Name: Christopher Soto Mark _____/50

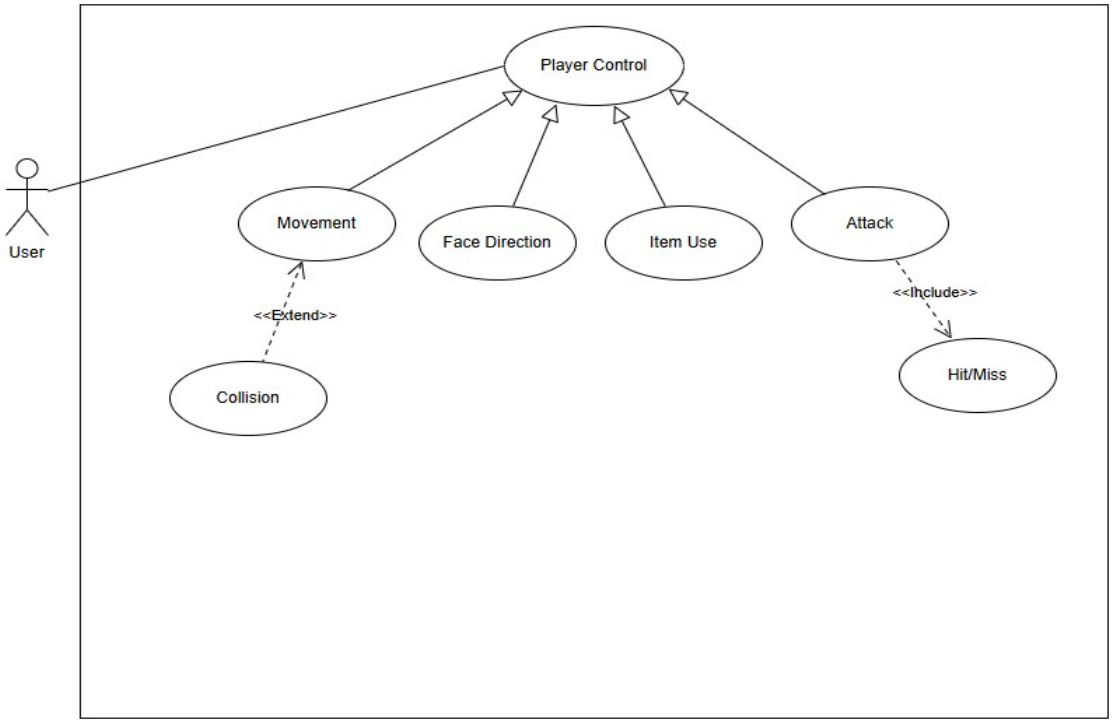
## 1. Brief introduction __/3

My feature for The Crawl videogame is the creating the player which includes class selection, controls, attacking, collisions.

When a player presses W A S or D their character will move up, down, left, or right across the screen and when player moves the mouse the character will face the direction of the mouse. This feature will change depending on the version of the videogame, if the game is on mobile then the movement and attacking is controlled by virtual joysticks on the screen. Additionally, when the player runs into a trap, wall, or enemy an event is played based on the scenario that is triggered. I am also responsible for creating different classes for the player to start as which determines the sprite the player will have for their character as well as changing their starting equipment.

## 2. Use case diagram with scenario __14

### Use Case Diagrams



## Scenarios

**Name:** Player Movement

**Summary:** The User inputs commands that moves player around the game environment.

**Preconditions:** Player has been initialized, and Game has started.

**Basic sequence:**

**Step 1:** Accept input from User.

**Step 2:** Determine command based on input.

**Step 3:** Game system attempts to update player position.

**Step 4:** Environment is checked for collisions.

**Step 5:** Player position is updated if valid.

**Exceptions:**

**Step 4:** If Wall collision is detected: player position is not updated in that direction.

**Step 4:** If a Trap is collided with: update player health with damage of trap.

**Post conditions:** Player position is updated based on input.

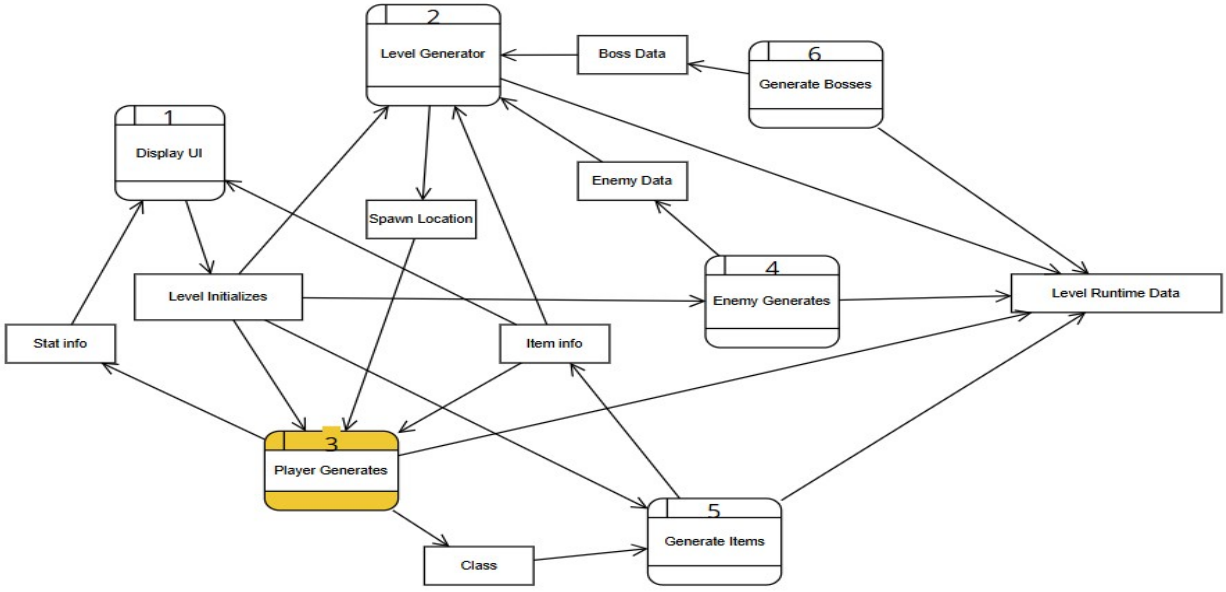
**Priority:** 1*

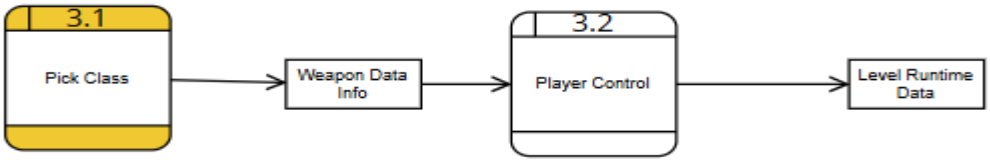
**ID:** C01

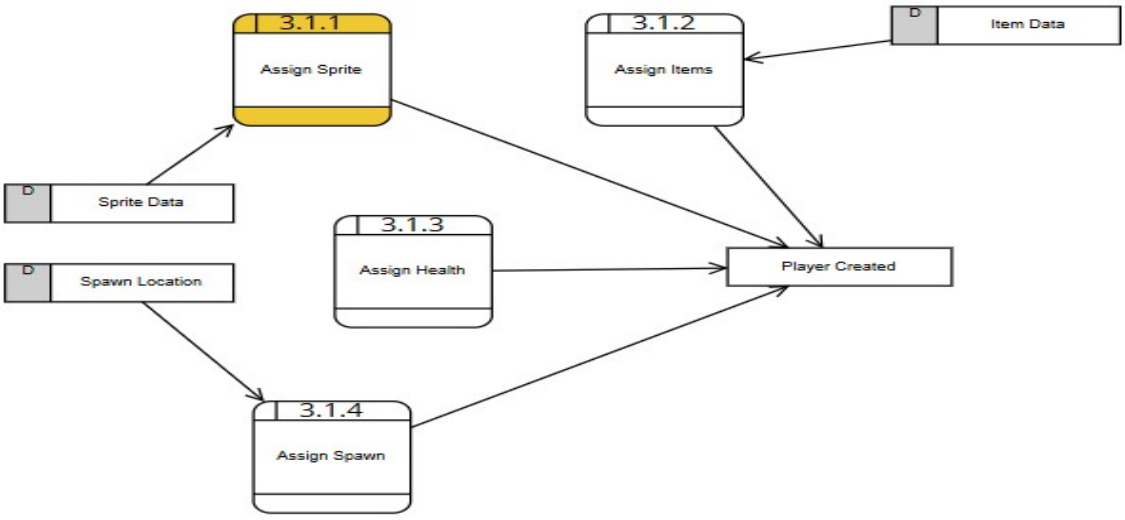
*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

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

### Data Flow Diagrams




## Process Descriptions

Assign Sprite*:

```
INPUT playerchoice
    IF playerchoice == “warrior” THEN
    FOR search sprite table for warrior
    ASSIGN player.sprite = warrior.png
    ELSE IF playerchoice == “archer” THEN
    FOR search sprite table for archer
    ASSIGN player.sprite = archer.png
    ELSE IF playerchoice == “mage” THEN
    FOR search sprite table for mage
    ASSIGN player.sprite = mage.png
    ELSE
    DISPLAY default warrior assigned due to error
    FOR search sprite table for warrior
    ASSIGN player.sprite = warrior.png
    ENDIF
```

## 4. Acceptance Tests _____9

**Example for Player Movement**

The test runs1000 times with the following characteristics:

- presses a movement key (WASD/arrow/joystick)
- player character should move smoothly in the input direction.
- Player character moves in the correct direction
- the movement should stop when input is released
- the player should not pass through walls or invalid areas
- player should be able to continue moving in other valid directions
- If player collides with trap the trap’s effect should be applied (e.g., damage, stun, slow)
- if the trap is disabled or the player has immunity, no effect should be applied

**Example for Player Attack**

The test runs1000 times with the following characteristics:

- the player clicks the mouse button

- the attack should launch in the direction of the mouse cursor relative to the player
- the attack should be visually aligned with that direction
- if the player is on attack cooldown, the attack should not trigger
- the attack hitbox overlaps with the target's hitbox
- damage should be applied to the target's health
- if the target is blocking or dodging, damage should be reduced or nullified
- if the hitboxes never overlap, no damage should be applied

**Example for Player Class Selection**

The test runs1000 times with the following characteristics:

- the player selects a class
- the player's sprite should update to match the class
- the correct weapon should be assigned
- the correct items should be assigned
- the game should not start until a valid class is selected
- if a locked/unavailable class is chosen, the system should prevent selection

## 5. Timeline _____/10

[Figure out the tasks required to complete your feature]

