

Functional Test Plan  
Death from Aside  
Submitted to EECS Software Engineering  
393  
Version 0.1

Rebecca Haluska, Ian Lavelle, Evangeline Chandran,  
Aaron Cutright

Submitted November 7, 2016

## SDD Revision History

Date	Version	Description	Author
11/4/16	0.1	Document skeleton & formatting, ToC, section descriptions, Items to be Tested/Not Tested	Rebecca Haluska
11/5/16	0.2	Skeletal changes, table of contents	Aaron Cutright
11/5/16	0.3	Functional Basic, Gameplay additions	Rebecca Haluska, Ian Lavelle, Aaron Cutright
11/5/16	0.4	UI Testing additions	Ian Lavelle, Rebecca Haluska
11/5/16	0.5	Interaction testing	Aaron Cutright, Rebecca Haluska
11/5/16	0.6	Test Risks/Issues	Evangeline Chandran
11/6/16	0.7	Numbering system for all Functional Requirements	Evangeline Chandran
11/5/16	1.1	Inspections	ALL

# Table of Contents

## **Introduction**

Purpose of the Test Plan Document

## **Functional Testing**

Test Risks/Issues

Items to be Tested/Not Tested

Functional Basic Requirements Testing

Functional Gameplay Requirements Testing

Functional UI Requirements Testing

Functional Interactions Requirements Testing

## **Inspection Reports**

# Introduction

## Purpose of the Test Plan Document

The Test Plan Document exists to provide definitions for the tests that we will be conducting in order to guarantee conformance to our software requirements specification and design document. This document provides a listing of all tests and descriptions of what they will consist of which will allow us to verify that our software has sufficient coverage and robustness.

## Functional Testing

Functional testing involves tests which quantify the extent to which the program fulfills its requirements functions and conforms to specifications such as the requirements documentation. Such tests include feature validation, ensurance of GUI design, presence of gameplay aspects, etc.

## Test Risks/Issues

Risks:

- As with all testing, there is a time cost associated with performing this suite, especially in a “regression test” scenario involving the full suite. Care must be taken not to over- or under-test.
- There is some danger of restricting the definition of requirement fulfillment to simply passing associated tests, or, similarly, failing to account for some aspects of requirement fulfillment in tests. This can partially be alleviated by document inspection.
- Because performing some of these tests will require modifications to the spawning rules that govern the game, there is a risk that such modifications may accidentally become incorporated into the production build. Symptoms of such include:
  - Enemies/platform do not spawn completely/dynamically
  - Enemies do not attack player
  - Player cannot attack enemies
  - Player cannot cross platforms

## Items to be Tested/Not Tested

Because the majority of the scope of our project is theoretically captured within the Software Requirements Specification, there is very little not covered within the scope of our functional tests. Those portions lying without it include: compatibility with other operating systems (not a requirement), function within a live system (not explicitly tested, but implicitly tested due to our

testing system), and compatibility/best practices within Unity, as well as non-functional criteria such as “80% of players will open and play our game more than once.”

## Functional Basic Requirements Testing

No.	Requirement Tested	Test Case Description	Fulfilled (Y/N)
BT-1	Functional, Basic Requirements - 1	Verify that, when the player starts the game, the player, monster, and initial platforms are all spawned	Y
BT-2	Functional, Basic Requirements - 2	Verify that, when the game is started, the monster entity begins moving right towards the player	Y
BT-3	Functional, Basic Requirements - 3	Verify that, upon player death, a score is displayed that is representative of how far they made it into the level and how many enemies they killed	Y
BT-4	Functional, Basic Requirements - 4	Verify that, when the player is hit by an enemy projectile or other hazard, the player's health decrements	Y
BT-5	Functional, Basic Requirements - 5	Verify that when the player's health reaches 0, the game ends and a score is displayed	Y
BT-6	Functional, Basic Requirements - 6, 7	Verify that pressing W, A, and D on a keyboard makes the player jump, move left, and move right, respectively	Y
BT-7	Functional, Basic Requirements - 8	Verify that moving the mouse will move a visible aiming reticle	Y
BT-8	Functional, Basic Requirements - 8	Verify that clicking the “fire” button will cause a projectile to move towards the player's reticle	Y
BT-9	Functional, Basic Requirements - 9	Verify that the player can land on a platform without falling through it	Y
BT-10	Functional, Basic Requirements - 10	Verify that platforms continue to spawn as the player moves right	Y
BT-11	Functional, Basic Requirements - 10	Verify that all platforms are reachable by the player	Y
BT-12	Functional, Basic	Verify that enemies spawn as the player	Y

