

Rapport TP 2-7, ACO : Mini-éditeur

Auteurs : DIALLO, Djiby – KHEIRI, Sarah

Encadrant de TP : BATEUX, Quentin

Master 1, Génie logiciel, Groupe 1.1

1 Introduction

Dans le cadre du module d'ACO, il nous avait été demandé de réaliser un mini-éditeur de texte offrant les fonctionnalités basiques que l'on retrouve dans la plupart des éditeurs de texte. La réalisation de ce mini-éditeur s'est faite tout d'abord en travaux dirigés, pour l'analyse des besoins et l'organisation des différentes parties du mini-éditeur, et en travaux pratiques pour leur application.

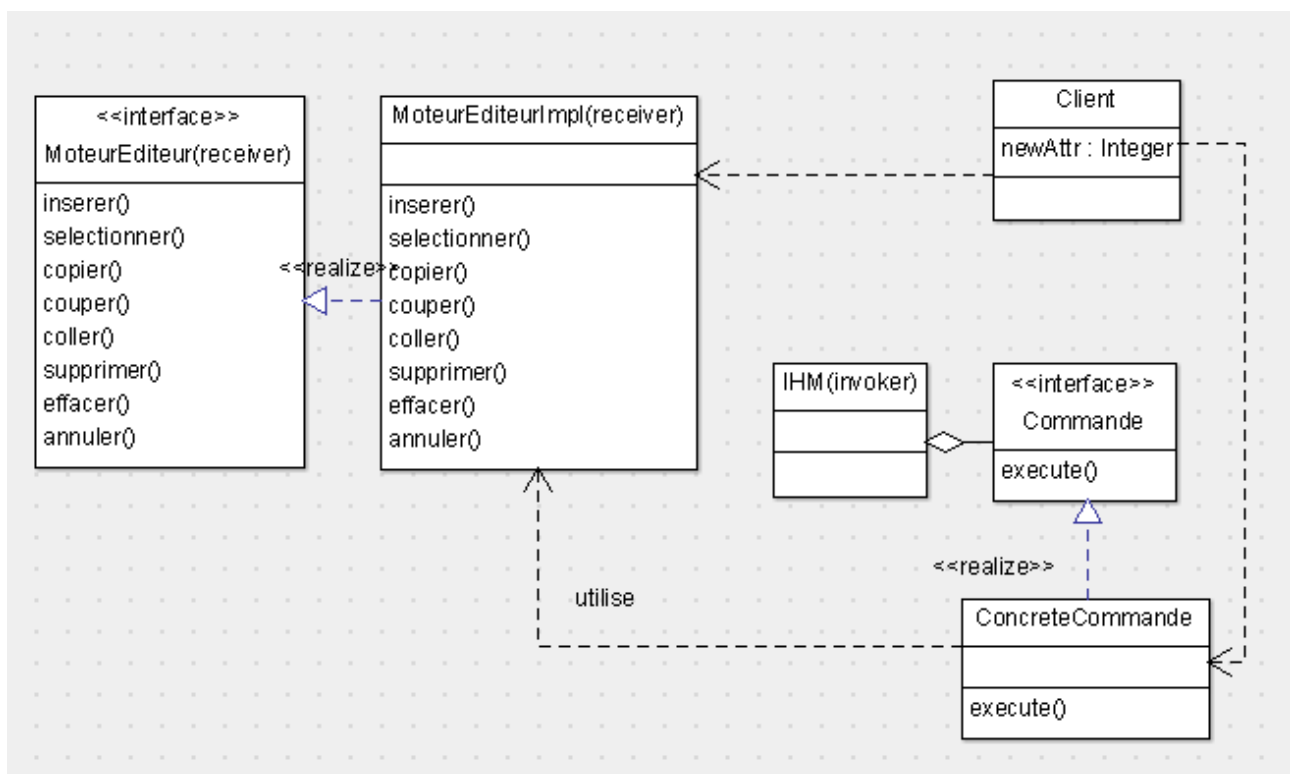
Ce mini-éditeur est conçu en spirale en trois versions successives et les outils utilisés pour sa conception sont Java, Eclipse, Junit ainsi que quelques patrons de conception (design pattern) vus en cours.

2 Les composants du mini-éditeur

Notre mini-éditeur est constitué d'une interface(front end) permettant à l'utilisateur de faire son édition de texte avec des boutons (ou options) représentant l'ensemble des actions qu'il peut réaliser, et d'un backend assurant le bon fonctionnement du front-end c'est à dire la satisfaction de toutes les demandes de l'utilisateur.

3 L'architecture du mini-éditeur

Le mini-éditeur a été réalisé en utilisant le patron de conception commande

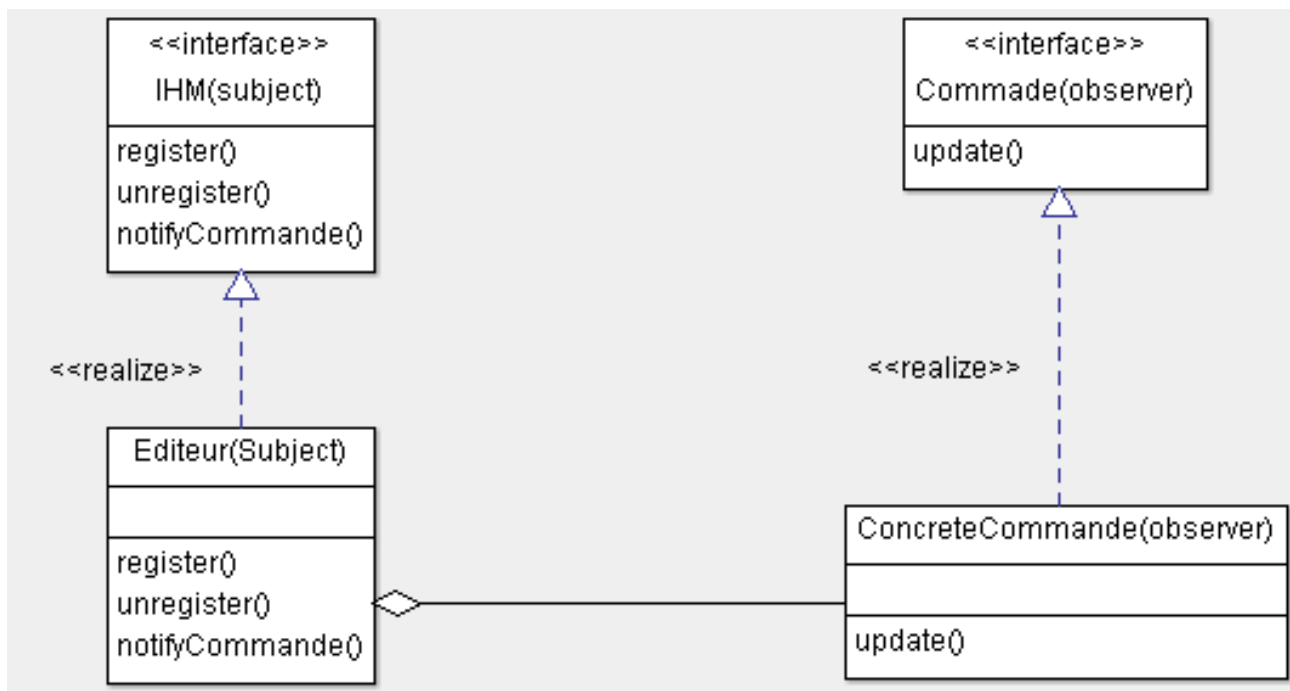


4 Première version

a) Réalisation

Pour cette version, notre mini-éditeur comportera uniquement les actions d'édition de base, dans notre cas l'utilisateur aura la possibilité d'insérer, de sélectionner, de copier, de couper, de coller, d'effacer (caractère par caractère), de supprimer (plusieurs caractères sélectionnés au préalable) ou d'annuler(reset de l'éditeur dans ce cas) ses actions.

A noter que nous avons implémenté le patron de conception observer entre l'IHM(plus précisément sa zone d'édition) et les commandes pour pouvoir informer ces dernières des éventuel changements comme par exemple le déplacement du curseur.



b) Fonctionnement

Pour éditer du texte dans notre mini-éditeur, l'utilisateur aura à sa disposition, lorsqu'il démarre le programme, une interface d'édition(fenêtre) avec les éléments suivants :

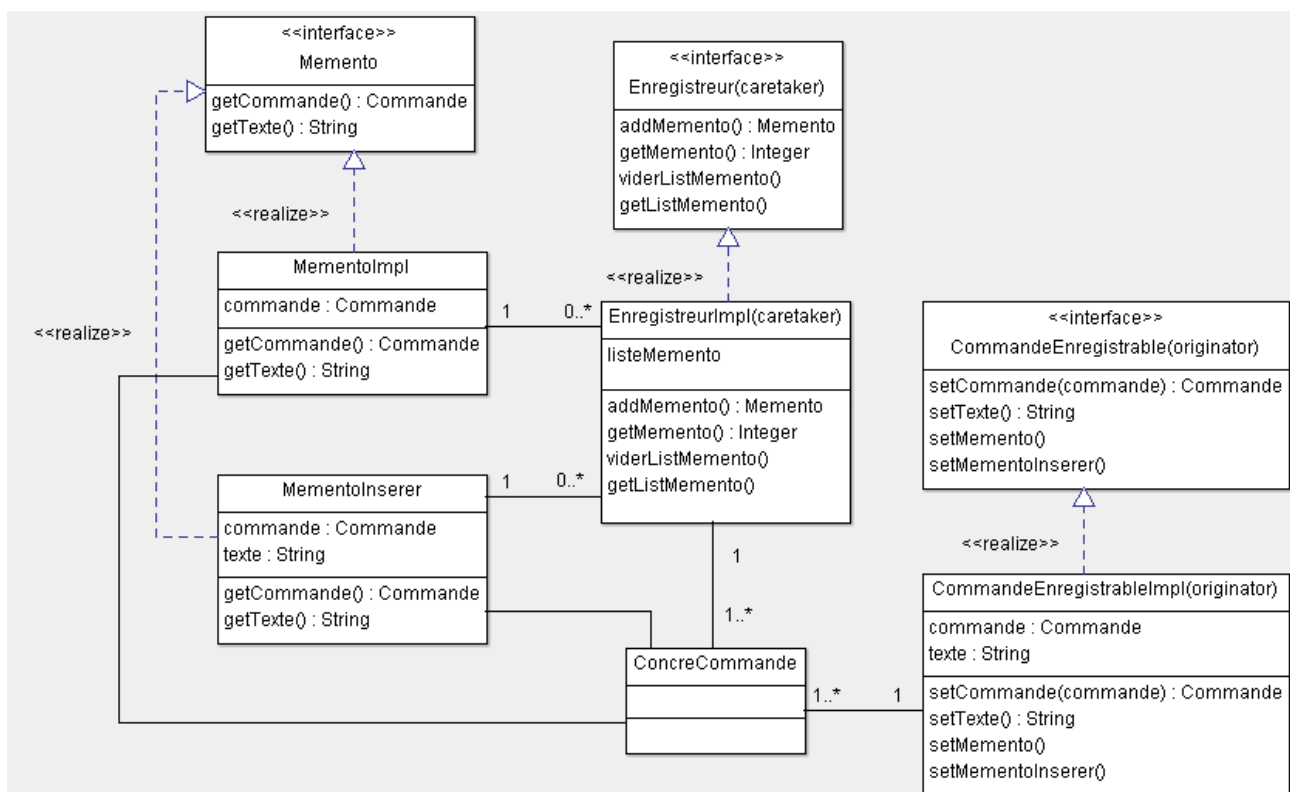
- Un menu d'édition avec les commandes suivantes : sélectionner, copier, couper, coller, supprimer
A noter que dans notre cas, la sélection fonctionne de la manière suivante : sélection du texte avec la souris puis menu édition puis la sélectionner
- Un bouton effacer pour effacer le texte éditer caractère par caractère
- Un bouton annuler pour remettre à zéro l'éditeur (reset).
- Une zone de texte pour insérer du texte par l'intermédiaire de notre commande insérer qui sera appelé automatiquement.
- L'utilisateur, s'il le veut, peut fermer l'éditeur en faisant fichier puis quitter ou bien en fermant la fenêtre d'édition.

5 Seconde version

a) Réalisation

Cette version va permettre à l'utilisateur d'enregistrer ses actions pour pouvoir les rejouer plus tard. Pour faire cela, nous avons utilisé le patron de conception Memento pour enregistrer les différentes actions de l'utilisateur.

A noter que notre refaire agit sur l'état actuel du système et non sur l'état d'enregistrement.



b) Fonctionnement

Pour enregistrer des actions dans notre mini-éditeur, l'utilisateur a à sa disposition un bouton enregistrer et un bouton rejouer.

- Le bouton enregistrer, une fois appuyer dessus, permet à l'utilisateur d'enregistrer tous ses actions qui suivront.
- Le bouton rejouer quant à lui permet à la fois d'arrêter l'enregistrement et de rejouer tous les actions enregistrées.

6 Troisième version

a) Réalisation

Dans cette dernière veriosn, l'utilisateur pourra réaliser le défaire/refaire avec des capacités quelconque dans le défaire(autrement dit on peut revenir au début)

Pour réaliser le défaire/refaire, nous avons choisi la solution du sauvegarde de l'état du système avant exécution de chaque action de l'utilisateur.

b) Fonctionnement

Pour pourvoir défaire ou refaire des actions dans notre mini-éditeur, l'utilisateur utilise deux boutons :

- Un bouton défaire pour défaire une action passée.
- Un bouton refaire pour la refaire une action défaite.

7 Conclusion

Notre travail consistait à réaliser un mini-éditeur dans le cadre du module ACO du premier semestre. Cela s'est fait en se basant sur un cahier des charges, qui spécifiait la production de trois versions successives, et en utilisant certains patrons de conception vu en cours. Ce projet nous a permis d'approfondir nos connaissances en programmation objet et d'acquérir des notions sur la conception de logiciel.