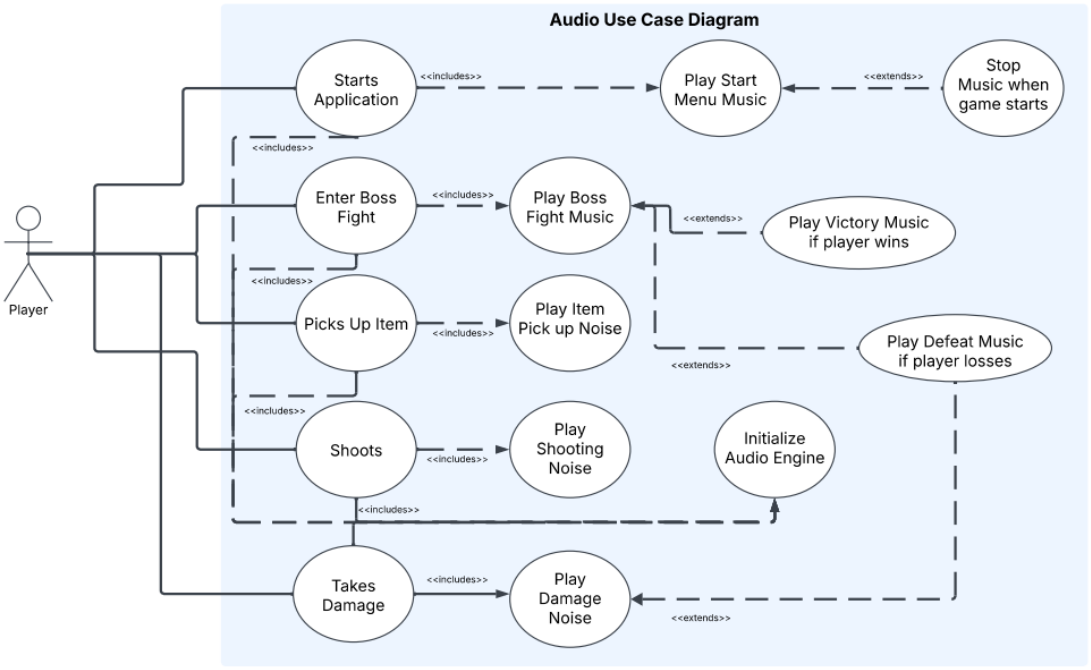
Name: Braxton Marlatt Mark _____/50

## 1. Brief introduction

I will implement the audio system for our game. The main menu and the final boss fight will both feature music. Unique sounds will play when the player shoots, picks up an item, and takes damage.

## 2. Use case diagram with scenario _14




Scenarios

**Name:** Start Application

**Summary:** The player loads the game.

**Actors:** Player

**Preconditions:** The game is on the computer

**Basic sequence:**

1) Player selects the application on their computer
2) Game is loaded
3) The main menu is show

**Exceptions:**

1) None, this will always occur when loading the game.

**Post conditions:** Player is in main menu.

**Priority:** 1*

**ID:** C01

**Name:** Enter Boss Fight
**Summary:** The player has completed the main portion of the game and is ready to fight the boss
**Actors:** Player
**Preconditions:** The player has collected all the necessary items to fight the boss.
**Basic sequence:**

**1)** Player collects items
**2)** Player travels to the boss area
**3)** Player commits to fight the boss

**Exceptions:**

**1)** The player does not have the necessary items to fight the boss

**Post conditions:** Player is fighting the boss
**Priority:** 1*
**ID:** C02

**Name:** Picks Up Item
**Summary:** The player picks up an item on their list in the main portion of the game
**Actors:** Player
**Preconditions:** The player has entered the main menu and selected to play the game
**Basic sequence:**

1) Player moves to area with item
2) Player moves over location with item
3) Player picks up item

**Exceptions:**

1) The player already has the item

**Post conditions:** Player has item in inventory and is one step closer to boss.
**Priority:** 1*
**ID:** C03

**Name:** Shoots
**Summary:** The player shoots their weapon at an enemy
**Actors:** Player
**Preconditions:** The player has collected their weapon and selected in in their inventory
**Basic sequence:**

1) Player hovers mouse over location to shoot
2) Player clicks to shoot
3) Bullet travels to location and hits enemy or misses

**Exceptions:**

1) The player has dropped their weapon
2) The player is out of ammo

**Post conditions:** Player deals damage to enemy or bullet disappears without hitting enemy
**Priority:** 1*
**ID:** C04

**Name:** Takes Damage
**Summary:** Player is hit by the weapon of an enemy
**Actors:** Player
**Preconditions:** Player is in area of enemy in the main game
**Basic sequence:**

1) Player enters vision of enemy
2) Enemy shoot projectile at player
3) Player is hit by projectile and takes damage

**Exceptions:**

1) The projectile misses

**Post conditions:** The players health is lower
**Priority:** 1*
**ID:** C05

**Name:** Play Boss Fight Music
**Summary:** Player is fighting the boss and intense music plays
**Actors:** Player
**Preconditions:** Player has made it to the boss fight, audio engine is initialized
**Basic sequence:**

1) Player enters the boss fight
2) Music Plays

**Exceptions:**

1) None, if the player is in the boss fight, music will play

**Post conditions:** Music is playing
**Priority:** 2*
**ID:** C06

**Name:** Play Main Menu Music
**Summary:** Player is in the main menu and music plays
**Actors:** Player
**Preconditions:** Player has loaded the game
**Basic sequence:**

1) Player loads the game
2) Player enters the main menu
3) Music Plays

**Exceptions:**

1) If the player starts the game, the music will stop

**Post conditions:** Music is playing
**Priority:** 2*
**ID:** C07

**Name:** Stop music when game starts
**Summary:** Player starts the game, and the main menu music starts
**Actors:** Player
**Preconditions:** Player has selected to play the game, audio engine is initialized
**Basic sequence:**

1) Player selects to play the game in the main menu
2) Scene is changed to main game

**Exceptions:**

1) None, if the player is playing the game the main menu music will stop

**Post conditions:** Music is stopped
**Priority:** 2*
**ID:** C08

**Name:** Play pick up noise
**Summary:** Player interacts with an item and a noise plays
**Actors:** Player
**Preconditions:** Player is playing the main game, audio engine is initialized
**Basic sequence:**

1) Player moves over the area of a item
2) Item is picked up
3) Noise plays to indicate items have been picked up

**Exceptions:**

1) None, if the player picks up an item the noise will play

**Post conditions:** Noise
**Priority:** 2*
**ID:** C9

**Name:** Play shooting noise
**Summary:** Player shoots and a noise plays
**Actors:** Player
**Preconditions:** Player is playing the main game, audio engine is initialized
**Basic sequence:**

1) Player shoots a weapon
2) Projectile is fired
3) Shooting noise is played

**Exceptions:**

1) None, if the player shoots the noise will play

**Post conditions:** Noise
**Priority:** 2*
**ID:** C10

**Name:** Play damage noise
**Summary:** Player takes damage from an enemy and a noise plays
**Actors:** Player
**Preconditions:** Player is playing the main game, audio engine is initialized
**Basic sequence:**

1) Player enters the area of an enemy
2) Enemy fires at the player and hits them
3) Damage is delt and the noise plays

**Exceptions:**

1) If the player takes damage and runs out of health, then defeat music will play instead

**Post conditions:** Noise
**Priority:** 2*
**ID:** C11

**Name:** Initialize Audio Engine
**Summary:** Player takes damage from an enemy and a noise plays
**Actors:** Game System
**Preconditions:** The game is launching or resuming from a state where no audio needs initialized
**Basic sequence:**

1) Game starts or resumes
2) Game system initializes the audio engine and all necessary resource
3) Audio system checks for available sound devices and loads assets
4) Background music starts playing if necessary

**Exceptions:**

1) If initialization fails, an error is logged, and a fallback mechanism is used.

**Post conditions:**

1) The audio engine is fully functional, ready to handle background music and sound effects.
2) Game sounds (such as shooting, item pickups, and damage sounds) can now play without issues.

**Priority:** 1*
**ID:** C12

**Name:** Play victory music if player wins
**Summary:** Player defeats the boss and music plays
**Actors:** Player
**Preconditions:** Player is playing the main game, audio engine is initialized
**Basic sequence:**

1) Player enters the area of the boss
2) Enemy defeats the boss
3) A victory screen is displayed and music plays

**Exceptions:**

1) None, if the player defeats the boss music plays

**Post conditions:** Victory Music

**Priority:** 2*
**ID:** C13

**Name:** Play defeat music if player losses
**Summary:** Boss defeats the player and music plays
**Actors:** Player
**Preconditions:** Player is playing the main game, audio engine is initialized
**Basic sequence:**

1) Player enters the area of the boss
2) Boss defeats the player
3) A defeat screen is displayed and music plays

**Exceptions:**

4) None, if the boss defeats the player music plays

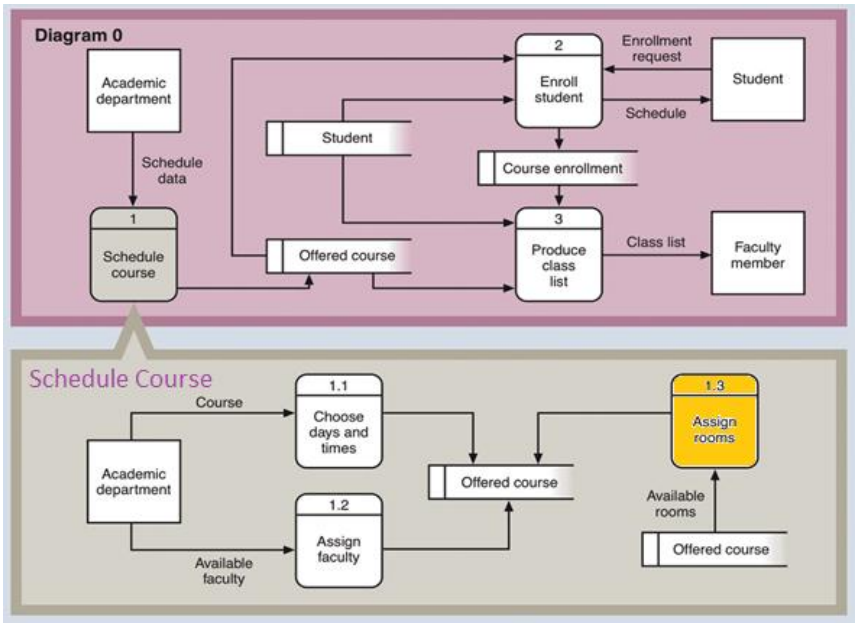
**Post conditions:** Defeat Music
**Priority:** 2*
**ID:** C14

