DevOps Project-1

- ## Jenkins Setup on Ubuntu:

```
apt-get update -y

sudo apt install openjdk-11-jdk -y

wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add
-

sudo sh -c 'echo deb https://pkg.jenkins.io/debian-stable binary/ > \
    /etc/apt/sources.list.d/jenkins.list'

sudo apt-get update -y

sudo apt-get install jenkins -y

systemctl start jenkins

systemctl enable jenkins
```

Maven Installation on Jenkins Server:

```
cd /opt

wget https://mirrors.estointernet.in/apache/maven/maven-
3/3.8.2/binaries/apache-maven-3.8.2-bin.tar.gz

tar -xvf apache-maven-3.8.2-bin.tar.gz

cd apache-maven-3.8.2/

vim .bashrc
```

✓ Add these 2 lines

```
export MVN_HOME=/opt/apache-maven-3.8.2/
export PATH=$PATH:/opt/apache-maven-3.8.2/bin

source ~/.bashrc
```

Nexus Setup on Amazon Linux:

```
sudo su -
yum update -y
yum install java-1.8* -y
cd /opt
wget https://download.sonatype.com/nexus/3/latest-unix.tar.gz
tar -xvf latest-unix.tar.gz
mv nexus-3.33.1-01/ nexus
```

#As a good security practice, Nexus is not advised to run nexus service as a root user, so create a new user called nexus and grant sudo access to manage nexus services as follows.

```
useradd nexus
```

✓ Give the sudo access to nexus user

```
visudo
nexus ALL=(ALL) NOPASSWD: ALL
```

✓ Change the owner and group permissions to /opt/nexus and /opt/sonatype-work directories.

```
chown -R nexus:nexus /opt/nexus
chown -R nexus:nexus /opt/sonatype-work
chmod -R 775 /opt/nexus
chmod -R 775 /opt/sonatype-work
```

✓ Open /opt/nexus/bin/nexus.rc file and uncomment run_as_user parameter and set as nexus user.

```
vi /opt/nexus/bin/nexus.rc
run_as_user="nexus"
```

✓ Create nexus as a service

```
ln -s /opt/nexus/bin/nexus /etc/init.d/nexus
```

✓ Switch as a nexus user and start the nexus service as follows.

```
su - nexus
```

✓ Enable the nexus services

```
sudo systemctl enable nexus
```

✓ Start the nexus service

```
sudo systemctl start nexus
```

✓ Access the Nexus server from Laptop/Desktop browser.

http://IPAddess/Hostname:8081/

Sonarqube Setup on Ubuntu:

```
cd /opt
wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-7.9.6.zip
apt install unzip
unzip sonarqube-7.9.6.zip
mv sonarqube-7.9.6 sonarqube

adduser sonar
```

✓ Give the sudo access to sonar user

```
visudo
sonar   ALL=(ALL)       NOPASSWD: ALL
```

✓ Change the owner and group permissions to /opt/sonarqube-/ directory.

```
chown -R sonar:sonar /opt/sonarqube/
chmod -R 775 /opt/sonarqube/
su - sonar
cd /opt/sonarqube/bin/linux-x86-64
./sonar.sh start
```

Tomcat Setup on Ubuntu:

```
apt-get update -y
sudo apt install openjdk-11-jdk -y
cd /opt
wget https://mirrors.estointernet.in/apache/tomcat/tomcat-9/v9.0.52/bin/apache-tomcat-9.0.52.tar.gz
tar -xvf apache-tomcat-9.0.52.tar.gz
mv apache-tomcat-9.0.52 tomcat
cd tomcat/
cd conf/
```

✓ vim tomcat-users.xml

```
<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"/>
```

✓ vi /opt/tomcat/webapps/manager/META-INF/context.xml

Comment this line

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

✓ vim /opt/tomcat/webapps/host-manager/META-INF/context.xml

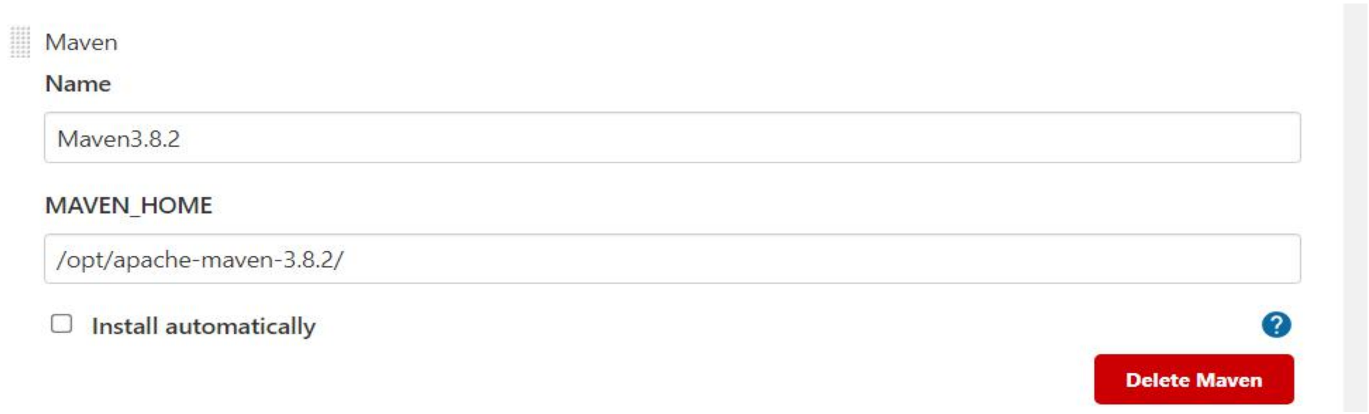
Comment this line

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

```
cd bin
./startup.sh
```

