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 | March 19, 2015 |
| Revision | 0.1 |
| Status | Draft |

# 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.

#### 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

#### Tests required

#### Expected results

### Validation testing

#### Components to be tested

#### Tests required

#### Expected results

## 2.5 Test record keeping

Mechanisms for storing and evaluating test results are specified.

## 2.6 Test metrics

A description of all test metrics to be used during the testing activity is noted here.

## 2.7 Testing tools and environment

A description of the test environment, including tools, simulators, specialized hardware, test files, and other resources is presented here.

# Test Procedure

This section describes as detailed test procedure including test tactics and test cases for the software.

## 3.1 Software (SCIís) to be tested

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

## 3.2 Testing procedure

The overall procedure for software testing is described.

### Unit test cases

The procedure for unit testing is described for each software component (that will be unit tested) is presented. This section is repeated for all components i.

### Integration testing

The integration testing procedure is specified.

### Validation testing

The validation testing procedure is specified.

### High-order testing (a.k.a. System Testing)

The high-order testing procedure is specified. For each of the high order tests specified below, the test procedure, test cases, purpose, specialized requirements and pass/fail criteria are specified. It should be noted that not all high-order test methods noted in Sections 3.2.4.n will be conducted for every project.

## 3.5 Test record keeping and test log

Mechanisms for storing and evaluating test results are specified. The test log is used to maintain a chronological record of all tests and their results.