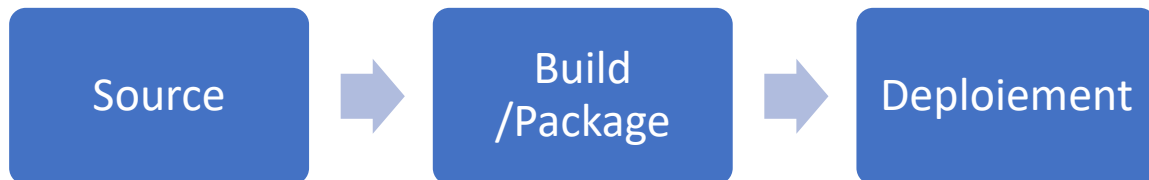


TP 6

Déployer une application Web Maven dans un serveur Tomcat



Objectifs

Le propos de cet atelier est de se familiariser avec l'interface graphique d'un serveur d'intégration contenue : Jenkins

Etape 1

Mise en place du serveur Tomcat sur une autre Machine Linux

- 1- Télécharger le package Tomcat depuis <https://tomcat.apache.org/download-80.cgi>

```
wget https://downloads.apache.org/tomcat/tomcat-8/v8.5.100/bin/apache-tomcat-8.5.100.tar.gz
```

```
sudo tar xvfz apache-tomcat-8.5.100.tar.gz -C /opt
```

```
sudo mv /opt/apache-tomcat-8.5.100/ /opt/tomcat8
```

- 2- Donnez les autorisations d'exécution à startup.sh et shutdown.sh qui sont sous bin

```
sudo chmod +x /opt/tomcat8/bin/startup.sh
```

```
sudo chmod +x /opt/tomcat8/bin/shutdown.sh
```

3- Créez des liens pour Tomcat **startup.sh** et **shutdown.sh**

```
sudo ln -s /opt/tomcat8/bin/startup.sh /usr/local/bin/tomcatup
```

```
sudo ln -s /opt/tomcat8/bin/shutdown.sh /usr/local/bin/tomcatdown
```

4- Changer le numéro de port

NB : Le serveur Jenkins et le serveur Tomcat, écoutent par défaut sur le numéro de Port 8080.

Si vous avez cloner la VM qui contient déjà une installation de Jenkins pour installer Tomcat, vous devez changer l'un de deux. Dans ce cas, il est recommandé de changer le numéro de port du serveur Tomcat.

- Pour changement du numéro de Port du serveur Tomcat, il faut éditez le fichier **/opt/tomcat8/conf/server.xml**

```
sudo gedit /opt/tomcat8/conf/server.xml
```

- Cherchez le bloc `<Connector port="8080"` et Changez le numéro de port 8080, par un autre port non utilisé.

```
<Connector port="8080" protocol="HTTP/1.1"
            connectionTimeout="20000"
            redirectPort="8443" />
<!-- A "Connector" using the shared thread pool-->
<!--
<Connector executor="tomcatThreadPool"
            port="8080" protocol="HTTP/1.1"
            connectionTimeout="20000"
            redirectPort="8443" />
```

- Par défaut le serveur Tomcat n'est accessible que depuis local (localhost/127.0.0.1) pour la gestion des applications. Il faut désactiver cette restriction d'accès pour pouvoir l'utiliser depuis l'extérieure. Pour la désactiver :

- Localisez les fichiers de configuration

```
sudo find / -name context.xml
```

```

hassen@hassen-virtual-machine: ~
Fichier Édition Affichage Rechercher Terminal Aide
hassen@hassen-virtual-machine:~$ sudo find / -name context.xml
/opt/tomcat8/conf/context.xml
/opt/tomcat8/webapps/host-manager/META-INF/context.xml
/opt/tomcat8/webapps/examples/META-INF/context.xml
/opt/tomcat8/webapps/manager/META-INF/context.xml
find: '/run/user/1000/gvfs': Permission non accordée
/home/hassen/Test/mavencalculatorwebapp/src/main/webapp/META-INF/context.xml
hassen@hassen-virtual-machine:~$

```

- Commentez les lignes contenant le champs « ClassName »

Fichier /opt/tomcat8/webapps/manager/META-INF/context.xml

```

...
<!--Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /-->
...

```

Fichier /opt/tomcat8/webapps/host-manager/META-INF/context.xml

```

...
<!--Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /-->
...

