SE 3XA3: Test Plan
Rogue Reborn

Group #6, Team Rogue++

Ian Prins	prinsij
Mikhail Andrenkov	andrem5
Or Almog	almogo

Due Wednesday, Dec 7$^{\text{st}}$, 2016

Contents

List of Tables

List of Figures

Table 1: **Revision History**

Date	Version	Notes
Dec 6	0.1	Initial draft

This document...

1 Functional Requirements Evaluation

Ori

2 Nonfunctional Requirements Evaluation

Mikhail

2.1 Usability

Mikhail

2.2 Performance

Mikhail

2.3 etc.

Mikhail

3 Comparison to Existing Implementation

Ori

4 Unit Testing

Mikhail

5 Changes Due to Testing

Mikhail

6 Automated Testing

6.1 Automated Testing Strategy

For this project we elected not to use a 3rd party testing library. We made this decision to ease configuration/installation problems and reduce our dependencies, as we judged it would not be necessary. Instead a series of files (labeled test.foobar.cpp) in the repository hold tests, which are run by our custom test runner. These automated tests are run on command by executing the produced executable, or by the continuous integration script run whenever changes are pushed to the central repository. The results of these tests are automatically reported, resulting in a failed or successful build.

6.2 Specific System Tests

The following is a list of all system tests in the project.

Name:	
Initial State:	
Input:	
Expected Output:	

Name:	Amulet Construction
Initial State:	None
Input:	Coordinate, context value
Expected Output:	Amulet object in valid initial state
Name:	Armor Construction 1
Initial State:	None
Input:	Coordinate
Expected Output:	Armor object in valid initial state
Name:	Armor Construction 2
Initial State:	None
Input:	Coordinate, context value, type value
Expected Output:	Armor object in valid initial state
Name:	Armor Identification
Initial State:	Cursed Armor
Input:	None
Expected Output:	Verification that armor is identified

Name:	Armor Identification
Initial State:	Cursed Armor
Input:	None
Expected Output:	Verification that armor is identified

7 Trace to Requirements

Ori

8 Trace to Modules

Ori

9 Code Coverage Metrics

Ori