

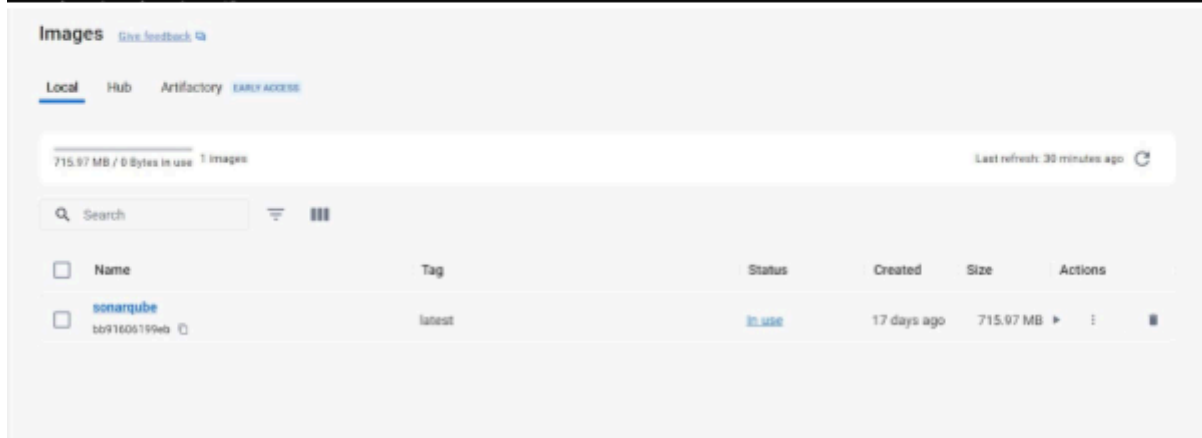
Experiment No 7

AIM: Installing SonarQube from the Docker Image

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

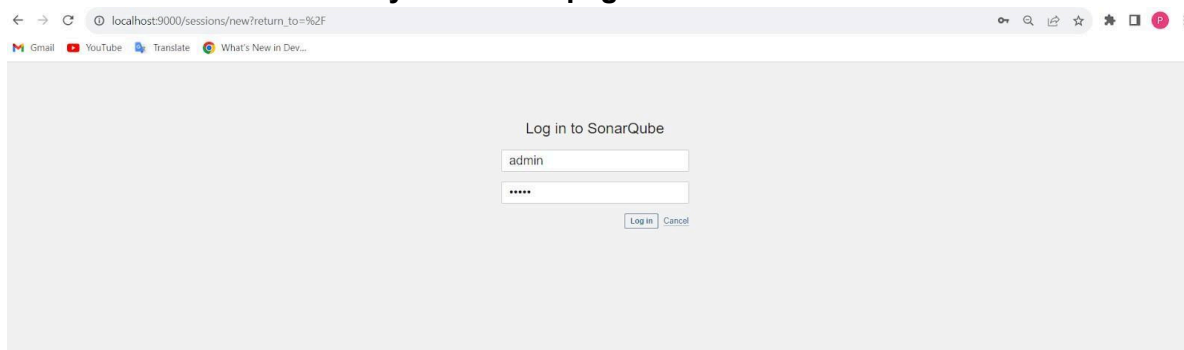
Sonarqube:latest

```
PS D:\Desktop\DockeFile> 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
44ba2882f8eb: Pull complete
2cabec57fa36: Pull complete
c20481384b6a: Pull complete
bf7b17ee74f8: Pull complete
38617faac714: Pull complete
b795b715553d: Pull complete
c5244f6c9231: Pull complete
Digest: sha256:1fffd122cfb37ce982289dc7f5d38bb702ba85af7b5a50f7cb077ae25e60b5b9a
Status: Downloaded newer image for sonarqube:latest
1442c4e613b25aaedec05c060f020a00802b1c6dbaa27e8c5c0dad4ed8fc1f76
```

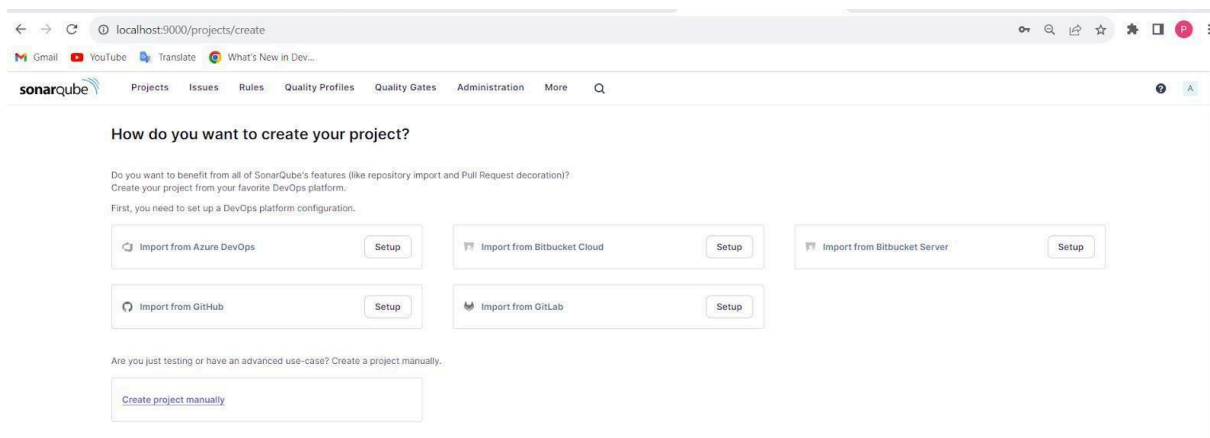


go to the SonarQube page by typing:
<http://localhost:9000/> on your browser.

