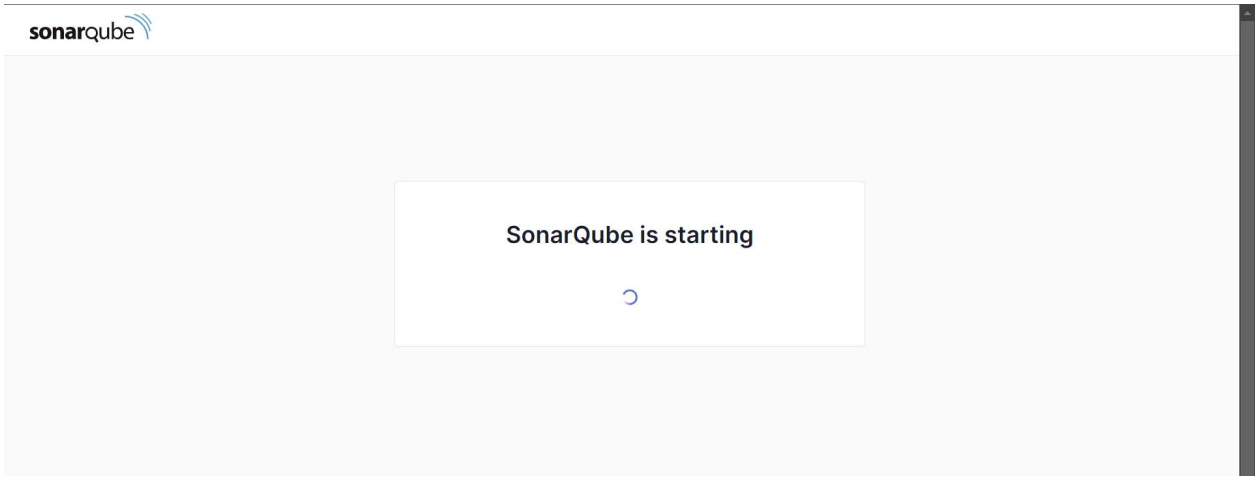
**Aim:** To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

1. Open docker desktop on your device. Then run the following command in powershell.

docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest

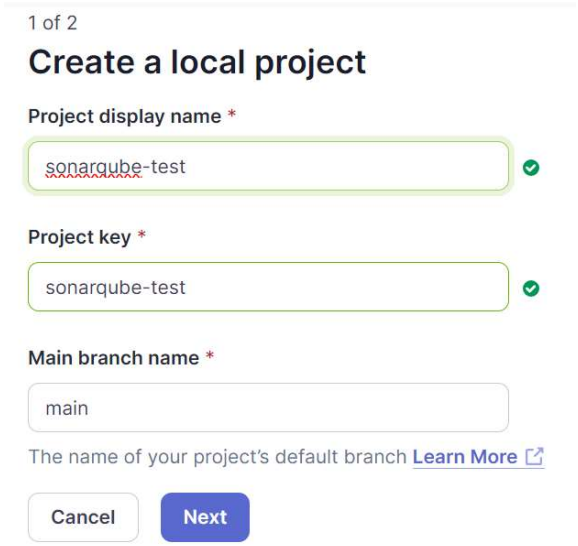
```
PS C:\Windows\system32> docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:
latest
Unable to find image 'sonarqube:latest' locally
latest: Pulling from library/sonarqube
7478e0ac0f23: Pull complete
90a925ab929a: Pull complete
7d9a34308537: Pull complete
80338217a4ab: Pull complete
1a5fd5c7e184: Pull complete
7b87d6fa783d: Pull complete
bd819c9b5ead: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Downloaded newer image for sonarqube:latest
5008a1b64daaff9ca9430d6426a4ceb303d964d039798555e9fb548ff7c89073
```

