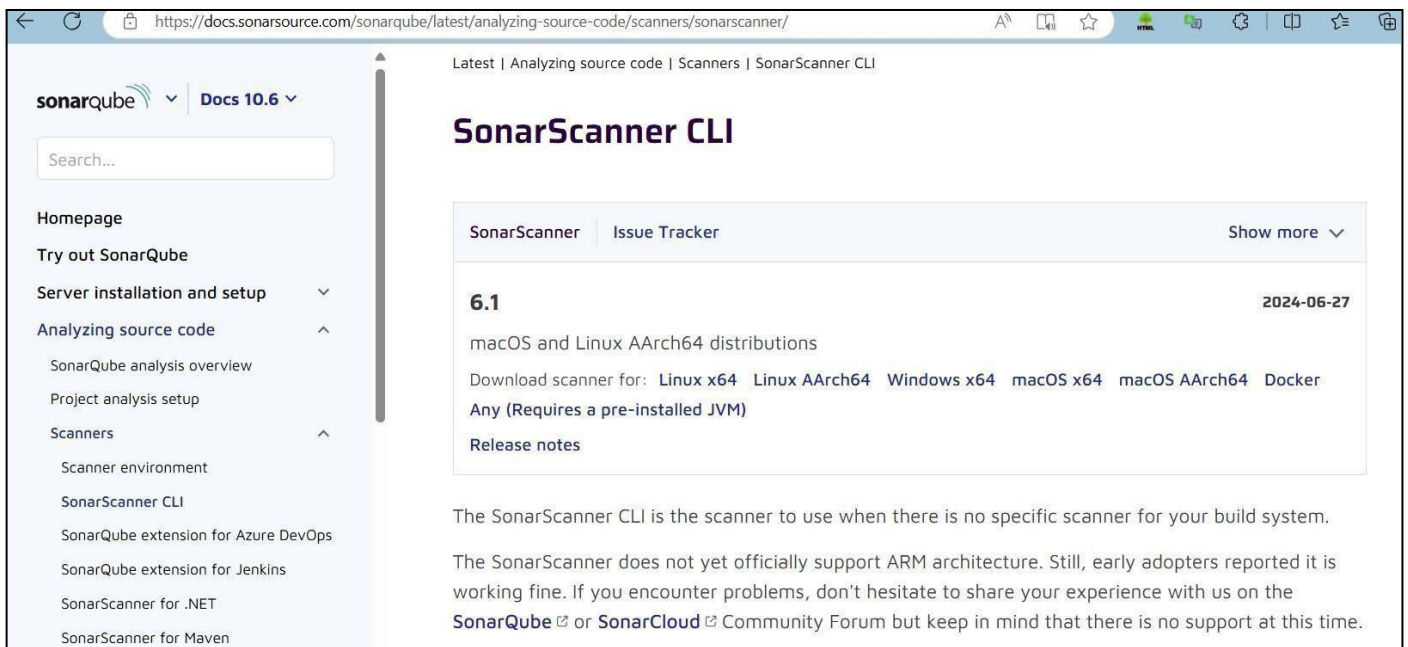


ADVANCE DEVOPS EXP 8

NAME: PRAJYOT SHINDE 57 / D15A

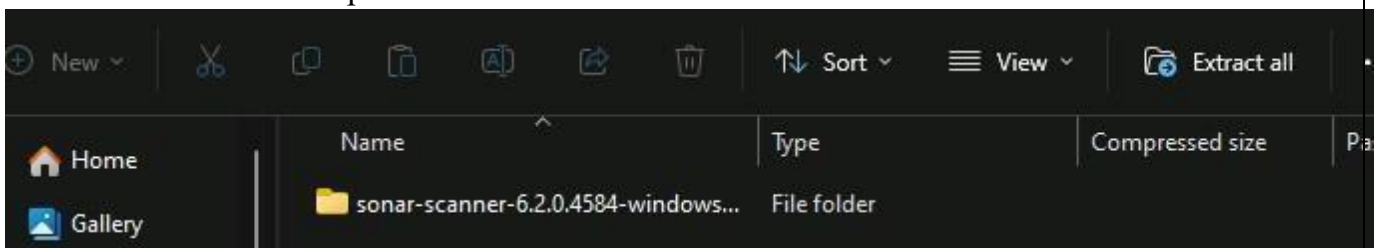
Aim: Create a Jenkins CICD Pipeline with SonarQube / GitLab Integration to perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities on a sample Web / Java / Python application.

Step 1: Download sonar scanner <https://docs.sonarsource.com/sonarqube/latest/analyzing-source-code/scanners/sonarscan>



[ner/](#) Visit this link and download the sonarqube scanner CLI.

Extract the downloaded zip file in a folder.

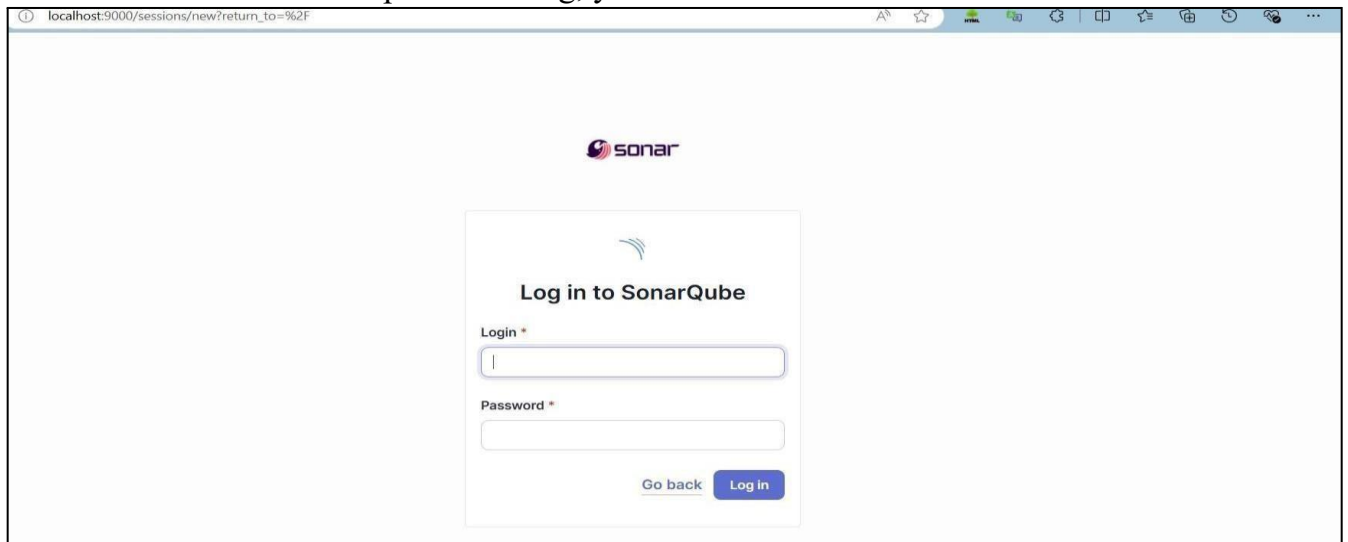


1. Install sonarqube image Command: **docker pull sonarqube**

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

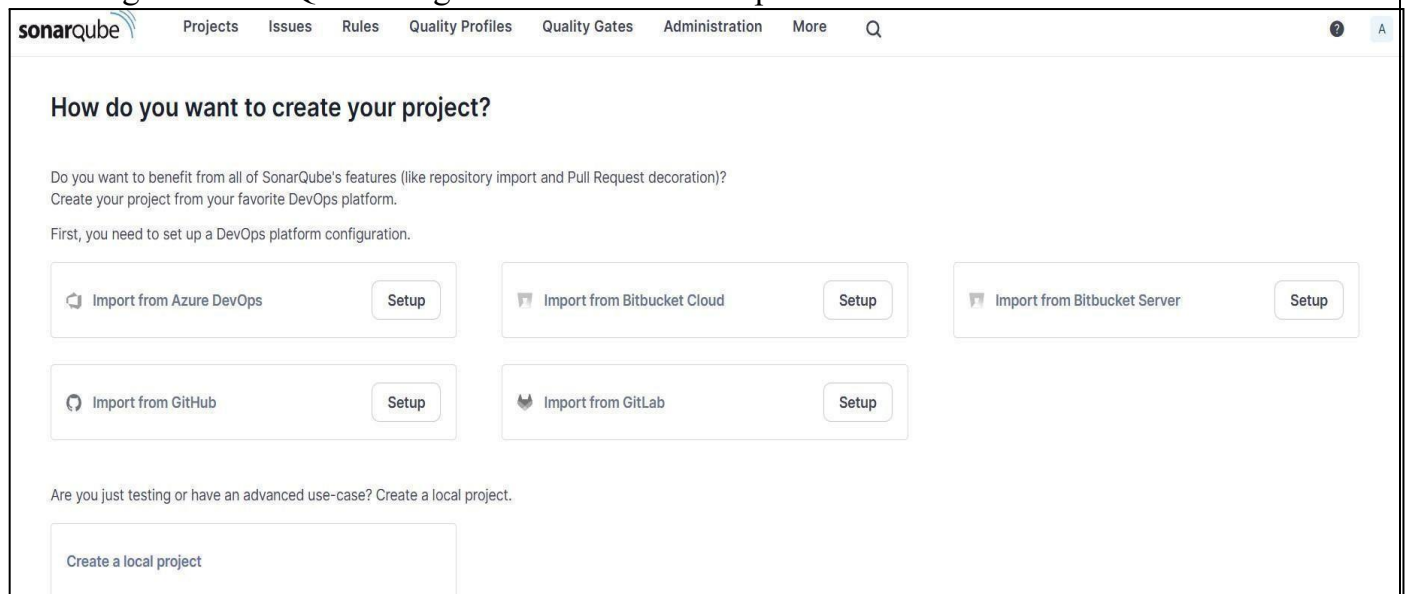
```
PS C:\Users\Soham Satpute> docker pull sonarqube
Using default tag: latest
latest: Pulling from library/sonarqube
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Image is up to date for sonarqube:latest
docker.io/library/sonarqube:latest
```

2. Once the container is up and running, you can check the status of



SonarQube at localhost port 9000.

3. Login to SonarQube using username admin and password admin.



4. Create a manual project in SonarQube with the name sonarqube

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](#)

Cancel

Next

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 new code. Learn more: [Defining New Code](#)

