



**Département d'informatique**

**FACULTÉ DES SCIENCES**

Université du Québec à Montréal

## **Travail pratique 2**

### **Système de gestion des examens médicaux**

**Version 1**

Cours : Conception – INF5153

Session : Été 2024

Enseignant : El Hachemi Alikacem

Date : 23 juin 2024

## Objectif du projet

L'objectif de ce projet est de mettre en applications les principes de conception SOLID et GRASP, les patrons de conceptions et autres règles pour une conception et une implémentation de qualités d'un système de gestion des examens médicaux dans un centre de soins.

## Présentation

L'objectif de ce travail est la réalisation d'une application de gestion des examens médicaux pour un centre de soin. Les examens médicaux dont il s'agit sont des examens que le médecin prescrit et sont réalisés dans un laboratoire médical, tels qu'analyse de sang, radiographie rayon X.

Les principales tâches de l'application sont les suivantes :

- Enregistrer les examens médicaux prescrits par un médecin pour un patient,
- Pour chaque examen prescrit, faire la demande de RDV à un laboratoire médical
- Enregistrer les RDV obtenus
- Enregistrer les examens faits par chaque patient

L'application implémente également un mécanisme de notification afin de notifier les parties concernées, des événements qui les concernent.

Le centre de soin est constitué de plusieurs services et dans lequel travaillent plusieurs médecins. Cependant, chaque médecin est affecté à un seul service. Chaque service est caractérisé par un nom de service, un numéro de téléphone et un courriel. Un médecin est caractérisé par un matricule, son nom, son numéro de téléphone et son courriel.

Exemples de services : Cardiologie, Néphrologie, gastroentérologie,

Le patient est caractérisé par un code (code patient), son nom, son numéro de téléphone et son courriel.

Le fonctionnement typique du centre de soin est le suivant :

- Le médecin prescrit un ou plusieurs examens médicaux en indiquant éventuellement certains des paramètres
- Le centre de soin envoie des demandes de RDV à des laboratoires pour la réalisation de ces examens
- Les laboratoires donnent des RDV aux patients en réponse aux demandes de RDV
- Les patients font les examens médicaux
- Le laboratoire communique les résultats au centre de soin

Pour certaines actions, des notifications sont envoyées. Par exemple, lorsque le laboratoire donne un RDV à un patient, ce dernier reçoit une notification par courriel ou SMS.

Pour cette application, il y a principalement quatre fonctionnalités :

1. Enregistrement des prescriptions des examens pour un patient
2. Prise de RDV. Il s'agit d'envoyer des demandes de RDV aux laboratoires pour chaque examen prescrit
3. Traitement des réponses des laboratoires pour les demandes de RDV. Il s'agit d'enregistrer les RDV obtenus pour faire les examens médicaux
4. Traitement des résultats des examens envoyés par les laboratoires. Il s'agit principalement d'enregistrer les informations relatives à l'examen médical fait par le patient.

D'autres modules seront nécessaires tel que la gestion des notifications.

Comme la plupart des applications, notre application disposera d'une interface graphique pour les interactions avec les utilisateurs, on traitera la partie interface graphique en deuxième lieu.

### **Prescription des examens**

Un médecin peut prescrire un ou plusieurs examens médicaux pour un patient. Pour les besoins des soins offerts par le centre, la liste des examens est la suivante :

- Analyse de sang. Les paramètres de cette analyse sont : Base, FSC, HDL et TSH
- Radiographie rayon X. Concerne une partie du corps : Membre inférieure, Membre supérieure, Thorax, Crane et Poumon
- Radiographie IRM. Concerne une partie du corps : Cerveau, Moelle épinière et Hanche
- Endoscopie. Il y a plusieurs types : Œsophage, bronchique et Urologique
- Analyse d'urine : Les paramètres de cette analyse sont : Base, Ph, protéinurie, glycosurie et cétosurie
- Myélogramme
- Échographie. Peut concerner les organes suivants : Thyroïde, Cerveau et obstétricale

Il y a aussi des examens qui sont des batteries d'examens. Ce sont des examens qui sont composés d'examens. Ainsi, on peut considérer les examens non composés comme élémentaires.

Nous avons :

- Examen anémie : composé des examens « analyse de sang : FSC », myélogramme et « analyse d'urine : protéinurie et glycosurie »
- Examen thyroïde : composé des examens « échographie : Thyroïde », « examen anémie » et « analyse de sang : TSH ».

Noter que les noms des examens sont standardisés et connus dans le milieu médical.

Une prescription d'un examen médical est caractérisée par le nom de l'examen et la date de la prescription. On lui attribue un numéro unique.

Noter que la liste des examens médicaux (élémentaires et composés) est prédéfinie et peut évoluer.

Pour cette fonctionnalité, on peut supposer que l'application propose une liste d'examens et une liste d'options pour chaque examen sur un panneau (Interface utilisateur), le médecin clique sur les examens médicaux à prescrire et fait les choix des options correspondantes.

## Prise de RDV

Pour cette fonctionnalité, le module doit créer une demande de RDV pour chaque examen « élémentaire » (incluant ceux qui constituent les examens composés). Une demande de RDV est donc associée à un examen. Elle comporte un numéro unique qui lui est attribuée et une date de la demande.

Chaque demande est envoyée à un laboratoire capable de réaliser l'examen souhaité. On dispose donc d'une liste de laboratoires. Chaque laboratoire est identifié par un code laboratoire, un nom, l'adresse du laboratoire, un numéro de téléphone et un courriel. On indique également la liste des examens médicaux que le laboratoire peut faire.

La demande envoyée au laboratoire doit inclure le numéro de la demande du RDV, le code du patient et la description de l'examen médical.

La description de l'examen médical est une information associée à l'examen médical. Elle est composée du nom de l'examen et ses paramètres.

Une demande est envoyée à un seul laboratoire sachant que plusieurs laboratoires peuvent prendre en charge l'examen médical. On utilisera une stratégie simple de choix de laboratoire, par exemple un choix aléatoire.

### Indication :

Pour une vraie application, les laboratoires seraient des systèmes externes avec lesquels l'application échange. Pour la réalisation de notre travail, nous allons simuler les opérations d'un laboratoire. On développera donc un module (faisant partie de l'application) qui implémente ces opérations.

Attention : ce module échange des données avec le (ou les modules) de l'application, mais ne peut pas avoir accès aux objets de l'application. Il a accès qu'aux objets (à priori juste un objet) permettant ces échanges.

## Attribution de RDV

Lorsqu'un laboratoire reçoit une demande de RDV, on détermine une disponibilité pour l'examen demandé et retourne la réponse au centre de soin.

La réponse retournée est reçue sous la forme d'une chaîne de caractères ayant un format standard. La chaîne de caractères est composée de couples propriété-valeur, séparés par une virgule.

Exemple :

codePatient = Alie14325, numDemande = 453623, dateRDV = 15-07-2024, heureRDV = 14 :30

Dans notre application, l'opération d'attribution de RDV sera implémentée dans le module qui simule le laboratoire.

## Traitement des RDV

L'application reçoit les réponses des laboratoires aux demandes de RDV. Ces réponses sont traitées pour créer les RDV qui seront communiqués aux patients. Un tel RDV est associé à la demande qui a été envoyé au préalable.

Noter que le mécanisme de notification aura la charge de notifier le patient de son RDV.

Rappel : les réponses reçus sont des chaînes de caractères composées d'informations sur le RDV. Voir la section « Attribution de RDV ».

## Gestion des résultats des examens

Lorsque le patient fait son examen médical, le laboratoire prépare les résultats et les transmet au centre de soin.

Les informations reçues sont constituées du code du patient, le numéro du RDV, la date de l'examen et l'URI du document qui contient les résultats (radiographies, résultats d'analyse, etc.) sur le serveur de fichiers mis en place pour partager des documents.

La transmission des résultats est une opération qui sera réalisée par le module qui simule le laboratoire.