2. Once the command is run, you can verify running of sonarqube by checking the url http://localhost:9000



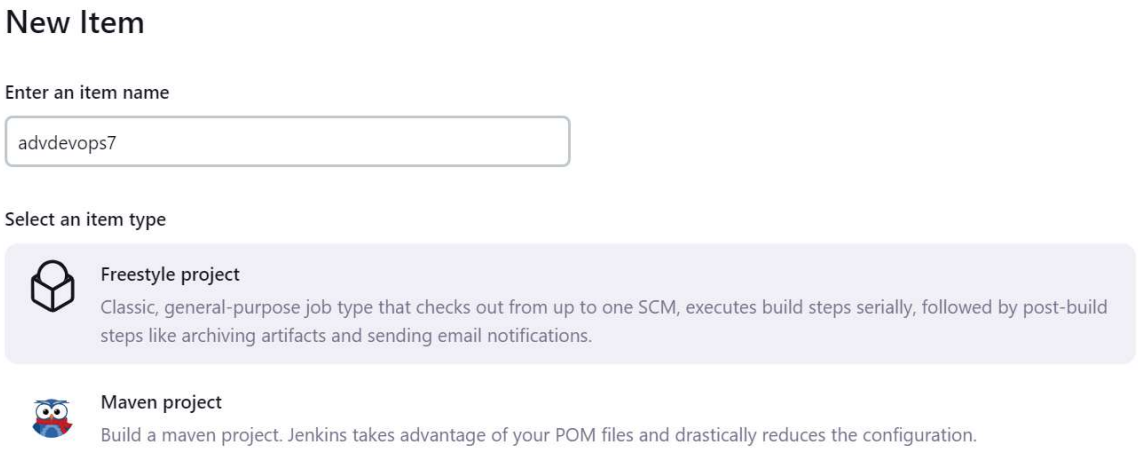

3. Login with username admin and password admin

4. Create a local project in sonarqube. Give the project a name, setup the project and click on create project.



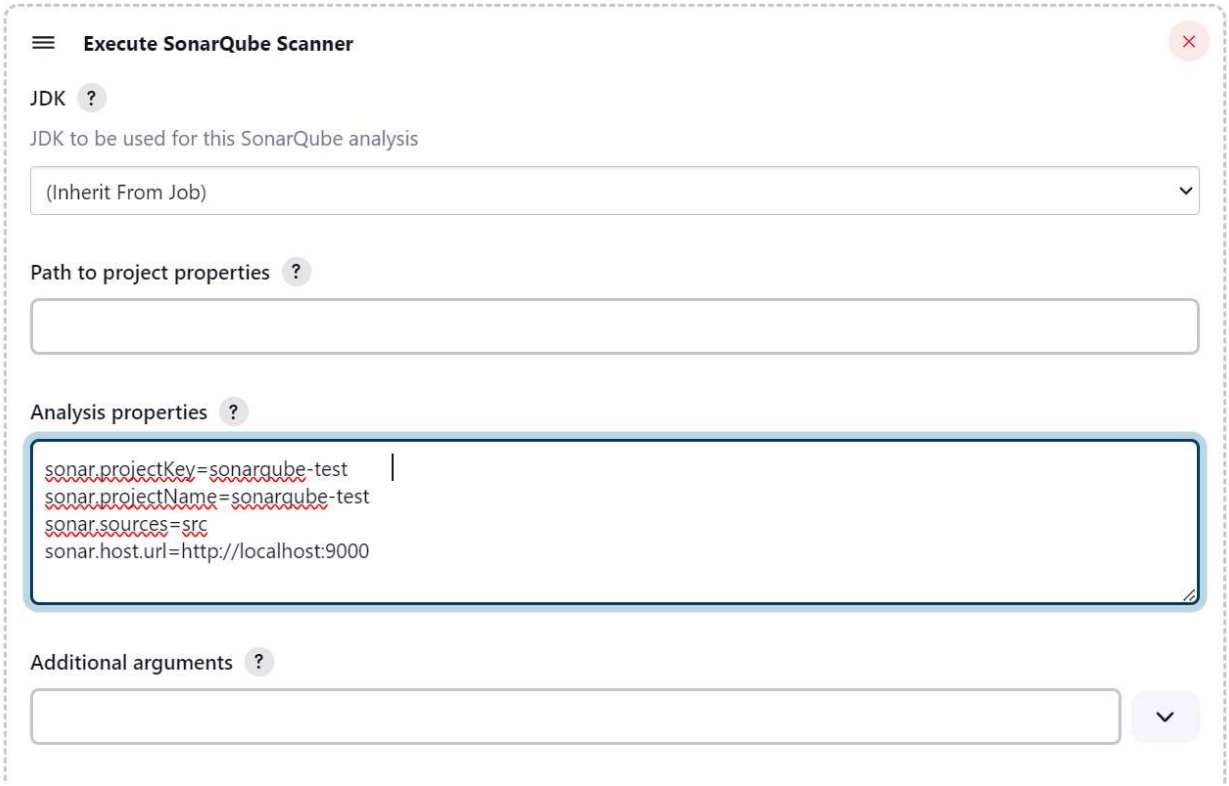

5. Go to jenkins, i.e http://localhost:8080. Go to Manage Jenkins -> Plugins -> Available Plugins. Search for SonarQube Scanner and install it.


6. Search for SonarQube Scanner under Global Tool Configuration. Choose the latest configuration and choose Install automatically.
7. Create a new project. Select the type as freestyle project.



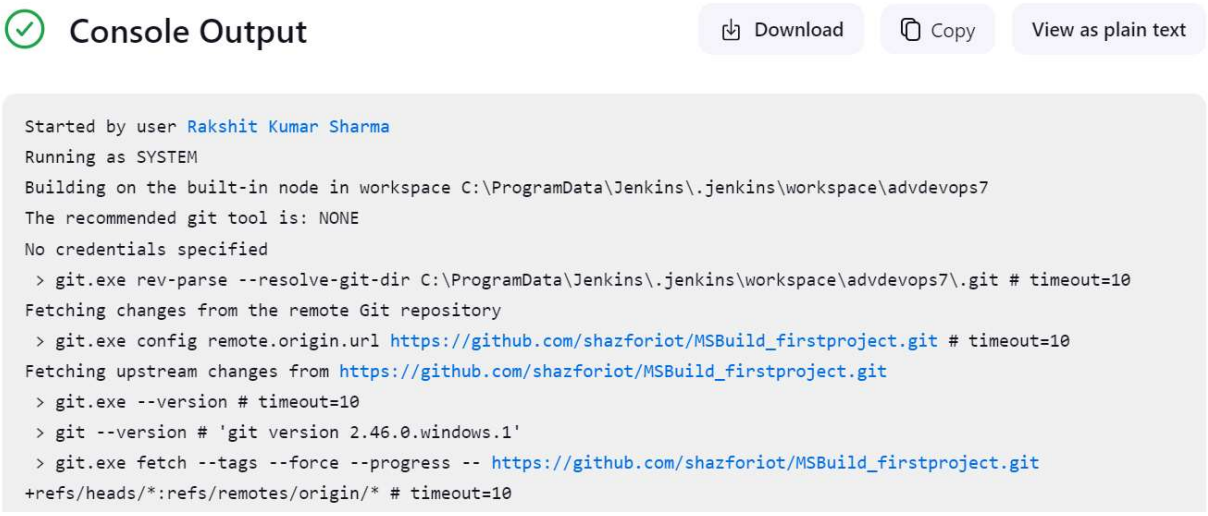
8. Select git in source code management. Enter the following as the url for repository https://github.com/shazforiot/MSBuild_firstproject.git
   It is a simple hello world project with no vulnerabilities and issues.


9. In build steps select Execute SonarQube Scanner. In analysis properties give the details about the sonarqube project.



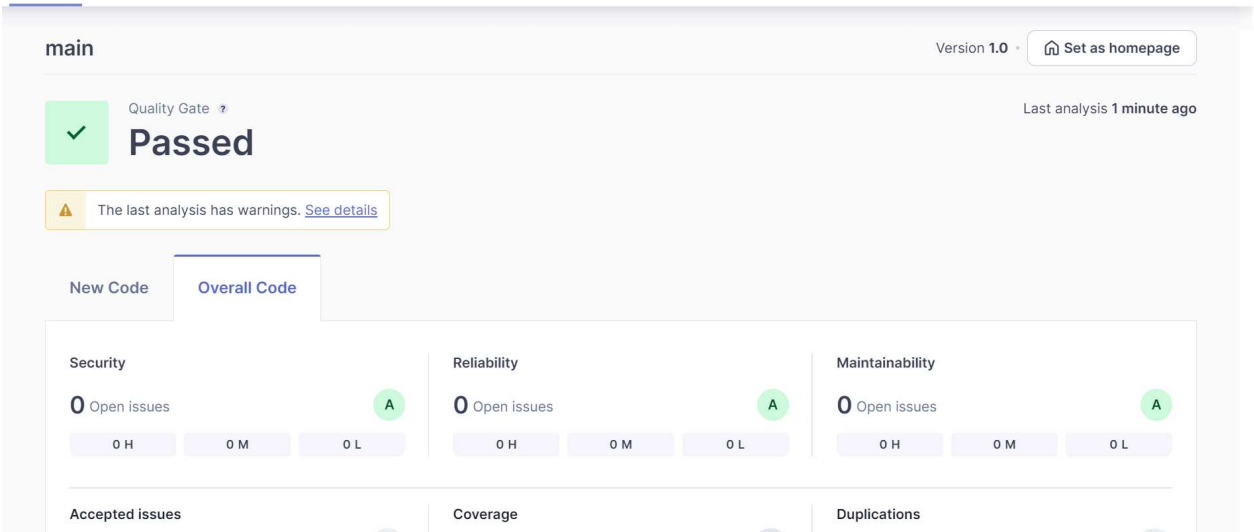
10. Go to http://localhost:9000/<user_name>/permissions and allow Execute Permissions to the Admin user.


11. Once everything is set up, Click on Build now to build the freestyle project.




Check the console output to see the successful build of the project.

12. Once Build is complete, check the sonarqube project. You will see a passed message, this indicated the code did not have any issues while building.



**Conclusion:**

In this experiment, we successfully set up a static application security testing (SAST) pipeline by integrating SonarQube with Jenkins. We initiated a SonarQube server using Docker, allowing for local access and code analysis. A new project was created in SonarQube to facilitate static code evaluation. We configured Jenkins by installing the SonarQube Scanner plugin and created a Freestyle project linked to a sample GitHub repository. By executing the SonarQube analysis within the Jenkins build steps, we provide