

Rapport tp1

Nahnah Abdesselam

et

Daher Bilal

Département d'informatique et de recherche opérationnelle
Faculté des arts et sciences

4 Octobre 2018

1) Les Packages :

-Components : ce package contient toutes les composantes graphique qui sont inclusent dans la fenetre principale du programme et qui permette de visualiser les détails du diagramme parsé

-ActionListeners: ce package contient toutes les actions qui sont reliés au différentes composante du programme et permet de faire différents actions sur ces composantes

-Controller: ce package contient les classes qui permettes d'ajouter les différentes composantes dans la fenetre principal et de définir l'emplacement et la taille de chaqu'une des ces composantes

-Comment: ce package contient les définitions des déclarations du diagramme dont classe ,associations,generalizations,aggregations , il contient aussi la classe UmlParser qui se charge de parser les fichier du type .ucd

-Notifiers: ce package contient toutes les classe Notifier . Une classe Notifier est une sous classe de la classe Observable qui est une classe prédéfinie de Java et qui permet de notifier les classes qui implémente l'interface Observer ,ces dernières doivent définir la méthode update qui se charge de mettre à jours les données à chaque changement de la classe notifier à laquel elle est reliés , dans notre cas ce sont les composantes qui implémentent l'interface Observer pour se mettre a jours a chaque Action performée sur ces dernières .

2) Les Pattern utilisé

- la classe Singleton : dans notre programme on a mis la fenêtre (JFrame) comme classe Singleton car on veut juste avoir une seule instance de cette fenêtre, de plus on pourrait facilement lancer des messages d'erreur de type JOptionPane qui prennent l'instance de la fenêtre pour pouvoir afficher le message et puisque on a une seule et unique objet de la fenêtre c'est facile d'y accéder à partir de n'importe quelle autre classe
- Le pattern de l'observateur (Observer Pattern) : ce pattern permet créer des classes qui observent d'autre classe de différent package et se met à jours à chaque changement effectué sur ces classes observées , et du coup dans notre cas puisque les ActionListener sont séparés des classes auxquelles ils y sont attachés il a fallu notifier les composantes qui devaient se mettre à jours à chaque Action performée comme par exemple il a fallu notifier les quatre JScrollPane du milieu de charger les bons données à chaque sélection de classe dans le JScrollPane du gauche .

3) Les classes Importantes

- Features: cette classe permet de définir les caractéristiques de chaque composante suivant le GridBagLayout comme leurs position x , position y , la taille de la colonne , la taille de la ligne , le margin ..etc
- MainController: la classe qui fait l'insertion de toutes les composantes dans la fenêtre et relier les classes (notifiers) à leurs observateurs
- DataApi : la classe comportant les hashMaps contenant les détails des classes , aggregations et relations , ces derniers sont déclarés static pour qu'on puisse y accéder de partout

- UmlParser: la classes qui fait le parsing des fichiers de type ucd et traite les erreurs , elle permet aussi de peupler les hashMaps de la classe DataApi avec les détails nécessaire

4)Comment ça Marche

-On commence par exécuter la classe main qui permet d'afficher la fenêtre , ensuite on clique sur le bouton charger fichier , ce dernier est relié a la class UploadFileListener qui permet d'afficher le JFileChooser pour faire la sélection d'un fichier de type .ucd . Une fois le fichier choisi ,la classe FileUploaderButton va appeler la classe UmlParser qui a son tours fera la lecture et le parsing du fichier et cela en le transformant en String et en faisant un split sur le (";") pour séparer chaque sous partie du fichier ,une fois que cela est fait on peut itérer sur chaque sous partie et extraire les données et les sauvegarder dans les hashMaps de la class DataApi .

Une fois le parsing du fichier est fini le ClassNotifier va mettre à jours toutes les classes lu et les afficher .

Maintenant qu'on a toutes les classes si on sélectionne une classe le ListSelectionListener sera exécuter qui a son tours notifie les autres classes Notifiers pour qu'elles se mettes à jours a leurs tours et pour qu'elles affichent les bonnes informations dépendamment de la classe choisie

5)Execution

-) par ligne de commande :

```
java -jar Uml_Parser.jar
```

ceci va executer le fichier jar

