Requirements Specification Document

RIP Mark

Best Seng Consulting

Nov 26, 2018

Revision History

Name	Date	Reason for Changes	Version
RSD	Oct 16, 2018	Rough Draft	0.9
RSD	Oct 30, 2018	Final Draft	0.9
RSD	Nov 7, 2018	Final Draft	1.0
RSD	Nov 8, 2018	Added storyboard to use cases	1.01
RSD	Nov 26, 2018	Final Copy	2.0

1 Introduction

1.1 Purpose

The UVic Food Bank & Free Store (FBFS) is UVic's own food bank for students and their families. Our client (JamStash) has pointed out several problems with the current inventory and data management system within the FBFS, and has asked us to assist them in finding a solution. Our goal is to implement a new web application system to solve the issues described below.

Currently, customers of the FBFS are not able to view the current inventory that is available without physically being in the store. By creating a web application, all customers will be able to view the current up to date inventory remotely, anytime. Admins and volunteers must also be able to manage inventory in real time to ensure the inventory in store is correctly reflected on the web application. Aside from being able to view and manage inventory, a critical issue which we are going to address, is the abuse of the current system in place at the FBFS. Since the current system is a pen and paper sign out sheet, some customers have found methods of unlawfully gaining more items than they are allowed. Therefore, customers of the FBFS must be monitored, and a flag on their account will be used to notify the volunteers and admins of unlawful habits. Additionally, gathering statistical data is a must, so admins can monitor inventory and usage statistics from an upper level management position.

Therefore, the purpose of this document is to thoroughly specify the web application solution, such that all issues stated above are resolved and can be smoothly executed in actual development of the product. This document (Requirements Specifications Document 2.0) will also play a key role in maintaining mutual understanding with our client. This document should be referred to and updated during development, and should be consulted when clarity of an issue is required.

1.2 Project Scope

The main objective is to have a proper system in place for the FBFS, so that customers can view which items are in store along with an inventory management system to allow admins and volunteers to add, update, or delete the inventory as it arrives and departs the FBFS. A volunteer and admin must also be able to store customer information and keep track of their use in order catch customers who are abusing the system by taking out more items than allowed, or by using a fraudulent V-Number. The benefits of this software include ease of use with minimal training, a convenient way to track inventory, and tracking customer transactions to curb abuse of the FBFS. Overall, the client's goals are to have an easy to use system for all users (Customers, Volunteers and Admins), a secure system which meets the client's standards, along with the other requirements as specified below.

1.3 Glossary of Terms

Term	Definition
Access Control	Controlling who has access to data
Admin	A super user of the system, who monitors the Food Bank & Free Store
Anytime	365 days per year 24 hours per day
Client	JamStash organization and their employees
Customer	Students and their families
Data	User information and family information including name and V-Number
Device	Cell phone, laptop, tablet or computer
Donor	Someone who gives food, or household items and clothing to the Food Bank & Free Store
Employee	Admin or Volunteer at the Food Bank & Free Store
Family	A customer who has family status
FBFS	The UVic Food Bank & Free Store
Honour System	A system where someone's word is assumed true
Inventory	Items in possession of the Food Bank & Free Store
Inventory Database	Database storing inventory information
Lock out	When one user accesses a resource so no one else can then access that resource
Manage	Includes viewing, creating, updating, and deleting
Must	Required to have
OneCard	Student card with student number on it
Real time	Simulated immediate response of one minute as specified in NFR-2
Service	The process and facility to get stuff from the Food Bank & Free Store
Should	Optional
Statistical Data	Data gathered from customers this data includes: Gender, International/Local Student, Family, Meal Plan, Grad/Undergrad
System	Software Implementation
Tester	The person testing the software
UI	User Interface
User	Admin, Customer, or Volunteer
UVic	University of Victoria
UVic Database	Database storing account and report information
V-Number	UVic Student ID Number
Volunteer	An unpaid employee of the Food Bank and Free Store
Web Application	Our system, made in an internet accessible format
Will	Feature/function will be implemented in the final product

1.4 References

[1]   "Jamstash", September, 2018. [Online]. Available:
        https://sites.google.com/view/jamstash/ . [Accessed: Sept. 24, 2018].

[2]   UVic Protection of Privacy Policy, June, 2017.[Online]. Available:
        https://www.uvic.ca/universitysecretary/assets/docs/policies/GV0235.pdf
        [Accessed: Sept 24, 2018]

1.5 Overview

Together the RSD is made up of seven total sections that outline our web application solution for JamStash. The first section is a brief overview and overall description of the project scope, along with a glossary of terms and references. The second section describes the context and origin of the system being specified. The third section summarizes the major behaviors and functional requirements of the system and includes use cases with their associated storyboards. The fourth section covers all the external interface requirements specified by the client. The fifth section covers the non functional requirements of the project. The sixth section contains the use case model, entity relationship diagram, data flow diagrams, traceability matrix, test cases and sequence diagrams. Lastly, the seventh section is the appendix, which contains a link to the interactive prototype.

2 Overall Description

2.1 Product Perspective

The requested software is designed to be a web application that replaces the current system in place at the FBFS. The client will not have to upgrade or integrate the old system in anyway, aside from importing account information from the old system. The new system must provide the user with the functionality and features required, as outlined in the client's RFP [1] and in more detail within this document.

2.2 Product Features

There are many features as per the client's request. These features include: allowing users to view the current inventory of the FBFS at any time and location, volunteers and admins are able to update the current inventory in real time, volunteers and admins are able to manage customer information and track a customer's usage of the FBFS, statistical information is stored safely on UVic servers and accessible in a readable form.

2.3 User Classes and Characteristics

It is important to note that all users (admins, customers and volunteers) are UVic students. customers are the primary user class using the web application, they will be checking their weekly limit and viewing the inventory. The admins and volunteers will be the ones managing the web application, so it important that they have a good understanding of how the system will work. Nowadays, it is safe to assume that most users will be familiar with how to use a basic web application.

To ensure that users know how to use the web application, admins will be trained by B.S. Consulting. It will then be the admins responsibility to train new volunteers on how to use the web application for managing inventory and customer accounts.

2.4 Operating Environment

Google Chrome and Internet Explorer are the recommended browsers as they can be accessed on both desktop/laptop computers and mobile devices (See Section 4.4 for more details). The client has specified that customer data must be kept on a server on campus according to the UVic privacy policy. Since the UVic database is being hosted by UVic, we cannot control the operating environment of the servers, therefore the server operating environment is to be considered out of scope for this part of the project.

2.5 Design and Implementation Constraints

A major constraint of the project is following the data storing policies of UVic. Since the FBFS is located on the UVic campus, the rules and regulations state that the server must be hosted by UVic on UVic servers and adhere to the UVic policies about data protection and privacy [2].

2.6 Assumptions and Dependencies

One of the assumptions is that the data collected and stored on UVic servers adheres to their policies outlined in the UVic Protection of Privacy Policy [2]. This means all user data is dependent on the UVic servers and its operating environment.

The next assumption is that our web application will be able to interface with the UVic servers when it comes to features that manage accounts and inventory. Since this information will all be stored on the UVic servers, the system must be able to access V-Numbers, and have the ability to modify FBFS's food inventory at anytime. Lastly, as mentioned in section 2.3 of the RSD, it is assumed most customers can use a basic web application. This assumption can be made since all users of the system are UVic students and will have had to interact with the UVic website in order to register as an active student. Therefore confirming that they have indeed used at least one basic web application.

3 System Features

3.1 Online Inventory

3.1.1 Description and Priority

The inventory of the FBFS will be viewable on the web application, which will show the current inventory of the FBFS to all users. Updates to the inventory made by volunteers and admins can also be seen in real time. This is of high priority to implement and it is essential to the system.

3.1.2 Functional Requirements

SREQ-1 Real Time Updating of Inventory: Users must have real time updates of the FBFS's inventory in order to see the current items available. This is a must for the client, as they need real time updates in order to track incoming and outgoing items, while still keeping customers updated with what items are in the store. Should the system fail to update, the inventory shown will be the most recent catalogue of the items that was stored on the database.

Rationale: Customers must be able to see the most recent catalogue of items so they know what is in the store. The necessity for this functional requirement comes from the client's RFP from section 2.0 [1].

SREQ-2 Searching the Inventory: Users must be able to search the online inventory through a search function, this function will search the inventory database for matching items to the users search entry. If the user inputs an invalid entry or an item that does not exist in the inventory database, they will be met with an error message.

Rationale: Customers, volunteers and admins should be able to search the stores inventory to see if a specific item is in the store. This will help with both inventory management and servicing customers.

SREQ-3 Managing Inventory: An admin or volunteer must be able to add, edit, and delete items from the inventory database.

Rationale: From the client's RFP, we need to be able to track inventory of the FBFS so that items can be added, removed, and changed to maintain accuracy.

SREQ-10 (Formally called NFR-3) Simultaneous Changes: When updating existing stock in the inventory database, or adding/removing any item, the inventory database must be locked out for the duration of the change. When making a change to user data, the UVic database will be locked out for the duration of the change.

Rationale: Multiple people changing the same data is a problem that needs to be dealt with, and lockout is the best way for us to deal with this problem.

3.1.3 Use Case(s) associated with the Feature or Functional Requirements

UC-3: Edit Existing Inventory
> **Actors:** Admin, Volunteer
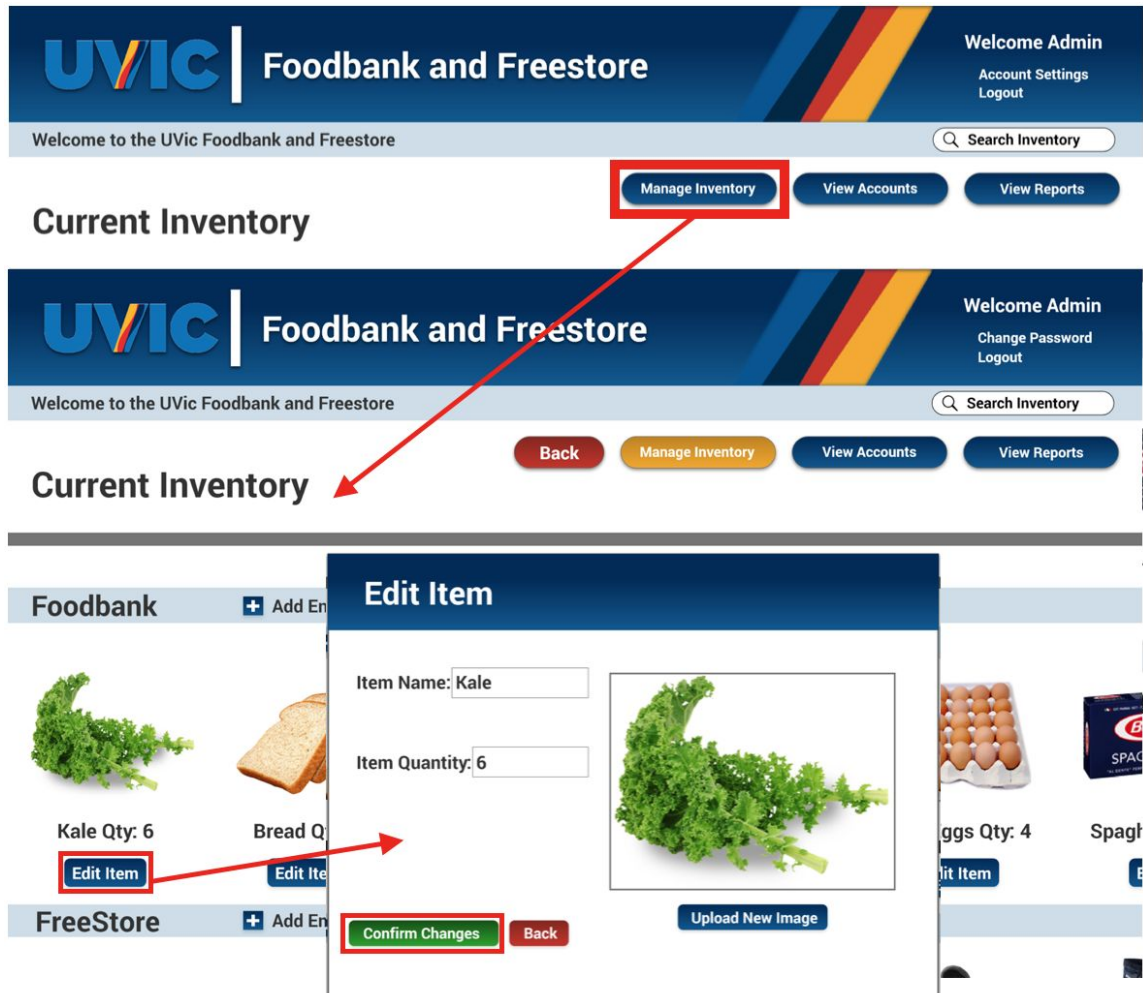> **Preconditions:** Using a web browser. Logged in as Admin/Volunteer
> **Steps:**
>> 1) Select 'Manage Inventory'
>> 2) Find the item that needs to be updated
>> 3) Select 'Edit' Item option
>> 4) Change the current amount to the desired amount. Change the name and picture associated with the item.
>
> **Success Conditions:** Inventory levels are correct; all items in the store display on the system.
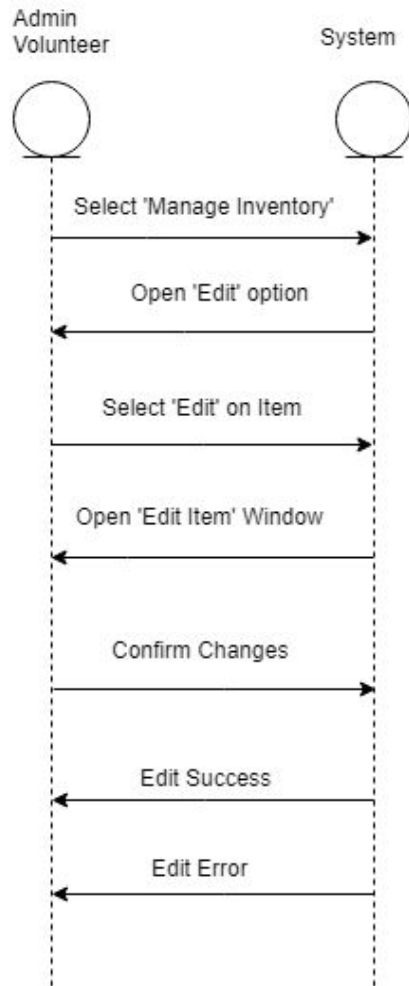> **Alternate Paths:** Item(s) did not change, as visible in view inventory. Repeat steps 1-4.
> **Storyboard:** Refer to Storyboard 3
> **Sequence Diagram:** Refer to Sequence Diagram 3

Storyboard 3: Editing an item in the current inventory

Sequence Diagram 3

UC-6: View Inventory

Actors: Admin, Volunteer, Customer
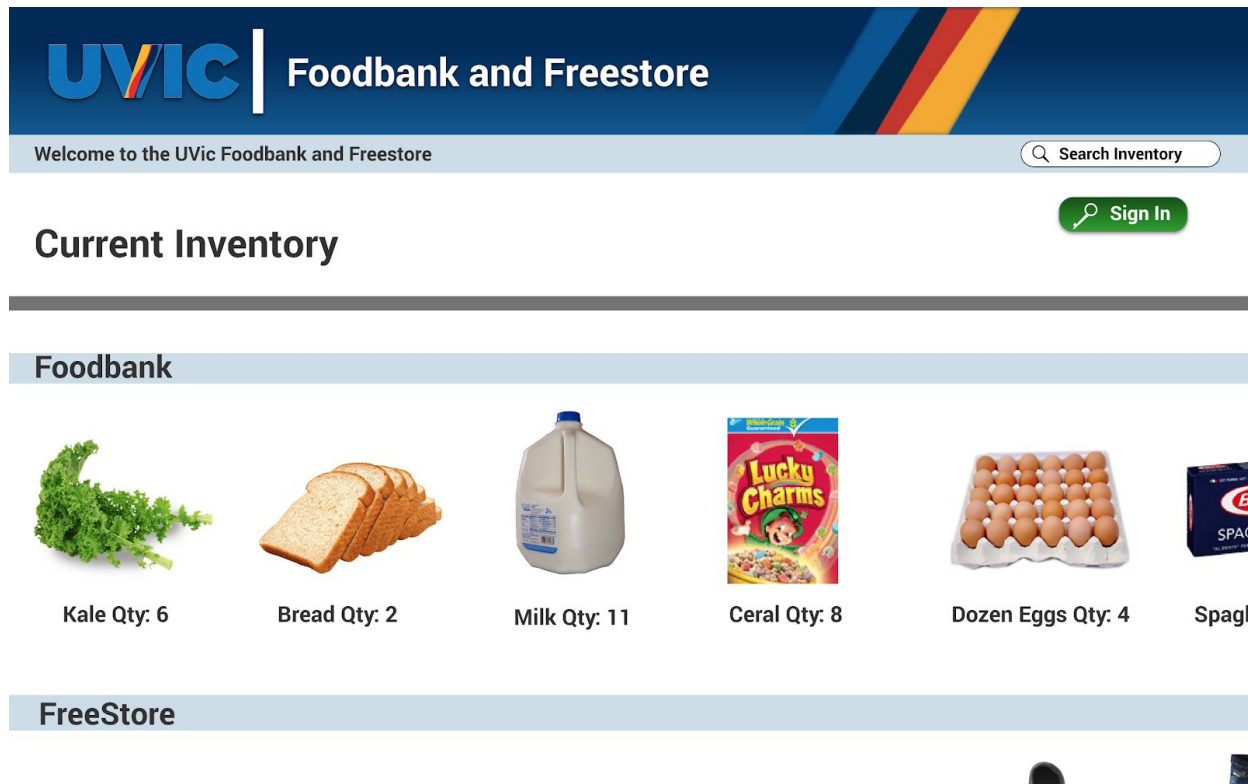**Preconditions:** Using a web browser. User does not have to be logged in.
Steps:
    1) Go to the FBFS web application
    2) View the current inventory
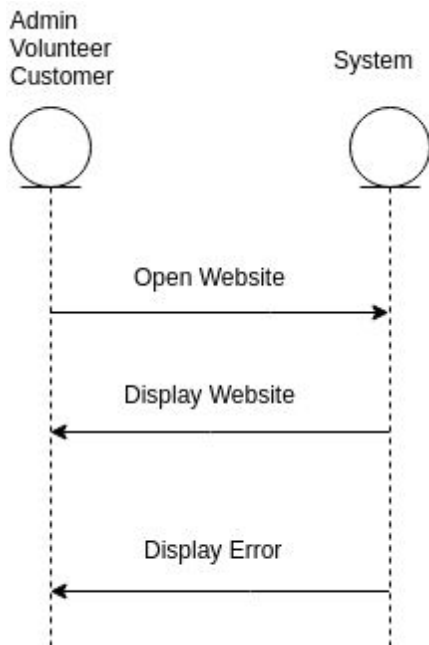**Success Condition:** Inventory is displayed on the main page
Alternate Paths: Inventory does not display. Repeat steps 1-2.
Storyboard: Refer to Storyboard 6
Sequence Diagram: Refer to Sequence Diagram 6

Storyboard 6: Viewing the current inventory

Sequence Diagram 6

UC-4: Add Item to Inventory

Actors: Admin, Volunteer

Preconditions: Using a web browser. Logged in as Admin/Volunteer

Steps:
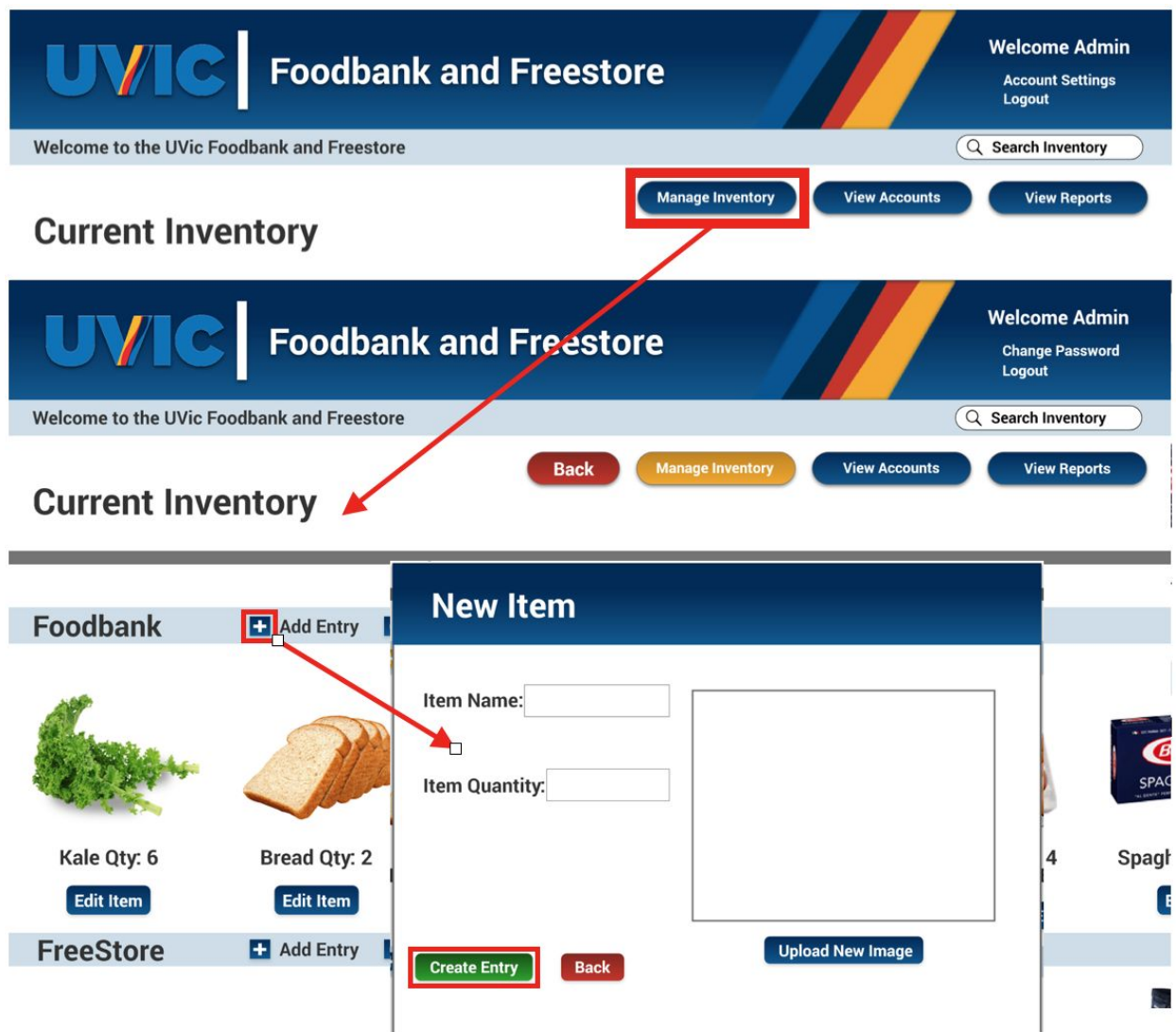
      1) Select 'Manage Inventory'

      2) Select 'Add Entry' option

      3) Add item name, quantity and picture.

      4) Publish new by selecting 'Create Entry'

Success Conditions: Inventory levels are correct; all items in the store display on the system.
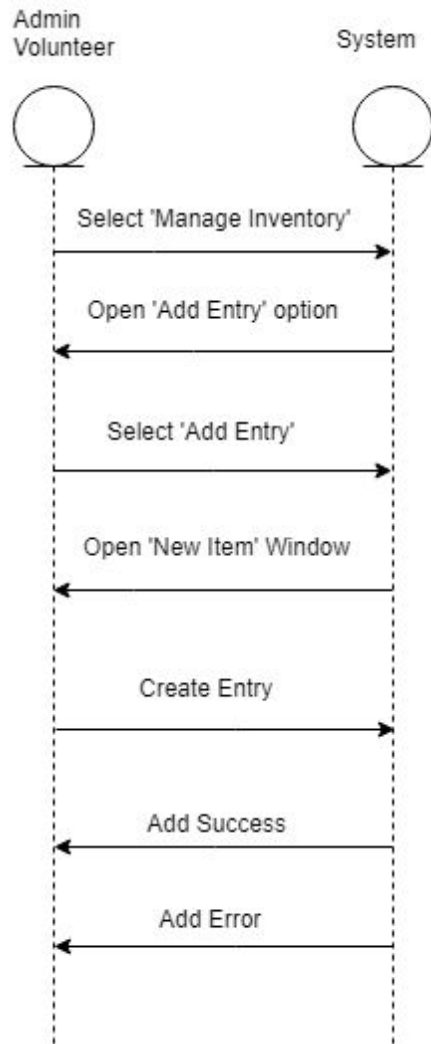
Alternate Paths: Item(s) did not change, as visible in view inventory. Repeat steps 1-4.

Storyboard: Refer to Storyboard 4

Sequence Diagram: Refer to Sequence Diagram 4

Storyboard 4: Adding items to the current inventory

Admin Volunteer   System

Select 'Manage Inventory'

Open 'Add Entry' option

Select 'Add Entry'

Open 'New Item' Window

Create Entry

Add Success

Add Error

Sequence Diagram 4

UC-5: Remove Item from Inventory

      Actors: Admin, Volunteer

   **Preconditions:** Using a web browser. Logged in as Admin/Volunteer

      Steps:

            1) Select 'Manage Inventory'

            2) Select 'Delete Entry' option

            3) Find the item that you want to remove

            4) Select 'Delete Item' to remove the item

      Success Conditions: Inventory levels are correct; all items in the store display on the system
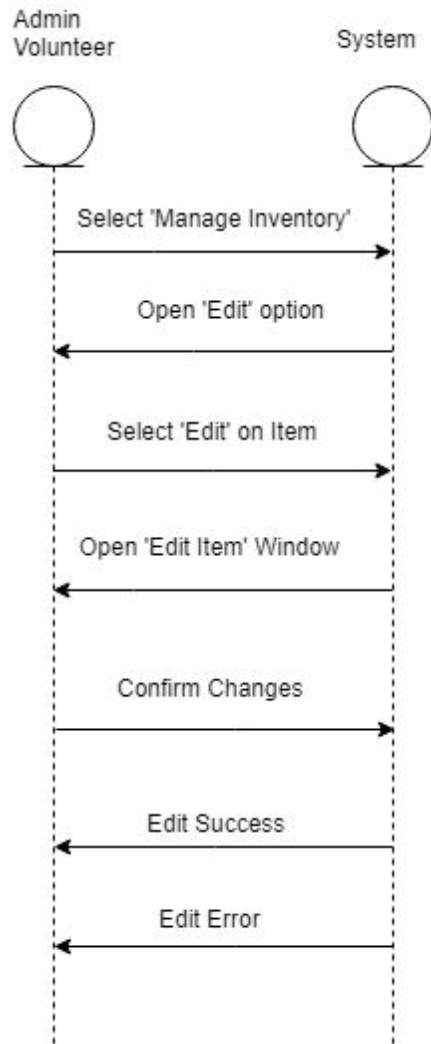
      Alternate Paths: Item(s) did not change, as visible in view inventory. Repeat steps 1-4.

      Storyboard: Refer to Storyboard 5

      Sequence Diagram: Refer to Sequence Diagram 5
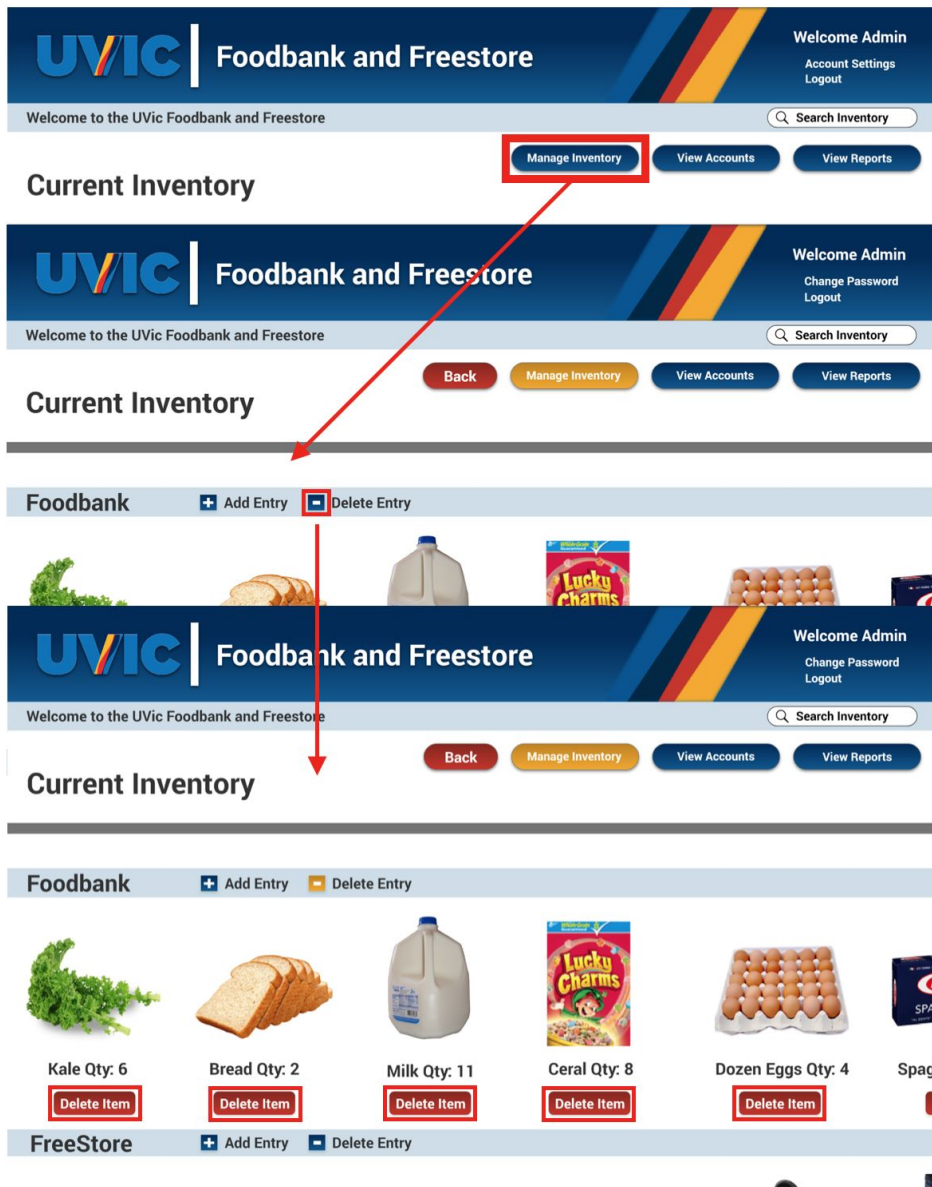
Storyboard 5: Removing items from the current inventory

Sequence Diagram 5

UC-7: Search for an Item:

    Actors: Admin, Volunteer, Customer

    Preconditions: Extends the "View Inventory" use case

    Steps:

        1) Click within the search bar

        2) Type in a search query

        3) Hit enter/return or click the magnifying glass to perform the search

    Success Conditions: User finds all items relating to the search terms

    Alternate Paths: No items found, error message appears or user searches the item manually in the catalog. Repeat steps 1-3.

    Storyboard: Refer to Storyboard 7

    Sequence Diagram: Refer to Sequence Diagram 7

Storyboard 7: Searching for an item

Sequence Diagram 7

3.2 Accounts

3.2.1 Description and Priority

Admins can create volunteer and other admin accounts provided a valid V-Number and OneCard is given. Both volunteers and admins can create an account for customers provided a valid V-Number and OneCard is given as well. The UVic database will be responsible for the account verification process by confirming the V-Number enter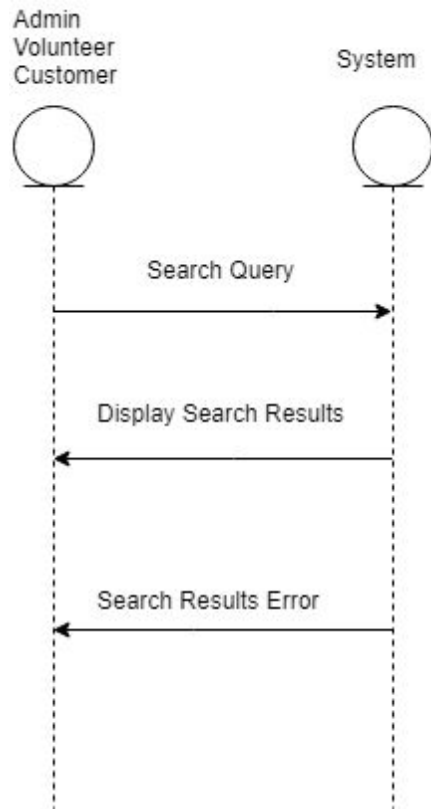ed is valid. This is of high priority as it ensures no fake or false accounts are created. A manual check of the potential account holder's OneCard is also needed when creating an account, ensuring the person in store signing up is the same person associated with V-Number being given. However, this is out of scope of the system and up to the staff of the FBFS to enforce. A customer with family members who also need the service of FBFS must declare their family information when registering as they will be given a different weekly limit. Weekly limits are of high priority to implement as accounts who abuse their respective limit will lose access to the FBFS. All accounts must also be able to be managed with up to date information as accounts become active or inactive and flagged for abusing the FBFS. Mitigating abuse of the service is another must have for the client and imperative for implementation; abuse can occur when customers take more than their weekly limit, falsely declare family status, create multiple accounts, or by using fake OneCards and V-Numbers.

3.2.2 Functional Requirements

SREQ-4 Accurate Tracking of Account Weekly Limit: The account limit will detail how much an account can withdraw from the FBFS during a weekly period. Accurate tracking of an account's limit must ensure that customers cannot abuse the FBFS by taking an excess amount of items. Customers must state if they are part of a family or not, however, the customers who do not qualify for the family limit must be barred from gaining access to the family limit. Customers caught abusing the FBFS will be flagged and their respective account terminated by admins or volunteers.

Rationale: Tracking of account limits is needed to ensure that customers know the amount of items they can take from the FBFS. It is also needed so customers cannot abuse the system by overtaking their allowed limit. This was requested by the clients in section 2 of their RFP [1].

SREQ-5 User Account Verification: Upon account creation, potential account holders must provide a OneCard and valid V-Number for the admin or volunteer to input to verify the account with. Accounts will be verified by their respective V-Number against the UVic database during account creation. Verification must be implemented in the system due to the customer abuse that can occur without it; abuse can arise when customers create multiple accounts with false OneCards or by utilizing a fake V-Number. The client's have specified in the RFP and client meetings that this functional requirement must be implemented due to the nature of abuse that can arise without it [1]. If the customer does not provide a OneCard and valid V-Number then the customer will not be allowed to create an account to use the FBFS services.

Rationale: Account verification is needed so customers do not abuse the system by signing up with false information in order to take more than their limit. The client's have specified in the RFP and client meetings that this functional requirement must be implemented due to the nature of abuse that can arise without it [1].

SREQ-6 Manage Accounts: Admins and volunteers will have the ability to manage customer accounts by entering the "View Accounts" section of the web application. Admins can also manage volunteer and other admin accounts under this section. Implementing this is a must because admins and volunteers need to be able to create, update and delete accounts so customers can use the system and be tracked for their usage, along with admins needing the ability to create, update and delete accounts for the staff of the FBFS. An account will fail to be created if all the needed information is not entered and an account will fail to be modified if the information entered is invalid.

Rationale: Being able to update an account's information is important because account holders can change information later that the system needs to track, for example the changing of a customer's family status. Being able to create an account is important so the staff of the FBFS can track the usage and abuse of the FBFS. Deleting an account is important because an account may need to be removed from the system due to inactivity or any other reason an admin or volunteer sees fit.

SREQ-7 Different Views: Depending on which account type you are logged in as (whether its admin, volunteer or customer) the web application will display different views. Each view will allow the user to only see what their user class is allowed to access. If the system displays the wrong view to the user, the user may be able to take advantage of the system and abuse the FBFS.

