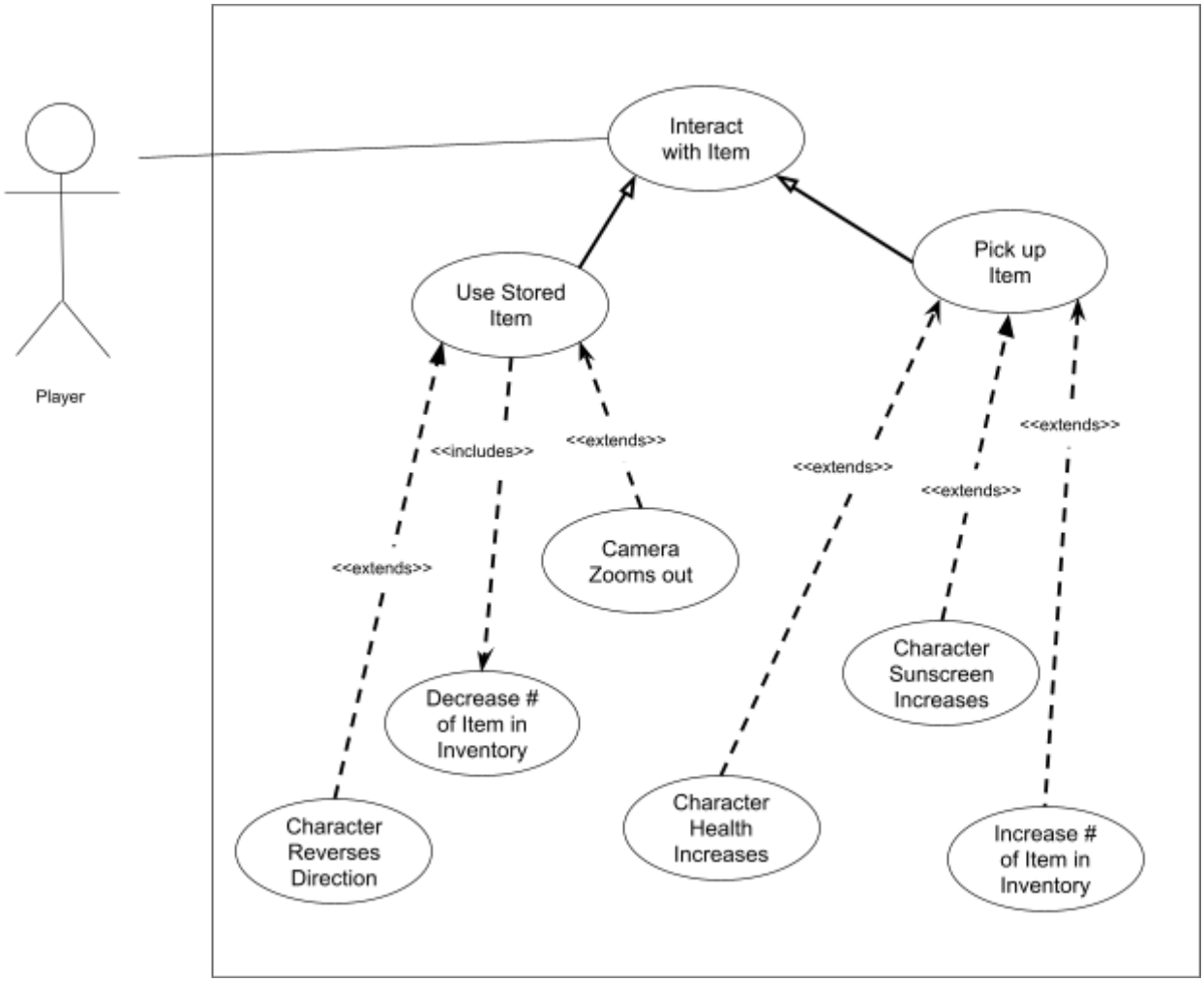
Logan Finley Mark _____/50

1. Brief introduction __/3

I am responsible for creating the different items and collectibles that can be found in Lake Runner. This includes getting art for the different items and animating them, as well writing the code to determine how each item affects the character and the environment.

2. Use case diagram with scenario __14

Use Case Diagrams



Scenarios

Name: Use stored item.
Summary: The player uses an item that they have stored in their inventory.
Actors: The player.
Preconditions: The player has to have at least one item in their inventory.
Basic sequence:

Step 1: The player selects an item by pressing a key on the keyboard..
Step 2: The player chooses to use the item by pressing a key on the keyboard

Exceptions:

Step 1: The item does not exist in the inventory. Display a UI element telling the user they do not have the requested item.

Post conditions:

- If the player used a Brain Blast Bar, the character is now running backwards.
- If the player used Sunglasses, the map is zoomed out and the player has unlimited build materials.

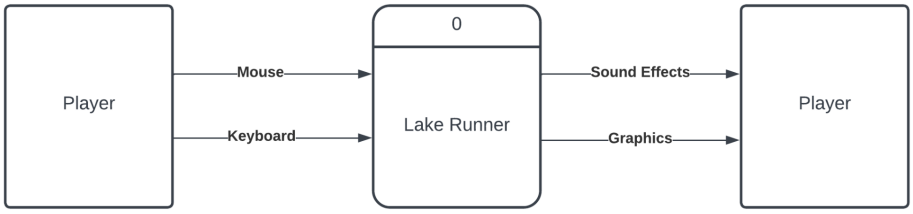
Priority: 1*

ID: P01

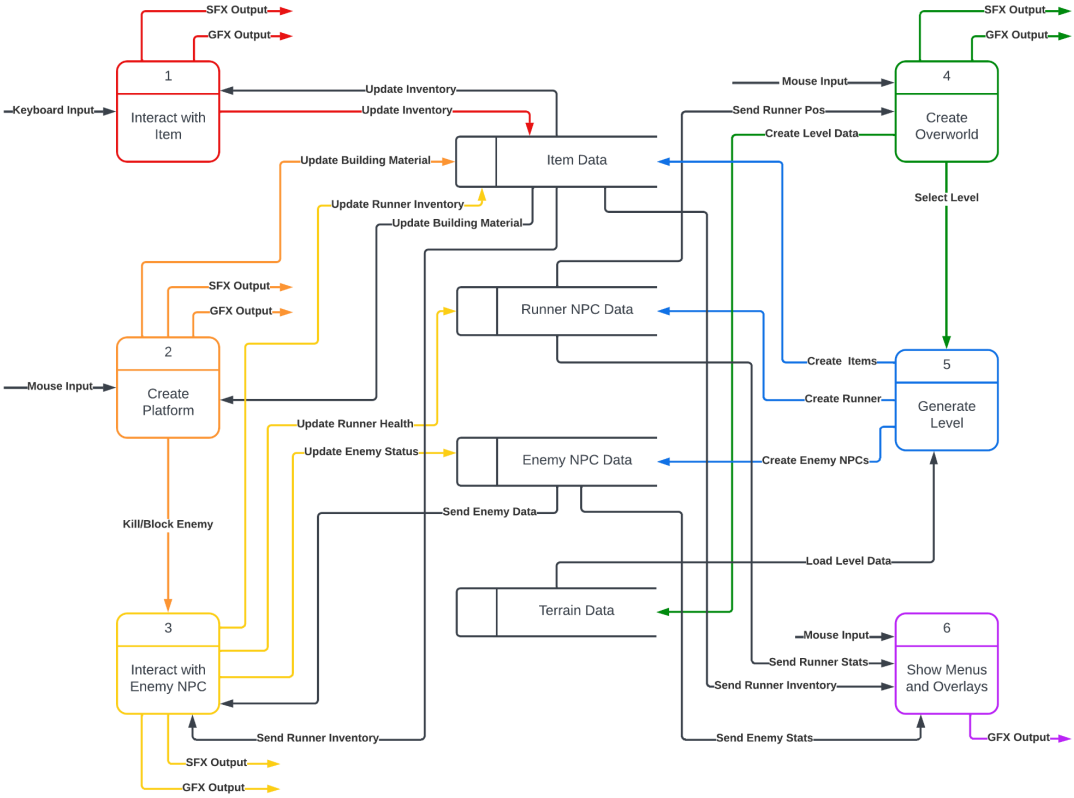
*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

3. Data Flow diagram(s) from Level 0 to process description for your feature _____14

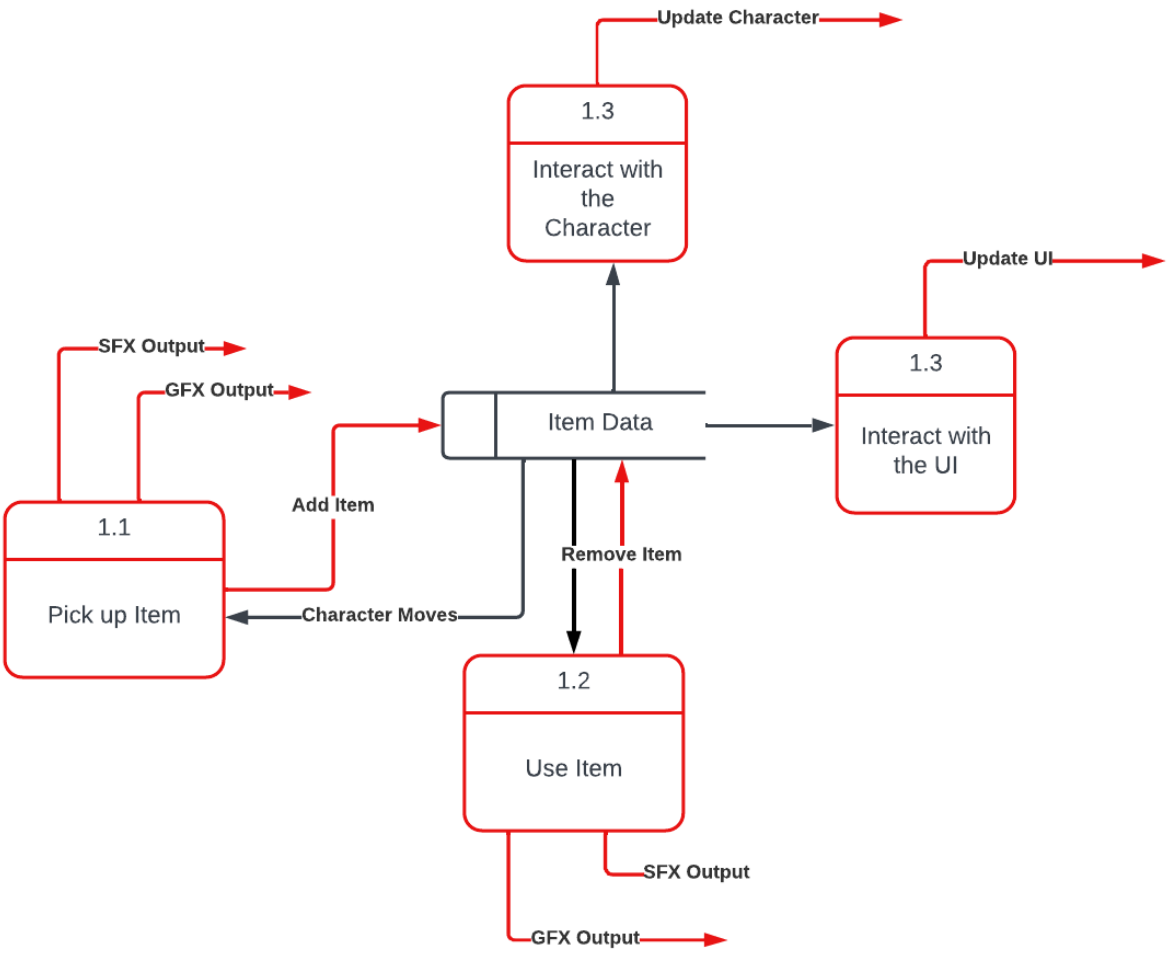
Context Diagram



Data Flow Diagram 0



Data Flow Diagram 1



Process Descriptions

1.1 Pick Up Item

IF the character moves AND picks up an item THEN
Add the item to the character's inventory
Play item-get sound effects
Display item in inventory bar
ENDIF

