Name: Autumn Austin   Mark _____/50

1. Brief introduction __/3

My feature for Gallion Gambit is the enemy creation and combat system.

From a dataset of enemies, an enemy will be chosen based on the level the player is at. Once the enemy is created, it will alternate turns with the player, choosing between status effects to give the player, and attacking it and dealing damage. If the player's health goes to 0, it results in a game over. If the enemy's health goes to 0, the player will receive awards and continue.

2. Use case diagram with scenario __14
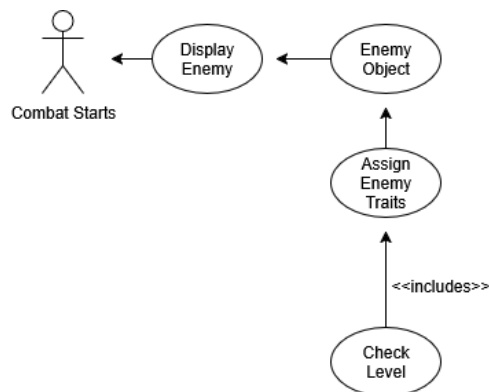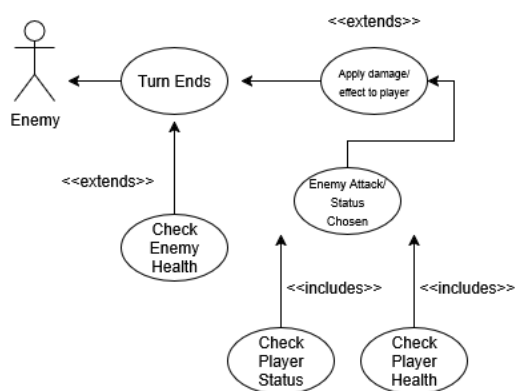
Use Case Diagrams

Diagram 01

Diagram 02

Enemy Scenarios

Name: Create Enemy (Diagram 01)

Summary: The enemy the player will have to fight will be chosen.

Preconditions: Player has started combat.

Basic sequence:

> **Step 1:** The level of the player will be read.
>
> **Step 2:** Based on the player level, the enemy will be chosen from a data set. The enemy's traits will be assigned by the data set.
>
> **Step 3:** The enemy will be displayed.

Exceptions:

> **Step 3:** If the player has won, the enemy will be hidden and a transisiton to the next scene will be triggered.
>
> **Step 3:** If the player has lost, the enemy will be hidden and a transisiton to the next scene will be triggered.

Post conditions: Player can choose their loot.

Priority: 1*

ID: C01

Name: Enemy Turn Behavior (Diagram 02)

Summary: What will happen once it's the enemy's turn in combat.

Preconditions: Enemy has been created and the enemy's turn has started.

Basic sequence:

> **Step 1:** The health of the player will be read.
>
> **Step 2:** The health of the enemy will be read.
>
> **Step 3:** The status effects of the player will be read
>
> **Step 4:** The enemy will choose between an attack or a status effect to the player.
>
> **Step 5:** The effect or damage will be applied to the player.
>
> **Step 6:** The enemies turn will end.

Exceptions:

> **Step 1:** If the player's health is 0, the enemy will not take their turn, and instead transisiton into the lose scene.
>
> **Step 2:** If the enemy's health is 0, the enemy will not take their turn, and instead will transition into the post match screen.
>
> **Step 3:** If the player already has a status effect, the enemy won't apply anpther one.
>
> **Step 4:** If the enemy chooses an attack, it will be read from their attacks in the enemy data set.
>
> **Step 4:** If the enemy chooses a status effects, it will be read from the effects in the enemy data set.
>
> **Step 4:** If the player already has an active status effect, the enemy will default to attacking.
>
> **Step 5:** If damage is selected and the player's health goes to 0 or below, the scene will transisiton to the lose screen.

> **Post conditions:** The game will transition to the screen that corresponds with the result of the battle.

> **Priority:** 2*
>
> **ID:** C01

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

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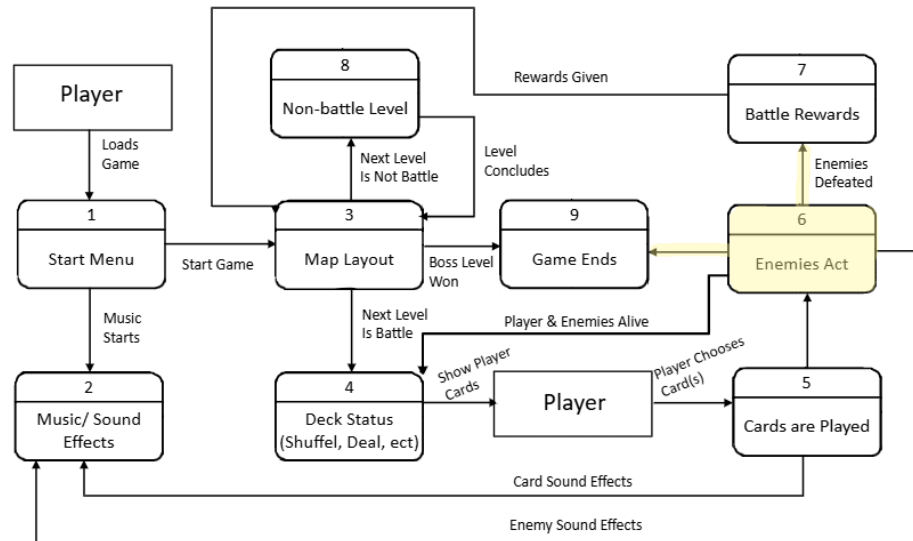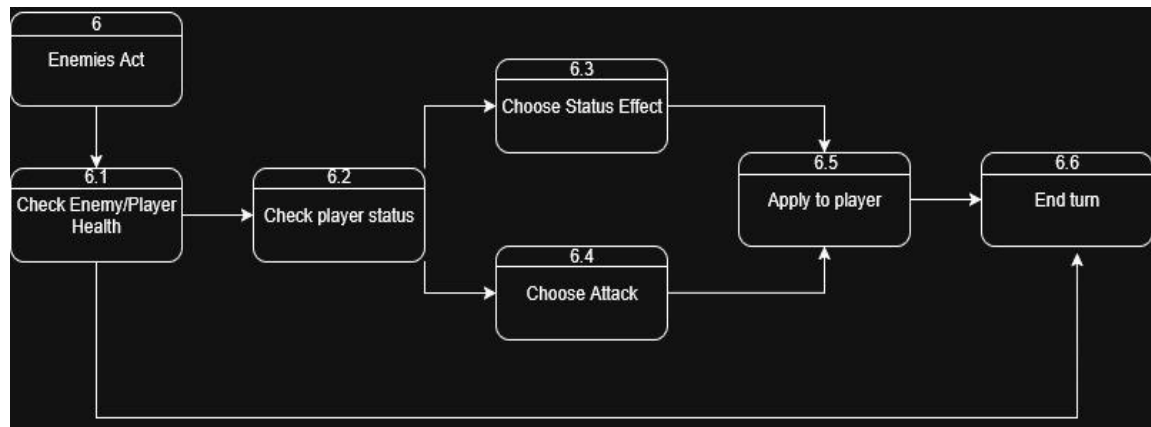
Data Flow Diagrams
Diagram 0

Diagram 6

Process Descriptions

As the player navigates though each level, it will alternate between battle levels and non-battle levels. On battle levels, the enemies will spawn based on the level the player is at. Once we reach the Enemies Act, the enemy will check it's own health, and the players health in order to see if it takes its turn. After checking health, it will check the status effect of the player. Based on that information, the effect/damage will be applied to the player, and the enemy will end its turn.

4. Acceptance Tests _____9

The enemy actions are all based around the status of the player, so we'll need to focus on how the enemy will read that input from the user. If the player doesn't have a status effect, the enemy will apply one. If the player has a status effect, then the enemy won't apply another one. The enemy will also need to not take its turns if it or the player has 0 or less health. So, we'll need to test each of these scenarios in order to ensure the player is alright.

NOTE: There shouldn't be a case where both the player AND the enemy's health is below 0, since their turns will end immediately once the opponent's health is below 0. If this does happen, we'll default to a player win since the enemy was defeated.

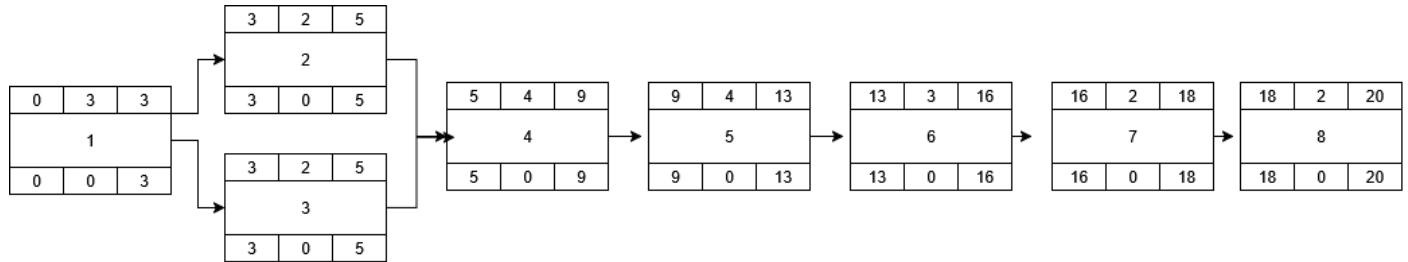| Player Input	Enemy Input	Result
Health > 0 AND No Status	Health > 0	Enemy should apply a status effect to the player, and end it's turn.
Health > 0 AND Has Status	Health > 0	Enemy will use an attack, and end it's turn.
Health <= 0 AND Has Status	Health > 0	Enemy will end its turn, and lose screen will be displayed.
Health <= 0 AND Has No Status	Health > 0	Enemy will end its turn, and lose screen will be displayed.
Health > 0 AND has status	Health <= 0	Enemy will end its turn, and win screen will be displayed.
Health > 0 AND has no status	Health <= 0	Enemy will end its turn, and win screen will be displayed.
Health <=0	Health <=0	Enemy will end its turn and the win screen will be displayed.

5. Timeline _____/10

Example:

Work items

Task	Duration (Hrs)	Predecessor Task(s)
1. Create Enemy Data (movesets, power level)	3	-
2. Create Enemy function	2	1
3. Create system for picking enemies based on level	2	1
4. Develop enemy's turn behavior	4	1, 2, 3
5. Test all enemy's turn behavior	4	1, 2, 4
6. Reevaluate leveling and enemy balancing	3	1, 4, 5
7. Install	2	6
8. Test with player function	2	7

Pert diagram

Gantt timeline