Rendering Invisibility

Report Name	Test Specification
Author (User Id)	Katherine Rose Farmer (krf)
Supervisor	Bernie Tiddeman (bpt)
Module	CS39440
Degree Scheme	G451 (Computer Graphics, Vision and Games inc. Industrial Placement)
Date	April 6, 2015
Revision	1.0
Status	Final

Introduction

Goals and objectives

The goal of this document is to inform about the test process that will surround my system for modelling invisibility. The current state of the system will be discussed as well as the design discussed in the design specification. The tests will take these into account. The objectives of the test process are to create a robust system as well as validating the success of the research project against the theory discussed in the outline project specification and design specification.

Statement of scope

The scope of the testing will include the overall system, the models, the UI of the webpages. The scope of the validation will include the models only. Any behaviour that is specific to the theories and the models will not be tested, it will only be validated through experiments.

Major constraints

Due to the highly graphical nature of the project, there will be minimum unit testing and very little automated testing, if any at all.

Test Plan

Software to be tested

The software to be tested is identified by name. Exclusions are noted explicitly

Two websites will be tested. One can be found at the site: http://krf12.github.io which is the project diary. The other can be found at the site: http://krf12.github.io/RenderingInvisibility

Testing strategy

Unit testing

Components to be tested

Singular JavaScript files such as index.js

Tests required

JavaScript files needs to compile correctly and complete the functions required by the system. This will include drawing spheres correctly, loading shaders in correctly, shaders producing expected

effects etc. As there is some overlap between models and shaders, I am expecting to only test any unique functions and also test one file as a basis for the rest. If this file's functions work correctly, I will assume that any other models similar to it will work as well. An example would be testing only the shell sphere and expecting that – due to the shaders being identical to the shell cylinder example – that the other shell models won't need testing as well.

Expected results

Each file will produce the expected results, based on the validation testing as well as any effects and methods I have entered into the files that are unique to that file.

System testing

Components to be tested

The two complete websites - : http://krf12.github.io and http://krf12.github.io/RenderingInvisibility.

Tests required

The systems will require being navigated and ensuring that the system navigates correctly from page to page. There will also need to be checking on the project diary that ensures the dates are correct and there are no overlaps. The models on the project site will need to be checked for correct titles and navigation between models.

There will also be an expectation that the pages will load correctly. There may be some performance testing to record which pages are loading the slowest.

Expected results

That both systems will have correct navigation, perform within reasonable expectations, due to the process-heavy models being created.

Validation testing

Components to be tested

Each of the models will be tested individually.

Tests required

There is a need to compile a list of effects that the system is expected to be able to emulate and then test based on these effects. As of this version, the system is able to be tested on chromatic effects and perfect invisibility.

Expected results

The effects are created as expected, or that the results are understood as to the failures of the tests.

2.5 Test record keeping

The results for tests will be documented in the final report for this project and will likely be included as an appendix with important results highlighted in the relevant section.

The results of unit and system tests will likely be kept in table format. The results of the validation testing will be recorded the same way science experiments are recorded, using the test, expected results, actual results and conclusion structure.

2.6 Test metrics

There will be checking for the amount of tests that pass as expected and a limit of 75% pass rate for the project. If the first round of testing produces lower than this then another round of testing after defect fixing will be needed.

2.7 Testing tools and environment

The tests will be carried out on multiple browsers, with chrome and firefox being the main browsers due to their extensive support of WebGL. There will be minimal testing on tablet and smartphone environments as the hardware requirements for the models are possibly out of range for most smartphones and tablets.

There will likely only be manual testing but I will look into using an FPS meter to check the performance of each of the models.