Use Case Name:

Primary Actor:

Precondition:

Main Flow:

Alternate Flows:

Postcondition:

Related Use Cases:

‘breevs

Farm to Database Table = FtDbT

Local Producer = L.P.

Volunteer = Vol.

Purchase Order = P.O.

Use Cases:

A L.P. is looking to sign up for the FtDbT service from their browser.

1. Primary Actor: L.P.
2. Precondition: must have internet access
3. Main Flow:
   1. L.P. navigates to FtDbT website.
   2. Select button to “join.”
   3. Select “Local Producer” button.
   4. Enters information:
      1. L.P. name
      2. Phone number
      3. Email address
      4. Address
      5. City
      6. State
      7. Zip
   5. Select “Submit” button.
4. Alternate Flows: N/A, user is forced to follow this flow.
5. Postcondition: User’s profile is created in DB. Browser is automatically directed to “add inventory” page.
6. Related Use Cases:
   1. L.P. or school logs in
   2. \*Possibly\* signing up a School but should be an application process which is more heavily vetted.

An existing L.P. wants to add food to Hub.

1. Use Case Name: Add Food to Hub
2. Primary Actor: L.P.
3. Precondition: L.P. must be an existing supplier in the system, be signed in and on the “Add Food” page in their browser.
4. Main Flow:
   1. L.P. enters following information about product.
      1. Name
      2. Quantity
      3. Unit
      4. Macro
      5. Harvest Date
      6. Expiration Date
      7. Price Per Unit
   2. Selects “Add” button – product will be stored in temp dictionary.
   3. Option to edit, add another item or submit final form will be selected.
5. Alternate Flows:
   1. If L.P. SELECTS EDIT at step c.
      1. L.P. is brought to step a with previously entered information populating form.
      2. L.P. makes necessary edits.
      3. L.P. will select submit when complete.
   2. If L.P. SELECTS ADD ANOTHER ITEM at step c.
      1. L.P. is brought back to step a.
      2. Main flow loops until L.P. selects submit.
   3. If L.P. SELECTS SUBMIT at step c.
      1. Products temp dictionary will be submitted to the “food” table.
6. Postcondition: New products in temp dictionary are added to the “food” table.
7. Related Use Cases:
   1. Schools purchasing food from the Hub.
   2. Food item has passed expiration date and needs to be removed from table.

Food item has passed expiration date and needs to be removed from “food” table.

1. Use Case Name: expired food item
2. Primary Actor: DB system/Vol.
3. Precondition: Food table must have item that has passed “expiration date”
4. Main Flow:
   1. Vol. will run SQL command to delete all expired food items from the DB.
   2. FtDbT DB will print to console a list of the deleted item(s).
   3. Food Item(s) will no longer be shown to schools querying FtDbT DB.
5. Alternate Flows:
   1. FtDbT DB system runs python script automatically once a day to execute Main Flow.
   2. FtDbT DB system will print removed food item(s) to file and email to all Vol’s and removed item’s L.P.
6. Postcondition: Expired food item(s) will no longer be in FtDbT DB.
7. Related Use Cases: N/A

School is looking to purchase food from Hub.

1. Use Case Name: School Purchase
2. Primary Actor: School
3. Precondition: School is registered in FtDbT system, and in account
4. Main Flow:
   1. School logs into account
5. Alternate Flows:
6. Postcondition:
7. Related Use Cases:

\*\*School is looking to purchase food from Hub.\*\*

\*\*Primary Actor:\*\* School

\*\*Precondition:\*\* School must be registered user in DB.

\*\*Main Flow:\*\*

1. School navigates to "purchase" section of website

2. School is able to navigate through the entire list of current inventory, or search for specific food items in current inventory

3. School selects items to "add to cart"

4. School selects "Purchase/Check out" button

5. P.O. is generated

\*\*Alternate Flows:\*\* N/A, School is forced to follow this flow.

\*\*Post-condition:\*\* Order will be submitted to Hub, with food item amounts in Hub updated as items are placed in the schools "cart"

\*\*Related Use Cases:\*\* P.O. generation