Software Requirements Specification

for

SushiTale

**Version 1.0**

**SushiWorld**

**03/02/2023**

**Table of Contents**

[**1. Introduction (Section: 30%)**](#_heading=h.1fob9te) **1**

[**1.1 Product Scope**](#_heading=h.fhwlwvipi58r) **1**

[**1.2 References**](#_heading=h.55y8i76q4jy0) **1**

[**2. Product Perspective (Section: 25%)**](#_heading=h.tyjcwt) **1**

[**2.1 Product Functions**](#_heading=h.bwtpfnzh8ik6) **1**

[**2.2 Assumptions and Dependencies (Risk and Mitigations)**](#_heading=h.167hsxaj6guo) **2**

[**3. External Interface Requirements (Section: 5%)**](#_heading=h.4d34og8) **2**

[**3.1 User Interfaces (UI Research)**](#_heading=h.jiu6bwdg7xsb) **2**

[**4. System Features (Section: 40%)**](#_heading=h.9roicp6392e9) **3**

[**4.1 Shop Room**](#_heading=h.6h2yo0a18c9t) **3**

[**4.2 Rest Room**](#_heading=h.8e25nd9pu926) **3**

[**4.3 Arrow Battle Room**](#_heading=h.najhr23h1g9g) **3**

[**4.4 Platform Battle Room**](#_heading=h.9wpbeaj7onhh) **4**

[**4.5 Final Boss Room**](#_heading=h.p62tbmovo5hz) **4**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction (Section: 30%)

## Product Scope

Sushi Tale, a role-playing game that takes you on a culinary adventure through a fantasy world filled with delicious sushi dishes. The game will be a single-player, turn-based RPG that takes place on a randomly generated map. The player will control a character who must explore the map, defeat enemies, and collect ingredients to prepare for the final boss battle.

## References

Room Mechanics

* Slay the Spire

Battle Mechanics

* Audition
* Cuphead

# Product Perspective (Section: 25%)

## Product Functions

* Gameplay Mechanics
  + The game will have the player spawn on an 8x6 map of multiple rooms the player has to choose which direction to go, and the aim is to reach the final boss room
  + The player will have a set of stats e.g health, strength, which will affect the performance during the combat.
  + The player will be able to upgrade their stats by going to specific rooms.
  + When the player runs out of health points, all progress is lost and needs to start a new game.
* Room Mechanics
* There should be 4 different room types
* Battle room 
  + Players battle enemies to gain currency/upgrades.
* Rest room
  + Players can choose to heal up or upgrade an ingredient acquired
* Event room
  + A random event would happen that would benefit or harm the player
* Shop room
  + Players can have a choice to purchase one of the ingredients with their currency
  + Once they leave the room, the shop will close forever

* Gear progression
  + There should be 4 upgradable ingredients that could be upgraded to level ?? from level 1.
    - Salmon - Increase max HP
    - Tuna - Increase time in battle
    - Swordfish - Increase damage
    - Squid - Increase Evade
* Battle Mechanics
  + Arrow Key room
    - When the battle begins, players will be given a set of arrow keys to input in order before the time runs out.
    - Based on the number of correct inputs, it would then determine the damage taken to the player/enemy depending on who’s turn to attack/defend.
  + Platform Battle Room
    - Player Shoots bullets, jumping up and down to dodge boss attacks
    - Bullets deal damage to bosses.
    - Varying boss attack patterns.
* Level Design
  + The difficulty level will decide the final boss stats
  + The difficulty of the combat will increase based on the number of rooms explored.
    - Inverted keys
    - Higher enemy stats
    - Unrevealed keys that reveal themselves when approaching it

### Assumptions and Dependencies (Risk and Mitigations)

Assumptions

1. Development delays: The game development might take longer.
2. If random rooms can’t be implemented, we can change them to fixed rooms
3. Wrong damage computations, fixed damage based on stats

Dependencies

1. None.

# External Interface Requirements (Section: 5%)

## User Interfaces (UI Research)

1. The How to Play button will be included in the main menu, explaining how to play the game.
2. Player stats and item levels will be accessible while exploring the map.
3. Pause for players to take a break
4. Option to allow players to choose out of two randomly generated room types when entering a new room.

# System Features (Section: 40%)

# Shop Room

Description

* Player would be able to choose if they want to buy an ingredient

Stimulus/Response Sequences

* Upon entering the shop, the player is able to see 2 choices of ingredients/items
* Each shop will have a different price
* Based on player’s money, they have a choice to buy or leave
* Once sold, no refunds
* Once the player leaves, the shop will be closed till the player finds another shop.

Functional Requirements

* REQ-1: Shop would sell a random ingredient that the player has not gotten
  + If all ingredients were present, the shop would sell upgrades
* REQ-2: The shop would disappear once the player leaves the room

# Rest Room

Description

* Player would be able to choose to heal or upgrade

Stimulus/Response Sequences

* Upon entering the room, the player will have to choose to heal up or upgrade an ingredient
* Heal up would heal the player a fixed value of health
* Upgrade an ingredient
  + Player would be required to fill a bar by spamming a key, this would decide their upgrade passing rate

Functional Requirements

* REQ-1: Rest room is a one-time use
* REQ-2: Create the upgrade mechanics. -> The bar would increase per key pressed, the bar would also be decreasing at a rate depending on the level of ingredient. The upgrade will then pass or fail based on the percentage of the bar filled
* REQ-3: Heal up the player's health

# Arrow Battle Room

Description

* Battle against enemies
* Dealing damage to the enemy using arrow keys

Stimulus/Response Sequences

* Upon entering an arrow key battle room, the enemy will have a specific set of fixed attack power. During every turn, the player will enter a set of arrow keys of varying orientations within the set time.
* If the Player successfully inputs all arrow keys, full damage will be dealt to the enemy.
* If the Player fails to clear within the time limit, the player will receive damage equivalent to enemy attack power minus the number of correctly pressed keys.
* If the Player fails any arrow input, the player will receive damage.

Functional Requirements

* REQ-1: Arrow Key generator -> function to create a new random set of arrow key inputs.
* REQ-2: Input Timer -> To check the time taken for the player to key in arrow keys
* REQ-3: Damage Calculation -> Compare player’s input, speed and accuracy of input to determine the damage to be dealt to the enemy. Other factors, such as the player's state may affect the damage value dealt.

## Platform Battle Room

Description

* Battle against enemies
* Dealing damage by shooting them while dodging

Stimulus/Response Sequences

* Upon entering the room, enemies would be on the right attacking the player to the left
* Player would then dodge/shoot until either side runs out of hp

Functional Requirements

* REQ-1: Create enemies attack patterns -> create several attack styles and random the pattern
* REQ-2: Platforming the area for the player to use.
* REQ-3: Physics affecting the player and bullets.

## Final Boss Room

Description

* Final room where the player fights against the boss

Stimulus/Response Sequences

* Upon entering the room, the player can choose from the previous type of battle room
* Difficulty would be increased based on player stats
* IF Player chooses Arrow
  + Rotating arrow
  + Arrow require multiple input
  + Reversed Arrow key (arrow key order starts from the back)
* IF Player choose Platform
  + Limit Player’s vision
  + Anti-gravity
  + Create mobs

Functional Requirements

* REQ-1: At least one battle room type is implemented
* REQ-2: Physics systems must be implemented.
* REQ-3: Increase of difficulty compared to the whole game

**Appendix A: Glossary**

**Appendix B: Analysis Models**