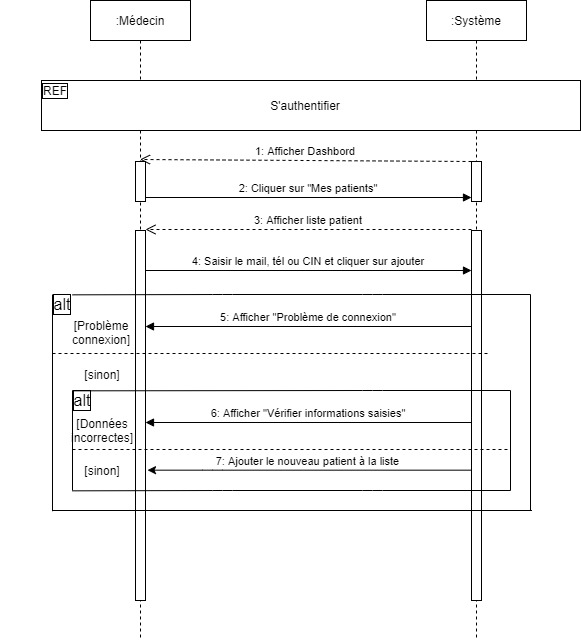
1. **Conception**

****

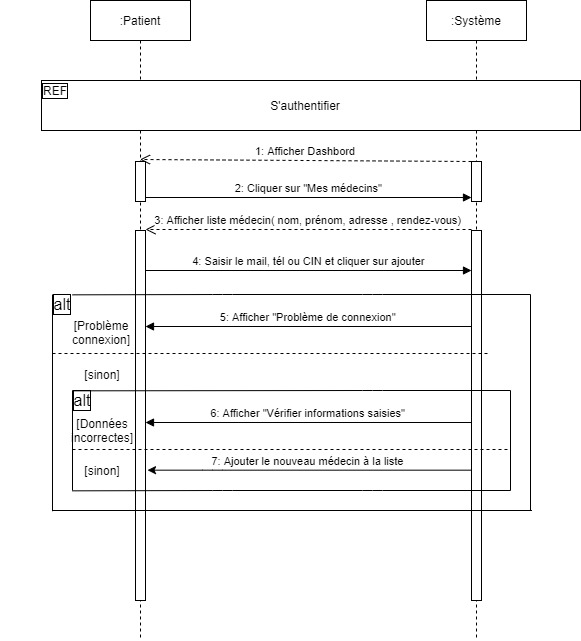
****

1. **Sprint 1 : Ajout médecin/ patient**

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

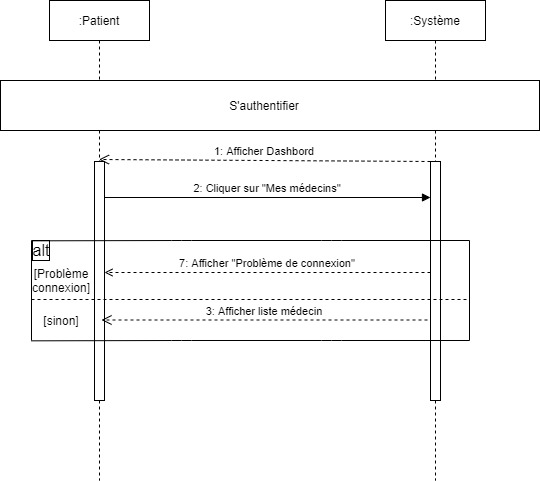
****

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

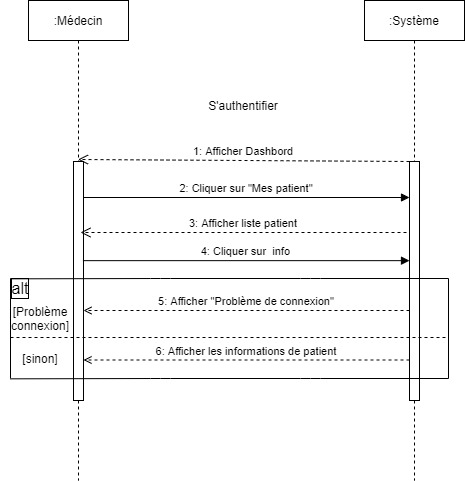
****

1. **Sprint 2 : Consultation liste médecin/ patient**

|  |  |
| --- | --- |
| **Consultation liste médecin** | |
| **Titre** | **Consultation liste médecin** |
| **But** | **Permet de consulter la liste des médecins** |
| **Acteurs** | **Patient** |
| **Pré-condition** | **Authentification, connexion** |
| **Scenario nominal** | |
| 1. **Le patient s’authentifier** 2. **Le système affiche Dashboard** 3. **Le patient demande mes médecins** 4. **Le système affiche page liste médecin** | |
| **Scenario alternatifs** | |
| **A chaque étape s’il y a un problème de connexion le système affiche « problème de connexion »** | |

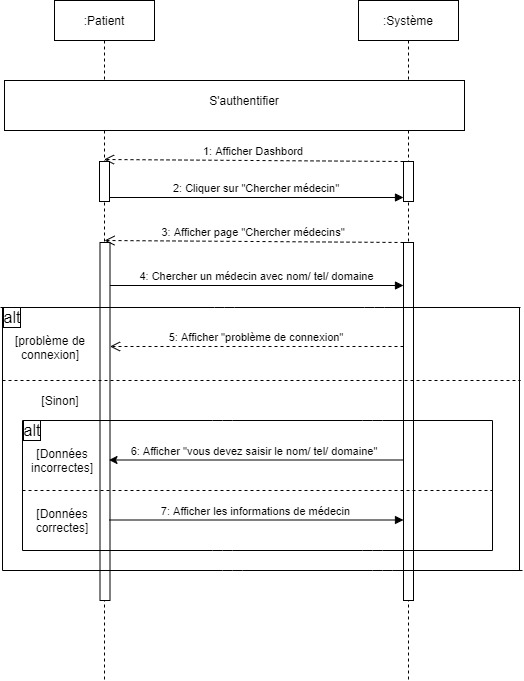
****

|  |  |
| --- | --- |
| **Consultions liste patient** | |
| **Titre** | **Consulter liste médecin patient** |
| **But** | **Permet au médecin de consulter la liste de ses patients** |
| **Acteurs** | **Médecin** |
| **Pré-condition** | **Authentification, connexion** |
| **Scenario nominal** | |
| 1. **Le médecin s’authentifier** 2. **Le système affiche Dashboard** 3. **Le médecin demande page mes patients / mes médecins** 4. **Le système affiche page liste médecin / patient** 5. **Le médecin clique sur info** 6. **Le système affiche les informations de patient** | |
| **Scenario alternatifs** | |
| **A chaque étape s’il y a un problème de connexion le système affiche « problème de connexion »** | |

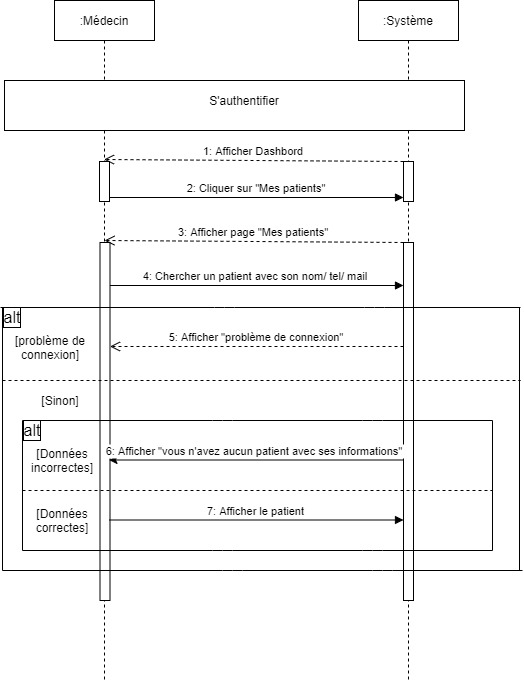
****

1. **Sprint 3 : Chercher médecin/ patient**

|  |  |
| --- | --- |
| **Chercher médecin** | |
| **Titre** | **Chercher médecin** |
| **But** | **Permet de chercher un médecin** |
| **Acteurs** | **Patient** |
| **Pré-condition** | **Authentification, connexion** |
| **Scenario nominal** | |
| 1. **Le patient s’authentifier** 2. **Le système affiche Dashboard** 3. **Le patient demande page chercher patients/ médecins** 4. **Le système affiche page chercher médecin / patient** 5. **Le patient saisie tél/mail/domaine pour chercher un médecin** 6. **Le système affiche le médecin** | |
| **Scenario alternatifs** | |
| **5.1 Si les informations sont incorrectes le système affiche « vous devez saisir le mail, tél ou domaine »**  **A chaque étape s’il y a un problème de connexion le système affiche « problème de connexion »** | |

****

|  |  |
| --- | --- |
| **Chercher Patient** | |
| **Titre** | **Chercher patient** |
| **But** | **Permet de chercher un patient** |
| **Acteurs** | **Médecin** |
| **Pré-condition** | **Authentification, connexion** |
| **Scenario nominal** | |
| 1. **Le médecin s’authentifier** 2. **Le système affiche Dashboard** 3. **Le médecin demande page mes patients** 4. **Le système affiche page mes patients** 5. **Le médecin saisie cin/tèl/mail pour chercher un patient** 6. **Le système affiche le patient/ médecin** | |
| **Scenario alternatifs** | |
| **5.1 Si données incorrectes le système affiche « Vous n’avez aucun patient avec ses informations »**  **A chaque étape s’il y a un problème de connexion le système affiche « problème de connexion »** | |

****

1. **Conclusion**

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

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

**Release 3 : Gestion rendez-vous**

1. **Introduction**

Dans ce release on va présenter le déroulement pour développer la version qui concerne la gestion de rendez-vous.

On a commencé par organiser les sprints.

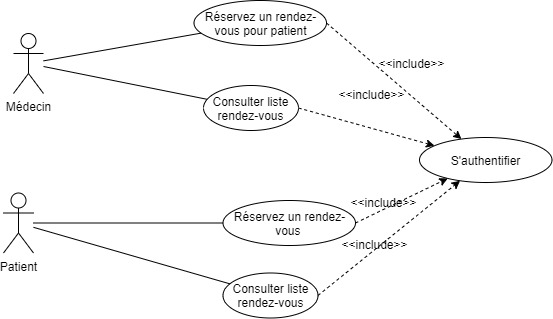
Ensuite, on a définir le backlog des sprints pour la version 3 ainsi que la conception.

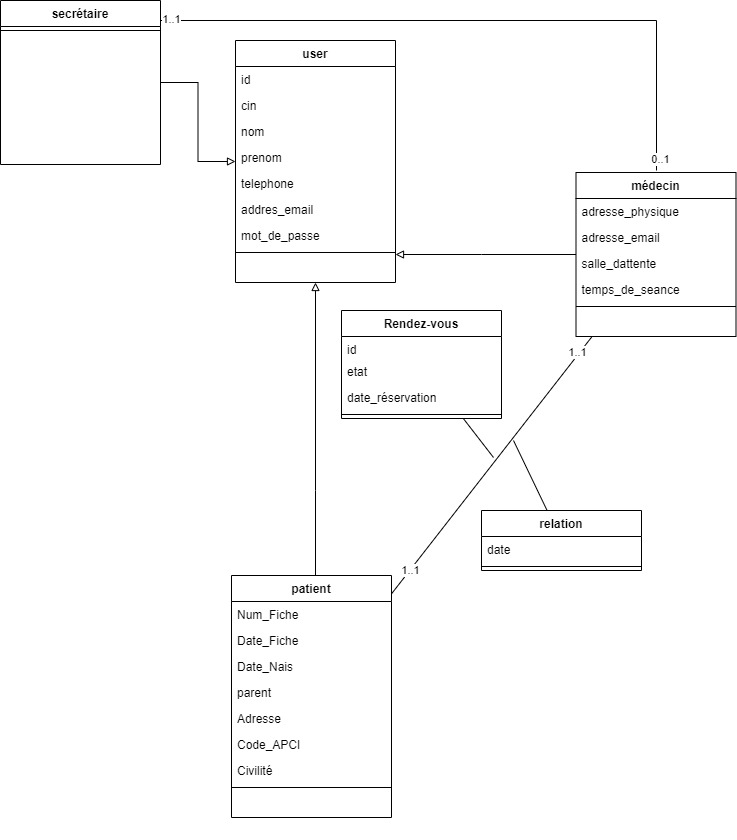
En fin, on a présenté chaque sprint.

1. **Organisations des sprints**
2. **Back log des sprints pour le release1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **3.1** | **En tant que patient je peux réserver un rendez-vous chez un médecin.** | **Gérer rendez-vous** | **3** | **3** |
| **3.2** | **En tant que médecin je peux réserver un rendez-vous pour un patient.** | **3** |
| **3.3** | **En tant que patient je peux consulter mes rendez-vous chez un médecin.** | **3** |
| **3.4** | **En tant que médecin je peux consulter mes rendez-vous.** | **3** |

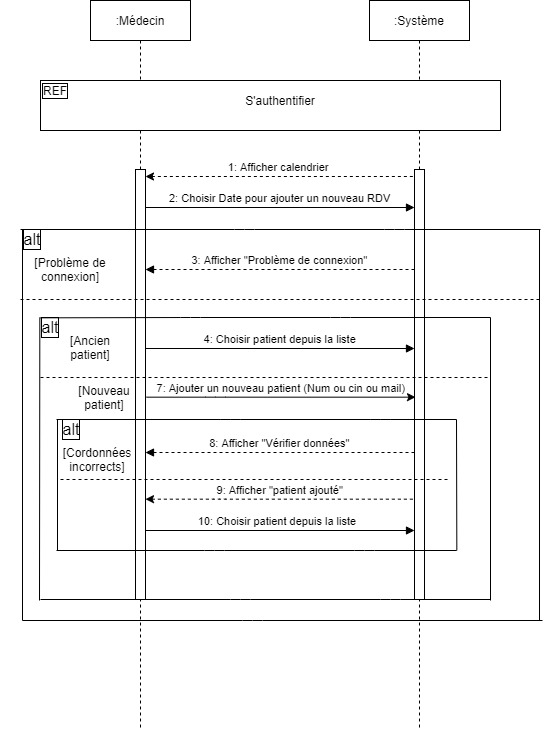
1. **Conception**

****

****

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

|  |  |
| --- | --- |
| **Réservation rendez-vous Patient** | |
| **Titre** | **Réserver rendez-vous** |
| **But** | **Permet de réserver un rendez-vous** |
| **Acteurs** | **Médecin** |
| **Pré-condition** | **Authentification, connexion** |
| **Scenario nominal** | |
| 1. **Le médecin s’authentifier** 2. **Le système affiche Dashboard** 3. **Le médecin choisir date et choisir un patient ou ajouter un nouveau puis le choisir depuis la liste** 4. **Le système ajoute le fichier à la liste** | |
| **Scenario alternatifs** | |
| **5.1 Si données incorrectes le système affiche « Vérifier données » sinon affiche « patient ajouté. Vous devez le choisir depuis la liste »**  **A chaque étape s’il y a un problème de connexion le système affiche « problème de connexion »** | |

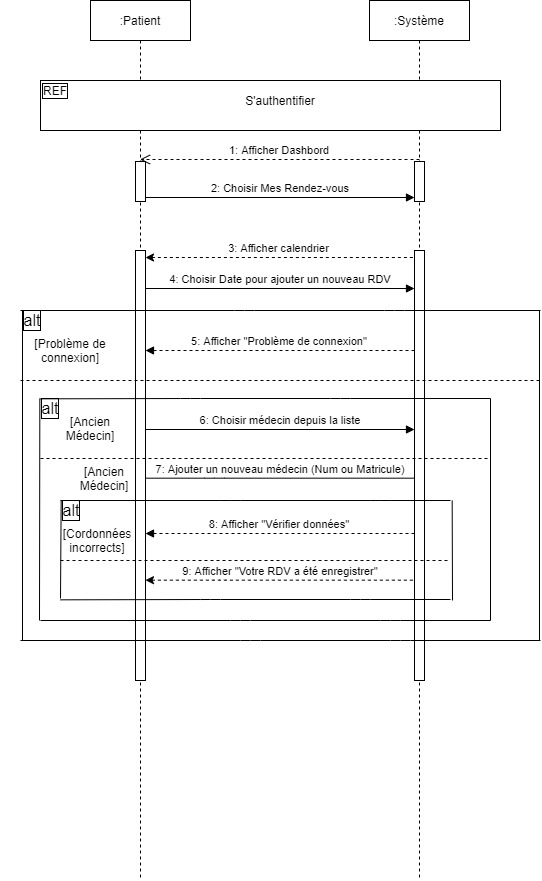
****

1. **Sprint 2 : Réservation rendez-vous patient chez un médecin**

|  |  |
| --- | --- |
| **Réservation rendez-vous** | |
| **Titre** | **Réserver rendez-vous** |
| **But** | **Permet de réserver un rendez-vous** |
| **Acteurs** | **Patient** |
| **Pré-condition** | **Authentification, connexion** |
| **Scenario nominal** | |
| 1. **Le patient s’authentifier** 2. **Le système affiche Dashboard** | |
| **Scenario alternatifs** | |
| **A chaque étape s’il y a un problème de connexion le système affiche « problème de connexion »** | |

