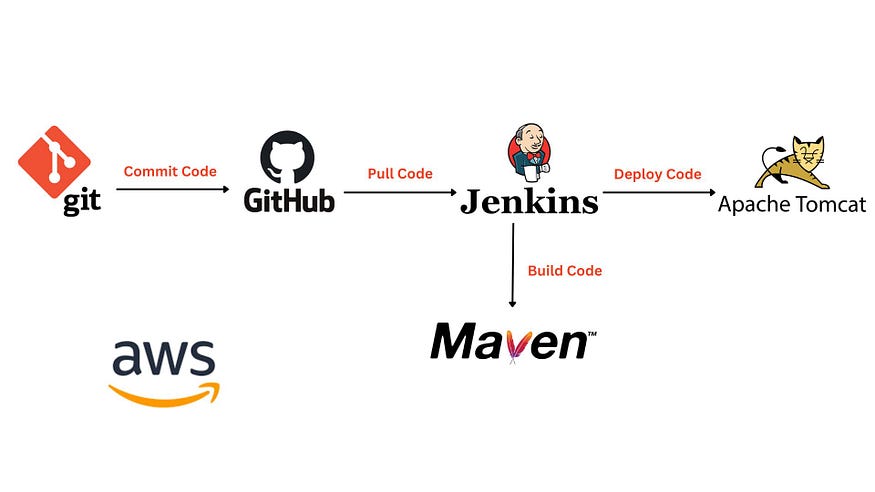
**we will build and deploy our application on the Tomcat server**

We are going to set up a CI/CD pipeline using GitHub Jenkins, Maven, and Tomcat.



**Setup CI/CD with Github, Jenkins, Maven and Tomcat**

1- Setup Jenkins

2- Setup and configure Maven and Git

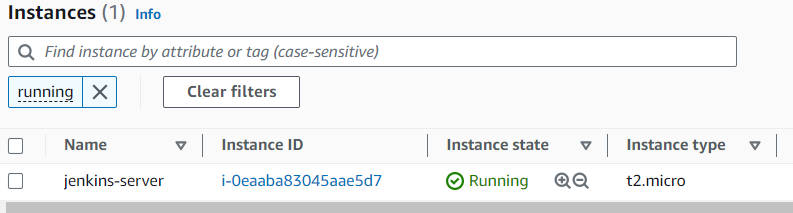
3- Setup Tomcat Server

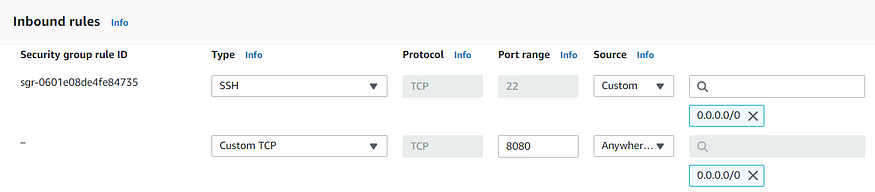
4- Integrate Github, Maven, Tomcat Server with Jenkins

5- Create a CI/CD job

6- Test the deployment

**To initiate the project, the first step involves setting up a Jenkins server.**





connect the instance to the terminal

apt update -y

apt install default-jdk

apt install maven

sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \

https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key

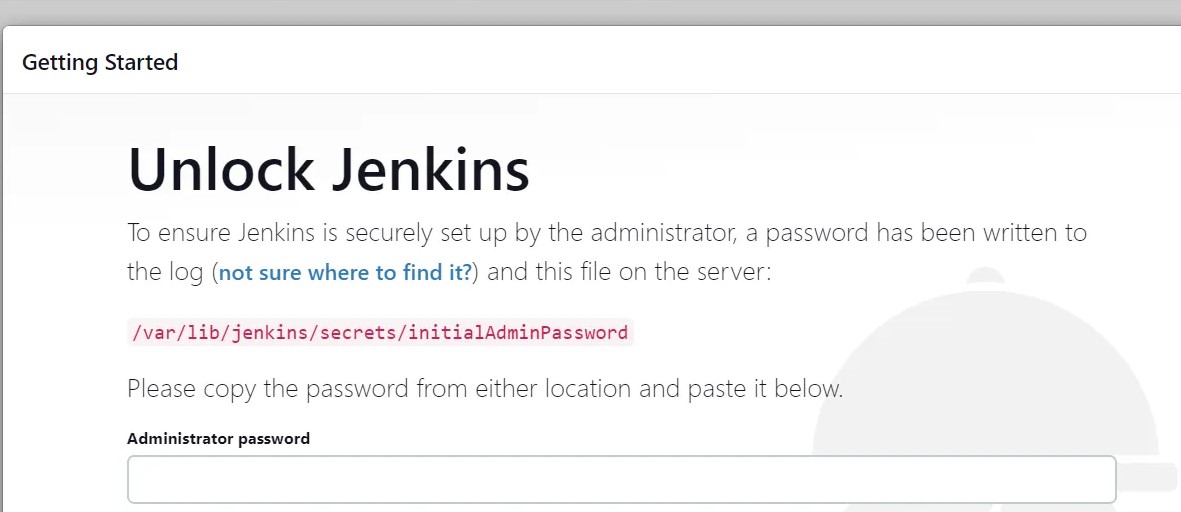
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \

https://pkg.jenkins.io/debian-stable binary/ | sudo tee \

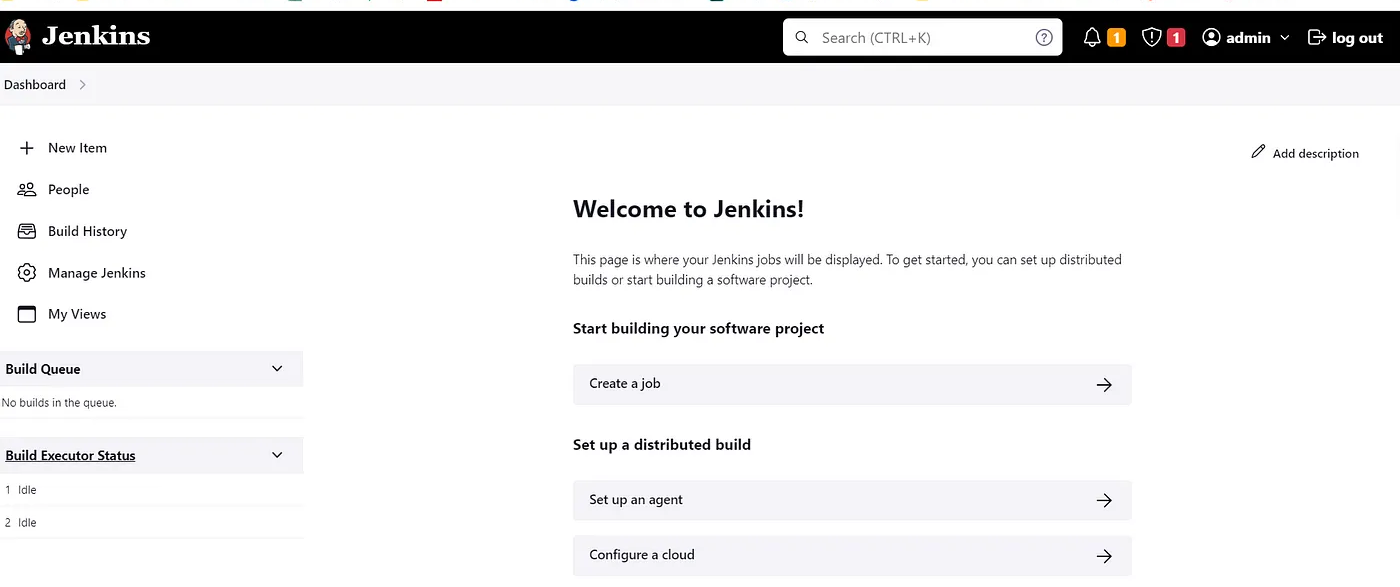
/etc/apt/sources.list.d/jenkins.list > /dev/null

sudo apt-get update

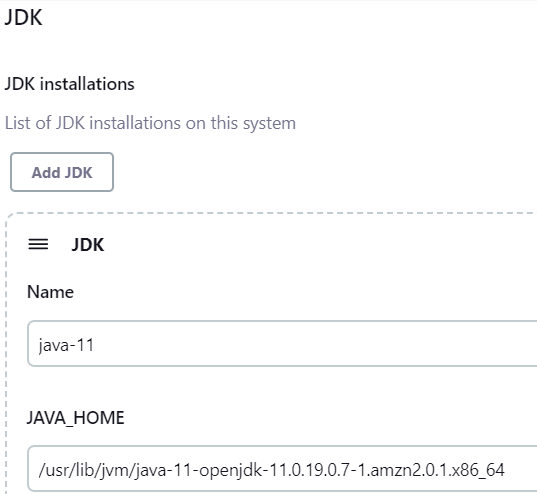
sudo apt-get install Jenkins



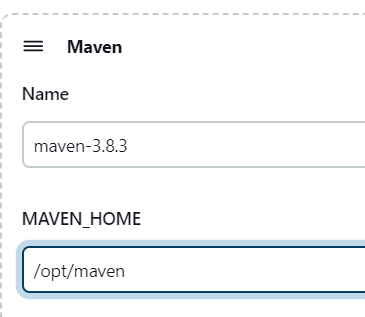
need to get the Administrator Password: cat /var/lib/jenkins/secrets/initialAdminPassword



Setup jdk in Jenkins: Manage Jenkins -> Tools -> JDK -> Add JDK



Maven -> Add Maven -> give name -> enter Maven home path -> /opt/maven



**Pipeline scripts :**

pipeline{

agent any

stages{

stage('clean workspace'){

steps{

cleanWs()

}

}

stage('checkout'){

steps{

checkout scmGit"(branches: [[name: '\*/master']], extensions: [], userRemoteConfigs: [[url: 'https://github.com/rahul11052001/train-ticket-reservation.git']])"

}

}

Stage (‘mvn compile’) {

Steps {

sh ‘mvn compile’

}

}

Stage (‘mvn test’) {

Steps {

sh ‘mvn test’

}

}

Stage (‘deploy to container’) {

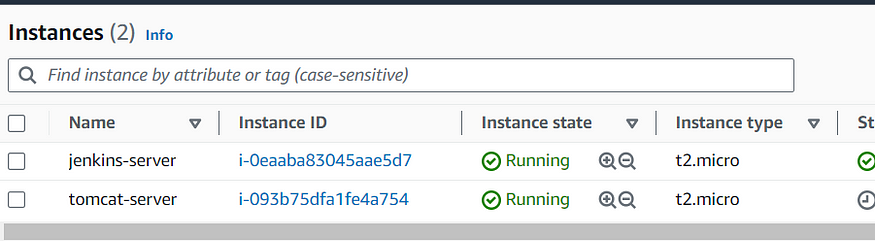
Steps {

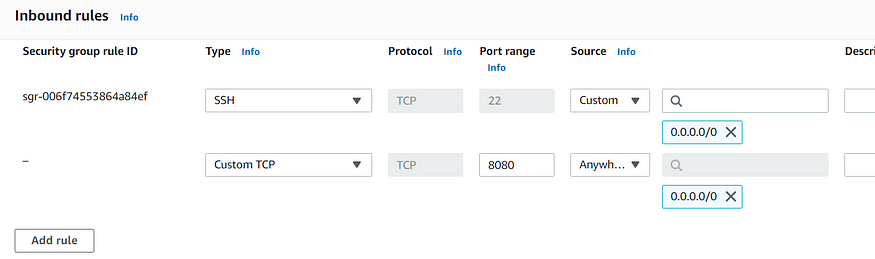
sh ‘’

}

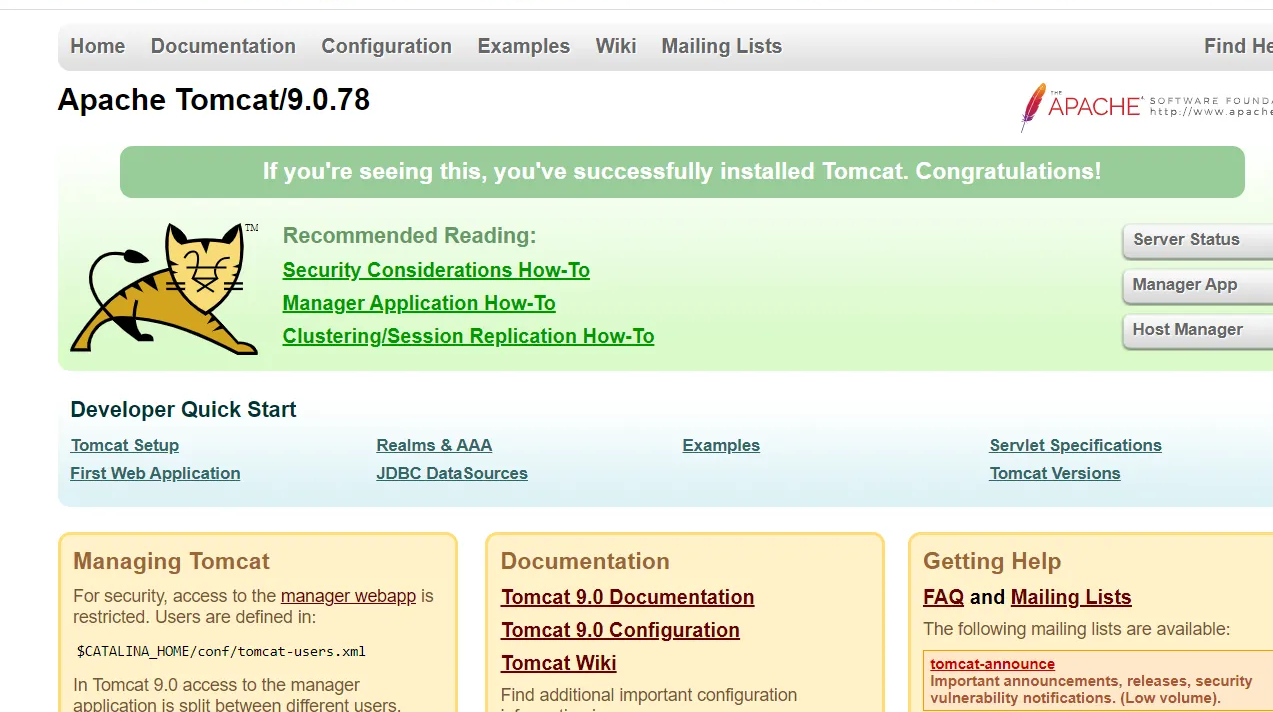
}

**Setup Tomcat Server:**





**Let's access the Tomcat server using public IP through port 8080**

****

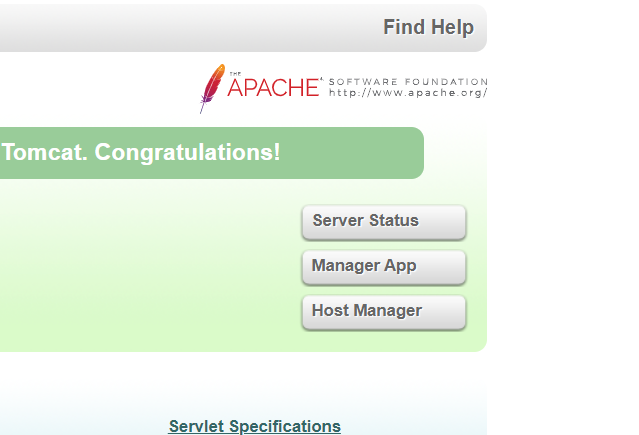
cd /opt/tomcat  
find / -name context.xml  
vim /opt/tomcat/webapps/host-manager/META-INF/context.xml  
vim /opt/tomcat/webapps/manager/META-INF/context.xml

**From both of the files comment these lines**

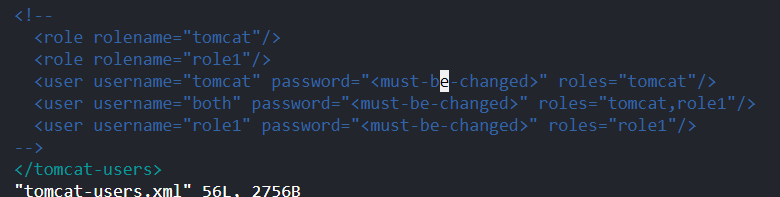


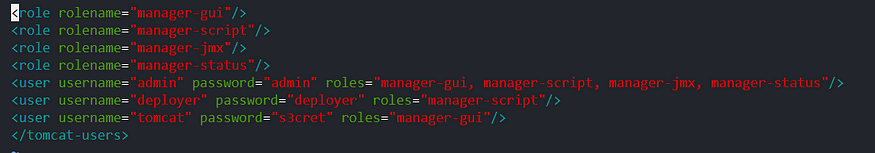
cd /opt/tomcat/bin  
 ./shutdown.sh  
 ./startup.sh

Again access the tomcat server using public ip through port 8080 and go to manager app.



Then it will be asking for credentials.





<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"/>

Lets Configure.

Jenkins -> Manage Jenkins -> Credentials -> System -> Global Credentials -> Add Credentials -> Lets add the credentials that we have updated.

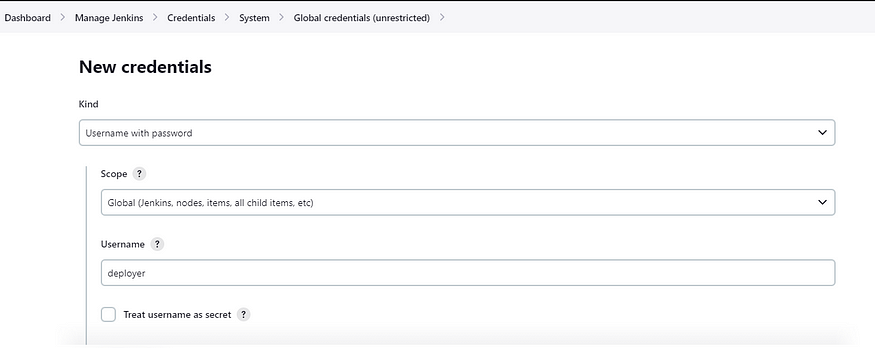
kind: Username with password

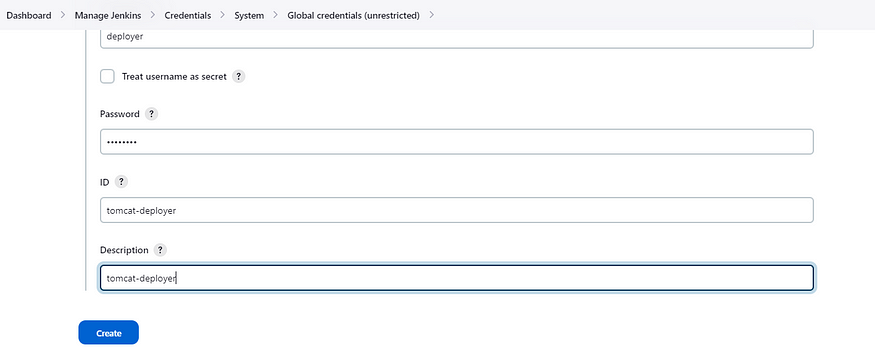
username : deployer

password: deployer

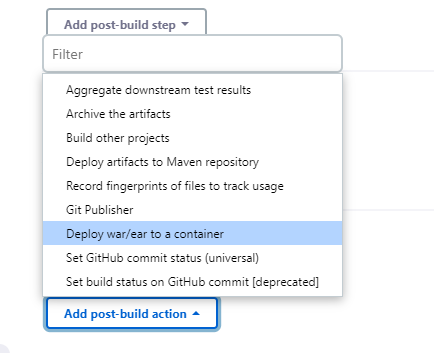
id: tomcat\_deployer

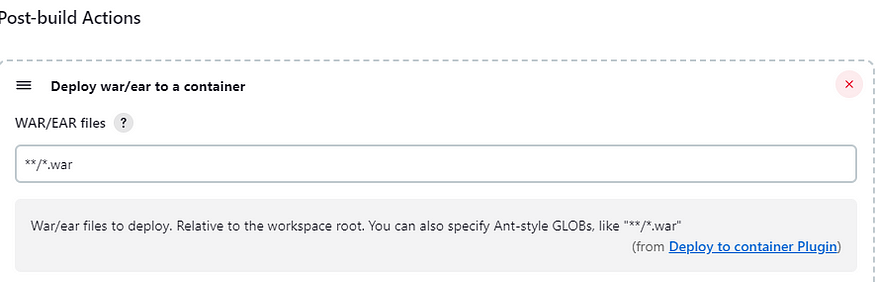
descripton: tomcat\_deployer

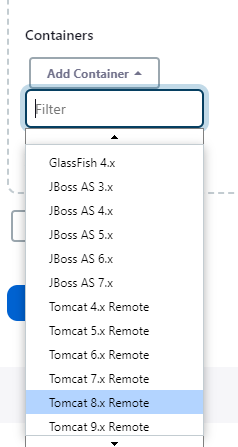


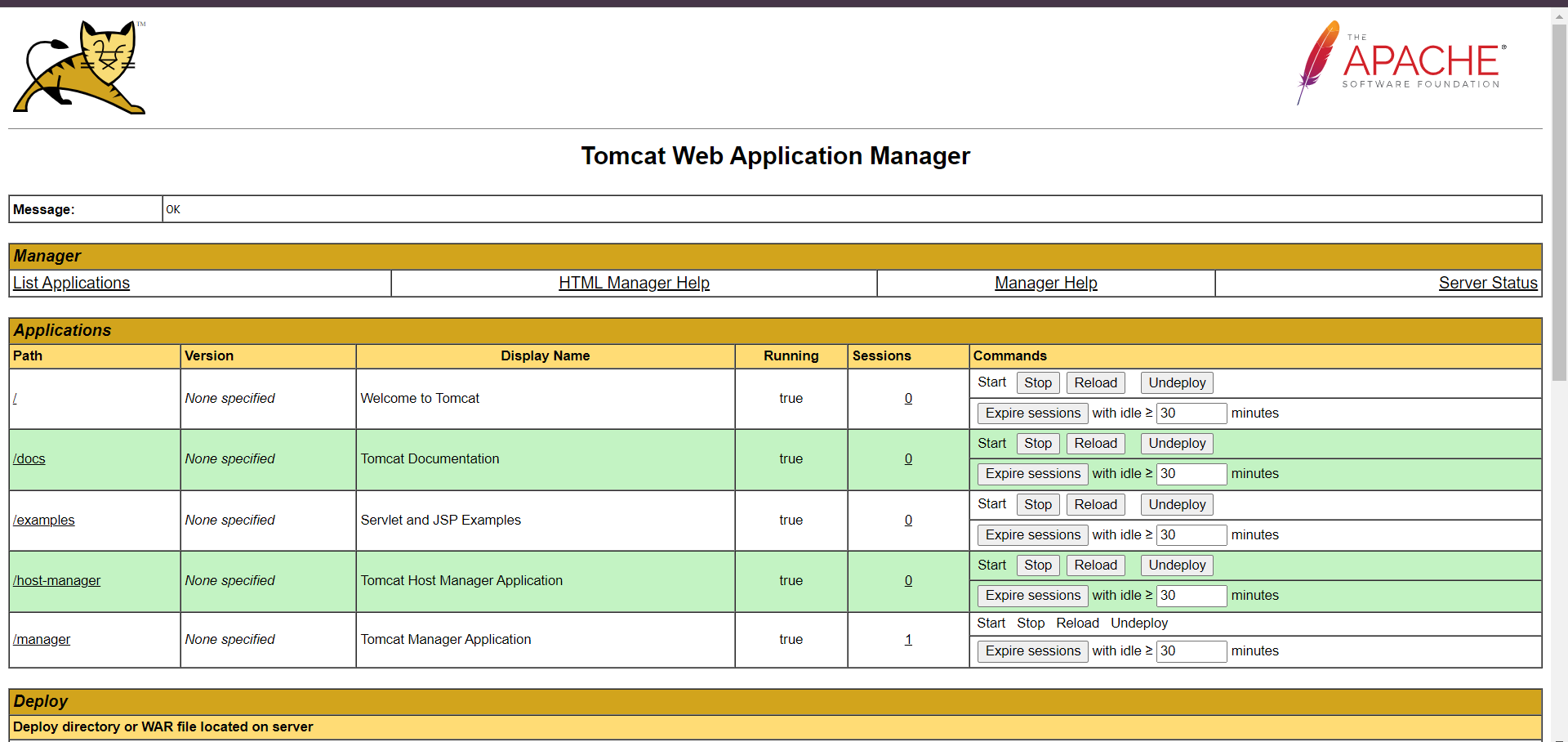


**Goto Post-build Actions section -> Add post build action → select Deploy war/ear to a container -> WAR/EAR Files : \*\*/\*.war -> Add Container -> Tomcat 8 x Remote -> select the Credentials -> Tomcat URL : http://<public-ip>:8080 -> Apply -> Save**

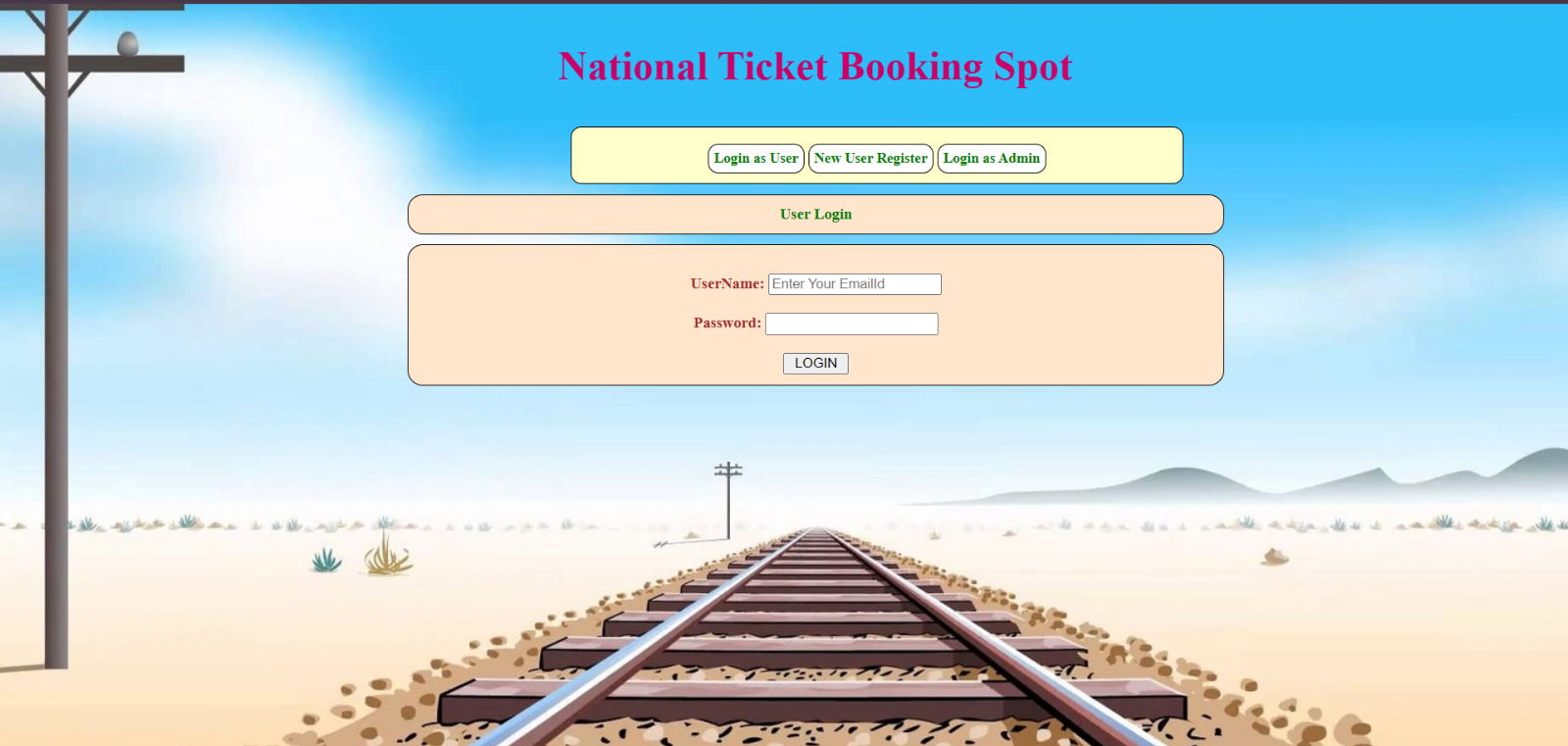






****

Next, verify that modifications in the source code are appropriately reflected in the Tomcat Server.

****