|  |  |
| --- | --- |
| page1image3828864 | **B. P. Poddar Institute of Management and Technology**  **Department of Computer Science & Engineering**  **Software Engineering Lab (ESC-591)**  **AY: 2021-22 ODD Semester**  **Assignment-3 (System Requirement Specification)** |

Group No. (Case Study No.)\_\_\_4\_\_\_\_\_

Case Study Title: MEDILAB AND DRUG STORE INFORMATION STORE\_\_\_\_\_

**Student Details:**

|  |  |  |
| --- | --- | --- |
| **Univ. Roll. No.** | **Name** | **Contributions in SRS (section No.)** |
| **11500119047** | **Soumyajit Pramanik** | **2.6, 3** |
| **11500119048** | **Tuhin Hazra** | **2.5, 5** |
| **11500119051** | **Tandrima Ghosh** | **1, 5** |
| **11500119052** | **Sneha Tiwari** | **1.3, 2, 4** |

**Please do not write anything below the dotted line >**

................................................................................................................

**Marks awarded (Total Marks = 20)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Student Name |  |  |  |  |
| Marks Awarded |  |  |  |  |

Signature of Faculty with date\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Case Study Description:**

MLDIS is a system designed to cater to the needs of a typical test laboratory where patients can go for different types of prescribed tests and also buy prescription drugs.

**1.0 Introduction**

The aim of the document is to familiarize the user with the basic functions of the medilab and drug store system. This project is a prototype designed mainly for medilab and drug stores to maintain the inventory and accounts details limited to the use of its employees. This software can be used by patients, physicians, and staff at the clinic. This management system is designed as such to ease the workload of both physicians & pharmacists. The following SRS document serves the purpose, scope, definitions, conventions, references and a complete overview of our system.

**1.1 Purpose**

The purpose of this SRS document is to give a detailed description of the requirements for the “Medilab and Drug store system”. It will illustrate the purpose, the functional and non-functional requirements of the software and complete declaration for the development of the software. The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system, its requirements with respect to its users. It will also explain system constraints, interface and interactions with other external applications. This is a Demo Project, so some of the functionalities may not be present in the actual software.

**1.2 Document Conventions**

The document convention is easy to understand and simplified .

* All major section headings are in bold .
* Hyperlinks (if any) are indicated by blue.
* MLDIS – MediLab & Drug Store Information Store
* DB - Database
* ER - Entity Relationship
* OS - Operating System
* DIC - Design and Implementation Constraints
* ONR - Other Non-Functional Requirements
* UCC - User Classes & Characteristics
* Req. - Requirement
* Admin - Administrator

**1.3 Intended Audience and Reading Suggestions**

Document is intended for project managers, developers, employees and users. These suggestions are also helpful for the testers and software distributors. Below are the section details which require the attention of each entity.

|  |  |
| --- | --- |
| **Intended Audience(s)** | **Reading Suggestions** |
| Patient/User | 1.0 to 1.3, 2.2, 2.3.1, 2.6, 2.7, 4.1, 4.3, 4.5, 4.6 |
| Employees | 1.0 to 1.3, 2.3.2, 3.1, 4.2, 4.4 |
| Developers | 1.0 to 1.4, 2.2, 2.5, 3.0 to 3.3, 5.0 to 5.5 |
| Manager | 1.0 to 1.3, 2.3.2, 3.1, 4.0 to 4.6 |

**1.4 Product Scope**

The medilab software will function at its best within its scope of work. Some of its limitations are listed below:

1. The software should be efficient and fast.
2. There should be validation checks to prevent any incorrect entry of data.
3. It will keep the record of availability of medicines and test dates.
4. The adding, editing and deleting of records must be possible.
5. Feedback can be given by the patient and can be viewed by the manager.
6. It will generate a bill indicating the total amount payable and all details of the sales transaction.
7. It will automatically update the system when an appointment booking is made.
8. Managers can generate sales statistics of drugs for any period and make assumptions wherever necessary.

**1.5 Reference(s)**

Following are the websites and online resources that were referred to prepare the SRS document –

**WEBSITES:**

# [Software Requirements Specification document with example](https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database)

* [SRS IDS Online Sample](https://cmps-people.ok.ubc.ca/rlawrenc/teaching/304/Project/Samples/Design/IDS_Design2.pdf)
* [Pharmacy Management System – Requirement Analysis Document](https://www.docsity.com/en/srs-pharmacy-document/4124217/)
* [Software Requirements Specification (SRS) Book E-Commerce System (BECS)](https://www.cse.msu.edu/~chengb/RE-491/Papers/SRS-BECS-2007.pdf)

# [Software Requirement Specification (SRS) Format - GFG](https://www.geeksforgeeks.org/software-requirement-specification-srs-format/)

# [How to Write a Software Requirements Specification (SRS Document)](https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document)

**E-BOOKS:**

# Software Engineering: A Practitioner's Approach – By Roger S. Pressman

* Fundamentals Of Software Engineering – By Rajib Mall
* Essentials of Software Engineering (English, Thangasamy S.)

**2.0 Overall Description**

The below sub-sections carefully describe the functions of the product. It broadly focuses on the user classes, its characteristics, contains complete user documentation and other pieces of information required to build the software.

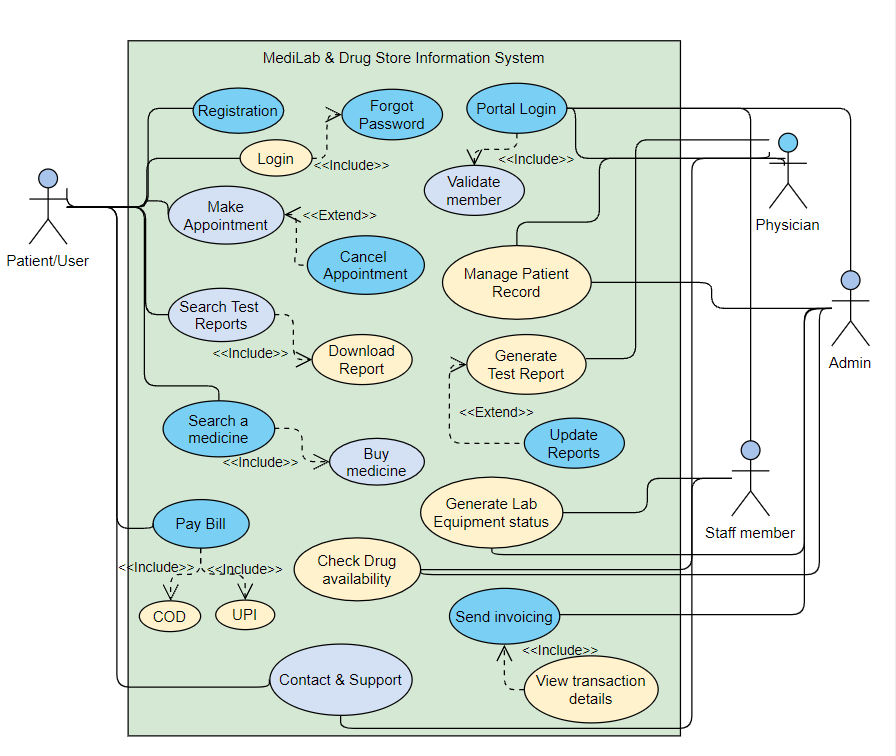
**2.1 Product Perspective**

A distributed medilab-drug store DB system stores the following information:

* **External users:** Patients who are taking appointments for tests or customers who are ordering drugs.
* **Internal users:** Physicians who send reports to their patients or pharmacists who take customer orders and inquiries. Finally, internal system users are IT personnel responsible for the maintenance of the database.
* **Staff:** This refers to a person who helps a user to order any drugs online and provides assistance during laboratory/radiology/pathology tests.

**2.2 Product Functions**

The major functions of the MediLab & Drug Store Information System has been shown below by the UML Use Case diagram:

****

**2.3 User Classes and Characteristics**

MLDIS is a web-based Medilab & Drug store system that allows patients to arrange appointments for tests, get reports of the same, order drugs, track the status of placed orders, and offers basic ordering process security. The system also supports internal use by physicians, pharmacists, managers and other staff members who receive user’s reqs such as slots for tests, drug orders, and status information. The system aids them in maintaining their equipment, products and of course, patient database. Some UCCs have been listed.

The system will support two types of user privileges, Customer, and Employee. Customers/users/patients will have access to customer functions, and the employees will have access to both customer and lab management functions.

**2.3.1** The customer should be able to do the following functions:

* Make an appointment
* Pre-booking (3 days in advanced or less)
* Immediate (same day or within an hour)
* Flexible Date/Time
* Payment accepted after test
* Confirmation within 15 minutes or less
* Cancel appointment
* Order medicines & track status
* Buy drugs/medicines (both prescribed and non-prescribed if permissible)
* Payment online or in cash
* Check status of delivery
* No return. Refund only for defective/expired products
* Check new/existing reports
* Check updates on current lab test report
* Search for previous reports
* Download a copy of results

**2.3.2** The Employee should have following management functionalities:

**ADMINISTRATOR**

* Manage patient record
* Manage Billing system
* Send invoicing
* Contact dealers, users/patients for due payments
* Look for supplies and other requirements
* Track status of equipments, drugs
* Place order for the same

**PHYSICIAN**

* Check patient’s medical history
* Generate/Update Test reports
* Upload reports of the patient after test
* Update reports in case of discrepancy

**STAFF MEMBER**

* Check drug availability
* Generate Lab Equipment status

**2.4 Operating Environment**

Operating environment for the MLDIS system is listed below:

* Distributed database
* Client/server system
* OS: Windows/Mac (compatible with Internet Explorer)
* DB: SQL server
* Configuration tools: Java, PHP

**2.5 Design and Implementation Constraints**

**Language DIC:**

* **For Website:** HTML, CSS, JavaScript , php.
* **For DB:** Relational Database (MySql)

**Other DIC:**

* **Password:** Minimum password length is 10. Format – alphanumeric only
* **Name:** Patient’s name should be in all upper case.
* **Ph. number:** Working 10-digit number.
* **Address:** Proper address of a patient
* **Payment:** For safety purpose Only upi and cash are allowed to make payment.

**2.6 User Documentation**

The user who can be patient/customer or physician or admin or staff member. The user who has visited the medilab and drug store for first time has to click on the register button to register themselves. Then he/she has to input full name, age, phone number, email, designation.

In “designation” user gets 4 options i.e, “patient/customer” or “physician” or “admin” or “staff member” from which user has to select. After that, user need to create a user id and password for log- in the website. After that user has to click in “log-in” button to log in with their created user id and password. If the user forgot password the by clicking in the “forgot password option user will receive an email to recreate password. Then log-in with the new password.

**2.6.1** Now for the **patient/customer** after log-in he/she can choose between two options i.e (i)“search medicine” and (ii)“search test”.

**(i) Search Medicine**

* After clicking on “search medicine” option the customer can type medicine name or manufacturer name in the search box.
* The customer can also choose the required medicine from the list of medicine given below the search box.
* As the customer writes in the search box the options of medicine appears that matches with the medicine in search box.
* On clicking on a particular medicine the details of the medicine are shown and “payment” option is provided.
* By clicking the “payment” option the customer is directed to payment gateways and the invoice is created.
* Then after filling the payment details “confirm payment” option appears. If customer click on “confirm payment” then product is purchased.

**(ii) Search Test**

* After clicking on “search test” option the patient can upload their medical documents such as doctor’s prescription in the “upload medical documents” box.
* Below that is the search box where patient can search for appropriate medical test. Patient can also choose from list of medical tests given below the search box.
* As the patient writes in the search box the options of medical test appears that matches with the one in search box.
* After clicking on appropriate medical test patient can choose the suitable time slot and date.
* After selecting the time and date patient can click on “confirm appointment” option to make an appointment patient can also cancel their appointment by going to the profile and clicking on ” cancel appointment” option that can be found beside the pending test.
* If there are any medical instructions needed to be followed then that will be informed via e-mail or message. After the test is done the test reports are uploaded in the patient’s portal and sent via registered email.
* The patient can click in their profile and click on test reports to get the list of test reports of test which has been done. The download option is exactly beside every test reports.
* Then patient can click to “payment” option to make online payment. Patient is directed to payment gateways and the invoice is created. Then after filling the payment details “confirm payment” option appears.

**2.6.2 For Physician:**

* Now for the **physician** after log-in he/she can click on the “schedule” option to check the schedule of medical test of different patients that has already been uploaded in physician’s routine in his/her portal.
* Physician can also access patient’s medical history by clicking on the “patient’s document” option available in the schedule of test. After completing a particular test physician can input the results of the test and generate the reports.
* After that he can click on the particular test which report has been generated from their schedule. Then clicking on the “upload report” option he can upload the test report that can be accessible by the patient.
* Physician can also check availability of lab equipment by clicking on the option “equipment” that will appear in their portal immediately after log-in.
* They can also order for equipment if the stock is low by clicking on “order” option located beside every equipment’s name in equipment portal.

**2.6.3** **For Admin:**

* Now for the **admin** after log-in he/she can click on “schedule” to schedule all the appointments made by the patient and save it in physician’s portal in schedule portal.
* Admin can also access the patient medical history if uploaded by patient. Then admin can input it in the test details of patient and save it in physician’s routine webpage.
* The medical history of patients can be updated by adding the recent test reports which would be saved in patient’s portal.
* If any low stock of medicine or lab equipment is reported by physician or staff member then required items can be brought and admin can update the list of medicines and lab equipment.
* The admin can also check invoice of patient/customer for the test done or medicine purchased and create transaction details and sent to the patient/customer email or upload in patient/customer portal which patient/customer will find beside every completed test’s “download” option.

**2.6.4 For Staff member:**

* Now for the **staff member** after log-in he/she can click on “check availability” option.
* Then find two options i.e (i) medicine to check medicine stock (ii) lab equipment to check equipment stock for the lab.
* After any low stock is reported and low stock or required medicines are brought the Staff members can update the stocks by clicking on the option “update” that is shown beside every item’s name.
* If there is any anomaly in the software the staff member can click on “Contact and support“ option to know about more details.

**2.7 Assumptions and Dependencies**

1. For getting a test done, a user can only book or cancel an appointment. He cannot pay the bill for tests via system. He/she can only do so while being present at the clinic.
2. Payment is only enabled for patients buying medicine or drugs online.
3. The system automatically updates information regarding the stock of lab equipment or drug availability. It will generate a warning or signal in case of any inadequacy.