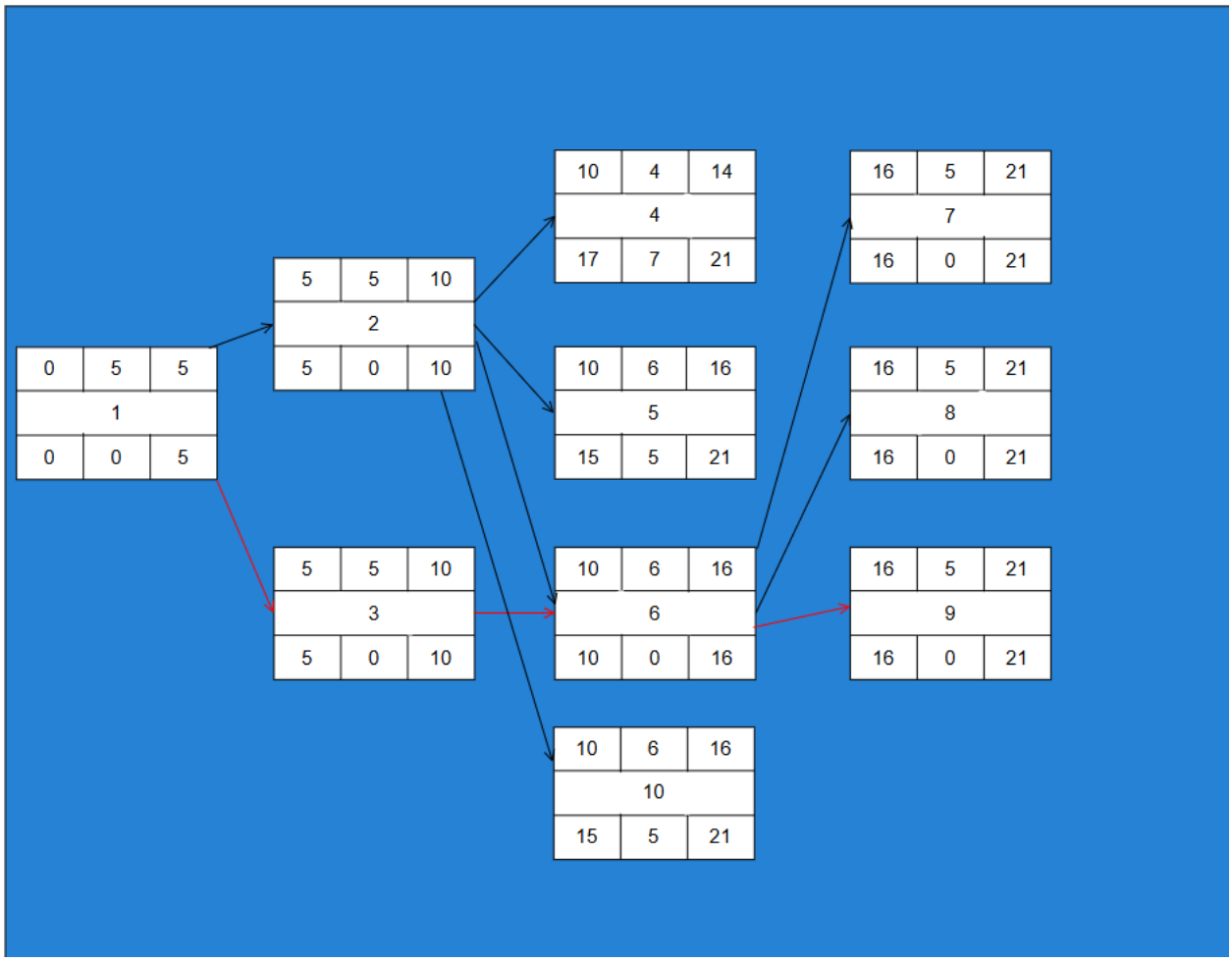
Example:

### Work items

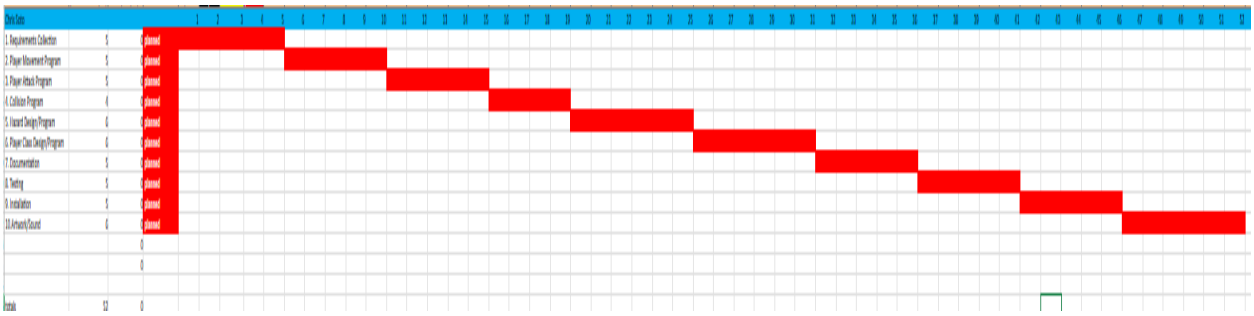
| Task | Duration (Hours) | Predecessor Task(s) |
|---|---|---|
| 1. Requirements Collection | 5 | - |
| 2. Player Movement Program | 5 | 1 |
| 3. Player Attacking Program | 5 | 1 |
| 4. Collision Program | 4 | 2 |
| 5. Hazard Design/program | 6 | 2 |

| | | |
|---|---|---|
| 6. Player Class Design/program | 6 | 2,3 |
| 7. Documentation | 5 | 6 |
| 8. Testing | 5 | 6 |
| 9. Installation | 5 | 6 |
| 10. Artwork/Sound | 6 | 2,3 |

## Pert diagram




## Gantt timeline


Task Name
1. Requirements Collection
2. Player Movement Program
3. Player Attack Program
4. Collision Program
5. Hazard Design Program
6. Player Class Design Program
7. Documentation
8. Testing
9. Installation
10.Artwork/Sound
Finish
planned