SAE 2.01
Rapport de la semaine 1

Sommaire

Mise en place du dépôt

Création du dépôt

L'une des premières étapes de cette SAÉ a été de créer un fork à partir du dépôt original. Pour cela, un groupe à été créé pour que nous puissions travailler directement ensemble. Ensuite, le rôle de reporter a été donné aux deux professeurs Adrien Krähenbühl et Raymond Schneider pour qu'ils puissent y avoir accès.

Clonages du dépôt

Pour pouvoir travailler localement, on a chacun fait un clone du dépôt et nous avons ensuite tous les deux testé le dépôt en créant un fichier et en faisant un push vers gitlab. On peut voir ces deux commit sous le nom de 1a0e994b et 394a5df5. On a bien sûr supprimé par la suite les fichiers de tests envoyés sur le dépôt.

Exécution de la commande javadoc

Création du projet sur IntelliJ

Avant de réécrire la commande javadoc proposée dans le fichier Java2Puml.java, nous avons d'abord créé un projet portant le nom "p21_projet" utilisant le SDK correto-18. On a ensuite pu compiler le projet et continuer la SAÉ.

Conception et exécution de la commande

Comme nous utilisons tous les deux des ordinateurs sous Windows, nous avons rencontré un problème sur la commande javadoc qui n'a pas été reconnue. Pour cela on a cherché le fichier javadoc.exe dans le répertoire bin de JDK et nous l'avons exécuté au lieu de taper la commande.

La commande utilisée demandait de renseigner un package, on a donc décidé de prendre le package western venant de la correction du cours de P21 et nous l'avons placé dans le répertoire src. Après avoir changé les paramètres de la commande, nous sommes arrivé à ce résultat :

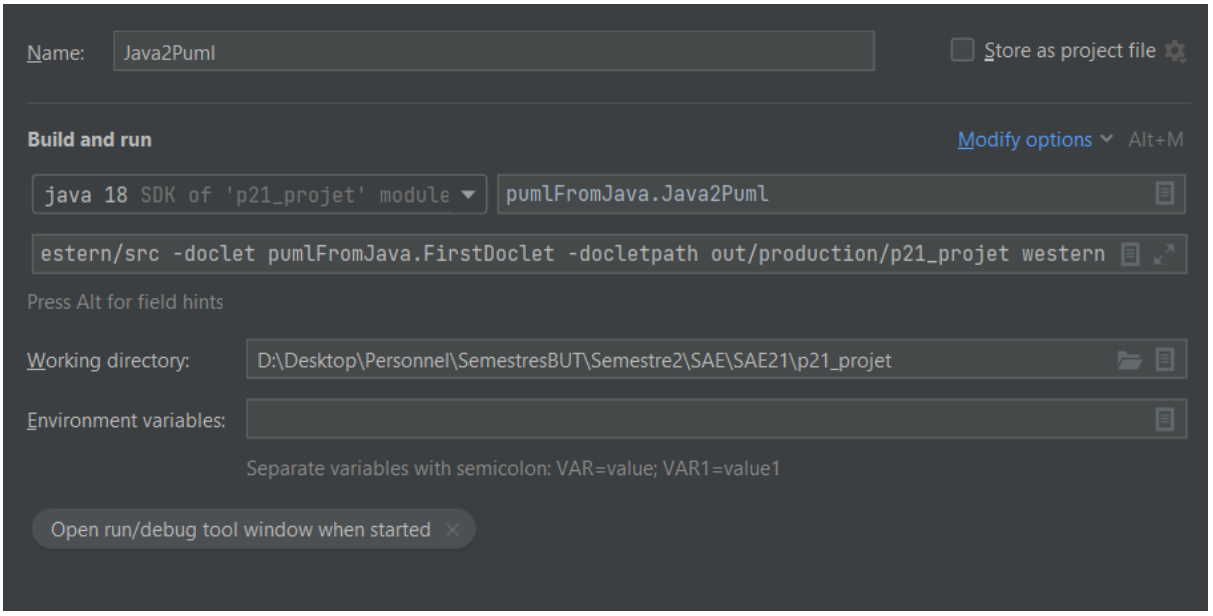
D:\Programmes\JDK\bin\javadoc.exe -private -sourcepath src/western/src -doclet pumlFromJava.FirstDoclet -docletpath out/production/p21_projet western

