**Requirement**

***Requirement ID: 1*** *Create Inventory Item*

**Use Case:** *Inventory Management*

**Rationale:** *The customer will need to enter the inventory items for the system to manage the inventory.*

**User Requirement:** *The System shall allow the Inventory Manager or Operations Manager to create an inventory item.*

***System Requirements*:**

1. *The System shall display the item creation screen.*
2. *The system shall accept item name, quantity, and optional low-level number as user inputs.*
3. *The user shall be allowed to validate input and edit inputs before submission.*
4. *The system shall add the item to the inventory database if the item does not exist or increment the item’s quantity in the database if it was already created.*
5. *The system shall return a confirmation message if the inventory database has been successfully updated.*

**Acceptance Criteria:** *The user shall be able to add item in database 100% of the time after having entered at least the item name and quantity fields.*

**Relates to/Dependencies:** *N/A*

**Priority:** *High*

**Team Owner:** *Marlon Lewis*

**Acceptance Test**

**Input**: [“white beads”, 12], [“white beads”, 8], [“red beads”, 15, 5], [“black beads”], [“blue beads”, 15, 4, 5]

**Process**:

1. Check for number of inputs.
2. If the number of inputs is less that 2, return “error”.
3. If the number of inputs is greater than 3, return “error”.
4. If 2 inputs, check if first input is string and 2nd is positive integer
   1. If input types check fails, return “error”.
   2. If input types are correct, check for matching string in database:
      1. If no matching string found, add new record to database where:
         1. Item name is the first input.
         2. The quantity is the second input.
         3. The low-level value is set to the ceiling-division of second input/4.
      2. If matching string found, return inputs as list.
5. If 3 inputs, check if first input is string, 2nd is positive integer and 3rd is positive integer greater than zero:
   1. If input types check fails, return “error”.
   2. If input types are correct, check for matching string in database:
      1. If no matching string found, add new record to database where:
         1. Item name is the first input.
         2. The quantity is the second input.
            1. The low-level value is set to the third integer input.
      2. If matching string found, return inputs as list.

**Output**: Returns “OK” when successfully updated, “error” when failed to create item and list obect containing inputs when match found in database.