**Henry Paredes Jr**

**COSC 603 – Spring 2016**

**27th February 2016**

The task list for the project made things very manageable. Following through with the tasks involved, task1 involves the setting up of the development environment. In my case, this involves the installation of the most current version Java JDK, the recommended Eclipse IDE which is the Eclipse Luna IDE for Java EE developers 4.4.1 the 32-bit version, and a configuration management tool GitHub desktop. After going through a few YouTube lessons on installing Java JDK, Eclipse Luna IDE, and GitHub the installations of these applications went through smoothly. After additional YouTube lessons on Javadoc and JAutodoc made the installation of Eclipse Luna IDE complete, when the settings were implemented in the IDE.

Task2 involved the setting up the GitHub project and the Git repository. Additional YouTube lessons again helped in this endeavor. After the lessons, the GitHub account was setup without any hitch and the repository was created. Task3 entails further training on Git/GitHub Desktop using Eclipse, and this was tested through the accomplishment of a “certificate of completion” of sorts. This accomplishment was documented by uploading the certificate to the GitHub repository that was created.

The rest of the task on the list dealt with the reengineering of the Legacy code on computing Fire Danger. Analysis of the code, which was coded in Fortran77 took a while since this computer language is not well structured. I had FORTRAN when I started in college, but the training we were giving was very structured. The documentation on Fortran77 was search on the internet instead and helped in unraveling what the code was trying to do. This was probably the most painstaking part of the project along with coding the algorithm in Java. I pretty much combined the analysis and coding in Java, and made the design of the Java code as simple as I can. At first I coded the Danger subroutine as one code, but the nesting in the computation was so deep that I had to simplify it by trying to make smaller methods instead to make it manageable. And now we test this code in Eclipse.