```

- Puis, Attribuez les autorisations nécessaires pour gérer le serveur Tomcat

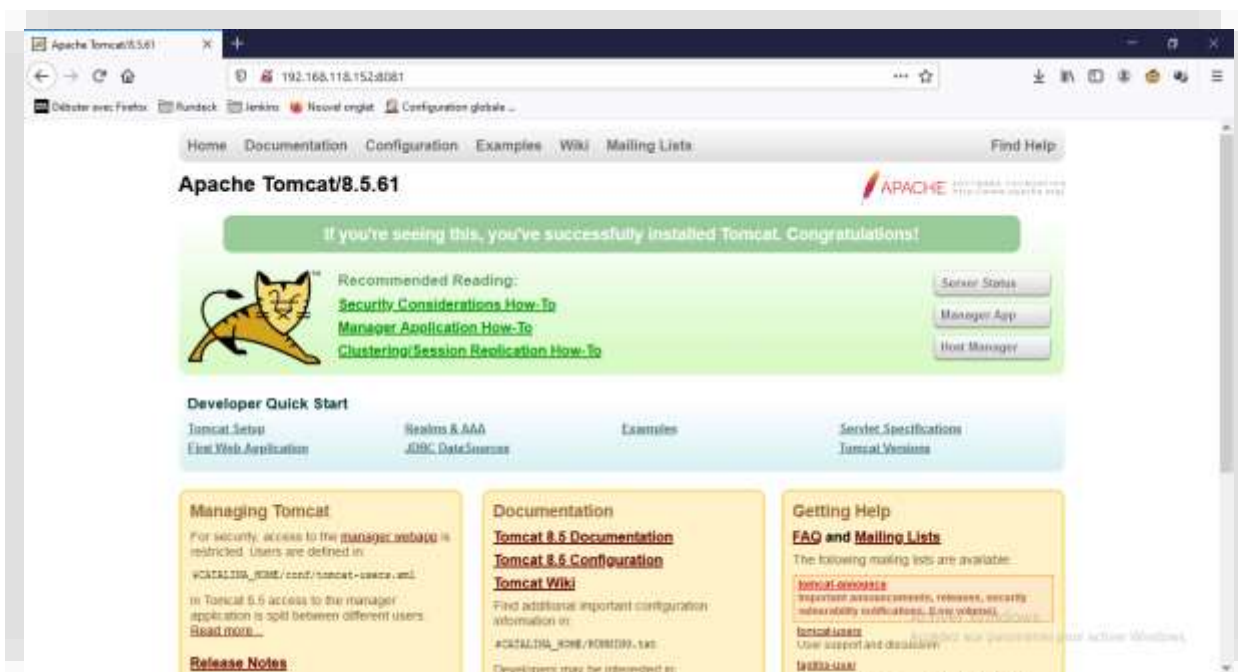
```
sudo nano /opt/tomcat8/conf/tomcat-users.xml
```

- Ajoutez les lignes suivantes, vérifiez bien les mots de passe, vous allez les utiliser ultérieurement dans Tomcat et Jenkins.

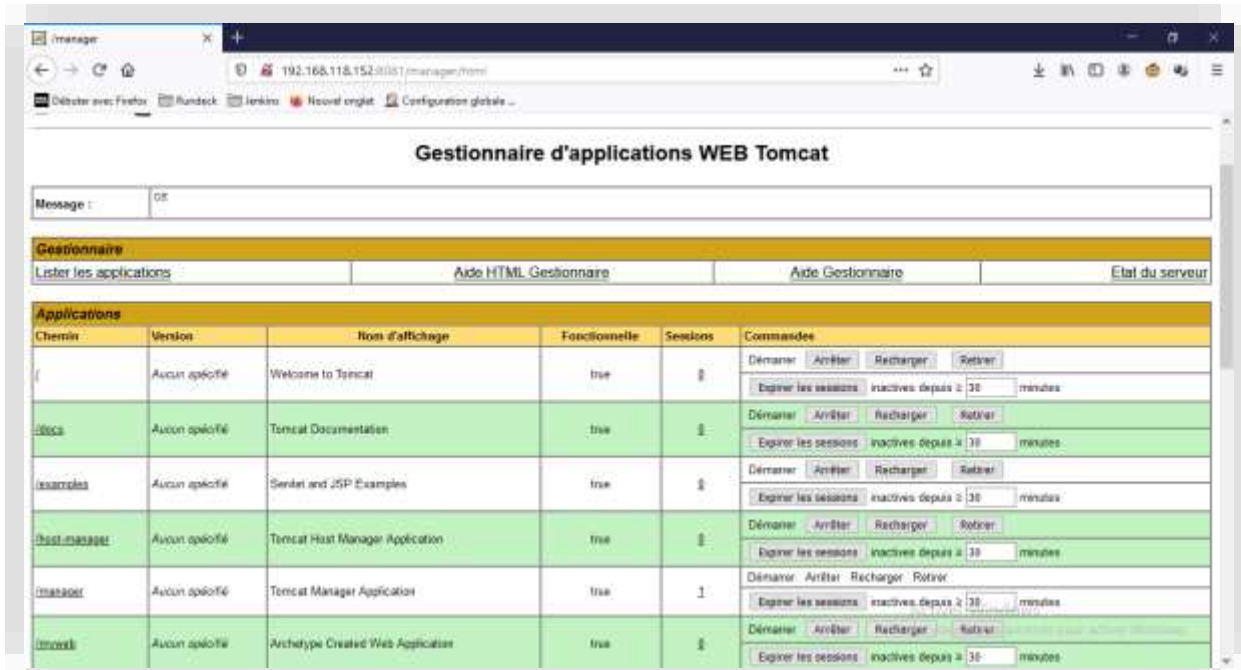
```
<tomcat-users>
...
<role rolename="manager-gui"/>
<role rolename="manager-script"/>
<role rolename="manager-jmx"/>
<role rolename="manager-status"/>
<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx,
manager-status"/>
<user username="deployer" password="deployer" roles="manager-script"/>
<user username="tomcat" password="s3cret" roles="manager-gui"/>

</tomcat-users>
```

- Vérifiez le bon fonctionnement de votre serveur Tomcat, Vérifiez que vous avez les autorisations pour déployer des applications sur le serveur.



- Cliquez sur « Manage App », Vous devez avoir la fenêtre suivante si vous avez bien configuré votre serveur.

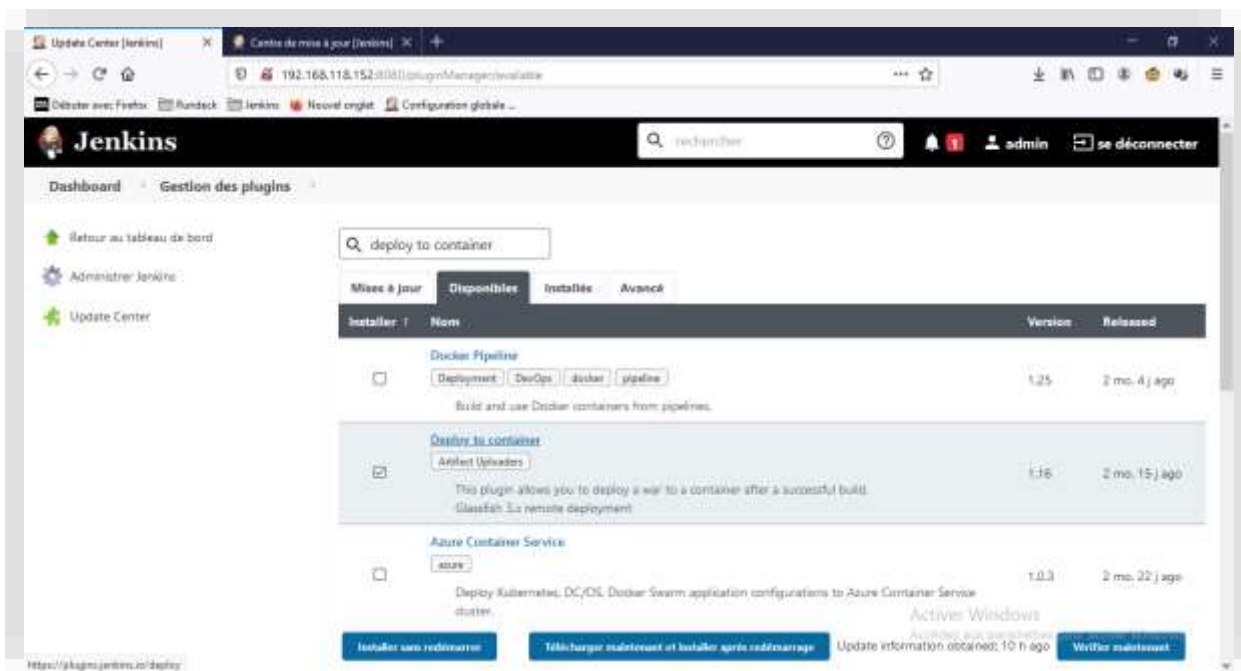


Etape 2

Installation du plugin « deploy to container »

1- Installation du plugin « deploy to container »