Installation is successful if you see this page



Update to new password



localhost:5000/projects/create

sonarqube

Projects Issues Rules Quality Profiles Quality Gates Administration More

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 DevOps Setup

Import from Bitbucket Cloud Setup

Import from Bitbucket Server Setup

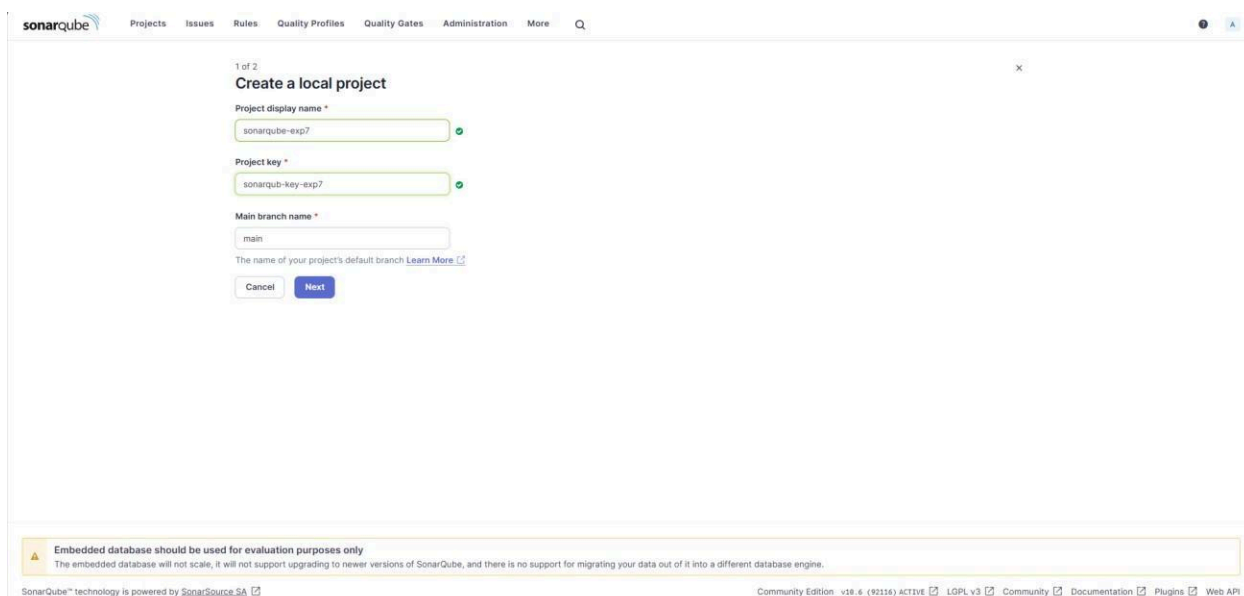
Import from GitHub Setup

Import from GitLab Setup

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

Create project manually

Create project manually: Here project name is "AdDevops"



sonarqube

Projects Issues Rules Quality Profiles Quality Gates Administration More

1 of 2 Create a local project

Project display name *
sonarqube-exp7

Project key *
sonarqub-key-exp7

Main branch name *
main

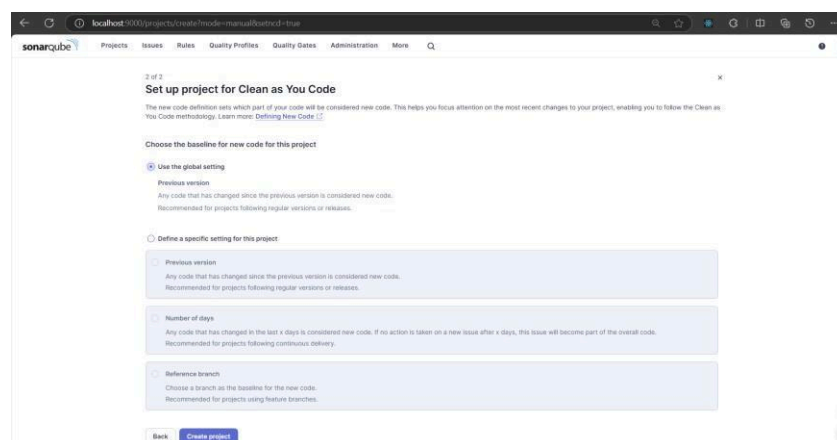
The name of your project's default branch [Learn More](#)

Cancel Next

Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

SonarQube™ technology is powered by [SonarSource SA](#)

Community Edition v10.6 (92116) [ACTIVE](#) [LGPL v3](#) [Community](#) [Documentation](#) [Plugins](#) [Web API](#)



localhost:5000/projects/create/branch-mode-manual/branch-mode

sonarqube

Projects Issues Rules Quality Profiles Quality Gates Administration More

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

open Jenkins

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

Plugins

Install	Name	Released
<input checked="" type="checkbox"/>	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
<input type="checkbox"/>	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
<input type="checkbox"/>	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](#)

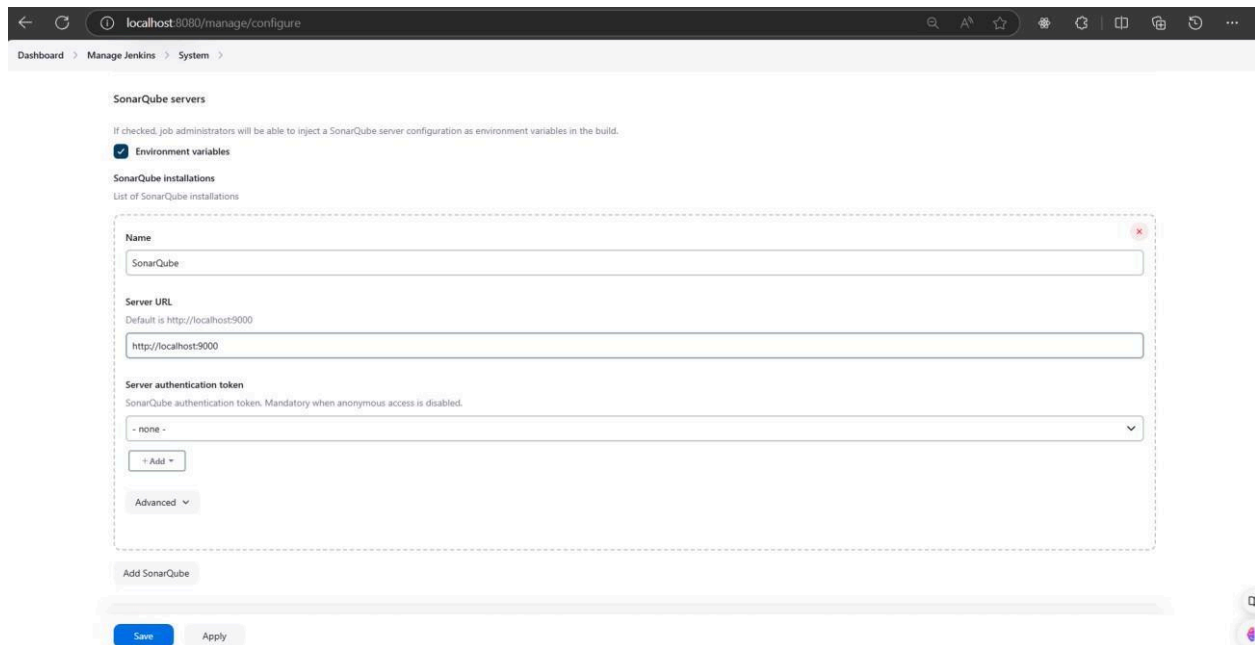
(you can start using the installed plugins right away)

→ ☐ 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.



The screenshot shows the 'SonarQube servers' configuration page in Jenkins. The breadcrumb trail is 'Dashboard > Manage Jenkins > System'. A note states: 'If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.' The 'Environment variables' checkbox is checked. Under 'SonarQube installations', there is a list of installations. One installation is shown with the following details: Name: SonarQube, Server URL: http://localhost:9000, and Server authentication token: none. There is an 'Add' button and an 'Advanced' dropdown. At the bottom, there is an 'Add SonarQube' button and 'Save' and 'Apply' buttons.

Dashboard > Manage Jenkins > System

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 -

+ Add

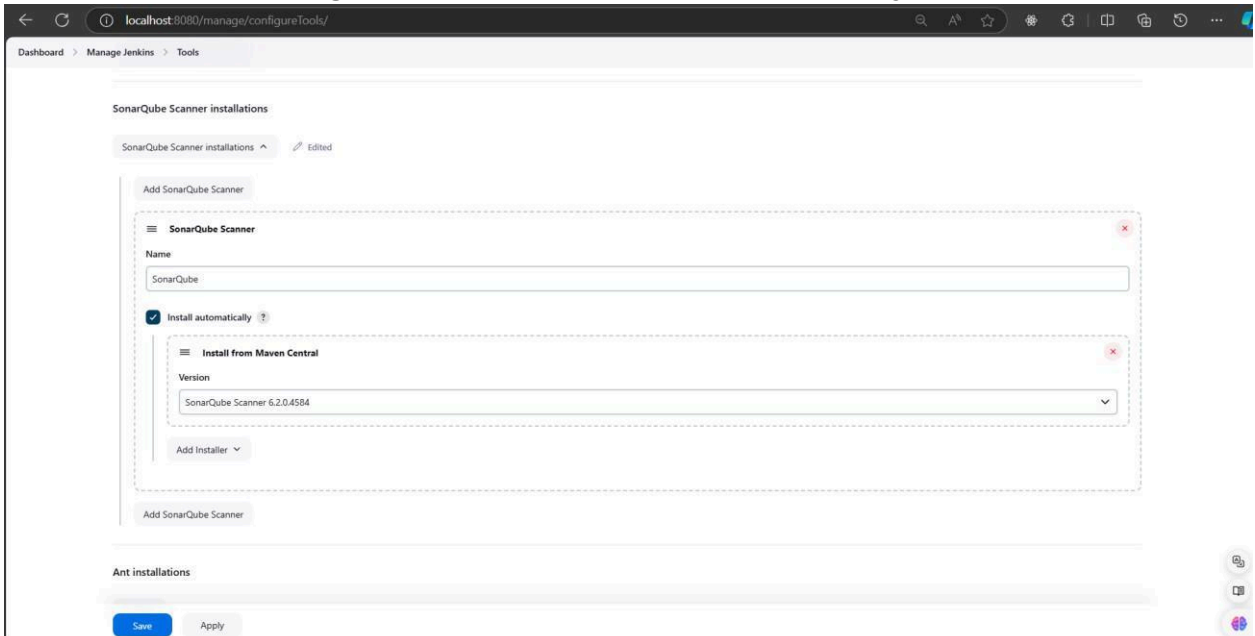
Advanced

Add SonarQube

Save Apply

Search SonarQube Scanner under Dashboard -> Manage Jenkins -> Global Tool Configuration.

Choose the latest configuration and choose Install Automatically.



The screenshot shows the 'SonarQube Scanner installations' configuration page in Jenkins. The breadcrumb trail is 'Dashboard > Manage Jenkins > Tools'. There is a section for 'SonarQube Scanner installations' with an 'Add SonarQube Scanner' button. A configuration is shown with the following details: Name: SonarQube, Install automatically: checked, and Install from Maven Central: SonarQube Scanner 6.2.0.4584. There is an 'Add Installer' button. At the bottom, there is an 'Add SonarQube Scanner' button and 'Save' and 'Apply' buttons.

