Use case: *VisualizeGraphs*

Primary actor: TCNJ Shop Employee
Goal in context: To view statistical data about different frozen food item popularity, the frequency of visits by varying demographics and communities, and the number of meals ordered compared to the number of people of the account-holder's household. Data will be visualized in the form of scatter-plots, as well as potentially bar graphs, and the user will be able to choose which graph they want to view.
Preconditions: Enough data has been collected by the application to be represented in graphical form without being biased or inaccurate.
Trigger: The TCNJ Shop Employee requests to view a graph of their choice through the graph user interface.
Scenario:
1. TCNJ Shop Employee accesses the graph user interface (GraphUI)
2. TCNJ Shop Employee chooses a graph that is available from a dropdown menu.
3. TCNJ Shop Employee selects "View Graph"
4. TCNJ Shop Employee views the graph that they have chosen, as well as a best fit line if applicable.

Exceptions:
1. *View Graph* is selected but TCNJ Shop Employee is not verified by TCNJ Shop: User is not permitted to access data graphs until the user is verified by TCNJ Shop.
2. *View Graph* is selected without a graph chosen in the dropdown menu: TCNJ Shop Employee is told to choose a graph to be displayed.
3. *View Graph* is selected with a graph chosen in the dropdown menu: TCNJ Shop Employee may view the graph, as well as a best fit line if applicable.

Priority: Medium-High, can be implemented after data is collected.
When available: End of implementation phase
Frequency of use: Several times per day.
Channel to actor: GraphUI
Secondary actors: Support technician
Channels to secondary actors: AdminUI
Open issues:
1. Is there a way to view the position of data points on the trendline and interact with the data, or is it a static image?
2. Can a user switch between graphs while viewing one?

Use Case: *OrderFood*

Primary actor: User with Food Insecurity
Goal in context: To select any number of meals the user would like to pick up the TCNJ Shop, as well as the type of each meal.
Preconditions: The user has created a basic account and hasn't yet placed an order within 24 hours (the flag "OrderedToday" should be false).
Trigger: The user with food insecurity decides to place an order for the day.
Scenario:
1. User with food insecurity selects the "Place Order" button.
2. User with food insecurity views the OrderUI screen.

3. User with food insecurity may press any of the "plus" or "minus" buttons next to each type of meal (frozen fish, frozen chicken) to add or remove a meal of that type to or from their cart. They may do this any number of times less than than the Shop's orderLimit variable (can be changed by TCNJ Shop Employee).
4. User presses the "Order" button to confirm the meals they selected.
5. User observes a confirmation message alerting them to when and where they can pick up their meals.
6. User presses the "Go back to main menu" button to return to the MainUI screen.

Exceptions:
1. Cart is empty and the *Order* button is pressed: User is told they must add at least one item to their shopping cart.
2. Cart is empty and the *Minus* button is pressed for one of the meals: User is told that they may not add a negative number of meals to their cart.
3. Cart is full and the *Plus* button is pressed for one of the meals: User is told that they may not exceed the Shop's orderLimit variable.

Priority: High, must be implemented for the program to accomplish its task.
When available: As soon as possible.
Frequency of use: Many times per day (once per day for each user)
Channel to actor: OrderUI
Secondary actors: Support technician
Channels to secondary actors: AdminUI
Open issues:
1. Is it possible to press the "plus" and "minus" buttons simultaneously, and if so, what happens?

Use Case: *ViewOrders*

Primary actor: TCNJ Shop Employee
Goal in context: To view a chronologically organized list of which orders have been recently placed for which meals, as well as who (which account) placed the meal.
Preconditions: System has been programmed to accept OrderInfo objects.
Trigger: The TCNJ Shop Employee requests to view the list of OrderInfo objects through the OrderListUI screen.
Scenario:
1. The TCNJ Shop Employee views the OrderListUI screen.
2. The TCNJ Shop Employee views the table on this screen representing the organized data for recent orders that have been placed.

Exceptions:
1. The list of OrderInfo objects is empty: Message is displayed "No orders have been placed yet."

Priority: High, required for TCNJ Shop Employee
When available: As soon as possible.
Frequency of use: Several times per day
Channel to actor: OrderListUI
Secondary actors: Support Technician
Channels to secondary actors: AdminUI
Open issues:
1. Is there a way to delete an order from the OrderList?

Use Case: *SetLimit*

Primary actor: TCNJ Shop Employee
Goal in context: To set a universal limit on the number of meals a user with food insecurity can add to their cart when placing an order.
Preconditions: Another TCNJ Shop Employee is not currently changing the value for the orderLimit simultaneously.
Trigger: The TCNJ Shop Employee decides to limit the cart when deemed appropriate.
Scenario:
1. The TCNJ Shop Employee presses the button "Edit Cart Limit" on the MainUI screen.
2. The TCNJ Shop Employee enters an integer into the text box.
3. The TCNJ Shop Employee presses the button "Submit" to close the editable text box.

Exceptions:
1. No value was entered in the text box when editing the cart limit: An error message is displayed and the user is prompted to enter a value.
2. A non-integer value was entered in the text box when editing the cart limit: An error message is displayed and the user is prompted to enter an integer.
3. An integer value less than or equal to zero was entered in the text box when editing the cart limit: An error message is displayed and the user is prompted to enter an integer greater than 1.
4. User closes the window or tab while editing the cart limit: Any value entered for the cart limit is not saved and must be reentered.

Priority: Low, can be replaced by a default value
When available: Near to deployment or afterwards
Frequency of use: Once or twice per day
Channel to actor: MainUI
Secondary actors: Support technician
Channels to secondary actors: AdminUI
Open issues:
1. Should there be a limit to how many characters are entered in the text box.

Use Case: *RemoveAccount*

Primary actor: TCNJ Shop Employee
Goal in context: Remove a user account that is misusing a system.
Preconditions: TCNJ Shop Employee is verified and allowed to remove user accounts.
Trigger: The TCNJ Shop Employee decides to delete a known account.
Scenario:
1. The TCNJ Shop Employee presses the button "Remove an account" on the MainUI screen.
2. The TCNJ Shop Employee enters an email address of a user with food insecurity.
3. The TCNJ Shop Employee presses the button "Remove" to remove the account.
4. The TCNJ Shop Employee observes the confirmation popup message.
5. The TCNJ Shop Employee presses the button "Confirm" to close the popup and editable text box or "Cancel" to edit the email that was entered.

6. An automatic email is sent to the user whose account was deleted, alerting them that their account was removed.

Exceptions:
1. An invalid email address is entered: Ask the TCNJ Shop Employee to reenter a correctly formatted email address.
2. Holder of the email address entered hasn't made an account with TCNJ-ShopRequest: Ask the TCNJ Shop Employee to enter the email address of a valid account holder.

Priority: Medium, can be implemented after deployment
When available: Near to deployment or afterwards.
Frequency of use: Up to several times per day.
Channel to actor: MainUI
Secondary actors: Support technician
Channels to secondary actors: AdminUI
Open issues:
1. Will there be a warning system that lets the misuser know of their offenses rather than immediately removing their account?
2. Will there be a text box allowing the TCNJ Shop Employee to explain a reason why they are removing an account.