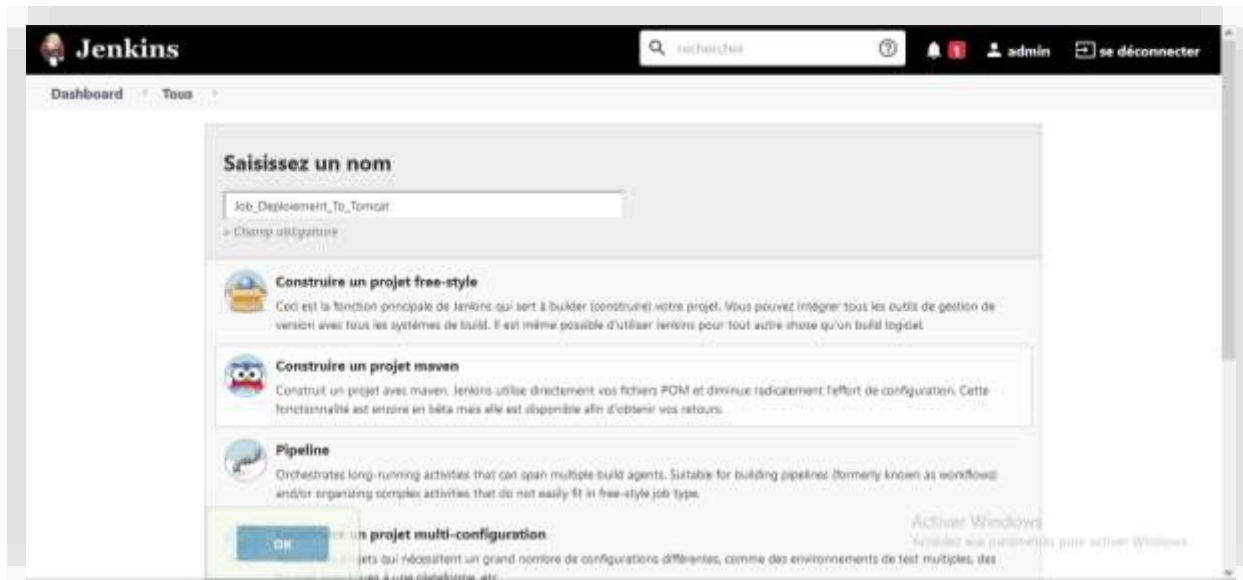
- Administrer Jenkins → Gestion de plugins → Disponibles → Deploy to container



Etape 3

Création du job de déploiement

1- Créez un nouveau Job de type « Projet Maven »



- Configuration le projet selon les instructions suivantes :

General

- **Description** : Ceci est un Projet de déploiement d'une application web JAVA Maven

Gestion de code source : Git

- **Repository URL**: Spécifiez le lien de votre repository sur Github ou Gitlab.
- **Credentials**: si votre projet est privé, créer votre credential. Sinon laissez vide.
- **Branch Specifier**:
 - **Pour Github** : main
 - **Pour Gitlab** : master

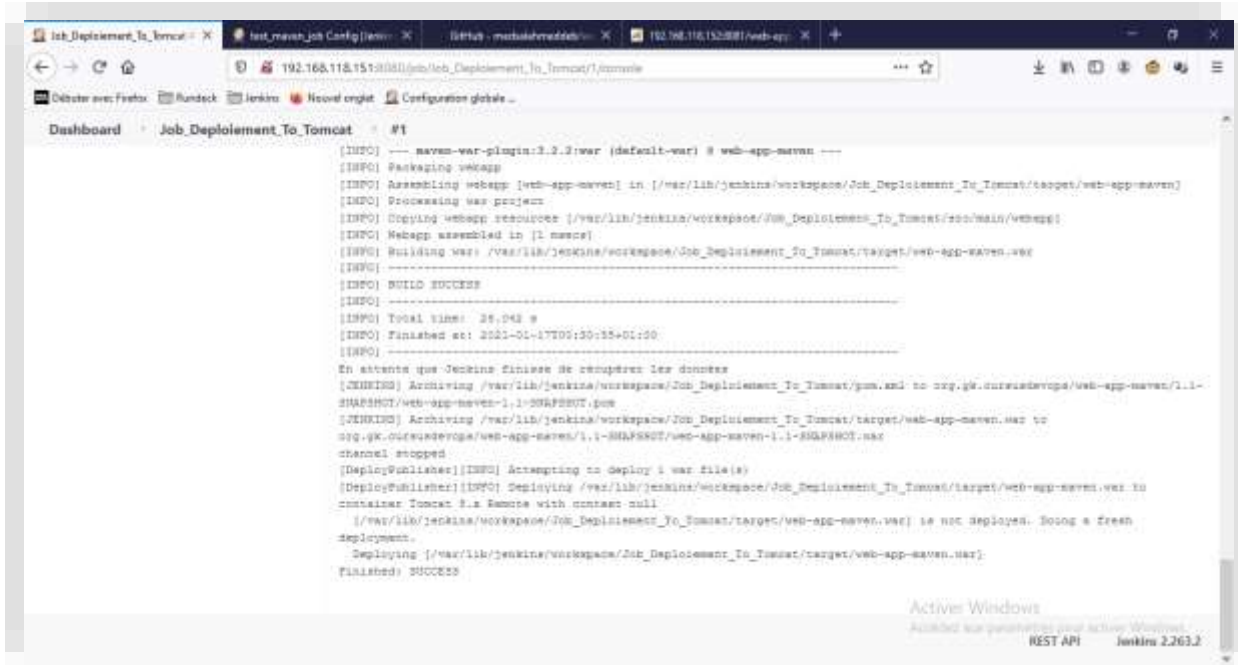
Build

- **POM Racine** : pom.xml
- **Goals et options**: clean install package

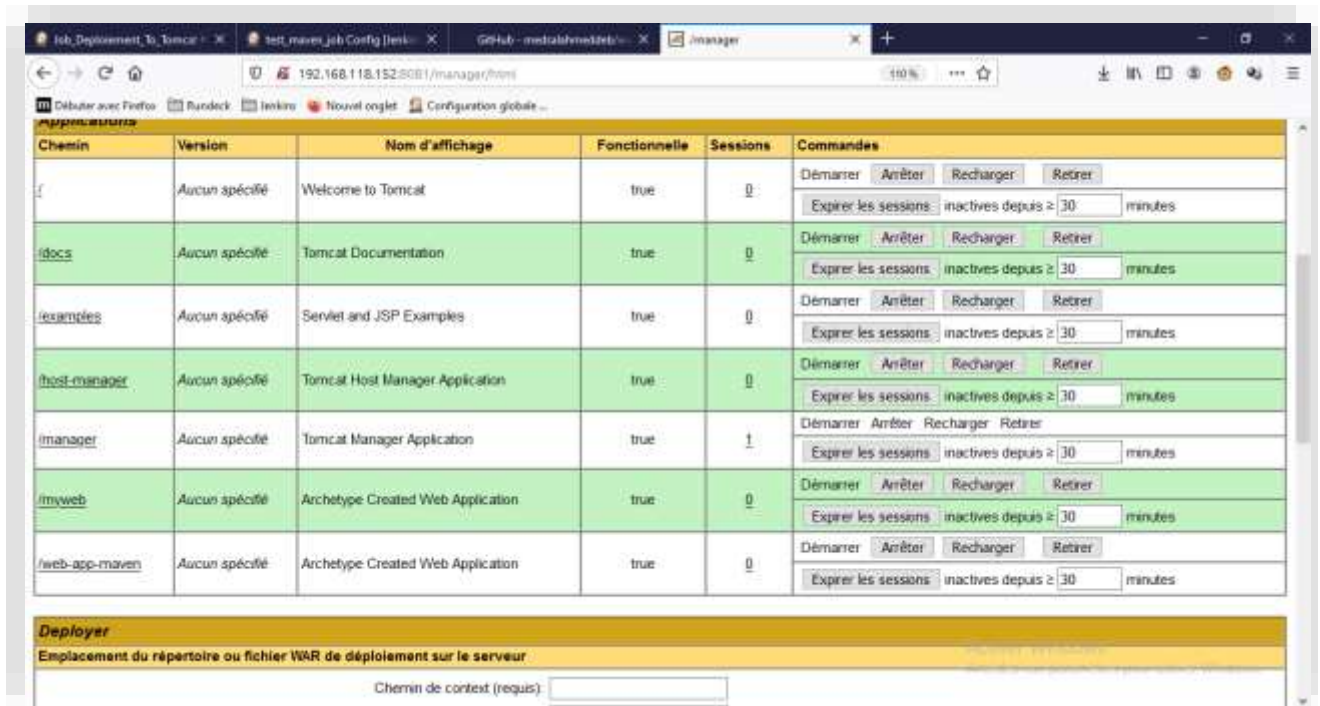
Actions à la suite du build : Deploy war/ear Container

- **Fichiers WAR/EAR**: saisir le nom de l'artifact ou bien `**/*.war`
- **Context path**: laissez vide
- **Containers**: Containers de type Tomcat version 8

- **Credentials** : ajoutez un nouveau credential pour l'utilisateur « deployer »
- **Tomcat URL** : <http://192.168.1.142:8081>
- Lancez le job



- Vérifiez que votre application est bien déployée sur le serveur Tomcat



- Accédez à votre projet : <http://192.168.1.142:8081/WebApp>

