

Advance DevOps

## Experiment 7

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

Steps:

1. Firstly, we will ensure whether docker is installed or not by running `docker -v` in the command prompt.

```
C:\Users\Dell>docker -v
Docker version 27.1.1, build 6312585
```

2. Run docker login command and add your username and password for docker.

```
C:\Users\Dell>docker login
Authenticating with existing credentials...
Stored credentials invalid or expired
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/

Username (dimple866): dimple866
Password:
Login Succeeded
```

3. Run docker pull SonarQube command to install SonarQube image.

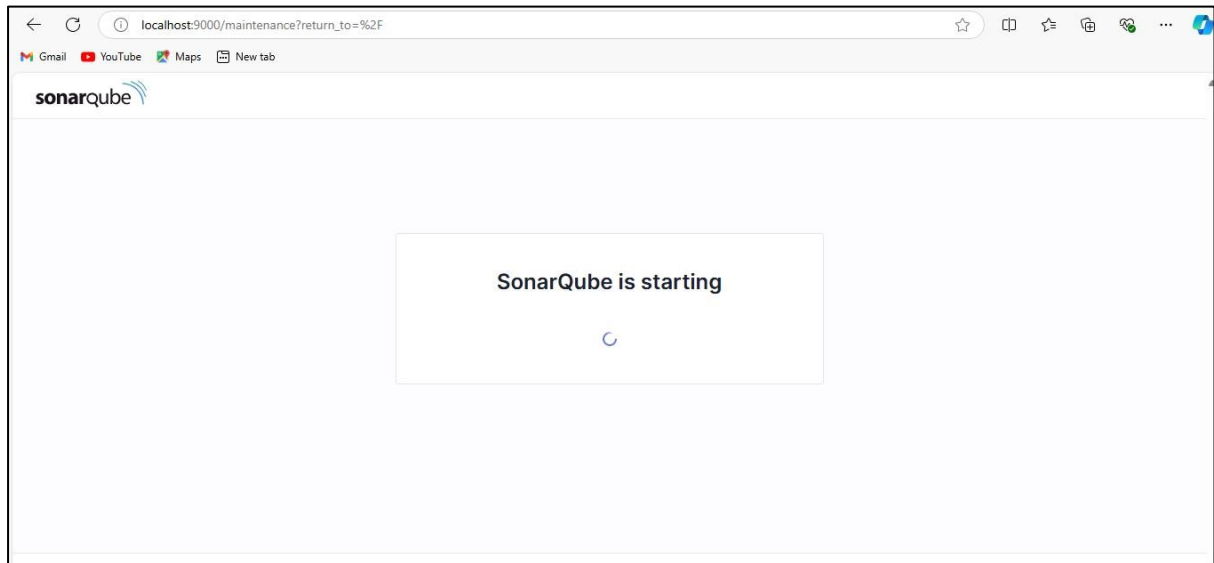
```
C:\Users\Dell>docker pull sonarqube
Using default tag: latest
7478e0ac0f23: Pull complete
90a925ab929a: Pull complete
7d9a34308537: Pull complete
80338217a4ab: Pull complete
1a5fd5c7e184: Pull complete
7b87d6fa783d: Pull complete
bd819c9b5ead: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Downloaded newer image for sonarqube:latest
docker.io/library/sonarqube:latest

What's next:
View a summary of image vulnerabilities and recommendations → docker scout quickview sonarqube
```

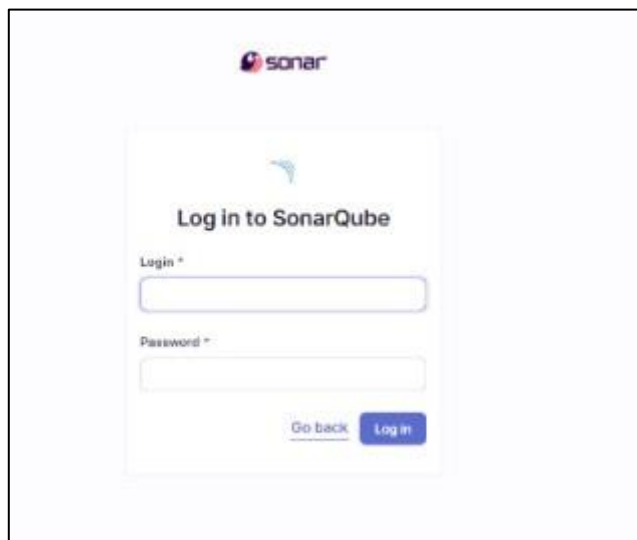
4. Run `docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest` Command to run the sonarqube.

```
C:\Users\Dell>docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
ac1f985dedebc00a642a4c69a502d611389e8f9fa46610febe75aa5021767cab
```

5. Once the container is running go to your web browser and check status of SonarQube at port 9000.




6. Once SonarQube is started it will redirect you to login page. The login and password both for SonarQube is 'admin'



7. Change the password for your SonarQube account.

## Update your password

 This account should not use the default password.

### Enter a new password

All fields marked with \* are required

Old Password \*

New Password \*

Confirm Password \*

Update

8. After changing the password, you will be directed to this screen. Click on Create a Local Project.

sonarqube

ProjectsIssuesRulesQuality ProfilesQuality GatesAdministrationMore

How do you want to create your project?

Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)?  
Create your project from your favorite DevOps platform.

First, you need to set up a DevOps platform configuration.

Import from Azure DevOpsSetup

Import from Bitbucket CloudSetup

Import from Bitbucket ServerSetup

Import from GitHubSetup

Import from GitLabSetup

Are you just testing or have an advanced use-case? Create a local project.

Create a local project

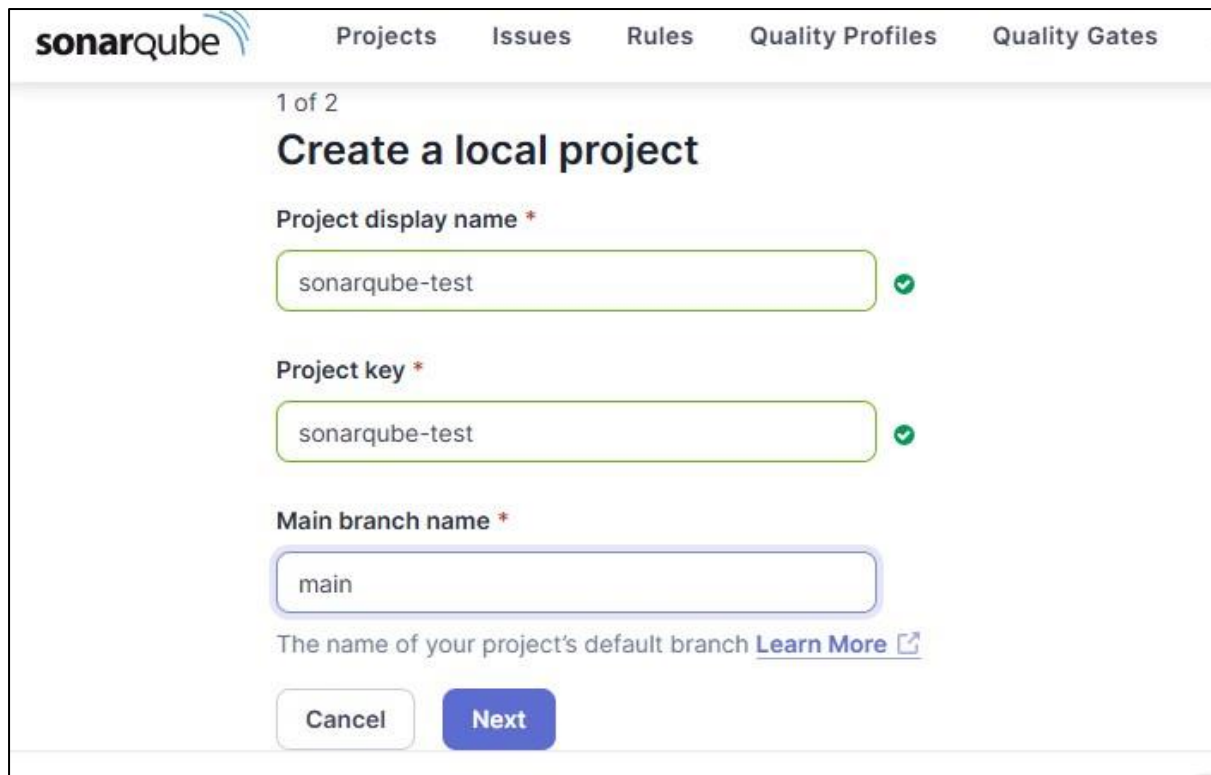
Get the most out of SonarQube!

Take advantage of the whole ecosystem by using SonarLint, a free IDE plugin that helps you find and fix issues earlier in your workflow. Connect SonarLint to SonarQube to sync rule sets and issue states.

Learn More

Dismiss

9. Add name of the project and project key and select the main branch name and click on next.



The image shows the 'Create a local project' form in SonarQube. The form is part of a two-step process, currently on step 1 of 2. It includes fields for 'Project display name', 'Project key', and 'Main branch name'. The 'Project display name' and 'Project key' fields are filled with 'sonarqube-test' and have green checkmarks indicating they are valid. The 'Main branch name' field is filled with 'main'. Below the fields, there is a link 'Learn More' and two buttons: 'Cancel' and 'Next'.

sonarqube

Projects Issues Rules Quality Profiles Quality Gates

1 of 2

## Create a local project

Project display name \*

sonarqube-test ✓

Project key \*

sonarqube-test ✓

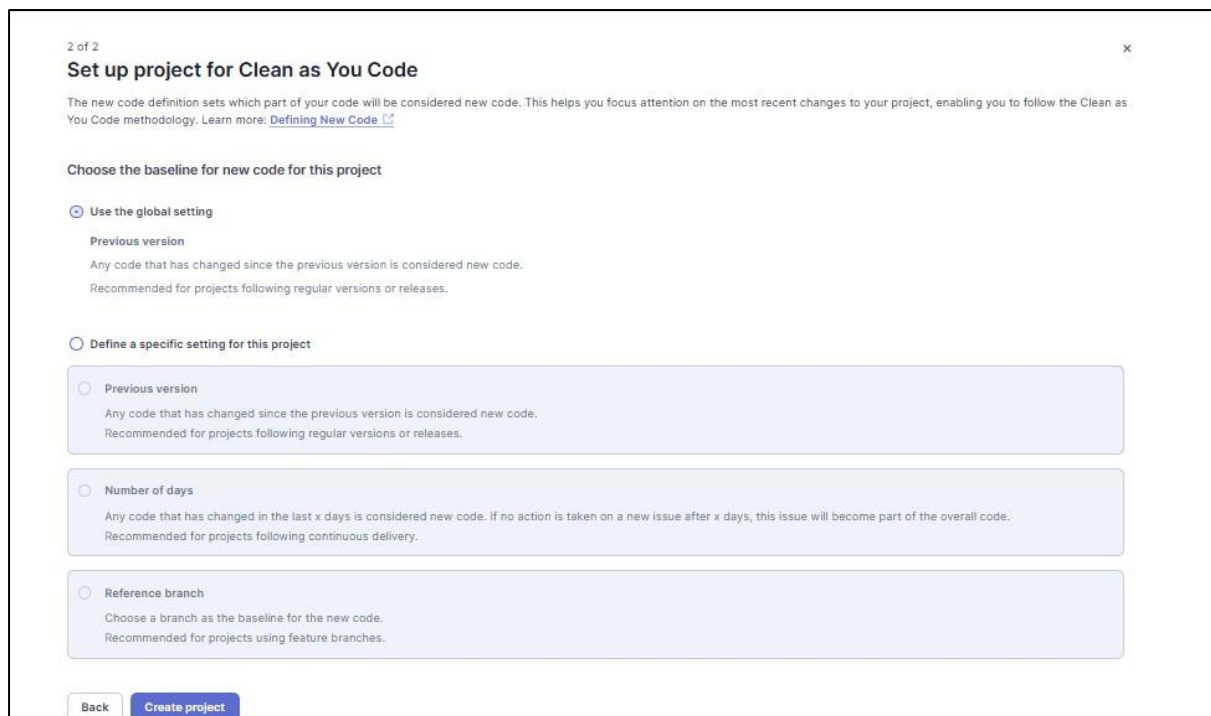
Main branch name \*

main

The name of your project's default branch [Learn More](#)

Cancel Next

10. Set up the project as required and click on create.



The image shows the 'Set up project for Clean as You Code' form in SonarQube. This is step 2 of 2. The form explains the 'Clean as You Code' methodology and provides options to choose the baseline for new code. The 'Use the global setting' option is selected. Below this, there are three options for defining a specific setting for this project: 'Previous version', 'Number of days', and 'Reference branch'. Each option has a brief description and a recommendation. At the bottom, there are 'Back' and 'Create project' buttons.

2 of 2

### Set up project for Clean as You Code

The new code definition sets which part of your code will be considered new code. This helps you focus attention on the most recent changes to your project, enabling you to follow the Clean as You Code methodology. Learn more: [Defining New Code](#)

Choose the baseline for new code for this project

☒ Use the global setting

**Previous version**  
Any code that has changed since the previous version is considered new code.  
Recommended for projects following regular versions or releases.

☐ Define a specific setting for this project

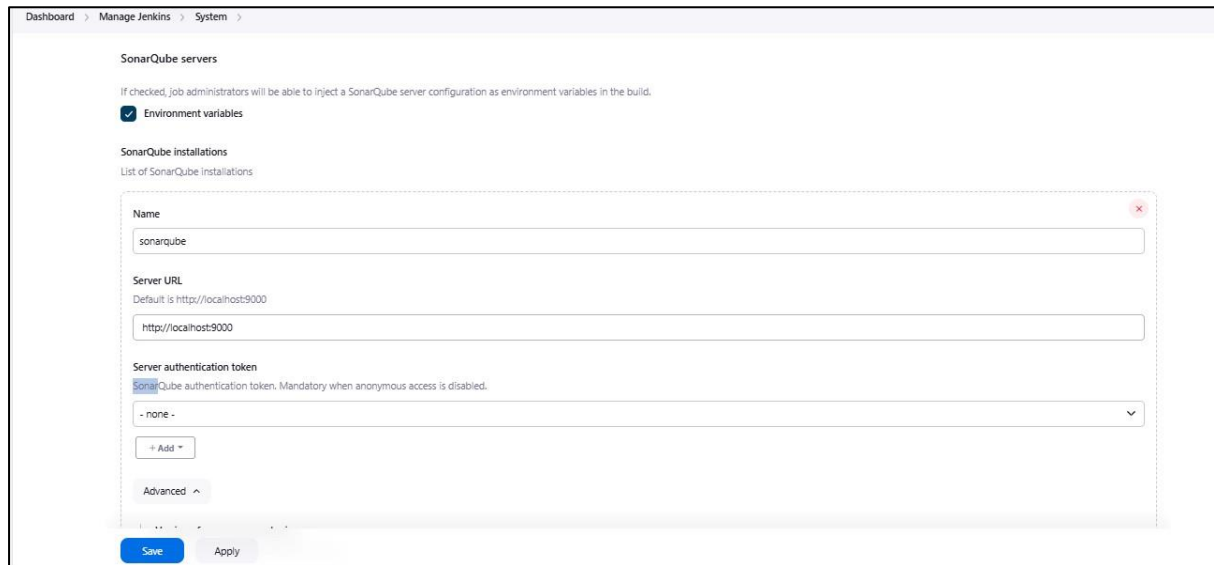
☐ Previous version  
Any code that has changed since the previous version is considered new code.  
Recommended for projects following regular versions or releases.

☐ Number of days  
Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will become part of the overall code.  
Recommended for projects following continuous delivery.

☐ Reference branch  
Choose a branch as the baseline for the new code.  
Recommended for projects using feature branches.

Back Create project

11. Go to Jenkins dashboard->Manage Jenkins->System and scroll down to SonarQube installations. Enter the name and URL in the fields and save the changes.



The screenshot shows the Jenkins 'System' configuration page under 'Manage Jenkins'. The 'SonarQube servers' section is active, with the 'Environment variables' checkbox checked. Below this, the 'SonarQube installations' section is visible, showing a list of installations. A new installation is being added with the following details:

- Name:** sonarqube
- Server URL:** http://localhost:9000
- Server authentication token:** - none -

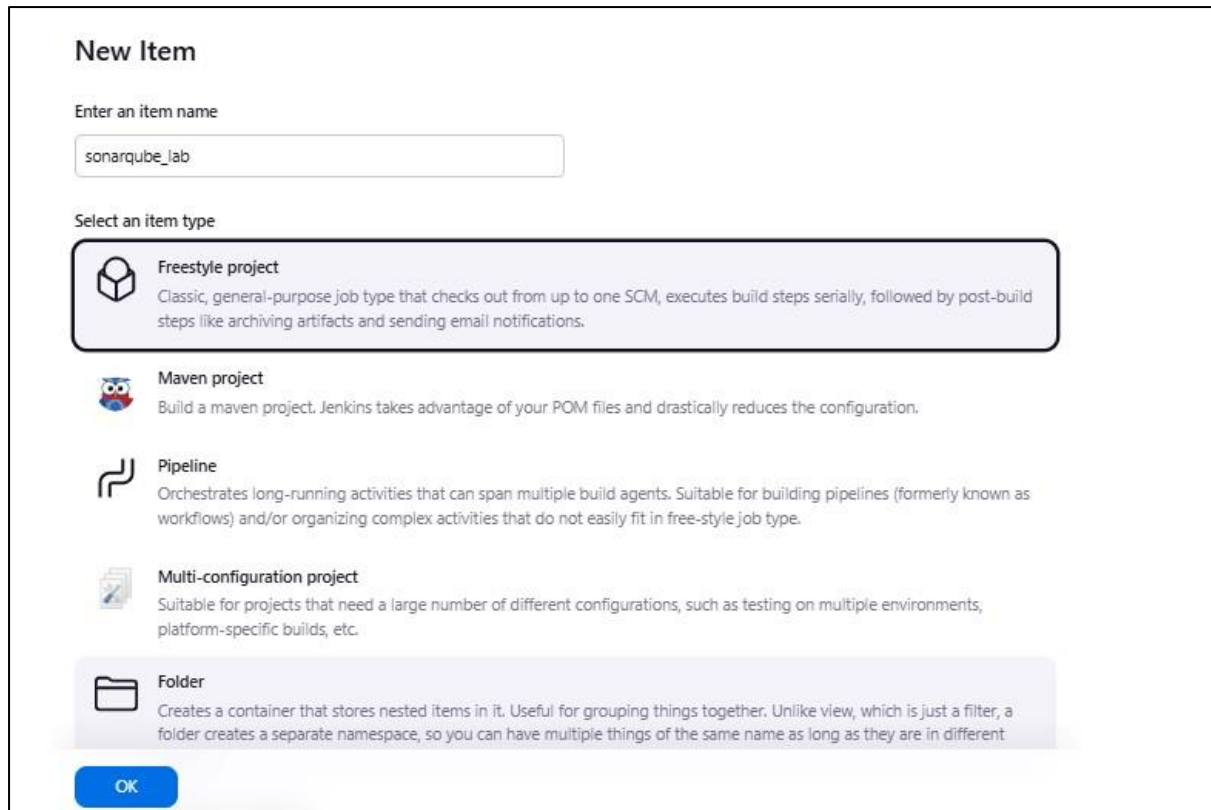
At the bottom of the form, there are 'Save' and 'Apply' buttons.

12. In SonarQube Scanner add the latest version then apply the changes and save it.



The screenshot shows the Jenkins configuration page for the 'SonarQube Scanner'. The 'Name' field is set to 'sonarqube\_lab'. The 'Install automatically' checkbox is checked. Under the 'Install from Maven Central' section, the 'Version' dropdown is set to 'SonarQube Scanner 6.2.0.4584'. At the bottom, there are 'Save' and 'Apply' buttons.

13. Go to Jenkins and then create a new item, enter the item name and select “Freestyle project” and then click on ok.



The image shows the 'New Item' dialog box in Jenkins. At the top, it says 'New Item'. Below that, there is a label 'Enter an item name' and a text input field containing 'sonarqube\_lab'. Underneath, there is a label 'Select an item type' and a list of options. The 'Freestyle project' option is selected and highlighted with a blue border. It has a cube icon and a description: 'Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.' Other options include 'Maven project' (owl icon), 'Pipeline' (curved arrow icon), 'Multi-configuration project' (document icon), and 'Folder' (folder icon). At the bottom left, there is a blue 'OK' button.

**New Item**

Enter an item name

sonarqube\_lab

Select an item type

**Freestyle project**  
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

**Maven project**  
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

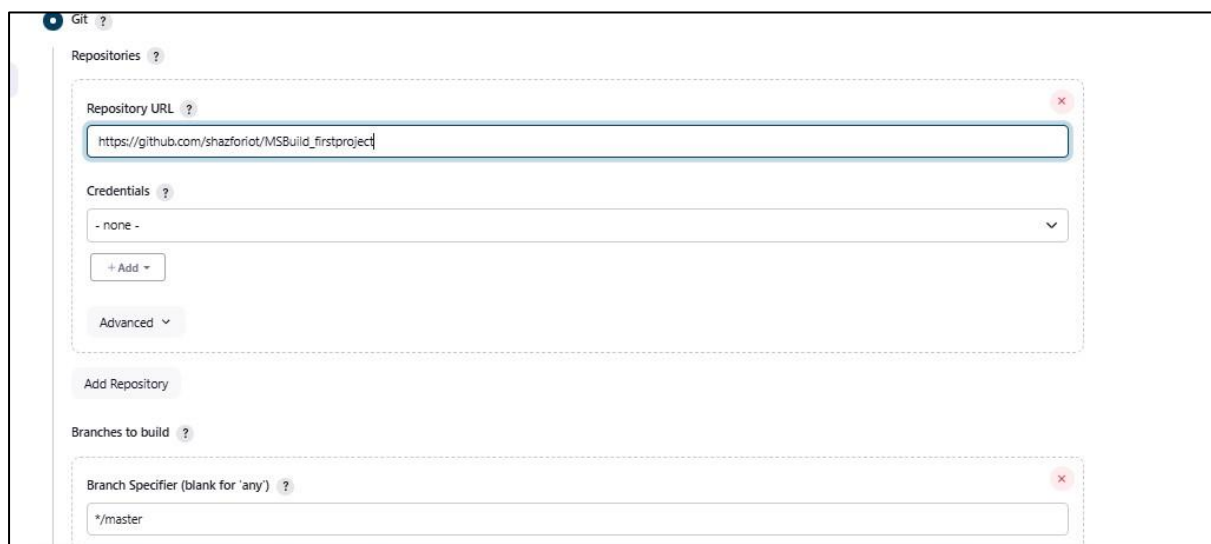
**Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

**Folder**  
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different

OK

14. Use this GitHub repository in Source Code Management.  
[https://github.com/shazforiot/MSBuild\\_firstproject](https://github.com/shazforiot/MSBuild_firstproject)



The image shows the 'Add Repository' dialog box in Jenkins. It has a title bar 'Git ?'. Below the title bar, there is a label 'Repositories ?'. The main form has three sections. The first section is 'Repository URL ?' with a text input field containing 'https://github.com/shazforiot/MSBuild\_firstproject'. The second section is 'Credentials ?' with a dropdown menu showing '- none -' and a '+ Add +' button. The third section is 'Branches to build ?' with a label 'Branch Specifier (blank for 'any') ?' and a text input field containing '\*/master'. At the bottom, there is a button 'Add Repository'.

Git ?

Repositories ?

Repository URL ?

https://github.com/shazforiot/MSBuild\_firstproject

Credentials ?

- none -

+ Add +

Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/master

15. In Analysis properties, mention the SonarQube Project Key, Login, Password, Source path and Host URL.

**Execute SonarQube Scanner**

JDK ?  
JDK to be used for this SonarQube analysis  
([Inherit From Job])

Path to project properties ?  
[Empty field]

Analysis properties ?  
sonar.projectKey=sonarqube  
sonar.login=admin  
sonar.password=123456  
sonar.host.url=http://localhost:9000  
sonar.sources=.

