École de technologie supérieure  
Département de génie logiciel et des TI  
Cours : LOG210 - Analyse et conception de logiciels  
Date de création : 2009-07-12 Date de la dernière mise à jour : 2012-05-26

Système de gestion hôtelière  
Itération 1

|  |  |
| --- | --- |
| Noms | Destrempes, Marc-Andre Desharnais, Martin |
| Codes permanents | DESM14119009 DESM21099102 |
| Équipe |  |
| Cours | LOG210 |
| Trimestre | Été 2012 |
| Groupe | 01 |
| Enseignant | François Caron |
| Chargé de laboratoire | Julie-Isabelle Mailhot |
| Date | 8 juin 2012 |

## Barème de correction

|  |  |  |
| --- | --- | --- |
|  | Pondération | Résultats |
| Sous-système fonctionnel | 30 |  |
| Rapport | | |
| Introduction | 0 |  |
| Concepts retenus | 15 |  |
| Glossaire | 10 |  |
| Modèle du domaine | 30 |  |
| Descritption sommaire du Sous-système réalisé | 15 |  |
| Conclusion | 0 |  |
| Qualité du travail | 0 |  |
| Orthographe\* | 0 |  |
| Total | 100 |  |

\* Il n'y a pas de limite quant au nombre de points pouvant être déduits.

# Introduction

Le but de ce projet est de créer une application pour gérer les réservations et les séjours des clients d'un petit hôtel. Cependant, ladite application ne s'occupe pas de la facturation.

Le présent document contient le glossaire des divers concepts du domaine, le modèle du domaine représentant les différentes interactions entre lesdits concepts, ainsi qu'une description du sous-système qui est réalisé. Le contenu représente la ligne de pensée dirigeante de la conception du cas d'utilisation « CU01 : Noter une réservation ».

# Concepts retenus

La table suivante présentera la liste des concepts qui ont été retenus pour la conception du cas d'utilisation CU01. Les concepts ont chacun une catégorie et une explication qui lui est associée.

Table 1.1 - Concepts identifiés dans le cas d'utilisation CU01 - Noter une réservation

|  |  |  |
| --- | --- | --- |
| Catégories de Larman | Concepts | Explications |
| Rôle | Client | Le client est la personne qui à un besoin qui doit être comblé. |
| Rôle | Commis | Le commis est la personne responsable qui doit répondre aux besoins du client. |
| Transaction d'affaires | Réservation | La réservation comblera le besoin du client. |
| Item d'une transaction | Détail de réservation | Sers à indiquer de façon détaillée le contenu d'une réservation. |
| Ce avec quoi la transaction est enregistrée & Conteneur | Agenda | C'est à cet endroit que sont enregistrées les réservations. |
| Lieu & Catalogue | Hôtel | Bâtiment qui contient les chambres. |
| Produit d'une transaction | Chambres | Les chambres sont ce que le client choisira de réserver. |
| Description d'une chose | Catégorie de chambres | Permets de mieux définir une chambre. |

# Glossaire

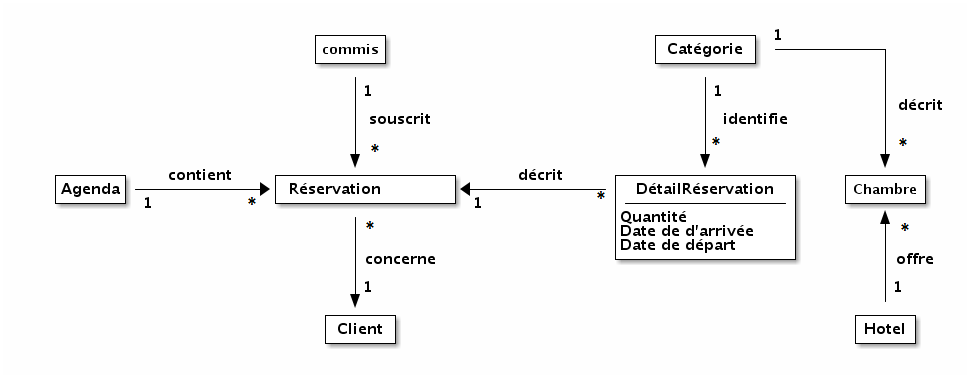
La table suivante présentera les divers concepts à l'aide d'une définition et des synonymes qui leur sont associés.

Table 1.2 - Glossaire

|  |  |  |
| --- | --- | --- |
| Concepts | Définitions | Synonymes |
| Agenda | Liste de toutes les réservations. | N/A |
| Catégorie de chambres | Classe dans laquelle ont réparti des éléments ayant des caractéristiques communes. | N/A |
| Chambre | Pièce d'une habitation où l'on dort. | N/A |
| Client | Personne qui achète un bien ou un service. | N/A |
| Commis | Employé subalterne dans un commerce. | N/A |
| Détail de réservation | Description détaillée d'une portion d'une réservation. | N/A |
| Hôtel | Entité offrant une liste de chambres classées par catégorie. | N/A |
| Réservation | Note précisant qu'un client à l'usage exclusif d'une chambre à un moment donné. | N/A |

# Modèle du domaine

Figure 1.1 - Modèle du domaine



Un hôtel offre des chambres qui ont chacune une catégorie et une même catégorie permet d'identifier plusieurs chambres.

