

Requirement Documentation - Revision 1

Mech-APEX Zero

Prepared by:
Tian Guo
Saim Zahid
Jonathan Yu
Yicheng Chen
Yanting Zhang

Professor: Dr. Jacques Carette
Team Name: Chicken and Sausages
Course: Software Engineering 4GP6
Date: February 11, 2018

Contents

1 Purpose of the Project	3
a Background	3
b Goals	4
2 Stakeholders	4
a Development Team	4
b Instructor	4
c Game Critics	4
d User Participation	4
3 Mandated Constraints	4
a Solution Constraints	4
b Implementation Environment of the Current System	4
c Partner or Collaborative Applications	5
d Off-the-Shelf Software	5
e Schedule Constraints	5
f Budget Constraints	5
4 Naming Conventions and Terminology	5
5 Relevant Facts and Assumptions	6
a Relevant Facts	6
b Assumptions	6
6 The Scope of the Work	6
a Existing Inspirations	6
b Context of the Work	7
7 The Scope of the Product	7
a Product Boundary	7
b Product Use Case (PUC) Table	8
c Individual Product Use Cases(PUCs)	10
8 Functional Requirements	23
a Core Mechanic	23
b Primary Gameplay Mode	34
c Alternate Game Modes	35
9 Look and Feel Requirements	42
a Appearance Requirements	42
b Style Requirements	42
c Requisite Assets	43
c.1 Audio	43
c.2 Visual	44
10 Usability and Humanity Requirements	44
a Ease of Use Requirements	44
b Personalization Requirements	45
c Learning Requirements	45
d Understandability and Politeness Requirements	46

e	Accessibility Requirements	47
11	Performance Requirements	48
a	Speed and Latency Requirements	48
b	Precision or Accuracy Requirements	48
c	Reliability and Availability Requirements	49
d	Robustness or Fault Tolerance Requirements	49
e	Capacity Requirements	50
12	Scalability and Extensibility Requirements	50
13	Longevity Requirements	51
14	Cultural Requirements	51
15	Legal Requirements	52
16	Release Requirements	53
17	Security Requirement	53
18	Project Schedule	53
19	Risks	54
20	Costs	55
21	User Documentation and Training	55
22	Waiting List	55
23	Ideas for Solution	55

1 Purpose of the Project

The purpose of this project is to produce the game designed by the team: Mech-APEX Zero. Mech-APEX Zero is the combination of classical 2D games like Metroid, Castlevania, and Street Fighter. This game will reinvigorate the love for such classics and the modern spin of new combat mechanics will attract new players.

a Background

The concept of this game was conceived to revitalize a very saturated genre of video games: Platformer. Currently on Steam, new releases of the Platformer Genre has 1167 results. The motivation of the team is to make a game that would stand out amongst the over one thousand competitors. In order to achieve our goal, we have examined the classics and commercial successes to create our project.

b Goals

The goals of Team Chicken Sausage is to produce an entertaining game to satisfies the needs of our clients. We want our audience to get the enjoyment of playing a platformer and to be dazzled and surprised by our creative story and universe.

2 Stakeholders

a Development Team

Prepared by team Chicken and Sausages. Team members: Tian Guo, Saim Zahid, Jonathan Yu, Yicheng Chen and Yanting Zhang. The grades of each student for this course is at stake.

b Instructor

Dr. Jacques Carette Professor for the Software Engineering (Game Design) capstone course (SE4GP6). Dr. Carette approves of the project content and final grades. Any requirements that he provides must be accommodated.

c Game Critics

- Dr. Jacques Carette - Professor for the Software Engineering (Game Design) capstone course (SE4GP6). Dr. Carette approves of the project content and final grades. Any requirements that he provides must be accommodated.
- Daniel Szymczak - Teaching assistant for the Software Engineering (Game Design) capstone course(SE4GP6). Mr. Szymczak grades the project content and provides feedback for improvements. He reports directly to Dr. Carette.
- Judges for the final demo - Other people or parties that will be judging the final demo of the game.

d User Participation

Player - The players who enjoy platformers and beat em up games.

3 Mandated Constraints

a Solution Constraints

- Any requirements that Dr. Carette describes must be met. Currently, he has mandated that the game engine must be Unity and Unity Assets are free to use with references.
- Any feedback and comments from Mr. Szymczak should be adjust and fix in the next revision.

b Implementation Environment of the Current System

- The final product will be able to run on Windows, Mac OS, and Linux systems, as the Unity library allows to build project for all three platforms.

- Due to the nature of the game being a combo focused action platformer, control methods such as mouse control or touch screen will not be ideal. The development team will focus its effort on establishing keyboard as the main method of controls. Controllers such as Dualshock 4 and Xbox Controller are options, but do not have plans to implement these currently.

c Partner or Collaborative Applications

- The product of this project will comply with all policies of the Unity Free User's Licence Agreement.
- This product will not be used for commercial purposes as the assets used were of other Games.

d Off-the-Shelf Software

- Unity
Usable as a free license for individual work.

e Schedule Constraints

- The deadline of the Project must be within the 2017-2018 Capstone Course terms. Revision 0 of the product must be completed by December of 2017.

f Budget Constraints

- No funding for the project will be provided. Any expenses that occur will be paid individually

4 Naming Conventions and Terminology

- Unity
A cross-platform game engine developed by Unity Technologies, which is primarily used to develop video games and simulations for computers, consoles and mobile devices.
- Platformer
A type of video game that involves the player jumping from platform to platform to cross the environment.
- RPG
RPG stands for role playing game is a type of video game in which players assume the roles of characters in a fictional setting.
- Mobile Suit
A humanoid mechanical weapon. Usually around 18-20 meters tall. Name created in the show "Mobile Suit Gundam". It is not a robot because a person must sit inside its cockpit to operate/pilot it.
- Respawn
The recreation of an entity after its death or destruction

- Check Point
The location where respawning takes place
- Product Use Case (PUC)
A product use case elaborates on a scenario, showing event name, trigger, preconditions, system requirements, and outcome.
- Input (IN)
- Output (OUT)

5 Relevant Facts and Assumptions

a Relevant Facts

- As with most games that utilize combos; combos that do more damage will require more keys to be pressed.

b Assumptions

- The player meets minimum hardware requirements to run the game.
- The player is running the latest version of Windows or Mac OS.
- The player is familiar with a mouse and keyboard setup.
- The player is able-bodied.
- The player has enough dexterity to hit multiple keys in quick succession to perform combos.
- The user can read English.

6 The Scope of the Work

a Existing Inspirations

The biggest inspiration for this project is the Metroid and Castlevania franchises. The cohesive aesthetics and level design are areas that this project will attempt to achieve the same level of competence as the classics of this Genre. We want to use the level designs of Castlevania as an inspiration for large levels, hidden corridors, and enemies to contest. The Aesthetic will be near future sci-fi, similar to Metal Slug series, where there are heavy machinery, giant mechanized weapons, but not too advanced in the future.

The fighting mechanics of Super Smash Bros Franchise will be used to inspire the combat mechanics of this game to introduce new elements of the old platformer formula. The typical platformer has simplistic combat mechanic such as Super Mario Bros franchise, where jumping on enemies and using items are the only way of dealing direct damage. Castlevania has the "wip attacks" and "spells". Metroid has the arm cannon shots, rockets, and special abilities from the suit. Attacks in this games are simple single button commands. In Super Smash Bros, there are light and heavy attacks paired with directional. Each combination of key presses gives different attack moves. There are also basic attack loops for repeated key presses and charged moves for holding down heavy attacks. We use similar mechanics for our platformer as a way to introduce exciting game play to the player.

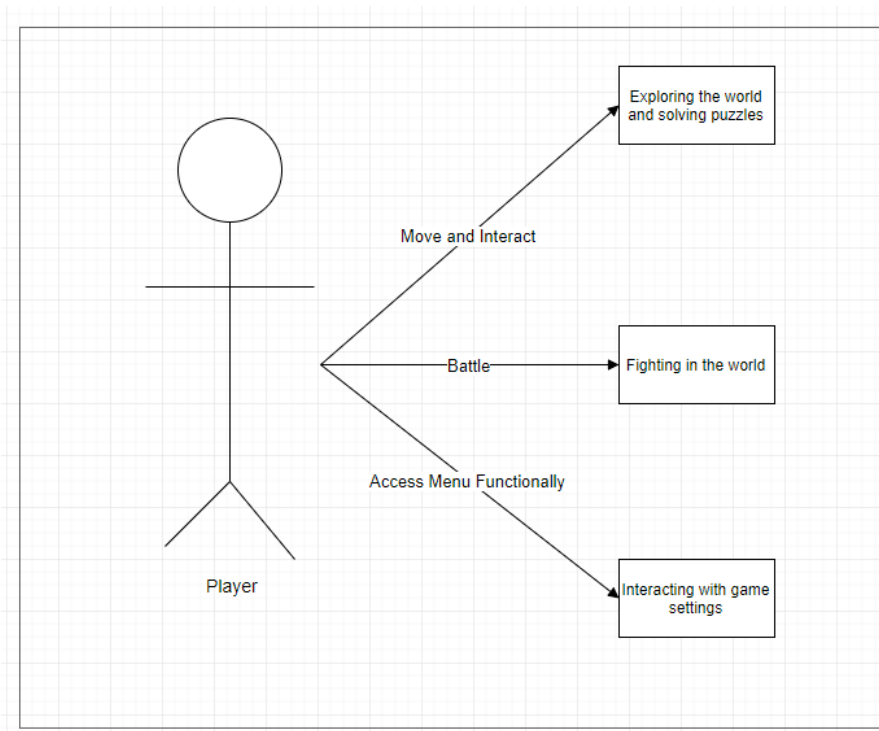
b Context of the Work

Gameplay and level design is the most important aspects of a platformer. Gameplay and combat will be more important for this particular game because of the combat mechanics that is included. We focus a lot of our efforts on refining the gameplay mechanics of the game. We do not have the time to be producing the assets needed for this game, so we will borrow assets from other games that fits our futuristic aesthetic such as Gundam and Megaman. Bugs and feedbacks will be dealt with as much as we can but our main focus will be on the gameplay and level design.

7 The Scope of the Product

a Product Boundary

Project boundaries identify what should be included in the project as well as what should not be included.



The focus will be to complete the fighting mechanic, various character sprite movesets, level design, and Story.

b Product Use Case (PUC) Table

PUC No.	PUC Name	Actor(s)	Input/Output
1	Moving in game	Player	Key Input(IN), x-axis position vector(IN/OUT)
2	Jumping in game	Player	KeyInput(IN), y-axis position vector(IN/OUT)
3	Interacting with items	Player	collision Boolean(IN), success Boolean(OUT)
4	Viewing Player and Status	Player	Key Input(IN), Player information(OUT)
5	Viewing Items	Player	Key Input(IN), items menu(OUT)
6	Performing a melee Attack	Player	Key Input(IN), character sprite Int,(OUT)
7	Performing a ranged Attack	Player	Key Input(IN), character sprite Int,(OUT)
8	Performing a combo Attack	Player	Key Input(IN), character sprite Int,(OUT)
9	Interacting with shop	Player	Key Input(IN), shop menu(OUT)
10	Pausing Game	Player	Key Input(IN), game menu(OUT)
11	Unpausing game	Player	Key Input(IN), current game level(OUT)
12	Use items	Player	Key Input(IN), success Boolean(OUT)
13	Starting a new game	Player	Key Input(IN), success Boolean(OUT)
14	Reaching Checkpoint	Player	Key Input(IN), success Message(OUT)
15	Defeating Boss	Player	Key Input(IN), success Message(OUT)
16	Defeating All Enemies	Player	Key Input(IN), success Message(OUT)
17	Loading next level	Player	Key Input(IN), success Boolean(OUT)
18	Player Death	Player	health <0 (In), reload checkpoint (OUT)

PUC No.	PUC Name	Actor(s)	Input/Output
19	Interacting with upgrades	Player	collision Boolean(IN), success Boolean(OUT)
20	View level objectives	Player	Key Input(IN), current Objectives(OUT)
21	Entering a new Environment	Player	2d velocity vector(IN), 2d position vector(OUT)
22	Viewing Progress	Player	Key Input(IN), Achievement Menu(OUT)
23	Exit game	Player	Key Input(IN)
24	Selecting new playable character	Player	Key Input(IN), success Message(OUT)
25	Displaying the Game Settings	Player	Key Input(IN), current game settings(OUT)
26	Changing the Game Settings	Player	Key Input(IN), current game settings(OUT)
27	Moving the Camera	Player	2D position vector(IN/OUT)

c Individual Product Use Cases(PUCs)

PUC No.1	Event: Moving in game
Trigger	The player presses a key to indicate the direction of the avatar
Preconditions	The player is in control of their avatar; Player is in the game world.
Procedure	<ol style="list-style-type: none"> 1. Add a positive force in the horizontal direction if the right key was pressed. Conversely a negative force if the left key was pressed 2. Set the animation to walk 3. If the force is negative flip the direction the character is facing
Outcome	The character moves in a horizontal direction

PUC No.2	Event: Jumping in the game
Trigger	The player presses the UP/W on their keyboard
Preconditions	The player is in control of their avatar; Player is in the game world
Procedure	<ol style="list-style-type: none"> 1. Move the players character upwards along the y-axis for the predetermined jump height 2. Bring the players characters back down, calculating the speed according to gravity, until they collide with a surface.
Outcome	The players character moves in a positive direction along the y-axis and comes back down

PUC No.3	Event: Interacting with Items
Trigger	The player presses a key to pick the items
Preconditions	The player is in control of their avatar; Player is in the game world
Procedure	<ol style="list-style-type: none"> 1. Determine the if the item is within the players range 2. Determine the type of the item: <ol style="list-style-type: none"> a. If the item is a melee weapon, replace the current melee weapon with the new one, if the current melee weapon is a non - starting weapon, then the old one will drop out. b. If the item is a ranged weapon, replace the current ranged weapon with the new one, if the old one is a non - starting weapon, then the old one will drop out. c. If the item is a passive item(Power ups), put it in the players item bag. d. If the item is a hp potion, recover player characters health bar.
Outcome	The player puts the item in the item bag or recover health bar or get a new weapon

PUC No.4	Event: Viewing Player and Status
Trigger	The player presses the button to open the status menu.
Preconditions	The player is in control of their avatar; Player is in the game world
Procedure	1.Display the current status of the player. Clearly labelling all information the player needs (current health, boost meter, gundam status etc.)
Outcome	The players can see the current status of their character and all the gear they have equipped.

PUC No.5	Event: Viewing Items
Trigger	The player presses i on their keyboard to open up an items menu
Preconditions	The player must be in the game world
Procedure	1. Game pauses the current environment and scene 2. An items menu is on top of the current environment
Outcome	An items menu appears to a player

PUC No.6	Event: Performing a melee Attack
Trigger	The player presses the button to perform a melee attack
Preconditions	The player is in control of their avatar; Player is in the game world
Procedure	<ol style="list-style-type: none"> 1. Determine which type of melee attack the player performed and trigger its animation. 2. Determine if the attack is successful by calculating if an enemies hitbox is within range of the attack. 3. Calculate how much HP the enemy will lose
Outcome	The players performs a melee attack on the enemy, resulting them in losing a predetermined amount of HP

PUC No.7	Event: Performing a ranged attack
Trigger	The player presses a button to attack
Preconditions	The player is in control of their avatar; Player is in the game world
Procedure	<ol style="list-style-type: none"> 1. Check if any special items are collect that modify the players attack 2. Instantiate a projectile prefab (projectile has its own script) in front of the player 3. Add a force to the projectile
Outcome	The players shoots a projectile

PUC No.8	Event: Performing a combo Attack
Trigger	The player presses the buttons mapped to a combo attack
Preconditions	The player is in control of their avatar; Player is in the game world
Procedure	<ol style="list-style-type: none"> 1. Determine which type of combo attack the player performed and trigger its animation. 2. Determine if the attack is successful by calculating if an enemies hit-box is within range of the attack. 3. Calculate how much HP the enemy will lose
Outcome	The players performs a combo attack on the enemy, resulting them in losing a predetermined amount of HP

PUC No.9	Event: Interacting with shop
Trigger	The player presses the button to interact with the shop
Preconditions	The player is in control of their avatar; Player is in the game world
Procedure	<ol style="list-style-type: none"> 1. Determine if the player is within the range of the shop. 2. A menu list of the shop is on top of the current environment. 3. Calculate how much gold would spend on the selected item.
Outcome	The player puts the item in the item bag.

PUC No.10	Event: Pausing the game
Trigger	The player pressing ESC button on keyboard
Preconditions	The player is in control of their avatar; Player is in the game world
Procedure	1. Set time scale to zero 2. Open a pause menu
Outcome	The game is paused

PUC No.11	Event: Unpausing the game
Trigger	The player pressing ESC button on keyboard while game is paused.
Preconditions	The player is in control of pause menu. Does not control avatar.
Procedure	1. Restore time scale to default 2. Close pause menu 3. Return to game
Outcome	The game is paused

PUC No.12	Event: Use Items
Trigger	The player press a button mapped to use items
Preconditions	The player is in control of their avatar; Player is in the game world.
Procedure	<ol style="list-style-type: none"> 1. Player enters item menu 2. Player highlights the item they want to use 3. Player presses to button to use the highlighted items
Outcome	The player will get hp recovered if they used a potion.

PUC No.13	Event: Starting a new game
Trigger	The player chooses the New Game option from the title screen.
Preconditions	The player is not currently in an active game session.
Procedure	<ol style="list-style-type: none"> 1. Determine if the start new game button has been selected 2. Display available avatars to choose
Outcome	The player start the new game and has control of the avatar.

PUC No.14	Event: Reaching Checkpoint
Trigger	Having the player avatar collided with an in-game checkpoint, reaching the next level or defeating certain type of enemy
Preconditions	The player is in control of their avatar; Player is in the game world.
Procedure	1.Determine if the player controls avatar to perform a collision with the checkpoint 2. Set player checkpoint to the most recent checkpoint
Outcome	A message display Progress has been saved successfully

PUC No.15	Event: Defeating Boss
Trigger	Bosss health is reduced to zero
Preconditions	The player is in control of their character; Located in the game world.
Procedure	1.Determine the chances for boss drops any scrap and gold.
Outcome	The player controls avatar to pick up any loot that the Boss was hoarding

PUC No.16	Event: Defeating All Enemies
Trigger	Enemys health is reduced to zero
Preconditions	The player is in control of their character; Located in the game world.
Procedure	1.Determine the chances for enemy drops any scrap.
Outcome	The player controls avatar to pick up any loot that the enemies was hoarding

PUC No.17	Event: Loading next level
Trigger	The player controls avatar reach the destination
Preconditions	The player is in control of their character; Located in the game world
Procedure	1.Determine and save the inventory and attribute of the player 2.Load the next level and the saved information
Outcome	The player start the new level and has control of the avatar

PUC No.18	Event: Player Death
Trigger	The player characters health is zero
Preconditions	The player is in control of their character; Located in the game world.
Procedure	1. Determine the last saved checkpoint. 2. Restore the players avatar to full HP.
Outcome	The players avatar is respawned at the last checkpoint.

PUC No.19	Event: Interacting with upgrade item
Trigger	Player moves towards item
Preconditions	The player is in control of their character; Located in the game world.
Procedure	1. Determine the if the item is within the players range 2. Increase relevant status as upgrade
Outcome	The main character status numbers increase

PUC No.20	Event: View level objectives
Trigger	Player hits a button while highlighting in the pause menu
Preconditions	Player must be in the pause menu associated with the gameworld
Procedure	<ol style="list-style-type: none"> 1. Check what the level the player is on 2. Grab the level objective text 3. Show level objective text
Outcome	Objective text is shown to the player

PUC No.21	Event: Entering a new Environment
Trigger	Player waits for 5 seconds
Preconditions	The player is in control of their character; Located in the game world, all enemies are defeated and boss is cleared
Procedure	<ol style="list-style-type: none"> 1. Kill all enemies and boss or complete puzzle objectives 2. Wait 5 seconds.
Outcome	Game will load next level.

PUC No.22	Event: Viewing Progress
Trigger	The player hits a button while highlighting View Progress in the Main Menu
Preconditions	The player is in the Main Menu
Procedure	1.Get list of levels and list of levels completed 2. Generate percentage of completion
Outcome	Achievement list

PUC No.23	Event: Exit game
Trigger	The player hits "ESC" button and select "Return to main menu" or press "alt" + "F4"
Preconditions	Player must be in the pause menu associated with the gameworld
Procedure	1. The player selects "Return to main menu" in the pause menu.
Outcome	Game exits and main menu shows up

PUC No.24	Event: Selecting new playable character
Trigger	Player hits button "Enter" while highlighting Playable character
Preconditions	Player must be in the character selection menu.
Procedure	1. The player presses "Enter" in the character selection menu.
Outcome	Player enters the game with the character selected.

PUC No.25	Event: Displaying the Game Settings
Trigger	Player hits "Options" option.
Preconditions	Player must be in the main menu/pause menu.
Procedure	1. Player hits "Options" option in main menu/pause menu
Outcome	The game settings menu shows up

PUC No.26	Event: Changing the Game Settings
Trigger	Player selects the setting options
Preconditions	Player must be in the game setting menu
Procedure	<ol style="list-style-type: none"> 1. Player selects the setting options. 2. a. If the player hits "Apply", then the settings changed will be saved. b. If the player did not hit apply and player hits "cancel", then the settings change will not be saved.
Outcome	The game settings are changed/unchanged depends on player's action

PUC No.27	Event: Moving the Camera
Trigger	Player movement
Preconditions	The player is in control of their character; Located in the game world, and the camera is fixed on the player character.
Procedure	1. The player moves, and the camera is fixed on the player character.
Outcome	The camera moves as player moves

8 Functional Requirements

a Core Mechanic

ID:001	Type:Functional Requirement
PUC: 1	Originator:Jonathan Yu
Description	The players avatar should be able to move along the horizontal axis, in the postive direction
Rationale	Avatar should be able to explore the level with the ability to move horizontally, in the postive direction.
Fit Criterion	Avatar moves towards the right when the Right key is pressed, which is the positive direction in our game.
Conflicts:N/A	Supporting Material:N/A

ID:002	Type:Functional Requirement	
PUC: 1	Originator: Saim Zahid	
Description	The players avatar should be able to move along the horizontal axis, in the negative direction	
Rationale	Avatar should be able to explore the level with the ability to move horizontally, in the negative direction.	
Fit Criterion	Avatar moves towards the left when the Left key is pressed, which is the negative direction in our game.	
Satisfaction: 4	Dissatisfaction: 5	Priority: Very High
Conflicts:N/A	Supporting Material:N/A	

ID:003	Type:Functional Requirement	
PUC: 1	Originator: Jonathan Yu	
Description	The players avatar should be able to jump, achieved by adding a vertical force in the positive Y direction.	
Rationale	Avatar should be able to explore the level with the ability to move Vertically.	
Fit Criterion	Avatar gains positive Y velocity when the Space key is pressed. Velocity decreases overtime due to gravity, eventually returning the player to back to the ground.	
Satisfaction: 4	Dissatisfaction: 5	Priority: Very High
Conflicts:N/A	Supporting Material:N/A	

ID:004	Type:Functional Requirement	
PUC: 2	Originator:Jonathan Yu	
Description	All characters in the game screen will be subjected to gravity, a downward force in the negative Y direction every ms.	
Rationale	Define one constant rate of downward motion.	
Fit Criterion	Use Rigid Body property of Unity	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts:N/A	Supporting Material:N/A	

ID:005	Type:Functional Requirement	
PUC:3	Originator:Jonathan Yu	
Description	After picking up an item, ensure that the attributes that the items modifies are accurate,in conjunction with a formula	
Rationale	It gives the player a feel of progression, making his machine stronger the longer he plays, also gives him a sub goal. The main idea is to destroy your enemies and take their gear to bolster your own	
Fit Criterion	<p>When the player collides with an item his attributes are changed depending on the type of item. For example if he collects and attack upgrade, he will do a larger % of damage.</p> <ol style="list-style-type: none"> 1. There will be damage upgrade items. When Avatar collides with item, there will be a multiplier to attack value. 2. There will be speed upgrade items. When Avatar collides with item, there will be multiplier to movement speed and jump velocity value. 3. There will be health upgrade items. When Avatar collides with item, maximum health value will be increased. 4. There will be usable items. When Avatar collides with item, item will be added to inventory, can be used by pressing the designated button and item will no longer be in inventory. 	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:006	Type:Functional Requirement	
PUC:4	Originator:Yanting Zhang	
Description	The player must be able to view their avatar's status	
Rationale	The player will want to check the current status of their avatar(current health, ammo count, etc.)	
Fit Criterion	The player will be able to open the status menu by pressing the player status menu button. This will display current attributes of both the avatar.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:007	Type:Functional Requirement	
PUC:4	Originator:Yanting Zhang	
Description	The player must be able to view the avatar's gear's status	
Rationale	The player will want to check the current status of their avatar gear.)	
Fit Criterion	The player will be able to open the status menu by pressing the player gear status menu button. This will display current attributes the avatar's gear.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:008	Type:Functional Requirement	
PUC:5	Originator:Yanting Zhang	
Description	The player must be able to view the item's menu.	
Rationale	The player will want to check current items of the avatar.	
Fit Criterion	The player will be able to open the item's menu by pressing the view item's menu button.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:009	Type:Functional Requirement	
PUC:4,5	Originator:Yanting Zhang	
Description	The system must be able to calculate and display the avatar's current attributes modified by the used items.	
Rationale	If the system cannot calculate and display the avatar's current attributes including the equipped items, the player will not know what effect has been applied by a consumed item.	
Fit Criterion	The avatar's modified attributes are calculated and displayed according to the used items.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:010	Type:Functional Requirement	
PUC:4,5	Originator:Yanting Zhang	
Description	The system must be able to calculate and display the avatar's current attributes modified by the used items.	
Rationale	If the system cannot calculate and display the avatar's current attributes including the equipped items, the player will not know what effect has applied on with the new used item.	
Fit Criterion	The avatar's modified attributes are calculated and displayed according to the the used items.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:011	Type:Functional Requirement	
PUC:6	Originator:Yanting Zhang	
Description	The player must be able to perform a melee attack	
Rationale	Since the game is based around fighting and defeating enemies, being able to perform a melee attack is crucial to the game. Melee attacks are the backbone of most fighting games.	
Fit Criterion	The player will press the button to perform a melee attack, the players avatar will display the attack on the screen.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:012	Type:Functional Requirement	
PUC:7	Originator:Saim Zahid	
Description	The player must be able to perform a ranged attack	
Rationale	Since the game is based around fighting and defeating enemies, being able to perform a ranged attack is crucial to the game. Almost all successful fighting games supplement melee fighting with ranged attacks.	
Fit Criterion	The player will press the button to perform a ranged attack, the players avatar will display the ranged attack on the screen.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:013	Type:Functional Requirement	
PUC:8	Originator:Saim Zahid	
Description	The player must be able to perform a combo attack	
Rationale	One of the main selling points of our game is being able to perform combo moves. It also adds a new dimesnsion to our games fighting elements. It also keeps it fresher for longer for players because learning and executing them takes a while.	
Fit Criterion	The player will press the buttons mapped to a combo attack, the players avatar will display the combo attack on the screen.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:014	Type:Functional Requirement	
PUC:6,7,8	Originator:Yanting Zhang	
Description	The system must be able to calculate the damage for obstacles according to the selected attack performed.	
Rationale	When player performs an attack on obstacles, the system should be able to determine how much damage will be applied against the obstacles.	
Fit Criterion	When player performs an attack on obstacles, the system should be able to determine how much health the obstacles must lose from different types of attacks.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:015	Type:Functional Requirement	
PUC:10	Originator:Yanting Zhang	
Description	The player must be able to pause the game	
Rationale	If the player wishes to do something while in game, they should be able to pause the game.	
Fit Criterion	If the player press the pause button, the gameplay will be paused.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:016	Type:Functional Requirement	
PUC:11	Originator:Yanting Zhang	
Description	The player must be able to resume the paused the game	
Rationale	If the game is paused, the player must be able to resume the game.	
Fit Criterion	If the player press the unpause button, the gameplay will be resumed.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Medium
Conflicts:N/A	Supporting Material:N/A	