- **Jenkins Job:**

Maven Project: https://github.com/GirijalaAditya/maven_project.git



Maven

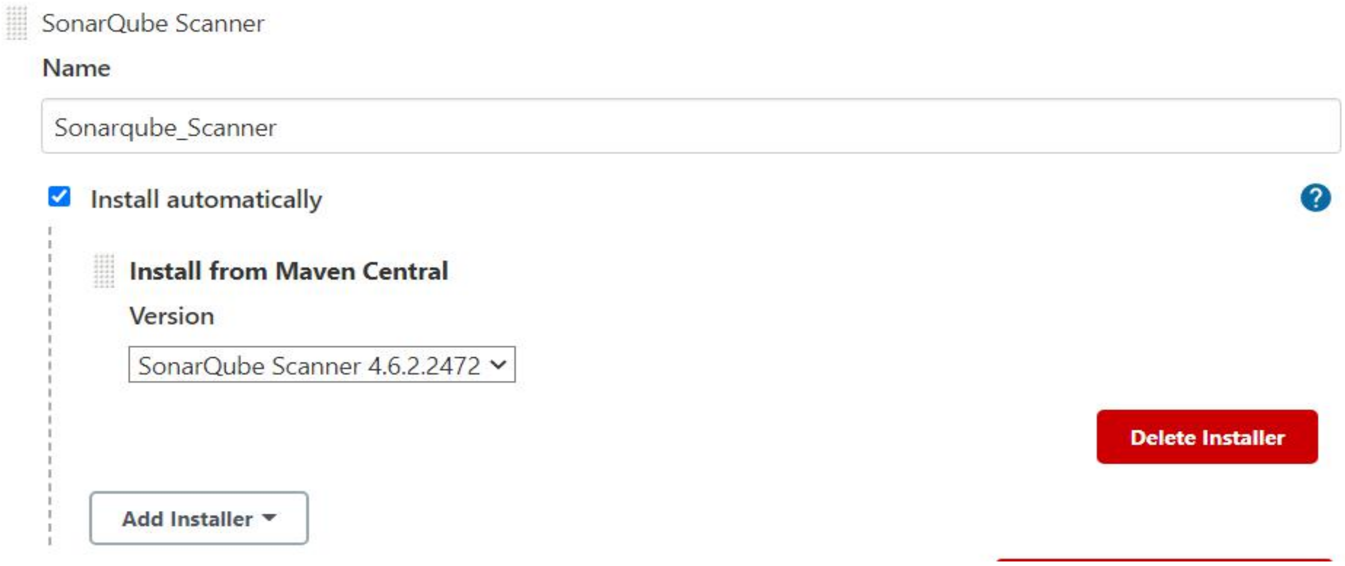
Name

Maven3.8.2

MAVEN_HOME

/opt/apache-maven-3.8.2/

☐ Install automatically



SonarQube Scanner

Name

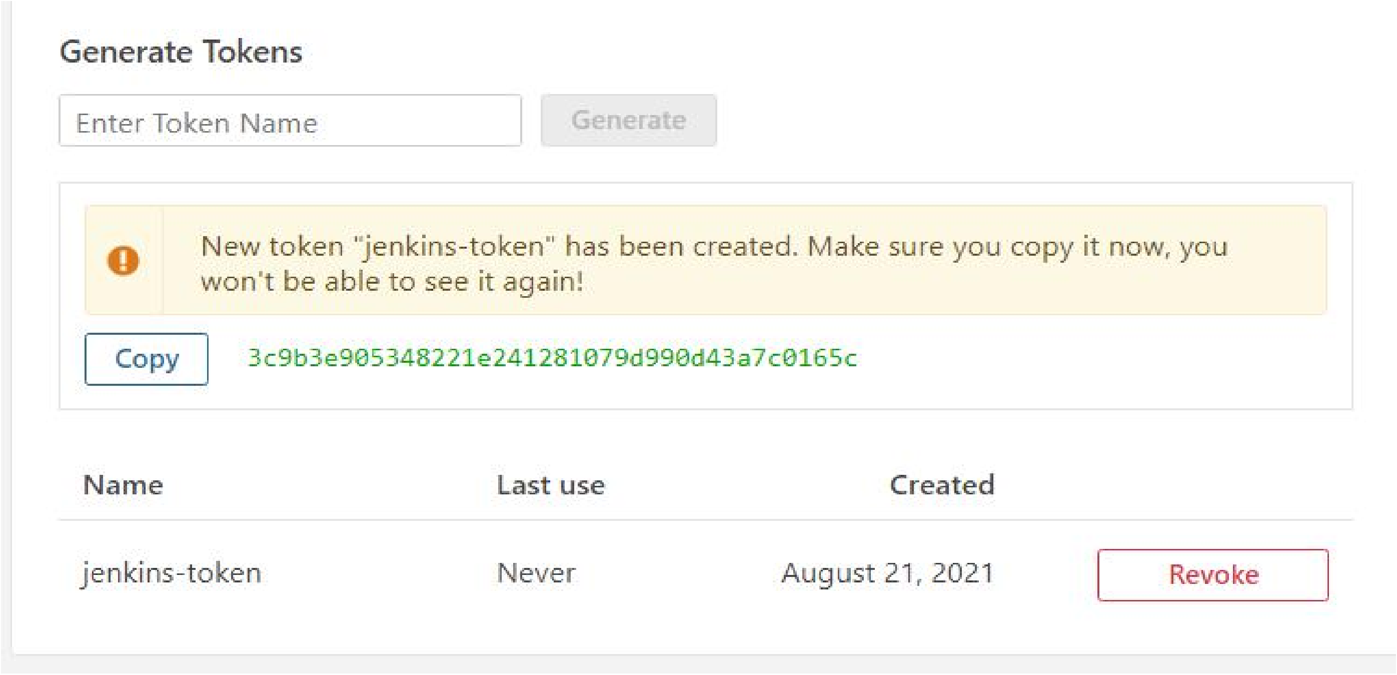
Sonarqube_Scanner

☑ Install automatically

Install from Maven Central

Version

SonarQube Scanner 4.6.2.2472



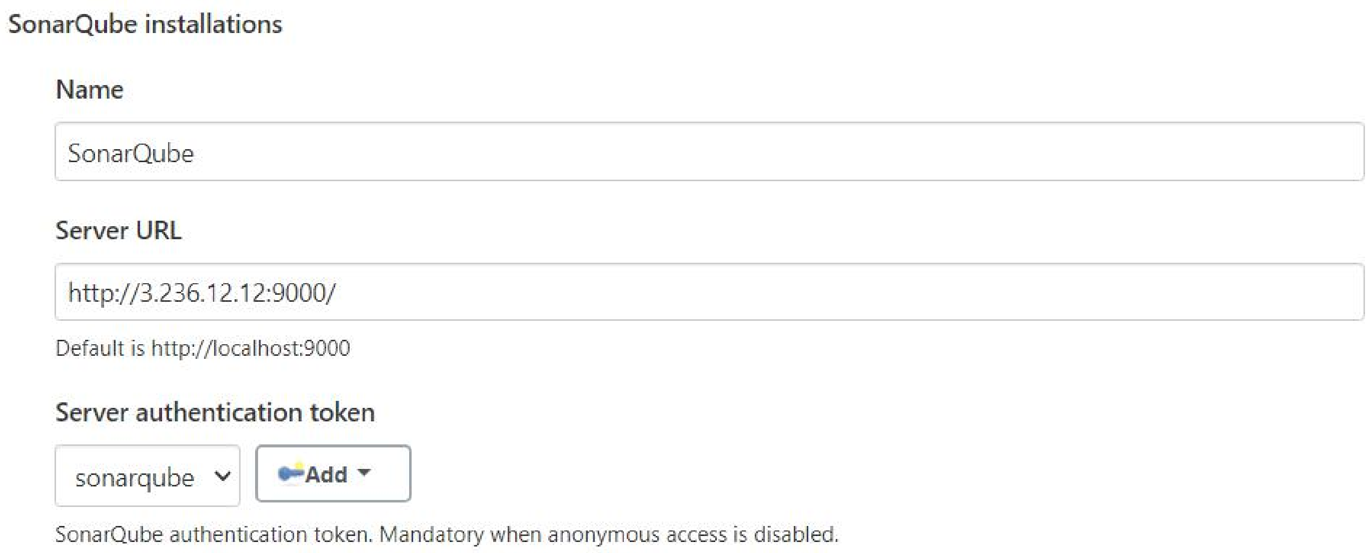
Generate Tokens

Enter Token Name

New token "jenkins-token" has been created. Make sure you copy it now, you won't be able to see it again!

3c9b3e905348221e241281079d990d43a7c0165c

Name	Last use	Created	
jenkins-token	Never	August 21, 2021	Revoke



SonarQube installations

Name

SonarQube

Server URL

http://3.236.12.12:9000/

Default is http://localhost:9000

Server authentication token

sonarqube

SonarQube authentication token. Mandatory when anonymous access is disabled.

✓ Add this file in pom.xml

```
<distributionManagement>

          <repository>
            <id>nexus</id>
            <name> Releases Nexus Repository</name>
            <url>http://3.235.140.90:8081/repository/maven-webapp-release/</url>
          </repository>

          <snapshotRepository>
            <id>nexus</id>
            <name>Snapshot Nexus Repository </name>
            <url>http://3.235.140.90:8081/repository/maven-webapp-release/</url>
          </snapshotRepository>

       </distributionManagement>
```

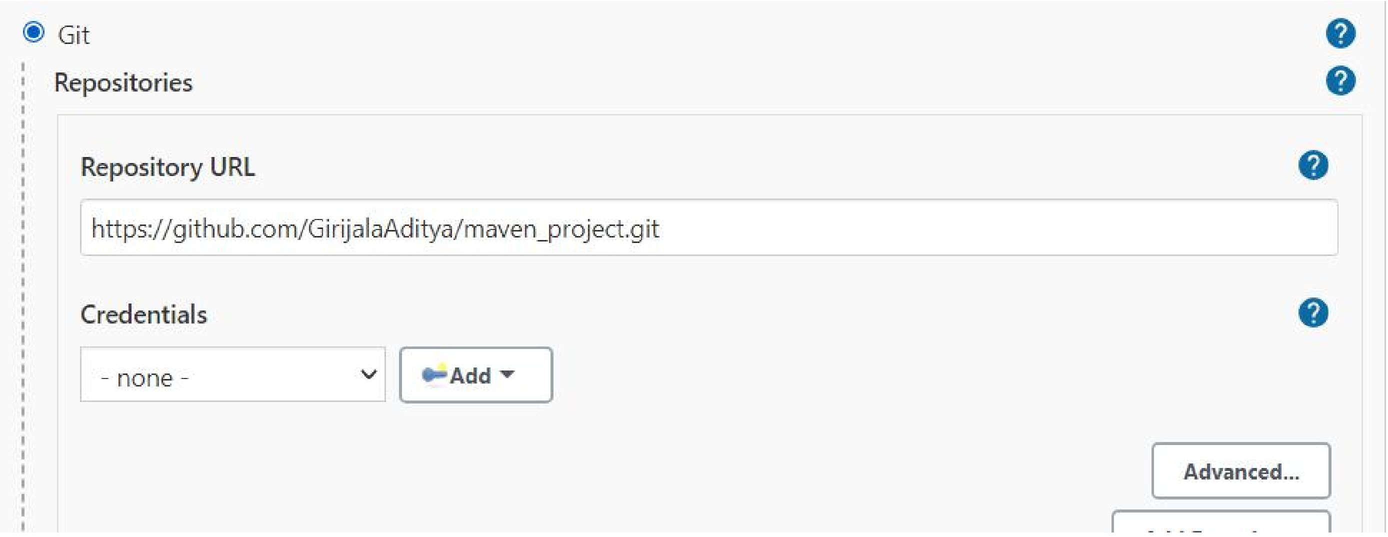
✓ In Jenkins server,

```
cd /opt/apache-maven-3.8.2/conf
vim settings.xml
```

