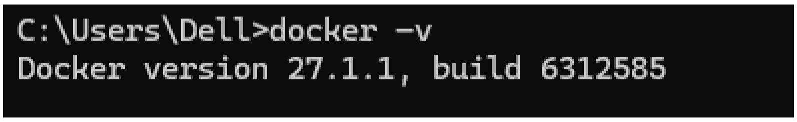
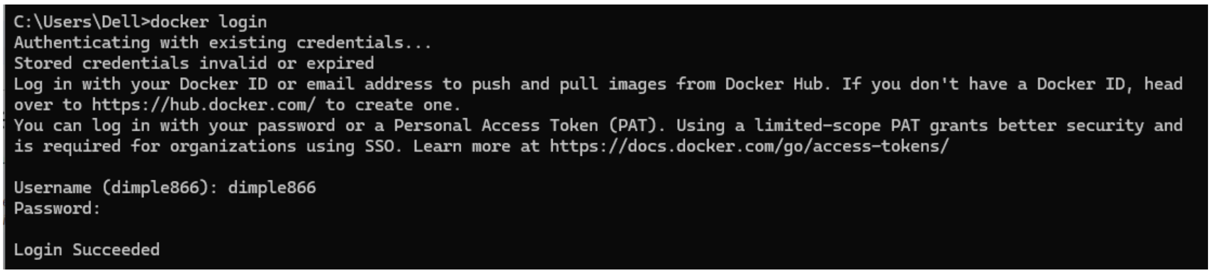
AIM:To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

STEP 1:In your command prompt to ensure whether the docker is installed or not.



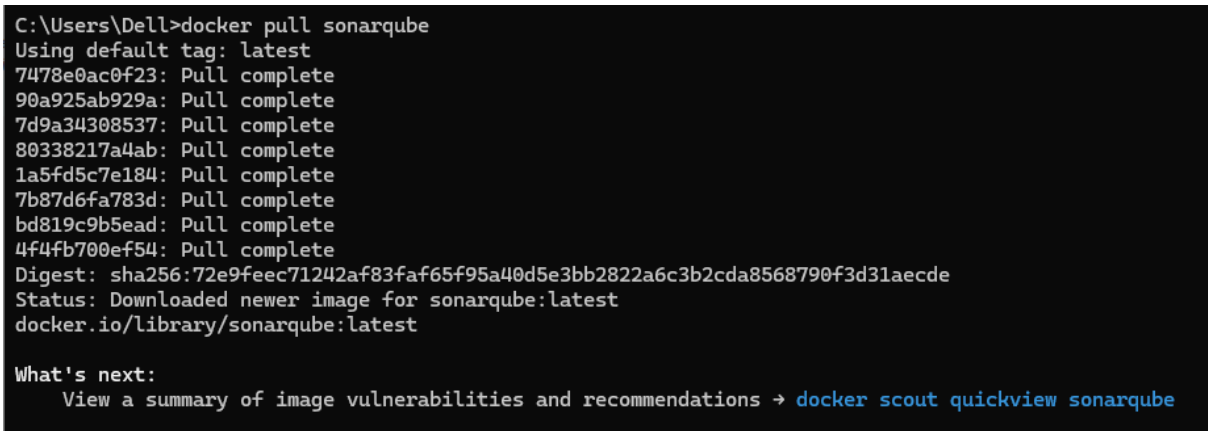
```
C:\Users\Dell>docker -v
Docker version 27.1.1, build 6312585
```

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



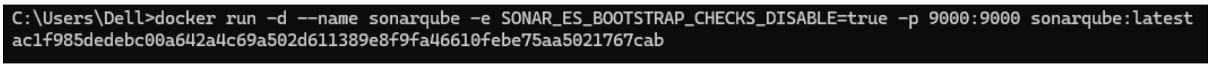
```
C:\Users\Dell>docker login
Authenticating with existing credentials...
Stored credentials invalid or expired
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head
over to https://hub.docker.com/ to create one.
You can log in with 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
```

STEP 3: Run docker pull sonarqube command to install sonarqube image without actually installing then.



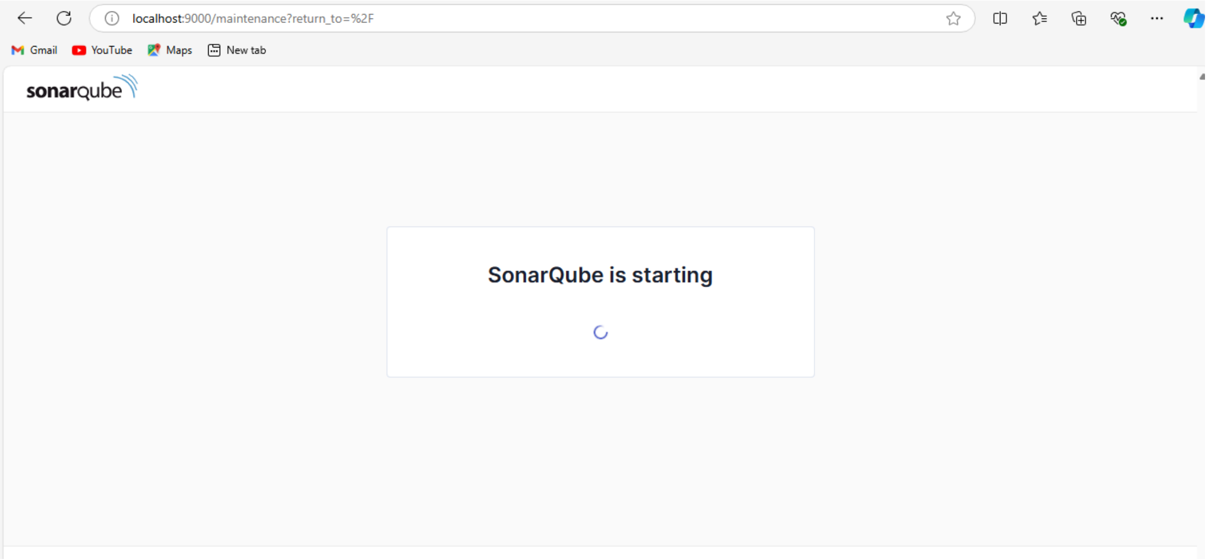
```
C:\Users\Dell>docker pull sonarqube
Using default tag: latest
7478e0ac0f23: Pull complete
90a925ab929a: Pull complete
7d9a34308537: Pull complete
80338217a4ab: Pull complete
1a5fd5c7e184: Pull complete
7b87d6fa783d: Pull complete
bd819c9b5ead: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
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
```

