



Projet de Modélisation Orientée Objet & Bases de Données

2015 - 2016

L3 Informatique

Soins et remboursement

Marinette Savonnet
Éric Leclercq
Joël Savelli

Département IEM

<http://depinfo.u-bourgogne.fr>

<http://ufrsciencestech.u-bourgogne.fr>

Université de Bourgogne

mail : `prenom.nom@u-bourgogne.fr`

1 Description du projet

On s'intéresse aux paiements de médicaments achetés dans une pharmacie par les usagers bénéficiaires de la sécurité sociale.

S'ils ont une mutuelle, alors on stocke leurs numéros d'adhérent à la mutuelle, les modes de remboursement. Le remboursement de la part complémentaire par la mutuelle dépend de la cotisation de chaque usager. Les droits de la mutuelle d'un usager peuvent être changés d'une année à l'autre, en fonction de sa cotisation.

Sur un médicament, les informations suivantes sont présentes : nom, classe thérapeutique, nom du fabricant, numéro du lot, date d'expiration, et taux de remboursement attribué par la sécurité sociale qui peut être changé dans le temps.

L'achat d'un médicament remboursable nécessite une ordonnance d'un médecin. Sur une ordonnance figurent les informations concernant le médecin et le patient, la date de consultation, et une liste de médicaments avec les prescriptions de traitements. D'après la liste de médicaments sur l'ordonnance, une préparatrice (ou préparateur) de la pharmacie cherche les produits et établit une facture. Chaque facture a une liste de médicaments avec leur quantité. Si une facture a des médicaments remboursables, alors on distingue la somme à payer par l'utilisateur lui-même, la somme à rembourser par la sécurité sociale et la somme à rembourser par la mutuelle. Un traitement prescrit sur une ordonnance est valable pour un nombre précis de fois, donc une ordonnance peut donner lieu à une ou plusieurs factures. Dans ce cas les copies de l'ordonnance et de la facture sont envoyées au centre de gestion de remboursement de l'utilisateur. Les remboursements effectifs doivent être suivis et vérifiés par la pharmacie.

Par ailleurs, on considère que les usagers bénéficiaires de la sécurité sociale doivent y contribuer pour chaque produit remboursable, en versant par exemple un euro, dans la limite de cinquante euros par an (franchise médicale).

Le site de l'assurance maladie vous propose différentes informations : <http://www.ameli.fr>

2 Travail attendu

Vous devez montrer à travers la réalisation du projet que vous maîtrisez d'une part les concepts de modélisation UML et d'autre part ceux des bases de données et les aspects techniques d'Oracle.

Le projet est à réaliser en binôme.

Ce projet donnera lieu à deux notes – une note sur l'analyse UML, une note sur l'implémentation en relationnel sous Oracle – différents rapports pour chaque UE sont à rendre.

2.1 Modélisation en UML

Le projet de l'U.E. MOO (Modélisation Orientée Objet) est constitué de deux parties : une étude bibliographique et une modélisation en UML.

Organisation du travail Vous devez respecter le planning suivant :

Jusqu'au TD de la semaine du 12 au 16 octobre 2015	Réflexion et recherche de documentation sur le sujet. Vous devez rendre à votre enseignant de TD, lors du TD, un rapport papier présentant votre étude bibliographique (environ 4 pages) et une ébauche des diagrammes use case et de classes.
Semaine du 30 novembre au 04 décembre 2015	Vous devez rendre à votre enseignant de TD le compte-rendu final de votre travail : reprise de l'étude bibliographique et modélisation en UML (format papier).

Étude bibliographique Votre étude bibliographique doit être structurée et rédigée en français correct. Vous pouvez citer des documents à condition que les parties citées apparaissent clairement dans votre dossier et que vos sources figurent dans la section bibliographie de votre dossier.

Le document que vous devez rendre doit comprendre un texte présentant les éléments d'information que vous avez trouvés sur le sujet. Ce texte doit ensuite être exploité pour déterminer les acteurs et les fonctionnalités qui permettront de réaliser le diagramme *Use Case* et les éléments qui relèvent du diagramme de classes.

Modèle UML Vous devez construire votre modèle UML comme indiqué ci-dessous :

Donner une description des fonctionnalités Vous devez partir des listes d'acteurs et de fonctionnalités proposées dans votre étude bibliographique (éventuellement complétées). Vous devez proposer un diagramme *Use Case* de premier niveau (avec des fonctionnalités générales) et détailler certaines de ces fonctionnalités par un diagramme de second niveau. Il est conseillé d'utiliser toutes les ressources d'UML pour le diagramme *Use Case* (inclusion et extension de fonctionnalités, généralisation d'acteurs). Votre dossier doit comprendre :

- une explication synthétique de vos diagrammes ;
- la description complète –sous forme de *diagramme de séquences*– d'au moins une fonctionnalité ;
- une conclusion qui établit un bilan en deux points : sur le travail mené pour construire le diagramme *Use Case* et sur les qualités/défauts de votre diagramme *Use Case*.

Donner une description structurelle Pour construire le diagramme de classes, vous devez :

- reprendre les informations liées au diagramme de classes dans votre étude bibliographique et les compléter en détaillant les principales informations nécessaires pour pouvoir traiter

chacune des fonctionnalités qui apparaît dans votre diagramme *Use Case*. Ces informations doivent être organisées en classes et relations entre classes ;

- vous devez ensuite définir de façon précise les relations entre classes (association, agrégation, composition, multiplicité, généralisation/spécialisation). Il faut dans votre commentaire, justifier les décisions prises et expliquer quels ont été les problèmes rencontrés et les hypothèses faites ;
- vous devez compléter ce diagramme par des *contraintes OCL* ;
- le dernier point de votre commentaire doit présenter un bilan en deux points : sur le travail mené pour construire le diagramme de classes et sur les qualités/défauts de votre diagramme.

Donner une description partielle du comportement du système Vous devez choisir un groupe de classes qui sont liées par leur comportement. Pour chacune des classes de ce groupe, vous devez donner toutes les opérations et attributs significatifs en justifiant par rapport aux fonctionnalités et à ce qui apparaît éventuellement dans votre étude bibliographique. Vous devez donner le *StateChart* de chacune de ces classes en justifiant le choix des états. Ce dernier doit montrer clairement les enchaînements possibles d'appels d'opérations sur les objets de la classe. Les événements reçus et envoyés doivent être justifiés. Vous devez enfin construire un *diagramme de communication* qui indique les événements envoyés et reçus. À partir de ce diagramme, vous devez vérifier que tous les événements envoyés et reçus par vos *StateCharts* sont cohérents.

La grille de notation (sur 40) est la suivante :

- 10 pour le travail bibliographique
- 25 pour la modélisation
- 2 pour la présentation et la qualité de rédaction du dossier
- 3 pour la cohérence du dossier et le recul que vous démontrez par rapport au sujet

2.2 Base de Données

Vous devez montrer à travers la réalisation du projet que vous maîtrisez les concepts des bases de données (modèle relationnel, formes normales, contraintes, langage de création et de manipulation de données, programmation PL/SQL, vues, etc.) et les aspects techniques plus spécifiques à Oracle (séquences, triggers, contrainte check, procédures et fonctions stockées, etc.).

- Choisir une partie cohérente du diagramme de classes et le traduire dans le modèle relationnel ;
- Implémenter votre schéma relationnel sous Oracle ;
- Fournir un jeu d'essai ;
- Établir un jeu d'une dizaine de requêtes traduisant un fonctionnement simplifié du système et les écrire en SQL. Ces requêtes devront reprendre l'ensemble des opérateurs SQL vus en cours ;
- Écrire des fonctionnalités en PL/SQL.

Contenu du rapport Votre rapport reprendra le diagramme *Use Case* et ses explications (tel que décrit dans le projet de Modélisation Objet) et le diagramme de classes et ses explications (tel que décrit dans le projet de Modélisation Objet). Puis, en utilisant la méthode vue en TD vous détaillerez le passage du diagramme de classes vers le modèle relationnel.

Vous devez ensuite expliquer comment vous avez implanté les fonctionnalités sous Oracle à l'aide de contraintes d'intégrité, de triggers, de programmes PL/SQL, etc. Le choix d'implanter telle partie du schéma et telles fonctionnalités sera justifié.

L'ensemble de vos scripts doit être fourni en annexe, **cet ensemble ne peut constituer la plus grande partie de votre rapport.**

Réalisation En terme de taille de réalisation votre schéma relationnel doit contenir au moins 10 tables, un jeu d'essai, un jeu d'une dizaine de requêtes SQL et 10 programmes PL/SQL incluant au moins trois triggers de taille raisonnable et des vues dont au moins une associée avec un trigger *instead of*.

Organisation du travail Vous devez respecter le planning suivant :

Semaine du 02 au 06 novembre 2015	Choix d'une partie cohérente de votre diagramme de classes, traduction dans le modèle relationnel Création des tables et du jeu de données
Semaine du 23 au 27 novembre 2015	Requêtes SQL
Semaine du 14 au 18 décembre 2015	Fonctionnalité(s) du diagramme use case implémentée(s) à l'aide de programmes PL/SQL : leur démonstration Remise du rapport papier final

Contraintes

- Vous devez réaliser votre projet dans les mêmes conditions que les TPs c'est-à-dire exclusivement sur le serveur Oracle de l'IEM que vous utilisez pendant les TPs.
- La démonstration s'effectuera dans la semaine du 14 au 18 décembre 2015, **dans les mêmes conditions que les TPs.**
- Lors de la démonstration vous rendrez votre rapport papier final.

Chaque rapport contiendra les difficultés rencontrées (par exemple le problème récurrent des tables en mutation) et les solutions mises en œuvre, une introduction et une conclusion.

La grille de notation (sur 20) est la suivante :

- 3 points pour le rapport
- 2 points pour la création du schéma relationnel (avec les contraintes d'intégrité référentielles) et du jeu d'essai
- 5 points pour les requêtes SQL
- 10 points pour les programmes PL/SQL et la démonstration