# Software Requirements Specification

# for

# EasyShop Management System

Version 1.0

Prepared by Rohit Tomer, Anaadhi Mongia and Rushil Jalal

MIT Manipal

9th March, 2024

Table of Contents

Table of Contents [ii](#__RefHeading___Toc441230970)

Revision History [ii](#__RefHeading___Toc441230971)

1. Introduction [1](#__RefHeading___Toc441230972)

1.1 Purpose [1](#__RefHeading___Toc441230973)

1.2 Document Conventions [1](#__RefHeading___Toc441230974)

1.3 Intended Audience and Reading Suggestions [1](#__RefHeading___Toc441230975)

1.4 Product Scope [1](#__RefHeading___Toc441230976)

1.5 References [1](#__RefHeading___Toc441230977)

2. Overall Description [2](#__RefHeading___Toc441230978)

2.1 Product Perspective [2](#__RefHeading___Toc441230979)

2.2 Product Functions [2](#__RefHeading___Toc441230980)

2.3 User Classes and Characteristics [2](#__RefHeading___Toc441230981)

2.4 Operating Environment [2](#__RefHeading___Toc441230982)

2.5 Design and Implementation Constraints [2](#__RefHeading___Toc441230983)

2.6 User Documentation [2](#__RefHeading___Toc441230984)

2.7 Assumptions and Dependencies [3](#__RefHeading___Toc441230985)

3. External Interface Requirements [3](#__RefHeading___Toc441230986)

3.1 User Interfaces [3](#__RefHeading___Toc441230987)

3.2 Hardware Interfaces [3](#__RefHeading___Toc441230988)

3.3 Software Interfaces [3](#__RefHeading___Toc441230989)

3.4 Communications Interfaces [3](#__RefHeading___Toc441230990)

4. System Features [4](#__RefHeading___Toc441230991)

4.1 System Feature 1 [4](#__RefHeading___Toc441230992)

4.2 System Feature 2 (and so on) [4](#__RefHeading___Toc441230993)

5. Other Nonfunctional Requirements [4](#__RefHeading___Toc441230994)

5.1 Performance Requirements [4](#__RefHeading___Toc441230995)

5.2 Safety Requirements [5](#__RefHeading___Toc441230996)

5.3 Security Requirements [5](#__RefHeading___Toc441230997)

5.4 Software Quality Attributes [5](#__RefHeading___Toc441230998)

5.5 Business Rules [5](#__RefHeading___Toc441230999)

6. Other Requirements [5](#__RefHeading___Toc441231000)

Appendix A: Glossary [5](#__RefHeading___Toc441231001)

Appendix B: Analysis Models [5](#__RefHeading___Toc441231002)

Appendix C: To Be Determined List [6](#__RefHeading___Toc441231003)

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

The purpose of this System Requirements Specifications(SRS) is to clearly define the software requirements for EasyShop Management System. Primarily it serves as a compass for the development team, providing a roadmap for the creation of the Supermarket Management System. It encapsulates the project’s objectives, guiding principles, and expectations. Additionally, it acts as a communication tool, ensuring that all stakeholders, from developers to project managers, share a unified vision. By defining the scope, functionalities, and constraints, the SRS sets the foundation for successful system development and deployment.

## Document Conventions

For writing the document for EasyShop management system I am using the following things in my Project. For making this document more readable and understandable I am using times new roman font style and 12 font size, and headings are bold.

## Intended Audience and Reading Suggestions

The intended audience for this SRS includes a diverse group of professionals. Developers will delve Into the technical details, understanding the system’s architecture and features. Project managers will Extract insights into project scope and timelines. Quality assurance teams will utilize the document

For test planning. Stakeholders, ranging from executives to end-users, will gain an overarching Understanding of the system’s purpose and capabilities. For optimal comprehension, readers are Encouraged to engage with the entire document, utilizing the table of contents for navigation.

## Product Scope

The EasyShop management system is comprehensive, aiming to transform how users shop. Beyond traditional checkout systems, it includes user profiles, loyalty programs, and a custom dashboard for inventory management as well as employee details. The database management system can be sold to supermarkets to help streamline their transactions and operations. From initial user sign-up to post-purchase feedback, the system’s scope simplifies the process, ensuring a smooth and enjoyable interaction for the employees and the customers.

## References

Abraham Silberschatz, Henry F. Korth, S. Sudarshan, “Database System Concepts”, (6e), McGraw Hill Education (India) Edition, 2013.

Ramez Elmasri, Shamkant B. Navathe, “Fundamentals of Database Systems”, Addison-Wesley Publications, 2013.

# Overall Description

## Product Perspective

The EasyShop Management System is an innovative, standalone product designed to optimize supermarket operations. Unlike traditional replacements, it seamlessly integrates into existing systems while offering novel features. It can efficiently interface with broader retail management systems, ensuring real-time synchronization with inventory management and facilitating smooth transactions. This concise System Requirements Specification outlines its key components, subsystem interconnections, and external interfaces for the development of a cutting-edge supermarket management solution.

## Product Functions

**Employee**

* Transactions
* Customer feedback
* Customer registrations
* Calculate total order amount

**Manager**

* Access employee details
* Access employee working hours
* Stock Management
* Stock Updation
* Access sale records

## User Classes and Characteristics

The Supermarket Management System caters to two distinct user classes: Supermarket Employees and Managers. Supermarket Employees, encompassing a broad range of technical expertise, engage in daily operations and sales transactions, highlighting the need for user-friendly interfaces. Managers, serving higher-level roles, oversee employee activities, generate reports, and manage inventory. Their access to advanced features and comprehensive reporting makes them crucial for managerial decision-making, emphasizing their significance in optimizing supermarket operations.

## Operating Environment

The supermarket management software operates in a Windows environment, utilizing C# for the frontend and SQL Plus for the backend. The frontend, developed with C#, ensures a user-friendly interface, while the backend leverages SQL Plus for efficient data management. Compatible versions and network considerations ensure smooth operation.

## Design and Implementation Constraints

The Supermarket Management System development faces constraints such as regulatory compliance for data security and financial transactions. It must align with existing hardware specifications, seamlessly integrate with other applications, and adhere to specified technologies, tools, and databases. Robust security measures, design conventions, and maintenance responsibilities are pivotal considerations.

## User Documentation

User documentation is positioned as a critical element in the system's lifecycle. Comprehensive user guides and FAQs, coupled with an intuitive interface, empower users to navigate the system effortlessly. User documentation is not merely a post-development artifact; it is a proactive measure to enhance user satisfaction, reduce support requests, and create an environment where users feel confident and in control. Also, we will be using DBMS (Database Management Systems) lab manual to learn and implement things into our project.

## Assumptions and Dependencies

Assumptions crucial to the Supermarket Management System include the seamless integration of third-party components, stable development and operating environments, and accurate adherence to assumed project constraints. Any discrepancies or alterations in these assumptions may impact the project's requirements and overall success. The Supermarket Management System relies on a consistent and secure data storage and network infrastructure.

# External Interface Requirements

## User Interfaces

The user interface is a cornerstone of the system's success. Crafted using C#, it is not just a visual layer but an interactive space that defines user interactions. Navigation menus, input forms, and feedback mechanisms are designed with user experience at the forefront. The interface is not static; it is a dynamic canvas that evolves with user needs and technological advancements, providing a delightful and immersive user experience.

## Hardware Interfaces

* Barcode scanner
* Fingerprint scanner
* Computer
* Central server
* Wi-Fi
* Printer

## Software Interfaces

Software interfaces form the backbone of system functionality. The SQL database management system, with its ability to store and retrieve data efficiently, is a critical software interface. Web browsers become the conduit through which users interact with the system, emphasizing the need for compatibility and responsiveness. Integration with third-party APIs introduces a layer of complexity, requiring seamless communication for secure and reliable payment processing.

## Communications Interfaces

The application connects to a central SQL server hosted on the same private Wi-Fi network. All systems connected to this Wi-Fi will be able to access the database using a front end.

# System Features

## Stock Management Page

4.1.1 Description and Priority

High Priority. This feature allows employees to see the items low in stock and demand of each item using its average sales so orders can be placed, and discounts can be applied accordingly.

4.1.2 Stimulus/Response Sequences

Stimulus 1: Employee or manager logs in

Response 1: A Dashboard is displayed which shows low stock items, their demand and their place in store

Stimulus 2: Employee clicks search and searches for an item.

Response 2: The items and their details are displayed

Stimulus 3: Employee clicks order

Response 3: The item is added to order database to be reviewed by manager

4.1.2 Functional Requirements

REQ-1: User Authentication.

* The system shall authenticate employees and managers using their login credentials.

REQ-2: Stock Management Dashboard -

* The dashboard shall display a list of low stock items, including the item name, current stock quantity, and their location in the store.
* Low stock items shall be determined based on a predefined threshold value.
* The dashboard shall also display the demand for each low stock item, calculated using the average sales data.

REQ-3: Item Search

* The system shall provide a search functionality to allow employees to search for specific items.
* The search shall support searching by item name, Barcode ID, or other relevant attributes.

REQ-4:  Item Details

* Upon clicking search the search, the system shall display detailed information about the items matching the search query.

REQ-5: Order Placement

* The system shall provide an "Order" button for each item on the item details page.
* When an employee clicks the "Order" button, the system shall add the item to an order database for review by the manager.

## Checkout Page

4.2.1 **Description and Priority**

High Priority. This feature allows employees to easily checkout by entering the barcode number or receiving barcode using a barcode scanner. The Page displays the costs of the items bought and automatically applies discount. The final cost is displayed. The employee can ask the user for their phone number to ask them for feedback and link the purchase to the user to keep a track of user purchases

4.2.1 **Stimulus/Response Sequence**

Stimulus 1: Employee enters the barcode

Response 1: Item is added to the bill and the total is updated

Stimulus 2: The employee clicks + or – next to item

Response 2: Item count is updated

Stimulus 2: Employee clicks submit

Response 2: Final bill is updated the page asks for user phone number, method of

payment and feedback

4.1.2 **Functional Requirements**

REQ-1: Barcode Entering

* The system shall provide an input field for employees to manually enter the barcode number of an item.

REQ-2: Item Addition to Bill

* When a valid barcode number is entered or scanned, the system shall add the corresponding item to the bill.
* The system shall display the item name, quantity, and price on the checkout page.
* The system shall update the total cost of the bill in real-time as items are added.

REQ-3: Item Quantity Adjustment

* The system shall provide "+" and "-" buttons next to each item on the bill to allow employees to adjust the quantity of the item.
* If the quantity of an item is reduced to zero, the system shall remove the item from the bill.

REQ-4: Automatic Discount Application

* The system shall automatically apply any applicable discounts to the items on the bill based on predefined discount rules.
* The system shall display the discounted price and the original price for each item on the bill.

REQ-5: Final Cost Calculation

* The system shall calculate the final cost of the bill by summing up the discounted prices of all items.
* The final cost shall be displayed prominently on the checkout page and can be printed

REQ-6: User Phone Number Input

* After the employee clicks the "Submit" button, the system shall prompt for the user's phone number.
* The system shall validate the entered phone number format and display an error message if the format is invalid.
* The entered phone number shall be linked to the purchase transaction for future reference and analysis.

REQ-7: Payment Method Selection

* The system shall provide options for the employee to select the payment method, such as cash, credit card, or mobile payment.

REQ-8: Feedback Collection

* After the payment is processed successfully, the system shall prompt the user for feedback on their purchase experience.
* The feedback collection interface shall include predefined rating scales or open-ended text fields for the user to provide their feedback.
* The collected feedback shall be stored and associated with the purchase transaction for future analysis and improvement.

**4.3 Management Page**

4.3.1 Description and Priority

Medium priority. A page for the manger to look at employee details like leaves, check in

, check out time, sales made and orders that need to be made

4.3.1 Stimulus/Response Sequence

Stimulus 1: Manager enters their log in detail and clicks submit

Response 1: If details are correct a dashboard is opened else an error message is displayed

Stimulus 2: Manager clicks on any of the employee

Response 2: Employee details are displayed

Stimulus 3: Manager clicks on orders

Response 3: Manager clicks on order panel

Stimulus 3: All order requests are visible to employee and the employee who made them

4.3.2 Functional Requirements

REQ 1: User Authentication

* The system shall authenticate managers using their login credentials before granting access to the Management Page.
* The system shall securely store and validate manager login information.

REQ 2: Employee Details

* The system shall provide a comprehensive view of each employee's details, including personal information, leave history, check-in and check-out times, and sales performance.
* The system shall allow managers to view and update employee information as needed.

REQ 3: Attendance Tracking

* The system shall record and display employee check-in and check-out times.
* The system shall calculate and present attendance metrics, such as total hours worked and any discrepancies in attendance.

REQ 4: Sales Performance

* The system shall track and display sales data for each employee, including the number of sales made, revenue generated, and any relevant performance metrics.
* The system shall provide visualizations and reports to help managers analyze sales performance trends and identify top-performing employees.

REQ 5: Order Management

* The system shall allow managers to view and manage order requests submitted by employees.
* The system shall display the details of each order request, including the requested items, quantities, and the employee who made the request.

# Other Nonfunctional Requirements

## Performance Requirements

5.1.1 Response Time

Performance Requirement: The system shall respond to user interactions within 1 second under normal load conditions.

Rationale: To ensure a responsive and efficient user experience, meeting user expectations.

5.1.2 Concurrent Users

Performance Requirement: The system shall support at least 30 concurrent users without significant

performance degradation.

Rationale: To accommodate peak usage scenarios and provide scalability.

5.1.3 Transaction Throughput

Performance Requirement: The system should support a minimum of 200 transactions per minute during

peak hours.

Rationale: With a high number of transactions, especially during peak hours, the system must handle a substantial workload efficiently to prevent bottlenecks and ensure a responsive experience for users.

## Safety Requirements

5.2.1 Error Handling

Safety Requirement: The system shall provide clear error messages to users in case of invalid inputs or system errors.

Rationale: Enhance user understanding and prevent unintended actions that may lead to data loss or

system misuse.

5.2.2 Backup and Recovery

Safety Requirement: Regular automated backups of user data shall be performed, and the system should support efficient data recovery processes.

Rationale: Mitigate the risk of data loss due to system failures or other unforeseen events.

5.2.3 Transaction Rollback in Case of Errors

Safety Requirement: In the event of transaction errors or system failures, the system must automatically rollback incomplete transactions to maintain data consistency.

Rationale: Implement proper error-handling mechanisms and ensure that the system can recover gracefully from unexpected failures without compromising data integrity.

## Security Requirements

5.3.1 User Authentication

Security Requirement: User authentication mechanisms to ensure access only to authorized users.

Rationale: Ensure the confidentiality of user data and protect against unauthorized access.

5.3.2 Data Encryption

Security Requirement: All sensitive user data stored in the database shall be encrypted using industry-standard protocols.

Rationale: Protect user privacy and prevent unauthorized interception of sensitive information.

5.3.3 Access Control

Security Requirement: The system shall implement role-based access control to restrict user access to specific functionalities based on their roles.

Rationale: Ensure that users have appropriate access levels, reducing the risk of unauthorized

actions.

## Software Quality Attributes

5.4.1 Usability

Quality Attribute: The system shall achieve a usability rating of at least 85% in user satisfaction surveys.

Rationale: To ensure an intuitive and user-friendly interface, enhancing the overall user experience.

5.4.2 Reliability

Quality Attribute: The system shall have a meantime between failures (MTBF) of at least 500 hours (about 3 weeks).

Rationale: To minimize disruptions and downtime, providing a reliable service to users.

5.4.3 Maintainability

Quality Attribute: The system's codebase shall follow industry best practices for maintainability, with

a maintainability index of at least 80.

Rationale: To facilitate ongoing development and ease of system maintenance.

5.4.4 Adaptability

Quality Attribute: The system shall be adaptable to changes in business rules within a maximum of two weeks.

Rationale: To support business agility and the ability to respond quickly to changing requirements.

## Business Rules

5.5.1 Administrator Access:

Principle: Only authorized administrators have the ability to configure system settings, manage user roles, and perform administrative tasks such as backup, recovery and inventory management.

5.5.2 Cashier Functions:

Principle: Cashiers have access to the transaction interface to process customer transactions, apply discounts, and generate receipts. They do not have access to sensitive administrative functions.

5.5.3 Customer Access:

Principle: Customers have access to their personal account information, purchase history, and loyalty points. They can view product details and place orders but do not have administrative privileges.

# Other Requirements

Appendix A: Glossary

*TCP – Transfer Control Protocol*

*IP – Internet Protocol*

*SRS: Software Requirements Specification - A document that outlines the functional and non-functional requirements of a software system.*

*SQL: Structured Query Language - A standard programming language used for managing and manipulating relational databases.*

*SQL Plus: An Oracle Database tool that allows users to interact with the Oracle Database by entering SQL commands.*

*Windows Forms: A graphical user interface (GUI) framework for creating desktop applications on the Microsoft Windows operating system.*

Appendix B: Analysis Models

*Entity-Relationship diagram submitted*

Appendix C: To Be Determined List

• Implementation of membership program

• Working of the payment interface