Name: Tavin Yorgason Mark \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/50

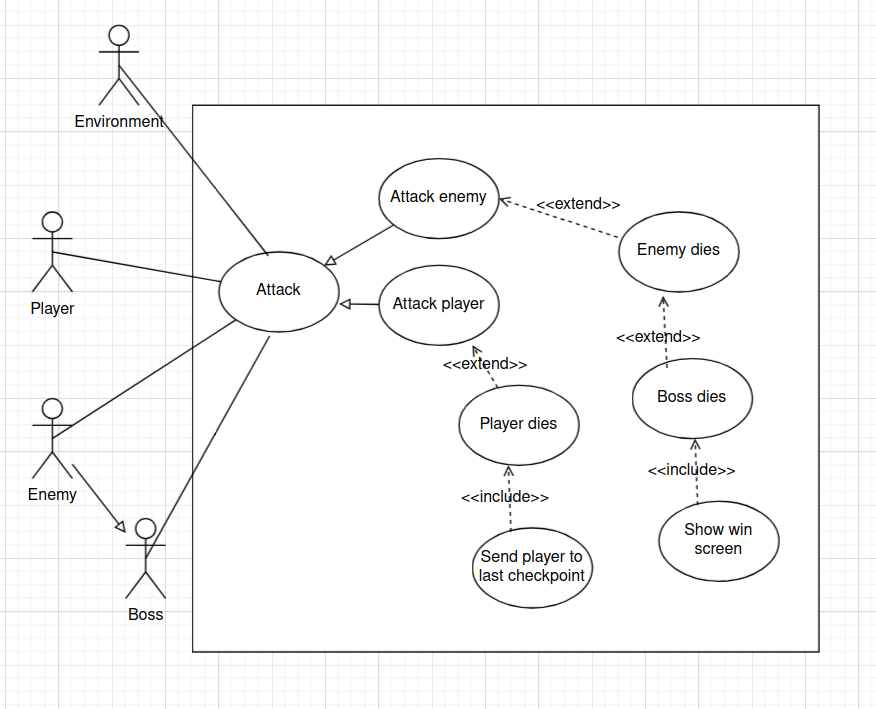
## Brief introduction \_\_/3

The Boss fight is an integral part of InBetween, and must utilize many of the mechanics and items that the player has collected along the way. It brings everything together and will be the hardest part of the game. The Boss fight will include multiple waves of varied attacks.

It will include many, if not all of the enemies that the player has previously seen. As the main enemy to kill, there will be a boss that resembles St. Michael, the Archangel of the purgatory. Once the player kills St. Michael, the game ends and an animation plays and the win screen is shown.

## Use case diagram with scenario \_\_14

### Use Case Diagram



### Scenario

**Name:** Boss Fight

**Summary:** The player attacks the enemies and the enemies and environment attack the Player. The fight continues until either the Boss dies or the Player dies.

