Contents

Contents ii

Revisions ii

1 Introduction 2

1.1 Document Purpose 2

1.2 Product Scope 2

1.3 Intended Audience and Document Overview 2

1.4 Definitions, Acronyms and Abbreviations 2

1.5 Document Conventions 3

1.6 References and Acknowledgments 3

2 Overall Description 3

2.1 Product Overview 3

2.2 Product Functionality 4

2.3 Design and Implementation Constraints 4

2.4 Assumptions and Dependencies 4

3 Specific Requirements 4

3.1 External Interface Requirements 4

3.2 Functional Requirements 7

3.3 Business Rules 8

3.4 User Scenario 9

3.5 Use Case Model 10

4 Other Non-functional Requirements 42

4.1 Performance Requirements 42

4.2 Safety and Security Requirements 42

4.3 Software Quality Attributes 42

5 Other Requirements 43

Appendix A – Data Dictionary 43

o

Revisions

| Version | Primary Author(s) | Description of Version | Date Completed |
| --- | --- | --- | --- |
| 1.0 | Full Name | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 14/12/2016 |

# 

# Introduction

## Document Purpose

The purpose of this document is to give a detailed description of the requirements for the “Global Medical System” software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

## Product Scope

This main objective of this software is to create a network between patient, doctor, pharmacy and medical company. “Global Medical System” is a both desktop/mobile application where a patient can make appointment from doctors, search inventory of pharmacies for medicine. A patient can also search the nearest pharmacy and specialised doctor near him by using the application.

This application includes special feature for doctors where they can check the appointment list of patients, update and view their profile for patient, checking the history of their patient, see the routine and the list of their consultation chamber. They can also get benefited by watching their monthly income by using this application.

## Intended Audience and Document Overview

This document is mainly written for the stakeholders of this project like developers (front end / back end), project manager, sponsors, testing and quality assurance team, reviewers and all type of customer including patient, doctor, pharmacy and medical company.

## Definitions, acronyms, and abbreviations

**Table-1 Definitions**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| User | Someone who interacts with the mobile phone application or website. |
| Admin/Administrator | System administrator who is given specific permission for managing and controlling the system. |
| Web Portal | A web application which present special facilities for patients & admin also. |
| GPS | Global Positioning System |
| GPS - Navigator | An installed software on mobile phone which could provide GPS connection and data, show locations on map and find paths from current position to defined destination |
| Stakeholder | Any person like patents, Doctor, Pharmacy, Medicine company who has interaction with the system  who is not a developer |

## References

[1] .Software Requirements third edition by Karl Wiegers and Joy Beatty.

[2]. IEEE Software Engineering Standards Committee, “IEEE Std 830 - 1998, IEEE Recommended Practice for Software Requirements Specifications”, October 20, 1998.

[3]. UML Weekend Crash Course™ by Thomas A. Pender.

[4]. https://www.google.com/search?q=sample+srs&ie=utf-8&oe=utf-8&client=firefox-b-ab

# Overall Description

## Product Overview

This application includes thousands of features for patients, doctors, pharmacies and pharmaceutical companies.

Sometime it is hard for the patients to find their medicines in pharmacies as they can’t find their needed medicine in one place. They have to travel more than one pharmacy to collect their desired set of medicine.by using this application patient can search the pharmacies containing their needed medicine. This application presents the feature of finding nearest pharmacy for patient.

Patient can also search doctors and make appointment from home.  
Doctors can check their patient’s history, appointment list and maintain their routine easily by using this application.

Pharmacy owners can easily manage their business by using an inventory system which is included in this application. They can insert, update and search medicines by using this inventory system. The patients and the pharmaceutical company can also check the status of this inventory system. The pharmacy owners also can request to get refilled from pharmaceutical company by using the application.

## Product Functionality

There is huge functionality to maintain the system. The major functionalities are,

1. Login panel for four different types of users
2. The system can contain patients profile with history
3. Inventory system for pharmacy owners
4. A portal to treat an emergency patient without any hassle by providing proper drugs
5. Different interface for different types of users
6. An platform friendly system for all kind of users (Mobiles, Computer etc.)

## Assumptions and Dependencies

To run this application user will need mobile or desktop platform which is run by electricity. So it is hard to use this application where electricity is not available. As the application fully depends on server source, user must have internet connection to use the application. Though this application is based on both mobile/desktop platforms, it will be hard to manage the inventory system by mobile for pharmacy owners. So it is advised to manage the inventory system by desktop platform.

## Constraints

1. The mobile application is constrained by the system interface to the GPS navigation system within the mobile phone.
2. The Internet connection is also a constraint for the application.
3. Both the web portal and the mobile application will be constrained by the capacity of the database. Since the database is shared between both application it may be forced to queue incoming requests and therefore increase the time it takes to fetch data.
4. Pharmacies can’t sell any illegal drugs.
5. Doctors can’t prescribe any illegal drugs.
6. Patients can’t consult with doctor if they have police obligation.

# Specific Requirements

## External Interface Requirements

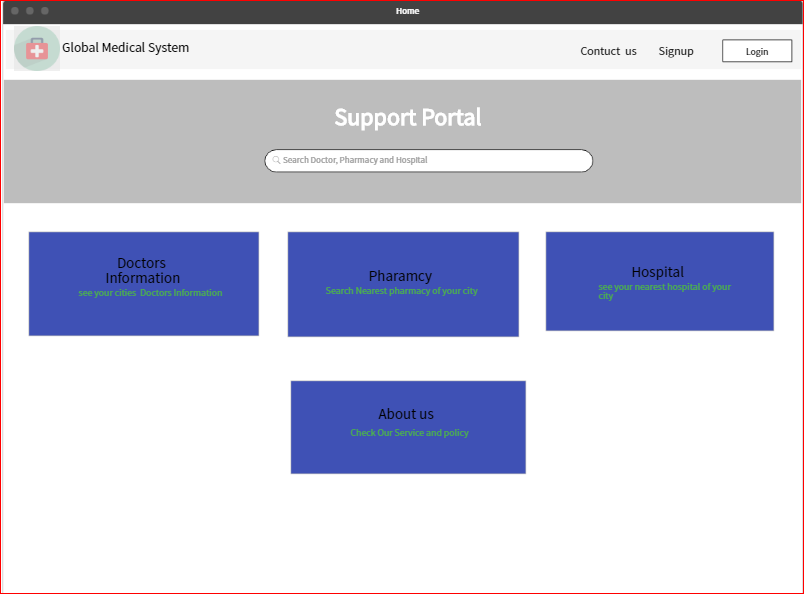
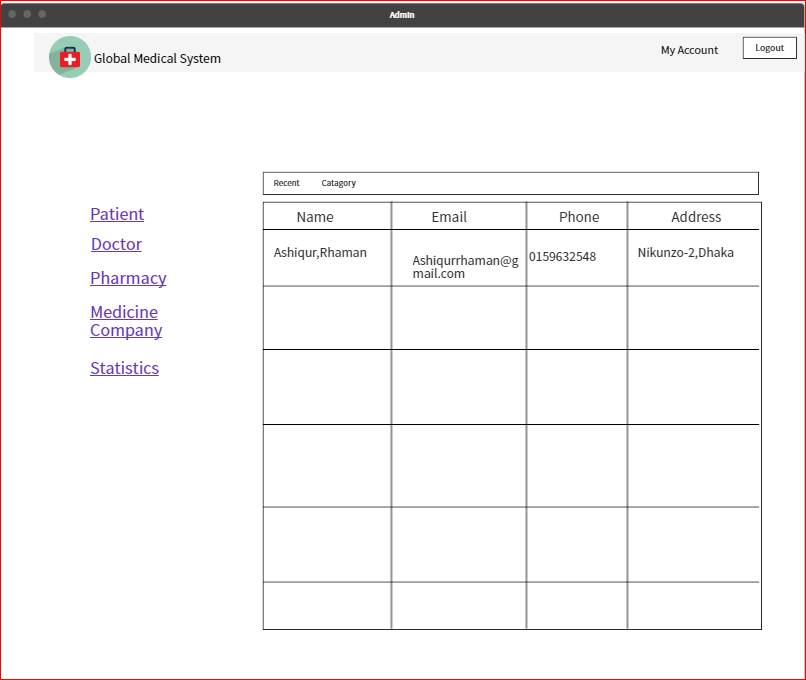
### User Interfaces

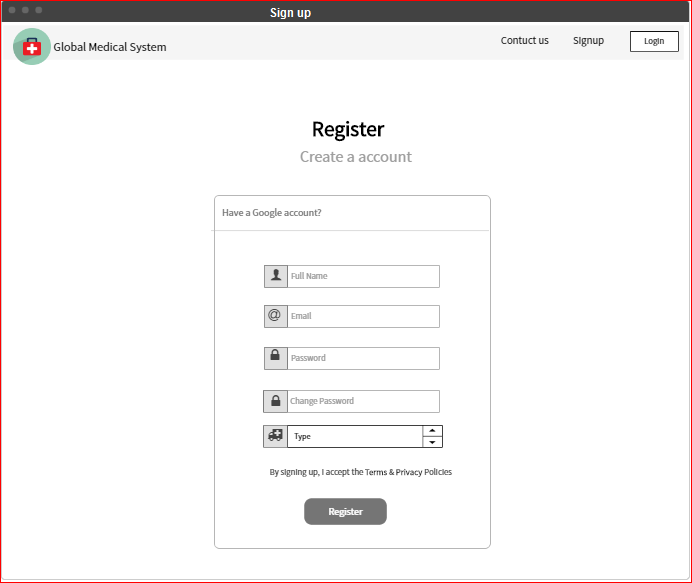
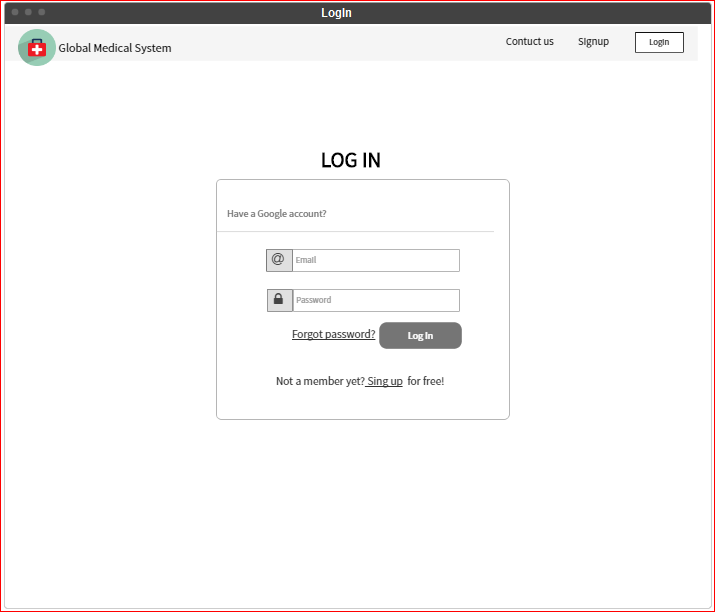
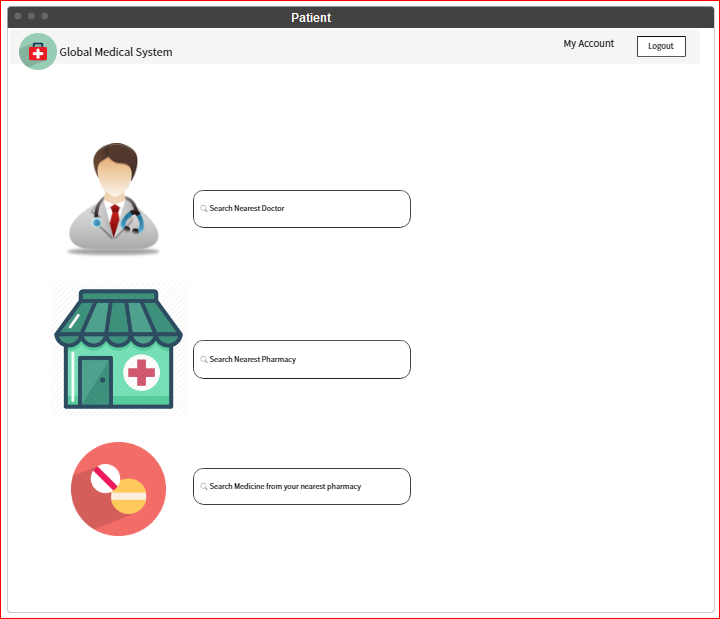
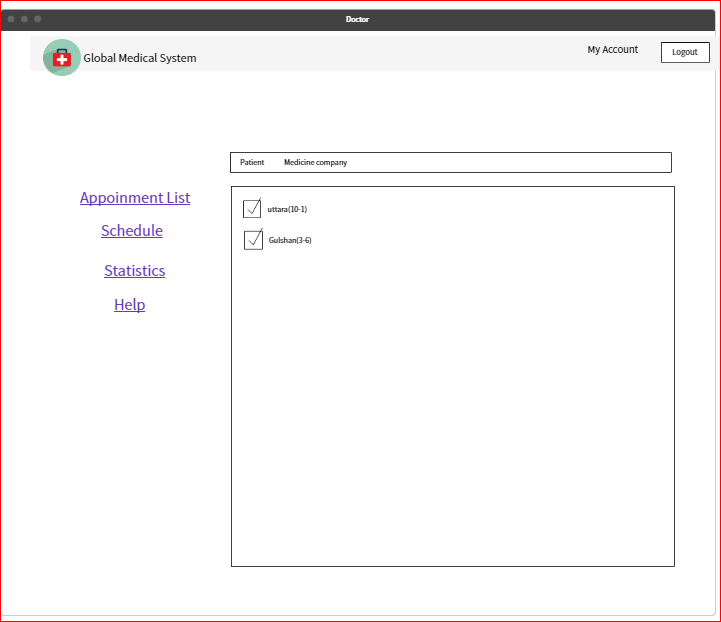
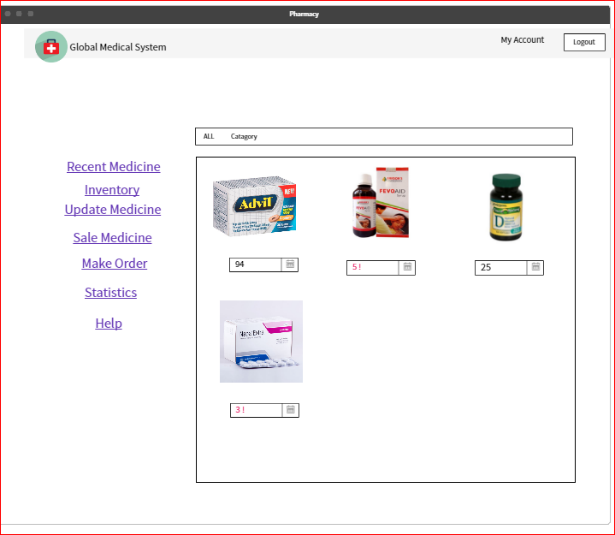
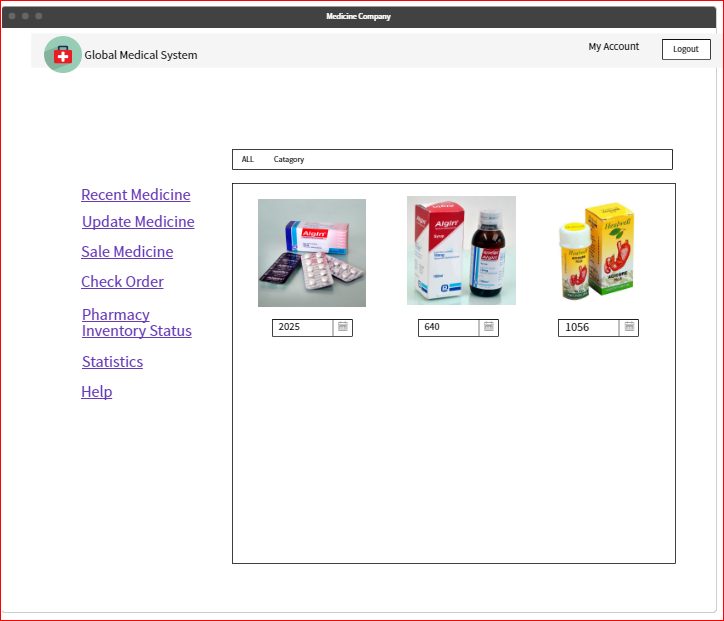
**->**GUI is only in English.

->Login and password is used for the identification of users.

->Only registered patients, doctors, Medicine Company and Pharmacy will be authorized to use the services.

->Limited to HTTP/HTTPS.

->This system is working for single server.



### Hardware Interfaces

|  |
| --- |
| Client Side |

|  |  |  |  |
| --- | --- | --- | --- |
| **Browser** | Processor | RAM | Disk Space |
| Google Chrome | All Intel or AMD-1GHz | 128MB | 100MB |
| Mozilla Firefox | All Intel or AMD-1GHz | 128MB | 100MB |
| Opera | All Intel or AMD-1GHz | 128MB | 100MB |
| Internet Explorer | All Intel or AMD-1GHz | 128MB | 100MB |

|  |
| --- |
| Server Side |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| RAD | All Intel or AMD-1GHz | 2 GHZ | 4 GHZ |
| Mysql Server | All Intel or AMD-1GHz | 2 GHZ | 3.5 GHZ |
| DB2 | All Intel or AMD-1GHz | 512MB | 500MB |

### Software Interfaces

1. Client on Internet
2. Web Browser, Operating System (any)
3. Client on Intranet

Web Browser, Operating System (any)

1. Web Server

WASCE, Operating System (any)

1. Data Base Server

DB2, Operating System (any)

1. Development End
2. RAD (J2EE, Java, Java Bean, Servest, HTML, XML, AJAX), DB2, WebSphere.

## Functional Requirements

1. Doctor has to login.
2. User id and password verification.
3. Save updated information.
4. Real time calculation.
5. Patient history can be sorted.
6. Search algorithm for nearest doctor.
7. Search algorithm for specialist.
8. Search algorithm for nearest pharmacy.
9. Login for pharmacy.
10. Selling history must be stored.

### Business Requirements

1. Police should verify doctor’s profile and certificate.
2. Police should verify license of pharmacies.
3. Pharmacies can’t input any illegal medicine in the application.
4. Doctors can’t input any illegal medicine in the prescription.

### External Interface Requirements

1. Input information.
2. Display information.
3. Output error message.
4. Output success message.
5. Print command.
6. Input charge per patient.
7. Input search parameter.
8. Input search parameter.
9. Display map.
10. Medicine name, price, amount
11. Reminder by sound.

### User Requirements

1. Doctors can update their information’s.
2. Doctors can make prescriptions.
3. Doctors can see previous appointment history.
4. Doctors can see patient previous history.
5. Patient can search nearest doctor.
6. Patient can search doctor by name.
7. Patient can search nearest pharmacies.
8. Patients can appointment with a doctor.
9. Pharmacist can update information.
10. Pharmacist can input medicine in inventory.
11. Pharmacist can search medicine.
12. Pharmacists can see sold medicines.
13. Pharmacists can get reminder when medicine is short in amount.

## Business Rules

1. Doctors must have all legal certificates and must be free from any kind of legal obligation.
2. Pharmacies have must be a trade licensed.
3. Pharmacies who sell special kind of medicine should have proper documents.
4. Medicine Company must have trade identity.
5. Illegal medicine can’t be sold.
6. Date expired medicine must not be included in the inventory of the pharmacy.
7. All doctor’s, pharmacies, medicine company must be included in our data base server for better outcome of searching.

## User Scenarios

Mr. Hamid is a 45 year old senior citizen. He has been suffering from diabetes for last 5 years. Doctor has prescribed him some unique foreign branded medicine. It is hard for him to find all the medicine in his area. So he has to go all the way from banani to shahbag in order to get all the medicine. Though he is a senior citizen but he loves to use different mobile application. He wants a mobile/web application through which he can easily search the inventories of the nearest pharmacies from his house.

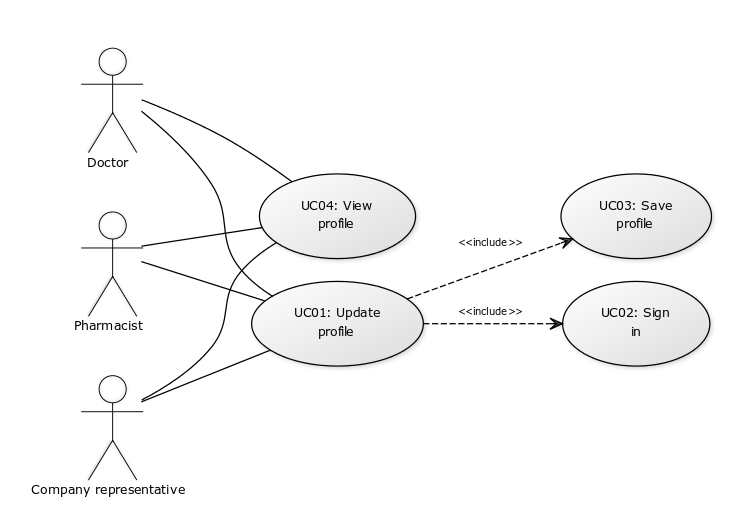
Since last 3-4 days he is having heavy pain in the left side of his chest. He wants to consult with Dr. Momenuzzaman who is one of best cardiologist in town. He wants to make an appointment to consult with Dr. Momenuzzaman. He wants to make it from home through his mobile phone. He also wants to find other cardiologist near him through the mobile application. So that he can compare between them by watching the patient review.

Dr. Momenuzzaman is a renowned cardiologist. For last couple weeks he his having some problem with his assistant Abdour Rahim. He figured out that Abdour Rahim is giving early appointment for patient by taking bribe. Dr. Momenuzzaman wants to fire Abdour Rahim for this hilarious work. But he also knows that he is totally dependent on his assistant as he has to maintain several chambers and consult with more than 100 patients every day. Abdour Rahim keeps track of all his chambers, routine, patient’s appointments, patient’s history and many more. He wants a mobile application where he can keep the list of his chambers, patient, and patient’s history. He is also curious to know how many patients he consults every day/week/month and he also wants to know from which chamber he consults most/least number of patient.

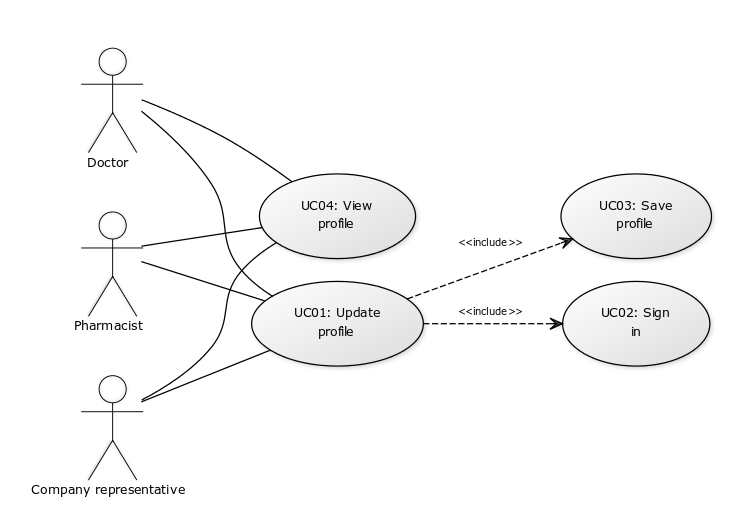
Mr. Rakib is pharmacy company owner and wants to increase the sale of his business. In order to meet the business goal, he wants to digitalize his pharmacy. He wants an inventory system where he can upload all the details of his medicines by using barcode scanner so that he can easily find in which place of the shelf his desired medicine is located by searching in the inventory system. He also wants to make the status of his inventory visible to all so that his customer can see which medicine has got in his inventory. He also wants the medical representatives to track down his inventory system so that they can easily fill it up.

Samson H. Chowdhury is the owner of square pharmaceutical company. Recently he is facing some problems with his medical representatives. As the number of pharmacy is increasing, so the medical representatives have to work harder to keep track and refill the inventories of too many pharmacies. So he comes up with the idea of a web application where a small team will keep tracking all the pharmacies inventory from the headquarter and refill their inventories fast. He also wants to add some features like knowing the best pharmacy, bills per month for pharmacy and by basing on these factors he wants to give some award to the pharmacy.

## Use Case ModelC:\Users\Aziz\Downloads\use case.png



### Profile update subsystem



**Author –** Ishmam Aziz

**Purpose**–Users will be able to see their profile and update their profile as they want.

**Priority** - High

**Preconditions**–User has registered for an account.

**Post conditions** –

**Actors**–Doctor, Pharmacist, Company representative

**Trigger** – It starts when user selects profile.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System shows profile. |
| 2. User selects edit. |  |
|  | 3. System checks if the user is signed in. |
|  | 4. System asks for changes. |
| 5. User edits his profile. |  |
| 6. User saves his profile. |  |
| 7. User tries to exit from editing. |  |
|  | 8. System checks if changes have been saved. |
| 9. User exits from editing. |  |

* Alternative Flow –

Step 4: User is not signed in. System asks user to sign in.

Step 9: Changes have not been saved. System asks user if he wants to save and go to step 5.

* 1. Exceptions –

**Notes/Issues** –

#### UC01 (Update Profile)

**Requirements Traceability –**

**Priority** - High

**Preconditions**–User must sign in to his account.

**Post conditions** –

**Actors**–Doctor, Pharmacist, Company representative

**Extends –**

**Includes**– UC02, UC03

#### UC01 (Sign in)

**Requirements Traceability –**

**Priority** - High

**Preconditions –**

**Post conditions**–User is signed in

**Actors**–Doctor, Pharmacist, Company representative

**Extends –**

**Includes**–

**Author –** Ishmam Aziz

#### 3.5.1.3 UC03 (Save Profile)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**– Changes have been made to the profile.

**Post conditions** –

**Actors**–Doctor, Pharmacist, Company representative

**Extends –**

**Includes**–

#### 3.5.1.3 UC03 (View Profile)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

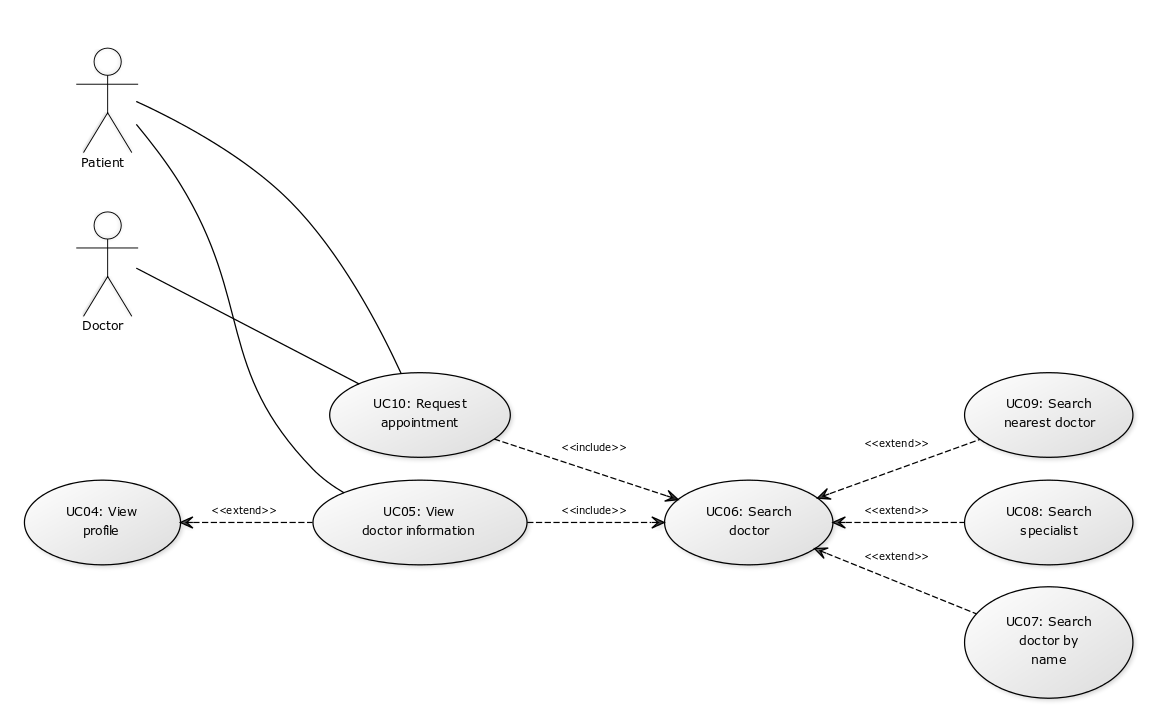
**Post conditions** –

**Actors**–Doctor, Pharmacist, Company representative

**Extends –**

**Includes**–

### Searching doctor subsystem



**Author –** Ishmam Aziz

**Purpose**– Patient will be able to search a doctor to see his location and other information or to make an appointment with him.

**Priority** - High

**Preconditions**–

**Post conditions** –

**Actors**–Patient

**Trigger** – It starts when patient selects search doctor.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System asks for input. |
| 2. Patient gives a name or selects a specialty or selects nearest doctor. |  |
| 3. Patient selects search. |  |
|  | 4. System checks if the user gave any input. |
|  | 5. System displays sorted list of doctors. |
| 6. Patient selects a doctor. |  |
|  | 7. System displays doctor’s location and information. |
| 8. Patient requests appointment. |  |
|  | 9. System asks patient to give his information. |
| 10. Patient gives information. |  |
|  | 11. System checks contact number. |
|  | 12. System checks for doctor’s confirmation. |
|  | 13. System gives details of the appointment. |
| 14. User exits from the system. |  |

* Alternative Flow –

Step 4: Patientdid not give any input. System gives error message and go to step 1.

Step 5: Patient’s input did not match any result. System gives error message and go to step 1.

Step 11: Patient did not give any contact number. System asks for contact number.

Step 12: Doctor did not confirm the appointment. System gives a message.

* 1. Exceptions –

**Notes/Issues** –

#### 

#### UC05 (View doctor information)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** – patient can request appointment.

**Actors**–Patient

**Extends –** UC04

**Includes**–

#### UC06 (Search doctor)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** – A doctor is selected.

**Actors**–Patient

**Extends –**

**Includes**–

#### UC07 (Search doctor by name)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** – A doctor is selected.

**Actors**–Patient

**Extends –** UC06

**Includes**–

#### UC08 (Search specialist)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** – A doctor is selected.

**Actors**–Patient

**Extends –** UC06

**Includes**–

#### UC09 (Search nearest doctor)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** – A doctor is selected.

**Actors**–Patient

**Extends –** UC06

**Includes**–

#### UC10 (Request appointment)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

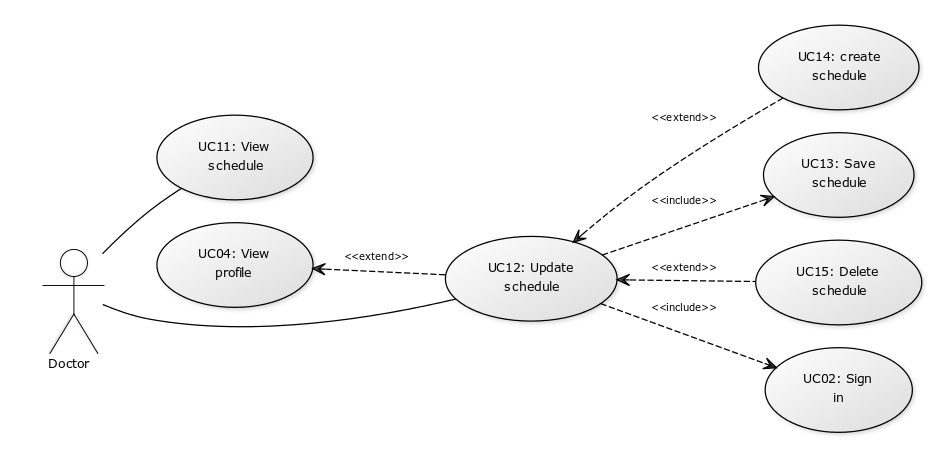
**Post conditions** – A doctor is selected.

**Actors**–Patient, Doctor

**Extends –**

**Includes** –

### Schedule handling subsystem



**Author –** Ishmam Aziz

**Purpose**– Doctor will be able to manage his time and work and give his location to the patients.

**Priority** - High

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Trigger** – It starts when doctor selects schedule.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System displays his schedule. |
| 2. Doctor selects update. |  |
|  | 3. System checks if doctor is signed in. |
| 4. Doctor selects create. |  |
|  | 5. System asks for details. |
| 6. Doctor gives details. |  |
|  | 7. System creates schedule. |
| 8. Doctor selects a schedule to delete. |  |
|  | 9. System deletes the schedule. |
| 10. Doctor saves changes. |  |
| 11. Doctor tries to exit. |  |
|  | 12. System checks if changes have been saved. |
| 13. Doctor exits. |  |

* Alternative Flow –

Step 3: Doctor is not signed in. System asks doctor to sign in and go to step 1.

Step 7: Doctor did not give time or the time overlaps other schedules. System gives error message and go to step 5.

Step 12: Changes have not been saved. System asks doctor if he wants to save.

* 1. Exceptions –

**Notes/Issues** –

#### UC11 (View schedule)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes**–

#### UC12 (Update schedule)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Extends –** UC01

**Includes** – UC02, UC13

#### UC13 (Save schedule)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**– Changes have been made in the schedule.

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes**–

#### UC14 (Create schedule)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Extends –** UC12

**Includes**–

#### UC15 (Delete schedule)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

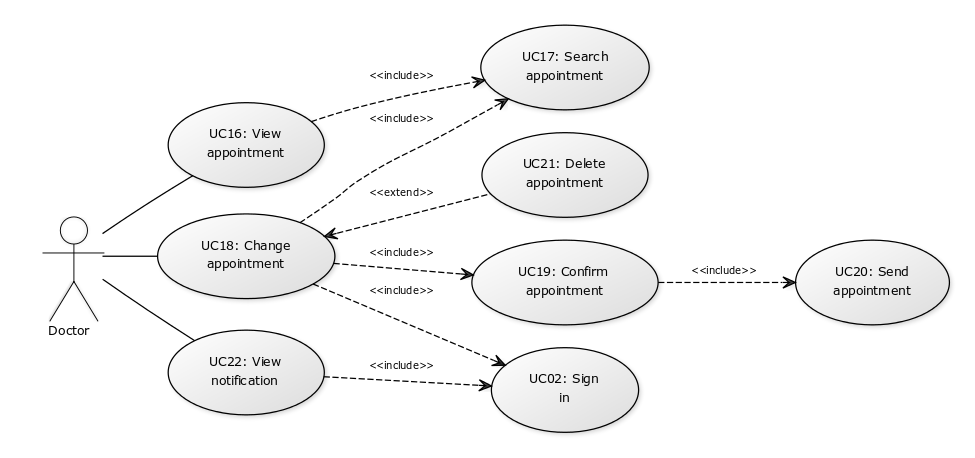
**Post conditions** – A schedule is selected.

**Actors**–Doctor

**Extends –** UC12

**Includes**–

### Appointment handling subsystem

**Author –** Ishmam Aziz

**Purpose**– Doctor will be able to make appointments with patients and see older appointments.

**Priority** - High

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Trigger** – It starts when doctor signs in.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System gives notifications. |
| 2. Doctor selects an appointment from the notifications or from appointment list. |  |
|  | 3. System gives details of the appointment. |
| 4. Doctor edits the appointment. |  |
|  | 5. System checks if doctor is signed in. |
| 6. Doctor confirms appointment. |  |
|  | 7. System sends appointment details to the patient. |
| 8. Doctor deletes the appointment. |  |
|  | 9. System checks if doctor is signed in. |
|  | 10. System sends cancelation message to the patient. |
| 11. Doctor exits. |  |

* Alternative Flow –

Step 5: Doctor is not signed in. System asks doctor to sign in and go to step 3.

Step 7: Doctor did not give any time. System gives error message.

Step 7: Appointment is past. System does not send any message.

Step 9: Doctor is not signed in. System asks doctor to sign in and go to step 3.

Step 10: Appointment is past. System does not send any message.

* 1. Exceptions –

**Notes/Issues** –

#### UC16 (View appointment)

**Requirements Traceability –**

**Priority** – High

**Preconditions**– An appointment is selected.

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes** – UC17

#### UC17 (Search appointment)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes** –

#### UC18 (Change appointment)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**– An appointment is selected.

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes** – UC17, UC19, UC02

#### UC19 (Confirm appointment)

**Requirements Traceability –**

**Priority** – High

**Preconditions**– Changes have been made in the appointment.

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes** – UC20

#### UC20 (Send appointment)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** –

**Actors**–Doctor, Patient

**Extends –**

**Includes** –

#### UC21 (Delete appointment)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Extends –** UC18

**Includes** –

#### UC22 (View notifications)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

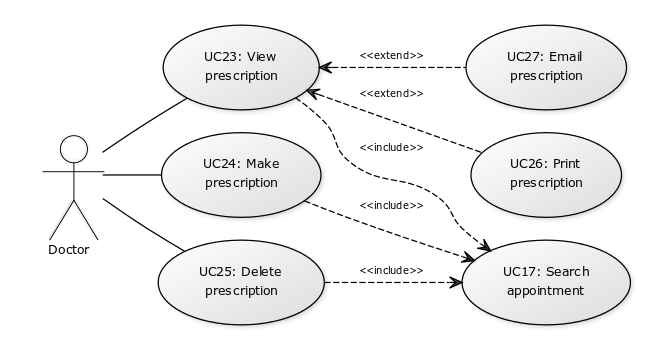
**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes** – UC02

### Prescription handling subsystem

**Author –** Ishmam Aziz

**Purpose**– Doctor will be able to make prescriptions and see older prescriptions.

**Priority** - Medium

**Preconditions**–An appointment is selected.

**Post conditions** –

**Actors**–Doctor

**Trigger** – It starts when doctor selects an appointment.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System gives details of the appointment. |
| 2. Doctor selects make appointment. |  |
|  | 3. System asks for details. |
| 4. Doctor edits the prescription. |  |
| 5. Doctor saves prescription. |  |
| 6. Doctor deletes prescription. |  |
| 7. Doctor prints prescription. |  |
|  | 8. System sends print command to the printer. |
| 9. Doctor emails prescription. |  |
|  | 10. System sends prescription to patient’s email address. |
| 11. Doctor exits. |  |

* Alternative Flow –

Step 10: Patient did not give any email address. System gives error message.

* 1. Exceptions – Printer is not responding. System gives error message.

**Notes/Issues** –

#### UC23 (View prescription)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**– An appointment is selected.

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes** – UC17

#### UC24 (Make prescription)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**– An appointment is selected.

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes** – UC17

#### UC25 (Delete prescription)

**Requirements Traceability –**

**Priority** – Low

**Preconditions**– A prescription is selected.

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes** – UC17

#### UC26 (Print prescription)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**– A prescription is selected.

**Post conditions** –

**Actors**–Doctor

**Extends –** UC23

**Includes** –

#### UC27 (Email prescription)

**Requirements Traceability –**

**Priority** – Low

**Preconditions**–

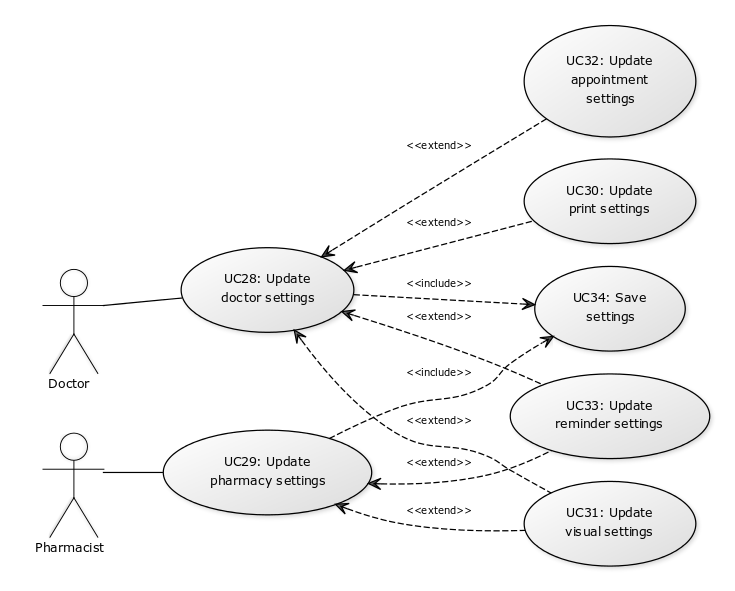
**Post conditions** –

**Actors**–Doctor

**Extends –** UC23

**Includes** –

### Settings handling subsystem

**Author –** Ishmam Aziz

**Purpose**– Users will be able to update their setting to make the system suitable.

**Priority** - High

**Preconditions**–

**Post conditions** –

**Actors**–Doctor, Pharmacist

**Trigger** – It starts when user selects settings.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System provides some options. |
| 2. User selects one option. |  |
|  | 3. System asks user to make changes. |
| 4. User makes changes. |  |
| 5. User saves changes. |  |
| 6. User tries to exit. |  |
|  | 7. System checks if the changes have been saved. |
| 8. Doctor exits. |  |

* Alternative Flow –

Step 7: Changes have not been saved. System asks user if he wants to save.

