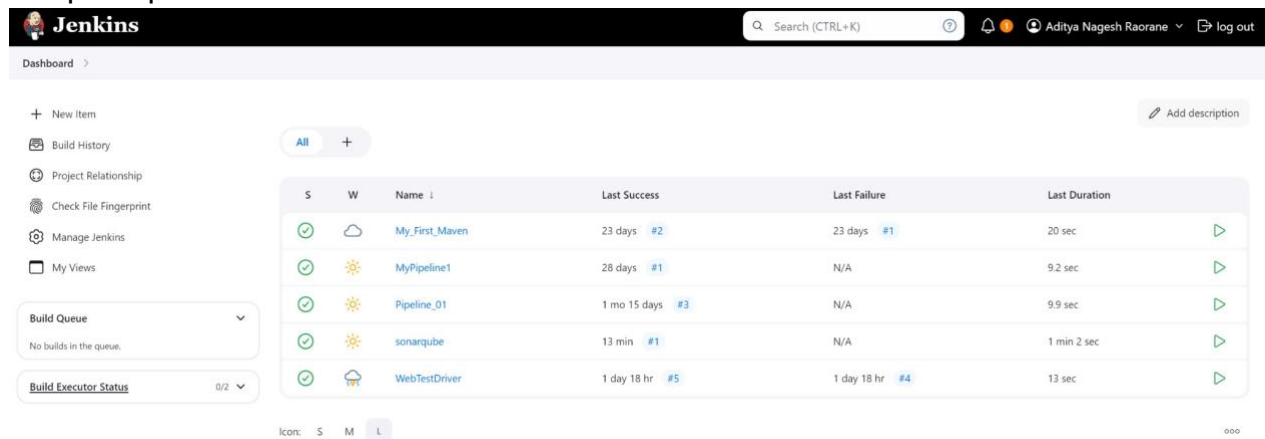


Aim: Create a Jenkins CICD Pipeline with SonarQube / GitLab Integration to perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities on a sample Web / Java / Python application.

1. Open up Jenkins Dashboard on localhost:8080.

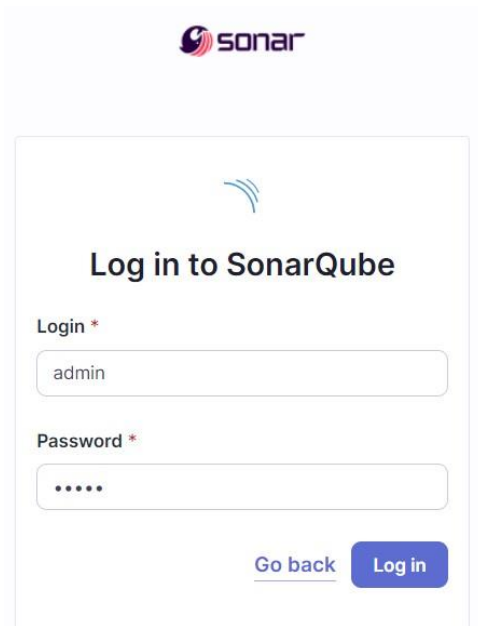


2. Run SonarQube in a Docker container using this command: a] `docker -v` b] `docker pull sonarqube` c] `docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest`

```
C:\Users\adity>docker -v
Docker version 27.0.3, build 7d4bcd8

C:\Users\adity>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: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecd
Status: Downloaded newer image for sonarqube:latest
4a6e73f4472de892b1ddead1abe77372a85a7b09408cce3a0abd37c5ab6b49a4
```

3. Once the container is up and running, you can check the status of SonarQube at **localhost port 9000**. The login id is **“admin”** and the password is **“aditya”**.



The image shows the SonarQube login interface. At the top is the Sonar logo. Below it is a large heading "Log in to SonarQube". There are two input fields: "Login *" with the text "admin" and "Password *" with masked characters. At the bottom right are two buttons: "Go back" (a link) and "Log in" (a button).

4. Create a local project in SonarQube with the name **sonarqube-test**.

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 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.

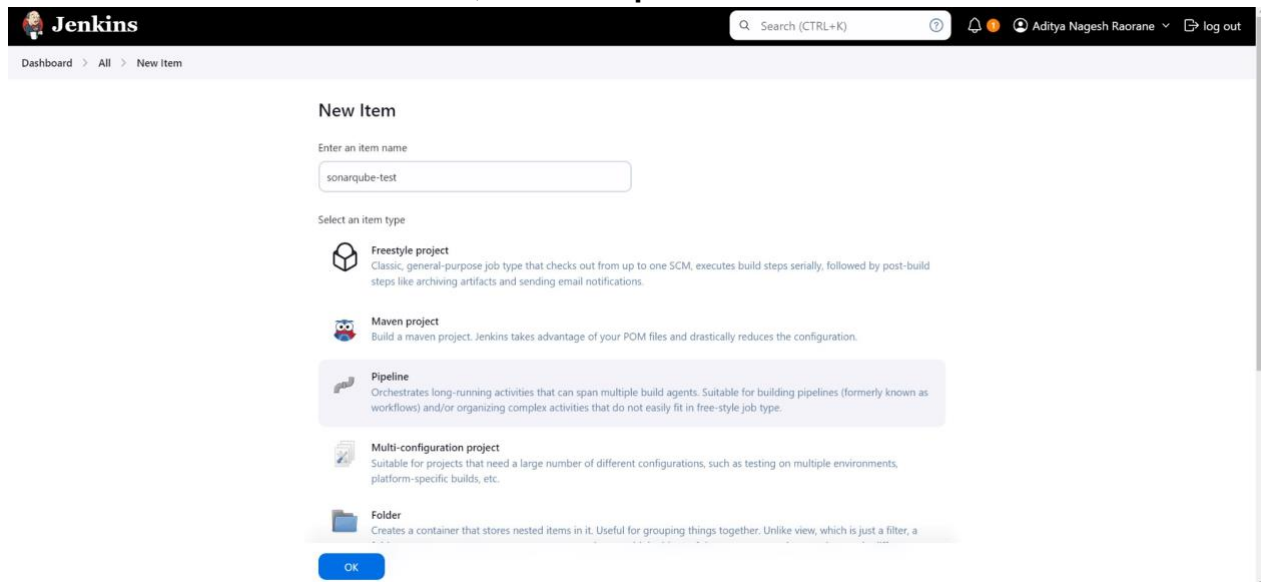
☐ Reference branch

Choose a branch as the baseline for the new code.
Recommended for projects using feature branches.

Back

Create project

Setup the project and come back to Jenkins Dashboard.

6. Create a New Item in Jenkins, choose **Pipeline**.7. Under **Pipeline Script**, enter the following -

```
node { stage('Cloning the GitHub
Repo')
{
    git 'https://github.com/shazforiot/GOL.git'
}
stage('SonarQube analysis') {
    withSonarQubeEnv('sonarqube') {
        bat
        "C:\\Users\\adity\\Downloads\\sonar-scanner-cli-6.1.0.4477-windows-x64\\sonar-s
canner-6.1.0.4477-windows-x64\\bin\\sonar-scanner.bat \
        -D sonar.login=<YOUR ID> \
        -D sonar.password=<YOUR PASSWORD> \
        -D sonar.projectKey=<YOUR PROJECT KEY> \
        -D sonar.exclusions=vendor/**,resources/**,**/*.java \
        -D sonar.host.url=http://localhost:9000/"
    }
}
}
```

The screenshot shows the Jenkins 'Configure' page for a pipeline named 'sonarqube-test'. The 'Pipeline' tab is selected. The 'Definition' dropdown is set to 'Pipeline script'. A code editor contains a Groovy script for a pipeline that clones a repository, sets environment variables, and runs a SonarQube analysis using the 'withSonarQubeEnv' step. The script includes parameters for SonarQube login, password, project key, and host URL. Below the editor, the 'Use Groovy Sandbox' checkbox is checked. At the bottom, there are 'Save' and 'Apply' buttons. The footer of the page shows 'REST API' and 'Jenkins 2.473'.


```
1 * node {
2   stage('Cloning the Github Repo') {
3     git 'https://github.com/shazforiot/600.git'
4   }
5   stage('SonarQube analysis') {
6     withSonarQubeEnv('sonarqube') {
7       bat "C:\Users\aditya\Downloads\sonar-scanner-cli-6.1.0.4477-windows-x64\sonar-scanner-6.1.0.4477-windows-x64\bin\sonar-scanner.bat \
8         -D sonar.login=admin \
9         -D sonar.password=aditya \
10        -D sonar.projectkey=sonarqube-test \
11        -D sonar.exclusions=vendor/**,resources/**,*/*.java \
12        -D sonar.host.url=http://localhost:9090/"
13     }
14   }
15 }
```

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

8. Run The Build.

The screenshot shows the Jenkins 'sonarqube-test' build configuration page. The 'Dashboard' breadcrumb is visible. The 'Status' tab is selected, showing a list of build configurations. Below the 'Status' tab, there are links for 'Changes' (represented by a code icon) and 'Build Now' (represented by a play button icon).

9. Check the console output once the build is complete.



Jenkins

Dashboard > sonarqube-test >

Status

</> Changes

▶ Build Now

⚙️ Configure

🗑️ Delete Pipeline

🔍 Full Stage View

🌊 SonarQube

📁 Stages

✎ Rename

❓ Pipeline Syntax

sonarqube-test

Stage View

Average stage times:
(Average full run time: ~14min 39s)

	Cloning the GitHub Repo	SonarQube analysis
#1	1s	14min 37s

Permalinks


- Last build (#1), 15 min ago
- Last stable build (#1), 15 min ago
- Last successful build (#1), 15 min ago
- Last completed build (#1), 15 min ago

Builds

Filter

Today

✓ #1 9:26 PM



Jenkins

Dashboard > sonarqube-test > #1

Status

</> Changes

Console Output

View as plain text

✓ Edit Build Information

⌚ Timings

🔗 Git Build Data

🔍 Pipeline Overview

📄 Pipeline Console

🗑️ Thread Dump

⏸️ Pause/resume

🔄 Replay

📋 Pipeline Steps

📁 Workspaces

Console Output

Started by user Aditya Nagesh Raorane

[Pipeline] Start of Pipeline

[Pipeline] node

Running on Jenkins in C:\ProgramData\Jenkins\jenkins\workspace\sonarqube-test

[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-test\.git # timeout=10

Fetching changes from the remote Git repository

> git.exe config remote.origin.url https://github.com/shazforiot/GOL.git # timeout=10

Fetching upstream changes from https://github.com/shazforiot/GOL.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/GOL.git +refs/heads/*:refs/remotes/origin/* # timeout=10

> git.exe rev-parse "refs/remotes/origin/master:{commit}" # timeout=10

Checking out Revision ba799ba7e1b576f04a4612322b0412c5e6e1e5e4 (refs/remotes/origin/master)

> git.exe config core.sparsecheckout # timeout=10

> git.exe checkout -f ba799ba7e1b576f04a4612322b0412c5e6e1e5e4 # timeout=10

> git.exe branch -a -v --no-abbrev # timeout=10

> git.exe branch -D master # timeout=10

> git.exe checkout -b master ba799ba7e1b576f04a4612322b0412c5e6e1e5e4 # timeout=10

Commit message: "Update Jenkinsfile"

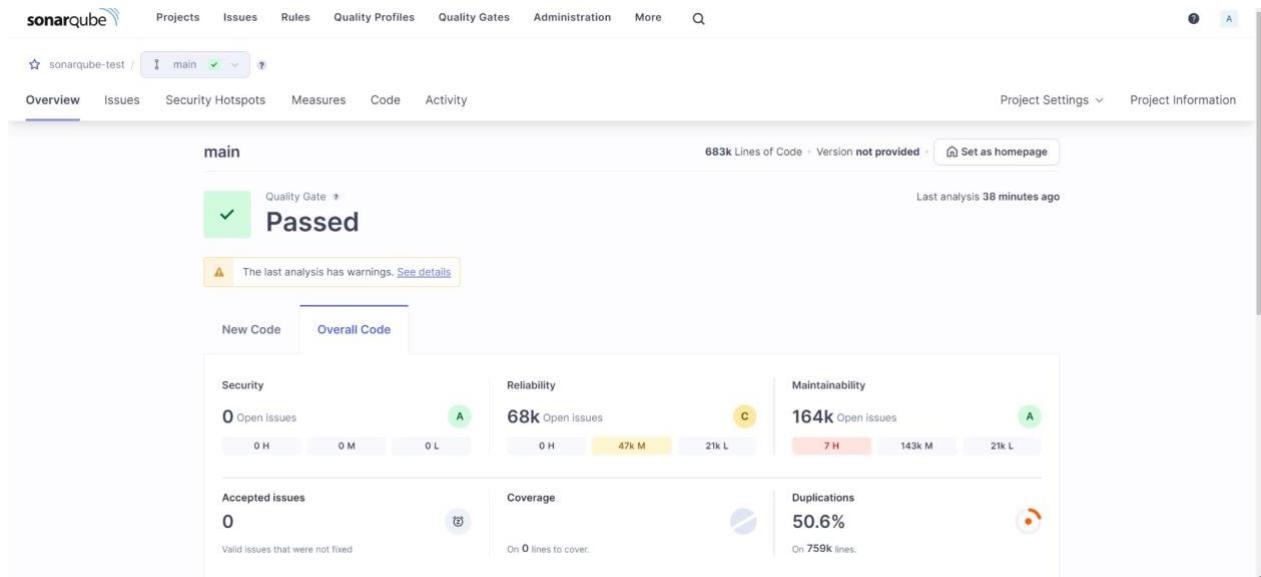
```
Dashboard > sonarqube-test > #1

line 41. Keep only the first 100 references.
21:37:59.929 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/gui/util/TextAreaCellRenderer.html for block at
line 17. Keep only the first 100 references.
21:37:59.929 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/gui/util/TextAreaCellRenderer.html for block at
line 296. Keep only the first 100 references.
21:37:59.929 WARN Too many duplication references on file gameoflife-web/tools/jmeter/docs/api/org/apache/jmeter/gui/util/TextAreaCellRenderer.html for block at
line 75. Keep only the first 100 references.
21:37:59.930 INFO CPD Executor CPD calculation finished (done) | time=153336ms
21:37:59.955 INFO SCM revision ID 'ba799ba7e1b576f04a4612322b0412c5e6e1e5e4'
21:40:14.276 INFO Analysis report generated in 5151ms, dir size=127.2 MB
21:40:35.678 INFO Analysis report compressed in 2138ms, zip size=29.6 MB
21:40:36.170 INFO Analysis report uploaded in 492ms
21:40:36.173 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=sonarqube-test
21:40:36.173 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
21:40:36.173 INFO More about the report processing at http://localhost:9000/api/ce/task?id=99fcd1e5-df4e-4f9f-8688-3169741e0856
21:40:53.466 INFO Analysis total time: 14:32.336 s
21:40:53.468 INFO SonarScanner Engine completed successfully
21:40:54.148 INFO EXECUTION SUCCESS
21:40:54.150 INFO Total time: 14:35.645s
[Pipeline] }
[Pipeline] // withSonarQubeEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

REST API Jenkins 2.473

10. After that, check the project in SonarQube.

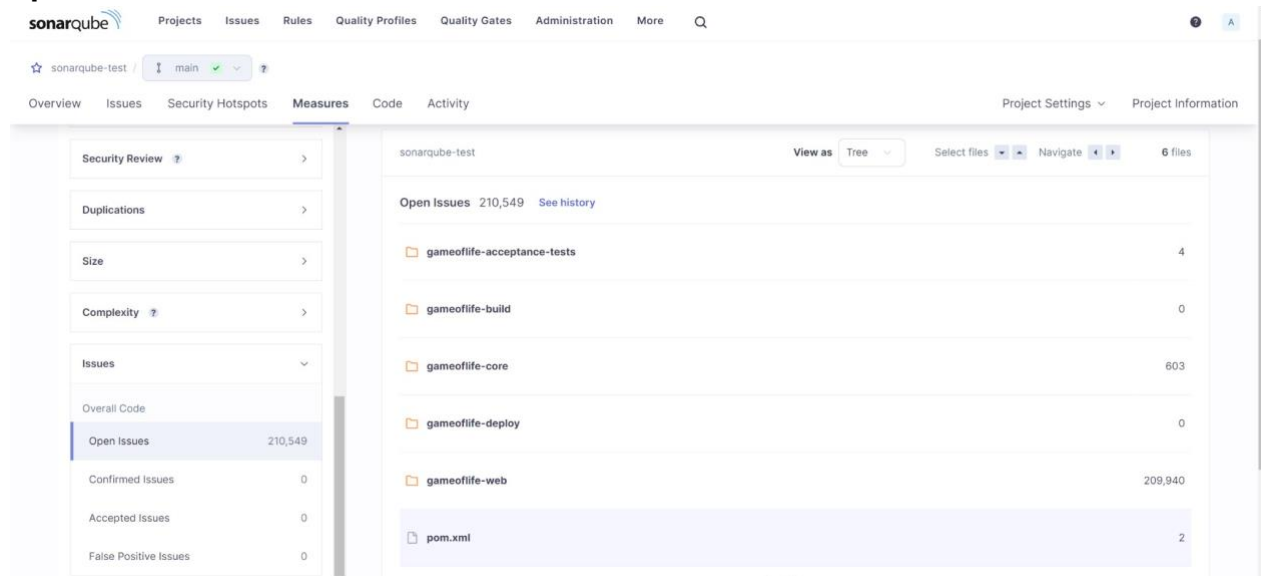
The screenshot displays the SonarQube web interface. On the left, there is a sidebar with navigation links: Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, and More. Below these links are filters for Quality Gate (Passed: 2, Failed: 0) and Reliability (A: 1, B: 0, C: 1, D: 0, E: 0). The main content area shows a list of projects. The first project is 'sonarqube PUBLIC', which has a 'Passed' status and a last analysis time of 1 hour ago. The second project is 'sonarqube-test PUBLIC', which also has a 'Passed' status and a last analysis time of 16 minutes ago. Below the project list, there is a detailed view for 'sonarqube-test PUBLIC' showing various metrics: Security (0), Reliability (68k), Maintainability (164k), Hotspots Reviewed (0.0%), Coverage (50.6%), and Duplications. The interface is clean and modern, with a light blue and white color scheme.