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



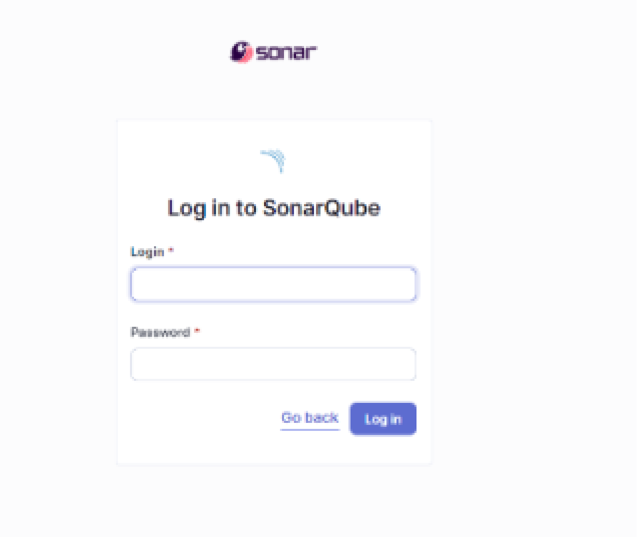
```
C:\Users\Dell>docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
ac1f985dedebc00a642a4c69a502d611389e8f9fa46610febe75aa5021767cab
```

STEP 5: Once the sonarqube is runned go to your web browser and whatever port number u have mentioned in the previous command open that page using localhost:9000.



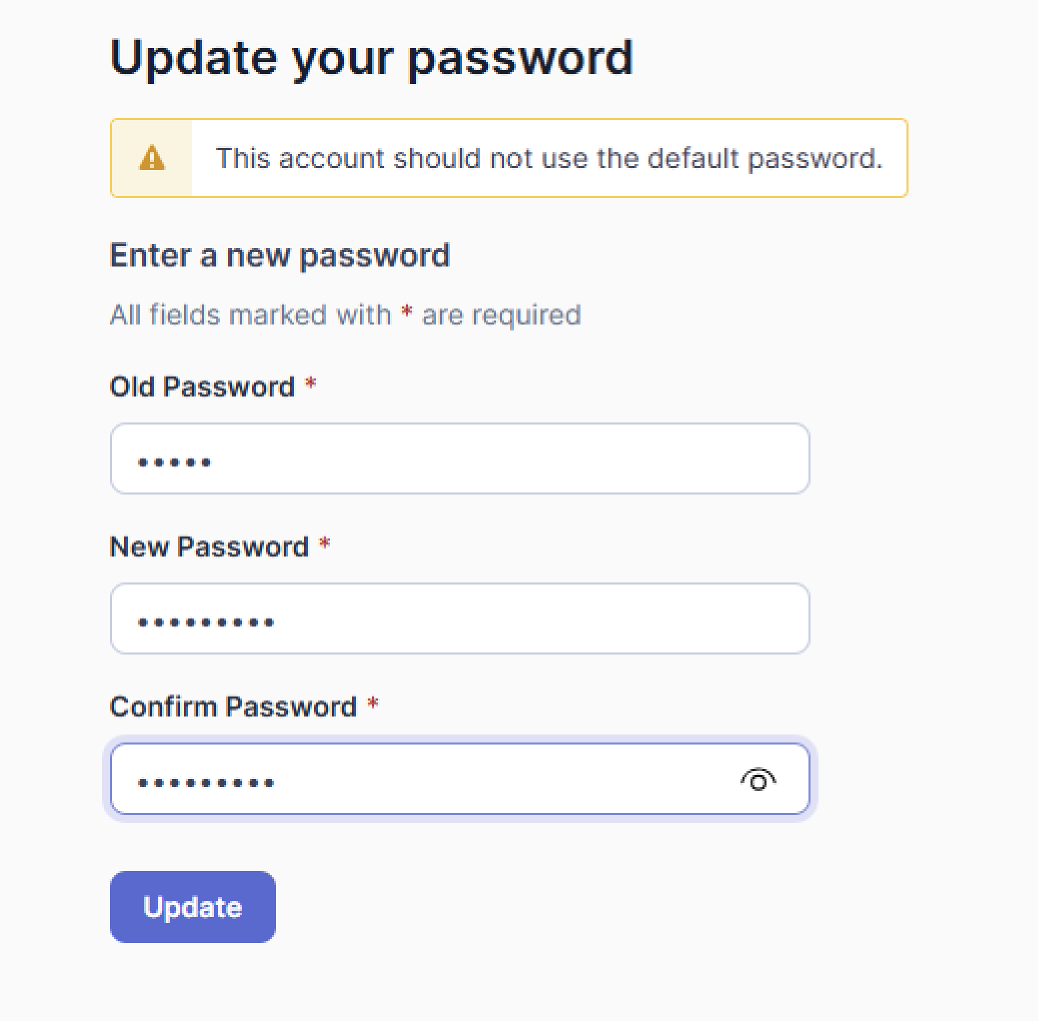
SonarQube is starting

STEP 6: Once sonarqube is started it will redirect you to login page .The login and password for sonarqube is both “admin”.



Log in to SonarQube

STEP 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 *

•••••

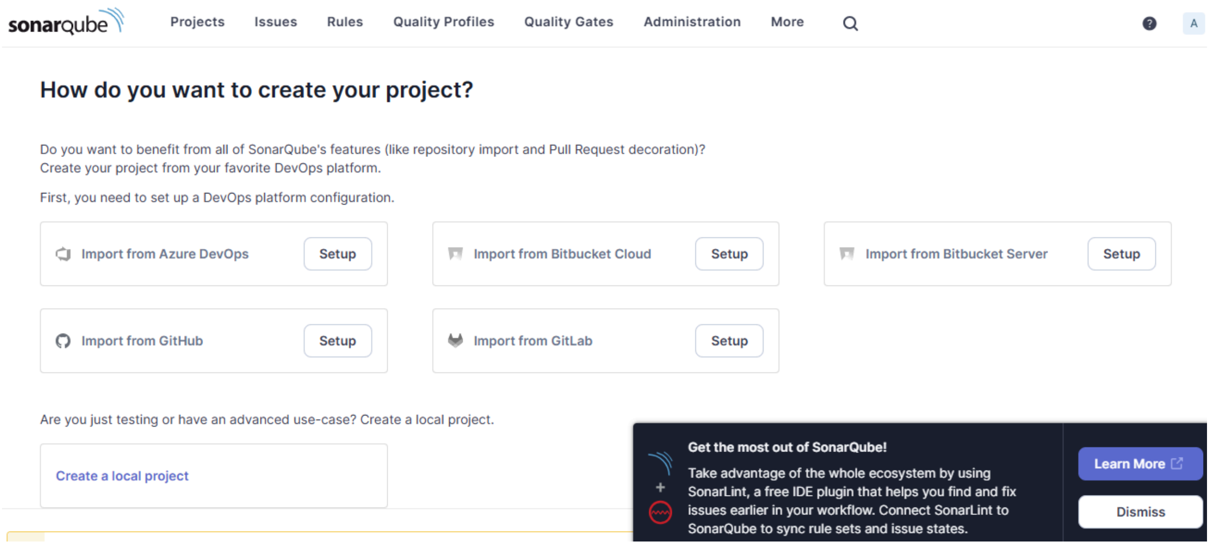
New Password *

••••••••••

Confirm Password *

••••••••••

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



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.

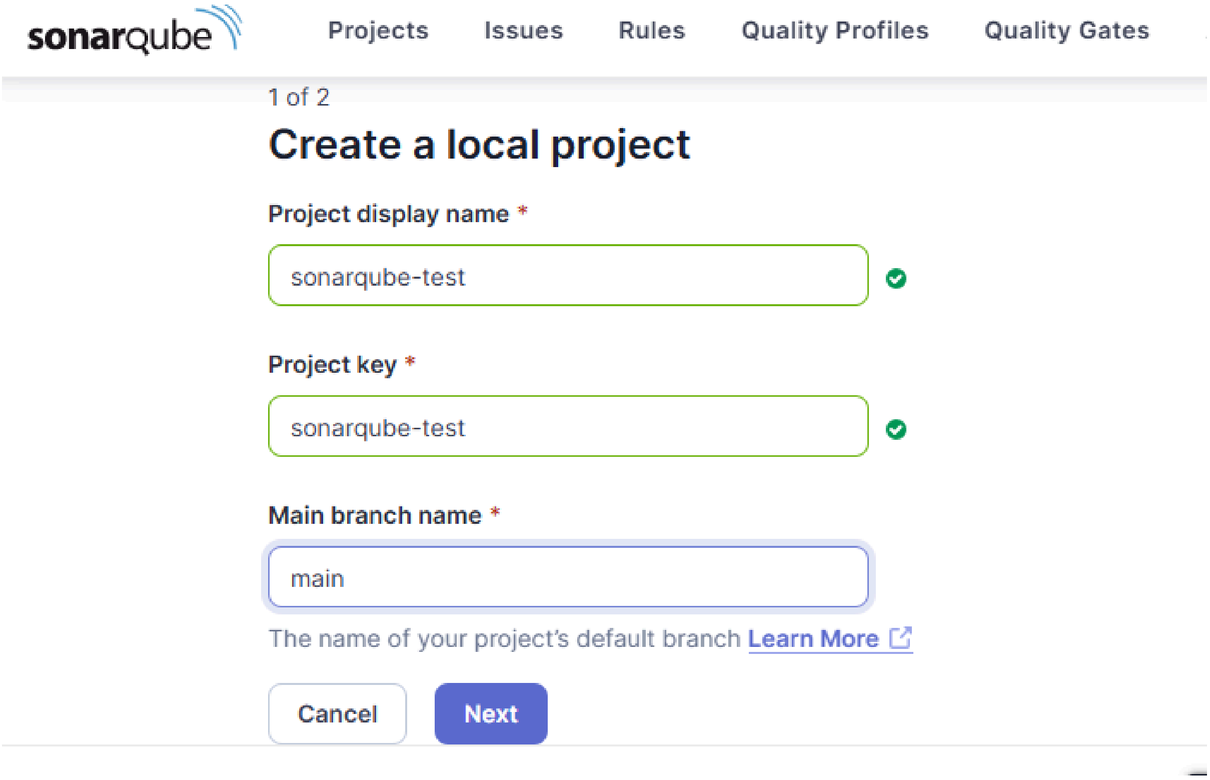
First, you need to set up a DevOps platform configuration.

Are you just testing or have an advanced use-case? 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.

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



1 of 2

Create a local project

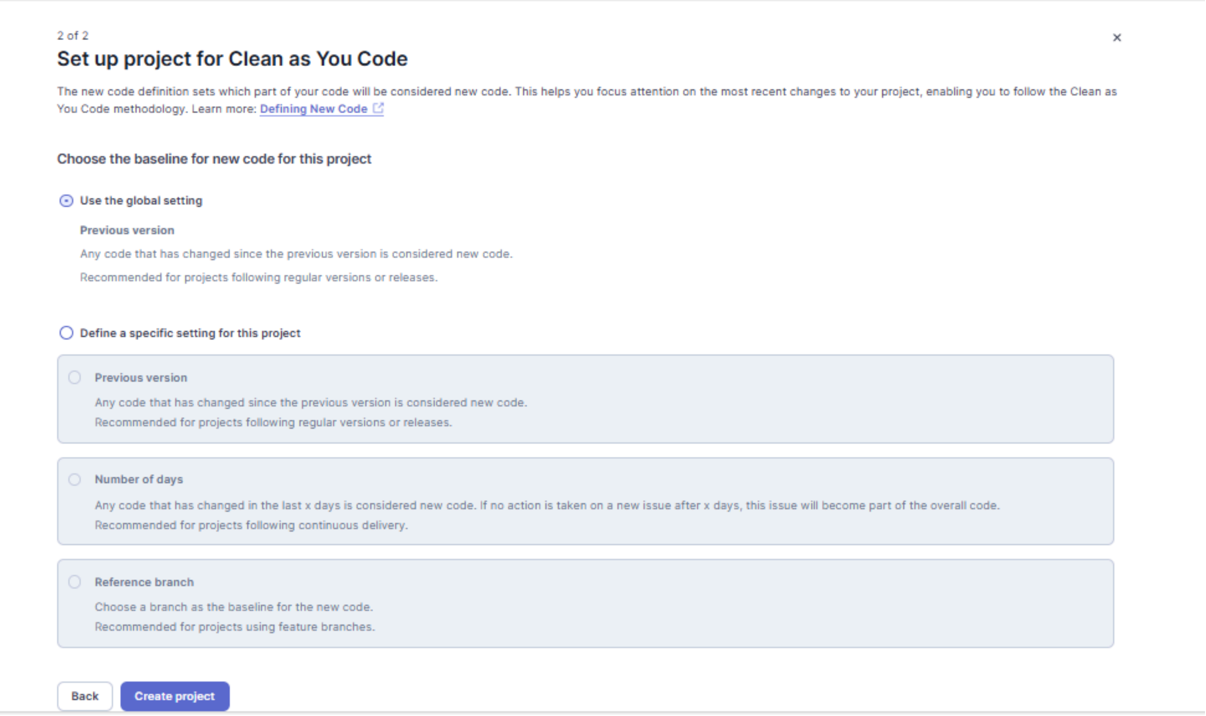
Project display name *: sonarqube-test

Project key *: sonarqube-test

Main branch name *: main

The name of your project's default branch Learn More

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



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.

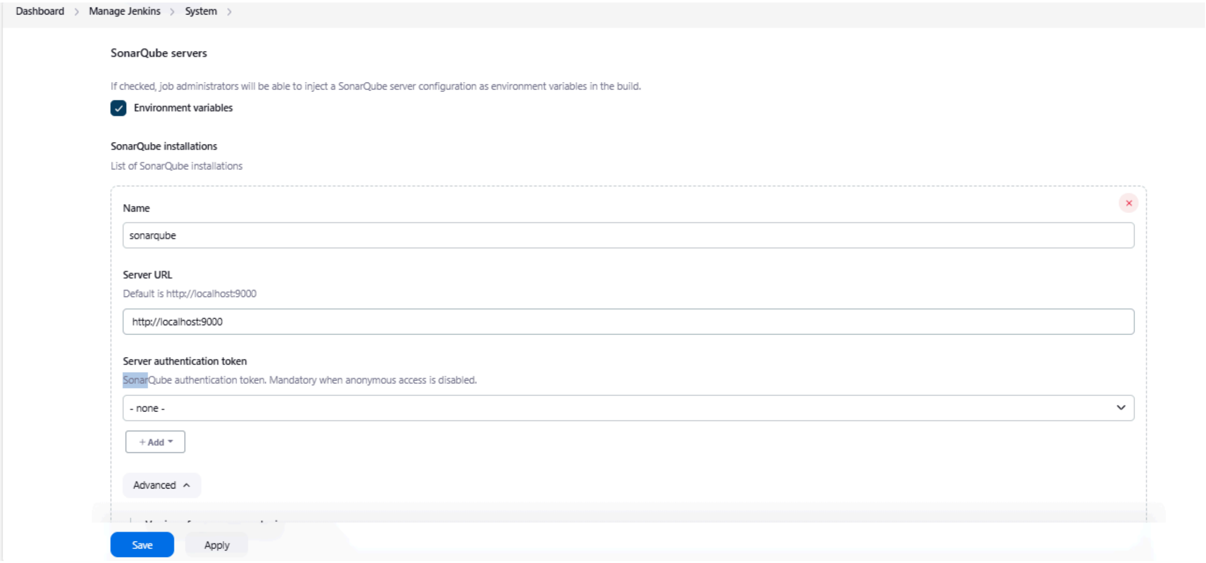
Recommended for projects following continuous delivery.

○ Reference branch

Choose a branch as the baseline for the new code.

Recommended for projects using feature branches.

STEP 11:Go to Manage Jenkins and then go to Systems and name to the environment variables then apply the changes and then save them.



