## **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**

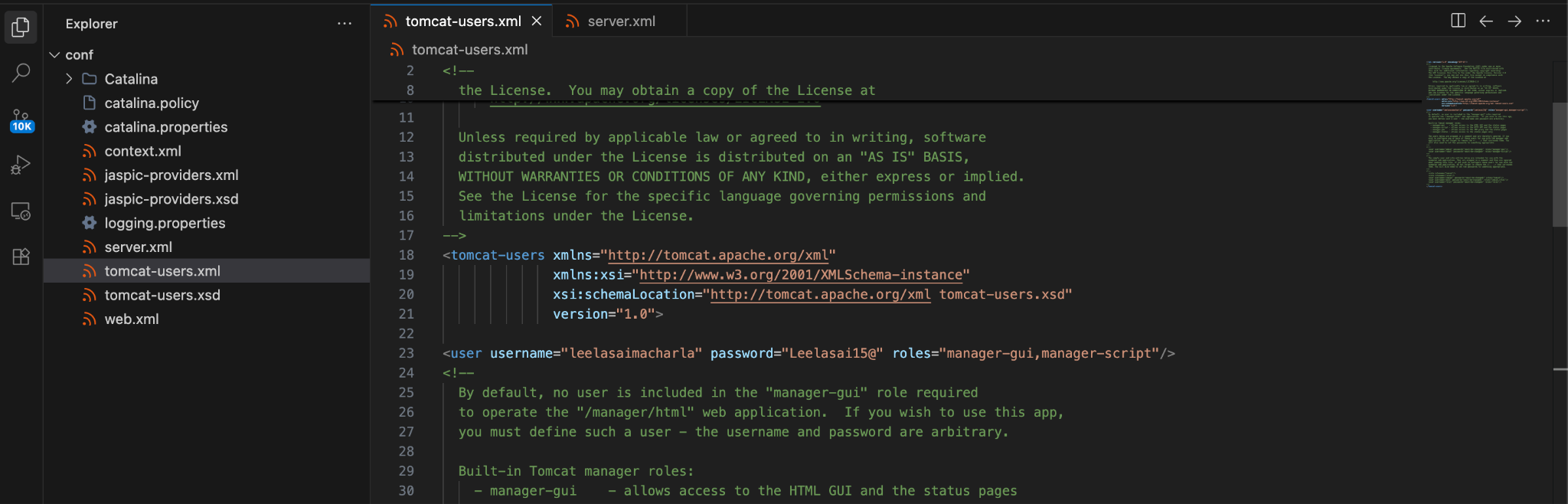
1. Install jenkins
2. Install Tomcat@9
3. Install Maven
4. Install JDK (with respect to the tomcat)
5. Install Git

### **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"/>

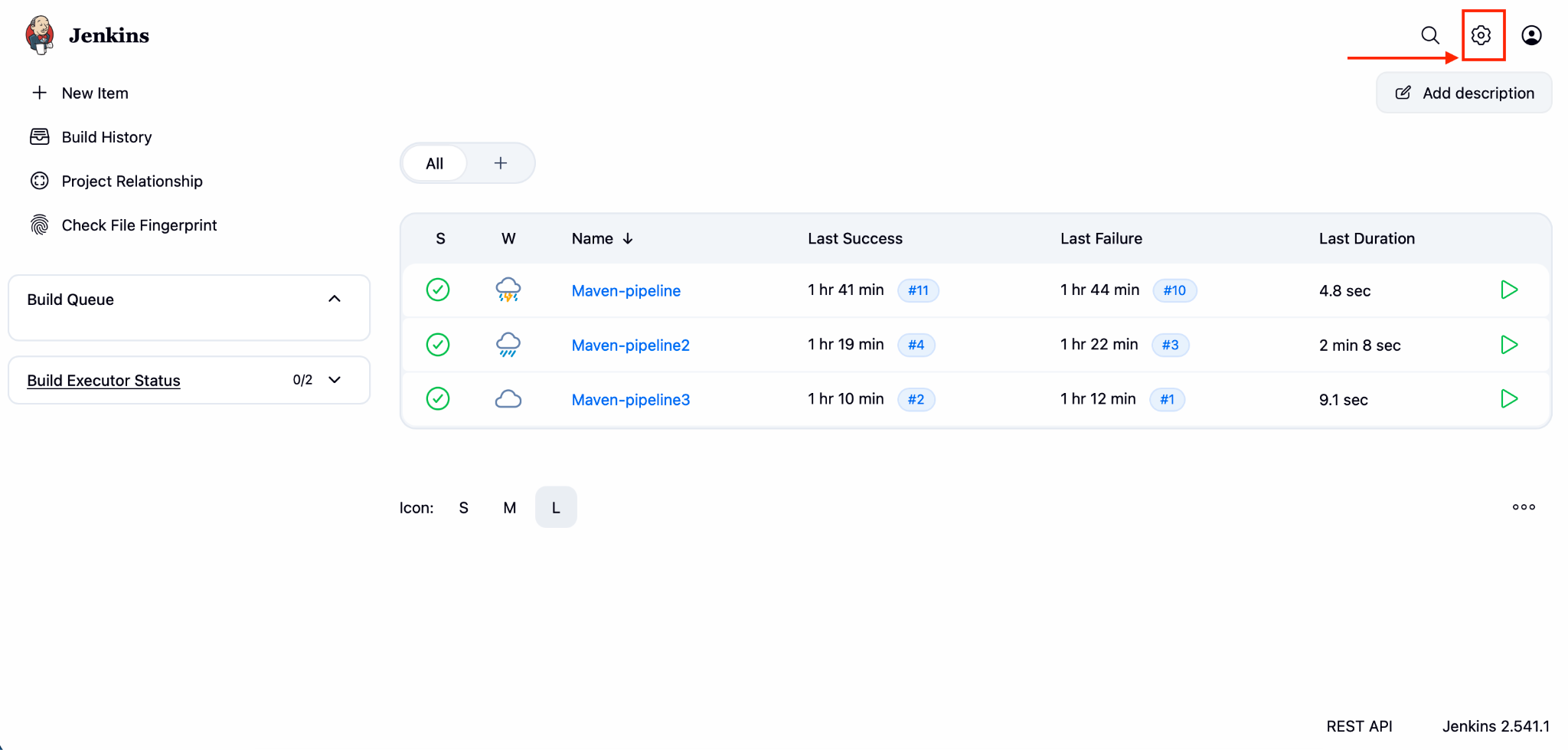


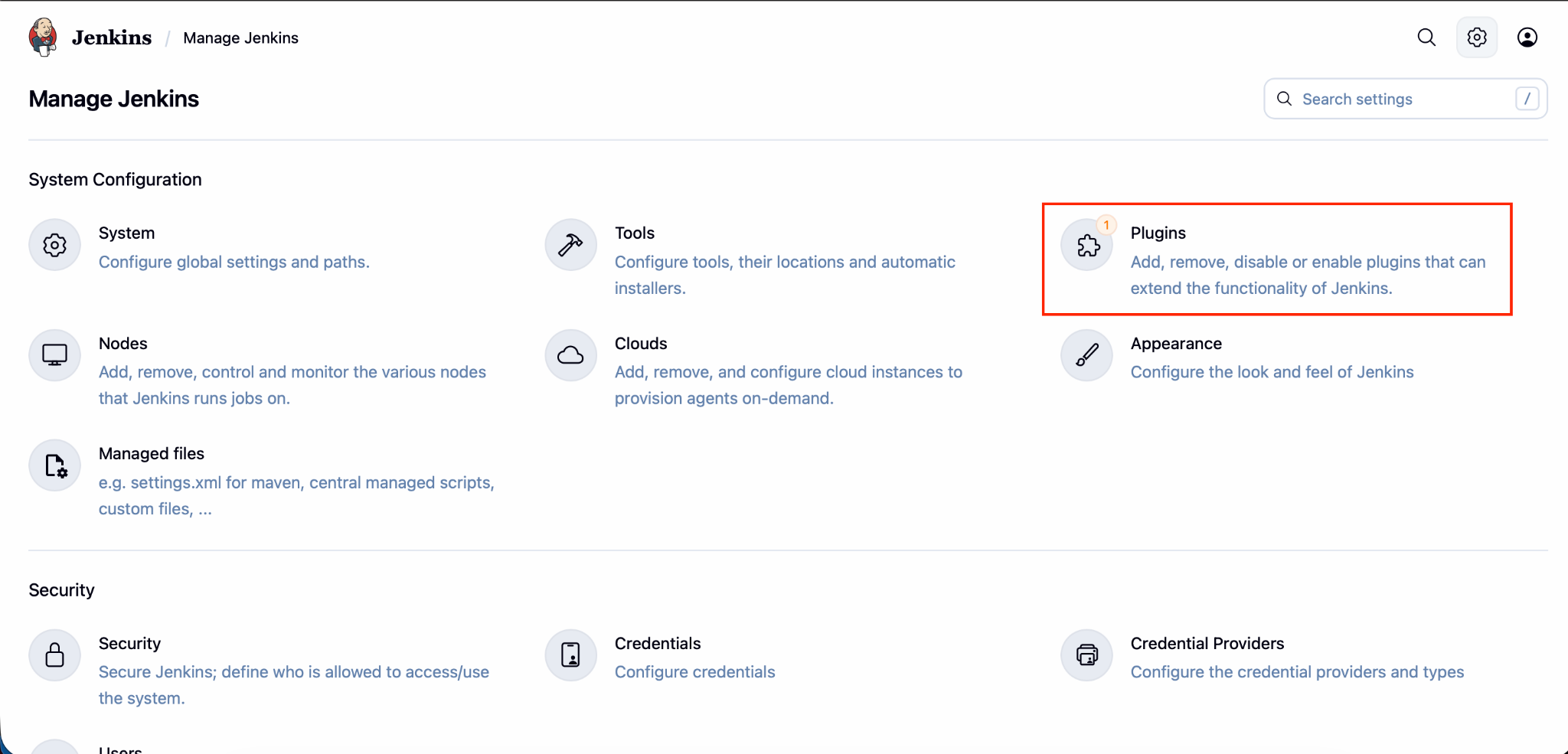
**IMP** : Make sure the port numbers and server ports of Jenkins and Tomcat are configured (shouldn’t use same port number or server port)

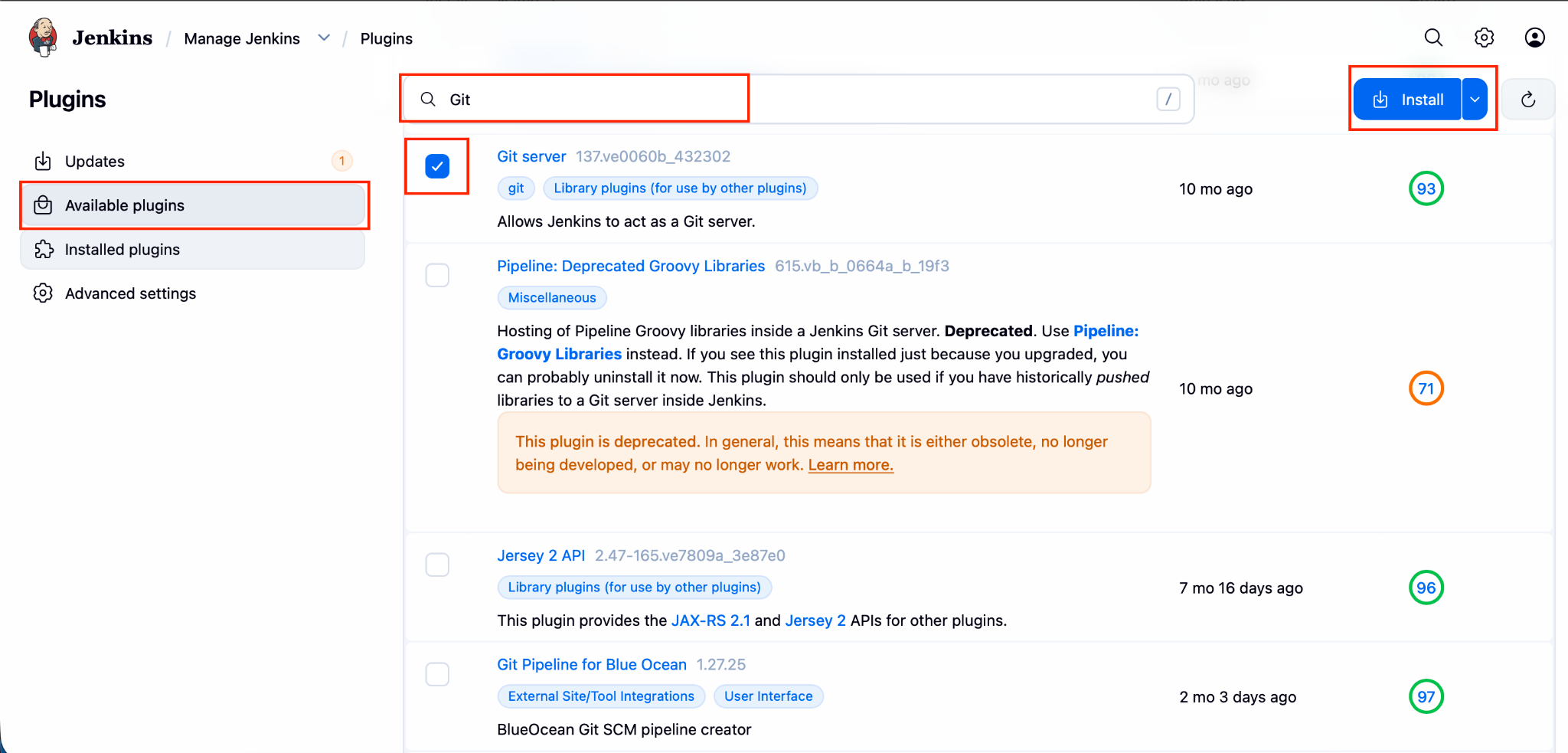
### **C. Required Jenkins Plugins**

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







### **D. Give the tomcat credentials**

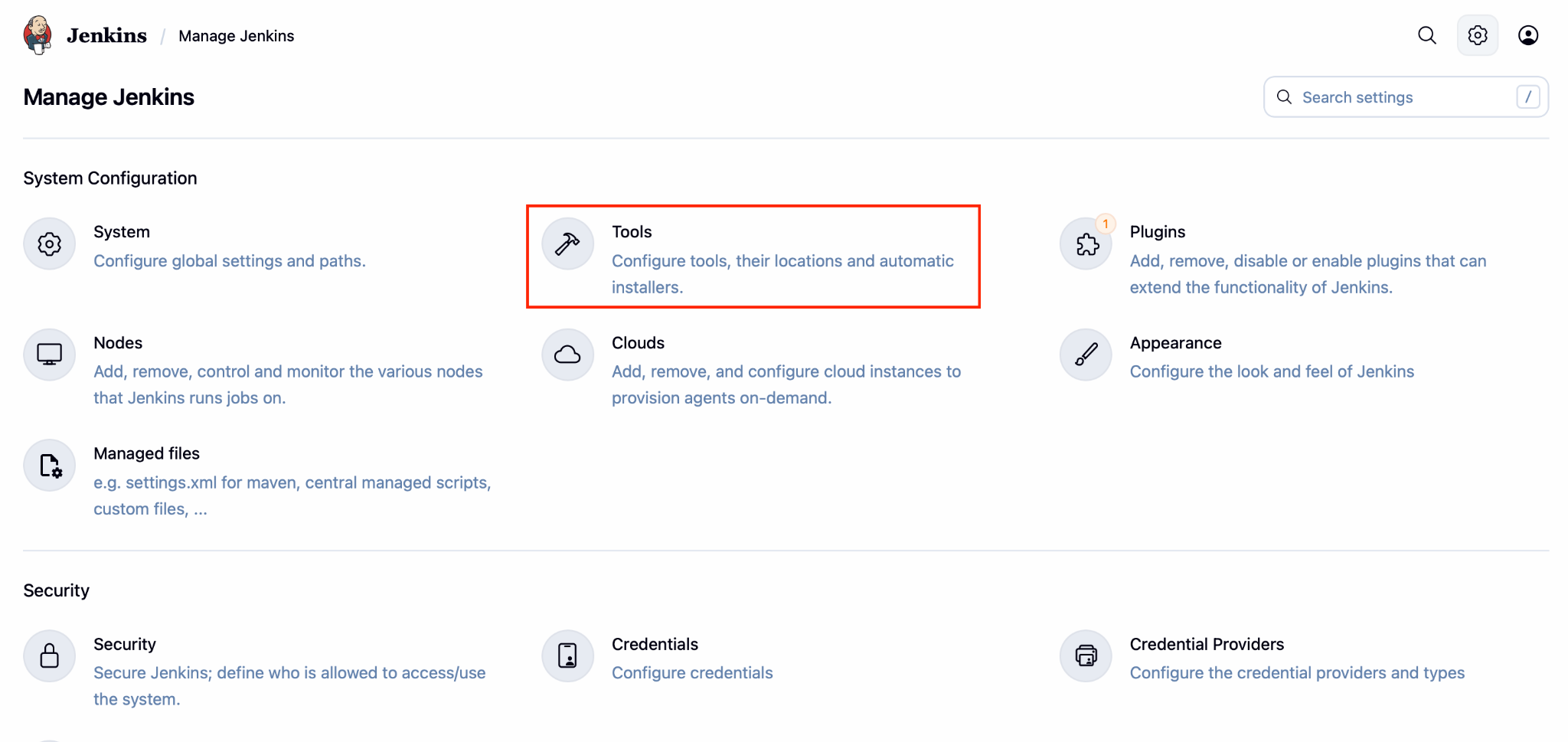
Go to **Manage Jenkins** > **Credentials** and give all the required fields. Then save.



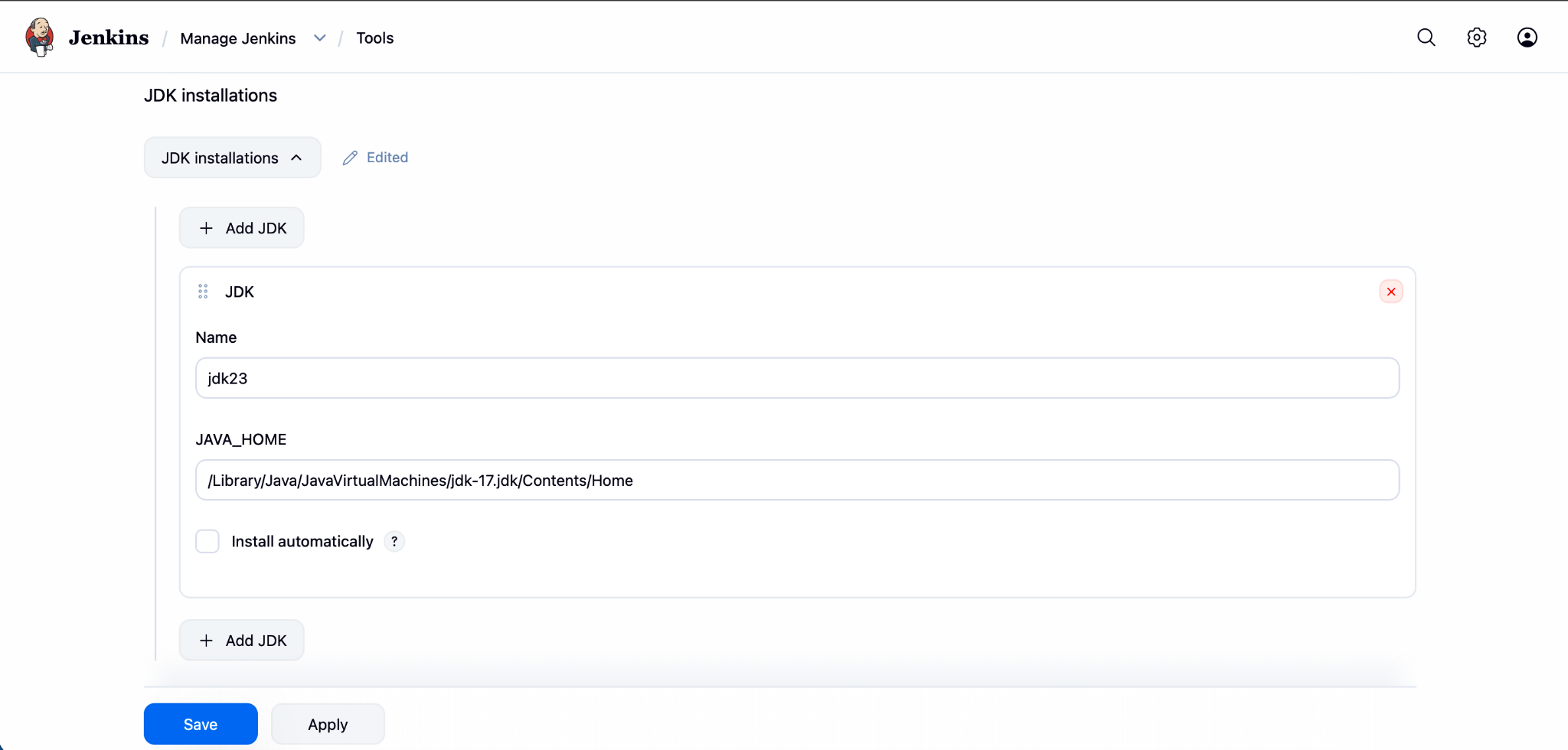
## **2. Jenkins Global Tool Configuration**

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

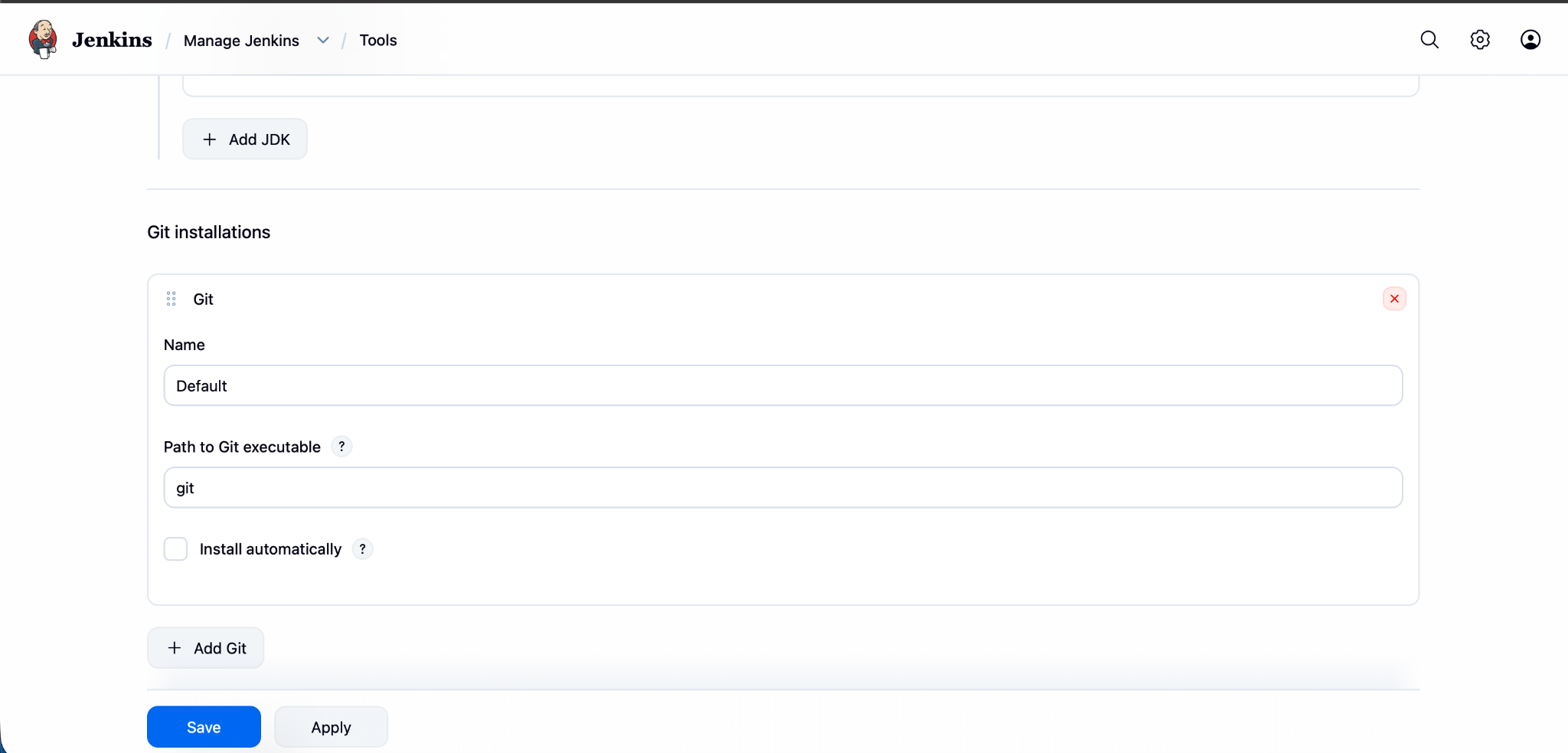
1. Go to **Manage Jenkins** > **Global Tool Configuration**.



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



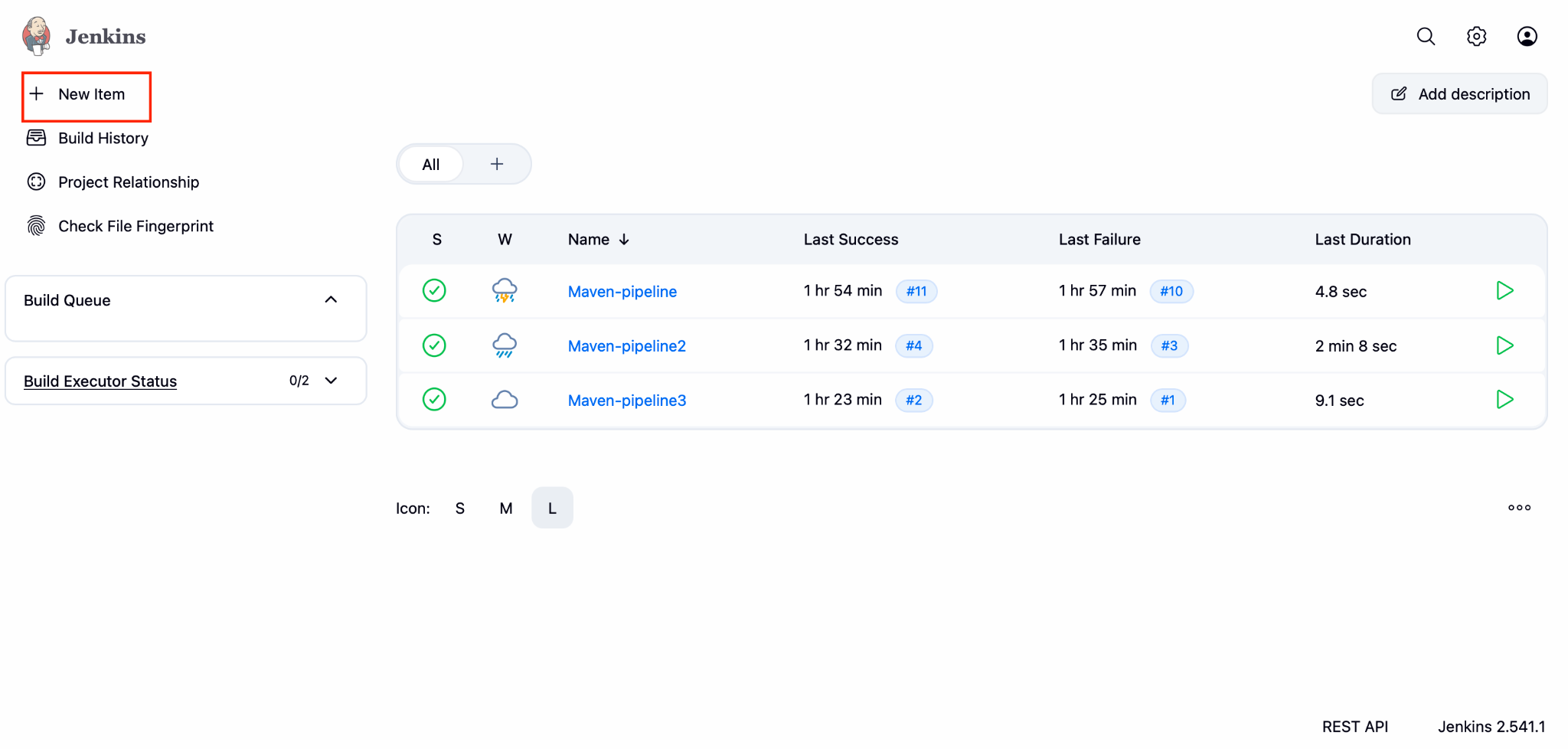
1. **Git:** Add Git, name it "Git", and check "Install automatically" or provide the path if installed.

.

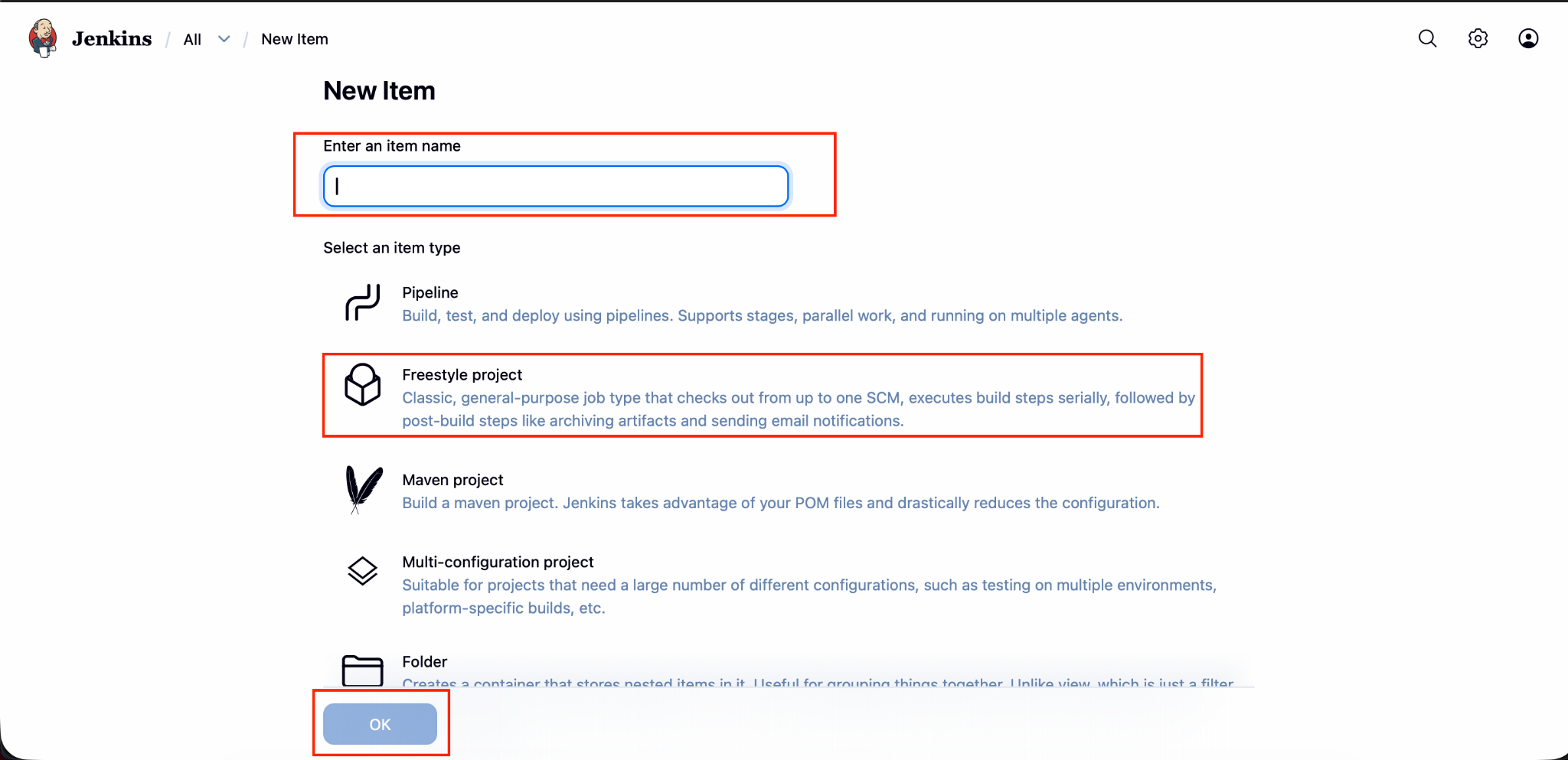
## **3. Creating the Jenkins Pipeline (Step-by-Step)**

### **Step 1: General & Source Code Management (SCM)**

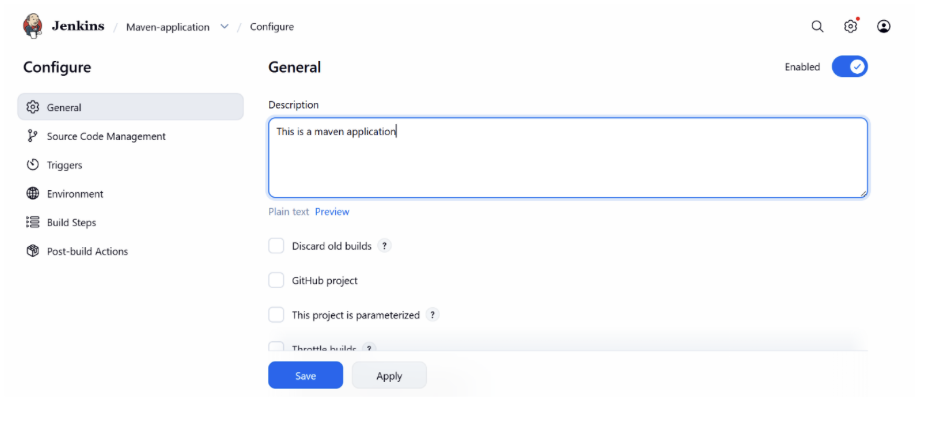
1. **New Item** > Enter EcommerceApp-Pipeline > Select **Maven project**.



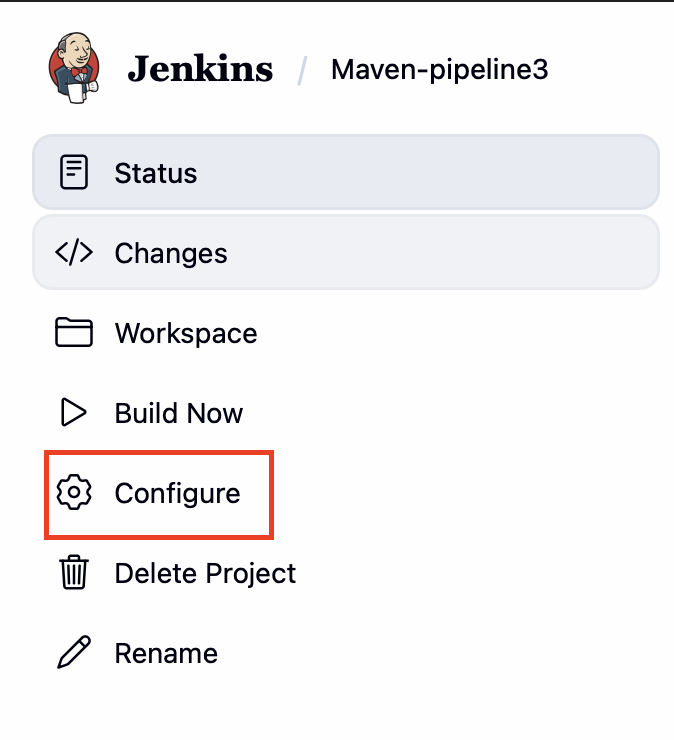
1. When you click the new item you will be directed to the next page as shown below.  
   Now give the name, select Freestyle project and click ok.



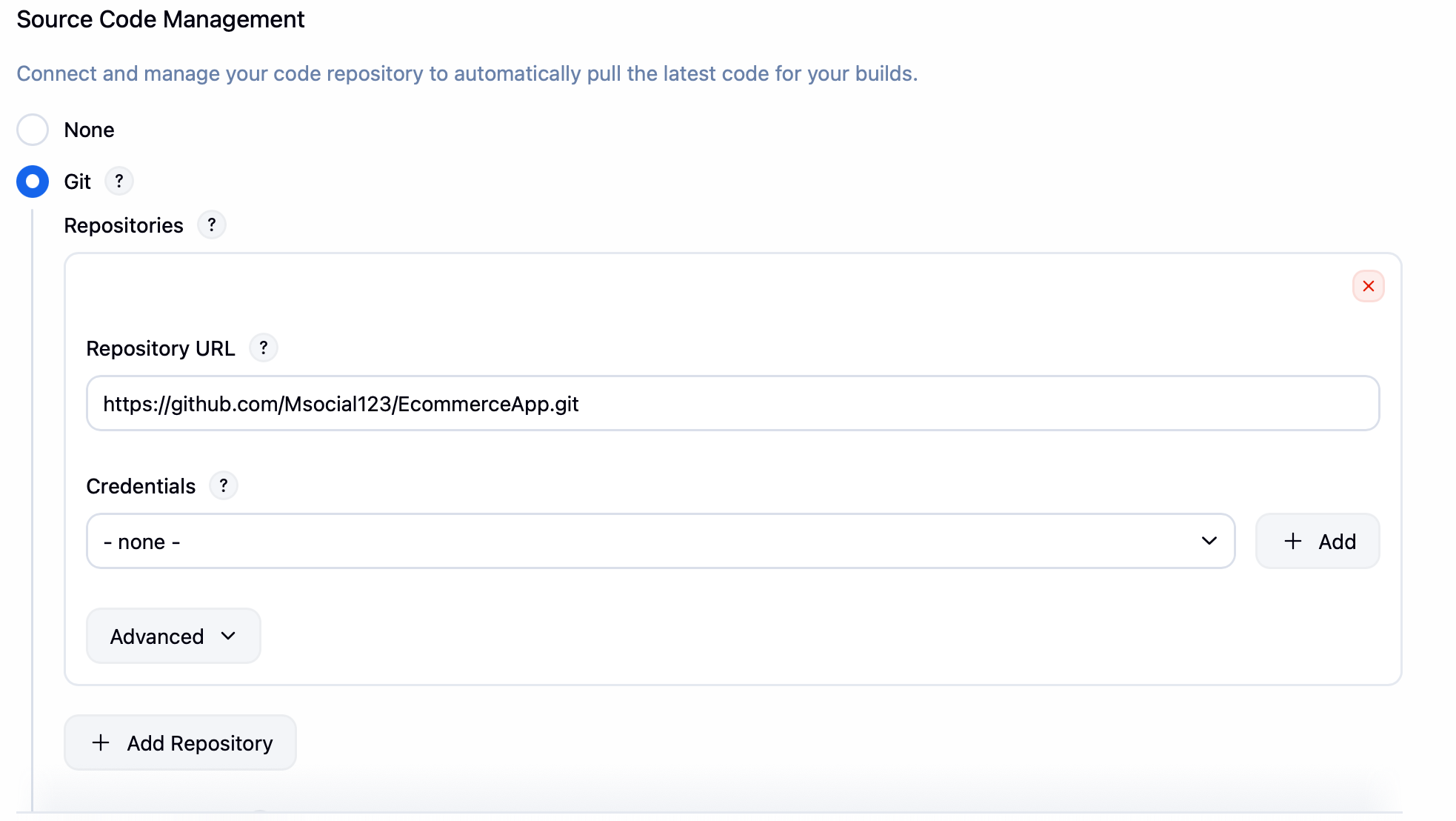
1. In general add a description.



