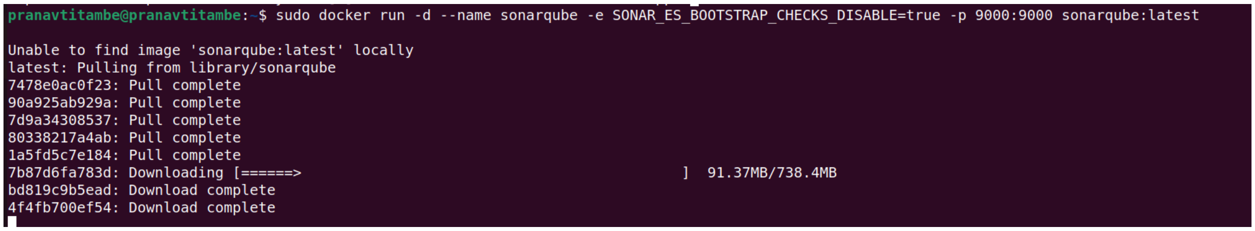
ADVANCE DEVOPS - 7

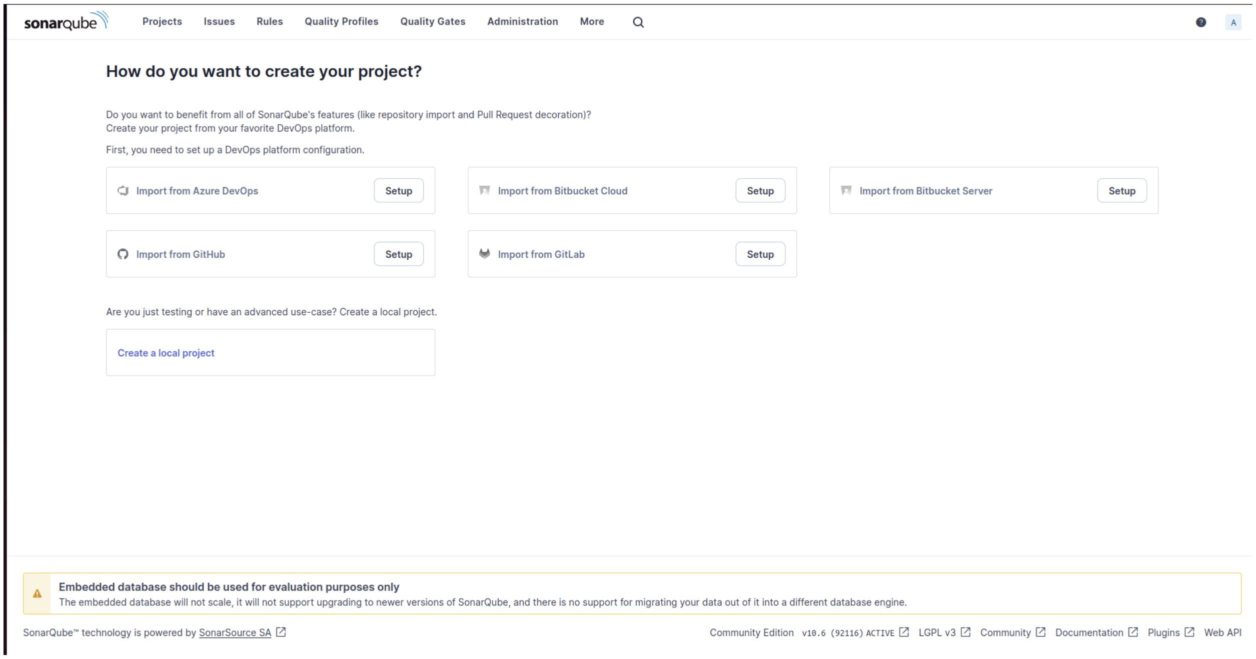
NAME - PRANAV TITAMBE D15A/62

Install SonarQube docker image from the docker registry.
SonarQube will run on http://localhost:9000



```
pranavtitambe@pranavtitambe:~$ sudo 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: Downloading [======>                                            ]  91.37MB/738.4MB
bd819c9b5ead: Download complete
4f4fb700ef54: Download complete
```



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.

