

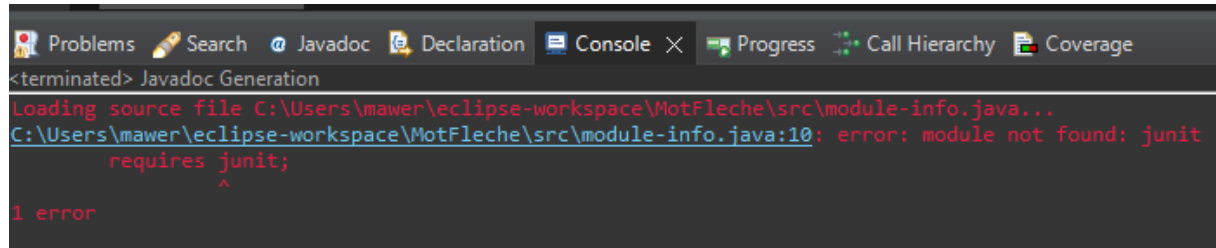
# JAVADOC

Projet JAVA | Mot Fléchés

Yohan Davion

## Erreur lors de la génération des javadocs :

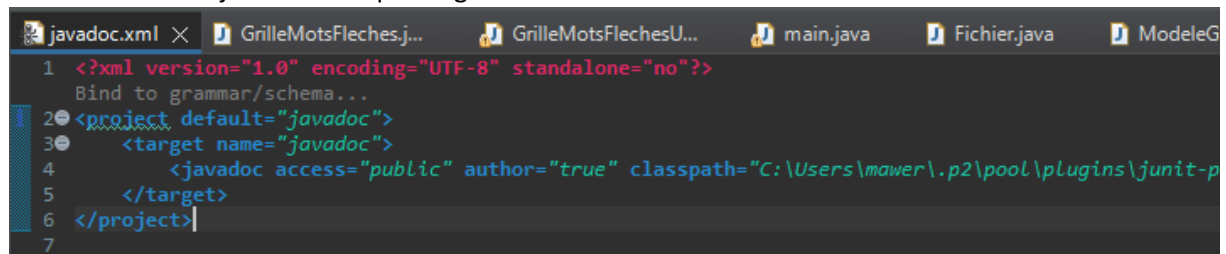
J'ai essayé à plusieurs reprises de générer la javadoc mais à chaque fois une erreur survient :



```
<terminated> Javadoc Generation
Loading source file C:\Users\mawer\workspace\MotFleche\src\module-info.java...
C:\Users\mawer\workspace\MotFleche\src\module-info.java:10: error: module not found: junit
    requires junit;
               ^
1 error
```

Pourtant le module JUnit est bien existant car il est utilisé pour exécuter les tests Unitaires.

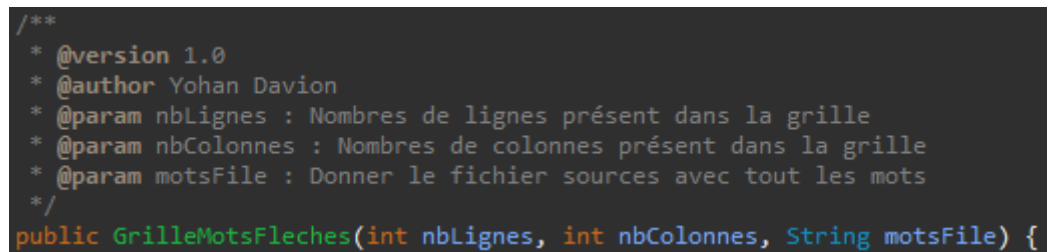
J'ai bien le fichier javadoc.xml qui est généré mais il ne contient rien de bien intéressant



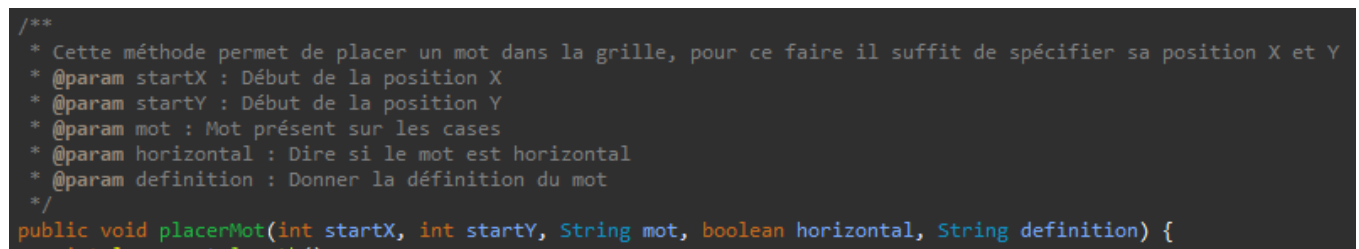
```
javadoc.xml x GrilleMotsFleches.j... GrilleMotsFlechesU... main.java Fichier.java ModeleG
1 <?xml version="1.0" encoding="UTF-8" standalone="no"?>
  Bind to grammar/schema...
2 <project default="javadoc">
3   <target name="javadoc">
4     <javadoc access="public" author="true" classpath="C:\Users\mawer\.p2\pool\plugins\junit-p
5   </target>
6 </project>
7
```

## JAVADOC des méthodes :

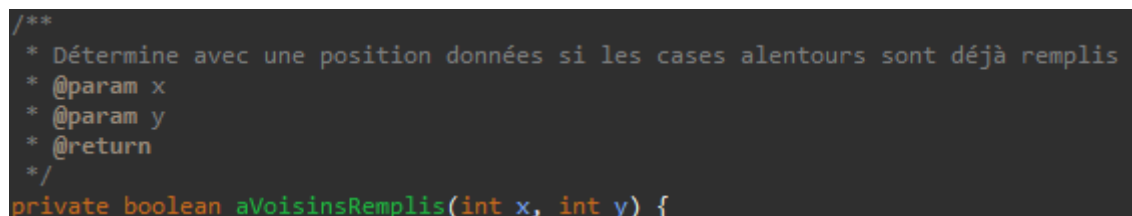
En ce qui concerne la javadoc des méthodes j'ai détailler le fonctionnement des méthodes tout en expliquant pour chaque paramètres sont utilisé (sauf pour les paramètres transparent comme x ou y par exemple).



```
/**
 * @version 1.0
 * @author Yohan Davion
 * @param nbLignes : Nombres de lignes présent dans la grille
 * @param nbColonnes : Nombres de colonnes présent dans la grille
 * @param motsFile : Donner le fichier sources avec tout les mots
 */
public GrilleMotsFleches(int nbLignes, int nbColonnes, String motsFile) {
```



```
/**
 * Cette méthode permet de placer un mot dans la grille, pour ce faire il suffit de spécifier sa position X et Y
 * @param startX : Début de la position X
 * @param startY : Début de la position Y
 * @param mot : Mot présent sur les cases
 * @param horizontal : Dire si le mot est horizontal
 * @param definition : Donner la définition du mot
 */
public void placerMot(int startX, int startY, String mot, boolean horizontal, String definition) {
```



```
/**
 * Détermine avec une position données si les cases alentours sont déjà remplis
 * @param x
 * @param y
 * @return
 */
private boolean aVoisinsRemplis(int x, int y) {
```