## 3. Data Flow diagram(s) from Level 0 to process description for your feature _____14

[Get the Level 0 from your team. Highlight the path to your feature]

Example:

### Data Flow Diagrams




### Process Descriptions

Assign rooms*:

WHILE teacher in two places at once OR two classes in the same room

Randomly redistribute classes

END WHILE

***Notes**: Yours should be much longer. You could use a decision tree or decision table instead if it is more appropriate.

## 4. Acceptance Tests _____9

[Describe the inputs and outputs of the tests you will run. Ensure you cover all the boundary cases.]

**Example for random number generator feature**

Run feature 1000 times sending output to a file.

The output file will have the following characteristics:

- Max number: 9
- Min number: 0
- Each digit between 0 and 9 appears at least 50 times
- No digit between 0 and 9 appears more than 300 times
- Consider each set of 10 consecutive outputs as a substring of the entire output. No substring may appear more than 3 times.

**Example for divide feature**

| Output | Numerator (int) | Denominator (int) | Notes |
|---|---|---|---|
| 0.5 | 1 | 2 | |
| 0.5 | 2 | 3 | We only have 1 bit precision for outputs. Round all values to the nearest .5 |
| 0.0 | 1 | 4 | At the 0.25 mark always round to the nearest whole integer |
| 1.0 | 3 | 4 | At the 0.75 mark always round to the nearest whole integer |
| 255.5 | 5 | 0 | On divide by 0, do not flag an error. Simply return our MAX_VAL which is 255.5. |

## 5. Timeline _____/10

[Figure out the tasks required to complete your feature]

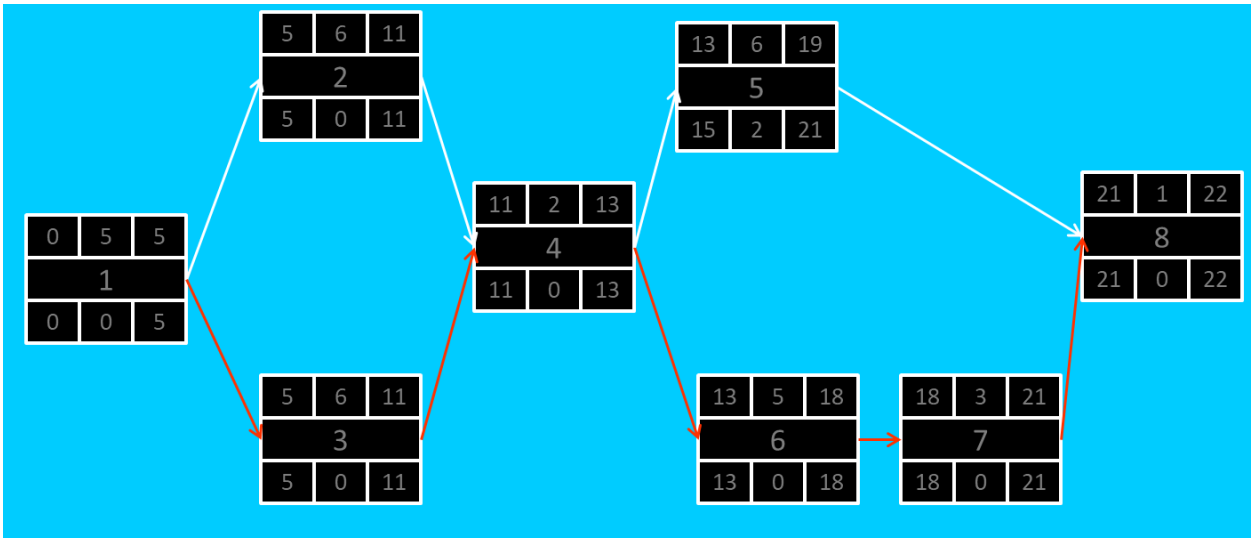
Example:

### Work items

| Task | Duration (PWks) | Predecessor Task(s) |
|---|---|---|
| 1. Requirements Collection | 5 | - |

| | | |
|---|---|---|
| 2. Screen Design | 6 | 1 |
| 3. Report Design | 6 | 1 |
| 4. Database Construction | 2 | 2, 3 |
| 5. User Documentation | 6 | 4 |
| 6. Programming | 5 | 4 |
| 7. Testing | 3 | 6 |
| 8. Installation | 1 | 5, 7 |

## Pert diagram




## Gantt timeline