

**RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE**

**MINISTÈRE DE L’ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE**

**UNIVERSITÉ DE LAGHOUAT**

FACULTÉ DES SCIENCES ET DE LA TECHNOLOGIE

DÉPARTEMENT D’INFORMATIQUE

**Mémoire de Fin d’Études**

En vue de l’obtention du diplôme de licence en Informatique

Spécialité : systeme d information

**Titre du Projet :**

**Application Web de Prise de Rendez-vous Médicaux**

**Réalisé par :**

ouarnoughi Mohamed Aissa

Beddine Mohamed

**Encadré par :**

Mr. Benhoucine Nasreddine



**Dédicace**

Je dédie ce travail à :

- Mon encadrant universitaire, pour ses conseils précieux et son accompagnement tout au long de ce projet.

- Mes parents, pour leur amour, leur soutien moral et leur patience durant mes études.

- Mes amis et collègues, pour leur aide, leurs encouragements et leur motivation pendant le développement de ce projet.

- Toutes les personnes qui ont contribué, de près ou de loin, à la réussite de ce travail.

**Le resume**

Ce projet présente le développement d'une application web destinée à la prise de rendez-vous médicaux. Son objectif principal est de réduire les longues files d'attente dans les hôpitaux, et de faciliter l'accès aux soins pour les habitants des zones rurales. L'application permet aux patients de consulter les disponibilités des médecins et de réserver un créneau adapté. Elle offre également aux médecins un système de gestion des rendez-vous pratique et efficace.

Abstract

This project introduces the development of a web application for booking medical appointments. Its main objective is to reduce long queues in hospitals and improve healthcare access for people living in rural areas. The application allows patients to view doctors’ availability and book convenient time slots. It also provides doctors with an efficient appointment management system.

[الملخص ]

يقدم هذا المشروع تطوير تطبيق ويب لحجز المواعيد الطبية، ويهدف بشكل رئيسي إلى تقليل الطوابير الطويلة في المستشفيات وتسهيل وصول سكان المناطق الريفية إلى الرعاية الصحية. يسمح التطبيق للمرضى بمشاهدة مواعيد الأطباء المتاحة وحجز الوقت المناسب، كما يوفر للأطباء نظامًا فعالًا لإدارة المواعيد.

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations

HTML: HyperText Markup Language

CSS : Cascading Style Sheets

JS : JavaScript

BD : Base de Données

API : Application Programming Interface

UI : User Interface

UX : User Experience

Dans un contexte où les files d’attente dans les établissements de santé deviennent de plus en plus longues, et où les habitants des zones rurales rencontrent des difficultés pour accéder à des soins médicaux en temps voulu, ce projet vise à proposer une solution numérique efficace : une application web de prise de rendez-vous médicaux.

Ce système permet aux patients de planifier leurs consultations à distance, en toute simplicité, via une interface intuitive et accessible. Il vise à améliorer la gestion des flux de patients, à optimiser le temps des médecins, et à garantir une meilleure organisation des rendez-vous.

Ce rapport présente les différentes étapes de conception et de développement de cette application, ainsi que les choix technologiques adoptés et les résultats obtenus.

1. Analyse du système
   1. Description du problème

Le système de santé dans plusieurs régions, notamment les zones rurales, souffre d’une mauvaise gestion des rendez-vous médicaux. Les patients doivent souvent se déplacer pour prendre un rendez-vous, ce qui peut être contraignant et inefficace. Cela engendre des files d’attente interminables, des retards, voire des pertes de rendez-vous. De plus, les ressources médicales ne sont pas utilisées de manière optimale, ce qui affecte la qualité des soins.

Face à ce constat, il devient essentiel de mettre en place une solution qui facilite la prise de rendez-vous à distance et qui améliore la coordination entre patients et professionnels de santé.

* 1. Objectifs du projet

- Réduire les files d'attente dans les centres de santé.

- Faciliter la prise de rendez-vous pour les patients, notamment ceux vivant dans les zones rurales.

- Offrir aux médecins un outil de gestion des rendez-vous simple et efficace.

- Améliorer l’organisation globale des consultations médicales.

- Réduire les rendez-vous manqués (no-shows) grâce à des rappels automatiques.

* 1. Public cible

- Les patients : toute personne souhaitant prendre rendez-vous chez un médecin de manière rapide et pratique.

- Les médecins : professionnels de santé cherchant à mieux gérer leur planning de consultations.

- Les secrétaires médicaux : personnel administratif en charge de l’organisation des rendez-vous.

* 1. Cahier des charges fonctionnel

- Inscription et authentification des utilisateurs (patients et médecins).

- Consultation de la liste des médecins disponibles.

- Réservation d’un créneau de rendez-vous selon la disponibilité du médecin.

- Annulation ou modification de rendez-vous.

- Espace personnel pour le patient (historique des rendez-vous, rappels...).

- Espace personnel pour le médecin (planning, notifications...).

* 1. Contraintes et exigences non fonctionnelles

- Interface utilisateur intuitive et responsive.

- Sécurité des données personnelles (chiffrement, authentification).

- Disponibilité et performance du système (temps de réponse rapide).

- Compatibilité avec les navigateurs courants (Chrome, Firefox, etc.).

- Système multilingue (optionnel).

1. Étude des solutions existantes

### ****6.1 Présentation de quelques solutions disponibles****

Avant de développer notre propre application, nous avons étudié plusieurs plateformes existantes qui proposent des services similaires. Cette analyse comparative permet de mieux comprendre les fonctionnalités standards, les forces et les limites de chaque solution, afin d’identifier les points à améliorer.

Parmi les solutions analysées :

- \*\*Doctolib (France)\*\*

Plateforme très populaire en Europe qui permet aux patients de prendre rendez-vous avec des médecins en ligne. Elle offre des rappels automatiques, la téléconsultation, et un espace personnel pour les médecins.

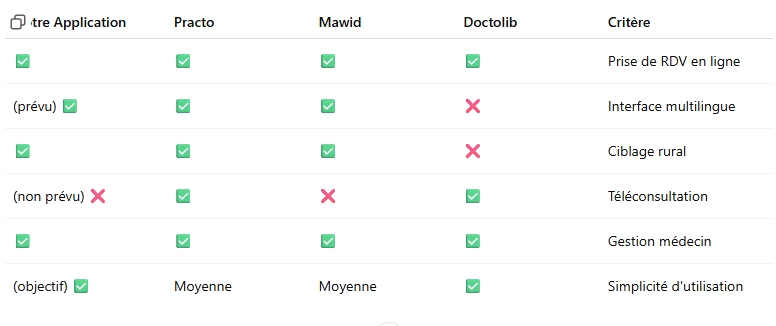
- \*\*Mawid (Arabie Saoudite)\*\*

Application officielle permettant la prise de rendez-vous dans les établissements publics de santé. Elle est intégrée dans le système de santé national, mais reste limitée à certains types de services.

- \*\*Practo (Inde)\*\*

Application mobile et web qui permet non seulement de réserver des rendez-vous, mais aussi de gérer les dossiers médicaux, d’accéder à la pharmacie en ligne, etc.

6.2 Analyse comparative



6.3 Conclusion

Aucune des solutions existantes n’est totalement adaptée au contexte spécifique de notre projet, notamment pour les zones rurales dans notre région. Certaines plateformes sont trop complexes ou nécessitent des équipements spécifiques. Notre application se distingue par sa simplicité, sa légèreté, et son accessibilité pensée pour tous, même avec une faible connexion internet.

## ****7. Conception du système****

### ****7.1 Diagramme de classes****

Le diagramme de classes suivant modélise les principales entités de notre application de prise de rendez-vous médicaux en ligne :

### ****7.2 Description du diagramme****

Ce diagramme met en évidence trois classes principales :

#### 🔹 ****Patient****

Représente un utilisateur du système souhaitant réserver un rendez-vous.

id : identifiant unique du patient.

name : nom complet du patient.

email : adresse électronique pour la communication.

phone : numéro de téléphone pour contact rapide.

#### 🔹 ****Doctor****

Représente un professionnel de santé inscrit sur la plateforme.

id : identifiant unique du médecin.

name : nom complet (souvent précédé de "Dr.").

email : adresse email professionnelle.

specialty : spécialité médicale (ex. : Cardiologie)

#### 🔹 ****Appointment****

Modélise le rendez-vous médical entre un patient et un médecin.

id : identifiant du rendez-vous

date : date prévue du rendez-vous.

time : heure du rendez-vous.

status : état actuel du rendez-vous (Pending, Confirmed, Cancelled, etc.).

patient\_id : référence au patient concerné.

doctor\_id : référence au médecin concerné.

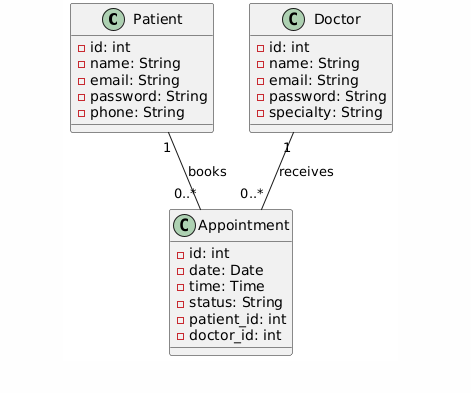
### ****7.3 Interprétation****

Ce diagramme illustre une relation de **"many-to-one"** entre les rendez-vous et les utilisateurs :

Un patient peut avoir plusieurs rendez-vous.

Un médecin peut gérer plusieurs rendez-vous.

Chaque rendez-vous est associé à un seul patient et un seul médecin.

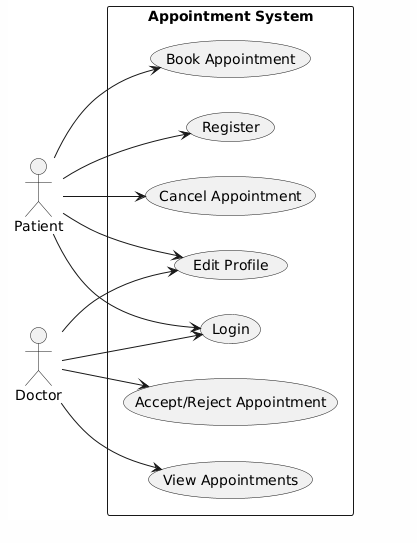


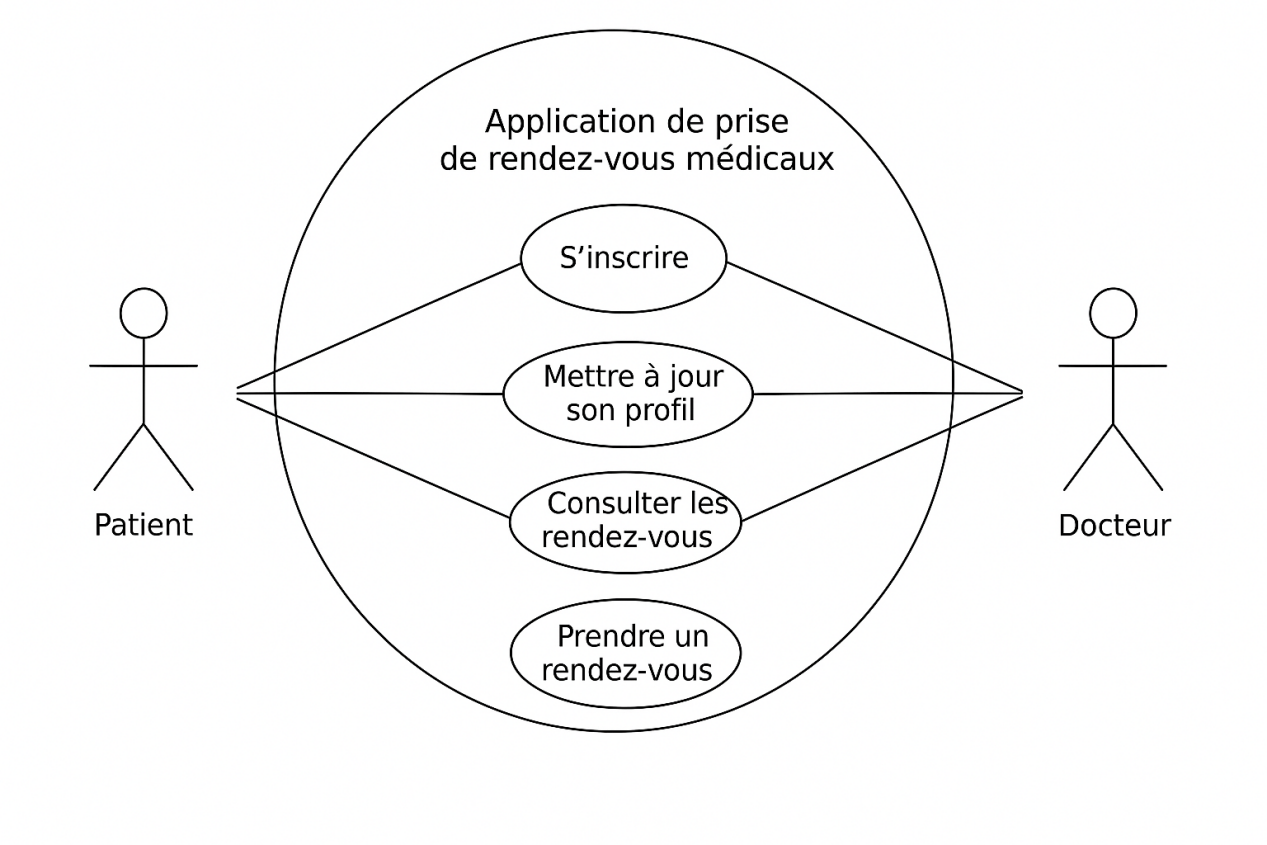
### ✅ ****Le diagramme de cas d'utilisation****

### ✅ ****Le diagramme d'objets (Object diagram)****

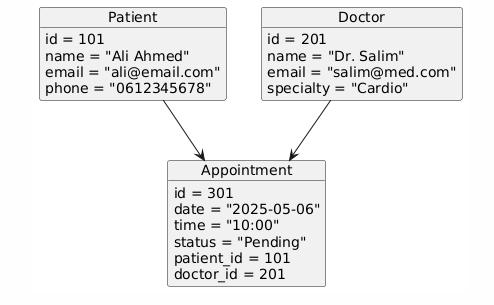
### ****Description des diagrammes UML****

Le **diagramme de cas d'utilisation** représente les différentes interactions entre les utilisateurs (acteurs) et le système. Il permet d’identifier les fonctionnalités principales offertes par l’application, telles que l’inscription, la prise de rendez-vous, et la consultation du planning. Ce diagramme facilite la compréhension des besoins fonctionnels du système.





Le **diagramme d’objets**, quant à lui, illustre une instance concrète du diagramme de classes à un moment donné. Il montre les objets en mémoire, leurs attributs spécifiques, et les liens qui existent entre eux pendant l’exécution du système. Il est utile pour visualiser l’état du système à un instant précis.



**Flux de communication :**

L’utilisateur (patient ou médecin) interagit avec l’interface utilisateur frontend, ce qui déclenche une requête HTTP vers le backend.

L’application PHP reçoit la requête, traite les données et effectue les opérations nécessaires (par exemple, interrogation de la base de données, mise à jour des enregistrements).

L’application PHP génère ensuite une réponse, qui est envoyée au frontend.

Le frontend reçoit la réponse, traite les données et met à jour l’interface utilisateur en conséquence.

**Considérations de sécurité :**

**Authentification et autorisation :** Le système met en œuvre une authentification des utilisateurs (connexion/déconnexion) et un contrôle d’accès basé sur les rôles afin de garantir que seuls les utilisateurs autorisés peuvent accéder et effectuer des actions spécifiques.

**Validation et assainissement des entrées :** Le code backend PHP valide et assainit toutes les entrées utilisateurs pour prévenir les vulnérabilités courantes des applications web, telles que l’injection SQL et les attaques de type cross-site scripting (XSS).

**Chiffrement des données :** Les données sensibles, telles que les mots de passe des utilisateurs, sont stockées dans la base de données MySQL en utilisant des techniques sécurisées de hachage et de chiffrement.

**Sécurité réseau :** La communication entre le frontend et le backend est sécurisée via HTTPS afin de protéger contre l’écoute clandestine et les attaques de type man-in-the-middle.

Cette architecture système fournit une base évolutive et sécurisée pour le système de rendez-vous médicaux, tirant parti des forces de PHP et MySQL pour offrir une application de gestion de santé robuste et conviviale.

### sequence-addapoArchitecture technique

#### 1. Architecture logicielle

**Client-Serveur**:

**Frontend (Client) :**

Technologies : HTML, CSS, JavaScript

Interface utilisateur pour patients et médecins

**Backend (Serveur) :**

Langage : PHP

Traitement logique, gestion des requêtes, sécurisation

API REST pour communication frontend-backend

**Base de données :**

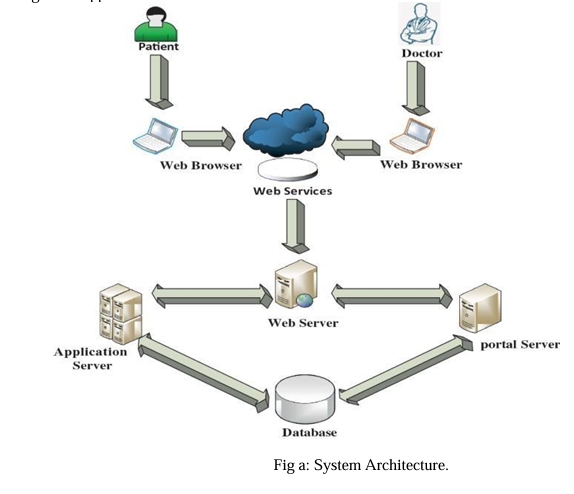
MySQL

Stockage des utilisateurs, rendez-vous, données médicales

#### 2. Architecture base de données

**Exemple simplifié des tables principales :**

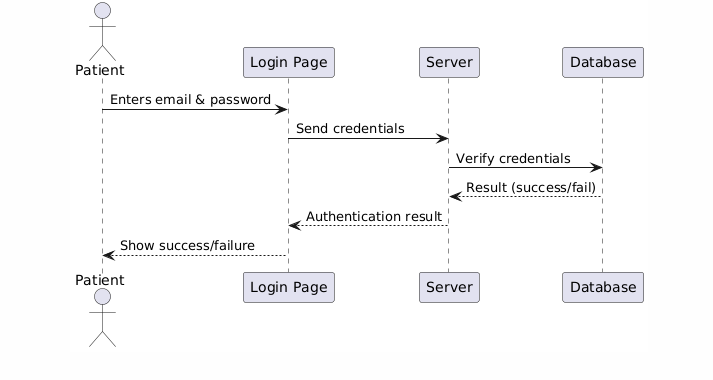
| **Table** | **Colonnes principales** |
| --- | --- |
| utilisateurs | id, nom, email, mot\_de\_passe (haché), rôle |
| patients | id\_utilisateur (clé étrangère), nss, adresse |
| médecins | id\_utilisateur (clé étrangère), spécialité |
| rendez\_vous | id, id\_patient, id\_medecin, date\_heure, statut |



Relations :

utilisateurs 1 → 1 patients ou médecins

rendez\_vous lié à patients et médecins par des clés étrangères



## 8 Développement du système

Le développement du système constitue une étape cruciale dans la réalisation de notre application de prise de rendez-vous médical. Ce système se compose de trois parties principales : le frontend, le backend, et la base de données.

### 1. Frontend

L’interface utilisateur a été conçue en utilisant HTML, CSS et JavaScript afin d’offrir une expérience utilisateur intuitive et responsive. Elle permet aux patients et aux médecins de naviguer facilement dans l’application, de s’inscrire, de se connecter, et de gérer les rendez-vous.

### 2. Backend

Le backend est développé en PHP, qui traite toutes les requêtes envoyées par le frontend. Il est responsable de la logique métier, de la gestion des sessions, et des interactions avec la base de données MySQL. Le backend assure également la validation et la sécurisation des données reçues.

### 3. Base de données

La base de données MySQL stocke toutes les informations essentielles, telles que les profils utilisateurs, les rendez-vous, et les horaires des médecins. Elle est conçue avec des relations claires pour garantir l’intégrité et la cohérence des données.

### Fonctionnalités principales

**Inscription et authentification :** Les utilisateurs peuvent créer un compte sécurisé et se connecter via un système d’authentification basé sur des rôles.

**Gestion des rendez-vous :** Les patients peuvent prendre, modifier ou annuler leurs rendez-vous. Les médecins peuvent consulter et gérer leurs plannings.

**Sécurité :** Toutes les entrées utilisateur sont validées et protégées contre les attaques courantes comme l’injection SQL et le XSS.

### Exemple de processus : prise de rendez-vous

Lorsqu’un patient choisit une date et un médecin, le frontend envoie une requête au backend. Celui-ci vérifie la disponibilité dans la base de données, puis enregistre le rendez-vous si possible. Enfin, une confirmation est renvoyée au patient.

## ****Technologies et outils utilisés****

Pour le développement de notre application, nous avons utilisé un ensemble de technologies et d’outils adaptés au développement web moderne. Voici une brève description de chacun :

### ****HTML (HyperText Markup Language)****

HTML est le langage de base utilisé pour structurer les pages web. Il permet de définir les titres, paragraphes, boutons, formulaires, et autres éléments visibles de l’interface utilisateur.

### ****CSS (Cascading Style Sheets)****

CSS est utilisé pour styliser les pages HTML. Il permet de définir les couleurs, les polices, les espacements, les alignements, et de rendre l’application plus esthétique et ergonomique.

### ****JavaScript (JS)****

JavaScript est un langage de programmation côté client qui permet d’ajouter de l’interactivité à l’interface utilisateur, comme la validation des formulaires, les alertes dynamiques ou la mise à jour du contenu sans recharger la page.

### ****PHP (Hypertext Preprocessor)****

PHP est un langage de script côté serveur utilisé pour gérer la logique métier de l’application. Il reçoit les données du frontend, traite les requêtes, interagit avec la base de données, et renvoie les résultats au client.

### ****MySQL****

MySQL est un système de gestion de base de données relationnelle. Il permet de stocker et d’organiser toutes les données nécessaires, telles que les informations des utilisateurs, les rendez-vous, et les horaires.

### ****Bootstrap****

Bootstrap est un framework CSS open-source qui facilite la création d’interfaces réactives et modernes. Il propose des composants prédéfinis comme les boutons, les menus, les formulaires, et s’adapte automatiquement aux différents écrans (ordinateurs, tablettes, smartphones).

### ****Visual Studio Code (VS Code)****

VS Code est un éditeur de code léger et puissant, utilisé pour écrire et organiser le code source du projet. Il propose des extensions utiles pour PHP, HTML, CSS et JavaScript, ainsi que des outils de débogage.

### ****XAMPP****

XAMPP est un environnement de développement local qui intègre Apache, MySQL, PHP, et d’autres outils. Il permet de tester l’application localement avant son déploiement sur un serveur réel.

### ****6. Résultats et discussion****

Le système de prise de rendez-vous médical a été développé et déployé avec succès, en s'appuyant sur le langage de programmation PHP et le système de gestion de base de données MySQL. Les premiers tests ainsi que les retours des utilisateurs ont été positifs, ce qui indique que le système répond aux objectifs fixés et offre une expérience de soins simplifiée tant pour les patients que pour les médecins.

## ****Conclusion****

Le système de prise de rendez-vous médical, développé en utilisant PHP et MySQL, a permis de répondre efficacement au problème des processus de réservation de rendez-vous inefficaces et contraignants. En offrant une plateforme conviviale, sécurisée et performante, le système a amélioré l'expérience globale des soins de santé pour les patients comme pour les médecins.

Grâce à une architecture évolutive et à des fonctionnalités robustes, cette solution représente une avancée prometteuse dans la gestion des rendez-vous médicaux.

Avec l’évolution continue du système, les développements futurs pourraient inclure l'intégration de fonctionnalités supplémentaires telles que le paiement en ligne, la téléconsultation, ainsi que l’analyse avancée des données, dans le but d’enrichir davantage les performances du système et sa valeur pour la communauté médicale.

## ****Défis rencontrés et perspectives futures****

Lors du développement du système de prise de rendez-vous médical, plusieurs défis ont été rencontrés, notamment la gestion de la sécurité des données personnelles, l’authentification des utilisateurs, et la synchronisation en temps réel des disponibilités des médecins. De plus, l’optimisation de l’interface pour une utilisation fluide sur différents appareils (ordinateurs, smartphones, tablettes) a nécessité des ajustements constants.

Pour l’avenir, plusieurs pistes d’amélioration sont envisagées :

**Intégration des paiements en ligne**, permettant aux patients de régler leurs consultations à distance.

**Ajout d’un module de téléconsultation**, facilitant les consultations à distance via visioconférence.

**Mise en œuvre de notifications intelligentes**, par email ou SMS pour rappeler les rendez-vous.

**Utilisation de l’intelligence artificielle**, pour proposer automatiquement des créneaux en fonction des habitudes de réservation.

**Déploiement dans un environnement cloud**, afin d’assurer une meilleure scalabilité et accessibilité.

Ces évolutions permettront de renforcer l’efficacité du système et de mieux répondre aux besoins croissants du secteur de la santé numérique.

## ****Références et bibliographie****

W3Schools. PHP Tutorial. Disponible sur : https://www.w3schools.com/php/

1. 1. https://www.php.net/docs.php

Mozilla Developer Network (MDN). HTML, CSS, JavaScript documentation. Disponible sur : [https://developer.mozilla.org/](https://developer.mozilla.org/" \t "_new)

MySQL Documentation. MySQL 8.0 Reference Manual. Disponible sur : [https://dev.mysql.com/doc/](https://dev.mysql.com/doc/" \t "_new)

1. https://dev.mysql.com/doc/

Bootstrap Official Site. Introduction to Bootstrap. Disponible sur : [https://getbootstrap.com/](https://getbootstrap.com/" \t "_new)

1. https://getbootstrap.com/docs/
2. XAMPP Apache Friends. Serveur local de développement. Disponible sur : [https://www.apachefriends.org/](https://www.apachefriends.org/" \t "_new)
3. https://www.w3schools.com/php/
4. https://www.php-fig.org/security/

Visual Studio Code. Documentation officielle. Disponible sur : [https://code.visualstudio.com/docs](https://code.visualstudio.com/docs" \t "_new)

1. https://www.sitepoint.com/scaling-php-applications/
2. https://www.tutorialspoint.com/mysql/index.html/

OpenClassrooms. Créer un site web dynamique avec PHP et MySQL. Disponible sur : https://openclassrooms.com/fr/

Stack Overflow. Discussions techniques sur les erreurs et les solutions PHP/MySQL. Disponible sur : [https://stackoverflow.com/](https://stackoverflow.com/" \t "_new)

GeeksforGeeks. How to Build a Doctor Appointment System using PHP and MySQL. Disponible sur : [https://www.geeksforgeeks.org/](https://www.geeksforgeeks.org/" \t "_new)

ResearchGate. Online Appointment Systems in Healthcare: A Literature Review. Disponible sur : [https://www.researchgate.net/](https://www.researchgate.net/" \t "_new)

IEEE Xplore. Web-based Medical Appointment Systems: Design and Security Challenges.

GitHub. Doctor Appointment System Projects (Exemples de projets open-source). Exemples consultés :

[https://github.com/SanVik2000/Doctor-Appointment-System](https://github.com/SanVik2000/Doctor-Appointment-System" \t "_new)

[https://github.com/Ritik2703/Online-Doctor-Appointment](https://github.com/Ritik2703/Online-Doctor-Appointment" \t "_new)

Elsevier Journal. E-health and Appointment Systems in Developing Countries.