Physics-Based Chipmunk2D Game

Software Requirements Specification

Based on the Volere Template

Steven Palmer
Emaad Fazal
Chao Ye

October 12, 2015

Contents

1 Project Drivers 1
1.1 The Purpose of the Project 1
1.2 The Stakeholders 1
1.2.1 The Client 1
1.2.2 The Customers 1
1.2.3 Other Stakeholders 1

2 Project Description 2
2.1 Game Overview 2
2.1.1 The Game World 2
2.1.2 The Hero 2
2.1.3 The Enemies 3
2.2 Mandated Constraints 4
2.3 Naming Conventions and Terminology 4

3 Functional Requirements 5
3.1 The Scope of the Work and the Product 5
3.1.1 The Context of the Work 5
3.1.2 Work Partitioning 6
3.1.3 Individual Product Use Cases 6
3.2 Functional Requirements 12

4 Non-functional Requirements 24
4.1 Look and Feel Requirements 24
4.2 Usability and Humanity Requirements 24
4.3 Performance Requirements 24
4.4 Operational and Environmental Requirements 25
4.5 Maintainability and Support Requirements 25
4.6 Security Requirements 26
4.7 Cultural Requirements 26
4.8 Legal Requirements 26

5 Project Issues 27
5.1 Open Issues 27
5.2 Off-the-Shelf Solutions 27

5.3 New Problems 27
5.4 Tasks 27
5.5 Migration to the New Product 27
5.6 Risks 28
5.7 Costs 28
5.8 User Documentation and Training 28
5.9 Waiting Room 28
5.10 Ideas for Solutions 28

List of Tables

1 List of terminology 5
2 List of events 8
3 List of project phases 27

List of Figures

1 Context diagram of the work 5
2 Flow diagram of the game 7

Revision History

Date	Version	Notes
October 7, 2015	1.0	Created document
October 7, 2015	1.1	Major edits in progress
October 8, 2015	1.2	Major event and reqs additions
October 9, 2015	1.3	Final version for rev 0 hand-in

1 Project Drivers

1.1 The Purpose of the Project

The purpose of this project is to produce a game that will be used as a demonstration for students in a third year software engineering game design course at McMaster University. The game will incorporate the Chipmunk2D physics library and highlight its capabilities.

1.2 The Stakeholders

1.2.1 The Client

The client for this project is Dr. Spencer Smith of the Computing and Software department at McMaster University.

1.2.2 The Customers

The customers for this project are students who will take the game design course in the future.

1.2.3 Other Stakeholders

Other stakeholders include future instructors of the game design course or other similar courses.

2 Project Description

2.1 Game Overview

For this project an action-role playing type game [don't say "type" unless you're following it with "of" —DS] will be created. It will consist of a game world within which a user-controlled hero makes progress by defeating enemies. The subsections that follow provide a more detailed explanation of the game.

2.1.1 The Game World

The game world is the virtual environment in which all gameplay takes place. This environment is made up of platforms which the hero and enemies are permitted to stand on, as well as obstacles which limit the possible movements of the hero throughout the game and hazards which can cause damage or unwanted effects to both the hero and enemies.

[Sounds very generic. You should be trying to set your game world apart. Is there a story? Theme? Some stylistic elements that are constrained? —DS]

2.1.2 The Hero

The hero is the protagonist of the game, and is controlled by the user.

2.1.2.1 Movement

The hero is able to move left or right, and to jump, in order to progress through the game. The hero can interact with several objects in the game. These objects include enemies, obstacles, and hazards. When the hero comes into contact with an object an event is triggered. Depending on the type of object, these events include:

1. If the object is an enemy the hero will lose health and be knocked back from the enemy.
2. If the object is an obstacle the hero will be stopped and unable to pass.
3. If the object is an environmental hazard the hero may lose health and be knocked back depending on the type of hazard.

2.1.2.2 Attack

The hero is able to attack enemies using weapons. Weapons are defined by their fire rate, power, bullet travel distance, and ammo capacity. These characteristics provide certain weapons with advantages over others depending on the type of enemy being fought.

