

UE PROGRAMMATION SYSTÈME

DOSSIER D'ARCHITECTURE LOGICIELLE

École d'ingénieurs Cesi spécialité informatique

Alexandre Lecomte

Martin Hubert

Florian Fritsch

Corentin Gandossi

Contents

Introduction	2
Détail des Design Pattern utilisés	3
Factory	3
Singleton	4
Observer	5
Strategy	6
Bridge	7
Diagrammes d'activité	7
Maître d'hôtel	7
Chef de rang	8
Serveur	8
Commis de salle	8
Client	8
Chef de cuisine	9
Chef de partie / Cuisinier	9
Commis de cuisine	9
Plongeur	9
Diagrammes de cas d'utilisation	9
Salle	9
Cuisine	10
Logiciel de gestion	10
Diagrammes de séquence	11
Salle	11
Cuisine	11
Explication du MCD	11

INTRODUCTION

Une grande chaîne internationale de restaurants a, depuis quelques temps, des problèmes de gestion résultant en une baisse de la satisfaction client (file d'attentes trop importantes, manque de matériel entraînant des attentes en salle, désorganisation du service...).

Le directeur du restaurant a donc contacté notre équipe d'étudiants ingénieurs pour que nous lui réalisions une application de gestion et de supervision du fonctionnement du restaurant afin d'améliorer l'accueil du public, le remplissage des salles, la gestion des réservations et l'organisation du travail en cuisine.

Nous devons concevoir et réaliser cette application, et analyser la modélisation pour proposer des améliorations ainsi que le gain lié (quantifié).

L'application doit donc représenter la simulation du restaurant en permettant la visualisation de l'état de chaque acteur à tout moment. Les paramètres du programme permettent de choisir les options propre au fonctionnement du restaurant (nombre d'employés et de clients, stock d'ingrédients, temps de chaque tâche...) et à l'environnement de simulation (niveau d'accélération du temps, durée d'un shift...).

Une fois la simulation réalisée, nous devons ensuite mettre en place une interface de gestion qui recense toutes les informations concernant les performances du restaurant, ainsi que d'implémenter un système de préconisations en fonction des points de blocage identifié. Cela permet au directeur du restaurant de visualiser les points critiques de son activité et de prendre les décisions adéquates.

DÉTAIL DES DESIGN PATTERN UTILISÉS

Un Design Pattern est une solution à un problème récurrent dans la conception d'applications orientées objet. Un patron de conception décrit alors la solution éprouvée pour résoudre ce problème d'architecture de logiciel. Les Design Patterns sont indépendants des langages de programmation utilisés. Dans le cadre de la conception du logiciel de simulation de restaurant, les design patterns nous ont permis de concevoir une application performante et évolutive.

Pour répondre à diverses problématiques d'architecture; nous avons utilisé:

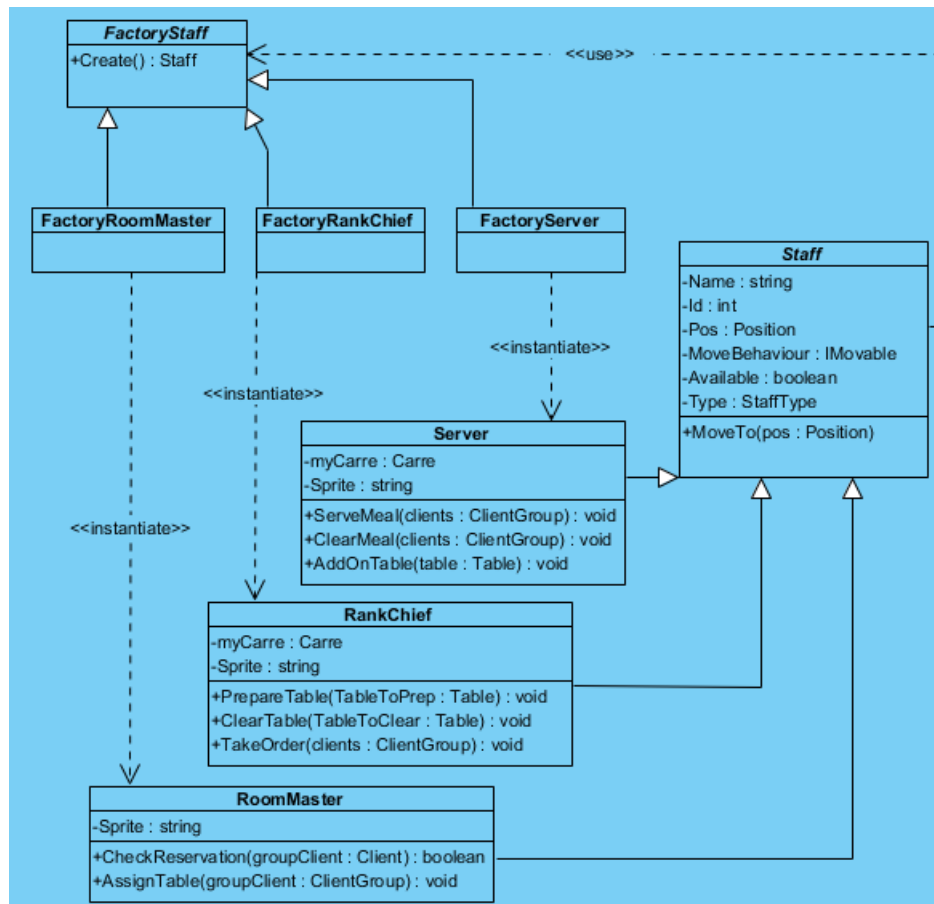
- Des design patterns de création:
 - Factory
 - Singleton
- Des design patterns de comportement:
 - Observateur
 - Strategy
- Un design pattern de structuration:
 - Bridge

Factory

Il est fréquent de devoir concevoir une classe qui va instancier différents types d'objets suivant un paramètre fourni. Par exemple une usine va fabriquer des produits en fonction du modèle qu'on lui indique.

On va donc déporter l'instanciation de nos différents types d'objets dans des usines (factory).

Dans notre application nous avons une équipe œuvrant dans la salle du restaurant (Serveur, maître d'hôtel...). Cette équipe est appelée Staff dans le diagramme UML ci-dessous. Le design pattern factory nous permet de déporter l'instanciation des différents membres du staff dans des classes dédiées.

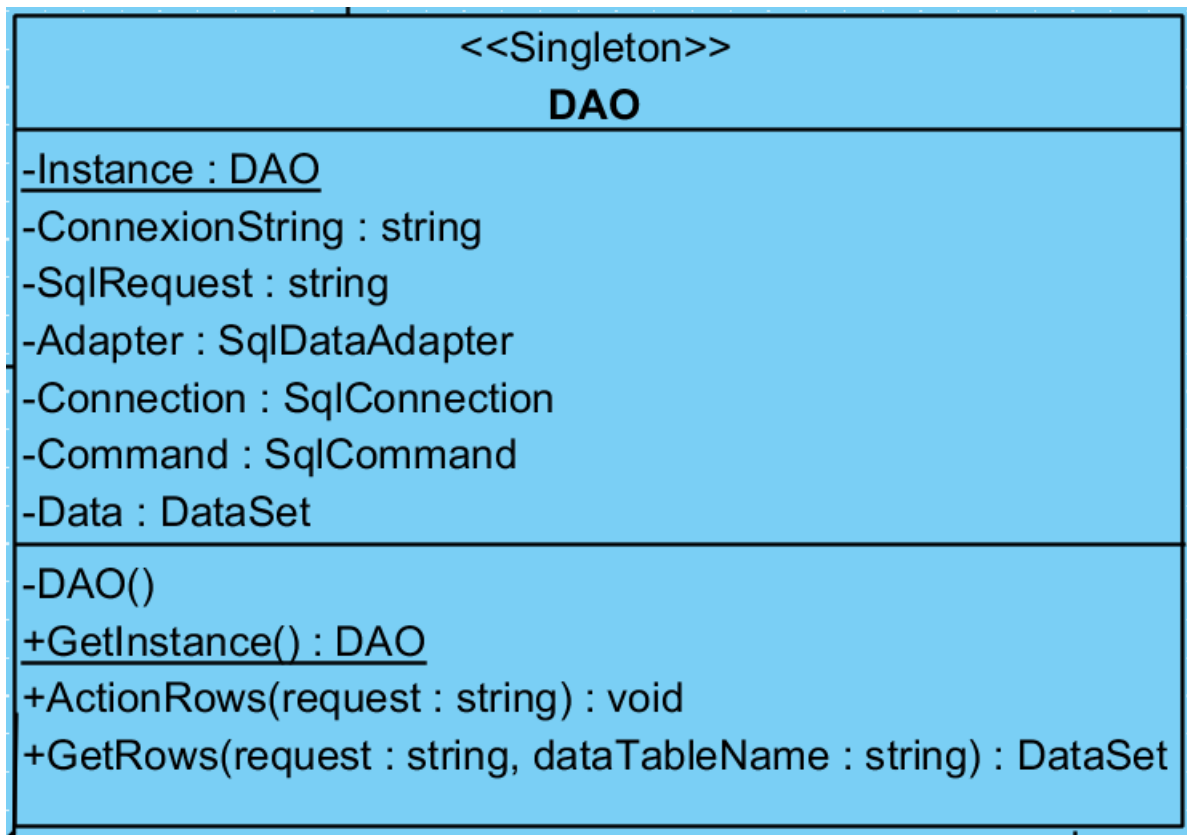


Singleton

Certaines applications possèdent des classes qui doivent être instanciées une seule et unique fois. C'est par exemple le cas d'une classe qui implémenterait un pilote pour un périphérique, ou encore un système de journalisation.

L'objectif du Singleton est donc de contrôler le nombre d'instance que peut retourner une classe. Dans son implémentation le design pattern singleton consiste à déclarer le constructeur de la classe avec une visibilité privée (pour contrôler en interne l'instanciation). On crée ensuite une méthode statique (souvent appelée `GetInstance`) pour obtenir l'instance de la classe. Si cette instance existe, elle est renvoyée sinon elle est instanciée.

Nous avons utilisé ce patron de conception pour l'accès à la base de données de l'application. On s'assurer donc qu'une seule connexion ne soit possible. Des connexions multiples à la base de données dans l'application pourrait porter atteinte à la cohérence des données stockées. Il est également utilisé par le module de log pour éviter les accès concurrents aux fichiers de logs et corrompre les données.

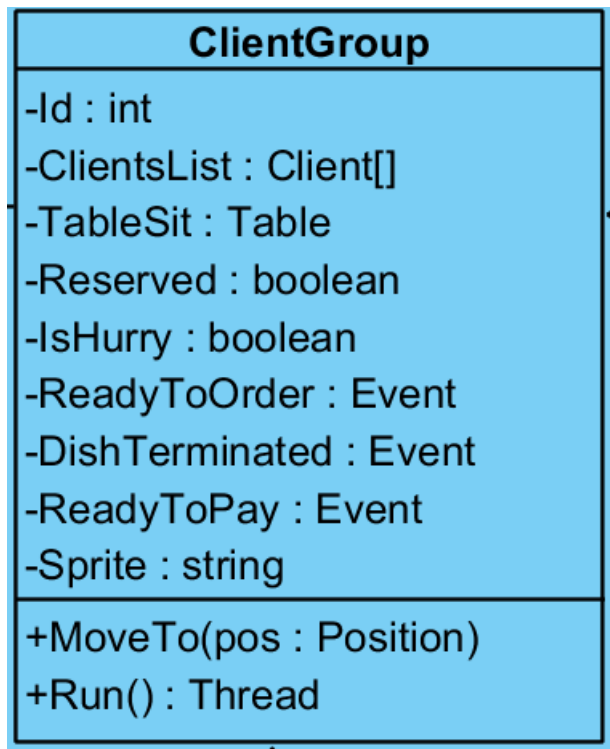


Observer

On trouve des classes possédant des attributs dont les valeurs changent régulièrement. De plus, un certain nombre de classes doit être tenu informé de l'évolution de ces valeurs.

Dans l'application le design pattern observer est notamment utilisé pour prévenir la vue (l'affichage) des changements réalisés dans le modèles (processus métiers). Nous l'utilisons également, pour que les clients puissent notifier le l'équipe de la salle de restauration lorsqu'ils ont besoin d'eux.

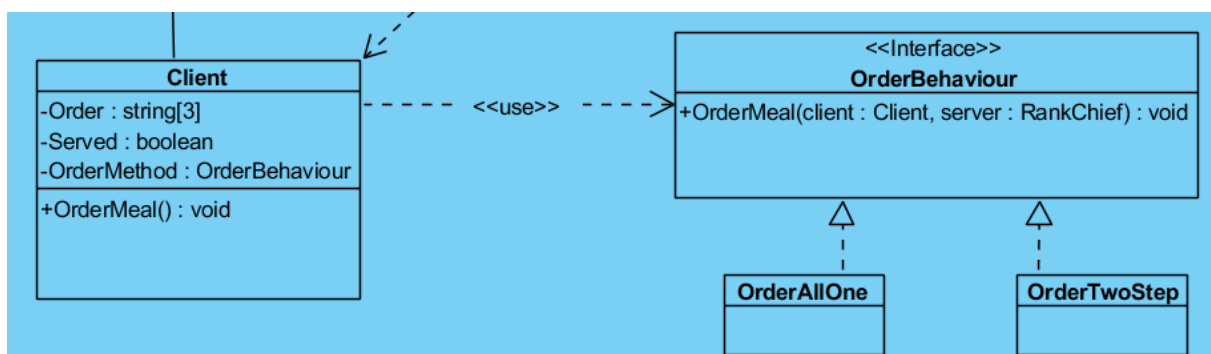
Dans le diagramme de classes, le design pattern observer est implémenter par des Events. Ces objets sont des types de delegates en C# .NET, langage qui sera utilisée dans l'application.



Strategy

Le design pattern stratégie permet de modifier le comportement d'une classe ou d'une de ses méthodes à durant l'exécution. Nous allons donc créer des objets représentant un ensemble de stratégies utilisées dans différents contextes.

La patron de conception stratégie nous permet de choisir la stratégie de commande d'un client. En effet, un client peut décider de commander l'ensemble de son repas en une seule fois ou alors en deux fois.

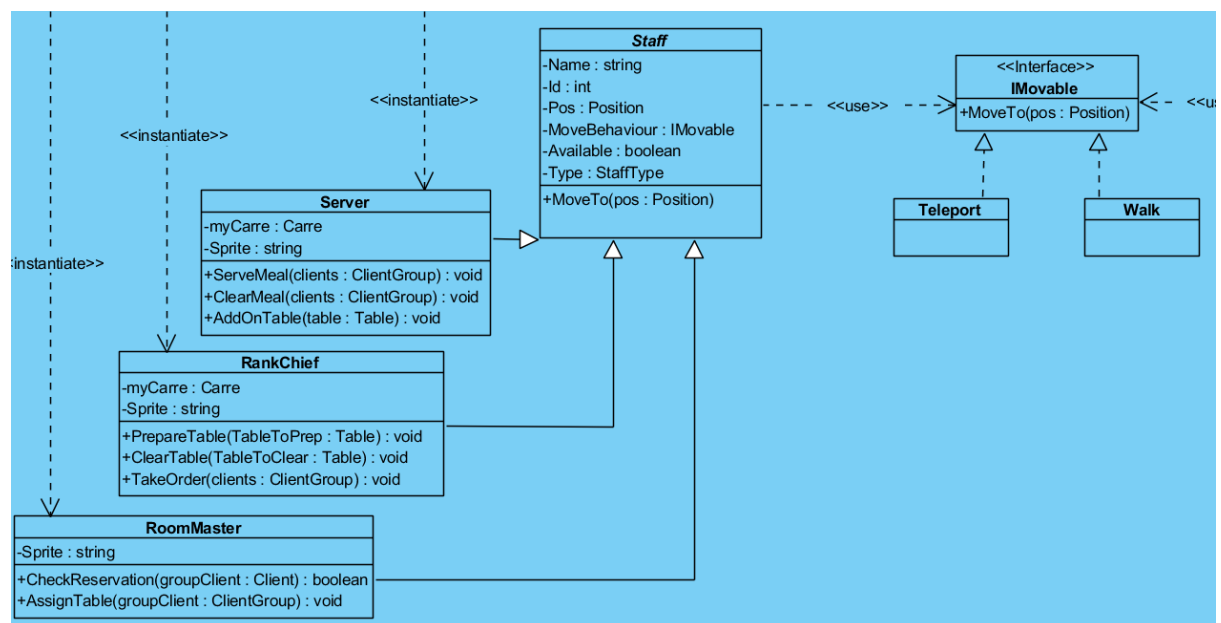


Bridge

Le design pattern bridge permet de découpler l'interface et son implémentation. Nous allons donc spécifier plusieurs interfaces différentes pour une seule et même interface.

L'interface agit comme un pont entre la classe qui l'implémente et les diverses implémentations possibles tout en les rendant indépendantes. Ces deux classes peuvent donc être modifiées sans porter atteinte à leur structure.

Nous utilisons ce patron de conception pour gérer les méthodes de déplacements des différents membres du staff.



DIAGRAMMES D'ACTIVITÉ

Maître d'hôtel

Le maître d'hôtel (MH) accueille les clients lorsqu'ils arrivent au restaurant et leur demande le nombre de personnes ainsi que s'ils ont une réservation. Le maître d'hôtel assigne ensuite une table dans un des rangs de la salle (si cela n'a pas déjà été fait au préalable lors de la réservation), celle-ci devant avoir une capacité supérieure ou égale au nombre de personnes du groupe. C'est au MH de prendre les décisions concernant les tables, son rôle étant de maximiser le nombre de couverts en fonction des places assises. Le MH a également la charge de faire payer les clients.

Chef de rang

Le rôle du chef de rang est d'installer les clients nouvellement arrivés et de prendre les commandes. Lorsqu'une table a été assignée à un groupe de clients, ceux-ci attendent l'arrivée du chef de rang afin d'être placés. Une fois assis, le chef de rang transmet les menus et patiente un moment avant de prendre la commande du groupe, celle-ci pouvant être prise en une ou deux fois. Une fois les choix faits, la commande est transmise dans son intégralité au chef de cuisine. Lorsque le chef de rang ne s'occupe pas de clients, il attend dans un coin de la salle. De plus, une fois qu'un groupe de clients part, et que la table a été débarrassée par le serveur, le chef de rang dresse

Serveur

Lorsque les clients ont pris leur commande, le serveur apporte sur la table le pain et l'eau (proportionnellement au nombre de clients). Il attend ensuite au comptoir des plats préparés et dès qu'une commande est prête il va la servir à la table correspondante (les plats de même type doivent être servis en même temps). Lors du repas, les serveurs restent à la disposition des clients pour répondre à leurs demandes. Lorsque les clients ont fini de manger, le serveur débarrasse la table et apporte la vaisselle sale au comptoir adéquat (pas plus de 5 en même temps).

Commis de salle

Le commis de salle est la personne qui assiste le serveur, pouvant assurer les tâches de service si les autres serveurs sont indisponibles.

Client

Le client est la personne qui bénéficie du service proposé par le restaurant. Il a la possibilité de réserver une table à l'avance ou d'arriver spontanément. Les clients se voient assigner une table par le maître d'hôtel et sont installés par le chef de rang. Ils commandent leur plats et sont servis par le serveur. Chaque client peut avoir des comportements et préférences différentes, et peut solliciter le serveur pendant le repas. A la fin du repas, le client paie l'addition au près du maître d'hôtel.

Chef de cuisine

Le chef de cuisine reçoit les commandes des chefs de rang et répartit leur réalisation entre les cuisiniers. Les commandes sont ordonnées de sorte à minimiser l'attente du client, en regroupant les commandes similaires, etc. De plus, si un plat ne peut être servi en raison du manque de fournitures, le chef de cuisine décide d'enlever le plat de la carte.

Chef de partie / Cuisinier

Le cuisinier (chef de partie) se charge de réaliser les plats donnés par le chef de cuisine. Pour cela, il peut décomposer les recettes en tâches élémentaires afin de les paralléliser. Lors de la réalisation d'une recette, il utilise des ustensiles, des machines et des ingrédients. Lors de l'utilisation d'un ustensile, il peut l'utiliser pour une autre tâche impliquant les mêmes ingrédients. Lorsqu'il a fini avec l'outil, il le libère afin qu'il soit nettoyé par le plongeur.

Commis de cuisine

Le commis de cuisine a la charge d'assister les cuisiniers en allant les chercher les ingrédients nécessaires dans le stock, en épluchant les fruits/légumes, et en apportant les plats prêts au comptoir adéquat.

Plongeur

Le rôle du plongeur est de remplir le lave-vaisselle à partir du comptoir des plats sales et de le vider lorsqu'il a fini. Le plongeur s'occupe également du lave-linge pour le remplir de nappes et serviettes sales. Il récupère aussi les ustensiles sales utilisés par les cuisiniers afin de les nettoyer pour les rendre à nouveau disponible.

DIAGRAMMES DE CAS D'UTILISATION

Salle

Lors de l'arrivée d'un groupe de clients, il est pris en charge par le maître d'hôtel qui leur assigne une table. Le chef de rang s'occupe ensuite de les placer, leur distribue la carte (en fonction de la disponibilité) et revient prendre les commandes. La commande est ensuite transmise à la cuisine pour la préparation. Pendant ce temps, le serveur apporte le pain et l'eau sur la table et se met en attente. À la fin de chaque plat, le serveur débrosse la table. A

tout moment, il peut répondre aux sollicitations du client. A la fin du repas, le règlement se déroule entre le client et le maître d'hôtel.

Cuisine

Avant l'arrivée des clients, un certain nombre de plats et les desserts sont préparés. À l'arrivée d'une commande, les différents plats sont répartis entre les cuisiniers par le chef de cuisine. Lors de la préparation d'une recette, le cuisinier utilise des ingrédients apportés et/ou préparés par le commis, ainsi que des ustensiles en fonction de leur disponibilité. Lors de la fin d'une tâche, le cuisinier libère l'outil afin qu'il soit nettoyé. C'est le rôle du plongeur que de s'occuper du lavage des ustensiles utilisés pour la préparation, ainsi que de gérer la vaisselle et le linge sale.

Logiciel de gestion

Le logiciel de gestion est utilisé par le gérant afin de visualiser l'activité de son restaurant, il peut donc identifier les points bloquants et mettre en place des solutions.

DIAGRAMMES DE SÉQUENCES

subsection*Salle

Chaque commande suit un ordre chronologique et implique la synchronisation des tâches réparties entre les différents acteurs.

Cuisine

Même scénario pour la cuisine, les tâches entre elles doivent être synchronisées afin de minimiser le temps mis par chaque tâche. La situation idéale implique qu'aucun cuisinier ne rencontre de problème lié à l'accès concurrent aux ressources.

EXPLICATION DU MCD

Nous disposons de « MaterielCommun » qui représente tout le matériel : « Assiettes », « Couverts » et « Verres ». Chacun de ces matériaux disposent de plusieurs types chacun comme des assiettes plates ou creuses par exemple.

« Salle » contient des chaises et cette salle comprends des « Carres ». Ces carrés contiennent des « Rangs » qui eux même contient des « Tables ». Ces tables ont un numéro et une capacité.

Il existe des « Recette » qui comprennent le nombre de personnes, ainsi que le temps de préparation, de cuisson et de pause. Ces recettes disposent d'un type qui définit si c'est un dessert ou un plat par exemple. Ces recettes disposent d'« Etape » qui sont composés d'un temps et d'une description. Dans ces étapes on va pouvoir y retrouver l'«Appareil », les « Denree » ainsi que les « Ustensile » que nous allons utiliser. Pour les ustensiles nous avons son nom, sa quantité et son temps de nettoyage. Pour les denrées nous disposons du nom, de sa quantité et du type de stockage (chambre froide ou congélateur). Pour terminer il y a l'appareil qui est composé d'un nom, d'une quantité et d'une capacité.