**Sonarqube installation**

1.install sonarqube using docker command

2. 1st install docker in ubuntu machine.

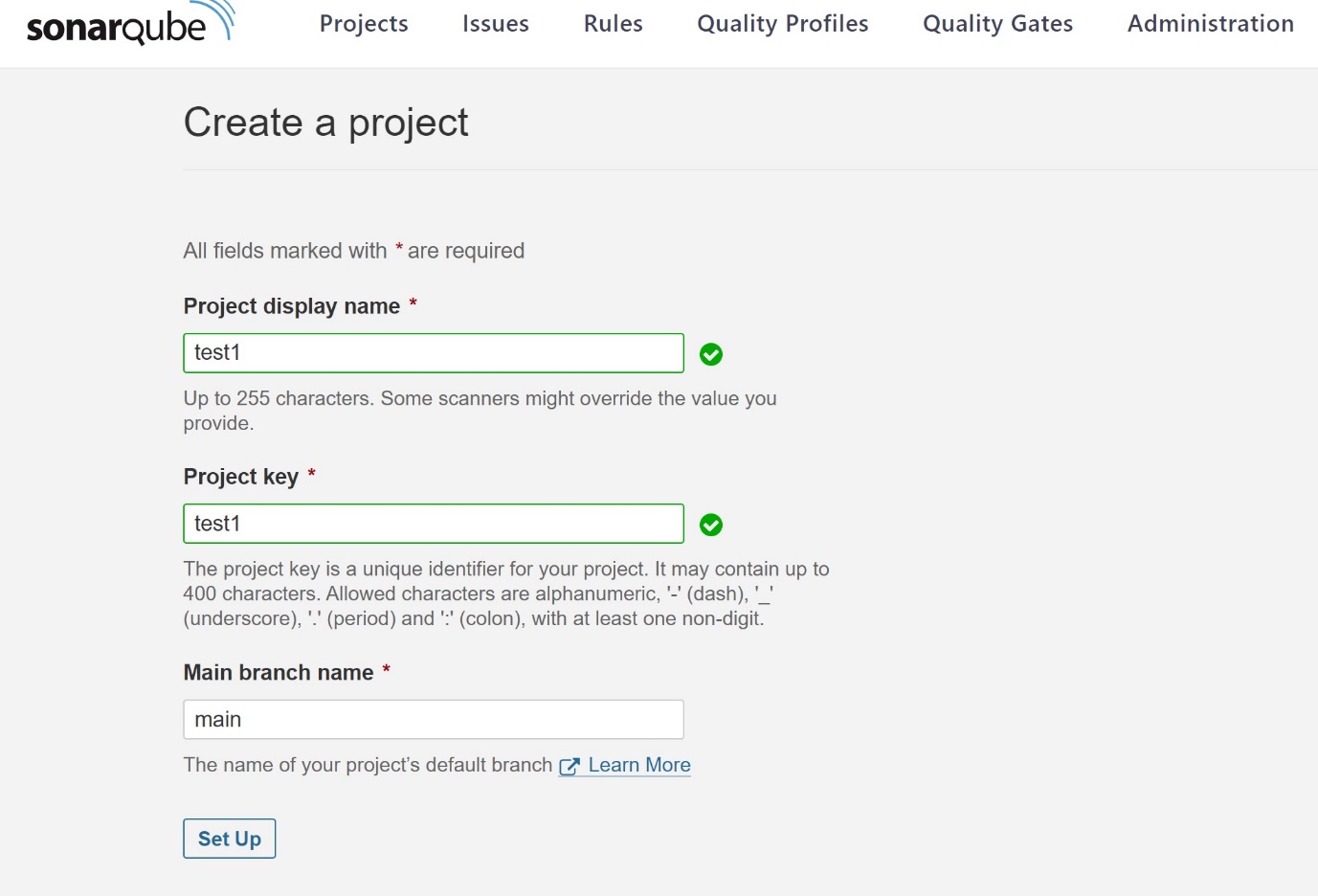
3.docker command is "docker run -itd -name sonar -p 9000:9000 sonarqube:latest" > this is command download image and run container with name "sonar" and exposed

4.with port 9000.

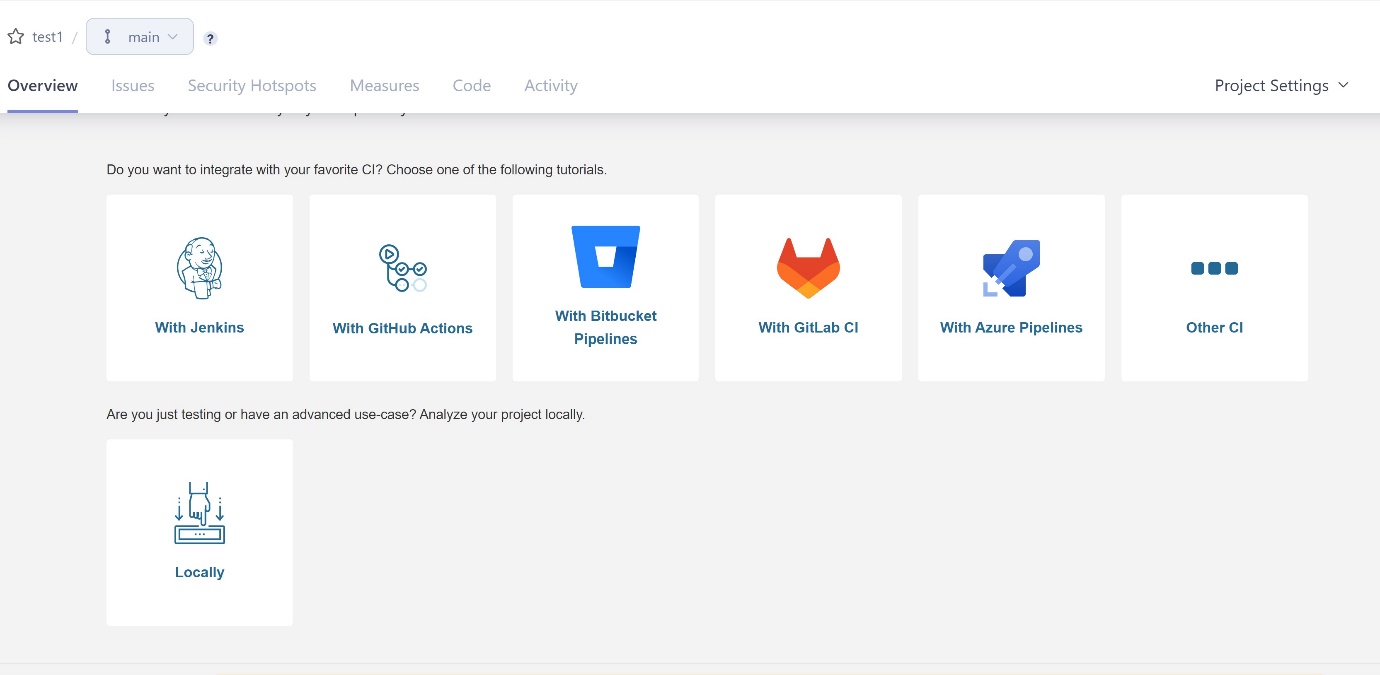
5.After deploy the sonarqube open with <Ipadress>:<port>.

6. the default username is "admin" and password id "admin" or "password" then set with new password.

Click on project select “manualy” then below page is shown.



## After setup “**How do you want to analyze your repository?**”



Select “withjenkins”

Then it will ask

## **Analyze your project with Jenkins**

## **Select your DevOps platform**

* Bitbucket Cloud
* Bitbucket Server
* GitHub
* GitLab

Select “github”

After select it will ask for pre requisites

Go with “continuous analysis” then “continue” again click “continue” in create Jenkins file select with build “maven”

Then “finish tutorial”

