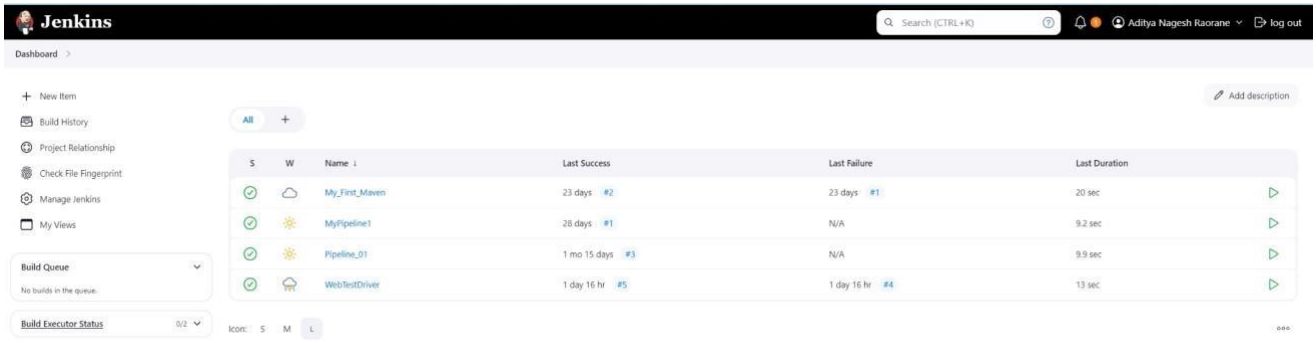
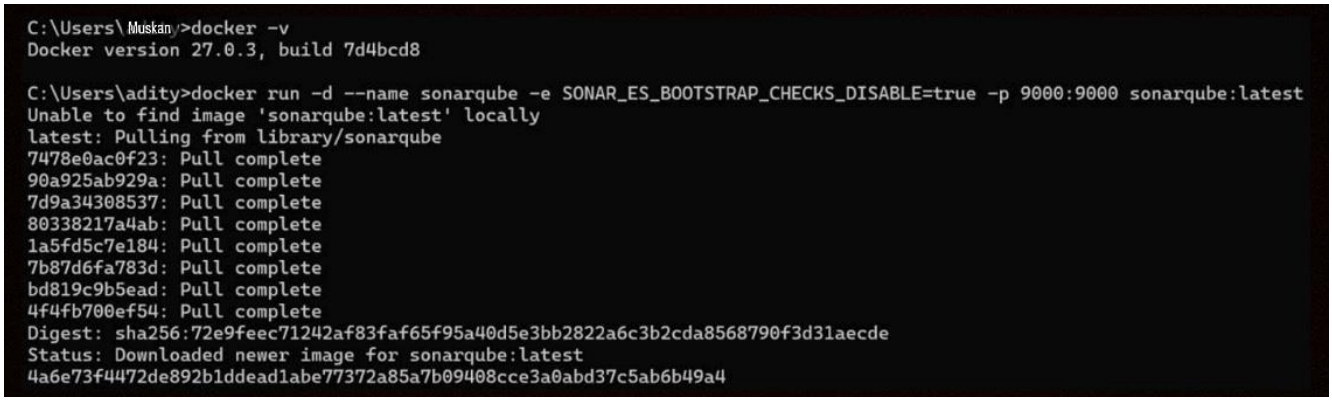
Aim: To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

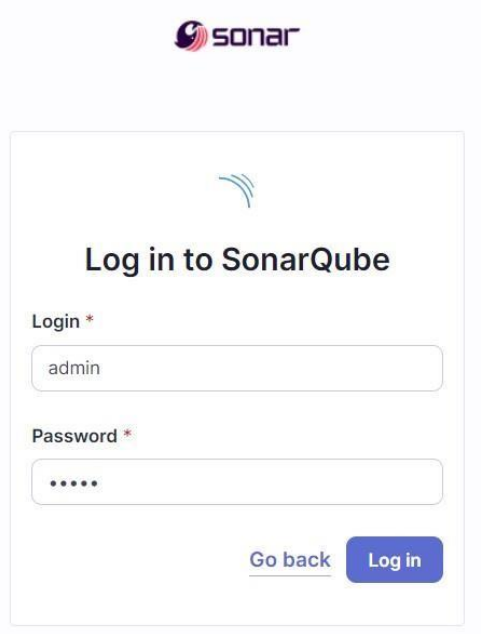
1. Open up Jenkins Dashboard on localhost, port 8080 or whichever port it is at for you.



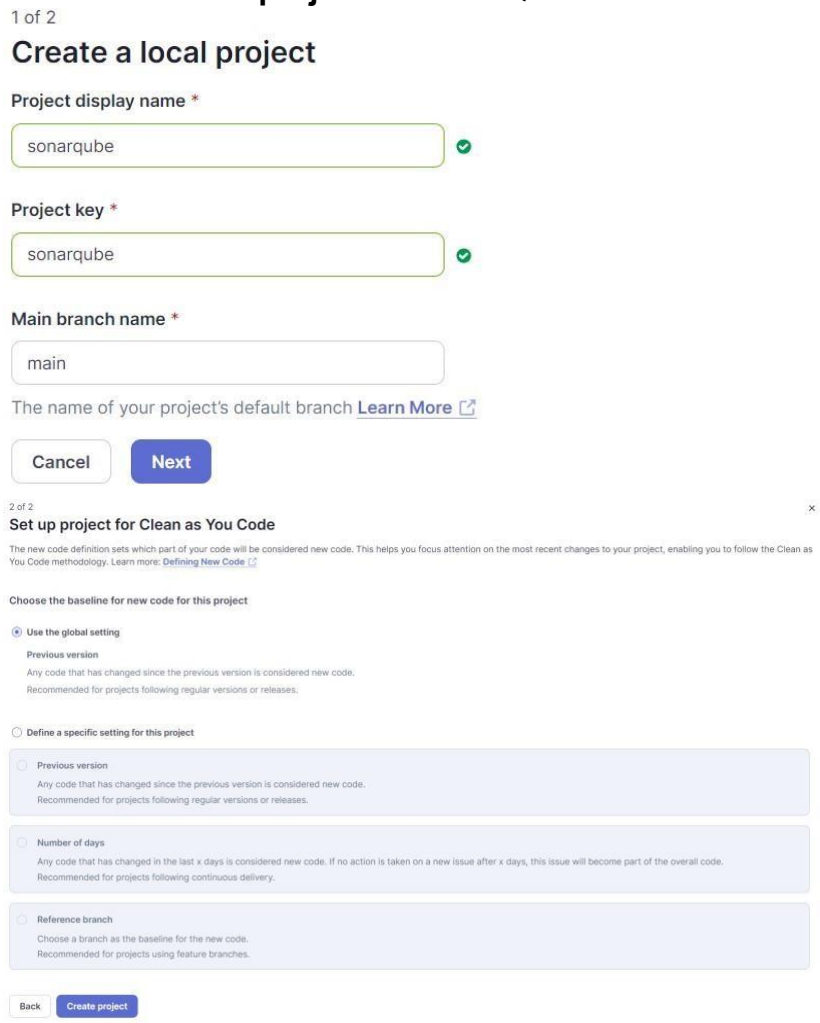
2. Run SonarQube in a Docker container using this command :a] docker -v b] docker pull sonarqube c] docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest



3. Once the container is up and running, you can check the status of SonarQube at **localhost port 9000**. The login id is "**admin**" and the password is **mus12**