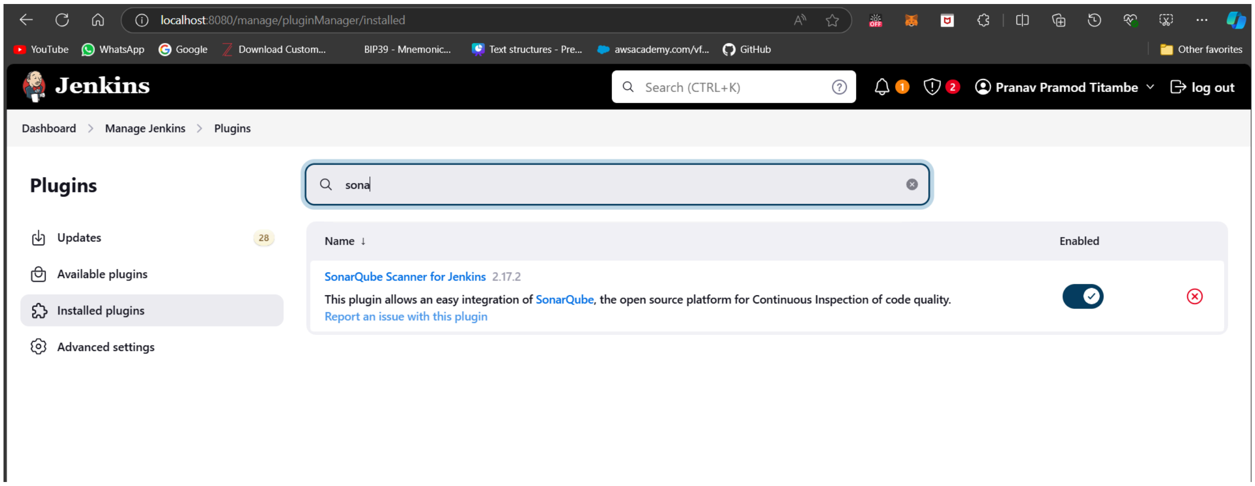
First, you need to set up a DevOps platform configuration.

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

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

Step 1 - Login to Jenkins dashboard.

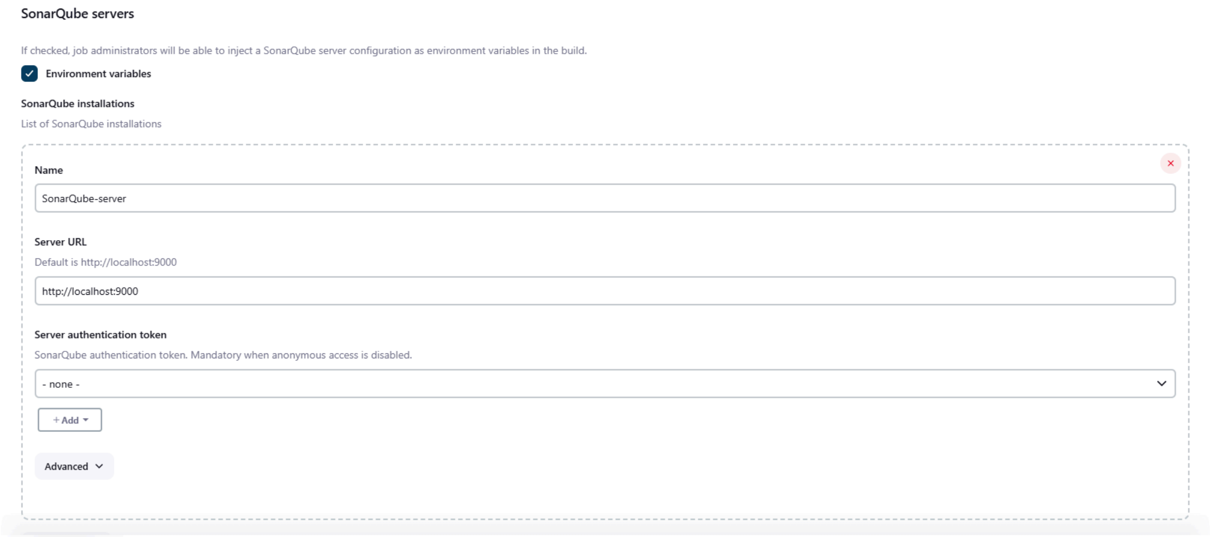


Plugins

sona

Name ↓	Enabled
SonarQube Scanner for Jenkins 2.17.2 This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality. Report an issue with this plugin	✓

Step 2 - Add the SonarQube Server to the Jenkins configuration in Configure System option.