Dashboard > Manage Jenkins > Tools

SonarQube Scanner installations

SonarQube Scanner installations ^ Edited

Add SonarQube Scanner

SonarQube Scanner

Name

SonarQube

☒ Install automatically ?

Install from Maven Central

Version

SonarQube Scanner 6.2.0.4584

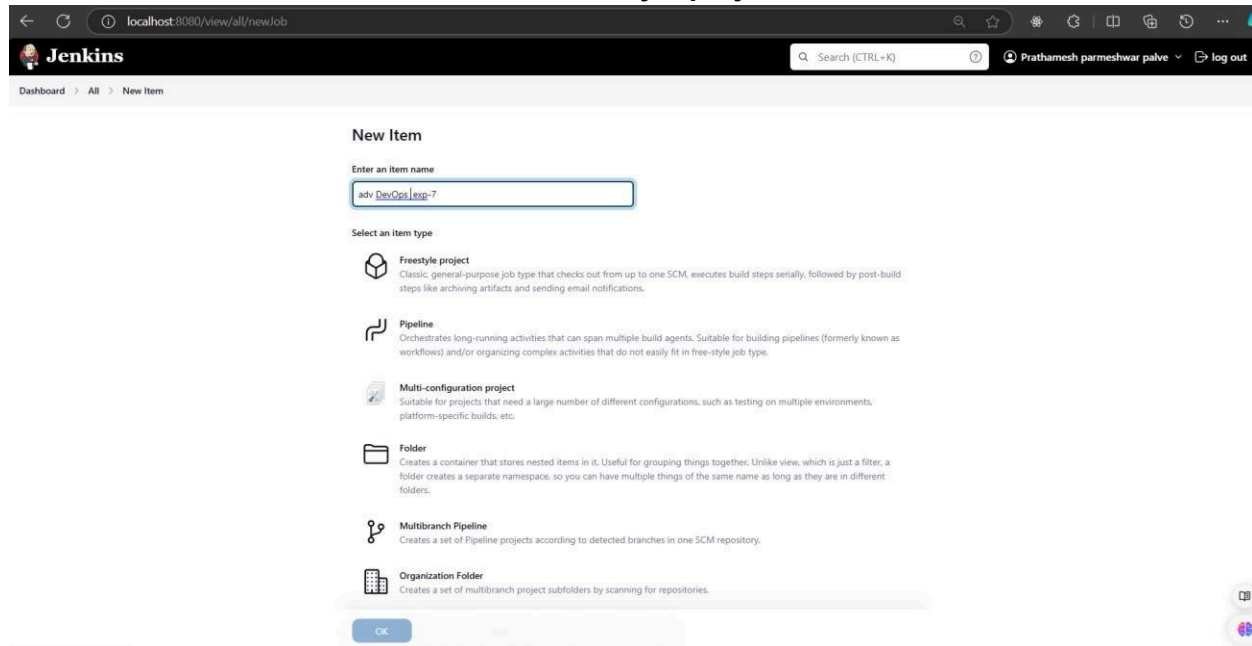
Add Installer

Add SonarQube Scanner

Ant installations

Save Apply

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



localhost:8080/view/all/newJob

Jenkins

Search (CTRL+K)

Prathamesh parmeshwar palve log out

Dashboard > All > New Item

New Item

Enter an item name

adv DevOps exp-7

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.
- 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 folders.
- Multibranch Pipeline**
Creates a set of Pipeline projects according to detected branches in one SCM repository.
- Organization Folder**
Creates a set of multibranch project subfolders by scanning for repositories.

OK

Choose this GitHub repository in Source Code Management.

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

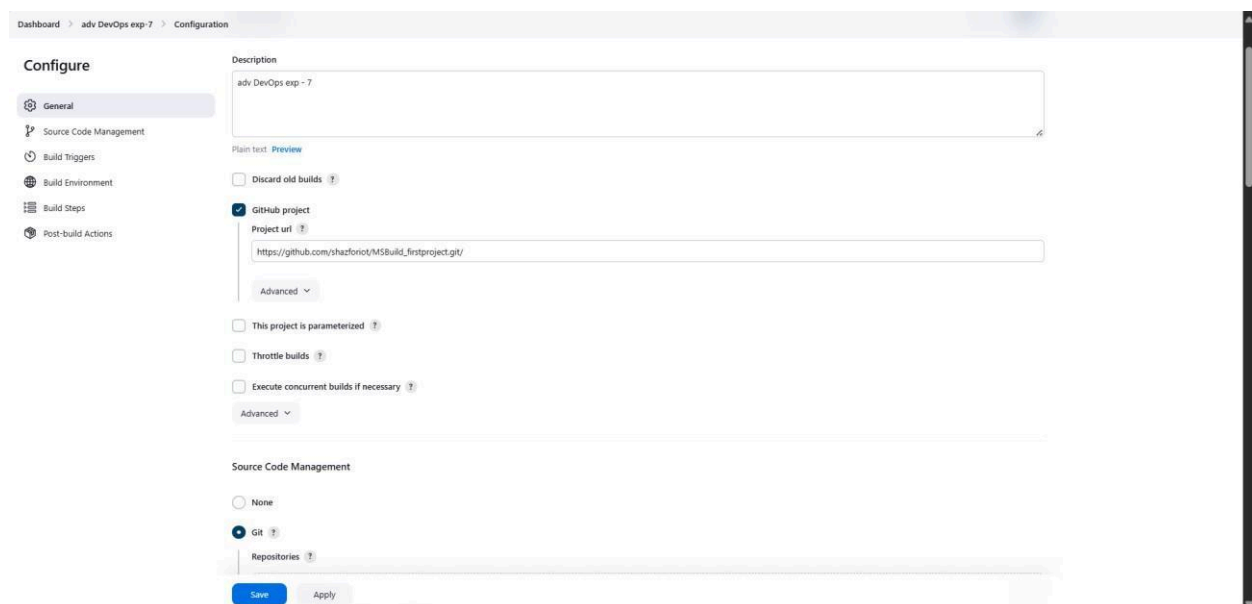
Under Build -> Execute SonarQube Scanner, enter these Analysis properties. Mention the SonarQube Project Key, Login, Password, and Host URL.

sonar.projectKey=SonarQueue-key

sonar.login=admin

sonar.password=admin

sonar.hosturl=<http://localhost:9000/>



Dashboard > adv DevOps exp-7 > Configuration

Configure

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions

Description

adv DevOps exp-7

Plain text Preview

☐ Discard old builds ?

☒ **GitHub project**

Project url ?

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

Advanced ▾

☐ This project is parameterized ?

☐ Throttle builds ?

☐ Execute concurrent builds if necessary ?

Advanced ▾

Source Code Management

☐ None

☒ **Git ?**

Repositories ?

Save Apply

The screenshot shows the 'Configure' page for a build step in SonarQube. The left sidebar contains a 'Configure' section with a 'Build Steps' tab selected. The main area is titled 'SonarQube Installation' and contains several configuration fields:

- SonarQube Installation:** A dropdown menu with 'SonarQube-server' selected.
- JDK:** A dropdown menu with '(inherit From Job)' selected.
- SonarQube Scanner:** A dropdown menu with 'SonarQube' selected.
- Path to project properties:** An empty text input field.
- Analysis properties:** A text area containing the following properties:

```
sonar.projectkey=sonarqube-key-exp?
sonar.projectname=sonarqube-exp?
sonar.host.url=http://localhost:9000
sonar.login=admin
sonar.password=3h4lve@08
```
- Additional arguments:** An empty text input field.
- JVM Options:** An empty text input field.

At the bottom of the configuration area, there is a 'Save' button and an 'Apply' button.

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

The screenshot shows the 'Administration' page in SonarQube, specifically the 'Global Permissions' section. The page has a navigation bar with 'Administration' selected. Below the navigation bar, there is a 'Global Permissions' section with a description: 'Grant and revoke permissions to make changes at the global level. These permissions include editing Quality Profiles, executing analysis, and performing global system administration.'

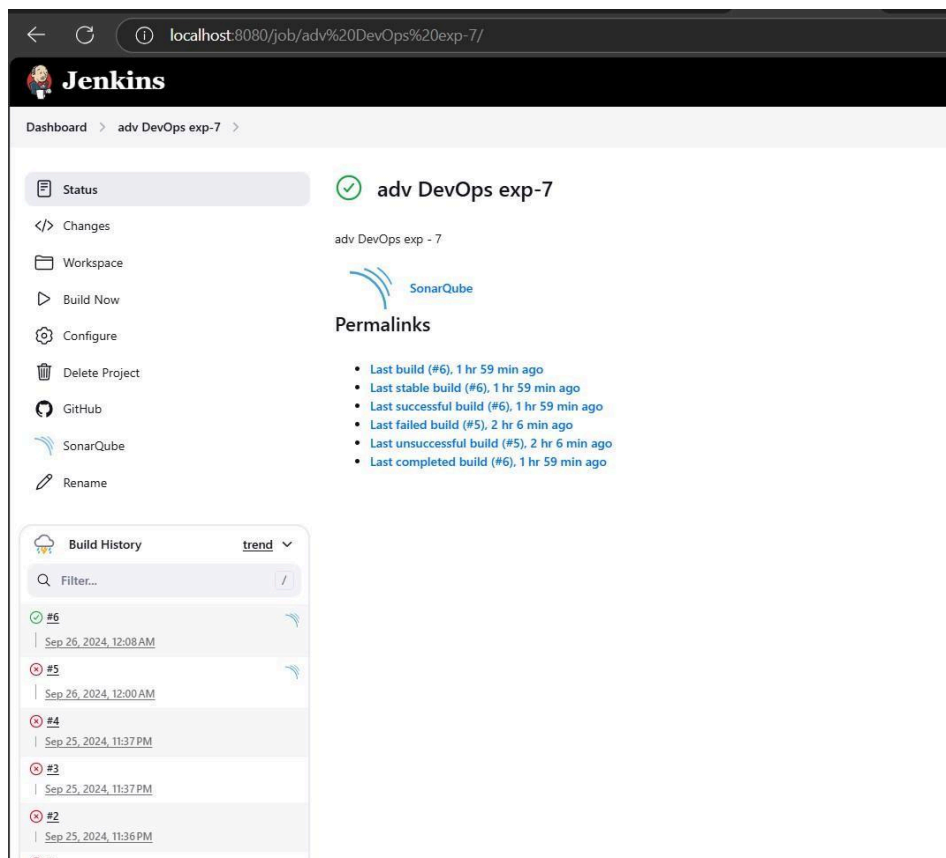
There are three tabs: 'All', 'Users', and 'Groups'. The 'Users' tab is selected. Below the tabs, there is a search bar labeled 'Search for users or groups...'. Below the search bar, there is a table with the following columns: 'User', 'Administer System', 'Administer', 'Execute Analysis', and 'Create'.

User	Administer System	Administer	Execute Analysis	Create
sonar-administrators 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
sonar-users 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
Anyone DEPRECATED 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
Administrator admin	<input type="checkbox"/>	<input type="checkbox"/> Quality Gates <input type="checkbox"/> Quality Profiles	<input checked="" type="checkbox"/>	<input type="checkbox"/> Projects

At the bottom of the table, there is a link '4 of 4 shown'.

At the bottom of the page, there is a warning message: 'Embedded database should be used for evaluation purposes only. The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.'

Build and Run:



The screenshot shows the Jenkins Dashboard for the job 'adv DevOps exp-7'. The left sidebar contains a 'Status' tab and a list of actions: Changes, Workspace, Build Now, Configure, Delete Project, GitHub, SonarQube, and Rename. The main content area displays the job's status as 'adv DevOps exp-7' with a green checkmark. Below this, there is a 'SonarQube' icon and a 'Permalinks' section listing various build links. At the bottom, a 'Build History' table shows a list of builds with their status, number, and timestamps.

Dashboard > adv DevOps exp-7

Status

adv DevOps exp-7

adv DevOps exp - 7

SonarQube

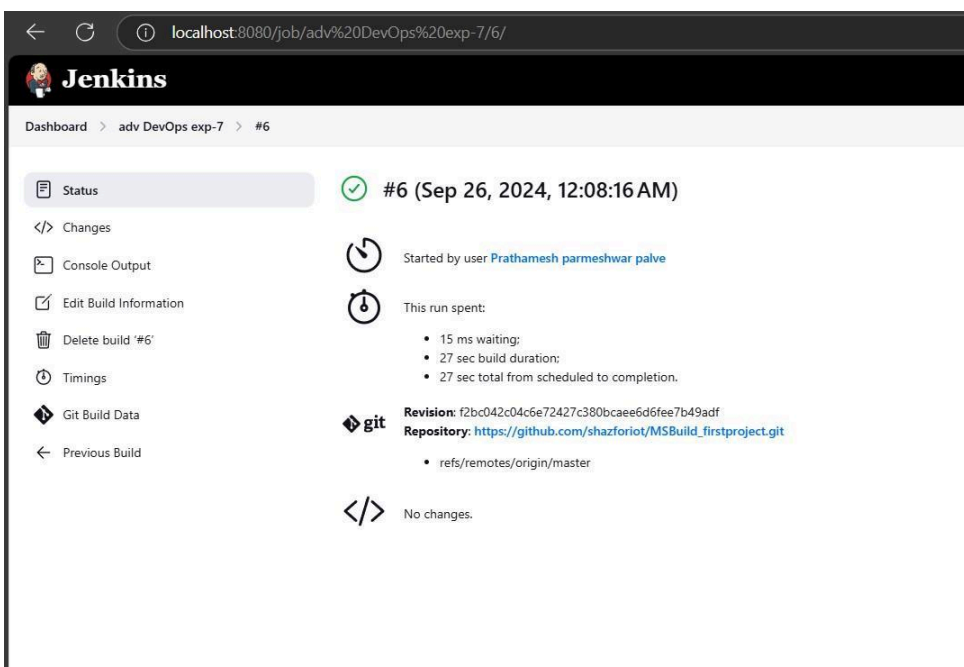
Permalinks

- Last build (#6), 1 hr 59 min ago
- Last stable build (#6), 1 hr 59 min ago
- Last successful build (#6), 1 hr 59 min ago
- Last failed build (#5), 2 hr 6 min ago
- Last unsuccessful build (#5), 2 hr 6 min ago
- Last completed build (#6), 1 hr 59 min ago

Build History trend

Filter...

Build Number	Status	Timestamp
#6	Success	Sep 26, 2024, 12:08 AM
#5	Failure	Sep 26, 2024, 12:00 AM
#4	Failure	Sep 25, 2024, 11:37 PM
#3	Failure	Sep 25, 2024, 11:37 PM
#2	Failure	Sep 25, 2024, 11:36 PM



The screenshot shows the Jenkins Build Details page for job 'adv DevOps exp-7' build #6. The left sidebar contains a 'Status' tab and a list of actions: Changes, Console Output, Edit Build Information, Delete build '#6', Timings, Git Build Data, and Previous Build. The main content area displays the build's status as '#6 (Sep 26, 2024, 12:08:16 AM)' with a green checkmark. Below this, there is a 'Started by user' section, a 'This run spent' section with a list of timings, a 'git' section with revision and repository information, and a 'No changes' section.

Dashboard > adv DevOps exp-7 > #6

Status

#6 (Sep 26, 2024, 12:08:16 AM)

Started by user Prathamesh parmeshwar palve

This run spent:

- 15 ms waiting:
- 27 sec build duration:
- 27 sec total from scheduled to completion.

git

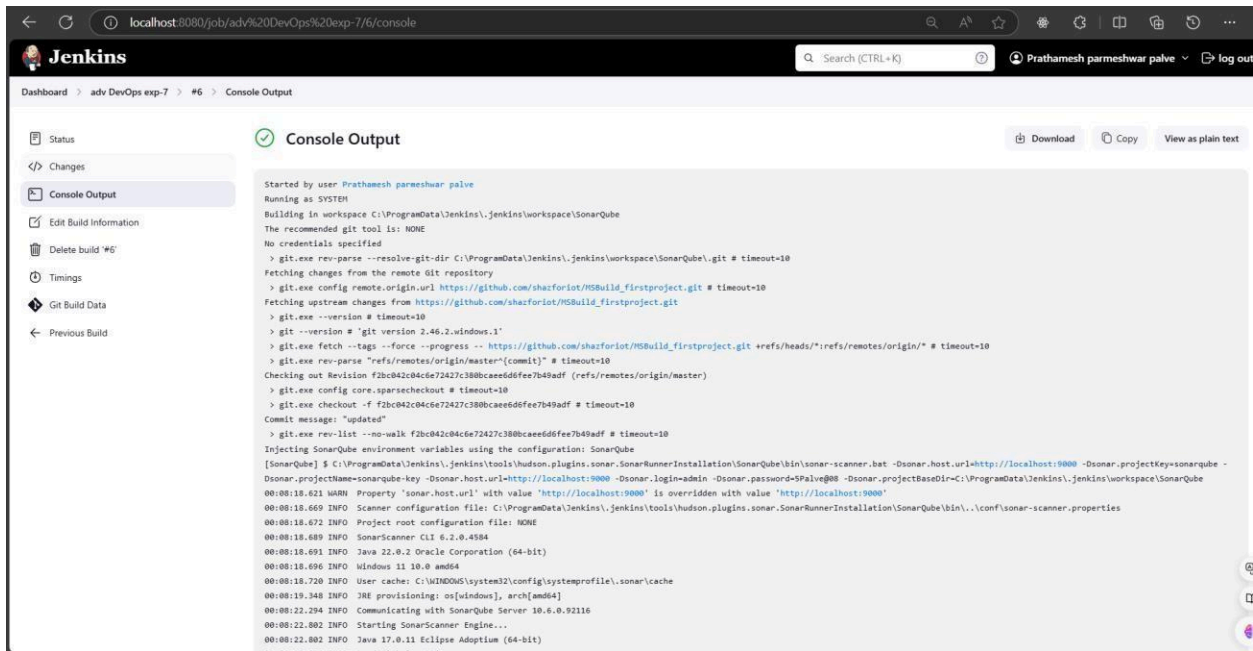
Revision: f2bc042c04c6e72427c380bcae6d6fee7b49adf

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

- refs/remotes/origin/master

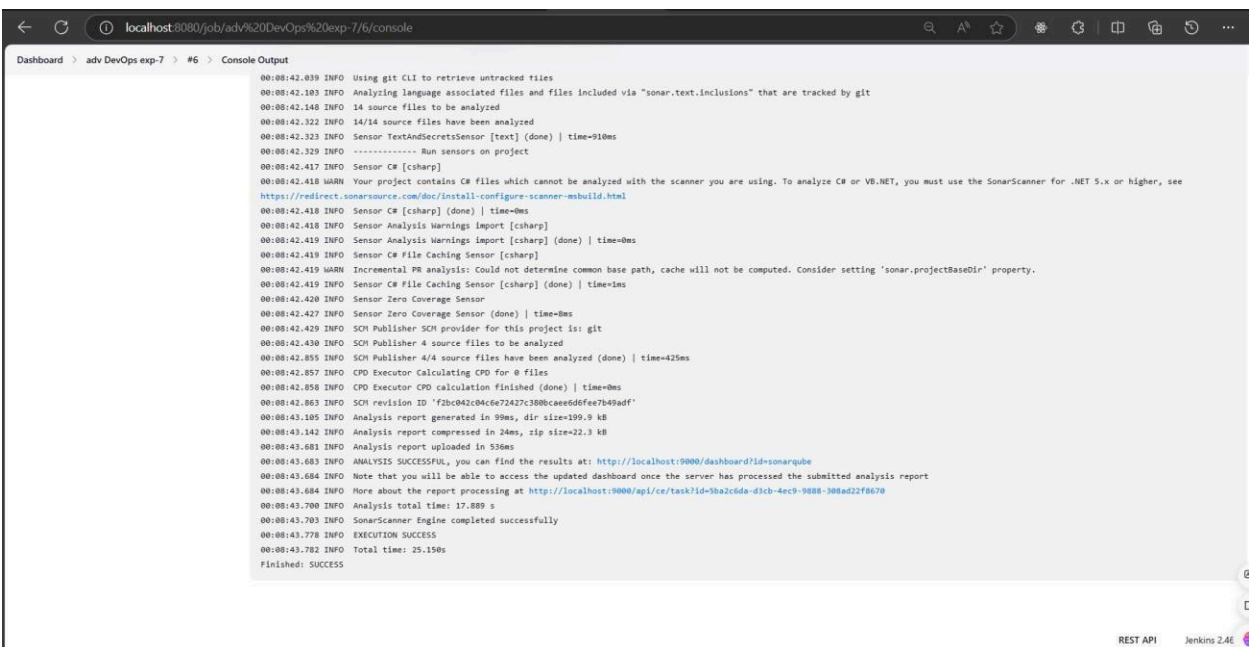
No changes.

Console Output:



The screenshot shows the Jenkins web interface with the 'Console Output' tab selected. The output text is as follows:

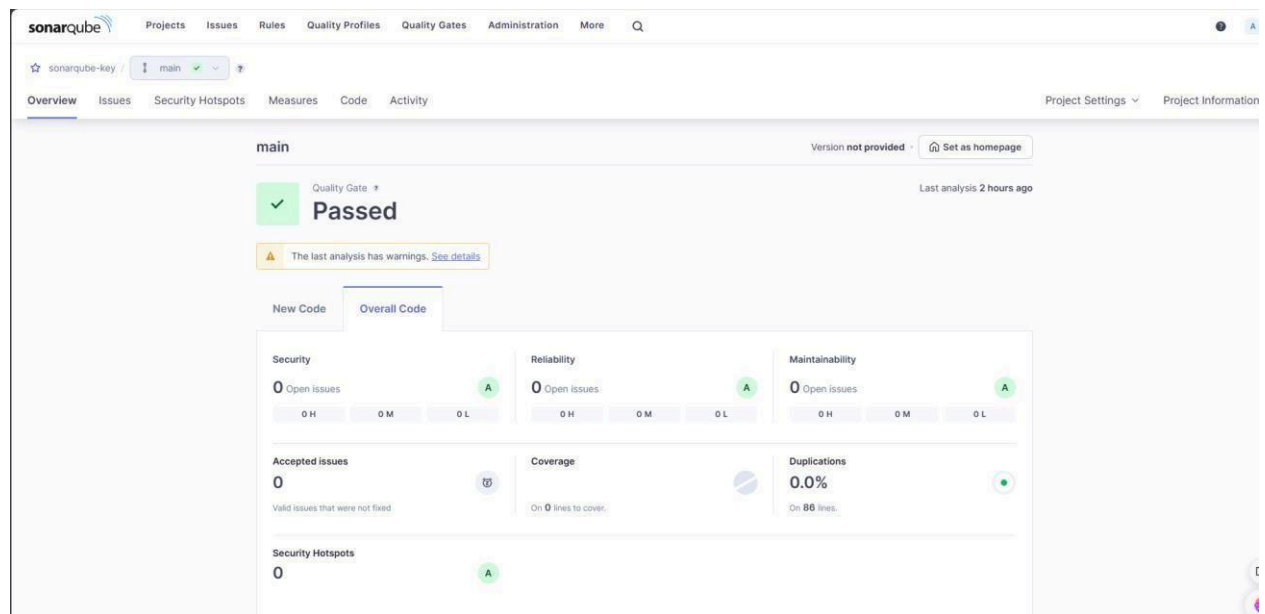
```
Started by user Prathamesh parmeshwar palve
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\workspace\SonarQube
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\workspace\SonarQube\.git # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/shaforiot/MSBuild_FirstProject.git # timeout=10
Fetching upstream changes from https://github.com/shaforiot/MSBuild_FirstProject.git
> git.exe --version # timeout=10
> git.exe --version # 'git version 2.46.2.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/shaforiot/MSBuild_FirstProject.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe rev-parse 'refs/remotes/origin/master[:commit]?' # timeout=10
Checking out Revision f2bc042c04c6e72427c380bcae6d6fee7b49adf (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
Commit message: 'updated'
> git.exe rev-list --no-walk f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
Injecting SonarQube environment variables using the configuration: SonarQube
[SonarQube] $ C:\ProgramData\Jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\SonarQube\bin\sonar-scanner.bat -Dsonar.host.url=http://localhost:9000 -Dsonar.projectKey=sonarqube -
Dsonar.projectName=sonarqube-key -Dsonar.host.url=http://localhost:9000 -Dsonar.login=admin -Dsonar.password=5Palve008 -Dsonar.projectBaseDir=C:\ProgramData\Jenkins\workspace\SonarQube
00:00:18.621 WARN Property 'sonar.host.url' with value 'http://localhost:9000' is overridden with value 'http://localhost:9000'
00:00:18.669 INFO Scanner configuration file: C:\ProgramData\Jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\SonarQube\bin\..conf\sonar-scanner.properties
00:00:18.672 INFO Project root configuration file: NONE
00:00:18.689 INFO SonarScanner 4.2.0.4564
00:00:18.691 INFO Java 21.0.2 Oracle Corporation (64-bit)
00:00:18.696 INFO Windows 11 10.0 amd64
00:00:18.720 INFO User cache: C:\WINDOWS\system32\config\systemprofile\.sonar\cache
00:00:19.348 INFO JRE provisioning: os[windows], arch[amd64]
00:00:22.294 INFO Communicating with SonarQube Server 10.6.0.92116
00:00:22.802 INFO Starting SonarScanner Engine...
00:00:22.802 INFO Java 17.0.11 Eclipse Adoptium (64-bit)
```



The screenshot shows the continuation of the Jenkins console output, detailing the analysis process and the final report upload:

```
00:00:42.809 INFO Using git CLI to retrieve untracked files
00:00:42.183 INFO Analyzing language associated files and files included via "sonar.text.inclusions" that are tracked by git
00:00:42.148 INFO 14 source files to be analyzed
00:00:42.322 INFO 14/14 source files have been analyzed
00:00:42.323 INFO Sensor TextAndSecretsSensor [text] (done) | time=910ms
00:00:42.329 INFO ----- Run sensors on project
00:00:42.417 INFO Sensor C# [csharp]
00:00:42.418 WARN Your project contains C# files which cannot be analyzed with the scanner you are using. To analyze C# or VB.NET, you must use the SonarScanner for .NET 5.x or higher, see
https://redirect.sonarsource.com/doc/install-configure-scanner-msbuild.html
00:00:42.418 INFO Sensor C# [csharp] (done) | time=0ms
00:00:42.418 INFO Sensor Analysis Warnings Import [csharp]
00:00:42.419 INFO Sensor Analysis Warnings Import [csharp] (done) | time=0ms
00:00:42.419 INFO Sensor C# File Caching Sensor [csharp]
00:00:42.419 INFO Incremental PM analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.projectBaseDir' property.
00:00:42.419 INFO Sensor C# File Caching Sensor [csharp] (done) | time=1ms
00:00:42.420 INFO Sensor Zero Coverage Sensor
00:00:42.427 INFO Sensor Zero Coverage Sensor (done) | time=0ms
00:00:42.429 INFO SCM Publisher SCM provider for this project is: git
00:00:42.430 INFO SCM Publisher 4 source files to be analyzed
00:00:42.855 INFO SCM Publisher 4/4 source files have been analyzed (done) | time=425ms
00:00:42.857 INFO CPD Executor Calculating CPD for 0 files
00:00:42.858 INFO CPD Executor CPD calculation finished (done) | time=0ms
00:00:42.863 INFO SCM revision ID 'f2bc042c04c6e72427c380bcae6d6fee7b49adf'
00:00:43.105 INFO Analysis report generated in 99ms, dir size=199.9 kB
00:00:43.142 INFO Analysis report compressed in 24ms, zip size=22.3 kB
00:00:43.601 INFO Analysis report uploaded in 536ms
00:00:43.603 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=sonarqube
00:00:43.604 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
00:00:43.604 INFO Here about the report processing at http://localhost:9000/api/cx/task?id=5ba2c6da-d3cb-4ec9-9088-308ad2f6670
00:00:43.700 INFO Analysis total time: 17.889 s
00:00:43.703 INFO SonarScanner Engine completed successfully
00:00:43.778 INFO EXECUTION SUCCESS
00:00:43.782 INFO Total time: 25.150s
Finished: SUCCESS
```

Project on sonarqube:



Conclusion: Thus, we have successfully installed SonarQube from Docker image.