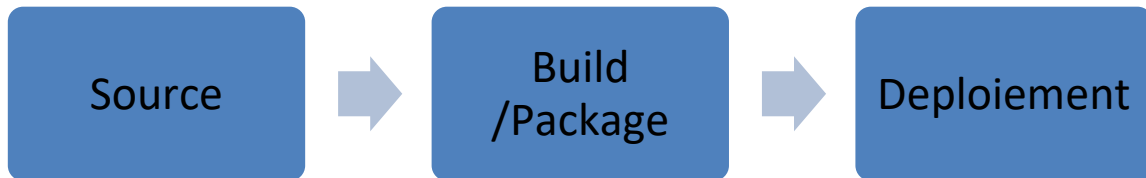


Déployer une application Web Maven dans un serveur Tomcat



Etape 1 : Mise en place du serveur Tomcat sur une deuxième Machine Linux

Installation du Java

- Nous utiliserons open java pour notre démo, Obtenez la dernière version de <http://openjdk.java.net/install/>

```
yum install java-1.8*  
#yum -y install java-1.8.0-openjdk
```

- Configuration de la variable d'environnement JAVA_HOME
 - Déterminez le bon emplacement de la version JAVA8.

```
find /usr/lib/jvm/java-1.8* | head -n 3
```

- Créez un fichier appelé java.sh dans le répertoire /etc/profile.d/

```
vi /etc/profile.d/java.sh
```

- Ajoutez le contenu suivant

```
#!/bin/bash  
export JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.332.b09-1.e17_9.x86_64  
export PATH=$PATH:$JAVA_HOME/bin
```

NB: vous devez changer le chemin du java /usr/lib/jvm/java-1.8.0-*****x86_64. Par le chemin généré dans la commande précédente.

- Enregistrez et fermez le fichier. Rendez-le exécutable à l'aide de la commande suivante.

```
chmod +x /etc/profile.d/java.sh
```

- Ensuite, définissez les variables d'environnement de manière permanente en exécutant la commande suivante :

```
source /etc/profile.d/java.sh
```

- Maintenant, vérifiez la version de fourni en utilisant la commande :

```
java -version
```

Installation de Tomcat

- Téléchargez les packages Tomcat depuis <https://tomcat.apache.org/download-80.cgi>

```
[root@worker-node2 ~]# wget https://downloads.apache.org/tomcat/tomcat-8/v8.5.78/bin/apache-tomcat-8.5.78.tar.gz
[root@worker-node2 ~]# sudo tar xvfz apache-tomcat-8.5.78.tar.gz -C /opt
[root@worker-node2 ~]# sudo mv /opt/apache-tomcat-8.5.78/ /opt/tomcat8
```

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

```
[root@worker-node2 ~]# sudo chmod +x /opt/tomcat8/bin/startup.sh
[root@worker-node2 ~]# sudo chmod +x /opt/tomcat8/bin/shutdown.sh
```

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

```
[root@worker-node2 ~]# sudo ln -s /opt/tomcat8/bin/startup.sh /usr/bin/tomcatup
[root@worker-node2 ~]# sudo ln -s /opt/tomcat8/bin/shutdown.sh /usr/bin/tomcatdown
```

- Démarrez le serveur Tomcat

```
[root@worker-node2 ~]# sudo tomcatup
Using CATALINA_BASE: /opt/tomcat8
Using CATALINA_HOME: /opt/tomcat8
Using CATALINA_TMPDIR: /opt/tomcat8/temp
Using JRE_HOME: /usr
Using CLASSPATH: /opt/tomcat8/bin/bootstrap.jar:/opt/tomcat8/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
```

- Vérifiez le bon fonctionnement du serveur Tomcat

http://<Public_IP>:8080

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

Si vous allez installer les deux serveurs sur la même machine, vous devrez changer l'un des deux. Dans ce cas, il est recommandé de changer le numéro de port du serveur Tomcat.

- Pour changer le numéro du Port du serveur Tomcat, éditez le fichier /opt/tomcat8/conf/server.xml

```
[root@worker-node2 ~]# vi /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" />
```

- 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'extérieur. Pour la désactiver :
 - Localisez les fichiers de configuration **context.xml**

```
[root@worker-node2 ~]# find / -name context.xml
```

```
/opt/tomcat8/conf/context.xml
/opt/tomcat8/webapps/examples/META-INF/context.xml
/opt/tomcat8/webapps/host-manager/META-INF/context.xml
/opt/tomcat8/webapps/manager/META-INF/context.xml
```

- Commentez les lignes contenant le champ « 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

```
[root@worker-node2 ~]# vi /opt/tomcat8/conf/tomcat-users.xml
```

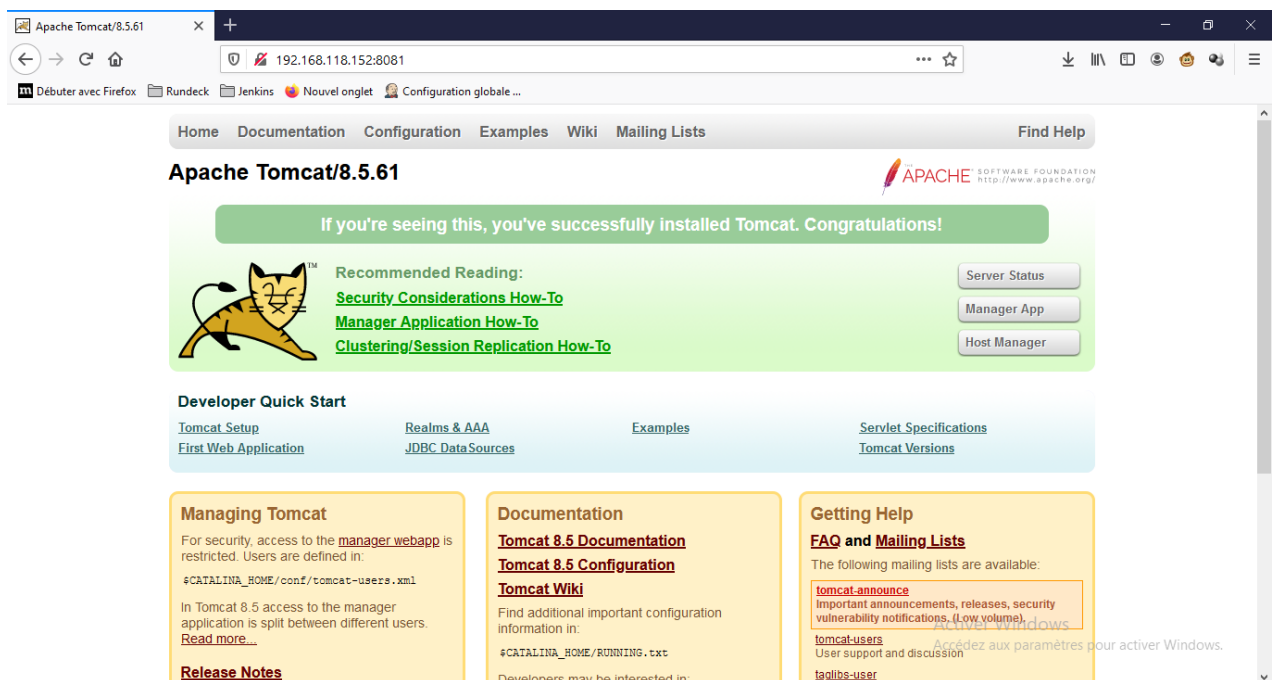
- Ajoutez les lignes suivantes, vérifiez bien les mots de passe, vous allez les utiliser ultérieurement dans Tomcat et Jenkins.

```
<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.

Déployer une application Web Maven dans un serveur Tomcat

The screenshot shows the Tomcat Manager web interface in a browser. The address bar shows the URL `192.168.118.152:8081/manager/html`. The page title is "Gestionnaire d'applications WEB Tomcat". Below the title, there is a message box and a navigation bar with links: "Lister les applications", "Aide HTML Gestionnaire", "Aide Gestionnaire", and "Etat du serveur". The main content area displays a table of applications.

Chemin	Version	Nom d'affichage	Fonctionnelle	Sessions	Commandes
/	Aucun spécifié	Welcome to Tomcat	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/docs	Aucun spécifié	Tomcat Documentation	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/examples	Aucun spécifié	Servlet and JSP Examples	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/host-manager	Aucun spécifié	Tomcat Host Manager Application	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/manager	Aucun spécifié	Tomcat Manager Application	true	1	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/myweb	Aucun spécifié	Archetype Created Web Application	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes

Etape 2 : Installation du plugin « deploy to container »

Administrer Jenkins > Gestion de plugins > Disponibles > Deploy to container

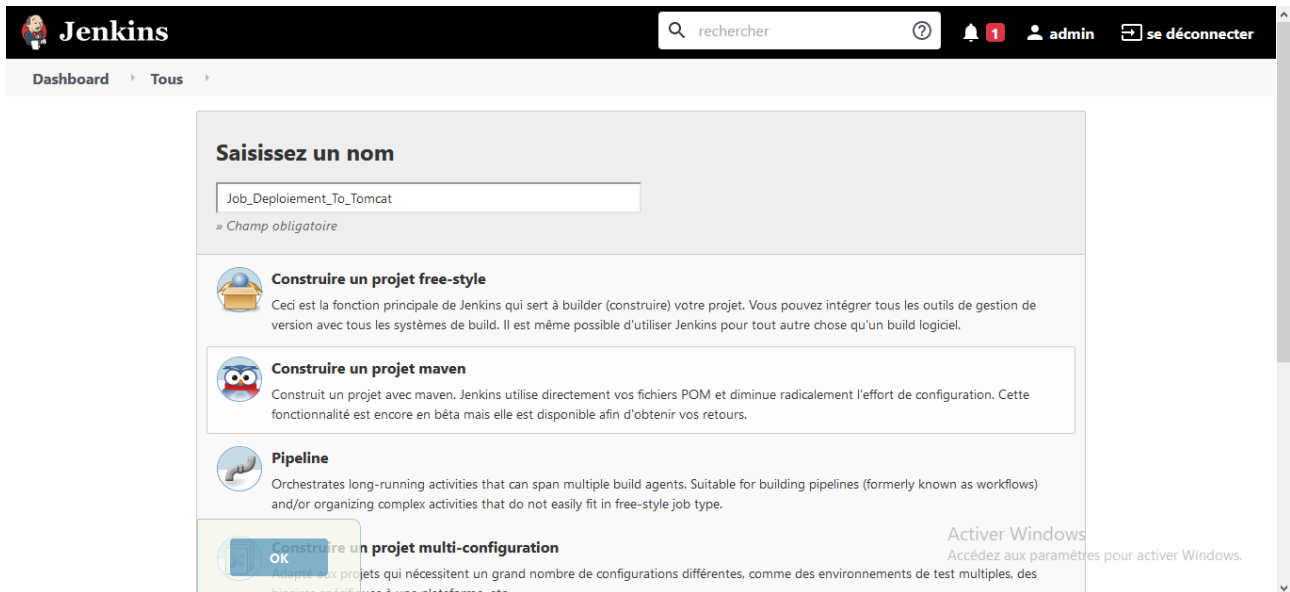
The screenshot shows the Jenkins Update Center interface. The breadcrumb navigation is "Administrer Jenkins > Gestion de plugins > Disponibles > Deploy to container". The search bar contains "deploy to container". The "Disponibles" tab is selected. The table lists available plugins:

Installer	Nom	Version	Released
<input type="checkbox"/>	Docker Pipeline Deployment DevOps docker pipeline Build and use Docker containers from pipelines.	1.25	2 mo. 4 j ago
<input checked="" type="checkbox"/>	Deploy to container Artifact Uploaders This plugin allows you to deploy a war to a container after a successful build. Glassfish 3.x remote deployment	1.16	2 mo. 15 j ago
<input type="checkbox"/>	Azure Container Service azure Deploy Kubernetes, DC/OS, Docker Swarm application configurations to Azure Container Service cluster.	1.0.3	2 mo. 22 j ago

Buttons at the bottom: "Installer sans redémarrer", "Télécharger maintenant et installer après redémarrage", "Update information obtained: 10 h ago", and "Vérifier maintenant".

Etape 3 : Création du job de déploiement

- 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

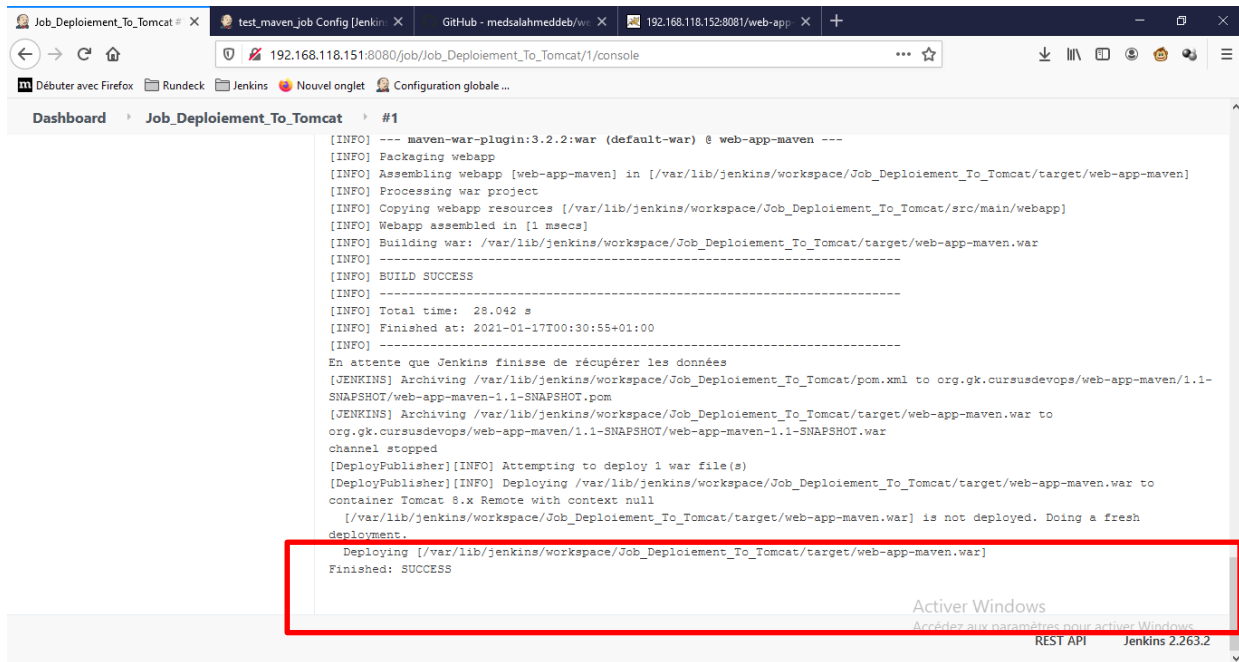
Build

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

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

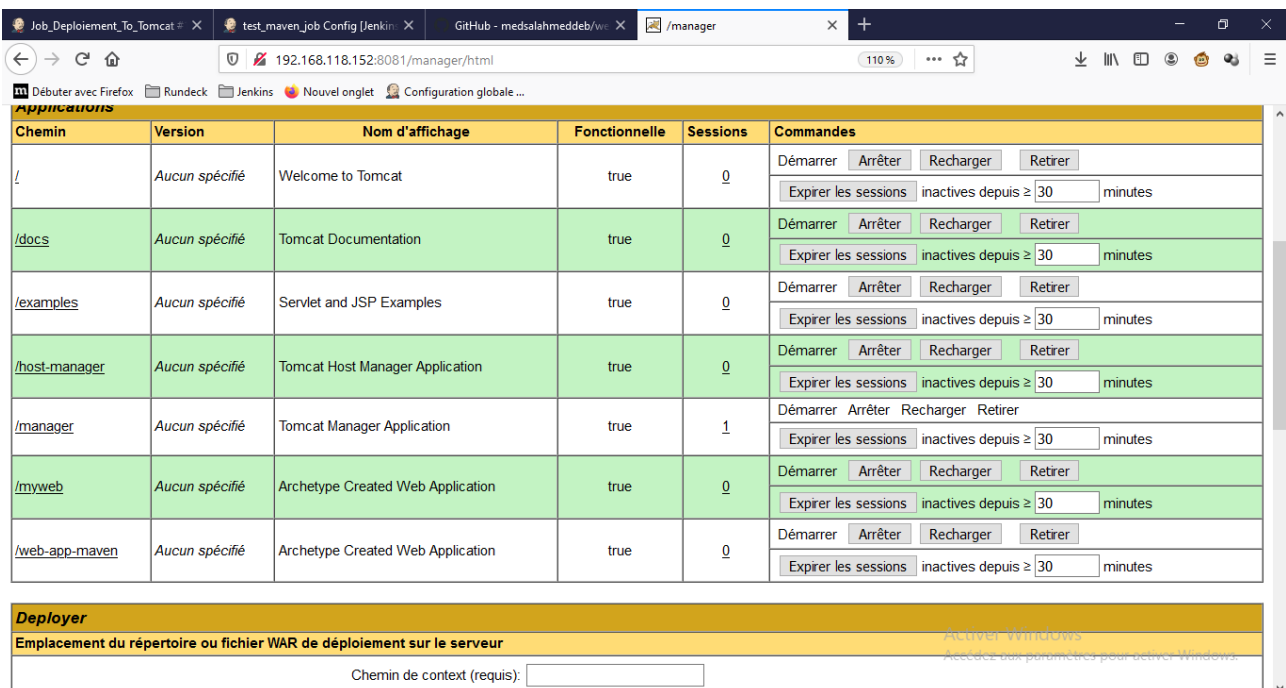
- **Fichiers WAR/EAR**: saisir le nom de l'artefact ou bien **/*.war
- **Context path**: laissez vide
- **Containers**: Containers de type Tomcat version 8
 - **Credentials** : ajoutez un nouveau credential pour l'utilisateur « deployer »
 - **Tomcat URL** : <http://192.168.118.152:8081>
- Lancez le job

Déployer une application Web Maven dans un serveur Tomcat



```
[INFO] --- maven-war-plugin:3.2.2:war (default-war) @ web-app-maven ---
[INFO] Packaging webapp
[INFO] Assembling webapp [web-app-maven] in [/var/lib/jenkins/workspace/Job_Deploiement_To_Tomcat/target/web-app-maven]
[INFO] Processing war project
[INFO] Copying webapp resources [/var/lib/jenkins/workspace/Job_Deploiement_To_Tomcat/src/main/webapp]
[INFO] Webapp assembled in [1 msecs]
[INFO] Building war: /var/lib/jenkins/workspace/Job_Deploiement_To_Tomcat/target/web-app-maven.war
[INFO] BUILD SUCCESS
[INFO] Total time: 28.042 s
[INFO] Finished at: 2021-01-17T00:30:55+01:00
[INFO] -----
En attente que Jenkins finisse de récupérer les données
[JENKINS] Archiving /var/lib/jenkins/workspace/Job_Deploiement_To_Tomcat/pom.xml to org.gk.coursudevops/web-app-maven/1.1-SNAPSHOT/web-app-maven-1.1-SNAPSHOT.pom
[JENKINS] Archiving /var/lib/jenkins/workspace/Job_Deploiement_To_Tomcat/target/web-app-maven.war to org.gk.coursudevops/web-app-maven/1.1-SNAPSHOT/web-app-maven-1.1-SNAPSHOT.war
channel stopped
[DeployPublisher][INFO] Attempting to deploy 1 war file(s)
[DeployPublisher][INFO] Deploying /var/lib/jenkins/workspace/Job_Deploiement_To_Tomcat/target/web-app-maven.war to container Tomcat 8.x Remote with context null
[/var/lib/jenkins/workspace/Job_Deploiement_To_Tomcat/target/web-app-maven.war] is not deployed. Doing a fresh deployment.
Deploying [/var/lib/jenkins/workspace/Job_Deploiement_To_Tomcat/target/web-app-maven.war]
Finished: SUCCESS
```

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



Chemin	Version	Nom d'affichage	Fonctionnelle	Sessions	Commandes
/	Aucun spécifié	Welcome to Tomcat	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/docs	Aucun spécifié	Tomcat Documentation	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/examples	Aucun spécifié	Servlet and JSP Examples	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/host-manager	Aucun spécifié	Tomcat Host Manager Application	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/manager	Aucun spécifié	Tomcat Manager Application	true	1	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/myweb	Aucun spécifié	Archetype Created Web Application	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes
/web-app-maven	Aucun spécifié	Archetype Created Web Application	true	0	Démarrer Arrêter Recharger Retirer Expirer les sessions inactives depuis ≥ 30 minutes

Deployer

Emplacement du répertoire ou fichier WAR de déploiement sur le serveur

Chemin de contexte (requis):

- Accédez à votre projet : <http://192.168.118.152:8081/web-app-maven/>

Déployer une application Web Maven dans un serveur Tomcat

