1. Prerequisites (System & Config)

Before creating the job in Jenkins, ensure these settings are active on your Laptop.

A. Tools should present in your Laptop

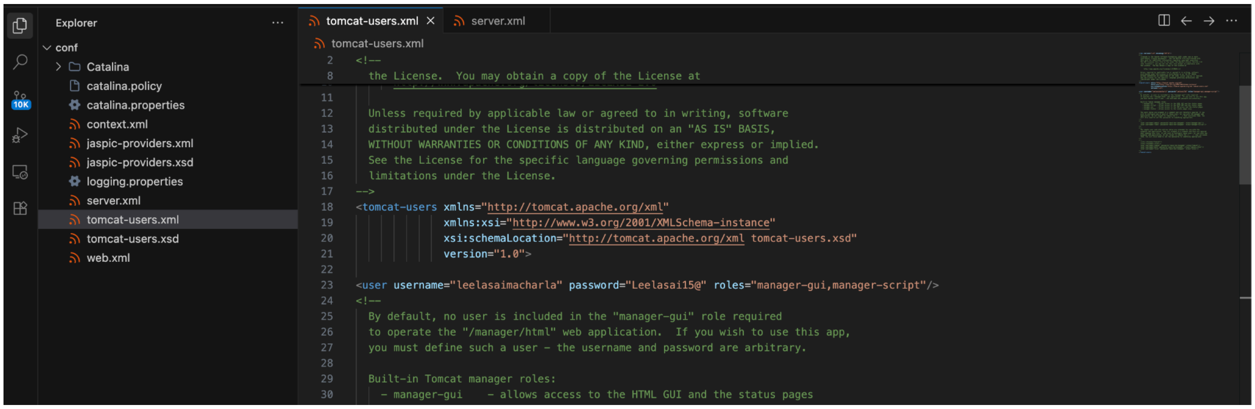
1. Install jenkins
2. Install Tomcat@9
3. Install Maven
4. Install JDK (with respect to the tomcat)
5. Install Git

B. Tomcat User Permissions

Jenkins needs access to the Tomcat Manager API. Ensure your `/opt/homebrew/etc/tomcat@9/tomcat-users.xml` has these specific roles:

XML

```
<user username="admin" password="yourpassword" roles="manager-gui,manager-script"/>
```



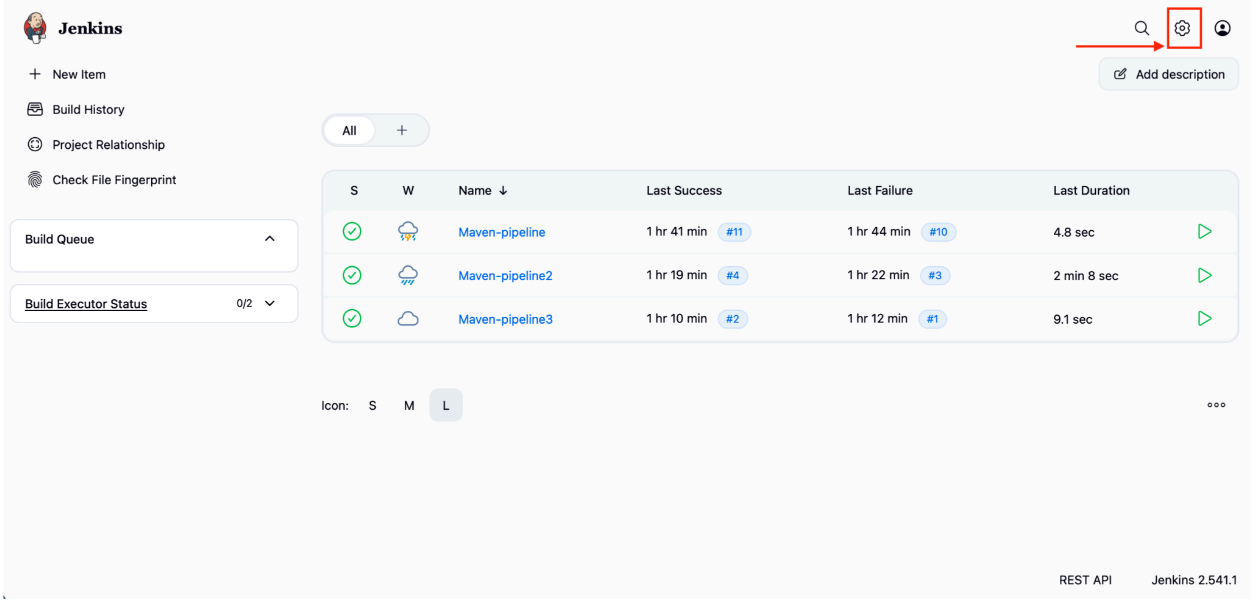
```
<!--
  the License.  You may obtain a copy of the License at

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->
<tomcat-users xmlns="http://tomcat.apache.org/xml"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
              version="1.0">

<user username="leelasaimacharla" password="Leelasai15@" roles="manager-gui,manager-script"/>
<!--
  By default, no user is included in the "manager-gui" role required
  to operate the "/manager/html" web application.  If you wish to use this app,
  you must define such a user - the username and password are arbitrary.

  Built-in Tomcat manager roles:
    - manager-gui    - allows access to the HTML GUI and the status pages
```

IMP : Make sure the port numbers and server ports of Jenkins and Tomcat are configured (shouldn’t use same port number or server port)

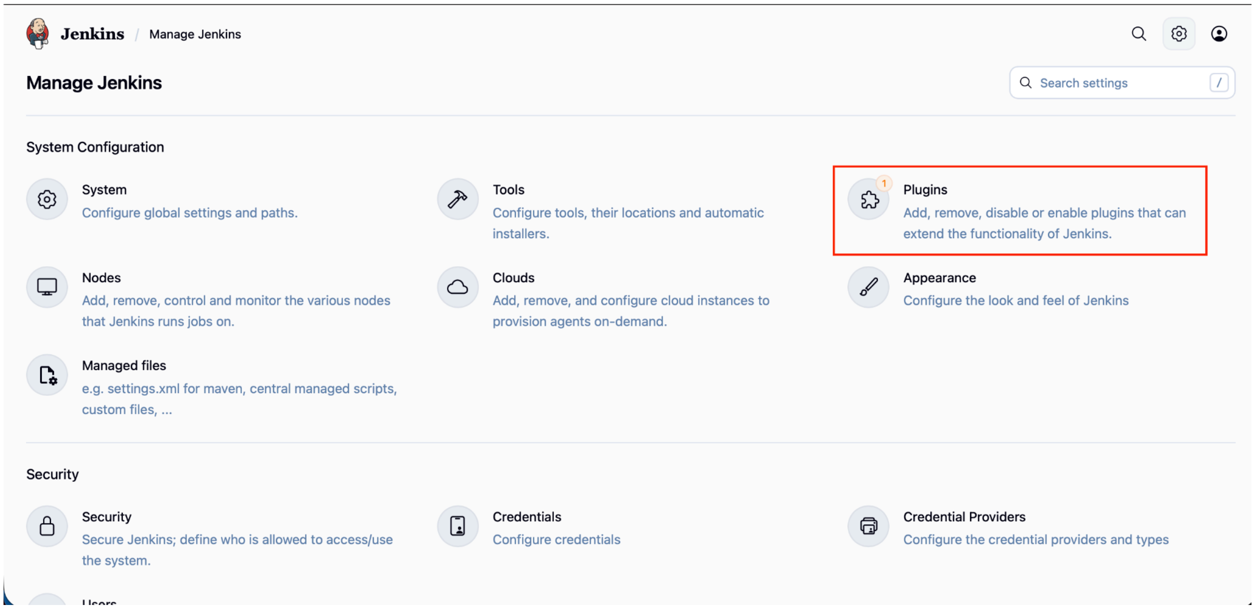
C. Required Jenkins Plugins

Go to **Manage Jenkins** > **Manage Plugins** > **Available** and install:

1. Deploy to container
2. Maven integration
3. Config file provider
4. Pipeline maven integration
5. Pipeline plugin api
6. Git Plugin



S	W	Name ↓	Last Success	Last Failure	Last Duration
✓		Maven-pipeline	1 hr 41 min #11	1 hr 44 min #10	4.8 sec
✓		Maven-pipeline2	1 hr 19 min #4	1 hr 22 min #3	2 min 8 sec
✓		Maven-pipeline3	1 hr 10 min #2	1 hr 12 min #1	9.1 sec



Manage Jenkins

System Configuration

System
Configure global settings and paths.

Tools
Configure tools, their locations and automatic installers.

Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

Nodes
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Clouds
Add, remove, and configure cloud instances to provision agents on-demand.

Appearance
Configure the look and feel of Jenkins

Managed files
e.g. settings.xml for maven, central managed scripts, custom files, ...

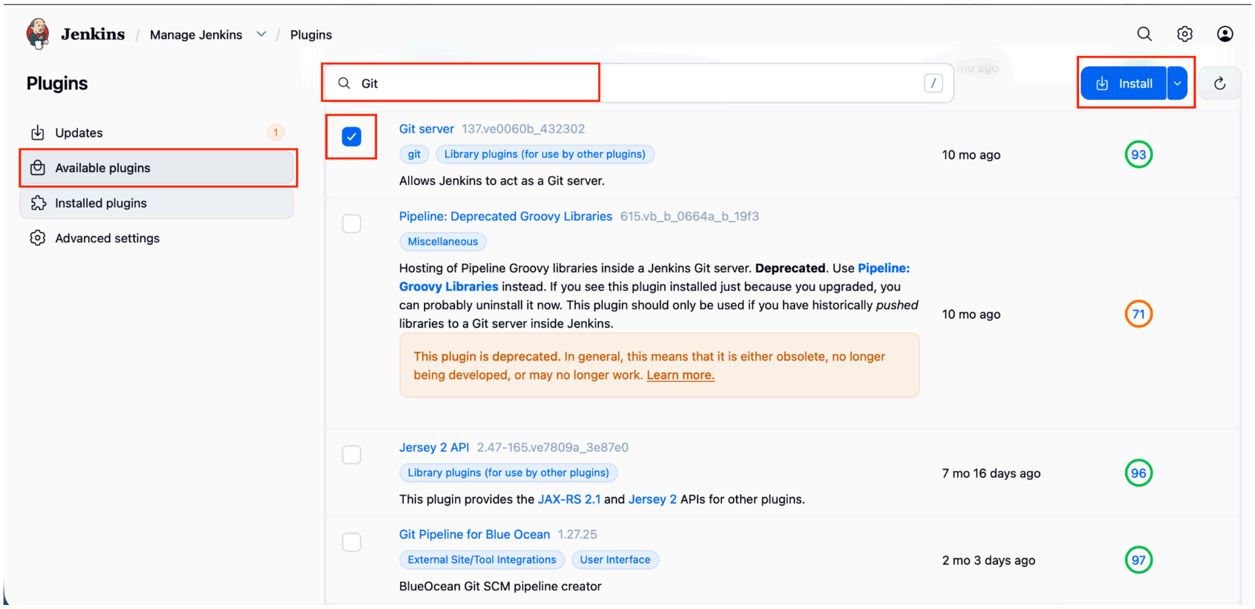
Security

Security
Secure Jenkins; define who is allowed to access/use the system.

Credentials
Configure credentials

Credential Providers
Configure the credential providers and types

Users



☑ Git server 137.ve0060b_432302
git | Library plugins (for use by other plugins)
Allows Jenkins to act as a Git server.
10 mo ago 93

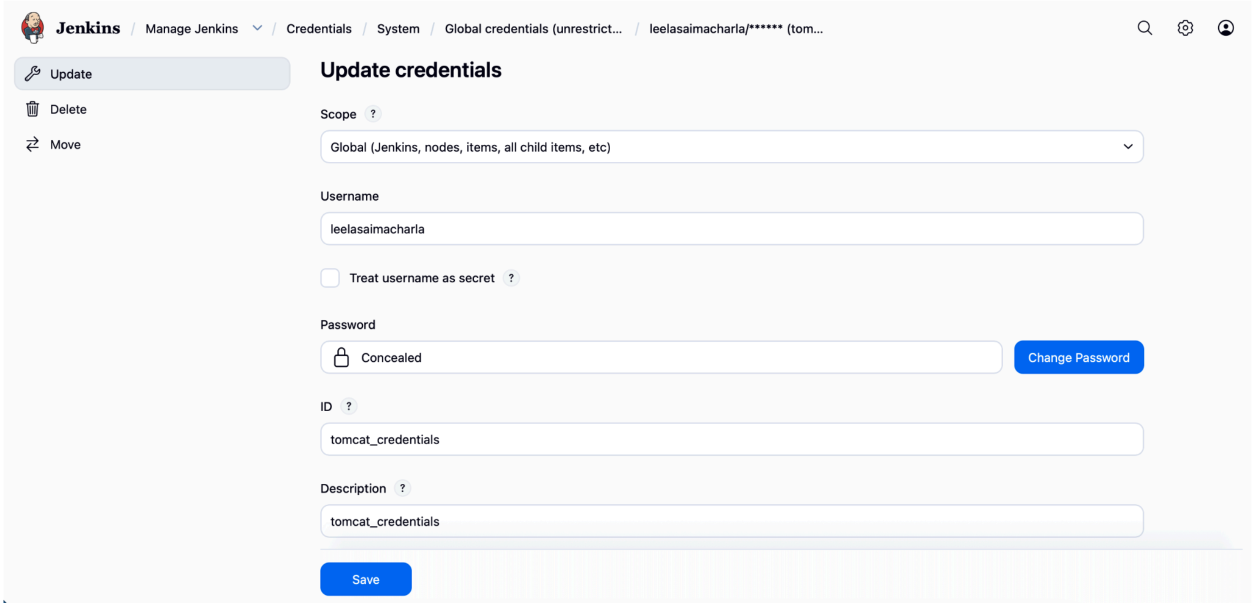
□ Pipeline: Deprecated Groovy Libraries 615.vb_b_0664a_b_19f3
Miscellaneous
Hosting of Pipeline Groovy libraries inside a Jenkins Git server. **Deprecated**. Use **Pipeline: Groovy Libraries** instead. If you see this plugin installed just because you upgraded, you can probably uninstall it now. This plugin should only be used if you have historically *pushed* libraries to a Git server inside Jenkins.
This plugin is deprecated. In general, this means that it is either obsolete, no longer being developed, or may no longer work. Learn more.
10 mo ago 71

□ Jersey 2 API 2.47-165.ve7809a_3e87e0
Library plugins (for use by other plugins)
This plugin provides the JAX-RS 2.1 and Jersey 2 APIs for other plugins.
7 mo 16 days ago 96

□ Git Pipeline for Blue Ocean 1.27.25
External Site/Tool Integrations | User Interface
BlueOcean Git SCM pipeline creator
2 mo 3 days ago 97

D. Give the tomcat credentials

Go to **Manage Jenkins** > **Credentials** and give all the required fields. Then save.


Jenkins / Manage Jenkins / Credentials / System / Global credentials (unrestrict... / leelasaimacharla/****** (tom...
Update
Delete
Move
Change Password
Save

Update credentials

Scope ?
Global (Jenkins, nodes, items, all child items, etc)

Username
leelasaimacharla

□ Treat username as secret ?

Password
Concealed

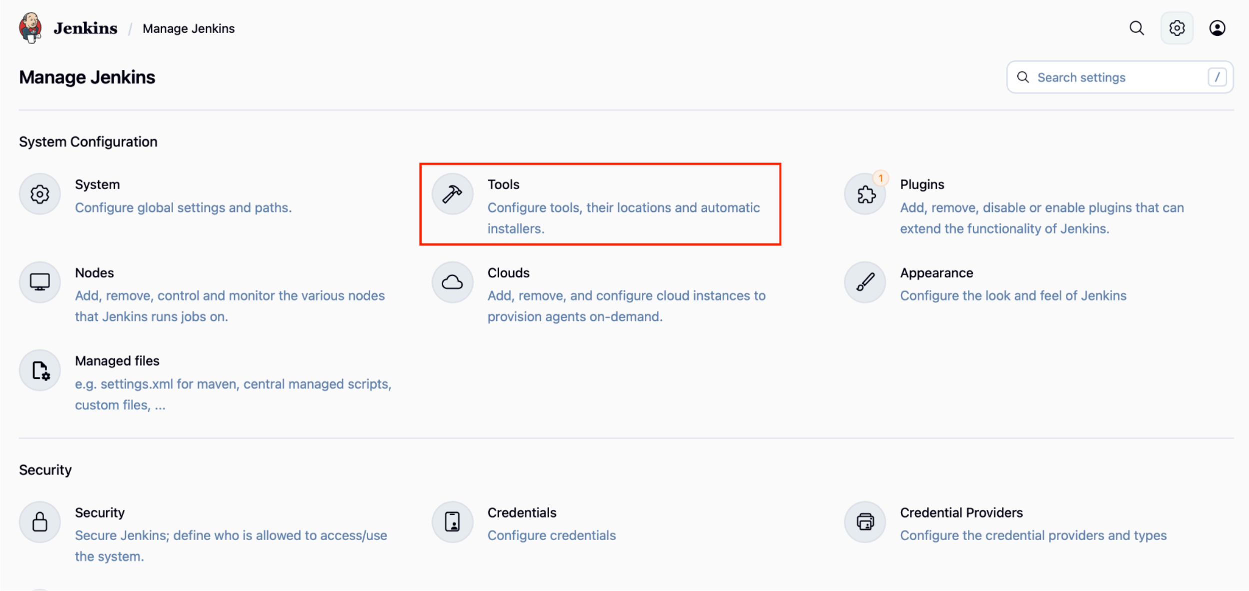
ID ?
tomcat_credentials

Description ?
tomcat_credentials

2. Jenkins Global Tool Configuration

You must tell Jenkins exactly where your Java and Maven are located on your Laptop.

1. Go to **Manage Jenkins** > **Global Tool Configuration**.



Jenkins / Manage Jenkins

Manage Jenkins

System Configuration

System
Configure global settings and paths.

Tools
Configure tools, their locations and automatic installers.

Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

Nodes
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Clouds
Add, remove, and configure cloud instances to provision agents on-demand.

Appearance
Configure the look and feel of Jenkins

Managed files
e.g. settings.xml for maven, central managed scripts, custom files, ...

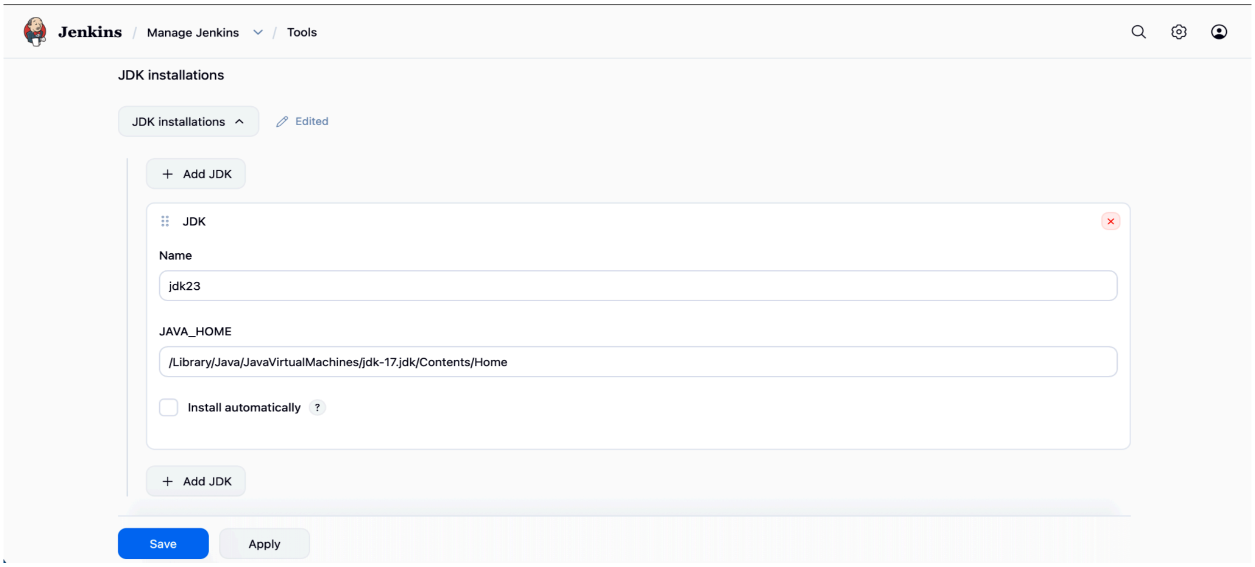
Security

Security
Secure Jenkins; define who is allowed to access/use the system.

Credentials
Configure credentials

Credential Providers
Configure the credential providers and types

2. **JDK:** Add JDK, name it "JDK17", and set the path to:
   - `/Library/Java/JavaVirtualMachines/jdk-17.jdk/Contents/Home`



Jenkins / Manage Jenkins / Tools

JDK installations

JDK installations — Edited

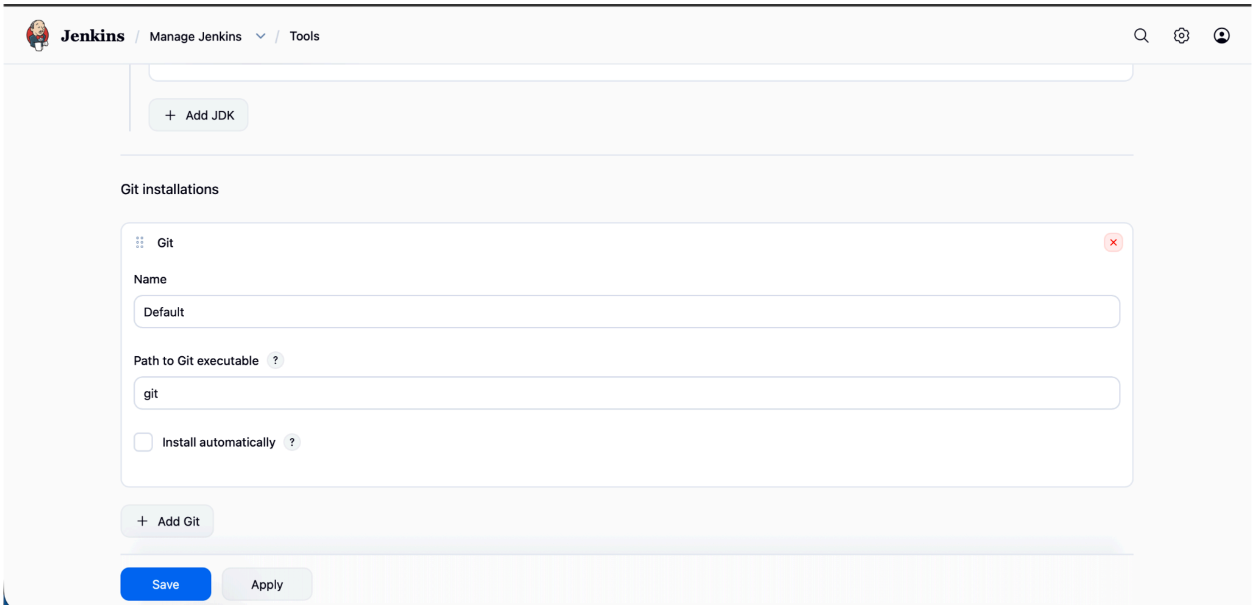
JDK

Name: jdk23

JAVA_HOME: /Library/Java/JavaVirtualMachines/jdk-17.jdk/Contents/Home

□ Install automatically

3. **Git:** Add Git, name it "Git", and check "Install automatically" or provide the path if installed.



Jenkins / Manage Jenkins / Tools

+ Add JDK

Git installations

Git

Name

Default

Path to Git executable ?

git

☐ Install automatically ?

+ Add Git

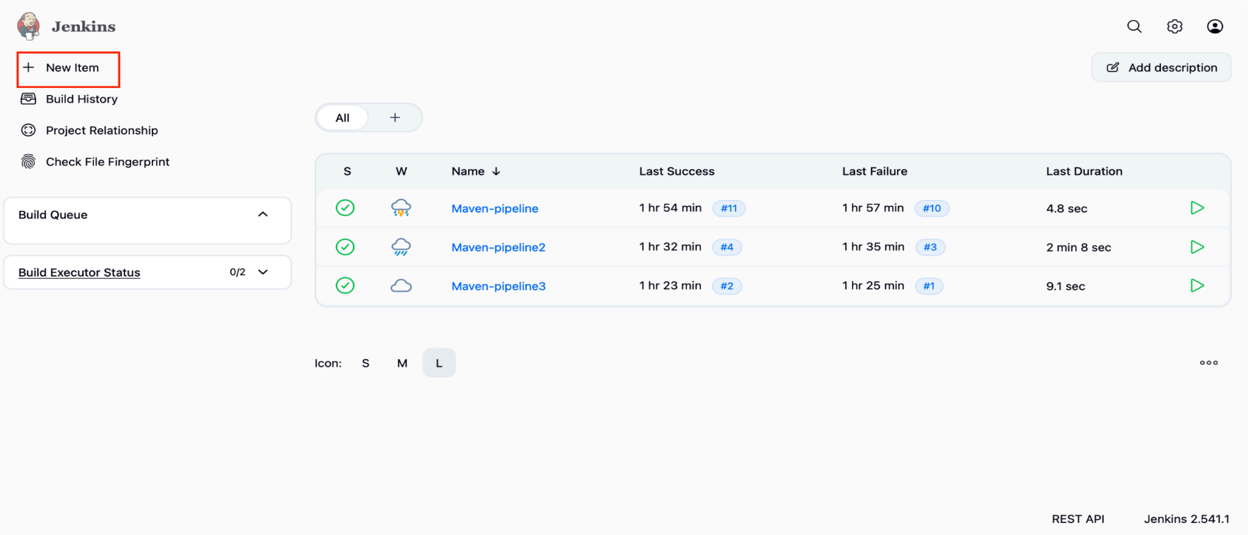
Save Apply

.

