

Projet de POO+DP 2008/2009

Comme chaque année le projet de POO consiste à prendre en main un code existant et de lui apporter des modifications structurelles sans en modifier les aspects fonctionnels.

Cette année le projet à re-concevoir est la partie interface graphique du logiciel unitex.

Dans l'archive Suivante , vous trouverez un projet eclipse complet dans le sous répertoire Src.

<http://www-igm.univ-mlv.fr/~unitex/Unitex2.0.zip>

Première mission

Lire le code

Par groupes de 2

Proposer une reconstruction (refactoring-reengineering):

- une architecture en au moins 5 paquetage de niveau 1 et dans lesquels sont répartis les classes.
- de nouveaux nom pour les classes Java du paquetage fr.umlv.unitex.

Il faut fournir les diagrammes de classes de la nouvelle organisation et expliquer (une à deux pages) les choix et l'architecture choisie.

Votre proposition doit pouvoir répondre aux questions suivantes:

- 1) Pouvoir extraire l'aspect éditeur de graphe afin qu'il puisse être utilisé dans unitex ou dans une applet indépendante.
- 2) Créer deux commandes interface console : grf2png et grf2svg écrites en java permettant de ...
- 3) Une gestion des préférence unique (transversale à tout le logiciel) qui fournisse des préférences globales (que les préférences de chaque langue puisse remplacer).
- 4) Harmonisation des classes graphiques utilisée, avec une explication motivation de chaque choix entre InternalFrame et Jdialog etc... nettoyer le code graphique pour le rendre plus homogène
- 5) Avoir différents FileChooser avec de la mémoire (chaque type de fichier nous emmène dans un répertoire différent chaque demande d'ouverture dans le logiciel est associé un a répertoire différent ...), proposer quelque chose de malin. Le Dessin patern et la qualité de la mise en oeuvre sera évaluée avec attention.

A rendre le 25 février 2009 avant 22h par mail avec comme sujet:[POOM1] avec en attachement une archive zip contenant un répertoire au nom des deux binômes contenant des fichiers au format pdf permettant d'évaluer le travail demandé.

Cella fournira une note sur 20 qui entrera pour 25% de la note final de projet.

Il faut fournir une version du code décrit dans votre document d'architecture. Cette deuxième partie peut être réalisée par groupe de 2 à 4.

Critères d'évaluation

- 1) le fonctionnement de l'interface est conservé, aucunes des fonctionnalités initiales n'a disparue.
- 2) Le système de tests de non régression est pertinent.
- 3) Les Critères précédent sur l'architecture sont effectif dans la solution proposée.

Un certain nombre de réponses peuvent être faites a ce projet :

Ne répondre qu'a la question Applet

Ne répondre qu'a la question de la fabrication de deux fonction en mode console

Ne répondre qu'a la question de l'internationalisation

Ne répondre qu'a la question des FilesChooseur

ne Répondre qu'à la question d'architecture sur l'homogénéisation des classes.

Votre document de présentation doit expliciter ce a quoi vous vous êtes attaqué et ce que vous avez obtenu. Le gain (ou la perte) en terme d'architecture doit être explicitée en une page, penser aux couts de maintenance de la nouvelle solution. Penser par exemple que si vous avec réaliser la commande `grf2png` avec un code radicalement différents de ce qui est utilisé dans l'interface graphique et que la maintenance doit être faites sur deux fois plus de fichier vous aurez raté l'objectif.

Barème:

1pts présentation des objectifs de la nouvelle version

1pts présentation des motivations pour les modifications réalisées

2 pts sur objectifs et motivations.

3 pts sur la qualité de l'intégration de votre travail avec le code existant.

3 pts sur les outils de tests.

3 pts sur la documentation.

10 pts sur la volumétrie du travail (2 pts par options réalisées).

A rendre le 12 avril avant 9h30 par mail avec comme sujet [POOM1] avec en attachement une archive zip contenant un répertoire au nom des membres du groupe.

Ce répertoire contenant un répertoire SrcFinal/ qui contient le projet eclipse final tel que vous souhaiter qu'il soit évalué.

Un répertoire patch/ qui contient le patch permettant d'obtenir votre répertoire final a partir de l'archive initiale.

Un répertoire doc/ qui contient les documents pdf d' accompagnement en particulier un document README.html qui contient la table des matières de l'ensemble des documents de documentation que vous avez créés en plus pour les développeurs suivants.

Pour les tests de non régression il faut qu'il soit placés dans le répertoire SrcFinal/test/ .