**Process**

We started this assignment by defining a problem large enough to be completed a two year period. In the end we came up with a marking/assignment type application. It would allow teachers to give students new assignments on-line and allow them to mark and edit student assignments. This program would also allow the students to upload and download work to be marked.

The steps we took to accomplish this task were to define a set of functions required by the overall system to enable it to work properly. After completing this was decided to create a methodology that would work for us. In the end we came up with a five week initial planning period, and an two week iteration period consisting of a review, analysis, design, implementation and testing loop. The next step was to come up with a architecture for our system. For this we concluded that the most appropriate architecture would be one following the Gartner Three Layer Model. For the presentation layer we would have a web based interface made out of HTML5 and CSS, the data management would be a simple file server into which the assignments would be deposited to when uploaded and a SQL database. The application layer model would be made from a java/python server.

The first iteration of the project consisted of designing skeleton class for the system revolving around the students, teachers and classes. To do this we implemented the Mediator design pattern along with the builder pattern. These two patterns allowed us to create different types of instances of the users when required. As well as being able to control the users and classes without touching the class itself by using managers. For the testing of this iteration we checked for objects not being null, no duplicates in lists, when deleting from one list check its removed from the list and placed into the temporary list and vice versa for when restoring the object.

**Review**

This assignment ran rather smoothly there were no major problems with our group sociology. The only technical problem we had was my laptop breaking mid way through the iteration limiting what coding I could do to the time I was a university. During our initial planning stage we decided on using a wiki to store notes of our meetings through the project this started to die off during our first iteration. On the other hand the rest of our iteration went through brilliantly all the classes and tests were completed and we had nothing left that would be required to be pushed to the next iteration. Compared to our last assignment this project seemed to flow a lot smoother and easier than the last. This would have been because in our last assignment we were creating a GUI, whereas in this assignment we stayed away from the GUI and focused on the application layer to start our iterations.

Software development is difficult for multiple reasons the main ones being that the customer isn’t sure exactly what they want from the product through the iterative and incremental lifecycle we can mitigate throughout the product as we give our partially completed software to them and get their feedback in return. The other main reason is that there is no one way of coding a problem with each solution having its own pros and cons, mitigating this is difficult as everyone has their own style of coding that they usually get stuck into a pattern of creating and changing their styles can be difficult. In saying this the best way to minimise this problem would be coding as simply as possible to avoid any accidental unwanted complications occurring.

Mathew Andela 1157447