ID:017	Type:Functional Requirement	
PUC: 16	Originator:Jonathan Yu	
Description	When the players avatar enters a level, a fixed amount of enemies are created. If all enemies are destroyed than, the game will load,to the next level.	
Rationale	The player requires a win condition to complete the level.	
Fit Criterion	1. Fixed number of enemies will be spawned when level is loaded. Enemy will spawn at fixed locations. Enemies aggro when player is within range. 2. When enemy counter reaches zero, level completes.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts:N/A	Supporting Material:N/A	

ID:018	Type:Functional Requirement	
PUC: 18	Originator:Jonathan Yu	
Description	When the players collides with an invisible preset boundary, the game reloads the level and the players lives are decremented by one, if the the players lives are less than 0 then the game loads the game over scene.	
Rationale	To prevent the player from accessing a part of the level and creating a natural boundary for the intended game world.	
Fit Criterion	Assert if the instance of the object and the player that collided with each other are still within the scene	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts:N/A	Supporting Material:N/A	

ID:019	Type:Functional Requirement	
PUC: 18	Originator:Jonathan Yu	
Description	When the players collides with an enemy, the player loses a percentage of their total health depending on what enemy he collided with, if the players health reaches below zero than the game reloads the current level and the players lives are decremented by one, if the the players lives are less than 0 then the game loads the game over scene.	
Rationale	One of the primary fail conditions for the player. A game needs to have challenges and consequences of failing the challenge. For this game, if the player is not player well enough and takes too much damage, the avater in the context of the game will be destroyed. The player will have to suffer losing the game and start over.	
Fit Criterion	Assert if the instance of the object and the player that collided with each other are still within the scene	
Satisfaction: 4	Dissatisfaction: 5	Priority: Very High
Conflicts:N/A	Supporting Material:N/A	

ID:020	Type:Functional Requirement	
PUC: 18	Originator:Jonathan Yu	
Description	All characters will have their own health pool.	
Rationale	Mobiles suits can only sustain so much damage before they are destroyed, health points are used as a representation of the about of damage they can take. For game play experience and real life representation, different machines will have various amounts of hp. Boss characters will have the most.	
Fit Criterion	Set various HP amount for characters.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts:N/A	Supporting Material:N/A	

ID:0021	Type:Functional Requirement	
PUC: 18	Originator:Jonathan Yu	
Description	When the player attacks, he will instantiate a new box collider that covers the range of his attack, if that box collides with another object it loses a percentage of their health that is scaled by the players attack, if their health goes below zero than play the death animation and destroy the object.	
Rationale	The rides in a mobile suit as a weapon the player needs a means to destroy the enemy mobile suits	
Fit Criterion	New box collider for select animation frames. On collision with enemy box, decrease enemy health.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts:N/A	Supporting Material:N/A	

ID:0022	Type:Functional Requirement	
PUC: 23	Originator:Yanting Zhang	
Description	The system must be able to return the game state to the title screen.	
Rationale	There must be a way for the player to exit the game from within it.	
Fit Criterion	If the player enters the pause menu and selects the quit game option, they will be asked to confirm they wish to quit. If they select yes, they will be brought to the title screen.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts:N/A	Supporting Material:N/A	

b Primary Gameplay Mode

ID:0023	Type:Functional Requirement	
PUC: 21	Originator:	
Description	The primary gameplay mode will be Story Mission Mode where players shall complete the each levels of the game. The win condition is to complete all levels.	
Rationale	Primary mode should be able to explore all levels by sequence and present the story.	
Fit Criterion		
Satisfaction:	Dissatisfaction:	Priority: Very High
Conflicts:	Supporting Material:	

c Alternate Game Modes

ID:0024	Type:Functional Requirement	
PUC: 24, 26	Originator: Tian Guo	
Description	Free Mission Mode allows player can choose which playable character to use and which level to play. After select level is completed, return to mission select screen.	
Rationale	Alternate mode allows the player more freedom.	
Fit Criterion	Have a mode that can be started from the main menu which allows the player to choose which playable character to use and which level to play.	
Satisfaction:	Dissatisfaction:	Priority: Medium
Conflicts:	Supporting Material:	

ID:0025	Type:Functional Requirement	
PUC: 25, 26	Originator: Tian Guo	
Description	A start menu that allows the player to select Story Mission Mode, Free Mission Mode, Options to change game settings, and Exit to terminate the game.	
Rationale	Game should begin at menu to give player option to choose how to play and options to change settings.	
Fit Criterion	<ol style="list-style-type: none"> 1. Game will boot into start menu. 2. Start menu shall contain Story Mission, Free Mission, Option, and Exit buttons. 3. Each button will execute the relevant functions. 	
Satisfaction:	Dissatisfaction:	Priority: Very High
Conflicts:	Supporting Material:	

ID:0026	Type:Functional Requirement	
PUC:10, 11, 25, 26	Originator: Tian Guo	
Description	During game missions the player can open the pause menu to pause the game. The menu will consist of Resume to unpause the game from pause menu, Options to change certain settings, and Back to Title to return to main menu.	
Rationale	During missions, player should have the ability to open pause menu if need be. Play can change settings in this screen if adjustments are needed.	
Fit Criterion	1.During mission player shall be able to open pause menu by pressing ESC key. 2. Pause menu shall pause the game until the menu is closed. 3. Pause menu shall have resume, options, and back to title buttons. 4. Pressing ESC again while in pause menu shall resume the game as well.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:0027	Type:Functional Requirement	
PUC:10, 11, 25, 26	Originator: Tian Guo	
Description	During game missions if the player has paused the game the player will have ability to return to their game session by pressing the Resume button. They could also use the "Back to Title" button to return to main menu.	
Rationale	During missions while the game is paused, the player should have the ability to return to the game.	
Fit Criterion	The pause menu should have a button called "Resume" which gives the player the ability to return back to their game session.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:0028	Type:Functional Requirement	
PUC: 28	Originator: Saim Zahid	
Description	Enemies will actively try to harm the player's avatar, potentially even kill the avatar.	
Rationale	Every game needs an element that is thrilling to the player, more often than not this is knowing that fact that their avatar can die. Adding this to our game will make it more marketable and will keep players intrested for longer. This will also make the game more challenging which is appealing to most gamers.	
Fit Criterion	When the player encounters an enemy, the enemy will use their equipped weapon to harm/kill the player. It will not cease until the player has killed it.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:029	Type:Functional Requirement	
PUC: 6,7,8	Originator: Saim Zahid	
Description	As the player progress's in the game and meets newer enemies, the players weapons will do the same amount of damage but the enemies will lose less health then the player's weapon's base damage, as the newer enemies will have better armor.	
Rationale	The game should feel progressively harder, in terms of defeating enemies, to make it more challenging. Rather than weakening the players weapons or giving them worse gear, giving the enemies better armor will give the player a better challenge, all the while not making them feel like they regressed by weakening their weapons.	
Fit Criterion	As the player encounters stronger enemies as the progress in the game, the enemies will have better armor, making them harder to defeat.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:030	Type:Functional Requirement	
PUC: 15	Originator: Saim Zahid	
Description	When the player defeats the main boss of each level the enemy should give the player feedback that it has been defeated. This feedback includes an empty hp bar and a distinct death animation.	
Rationale	The player needs to know when they have deafted the main boss. A depleting HP bar shows how much more damage the boss can sustain and also when players see they have an alomst empty HP bar on a tough boss fight it gives them the extra push to succeed. A death animation is good feedback to the player that the boss has ben defeated.	
Fit Criterion	When the player defeats a boss, the boss' sprite will show a distinct death animation marking the end of a fight.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:0031	Type:Functional Requirement	
PUC: 16	Originator: Saim Zahid	
Description	When the player defeats an enemy, the enemy should give the player feedback that it has been defeated. This feedback includes an empty hp bar and a death animation.	
Rationale	The player needs to know when they have deafted an enemy A depleting HP bar shows how much more damage the enemy can sustain. A death animation is good feedback to the player that the enemy has ben defeated.	
Fit Criterion	When the player defeats an enemy, the enemy's sprite will show a death animation marking the end of a fight.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:0032	Type:Functional Requirement	
PUC: 9	Originator: Saim Zahid	
Description	When the player defeats an enemy, theres a chance they will drop scrap.	
Rationale	Rewarding the player after defeating an enemy is crucial for them to like the game. Making the player work hard to acieve an objective then not rewarding them is not a good requirement. This also increases the longevity of the game as the player will want to play more to collect more scrap to upgrade their suit.	
Fit Criterion	When the player defeats an enemy, they might drop a piece of scrap that can be walked over to pick up.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:0033	Type:Functional Requirement	
PUC: 9	Originator: Saim Zahid	
Description	When the player defeats a boss, the boss will drop scrap	
Rationale	Rewarding the player after defeating a boss is crucial for them to like the game. Making the player work hard to achieve an objective then not rewarding them is not a good requirement. This also increases the longevity of the game as the player will want to play more to collect more scrap to upgrade their suit.	
Fit Criterion	When the player defeats a boss, the boss will drop scrap that can be walked over to pick up.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:0034	Type:Functional Requirement	
PUC: 3	Originator: Saim Zahid	
Description	When the player encounters a useable weapon they can pick it up to use it.	
Rationale	When the player collides with a weapon they can pick it up. This will keep the combat fresh as they will get to use newer/better weapon. This will also allow them to personalize their avatar as they can use whichever type of weapon they like.	
Fit Criterion	When a player picks up a weapon it becomes their currently equipped weapon and they can use it freely.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:0035	Type:Functional Requirement	
PUC: 18	Originator: Saim Zahid	
Description	The player's avatar will be able to lose HP, when they get hit by enemies basic attacks.	
Rationale	Giving the enemies a basic attack will help players understand the enemies patterns and help beat them. A completely unpredictatble enemy is very hard to beat and frustrarting the player is not good in many cases.	
Fit Criterion	Every enemy in the game will have a basic attack they use to harm the enemy.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:0036	Type:Functional Requirement	
PUC: 18	Originator: Saim Zahid	
Description	The player's avatar will be able to lose HP, when they get hit by enemies special attacks.	
Rationale	Giving the enemies a special attack will add a new dimension to the fight. They will be alot stronger and have a new animation. This will make the fight alot more thrilling all the while making it more challenging.	
Fit Criterion	Every enemy in the game will have a special attack they use to harm the enemy. There will be a certain trigger condition that actiavtes this type of attack. Mostly it will be when the enemy falls below an HP threshold.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

9 Look and Feel Requirements

a Appearance Requirements

ID:101	Non-Functional Requirement(Appearance)	
PUC: 1, 2, 10, 25, 26	Originator: Tian Guo	
Description	Characters must look distinct from each other. Player must be able to tell difference between the player avatar and enemy easily. Foreground and background must differentiate easily.	
Rationale	Object on screen must represent itself clearly	
Fit Criterion	1.Player can easily distinguish self from others. 2. Player can easily distinguish foreground and background	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

b Style Requirements

ID:101	Type:Non-Functional Requirement(Look and Feel - Style)	
PUC: 19, 21	Originator: Tian Guo	
Description	Objects in the game must exert futuristic style	
Rationale	The look of the game should reflect the Sci-Fi setting	
Fit Criterion	Objects should look mechanical, has electronics, or seems to be made from metal.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

