



Projet	<i>Ateliers de la formation «Mettre en œuvre une démarche d'intégration continue»</i>
Auteur(s)	<i>Hafid OUAHIDI</i>
Date	<i>27/10/2014</i>

Référence du Document	<i>UNI-WORKSHOP-DIN</i>
Document Version	<i>V1.0</i>
État de document	<i>Initial</i>
Remarques	



UNIS POUR L'INNOVATION

I - Introduction

Le présent document présente les ateliers de mise en pratique de la formation «Mettre en œuvre une démarche d'intégration continue».

L'ensemble des ateliers concernent la gestion du module de gestion de l'inventaire de l'application «UniLibraryWeb» dont le cahier de charge est présenté ci-dessous.

II - Cahier de charge «UNI Library»

Un établissement scolaire souhaite se doter d'une nouvelle application pour moderniser son système de gestion de la bibliothèque. Ce nouveau système doit assurer :

- * La réservation des médias par les différents lecteurs,
- * L'enregistrement des emprunts,
- * La gestion de l'inventaire,
- * La validation des demandes d'achats des nouveaux médias,
- * La gestion des utilisateurs.

Les médias de la bibliothèque sont organisées en différentes catégories hiérarchisées. Le nombre d'exemplaires disponibles de chaque média dépend des réservations, l'achat de nouveaux, la perte ou l'état des exemplaires.

Un lecteur peut réserver différents types de médias (Livre , CD, DVD...) :

- Scénario I : Si le lecteur connaît la référence exacte du support, il peut le réserver en indiquant la référence et la période d'emprunt (La date ne peut pas dépasser 2 semaines).
- Scénario II : Le cas contraire, le lecteur peut effectuer une recherche multi-critères pour identifier le ou les supports souhaités.

Après la phase de réservation, l'utilisateur s'adresse au personnel de la bibliothèque pour récupérer le ou les différents médias. Le personnel s'assure lors de l'enregistrement de l'emprunt de respect du règlement de la bibliothèque (Par exemple : la limite d'emprunt est fixée à 3 supports par période 2 semaines, ...)

Les lecteurs peuvent envoyer des propositions d'achats de nouveaux médias qui sont validées par le responsable de la bibliothèque. Ce dernier refuse, accepte et priorise les demandes en respectant le budget et la stratégie de l'établissement.

1. Atelier I : Mise en place de l'environnement

La procédure d'installation de l'environnement est la suivante :

1. Création de l'arborescence suivante

JavaEE

- |__ IDE : Installation Eclipse Helios JEE
- |
- |__ Softwares : Installation des logiciels (JavaSE, Maven Ant,...)
- |
- |__ Servers : Installation des serveurs (Tomcat ...)
- |
- |__ Workspace : Espace de travail destiné aux projets
- |
- |__ Downloads : Copies des packages à installer. Les packages doivent être récupérés du même répertoire sur le poste formateur
- |
- |__ Distribution : Dossier contenant l'ensemble des packages nécessaires à la préparation des ateliers

Note : Dans la suite du document la variable `JavaEE_HOME` référence le répertoire d'installation du répertoire `JavaEE`

2. Installation de la plateforme JavaSE

Après installation du JDK dans le répertoire `Softwares`, il faut ajouter les variables d'environnement suivantes :

- `JAVA_HOME = ../JavaEE/Softwares/Java/jdk1.7.x`
- `PATH=$PATH:$JAVA_HOME/bin` ou `PATH=... ; %JAVA_HOME%/bin` sous Windows

3. Installation d'Eclipse : Il suffit de décompresser le zip `eclipse-jee-luna.xx.zip` sous le répertoire `JavaEE/IDE`

4. Installation du serveur tomcat-7.xx : Il suffit à l'image d'eclipse de décompresser l'archive de tomcat sous le répertoire `JavaEE/Servers`

5. Afin de tester l'installation de tomcat, tapez dans la commande suivante :

`./$CATALINA_HOME/bin/catalina.sh start` ou `CATALINA_HOME/bin/catalina.bat start` sous windows (`CATALINA_HOME` est le répertoire d'installation du serveur tomcat)

Le démarrage normal du serveur se termine avec le message suivant :

```
27 oct. 2014 23:46:23 org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-bio-8080"]
27 oct. 2014 23:46:23 org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["ajp-bio-8009"]
27 oct. 2014 23:46:23 org.apache.catalina.startup.Catalina start
INFO: Server startup in 6035 ms
```

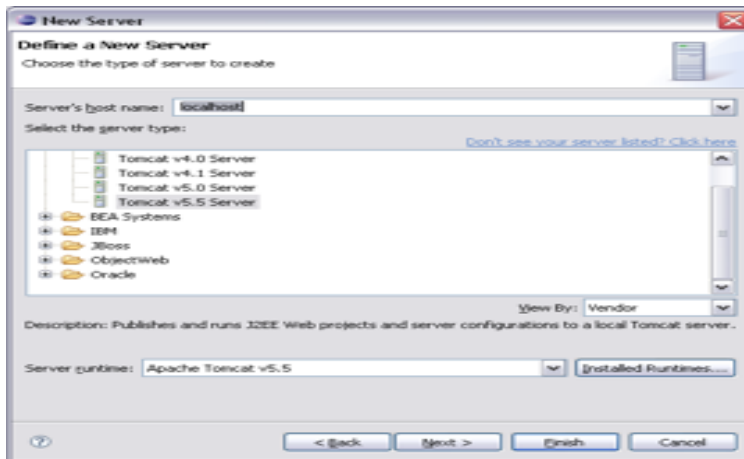
Note : Pour arrêter le serveur tomcat, tapez catalina.bat stop

Ajoutez un utilisateur au domaine de sécurité par défaut représenté par le fichier tomcat-users.xml :

login : admin

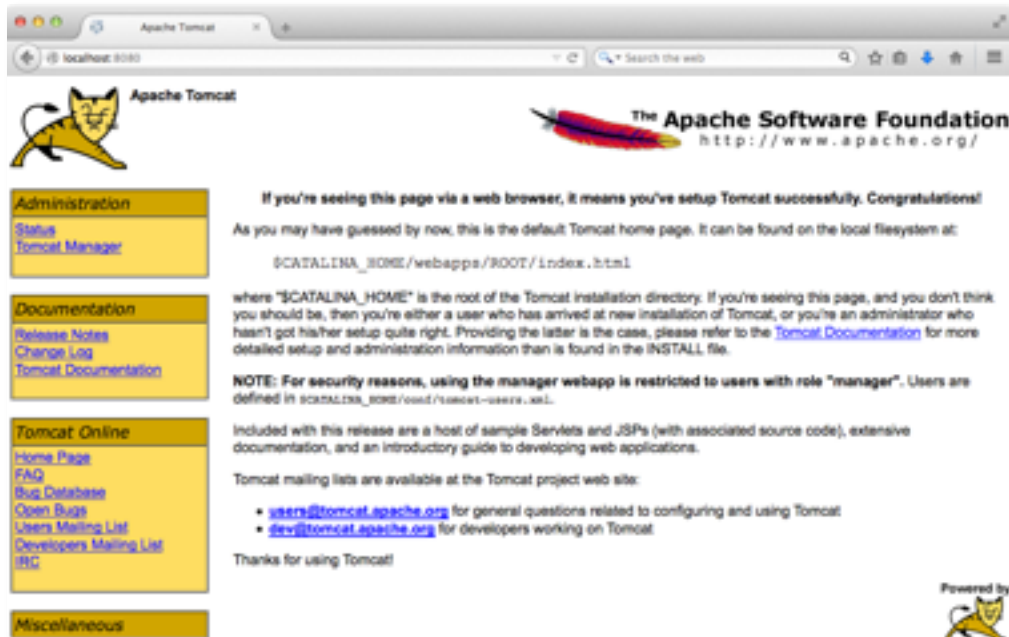
password : admin

6. Configuration d'une instance tomcat sous eclipse
 - a. Ouvrir les préférences d'eclipse (menu Window->preferences)
 - b. Sélectionnez la rubrique Serveurs
 - c. Ajoutez un nouveau serveur de type Apache Tomcat-6.x
 - d. Cochez la case créer une instance du serveur
 - e. Sélectionnez le chemin d'installation du serveur Tomcat-6.x
 - f. Précisez la bonne version du JDK



- g. Puis valider la configuration

- h. Après démarrage du tomcat, la page d'accueil doit se présenter comme la figure ci-dessus.



7. Préparation de la base de données

- Démarrez le service MySQL si ce n'ai pas le cas
- Sous une console (ou une fenêtre Dos sous Windows), lancez le client mysql

```
MYSQL_HOME/bin/mysql -u root
```

- puis tapez la commande ci-dessous pour créer la base de données *uni_library_db*, créer le schéma de la base et importer les exemples fournis de gestion de l'inventaire de la bibliothèque «Uni Library»

```
mysql>source $JavaEE_HOME/Distribution/uni_library_db.sql
```

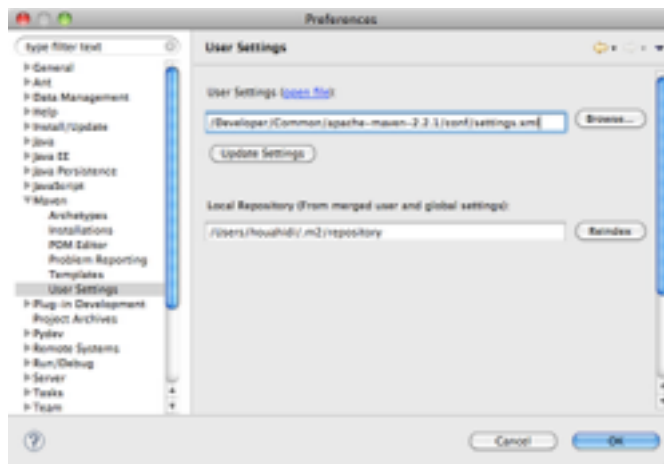
\$JAVA_HOME est le répertoire racine de l'environnement de travail (voir le premier schéma)

- Copiez le pilote MySQL «*mysql-connector-java-5.x.x.jar*» qui se trouve dans le répertoire *\$JavaEE_HOME/Distribution* dans le dossier des bibliothèques du serveur.

```
«$TOMCAT_HOME/lib»
```

8. Installation de MAVEN

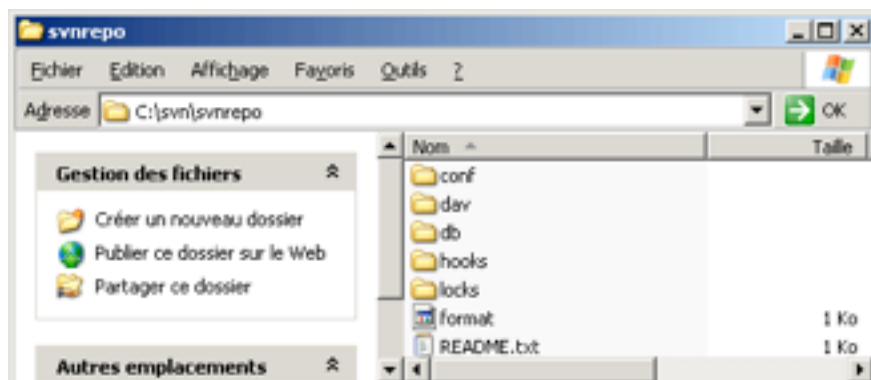
- a. Décompressez le zip de Maven dans le répertoire \$JavaEE_HOME/Softwares/apache-maven-2.x.x
- b. Modifiez les préférences du plugin Maven à partir des préférences d'eclipse (menu Help->preferences->Maven) :
 - 1) Sélectionnez la rubrique Installation
 - 2) Ajoutez le package Maven installé précédemment. le résultat est illustré par la figure.
- 3) Dans la rubrique «User Setting», Indiquez le chemin du fichier de configuration de Maven (\$MAVEN_HOME/conf/settings.conf (voir figure ci-dessous)



