Appendix B. Software Construction Strategy

Describe how you will address the following:

* Have you agreed on a programming language, compiler, editor, debugger?

**Our team agreed on java as our compiler language and Netbeans 8 or later as our IDE. Our team agreed on using JDK 7 as our language version.**

* Will you start with logic or user interface?

**We will start with the user interface then the logic to make the testing more convenient.**

* Will you start with the main menu and move to each module, then per function/class?

**Yes**

**Process:**

**1. The developer will make user interface from the console like a menu (if there is no GUI yet) for his assigned task (Ex: Add Book in the menu)**

**2. After making the user interface, the developer will make a method and start working with the logic**

* How will you check compliance of your code with the coding standards defined for your team?

**After one of our member made the methods work for the component of the program, our team will check if that member followed the coding standards that the Dev team agreed on.**

* What is your coding practice with regards to error handling? security (such as validating user inputs and referential integrity)?

**Our program will check whether the user input is correct and if not, an error message will print and will give the user a 2nd chance to input again.**

**Process:**

**1. The developer runs the program**

**2. Every time after the user inputs, the program will check if the user input is valid and if not, an error message will be printed. The program will give the user a 2nd chance to input again**

**Example: User inputs a character but the required input should be a number from 1-9. An error message will print and say that input from numbers 1 to 9 only. Please input again. After the message, it will print “Input:” to the user to let the user know he can input again.**

* Will programmers code in pairs ("pair programming"), or individually, or a combination of both?

**Individual**

**Process:**

**1. When an assigned task is given to the developer, the developer codes and is expected to finish based on the estimation indicated in user stories in the software requirement specification.**

**2. After finishing the assigned task on or before the deadline, the developer checks for bugs and if there is bugs, he will be given 2-3 days to debug the program.**

**3. After the developer debugs his method/s, this developer should help another developer on finishing his task when his assigned task was already overdue on the deadline.**

**4. If the developer did not finish his assigned task, one or two developer from the team should help him.**

**5. If the developer did not debug the program on the deadline, one or two developer from the team should help him.**

* How will you use GitHub to manage version control of your code?

**Whenever we start to develop critical modules of the software, we will make a separate**

**version to avoid messing up the other modules.**

* How will you handle revisions? What revision control tool will you use?

**We will use Git.**

* Will you allow the use of non-standard language features?

**Yes**

* Will programmers write test cases for their code before writing the code itself?

**We think both can apply because sometimes after writing the code the developer can picture what possible errors can a method/function does.**

**Process:**

**1. The developer will think some of the test cases before writing the method**

**2. After the developer is done writing his method, the developer if possible will make another test case to the method he wrote.**

**3. With all of the test cases, the developer will manually trace with his test cases to see if there are errors in his code.**

**4. After the developer thinks that there are no errors, the developer compiles his code to check whether there are errors.**

**5. If there are errors, the developer debugs the program. If none, the developer can test his program**

* Will programmers step through their code in the debugger before they check-in their code (to the repository)?

**Yes**

**Process:**

**1. The developer steps in his code to the debugger**

**2. If there is an error or bug, the developer debugs them**

**3. After debugging, the developers check-in his code**

* Will programmers integration-test their code before they check-in their code to the code repository?

**Yes**

* Will programmers review or inspect each other’s’ code?

**Yes, in order to verify if they followed the coding standard agreed by the developers.**

**Process:**

**1. After the developer finish his method/s, he checks in his code to the repository**

**2. The developer in the team checks his code whether his code matches in the coding standard**

**3. If some parts of the code does not match in the coding standard, one of the developer in our team edits the code and place a comment on what’s wrong with his code**