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

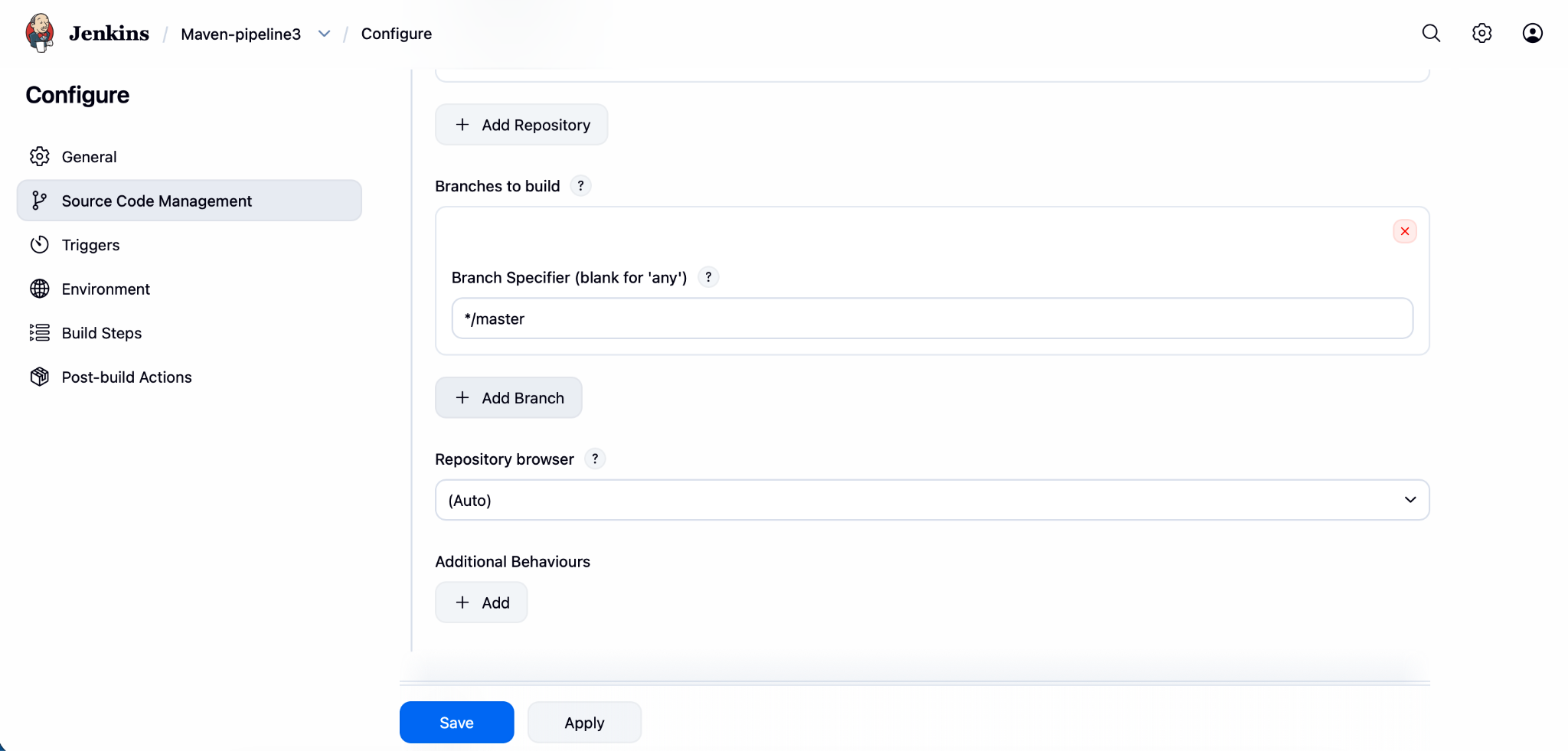


1. Now if we scroll little bit down we will find **Source Code Management** section



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

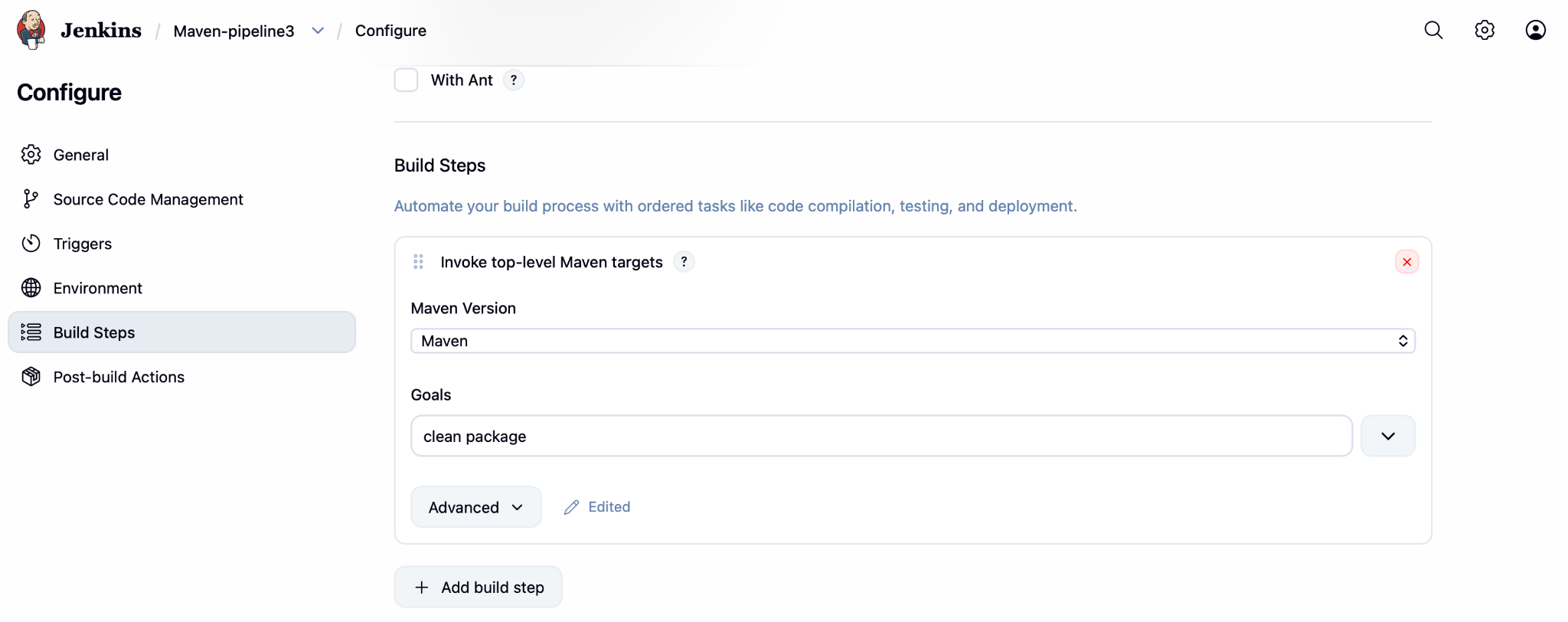
6. Now if we scroll a little down we will find **Branches to build** section. Here we need to provide in which branch the project is present(main/master).



7. Leave Triggers and Environment sections

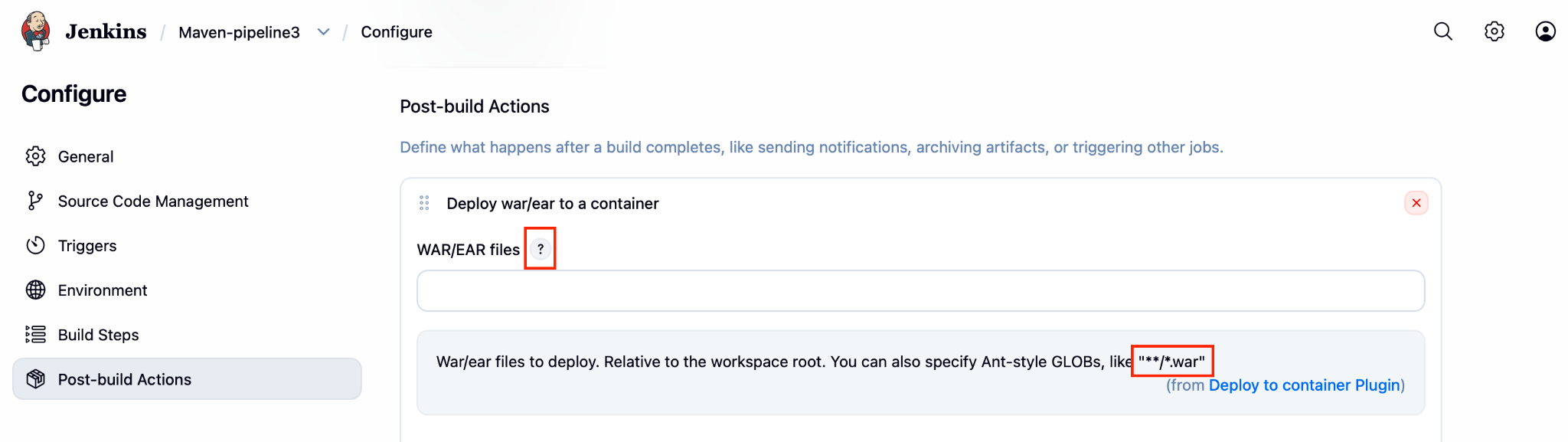
8. **Build Settings**

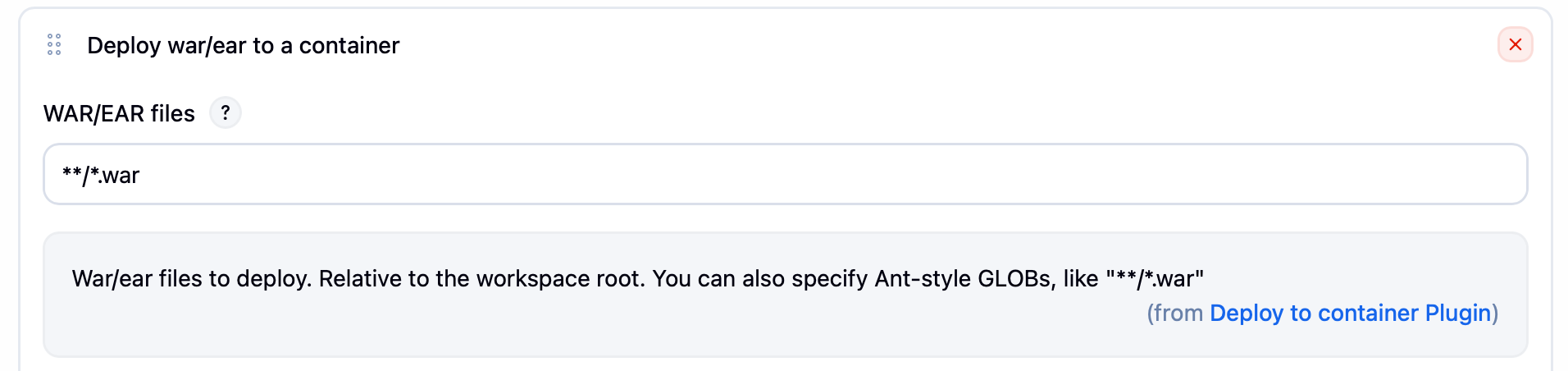
1. **Root POM:** EcommerceApp/pom.xml (Based on your logs, the POM is inside a subfolder).
2. **Goals and options:** clean package



9. Now go to **Post\_build Actions**

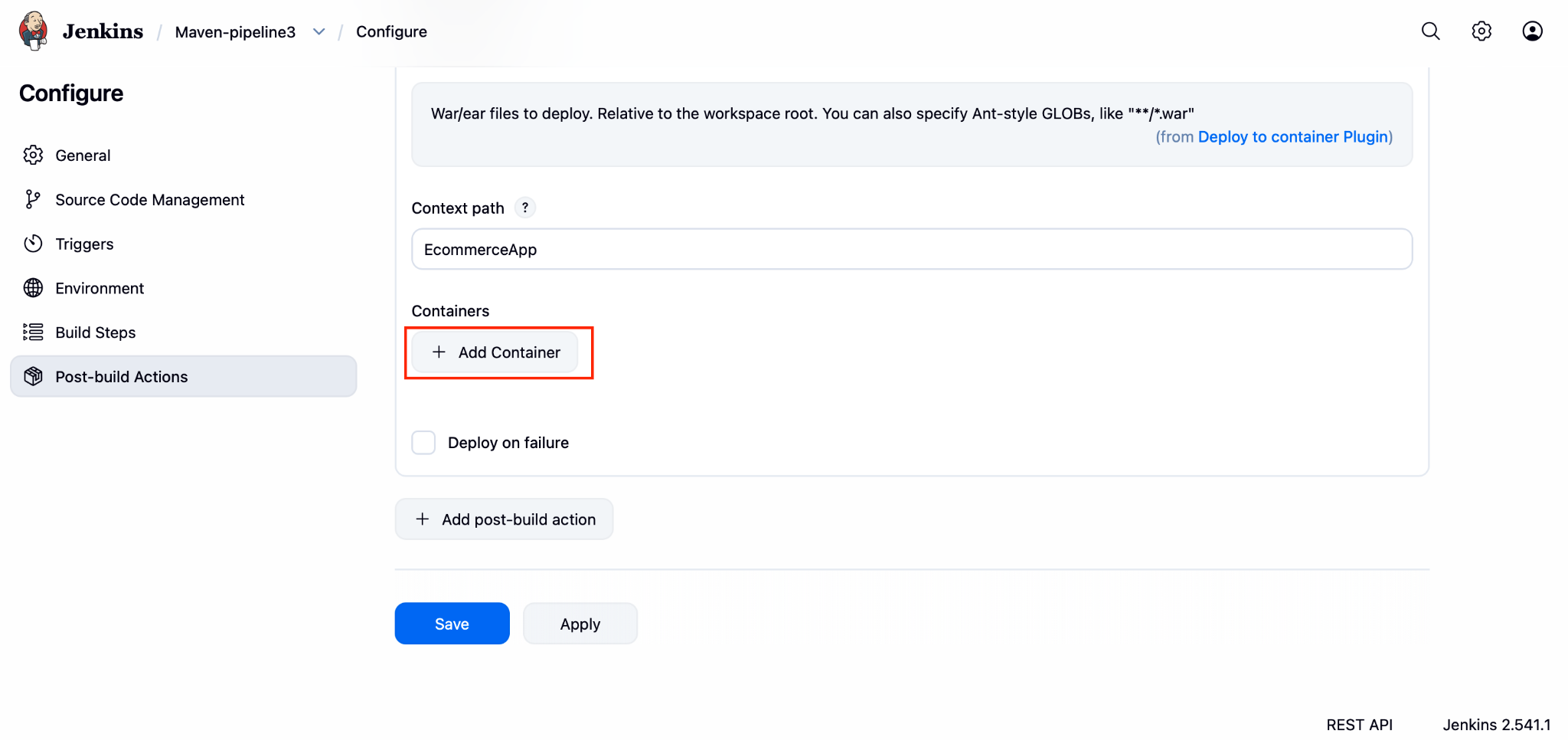
Click the question mark we will get text bar below copy **\*\*/\*war** without (“ ”)

