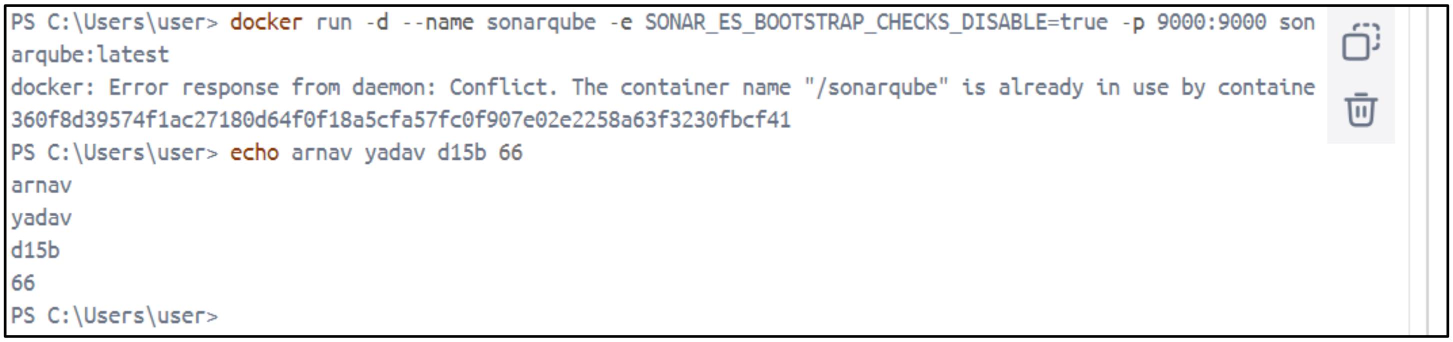
Experiment No 7

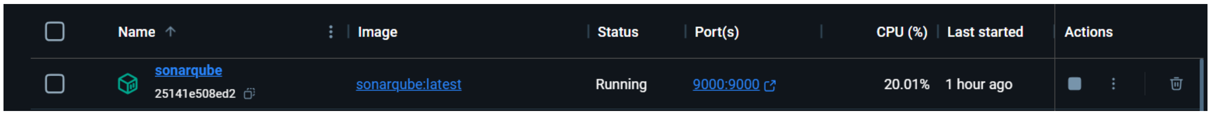
Aim: Installing SonarQube from the Docker Image

Theory:
SonarQube is an open-source platform for continuous inspection of code quality. It helps developers manage code quality and security by identifying bugs, vulnerabilities, and code smells. Running SonarQube in a Docker container simplifies the installation process, allowing for a quick setup and configuration. This experiment also integrates SonarQube with Jenkins, enabling automated code analysis in CI/CD pipelines.

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



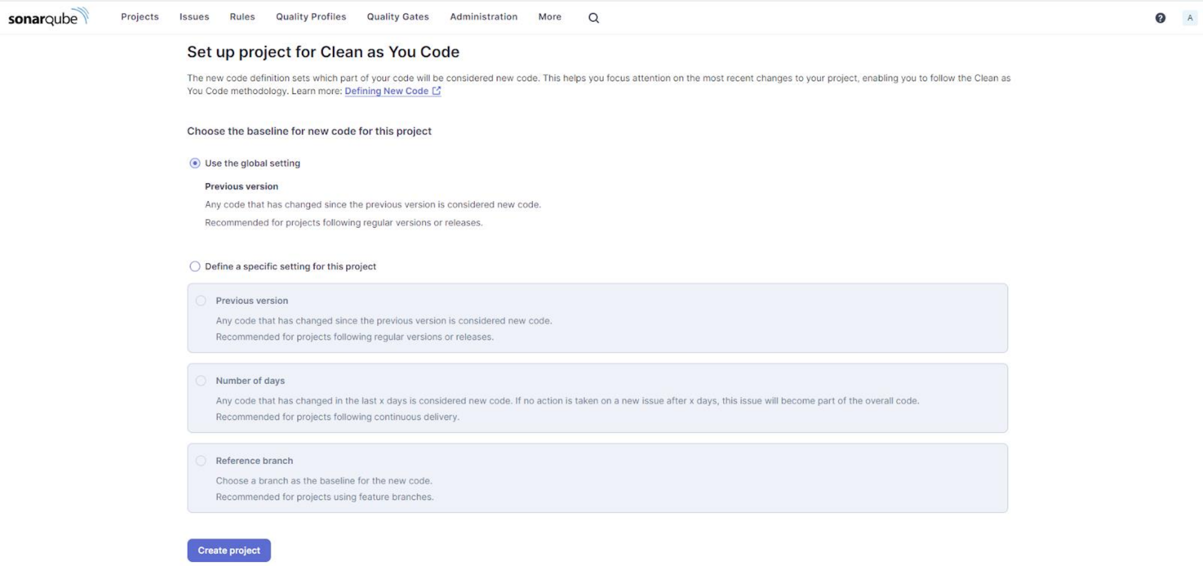
```
PS C:\Users\user> docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 son
arqube:latest
docker: Error response from daemon: Conflict. The container name "/sonarqube" is already in use by containe
360f8d39574f1ac27180d64f0f18a5cfa57fc0f907e02e2258a63f3230fbcf41
PS C:\Users\user> echo arnav yadav d15b 66
arnav
yadav
d15b
66
PS C:\Users\user>
```



