Behavioral metrics of SonarQube

Improve your Java(Script) even further

Johan Janssen (Info Support) @johanjanssen42

Agenda

- Basic features
- Analyzing data
- Code review
- Hunting bad design and architecture
- Testing
- Other languages
- Leftovers
- Using SonarQube on existing projects
- Tips / summary
- Questions

Quality for thousands of years

Quality after years of improvements

Quality for 39 days

Bugs

- Microsoft Applications: "about 10 - 20 defects per 1000 lines of code during in-house testing, and 0.5 defect per 1000 lines of code in released product (Dave Moore 1992).“

Bugs

- NASA's Spirit rover became unresponsive on January 21, 2004, a few weeks after landing on Mars. (Wikipedia)

- In January 2009, Google's search engine erroneously notified users that every web site world wide was potentially malicious, including its own. (Wikipedia)

Find them as fast as possible

Why should I use SonarQube?

- I'm told to use it.
- We need to achieve certain results (SIG…)
- I want to improve my coding standards

SonarQube

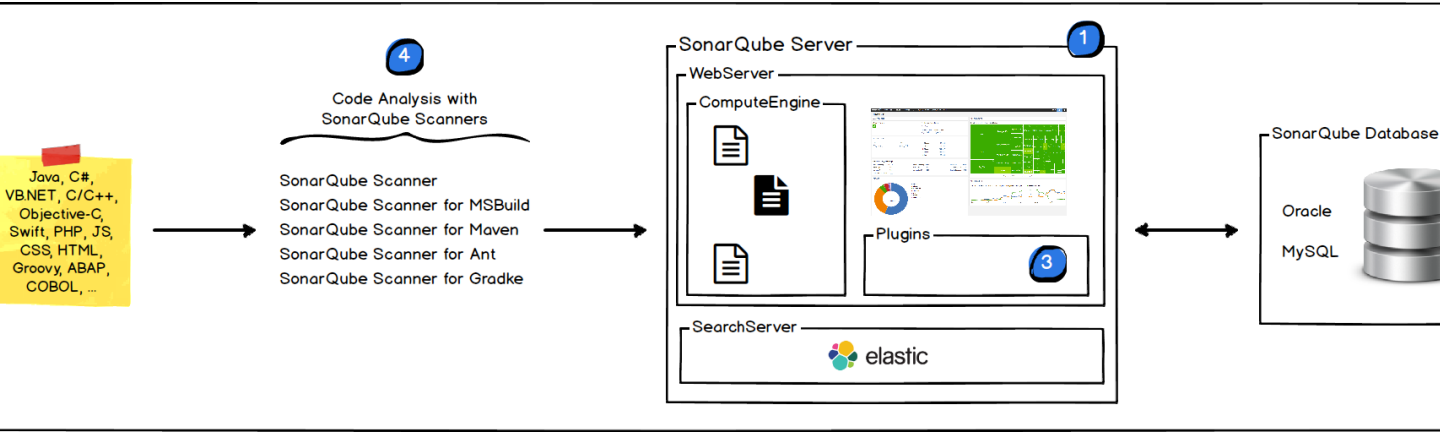
- Platform to manage code quality
- Open source, possible to pay for support and some plugins
- Since 2006; now they have 200 customers and SonarQube is used in 15.000 organizations
- Active community: support, plugins, books

SonarQube

2013:

- 5 releases of SonarQube platform
- 130 releases of ecosystem products
- 75,000 downloads of SonarQube
- 13,000+ messages on mailing lists

SonarQube architecture



SonarQube basic features

Home

Configure widgets

Welcome to SonarQube Dashboard

Since you are able to read this, it means that you have successfully started your SonarQube server. Well done!

If you have not removed this text, it also means that you have not yet played much with SonarQube. So here are a few pointers for your next step:

- Do you now want to run analysis on a project?
- Maybe start customizing dashboards?
- Or simply browse the complete documentation?
- If you have a question or an issue, please visit the Get Support page.

MY FAVOURITES

QG	NAME	LAST ANALYSIS
No data		

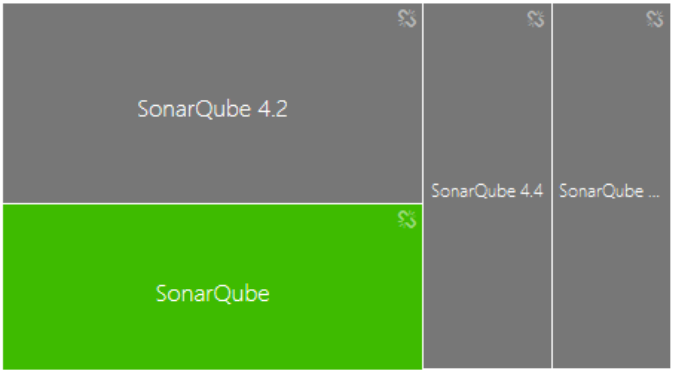
PROJECTS

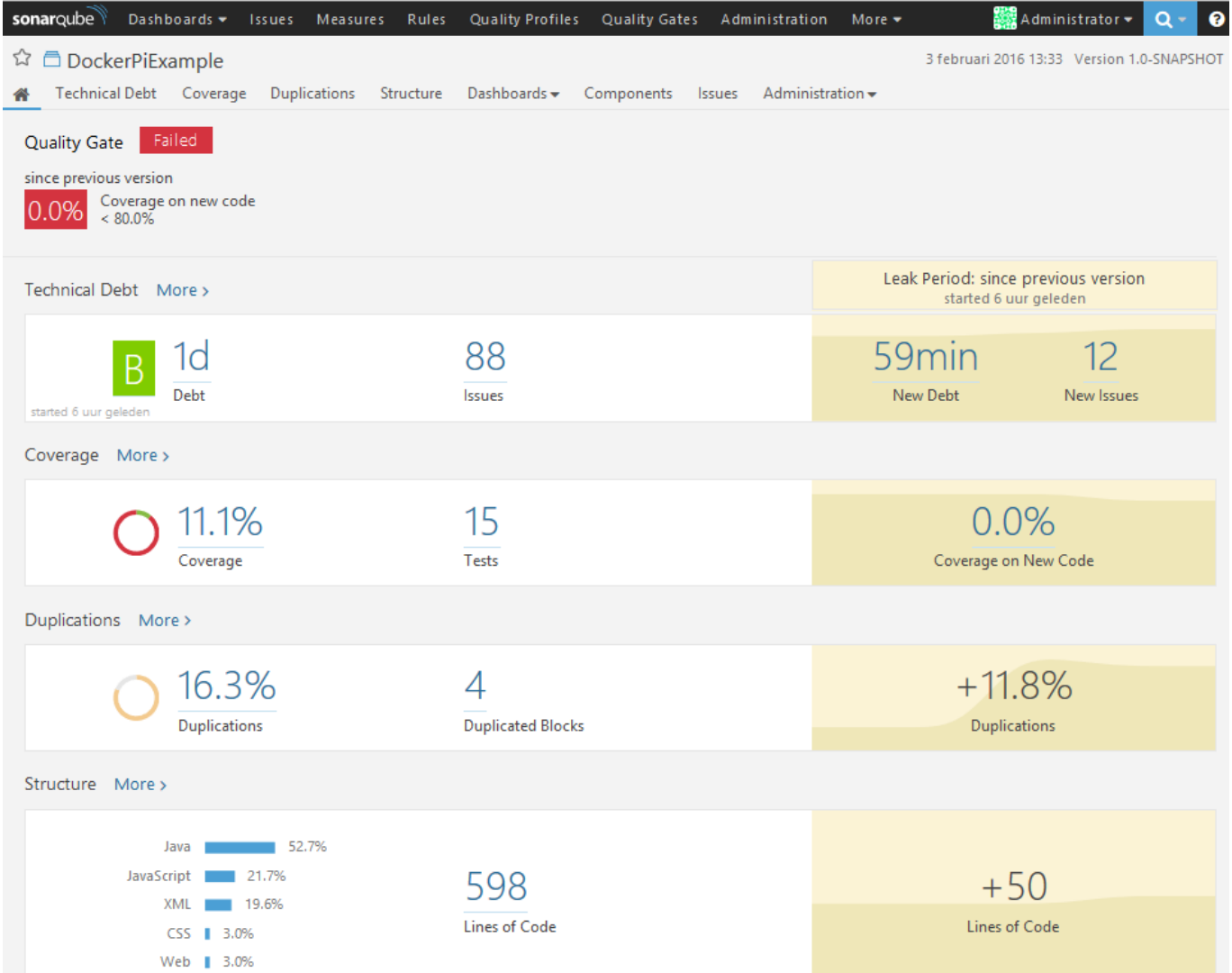
QG	NAME	VERSION	LOC	TECHNICAL DEBT	LAST ANALYSIS
✗	DockerPiExample	1.0-SNAPSHOT	598	1d 7h	13:33
✓	SonarQube	5.3.1-SNAPSHOT	171.343	47d	12:53
✓	SonarQube 4.2	4.2.1-SNAPSHOT	203.921	382d	11:07
✗	SonarQube 4.3	4.3.4-SNAPSHOT	104.833	41d	10:35
✗	SonarQube 4.4	4.4.1-SNAPSHOT	114.901	45d	10:42
✓	TestApplication	1.0-SNAPSHOT	225	4h 3min	10:37

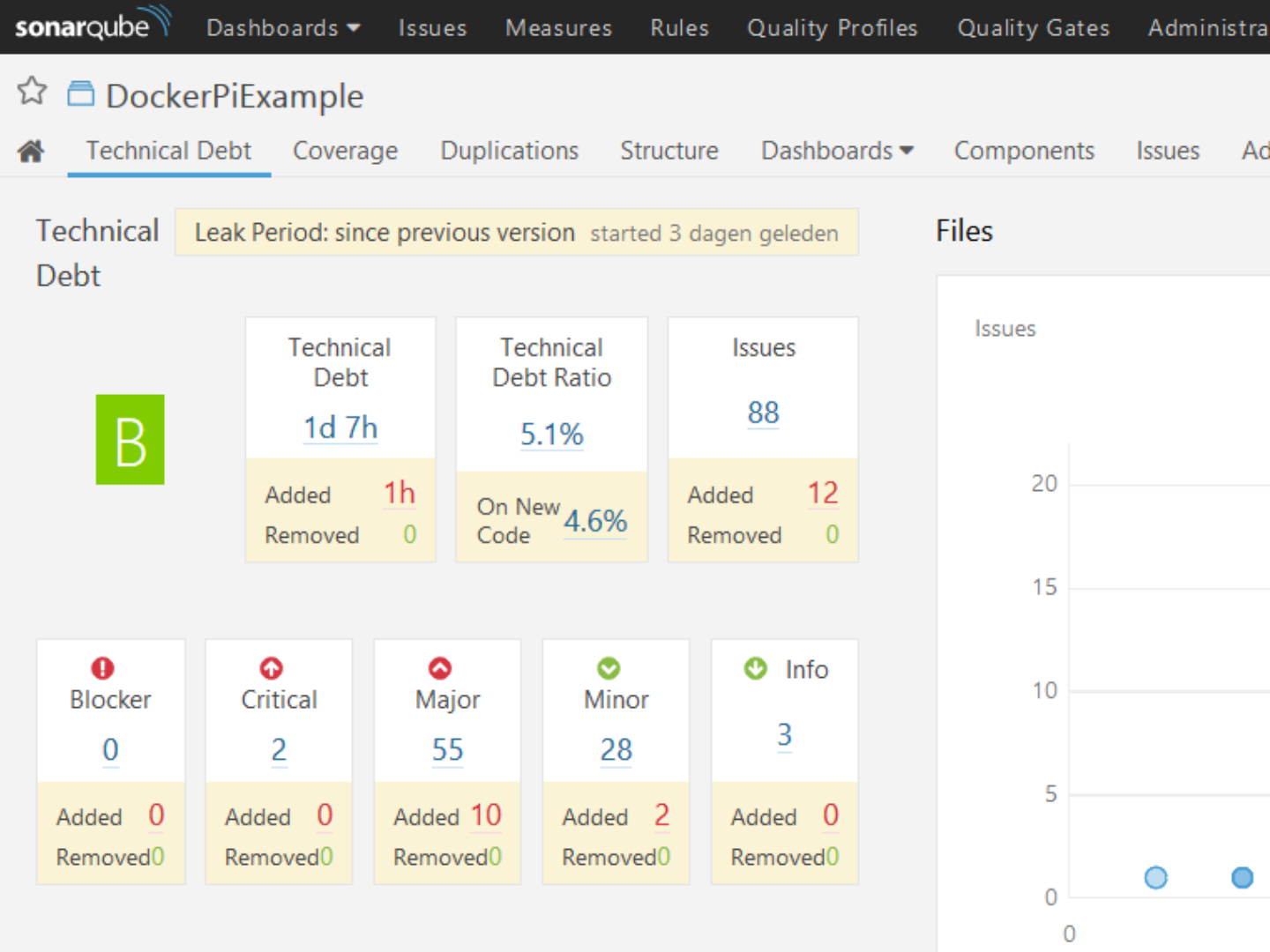
6 results

PROJECTS

Size: Lines of code Color: Coverage




sonarqube
Dashboards
Issues
Measures
Rules
Quality Profiles
Quality Gates
Administration
More
Administrator
DockerPiExample
3 februari 2016 13:33
Version 1.0-SNAPSHOT
Technical Debt
Coverage
Duplications
Structure
Dashboards
Components
Issues
Administration
Quality Gate
Failed
since previous version
0.0%
Coverage on new code
< 80.0%
Technical Debt
More ›
Leak Period: since previous version
started 6 uur geleden
B
1d
Debt
started 6 uur geleden
88
Issues
59min
New Debt
12
New Issues
Coverage
More ›
11.1%
Coverage
15
Tests
0.0%
Coverage on New Code
Duplications
More ›
16.3%
Duplications
4
Duplicated Blocks
+11.8%
Duplications
Structure
More ›
Java
52.7%
JavaScript
21.7%
XML
19.6%
CSS
3.0%
Web
3.0%
598
Lines of Code
+50
Lines of Code


sonarqube
Dashboards
Issues
Measures
Rules
Quality Profiles
Quality Gates
Administra
DockerPiExample
Technical Debt
Coverage
Duplications
Structure
Dashboards
Components
Issues
Technical Debt
Leak Period: since previous version started 3 dagen geleden
B
Technical Debt
1d 7h
Added 1h
Removed 0
Technical Debt Ratio
5.1%
On New Code 4.6%
Issues
88
Added 12
Removed 0
Blocker
0
Added 0
Removed0
Critical
2
Added 0
Removed0
Major
55
Added 10
Removed0
Minor
28
Added 2
Removed0
Info
3
Added 0
Removed0
Files
Issues
20
15
10
5
0
0

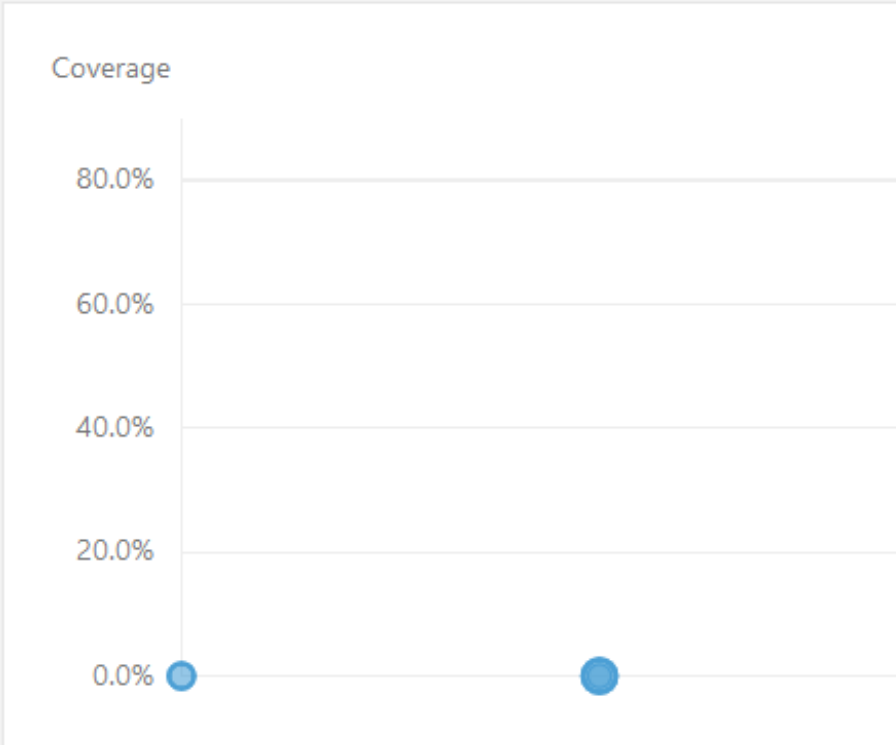
DockerPiExample

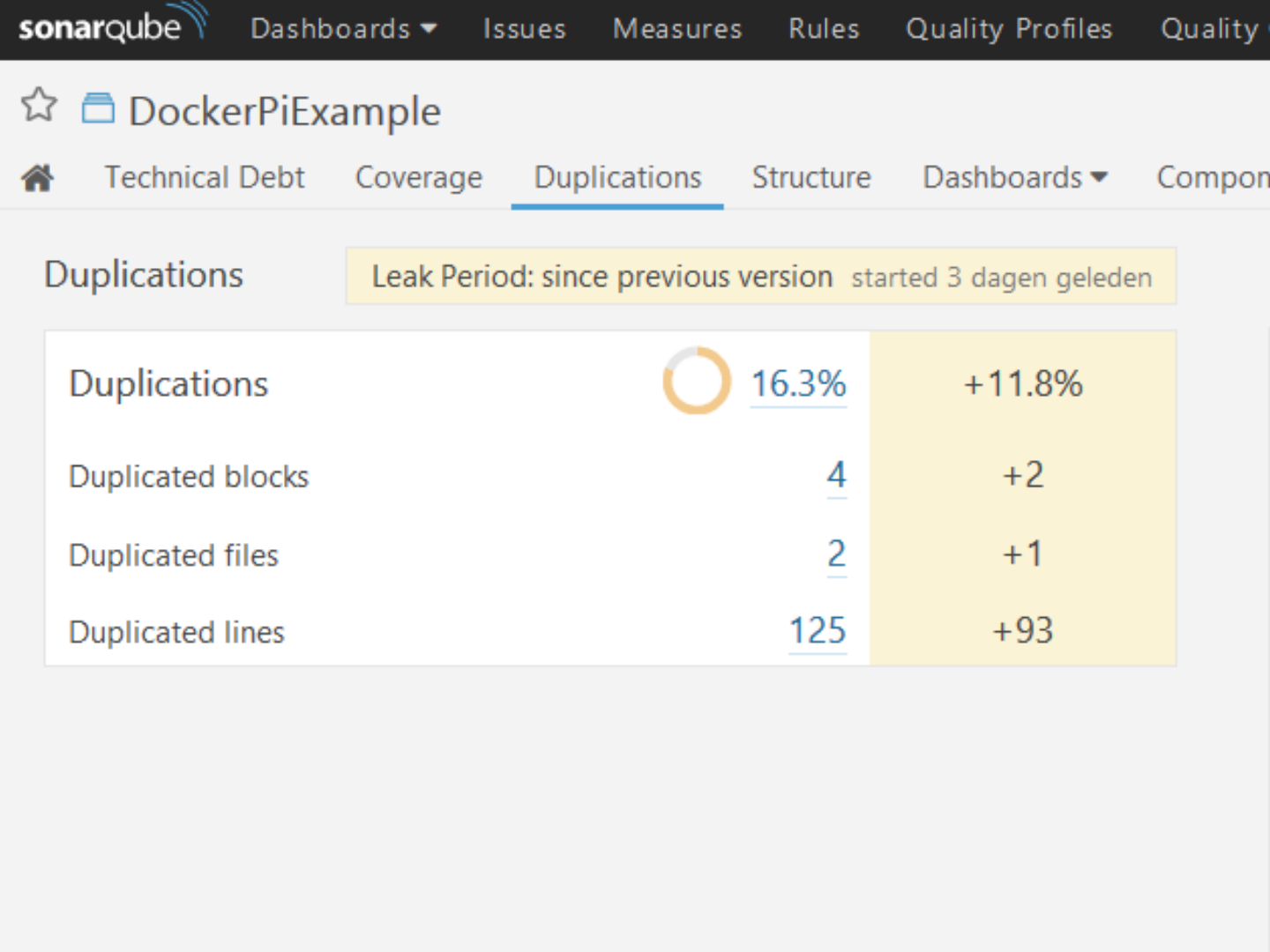
Coverage

Leak Period: since previous version started 3 dagen geleden

Coverage	11.1%	-0.9%
Line coverage	12.8%	-0.9%
Uncovered lines	116	+9
Lines to cover	133	+9

Files




sonarqube
Dashboards
Issues
Measures
Rules
Quality Profiles
Quality
DockerPiExample
Technical Debt
Coverage
Duplications
Structure
Dashboards
Compon
Duplications
Leak Period: since previous version started 3 dagen geleden
Duplications
16.3%
+11.8%
Duplicated blocks
4
+2
Duplicated files
2
+1
Duplicated lines
125
+93

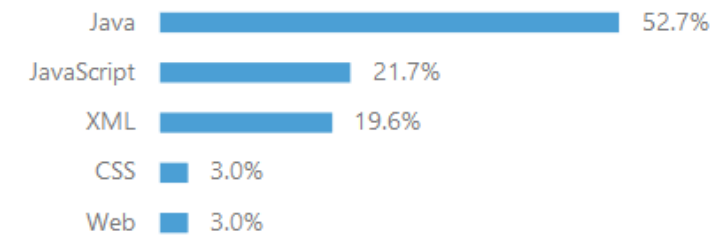
sonarqube Dashboards Issues Measures Rules Quality Profiles Quality Gates Administration

DockerPiExample

Technical Debt Coverage Duplications Structure Dashboards Components Issues Administra

Structure

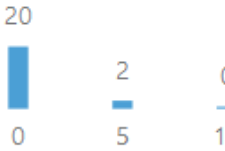
Lines of code	598	+50
Classes	20	+1
Directories	12	+0
Files	26	+1
Functions	46	+1
Lines	769	+62
Statements	227	+7



Complexity

Complexity /function

Complexity /file

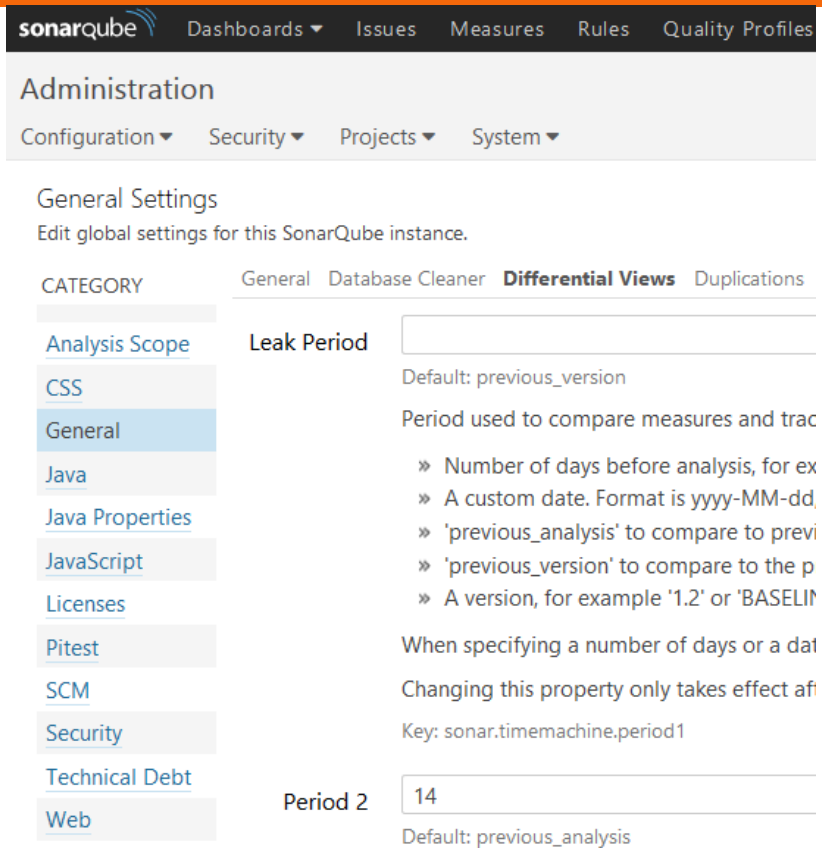


Complexity /class

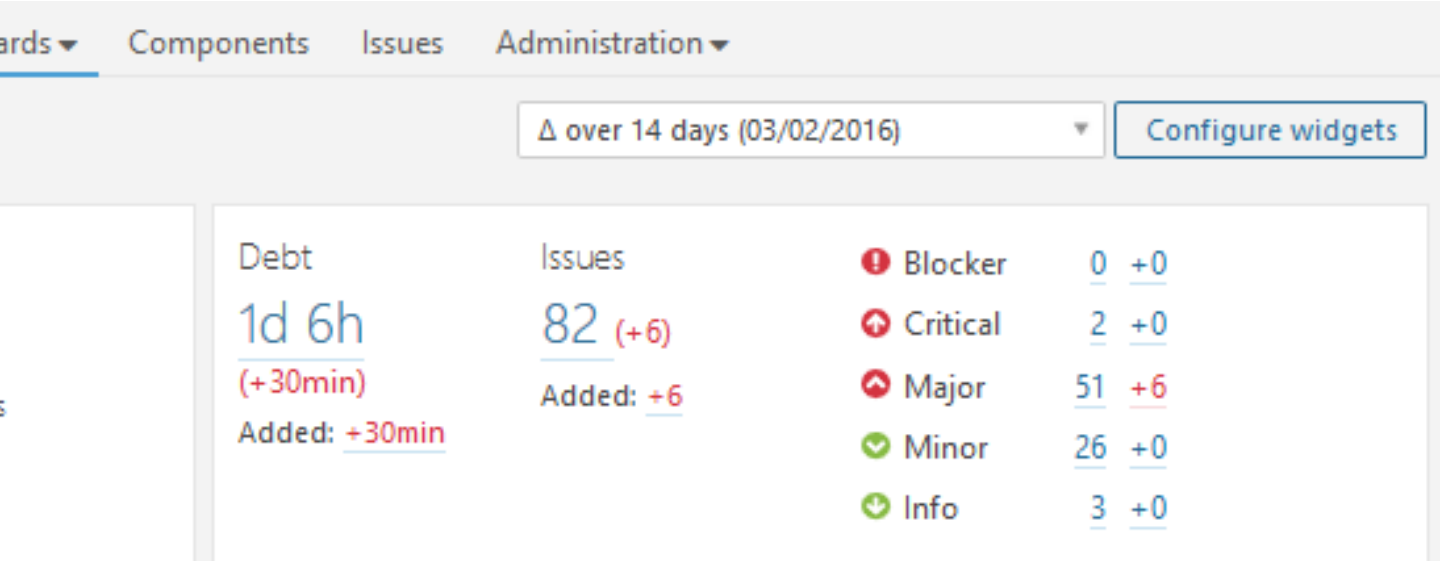
Complexity	84	+4
Complexity /function	1.8	+0.0
Complexity /file	3.5	+0.0

Complexity /class	3.2	+0.1

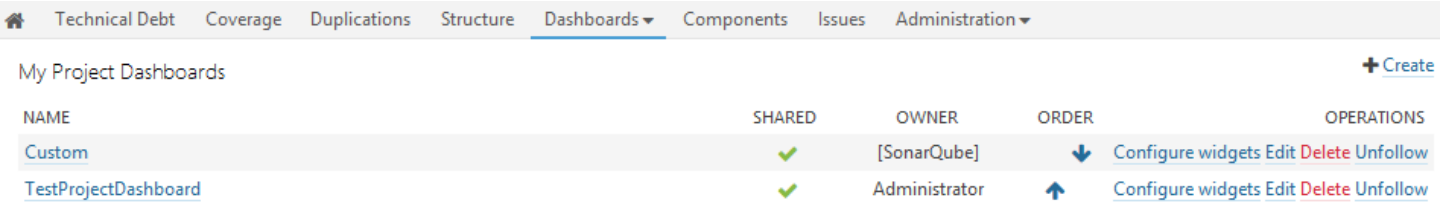
Differential views Leak Period



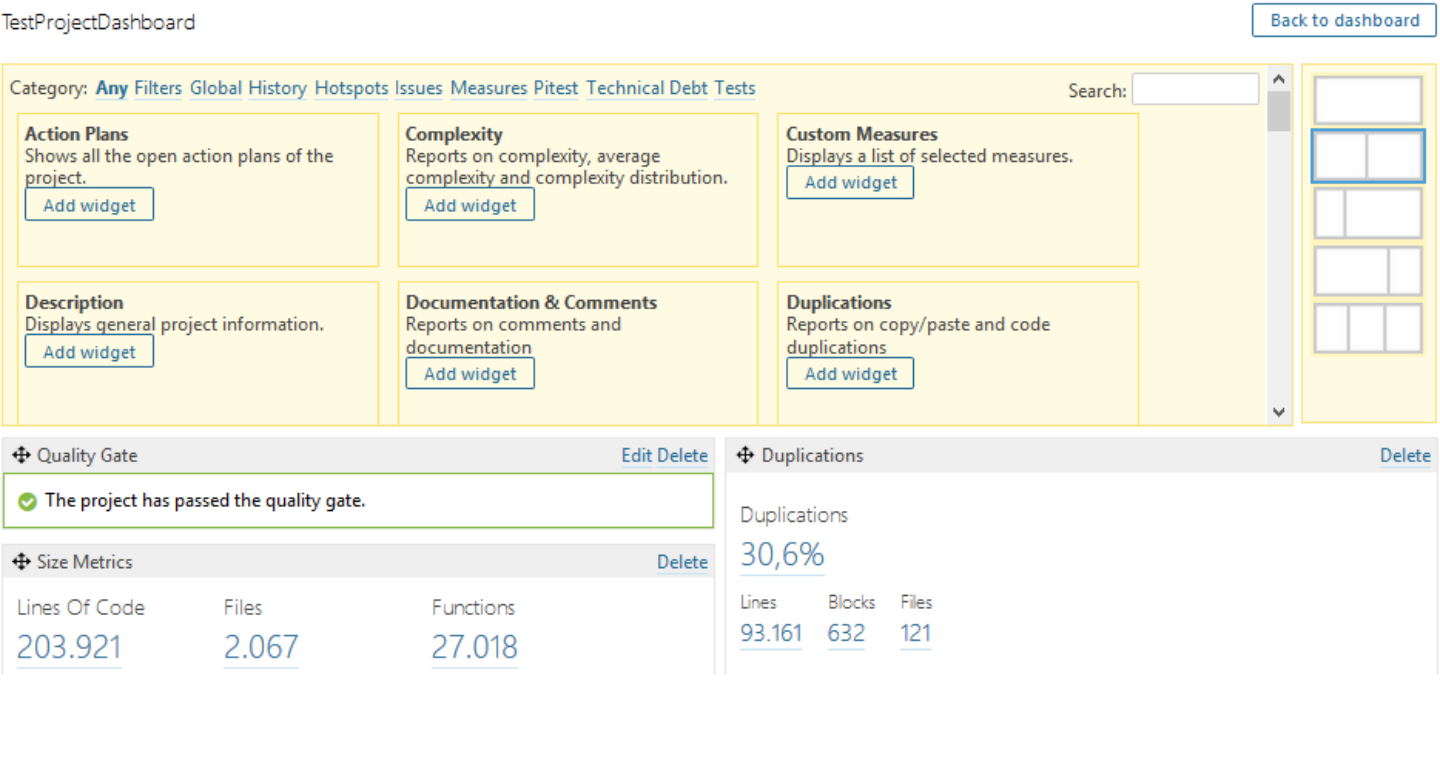
Differential views



Configure dashboards



Configure dashboards



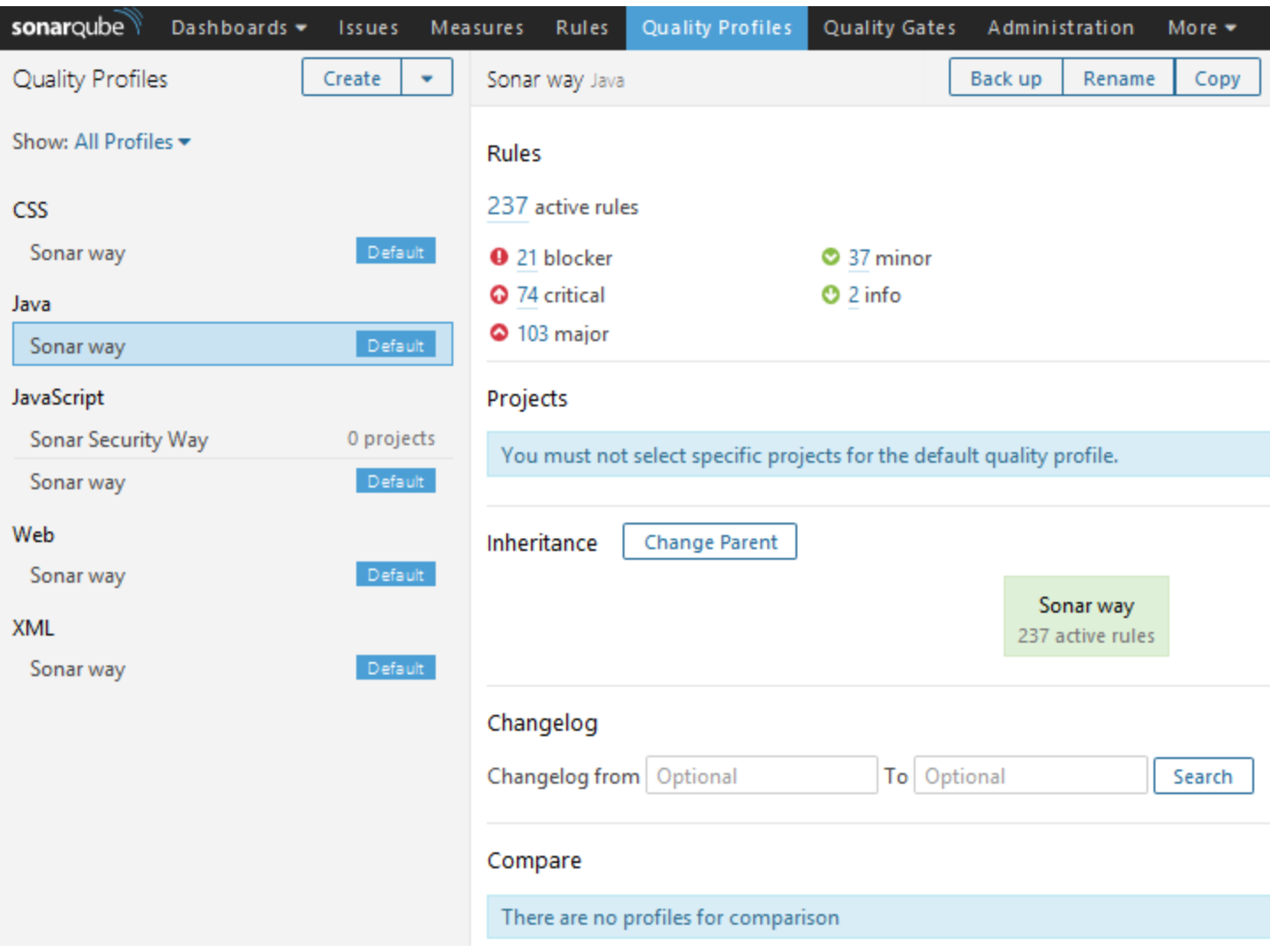
Overview

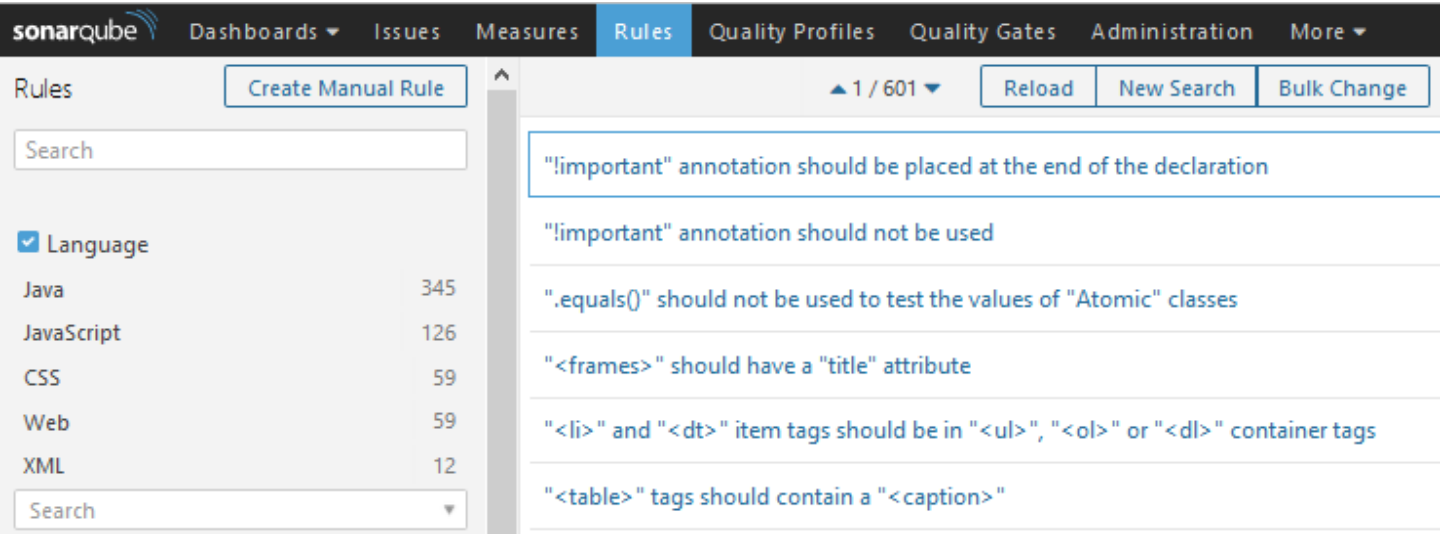
Profile

Rules

Projects

THE PROJECT


sonarqube
Dashboards
Issues
Measures
Rules
Quality Profiles
Quality Gates
Administration
More
Quality Profiles
Create
Show: All Profiles
CSS
Sonar way
Default
Java
Sonar way
Default
JavaScript
Sonar Security Way
0 projects
Sonar way
Default
Web
Sonar way
Default
XML
Sonar way
Default
Sonar way Java
Back up
Rename
Copy
Rules
237 active rules
21 blocker
74 critical
103 major
37 minor
2 info
Projects
You must not select specific projects for the default quality profile.
Inheritance
Change Parent
Sonar way
237 active rules
Changelog
Changelog from
Optional
To
Optional
Search
Compare
There are no profiles for comparison


sonarqube
Dashboards
Issues
Measures
Rules
Quality Profiles
Quality Gates
Administration
More
Rules
Create Manual Rule
Search
Language
Java 345
JavaScript 126
CSS 59
Web 59
XML 12
Search
1 / 601
Reload
New Search
Bulk Change
"!important" annotation should be placed at the end of the declaration
"!important" annotation should not be used
".equals()" should not be used to test the values of "Atomic" classes
"<frames>" should have a "title" attribute
"<li>" and "<dt>" item tags should be in "<ul>", "<ol>" or "<dl>" container tags
"<table>" tags should contain a "<caption>"

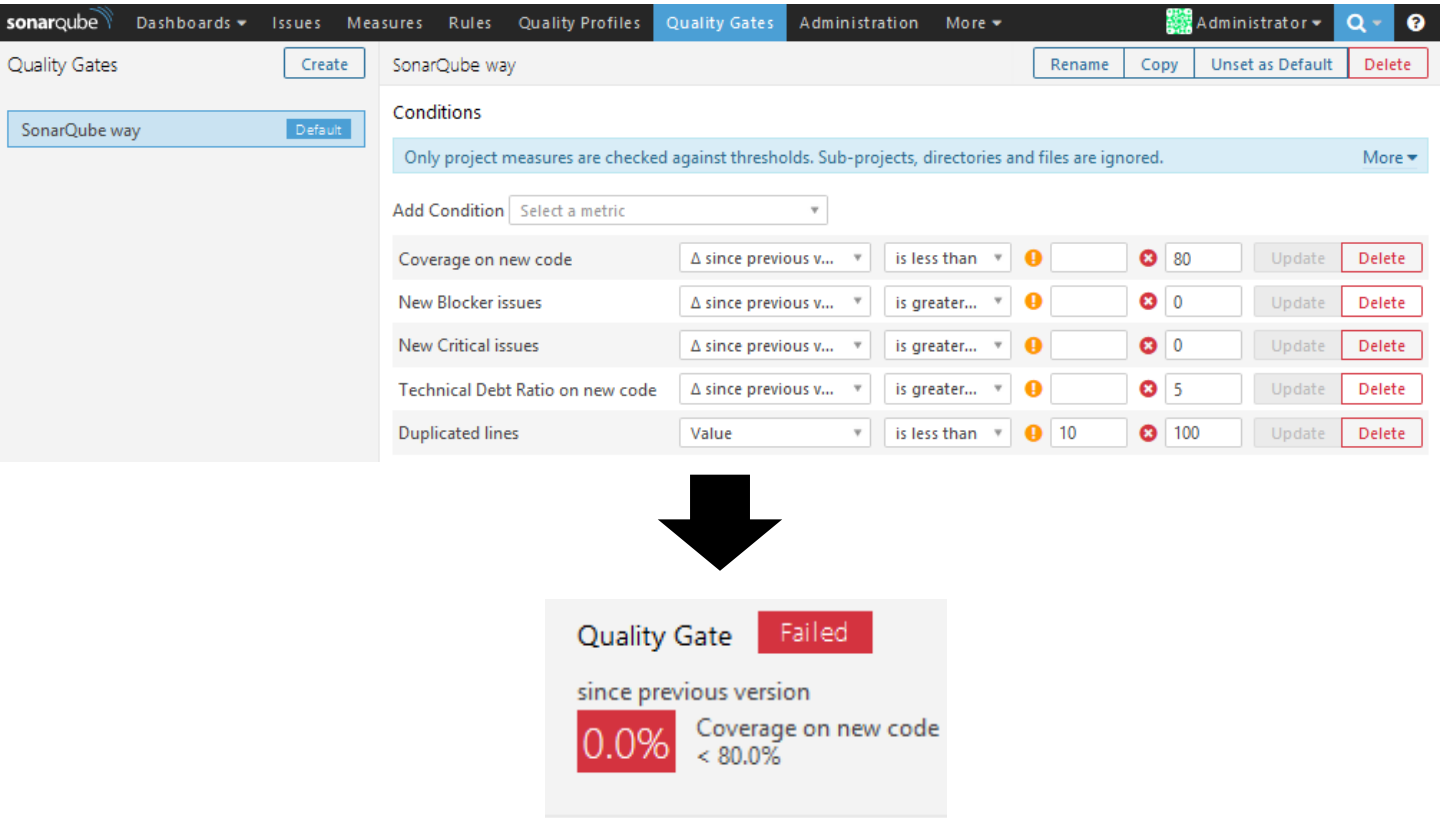
Overview

Projects

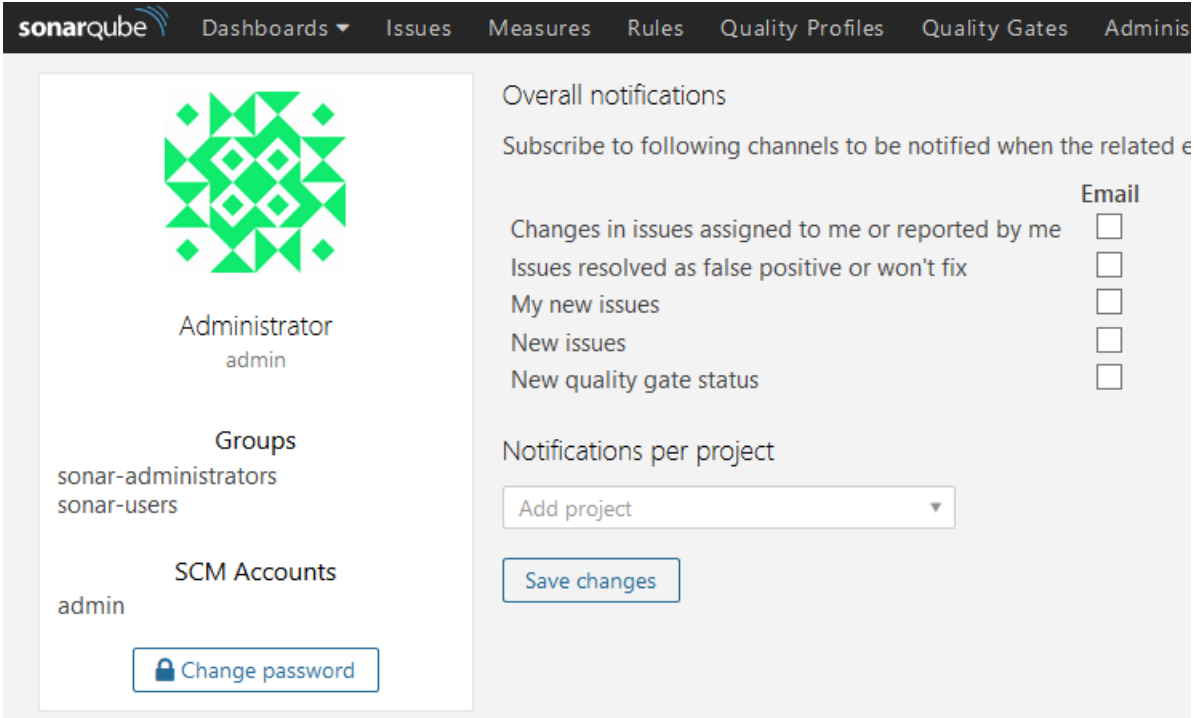
THE PROJECT

Quality Gates

Quality Gates

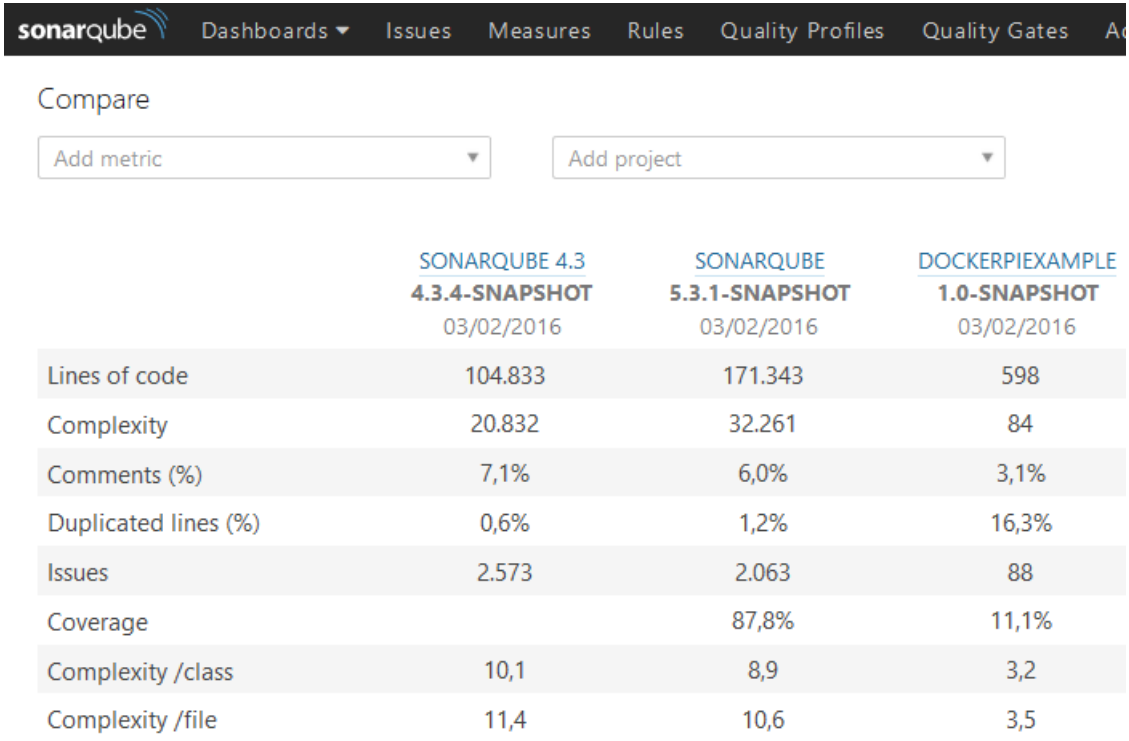


Notifications



Analysing data

Compare

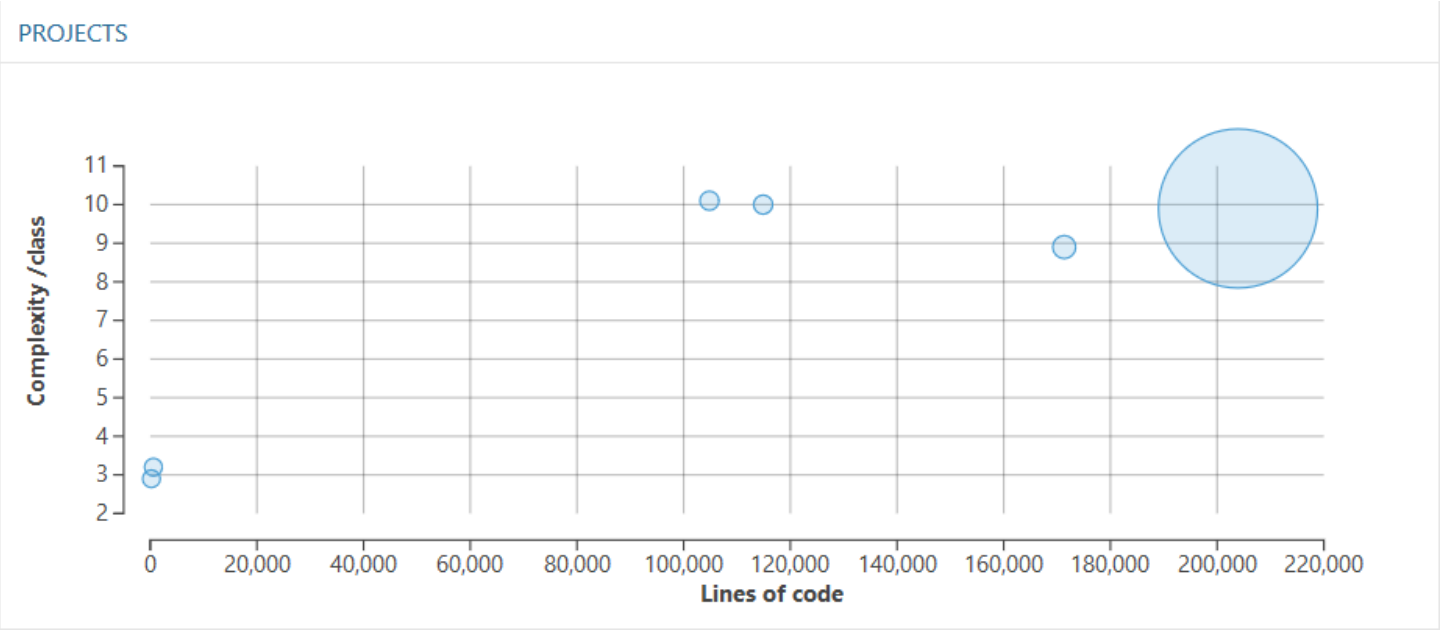


	SONARQUBE 4.3 **4.3.4-SNAPSHOT** 03/02/2016	SONARQUBE **5.3.1-SNAPSHOT** 03/02/2016	DOCKERPIEXAMPLE **1.0-SNAPSHOT** 03/02/2016
Lines of code	104.833	171.343	598
Complexity	20.832	32.261	84
Comments (%)	7,1%	6,0%	3,1%
Duplicated lines (%)	0,6%	1,2%	16,3%
Issues	2.573	2.063	88
Coverage		87,8%	11,1%
Complexity /class	10,1	8,9	3,2
Complexity /file	11,4	10,6	3,5

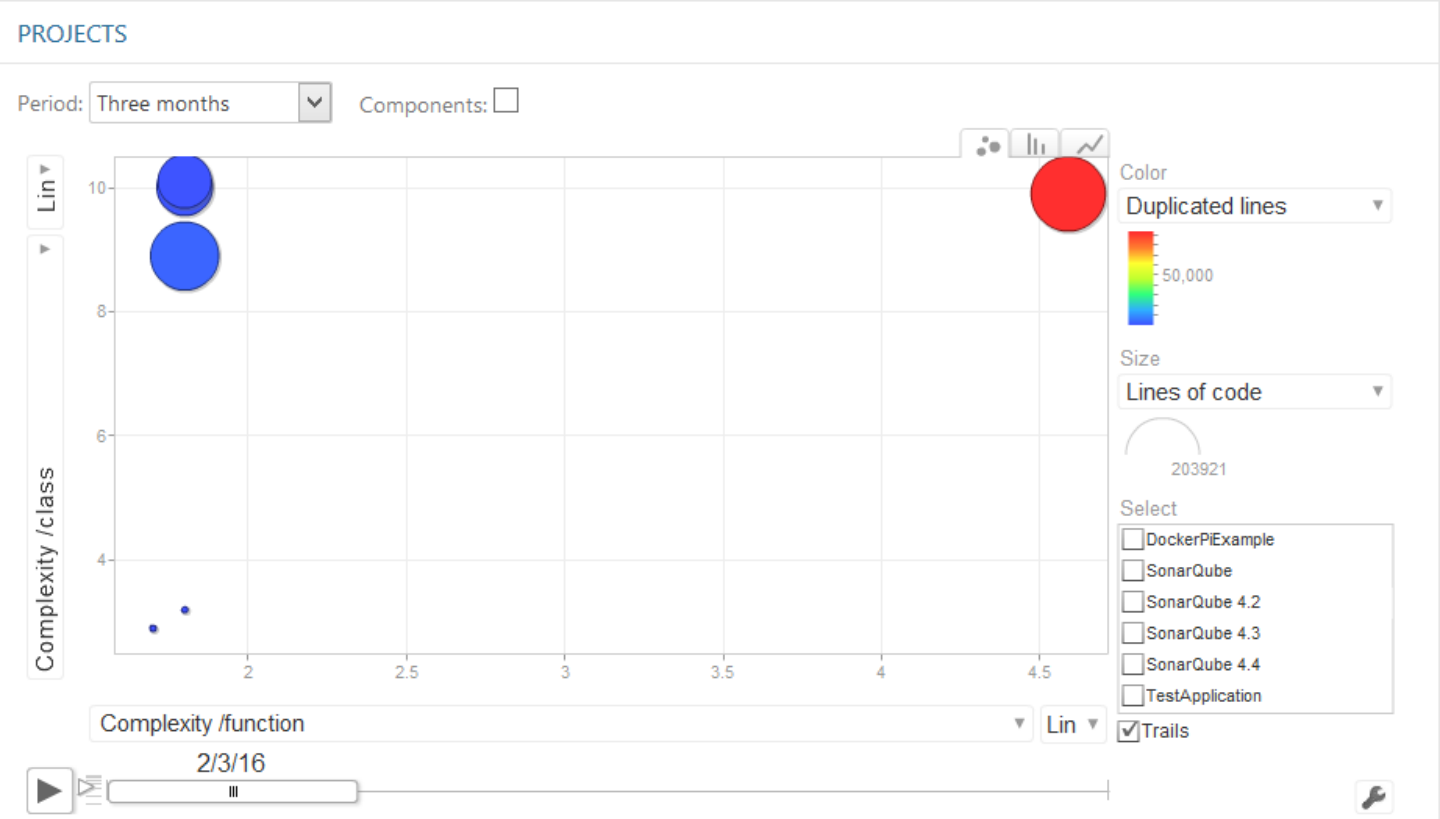
Measures

Dashboards
Issues
Measures
Rules
Quality Profiles
Quality Gates
Administration
More
Administrator
New Search
Save As
Change Columns
Components: Files
Metric: Duplicated lines > 5
+ More Criteria
Search
NAME
LAST ANALYSIS
LOC
ISSUES
src/main/java/com/dockerpi/sonar/metrics/LCOM4.java
03/02/2016
17
3
src/main/java/com/dockerpi/sonar/metrics/SomeViolations.java
03/02/2016
82
22
src/main/java/com/mycompany/testapp/examples/metrics/SomeViolations.java
03/02/2016
82
21

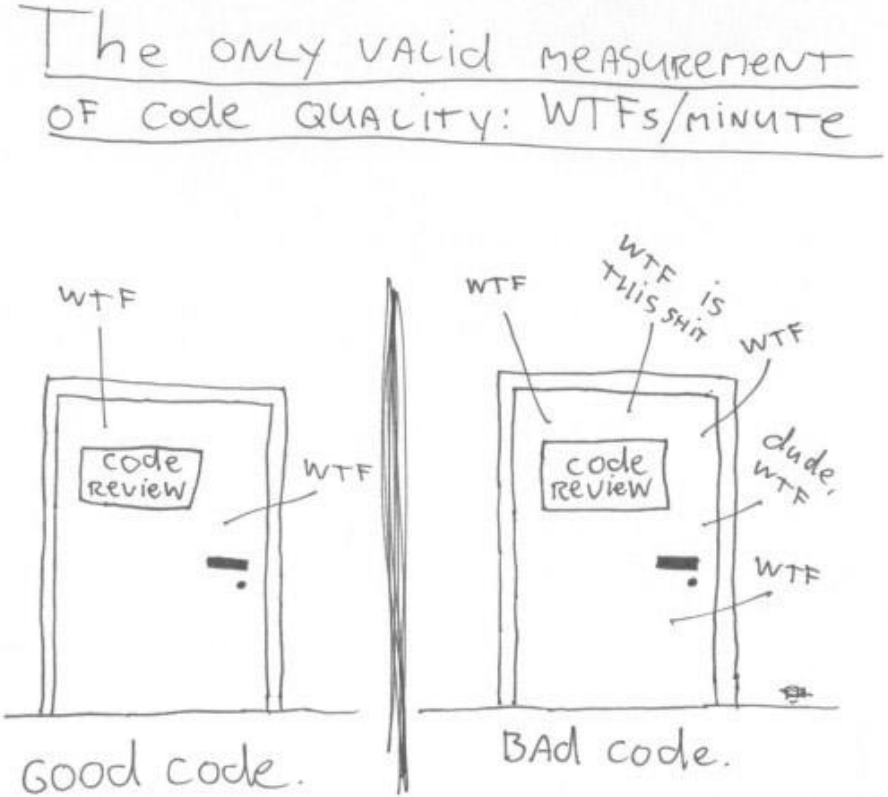
Measure Filter as Bubble Chart



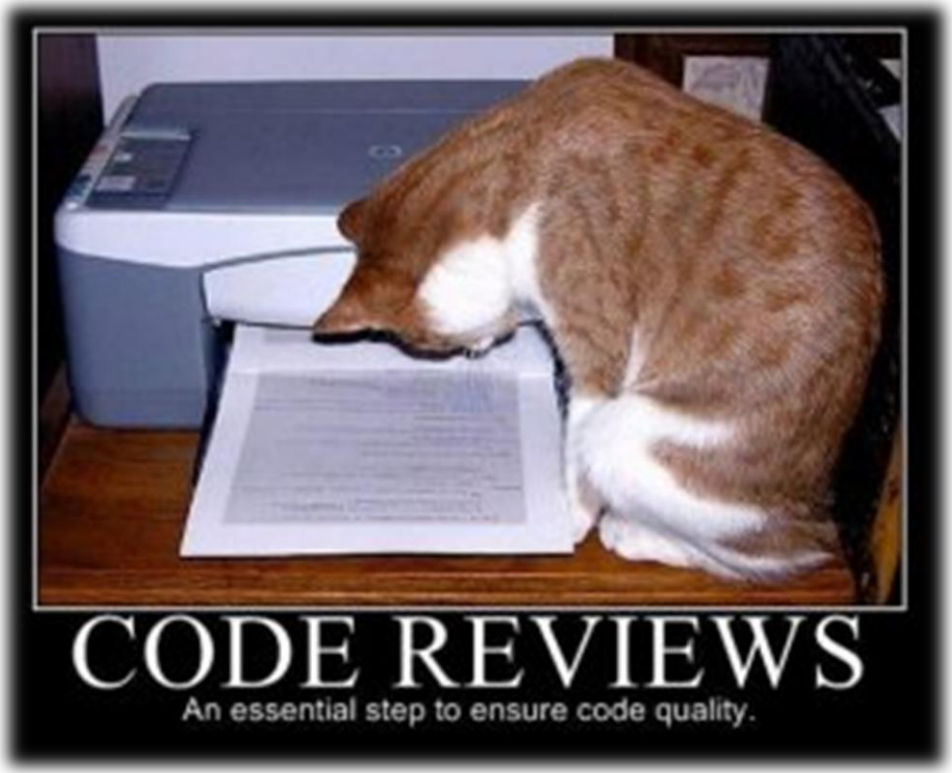
Filter Motion Chart



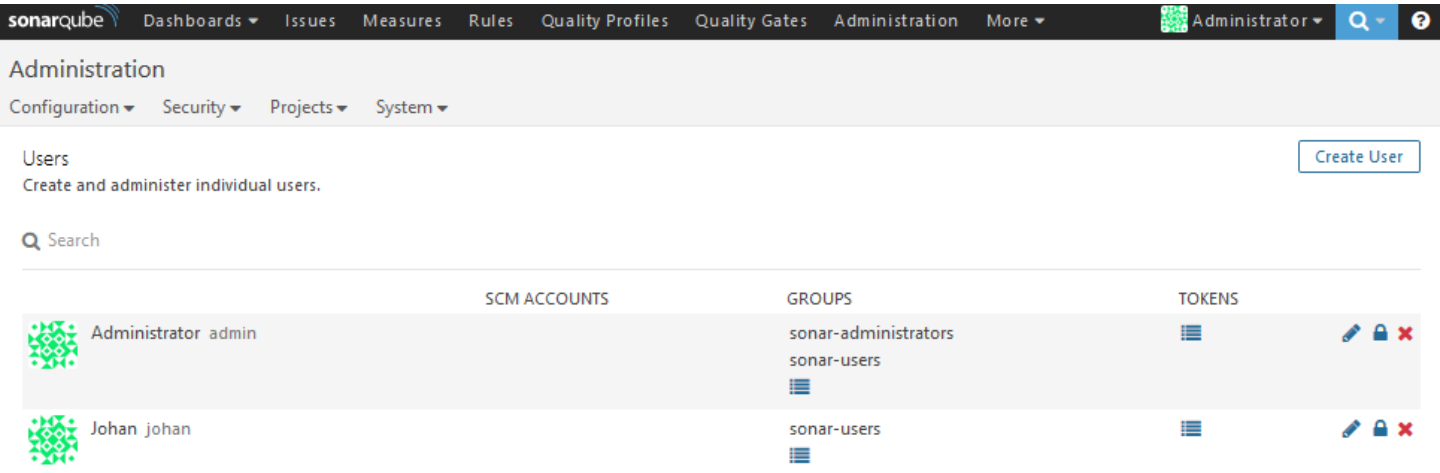
What is good code?

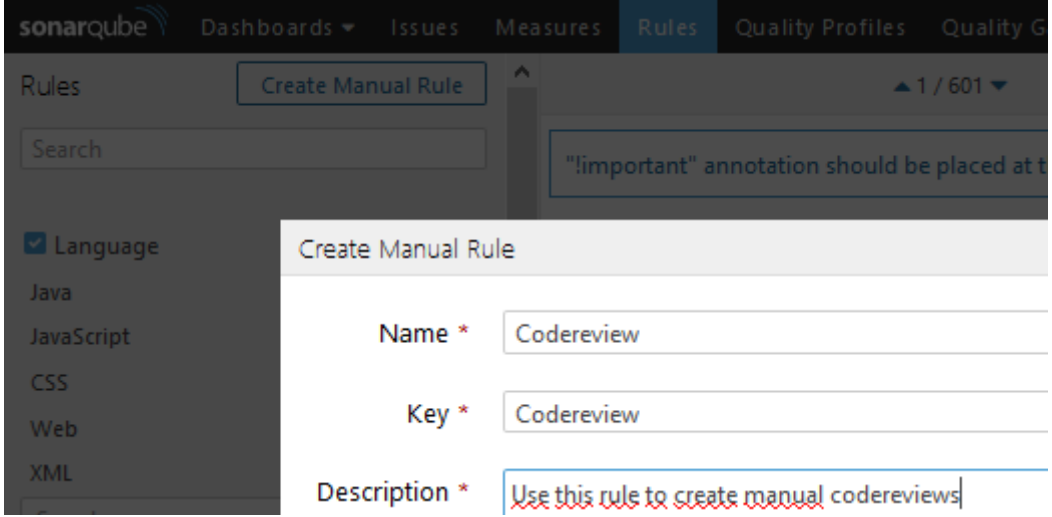


Code review

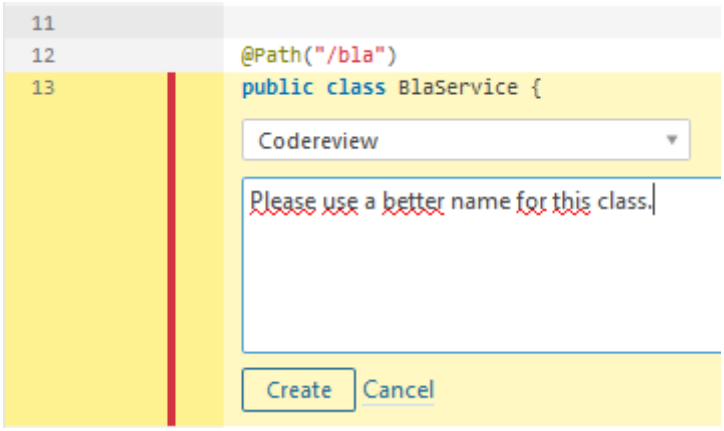


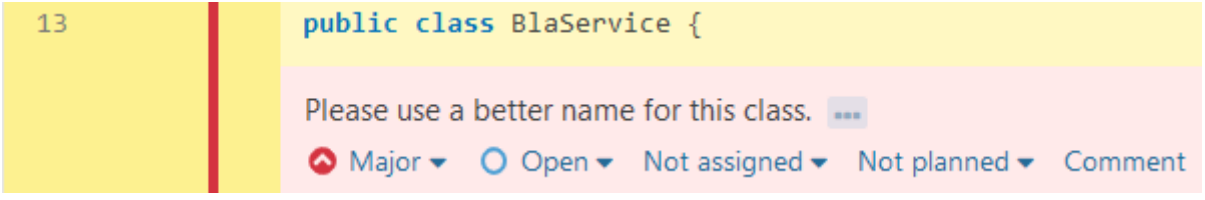
Roles


sonarqube
Dashboards
Issues
Measures
Rules
Quality Profiles
Quality Gates
Administration
More
Administrator
Administration
Configuration
Security
Projects
System
Users
Create and administer individual users.
Create User
Search
SCM ACCOUNTS
GROUPS
TOKENS
Administrator admin
sonar-administrators
sonar-users
Johan johan
sonar-users


sonarqube
Dashboards
Issues
Measures
Rules
Quality Profiles
Quality G
Rules
Create Manual Rule
1 / 601
Search
"!important" annotation should be placed at t
Language
Java
JavaScript
CSS
Web
XML
Create Manual Rule
Name *
Codereview
Key *
Codereview
Description *
Use this rule to create manual codereviews

12
13
14
15
16
17
18
19
20
Get Permalink
Add Manual Issue
@GET
@Path("/{param}")
public Response ge
logger.inf


11
12
@Path("/bla")
13
public class BlaService {
Codereview
Please use a better name for this class.
Create
Cancel


13
public class BlaService {
Please use a better name for this class.
Major
Open
Not assigned
Not planned
Comment

Why code reviews?

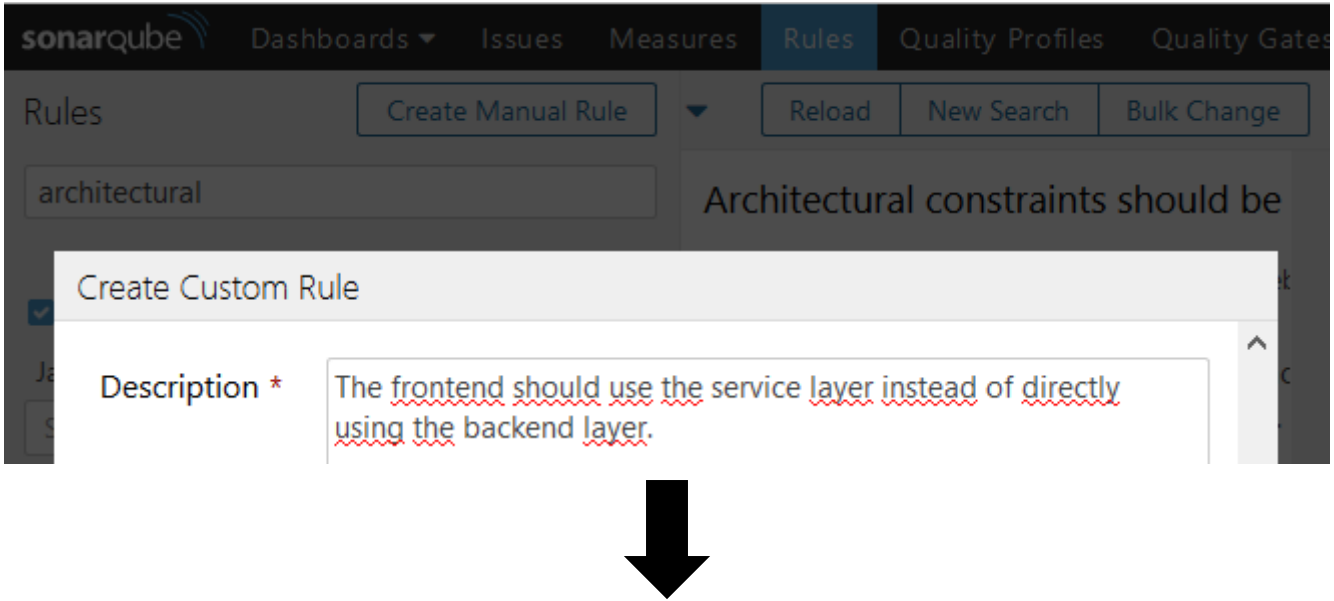
“We all need people who will give us feedback. That's how we improve.”

- Bill Gates

Hunting bad design and architecture

Hunting bad design and architecture

Architectural constraints



Custom Rules Create

NotFromFrontendToBackend Major fromClasses: com.dockerpi.sonar.layers.frontend.**
toClasses: com.dockerpi.sonar.layers.backend.**
Delete

Architectural constraints

```
package com.dockerpi.sonar.layers.frontend;

import com.dockerpi.sonar.layers.backend.PersonBackend;

public class DirectFromFrontendToBackend {

        private PersonBackend personBackend = new PersonBackend();
```

com.dockerpi.sonar.layers.frontend.DirectFromFrontendToBackend must not use com.dockerpi.sonar.layers.backend.PersonBackend ...

Major ▾ Open ▾ Not assigned ▾ Not planned ▾ Comment

Testing

Now manual execution of unit tests

```
mvn clean install
mvn sonar:sonar
```

```
mvn clean org.jacoco:jacoco-maven-
     plugin:prepare-agent install
     -Dmaven.test.failure.ignore=true
mvn sonar:sonar
```

Coverage per test

- Configure the POM (see next slide)
- Activate profile to get coverage per test information

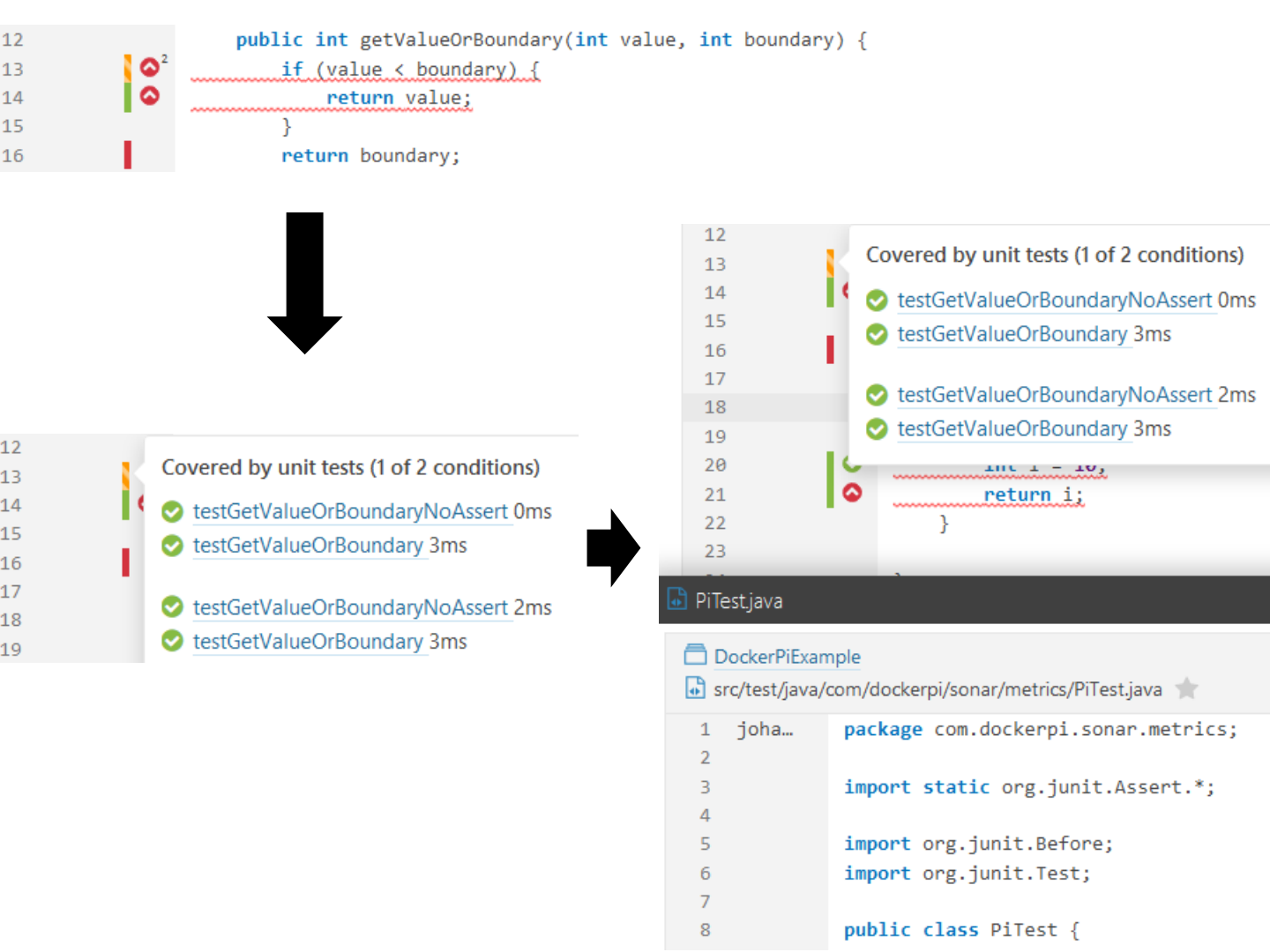
```
mvn org.jacoco:jacoco-maven-
    plugin:prepare-agent clean
    install -Pcoverage-per-test
```

- Analyze the project

```
mvn sonar:sonar
```

```xml
<profiles>
    <profile>
        <id>coverage-per-test</id>
        <build>
            <plugins>
                <plugin>
                    <groupId>org.apache.maven.plugins</groupId>
                    <artifactId>maven-surefire-plugin</artifactId>
                    <!-- Minimal supported version is 2.4 -->
                    <version>2.13</version>
                    <configuration>
                        <properties>
                            <property>
                                <name>listener</name>
                                <value>org.sonar.java.jacoco.JUnitListener</value>
                            </property>
                        </properties>
                    </configuration>
                </plugin>
            </plugins>
        </build>

        <dependencies>
            <dependency>
                <groupId>org.codehaus.sonar-plugins.java</groupId>
                <artifactId>sonar-jacoco-listeners</artifactId>
                <version>1.2</version>
                <scope>test</scope>
            </dependency>
        </dependencies>
    </profile>
</profiles>
```


12
13
14
15
16
public int getValueOrBoundary(int value, int boundary) {
if (value < boundary) {
return value;
}
return boundary;
Covered by unit tests (1 of 2 conditions)
testGetValueOrBoundaryNoAssert 0ms
testGetValueOrBoundary 3ms
testGetValueOrBoundaryNoAssert 2ms
testGetValueOrBoundary 3ms
Covered by unit tests (1 of 2 conditions)
testGetValueOrBoundaryNoAssert 0ms
testGetValueOrBoundary 3ms
testGetValueOrBoundaryNoAssert 2ms
testGetValueOrBoundary 3ms
return i;
PiTest.java
DockerPiExample
src/test/java/com/dockerpi/sonar/metrics/PiTest.java
joha…
package com.dockerpi.sonar.metrics;
import static org.junit.Assert.*;
import org.junit.Before;
import org.junit.Test;
public class PiTest {

Integration testing

Unit Tests Coverage

30.8% ↗

Line Coverage	Branch Coverage
34.3% ↗	0.0%

Unit Test Success

100.0%

Failures	Errors	Tests	Execution Time
0	0	3 ↗	3 ms ↗

Integration Tests Coverage

46.2% ↘

Line Coverage	Branch Coverage
45.7% ↘	50.0% ↗

Overall Coverage

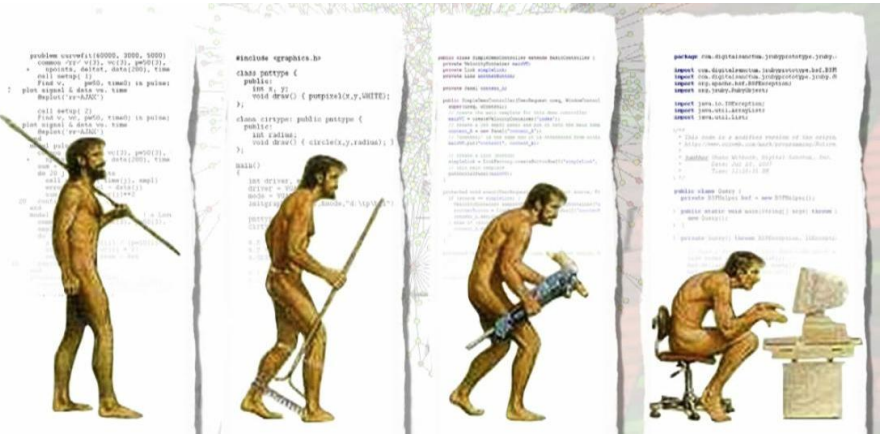
61.5%

Line Coverage	Branch Coverage
62.9%	50.0%

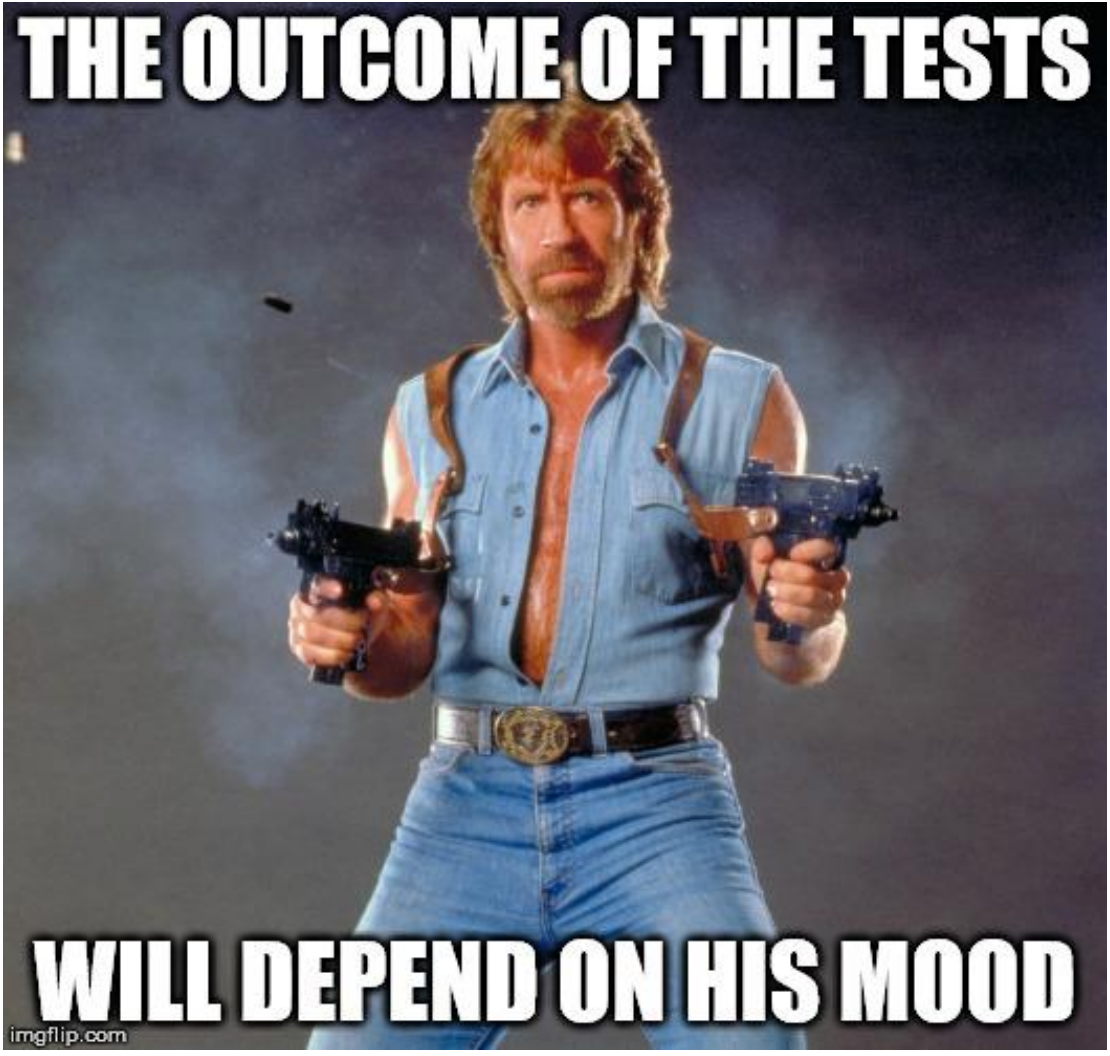
Integration testing

- ! Unit testing

Integration testing



Integration testing


THE OUTCOME OF THE TESTS
WILL DEPEND ON HIS MOOD
imgflip.com

Integration testing

- Add integration test coverage widget to SonarQube
- Download Jacoco agent
- Configuration for Tomcat's catalina.bat:

```
set JACOCO=-
javaagent:$path$\lib\jacocoagent.jar,
destfile=$resultpath$\jacoco.exec,
append=false,includes=com.dockerpi.*
set JAVA_OPTS=%JAVA_OPTS% %JACOCO%
```

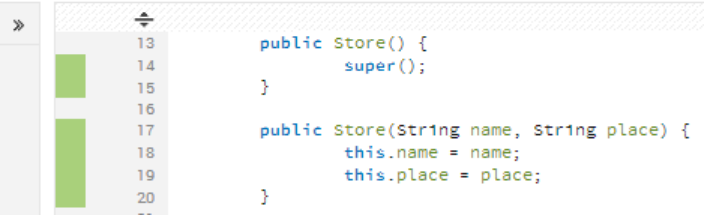
Integration testing

- Start Tomcat
- Execute tests (manual, integration, performance…)
- Stop Tomcat and execute SonarQube analysis

```
mvn clean install sonar:sonar
      -Dsonar.dynamicAnalysis=reuseReports
      -Dsonar.jacoco.itReportPath=
      $resultpath$\jacoco.exec
```

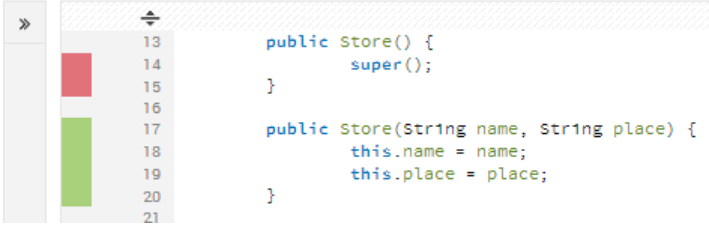
Unit and integration testing combined

Unit Tests		Integration Tests	
Coverage	83.3%	Coverage	50.0%
Line coverage	83.3%	Line coverage	50.0%
Lines to cover	12 ›	Lines to cover	12 ›
Uncovered lines	2 ›	Uncovered lines	6 ›



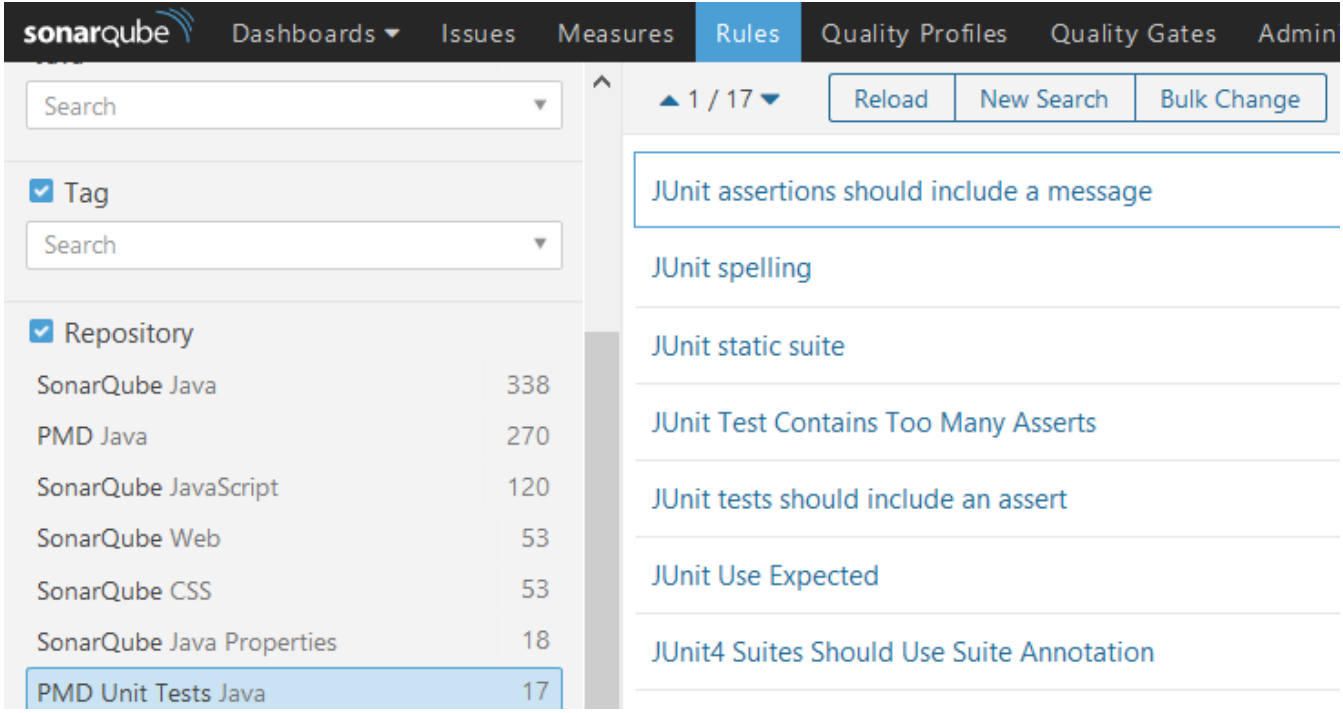
```
13  public Store() {
14          super();
15  }
16
17  public Store(String name, String place) {
18          this.name = name;
19          this.place = place;
20  }
```

Unit Tests		Integration Tests	
Coverage	83.3%	Coverage	50.0%
Line coverage	83.3%	Line coverage	50.0%
Lines to cover	12 ›	Lines to cover	12 ›
Uncovered lines	2 ›	Uncovered lines	6 ›



```
13  public Store() {
14          super();
15  }
16
17  public Store(String name, String place) {
18          this.name = name;
19          this.place = place;
20  }
21
```

PMD rules for unit testing



```
        assertTrue(a==b);
```

JUnit assertions should include a message ...

Minor ▾ Open ▾ Not assigned ▾ Not planned ▾ 10min debt Comment

Use assertSame(x, y) instead of assertTrue(x==y), or assertNotSame(x,y) vs assertFalse(x==y) ...

Minor ▾ Open ▾ Not assigned ▾ Not planned ▾ 10min debt Comment

```
        assertEquals("", true, a);
```

Use assertTrue(x)/assertFalse(x) instead of assertEquals(true, x)/assertEquals(false, x) or assertEquals(Boolean.TRUE, x)/assertEquals(Boolean.FALSE, x). ...

Minor ▾ Open ▾ Not assigned ▾ Not planned ▾ 10min debt Comment

```
        assertTrue("", a.equals(b));
        assertTrue("", c==null);
```

Use assertNull(x) instead of assertTrue(x==null), or assertNotNull(x) vs assertFalse(x==null) ...

Minor ▾ Open ▾ Not assigned ▾ Not planned ▾ 10min debt Comment

Pitest plugin

```
public int getValueOrBoundary(int value, int boundary) {
    if (value < boundary) {
        return value;
    }
    return boundary;
}
```

```
@Test
public void testGetValueOrBoundary() {
    assertTrue(pi.getValueOrBoundary(20, 100) < 101);
}

@Test
public void testGetValueOrBoundaryNoAssert() {
    pi.getValueOrBoundary(40, 200);
}
```

Pitest plugin

- Enable ‘Survived mutant’ rule in SonarQube Quality Profile
- Configure Pitest for instance with Maven (see next slides)

Pitest plugin

- Execute tests and send results to SonarQube

```
mvn org.pitest:pitest-
     maven:mutationCoverage
mvn sonar:sonar
     -Dsonar.pitest.mode=reuseReport
```

Pitest plugin

- Maven configuration inside the build/plugins section

```
<plugin>
  <groupId>org.pitest</groupId>
  <artifactId>pitest-maven</artifactId>
  <version>LATEST</version>
  <configuration>
    <inScopeClasses>
      <param>com.example*</param>
    </inScopeClasses>
    <targetClasses>
      <param>com.example*</param>
    </targetClasses>
    <outputFormats>
      <outputFormat>XML</outputFormat>
    </outputFormats>
  </configuration>
```

Pitest plugin

```
    public int getValueOrBoundary(int value, int boundary) {
        if (value < boundary) {
```

A relational operator has been replaced by a boundary counterpart without breaking the tests

Major ▾ Open ▾ Not assigned ▾ Not planned ▾ Comment

A conditional expression has been negated without breaking the tests

Major ▾ Open ▾ Not assigned ▾ Not planned ▾ Comment

```
            return value;
```

The return value of a method call has been replaced without breaking the tests

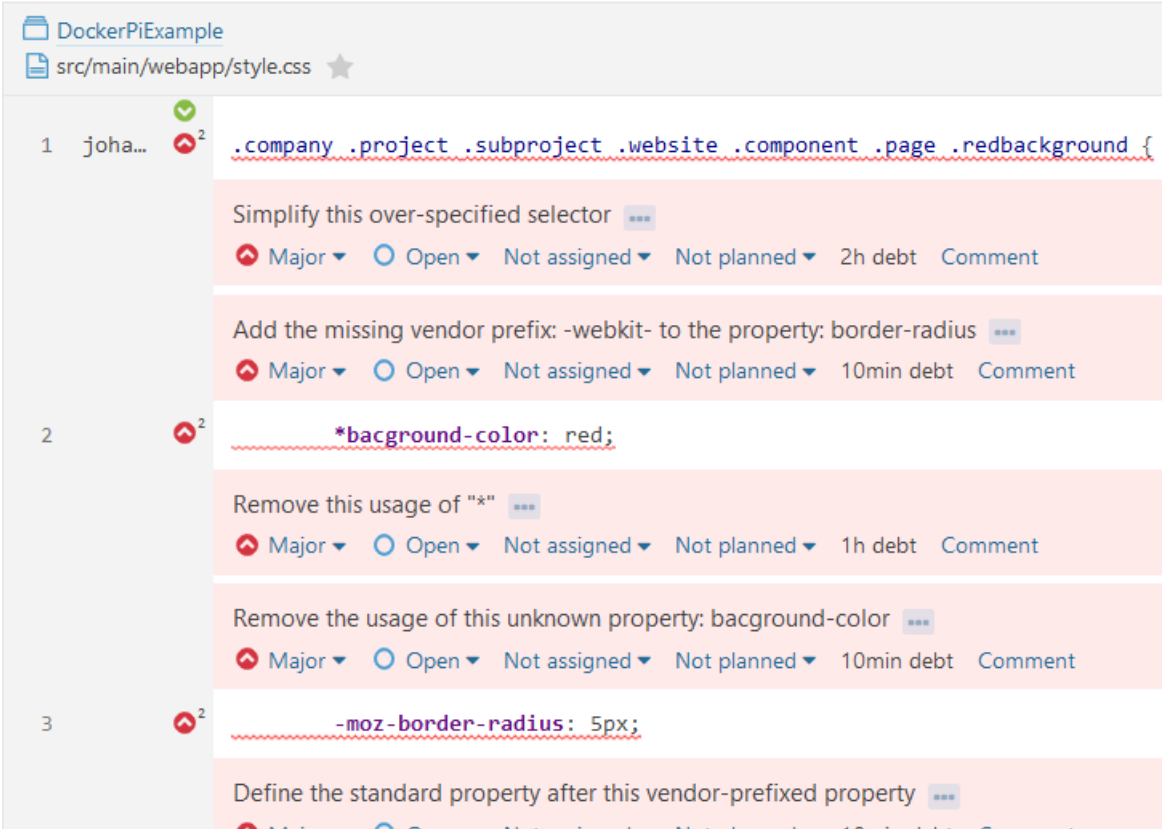
Major ▾ Open ▾ Not assigned ▾ Not planned ▾ Comment

Other languages

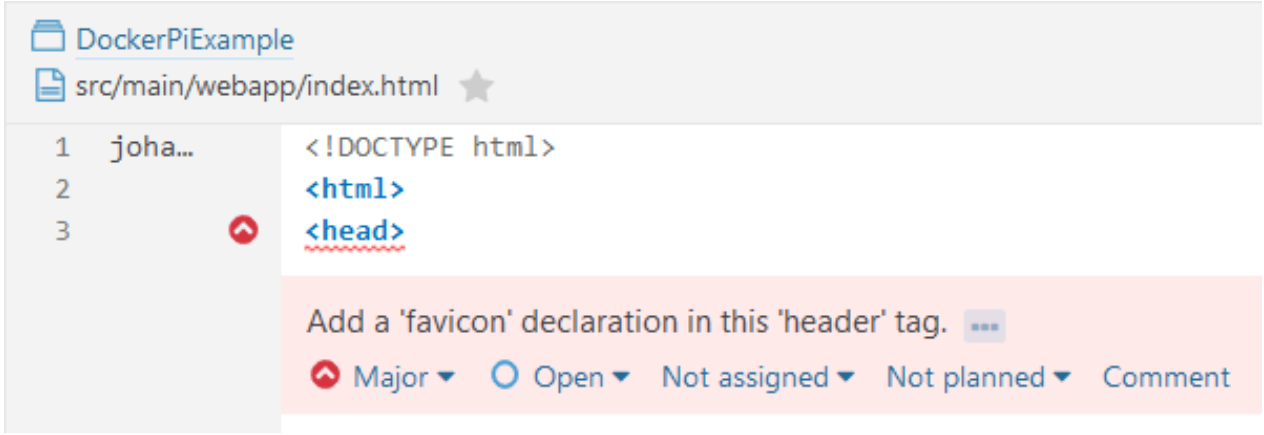
JavaScript

DockerPiExample
src/main/webapp/sonarviolations.js
1 joha… function nesting(temp) {
2 if(temp != "") {
Replace "!=" with "!==". ...
Major Open Not assigned Not planned 5min debt Comment
3 alert("No var supplied");
Remove this usage of alert(...). ...
Major Open Not assigned Not planned 10min debt Comment
4 }
5 if (temp < 100) {
Merge this if statement with the nested one. ...
Major Open Not assigned Not planned 5min debt Comment
6 if (temp < 75) {
7 // TODO do something
Complete the task associated to this TODO comment. ...

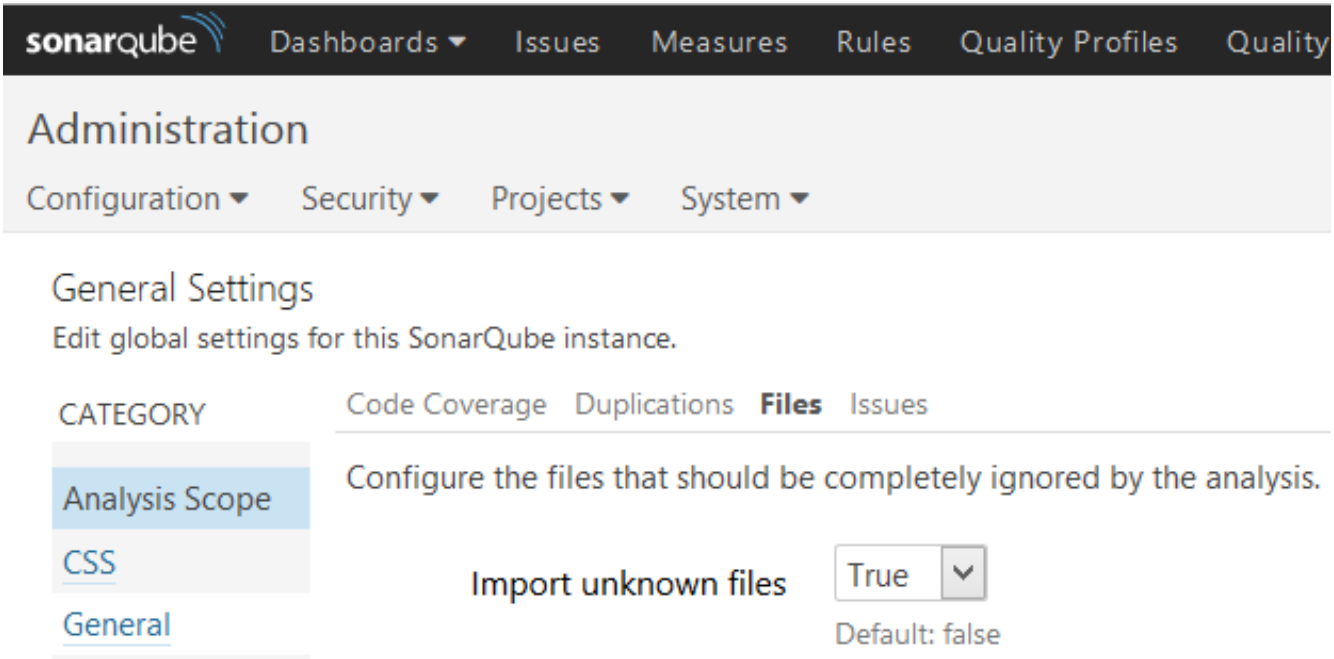
CSS


DockerPiExample
src/main/webapp/style.css
1 joha…
.company .project .subproject .website .component .page .redbackground {
Simplify this over-specified selector
Major Open Not assigned Not planned 2h debt Comment
Add the missing vendor prefix: -webkit- to the property: border-radius
Major Open Not assigned Not planned 10min debt Comment
2
*bacground-color: red;
Remove this usage of "*"
Major Open Not assigned Not planned 1h debt Comment
Remove the usage of this unknown property: bacground-color
Major Open Not assigned Not planned 10min debt Comment
3
-moz-border-radius: 5px;
Define the standard property after this vendor-prefixed property

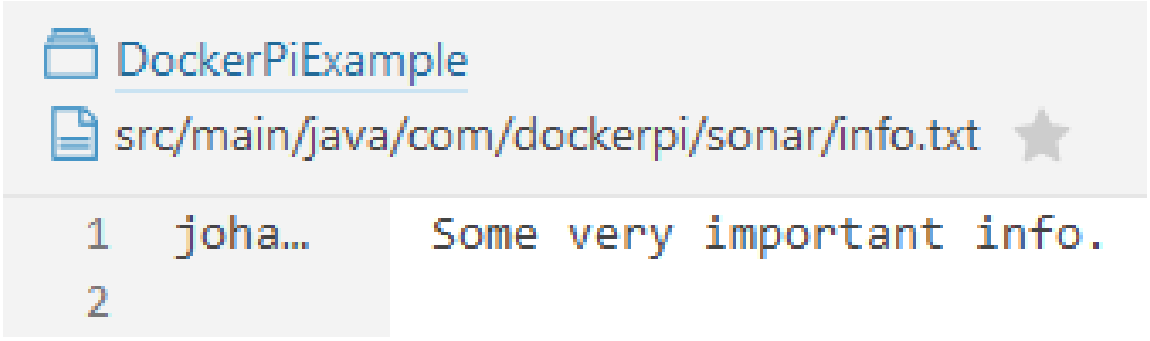
Web plugin


DockerPiExample
src/main/webapp/index.html
1 joha…
2
3
<!DOCTYPE html>
<html>
<head>
Add a 'favicon' declaration in this 'header' tag.
Major
Open
Not assigned
Not planned
Comment

Import unknown files



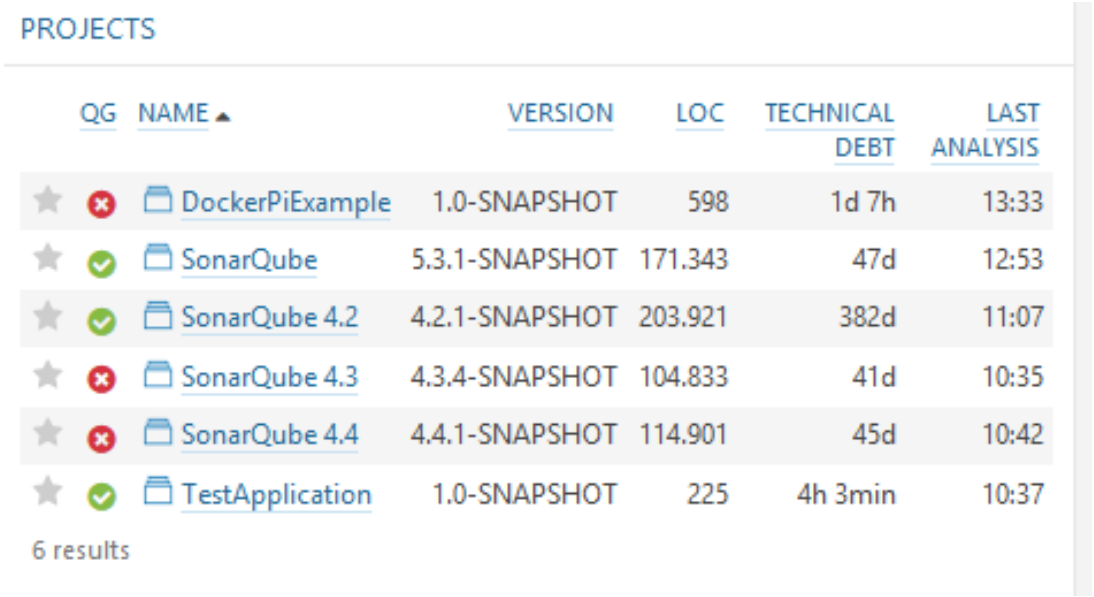
Import unknown files



Leftovers

Branches

- `mvn sonar:sonar -Dsonar.branch=4.4`



PROJECTS

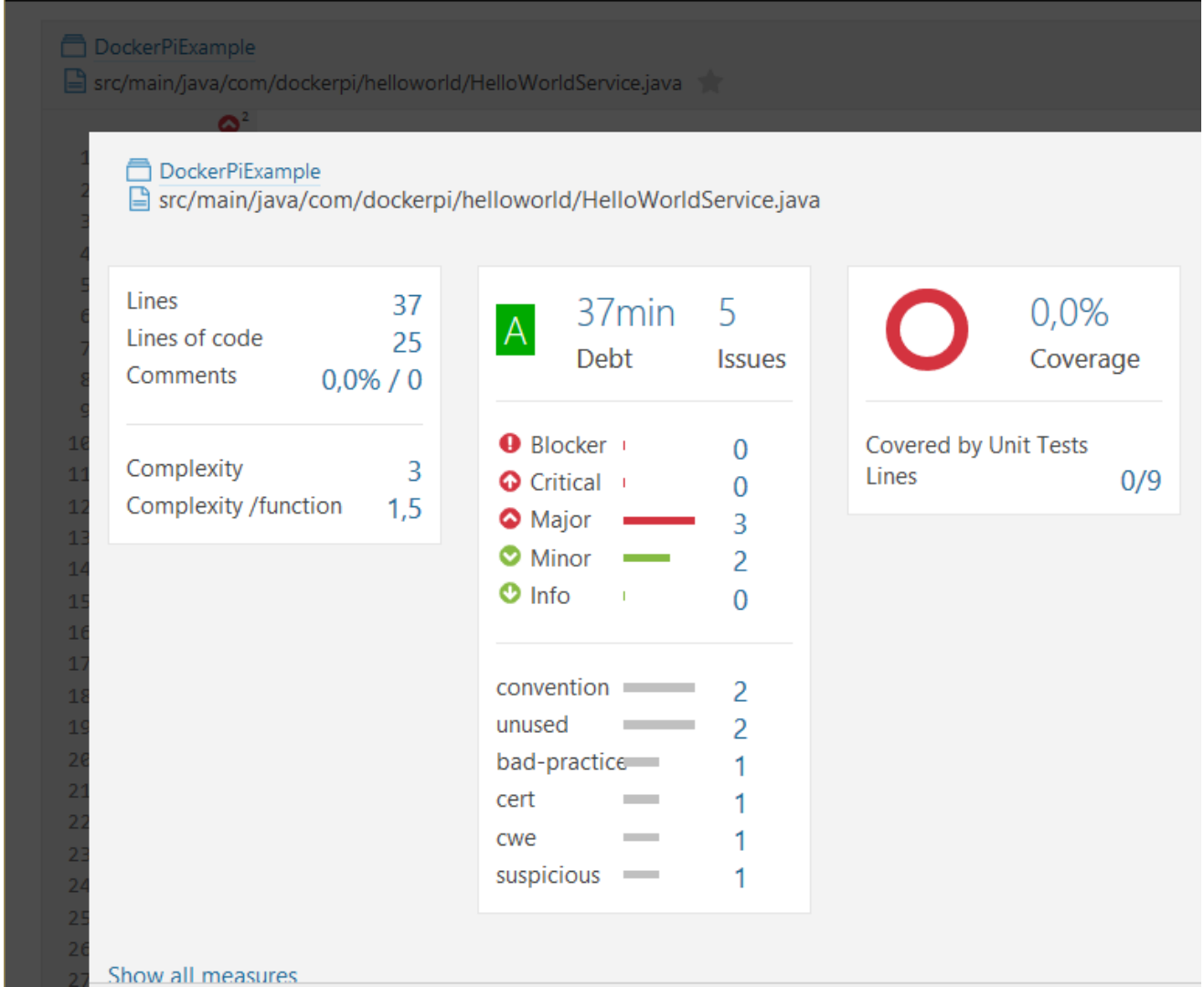
QG	NAME	VERSION	LOC	TECHNICAL DEBT	LAST ANALYSIS
✗	DockerPiExample	1.0-SNAPSHOT	598	1d 7h	13:33
✓	SonarQube	5.3.1-SNAPSHOT	171.343	47d	12:53
✓	SonarQube 4.2	4.2.1-SNAPSHOT	203.921	382d	11:07
✗	SonarQube 4.3	4.3.4-SNAPSHOT	104.833	41d	10:35
✗	SonarQube 4.4	4.4.1-SNAPSHOT	114.901	45d	10:42
✓	TestApplication	1.0-SNAPSHOT	225	4h 3min	10:37

6 results

Exclusions

- Ignore files
- Ignore issues
- Ignore duplications
- Ignore code coverage
- Example:

```
mvn sonar:sonar -Dsonar.exclusions=
     src/main/webapp/lib/*.js
```


DockerPiExample
src/main/java/com/dockerpi/helloworld/HelloWorldService.java
DockerPiExample
src/main/java/com/dockerpi/helloworld/HelloWorldService.java
Lines 37
Lines of code 25
Comments 0,0% / 0
Complexity 3
Complexity /function 1,5
A
37min
Debt
5
Issues
Blocker 0
Critical 0
Major 3
Minor 2
Info 0
convention 2
unused 2
bad-practice 1
cert 1
cwe 1
suspicious 1
0,0%
Coverage
Covered by Unit Tests
Lines 0/9
Show all measures

Complexity

Complexity	3
Complexity /class	3,0
Complexity /file	3,0
Complexity /function	1,5

Issues

Issues	5
Major issues	3
Minor issues	2
Open issues	5

Size

Classes	1
Files	1
Functions	2
Lines	37
Lines of code	25
Statements	7

Documentation

Comment lines	0
Comments (%)	0,0%
Public API	3
Public documented API (%)	0,0%
Public undocumented API	3

