Software Requirement Specification

for

Supermarket Automation Software (SAS)

by

Group-18

Devendra Palod (*20CS10024*)

Subhajyoti Halder (*20CS10064*)

Deepiha S. (*20CS30015*)

Department of Computer Science and Engineering

Indian Institute of Technology Kharagpur

Spring Semester, 2021-22

Contents

Chapter 1

Introduction

The Software Requirements Specification (SRS) document specifies both the functional and the non-functional requirements for the Supermarket Automation Software (SAS).

1.1 Purpose

The Supermarket Automation Software (SAS) is aimed at the automation of various processes that happens in a supermarket, like generation of bills after sales transactions, maintenance of inventory of various items, review sales statistics, keeping product details up-to-date, etc, with various access levels depending on job designation.

1.2 Document Conventions

- Document Format: Report
- Font typefaces: Computer Modern Roman (cmr), Times (ptm).
- Font sizes: default=10pt
- Page size: A4 (210 x 297mm)
- Margins: top=30mm, left=30mm

1.3 Intended Audience and Reading Suggestions

This document is intended for software developers, supermarket employees (like, managers, sales clerks, etc.), testers, documentation writers, etc.

This SRS contains the overall structure of the SAS as well as its hardware and software requirements for its smooth functioning. The reader must have basic knowledge of *Object Oriented Programming in Python*, *Django Framework* (for Backend), *HTML, CSS, Javascript, ReactJS* (for Frontend) and *Database Management*.

1.4 Product Scope

Supermarket Automation Software (SAS) is aimed at the automation of various processes that happens in a modern-day supermarket.

- SAS at the end of every sales transaction prints the bill containing the serial number of the sales transaction, the name of the product item, product id, quantity, unit price, and item price. The bill will also indicate the total amount payable.

- SAS also supports inventory management, the inventory of a product item should be decreased whenever a product item is sold. SAS will also support an option by which an employee can update the inventory whenever a new supply arrives.
- The manager upon query will also be able to see the inventory details and change the price at which a product item is sold as the prices of the different items vary on a day-to-day basis.
- SAS supports this process of updating the price, the software itself will create sales statistics that will indicate the quantity of a product item sold, the price realized, and the profit made over any particular day or any particular period.

1.5 References

The basic outline of the SRS document was provided.

Chapter 2

Overall Description

2.1 Product Perspective

This application automates various processes of a supermarket, all while reducing human efforts and eliminating the chances of human error. This also decreases the customer wait time, thus enriching the customer experience. SAS makes it easy to maintain track of transactions and inventory records, as well as providing detailed statistics on previous transactions, which can help in keeping track of the supermarket's sales and developing new sales promotion techniques.

2.2 Product Functions

The function of SAS is automating various processes that occurs daily in a supermarket.

- **Multiple Logins**: It supports manager login, employee login and sales clerk login where each login is password protected and can support multiple users.
- **Maintenance of the Inventory**: It supports inventory management by updating it whenever it's sold by the sales clerk during billing. The employee can update the inventory when new supplies arrive and the manager can change the price of the items on a day to day basis based on sales statistics and market prices.
- **Sales statistics**: It supports printing of the sales statistics for every item the supermarket deals with for any particular period of time. The sales statistics also indicate the quantity of an item sold, the price realised, and the profit thus facilitating supermarket business strategies.
- **Printing of the bill**: At the end of every transaction, it prints a bill containing the order number, names of items sold with their corresponding product-ids, quantity, unit price and item price,transaction id, and the grand total.

2.3 User Classes and Characteristics

There are three kinds of users who will be using the Supermarket Automation Software (SAS).

- **Sales Clerk**: They carry out the transaction with the customers by creating and printing bills for the transactions while updating the inventory.
- **Employee**: They oversee the supermarket and add & update the quantity of various product items in the inventory according to arrival of new supplies.
- **Manager**: They review the supermarket sales through the statistics developed by the SAS based on transactions and change the price of the items sold if required for better profits.

2.4 Operating Environment

The operating environment for the Supermarket Automation Software is listed below:

- Operating system: Windows, Linux and MacOS.
- Platform: Any machine that supports HTML5 supported web browser.
- Database: SQL
- hardware requirements:
  - bar-code reader
  - weighing machine
  - printer for printing bills

2.5 Design and Implementation Constraints

The constraints on the project is provision of the database resources. The more robust and fast the database, the better the performance of the software, also the storage of sales statistics should be done in a secure environment. On the hardware side , the bar-code reader and the weighing machine should be compatible in order to operate smoothly.

2.6 User Documentation

A user's manual would be handy for quick references to the various features that the software has. But a brief description and hands on tutorial would be sufficient for understanding the workings of the Supermarket Automation Software.

2.7 Assumptions and Dependencies

The users are supposed to know the basics of using computers and be comfortable with the English Language, since the whole software is implemented in the English Language. It is assumed that a bar code reader and an automatic weighing scale is provided to the sales clerk without which completing the sales transaction would be very difficult. The software also requires a printer to print the bills after each transaction and sales statistics.

Chapter 3

External Interface Requirements

3.1 User Interfaces

Manager Interface:

- View & update product price.
- View and plot sales statistics.

Employee Interface:

- View & update product Stock.
- Add new product to inventory.

SalesClerk Interface:

- Complete a customer transaction & update inventory.
- Generate bill on demand.

3.2 Hardware Interfaces

- **scanner**: scans the bar-code of the product and send the product-id to the software.
- **weighing machine**: weighs the product and sends it to the software.
- **printer**: print the bills generated at the end of each sales transaction.

3.3 Software Interfaces

Inventory Query:

The manager can raise a query to see the details of any product and the software can provide the details of the product. The manager is now able to update the price of the product in the database based on sales statistics.

Add To Inventory:

The employee can update the quantity of the product in the database which is already there in the inventory. The employee can also add new products to the inventory.

New Transaction:

The sales clerk enters the details of the product in the bill and also adds the quantity of the product. On clicking the print button the bill is generated. The inventory is updated automatically after every successful purchase. Now the sales clerk is ready for printing another bill.

Sales Statistics:

The Manager can view the statistics related to previous sales made, quantity sold, price realised, and net profit for a single day, or for any particular period of time.

3.4 Communication Interfaces

Any changes made in the inventory by employees or in the product details by the manager will automatically be updated. Bills are stored and used for generating sales statistics. All the logins for different user profiles will also be password protected.

Chapter 4

System Features

4.1 Maintaining Inventory

4.1.1 Description and Priority

The Supermarket Automation System maintains the inventory of all the products that are sold. This feature is a high priority because, without an inventory, a supermarket cannot function.

4.1.2 Stimulus/Response Sequences

The manager can access the inventory and increase or decreases the price of the products according to sales. The employee can update the quantity of items or add a new product upon arrival of new stock.

4.1.3 Functional Requirements

Django: If no database is present, then the inventory cannot be handled. Thus it is necessary to have a database at hand. Django implements the database internally by using SQLite.
Python: The programming language required to run the application

REQ-1: Django req

REQ-2: Python req

4.2 Manager, Employee and Sales Clerk Login

4.2.1 Description and Priority

The Supermarket Automation System handles logins for Managers, Employees and Sales Clerks such that they can operate without any interference from each other. Manager and Employee Login allows the user to access the database and change the price and quantity of any item respectively. Sales Clerk Login allows the user to access the database and sell any product. This feature is a high priority because, without it, the database cannot be managed during the runtime of the software.

4.2.2 Stimulus/Response Sequences

The manager, employee and sales clerk can use their respective login for using the software.

4.2.3 Functional Requirements

Django: Without Django, we cannot use the Login feature, which is used for Logging in a user.
Python: The programming language required to run the application

REQ-1: Django required REQ-2: Python required

4.3 Plot Support for Sales Statistics

4.3.1 Description and Priority

The Supermarket Automation System provides the ability to view graphs that plot out the sales statistics of the supermarket for any particular day or period. The sales statistics indicate the quantity of an item sold, the price realised and the profit made. Since this is just for visualisation purposes, it is a low priority but it can be very useful and important too from a business point of view.

4.3.2 Stimulus/Response Sequences

The manager can decide the time frame for which the statistics are to be plotted. Upon clicking the plot button, a graph comes up that shows the desired statistics.

4.3.3 Functional Requirements

Django: If no database is present, then the inventory cannot be handled. Thus it is necessary to have a database at hand. Django implements the database internally by using SQLite.
Matplotlib: Matplotlib is required to build graphs
Python: The programming language required to run the application
REQ-1: Django required
REQ-2: Python required
REQ-3: Matplotlib required

Chapter 5

Other Non-functional Requirements

5.1 Performance Requirements

- High-speed network.
- High-level connectivity.
- Server reliability.
- Handle the queries properly.

5.2 Safety Requirements

The users must remember their login credentials, as the feature for resetting the credentials has not been implemented yet.

5.3 Security Requirements

The passwords are encrypted for protecting the user's information. Since there are separate logins for managers, employees and sales clerk, only managers and employees can modify the inventory of the supermarket. No sales clerk is allowed to modify the price of the items.

5.4 Software Quality Attributes

- Every user of the software whether that is a manager, employee or sales clerk has been given a unique ID and password which is stored in the database.
- The software is available for the working hours of the supermarket.
- The software has a very good user interface which provides easy to use and accessible software to the users.
- He software is easy to navigate and smooth.
- The software should handle the load and should not crash easily.

5.5 Business Rules

The SAS can be used by managers, employees and sales clerks of the supermarket to which the software is sold.

Chapter 6

Other Requirements

Since the database stores a lot of data in it, like the sales transaction and the items in the inventory, it is suggested that the platform on which this software is to be deployed must have sufficient reserve memory in it.

6.1 Appendix A: Analysis Models

6.1.1 Use Case diagram

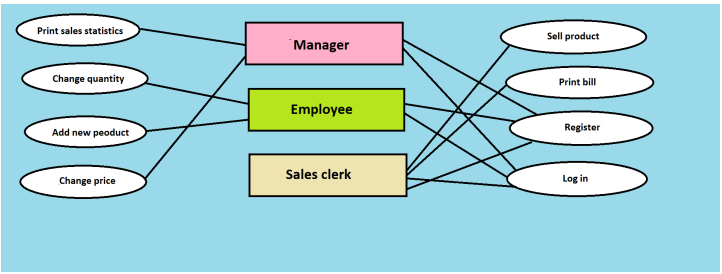

Figure 6.1

6.1.2 Class Diagram

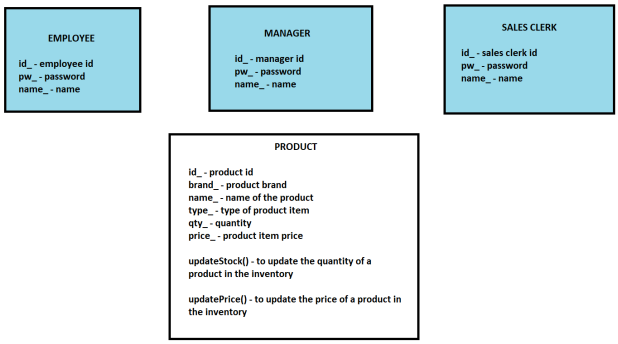

Figure 6.2

The class diagram is the one of the most important part of object oriented modelling. It can be used for both conceptual as well as systematic application for detailed modelling translating the models into programming code. Class represent both the main elements,interactions in the application and the classes to be programmed.

There are 4 classes in the project -

- employee
- manager
- salesClerk
- product

6.2 Appendix B: To Be Determined List