

RAPPORT DE SPECIFICATION DES DESIGN PATTERN

Un patron de conception est un arrangement caractéristique de modules, reconnu comme bonne pratique pour résoudre un problème de conception d'un logiciel. Il décrit une solution standard, utilisable dans la conception de différents logiciels.

Ainsi pour la conception de notre application, nous avons opté pour l'utilisation des design pattern ci-dessous :

- MVC
- Singleton
- Builder
- Decorator
- FactoryMethod
- Prototype

1. Design Pattern MVC

Bien qu'imposé lors de l'énoncé du projet, le DP MVC est un design pattern de conception qui nous sera d'une très grande utilité dans la structuration de notre code. Il sera donc subdivisé en 3 parties à savoir :

- 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.

2. Design pattern Singleton

En génie logiciel, le singleton est un patron de conception (design pattern) dont l'objectif est de restreindre l'instanciation d'une classe à un seul objet (ou bien à quelques objets seulement). Il est utilisé lorsqu'on a besoin exactement d'un objet pour coordonner des opérations dans un système.

3. Design pattern Builder

Il permet de résoudre les problèmes de programmation dans la programmation orientée objet (POO). Ils facilitent la programmation pour les développeurs : ainsi,

vous n'avez pas à repenser la routine de chaque étape récurrente puisque des propriétés qui peuvent être modifiées ont déjà été établies.

4. Design pattern Decorator

A l'aide du Design pattern Decorator, tout objet peut être étendu par un comportement souhaité sans affecter le comportement des autres objets de la même classe.

5. Design pattern Prototype

Prototype est un patron de conception qui crée de nouveaux objets à partir d'objets existants sans rendre le code dépendant de leur classe.