

Document Information

Analyzed document	Zong_Xingrong_A3.pdf (D103012042)
Submitted	4/27/2021 11:41:00 AM
Submitted by	
Submitter email	xz222bb@student.lnu.se
Similarity	66%
Analysis address	mymoodle.LNU@analys.orkund.se

Sources included in the report

SA	Linnéuniversitetet / demir_dennis_A3.pdf		
	Document demir_dennis_A3.pdf (D99141153)		2
	Submitted by: dd222gc@student.lnu.se Receiver: mymoodle.LNU@analys.orkund.se		
SA	Linnéuniversitetet / Hmidi_Katronnada_A3.pdf		
	Document Hmidi_Katronnada_A3.pdf (D101585155)		1
	Submitted by: kh222yg@student.lnu.se Receiver: mymoodle.LNU@analys.orkund.se		
SA	Linnéuniversitetet / Heiding_Camilla_A3.pdf		
	Document Heiding_Camilla_A3.pdf (D66688572)		2
	Submitted by: ch223cd@student.lnu.se Receiver: mymoodle.LNU@analys.orkund.se		

Entire Document

Assignment 3 [Subtitle] Author: Xingrong Zong

Supervisor: [Supervisor]

52%

MATCHING BLOCK 1/5

SA

demir_dennis_A3.pdf (D99141153)

Semester: VT21 Course code: 2DV608 Report Table of contents 1 Task 1 – Codebase Analysis
 3 2 Task 2 – Re-engineering Plan 7 3 Task 3 – Re-engineering 8 3 (8) 1 Task 1 – Codebase Analysis Figure 1: System Analysis In Figure 1 the Size of the system analysis is shown. There are a total of 4688 lines among which 509 lines are comments and 3130 lines are actual code. There are 1 package and 19 components (classes). In

Issues section, there are eight issues detected among which seven of them

41%

MATCHING BLOCK 2/5

SA

Hmidi_Katronnada_A3.pdf (D101585155)

are threshold violations and the other one is cycle groups. Under Structure section, it shows the system maintainability level is quite high. The system average component dependency (ACD) is 5.63 which means each class depend on average upon 5.63 other classes. There are

four

68%

MATCHING BLOCK 4/5

SA

Heiding_Camilla_A3.pdf (D66688572)

cyclic components which result in a structural debt of 46. To remove all the cyclic dependencies among these four cyclic components there are three dependencies to remove. 4 (8) Figure 2: Issues detected by Sonargraph, 8 warnings In Figure 2 the Issues section of the analysis is shown, it shows that there are eight issues detected as already shown in Figure 1. As mentioned, seven of them are threshold violations and one is

cycle group. All issues have the severity of a warning.

Figure 3:

Cycle

Group, 4 cyclic components in exploration view

5 (8) Figure 4: Component cycle group, 4 cyclic components in cycle

72%

MATCHING BLOCK 5/5

SA

Heiding_Camilla_A3.pdf (D66688572)

view In Figure 3 and 4, the dependencies among the cyclic components from the Figure 1 are shown using the exploration view and cycle view of Sonargraph. The classes AboutDialog, imageRescaleDialog, brickMosaic and ColorReduceDialog components have both upward and downward going dependencies which means there is high coupling between the components. Since this issue only have the severity of a warning, it is not critical for the application but make maintenance more difficult. The source of the problem is probably a bad architecture, if it would for example have used a layered architecture there should not be any upward going dependencies. Each layer should only depend on the layer below. 6 (8)

Figure 5: Threshold violations in exploration view Figure 5 shows the issues of threshold violations

which are the methods `createDialog()`, `doInBackground`, `createDialog()`, `actionPerformed(ActionEvent)` and `initialize()` in four different classes. As seen in Figure 2, the allowed modified cyclomatic complexity is 0 to 15, but methods have 20 and 24. These values mean

60%

MATCHING BLOCK 3/5

SA

demir_dennis_A3.pdf (D99141153)

that the code has 20 and 24 different linearly independent execution paths so technically need 20 and 35 different tests only to test

these two methods. It is not that critical because the severity is warning. To solve the problem the method could be split so it would both increase cohesion and decrease complexity. Also, the allowed number of statements should be 0 to 100, but these methods all have over 100 statements. However, they are all at the severity of warning. To solve this issue the duplicate code could be generalize into a method.

7 (8) 2 Task 2 – Re-engineering Plan In order to remove the threshold violations. The plan assumes that the over ranged statements are duplicate codes that can be removed and replaced by methods. Remove threshold violations in `brickMosaic.java`, `ImageRescaleDialog.java`, `DocWriterTask.java`, `ColorReduceDialog.java` Make duplicate codes into methods.

8 (8) 3 Task 3 – Re-engineering The number of statements decreases a bit, but it shows more issue at severity of error that java file parse error because cannot find the new created parameters and methods.

Hit and source - focused comparison, Side by Side

Submitted text As student entered the text in the submitted document.
Matching text As the text appears in the source.

1/5	SUBMITTED TEXT	95 WORDS	52% MATCHING TEXT	95 WORDS
	<p>Semester: VT21 Course code: 2DV608 Report Table of contents 1 Task 1 – Codebase Analysis 3 2 Task 2 – Re-engineering Plan 7 3 Task 3 – Re-engineering 8 3 (8) 1 Task 1 – Codebase Analysis Figure 1: System Analysis In Figure 1 the Size of the system analysis is shown. There are a total of 4688 lines among which 509 lines are comments and 3130 lines are actual code. There are 1 package and 19 components (classes). In</p>		<p>Semester: Spring 2021 Course code: 2DV608 Table of contents 2. Task 2 - Re-Engineering Plan 3. Task 3 - Re-Engineering 1. Task 1 - Code analysis 3 1. Task 1 - Code base analysis In this first the first of the system is shown. seen in the second there are total of 9044 lines among which 1656 are comments and 5205 actual code. There are 1 package and 21 components which are classes. In</p>	
SA	demir_dennis_A3.pdf (D99141153)			

2/5	SUBMITTED TEXT	45 WORDS	41% MATCHING TEXT	45 WORDS
	<p>are threshold violations and the other one is cycle groups. Under Structure section, it shows the system maintainability level is quite high. The system average component dependency (ACD) is 5.63 which means each class depend on average upon 5.63 other classes. There are</p>		<p>are threshold violations and one is a cycle Adding to that, we can also see that maintainability level which is quite high. The average component dependency (ACD) is 5.63 which means, on each class depends upon 5.63 other classes. In there are</p>	
SA	Hmidi_Katronnada_A3.pdf (D101585155)			

4/5	SUBMITTED TEXT	79 WORDS	68% MATCHING TEXT	79 WORDS
	cyclic components which result in a structural debt of 46. To remove all the cyclic dependencies among these four cyclic components there are three dependencies to remove. 4 (8) Figure 2: Issues detected by Sonargraph, 8 warnings In Figure 2 the Issues section of the analysis is shown, it shows that there are eight issues detected as already shown in Figure 1. As mentioned, seven of them are threshold violations and one is		cyclic components which result in a structural debt of 173 which quite high. remove all the cyclic dependencies among these 14 dependencies; components there are 11 dependencies; dependencies to remove. Figure 2 System analysis part 2 Figure 3 Issues detected by Sonargraph, 2 warnings errors In figure 3 the issues section of the analysis is shown which show that there are four issues detected as already shown in figure 2. As said three of them are cycle groups and one is	

SA Heiding_Camilla_A3.pdf (D66688572)

[illegible]

3/5	SUBMITTED TEXT	24 WORDS	60% MATCHING TEXT	24 WORDS
	that the code has 20 and 24 different linearly independent execution paths so technically need 20 and 35 different tests only to test		that the code has 156 different linear-independent execution paths, which mean you would technically need 156 different tests only to test	
SA	demir_dennis_A3.pdf (D99141153)			