Summary

In conclusion, we have decided to use a combination of J-Unit testing and visual testing to ensure correct functionality. J-Unit testing creates a separate class that involves: setting up the class that will be tested, testing that class, and then tearing everything down. This style of testing is convenient and works well for testing the model. The Junit testing will not work with the view, however, so to test the view we will check each panel, and decide if it looks as anticipated. Many of the classes in the model and the controller are dependent of other classes. To test classes that have dependency’s, we require either a “mock” or “dummy” class. A mock or dummy class essentially just pretends to be the object that is required, to allow testing. Since we are developing software using test-driven development, the construction phase will be more streamline and time efficient leading to better designed code.