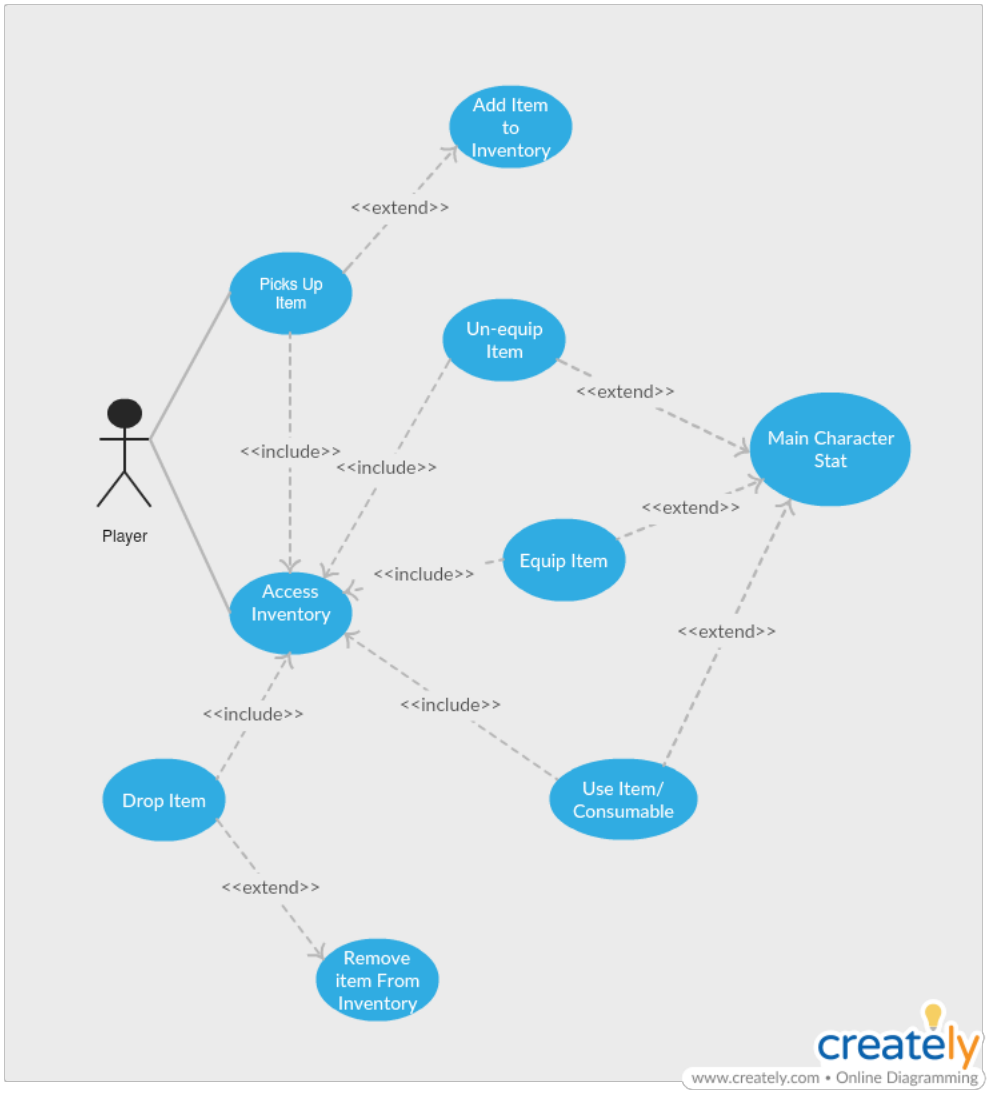
Name_____Kyle Hild_____ Mark _____/50

1. Brief introduction _/3

My feature that I am building is the item system of the game. This system will describe how the player will interact with items from equipping, un-equipping, dropping, using, storing, and picking up items in the game world.

