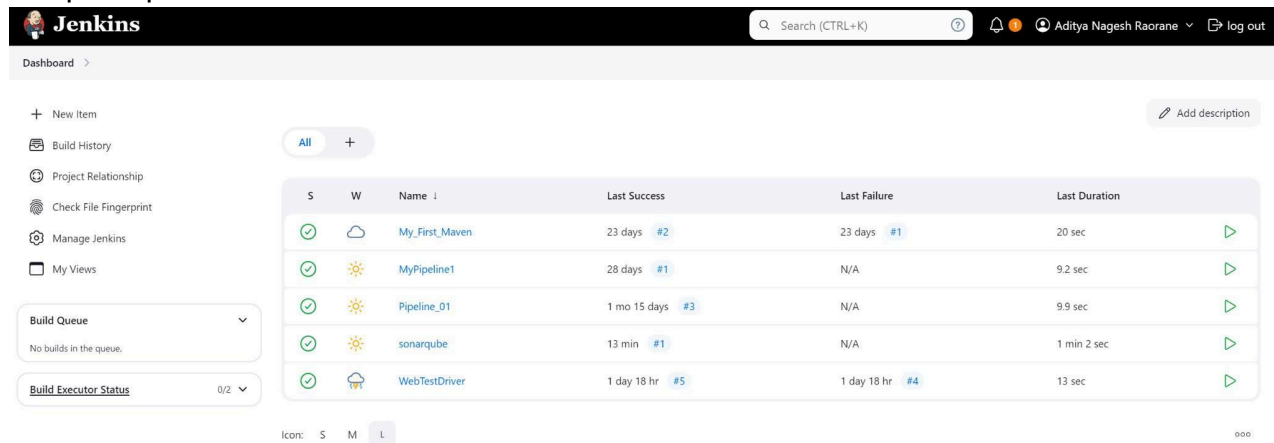


Aim: Create a Jenkins CICD Pipeline with SonarQube / GitLab Integration to perform a static analysis of the code to detect bugs, code smells, and security vulnerabilities on a sample Web / Java / Python application.

1. Open up Jenkins Dashboard on localhost:8080.



The screenshot shows the Jenkins Dashboard interface. On the left, there is a sidebar with navigation links: New Item, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, and My Views. Below these are sections for 'Build Queue' (showing 'No builds in the queue') and 'Build Executor Status' (showing '0/2'). The main area displays a table of builds with columns: S (Status), W (Icon), Name, Last Success, Last Failure, and Last Duration. The table lists five builds: 'My_First_Maven', 'MyPipeline1', 'Pipeline_01', 'sonarqube', and 'WebTestDriver'. Each build has a status icon (green checkmark or yellow sun) and a duration. A search bar and a user profile 'Aditya Nagesh Raorane' are visible at the top right.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☁	My_First_Maven	23 days #2	23 days #1	20 sec
✓	☀	MyPipeline1	28 days #1	N/A	9.2 sec
✓	☀	Pipeline_01	1 mo 15 days #3	N/A	9.9 sec
✓	☀	sonarqube	13 min #1	N/A	1 min 2 sec
✓	☁	WebTestDriver	1 day 18 hr #5	1 day 18 hr #4	13 sec

2. Run SonarQube in a Docker container using this

command: a) `docker -v`

b) `docker pull sonarqube`

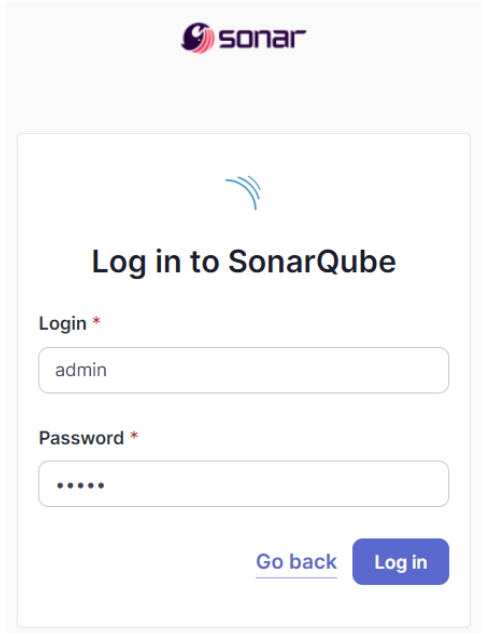
c) `docker run -d --name sonarqube -e`

`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



The image shows the SonarQube login page. At the top is the Sonar logo. Below it is a blue icon of three curved lines. The main heading is "Log in to SonarQube". There are two input fields: "Login *" with the value "admin" and "Password *" with five dots. At the bottom, there is a "Go back" link and a "Log in" button.

“**aditya**”.

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

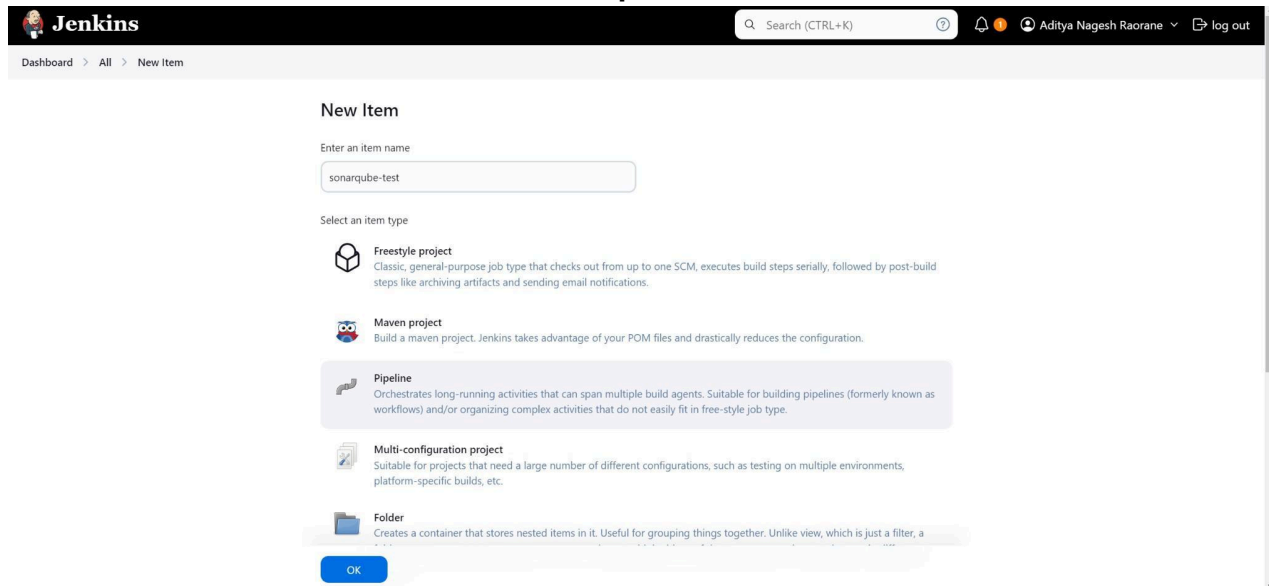
Choose a branch as the baseline for the new code.
Recommended for projects using feature branches.

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-
            scanner-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/"
        }
    }
}
```

Dashboard > sonarqube-test > Configuration

Configure

- General
- Advanced Project Options**
- Pipeline

Pipeline

Definition

Pipeline script

```
1 node {
2   stage("Cloning the Github Repo") {
3     git "https://github.com/shazforiot/sox.git"
4   }
5   stage("sonarqube analysis") {
6     withSonarQubeEnv("sonarqube") {
7       bat "cd \\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:9000/"
13     }
14   }
15 }
```

☒ Use Groovy Sandbox ?


[Pipeline Syntax](#)

[Save](#) [Apply](#)

REST API Jenkins 2.473

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


8. Run The Build.



Dashboard > sonarqube-test >

- Status**
- [Changes](#)
- [Build Now](#)

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
1s	14min 37s

#1
Sep 17 21:26 No Changes

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

Console Output

Download Copy View as plain text

Skipping 4,252 KB.. [Full Log](#)

20:16:42.864 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.
20:16:42.864 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.
20:16:42.864 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.
20:16:42.864 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.
20:16:42.865 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.
20:16:42.865 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.
20:16:42.865 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 776. Keep only the first 100 references.
20:16:42.865 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 778. Keep only the first 100 references.
20:16:42.865 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 530. Keep only the first 100 references.
20:16:42.865 WARN Too many duplication references on file gameoflife-

```
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 SCH revision ID 'ba7990a7e1b576f04a4612322b0412c5e6e1e5e4'
21:40:14.276 INFO Analysis report generated in 5151ms, dir size=127.2 MB
21:40:35.678 INFO Analysis report compressed in 21388ms, 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. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, and More. The left sidebar shows filters for Quality Gate (2 Passed, 0 Failed) 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' with a 'Passed' status. The second project is 'sonarqube-test PUBLIC', also with a 'Passed' status. Below the project name, it shows 'Last analysis: 16 minutes ago' and '683k Lines of Code · HTML, XML, ...'. A summary bar at the bottom of the project details shows: Security (A 0), Reliability (C 68k), Maintainability (A 164k), Hotspots Reviewed (E 0.0%), Coverage (50.6%), and Duplications (50.6%).

The screenshot shows the SonarQube Overview page for the 'sonarqube-test' project on the 'main' branch. The Quality Gate is 'Passed' (green checkmark). The last analysis was 38 minutes ago. A warning indicates the last analysis has warnings. The page displays various metrics:

- Security:** 0 Open Issues (Grade A)
- Reliability:** 68k Open Issues (Grade C)
- Maintainability:** 164k Open Issues (Grade A)
- Accepted Issues:** 0
- Coverage:** On 0 lines to cover.
- Duplications:** 50.6% (On 759k lines)

Navigation tabs include Overview, Issues, Security Hotspots, Measures, Code, and Activity. Project Settings and Project Information links are also present.

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

11. Code Problems - Open

The screenshot shows the SonarQube Measures page for the 'sonarqube-test' project on the 'main' branch. The 'Measures' tab is selected. The page displays a list of measures and their values:

Measure	Value
Open Issues	210,549
gameoflife-acceptance-tests	4
gameoflife-build	0
gameoflife-core	603
gameoflife-deploy	0
gameoflife-web	209,940
pom.xml	2

Navigation tabs include Overview, Issues, Security Hotspots, Measures, Code, and Activity. Project Settings and Project Information links are also present.

Issues

Consistency

The screenshot displays the SonarQube web interface for a project named 'sonarqube-test'. The 'Issues' tab is active, showing a list of issues under the 'Consistency' category. The left sidebar contains filters and a 'Clean Code Attribute' section with a table listing attributes: Consistency (197k), Intentionality (14k), Adaptability (0), and Responsibility (0). The main panel shows three issues:

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

A warning banner 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, now showing 'Intentionality' issues. The left sidebar is identical to the previous screenshot. The main panel shows three issues:

- Use a specific version tag for the image.** (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.** (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.** (Maintainability, No tags, L12, 5min effort, 4 years ago, Code Smell, Major)

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

Code Smells

The screenshot shows the SonarQube interface for a project named 'sonarqube-test'. The 'Issues' tab is active, displaying a list of 253 issues. The left sidebar shows filters for Severity (High, Medium, Low) and Type (Bug, Vulnerability, Code Smell). The main area displays three issues, all of type 'Code Smell' and severity 'Low'. Each issue has a description, a 'Bulk Change' button, and a 'Reliability' button. The issues are related to adding an 'alt' attribute to images in various HTML files. A warning message at the bottom states: 'Embedded database should be used for evaluation purposes only'.

Severity

- High: 0
- Medium: 0
- Low: 253

Type

- Bug: 14k
- Vulnerability: 0
- Code Smell: 253

Issues

- gameoflife-web/tools/meter/printable_docs/building.html: Add an "alt" attribute to this image. (Intentionality, Reliability, accessibility wcag2-a)
- gameoflife-web/tools/meter/printable_docs/changes.html: Add an "alt" attribute to this image. (Intentionality, Reliability, accessibility wcag2-a)
- gameoflife-web/tools/meter/printable_docs/changes_history.html: Add an "alt" attribute to this image. (Intentionality, Reliability, accessibility wcag2-a)

Embedded database should be used for evaluation purposes only

Bugs

The screenshot shows the SonarQube interface for a project named 'sonarqube-test'. The 'Issues' tab is active, displaying a list of 13,619 issues. The left sidebar shows filters for Severity (High, Medium, Low) and Type (Bug, Vulnerability, Code Smell). The main area displays three issues, all of type 'Bug' and severity 'Major'. Each issue has a description, a 'Bulk Change' button, and a 'Reliability' button. The issues are related to adding 'lang' and 'xml:lang' attributes to HTML elements and adding 'cth>' headers to HTML tables. A warning message at the bottom states: 'Embedded database should be used for evaluation purposes only'.

