Une image contenant capture d’écran, Symétrie, cercle, Caractère coloré

Description générée automatiquementUne image contenant texte, Police, Graphique, logo

Description générée automatiquement

***Informatique De Gestion***

***AEC – DEC***

***Projet de fin De Session – Gestion d’un hôpital***

***Cahier de l’étudiant***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Spécifications de l’évaluation :** | | | | | | |
| Type d’épreuve : | 1ere partie | Version: | A | Durée : | 1 | mois |

*Programmation Orientée Objet et Applications* – *Prof : Abdourahmane Diawara*

|  |  |  |  |
| --- | --- | --- | --- |
| **Identification de l’élève** | | | |
| Nom : |  | Groupe : | 421-422-424 |
| Prénom : |  | Date : | 2024-03-20 |
| Code étudiant : |  | Signature de l’étudiant : |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Espace réservé à l’enseignant** | | | | | Verdict | |
| Note finale : |  | /100 |  | | Succès |  |
| Seuil de réussite |  | 60/100 |  | | Échec |  |
| Signature de l’enseignant: | | | |  | | |

***PROJET Gestion d’un hopital EN C#***

***COURS : Programmation orientée Objet***

***PROF: ABDOURAHMANE DIAWARA***

|  |
| --- |
| ***Objectif*** |

Créer une application Windows Forms qui s’occupe de la ***gestion d'un hôpital,*** en POO avec le langage C#, sans Base de données.

|  |
| --- |
| ***Énoncé du projet*** |

Dans ce projet captivant, vous allez créer une application de gestion d'un hôpital en utilisant Windows Forms et le langage de programmation C#. L'objectif est de concevoir une interface conviviale et fonctionnelle qui permettra de gérer efficacement les patients, les médecins, les rendez-vous, les salles d'attente, etc.

Cette interface sera réalisée avec Windows Forms et le langage utilisé est C#

|  |
| --- |
| ***Cahier des charges*** |

* *Le patient*

Dans votre application de gestion d'hôpital, vous devez stocker différentes informations sur chaque patient pour les modéliser de manière adéquate. Voici quelques exemples d'informations que vous pourriez inclure :

1. Nom complet du patient

2. Date de naissance

3. Genre

4. Numéro de téléphone

5. Adresse

6. Numéro d'identification unique du patient (par exemple, un numéro de sécurité sociale ou un identifiant interne à l'hôpital)

7. Antécédents médicaux (allergies, maladies chroniques, interventions chirurgicales passées, etc.)

8. Médecin traitant du patient

9. Liste des rendez-vous passés et futurs

10. Détails des traitements médicaux en cours

Ces informations vous permettront de gérer efficacement les dossiers des patients, de planifier les rendez-vous, de suivre leur historique médical et de faciliter la communication avec les médecins traitants.

* *Le médecin*

Pour représenter les médecins dans votre application, nous devez prendre en compte plusieurs caractéristiques importantes. Voici quelques exemples :

1. Nom complet du médecin

2. Spécialité médicale

3. Numéro de téléphone professionnel

4. Adresse de consultation

5. Emploi du temps (jours et heures de disponibilité)

6. Liste des patients actuellement suivis

7. Informations sur les qualifications et les certifications du médecin

8. Historique des rendez-vous et des prescriptions médicales

Ces informations vous permettront de gérer efficacement les médecins et de faciliter la prise de rendez-vous pour les patients.

* *Rendez-vous*

Pour gérer les rendez-vous entre les patients et les médecins, vous devrez stocker les informations importantes pour chaque rendez-vous.

1. Date et heure du rendez-vous

2. Médecin responsable du rendez-vous

3. Patient concerné par le rendez-vous

4. Motif ou raison du rendez-vous

5. Statut du rendez-vous (confirmé, annulé, en attente)

6. Durée estimée du rendez-vous

7. Éventuels rappels ou notifications liés au rendez-vous

Ces informations vous aideront à organiser les rendez-vous, à suivre les disponibilités des médecins et à communiquer efficacement avec les patients.

* *Salle d’attente*

Vous pouvez organiser les salles d'attente en mettant en place les fonctionnalités suivantes pour gérer l'affluence :

1. Système de réservation de créneaux horaires : Les patients pourront réserver un créneau spécifique pour éviter une affluence excessive dans la salle d'attente.