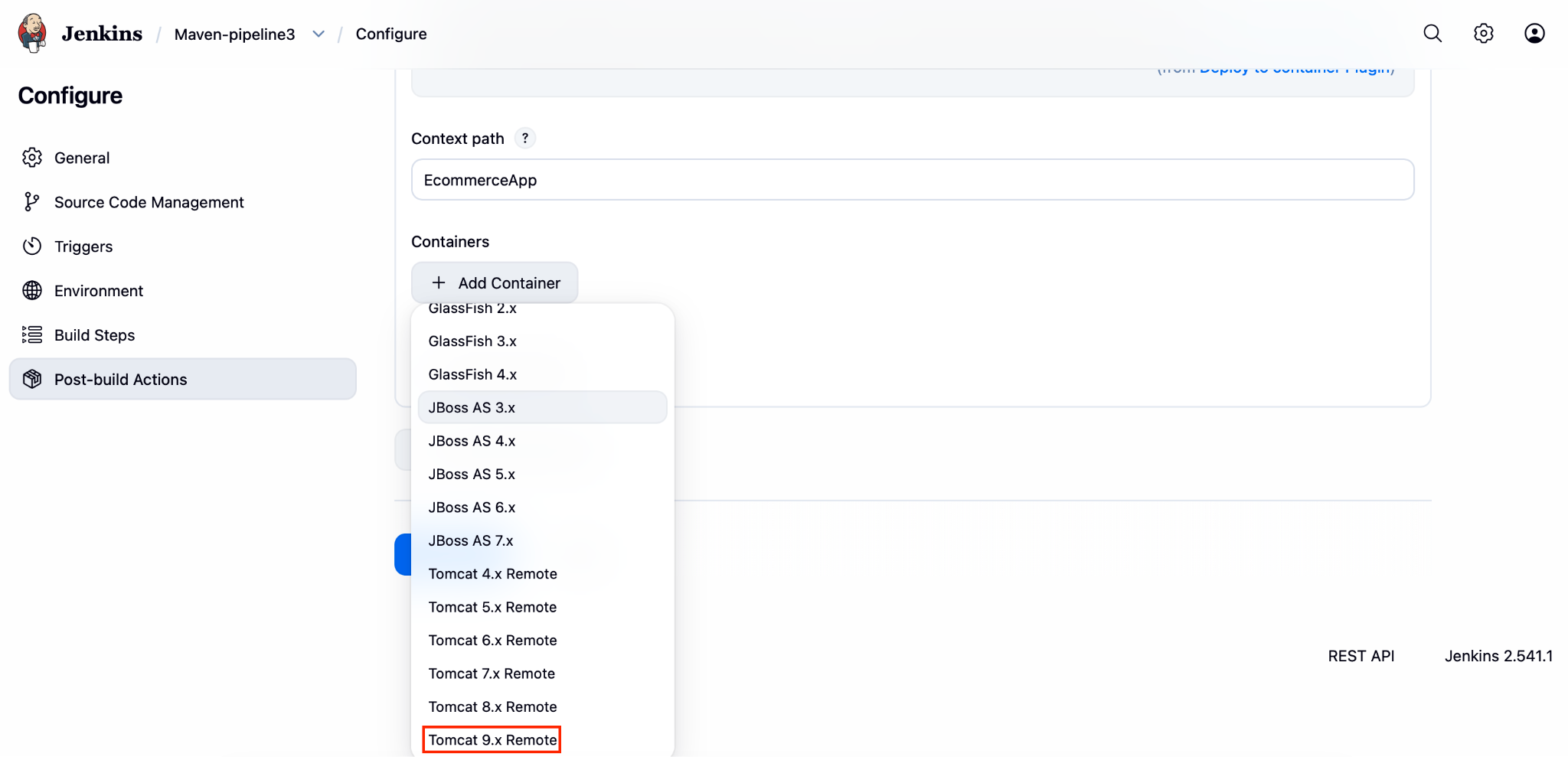
****

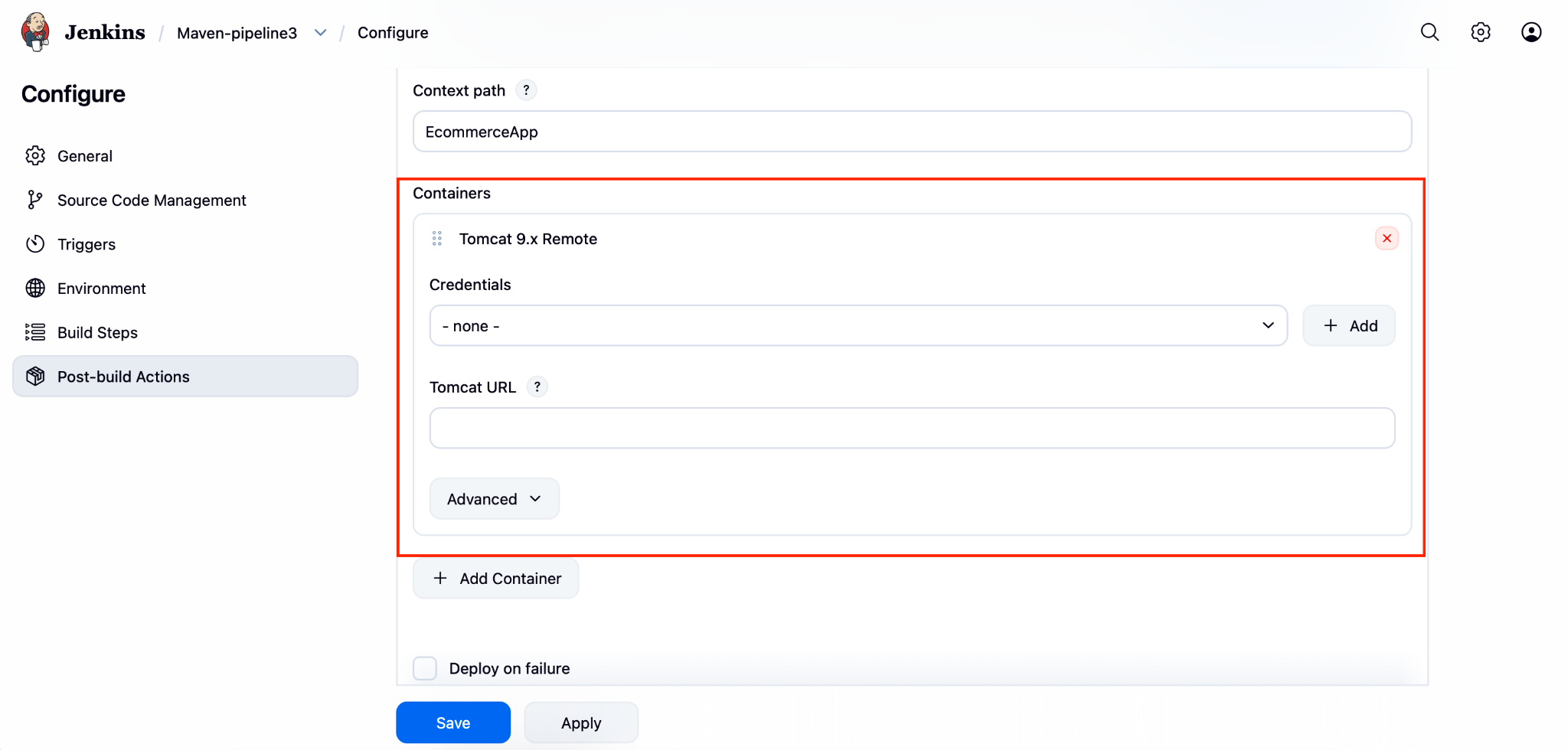
****

10. Give **Context Path**

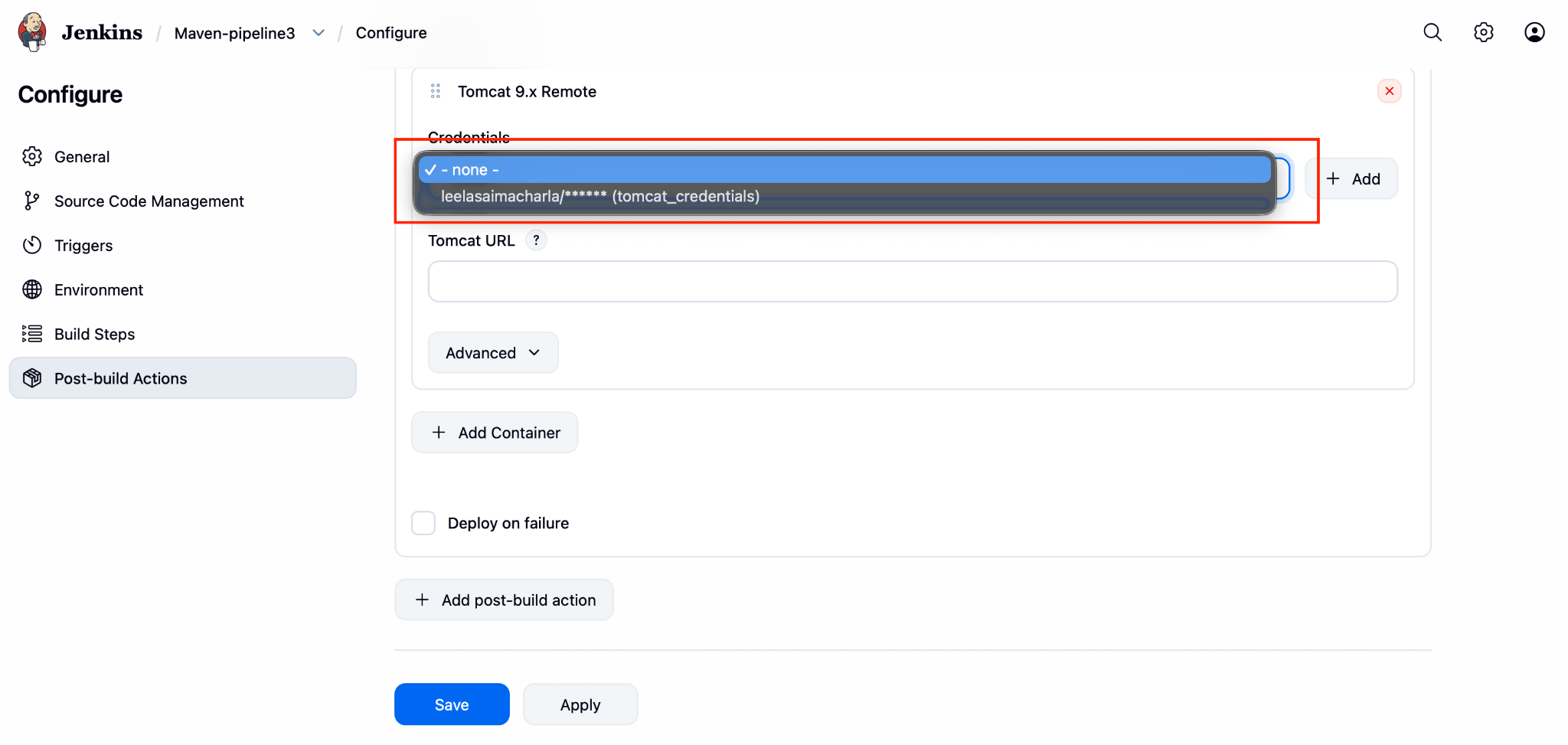
11. Now for **Containers** click add container and choose **Tomcat9x Remote**