Severity

- High: 0
- Medium: 14k
- Low: 0

Type

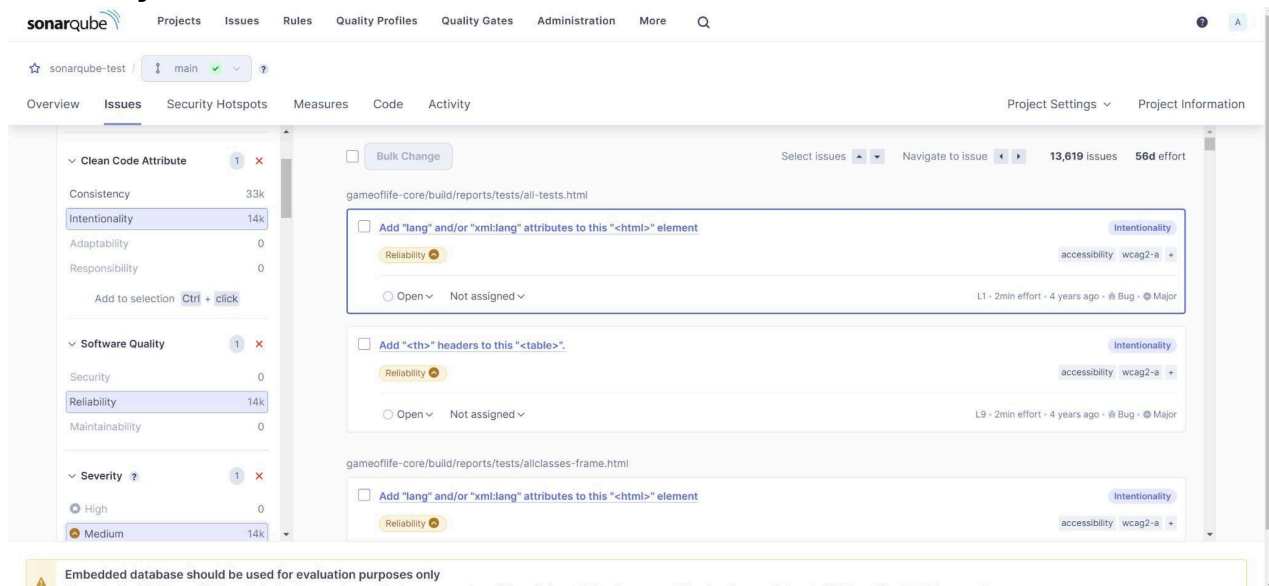
- Bug: 14k
- Vulnerability: 0
- Code Smell: 253

Issues

- gameoflife-core/build/reports/tests/all-tests.html: Add "lang" and/or "xml:lang" attributes to this "<html>" element. (Intentionality, Reliability, accessibility wcag2-a)
- gameoflife-core/build/reports/tests/all-classes-frame.html: Add "cth>" headers to this "<table>". (Intentionality, Reliability, accessibility wcag2-a)
- gameoflife-core/build/reports/tests/all-classes-frame.html: Add "lang" and/or "xml:lang" attributes to this "<html>" element. (Intentionality, Reliability, accessibility wcag2-a)

