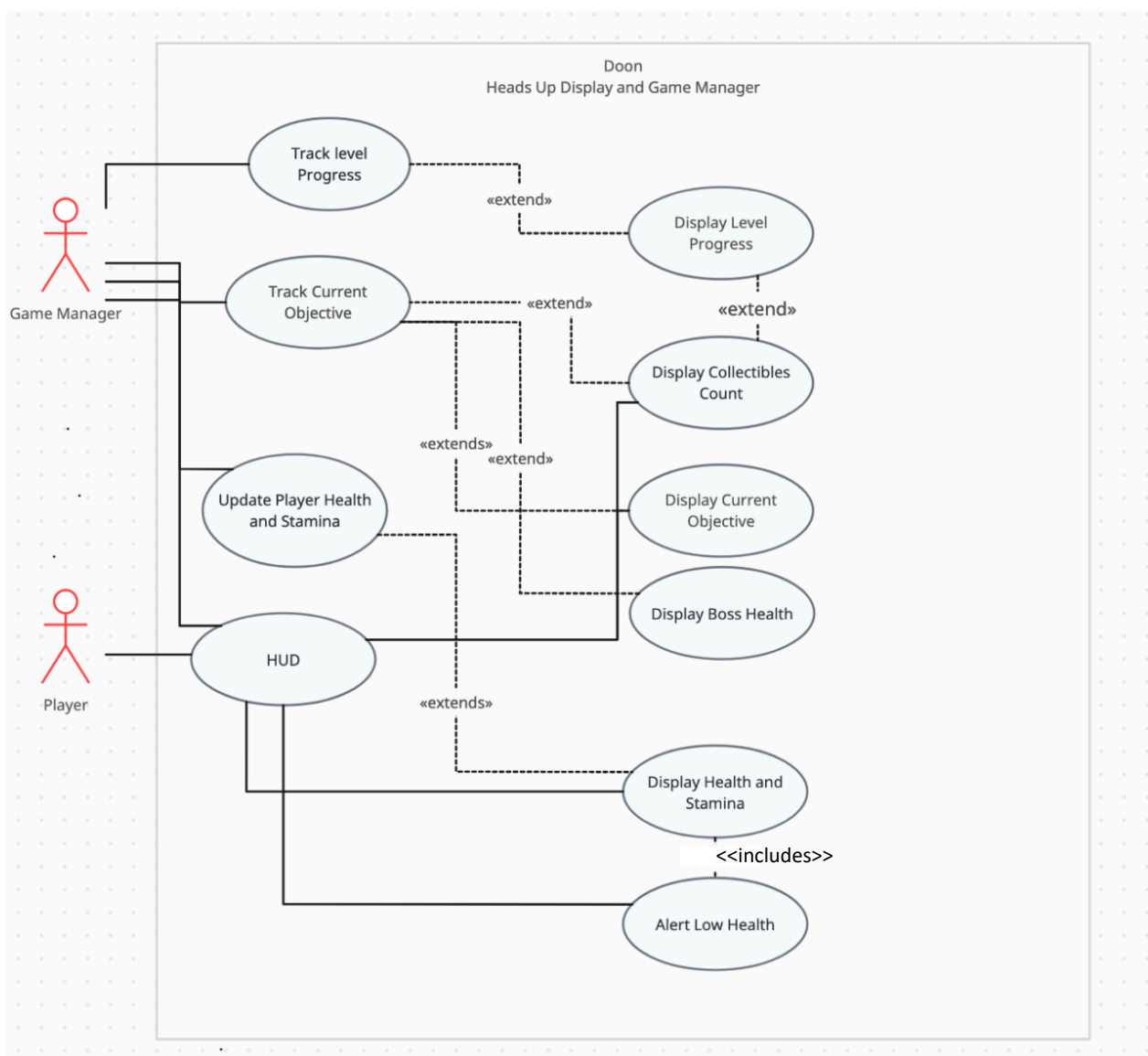


## 1. Brief introduction \_/3

The features I am responsible for in the development of Doon are the game manager and the HUD. The HUD is vital for displaying the necessary information to the player for an enjoyable and straightforward gaming experience. It will display the current objective, the players health and stamina, level progress, in levels with boss enemies, it will display their health, in levels with collectibles it will display how many are needed to complete the level. I will also be responsible for the game manager which will provide all necessary data for the HUD. This means managing and keeping track of the aforementioned variables in a neat and consistent manner.

## 2. Use case diagram with scenario \_14

### Use Case Diagrams



## Scenarios

**Name:** HUD

**Summary:** The player views the hud

**Actors:** Player and game manager.

**Preconditions:** Player and game manager have been instantiated

**Basic sequence:**

**Step 1:** The level is loaded

**Step 2:** Display the hud

**Post conditions:** The Hud is displayed

**Priority:** 2

**ID:** 01

**Name:** Display Health and Stamina

**Summary:** The player views Health and Stamina Inside the hud

**Actors:** Player and game manager.

**Preconditions:** Player, game manager and HUD have been instantiated

**Basic sequence:**

**Step 1:** The HUD is loaded

**Step 2:** The data is retrieved from the game manager

**Post conditions:** The Hud is displayed

**Priority:** 2

**ID:** 02

**Name:** Alert Low Health

**Summary:** The player is notified of low health in the HUD

**Actors:** Player and game manager.

**Preconditions:** Player, game manager, HUD have been instantiated

**Basic sequence:**

**Step 1:** The Health and stamina is loaded

**Step 2:** Display notification when health is low

**Post conditions:** The low health notification is displayed

**Priority:** 3

**ID:** 03

**Name:** Update player health and stamina

**Summary:** The player health and stamina is updated

**Actors:** game manager.

**Preconditions:** game manager has been instantiated

**Basic sequence:**

**Step 1:** The level is loaded

**Step 2:** The health and stamina is updated

**Post conditions:** The health and stamina is updated

**Priority:** 1

**ID:** 04

**Name:** Track current objective

**Summary:** The game manager keeps track of the current objective

**Actors:** game manager.

