# Software Requirements Specification

for

# Medical Shop Automation

**Version 1.0 approved**

**Prepared by:**

1. **Gaurav Malakar**
2. **Manami Mondal**
3. **Prakhar Singh**

**CSE, IIT Kharagpur**

**Team: Zenith**

**Group: 34**

**15th March, 2022**

# Table of Contents

## Revision History

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

# 1. Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, document conventions, intended audience and reading suggestions, product scope and references of the SRS. The aim of this document is to gather and analyze and give an in-depth insight into the complete **Medical Shop Automation (MSA) software system** by defining the problem statement in detail. The detailed requirements of the **Medical Shop Automation** are provided in this document.

## 1.1 Purpose

The purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, and software requirements. It defines how our client, team and audience see the product and its functionality. Hence, the purpose, in short, is to build an online system for a retail medicine shop that deals with a large number of medicines procured from various manufacturers. This therefore eases the workload of medical shop professionals.

## 1.2 Document Conventions

The standard font used throughout the document is Arial, with font size 12. The Titles of the various sections of this SRS document have been represented in bold, with font size 18 and font Arial. Some Sub-Headings are with font size 14. Important parts of the document have been indicated in bold.

## 1.3 Intended Audience and Reading Suggestions

This project is a prototype for medical shop automation. Hence, this project is useful for the whole general public in general and the patients to be specific. This project also indirectly benefits the medical shop

owners to keep track of records of the medicines available providing a smooth flow of supply.

Thus, this document is intended to assist the users when they use the software and for developers and the project managers to plan their project and implement the software required. This Software Requirement Specification document is divided into five subsections.

Section 1: Introduction

Section 2: Overall Description of the Software

Section 3: External Interface Requirements

Section 4: System features.

Section 5: Other non-functional requirements

Section 6: Other requirements

### 1.4 Product Scope

Primarily, the scope pertains to the automated medical shop product features for making this project live. It focuses on the medical shop and the automation of the maintenance of the list of different medicines procured from various manufacturers in wall mounted and numbered racks, thereby easing the flow and maintenance of records of medicines available and making it time efficient.

The standard can be used to create software requirements specifications directly or can be used as a model for defining an organization or project specific standard.

### 1.5 References

The basic outline of an SRS document has been given to us. We also took references from the following websites for better understanding:

1) https://www.freeprojectz.com/projects/medical-shop-automation-system
2) https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document

## 2. Overall Description

### 2.1 Product Perspective

This application is useful for the medical shops for automating its daily tasks and freeing up the medical shop professionals from the mundane and repetitive task as well as eliminating human error. This in turn increases productivity and eases in keeping track of records of medicines available, by allowing the shop owner to maintain as few inventories for each item as reasonable, to reduce inventory overheads after being inspired by the just-in-time (JIT) philosophy. Hence, the management becomes more efficient and effective.

## 2.2 Product Functions

The following features are proposed for this application:

i) Registration and Log-in feature for users.
ii) Feature to calculate average sale for a certain period.
iii) Feature to calculate total revenue and profit over a period of time.
iv) Feature to print cash receipt and cheques for sale and supply of medicines.
v) Feature to search for available medicines by name
vi) Feature to notify user about expired and required medicines at EOD.

## 2.3 User Classes and Characteristics

There is only one type of role:

Medicine shop owner --- Responsible for maintaining different medicines in wall mounted and numbered racks and also an appropriate vendor list for replacing the expired medicines and also providing for a cheque favouring the vendor for the items supplied (using the software)

## 2.4 Operating Environment

Application will be deployed as a web app:

- Operating system: Windows / Linux / MacOS
- Frontend: HTML, CSS, JavaScript
- Backend: Django
- Database: SQL

## 2.5 Design and Implementation Constraints

Design is decoupled frontend (in HTML, CSS, JavaScript) and backend (in Django). Implementation constraint would be the robustness / performance on Database side.

The other constraints are:

- Cannot be implemented as an Android/IOS App
- Doesn't work on outdated/slow servers
- No other languages except English is used in this implementation

### 2.6 User Documentation

License and User Manual will be provided along with the software.

### 2.7 Assumptions and Dependencies

• The user is supposed to know the basics of computer handling

• The user is supposed to know and operate in English language as the website is implemented in English language

## 3. External Interface Requirements

### 3.1 User Interfaces

i) Register and Sign-in facility for users.

ii) GUI along with buttons and frames for all the required tasks

iii) Generates cheques for medicine purchase.

iv) Generates cash receipts for medicine sale.

v) Whenever new medicines are bought by the owner it stores the medicine details and generates a code number for the medicine.

vi) Calculate average medicine sale per week.

vii) Generates list of expired medicines at the end of each day.

viii) Generates list of medicine required to be ordered at the end of each day.

### 3.2 Hardware Interfaces

There is no special hardware interface requirement

### 3.3 Software Interfaces

i) Front End – HTML, CSS, JavaScript

ii) Back End – Django

iii) Database – SQL

### 3.4 Communications Interfaces

1. Login will be managed by password encryption in the database.

2. New users have to register first and then log in whereas old users can log-in directly.

3. The owner will get notifications regarding the expired medicines at EOD.

4. The owner will get notifications regarding medicines to be ordered at EOD.

## 4. System Features

### 4.1 Login and Registration page

#### 4.1.1 Description and Priority

This will be the first page of our website where new users can register and the old users can login with their password. Passwords stored in the database will be encrypted using a hashing algorithm.

#### 4.1.2 Stimulus/Response Sequences

i) After registration new users will be asked to go back to the login page and login from there.

ii) After Log-in the users will be redirected to the next page which stores most of the system features along with their previously stored details.

#### 4.1.3 Functional Requirements

Whole login will be extensively managed by Django and its dependencies hence Python and Django will be required.

## 4.2 Cash Receipt Printing

### 4.2.1 Description and Priority

The website has feature to print cash receipt. Whenever the user sells medicine to any customer, he should enter the order details and a suitable cash receipt containing the details of the transaction will be printed.

### 4.2.2 Stimulus/Response Sequences

i) After pressing the sell button, the user is redirected to a page and asked to enter the order details and finally he should press the confirm order button.

ii) After pressing confirm order button, a pdf containing the cash receipt is created and displayed in a new tab.

### 4.2.3 Functional Requirements

This needs a database for managing the data of requests and also a simple algorithm to calculate the order total. So, a database management system is needed which will be implemented with Django. Required: PostgreSQL database

## 4.3 Average

Average feature calculates the average medicine sale for one week. It also allows us to manually enter the start and end date for getting average sale between them.

## 4.4 New Med

### 4.4.1 Description and Priority

A feature to store the details of any medicine whose details were not entered earlier in the system. A button containing this feature will be available on the second page of the website.

### 4.4.2 Stimulus/Response Sequences

i) After pressing the button, the user will be redirected to a page where he can enter the details of the medicine.

ii) After pressing confirm button, a code number will be generated for the medicine and displayed on the screen.

#### 4.4.3 Functional Requirements

This needs a database for storing the medicine details and also a simple algorithm to generate a unique code for the new medicine.

### 4.5 Supply

Whenever medicines are supplied to the shop from a vendor this feature is used to store the details of the supply. The user needs to enter all relevant details regarding the medicines in the supply and the vendor who made this supply. All these details are then stored in the database for future use.

## 5. Other Non-functional Requirements

### 5.1 Performance Requirements

For proper loading and smooth functioning of the website, user needs to have a high-speed reliable internet connection. For opening the website, any recent version of a browser would be fine.

### 5.2 Safety Requirements

The user should remember his login credentials and they should not share it with anyone else.

### 5.3 Security Requirements

For accessing the data, user id and password is required and the data for each user is secured in portal by the password.

### 5.4 Software Quality Attributes

#### 5.4.1 Reliability

The website can be used by multiple users. It will be deployed only after rigorous testing of its features to ensure zero bugs in the implementation.

#### 5.4.2 Maintenance

The website can be upgraded from time to time by the admin and new features can be added.

#### 5.4.3 Portability

The website can be accessed from any place that satisfies the minimal requirements.

#### 5.4.4 Availability

The MSA website would be available 24 x 7.

### 5.5 Business Rules

This website can be used after proper contract agreement with the company developing it.

If any issues are faced by the user, admin should be contacted.

# 6. Other Requirements

Licencing and copyright are required. Database will be required to be hosted in an online platform

# Appendix A:

## Glossary

- SQL: Structured Query Language
- HTML: Hyper Text Markup Language
- CSS: Cascading Style Sheets
- JS: JavaScript
- EOD: End of Day

# Appendix B:

## Analysis Models

Appendix second content.

# Appendix C:

To Be Determined List