

Eric Chan  
Plagiarism Checker

### **Project Post Mortem**

What had worked well in doing this project was having all the requirements set before beginning to code. Having the requirements all laid out allowed for me to just implement each function and procedure and then piecing the various components together. Planning the algorithm I had to implement beforehand made it simpler for me to code it as well since I had figured that it would work on paper. Writing out the requirements before I coded also allowed for me to see what could and could not work which saved me a lot of time coding since I wouldn't be implementing a feature just to scrap it because it turned out that it wouldn't work. The requirements became a checklist of things for me to implement which also saved a lot of time as opposed to developing my initial image of the system and adding more features. The latter would have required time for me to think about where the features would fit in logically. This also saved some time when testing since each requirement would become a separate unit which I could test individually for bugs and logical errors. The models from the architecture and design document helped me visualize the system to be implemented.

The progression of testing from unit to component to system allowed for me to locate what and where something went wrong a lot quicker than if I had tested the system in its entirety. It was a lot faster and easier to locate logical errors in my functions and procedures when I was testing each of them individually. Moving up the testing to in pairs and in threes and etcetera, I was able to find and fix most of the bugs before piecing all the components together.

Overall the waterfall model made it simpler to implement all the features that I had set out to do and following the plan made the entire process of developing a software much less stressful. Using the waterfall model, however, didn't allow for me to ask others for their input on my design choices for the user interface. Although I think the user interface is easy to figure out how to use, others might not agree.