Name: Tyler LaFleur Mark _____/50

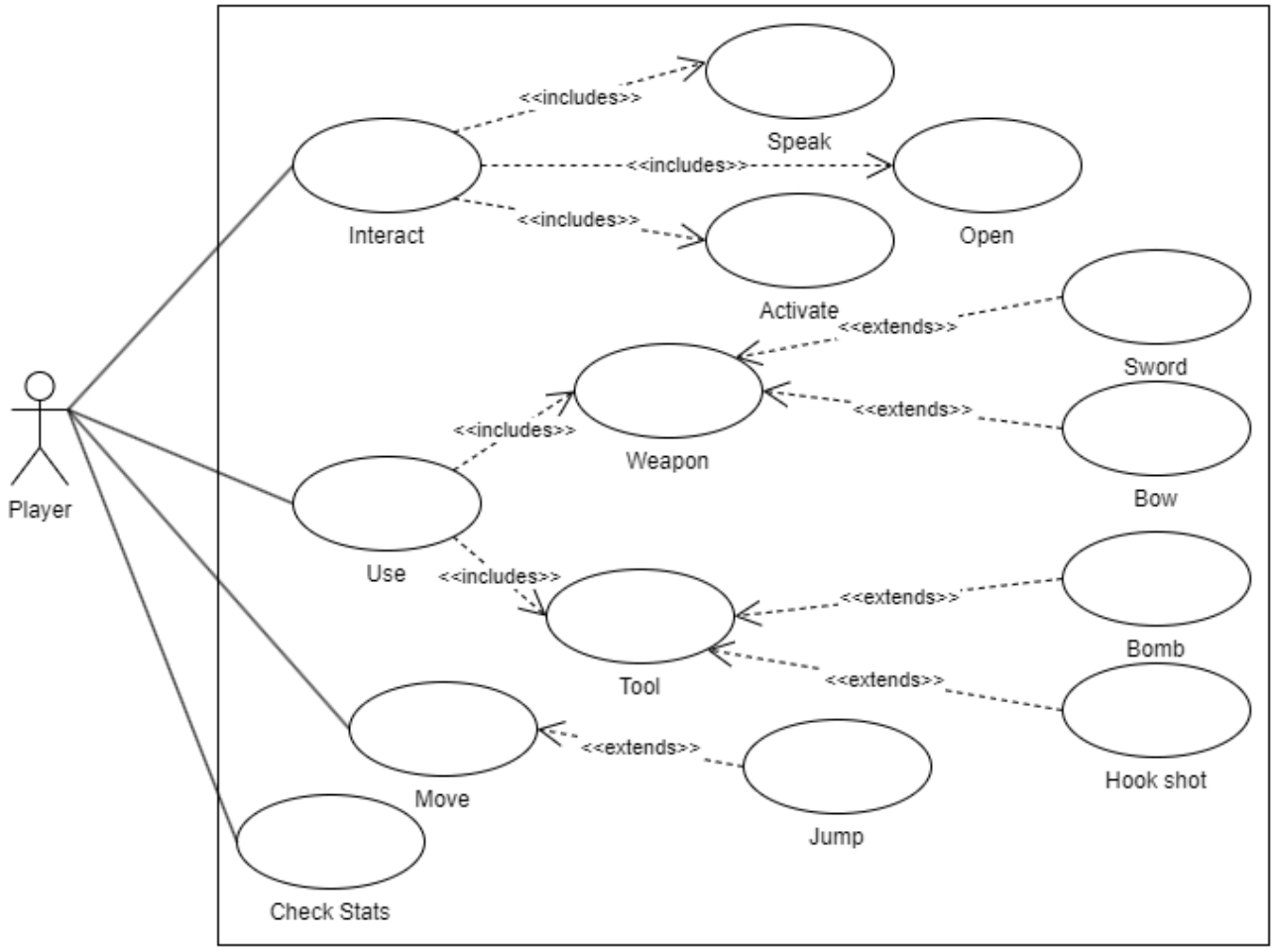
1. Brief introduction __/3

I will be working on making the Player Character(PC) as well as all related functionality for that player, such as movement, interacting with other objects, using tools, and stats.

The player will need to be able to move around, so I intend to add in omnidirectional movement. They need to be able to interact with the world around them, to include things such as opening treasure chests, activating switches, and talking with Non-Player Characters(NPCs). The player also needs to be able to use tools, such as swords and bows for combat, as well as utility tools like bombs or hook shots to be able to uncover hidden locations or travel around the world. Lastly, as they progress through the game, their stats will improve allowing for better performance.

2. Use case diagram with scenario __14

Use Case Diagram



Scenarios

Name: Weapon

Summary: The player uses a weapon to make an attack.

Actors: The player of the game.