Weapons are divided into three classes: pistols, shotguns, and rifles. Each class of weapons has its own benefits and faults. Pistols are weak in terms of their power but have infinite ammo and are always available for the player to use. They provide a way for the user to conserve ammo in dire situations. They also provide a last ditch effort when no ammo is available for the other two weapon classes. Shotguns are very powerful but have very short range and limited ammo capacity. They provide an efficient way to deal with a large amount of enemies at once. Rifles provide good range to hit enemies from a distance, and tend to be more powerful than pistols but weaker than shotguns.

2.1.2.3 Score

As the hero defeats enemies, his or her score is increased. The amount by which the hero's score is increased depends on how powerful the defeated enemy was. Increasing the hero's score offers access to more powerful weapons and allows navigation into deeper areas of the game world where more powerful enemies lurk.

2.1.3 The Enemies

Enemies are found throughout the game and are programmed to attack the hero with the goal of defeating him or her. When an enemy is encountered during the game, combat will ensue if the hero gets within a certain range. The user may attempt to avoid the enemy or to flee if he or she has come close enough to trigger combat.

2.1.3.1 Movement

[Even though you describe "AI" in your naming conventions, you should still clarify what it means the first time it is seen prior to that section. —DS]

Enemy movement is controlled by the game AI. When the hero is within a certain range of an enemy, the game AI will respond by moving the enemy in a manner defined in the game code.

2.1.3.2 Attack

Enemy attacks are also controlled by the game AI. When the hero is within a certain range of an enemy, the game AI will respond by attacking in a manner defined in the game code. Unlike the hero, the enemies do not use weapons, but instead have inherent attacks which vary by enemy type.

2.2 Mandated Constraints

The project is subject to the following mandated constraints:

[Include why these constraints are mandated. In this case, because of your client. —DS]

1. The game must make significant use of the Chipmunk2D physics library.
2. The game must support all major PC operating systems.
3. Project milestones must be completed by the dates given in the CS 4ZP6 syllabus.
4. The project must be fully completed by April 1, 2015.

2.3 Naming Conventions and Terminology

The terminology used in this project is given in Table 1.

Table 1: List of terminology

Term	Definition
AI	Artificial intelligence
Bounds	The boundaries inside which game play occurs
Enemy	Hostile character; attacks hero if hero is in range
Hazard	An environmental object that causes negative effects to the hero
Hero	The main character of the game controlled by the user
Hit Points	The amount of damage the hero or an enemy can take before being defeated
Obstacle	A barrier that the hero cannot cross
Score	Measure of progress in the game
User	Player of the game

3 Functional Requirements

3.1 The Scope of the Work and the Product

3.1.1 The Context of the Work

A context diagram of the the work is given in Figure 1.

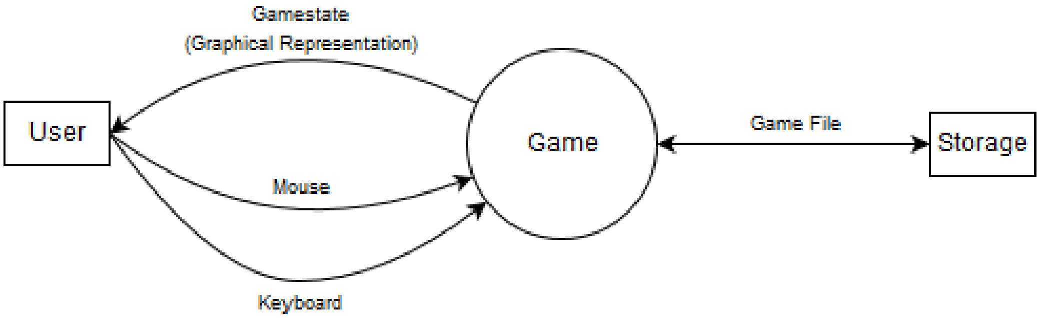

Figure 1: Context diagram of the work

[You have no constraints on the input type. This should be reflected in the diagram —DS]

3.1.2 Work Partitioning

The flow diagram in Figure 2 gives a rough representation of the operation of the envisioned game. The user interfaces include a main menu as well as an in-game menu, and an in-game interface in which all game play takes place. The events are listed in Table 2.

3.1.3 Individual Product Use Cases

Due to the nature of the project, the product use cases are essentially equivalent to the events identified in the work partitioning.

Use Case #: 1

Name:	New Game
Trigger:	The user selects to start a new game
Precondition:	The main menu is open
Postcondition:	A new game commences

Use Case #: 2

Name:	Load Game
Trigger:	The user selects to load a game
Precondition:	The main menu or in-game menu is open
Postcondition:	A saved game state is loaded and the game commences from that point

Use Case #: 3

Name:	Save Game
Trigger:	The user selects to save a game
Precondition:	The in-game menu is open
Postcondition:	A saved game state is created

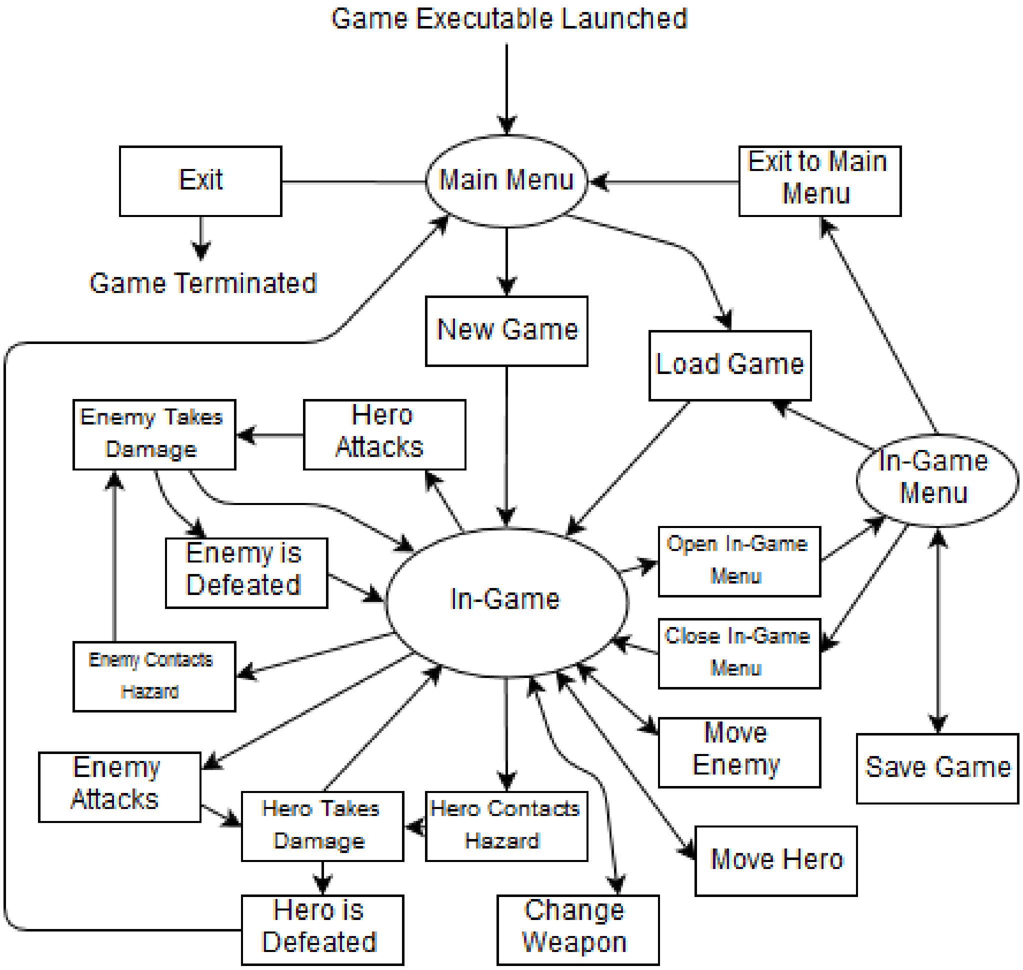

Figure 2: Flow diagram of the game. Ovals represent user interfaces and rectangles represent events.

Table 2: List of events

	Event Name	Inputs/Outputs	Summary
1.	New Game	Mouse (in) [Not necessarily mouse —DS]	A new game is started
2.	Load Game	Game File (in)	A game file is loaded
3.	Save Game	Game File (out)	A saved game file is created
4.	Exit to Main Menu	Mouse (in)	Exit from current game to main menu
5.	Move Hero	Keyboard (in) Gamestate (out)	The hero moves through the game world
6.	Hero Attack	Keyboard (in) Gamestate (out)	The hero attacks an enemy
7.	Change Weapon	Keyboard (in) Gamestate (out)	The hero's current weapon is changed
8.	Hero Contacts Hazard	Gamestate (out)	The hero comes into contact with a hazard
9.	Hero Takes Damage	Gamestate (out)	The hero loses hit points
10.	Hero is Defeated	Gamestate (out)	The hero's hit points reach zero and the game ends
11.	Move Enemy	Gamestate (out)	An enemy moves through the game world
12.	Enemy Attack	Gamestate (out)	An enemy attacks the hero
13.	Enemy Contacts Hazard	Gamestate (out)	An enemy comes into contact with a hazard
14.	Enemy Takes Damage	Gamestate (out)	An enemy loses hit points
15.	Enemy is Defeated	Gamestate (out)	An enemy's hit points reach zero
16.	Open In-Game Menu	Keyboard (in)	The in-game menu is opened
17.	Close In-Game Menu	Mouse (in)	The in-game menu is closed
18.	Exit Game	Mouse (in)	The game application is terminated

Use Case #: 4

Name:	Exit to Main Menu
Trigger:	The user selects exit game
Precondition:	The in-game menu is open
Postcondition:	Current game is ended and main menu is opened

Use Case #: 5

Name:	Move Hero
Trigger:	Inputs from user related to controlling the hero movement
Precondition:	In-game
Postcondition:	Hero moves according to input

Use Case #: 6

Name:	Hero Attack
Trigger:	Inputs from user related to hero attack
Precondition:	In-game
Postcondition:	Hero's currently selected attack is activated

Use Case #: 7

Name:	Change Weapon
Trigger:	Input from user related to hero weapon (hotkeys)
Precondition:	In-game
Postcondition:	Hero's currently active weapon is switched according to input

Use Case #: 8

Name:	Hero Contacts Hazard
Trigger:	Hero comes into contact with hazard
Precondition:	In-game
Postcondition:	Hero is affected by the hazard

Use Case #: 9

Name:	Hero Takes Damage
Trigger:	Enemy contacts hazard or enemy is attacked by hero
Precondition:	In-game
Postcondition:	Enemy hit points are reduced

Use Case #: 10

Name:	Hero is Defeated
Trigger:	Hero hit points reach zero
Precondition:	In-game
Postcondition:	Current game is ended and main menu is opened

Use Case #: 11

Name:	Move Enemy
Trigger:	Hero comes within specific distance of enemy
Precondition:	In-game
Postcondition:	Enemy moves according to game AI

Use Case #: 12

Name:	Enemy Attack
Trigger:	Hero comes within specific distance of enemy
Precondition:	In-game
Postcondition:	Enemy attack is activated according to game AI

Use Case #: 13

Name:	Enemy Contacts Hazard
Trigger:	Enemy comes into contact with hazard
Precondition:	In-game
Postcondition:	Enemy is damaged by the hazard

Use Case #: 14

Name:	Enemy Takes Damage
Trigger:	Enemy contacts hazard or enemy is attacked by hero
Precondition:	In-game
Postcondition:	Enemy hit points are reduced

Use Case #: 15

Name:	Enemy is Defeated
Trigger:	Enemy hit points reach zero
Precondition:	In-game
Postcondition:	Enemy is removed from the game

Use Case #: 16

Name:	Open In-game Menu
Trigger:	User input (hotkey)
Precondition:	In-game
Postcondition:	The in-game menu is opened

Use Case #: 17

Name:	Close In-game Menu
Trigger:	User selects close menu option
Precondition:	In-game menu is open
Postcondition:	In-game

Use Case #: 18

Name:	Exit Game
Trigger:	The user selects exit game
Precondition:	The main menu is open
Postcondition:	The application is terminated

[What about upgrading the hero? Completing the game? etc. —DS]

3.2 Functional Requirements

Requirement #: 1 **Requirement Type:** 3.2 **Use Case:** 1

Description:	The user shall have the ability to start a new game
Rationale:	The user must be able to start a new game in order to play the game
Fit Criterion:	A new game is able to be started

Cust. Satisfaction:	1	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials:	None
History:	Created October 7, 2015

Requirement #: 2 **Requirement Type:** 3.2 **Use Case:** 2

Description: The user shall have the ability to load a saved game state
Rationale: The user must be able to load his or her saved progress to continue the game
Fit Criterion: Saved game state is successfully loaded

Cust. Satisfaction:	1	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 7, 2015

Requirement #: 3 **Requirement Type:** 3.2 **Use Case:** 3

Description: The user shall have the ability to save the current game state
Rationale: The user must be able to save his or her progress
Fit Criterion: Game state is successfully saved

Cust. Satisfaction:	1	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 8, 2015

Requirement #: 4 **Requirement Type:** 3.2 **Use Case:** 4

Description: The user shall have the ability to exit the current game
Rationale: The user requires a method of quitting a game in progress
Fit Criterion: User is able to exit the current game and return to the main menu

Cust. Satisfaction:	1	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 9, 2015

Requirement #: 5 **Requirement Type:** 3.2 **Use Case:** 5

Description: The user shall be able to move the hero to the left and right
Rationale: The hero must be able to be moved left and right to navigate the game world
Fit Criterion: The hero moves left and right correctly based on specific user inputs

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 8, 2015

Requirement #: 6 **Requirement Type:** 3.2 **Use Case:** 5

Description: The user shall be able to make the hero jump
Rationale: The hero must be able to jump to reach the intended areas of the game world
Fit Criterion: The hero is able to jump based on a specific user input

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 8, 2015

Requirement #: 7 **Requirement Type:** 3.2 **Use Case:** 5

Description: The hero shall be subject to the laws of physics
Rationale: The game world's laws of physics should apply to the hero
Fit Criterion: The hero's movement responds appropriately to the laws of physics

Cust. Satisfaction:	5	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 9, 2015

Requirement #: 8 **Requirement Type:** 3.2 **Use Case:** 5

Description: The hero shall remain in bounds
Rationale: The hero must remain in the intended boundaries of play for the game to function properly
Fit Criterion: Hero is unable to pass through walls and other obstacles

Cust. Satisfaction:	2	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 7, 2015

Requirement #: 9 **Requirement Type:** 3.2 **Use Case:** 5

Description: All intended areas of the game shall be reachable
Rationale: All areas of the game where the hero is intended to be should be reachable
Fit Criterion: All areas reachable when testing

Cust. Satisfaction:	2	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 7, 2015

Requirement #: 10 **Requirement Type:** 3.2 **Use Case:** 6

Description: The hero shall be able to successfully carry out attacks on enemies
Rationale: The hero must be able to carry out attacks to damage and defeat enemies
Fit Criterion: Attack action occurs based on a particular user input

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 8, 2015

Requirement #: 11 **Requirement Type:** 3.2 **Use Case:** 7

Description: The user shall have access to all weapons available to the hero
Rationale: All available weapons should be accessible
Fit Criterion: Each weapon is accessible by a specific user input (hotkey)

Cust. Satisfaction:	5	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 8, 2015

Requirement #: 12 **Requirement Type:** 3.2 **Use Case:** 9

Description: The hero's hit points shall be reduced when he/she takes damage
Rationale: The hero must be able to be damaged
Fit Criterion: The hero's hit points are reduced upon taking damage

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 9, 2015

Requirement #: 13 **Requirement Type:** 3.2 **Use Case:** 9, 8

Description: The hero shall be knocked back when taking damage from a hazard
Rationale: Hazards causing damage are intended to cause knock-back
Fit Criterion: The hero is knocked back when taking damage from a hazard

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 9, 2015

Requirement #: 14 **Requirement Type:** 3.2 **Use Case:** 10

Description: A game over message followed by a return to the main menu shall occur when the hero is defeated
Rationale: The game is over when the hero is defeated
Fit Criterion: Game displays game over message and returns to main menu upon hero defeat

Cust. Satisfaction: 1 **Cust. Dissatisfaction:** 3
Priority: High **Conflicts:** None

Supporting Materials: None
History: Created October 9, 2015

Requirement #: 15 **Requirement Type:** 3.2 **Use Case:** 11

Description: The enemy shall be able to move
Rationale: The enemy must be able to move in combat against the hero
Fit Criterion: Enemy movement occurs when hero comes within a set distance

