Projet de Programmation

En C++

Sujet n°6 : **Système de gestion des patients dans un centre de santé**

**Membres du groupe :**

* TSAFACK NGOUFACK Ernis Merkel
* NANA Laurianne Audrey
* NOUBISSI MEIKEM Indira Leslie
* TANKAM SOH Daril

**Chargée de cours** : Ing. ATANGANA Guy Martial

**Plan**

1. **Introduction**
   1. But du document
   2. Portée du projet
   3. Définitions, Acronymes et Abréviations
   4. Références
   5. Vue d’ensemble du reste du document
2. **Description Générale** 
   1. Contexte du produit
   2. Besoins des utilisateurs
   3. Caractéristiques du produit
   4. Contraintes
   5. Hypothèses et Dépendances
3. **Exigences spécifiques** 
   1. Exigences fonctionnelles
   2. Exigences non fonctionnelles
   3. Exigences d’Interface Externe
   4. Autres exigences
4. **Préparation à la livraison** 
   1. Plan de déploiement
   2. Exigences de formation
   3. Exigences de Documentation
   4. Exigences de Maintenance
5. **Annexes**
6. **Introduction**

**1.1 But du document**

Ce document est destiné à toute personne souhaitant réaliser un système de gestion pour un centre de santé de A à Z que vous soyez débutant ou aguerri. Il est également fait pour présenter le résultat de notre travail et les méthodes employés pour réaliser notre système de gestion des patients dans un centre de santé afin de permettre à nos camarades de pouvoir reproduire ce projet de façon ludique et efficace.

**1.2 Portée du projet**

Ce projet est une application console C++ permettant la gestion des informations personnels des patients (nom, prénom, âge, sexe, antécédents médicaux, adresse, numéro de téléphone, numéro de dossier, groupe sanguin, raison de l’hospitalisation), la gestion des informations liées à leur prise en charge (consultation, symptômes, traitement prescrit, diagnostique, résultat d’examen, ordonnance électronique, envoie des rappels aux patients) dans un centre de santé. Elle permet également au personnel médical de mieux gérer les admissions, la facturation des actes médicaux, et l’impression des documents médicaux ou justificatifs.

**1.3 Définitions, Acronymes et Abréviations**

Console :

Système de gestion des patients :

**1.4 Références**

Spécifications IEEE 830

Documentation interne du centre de santé (procédures administratives et médicales)

**1.5 Vue d’ensemble du Reste du document**

Ce document présente les spécifications fonctionnelles et techniques du système, les exigences de maintenance et de formation, notre plan de déploiement, les contraintes liées au développement et les perspectives d’évolution.

**2. Description Générale**

**2.1 Contexte du produit**

Ce système est destiné à un centre de santé et permet une prise en charge plus rapide des patients faciliter par l’accessibilité de leur carnet de santé électronique et un meilleur suivi de ceux-ci. Il est compatible avec n’importe quel environnement de développement dont Windows.

**2.2 Besoins des utilisateurs et Objectifs du projet**

Ce projet vise à améliorer l’efficacité de la gestion des informations des patients dans une structure de santé. Elle permet de mieux gérer les consultations, l’émission de facture, l’organisation des rendez-vous médicaux. Elle permet également d’alléger le travail du personnel médical en gérant les rappels aux patients, d’avoir accès plus facilement aux infos liées aux consultations des patients et d’ainsi mettre à jour leur dossier médicale plus rapidement.

**2.3 Caractéristiques du produit**

* A l’ouverture du système le menu permettra de se connecter dans celui-ci en tant que patient, médecin ou corps médical (en fonction du type de connexion choisit vous pourrez accéder aux fonctionnalités propres à cet état).
  + Lorsqu’on se connecte en tant que patient, on accède grâce à notre nom à notre dossier médical qui sera affiché avec nos différentes informations et rappels.
  + Lorsqu’on se connecte en tant que médecin, en entrant notre nom on accède à la liste des patients que nous devons consulter, nous pouvons inscrire nos diagnostiques, traitements dans son dossier médical, on peut également accéder à la liste des patients que nous avons déjà eu à consulter et pouvoir modifier leurs informations dans le cadre d’une deuxième consultation par exemple.
  + Lorsqu’on se connecte en tant que personnel administratif, on peut soit ajouter un patient (on lui crée un dossier médical avec toutes ces informations comme le nom, le prénom, l’âge, l’adresse, les antécédents médicaux et la soit le supprimer, envoyer des rappels aux patients, gérer les rendez-vous médicaux, gérer les factures, les ordonnances.
* Stockage local des données dans des fichiers

**2.4 Contraintes**

Les contraintes liées à ce projet sont :

* Console uniquement : l’interface graphique n’est pas optimisée et ne permet pas de revenir en arrière après une action
* C++ : prend plus de temps à mettre en place qu’avec des Framework.
* Sécurité : Le code source accessible peut poser des problèmes de sécurité
* Disponibilité 24h/24 et 7j/7

**2.5 Hypothèses et Dépendances**

* Aucune connexion n’est requise pour utiliser le système de gestion
* L’utilisation de ce système est simple et intuitive
* Ce projet est conçu avec l’hypothèse d’une commercialisation future

**3. Exigences Spécifiques**

**3.1 Exigences fonctionnelles (ERF)**

ERF001 : Créer un dossier médical pour le patient

ERF002 : Afficher les informations d’un patient

ERF003 : Créer un historique de consultation

ERF004 : Gérer les rendez-vous médicaux

ERF005 : Gérer les rappels aux patients

ERF006 : Gérer les factures

ERF007 : Gérer les fichiers d’ordonnance

ERF008 : Ajouter et supprimer des patients

ERF009 : Consulter et modifier un dossier médical

**3.2 Exigences non fonctionnelles (ENRF)**

ENRF001 : Interface simple et intuitive

ERNF002 : Temps de chargement inférieur à 2s

ERNF003 : Compatibilité C++ standard

ERNF004 : Code modulaire et commentaires clairs

ERNF005 : Fiabilité sans crash

**3.3 Exigences d’Interface externe**

UI : Interface en ligne de commande uniquement, navigation par menu numéroté et saisie clavier

Compatible avec imprimantes texte (via export fichier texte).

Aucune interface externe.

**3.4 Autres exigences**

Export de données patients au format texte pour archivage manuel.

Fonction de recherche simple par identifiant du patient (matricule).

**4. Préparation à la livraison**

**4.1 Plan de déploiement**

L’application sera fournie sous forme de fichier exécutable sous Windows, accompagné des fichiers source. Installation simple par copie des fichiers.

**4.2 Exigences de formation**

Ce projet ne demande aucune formation spécifique.

**4.3 Exigences de documentation**

Le code sera commenté pour plus de compréhension.

Un fichier README sera fourni avec.

Ainsi qu’un diagramme UML pour accompagner le projet.

**4.4 Exigences de maintenance**

Ajout de nouvelles fonctionnalités possibles (sécurisation via des mots de passe, possibilité d’enregistrer les paramètres vitaux d’un patient, annuler des rendez-vous ou avancer un rendez-vous, Gérer la présence des médecins à l’hôpital).

Amélioration de l’interface graphique (grâce à Qt).

**5. Annexes**

* **Glossaire**
* UI : Interface Utilisateur
* C++ : langage de programmation
* ERF : Exigence Fonctionnelle
* ERNF : Exigence Non Fonctionnelle
* **Prototype**

Voir l’interface sur console.

* **Historique**

Version initiale du projet réalisée dans le cadre d’un projet étudiant.