Name Ryan Wagner Mark \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/50

## Brief introduction \_\_/3

I have two main features I would like to focus on. First, I am going to oversee gui menus such as the main menu and pause menu. Finally, I will oversee level design. There will be four main levels.

## Use case diagram with scenario \_\_/14

### Use Case Diagrams

Diagram

Description automatically generated

Diagram

Description automatically generated

### Scenarios

**Name:** New Game

**Summary:** The player starts a new game.

**Actors:** Player

**Preconditions:** Game has been initialized.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Load scene 1.

**Exceptions:**

**Step 1:** A save-file already exists: prompt for overwrite.

**Post Conditions:** Scene 1 is loaded.

**Priority:** 1

**ID:** MM01

**Name:** Load Game

**Summary:** The player loads previously saved game.

**Actors:** Player

**Preconditions:** Game has been initialized, save state exists.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Load saved scene.

**Exceptions:**

**Step 1:** No save data exists: display no save data exists.

**Post Conditions:** Load saved game state.

**Priority:** 3

**ID:** MM02

**Name:** Options

**Summary:** The player accesses the options menu.

**Actors:** Player

**Preconditions:** Game has been initialized.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Load options menu

**Step 3:** Allow the modification of any option values within bounds.

**Step 4:** Save changes and return to main menu.

**Exceptions:**

**Step 1:** Player attempts to set setting outside of bounds: setting defaults to relative extreme and bounds are printed to screen.

**Post Conditions:** Write settings to file.

**Priority:** 2

**ID:** MM03

**Name:** Help Menu

**Summary:** The player accesses the help menu.

**Actors:** Player

**Preconditions:** Game has been initialized.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Load help menu.

**Step 3:** Upon close menu trigger, return to main menu.

**Post Conditions:** Return to main menu.

**Priority:** 2

**ID:** MM04

**Name:** Quit Game

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

**Actors:** Player

**Preconditions:** Game has been initialized.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Exit.

**Post Conditions:** Exit.

**Priority:** 2

**ID:** MM05

**Name:** Resume

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

**Actors:** Player

**Preconditions:** Game has been initialized.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Set timescale to 1.

**Post Conditions:** Timescale set to 1.

**Priority:** 1

**ID:** PM01

**Name:** Restart Level

**Summary:** The player reloads current level.

**Actors:** Player

**Preconditions:** Game has been initialized, player in level.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Load scene.

**Exceptions:**

**Step 1:** Not in level: do nothing.

**Post Conditions:** Load scene.

**Priority:** 3

**ID:** PM02

**Name:** Options

**Summary:** The player accesses the options menu.

**Actors:** Player

**Preconditions:** Game has been initialized.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Load options menu

**Step 3:** Allow the modification of any option values within bounds.

**Step 4:** Save changes and return to pause menu.

**Exceptions:**

**Step 1:** Player attempts to set setting outside of bounds: setting defaults to relative extreme and bounds are printed to screen.

**Post Conditions:** Write settings to file.

**Priority:** 2

**ID:** PM03

**Name:** Help Menu

**Summary:** The player accesses the help menu.

**Actors:** Player

**Preconditions:** Game has been initialized.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Load help menu.

**Step 3:** Upon close menu trigger, return to pause menu.

**Post Conditions:** Return to pause menu.

**Priority:** 2

**ID:** PM04

**Name:** Quit Game

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

**Actors:** Player

**Preconditions:** Game has been initialized.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Exit.

**Post Conditions:** Exit.

**Priority:** 2

**ID:** PM05

**Name:** Load Level

**Summary:** The player loads a level.

**Actors:** Player

**Preconditions:** Game has been initialized.

**Basic Sequence**

**Step 1:** Take button input.

**Step 2:** Load selected scene.

**Exceptions:**

**Step 1:** Save data flag: loads based on save data instead of default.

**Step 2:** If any save data lacks enemies/chests/lighting report none.

**Post Conditions:** Scene is loaded.

**Priority:** 1

**ID:** LM01

**Name:** Spawn Enemies

**Summary:** The level manager spawns the enemies.

**Actors:** LevelManager

**Preconditions:** Called by LevelManager.

**Basic Sequence**

**Step 1:** Spawn enemies at locations given by LevelManager.

**Exceptions:**

**Step 1:** LevelManager reports spawn none: spawn none.

**Post Conditions:** None

**Priority:** 1

**ID:** LM02

**Name:** Spawn Loot Chests

**Summary:** The LevelManager spawns the loot chests

**Actors:** LevelManager

**Preconditions:** Called by LevelManager.

**Basic Sequence**

**Step 1:** Spawn chests at locations given by LevelManager

**Step 2:** Randomize loot based on rarity given by LevelManager

**Exceptions:**

**Step 1:** LevelManager reports spawn none: spawn none.

**Post Conditions:** None

**Priority:** 1

**ID:** LM03

**Name:** Spawn Lighting

**Summary:** The LevelManager spawns the lighting.

**Actors:** LevelManager

**Preconditions:** Called by LevelManager.

**Basic Sequence**

**Step 1:** Spawns lights at locations given by LevelManager

**Exceptions:**

**Step 1:** LevelManager reports spawn none: spawn none.

**Post Conditions:** None

**Priority:** 1

**ID:** LM04

**Name:** Cutscene Manager

**Summary:** The LevelManager calls the cutscene manager.

**Actors:** LevelManager

**Preconditions:** Called by LevelManager

**Basic Sequence**

**Step 1:** Load Cutscene called by LevelManager

**Step 2:** After cutscene, return control to player.

**Post Conditions:** Exit.

**Priority:** 2

**ID:** LM05

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

### Data Flow Diagrams

Diagram

Description automatically generated

### Process Descriptions

Main Menu:

Display Main Menu

New Game:

Load Scene 1

Load Game:

Load Saved Game from saved state in saved data

Pause Menu:

Display Pause Menu

Resume Game:

Set timescale to 1

Close pause menu

Restart Level:

Reload current scene

Set timescale to 1

Close pause menu

Options:

Load the options pane from file

Save any changes

Write changes to file

Return to previous menu

Help Menu:

Display the help menu

Quit Game:

Quit the game

LoadLevel:

Takes data from Level file

Loads tiles and walls for level

Loads other sprites

Calls Spawn Lighting

Calls Spawn Enemies

Calls Spawn Loot Chests

Calls Cutscene Manager

Spawn Enemies:

Spawns enemies at locations given by LoadLevel

Sets behaviour based on data given by LoadLevel

Spawn Loot Chests:

Spawns loot chests at locations given by LoadLevel

Randomizes content by rarity given by LoadLevel

Spawn Lighting:

Spawns lighting at locations given by LoadLevel

Sets brightness given by LoadLevel

Cutscene Manager:

Loads Cutscene given by LoadLevel

Spawns any sprites necessary

Runs cutscene

Deletes any sprites unnecessary

Gives player control

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

**Button Test**

All button functionalities will be tested by attempting to use each button for its purpose.

Due to the functionality of the buttons, the results will be output to a file based on a debug option in the script. Each result will be styled in the form of XMXX – Successful where the X’s are replaced by the ID of the button.

**LevelManager Test**

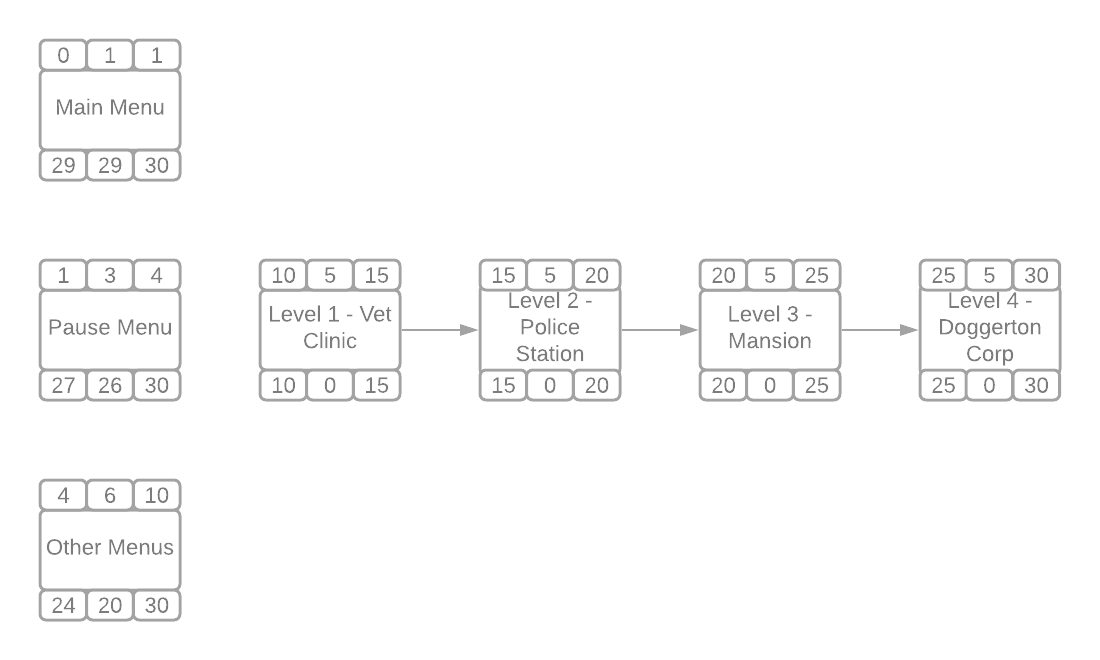
The functionality of the LevelManager will be tested by including a debug option to provide verification that each object was spawned successfully. The output of all objects attempted to be spawned will be output to a file, and then after spawning is complete, the current state of all objects in the game will be appended to the file. The data will be compared for any discrepancies.

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

### Work items

|  |  |  |
| --- | --- | --- |
| Task | Duration (Hrs) | Predecessor Task(s) |
| 1. Main Menu | 1 | - |
| 2. Pause Menu | 3 | - |
| 3. Other Menus | 6 | - |
| 4. Level 1 – Vet Clinic | 5 | - |
| 5. Level 2 – Police Station | 5 | 4 |
| 6. Level 3 - Mansion | 5 | 5 |
| 7. Level 4 – Doggerton Corp | 5 | 6 |

### Pert diagram



### Gantt timeline

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 5 |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 6 |  |  |  |  |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |