ADVANCE DEVOPS EXP-7

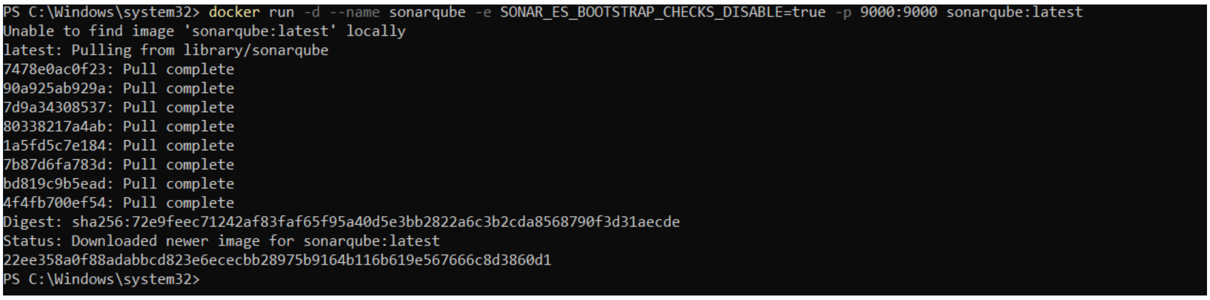
Aim :- To understand Static Analysis SAST process and learn to integrate Jenkins SAST to SonarQube/GitLab.

Step 1: Installing SonarQube from the Docker Image

$ docker run -d --name sonarqube -e

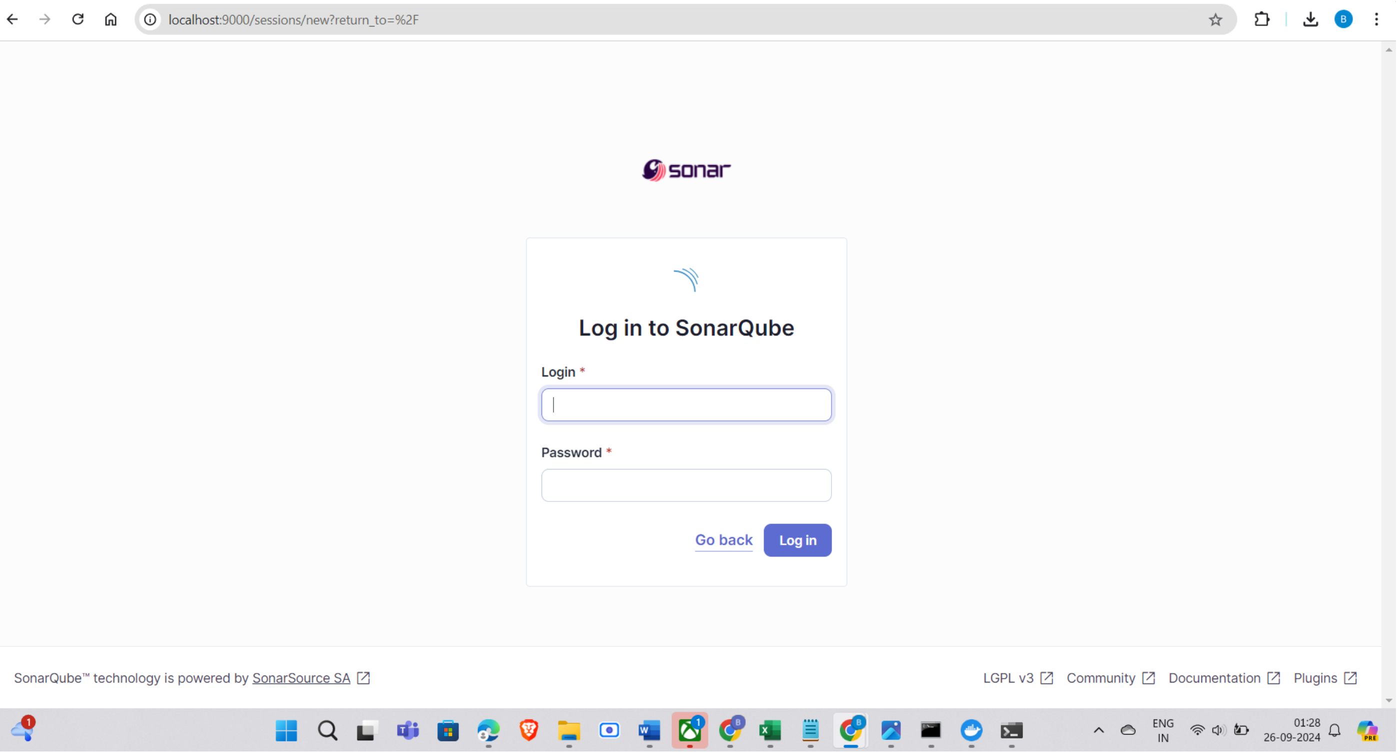
SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000

sonarqube:latest



```
PS C:\Windows\system32> 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: Pull complete
bd819c9b5ead: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Downloaded newer image for sonarqube:latest
22ee358a0f88adabbcd823e6ececbb28975b9164b116b619e567666c8d3860d1
PS C:\Windows\system32>
```

Step 2: After installation of SonarQube, go to the SonarQube page by typing: http://localhost:9000/ on your browser. If you see such page then you have successfully installed it.



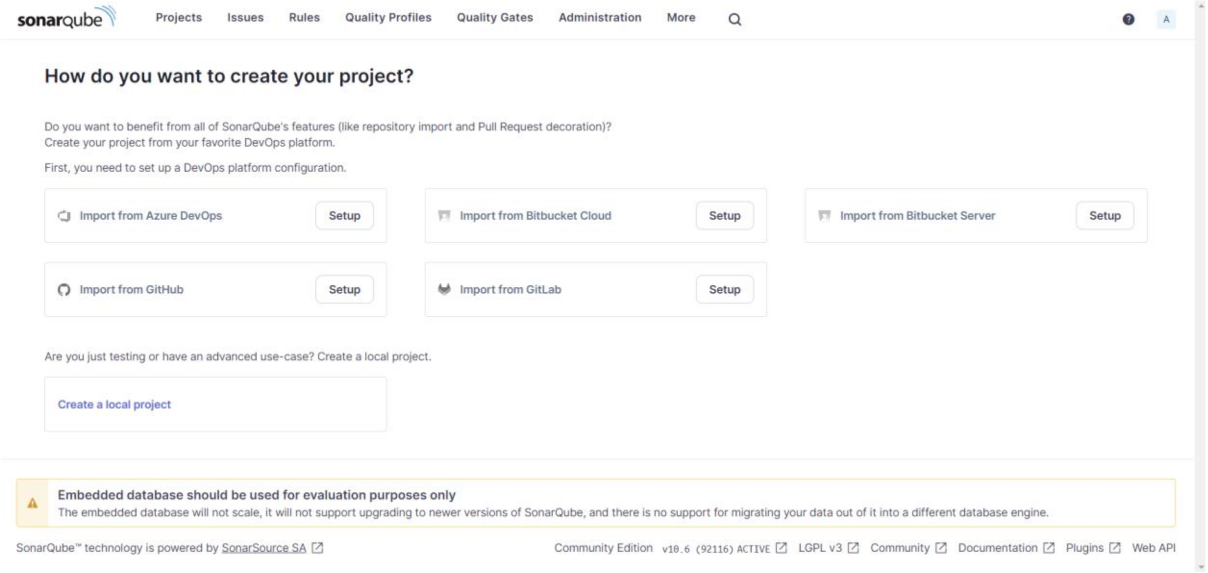
localhost:9000/sessions/new?return_to=%2F

Log in to SonarQube

Login *

Password *

Step 3: Login using the username as “admin” and password as “admin”. And then you will see the home page of SonarQube



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.

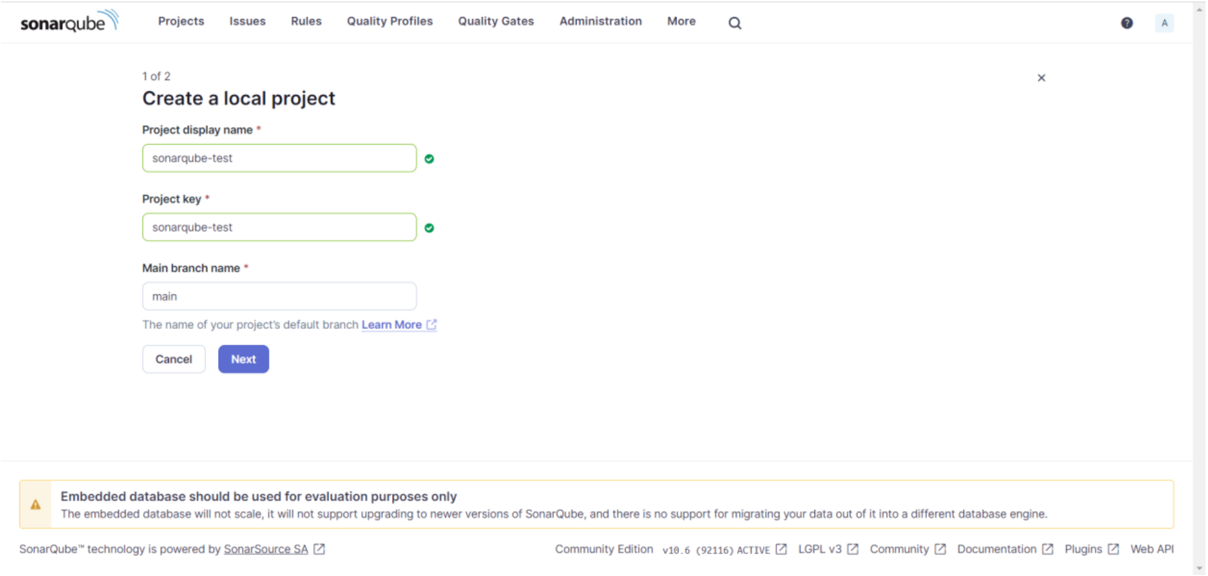
Are you just testing or have an advanced use-case? Create a local project.

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 4 :- give the name project



1 of 2

Create a local project

Project display name *: sonarqube-test

Project key *: sonarqube-test

Main branch name *: main

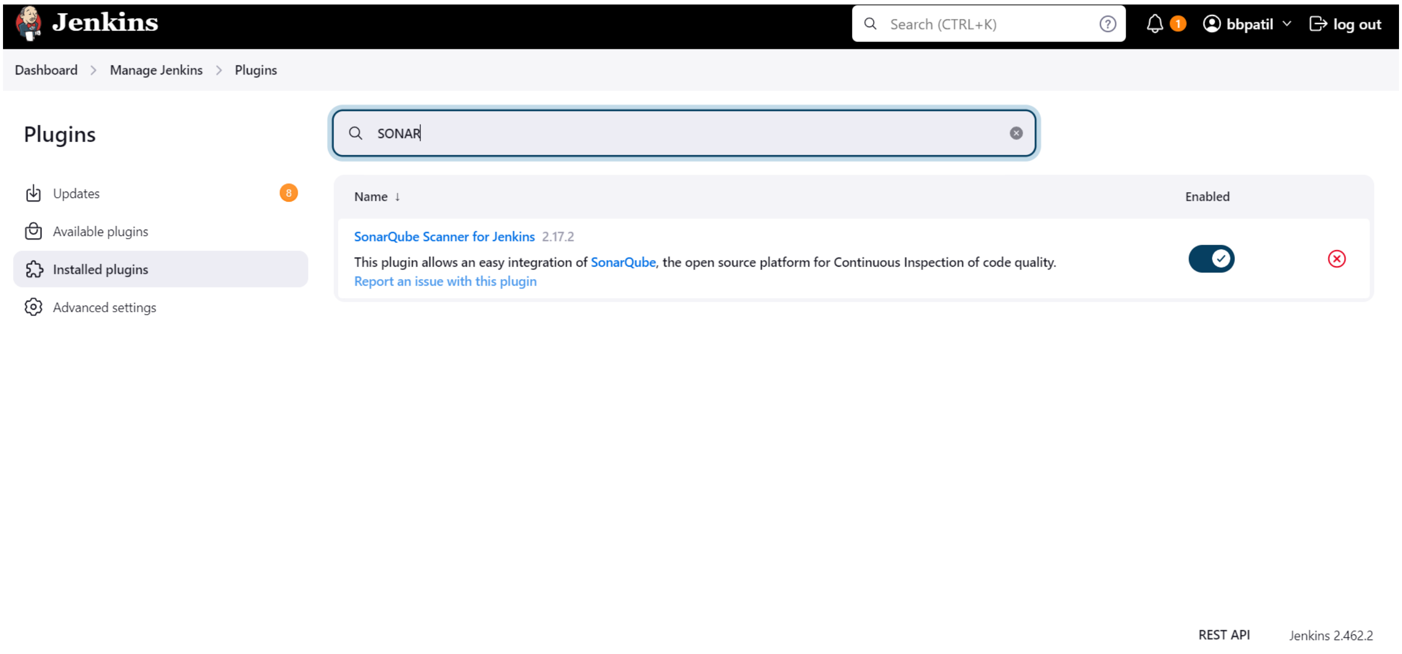
The name of your project's default branch Learn More

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 5: Go to Dashboard > Manage Jenkins > Plugin Manager and search for SonarQube Scanner under Available plugins for Jenkins and install it



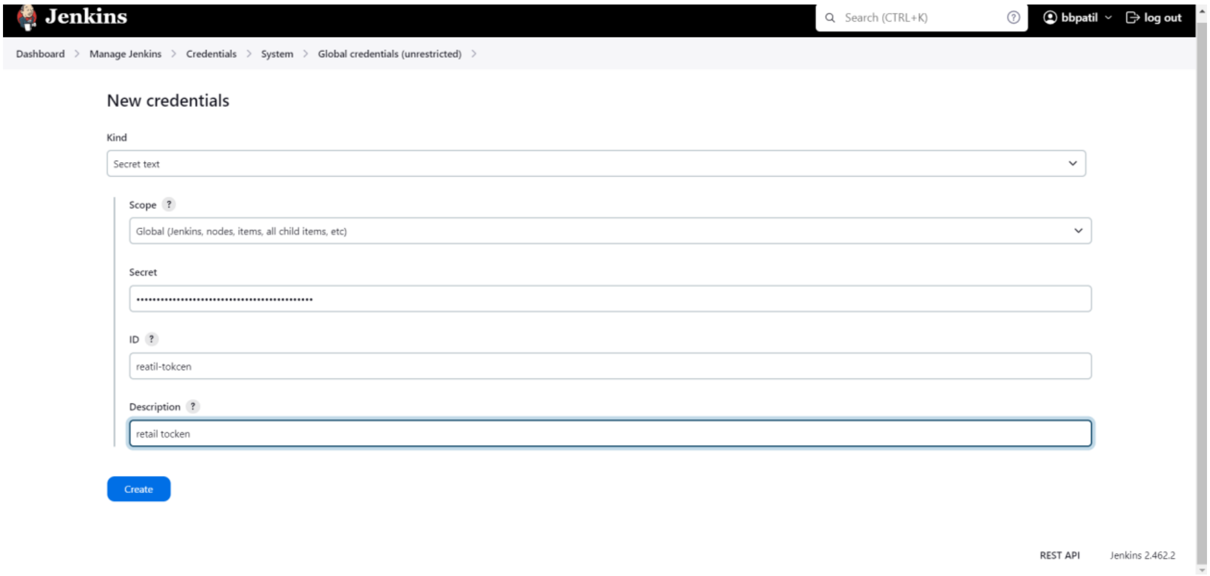
Dashboard > Manage Jenkins > Plugins

Plugins

SONAR

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 6:- add authentication all details



Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

New credentials

Kind: Secret text

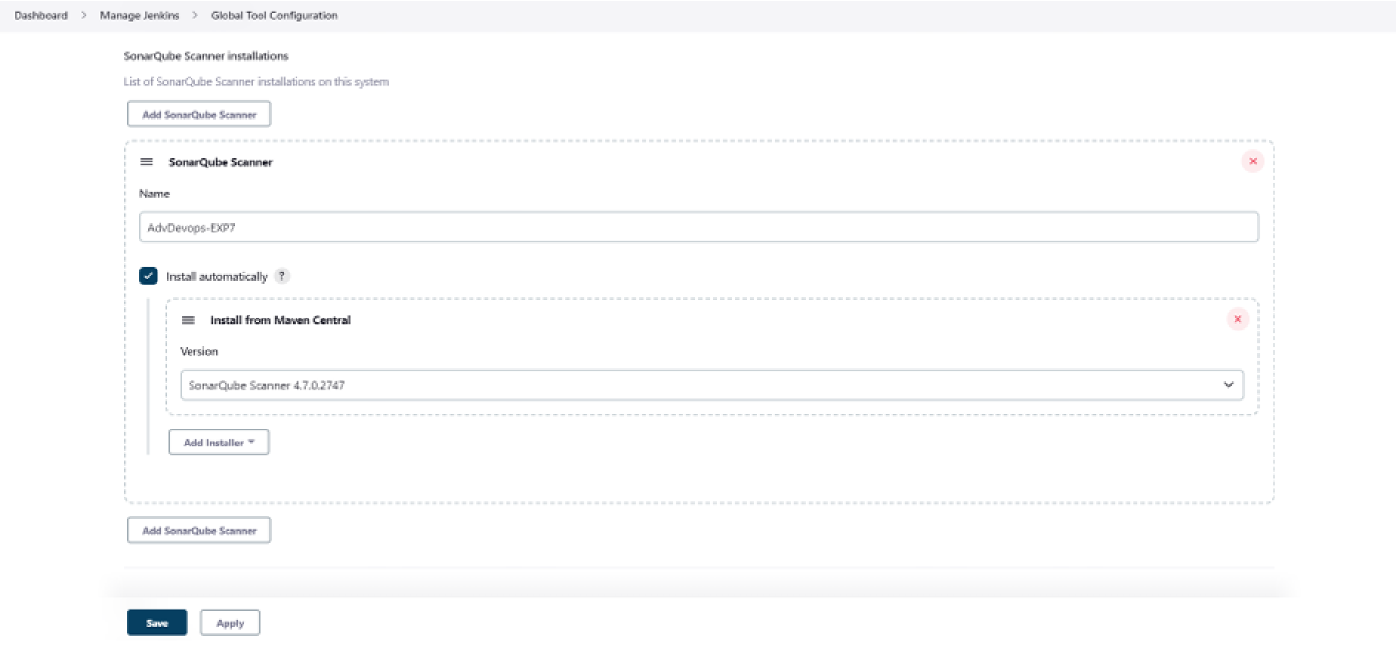
Scope: Global (Jenkins, nodes, items, all child items, etc)

Secret: ••

ID: reatil-tokcen

Description: retail tocken

Step 7: Search SonarQube Scanner under Dashboard > Manage Jenkins > Global Tool Configuration. Choose the latest configuration and choose Install Automatically



Dashboard > Manage Jenkins > Global Tool Configuration

SonarQube Scanner installations

List of SonarQube Scanner installations on this system

SonarQube Scanner

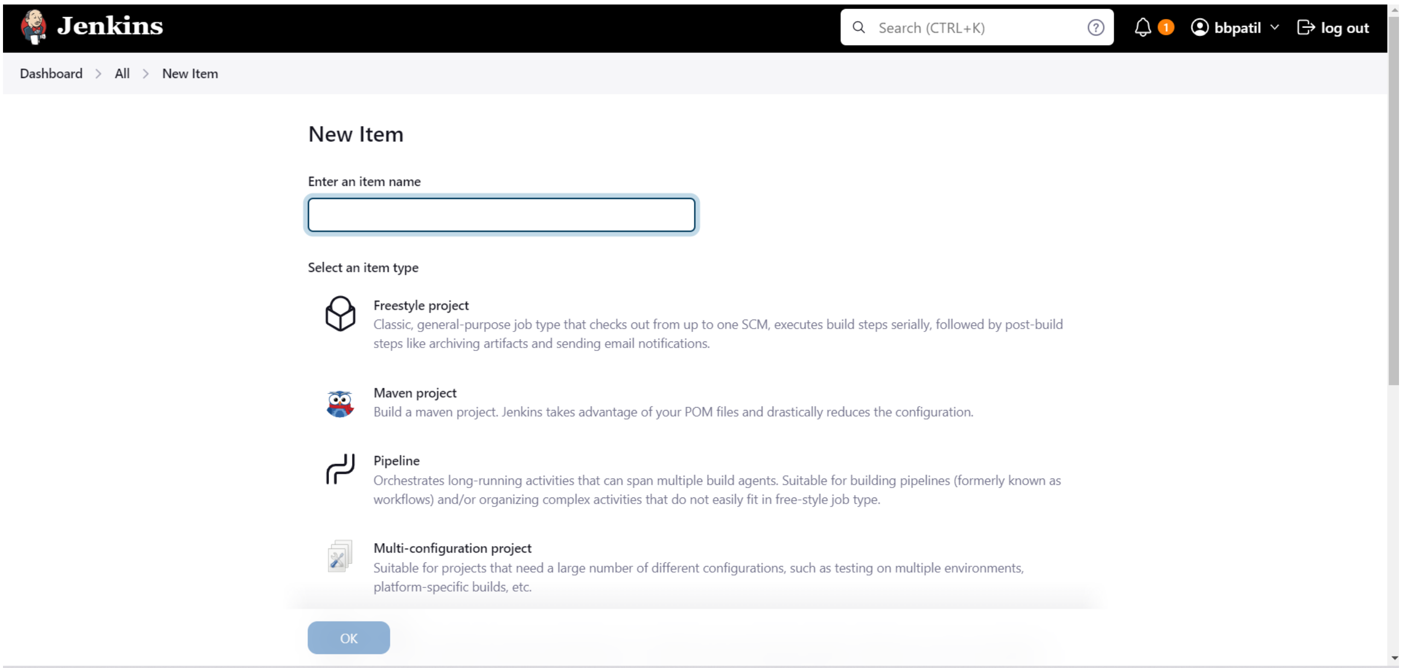
Name: AdvDevops-EXP7

☑ Install automatically

Install from Maven Central

Version: SonarQube Scanner 4.7.0.2747

Step 8: After the configuration, create a New Item in Jenkins, choose a freestyle project.



Dashboard > All > New Item

New Item

Enter an item name

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.

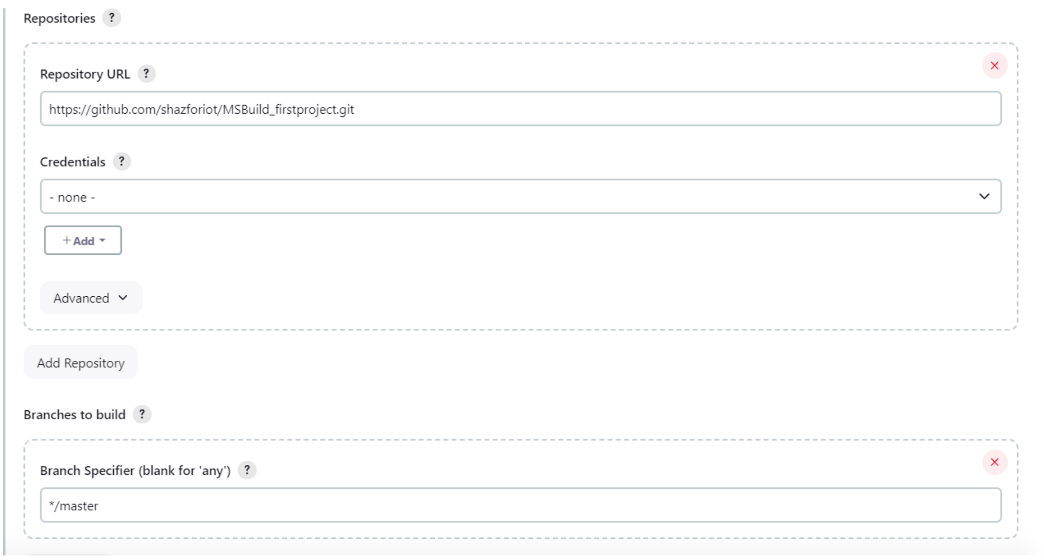
Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

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

Step 9 :- add the git url and also all details

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

step 10 :- under the bulid command all the details plz correctly and give
Mention the SonarQube Project Key, Login, Password, and
Host URL. sonar.projectKey=retail-stock
sonar.login=*your username*
sonar.password=*your password*
sonar.hosturl=http://localhost:9000



Repositories ?

Repository URL ?

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

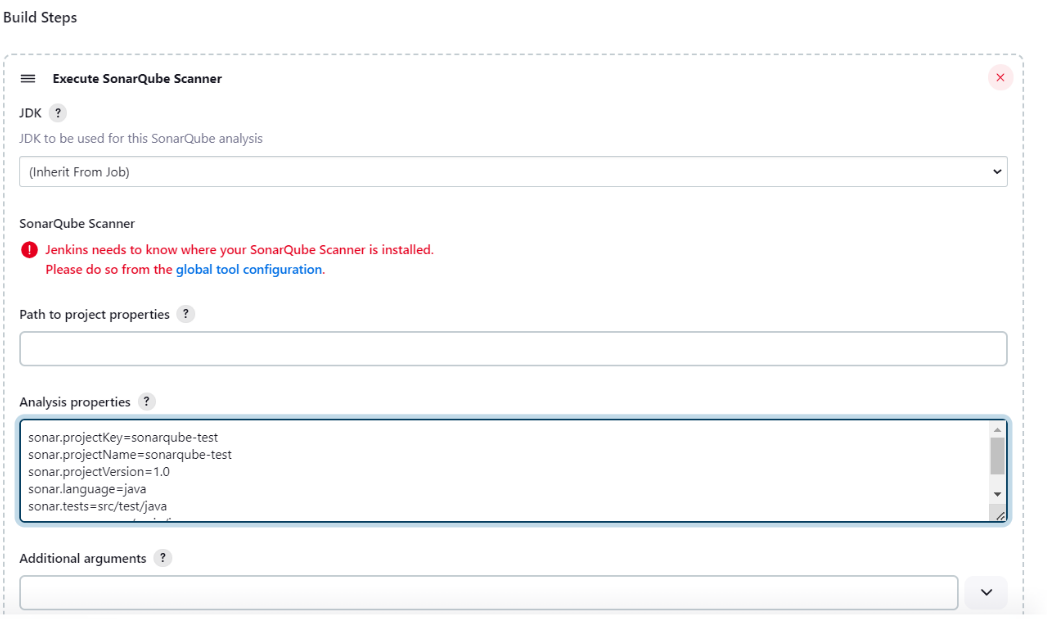
Credentials ?

- none -

Branches to build ?

Branch Specifier (blank for 'any') ?

*/master



Build Steps

Execute SonarQube Scanner

JDK ?

JDK to be used for this SonarQube analysis

(Inherit From Job)

SonarQube Scanner

Jenkins needs to know where your SonarQube Scanner is installed.
Please do so from the global tool configuration.

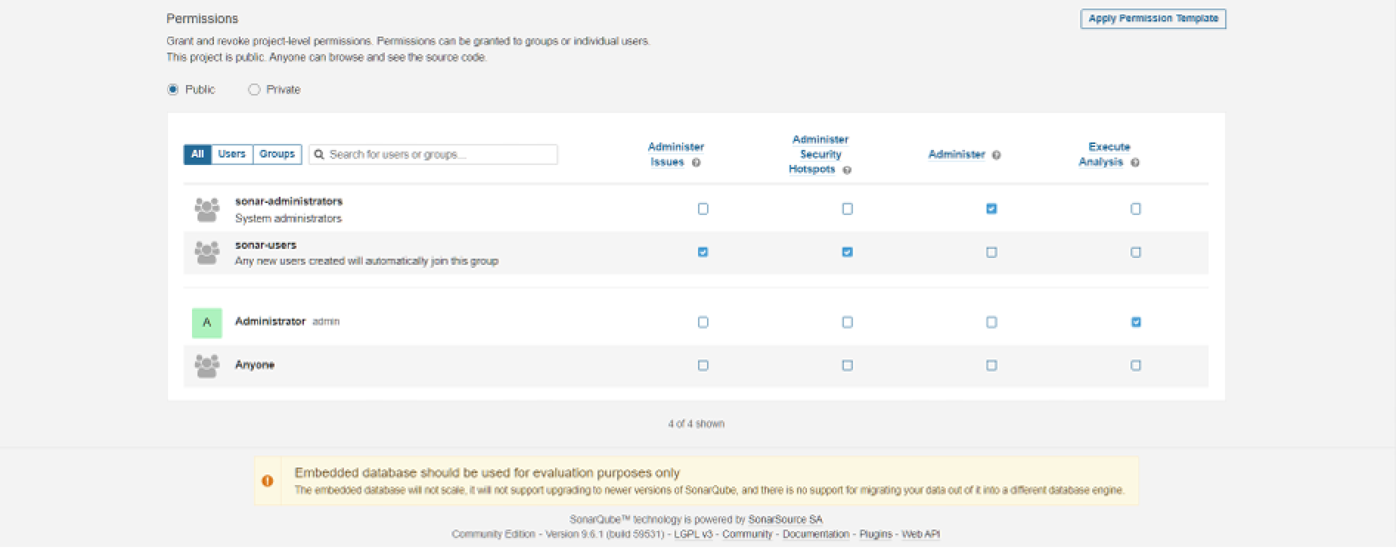
Path to project properties ?

Analysis properties ?

```
sonar.projectKey=sonarqube-test
sonar.projectName=sonarqube-test
sonar.projectVersion=1.0
sonar.language=java
sonar.tests=src/test/java
```

Additional arguments ?

Step 11: 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

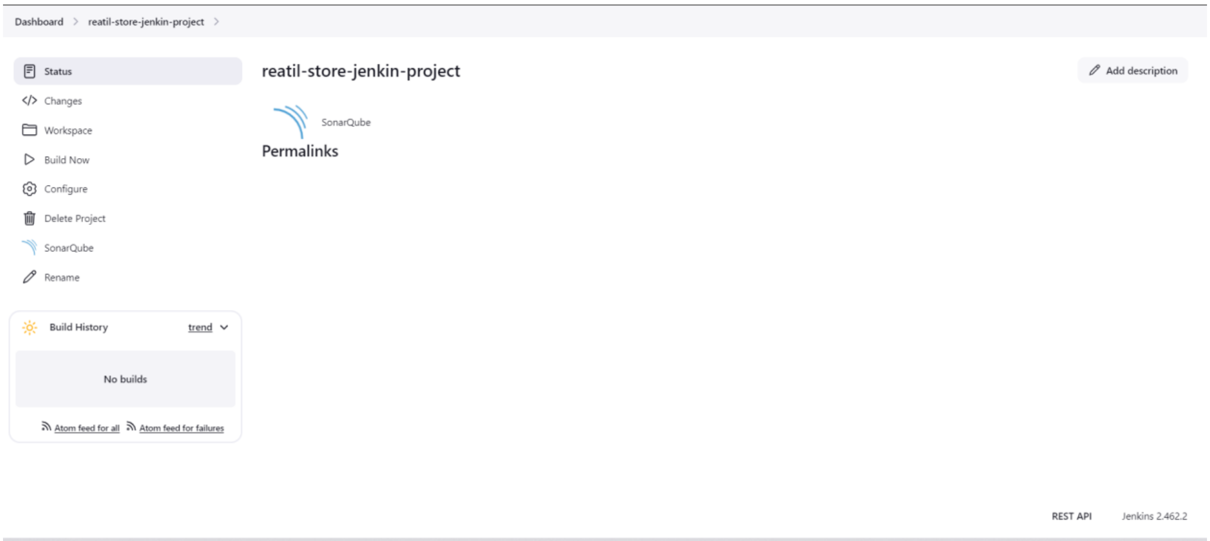
	Administer Issues	Administer Security Hotspots	Administer	Execute Analysis
sonar-administrators System administrators	□	□	☑	□
sonar-users Any new users created will automatically join this group	☑	☑	□	□
Administrator admin	□	□	□	☑
Anyone	□	□	□	□

4 of 4 shown

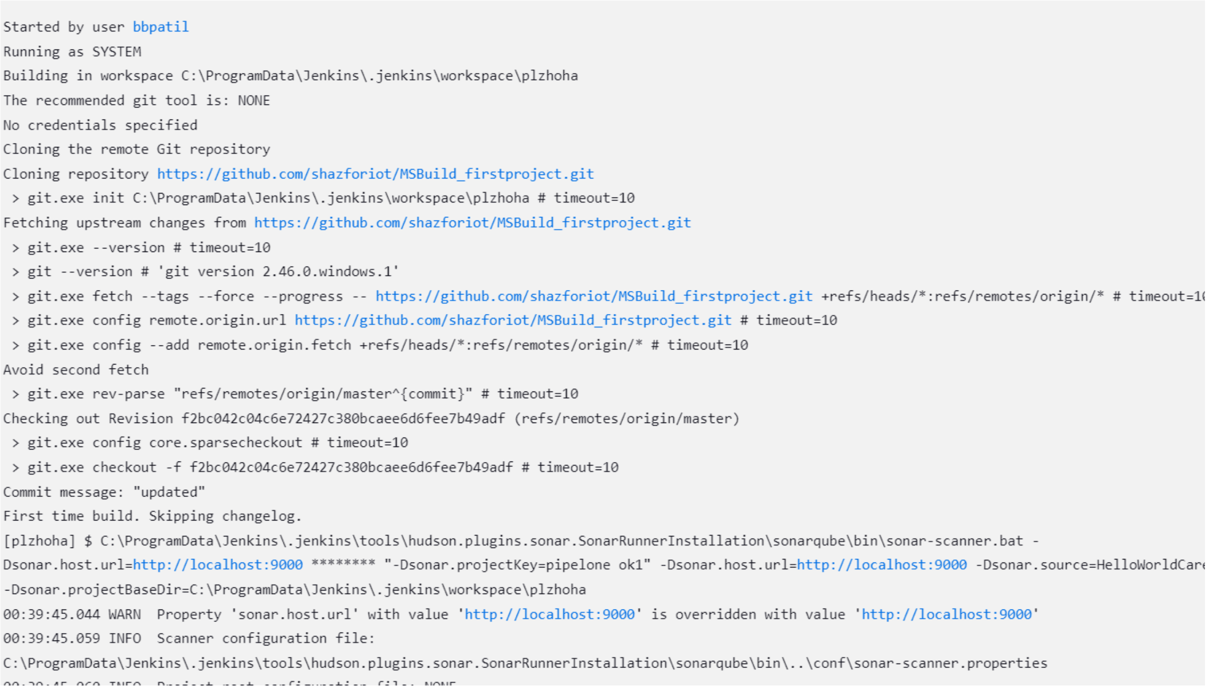
Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 9.6.1 (build 59531) - LGPL v3 - Community - Documentation - Plugins - Web API

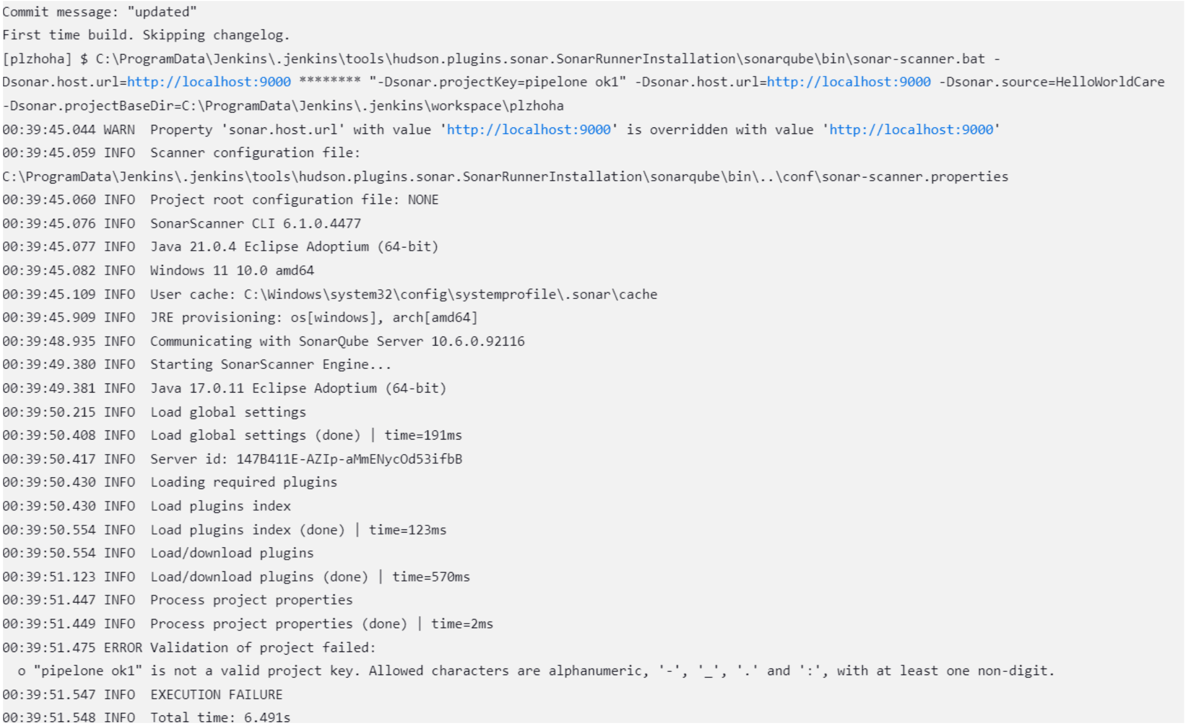
Step 12: Run The Build



console output

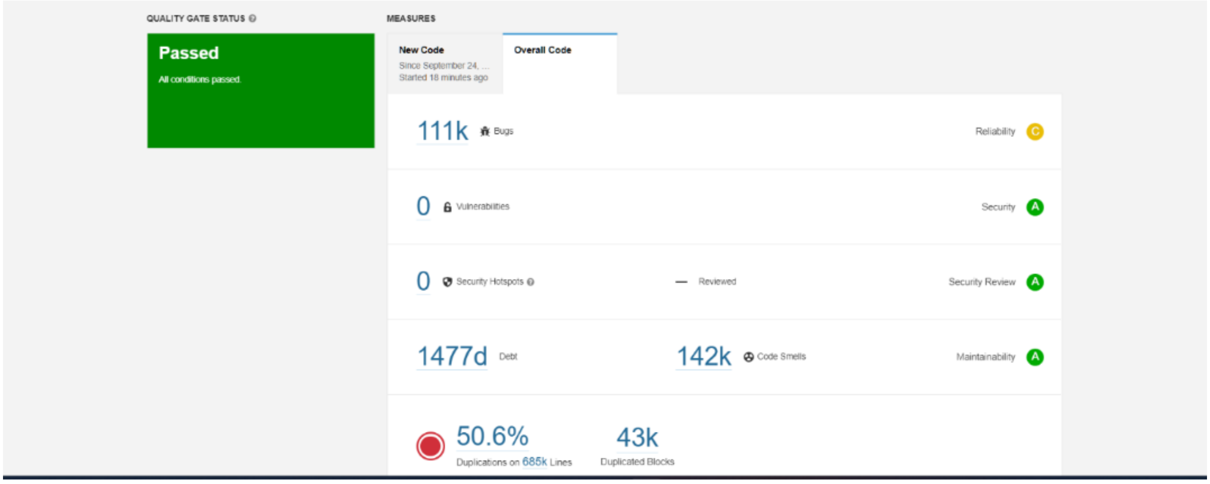


```
Started by user bbpatil
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\plzhoha
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/shazforiot/MSBuild_firstproject.git
 > git.exe init C:\ProgramData\Jenkins\.jenkins\workspace\plzhoha # timeout=10
Fetching upstream changes from https://github.com/shazforiot/MSBuild_firstproject.git
 > git.exe --version # timeout=10
 > git --version # 'git version 2.46.0.windows.1'
 > git.exe fetch --tags --force --progress -- https://github.com/shazforiot/MSBuild_firstproject.git +refs/heads/*:refs/remotes/origin/* # timeout=1
 > git.exe config remote.origin.url https://github.com/shazforiot/MSBuild_firstproject.git # timeout=10
 > git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
 > 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"
First time build. Skipping changelog.
[plzhoha] $ C:\ProgramData\Jenkins\.jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube\bin\sonar-scanner.bat -
Dsonar.host.url=http://localhost:9000 ******** "-Dsonar.projectKey=pipelone ok1" -Dsonar.host.url=http://localhost:9000 -Dsonar.source=HelloWorldCar
-Dsonar.projectBaseDir=C:\ProgramData\Jenkins\.jenkins\workspace\plzhoha
00:39:45.044 WARN  Property 'sonar.host.url' with value 'http://localhost:9000' is overridden with value 'http://localhost:9000'
00:39:45.059 INFO  Scanner configuration file:
C:\ProgramData\Jenkins\.jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube\bin\..\conf\sonar-scanner.properties
00:39:45.060 INFO  Project root configuration file: NONE
```



```
Commit message: "updated"
First time build. Skipping changelog.
[plzhoha] $ C:\ProgramData\Jenkins\.jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube\bin\sonar-scanner.bat -
Dsonar.host.url=http://localhost:9000 ******** "-Dsonar.projectKey=pipelone ok1" -Dsonar.host.url=http://localhost:9000 -Dsonar.source=HelloWorldCare
-Dsonar.projectBaseDir=C:\ProgramData\Jenkins\.jenkins\workspace\plzhoha
00:39:45.044 WARN  Property 'sonar.host.url' with value 'http://localhost:9000' is overridden with value 'http://localhost:9000'
00:39:45.059 INFO  Scanner configuration file:
C:\ProgramData\Jenkins\.jenkins\tools\hudson.plugins.sonar.SonarRunnerInstallation\sonarqube\bin\..\conf\sonar-scanner.properties
00:39:45.060 INFO  Project root configuration file: NONE
00:39:45.076 INFO  SonarScanner CLI 6.1.0.4477
00:39:45.077 INFO  Java 21.0.4 Eclipse Adoptium (64-bit)
00:39:45.082 INFO  Windows 11 10.0 amd64
00:39:45.109 INFO  User cache: C:\Windows\system32\config\systemprofile\.sonar\cache
00:39:45.909 INFO  JRE provisioning: os[windows], arch[amd64]
00:39:48.935 INFO  Communicating with SonarQube Server 10.6.0.92116
00:39:49.380 INFO  Starting SonarScanner Engine...
00:39:49.381 INFO  Java 17.0.11 Eclipse Adoptium (64-bit)
00:39:50.215 INFO  Load global settings
00:39:50.408 INFO  Load global settings (done) | time=191ms
00:39:50.417 INFO  Server id: 147B411E-AZIp-aMmENycOd53ifbB
00:39:50.430 INFO  Loading required plugins
00:39:50.430 INFO  Load plugins index
00:39:50.554 INFO  Load plugins index (done) | time=123ms
00:39:50.554 INFO  Load/download plugins
00:39:51.123 INFO  Load/download plugins (done) | time=570ms
00:39:51.447 INFO  Process project properties
00:39:51.449 INFO  Process project properties (done) | time=2ms
00:39:51.475 ERROR Validation of project failed:
  o "pipelone ok1" is not a valid project key. Allowed characters are alphanumeric, '-', '_', '.' and ':', with at least one non-digit.
00:39:51.547 INFO  EXECUTION FAILURE
00:39:51.548 INFO  Total time: 6.491s
```

Due to blocking system git this screenshoot form clg pc



Passed

All conditions passed.

111k Bugs — Reliability C

0 Vulnerabilities — Security A

0 Security Hotspots — Reviewed — Security Review A

1477d Debt — 142k Code Smells — Maintainability A

50.6% Duplications on 685k Lines — 43k Duplicated Blocks