**Actors:** Accountant.

**Preconditions:** Player has reached Boss fight room.

**Basic sequence:**

**Step 1:** Boss spawns in.

**Step 2:** Normal enemies spawn in.

**Step 3:** Enemies attack player.

**Step 4:** Player attacks enemies.

**Step 5:** Environment attacks player.

**Step 6:** Normal enemies die.

**Step 7:** Boss dies.

**Step 8:** Win screen shows.

**Exceptions:**

**Step 3:** Player attacks before enemies.

**Step 6:** Player dies without any enemies dying.

**Step 7:** Player dies without the boss dying.

**Step 8:** If the player has died, go back to last checkpoint.

**Post conditions:** Either Player or Boss dies.

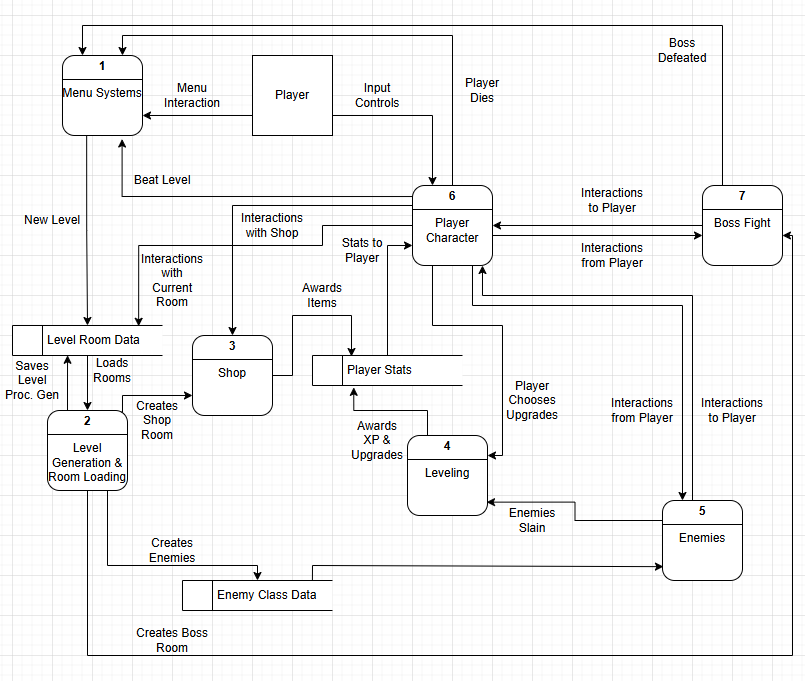
**Priority:** 1

**ID:** C01

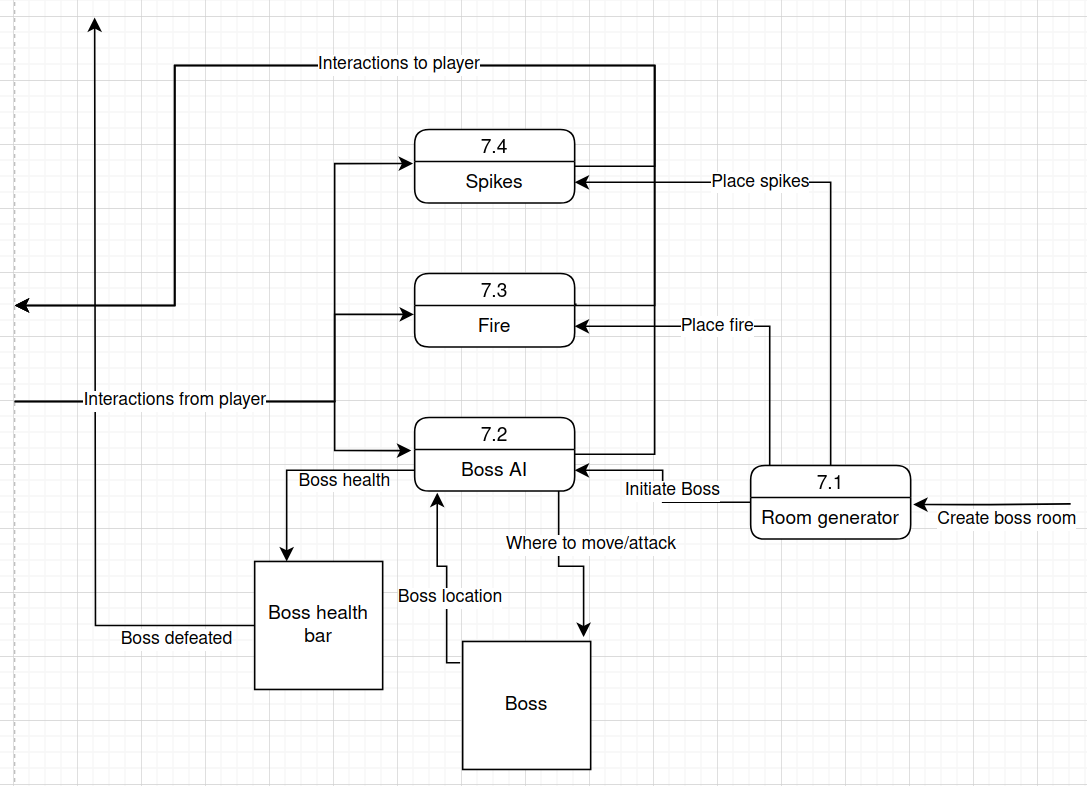
\*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

## Data Flow diagram(s) from Level 0 to process description for your feature \_\_\_\_\_\_\_14

### Diagram 0



### Level 1 diagram, section 7: Boss Fight



### Process Descriptions

**Room Generator:**

for i in 10:

instantiate(spike)

instantiate(fire)

instantiate(Boss)

**Boss AI:**

followPlayer()

attack1()

attack2()

for i in 5:

spawnEnemy1()

if health <= maxHealth \* (2/3):

for i in 3:

spawnEnemy2()

attack3()

attack4()

if health <= maxHealth / 3:

for i in 2:

spawnEnemy3()

attack5()

attack6()

**Fire:**

if collision == player:

for ii in 7:

player.damage(1)

**Spike:**

if collision == player:

player.damage(10)

## Acceptance Tests \_\_\_\_\_\_\_\_9

Testing will ensure that all attacks, enemies, and environmental damage is dealt with properly. To test everything in the level, play the level and get hit by every enemy attack and die to every enemy attack. Also, run into the player into the spikes and fire. Experiment with every attack and move the player can do and ensure that they interact with the boss as intended.

**Input/output examples**

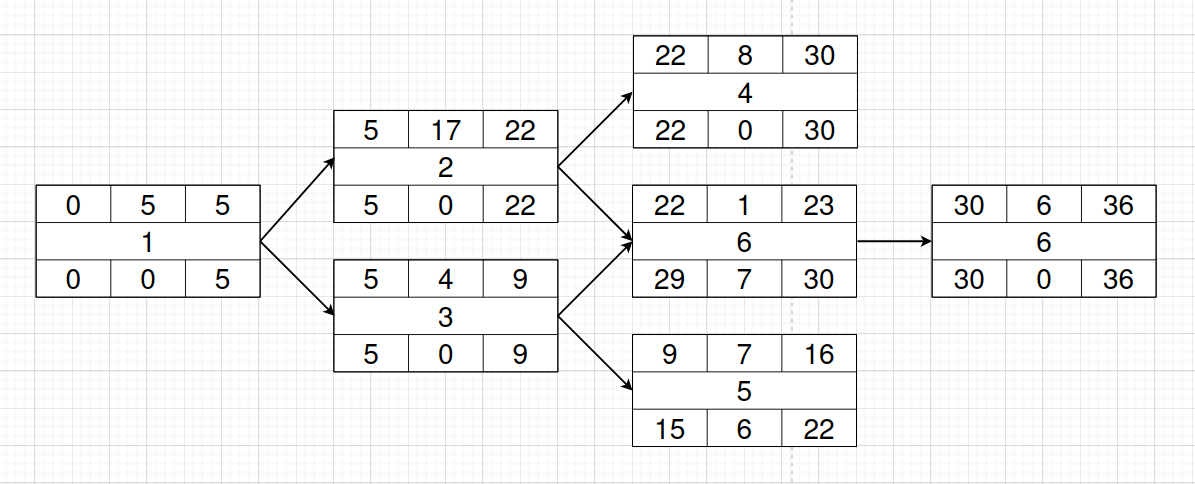
|  |  |  |
| --- | --- | --- |
| Input | Output | Notes |
| Player run into fire | 7 damage to player | Damage should be spread out |
| Player run into spike | 10 damage to player |  |
| Player damage boss | Boss health bar goes down |  |
| Kill boss | Win screen |  |
| Player die | Death screen | Death screen should send them back to the last checkpoint when they respawn |
| Boss at 2/3 or 1/3 health | Boss uses new attacks and spawns more enemies |  |
| Player gets hit by boss attack | Damage to player | Damage varies between different attacks |
| Player enters room | Boss follows player and walking animation plays |  |

## Timeline \_\_\_\_\_\_\_\_\_/10

### Work items

|  |  |  |
| --- | --- | --- |
| Task | Duration (hrs) | Predecessor Task(s) |
| 1. Plan Boss fight details | 5 | - |
| 2. Program Boss | 17 | 1 |
| 3. Program environment | 4 | 1 |
| 4. Create sprites and animations for boss | 8 | 2 |
| 5. Create sprites and animations for environment | 7 | 3 |
| 6. Add in enemies made by enemy designer. | 1 | 2, 3, Prototype enemies from enemy designer |
| 7. Testing and balancing | 6 | 4, 5, 6 |

### Pert diagram



### Gantt timeline

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | 5 |  |  |  |  | 1 | 0 |  |  |  | 1 | 5 |  |  |  | 2 | 0 |  |  |  | 2 | 5 |  |  |  | 3 | 0 |  |  |  | 3 | 5 |  |  |  | 4 | 0 |  |  |  | 4 | 5 |  |  |
| 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |