Development, Coding, and Testing

1. Using your existing design, requirements and specification documents, begin writing code for your project. Within your requirements document, you should have several distinct tasks to complete. Choose one or two of these tasks and begin work on them.
2. Determine several simple use cases for each of the parts of the tasks you choose to complete. These may be two to three sentences each. It is highly recommended that you incorporate these descriptions into your code as comments.
3. Develop and document test cases for each of the requirements you attempt to complete. Where appropriate, identify boundary conditions, equivalence classes, etc.
4. You are responsible to test your programs on your own. You may find it necessary to write short driver programs or stubs to unit test your existing code.
5. While programming it is highly recommended that you follow an Agile method such as Extreme Programming (XP) or Scrum. It is not required that you implement one of these methods entirely. Rather, make use of the most useful parts of such methods. At minimum you must record a list of issues you encounter while programming. Recommendations include the following:
	1. Utilize pair programming
	2. Regularly assess your progress
	3. Keep track of future, existing, and completed issues and their importance
6. Determine rough time estimates for each of the tasks in your requirements list. Use your previous experience with class assignments, internships, or jobs to determine these estimates. If you are unable to determine an estimate, the problem for which you are creating an estimate is likely too large and should be divided into smaller parts.
7. If you determine that modifications must be made to your designs to complete these tasks, update your design documentation to reflect any modifications made to your code. This may include modification to UML or ER diagrams, requirements documentation, specification documentation, or even your domain analysis.
8. Submit your updated design, code, test cases, brief use cases, issue reports, time estimates, and a brief progress report at the end of the first milestone for project 2. Be sure to submit any short driver programs or stubs you create.