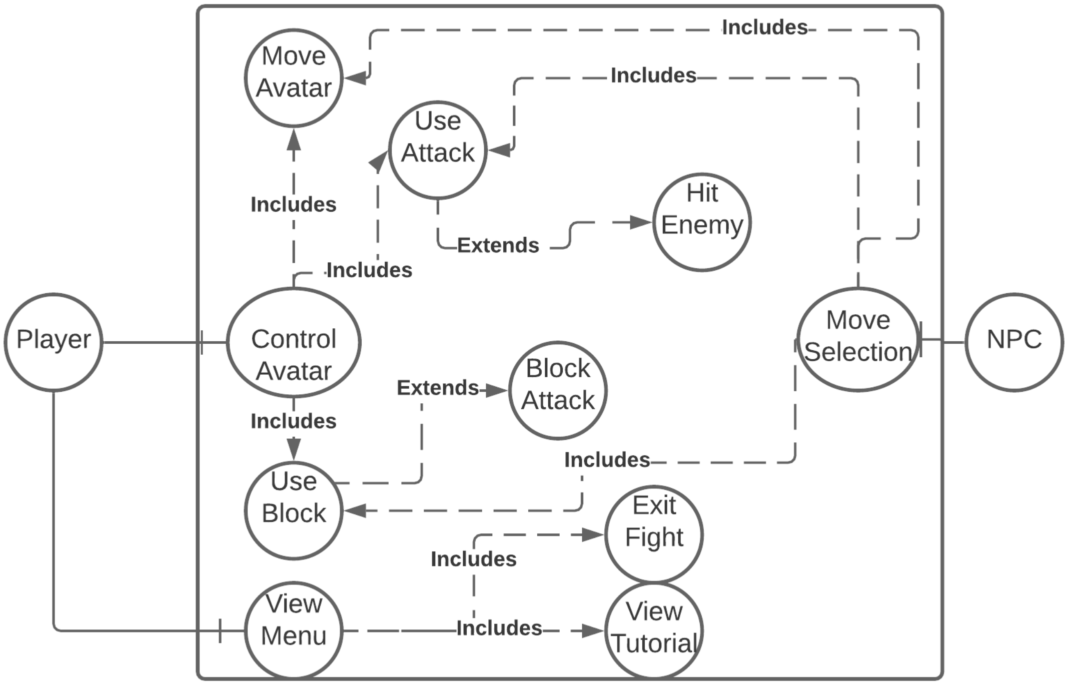
Kaleb Johnson

Introduction:

The fighting game system is a segment where the player fights against whatever NPC they initiated combat with in the overworld. The game is based on standard 2D fighters, in the vein of Street Fighter 2.

Use Case Diagram



Scenarios

Name: Control Avatar
Summary: The player controls their character
Actors: Player
Preconditions: The fight has begun
Basic Sequence:

1. The player inputs commands
2. Those commands call Move, Attack, or Block functions.
3. When health of one character is reduced to zero, stop accepting inputs and display a win/loss screen.

Exceptions: During the pause menu: ignore all character inputs.
Post Conditions:
Priority: 1

Name: Move Avatar
Summary: The controller moves a character
Actors: Player, NPC
Preconditions: The fight has begun and the avatar is not in hitstun.

Basic Sequence:

1. Accept directional input
2. Move avatar in that direction

Exceptions: During the pause menu: ignore all inputs.
Post Conditions: Avatar is moved.
Priority: 1

Name: Use Attack
Summary: Uses an attack
Actors: Player, NPC
Preconditions: The fight has begun and the avatar is not in hitstun.
Basic Sequence:

1. Accept attack input
2. Begin appropriate attack animation
3. Check for enemy collision
4. On hit, deal damage and go to Hit Enemy.
5. Otherwise, finish animation then reset to neutral.

Exceptions: During the pause menu: ignore all inputs.
Post Conditions: Attack animation is completed.
Priority: 1

Name: Use Block
Summary: Transitions to blocking state, which stops attacks
Actors: Player, NPC
Preconditions: The fight has begun and the avatar is not in hitstun.
Basic Sequence:

1. Accept block input
2. Transition to block sprite
3. If hit, transition to Block Attack
4. Otherwise, block until the input is released.

Exceptions: During the pause menu: ignore all inputs.
If attacking or moving: ignore block input.
Post Conditions:
Priority: 2

Name: View Menu
Summary: Brings up the menu.
Actors: Player
Preconditions: Not in menu
Basic Sequence:

1. Press the menu button
2. Show the menu

Exceptions: If in menu: exit the menu instead.
Post Conditions: Menu is displayed.
Priority: 1

Name: Move Selection
Summary: The NPC function for picking a move.
Actors: NPC

Preconditions: The fight has begun
Basic Sequence:

1. Depends on the NPC

Exceptions: During the pause menu: halt
Post Conditions:
Priority: 1

Name: Hit Enemy
Summary: Occurs on collision of an attack with a non-blocking enemy.
Actors: Player, NPC
Preconditions:
Basic Sequence:

1. Attack collides with an enemy
2. Check the enemy's block state
3. If

Exceptions: During the pause menu: halt
Post Conditions:
Priority: 1

Name: Block Attack
Summary: Block the enemy's attack
Actors: Player, NPC
Preconditions: Blocking
Basic Sequence:

1. Detect collision of enemy attack with the user block
2. Perform the necessary modifiers to the attack animation
3. Push the blocking user back.

Exceptions: During the pause menu: halt
Post Conditions:
Priority: 2

Name: Exit Fight
Summary: Exits the fight from the pause menu
Actors: Player
Preconditions: In menu
Basic Sequence:

1. Select exit option
2. Display a loss animation.
3. Move to the overworld scene

Exceptions:
Post Conditions: Returned to the overworld
Priority: 2

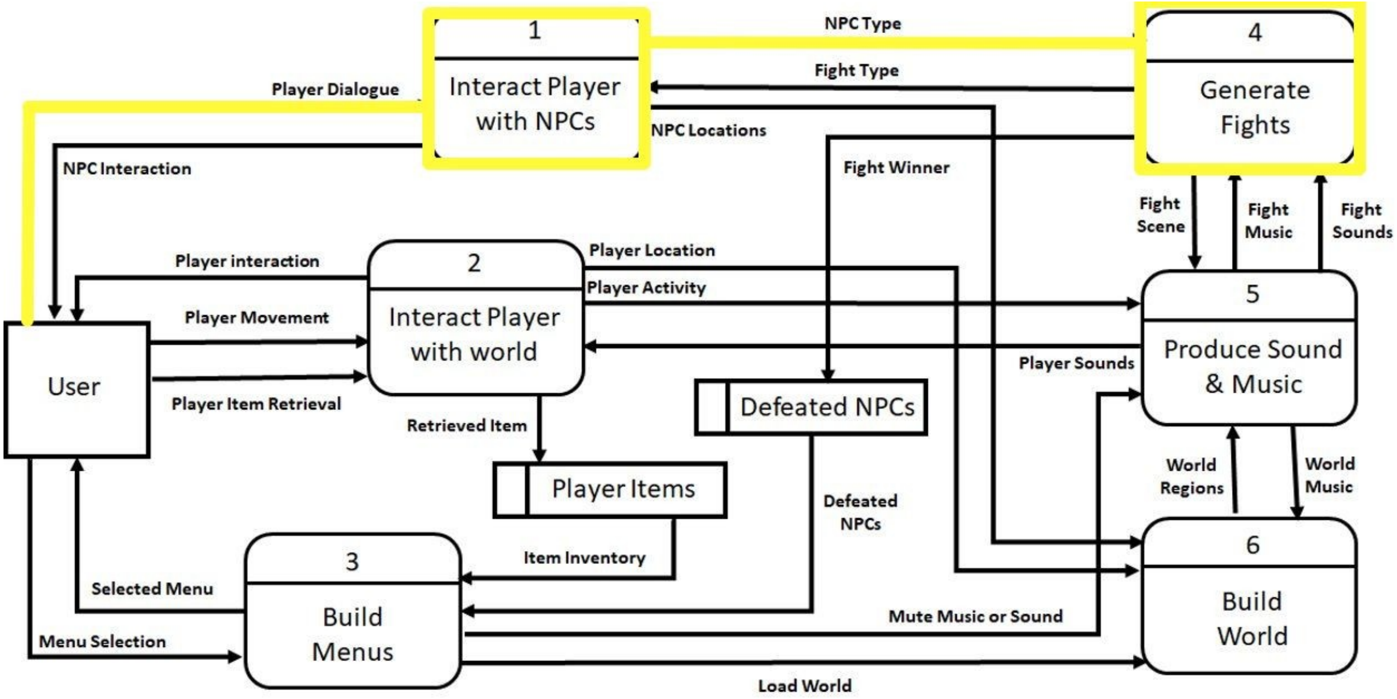
Name: View Tutorial
Summary: Shows the tutorial
Actors: Player
Preconditions: In menu
Basic Sequence:

1. Select tutorial from menu

2. Show tutorial screen

Exceptions:
Post Conditions: Tutorial is displayed
Priority: 3

Data Flow Diagram



While the player is interacting with an NPC, if they choose a fight option in their dialogue with the NPC, begin the fight.
The type of the fight depends on the NPC that the player chose to fight.
The fight type is communicated to the audio system, which generates the appropriate music/sounds.
Once the fight is concluded, return to the dialogue with an acknowledgment of the player winning or losing.

Acceptance Tests

Attack testing:

Have a handler cycle through each unique attack, with the enemy collision box taking up the entire arena. Record damage values to a file.

Compare the damage values to the assigned values of the attacks.

Cycle through each attack with the enemy collision box at a normal size at the opposite side of the arena as the attacker. Record if the attack hits to a file.

Each attack should miss.

Cycle through each attack with the enemy collision box at a normal size adjacent to the attacker. Record if the attack hits to a file.

Each attack should hit.

Block Testing

Cycle through each attack with the enemy blocking adjacent to the attacker. Record the attack, whether it collided with the enemy, if it was blocked, and the location of the enemy before and after the attack. Do this with both player and NPC characters blocking.

Each attack should collide, be blocked, and create the assigned amount of pushback occurred.

Timeline:

Work items:

Task	Duration	Predecessor Tasks
1.Player Controls	3	-
2. Player Moves	5	1
3. Interface	3	-
4. NPC AI	5	-
5. Transitions	1	3
6. NPC Moves	10	4
7. Move Animations	5	2, 6
8. Hit/Block Interactions	5	2, 6
9. Dr-BC Mode	1	5, 7, 8
10. Special Moves and Meter	5	7, 8
11. Autoplay	10	5, 7, 8
12. Testing	7	9, 10, 11

Gantt:

1	🟥	🟥	🟥																									
2				🟥 1	🟥	🟥	🟥	🟥																				
3	🟩	🟩	🟩																									
4	🟦	🟦	🟦	🟦	🟦																							
5				🟩 3																								
6						🟦 4	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦													
7																🟥 6	🟥	🟥	🟥	🟥								
8																🟥 6	🟥	🟥	🟥	🟥								
9																					🟥 8							

10																					🟥 8	🟥	🟥	🟥				
11																					🟥 8	🟥	🟥	🟥	🟥	🟥	🟥	🟥
12																												

1										
2										
3										
4										
5										
6										
7										
8										
9										
10										
11	🟥	🟥	🟥							
12				🟥 11	🟥	🟥	🟥	🟥	🟥	🟥