Cégep Marie - Victorin

Projet développement d’une application web

Conception projet en app web pour système hospitalière

Jimmy, Shaheem, Yanni - JSY

Table des matières

[Jimmy, Shaheem, Yanni - JSY 0](https://collegemv-my.sharepoint.com/personal/2233669_collegemv_qc_ca/Documents/Desktop/applicationWeb/projet/documentSEL.docx#_Toc159847466)

[1.1 Description du projet 1](#_Toc159847467)

[1.2 Agile 1](#_Toc159847468)

[1.3 Vues d’ensemble des fonctions pour l’application 2](#_Toc159847469)

[1.4 Descriptions des utilisateurs. 2](#_Toc159847470)

[1.5 Contrainte de l’application 3](#_Toc159847471)

[2.1 Définitions, acronymes et abréviations 4](#_Toc159847472)

[2.2 Use Case 5](#_Toc159847473)

## 1.1 Description du projet

L’application intitulée MedHub sera une application de gestion des soins de santé d’une manière efficace et rapide visant à faciliter la communication entre patients et médecins. L’application permettra aux patients de prendre des rendez-vous, de consulter leurs résultats médicaux et d’accéder aux ressources médicales telles que des conseils de santé ou des ressources éducatives en général. Le logiciel permettra aux médecins de voir les réservations faites par les patients et y modifier en cas d’indisponibilité, d’accéder aux dossiers médicaux des patients et de demander de nouvel approvisionnement tel que des médicaments ou des équipements. Cependant, MedHub ne pourra pas partager des informations médicales sans consentement et elle ne pourra pas effectuer des procédures médicales à distance.

## 1.2 Agile

La méthode agile sert à gérer notre projet de façon efficace et plus collaborative entre les membres de l’équipe et ainsi distribué les taches et de les gérer au fur et à mesure. Grâce à cette méthode, le projet peut être beaucoup mieux adapté selon les évènements ou problèmes. L’un des points forts de la méthode agile est ses livraisons fréquentes, qui permet de maintenir un avancement constant du projet et permet à l’équipe de voir visuellement le progrès accomplit par l’équipe grâce à l’application Trello.

## 1.3 Vues d’ensemble des fonctions pour l’application

MedHub offre de diverses fonctionnalités visant à simplifier et améliorer la gestion des soins de santé pour les patients et les médecins. Voici une vue d’ensemble des principales fonctionnalités du logiciel :

Côté patient :

* L’application permet de faire une réservation aux médecins selon les besoins et de l’annuler au besoin.
* Recevoir leurs résultats médicaux
* De rechercher des médecins
* Accéder à des ressources médicales éducatives.
* Inscription

Côté médecin :

* Modifier le rendez-vous ou l’annuler.
* Accéder au dossier médical de ses patients
* Envoyer les résultats médicaux aux patients
* Demande de matériel
* Inscription

Côté administrateur :

* Maintenance de l’application
* Gestion des utilisateurs (HCM)
* Confirmation/refus de l’inscription d’un médecin

## 1.4 Descriptions des utilisateurs.

**Patients**

Les patients auront accès au côté « client » de l’application. Ils sont les utilisateurs qui pourront prendre des rendez-vous et son considérés comme les « consommateurs » de l’application. Ils pourront aussi accéder aux soins médicaux et à des conseils laissés par les médecins dans leurs dossiers médicaux. L’application aura beaucoup de descriptions détaillées pour pouvoir guider les clients qui ne sont pas à l’aise avec ce type d’application.

**Médecins**

Les médecins auront accès au côté « Médecin » de l’application, ceux qui offrent leurs services aux patients. Les médecins ont plusieurs fonctionnalités dans l’application, leur permettant de gérer efficacement leur pratique médicale. Ils pourront modifier, supprimer et consulter des rendez-vous créés par les patients. Concernant les dossiers des patients, ils auront un accès complet aux dossiers médicaux de chaque patient. De plus, les médecins auront la possibilité de demander des soins médicaux spécifiques en besoin.

**Administrateur**

L’administrateur détient un accès complet à l’ensemble de l’application. Il aura accès à toutes les permissions pour gérer chaque aspect du système. En plus d’avoir toutes les fonctionnalités des médecins, il est aussi responsable de la gestion globale de l’application. Cela comprend la supervision des utilisateurs, la gestion des mises à jour de l’application et aussi la personne qui confirme ou refuse l’inscription d’un médecin.

## 1.5 Contrainte de l’application

Pour la conception du frontend, nous avons opté pour l’utilisation de la librairie React JS, reconnu pour sa flexibilité et son efficacité dans le développement des applications web.

En ce qui concerne la base de données, nous avons décidé d’utiliser MariaDB.

Pour le développement du backend, nous avons choisi de le faire en JavaScript, un langage de programmation. L’application sera hébergée sur AWS, une infrastructure infonuagique fiable et sécurisée.

Limites de l’application :

* Les maintenances et les mises à jour se passeront durant la nuit ou est-ce qu’il aura le moins de personnes connectées possible.
* Seulement les utilisateurs en provenance du Québec peuvent utiliser l’application.
* L’application ne permet pas de procédures médicales à distance.
* Il faut avoir un avoir minimalement 14 ans pour créer un compte.
* Elle est seulement conçue uniquement pour le navigateur web.
* L’application sera exclusivement en langue française et en anglais.

## 2.1 Définitions, acronymes et abréviations

Trello : Application qui permet de gérer les tâches et de les séparer dans une équipe.

Agile : Une méthode de gestion de projet caractérisé par sa collaboration et sa flexibilité.

HCM: Human capital management

Définition HCM: Fonction pour gérer les employés (docteur) qui est l’administrateur dans ce projet.

Frontend : la partie de l’application qui est l’interface utilisateur et qui interagit directement avec les utilisateurs.

Backend : la partie qui gère la logique en arrière de l’application et est invisible pour les utilisateurs consommateurs.

BD: base de données

Base de données : Un ensemble de données organisées permettant le stockage et la gestion des informations.

CSS: Feuille de style en cascade, Cascading Style Sheets en anglais

Définition CSS: Permets de styliser un site web HTML.

HTML: HyperText Markup Langage

Définition HTML: Un langage simple pour la représentation de page web.

Axios: Bibliothèque pour communiquer avec le serveur.

Bootstrap: Cadriciel pour développer du CSS simplement et rapidement.

Express: Crée un serveur web service.

Node/ npm: Gestionnaire de paquets pour un environnement JavaScript.

SQL: Structured Query Language

Définition: Langage pour la base de données

React js : une librairie JavaScript utilisée pour faciliter le développement des applications web.

Javascript : un langage de programmation orienté vers le frontend, utilisé pour créer des fonctionnalités interactives dans les navigateurs web.

MariaDB : Un type de base de données.

AWS: Amazon Web Services

Définition AWS: Une plateforme de services infonuagique d’Amazon qui offre divers services comme le stockage ou les hébergements des applications.

## 2.2 Use Case

|  |  |
| --- | --- |
| Id Use case | v.1 |
| Nom use case | Connection |
| Objectif | Authentification |
| Acteur principal | Patient, Docteur, Administrateur, Frontend |
| Acteur secondaire | Serveur, bd, backend |
| Pré condition | Qu’on rentre l’url sur un moteur de recherche. L’utilisateur sera sur la page de connexion. |
| Scénario | Nominal : La personne se connecte, rentre son login et son mot de passe, et accèdera à la page 2fa, il recevra par email son code 2fa  Exception : S'il n’y a pas connexion, internet, ou le compte n’est pas existant. L’user a perdu son mot de passe. |
| Post condition | L’utilisateur accédera à la page home de l’application |

