Équipe 3

INF1163-01

Modélisation et conception orientée objet

Première itération

Cas d’utilisations et planification

Rédigé par:

Jonathan Fillion

Joel Girard

Marie-Hélène Tanguay-Bérubé

Jean-Michel Charbonneau

Université du Québec en Outaouais

31 octobre 2017

Décisions concernant l’implémentation du logiciel

Les DSS permettent d’identifier quelques fonctionnalités qui pourront être implémentés sous forme de méthodes.

Le système nécessite un login afin de pouvoir interagir avec celui-ci, pour des raisons de sécurité et de gestion.

La fonction **ouvrirSession(nom, mot de passe)** s’occupe de gérer la connection au système.

Une fois connecté, le système offre des options d’opérations. Chaque opération a sa méthode qui s’occupe de créer l’objet particulier et de sauvegarder les données temporelles ainsi que l’identifiant d’employé responsable de l’opération.

Parmis ces opérations, on retrouve :

* **créerUneLocation()**
* **retournerUnVéhicule()**
* **modifierParamètres()**
* **faireUneRéservation()**
* **ajouterVéhicule(véhicule) \*\*Cette classe nécessite seulement cette étape.**

Lors que le préposé effectue une opération concernant une location, celle-ci doit être associé à un client particulier. La fonction **associerClient()** permet d’effectuer cette tâche.

Pour le cas créerUneLocation(), une fois le client trouvé, le préposé entre les dates et le système détermine ce qui est disponible. Le préposé choisit ce qui est disponible en accord avec les désirs du client et finalise la création avec **finaliserLocation(véhicule, paramètres)**.

Pour le cas retournerUnVéhicule(), le commis sélectionne le client parmi une liste. Lorsque fait, le préposé choisit parmi une liste de location(s) active(s) du client. Pour ce faire, la fonction **terminerLocation(locationActive, paramsFin)** est utilisée, paramsFin représente le kilométrage final ainsi que l’inspection, etc.. . Le système envoi le solde restant et le commis peut collecter le paiement avec **faireUnPaiement(solde)**.

Pour le cas modifierParamètres(), le système envoi les anciens paramètres pour que l’interface les affichent. L'employé change ensuite les valeurs qu’il veut changer et fait appel à la fonction **sauvegarderNouveauParamètres(paramètres)**.

Pour la cas faireUneRéservation(), le procédé est le même que la location sauf qu’on ne se rend pas jusqu’à la fin. On arrête après avoir sélectionné le véhicule.

Pour se déconnecter, l’utilisateur utilise **fermerSession()**.

Identification des patrons de conception

En analysant le modèle du domaine, on peut cerner la présence de quelques patrons de conception.

Le patron expert en information est présent à quelques endroits :

* Location doit être experte en information concernant le paiement car elle connaît le nombre de véhicules loués ainsi que la durée de location et les modalités.
* Les locations sont contenus dans le RegistreLocation, RegistreLocation a accès à toutes les locations actives et les locations inactives, il pourra donc déterminer avec précision les disponibilités des véhicules.

Le patron de création :

* Magasin est créateur de véhicule et destructeur aussi. La liste de véhicule est contenu dans magasin.
* Location contient et créer paiement.
* Magasin contient les clients et doit donc les créer.
* RegistreLocation doit créer nouvelle location

Pour éviter le couplage inutile :

* RegistreLocation contient les locations et location contient paiement, ainsi, si une modification doit être apportée, RegistreLocation demande à la location particulière d’effectuer un changement à son objet paiement. On évite donc des situations de concurrence où un objet modifie quelque chose et qu’un autre objet en ignore les circonstances.

Le patron contrôleur sera appliqué :

* L’interface graphique envoit tous les événements à la classe Magasin et celui-ci décide des opérations à effectuer afin de répondre au besoins de l’utilisateur.

La classe Magasin doit être statique est permettra à l’interface utilisateur de communiquer via l’application. Magasin distribue les tâches aux différentes classes spécialisée de l’application.

Cas d’utilisations

**Cas “Créer une location”**

|  |  |
| --- | --- |
| 1. Un client demande une location. |  |
| 2. Le commis ouvre sa session employé. | 3. Le système connecte le commis à son compte. Affiche un menu d’options. |
| 4.Le commis sélectionne compte client. | 5. Le système offre des outils de recherche pour trouver le client. |
| 6. Au moyen d’outils, le commis retrouve le compte client. | 7. Le système offre un choix d’opérations concernant le client. |
| 8. Le commis sélectionne “Créer une location”. | 9. Le système demande de sélectionner la durée de location. |
| 10. Le commis sélectionne la durée de location. | 11. Selon la durée, le système offre une sélection de classes de véhicules. |
| 12. Le commis sélectionne la classe de transport. | 13. Selon la classe et la durée, le système affiche une sélection de véhicules. |
| 14. Le commis sélectionne le véhicule désiré par le client. | 15.Le système demande de sélectionner le choix d’assurance et le choix des frais de distance. |
| 16.Le commis sélectionne le besoin d’assurance et les frais de distance. | 17. Le système demande de confirmer le kilométrage. |
| 18. Le commis doit vérifier le kilométrage de départ. Il entre l’information. | 19. Le système demande une confirmation, annonce le prix ainsi que le montant du premier paiement. |
| 20. Si le client accepte, le commis confirme l’achat. | 21. Le système demande le premier paiement. |
| 21. Le commis effectue le paiement avec le client. (Voir cas d’utilisation Paiement). | 22. Le système sauvegarde l’information et retourne au menu du commis. |

\* En tout temps, le commis peut annuler la transaction en tout temps.

\* En tout temps, le commis peut suspendre la transaction et la reprendre au même endroit.

\* En tout temps, le commis peut transformer la location en une réservation.

6A. Dans le cas où le client n’existe pas,

1. Le commis sélectionne l’option “créer un compte client”.
2. Une fois le compte client créé, le système s’y connecte afin de continuer l’interaction.
3. Retour à l’étape 7.

11A. Dans le cas où il y a aucune disponibilité pour la durée sélectionné,

1. Le système affiche un message d’avertissement.
2. Le commis a le choix d’annuler la transaction ou de faire une réservation future.

11B. Dans le cas où certaines classes sont non disponibles,

1. Le système affiche les classes disponibles et les futures disponibilités des indisponibles.
2. Retour à l’étape 11.

12A. Le client sélectionne la classe transport lourd,

1. Le système vérifie si le client possède la classe nécessaire.
2. S’il ne possède pas la classe nécessaire, le système affiche un avertissement.

16A. Le client possède sa propre assurance,

1. Le commis sélectionne que le client possède sa propre assurance.
2. Le système vérifie si celle-ci est encore valide.
3. Si l’assurance n’a jamais été enregistrée, ou l’ancienne sauvegardée est invalide, le système demande de remplir l’information de la police d’assurance du client.
4. Continuation à 17.

**Modification des paramètres par le gérant**

|  |  |
| --- | --- |
| 1.Le superviseur veut modifier les paramètres |  |
| 2. Le patron se connecte à la plateforme de gestion. | 3. Le système offre le menu employé avec options de gestionnaire. |
| 4. Le superviseur sélectionne de modifier les paramètres. | 5. Le système offre une sélection de catégories de paramètres. |
| 6. Le superviseur sélectionne la catégorie à modifier. | 7. Le système affiche les valeurs de la catégorie ainsi que les valeurs courantes. |
| 8. Le patron modifie les valeurs courantes. |  |
| 9. Le patron sélectionne “sauvegarder” | 10. Le système sauvegarde les valeurs et retourne au menu des catégories (Étape 5). |
| 11. Le patron quitte sa session |  |

\* À tout moment, le superviseur peut annuler le processus de modification.

6A. Dans le cas où le superviseur sélectionne la mauvaise catégorie.

1- Le patron sélectionne la mauvaise catégorie

2- Le système affiche le menu suivant.

3- Le patron sélectionne le bouton de retour.

4- Le système affiche le menu précédent.

9A. Dans le cas où le superviseur entre une donnée erronée.

1- Le superviseur sauvegarde les donnés avec une erreur.

2- Le système avertit qu’une erreur est présente.

3- L’utilisateur fait la correction et soumet les modifications.

**Ajouter un véhicule**

|  |  |
| --- | --- |
| 1. Le patron veut ajouter un véhicule à la flotte. |  |
| 2. Le patron se connecte à la plateforme. | 3. Le système offre le menu de l’utilisateur. |
| 4. Le patron sélectionne l’option d’ajouter un véhicule. | 5. Le système affiche le menu d’ajout. |
|  | 6. Le système demande les informations du véhicule. |
| 7. Le patron entre les informations. |  |
| 8. Le patron sauvegarde l’information | 9. Le système présente l’information afin de la confirmer. |
| 10. Le patron confirme l’information. | 11. Le système sauvegarde l’information. |
|  | 12. Le système retourne le menu principal. |
| 13. Le patron sélectionne de quitter. |  |

\* En tout temps, le patron peut annuler l’ajout d’un véhicule.

4A. Mauvaise sélection de menu

1- Le patron sélectionne le mauvais menu.

2- Le système affiche le menu sélectionné.

3- La patron clique sur le bouton de retour.

4- Retour à #4.

8A. Mauvaise information comme entrée.

1- La patron a entré des informations non-conformes.

2- Le patron sélectionne “sauvegarder”.

3- Le système détecte une erreur.

4- Le patron modifie l’information erronée.

5- Le patron sauvegarde.

6- Continuation à #9.

**Réserver un véhicule**

|  |  |
| --- | --- |
| 1. Un client veut réserver un véhicule. |  |
| 2. Le commis entre ses informations pour se connecter. | 3. Le système vérifie le login. |
|  | 4. Le système affiche le menu d’options |
| 5. Le commis sélectionne “Connexion au compte client” | 6. Le système offre l’outil de recherche pour trouver le client. |
| 7. Le commis sélectionne le compte client | 8. Le système offre les options du compte client. |
| 9. Le commis sélectionne “Réserver un véhicule” | 10. Le système demande de sélectionner la durée de location. |
| 11. Le commis entre les dates de début et de fin de location. | 12. Selon la disponibilité, le système offre une sélection de classes de véhicule. |
| 13. Le commis sélectionne la classe de véhicule. | 14. Le système, toujours selon la disponibilité, offre une sélection de véhicule contenu dans la classe |
| 15. Le commis sélectionne le véhicule. | 16. Le système confirme la réservation et met la location en suspens. Il met le véhicule indisponible pour la durée sélectionnée. |
| 17. Le commis ferme le compte client. | 18. Le système affiche le menu employé. |
| 19. Le commis ferme sa session. |  |

\* En tout temps, l’employé peut annuler la réservation.

3A. Mauvaises informations de connection.

1- Le commis entre les mauvaises informations.

2- Le système constate une erreur dans les informations.

3- Le système affiche une erreur.

4- Retour à #2.

5A. Le client ne possède pas de compte.

1- L’employé sélectionne création d’un compte client.

2- Le système demande les informations du nouveau client.

3- L’employé entre les informations du client.

4- L’employé clique sauvegarder client.

5- Retour au #8.

12A. Il ne reste plus aucun véhicule pour ce lapse de temps.

1- Le système affiche une erreur indiquant la non disponibilité d’un véhicule.

2- Le système retourne à la sélection de la date.

3- Retour # 10.

Précision sur la reprise d’une réservation : Une réservation devrait être reprise comme étant une location lorsque le client se présente pour le véhicule et la continuation se fait comme une location.

**Cas 2 “Fin de location”**

|  |  |
| --- | --- |
| 1. Le client revient avec un véhicule. |  |
| 2. Le commis ouvre sa session employé. | 3. Le système connecte le commis à son compte. Affiche un menu d’options. |
| 4.Le commis sélectionne compte client. | 5. Le système offre des outils de recherche pour trouver le client. |
| 6. Au moyen d’outils, le commis retrouve le compte client. | 7. Le système offre un choix d’opérations concernant le client. |
| 8. Le commis sélectionne l’option “fin de location” | 9. Le système affiche les locations actives du client. |
| 10. Le commis sélectionne la location particulière qui doit être terminée. | 11. Le système affiche les détails de la location. |
|  | 12. Le système demande de confirmer la fin de location. |
| 13. Le commis confirme la fin de location. | 14. Le système demande le statut d’inspection, le kilométrage final ainsi que le niveau d’essence. |
| 15. Le commis vérifie les informations demandées |  |
| 16. Le commis rempli les informations demandées. | 17. Le système calcule le prix final. |
| 18. Le commis effectue le paiement avec le client. | 19. Le système remet le véhicule en disponibilité. |

\* En tout temps, le commis peut annuler le retour de la voiture.

\* En tout temps, le commis peut augmenter le temps de location si le client le désire.

10A. Le client veut annuler plusieurs locations,

1. Le commis sélectionne plusieurs locations.
2. Continuer à 11.

17A. Le client ramène le véhicule en retard,

1. Le système affiche un message d’avertissement avec les frais supplémentaires.
2. Le système ajoute les frais au total.
3. Retour à 18.

