Name\_\_Augustine Almanza\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Mark \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/50

## Brief introduction \_\_/3

My main feature in the Alpine Chronicles Videogame is to make a demo mode for the game that runs any time there is not a user playing the game.

When the game is loaded onto a machine my job is to make sure that a game demo mode loads automatically after a given amount of idle time. In this state, the game will “play” itself by having the player character fight enemies, pick up items, etc.

My secondary responsibilities are to create and implement non-player characters (NPCs) and Enemies for the player to interact with or fight.

When an enemy is spawned in, it will have some sort of idle state before it notices or “spots” a player. This will be determinant of the proximity the player is to the enemy. Once a player has been “spotted” the enemy will attack the player (depending on the type of enemy). When an enemy is defeated, it will drop a given item. This could be anything from an item that boosts the player’s stats or increases the player’s score.

My main feature is the demo mode of the game, and it will be demonstrated in the diagrams below. I will not be including an NPC/Enemy feature in my champion doc as it is a secondary feature.

## Use case diagram with scenario \_\_14

### Use Case Diagrams

Diagram

Description automatically generated

### Scenarios

**Scenario 1 (Use Case Diagram):**

**Name:** Demo Mode

**Summary:** When the game is left idle at the main menu, demo mode loads automatically after a given amount of time. In this state, the game will “play” itself by having the player character fight enemies, pick up items, etc.

**Actors:** Player.

**Preconditions:** The game has been booted on a machine.

**Basic sequence:**

**Step 1:** The main menu is left idle by the player for a given amount of time. This is stored in a timer.

**Step 2:** Once the given amount of time for the timer has run out, the game will begin to load a scene.

**Step 3:** Once the scene is loaded, the AI or computer will take control of the player character, an enemies will begin to spawn.

**Step 4:** The computer will task the player character with fighting enemies and using items. Enemies will continue to spawn endlessly. It is not expected for the player character to survive past 5 minutes in demo mode.

**Step 5:** The player character is killed, and the main menu is loaded.

**Step 6:** The system will repeat Step 1.

**Exception 1:**

**Step 1:** The player selects a main menu option.

**Post conditions:** Demo mode will not be loaded.

**Exception 2:**

**Step 1:** The player presses the ‘esc’ key.

**Step 2:** Step 5 of the basic sequence is repeated.  
**Post conditions:** None.

**Exception 3:**

**Step 1:** The player character is not killed after 5 minutes in demo mode.

**Step 2:** Step 5 of the basic sequence is repeated.  
**Post conditions:** None.

**Priority:** 1\*

**ID:** DM1

\*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

### Data Flow Diagrams

Text, application, table

Description automatically generated with medium confidence

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

### Process Descriptions

Initialize Enemy:

WHILE Player is in given Level.

Set enemy Type.

Set enemy class.

END WHILE

Spawn Enemy:

WHILE enemy information is set.

Place a new enemy in the scene at given positions.

END WHILE

Control Enemy:

IF Player is a given distance away from enemy

Attack Player

END IF

ELSE

Move Enemy

END ELSE

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

Randomly redistribute classes

END WHILE

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

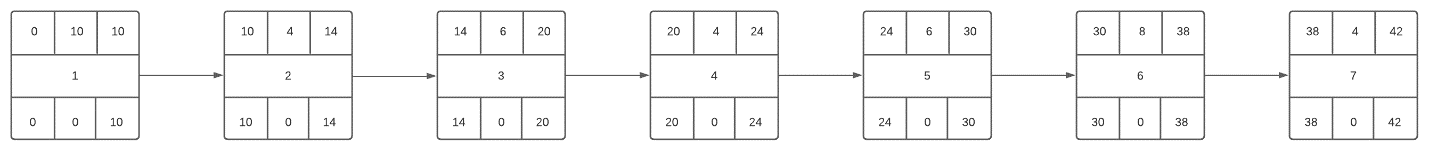
Initialize Enemy information 1000 times for each enemy configuration and record the spawning process in a separate file. Each enemy configuration should be different per level/scene.

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

### Work items

|  |  |  |
| --- | --- | --- |
| Task | Duration (PHrs) | Predecessor Task(s) |
| 1. Research | 10 | - |
| 2. Basic NPC/Enemy | 4 | 1 |
| 3. Art/Animation | 6 | 2 |
| 4. Attack System | 4 | 3 |
| 5. Movement | 6 | 4 |
| 6. Demo Mode | 8 | 5 |
| 7. Advanced Attack | 4 | 6 |

### Pert diagram



### Gantt timeline

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |