

Project Software Reengineering

Robbe Claessens
robbe.claessens@student.uantwerpen.be

Tim Leys
tim.leys@student.uantwerpen.be

March 25, 2019

1 Introduction

2 Tool Choice

2.1 Tools for Test Coverage

To select the tool for the test coverage, we selected four of the most widely used coverage tools for Java and compared their results. The tools that were reviewed for this project were: EMMA, JaCoCo, IntelliJ IDEA built in analyzer, and Clover. The results of each tool are presented in Table 1.

As seen in Table 1, most tools produce similar results, with the exception of Clover. Clover also does not feature line coverage, but features statement coverage. However, though we would expect similar results to the line coverage, it is much less.

The built-in tool of IntelliJ does not feature any way of showing the branch coverage and produced a different output in a previous run.

This left us with two remaining tools, EMMA and JaCoCo. Both tools are able to compute the line coverage, the branch coverage, and the method coverage and produce similar results. Because the JaCoCo tool is integrated in the IDE that was used in this project and is compatible with the maven site plugin, this was the best choice for the project.

	EMMA	JaCoCo	IJ IDEA	Clover
Line Coverage	71.2%	71%	71%	57.2% (statement)
Branch Coverage	46.7%	46%	/	48.7%
Method Coverage	72.1%	71%	71%	62.8%

Table 1: Test coverage according to EMMA

	Coverage
Line	71.2%
Branch	46.7%
Method	72.1%

Table 2: Test coverage according to EMMA

2.2 Tools for Mutation Tests

3 Test Coverage

3.1 Initial Test Coverage

	Coverage
Line	58.3%
Class	82.8%
Method	63.3%

Table 3: Test coverage according to IntelliJ IDEA

	Coverage
Line	71%
Branch	46%
Method	71%
Class	89%

Table 4: Test coverage according to JaCoCo