□	Name	Image	Status	Port(s)	CPU (%)	Last started	Actions
□	sonarqube 25141e508ed2	sonarqube:latest	Running	9000:9000	20.01%	1 hour ago	

go to the SonarQube page by typing:
http://localhost:9000/ on your browser.

Create project manually:



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

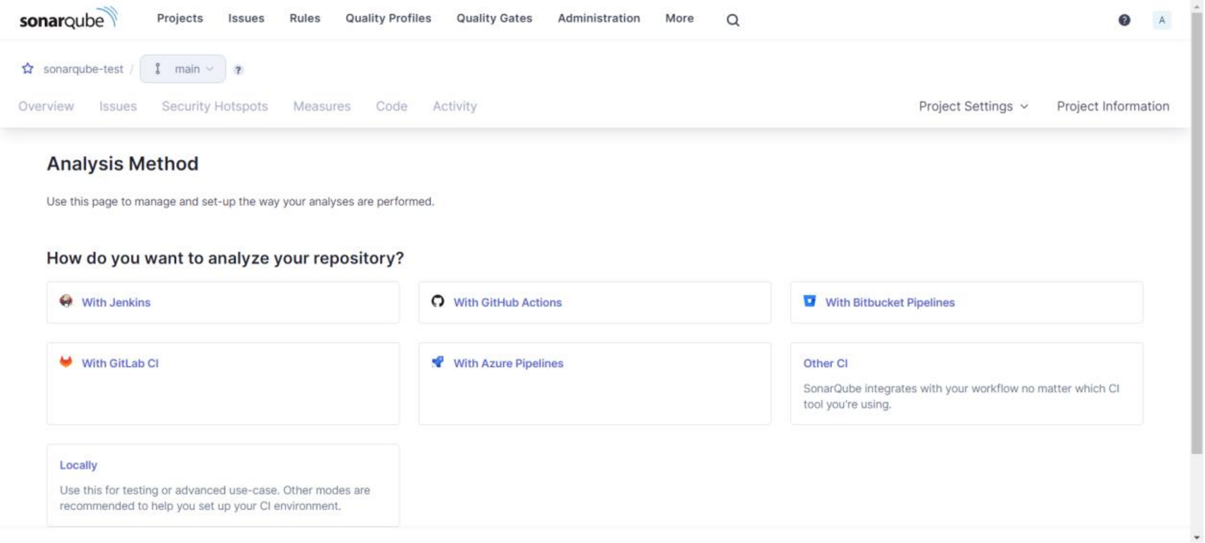
Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will become part of the overall code.

Recommended for projects following continuous delivery.

○ Reference branch

Choose a branch as the baseline for the new code.

Recommended for projects using feature branches.



Analysis Method

Use this page to manage and set-up the way your analyses are performed.

How do you want to analyze your repository?

