**IMPLEMENTATION**

We coded the speed control module of our automated commuter train system based on the specifications of the module outlined in our design document. We used the java language in Eclipse IDE. The two files are named Train.java and ControlSpeed.java. The source code for these files is in our first Appendix.

We created a Train object (Train.java) that contains all of the flags, speed variables, and speed control functions. This includes Phase I and II brake flags, the current status of the train, stop request flags, etc. These are all private variables that can only be changed by functions within the Train class.

Our user interface and testing is done in the ControlSpeed.java class. This is where all of the test cases are made. We created a separate Train object for each test case and then ran each test case. The results, such as any notifications and the current speed are printed to the screen. The current speed is printed to the screen each time it is updated with a minor delay so that the user can realistically see how the train speed would actually be affected.

**TESTING**

The purpose of our test cases is to assure that our ControlSpeed module works correctly in all situations as stated in our design document. The first test case checks for if the train’s current speed is below the minimum speed set for the train (in our test the minimum speed is 30 mph). The second test case checks for if the train’s current speed is above the maximum speed set for the train (in our test the maximum is 60 mph). The third test case checks for if a stop request has been received, but there is no location nearby to stop at. The fourth test also checks for a stop request, but this time when there IS a location nearby to stop at. The fifth test case checks for an emergency stop request, where the Phase I and II brakes will be applied. Finally, the sixth test checks for if there is an engine failure and creates an emergency stop request.