



Computer Engineering Department
Software Engineering

Academic Year: 2020/2021	20/11/2020
Semester: First Semester	Dr. Haya Samaneh

Students	
1- Abdelrahman Baba	2- Maysem Mousa

1- Mocking:

We updated our project to allow it to send Emails after each search operation to the user's Email, then we updated our test cases to test this new feature.

We used the mocking technique to verify that everything is done successfully at our end without calling the actual 3th-party service, by verifying the number of time the service is expected to be called and testing sample expected returns.

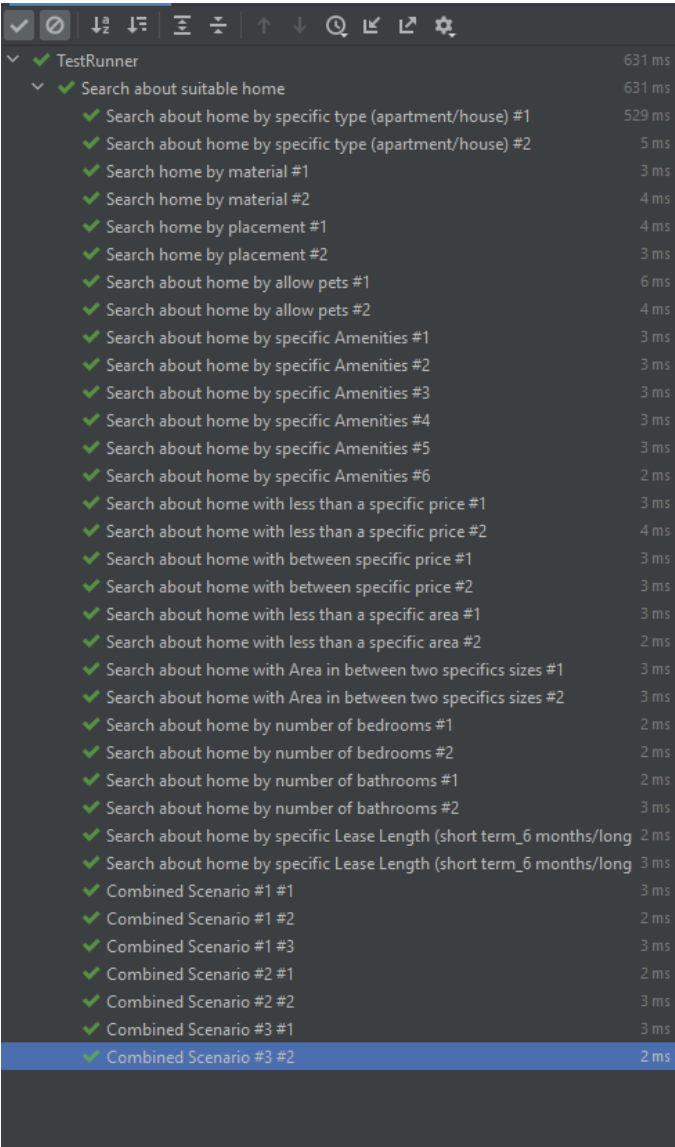
Sample code of our .feature file:

```
35 > Scenario Outline: Search about home by specific type (apartment/house)
36   When I search about home by type "<type>" specification
37   Then A list of homes that matches specifications "<result>" should be returned and printed on the console
38   And An Email with a list of search results will be sent to "maysam.m.mousa@gmail.com" customer email
39   Examples:
40     | type | result |
41     | House | 01 |
42     | Apartment | 02 |
43
44 > Scenario Outline: Search home by material
45   When I search about home by material "<material>"
46   Then A list of homes that matches specifications "<result>" should be returned and printed on the console
47   And An Email with a list of search results will be sent to "maysam.m.mousa@gmail.com" customer email
48
49   Examples:
50     | material | result |
51     | Wood | 01 |
52     | Brick | 02 |
53
54 > Scenario Outline: Search home by placement
55   When I search about home by placement as "<place>"
56   Then A list of homes that matches specifications "<result>" should be returned and printed on the console
57   And An Email with a list of search results will be sent to "maysam.m.mousa@gmail.com" customer email
58
59   Examples:
60     | place | result |
61     | Village | 01 |
62     | City | 02 |
63
64
65 > Scenario Outline: Search about home by allow pets
66   When I search about home by pets "<pets>" specification
67   Then A list of homes that matches specifications "<result>" should be returned and printed on the console
68   And An Email with a list of search results will be sent to "maysam.m.mousa@gmail.com" customer email
69
```