Additional arguments ?  
[Empty field]

JVM Options ?  
[Empty field]

16. Now, you need to grant the local user (here admin user) permissions to Execute the Analysis stage on SonarQube. For this go to [http://localhost:<port\\_number>/admin/permissions](http://localhost:<port_number>/admin/permissions) and check the 'Execute Analysis' checkbox under Administrator.

**sonarqube** Projects Issues Rules Quality Profiles Quality Gates Administration More

**Administration**  
Configuration Security Projects System Marketplace

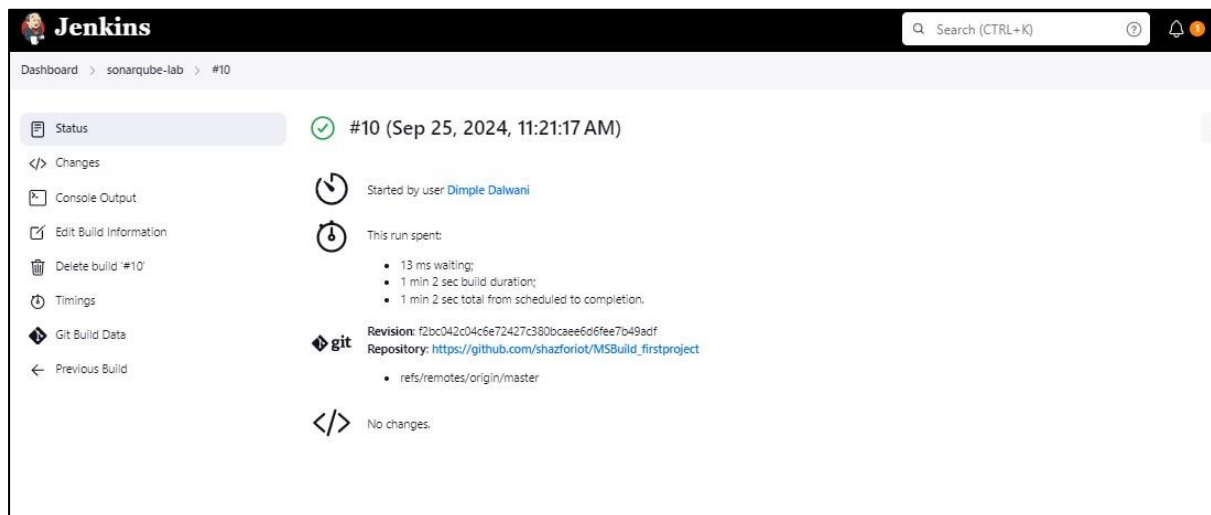
**Global Permissions**  
Grant and revoke permissions to make changes at the global level. These permissions include editing Quality Profiles, executing analysis, and performing global system administration.

All Users Groups Search for users or groups...

	Administer System ?	Administer ?	Execute Analysis ?	Create ?
<b>sonar-administrators</b> System administrators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Quality Gates <input checked="" type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input checked="" type="checkbox"/> Projects
<b>sonar-users</b> Every authenticated user automatically belongs to this group	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Projects
<b>Anyone DEPRECATED</b> Anybody who browses the application belongs to this group. If authentication is not enforced, assigned permissions also apply to non-authenticated users.	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input type="checkbox"/>	<input type="checkbox"/> Projects
<b>A Administrator</b> admin	<input checked="" type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input type="checkbox"/> Projects

