

Project Intermediate Report

Software Reengineering

Van Muylder Ben & Geeraert Lander

April 17, 2019

0 Introduction

1 Tools Used

First and foremost we want to mention that often use JetBrains IntelliJ and Codescene as a starting point as they both provide functionality for multiple of the below mentioned subsections.

1.1 Duplicate Code

As an initial tool, we'll use IntelliJ (which provides simple assisting messages when it detects duplicate code) on the relevant parts of the code. Furthermore we'll use iClones to help us with further detection of duplicate code.

666	CloneClass	143		
667	334	./src/main/java/org/jfree/chart/renderer/category/LayeredBarRenderer.java	296	314
668	333	./src/main/java/org/jfree/chart/renderer/category/StatisticalBarRenderer.java	326	344
669	332	./src/main/java/org/jfree/chart/renderer/category/StatisticalBarRenderer.java	474	491
670	CloneClass	4		
671	8	./src/main/java/org/jfree/chart/renderer/category/LayeredBarRenderer.java	361	392
672	10	./src/main/java/org/jfree/chart/renderer/category/StatisticalBarRenderer.java	281	315
673	9	./src/main/java/org/jfree/chart/renderer/category/StatisticalBarRenderer.java	429	463
674	CloneClass	201		
675	468	./src/main/java/org/jfree/chart/axis/ValueAxis.java	850	867
676	469	./src/main/java/org/jfree/chart/axis/ValueAxis.java	899	916
677	CloneClass	7		
678	18	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	2379	2405
679	17	./src/main/java/org/jfree/chart/plot/XYPlot.java	2515	2542
680	CloneClass	14		
681	32	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	2568	2586
682	31	./src/main/java/org/jfree/chart/plot/XYPlot.java	2606	2625
683	CloneClass	17		
684	38	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	2651	2675
685	37	./src/main/java/org/jfree/chart/plot/XYPlot.java	2714	2739
686	CloneClass	22		
687	47	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	2749	2776
688	48	./src/main/java/org/jfree/chart/plot/XYPlot.java	2747	2774
689	CloneClass	41		
690	89	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	3578	3597
691	90	./src/main/java/org/jfree/chart/plot/XYPlot.java	3226	3245
692	CloneClass	58		
693	139	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	3631	3644
694	138	./src/main/java/org/jfree/chart/plot/XYPlot.java	3342	3354
695	CloneClass	93		
696	221	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	3993	4009
697	222	./src/main/java/org/jfree/chart/plot/XYPlot.java	3898	3913
698	CloneClass	112		
699	265	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	4215	4230
700	264	./src/main/java/org/jfree/chart/plot/XYPlot.java	4240	4253
701	CloneClass	155		
702	358	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	4869	4891
703	357	./src/main/java/org/jfree/chart/plot/XYPlot.java	5469	5491
704	CloneClass	158		
705	363	./src/main/java/org/jfree/chart/plot/CategoryPlot.java	4969	4989
706	364	./src/main/java/org/jfree/chart/plot/XYPlot.java	5531	5550

Figure 1: iClones Duplication Report

The duplication report showed us the presence of many duplications, mostly in the test classes, but some in the source classes as well. The question still remains how many of these duplications will actually be relevant for this assignment.

1.2 Metrics and Visualisation

For metrics and visualisation (as well as other things) we will mostly be using Codescene.

Analysis Scope

Project information, analysis scope, evolutionary data, and file content metrics.

Project	Softengineering	Commits	2,579
Includes History From	2007-06-29 13:35:09 (UTC)	Entities	1,135
Analyzed At	2019-03-27 11:00:59 (UTC)	Changed entities	15,289
Duration	1 minute and 5 seconds	Authors	15
		Active Authors	2

File Content

Language	Files	Code	Comment	Blank
Java	987	132,329	133,625	26,521
Properties	42	1,480	0	60
Markdown	1	1,180	0	236
Text	3	1,170	0	203
HTML	46	362	0	0
XML	1	216	0	29
JavaScript	5	199	65	33
Shell Script	1	0	1	1

Figure 2: Codescene Analysis

<https://codescene.io/projects/4473/jobs/12562/results>

1.3 Mining Repositories

We used Gsource, but as this mostly only provides a visual interactive history of your repository, this tool will be of no further use to us.

We'll also use Codescene for this.

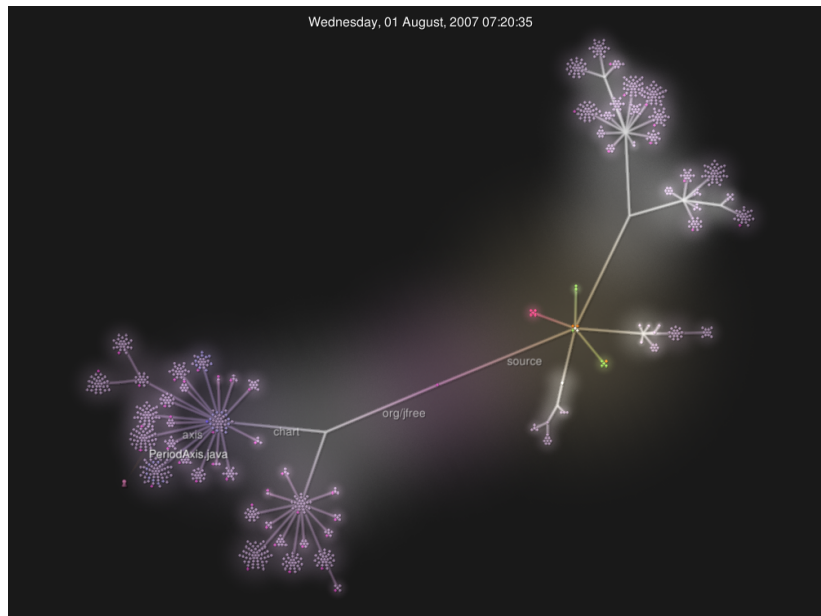


Figure 3: Gsrouce in process

1.4 Refactoring Assistants

As a baseline, we'll use IntelliJ to help us refactor (and specify what and how we should refactor) the project as well as Codescene which gives clear indications on what it thinks should be refactored. We will also have to lookup whether the assistance provided by codescene is relevant for our assignment.

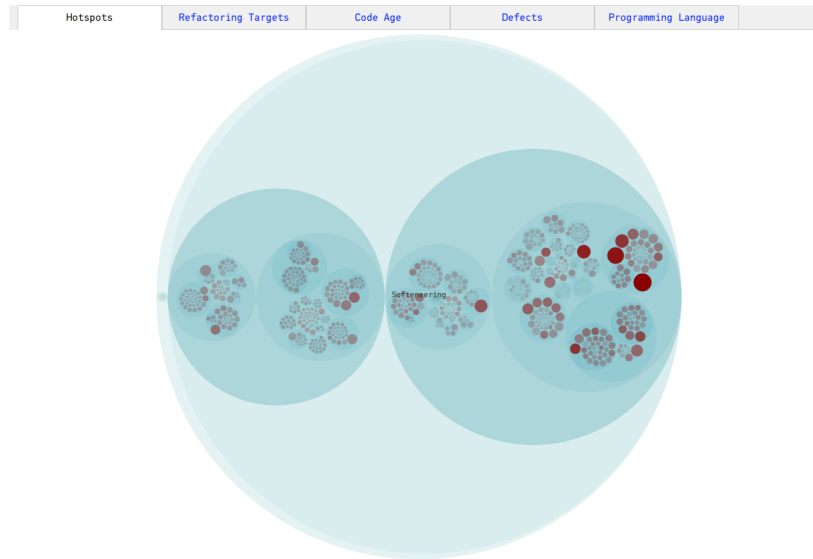


Figure 4: Codescene Hotspots

Here we can clearly see that both the classes `XYPlot` and `CategoryPlot` are highlighted as hotspots. This will most presumably be the classes most relevant for this assignment to refactor.

1.5 Test Coverage and Dynamic Analysis

For test coverage, we plan on using `LittleDarwin` to perform mutation testing on the project. However, during our assignments for the Software Testing course, we already used `LittleDarwin` for that project, and even on such a small project it took a fairly long while to run `LittleDarwin`. Because of this, and because of the other tasks, assignments... we have, we have not yet used `LittleDarwin`.

We also used `cobertura` (which was already present in the project) to get an initial overview of the test coverage.

Package	# Classes	Line Coverage	Branch Coverage	Complexity
All Packages	658	57%	49%	2.67
org.freetest	295	90%	90%	2.474
org.freetest.configuration	26	90%	90%	2.390
org.freetest.kit	47	90%	90%	9.025
org.freetest.kitblock	21	70%	90%	2.81
org.freetest.kitblock	3	90%	90%	2.18
org.freetest.kitblock	12	90%	90%	2.098
org.freetest.kitblock	6	90%	90%	1.113
org.freetest.kitblock	14	90%	90%	2.071
org.freetest.kitblock	19	90%	90%	1
org.freetest.kitblock	7	90%	90%	1.844
org.freetest.kitblock	36	90%	90%	2.237
org.freetest.kitblock	10	90%	90%	2.701
org.freetest.kitblock	3	90%	90%	3.821
org.freetest.kitblock	495	90%	90%	2.781
org.freetest.kitblock	14	90%	90%	1.46
org.freetest.kitblock	15	90%	90%	2.462
org.freetest.kitblock	27	90%	90%	3.049
org.freetest.kitblock	44	90%	90%	3.072
org.freetest.kitblock	1	90%	90%	1
org.freetest.kitblock	8	90%	90%	3
org.freetest.kitblock	10	90%	90%	2.744
org.freetest.kitblock	24	90%	90%	2.054
org.freetest.kitblock	13	90%	90%	3.143
org.freetest.kitblock	32	90%	90%	3.409
org.freetest.kitblock	27	90%	90%	2.811
org.freetest.kitblock	7	90%	90%	2.802
org.freetest.kitblock	5	90%	90%	1.82
org.freetest.kitblock	6	90%	90%	2.022
org.freetest.kitblock	26	90%	90%	3.244
org.freetest.kitblock	1	90%	90%	2.887
org.freetest.kitblock	3	90%	90%	4.402
org.freetest.kitblock	1	90%	90%	4.5
org.freetest.kitblock	6	90%	90%	5.722
org.freetest.kitblock	6	90%	90%	1
org.freetest.kitblock	18	90%	90%	2.54
org.freetest.kitblock	27	90%	90%	2.373
org.freetest.kitblock	4	90%	90%	1.854
org.freetest.kitblock	8	90%	90%	2.807
org.freetest.kitblock	52	90%	90%	2.963

Figure 5: Cobertura Coverage Report

1.6 Feature Location and Traceability

Again, we will use IntelliJ as a baseline, but if it seems necessary, or if it seems that the functionality provided by IntelliJ is too limited, we'll look at other tools.

(we provide no screenshot for this subsection)