Choose the baseline for new code for this project

☒ Use the global setting

Previous version

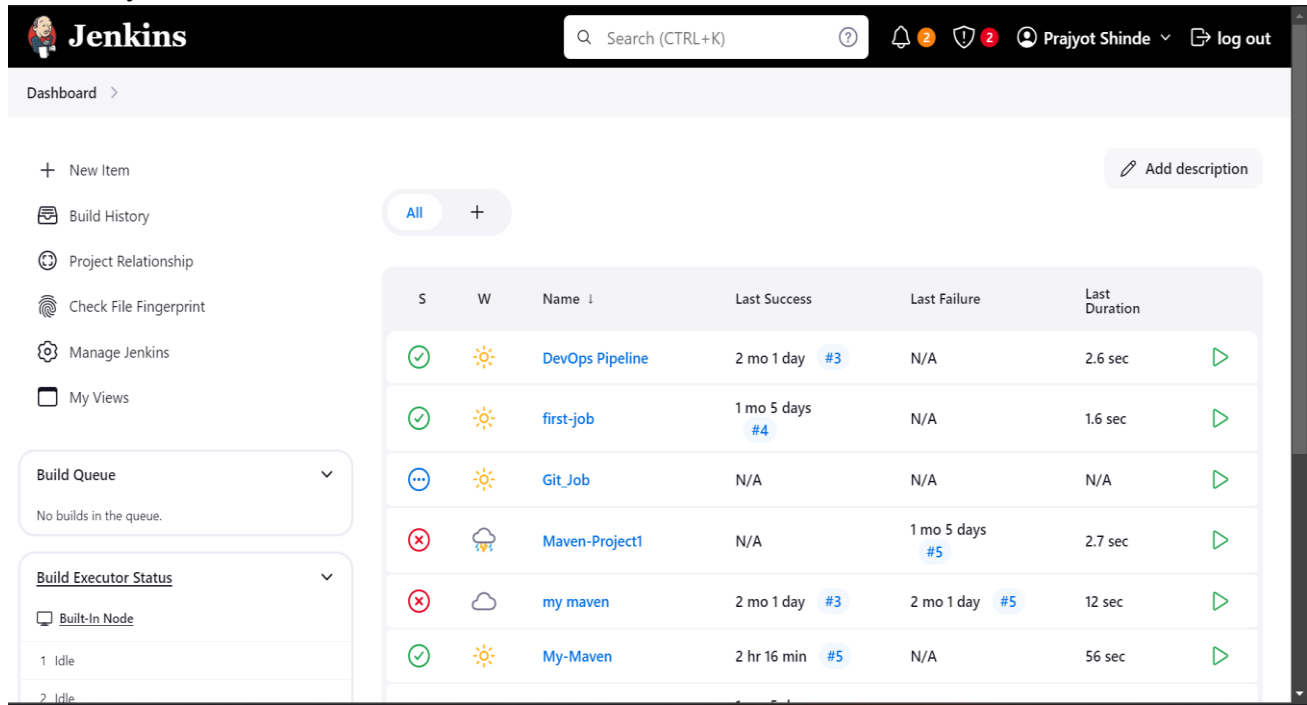
Any code that has changed since the previous version is considered new code.

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.

5. Open up Jenkins Dashboard on localhost, port 8080 or whichever port it is at for you.



The screenshot shows the Jenkins Dashboard. At the top, there's a search bar and a user profile for 'Prajyot Shinde'. The main content area displays a table of builds. On the left, there's a sidebar with navigation links like 'New Item', 'Build History', 'Project Relationship', 'Check File Fingerprint', 'Manage Jenkins', and 'My Views'. Below these are sections for 'Build Queue' (showing no builds) and 'Build Executor Status' (showing two idle executors).

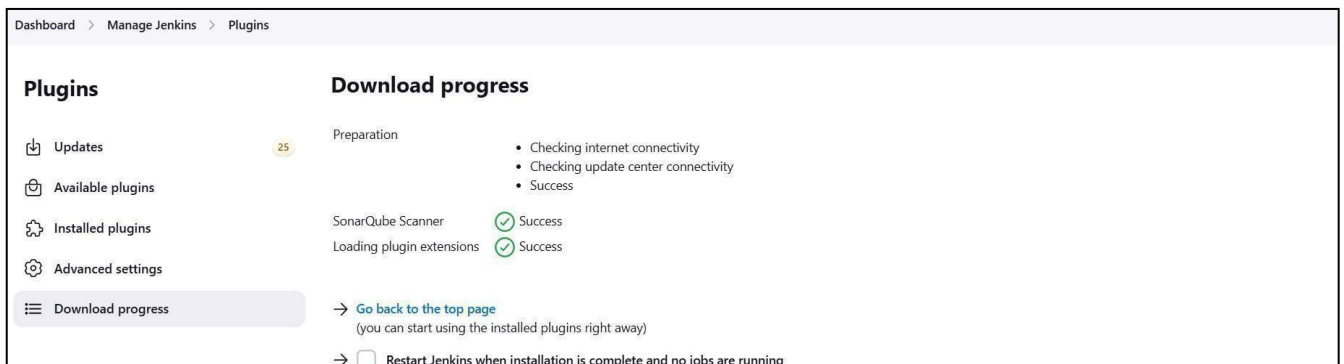
S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	DevOps Pipeline	2 mo 1 day #3	N/A	2.6 sec
✓	☀	first-job	1 mo 5 days #4	N/A	1.6 sec
⋯	☀	Git_Job	N/A	N/A	N/A
✗	☁	Maven-Project1	N/A	1 mo 5 days #5	2.7 sec
✗	☁	my maven	2 mo 1 day #3	2 mo 1 day #5	12 sec
✓	☀	My-Maven	2 hr 16 min #5	N/A	56 sec

