

UNIVERSITE LOYOLA DU CONGO (ULC-ICAM)
Faculté des sciences et technologies(FAST)



DEPARTEMENT GENIE INFORMATIQUE

Projet programmation concurrente
DESIGNS PATTERNS

Fait par le **GROUPE 1:**

KASONGO MWANA NDJIBU Jude

KETATE KIBUSHI Isaac

NYIMI MVULA Francis

LUKALI Emmanuel

Dirigé par :

Mr KALOMBO Oscar

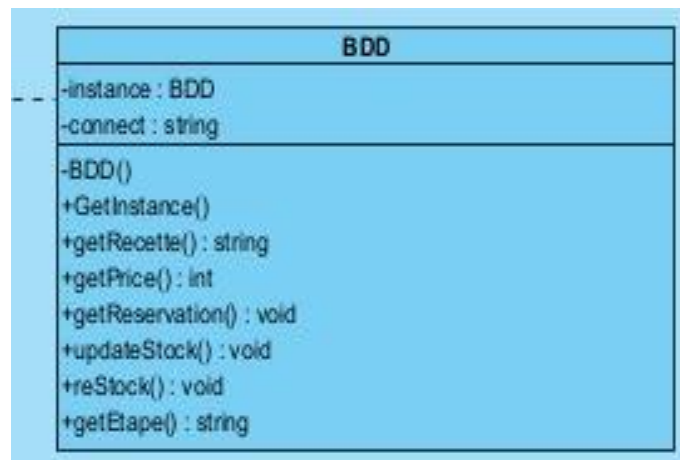
ANNEE ACADEMIQUE : 2024-2025

DESIGNS PATTERNS

Singleton

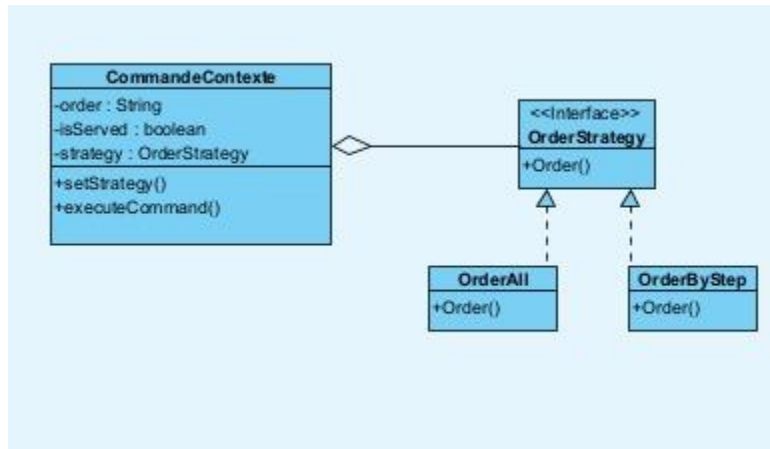
Singleton est un patron de conception de création qui garantit que l'instance d'une classe n'existe qu'en un seul exemplaire, tout en fournissant un point d'accès global à cette instance. Dans son implémentation le design pattern singleton consiste à déclarer le constructeur de la classe avec une visibilité 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.



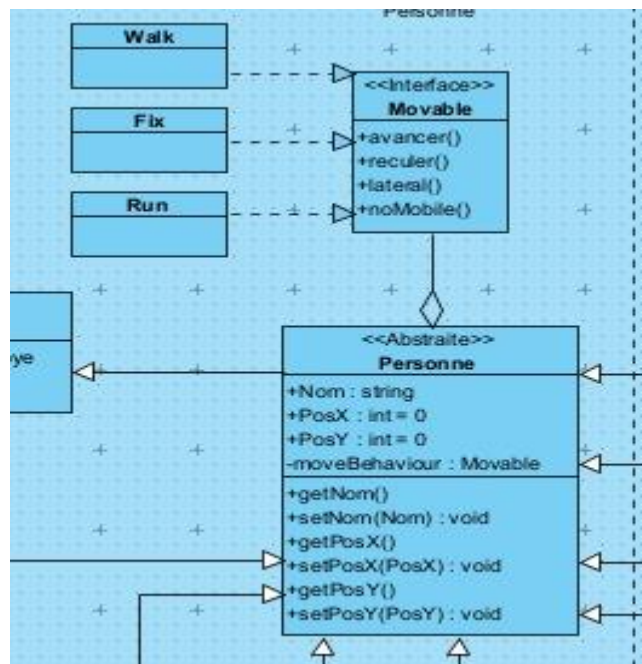
Strategy

Le design pattern stratégie permet de modifier le comportement d'une classe ou d'une de ses méthodes durant l'exécution. Le 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 Pont est un patron de conception structurel qui permet de séparer une grosse classe ou un ensemble de classes connexes en deux hiérarchies — abstraction et implémentation — qui peuvent évoluer indépendamment l’une de l’autre. Nous utilisons ce patron de conception pour gérer les méthodes de déplacements de différentes personnes.



MVC

Modèle-vue-contrôleur ou MVC est un motif d'architecture logicielle destiné aux interfaces graphiques, lancé en 1978 et très populaire pour les applications web. Le motif est composé de trois types de modules ayant trois responsabilités différentes : les modèles, les vues et les contrôleurs.

Dans notre, nous avons :

Un modèle (Model) contient les données à afficher ;

Une vue (View) contient la présentation de l'interface graphique ;

Un contrôleur (Controller) contient la logique concernant les actions effectuées par l'utilisateur.