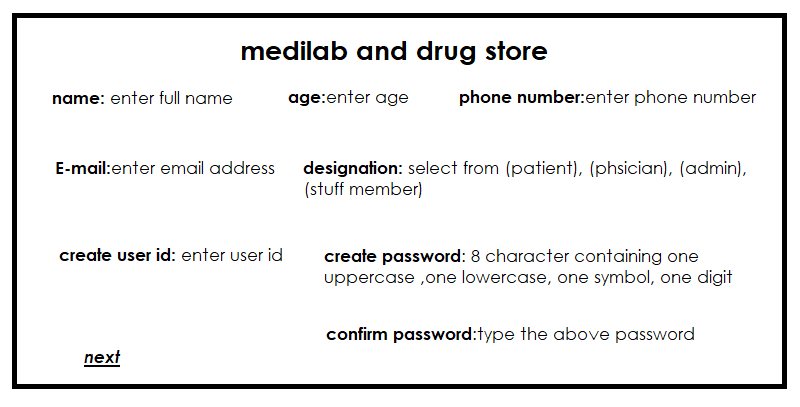
**3.0 External Interface Req.**

The software is going to interact with is

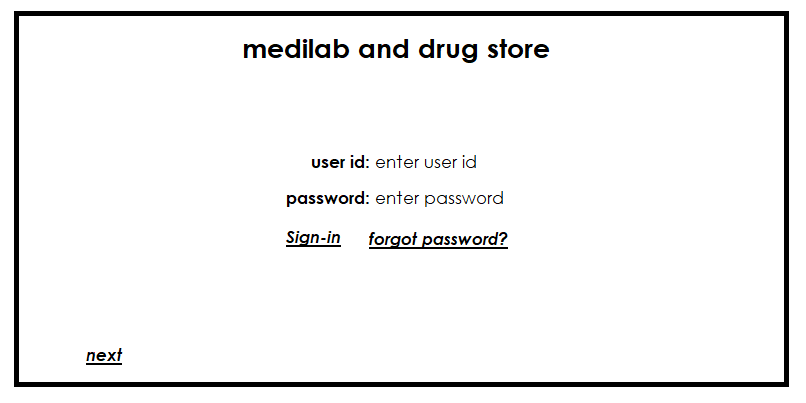
* payment gateways such as RuPay
* Lab equipment supplies medilab

**3.1 User Interfaces**

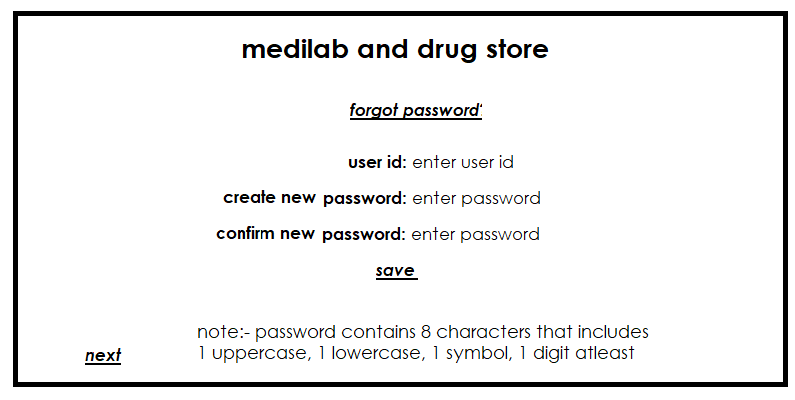
**3.1.1** **NEW REGISTRATION**

****

**3.1.2 LOGIN**

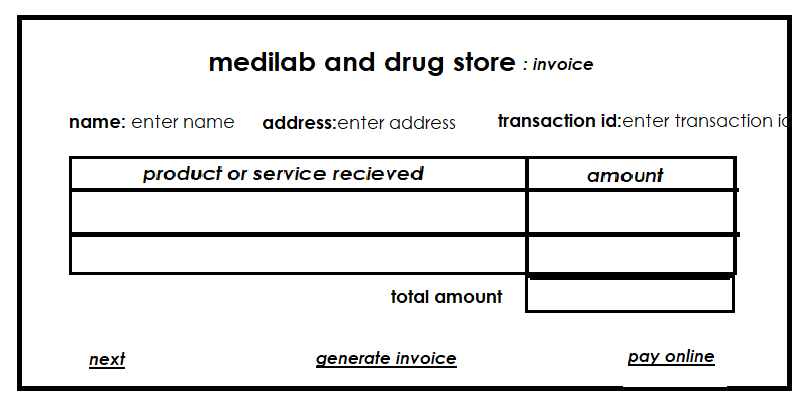
****

After clicking to forgot password the link opens for the email sent to user



The user who is a new comer has to register in the website to create user id and password in respective tabs eg. Any patient will enter patients portal, a physician will enter physician portal .

Then log in the website portal with created user id and password. If the user has lost password or user id then he/she can go for forgot password by which he will be sent a message in registered email to change his password.

**3.1.3 PAYMENT:**

**3.2 Hardware Interfaces:**

The users inputs through the system which has a mimimum requirements to build this project

**1. Processor:** 1.6Ghz and above

**2. Processor speed:** 1.7GHZ

**3. RAM:** 1GB RAM

**4. Hard disk:** 10GB HHD

**5. Monitor:** 15” colour monitor

**6. CD drive:** 52- X CD ROM

**3.3 Software Interfaces:**

The software interacts with hardware throughout the system. The minimum software requirement is:

**1. Operating System:** Windows XP, Windows 10, Linux

**2. Base language:** JAVA, PHP

**3. Website:** HTML, CSS

**4. DB:** MySQL server

**5. Browser:** Google Chrome, Firefox, Microsoft Edge

**6. Web Server:** Tomcat 7

Some Databases are required for the following:

1. Person ID

2. Medical Record History

3. Test reports

4. Payment details

5. Medicine and equipments stock

6. Invoice of patients for tests done

7. Bill of customers for medicines

**3.4** **Communication Interfaces:**

When a patient came across the receptionists at first registration was done. After logging in the instructions for patients about what conditions to follow will be sent to patients by email or by message a day before the allocated time for medical test. If any error message is displayed to customer/patient due to software faults during registration, or login, or payment method, etc then they can use contact and support option.

The patient's test reports will also be sent via email that patient entered during registration.

**4.0 System Features**

Following is the list of system functions and their respective users. Six major functions helpful to both customers and employees have been broadly mentioned in the later sections.

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Function/Feature of the system (USE CASE)** | **User (ACTOR)** |
| 1. | Registration | Physician, Patient/User |
| 2. | Login | Physician, Patient/User, Staff member |
| 3. | Manage the patient history | Admin, Physician, Staff member |
| 4. | Track status of the lab equipments | Admin, Staff member |
| 5. | Generate test report | Physician |
| 6. | Check availability of drugs | Admin, Staff member |
| 7. | Search for Medicine and Test availability | Patient/User |
| 8. | Request Medication | Patient/User |
| 9. | Make an appointment | Patient/User |
| 10. | Cancel an appointment or order | Patient/User |
| 11. | Contact & Support | Patient/User, Physician |
| 12. | Payment & transaction details | Patient, Admin |

**4.1 Make Appointment**

**Actor(s):** Patient

**Use Case Scenario Description**

A patient/user needs to get certain tests done. He/she will connect to our MediLab system to get an appointment for a test. The patient will need to see the reports and later on buy medicines. Hence, the patient can easily find medicines on the Drug Store Information system and purchase them.

**Use Case Description:**

A patient will make an appointment to get a test done. Patients can fix an appointment according to their convenience.

**Flow of Events/Scenario:**

1. A new user is required to register in order to use the system and make appointments.
2. A new patient ID and password will be generated once basic info of the user is collected.
3. Existing users can easily Login by entering a password.
4. The patient will click on ‘Make Appointment’.
5. He’ll find an option to select a time slot.
6. Click ‘Confirm’ to book.

**Alternate flow of events:**

**2a)** In case, the patient forgets his password, he’ll find the included ‘Forgot Password’ function which will direct him to the page to generate a new password.

**4a)** If for some reason, the patient wants to cancel any pre-booked appointment, he can easily do that using the software.

**4.2 Generate Test Report**

**Actor(s):** Physician

**Use Case Description:**

The physician at the clinic will perform the required tests on his patients. Once done he will upload the reports using the system.

**Flow of Events/Scenario:**

1. Physicians will search patients by entering their name or ID.
2. After generating the test report, he will upload it on the system.
3. He can also look for any previous reports/patient history if needed.

**Alternate:**

1. In case of any discrepancy in the report, the physician can update any existing report. Changes will be reflected every time.

**4.3 Search Test Report**

**Actor(s):** Patient

**Use Case Description:**

A patient will search for the test reports once uploaded. He can look for any previous reports as well. He can also download a copy of his reports.

**Flow of Events/Scenario:**

1. Once test reports are uploaded, a notification via email will be sent to the user.
2. The patient/user can directly click over ‘Reports section’ to open & check reports.
3. He can also search previous tests reports; all sorted by date & time.
4. He can click on ‘Download’ to receive a copy of it in PDF format.

**Alternate flow of events:**

**2a)**  In case, patient looks for a report which is not yet uploaded, the status will show ‘Pending’.

**3a)**  If there’s a glitch due to which patient is unable to open or download the reports, he can get help from ‘Contact & Support’. He’ll receive a copy of the desired report via email.

**4.4 Generate Lab Equipment status**

**Actor(s):** Staff member, Admin

**Use Case Description:**

The job of a staff member is to check if the equipment required for the tests are available in the stock. Thus, he needs to check the status of the equipment at any given point. The Admin also has the permission to view it.