6. Go to Manage Jenkins and search for SonarQube Scanner for Jenkins and install it.



The screenshot shows the 'Manage Jenkins > Plugins' page. A search bar at the top contains 'sonarq'. Below it, a table lists available plugins. The 'SonarQube Scanner' plugin is highlighted, showing its version (2.17.2) and release date (6 mo 29 days ago). The description states it allows easy integration of SonarQube for code quality inspection.

Install	Name	Released
<input type="checkbox"/>	SonarQube Scanner 2.17.2 External Site/Tool Integrations Build Reports This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.	6 mo 29 days ago



The screenshot shows the 'Download progress' page for the SonarQube Scanner plugin. It displays the progress of the installation, including 'Preparation' (checking internet and update center connectivity) and 'SonarQube Scanner' (loading plugin extensions). Both steps are marked as 'Success'. At the bottom, there are links to 'Go back to the top page' and a checkbox to 'Restart Jenkins when installation is complete and no jobs are running'.

Download progress

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

7. Under Jenkins 'Manage Jenkins' then go to 'system', scroll and look for SonarQube

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

Servers and enter the details.

Dashboard > Manage Jenkins > Tools

Add Git

Gradle installations

Add Gradle

SonarScanner for MSBuild installations

Add SonarScanner for MSBuild

SonarQube Scanner installations

Add SonarQube Scanner

Ant installations

Enter the Server Authentication token if needed.

In SonarQube installations: Under **Name** add <project name of sonarqube> for me
adv_devops_7_sonarqube

In **Server URL** Default is <http://localhost:9000>

8. Search for SonarQube Scanner under Global Tool Configuration.
Choose the latest configuration and choose Install automatically.

Dashboard > Manage Jenkins > Tools

Check the “Install automatically” option. → Under name any name as identifier → Check

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

Save

Apply

9. After configuration, create a New Item → choose a pipeline project.

New Item

Enter an item name

AdDevops-8

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.

OK

10. Under Pipeline script, enter the following:

```
node {  
  stage('Cloning the GitHub Repo') { git  
    'https://github.com/shazforiot/GOL.git'  
  } stage('SonarQube  
  
analysis') {  
  
  withSonarQubeEnv('<Name_of_SonarQube_environment_on_Jenk  
ins>') { sh """  
    <PATH_TO_SONARQUBE_SCANNER_FOLDER>/bin/sonar-scanner \  
    -D sonar.login=<SonarQube_USERNAME> \  
    -D sonar.password=<SonarQube_PASSWORD> \  
    -D sonar.projectKey=<Project_KEY> \  
    -D sonar.exclusions=vendor/**,resources/**,**/*.java \  
    -D sonar.host.url=<SonarQube_URL>(default: http://localhost:9000/)  
    """  
  }  
}  
}
```

It is a java sample project which has a lot of repetitions and issues that will be detected by SonarQube.

Search (CTRL+K)

Prajyot Shinde

log out

Dashboard > SonarQube-Pipeline >

Status

Changes

Build Now

Configure

Delete Pipeline

Full Stage View

SonarQube

Stages

Rename

Pipeline Syntax

Build History

SonarQube-Pipeline

Add description

Stage View

Average stage times:

(Average full run time: ~11min 56s)

	Cloning the GitHub Repo	SonarQube analysis	Declarative: Post Actions
#14 Sep 30 15:07 No Changes	1s	11min 54s	
#13 Sep 30 15:05 No Changes	1s	144ms failed	
#12 Sep 30 14:56 No Changes	1s	1min 29s	426ms

11.Check console

Search (CTRL+K)

Prajyot Shinde

log out

Dashboard > SonarQube-Pipeline > #14

Status

Changes

Console Output

View as plain text

Edit Build Information

Delete build '#14'

Timings

Git Build Data

Pipeline Overview

Pipeline Console

Replay

Pipeline Steps

Console Output

Skipping 4,248 KB.. Full Log

15:16:52.071 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/protocol/http/gui/HTTPArgumentsPanel.html for block at line 798. Keep only the first 100 references.