* 1. Exceptions –

**Notes/Issues** –

#### UC28 (Update doctor settings)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Extends –**

**Includes** – UC34

#### UC29 (Update pharmacy settings)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** –

**Actors**–Pharmacist

**Extends –**

**Includes** – UC34

#### UC30 (Update print settings)

**Requirements Traceability –**

**Priority** – Low

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Extends –** UC28

**Includes** –

#### UC31 (Update visual settings)

**Requirements Traceability –**

**Priority** – Low

**Preconditions**–

**Post conditions** –

**Actors**–Doctor, Pharmacist

**Extends –** UC28, UC29

**Includes** –

#### UC32 (Update appointment settings)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** –

**Actors**–Doctor

**Extends –** UC28

**Includes** –

#### UC33 (Update reminder settings)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** –

**Actors**–Doctor, Pharmacist

**Extends –** UC28, UC29

**Includes** –

#### UC34 (Save settings)

**Requirements Traceability –**

**Priority** – High

**Preconditions**– Changes have been made.

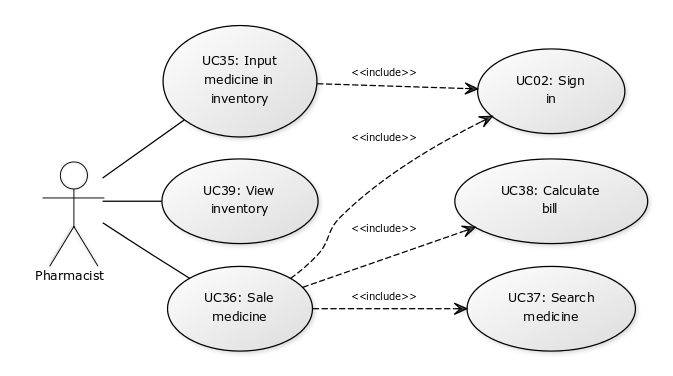
**Post conditions** –

**Actors**–Doctor, Pharmacist

**Extends –**

**Includes** –

### Inventory management subsystem

**Author –** Ishmam Aziz

**Purpose**– Pharmacist will be able to manage the inventory. He will input medicine in inventory and sale medicine to the customer.

**Priority** - High

**Preconditions**–

**Post conditions** –

**Actors**–Pharmacist

**Trigger** – It starts when pharmacist selects inventory.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System provides some options. |
| 2. Pharmacist selects input medicine |  |
|  | 3. System asks user for details. |
| 4. Pharmacist gives details. |  |
|  | 5. System adds the medicine in inventory. |
| 6. Pharmacist selects sale medicine. |  |
|  | 7. System asks for medicine name. |
| 8. Pharmacist gives medicine name. |  |
|  | 9. System shows the location in the storage. |
|  | 10. System asks for amount. |
| 11. Pharmacist gives amount. |  |
|  | 12. System shows total bill. |
| 13. Pharmacist selects view inventory. |  |
|  | 14. System displays the inventory. |
| 15. Pharmacist exits. |  |

* Alternative Flow –

Step 3: Pharmacist is not signed in. System asks pharmacist to sign in and go to step 1.

Step 5: Pharmacist did not give medicine name. System asks pharmacist for medicine name.

Step 5: Pharmacist did not give amount. System asks pharmacist for amount.

Step 5: Pharmacist did not give price. System asks pharmacist for price.

Step 5: Pharmacist did not give location. System asks pharmacist for location.

Step 9: The medicine is not available in the inventory. System gives error message and go to step 1.

Step 12: Not enough medicine in the inventory. System gives error message and go to step 10.

* 1. Exceptions –

**Notes/Issues** –

#### UC35 (Input medicine in inventory)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** –

**Actors**– Pharmacist

**Extends –**

**Includes** – UC02

#### UC36 (Sale medicine)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** –

**Actors**– Pharmacist

**Extends –**

**Includes** – UC02, UC37, UC38

#### UC37 (Search medicine)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** –

**Actors**– Pharmacist

**Extends –**

**Includes** –

#### UC38 (Calculate bill)

**Requirements Traceability –**

**Priority** – Medium

**Preconditions**–

**Post conditions** –

**Actors**–Pharmacist

**Extends –**

**Includes** –

#### UC39 (View inventory)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

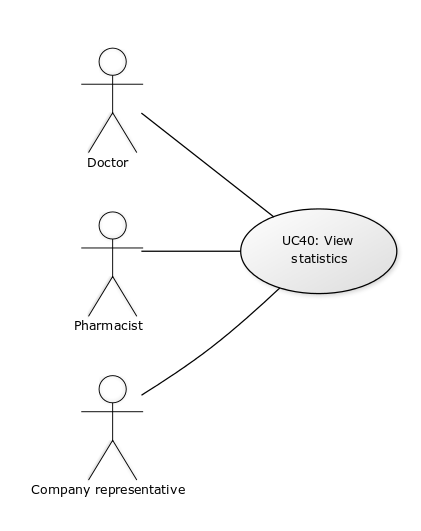
**Post conditions** –

**Actors**– Pharmacist

**Extends –**

**Includes** –

### Statistics subsystem



**Author –** Ishmam Aziz

**Purpose**– Users will be able to analyze and improve their business with the statistical data.

**Priority** - High

**Preconditions**–

**Post conditions** –

**Actors**–Doctor, Pharmacist, Company representative

**Trigger** – It starts when user selects view statistics.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System provides some viewing options. |
| 2. User selects one option. |  |
|  | 3. System displays statistical data accordingly. |
| 4. User exits. |  |

* Alternative Flow –
  1. Exceptions –

**Notes/Issues** –

#### UC40 (View statistics)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

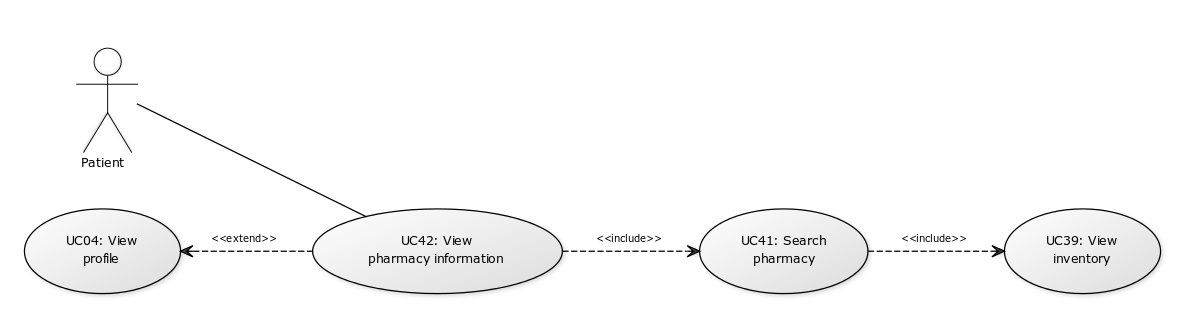
**Post conditions** –

**Actors**– Doctor, Pharmacist, Company representative

**Extends –**

**Includes** –

### Searching pharmacy subsystem

**Author –** Ishmam Aziz

**Purpose**– Patient will be able to find the nearest pharmacy with its location where the desired medicine is available.

**Priority** - High

**Preconditions**–

**Post conditions** –

**Actors**–Patient

**Trigger** – It starts when patient selects search pharmacy.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System asks for medicine name. |
| 2. Patient gives medicine name. |  |
|  | 3. System displays sorted list of pharmacies based on distance from the patient. |
| 4. Patient selects a pharmacy. |  |
|  | 5. System displays information of the pharmacy. |
| 6. User exits. |  |

* Alternative Flow –

Step 3: No result. System gives error message and go to step 1.