ID:102	Type:Non-Functional Requirement(Look and Feel - Style)	
PUC: 2, 6, 7, 8	Originator: Tian Guo	
Description	Playing the game must feel like mix of action and platforming	
Rationale	Playing the game should feel like brawler or fighting game from the combat mechanic. Traversing the game should feel like platformer and Metroidvania games.	
Fit Criterion	Game should have solid fighting game mechanics and platformer level design.	
Satisfaction:	Dissatisfaction:	Priority: High
Conflicts:	Supporting Material:	

c Requisite Assets

c.1 Audio

Asset Type	Number Required	Rationale
Character Sound Effect	2	Require minimum two sets for playable characters' sound effects
Enemy Character Sound Effects	6	Require minimum 6 different sets of sound effects for enemy types and bosses
Background Sound Track	6	Require minimum 6 Tracks for different scenery

c.2 Visual

Asset Type	Number Required	Rationale
Main Character Sprite	2 Sets	Require minimum two sets for playable
Enemy Character Sprite	6 Sets	Require minimum 6 different enemy types and bosses
Background Sprite	6 Sets	Require minimum 6 sets for different scenery
Foreground Sprite	6 Sets	Require minimum 6 sets for foreground objects and platforms

10 Usability and Humanity Requirements

a Ease of Use Requirements

ID:201	Type: Non-Functional Requirement(Ease of Use)	
PUC: 1, 2, 20	Originator: Tian Guo	
Description	Requirement on how easily player can play the game physically.	
Rationale	Player should not be hindered by the controls while playing	
Fit Criterion	Button layout for controlling player avatar must be intuitive and similar to other games of the genre.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

b Personalization Requirements

ID:202	Type: Non-Functional Requirement(Personalization)	
PUC: 26	Originator: Saim Zahid	
Description	Customization requirements	
Rationale	Player should be able to play the game as per personal preference.	
Fit Criterion	Player must be able to change the video and audio settings, pausing and unpausing the game, and changing the mobile suit of the character as per preference.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

c Learning Requirements

ID:203	Type:Non-Functional Requirement(Learning)	
PUC:	Originator: Tian Guo	
Description	Requirement of how easily a new player should be able to pick up the game	
Rationale	Player should not spend large amount of time to learn to play the game	
Fit Criterion	The main control and concept of the game must be very intuitive and easy to learn. Player must be able to grasp the control of the game within few minutes of first stage.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

d Understandability and Politeness Requirements

ID:204	Type:Non-Functional Requirement(Understandability and Politeness)	
PUC:1, 2, 6, 7, 8, 14, 15, 16	Originator: Yicheng Chen	
Description	Requirement of how easily a player understand the game and in what level of politeness that the language of the game is representing.	
Rationale	Players should not have a hard time understanding the language of the game. The dialogue and context should not contain hate speech	
Fit Criterion	The conversation, subtitles, naming, lores must be in English and have appropriate grammar. And they should not contain hate speech at all. Players must have a comfortable gaming environment during the game.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

ID:213	Type:Non-Functional Requirement(Understandability and Politeness)	
PUC:1, 2, 6, 7, 8, 14, 15, 16	Originator:Tian Guo	
Description	Requirement of how to display tutorial and helpful information	
Rationale	The game shall have information displayed for player to see.	
Fit Criterion	Information must be displayed in text boxes clearly readable by the player.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

ID:217	Type:Non-Functional Requirement(Understandability and Politeness)	
PUC:13, 14, 15	Originator:Tian Guo	
Description	Requirement of how to display story and world relevant information	
Rationale	The game shall have information displayed for player to see.	
Fit Criterion	Information must be displayed in text boxes clearly readable by the player.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

e Accessibility Requirements

ID:205	Type:Non-Functional Requirement(Accessibility)	
PUC: 1	Originator: Tian Guo	
Description	How easily targeted audience can get ahold of the product	
Rationale	Game should be playable on common hardware and platforms.	
Fit Criterion	The program must be able to run on PC, Mac, and Linux platforms.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

11 Performance Requirements

a Speed and Latency Requirements

ID:206	Type:Non-Functional Requirement(Performance - Speed and Latency)	
PUC: 1, 2, 6, 7, 8	Originator: Tian Guo	
Description	Requirement of how fast and responsive the game should be.	
Rationale	Due to the games genre as a platformer and the focus on combos, the game must feel very responsive. The games theme is Sci-Fi and Mecha, therefore the speed does not need to be very fast.	
Fit Criterion	Players input must not have large latency. Response from the game should be almost instantaneous.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

b Precision or Accuracy Requirements

ID:207	Type:Non-Functional Requirement(Performance - Precision or Accuracy)	
PUC: 6, 7, 8	Originator: Tian Guo	
Description	Requirement of how accurate and precise hitboxes are.	
Rationale	Unlike traditional fighting games, there will not be high, mid, and low strikes and blocks. The hitbox of characters will be more lenient than games such as Tekken, Street Fighter, or Mortal Kombat.	
Fit Criterion	If the attack appears to have landed on enemies, the hit must register.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

c Reliability and Availability Requirements

ID:208	Type:Non-Functional Requirement(Performance - Reliability and Availability)	
PUC: All	Originator: Yicheng Chen	
Description	Requirement for party of liability and availability of the party.	
Rationale	Game should have reliable party. And game should have available support.	
Fit Criterion	Development team must be liable for all issues of the game. Dev team must be available to solve issues before April 28th 2018.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

d Robustness or Fault Tolerance Requirements

ID:209	Type:Non-Functional Requirement(Performance - Robustness or Fault Tolerance)	
PUC: All	Originator: Tian Guo	
Description	Requirement of the frequency of crashes and unexpected errors.	
Rationale	Any crashes or unexpected errors that disrupts gameplay should not be occurring.	
Fit Criterion	The game will not crash unexpectedly under normal circumstances. The game will perform at acceptable level on all platform and machines that it is running on.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

e Capacity Requirements

ID:210	Type:Non-Functional Requirement(Performance - Capacity)	
PUC: 1, 17, 21	Originator: Yichneg Chen	
Description	The system should be able to create and maintain at most thirty(30) characters in one game level, and at most ten(10) characters on the screen.	
Rationale	With too many characters showing up on one screen could dramtically affect game performance.	
Fit Criterion	The game must allow fifteen to thirty characters to show up in one level. Fifteen to thirty enemies is a good interval for balance of the level. The player would not feel the level is too difficult or take too long. The upper limit of thirty also would not significantly affect game performance.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