Under different tabs, check all different issues with the code.

11. Code Problems

Open Issues



Consistency

The screenshot displays the SonarQube web interface for a project named 'sonarqube-test'. The 'Issues' tab is active, showing a list of 196,662 issues with a total effort of 3075d. The left sidebar contains filters and a 'Clean Code Attribute' section with the following data:

Clean Code Attribute	Count
Consistency	197k
Intentionality	14k
Adaptability	0
Responsibility	0

The main content area shows three issues related to HTML attributes:

- Insert a <IDOCETYPE> declaration to before this <html> tag.** (Consistency, Reliability, user-experience, L1 - 5min effort - 4 years ago - Bug - Major)
- Remove this deprecated "width" attribute.** (Consistency, Maintainability, html5 obsolete, L9 - 5min effort - 4 years ago - Code Smell - Major)
- Remove this deprecated "align" attribute.** (Consistency, Maintainability, html5 obsolete, L11 - 5min effort - 4 years ago - Code Smell - Major)

A warning message at the bottom states: 'Embedded database should be used for evaluation purposes only'.

Intentionality

The screenshot displays the SonarQube web interface for the same project, 'sonarqube-test', but with the 'Intentionality' filter applied. The 'Issues' tab shows 13,887 issues with a total effort of 59d. The left sidebar's 'Clean Code Attribute' section is updated as follows:

Clean Code Attribute	Count
Consistency	197k
Intentionality	14k
Adaptability	0
Responsibility	0

The main content area shows three issues related to Dockerfile tags and variable quoting:

- Use a specific version tag for the image.** (Intentionality, Maintainability, No tags, L1 - 5min effort - 4 years ago - Code Smell - Major)
- Surround this variable with double quotes; otherwise, it can lead to unexpected behavior.** (Intentionality, Maintainability, No tags, L12 - 5min effort - 4 years ago - Code Smell - Major)
- Surround this variable with double quotes; otherwise, it can lead to unexpected behavior.** (Intentionality, Maintainability, No tags, L12 - 5min effort - 4 years ago - Code Smell - Major)

A warning message at the bottom states: 'Embedded database should be used for evaluation purposes only'.

Code Smells

sonarqube

Projects Issues Rules Quality Profiles Quality Gates Administration More Q

sonarqube-test / main

Overview Issues Security Hotspots Measures Code Activity

Project Settings Project Information

Severity

- High 0
- Medium 0
- Low 253

Type

- Bug 14k
- Vulnerability 0
- Code Smell 253

Add to selection Ctrl + click

Scope

Status

Security Categories

Bulk Change

Select issues Navigate to issue 253 issues 2d 5h effort

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

Add an "alt" attribute to this image.

Reliability Low

Intentionality

accessibility wcag2-a

Open Not assigned

L29 - 5min effort - 4 years ago - @ Code Smell - @ Minor

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

Add an "alt" attribute to this image.

Reliability Low

Intentionality

accessibility wcag2-a

Open Not assigned

L31 - 5min effort - 4 years ago - @ Code Smell - @ Minor

gameoflife-web/tools/meter/printable_docs/changes_history.html

Add an "alt" attribute to this image.

Intentionality

Embedded database should be used for evaluation purposes only

Bugs

sonarqube

Projects Issues Rules Quality Profiles Quality Gates Administration More Q

sonarqube-test / main

Overview Issues Security Hotspots Measures Code Activity

Project Settings Project Information

Severity

- High 0
- Medium 14k
- Low 0

Type

- Bug 14k
- Vulnerability 0
- Code Smell 253

Add to selection Ctrl + click

Scope

Status

Security Categories

Bulk Change

Select issues Navigate to issue 13,619 issues 56d effort

gameoflife-core/build/reports/tests/all-tests.html

Add "lang" and/or "xml:lang" attributes to this "<html>" element

Reliability Low

Intentionality

accessibility wcag2-a

Open Not assigned

L1 - 2min effort - 4 years ago - @ Bug - @ Major

gameoflife-core/build/reports/tests/allclasses-frame.html

Add "lang" and/or "xml:lang" attributes to this "<html>" element

Reliability Low

Intentionality

accessibility wcag2-a

Open Not assigned

L9 - 2min effort - 4 years ago - @ Bug - @ Major

gameoflife-core/build/reports/tests/allclasses-frame.html

Add "lang" and/or "xml:lang" attributes to this "<html>" element

Reliability Low

Intentionality

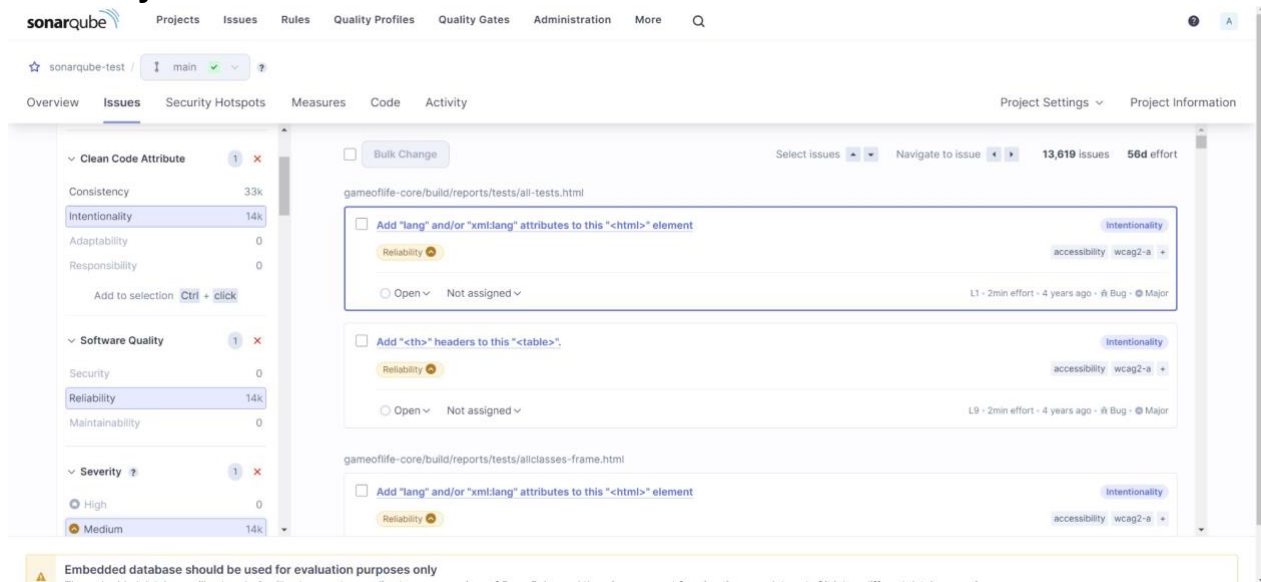
accessibility wcag2-a

Open Not assigned

L9 - 2min effort - 4 years ago - @ Bug - @ Major

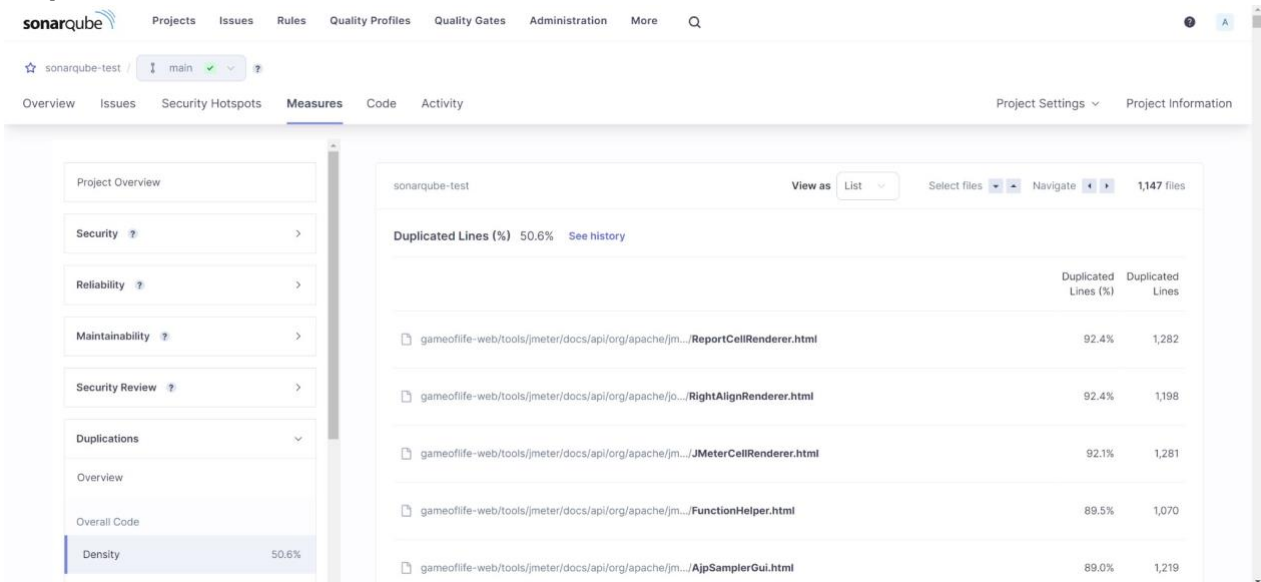
Embedded database should be used for evaluation purposes only

Reliability



The screenshot shows the SonarQube interface for the 'sonarqube-test' project, specifically the 'Issues' tab under the 'Reliability' category. The left sidebar displays a 'Clean Code Attribute' section with 'Intentionality' at 14k and 'Software Quality' at 0. The main area shows a list of issues, including 'Add "lang" and/or "xml:lang" attributes to this "<html>" element' and 'Add "<th>" headers to this "<table>"'. The bottom of the page features a yellow warning banner: 'Embedded database should be used for evaluation purposes only'.

