

Programmation Orientée Objet

Projet Gestion Hôtelière

3^e année ESIEA-FA-INF3034

Safa ZOGHLAMI

Présentation générale :

L'objectif du projet est de pouvoir fournir une petite application en Java offrant une interface pour la gestion des réservations dans un hôtel. De ce fait, nous allons nous placer dans le contexte d'un logiciel de gestion hôtelière qui peut être utilisé pour gérer plusieurs activités telles que le stockage des détails des clients, la réservation des chambres de quatre types différents, la gestion des commandes de repas. Cette application permet également d'afficher les différentes caractéristiques et les disponibilités des chambres.

Il est essentiel de ne pas perdre l'information des réservations entre deux utilisations du programme et c'est pourquoi nous allons utiliser les fichiers pour stocker l'état actuel de l'hôtel (coordonnées du client, chambres réservées, repas commandés) une fois le programme est terminé. Après, lorsque nous redémarrons le programme, les anciens détails ne seront pas perdus et le programme lit le fichier quand il redémarre pour connaître le statut précédent de l'hôtel. Une exception définie par l'utilisateur est levée si l'utilisateur essaie de réserver une chambre déjà attribuée. La gestion des exceptions doit être correctement effectuée pour traiter toute sorte d'exceptions inattendues.



Nous allons mettre en œuvre différentes notions :

- Classes et objets, héritage et redéfinition de méthodes.
- Gestion des fichiers avec des objets.
- Les collections
- Les Interfaces
- La gestion des exceptions.

L'hôtel propose au maximum quatre catégories de chambres différentes (chambre de luxe lit double, chambre normale lit double, chambre de luxe lit simple et chambre normale lit simple).

L'application doit obligatoirement proposer les tâches suivantes :

1. Afficher les détails des chambres.
2. Afficher la disponibilité des chambres.
3. Permettre les tâches de gestion des réservations des chambres : réservation, modification, annulation et suppression d'une réservation.
4. Commander un repas.
5. Enregistrer la facture lors de départ du client. Cette facture regroupe le prix de la chambre et la quantité des prestations consommées pendant le séjour (petit déjeuner, déjeuner, etc).

Vous pouvez rajouter d'autres fonctionnalités que vous jugez nécessaires pour votre projet.

Travail à réaliser

Vous développerez cette application à partir des besoins exprimés ci-dessus, à vous d'analyser et concevoir une solution fonctionnelle. Vous vous aiderez pour cela du langage UML pour conceptualiser les différentes entités et fonctionnalités de votre projet.

Avant de livrer votre solution, vous prendrez bien évidemment soin de réaliser les tests nécessaires pour s'assurer de son bon fonctionnement.

Le projet est à réaliser obligatoirement en binôme sous l'environnement Linux, et doit être déposé sur la plate-forme pédagogique Moodle au plus tard le 06/01/2024 à 23h55, sous la forme d'une archive .zip à vos noms et prénoms (NOM1_NOM2.zip), contenant tous les fichiers sources du projet ainsi qu'un rapport de trois pages au format *pdf*.

Votre programme doit obligatoirement présenter les différentes thématiques suivantes :

- Héritage
- Sérialisation
- Gestion des exceptions
- Les collections

Les principaux critères d'évaluation de votre projet sont : la qualité globale de la réalisation, la qualité du code, les livrables fournis.

NB : Le rapport ne devra pas comporter de code et uniquement décrire :

- Les contributions de chaque étudiant.
- Le travail réalisé.
- Les difficultés rencontrées et les solutions qui leur ont été apportées.

Techniques à utiliser :

Sérialisation: En informatique, la sérialisation est un processus visant à coder l'état d'une information qui est en mémoire (une structure de donnée, un objet, etc.) sous la forme condensée, par exemple une suite d'octets. Cette suite pourra par exemple être utilisée pour la sauvegarde (persistance) ou le transport sur le réseau (proxy). L'activité symétrique, visant à décoder cette suite pour créer une copie conforme de l'information d'origine, s'appelle la désérialisation.

Java intègre naturellement des méthodes de sérialisation/désérialisation permettant de sauvegarder/charger des objets dans des fichiers.

Pour cela, la classe que l'on désire sérialiser (et toutes les classes que l'on utilise au sein de cette classe) doivent simplement implémenter l'interface `Serializable`.

L'interface `Serializable` ne possède pas de méthode à redéfinir. Elle sert uniquement à identifier les classes pouvant être sérialisées.

<http://docs.oracle.com/javase/7/docs/api/index.html?java/io/Serializable.html>

Attention !

- Toutes similitudes détectées entre projets de classes ou plagiat sur les projets déjà existants sur internet (y compris chatGPT), etc, entrainera un conseil de discipline.
- Un projet qui contient un fichier manquant ou sans rapport entrainera un zéro.
- Tout projet envoyé par E-mail ne sera pas accepté. Vous devez obligatoirement déposer votre travail sur Moodle.
- Aucun projet ne sera accepté après la date limite.