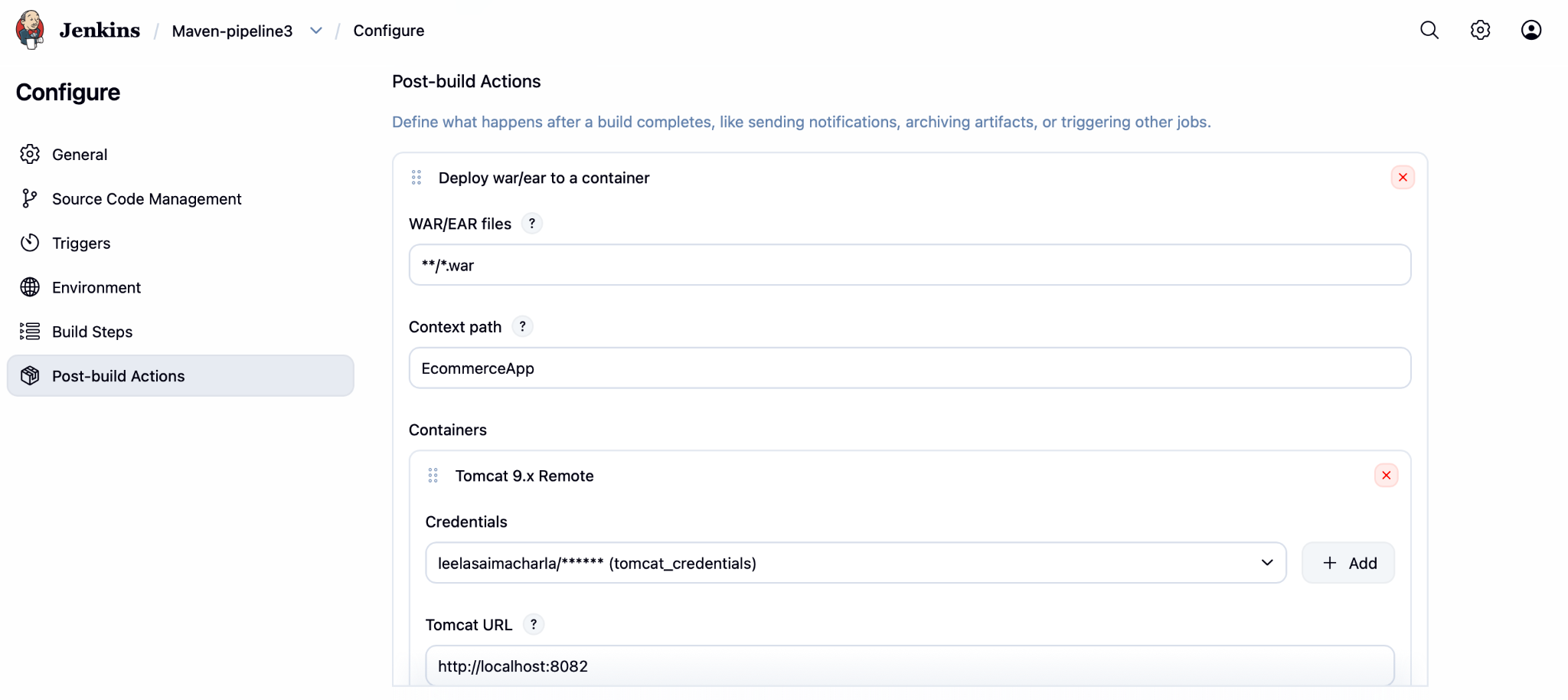


****

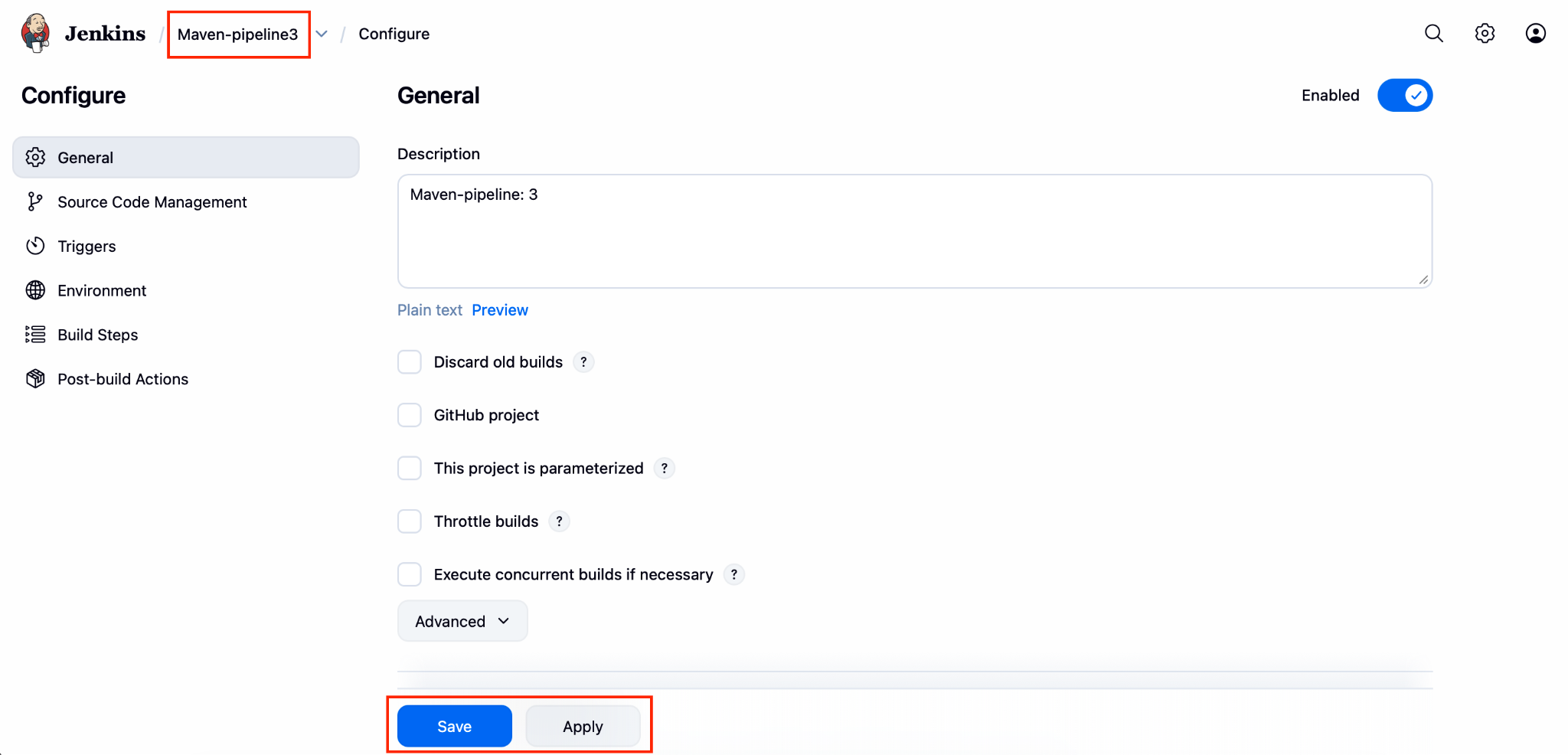
****

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

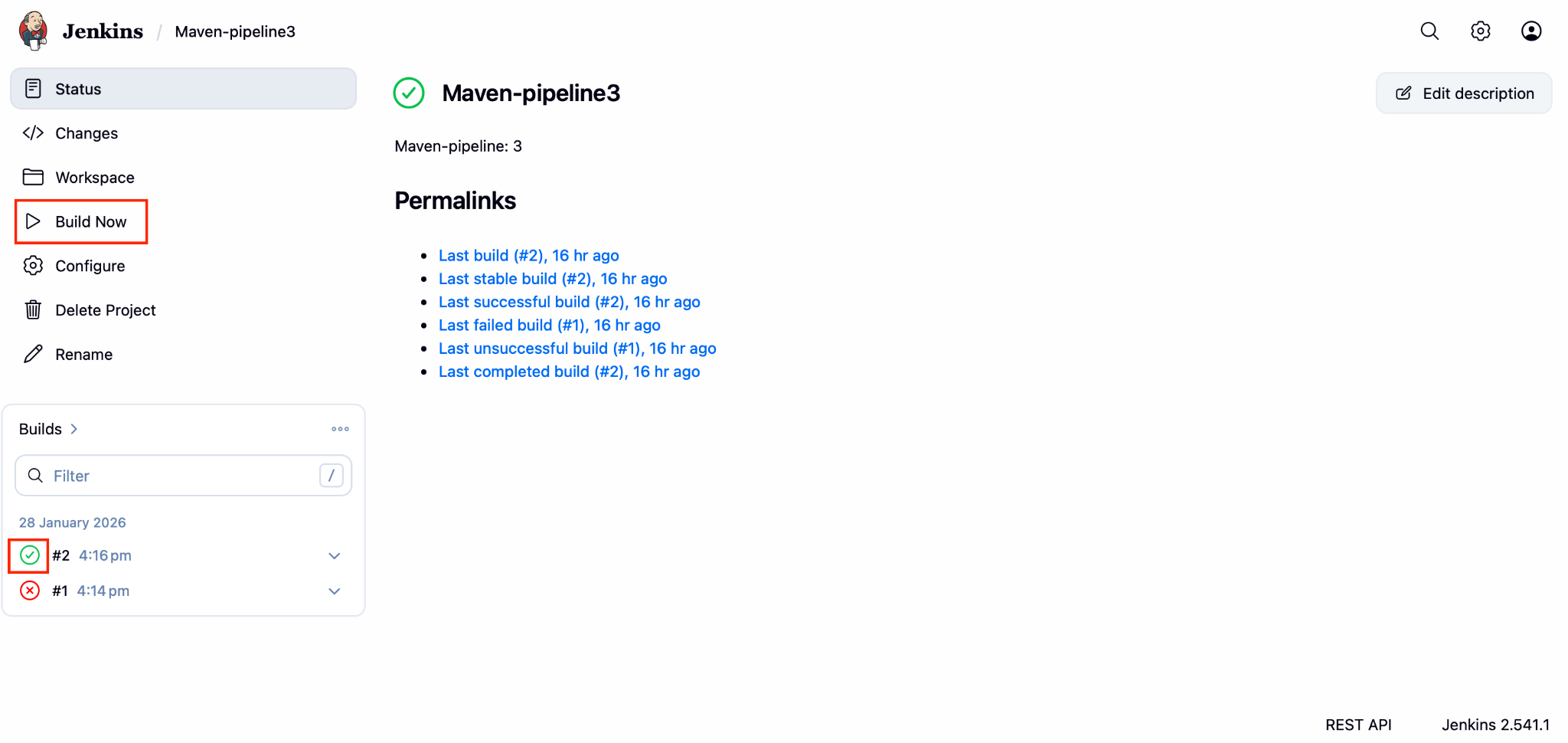


****

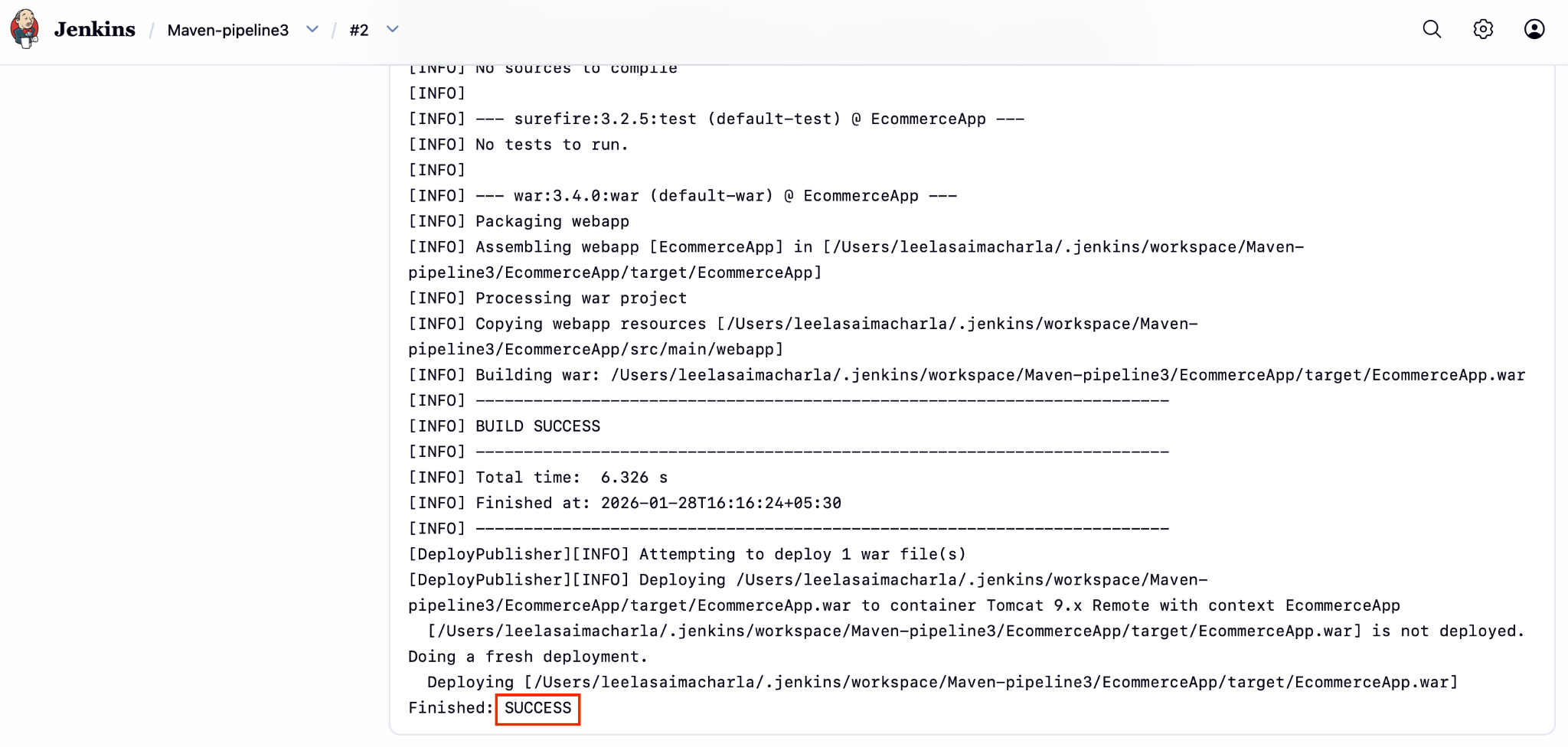
13. Now click **apply and save.** After successfully saving click on pipeline name (“**Maven-pipeline3**”)



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

****

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.



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

