

# RAPPORT DE CONCEPTION

*TP SR encadré par Mr. Charles Quéguiner*

Rapport sur la réalisation et la conception d'un éditeur de texte en trois versions différentes pour le cours d'ACO.

Maël Nogues & Mathieu Grandmontagne

06/12/2016



# Rapport de conception

*TP SR encadré par Mr. Charles Quéguiner*

Lors de ce semestre nous avons vu plusieurs patrons de conception différents. Pour mieux comprendre à quoi ils servent et comment les mettre en œuvre, nous avons réalisé en séances de TP un éditeur de texte comportant différentes versions permettant chacune des actions différentes, supplémentaires par rapport à la version précédente.

## EDITEUR VERSION 1 :

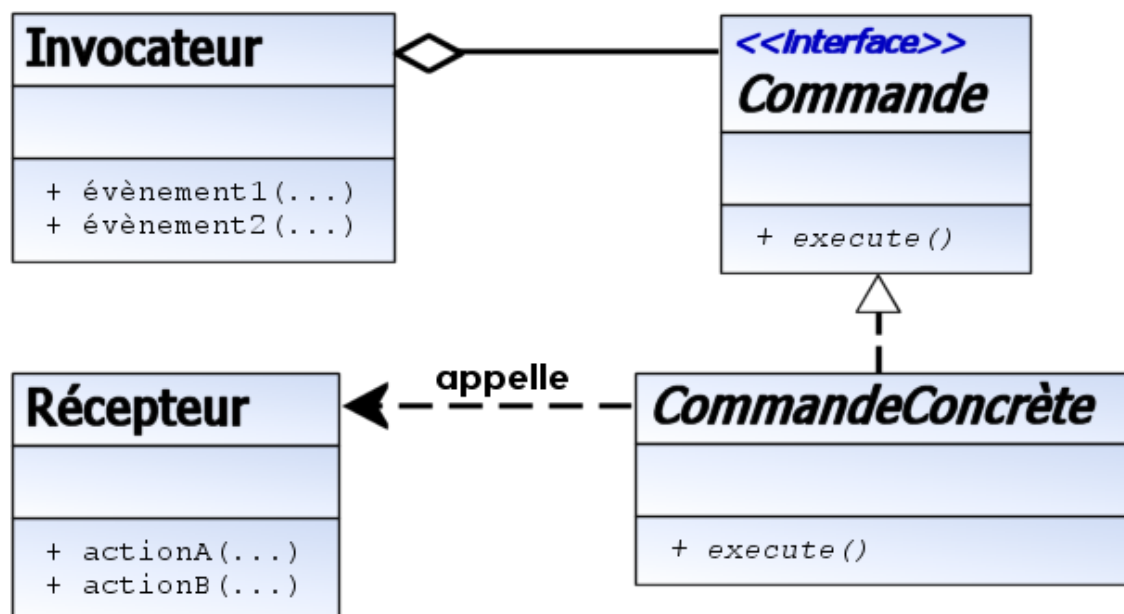
La première version de l'éditeur de texte est composée d'une interface graphique dans laquelle il est possible de saisir, copier, couper, coller, supprimer et sélectionner du texte.

Le diagramme de classe correspondant à cette version se situe dans le dossier contenant ce rapport sous le nom de v1.png.

### Patron de conception commande :

Le patron de conception commande permet de séparer le code initiateur d'une action du code de l'action elle-même. Il est utilisé dans les interfaces graphiques afin de pouvoir connecter plusieurs commandes différentes à un même élément du menu par exemple.

Nous utilisons le patron de conception commande comme vu en TD, pour séparer le code de l'interface graphique de celui des commandes.



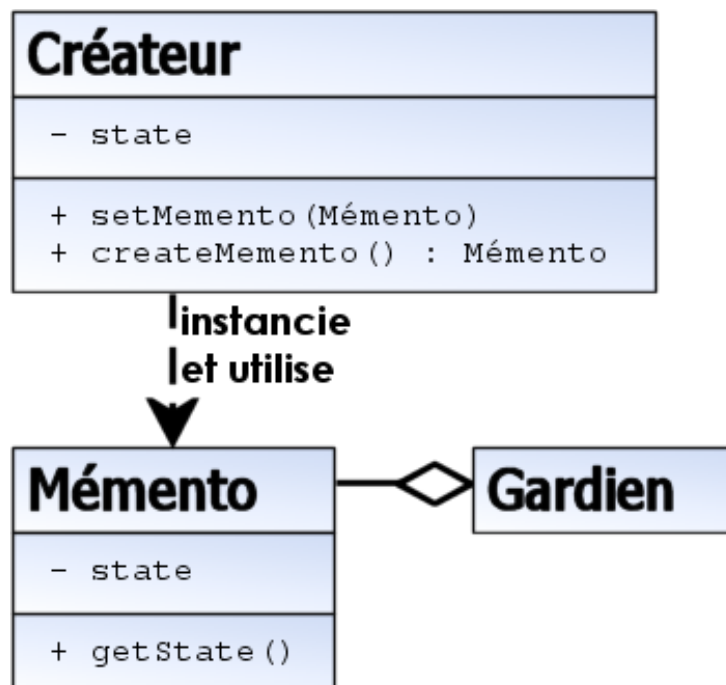
## EDITEUR VERSION 2 :

La seconde version de l'éditeur de texte ajoute la possibilité d'enregistrer des macros en appuyant sur un bouton pour lancer l'enregistrement, puis sur un autre bouton pour le stopper. Après avoir enregistré une macro, il est possible de la rejouer en pressant un troisième bouton.

Le diagramme de classe correspondant à cette version se situe dans le dossier contenant ce rapport sous le nom de v2.png.

### Patron de conception memento :

Le patron de conception memento permet de restaurer un objet à un état précédent sans compromettre l'encapsulation des données de cet objet. Ce patron de conception est utilisé dans notre éditeur afin de conserver les commandes d'une macro pour pouvoir les réutiliser plus tard.



## EDITEUR VERSION 3 :

La troisième version de l'éditeur de texte ajoute la gestion d'un historique des commandes réalisées, permettant l'annulation des commandes, ainsi que la réapplication des commandes annulées.

Le diagramme de classe correspondant à cette version se situe dans le dossier contenant ce rapport sous le nom v3.png.

### Patron de conception memento :

Pour l'implémentation de la gestion de l'historique des commandes, nous avons choisi de réutiliser le patron memento pour sauvegarder l'état du buffer contenant le texte, ainsi que celui de la sélection pour pouvoir défaire une action simplement en restaurant l'état du buffer et de la sélection avant l'action ayant modifiée le buffer (copier, coller, couper, supprimer).

## OUTILS DE TESTS MOCKITO :

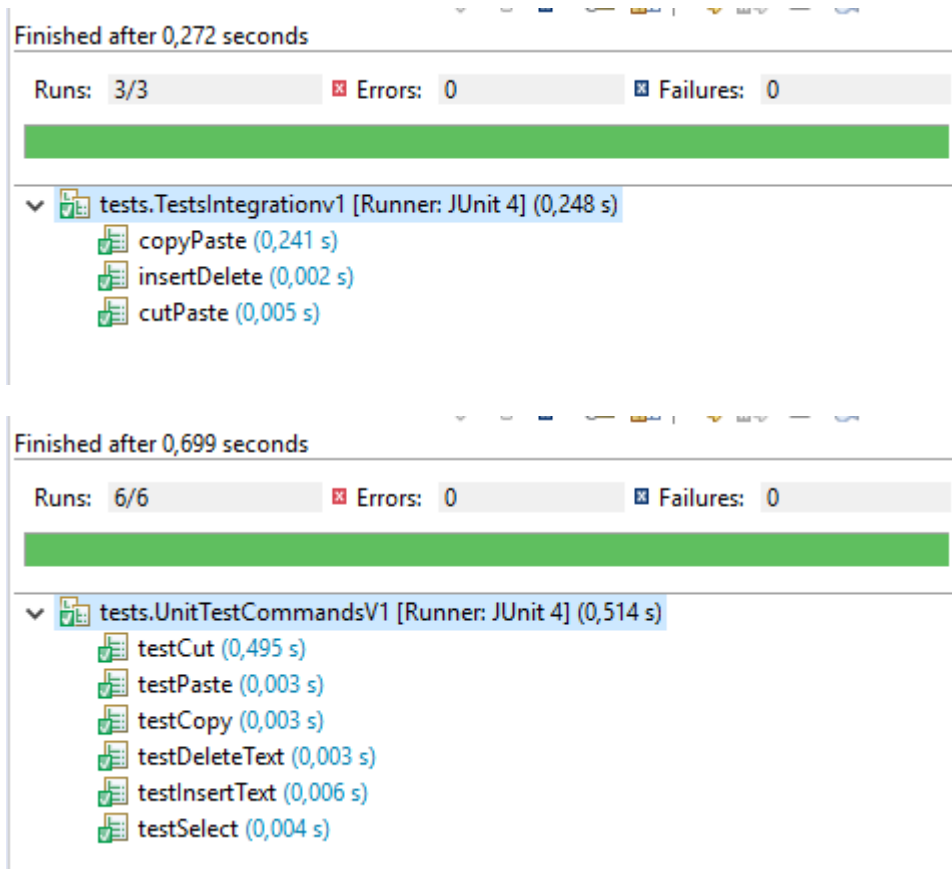


Mockito est un framework de tests en open source qui permet d'utiliser des objets de tests particuliers capables de vérifier les sorties d'un objet et de les comparer avec des sorties valides que l'on donne à l'objet Mock qui fait ces tests.

Le framework mockito permet donc d'écrire des tests beaucoup plus concis et clairs afin de tester plus facilement les objets de notre programme.

## RESULTAT DES TESTS :

Tests de la version 1 :



Finished after 0,272 seconds

Runs: 3/3    ✖ Errors: 0    ✖ Failures: 0

▼ tests.TestsIntegrationv1 [Runner: JUnit 4] (0,248 s)

- ✓ copyPaste (0,241 s)
- ✓ insertDelete (0,002 s)
- ✓ cutPaste (0,005 s)

Finished after 0,699 seconds

Runs: 6/6    ✖ Errors: 0    ✖ Failures: 0

▼ tests.UnitTestCommandsV1 [Runner: JUnit 4] (0,514 s)

- ✓ testCut (0,495 s)
- ✓ testPaste (0,003 s)
- ✓ testCopy (0,003 s)
- ✓ testDeleteText (0,003 s)
- ✓ testInsertText (0,006 s)
- ✓ testSelect (0,004 s)

## Tests de la version 2 :

Finished after 0,308 seconds

Runs: 1/1 Errors: 0 Failures: 0

▼ tests.TestsIntegrationv2 [Runner: JUnit 4] (0,269 s)

     insertDeleteReplay (0,269 s)

---

Finished after 0,612 seconds

Runs: 6/6 Errors: 0 Failures: 0

▼ tests.UnitTestCommandsV2 [Runner: JUnit 4] (0,072 s)

     testCopyRecordable (0,053 s)

     testCutRecordable (0,004 s)

     testDelTextRecordable (0,003 s)

     testInsTextRecordable (0,004 s)

     testSelectRecordable (0,004 s)

     testPasteRecordable (0,004 s)

## Tests de la version 3 :

Finished after 0,295 seconds

Runs: 1/1 Errors: 0 Failures: 0

▼ tests.TestsIntegrationv3 [Runner: JUnit 4] (0,284 s)

     test (0,284 s)

---

Finished after 0,587 seconds

Runs: 2/2 Errors: 0 Failures: 0

▼ tests.UnitTestCommandsV3 [Runner: JUnit 4] (0,570 s)

     testUndo (0,567 s)

     testRedo (0,003 s)