

Compte Rendu de réunion

Réunion du : Mardi 27 janvier 2009

Objet : Projet Econet

Participants :

- M2 ALMA
- Encadrants
- Mathieu Aubry
- Pascal André
- Tima Camara
- Jean-Claude Royer
- Aurélia Couvrant
- Mathieu Venisse

Prochaine réunion : Jeudi 29 janvier 2009 au LINA, Salle 113 à 9h15

Pièces Jointes :

- Documentation sur le projet CoCoMe
- Rapport ALMA promo 2008
- Projet testjdt3.jar

Ordre du jour

1. Présentation du diagramme UML de la structure du projet.
2. Règles d'extraction d'un code java.
3. Installation du plugin TESTJDT3.
4. Présentation du projet CoCoMe.

1. Présentation du diagramme UML de la structure du projet

M. Royer nous a présenté les différentes classes utilisées dans le processus d'extraction d'une architecture à composants d'un code java, chacune ayant un rôle précis.

- *TypesTable* : est une hashmap qui représente la table de symboles du code analysé.
- *MyIType* : décrit le type d'une entrée de la table de données.
- *Activator* : décrit le plugin.
- *Communication* : est utilisé dans le processus de découverte de communications entre composants.
- *Provided* : décrit les interfaces fournies d'un composant
- ...

2. Règles d'extraction d'un code java

Nous avons vu par la suite les différentes étapes du processus d'extraction d'une architecture à composants d'un code java réalisées par le plugin testjdt3.

- Dans un premier temps il collecte les différentes entités classes et interfaces existants dans le projet java.
- Puis, identifie les types de données définis à l'intérieur de chaque entité et les stocke dans la table *TypesTable*.
- Analyse la composition d'un composant (pour voir s'il est composite ou pas)
- Regarde les communications entre les différents composants. Si deux composants communiquent, il n'autorise pas dans les méthodes des paramètres de type composant, seuls les datatypes sont autorisés.
- Ensuite il trouve les interfaces fournies en fonction de l'étape précédente...

3. Installation du plugin testjdt3

- Installer le plugin JDT dans eclipse. Vous le trouverez sur le lien suivant.
<http://download.eclipse.org/eclipse/downloads/drops/R-3.4.1-200809111700/index.php>
- Importer le projet testjdt3 dans le workspace et exécuter le avec le bouton *run* (bouton vert). Une nouvelle configuration d'eclipse doit souffrir avec l'apparition du champ *TESTJDT3* dans la barre du menu.

Vous pouvez tester l'extraction d'un code java en cliquant sur l'onglet *visit* de celui ci, puis en fournissant le nom d'un projet java qui existe dans le workspace de cette configuration, ou plus simplement qui apparaît dans la liste des projets java figurant dans la fenêtre d'exploration des projets.

- Pour que le plugin soit visible lors du démarrage de la première configuration d'eclipse sans avoir à lancer une autre configuration :
 - Créer le jar du plugin en cliquant sur le projet TESTJDT3 bouton droit export /Plug-in developpement/ Deployable plug-ins and fragments.
 - Copier le jar obtenu dans la liste des plugins de la configuration de démarrage et redmarraer eclipse.

4. Présentation du projet CoCoMe

Par la suite M. André nous a présenté les morceaux d'Econet, le processus d'extraction d'un code annoté sur lequel des étudiants de la promo ALMA 2008 ont travaillé, puis le projet CoCoMe.

CoCoMe est un projet sur lequel nous allons effectuer nos tests, c'est un projet java qui représente une application d'achat électronique. Son architecture est décrite sous forme de composants.

Nous avons évoqué aussi des points que nous pourrions étudier par la suite, comme le contrôle de cycle entre composants, ainsi que la cohérence sur la communication.

Dans un premier temps, M. Royer nous a demandé de nous concentrer sur le calcul de l'information de la table. Et d'essayer d'élaborer une structure de JDT à travers un diagramme de classes UML.

Pour ceux qui n'étaient pas là, si vous avez des questions à poser vous devez vous adresser à M. Ardourel si celles ci portent sur les plugins, à M. André sur les tests et/ou l'environnement du projet à savoir toute question sur Econet et à M. Royer sur JDT.

Tima Camara