

## Screenshots for Experiment 8.

```

erShell does not load commands from the current location by default. If you trust this command, instead type: ".\sonar-s
canner.bat". See "get-help about_Command_Precedence" for more details.
PS C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin> .\sonar-scanner.bat
11:02:02.120 INFO Scanner configuration file: C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin\..\conf\sonar-s
canner.properties
11:02:02.124 INFO Project root configuration file: NONE
11:02:02.140 INFO SonarScanner CLI 6.2.0.4584
11:02:02.142 INFO Java 17.0.12 Eclipse Adoptium (64-bit)
11:02:02.142 INFO Windows 11 10.0 amd64
11:02:02.160 INFO User cache: C:\Users\navan\.sonar\cache
11:02:02.644 INFO JRE provisioning: os[windows], arch[amd64]
11:02:06.241 INFO EXECUTION FAILURE
11:02:06.243 INFO Total time: 4.126s
11:02:06.244 ERROR Error during SonarScanner CLI execution
java.lang.IllegalStateException: Error status returned by url [https://api.sonarcloud.io/analysis/jres?os=windows&arch=a
md64]: 401
    at org.sonarsource.scanner.lib.internal.http.ServerConnection.callUrl(ServerConnection.java:182)
    at org.sonarsource.scanner.lib.internal.http.ServerConnection.callApi(ServerConnection.java:145)
    at org.sonarsource.scanner.lib.internal.http.ServerConnection.callRestApi(ServerConnection.java:123)
    at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreMetadata(JavaRunnerFactory.java:159)
    at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.getJreFromServer(JavaRunnerFactory.java:138)
    at org.sonarsource.scanner.lib.internal.JavaRunnerFactory.createRunner(JavaRunnerFactory.java:85)
    at org.sonarsource.scanner.lib.internal.ScannerEngineLauncherFactory.createLauncher(ScannerEngineLauncherFactory
.java:53)
    at org.sonarsource.scanner.lib.ScannerEngineBootstrapper.bootstrap(ScannerEngineBootstrapper.java:118)
    at org.sonarsource.scanner.cli.Main.analyze(Main.java:75)
    at org.sonarsource.scanner.cli.Main.main(Main.java:63)
11:02:06.246 ERROR Re-run SonarScanner CLI using the -X switch to enable full debug logging.
PS C:\sonar-scanner\sonar-scanner-6.2.0.4584-windows-x64\bin> |

```

Dashboard &gt; All &gt;

Enter an item name

sonarqubetest1

» Required field

**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.

OK

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

Dashboard > SonarQube Pipeline > Configuration

Configure

General

Advanced Project Options

Pipeline

Pipeline

Definition

Pipeline script

Script ?

```
1 * node {
2 *   stage('Cloning the Github Repo') {
3 *     git 'https://github.com/shazforiot/MSBuild_firstproject.git'
4 *   }
5 *   stage('SonarQube analysis') {
6 *     withSonarQubeEnv('sonarqube-test1') {
7 *       bat "C:/sonar-scanner/sonar-scanner-6.2.0.4584-windows-x64/bin/sonar-scanner.bat \
8 *         -D sonar.login=admin \
9 *         -D sonar.password=admin123 \
10 *         -D sonar.projectKey=sonarqube-test1 \
11 *         -D sonar.exclusions=vendor/**,resources/**,*.java \
12 *         -D sonar.host.url=http://localhost:9080/"
13 *     }
14 *   }
15 }
```

☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

Save

Apply

Dashboard > SonarQube Pipeline >

**Status** **SonarQube Pipeline**

</> Changes

▷ Build Now

⚙️ Configure

🗑️ Delete Pipeline

↔️ Move

🔍 Full Stage View

🌊 SonarQube

📁 Stages

✎ Rename

❓ Pipeline Syntax

📊 Build History trend ▾

🔍 Filter... /

**Stage View**

Average stage times:  
(Average full run time: ~34s)

Stage	Duration	Status
Cloning the GitHub Repo	3s	Success
SonarQube analysis	30s	Success
Test	2s	Failed (133ms)
Deploy	9s	Failed (314ms)

📄 .git Logs

## Console Output

```
Started by user jai navani
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in C:\ProgramData\Jenkins\.jenkins\workspace\SonarQube Pipeline
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Cloning the GitHub Repo)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\SonarQube Pipeline\.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 2.45.2.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 f2bc042c04c6e72427c380bcae6d6fee7b49adf (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
> git.exe branch -a -v --no-abbrev # timeout=10
> git.exe branch -D master # timeout=10
> git.exe checkout -b master f2bc042c04c6e72427c380bcae6d6fee7b49adf # timeout=10
```

```

11:22:18.236 INFO Sensor C# File Caching Sensor [csharp]
11:22:18.237 WARN Incremental PR analysis: Could not determine common base path, cache will not be computed. Consider setting 'sonar.projectBaseDir'
property.
11:22:18.237 INFO Sensor C# File Caching Sensor [csharp] (done) | time=1ms
11:22:18.237 INFO Sensor Zero Coverage Sensor
11:22:18.251 INFO Sensor Zero Coverage Sensor (done) | time=14ms
11:22:18.256 INFO SCM Publisher SCM provider for this project is: git
11:22:18.257 INFO SCM Publisher 4 source files to be analyzed
11:22:18.789 INFO SCM Publisher 4/4 source files have been analyzed (done) | time=531ms
11:22:18.793 INFO CPD Executor Calculating CPD for 0 files
11:22:18.795 INFO CPD Executor CPD calculation finished (done) | time=0ms
11:22:18.810 INFO SCM revision ID 'f2bc042c04c6e72427c380bcaee6d6fee7b49adf'
11:22:19.074 INFO Analysis report generated in 134ms, dir size=201.0 kB
11:22:19.137 INFO Analysis report compressed in 45ms, zip size=22.5 kB
11:22:19.351 INFO Analysis report uploaded in 212ms
11:22:19.353 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=sonarqube-test1
11:22:19.354 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
11:22:19.354 INFO More about the report processing at http://localhost:9000/api/ce/task?id=971ae2f2-4e0f-49a7-88c4-ad4a6ccedd8
11:22:19.366 INFO Analysis total time: 24.819 s
11:22:19.368 INFO SonarScanner Engine completed successfully
11:22:19.454 INFO EXECUTION SUCCESS
11:22:19.455 INFO Total time: 29.696s
[Pipeline] }
[Pipeline] // withSonarQubeEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

The screenshot displays the SonarQube web interface for a project named 'sonarqube-test1'. The main section shows a 'Quality Gate' status of 'Passed' with a green checkmark. A warning message indicates 'The last analysis has warnings. See details'. The dashboard includes several metrics:

- Security:** 0 Open Issues, Grade A.
- Reliability:** 0 Open Issues, Grade A.
- Maintainability:** 0 Open Issues, Grade A.
- Accepted Issues:** 0.
- Coverage:** (Visual progress bar).
- Duplications:** 0.0%.

The interface also shows a 'main' branch selected, a 'Set as homepage' button, and a 'Version not provided' message.