# 4. Environment Requirements

## 4.1 Application Requirements

The Plotter application requires that the machine it is running on has Java version 1.7 or later installed. No other environmental specifications are necessary for the program to run.

## 4.2 Test Environment Requirements

The Java testing framework TestNG has been selected as the testing framework that will be used for unit, functional, integration, and end-to-end tests. TestNG can be set up in an Eclipse IDE and easily facilitated from there. For perf and load testing, Apache JMeter has been selected as the preferred testing framework. JMeter comes with its own user interface where test set-up and executions can be facilitated. In order for both TestNG and JMeter tests to run successfully, the machine where the tests are hosted must have Java version 1.7 or later installed.

# 5. Test Schedule

* Get familiar with Plotter application 04/03/2017 – 04/09/2017
* Test framework analysis/selection 04/10/2017 – 04/16/2017
* Software plan and test case development 04/17/2017 – 04/23/2017
* Test implementation 04/23/2017 – 04/28/2017
* Test execution and analysis 04/28/2017 – 05/02/2017

# 6. Control Procedures

## 6.1 Reviews

After each phase of the test schedule (outlined in section 5), the team will review the product of that phase together to ensure all team members are on the same page, and to identify any gaps that may exist. Furthermore, the review is also a way for the team to brainstorm on any possible additions or improvements we can make in that phase of the test schedule. Reviews are scheduled to occur on a set day and time of the week (Sunday at 12 pm). Impromptu review meetings can also be scheduled in emergency cases.

## 6.2 Bug Review meetings

Bug review meetings will be held after a set of tests are run. The goal of these meetings is to review with the quality team the different defects that are discovered with each type of test, the priority and severity of the bug, and what action steps to take next. High priority bugs will be escalated immediately while low and medium priority bugs will be written up and reported out. This meeting will alsomhelp prevent duplicate defects from being written up.

## 6.3 Change Request

When tests are in the process of being run, changes should not be made to the application in the testing environment apart from high priority defects. High priority defects that come out of testing (that are identified as high priority in the bug review meeting) will be escalated to a management board. There, it will be determined whether the bug in question is high enough in severity where a fix needs to be pushed out to the testing environment (which would interrupt tests and possibly cause testing to have to start over).

## 6.4 Defect Reporting

Defects will be written up by the quality team as they are discovered and reviewed in bug review meetings. A defect should have the following information (this will benefit both developers in fixing the bug and quality in verifying the bug):

* Descriptive title
* Version where bug occurs
* Priority and severity (as discussed in the bug review meeting)
* Steps to reproduce (should come from the test case)
* Any supporting files that could be useful to the developer such as screenshots or an exported application file

The defect will also contain a status so that it can be tracked where the bug is in the validation process (not started, in progress, verify, and closed). When the developer begins work on the fix, it is their responsibility to update the status of the bug to in progress, and similarly put it in verify when it is ready to be re-tested. If applicable, the developer should also update the bug with the “fixed version” so the tester can be sure they are re-testing in the correct version. Once the tester has verified the fix, the bug will be moved to a closed status.

# 7. Functions To Be Tested

The following is a list of functions that will be tested:

* Plotting a linear function
* Plotting a quadratic function
* Plotting a polynomial function
* Draw button (updates the graph)
* Zooming in
* Zooming out
* Integral Panel
  + Changing x1 and x2 values
  + Trapezium
  + Simpson
  + Gauss
* Showing/hiding degree of freedom (correct for different types of functions)
* Position indicator (bottom right corner)
* Moving around the graph
  + Click and drag
  + Using arrows
* Visualization
  + Cartesian 2D
  + Polar 2D
  + Cartesian 3D
* Changing colors (test different components change to correct colors)
* Saving and/or exporting a graph