15:16:52.071 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/protocol/http/gui/HTTPArgumentsPanel.html for block at line 810. Keep only the first 100 references.

15:16:52.071 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/protocol/http/gui/HTTPArgumentsPanel.html for block at line 823. Keep only the first 100 references.

15:16:52.071 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/protocol/http/gui/HTTPArgumentsPanel.html for block at line 844. Keep only the first 100 references.

15:16:52.071 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/protocol/http/gui/HTTPArgumentsPanel.html for block at line 509. Keep only the first 100 references.

15:16:52.071 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/protocol/http/gui/HTTPArgumentsPanel.html for block at line 1065. Keep only the first 100 references.

15:16:52.071 WARN Too many duplication references on file gameoflife-

12.Now, check the project in SonarQube:

The screenshot shows the SonarQube interface for a project named 'sonarqube-test'. The main navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, and More. The project overview page displays the following information:

- main** branch: 683k Lines of Code, Version not provided. A button 'Set as homepage' is available.
- Quality Gate:** Passed (indicated by a green checkmark). Last analysis 8 hours ago.
- Warnings:** The last analysis has warnings. [See details](#)
- Code Quality Metrics:**
 - Security:** 0 Open issues (Grade A)
 - Reliability:** 68k Open issues (Grade C)
 - Maintainability:** 164k Open issues (Grade A)

13.code problems consistency:

The screenshot shows the 'Issues' page in SonarQube. The left sidebar contains filters for 'My Issues' and 'All', and a 'Clean Code Attribute' section with the following counts:

- Consistency: 197k
- Intentionality: 14k
- Adaptability: 0
- Responsibility: 0

The main area displays a list of issues for the file 'gameoflife-acceptance-tests/Dockerfile'. The issues are:

- Issue 1:** Use a specific version tag for the image. (Maintainability, L1, 5min effort, 4 years ago, Code Smell, Major)
- Issue 2:** Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. (Intentionality, L12, 5min effort, 4 years ago, Code Smell, Major)
- Issue 3:** Surround this variable with double quotes; otherwise, it can lead to unexpected behavior. (Intentionality, L12, 5min effort, 4 years ago, Code Smell, Major)

14.Intentionality:

The screenshot displays a software quality tool interface. On the left, a sidebar shows filters for 'My Issues' and 'All'. Under 'Filters', there's a 'Clear All Filters' button and a section for 'Issues in new code'. A 'Clean Code Attribute' filter is active, showing a list of attributes: Consistency (197k), Intentionality (14k), Adaptability (0), and Responsibility (0). The main panel shows a list of issues under the path 'gameoflife-acceptance-tests/Dockerfile'. The first issue is 'Use a specific version tag for the image.' with a 'Maintainability' tag. The second and third issues are 'Surround this variable with double quotes; otherwise, it can lead to unexpected behavior.' with an 'Intentionality' tag. The interface includes a 'Bulk Change' button, 'Select issues' and 'Navigate to issue' dropdowns, and a summary of '13,887 issues' and '59d effort'.

15.Bugs

The screenshot displays a software quality tool interface. On the left, a sidebar shows filters for 'Software Quality' and 'Severity'. Under 'Software Quality', there's a list of attributes: Security (0), Reliability (14k), and Maintainability (0). Under 'Severity', there's a list of types: Bug (14k), Vulnerability (0), and Code Smell (268). The main panel shows a list of issues under the path 'gameoflife-core/build/reports/tests/all-tests.html'. The first issue is 'Add "lang" and/or "xml:lang" attributes to this "<html>" element' with a 'Reliability' tag. The second issue is 'Add "<th>" headers to this "<table>." with a 'Reliability' tag. The interface includes a 'Bulk Change' button, 'Select issues' and 'Navigate to issue' dropdowns, and a summary of '13,619 issues' and '56d effort'. A yellow banner at the bottom states 'Embedded database should be used for evaluation purposes only'.

