

INF1163 : Modélisation et conception orientée objet

Projet

Système de gestion d'une agence de location de véhicules

Travail en équipe (4 membres par équipe)

Le but de ce projet est d'appliquer les principes de l'analyse et de la conception orientées objet vus en classe, notamment l'approche unifiée et UML. L'implémentation se fera en Java avec la plateforme de développement Eclipse. Le projet vous donne aussi l'occasion d'expérimenter avec quelques outils logiciels performants utilisés dans un environnement de développement professionnel (Git, GitHub, SQLite).

La collaboration entre les membres de chaque équipe est indispensable afin de pouvoir respecter les délais impartis. Les étudiants doivent donc apprendre à travailler en équipe. Ils doivent notamment apprendre à se répartir les tâches équitablement, à développer leur sens de la communication et à synchroniser leur travail afin d'être productifs dans leurs efforts. Il faudra donc nommer un chef d'équipe pour coordonner le travail et veiller à la bonne progression du projet. Le rôle de chef d'équipe pourra être joué, à tour de rôle et pour une période déterminée à l'avance, par chacun des membres de l'équipe. Afin de gérer efficacement le développement de votre code source et de pouvoir rapidement revenir à une version qui fonctionne en cas de problème, il est fortement conseillé d'utiliser l'outil de gestion de version Git (<http://git-scm.com/>). Afin de permettre à plusieurs membres de l'équipe de développer en parallèle et d'intégrer automatiquement leurs différentes portions de code, il est fortement suggéré d'utiliser l'outil de gestion de contrôle GitHub (<https://github.com/>). GitHub est en fait la partie serveur pour l'outil Git. Git est gratuit et peut être installé sur votre PC. GitHub est un serveur sur le nuage qui requiert que vous ouvriez un compte (gratuit).

L'objectif final du projet consiste à développer un système logiciel de gestion pour une agence de location de véhicules. En raison des contraintes temporelles, on se limitera au développement d'une version simplifiée qui implémente principalement la logique d'affaires de l'agence de location de véhicules et qui offre des interfaces utilisateur simples. La composante base de données ne faisant pas partie des objectifs de ce cours, on pourra se contenter d'utiliser des fichiers Java ou Excel pour stocker les données relatives à la gestion de l'agence. Toutefois, pour les équipes qui le voudront, il est conseillé d'utiliser le système de gestion de bases de données simple et pratique suivant : SQLite (<http://www.sqlite.org/>). Il s'agit d'un SGDB léger et gratuit qui s'installe très facilement sur n'importe quel PC.

Brève description du système

Une agence de location de véhicules vous engage avec le mandat de développer un logiciel pour la gestion de ses opérations quotidiennes. Le système informatique doit avoir une interface utilisateur intuitive et être facilement portable sur différentes plateformes d'exécution.

L'agence loue différents types de véhicules, à la journée, à la semaine ou au mois. Les clients peuvent réserver à l'avance un type de véhicule particulier. Le coût de location d'un véhicule est constitué d'un prix de base auquel on ajoute différents suppléments représentant les différentes options choisies par les clients. Ces suppléments peuvent également représenter des frais de maintenance ou de réparations négligées par les clients.

Le prix de base de la location d'un véhicule dépend avant tout de sa catégorie. La facturation se fait par tranche de 24 h à partir de la signature du contrat de location. Les différentes catégories de véhicules avec leur tarif de location sont :

- 1) Classe économique : 39 \$ par jour ;
- 2) Classe moyenne : 49 \$ par jour ;
- 3) Classe confort : 59 \$ par jour ;
- 4) Classe luxe : 99 \$ par jour ;
- 5) Classe de véhicules utilitaires (fourgonnettes, pick-up, etc.) : 89 \$ par jour.

Un supplément de 17,30 \$ par jour est à prévoir à titre d'assurance contre les dégâts matériels au véhicule, à moins que le client ne soit couvert par sa propre police d'assurance.

Afin de représenter les frais d'usure encourus, deux types de forfait sont offerts :

- 1) Un supplément de 15 ¢ par kilomètre parcouru au delà des 400 premiers kilomètres;
- 2) Un supplément de 10,40 \$ par jour pour un forfait avec un kilométrage illimité.

Les clients retardataires se voient imposer un supplément de 10% du prix de la location quotidienne, par heure de retard jusqu'à concurrence du tarif quotidien.

Les taxes provinciales et fédérales s'appliquent à tous ces éléments.

Un supplément de 119 ¢ / litre est calculé si le réservoir d'essence n'est pas plein au retour.

Par ailleurs, une inspection de la carrosserie est faite au retour du véhicule. Si le client n'avait pas acheté la couverture d'assurance proposée par l'agence, le montant estimé des réparations à effectuer est ajouté à sa facture ; à charge pour lui de réclamer un remboursement auprès de sa propre compagnie d'assurances.

En toute circonstance, le client doit prouver son aptitude à conduire le véhicule en fournissant un permis de conduire valide. Une vérification de la classe de conduite sera nécessaire pour certains types de véhicule (ex. : grosses fourgonnettes). Une personne ne peut louer un véhicule que si elle est âgée de 25 ans et plus.

Avant de prendre possession du véhicule loué, le client doit effectuer un premier versement correspondant au tarif de base de la location. Trois types de paiement sont disponibles : paiement en espèces, paiement par carte de crédit et paiement par carte de débit. Le restant de la facture est payé à la remise du véhicule. Dans le cas d'un paiement en espèces ou par carte de débit, l'agence exige une caution en espèces de 250 \$ remboursable au retour du véhicule.

Tous les montants indiqués ci-dessus correspondent aux tarifs en usage lorsque vous avez été contactés pour mener à bien ce projet et doivent être considérés plutôt comme des exemples que comme des valeurs figées. **Le système doit donc pouvoir s'adapter à toutes modifications des différents montants ci-dessus mentionnés.**

Le système logiciel projeté doit permettre le fonctionnement habituel attendu d'un tel commerce. Pour toute autre information non fournie mais pertinente, vous êtes libres de faire un certain nombre d'**hypothèses raisonnables** que **vous devez clairement mentionner dans votre livrable final.**

Plan de travail

Ce plan de travail a pour principal objectif de vous guider en s'assurant que le travail progresse, dans les délais impartis, vers le but final.

À noter que deux livrables intermédiaires sont obligatoires et doivent être remis à échéance sous peine de pénalités (5% de la note du projet pour chaque livrable intermédiaire en retard ou non remis).

Itération 1 (2 semaines : du 26 octobre au 9 novembre) :

Identifier tous les cas d'utilisation appropriés et les représenter dans un diagramme de cas d'utilisation. Classer les cas utilisation identifiés par priorité et développer les deux cas jugés les plus prioritaires.

Proposer un modèle du domaine. Coder en Java et tester les classes conceptuelles retenues. Proposer un DSS pour chaque cas d'utilisation développé, et le cas échéant quelques contrats.

Faire une première esquisse à la main de l'interface utilisateur principale. Dans cette première itération, aucun code relatif à l'interface utilisateur ne devrait être développé.

Construire une première « base de données » des données importantes de l'agence de location pour des fins de tests.

Attention : livrable intermédiaire obligatoire à rendre le jeudi 2 novembre (minuit) : diagramme des cas d'utilisation, un des deux cas d'utilisation documenté au format détaillé, et l'esquisse de la principale interface utilisateur. Déposez dans Moodle votre livrable sous forme d'un **fichier unique**, compressé au besoin, et portant le nom : Equipe_X_Livrable1 (le X devra être substitué par le numéro assigné à votre équipe). **Tous vos textes et diagrammes doivent être incorporés dans un fichier unique Word ou PDF. Aucun diagramme au format mdl ou sms ne sera accepté.** Le but de me rendre ce livrable après juste une semaine c'est de me permettre de vous donner du feedback lors de notre séance de cours du 9 novembre afin que vous puissiez vous assurer que vous êtes sur la bonne voie à la fin de cette première itération.

Itération 2 (2 semaines : du 9 au 23 novembre) :

Développer deux ou trois autres cas d'utilisation (selon leur complexité). Ajuster au besoin le modèle du domaine. Proposer des DSS et le cas échéant des contrats, pour ces cas d'utilisation.

Pour les 2 cas d'utilisation développés lors de la première itération, proposer des diagrammes d'interaction. En déduire un premier diagramme de classes. Coder ces classes et les tester.

Faire des esquisses des autres interfaces utilisateur jugées nécessaires selon les cas d'utilisation développés. Commencer à les coder en utilisant, par exemple, la bibliothèque Swing de Java.

Attention : Livrable obligatoire (jeudi 23 novembre au plus tard minuit): modèle conceptuel et 2 diagrammes DSS. Tous vos textes et diagrammes doivent être incorporés dans un fichier unique Word ou PDF. **Aucun diagramme au format mdl ou sms ne sera accepté.** Déposez dans Moodle votre livrable sous forme d'un **fichier unique** (compressé au besoin) et portant le nom : Equipe_X_Livrable2.

Itération 3 (2 semaines : du 23 novembre au 7 décembre) :

Pour les cas d'utilisation développés lors de la seconde itération, proposer des diagrammes d'interaction. En déduire le diagramme de classes final.

Finaliser le code de toutes les classes retenues.

Finaliser toutes les interfaces utilisateur.

Tester toutes les fonctionnalités implémentées.

Il n'y a rien à rendre ici, mais lisez la suite.

Présentation orale des projets : jeudi 7 décembre en classe (état d'avancement et démo d'un prototype).

Livrable final dû le lundi 11 décembre à minuit au plus tard

Le livrable final du projet comprenant :

- Tous les artefacts UML adéquats et finaux qui justifient la solution proposée (au minimum : cas d'utilisation développés, DSS, contrats le cas échéant, diagrammes d'interaction, diagramme de classes).
- Un bref texte explicatif résumant vos choix de conception et liant ensemble tous les plans et documents de conception fournis.
- Le code source en Java, adéquatement documenté.
- Toutes les explications nécessaires pour permettre à une tierce personne de compiler, exécuter et utiliser votre prototype. Si vous avez utilisé des bibliothèques non standard, il faut le préciser et donner leur URL et toute autre explication nécessaire à leur intégration dans votre code.
- Une annexe qui synthétise la contribution de chaque membre de l'équipe au projet. Par exemple : « Jean a collaboré à l'écriture des cas d'utilisation 2 et 4, a développé entièrement le diagramme de séquence 9 et a codé 20% de l'application (ou a codé 2 des 3 interfaces utilisateur) », etc.

À l'exception du code source, tous les autres éléments du livrable final doivent être incorporés dans **un fichier unique au format Word ou PDF**. Votre texte (livrable final sauf code source) doit être tapé en utilisant un traitement de texte et vérifié en utilisant un correcteur orthographique.

Ce livrable final constitué (i) d'un fichier unique contenant votre texte et vos diagrammes et (ii) d'un répertoire contenant tous les fichiers de code source, doit être compressé dans un fichier qui portera obligatoirement le nom : **INF1163_Projet_Equipe_X.zip** et devra être déposé via Moodle.

Toute soumission qui ne se conformera pas à ces directives sera refusée.