**Preconditions:** game manager has been instantiated

**Basic sequence:**

**Step 1:** The game manager is loaded

**Step 2:** The current objective is tracked

**Post conditions:** The current objective is tracked

**Priority:** 1

**ID:** 05

**Name:** Track level progress

**Summary:** The game manager keeps track of the current level progress

**Actors:** game manager.

**Preconditions:** game manager has been instantiated

**Basic sequence:**

**Step 1:** The game manager is loaded

**Step 2:** The current level progress is tracked

**Post conditions:** The current level progress is displayed

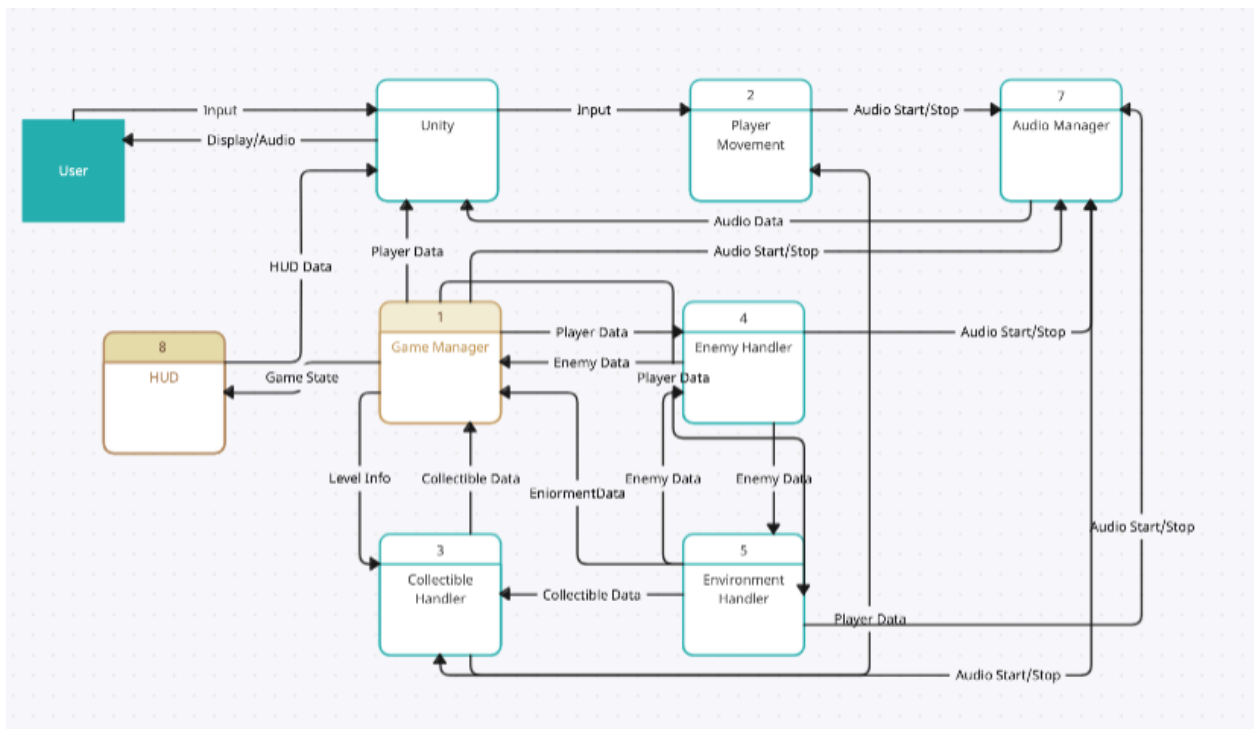
**Priority:** 1

**ID:** 06

### 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]

#### Data Flow Diagrams



## Process Descriptions

Game Manager

WHILE game running

Fetch data from Collectible handler, Environment Handler and Enemy Handler

Update with current data

Send data to HUD

END WHILE

HUD:

WHILE game running

Fetch data from Game Manager

Draw HUD overlay above game layer

IF initial startup or player input ESC

Draw menu over other layers

END IF

END WHILE

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

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

Make sure hud is displayed at all times and displaying the correct information at all times from the game manager.

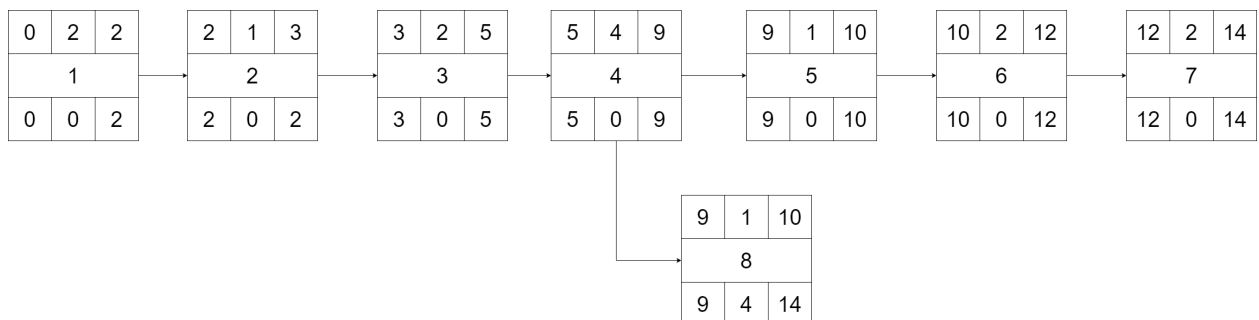
## 5. Timeline \_\_\_\_/10

[Figure out the tasks required to complete your feature]

### Work items

Task	Duration (Wks)	Predecessor Task(s)
1. Requirements Collection	2	-
2. System Breakdown	1	1
3. Section Design	2	2
4. Programming	4	3
5. Validation	1	4
6. Project Integration	2	5
7. Testing	2	6
8. Documentation	1	4

### Pert diagram



### Gantt timeline

