

**Projet Fil Rouge**

**Pour la première année**

**Youcode Safi**

**Application web pour l’informatisation du dossier médicale**

**Sujet**

**Soutenu par**

**Soumaya AMGHAR**

**Le 17 juin2022**

| **Encadrer par** | ECHCHADI Youness |
| --- | --- |
| **Co-Encadrer par** | ABID Abdeladim |

**Remerciement**

Je tiens tout d’abord à remercier tous les membres du jury qui ont accepté d’examiner ce travail et de l’enrichir par leurs propositions.

Je tiens à remercier sincèrement :

-Le **Pr ECHCHADI Youness** notre formateur, pour l’orientation, la confiance, la patience qui ont constitué un apport considérable sans lequel ce travail n’aurait pas pu être mené au bon port.

-Le **Pr ABID Abdeladim**, notre Co-formateur, pour son soutien, ses précieux conseils et son aide durant cette période de formation.

-Melle **ISLIS Kaoutar** pour son encadrement, ses conseils, ses explications et son aide dans la rédaction de ce rapport.

-Ma reconnaissance va également au staff **Mm OULD HNINI Siham et Mr BELKHADIR Adil**, pour leurs supports et tous les efforts qu’ils ont fait pour qu’ils nous offrent un environnement confortable pour étudier. Je suis très reconnaissante pour le temps qu’ils ont pris pour nous aider à réussir notre formation. Merci pour votre confiance et votre soutien inconditionnel.

Je ne peux terminer ces remerciements sans citer évidemment toute l’équipe de Youcode Safi, c’était plus qu’une équipe c’était une vraie famille. Mes sincères remerciements à tous les professeurs qui m’ont enseigné et qui par leurs compétences m’ont soutenue dans la poursuite de mes études. Enfin, je remercie tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail.

**Résumé**

Le dossier patient est un élément central dans la démarche de soins puisqu’il regroupe l’ensemble des informations concernant la personne soignée. Son informatisation est sans doute un impératif de la qualité des soins puisqu’il constitue un outil au service de la traçabilité des actes médicaux, la sécurité des soins, la continuité des soins, l’aspect médico-économique, la coordination et la communication entre les professionnels et aussi l’évaluation des pratiques professionnelles**.**

Dans ce projet nous avons créé une plateforme de gestion de l’historique médical des patients. Cette plateforme qui s’appelle “DMI” est désignée au médecins et praticiens de la santé au Maroc

**Sommaire**

[**Liste des figures 1**](#_heading=h.30j0zll)

[**Liste des abréviations 2**](#_heading=h.3znysh7)

[**Introduction 3**](#_heading=h.2et92p0)

[**Matériel et méthodes**](#_heading=h.1t3h5sf) **6**

[**Résultats et discussion**](#_heading=h.2s8eyo1) **13**

[**Conclusion**](#_heading=h.26in1rg) **22**

**Liste des figures**

Figure 1:Diagramme de cas d’utilisation……………………………………………………..14

Figure 2: Diagramme de classe ………………………………………………………........... 15

Figure 3: Diagramme de séquence de la partie Admin……………………………………….15

Figure 4:Diagramme de séquence de la partie utilisateur ……………………………………16

Figure 5: Diagramme d’activité………………………………………………………………18

Figure 6: Maquette de la partie Admin……………………………………………………….19

Figure 7: Maquette du “login” et "register" de la partie utilisateur…………………………..20

Figure 8: Maquette du “dashboard” de l’utilisateur…………………………………………..21

Figure 9:Les tableaux de la base de données de notre système………………………………21

**Liste des abréviations**

DMI: dossier médical informatisé.

UML: langage de modélisation unifié.

MVC: model-view-controller.

HTML: HyperText Markup Language.

CSS: Cascading style sheets.

PHP: HyperText Preprocessor.

SGBD: Système de gestion de base de données.

**Introduction**

## I. Introduction générale :

Le dossier médical se place au cœur du système de santé. Il regroupe l’ensemble de documents qui retrace des épisodes ayant affecté la santé de cette personne (lettre, notes, compte rendu, résultats de laboratoire, film radiologique, ...).

Par ailleurs, la bonne tenue du dossier médical est un préalable indispensable pour disposer de données nécessaires à l’évaluation qualitative des soins dont l’amélioration doit être un objectif prioritaire. La qualité des soins est appréciée à base de certaines caractéristiques à savoir l’acceptabilité, l’accessibilité, le caractère approprié, la continuité, la délivrance au bon moment, l’efficacité, l’efficience et la sécurité. Particulièrement, la tenue du dossier du patient influence la continuité des soins c’est à dire la coordination entre les différents professionnels intervenant auprès du patient suppose le partage des informations dans un support accessible , la sécurité c'est à-dire la présence d’informations spécifiques facilement identifiables et accessibles permet la prévention des risques pour le patient , et l’efficacité des soins c'est-à-dire la prise de décision médicale est facilitée par la structuration du dossier et les informations qui y sont consignées. Le dossier médical du patient est un outil très important pour l’organisation, la planification et la traçabilité de soins. Sa tenue est considérée comme un critère majeur de qualité. Sa qualité et son utilisation doivent être régulièrement évaluées pour être améliorées et conduire ainsi à une meilleure prise en charge du patient et à l’optimisation du fonctionnement de l’établissement. Il constitue la porte d’entrée au système de soins et qui est fondé sur une communication appropriée. De plus, Il abrite toute l’anamnèse du patient. Ses informations administratives, médicales, et paramédicales sont conservés dans un dossier unique appelé dossier médical et qui sont régit par le secret professionnel. Il est un outil de communication, de coordination et d’information entre les acteurs de soins et avec les patients. Il permet de suivre et de comprendre le parcours hospitalier du patient. Il est un élément primordial de la qualité des soins en permettant leur continuité dans le cadre d’une prise en charge pluriprofessionnelle et pluridisciplinaire.

D’autant plus qu’il est élémentaire, le dossier médical du patient devient encore plus fécond quand on l’associe à l'évolution des nouvelles technologies d'Information et de communication. En effet, il devient possible, et surtout nécessaire, de pouvoir favoriser des échanges d'informations plus rapides et systématiques entre tous les acteurs professionnels de santé, de mutualiser les connaissances et les informations propres à chaque patient, et donc à chaque parcours de soins. Cette démarche doit être synonyme de meilleure efficacité, de rapidité, d'augmentation du service rendu, d'adaptation à la demande croissante d'excellence, de diminution du risque, lié à l’iatrogénie.la base de toutes cette démarche revient à l’informatisation du dossier médical du patient c’est-à dire à la naissance du dossier médical informatisé (DMI).

Ce travail met le point sur l’informatisation du dossier médical au Maroc. La problématique s’articule autour : de la création d’une plateforme désignée aux médecins et tout praticien de santé.

**Matériel et méthodes**

1. **Conception :**

Pour la conception de notre système nous avons adopté une méthode orientée objet. En effet, cette dernière est une approche incontournable dans le cadre du développement des applications.   
Pour mieux présenter l’architecture de notre système, on va choisir le langage de modélisation le plus adopté UML :  
C'est un langage de modélisation, défini comme une norme de modélisation objet qui sert à décrire et à documenter un système d'information.  
En utilisant ce langage, les objectifs de la modélisation objet suivant sont assurés :

* Formaliser la conception d'applications.
* 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.

Dans ce projet nous avons utilisé quatre types de diagrammes pour modéliser notre système :

1. **Les diagrammes de cas d’utilisation**, qui décrivent la façon dont le système sera utilisé. Ils montrent les relations entre les acteurs (ou agents externes au système à concevoir) et les cas d’utilisation du système.
2. **Les diagrammes de séquence**, qui montrent l’interaction entre plusieurs objets organisés dans le temps. Cette interaction est un ensemble de messages échangés entre les objets pour effectuer une opération ou obtenir un résultat.
3. **Les diagrammes de classes**, qui expriment de manière générale la structure statique d’un système, en termes de classes et de relations entre ces classes.
4. **Les diagrammes d’activités**, qui sont des variantes des diagrammes d’états, organisés par rapport aux actions (ou opérations). Ils sont utilisés pour représenter le comportement interne d’une méthode ou d’un cas d’utilisation.
5. **Maquette :**

Le maquettage est un processus de concrétisation graphique de l’interfaced’une interface digitale (site web, logiciel, application…). La maquette permet de visualiser les premières versions du design graphique. Dans ce projet nous avons adopté Figma comme outil de maquettage.

**1.Figma**

Figma est une plateforme collaborative pour éditer des graphiques vectoriels et faire du prototypage. Elle permet de concevoir des designs systèmes pour faciliter la création de sites web et d’applications mobiles. C’est une solution à destination des UI et UX designers et des développeurs. L’interface propose de nombreuses fonctionnalités :

* **De design :** avec des outils de conception pour le web, des fonctions de mise en page automatique, des plugins pour réduire les tâches répétitives.
* **De prototypage :** pour tester les concepts très tôt en cours de design.
* **De design system :** pour concevoir des designs cohérents avec des bibliothèques mises à jour en permanence.
* **Collaboratives :** pour travailler à plusieurs et en même temps sur un projet, revenir sur une version antérieure si nécessaire ou encore afficher le travail d’un seul collaborateur par exemple.

**III. Implémentation :**

Pour le développement du code source de ce projet nous avons utilisé le pattern modèle-vue-contrôleur (en abrégé MVC, de l'anglais model-view-controller), qui est un modèle destiné à répondre aux besoins des applications interactives en séparant les problématiques liées aux différents composants au sein de leur architecture respective.  
Ses avantages :

* Séparation des compétences (design, base de données, application)
* Simplicité de mise à jour
* Vitesse de création de pages.

Ce paradigme regroupe les fonctions nécessaires en trois catégories :

* Un modèle (Modèle de données) ;
* Une vue (présentation, interface utilisateur) ;
* Un contrôleur (logique de contrôle, gestion des événements, synchronisation).

Nous expliquons ces trois parties l'une après l'autre :

* **Le modèle (ou Model)** :

Le modèle représente les structures de données. Typiquement, les classes modèles contiennent des fonctions qui aident à récupérer, à insérer et à mettre à jour des informations de la base de données.  
Par exemple, lorsque nous disons « le contrôleur récupère les données d’un outil», il va en fait, faire appel au modèle Outil (« Tool »). C'est le modèle qui peut récupérer les données de cet outil, généralement via une requête au serveur SQL. Au final, il permet au contrôleur de manipuler les outils mais sans savoir comment ils sont stockés, gérés, etc. C'est une couche d'abstraction.

* **La vue (ou View) :**

La vue correspond à l'interface avec laquelle l'utilisateur interagit. Elle se présente sous la forme d'une Template représentant l'interface.  
Reprenons l'exemple de l’outil. Ce n'est pas le contrôleur qui affiche le formulaire, il ne fait qu'appeler la bonne vue. Si nous avons une vue formulaire, les balises HTML du formulaire d'édition de l’outil y seront et finalement le contrôleur ne fera qu'afficher cette vue sans savoir vraiment ce qu'il y a dedans.  
Donc en pratique, c'est le « designer » d'un projet qui travaille sur les vues. La séparation de vues et contrôleurs permet aux designers et aux développeurs PHP de travailler ensemble sans besoin de contact direct.

* **Le contrôleur (ou Controller)** :

Il gère l'interface entre le modèle et le client. Il va interpréter la requête de ce dernier pour lui envoyer la vue correspondante. Il effectue la synchronisation entre le modèle et les vues.  
Il est la couche qui se charge d'analyser et de traiter la requête de l'utilisateur. Le contrôleur contient la logique de notre application et va se contenter « d'utiliser » les autres composants : les modèles et les vues. Concrètement, un contrôleur va récupérer, par exemple, les informations sur l'utilisateur courant, vérifier qu'il a le droit de modifier un tel outil, récupérer les données de cet outil et demander la page du formulaire d'édition de l’outil.

1. **Visual Studio Code :**

Pour l’écriture du code nous avons recouru à l’outil Visual Studio Code qui est un éditeur de code source qui peut être utilisé avec une variété de langages de programmation. Il a été annoncé le 29 avril 2015 par Microsoft lors de la conférence Build 2015. Une version préliminaire a été publiée peu de temps après.

1. **HTML5 :**

Dans l’implémentation du code nous avons utilisé le langage HTML signifie « ***HyperText Markup Language*** » qu’on peut traduire par « langage de balises pour l’hypertexte ». Il est utilisé afin de créer et de représenter le contenu d’une page web et sa structure.

1. **Tailwind :**

Nous avons aussi utilisé le Tailwind qui est un framework de CSS. Il est utilisé afin de styler le site web, il permet de contrôler de manière précise comment un site web est présenté.

1. **JavaScript :**

Le **JavaScript** a aussi été utilisé. Il est un langage de programmation de scripts principalement employé dans les pages web interactives. Avec les langages HTML et CSS, JavaScript est au cœur des langages utilisés par les développeurs web.

1. **PHP :**

Nous avons utilisé le langage PHPsignifie «***Hypertext Preprocessor »*** qui est un langage de programmation libre principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale.

1. **XAMPP :**

Pour une base de données installée sur le Local nous avons utilisé Xampp et phpMyAdmin.

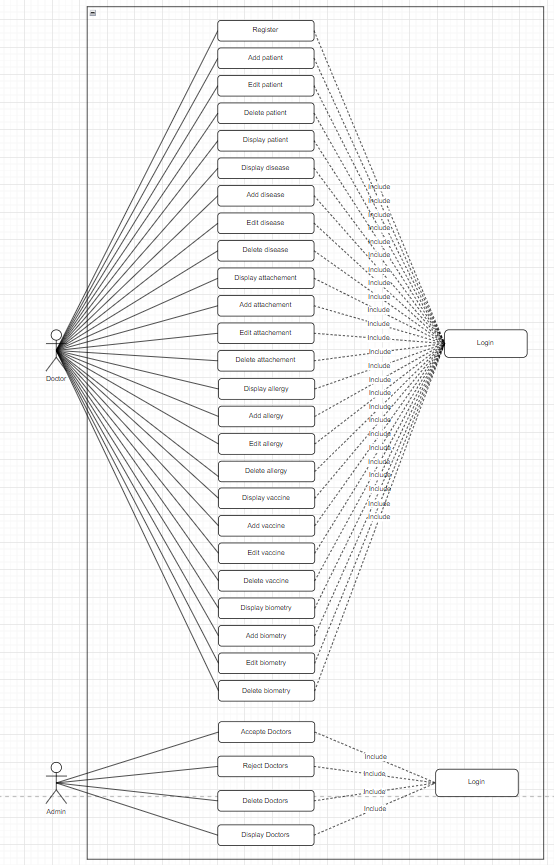
XAMPP est une distribution Apache entièrement gratuite et facile à installer contenant MySQL, PHP et Perl. Le paquetage open source XAMPP a été mis au point pour être incroyablement facile à installer et à utiliser.

1. **PhpMyAdmin :**

PhpMyAdmin est un outil logiciel gratuit écrit en PHP, destiné à gérer l'administration de MySQL sur le Web. PhpMyAdmin prend en charge un large éventail d'opérations sur MySQL et Maria DB. Les opérations fréquemment utilisées (gestion des bases de données, des tables, des colonnes, des relations, des index, des utilisateurs, des autorisations, etc.) peuvent être effectuées via l'interface utilisateur, tout en vous permettant d'exécuter directement n'importe quelle instruction SQL.

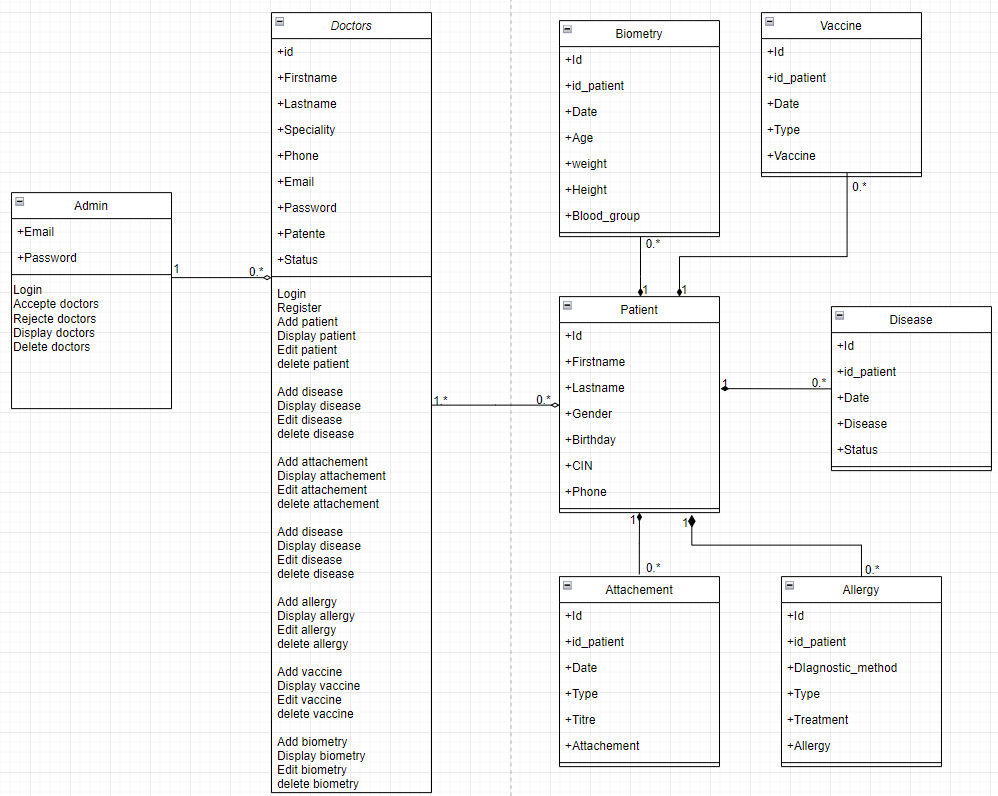
**Résultats et Discussion**

1. **Conception :**
2. **Diagramme de cas d’utilisation :**

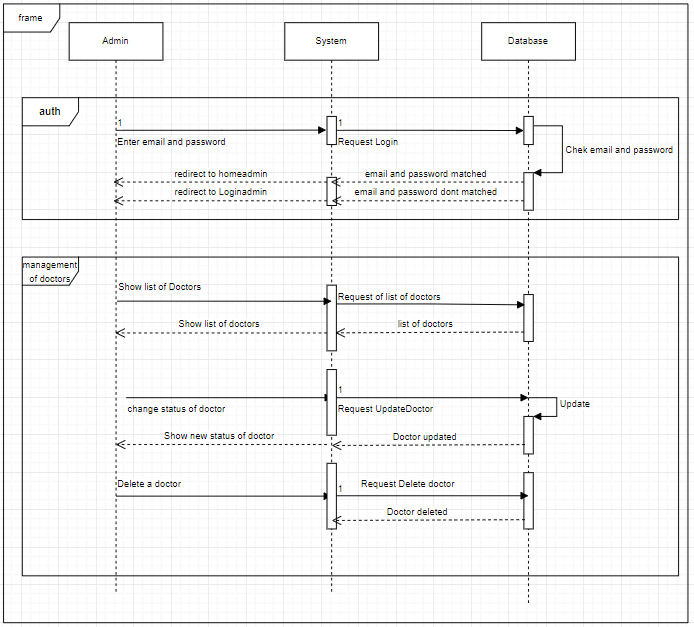
****

**Figure 1 :** Diagramme de cas d’utilisation

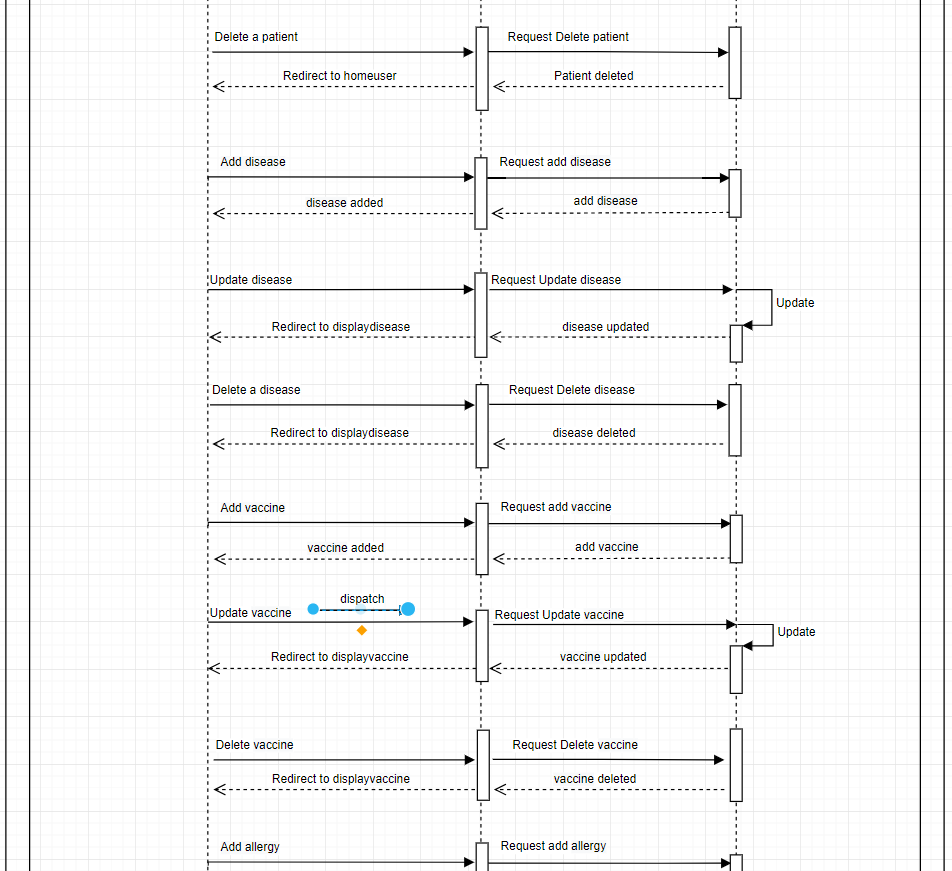
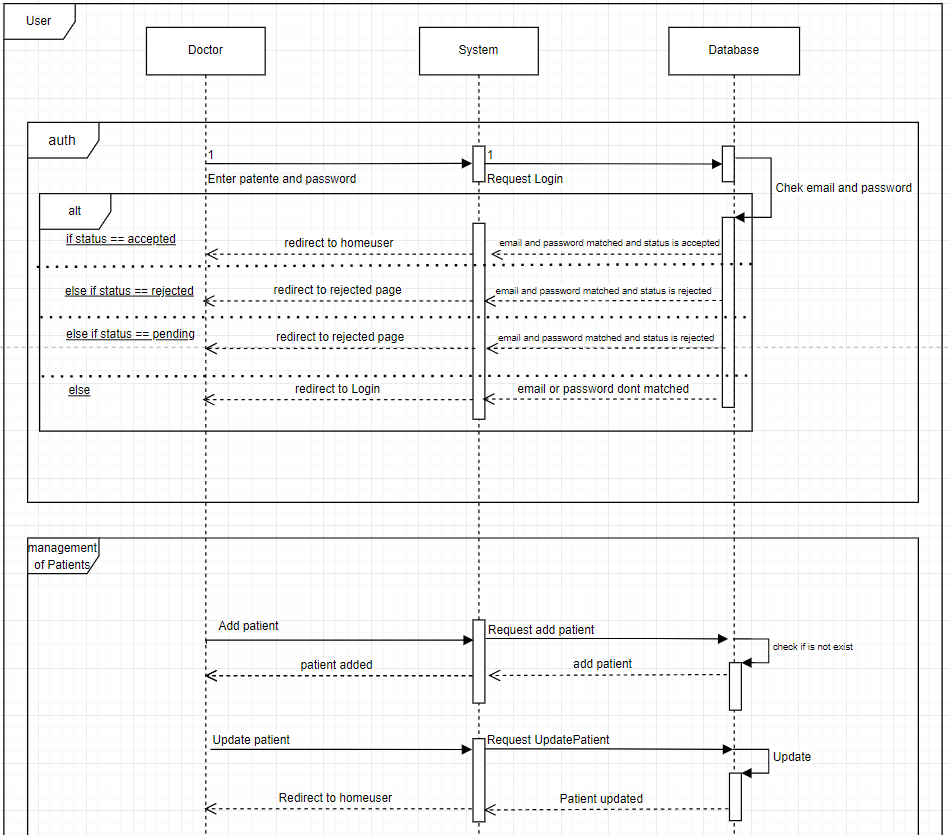
1. **Diagramme de classe :**

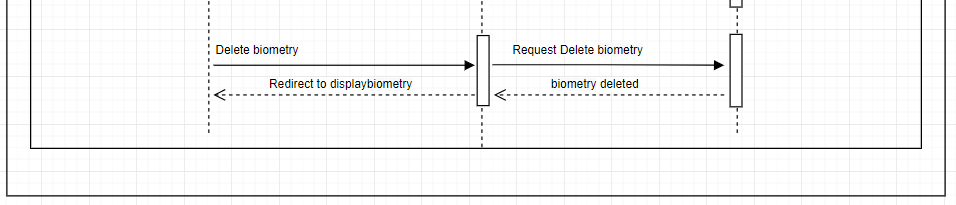
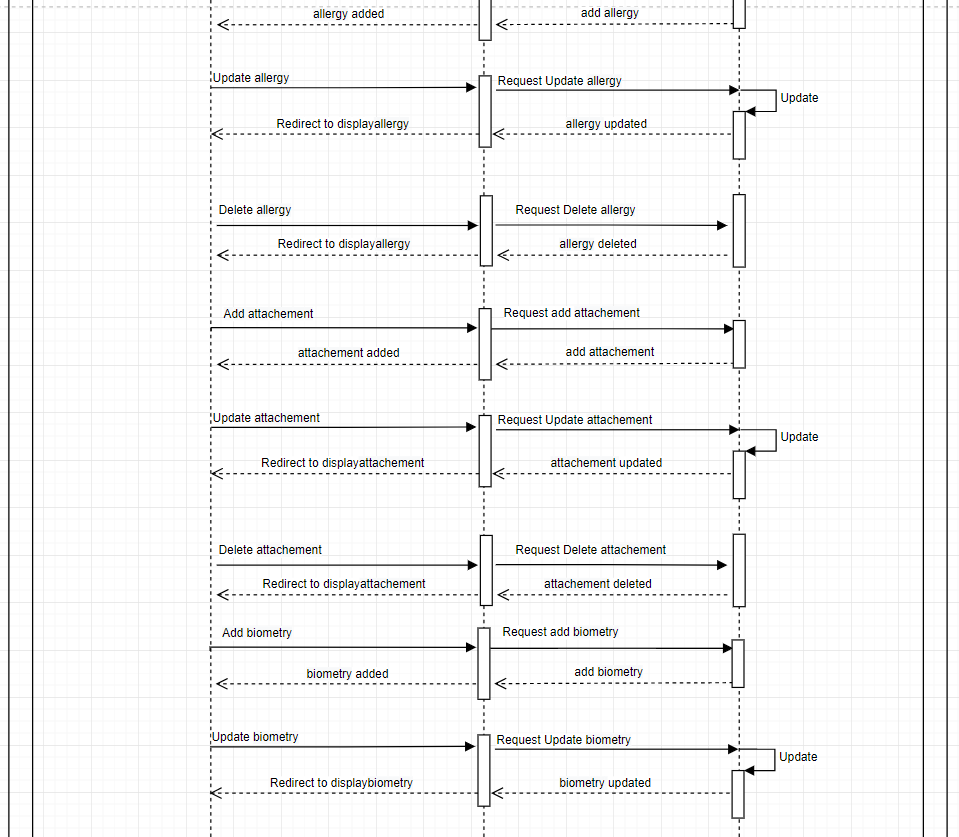


**Figure 2 :** Diagramme de classe

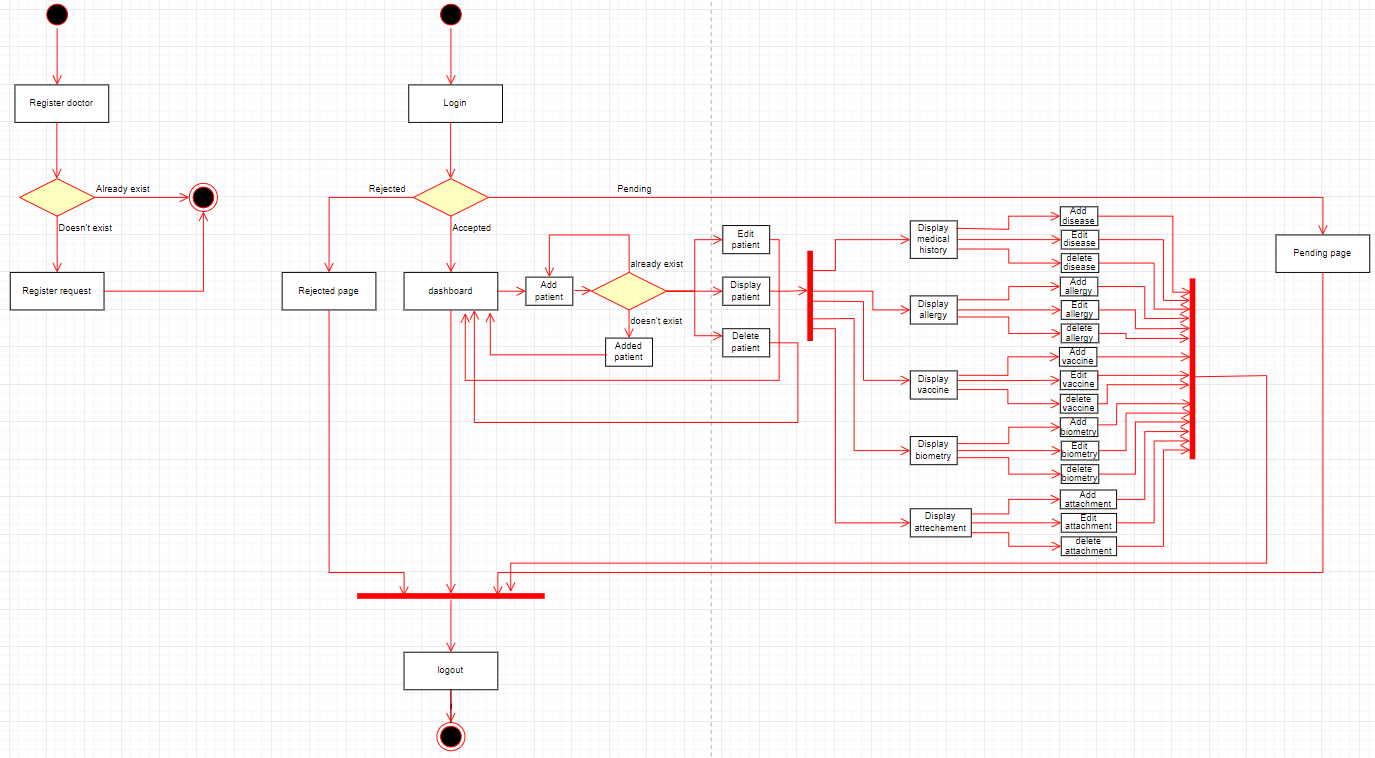
1. **Diagramme de séquence**

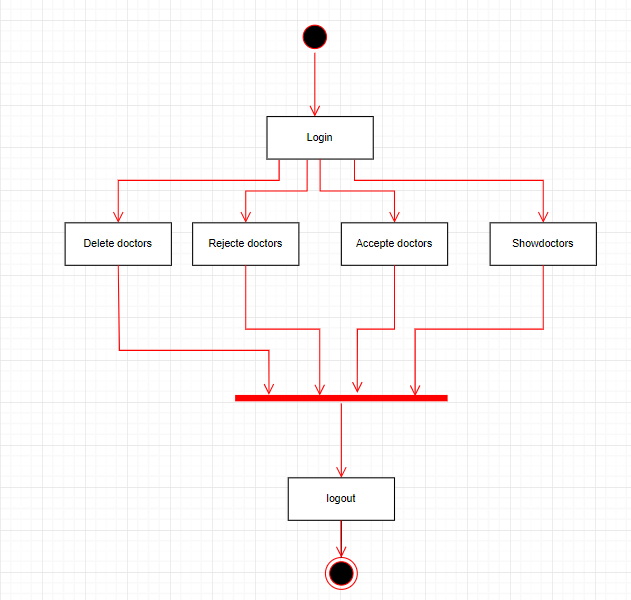
**Figure 3 :** Diagramme de séquence de la partie Admin



**Figure 4 :** Diagramme de séquence de la partie utilisateur

1. **Diagramme d’activité**

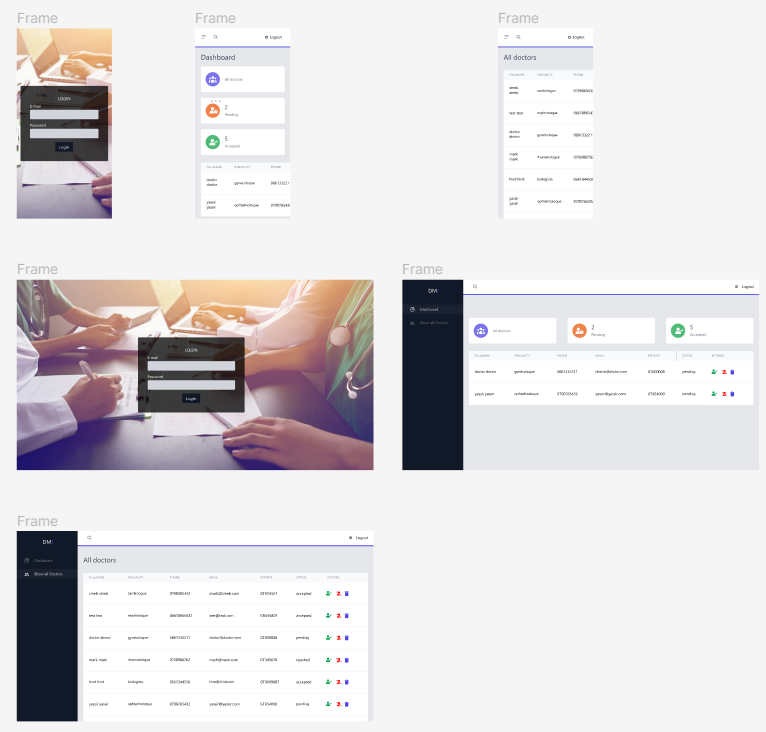
****



**Figure 5 :** Diagramme d’activité

1. **Maquette :**

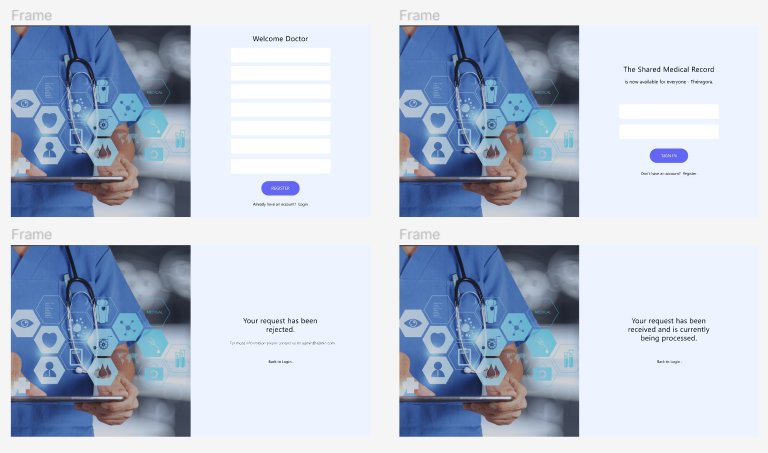
Pour la maquette nous avons utilisé l’outil Figma pour construire le design du site web suite à deux dimensions 1500px pour le web et 400px pour le mobile comme le montre le figure 6.

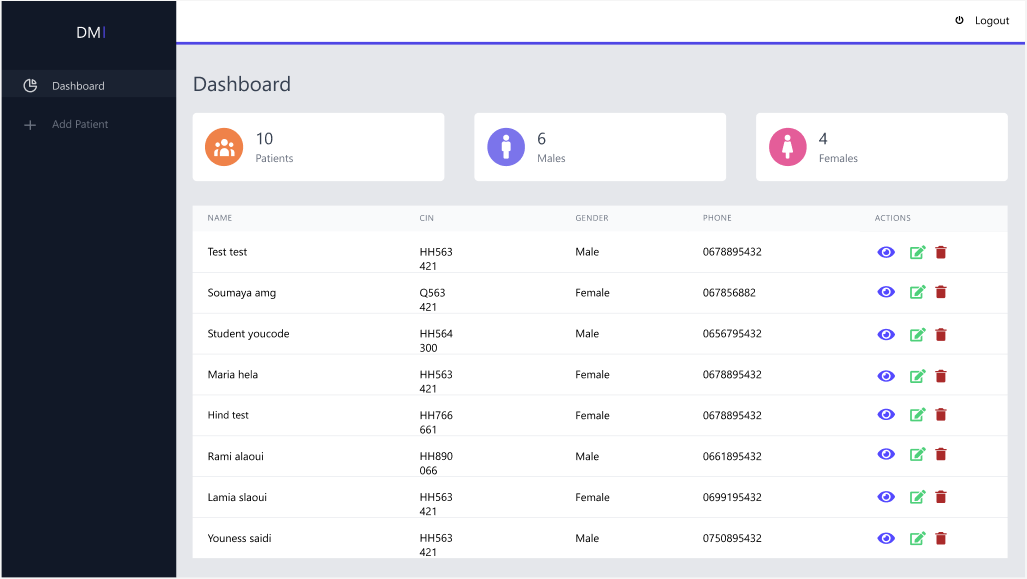
**Figure 6 :** Capture d'écran de la maquette de la partie Admin

Pour l’utilisateur il a le droit de s’inscrire puis pour se connecter, mais il doit tout d’abord être accepté par l’Admin. L’admin vérifie la patente du médecin si elle est correcte, il l'accepte sinon il la rejette (figure 7).

Si l’Admin a accepté l’inscription du médecin, il peut alors accéder au Dashboard comme le montre le figure 8.

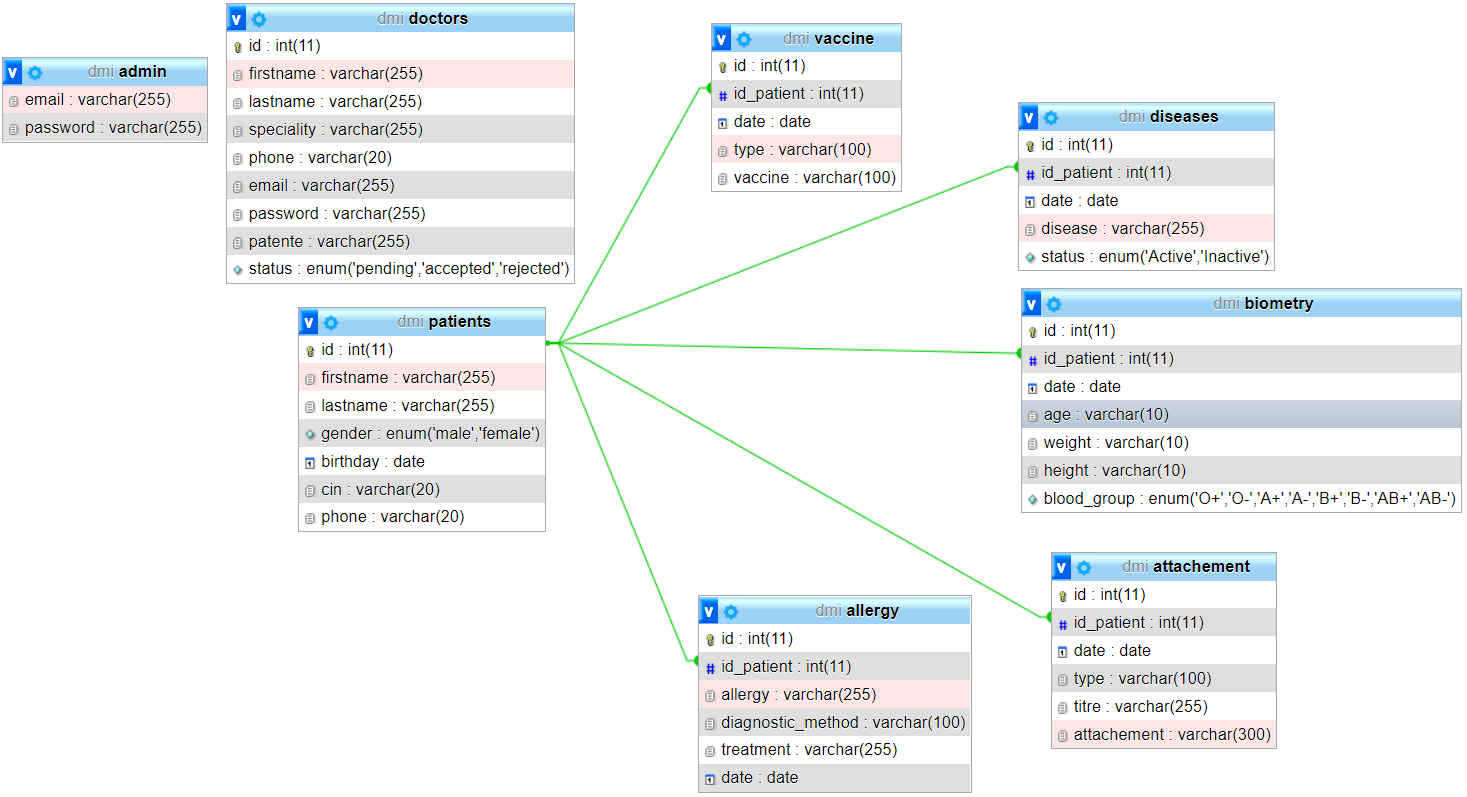
Après avoir accepté, l’utilisateur peut ajouter, modifier ou supprimer des patients. Il peut aussi afficher d’autres informations en cliquant sur le bouton yeux. Il peut voir l’historique médical des patients, leurs allergies, leurs biométries, leurs vaccinations ainsi que leurs pièces attachées (analyses médicales, radio, ordonnance…).

**Figure 7 :** Maquette du “login” et “register” de la partie utilisateur

**Figure 8 :** Maquette du dashboard de la partie utilisateur

1. **Implémentation :**

**Schéma de la base de données**  
Nous avons utilisé le système de gestion de base de données relationnel MySQL pour enregistrer les tables de notre système.  
 Nous avons choisi ce SGBD vu qu’il est très léger et simple à utiliser et configurer, de plus il est open source et tous les hébergeurs web le fournissent avec leurs packs d’hébergement.   
La figure 9 montre les relations entre les différentes tables de notre système.



**Figure 9 :** Les tableaux de la base de données de notre système.

**Conclusion**

Notre étude visait à créer une plateforme de l’informatisation du dossier médical. Qui vise l’amélioration de la qualité des soins dans l’établissement hospitalier. Cette plateforme constitue un plan idéal pour construire un réseau important entre les médecins pour partager les dossiers des patients.

Il ne reste que la généralisation de cette application au niveau national avec surtout une sensibilisation pour les utilisateurs. Pour la simple raison qui est l’importance et la sensibilité des données qu’elle contient.