Software Requirements Specification

for

SmartShoppers Online System

Version 1.0 Approved

Prepared by

Amer Alshoghri (214992291)
Julia Paglia (214762926)
Shawn Verma (215064447)
Joshua Abraham (214334643)
Dennis Bragilevsky (214230445)
Sam Guraya (214268916)

November 12, 2020

Table of Contents

Revision History

Name	Date	Reason for Changes	Version
SRS SmartShoppers Online System	Nov 13, 2020	Initial Release	Version 1.0

1. Introduction

1.1 Purpose

The purpose of this document is to provide a complete and thorough account of all the requirements for the SmartShoppers online system. The SmartShoppers system is an interface to allow customers to find items in stores with a greater precision and efficiency. This system will also give the best ordering of items within the store selected by the customer, to allow for a more efficient and enjoyable shopping experience. Other additional features such as suggested items, and sale items will also be included within the system-to-be to better enhance the user experience.

This document describes all the facets of the online system, including interactions with the system as a customer, store manager, and system administrator, differentiating the functionality based on the user. In addition to the specific users, project stakeholders are also included. This document also includes all specific functional and non-functional system requirements, and the high-level system interactions. The overall scope of this document will cover SmartShoppers online system at its *Version 1.0* iteration.

1.2 Document Conventions

In the context of this document, references to other sections will be italicized. Additional wording references include:

- If the document mentions "product", it is referencing the SmartShoppers online system-to-be.
- If the document mentions "items", it is referencing the items available in SmartShoppers store locations.
- The document mentions "quantity" when referring to the amount of an item a customer is adding to their list. When referencing the available store inventory, the document describes this as "stock".

1.3 Intended Audience and Reading Suggestions

This SRS document is intended for the software developers integrating the online system, the project manager for their understanding of the system, and quality assurance testers to explain planned behaviours.

The *1. Introduction* section along with all its subsections of the document is intended for all readers. For *2. Overall Description*, the same applies as it did for section 1. Section *3. External Interface Requirements* is intended for software developers and quality assurance testers. For section *4. System Features* and section *5. Other Nonfunctional Requirements*, the same applies as it did for section 3. The project manager must have understanding of all sections where software developers are the intended readers.

1.4 Stakeholders

The stakeholders for the SmartShoppers online system are ShoppersLand Inc, the retail chain that is designating this online system to be made, with Dr. Song Wang being the representative proposing this product, the employees of SmartShoppers, as they will be using the system for organizational and optimization purposes, and customers of SmartShoppers, as the online system is designed for customers to find products with greater precision. Finally, the developers creating this product are stakeholders as they will both create and maintain the product for SmartShoppers.

1.5 Product Scope

The SmartShoppers online system is an interface with the goal of allowing customers to find items in their stores with a greater precision and greater efficiency than its current system. This system will allow customers to efficiently go through a list of their specified items depending on the store the customer visits. By gathering their shopping list and their location, an algorithm will determine the best order of which to gather the items to speed up the customer's shopping time.

Due to SmartShoppers' goal of being a company that strives to provide the best shopping experience for their customers, this online system will benefit SmartShoppers Inc. as customers will constantly free up more time for themselves, attract more customers by increasing satisfaction due to the system's customer-focused mindset. With this system, employees and managers will benefit due to more downtime to work on store tasks and focus on customers that truly need help. This is to support their endeavour of continually trying to find innovative ways to help customers have a better experience in their stores.

1.6 References

For the use case diagrams for the customer actor, the manager actor, and the system administrator actor, please see *use_case.pdf*

The reference mentioned about also outlines fully dressed use cases, containing a detailed description of each use case.

2. Overall Description

2.1 Product Perspective

SmartShoppers online system is to be a new, self-contained online system for customers of ShoppersLand Inc. This SRS defines the SmartShoppers system, including the system overview, interfaces, features, and system requirements.

ShoppersLand Inc. currently does not have an online system for customers to access. The system-to-be created should assist customers when shopping at ShoppersLand. The system will be a website platform that allows users to create a shopping list and find the items within the list. Once a user accesses the SmartShoppers platform, they will be able to select a store, where they can view and select available items from the store's catalogue. The user is then able to add items to their shopping list. Once the user has finished adding items to their shopping list, the system will find the best path ordered by isles for the user to find each of the items in the shopping list, in store.

The system will allow store managers to update the store inventory, prices, sales. As well, system administrators will have the functionalities of adding and removing stores and their respective managers.

2.2 Product Functions

This section provides a high-level overview of the system's functions organized per user.

Product:

- The product should have a login service for customers, managers, and system administrators
- The product should store all relevant item information, including name, description, image, price, available stock, sizes (if applicable), location within the store, and sale details (if applicable)
- The product should maintain the best order of items for a customer's active shopping list, based on the item aisle location within the store
- The product should provide item suggestions to customers based on frequency of searches

Customer:

- Customers should be authenticated via unique login credentials
- Customers should be able to view a list of nearby stores given their postal code/city and province
- Customers should be able to change their selected (and preferred) store location
- Customers should be able to search for items by name or category
- Customers should be able to view list of items on sale
- Customers should be able to view an item's details, including name, description, image, price, available stock, sizes (if applicable), location within the store, and sale details (if applicable)
- Customers should be able to add an item to their shopping list

- Customers should be able to edit their shopping list (edit quantity, remove items, empty list)
- Customers should be able to view a list of suggested items
- Customers should be able to view their active shopping list
- Customers should be able to view a history of their previous inactive shopping lists

Store Manager:

- Store managers should be authenticated via unique login credentials
- Store managers should be able to select a store they want to edit items for
- Store managers should be able to search for items by name
- Store managers should be able to edit sale items
- Store managers should be able to edit item details
- Store managers should be able to add new items to their selected store location
- Store managers should be able to remove items from their selected store location

System Admin:

- The system admin should be authenticated via unique login credentials
- The system admin should be able to add and remove store locations
- The system admin should be able to add and remove store managers
- The system admin should be able to assign a manager to one or more store locations

2.3 User Classes and Characteristics

The following are the different classes of users that will be utilizing this product.

1. Primary Users
   - This class will include customers who are shopping on the website. They will have no access to other user's accounts. At a high level, their usage is simply limited to browsing items and editing their active shopping list.
   - Each customer will have their own profile where they can update their personal information and save their preferred store location.
   - The product will handle the authentication of all customers.

2. Secondary Users
   - This class will include store managers who are responsible for editing sales, updating item inventory, and editing item information. They will only have access to the details of their associated store locations.
   - This class also includes system admins who are responsible for editing store locations, editing managers, and assigning managers to stores.
   - The product will handle authentication of a store manager and system admin.

2.4 Operating Environment

The SmartShoppers system will be deployed as an online service product and use remote databases to store data. It will therefore be accessible through the world-wide-web, remotely from any internet capable device. The system will be web-based to allow for usage across platforms, including desktop, Android, and iOS.

2.5 Design and Implementation Constraints

2.5.1 Hardware Constraints

The system should have flexible hardware usage policies to accommodate a different number of users at any given point in time. For instance, Amazon AWS provides storage services that are cost and space flexible depending on how many users are using the specific service.

Using in-house hardware, while more adaptable, requires a larger investment, bigger overhead, and requires investing in maintenance and operational (i.e. DevOps) teams to upkeep the servers.

2.5.2 Database Constraints

A database is required to store user information along with related passwords and other information. A secondary database is not required, but heavily preferred, for storing images or other media storage services (For example, a CDN or AWS S3).

2.5.3 Design Conventions

This product should be designed in the context of a SaaS product, as the organization this software will be delivered to will not be responsible for maintaining the product.

2.5.4 Application Constraints

This product requires that it is connected to the internally used inventory system by the physical stores to allow dynamic updating of in-store inventory to reflect the purchasing of products. This is an external system that does not exist in this document but would be connected to function properly.

2.6 User Documentation

There will be one core piece of documentation that will be included with the software release, which is the user manual that will detail the overall usage of this product, and all features that are available within it. This document will cover program usage from the system administrator and store managers.

2.7 Assumptions and Dependencies

It will be assumed that the online system will have the appropriate infrastructure to support the load of the users through cloud services. It is also assumed that the cloud services used will be reliable and dependable to ensure high uptime, and efficient load balancing.

It will be assumed that the product will not need to update inventory counts, but rather the in-person store system will update inventory quantities as customers go in-store to purchase items in real-time.

3. External Interface Requirements

3.1 User Interfaces

SmartShoppers Online System

Customer User Interface:

1) Login Page
    a) The login page shall request a login authentication in the form of username and password, and a sign up if it is the first time the user accesses the website.
    b) A warning will be shown to the customer if the username/password is incorrect.
    c) This login is to display the saved cart of the user and their preferred store location to their account.

2) Select Store Page
    a) The select store page will be an introductory screen to the SmartShoppers website, where the customer can input their postal code and/or the customer's city and province.
    b) This will generate a list of stores within their area to select from as the location the customer prefers to shop at.

3) Main Page
    a) The main page will include a search bar for the customers to search for products they want to see if it is available at their preferred location.
    b) There shall include a "Suggested Items" button that directs the customer to the Suggested Item Page (listed below)
    c) There shall include a "Sale Items" button that directs the customer to the Sale Item Page (listed below)
    d) There shall be a "Change Preferred Store" button that allows the user to change their preferred store location.
    e) There shall be a "View Shopping List" button to view the active shopping list of the customer.
    f) There shall be a "Order History" button to view previous shopping lists (listed below).

4) Item Page
    a) When selecting an item, a web page will display the item, along with every parameter of the item.
        i) Item name, description of item, item sizing options, item price, an image of the item, item location within the store, current stock of the item, sale price, and sale duration for currently discounted items.
    b) It will display the size of which to select (if possible), and the amount to select (if possible).
    c) It will display an "Add to shopping list" button to add the selected item, along with its size and quantity (a warning will be given if the item cannot be added for any reason).

5) Suggested Item Page
   a) The suggested item page will display a list of suggested items for the customer.
   b) This suggested list shall consist of items that are the most frequently searched for items at that store location the customer has selected under their account.

6) Shopping List Page
   a) When the customer has selected all the items they wish to purchase from the store, the customer can click their "Shopping List" button which will output an ordered list of items and where to find the items in the best gathering order (starting at the front of the store).
   b) The customer can add more items to their shopping list and items will be added in a way to maintain the best order.

7) Sale List Page
   a) The sale list page will display a list of discounted items for the customer at the store location that is saved under their selected location.

8) Order History Page
   a) The order history page will display previous lists that were created by a customer.
   b) These previous lists could have been listed as complete, cancelled by the customer, or deleted by the system.
   c) These previous lists cannot be edited by the customer.

Manager User Interface:

1) Login Page
   a) The login page shall request a login authentication in the form of username and password.
   b) A warning will be shown to the manager if the username/password is incorrect.
   c) The login is meant specifically for managerial users.

2) Main Page
   a) The main page will show all the actions that a managerial user can proceed with as buttons that the user can select.
      i) Select Store
      ii) Add Item
      iii) Remove Item
      iv) Edit Item
      v) Search Item by Name

3) Select Store
   a) This process flow will allow for the manager to select the store from the stores they are assigned to continue any of the other flows (ii-v).
   b) If the manager is assigned multiple stores, it would list all their stores. If the manager is assigned one store, only that store will be displayed.

4) Add Item
   a) This process flow will ask for all the parameters that all items need to be added to the inventory of the store.
      i) Item name, description of item, item sizing options, item price, an image of the item, item location within the store, current stock of the item, sale price, and sale duration for currently discounted items.
      ii) A warning will be shown to the manager if any of the fields are incorrect for any reason (i.e. sale price > item price).

5) Remove Item
   a) This process flow will ask for an item to be removed, where the manager will then input.
   b) This item will then need to be confirmed to be removed and once done so, will disappear from the product list for that specific store the manager is logged in for.

6) Edit Item
   a) This process flow will ask for an item to be edited, where the manager will then input. All parameters of the item will then be able to be edited, where the manager will input the edit (i.e. sale on the item, change of description, etc.)

7) Search Item by Name
   a) This process flow will ask for an item to be searched for, where the manager will then input the item name.
   b) If no item is found, the system will notify the manager.

System Administrator User Interface:

1) Login Page
   a) The login page shall display a login authentication for the user.
   b) The login is meant specifically for system administrator.

2) Main Page
   a) The main page will show all the actions that an administrative user can do as buttons that the user can select.
      i) Add Manager
      ii) Remove Manager
      iii) Add Store
      iv) Remove Store

3) Add Manager
   a) This process flow will add a new manager to the system and assign them to a store location or multiple store locations.
   b) It will assign the manager an authenticated username and password for a manager-level login.

4) Remove Manager
   a) This process flow will remove a new manager from the system using their username.

b) This will remove the manager from all corresponding stores that the manager is assigned to.

5) Add Store
    a) This process will add a store to the list of stores for SmartShoppers Inc. and will ask for a location parameter.

6) Remove Store
    a) This process will remove a store from the list of stores for SmartShoppers Inc. and will ask for the store that needs to be removed.

3.2 Hardware Interfaces

This is a standalone piece of software that will be supported on any device (mobile and desktop) through a website. This is to maximize the number of users reached to bring more traffic into SmartShoppers stores. Due to the nature of this online system, there are no hardware components to the product.

3.3 Software Interfaces

The SmartShoppers online system must store many items available in the store and login credentials for all users. As such, two data tables shall be implemented, one that can be accessed for any item-related requests (i.e. add, remove, edit, search) and one for verifying credentials of its users (i.e. customers, managers, system administrators) with the use of an external authentication system.

The queries for the item table will allow for additions, removals, or editing, which shall result in a confirmation response or a denial response if the proper parameters are not added. While a query for a search shall result in either the searched item, a list of items belonging to the searched category or a denial response as the item/category does not exist. The queries for the latter shall result in a successful login response if valid within the data table, and an unsuccessful login response if the username/password are invalid.

3.4 Communications Interfaces

The SmartShoppers online system shall be a web service that is accessible to any user with an internet connection. In addition, special system access will be available to predefined store devices where manager-level authentications are valid (manager-level authentication will not work on all devices). Since this system will need a web browser to run on, there shall be the need for HTTP requests to be sent from the user interface to the core API. With every request from the user sent to the backend, the backend shall manipulate the request and follow the command issued. These requests can include any of the following:

1) A request to access the item data table to either add, remove, or edit an item (for either the customer or manager).
2) A request to access the user data table to either add or remove authentication for managers.
3) A login/logout request from the user.
4) A request to search for an item or an item category.

These requests will be processed by the backend and send the necessary data to the frontend to accurately display the webpage for the user.

The network shall be secure, and users/employees will need to authenticate themselves in the system before they are granted appropriate access to the website. For this, the security protocol used in this online system is Password Authentication Protocol (PAP).

4. System Features

4.1 Customer Signup

4.1.1 Description and Priority

Description: This feature is a core aspect of the system-to-be, where customers will be able to sign up so that they may be able to login in the future. All other features require a customer to be logged in to use the system, so a signup feature is essential.

Priority: This feature is of **High** priority, as all other products components require customers to be logged in, and by extension, they must have signed up at one point.

- **Medium Benefit**: 6 – Signing up is required but is only helpful for things like viewing historical lists and saving preferences.
- **Medium Risk**: 5 – Though this is required to use the product from the customer's perspective, there are no changing components based on signing up.
- **Medium Cost**: 5 – Creating a fully secure and scalable authentication system can be expensive.
- **High Penalty**: 9 – If this process is incorrect, the use of the product may be reduced to nothing.

4.1.2 Stimulus/Response Sequences

1) A customer signs up for an account with a username and password
2) The system confirms that the username is unique, and the password meets adequate standards
3) The system creates an account for the customer and logs them in

4.1.3 Functional Requirements

REQ-1: The system shall only accept a unique username.

REQ-2: The system shall only accept a password that meets the validation systems requirements.

REQ-3: The system shall create a new profile for the specified user.

4.2 Login

4.2.1 Description and Priority

Description: This component is a core requirement for enabling any user functionality. This feature will directly impact overall store and active list management by authenticating customers, managers, and system administrators prior to them completing any actions on the system-to-be.

Priority: This feature is of **High** priority, as all other components are dependent on the successful authentication of any user.

- **High Benefit**: 7 – The login functionality will ensure that the system is secure. Customers will get a more tailored shopping experience since their preferences will be saved to their profiles.
- **High Risk**: 8 – All management functionality is dependent on whether the store manager has logged in. All shopping list functionality is dependent on whether the customer has logged in. Without it, no users will be able to manage their items.
- **Medium Cost**: 5 – Creating a fully secure and scalable authentication system can be expensive.
- **High Penalty**: 7 – If the login feature is erroneous, the store will not be able to authenticate any of its managers and, therefore, will not be able to successfully manage its items. Similarly, customers will not be able to manage their active lists. This will cause organizational issues for the store and create a negative experience for the customer.

4.2.2 Stimulus/Response Sequences

For Customers:
1) The customer logs in to the system with their username and password
2) The system authenticates the customer, or displays an error if there is an invalid input
3) The system grants the customer access to all features related shopping list management and viewing items

For Store Managers:
1) The manager logs in to the system with their employee ID and password
2) The system authenticates the manager
3) The system grants the manager access to all features related to inventory management

For System Administrator:
1) The system admin logs in to the system with their username and password
2) The system authenticates the system admin
3) The system grants the admin access to all features related to manager and store location management

4.2.3 Functional Requirements

REQ-1:   The system shall authenticate customers based on their username and password.

REQ-2:   The system shall authenticate store managers based on their employee ID and password.

REQ-3:   The system shall authenticate system administrators based on their username and password.

REQ-4:   The system shall allow customers to only have read access to items and inventory.

REQ-5:   The system shall allow store managers to only have read and write access to items and inventory.

REQ-6:   The system shall allow system administrators to have unrestricted access to the system.

REQ-7:   The system shall authenticate the user prior to any subsequent action.

REQ-8:   The system shall notify the user of any failed login attempts.

4.3 Store Picker

4.3.1 Description and Priority

Description: This requirement is a core aspect of the system-to-be, where customers will be able to select a store based on their geographical location. This feature is the initial point of entry for the system, and the store dictates the items available, their respective prices and in store locations. Customers are also able to update their store location and save their preferred location.

Priority: This feature is of **High** priority, as all other product components require store specific information to operate.

- **Medium Benefit**: 5 – Allowing customers to select a store will create a more tailored experience and will allow them to gain specific information about local sales, and store specific item locations.
- **High Risk**: 7 – All subsequent components of the product from the customer's perspective are dependent on this functionality.
- **Medium Cost**: 4 – The requirement of a paid for mapping service will result in additional incurred costs.
- **High Penalty**: 8 – If the store selector is erroneous, this could pose an issue for other aspects of the program, which are all dependent on the current store selection.

4.3.2 Stimulus/Response Sequences

1) A customer logs into the system with a username and password
2) The system authenticates the customer
3) The customer is prompted to insert their postal code, or their city and province, if no store is saved on the system for that customer
4) The postal code entered is verified or their city and province is verified
5) A set of nearest stores is listed to the customer
6) The customer selects their preferred store
7) The customer chooses to save this selection as their preferred location
8) The system updates their file system with this change in information about the customer profile

4.3.3 Functional Requirements

REQ-1: The system shall authenticate the customer's username and password.

REQ-2: The system shall list the stores closest to the customer at the top of the list.

REQ-3: The system shall authenticate the postal code is valid.

REQ-4: The system shall authenticate the province and city entered.

REQ-5: The system shall store the customer's preferred store location for subsequent use if the customer selects this option.

4.4 Item Search

4.4.1 Description and Priority

Description: Customers can search for store items by their name or category and get back a list of items based on their query. This is an important aspect of the program as searching is a necessity to add items to the customer's list.

Priority: This feature is of **High** priority, as it is a required step to use the product and has many other actions that are dependent on it.

- **High Benefit**: 8 – Allowing customers to search for items by item name is essential in the creation of a list.
- **Medium Risk**: 6 – Without this within the system, only manually looking through store items will be available, hindering the useability of this product.
- **Low Cost**: 2 – This is purely a software-based solution that only requires querying a list of items.
- **Medium Penalty**: 6 – If the search system is not implemented there could be a drastic affect towards the useability of the product. This could impact the widespread adoption of the product, and adversely affect the stores revenue.

4.4.2 Stimulus/Response Sequences

1) A customer logs into the system with a username and password
2) The system authenticates the customer
3) The customer is prompted to insert their postal code, or their city and province, if no store is saved on the system for that customer
4) A specific store's data is loaded in by the system based off the customer's selection or their saved preference
5) The customer searches for an item by name or by category
6) The system returns a list of all items that contain the query specified
7) If no items match the query, this should also be specified

4.4.3 Functional Requirements

REQ-1:   The system shall authenticate the customer's username and
         password.

REQ-2:   The system shall only allow a valid store location.

REQ-3:   The system shall support search item name via text input.

REQ-4:   The system shall support search item category via text input.

REQ-5:   The system shall return all items that match the query.

REQ-6:   The system shall inform the customer, if they enter an invalid query,
         or one with no items that match their query.

4.5 View Item Details

4.5.1 Description and Priority

Description: Customers can select items that they see on a list and view details about
the item such as the item description, size options, price, an image of the item,
location within the store, current stock of the item, sale price and sale duration
for currently discounted items. This is not a critical aspect of the program;
however, it is helpful so that shoppers can have more information about the
items they are looking at.

Priority: This feature is of **Medium** priority, as it is not essential for use of the item but
offers important information that can greatly benefit customers.

- **Medium Benefit**: 6 – Allowing customers to view an item's details is an
  important part of many shopping systems but is not functionally critical.
- **Medium Risk**: 5 – Without this within the system, item information
  would not be available to the customers, which could negatively affect
  their shopping experience and ergo the stores sales.
- **Low Cost**: 2 – This is purely a software-based solution that only
  requires retrieving details about store items.
- **Medium Penalty**: 6 – If displaying additional information is not
  implemented that could drastically affect the useability of the product.
  This could impact the widespread adoption of the product, and
  adversely affect store revenue.

4.5.2 Stimulus/Response Sequences

1) A customer logs into the system with a username and password
2) The system authenticates the customer
3) The customer is prompted to insert their postal code if no store is saved on
   the system for that customer
4) A specific store's data is loaded in by the system based off the customer's
   selection or their saved preference
5) The customer selects an item they wish to view details about

6) The system displays additional information about the item including price, description, an image of the item selected, in store location, and current stock of the item

7) If an item is currently on sale the system shall display the sale price, along with the duration of the sale (where applicable)

4.5.3 Functional Requirements

REQ-1:   The system shall authenticate the customer's username and password.

REQ-2:   The system shall display more details about an item if the customer selects the item.

REQ-3:   The system shall show the most updated information about the specified store item.

REQ-4:   The system shall show any updates in price due to a sale, along with the sale duration (if applicable).

4.6 Active Shopping List

4.6.1 Description and Priority

Description: Customers have access to an active shopping list for the store they have selected. The customer can choose to add, remove, or update items to their list. The shopping list will maintain the total price of the list, as well as user selected details (quantity and size).

Priority: This feature is of **High** priority, as it is a component that is essential for the use of this product by customers. Many other critical features also directly require this function as a previous step.

- **High Benefit**: 10 – Allowing customers to add items to their shopping list details is an essential aspect of the program and cannot be supplemented.
- **High Risk**: 8 – Most key features of this product depend on items existing within the shopping list.
- **Low Cost**: 3 – This is purely a software-based solution, however there can be many items in a list that need to be stored.
- **High Penalty**: 10 – This feature is the core backbone of the entire product; if it is erroneous the entire system becomes obsolete.

4.6.2 Stimulus/Response Sequences

Common Sequence
1) A customer logs into the system with a username and password
2) The system authenticates the customer
3) The customer is prompted to insert their postal code if no store is saved on the system for that customer
4) A specific store's data is loaded in by the system based off the customer's selection or their saved preference

Add Item to Shopping List

1) A customer selects to add an item from: a search result, the more item detail section, on-sale items, or in the suggested item list
2) The customer adds the item to their shopping list
3) If there is no active shopping list a new list is made, if there is an active list, this item gets appended to it
4) The system has the option of including the quantity desired and size (if applicable)
5) The system updates the order of the list to reflect the list in the best order to find items

Edit Item on Shopping List (After at least one addition is made)

1) The customer clicks on an item in their active shopping list
2) The customer edits the quantity or size of the item on their shopping list
3) The system updates the shopping list and total price accordingly
4) The system updates the order of the list to reflect the list in the best order to find items

Remove Item on Shopping List (After at least one addition is made)

1) The customer removes an item from their active shopping list
2) The system updates the shopping list and total price accordingly
3) The system updates the order of the list to reflect the list in the best order to find items

Completion of List (After at least one addition is made)

1) The customer clicks the complete button on their list
2) The system sets the list status to complete
3) The system makes the list non-editable

Cancellation of List (After at least one addition is made)

1) The customer clicks the cancel button on their list
2) The system sets the list status to cancelled
3) The system makes the list non-editable

4.6.3 Functional Requirements

REQ-1:   The system shall authenticate the customer's username and password.

REQ-2:   Items shall only be added to, updated, or removed from an active list.

REQ-3:   If a customer tries to add an item and there is no active list, a new list shall be made automatically.

REQ-4:   The system shall only allow selected items to be added to the active list.

REQ-5:   The system shall only accept a valid quantity (> 0) and size or use default values.

REQ-6:   The system shall limit the quantity that a customer can add to a list based on the maximum available stock in store, if no items are in stock it will show out of stock.

REQ-7:   The system shall notify customers if an item on their active list is no longer in stock, or in a stock that is less than their specified quantity.

REQ-8:   The system shall notify customers if an item on their active list is removed from the store.

REQ-9:   The system shall only allow item removals when there is at least 1 item in the list.

REQ-10:   The system shall allow updates to any item on the list, including changing a valid quantity (1 to max quantity in store), and a valid size (where applicable).

REQ-11:   The system shall update the price and list details to reflect any changes to the active list.

REQ-12:   The system shall update the price of the active shopping list if there is a change in stock that reduces the items on a customer's list, or if an item is removed from the store.

REQ-13:   The system shall update the price of the active shopping list if an item changes price. This should include the item price is updated, the item is put on sale, or the item is no longer on sale.

REQ-14:   The system shall update the item sequence in the active shopping list to reflect the updated order of finding items within the store.

REQ-15:   The system shall update the best ordering of items when items are moved in the store.

REQ-16:   The system shall update the best ordering of items when an item on the active list is no longer in stock.

REQ-17:   The system shall update the best ordering of items when an item on the active list is removed from the store.

REQ-18:   The system shall update the active list when an item on the active list is removed from the store.

REQ-19:   The customer shall be able to make a shopping list complete, and it will be put into the historical list as a complete list.

REQ-20:   To make a list complete, there shall at least one item on it

REQ-21:   The system shall set the status of a list to cancelled if a customer deletes their active list.

REQ-22:   To cancel a list, there shall be at least one item on it.

REQ-23:   The system shall limit the customer to 1 active list at a time.

REQ-24:   If a customer changes their preferred store, the active list, price and ordering of items on the list shall change to reflect the new stores prices, stock availability, and item locations. For any items that do not exist or are of less stock then required will be removed from the list.

REQ-25:   If a list is complete, cancelled, or deleted, the system shall disallow adding, removing, or updating items. The list shall no longer need to maintain changes in stock, locations of items, or changing prices, meaning the list is fully static.

4.7 View Suggested Items

4.7.1 Description and Priority

Description: Customers can view a list of recommended items based on what other items are commonly searched for within the currently specified store. The customer should then be able to add items from this list to their active shopping list.

Priority: This feature is of **Medium** priority, as it is a component that does not directly impact the main functionality of the system.

- **Medium Benefit**: 5 – Having suggested items can help improve the experience of customers and improve sales for the store.
- **Low Risk**: 2 – No other aspect of the program requires suggested items to function.
- **Medium Cost**: 4 – This could be a fairly data expensive solution that requires a lot of tracking and programmatic maintenance which could be expensive for larger systems.
- **Medium Penalty**: 4 – Poor suggestions may result in less sales for the store and reduce the appeal of using this product.

4.7.2 Stimulus/Response Sequences

1) A customer logs into the system with a username and password
2) The system authenticates the customer
3) The customer is prompted to insert their postal code if no store is saved on the system for that customer
4) A specific store's data is loaded in by the system based off the customer's selection or their saved preference
5) A customer selects the recommended items list
6) The system curates a list of recommended items based off frequently purchased items within a store
7) The customer can view additional information about the suggested items, such as price, description, item picture, and location within the store
8) The customer can directly add items into their active shopping list from the recommended list section
9) The system updates the active shopping list's contents, price, and ideal order

4.7.3 Functional Requirements

REQ-1: The system shall authenticate the customer's username and password.

REQ-2: The system shall update the suggested list as new items are added into the store

REQ-3: The system shall update the suggested list to reflect changes of new searches within the store from all customers.

REQ-4: If a customer tries to add an item and there is no active list, a new one will be made automatically.

REQ-5: The system shall only accept a valid quantity (> 0) and size or use default values.

REQ-6: The system shall limit the quantity that a customer can add to a list based on the maximum available stock in store. Customers will be warned if their quantity exceeds the stock.

REQ-7: The system shall update the price and list details to reflect any changes to the active list.

REQ-8: The system shall update the ordering of the active shopping list to reflect the updated order of finding items within the store.

4.8 View Sale Items

4.8.1 Description and Priority

Description: Customers can view a list of items that are currently on sale within the currently specified store. The customer should then be able to add items from this list to their active shopping list.

Priority: This feature is of **Medium** priority, as it is a component that does not directly impact the main functionality of the system.

- **Medium Benefit**: 6 – Having sale items can help improve the experience of customers and improve sales for the store.
- **Low Risk**: 3 – No other aspect of the program requires a sale item list to function; however, sale items can exist in the main search and recommended item sections.
- **Low Cost**: 2 – This solely requires updating the prices of items and maintaining a secondary list of sale items per store.
- **Medium Penalty**: 4 – If sale items are constantly going out of stock, this may cause negative side effects for the ordering logic needing to be constantly updated.

4.8.2 Stimulus/Response Sequences

1) A customer logs into the system with a username and password
2) The system authenticates the customer
3) The customer is prompted to insert their postal code if no store is saved on the system for that customer
4) A specific store's data is loaded in by the system based off the customer's selection or their saved preference
5) A customer selects the sale items list
6) The system fetches a list of currently on sale items
7) The customer can view additional information about the sale items, such as the sale price, duration of sale (where applicable), description, item picture, and location within the store
8) The customer can directly add items into their active shopping list from the sale list section
9) The system updates the active shopping list's contents, price, and ideal order

4.8.3 Functional Requirements

REQ-1:   The system shall update the sale list as new items are put on sale by the store manager.

REQ-2:   The system shall update the price and details about the sale items on the sale list.

REQ-3:   The system shall remove items from the sale list when the store manager removes the sale, or the sale duration is over.

REQ-4:   If a customer tries to add an item from the sale list and the customer has no active shopping list, a new one shall be made automatically.

REQ-5:   The system shall only accept a valid quantity (> 0) and size or use default values.

REQ-6:   The system shall limit the quantity that a customer can add to a list based on the maximum available stock in store.

REQ-7:   The system shall update the price and list details to reflect any changes to the active list.

REQ-8:   The system shall update the ordering of the active shopping list to reflect the updated order of finding items within the store.

4.9 View Historical Shopping Lists

4.9.1 Description and Priority

Description: Customers can view their historical (complete, deleted, cancelled) lists including the specific store that this relates to, and the price of items, quantities, sizes, and total price at the time that the list was made historical.

Priority: This feature is of **Low** priority, as it is a component that does not directly impact the main functionality of the system and does not pose immediate obvious benefit.

- **Medium Benefit**: 4 – Having a historical account of shopping lists is not essential but could be considered a nice feature by some users.
- **Low Risk**: 2 – Since the historical shopping lists are static, and immutable there is no interaction with other parts of the system.
- **Medium Cost**: 4 – Storing all previous lists of every customer on the system for an indefinite amount of time may result in a large amount of needed data storage.
- **Low Penalty**: 2 – These lists are static and have no direct side effects on other components.

4.9.2 Stimulus/Response Sequences

1) A customer logs into the system with a username and password
2) The system authenticates the customer
3) The customer is prompted to insert their postal code if no store is saved on the system for that customer
4) A specific store's data is loaded in by the system based off the customer's selection or their saved preference

5) A customer selects their list of historical shopping lists
6) The system fetches a list of all the customers historical shopping lists
7) The customer selects a specific historical shopping list to view more details
8) The system fetches the historical list, including the store location, items purchased, prices, quantity of items purchased, quantities of items purchased, and total price

4.9.3 Functional Requirements

REQ-1: The system shall authenticate the customer's username and password.

REQ-2: The system shall maintain all historical lists from a customer.

REQ-3: Historical lists shall not be editable.

REQ-4: Details about a historical list shall be shown, including historical prices, items purchased (including quantity, and size), the historical total price, and store location the list is from. The system will only include items that were on the list at time of becoming static, not including any deleted items.

4.10 Store Manager Item Search

4.10.1 Description and Priority

Description: Store managers can search for store items by name and get back a list of items based on their query. This is an important feature of the system because searching an item is necessary to select and edit that item.

Priority: This feature is of **High** priority, as many other actions are dependent on searching for the correct item.

- **High Benefit**: 8 – Allowing managers to search for items by name is an essential step to editing the information of that item.
- **High Risk**: 8 – Without this feature, managers would be required to manually search through all store items, hindering the useability of this system and introducing a higher chance of human error.
- **Low Cost**: 2 – This is purely a software-based solution, and only requires an implementation to filter a list of items.
- **Medium Penalty**: 4 – If this feature fails, managers will not be able to correctly find items and consequently will not be able to make any changes to the item they are searching for.

4.10.2 Stimulus/Response Sequences

1) The manager logs into the system and is authenticated based on their employee ID and password
2) The manager picks the store they wish to edit out of the list of stores they are assigned to
3) The selected store data is loaded into the system
4) The manager searches for an item by name
5) The system returns a list of all items at their location that contain the query specified within the searched item name
6) If no items match the query, the system will specify this

4.10.3 Functional Requirements

REQ-1:   Store managers shall only be able to search an item by name with text input.

REQ-2:   The system shall return all items that match the query, specific to their selected store location.

REQ-3:   The system shall notify the manager if no items are found with their search.

4.11 Store Manager Item Select to View Details

4.11.1 Description and Priority

Description: Similar to customers, store managers can select an item that they see after searching and view details about the item. This is not a core feature of the system; however, it is helpful so that store managers can verify the item they are looking for in order to edit.

Priority: This feature is of **Medium** priority, as it does not directly impact the core functionality of the system but is important for validation.

- **Medium Benefit**: 6 – Allowing store managers to view an item's details is important to ensure they are moving forward with editing the correct item, but this feature is not critical. Similarly, managers can use this feature to verify that item information has been updated correctly after an edit.
- **High Risk**: 7 – Without this feature, item information would be available to customers but not store managers, which may cause issues if the managers cannot see updates that the customers are seeing.
- **Low Cost**: 2 – This is purely a software-based solution, and only requires the retrieval of item details.
- **Medium Penalty**: 6 – If store managers cannot view the details of an item, they cannot verify that they are managing their desired items correctly. Any discrepancies in the item details observed by managers vs. customers may result in a negative customer experience.

4.11.2 Stimulus/Response Sequences

1) The manager logs into the system and is authenticated based on their employee ID and password
2) The manager picks the store they wish to edit
3) The selected store data is loaded into the system
4) The manager searches for an item by name
5) The system returns a list of all items at their location that contain the query specified within the searched item name
6) If no items match the query, the system should specify this
7) The manager selects an item from the returned list of items
8) The system displays additional item details including description, location, price, image, stock, sizes (if applicable), and sale details

4.11.3 Functional Requirements

REQ-1: The system shall support search item name via text input for managers.
REQ-2: The system shall return all items that match the query, specific to their selected store location.
REQ-3: The system shall notify the manager if no items are found with their search.
REQ-4: The system shall display the most recently updated details for any selected item.

4.12 Store Manager Edit Item Details

4.12.1 Description and Priority

Description: Store managers can edit the details of any item at their selected store location. These details include item name, item image, description, location within the store, available stock, price, and sizes (if applicable).

Priority: This feature is of **High** priority, as the organization and management of inventory items is completely dependent on this functionality. All customers will be able to see the changes made to store items and their shopping experience will be dependent on this working correctly.

- **High Benefit**: 9 – Allowing store managers to edit item details is a critical aspect of the system-to-be because it ensures that the store's item information is always up to date. Updated information will result in less confusion and higher efficiency in terms of the customer's overall shopping experience.
- **High Risk**: 8 – Without this feature, there is no way to ensure item details are always updated. The customer's shopping list and overall shopping experience are directly dependent on this feature.
- **Low Cost**: 2 – This is purely a software-based solution, and only requires the updating of item details.
- **High Penalty**: 8 – If store managers cannot edit the details of an item, the system will never be updated to reflect in-store changes.

4.12.2 Stimulus/Response Sequences

1) The manager logs into the system and is authenticated based on their employee ID and password
2) The manager picks the store they wish to edit
3) The selected store data is loaded into the system
4) The manager searches for an item by name
5) The system returns a list of all items at their location that contain the query specified within the searched item name
6) If no items match the query, the system should specify this
7) The manager selects an item from the returned list of items
8) The system displays additional item details including description, and available stock, etc.
9) The manager edits item details which may include item name, item image, description, location within the store, available stock, price, and available sizes
10) The system saves the newly updated item details

4.12.3 Functional Requirements

REQ-1:   The system shall update item details based on input(s) from the store manager, including name, image, description, price, available stock, sizes, and location in the store.

REQ-2:   The system should allow for updating details based on their original types (i.e. name is a text field, stock is an integer, etc.) if a manager chooses to do so.

REQ-3:   The system shall update the stored item data after an edit has been made.

REQ-4:   The system shall display the most recently updated details for any selected item.

REQ-5:   The system shall update the item details for all customers who have the edited item on their active list.

REQ-6:   If an item is restocked, the system shall notify all customers who have that item on their list that the item has been restocked by adding a warning on the active list.

4.13 Store Manager Edit Sale Items

4.13.1 Description and Priority

Description: Store managers can edit the items that are on sale at any given time. This component is required so that customers can search items on sale. This feature is essentially responsible for the management of all sales for the given store location.

Priority: This feature is of **Medium** priority, as the management of sale items is completely dependent on this functionality. All customers will be able to see items on sale based on what is set by the store managers and their shopping experience will be affected by the functionality of this feature.

- **Medium Benefit**: 6 – Allowing store managers to put items on sale will encourage customers to purchase these items and creates an easier shopping experience for the customers.
- **High Risk**: 7 – Without this feature, there is no way to manage sales happening within the store's location. The customer will be unable to search for items on sale and their overall shopping experience is dependent on this feature.
- **Low Cost**: 2 – This is purely a software-based solution, and only requires the updating of item details.
- **Medium Penalty**: 6 – If store managers cannot edit the sales of items in their selected store, the store cannot promote certain items appropriately. Customers will not be inclined to purchase additional items from the store outside of simply their daily needs.

4.13.2 Stimulus/Response Sequences

1) The manager logs into the system and is authenticated based on their employee ID and password
2) The manager picks the store they wish to edit
3) The selected store data is loaded into the system
4) The manager searches for an item by name
5) The system returns a list of all items at their location that contain the query specified within the searched item name
6) If no items match the query, the system should specify this

7) The manager selects an item from the returned list of items
8) The system displays additional item details including description, price, and quantity in inventory
9) The manager edits item details which may include price, sales tag, and sale duration (if applicable), to either put an item on sale or remove a sale from an item
10) The system saves the newly updated item sale details

4.13.3 Functional Requirements

REQ-1: Store managers shall set a new sale price for any new sale item for their selected store.
REQ-2: Store managers shall be able to set a duration for a sale on any item within their selected store.
REQ-3: Store managers shall be able to remove a sale from any item within their selected store.
REQ-4: The system shall display the most recently updated sale details for any selected item.
REQ-5: The system shall update the prices of any new sale items on all active customers lists.
REQ-6: When a sale ends, the system shall revert the item price back to its original value and in all active customer lists.
REQ-7: The system should add all new sale items to the list of existing sale items.
REQ-8: When a sale ends, the system will remove a sale item from the list of existing sale items.

4.14 Store Manager Edit Inventory

4.14.1 Description and Priority

Description: Store managers can add or remove any item from the system for their selected store location. Items are added with the following details: item name, item image, description, location within the store, available stock, price, and sizes (if applicable). This is a core feature for the management of items within each store to reflect new arrivals as well as any items that may be discontinued.

Priority: This feature is of **High** priority, as the organization and management of new and discontinued items is completely dependent on this functionality. It is essential to ensure that the system is always up to date with added/removed items to avoid any issues with customers accessing items that do not exist.

- **High Benefit**: 8 – Allowing store managers to add and remove items is a critical aspect of the system-to-be because it ensures that the store's inventory is always up to date.
- **Medium Risk**: 6 – Without this feature, customers will not be aware of any new items in stock. In addition, they may simply see that an item is "out of stock" when it has been discontinued.
- **Low Cost**: 2 – This is purely a software-based solution, and only requires updates to add/remove from the item list.
- **High Penalty**: 8 – If store managers cannot add or remove items to/from the system, all users will be misinformed about what items are available at the store location.

4.14.2 Stimulus/Response Sequences

Adding a New Item to Inventory:
1) The manager logs into the system and is authenticated based on their employee ID and password
2) The manager picks the store they wish to edit
3) The selected store data is loaded into the system
4) The manager creates a new item with all necessary details including item name, item image, description, location within the store, available stock, price, and sizes (if applicable)
5) If it does not already exist, the system adds the item to the store's item list
6) If the item already exists, the system will notify the manager

Removing an Item from Inventory:
1) The manager logs into the system and is authenticated based on their employee ID and password
2) The manager picks the store they wish to edit
3) The selected store data is loaded into the system
4) The manager searches for and selects an item
5) The manager removes the selected item
6) The system removes the item from the store's item list
7) The system saves the newly updated list of items
8) The system updates all affected active shopping lists

4.14.3 Functional Requirements

REQ-1:   All items in the item list must have a unique integer ID, which cannot be edited.

REQ-2:   Store managers must select a store location that they are associated with prior to completing any subsequent actions.

REQ-3:   Store managers should be able to add an item to the system after specifying all appropriate item details.

REQ-4:   Store managers should be able to remove an existing item from the system.

REQ-5:   The system should update the stored list of items (inventory) after an item addition or removal has been made.

REQ-6:   If an item has been removed from a selected store location, the system should update all active customer lists for that store location.

REQ-7:   If an item has been removed from a selected store location, the system should update all customers with active lists for that store location that the item has been discontinued.

REQ-8:   If an item has been removed from a selected store location, the system should remove an item from the list of suggested items for that store location.

REQ-9:   If an item has been removed from a selected store location, the system should remove an item from the list of sale items for that store location.

REQ-10:   If an item has been removed from a selected store location, the system should no longer account for an item in the best ordering of any active customer lists associated to that store location.

4.15 System Administrator Edit Stores

4.15.1 Description and Priority

Description: System administrators can add or remove a store location in the system. Store locations are added with the following details: name, location, and associated manager(s). This is a core feature for the management of store locations that may be visited by customers.

Priority: This feature is of **High** priority, as the organization and management of store location is completely dependent on this functionality. All locations must be accounted for in order for customers to view all their options.

- **High Benefit**: 10 – Allowing system admins to add and remove store locations is a critical aspect of the system-to-be because it ensures that the system is always aware of all active store locations. These locations will be read directly by the customer to provide them with a more tailored shopping experience.
- **Medium Risk**: 6 – Without this feature, customers will not be aware of all possible store locations.
- **Medium Cost**: 4 – The implementation of this feature will require some more expensive resources, as this system will need to be scalable for multiple stores and their corresponding manager(s).
- **High Penalty**: 8 – If this feature is erroneous, the system cannot provide the customer with an accurate list of nearby store locations. The system may send the customer to a store that is further away or, more severely, a location that has been closed.

4.15.2 Stimulus/Response Sequences

Adding a New Store Location:
1) The system admin logs into the system and is authenticated based on their username and password
2) The system administrator creates a new store with all necessary details including name, location, and associated manager(s)
3) If it does not already exist (not at the same location), the system adds the store to its collection of store locations

Removing a Store Location:
1) The system admin logs into the system and is authenticated based on their username and password
2) The system administrator searches for a store by location
3) The system returns the store based on the input location
4) The system administrator removes a store
5) The system removes the store from its collection of store locations
6) The system saves the newly updated collection of store locations
7) The system updates all active shopping lists associated with the removed location

4.15.3 Functional Requirements

REQ-1: System administrators should be able to add a new store location to the system after specifying name, location, and associated manager(s).

REQ-2: System administrators must associate at least one store manager to each store location.

REQ-3: System administrators should be able to remove an existing store location from the system.

REQ-4:   The system should update the stored collection of store locations after an addition or removal has been made.

REQ-5:   The system should update the status of all active lists for that store location to "Deleted".

REQ-6:   When a system admin removes a store from the system, all associated store managers are also removed.

REQ-7:   When a system admin removes a store from the system, the system should remove that preferred location from all customer profiles.

4.16 System Administrator Edit Store Managers

4.16.1 Description and Priority

Description: System administrators can add or remove store managers to/from any store in the system. This is a core feature to limit the system access to solely store managers.

Priority: This feature is of **High** priority, as the management of store managers is completely dependent on this functionality. This association will be required for authenticating the store managers when using the system.

- **High Benefit**: 9 – Allowing system admins to add and remove store managers is a critical aspect of the system-to-be because it dictates which users will have managerial access to the system.
- **High Risk**: 9 – If admins cannot add managers to the system, there will be no way to authenticate anyone trying to access the system as a store manager. All store manager functionality is dependent on the ability for a manager to log in.
- **Low Cost**: 2 – This is purely a software-based solution, and only requires updates to the collection of store managers and their relationship to their corresponding store locations.
- **High Penalty**: 10 – If this feature fails, the system cannot grant system access to any store managers, essentially preventing anyone from utilizing any store manager features. This will prevent the proper management of all stores and result in a negative customer experience.

4.16.2 Stimulus/Response Sequences

1) The system administrator creates a new store manager with all necessary details and credentials
2) The system administrator searches for a store by location
3) The system returns the store based on the input location
4) The system administrator assigns the new manager to a store
5) The system adds the store manager to its collection of store managers and associates the manager to the specified store location
6) The system administrator searches for a store manager
7) The system administrator removes the manager
8) The system removes the store manager from its collection of store managers and disassociates the manager from the specified store location

4.16.3 Functional Requirements

REQ-1:   All store managers in the manager collection must have a unique employee ID, which cannot be edited.
REQ-2:   System administrators should be able to create new store managers by specifying first name, last name, and associated store locations.
REQ-3:   System administrators must associate each store manager to at least one store location.
REQ-4:   The system should update the stored collection of store managers after a manager addition or removal has been made.
REQ-5:   The system should handle a many-to-many relationship between store managers and store locations.
REQ-6:   The system shall disallow the ability to remove a manager from a store if they are the only manager assigned to it.

5. Other Non-Functional Requirements

5.1 Performance Requirements

These requirements describe the qualities that the product must have under different circumstances.

5.1.1 Load Requirements

- The product should have less than 80 percent CPU load. This is to support the usual load received by the store servers, which is expected to scale up/down based on how many customers tend to access the product.
- The system main memory should be less than 80 percent utilized and should store and retrieve any customers saved shopping lists. It is expected to scale up/down based on a future increase/decrease in customers accessing the system to run commands.

5.1.2 Performance Requirements for Individual Features

- The product should respond to search queries in less than 1 second.
- The product should respond to commands and reflect the updated information in internal databases in less than 1.5 seconds.
- All other features that are a result of a command-query combination should execute, update internal values, and return expected values within a 2 second interval.

5.2 Safety Requirements

As this product is purely software and does not involve interaction with a physical system, the only safety requirement for the customer would be potential data breaches of customer identification data. The actions taken to prevent this will be covered in length in *5.3 Security Requirements*.

5.3 Security Requirements

- The software system needs to implement robust security measures to prevent unauthorized access and/or data breaches, which can result in the exposure of sensitive user data.
- All security features should be incorporated into the product design from the very beginning. Adding security systems later in the development process can lead to unnecessary risk, cost, and development hours.
- All users of the system are to be uniquely identified. This can be done using a basic database schema that correlates users, their associated passwords, and a hidden internal reference ID that is unique to every user.
- Customer passwords must be encrypted before storage. This can be done manually or by outsourcing the authentication features to popular providers such as Google OAuth.
- Each customer must only be able to view their own personal details that are stored in the system. Users should, under no circumstances, be able to view or alter the information of other customers.
- All fields accepting input from the user (i.e. login forms) should have inputs sanitized to prevent any SQL injection attacks.

5.4 Software Quality Attributes

Testability: This software should be modular enough to allow testing tools to perform critical validation testing on each component. Stress testing can be used to simulate real Customer usage to ensure that the system meets all performance requirements as indicated in *5.1 Performance Requirements*. The tools to design and build test cases would be included in the project testing plan and are outside the scope of this document.

Availability: The software being available most of the time is extremely important. Without consistent availability, large clients can incur significant financial losses during service downtimes. It must be ensured that the service is available at least 99.5% of the time, or 48 hours of downtime per year. To accomplish this, any one of the many available cloud providers can offer solutions that guarantee this.

Flexibility: This software should easily be able to accommodate any type of store. This can be done by separating any display logic and store-specific logic from the logic that dictates how our software is controlled. An example of a software architecture style that would allow this includes the Model View Controller design. Under the assumption of an Agile workflow, A new feature should require no more than 2 sprints to implement and should require no more than 1 sprint to test and validate.

Usability: All graphic interfaces that interface directly with the user should have simple menus and use intuitive, straightforward layouts. It should take users no more than 2 clicks to access common functionality, and no more than 4 clicks to access functionality that is occasionally used.

Portability: This software should be designed with the ability to easily port the system logic onto different environments. This can include the requirement for this system's functionality under new operating systems and/or web browsers. This can be done using object-compatible programming languages or source-compatible programming languages. This system must achieve a level of portability such that porting to a new OS architecture or web browser takes less than 3 sprints in an agile context.

5.5 Business Rules

Customer

- Customers can authenticate themselves with their username and password
- A customer can view their own profile details (preferred store location, and their active list, and historical lists)
- A customer can perform queries against the system to see data that may be computed using their own personal information (i.e. searching nearby stores)
- A customer can change their preferred location if one has been previously saved
- A customer cannot view the personal data of any other customers
- A customer cannot view or edit any manager details
- A customer cannot edit any store details
- A customer cannot edit any item details

Manager

- Managers can authenticate themselves with their employee ID and password
- Managers can perform queries against the system to retrieve existing attributes in the system pertaining to items and inventory
- Managers can perform commands to change the details of current items in the system
- Managers cannot view any customer's identification details
- Managers cannot view or change the details of other managers
- Managers cannot edit any store details

System Administrator

- The system administrator can perform queries on the system and have direct access to all data in the system
- The system administrator can perform queries/commands that display/change the details of any manager respectively
- The system administrator can perform queries/commands that display/change the details of a store respectively

6. Other Requirements

No other requirements are applicable to this project. This SRS has covered all the aspects of the proposed system-to-be, SmartShoppers, including product scope, functional requirements, non-functional requirements, operating environment, guidelines, etc.

Appendix A: Glossary

Active Shopping List

The active shopping list is defined as the current list that the customer can edit. This list is the one that the customer can add items, remove, and edit items on the list.

Historical Shopping List

Historical shopping lists have three key types that can exist. The first is a list that the customer has made complete, where they have gone to the store and used this list. The system will then set the status to complete. The secondary type of list is deleted, where the customer deletes their active list to create a new one, the status will be set to deleted. A list will be cancelled if the store the list was associated with is closed down. All three types of lists are static, cannot be edited and do not need to be maintained by the system to change to updates in a store.

Best Order

The best order for shopping list items is defined by the aisle location in ascending order starting from aisle one which is closest to the front of the store. All items within an aisle can be sorted in any order internally.

Item

Within the confines of this document, an item refers to a physical element that a customer can purchase within the store.

Product

Within this document, product refers to the system-to-be.

Manager

The manager refers to the store manager within this document, as in a user who handles managing the physical store and contents within it.

System Administrator

There is one system administrator within this system who controls the addition and removal of all managers, stores, and system level features.

Quantity

Quantity refers to the number of items that a customer can add into their list, which is not the same as stock as defined below.

Stock

Stock is defined as the physical number of items within the store that managers add into the overall inventory.