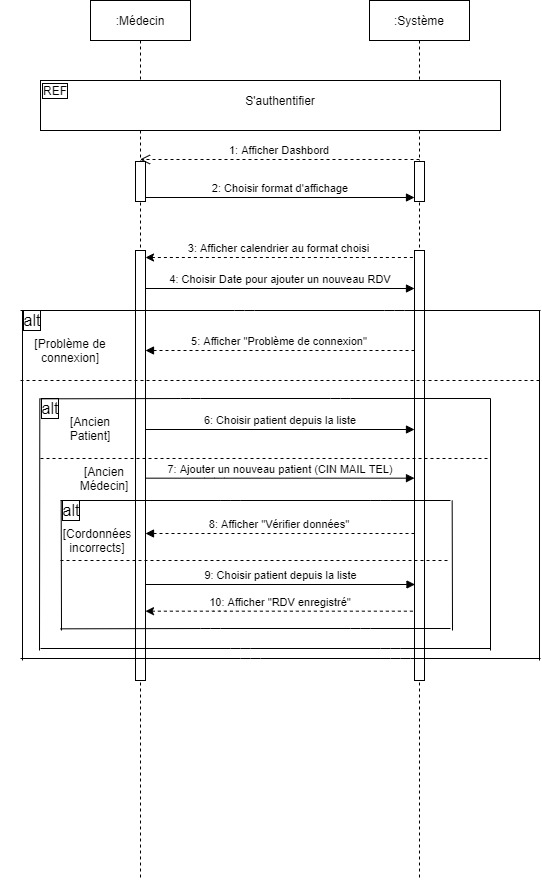
1. **Sprint 3 : Consultation liste rendez-vous (patient)**

|  |  |
| --- | --- |
| **Réservation rendez-vous Patient** | |
| **Titre** | **Consulter rendez-vous** |
| **But** | **Permet de consulter un rendez-vous** |
| **Acteurs** | **Patient** |
| **Pré-condition** | **Authentification, connexion** |
| **Scenario nominal** | |
| 1. **Le patient s’authentifier** 2. **Le système affiche Dashboard (calendrier)** 3. **Le patient choisir l’affichage par jour par semaine ou aujourd’hui** 4. **Le système répond au choix choisi.** | |
| **Scenario alternatifs** | |
| **A chaque étape s’il y a un problème de connexion le système affiche « problème de connexion »** | |

****

1. **Sprint 4 : Consultation liste rendez-vous (médecin)**

|  |  |
| --- | --- |
| **Consultation rendez-vous Médecin** | |
| **Titre** | **Consulter rendez-vous** |
| **But** | **Permet de consulter un rendez-vous** |
| **Acteurs** | **Médecin** |
| **Pré-condition** | **Authentification, connexion** |
| **Scenario nominal** | |
| 1. **Le médecin s’authentifier** 2. **Le système affiche Dashboard (calendrier)** 3. **Le médecin choisir le format d’affichage par jour, aujourd’hui ou par semaine** 4. **Le système répond au choix choisi par le médecin** | |
| **Scenario alternatifs** | |
| **2.1 On cliquant sur une date le médecin peut ajouter un rendez-vous**  **2.1.1 si ancien patient, il le choisi depuis la liste sinon il va l’ajouter puis le choisir.**  **A chaque étape s’il y a un problème de connexion le système affiche « problème de connexion »** | |

****

1. **Conclusion**

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

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

**Release 4 : Gestion consultation**

1. **Introduction**

Dans ce release en va présenter le déroulement pour développer la version qui concerne la gestion de consultation.

On a commencé par organiser les sprints.

Ensuite, on a définir le backlog des sprints pour la version 4 ainsi que la conception.

En fin, on a présenté chaque sprint.

1. **Organisations des sprints**
2. **Back log des sprints pour le release1**
3. **Sprint1 : Analyse et conception (DCU sprint1, description textuelle, DC, DS), Interface**
4. **Sprint 2, sprint3 ,…**
5. **Conclusion**

**Conclusion Générale et perspectives**

**Bibliographie**

<https://fr.reactjs.org/>

<https://laravel.com/>

<https://github.com/>

<https://www.youtube.com/watch?v=6c3l3C3_lvY&list=PLpG9MHuD988g-7vniIEs0yk8bmxL35v7c>