

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:In your command prompt to ensure whether the docker is installed or not.

```
C:\Users\Dell>docker -v
Docker version 27.1.1, build 6312585
```

STEP 2:Run docker login command and add your username and password for docker.

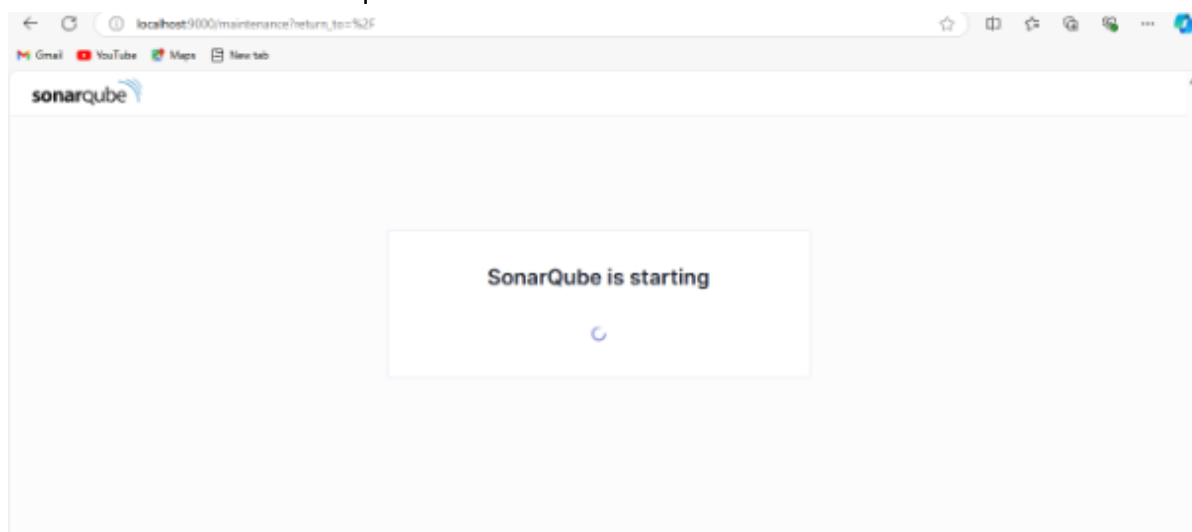
```
C:\Users\Dell>docker pull sonarqube
Using default tag: latest
7478e0ac0f23: Pull complete
90a925ab929a: Pull complete
7d9a34308537: Pull complete
80338217a4ab: Pull complete
1a5fd5c7e184: Pull complete
7b87d6fa783d: Pull complete
bd819c9b5ead: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:72e9feec71242af83faf65f95a40d5e3bb2822a6c3b2cda8568790f3d31aecde
Status: Downloaded newer image for sonarqube:latest
docker.io/library/sonarqube:latest

What's next:
  View a summary of image vulnerabilities and recommendations → docker scout quickview sonarqube
```

STEP 3:Run docker pull sonarqube command to install sonarqube image without actually installing then.

```
C:\Users\Dell>docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest
ac1f985dedebc00a642a4c69a502d611389e8f9fa46610febe75aa5021767cab
```

STEP 4:Run docker run -d --name sonarqube -e SONAR_ES_BOOTSTRAP_CHECKS_DISABLE=true -p 9000:9000 sonarqube:latest Command to run the sonarqube.




STEP 5: Once the sonarqube is runned go to your web browser and whatever port number u have mentioned in the previous command open that page using localhost:9000.



STEP 6: Once sonarqube is started it will redirect you to login page .The login and password for sonarqube is both “admin”.

Update your password

 This account should not use the default password.

Enter a new password

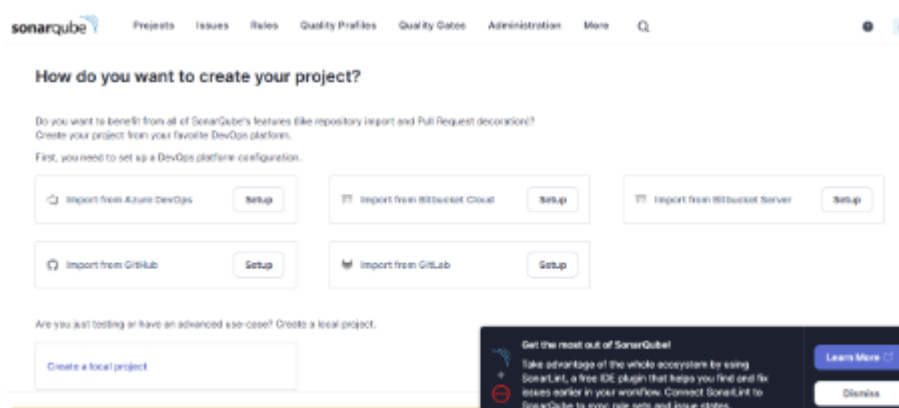
All fields marked with * are required

Old Password *

New Password *

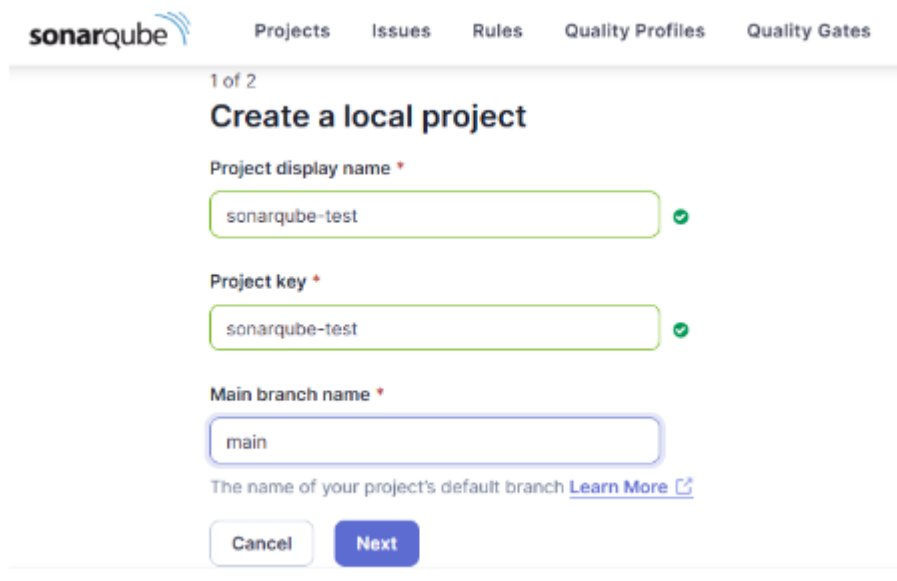
Confirm Password *

STEP 8: After changing the password, you will be directed to this screen. Click on Create a Local Project.



The screenshot shows the SonarQube web interface. At the top is a navigation bar with links: Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, and More. Below the navigation bar is a section titled "How do you want to create your project?". It contains a paragraph explaining that to benefit from SonarQube's features, users should create a project from a DevOps platform. Below this are five buttons: "Import from Azure DevOps", "Import from Bitbucket Cloud", "Import from Bitbucket Server", "Import from GitHub", and "Import from GitLab". Each button has a "Setup" link next to it. At the bottom left, there is a button "Create a local project". At the bottom right, there is a dark blue banner with the text "Get the most out of SonarQube!" and a "Learn More" button.

STEP 9: Add name of the project and project key and select the main branch name and click on next.



The screenshot shows the 'Create a local project' form in SonarQube. The form is titled '1 of 2' and 'Create a local project'. It has three input fields: 'Project display name' with the value 'sonarqube-test', 'Project key' with the value 'sonarqube-test', and 'Main branch name' with the value 'main'. Each field has a green checkmark to its right. Below the 'Main branch name' field, there is a link 'Learn More' with an external link icon. At the bottom of the form, there are two buttons: 'Cancel' and 'Next'.

sonarqube Projects Issues Rules Quality Profiles Quality Gates

1 of 2

Create a local project

Project display name *

sonarqube-test ✓

Project key *

sonarqube-test ✓

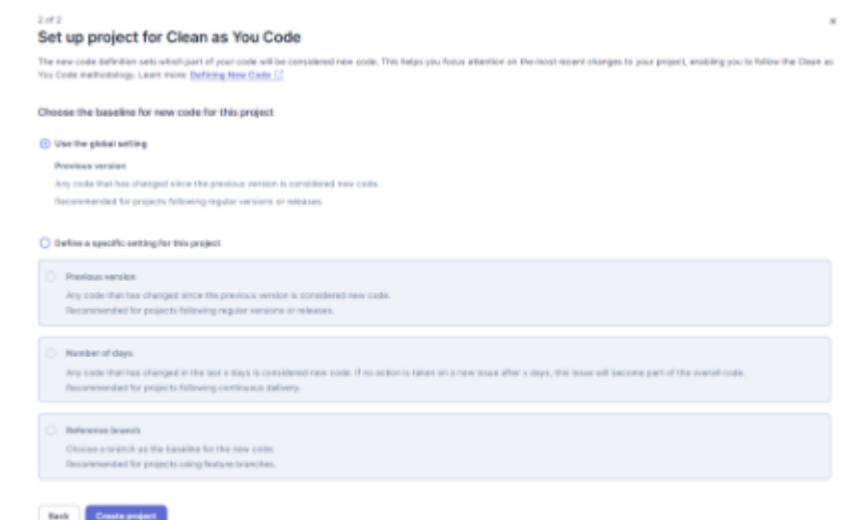
Main branch name *

main

The name of your project's default branch [Learn More](#)

Cancel Next

STEP 10: Set up the project as required and click on create.



The screenshot shows the 'Set up project for Clean as You Code' form in SonarQube. The form is titled '2 of 2' and 'Set up project for Clean as You Code'. It has a section 'Choose the baseline for new code for this project' with three radio button options: 'Use the global setting' (selected), 'Previous version', and 'Define a specific setting for this project'. The 'Define a specific setting for this project' option is expanded, showing three sub-options: 'Previous version', 'Number of days', and 'Reference branch'. At the bottom of the form, there are two buttons: 'Back' and 'Create project'.

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

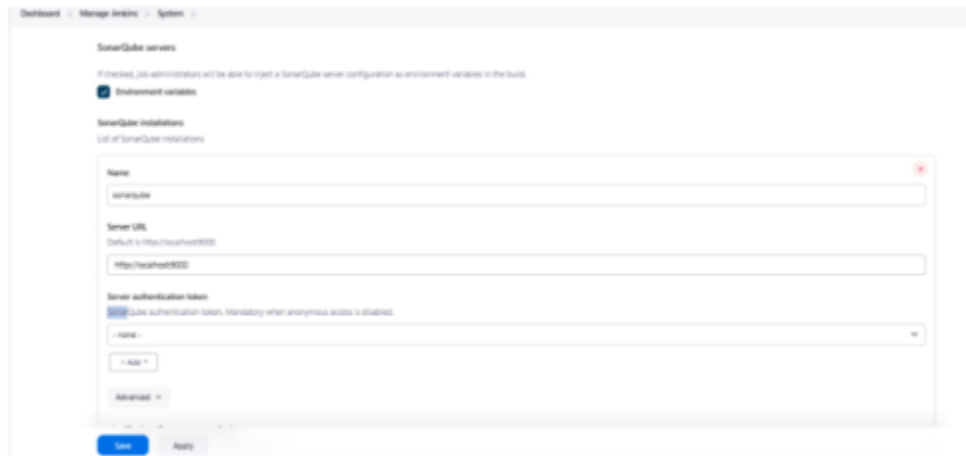
Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will become part of the overall code.
Recommended for projects following continuous delivery.

☐ Reference branch

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

Back Create project

STEP 11: Go to Manage Jenkins and then go to Systems and name to the environment variables then apply the changes and then save them.



The screenshot shows the 'Manage Jenkins' > 'System' configuration page. Under 'SonarQube servers', the 'Environment variables' checkbox is checked. Below, the 'SonarQube installations' section is visible, showing a list of installations. The first installation is named 'sonarqube'. The 'Server URL' is set to 'https://localhost:9000'. The 'Server authentication token' is set to 'token'. The 'Add' button is visible at the bottom of the installation list.

STEP 12: In SonarQube Scanner add the latest version then apply the changes and save it.



The screenshot shows the 'SonarQube Scanner' configuration page. The 'Name' field is set to 'sonarqube_lab'. The 'Install automatically' checkbox is checked. Under the 'Install from Maven Central' section, the 'Version' dropdown is set to 'SonarQube Scanner 6.2.0.4504'. The 'Add Installer' button is visible. At the bottom, the 'Add SonarQube Scanner' button is visible. The 'Ant installations' section is also visible at the bottom.

STEP 15:Add the script to the pipeline script.

Definition

Pipeline script

Script ?

```
1 * node {
2 *   stage('Cloning the GitHub Repo') {
3 *     git 'https://github.com/shacforiot/60L.git'
4 *   }
5 *
6 *   stage('SonarQube analysis') {
7 *     withSonarQubeEnv('sonarqube1eb') {
8 *       bat --
9 *       C:\Users\saira\Downloads\sonar-scanner-cli-0.1.0.4477-windows-x64\sonar-scanner-0.1.0.4477-windows-x64\bin\sonar-scanner.bat ^
10 *      -D sonar.login=admin ^
11 *      -D sonar.password=123456 ^
12 *      -D sonar.projectKey=sonarqube27 ^
13 *      -D sonar.exclusions=vendor/**,resources/**,**/*.java ^
14 *      -D sonar.host.url=http://localhost:9000/
15 *     }
16 *   }
17 * }
18 }
```

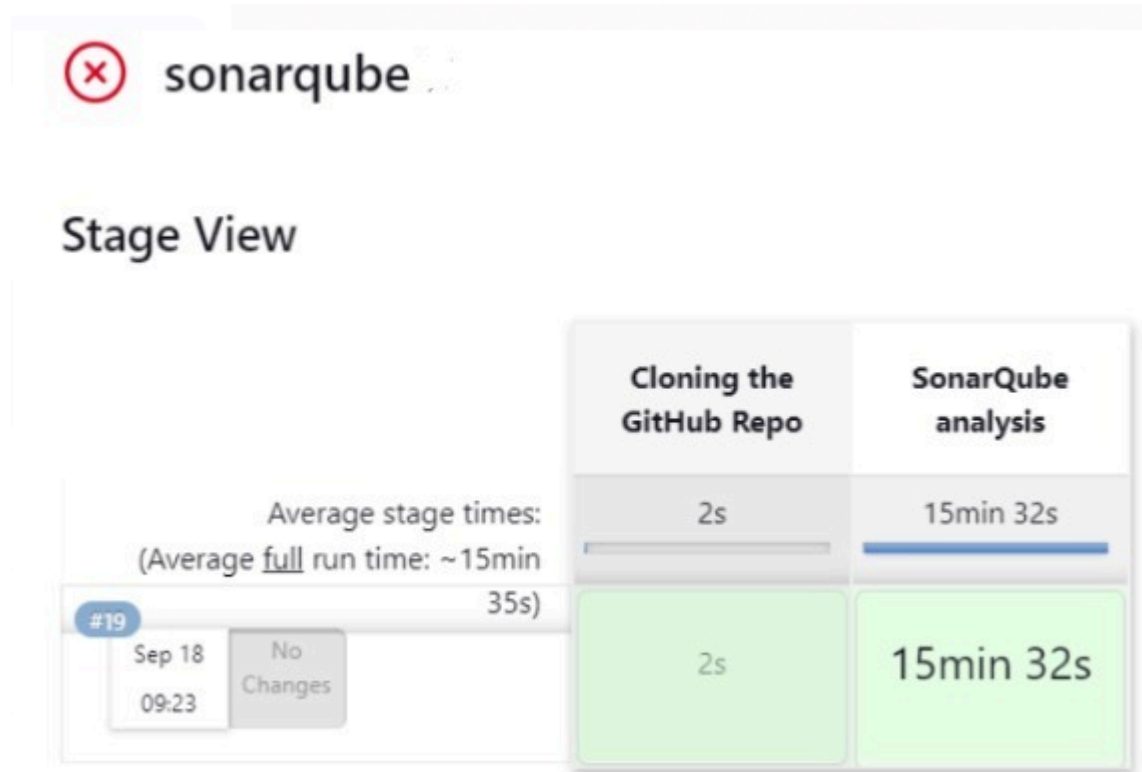
try sample Pipeline...

☒ Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

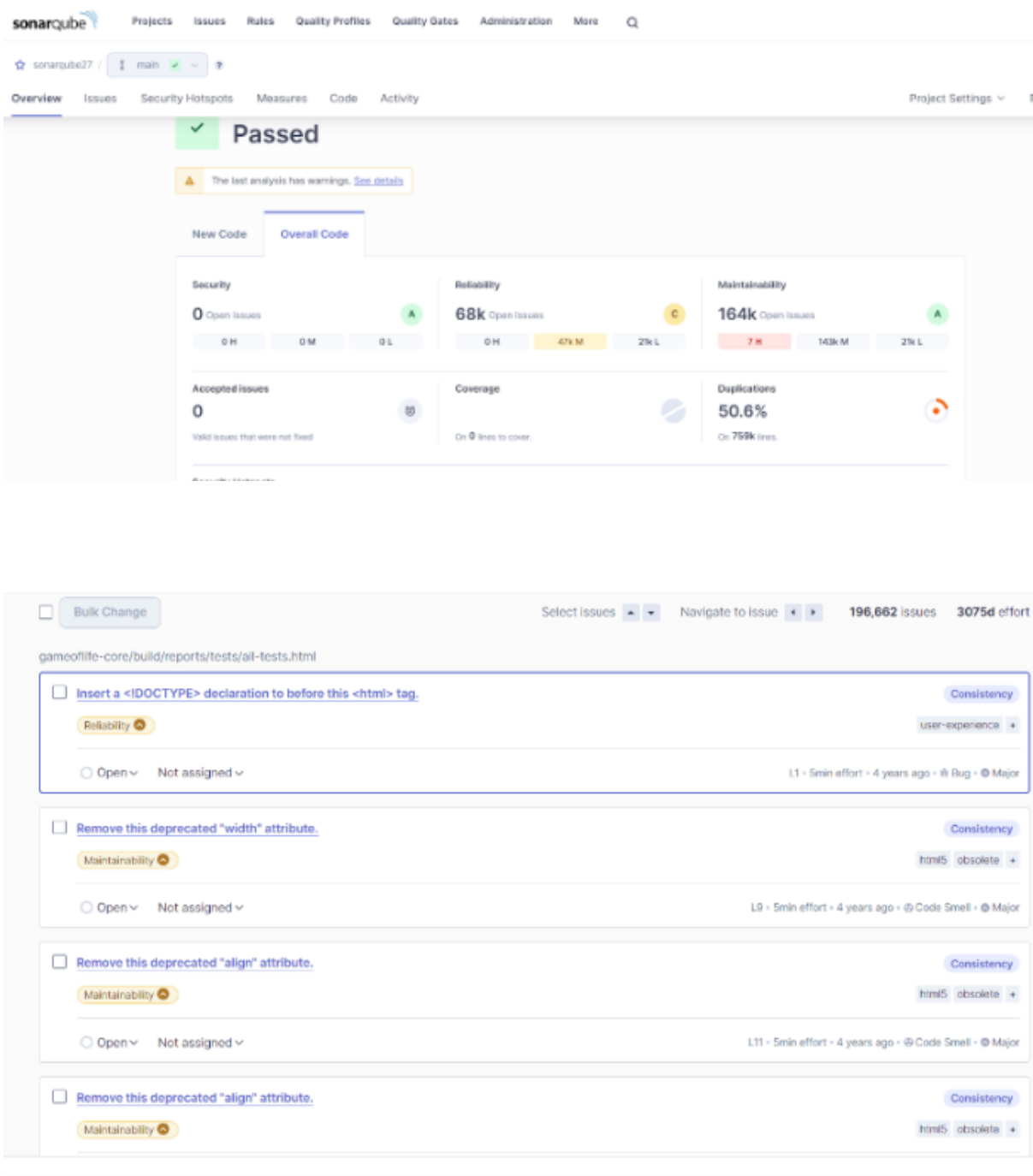
STEP 16:Go to the job you had just built and click on Build Now



STEP 17: Check the console output

```
09:38:08.069 INFO SCM revision ID 'ba799ba7e1b576f04a6612322b0412c5e6e1e5ed'
09:38:20.750 INFO Analysis report generated in 4390ms, dir size=127.2 MB
09:38:45.857 INFO Analysis report compressed in 25009ms, zip size=29.6 MB
09:38:46.532 INFO Analysis report uploaded in 675ms
09:38:46.533 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=sonarqube27
09:38:46.533 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
09:38:46.533 INFO More about the report processing at http://localhost:9000/api/ci/task?id=46576333-cbde-4277-89d7-471ee554de32
09:39:00.038 INFO Analysis total time: 15:24.256 s
09:39:00.041 INFO SonarScanner Engine completed successfully
09:39:00.810 INFO EXECUTION SUCCESS
09:39:00.811 INFO Total time: 15:29.301s
[Pipeline] }
[Pipeline] // withSonarQubeEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

STEP 18: Go to SonarQube and check the project linked.



STEP 19:Check all the issues with the code.

gameoflife-acceptance-tests/Dockerfile

☐

Use a specific version tag for the image.

Intentionality

Maintainability

No tags

Open

Not assigned

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

Open

Not assigned

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

Open

Not assigned

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

Open

Not assigned

☐

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

Intentionality

Reliability

accessibility wcag2-a

Open

Not assigned

L1 • 2min effort • 4 years ago • Bug • Major

☐

Add "<th>" headers to this "<table>".

