

## **JENKINS AND TOMCAT SERVER SIMULATION**

What is Jenkins?

Jenkins is an open-source automation tool written in Java programming language that allows continuous integration.

Jenkins builds and tests our software projects, which continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build.

Jenkins builds and tests our software projects which continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build.

It also allows us to continuously deliver our software by integrating with a large number of testing and deployment technologies.

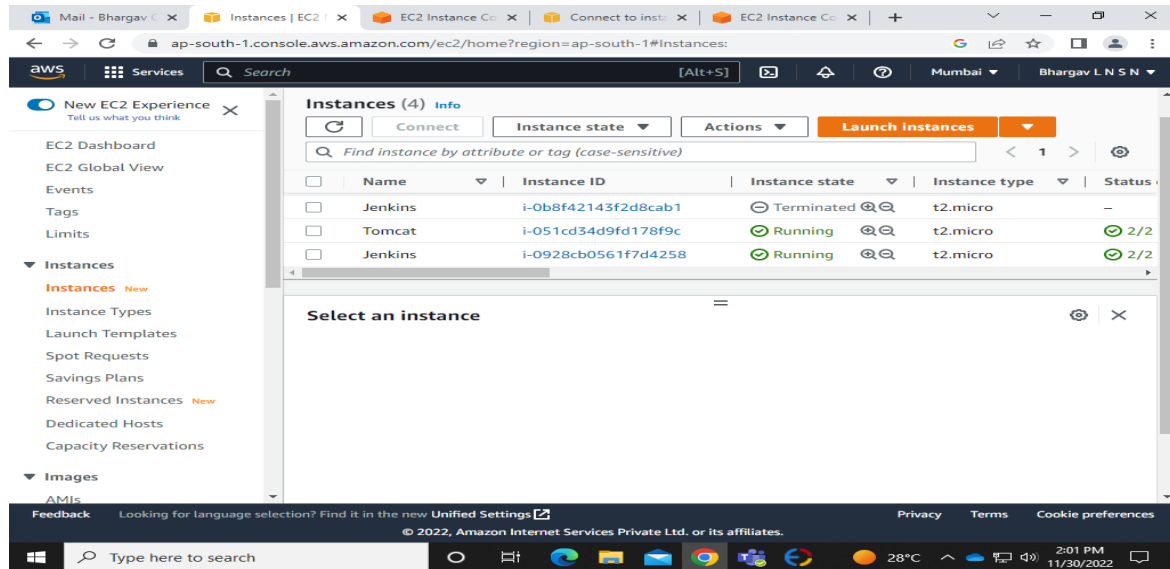
Jenkins offers a straightforward way to set up a continuous integration or continuous delivery environment for almost any combination of languages and source code repositories using pipelines, as well as automating other routine development tasks.

What is Tomcat?

It is an open-source Java servlet container that implements many Java Enterprise Specs such as the Websites API, Java-Server Pages and last but not least, the Java Servlet.

It allows the users to run Servlet and JAVA Server Pages that are based on the web-applications. It can be used as the HTTP server. The performance of the Tomcat server is not as good as the designated web server.

## Step-1 Launch 2 EC2 Ubuntu Instances



## Step 2: Install Jenkins through EC@ connect with the following commands

sudo su

sudo apt-get update -y

sudo apt-get install openjdk-11-jdk

sudo apt-get install git maven -y

wget <http://mirrors.jenkins.io/war-stable/latest/jenkins.war>

java -jar jenkins.war

-ssltn

**Step 3: Install Tomcat on second machine with following commands**

```
sudo apt-get update -y
```

```
sudo apt-get install openjdk-11-jdk
```

```
sudo apt-get install -y tomcat9
```

```
sudo apt-get install -y tomcat9-admin
```

```
cd /var/lib/tomcat9/conf
```

```
sudo vim tomcat-users.xml
```

```
sudo service tomcat9 restart
```

```
sudo passwd ubuntu
```

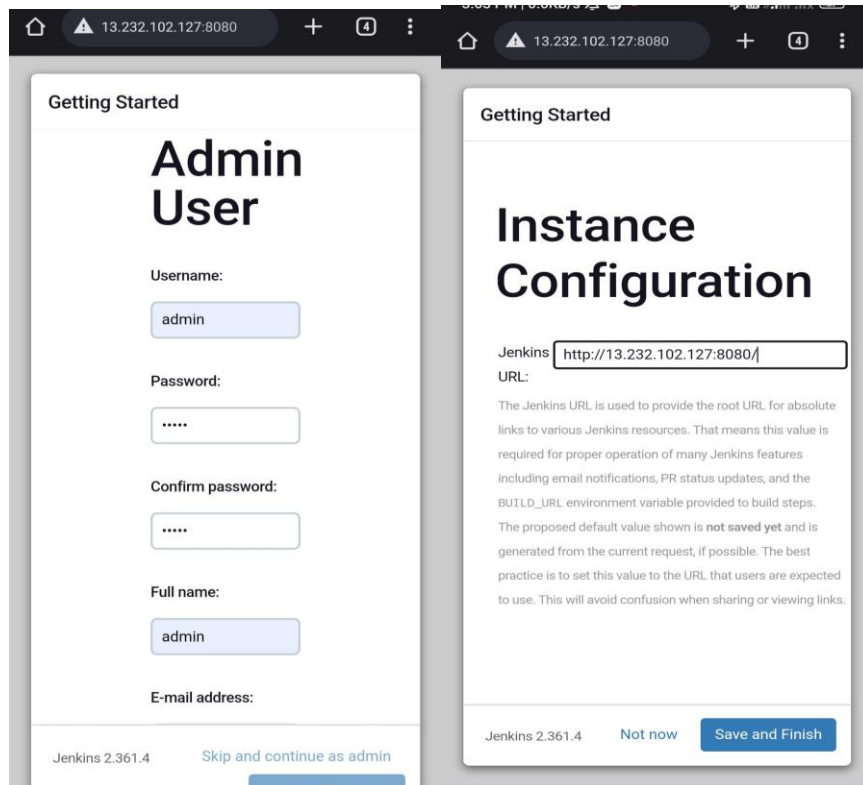
```
sudo service ssh restart
```

```
sudo service sshd restart
```

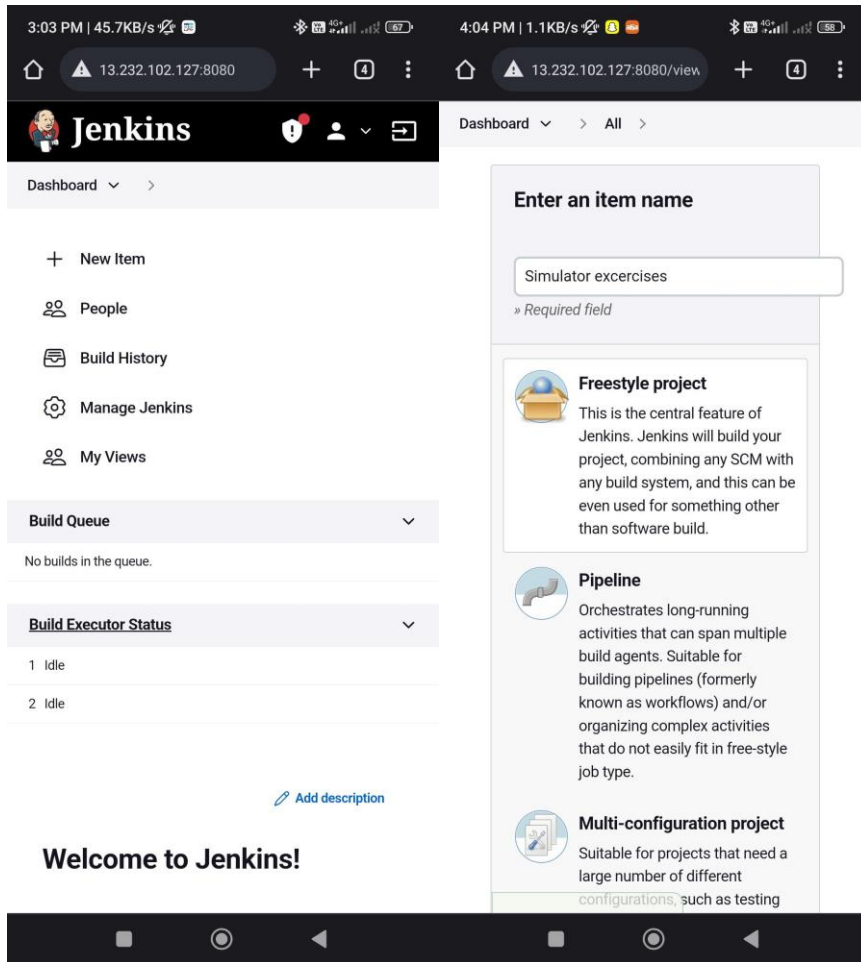
Step 4: To have password less connection with two servers

Step 5: With the public IP of the Jenkins server copy that and paste in new tab using default port 8080 and use the password to unlock the jenkins

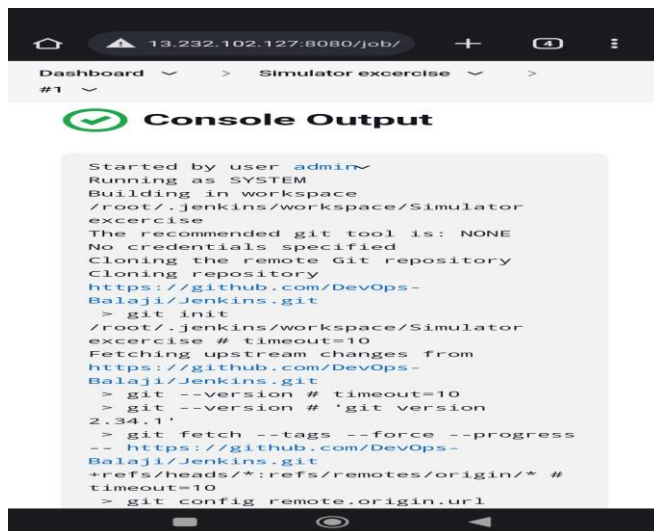
<http://3.109.185.168:8080/>



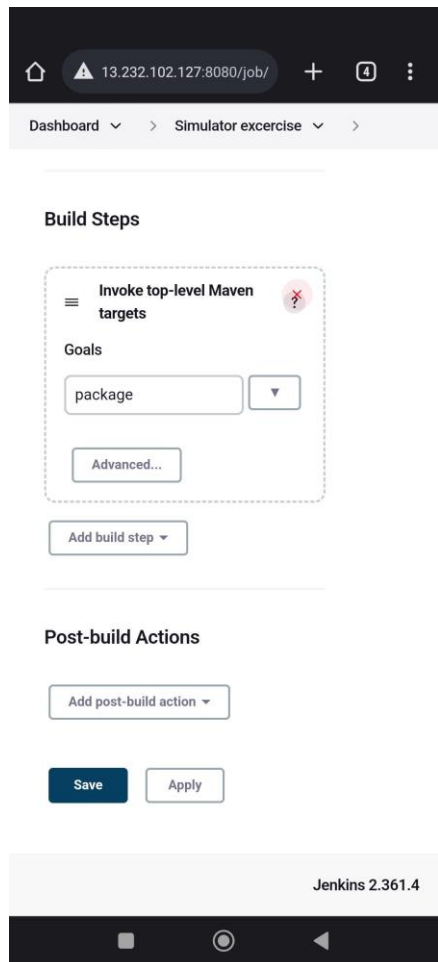
Step 6: **Create a new item in jenkins dashboard and select freestyle project.**



Step 7: Go to the source code management and select Git & Paste Git Repository URL



### Step-9 Add Build step & select invoke top level maven targets



Step-10: Go to manage jenkins & manage plugins and install Deploy to container.

Step-11: Go to the Post-Build Actions & Add the container & give the credentials and build

Dashboard > Simulator exercise >

WAR/EAR files ?

Context path ?

Containers

Tomcat 9.x Remote

Credentials

+ Add

Tomcat URL ?

Advanced...

Add Container

Save Apply

13.232.102.127:8080/job/

Dashboard > Simulator exercise >

Scope ?

Username ?

☐ Treat username as secret ?

Password ?

ID ?

Description ?

Advanced...

Save Apply

Dashboard > Manage Jenkins >

Update Center

Pipeline: REST API	Success
JavaScript GUI Lib:	Success
Moment.js bundle	Success
Pipeline: Stage View	Success
Git	Success
SSH Build Agents	Success
Matrix Authorization Strategy	Success
PAM Authentication	Success
LDAP	Success
Email Extension	Success
Mailer	Success
Loading plugin extensions	Success
Deploy to container	Installing
Loading plugin extensions	Pending

→ [Go back to the top page](#)  
(you can start using the installed plugins right away)

→ ☐ Restart Jenkins when installation is complete and no jobs are running

Jenkins 2.361.4

Step-12 Take the public ip of the tomcat server and open in the new tab

