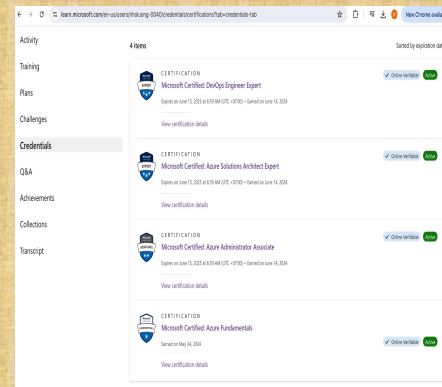
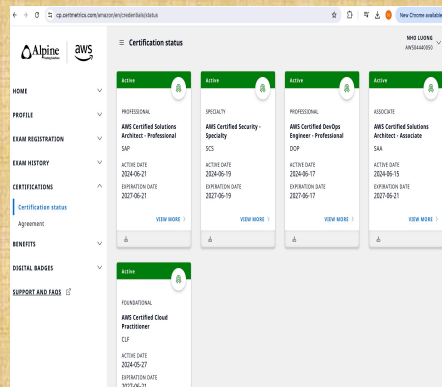


SonarQube Overview

Author: Nho Luong
Skill: DevOps Engineer Lead



About SonarQube

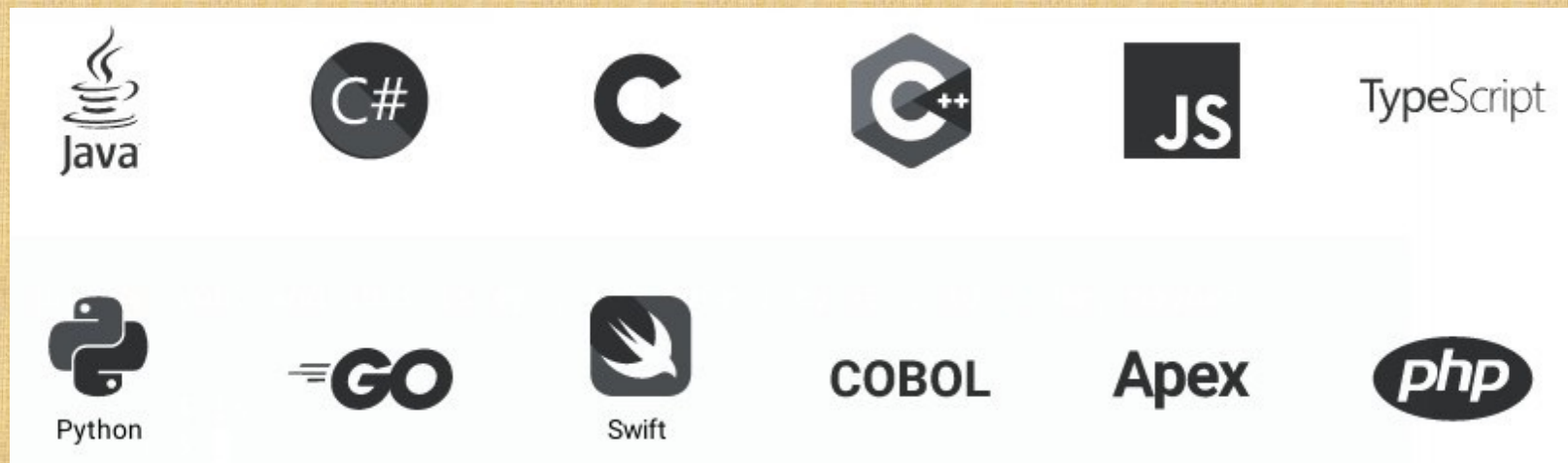


"SonarQube empowers all developers to write cleaner and safer code."

~ <https://www.sonarqube.org/>

- **Code inspection tool**
- **Can be used in CI/CD environment**
- **Available with probably many build management tools**

Multi-language Support



Detect Possible Code Issues



Find bugs and performance issues that may not be easy to find

Detect Possible Code Issues



src/.../com/example/demo/DemoApplicatio...

Add at least one assertion to this test case.

Code Smell

1 of 1 shown

```
7 import org.springframework.boot.test.web.client.TestRestTemplate;
8 import org.springframework.beans.factory.annotation.Autowired;
9 import org.springframework.test.context.junit4.SpringRunner;
10 import static org.assertj.core.api.Assertions.assertThat;
11
12 @RunWith(SpringRunner.class)
13 @SpringBootTest(webEnvironment = WebEnvironment.RANDOM_PORT)
14 public class DemoApplicationTests {
15
16     @Test
17     public void contextLoads() {
18
19
20     @Autowired
21     private TestRestTemplate restTemplate;
22 }
```

Add at least one assertion to this test case. [See Rule](#) 5 days ago ▾ L17 🔑

Code Smell ▾ ! Blocker ▾ ○ Open ▾ Not assigned ▾ 10min effort Comment tests ▾

SonarQube detects possible Code Smells

Detect Possible Code Issues

A screenshot of the SonarQube web interface. On the left, a sidebar shows a file path "src/.../com/example/demo/DemoApplicatio..." and a list of issues. The first issue is highlighted with a blue box and contains the text "Make this method 'public'." and "Vulnerability". Below this, it says "1 of 1 shown". The main area displays a Java code snippet for a Spring Boot application. The code includes package declarations, imports for Spring Boot and Spring Web, and a class definition for "DemoApplication". The class has annotations for "@SpringBootApplication" and "@RestController", and a method "home()" with a "@GetMapping('/')". A red squiggly line under the "home()" method indicates a code issue. A tooltip is visible over the code, showing the same issue description: "Make this method 'public'." with a "See Rule" link. Below the description, there are tags for "Vulnerability", "Blocker", "Open", "Not assigned", and "2min effort". On the right side of the tooltip, it shows "5 days ago", "L12", and a link icon. At the bottom right of the tooltip, it shows "owasp-a6, spring".

src/.../com/example/demo/DemoApplicatio...

Make this method "public".
Vulnerability

1 of 1 shown

```
1 package com.example.demo;
2
3 import org.springframework.boot.*;
4 import org.springframework.boot.autoconfigure.*;
5 import org.springframework.web.bind.annotation.*;
6
7 @SpringBootApplication
8 @RestController
9 public class DemoApplication {
10
11     @GetMapping("/")
12     String home() {
13
14         return "Spring is here!";
15     }
16 }
```

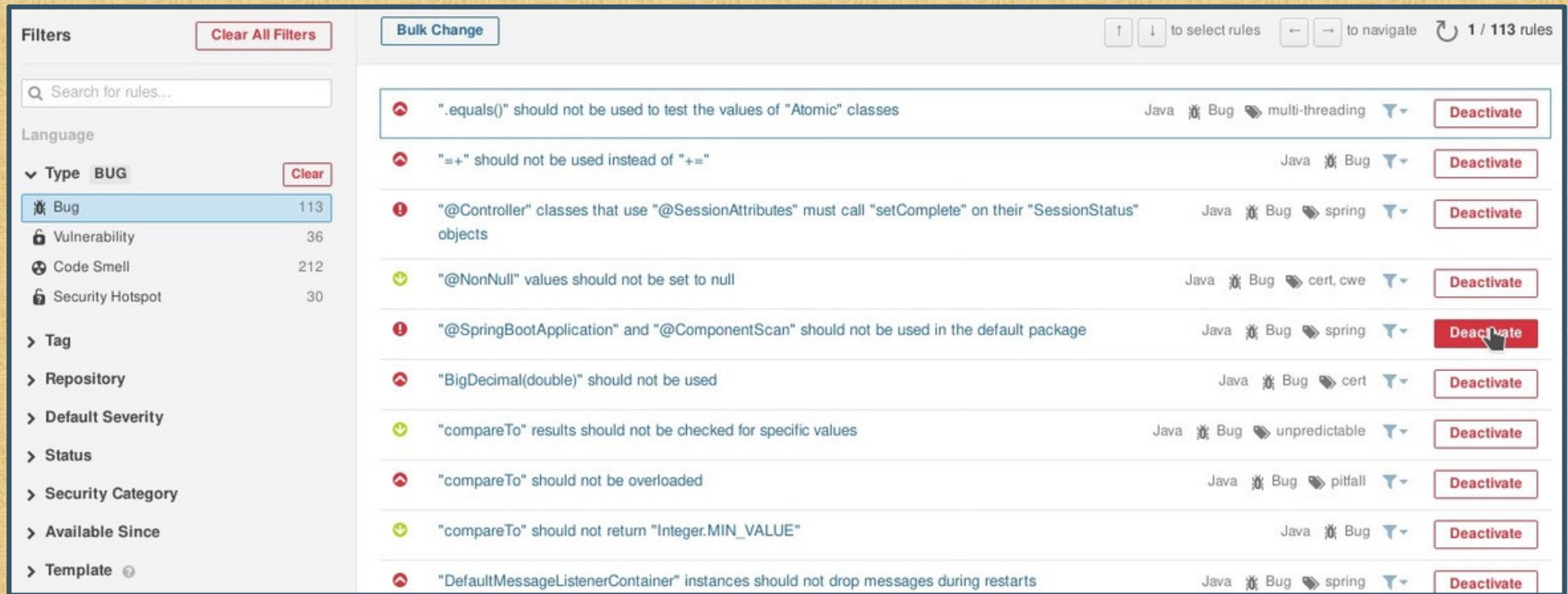
Make this method "public". [See Rule](#) 5 days ago L12

Vulnerability Blocker Open Not assigned 2min effort Comment

owasp-a6, spring

Don't worry about security invulnerability

Detect Possible Code Issues



Filters

Clear All Filters

Bulk Change

1 / 113 rules

Search for rules...

Language

Type: **BUG** (113)

Vulnerability (36)

Code Smell (212)

Security Hotspot (30)

Tag

Repository

Default Severity

Status

Security Category

Available Since

Template

Rule	Language	Severity	Tags	Action
".equals()" should not be used to test the values of "Atomic" classes	Java	Bug	multi-threading	Deactivate
"=+" should not be used instead of "+="	Java	Bug		Deactivate
"@Controller" classes that use "@SessionAttributes" must call "setComplete" on their "SessionStatus" objects	Java	Bug	spring	Deactivate
"@NonNull" values should not be set to null	Java	Bug	cert, cwe	Deactivate
"@SpringBootApplication" and "@ComponentScan" should not be used in the default package	Java	Bug	spring	Deactivate
"BigDecimal(double)" should not be used	Java	Bug	cert	Deactivate
"compareTo" results should not be checked for specific values	Java	Bug	unpredictable	Deactivate
"compareTo" should not be overloaded	Java	Bug	pitfall	Deactivate
"compareTo" should not return "Integer.MIN_VALUE"	Java	Bug		Deactivate
"DefaultMessageListenerContainer" instances should not drop messages during restarts	Java	Bug	spring	Deactivate

Customizable rules allow you to fit guidelines for your projects

Getting Started

Using SonarCloud



Free for public projects

- **Unlimited** lines of code
- Project **visible** to everyone
- Access to **full feature set**
- **Restrict** access to work on your project

Paid for private projects

- Create **private projects**, priced per lines of code
- Restrict **complete** access to your project
- Pricing: 100k lines → 10€/month

- Login at <https://sonarcloud.io> You can either create an
- account or use single-sign on with GitHub, Bitbucket, Azure DevOps

Getting Started

Using SonarCloud

A screenshot of the SonarCloud website's landing page. The page has a white background with a large orange wave graphic on the right side. In the top left corner is the "sonarcloud" logo with a cloud icon. In the top right corner are links for "Pricing" and "Log in". The main heading is "Clean Code Rockstar Status" in orange and black. Below it is the text "Eliminate bugs and vulnerabilities. Champion quality code in your projects." and "Go ahead! Analyze your repo:". There are three buttons for "GitHub", "Bitbucket", and "Azure DevOps". A note says "Free for Open-Source Projects". On the right, three cartoon characters are standing: a woman with blonde hair pointing up, a man with a beard and sunglasses wearing a red cap, and a man with dark skin wearing a purple shirt that says "BUGS NO MORE". The man in the middle wears a red shirt that says "KEEP CALM AND WRITE CLEAN CODE".

sonarcloud

Pricing Log in

Clean Code Rockstar Status

Eliminate bugs and vulnerabilities.
Champion quality code in your projects.

Go ahead! Analyze your repo:

GitHub Bitbucket Azure DevOps

Free for Open-Source Projects

<https://sonarcloud.io>

Getting Started



SonarQube + PostgreSQL // Docker

Prerequisites (Linux):

These additional configurations are needed due to the usage of an embedded ElasticSearch

```
$ sysctl -w vm.max_map_count=262144  
$ sysctl -w fs.file-max=65536  
$ ulimit -n 65536  
$ ulimit -u 4096
```

Note: Properties set via `sysctl` are not persistent after restart of the host system.

Make sure to edit these values in `/etc/sysctl.d/...` as well.

Getting Started



SonarQube + PostgreSQL // Docker

docker-compose + bitnami image:

```
$ curl -LO https://raw.githubusercontent.com/bitnami/bitnami-docker-sonarqube/master/docker-compose.yml
```

```
$ docker-compose up
```

```
sonarqube_1 | Welcome to the Bitnami sonarqube container
sonarqube_1 | Subscribe to project updates by watching https://github.com/bitnami/bitnami-docker-sonarqube
sonarqube_1 | Submit issues and feature requests at https://github.com/bitnami/bitnami-docker-sonarqube/issues
sonarqube_1 |
postgresql_1 | postgresql 11:25:05.96
postgresql_1 | postgresql 11:25:05.98 Welcome to the Bitnami postgresql container
postgresql_1 | postgresql 11:25:05.99 Subscribe to project updates by watching https://github.com/bitnami/bitnam
postgresql_1 | postgresql 11:25:06.00 Submit issues and feature requests at https://github.com/bitnami/bitnami-d
postgresql_1 | postgresql 11:25:06.01 Send us your feedback at containers@bitnami.com
postgresql_1 | postgresql 11:25:06.02
postgresql_1 | postgresql 11:25:06.09 INFO ==> ** Starting PostgreSQL setup **
postgresql_1 | postgresql 11:25:06.37 INFO ==> Validating settings in POSTGRES*_ env vars..
postgresql_1 | postgresql 11:25:06.38 WARN ==> You set the environment variable ALLOW_EMPTY_PASSWORD=yes. For s
ag in a production environment.
postgresql_1 | postgresql 11:25:06.40 INFO ==> Loading custom pre-init scripts...
postgresql_1 | postgresql 11:25:06.43 INFO ==> Initializing PostgreSQL database...
postgresql_1 | postgresql 11:25:06.47 INFO ==> postgresql.conf file not detected. Generating it...
postgresql_1 | postgresql 11:25:06.53 INFO ==> pg_hba.conf file not detected. Generating it...
sonarqube_1 | nami INFO Initializing postgresql-client
```

Getting Started



Maven / Gradle goals

Maven (pom.xml)

```
<build>
  <plugins>
    <plugin>
      <groupId>org.sonarsource.scanner.maven</groupId>
      <artifactId>sonar-maven-plugin</artifactId>
      <version>3.6.0.1398</version>
    </plugin>
  </plugins>
</build>
```

Gradle (build.gradle)

```
plugins {
  id 'org.sonarqube' version '2.8'
}
```


Getting Started



Maven / Gradle goals

Maven

```
# Alternatively, write -D variables into your .m2/settings.xml
$ mvn sonar:sonar -Dsonar.host.url=<YOUR_URL> \
    -Dsonar.login=<YOUR_SONAR_TOKEN>
```

Gradle

```
# In ./gradle.properties
systemProp.sonar.host.url=<YOUR_URL>
systemProp.sonar.login=<YOUR_SONAR_TOKEN>
# In console (or IDE):
$ gradle sonarqube
```


Getting Started

Sonar Scanner



Docker

Advantage of using Docker: Can be used in CI/CD

```
$ docker run -e  
SONAR_HOST_URL=<YOUR_SONAR_URL> \  
-e  
SONAR_TOKEN=<YOUR_SONAR_TO \  
KEN>  
--user="$(id -u):$(id -g)" \ -it \ -v  
"$PWD:/usr/src" \ sonarsource/sonar-  
scanner-cli
```

sonar-project.properties

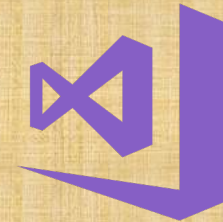
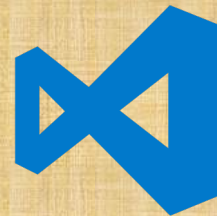
```
sonar.projectName=<YOUR_PROJECT_NAME> # e. g. "Hello World"  
sonar.projectKey=<YOUR_PROJECT_KEY> # e. g. com.example:hello-world  
sonar.projectVersion=<YOUR_PROJECT_VERSION> # e. g. 1.0  
sonar.sources=src sonar.java.source=1.8 sonar.java.target=1.8  
sonar.java.binaries=build/classes
```

sonarlint

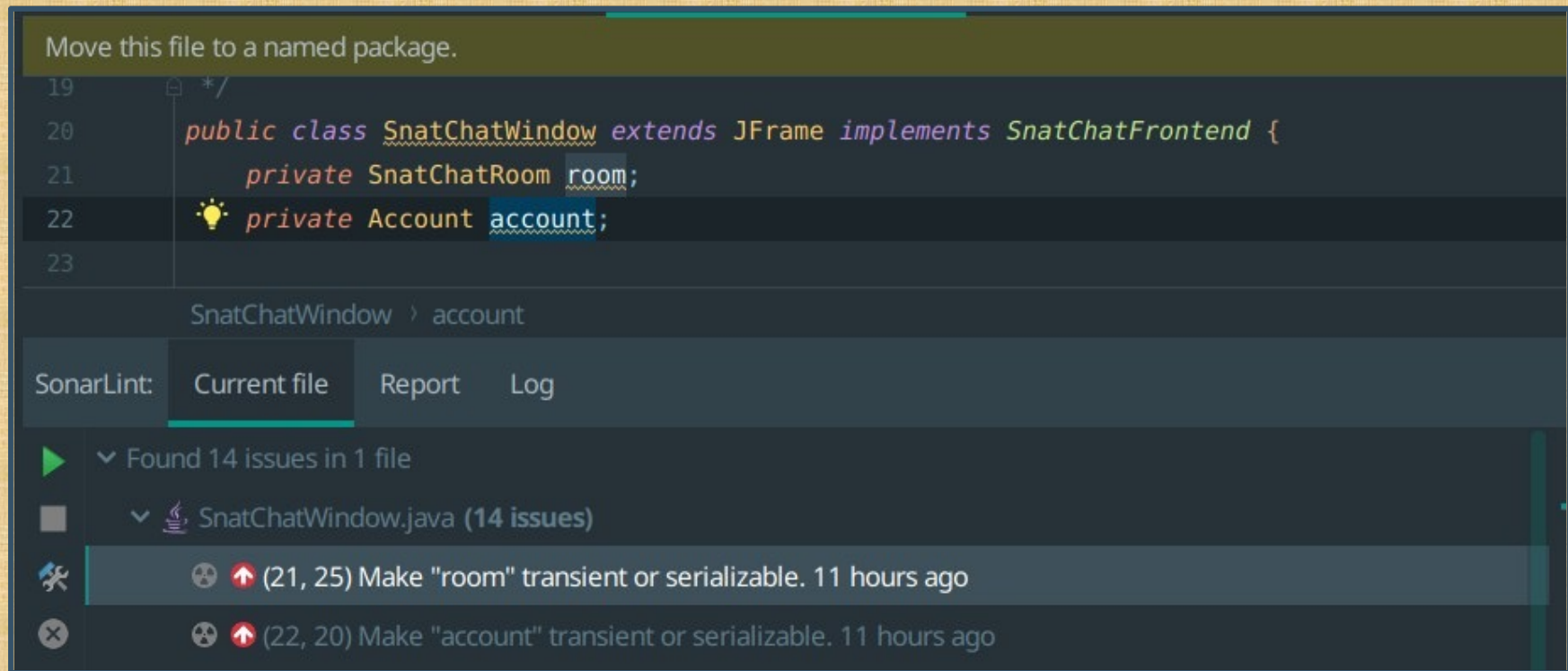


Fix issues before they exist
~ <https://www.sonarlint.org>

Available on:



IDE Integration



How it looks like in IntelliJ + Java

Custom Rules



Custom resources should be closed	Rule Template	Java	Bug	denial-of-service	▼
Track comments matching a regular expression	Rule Template	Java	Code Smell		▼
Track uses of disallowed classes	Rule Template	Java	Code Smell		▼
Track uses of disallowed constructors	Rule Template	Java	Code Smell		▼
Track uses of disallowed dependencies	Rule Template	Java	Code Smell	maven	▼
Track uses of disallowed methods	Rule Template	Java	Code Smell		▼

Predefined rule templates can be found in **Rules**
> *Filters* > *Templates* > *Show Templates Only*

Custom Rules



Update Custom Rule

Name *

Usage of System.exit(0)

Key

squid:usage_system_exit_0

Severity

! Blocker

Status

Ready

Description *

Don't use it. Instead, I don't know. Make something up from your own.

[Markdown Help](#) : *Bold* ``Code`` * Bulleted point

MethodName

exit

ClassName

java.lang.System

ArgumentTypes

Save

Cancel

Custom Rules



A screenshot of the SonarQube web interface. The top part shows a code editor with a Java file named "src/main/java/SnatChat.java". The code has two lines highlighted in yellow: line 11, "System.out.println(Message.rot13(test));", and line 13, "System.exit(0);". A red vertical line is positioned between the code editor and the right-hand panel. The right-hand panel displays a "Remove this forbidden call" message for the "System.exit(0)" call. Below this message, there are several tabs: "Vulnerability", "See Rule", and "Manual". The "Vulnerability" tab is active, showing a "Vulnerability" icon, a "Blocker" severity level, and a "Manual" button. To the right of the tabs, it says "6 hours ago" and "Not assigned". Below the tabs, there is a "Comment" button. The bottom part of the screenshot shows a detailed view of the "Usage of System.exit(0)" rule. It includes a title "Usage of System.exit(0)" and a link "squid:usage_system_exit_0". Below the title, there are several tags: "Security Hotspot", "Blocker", "Main sources", "No tags", "Available Since Nov 13, 2019", "SonarAnalyzer (Java)", and "Custom Rule (Show Template)". The main content area of this panel contains the text: "Don't use it. Instead, I don't know. Make something up from your own."

Further advanced custom rule implementations are available. You can checkout <https://github.com/SonarSource/sonar-custom-rules-examples/tree/master/java-custom-rules> if you need an example

You can find an example project with Java and Gradle in my Git repository in folder “gradle-example”



Thank You