

Grille d'auto-évaluation sur des bonnes pratiques de programmation Java (Initial)

<mailto:equipe-gl@telecom-bretagne.eu>

v0.3



Commentaires

-
- Mon *code* est écrit en anglais (noms de variables, méthodes, fonctions, classes, packages, etc.) [P11]
 - Mes *commentaires* sont écrits dans une seule langue (préférentiellement l'anglais.) [P1, Po1, Pj1, P11]
 - Les *commentaires* (javadoc) des méthodes publiques en permettent la réutilisation. [P1]
 - Il y a dans le code des *commentaires* à destination des mainteneurs (hors javadoc). [P1,P5]

Style et conventions

-
- Les indentations sont homogènes. [NB : IDE eclipse peut gérer cela] [P2]
 - La forme des noms respecte le style du langage (Par exemple, en Java : CamelCase pour les variables, CAPITALES pour les constantes) [P2]
 - L'ordre des déclarations est toujours le même (Par exemple : attributs, constructeurs, méthodes publiques puis privées.) [P3]
 - Les noms sont porteurs de sens. [P12]

Modularité

-
- Le paquetage racine ne contient directement aucune classe. [Pj5]
 - Le fichier package.info décrit le contenu et la *responsabilité* de chaque paquetage. [Pj1]
 - Chaque classe a une *responsabilité* de calcul et ne se limite pas à stocker des données. [Po2, Po6]
 - Il n'y a pas d'attribut public (encapsulation.) [Po3]
 - Les méthodes ne sont pas trop longues (ex. 15 lignes de code.) [P6, P7]

Sûreté

-
- Les méthodes se protègent (test des paramètres, levée d'exception.) [P15]
 - Chaque méthode possède au moins un *test*.
 - Les *tests* sont automatisés et s'enchaînent pour vérifier la *non-régression*.

Évolutivité/*Maintenance*

-
- Les classes importées sont nommées explicitement. [Pj8]
 - Il n'y a pas de valeur utilisée sans être déclarée par une constante. [P13]
 - Le code est factorisé. [P4,Po2]