

## 1. Brief introduction \_\_/3

My Feature for the game "Corsair Clash" is the pause function for battles.

When a battle encounter starts my feature is supposed to pause the battle simulation in its tracks. When a keyboard button is pressed, my feature should set all objects and their time deltas to 0. Both Character objects and projectile assets should have their time delta set to 0. Thus creating a Pause effect on all assets. Ensuring that all no progress in the battle will be made.

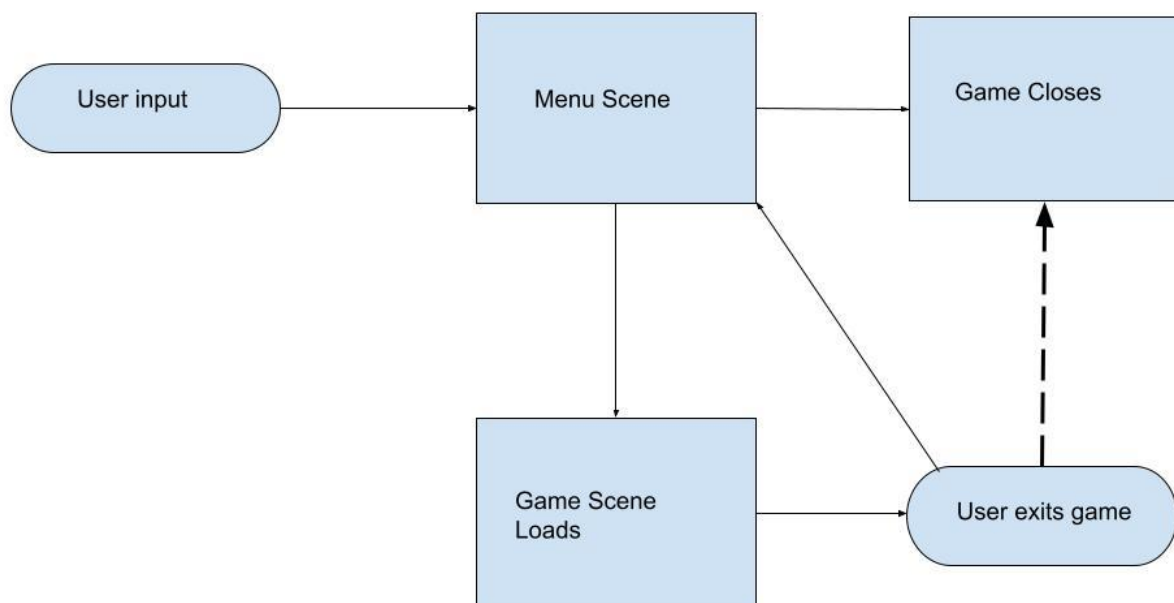
I am also responsible for the Main Menu interphase. When a user selects play on the menu screen, it should direct them to the main game.

The last feature I am responsible for is the health bar and the variable associated with it. When an enemy fires its projectiles against the player. The health bar needs to react to the damage taken and give a visual representation of the player's current health.