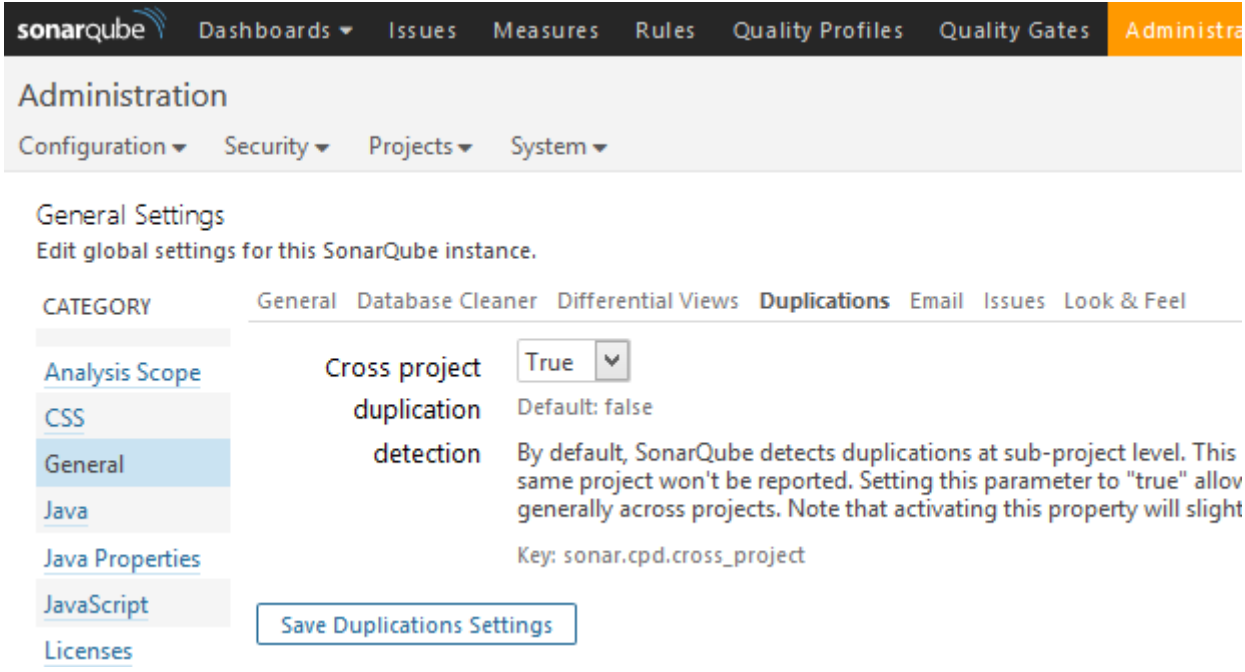
Mutation analysis

Detected Mutations	0
Killed Mutations	0
Memory Error Mutations	0
Non Covered Mutations	5
Survived Mutations	0
Timed Out Mutations	0
Total Mutations	5
Unknown Status Mutations	0

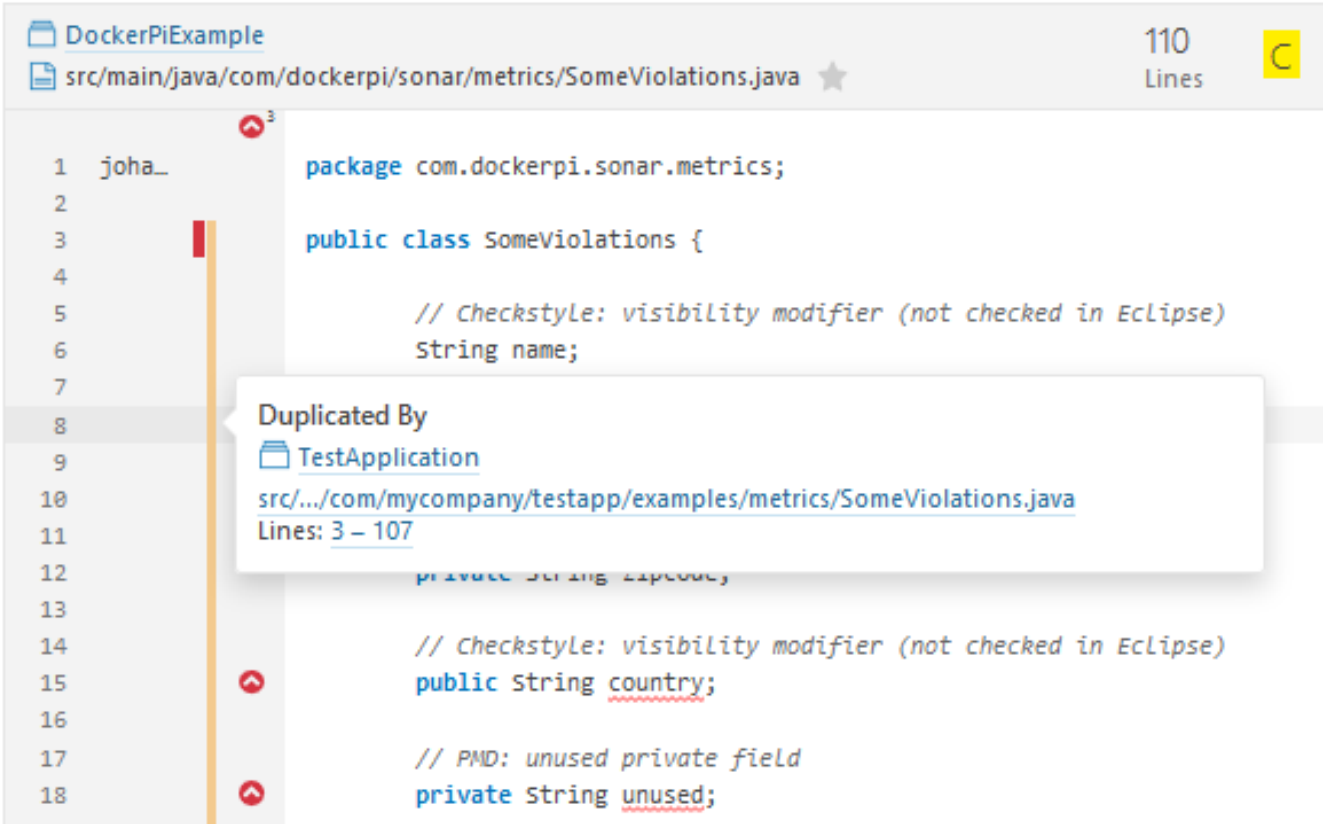
Technical Debt

SQALE Rating	A
Technical Debt	37min

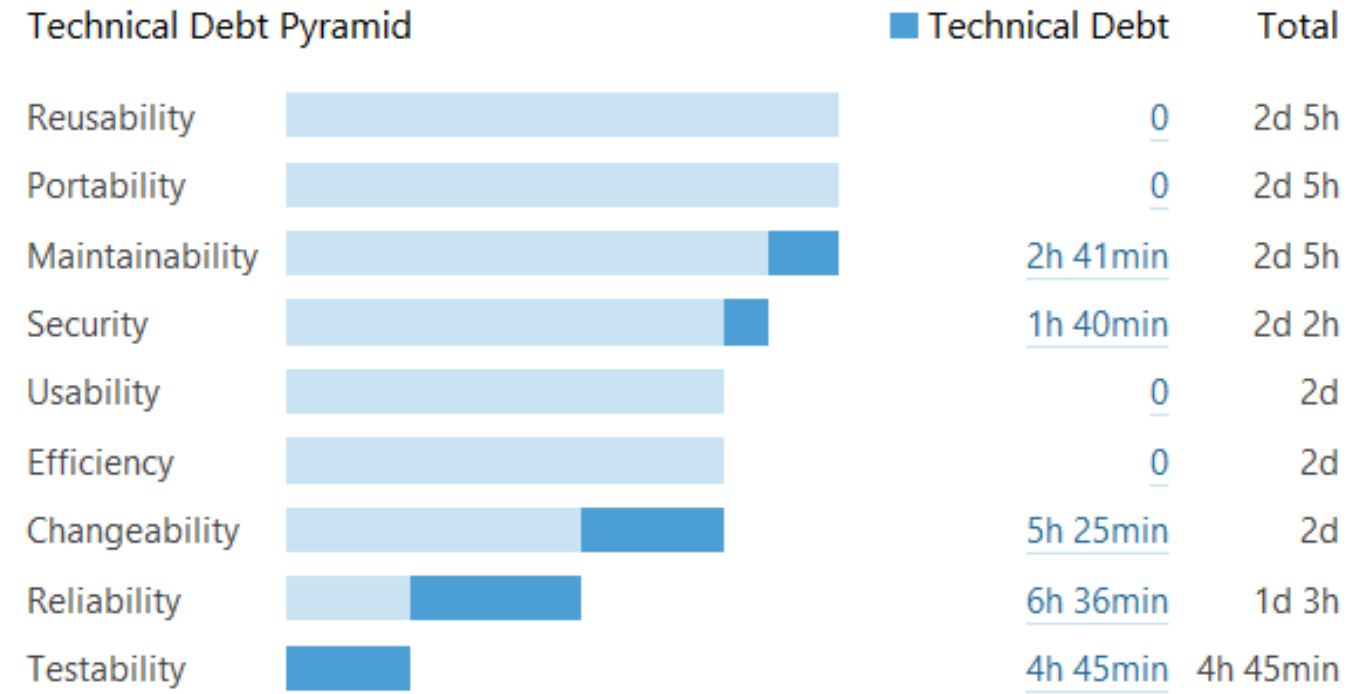
Duplications across projects



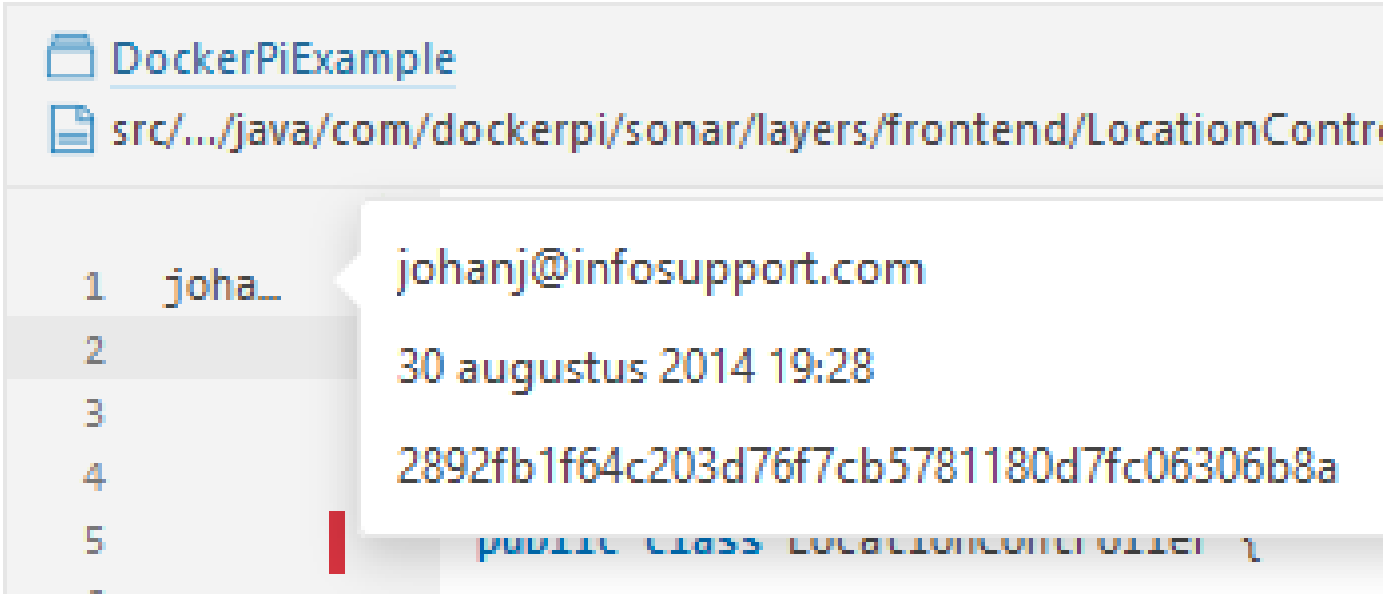
Duplications across projects



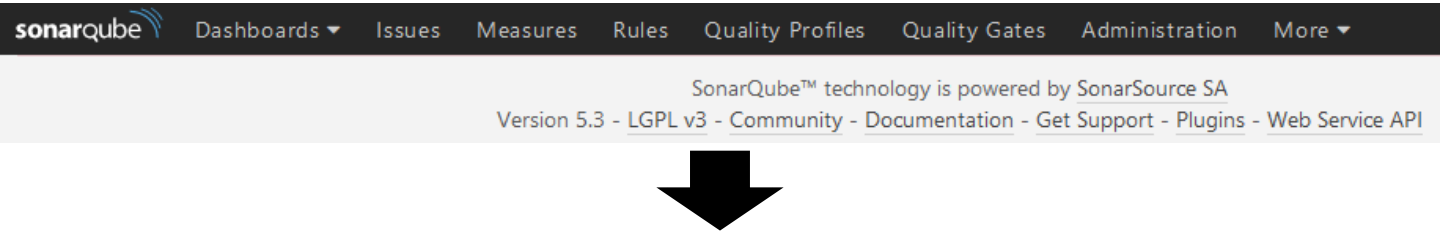
Technical debt pyramid

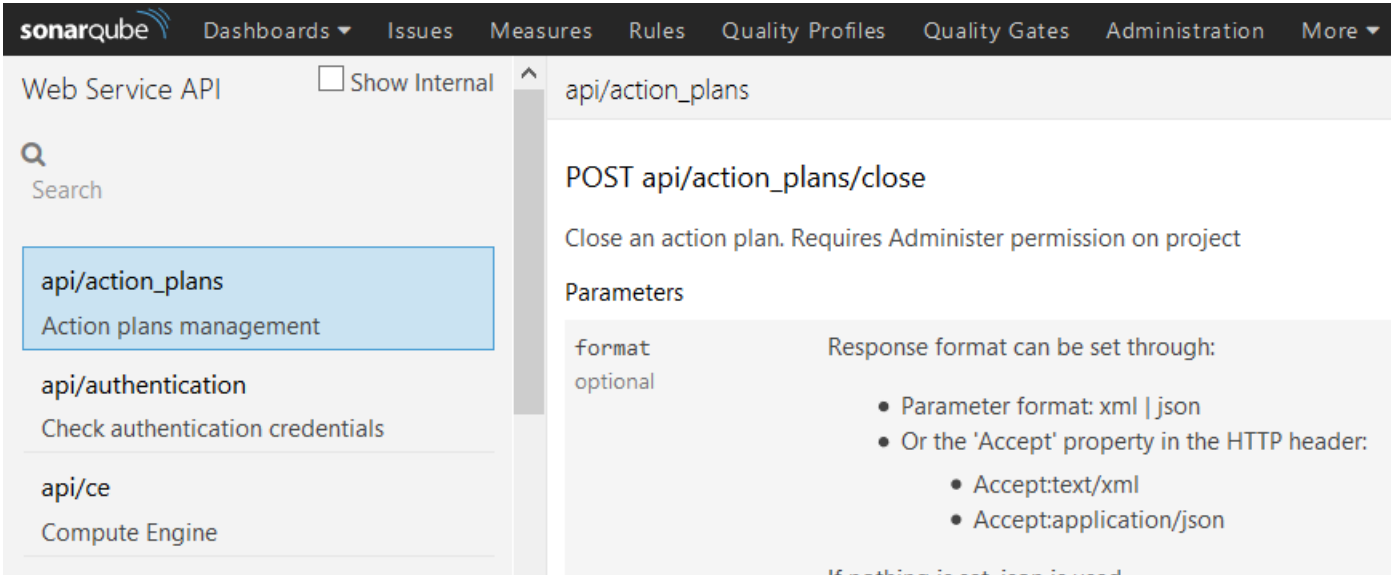


SCM Activity

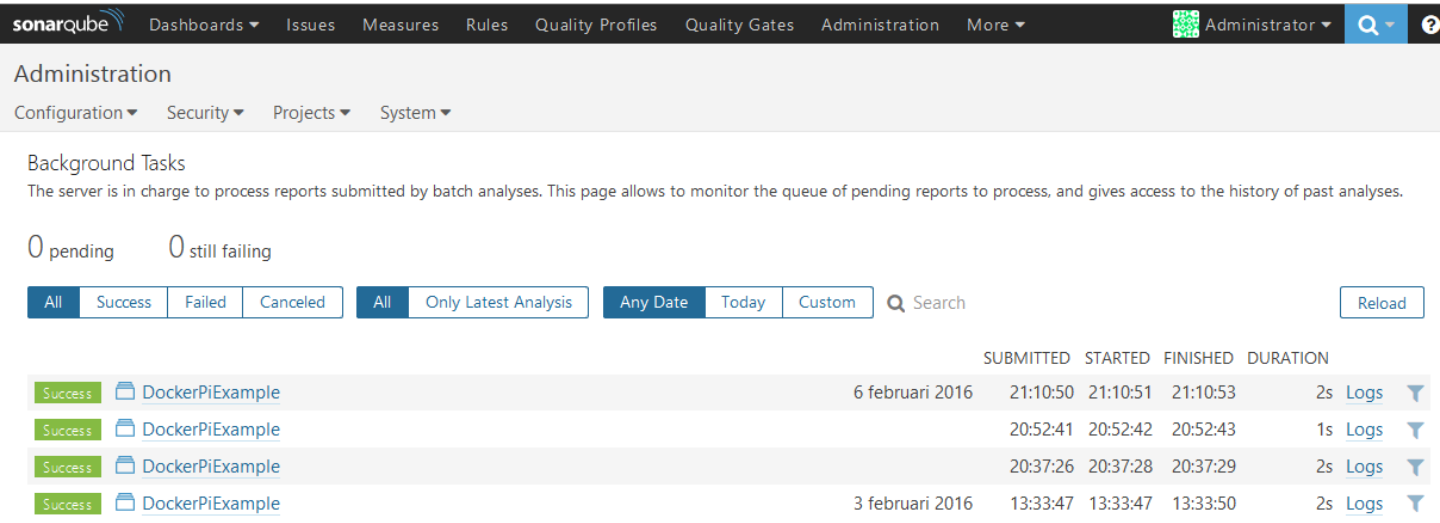


Web Service API

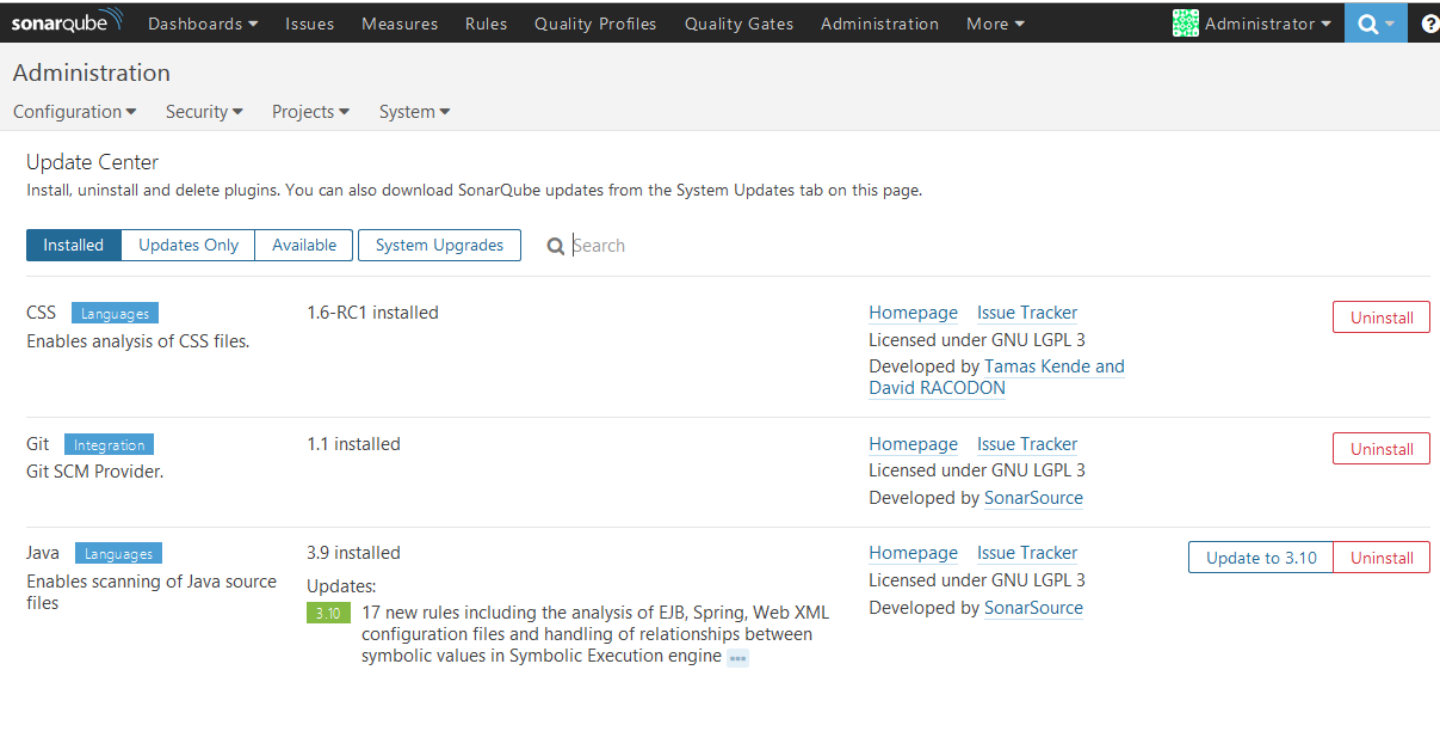




Background Tasks

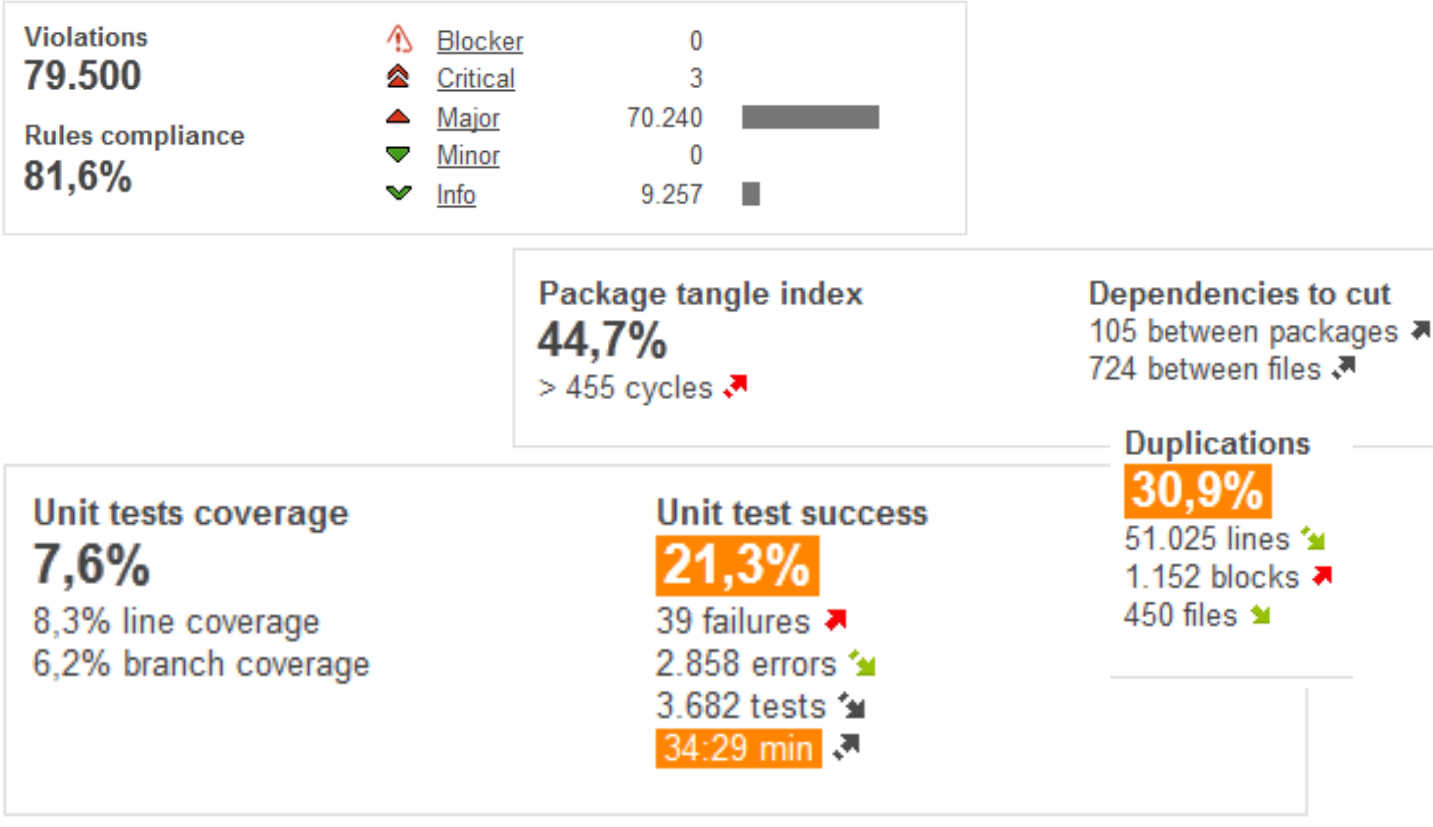


Update center



Using SonarQube on existing projects

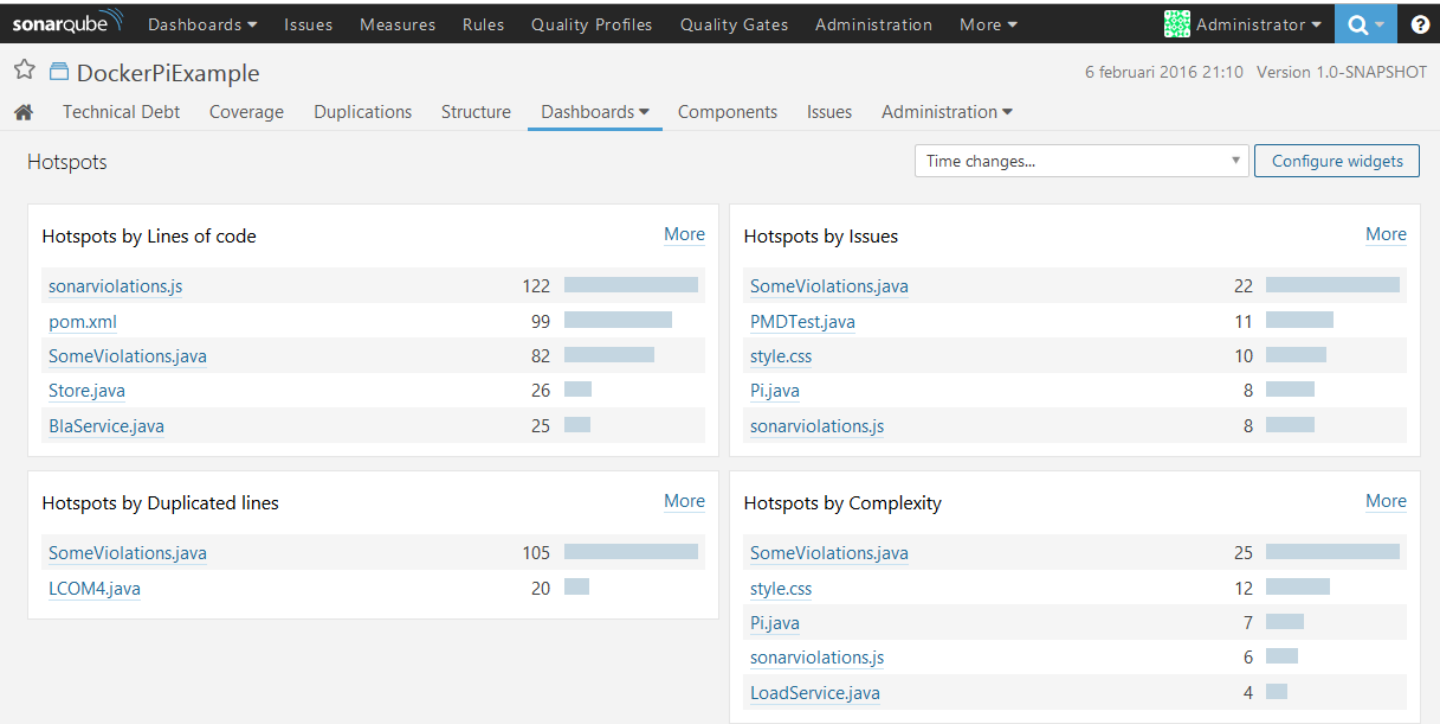
Using SonarQube on existing projects



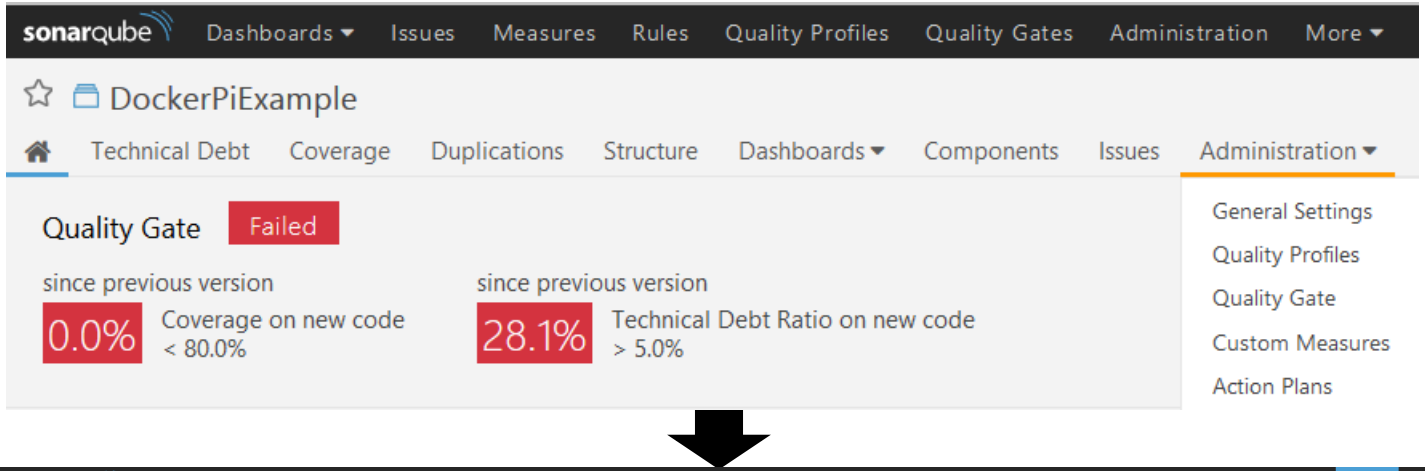
Using SonarQube on existing projects

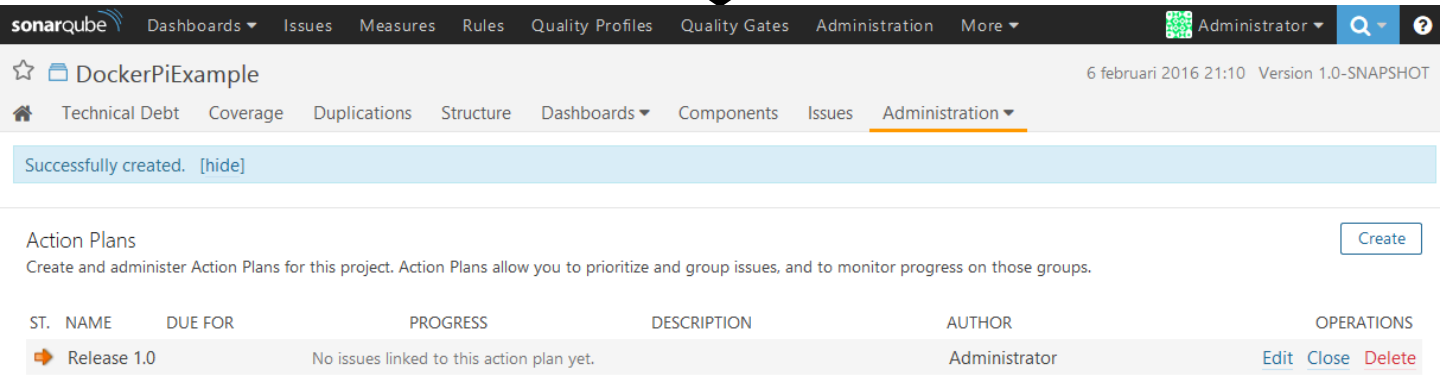
- Identity hotspots
- Use action plans
- Maybe use a less strict quality profile
- Add tasks/stories in the sprint to improve quality
- Monitor quality of new code

Hotspots

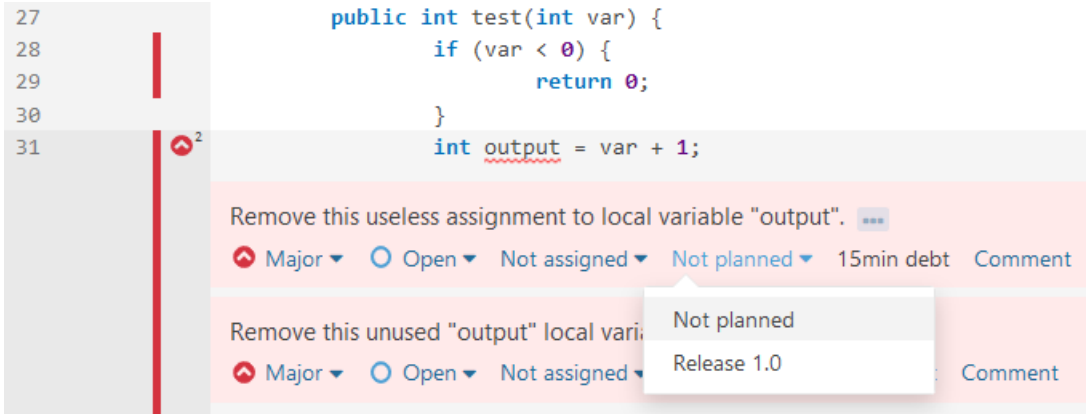

sonarqube
Dashboards
Issues
Measures
Rules
Quality Profiles
Quality Gates
Administration
More
Administrator
DockerPiExample
6 februari 2016 21:10 Version 1.0-SNAPSHOT
Technical Debt
Coverage
Duplications
Structure
Dashboards
Components
Issues
Administration
Hotspots
Time changes...
Configure widgets
Hotspots by Lines of code
More
sonarviolations.js 122
pom.xml 99
SomeViolations.java 82
Store.java 26
BlaService.java 25
Hotspots by Issues
More
SomeViolations.java 22
PMDTest.java 11
style.css 10
Pi.java 8
sonarviolations.js 8
Hotspots by Duplicated lines
More
SomeViolations.java 105
LCOM4.java 20
Hotspots by Complexity
More
SomeViolations.java 25
style.css 12
Pi.java 7
sonarviolations.js 6
LoadService.java 4

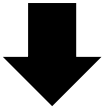
Planning issues

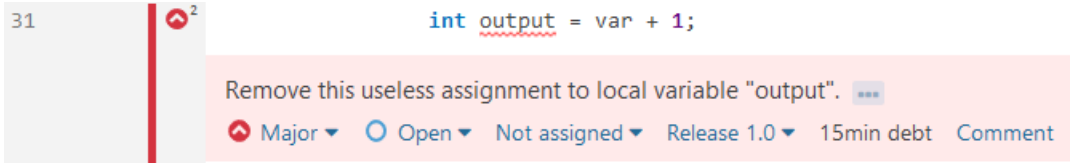




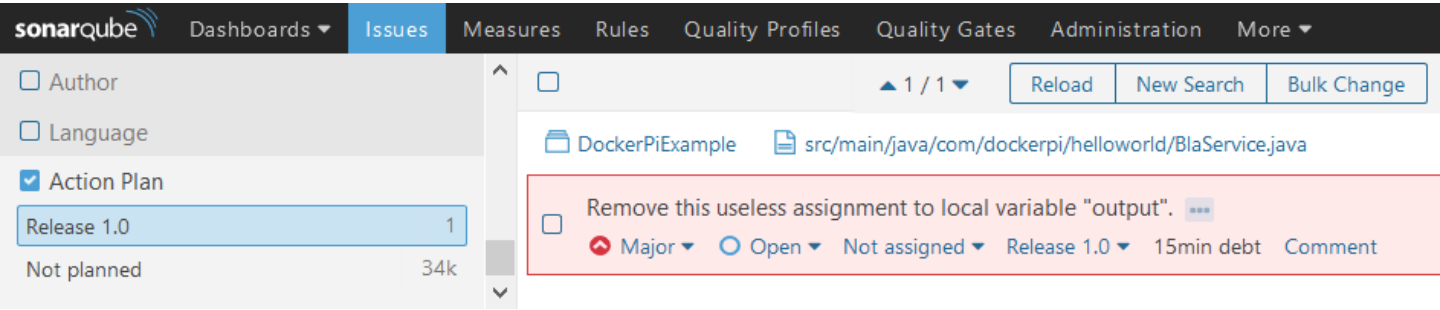
Planning issues







Issues


sonarqube
Dashboards
Issues
Measures
Rules
Quality Profiles
Quality Gates
Administration
More
Author
Language
Action Plan
Release 1.0 1
Not planned 34k
1 / 1
Reload
New Search
Bulk Change
DockerPiExample
src/main/java/com/dockerpi/helloworld/BlaService.java
Remove this useless assignment to local variable "output".
Major
Open
Not assigned
Release 1.0
15min debt
Comment

Tips for using SonarQube in a project

- Create stories/tasks to improve the quality
- Perform manual code reviews
- Agree on a standard for items SonarQube cannot check
- Fix items or mark them as false positive
- Use separate logins
- Settings/reviews are project specific

Tips for using SonarQube in a project

- Options to use SonarQube
  - Central server (commit AND nightly)
  - IDE plugin
  - Local in your development environment
- Check code quality before ‘To verify’ step
  - First merge your code so it is up to date
- Verifier should also verify the quality

25

Accept failure, but improve continuously

“Once we accept our limits, we go beyond them.”

- Albert Einstein

Questions

- Ask me!
- Sonarqube.org
- Nemo example dashboard
- SonarQube books

johan.janssen@infosupport.com
@johanjanssen42