1. Introduction

The Player Controller will allow the user/player to interact with the world and systems that Sandstorm Games’ will make. The player controller will feature systems to allow the player to move, systems to allow the player to collect and use items/tools, and systems to allow the player to interact with the world and enemies.

2. Use case diagram with scenario

2.1 Use Case Diagram

A diagram of a game

AI-generated content may be incorrect.

2.2 Scenarios

**Move:** allows the player to move in the around the world

**Actors:** Player

**Preconditions:** input is received

**Basic Sequence:**

**.1** determine if input is an accepted input

**.2** determine movement direction based on input

**.3** move player in the determined direction

**Exceptions:**

**.1** the input is not meant for player movement, ignore input

**.2** an obstacle is in the player’s path, so no movement happens

**Postconditions**: move player left or right

**Priority**: 1

**ID**: PC01

**Jump:** allows the player to jump to cross gaps in the world

**Actors:** Player

**Preconditions:** input is received

**Basic Sequence:**

**.1** determine if input is an accepted input

**.2** determine movement direction based on input

**.3** move player in the determined direction

**Exceptions:**

**.1** the input is not meant for player movement, ignore input

**.2** an obstacle is in the players path, so no movement happens

**Postconditions**: player jumps in left or right

**Priority**: 2

**ID**: PC02

**Collision:** allows the player to receive damage from enemies and climb walls

**Actors:** Player

**Preconditions:** player has collided with an object

**Basic Sequence:**

**.1a** the player collided with a wall

**.2a** so, the player starts climbing the wall

**.1c** the player collided with enemy

**.2c** so, the player takes damage

**Exceptions:**

**.1c** the player is attacking the enemy

**.2c** the player damages the enemy and takes no damage

**Postconditions**: collision event resolves

**Priority**: 3

**ID**: PC03

**Inventory:** allows the player to collect and use items/tools

**Actors:** Player

**Preconditions:** input is received

**Basic Sequence:**

**.1a** the player pressed the input for item collection

**.2a** the player is on an item in the world

**.3a** the player collects the item

**.3aa** the collected item is a treasure, the player’s score is increased

**.3ab** the collected item is a tool, the item is added to the inventory

**.1b** the player pressed input for equipping tool

**.2b** the inventory slot opens in accordance with input

**.2ca** the inventory slot has an item, so the item is equipped

.**2cb** the inventory slot is empty, so no item is equipped

**Exceptions:**

**.1a/b** the input is for none of the above actions, so input is ignored

**.2a** the player is not on an item in the world, so input is ignored

**Postconditions**: input is resolved

**Priority**: 3

**ID**: PC04

3. Data Flow diagram(s) from Level 0 to process description

3.1 Data Flow Diagram(s)

A diagram of a game system

AI-generated content may be incorrect.

A diagram of a game

AI-generated content may be incorrect.

3.2 Process Description

**2.1** Player Movement:

WHILE *right input* is pressed

move player right

END WHILE

WHILE *left input* is pressed

move player left

END WHILE

IF *jump input* is pressed AND player is on ground

player jumps

ENDIF

WHILE player is collided with wall surface AND player is moving in the direction of the wall

player climbs wall

END WHILE

**2.2** Player Attack:

IF tool is being held AND *use input* is pressed

Player uses tool

IF tool hits enemy AND tool can deal damage

Player gets some knockback force applied AND damages enemy

ELSE IF tool hits environment AND tool can affect environment

The player digs through environment

END IF

ENDIF

IF *swap item* input is pressed/used

The player swaps to the selected inventory slot

IF there is an item in inventory slot

Item is held by player

ELSE

Player switches to empty inventory slot, and holds nothing

END IF

END IF

**2.3** Item collection:

IF *pick up item* input is pressed AND player is on item in world

Item is picked up

IF an item is in the selected inventory slot

Picked up item replaces the item in that slot

ELSE

Picked up item is place in the empty inventory slot

END IF

END IF

**2.4** Health Management:

IF enemies attacks at player AND hits player

Player health is lowered for the amount of damage taken

END IF

4. Acceptance Tests

To test the movement system, we can test/call the movement functions and the log the change in position that the function causes.

To test the input system, we can test for race conditions (like if the player presses two buttons at the same time) by having many inputs inputted at the same time and log what happens.

To test the inventory system, we can have an automated system pick up every item in the game in every possible order, and log the events that occur. To watch for potential issues.

5. Timeline

5.1 Work items

|  |  |  |
| --- | --- | --- |
| Task | Duration (in relative expected work load) | Predecessor Task(s) |
| **1**. Make Player Character | 1 | - |
| **2.** Input system | 1 | **1.** |
| **3.** Movement system | 2 | **2.** |
| **4.** Wall climbing system | 4 | **2.** |
| **5.** Inventory system | 3 | **3.**, **4.** |
| **6.** Interaction system (item collection) | 2 | **5.** |
| **7.** Health System | 1 | **5.** |
| **8.** Polish/testing movement / climbing system | 2 | **7**. |
| **9.** Polish/testing Input system (allow for different kinds of input, ie controller) | 2 | **7.** |
| **10.** finish / final touchup the player controller system | 1 | **9.**,**8.** |
| **Total** | 19 |  |

5.2 Pert diagram

A screenshot of a computer

AI-generated content may be incorrect.

5.3 Gantt timeline

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1.** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2.** |  | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3.** |  |  | 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4.** |  |  | 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5.** |  |  |  |  |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  |
| **6.** |  |  |  |  |  |  |  |  |  | 5 |  |  |  |  |  |  |  |  |
| **7.** |  |  |  |  |  |  |  |  |  | 5 |  |  |  |  |  |  |  |  |
| **8.** |  |  |  |  |  |  |  |  |  |  |  | 7 |  |  |  |  |  |  |
| **9.** |  |  |  |  |  |  |  |  |  |  |  | 7 |  |  |  |  |  |  |
| **10.** |  |  |  |  |  |  |  |  |  |  |  |  |  | 9 |  |  |  |  |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |