Continuous Integration and Continuous Deployment Lab

Experiment – 6

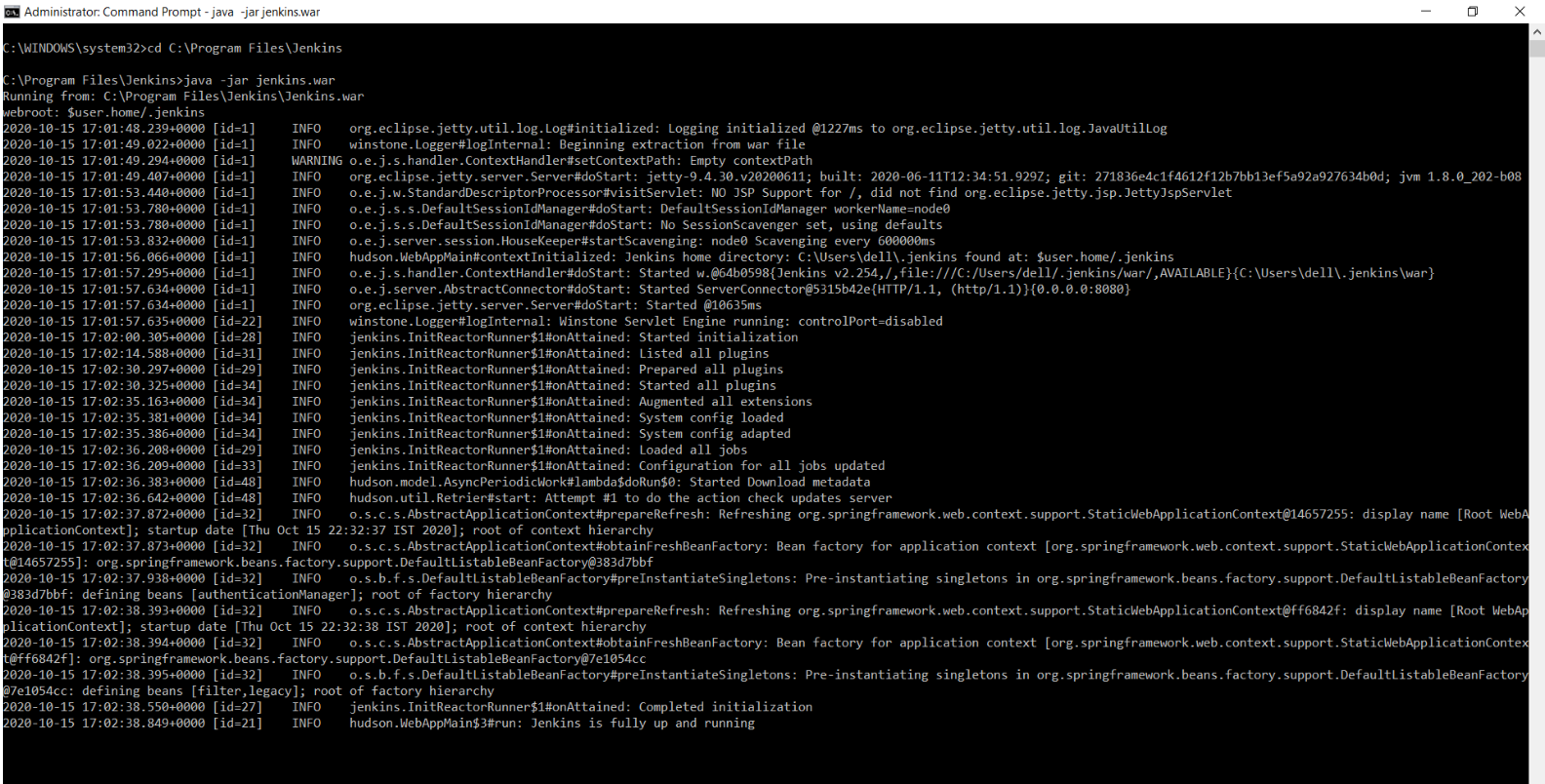
**Name – ABHISHEK SHARMA**

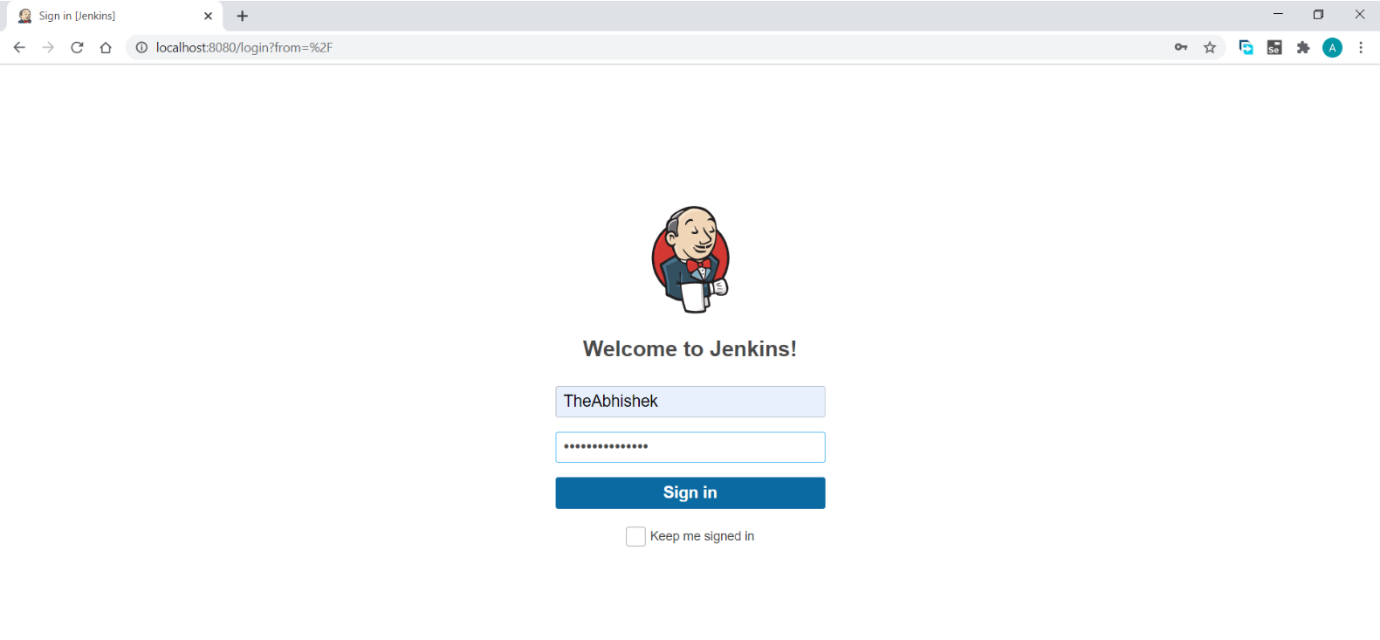
**Roll No – R171218005**

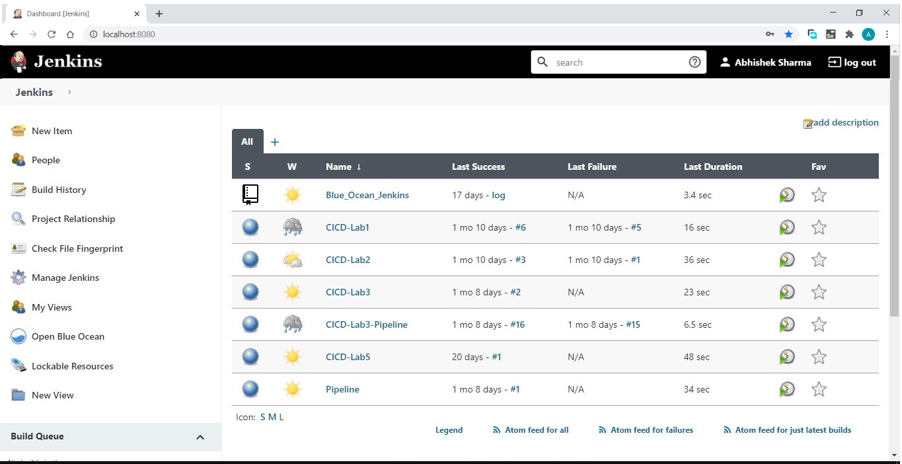
**Create Pipeline using Jenkinsfile**

Start the Jenkins service using the java -jar jenkins.war command in the command prompt

Go to the Jenkins dashboard at localhost:8080





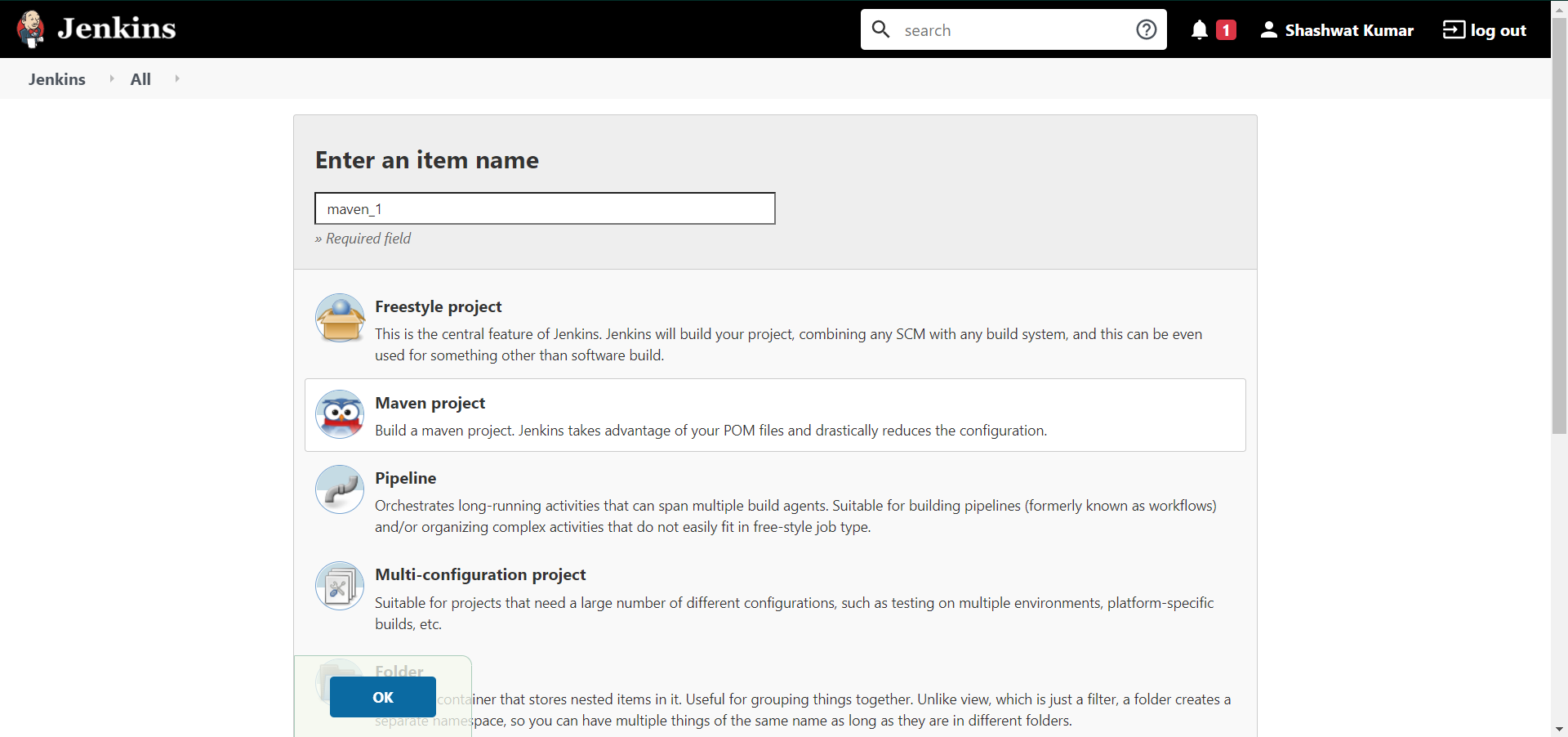


Click on the new item option

Select the maven project option

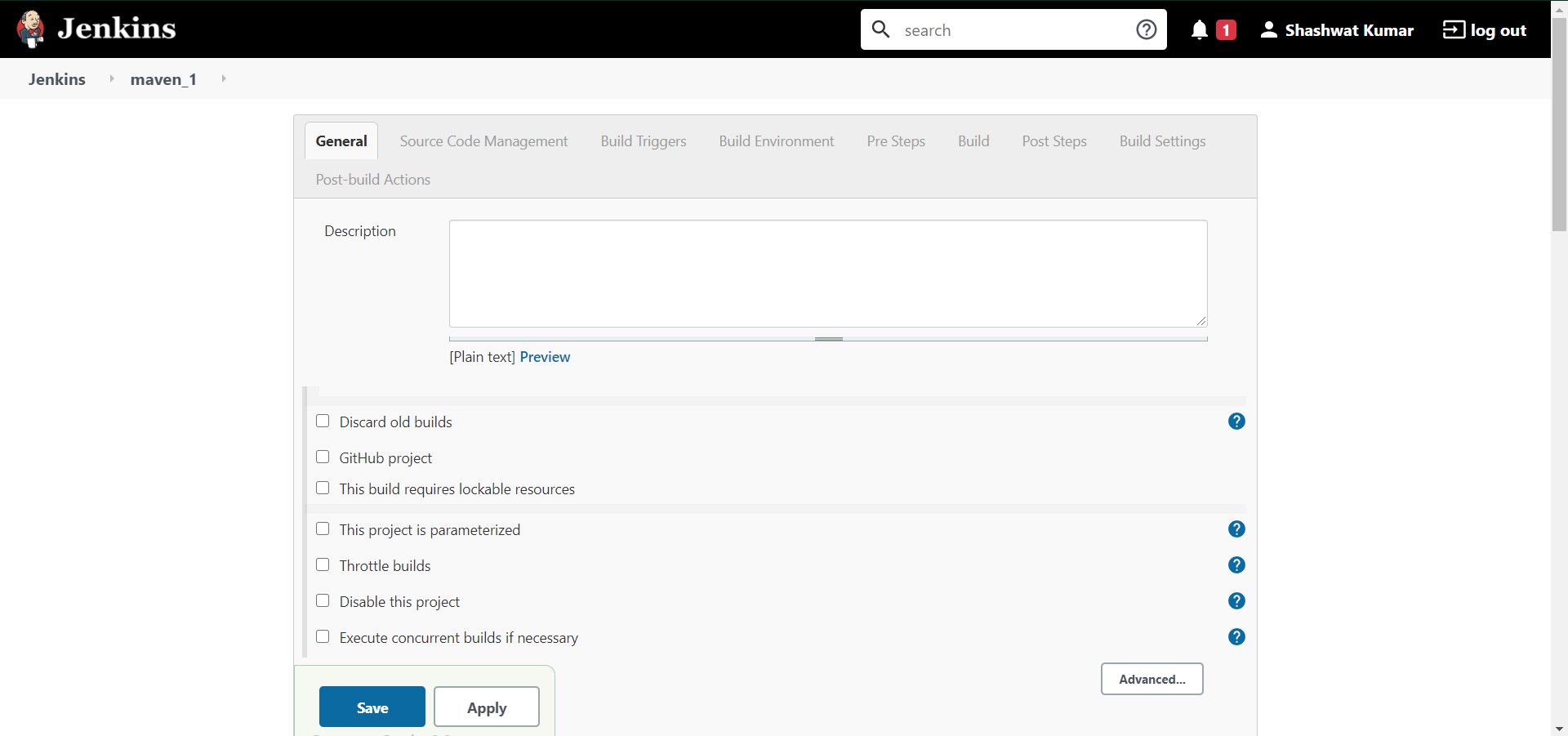
Give a name to the project

Click ok



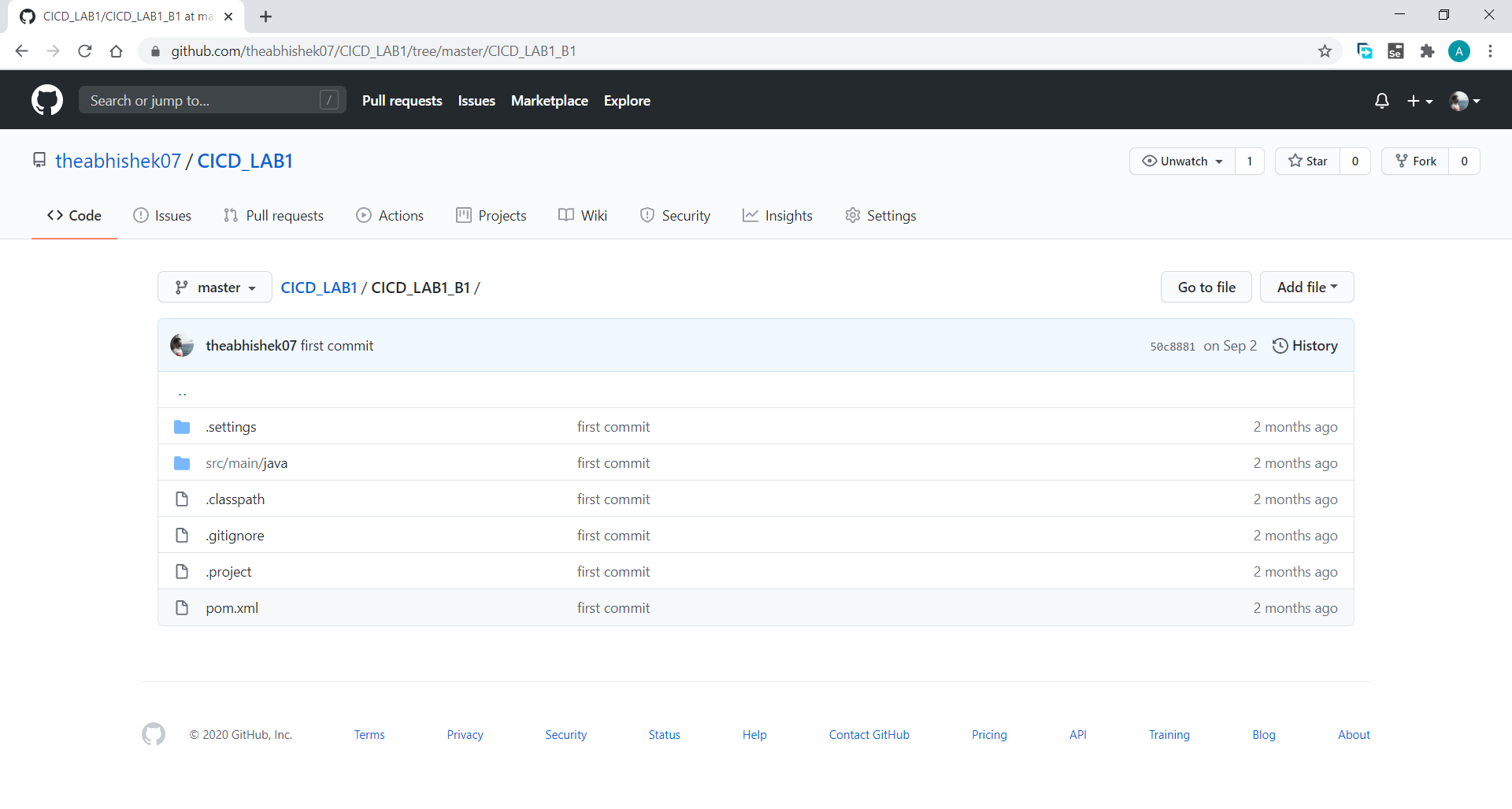
You will be directed to a page similar to this.

Click on the source code management tab to add the github repository



Visit the repository where you saved your maven project

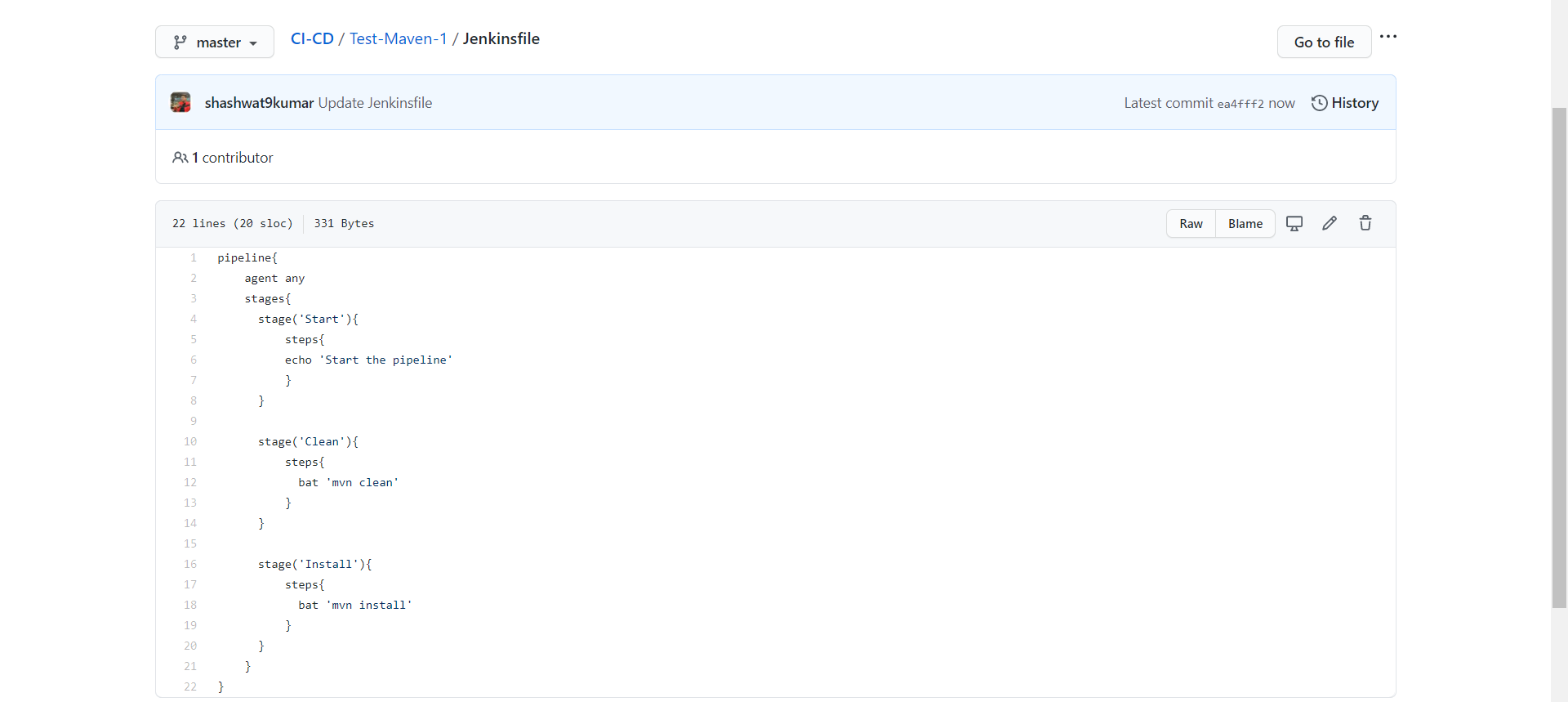
Copy the location of the github repository



The github repository has to contain a file named “Jenkinsfile” that tells the Jenkins server what tasks to do.

In this example, it asks the Jenkins server to run 3 commands in 3 stages namely

echo “Start the Pipeline” ; mvn clean ; mvn install



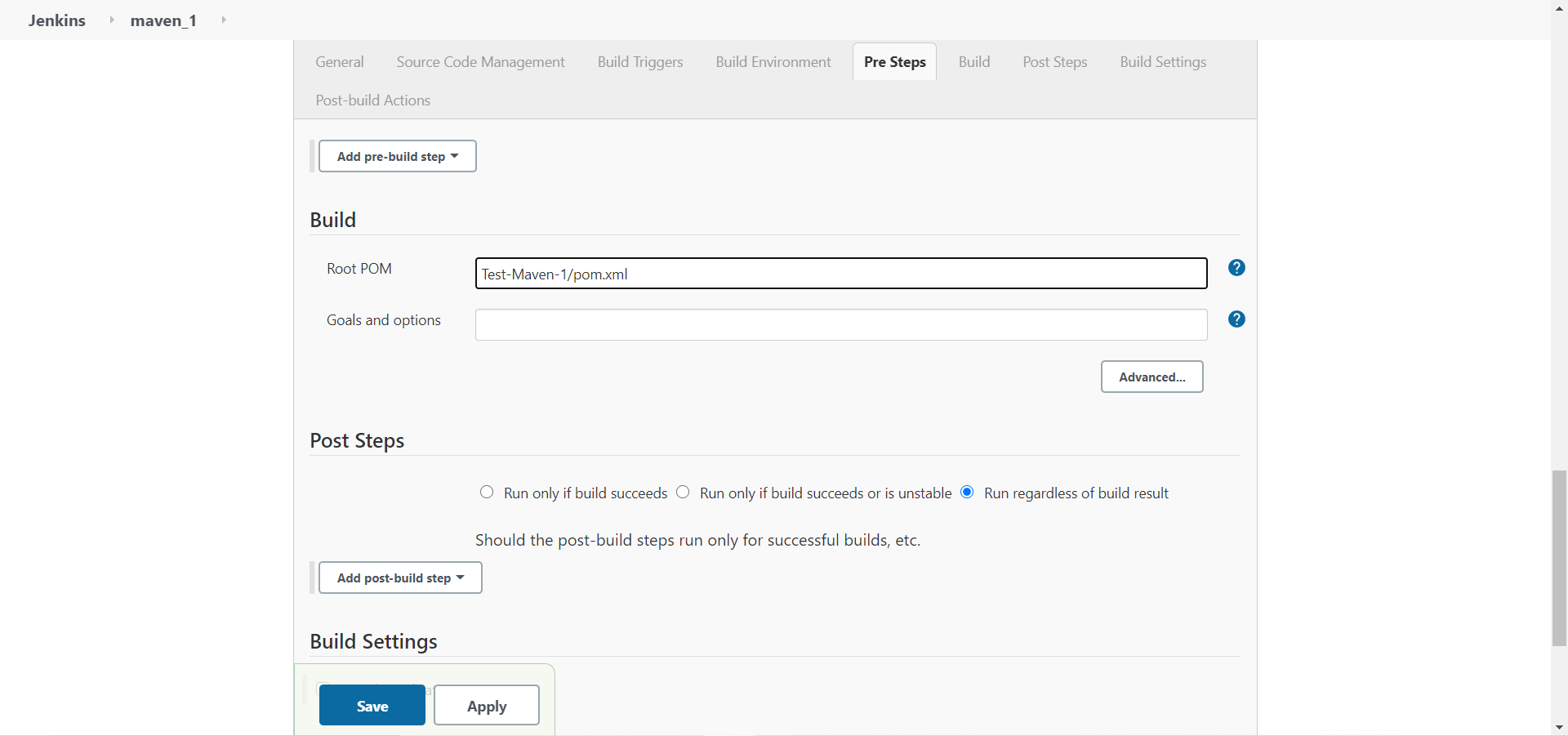
Go down to the Build Section

Add the exact location to the pom.xml file.

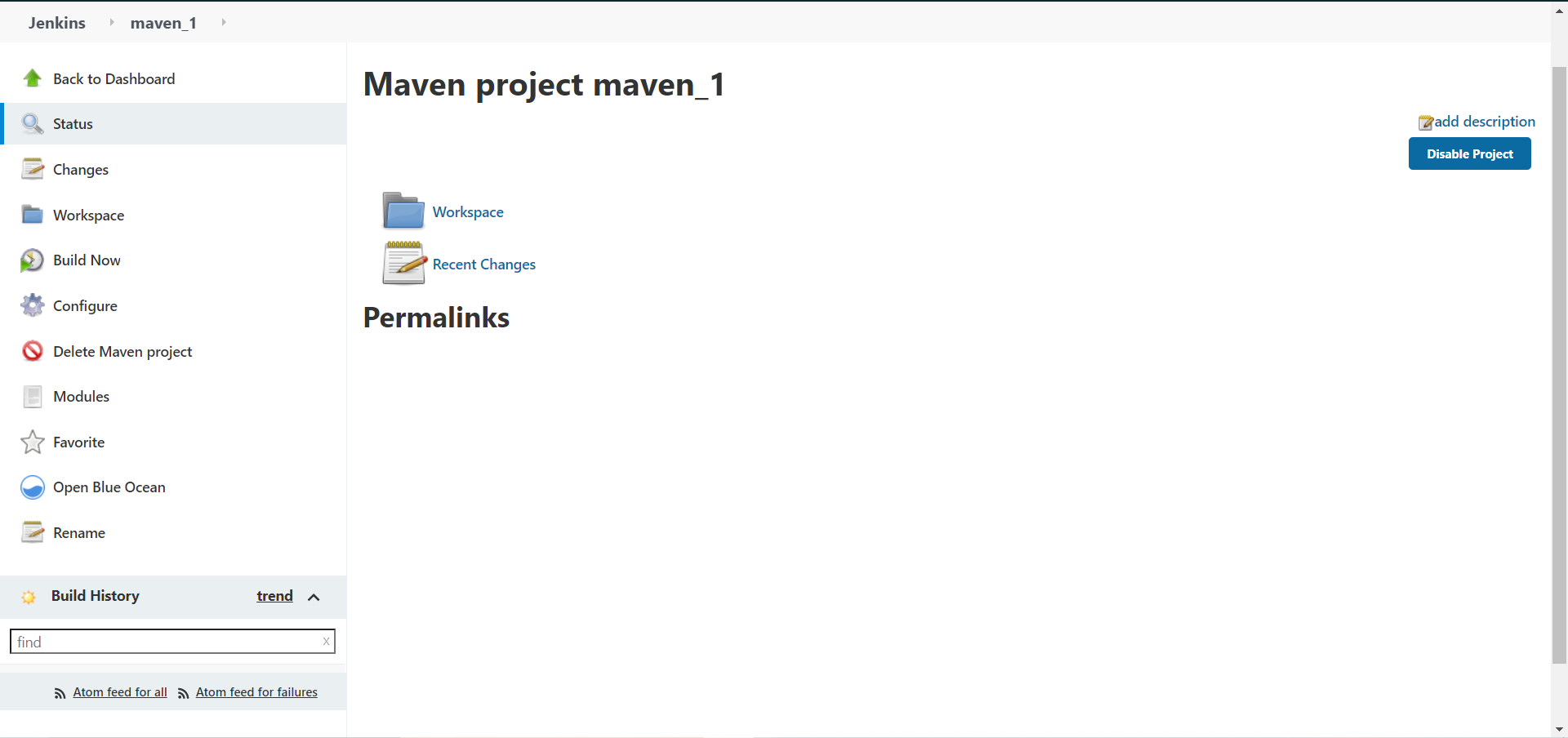
If it is inside some folder, specify it as {folder\_name}/pom.xml

In the goals specify “clean, test and install” as the maven goals

Click save and apply

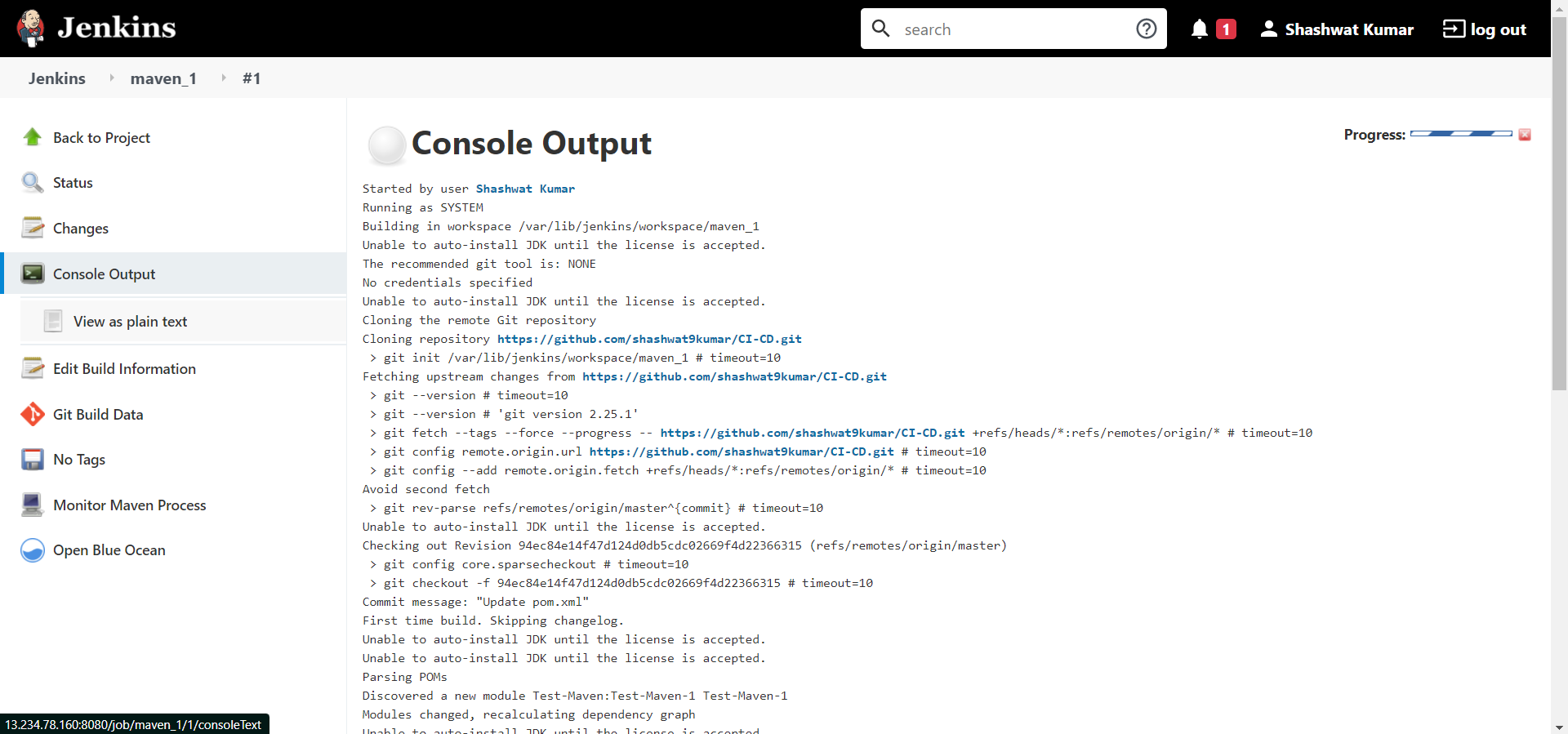


You would be redirected to a dashboard like this



Click on Build Now option to build the project.

Select the console output option to check how the build is running



If the build runs successfully it would return “BUILD SUCCESS”. It would return failure in case of any errors in the project