Preconditions: Game is running and player has control of character.

Basic sequence:

Step 1: Check to see if a weapon is equipped.

Step 2: Execute attack.

Step 3: Check to see what is hit.

Step 4: Update world.

Exceptions:

Step 1: If no weapon is equipped, then skip the sequence.

Step 2.1: If a melee weapon is equipped, execute animation and check the immediate area for step 3.

Step 2.2: If a ranged weapon is equipped, then execute animation, create projectile, and loop steps 3 and 4 until the projectile hits something or exits bounds.

Post conditions: World is updated to reflect what has changed because of the attack.

Priority: 1*

ID: P01

*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

Diagram 0

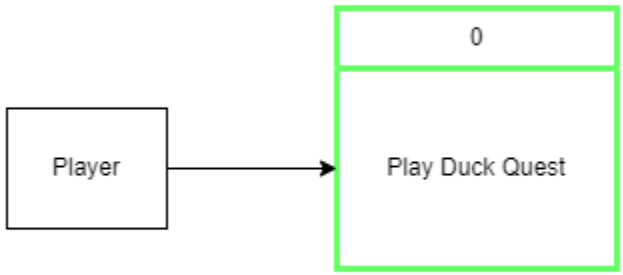


Diagram 1 Level 0

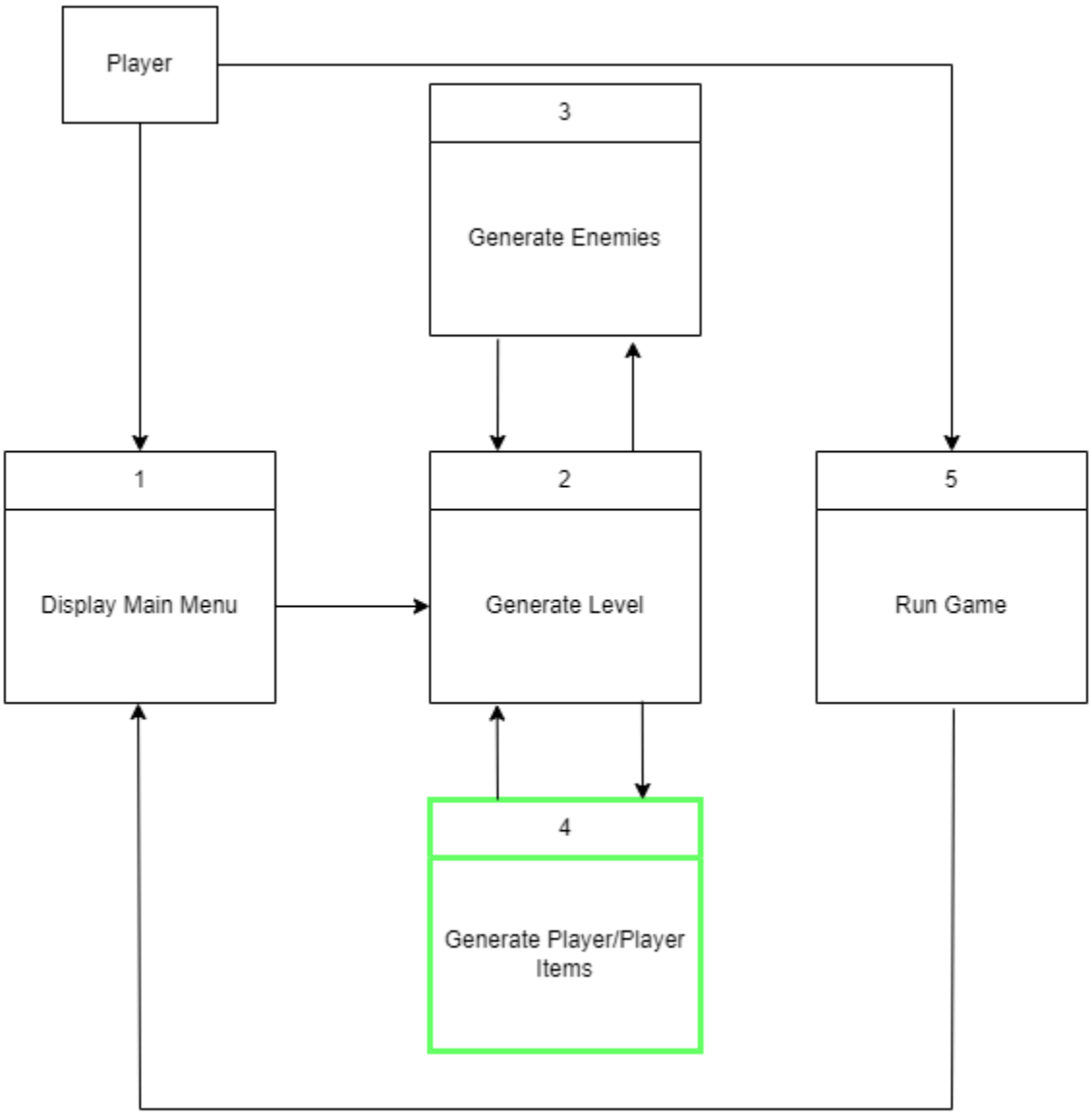


Diagram 2 (Player actions)

