

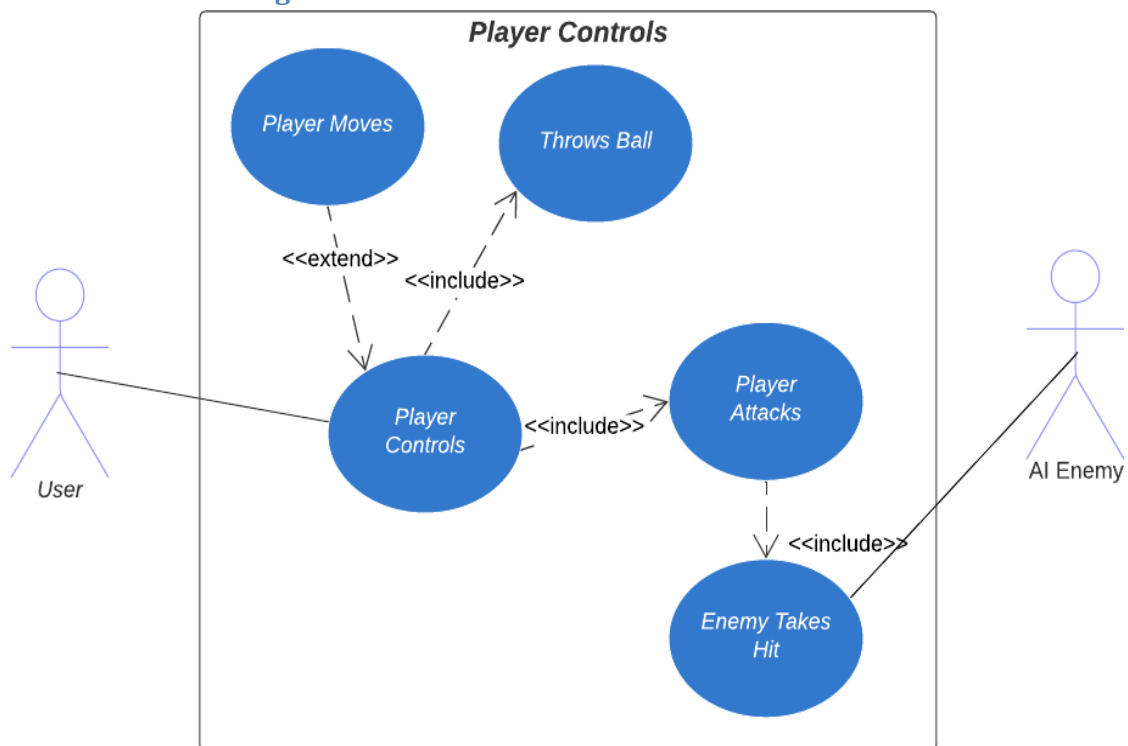
[Instructions: Remove everything that is not a heading below and fill in with your own diagrams, etc.]

1. Brief introduction __/3

This feature is the player character, which the user will control to interact with the game environment. The user will be able to move the character up, down, left and right. The player will also be able to dodge incoming balls and will be able to throw their own balls at the enemies to eliminate them. The player will have, health as well as stamina that can be used for a power attack.

2. Use case diagram with scenario __14

Use Case Diagrams



Scenarios

Name: Player Controls

Summary: The user uses the player controls to move and throw the dodgeball at the enemy

Actors: User and AI enemy.

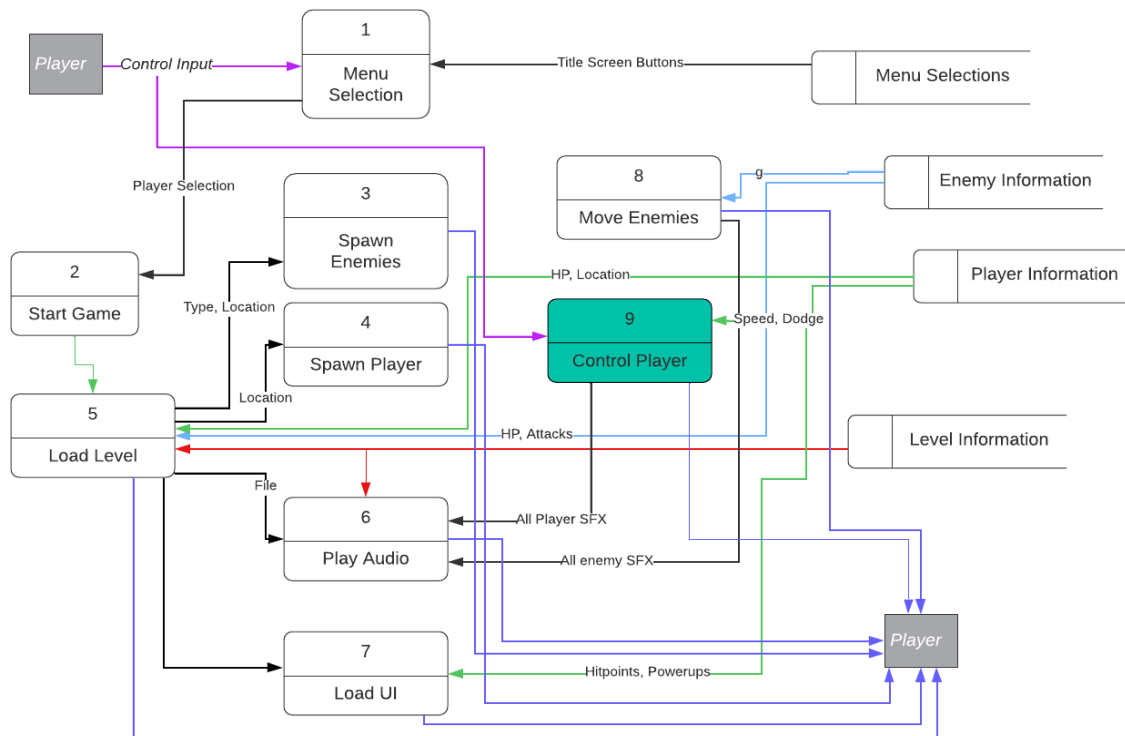
Preconditions: Level has been started.

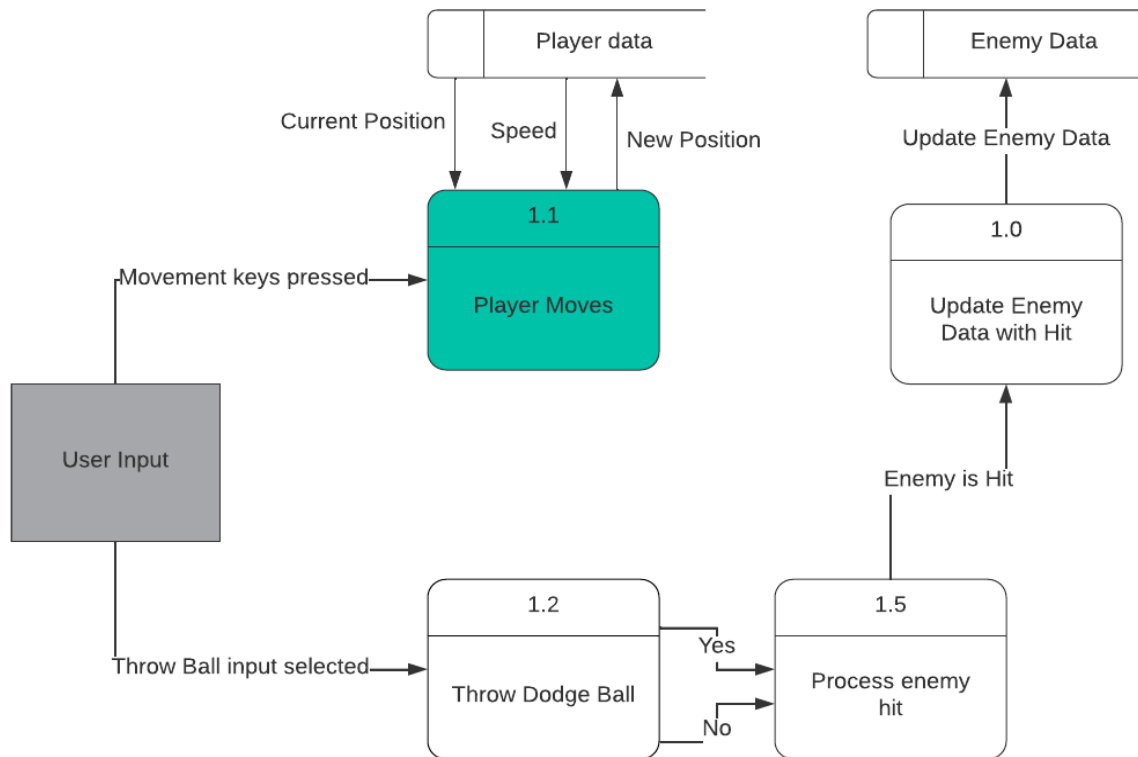
Basic sequence:

Step 1: User controls the player to move into position.

Step 2: Using the controls the user throws a ball at enemy.

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





Process Descriptions

Throw Dodge Ball:

```

WHILE player health is above death level
    IF up key pressed AND no collision detected
        Move player up on the map
    IF down key pressed AND no collision detected
        Move player down
    IF left key pressed AND no collision detected
        Move player left
    IF right key is pressed AND no collision detected
        Move player right
    IF dodge key pressed AND no collision detected
        IF get random number > midpoint of range
            IF roll will NOT cause collision
                roll player out of way.
        ELSE
            Dodge fails, player takes damage
    END IF
END WHILE
    
```

4. Acceptance Tests _____9

I intend to test the player collision detection when moving around obstacles in order to test the dodge function when dodging will result in colliding with an object such as terrain.

- Test with player character next to object with no gap (collision boxes overlapping).
- Test with player within dodge boundaries (< 1).
- Test with player at edge of dodge boundaries (≥ 1).

Example for divide feature

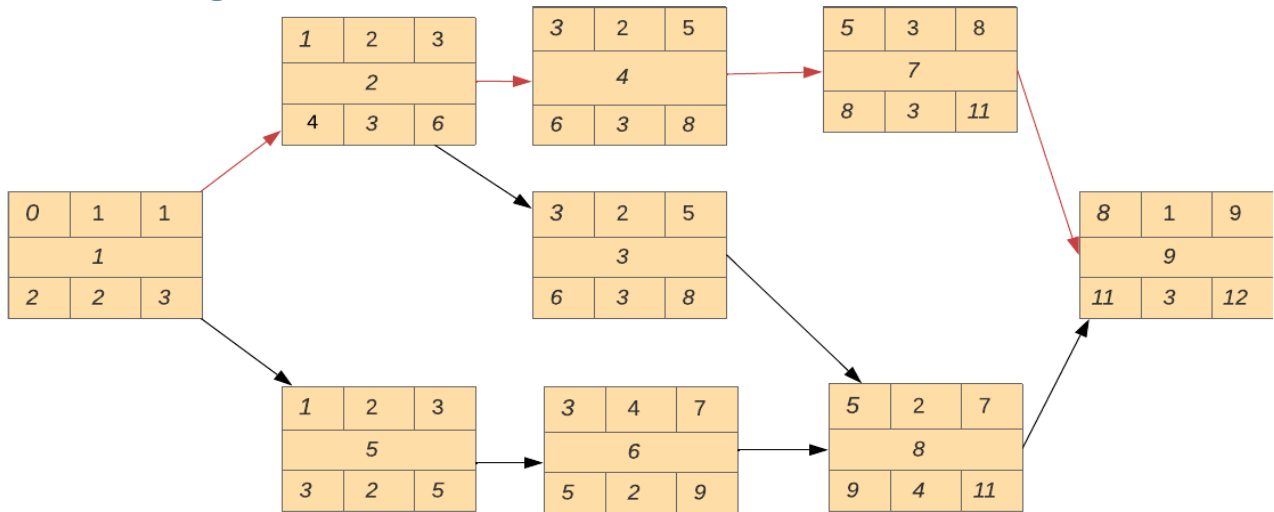
Distance from object (Unity Units 1 = meter)	Expected Result
0	Dodge fails, player takes damage
.25	Dodge fails, player takes damage
1	Dodge success, player escapes

5. Timeline _____/10

Work items

Task	Duration (PWks)	Predecessor Task(s)
1. Requirements Collection	1	-
2. Player Class Design	2	1
3. Player Controls	2	2
4. Physics Logic	2	2
5. Asset Design	2	1
6. Player Animation	4	5
7. Documentation	2	4
8. Testing	3	3,6
9. Release	1	7,8

Pert diagram



Gantt timeline