**Flow of Events/Scenario:**

1. The staff member will click on ‘Generate Lab Equipment status’ to see if there’s enough stock of both reusable and non-reusable tools.
2. The system automatically keeps updating the status.
3. At the end of the day, staff members can send a copy of the equipment status to Admin if needed.

**Alternate flow of events:**

**3a)** The system will give some kind of red signal or warning if immediate refills for tools are required.

**4.5 Search medicines**

**Actor(s):** Patient

**Use Case Description:**

A patient can search and buy a medicine using the system. The medicines will be delivered at home.

**Flow of Events/Scenario:**

1. Any user can look for medicines or prescribed drugs using the ‘Search Medicine’ option.
2. He then needs to click on ‘Buy medicine’.
3. Now he can proceed to the payment.
4. The medicine/drug will be delivered at the user's doorstep within 5 business days.
5. The status of the product can be monitored.
6. Once Out for Delivery, the user will be informed via email.

**Alternate flow of events:**

**4a)** In case of a delayed delivery, it will be notified via mail with a valid reason.

**6a)** Once the product is received, if there’s any issue (say, poor packaging, broken seal, expired or wrong medicine), patients can get support from ‘Contact & Support’. A fresh vial of the same drug will be delivered soon.

**4.6 Pay Bill**

**Actor(s):** Patient

**Use Case Description:**

To purchase a medicine/drug, patients can easily find modes of payment over the system. A secured gateway for payment will be generated.

**Flow of Events/Scenario:**

1. Patient will click on ‘Proceed to Payment’.
2. Two modes of payment will appear: COD (Cash on Delivery) and UPI.
3. Once selected, the user will be redirected to a secured Payment Gateway page (preferably RazorPay).
4. A confirmation message will pop up once an order is successfully placed.
5. Within minutes, an email with the confirmation & transaction details will be sent by the system.

**Assumption:**

For security reasons and protection against frauds, we do not prefer keeping Credit Card details or any financial data of the patients. Hence, only 2 modes of payments are available for simplicity.

**Alternate flow of events:**

**4a)** Incase, a confirmation doesn’t show up after payment, the user will be requested to wait for the next 30 minutes.

If still no message is received, the patient can get help from ‘Contact & Support’.

**5.0 Other Non-functional Requirements**

Here are some ONRs of the system:

**5.1 Performance Requirements**

* **Usability:** The application should load and be usable within 3 seconds.
* **Response Time:** The application should update the interface on interaction within 2 seconds.
* **Throughput:** Throughput of the system should be normalized to prevent redundant data and improve performance
* **Capacity:** The system should be distributed to prevent outage.

**5.2 Safety Requirements**

* Secure any transmissions of private information (name, address, phone no. of the patient, test reports) between the patient and the company.
* Databases should use [Sharding](https://docs.mongodb.com/manual/sharding/#:~:text=Sharding%20is%20a%20method%20for,capacity%20of%20a%20single%20server.) to be redundant to prevent loss of data.
* Backups of the databases should be done hourly and be kept for one week.
* Prevent false information from being used as payment.
* Prevent false information gives from being used when registering.

**5.3 Security Requirements**

* Any keys used for the REST API should be stored securely.
* Only the REST API should be able to connect to the databases.
* Databases should be behind a firewall.

**5.4 Software Quality Attributes**

* **Availability:** Because this application is critical to business communication, we will have a goal of more than 95% availability.
* **Correctness:** The application should never allow anyone to read messages or discussions not intended for that person.
* **Maintainability:** The application should use continuous integration so that features and bug fixes can be deployed quickly without downtime.
* **Usability:** The interface should be easy to learn without a tutorial and allow users to accomplish their goals without errors.

**5.5 Other Requirements**

Each user of the SRS is required to login in his/her account to perform different activities like test, buy medicine, transactions, update inventory, view drug requirements and update process etc.

DATABASE REQUIREMENTS:

All the data is saved in the data base:

* report information
* schedule information
* medicine update information

The database allows concurrent access by various staff and is kept consistent at all the times requiring a good database design.

-------------------- xxxxx ---------------------