✓ Add this in <servers> block

```
   <server>
     <id>nexus</id>
     <username>aditya</username>
     <password>aditya</password>
   </server>
```

- **Job Setup:**



● Git

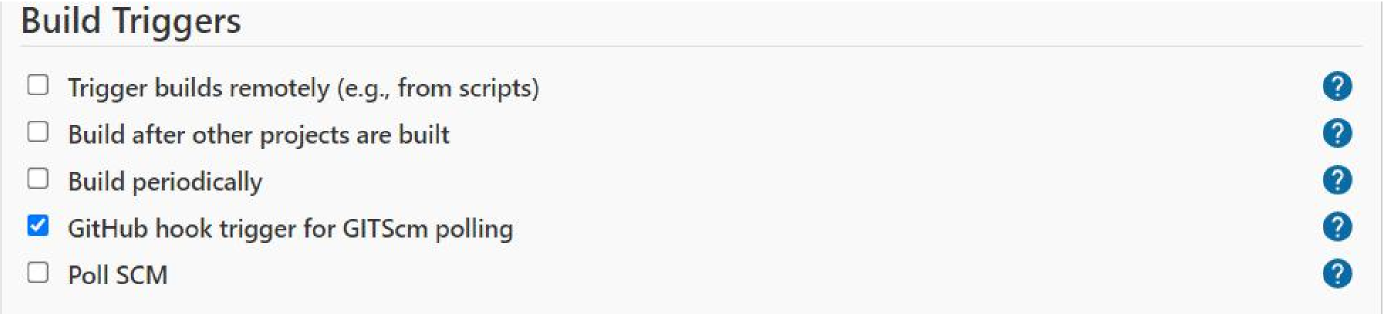
Repositories

Repository URL

https://github.com/GirijalaAditya/maven_project.git

Credentials

- none -



Build Triggers

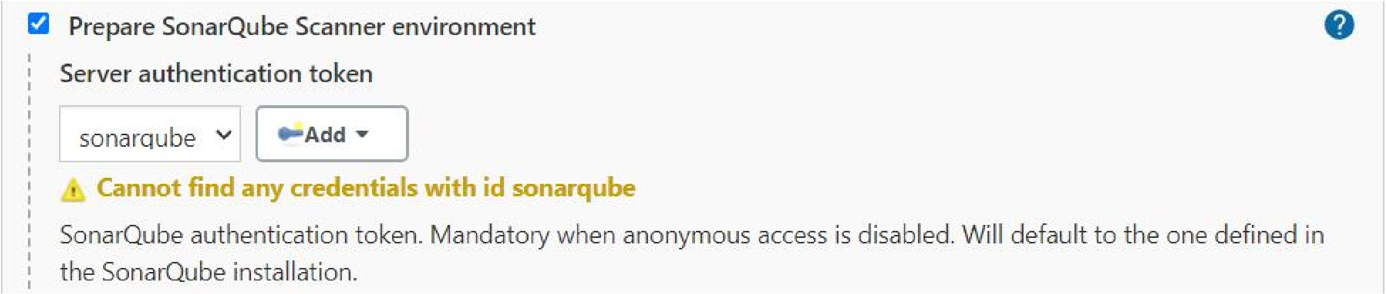
□ Trigger builds remotely (e.g., from scripts)

