

Pharmacy Management System

By

Group 3S

The report submitted for the course of software engineering group project for the degree of

Bachelor of Science in

Computer Science (B.Sc. computer science)

Name of the Supervisor

Dr. A. Ramanan

(Senior lecturer)



B.Sc.in Computer Science-2014/2015

Department of Computer Science

Faculty of Science

University of Jaffna.

Sri Lanka.

Executive Summary

Our team is group 3S,

The members of our team are:

S.Jathurshan (2014/CSC/003)

M.Kaushik (2014/CSC/010)

S.Thilakarathna (2014/CSC/030)

C.Lahiru (2014/CSC/034)

A.Tharshikan (2014/CSC/036)

Group mentor:

Senior lecturer Dr.A.Ramanan

We are undergraduate students and following Computer Science at Faculty of Science of University of Jaffna. This report is contained the detailed information of our second year group project. The title of our project is “Pharmacy Management Application”. The aim of our project to provide a good standalone software for medicine stock and billing management with efficient manner.

Why we are choosing this idea to our project? There are many pharmacies surrounding us but there are few pharmacies have the application to manage the stocks and billing. Unfortunately, many pharmacies that they do not have the complete application to manage the pharmacy, and another thing is the keeping their medicine records and sales details manually with large number of files, books and hard copies with large space. If the pharmacy people want to know any particular details of medicine, they must search hole files of the form of hard copy. This method is wasting their own time and, it will make them annoyance and frustrated. And also each medicines have unique brand name, generic name, and different dosages, these kind of information are hard to find manually.

This software saving the human time, we can store the medicine details in proper way, billing and billing records can be secured, and application will give the automatic alert and notifications about the expiry date of the medicine and low stock of medicine.

Main advantages of our application are:

- Reducing human effort, human errors, time and make easy to manage huge database.
- Solve the conflict during the stock entry.
- Easy to maintain the stock details and sales details.
- Do not need larger space comparing to the physical records.
- Secure

The result of our project is complete standalone application for pharmacy management. The application is password protected and all the password is encrypted by md5 hashing algorithm. There are two panels “Admin” and “User”. Admin can access fields of Medicine, Staff, Supplier, Sales, change password. User has restricted access on the application (mainly billing section).

“ATLANTA Pharmacy” is involving in our project, because we implemented the standalone application for the “ATLANTA Pharmacy”. They liked to work with us and they are happy to use our product. We got the real data from them.

We designed and implemented our product as a user friendly. We can access easily. In future, we have a plan to develop our application with barcode reader. It will help to our customers to handle our application easily.

‘Group 3S’ have many challenges during the project period. In our initial stage we do not have a good client because of these we could not get requirements on time, after that we used C# as a programming language to implement our application. This programming language is new to our group members. We learned C# in short time of period for this purpose. Finally, group 3S produce their end product.

Declaration of Authorship

Hereby we, Mr.S.Jathurshan, Mr.M.Kausik, Mr.S.Thilakarathna, Mr.C.Lahiru, A.Tharshikan are members of group 3S certify that, our project is title as “Pharmacy Management Application” and this is completely our own work. The project of “Pharmacy Management Application” was done by group 3S.

This project was not submitted in identical or similar form to another degree program or same degree program.

Date

.....
Mr.S.Jathurshan
(2014/CSC/003)

.....
Mr.M.Kausik
(2014/ CSC /010)

.....
Mr.S.Thilakarathna
(2014/ CSC /030)

.....
Mr.C.Lahiru
(2014/ CSC /034)

.....
A.Tharshikan
(2014/ CSC /036)

.....
Dr.A.Ramanan
Supervisor,
Department of Computer Science,
Faculty of Science.

.....
Dr.K.Thabotharan
Head of the Department,
Department of Computer Science,
Faculty of Science.

Contribution of each member

Mr.S.Jathurshan

He gave the idea of “Pharmacy Management Application” for the group project of software engineering course unit. He designed the whole interface of this application, arranging the button functions of each interface based on functional requirements, took part application testing and he has learned C# programming language in short time of period for this project work. This project report is prepared by him.

He has coded the following interfaces using C# programming language:

- Login interface
- Admin interface
- User interface
- Change password interface
- Add Medicine interface
- Add Staff interface
- Add Supplier interface
- Modify Medicine Details interface
- Modify Staff Details interface
- Modify Supplier Details interface
- Retails Sales Details interface (billing part of the application)

And also coded for validation of Email Id, NIC No, Contact number, Numerical and character validation of textboxes.

Mr.M.Kausik

He has learned C# programming language in short time of period and he has coded the following interfaces of the application:

- Login interface
- Admin interface
- User interface
- Change password interface
- Add Medicine interface
- Add Staff interface
- Add Supplier interface
- Modify Medicine Details interface
- Modify Staff Details interface
- Modify Supplier Details interface
- Retails Sales Details interface (billing part of the application)
- Delete Medicine Details interface
- Delete Staff Details interface
- Delete Supplier Details interface
- View Purchase interface
- View Expiry Date interface
- View Stock interface
- View Supplied Details interface
- View Staff Details interface
- View Sales interface

He has done hashing for password encryption and decryption using md5 hashing algorithm and took part of application testing.

Mr.S.Thilakarathna & Mr.A.Tharshikan

He collected the information from the pharmacy people

- How they buy medicines from the medical agents,
- What kind of medicines that they sell to the customers?
- Details of medicines.
- How they keep the record of new stock, sales, orders and purchases
- How they remove the expiry stocks from the record and return to the agent.

And also he collected some medicine labels and containers and details of manual billing application. He got the functional and non-functional requirements from them.

Mr.C.Lahiru

He good in SQL database and help to connect the SQL server to visual studio platform. He helped to retrieve the data from database with SQL codes, he created database, tables and he entered the data to the tables.

Consent of the client

<p>මෙයි අංකය සෙවා තුව. My Number</p> <p>මෙයි අංකය තුමතු තුව. Your Number</p> <p>දුරකථන තොගලපෝ Telephone</p>	 <p>යාපනය විශ්වවිද්‍යාලය, ශ්‍රී ලංකාව යාම්ප්පාණප් පල්කලෙකක්මකම, මහත්මය මූල්‍ය UNIVERSITY OF JAFFNA, SRI LANKA</p>	<p>රු.පෙ.අංකය 57, තිරුනේල්වේලු, යාපනය.</p> <p>ත.පෙ.න් 57, තිරුනේල්වේලු, යාපනය.</p> <p>P.O. Box 57, Thirunelveli, Jaffna.</p>										
<p>021-221-8194</p> <p>08/02/2017</p>												
<p>To:</p> <p>Dr. T. Sethirabalan, ATLANTA Pharmacy, 54 Uduvil Road, Maruthanarmadam.</p>												
<p>Dear Sir,</p> <p>Requesting Permission to study the ATLANTA Pharmacy System to carry out a Student Project</p> <p>The following five undergraduate students following Computer Science at this Department would like to do a project entitled “ATLANTA Pharmacy Management System” as part of their Second year course CSC241SC2 - Group Project. They would like to study how the day to day operations are managed at the RVG Pharmacy; therefore they can design their system accordingly. Therefore please be good enough to grant permission to carry out a study about the to day operations at your establishment, to enable them to complete their project successfully.</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">Mr. M. Kausik</td> <td style="width: 50%;">2014/CSC/010</td> </tr> <tr> <td>Mr. S. Jathurshan</td> <td>2014/CSC/003</td> </tr> <tr> <td>Mr. Sahan Tilekaratne</td> <td>2014/CSC/030</td> </tr> <tr> <td>Mr. Chalaka Lahiru</td> <td>2014/CSC/034</td> </tr> <tr> <td>Ms. A. Tharsikan</td> <td>2014/CSC/036</td> </tr> </table>			Mr. M. Kausik	2014/CSC/010	Mr. S. Jathurshan	2014/CSC/003	Mr. Sahan Tilekaratne	2014/CSC/030	Mr. Chalaka Lahiru	2014/CSC/034	Ms. A. Tharsikan	2014/CSC/036
Mr. M. Kausik	2014/CSC/010											
Mr. S. Jathurshan	2014/CSC/003											
Mr. Sahan Tilekaratne	2014/CSC/030											
Mr. Chalaka Lahiru	2014/CSC/034											
Ms. A. Tharsikan	2014/CSC/036											
<p>Thank you, Yours Sincerely,</p> <p><i>K. Thabotharan</i></p> <p>Dr. K. Thabotharan, Head, Department of Computer Science, University of Jaffna, Jaffna.</p>												
<p>HEAD Department of Computer Science University of Jaffna</p>												

To Dr. T. Sheethalabalan.

Head

Department of Computer Science

09/02/17

Thank you for your studies to
do his project, I removed
the tie project were us.
Yours OI.

Thiriyag

Atlanta Hospital
No.54, Uduvil Road,
Maruthanarmadam,
Jaffna
Tel.021 2242299

Dr.T.Sheetalabalan (MD)
SLMC Reg. No. 25246
Jaffna - Sri Lanka

Letter of approval of the final software

Comments from supervisor

Table of content

List of figures

List of tables

Introduction

The project titled “Pharmacy Management Application”, this is a standalone application for medical and pharmacy information.

This is a user friendly and interactive application for managing both stock and billing which helps in maintain the records of the medicine, the users and store details and also reduce the work of searching the stock details. The main aim of this application is to apply technology is supporting the pharmacist during the data management and reduce human effort on searching and automation of the billing.

The project has been developed on the basis of “stock managing” and it’s “billing process” being presently used in the medical stores for storing and retrieving the available information in the store. The application has auto notification features for low stocks and expiry information of stocks. The expiry alert will be given in current date and the low stock alert will be given, when the particular stock count reach 50. Each medicines and other items in the pharmacy are labelled by unique Id, this method is useful for avoiding the problem during the stock details entry and it is easy for searching.

The user has to get his username and password from the admin by providing the name, address, phone number, Id proof and role. They can get the access to the application. Without the username and password, he cannot get access to the application. The password is encrypted by hashing algorithm(md5).

In this application the users will be specified by their role in the pharmacy and the user can access only few functions (view sales, view stock, view purchase, and retail sales (billing section)). And the admin can access and manage all the features of the application.

Purpose of the project

The main purpose of the project is replacing the manually maintained record of sales, purchase, billing, stock, staffs and suppliers by automated application. Another thing is to give a good application for pharmacist to avoid their work pressure during the billing period, no one cannot cheat the administration and owner and many small pharmacies do not have application like this, one of the major problem is their economic status, but they wish to use the application for maintaining their pharmacy details.

Scope of the project

We are introducing the software “Pharmacy Management System” which is implemented for “ATLANTA Pharmacy”. The application is designed simply and easy to access. The major function of the application is maintaining medicine details and billing section and dealing the supplier information. Supplier details are related with the medicine details, by these return of medicines is easy to handle.

Owner(admin), manager, staff (person who are in the counter) are the end users of this application. Owner can access all the functionalities of the application without any restrictions. But, the manager and staffs can access with limited restrictions. Only admin can make any changes to the data. This will help to keep their details secretly.

We hope that, most of the pharmacist like to use our application. We have decided to include the barcode system in future to this application.

Overview of the document

In documentation overview, we have included the problem, challenges, and technics used in our project.

- First stage: The technics used for requirements gathering, and problems of getting the functional and non-functional requirements, and what are the challenges we faced to get the requirements.
- Second stage: Idea of Software design, software design approach, we illustrated with the diagram to the application, data model and challenges during the designing work.
- Third stage: Implementation work and strategy, tools and techniques and justification of special features.

And also included the references, user and development documentation.

Requirement Engineering

Our project “Pharmacy Management System” needs requirements to design and implementation of application. Initially we got the requirements from “RVG Pharmacy” and we started our designing work to the application with their requirement. Unfortunately, we could not work with them. But, we were implementing our application with their requirements and we did not have a pharmacy for our project.

After 2 months our team were connected with “ATLANTA Pharmacy”. We showed our application until we have implemented. The Administration of “ATLANTA Pharmacy” were happy to see that. And they gave some additional requirements with few corrections.

Finally, we implemented our standalone application, with requirements which we were getting from the pharmacist.

Techniques used for requirements gathering

First we approached the owner of the pharmacy with the question “what are the problems you are facing in your business in daily basis?”. Our team described them about our idea for them, how to manage their conflicts and problems of pharmacy management and we told them the basic functionality of our application. Addition to that we explained to importance of the system to the pharmacy. After that we requested for the requirements of them. They listed out their requirements, it was easy to reach our complete application. we needed the medicine details because, we want real data to the process. Our client gave us some labels, boxes and container of medicines and gave the approval to the project.

Functional requirements

- Generate report weekly on information about the drugs and it exports the information as output document.
- Store the detailed information about each medicine including actual name formula of medicine.
- When the user searches the item on search bar the related things were displayed in the screen and can select the actual item that the user needs and searching must be multiple searching.
- Give an alert when the medicine outdated.
- Give an alert for the user when medicine stock is low.
- Automated billing system and prepare bill for the sales.
- Managing the details of medicine, staff, supplier and sales.
- Date and time display.

Sample details for medicine for real data.



PHARMACY MANAGEMENT SYSTEM

41) LIDOCAINE HYDROCHLORIDE JELLY USP 30ml 0ct/15 Mar/18	Analgesic Jelly
42) SALBUTAMOL Pressurised Inhaler 100mcg 20 doses 07/13 07/16	SALBUTAMOL
43) Dymetazoline Hydrochloride 0.025% 10ml Jun/14 May/17	Gouttes Nasales
44) Clonex G (Antimicrobrial Cream) 30gm 07/14 06/17	Clonex G Cream
45) Fidra Cream (Adapalene) 0.1% 10gm May/14 Apr/17	Fidra cream
46) Painzeel 30g	Methyl Salicylate Ointment BP
47) Aqueous Cream with Glycerine 50g 15.06 17.06	D - Soft cream
48) Elamar cream 20g Dec/2015 Nov/2018	-
49) Benzyl Benzoate 25% w/v 100ml 08/05 12/04/17	-
50) Asithalin (Respirator Solution) 15ml 12/01/14 11/01/16	Sulbutamol Respirator Soln
51) Mometasone Furoate Cream USP 5g 03/05 02/017	Mometasone cream
52) Miconazole Cream BP 15g 07/01/15 06/01/17	Unimico cream
53) Dymetazoline Hydrochloride Nasal Soln BP 0.025% 10ml 11/2014 07/2017	Nasal solution (PRESERVATIVE)
54) Silver Sulfadiazine Cream USP 25g 12/01/15 11/01/18	Silver sulfadiazine cream
55) KETOPROFEN 15g 1/7/017 1/7/2017	Fastnum gel
56) Antifongique, anti-inflammatoire 15g Dec/14 Nov/17	Transfip-co cream
57) Terbinafine Hydrochloride Cream 10g 1/feb/1014 31/jan/117	TERBINAFINE cream
58) SILVIRIN 20g 13/11/14 (2/11/17	Silver Sulfadiazine zinc cream USP

Name	Ref	Ex
SPMC		
1) Benzhexol Tablets B.P. 10x10 tablets 2mg	Aug/14	Nov/16
2) B-HIST - 8 10x10 tablets 8mg	Apr/2015	Mar/2017
3) Walcott 10 strips of 0.05mg 29/06/2015 28/06/2016		
4) Gastro-resistant Bismacetyl 10x10 tablets 5mg	18/2015	10/2018
5) Myoxine Antihistamine Tablets IP 10x10 Tablets 10mg	03/2016	02/2019
6) Bromhexine Hydrochloride Tablets BP 10x10 8mg	16/2013	01/2017
7) Diazepam tablets B.P. 100tablets 5mg 03/2016	02/2019	Diazepam
8) Domperidone Tablets BP 0.05mg 03/2015	02/2018	Domperidone
9) " 1000Tablets 0.05mg 03/2015 03/2017	"	
10) Domperidone Tablets BP 10x10 5mg 03/2015 09/2018		Domperidone
11) Gastroresistant Domperidone Tablets BP 10x10 25mg Mar/2015 Feb/2017		
12) " 10x10 5mg Jul/2015 Jun/2018	"	
13) Slow Domperidone Tablets BP 10 strips of 10mg 04/2014 03/2017		
(Sustained Release) tablet each		
14) ADDELAM-50 TAB 80x100	09/2015	08/2018
15) Domperidone Tablets BP 2x5 Blister of 10 tablets each 10mg Mar/2016 Feb/2018		Domperidone
16) Domperidone Tablets 2x50 tablets 10mg 09/2015 09/2019		"
17) " 10x10 10 mg Jan/2015 Dec/2018		"
18) Domperidone Capsules BP 10x10 capsules 100mg 07/2015 06/2017		Doxycycline
19)		

E	1) Esomeprazole Tablets 10x10tablets 20mg 09/2015 06/2017	Esomeprazole
2)	" 10x10 20mg April 2015 Mar/17 "	
21)	" 10x10 20mg 11/15 June 16 "	
22) Standard Salvee 300 10.0mg		Etoffeline
Tablets of Etoffeline and Thiophylline (Etoffeline 10mg) Thiophylline 30mg	09/15 08/18	
23)		
24) Glipizide Tablets BP 10x10 80mg	16/2015 16/2016	Glipizide
25) Glipizide Tablets BP 10x10 80mg	03/2015 02/2018	Glipizide
26) " 80mg 04/2014 03/2017		
27) Glipizide Tablets BP 10x10 80mg 06/01/14 07/2017		Glipizide
28) " 10 Blister Strips of 10 tablets 80mg 02/14 Nov/16 "		
29) " 10 Blister Strips of 8 tablets 80mg 11/2015 10/2017 "		
30) " 10 Blister Strips of 8 tablets 80mg 06/2015 10/2017 "		
Amberol (cream) & Jamps		
31) Balsamethasone Valerate Cream BP 15g 12/2015 11/2017	Balsamethasone.	
32) Clofazimil Cream BP 0.05% w/v 0.5g 11/15 10/18	Clofazimil Cream	
33) " 0.05% w/v 0.5g 09/15 10/17	Balsamethasone.	
34) Ciprofloxacin Hydrochloride Solution 5ml 12/15 10/17	Ciprofloxacin (Eye/Ear Jamps)	
35) Aqueous Cream (Emollient Moisturiser 100g 15.06 17.06	Cetrimide Shampoo Aqueous cream	
36) " 100g 09/15 10/17 betanore.		
37) GENTAMICIN sulphate & Dexamethasone 5ml 0.05% w/v sodium phosphate 0.3% w/v 0.1% w/v	Beglomin Lotion	
(Eye/Ear Jamps)		
38) Beclomethasone and Miconazole Lotion 10ml 05/15 09/17	Beglomin Lotion	
39) Gentamycin Eye/Ear Drops 0.3% w/v 10x5ml 09/15 08/18	GIARACIN	
40) Ciprofloxacin Eye/Ear Jamps 5ml 08/15 07/18	CIPLOX.	

41) LIDOCAINE HYDROCHLORIDE JELLY USP 30ml 0ct/15 Mar/18	Analgesic Jelly
42) SALBUTAMOL Pressurised Inhaler 100mcg 20 doses 07/13 07/16	SALBUTAMOL
43) Dymetazoline Hydrochloride 0.025% 10ml Jun/14 May/17	Gouttes Nasales
44) Clonex G (Antimicrobrial Cream) 30gm 07/14 06/17	Clonex G Cream
45) Fidra Cream (Adapalene) 0.1% 10gm May/14 Apr/17	Fidra cream
46) Painzeel 30g	Methyl Salicylate Ointment BP
47) Aqueous Cream with Glycerine 50g 15.06 17.06	D - Soft cream
48) Elamar cream 20g Dec/2015 Nov/2018	-
49) Benzyl Benzoate 25% w/v 100ml 08/05 12/04/17	-
50) Asithalin (Respirator Solution) 15ml 12/01/14 11/01/16	Sulbutamol Respirator Soln
51) Mometasone Furoate Cream USP 5g 03/05 02/017	Mometasone cream
52) Miconazole Cream BP 15g 07/01/15 06/01/17	Unimico cream
53) Dymetazoline Hydrochloride Nasal Soln BP 0.025% 10ml 11/2014 07/2017	Nasal solution (PRESERVATIVE)
54) Silver Sulfadiazine Cream USP 25g 12/01/15 11/01/18	Silver sulfadiazine zinc cream USP
55) KETOPROFEN 15g 1/7/017 1/7/2017	Fastnum gel
56) Antifongique, anti-inflammatoire 15g Dec/14 Nov/17	Transfip-co cream
57) Terbinafine Hydrochloride Cream 10g 1/feb/1014 31/jan/117	TERBINAFINE cream
58) SILVIRIN 20g 13/11/14 (2/11/17	Silver Sulfadiazine zinc cream USP

PHARMACY MANAGEMENT SYSTEM

1) Metformine Tablets IP	250mg	09/14	08/16	Metformex
2) Metoclopramide Tablets BP	10mg	07/15	06/18	Metoclopramide
3) Metronidazole Tablets BP	500mg	05/04	04/018	Atmetronidazole
4) Amodia 400				
- Mefenidazole tablet BP 400	400mg	10/13	9/18	
5) Mifeprex				
- Levonorgestrel	0.05mg	15/7/2015	25/6/2020	
- Ethynodiol	0.03mg			
6) Nifedipine Retard	20mg	07/14	06/17	
7) Proprianol tablets	40mg	07/2015	06/2017	
8) GABA Met 75				
- Pregabalin capsules	75mg	04/2015	03/2017	
9) Prednisolone tablets	5mg	12/2014	11/2017	
10) Ranitidine tablets	150mg	09/15	05/18	
11) Benfotiamine	800mg	07/2015	06/2017	
12) Recoplus	280mg	1/2016	06/2017	
- Multivitamin B Complex				
- Folic Acid				
- Vitamin C				
- Calcium				
- Zinc				
(13) Gas-mod.	40mg	17/06/2014	07/06/2017	
- simethicone				
Dose:- Infant 0.3ml (1/2 dropperful) 3-4 times daily				
14) Calcium lactate	800mg	06/2015	03/2017	
15) Livitene syrup	200ml	16/2015	06/2017	
- (Vit B-complex)				
- Cysteine & Zinc				
16) Lysofit 4mg	300mg	02/2016	7/2017	
17) Vitamin B tablets	29mg	05/2015	4/18	
18) Anemidox	700mg	11/2015	1/2017	
- Capsules of				
- Vitamins with				
- Iron				

25) Delonil	Sabutamol			
- Expectorant	Sulfite, Bromhexine, Hcl, Guaifenesin	06	06-2015 - 06-2017	100mL
- & Menthyl Syrup				
26) Paediatric Paracetamol	PARAKID Syringes	120	1-6+275	Second
- Oral Solution BP			6-12-475	
27) Chlorphenamine	Oral solution		12-2015 - 11-2017	75mL
- BP				
28) Sabutamol			12-2015 - 02-2018	75mL
- Oral Solution BP				
29) Brufen	Chlorphenamine bis-glyc-75	08-2015 - 07-2016	750mL	
- Expectorant	Expectorant Child 6-7-75			

* Celecoxib 100	100mg	MAY15	MAY17	Celecoxib
* Cefuroxime Axetil	500mg	OCT15	SEP17	Cefuroxime Axetil
colour - Titanium Dioxide				
* Cefixime 200mg		MAR15	FEB18	Cefixime Trihydrate
colour - Titanium Dioxide	(IWA-FIXIM-200)			
* Cefixime Dispersible	100mg	FEB16	JAN19	Cefixime Trihydrate
THEOFIX				
* Cefalexin Capsules	500mg	MAY15	APR17	Cefalexin monohydrate
* Cefalexin	250mg	JUL14	JUN17	Cefalexin monohydrate
T THEOFLEX-250				
* Carbamazepine	200mg	DEC14	NOV19	Carbamazepine
ZEPTEL 200				

As directed by physician				
	MFD		EXP	
* Clapidogetel	75mg	JAN15	JAN17	Clapidogetel Bisulfate
* Ciproquin-500	500mg	SEPI5	AUG18	Ciprofloxacin Hydrochloride
colour - Titanium Dioxide				
* Ciprofloxacin Forte	500mg	MAY15	APR18	Ciprofloxacin Hydrochloride
colour - Titanium Dioxide				
* Ciprofloxacin	500mg	DEC15	NOV17	" "
PRONAT-500				
* Ciprofloxacin	250mg	MAR14	FEB17	" "
UltraFlox-250				
* Chlropheniramine	4mg	APR14	MAR17	Chlropheniramine Maleate
CHLPHEN-4				
* CINNARIZINE	20mg	JUN15	MAY18	" "
VERTIGEN				
* Cetirizine	10mg	SEPI5	Aug18	Cetirizine Hydrochloride
CETBARD				
colour - Titanium Dioxide				
* Cloxacillin	250mg	AUG14	NOV16	Cloxacillin Sodium
* Cetirizine	10mg	MAY15	APR16	Cetirizine Dihydrochloride
* Celecoxib	200mg	OCT15	OCT18	Celecoxib
SELECAP-200				
* CELECOXIB	200mg	JUN14	MAY16	CELECOXIB COL-CIBRA
ZINAXIN				
* Cefuroxime Axetil	250mg	JUL15	JUN18	Cefuroxime

PHARMACY MANAGEMENT SYSTEM

1) Peridone	Damperidone Suspension,	1/2 to 1 TS 3-4 Times per day	9-2014	08-2016	Quinidine 100 mg Yellow.	-
2) Cloxacillin.			July-2015	Jun-2017	Sunset yellow 100	
3) Sodium						
3) Mefenamic Acid	Mefenac-A		06-2015	05-2017	4 100	
4) D-Oral	Prednisolone		Sep-2015	Sep-2017	60	
5) Amoxicillin Oral			July-2015	Jun-2017	Sunset yellow 100	
6) Cephalexin Oral	Cephalexin.		9-2015	2-2017	Sunset yellow 100 FCF.	
7) Moxycycline Oral Suspension	Moxycycline Oral Suspension		Dec-2015	Nov-2017	Turquoise 170	
8) Broncodil	Salbutamol (2-6) 1/2-1 TS		12-2015	11-2017		
Bronchodilator	Oral Solution (G-2) 1-7 TS				100	
9) Litis	Litis Plus Syrup	Mucolitic Bronchodilator	08-2015	07-2018	Sunset yellow 100 FCF	
10) Paediatric Paracetamol Oral Solution	Kelvin Syrup	15mg/5ml 60mg/5ml	08-2015	07-2018	Ponceau 60	
11) Terbutaline Sulphate Bromhexine Hcl	Exbro		05-2013	Apr-2016	-	100
Ciautafenestin & menthol Syr.						
12) Cetirizine Hydrochloride oral Syr.	Zocet		10-2015	03-2018	Ponceau 60	
13) Prolonged release Adiflam.	Diclofenac tab					
14) Decofed-X	Expectorant	Ant-10ml 10-2015	07-2018			
		Mucolitic 16-5ml				
		Bronchodilato 12ml				
15) Mafero	Riferonac acid		02-2016	01-2018	-	60
16) Cefalexin oral Suspension	Cefalexin Oral Dry Syrup		08-2015	07-2017	-	100
17) Paediatric Paracetamol Oral Suspension	Anodyne-A		06-2012	05-2018	-	100
18) Cefalexin Oral Suspension RP	Cephast		Feb-2015	Jan-2017	Sunset yellow FCF	100
19) Alendronate Mg 104	Magene A-4-240	104g	08-2015	05-2018	-	200ml
20) Cof-kid	Bronchodilator		02-2015	01-2017	Sunset yellow FCF	100ml
	Expectorant					
	Mucolitic					
	Oral Liquid					
21) Cephalexin Oral Suspension	Safex		Oct-2015	Sep-2017	Turquoise & Brilliant Blue FCF	100ml
22) Axekt.	Paracetamol/120		03-2015	03-2018	-	100ml.
23) Throphylax And Enalaflexin			01-2016	12-2017	-	50ml
	Oral Solution					
	disp.					
24) Fred Kid	Prednisone		10-2014	09-2017	Carmesine 60ml.	
	Syrup					

Medicinal Term	MFD	EXP	Product form Country name
* Nifedipine Extended Release Tablets 20mg	Feb 15	JAN 18	Nefix™ 20ER
* Norgoxane Tablets USP 40mg	June 16	May 2018	Normac
* Nasotriptan Tablets USP 5mg	Dec 15	Nov 18	NCNorm-e
* Pantoprazole Sodium delayed release tablets USP 40mg magis-		APR 17	Psovac 40
* Pantoprazole tablets 20mg	Jan 15	May 17	Pantacid 20
* Paracetamol Tablets BP 500mg	Aug 15	July 19	Regimel
* Thepar-500	Nov 15	Oct 18	Angevic
* Calcium Citrate Zinc 119mg + VIT D ₃ & Magnesium	Sep 15	Feb 17	Maxical
* Sildenafil Citrate Tablets 50mg	Dec 14	Nov 17	Endora-50
* Colusommine, MSA Multi Vitamin & Mineral Tablets 250mg	Nov 16	Sep 18	NECAZ

# UNIFLAME PHARMA- -TICAL LABORATORIES Omlek* 20	20mg	June/2015	May/2017	Medical use Gantek * 20 Omeprazole Delayed-Release Capsules USP
# MOVCOL	—	28/12/2015	28/12/2017	(Polyethylene glycol 3350 + Sodium chloride + Sodium bicarbonate + Potassium chloride)
# Pantoprazole Tablets Pazel	40mg	12/2015	11/2018	Pantoprazole Delayed Tablets
# —	20mg	09/2015	09/2018	OMEPRAZOLE CAPSULES
# Protag-110	40mg	05/2015	04/2017	Pantoprazole Sodium delayed release tablets USP
# PANFRED 20	20mg	June/2015	May/2017	
# OMMEED-20	20mg	12/2015	11/2018	Gastro - Resistant Omeprazole Capsules BP 20mg
# Neoprox® Tablets	500mg	09/2012	08/2015	Neoprox® Tablets 500mg [Naproxen] Tablets USP.
# THEOFIX-100 DT	100mg	01/05/2013	30/04/2016	Cefixime Dispersible Tablets 100mg
# TAMIGEN-S EYE/EAR DROPS	5ml	June/2014	May/2016	Gentamicin Sulphate & Dexamethasone, Sodium Phosphate EYE/EAR DROPS
# EAPOL	20mg	AT		

PHARMACY MANAGEMENT SYSTEM

# CAPTOL	25mg	May/2018	Apr/2016	Captopril Tablets BP
# Beclomethasone Lotion (Emol)	02/2018	01/2016		Beclomethasone and Miconazole Lotion.
# — (100mcg)	10/2015	09/2016		Levthyroxine Tablets B.P. 100mcg
# — 0.5mg	01/2014	08/2016		Dexamethasone Tablets BP 0.5mg.
# — 5mg	07/2018	06/2016		Cetirizine Tablets B.P. 5mg
# Olmoxin 200mg	01/2014	12/2016		Aldendazole USP
# — 2mg	10/2013	09/2015		Diazepam Tablets B.P. 2mg
# Salbutamol (Emol)	12/2015	02/2018		Oral Solution B.P.
# Paracet 120 (Emol)	Jun/2015	May/2018		Paediatric Paracetamol Oral Solution BP
# Pintan Expectorant (Emol)	03/2015	09/2016		Chlorpheniramine Expectorant Syrup.
# Cephalexin dry syrup (Emol) 200mg	09/2015	02/2017		Cephalexin Oral Suspension IP 125mg/5ml
# PERFDONE (Emol) (mg)	07/2018	08/2016		Domperidone Suspension
color: Quinoline Yellow (WS)				
# Mefen-P Suspension	08/2015	04/2017		Mefenamic Acid Suspension

# Amitriptyline O.S.	0.05mg	09/2015	03/2018	
Colour: Quinoline Yellow WS & Titanium Dioxide BP.				
# Amoxicillin Capsule	500mg	07/2015	06/2017	
	0.50mg	03/2014	03/2018	
# Amoxicillin	600mg	01/2010	31/12/2017	
Colour: Clavulanic Acid				
# Aclaracal G.S	62.5mg	Sept/2015	Aug/2017	
Colour: Amoxycycline				
# Atenolol tablets	50mg	09/2013	04/2016	
Colour: Atenolol				
# ATORVASTATIN CALCIUM Tablets	10mg	01/2015	09/2018	
Colour: Calcium				
# Laotidine 20 mg	20mg	06/2015	05/2018	
Colour: Brilliant Blue FCF				
# Ferric sulphate 200mg	200mg	01/2015	01/2018	
Colour: Ponceau AP				
# Fluconazole 150mg	150mg	09/2015	09/2018	
Colour: Fluconazole				
# Flunarizine tablets 5mg	5mg	05/15	09/14/8	
Colour: Flunarizine				
# DONAPRIZINE-S				

# Fluoxetine Capsule	20mg	09/2015	09/2018	
# FOLIC ACID Tablets	1mg	01/2014	July/2017	
CNI - INGAFO-1				
BN FOLEG - 1	mg	02/2015	July/2017	
Folic Acid tablets				
# FUDROSEMIDE Tablets (Frusemide tablets)	10mg	11/2015	10/2017	
SANMAG	675mg	01/sep/15	01/sep/18	
Magnesium Trisilicate	(Quinone)			
Aluminium hydroxide				
Activated charcoal				
Oral suspension				
BN				
Desage: 1-20 measuring spoonful				
Aluminate-Magnesia	120mg			
1 Simezthione	Orange Flavour	clr.: Sweet Yellow FCF		
Oral suspension				
CN				
Antifug-D.S	Mint Cardamom	Quinoline Yellow WS		
Ec	Erythrosine			
Anbucilladol	10.5mg/5ml	6/7/2015	24/2017	
suspension B.P.	10ml			
Amoxicillin & Potassium for oral suspension				
CN				
Amoxycillin	100mg	03/2014	07/2016	
Amoxycillin & Potassium for oral suspension				
CN				
INDICLAV 150	150mg	02/2016	01/2018	
Amoxicillin and clavulante				
Potassium for oral suspension	100mg	01/09/2015	31/03/2017	
CN				
The oclav				

# Acrelofenac Tablets	100mg	06/2015	05/2017	
Colour: Lake Tetrazine, Lake Brilliant Blue FCF & Titanium Dioxide BP				
Dosage: As directed by the physician.				
# Aciclovir Tablets	200mg	01/15	10X10 Tablets	
Colour: Aciclovir				
CN Albenent	200mg	Oct/2014	Sept/2017	
Albendazole tablet				
Single dose treatment for worm eradication				
# Alphadol	0.25mcg	01/2015	01/2017	
Alpha-facalcidol Capsules				
Colour: Ponceau 4R suppository				
Aldren 70	70mg	01/17	10/17	
Alendronate				
Sodium tablets				
Alpha D2-Bigual	0.05mg	01/2013	01/2014	
Ex: exclusively according to the doctor's instructions				
# Sedalam	0.5mg	11/2014	11/2016	
Alprazolam amp. tablets				
CN Amitab-10	10mg	05/2015	04/2018	
Amitriptyline Tablets				
Colour: Quinoline Yellow WS & Titanium Dioxide BP				
# Manoest 0.5	0.5mg	01/2015	01/2018	
Alprazolam tablets				

Metformin tablets	850 mg	18/17	18/16
Theone f - 850	850mg	01/07/2014	30/06/2017
Metformin Hydrochloride Tablets	500 mg	03/08/2015	09/08/2018
PREGNANCY CLASS B	8mg	JAN 13	JAN 18
GSK Methylprednisolone 8mg Tablets			
Zopan DS	500mg	03/2015	05/2018.
(C Metformin Acid)			
Me洛xicam Tablets		11/2013	20/12/16
C N Omeprazole-15			
Meferenamic Acid	850mg	08/2014	07/2017.
Loratadine tablets	10mg	01/2015	10/2018
C N → Hylotac Lusartan Potassium Tablets	50mg	01/2015	12/2017
COPAMGD	10mg	08/2015	07/2018.
Loratadine tablets			
C N Osartil	50mg	May/15	APR/16
Losartan Potassium tablets			
Coperamide Hydrochloride capsules	0mg	01/2015	02/2018
Levothyroxine tablets	100mcg	10/2015	09/2016

Non-functional requirements

- It has the instruction menu to explain users about how to use it.
- It is available based on the user needs (work properly, do sales and purchases efficiently including safe data management of the pharmacy)
- Pharmacy system is password protected to control the system in proper way.
- Data are protected and controlled by the administrator.
- The owner(admin) can handle all the data without any restrictions.
- The users have restrictions to handle the application

Domain requirements

These requirements that reflect the characteristic and features of the “Pharmacy Management System” and this is new functional requirements, constraints on existing requirements or define some specific requirements.

- Our system should be safe, unbreakable and password protected and passwords are stored as encrypted form.
- Data should be integrity.
- System should allow authenticated people.
- Stock, sales, purchase and billing report should be provided to the admin.

Modules of the system

- Admin module
- User module
- Manage medicine module
- Manage staff module
- Manage supplier module
- Manage sales module
- Change password module.

Admin can access all the modules and user only access sales module with some restrictions.

Hardware Requirement

Minimum hardware and Software Requirements for System Installation

Hardware:

- CPU - Pentium 4 2.8 GHz
- RAM - 4GB or above
- 40 GB or above hard disk
- printer

Software Requirements

- Windows 7, Windows 8, Windows 8.1, Windows 10
- Visual studio tool.
- A good antivirus that can either be McAfee, K7 or A firewall and the most important thing is to keep your antivirus always updated same with your operating system.

Challenges in getting requirements

In the requirement getting we had many problems. Because, we were frustrated with searching the suitable client for our project in Jaffna town. After that we got the clients at other district (Batticaloa and Kegalle), but, we are permitted to searching a client from jaffna town.

We talked with many owner of pharmacy, but, they did not interest to use the computer based system to manage their pharmacy. Because, they used to familiar with their manual records and some of the pharmacies have small business and resources, they told us that is easy to maintain with manual records. Some of them thought the computer system that the hardware and software requirement cost will be very high, and also this technology were new for them. Some other owners got fear to give their medicine and pharmacy details to us.

We have got more than one month of period for a suitable client (RVG Pharmacy) for our project. We implemented our application with their requirements (RVG Pharmacy) until middle stage of our project. Unfortunately, we could not work with them. After 2 months our team were connected with “ATLANTA Pharmacy”. We showed our application until we have implemented. The Administration of “ATLANTA Pharmacy” were happy to see that. And they gave some additional requirements with few corrections.

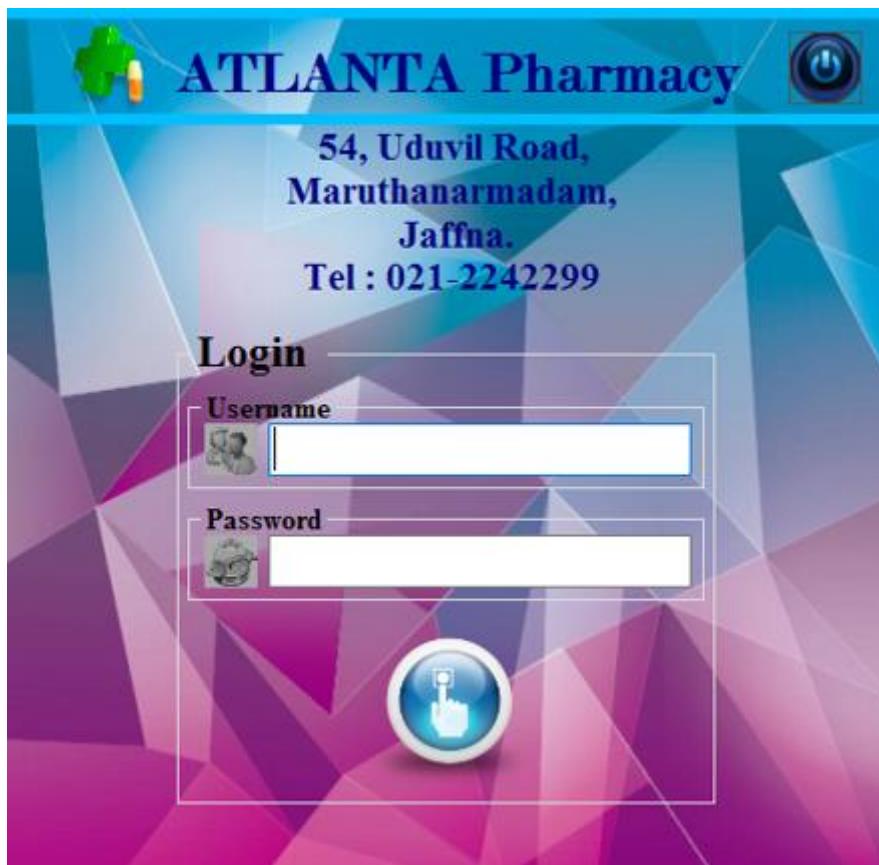
Software design

This section we expect to analyse and discuss the proposed system for the “Pharmacy Management System”. After the requirement analysis and specification, this section will give a better understanding of the system behaviour and interactions with the diagrams used. First, we want to describe the reader about the functions of the proposed system with detailed understanding.

For this purpose, we have included:

- Software Design.
- Software design approach.
- Context diagram, that defines the boundary between the system.
- Interactive diagram, that describes how a group of objects collaborate with in the use-case.
- Structural diagram, we have showed the class diagram, it is useful for showing the structure of the system.
- Behaviour diagram, here we have showed the sequence diagram to provide better clarity of the sequence of activities, and use-case diagrams to identify the main components of the proposed system.
- Data model, here we showed the ER-diagram to give an idea of database management.

Software design



Welcome Logged in as : Admin

12:30:55 AM 4/26/2017

Medicine	Staff	Supplier	Sales
Add Medicine	Add Staff	Add Supplier	Retail Sales
View Stock	View Staff	View Supplier	View Sales
Expiry Date	Modify Item	Delete Supplier	View Purchase
Delete Item	Modify Staff	Modify Supplier	

Expired Stock

mID	mName	quantity
*		

Low Stock

mID	mName	quantity
M2	Cefuroxime Ax...	1
m61	amoxilne	48
m63	azithromycin	41
*		



This screen has a light blue header with a back arrow icon on the left and a save/cancel icon on the right. The main title is "Change Your Password". Below the title is a form with four fields: "Username", "Old Password", "New Password", and "Confirm New Password", each with its own input field.

Add New Medicine

Category	<input type="text"/>	Supplier ID	<input type="text"/>
Medicine Id	<input type="text"/>	Generic Name	<input type="text"/>
Medicine Name	<input type="text"/>	Quantity	<input type="text"/>
Batch No	<input type="text"/>	Entry Date	4/26/2017
Dosage	<input type="text"/>	Manufacture Date	4/26/2017
Manufacturer	<input type="text"/>	Expiry Date	4/26/2017
Imported By	<input type="text"/>	Buying Price	<input type="text"/>
Brand Name	<input type="text"/>	Selling Price	<input type="text"/>

View Stock

Choose Type	Medicine Name	<input type="button" value="Search"/>
<input type="text"/>		

Modify Medicine Details

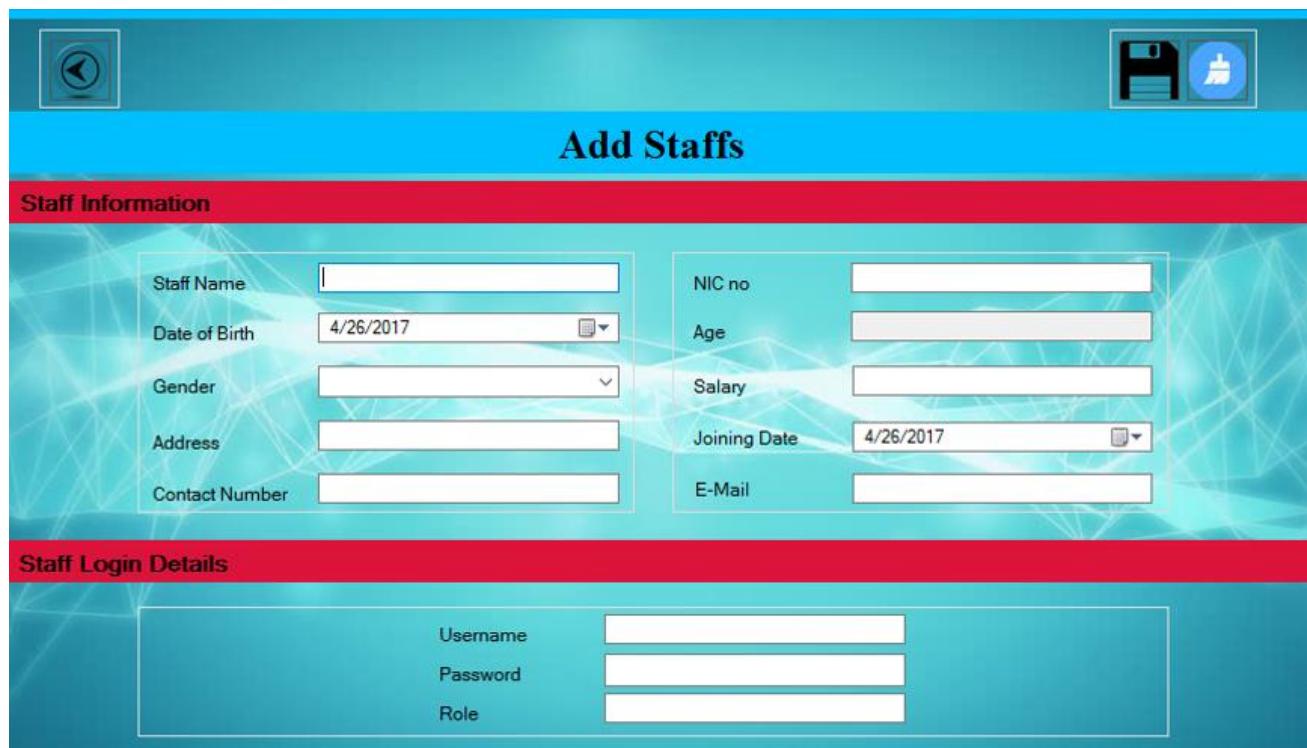
Choose Type	Medicine Name	Choose Medicine Id																																	
<input type="text"/>	<input type="text"/>	<input type="text"/>																																	
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">Category</td> <td style="width: 85%;"></td> </tr> <tr> <td>Medicine Id</td> <td><input type="text"/></td> </tr> <tr> <td>Medicine Name</td> <td><input type="text"/></td> </tr> <tr> <td>Batch No</td> <td><input type="text"/></td> </tr> <tr> <td>Dosage</td> <td><input type="text"/></td> </tr> <tr> <td>Manufacturer</td> <td><input type="text"/></td> </tr> <tr> <td>Imported/Distributed by</td> <td><input type="text"/></td> </tr> <tr> <td>Brand Name</td> <td><input type="text"/></td> </tr> <tr> <td>Generic Name</td> <td><input type="text"/></td> </tr> <tr> <td>Quantity</td> <td><input type="text"/></td> </tr> <tr> <td>Entry Date</td> <td>3/5/2017 <input type="button" value="..."/></td> </tr> <tr> <td>Manufacture Date</td> <td>4/26/2017 <input type="button" value="..."/></td> </tr> <tr> <td>Expiry Date</td> <td>4/26/2017 <input type="button" value="..."/></td> </tr> <tr> <td>Buying Price</td> <td><input type="text"/></td> </tr> <tr> <td>Selling Price</td> <td><input type="text"/></td> </tr> <tr> <td>Supplier ID</td> <td><input type="text"/></td> </tr> </table>				Category		Medicine Id	<input type="text"/>	Medicine Name	<input type="text"/>	Batch No	<input type="text"/>	Dosage	<input type="text"/>	Manufacturer	<input type="text"/>	Imported/Distributed by	<input type="text"/>	Brand Name	<input type="text"/>	Generic Name	<input type="text"/>	Quantity	<input type="text"/>	Entry Date	3/5/2017 <input type="button" value="..."/>	Manufacture Date	4/26/2017 <input type="button" value="..."/>	Expiry Date	4/26/2017 <input type="button" value="..."/>	Buying Price	<input type="text"/>	Selling Price	<input type="text"/>	Supplier ID	<input type="text"/>
Category																																			
Medicine Id	<input type="text"/>																																		
Medicine Name	<input type="text"/>																																		
Batch No	<input type="text"/>																																		
Dosage	<input type="text"/>																																		
Manufacturer	<input type="text"/>																																		
Imported/Distributed by	<input type="text"/>																																		
Brand Name	<input type="text"/>																																		
Generic Name	<input type="text"/>																																		
Quantity	<input type="text"/>																																		
Entry Date	3/5/2017 <input type="button" value="..."/>																																		
Manufacture Date	4/26/2017 <input type="button" value="..."/>																																		
Expiry Date	4/26/2017 <input type="button" value="..."/>																																		
Buying Price	<input type="text"/>																																		
Selling Price	<input type="text"/>																																		
Supplier ID	<input type="text"/>																																		

View Expiry Date

Expiry Date before the Date	<input type="text" value="4/26/2017"/> <input type="button" value="..."/>		
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;"></td> <td style="width: 85%;"></td> </tr> </table>			



This screenshot shows the 'Delete Medicine Details' page of the Pharmacy Management System. At the top, there are three dropdown menus: 'Choose Type', 'Medicine Name', and 'Select Medicine Id'. To the right of these are three icons: a magnifying glass, a trash can, and a circular arrow. Below the header, the main title 'Delete Medicine Details' is centered. A large gray rectangular area occupies the center of the page, likely a placeholder for a confirmation message or a list of deleted items.



