**Henry Paredes Jr**

**COSC 603 – Spring 2016**

**Project2**

**Description (2-3 paragraphs) of what you learned from this project and how it relates to some of the topics covered in lecture.**

*Project2 might have a lot of tasks involved, but from lessons learned from Project1 the versioning tasks is easier to execute this time around. The task list provided a recipe like instruction thread that was easy to follow. Refactoring operations took some getting used to, once the rationale for the operations was becoming familiar you gain confidence in doing them.*

*I am still having trouble pushing the versions of the project I have committed from Eclipse to the local repository. Just like last time I basically drag and drop the project to my GitHub account in the browser. I’m not sure if I need to somehow use the GitHub Desktop for this. I’ll keep digging.*

**Description (2-3 paragraphs) of what you liked about Eclipse’s support for refactoring including its strengths and limitations as well as your impression of JDeodorant.**

*I was able to use only a handful of the refactoring operations. Whatever was called for on the tasks at hand. But from these few operations came a better understanding on the usefulness of refactoring. However, from these strength also came weakness such as the false positives that can result. Like anything else there will always be exceptions to rules - sometimes.*

*With respect to Project1, applying the bad JDeodorant operations just learned produced inconclusive result. At first I thought it was something I didn’t set right, but thinking about it more makes me believe that it was how I reengineered the program.*

**Description (1-2 paragraphs) of how/why unit tests are important when doing refactoring.**

*Unit tests facilitates the decomposition of program into smaller pieces that are easy to put into code. Refactoring allows us to decompose the larger chunks of code into possibly unit codes. This facilitates a simple design to our finish product.*