Name: Gibson Moseley Mark ____________________/50

1. Brief introduction __/3

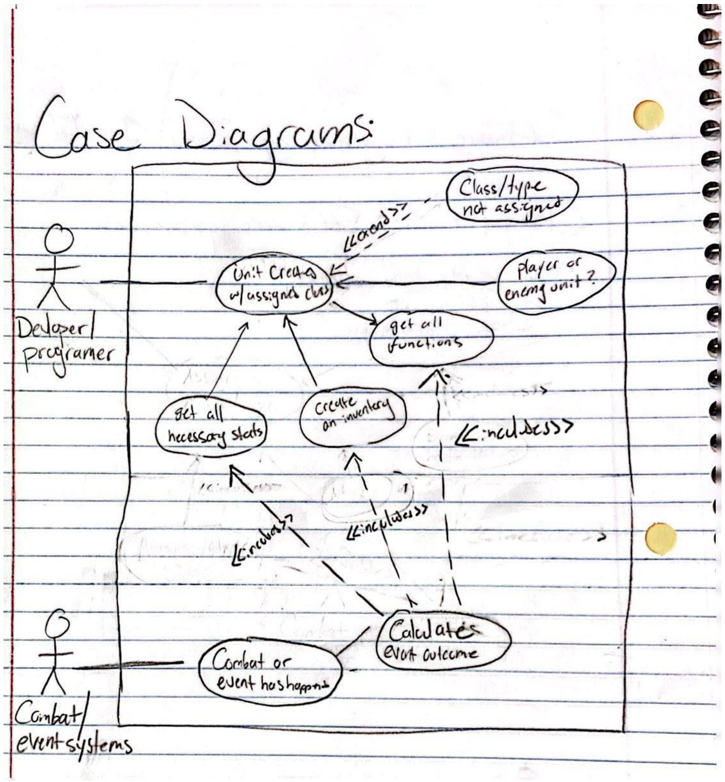
My Feature is to make all the unit classes so they can be assigned to unit objects in the game.

When a unit is wanted to be created I need to make sure that someone can apply my code to their game object and turn it into a unit now able for the player to control (I do not create the object or the logic, just the units themselves and their stats. This also includes all functions needed for other processes to work). So I need to add a way for people to differentiate classes and then give the unit the correct stats and attacks based on the class that was chosen. The important features of each class are movement, attacks, health, attack stats, luck, range, strengths, and weaknesses, and distinction between player and enemy units. On top of that, I will also need to add functions that allow a unit to take damage or be healed, and whatever other people will require to do their interactions properly.

This means I have to create Three categories: Stats (health, movement, etc.), Unit Distinction, and Interaction (taking damage, gaining life, etc.).

2. Use case diagram with scenario __14

Use Case Diagrams (IGNORE INVENTORY WE HAVE REMOVED IT)



Scenarios

Name: Create a Unit

Summary: When the person creates a unit we need to check what class/type they have been assigned and then apply all necessary stats, functions, and parameters needed for that unit.

Actors: Developer making the Unit

Preconditions: There is a game object the unit code has been assigned to.

Basic sequence:

Step 1: The Unit is created and is assigned a class/type

Step 2: Get the Stats and set the right value

Step 3: The Unit will gain All necessary functions

Step 4: Then set if the Unit will be controlled by Player or Enemy

Exceptions:

Step 1: No class/type has been assigned

Postconditions: the unit has all necessary requirements and functions of the game to work and for other systems to interact with them.

Priority: 1*

ID: GM1

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

Name: Giving Out Data

Summary: Other systems will need to access data from the units so some functions and variables will just be public. So

Actors: Systems or Developers (For example the combat system)

Preconditions: There is a Unit they are referencing.

Basic sequence:

Step 1: Call one of the public functions depending on what data you want or what you want to manipulate.

Step 2: Store or use the data as the system needs.

Exceptions:

Step 1: None

Postconditions: the unit has all necessary requirements and functions of the game to work and for other systems to interact with them.

Priority: 1*

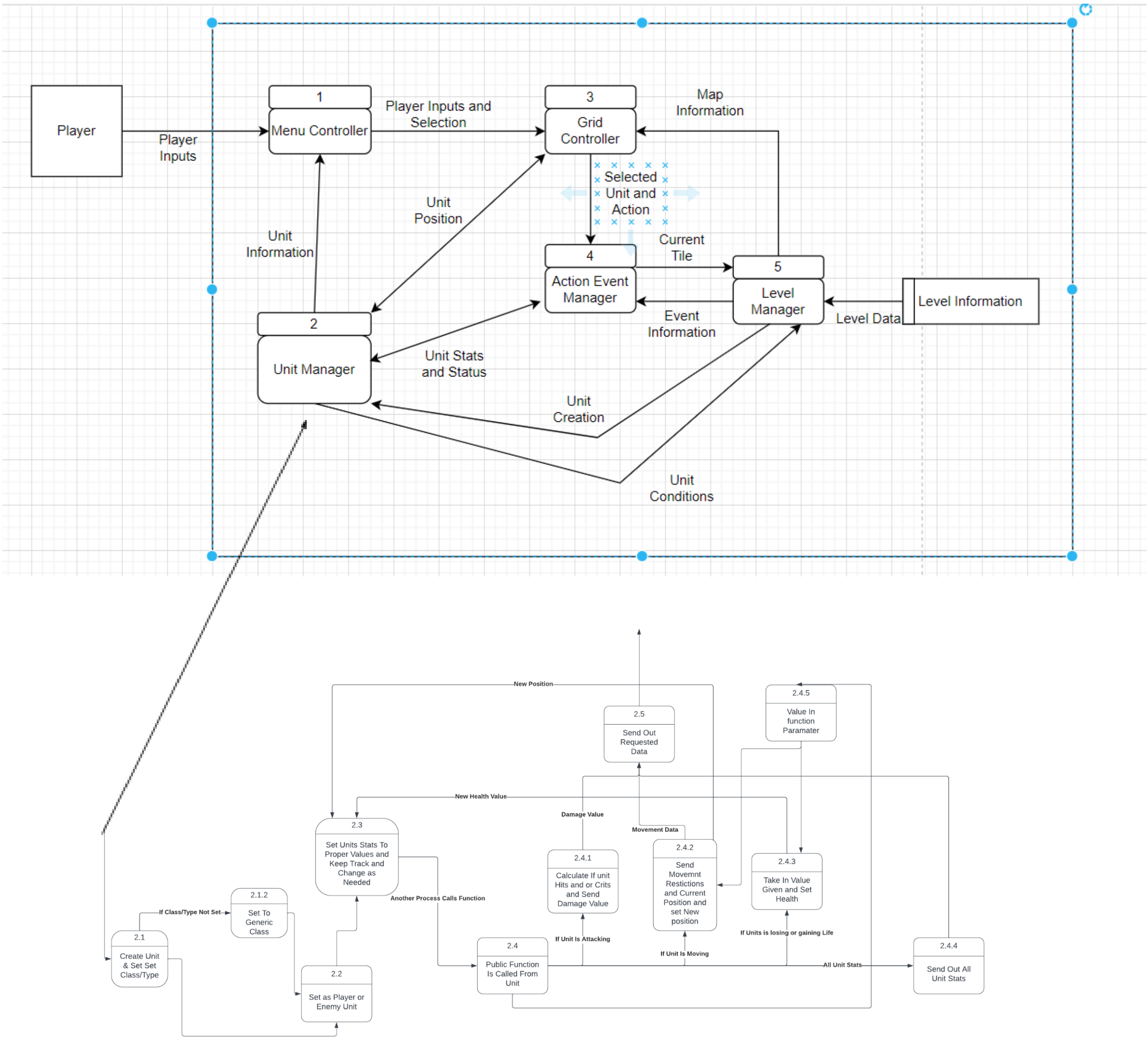
ID: GM2

*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

(Mine connects with 2 and outputs to 1-5).

Data Flow Diagrams



Process Descriptions

Set Values:

- This is when the class is set and all values need to be set accordingly. This will be just a simple check and change of values. I will have to set what attacks are available to the unit and the range of them. This is just a basic setup.

Attack:

- Each attack will have a public function that other systems can call on to find out how the attacks work if the unit wants to deal damage and so on.

Movement:

- This will be a function that will set their movement values to show how many tiles they use. It will also hold its current position on the board and update itself when the Unit moves. Allowing systems like the enemy AI to “see” where the other units are.

Lose/Gain of Life:

- This is just a function people can call to edit unit health if they take damage or gain life. They could lose life off being attacked or could gain life by healing on a tile. This will just make it easy to manipulate those health values at will.

4. Acceptance Tests _______9

Since all of the unit codes as I imagine them right now will all be contained in one script that will be assigned to objects, my plan to test all my features is to make multiple test scripts that will call each function and then see if the values or functions ran properly. For the attack function, I want to try that against another unit or call it on itself so I know we can differentiate different attacks. So the text script will be super simple. I'll just create a script that calls all the functions on a unit and then figure out what parameters need to be added. The output I’m looking for in the set values is that when I set the class for the Unit all values are being set correctly so I’ll have a function that prints them all out. For Attack, I should see numbers that correspond to what attack I call in the function and hope to see the correct values printed out including a critical hit and miss chance. My ideal output for the movement is that when I move the unit I store the new place on the grid and I don't allow them to move on to terrain tiles that they can’t move on. The output for the gain and lose life function is that I can change the health value to whatever I want and then can make the unit die or gain as much life as it should. As I don't know exactly what the code will look like I don't have inputs and outputs yet so Ideally I am just able to manipulate the values correctly.

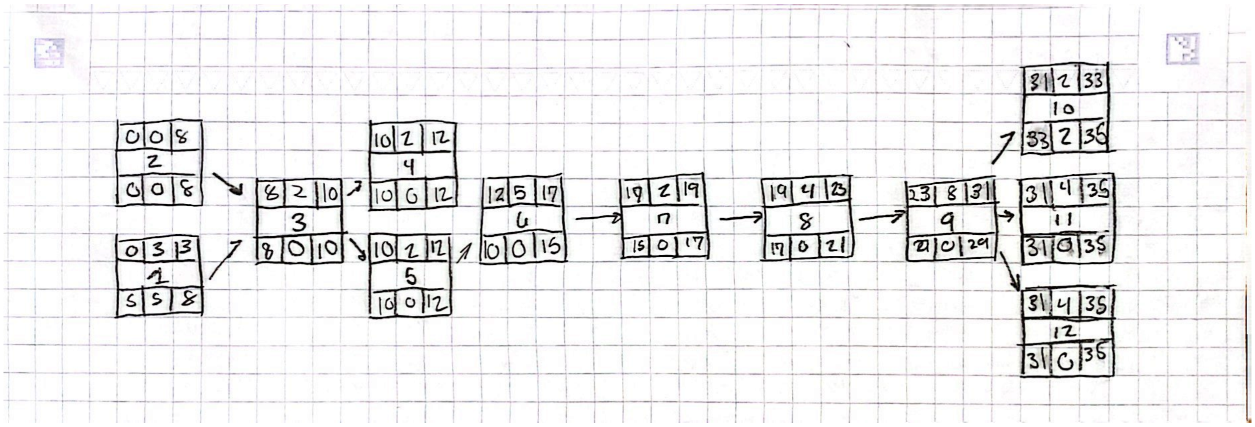
5. Timeline ________/10

Work items

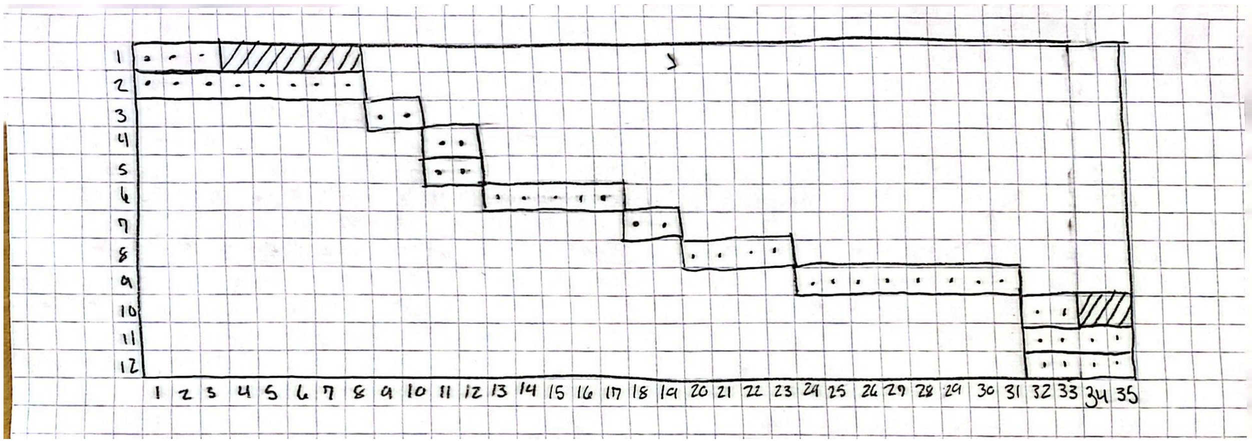
Task	Duration (PWks)	Predecessor Task(s)
1. Health	3	
2. Attacks	8	
3. Attack Damage Values	2	1,2
4. Miss & Crit Chance	2	2, 3,
5. Attack Range	2	2, 3

6. Movement and Restrictions	5	5
7. Player Units & Enemy Units Distinction	2	6
8. Base Class	4	7
9. Classes (5 as of now)	8	8
10. BC Mode	2	9
11. Unit Sprites	4	9
12. Unit Animation	4	9

Pert diagram



Gantt timeline



1
2
3
4
5
6
7
8
9
10
11
12
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 31 32 33 34 35