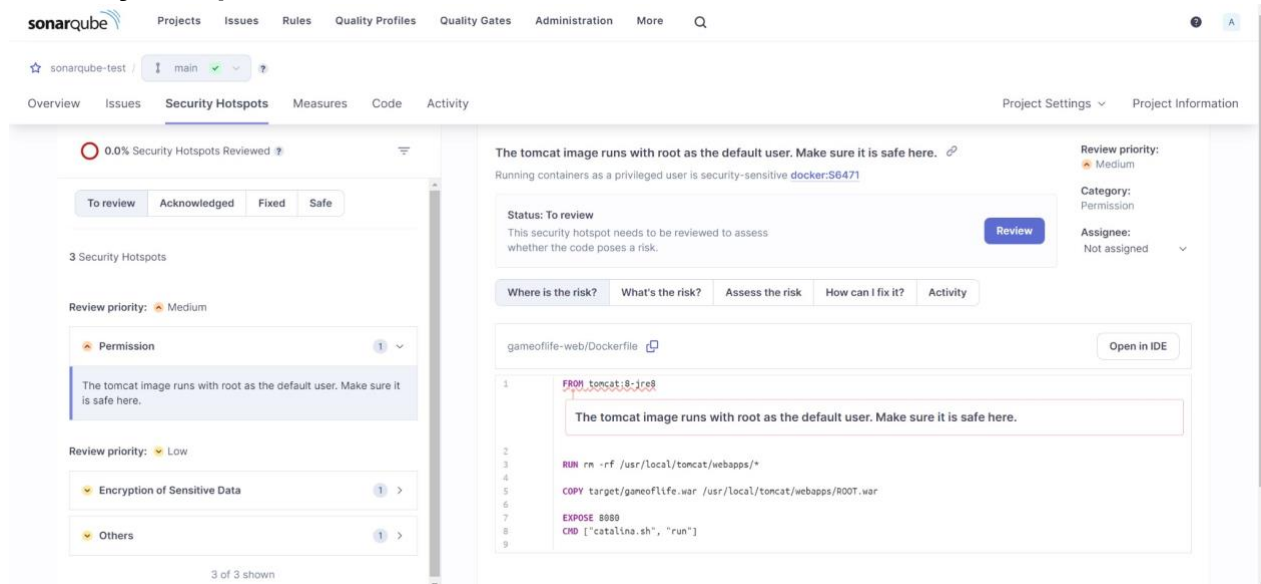
Duplicates



The screenshot shows the SonarQube interface for the 'sonarqube-test' project, specifically the 'Measures' tab under the 'Duplicates' category. The left sidebar displays a 'Project Overview' section with 'Security' at 0, 'Reliability' at 0, 'Maintainability' at 0, and 'Security Review' at 0. The main area shows a table of duplicated lines, including files like 'ReportCellRenderer.html', 'RightAlignRenderer.html', 'JMeterCellRenderer.html', 'FunctionHelper.html', and 'AjpSamplerGui.html'.

File	Duplicated Lines (%)	Duplicated Lines
gameoflife-web/tools/jmeter/docs/api/org/apache/jm.../ReportCellRenderer.html	92.4%	1,282
gameoflife-web/tools/jmeter/docs/api/org/apache/jo.../RightAlignRenderer.html	92.4%	1,198
gameoflife-web/tools/jmeter/docs/api/org/apache/jm.../JMeterCellRenderer.html	92.1%	1,281
gameoflife-web/tools/jmeter/docs/api/org/apache/jm.../FunctionHelper.html	89.5%	1,070
gameoflife-web/tools/jmeter/docs/api/org/apache/jm.../AjpSamplerGui.html	89.0%	1,219

Security Hotspot



The screenshot shows the SonarQube interface for a project named 'sonarqube-test'. The 'Security Hotspots' tab is active, displaying a summary of 3 security hotspots. The first hotspot is titled 'The tomcat image runs with root as the default user. Make sure it is safe here.' and is categorized as 'Permission' with a 'Medium' review priority. The second hotspot is 'Encryption of Sensitive Data' with a 'Low' review priority. The third hotspot is 'Others'. The interface also shows a list of hotspots with their status (To review, Acknowledged, Fixed, Safe) and a 'Review' button. The 'Where is the risk?' tab is selected, showing the location of the hotspot in the code (gameoflife-web/Dockerfile).

0.0% Security Hotspots Reviewed

To review Acknowledged Fixed Safe

3 Security Hotspots

Review priority: Medium

Permission

The tomcat image runs with root as the default user. Make sure it is safe here.

Review priority: Low

Encryption of Sensitive Data

Others

3 of 3 shown

The tomcat image runs with root as the default user. Make sure it is safe here.

Running containers as a privileged user is security-sensitive docker:S6471

Status: To review

This security hotspot needs to be reviewed to assess whether the code poses a risk.

Review

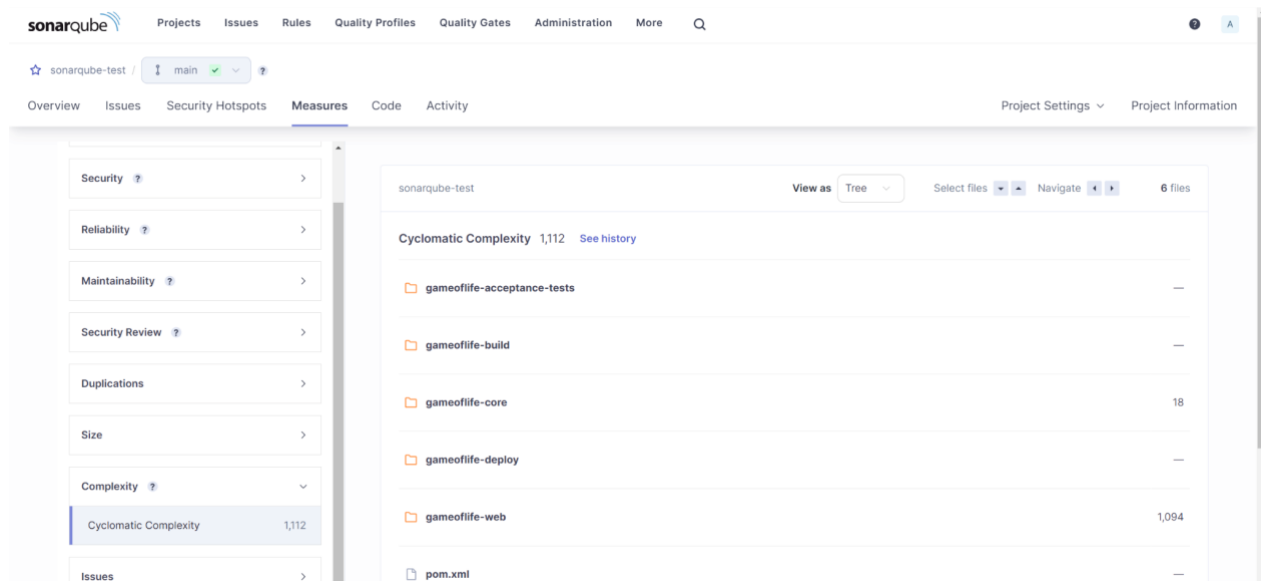
Where is the risk? What's the risk? Assess the risk How can I fix it? Activity

gameoflife-web/Dockerfile

Open in IDE