Intentionality

Reliability

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

Intentionality

Reliability

accessibility wcag2-a

Open

Not assigned

L1 • 2min effort • 4 years ago • Bug • Major

☐

Add "<th>" headers to this "<table>".

Intentionality

Open

Not assigned

☐ Bulk Change

Select issues Navigate to issue 268 issues 2d 5h effort

gameoflife-acceptance-tests/Dockerfile

☐ Use a specific version tag for the image.

Intentionality

Maintainability

No tags

Open Not assigned

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

Open Not assigned

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

Open Not assigned

L12 • 5min effort • 4 years ago • Code Smell • Major

☐ Surround this variable with double quotes; otherwise, it can lead to unexpected behavior.

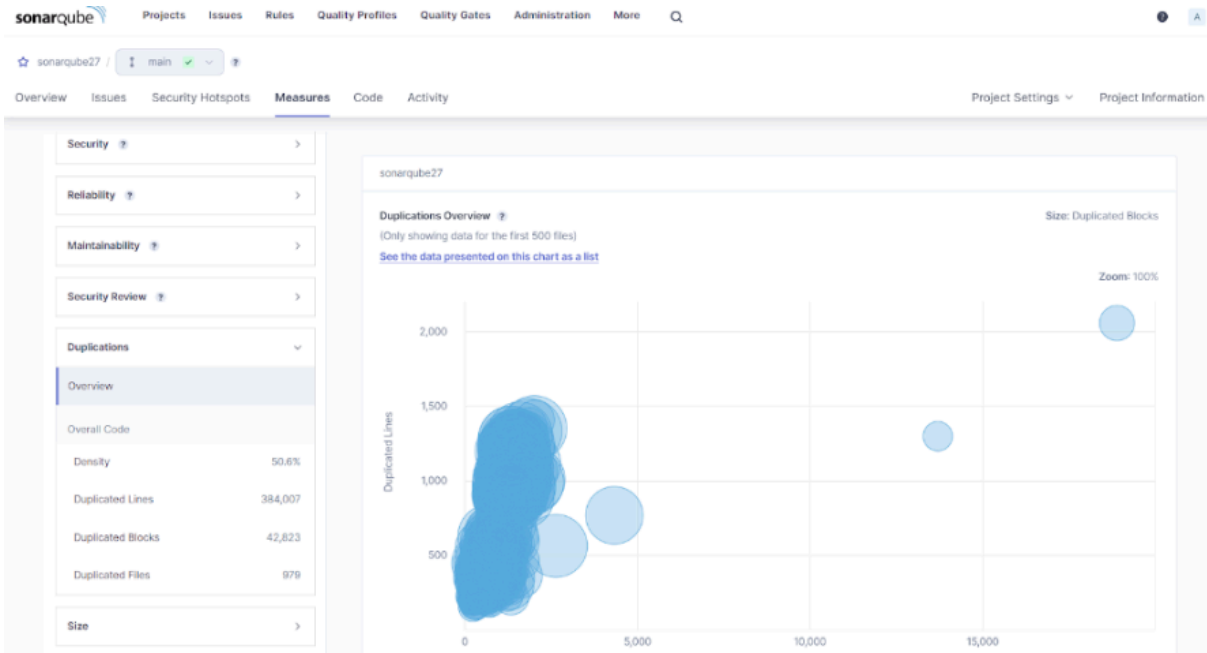
Intentionality

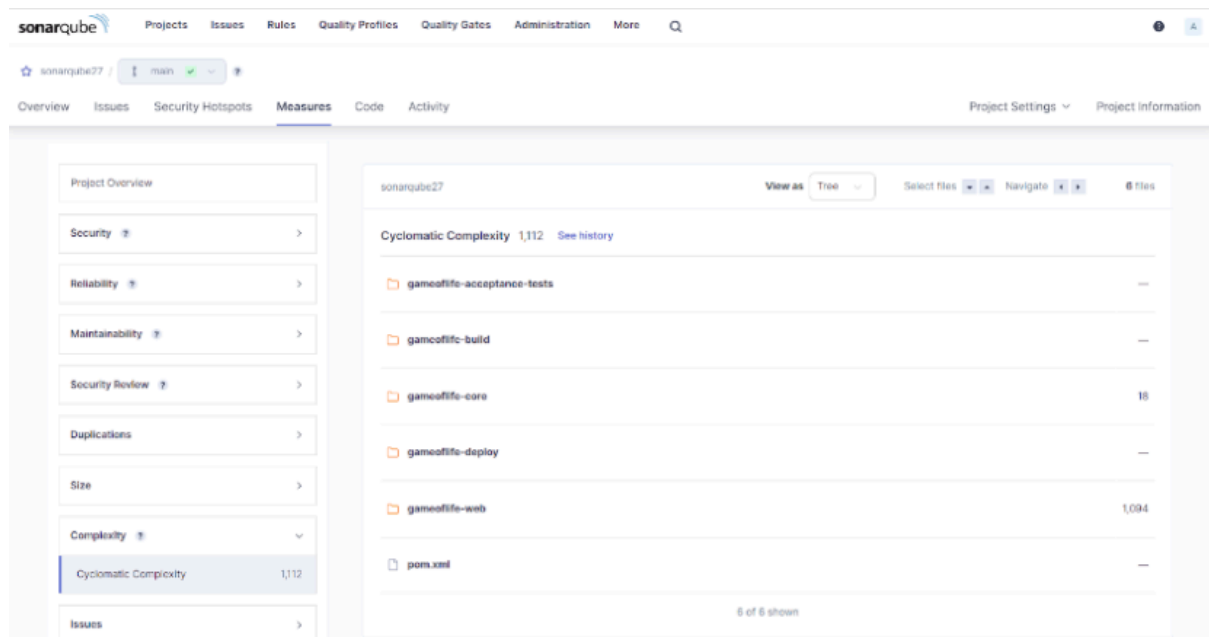
Maintainability

No tags

Open Not assigned

L12 • 5min effort • 4 years ago • Code Smell • Major





Conclusion: We established a Jenkins CI/CD pipeline with SonarQube to perform static analysis on a sample web/Java/Python application. This pipeline automated the detection of bugs, code smells, and security vulnerabilities. During the process, we faced issues with SonarQube server connectivity and Jenkins job configurations. These were resolved by adjusting network settings and refining the pipeline script. The integration successfully enhanced our code quality and security by providing continuous and automated feedback throughout the development lifecycle.