2. Use case diagram with scenario _14

Use Case Diagrams



Scenarios

Name: Picks Up item

Summary: The Player will pick up items in the world

Actors: Player
Preconditions: Item is found in the world.
Basic sequence:
Step 1: Player encounters the item
Step 2: Player presses the appropriate key to pick up the item
Step 3: Adds the item to the players inventory if there is room
Exceptions:
Step 1: There is not enough room in players inventory: Error message explaining
Post conditions: Item is in inventory
Priority: 1*
ID: C01
*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

Name: Access Inventory
Summary: Player will be able to access their items to drop, equip, un-equip, and use items
Actors: Player
Preconditions: Player presses their inventory key.
Basic sequence:
Step 1: Player opens up their inventory
Step 2: Player can select items to manipulate and see visually their inventory
Exceptions:
Step 1: If inventory is empty then display empty inventory
Post conditions: Item is in inventory
Priority: 1*
ID: C02
*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

Name: Drop Item
Summary: The player Access's their inventory then selects drop item. The item will then be removed from their inventory.
Actors: Player
Preconditions: Player Access' their inventory, selects an item to drop.
Basic sequence:
Step 1: Player access' their inventory
Step 2: Player selects item
Step 3: Player selects drop item
Exceptions:
Step 1: Drop Item is accidentally selected

Step 2: Display error message before dropping the item to make sure this is what the player intended

Post conditions: Item is removed from inventory freeing up room

Priority: 2*

ID: C03

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

Name: Un-Equip Item

Summary: The Player will be able to Un-Equip an item that have currently equipped

Actors: Player

Preconditions: The player no longer wants the item equipped maybe to remove a negative status effect or change items to fit the situation

Basic sequence:

Step 1: Player access' their inventory

Step 2: Player can view their equipped items

Step 3: Player can select the item they wish to un-equip

Step 4: Player un-equips an item

Exceptions:

Step 1: If the item is not equipped, Tell the player the item they selected is not equipped

Step 2: Player accidentally selects un-equip item: Move item to inventory

Step 3: Player selects un-equip item and their inventory is full: Tell the player that action is not allows as their inventory is full

Post conditions: Item is in inventory, Character Status is adjusted accordingly

Priority: 2*

ID: C04

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

Name: Equip Item

Summary: The player can select an item and equip it

Actors: Player

Preconditions: Item in their inventory is equipped.

Basic sequence:

Step 1: Player selects item to equip

Step 2: Player equips the item

Step 3: Removes from inventory to show equipped status

Step 4: Change the player stats accordingly

Exceptions:

Step 1: There is an item already equipped: Error message asking if they would like to swap the items

Post conditions: Item is in the equipped slot with character stats changed accordingly

Priority: 2*

ID: C05

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

Name: Use Item/Consumable

Summary: The player can select a consumable item and use it

Actors: Player

Preconditions: Consumable item must be in their inventory, and they cannot already have the same consumable effect applied to them.

Basic sequence:

Step 1: Player access' their inventory

Step 2: Player selects item to use

Step 3: Player selects use item

Step 4: Item is destroyed and status is applied

Exceptions:

Step 1: They already have the consumable effect applied to them: Give them an error message explaining that the effect has been applied.

Post conditions: Item is destroyed, removed from inventory, and status applied to player.

Priority: 3*

ID: C06

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

Name: Main Character Stat

Summary: The Players Character stats are effected by their actions

Actors: Player

Preconditions: Item is used, equipped, or un-equpped

Basic sequence:

Step 1: Player selects item to equip, use, or un-equip

Step 2: Player status' change accordingly

Exceptions:

Step 1: If an action cannot be performed for any reason: Display error message do not change stats

Post conditions: Character Stats' are changed

Priority: 1*

ID: C07

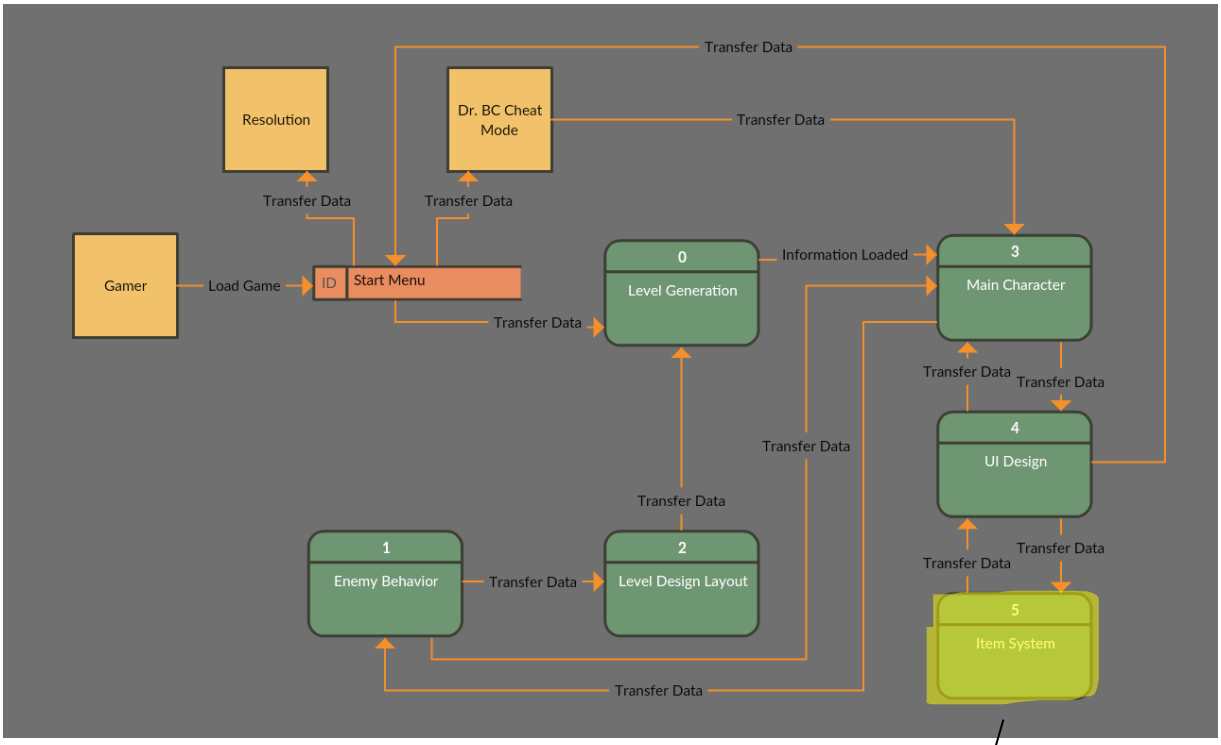
*The priorities are 1 = must have, 2 = essential, 3 = nice to have.

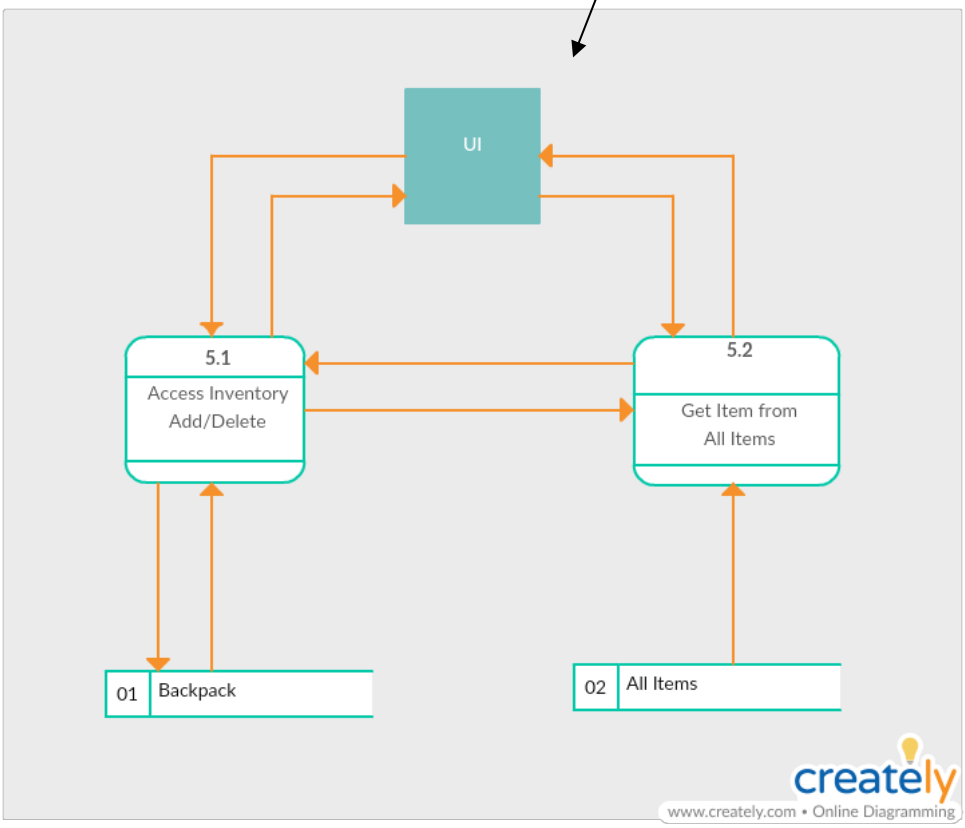
3. Data Flow diagram(s) from Level 0 to process description for your feature _____14

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

Example:

Data Flow Diagrams





Process Descriptions

Access Inventory*:

IF UI needs to access inventory
Traverse backpack and display information
END IF
IF UI needs to add an item to backpack
Send an item key to GET ITEM FROM ALL ITEMS
Then add it to empty backpack slot
END IF
IF UI needs to delete an item
Find item in backpack and delete
ENDIF
IF item is being equipped
Change item status to equip
return item info
END IF

Get Items From All Items*:

IF UI Needs Item Info
Get item info from All Items
Return item info
END IF

IF Access Inventory Needs Item Info
Get item info from All Items
Return item info
END IF

4. Acceptance Tests _____9

Access Inventory Add/Delete

Add every item to every slot in the back pack
(Test communication with Get Item From All Items)

The output should (essentially testing bounds)

- Display every item in every slot
- No two items should be in the same slot
- Backpack should not be overfilled
- Every slot in the backpack will have something in it

Get items From All Items

Be able to pull items from All Items and send the item information.

The input should be able to take an item key and return the correct item information

The output should be

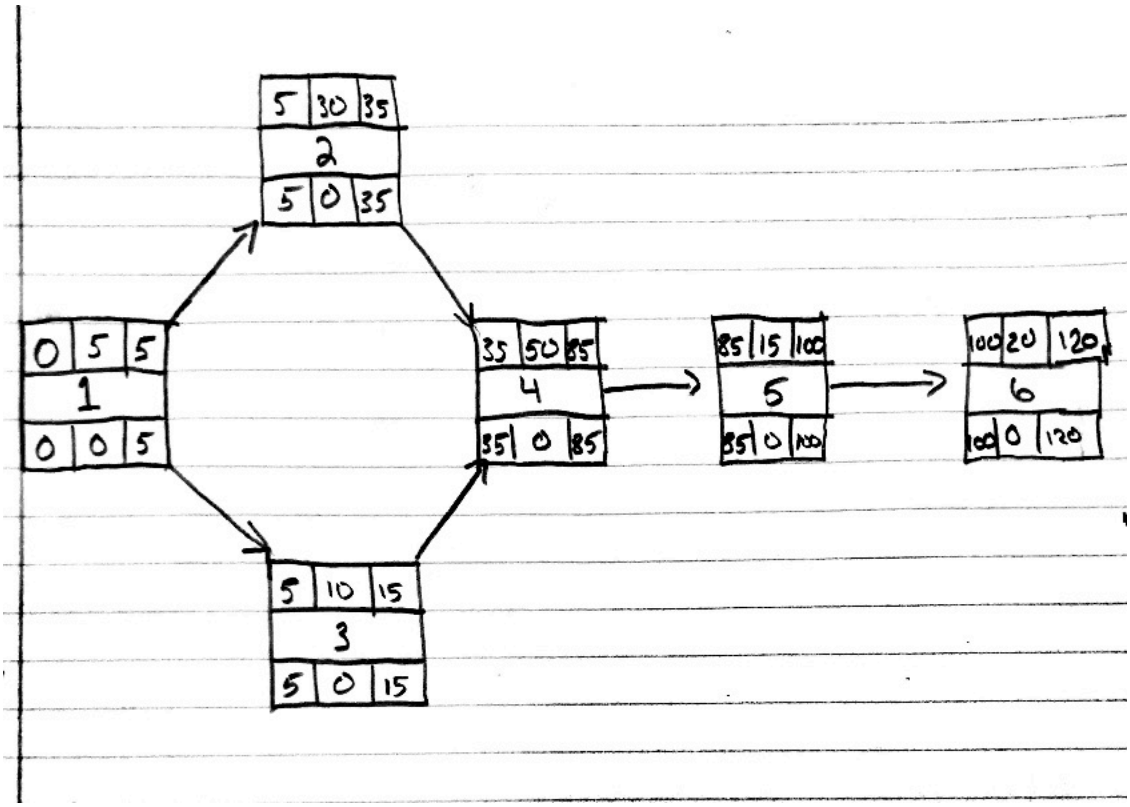
- Correct item info for every key generated

5. Timeline _____/10

Work items

Task	Duration Hours	Predecessor Task(s)
1. Collect Required information	5	-
2. Construct all Item Database	30	1
3. Construct Backpack Database	10	1
4. Programming	50	2,3
5 Testing	15	4
6. Installation of Item System	20	5

Pert diagram



Gantt timeline

	predicted time(hrs)	key	completed	this week	planned
Kyle Hild		hours	5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100 105 110 115 120		
Collect Required Information	5				
Construct Item Database	30				
Construct Backback Data	10				
Programming	50				
Testing	15				
Installation of Item System	20				