

## **Code Testing and Management**

The purpose of this document is to outline the policies used in the submission of code by members of this team to be integrated into the overall project codebase. These policies are intended to reduce the possibility of exposing the team to nonfunctional, buggy, or otherwise undesirable code.

### **Source management:**

All source code for the project should be stored on GitHub, where we have a dedicated repository for this project. (<https://github.com/BU-ECE-504-S14/RayTracer>)

The Master branch is to be used for the most polished and functional version of the code. It will be updated rarely, and should include only features that are sufficiently complete.

The Production branch is to be used to consolidate code changes introduced by individual team members into a unified codebase. It will be merged into the Master branch before deliverable deadlines, or otherwise at the discretion of the team.

Each team member will also have a remote branch, which they are responsible for keeping up to date with their code changes and the Production branch. Team members will submit code to the Production branch through pull requests, which must be approved by another member of the team before they can be carried out and the submitter's code is merged into Production.

### **Quality control:**

While the decision of whether to integrate code into the Production branch lies with the team member reviewing the pull request, the responsibility for ensuring that the code meets our standards of quality lies with the team member who wrote it.

The following general guidelines apply when creating and reviewing code:

#### **1. Code should be well-documented.**

- a. Java-doc comments on public methods, classes, and class variables
- b. Comments, where necessary, to explain complicated or unusual code

#### **2. Code should be as bug-free as possible.**

- a. Unit tests where possible, though unit testing graphics code can be extremely difficult.

#### **3. Code should be readable.**

- a. Descriptive variable names and sane formatting.

**4. The project directory should be kept tidy.**

- a. Assets are in the right place. (Libraries in the /lib directory, tests in /test...)

There will be exceptions to these rules, and it's up to the team member reviewing the code to make a reasonable decision and provide useful feedback in the event that the pull request is denied.