Projet de Programmation

**Intitulé : Système de gestion des centres d’isolement**

Equipe 6 : Nada Ben Ali

Yasmine Jerbi

Yasmine Ben Dhiab

Wejden Ben Romdhane

**Encadrée Par : Mme Raoudha Chebil**

Lien Github : https://github.com/nada191/Sujet3Eq6

Année Universitaire : 2019-2020

**SOMMAIRE**

1. Introduction
2. Modélisation
3. Environnement de Développement
4. Réalisation du Projet
5. Répartition des taches
6. Développement et moyens techniques
7. Moyens techniques
8. Difficultés rencontrés
9. Références Consultés
10. Conclusion
11. **Introduction :**

Dans le Cadre du projet de programmation et dans le but de répondre aux besoins de notre pays pendant cette période Critique notre équipe était chargé d’élaborer un Système de Gestion des Centres d’isolement qui prennent en charge les personnes concernées par Le Covid-19 (Un Sujet Contact ou venant de l’étranger)

1. **Modélisation :**

|  |
| --- |
| **Classe Personne concernée :** |
| **Attributs** :  Private Calendar date\_arrivee;  Private String nom, prénom;  Private long num\_cin;  Private int Etat\_santé;  Private Calendar date\_depart; |
| **Méthodes :**  Constructeur  Getters et setters  Public boolean fin\_confinement ()  Public void afficher\_personne () |

|  |
| --- |
| **Classe Centre\_isolement :** |
| **Attributs** :  Private int num\_ref  //numéro de référence  Private String Adresse ;  Private String gouv\_appartenance ;  Private int max  //la capacité maximale du centre  Private int nb\_actuel  Private int nb\_ac  //le nombre de personnes acceuillis dans le centre  Private int nb\_malades  Private Vector <Personne\_concernee> personnes |
| **Méthodes :**  Constructeur  Getters et setters  Public boolean fin\_confinement ()  Public void afficher\_personne ()  Public Personne\_concernee get\_p (int i)  //afficher les informations à propos d'une personne donnée  Public void changer (long cin, int e)  //mettre à jour l'état d'une personne donnée  Public int état (long cin)  //afficher l'état d'une personne donnée  Public void afficher\_departs ()  /\*affiche les personnes saines du centre ayant terminées leur période de confinement \*/  Public void hospitalisés ()  // affiche les personnes dont la situation est grave  Public boolean existe (long cin)  //vérifier l'existence d'une personne dans le centre |

|  |
| --- |
| **Classe Gouvernorat :** |
| **Attributs** :  Private String nom;  Private int nb\_ac;  //nombre de centres par gouvernorat;  private Vector <Centre\_isolement> centres ; |
| **Méthodes :**  Constructeur  Public void Ajouter\_centre (Centre\_isolement c)  Public void afficher\_centres ()  Public int personnes\_contamines ()  //donne le pourcentage de contamination dans le gouvernorat  Public Int personne\_contaminés (int ref)  //donne le pourcentage de contamination dans un centre donné  Public void supprimer (long cin, int ref)  Public void changer (long cin, int e, int ref)  //permet de mettre à jours les états des patients  Public void afficher\_departs (int ref)  /\*affiche les personnes saines d'un centre donné ayant terminé leur période de confinement \*/  Public void hospitalisés (int ref)  /\* affiche les personnes dont la situation est grave pour guider l'utilisateur vers la suppression manuelle \*/  Public int [][] disponible ()  /\*donne une matrice dont la première ligne correspond aux références et la deuxième ligne aux nombres de chambres disponibles  Public void ajouter (Personne\_concernee p)  Boolean existe\_centre\_cin (int ref, long cin)  Boolean existe\_centre (int ref)  /\*les deux dernières méthodes permettent de vérifier l'existence des personnes et des centres pour mieux guider l'utilisateur en cas des entrées erronées |