12 Scalability and Extensibility Requirements

ID:211	Type:Scalability	
PUC: All	Originator: Yichneg Chen	
Description	The game should be coded in a way where adding new levels are possible without changing previously written code	
Rationale	The game can be expanded upon later, downloadable content could be added, most importantly code can be reused for different games.	
Fit Criterion	Game will follow standard coding practices and split in module and scripts to prevent difficulty in replacing a particular part.	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

13 Longevity Requirements

ID:212	Type:Non-Functional Requirement(Performance - Longevity)	
PUC: All	Originator: Yichneg Chen	
Description	Requirement of a minimum of Fifteen(15) minutes game play to be played by the players.	
Rationale	The final product should be able to be played for a minimum of fifteen(15) minutes for graded demonstration.	
Fit Criterion	The player must be able to play the game for at least fifteen minutes. And the player must be able to explore	
Satisfaction: 5	Dissatisfaction: 5	Priority: Very High
Conflicts: N/A	Supporting Material: N/A	

14 Cultural Requirements

ID:214	Type:Non-Functional Requirement(Performance - Longevity)	
PUC: 21	Originator: Jonathan Yu	
Description	The product shall not have offensive symbols to any religious or ethnic group.	
Rationale	Although I believe that game should not be censored in any form but having our game potential banned within our country is harmful.	
Fit Criterion	The product must pass through Canadian cultural policy	
Priority	Low	

15 Legal Requirements

ID:215	Type:Legal Requirements
PUC: All	Originator: Jonathan Yu
Description	The product shall adhere to all terms and conditions set out by Unity.
Rationale	Breaking Unitys term and conditions allows them the right to deny us the usage of their engine.
Fit Criterion	The product will not break any of the terms and conditions set by Unity.
Priority	Very High

ID:216	Type:Legal Requirements
PUC: All	Originator: Jonathan Yu
Description	The product shall adhere to the standards set by the professor
Rationale	The professor grades our product, not following standards can affect our grades in a negative way.
Fit Criterion	Product must receive passing grade
Priority	Very High

16 Release Requirements

ID:218	Type:Release Requirements
PUC: All	Originator: Tian Guo
Description	The product shall be able to run on standard hardware requirements on release
Rationale	Product should be able to run on computer hardware that most people has or better.
Fit Criterion	Should be able to run on the following detailed specification as a bare minimum: Desktop: OS: Windows XP SP2+, Mac OS X 10.9+, Ubuntu 12.04+, SteamOS+. Graphics card: DX9 (shader model 3.0) or DX11 with feature level 9.3 capabilities. CPU: SSE2 instruction set support. Audio levels: Recommended to be listened to at a comfortable level.
Priority	Very High

17 Security Requirement

ID:218	Type:Non-Functional Security Requirements
PUC: All	Originator: Jonathan Yu
Description	The product shall protect all sensitive information
Rationale	The risk of sensitive information being released is worth the cost of protecting it
Fit Criterion	All sensitivity information shall be encrypted
Priority	Very High

18 Project Schedule

The project will follow the guide up on the courses webpage. The work will be done before the deadline of every task set up by Dr. Carette.

- Main Character Sprite Animation: Nov 3
- Character Sprite Animation: Nov 10
- Behaviour of Enemies: Nov 4
- Button Mapping and Hitbox set up: Nov 17
- Level Design and Platform set up: Nov 24
- Finalization and Testing: Dec 1
- Movement and Combat Mechanic bug fix: Feb 19
- Boost meter mechanic implementation: Feb 19
- General bug fixes from Git Feedback: Feb 23
- New level designs: Mar 2
- New bosses implementation with new levels: Mar 2
- Verification and Validation Testing: Mar 4
- Final implementation: Mar 25

Schedule subject to change based on progress.

19 Risks

- The product demands a large portion of time and commitment, once the project has been started, future commitments and opportunity are now lost.
- Failure to satisfy any of the functional requirement would cause the game to break and deliver unsatisfactory experience to the player.
- Learning to use sprite animator in Unity is one of the most important tool to learn for a 2D sprite based game. The quality of sprite animation can significantly affect the quality of a 2D action game.
- The feel of the game play depends heavily on the implementation of sprite animations, movement speed adjustments, and physics. Failure to properly implement these elements would make the game feel clunky to the player.
- Failure to design interesting levels will make the game feel bland and sub-par comparing to any other beat em ups.
- Developer team attend the same meeting physically due to work and class schedules,
- If the game is a massive hit with the community there could be potential copyright claims.
- if the project isnt implement up to standard than our course grade will be affected.
- the amount of bugs within the game can affect the quality of the game. For details refer to "Bug Impact" section of RevisionPlanRev1.pdf.

20 Costs

- The entire product will be funded by team. As of now the free version of Unity is being utilized coupled with free, open-source assets. In the future we might have to buy certain assets that fit the theme of the game better. The financial costs will be limited to purchases of assets.
- The game should be completed by the end of April 2018. Considering the level design, multiple playable characters, multiple weapons, combo based attacks which require a unique animation the game will take entire allotted time to complete.

21 User Documentation and Training

- User Documentation Requirements.
 - The game will include a user manual outlining all basic movement controls, basic attacks, environment interaction keys, and all combos included in the game. This manual can be accessed by the start menu and the pause menu.
- Training Requirements
 - The game will be very easy to learn and understand, accordingly we intend for the player to learn as they go. They should be familiar with the the basic movement and attacking commands within the completion of the first level of the game. We encourage player autonomy therefore the player can learn which ever combos they wish to use by looking them up in the user manual. As of now we do not intend to put in a standalone training level before the game actually starts.

22 Waiting List

- Expansions to the game introducing even more combos.(PUC No.8)
- Expansion of the number of levels
- Expansion of the number of enemies

23 Ideas for Solution

Enemy Ai is very large and difficult problem within many games. There are many ways in order to implement the ai within the game and here are some ideas. The first idea, that will apply to many enemies, is to have the Ai constantly grab the location of the players avatar and add force in the direction of the player, if the player location is close enough than the enemy should attack. If the enemy is a ranged enemy and they share the same y coordinate with the player, then the enemy should shoot a bullet in the direction of the player.

The second idea is enemies can have few attacks that can be cycled through against the player. Enemies once within aggression range will move towards the player. Enemy can shoot at the player while moving, and can also start using melee attack cycles when close enough to the player.

The third idea is to have predetermined movement for enemies. Enemies will move in the same direction regardless of the player.