2. Notifications et rappels : Les patients recevront des notifications et des rappels avant leur rendez-vous pour les informer de l'heure exacte et réduire le temps d'attente.

3. Statut en temps réel : Les patients pourront voir en temps réel l'état de leur rendez-vous, s'ils sont en avance, à l'heure ou en retard. Cela permettra de mieux gérer l'affluence.

4. Estimation du temps d'attente : mettre en place une fonctionnalité qui estime le temps d'attente moyen dans la salle d'attente, ce qui permettra aux patients de mieux planifier leur arrivée.

5. Gestion des priorités : implémenter un système de priorités pour les patients ayant des besoins médicaux urgents, afin de s'assurer qu'ils soient pris en charge rapidement.

Ces fonctionnalités aideront à gérer l'affluence dans les salles d'attente et à améliorer l'expérience des patients.

|  |
| --- |
| ***Implémentations*** |

Voici comment vous pouvez implémenter chaque fonctionnalité de votre application de gestion d'hôpital :

1. Système de réservation de créneaux horaires :

* Créer une interface utilisateur avec un calendrier pour sélectionner la date et l'heure du rendez-vous.
* Implémenter une classe "RendezVous" qui stocke les informations du rendez-vous, telles que la date, l'heure, le médecin et le patient.
* Utiliser des listes pour gérer les rendez-vous réservés et disponibles.

2. Notifications et rappels :

* Utiliser la fonctionnalité de messagerie de Windows Forms pour envoyer des notifications et des rappels aux patients avant leur rendez-vous.
* Créer une classe "Notification" qui stocke les informations de la notification, comme le destinataire, le contenu et l'heure d'envoi.

3. Statut en temps réel :

* Afficher le statut du rendez-vous dans une interface utilisateur en utilisant des labels ou des icônes pour indiquer si le patient est en avance, à l'heure ou en retard.
* Mettre à jour le statut en fonction de l'heure actuelle et de l'heure du rendez-vous.

4. Estimation du temps d'attente :

* Calculer le temps d'attente moyen en fonction des rendez-vous précédents à l'aide de la classe "RendezVous".
* Afficher cette estimation dans l'interface utilisateur pour que les patients puissent mieux planifier leur arrivée.

5. Gestion des priorités :

* Ajouter une fonctionnalité permettant aux médecins d'attribuer des priorités aux patients en fonction de leurs besoins médicaux urgents.
* Utiliser une classe "Patient" avec une propriété de priorité pour gérer cette fonctionnalité.

|  |
| --- |
| ***Résumé*** |

Pour implémenter votre application, vous pouvez envisager les classes suivantes entre autres et les liens entre elles :

1. Classe "Patient" :

* Cette classe représente les informations d'un patient, telles que le nom, l'âge, le numéro de téléphone, etc.
* Elle peut être liée à la classe "RendezVous" pour associer les rendez-vous à un patient spécifique.

2. Classe "Medecin" :

* Cette classe représente les informations d'un médecin, telles que le nom, la spécialité, etc.
* Elle peut être liée à la classe "RendezVous" pour associer les rendez-vous à un médecin spécifique.

3. Classe "RendezVous" :

* Cette classe stocke les informations d'un rendez-vous, comme la date, l'heure, le patient, le médecin, etc.
* Elle peut être liée à la classe "Patient" et à la classe "Medecin" pour établir la relation entre les rendez-vous, les patients et les médecins.

4. Classe "Notification" :

* Cette classe représente les informations d'une notification, telles que le contenu, le destinataire, l'heure d'envoi, etc.
* Elle peut être liée à la classe "Patient" pour envoyer des notifications aux patients concernés.

5. Classe "InterfaceUtilisateur" :

* Cette classe est responsable de l'interface utilisateur de l'application, où les utilisateurs peuvent interagir avec les différentes fonctionnalités.
* Elle peut utiliser des instances des autres classes pour afficher les informations, gérer les événements utilisateur, etc.

***Questions***

**PARTIE 1 :**

1. Faites le croquis du diagramme de classe et montrez les liaisons qui existent entre vos classes. Vous pouvez utiliser ce lien pour télécharger le logiciel UML.
   1. <http://dia-installer.de/index.html.fr>

**PARTIE 2 :**

Implémentez tout le projet avec Windows Form