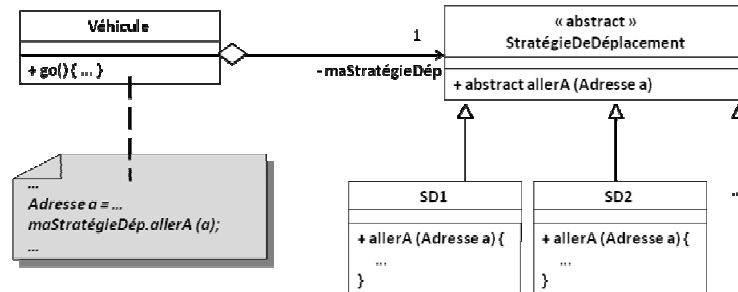


UE MCDC - Design Patterns - Feuille d'exercices n°1

EXERCICE 1

Le diagramme de classes ci-dessous définit une application basée sur le design pattern Stratégie : la classe Véhicule utilise une stratégie de déplacement dont le supertype est StratégieDeDéplacement.



Supposons que les différentes implantations de la méthode allerA(Adresse a) dans les classes SD1, SD2... ne varient que pour une petite partie : le calcul de l'itinéraire.

- Quel design pattern convient pour éviter la duplication du code commun ?
- Critiquez les mises en œuvre (au moyen de ce pattern) proposées par Alice, Bob, Carole et David (cf. Annexe 1).
- Concevoir la solution en modifiant le diagramme de classes ci-dessus. Donner les éléments de code utiles.
- Donner un exemple de code de déploiement (création et configuration d'un véhicule avec une stratégie au choix).

EXERCICE 2

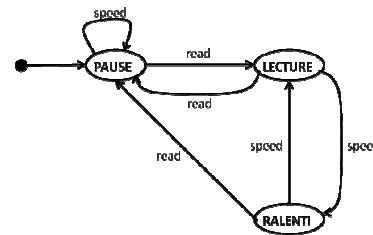
On considère un objet de type Lecteur qui permet de lire une vidéo. On suppose que la vidéo peut être lue en boucle (elle n'a pas de début ni de fin).

A tout instant, le lecteur est soit en PAUSE, soit en cours de LECTURE en mode normal, soit en cours de lecture en mode RALENTI.

Deux actions sur le lecteur sont possibles : READ et SPEED. Pour cela, le lecteur offre deux méthodes sans paramètre : read() :void et speed() :void.

L'effet de l'action dépend de la situation :

- l'action READ, quand le lecteur est en PAUSE, provoque la LECTURE de la vidéo en mode normal
- l'action READ met le lecteur en PAUSE quand la vidéo est en cours de LECTURE en mode normal ou RALENTI
- l'action SPEED, quand le lecteur est en LECTURE en mode normal, fait passer en mode RALENTI et inversement
- l'action SPEED est sans effet quand le lecteur est en PAUSE.



- Concevoir l'application et décrire la solution (diagramme de classes, etc.).
- La solution met en œuvre un design pattern : lequel ? De quel autre pattern est-il proche et en quoi diffère-t-il ?

EXERCICE 3

Vous intervenez dans le cadre d'un projet de maintenance d'une application de robotique développée par un ancien étudiant du Master Informatique qui a mis en application ses connaissances sur les patrons de conception. Dans cette application, des robots localisés géographiquement sont capables de connaître leur position et de se déplacer en une position donnée. Pour cela, les robots peuvent utiliser différentes méthodes pour calculer le chemin à suivre.

Malheureusement, la documentation sur l'application a disparu ! Vous devez donc reprendre le code (cf. Annexe 2).

- Donnez-le diagramme de classes et retrouvez les patrons de conception utilisés. Pour chacun identifiez les différents participants. Expliquez.
- Quel est l'impact sur le code existant des évolutions ci-dessous ?
 - Ajout d'une classe CC6 qui implante CalculateurDeChemin
 - Ajout d'une classe RobotBis qui utilise un CalculateurDeChemin comme Robot