Name: Jackie Lee Mark _____/50

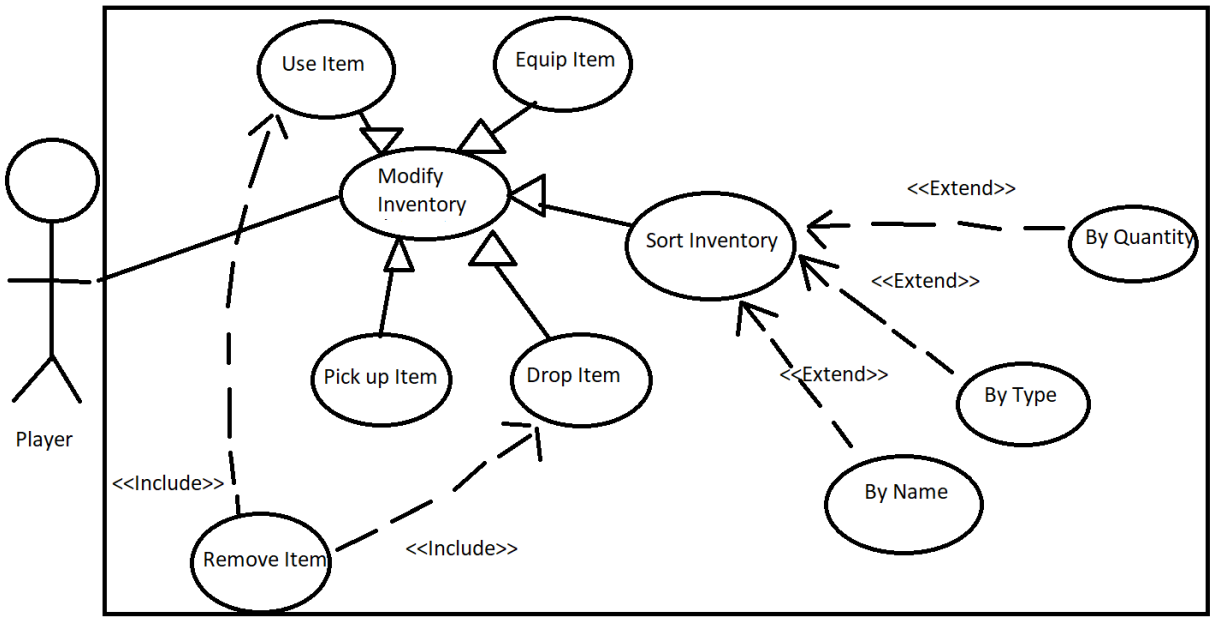
1. Brief introduction _/3

My role is to implement the inventory system for the main character. I will be designing an inventory screen that will organize the collectables and items that the player will find throughout the game. The player should be able to place items into their inventory and be able to drop them.

At the start of the game, the player will have an empty inventory. The inventory will have a maximum limit such that it cannot be overstocked with collectibles. The items picked up in a level will be carried on to the next level which will benefit players to explore the level. As TL2, I will be coding in features that will make navigating the inventory easier with sorting. The items associated in the inventory will also have to be coded so they operate correctly.

2. Use case diagram with scenario _14

Use Case Diagrams



Scenarios

[You will need a scenario for each use case]

Name: Use Item

Summary: The player uses a consumable i.e a potion.

Actors: Player

Preconditions: Consumable must be available for use and provide an effect

Basic sequence:

Step 1: Select Item

Step 2: Select Use Item

Step 3: Apply item effect

Exceptions:

Step 1: Item has no effect (using a potion at full hp)

Post conditions: Item quantity goes down, item effect is applied

Priority: 2*

ID: S01

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

Name: Drop Item

Summary: The player discards an item on the ground (i.e a book).

Actors: Player

Preconditions: Item must be a non-key item and be available in inventory.

Basic sequence:

Step 1: Select Item

Step 2: Select Drop Item

Step 3: Item Quantity Decreased by 1

Step 4: Item appears in overworld

Exceptions:

Step 1: Item has to not be a key item

Post conditions: Item quantity goes down, appears in the overworld on the floor

Priority: 2*

ID: S02

Name: Pick Up Item

Summary: The player picks up an item to add to their inventory.

Actors: Player

Preconditions: Inventory must not be full, and item can be picked up.

Basic sequence:

Step 1: Pick up Item on floor

Step 2: Adds quantity plus one for that item in inventory

Step 3: Takes up a slot in inventory

Exceptions:

Step 1: Inventory is full

Step 2: Item cannot be picked up (items that are not collectables

Post conditions: Item appears in inventory, inventory space is filled up by one

Priority: 2*

ID: S03

Name: Equip Item
Summary: The player equips an item in their inventory (accessories)
Actors: Player
Preconditions: Item is equipable and the player can equip said item (i.e does not have an accessory equipped already).
Basic sequence:

Step 1: Select Item
Step 2: Select Equip Item
Step 3: Apply item effect (if any)
Step 4: Display item equipped on player (if applicable)

Exceptions:

Step 1: Item cannot be equipped

Post conditions: Item quantity remains the same, displays that it is currently equipped on the player.
Priority: 2*
ID: S04

Name: Sort Inventory
Summary: The system sorts the inventory based on the player's preferences
Actors: Player
Preconditions: Inventory is not empty
Basic sequence:

Step 1: Select Sort Inventory
Step 2: Choose between different sort options
Step 3: Inventory will be sorted based on option selected.

Exceptions:

Step 1: Inventory is not empty

Post conditions: Inventory is sorted based on option selected.
Priority: 3*
ID: S05

Basic sequence:

Step 1: Accept input of first number.
Step 2: Continue to accept numbers until [calculate] is entered.
Step 3: Accept calculate command.
Step 4: Calculate and show result.

Exceptions:

Step 1: [calculate] is pressed before any input: Display 0.

Step 2: A button other than [calculate] or a number input is pressed: ignore input.

Post conditions: Calculated value is displayed.

Priority: 2*

ID: S06

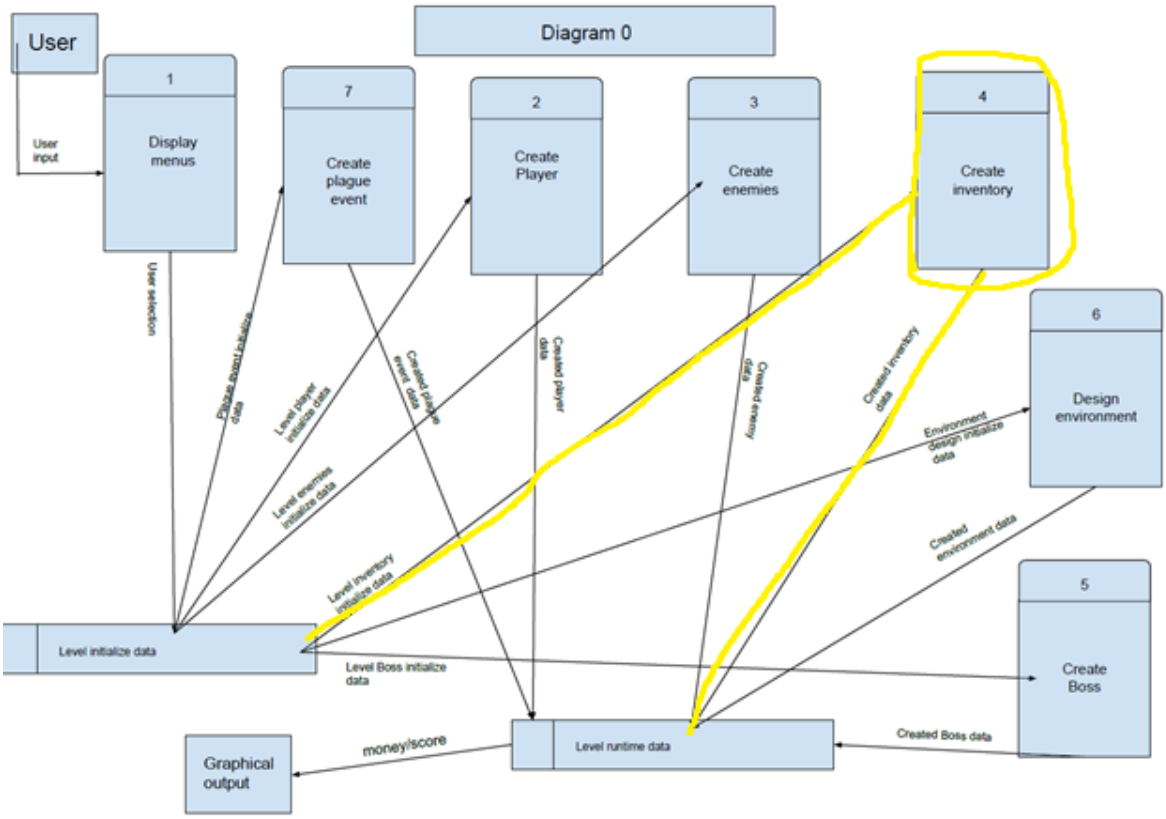
*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

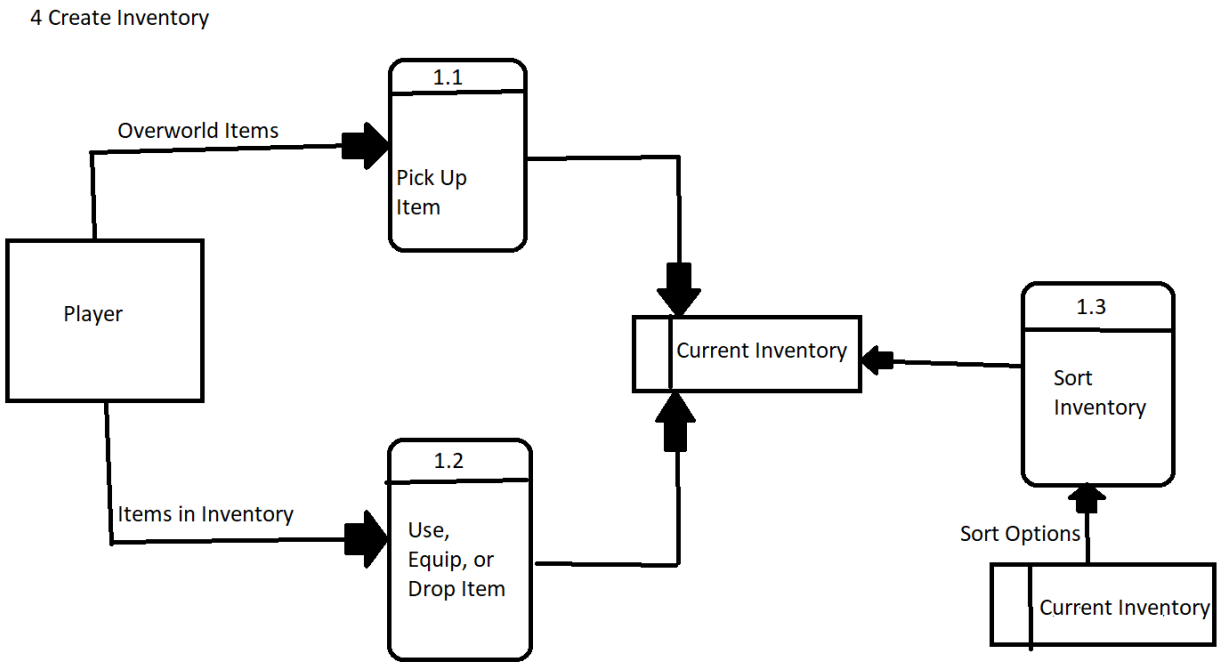
3. Data Flow diagram(s) from Level 0 to process description for your feature _____14

[Get the Level 0 from your team. Highlight the path to your feature]

Example:

Data Flow Diagrams





Process Descriptions

Create Inventory:

- If interacting with a overworld item, you can pick it up if your inventory is full otherwise it will not be picked up
- With items in your inventory you can choose to use an item if it is a consumable or equip if it's a an accessory
    - Using or equipping an item that is not applicable will bring up an error that prevents the player from doing so
    - Dropping an item will clear a spot in their inventory as long as the item was not a key item. It will appear in the overworld so it can be picked up again.
- There will be sorting options that can be selected when the inventory is open, choosing one of the options will have the system sort based on what is selected unless the inventory is empty.

4. Acceptance Tests _____9

The predetermined value of the inventory should be max amount of items that the player can carry throughout the game. Picking up items will require checking to see that the inventory is not already full and that if it is picked up then the item will disappear from the overworld and take up room in their inventory. Many of the tests will involve seeing how an action impacts their current inventory and the overworld.

Example for Inventory Features

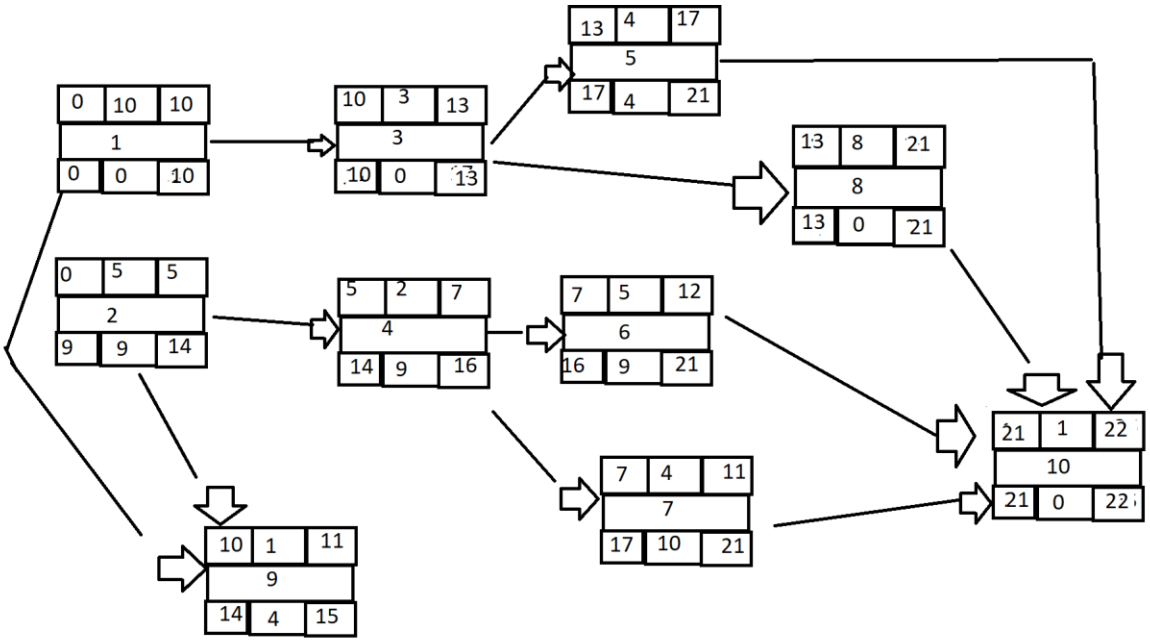
Output	Option	Notes
+1 Inventory space used Item disappears form the overworld	Pick Up	Inventory must not be full, otherwise display an error message
Equipped Item shows on Player Item is bordered showing it's equipped	Equip	Inventory space used remains the same, item must be equipable otherwise display an error message
Inventory space used goes down. Apply effect of used item	Use	Display error message if it's not a consumable.
Inventory Space used goes down. Item appears in overworld	Drop	Must not be a key item for progression, display error message if so
Inventory sorted	Sort	Display an error message if trying to sort an empty inventory

5. Timeline _____/10

Work items

Task	Duration (PWks)	Predecessor Task(s)
1. Design Items and Effects	10	-
2. Design Inventory Interface	5	-
3. Implement Item Sprites/Images	3	1
4. Implement Item Effects/Pickup	2	2
5. Implement Item Use	4	3
6. Implement Item Drop	5	4
7. Implement Item Equip	4	4
8. Implement Item Sort	8	3
9. Documentation	1	1, 2
10. Installation	1	5, 6, 7, 8

Pert diagram



Gantt timeline (Red: Work Hours, Blue: Slack Time)