Step implementation:

```
@And("An Email with a list of search results will be sent to {string} customer email")
public void anEmailWithAListOfSearchResultsWillBeSentToCustomerEmail(String email) {
    EmailService mockedEmail = mock(EmailService.class);
    mockedEmail.sendMail(email,searchResults);
    verify(mockedEmail,times(wantedNumberOfInvocations: 1)).sendMail(email,searchResults);
}
}
```

Junit tests Results:

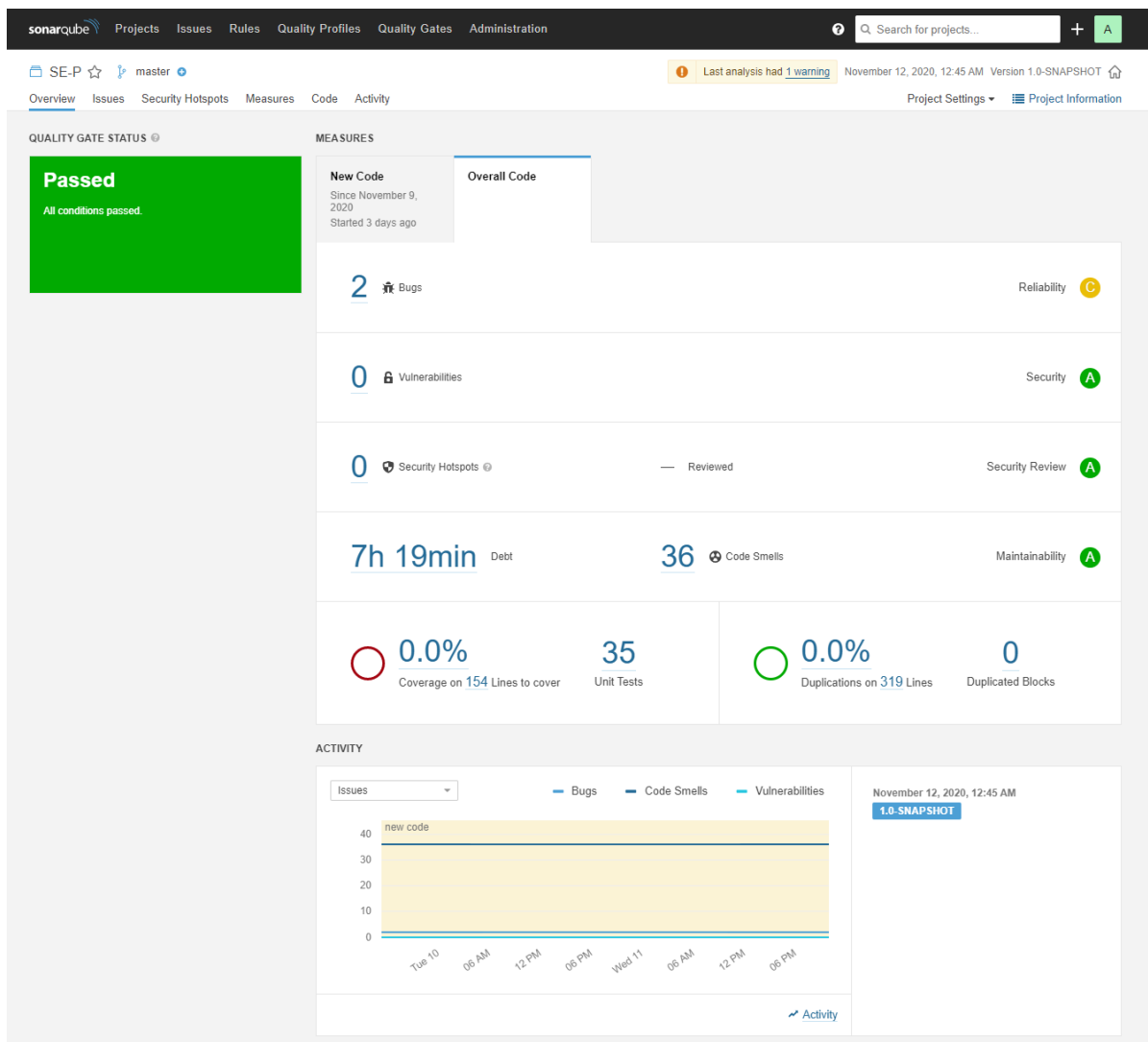


✓ TestRunner	631 ms
✓ Search about suitable home	631 ms
✓ Search about home by specific type (apartment/house) #1	529 ms
✓ Search about home by specific type (apartment/house) #2	5 ms
✓ Search home by material #1	3 ms
✓ Search home by material #2	4 ms
✓ Search home by placement #1	4 ms
✓ Search home by placement #2	3 ms
✓ Search about home by allow pets #1	6 ms
✓ Search about home by allow pets #2	4 ms
✓ Search about home by specific Amenities #1	3 ms
✓ Search about home by specific Amenities #2	3 ms
✓ Search about home by specific Amenities #3	3 ms
✓ Search about home by specific Amenities #4	3 ms
✓ Search about home by specific Amenities #5	3 ms
✓ Search about home by specific Amenities #6	2 ms
✓ Search about home with less than a specific price #1	3 ms
✓ Search about home with less than a specific price #2	4 ms
✓ Search about home with between specific price #1	3 ms
✓ Search about home with between specific price #2	3 ms
✓ Search about home with less than a specific area #1	3 ms
✓ Search about home with less than a specific area #2	2 ms
✓ Search about home with Area in between two specifics sizes #1	3 ms
✓ Search about home with Area in between two specifics sizes #2	3 ms
✓ Search about home by number of bedrooms #1	2 ms
✓ Search about home by number of bedrooms #2	2 ms
✓ Search about home by number of bathrooms #1	2 ms
✓ Search about home by number of bathrooms #2	3 ms
✓ Search about home by specific Lease Length (short term_6 months/long	2 ms
✓ Search about home by specific Lease Length (short term_6 months/long	3 ms
✓ Combined Scenario #1 #1	3 ms
✓ Combined Scenario #1 #2	2 ms
✓ Combined Scenario #1 #3	3 ms
✓ Combined Scenario #2 #1	2 ms
✓ Combined Scenario #2 #2	3 ms
✓ Combined Scenario #3 #1	3 ms
✓ Combined Scenario #3 #2	2 ms

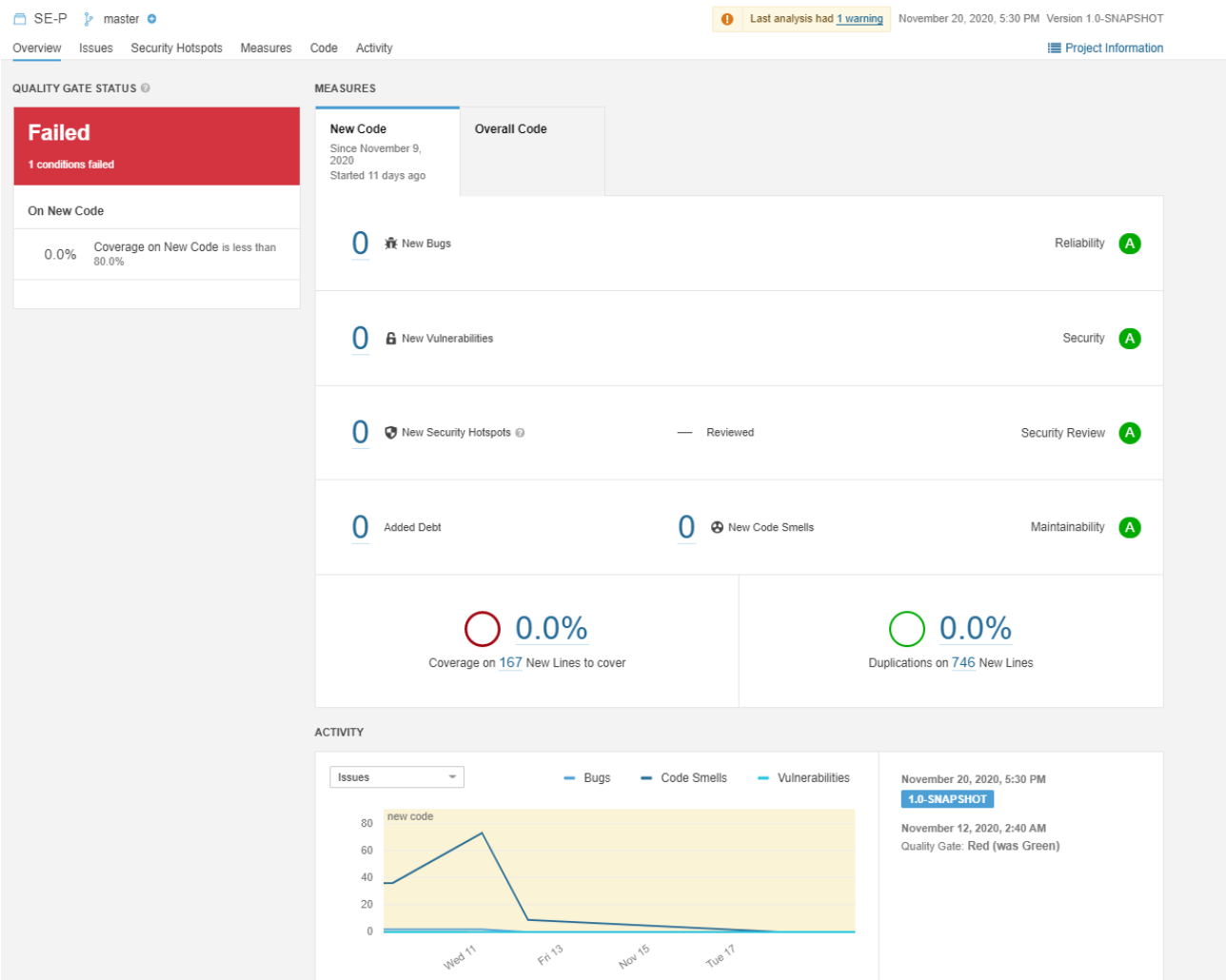
2- Refactoring :

We refactored our code using [composition pattern](#). As known in composition patterns, on class that implements the interface must have a reference to the interface except the Composite class, that was implemented in our code and it perfectly ran. we started by implementing the 13th scenario then we refactored the hole code and fixed all bad smells and bugs found by sonar. Also we maintained a very high code coverage as shown below :

Sonar Summery before:



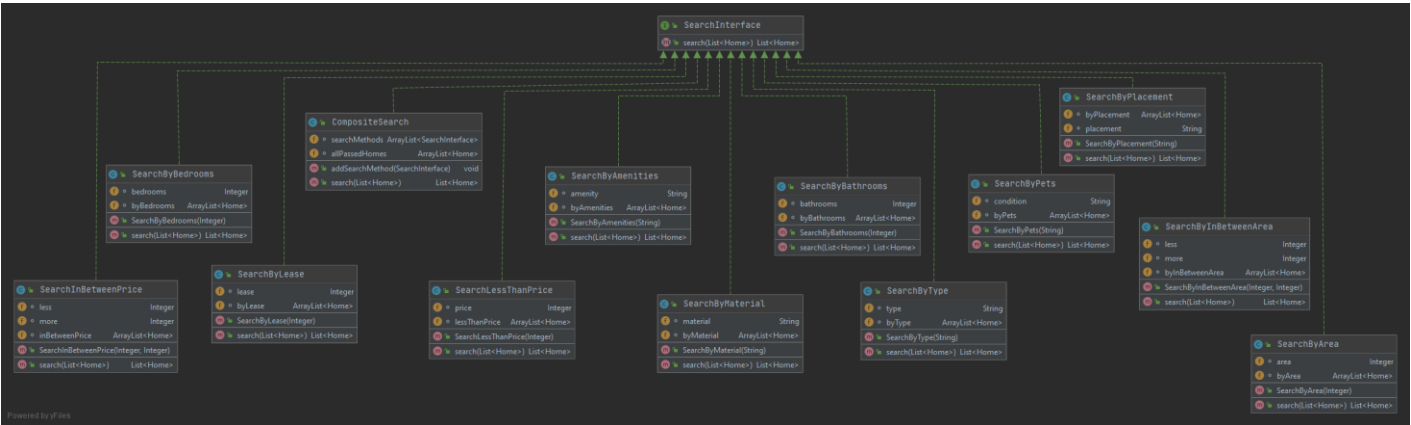
Sonar Summery after:



Class Diagram:

Note: for better view check the png file included, OR FOLLW LINK:

<https://imgur.com/GJZknnQ>



Code coverage:

Coverage: TestRunner ×				
100% classes, 99% lines covered in package 'searchfeature'				
Element	Class, %	Method, %	Line, %	Branch, %
CompositeSearch	100% (1/1)	100% (2/2)	100% (8/8)	100% (2/2)
SearchByAmenities	100% (1/1)	100% (2/2)	100% (9/9)	100% (6/6)
SearchByArea	100% (1/1)	100% (2/2)	100% (8/8)	100% (4/4)
SearchByBathrooms	100% (1/1)	100% (2/2)	100% (8/8)	100% (4/4)
SearchByBedrooms	100% (1/1)	100% (2/2)	100% (8/8)	100% (4/4)
SearchByInBetweenArea	100% (1/1)	100% (2/2)	100% (10/10)	83% (5/6)
SearchByLease	100% (1/1)	100% (2/2)	100% (9/9)	100% (4/4)
SearchByMaterial	100% (1/1)	100% (2/2)	100% (8/8)	100% (4/4)
SearchByPets	100% (1/1)	100% (2/2)	90% (10/11)	70% (7/10)
SearchByPlacement	100% (1/1)	100% (2/2)	100% (8/8)	100% (4/4)
SearchByType	100% (1/1)	100% (2/2)	100% (8/8)	100% (4/4)
SearchInBetweenPrice	100% (1/1)	100% (2/2)	100% (9/9)	100% (6/6)
SearchLessThanPrice	100% (1/1)	100% (2/2)	100% (8/8)	100% (4/4)

Element	Class, %	Method, %	Line, %	Branch, %
Home	100% (1/1)	100% (22/22)	100% (34/34)	100% (0/0)

Composition search class code:

Note: comments contains explanation.

```
public class CompositeSearch implements SearchInterface {

    // a reference to the 'SearchInterface' only exists on the composite class
    ArrayList<SearchInterface> searchMethods = new ArrayList<>();

    // final home list that will be returned after filtering
    ArrayList<Home> allPassedHomes;

    // to add a new type of filter (any obj of the previous 12 classes) we use this method
    public void addSearchMethod(SearchInterface search) {
        searchMethods.add(search);
    }


    // 'search' is the method that all 13 classes implements but here in the composite class
    // we treat it in a special way such that it's behavior is not like others
    public List<Home> search(List<Home> homeList) {

        // we assign the 'homeList' to our final reference 'allPassedHomes'
        allPassedHomes = (ArrayList<Home>)homeList;

        // in this for loop, we will iterate the 'searchMethods' which has different objects from the 12
        // previous classes, we will filter 'homeList' using the first obj algo then pass it again to the
        // second obj algo then to the third till the 'homeList' passed all filters in the 'searchMethods' arraylist
        // and when that is done, we achieved an advanced search with multiple filters at the same time using
        // Composition Pattern
        for (SearchInterface obj : searchMethods)
            allPassedHomes = (ArrayList<Home>)obj.search(allPassedHomes);

        // return the filtered homeList
        return allPassedHomes;
    }
}
```

Jenkins:

 Jenkins

Jenkins

Test

#23

Back to Project

Status

Changes

Console Output

View as plain text

Edit Build Information

Delete build #23

Git Build Data

No Tags

Previous Build

Console Output

Started by user Abdelrahman babaa

Running as SYSTEM

Building in workspace C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Test

The recommended git tool is: NONE

using credential 52004468-3268-4f99-9c67-ab2440881868

> C:\Program Files\Git\bin\git.exe rev-parse --is-inside-work-tree # timeout=10

fetching changes from the remote git repository

> C:\Program Files\Git\bin\git.exe config remote.origin.url https://github.com/MaysaM-R-Mousa/SE-P.git # timeout=10

fetching upstream changes from https://github.com/MaysaM-R-Mousa/SE-P.git

> C:\Program Files\Git\bin\git.exe --version # timeout=10

> git --version # 'git version 2.24.1.windows.2'

using GIT_ASKPASS to set credentials

> C:\Program Files\Git\bin\git.exe fetch --tags --force --progress -- https://github.com/MaysaM-R-Mousa/SE-P.git --refs/heads/*:refs/remotes/origin/* # timeout=10

> C:\Program Files\Git\bin\git.exe rev-parse --refs/remotes/origin/main^{commit} # timeout=10

Checking out Revision d486ebc7345db0bc21841022521ad150381d55c (refs/remotes/origin/main)

> C:\Program Files\Git\bin\git.exe config core.sparsecheckout # timeout=10

> C:\Program Files\Git\bin\git.exe checkout -f d486ebc7345db0bc21841022521ad150381d55c # timeout=10

Commit message: "add mail"

> C:\Program Files\Git\bin\git.exe rev-list --no-walk d486ebc7345db0bc21841022521ad150381d55c # timeout=10

[Test] \$ cmd.exe /C "C:\Users\babaa\Downloads\Compressed\apache-maven-3.6.3\bin\mvn.cmd sonar:sonar -Dsonar.host.url=http://localhost:9000 -Dsonar.log.level=DEBUG&&1f9501sc7273ac0f3c9f&8& exit %ERRORLEVEL%"

[INFO] Scanning for projects...

[INFO]

[INFO] -----< org.example:SE-P >-----

[INFO] Building SE-P 1.0-SNAPSHOT

[INFO] -----[jar]-----

Downloading from central: https://repo.maven.apache.org/maven2/io/cucumber/messages/maven-metadata.xml

Progress (1): 1.7 kB

Downloaded from central: https://repo.maven.apache.org/maven2/io/cucumber/messages/maven-metadata.xml (1.7 kB at 2.0 kB/s)

Downloading from central: https://repo.maven.apache.org/maven2/io/cucumber/messages/13.2.1/messages-13.2.1.pom

Progress (1): 2.7/4.9 kB

Progress (1): 4.9 kB

Downloaded from central: https://repo.maven.apache.org/maven2/io/cucumber/messages/13.2.1/messages-13.2.1.pom (4.9 kB at 22 kB/s)

[INFO]

[INFO] --- sonar-maven-plugin:3.7.0.1746:sonar (default-cli) @ SE-P ---

[INFO] User cache: C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Test\sonar\cache

[INFO] SonarQube version: 8.5.1

[INFO] Default locale: "en_US", source code encoding: "UTF-8"

[INFO] Load global settings

[INFO] Load global settings (done) | time=41ms

[INFO] Server id: 8F41A1F2-AD04F0-30E1a0vUcxX

[INFO] User cache: C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Test\sonar\cache

[INFO] Load/download plugins

[INFO] Load plugins index

[INFO] Load plugins index (done) | time=35ms

[INFO] Load/download plugins (done) | time=359ms

[INFO] Process project properties

[INFO] Process project properties (done) | time=7ms

[INFO] Execute project builders

[INFO] Execute project builders (done) | time=3ms

[INFO] Project key: org.example:SE-P

[INFO] Base dir: C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Test

[INFO] Working dir: C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Test\target\sonar

[INFO] Load project settings for component key: 'org.example:SE-P'

[INFO] Load project settings for component key: 'org.example:SE-P' (done) | time=15ms

[INFO] Load quality profiles

[INFO] Load quality profiles (done) | time=40ms

[INFO] Auto-configuring with CI 'Jenkins'

[INFO] Load active rules

[INFO] Load active rules (done) | time=784ms

[INFO] Indexing files...

[INFO] Project configuration:

[INFO] 19 files indexed

[INFO] 0 files ignored because of scm ignore settings

[INFO] Quality profile for Java: Sonar way

[INFO] Quality profile for XML: Sonar way

[INFO] ----- Run sensors on module SE-P -----

[INFO] Load metrics repository

[INFO] Load metrics repository (done) | time=13ms

[INFO] Sensor JavaSquidSensor [Java]

[INFO] Configured Java source version (sonar.java.source): 7

[INFO] JavaClasspath Initialization

[INFO] JavaClasspath initialization (done) | time=8ms

[INFO] JavaTestClasspath Initialization

[INFO] JavaTestClasspath initialization (done) | time=2ms

[INFO] Java Main Files AST scan

[INFO] 16 source files to be analyzed

[INFO] Load project repositories

[INFO] Load project repositories (done) | time=15ms

[INFO] 16/16 source files have been analyzed

[INFO] Java Main Files AST scan (done) | time=2295ms

[INFO] Java Test Files AST scan

[INFO] 2 source files to be analyzed

[INFO] Java Test Files AST scan (done) | time=113ms

[INFO] 2/2 source files have been analyzed

[INFO] Java Generated Files AST scan

[INFO] 0 source files to be analyzed

[INFO] Java Generated Files AST scan (done) | time=8ms

[INFO] 0/0 source files have been analyzed

[INFO] Sensor JavaSquidSensor [Java] (done) | time=2531ms

[INFO] Sensor CSS Rules [cssfamily]

[INFO] No CSS, PHP, HTML or VueJS files are found in the project. CSS analysis is skipped.

[INFO] Sensor CSS Rules [cssfamily] (done) | time=3ms

[INFO] Sensor JaCoCo XML Report Importer [Jacoco]

[INFO] 'sonar.coverage.jacoco.xmlReportPaths' is not defined. Using default locations: target/site/jacoco/jacoco.xml,target/site/jacoco-it/jacoco.xml,build/reports/jacoco/test/jacocoTestReport.xml

[INFO] No report imported, no coverage information will be imported by JaCoCo XML Report Importer

[INFO] Sensor JaCoCo XML Report Importer [Jacoco] (done) | time=3ms

[INFO] Sensor CE Properties [csharp]

[INFO] Sensor CE Properties [csharp] (done) | time=8ms

[INFO] Sensor SurefireSensor [Java]

[INFO] parsing [C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\workspace\Test\target\surefire\reports]

[INFO] Sensor SurefireSensor [Java] (done) | time=42ms

[INFO] Sensor JavaMLSensor [Java]

[INFO] 1 source files to be analyzed

[INFO] Sensor JavaMLSensor [Java] (done) | time=181ms

[INFO] 1/1 source files have been analyzed

[INFO] Sensor HTML [web]

[INFO] Sensor HTML [web] (done) | time=2ms

[INFO] Sensor XML Sensor [xml]

[INFO] 1 source files to be analyzed

[INFO] Sensor XML Sensor [xml] (done) | time=94ms

[INFO] Sensor VB.NET Properties [vbnet]

[INFO] 1/1 source files have been analyzed

[INFO] Sensor VB.NET Properties [vbnet] (done) | time=1ms

[INFO] ----- Run sensors on project -----

[INFO] Sensor Zero Coverage Sensor

[INFO] Sensor Zero Coverage Sensor (done) | time=14ms

[INFO] Sensor Java CPD Block Indexer

[INFO] Sensor Java CPD Block Indexer (done) | time=16ms

[INFO] SCM Publisher SCM provider for this project is: git

[INFO] SCM Publisher 3 source files to be analyzed

[INFO] SCM Publisher 3/3 source files have been analyzed (done) | time=48ms

[INFO] CPD Executor 1 file had no CPD blocks

[INFO] CPD Executor Calculating CPD for 15 files

[INFO] CPD Executor CPD calculation finished (done) | time=10ms

[INFO] Analysis report generated in 51ms, dir size=134 kB

[INFO] Analysis report compressed in 71ms, zip size=40 kB

[INFO] Analysis report uploaded in 19ms

[INFO] ANALYSIS SUCCESSFUL, you can browse http://localhost:9000/dashboard?id=org.example:SE-P

[INFO] Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report

[INFO] More about the report processing at http://localhost:9000/api/ci/task?id=AX0wZFlkXz_MQq1TK1K1

[INFO] Analysis total time: 5.370 s

[INFO] -----

[INFO] BUILD SUCCESS

[INFO] -----

[INFO] Total time: 8.211 s

[INFO] Finished at: 2020-11-20T18:02:56+02:00

[INFO] -----

Finished: SUCCESS

REST API Jenkins 2.249.3