

Projet Java

Synchroniseur de dossiers

Nom | Date | //

# Introduction

Cahier des charges :

Développé un outil permettant de synchroniser le contenue de deux dossiers distincts, c’est-à-dire ;

Soient D et D’ deux dossiers d’un système de fichier, si une modification est faite dans le dossiers D celle-ci est répété dans le dossier D’, de même si une modification est faite dans le dossier D’ celle-ci est répéter dans le dossiers D. Une modification se caractérise par la création, la suppression ou la modification d’un fichier ou d’un dossier.

De plus, la modification doit se déclencher automatiquement sans que l’utilisateur de l’outil n’ait à le faire, pour faciliter le développement elle doit dans un premier temps se déclencher si une modification est réaliser dans D ou D’ puis dans un second temps automatiquement.

En plus de ces spécification, l’outils devra aussi pouvoir synchroniser deux dossiers originaires de deux machines différentes, et les interactions avec l’outils devront pouvoir se réaliser grâce à un interface graphique utilisateur.

N.B. :

L’outil doit être utilisable et robuste et devra comporter un manuel d’utilisation et intégré une fonctionnalité additionnelle non détaillé ci-dessus.

## Interprétation.

Après avoir étudiée le cahier des charges nous nous sommes mis d’accord pour développer une structure de données qui réalisera, pour résumer de façon grossière, les opérations suivantes :

Soient deux dossiers D et D’ d’un même système de fichiers. L’outil que nous allons développer identifiera l’un des dossiers comme étant le dossier source et l’autre comme étant le dossier cible (choix fait par l’utilisateur), puis synchronisera le dossiers cible sur le dossier source (notons que source et cible pourront être inversé à tout moment au choix de l’utilisateur) mais l’inverse ne sera pas possible. Pour ce faire l’outils devra analyser le contenue de la source et de la cible puis ajouter dans la cible les dossiers étant dans la source mais dans celle-ci, supprimer ceux étant dans celle-ci mais pas dans la source et mettre à jours les dossiers qu’elle partage avec la cible mais adoptant les modifications de la source.

Pour conclure, on a décider de se donner une comme restriction que la synchronisation devra se faire seulement dans un sens : d’un dossier source vers un dossier cible. Se que revient à pouvoir la faire dans les deux sens en permutant source et cible selon ce que l’on veut.

Sommaire

# Diagramme de cas d’utilisation …………………………………… n

# Diagramme des classes ……………………………………………… n

# Diagramme de séquence …………………………………………….. n

# Diagramme d’état ………………………………………………………. n

# Diagramme des classes détaillé et restriction OCL………. n

# Diagramme d’objet ……………………………………………………… n

# // A voir //Diagramme de communications …………………n

# Diagramme de déploiement ………………………………………. n

# Difficultés rencontrées …………………………………………………. n

# Notes à propos du manuel d’utilisation ………………………. n

# Conclusion …………………………………………………………………… n

# Annexe et bibliographie ………………………………………………. n

Diagramme de cas d’utilisation

Une image contenant diagramme

Description générée automatiquement

Commentaires :

Diagramme des classes

Une image contenant diagramme

Description générée automatiquement

Commentaires :

Diagramme de séquence

Une image contenant diagramme

Description générée automatiquement

Une image contenant diagramme

Description générée automatiquement

Une image contenant diagramme

Description générée automatiquement

Diagramme d’état

Diagramme des classes détaillé et restriction OCL

Diagramme d’objet

Diagramme de déploiement