|  |  |
| --- | --- |
| Id Use case | v.1 |
| Nom use case | Modification de mot de passe, s’il est oublié |
| Objectif | Authentification |
| Acteur principal | Patient, Docteur, Administrateur, Frontend |
| Acteur secondaire | Serveur, bd, backend. |
| Pré condition | Qu’on rentre l’url sur un moteur de recherche. L’utilisateur sera sur la page de connexion et clique sur l’option forget password/mot de passe oublié. |
| Scénario | Nominal : Redirection de page pour changer le mot de passe, l’user recevra un email contenant un code, ensuite le code sera validé pour qu’il puisse changer le mot de passe.  Exception : S'il n’y a pas connexion, internet, ou le compte n’est pas existant. L’user a perdu son mot de passe. |
| Post condition | L’utilisateur accédera à la page home de l’application |

|  |  |
| --- | --- |
| Id Use case | v.1 |
| Nom use case | Création de réservation |
| Objectif | Réservation |
| Acteur principal | Patient, Frontend docteur |
| Acteur secondaire | Serveur, bd, backend, |
| Pré condition | L’utilisateur est sur la page de home. Click soit sur le bouton réservation ou sur l’option réservation dans la barre de navigation |
| Scénario | Nominal : Le patient se trouve sur la page réservation, un formulaire.  Exception : S'il n’y a un problème de serveur ou de connexion ou qu’il n’y est pas de docteur de disponible. |
| Post condition | Le patient reçoit une confirmation par email et la page réservation affiche un message. |

|  |  |
| --- | --- |
| Id Use case | v.1 |
| Nom use case | Modification/Annulation de réservation |
| Objectif | Réservation |
| Acteur principal | Patient, Docteur, Frontend |
| Acteur secondaire | Serveur, bd, backend |
| Pré condition | Le docteur est sur la page de home. |
| Scénario | Nominal : Le docteur sélectionne la réservation de sa liste de la page home et soit l’annule ou la modifie (la date), le formulaire de modification collapse.  Exception : S'il n’y a pas connexion, internet. |
| Post condition | Le docteur restera à la page home de l’application. Le docteur et le patient recevra un email concernant le changement. |

|  |  |
| --- | --- |
| Id Use case | v.1 |
| Nom use case | Inscription |
| Objectif | S’inscrire dans l’application |
| Acteur principal | Patient, Docteur, Frontend |
| Acteur secondaire | Serveur, bd, backend |
| Pré condition | L’utilisateur est sur la page de connexion. |
| Scénario | Nominal : L’utilisateur sélectionne l’option sign-up/inscription et le redirige vers la page création de compte  Exception : S'il n’y a pas connexion, internet. Informations erronés |
| Post condition | L’utilisateur sera redirigé vers la page connexion |

|  |  |
| --- | --- |
| Id Use case | v.1 |
| Nom use case | Résultat médicaux |
| Objectif | Envoie de résultat médicale |
| Acteur principal | Patient, Docteur, Frontend |
| Acteur secondaire | Serveur, bd, backend |
| Pré condition | Le docteur est sa page accueil, il a un formulaire pour le résultat |
| Scénario | Nominal : Le patient voit sa liste de résultat.  Exception : S'il n’y a pas connexion, internet. |
| Post condition | La table résultat a une nouvelle ligne de données. |

|  |  |
| --- | --- |
| Id Use case | v.1 |
| Nom use case | Résultat médicaux |
| Objectif | Réception de résultat médicale |
| Acteur principal | Patient, Docteur, Frontend |
| Acteur secondaire | Serveur, bd, backend |
| Pré condition | Le patient est sur sa page d’accueil/home. |
| Scénario | Nominal : Le patient voit sa liste de résultat.  Exception : S'il n’y a pas connexion, internet. |
| Post condition | Le patient reste sur sa page. |

|  |  |
| --- | --- |
| Id Use case | v.1 |
| Nom use case | Demande de matériel |
| Objectif | Demande de matériel |
| Acteur principal | Docteur, Frontend |
| Acteur secondaire | Serveur, backend |
| Pré condition | Le docteur est sur sa page d’accueil/home. |
| Scénario | Nominal : Le docteur clique sur un bouton sur la barre de navigation,  Ça le redirige sur un formulaire et il fait sa demande. L’envoie se fait par l’email.  Exception : S'il n’y a pas connexion, internet. |
| Post condition | Le patient reste sur sa page. |