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

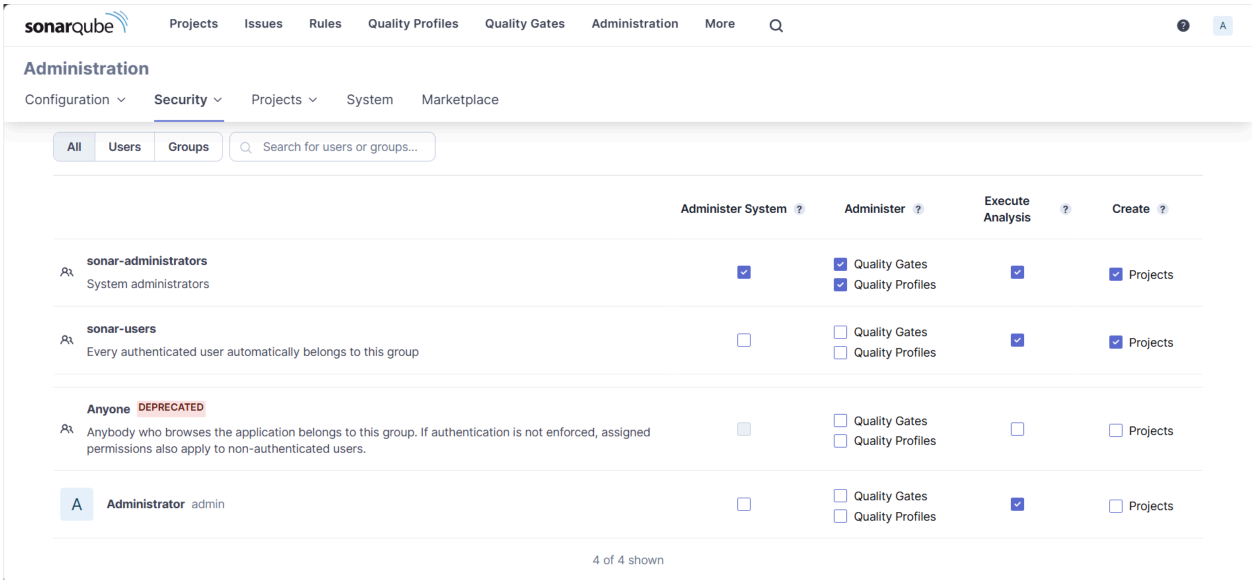
List of SonarQube installations

Name: SonarQube-server

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

Server authentication token
SonarQube authentication token. Mandatory when anonymous access is disabled.
- none -

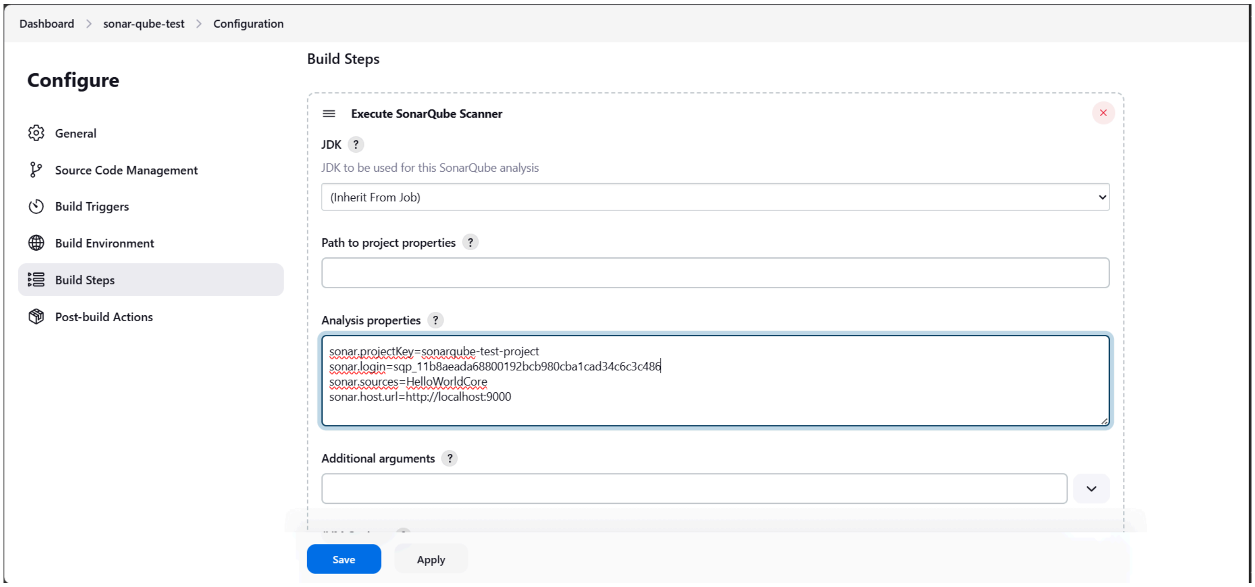
Step 3 - Login to SonarQube dashboard which is by default installed on http://localhost:9000



	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 4 - Create a new free style project and add the github link in the github section of this following new item and add these properties in the build steps.



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

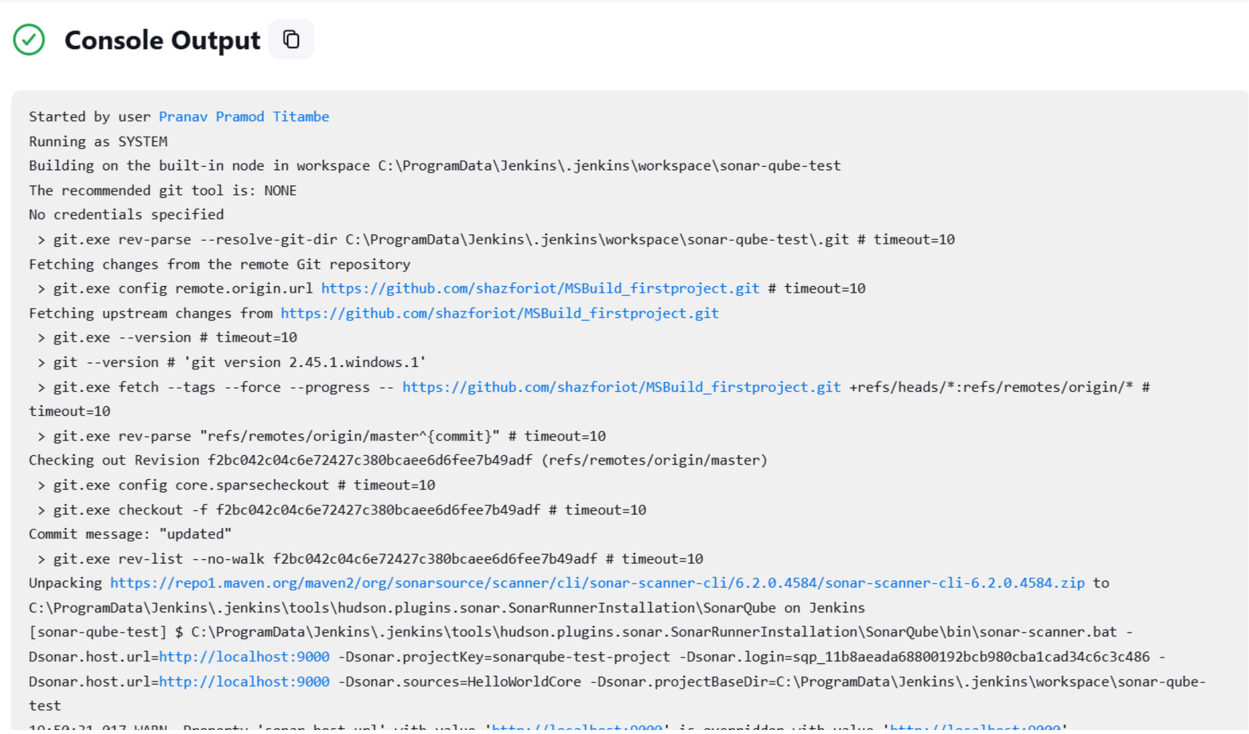
Path to project properties:

Analysis properties:

```
sonar.projectKey=sonarqube-test-project
sonar.login=sqp_11b8aeada68800192bcb980cba1cad34c6c3c486
sonar.sources=HelloWorldCore
sonar.host.url=http://localhost:9000
```

Additional arguments:

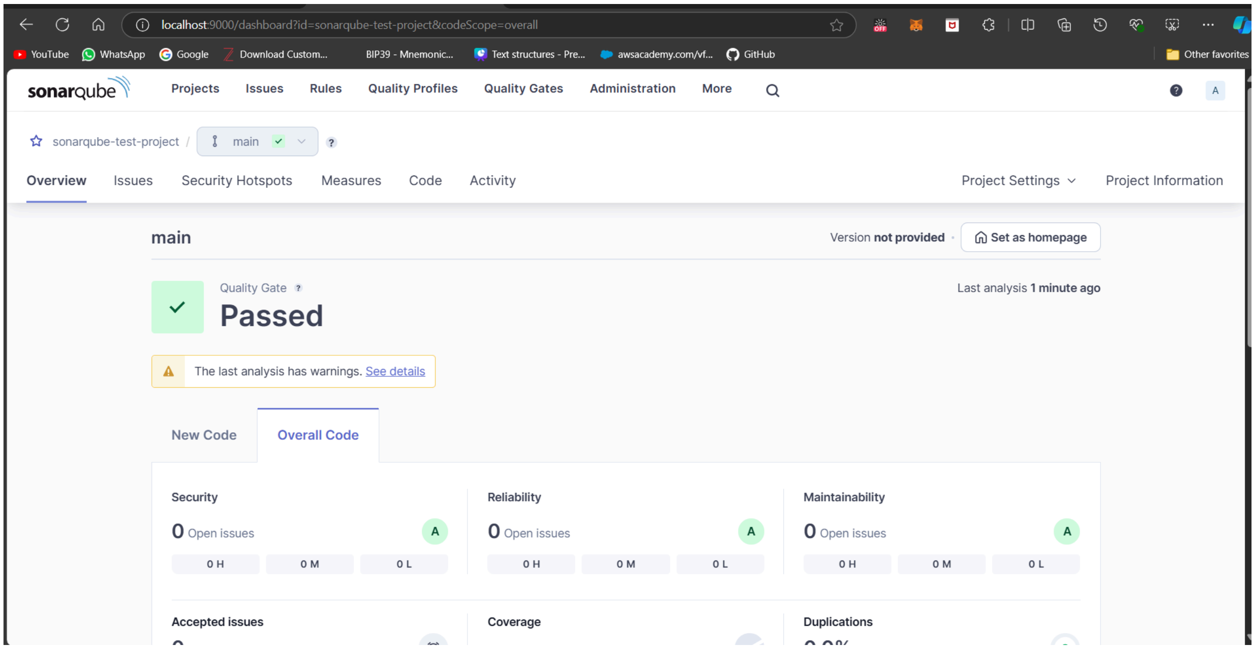
Step 5 - Click on Bulid now option to build your free style project



Console Output

```
Started by user Pranav Pramod Titambe
Running as SYSTEM
Building on the built-in node in workspace C:\ProgramData\Jenkins\.jenkins\workspace\sonar-qube-test
The recommended git tool is: NONE
No credentials specified
 > git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\sonar-qube-test\.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
Unpacking https://repo1.maven.org/maven2/org/sonarsource/scanner/cli/sonar-scanner-cli/6.2.0.4584/sonar-scanner-cli-6.2.0.4584.zip to
C:\ProgramData\Jenkins\.jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\SonarQube on Jenkins
[sonar-qube-test] $ C:\ProgramData\Jenkins\.jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\SonarQube\bin\sonar-scanner.bat -
Dsonar.host.url=http://localhost:9000 -Dsonar.projectKey=sonarqube-test-project -Dsonar.login=sqp_11b8aeada68800192bcb980cba1cad34c6c3c486 -
Dsonar.host.url=http://localhost:9000 -Dsonar.sources=HelloWorldCore -Dsonar.projectBaseDir=C:\ProgramData\Jenkins\.jenkins\workspace\sonar-qube-
test
19:59:21.017 WARN  Property 'sonar.host.url' with value 'http://localhost:9000' is overridden with value 'http://localhost:9000'
```

Step 6 - Check your SonarQube dashboard and you will get the following analysis.



main

Version **not provided**

Quality Gate

Passed

Last analysis **1 minute ago**

The last analysis has warnings. See details

New Code | Overall Code

Security	Reliability	Maintainability
0 Open issues A	0 Open issues A	0 Open issues A
0 H · 0 M · 0 L	0 H · 0 M · 0 L	0 H · 0 M · 0 L
Accepted issues	Coverage	Duplications