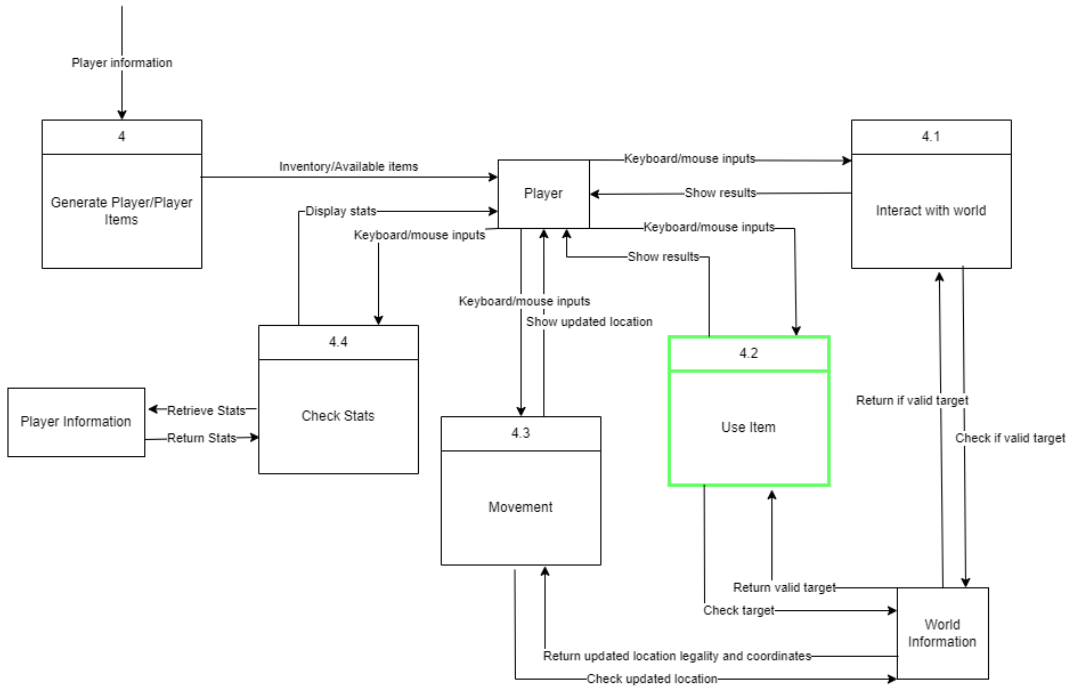
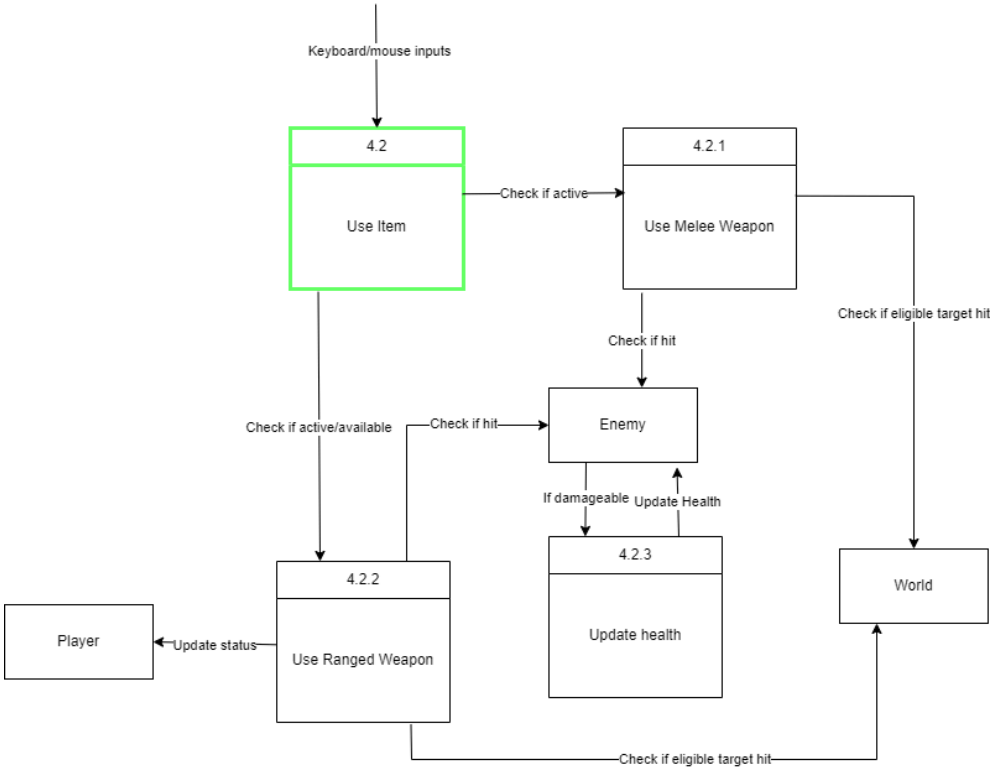


Diagram 3 (Use Item)



Process Descriptions

```
If player attacks then
    If melee weapon equipped then
        Perform animation
        Check collision
        Update game world
    Else If ranged weapon equipped then
        Perform animation
        Create projectile object
        Check projectile collision
        Update game world
    Else
        Do Nothing
    Endif
Endif
```

4. Acceptance Tests _____9

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

Test will input player actions to test if actions are able to be performed only under the correct circumstances and that damage and actions are only done to proper targets. These will include:

- Melee Attacks
- Ranged Attacks
- Interactions such as opening chests, talking with NPCs, and operating switches
- Tools only interact with expected actors (i.e. bombs explode breakable walls and only breakable walls)
- Hazards properly damage the player (and other actors if necessary)
- Stats properly update and reflect accordingly

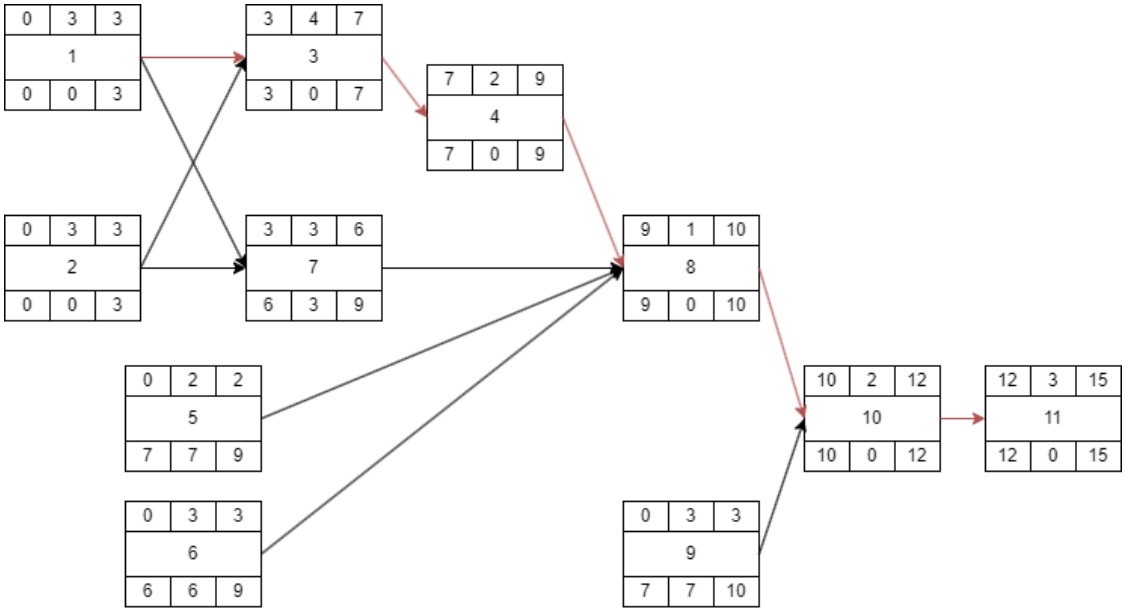
Test will output logs able to confirm whether or not things acted as expected. The test can also include a gauntlet that the PC can only properly progress through if the expected results are met otherwise it will not be completed. Player inventory and stats, as well as battle log can be exported so that it can be referenced.

5. Timeline _____/10

Work items

Task	Duration (hours)	Predecessor Task(s)
1. Create weapon types	3	-
2. Create tools	3	-
3. Sprite creation	4	1,2
4. Sprite animation	2	3
5. Implement movement functions	2	-
6. Implement interaction functions	3	-
7. Implement use functions	3	1,2
8. Link animations to functions	1	4, 5, 6, 7
9. Implement stat functions	3	-
10. User Documentation	2	8,9
11. Testing	3	10

Pert diagram



Gantt timeline

	1	2	3	4	5	6	7	8	9	10	11	12	11	14	15
1															
2															
3				1,2											
4								3							
5															
6															
7				1,2											
8										4,5,6,7					
9															
10											8,9				
11													10		