□ Build after other projects are built

□ Build periodically

☑ GitHub hook trigger for GITScm polling

□ Poll SCM



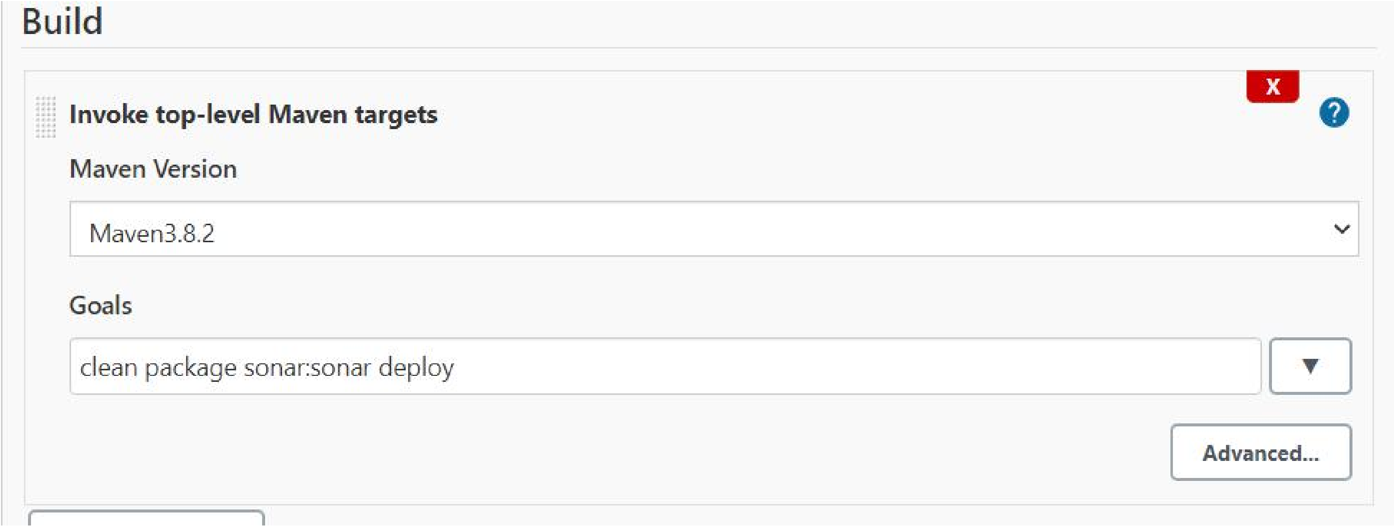
☑ Prepare SonarQube Scanner environment

Server authentication token

sonarqube

⚠ Cannot find any credentials with id sonarqube

SonarQube authentication token. Mandatory when anonymous access is disabled. Will default to the one defined in the SonarQube installation.



Build

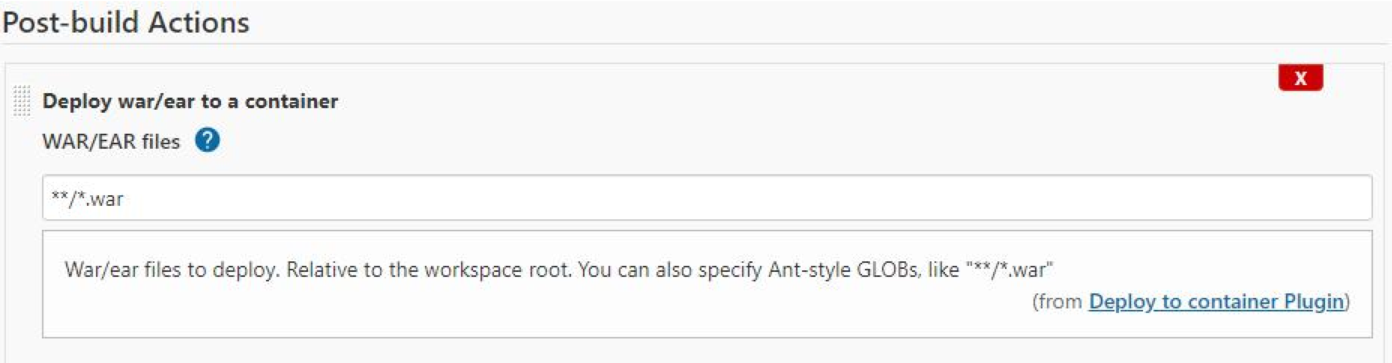
Invoke top-level Maven targets

Maven Version

Maven3.8.2

Goals

clean package sonar:sonar deploy



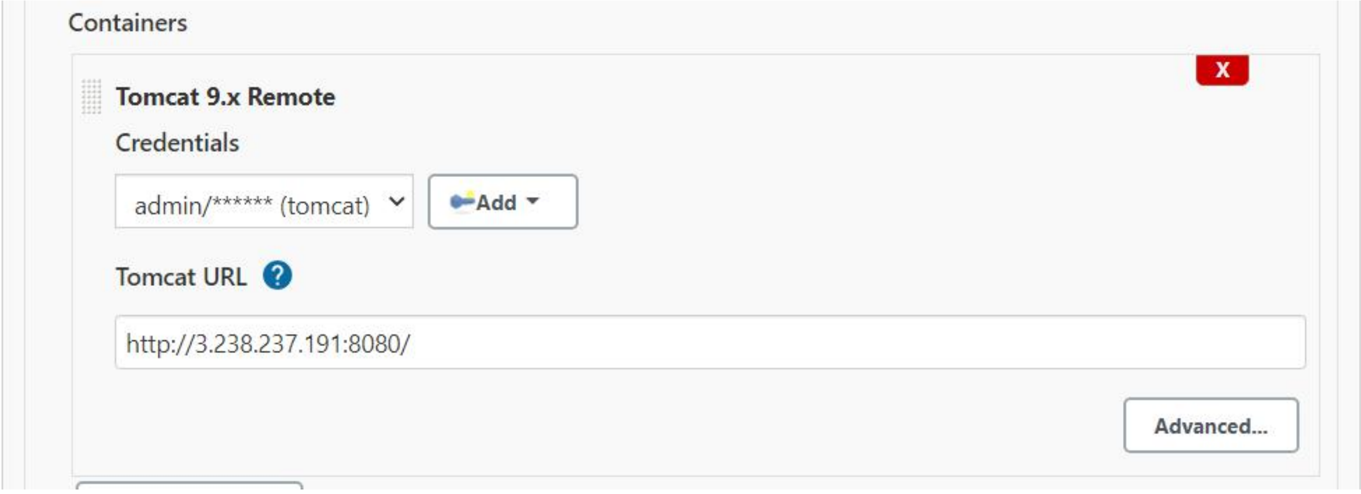
Post-build Actions

Deploy war/ear to a container

WAR/EAR files

**/*.war

War/ear files to deploy. Relative to the workspace root. You can also specify Ant-style GLOBs, like "**/*.war"
(from Deploy to container Plugin)