17B. Le client a abîmé le véhicule et n’a pas souscrit à l’assurance offert par le marchand,

1. Le commis constate des dommages.
2. Le système demande de remplir un rapport contenant description des dommages, photos et signature du client.
3. Le système met le statut de la facture à “à compléter” et le client aura des frais supplémentaires à payer dans le futur.
4. Retour à 18 pour payer la location.

17C. Le client n’a pas rempli le réservoir à essence,

1. Le commis constate que le réservoir n’est pas plein.
2. Le système met le statut de la facture à “à compléter”
3. Le commis devra remplir le véhicule et inscrire le nombre de litres.
4. Le montant sera facturé au client plus tard.
5. Retour à 18 pour terminer le paiement principal.

19A. Le véhicule nécessite des réparations,

1. Le responsable ou commis doit faire effectuer les réparations.
2. L’employé doit ensuite inscrire le montant à la facture du client fautif et numériser le reçu de la facture.
3. La facture est envoyée au client.
4. Le système remet le véhicule disponible et le retire de la liste “à compléter”.

Priorisation des cas d’utilisation selon l’ordre d’importance

L’ordre d’importance se caractérise par la complexité du cas et de la fréquence d’utilisation auquel celui-ci fera face lors de l’utilisation du système.

1. Louer un véhicule est le cas le plus important de l’application.
2. Retourner un véhicule est important pour le bon fonctionnement de l’application

Les deux cas ci-dessus sont complexes et seront utilisés fréquemment. Leur complexité est principalement dû au fait qu’ils seront les créateurs et modificateurs de la facturation. Une attention particulière doit être portée.

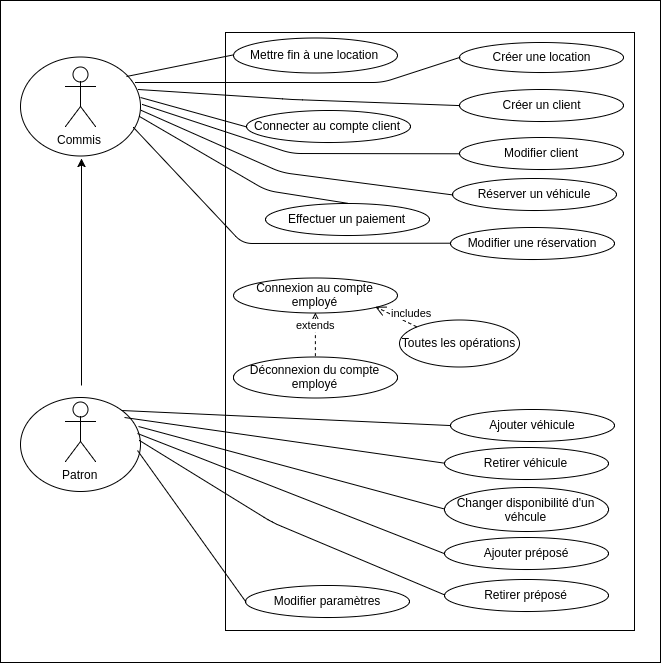
3. Réserver un véhicule.  
 4. Créer un compte client / Modifier compte client.

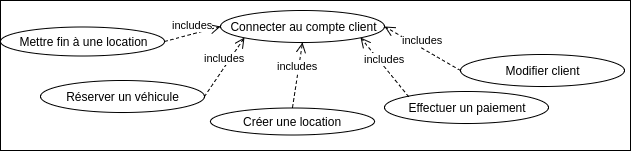
L’architecture repose sur le principe que chaque client possède un compte. Cela permettra de regrouper l’information selon un ordre pertinent.

5. Effectuer un paiement doit être adéquate et éviter toute erreur.

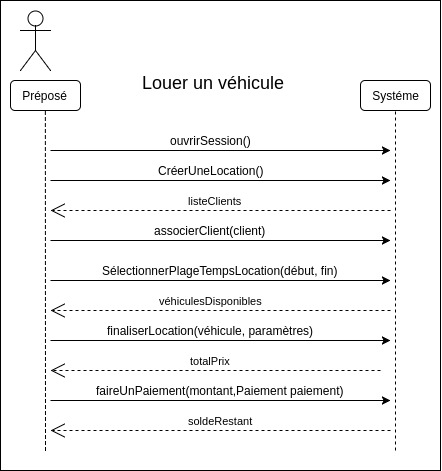
En ce qui concerne les autres cas, il est difficile de définir des priorités. Celles-ci sont variables. Il m’est possible de les énumérer dans un ordre quasi-ordonné.

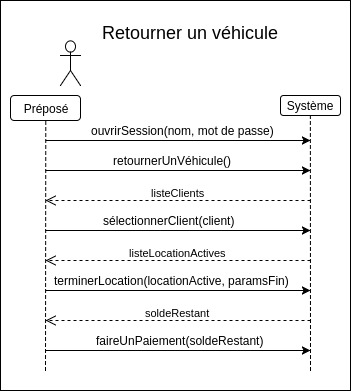
Connexion au compte employé, déconnexion du compte employé, ajouter véhicule, retirer véhicule, changer disponibilité d’un véhicule, ajouter préposé, retirer préposé, modifier paramètres (facturation, application).

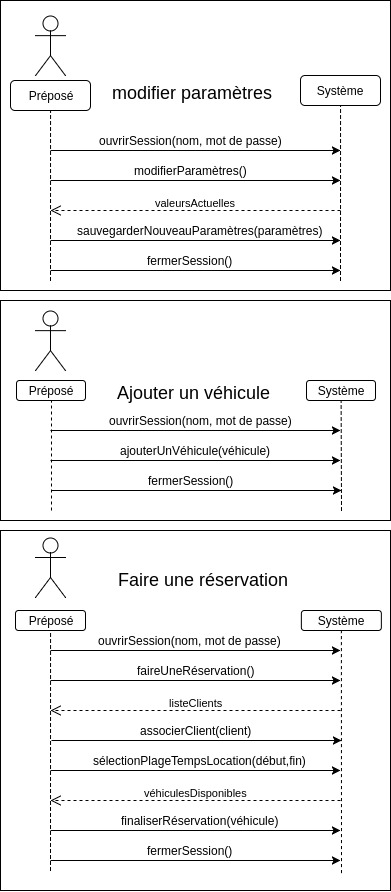


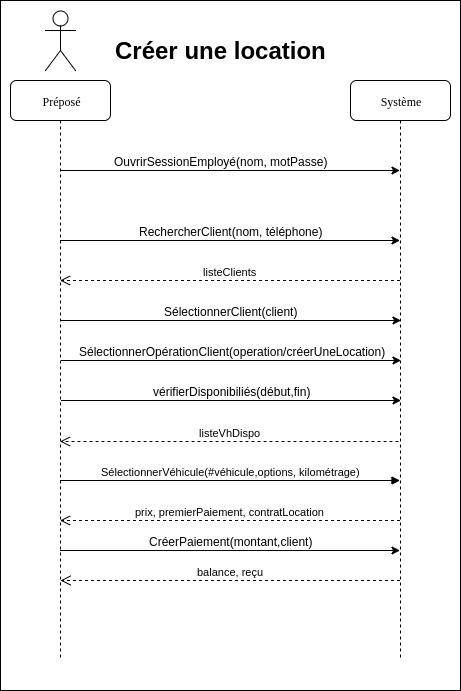


Diagrammes de séquence du système

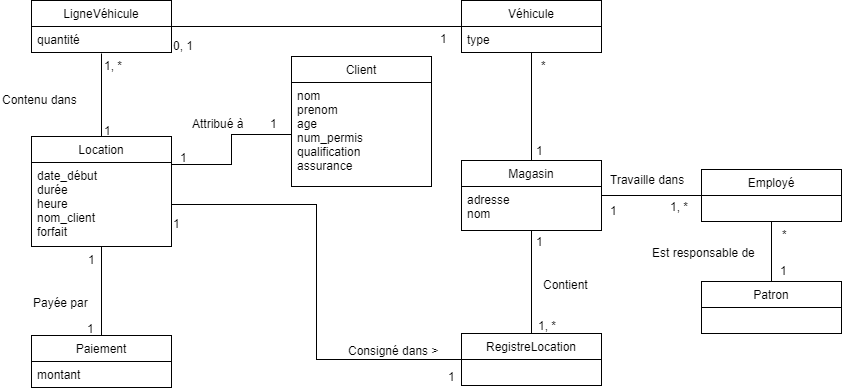








Modèle du domaine



Planification de l’interface graphique