	Requirements - 11	proceeds	
BT-13	Functional, Basic Requirements - 11	Verify that enemies will fire at or otherwise try to hinder the player	Y
BT-14	Functional, Basic Requirements - 12	Verify that a variety of enemies spawn as the player proceeds.	Y
BT-15	Functional, Basic Requirements - 13	Verify that a variety of items and weapons spawn as the player proceeds	Y
BT-16	Functional, Basic Requirements - 13	Verify that items and weapons can be picked up	Y
BT-17	Functional, Basic Requirements - 13	Verify that consumable items can be used	Y
BT-18	Functional, Basic Requirements - 13	Verify that immediate effect items take effect once the player touches them	Y
BT-19	Functional, Basic Requirements - 14	Verify that more enemies spawn as the player progresses and time increases	Y
BT-20	Functional, Basic Requirements - 14	Verify that enemy difficulty increases as the player progresses and time increases	Y

## Functional Gameplay Requirements Testing

No.	Requirement Tested	Test Case Description	Fulfilled (Y/N)
GT-1	Functional, Gameplay Requirements - 1	Verify that items spawned are a mix of consumable items, passive items, and weapons.	Y
GT-2	Functional, Gameplay Requirements - 1	Verify that item spawning only occurs to the right of the visible area, and falls both within the visible area and to the left/above/below the visible area.	Y
GT-3	Functional, Gameplay Requirements - 2	Verify that enemy spawning only occurs to the right of the visible area, and falls both within the visible area and to the left/above/below the visible area.	Y

GT-4	Functional, Gameplay Requirements - 3	Verify that each item has an associated graphical representation on the player once picked up.	Y
GT-5	Functional, Gameplay Requirements - 4	Verify that picked up consumable items give their corresponding benefit/detriment when used, and can only be used once.	Y
GT-6	Functional, Gameplay Requirements - 4	Verify that consumable items are correctly triggered by the designated "use" button.	N
GT-7	Functional, Gameplay Requirements - 5	Verify that passive items give a benefit/detriment to the player permanently when picked up, and do not need to be used.	Y
GT-8	Functional, Gameplay Requirements - 5	Verify that passive item effects do not persist after a player's death into the next game.	Y
GT-9	Functional, Gameplay Requirements - 6	Verify that each weapon produces a damaging particle. Note that, ex, a "sword blade" does count as a particle, as particles are as recognized by the engine.	Y
GT-10	Functional, Gameplay Requirements - 7	Verify that certain weapon combinations, when picked up, can combine into a more powerful weapon, and that this new weapon replaces the old ones.	Y
GT-11	Functional, Gameplay Requirements - 8	Verify that total weapons firing are equal to total weapons, ex: counting 2 combined weapons as 1 weapon total, 2 uncombined as 2.	Y
GT-12	Functional, Gameplay Requirements - 9	Verify that weapons can be fired either by a mouse click, or automatically, depending on the weapon being used.	Y
GT-13	Functional, Gameplay Requirements - 10	Verify that weapon bullet causes damage to enemies and/or platforms on collisions as expected.	Y
GT-14	Functional, Gameplay Requirements - 11	Verify that bullet collision with Monster applies damage, and that damage is represented as proportional slowdown to Monster.	Y
GT-15	Functional, Gameplay Requirements - 11	Verify that slowdown applied to monster does not result in backwards motion, and can be recovered with time (ex: speed	Y

		60->55->56...60.	
GT-16	Functional, Gameplay Requirements - 12	Verify that if the player touches the chasing monster, the player immediately dies and the game ends.	Y
GT-17	Functional, Gameplay Requirements - 13	Verify that collisions trigger decrease in player health proportional to collision damage (some bullets cause more damage than others).	Y
GT-18	Functional, Gameplay Requirements - 14	Verify that some enemies can cause terrain modifying effects, eg a patch of fire or ice	Y
GT-19	Functional, Gameplay Requirements - 15	Verify that platform effects are correctly applied to player, eg icy platforms make the player retain momentum longer.	Y
GT-20	Functional, Gameplay Requirements - 16	Verify that all created entities are removed from the game as the player proceeds further into the level.	Y

## Functional UI Requirements Testing

No.	Requirement Tested	Test Case Description	Fulfilled (Y/N)
UT-1	Functional, UI Requirements - 1	Verify that main screen loads on game start	Y
UT-2	Functional, UI Requirements - 1	Verify that all buttons on main screen work correctly	Y
UT-3	Functional, UI Requirements - 2	Verify the About screen displays information about the game's license, developers, and version	Y
UT-4	Functional, UI Requirements - 2	Verify that the player can return to the main menu from the About screen	Y
UT-5	Functional, UI Requirements - 3	Verify that gameplay screen consists of the previously-defined gameplay arena	Y
UT-6	Functional, UI Requirements - 4	Verify that "Game Over" screen loads at player death.	Y
UT-7	Functional, UI	Verify that all buttons on "Game Over" screen work	Y



	Requirements - 4	correctly	
UT-8	Functional, UI Requirements - 4	Verify that "Game Over" screen displays accurate final score	Y
UT-9	Functional, UI Requirements - 5	Verify that the player's score is incremented by the distance traveled into the level and number and type of enemies killed	Y
UT-10	Functional, UI Requirements - 6	Verify that health counter is present and visible in gameplay screen	Y
UT-11	Functional, UI Requirements - 6	Verify that health counter correctly decrements and increments on damage and heal, respectively	Y
UT-12	Functional, UI Requirements - 7	Verify that player scores are saved to score file on game over	Y
UT-13	Functional, UI Requirements - 7	Verify that scores saved to score file correctly represent player score	Y
UT-14	Functional, UI Requirements - 7	Verify that score file is used to correctly populate high score screen	Y
UT-15	Functional, UI Requirements - 8	Verify that, when a weapon is merged or an item is picked up, a pop up dialogue is displayed giving the player information on the item/merge, and that this dialogue disappears after a short time	Y
UT-16	Functional, UI Requirements - 9	Verify that all status effects have a corresponding icon	Y

## Functional Interaction Requirements Testing

No.	Design Element to be Tested	Test Description	Fulfilled (Y/N)
IT-1	Weapon-weapon Interaction	Verify that weapons that can merge together can successfully merge	Y
IT-2	Weapon-weapon Interaction	Verify that weapon merges follow and respect both order and orientation--that is, two guns that can merge, but not with each other, cannot merge. Three guns that can merge, two into one gun and one into the product, will all merge together to form the final,	Y

		three-component gun	
IT-3	Weapon-enemy Interaction	Verify that projectiles interact with enemies by applying damage and status effects as prescribed by bullet characteristics	Y
IT-4	Weapon-enemy Interaction	Verify that, if enemies express resistances, resistances are respected by weapon-enemy	Y
IT-5	Weapon-terrain Interaction	Verify that, if a projectile has an effect that affects platforms, that effect is applied on collision	Y
IT-6	Weapon-terrain Interaction	Verify that, unless prohibited by projectile effect, projectiles block bullets	Y
IT-7	Enemy-enemy Interaction	Verify that the Monster “destroys” enemies (this is trivially covered by deletion of offscreen enemies)	Y
IT-8	Enemy-terrain Interaction	Verify that enemies with platform-affecting abilities correctly apply these abilities to associated platforms	Y
IT-9	Enemy-terrain Interaction	Verify that platform effects are correctly applied to enemies, respecting immunities	Y
IT-10	Terrain-terrain Interaction	Verify that platform clipping, obstruction, and unreachability is minimized (not absent, but game should be playable)	Y

## Inspection Reports

Inspector Name:	Rebecca Haluska
Inspection Date:	November 5, 2016
Inspection Length:	1 hour
Correction Length:	1 hour

Location	Description	Status
Test Risks/Issues	Formatting issue: incorrect nesting of “symptoms”	Fixed
Test Risks/Issues	Missing risk cases for modification of parameters,	Fixed

	"feature creep"	
Functional Interaction Testing	Test cases are too broad, require being split into sub-cases	Fixed
Functional Gameplay Testing	Some test cases are too broad, require being split into sub-cases	Fixed
GT-2	Wording does not fully capture requirements	Fixed
Functional Gameplay Testing	Duplicated test identifier	Fixed
Functional Interaction Testing	Improper capitalization of some "Design Element to be Tested" fields	Fixed

Inspector Name:	Evangeline Chandran
Inspection Date:	November 5, 2016
Inspection Length:	1 hour
Correction Length:	1 hour

Location	Description	Status
Functional Basic Requirements Testing	Terminology vague	Fixed
All Requirements Testing	Lacking a proper numbering system	Fixed
Gameplay Requirements Testing	Confusing wording	Fixed
Functional Basic Requirements Testing	Punctuation/wording inconsistencies	Fixed
Gameplay Requirements Testing	Punctuation/wording inconsistencies	Fixed

Inspector Name:	Ian Lavelle
-----------------	-------------

Inspection Date:	November 5, 2016
Inspection Length:	1 hour
Correction Length:	1 hour

Location	Description	Status
Functional Basic Requirements Testing	The description of the score is inconsistent with the requirements specification	Fixed
Functional Gameplay Requirements Testing	The description of an item's representation when picked up is somewhat unclear	Fixed
Functional Basic Requirements Testing	Coverage of Basic Requirement 1 is missing	Fixed
All Requirements Testing	Now that test identifiers have been added, the letters next to each requirement are redundant and should be removed	Fixed

Inspector Name:	Aaron Cutright
Inspection Date:	November 5, 2016
Inspection Length:	1 hour
Correction Length:	1 hour

Location	Description	Status
Functional Basic, Functional Gameplay	Some tests needed to be split into parts so that they could be independently verified	Fixed
Functional Basic	Miscellaneous grammar inconsistencies	Fixed
Functional Basic	Needed identifiers column ("No.")	Fixed
Functional Gameplay	Grammar inconsistencies	Fixed