Toutes les réservations sont conservées dans un agenda ce qui permet d'avoir un seul endroit ou toutes les réservations sont stockées. Une réservation est faite par un commis et concerne un seul client.

Une réservation est constituée de plusieurs détails où le client indique les catégories de chambres qu'il désire réserver ainsi que leurs quantités, les dates d'arrivée et de départ.

# Description sommaire du sous-système réalisé

### Liste des packages

#### hotel

Ce package contient les classes logicielles représentent la plupart des concepts du modèle du domaine. De plus, il contient les différentes classes représentant les systèmes qui servent à manipuler ces classes logicielles.

e.g. La classe « ReservationSystem » assume la responsabilité de contrôleur selon le principe GRASP. Il est utilisé afin de manipuler les instances de la classe « Reservation ».

#### hotel.gui

Ce package contient principalement les fenêtres de l'application et, dans une moindre mesure, un certain nombre de composants réutilisables qui ont trait aux interfaces graphiques utilisateurs.

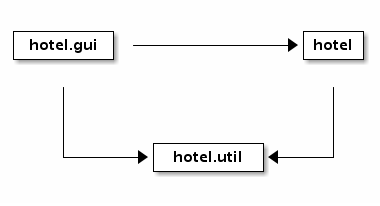
e.g. La classe « ReservationForm » représente la vue permettant de visualiser une réservation en cours d'ajout.

#### hotel.util

Ce package contient un agrégat de petits composants hétérogènes pouvant être utilisé par tous les packages.

e.g. La classe « ValidationException » représente une erreur de la validation de la logique d'affaires.

### Interactions entre les composants

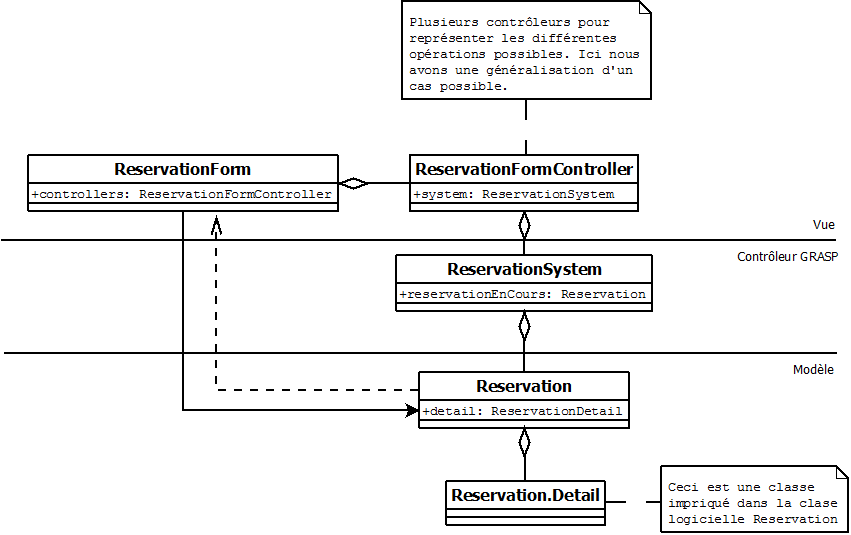


Le package « hotel.gui » utilise les packages « hotel » et « hotel.util ».

Le package « hotel » utilise le package « hotel.util ».

Le package « hotel.util » ne dépend de personne.

### Description de MVC



Lorsqu'une modification doit être effectuée, le contrôleur « ReservationFormController » est appelé par le patron observateur. Celui-ci utilise le contrôleur GRASP « ReservationSystem » qui sert de proxy entre lui et les classes du modèle du domaine. C'est ce proxy qui est responsable d'effectuer la manipulation demandée dans le modèle. Lorsque le modèle est mis à jour, il prévient la vue « ReservationForm » à l'aide du patron observateur et celle-ci utilise sa connaissance directe des classes du modèle pour mettre à jour ses informations.

# Conclusion

Le travail accompli permet à un client qui appelle à un hôtel d'effectuer une réservation pour une à plusieurs catégories de chambres à des dates différentes. Il est aussi possible d'annuler une réservation qui n'a pas encore été confirmée. Dans le cas où la réservation est annulée, celle-ci n'est pas enregistrée dans le système.

Pour ce faire, nous avons défini un modèle du domaine qui contient tous les concepts que nous avons définis. Par la suite, nous avons mis sur pied une architecture MVC afin d'implémenter ce système tout en conservant une séparation claire entre les sections chargées d'afficher les données (vue), modifier les données (contrôleur) et représenter les données (modèle).