Code smells:

Sonarqube-test / main

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

Type 1 x

- Bug 14k
- Vulnerability 0
- Code Smell 253

Add to selection Ctrl + click

Scope

Status

Security Category

gameoflife-web/tools/jmeter/printable_docs/building.html

☐ Add an "alt" attribute to this image. Intentionality

Reliability accessibility wcag2-a

Open Not assigned L29 - 5min effort - 4 years ago - Code Smell - Minor

gameoflife-web/tools/jmeter/printable_docs/changes.html

☐ Add an "alt" attribute to this image. Intentionality

Reliability accessibility wcag2-a

Open Not assigned L31 - 5min effort - 4 years ago - Code Smell - Minor

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

Duplications:

Sonarqube-test / main

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

Coverage

Duplications

- Overview
- New Code
- Duplicated Lines 0
- Duplicated Blocks 0

Overall Code

Density 50.6%

Duplicated Lines 384,007

Duplications Overview

(Only showing data for the first 500 files)

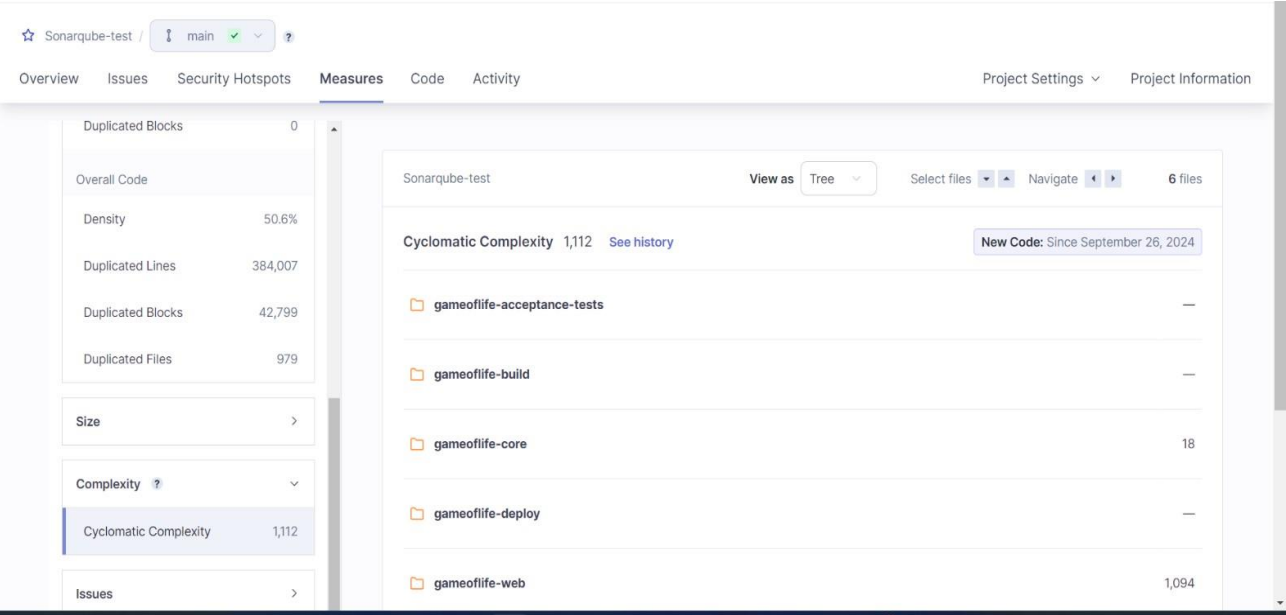
[See the data presented on this chart as a list](#)

Size: Duplicated Blocks

Zoom: 100%

Duplicated Lines

Cyclomatic Complexities:



In this way, we have integrated Jenkins with SonarQube for SAST.