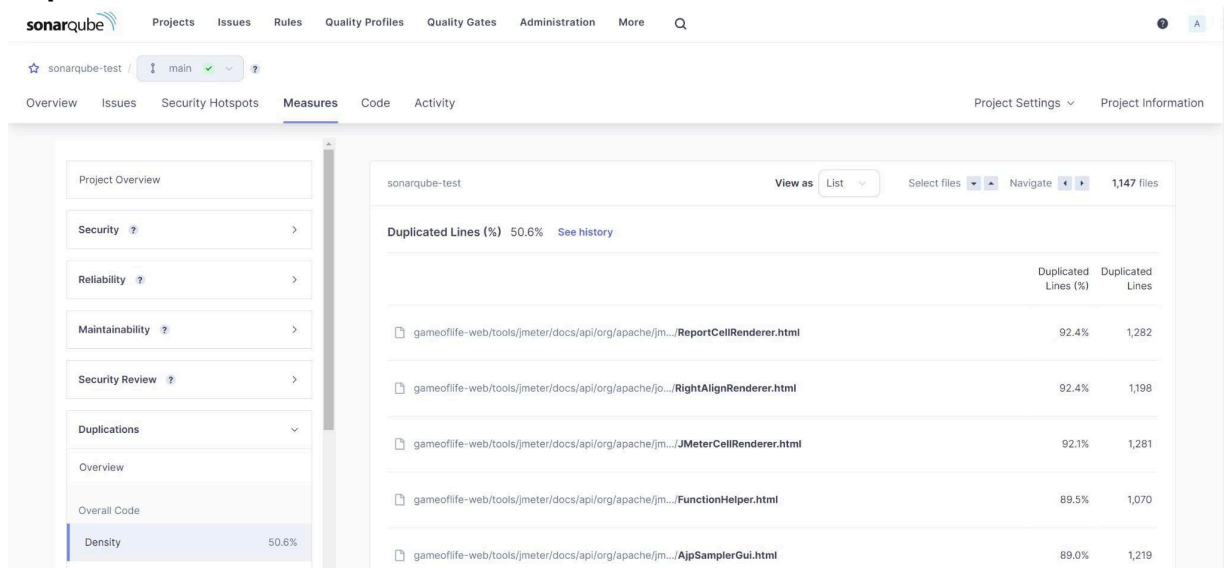
Embedded database should be used for evaluation purposes only

Reliability



The screenshot shows the SonarQube interface for the 'sonarqube-test' project, specifically the 'Issues' tab. The left sidebar displays a 'Clean Code Attribute' section with 'Intentionality' at 14k, and a 'Software Quality' section with 'Reliability' at 14k. 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>"'. A yellow banner at the bottom states: 'Embedded database should be used for evaluation purposes only'.

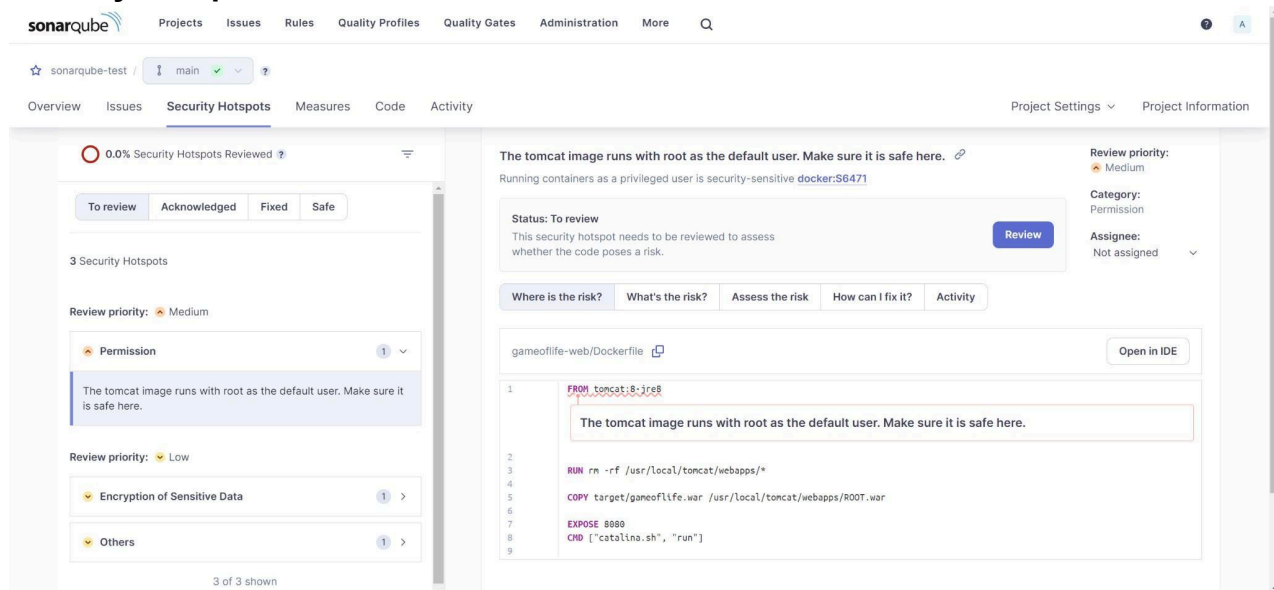
Duplicates



The screenshot shows the SonarQube interface for the 'sonarqube-test' project, specifically the 'Measures' tab. The left sidebar displays a 'Project Overview' section with 'Density' at 50.6%. The main area shows a table of duplicated lines with columns for 'Duplicated Lines (%)' and 'Duplicated Lines'. The table lists several files with their respective duplication percentages and line counts.

