

Rapport projet mini-éditeur

Fritzgy LUBIN
Abdou IBOURAIMA

M1 MIAE
Groupe 1B
2021-2022
ACO

Table des matières

Architecture.....	3
Diagramme UML.....	3
Commentaires.....	4
Rapport des tests.....	5
Tests unitaires.....	5
Tests d'intégration.....	5
Guide d'utilisation.....	6

Architecture

Diagramme UML

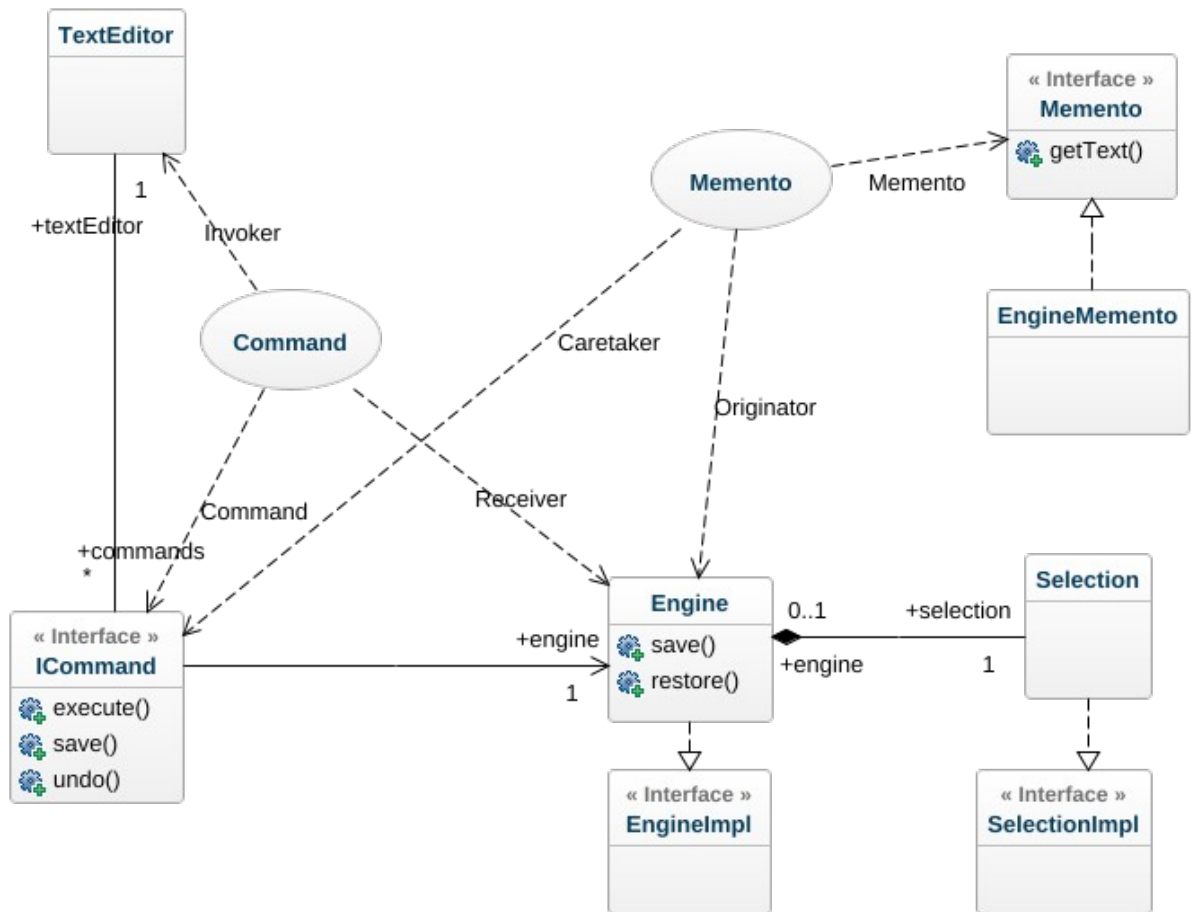


Illustration 1: Diagramme de classe UML

Commentaires

La classe **TextEditor** est l'IHM/Invoker du projet, elle s'occupe d'*exécuter les concrete command* (**InsertCommand**, **CutCommand**, **CopyComand**, **PasteCommand**, **DeleteCommand**, **ReplayCommand**) en fonction de l'action de l'utilisateur.

Pour la **V3**, on a mis en place un **système de commande inversée**, à partir d'une instance de commande donné une commande inverse est générée et appliquée.

Les instances des commandes précédentes et suivantes sont enregistrées dans la classe **TextEditor** sous deux objets de type **Stack<ICommand>**. À l'appel de l'action **undo** par exemple, on récupère le premier élément de la pile et on appelle sa fonction *undo*, qui va s'occuper de **mettre à jour l'état de l'Engine**.

Rapport des tests

Tests unitaires

Voir : <https://gitlab.istic.univ-rennes1.fr/17012392/text-editor/-/blob/V3/test-reports/unit-tests.pdf>

Tests d'intégration

Voir : <https://gitlab.istic.univ-rennes1.fr/17012392/text-editor/-/blob/V3/test-reports/integration-tests.pdf>

Guide d'utilisation

- Pour lancer le projet, veuillez exécuter la **classe Main** dans le **package App**.
- L'utilisateur peut rentrer du texte directement dans la fenêtre affichée et exécuter les actions de l'**onglet Edit**.