2. Atelier II : Mise en oeuvre de la gestion des versions avec SVN

L'objectif de l'exercice est d'illustrer la gestion des versions avec SVN :

- 1) Exécutez le programme svn-x.x.x-setup.exe
- 2) Ajoutez la variable d'environnement SVN_EDITOR=c:/windows/nodepad.exe
- 3) Créez un répertoire qui va contenir le dépôt, par exemple C:\svn\svnrepo
- 4) Exécutez la commande svnadmin create c:\svn\svnrepo



5) Activez le contrôle d'accès en supprimant les # dans le fichier svnservice.conf

```
# [general]
...
anon-access = read
auth-access = write
...
password-db = passwd
```

6) Ajoutez plusieurs utilisateurs dans le fichier passwd

```
# [users]
...
usera= usera
userb=userb
```

7) Démarrez le serveur svn avec la commande suivante : `svnserve --daemon --root c:\svn\svnrepo`

8) Créez un projet UniLibraryWeb avec la commande <svn://localhost/UniLibraryWeb>

- 1) Ajoutez un commentaire : création de projet UniLibraryWeb
- 2) Validez la modification en utilisant un compte parmi les utilisateurs ajoutés

9) Installez le plugin eclipse en utilisant l'url <http://subclipse.tigris.org/>

10) Importez l'archive war UniLibraryWeb.war dans eclipse

11) Apportez des modifications à l'application pour mettre en oeuvre les différentes actions svn :

- 1) update
- 2) commit
- 3) ajout de tags
- 4) Création de branches
- 5) ...

3. Atelier III : Mise en oeuvre de la gestion des tâches avec Ant et Maven

L'objectif de l'exercice est d'illustrer la gestion des tâches avec les frameworks Ant et Maven de la communauté Apache :

Ant :

- 1) Ajoutez le fichier de définition des tâches build.xml au projet
- 2) Définissez les tâches suivantes
 - 1) compilation
 - 2) Génération de l'archive war

3) Déploiement

4) ...

Maven :

- 1) Transformez le projet UniLibraryWeb en un projet Web maven
- 2) Déclarez toute les bibliothèques de l'application sous forme de dépendances Maven
- 3) Testes les taches maven suivantes :
 - 1) compilation
 - 2) Génération de l'archive
 - 3) Déploiement
 - 4) ...

4. Atelier IV : Mise en oeuvre de l'intégration continue avec Jenkins

L'objectif de l'exercice est d'illustrer la mise en oeuvre de l'intégration continue avec la solution Jenkins.

- 1) Installez l'archive war de jenkins dans le répertoire de déploiement de Tomcat
- 2) Démarrez le serveur en utilisant le script catalina.bat start
- 3) Accédez à la page d'accueil de jenkins <http://localhost:8080/jenkins>
- 4) Paramétrez sous jenkins les éléments suivants :
 - 1) JAVA_HOME
 - 2) ANT
 - 3) MAVEN
- 5) Configurez la sécurité globale Jenkins et les comptes utilisateurs
- 6) Ajoutez les jobs suivants:
 - 1) compilation
 - 2) Génération de l'archive
 - 3) Déploiement

5. Atelier V : Mise en oeuvre des tests unitaires et d'intégration

L'objectif de l'exercice est d'implémentez les tests avec le framework Junit et HttpUnit.

- 1) Implémentez un exemple de tests de la couche d'accès aux données (CatégorieDaoTestCase) avec Junit
- 2) Implémentez un exemple de tests fonctionnels avec HttpUnit (gestion des catégories)
- 3) Exécutez un job Jenkins pour l'exécution des tests avec Maven