After creating the project file integrate with Jenkins

1. 1st install the plugins

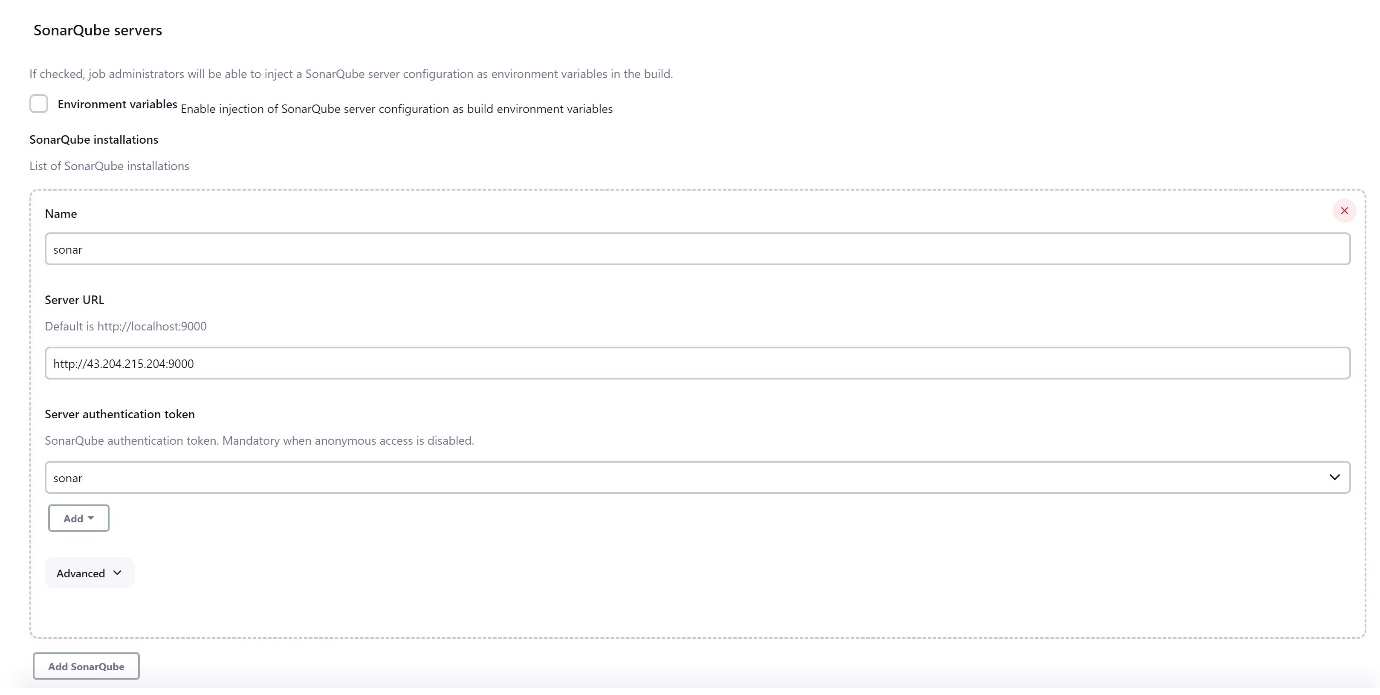
a.Sonarqube scanner

b. Sonar quality gate

c. Sonar generic coverage

those three plugin should be install , after install restart the Jenkins server

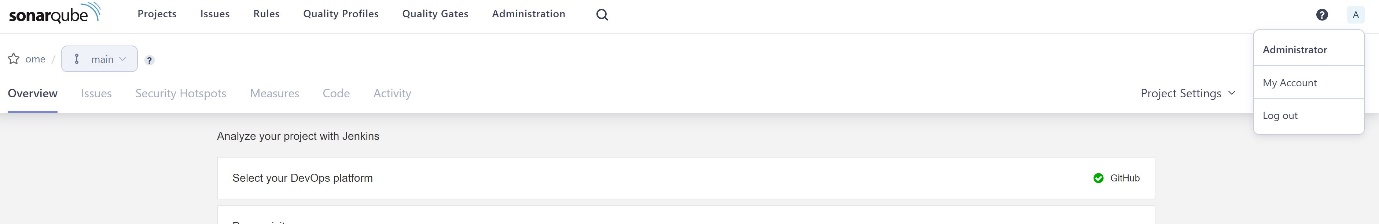
then go to MANAGE JENKINS >>CONFIGURE SYSTEM >>



Server name u give whatever its not any defaults like “sonar”

Server url is your might know I am not

In “server authentification token” add “jenkins”

Before that go to sonarqube server at right side section 

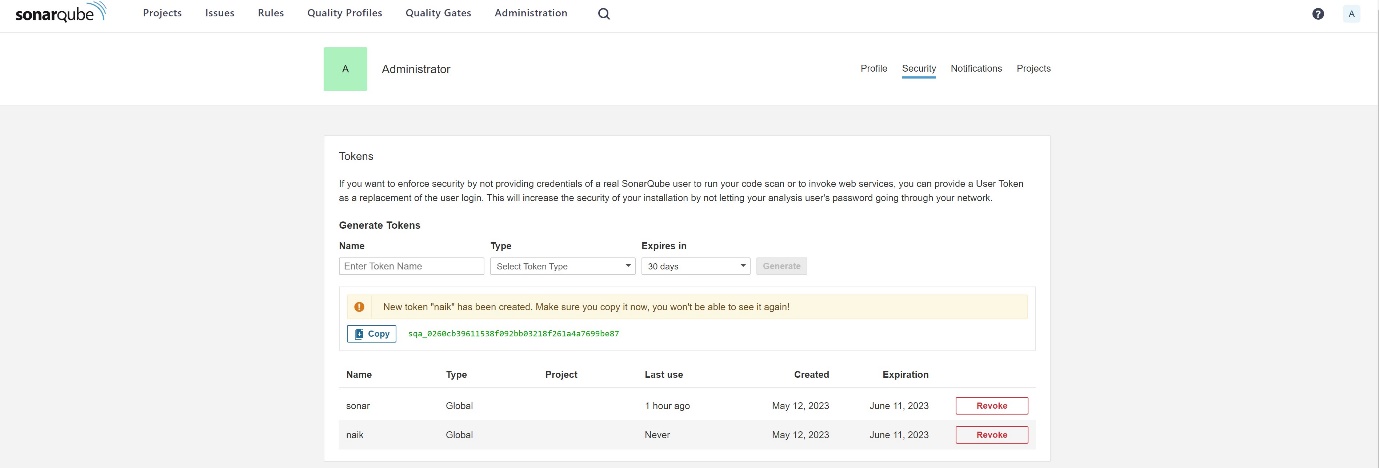
Select “my account” then.

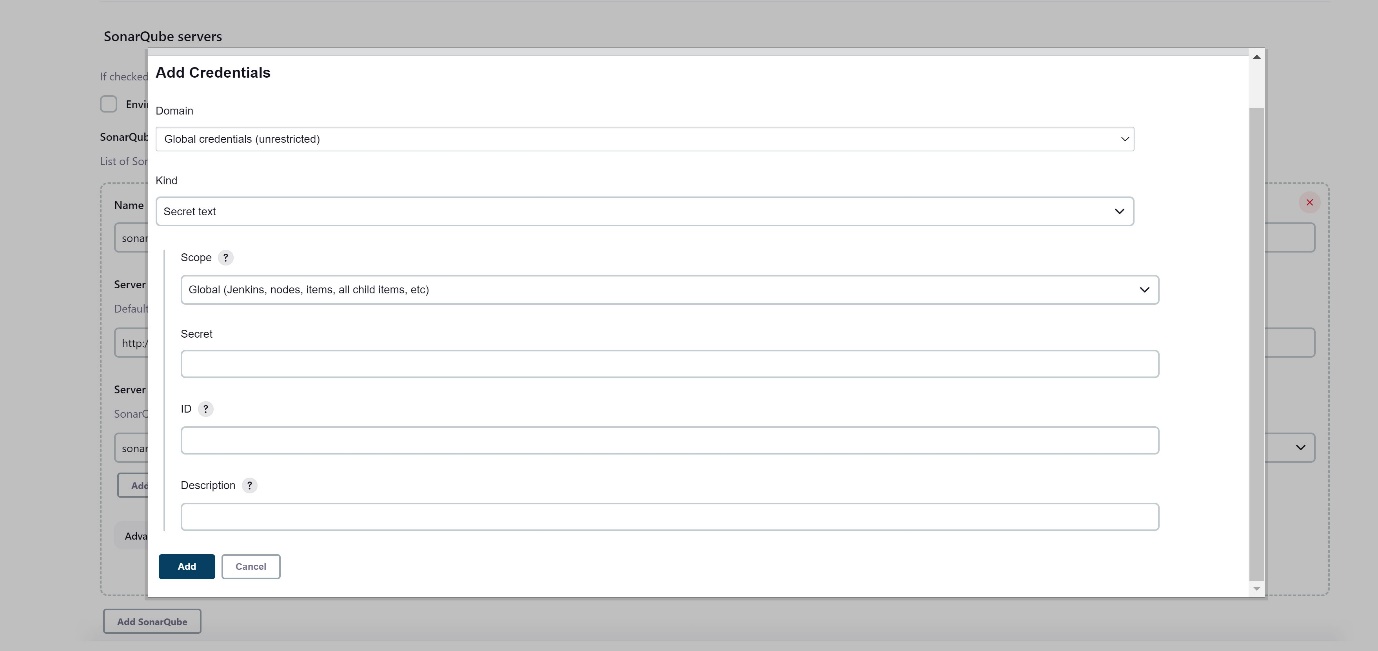
Click on security

Give any name for “token”

Select “global analysis token” then click on generte

After generate copy the token code..use it in credentials.





Click on add credential click on “kind”, select ”secrete text”

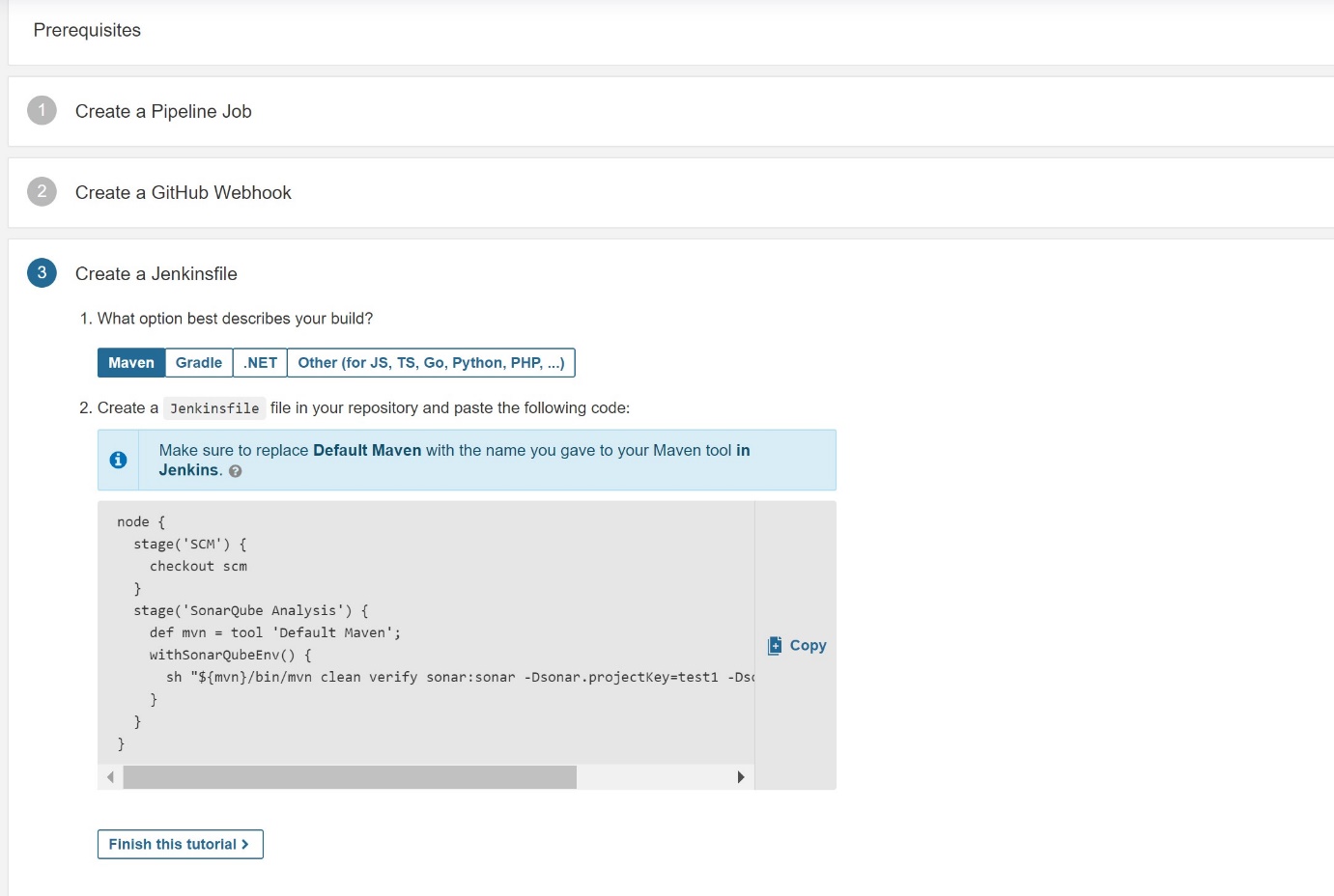
Paste the token id in secrete section click add

Then select the new added credential “appy” “save”.

Now integration part is over

Then pipline is

After create the project the project at last this code will appear copy and paste in Jenkins pipeline



stage("sonar scan"){

steps{

withSonarQubeEnv("sonar") {

sh "mvn clean verify sonar:sonar -Dsonar.projectKey=sonarscan -Dsonar.projectName='sonarscan'"

}

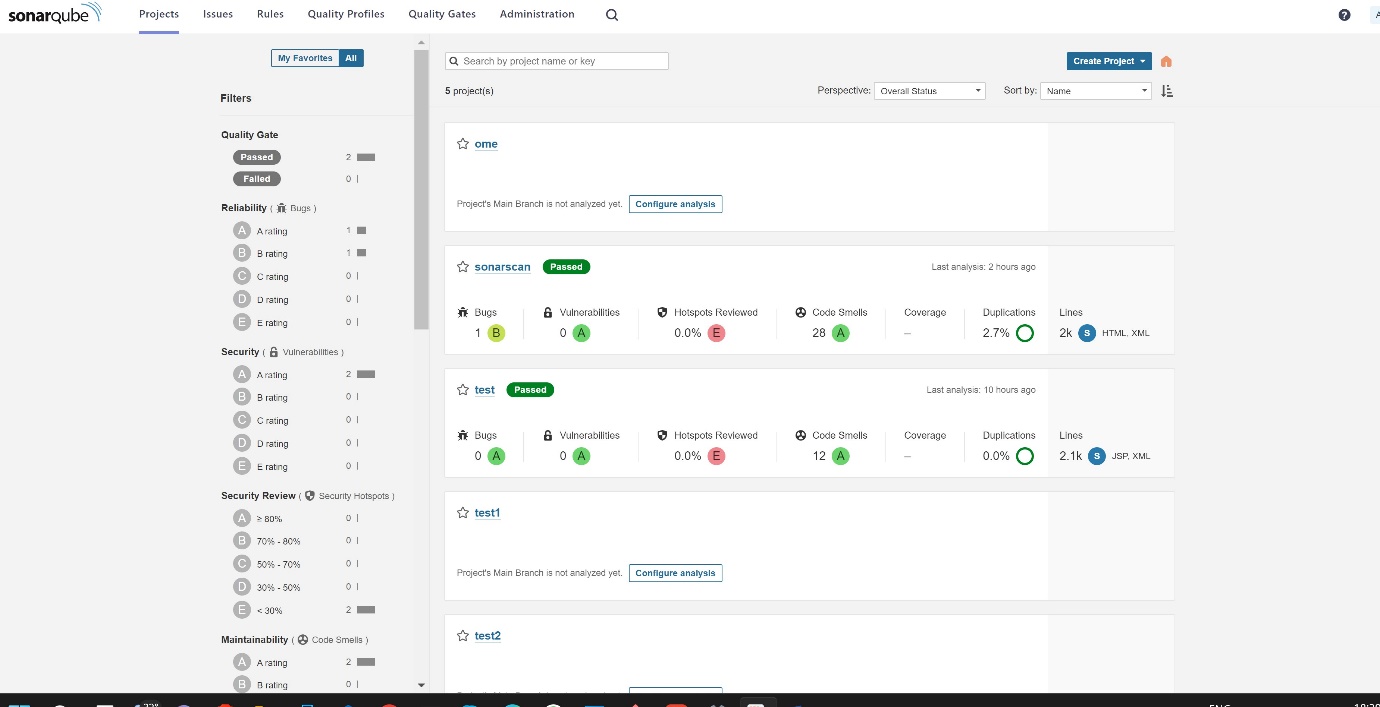
}

This is the stage part here squbeEnv is we gave the name during the sonarqube integration “sonarqube installation “ name

Remove the sh ”${mvn}/bin” from pipeline no need

That’s all for .

After build with your code sonarqube will scan and show the result



If code will correct it show passed. Further information go through sonarqube documentation. Thanks.