Cust. Satisfaction: 5 **Cust. Dissatisfaction:** 5
Priority: High **Conflicts:** None

Supporting Materials: None
History: Created October 9, 2015

Requirement #: 16 **Requirement Type:** 3.2 **Use Case:** 11

Description: The enemy shall be subject to the laws of physics
Rationale: The game world's laws of physics should apply to the enemy
Fit Criterion: The enemy's movement responds appropriately to the laws of physics

Cust. Satisfaction: 5 **Cust. Dissatisfaction:** 5
Priority: High **Conflicts:** None

Supporting Materials: None
History: Created October 9, 2015

Requirement #: 17 **Requirement Type:** 3.2 **Use Case:** 11

Description: The enemy shall remain in bounds
Rationale: The enemy must remain in the intended boundaries of play for the game to function properly
Fit Criterion: Enemy is unable to pass through walls and other obstacles

Cust. Satisfaction: 2 **Cust. Dissatisfaction:** 5
Priority: High **Conflicts:** None

Supporting Materials: None
History: Created October 7, 2015

Requirement #: 18 **Requirement Type:** 3.2 **Use Case:** 12

Description: The enemies shall be able to successfully carry out attacks on the hero
Rationale: The enemy must be able to carry out attacks to damage and defeat the hero
Fit Criterion: Enemy attack actions occur when hero comes within a set distance

Cust. Satisfaction: 3 **Cust. Dissatisfaction:** 5
Priority: High **Conflicts:** None

Supporting Materials: None
History: Created October 8, 2015

Requirement #: 19 **Requirement Type:** 3.2 **Use Case:** 14

Description: An enemy's hit points shall be reduced when it takes damage
Rationale: Enemies must be able to be damaged
Fit Criterion: Enemy's hit points are reduced upon taking damage

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 9, 2015

Requirement #: 20 **Requirement Type:** 3.2 **Use Case:** 14, 13

Description: An enemy shall be knocked back when taking damage from a hazard
Rationale: Hazards causing damage are intended to cause knock-back
Fit Criterion: An enemy is knocked back when taking damage from a hazard

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 9, 2015

Requirement #: 21 **Requirement Type:** 3.2 **Use Case:** 15

Description: Enemies shall be removed from the game when defeated

Rationale: When enemies are defeated they should no longer be active

Fit Criterion: Enemy is successfully removed when defeated

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None

History: Created October 9, 2015

Requirement #: 22 **Requirement Type:** 3.2 **Use Case:** 15

Description: The hero's score shall increase when an enemy is defeated (by an amount proportional to enemy difficulty)

Rationale: The hero's score determines progress and must increase upon defeating enemies

Fit Criterion: Score increases when enemy is defeated

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None

History: Created October 9, 2015

Requirement #: 23 **Requirement Type:** 3.2 **Use Case:** 16

Description: The user shall be able to open the in-game menu while in-game

Rationale: The in-game menu must be accessible to allow user to save/load games and exit to the main menu

Fit Criterion: The user is able to open the in-game menu

Cust. Satisfaction: 1 **Cust. Dissatisfaction:** 5
Priority: High **Conflicts:** None

Supporting Materials: None
History: Created October 8, 2015

Requirement #: 24 **Requirement Type:** 3.2 **Use Case:** 17

Description: The user shall be able to close the in-game menu
Rationale: The in-game menu must have a way of being closed to resume game play
Fit Criterion: The user is able to close the in-game menu

Cust. Satisfaction: 1 **Cust. Dissatisfaction:** 5
Priority: High **Conflicts:** None

Supporting Materials: None
History: Created October 8, 2015

Requirement #: 25 **Requirement Type:** 3.2 **Use Case:** 16, 17

Description: All gameplay shall be paused when the in-game menu is open
Rationale: The hero should be safe from harm while accessing the in-game menu
Fit Criterion: The game is paused when the in-game menu is opened and resumed when closed

Cust. Satisfaction: 1 **Cust. Dissatisfaction:** 5
Priority: High **Conflicts:** None

Supporting Materials: None
History: Created October 8, 2015

Requirement #: 26 **Requirement Type:** 3.2 **Use Case:** 18

Description: The user shall have the ability to exit the application
Rationale: The user must be able to terminate the game when done playing
Fit Criterion: User is able to successfully terminate application