4. **Create a local project in SonarQube** with the name **sonarqube**

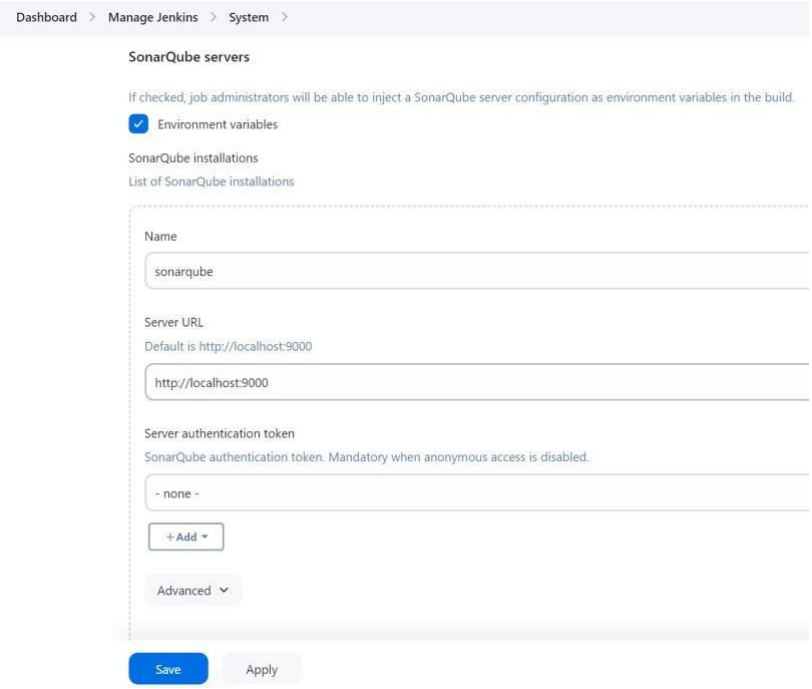


5. Setup the project and come back to Jenkins Dashboard. Go to **Manage Jenkins → Plugins** and search for **SonarQube Scanner** in **Available Plugins** and install it.