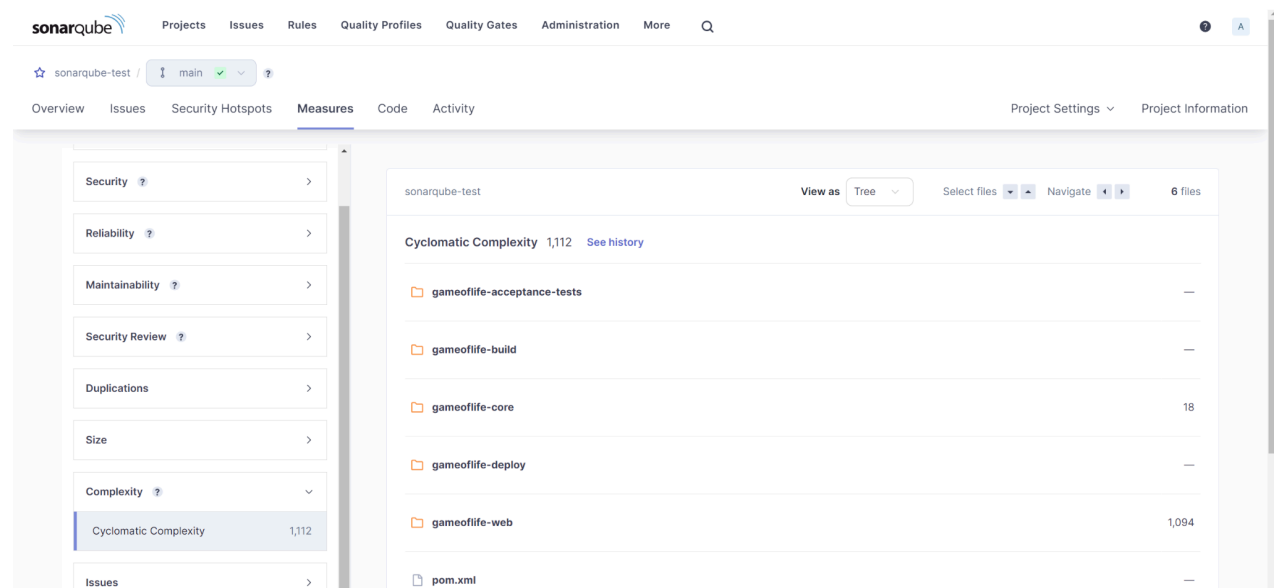
File	Duplicated Lines (%)	Duplicated Lines
gameoflife-web/tools/jmeter/docs/api/org/apache/jm.../ReportCellRenderer.html	92.4%	1,282
gameoflife-web/tools/jmeter/docs/api/org/apache/jm.../RightAlignRenderer.html	92.4%	1,198
gameoflife-web/tools/jmeter/docs/api/org/apache/jm.../JMeterCellRenderer.html	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 status is 'To review'. The interface includes filters for 'To review', 'Acknowledged', 'Fixed', and 'Safe'. A list of hotspots shows 'Permission' (1), 'Encryption of Sensitive Data' (1), and 'Others' (1). On the right, a detailed view of the selected hotspot is shown, including a code snippet from 'gameoflife-web/Dockerfile' that runs 'FROM tomcat:8-jre8' and 'RUN rm -rf /usr/local/tomcat/webapps/*'. The code also includes 'COPY target/gameoflife.war /usr/local/tomcat/webapps/ROOT.war', 'EXPOSE 8080', and 'CMD ["catalina.sh", "run"]'. The interface also has tabs for 'Where is the risk?', 'What's the risk?', 'Assess the risk', 'How can I fix it?', and 'Activity'.

Cyclomatic Complexity



The screenshot shows the SonarQube interface for the same project '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 includes a sidebar with a list of measures: 'Security', 'Reliability', 'Maintainability', 'Security Review', 'Duplications', 'Size', 'Complexity', and 'Issues'. The 'Complexity' measure is expanded, showing 'Cyclomatic Complexity' with a value of 1,112. The main area shows a tree view of the project structure with 6 files: 'gameoflife-acceptance-tests', 'gameoflife-build', 'gameoflife-core' (18), 'gameoflife-deploy', 'gameoflife-web' (1,094), and '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.