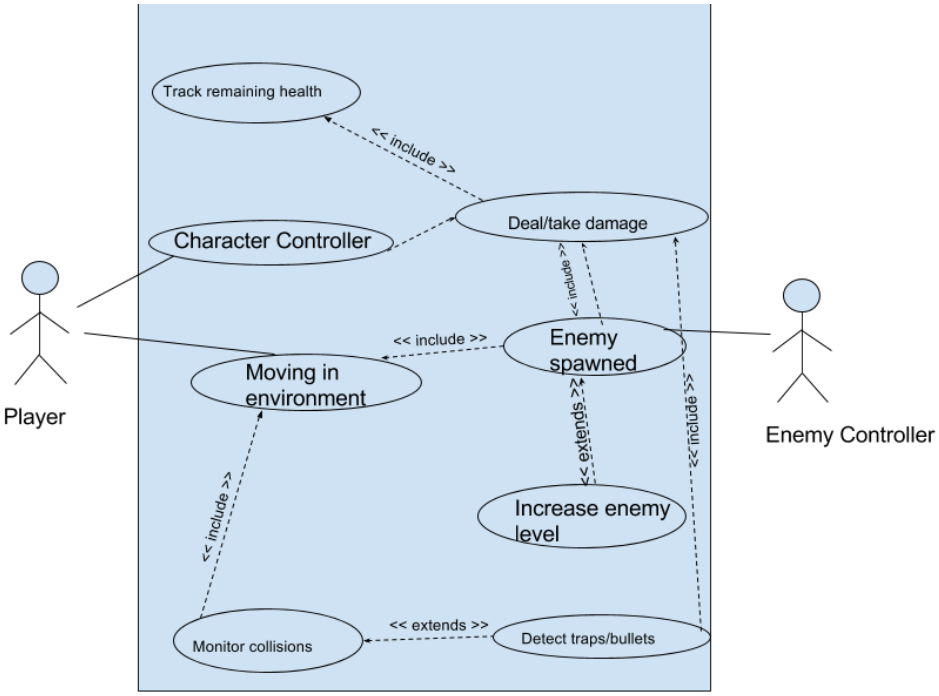
Name: Drew Stevens______________________ Mark ______________________/50

1. Brief introduction __/3

My feature that I will be creating will be the enemy AI system. This includes their movement, combat style, health, death, and spawning. This will have a prototype enemy and the different enemies based on wanted health, attack, defense, etc. in order to have different enemy types and keep the game interesting.

2. Use case diagram with scenario __14

Use Case Diagram



Scenario

Name: Maintaining enemies

Summary: The enemies that the user will interact with to gain score and to create an end game condition.

Actors: Player, Enemy Controller

Preconditions: UI has called upon the GameManager to create the environment to create the enemies and begin the game.

Basic sequence:

Step 1: Enemies are created and placed at locations requested.

- Game Manager invokes the Enemy Controller to spawn specific enemies.

Step 2: Enemy spawned is given base amount of health, damage, and armor based on difficulty.

Step 3: When player comes onto screen, move and chase player in environment.

Step 4: If collision with player, then deal damage to player and play a sound.

Step 5: If shot by player, take damage, play a sound and lose health.

- Track enemy health remaining

Step 6: If collision with trap, take damage, play a sound, and lose health.

- Track enemy health remaining

Exceptions:

Step 1: Player’s score passes a threshold, increase health, damage, and armor of enemies spawned.

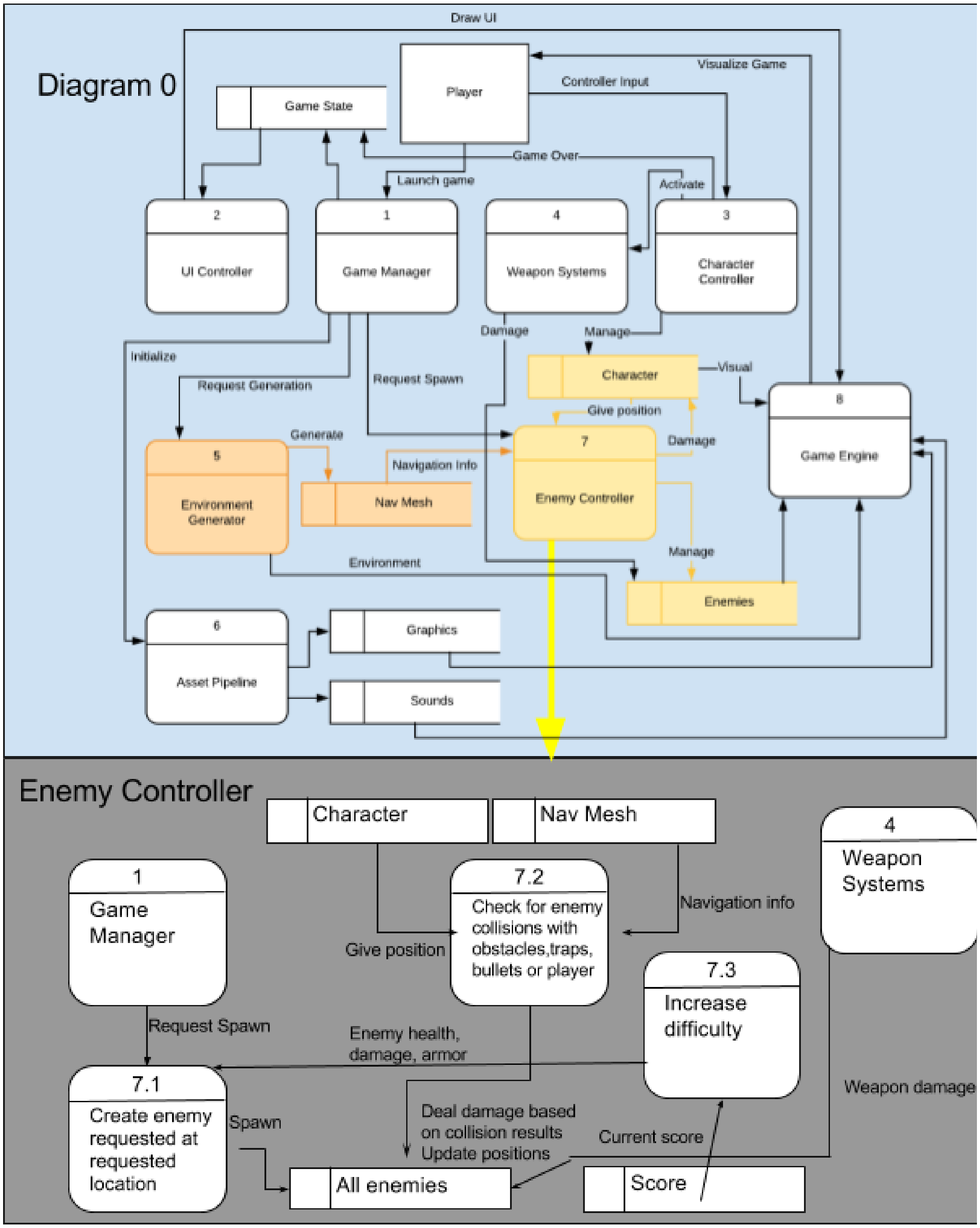
Post conditions: Game over and restart screen.

Priority: 1

ID: C06

3. Data Flow diagram(s) from Level 0 to process description for your feature ______14

Data Flow Diagram



Process Descriptions

Create enemy requested at requested location:

```
IF GameManager requests spawn of enemy THEN
    Create enemy requested
    Initialize health, damage, and armor from 7.3
    IF GameManager requests spawn where obstacle is THEN
        Return to GameManager with invalid location ELSE
        Place enemy and begin movement
        Give NavMesh information of this entity
        Begin movement
    ENDIF
ENDIF
```

Check for enemy collisions:

```
IF Navigation Info returns a collision
    IF collision is with player THEN
        Deal damage based on enemy’s power
        Stop movement for one cycle
    ELSE
        Turn enemy to attempt to get the enemy out of the current position that is causing collision
    ENDIF
ELSE
    Retrieve character position from Character and move in that direction
ENDIF
```

Increase difficulty:

```
IF Current score is lower than the easyThreshold, return lowest difficulty settings to 7.1
ELSE IF Current score is in the mediumThreshold, return medium difficulty settings to 7.1
ELSE IF Current score is in the hardThreshold, return hard difficulty settings to 7.1
ELSE IF hardThreshold is exceeded, increase health, damage, and armor by 5% every x points
ENDIF
```

4. Acceptance Tests _______9

The tests for this will include the following:

1. Current score set at the easyThreshold
    a. Tests will include ensuring each enemy spawns with given health and dies in the proper amount of shots from each weapon that the weapon controller has created.

        i. This will include a ‘God mode’ so that the player cannot be harmed in order to ensure the correct amount of bullets are used.

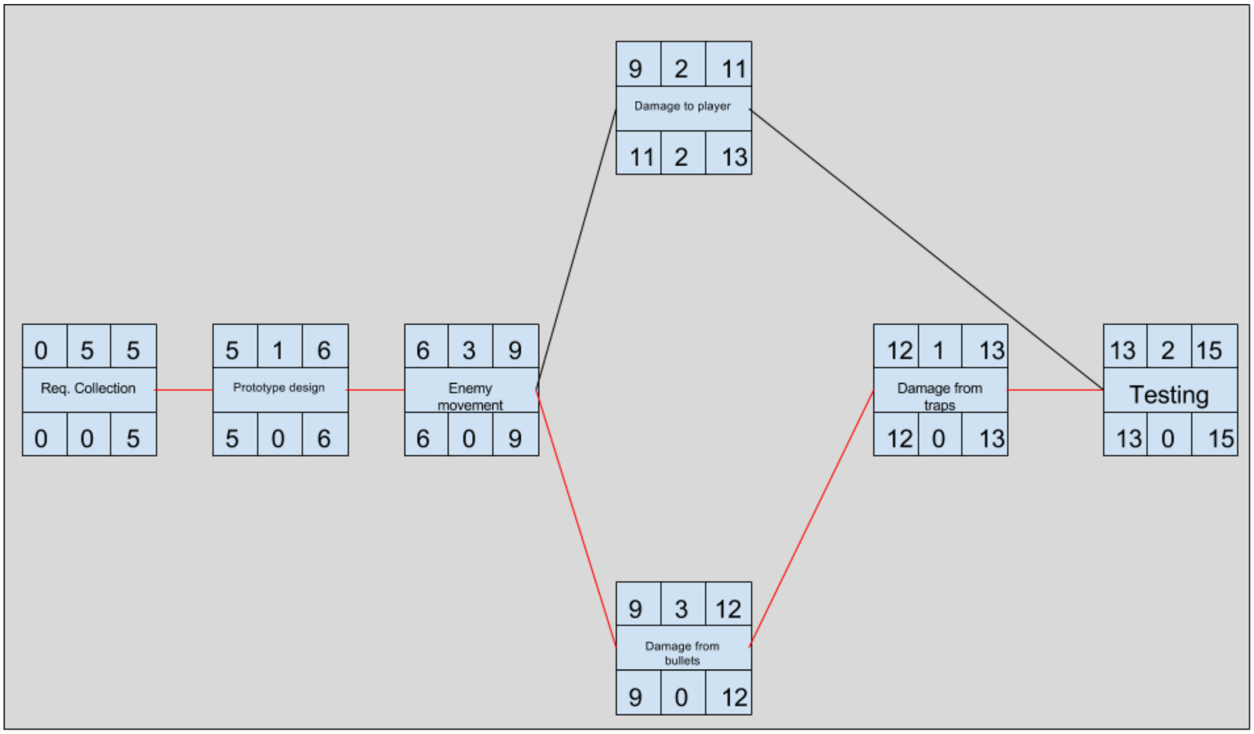
2. Current score set at the mediumThreshold
    a. Tests will include ensuring each enemy spawns with given health and dies in the proper amount of shots from each weapon that the weapon controller has created.
        i. This will include a ‘God mode’ so that the player cannot be harmed in order to ensure the correct amount of bullets are used.
3. Current score set at the hardThreshold but not yet exceeded
    a. Tests will include ensuring each enemy spawns with given health and dies in the proper amount of shots from each weapon that the weapon controller has created.
        i. This will include a ‘God mode’ so that the player cannot be harmed in order to ensure the correct amount of bullets are used.
4. Current score set above the hardThreshold
    a. Verify that every x score the player gets will increase the health by 5%.
    b. Tests will include ensuring each enemy spawns with given health and dies in the proper amount of shots from each weapon that the weapon controller has created.
        i. This will include a ‘God mode’ so that the player cannot be harmed in order to ensure the correct amount of bullets are used.
5. Game manager receives successful spawns for each type of enemy.
    a. This will be tested and determined by using a log file to ensure that no map that was generated for 1000 map generations cause an exception.
6. Character can take damage based on difficulty.
    a. This will be tested by only spawning one enemy on each score threshold and verifying that the correct amount of damage is dealt.
7. Verify that each trap does the right amount of damage
    a. This will be tested by spawning one type of enemy at a time. Then, will verify at 50 different coordinates in 10 different map generations that the enemies take the proper amount of damage when collision occurs.
        i. God mode will be needed for this to place the trap and prevent damage.
8. Verify that score increases upon enemy death
    a. This will be tested using a God mode in order to kill specific types of monsters using both traps and bullets. Verification will happen instantly when the score on-screen is updated with the correct amount increased.
9. Verify that enemies cannot collide with each other
    a. This will be tested by general gameplay and ensuring that they cannot get stuck within each other and break.
10. Verify that each weapon the weapon controller sends will have bullets that can damage enemies accordingly
    a. This will be tested using a God mode and verified using a log file to determine health loss.

5. Timeline ________/10

Work items

Task	Duration (PWks)	Predecessor Task(s)
1. Requirements Collection	5	-
2. Prototype Design	1	1
3. Enemy Movement	3	2
4. Dealing damage to player	2	2,3
5. Damage from bullets	3	2,3
6. Damage from traps	1	3,5
7. Testing	2	4, 5, 6, 7

Pert diagram



Gantt timeline

1	🟥 -	🟥	🟥	🟥	🟥										
2						🟥 1									
3							🟥 2	🟥	🟥						
4										🟥 2,3	🟥	🟦	🟦		
5										🟥 2,3	🟥	🟥			

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
6													🟥 3,5		
7														🟥 4,5,6,7	🟥