Containers

Tomcat 9.x Remote

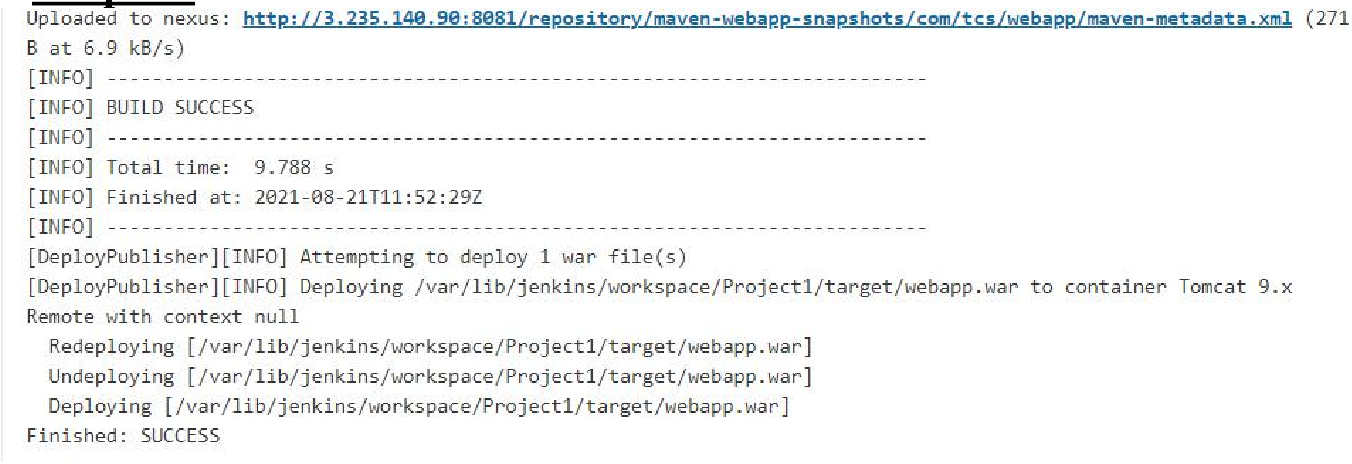
Credentials

admin/****** (tomcat)

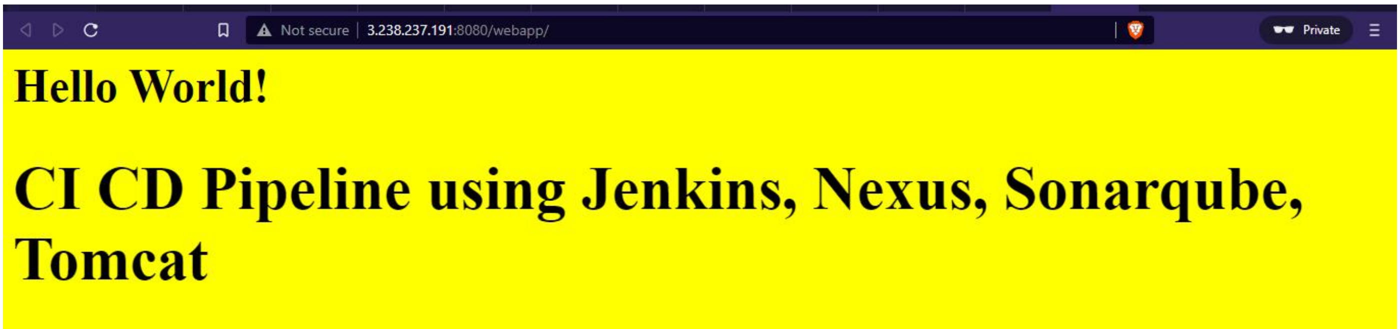
Tomcat URL

http://3.238.237.191:8080/

- **Outputs:**



```
Uploaded to nexus: http://3.235.140.90:8081/repository/maven-webapp-snapshots/com/tcs/webapp/maven-metadata.xml (271
B at 6.9 kB/s)
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  9.788 s
[INFO] Finished at: 2021-08-21T11:52:29Z
[INFO] ------------------------------------------------------------------------
[DeployPublisher][INFO] Attempting to deploy 1 war file(s)
[DeployPublisher][INFO] Deploying /var/lib/jenkins/workspace/Project1/target/webapp.war to container Tomcat 9.x
Remote with context null
  Redeploying [/var/lib/jenkins/workspace/Project1/target/webapp.war]
  Undeploying [/var/lib/jenkins/workspace/Project1/target/webapp.war]
  Deploying [/var/lib/jenkins/workspace/Project1/target/webapp.war]
Finished: SUCCESS
```



Hello World!

CI CD Pipeline using Jenkins, Nexus, Sonarqube, Tomcat

- **Declarative Pipeline:**

```
pipeline
{
   agent any
   tools
   {
      maven 'Maven3.8.2'
   }
   stages
   {
      stage("Code Checkout")
      {
         steps
         {
            git branch: 'main', url:
'https://github.com/GirijalaAditya/maven_project.git'
         }
      }
      stage("Code Build")
      {
         steps
         {
            sh "mvn clean package"
         }
      }
      stage("SonarQube Analysis")
      {
         steps
         {
            withSonarQubeEnv('SonarQube')
            {
               sh "mvn sonar:sonar"
            }
         }
      }
      stage("Nexus Upload")
      {
         steps
         {
                sh "mvn deploy"
            }
        }
        stage("Deploy to Tomcat")
        {
            steps
            {
                deploy adapters: [tomcat9(credentialsId: 'tomcat', path: '', url: 'http://3.238.237.191:8080/')], contextPath: null, war: '**/*.war'
            }
        }
    }
}
```