## 2. Use case diagram with scenario \_14

### Use Case Diagrams

#### Main Menu Use Case Diagram



**Name:** Main Menu

**Summary:** Initiates the main menu and loads the Game or Closes application

**Actors:** Player

**Preconditions:** Game File has been executed

**Basic sequence:**

**Step 1:** Mouse clicks on button panel that represents game start

**Step 2:** Game Loads

**Step 3:** User clicks exit game button panel in game, is brought to Main Menu screen again.

**Step 4:** Application closes

**Exceptions:**

**Step 1:** User Force quits application from outside application means

**Step 2:** User is not brought to the main menu screen, application closes.

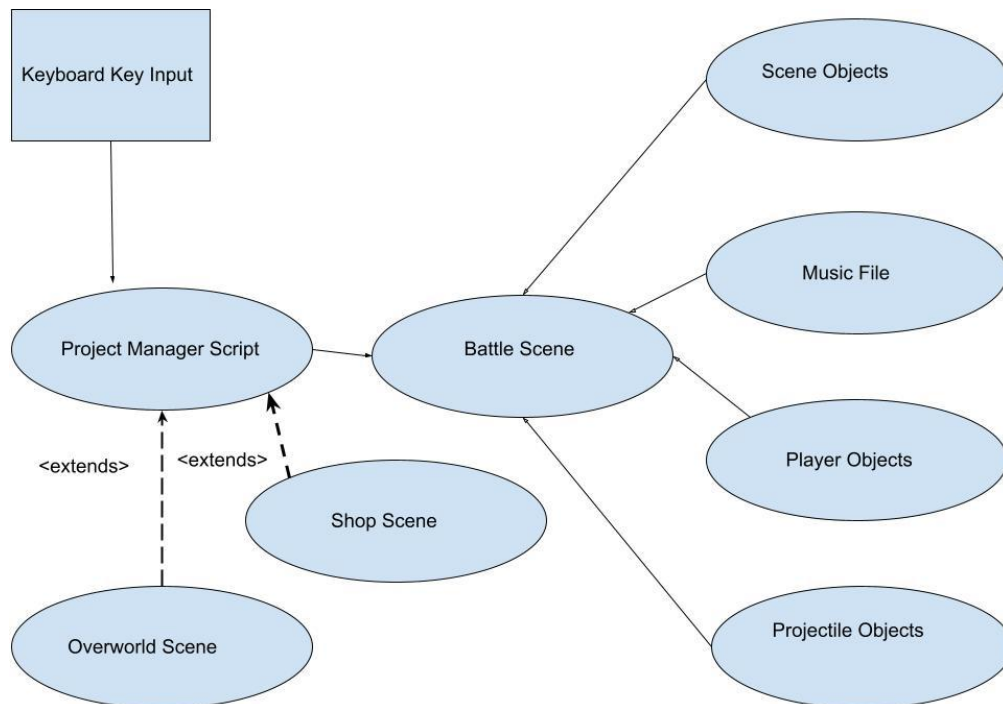
**Post conditions:** Game application has closed.

**Priority:** 2\*

**ID:** M1

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

Pause Feature:



**Name:** Pauses

**Summary:** Pauses the the movement and actions of all Objects in Scene

**Actors:** Player

**Preconditions:** Game File has been executed

**Basic sequence:**

**Step 1:** Keyboard input designated for pause button is pressed

**Step 2:** All assets on screen have stopped

**Step 3:** Pressing same key again will unpause on screen assets

**Step 4:** Objects resume movement and speed

**Exceptions:**

**Step 1:** Pause key pressed during overworld Scene

**Step 2:** Pause key is pressed during shop scene

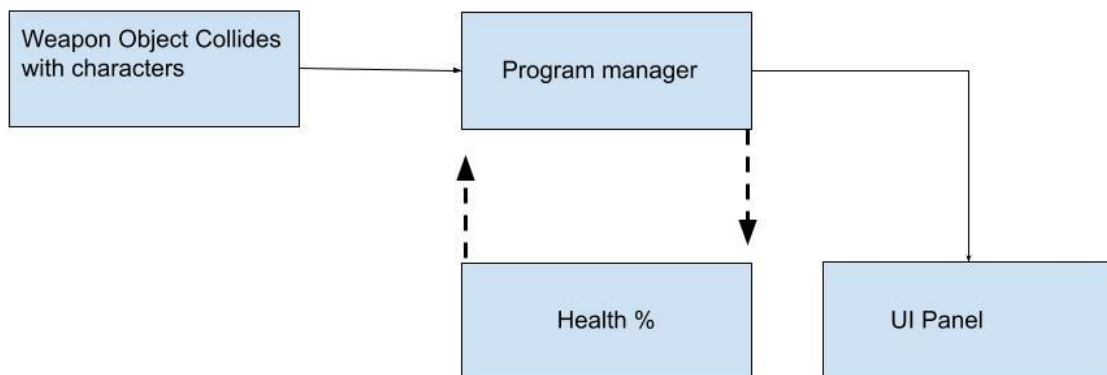
**Post conditions:** X amount of time has passed in the real world, no in game changes have occurred..

**Priority:** 2\*

**ID:** P1

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

## Health Bar Use Case:



**Name:** Health Bar

**Summary:** Adjusts The UI panel To depict an accurate health gage for players

**Actors:** Enemy AI

**Preconditions:** Battle Scene must be loaded

**Basic sequence:**

**Step 1:** Enemy projectile collides with Player

**Step 2:** Health bar depletes in accordance to the damage points from enemy projectile

**Step 3:** Health Bar reaches (current\_health <= 0)

**Step 4:** Battle scene ends

**Exceptions:**

**Step 1:** Player health variable is not properly set

**Step 2:** Player never gets hit

**Post conditions:** Health Bar Panel has decreased OR scene ends

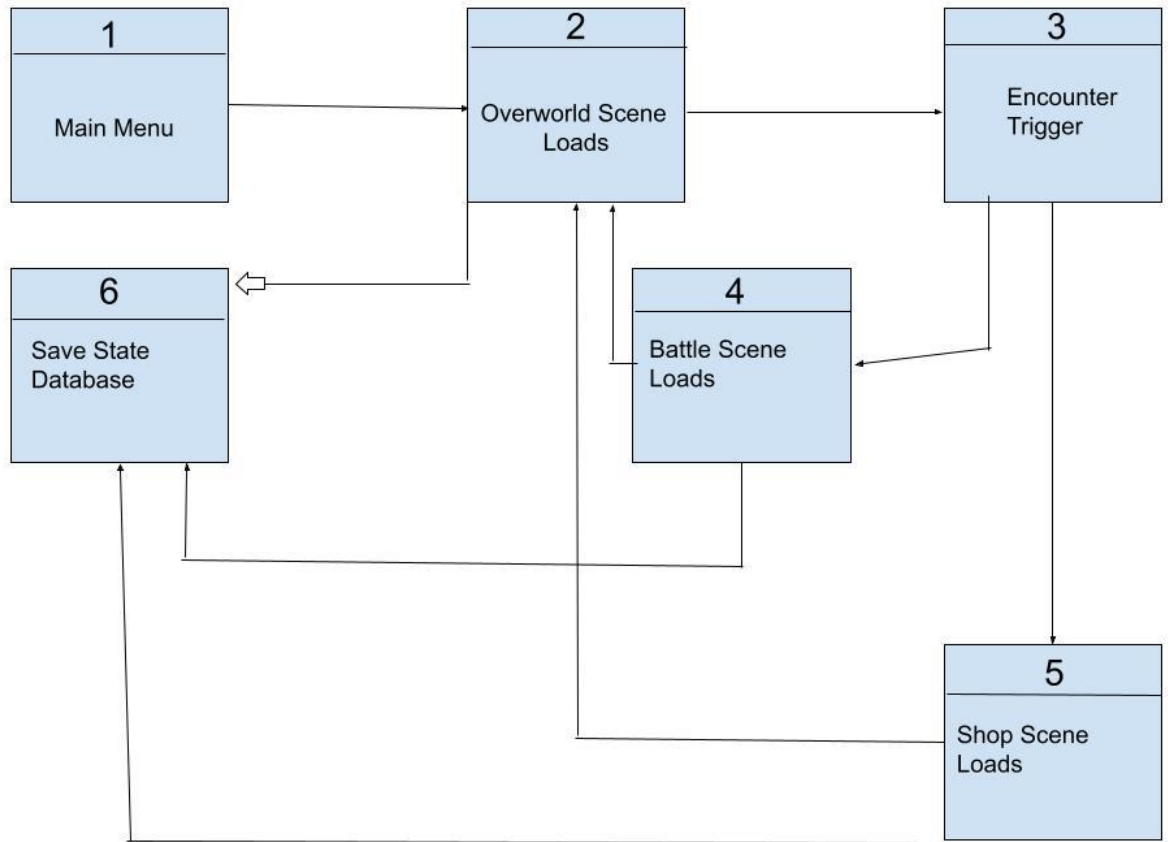
**Priority:** 1\*

**ID:** HB1

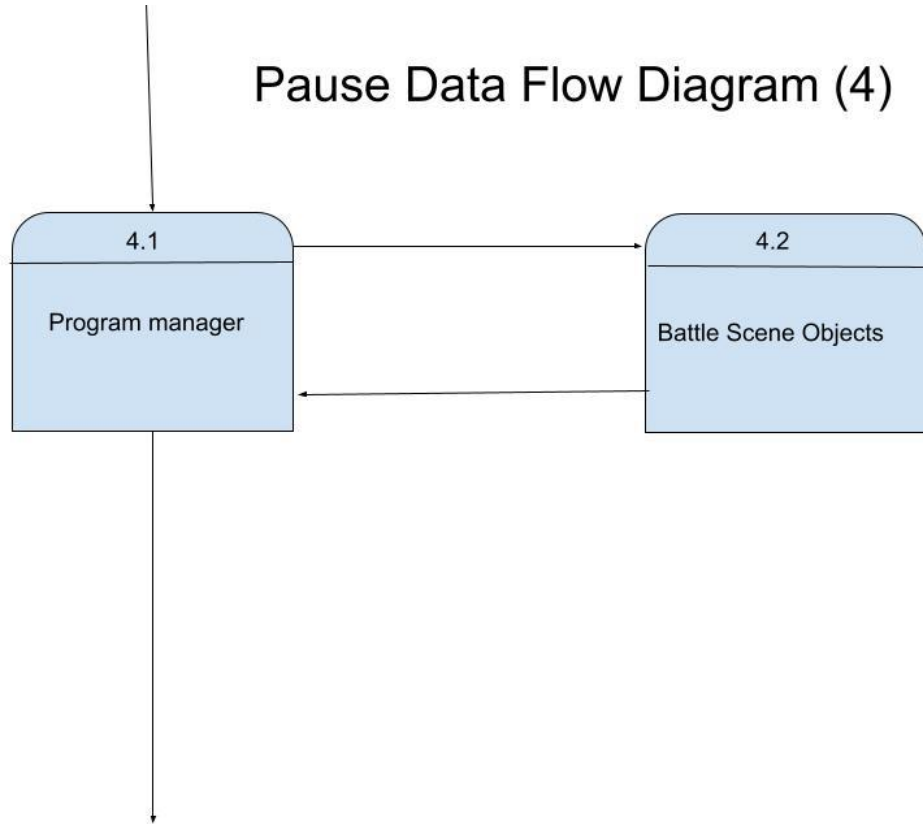
The priorities are 1 = must have, 2 = essential, 3 = nice to have.

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

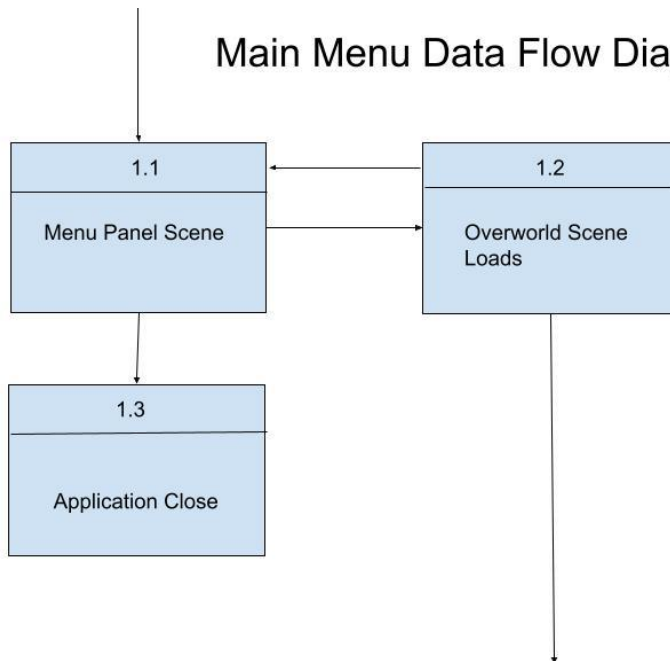
Data Flow SCENE 0:



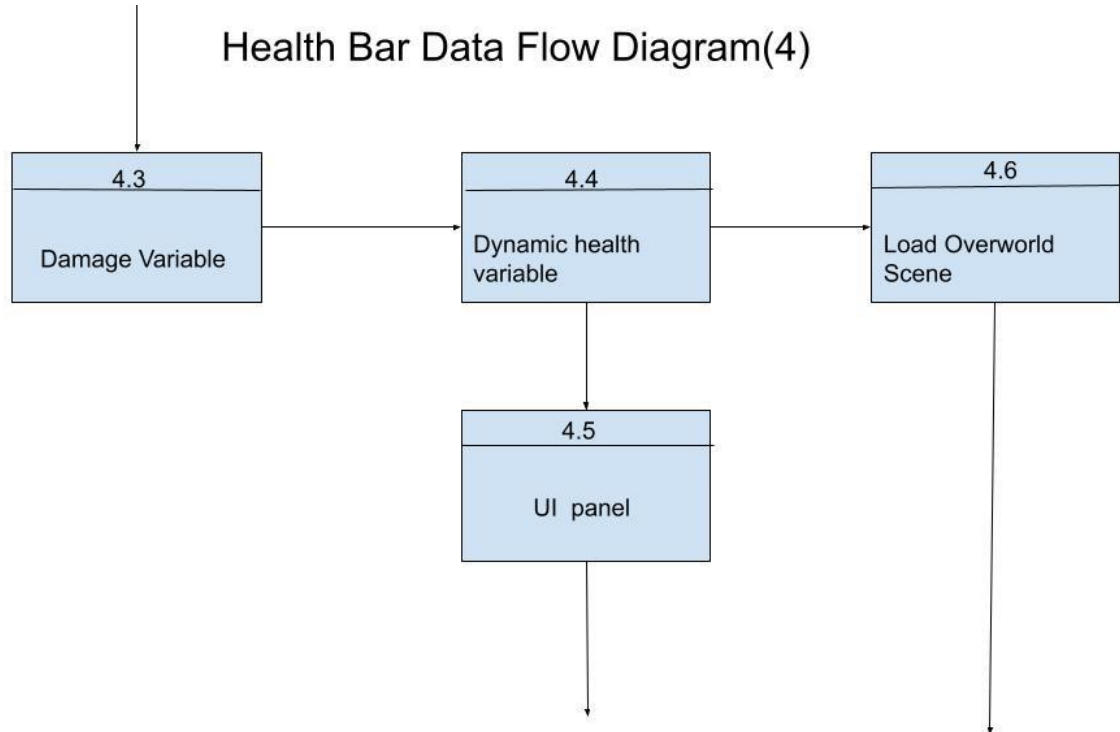
Pause Data Flow Diagram (4)



Main Menu Data Flow Diagram (1)



### Health Bar Data Flow Diagram(4)



#### Process Descriptions

Pause Feature\*:

time flag variable = false (not paused)

WHILE game is running

IF time flag = False, meaning game is not paused

THEN pause if conditionals are cycled

time flag variable = true, meaning the game is now paused

IF current scene = Battle Scene, when key is pressed

then all objects and characters movement are paused.

IF Current Scene = Overworld Scene, when key is pressed

then game objects stay in motion, nothing is paused

IF Current Scene = Shop Scene, when key is pressed

then game objects stay in motion, nothing is paused

IF time flag = true, meaning game is paused

THEN

IF current scene = Battle Scene, game objects are now unpaused, and resume set speed and motion prior

END WHILE

Main Menu Feature\*

WHILE game application is running

IF current scene = Main Menu AND user input from mouse clicks on Menu Panel "play"

```

THEN
    load overworld game Scene
IF current scene= Overworld AND user clicks on exit button panel
THEN
    Load Main Menu Scene
IF current scene = Main Menu AND user clicks on exit button panel
THEN
    close application
END WHILE

```

Health Bar Feature\*

```

WHILE current scene = battle scene
    IF enemy_projectile_collison = player
    THEN
        health = health - enemy_damage
        health panel = health - enemy_damage
    IF health <= 0
        loss flag triggered
        loss counter variable updated
        load overworld scene
    END WHILE

```

#### 4. Acceptance Tests \_\_\_\_\_9

##### Acceptance Test: Health Bar

Test damage counter variable on health counter

The output will have the following characteristics

- Player health = 10
- Max enemy damage = 10
- Min enemy damage = 1
- Number of enemies = 4

Health bar test output:

| Test# | Damage | Player Health | Result                           |
|-------|--------|---------------|----------------------------------|
| 1     | 1      | 9             | Health = High                    |
| 2     | 5      | 5             | Health = Medium                  |
| 3     | 7.5    | 2.5           | Health = Critical                |
| 4     | 10     | 0             | Player Death,<br>Overworld Loads |

### Acceptance Test: Main Menu

Explore MAIN MENU decision branches

The output will have the following characteristics

- 1->Menu -> play -> overworld exit -Menu -> close game
- 2->Menu → close game
- 3->Menu -> play -> Force quit

Pause Feature output:

- 1) branch sequence 1 explored intended use cases possibilities, the game closed properly from user input
- 2) branch sequence 2 explored intended use cases possibilities, the game closed properly from user input
- 3) branch sequence 3 explored unintended use case possibilities, the game did not close properly or circle back to the main menu.

### Acceptance Test: Pause Feature

Press pause key trigger with different intervals of time in between each press of the key..

The output file will have the following characteristics:

- Number of time key was pressed: 5
- Max time interval: 10 seconds
- Min time interval: 1 second
- Time tested is between a set of pause key inputs, a set is when you initially press the pause key. Which sets the pause flag to true, which is then followed by the action of pressing the pause key again, setting the pause flag to false. The game is now unpaused.

Pause Feature output:

| Test# | Time Interval (Seconds) | Objects in scene are successful paused |  |
|-------|-------------------------|--|--|
| 1     | 1                       | true                                   |  |



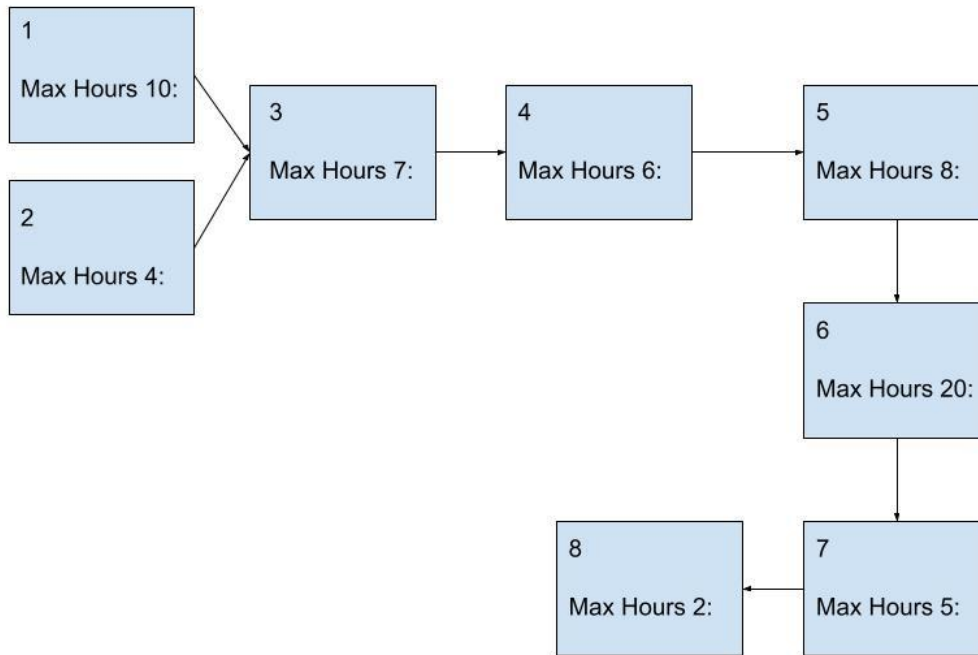
|   |     |      |  |
|---|-----|------|--|
| 2 | 2.5 | true |  |
| 3 | 5   | true |  |
| 4 | 7.5 | true |  |
| 5 | 10  | true |  |

## 5. Timeline \_\_\_\_/10

### Work items

| Task                          | Duration (PWks) | Predecessor Task(s) |
|-------------------------------|-----------------|---------------------|
| 1. Time_delta_stop function   | 4               | -                   |
| 2. Menu UI design             | 1               |                     |
| 3. Health Bar UI Master class | 3               | 2                   |
| 4. Health Bar UI sub class    | 3               | 2, 3                |
| 5. User Documentation         | 2               | 4                   |
| 6. Programming                | 9               | 1,2,3,4             |
| 7. Testing                    | 3               | 6                   |
| 8. Installation               | 1               | 5, 6,7              |

## Pert diagram



## Gantt timeline

