



REPUBLIQUE DU CAMEROUN  
Paix – Travail – Patrie  
**UNIVERSITE DE DOUALA**  
**ECOLE NATIONALE SUPERIEURE  
POLYTECHNIQUE DE DOUALA**  
B.P. 2701 Douala  
Tél. (237) 697 542 240  
Site web : [www.enspd-udo.cm](http://www.enspd-udo.cm)

REPUBLIC OF CAMEROON  
Peace – Work – Fatherland  
**THE UNIVERSITY OF DOUALA**  
**NATIONAL HIGHER POLYTECHNIC  
SCHOOL OF DOUALA**  
P.O.Box:2701 Douala  
Phone : (237) 697 542 240  
Email: [contact@enspd-udo.cm](mailto:contact@enspd-udo.cm)



## RAPPORT DE PROJET

**THEME : CONCEPTION ET REALISATION D'UN  
SYSTEME INTELLIGENT DE GESTION DES RENDEZ-  
VOUS MEDICAUX : MEDISMART**

N°	NOMS	MATRICULES
1	NGACHILI NJANKOUO FRED	24G01119
2	LEUKAM TCHEUMALEU MAXIME	24G01101
3	DJOGU KAMENI DARELLE	22G00078
4	FADIMATOU ABDOU	24G01090
5	MBAIAMMADJI SYLVESTRE BANYO	24G01339

**FILIERE : GIT / GLO**

**SOUS LA SUPERVISION DE :**

**NIVEAU : 4**

**Dr. IHONOCK**

**ANNEE SCOLAIRE: 2025-2026**

# SOMMAIRE

<b>SOMMAIRE .....</b>	<b>1</b>
<b>LISTE DE FIGURES.....</b>	<b>2</b>
<b>LISTE DES TABLEAUX .....</b>	<b>3</b>
<b>INTRODUCTION GENERALE .....</b>	<b>4</b>
<b>CHAPITRE I : PRESENTATION DU PROJET.....</b>	<b>5</b>
<b>CHAPITRE II : ETUDE DE L’EXISTANT ET ANALYSE DU PROJET.....</b>	<b>7</b>
<b>CHAPITRE III : CAHIER DE CHARGES ET CONCEPTION .....</b>	<b>11</b>
<b>SECTION 1 : CAHIER DE CHARGES .....</b>	<b>11</b>
<b>SECTION 2 : CONCEPTION DE LA SOLUTION .....</b>	<b>26</b>
<b>CHAPITRE IV : IMPLÉMENTATION, INTERPRÉTATION ET DISCUSSION DES RÉSULTATS .....</b>	<b>34</b>
<b>SECTION 1 : ENVIRONNEMENT DE TRAVAIL .....</b>	<b>34</b>
<b>SECTION 2 : TESTS, INTERPRETATION ET DISCUSHION DES RESULTATS .....</b>	<b>37</b>
<b>CONCLUSION GENERALE .....</b>	<b>44</b>
<b>TABLE DES MATIERES .....</b>	<b>45</b>

## LISTE DE FIGURES

Figure 1 : Matrice FAST .....	9
Figure 2 : Matrice SWOT.....	10
Figure 3 : Diagramme de Gantt.....	24
Figure 4 : Niveaux d'abstraction de Merise.....	26
Figure 5 : Diagramme de cas d'utilisation .....	31
Figure 6 : Diagramme de classes.....	32
Figure 7 : Logo React.js .....	35
Figure 8 : Logo Laravel.....	35
Figure 9 : Logo MySQL.....	36
Figure 10 : Page d'accueil Medismart .....	39
Figure 11 : Recommandations IA .....	39
Figure 12 : Formulaire de prise de rendez-vous.....	40
Figure 13 : Champs du formulaire de prise de rendez-vous .....	40
Figure 14 : Interface de définition du créneau horaire du médecin.....	41
Figure 15 : Dashboard médecin .....	41
Figure 16 : Paramètres patients .....	42
Figure 17 : Modification du statut de consultation .....	42
Figure 18 : Téléversement des dossiers patients .....	43

## LISTE DES TABLEAUX

Tableau 1 : Risques techniques .....	18
Tableau 2 : Risques projet.....	19
Tableau 3 : Risques métier .....	20
Tableau 4 : Les coûts humains .....	20
Tableau 5 : Les coûts matériels .....	21
Tableau 6 : Les coûts logiciels .....	22
Tableau 7 : Les coûts d'infrastructure .....	22
Tableau 8 : Les coûts de maintenance.....	23
Tableau 9 : Bilan global .....	24
Tableau 10 : Cas d'utilisation .....	30
Tableau 11 : Test d'authentification .....	37
Tableau 12 : Test de prise de rendez-vous .....	38
Tableau 13 : Test de définition du créneau horaire du médecin .....	38

## INTRODUCTION GENERALE

La digitalisation du secteur de la santé constitue aujourd'hui un levier essentiel pour améliorer l'organisation et la qualité des services médicaux. La gestion des rendez-vous médicaux, encore souvent réalisée de manière manuelle ou peu optimisée, entraîne des problèmes tels que les conflits de planning, les longues attentes et les absences non signalées, affectant aussi bien les patients que les professionnels de santé.

Face à ces limites, l'adoption de systèmes intelligents apparaît comme une solution efficace pour automatiser et optimiser la gestion des rendez-vous. Ces systèmes permettent de centraliser les informations, d'optimiser les plannings, de gérer les priorités médicales et de réduire les inefficacités organisationnelles.

Ce projet porte sur la conception d'un système intelligent de gestion des rendez-vous médicaux visant à faciliter la prise de rendez-vous pour les patients et à améliorer la gestion des consultations pour les professionnels de santé. Le système intègre des fonctionnalités telles que l'ordonnancement intelligent, les notifications automatiques et la sécurisation des données médicales. Ce rapport présente l'analyse des besoins, la conception du système, l'estimation des coûts et les perspectives d'évolution du projet.

# CHAPITRE I : PRESENTATION DU PROJET

Cette partie présente le contexte, la problématique, les objectifs et les cibles du projet de conception d'un système intelligents de gestion de rendez-vous médicaux.

## I.1. DESCRIPTION DU PROJET

Ce projet consiste à concevoir et à développer un système intelligent de gestion des rendez-vous médicaux destiné à moderniser et optimiser l'organisation des consultations au sein des structures de santé. Il permet aux patients de prendre, modifier ou annuler leurs rendez-vous en ligne de manière simple et rapide. Les professionnels de santé disposent quant à eux d'un outil centralisé pour gérer efficacement leurs plannings et les dossiers patients. Le système intègre un moteur d'ordonnancement capable de proposer des créneaux optimisés en fonction des disponibilités, du type de consultation et des priorités médicales. Des notifications automatiques sont mises en place afin de réduire le taux d'absentéisme et d'améliorer la ponctualité. Une attention particulière est accordée à la sécurité et à la confidentialité des données médicales. L'architecture du système est modulaire et évolutive, permettant l'intégration future de fonctionnalités telles que la téléconsultation et l'analyse statistique. Ce projet s'inscrit dans une démarche de modernisation et de digitalisation du secteur de la santé.

## I.2. CONTEXTE DU PROJET

Le secteur médical fait face à des défis majeurs en matière de gestion des rendez-vous : taux d'absentéisme élevé, temps d'attente prolongés, difficultés de coordination entre praticiens, et surcharge administrative. MEDISMART vise à digitaliser et optimiser l'ensemble du processus de prise de rendez-vous médical.

## I.3. OBJECTIFS DU PROJET

L'objectif général de ce projet est de concevoir et développer un système intelligent de gestion des rendez-vous médicaux permettant d'optimiser la planification des consultations, de réduire le temps d'attente et l'absentéisme des patients, tout en assurant la sécurité et la confidentialité des données médicales. Les objectifs spécifiques quant à eux, incluent de:

- Mettre en place une plateforme sécurisée de prise, modification et annulation des rendez-vous médicaux.
- Optimiser les plannings des professionnels de santé à l'aide d'un mécanisme d'ordonnancement intelligent.
- Réduire l'absentéisme grâce à l'envoi de notifications et de rappels automatiques.

- Centraliser les informations patients pour faciliter la gestion et le suivi des dossiers médicaux.
- Améliorer la communication entre patients, professionnels de santé et personnel administratif.
- Garantir la protection et la confidentialité des données médicales.
- Concevoir une architecture évolutive permettant l'intégration de fonctionnalités futures.

#### **I.4. CIBLES DU PROJET**

Le système est destiné aux acteurs suivants :

- **Patients** : personnes souhaitant planifier facilement leurs consultations médicales et recevoir des rappels automatiques.
- **Professionnels de santé (médecins, spécialistes, infirmiers)** : utilisateurs principaux pour la gestion des rendez-vous, du planning et des dossiers patients.
- **Personnel administratif (réceptionnistes, secrétaires médicales)** : gestion et supervision des rendez-vous et des flux de patients.
- **Structures de santé (cliniques, hôpitaux, cabinets médicaux, centres de santé)** : optimisation des ressources, amélioration de la qualité des services et de la satisfaction des patients.
- **Responsable et décideurs** : accès aux statistiques et indicateurs pour une meilleure prise de décision.

## CHAPITRE II : ETUDE DE L'EXISTANT ET ANALYSE DU PROJET

Ce chapitre présente une analyse de l'existant afin d'identifier les pratiques actuelles, leurs limites et les opportunités d'amélioration, en vue de la conception d'un nouveau système plus performant.

### II.1. PRESENTATION DES SYSTEMES EXISTANTS

A l'issue de l'étude que nous avons eu à mener, il en ressort qu'il existe plusieurs solutions numériques tant web que mobiles dont le fonctionnement est essentiellement basé sur la gestion intelligente des rendez-vous médicaux :

- **clikOdoc** : c'est une solution de prise de rendez-vous médicaux en ligne permettant aux patients de réserver des consultations et aux professionnels de santé de gérer leurs agendas.
- **Doctolib** : c'est une plateforme de gestion des rendez-vous médicaux qui met en relation patients et praticiens à travers des interfaces web et mobiles.
- **Zocdoc** : c'est une application internationale offrant un service de recherche de professionnels de santé et de prise de rendez-vous médicaux en ligne.

### II.2. ANALYSE DES FORCES ET FAIBLESSES DE L'EXISTANT

#### II.2.1. Forces des systèmes actuels

- Facilitation de la prise de rendez-vous en ligne pour les patients.
- Réduction des files d'attente et de la charge administrative.
- Envoi de rappels automatiques limitant les rendez-vous manqués.
- Disponibilité d'outils de téléconsultation pour certains systèmes.
- Amélioration de la visibilité des praticiens et de l'accès aux soins.
- Interfaces généralement simples et accessibles au grand public.

#### II.2.2. Faiblesses des systèmes actuels

- **Coût élevé** pour les professionnels de santé, rendant l'accès difficile pour les petites structures.
- **Manque de personnalisation** selon les réalités locales (contextes africains ou pays en développement, par exemple).
- **Dépendance forte** à une connexion Internet stable.



- **Intelligence limitée** dans l'optimisation automatique des plannings (peu de priorisation médicale réelle).
- **Faible intégration** avec certains systèmes internes des hôpitaux ou dossiers médicaux existants.

### II.3. ANALYSE FAST ( FUNCTION ANALYSIS SYSTEM TECHNIQUE)

L'analyse FAST est une méthode utilisée en gestion de projets pour clarifier les besoins et les exigences d'un projet. Elle se concentre sur la définition des fonctions et des caractéristiques d'un système, en impliquant les parties prenantes dans un processus collaboratif. Celle-ci passe par :

- **La fonction principale**
  - Gérer intelligemment les rendez-vous médicaux
- **L'analyse fonctionnelle**
  - Permettre la prise de rendez-vous en ligne.
  - Modifier et annuler les rendez-vous.
  - Optimiser le planning des praticiens.
  - Gérer les priorités médicales.
  - Envoyer des notifications automatiques.
  - Gérer les dossiers patients.
  - Sécuriser les données médicales.
- **Les détails des activités**
  - Afficher les créneaux disponibles en temps réel pour chaque praticien
  - Choisir le type de consultation
  - Mise à jour automatique du planning du praticien après modification ou annulation
  - Informer le patient et le praticien via notification
  - Eviter les conflits et les chevauchements de créneaux
  - Regrouper les rendez-vous similaires pour une meilleure gestion du temps
  - Identifier les rendez-vous urgents ou critiques
  - Ajuster l'ordre des rendez-vous
  - Envoyer des rappels par SMS, email ou notifications mobiles
  - Informer sur les changements de planning ou les annulations
  - Enregistrer les informations essentielles
  - Permettre un accès rapide aux données

- Protéger les informations par authentification et contrôle d'accès
- Assurer la confidentialité et l'intégrité des données stockées et échangées
- Proposer les meilleurs créneaux pour chaque patient
- Fournir des suggestions pour réduire les temps d'attente

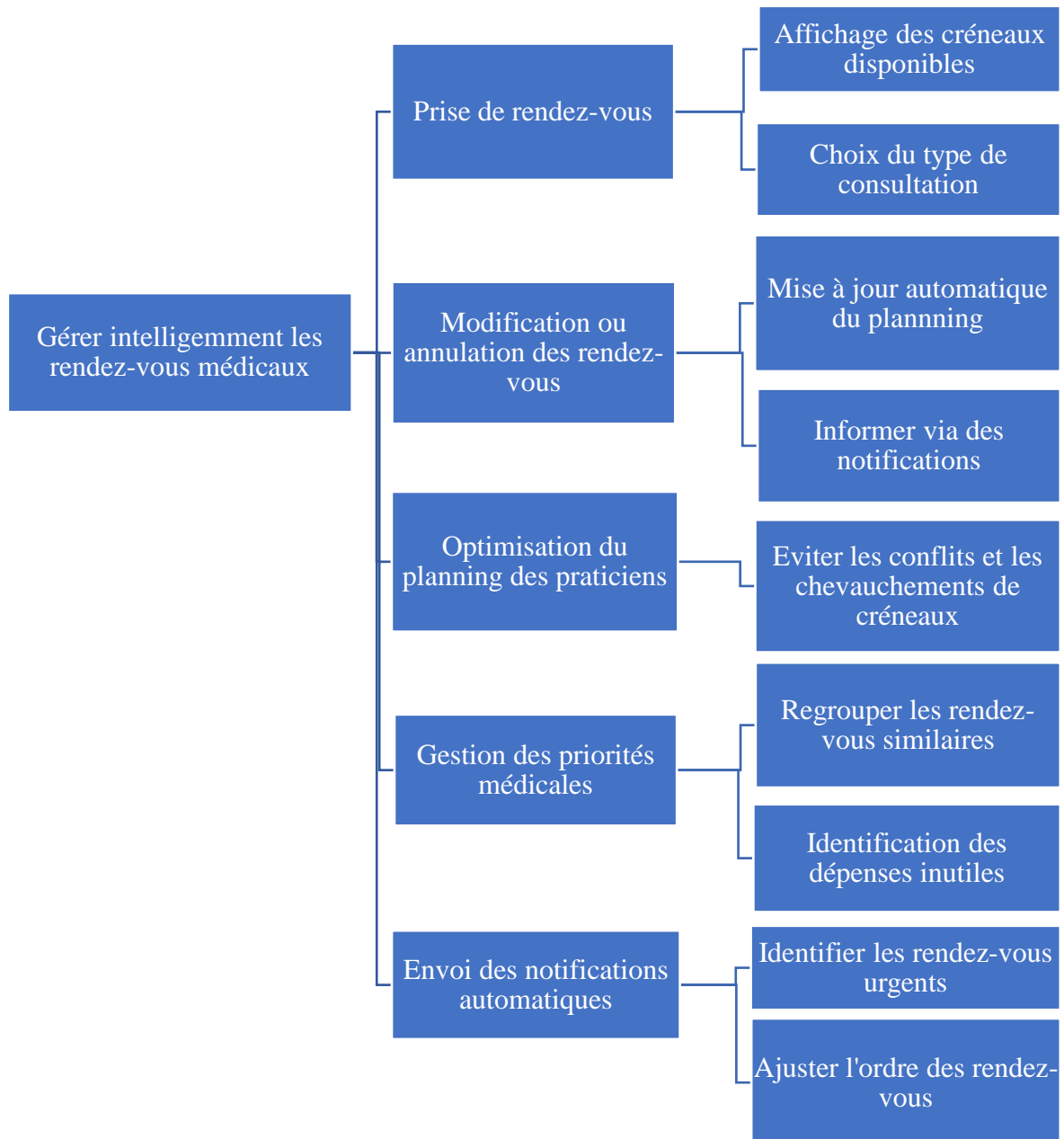


Figure 1 : Matrice FAST

## II.4. ANALYSE SWOT (STRENGTHS WEAKNESSES OPPORTUNITIES THREATS) DU PROJET

L'analyse SWOT est un outil stratégique pour évaluer les forces, faiblesses, opportunités et menaces d'un projet, d'une entreprise ou d'une situation particulière.

INTERNE	<p><b>FORCES (S)</b></p> <ul style="list-style-type: none"> <li>- Système complet couvrant prise de rendez-vous, téléconsultation, notifications, facturation.</li> <li>-Intégration d'intelligence artificielle pour optimisation des agendas.</li> <li>- Architecture modulaire et performante.</li> <li>- Conformité RGPD, HDS et normes médicales.</li> <li>- Expérience utilisateur fluide : web.</li> </ul>	<p><b>FAIBLESSES (W)</b></p> <ul style="list-style-type: none"> <li>- Dépendance à des services tiers (paiement, SMS, visioconférence).</li> <li>- Complexité technique importante (IA, sécurité, micro services).</li> <li>- Coût de développement et de maintenance élevé</li> <li>- Temps de mise en œuvre long (1 mois).</li> </ul>
	<p><b>OPPORTUNITES (O)</b></p> <ul style="list-style-type: none"> <li>- Digitalisation croissante du secteur médical.</li> <li>- Demande forte pour la réduction des délais et absences.</li> <li>- Potentiel d'expansion internationale.</li> <li>- Partenariats avec hôpitaux, cliniques et assurances.</li> <li>- Intégration future du DME, e-prescription, télésurveillance.</li> </ul>	<p><b>MENACES (T)</b></p> <ul style="list-style-type: none"> <li>- Concurrence forte (Doctolib, plateformes locales).</li> <li>- Risques de cybersécurité élevés.</li> <li>- Changements réglementaires possibles.</li> <li>- Faible adoption par certains praticiens peu digitalisés.</li> </ul>

Figure 2 : Matrice SWOT

## CHAPITRE III : CAHIER DE CHARGES ET CONCEPTION

Ce chapitre est consacré à l'analyse des besoins et à l'élaboration du cahier des charges du système. Il vise à formaliser les exigences fonctionnelles et non fonctionnelles nécessaires à la conception d'un système intelligent de gestion des rendez-vous médicaux.

### SECTION 1 : CAHIER DE CHARGES

#### III.1. EXIGENCES FONCTIONNELLES

Nous avons choisi de décomposer l'ensemble des besoins fonctionnels en modules, permettant ainsi d'appliquer la notion de programmation orientée fonctionnalité. Après un découpage, nous avons comme modules :

##### III.1.1. Module de Prise de Rendez-vous

- **Recherche et sélection du praticien**
  - Recherche multicritère : spécialité, nom, localisation, établissement
  - Filtres avancés : conventionné secteur 1/2, langues parlées, accessibilité PMR
  - Affichage des profils praticiens : photo, CV, spécialités, tarifs.
- **Sélection du créneau**
  - Calendrier interactif avec disponibilités en temps réel
  - Vue par jour/semaine/mois
  - Durées de consultation variables selon le type de RDV
  - Motifs de consultation prédéfinis (consultation, contrôle, urgence)
  - Proposition de créneaux alternatifs si indisponibilité
- **Confirmation et validation**
  - Création de compte patient simplifié ou connexion existante
  - Récapitulatif complet du RDV
  - Ajout au calendrier (personnalise)
  - Service google pour identification rapide
- **Gestion post-réservation**
  - Modification de RDV (dans les limites définies)
  - Annulation
  - Historique complet des actions

### **III.1.2. Module Gestion Intelligente des Agendas**

#### **➤ Configuration des agendas**

- Définition des horaires de travail par praticien
- Gestion des absences (congés, formations, gardes)
- Paramétrage des durées par type de consultation
- Temps de battement configurable entre consultations

#### **➤ Intelligence artificielle d'optimisation**

- Prédiction des absences patients basée sur l'historique
- Surbooking intelligent (overbooking calculé)
- Suggestion de réorganisation en cas d'annulation
- Priorisation automatique selon urgence
- Détection des patterns temporels (heures creuses/pleines)
- Machine learning pour améliorer les prédictions

#### **➤ Gestion des urgences**

- Créneaux d'urgence réservés et modulables
- Système de priorité pour cas urgents
- Réorganisation automatique du planning si nécessaire
- Notification des praticiens pour validation

#### **➤ Chatbot**

- **Conversation en temps réel**
- **Historique de la conversation (session)**
- **Accès aux FAQ**

#### **➤ Intelligence artificielle de recommandations de médecin pour la prise de rendez-vous**

- **Système de description des symptômes**
- **Analyse IA en temps réel**
- **Affichages médecins correspondants**
- **Badge "Recommandé par l'IA" sur les médecins suggérés**
- **Tri par disponibilité/proximité**

### III.1.3. Module Notifications et Rappels

#### ➤ **Système de rappels automatiques**

- Rappel à J-7 (premier rappel)
- Rappel à J-1 (rappel de confirmation)
- Rappel à H-2 (dernier rappel)
- Multi-canal : SMS, email, notification push, appel automatique
- Confirmation en un clic depuis le message
- Gestion des préférences de notification par patient

#### ➤ **Notifications praticiens**

- Alerte nouvelle prise de RDV
- Notification d'annulation
- Rappel début de journée avec liste patients
- Alerte patient en salle d'attente
- Notification retard/absence patient

#### ➤ **Notifications systèmes**

- Alertes techniques (dysfonctionnements)
- Notifications administratives
- Rappels de tâches en attente

### III.1.4. Module Gestion des Patients

#### ➤ **Dossier patient**

- Données d'identité : nom, prénom, date de naissance, sexe
- Coordonnées : adresse, téléphone, email
- Numéro de sécurité sociale
- Mutuelle et informations de couverture
- Médecin traitant
- Personnes à contacter en cas d'urgence
- Allergies et antécédents majeurs
- Historique des consultations avec dates et praticiens
- Documents joints (ordonnances, résultats d'examens)

#### ➤ **Gestion des accès**

- Portail patient avec authentification sécurisée (2FA)

- Consultation de l'historique médical
- Téléchargement des documents
- Mise à jour des informations personnelles
- Gestion des autorisations d'accès

### **III.1.5. Module Téléconsultation**

#### **➤ Visioconférence intégrée**

- Système de vidéo HD sécurisé et crypté
- Partage d'écran et de documents
- Salle d'attente virtuelle
- Enregistrement de consultation (avec consentement)
- Chat textuel intégré
- Compatibilité multi-appareils (desktop, mobile, tablette)

#### **➤ Gestion des téléconsultations**

- Planification comme RDV classique
- Vérification technique préalable (test connexion)
- Envoi automatique du lien de connexion
- Gestion des problèmes techniques avec bascule téléphone

### **III.1.6. Module Facturation et Paiement**

#### **➤ Gestion de la facturation**

- Création automatique de factures
- Nomenclature NGAP et CCAM intégrée
- Gestion du tiers payant
- Édition de feuilles de soins
- Transmission sécurisée aux organismes de santé
- Génération d'attestations fiscales

#### **➤ Paiement en ligne**

- Paiement sécurisé par carte bancaire (3D Secure)
- Stockage sécurisé des moyens de paiement
- Paiement à la réservation ou après consultation
- Gestion des remboursements
- Reçus automatiques

- Compatibilité avec standards PCI-DSS

### III.1.7. Module Reporting et Analytics

#### ➤ Tableaux de bord praticiens (KPIs affichés)

- Taux de remplissage de l'agenda
- Taux d'absences patients
- Durée moyenne de consultation
- Chiffre d'affaires et évolution
- Nombre de nouveaux patients
- Répartition par type de consultation
- Note moyenne et avis patients

#### ➤ Tableaux de bord administratifs (KPIs affichés)

- Statistiques globales multi-praticiens
- Taux d'utilisation de la plateforme
- Performances par établissement
- Évolution du nombre d'utilisateurs
- Analyses prédictives et tendances
- Rapports d'incidents et disponibilité

#### ➤ Exports et rapports

- Export Excel/CSV/PDF
- Rapports personnalisables
- Programmation d'envois automatiques
- Visualisations graphiques interactives

## III.2. EXIGENCES NON FONCTIONNELLES

- **Convivialité** : le système doit être convivial et facile à utiliser avec une interface utilisateur intuitive. Les fonctionnalités devront être accessibles rapidement sans nécessiter une courbe d'apprentissage complexe.
- **Performance** : le système doit être rapide et réactif avec des temps de chargement courts et une fluidité lors de la navigation entre les différentes sections.
- **Disponibilité multiplateforme** : le système doit être disponible sur différentes plateformes, telles que les smartphones, les tablettes et ordinateurs afin de permettre aux utilisateurs d'y accéder depuis n'importe quel appareil.



- **Confidentialité des données** : l'application doit garantir la confidentialité et la sécurité des données personnelles et financières des utilisateurs.
- **Sauvegarde et récupération des données** : l'application doit offrir des fonctionnalités de sauvegarde régulière des données et la possibilité de les restaurer en cas de perte ou de corruption accidentelle.
- **Evolutivité** : l'application doit être conçue pour être évolutive, afin de pouvoir prendre en charge un nombre croissant d'utilisateurs et de transactions avec des performances optimales.

### III.3. CONTRAINTES TECHNIQUES

#### III.3.1. Technologies recommandées

- **Frontend**
  - **Web** : React.js ou Vue.js avec TypeScript
  - **Mobile Android** : on verra
  - **Framework CSS** : Tailwind CSS ou Material Design
- **Backend**
  - **Langage** : Laravel
  - **ORM** : Eloquent ORM
  - **Validation** : Joi, Zod ou Pydantic
- **Intelligence artificielle**
  - **Framework ML** : TensorFlow ou PyTorch
  - **Prédiction** : Scikit-learn
  - **NLP** : spaCy ou Transformers
  - **Serving** : TensorFlow Serving ou TorchServe
- **Infrastructure cloud**
  - **Cloud** : AWS, Google Cloud ou Azure
  - **Conteneurisation** : Docker
  - **Orchestration** : Kubernetes
  - **CI/CD** : GitLab CI, GitHub Actions ou Jenkins
  - **Monitoring** : Prometheus + Grafana
  - **Logs** : ELK Stack ou Loki

### III.3.2. Sécurité et conformité

#### ➤ Sécurité applicative

- Authentification multi-facteurs (2FA/MFA)
- Chiffrement des données en transit (TLS 1.3)
- Chiffrement des données au repos (AES-256)
- Gestion sécurisée des secrets (Vault)
- Protection contre OWASP Top 10
- Rate limiting et protection DDoS
- Audit trails complets

#### ➤ Conformité réglementaire

**RGPD** (Règlement Général sur la Protection des Données) :

- Consentement explicite pour traitement des données
- Droit à l'oubli (suppression des données)
- Portabilité des données
- Registre des traitements
- DPO (Data Protection Officer) désigné
- Analyses d'impact (DPIA)

**Hébergement des données de santé (HDS) :**

- Certification HDS obligatoire pour l'hébergeur
- Traçabilité complète des accès
- Sauvegarde et plan de reprise d'activité
- Politique de gestion des incidents

**Normes médicales :**

- Respect du secret médical
- Conformité aux standards HL7 FHIR
- Interopérabilité avec systèmes existants

### III.3.3. Performance et scalabilité

#### ➤ Objectifs de performance

- Temps de réponse API : < 200ms (95e percentile)
- Temps de chargement pages : < 2s
- Disponibilité : 99.9% (SLA)

- Capacité : support de 100 000 utilisateurs simultanés
- Évolutivité horizontale automatique

➤ **Stratégies d'optimisation**

- Mise en cache multi-niveaux
- CDN pour contenus statiques
- Lazy loading et code splitting
- Optimisation des requêtes DB (indexation)
- Compression des données
- Load balancing

### III.3.4. Intégrations

➤ **Intégrations tierces**

- **Paieement** : Stripe, PayPal, ou solution bancaire locale
- **SMS** : Twilio, Vonage ou OVH Telecom
- **Email** : SendGrid, Amazon SES ou Mailgun
- **Calendriers** : Google Calendar, Outlook, Apple Calendar (API)
- **Cartes** : Google Maps API ou OpenStreetMap
- **Visioconférence** : Twilio Video, Zoom SDK ou solution propriétaire

➤ **APIs exposées**

- API REST documentée (OpenAPI/Swagger)
- Webhooks pour événements clés
- SDK pour intégrations tierces

## III.4. ANALYSE DES RISQUES

### III.4.1. Risques techniques

*Tableau 1 : Risques techniques*

Risques	Probabilité	Impact	Mesure d'atténuation
Problèmes de performance à l'échelle	Moyenne	Élevé	Tests de charge réguliers, architecture scalable

<b>Faibles de sécurité</b>	Moyenne	Critique	Audits sécurité, pentesting, formation équipe
<b>Indisponibilité du système</b>	Faible	Critique	Redondance, monitoring, PRA robuste
<b>Bugs critiques en production</b>	Moyenne	Élevé	Tests exhaustifs, déploiement progressif, rollback rapide

### III.4.2. Risques projet

Tableau 2 : Risques projet

<b>Risques</b>	<b>Probabilité</b>	<b>Impact</b>	<b>Mesures d'atténuation</b>
<b>Dépassement budget</b>	Moyenne	Élevé	Suivi rigoureux, validation jalons, réserve 15%
<b>Retard planning</b>	Élevée	Moyen	Méthodologie agile, priorisation MVP, ressources tampons
<b>Turnover équipe</b>	Moyenne	Moyen	Documentation, pair programming, rétention talents
<b>Changement réglementation</b>	Faible	Élevé	Veille juridique, architecture modulaire

### III.4.3. Risques métier

Tableau 3 : Risques métier

<b>Faible adoption par praticiens</b>	<b>Moyenne</b>	<b>Critique</b>	<b>Co-conception, formation, support, incitations</b>
<b>Faible adoption par patients</b>	Moyenne	Élevé	UX soigné, campagne communication, partenariats
<b>Concurrence accrue</b>	Élevée	Moyen	Innovation continue, différenciation IA, service client
<b>Non-conformité réglementaire</b>	Faible	Critique	Expert juridique, audits réguliers, certifications

## III.5. ESTIMATION DES COÛTS

Dans le cadre de ce projet, l'estimation des coûts à été réalisée selon une approche hybride. Cette approche combine une estimation par postes de coûts, permettant d'identifier et de chiffrer les ressources humaines, matérielles, logicielles et d'infrastructure, avec l'utilisation de la méthode COCOMO (Constructive Cost Model) de manière simplifiée. L'estimation par postes constitue la base principale de l'analyse, tandis que COCOMO est utilisée comme outil de validation afin d'obtenir un ordre de grandeur fiable. Cette démarche permet d'assurer à la fois le réalisme, la clarté et la rigueur méthodologique de l'estimation financière du projet.

### III.5.1. Les coûts humains

Ils constituent la part la plus importante de l'estimation, car ils correspondent aux efforts de conception, de développement, de test et de gestion du projet.

Tableau 4 : Les coûts humains

<b>Rôles</b>	<b>Coût mensuel</b>	<b>Durée</b>	<b>Coût total (FCFA)</b>
<b>Chef de projet</b>	330.000	1mois	330.000

<b>Analyste/ Concepteur</b>	220.000	1mois	220.000
<b>Dev backend</b>	286.000	1mois	286.000
<b>Dev frontend</b>	286.000	1mois	286.000
<b>Testeur</b>	132.000	1mois	132.000
<b>Total</b>			<b>1.254.000</b>

### III.5.2. Les coûts matériels

Ils concernent les équipements nécessaires à la réalisation du projet, tels que les ordinateurs et les moyens d'accès à Internet. Dans le cadre de ce projet, ces coûts restent limités, certains équipements étant déjà disponibles.

*Tableau 5 : Les coûts matériels*

<b>Elément</b>	<b>Coût d'achat</b>	<b>Durée de vie</b>	<b>Durée du projet</b>	<b>Coût amorti</b>
<b>Ordinateur portable</b>	250.000	36 mois	1 mois	34.722
<b>Connexion Internet</b>	20.000	/	1 mois	20.000
<b>Total</b>				<b>54.722</b>

### III.5.3. Les coûts Logiciels

Ils sont réduits grâce à l'utilisation majoritaire de solutions Open source, notamment pour le système d'exploitation, les environnements de développement, les frameworks et la base de données.

Tableau 6 : Les coûts logiciels

Logiciel	Type	Coût (FCFA)
OS (Linux, Windows)	Open source	0
IDE (VS Code)	Gratuit	0
Base de données (MySQL)	Open source	0
Framework (React)	Open source	0
Total		0

#### III.5.4. Les coûts d'Infrastructure

Ils regroupent les dépenses liées à l'hébergement du système, à la gestion du serveur et à l'acquisition éventuelle d'un nom de domaine. Ces coûts sont estimés sur une période donnée correspondant à la phase de déploiement et de tests du système.

Tableau 7 : Les coûts d'infrastructure

Elément	Durée	Coût (FCFA)
Hébergement serveur	1 an	30.000
Nom de domaine	1 an	10.000
Total		40.000

#### III.5.5. Les coûts de maintenance (post-déploiement)

Ils prennent en compte les interventions nécessaires après le déploiement, notamment la correction des anomalies et l'évolution du système. Ils permettent d'assurer la pérennité et la fiabilité de l'application dans le temps.

Tableau 8 : Les coûts de maintenance

Type	Coût estimé (FCFA)
Maintenance corrective	50.000
Maintenance évolutive	80.000
<b>Total</b>	<b>130.000</b>

### III.5.6. Les coûts imprévus

Une marge pour les imprévus est intégrée à l'estimation afin de couvrir les risques liés aux retards, aux ajustements techniques ou aux changements mineurs des besoins. Cette marge correspond généralement à un pourcentage du coût total estimé. Il est recommandé de prévoir 10% du coût total notamment : CI =

### III.5.7. La méthode COCOMO

- Type de projet : organique (équipe de petite taille, exigences bien définies, maîtrise des technologies utilisées, complexité modérée)
- Taille estimée : 10 KLOC
- Effort (en homme-mois) :  $E = 2.4 * (KLOC)^{1.05}$
- Durée de développement (en mois) :  $D = 2.5 * (E)^{0.38}$
- Calcul de l'effort :  $E = 2.4 * (10)^{1.05}$  ;  $E = 2.4 * 11.22 = 26.9$  homme-mois soit 27 homme-mois
- Calcul de la durée du projet :  $D = 2.5 * (26.9)^{0.38}$  ;  $D = 2.5 * 3.5 = 8.75$  mois soit 9 mois
- Taille de l'équipe :
  - Nombre de personnes =  $E / D$  ;  $26.9 / 8.75 = 3$  personnes
- Estimation du coût du projet : en supposant un coût moyen de 250.000 FCFA par personne et par mois : Coût total =  $26.9 * 250.000 = 6.725.000$  FCFA



### III.5.8 Bilan global

Tableau 9 : Bilan global

Poste	Coût
Coûts humains	1.254.000
Matériels	54.722
Logiciels	0
Infrastructure	40.000
Maintenance	130.000
Imprévus	147 872

Le coût total estimé du projet est de 1.626.594 FCFA ( un million six-cent vingt-six mille cinq-cents quatre-vingt-quatorze).

### III.6. LA PLANIFICATION DU TRAVAIL

La planification de notre travail représentée ci-dessous, décrit l'ensemble des actions que nous avons menées du début à la fin de notre projet. Chaque tâche effectuée et les jalons correspondant sont illustrés dans ce diagramme.

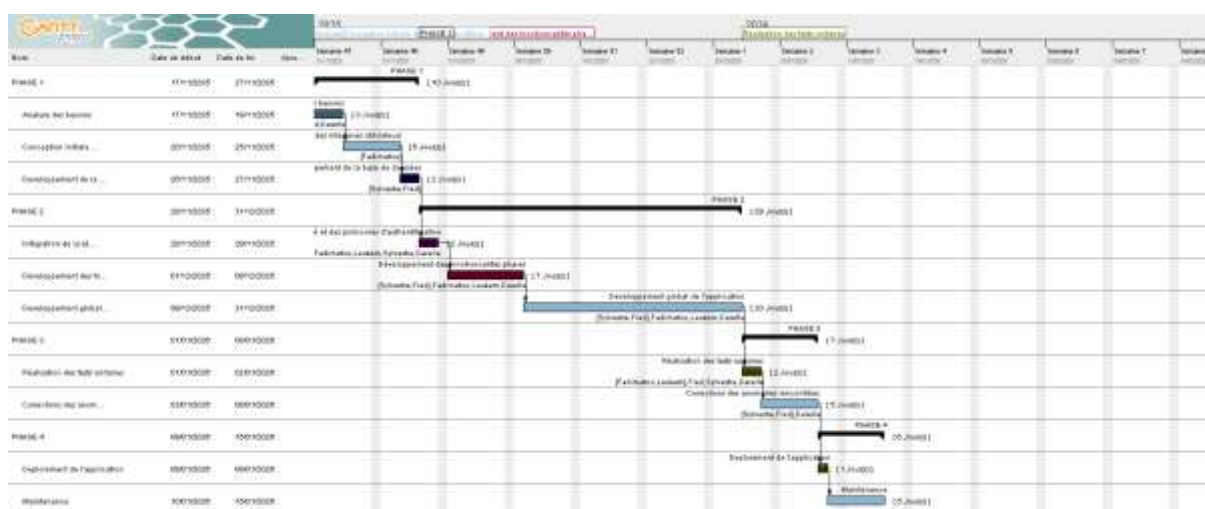


Figure 3 : Diagramme de Gantt

### **III.7. LIVRABLES ATTENDUS**

Au terme de la réalisation de ce projet, les livrables attendus dans le cadre de l'élaboration de cette application sont :

- Une application résolvant le problème soulevé
- Le cahier de charges du projet
- Le dossier de conception du projet
- Le manuel d'utilisation de l'application
- Le dossier technique de réalisation

## SECTION 2 : CONCEPTION DE LA SOLUTION

La conception d'un projet informatique est une phase primordiale pour définir les objectifs et les fonctionnalités de celui-ci. Elle consiste à définir comment le logiciel va être construit, en traduisant les besoins fonctionnels et non fonctionnels en solutions techniques concrètes.

### III.8. LANGAGE DE MODELISATION

#### III.8.1. Merise

Il s'agit d'une méthode d'analyse, de conception et de documentation des systèmes d'information. La méthode MERISE (Méthode d'Etude et de Réalisation pour les Systèmes d'Entreprise) comprend quatre niveaux d'abstraction à savoir :

- Le **niveau conceptuel** : il désigne les choix fondamentaux de gestion, et les objectifs de l'organisation, les enchainements et les règles de gestion. Il possède deux modèles : le Modèle Conceptuel de Données (MCD) et le Modèle Conceptuel de Traitement (MCT).
- Le **niveau organisationnel** : il définit la répartition de fonctionnement, les postes de travail, la volumétrie des données. Il se compose également de deux modèles : le Modèle Organisationnel de Données (MOD) et le Modèle Organisationnel de Traitement (MOT).
- Le **niveau logique** : ici on fait référence aux différents moyens mis en œuvre pour l'organisation, la structuration et la sauvegarde des données (bases de données)
- Le **niveau physique** : c'est le comment faire, il est question des ressources physiques nécessaire pour mettre en œuvre le projet.

	Données	Traitements	
<b>Niveau conceptuel</b>	<b>MCD</b> <i>Modèle Conceptuel de Données</i>	<b>MCT</b> <i>Modèle Conceptuel de Traitements</i>	<b>SIO</b> <i>Système d'Information Organisationnel</i>
<b>Niveau organisationnel</b>	<b>MOD</b> <i>Modèle Organisationnel de Données</i>	<b>MOT</b> <i>Modèle Organisationnel de Traitements</i>	
<b>Niveau logique</b>	<b>MLD</b> <i>Modèle Logique de Données</i>	<b>MLT</b> <i>Modèle Logique de Traitements</i>	<b>SII</b> <i>Système d'Information Informatisé</i>
<b>Niveau physique</b>	<b>MPD</b> <i>Modèle Physique de Données</i>	<b>MPT</b> <i>Modèle Physique de Traitements</i>	

Figure 4 : Niveaux d'abstraction de Merise

**Avantages** : MERISE permet une approche globale prenant en compte à la fois la modélisation des données et celle des traitements. Ce qui offre une meilleure adaptation, représentation et compréhension du système ainsi on obtient une application qui répond au mieux aux besoins.

**Inconvénients** : le fait d'employer une double démarche d'analyse peut se révéler fastidieux. Ceci rend également le temps d'élaboration du projet beaucoup plus long. Les développements faits entre les données et les traitements sont isolés et donc il n'est pas possible d'effectuer une fusion.

### III.8.2. UML (Unified Modeling Language)



Langage de Modélisation Unifié traduction de l'anglais Unified Modeling Language. Contrairement à MERISE, il s'appuie sur les diagrammes pour permettre de représenter graphiquement et de communiquer les divers aspects du système d'information. Il se compose de quatorze diagrammes.

Un diagramme est un élément graphique qui décrit une vue. On distingue deux grands groupes de diagrammes :

- **Diagrammes structurels ou diagrammes statiques** qui comprennent les diagrammes de classes, d'objets, de composants, de déploiement, de paquetages, de structures composites et de profil.
- **Diagrammes comportementaux ou diagrammes dynamiques** qui comprennent les diagrammes de cas d'utilisation, d'activités et d'états-transitions.
- **Diagrammes d'interaction** qui comprennent les diagrammes de séquence, de communication, de temps et le diagramme global d'interaction.

**Avantages** : le langage UML est formel et normalisé, il est donc facilement utilisable, de plus il est axé sur l'utilisation d'outils pour plus de précision. Le fait d'être graphique rend la compréhension de système abstraits et complexes mieux cernables. Il s'adapte facilement à la programmation.

**Inconvénients** : l'apprentissage nécessaire pour avoir une bonne maîtrise du langage et des outils est assez important. Pour fonctionner correctement, diagrammes UML doivent être synchronisés avec le code du logiciel, qui nécessite du temps pour mettre en place et à entretenir, et ajoute du travail à un projet de développement logiciel.

### III.8.3. Choix du langage de modélisation

En se basant sur les objectifs fixés pour la réalisation de notre application, il est évident pour nous que nous faisons face à une solution constituée de plusieurs modules qui devraient

rester ouverts à de possibles améliorations futures. De ce fait, nous trouvons important et même nécessaire d'utiliser un langage universel pour la modélisation afin de clarifier la conception et de mieux gérer les changements ultérieurs. Notre choix s'est donc porté sur le langage UML puisqu'il convient pour toutes les méthodes objet et permet la représentation d'un aspect spécifique du système grâce à ses différents diagrammes.

**UML** (Unified Modeling Language) est un langage de modélisation unifié qui permet de modéliser une solution informatique de façon standard dans le cadre d'une conception orientée objet. Il offre des règles, des notations et des diagrammes pour modéliser la structure, les fonctionnalités et les interactions du système afin de faciliter la communication, la compréhension et la documentation du système.

### III.9. METHODOLOGIE DE TRAVAIL : APPROCHE AGILE SCRUM

Scrum est une méthodologie agile de gestion de projet qui permet de livrer des produits par itérations courtes. Pour la réalisation de ce projet, nous avons utilisé la méthodologie **Scrum**, pour organiser notre travail sur une période d'un mois. Cette méthodologie facilite l'adaptation aux changements et encourage la collaboration étroite entre les membres de l'équipe et les parties prenantes.

#### III.9.1 Justification de l'utilisation de Scrum

Étant donné que le projet devait être développé en seulement un mois et nécessitait l'intégration progressive de fonctionnalités intelligentes, Scrum a été choisi pour sa capacité à organiser le travail en sprints courts, à prioriser les tâches et à ajuster rapidement les objectifs selon les besoins.

#### III.9.2. Description des éléments Scrum utilisés

La méthodologie Scrum repose sur plusieurs éléments clés qui ont été utilisés dans ce projet :

- **Le Product Backlog** : a permis de recenser et de prioriser l'ensemble des fonctionnalités du système, telles que la gestion des rendez-vous, des utilisateurs et des notifications.
- **Les Sprints** : d'une durée d'une semaine, ont servi à organiser le travail en itérations courtes avec des objectifs précis.
- **Le Sprint Planning** : a permis de sélectionner les tâches à réaliser à chaque itération.
- **Les Sprints Review** : ont permis de valider les fonctionnalités développées et d'ajuster les priorités pour la suite du projet.
- **Le Daily Scrum** : a facilité l'identification des tâches en cours et des difficultés.

### III.9.3. Formation de l'équipe

Notre équipe se composait des rôles suivants :

- **Product Owner** : responsable de la définition des exigences et de la priorisation des fonctionnalités. Il a assuré d'aligner le projet avec les attentes des utilisateurs.
- **Scrum Master** : il a assuré le bon déroulement du processus Scrum, facilitant les réunions et résolvant les obstacles rencontrés par l'équipe.
- **L'équipe de développement** : elle a travaillé sur la mise en œuvre des fonctionnalités définies dans le backlog.

### III.9.4. Application concrète de Scrum

La méthodologie Scrum a été appliquée pour structurer le développement du système intelligent de gestion des rendez-vous médicaux sur une durée d'un mois. Le projet a été organisé en quatre sprints hebdomadaires, permettant une progression itérative et un suivi régulier de l'avancement. Un Product Backlog a été défini au début du projet, regroupant les fonctionnalités principales telles que la prise de rendez-vous, la gestion des utilisateurs, les notifications et les fonctionnalités intelligentes d'aide à la planification. A chaque sprint, les fonctionnalités prioritaires ont été sélectionnées lors de la phase de planification et développées progressivement. A la fin de chaque sprint, les fonctionnalités réalisées ont été évaluées afin de vérifier leur conformité aux objectifs fixés et d'ajuster les priorités pour le sprint suivant. Cette approche a permis une meilleure organisation du travail, une adaptation rapide aux besoins du projet et une livraison progressive du système.

### III.9.5. Bénéfices de Scrum pour le projet

- ✓ Meilleure organisation du travail
- ✓ Livraison progressive de fonctionnalités
- ✓ Adaptation rapide aux modifications
- ✓ Suivi clair de l'avancement

## III.10. MODELISATIONS

### III.10.1 Description fonctionnelle

#### III.10.1.1. Identification des acteurs

- Les patients
- Les médecins / praticiens
- L'administrateur système

### III.10.1.2. Cas d'utilisations

Un cas d'utilisation est un artefact qui définit une séquence d'actions qui produisent un résultat concret pour la valeur. En fonction des acteurs, nous avons pu ressortir les cas d'utilisations suivants :

Tableau 10 : Cas d'utilisation

Acteurs	Cas d'utilisations
Les patients	<ul style="list-style-type: none"><li>• Recherche d'un praticien</li><li>• Prise de rendez-vous</li><li>• Annulation d'un rendez-vous</li><li>• Modification d'un rendez-vous</li><li>• Télécharger des documents</li><li>• Accéder à la téléconsultation</li><li>• Consulter son historique</li></ul>
Les médecins / praticiens	<ul style="list-style-type: none"><li>• Configurer leurs agendas</li><li>• Consulter les dossiers patients</li><li>• Gérer les disponibilités</li><li>• Valider les rendez-vous</li><li>• Consulter les statistiques</li><li>• Démarrer la téléconsultation</li><li>• Générer la facturation</li></ul>
L'administrateur	<ul style="list-style-type: none"><li>• Gérer les établissements</li><li>• Gérer les praticiens</li><li>• Gérer les patients</li><li>• Configurer le système</li><li>• Consulter les logs</li><li>• Extraire rapports</li></ul>

### III.10.1.3. Illustration du diagramme de cas d'utilisation



Figure 5 : Diagramme de cas d'utilisation



### III.10.2. Modélisation statique : Diagramme de Classe

Le diagramme de classes représente clairement la structure d'un système particulier en modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets. Très utilisés par les ingénieurs logiciels pour documenter l'architecture des logiciels, les diagrammes de classe sont un type de diagramme de structure qui décrit ce qui doit être présent dans le système modélisé.

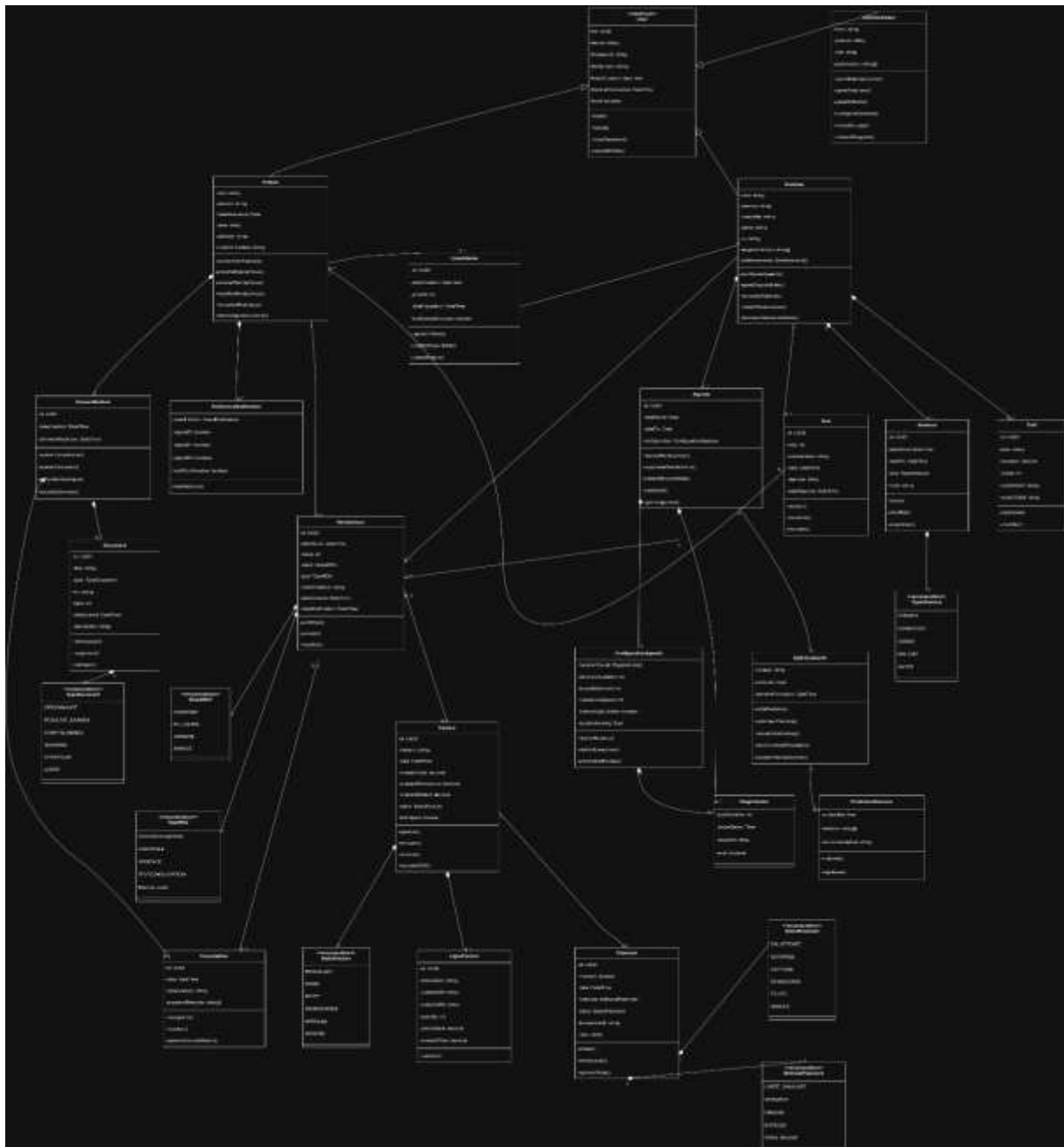


Figure 6 : Diagramme de classes

### **III.10.3. Modélisation dynamique : Diagramme de Séquences**

Le diagramme de séquence est un type de diagramme utilisé en génie logiciel pour modéliser la séquence d'interactions entre les objets d'un système au fil du temps. Il met l'accent sur la chronologie des messages échangés entre les objets pour accomplir une certaine fonctionnalité.

## **III.11. ARCHITECTURE DU SYSTÈME**

### **III.11.1. Architecture globale**

**Type :** Architecture microservices cloud-native

- Frontend : Application web responsive + Applications mobiles natives
- Backend : API REST/GraphQL
- Base de données : PostgreSQL (données relationnelles) + MongoDB (données non structurées)
- Cache : Redis
- Message broker : RabbitMQ ou Kafka
- Stockage fichiers : S3 ou équivalent

### **III.11.2. Services principaux**

- Service d'authentification et autorisation
- Service de gestion des rendez-vous
- Service de gestion des agendas
- Service de notifications
- Service de gestion des patients
- Service de téléconsultation
- Service de facturation
- Service d'analytics
- Service IA/ML pour optimisation

# CHAPITRE IV : IMPLÉMENTATION, INTERPRÉTATION ET DISCUSSION DES RÉSULTATS

Ce chapitre décrit l'ensemble des outils technologiques, des moyens de validation du travail, les interprétations vis-à-vis des résultats et les suggestions apportées en vue d'une amélioration future.

## SECTION 1 : ENVIRONNEMENT DE TRAVAIL

### IV-1. OUTILS DE DEVELOPPEMENT

Un environnement de développement est un ensemble d'outils qui permettent d'augmenter la productivité des programmeurs qui développent des logiciels. Il comporte un :

- ✓ Un éditeur de texte destiné à la programmation
- ✓ Des fonctions qui permettent, par pression sur un bouton, de démarrer le compilateur ou l'éditeur de liens
- ✓ Un débogueur en ligne qui permet d'exécuter ligne par ligne le programme

L'objectif d'un environnement de développement est :

- ✓ D'augmenter la productivité des programmeurs en automatisant une partie des activités et en simplifiant les opérations
- ✓ D'améliorer la qualité de la documentation en rapport avec le logiciel ou le site en construction

Dans le cadre de notre projet, nous avons utilisés les environnements de développement suivants :



#### ➤ **Visual Studio Code**

Visual Studio Code (VS Code) est un éditeur de code source léger, moderne et très populaire développé par Microsoft. C'est un éditeur de code puissant et polyvalent offrant une interface utilisateur conviviale, des fonctionnalités étendues et une flexibilité grâce à sa personnalisation et à son intégration avec une large gamme de langages et d'outils de développement.



### **Gantt Project**

Gantt Project est un logiciel de gestion de projet open source spécialement conçu pour la création de diagrammes de Gantt. Il offre aux utilisateurs les outils nécessaires pour planifier, suivre et gérer efficacement les projets.

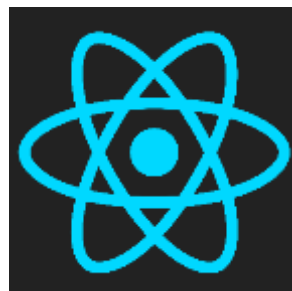


**Draw.io** est un outil de création de diagrammes utilisé pour représenter graphiquement des idées, des systèmes ou des processus.

## **IV-2. CHOIX TECHNOLOGIQUES**

### **IV-2.1. Programmation Frontend : Bibliothèque Javascript React.js**

**React (ou React.js)** est une bibliothèque Javascript utilisée pour créer des interfaces utilisateur (UI), elle permet de construire des pages web dynamiques, rapides et interactives en découpant l'interface en composants réutilisables.



*Figure 7 : Logo React.js*

### **IV-2.2. Programmation Backend : Framework Laravel**

**Laravel** est un framework PHP utilisé pour développer des applications web côté serveur (backend). Il fournit une structure et des outils prêts à l'emploi pour créer des applications web sécurisées, maintenables et performantes.



*Figure 8 : Logo Laravel*

### IV-2.3. Système de gestion de Base de Données

La création d'une application web demande à ce que celle-ci soit capable de communiquer avec la base de données, afin de conserver les données propres à l'application et à chaque utilisateur. Un **SGBD** est donc un logiciel qui permet de gérer les bases de données. On retrouve plusieurs SGBD permettant de réaliser les applications web : MONGO DB, MySQL, SQL server, Oracle server, etc. Dans le cadre de notre projet, notre choix s'est porté sur le SGBD **MySQL**.

- **Avantages de MySQL**

- Il est gratuit et Open Source
- Il possède une performance élevée
- Il est adapté à l'évolution
- Il est facile à utiliser
- Il possède une compatibilité multiplateforme

- **Limites de MySQL**

- La scalabilité verticale
- La gestion des données non structurées
- Limites de taille des bases de données
- Absences de certaines fonctionnalités avancées



*Figure 9 : Logo MySQL*

## SECTION 2 : TESTS, INTERPRETATION ET DISCUSSION

### DES RESULTATS

Dans cette partie, nous parlerons d'une série de tests que nous avons effectués sur l'application afin de vérifier son bon fonctionnement et la fiabilité de ses fonctionnalités. De ce fait, nous avons documenté chacun de ces tests de manière détaillée, en fournissant des explications claires.

#### IV-3. TESTS FONCTIONNELS

Dans cette partie, nous parlerons d'une série de tests que nous avons effectués sur l'application afin de vérifier son bon fonctionnement et la fiabilité de ses fonctionnalités. De ce fait, nous avons documenté chacun de ces tests de manière détaillée, en fournissant des explications claires. . Dans le cadre de notre phase de test, nous évaluerons les fonctionnalités ci-après :

- Authentification
- Prise de rendez-vous
- Définition des disponibilités du médecin

##### IV-3.1. Authentification

Prérequis : posséder un compte en tant que patient, administrateur ou médecin

Environnement de test : ordinateur portable

Testeur : Equipe projet

Tableau 11 : Test d'authentification

N°	Action	Attendu	Résultat
1	Entrer l'url de l'application dans la barre de recherche	Affichage de l'interface d'accueil	OK
2	Cliquer sur le bouton « Connexion »	Affichage du formulaire de connexion	OK
3	Remplir des informations erronées	Message d'erreur	OK
4	Remplir des informations correctes	Accès à la page d'accueil personnalisée	OK

### IV.3.2. Prise de rendez-vous

Prérequis : posséder un compte en tant que patient

Environnement de test : ordinateur portable

Testeur : Equipe projet

Tableau 12 : Test de prise de rendez-vous

N°	Action	Attendu	Résultat
1	Cliquez sur le bouton « rendez-vous » de la page d'accueil	Affichage du formulaire de prise de rendez-vous	OK
2	Choisir le type de rendez-vous	Affichage du pop-up du type de rendez-vous	OK
3	Choisir la date et l'heure du rendez-vous	Affichage du pop-up du créneau horaire	OK
4	Valider la demande de prise de rendez-vous	Message de confirmation d'envoi	OK

### IV.3.3. Définition des disponibilités des médecins

Prérequis : posséder un compte en tant que médecin

Environnement de test : ordinateur portable

Testeur : Equipe projet

Tableau 13 : Test de définition du créneau horaire du médecin

N°	Action	Attendu	Résultat
1	Cliquez sur l'icône du médecin de la page d'accueil	Affichage du tableau de bord du médecin	OK
2	Dans la barre verticale, cliquez sur « Mes créneaux »	Affichage de l'interface de définition des créneaux	OK
3	Ajuster le planning	Changement des horaires	OK
4	Enregistrer les créneaux définis	Message de confirmation	OK

## IV-4. SCHEMAS ILLUSTRATIFS

La figure ci-dessous représente la première interface qui s'ouvre à l'utilisateur lorsqu'il accède à l'application et elle se compose des boutons de « rendez-vous », « contact » ainsi qu'une barre de recherche.



Figure 10 : Page d'accueil Medismart

Cette interface est celle des recommandations fournies par l'intelligence artificielle de notre système suite à des problèmes posés par les patients.

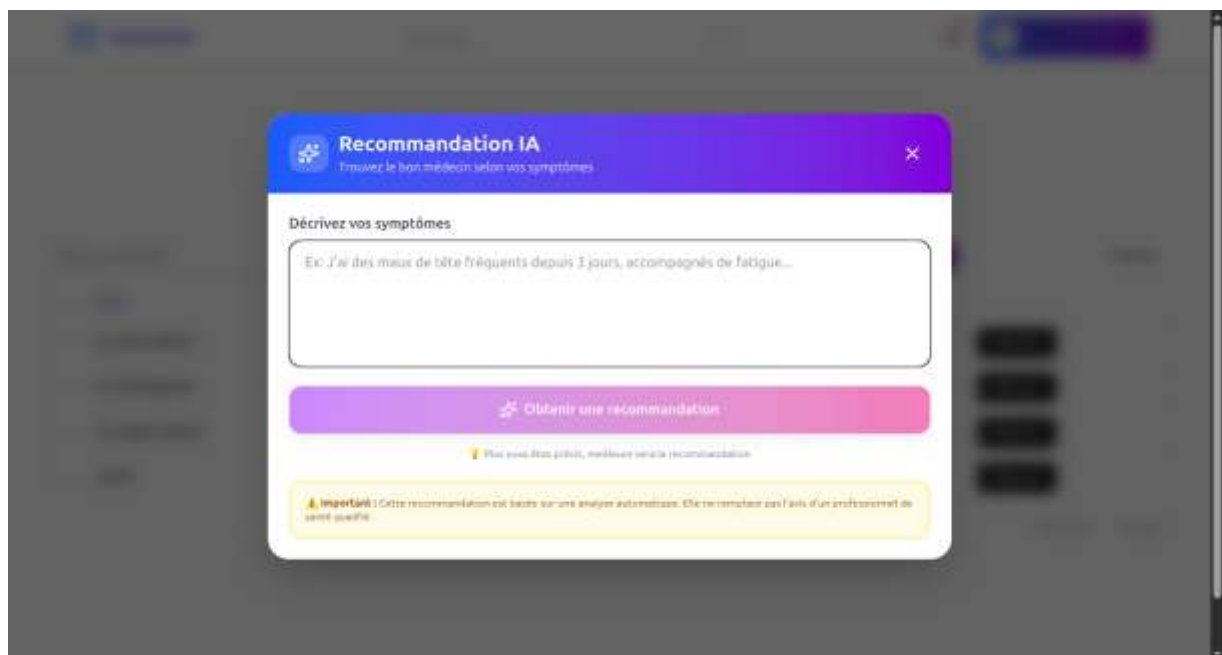


Figure 11 : Recommandations IA



Ici, nous avons une liste de choix du type de rendez-vous fournie par le formulaire de prise de rendez-vous.

The screenshot shows the MedisMat interface for Dr. Marie Dubois, a Cardiologist at Hôpital Central de Douala. The 'Type de consultation' section offers four options: 'Consultation générale' (selected), 'Téléconsultation', 'Consultation d'urgence', and 'Bilan de santé'. Below this, there is a 'Horaire' section with a button to 'Choisir l'heure' for the date '19 Dec 2023'. The interface includes a sidebar with filters and a top navigation bar with the MedisMat logo and user profile.

Figure 12 : Formulaire de prise de rendez-vous

Ceci est le formulaire de prise de rendez-vous avec choix du type de rendez-vous ainsi que de la date et l'heure du rendez-vous.

This screenshot shows the same MedisMat interface as Figure 12, but with the date and time selection options expanded. A calendar for December 2023 is displayed, with the 19th highlighted. To the right of the calendar, a list of time slots is shown for Friday, 19 Dec 2023, ranging from 09h00-10h00 to 16h00-17h00. The 'Bilan de santé' option is also visible in the 'Type de consultation' section. The interface includes a sidebar with filters and a top navigation bar with the MedisMat logo and user profile.

Figure 13 : Champs du formulaire de prise de rendez-vous

Ceci est l'interface de définition et de gestion des disponibilités des praticiens ou médecins.

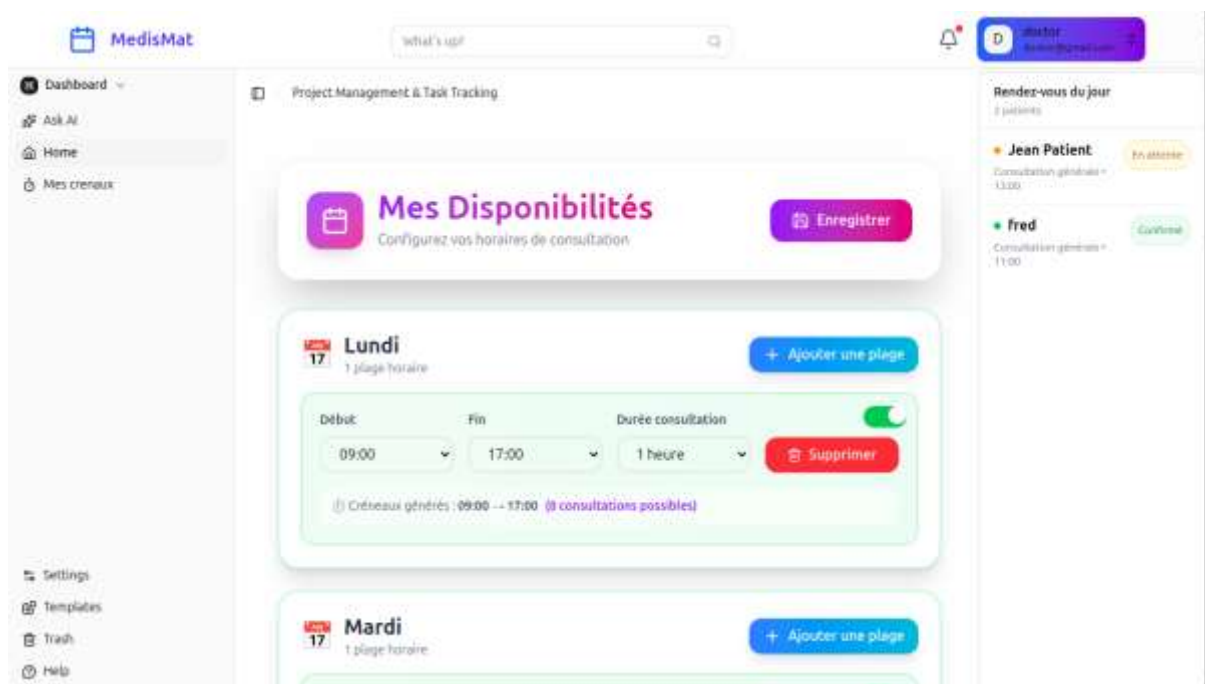


Figure 14 : Interface de définition du créneau horaire du médecin

La figure ci-dessous représente le tableau de bord du médecin sur lequel il peut voir tous les rendez-vous de la journée, ceux annulés ainsi que son planning de la semaine.

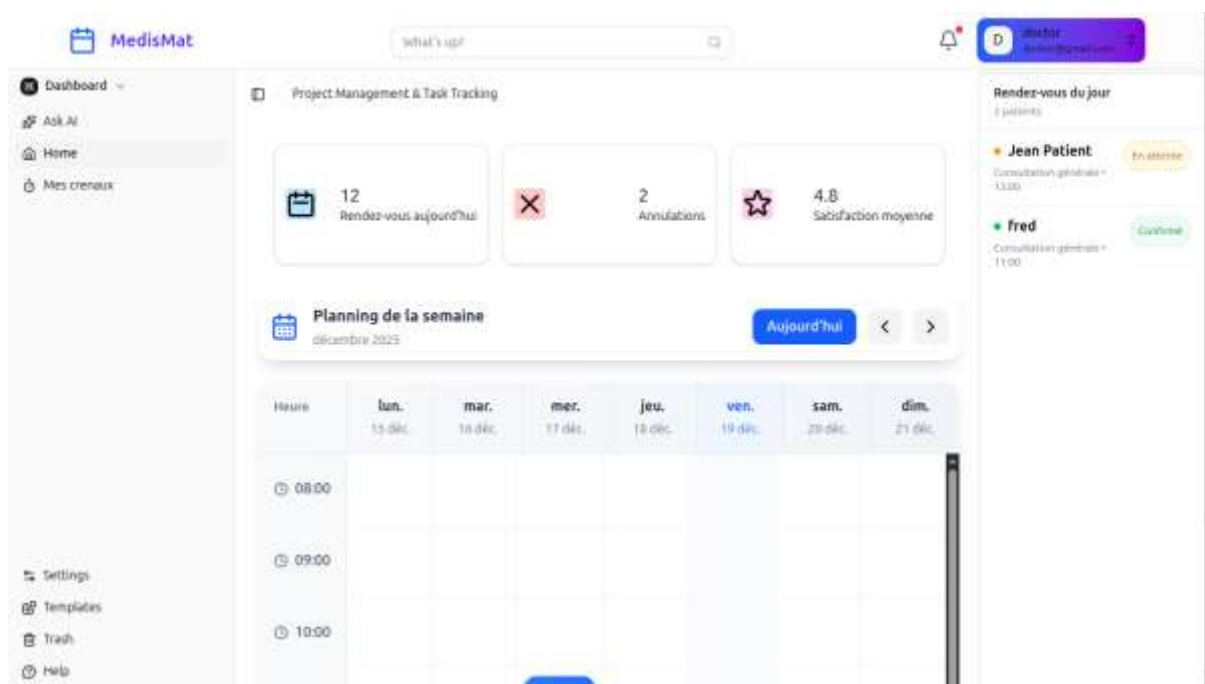


Figure 15 : Dashboard médecin

Ci-dessous l'interface des paramètres patients, l'historique des consultations, les documents générés ainsi que leurs informations personnelles.

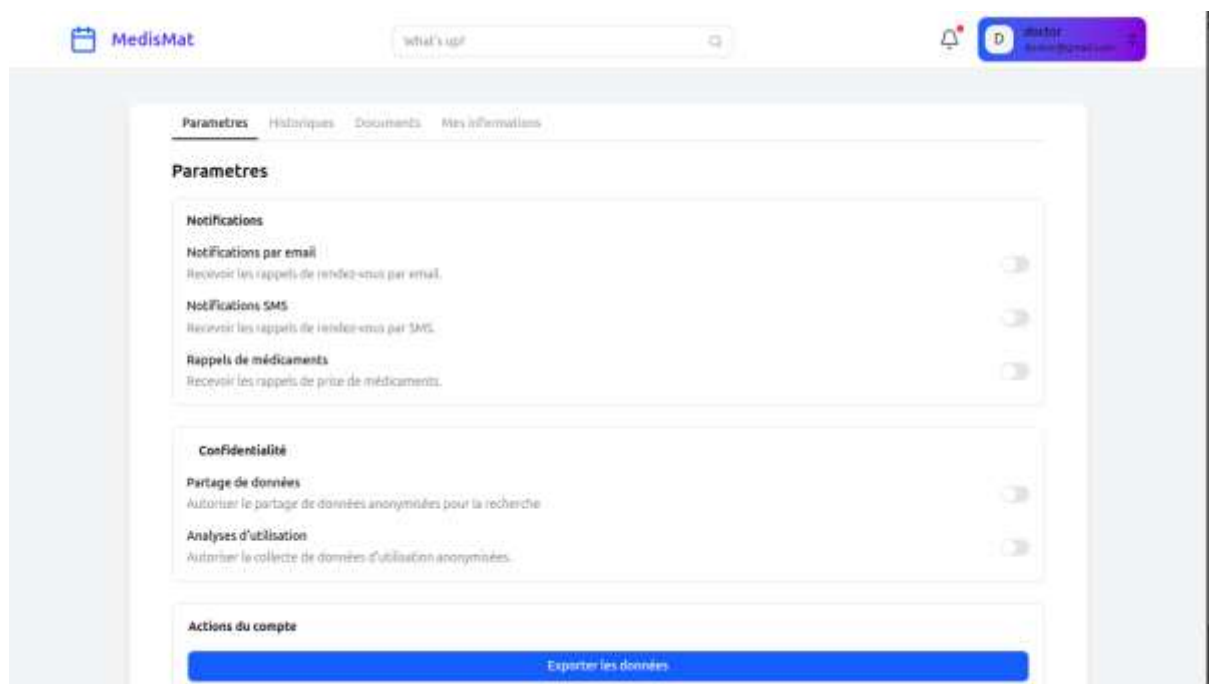


Figure 16 : Paramètres patients

Ci-dessous, l'interface de modification du statut du rendez-vous (confirmé, en attente, rejeté).

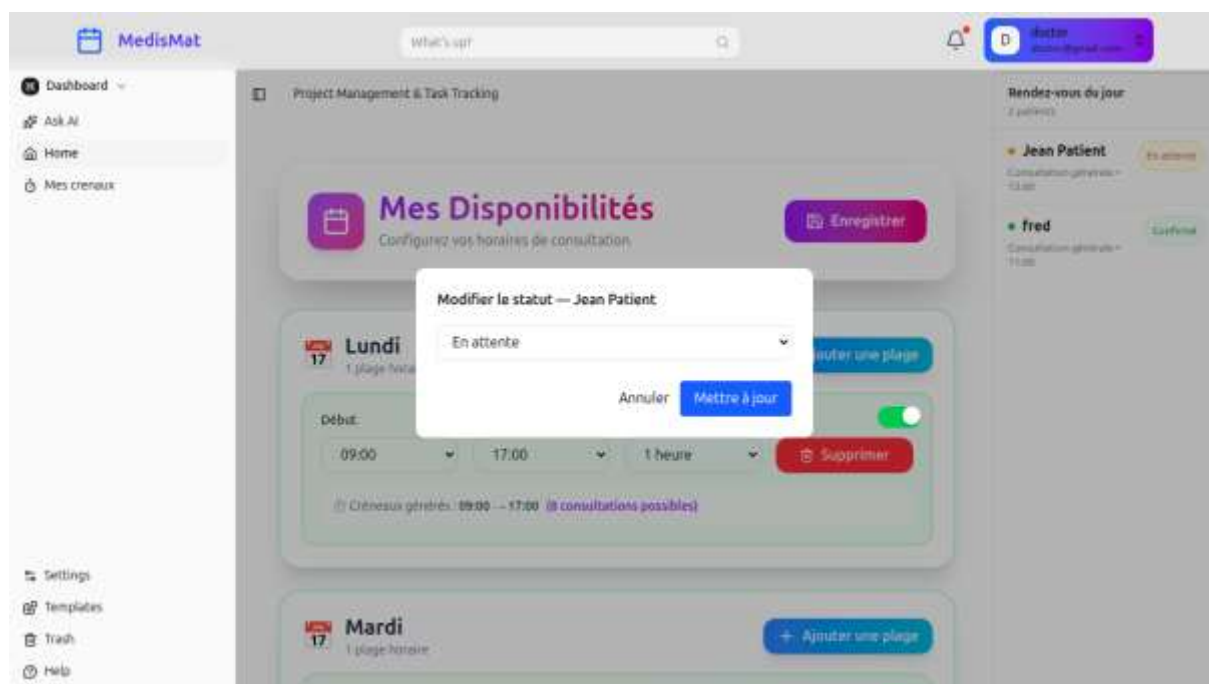


Figure 17 : Modification du statut de consultation

Ci-dessous, l'interface de téléversement des documents patients.

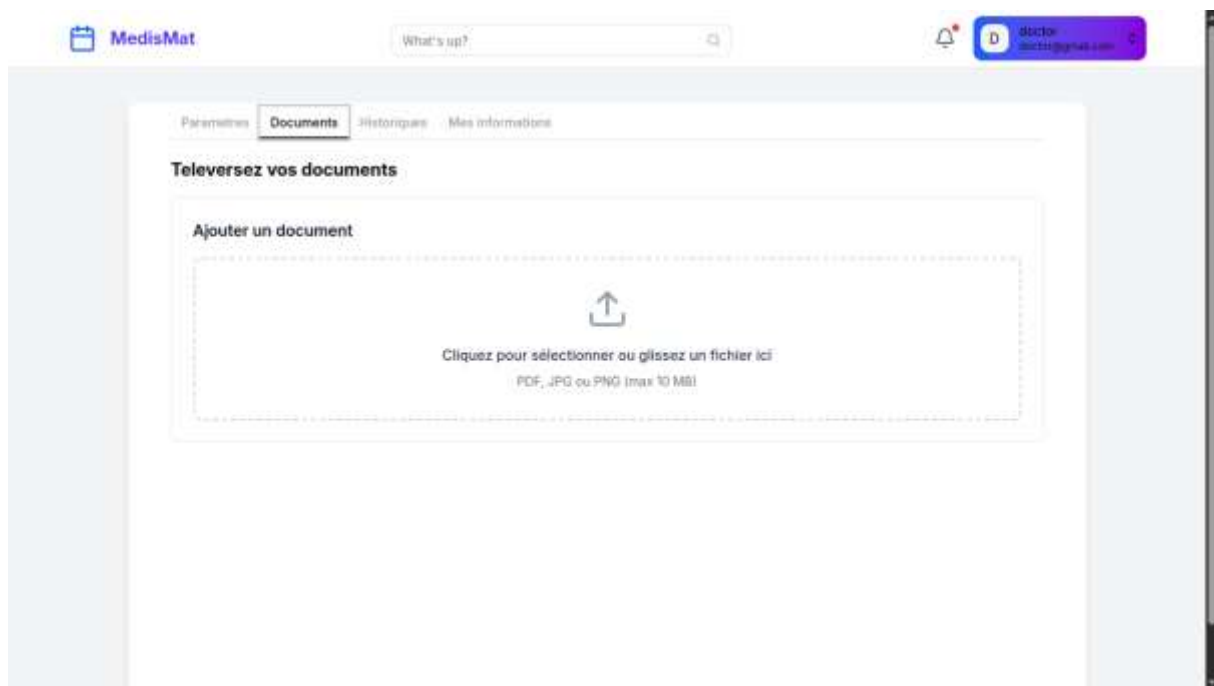


Figure 18 : Téléversement des dossiers patients

#### IV-5. INTERPRETATION ET DISCUSSION DES RESULTATS

- **Authentification** : les tests montrent que le système valide correctement les identifiants, refuse les accès non autorisés et applique une gestion efficace des rôles, garantissant la sécurité et la confidentialité des données.
- **Prise de rendez-vous** : les résultats confirment que les rendez-vous sont enregistrés sans erreur, intégrés immédiatement au planning du praticien et que les conflits de créneaux sont systématiquement évités.
- **Définition des créneaux horaires des médecins** : les tests démontrent que les plages de disponibilité sont correctement définies, mises à jour en temps réel et qu'aucun chevauchement incohérent n'est autorisé.

#### IV-6. SUGGESTIONS

- **Intégration de la téléconsultation** pour permettre des consultations à distance.
- **Rappels intelligents personnalisés** afin de réduire davantage les absences.
- **Analyse statistique des rendez-vous** pour aider à la prise de décision.
- **Interopérabilité avec d'autres systèmes de santé** (dossiers médicaux, laboratoires).
- **Amélioration de l'ergonomie mobile** pour une meilleure expérience utilisateur.

## CONCLUSION GENERALE

Ce projet a porté sur la conception d'un système intelligent de gestion des rendez-vous médicaux visant à améliorer l'organisation des consultations et à optimiser l'interaction entre patients et professionnels de santé. A travers l'analyse des besoins, la modélisation du système et la définition de son architecture, une solution numérique cohérente et adaptée aux contraintes du domaine médical a été proposée.

L'étude fonctionnelle appuyée par des outils tels que la matrice FAST et les diagrammes UML, a permis d'identifier les fonctions essentielles du système ainsi que les mécanismes nécessaires à leur mise en œuvre. L'estimation des coûts, réalisée selon une approche hybride combinant une analyse par postes et la méthode COCOMO, a permis d'évaluer la faisabilité économique du projet tout en assurant une rigueur méthodologique.

Le système proposé contribue à la réduction des conflits de planning, à l'amélioration de la ponctualité et à une meilleure gestion des priorités médicales, tout en garantissant la sécurité et la confidentialité des données. En perspective, ce projet pourrait être enrichi par l'intégration de fonctionnalités telles que la téléconsultation, l'analyse statistique avancée ou l'interopérabilité avec d'autres systèmes de santé. Ainsi, ce travail constitue une base solide pour la mise en place d'une solution moderne et évolutive au service de la digitalisation du secteur médical.

## TABLE DES MATIERES

<b>SOMMAIRE .....</b>	<b>1</b>
<b>LISTE DE FIGURES .....</b>	<b>2</b>
<b>LISTE DES TABLEAUX .....</b>	<b>3</b>
<b>INTRODUCTION GENERALE .....</b>	<b>4</b>
<b>CHAPITRE I : PRESENTATION DU PROJET .....</b>	<b>5</b>
<b>I.1. DESCRIPTION DU PROJET .....</b>	<b>5</b>
<b>I.2. CONTEXTE DU PROJET .....</b>	<b>5</b>
<b>I.3. OBJECTIFS DU PROJET .....</b>	<b>5</b>
<b>I.4. CIBLES DU PROJET .....</b>	<b>6</b>
<b>CHAPITRE II : ETUDE DE L'EXISTANT ET ANALYSE DU PROJET .....</b>	<b>7</b>
<b>II.1. PRESENTATION DES SYSTEMES EXISTANTS .....</b>	<b>7</b>
<b>II.2. ANALYSE DES FORCES ET FAIBLESSES DE L'EXISTANT .....</b>	<b>7</b>
<b>II.2.1. Forces des systèmes actuels .....</b>	<b>7</b>
<b>II.2.2. Faiblesses des systèmes actuels .....</b>	<b>7</b>
<b>II.3. ANALYSE FAST ( FUNCTION ANALYSIS SYSTEM TECHNIQUE) .....</b>	<b>8</b>
<b>II.4. ANALYSE SWOT (STRENGTHS WEAKNESSES OPPORTUNITIES THREATS) DU PROJET .....</b>	<b>9</b>
<b>CHAPITRE III : CAHIER DE CHARGES ET CONCEPTION .....</b>	<b>11</b>
<b>SECTION 1 : CAHIER DE CHARGES .....</b>	<b>11</b>
<b>III.1. EXIGENCES FONCTIONNELLES .....</b>	<b>11</b>
<b>III.1.1. Module de Prise de Rendez-vous .....</b>	<b>11</b>
<b>III.1.2. Module Gestion Intelligente des Agendas .....</b>	<b>12</b>
<b>III.1.3. Module Notifications et Rappels .....</b>	<b>13</b>
<b>III.1.4. Module Gestion des Patients .....</b>	<b>13</b>
<b>III.1.5. Module Téléconsultation .....</b>	<b>14</b>
<b>III.1.6. Module Facturation et Paiement .....</b>	<b>14</b>

<b>III.1.7. Module Reporting et Analytics</b> .....	15
<b>III.2. EXIGENCES NON FONCTIONNELLES</b> .....	15
<b>III.3. CONTRAINTES TECHNIQUES</b> .....	16
<b>III.3.1. Technologies recommandées</b> .....	16
<b>III.3.2. Sécurité et conformité</b> .....	17
<b>III.3.3. Performance et scalabilité</b> .....	17
<b>III.3.4. Intégrations</b> .....	18
<b>III.4. ANALYSE DES RISQUES</b> .....	18
<b>III.4.1. Risques techniques</b> .....	18
<b>III.4.2. Risques projet</b> .....	19
<b>III.4.3. Risques métier</b> .....	20
<b>III.5. ESTIMATION DES COÛTS</b> .....	20
<b>III.5.1. Les coûts humains</b> .....	20
<b>III.5.2. Les coûts matériels</b> .....	21
<b>III.5.3. Les coûts Logiciels</b> .....	21
<b>III.5.4. Les coûts d'Infrastructure</b> .....	22
<b>III.5.5. Les coûts de maintenance (post-déploiement)</b> .....	22
<b>III.5.6. Les coûts imprévus</b> .....	23
<b>III.5.7. La méthode COCOMO</b> .....	23
<b>III.5.8 Bilan global</b> .....	24
<b>III.6. LA PLANIFICATION DU TRAVAIL</b> .....	24
<b>III.7. LIVRABLES ATTENDUS</b> .....	25
<b>SECTION 2 : CONCEPTION DE LA SOLUTION</b> .....	26
<b>III.8. LANGAGE DE MODELISATION</b> .....	26
<b>III.8.1. Merise</b> .....	26
<b>III.8.2. UML (Unified Modeling Language)</b> .....	27
<b>III.8.3. Choix du langage de modélisation</b> .....	27

<b>III.9. METHODOLOGIE DE TRAVAIL : APPROCHE AGILE SCRUM.....</b>	<b>28</b>
<b>III.9.1 Justification de l'utilisation de Scrum .....</b>	<b>28</b>
<b>III.9.2. Description des éléments Scrum utilisés.....</b>	<b>28</b>
<b>III.9.3. Formation de l'équipe .....</b>	<b>29</b>
<b>III.9.4. Application concrète de Scrum .....</b>	<b>29</b>
<b>III.9.5. Bénéfices de Scrum pour le projet .....</b>	<b>29</b>
<b>III.10. MODELISATIONS .....</b>	<b>29</b>
<b>III.10.1 Description fonctionnelle .....</b>	<b>29</b>
<b>III.10.1.1. Identification des acteurs .....</b>	<b>29</b>
<b>III.10.1.2. Cas d'utilisations.....</b>	<b>30</b>
<b>III.10.1.3. Illustration du diagramme de cas d'utilisation .....</b>	<b>31</b>
<b>III.10.2. Modélisation statique : Diagramme de Classe.....</b>	<b>32</b>
<b>III.10.3. Modélisation dynamique : Diagramme de Séquences .....</b>	<b>33</b>
<b>III.11. ARCHITECTURE DU SYSTÈME.....</b>	<b>33</b>
<b>III.11.1. Architecture globale .....</b>	<b>33</b>
<b>III.11.2. Services principaux .....</b>	<b>33</b>
<b>CHAPITRE IV : IMPLÉMENTATION, INTERPRÉTATION ET DISCUSSION DES</b>	
<b>RÉSULTATS .....</b>	<b>34</b>
<b>SECTION 1 : ENVIRONNEMENT DE TRAVAIL .....</b>	<b>34</b>
<b>IV-1. OUTILS DE DEVELOPPEMENT .....</b>	<b>34</b>
<b>IV-2. CHOIX TECHNOLOGIQUES .....</b>	<b>35</b>
<b>IV-2.1. Programmation Frontend : Bibliothèque Javascript React.js.....</b>	<b>35</b>
<b>IV-2.2. Programmation Backend : Framework Laravel.....</b>	<b>35</b>
<b>IV-2.3. Système de gestion de Base de Données .....</b>	<b>36</b>
<b>SECTION 2 : TESTS, INTERPRETATION ET DISCUSHION DES RESULTATS .....</b>	<b>37</b>
<b>IV-3. TESTS FONCTIONNELS .....</b>	<b>37</b>
<b>IV-3.1. Authentification.....</b>	<b>37</b>



<b>IV.3.2. Prise de rendez-vous .....</b>	<b>38</b>
<b>IV.3.3. Définition des disponibilités des médecins .....</b>	<b>38</b>
<b>IV-4. SCHEMAS ILLUSTRATIFS .....</b>	<b>39</b>
<b>IV-5. INTERPRETATION ET DISCUSSION DES RESULTATS.....</b>	<b>43</b>
<b>IV-6. SUGGESTIONS.....</b>	<b>43</b>
<b>CONCLUSION GENERALE .....</b>	<b>44</b>
<b>TABLE DES MATIERES .....</b>	<b>45</b>