Cust. Satisfaction: 1 **Cust. Dissatisfaction:** 2
Priority: Medium **Conflicts:** None

Supporting Materials: None
History: Created October 8, 2015

[How many levels will there be? How does the player progress? What happens if the player beats the game? —DS]

[All of your requirements (bar one) are high priority. This should not be the case —DS]

4 Non-functional Requirements

4.1 Look and Feel Requirements

Requirement #: 27 **Requirement Type:** 4.1 **Use Case:** N/A

Description: The game shall use 2-D graphics
Rationale: The game is intended to be a 2-D game
Fit Criterion: 2-D graphics are used for the game

Cust. Satisfaction:	5	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 8, 2015

[What is the theme of the game? Is there a certain aesthetic you are aiming to achieve? —DS]

4.2 Usability and Humanity Requirements

Requirement #: 28 **Requirement Type:** 4.2 **Use Case:** N/A

Description: The game shall be entertaining
Rationale: A game should be fun
Fit Criterion: The game should be ranked at least 7/10 for entertainment based on a usability study

Cust. Satisfaction:	5	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 7, 2015

4.3 Performance Requirements

Requirement #: 29 **Requirement Type:** 4.3 **Use Case:** N/A

Description: The game shall maintain an average framerate of at least 30 fps
Rationale: A framerate of 30 fps or greater will ensure smooth animation
Fit Criterion: The game runs at 30 fps when testing

Cust. Satisfaction:	5	**Cust. Dissatisfaction:**	5
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 7, 2015

4.4 Operational and Environmental Requirements

There are no operational and environmental requirements related to this project.

4.5 Maintainability and Support Requirements

Requirement #: 30 **Requirement Type:** 4.5 **Use Case:** N/A

Description: The game shall support Windows, Linux, and OS X operating systems
Rationale: Students use a variety of operating systems
Fit Criterion: The game compiles and runs on each operating system

Cust. Satisfaction:	3	**Cust. Dissatisfaction:**	3
Priority:	High	**Conflicts:**	None

Supporting Materials: None
History: Created October 7, 2015

4.6 Security Requirements

There are no security requirements related to this project.

4.7 Cultural Requirements

Requirement #: 31	**Requirement Type:** 4.7	**Use Case:** N/A	
Description:	The game shall use the English language		
Rationale:	Students at McMaster University are expected to speak English		
Fit Criterion:	The game uses proper English free of spelling and grammar errors		
Cust. Satisfaction:	1	**Cust. Dissatisfaction:**	3
Priority:	Medium	**Conflicts:**	None
Supporting Materials:	None		
History:	Created October 7, 2015		

4.8 Legal Requirements

There are no legal requirements related to this project.

5 Project Issues

5.1 Open Issues

There are no open issues at this time. This section will be updated as required.

[What about finding your theme or determining how to create the RPG elements? —DS]

5.2 Off-the-Shelf Solutions

There are no off-the-shelf solutions. [What about open-source games created in Chipmunk2D? —DS]

5.3 New Problems

No new problems are expected to arise as a result of this project.

5.4 Tasks

The project will be broken down into the phases given in Table 3.

Table 3: List of project phases

	Phase Name	**Summary**
1.	Interfaces	Programming game interfaces (i.e. working menu systems).
2.	Structures	Programming of game structures and classes.
3.	Mechanics and AI	Programming of game mechanics and AI including physics implementation.
4.	Game Story and Objectives	Planning and programming of the game plot
5.	Graphics and Sound	Addition of textures and sound to the game to provide an enhanced audiovisual experience. This phase is non-crucial.

5.5 Migration to the New Product

There is no product being replaced, and thus no migration is required.

5.6 Risks

There are no risks associated with this project. [Are you sure? —DS]

5.7 Costs

There are no costs associated with this project.

5.8 User Documentation and Training

User documentation will be created as per the CS 4ZP6 guidelines. Training will be provided via built-in tutorials throughout the game.

5.9 Waiting Room

At this point in the project timeline, there are no backlogged requirements. This section will be updated as required.

5.10 Ideas for Solutions

There are no ideas for solutions at this time. This section will be updated as required.