

**Super Wizard Fight 3D Design and Development**

Ryan Cologgi, David Grougan, Tristan Fitzwater, Sarah Kyne, Cassandra Méndez, Brandon Pagan

CSE 5912, Spring 2019

Instructor: Dr. Roger Crawfis

The Ohio State University

## **Narrative**

In a two-dimensional world, wizards existed in peace and harmony. Until one day, four brave wizards ventured off in search of more: three dimensions. Four may enter, but only one may stay three-dimensional. All wizards must fight to become the Super Wizard Fighter.

## **Domain: Summary of Gameplay**

SWF3D is a couch multiplayer inspired by games such as the Binding of Isaac, HYPERJAM, and Fortnite. The dungeon will be set up as a large central room surrounded by many smaller rooms. Rounds are intended to be relatively quick at an estimated 10-15 minutes. As a competitive rogue-lite dungeon crawler, players are encouraged to roam the dungeon and collect items from the rooms in the first minutes of the round to arm themselves for the duel. As time goes on, rooms get destroyed, forcing players to the center. Players compete to be the best Wizard fighter by defeating each other in combat.

## **Target Audience**

This party game is intended for all ages. It's meant to be a casual game to be played among friends that does not require heavy skill, but some savvy and luck.

## **Functional Requirements**

### **Environment**

1. Dungeon map must be procedurally generated with a large, central battle room and bordering rooms with an array of items.
2. Dungeon map must not have any inaccessible parts (i.e. rooms that are disconnected).
3. Dungeon rooms must be procedurally destroyed from the outside to the center battle room.
4. Dungeon rooms being destroyed must show animation when a player is nearby.
5. Dungeon rooms being destroyed must kill the player inside if he/she does not exit within the allocated time.
6. Dungeon rooms must be randomly decorated with dungeon assets like tables, bookshelves, and statues.
7. Camera must be in a top-down orientation.
8. Camera must follow the player slightly while staying locked to the room.
9. Static assets (e.g. barrels, chairs, tables) must have colliders such that players can interact and destroy them.

### **Players**

1. Players must be able to move, controlled by Left Stick on gamepad or WASD on keyboard.
2. Players must be able to turn with Right Stick on gamepad or Mouse on keyboard.
3. Players must be able to attack with Triggers on gamepad or Space on keyboard.
4. Players must not be able to move vertically (i.e. jump).
5. Players must begin the game distributed in separate corners.
6. Players must be able to navigate to unique rooms.

7. Players must be given basic combat and must have the alter basic combat with item pickup.
8. Players must be given the opportunity to fight Non-Playable Characters.
9. Players must have states so they can take damage when hit, recover with specific items, and be killed.
10. Players must be able to die from interactions with opponents, NPCs, or being destroyed in a room.
11. Players must be able to pick up items and interact, whether it's swapping Primary Items, storing Activatable Items, or consuming Passive Items.
12. Players must have visible health and statistics.
13. Players must be able to access a triggered minimap to show their location.

### **Items & NPCs**

1. Items must be characterized by Primary, Activatable, and Passive.
2. Player item inventory must be limited to one Primary and one Activatable item.
3. Passive items must be consumed immediately.
4. Items must be spawned from chests or from an NPC when it's killed.
5. Higher value items must spawn later into the game.
6. NPCs must be controlled by AI.
7. NPCs must be confined to their respective spawning rooms.

### **Game Mechanics**

1. Game must have save and load states.
2. Game must initialize by showing Unity and HeadSmash load screen, followed by the menu.
3. Menu must have options to Start Local, Start Server, Options, Credits, and Quit.
4. Start Local submenu must ask for number of players and prompt for Xbox controller or keyboard input before launching game.
5. Start Server submenu must start a server game through the Bolt client.
6. Options submenu must be able to toggle music and sound effects and change brightness.
7. Credits submenu must show the names of all developers and have an option to Return to menu.
8. Quit option must quit the game cleanly.
9. When in-game, Pause menu must be triggered by Start on controller or Esc on keyboard.
10. Pause menu must have options to Save, Quit to Menu, or Quit.
11. Game must have music for menus and gameplay.
12. Game must have sound effects for menu clicks, selection, walking, entering doors, NPCs, chests opening, item pickup, item use, combat, and all other major mechanics.