6. Under '**Manage Jenkins → System**', look for **SonarQube Servers** and enter these details.

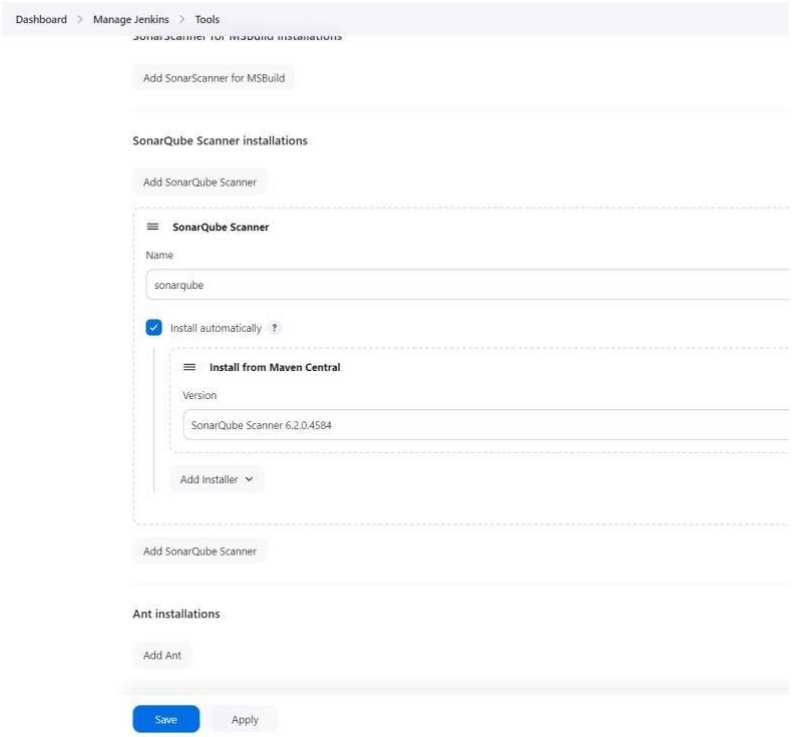
Name : sonarqube

Server URL : http://localhost:9000

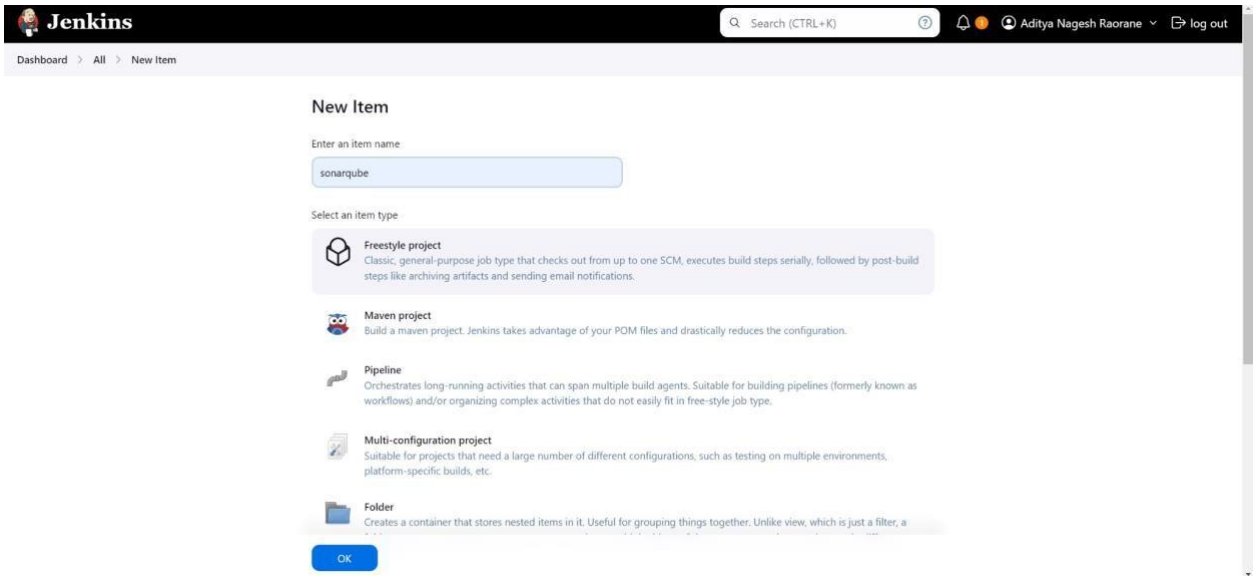


7. Search for SonarQube Scanner under Global Tool Configuration. Choose the latest configuration and choose Install automatically.

Manage Jeknins → Tools → SonarQube Scanner Installation

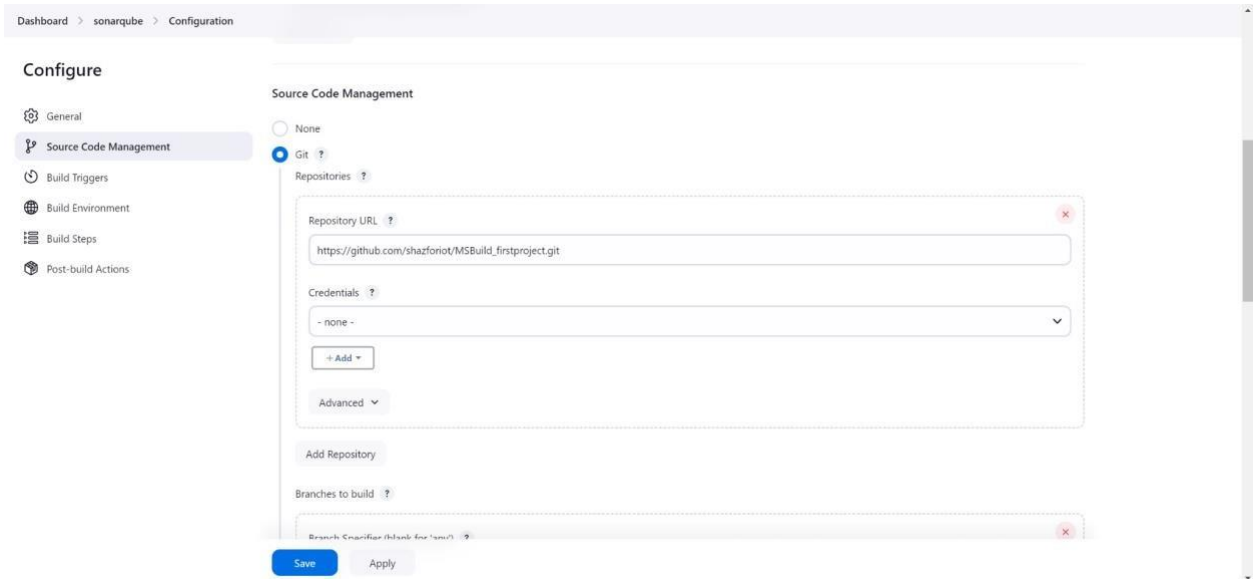


8. After the configuration, create a **New Item** in Jenkins, choose a **freestyle project** named **sonarqube**.



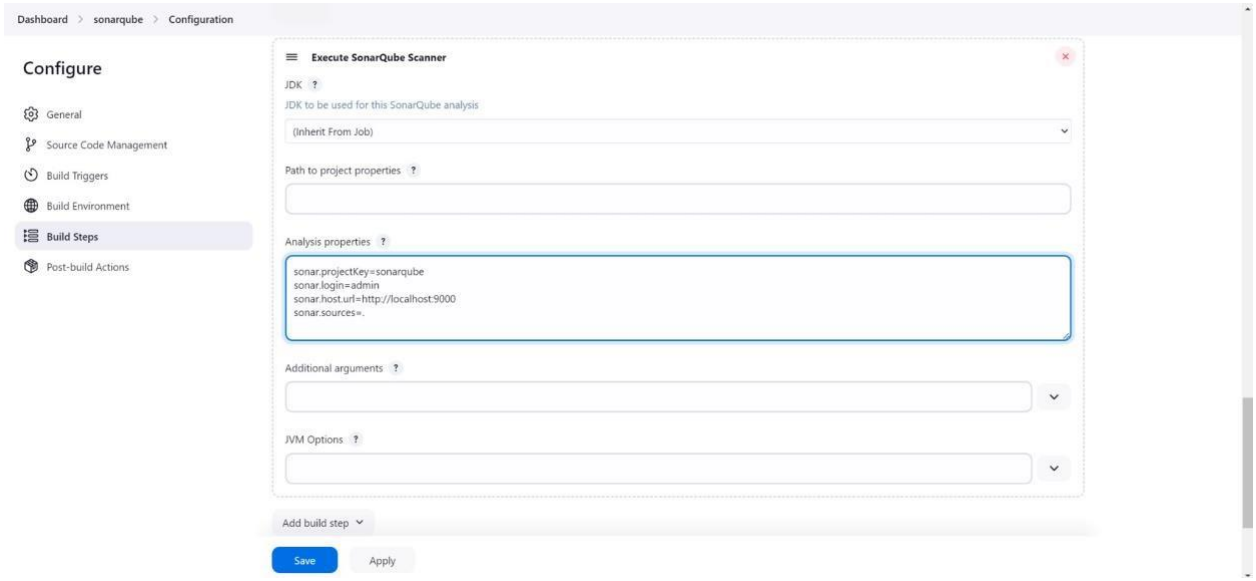
9. Choose this GitHub repository in **Source Code Management**.
   https://github.com/shazforiot/MSBuild_firstproject.git

It is a sample hello-world project with no vulnerabilities and issues, just to test the integration.