1.2 Use Item

IF the character has an item AND the player uses it THEN
Remove the item from the character's inventory
Play item-used sound effects
Remove the item from the character's inventory bar
ENDIF

1.3 Interact with the UI

Update the UI to reflect the current status the character's inventory

1.4 Interact with the character

IF the player used an item THEN
Apply the effects and update the character NPC
ENDIF

4. Acceptance Tests _____9

Character Interaction with Items

Whenever the character runs over an item on the ground, there are only two possible outcomes:

- The character picks the item up
- The character runs past the item without picking it up

The character should only pick up the item if they have enough room in their inventory or attribute bars to accommodate the item. To test whether this feature is working correctly, I would like to run some tests where the player runs over the same item 1000 times. Before each test begins, I will save the player's current status in a file and then append to that file the player's status after they've run over the item. If in one of the trials, the character's final status was the same as it was initially, then the test failed and there's a bug somewhere in the code.

Item Interactions with Menus

Whenever the character picks up an attribute item (such as the sunscreen bottle or the aloe vera jar), the item is immediately used and the corresponding menu bar is updated in the UI. To test whether this feature is functional, I'll first make the assumption that there are no errors in the way that the value of the character's attribute is displayed graphically through the UI. I would like to run a series of tests where the player runs over an attribute item repeatedly. The data collected would be the player's attribute status before and after picking up the item. If in any of the trials, the player's attribute status is unchanged then there is an error somewhere in the code.

Item Interactions with Character

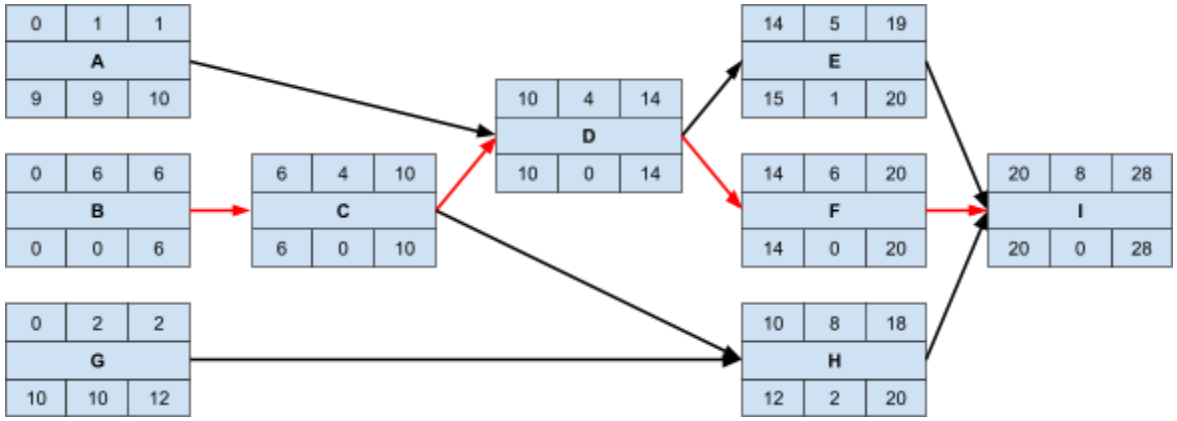
Whenever the character uses an inventory item (such as the Brain Blast Bar), they should be affected somehow. Specifically how they are affected depends on the type of item. For this reason, I will run multiple tests on the different inventory items to ensure they all work properly. For the Brain Blast Bar, I will give the character a large amount of the item and have them try to use it repeatedly. Since the BBB causes the character to run backward, I will record the position of the character when they use the item and their final position. If for any of the trials, the final position is greater than the initial position then there is an issue with the feature.

5. Timeline _____/10

Work items

Task	Duration (Hours)	Predecessor Task(s)
A. Set up Unity	1	-
B. Requirements Collection	6	-
C. Design Item Sprites	4	B
D. Import Item Sprites Into Game	4	A, C
E. Program Item & Menu Interactions	6	D
F. Program Item & NPC Interactions	6	D
G. Create Item Sound Effects	2	-
H. Create Item Animations	8	C, G
I. Full Implementation Testing	8	E, F, H

Pert diagram



Gantt timeline