# Software Requirements Specification

**for**

# Smart Shoppers System

**Version 1.0 approved**

**Prepared by**
Kenan Li
Esra Kastrati
Mostafa Mohamed
Yassin Mohamed

**November 14, 2020**

# Table of Contents

# Revision History

| Name | Date | Reason For Changes | Version |
|---|---|---|---|
| Initial Release | Nov 14, 2020 | Initial SRS document | 1.0 |
| | | | |

# 1. Introduction

## 1.1 Purpose

The purpose of this requirement specification document is to detail the functions and requirements for the new online shopping list system for ShoppersLand Inc. This system is needed to improve the shopping experience for customers of ShoppersLand Inc. a leading retail chain in Canada that sells a large range of products including food, clothing, electronics, and pharmaceutical drugs by providing an innovative way to help shoppers have a better experience in their store.

## 1.2 Document Conventions

Each section of the document is numbered and bolded to allow for easy navigation. When determining the priority of each requirement, the relative benefit, cost, risk, and penalty is taken into account. Each sub requirements has a identifier which shows the section it is in as well as the requirement number

## 1.3 Intended Audience and Reading Suggestions

**Project Managers –** Use this document to evaluate requirements to see if it meets clients (Shoppers Lands Inc.) need

**Developers –** use this document to develop the smart shopper's system following all requirements

**Testers –** use this document to ensure all requirements are met

**Users –** use this document to understand what the system can do

The rest of this document describes the product perspective, available functions the product can perform, the different user classes that will use this system, constraints, and assumptions and dependencies that are made with this project. The document also includes external interface requirements involving user, hardware, software, and communications interfaces. It will also include detailed information on system features and non-functional requirements.

## 1.4 Stakeholders

The stakeholders for our project is the Project Management Team(Our Team) which is responsible for creating the SRS document, which includes why the software is being made, the requirements and its use case. The development team, which is responsible for developing the software using the SRS document provided by the Project Management Team. Professor Song Wang who provided us with the project, general description of the problem and is there for additional support. Additionally, potential customers who will be using this system to make a shopping list of the items that they intend to purchase when they arrive at the store and our immediate stakeholder ShopperLand Inc. as we will be creating the system for ShopperLand Inc. to provide customers that are shopping at their stores with a better experience.

## 1.5 Product Scope

The purpose of the product is to develop an online shopping list system for ShoppersLand Inc. that allows customers to find products in their store faster with a higher precision. Users shall be able to create an account and specify their postal code and/or their city and province. They shall then be able to select a store location that they will be shopping at.The user shall be allowed to change the store location any time they wish. Users shall be able to search for available products, choose from the list of items on sale or suggested items at their set store location, view the item information (description, price, where it is located in store) and create a shopping list of products they intend to buy. An algorithm will be used to sort the items in the list to provide the users with the best order in which to get the item. This product shall not allow users to purchase items online or request items for delivery.

## 1.6 References

"Browser Standards — Office of Information Technology", *Oit.uci.edu*, 2020. [Online]. Available: https://www.oit.uci.edu/browser-standards/. [Accessed: 14- Nov- 2020]

"User Interface Design Basics | Usability.gov", *Usability.gov*, 2020. [Online]. Available: https://www.usability.gov/what-and-why/user-interface-design.html. [Accessed: 15- Nov- 2020].

S. Wang, *EECS4312 Software Engineering Requirements Course Project*, Version 1. 2020.

# 2. Overall Description

## 2.1 Product Perspective

This system is an upgrade of the usual shopping system, where customers walk around the store with a written shopping list searching through different lanes to find what they are looking for. This usually requires a lot of time locating the item, deciding on which brand to purchase and if the price is right.

With this new system users will be able to view the items they are looking for online, look at the price and description and decide if the item is right for them. They can then mark the items down on an online shopping list and be told where to look for them when they get in the store.

## 2.2 Product Functions

- Account Signup/Login
- Select and Change store location
- Create Shopping List
- View Shopping List content
- View items of sale
- View suggested items
- Search for items by name and category
- View item details (name,price, stock, description, location in store)
- Add/Delete items from shopping list
- Add/Delete available store items
- Update item information
- Update items on sale
- Add/Delete employee
- Sort Shopping List

## 2.3 User Classes and Characteristics

**End-User**

Customers who use this system to search for products they wish to buy and create a shopping list for when they go to the store next time. They will have the ability to choose which store location they intend to go to. They will also have the ability to search for an item either by name or category, they can then select the item to view the item's information such as its current price, description, or etc. They can then choose to add the item to their shopping list, which can then be used during their trip to the selected store.

**Admin**

Employees who are using the system to make changes to items or update the system on items that are on sale at a specific location. They will have the ability to search for an item by name or category, when the item is found they can choose to update the information of the item such as its current price, description, if it is on sale, or etc. They can also choose to delete the item from the store or add new items changing the available items customers will be able to add to their shopping list. Depending on their job level they will also have the ability to add or delete employees from the system.

## 2.4 Operating Environment

The system will be an online service that will exist alongside Shopper Land’s website. Customers will be able to access the system using a set of supported web browsers (Google Chrome V.83, Safari V.13.1, Firefox V.76, Microsoft Edge V.45.1, Opera V.88). Users account information, store locations, and information on items will be stored in a database to be used by the system.

## 2.5 Design and Implementation Constraints

Shoppers Land Inc. will be responsible for maintaining the delivered software and supplying the database of items in store and store locations. An additional database will be required to store users account information(passwords, usernames). Language of the system will be in english as it is the most used and

spoken language around the world, thus supporting a wider array of people. The system should also support multiple web browsers.

## 2.6 User Documentation

The software will include a user manual detailing how to use the software and its available features. In addition to this, an interactive tutorial shall be provided along with the software to provide step-by-step instructions on how to use it, so that users will be able to navigate and perform functions as intended.

## 2.7 Assumptions and Dependencies

This project assumes the users have a device that can access the website as well as have sufficient knowledge using a computer or mobile device to access the software. The project is dependent on the database provided by Shoppers Land Inc. to display available store locations and information about products that are available in store and the role of employees to keep item information up-to-date.

# 3. External Interface Requirements

## 3.1 User Interfaces

*The system will provide the user with an accessible User Interface (UI) that will allow them to interact with the system and execute all of the available features and services. The UI will adhere to best practice guidelines to ensure its accessibility to users who are used to certain conventions, and relative newcomers to the platform. The UI will run on any modern web browser and will support and adapt to smartphone browsers. The User will always have access to a "Clear Cart" and a "Help" button; the latter of which goes over the function of the system in case users are newcomers.*

**3.1-R1** The system will provide the user with an interface that follows best practice accessibility and usability guidelines[1].

*Priority: 1*

**3.1-R2** The UI will implement fonts that are easily readable by users and are of appropriate size for average screens.

*Priority: 2*

**3.1-R3** The UI will support all resolutions equal to or higher than 852x480 pixels, and will dynamically adapt the elements on screen to fit to the current browser size and layout.

*Priority: 2*

**3.1-R4** The UI will have a "Clear Cart" button that removes all items from a user's shopping list.

*Priority: 1*

[1] https://www.usability.gov/what-and-why/user-interface-design.html

**3.1-R5** The UI will have a visible indication of the current address of the user, and will allow the user to change it by clicking the address.

*Priority: 1*

**3.1-R6** The UI will have a visible indication of the address of the branch the user is currently browsing, and will allow the user to see other available branches and switch to them by clicking on the address.

*Priority: 1*

**3.1-R7** The UI will have a "Calculate Route" button that will only be clickable if the size of the user's shopping list is non-zero. This button will output the items in the user's shopping list by their order in the store, starting at the front.

*Priority: 1*

**3.1-R8** The UI will provide the user with an overview of the items available for sale in the shop, separated into categories.

*Priority: 1*

**3.1-R9** The user will be able to navigate into each category and only be shown the subset of items that fit into it.

*Priority: 1*

**3.1-R10** The user will always have access to a search bar through which he can filter the current branch's items on sale through text.

*Priority: 1*

**3.1-R11** The user will always have access to a "View Cart" button that prominently displays the user's cart when clicked.

*Priority: 1*

**3.1-R12** The user can click on an item to be provided with more details about it, including a description, price, and its location in the store.

*Priority: 2*

**3.1-R13** When using the search function, the user will be provided a list of suggested items based on other users' trends.

*Priority: 3*

## 3.2 Hardware Interfaces

*The system will run in a web browser on internet-capable devices, this makes the Hardware Interface requirements minimal, as they are handled by the underlying platform on which the system is being used.*

**3.2-R1** The system will run on any web-capable device whose CPU follows the x86, x64, or ARM instruction set/architecture.

*Priority: 1*

## 3.3 Software Interfaces

*The system will run on all modern JavaScript capable web browsers found commonly on devices today (Edge, Chrome, etc.). It will communicate with a database containing a record of the items in each branch to show the user the available stock.*

**3.3-R1** The system will be compatible with the Google Chrome web browser (ver. 86.0.4240.198 or greater)

*Priority: 1*

**3.3-R2** The system will be compatible with the Microsoft Edge web browser (ver. 45.1 or greater)

*Priority: 1*

**3.3-R3** The system will be compatible with the Mozilla Firefox web browser (ver. 76.0 or greater)

*Priority: 1*

**3.3-R4** The system will be compatible with the Opera web browser (ver. 88 or greater)

*Priority: 2*

**3.3-R5** The system will be compatible with all Chromium-based web browsers

*Priority: 3*

**3.3-R6** The system will retrieve the current list of items from a database containing data from all branches.

*Priority: 1*

## 3.4 Communications Interfaces

*The system will run on a web server, and will communicate with web browsers through GET and POST messages following the HTTPS communication protocol. The security and integrity of the connection is ensured through TLS encryption. The system will also communicate with the database through a series of SQL queries.*

**3.4-R1** The system will run on a web server, and will communicate with web browsers via HTTP protocol.

*Priority: 1*

**3.4-R2** The system will communicate back and forth with the user's browser through GET and POST messages.

*Priority: 1*

**3.4-R3** The connection between the server and the web browser is secured through TLS encryption

*Priority: 1*

**3.4-R4** The system will communicate with the database through SQL queries

*Priority: 1*

# 4. System Features

For the system features, we will be separating them into two groups depending on which user class they belong to. (i.e End-User or Admin).

# 4.1 End-User

### 4.1.1 Creating/Editing Shopping List

***i. Description and Priority***

This feature provides the user the ability to create their shopping list for viewing purposes, users would also be able to edit the shopping list through searching for items and adding/removing them from them. it is of High priority seeing as it is one of the main functions of our system, and it responds most to our user needs.
benefit: 9
penalty:7
cost: 4
risk: 2

***ii. Stimulus/Response Sequences***

*<User creates empty shopping list> - <User Searches item> - <Query System Database> - <Display corresponding Result> - <User specifies desired amount of item> -<User adds*

*item to shopping list> - <Shopping Cart updates> - <User inspects shopping cart> - <User deletes Item from shopping cart> - <Shopping List updates>*

#### *iii. Functional Requirements*

4.1.1-R1: The user shall have the ability to create an empty shopping list.
*Priority: 1*
*Error Response: N/A*

4.1.1-R2: The user shall have the ability to search for items using name.
*Priority: 1*
*Error Response: invalid entry → system displays "item not found".*

4.1.1-R3: The user shall have the ability to search for items using category.
*Priority: 1*
*Error Response: invalid entry → system displays "item not found".*

4.1.1-R4: The user shall have the ability to view item information.
*Priority: 1*
*Error Response: N/A*

4.1.1-R5 The user shall have the ability to pick a specific amount of the item.
*Priority: 1*
*Error Response: invalid number (out of bounds) → items not added to shopping list.*

4.1.1-R6: The user shall have the ability to add items to the shopping list.
*Priority: 1*
*Error Response: no existing shopping list → prompts user to create empty shopping list.*

4.1.1-R7: The user shall have the ability to view shopping lists.
*Priority: 1*
*Error Response: N/A*

4.1.1-R8: The user shall have the ability to delete items from the shopping list.
*Priority: 1*
*Error Response: N/A*

### 4.1.2 Locating Stores

#### *i. Description and Priority*

This feature provides the user the ability to view nearby stores. the user will be prompted to enter postal code or city to show nearby stores when opening the system. The user can save preferred locations for future use. The user can also search for a specific store unbounded by their current location. This feature is of high priority.

benefit: 9
penalty:5
cost: 8
risk: 8

#### *ii. Stimulus/Response Sequences*

*<User enters postal code and/or city> - <System queries Available Stores> - <Display corresponding Result> - <User specifies desired stores> -<System displays store content> - <User can save Store to favorites> - <User can search for Store using name>*

***iii. Functional Requirements***

4.1.2-R1: The user shall have the ability to enter their Postal Code.
*Priority: 1*
*Error Response: Invalid Code → System prompts user to enter another code*
4.1.2-R2: The user shall have the ability to enter their city/province.
*Priority: 1*
*Error Response: invalid entry → system prompts user for another entry.*
4.1.2-R3: The user shall have the ability to save desired stores for future use.
*Priority: 2*
*Error Response: store already saved → system displays "store already saved!".*
4.1.2-R4: The user shall have the ability to search for a store using the name.
*Priority: 1*
*Error Response: invalid entry → system displays "store not found".*

## 4.2 Admin

### 4.2.1 System Management

***i. Description and Priority***

This feature provides the employees with the ability to update available items on the system. employees can also edit information for existing items (i.e price, available number, description..etc.). Employees can also search for items using name or category.
benefit: 9
penalty:7
cost: 3
risk: 7

***ii. Stimulus/Response Sequences***

*<Employee adds/remove item from store> - <System updates database> - <Employee Searches item> - <Query System Database> - <Display corresponding Result> - <Employee edits specific item> -<System updates database> - <Shopping List updates>*

***iii. Functional Requirements***

4.2.1-R1: The employee shall have the ability to add/delete items from the store.
*Priority: 1*
*Error Response: N/A*
4.2.1-R2: The employee shall have the ability to search for items using name.
*Priority: 1*
*Error Response: invalid entry → system displays "item not found".*
4.2.1-R3: The employee shall have the ability to search for items using category.
*Priority: 1*
*Error Response: invalid entry → system displays "item not found".*
4.2.1-R4: The employee shall have the ability to view/edit item information.
*Priority: 1*
*Error Response: N/A*

### 4.2.2 Employee Management

#### *i. Description and Priority*

This feature provides certain employees (possible managers) with the ability to update current employees on the system, in order to control access as needed.
benefit: 7
penalty:5
cost: 2
risk: 4

#### *ii. Stimulus/Response Sequences*

*<Manager searches for employee name> - <Manager removes employee from system> - <Authorized access to system modified> - <Manager adds new employee to system> - <Authorized access to system modified>*

#### *iii. Functional Requirements*

4.2.2-R1: The manager shall have the ability to add/delete employees from the store.
*Priority: 1*
*Error Response: N/A*
4.2.2-R2: The manager shall have the ability to search for employees using name.
*Priority: 1*
*Error Response: invalid entry → system displays "employee not found".*

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

- Search requests must respond within a 1000 milliseconds (synchronous database request).
  This requirement is essential seeing as users will constantly search through available items and/or stores. This will also apply to employees and managers when searching for stores. 1000 is a wide enough window seeing as querying the database will take a varying amount of time.
- System can support 200 active users.
  This requirement ensures that the system can withstand a decent amount of people. Since interactions with the system from the user side are observational or affect local storage, 200 is not too high a number.
- System launches on a web browser in 5 seconds or less.

## 5.2 Security Requirements

- There needs to be specified roles for the users. These are "user", "employee" and "manager". Each type of user would be able to interact with the system differently. (i.e a "user" cannot remove an "employee" from the system).
- User accounts will be created from employees and managers, this is due to the scope of their access to the system.
- System should be secure enough to prevent data breach

## 5.3 Scalability Requirements

- Database Scalability: adding servers or storage would allow our system to support a wider range of stores, and store more items.
- Administrative Scalability: the system will be able to scale up to 400 concurrent users.

## 5.4 Software Quality Attributes

**Adaptability -** The system will function on various web browsers and hardware devices

**Availability** – The system should be accessible on the client website most of the time

**Correctness** – All items/users will have the correct information.

**Flexibility** - The system can run automatically for the user without developer interaction.

**Interoperability** - Interact with Google Maps API to provide closest stores and distance.

**Maintainability** - The system will be developed in a way to make adding functionalities easier, this would also benefit developers when fixing possible bugs.

**Portability** - The system should run on various operating systems.

**Reliability** – The system should provide accurate item information, price, location to ensure customers are satisfied.

**Robustness** - Errors in querying the database will be displayed to the user in real-time. Unprecedented system failures will be logged for debugging.

**Testability** - updates to the system can be tested locally to see functionality before implementation.

**Usability** – User's experience with the system is simple and intuitive to use.

## 5.5 Business Rules

- *Business factors such as cost and partnerships to enhance scalability and reach will be considered*

# Appendix A: Glossary

## A.1 Definitions

**Communication Protocol** - Set of rules that allow two or more entities of a communication system to transmit information

**GET** - an HTTP method used to request data from a resource

**Instruction Set/Architecture -** Computer architecture

**Javascript** - Programming Language

**Operating System** - Interface between user and hardware

**POST -** an HTTP method used to send data to a server

**Shopping List** - A list of items the customer intends to purchase when they arrive at the store location

## A.2 Acronyms

**ARM** - Advanced RISC machine

**API** - Application Programming Interface

**CPU** - Central Processing Unit

**DB** - Database

**HTTP** - Hypertext Transfer Protocol

**SQL** - Structured Query Language

**SRS** - Software Requirements Specification

**TLS** - Transport Layer Security

**UI** - User Interface