Rationale: The need for different views is to ensure that each user type can only see system features that they have access to. This is needed so a customer can not have access to deleting accounts or generating false reports.

3.2.3 Use Case(s) associated with the Feature or Functional Requirements

UC-1: Login

    Actors: Admin, Volunteer, Customer

    Preconditions: Using a web browser. User has an account

    Steps:

1) Click the 'Sign In' button on the top right of any page
2) User enters their username and password
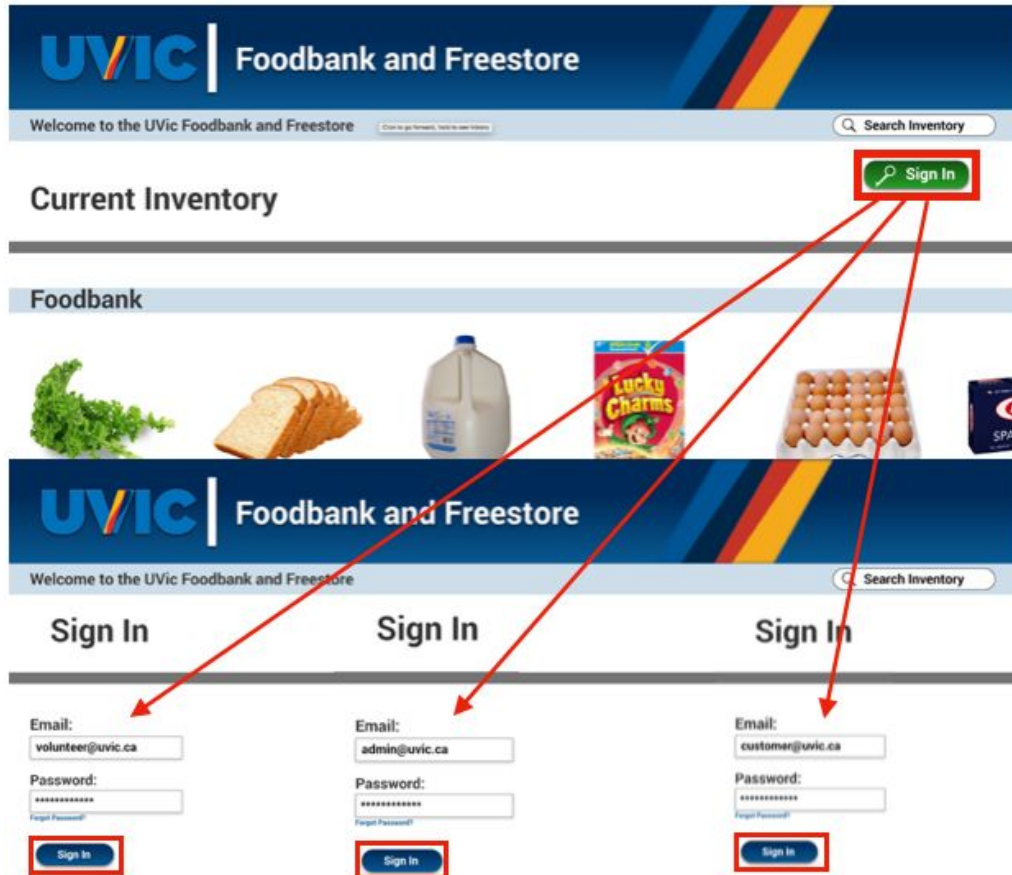3) User clicks the 'Sign In' button

    Success conditions: The user is now logged into their account

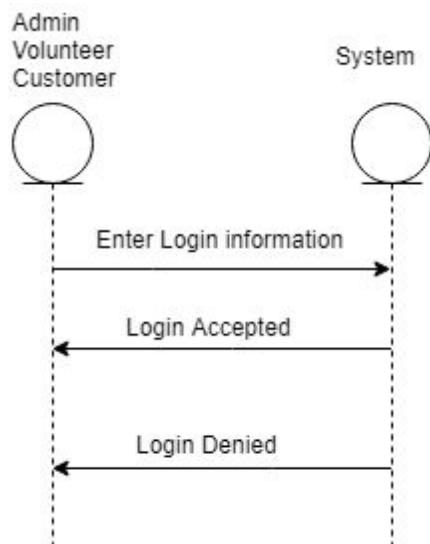   **Alternate paths:** User enters wrong username or password and is denied access. Repeat steps 1-3.

    Storyboard: Refer to Storyboard 1

    Sequence Diagram: Refer to Sequence Diagram 1

Storyboard 1: Login for each account type

Sequence Diagram 1

UC-2: Logout

Actors: Admin, Volunteer, Customer

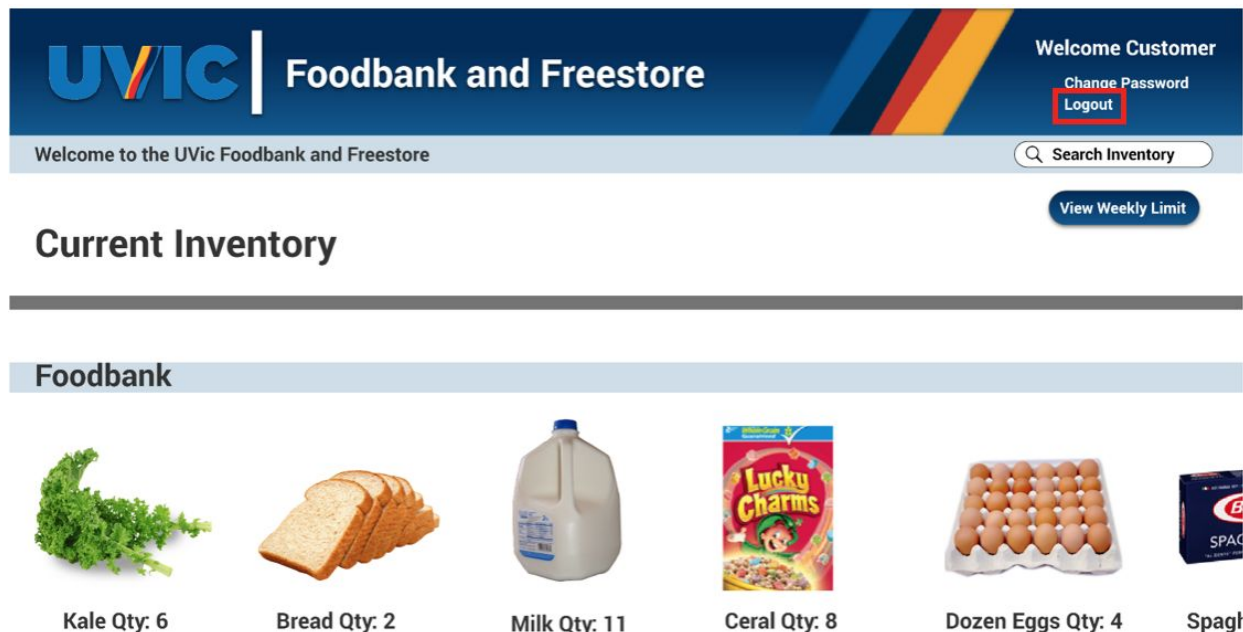**Preconditions:** Using a web browser. User must be logged in

Steps:

1) Click the 'Logout' button

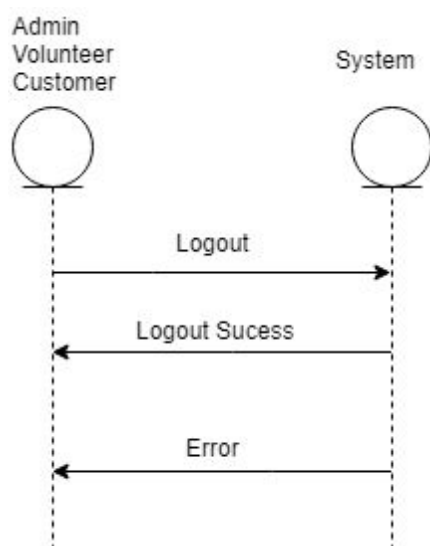**Success conditions:** The user is now logged out their account

Alternate paths: Page refreshes with an error print out and does not logout the user. Repeat step 1.

Storyboard: Refer to Storyboard 2

Sequence Diagram: Refer to Sequence Diagram 2

Storyboard 2: Logout for each account type

Sequence Diagram 2

UC-8: Create Customer Account:

Actors: Admin, Volunteer

Preconditions: Using a web browser. Logged in as Admin/Volunteer. Customer is in store requesting an account

Steps:
1) Select 'View Accounts'
2) Select 'Create Customer Account'
3) Fill in required customer information (First name, last name, password, V-Number, UVic email and demographic information)
4) Have customer agree to terms and conditions
5) Select 'Create'

Success Conditions: The newly created account will appear under the "View Accounts" tab and the page will change back to "View Accounts"

Alternate Paths: Missing information; account will not be created until all information is entered. Repeat steps 3-5.

Storyboard: Refer to Storyboard 8

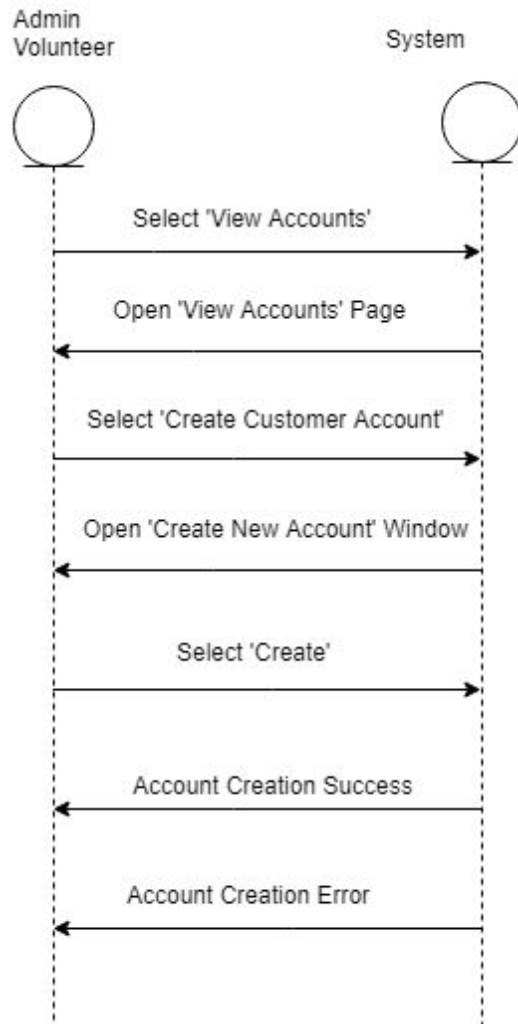**Sequence Diagram:** Refer to Sequence Diagram 8

Storyboard 8: Creating a new customer account

Sequence Diagram 8

UC-9: Update Customer Account information:
    Actors: Admin, Volunteer
    Preconditions: Using a web browser. Logged in as Admin/Volunteer
    Steps:
        1) Select 'View Accounts'
        2) Find or search for the customer whose information you would like to edit
        3) Select the account from the list by either double clicking or selecting 'Select Account'
        4) Select 'Update Account Information'
        5) Modify the information as needed
        6) Select 'Confirm Changes'
        7) Review that the changes have taken place
        8) Select 'Confirm Changes'
    Success Conditions: Account information displayed is the new updated information

Alternate Paths: Changes not updated. Repeat steps 5-8.
Storyboard: Refer to Storyboard 9
Sequence Diagram: Refer to Sequence Diagram 9
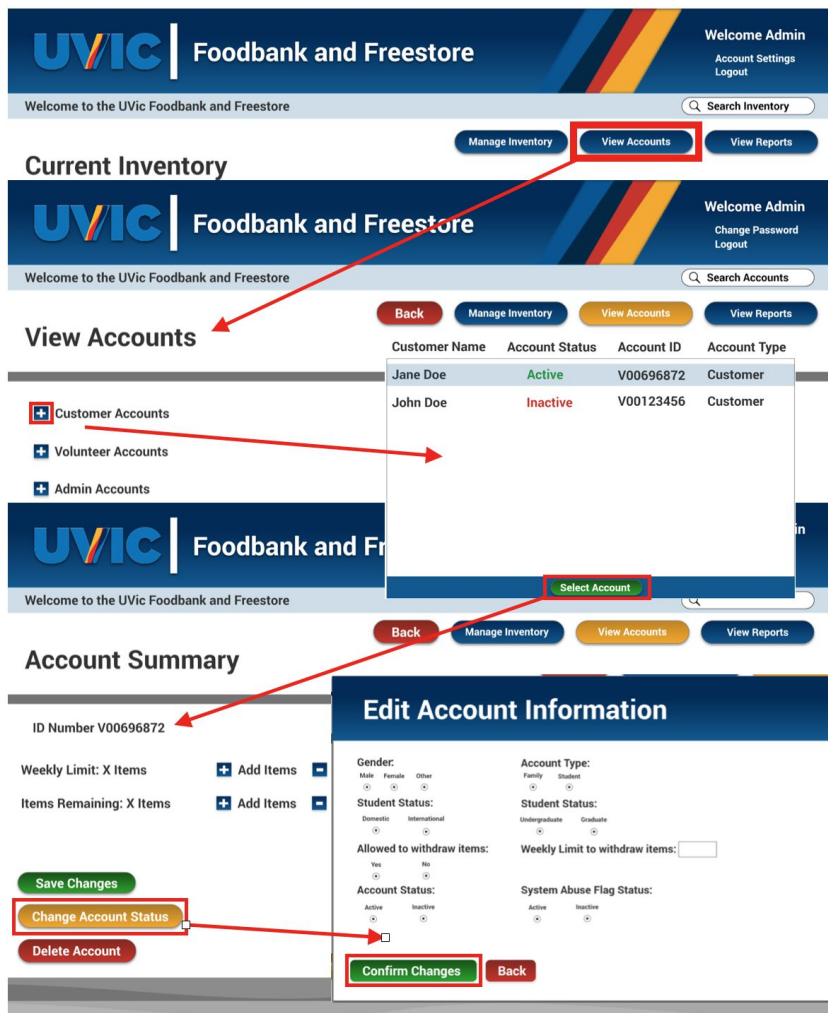
Storyboard 9: Editing an account's information
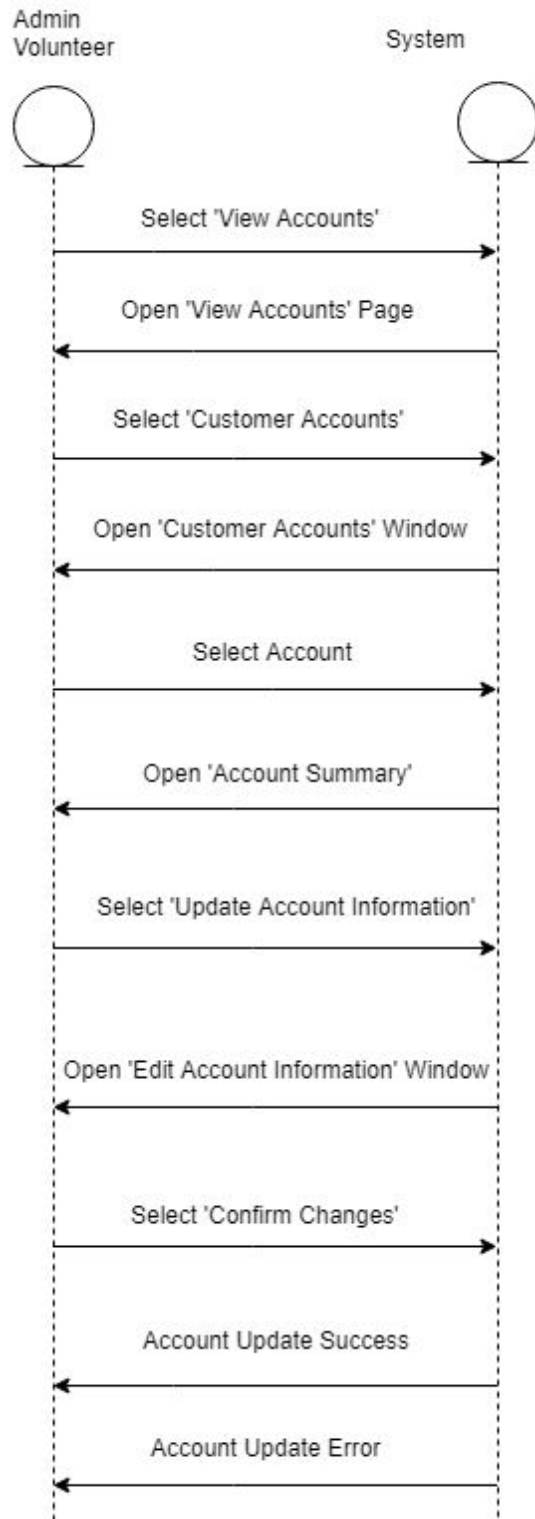
Sequence Diagram 9

UC-10: Delete Customer Account:

      Actors: Admin, Volunteer

      Preconditions: Using a web browser. Logged in as Admin/Volunteer

      Steps:

          1) Go to 'View Accounts'

          2) Find or search for the customer whose account needs to be removed

          3) Select the account from the list by either double clicking or selecting 'Select Account'

          4) Select 'Delete Account'

          5) Verify Delete

      Success Conditions: The customer account should no longer appear under the "View Accounts" page.

      Alternate Paths: Customer account is not deleted. Repeat steps 2-5.

      Storyboard: Refer to Storyboard 10

      Sequence Diagram: Refer to Sequence Diagram 10

Storyboard 10: Deleting an account

Admin
Volunteer   System

Select 'View Accounts'

Open 'View Accounts' Page

Select 'Customer Accounts'

Open 'Customer Accounts' Window

Select Account

Open 'Account Summary'

Select 'Delete Account'

Open 'Delete Account Confirmation' Window

Select 'Yes'

Account Delete Success

Account Delete Error

Sequence Diagram 10

UC-11: View Customer Account

  Actors: Admin, Volunteer

  Preconditions: Using a web browser. Logged in as Admin/Volunteer

  Steps:

  1) Select 'View Accounts'

  2) Select or search customer account to view

  3) View customer account

  Success Conditions: Customer account viewed

  Alternate Paths: Screen displays error message; Customer account not found. Repeat steps 1-3.
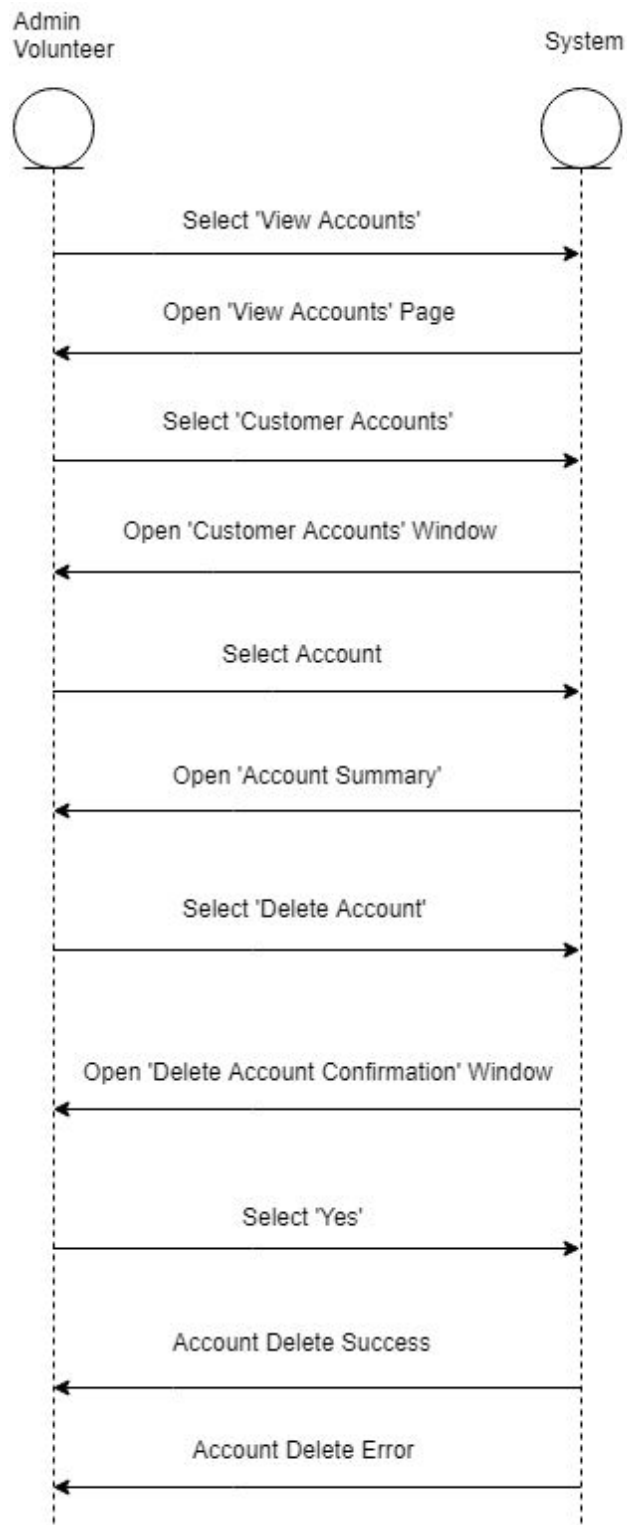
  Storyboard: Refer to Storyboard 11

  Sequence Diagram: Refer to Sequence Diagram 11

Storyboard 11: Viewing an account

Sequence Diagram 11

UC-12: View Volunteer Account

    Actors: Admin

    Preconditions: Using a web browser. Logged in as Admin

    Steps:

        1) Select 'View Accounts'

        2) Select or search volunteer account to view

        3) View volunteer account

    Success Conditions: Volunteer account viewed

    Alternate Paths: Screen displays error message; Volunteer account not found. Repeat steps 1-3.

    Storyboard: Refer to Storyboard 11, but click Volunteer instead of Customer

    Sequence Diagram: Refer to Sequence Diagram 12

Sequence Diagram 12

UC-13: View Admin Account

      Actors: Admin

      Preconditions: Using a web browser. Logged in as Admin
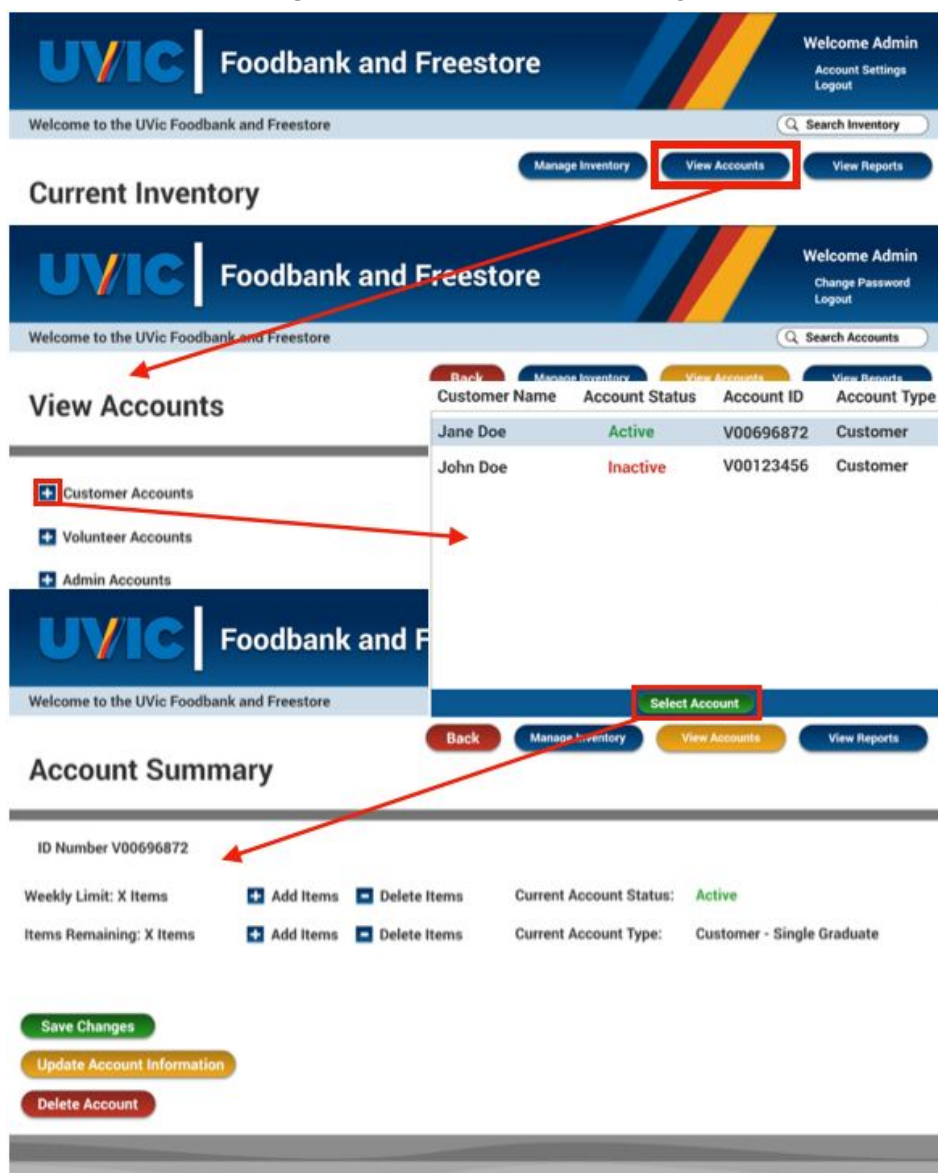
      Steps:

            1) Select 'View Accounts'

            2) Select or search admin account to view

            3) View admin account

      Success Conditions: Admin account viewed

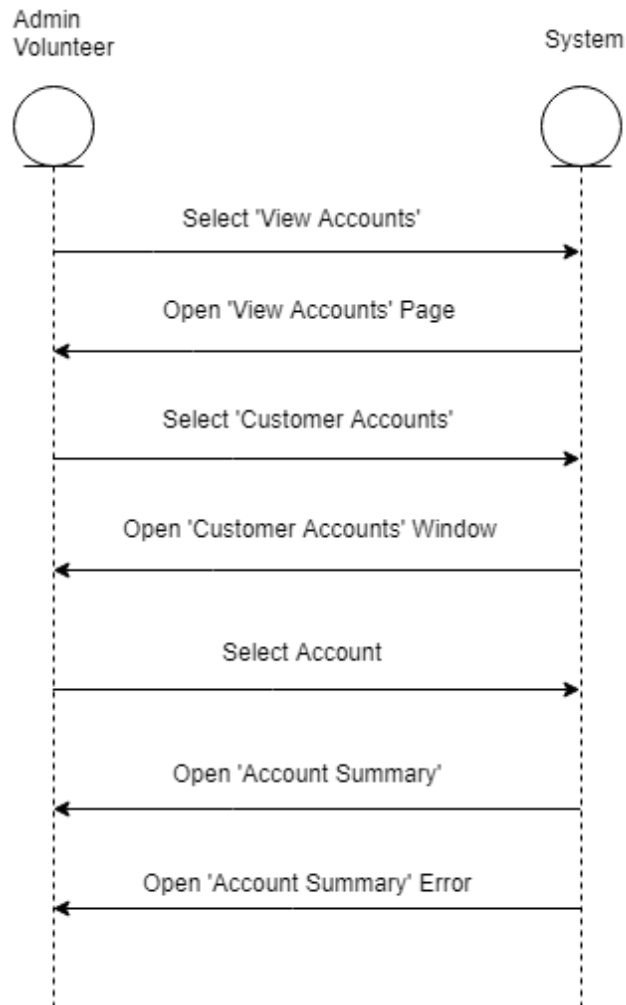   **Alternate Paths:** Screen displays error message; Admin account not found. Repeat steps 1-3.

      Storyboard: Refer to Storyboard 11, but click Admin instead of Customer

      Sequence Diagram: Refer to Sequence Diagram 13

Sequence Diagram 13

UC-14: Create Volunteer Account

      Actors: Admin

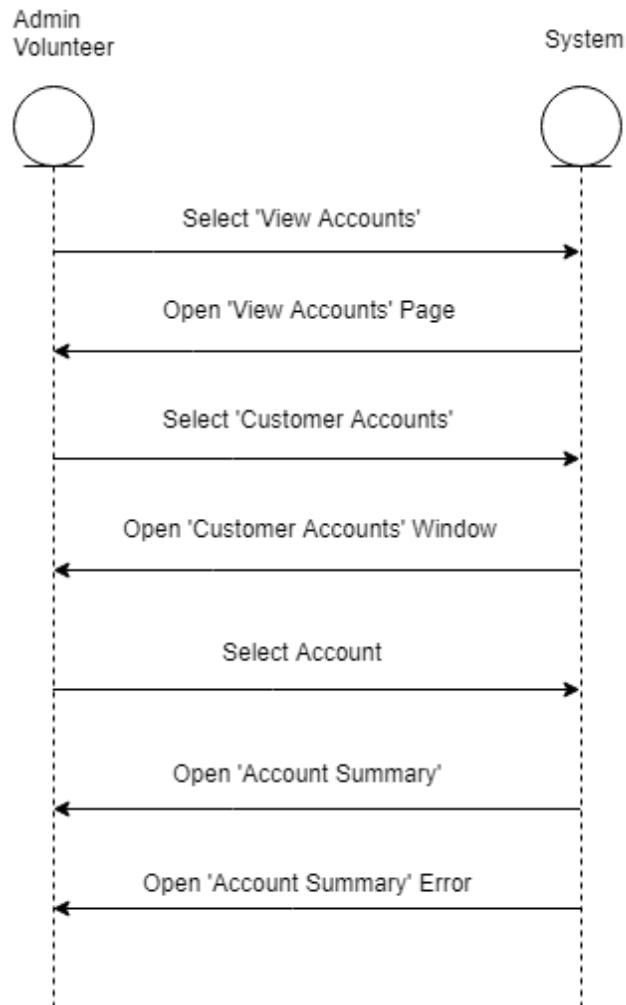   **Preconditions:** Using a web browser. Logged in as Admin, Volunteer is present

      Steps:

            1) Select 'View Accounts'

            2) Select 'Create Volunteer Account'

            3) Fill in volunteer information (First name, last name, password, V-Number, UVic email and demographic information)

            4) Have volunteer agree to terms and conditions, and click "Terms and Conditions" button

            5) Select 'Create'

      Success Conditions: The newly created account will appear under the 'View Accounts' tab

      Alternate Paths: Missing information, account will not be created until all information is entered. Repeat steps 3-5.

Storyboard: Refer to Storyboard 12
Sequence Diagram: Refer to Sequence Diagram 14

Storyboard 12: Creating a volunteer account

Sequence Diagram 14

UC-15: Create Admin Account
    Actors: An already Admin and new (to be) Admin
    Preconditions: Using a web browser. Logged in as Admin, new Admin is present
    Steps:
        1) Select 'View Accounts'
        2) Select 'Create Admin Account'
        3) Fill in admin information (First name, last name, password, V-Number, UVic email)
        4) Have new admin agree to terms and conditions, and click 'Terms and Conditions' button
        5) Select 'Create'
    Success Conditions: The newly created account will appear under the 'View Accounts' tab.
    Alternate Path: Missing information, account will not be created until all information is entered. Repeat steps 1-5.

Storyboard: Refer to Storyboard 13
Sequence Diagram: Refer to Sequence Diagram 15
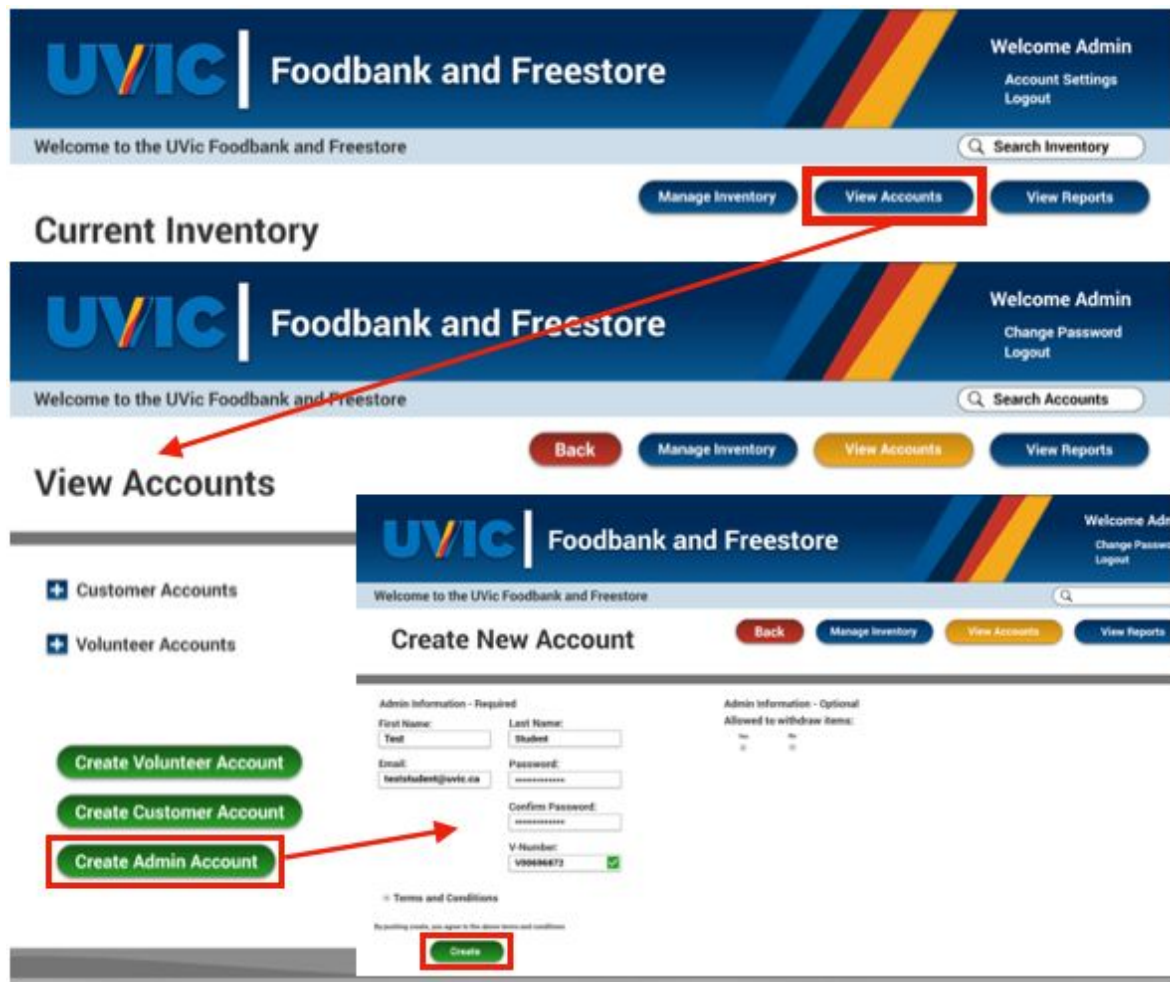
Storyboard 13: Creating an admin account
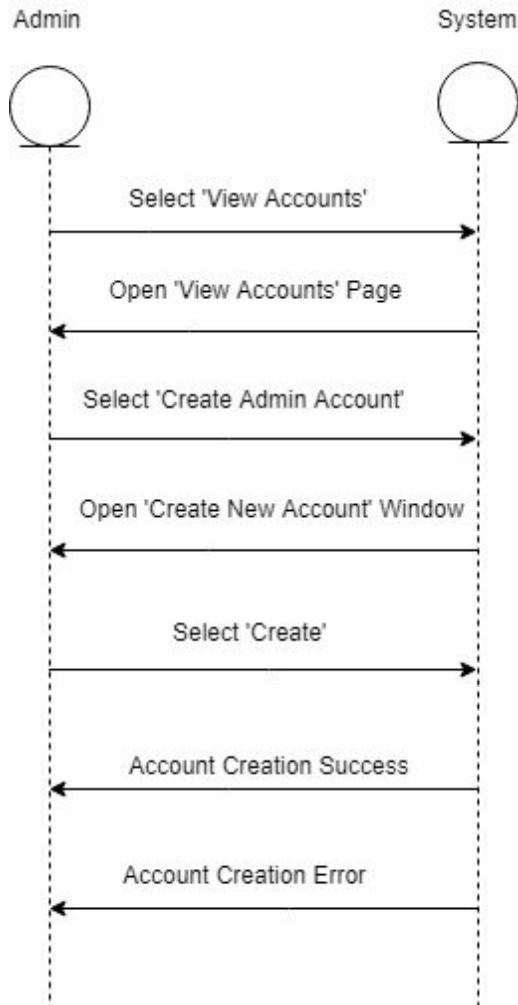
Sequence Diagram 15

UC-16: Edit Volunteer account:

    Actors: Admin

    Preconditions: Using a web browser. Logged in as Admin

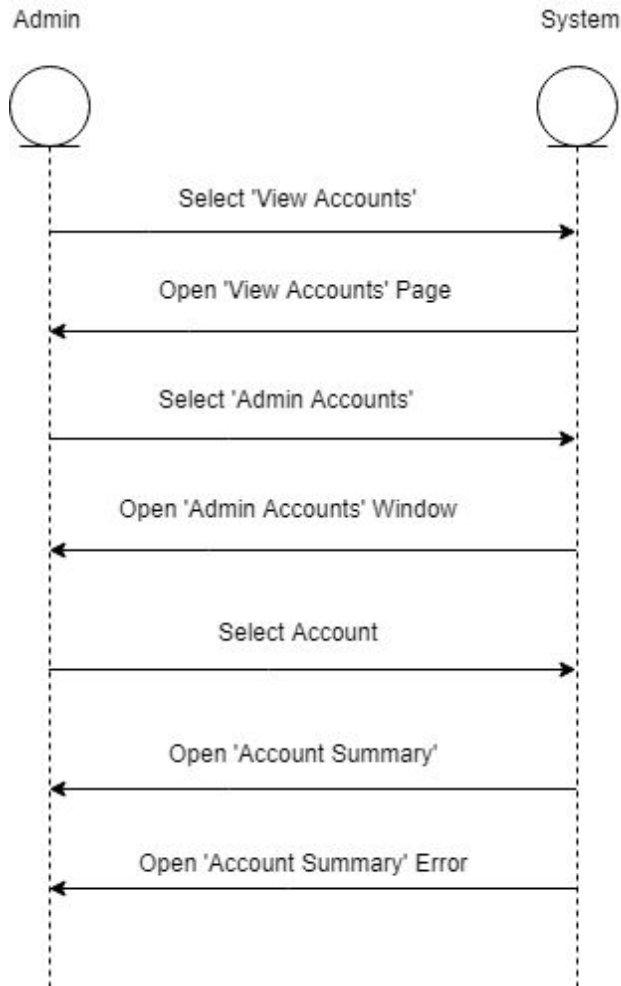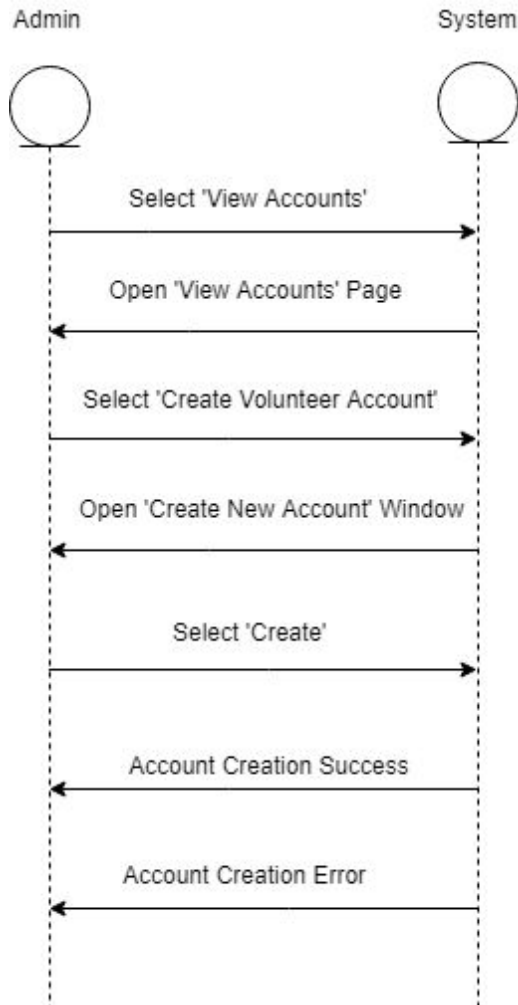   **Steps:**

        1) Select 'View Accounts'

        2) Search or select a volunteer account

        3) Select 'Select Account'

        4) Select 'Update Account Information'

        5) Modify the information as needed

        6) Select 'Confirm Changes'

        7) Review that the changes have taken place

        8) Select 'Save Changes'

    Success Conditions: Volunteer account will have the latest information.

    Alternate Paths: Changes not updated. Repeat steps 5-8.

    Storyboard: Refer to Storyboard 9, but click Volunteer instead of customer

Sequence Diagram: Refer to Sequence Diagram 16

Admin   System

Select 'View Accounts'

Open 'View Accounts' Page

Select 'Volunteer Accounts'

Open 'Volunteer Accounts' Window

Select Account

Open 'Account Summary'

Select 'Update Account Information'

Open 'Edit Account Information' Window

Select 'Confirm Changes'

Account Update Success

Account Update Error
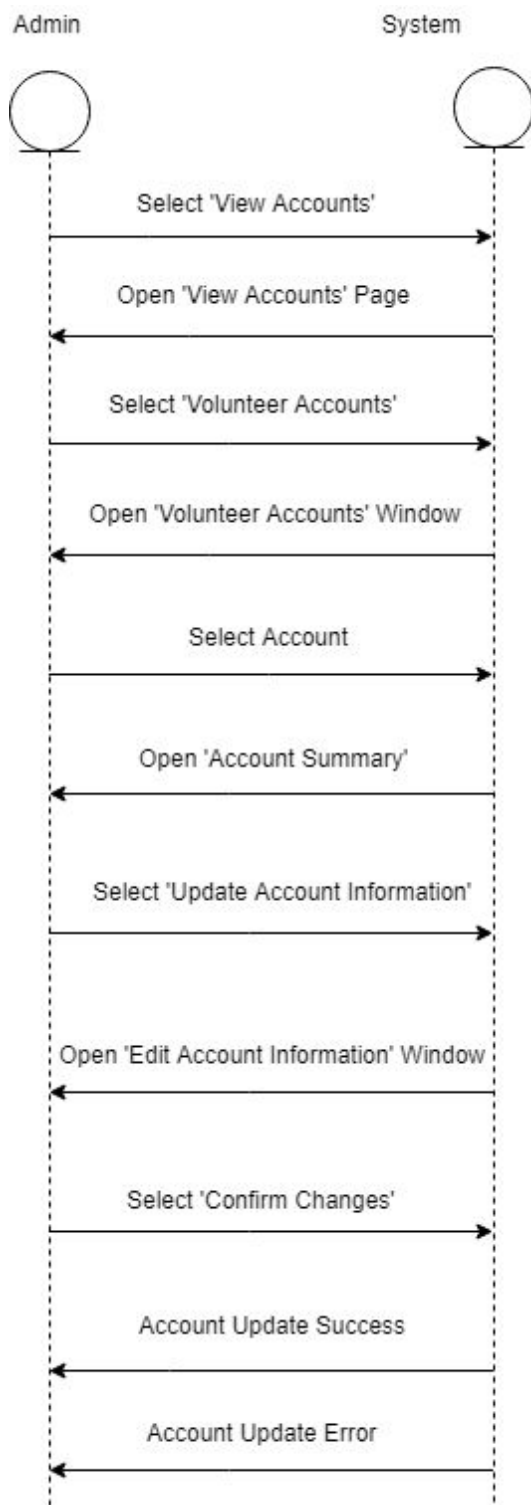
Sequence Diagram 16

UC-17: Edit Admin Account:

Actors: Admin

Preconditions: Using a web browser. Logged in as Admin

Steps:

1) Select 'View Accounts'
2) Search or select a Admin account
3) Select 'Select Account'
4) Select 'Update Account Information'
5) Modify the information as needed
6) Select 'Confirm Changes'
7) Review that the changes have taken place
8) Select 'Save Changes'

Success Conditions: Admin account will have the latest information.

Alternate Paths: Changes not updated. Repeat steps 5-8.

Storyboard: Refer to Storyboard 9, but click Admin instead of customer

Sequence Diagram: Refer to Sequence Diagram 17
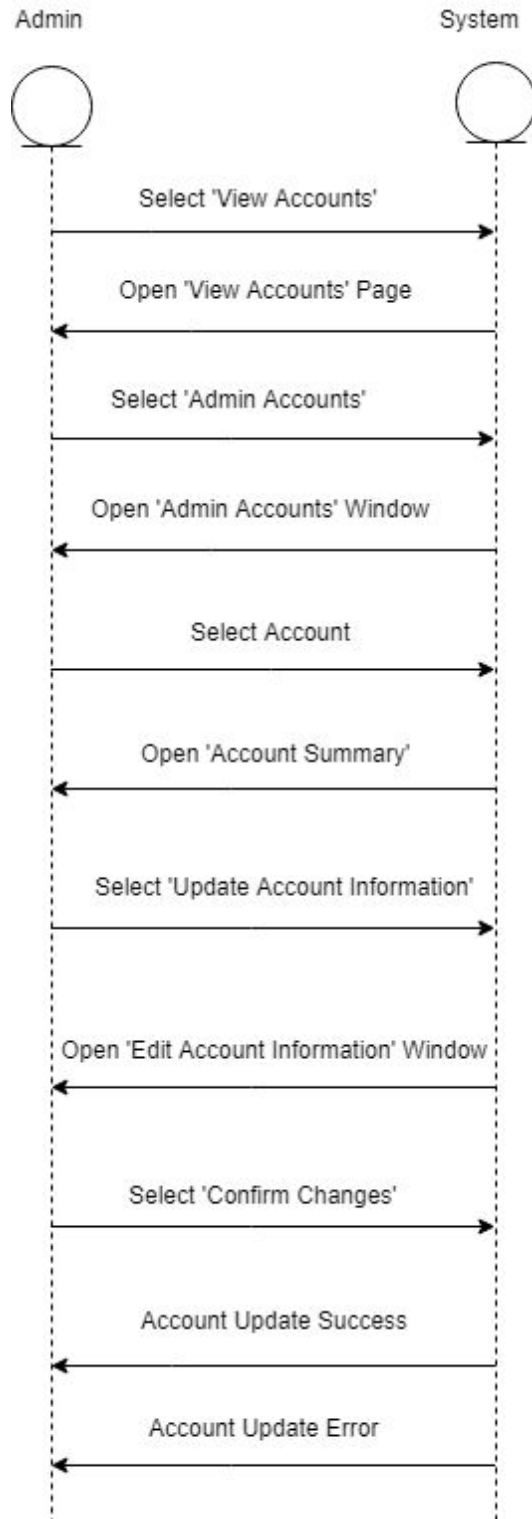
Sequence Diagram 17

UC-18: Delete Volunteer Account:

    Actors: Admin

    Preconditions: Using a web browser. Logged in as Admin

    Steps:

        1) Select 'View Accounts'

        2) Search or select a volunteer account

        3) Select 'Select Account'

        4) Select 'Delete Account'

        5) Verify Delete

    Success Conditions: Volunteer account will be removed from the system.

    Alternate Paths: Volunteer account not deleted. Repeat steps 2-5.

    Storyboard: Refer to Storyboard 10, but click Volunteer instead of customer

    Sequence Diagram: Refer to Sequence Diagram 18

Admin   System

Select 'View Accounts'

Open 'View Accounts' Page

Select 'Volunteer Accounts'

Open 'Volunteer Accounts' Window

Select Account

Open 'Account Summary'

Select 'Delete Account'

Open 'Delete Account Confirmation' Window

Select 'Yes'

Account Delete Success

Account Delete Error

Sequence Diagram 18

UC-19: Delete Admin Account:

Actors: Admin

Preconditions: Using a web browser. Logged in as Admin

Steps:

1) Go to 'View Accounts' screen
2) Search or select an Admin account
3) Select 'Select Account'
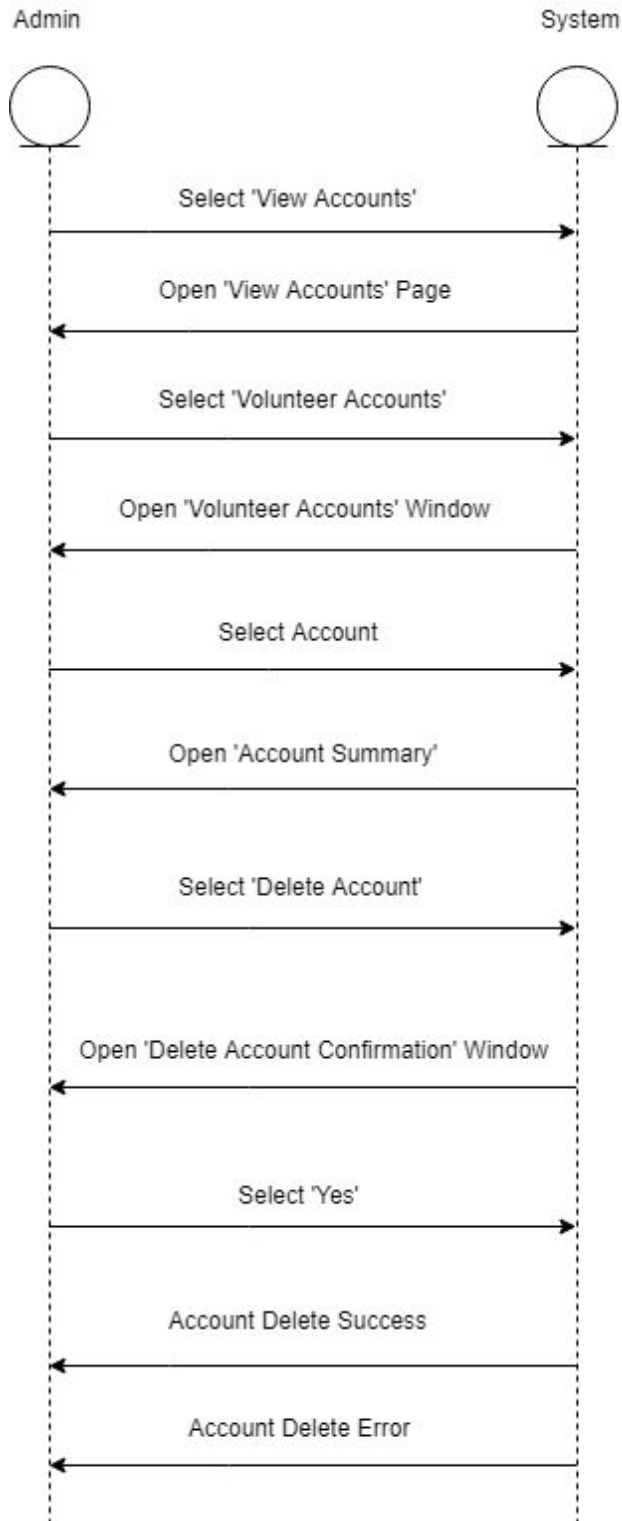4) Select 'Delete Account'
5) Verify Delete

Success Conditions: Admin account will be removed from the system.

Alternate Paths: Admin account not deleted. Repeat steps 1-5.

Storyboard: Refer to Storyboard 10, but click Admin instead of customer

Sequence Diagram: Refer to Sequence Diagram 19

```
Admin                                    System

  ○                                        ○
  │                                        │
  │          Select 'View Accounts'        │
  │───────────────────────────────────────▶│
  │                                        │
  │        Open 'View Accounts' Page       │
  │◀───────────────────────────────────────│
  │                                        │
  │          Select 'Admin Accounts'       │
  │───────────────────────────────────────▶│
  │                                        │
  │       Open 'Admin Accounts' Window     │
  │◀───────────────────────────────────────│
  │                                        │
  │            Select Account              │
  │───────────────────────────────────────▶│
  │                                        │
  │         Open 'Account Summary'         │
  │◀───────────────────────────────────────│
  │                                        │
  │          Select 'Delete Account'       │
  │───────────────────────────────────────▶│
  │                                        │
  │  Open 'Delete Account Confirmation' Window │
  │◀───────────────────────────────────────│
  │                                        │
  │              Select 'Yes'              │
  │───────────────────────────────────────▶│
  │                                        │
  │         Account Delete Success         │
  │◀───────────────────────────────────────│
  │                                        │
  │          Account Delete Error          │
  │◀───────────────────────────────────────│
  │                                        │
```
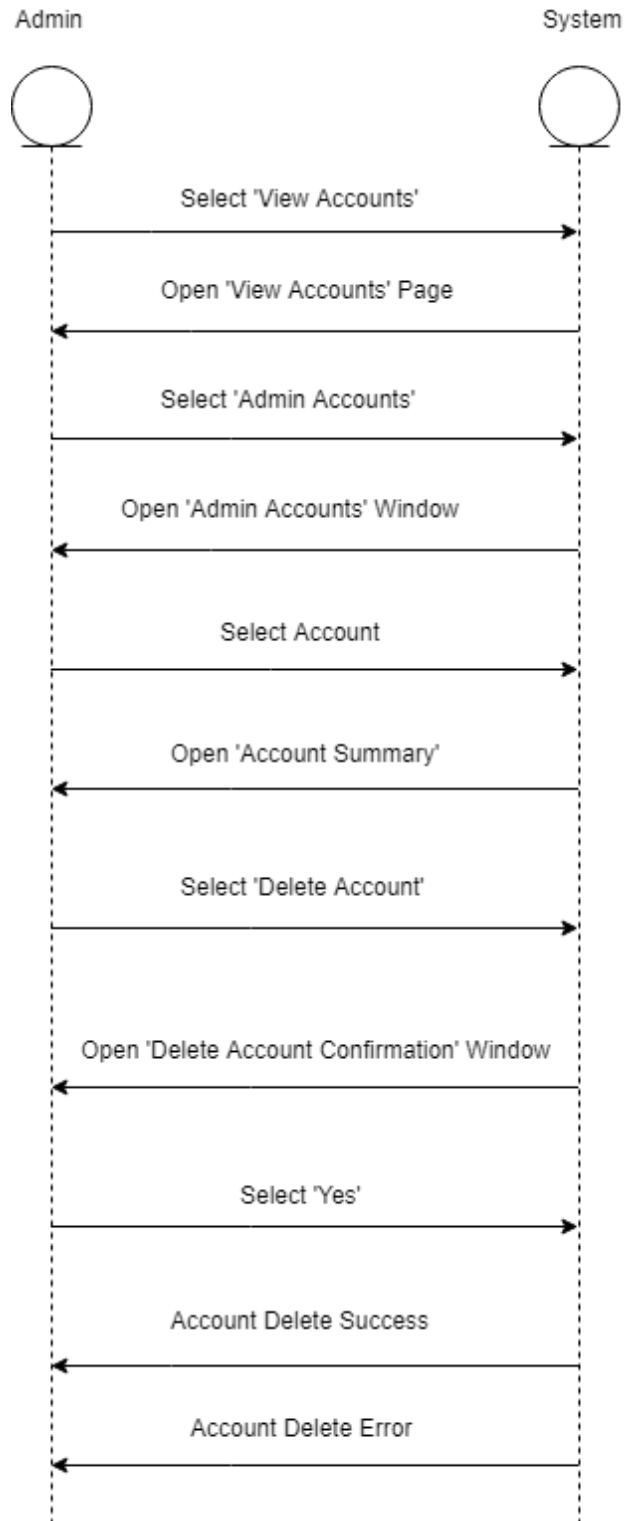
Sequence Diagram 19

UC-20: View Weekly Limit:

Actors: Customer

Preconditions: Using a web browser. Logged in as Customer

Steps:

1) Go to 'View Weekly Limit' screen

Success Conditions: Customer views their weekly limit.

Alternate Paths: 404 Page Not Found, or any 500 level http error. Refresh the page and repeat step 1.

Storyboard: Refer to Storyboard 14

Sequence Diagram: Refer to Sequence Diagram 20

Storyboard 14: Viewing your weekly limit as a customer
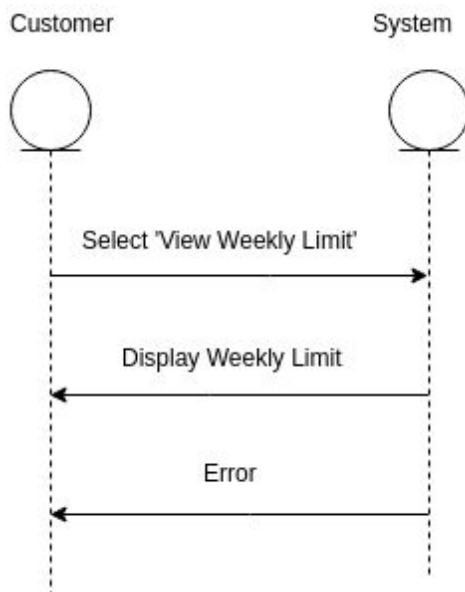
Sequence Diagram 20

UC-22: Update Weekly Limit

      Actors: Admin, Volunteer

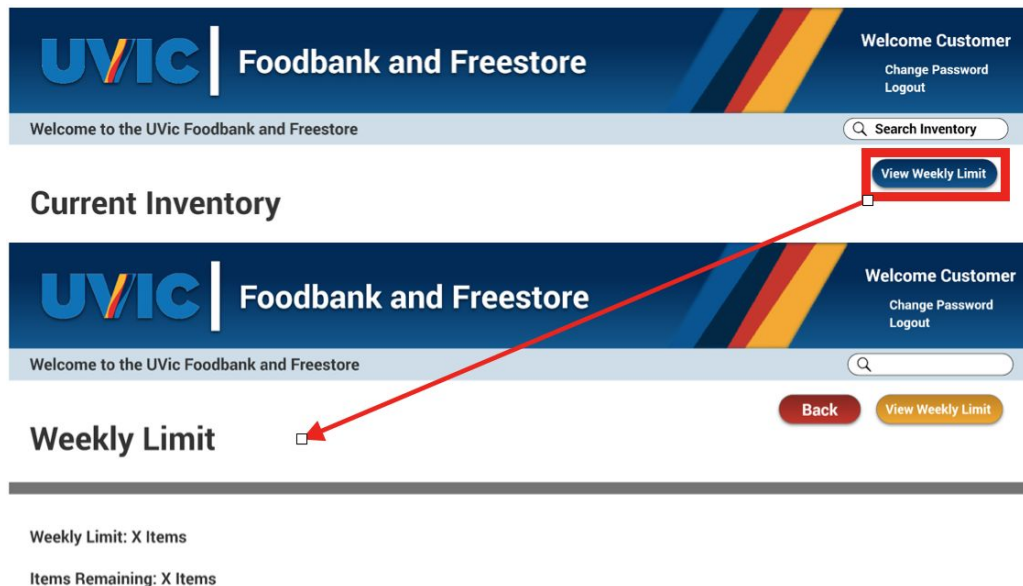   **Preconditions:** Using a web browser. Logged in as Admin/Volunteer

      Steps:

            1) Select 'View Accounts'

            2) Expand account type of choice

            3) Find desired account and click 'Select Account'

            4) Edit weekly limit as desired

      Success Conditions: User account weekly limit is update and accurately tracks how much is left

      Alternate Paths: Weekly limit has already been reached; transaction aborted.

      Storyboard: Refer to Storyboard 15

      Sequence Diagram: Refer to Sequence Diagram 22

Storyboard 15: Update weekly limit
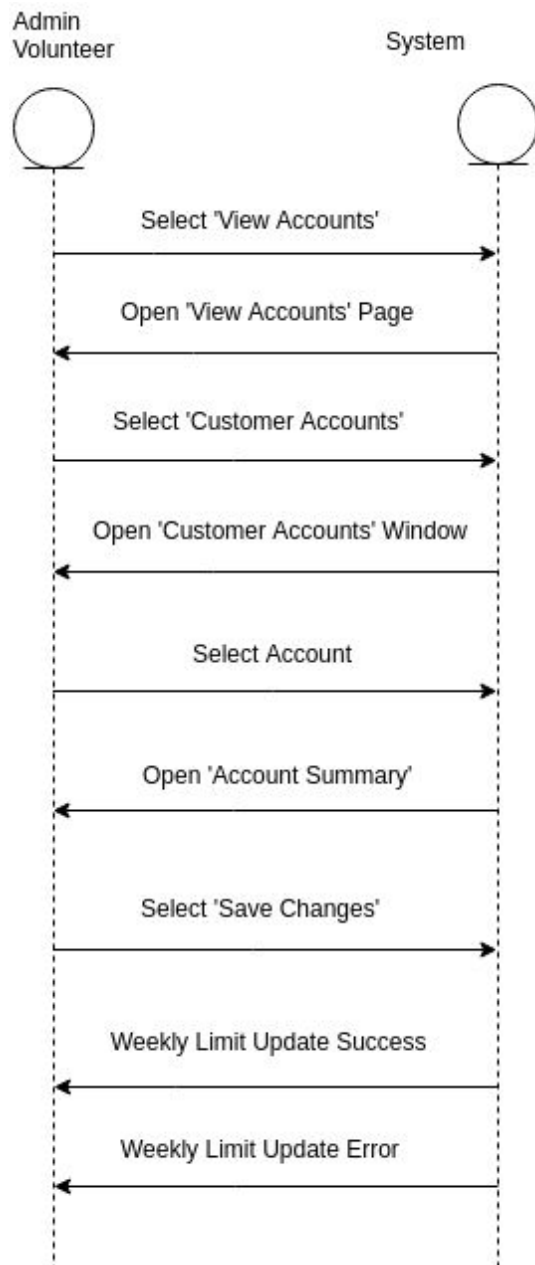
Sequence Diagram 22

UC-23: Change Password:

      Actors: Customer, Volunteer, Admin

      Preconditions: Using a web browser. Logged in as Customer/Volunteer/Admin

      Steps:

            1) Select "Change Password"

            2) Fill out new desired password

            3) Select 'Update Password'

Success Conditions: Taken back to the main inventory page
Alternate Paths: 404 Page Not Found, or any 500 level http error. Repeat steps 1-3.
Storyboard: Refer to Storyboard 16
Sequence Diagram: Refer to Sequence Diagram 23

Storyboard 16: Changing your password
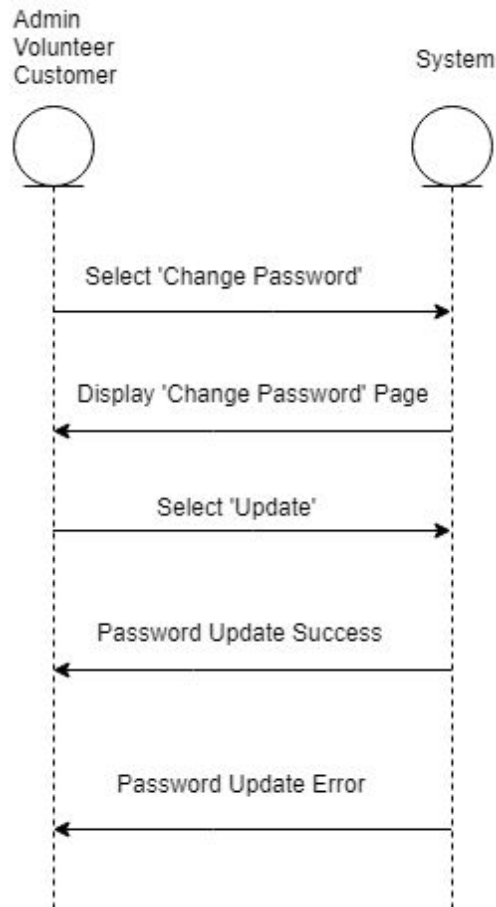
Sequence Diagram 23

UC-24: Flag Account

    Actors: Admin, Volunteer

    Preconditions: Using a web browser. Logged in as Admin/Volunteer

    Steps:

        1) User searches for the Customer/Volunteer account in question

        2) User clicks "View Account" once the account in question is found

        3) User clicks "Edit Account"

        4) User toggles "System Abuse Flag Status" to Active

        5) User then pushes "Confirm Changes"

    Success conditions: The account in question is now marked with the flagged system abuse status.

    Alternate paths: Account fails to receive the status and error is displayed. User repeats steps 1-5.

    Storyboard: Refer to Storyboard 9

    Sequence Diagram: Refer to Sequence Diagram 24

Admin
Volunteer

System

Select 'View Accounts'

Open 'View Accounts' Page

Select 'Customer Accounts'

Open 'Customer Accounts' Window

Select Account

Open 'Account Summary'

Toggles 'System Abuse Flag Status'
Select 'Save Changes'

Account Status Success

Account Status Error
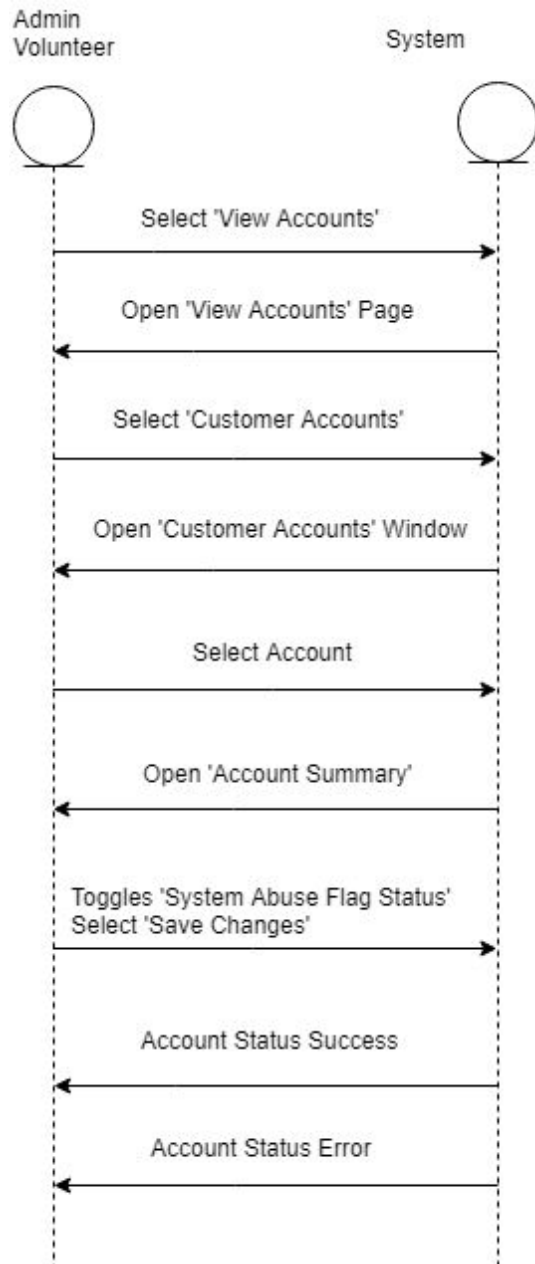
Sequence Diagram 24

3.3 Statistics Gathering

3.3.1 Description and Priority

The system will be able to gather and keep track of user data to find where inventory comes from and where it goes as well as customer demographics. The FBFS has data collection and tracking as a medium priority, due to the admins making use of the statistics for inventory assistance and misuse prevention.

3.3.2 Functional Requirements

SREQ-8 View Statistic Reports: Only admins have the authority to view statistical reports and filter statistical reports.

Rationale: Access authority is needed to ensure the privacy of customers and volunteers. This was requested by the clients in the second design meeting.

SREQ-9 Generate Statistics Report: Admins must be able to generate a statistics report based on filters on the data records. These filters allow for a more specific statistics report on the demographic that the admin wants. Admins will be able to filter the reports by gender, level of education, student status and family status, along with new and old inventory levels. Generated reports will be displayed in the browser for the admin to view. This requirement was requested by the client's in section 2 of their RFP [1].

Rationale: Admins must be able to generate statistics about the customers for information, inventory assistance, misuse prevention and business purposes. This requirement was requested by the client's in section 2 of their RFP [1].

3.3.3 Use Case(s) associated with the Feature or Functional Requirements

UC-21: Generate Statistic Reports
    Actors: Admin
    Preconditions: Using a web browser. Logged in as Admin; FBFS has statistical data
    Steps:
        1) Select 'View Reports'
        2) Select 'Generate New Report'
        3) Enter a name for the report
        4) Check the boxes that apply
        5) Select 'Generate Report'
    Success Conditions: Report generated appears in list of saved reports
    Alternate Paths: No data, no filters selected, or wrong filters (reset) cause an error print out, page reset. Repeat steps 2-5.
    Storyboard: Refer to Storyboard 16
    Sequence Diagram: Refer to Sequence Diagram 21
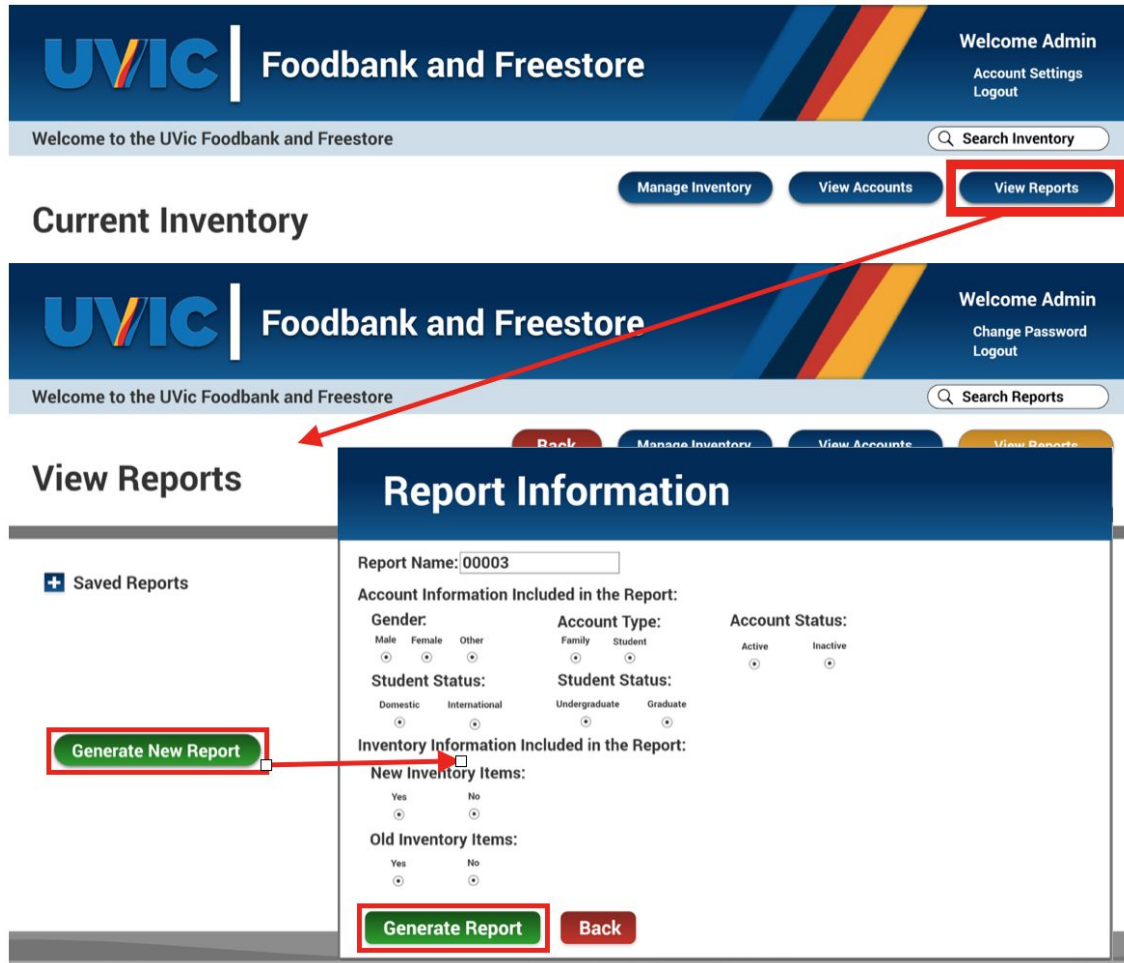
Storyboard 16: Generating a new statistical report
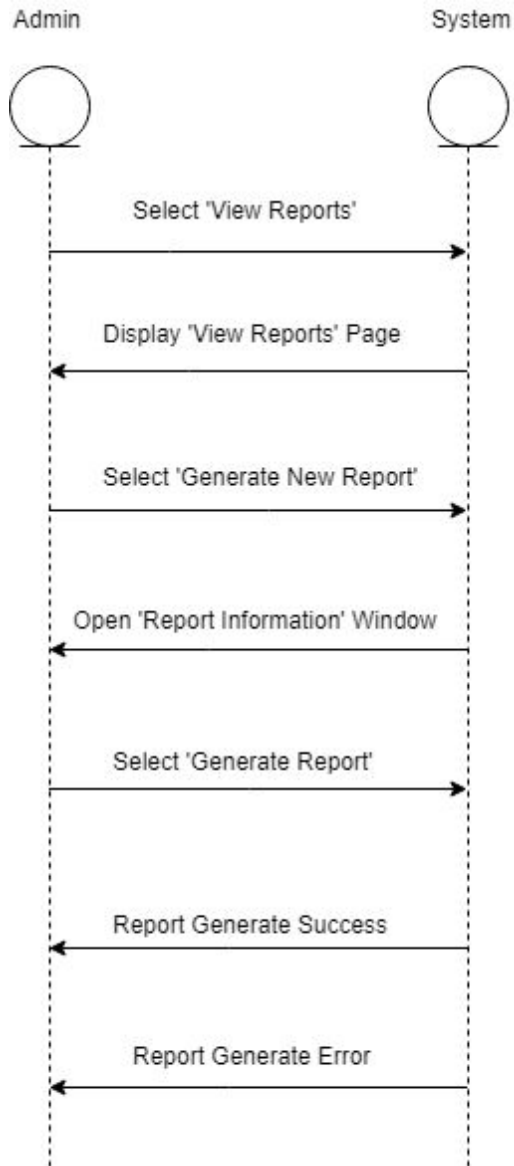
Sequence Diagram 21

4 External Interface Requirements

4.1 User Interfaces

IREQ-1 Search Bar: Description: Users must be able search for an item or multiple items in the search bar, as seen is UC-7.

Rationale: The client wants users to be able to find inventory without having to go through the whole catalog. This requirement originated from a meeting with the client.

4.2 Hardware Interfaces

IREQ-2 Device Compatibility: The system must be accessible on all personal computers, mobile devices and laptops with a recent version (released after 2007) of Windows, MacOS, Linux, Android, or IOS.

Rationale: Users will be attempting to access the web application from a variety of devices. The client has identified functionality on operating systems and devices is crucial. This requirement originated from an email with the clients.

4.3 Software Interfaces

IREQ-3 Data Storage: All data must be stored on the UVic database and adhere to their storage policies [2].

Rationale: The data that is being collected is confidential, and can only legally be stored in UVic database. This requirement originated from section 6 of the client's RFP [1].

4.4 Communications Interfaces

IREQ-4 Browser Compatibility: The system must be accessible via the Internet and will be optimized for Internet Explorer (Version 11.0.85 and beyond) and Google Chrome (Version 69.0.3497.92 and beyond).

Rationale: Users will be attempting to access the web application from Google Chrome and Internet explorer. Functionality of the web application with these browsers is crucial. This requirement originated from an email with the clients.

5 Other Non Functional Requirements

5.1 Performance Requirements

NFR-1 Response Time: The system must respond to queries within three seconds of making the request on the system.

Rationale: The client requested in an email that the system must respond to queries within three seconds.

NFR-2 Inventory Update Time: When a Volunteer or Admin modifies the inventory, they must have their changes viewable within one minute.

Rationale: Our client wants changes visible right away. A one minute cap works for that without stressing the system for extremely fast changes.

NFR-4 UVic Database Capacity: The FBFS user database must support up to 1500 total customer accounts.

Rationale: The client said that there are around 120 new users a year, and accounts are removed a year after they leave UVic. For a typical four year degree that is 600 users, so we doubled that plus 300 to give a good upper bound on the total amount of accounts stored in the UVic database at one time.

NFR-4.1 Inventory Database Capacity: The FBFS inventory database must support up to 10000 unique items.

Rationale: There are a lot of food items, and other items, but not more than 10000, meaning 10000 is a safe upper bound.

NFR-5 Account Removal: A customer account must be removed from storage one year after the user leaves UVic. An admin or volunteer account will only be removed manually.

Rationale: The client specified by email that a customer account will be removed one year after the individual leaves UVic.

NFR-6 Time or Space Bound: One hundred users are expected to be able to use the remote system at any given time with everything functioning as per the requirements in this document.

Rationale: The client specified by email that 10 users generally use the system at any given time, so 100 was chosen as a safe upper bound.

NFR-7 Downtime: The system must have less than 10% downtime per month, where the system does not work as per the requirements in this document or the web application is offline.

Rationale: The system needs to be reliable, and work properly most of the time to avoid people not getting access to the service.

NFR-8 Usability: Any user should be able to reach all parts of the system within five page changes from the home page, excluding log in.

Rationale: The client requested that everything should be easy to use and quick to access.

5.2 Safety Requirements

NFR-9 Access Control: Customer data, statistical data, reports, and any other personal data must be accessible only to volunteers and admins as per UVic Protection of Privacy Policy [2].

Rationale: Requested by the client in the RFP, personal data in a UVic service needs to abide by the UVic Privacy Policy in who can access data.

NFR-10 Data Release: User data must not be released to unaffiliated parties. Customer statistical data must not be released either.

Rationale: By the privacy policy [2], data must be kept confidential.

5.3 Security Requirements

NFR-11 Privacy Policy: All user data must be on a UVic server and must follow UVic policies for protection of private information as per the UVic Protection of Privacy Policy [2].

Rationale: From the first design meeting, the client requested that all data must follow the UVic privacy policy [2] to abide by laws for UVic services.

NFR-12 ID Validation: Customers must have a OneCard and valid V-Number verified by the UVic database.

Rationale: False/additional V-Numbers must be prevented in order to eliminate the abuse found in the current system.

NFR-12.1 ID Verification: During account creation, a volunteer or admin must cross check the V-Number and name on a customer's OneCard with the customer's associated account to confirm the identity of the customer.

Rationale: Customers with fake IDs are considered to misuse the system and must be prevented from creating an account.

It should be noted that statistical data runs on an honor based system. As a result, verification of this information runs outside the scope of this project. Check glossary for 'statistical data' to infer what data is based on the honor system.

5.4 Software Quality Attributes

NFR-13 Inventory Management Time: Volunteers and Admins must be able to manage inventory (UC-3, UC-4, or UC-5) within one minute but preferably within seconds.

Rationale: Managing inventory per item should not take a lot of time. Therefore a good upper bound is one minute to make sure it is done efficiently.

NFR-14 Account Creation Time: Any admin or volunteer should be able to create and manage a customer account within five minutes maximum.

Rationale: The system should be easy and quick to use.

NFR-15 Training Time: Any volunteer or admin should be able to learn to create and manage a customer account within ten minutes maximum.

Rationale: The system should be easy and quick to use.

NFR-16 Readability: Customers must be able to read the inventory with ease and have no trouble navigating/using the system.

Rationale: The systems should be easy to use.

6 Analysis Models

6.1 Use Case Model

The Use Case Diagram below (fig. 1) shows the actors and behaviours of the system. The actors include the admins, volunteers and customers. Each actor has a set of actions that they can accomplish with the new system. Everyone can view inventory, search for an item, login, logout, and change password. Admins must be able to manage accounts of both volunteers and customers. Volunteers must be able to manage the customer accounts. The diagram below shows the actors and their use cases with respective line colours for clarification.

Figure 1: Use case model
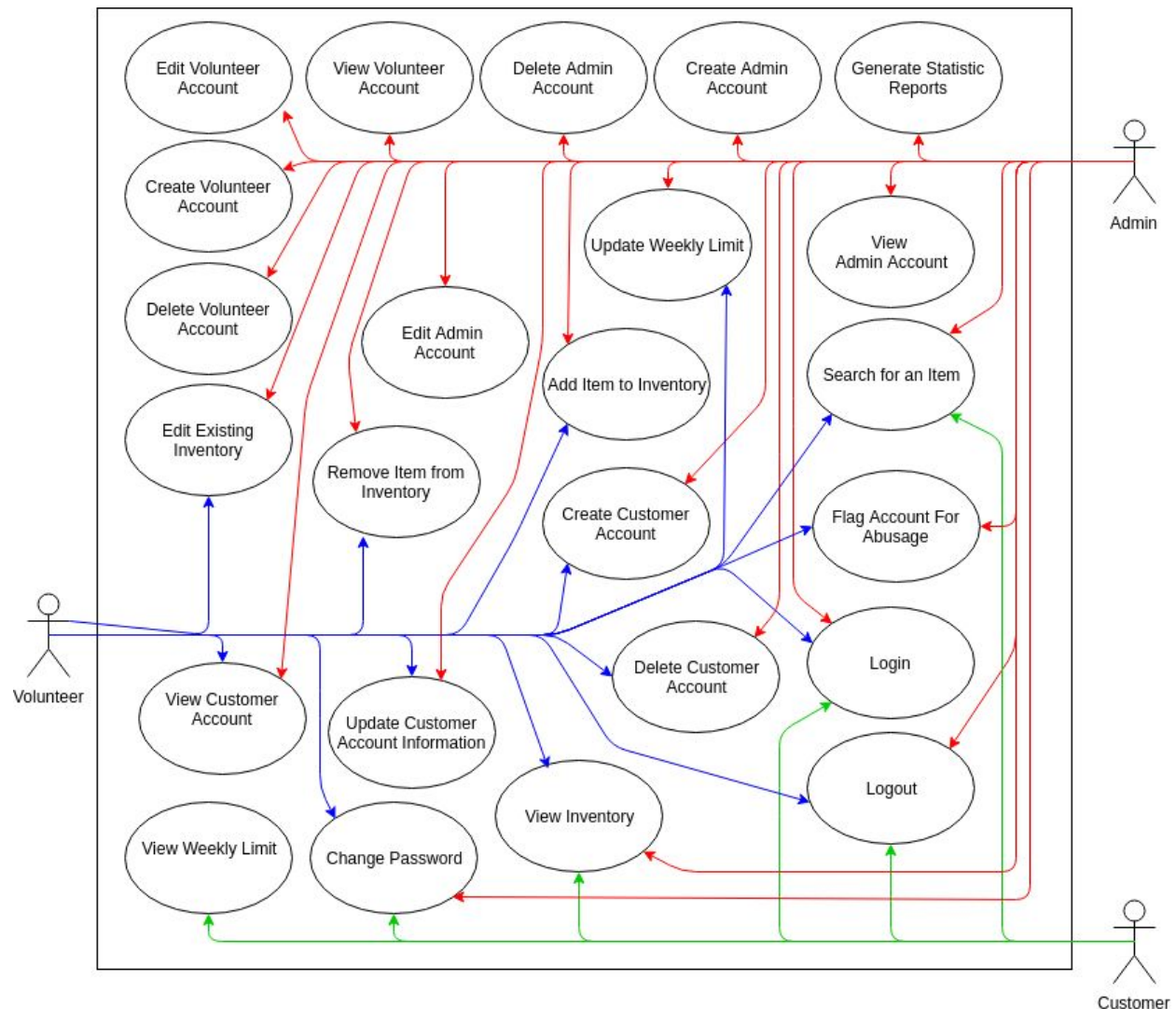
6.2 Entity Relation Diagram

The Entity Relation Diagram below (fig. 2) shows how the entities relate in the system. An admin account must have a list of reports that they have generated for future reference. A report must have a list of customer accounts and inventory that have been tracked in the report. A customer account has a list of items they have taken, for abuse tracking. An inventory item must have a category associated with it for easier searching and filtering. Inventory must have the name, quantity, and the associated category. Lastly, each category must have a name.

Figure 2: ER diagram
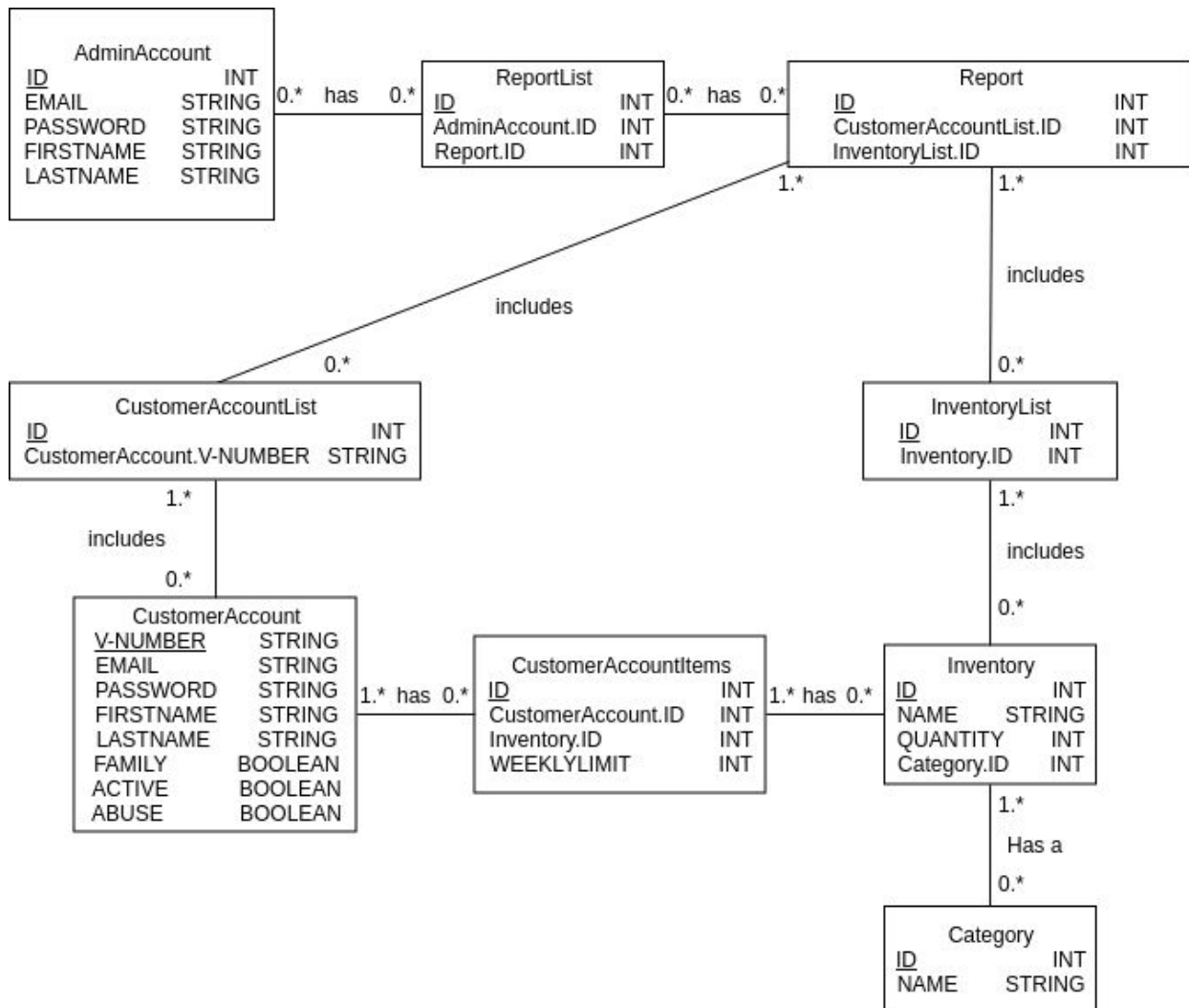
6.2.1 Data Dictionary

Entity	Data Type
AdminAccount Or VolunteerAccount (no report permissions)	ID [Int] + Email [String] + FirstName [String] + LastName [String] + Password [String]
ReportList	ID [Int] + AdminAccount.ID [Int] + Report.ID [Int]
Report	ID [Int] + CustomerAccountList.ID [Int] + InventoryList.ID [Int]
CustomerAccountList	ID [Int] + CustomerAccount.ID [Int]
InventoryList	ID [Int] + Inventory.ID [Int]
CustomerAccount	V-NUMBER [String] + Email [String] + Password [String] + FirstName [String] + LastName [String] + Family [Boolean] + Active [Boolean] + Abuse [Boolean]
Inventory	ID [Int] + Name [String] + Quantity [Int] + Category.ID [Int]
Category	ID [Int] + Name [String]
CustomerAccountItems	ID [Int] + CustomerAccount.ID [Int] + Inventory.ID [Int] + WeeklyLimit [Int]

6.3 Data Flow Diagrams

6.3.1 Context Diagram

The Context Diagram below (fig. 3) shows the entities that interact with the system. The database is an external UVic system so is treated as an entity. The UVic database is used for storing sensitive account and report information. Commands are actions allowed by the system, and messages are information sent from the system.

Figure 3: DFD Level 0
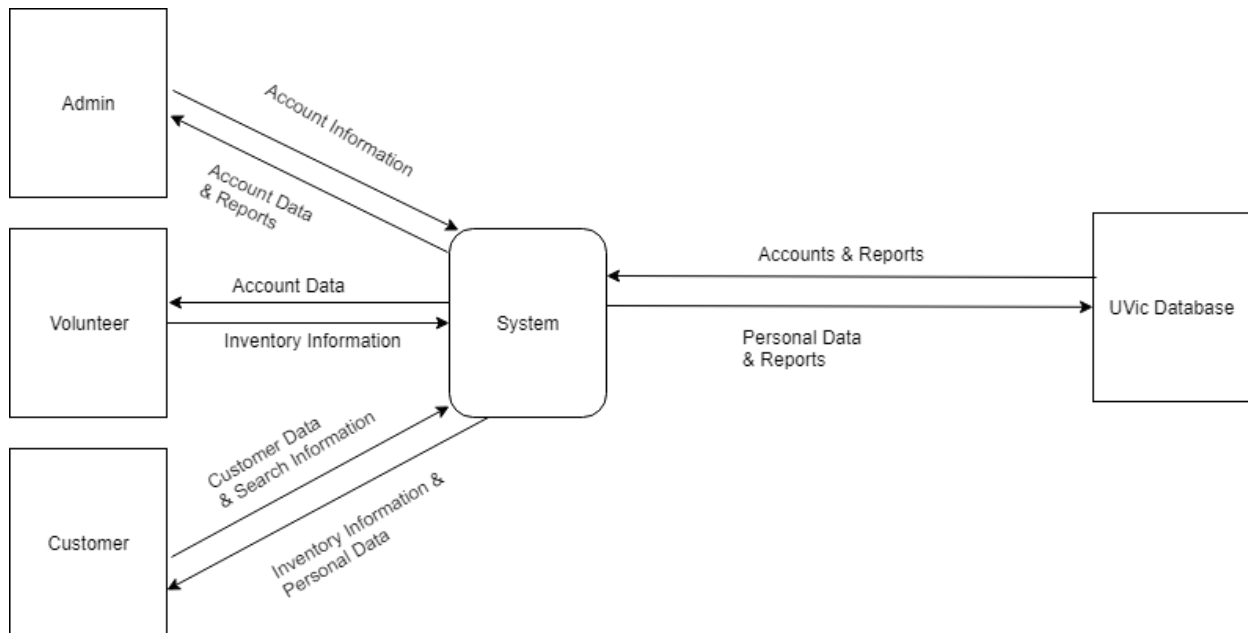
6.3.2 DFD Level 1

The Data Flow Level 1 Diagram below (fig. 4) shows a more detailed flow of data through the system. Admins can generate reports, manage admin accounts, volunteer accounts, and customer accounts. Admin and Volunteers can also manage inventory.

Volunteers and admins, must be able to manage the inventory and customer accounts. A customer must be able to view current inventory in stock as well as their own account to keep updated on their weekly limits. In fig. 4, the report and account datastores are saved within the UVic Database and the inventory is saved in the internal inventory database.

Figure 4: DFD Level 1

6.3.2 DFD Level 2

The Data Flow Level 2 Diagram below (fig. 5) shows an even more detailed flow of data through the system. Management processes are broken into individual processes such as create, update, and delete.

Figure 5: DFD Level 2
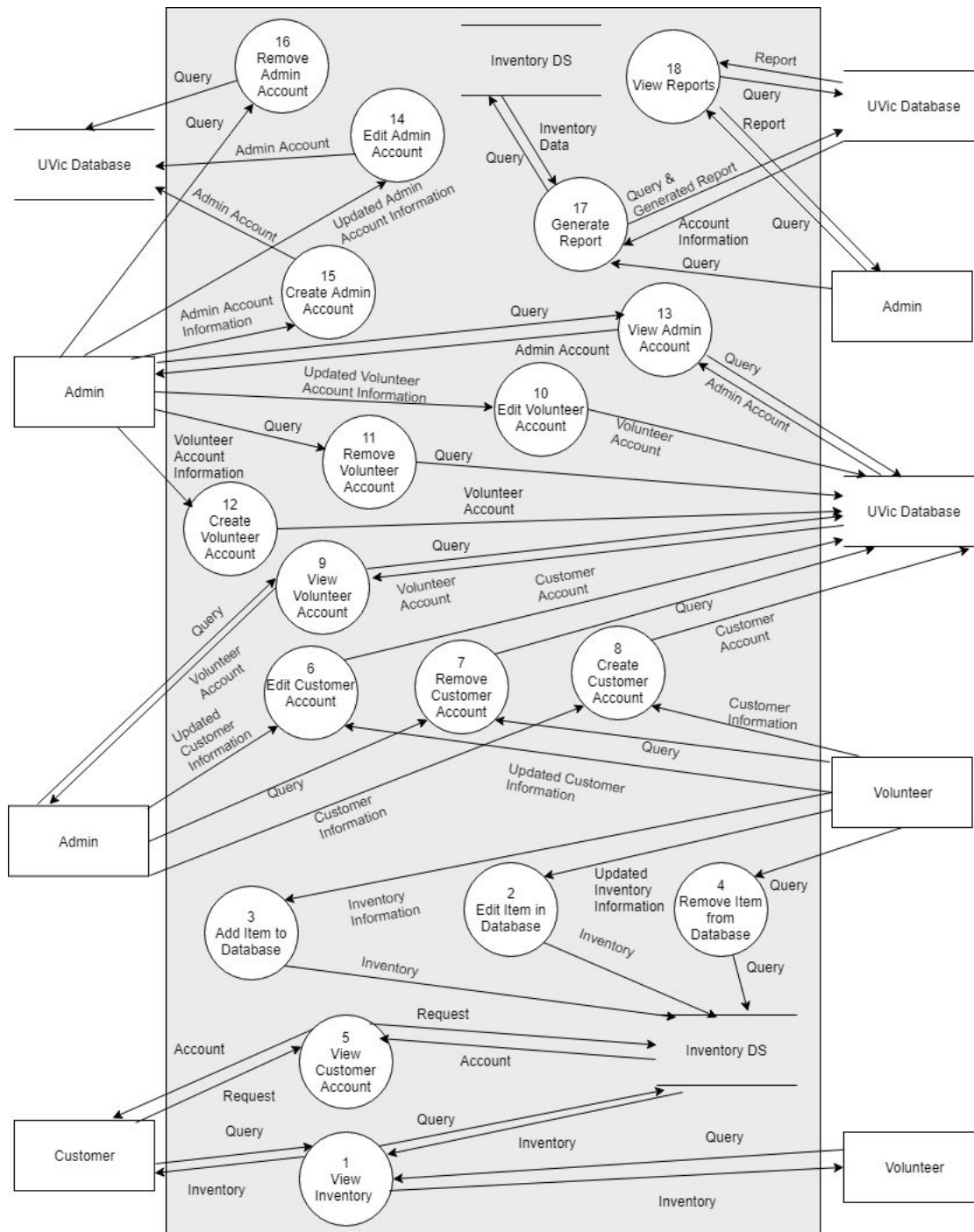
6.4 Traceability Matrix and Test Cases

6.4.1 Traceability Matrix Feature 1

FR-1: The system must provide a way of managing and looking up inventory. It must also be correct and quick to update changes in inventory.

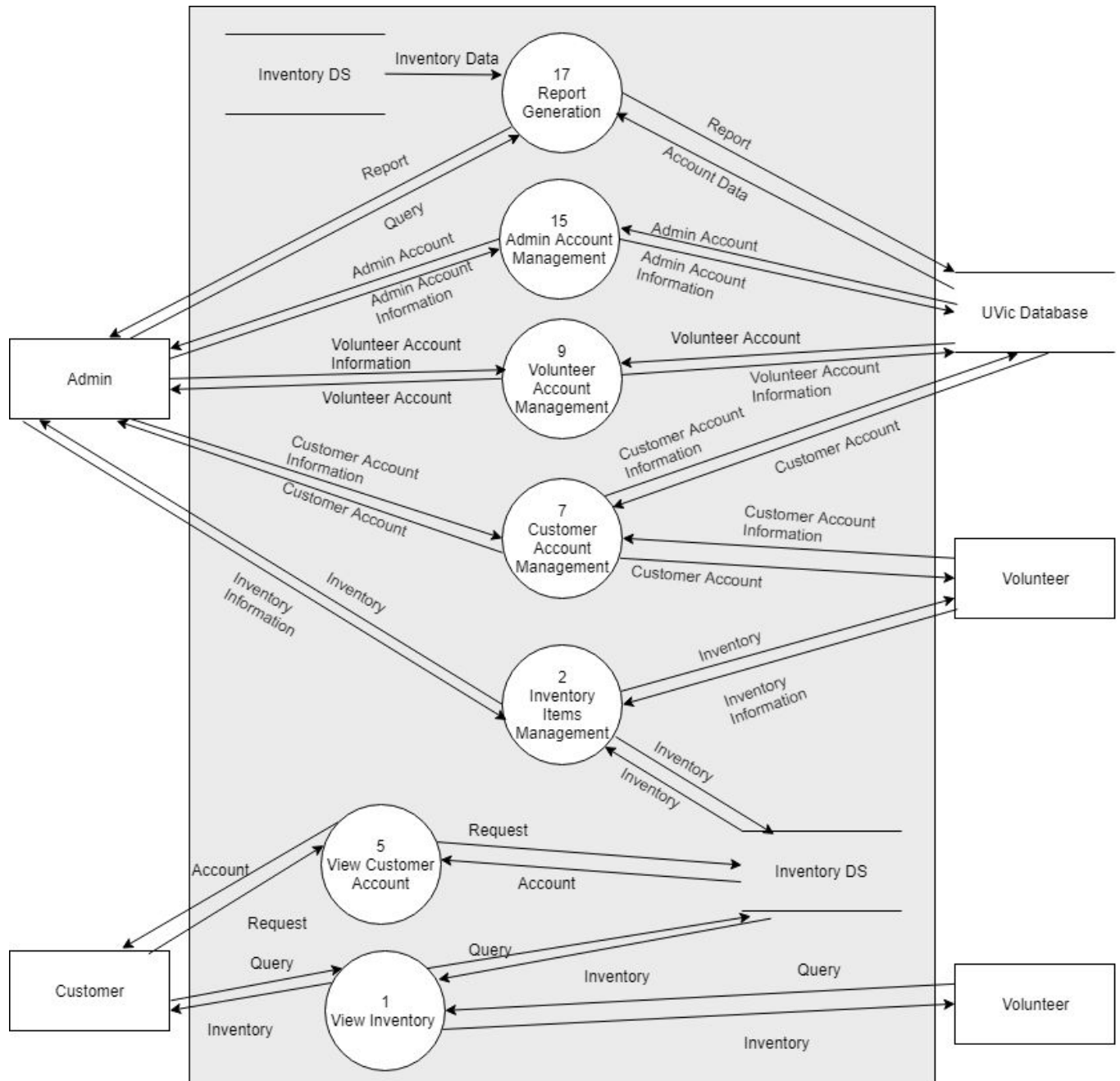| Feature Number	Feature Name	Requirement Number	Requirement Name	TC 1	TC 2	TC 9	TC 13	TC 14	TC 27
FR-1	Inventory	SREQ-1	Real Time Updating of Inventory	X					
		SREQ-3	Managing Inventory		X				
		IREQ-1	Search Bar			X			
		NFR-1	Response Time				X		
		NFR-2	Inventory Update Time					X	
		NFR-13	Inventory Management Time						X
		UC-4	Add Item to Inventory		X				X
		UC-3	Edit Existing Inventory		X	X			X
		UC-5	Remove Item from Inventory		X	X			X
		UC-6	View Inventory	X		X			

6.4.2 Traceability Matrix Feature 2

FR-2: The system must be secure and keep personal information safe (such as statistical data) and allow appropriate access to certain features.

Feature Number	Feature Name	Requirement Number	Requirement Name	TC 22	TC 24	TC 25	TC 26
FR-2	Theft Prevention	NFR-9	Access Control	X			
		NFR-10	Data Release	X			
		NFR-11	Privacy Policy		X		
		NFR-12	ID Validation			X	
		NFR-12.1	ID Verification				X

6.4.3 Traceability Matrix Feature 3

FR-3: The system must allow for the existence of user accounts, each with a respective weekly limit, and that are managed and controlled by admins/volunteers.

Feature Number	Feature Name	Requirement Number	Requirement Name	TC 18	TC 28	TC 3	TC 4	TC 5	TC 25	TC 31	TC 32	TC 33	TC 34
FR-3	Account System	NFR-5	Account Removal	X									
		NFR-14	Account Creation Time		X								
		SREQ-4	Accurate Tracking of Account Limit			X							
		SREQ-5	User Account Verification				X						
		SREQ-6	Manage Accounts					X					
		UC-1	User Login							X			
		UC-2	User Logout							X			
		UC-8	Create Customer Account		X		X						
		UC-9	Update Customer Account Information					X					
		UC-10	Delete Customer Account	X				X					
		UC-11	View Customer Account					X	X				
		UC-12	View Volunteer Account					X	X				
		UC-13	View Admin Account								X		
		UC-14	Create Volunteer Account		X		X						
		UC-15	Create Admin Account		X		X						
		UC-16	Edit Volunteer Account									X	
		UC-17	Edit Admin Account									X	
		UC-18	Delete Volunteer Account					X					
		UC-19	Delete Admin Account					X					
		UC-20	View Weekly limit			X							
		UC-22	Update Weekly limit			X		X					
		UC-23	Change Password										X

6.4.4 Traceability Matrix Feature 4

FR-4: The system can generate statistical reports for admin's. The system must be able to store and filter this data.

Feature Number	Feature Name	Requirement Number	Requirement Name	TC 6	TC 8	TC 7
FR-4	Statistical Data	SREQ-7	Different Views	X		
		SREQ-9	Generate Statistics Report		X	
		SREQ-8	View Statistic Reports			X

6.4.5 Traceability Matrix Feature 5

FR-5: Users must be able to access and use system.

Feature Number	Feature Name	Requirement Number	Requirement Name	TC 10	TC 11	TC 12	TC 15	TC 16	TC 18	TC 19	TC 20	TC 21	TC 29	TC 30
FR-5	System Performance	IREQ-2	Device Compatibility	X										
		IREQ-3	Data Storage		X									
		IREQ-4	Browser Compatibility			X								
		SREQ-10	Simultaneous Changes				X							
		NFR-4	UVic Database Capacity					X						
		NFR-4.1	Inventory Database Capacity						X					
		NFR-6	Time or Space Bound							X				
		NFR-7	Down Time								X			
		NFR-8	Usability									X		
		NFR-15	Training Time										X	
		NFR-16	Readability											X

6.4.6 Test Cases

TC#1: Customers must be able to view the current available inventory, consistent with the real time in-store inventory situation. If what customers view on the FBFS web application is consistent with real time in-store inventory situation then the test passes.

TC#2: Items in inventory will be added, made changes to, and deleted. If all changes are seen when the inventory page is refreshed then the test passes.

TC#3: A test account will be made and the customer limit set to 2. One item should be withdrawn and the new remaining limit shown under the account summary should be 1. If it is still 2, then there is an error in the system and the customer tracking has failed.

TC#4: A test account with a false V-Number will attempt to be created and must fail on creation. Another test account with a valid V-Number will be created and must succeed.

TC#5: A test account for each type (admin, volunteer and customer) will be created and will undergo being modified and deleted. If the account is visible, able to be modified with no errors and deleted then the test passes.

TC#6: One test account of each authority is created (admin, customer and volunteer) to ensure that each account can only view the parts of the web application that apply to their authority.

TC#7: One test account of each authority is created (admin, customer and volunteer) to ensure that each account can only access the parts of the web application that apply to their authority. If every test account is unable to access/see anything outside of their authority then the test passes.

TC#8: A report is generated after the "Generate New Report" button is selected, the test passes if the new report appears under "Saved Reports".

TC#9: An item that is already existing in the system is searched for by title or by category. If the item in question is found then the test passes.

TC#10: All use cases will be attempted on each device and operating system. The test passes for a device with a given operating system if all use cases can be completed.

TC#11: A script will be written to ensure all saved data is in stored inside UVic's database, if any data is found outside of the UVic database system then the test fails.

TC#12: All use cases will be attempted on each version of Internet Explorer after and including Version 11.0.85 and on each version of Google Chrome after and including Version 69.0.3497.92. The test passes if all use cases can be completed on all specified versions of Internet Explorer and Google Chrome.

TC#13: Several types of queries will be sent to the system (Such as creating accounts), if the system responds within three seconds then the test passses.

TC#14: Test modifications will be applied to the inventory. If they are visible within one minute then the test passes.

TC#15: Have two testers attempt to change the inventory at the same time, if all but one user is locked out from changing the inventory than the test succeeds.

TC#16: Verify that the system can support up to 1500 customer accounts, if the system can support 1500 customer accounts and function within the limits of all other tests then the test passes.

TC#17: If the database can support up to 10 000 unique items in the inventory then the test passes.

TC#18: Ensure accounts are removed one year after account owner has left UVic, by using modified timestamps. If the account is successfully deleted then the test passes.

TC#19: Simulate 100 users accessing the system at once, and verify that the system is fully functional, if the system runs within the bounds of other tests then the test passes.

TC#20: Measure the amount of time that the system does not work as per requirements in this document or is inaccessible. If this measured time is less than 10% of the total time running then the test passes.

TC#21: All use cases should be accessible within 5 webpage changes, only then will this test pass.

TC#22: Verify that only admins and volunteers can access personal data by having someone attempt to hack into the system. If the 'hacker' is unable to do so then the test passes.

TC#24: System must be checked against UVic Protection of Privacy Policy to ensure it is being kept to the policies standards. If it isn't then the test fails.

TC#25: Have a tester try to 'cheat' the system and gain more items than their actual allotment. (This test must ignore the fact family status is based on an honour system where the word of the customer is trusted.) If the tester is unable to 'cheat' the system then the test passes.

TC#26: Attempt to create a fake ID that can trick the system into believing it's an authentic ID, if this is not possible then the test passes.

TC#27: Have a volunteer or admin run through UC-3,4, and 5 with a timer, if each can be completed under one minute then the test passes.

TC#28: Have a volunteer or admin create an account from scratch and time the process. Then have the volunteer or admin make several changes to the account, also with a timer. If the timer does not surpass 5 minutes then the test passes.

TC#29: Have an untrained volunteer create and manage a customer account, if the volunteer is able to do this without much help under ten minutes then the test passes.

TC#30: Have a random test subject read the inventory, if the inventory is easy to read and understand then the test passes.

TC#31: Have a tester log in and out. The system must log into the correct account, it must also log out of the account successfully, if this can be done with no troubles then the test passes.

TC#32: Using an admin account, create a new admin account. If the newly created admin account is visible under the "Admin Accounts" tab on the "View Accounts" page by both admin accounts, then the test passes.

TC#33: Verify all account information can be edited (demographic information and flagging). If so then this test passes.

TC#34: Ensure password can be changed and the password updates correctly, allowing the user to log in using the new password. If this can be done then the test passes.

7 Appendix

Interactive UI Model:
https://xd.adobe.com/view/4d79f1fd-1802-4820-7537-ad75ddb2f68c-3b48/?fbclid=IwAR33X-7vUwtkxg_eGb_iPjyV8BDQtXc_XLdAGw4XnNR5Z-lhEhxYCVhABaE&fullscreen