* 1. Exceptions –

**Notes/Issues** –

#### UC41 (Search pharmacy)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** –

**Actors**– Patient

**Extends –**

**Includes** – UC39

#### UC42 (View pharmacy information)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

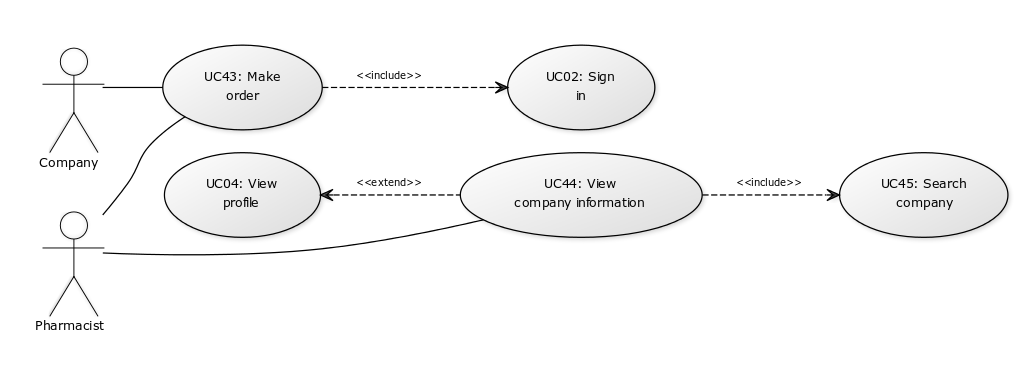
**Post conditions** –

**Actors**– Patient

**Extends –** UC04

**Includes** – UC41

### Order handling subsystem

**Author –** Ishmam Aziz

**Purpose**– Pharmacist will be able to make order through the system.

**Priority** - High

**Preconditions**–

**Post conditions** –

**Actors**–Pharmacist, Company representative

**Trigger** – It starts when pharmacist selects make order.

**Flow of Events**

* 1. Basic Flow –

|  |  |
| --- | --- |
| Actor’s action | System’s action |
|  | 1. System asks for company name. |
| 2. Pharmacist gives company name. |  |
|  | 3. System displays company information. |
|  | 4. System asks for medicine name. |
| 5. Pharmacist gives medicine name. |  |
|  | 6. System asks for amount. |
| 7. Pharmacist gives amount. |  |
|  | 8. System sends order to the company. |
| 9. Pharmacist exits. |  |

* Alternative Flow –

Step 3: No result. System gives error message and go to step 1.

* 1. Exceptions –

**Notes/Issues** –

#### UC43 (Make order)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** –

**Actors**– Pharmacist, Company representative

**Extends –**

**Includes** – UC02

#### UC44 (View company information)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** –

**Actors**– Pharmacist

**Extends –** UC04

**Includes** – UC45

#### UC45 (Search company)

**Requirements Traceability –**

**Priority** – High

**Preconditions**–

**Post conditions** –

**Actors**– Pharmacist

**Extends –**

**Includes** –

# Other Non-functional Requirements

1. System has to be secured.
2. Interface has to be easy to understand.
3. System has to work properly.
4. System has to be reliable.
5. System has to be extendable.
6. System has to be testable and reusable.

## System Requirements

1. Android device. (phone/tab)
2. Database.
3. Internet connection.
4. Printer.
5. Domain.

## Safety and Security Requirements

1. System has to be secured.
2. Communication Security.
3. Security of the communication between the system and server.

The messages should be encrypted for log-in communications, so others cannot get user-name and password from those messages.

1. User Creating Account Security (If a user wants to create an account and the desired user name is occupied, the user should be asked to choose a different user name).
2. Admin Login Account Security.
3. Patients Login Account Security.
4. Doctor Login Account Security.
5. Pharmacy Login Account Security.
6. Medicine Company Login Account security.
7. If Doctor, Patients, Pharmacy, Medicine Company IP address should not be able to log-in for a certain time period after three times of failed log-in attempts.

## Software Quality Attributes

The requirements in this section specify the required reliability, availability, security and maintainability of the software system

**RELIABILITY:** The reliability that the system gives the right result on a search.

Measurements obtained from 1000 searches during testing more than 98% time are reliable.

**AVAILABILITY:**

1. Our app should be available anywhere any time in our country.
2. The application should be connected to the Internet. In order for the application to communicate with the database.
3. The application should be connected to the GPS device. In order for the application to get the users location, the map and to calculate the distance.

**MAINTAINABILITY:** The administrators should maintain easily.

The application should be easy to extend. The code should be written in a way that it favors implementation of new functions. In order for future functions to be implemented easily to the application.

**USABILITY**: This system is so users friendly that it is easy to understand and easy to use, so user should be satisfy maximum time for their need.

**PROBABILITY**: Application portability.

The application should be portable with IOS and Android

Appendix A – Data Dictionary

|  |  |  |  |
| --- | --- | --- | --- |
| **Data elements** | **Description** | **Composition or data type** | **Length** |
| Doctor\_name | Full name of the doctor | String | 50 |
| Doctor\_id | Unique id of the doctor | Integer | 8 |
| Doctor\_specialty | Specialty of the doctor | String | 20 |
| Doctor\_address | Address of the doctor | String | 200 |
| Doctor\_nid | National id of the doctor | String | 50 |
| Pharmacy\_id | Unique trade id of the pharmacy | integer | 20 |
| Pharmacy\_location | Location of the pharmacy | String | 200 |
| Pharmacy\_name | Name of the pharmacy | string | 50 |
| Company\_id | Uniqe trade id of the company | integer | 8 |
| Company\_name | Name of the company | string | 50 |
| User\_id | Unique user id for log in purpose | string | 50 |
| Password | Password for log in purpose | String | 50 |
| Doctor\_email | e-mail address of the doctor | string | 50 |
| Appointment\_id | Unique id for each appointment | integer | 10 |
| Appointment\_date | Date of the appointment | date | dd/mm/yyyy |
| Appointment\_time | Time of the appointment | time | hh:mm |
| Patient\_name | Name of the patient | string | 50 |
| Patient\_problem | Problem of the patient for which the appointment is created | string | 200 |
| Patient\_phone | Phone number of the patient | string | 14 |
| Patient\_email | e-mail address of the patient | string | 50 |
| Patient\_age | Age of the patient | integer | 3 |
| Patient\_blood\_group | Blood group of the patient | string | 3 |
| Schedule\_id | Unique id for each schedule | integer | 8 |
| Schedule\_day | Day of the week for the schedule | string | 3 |
| Schedule\_start\_time | Starting time of the schedule | time | hh/mm |
| Schedule\_end\_time | Finishing time of the schedule | time | hh/mm |
| Schedule\_details | Details of the schedule | string | 500 |
| Prescription\_id | Unique id for each prescription | integer | 8 |
| Prescription\_details | Details of the prescription | string | 1000 |
| Medicine\_barcode | Barcode number of the medicine | integer | 20 |
| Medicine\_name | Name of the medicine | string | 50 |
| Medicine\_price | Unit price of the medicine in taka | integer | 5 |
| Medicine\_amount | Amount of the medicine in inventory | integer | 8 |
| Order\_id | Unique id for each order | integer | 8 |
| Order\_medicine | Barcode of the medicine which has been ordered | integer | 20 |
| Order\_amount | Amount of the medicine ordered | integer | 8 |