

---

# **Software Requirements Specification**

for

## **Medicine Shop Automation Software**

**Version 1.0 approved**

**Prepared by Abhishek Kumar Sah(19CS10004), Hritaban  
Ghosh(19CS30053), and Saumyak Raj(19CS30040)**

**Code Cannibals I.I.T. KHARAGPUR**

**20th March 2020**

# Table of Contents

## Table of Contents

<b>1. Introduction</b>	<b>3</b>
1.1 Purpose	3
1.2 Document Conventions	3
1.3 Intended Audience and Reading Suggestions	3
1.4 Product Scope	3
1.5 References	3
<b>2. Overall Description</b>	<b>4</b>
2.1 Product Perspective	4
2.2 Product Functions	4
2.3 User Classes and Characteristics	4
2.4 Operating Environment	5
2.5 Design and Implementation Constraints	5
2.6 Assumptions and Dependencies	5
<b>3. External Interface Requirements</b>	<b>5</b>
3.1 User Interfaces	5
3.2 Hardware Interfaces	6
3.3 Software Interfaces	6
<b>4. System Features</b>	<b>6</b>
4.1 Shop Owner Registration and Login	6
4.2 Add Vendor Information	7
4.3 New Supply	7
4.4 Search for Medicine	9
4.5 Calculate and Update Threshold Quantities For Medicine	10
4.6 Generate List Of Expired Medicines	10
4.7 Sell Medicines	11
4.8 Calculate Revenue and Profit	11
4.9 Vendor wise payment	12
4.10 Generate List of Below Threshold Medicine	12
<b>5. Other Nonfunctional Requirements</b>	<b>13</b>
5.1 Performance Requirements	13
5.3 Security Requirements	13
5.4 Software Quality Attributes	13
<b>6. Other Requirements</b>	<b>13</b>
<b>Appendix: Analysis Models</b>	<b>14</b>
<b>Use Case Diagram For MSA</b>	<b>14</b>
<b>Class Diagram For MSA</b>	<b>15</b>
<b>Entity Relationship Diagram For MSA</b>	<b>16</b>

# 1. Introduction

## 1.1 Purpose

*The purpose of this document is to specify the software requirements for the Medicine Shop Automation Software. This software aims to reduce the tremendous amount of labor involved on the part of the shop owner by automating the management of the Medicine Shop and its inventories. This product covers all aspects of the procedures involved in a medicine shop.*

*Release Number: 1.0*

## 1.2 Document Conventions

*This document follows the standard IEEE convention throughout.*

## 1.3 Intended Audience and Reading Suggestions

*This project is a prototype for the medicine shop automation system and it is restricted within the college premises. This has been implemented under the guidance of college professors and teaching assistants. This software is useful for the shop owner of a medicine shop.*

## 1.4 Product Scope

*The purpose of the Medicine Shop Automation Software is to automate the management of a Medicine Shop and to ease the workload on the shop owner. It is a convenient and easy-to-use application for the shop owner who has to deal with the management of the medicine shop's inventories and undergo various types of procedures involving transactions with vendors of the medicines and customers who buy the medicines. The system is based on a relational database with its Medicine Inventory, and Transaction Records. It can handle efficient storage of data and provides the user with various functionalities. Above all the software aims to provide a comfortable user experience to the medicine shop owners.*

## 1.5 References

1. Rajib Mall "Fundamentals of Software Engineering"
2. [How to Draw UML Use Case Diagram - Determine Subject Boundaries, Define Actors, Describe Use Cases.](#)
3. <https://en.wikipedia.org/wiki/SRS>

## 2. Overall Description

### 2.1 Product Perspective

*The Medicine Shop Automation Software is developed to facilitate the process of automation in the context of a medicine shop. This software will automate several processes that if done manually takes a tremendous amount of effort and time. It helps manage a huge inventory of medicines, helps keep track of transactions both from the vendor end and the customer end. It also follows in steps with the Just-In-Time Philosophy and suggests the shop owner the right amount of medicines to be stored in the inventory. It also keeps track of the medicines which have expired and those that are low in quantity and provides appropriate reports to the shop owner so that he can re-order and replace the medicines in the inventory. Automation of such processes which otherwise would require a humongous amount of labor is a major motivator for the Product. The Product intends to ease the workload of the shop owner and help him in smoother operation of his/her shop.*

### 2.2 Product Functions

*Functions that can be performed by the software:*

- *The shop owner can query about any medicine at any time using the search bar by entering its generic name or trade name. The software will in turn provide the owner with the medicine ID and the quantity of the medicine present in the store.*
- *At the time of every sale, the shop owner enters the medicine code along with the quantity to be sold, and the software prints a cash receipt for the same.*
- *When new supplies arrive, the shop owner would enter the medicine code, quantity, batch number, expiry date, and vendor number. The software will print a cheque favoring the vendor for the items he had supplied.*
- *Whenever a new kind of medicine arrives in the shop the owner will enter the medicine details, vendor details, and price details, and the software will generate a printable medicine code for the owner which will be used to identify the rack where that will be kept.*
- *The software is also capable of providing the net revenue and profit for any entered period. It shows vendor-wise payments for the entered period.*
- *The software is capable of calculating and updating the threshold quantity for every medicine based on sales in the previous week.*
- *The software generates the list of medicines to be ordered which fall below the threshold at the end of every day, along with the address of the vendor, the quantity, and the medicine description.*
- *The software generates a list of expired medicines at the end of every day, along with the vendor-wise list, so that they can be purchased easily.*

### 2.3 User Classes and Characteristics

*There is just one User (Actor) for this software, which is the Owner of the shop. He will be able to maintain a database of medicines present and which have expired in his inventory, query about the medicines using the search bar, add new medicine to the inventory, prepare cash receipt after a sale, prepare a vendor-wise list of expired medicine in the inventory, generate*

medicines to be ordered, calculate and set the threshold of medicine for next week and calculate revenue.

## 2.4 Operating Environment

The software is a Python application that also makes use of a relational database. It is designed to work flawlessly and without issues on any major Operating Systems Windows 10/Linux/Mac OS with python 3.7 or more with required libraries (like SQLite and Django).

## 2.5 Design and Implementation Constraints

There will be four major constraints in developing this product:

- The user interface must be intuitive enough so that no training is required by customers, members, or store personnel.
- A good form of integration between the database and the application.
- Making real-time updates to the database and allowing access at the same time.
- Memory requirements should not be high, a limited amount of memory can cause issues if the database is too large.
- The software and database will be password protected so that no unauthorized person cannot access it.

## 2.6 Assumptions and Dependencies

The software will be made with the following assumptions:

- The users have computers with Windows 10/Linux/ Mac installed.
- Users should not tamper/experiment with the source code/executable file of the software.
- The user should have good knowledge about the basic attributes of an object and fill in the details in the forms properly.
- The software requires a printer(optional).

The main dependencies of the working and performance of the software are:

- The python and all other platforms should be functioning properly.
- All the tools on which the software is dependent must be working properly.
- The software will also depend on the database and the interaction of the python application with the database.

# 3. External Interface Requirements

## 3.1 User Interfaces

The home screen will be a login screen with the shop logo from where the owner will log in to the software.

There will be one header that will be displayed on all pages.

The header will contain one home button and different drop-down menus like Vendor Management, Medicine Management with options related to these headings. On the rightmost side of the header, there will be a profile button. Clicking the button will bring down a drop-down menu with the options to change login details and Log out. Choosing any feature from the drop-down menu will bring the required page with a form or list, whichever is required by the feature selected by the owner.

## 3.2 Hardware Interfaces

*A computer with a monitor, a keyboard, a mouse, a printer, and other common hardware peripherals. For printing lists, a hardware printer is used, but if it is not available it can be printed in pdf format also. Use of SSD (Solid State Drive) is recommended for fast retrieval of data from the database. Recommended requirements: 8GB RAM, with an i3 or more powerful processor*

## 3.3 Software Interfaces

*Language used: Python*

*Platform: Windows 10 20H2 / Linux / Mac*

*Tools: Django*

*Database: SQLite.*

*Browser: Latest version of chromium-based browsers like Google Chrome or Microsoft Edge*

# 4. System Features

## 4.1 Shop Owner Registration and Login

### 4.1.1 Description and Priority

*This feature requires prior registration and login by the shop owner. Registration of the shop owner requires details of his unique username, name, email address, and password to be set. A unique username and password can be used as authentication during the login. This feature registers the shop owner as an authorized user of the software and enables him to login in thereafter to access his Medicine Shop's Inventory. This feature ensures that no unauthorized person can access valuable shop information. Having this feature is essential from the point of view of security.*

### 4.1.2 Stimulus/Response Sequences

#### **Input:**

1. Ask for login or registration.
2. For a new registration, take in the details of the shop owner
  - Name,
  - Email
  - Chosen username (should be unique)
  - Chosen password.
3. For existing users, the username and password are checked with existing data and verified. If correct, proceed to the Medicine Shop Inventory, else prompt to type again.

### 4.1.3 Functional Requirements

- 1) *The Shop owner must log in with the unique username and password he/she had provided at the time of registration.*
- 2) *In case they forget their password, a mail to the verified email address of the user must be used to reset it.*

## 4.2 Add Vendor Information

### 4.2.1 Description and Priority

*This feature enables the shop owner to add vendor information into the Vendor Database of the system. The shop owner can add the vendor name, and basic contact details of the vendor such as vendor address, vendor phone number, and email address. The system automatically allocates the vendor a vendor ID which the shop owner uses to keep track of the medicines supplied by the vendor and to keep tabs on the payments made by him to the vendor on various transactions. This feature is highly essential for the system as it maintains integrity and enables the proper functioning of the database.*

### 4.2.2 Stimulus/Response Sequences

**Input:**

1. *The user taps on the Add Vendor button to enter details of the vendor into the system. The system takes in the following details of the vendor*
  - *Name,*
  - *Address*
  - *Phone Number*
  - *Email Address*

**Output:**

1. *The Vendor information has been appended into the vendor database.*

### 4.2.3 Functional Requirements

1. *The feature can be accessed only when the user is logged in to the system*

## 4.3 New Supply

### 4.3.a Add Medicine

#### 4.3.a.1 Description and Priority

*This feature enables the user to add medicines to his inventory. The feature requires vendor information associated with the medicine to be already present in the Vendor Database. Thus it has a vendor verification check to ensure that the vendor is in the shop's database. In the case of new medicine, it makes use of System Feature 4.3.b. The shop owner upon receiving a new supply at his doorstep enters the information of all the medicine items he received such as medicine ID, quantity, batch number, expiry date, and vendor ID. After all, information has been properly and correctly entered, System Feature 4.3.c is initiated by the click of a button.*

#### 4.3.a.2 Stimulus/Response Sequences

**Input:**

1. *The user has to press the New Supply button to enter the information of the new supply he has received. The system takes in the following details of the vendor*
  - *Medicine ID*
  - *Quantity*
  - *Batch number*
  - *Expiry Date*

- Vendor ID

**Output:**

1. This feature automatically updates the inventory database to store all the information just entered as input.

**4.3.a.3 Functional Requirements**

1. The feature can be accessed only when the user is logged in to the system
2. The feature requires valid Medicine ID and Vendor ID thus requiring the medicine database and vendor database to have the required information.

**4.3.b Add New Medicine****4.3.b.1 Description and Priority**

*This feature enables the user to add new medicines to his Medicine Database. It allows the user to add details of new medicine in case the trade name he entered while adding medicine was not found in the Medicine Database.*

**4.3.b.2 Stimulus/Response Sequences****Input:**

1. Additionally, if the trade name check failed and the medicine is new, the system will further ask us to add new medicine. There will be an option available to add new medicine which requires the following details
  - Medicine Trade Name
  - Generic Name
  - Vendor ID
  - Unit Selling Price
  - Purchase Price

**Output:**

1. This feature automatically updates the Medicine Database with the new medical information.

**4.3.b.3 Functional Requirements**

1. The feature can be accessed only when the user is logged in to the system
2. This feature presents itself in case trade name check fails thus is sub-part of the System Feature 4.3.a.

**4.3.c Print Code Number(Medicine ID) of the Medicine****4.3.c.1 Description and Priority**

*If the System Feature 4.3.b had been initiated then as a result that this system feature will be sequentially initiated. This feature allows the shop owner to print out a code of the medicine which he/she would paste in the rack where the medicine would be stored.*

**4.3.c.2 Stimulus/Response Sequences****Input:**

1. The user after entering all the information of the new medicine using System Feature 4.3.b has the option to print out the code of the medicine. He has to press the Print Code button to initiate the feature.

**Output:**



1. *This feature outputs a printable PDF containing the code number of the medicine.*

#### 4.3.c.3 Functional Requirements

1. *The feature can be accessed only when the user is logged in to the system*
2. *This feature presents itself after entering the details of the new medicine, thus is a sub-part of the System Feature 4.3.b.*

### 4.3.d Print Cheque

#### 4.3.d.1 Description and Priority

*After all, information has been entered, the software further generates a cheque favoring the vendor for all the items supplied and this part of the System Feature is executed after all the information of the supply has been entered.*

#### 4.3.d.2 Stimulus/Response Sequences

**Input:**

1. *The user after entering all the information of the supply using System Feature 4.3. a has the option to print out a cheque favoring the vendor for the items supplied. He has to press the Print Cheque button to initiate the feature.*

**Output:**

1. *This feature outputs a printable PDF listing the amount favored to all the vendors for the items they have supplied.*

#### 4.3.d.3 Functional Requirements

1. *The feature can be accessed only when the user is logged in to the system*
2. *This feature presents itself at the end of all entries of the new supply, thus is a sub-part of the System Feature 4.3.a.*

## 4.4 Search For Medicine

#### 4.4.1 Description and Priority

*This feature enables the user to search for medicines by their trade name or generic name and in turn, the system displays the medicine ID of the medicine and the quantity present. This feature is essential because the shop owner must be able to query about his inventory at his will without having to actually hand-count the items or search for them physically in his inventory.*

#### 4.4.2 Stimulus/Response Sequences

**Input:**

1. *The user has to type in the keyword by which he wishes to search in the search bar.*
2. *This feature internally makes use of the data stored in the Medicine Database.*

**Output:**

1. *This feature lists out the codes and quantities of the medicines stored in the Medicine Database that match with the keyword typed in by the user either by trade name or generic name.*

#### 4.4.3 Functional Requirements

1. The feature can be accessed only when the user is logged in to the system.

## 4.5 Calculate and Update Threshold Quantities for Medicines

### 4.5.1 Description and Priority

*This feature requires at least one Medicine Information to be stored in the Medicine database and at least one week of data in the Transaction Database. It calculates the threshold quantity of medicine based on the average number of sales made in the past week. It automatically updates the Medicine Database to the new threshold values. This is essential because it helps the shop owner keep up with the latest trends on the purchase of different medicines and helps boost his sales and avoids the purchase of overhead inventory. This feature is inspired by the Just-In-Time (JIT) Philosophy.*

### 4.5.2 Stimulus/Response Sequences

#### **Input:**

1. The user has to press the Calculate and Update Threshold button to activate and use this feature.
2. This feature internally makes use of the data stored in the Medicine Database and the Transactions Database.

#### **Output:**

2. This feature automatically updates all the threshold quantities of the medicines stored in the Medicine Database.

### 4.5.3 Functional Requirements

1. The feature can be accessed only when the user is logged in to the system
2. The feature requires at least one Medicine Information to be stored in the Medicine database and at least one week of data in the Transaction Database.

## 4.6 Generate List of Expired Medicines

### 4.6.1 Description and Priority

*This feature helps the owner to find a vendor-wise list of all expired medicines. The owner presses a button and a detailed list of expired medicines along with the vendors is generated. The shop owner presses this every day to ensure that expired medicines are never sold to the customers.*

### 4.6.2 Stimulus/Response Sequences

#### **Input:**

1. The user presses the expired medicine button and no other detail is required.
2. This feature makes use of the inventory and expired item database internally.

#### **Output:**

1. This feature generates a detailed list of expired medicines along with the respective vendors.

### 4.6.3 Functional Requirements

1. The feature can be accessed only when the user is logged in to the system

## 4.7 Sell Medicines

### 4.7.1 Description and Priority

*This feature helps the owner to sell his medicines. As soon as a customer purchases medicines, the owner makes an entry and the software will remove those medicines from the inventory. After this, a transaction list is updated with this current transaction, and the cash receipt is printed. Also, if enough medicine is not available, the software will inform the shop owner about the same.*

### 4.7.2 Stimulus/Response Sequences

#### **Input:**

1. The user presses the sell medicine button and enters details of the customer( to be used later to print the transaction receipt). He has to enter the corresponding medicine id and quantity to be ordered.
2. This then uses the data stored in the inventory database for price calculations.

#### **Output:**

1. This feature updates the inventory database by deleting the quantities of the respective medicines, then it updates the transaction database by adding the respective transactions. Finally, a receipt is printed for giving the customer
2. In case the required medicine is not present in sufficient quantity, the owner is alerted.

### 4.7.3 Functional Requirements

1. The feature can be accessed only when the user is logged in to the system.

## 4.8 Calculate Revenue and Profit

### 4.8.1 Description and Priority

*This feature helps the owner to calculate his net profit and net expenditure in a given interval of time. The owner has to enter the start and end date for revenue calculation and he gets a detailed report of all medicines sold and bought along with the net profit and net expenditure over the period.*

### 4.8.2 Stimulus/Response Sequences

#### **Input:**

1. The user presses calculate revenue button and enter the start and end date for this calculation

#### **Output:**

1. He receives a detailed report of all medicines sold in the period along with their quantity and id, and similarly all medicines bought from the different vendors in the given time.
2. In the report, the net profit and net expenditure over the period are mentioned.

### 4.8.3 Functional Requirements

1. *The feature can be accessed only when the user is logged in to the system*

## 4.9 Vendor Wise Payment

### 4.9.1 Description and Priority

*This feature helps the owner to know about all his transactions with a particular vendor in a given interval of time. The owner has to enter the start and end date, and the respective vendor id and he gets a detailed report of all medicines bought from the vendor along with the net expenditure over the period.*

### 4.9.2 Stimulus/Response Sequences

**Input:**

1. *The user presses the vendor transaction button and enters the start and end date, and vendor id for this calculation*

**Output:**

1. *He receives a detailed report of all medicines bought in the period along with their quantity and id from that particular vendor.*
2. *In the report, the net expenditure over the period with the selected vendor is also mentioned.*

### 4.9.3 Functional Requirements

1. *The feature can be accessed only when the user is logged in to the system*
2. *This feature will work if the vendor id is correct i.e. if the selected vendor exists.*

## 4.10 Generate List of Below Threshold Medicines

### 4.10.1 Description and Priority

*This feature helps the owner to know if any medicine has gone below the threshold. The threshold limit is calculated every week based on the sales of the previous week. The user has to press this generate list button at the end of every day and the list is generated along with details of the vendor, medicine id, and the required quantity.*

### 4.10.2 Stimulus/Response Sequences

**Input:**

1. *The user only has to press the generate list of below threshold medicines, and no other details are required.*
2. *This feature internally uses the Medicine database and Inventory Database for its computations.*

**Output:**

1. *He receives a detailed report of all medicines below their respective threshold mark with vendor details, medicine id, and quantity required.*

### 4.10.3 Functional Requirements

1. *The feature can be accessed only when the user is logged in to the system*

## 5. Other Nonfunctional Requirements

### 5.1 Performance Requirements

*The software should perform smoothly and efficiently. The performance of the software will greatly depend on the processor and speed of accessing data from the database if a large amount of data is stored. The software uses a few computations that are not computationally heavy but are very much dependent on the database and processing and data handling power of the computer if huge data is stored.*

### 5.2 Security Requirements

*Security requirements ensure that the software is protected from unauthorized access to the system and its stored data. It considers owner authorization and authentication.*

### 5.3 Software Quality Attributes

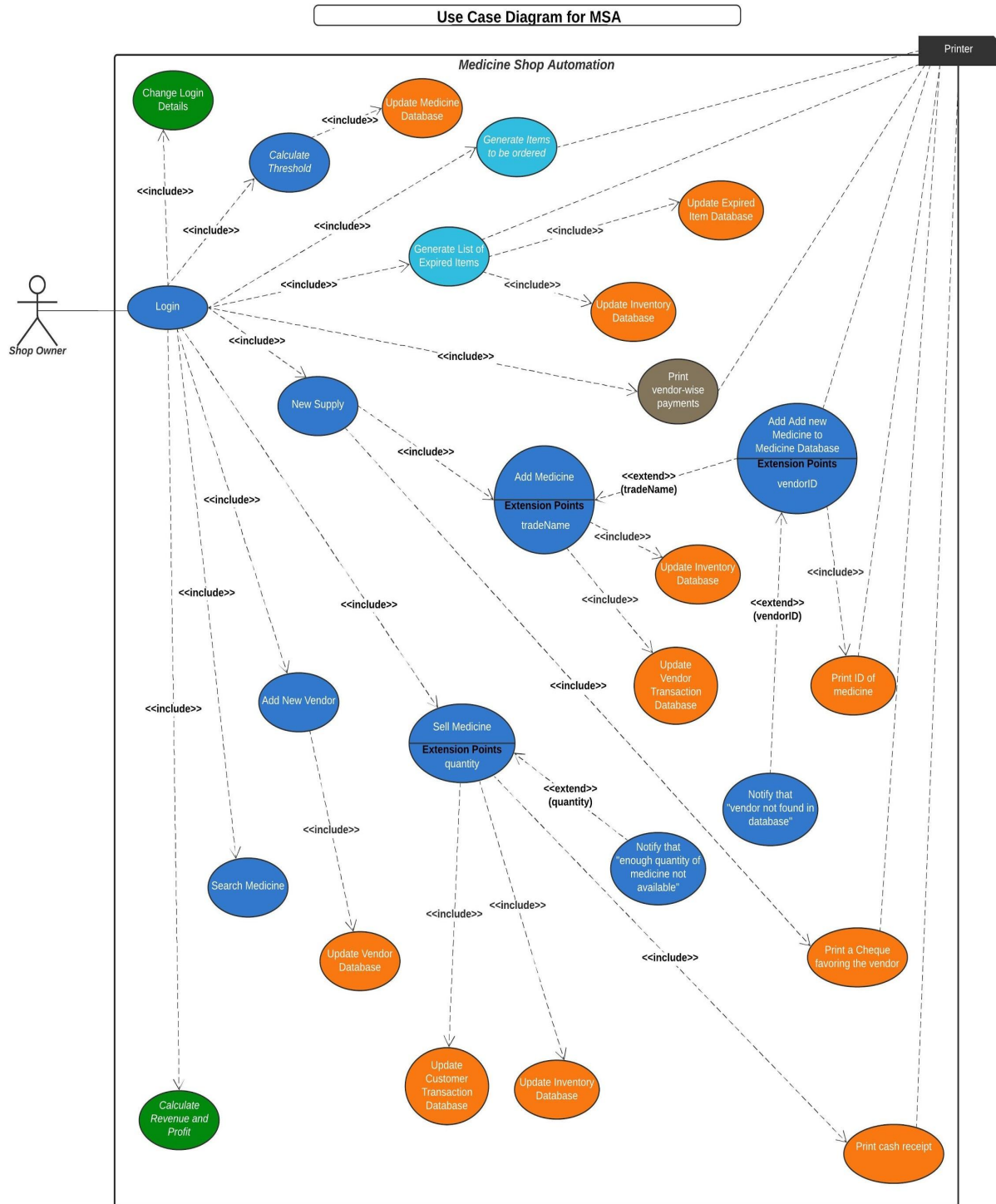
*The software must be easy to use and should run without issues in Windows. It should be correct, easily maintainable, and extensible. The software should also be reliable as it contains a lot of important user data. The software must also ensure that unauthorized persons cannot access the database. (so password authentication is included).*

## 6. Other Requirements

*The use of the software will be guided by the rights a user is provided. No legal issues must be there with the use of the software. However, the tools used here have some specific licenses. The license terms must be followed to avoid any legal issues in the future. A user manual will also assist the software so that users can get the best out of this software.*

## Appendix: Analysis Models

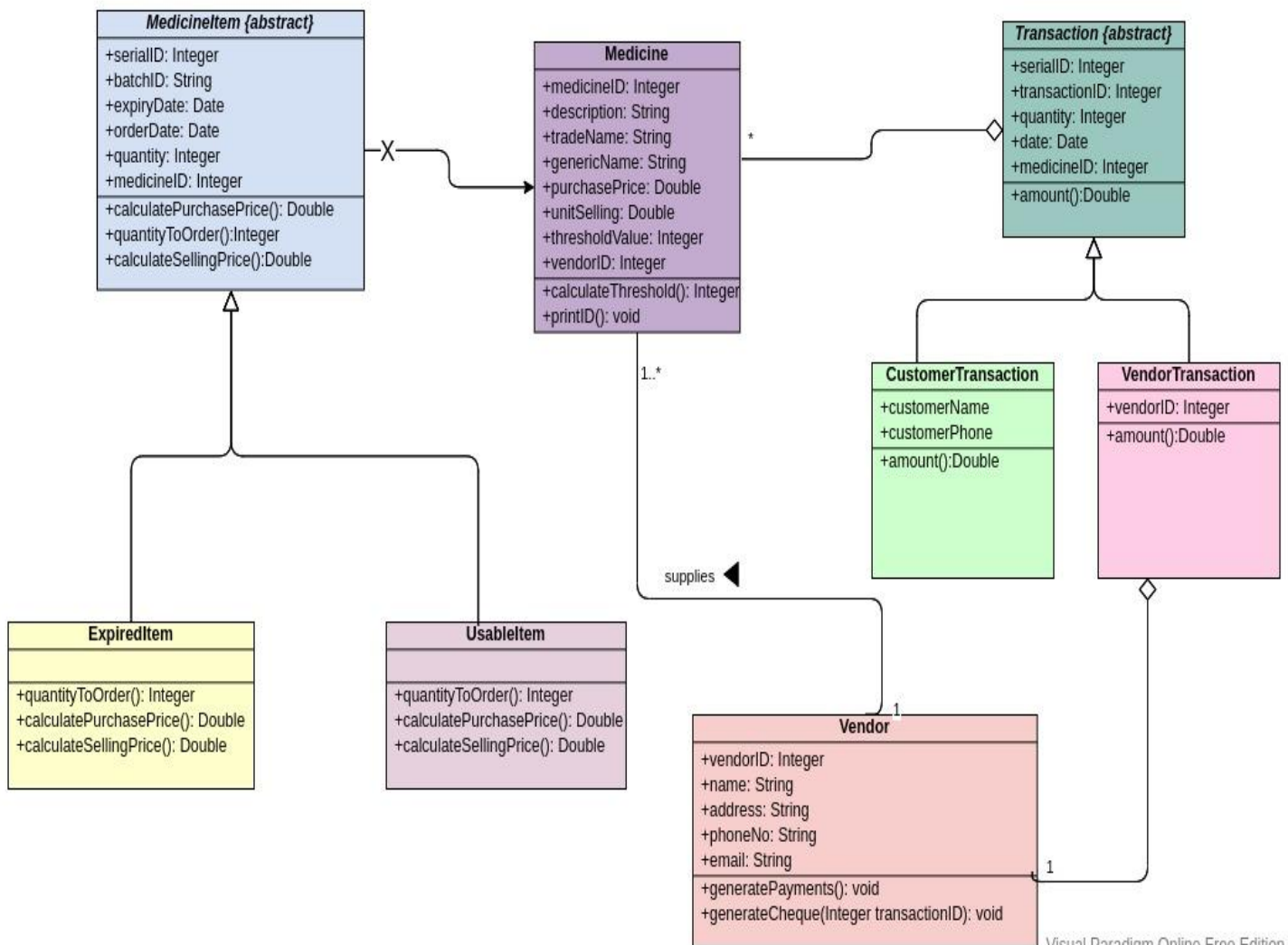
### 1. Use Case Diagram for MSA



## 2. Class Diagram for MSA

Visual Paradigm Online Free Edition

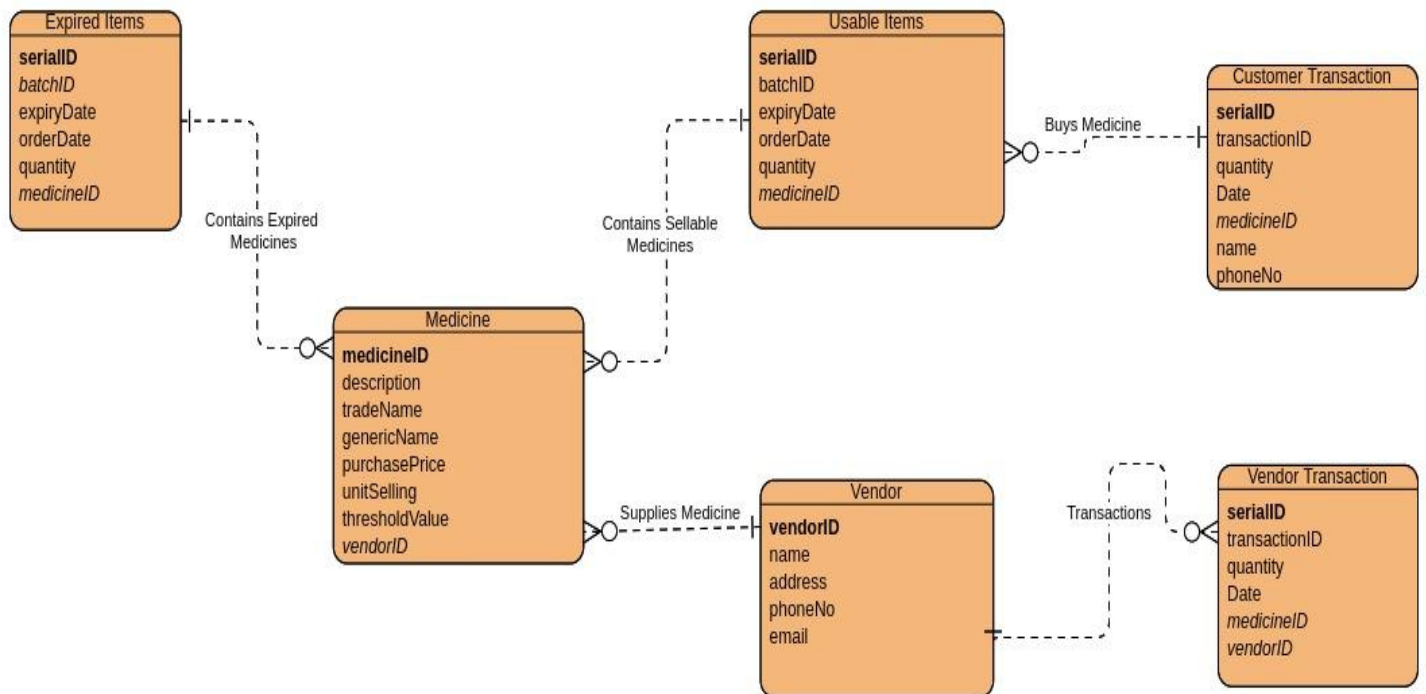
### CLASS DIAGRAM FOR MSA



Visual Paradigm Online Free Edition

### 3. Entity Relationship Diagram

Visual Paradigm Online Free Edition

**ENTITY RELATIONSHIP DIAGRAM FOR MSA**

Visual Paradigm Online Free Edition