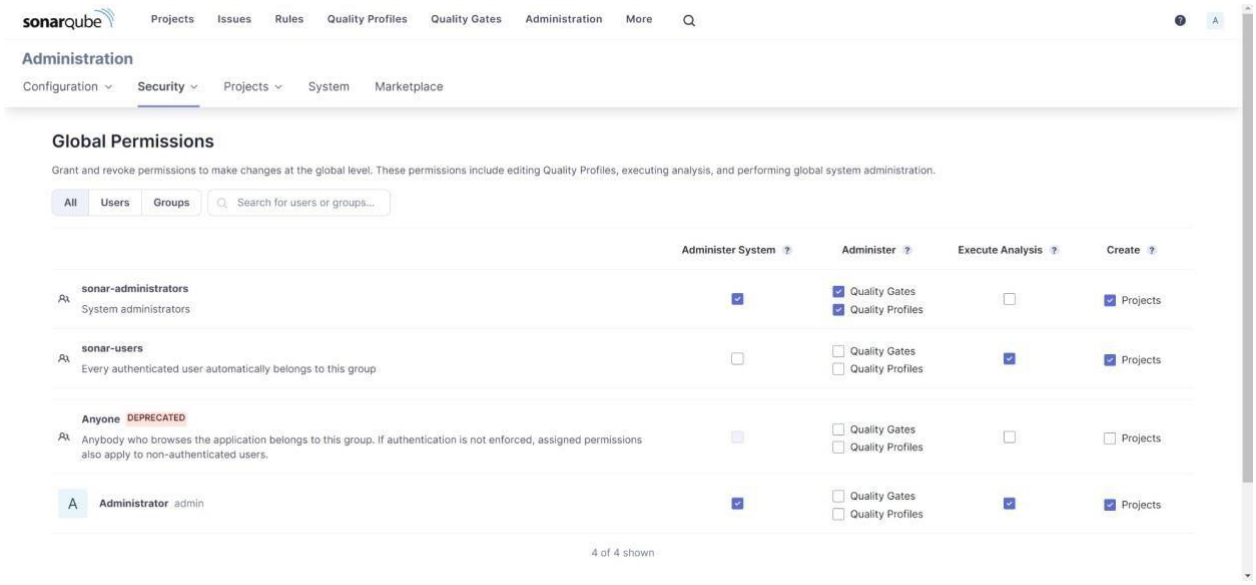


10. Under **Build-> Execute SonarQube Scanner**, enter these **Analysis Properties**. Mention the SonarQube Project Key, Login, Password, Source path and Host URL. sonar.projectKey=sonarqube sonar.login=admin sonar.password=aditya sonar.sources=.

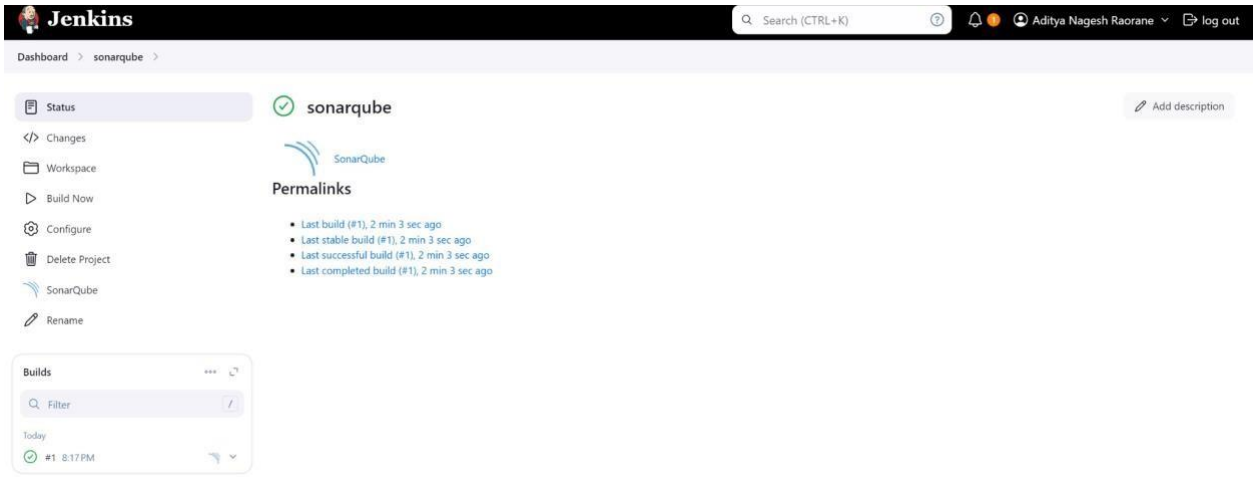
sonar.host.url=http://localhost:9000

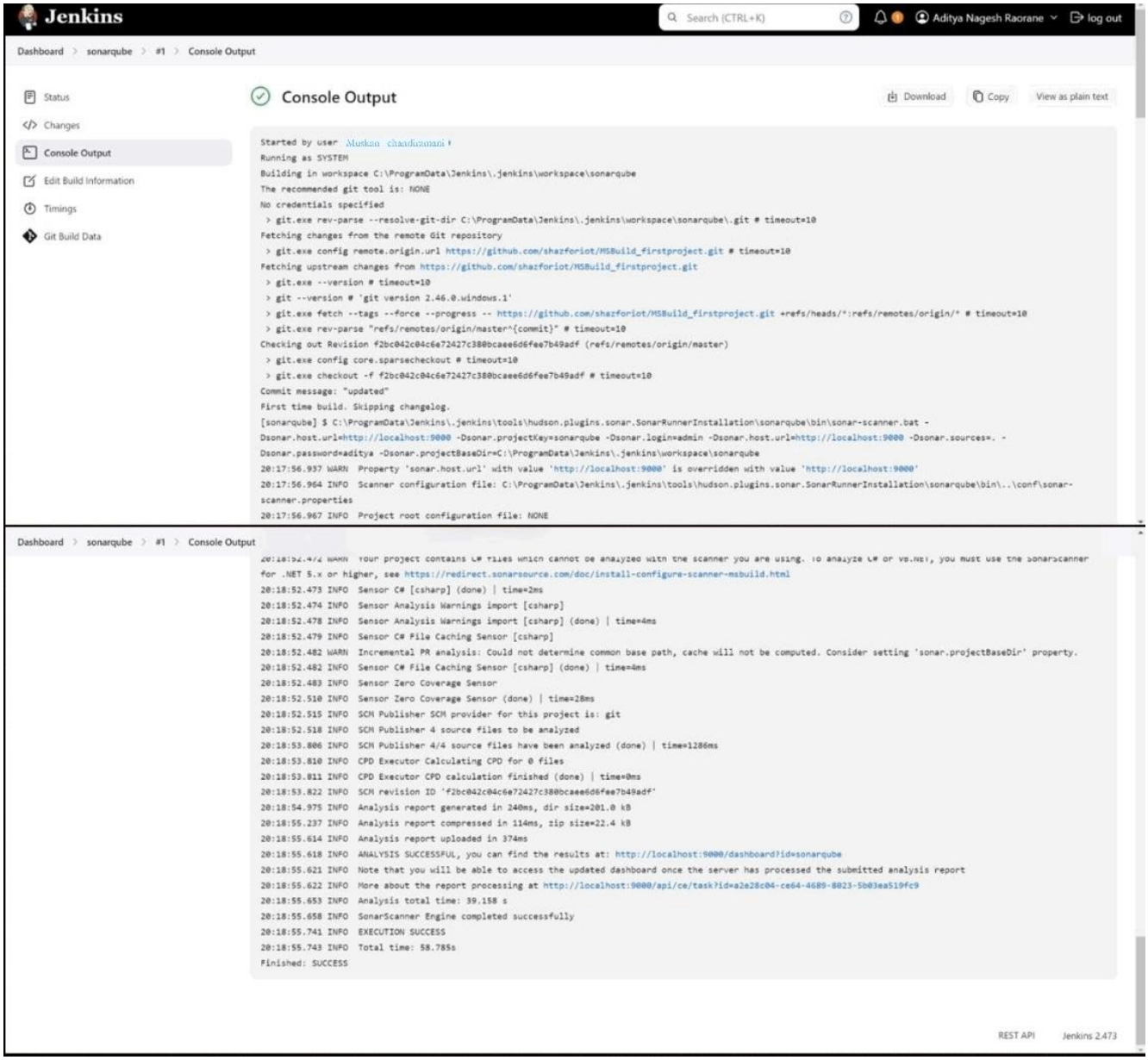


11. Go to http://localhost:9000/admin/permissions and allow Execute Permissions to the Admin user.

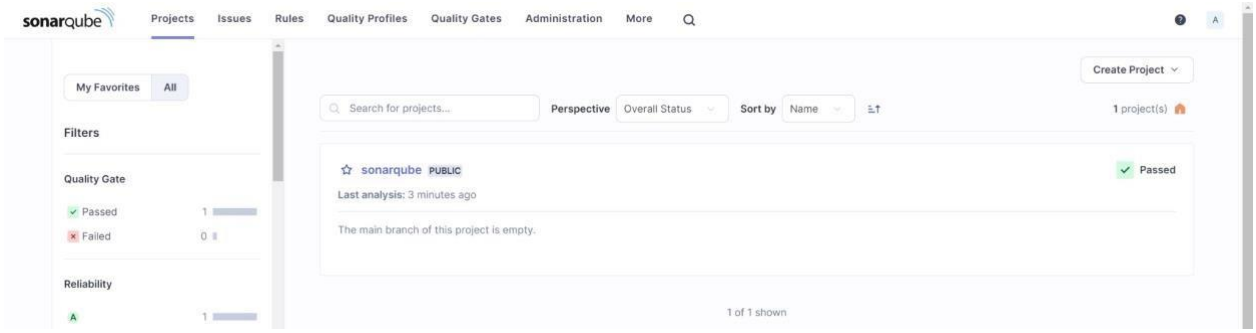


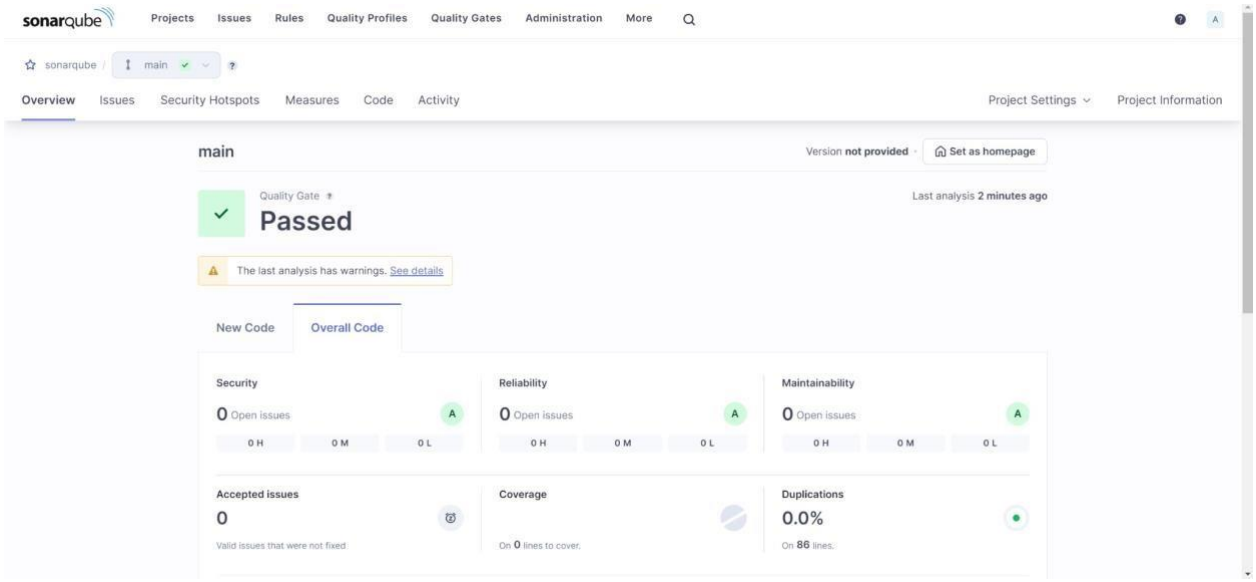
12. Run The **Build** and check the **console output**.





13. Once the build is complete, check the project in SonarQube.





In this way, we have integrated Jenkins with SonarQube for SAST.

Conclusion:

In this experiment, we have understood the importance of SAST and have successfully integrated Jenkins with SonarQube for Static Analysis and Code Testing.