SonarQube servers

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

☑ Environment variables

SonarQube installations

List of SonarQube installations

Name: sonarqube

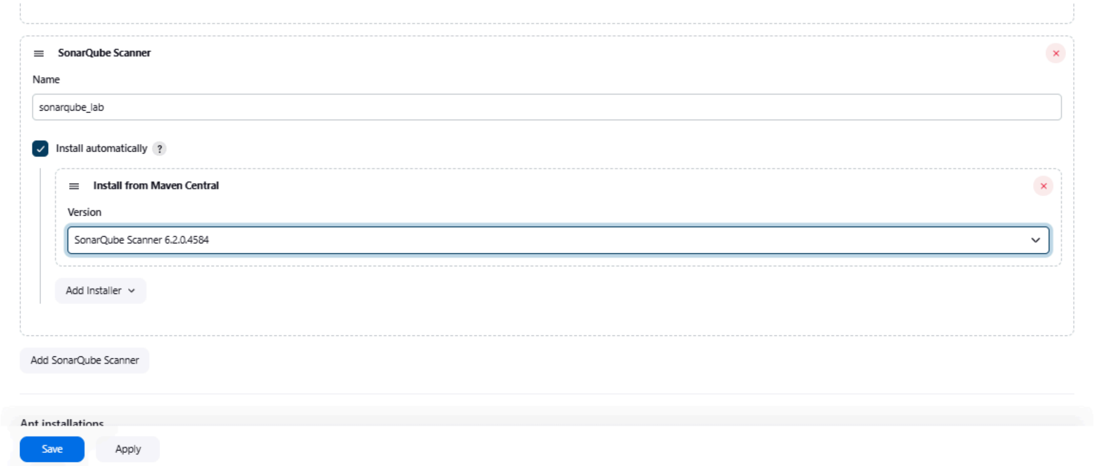
Server URL: Default is http://localhost:9000

http://localhost:9000

Server authentication token: SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -

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



SonarQube Scanner

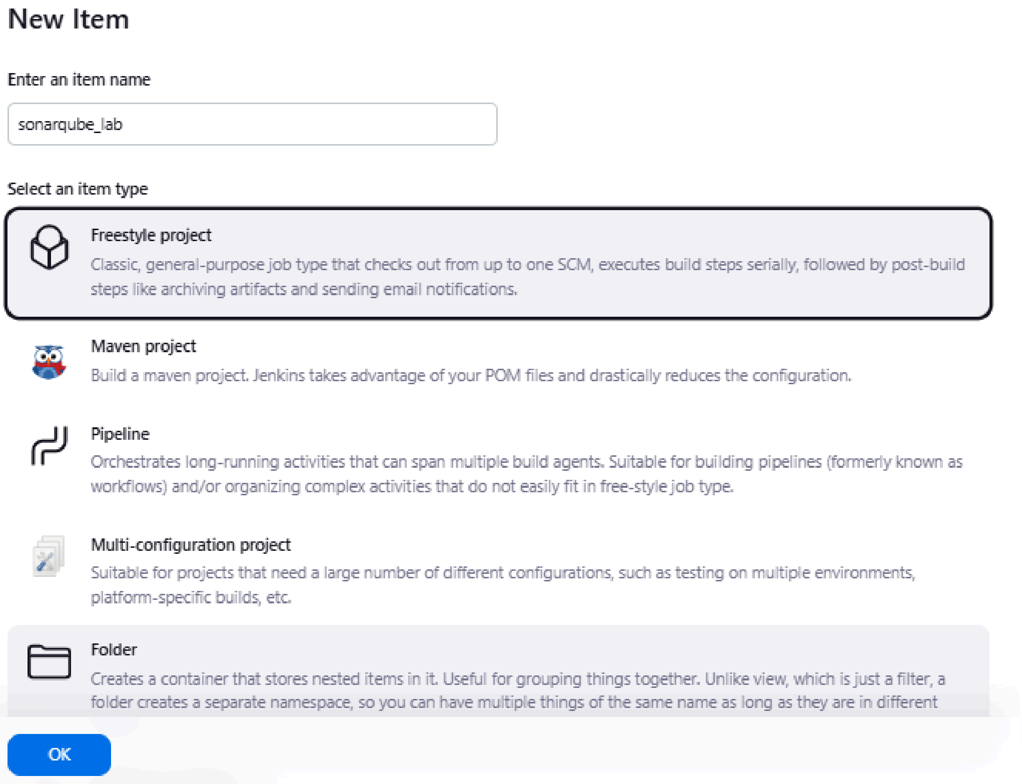
Name: sonarqube_lab

☑ Install automatically

Install from Maven Central

Version: SonarQube Scanner 6.2.0.4584

STEP 13: Go to Jenkins and then create a new item enter the item name and select an iten type to “Freestyle project” and then click on ok.



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.

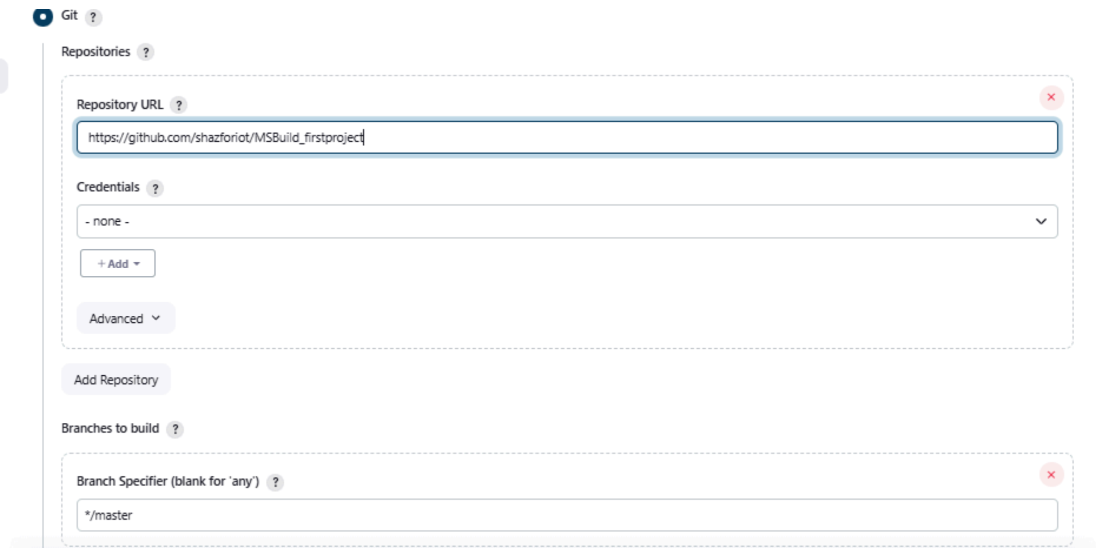
Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

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

STEP 14: Use this github repository in Source Code Management.
https://github.com/shazforiot/MSBuild_firstproject



● Git

Repositories

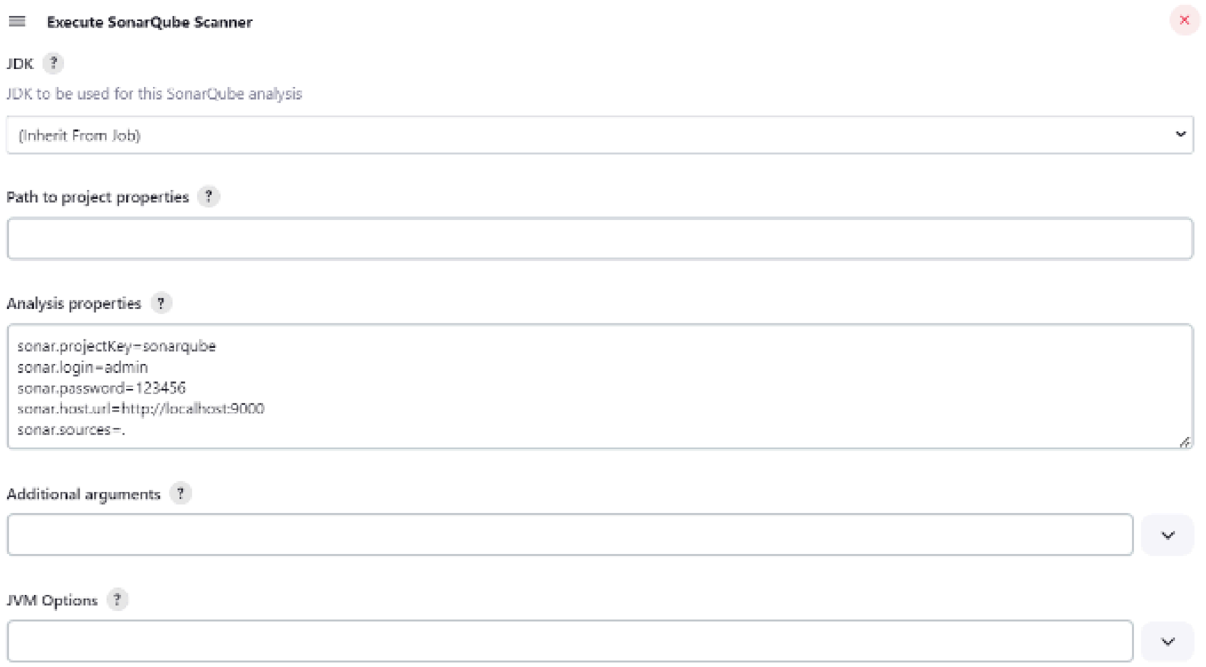
Repository URL
https://github.com/shazforiot/MSBuild_firstproject

Credentials
- none -

Branches to build

Branch Specifier (blank for 'any')
*/master

STEP 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 ?

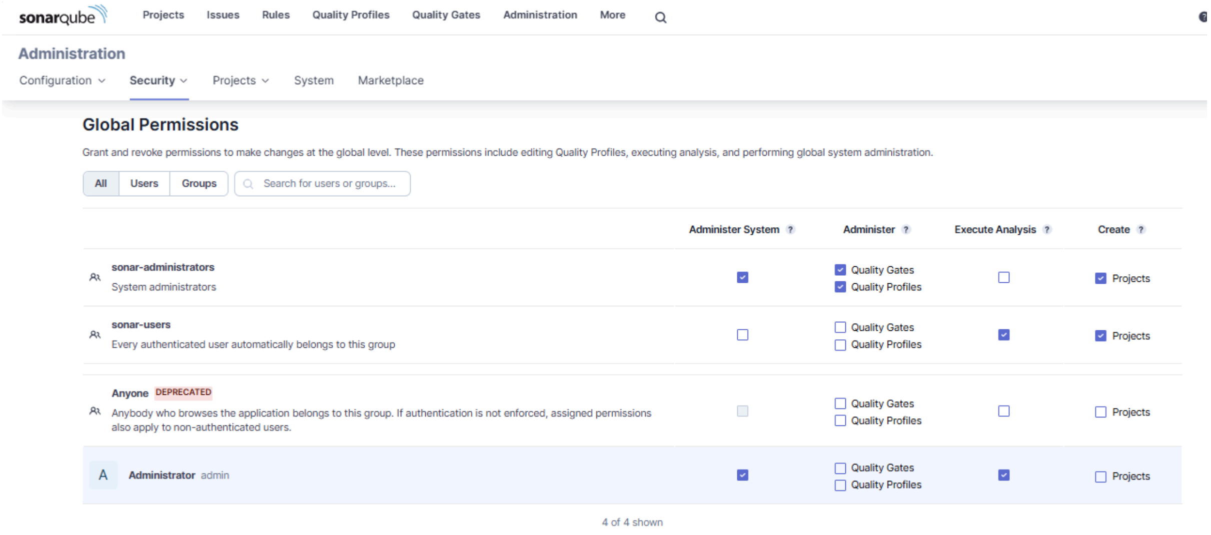
Analysis properties ?

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

Additional arguments ?

JVM Options ?

STEP 16: Now, you need to grant the local user (here admin user) permissions to Execute the Analysis stage on SoanrQube.
For this http://loaclhost:<port_number>/admin/permissions and check the ‘Execute Analysis’ checkbox under Administrator.



Global Permissions

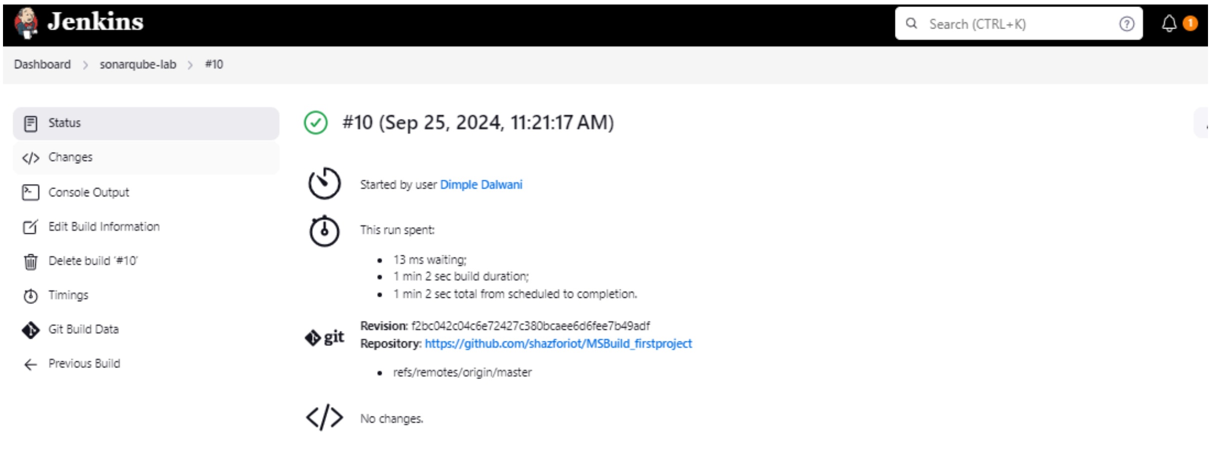
Grant and revoke permissions to make changes at the global level. These permissions include editing Quality Profiles, executing analysis, and performing global system administration.

All | Users | Groups | Search for users or groups...

	Administer System ?	Administer ?	Execute Analysis ?	Create ?
sonar-administrators System administrators	☑	☑ Quality Gates ☑ Quality Profiles	□	☑ Projects
sonar-users Every authenticated user automatically belongs to this group	□	□ Quality Gates □ Quality Profiles	☑	☑ Projects
Anyone DEPRECATED Anybody who browses the application belongs to this group. If authentication is not enforced, assigned permissions also apply to non-authenticated users.	□	□ Quality Gates □ Quality Profiles	□	□ Projects
A **Administrator** admin	☑	□ Quality Gates □ Quality Profiles	☑	□ Projects

4 of 4 shown

STEP 17: Go to the job you had just built and click on Build Now.



#10 (Sep 25, 2024, 11:21:17 AM)

Started by user Dimple Dalwani

This run spent:

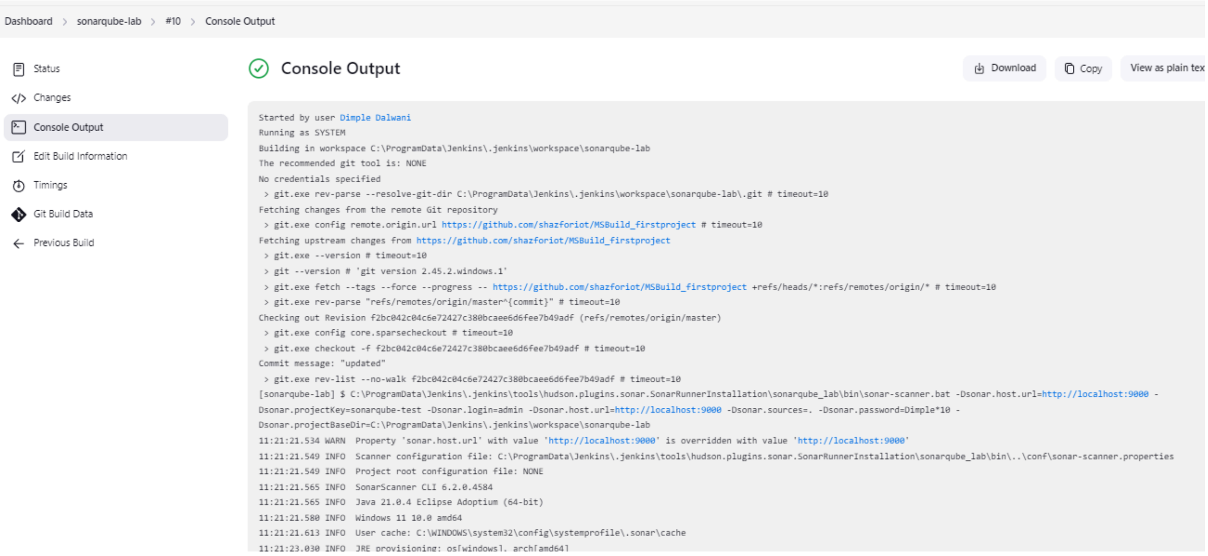
- 13 ms waiting;
- 1 min 2 sec build duration;
- 1 min 2 sec total from scheduled to completion.

Revision: f2bc042c04c6e72427c380bcaee6d6fee7b49adf
Repository: https://github.com/shazforiot/MSBuild_firstproject

- refs/remotes/origin/master

No changes.

