Task 1

Size

1. Total LOC = 22539. Total LOC in main.java.memoranda = 2187.
2. Largest code file is main.java.memoranda.ui.htmleditor.HTMLEditor.java with 2144 LOC. Largest code file in main.java.memoranda is EventsManager.java with 329 LOC.
3. The metrics tools counted 28 LOC in CurrentNote.java, but the source code file itself has a line count of 38. It seems as though the metric tool used any line with text on it (including comments) to determine the Total LOC. The only thing not included was blank lines.

Cohesion

1. This metrics tool calculates LCOM2 by finding subtracting the set of methods defined by the class from the average number of methods that access each field defined by the class divided by 1 minus the set of methods defined by the class.

   Henderson-Sellers defines Lack of Cohesion in Methods as follows. Let:

   M be the set of methods defined by the class
   F be the set of fields defined by the class
   $\rho(f)$ be the number of methods that access field f, where f is a member of **F**
   $<\rho>$ be the mean of $\rho(f)$ over **F**.

   Then:

   $$\text{Lack of Cohesion in Methods} = \frac{<\rho> - |\mathbf{M}|}{1 - |\mathbf{M}|}$$

   Note 1: I have only included methods if they access at least one field.

   Note 2: I have only included fields if they are accessed by at least one method in the class.

2. Several files in main.java.memoranda have the highest cohesion with a lack of cohesion value of 0. When looking at Note.java, it is apparent that this high cohesion is due to the fact that the class has no defined methods. As a result, both <p> and |M| = 0.

Complexity

1. The cyclomatic complexity in main.java.memoranda is 1.746.
2. The class with the highest cyclomatic complexity in main.java.memoranda is EventsManager.java, with a value of 2.5 (mean).
3. To reduce the complexity of EventsManager.java, I modified the worst method, getRepeatableEventsforDate, to extract certain conditional checks that returned the same result into a separate method (called isRepeated). This reduced the method's complexity from 16 12, and EventsManager.java's complexity to 2.3 (mean).

Coupling

1. Afferent couplings determine a package's responsibility (the number of classes in other packages that depend on it), while efferent couplings indicate a packages dependency on externalities (the number of classes in other packages it relies on).
2. Main.java.memoranda.util has the highest afferent coupling value at 57.
3. Main.java.memoranda.ui has the highest efferent coupling value at 49.

Worst Quality

- Of all the different metrics measured with this tool, only three are highlighted in red as main concerns. Using those metrics as the basis for determining the worst quality will mean selecting the class with the highest McCabe Cyclomatic Complexity, the highest average Number of

Parameters, and the highest average Nested Block Depth. Looking in main.java.memoranda, there are two classes that are highlighted at least twice – EventsManager.java and NotesImpl.java. However, although NotesImpl has both the highest complexity and nested block depth, it has a decent average number of parameters. On the other hand, EventsManager comes in second worst place for each of the three metrics (after my complexity fix). Taking all metrics into account therefore, I would have to say that EventsManager.java has the worst overall quality within its package.

Task 2

Metric	Total	Mean	Std. D...	Maxim...	Resource causing Maximum	Method
McCabe Cyclomatic Complexity (avg/max		2.237	2.839	42	/SER316-Spring-2018/src/main/java/memoranda...	setTableProperties
Number of Parameters (avg/max per meth		0.929	1.097	9	/SER316-Spring-2018/src/main/java/memoranda...	setImageProperties
Nested Block Depth (avg/max per method		1.39	0.955	8	/SER316-Spring-2018/src/main/java/memoranda...	getNotesForPeriod
Afferent Coupling (avg/max per packageFr		19.333	19.653	57	/SER316-Spring-2018/src/main/java/memoranda...	
Efferent Coupling (avg/max per packageFr		11.444	15.276	49	/SER316-Spring-2018/src/main/java/memoranda...	
Instability (avg/max per packageFragment		0.36	0.247	0.778	/SER316-Spring-2018/src/main/java/memoranda...	
Abstractness (avg/max per packageFragm		0.111	0.137	0.333	/SER316-Spring-2018/src/main/java/memoranda...	
Normalized Distance (avg/max per packag		0.529	0.237	1	/SER316-Spring-2018/src/main/java/memoranda...	
Depth of Inheritance Tree (avg/max per typ		2.652	1.934	6	/SER316-Spring-2018/src/main/java/memoranda...	
Weighted methods per Class (avg/max pe	3251	14.135	25.519	242	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Children (avg/max per type)	60	0.261	1.405	16	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Overridden Methods (avg/max	59	0.257	0.691	4	/SER316-Spring-2018/src/main/java/memoranda...	
Lack of Cohesion of Methods (avg/max pe		0.262	0.398	1.2	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Attributes (avg/max per type)	1326	5.765	14.118	101	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Static Attributes (avg/max per	136	0.591	1.793	12	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Methods (avg/max per type)	1269	5.517	6.833	42	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Static Methods (avg/max per t	184	0.8	2.539	18	/SER316-Spring-2018/src/main/java/memoranda...	
Specialization Index (avg/max per type)		0.15	0.487	5	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Classes (avg/max per package	230	25.556	29.833	92	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Interfaces (avg/max per packa	16	1.778	3.292	11	/SER316-Spring-2018/src/main/java/memoranda	
Number of Packages	9					
Total Lines of Code	22537					
Method Lines of Code (avg/max per meth	15633	10.759	28.207	346	/SER316-Spring-2018/src/main/java/memoranda...	jbInit

Metric	Total	Mean	Std. D...	Maxim...	Resource causing Maximum	Method
McCabe Cyclomatic Complexity (avg/max		2.237	2.839	42	/SER316-Spring-2018/src/main/java/memoranda...	setTableProperties
Number of Parameters (avg/max per meth		0.929	1.097	9	/SER316-Spring-2018/src/main/java/memoranda...	setImageProperties
Nested Block Depth (avg/max per method		1.39	0.955	8	/SER316-Spring-2018/src/main/java/memoranda...	getNotesForPeriod
Afferent Coupling (avg/max per packageFr		21.6	20.011	57	/SER316-Spring-2018/src/main/java/memoranda...	
Efferent Coupling (avg/max per packageFr		10.6	14.263	49	/SER316-Spring-2018/src/main/java/memoranda...	
Instability (avg/max per packageFragment		0.335	0.243	0.778	/SER316-Spring-2018/src/main/java/memoranda...	
Abstractness (avg/max per packageFragm		0.172	0.301	1	/SER316-Spring-2018/src/main/java/memoranda...	
Normalized Distance (avg/max per packag		0.522	0.251	1	/SER316-Spring-2018/src/main/java/memoranda...	
Depth of Inheritance Tree (avg/max per typ		2.652	1.934	6	/SER316-Spring-2018/src/main/java/memoranda...	
Weighted methods per Class (avg/max pe	3251	14.135	25.519	242	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Children (avg/max per type)	60	0.261	1.405	16	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Overridden Methods (avg/max	59	0.257	0.691	4	/SER316-Spring-2018/src/main/java/memoranda...	
Lack of Cohesion of Methods (avg/max pe		0.262	0.398	1.2	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Attributes (avg/max per type)	1326	5.765	14.118	101	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Static Attributes (avg/max per	136	0.591	1.793	12	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Methods (avg/max per type)	1269	5.517	6.833	42	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Static Methods (avg/max per t	184	0.8	2.539	18	/SER316-Spring-2018/src/main/java/memoranda...	
Specialization Index (avg/max per type)		0.15	0.487	5	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Classes (avg/max per package	230	23	28.174	92	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Interfaces (avg/max per packa	16	1.6	3.169	11	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Packages	10					
Total Lines of Code	22584					
Method Lines of Code (avg/max per meth	15633	10.759	28.207	346	/SER316-Spring-2018/src/main/java/memoranda...	jbInit

After refactoring, a number of metrics changed for the better. Both afferent and efferent coupling, for instance, improved. This change was because I moved classes with high responsibility and some dependencies into an entirely new package. In their original package, they yielded high coupling scores for the main package. As a result of moving them, the responsibility and dependency count of the original package decreased, spreading the load onto the new package and thereby decreasing the overall coupling averages in all.

Task 3

1. For the smell within a class, I found a method in the TaskListImpl.java class that exhibited a Long Parameter List (createTask). This is a code smell because it makes the method difficult to read and understand. To remove the smell, I extracted the parameters into their own object, called TaskParameters in main.java.memoranda. In order to get the project to build again, I also had to modify the TaskList interface. So in total, I added TaskParameter.java to main.java.memoranda and modified the method, createTask, within both main.java.memoranda.TaskListImpl.java and main.java.memoranda.interfaces.TaskList.java.
2. Between classes, I found a speculative generality code smell present among interfaces and classes that don't seem to be implemented or used anywhere in the code. One specific class in main.java.memoranda is the DefaultEventNotifier.java.

Metric	Total	Mean	Std. D...	Maxim...	Resource causing Maximum	Method
McCabe Cyclomatic Complexity (avg/max		2.226	2.829	42	/SER316-Spring-2018/src/main/java/memoranda...	setTableProperties
Number of Parameters (avg/max per meth		0.92	1.072	9	/SER316-Spring-2018/src/main/java/memoranda...	setImageProperties
Nested Block Depth (avg/max per method		1.387	0.951	8	/SER316-Spring-2018/src/main/java/memoranda...	getNotesForPeriod
Afferent Coupling (avg/max per packageFr		21.8	20.178	57	/SER316-Spring-2018/src/main/java/memoranda...	
Efferent Coupling (avg/max per packageFr		10.7	14.304	49	/SER316-Spring-2018/src/main/java/memoranda...	
Instability (avg/max per packageFragment		0.336	0.243	0.778	/SER316-Spring-2018/src/main/java/memoranda...	
Abstractness (avg/max per packageFragm		0.172	0.301	1	/SER316-Spring-2018/src/main/java/memoranda...	
Normalized Distance (avg/max per packag		0.522	0.251	1	/SER316-Spring-2018/src/main/java/memoranda...	
Depth of Inheritance Tree (avg/max per ty		2.645	1.933	6	/SER316-Spring-2018/src/main/java/memoranda...	
Weighted methods per Class (avg/max pe	3263	14.126	25.467	242	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Children (avg/max per type)	60	0.26	1.403	16	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Overridden Methods (avg/max	59	0.255	0.69	4	/SER316-Spring-2018/src/main/java/memoranda...	
Lack of Cohesion of Methods (avg/max pe		0.265	0.399	1.2	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Attributes (avg/max per type)	1333	5.771	14.087	101	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Static Attributes (avg/max per	136	0.589	1.79	12	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Methods (avg/max per type)	1282	5.55	6.851	42	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Static Methods (avg/max per t	184	0.797	2.534	18	/SER316-Spring-2018/src/main/java/memoranda...	
Specialization Index (avg/max per type)		0.149	0.486	5	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Classes (avg/max per package	231	23.1	28.201	92	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Interfaces (avg/max per packa	16	1.6	3.169	11	/SER316-Spring-2018/src/main/java/memoranda...	
Number of Packages	10					
Total Lines of Code	22649					
Method Lines of Code (avg/max per meth	15658	10.681	28.101	346	/SER316-Spring-2018/src/main/java/memoranda...	jbInit

3. Overall, Number of Parameters decreased from an average of 0.929 to 0.92 for the whole project. However, coupling increased overall because I introduced more dependencies between packages by extracting the task parameters into their own separate object.