## Notification

À la suite d'un événement lors de certaines opérations du centre de soin, le module de gestion des notifications envoie une notification à l'entité qui « écoute » cet événement.

Les opérations qui génèrent un événement sont :

- Attribution-RDV : cet événement se produit à la réception de la réponse du laboratoire pour une demande de RDV. Les « écouteurs » qui doivent être notifiés sont le patient et le service (dans lequel le patient fait ces soins).
- Résultat-Examen : cet événement se produit à la réception des résultats du laboratoire. Les « écouteurs » qui doivent être notifiés sont le médecin et le service (dans lequel le patient fait ces soins).

La notification peut se faire de plusieurs manières. Pour la version actuelle, nous avons deux types de notifieurs :

- Un notifieur courriel qui implémente l'envoi par courriel d'un message à l'écouteur
- Un notifieur SMS qui implémente l'envoi par SMS d'un message à l'écouteur

La conception de ce module doit permettre la modification (ajout, suppression) dans la liste des évènements, des écouteurs et des notifieurs facilement.

**Note Importante :** le module de notification doit être générique et ne pas avoir d'association vers les classes du centre de soin. Indication : utilisez les interfaces Java

### **Données de l'application**

Pour notre application, nous avons besoin de représenter des données pour son fonctionnement. Il s'agit de :

- Définition des examens médicaux : enregistrement de la liste des examens médicaux (élémentaires et composés) pouvant être prescrits aux patients du centre de soin.
- Définition des évènements, des écouteurs (pour chaque évènement) et des notifieurs (pour chaque écouteur).  
Sachant que pour un évènement, il peut y avoir plusieurs notifieurs et chaque notifieur notifie un ou plusieurs écouteurs.

Développer le modèle de données permettant de stocker ces données et qui sera initialisé au démarrage de l'application. Votre développement devra permettre la modification (ajout et suppression) de façon simple.

### **Scénario d'exécution**

Élaborer des scénarios d'exécution permettant de mettre en actions les différentes fonctionnalités de l'application.

## Recommandations et indications

- Commencez à travailler sur ce projet dès maintenant et cela de manière continue afin de faire le point pendant les séances de cours (ou de laboratoires), cela permettra d'apporter les clarifications et adaptations nécessaires, si besoin.
- Un effort particulier doit être fournis dans la conception. Il faut mettre en application les patrons de conceptions et autres principes vus en cours. C'est le principal objectif de ce travail.
- Des exigences additionnelles ou des changements peuvent être éventuellement apportés dans la description du projet afin de favoriser l'utilisation des patrons et les principes de conception.
- Il est primordial d'utiliser **Visual Paradigm** pour la réalisation de vos modèles
- Réutiliser le logger que vous avez implémenté dans le TP1
- Il est nécessaire d'utiliser un dépôt Git pour le projet et de m'envoyer une invitation à votre dépôt.
- L'implication de tous les membres de l'équipe dans la réalisation (codage) est impérative

## Tâches

- Élaborer le (ou les) diagrammes de classes de l'application. Afin d'alléger les diagrammes, il n'est pas nécessaire d'y inclure les constructeurs, les setters et les getters
- Faire la conception et la réalisation de l'application de gestion des examens médicaux
- Identifier et utiliser les patrons de conception dans votre solution. Documenter l'utilisation de chaque patron de conception en indiquant :
  - o Le patron utilisé
  - o Les classes/Interfaces, etc. qui constituent le patron
  - o Courte justification et description du patron
- Implémenter les scénarios d'exécution de l'application

## Remise

- Remettre dans un zip :
  - o Le ou les diagrammes de classes
  - o Un document PDF qui documentent les patrons de conception utilisé
  - o Le projet IntelliJ (ou Eclipse) zippé.
- Remise dans Moodle (impérativement) au plus tard, le **14 juillet 2024, 23h59**.

Attention : la qualité du code et des diagrammes est importante. On peut perdre des points. Bien commenter le code, respecter les conventions de codage, **modularité**, etc.