```
1 FROM tomcat:8-jre8
2
3 RUN rm -rf /usr/local/tomcat/webapps/*
4
5 COPY target/gameoflife.war /usr/local/tomcat/webapps/ROOT.war
6
7 EXPOSE 8080
8 CMD ["catalina.sh", "run"]
9
```

Cyclomatic Complexity



The screenshot shows the SonarQube interface for a project named 'sonarqube-test'. The 'Measures' tab is active, displaying a list of measures. The 'Cyclomatic Complexity' measure is selected, showing a value of 1,112. The interface also shows a list of files with their cyclomatic complexity values: gameoflife-acceptance-tests (—), gameoflife-build (—), gameoflife-core (18), gameoflife-deploy (—), gameoflife-web (1,094), and pom.xml (—).

Security

Reliability

Maintainability

Security Review

Duplications

Size

Complexity

Cyclomatic Complexity 1,112

Issues

sonarqube-test

View as Tree

Select files

Navigate

6 files

Cyclomatic Complexity 1,112 See history

gameoflife-acceptance-tests

gameoflife-build

gameoflife-core 18

gameoflife-deploy

gameoflife-web 1,094

pom.xml

In this way, we have created a CI/CD Pipeline with Jenkins and integrated it with SonarQube to find issues in the code like bugs, code smells, duplicates, cyclomatic complexities, etc.

Conclusion:

In this experiment, we performed a static analysis of the code to detect bugs, code smells, and security vulnerabilities on our sample Java application.