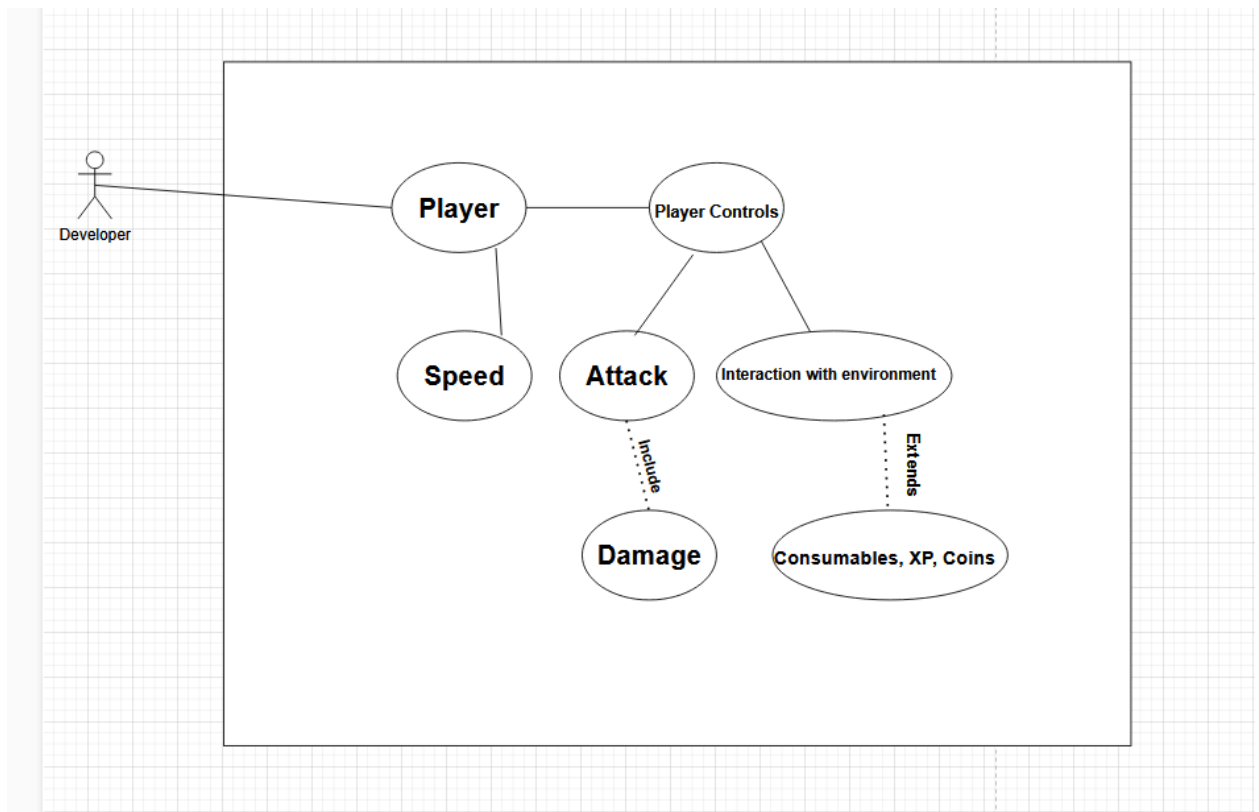


1. Brief Introduction __/3

For our group project my team, SuperNova Nerds, are designing a game in which I will be working on input controls related to the controls of the player, attacking, and its movement interactions. This includes:

- Basic movement including movement up, down, left, and right.
- Advanced movement including the blending of the basic movements, such as right-up or left-down etc..
- Mobile Controls involving touch screen controls, and movements controls.
- Porting to Android, IOS, and other platforms.
- Interactions with the player sprite including basic animations related to movements, and attacks.
- Player attacks and their related interactions to things such as environment and enemies.

2. Use case diagram with scenario __/14



Scenarios

1. **Name:** Speed
 - 1.1. **Summary:** Defines the speed of the player based on the movement of the player and what controls have been inputted.
 - 1.2. **Actors:** Developer
 - 1.3. **Precondition:** Player must be in the game environment.
 - 1.4. **Basic sequence:**

- 1.4.1. Player is spawned.
- 1.4.2. Player must input a key.
- 1.4.3. Player sprite must cross the environment area according to input at a predetermined rate.

1.5. Exceptions:

- 1.5.1. Player is not initialized properly.
- 1.5.2. Player is dead.
- 1.5.3. Player has a debuff affecting speed or inputs.

1.6. Post conditions:

- 1.6.1. Player stops inputting the movement key.
- 1.6.2. Player dies.

2. Name: Player Controls

2.1. **Summary:**Interacts with the player sprite based on the controls selected and their given input to move the player around the environment.

2.2. **Actors:** Developer

2.3. **Precondition:** Player must be in the game environment.

2.4. Basic sequence:

- 2.4.1. Player is spawned.
- 2.4.2. Player must input a key or touch screen manipulation.
- 2.4.3. Player sprite must cross the environment area according to input and moves to desired area.

2.5. Exceptions:

- 2.5.1. Player is not initialized properly.
- 2.5.2. Player is dead.
- 2.5.3. Player has a debuff affecting speed or inputs.

2.6. Post conditions:

- 2.6.1. Player stops inputting the movement key.
- 2.6.2. Player dies.

3. Name: Attack

3.1. **Summary:**Actors: Developer

3.2. **Precondition:** Player must be in the game environment and have aimed correctly at the enemy.

3.3. Basic sequence:

- 3.3.1. Player moves in range of the enemy.
- 3.3.2. Player aims.
- 3.3.3. Player attacks.
- 3.3.4. Attack hits dealing damage according to preset and scaled values.

3.4. Exceptions:

- 3.4.1. Player cannot deal damage to enemy type yet.
- 3.4.2. Player misses the attack.
- 3.4.3. Player dies.

3.5. Post conditions:

- 3.5.1. Enemies will take damage related to Player level and other factors.

4. Name: Interaction with environment

- 4.1. **Summary:** Player interacts with items within the current space such as shops/vendors, item pickups, or item usage.

- 4.2. **Actors:** Developer

- 4.3. **Precondition:** Player must be in range of interaction and alive.

4.4. Basic sequence:

- 4.4.1. Player goes within range of the item or area.
- 4.4.2. Player presses the key to interact with the item or area.
- 4.4.3. Interaction is prompted and occurs.
- 4.4.4. Continuation of interaction if required. (Buying items, swapping items)

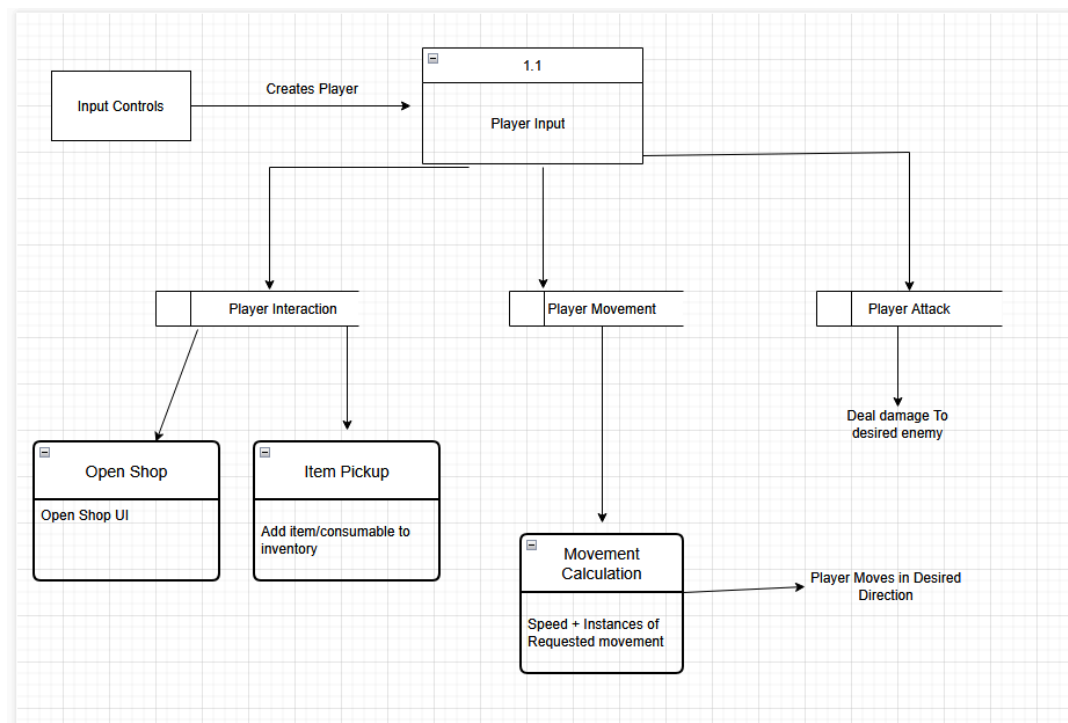
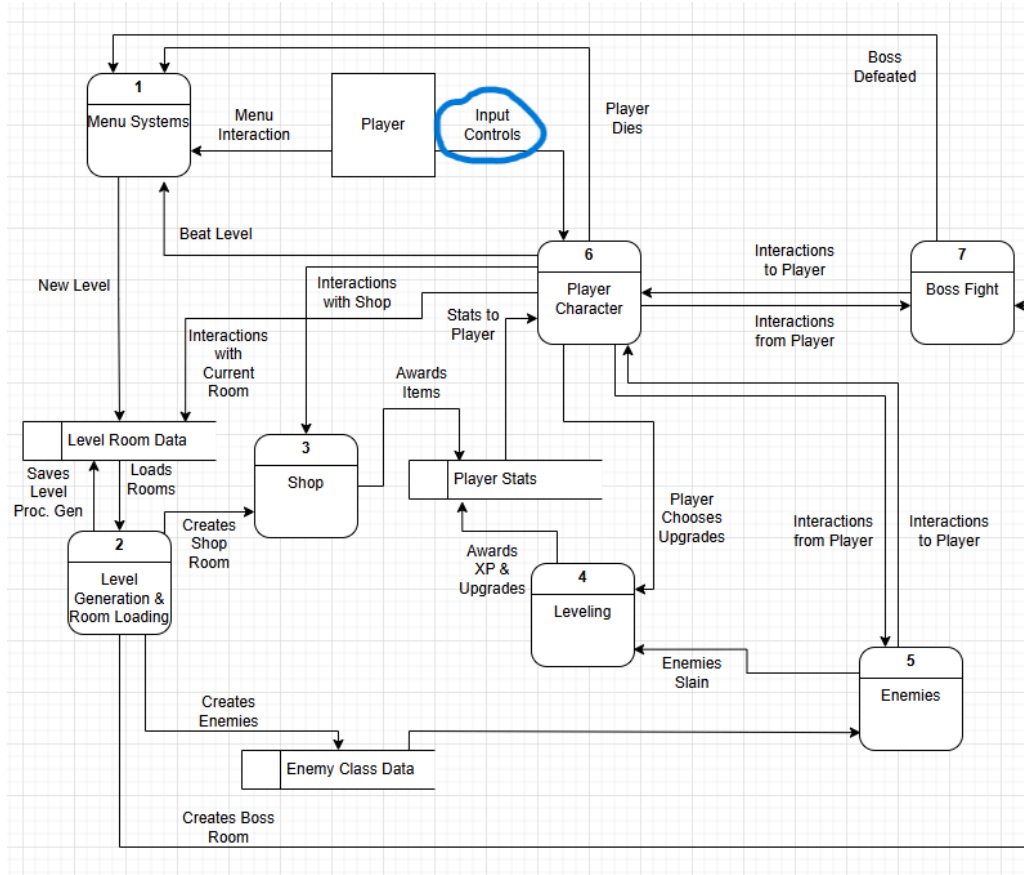
4.5. Exceptions:

- 4.5.1. Player has full inventory.
- 4.5.2. Player does not have sufficient funds to purchase items.
- 4.5.3. Player cannot use item.

4.6. Post conditions:

- 4.6.1. Player and item interact creating outcome desired.

3. Data Flow diagram(s) from Level 0 to process description for your feature __/14



Process Descriptions:

Player Controls:

Create Player():

Add Player Attack;

Add Player Speed;

Add Player Controls;

Add Player Interaction;

Wait for inputs();

Receive inputs();

if(input == PlayerControl)

{

Move Player within environment(based on speed value);

}

if(input == PlayerAttack)

{

Attack Enemy(determined by player aim/cursor);

}

if(input == PlayerInteract)

{

If(interact == shop)

{

Open ShopUI();

}

If(interact == item pickup)

{

Add item to inventory;

If (inventory = full)

{

Display message informing player;

}

}

4. Acceptance Tests __/9

There will be multiple tests run with these controls to ensure 3 things: Proper movement, Attacks, and intractability. Each of these tests will be run for each type of movement, attack, and interactions.

- Move in a direction 100 times, in any environment.
- Move along objects and ensure no clipping occurs 100 times.
- Move into an enemy's sprite directly 100 times.
- Interact with an item 100 times, in any environment.
- Interact with the shop 100 times, in the first environment it appears.
- Interact with a pickup 100 times.

- Attack 100 of each enemy type.
- Attack 100 times towards a boss type.
- Attack with no enemies present.

Action Taken	Non-Occurrence? (_/100)	Average FPS	Issues?
Interaction	5	45	Shop wouldn't display it every time.
Movement	2	54	No, User error
Attack	0	32	N/A

5. Timeline ___/10

Task	Duration (PWks)	Predecessor Task(s)
1.Predefine Workload	2	-
2. Collaboration between members	3	1
3. Organization of Git + Files	3	1
4. Character Design + Theme design	8	2,3
5. Documentation and Revision	6	2,4
6. Programming and Haptic integration + Mobile	16	1,5
7.Testing Features and capability of mobile devices	4	3,6
8.Install and Integration	4	5,6,7

5.2 Pert Diagram