With Jenkins

With GitHub Actions

With Bitbucket Pipelines

With GitLab CI

With Azure Pipelines

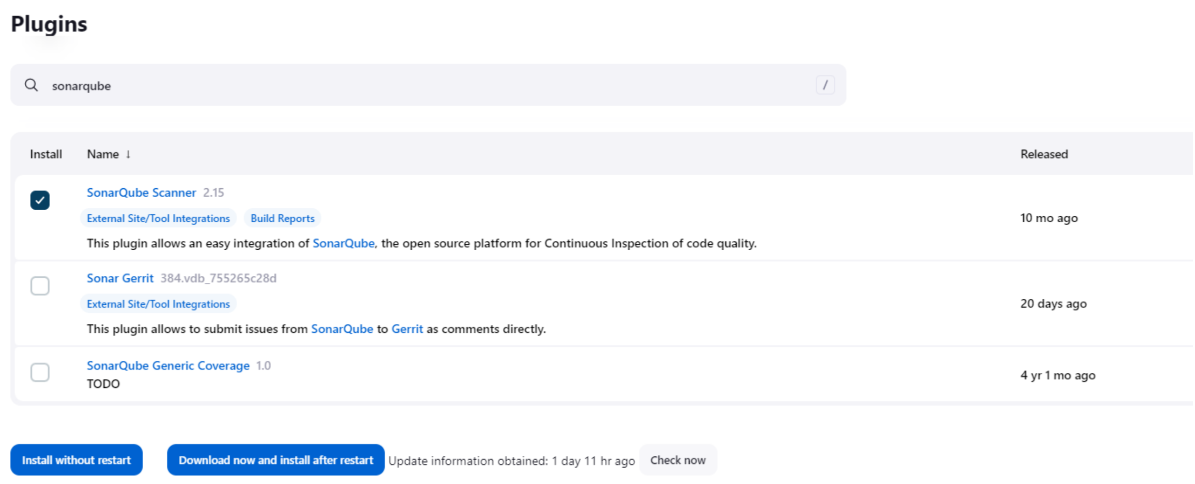
Other CI

SonarQube integrates with your workflow no matter which CI tool you're using.

Locally

Use this for testing or advanced use-case. Other modes are recommended to help you set up your CI environment.

open Jenkins
Go to Dashboard ->Manage Jenkins -> Plugin Manager and search for
SonarQube Scanner under Available plugins for Jenkins and install without restart.

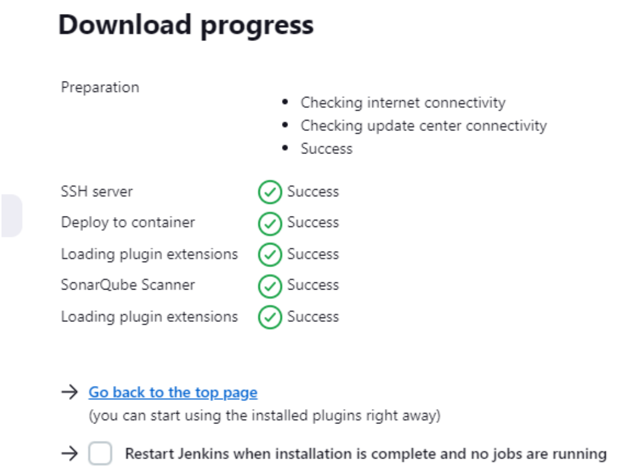


Plugins

sonarqube

Install	Name ↓	Released
☑	SonarQube Scanner 2.15 External Site/Tool Integrations · Build Reports This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.	10 mo ago
☐	Sonar Gerrit 384.vdb_755265c28d External Site/Tool Integrations This plugin allows to submit issues from SonarQube to Gerrit as comments directly.	20 days ago
☐	SonarQube Generic Coverage 1.0 TODO	4 yr 1 mo ago

Install without restart
Download now and install after restart
Update information obtained: 1 day 11 hr ago
Check now



Download progress

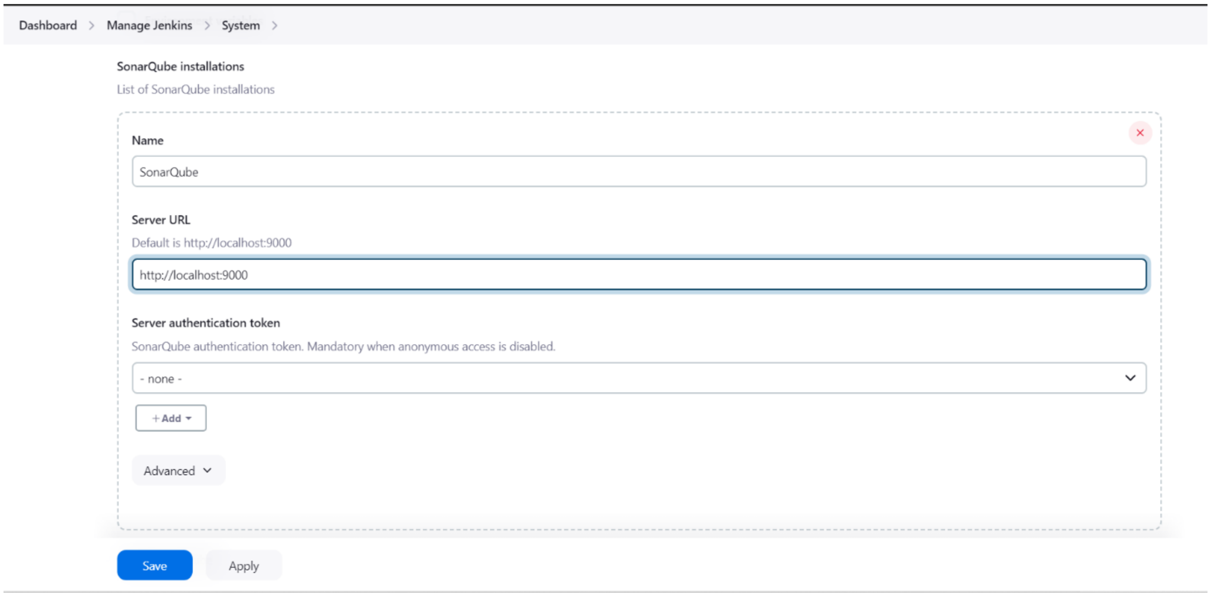
Preparation
- Checking internet connectivity
- Checking update center connectivity
- Success

SSH server	✓ Success
Deploy to container	✓ Success
Loading plugin extensions	✓ Success
SonarQube Scanner	✓ Success
Loading plugin extensions	✓ Success

→ Go back to the top page
(you can start using the installed plugins right away)

→ ☐ Restart Jenkins when installation is complete and no jobs are running

```
Under Jenkins ,
Dashboard -> Manage Jenkins -> Configure System ,
Look for SonarQube Servers and enter the details. Enter the Server Authentication
Token
if needed.
```



SonarQube installations

List of SonarQube installations

Name: SonarQube

Server URL

Default is http://localhost:9000

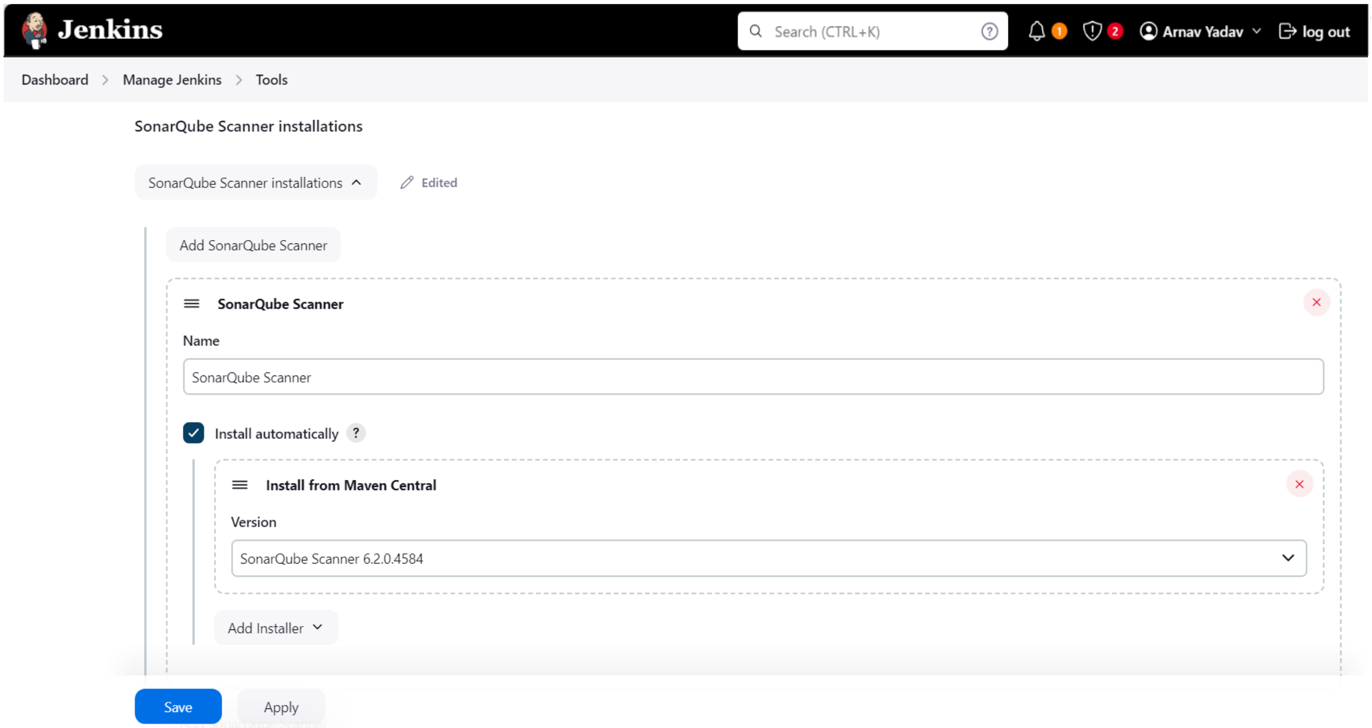
http://localhost:9000

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

- none -

Search SonarQube Scanner under Dashboard -> Manage Jenkins -> Global Tool Configuration.
Choose the latest configuration and choose Install Automatically.



SonarQube Scanner installations

SonarQube Scanner

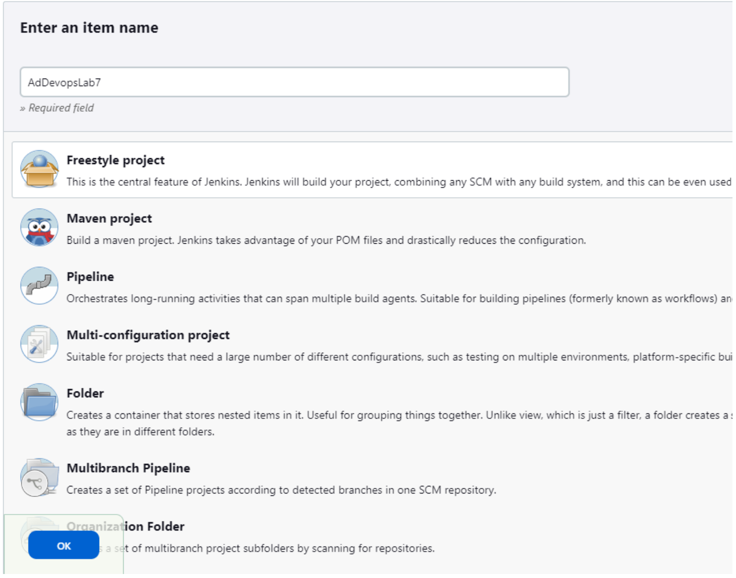
Name: SonarQube Scanner

☑ Install automatically

Install from Maven Central

Version: SonarQube Scanner 6.2.0.4584

create a New Item in Jenkins, choose a freestyle project.



Enter an item name

AdDevopsLab7

» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used

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) an

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

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
as they are in different folders.

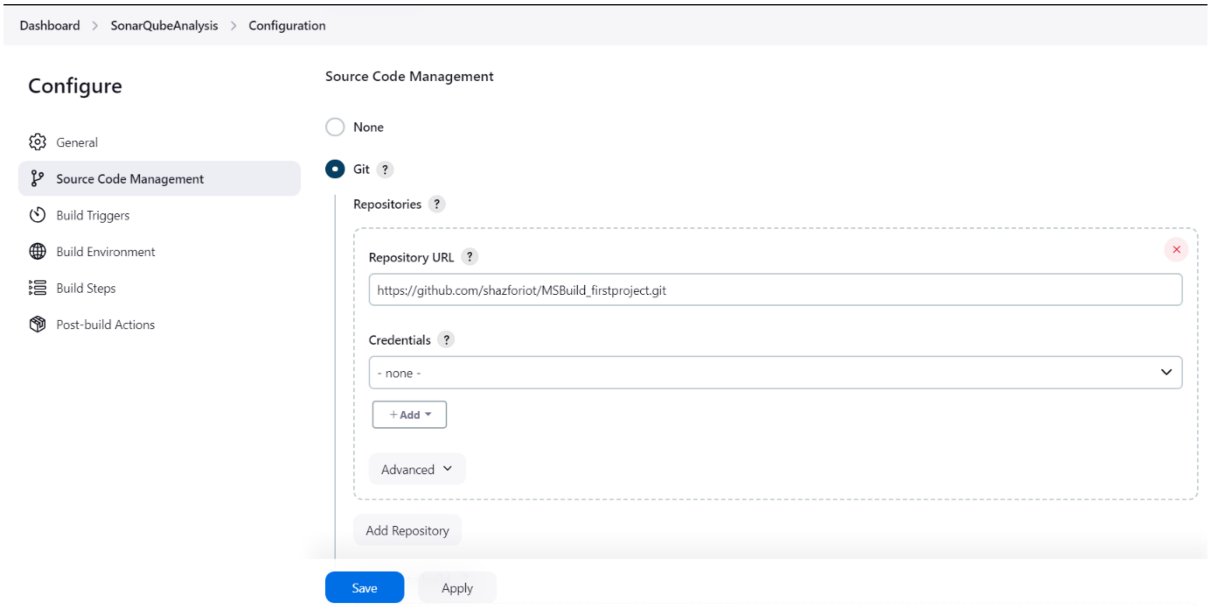
Multibranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.

Organization Folder
s a set of multibranch project subfolders by scanning for repositories.

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

Under Build ->Execute SonarQube Scanner, enter these Analysis properties. Mention the SonarQube Project Key, Login, Password, and Host URL.

```
sonar.projectKey=AdDevops
sonar.login=admin
sonar.password=abc
sonar.hosturl=http://localhost:9000/
```



Source Code Management

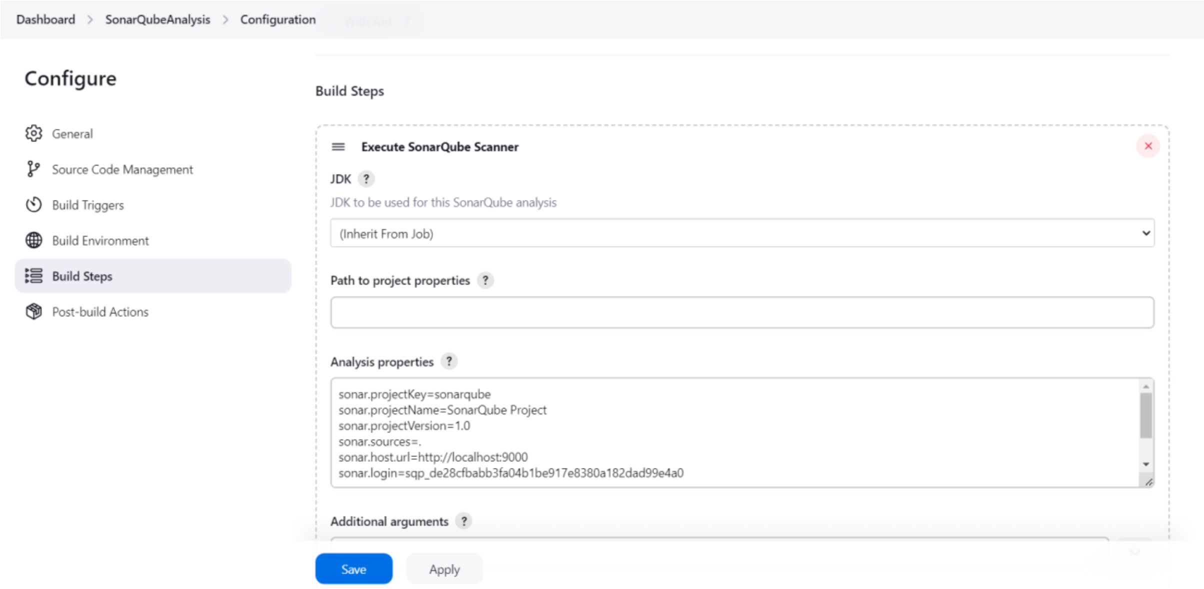
○ None

● Git

Repositories

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

Credentials: - none -



Build Steps

Execute SonarQube Scanner

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

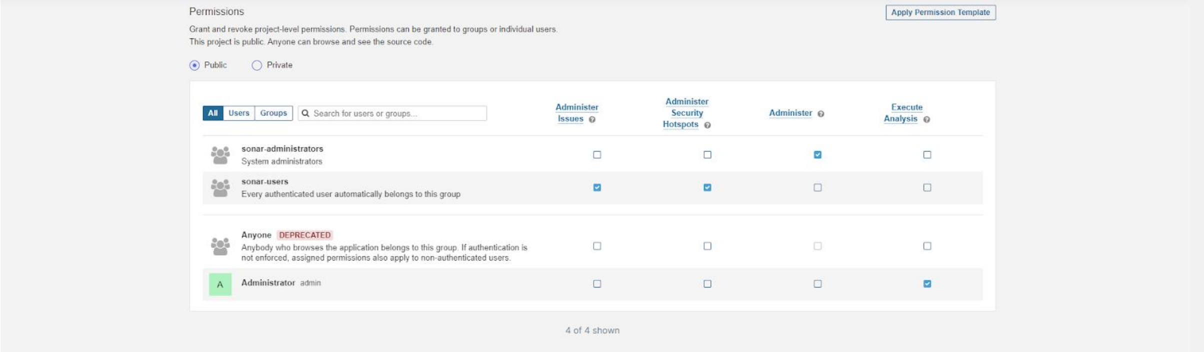
Path to project properties

Analysis properties

```
sonar.projectKey=sonarqube
sonar.projectName=SonarQube Project
sonar.projectVersion=1.0
sonar.sources=.
sonar.host.url=http://localhost:9000
sonar.login=sqp_de28cfbabb3fa04b1be917e8380a182dad99e4a0
```

Additional arguments

Go to http://localhost:9000/ and enter your previously created username.
Go to Permissions and grant the Admin user Execute Permissions.



Permissions

Grant and revoke project-level permissions. Permissions can be granted to groups or individual users.
This project is public. Anyone can browse and see the source code.

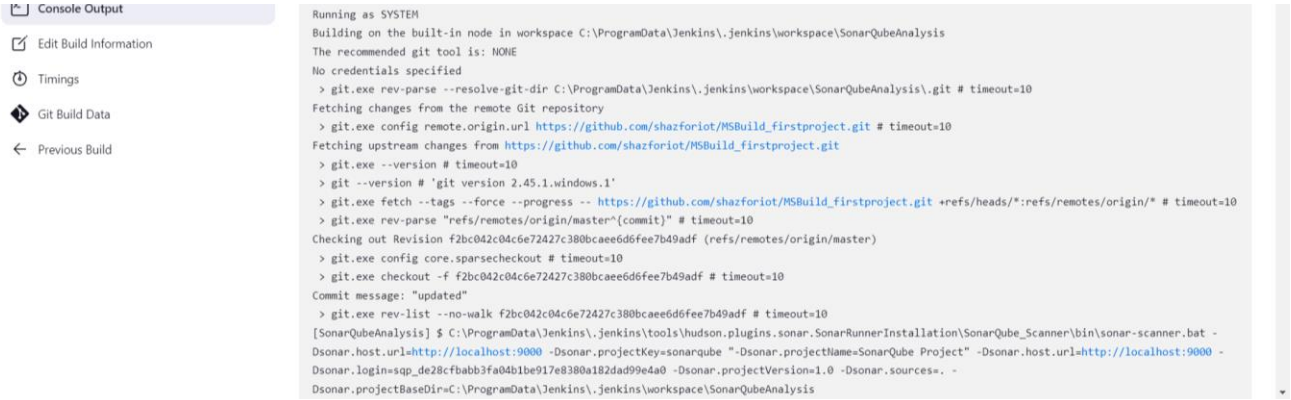
● Public ○ Private

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

4 of 4 shown

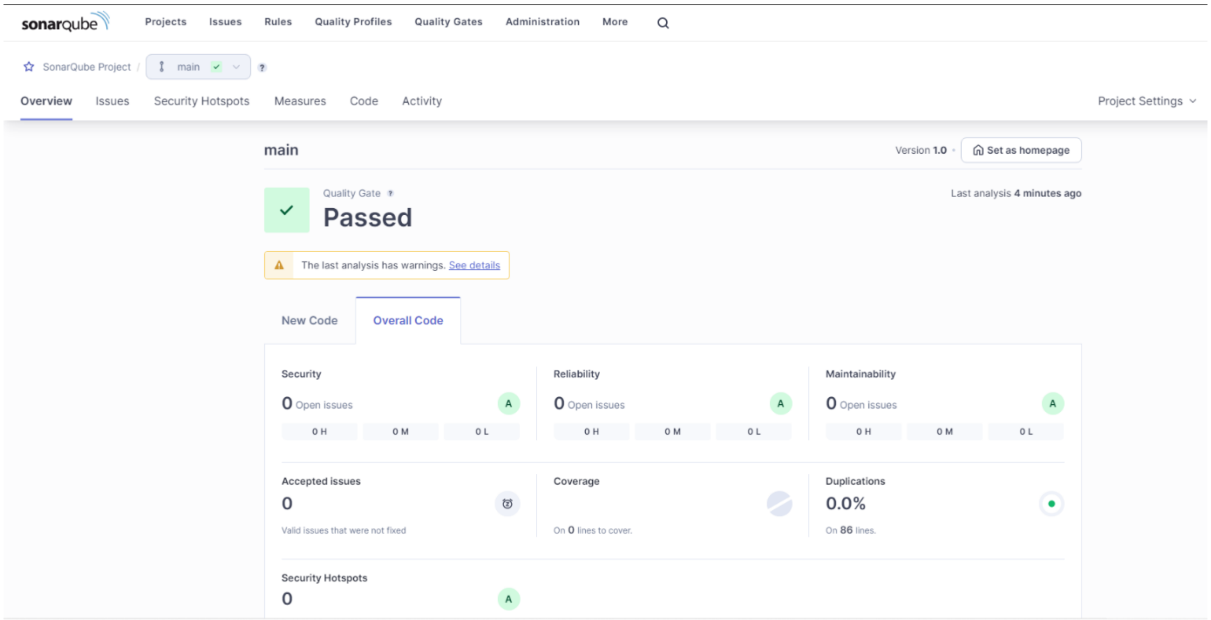
Build and Run:

Console Output:



```
Running as SYSTEM
Building on the built-in node in workspace C:\ProgramData\Jenkins\.jenkins\workspace\SonarQubeAnalysis
The recommended git tool is: NONE
No credentials specified
 > git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\SonarQubeAnalysis\.git # timeout=10
Fetching changes from the remote Git repository
 > git.exe config remote.origin.url https://github.com/shazforiot/MSBuild_firstproject.git # timeout=10
Fetching upstream changes from https://github.com/shazforiot/MSBuild_firstproject.git
 > git.exe --version # timeout=10
 > git --version # 'git version 2.45.1.windows.1'
 > git.exe fetch --tags --force --progress -- https://github.com/shazforiot/MSBuild_firstproject.git +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
[SonarQubeAnalysis] $ C:\ProgramData\Jenkins\.jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\SonarQube_Scanner\bin\sonar-scanner.bat -
Dsonar.host.url=http://localhost:9000 -Dsonar.projectKey=sonarqube "-Dsonar.projectName=SonarQube Project" -Dsonar.host.url=http://localhost:9000 -
Dsonar.login=sqp_de28cfbabb3fa04b1be917e8380a182dad99e4a0 -Dsonar.projectVersion=1.0 -Dsonar.sources=. -
Dsonar.projectBaseDir=C:\ProgramData\Jenkins\.jenkins\workspace\SonarQubeAnalysis
```

Project on sonarqube:



main

Version 1.0

Quality Gate

Passed

Last analysis 4 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

Conclusion: Thus, we have successfully installed SonarQube from Docker image. Running SonarQube in a Docker container simplifies the installation process, allowing for a quick setup and configuration. This experiment also integrates SonarQube with Jenkins, enabling automated code analysis in CI/CD pipelines.