L'exécution de la commande a retourné dans le terminal les informations suivantes :

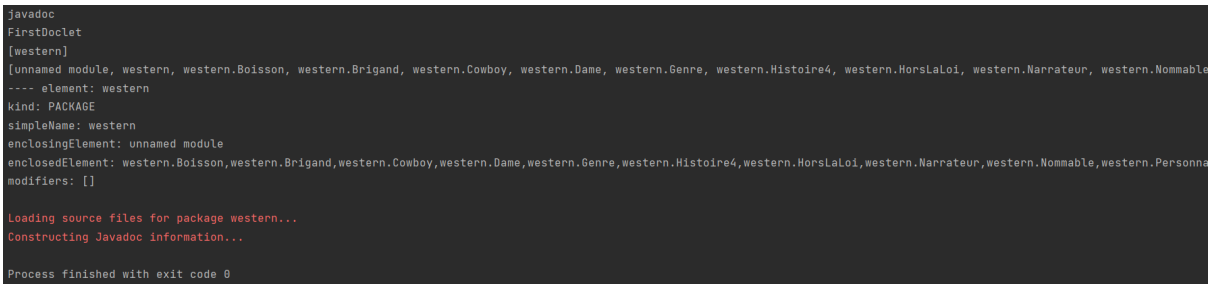
```
PS D:\Desktop\Personnel\SemestresBUT\Semestre2\SAE\SAE21\p21_projet> D:\Programmes\JDK\bin\javadoc.exe -private -sourcepath src/western/src -doclet pumlFromJava.FirstDoclet -docletpath
 out/production/p21_projet western
Loading source files for package western...
Constructing Javadoc information...
FirstDoclet
enclosingElement: unnamed module
enclosedElement: western.Boisson,western.Brigand,western.Cowboy,western.Dame,western.Genre,western.Histoire4,western.HorsLaLoi,western.Narrateur,western.Nommable,western.Personnage,wes
tern.Ripou,western.Substantif
modifiers: []
```

Edition de la configuration d'exécution

Pour la configuration de l'exécution, nous avons simplement entré dans la case "main class" le nom de la classe contenant le main (Java2Puml) et ensuite nous avons entré les arguments de la commande précédente dans la case "Program arguments".



Une fois exécuté, le terminal a affiché un code sortie valant 0, ce qui veut dire qu'aucun problème n'est survenu.



Production des diagrammes Plant UML

Diagramme de séquence

Pour faire le diagramme de séquence, on s'est basé sur l'appel du doclet effectué dans Java2Puml. On a ensuite remonté chaque méthode appelée sur les variables pour définir les classes utilisées et pour remplir le diagramme.

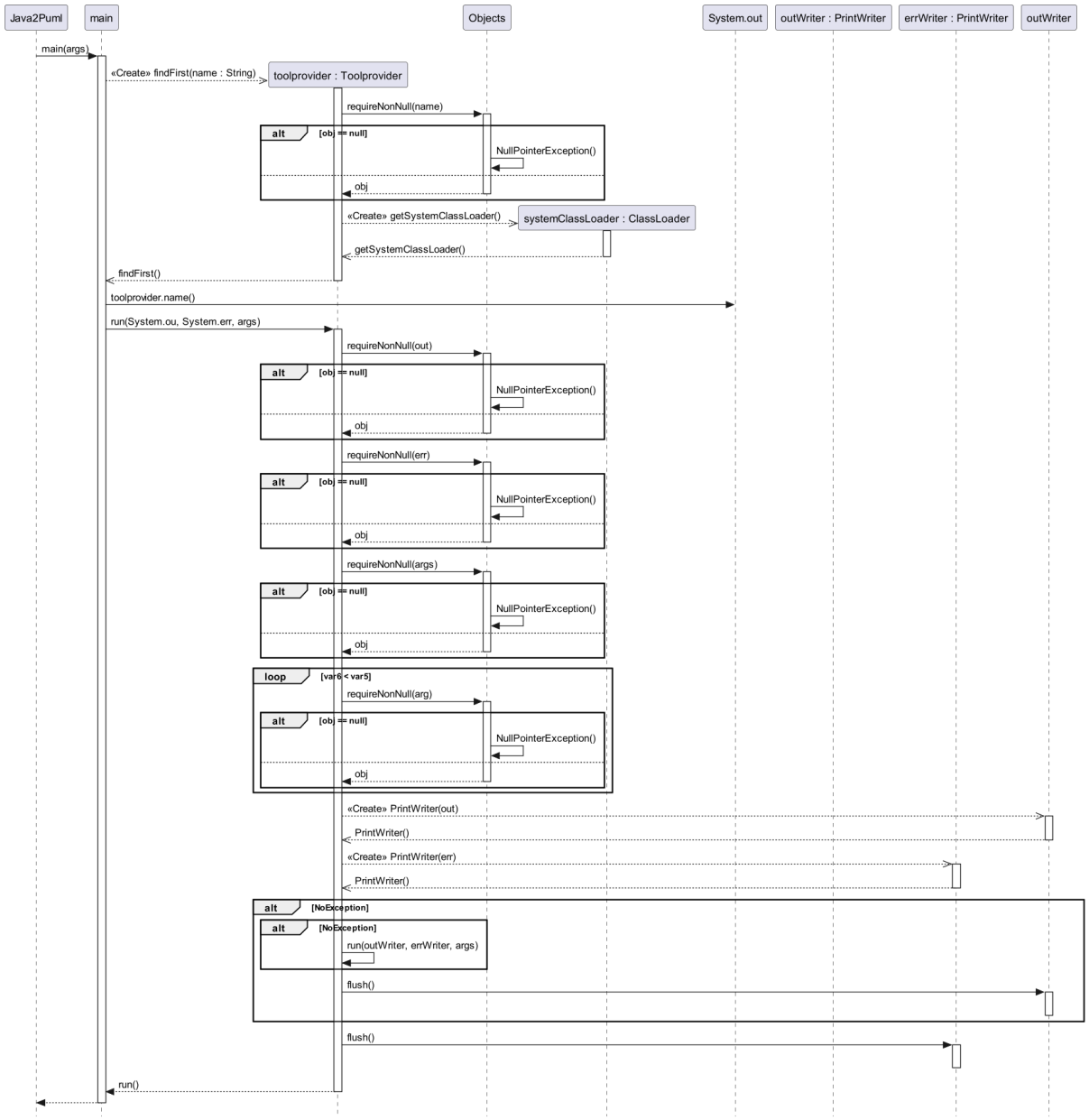


Diagramme de séquence de l'appel du doclet FirstDoclet

Diagramme de classe de conception et d'analyse

Pour ces deux diagrammes, on s'est basé sur le diagramme de séquence fait précédemment en prenant les classes utilisées ainsi qu'en prenant leurs méthodes et variables nécessaires pour leur fonctionnement. La majorité des méthodes venant des classes montrées ne sont pas représentées à cause de leur trop grand nombre ainsi à cause du fait qu'elles sont inutiles dans ce cas de figure.

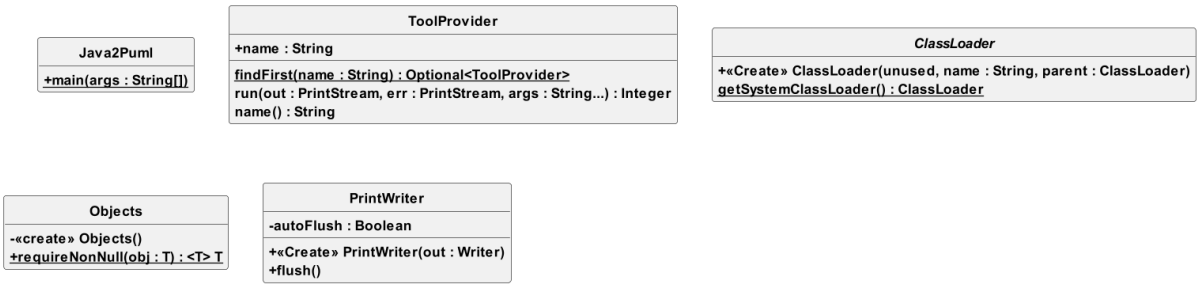


Diagramme de classe de conception

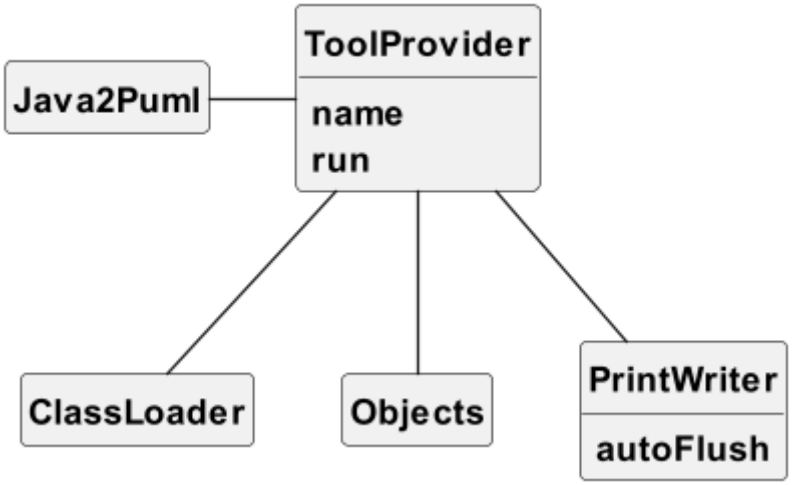
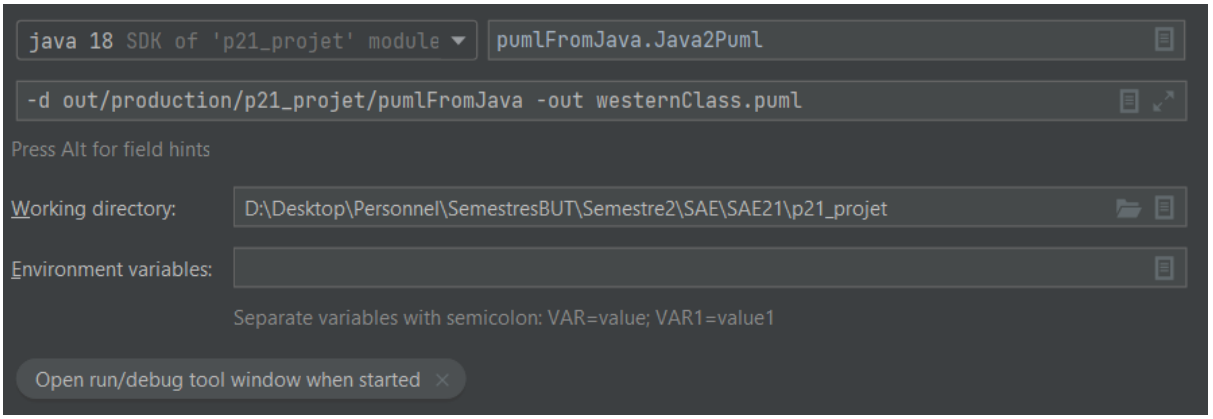


Diagramme de classe d'analyse

Rédaction du code du doclet PumlDoclet

Le doclet PumlDoclet n'a pas pu être fini dans les temps, cependant nous avons réussi à avancer dans cette étape. En premier, nous avons créé un nouveau fichier nommé PumlDoclet.java qui implémente l'interface Doclet et nous avons mis en place les bases des méthodes qui devaient être redéfinies. On a passé beaucoup de temps à essayer de définir les options -out et -d dans la méthode getSupportedOptions(), cependant, comme tout le code du fichier, la majorité risque d'être réécrite puisque les informations ne sont pas sûres.

Avant de rendre ce travail, nous avons fait une nouvelle configuration pour l'exécution du doclet.



Nous nous sommes arrêtés à ce niveau là et nous sommes pour l'instant bloqués sur l'option -out qui n'est pas reconnu.