**Unit 3 Status Report**

Date: May 11, {2018}

To: George Peck

From: {Ian Chiu} Group Members: maxwell wang, darren chou, ian chiu

Subject: Status Report {Period 1}

Accomplishments:

* We completed the first and second GUIs that had been started last week. The first GUI was finished completely – not only is the visual aspects complete, but it is now able to send the information inputted into it to the Student class. The second GUI is almost finished: there is still some left that needs to be done in terms of retaining information and sending it to the algorithm class; however, we were able to overcome some of the difficulties in the beginning of the week. (see problems/risks for more information).
* We also implemented a sorting method for the student class, but some people disagree with this one because they believe that it should honestly be in the lynder class itself. Basically, we had a sorting method which said that we would take in an integer array and then it would sort the numbers using quicksort because that is the fastest way to sort things. Then, we would be able to build off of my last point in which we add all the students ratings of the other student and then come up with a list of the people in order of the ones that they rate the students.
* One groupmate disagreed with the student class sort method, and so we then moved it from the student class to the lynder class because apparently that would be better. It was also changed from an integer array into a priority queue that was based on the names of the students that have rated every student in the master list. However, this is also the same as the one before because the other ones will still be sorted by the amount of ratings that they got, and so it is only the little people changing over time. This is helpful if it works because then, the people will all be sorted perfectly and then the program will work splendidly.

Problems/Risks:

* Another problem that we stumbled upon was that there was a very serious debate about where the list of integers should be, sorry, array of integers should be. Some thought that this array of integers should be in the student class, because that is where one can store the ratings that other people have of them in the array, but some thought that it should instead be in the lynder class, because that is very important to be able to reach the master list of students and then have all the students have their own arrays. This same coworker also believed that the array of integers should instead be a priority queue of students’ names sorted by their ratings. After much debate, mostly dominated by the second coworker, we decided on using the second method of approaching this problem.
* Another major problem, from the GUI standpoint, was that we found difficult to reset the GUI for every student. Essentially, it would be creating a loop inside the GUI that would map the rating for each student. However, we found it difficult because the implementation was a bit tricky and there were constantly errors that were difficult to find the source of, such as an off by one error, or an endless loop. However, yesterday, we made a “breakthrough” where the designer for the GUI realized that it may be easier to actually create a list of names in the GUI, and allow the student to individually click through each one and give them a ranking, eliminating the need for loops in the GUI.

Next Steps:

* Next ,we will make these interfaces and classes interact with each other and create the main class, which will use all of these classes to create the Lynder application. I think that it will be difficult to properly implement all of these seamlessly, because there is a large amount of classes that need to be put together – for example, the three GUIs will have to be able to work seamlessly and close while the next one opens, but also retain the information that had been inputted into it.
* Also, we will begin work on the JUnit test around next week, because that is when we hope to approximately finish most of the Lynder application. Finishing it will allow us to start work on the testing of the GUIs, the algorithm, and other properties of the class.