This screenshot shows the 'Add Staffs' page. At the top, there are two icons: a left arrow and a floppy disk with a checkmark. The main title 'Add Staffs' is centered below the header. The page is divided into two main sections: 'Staff Information' and 'Staff Login Details'. The 'Staff Information' section contains fields for Staff Name, Date of Birth, Gender, Address, Contact Number, NIC no., Age, Salary, Joining Date, and E-Mail. The 'Staff Login Details' section contains fields for Username, Password, and Role.

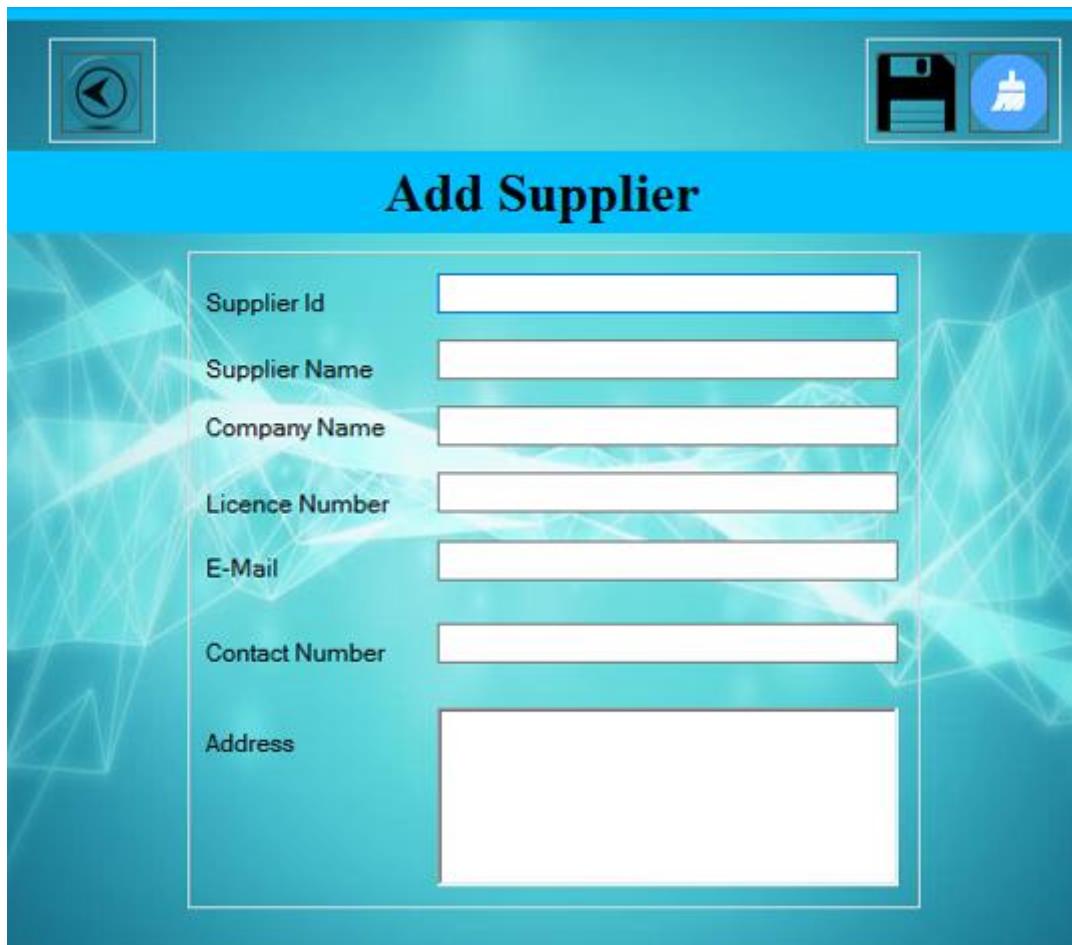
This screenshot shows the 'View Staff Details' page of the Pharmacy Management System. At the top, there are search fields for 'staffName' and 'NICno', each with a dropdown arrow. To the right of these fields are three icons: a magnifying glass, a clipboard, and a trash can. Below the search bar, the title 'View Staff Details' is centered in a blue header. A large, empty gray rectangular area occupies the main content space.

This screenshot shows the 'Delete Staff Details' page of the Pharmacy Management System. It features a search interface with 'Name' and 'NIC no' fields and standard search, edit, and delete icons. The main area is a large gray rectangle.



Modify Staff Details

Staff Name	<input type="text"/>	Staff NIC	<input type="text"/>
Date of Birth	Wednesday, April 26, 2017 <input type="button" value="▼"/>		
Gender	<input type="button" value="▼"/>		
Address	<input type="text"/>		
Contact Number	<input type="text"/>		
NIC no	<input type="text"/>		
Age	<input type="text"/>		
Salary	<input type="text"/>		
Joining Date	Wednesday, April 26, 2017 <input type="button" value="▼"/>		
E-Mail	<input type="text"/>		
Username	<input type="text"/>		
Password	<input type="text"/>		
Role	<input type="text"/>		



Add Supplier

Supplier Id	<input type="text"/>
Supplier Name	<input type="text"/>
Company Name	<input type="text"/>
Licence Number	<input type="text"/>
E-Mail	<input type="text"/>
Contact Number	<input type="text"/>
Address	<input type="text"/>

This screenshot shows the 'View Supplier Details' page of the Pharmacy Management System. At the top left is a back arrow icon. To its right are two input fields: 'Supplier Name' and 'Supplier ID', each with a dropdown arrow. On the far right is a magnifying glass search icon. Below these elements is a blue header bar with the text 'View Supplier Details' in white. The main content area is a large gray rectangular box with a thin black border.

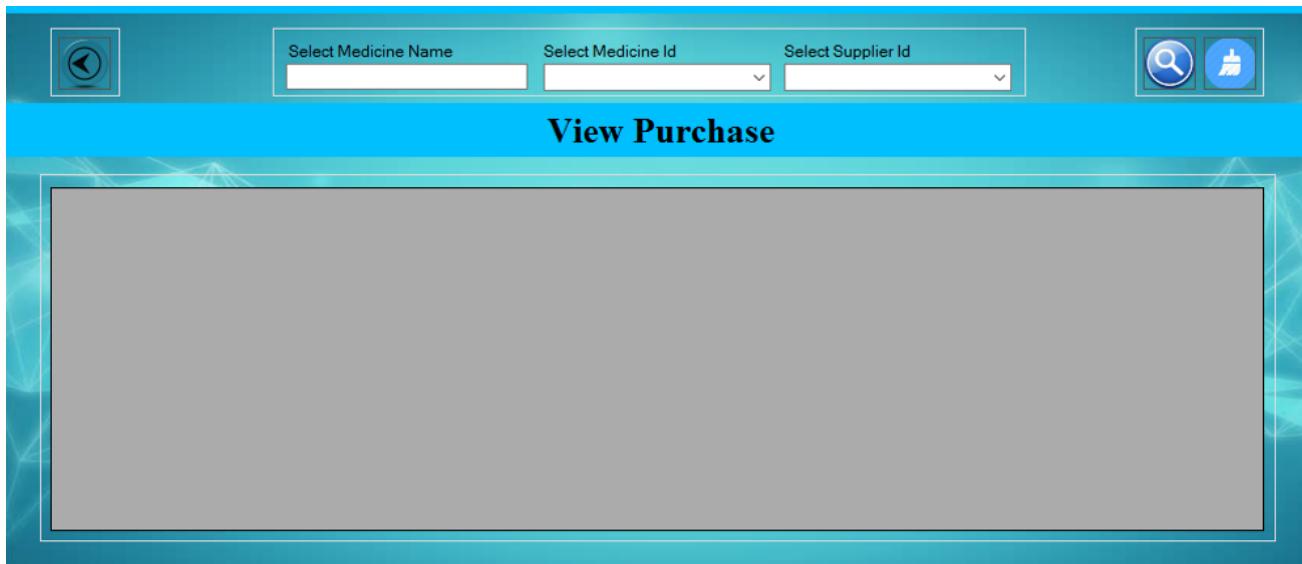
This screenshot shows the 'Delete Supplier Details' page of the Pharmacy Management System. It features a layout similar to the 'View Supplier Details' page, with a back arrow icon, 'Supplier Name' and 'Supplier ID' input fields with dropdown arrows, and a magnifying glass search icon. The main title 'Delete Supplier Details' is centered above a large gray rectangular content area. The entire interface has a teal header and a teal footer bar.

Retail Sales Detail

MedicineID	Medicine Name	Quantity	Expiry Date	Dosage	Unit Price	Amount(Retail)	
						Amount	<input type="text"/>
						Discount(%)	<input type="text"/>
						Calculate Total	
						Total Amount	<input type="text"/>
						Paid	<input type="text"/>
						Return	<input type="text"/>
						Generate Bill	
						Print Bill	

View Sales

Bill No	<input type="text"/>	search with date	<input type="text"/> 4/26/2017	
Sales of particular bill_no				
Bill Details of sales				



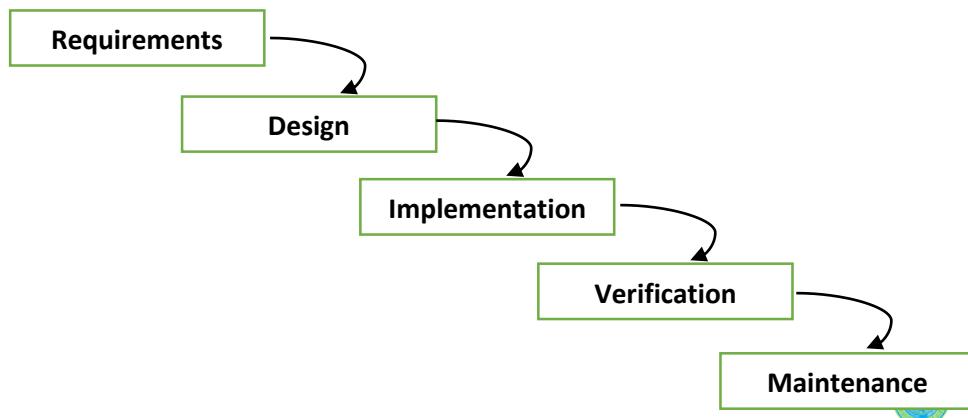
Software design approach.

In the software design, we first designed the interface with the corresponding functional and non-functional requirements. And here we are using the Top-down approach to design and implement the system.

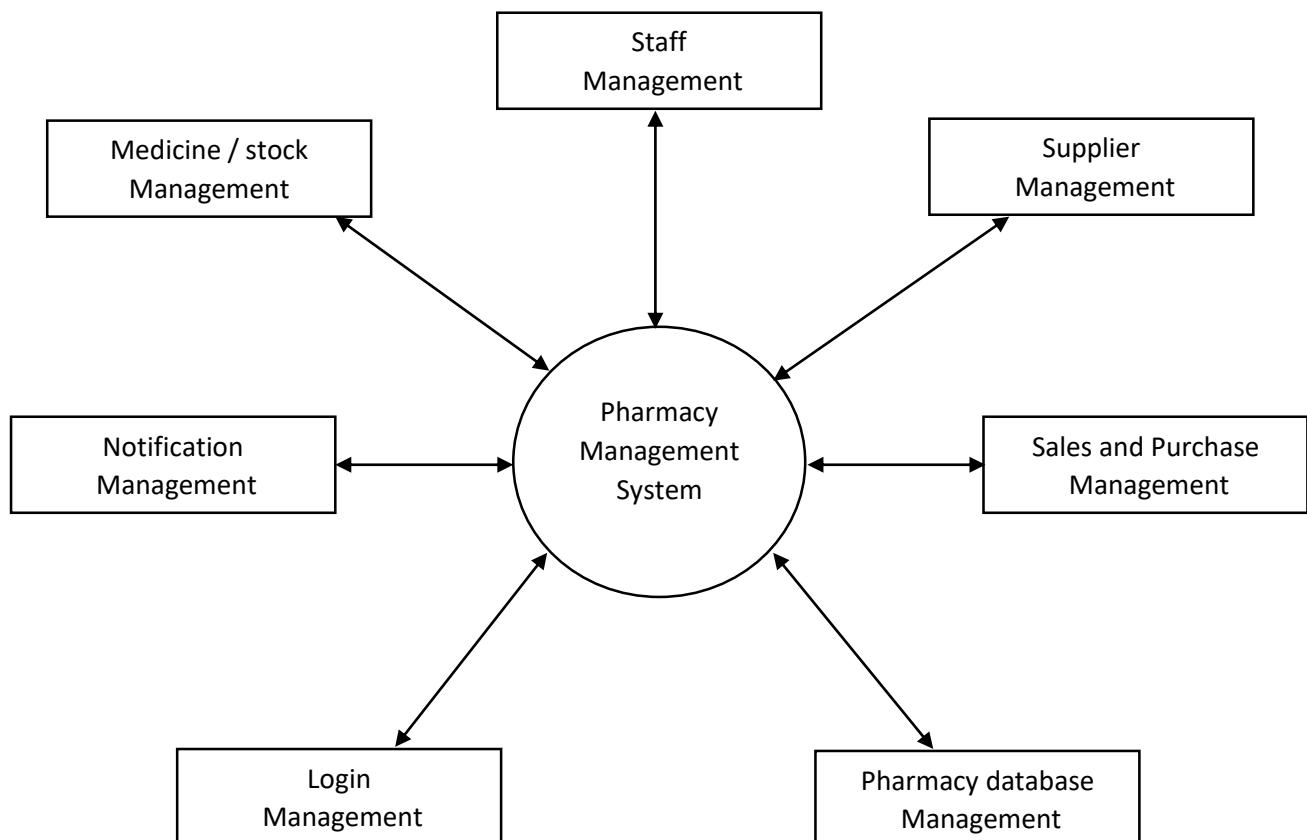
Top-down approach

Top-down design approach starts by identifying the major modules of the system. Decomposing them into their lower level modules and iterating until the desired level of detail is achieved. This is stepwise refinement, starting from an abstract design in each step the design is refined to a more concrete level, until we reached a level where no more refinement is needed and the design can be implemented directly.

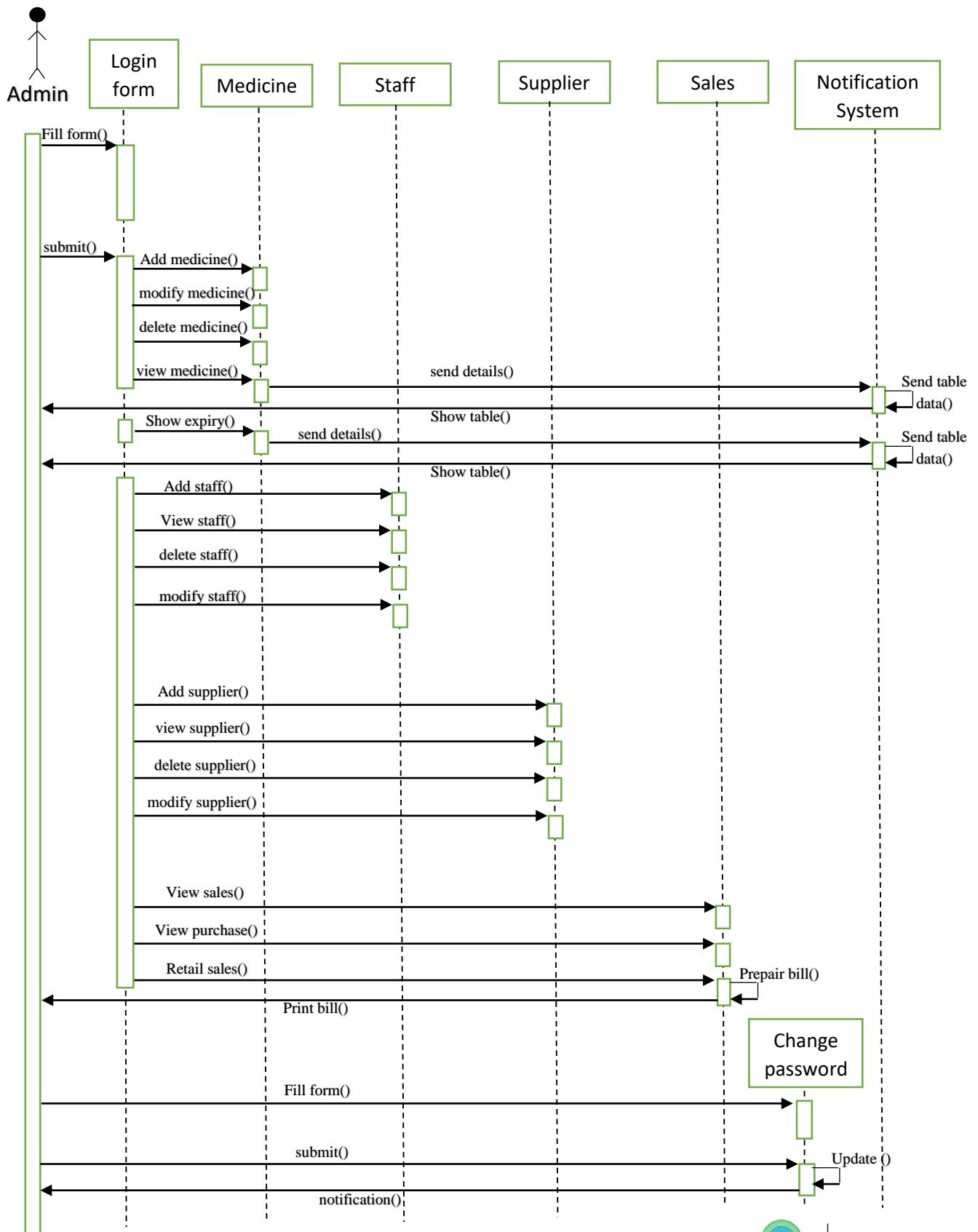
- C# library functions.
- Case studies based on our project.
- Top-down design and structure chart
- Basic concepts about functions.
- Building programs from existing information.

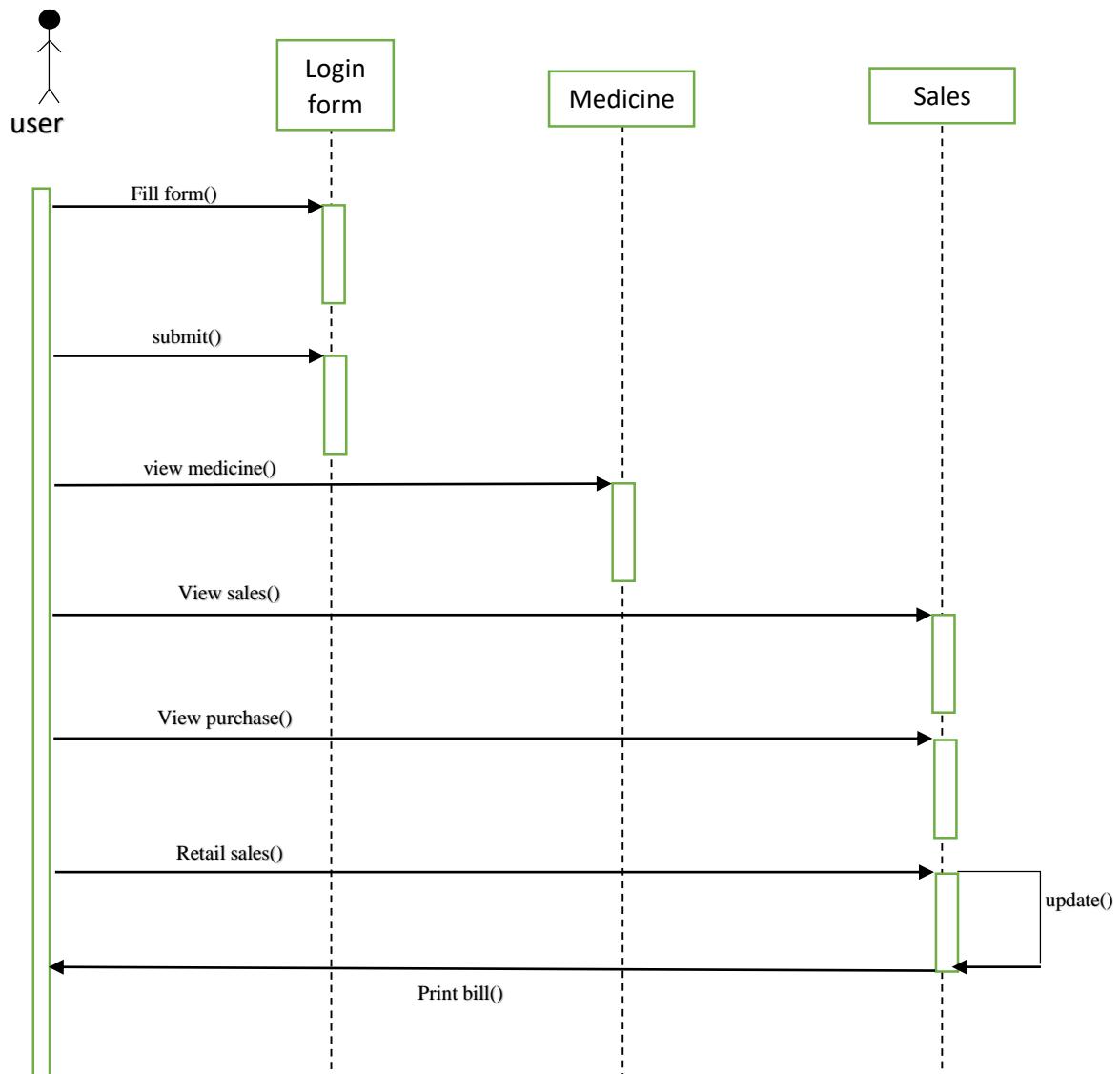


Context diagram



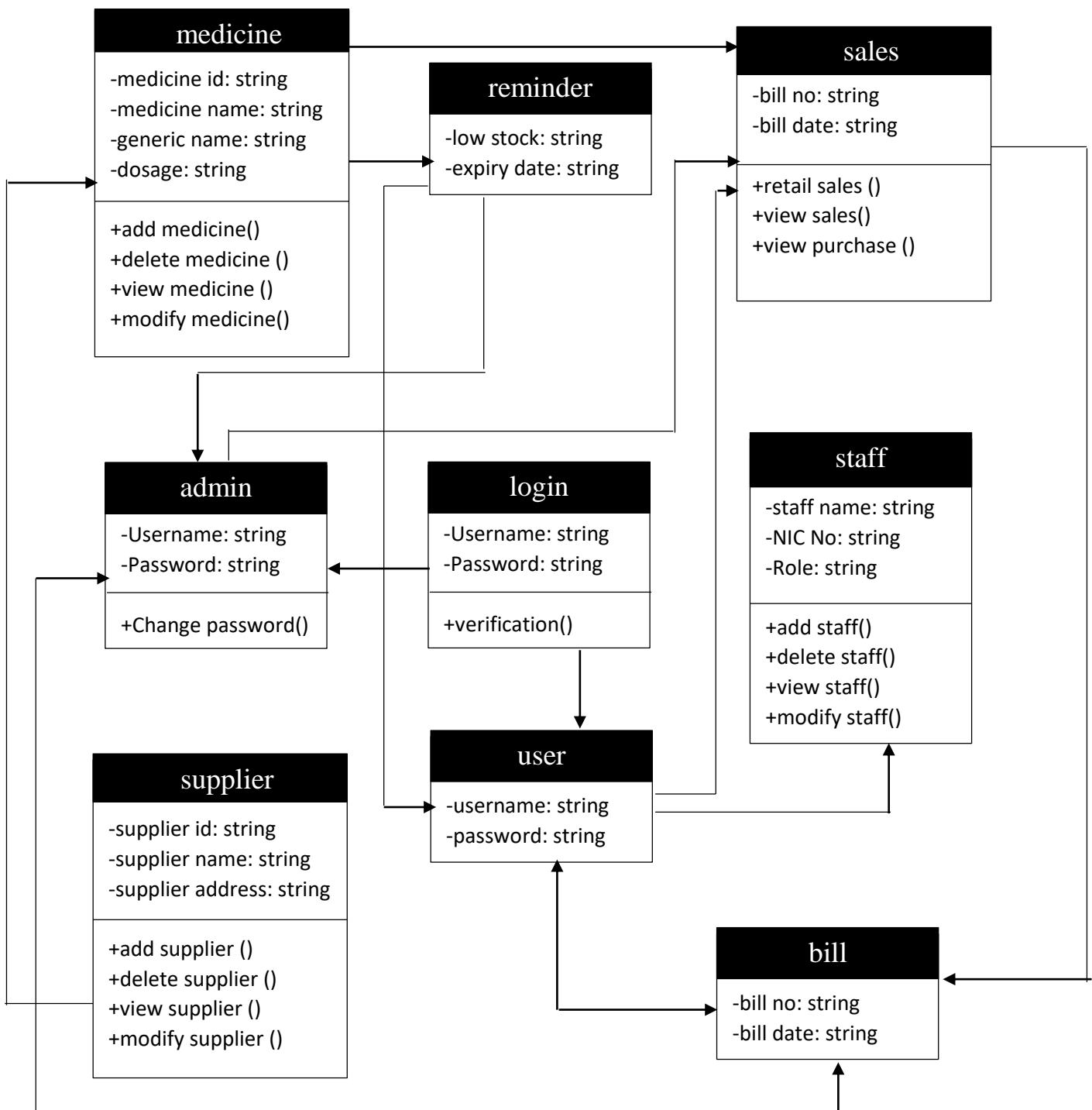
Interaction diagram





Structural Diagram

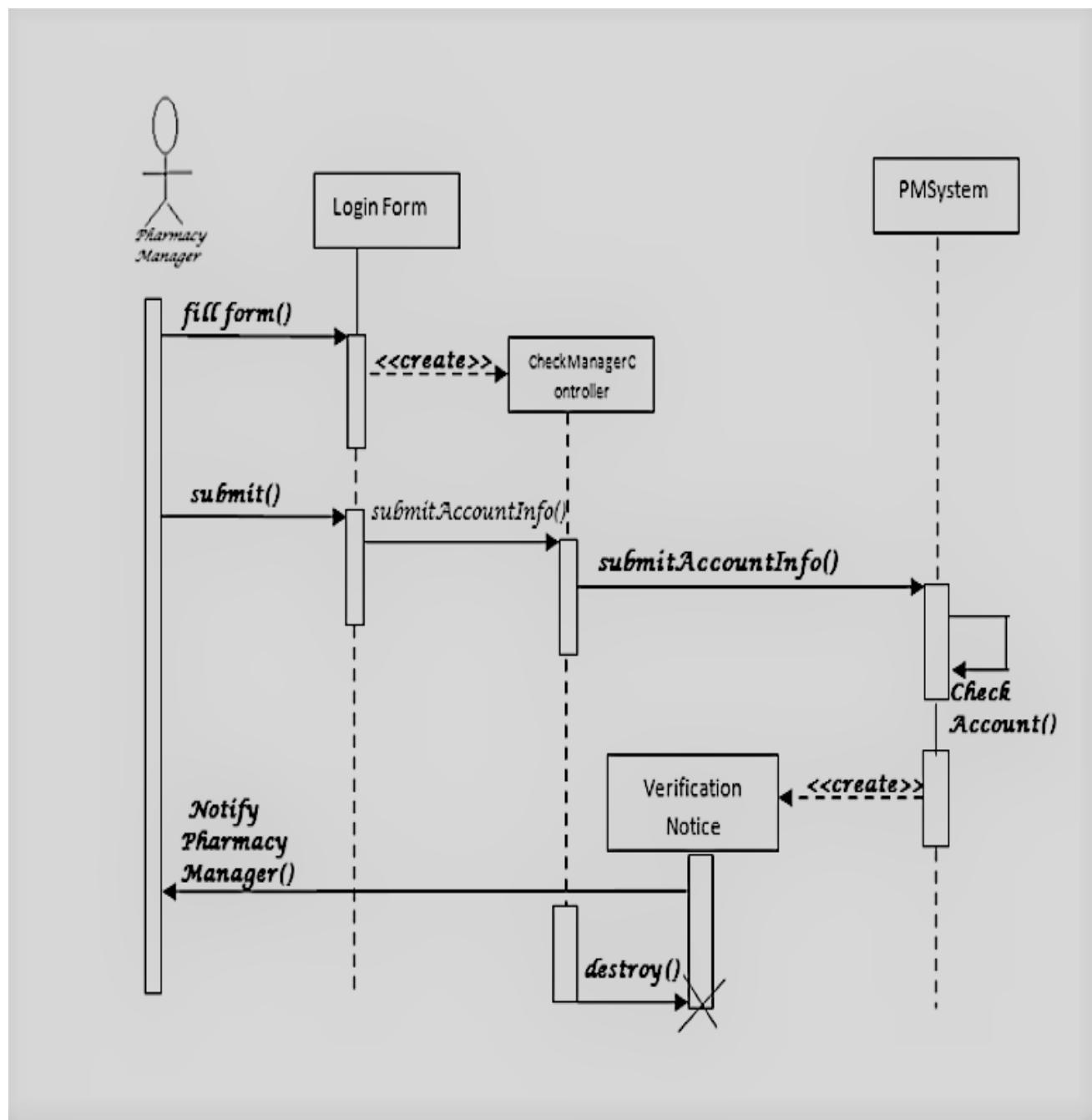
UML-Class diagram



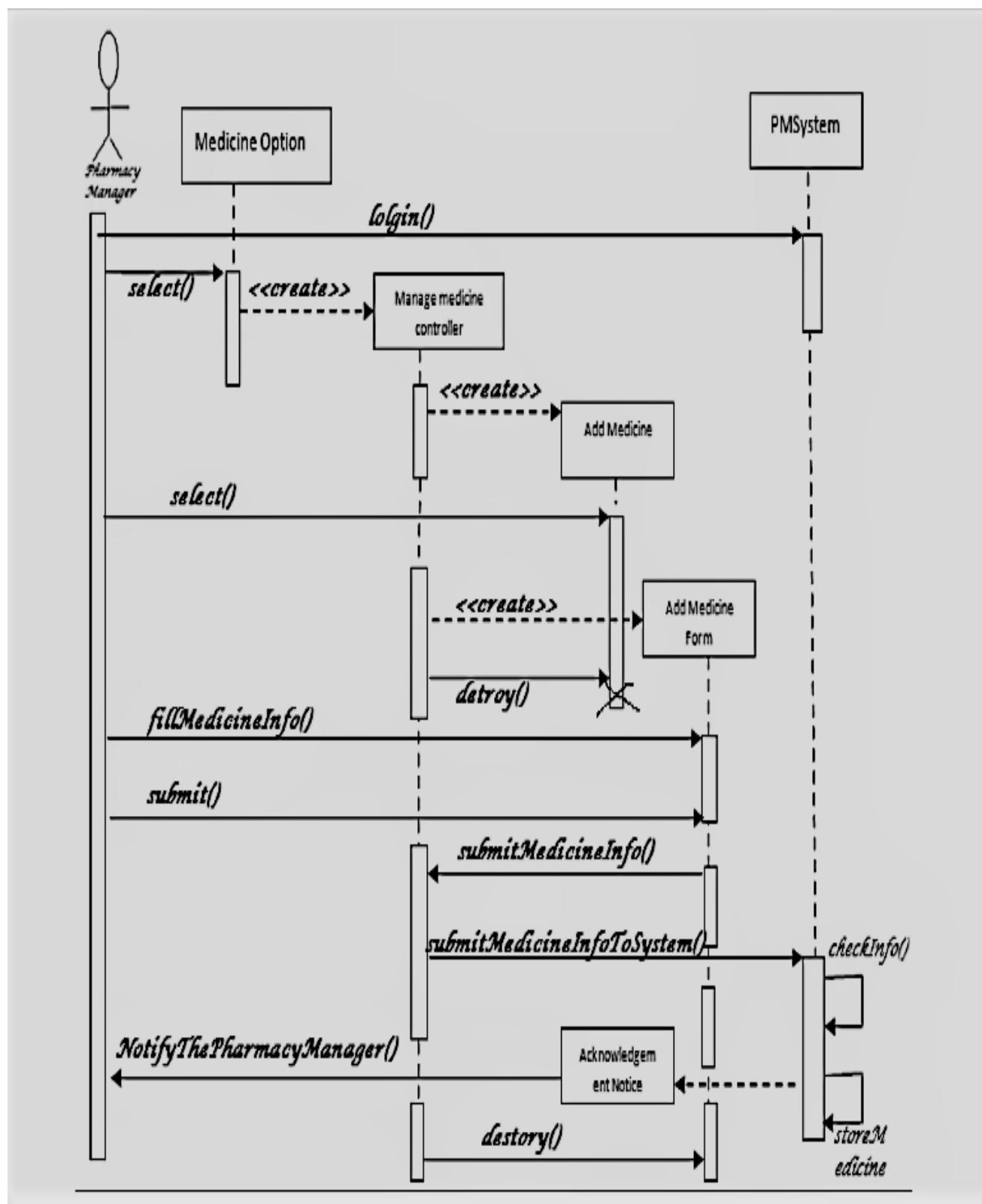
Behaviour Diagram

Sequence diagram

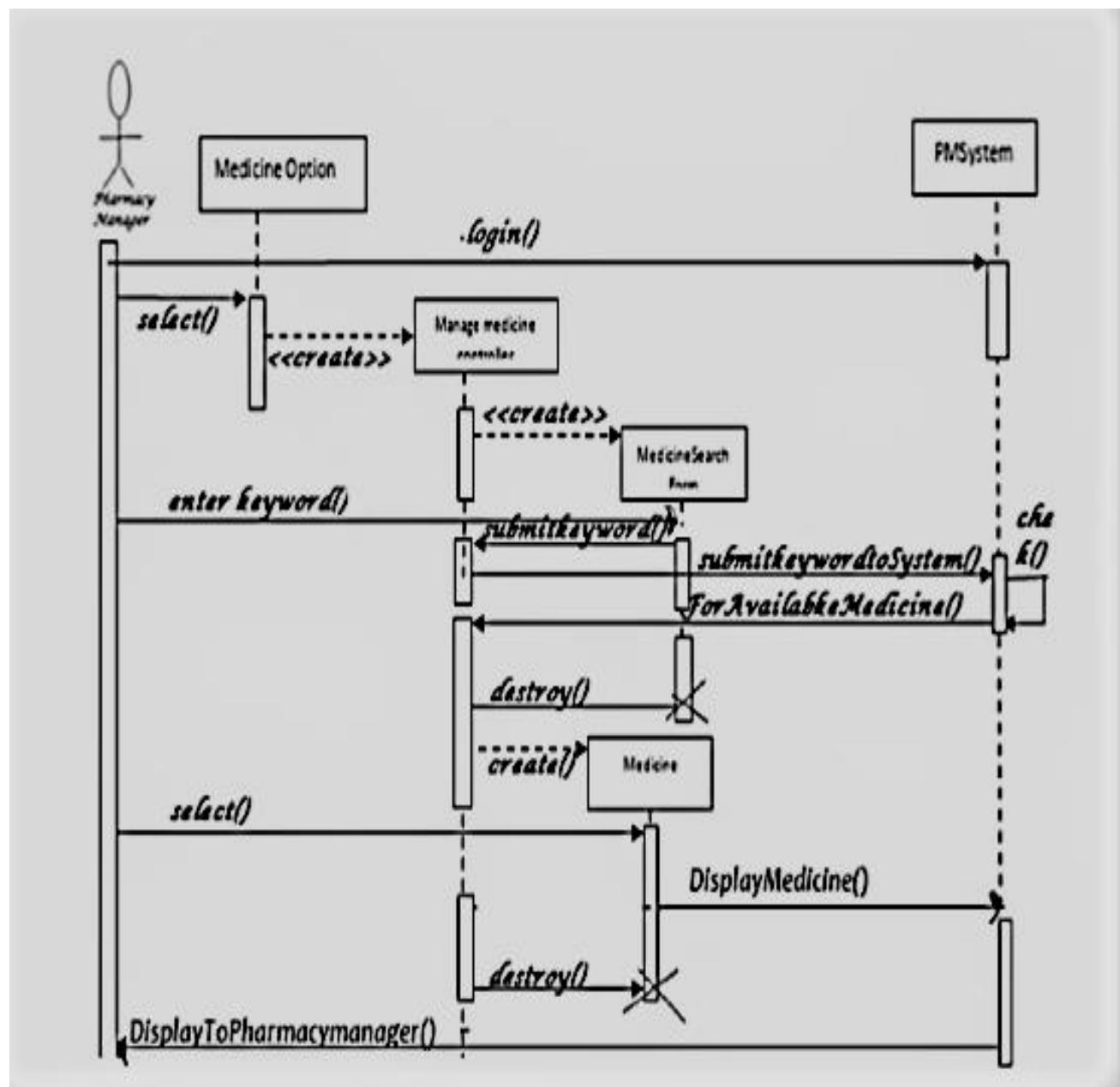
Login page



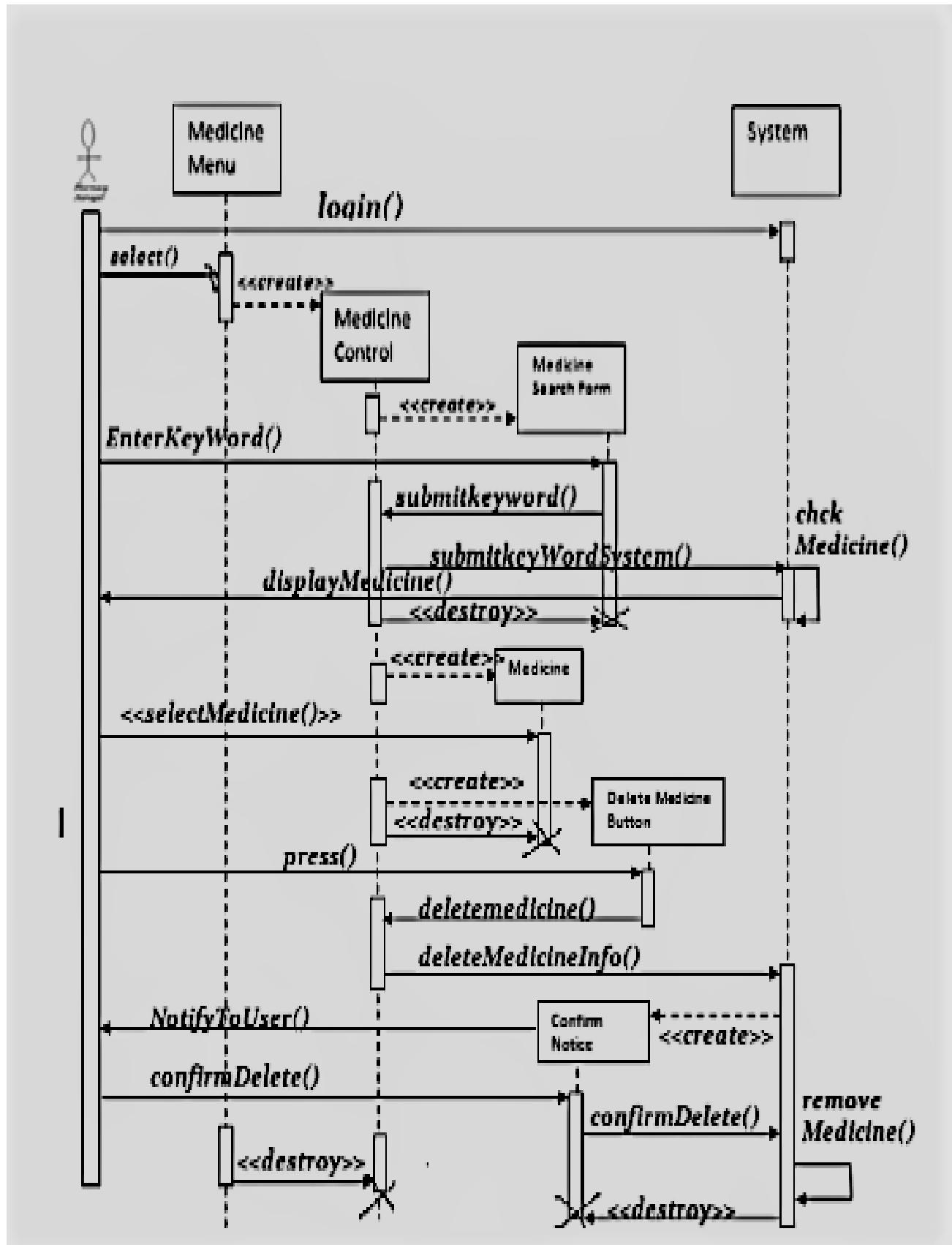
Add medicine



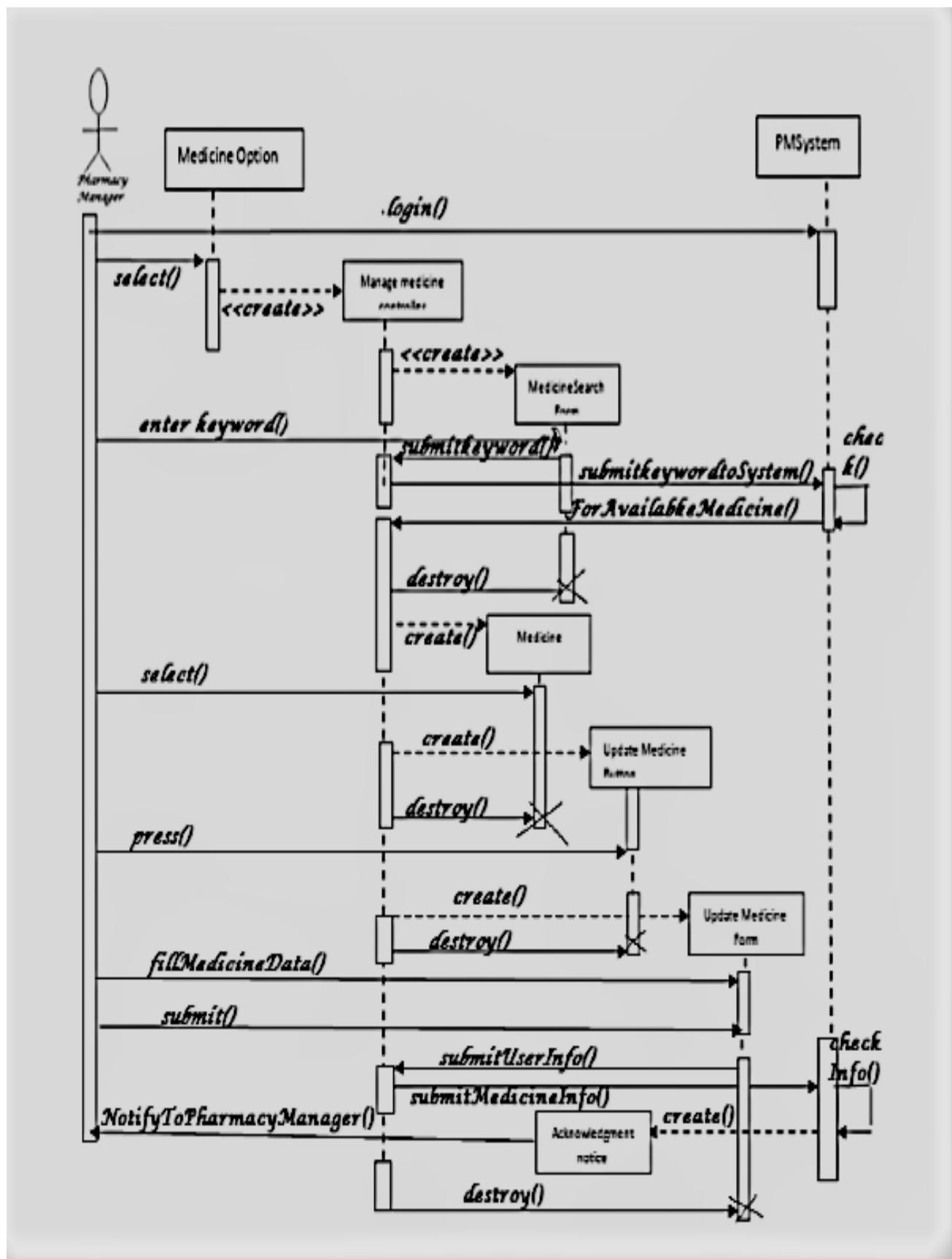
Search medicine



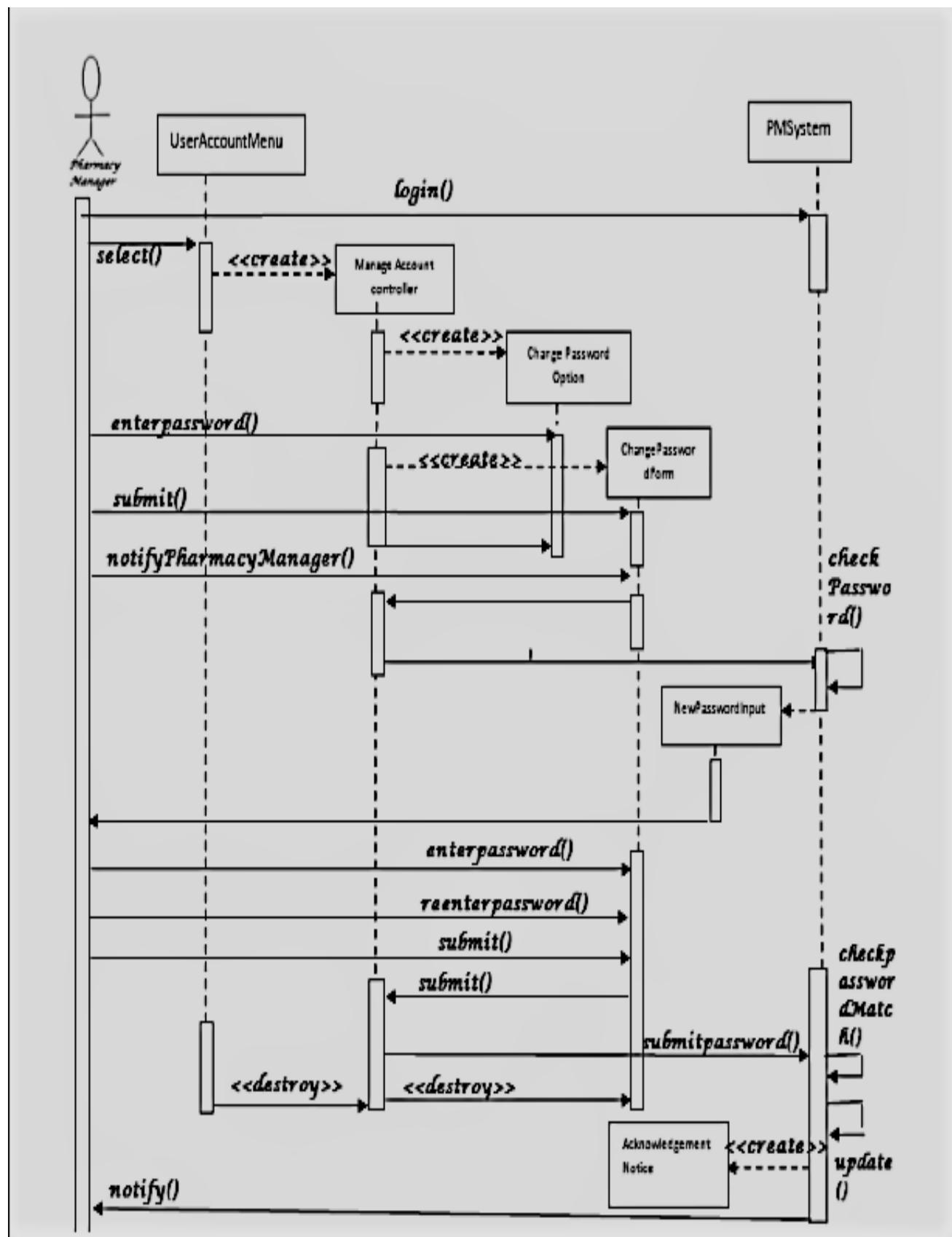
Delete medicine



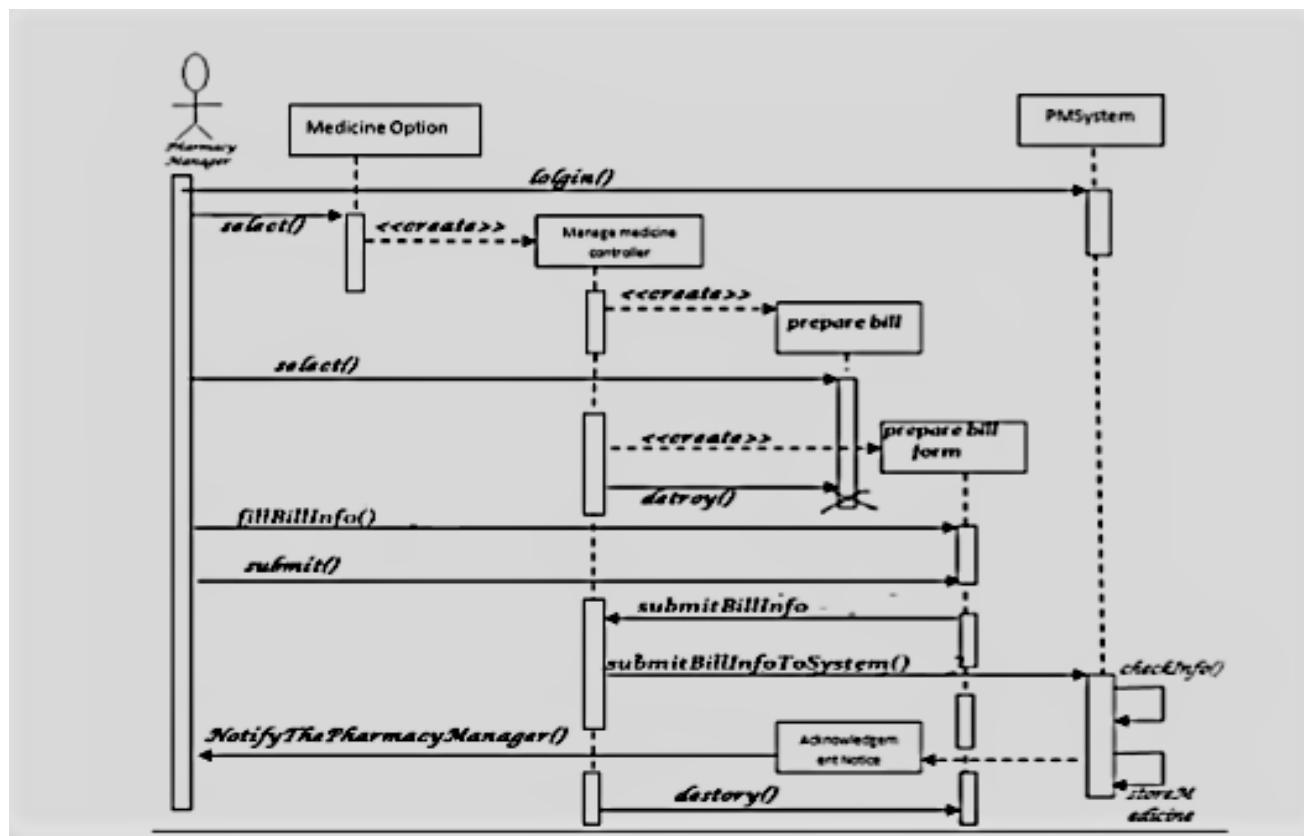
Modify medicine



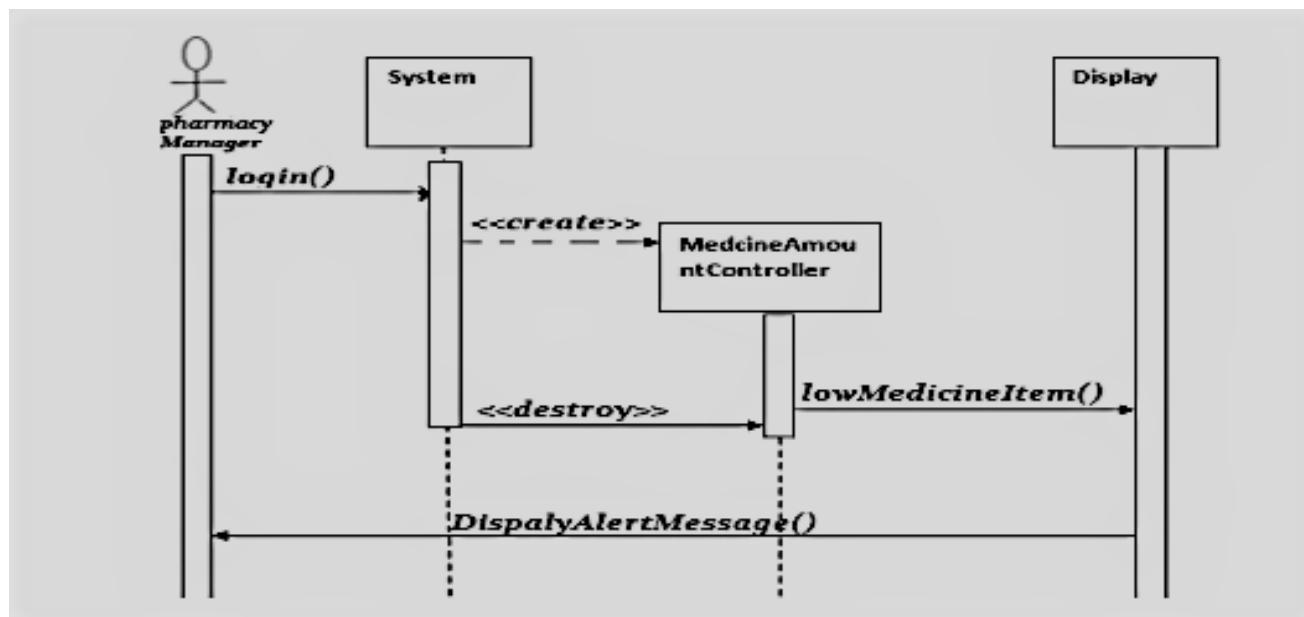
Change password



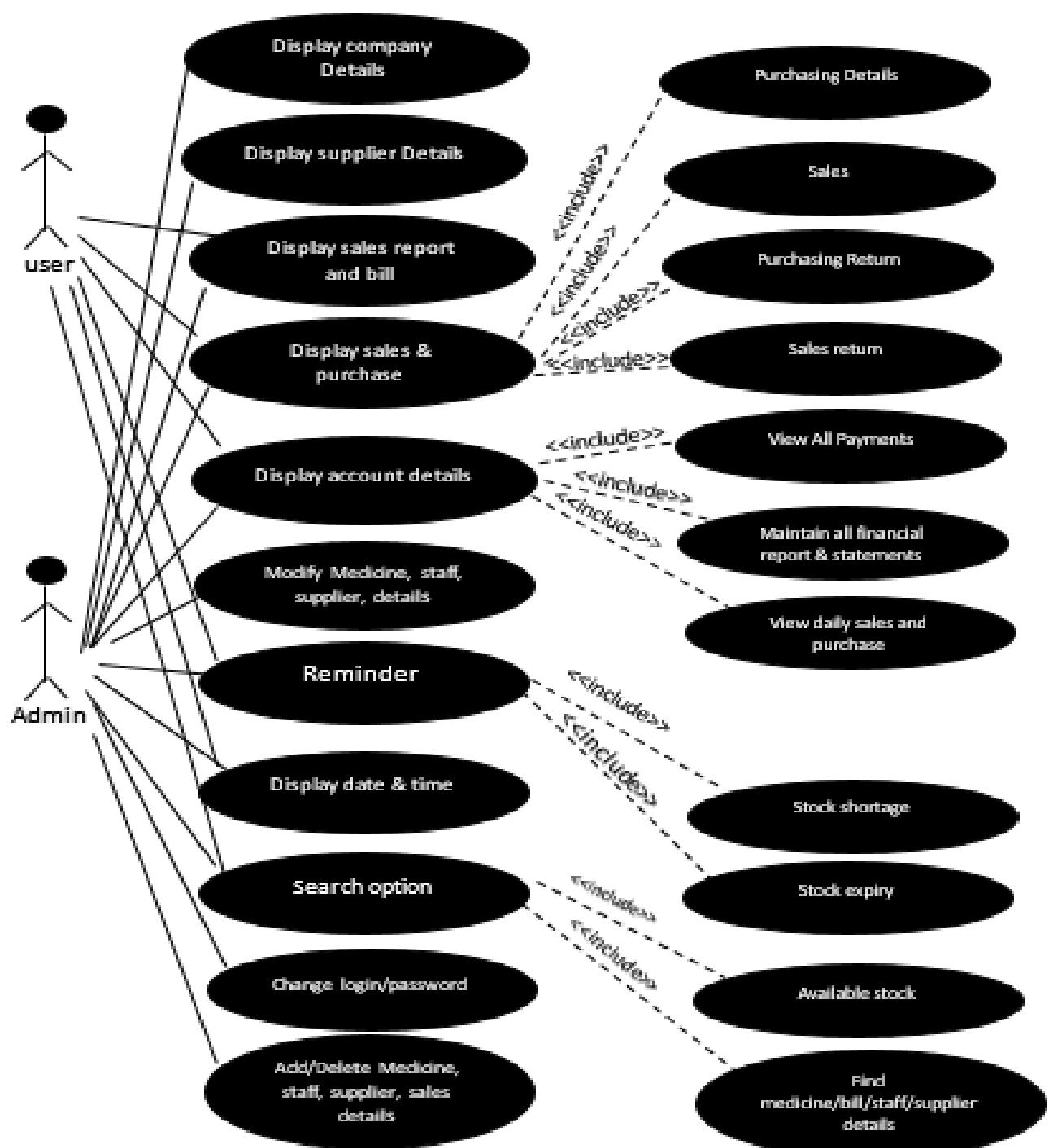
Billing



Remainder



Use-case diagram



ER diagram

Challenges we faced in designing

The course of software engineering is new to us and software implementation and system designing are new to our group, in early days we did not hear these kind of terms after that we realised in designing part of the project we have to consider requirements, software design approaches, software design.

First time, we have done a standalone system for a group project in our life time. For this project purpose we learned lot of things in short time and this is little off to us. But, we managed that, and learned that, what we need for the purpose of our group activity.

In the middle of the implementation process, our client suggested to change some functional requirements of the software design and it caused some change in the application, these changes took our valuable time to redesign our application.

And we spent lot of time to software designing part. Because, we plan to design stylish, good looking and neat software interface.

Mainly, these kind of challenges we faced during the software design. Addition to that some little issues we faced like group management.

implementation

This section describe how we implemented our system with the requirements. Here we described about what kind of techniques and tools we used, what are the special features in our application and the challenges we faced during the implementation. This section should give detailed understanding of implementation part.

- The application has two main modules that are admin and user modules. Only admin can make changes or modify the details of medicine, staff, supplier and sales modules. And user access only the sales modules.
- Some of the staffs can act as user with admin's knowledge, only admin can change or create the password in the change password module.
- The application will show the details about the low stock and expired stock in the table in admin module.
- Finally, this application will produce an invoice for each sale, and this system will record each sales and purchases according to the date.

Each and every modules of the application were implemented using above outlines that we explained.

Tools and techniques, and justification

- We have used C# programming language for our designing and implementation process and visual studio as a platform.
- Object-oriented concept is used in this application.
- SQL server is used for connecting database with our application to the process

C# programming language

C# is a multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines.

- The language is intended to be a simple, modern, general-purpose, object-oriented programming language.
- The language is intended for use in developing software components suitable for deployment in distributed environments.

- Portability is very important for source code and programmers, especially those already familiar with C and C++.
- Support for internationalization is very important.
- C# is intended to be suitable for writing applications for both hosted and embedded systems, ranging from the very large that use sophisticated operating systems, down to the very small having dedicated functions.
- Although C# applications are intended to be economical with regard to memory and processing power requirements, the language was not intended to compete directly on performance and size with C or assembly language

Microsoft SQL server

“Microsoft SQL Server” is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run either on the same computer or on another computer across a network (including the Internet).

Object oriented concept

Object-oriented programming (OOP) is a programming paradigm based on the concept of "objects", which may contain data, in the form of fields, often known as attributes; and code, in the form of procedures, often known as methods.

A feature of objects is that an object's procedures can access and often modify the data fields of the object with which they are associated (objects have a notion of "this" or "self"). In OOP, computer programs are designed by making them out of objects that interact with one another. There is significant diversity of OOP languages, but the most popular ones are class-based, meaning that objects are instances of classes, which typically also determine their type.

In our application medicine and sales modules are containing the main functions. In medicine module, add, delete and show expiry are main functionalities and sales module, retail sale is important functionality. And here, following are the coding for these functionality.

Coding for add medicine

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace Pharmacy_Management_System
{
    public partial class Add_Medicine : Form
    {
        SqlConnection con;
        SqlCommand com;
        string sql;
        public Add_Medicine()
        {
            InitializeComponent();
            con = new SqlConnection(@"Data
Source=.\SQLEXPRESS;AttachDbFilename=C:\Pharmacy_Management_System\Pharmacy_Management_System
\Medicine.mdf;Integrated Security=True;User Instance=True");
        }

        private void Add_Medicine_Load(object sender, EventArgs e)
        {
            textBox12.Text = dateTimePicker1.Value.ToShortDateString();

            con.Open();
            SqlCommand cmd0 = new SqlCommand("SELECT mID FROM details_M", con);

            SqlDataReader reader0 = cmd0.ExecuteReader();
            AutoCompleteStringCollection MyCollection0 = new AutoCompleteStringCollection();
            while (reader0.Read())
            {
                MyCollection0.Add(reader0.GetString(0));
            }
            textBox1.AutoCompleteCustomSource = MyCollection0;
            con.Close();

            con.Open();
            SqlCommand cmd = new SqlCommand("SELECT mName FROM details_M", con);

            SqlDataReader reader = cmd.ExecuteReader();
            AutoCompleteStringCollection MyCollection = new AutoCompleteStringCollection();
            while (reader.Read())
            {
                MyCollection.Add(reader.GetString(0));
            }
            textBox2.AutoCompleteCustomSource = MyCollection;
            con.Close();

            con.Open();
            SqlCommand cmd1 = new SqlCommand("SELECT batchNo FROM details_M", con);

```

```

SqlDataReader reader1 = cmd1.ExecuteReader();
AutoCompleteStringCollection MyCollection1 = new AutoCompleteStringCollection();
while (reader1.Read())
{
    MyCollection1.Add(reader1.GetString(0));
}
textBox3.AutoCompleteCustomSource = MyCollection1;
con.Close();

con.Open();
SqlCommand cmd2 = new SqlCommand("SELECT dosage FROM details_M", con);

SqlDataReader reader2 = cmd2.ExecuteReader();
AutoCompleteStringCollection MyCollection2 = new AutoCompleteStringCollection();
while (reader2.Read())
{
    MyCollection2.Add(reader2.GetString(0));
}
textBox4.AutoCompleteCustomSource = MyCollection2;
con.Close();

con.Open();
SqlCommand cmd3 = new SqlCommand("SELECT manufacturer FROM details_M", con);

SqlDataReader reader3 = cmd3.ExecuteReader();
AutoCompleteStringCollection MyCollection3 = new AutoCompleteStringCollection();
while (reader3.Read())
{
    MyCollection3.Add(reader3.GetString(0));
}
textBox5.AutoCompleteCustomSource = MyCollection3;
con.Close();

con.Open();
SqlCommand cmd4 = new SqlCommand("SELECT importBy FROM details_M", con);

SqlDataReader reader4 = cmd4.ExecuteReader();
AutoCompleteStringCollection MyCollection4 = new AutoCompleteStringCollection();
while (reader4.Read())
{
    MyCollection4.Add(reader4.GetString(0));
}
textBox9.AutoCompleteCustomSource = MyCollection4;
con.Close();

con.Open();
SqlCommand cmd5 = new SqlCommand("SELECT brandName FROM details_M", con);

SqlDataReader reader5 = cmd5.ExecuteReader();
AutoCompleteStringCollection MyCollection5 = new AutoCompleteStringCollection();
while (reader5.Read())
{
    MyCollection5.Add(reader5.GetString(0));
}
textBox10.AutoCompleteCustomSource = MyCollection5;
con.Close();

con.Open();
SqlCommand cmd6 = new SqlCommand("SELECT genericName FROM details_M", con);

```

```

SqlDataReader reader6 = cmd6.ExecuteReader();
AutoCompleteStringCollection MyCollection6 = new AutoCompleteStringCollection();
while (reader6.Read())
{
    MyCollection6.Add(reader6.GetString(0));
}
textBox11.AutoCompleteCustomSource = MyCollection6;
con.Close();
}

private void textBox4_KeyPress(object sender, KeyPressEventArgs e)
{
    e.Handled = char.IsNumber(e.KeyChar) || e.KeyChar == 8 || e.KeyChar == 109 ||
e.KeyChar == 108 || e.KeyChar == 103 || e.KeyChar == 107 ? false : true;
}

private void textBox6_KeyPress(object sender, KeyPressEventArgs e)
{
    e.Handled = char.IsNumber(e.KeyChar) || e.KeyChar == 8 ? false : true;
}

private void textBox7_KeyPress(object sender, KeyPressEventArgs e)
{
    e.Handled = char.IsNumber(e.KeyChar) || e.KeyChar == 8 || e.KeyChar == 46 ? false
: true;
}

private void textBox8_KeyPress(object sender, KeyPressEventArgs e)
{
    e.Handled = char.IsNumber(e.KeyChar) || e.KeyChar == 8 || e.KeyChar == 46 ? false
: true;
}

private void button3_Click(object sender, EventArgs e)
{
    Login_Admin la = new Login_Admin();
    la.Show();
    this.Hide();
}

private void button1_Click(object sender, EventArgs e)
{
    try
    {
        string x, ab, cd, ef, gh, ij, kl, mn, op, qr, st, uv, xy, mk, im, br, gn, cv;

        x = comboBox1.Text;
        ab = textBox1.Text;
        cd = textBox2.Text;
        ef = textBox3.Text;
        gh = textBox4.Text;
        ij = textBox5.Text;
        im = textBox9.Text;
        br = textBox10.Text;
        gn = textBox11.Text;
        kl = textBox6.Text;
        mn = textBox12.Text;
        op = dateTimePicker2.Text;
        qr = dateTimePicker3.Text;
        st = textBox7.Text;
        uv = textBox8.Text;
    }
}

```