3. Creating the Jenkins Pipeline (Step-by-Step)

Step 1: General & Source Code Management (SCM)

1. **New Item** > Enter `EcommerceApp-Pipeline` > Select **Maven project**.



Jenkins

+ New Item

Build History

Project Relationship

Check File Fingerprint

Build Queue

Build Executor Status 0/2

Add description

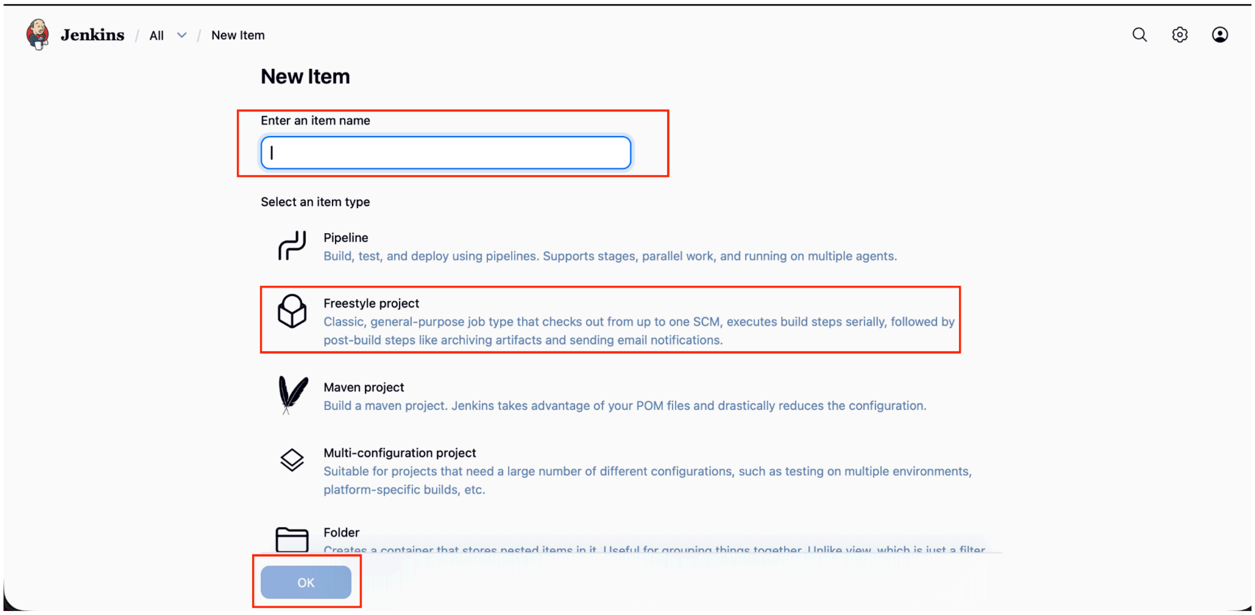
All +

S	W	Name ↓	Last Success	Last Failure	Last Duration
✓		Maven-pipeline	1 hr 54 min #11	1 hr 57 min #10	4.8 sec
✓		Maven-pipeline2	1 hr 32 min #4	1 hr 35 min #3	2 min 8 sec
✓		Maven-pipeline3	1 hr 23 min #2	1 hr 25 min #1	9.1 sec

Icon: S M L

REST API Jenkins 2.541.1

2. When you click the new item you will be directed to the next page as shown below. Now give the name, select Freestyle project and click ok.



Jenkins / All / New Item

New Item

Enter an item name

Select an item type

Pipeline
Build, test, and deploy using pipelines. Supports stages, parallel work, and running on multiple agents.

Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

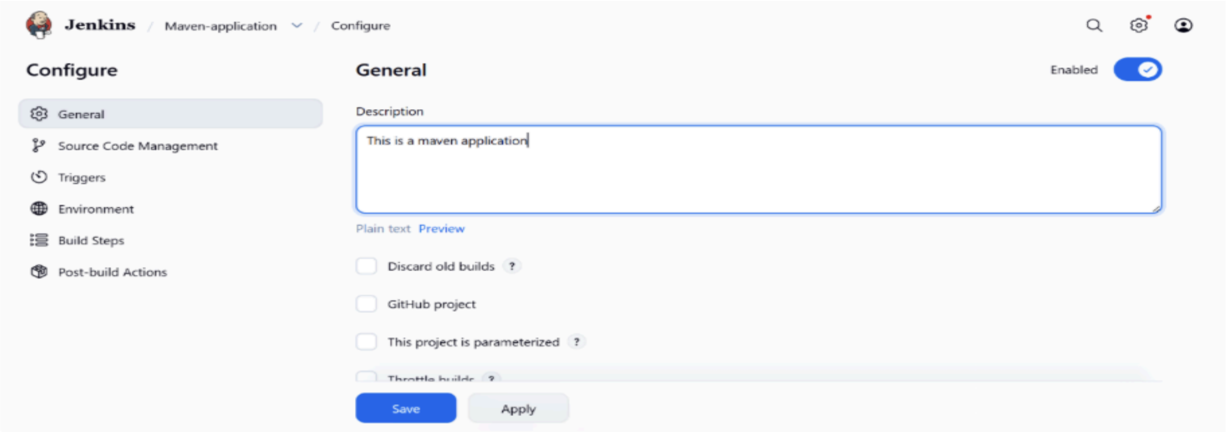
Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter

OK

3. In general add a description.



Jenkins / Maven-application / Configure

Configure

- General
- Source Code Management
- Triggers
- Environment
- Build Steps
- Post-build Actions

General

Enabled

Description

This is a maven application

Plain text Preview

□ Discard old builds ?

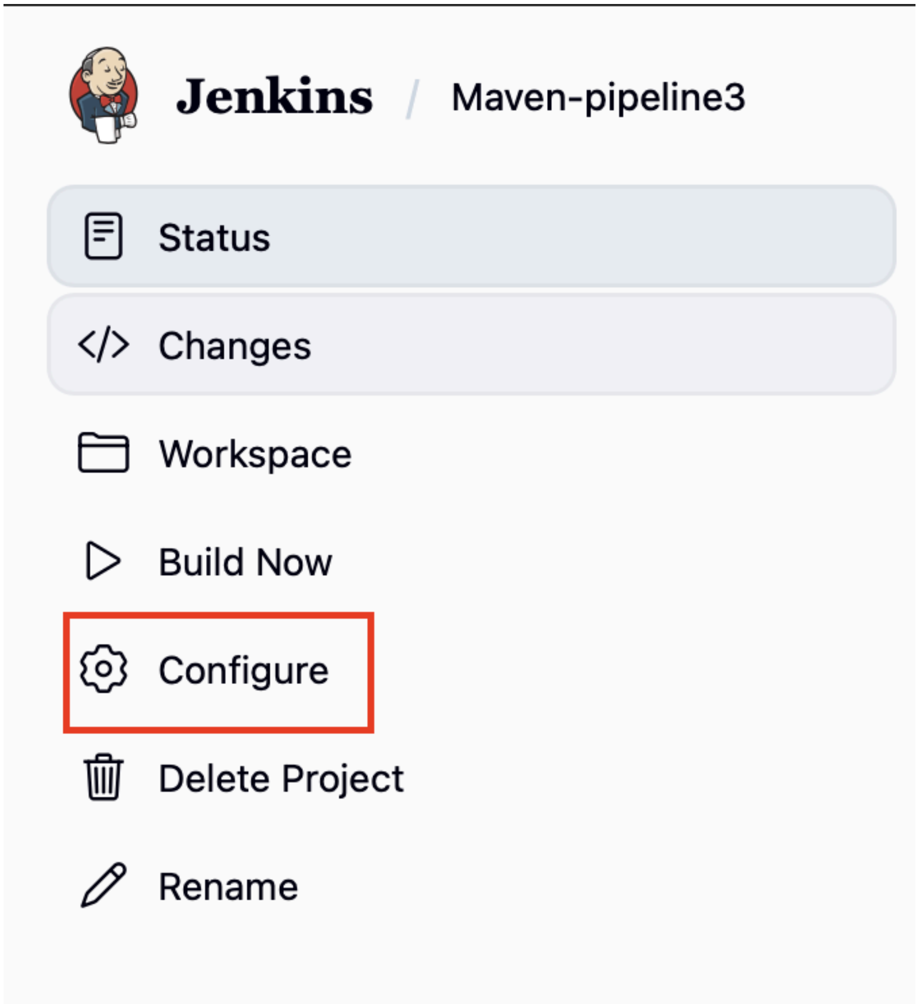
□ GitHub project

□ This project is parameterized ?

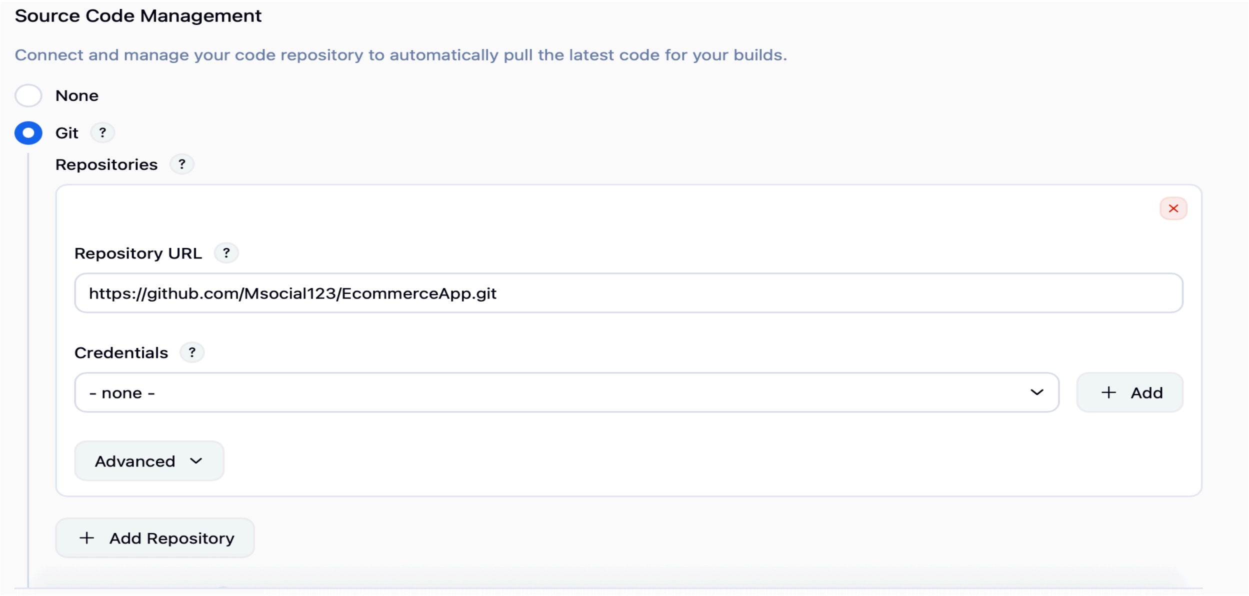
□ Throttle builds ?

Save Apply

4. Click on configure. Then we will go to configure page



5. Now if we scroll little bit down we will find **Source Code Management** section


Source Code Management

Connect and manage your code repository to automatically pull the latest code for your builds.

○ None

● Git

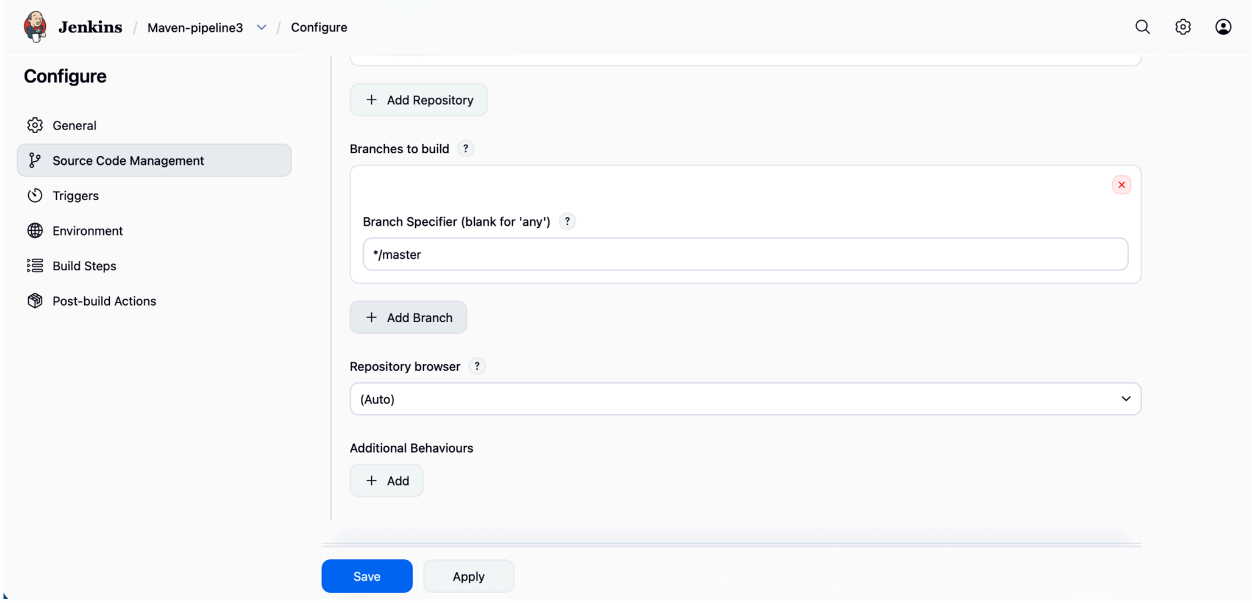
Repositories

Repository URL: https://github.com/Msocial123/EcommerceApp.git

Credentials: - none -

By default it will be **None** we need to change it to **Git** and provide the repo url in which the project is present.

6. Now if we scroll a little down we will find **Branches to build** section. Here we need to provide in which branch the project is present(main/master).



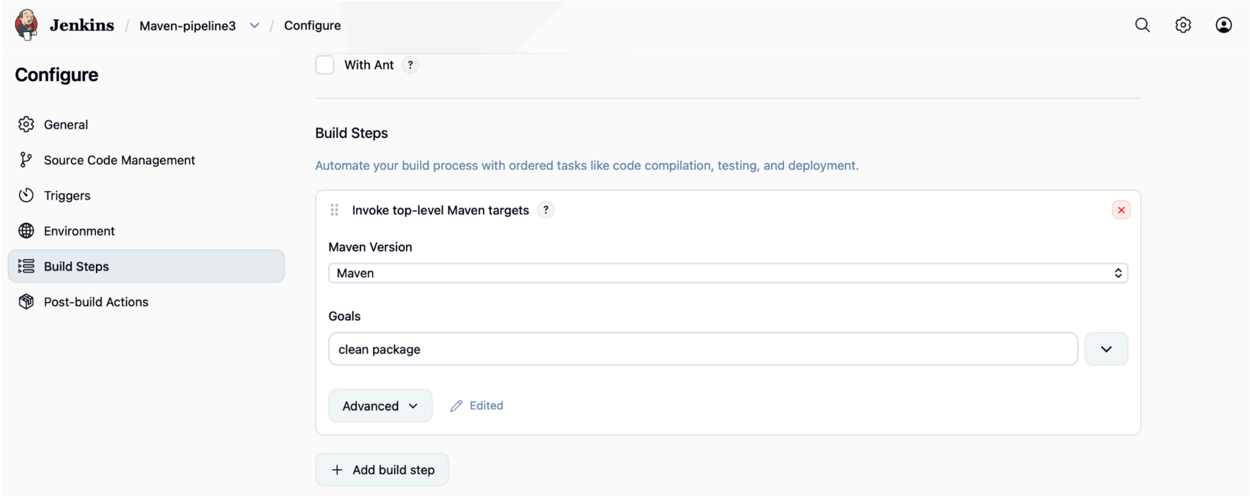
Branch Specifier (blank for 'any'): */master

Repository browser: (Auto)

7. Leave Triggers and Environment sections

8. **Build Settings**

1. **Root POM:** `EcommerceApp/pom.xml` (Based on your logs, the POM is inside a subfolder).
2. **Goals and options:** `clean package`



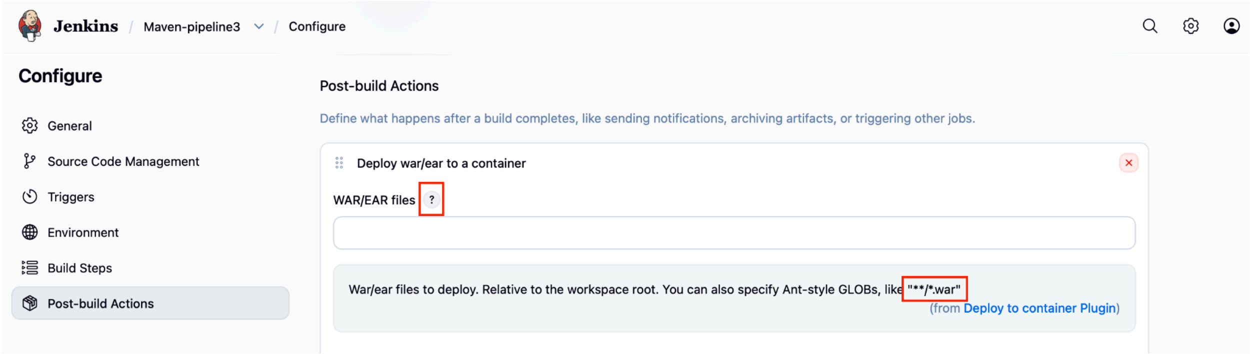
Automate your build process with ordered tasks like code compilation, testing, and deployment.

Maven Version: Maven

Goals: clean package

9. Now go to **Post_build Actions**

Click the question mark we will get text bar below copy ****/*war** without (“ ”)

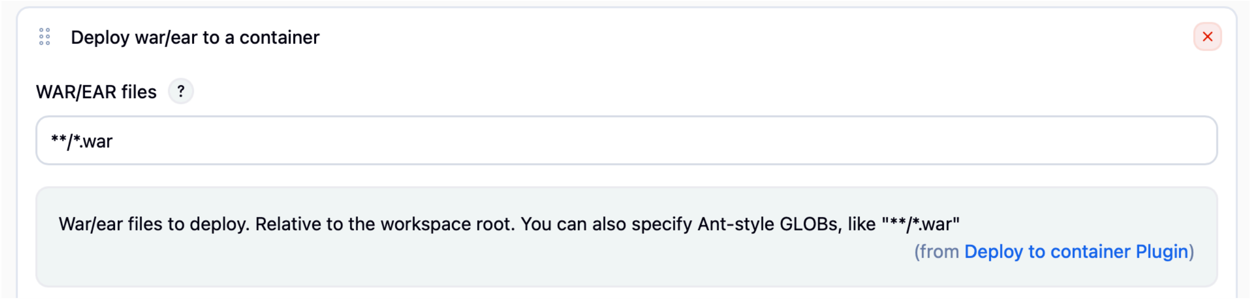


Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

WAR/EAR files

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

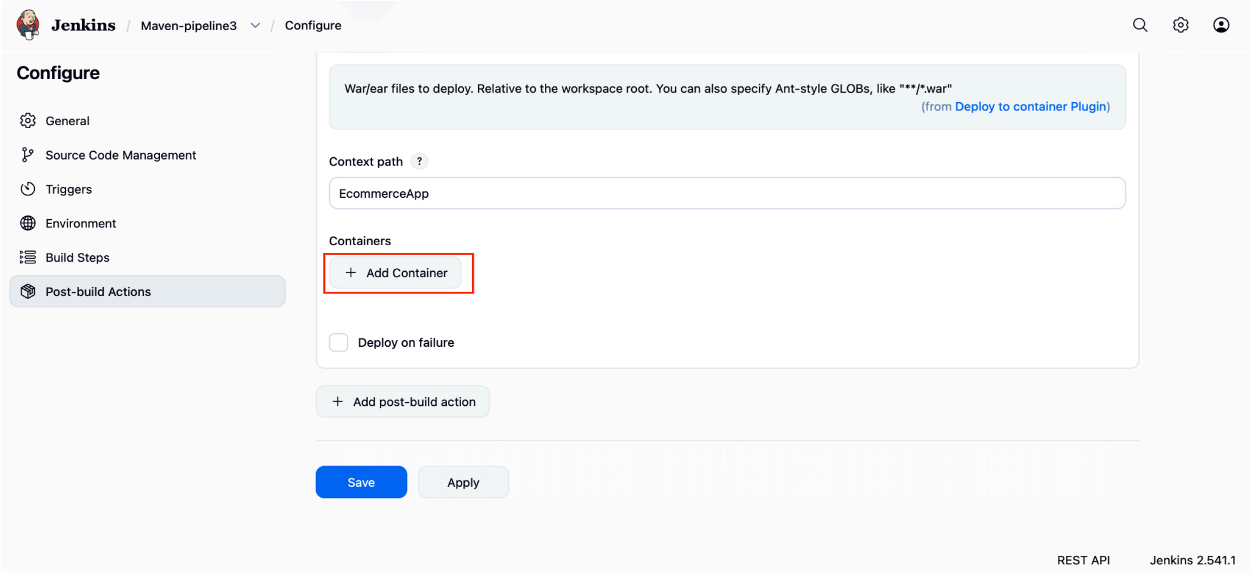


WAR/EAR files: **/*.war

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

10. Give **Context Path**

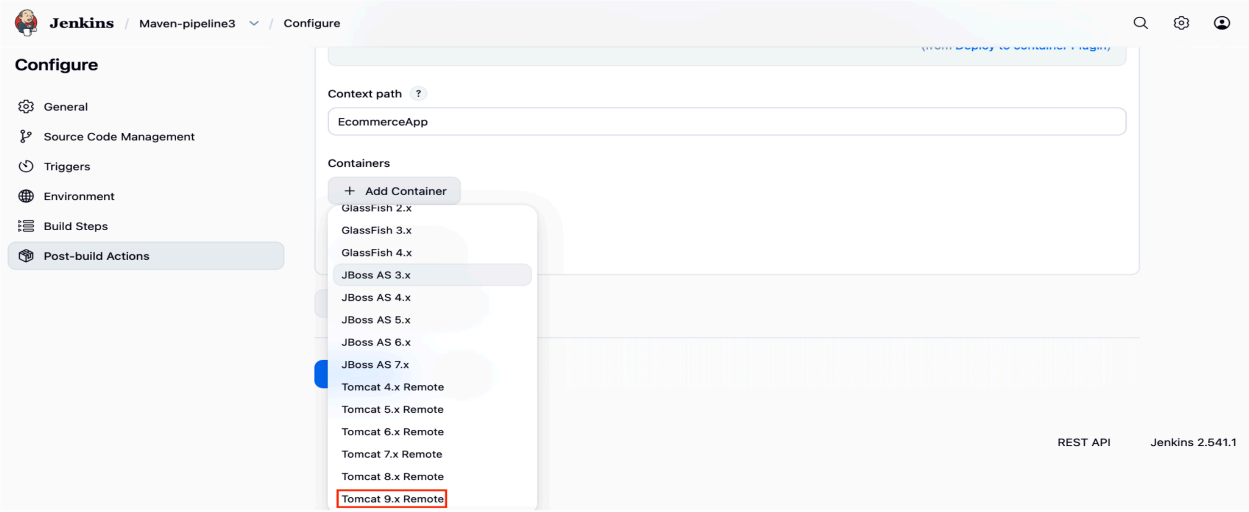
11. Now for **Containers** click add container and choose **Tomcat9x Remote**



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

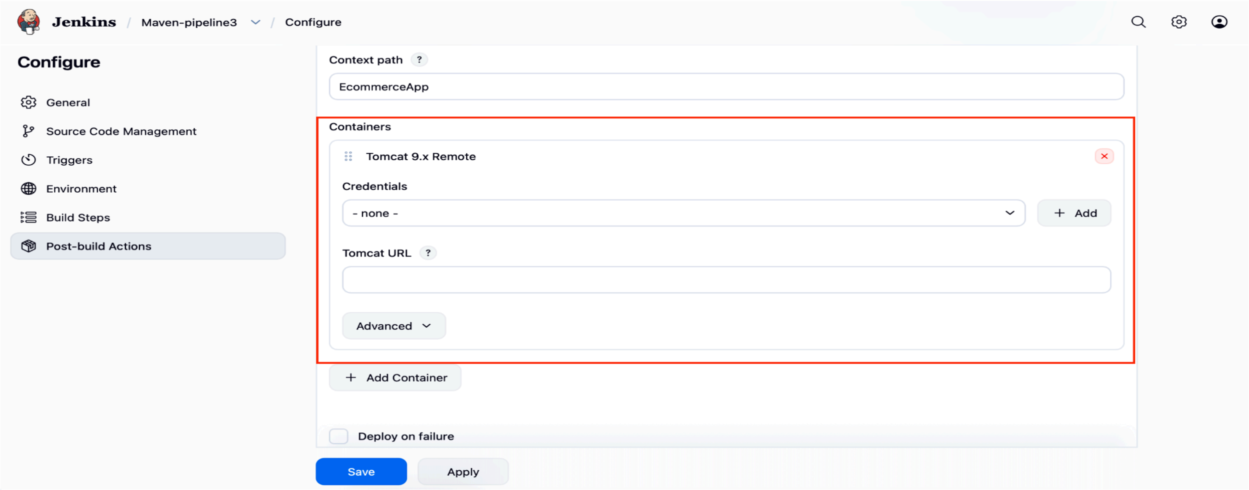
Context path: EcommerceApp

□ Deploy on failure



Context path: EcommerceApp

Containers



Context path: EcommerceApp

Containers

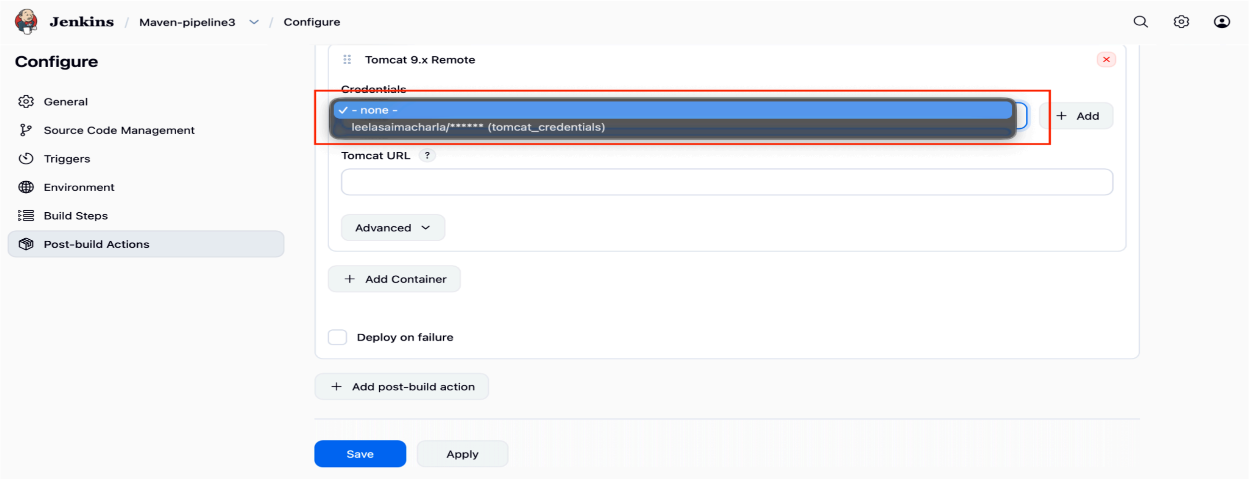
Tomcat 9.x Remote

Credentials: - none -

Tomcat URL:

□ Deploy on failure

12. Now Click none and select the previous created credentials and give the Tomcat url



Tomcat 9.x Remote

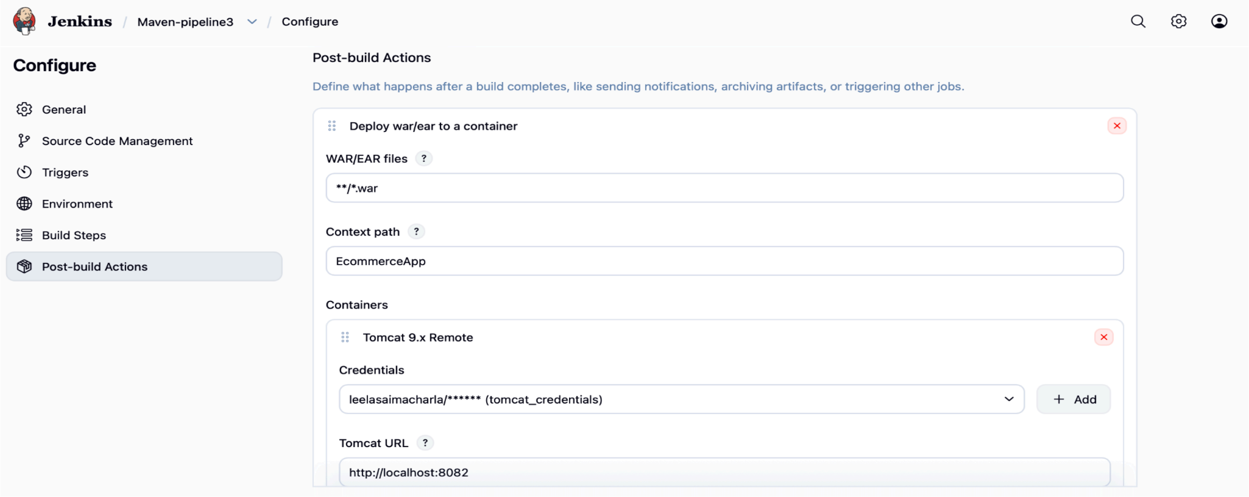
Credentials

✓ - none -

leelasaimacharla/****** (tomcat_credentials)

Tomcat URL:

□ Deploy on failure



Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

Deploy war/ear to a container

WAR/EAR files: `**/*.war`

Context path: EcommerceApp

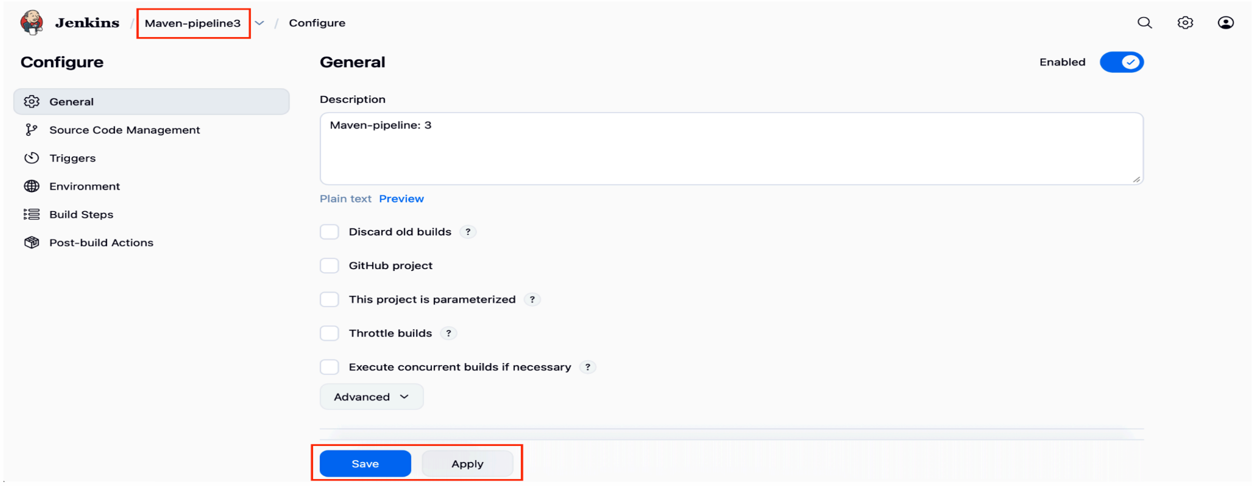
Containers

Tomcat 9.x Remote

Credentials: leelasaimacharla/****** (tomcat_credentials)

Tomcat URL: http://localhost:8082

13. Now click **apply and save.** After successfully saving click on pipeline name (“**Maven-pipeline3**”)



General

Enabled

Description: Maven-pipeline: 3

□ Discard old builds

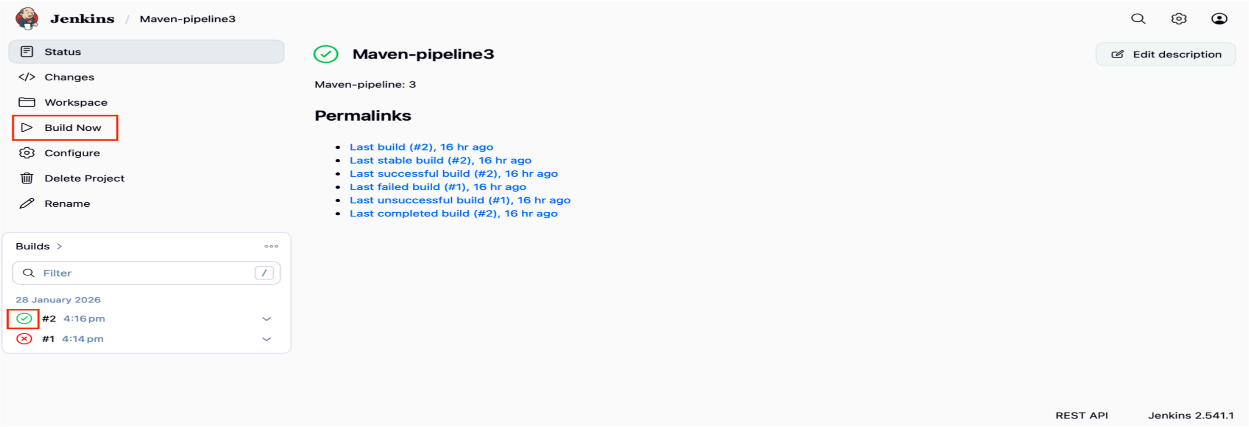
□ GitHub project

□ This project is parameterized

□ Throttle builds

□ Execute concurrent builds if necessary

14. Now to run the pipeline click on **Build Now**



Maven-pipeline3

Maven-pipeline: 3

Permalinks

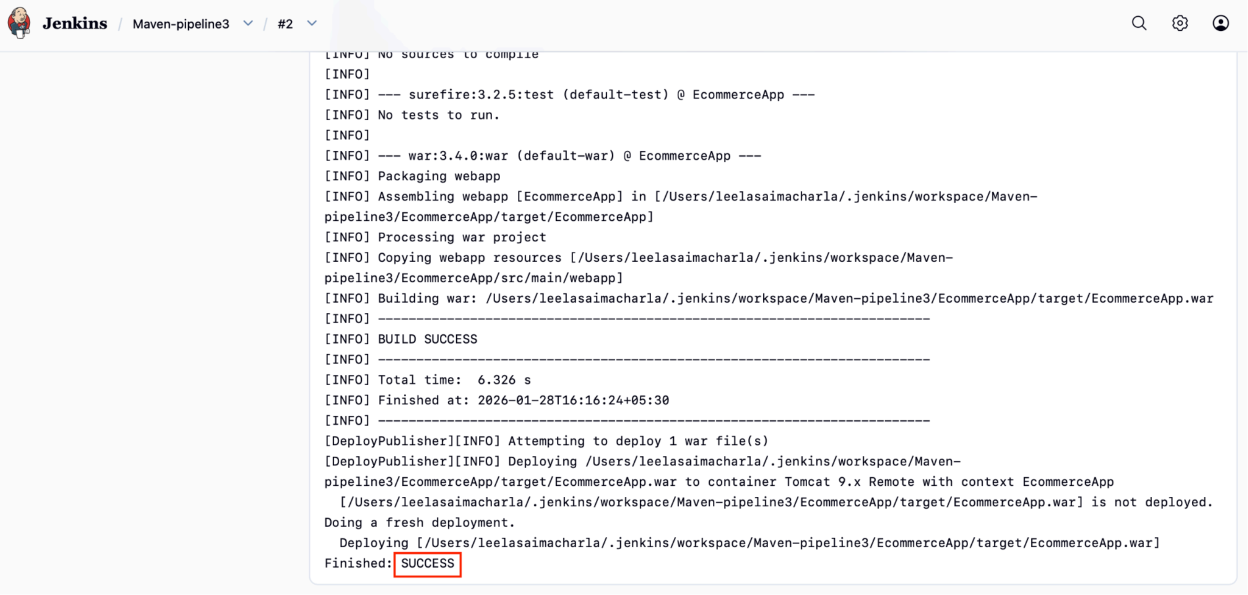
- Last build (#2), 16 hr ago
- Last stable build (#2), 16 hr ago
- Last successful build (#2), 16 hr ago
- Last failed build (#1), 16 hr ago
- Last unsuccessful build (#1), 16 hr ago
- Last completed build (#2), 16 hr ago

28 January 2026

#2 4:16 pm

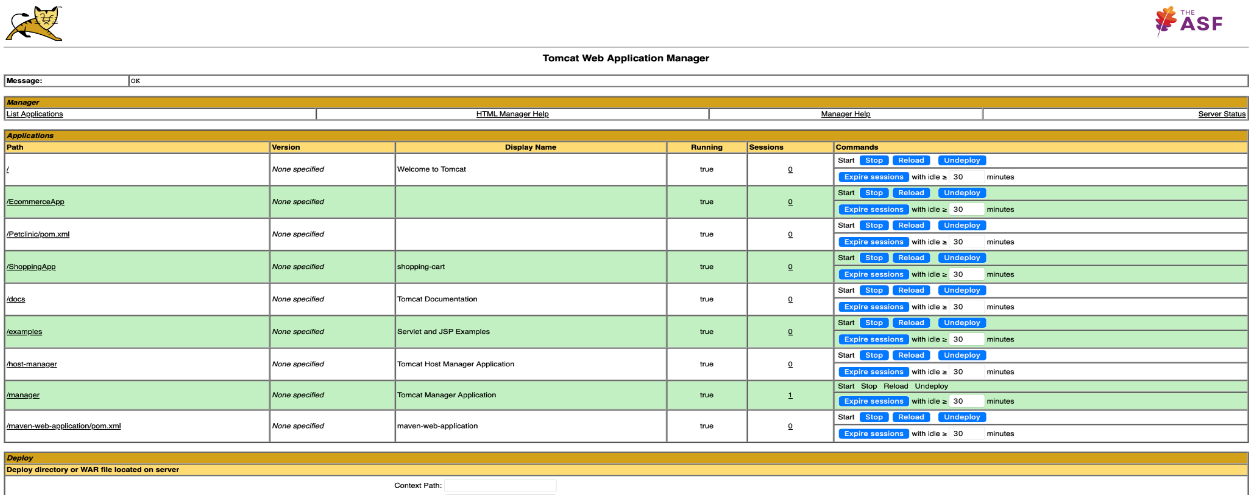
#1 4:14 pm

15. If we click the tick mark or wrong mark we will be redirected to the console output. If we find **success** at the last line then our pipeline is running successfully else we can see the errors and we should rectify them.



```
[INFO] No sources to compile
[INFO]
[INFO] --- surefire:3.2.5:test (default-test) @ EcommerceApp ---
[INFO] No tests to run.
[INFO]
[INFO] --- war:3.4.0:war (default-war) @ EcommerceApp ---
[INFO] Packaging webapp
[INFO] Assembling webapp [EcommerceApp] in [/Users/leelasaimacharla/.jenkins/workspace/Maven-
pipeline3/EcommerceApp/target/EcommerceApp]
[INFO] Processing war project
[INFO] Copying webapp resources [/Users/leelasaimacharla/.jenkins/workspace/Maven-
pipeline3/EcommerceApp/src/main/webapp]
[INFO] Building war: /Users/leelasaimacharla/.jenkins/workspace/Maven-pipeline3/EcommerceApp/target/EcommerceApp.war
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  6.326 s
[INFO] Finished at: 2026-01-28T16:16:24+05:30
[INFO] ------------------------------------------------------------------------
[DeployPublisher][INFO] Attempting to deploy 1 war file(s)
[DeployPublisher][INFO] Deploying /Users/leelasaimacharla/.jenkins/workspace/Maven-
pipeline3/EcommerceApp/target/EcommerceApp.war to container Tomcat 9.x Remote with context EcommerceApp
  [/Users/leelasaimacharla/.jenkins/workspace/Maven-pipeline3/EcommerceApp/target/EcommerceApp.war] is not deployed.
Doing a fresh deployment.
  Deploying [/Users/leelasaimacharla/.jenkins/workspace/Maven-pipeline3/EcommerceApp/target/EcommerceApp.war]
Finished: SUCCESS
```

16 . Go to Tomcat Server and Click on Manager App and Login with Tomcat Credentials



Tomcat Web Application Manager

Message:	OK

Manager			
List Applications	HTML Manager Help	Manager Help	Server Status

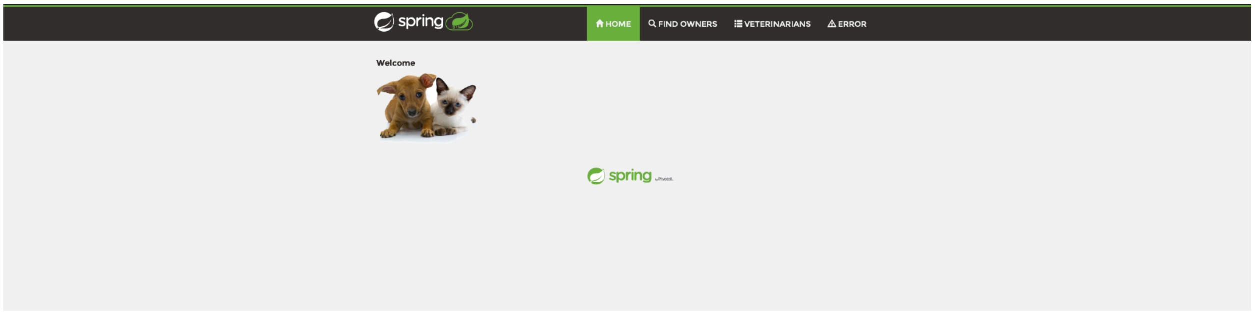
Applications

Path	Version	Display Name	Running	Sessions	Commands
/	*None specified*	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/EcommerceApp	*None specified*		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/Petclinic/pom.xml	*None specified*		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/ShoppingApp	*None specified*	shopping-cart	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	*None specified*	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	*None specified*	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	*None specified*	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	*None specified*	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/maven-web-application/pom.xml	*None specified*	maven-web-application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Deploy

Deploy directory or WAR file located on server

Context Path:



Welcome