Name: Connor McElroy Mark ______________________/50

[**Instructions**: Remove everything that is not a heading below and fill in with your own diagrams, etc.]

1. Brief introduction __/3

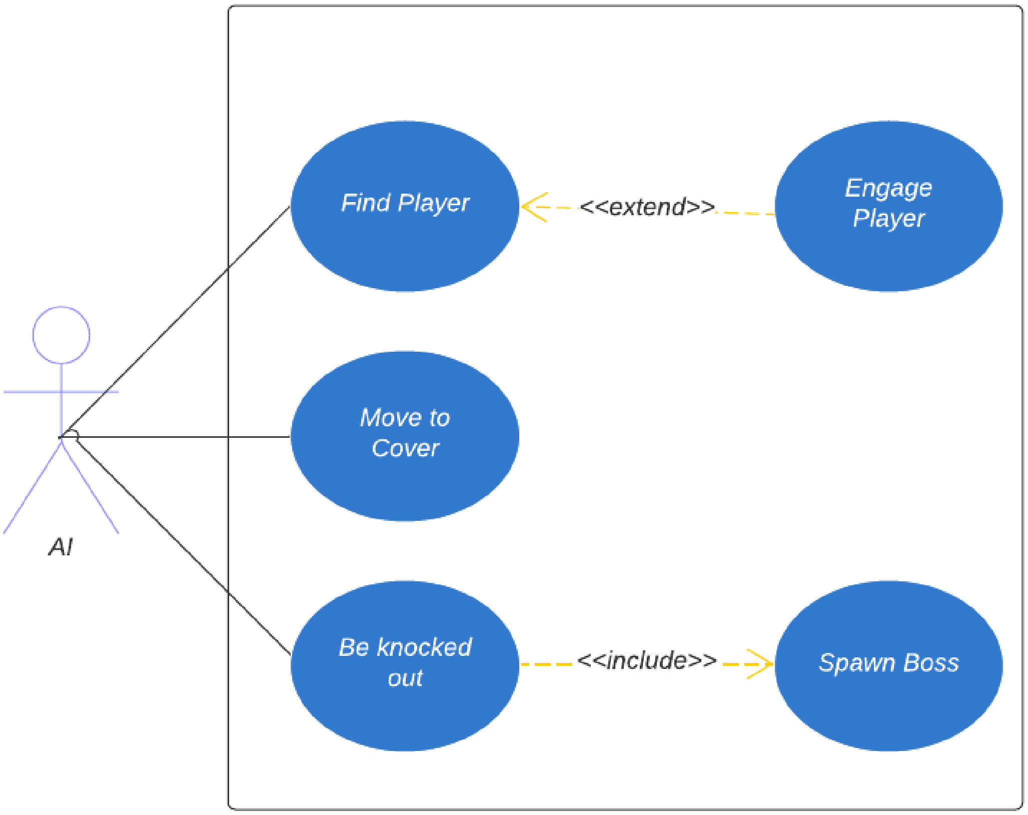
My features for the Incoming!! video game are the dodgeball projectiles and the AI behavior.

Firstly, I have to design the dodgeballs, the bread and butter projectile the game revolves around. I have to ensure that thrown dodgeballs bounce off level objects in a realistic and predictable way to create the high-intensity all-directions danger field we want in the game when throwing these dodgeballs around.

Secondly, I am responsible for designing the AI behavior trees. The AI's in this game must be able to maneuver around cover and around each other, targeting the player when they are able and behaving intelligently when they are not. Also, the boss character AI must inherit from this to function similarly, but not the same.

2. Use case diagram with scenario __14

Use Case Diagrams



Scenarios

Name: Find Player

Summary: The AI receives information on the player's location

Actors: AI

Preconditions: The AI has been spawned

Basic sequence:

Step 1: Receive player location

Name: Move to cover

Summary: The AI choses nearby cover and moves to it

Actors: AI

Preconditions: The AI has been spawned

Basic sequence:

Step 1: Calculate the nearest viable cover from a list of level objects

Step 2: Navigate to that cover

Name: Be knocked out

Summary: The AI is struck with a dodgeball and outed

Actors: AI

Preconditions: The AI has been spawned

Basic sequence:

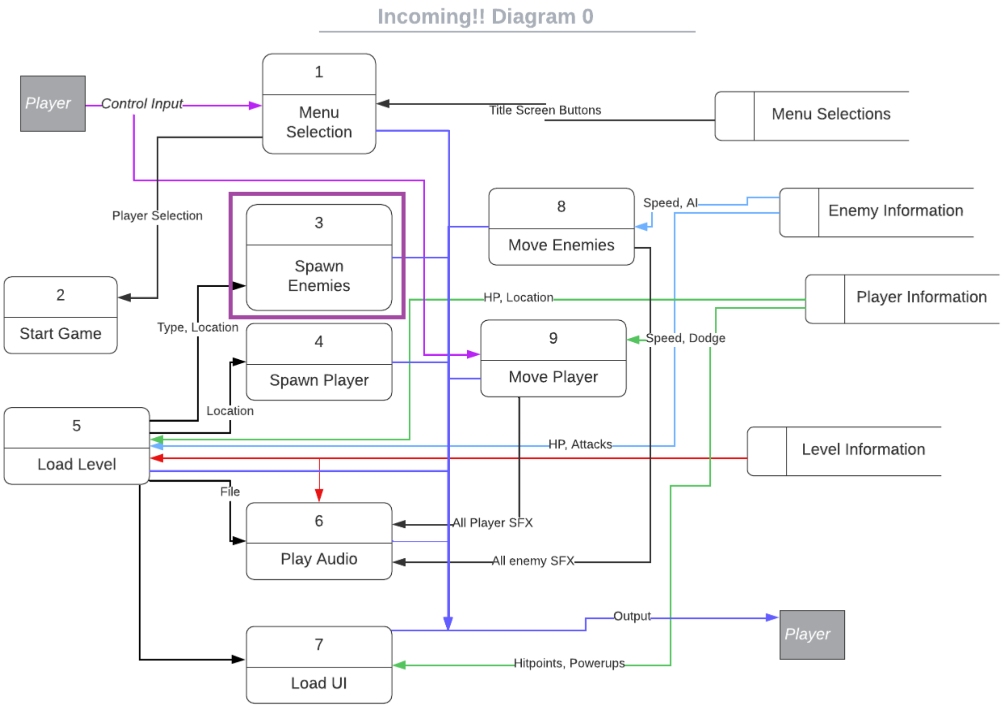
Step 1: The AI is struck

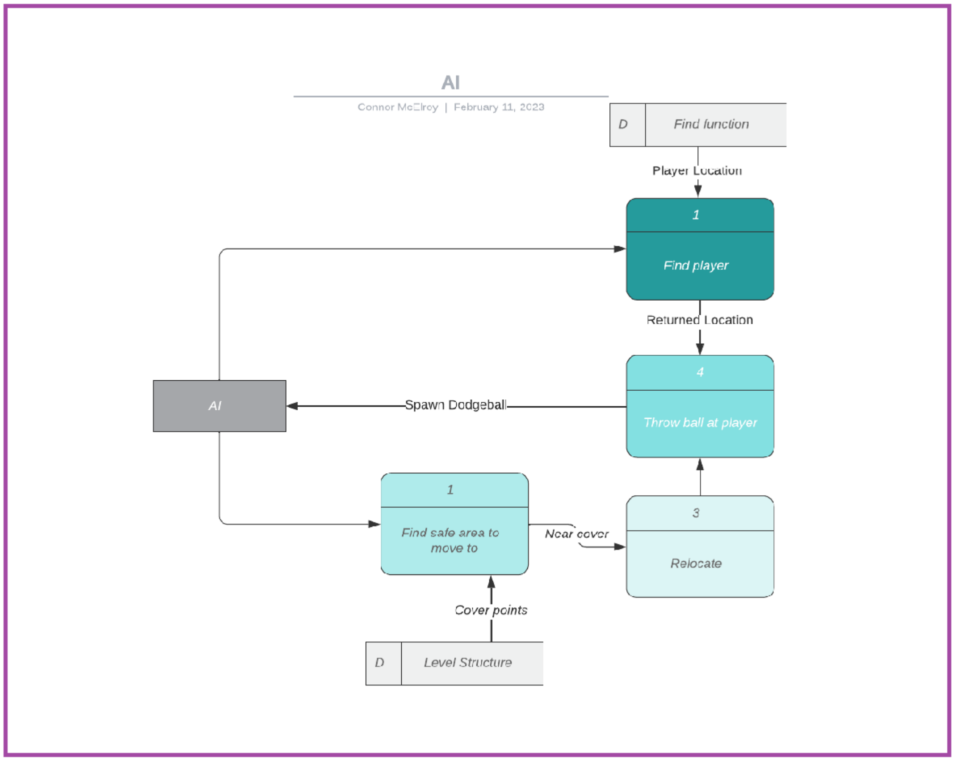
Step 2: The AI goes through an outing animation

Step 3: The AI is despawned

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

Data Flow Diagrams





Process Descriptions

```
Find safe area:
        FOR each cover spot
           IF nearer, make nearest
        END FOR
Relocate
        MOVE to cover
Find player:
        RETURN player location
Throw ball at player:
        INSTANTIATE dodgeball class
```

***Notes**: Yours should be much longer. You could use a decision tree or decision table instead if it is more appropriate.

4. Acceptance Tests _______9

For the dodgeballs:

I intend to run the feature in a blank level such that progressively larger numbers of dodgeballs will spawn and be accelerated randomly into level geometry

This will test the following:

- The system is able to handle well beyond the number of possible in-game concurrent active dodgeballs without a hit to performance.
- The dodgeballs will not interact with each other's geometry or incur changes to each other's physics.

Example test for stress:

# dodgeballs	1	5	10	…
avg FPS	60	59	55	…

For the AI's:

I intend to run the feature in a blank level such that progressively larger numbers of AI's will spawn and be released to follow their patrol code, randomly occasionally introducing a player.

This will test the following:

- The system is able to handle well beyond the number of possible in-game concurrent active AI's without a hit to performance.

- The AI’s will not interact with each other’s geometry or impede each other’s pathfinding or decision making

Example test for stress:

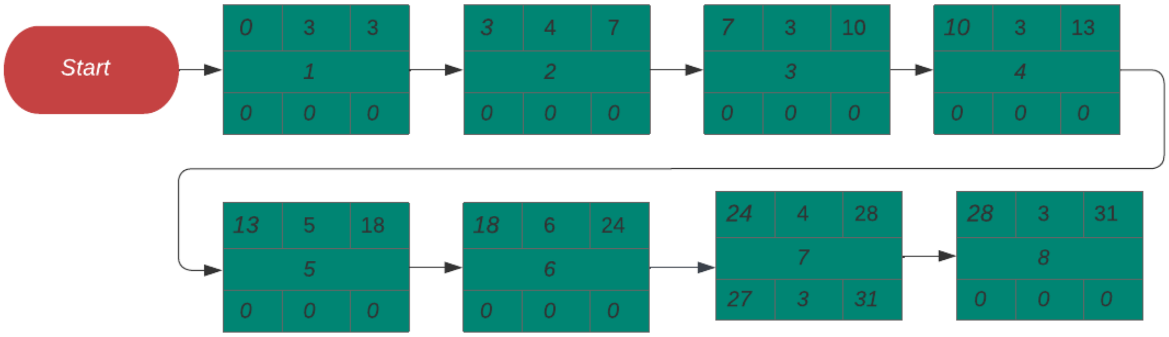
# AIs	1	5	10	…
FPS	60	59	58	…
Geom errors?	no	no	yes	…
Broken pathing?	no	yes	yes	…

5. Timeline _______/10

Work items

Task	Duration (Hours)	Predecessor Task(s)
1. Requirements Collection	3	-
2. Navigation mesh	4	1
3. Covered movement	3	1, 2
4. Dodgeball attacks	3	2
5. Group interactions	5	3, 4
6. Boss AIs	6	5
7. Animations	4	6
8. Testing	3	6, 7

Pert diagram



Gantt timeline

1	🟥 1	🟥	🟥																											
2				🟥 2	🟥	🟥	🟥																							
3								🟥 3	🟥	🟥																				
4											🟥 4	🟥	🟥																	
5														🟥 5	🟥	🟥	🟥	🟥												
6																			🟥 6	🟥	🟥	🟥	🟥	🟥						
7																									🟥 7	🟥	🟥	🟥	🟦	🟦	🟦
8																													🟥 8	🟥	🟥
	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