UNIVERSITÉ DE NANTES

Master 1 ALMA

Projet de Conception de Logiciels Extensibles

Documentation Développeur

Étudiants :
Coraline MARIE, Vincent RAVENEAU et Quentin MORICEAU

Encadrant :
Gilles ARDOUREL

30 mars 2014

Table des matières

Préface

Les architectures à plugins sont aujourd'hui considérées comme étant l'avenir des logiciels. En effet, les plugins sont souvent la base d'une architecture logicielle modulaire, comme c'est le cas pour la plate-forme Eclipse et les bundles OSGi.

L'actuel Master Alma de l'Université de Nantes, offre la possibilité aux étudiants d'étudier la *Conception de logiciels extensibles*. Les points abordés dans cette matière présentent plusieurs techniques d'architectures à plugins, permettant de développer des logiciels extensibles.

L'Éditeur d'Armure se trouve être un logiciel extensible réalisé comme projet de fin de semestre du module *Conception de logiciels extensibles*.

Cette documentation a été écrite dans le but de commenter le projet Éditeur d'Armure. Chaque particularité du logiciel y est décrite ainsi que des exemples de manipulations afin de permettre à n'importe quel développeur de pouvoir créer un plugin compatible avec l'Éditeur d'Armure.

Informations générales

L'Éditeur d'Armure est un logiciel obéissant à un scénario composé par les étudiants du Master 1 Alma : un utilisateur quelconque souhaîte créer ou éditer des données, décrivant une armure de protection/combat (telle celle d'Iron Man).

Ce projet, à but pédagogique, a été conçu afin de permettre aux étudiants de parfaire leurs connaissances en *Conception de logiciels extensibles* en créant leur propre architecture à plugins.

1.1 Récupération du code source

Le code source de l'application est disponible sur un dépôt Git hébergé par le site GitHub. Pour le moment, le code source peut être visualisé et télécharger par tout le monde. Il est possible de le récupérer, sous forme d'archive, ou directement par l'intermédiaire de Git.

```
git clone https://github.com/Lerian/ProjetCLE.git
```

1.2 Importation et architecture

L'Éditeur d'Armure est un logiciel conçus en `Java 7` avec l'IDE Eclipse. Pour accéder au code source sans soucis, il est conseillé d'importer au préalable les projets sous Eclipse avant d'y faire des modifications.

Le code source est divisé en plusieurs projets :

- Application projet : plugin principal
- GestionnairePlugins : gestionnaire de plugin indispensable au bon fonctionnement du logiciel.
- CreationArmure : plugin de création
- CreationArmureFichier : plugin de création
- ModificationArmure : plugin de modification de données
- AffichageConsole : plugin d'affichage
- AffichageGraphique : plugin d'affichage

Fonctionnement des plugins

L'Éditeur d'Armure est un logiciel composé d'un gestionnaire de plugin et de plusieurs plugins.

Utilisation des plugins

Tutoriel : comment créer un plugin

Pour créer un nouveau plugin il est conseillé d'utilisé l'IDE Eclipse et d'y importer les projets disponible sur GitHub par la commande :

```
git clone https://github.com/Lerian/ProjetCLE.git
```

Ou en passant par le plugin git sous eclipse.

Une fois les projets correctements importé, vous pouvez commencé la création de votre plugin.

Pour commencé, créez un nouveau projet java à placer dans le dossier ProjetCLE que vous venez d'importer.

Créez ensuite une classe qui sera votre classe principal. Cette classe devra implémenter l'interface IPlugin du projet GestionnairePlugins.

Si vous souhaitez creez un plugin pour le logiciel d'édition d'armure, vous devez choisir le type de plugin à creez :

- Créateur : implémenter l'interface ICreateur du projet ArmorEditor.
- Modificateur : implémenter l'interface IModificateur du projet ArmorEditor.
- Afficheur : implémenter l'interface IAfficheur du projet ArmorEditor.

Si des erreurs apparaisse ajouter les projets GestionnairePlugins et ArmorEditor à votre build path.

Vous pouvez maintenant créer votre plugin en implémentant les méthodes des interfaces.

La méthode run est executé lors de l'ajout de votre plugin au lancement de la plateforme.
La méthode type doit retourné le type du plugin.
La méthode receiveProperties fournie un objet Properties qui contient : "pathToHome", "pathToInit" ainsi que les informations qui peuvent vous être utils à mettre dans un fichier .init à créer dans le dossier resources de votre projet.

En ce qui concerne le programme d'édition d'armure, si vous souhaitez créer

un afficheur, vous devez mettre le code principal de votre affichage dans la méthode affiche().

Pour le modificateur vous devez implementer les différentes méthodes qui permettents de changé les valeurs des armes et armures.

Et enfin pour le créateur, vous devez construire votre armure dans la méthode cree().

Il vous suffit ensuite de déclarer votre plugin, si vous construiser un plugin général, il est à déclaré dans le fichier init du gestionnaire de plugin, par contre si vous souhaitez construire un nouveau composant pour l'éditeur d'armure, vous devez déclarer le plugin comme étant une dépendance de l'éditeur d'armure et donc le décalré dans armorEditor.init en suivant la syntaxe décrite.

Voila vous avez créer votre plugin, félicitation !