|  |
| --- |
| **Classe Manipulation :** |
| **Attributs** :  Private Vector <Gouvernorat> pays ;  Private float [][] distances  //matrice des distances entre les différents gouvernorats du pays |
| **Méthodes :**  Constructeur  Public void afficher (int ref)  //affiche les personnes sains d'un centre donné ayant terminés leur période de confinement  Public void afficher\_centres (String g)  Public void afficher\_personnes (int ref)  //affiche les personnes dans un centre donné  Public void Ajouter\_centre (String g)  Public void supprimer\_personne (long cin, int ref)  Public void statistiques\_gouv (String gouv)  //donne le pourcentage de contamination dans le gouvernorat  Public void statistiques\_centre (int ref)  //donne le pourcentage de contamination dans le gouvernorat associé au centre de référence ref  Public Vector<Gouvernorat> plus\_proche (String g)  //retourne un tableau trié des gouvernorats en tenant compte des distances (du plus proche au plus loin)  Public Void Demande (int nb, String gouv)  //permet l'affectation d'un nombre des patients d'un gouvernorat donné  Public void changer\_sante (long cin, int e, int ref)  //mettre à jour l'état d'une personne donnée dans un centre donné  Public void évaluation (int ref)  Fonctionnalité supplémentaire : permet à l'utilisateur de faire un bilan des évaluations des patients portant sur l'équipe médicale, la nourriture et surtout les mesures d'hygiènes afin de pouvoir prendre les mesures nécessaires pendant cette période critique  Public void hospitalisées (long ref)  // affiche les personnes dont la situation est grave dans un centre donné  Public void Demande\_docteur (String spec, String gouv)  Fonctionnalité supplémentaire : permet à l'utilisateur de visualiser les médecins spécialistes disponibles en cas d'absence d'un médecin ou plusieurs ... en effet cette méthode prend en compte les distances entre les gouvernorats. On a utilisé pour ceux-ci la méthode plus proche  Public boolean existe (String gouv)  //vérifie l'existence du gouvernorat donné  Public boolean existe (int ref)  //vérifie l'existence du centre donné  Public boolean existe\_centre\_cin (int ref, long cin)  //vérifie l'existence d'une personne donnée dans un centre donné |

1. **Environnement de Développement :**

Notre Equipe a choisi le langage Java pour développer le Projet. En effet Ce choix était basé sur la facilité de la manipulation des dates qui sont primordiales dans la conception de notre solution.

1. **Réalisation du Projet :**

1. Répartition du Travail :

* **Nada Ben Ali :**
  + Implémentation de la classe personne concernée
  + Méthode Demande de la classe ‘ manipulation ‘
  + Gestion des affichages
* **Yasmine Ben Dhiab :**
  + Implémentation de la classe Gouvernorat
  + Implémentation de la fonctionnalité Demande\_docteur
* **Wejden Ben Romdhane:**
  + Implémentation des méthodes manipulation autre que “ plus proche” et “Demande”
  + Gestion des affichages
* **Yasmine Jerbi :**
  + Implémentation de la classe centre\_isolement
  + Méthode Demande de la classe ‘ manipulation ‘
  + Méthode plus\_proche de la classe ‘ manipulation ‘
  + Implémentation de la fonctionnalité évaluation

2. Développement et moyens techniques :

A- Moyens techniques

Pour améliorer et répondre aux objectifs de notre système de gestion des centres d’isolement on a utilisé des fichiers d’initialisation :

* ‘Personnes.txt’ contenant les attributs des personnes concernées
* ‘Centres.txt’ contenant tous les centres du pays avec leurs attributs
* ‘Gouvernorats.txt’ contenant les gouvernorats el les distances en klm
* ‘Evaluation.txt’ Contenants les notes des personnes lors de leurs départs
* ‘Docteurs.txt’ contenant la liste des médecins disponibles des différents

Gouvernorats

B- Difficultés rencontrés

* Délicatesse des méthodes d’affectation des personnes concernées

C- Références Consultés

Durant notre travail nous avons eu juste besoin de consulter des références manipulant les bibliothèques prédéfinis et les tris des tableaux

1. **Conclusion :**

Ce projet s'est révélé plus enrichissant que nous ne l'avions pensé. En effet, il nous a permis de montrer nos compétences et nos connaissances acquises dans le module de POO et surtout de réaliser un travail en équipe : répartir les tâches, écouter les autres et partager les idées. Enfin, ce dernier nous a apporté une expérience dans le déroulement d'un projet qui va nous aider dans notre future carrière.