4 of 4 shown

17. Go to the job you have just built and click on Build Now.

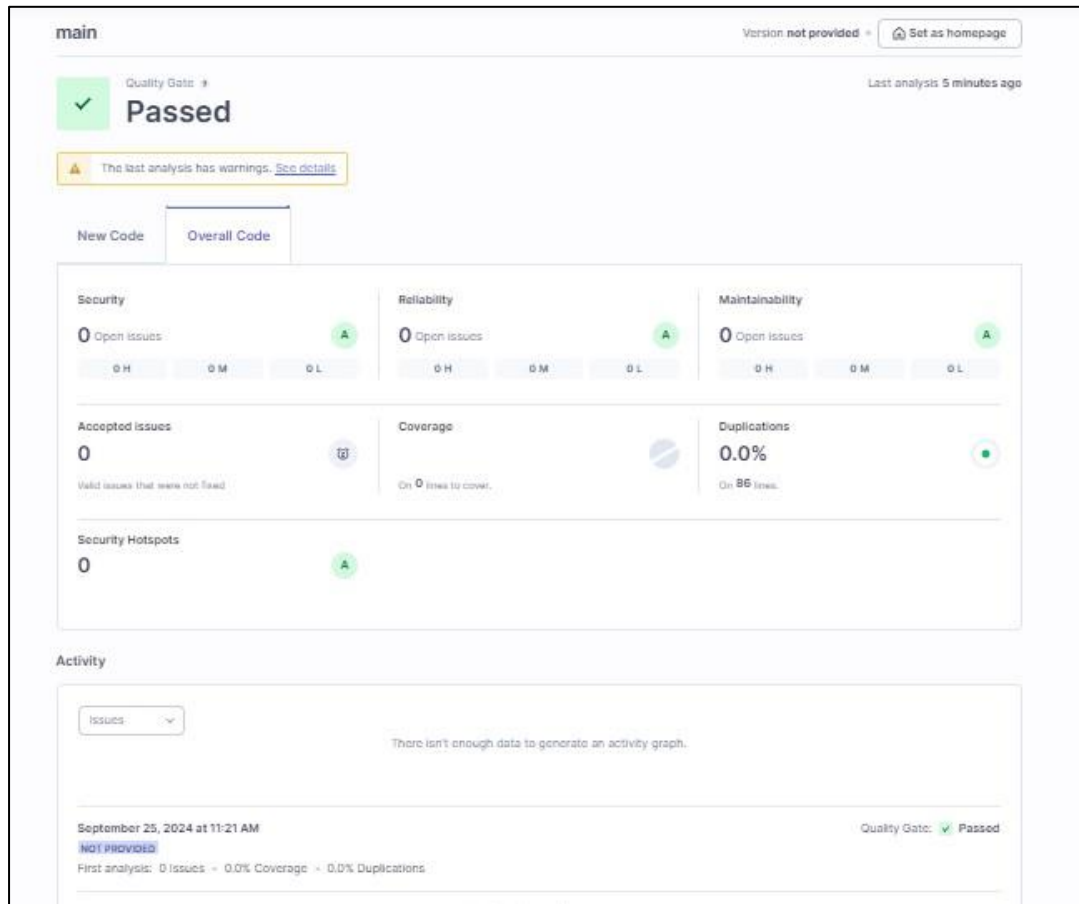


18. Check the console Output

```
for block at line 17. Keep only the first 100 references.
23:13:58.632 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/gui/util/TextAreaCellRenderer.html
for block at line 296. Keep only the first 100 references.
23:13:58.632 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/gui/util/TextAreaCellRenderer.html
for block at line 75. Keep only the first 100 references.
23:13:58.632 INFO CPD Executor CPD calculation finished (done) | time=94361ms
23:13:58.695 INFO SCM revision ID 'ba799ba7e1b576f04a612322b0412c5e6e1e5e4'
23:15:46.177 INFO Analysis report generated in 14542ms, dir size=127.2 MB
23:15:55.734 INFO Analysis report compressed in 9547ms, zip size=29.6 MB
23:15:59.127 INFO Analysis report uploaded in 3391ms
23:15:59.132 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://127.0.0.1:9000/dashboard?id=sonarqube-test
23:15:59.132 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
23:15:59.132 INFO More about the report processing at http://127.0.0.1:9000/api/ce/task?id=fbad731f-dcba-45c3-bfdd-b2ed2fec3a9e
23:16:05.629 INFO Analysis total time: 10:30.120 s
23:16:05.636 INFO SonarScanner Engine completed successfully
23:16:06.248 INFO EXECUTION SUCCESS
23:16:06.273 INFO Total time: 10:47.728s
[Pipeline] }
[Pipeline] // withSonarQubeEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```



19. Go back to SonarQube and check the project.



Conclusion: While performing this experiment there was an issue in creating sonarqube docker image and we resolved it by logging in to the docker desktop and performing it through the terminal. Other than this we created a freestyle project and entered the sonarqube credentials and then performed build.