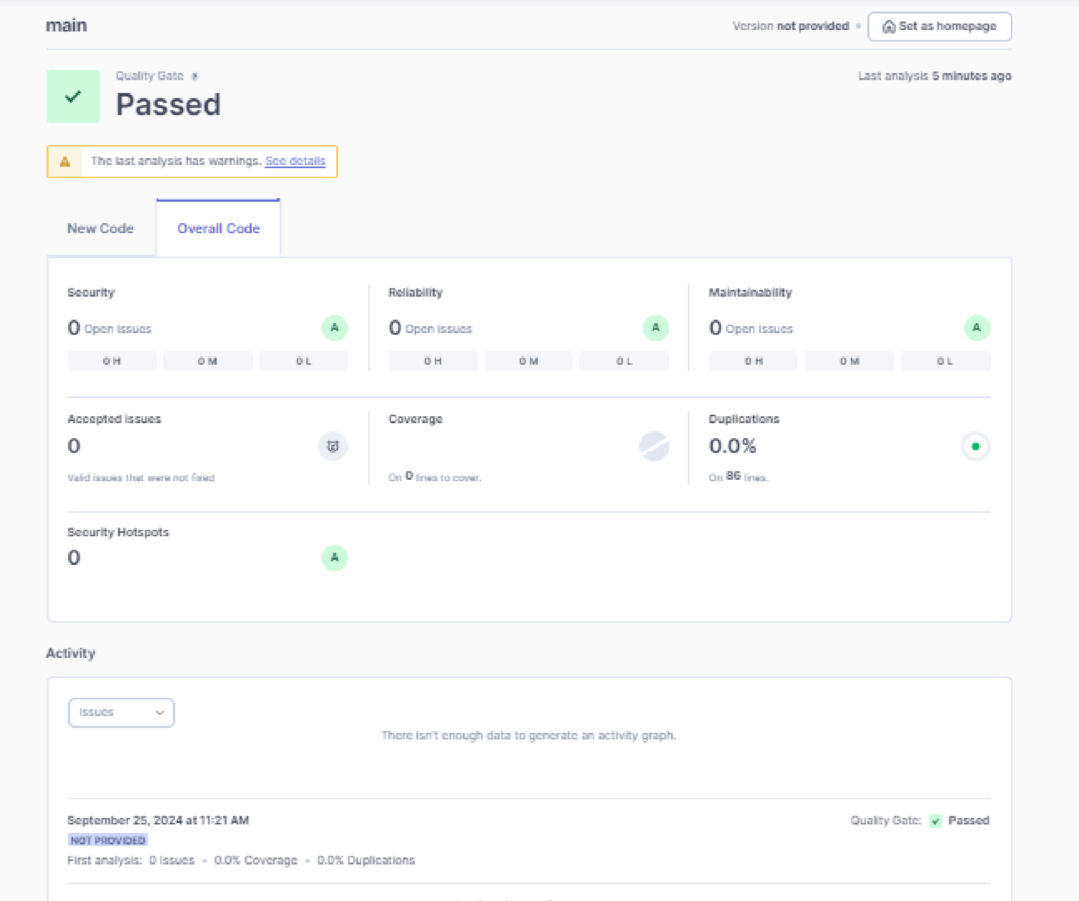
STEP 18: Check the console Output



Console Output

```
Started by user Dimple Dalwani
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\sonarqube-lab
The recommended git tool is: NONE
No credentials specified
 > git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\sonarqube-lab\.git # timeout=10
Fetching changes from the remote Git repository
 > git.exe config remote.origin.url https://github.com/shazforiot/MSBuild_firstproject # timeout=10
Fetching upstream changes from https://github.com/shazforiot/MSBuild_firstproject
 > git.exe --version # timeout=10
 > git --version # 'git version 2.45.2.windows.1'
 > git.exe fetch --tags --force --progress -- https://github.com/shazforiot/MSBuild_firstproject +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision f2bc042c04c6e72427c380bcaee6d6fee7b49adf (refs/remotes/origin/master)
 > git.exe config core.sparsecheckout # timeout=10
 > git.exe checkout -f f2bc042c04c6e72427c380bcaee6d6fee7b49adf # timeout=10
Commit message: "updated"
 > git.exe rev-list --no-walk f2bc042c04c6e72427c380bcaee6d6fee7b49adf # timeout=10
[sonarqube-lab] $ C:\ProgramData\Jenkins\.jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube_lab\bin\sonar-scanner.bat -Dsonar.host.url=http://localhost:9000 -
Dsonar.projectKey=sonarqube-test -Dsonar.login=admin -Dsonar.host.url=http://localhost:9000 -Dsonar.sources=. -Dsonar.password=Dimple*10 -
Dsonar.projectBaseDir=C:\ProgramData\Jenkins\.jenkins\workspace\sonarqube-lab
11:21:21.534 WARN  Property 'sonar.host.url' with value 'http://localhost:9000' is overridden with value 'http://localhost:9000'
11:21:21.549 INFO  Scanner configuration file: C:\ProgramData\Jenkins\.jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube_lab\bin\..\conf\sonar-scanner.properties
11:21:21.549 INFO  Project root configuration file: NONE
11:21:21.565 INFO  SonarScanner CLI 6.2.0.4584
11:21:21.565 INFO  Java 21.0.4 Eclipse Adoptium (64-bit)
11:21:21.580 INFO  Windows 11 10.0 amd64
11:21:21.613 INFO  User cache: C:\WINDOWS\system32\config\systemprofile\.sonar\cache
11:21:23.030 INFO  JRE provisioning: os[windows], arch[amd64]
```

STEP 19: Go back to SonarQube and check the project linked.



main

Version not provided

Quality Gate

Passed

Last analysis 5 minutes ago

The last analysis has warnings. See details

Security: 0 Open issues — A — 0 H, 0 M, 0 L

Reliability: 0 Open issues — A — 0 H, 0 M, 0 L

Maintainability: 0 Open issues — A — 0 H, 0 M, 0 L

Accepted issues: 0 — Valid issues that were not fixed

Coverage — On 0 lines to cover.

Duplications: 0.0% — On 86 lines.

Security Hotspots: 0 — A

Activity

There isn't enough data to generate an activity graph.

September 25, 2024 at 11:21 AM

NOT PROVIDED

First analysis: 0 Issues • 0.0% Coverage • 0.0% Duplications

Quality Gate: ✓ Passed

Conclusion: We successfully integrated Jenkins for Static Application Security Testing (SAST) with SonarQube. The process involved setting up Jenkins to trigger SAST scans and report results to SonarQube. We encountered issues with Jenkins job configurations and SonarQube plugin compatibility, which we resolved by updating plugins and refining job settings. This integration enhanced our code security by automating vulnerability detection and facilitating continuous improvement through real-time feedback in our CI/CD pipeline..