

Cahier des charges



Le projet consiste à concevoir un système de gestion de planning pour un cabinet médical.

Ce système doit permettre aux secrétaires de consulter et gérer les rendez-vous.

1. Présentation du projet

1.1 Contexte

Le projet MédiPlan vise à concevoir une solution de gestion de planning pour un cabinet médical, remplaçant les processus manuels actuels. L'objectif est de faciliter la gestion des rendez-vous, optimiser la gestion des plannings des médecins, et créer des alertes. Cette application, accessible via une interface JavaFX, permettra une gestion plus fluide des créneaux horaires.

1.2 Objectifs

Le système de gestion de planning doit atteindre les objectifs suivants :

- Optimiser la gestion du planning du secrétariat.
- Améliorer la communication et la coordination entre les différents acteurs (médecins, secrétaire).
- Fournir un accès sécurisé au secrétaire.
- Faciliter la gestion des rendez-vous en choisissant des créneaux horaires disponibles et en permettant les réservations instantanées.

1.3 Description de l'existant

Actuellement, le cabinet médical utilise un système manuel pour la gestion des plannings des médecins, ce qui génère des erreurs humaines, des doublons de rendez-vous et une gestion difficile des créneaux horaires. Aucun outil numérique n'est utilisé pour gérer les réservations des salles de consultations. L'objectif du projet est donc de développer un logiciel de gestion de planning qui viendra remplacer cette gestion manuelle par une solution informatisée, intuitive et performante.

Environnement logiciel et matériel :

- Le système sera accessible via une installation interne.

- La base de données sera gérée par MySQL.
- Les utilisateurs auront accès à des ordinateurs ou dispositifs mobiles connectés au réseau interne du cabinet.

1.4 Critères d'acceptabilité du produit

- **Procédure de validation** : Le produit sera testé lors de la phase de validation, avec des tests fonctionnels et de performance réalisés sur la base des critères définis.
 - **Critères d'acceptation** :
 - La gestion des plannings doit être fluide, sans erreurs de réservation.
 - Le système doit être accessible à tout moment, avec un taux de disponibilité supérieur à 99%.
 - Le temps de réponse du système lors de la consultation des plannings ne doit pas excéder 5 secondes.
-

2. Expression des besoins

2.1 Besoins fonctionnels

Les besoins fonctionnels sont les suivants :

1. Gestion des utilisateurs : Création, modification et suppression des comptes utilisateurs.
2. Consultation des plannings.
3. Gestion des conflits de réservation : Ne pas mettre 2 patients sur le même créneau.
4. Alertes : Envoi d'une alerte lorsqu'un rendez-vous est proche.

2.2 Besoins non fonctionnels

1. Sécurité : Le système doit être sécurisé et un chiffrement des mots de passe.
 2. Accessibilité : Le système doit être accessible localement.
 3. Fiabilité : Le système doit garantir un taux de disponibilité de 99% pendant les heures de service.
-

3. Contraintes

3.1 Coûts

- **Budget alloué** : Aucun.
- **Moyens matériels et logiciels** :
 - Serveurs pour héberger l'application et la base de données, MySQL.
 - Outils de développement (IntelliJ.) et logiciels de gestion de projet (SceneBuilder).

3.2 Délais

- **Date de livraison** : La solution doit être opérationnelle dans 2 mois après le début du développement.
- **Échéances intermédiaires** :
 - Phase de conception : 5 jours.
 - Phase de développement : 4 semaines.
 - Phase de test et validation : 2 semaines.
 - Phase de déploiement : 1 semaine.

3.3 Autres contraintes

- **Normes techniques** : Le projet doit respecter les normes de sécurité des données personnelles des patients (RGPD).
 - **Conformité aux exigences légales** : Respect des réglementations en matière de santé et de confidentialité des données.
-

4. Déroulement du projet

4.1 Planification

- Phase 1 : Analyse des besoins et spécifications (1 semaine).
- Phase 2 : Conception et architecture du système (2 semaines).
- Phase 3 : Développement du système (3 semaines).
- Phase 4 : Tests et validation (2 semaines).

4.2 Plan d'assurance qualité

Le contrôle qualité sera assuré par la mise en place de :

- Tests fonctionnels pour vérifier que chaque fonctionnalité répond aux besoins définis.
- Tests de performance pour s'assurer que le système peut supporter la charge des

utilisateurs.

- Tests de sécurité pour valider la conformité aux normes de protection des données.

4.3 Documentation

La documentation suivante sera fournie :

- **Documentation utilisateur** : Leurs seront donner leur identifiant et leur mot de passe.
- **Documentation technique** : Description de l'architecture du système, des bases de données.
- **Manuel d'installation et de déploiement.**