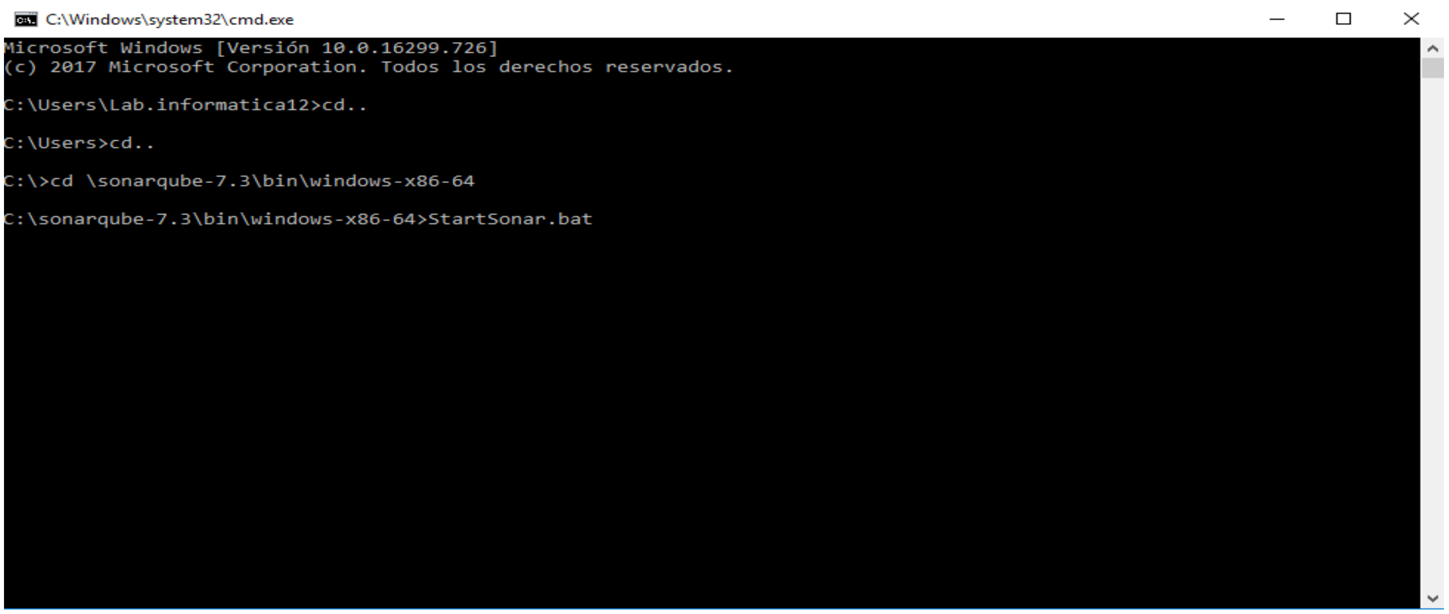
Prueba de Software con SonarQube

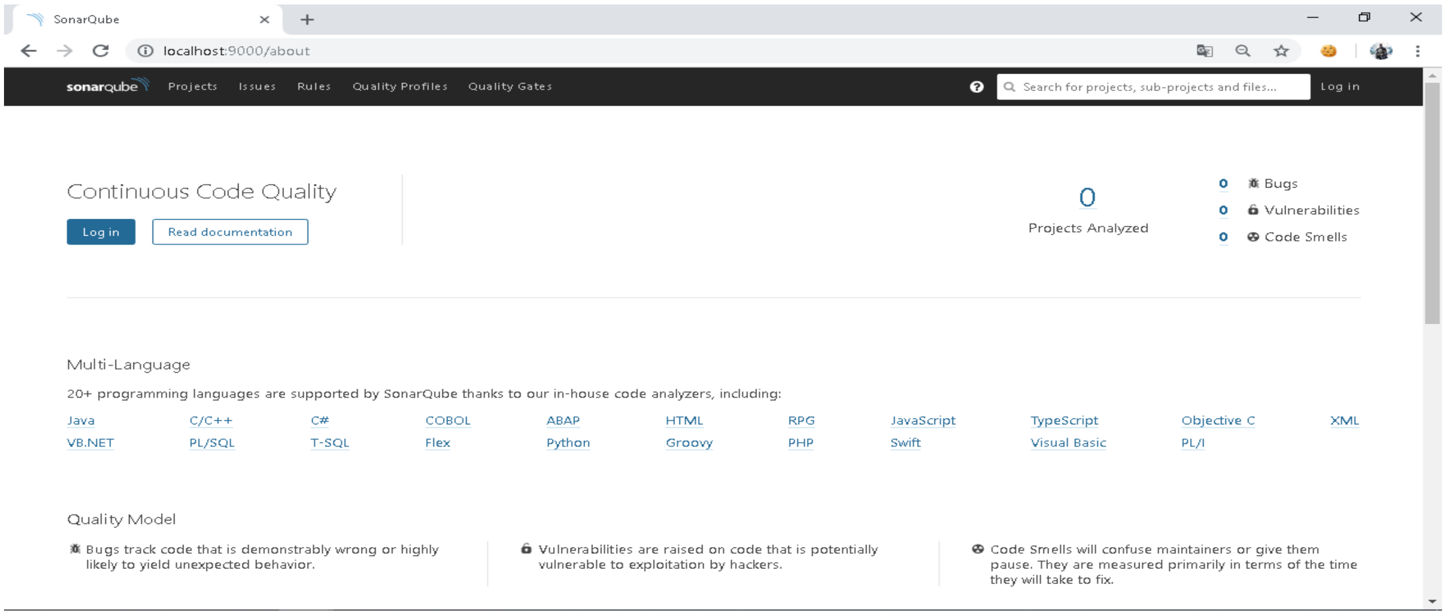
En este proceso se realizara el paso a paso de como validar código con SonarQube, teniendo en cuanta los diferentes lenguajes de programación.

1. Iniciar el servicio de SonarQube desde la consola CMD, para este caso la ubicación es la siguiente: C:\sonarqube-7.3\bin\windows-x86-64\StarSonar.bat



```
Microsoft Windows [Versión 10.0.16299.726]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Lab.informatica12>cd..

C:\Users>cd..

C:\>cd \sonarqube-7.3\bin\windows-x86-64

C:\sonarqube-7.3\bin\windows-x86-64>StartSonar.bat
```

2. Una vez iniciado el servicio de SonarQube, Ingresamos al navegador web y digitamos la siguiente dirección URL http://localhost:9000 , allí deberíamos poder acceder al SonarQube y administrar nuestros proyectos.



Continuous Code Quality

0 Projects Analyzed

0 Bugs
0 Vulnerabilities
0 Code Smells

Multi-Language

20+ programming languages are supported by SonarQube thanks to our in-house code analyzers, including:

Java, C/C++, C#, COBOL, ABAP, HTML, RPG, JavaScript, TypeScript, Objective C, XML, VB.NET, PL/SQL, T-SQL, Flex, Python, Groovy, PHP, Swift, Visual Basic, PL/I

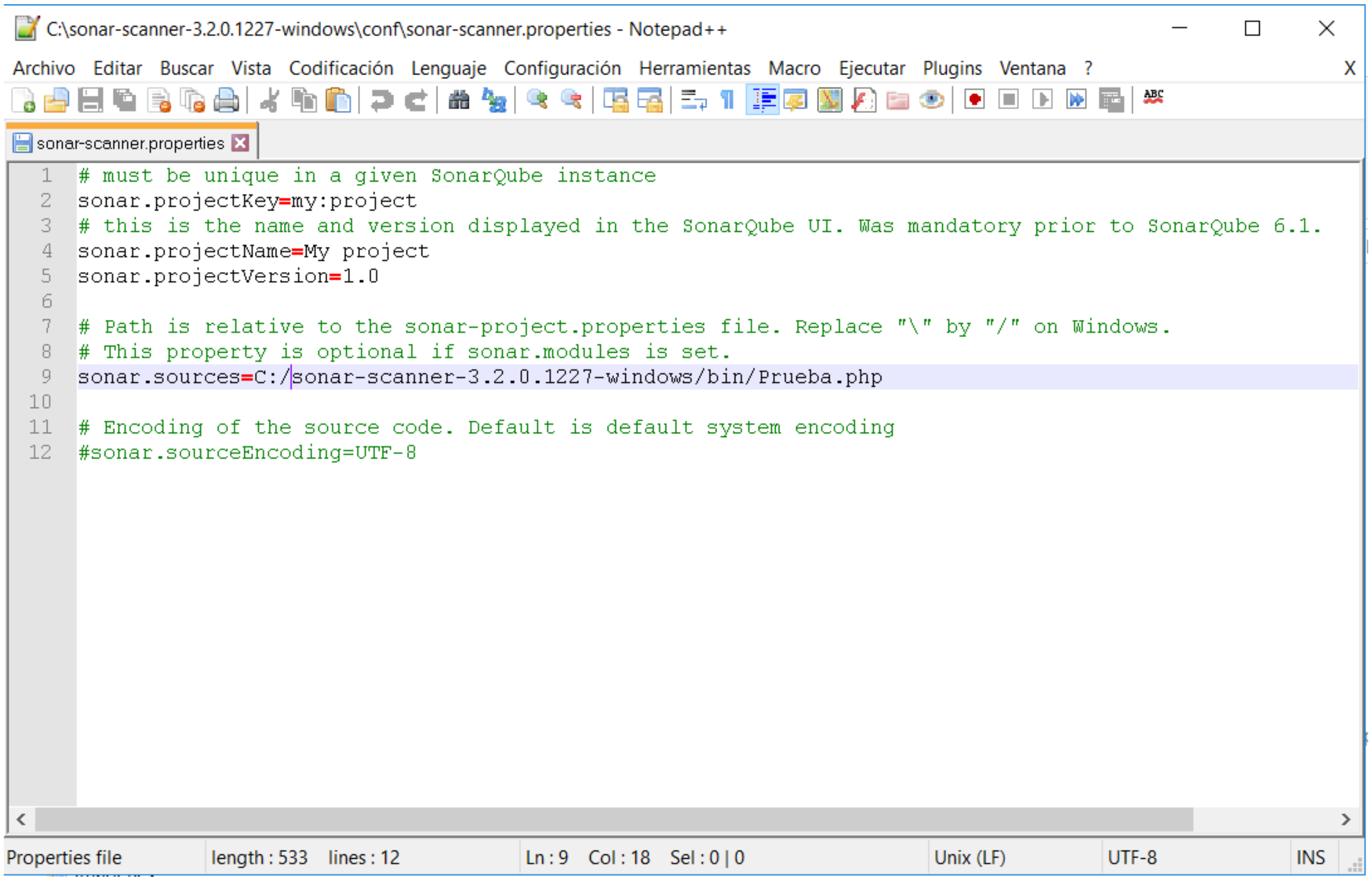
Quality Model

Bugs track code that is demonstrably wrong or highly likely to yield unexpected behavior.

Vulnerabilities are raised on code that is potentially vulnerable to exploitation by hackers.

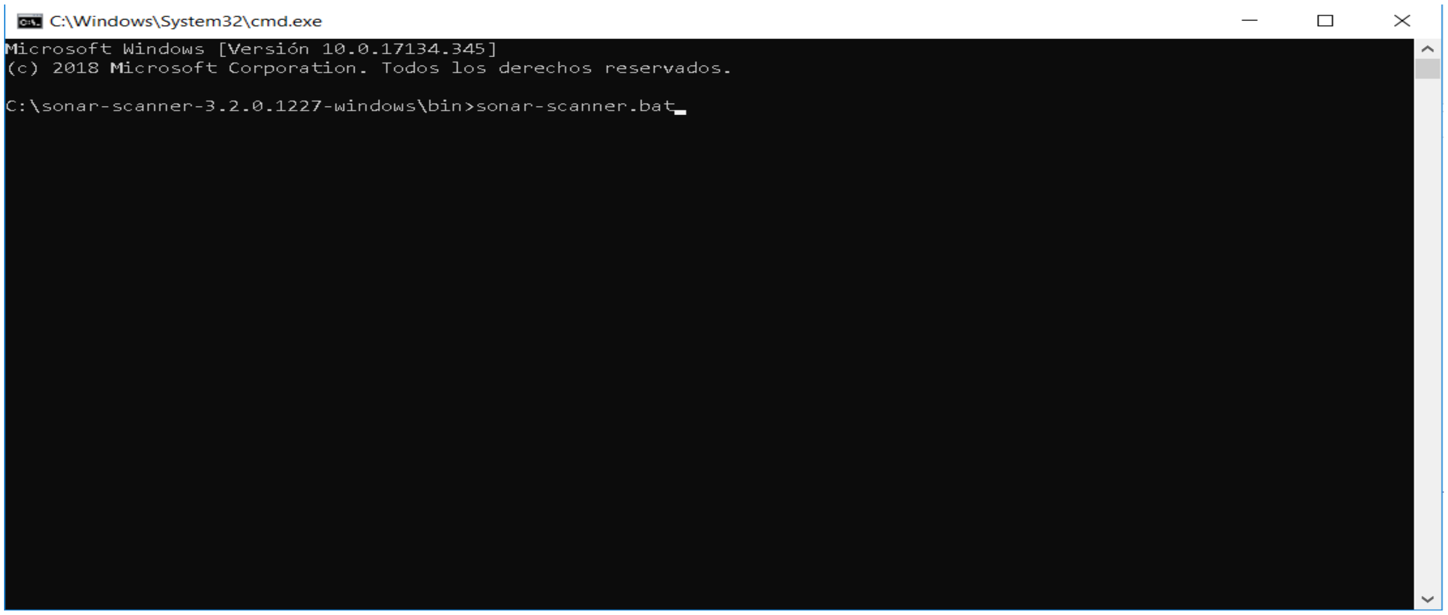
Code Smells will confuse maintainers or give them pause. They are measured primarily in terms of the time they will take to fix.

3. Procedemos a validar un código en el aplicativo, para este caso del lenguaje de programación PHP, inicialmente realizamos la configuración el archivo C:\sonar-scanner-3.2.0.1227-windows\conf\sonar-scanner.properties, posteriormente, compilar el código que vamos a validar.

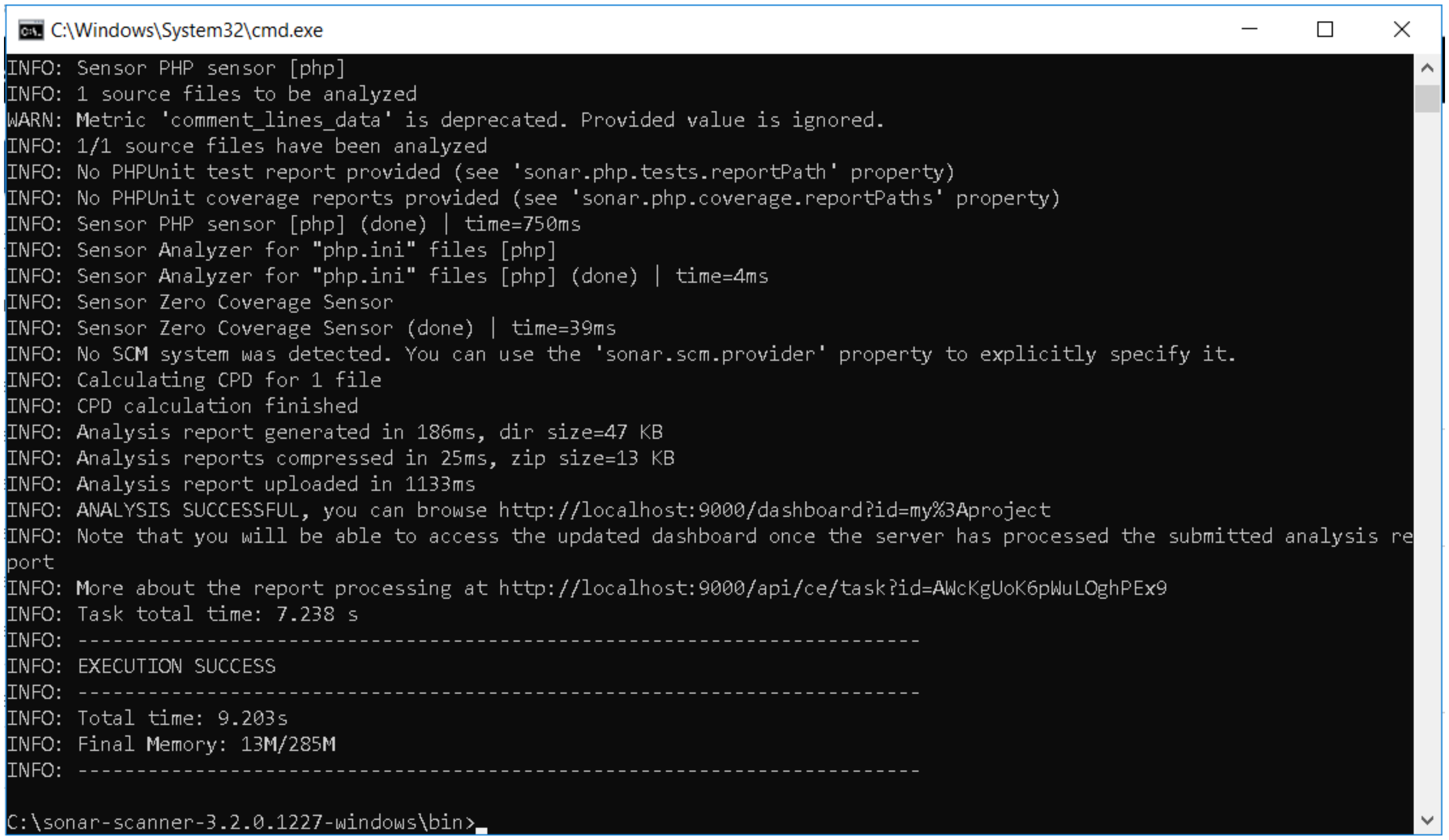

C:\sonar-scanner-3.2.0.1227-windows\conf\sonar-scanner.properties - Notepad++
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Herramientas Macro Ejecutar Plugins Ventana ?
sonar-scanner.properties
Properties file
length : 533 lines : 12
Ln : 9 Col : 18 Sel : 0 | 0
Unix (LF)
UTF-8
INS

```
# must be unique in a given SonarQube instance
sonar.projectKey=my:project
# this is the name and version displayed in the SonarQube UI. Was mandatory prior to SonarQube 6.1.
sonar.projectName=My project
sonar.projectVersion=1.0

# Path is relative to the sonar-project.properties file. Replace "\" by "/" on Windows.
# This property is optional if sonar.modules is set.
sonar.sources=C:/sonar-scanner-3.2.0.1227-windows/bin/Prueba.php

# Encoding of the source code. Default is default system encoding
#sonar.sourceEncoding=UTF-8
```

4. Ingresamos a la consola CMD y accedemos a la siguiente ubicación: C:\sonar-scanner-3.2.0.1227-windows\bin y ejecutamos el sonar-scanner.bat

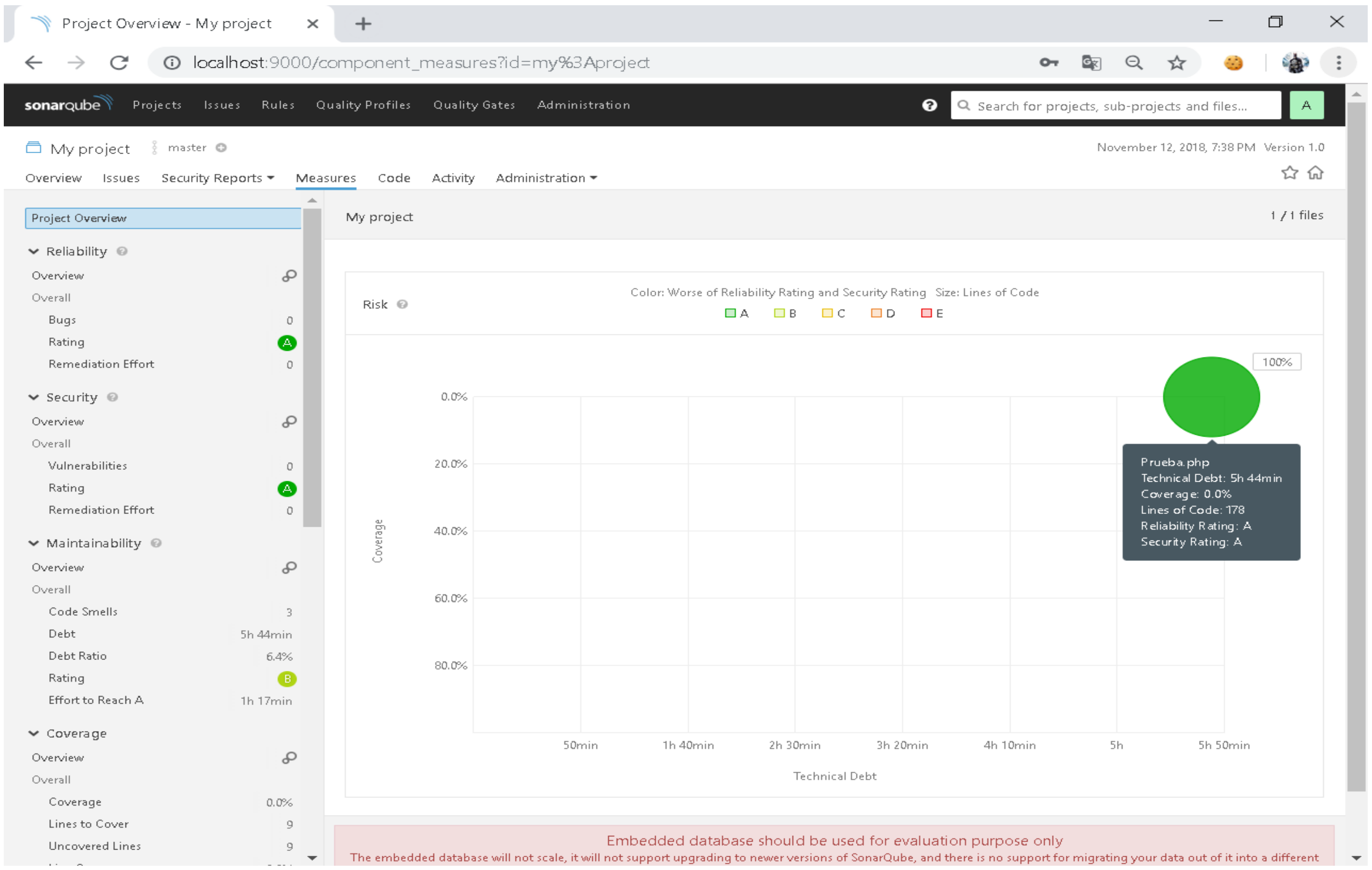


```
Microsoft Windows [Versión 10.0.17134.345]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\sonar-scanner-3.2.0.1227-windows\bin>sonar-scanner.bat
```



```
INFO: Sensor PHP sensor [php]
INFO: 1 source files to be analyzed
WARN: Metric 'comment_lines_data' is deprecated. Provided value is ignored.
INFO: 1/1 source files have been analyzed
INFO: No PHPUnit test report provided (see 'sonar.php.tests.reportPath' property)
INFO: No PHPUnit coverage reports provided (see 'sonar.php.coverage.reportPaths' property)
INFO: Sensor PHP sensor [php] (done) | time=750ms
INFO: Sensor Analyzer for "php.ini" files [php]
INFO: Sensor Analyzer for "php.ini" files [php] (done) | time=4ms
INFO: Sensor Zero Coverage Sensor
INFO: Sensor Zero Coverage Sensor (done) | time=39ms
INFO: No SCM system was detected. You can use the 'sonar.scm.provider' property to explicitly specify it.
INFO: Calculating CPD for 1 file
INFO: CPD calculation finished
INFO: Analysis report generated in 186ms, dir size=47 KB
INFO: Analysis reports compressed in 25ms, zip size=13 KB
INFO: Analysis report uploaded in 1133ms
INFO: ANALYSIS SUCCESSFUL, you can browse http://localhost:9000/dashboard?id=my%3Aproject
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis re
port
INFO: More about the report processing at http://localhost:9000/api/ce/task?id=AWcKgUoK6pWuLOghPEx9
INFO: Task total time: 7.238 s
INFO: ------------------------------------------------------------------------
INFO: EXECUTION SUCCESS
INFO: ------------------------------------------------------------------------
INFO: Total time: 9.203s
INFO: Final Memory: 13M/285M
INFO: ------------------------------------------------------------------------

C:\sonar-scanner-3.2.0.1227-windows\bin>_
```

5. Validamos la compilación del archivo prueba en la administración de SonarQube, en esta caso fue Prueba.php



Reliability
Overview
Overall
Bugs 0
Rating A
Remediation Effort 0

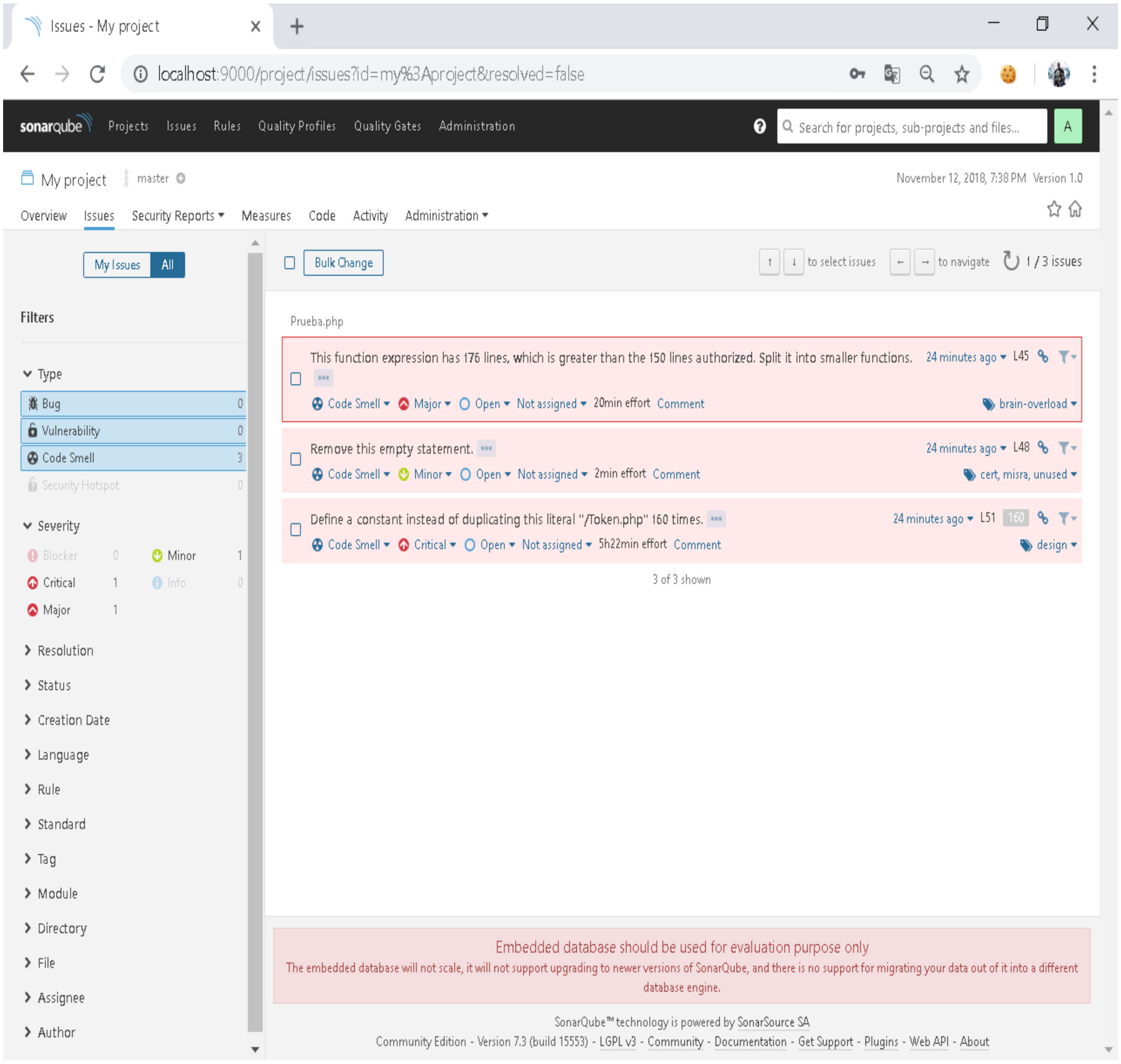
Security
Overview
Overall
Vulnerabilities 0
Rating A
Remediation Effort 0

Maintainability
Overview
Overall
Code Smells 3
Debt 5h 44min
Debt Ratio 6.4%
Rating B
Effort to Reach A 1h 17min

Coverage
Overview
Overall
Coverage 0.0%
Lines to Cover 9
Uncovered Lines 9

Prueba.php
Technical Debt: 5h 44min
Coverage: 0.0%
Lines of Code: 178
Reliability Rating: A
Security Rating: A

Embedded database should be used for evaluation purpose 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



Filters

Type

Bug	0
Vulnerability	0
Code Smell	3
Security Hotspot	0

Severity

Blocker	0	Minor	1
Critical	1	Info	0
Major	1		

Resolution
Status
Creation Date
Language
Rule
Standard
Tag
Module
Directory
File
Assignee
Author

Prueba.php

This function expression has 176 lines, which is greater than the 150 lines authorized. Split it into smaller functions. 24 minutes ago L45
Code Smell Major Open Not assigned 20min effort Comment brain-overload

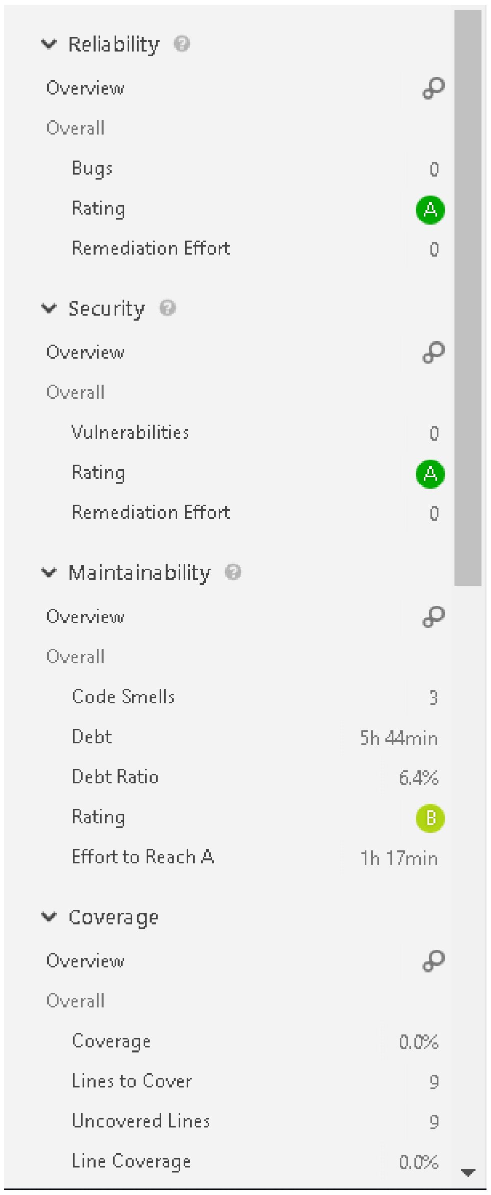
Remove this empty statement. 24 minutes ago L48
Code Smell Minor Open Not assigned 2min effort Comment cert, misra, unused

Define a constant instead of duplicating this literal "/Token.php" 160 times. 24 minutes ago L51 160
Code Smell Critical Open Not assigned 5h22min effort Comment design

3 of 3 shown

Embedded database should be used for evaluation purpose 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 - Version 7.3 (build 15553) - LGPL v3 - Community - Documentation - Get Support - Plugins - Web API - About



Reliability

Overview

Overall

Bugs	0
Rating	A
Remediation Effort	0

Security

Overview

Overall

Vulnerabilities	0
Rating	A
Remediation Effort	0

Maintainability

Overview

Overall

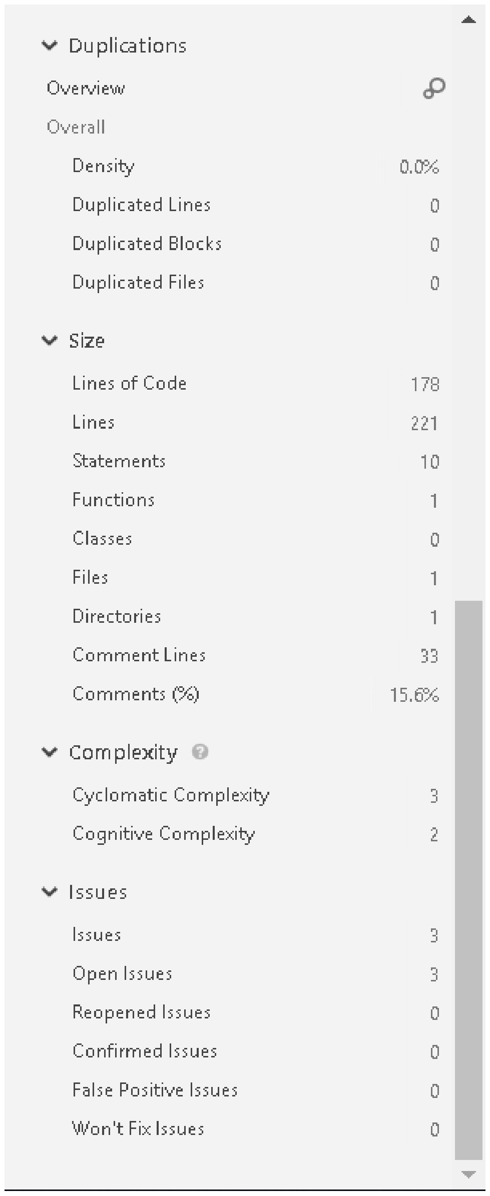
Code Smells	3
Debt	5h 44min
Debt Ratio	6.4%
Rating	B
Effort to Reach A	1h 17min

Coverage

Overview

Overall

Coverage	0.0%
Lines to Cover	9
Uncovered Lines	9
Line Coverage	0.0%



Duplications

Overview

Overall

Density	0.0%
Duplicated Lines	0
Duplicated Blocks	0
Duplicated Files	0

Size

Lines of Code	178
Lines	221
Statements	10
Functions	1
Classes	0
Files	1
Directories	1
Comment Lines	33
Comments (%)	15.6%

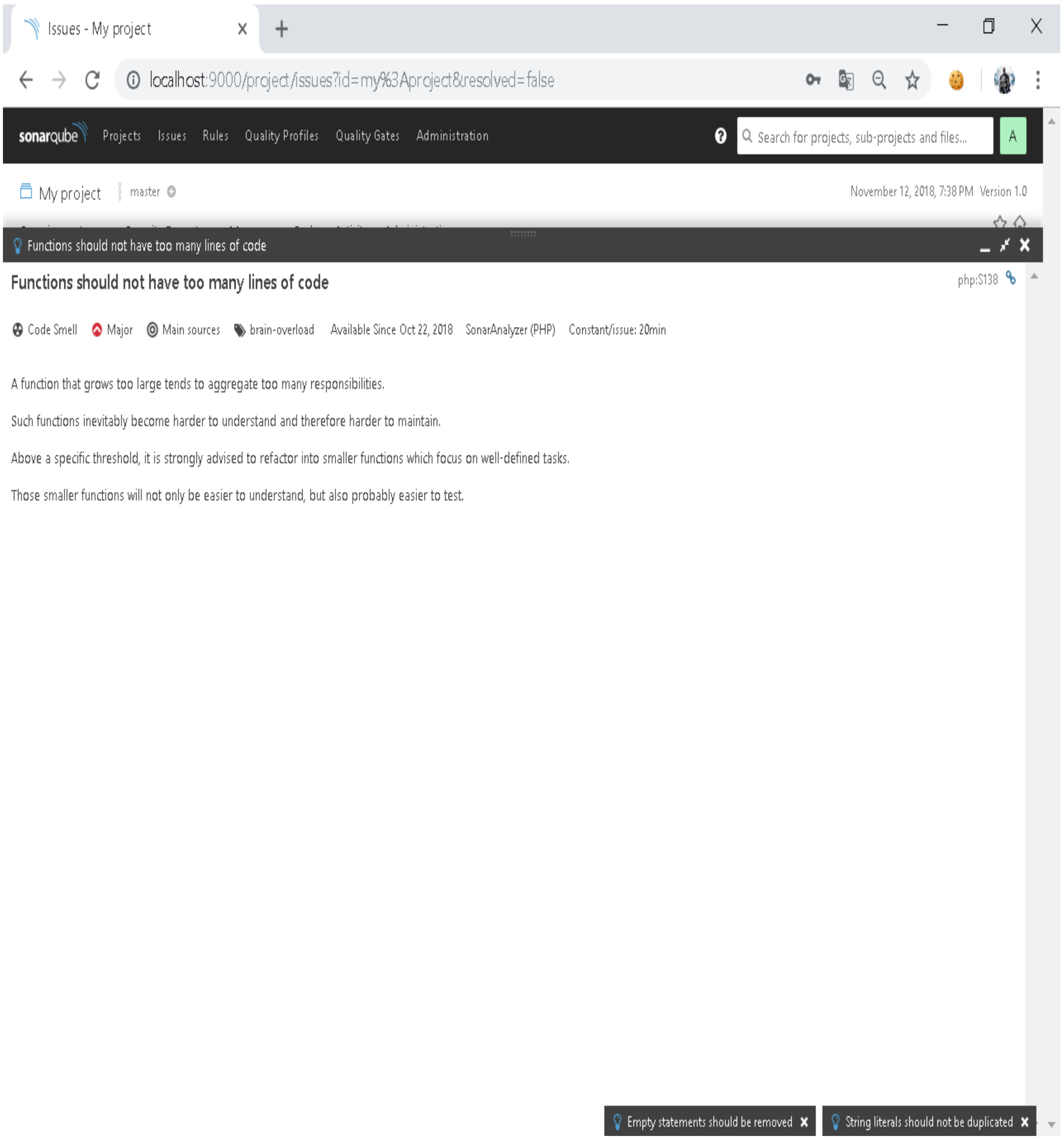
Complexity

Cyclomatic Complexity	3
Cognitive Complexity	2

Issues

Issues	3
Open Issues	3
Reopened Issues	0
Confirmed Issues	0
False Positive Issues	0
Won't Fix Issues	0

- **This function expression has 176 lines, which is greater than the 150 lines authorized. Split it into smaller functions.**


Issues - My project
localhost:9000/project/issues?id=my%3Aproject&resolved=false
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects, sub-projects and files...
My project
master
November 12, 2018, 7:38 PM Version 1.0
Functions should not have too many lines of code

Functions should not have too many lines of code

php:S138

Code Smell · Major · Main sources · brain-overload · Available Since Oct 22, 2018 · SonarAnalyzer (PHP) · Constant/issue: 20min

A function that grows too large tends to aggregate too many responsibilities.

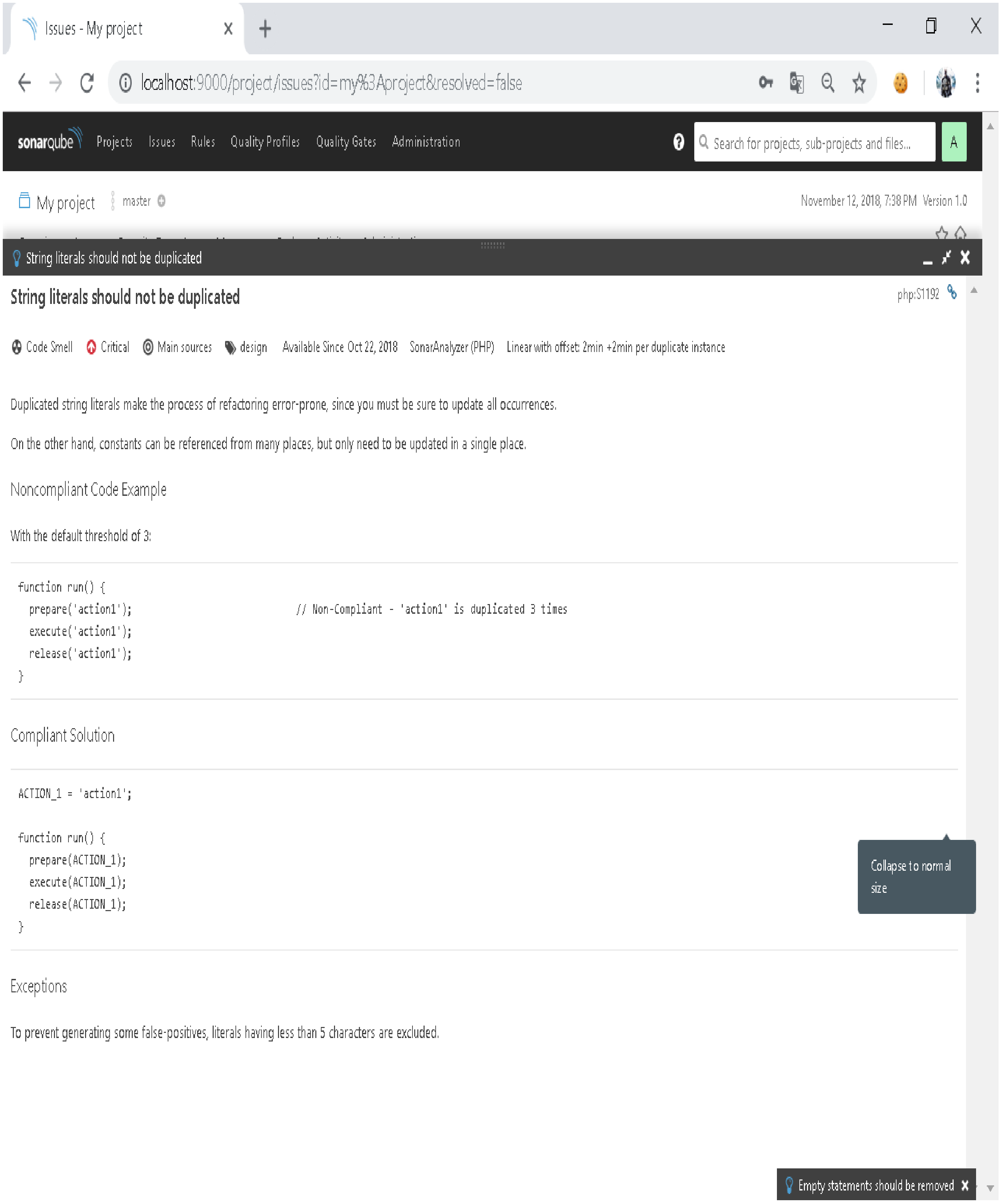
Such functions inevitably become harder to understand and therefore harder to maintain.

Above a specific threshold, it is strongly advised to refactor into smaller functions which focus on well-defined tasks.

Those smaller functions will not only be easier to understand, but also probably easier to test.

Empty statements should be removed
String literals should not be duplicated

- **Remove this empty statement.**


Issues - My project
localhost:9000/project/issues?id=my%3Aproject&resolved=false
sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration
Search for projects, sub-projects and files...
My project master
November 12, 2018, 7:38 PM Version 1.0
String literals should not be duplicated

String literals should not be duplicated

php:S1192

Code Smell Critical Main sources design Available Since Oct 22, 2018 SonarAnalyzer (PHP) Linear with offset: 2min +2min per duplicate instance

Duplicated string literals make the process of refactoring error-prone, since you must be sure to update all occurrences.

On the other hand, constants can be referenced from many places, but only need to be updated in a single place.

Noncompliant Code Example

With the default threshold of 3:

```
function run() {
  prepare('action1');                              // Non-Compliant - 'action1' is duplicated 3 times
  execute('action1');
  release('action1');
}
```

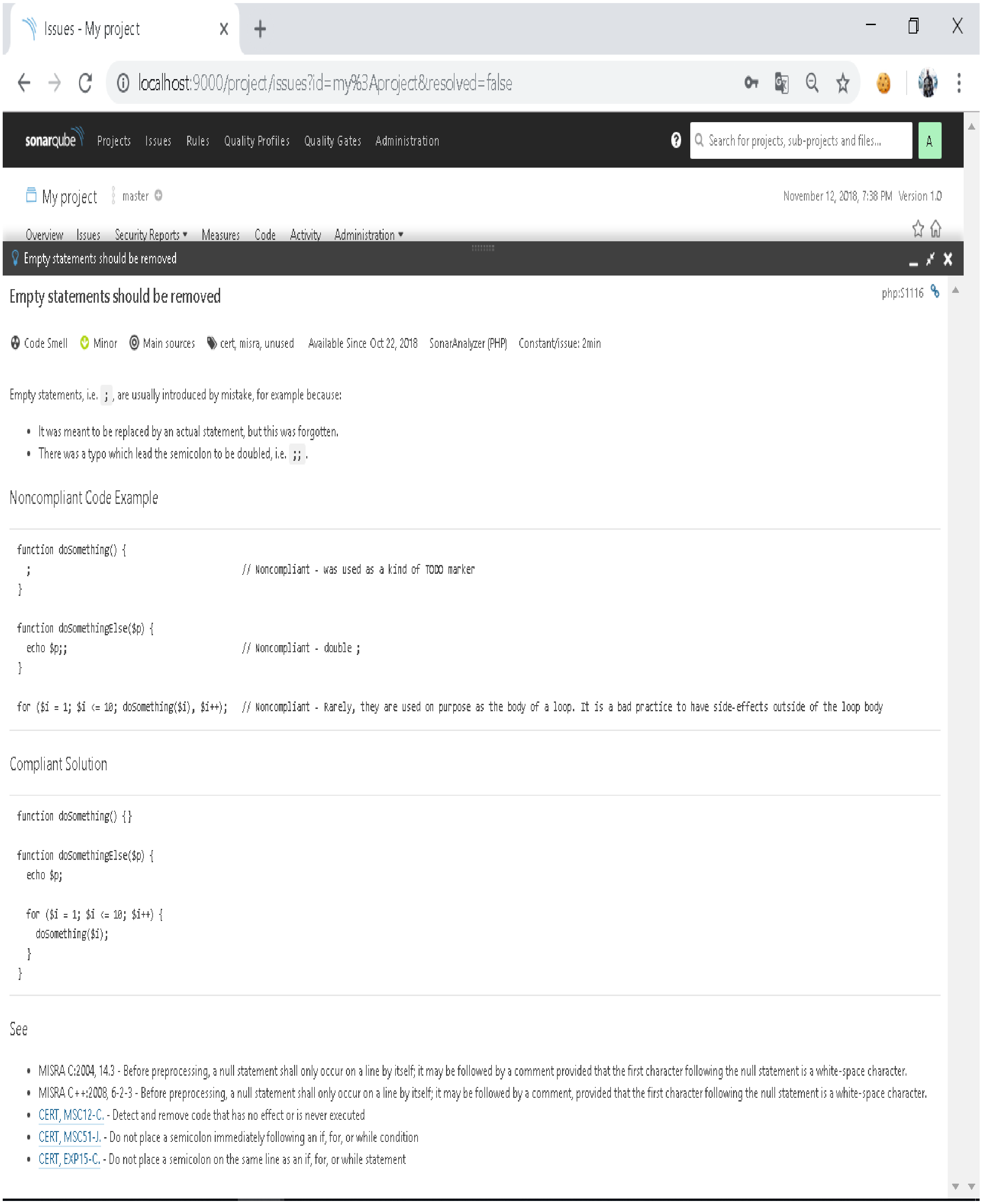
Compliant Solution

```
ACTION_1 = 'action1';

function run() {
  prepare(ACTION_1);
  execute(ACTION_1);
  release(ACTION_1);
}
```

Exceptions

To prevent generating some false-positives, literals having less than 5 characters are excluded.

- **Define a constant instead of duplicating this literal "/Token.php" 160 times.**


Issues - My project
localhost:9000/project/issues?id=my%3Aproject&resolved=false
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects, sub-projects and files...
A
My project
master
November 12, 2018, 7:38 PM Version 1.0
Overview
Issues
Security Reports
Measures
Code
Activity
Administration
Empty statements should be removed

Empty statements should be removed

php:S1116

Code Smell Minor Main sources cert, misra, unused Available Since Oct 22, 2018 SonarAnalyzer (PHP) Constant/issue: 2min

Empty statements, i.e. `;`, are usually introduced by mistake, for example because:

- It was meant to be replaced by an actual statement, but this was forgotten.
- There was a typo which lead the semicolon to be doubled, i.e. `;;`.

Noncompliant Code Example

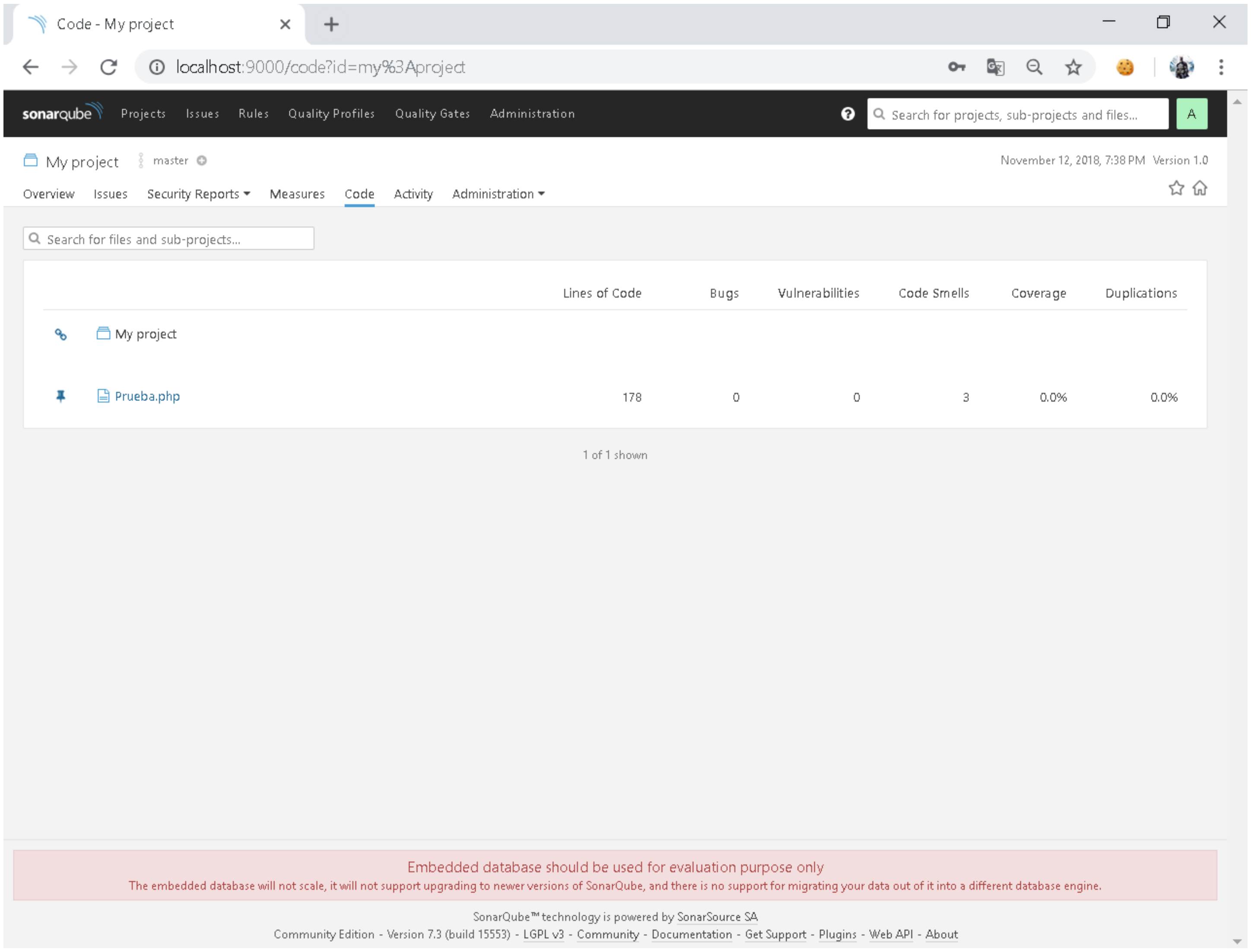
```
function doSomething() {
  ;                                         // Noncompliant - was used as a kind of TODO marker
}

function doSomethingElse($p) {
  echo $p;;                                 // Noncompliant - double ;
}

for ($i = 1; $i <= 10; doSomething($i), $i++);   // Noncompliant - Rarely, they are used on purpose as the body of a loop. It is a bad practice to have side-effects outside of the loop body
```

Compliant Solution

```
function doSomething() {}

function doSomethingElse($p) {
  echo $p;

  for ($i = 1; $i <= 10; $i++) {
    doSomething($i);
  }
}
```

See

- MISRA C:2004, 14.3 - Before preprocessing, a null statement shall only occur on a line by itself; it may be followed by a comment provided that the first character following the null statement is a white-space character.
- MISRA C++:2008, 6-2-3 - Before preprocessing, a null statement shall only occur on a line by itself; it may be followed by a comment, provided that the first character following the null statement is a white-space character.
- CERT, MSC12-C. - Detect and remove code that has no effect or is never executed
- CERT, MSC51-J. - Do not place a semicolon immediately following an if, for, or while condition
- CERT, EXP15-C. - Do not place a semicolon on the same line as an if, for, or while statement


Code - My project

localhost:9000/code?id=my%3Aproject

My project master

November 12, 2018, 7:38 PM Version 1.0

	Lines of Code	Bugs	Vulnerabilities	Code Smells	Coverage	Duplications
My project						
Prueba.php	178	0	0	3	0.0%	0.0%

1 of 1 shown

Embedded database should be used for evaluation purpose 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 - Version 7.3 (build 15553) - LGPL v3 - Community - Documentation - Get Support - Plugins - Web API - About