```

cv = textBox13.Text;

if ((x == "") || (ab == "") || (cd == "") || (ef == "") || (gh == "") || (ij ==
== "") || (kl == "") || (mn == "") || (op == "") || (qr == "") || (st == "") || (uv == "") ||
(im == "") || (br == "") || (gn == "") || (cv == ""))
{
    MessageBox.Show("please add the full details", "Warning");
}
else
{

    con.Open();
    sql = "insert into details_M values('"+ x + "','" + ab + "','" + cd +
"','" + ef + "','" + gh + "','" + im + "','" + br + "','" + gn + "','" + ij + "','" + kl +
"','" + mn + "','" + op + "','" + qr + "','" + st + "','" + uv + "','" + cv + "')";
    com = new SqlCommand(sql, con);
    com.ExecuteNonQuery();

    con.Close();
    MessageBox.Show("successfully added", "Add", MessageBoxButtons.OK,
MessageBoxIcon.Information);
}

textBox4.Text = "";
textBox5.Text = "";
textBox2.Text = "";
textBox1.Text = "";
textBox3.Text = "";
textBox6.Text = "";
textBox7.Text = "";
textBox8.Text = "";
textBox9.Text = "";
textBox10.Text = "";
textBox11.Text = "";
dateTimePicker1.Text = "";
dateTimePicker2.Text = "";
dateTimePicker3.Text = "";
textBox13.Text = "";

}

catch (Exception ex)
{
    MessageBox.Show("something went wrong, try again..!!!");
}
}

private void button2_Click(object sender, EventArgs e)
{
    textBox4.Text = "";
    textBox5.Text = "";
    textBox2.Text = "";
    textBox1.Text = "";
    textBox3.Text = "";
    textBox6.Text = "";
    textBox7.Text = "";
    textBox8.Text = "";
    textBox9.Text = "";
    textBox10.Text = "";
    textBox11.Text = "";
    dateTimePicker1.Text = "";
}

```

```

        dateTimePicker2.Text = "";
        dateTimePicker3.Text = "";
        textBox13.Text = "";
    }

    private void textBox1_TextChanged(object sender, EventArgs e)
    {

    }

    private void textBox1_KeyDown(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Enter)
        {
            textBox2.Focus();
        }
    }

    private void textBox2_KeyDown(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Enter)
        {
            textBox3.Focus();
        }
    }

    private void textBox3_KeyDown(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Enter)
        {
            textBox4.Focus();
        }
    }

    private void textBox4_KeyDown(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Enter)
        {
            textBox5.Focus();
        }
    }

    private void textBox5_KeyDown(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Enter)
        {
            textBox9.Focus();
        }
    }

    private void textBox9_KeyDown(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Enter)
        {
            textBox10.Focus();
        }
    }

    private void textBox10_KeyDown(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.Enter)
        {
    }

```

```

        textBox13.Focus();
    }

}

private void textBox11_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        textBox6.Focus();
    }
}

private void textBox6_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        dateTimePicker2.Focus();
    }
}

private void dateTimePicker1_KeyDown(object sender, KeyEventArgs e)
{
}

private void dateTimePicker2_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        dateTimePicker3.Focus();
    }
}

private void dateTimePicker3_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        textBox7.Focus();
    }
}

private void textBox7_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        textBox8.Focus();
    }
}

private void textBox8_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        button1.PerformClick();
    }
}

private void dateTimePicker2_ValueChanged(object sender, EventArgs e)
{
    if (dateTimePicker2.Value > DateTime.Now)
    {
        dateTimePicker2.Value = DateTime.Now;
    }
}

```

```
        MessageBox.Show("select the correct date ..!!", "warning");

    }

}

private void dateTimePicker3_ValueChanged(object sender, EventArgs e)
{
    if (dateTimePicker3.Value < DateTime.Now)
    {
        dateTimePicker3.Value = DateTime.Now;
        MessageBox.Show("select the correct date ..!!", "warning");
    }
}

private void textBox13_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        textBox11.Focus();
    }
}

}
```

Coding for delete medicine

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace Pharmacy_Management_System
{
    public partial class Delete_Medicine_Details : Form
    {
        SqlConnection con;
        SqlCommand com;
        string sql;
        public Delete_Medicine_Details()
        {
            InitializeComponent();
            con = new SqlConnection(@"Data
Source=.\SQLEXPRESS;AttachDbFilename=F:\Pharmacy_Management_System\Pharmacy_Management_System
\Medicine.mdf;Integrated Security=True;User Instance=True");
        }

        private void label1_Click(object sender, EventArgs e)
        {
```

```

        }

e)    private void dataGridView1_CellContentClick(object sender, DataGridViewCellEventArgs
{
}

private void Delete_Medicine_Details_Load(object sender, EventArgs e)
{
    con.Open();
    SqlCommand cmd = new SqlCommand("SELECT mName FROM details_M", con);

    SqlDataReader reader = cmd.ExecuteReader();
    AutoCompleteStringCollection MyCollection = new AutoCompleteStringCollection();
    while (reader.Read())
    {
        MyCollection.Add(reader.GetString(0));
    }
    textBox1.AutoCompleteCustomSource = MyCollection;
    con.Close();
}

private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    con.Open();
    SqlCommand cmd = new SqlCommand("Select mName from details_M where category = '" +
+ comboBox1.Text + "'", con);

    SqlDataReader reader = cmd.ExecuteReader();
    AutoCompleteStringCollection MyCollection = new AutoCompleteStringCollection();
    while (reader.Read())
    {
        MyCollection.Add(reader.GetString(0));
    }
    textBox1.AutoCompleteCustomSource = MyCollection;
    con.Close();
}

private void button1_Click(object sender, EventArgs e)
{
    try
    {
        if ((textBox1.Text == "") || (comboBox1.Text == ""))
        {
            MessageBox.Show("sorry,please add the full details", "Warning");
        }
        con.Open();

        //com = new SqlCommand("select * from details where CONVERT
        (varchar,expiryDate) ='" + dateTimePicker1.Text + "'", con);
        sql = "select * from details_M where CONVERT(varchar,mID) = '" +
        comboBox2.Text + "'";

        SqlDataAdapter da = new SqlDataAdapter(sql, con);

        DataTable di = new DataTable();
        da.Fill(di);
        dataGridView1.DataSource = di;
        con.Close();
    }
}

```

```

        catch (Exception ex1)
        {
            MessageBox.Show("sorry there are something went wrong..!!!");
        }

    }

    private void button2_Click(object sender, EventArgs e)
    {
        textBox1.Text = "";
        comboBox1.Text = "";
    }

    private void button3_Click(object sender, EventArgs e)
    {
        try
        {
            // con.Open();
            if ((textBox1.Text == "") || (comboBox1.Text == ""))
            {
                MessageBox.Show("sorry,please add the full details", "Warning");
            }
            else
            {

                con.Open();

                DialogResult result = MessageBox.Show("Do you want to Remove the Member ?",
                "Warning", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

                if (result == DialogResult.Yes)
                {
                    sql = ("delete from details_M where CONVERT(VARCHAR,mID) ='" +
                comboBox2.Text + "'");

                    com = new SqlCommand(sql, con);

                    com.ExecuteNonQuery();

                    MessageBox.Show("successfully deleted", "Delete", MessageBoxButtons.OK,
                    MessageBoxIcon.Information);

                    textBox1.Text = "";
                    comboBox2.Text = "";
                }
                else
                {
                    MessageBox.Show("not deleted", "Warning", MessageBoxButtons.OK,
                    MessageBoxIcon.Information);
                    textBox1.Text = "";
                    comboBox2.Text = "";

                }
                con.Close();
            }
            //con.Close();
        }

        catch (Exception ex1)
        {
            MessageBox.Show("sorry there are something went wrong..!!!");
        }

    }
}

```

```
private void button4_Click(object sender, EventArgs e)
{
    Login_Admin la = new Login_Admin();
    la.Show();
    this.Hide();
}

private void textBox1_TextChanged(object sender, EventArgs e)
{
    try
    {
        con.Open();
        com = new SqlCommand("select * from details_M where mName = '" +
textBox1.Text + "'", con);
        SqlDataReader dr;
        dr = com.ExecuteReader();
        while (dr.Read())
        {
            comboBox2.Items.Add(dr["mID"].ToString());
        }
        dr.Close();
        con.Close();
    }
    catch (Exception ex1)
    {
        MessageBox.Show("sorry there are something went wrong, please try
again...!!!");
    }
}
```

Coding for view expiry

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace Pharmacy_Management_System
{
    public partial class View_Expiry_Date : Form
    {
        SqlConnection con;
        SqlCommand com;
        string sql;
        public View_Expiry_Date()
        {
            InitializeComponent();
            con = new SqlConnection(@"Data
Source=.\SQLEXPRESS;AttachDbFilename=F:\Pharmacy_Management_System\Pharmacy_Management_System
\Medicine.mdf;Integrated Security=True;User Instance=True");
        }
    }
}
```

```

private void dataGridView1_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
}

private void button4_Click(object sender, EventArgs e)
{
    Login_Admin la = new Login_Admin();
    la.Show();
    this.Hide();
}

private void button6_Click(object sender, EventArgs e)
{
    try
    {
        if ((dateTimePicker1.Text == ""))
        {
            MessageBox.Show("sorry,please add the full details", "Warning");
        }
        con.Open();

        //com = new SqlCommand("select * from details where CONVERT
        (varchar,expiryDate) ='" + dateTimePicker1.Text + "'", con);
        sql = "select * from details_M where expiryDate ='" + dateTimePicker1.Text +
        "'";
        SqlDataAdapter da = new SqlDataAdapter(sql, con);

        DataTable di = new DataTable();
        da.Fill(di);
        dataGridView1.DataSource = di;
        con.Close();
    }
    catch (Exception ex1)
    {
        MessageBox.Show("sorry there are something went wrong, please try
again..!!!");
    }
}

private void button5_Click(object sender, EventArgs e)
{
    try
    {
        if ((dateTimePicker1.Text == ""))
        {
            MessageBox.Show("sorry,please add the full details", "Warning");
        }
        else
        {

            con.Open();

            DialogResult result = MessageBox.Show("Do you want to Remove the Member
?", "Warning", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

            if (result == DialogResult.Yes)
            {
                SqlCommand cmd = new SqlCommand("UPDATE details_M SET quantity = 0
where expiryDate ='" + dateTimePicker1.Text + "'", con);
            }
        }
    }
}

```

```
cmd.ExecuteNonQuery();  
  
SqlDataAdapter da = new SqlDataAdapter(sql, con);  
  
DataTable di = new DataTable();  
da.Fill(di);  
dataGridView1.DataSource = di;  
  
MessageBox.Show("successfully deleted", "Delete",  
MessageBoxButtons.OK, MessageBoxIcon.Information);  
  
dateTimePicker1.Text = "";  
}  
else  
{  
    MessageBox.Show("not deleted", "Warning", MessageBoxButtons.OK,  
MessageBoxIcon.Information);  
    dateTimePicker1.Text = "";  
  
}  
con.Close();  
}  
}  
catch (Exception ex1)  
{  
    MessageBox.Show("sorry there are something went wrong, please try  
again..!!");  
}  
}  
}  
}
```

Coding for retail sales and billing

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;
using Pharmacy_Management_System.Properties;

namespace Pharmacy_Management_System
{
    public partial class Retail_Sales_Details : Form
    {
        SqlConnection con;
        SqlCommand com,com1;
        String sql,sql1;
        public Retail_Sales_Details()
        {
            InitializeComponent();
            con = new SqlConnection(@"Data
Source=.\SQL EXPRESS;AttachDbFilename=C:\Users\Jathurshan
```

```

Sumaniran\Desktop\Pharmacy_Management_System\Pharmacy_Management_System\Medicine.mdf;Integrated Security=True;User Instance=True");
}

private List<Retail> shoppingCart = new List<Retail>();

private void label6_Click(object sender, EventArgs e)
{

}

private void label15_Click(object sender, EventArgs e)
{
}

private void label5_Click(object sender, EventArgs e)
{
}

private void textBox12_TextChanged(object sender, EventArgs e)
{
    if ((textBox12.Text == ""))
    {
        MessageBox.Show("Sorry,please add the full details", "Warning");
    }

    con.Open();

    com = new SqlCommand("select mName,expiryDate,dosage,sellingPrice from details_M
where CONVERT (varchar,mID) ='" + textBox12.Text + "'", con);
    //sql = "select * from details where CONVERT (varchar,mID) ='" + textBox12.Text +
""";
    SqlDataReader dr;
    dr = com.ExecuteReader();
    while (dr.Read())
    {
        textBox2.Text = dr[0].ToString();
        textBox14.Text = dr[1].ToString();
        textBox4.Text = dr[2].ToString();
        textBox5.Text = dr[3].ToString();

    }
    dr.Close();
    con.Close();

/* SqlDataAdapter da = new SqlDataAdapter(sql, con);

DataTable di = new DataTable();
da.Fill(di);
dataGridView1.DataSource = di;

*/
}

private void textBox3_TextChanged(object sender, EventArgs e)
{
    double anDouble;
    anDouble = Convert.ToInt32(textBox5.Text);
}

```

```

anDouble = Double.Parse(textBox5.Text);
if (!string.IsNullOrEmpty(textBox3.Text) && !string.IsNullOrEmpty(textBox5.Text))
    textBox6.Text = (Double.Parse(textBox3.Text) * anDouble).ToString();
}

private void Retail_Sales_Details_Load(object sender, EventArgs e)
{
    textBox13.Text = dateTimePicker1.Value.ToShortDateString();

    con.Open();

    com = new SqlCommand("select max(convert(int,billNo)) from retails", con);

    SqlDataReader dr2;
    dr2 = com.ExecuteReader();
    String x1 = "";
    while (dr2.Read())
    {
        x1 = dr2[0].ToString();

    }
    dr2.Close();

    double anDouble;
    anDouble = Convert.ToInt32(x1);
    anDouble = Double.Parse(x1);

    anDouble = anDouble + 1;
    textBox1.Text = anDouble.ToString();
    con.Close();

    con.Open();
    SqlCommand cmd = new SqlCommand("SELECT billNo FROM retails", con);

    SqlDataReader reader = cmd.ExecuteReader();
    AutoCompleteStringCollection MyCollection = new AutoCompleteStringCollection();
    while (reader.Read())
    {
        MyCollection.Add(reader.GetString(0));
    }
    textBox1.AutoCompleteCustomSource = MyCollection;
    con.Close();

    con.Open();
    SqlCommand cmd0 = new SqlCommand("SELECT mID FROM details_M", con);

    SqlDataReader reader0 = cmd0.ExecuteReader();
    AutoCompleteStringCollection MyCollection0 = new AutoCompleteStringCollection();
    while (reader0.Read())
    {
        MyCollection0.Add(reader0.GetString(0));
    }
    textBox12.AutoCompleteCustomSource = MyCollection0;
    con.Close();
}

private void button1_Click(object sender, EventArgs e)
{
    try
    {

```

```

        con.Open();

        com = new SqlCommand("select SUM(amount) from retails where billNo ='" +
textBox1.Text + "'", con);
        //sql = "select * from details where CONVERT (varchar,mID) ='" +
textBox12.Text + "'";
        SqlDataReader dr;
        dr = com.ExecuteReader();
        while (dr.Read())
        {
            textBox7.Text = dr[0].ToString();
        }
        dr.Close();

        con.Close();

    }

    catch (Exception ex2)
    {
        MessageBox.Show("this Bill No is already available");
    }

}

private void textBox8_TextChanged(object sender, EventArgs e)
{
    if (!string.IsNullOrEmpty(textBox7.Text) && !string.IsNullOrEmpty(textBox8.Text))
        textBox9.Text = ((Double.Parse(textBox7.Text)) - ((Double.Parse(textBox7.Text) *
Double.Parse(textBox8.Text)/100))).ToString();
}

private void textBox10_TextChanged(object sender, EventArgs e)
{
    if (!string.IsNullOrEmpty(textBox9.Text) &&
!string.IsNullOrEmpty(textBox10.Text))
        textBox11.Text = (Double.Parse(textBox10.Text) -
Double.Parse(textBox9.Text)).ToString();
}

private void button2_Click(object sender, EventArgs e)
{
    try
    {
        string ab, cd, ef, gh, ij, kl, mn, mk;

        ij = textBox1.Text;
        kl = textBox13.Text;
        ab = textBox7.Text;
        cd = textBox8.Text;
        ef = textBox9.Text;
        mn = textBox10.Text;
        gh = textBox11.Text;

        if ((ij == "") || (kl == "") || (ab == "") || (cd == "") || (ef == "") || (ef
== "") || (gh == ""))
        {
            MessageBox.Show("please add the full details", "Warning");
        }
    }
}

```

```

        else
        {
            con.Open();
            sql = "insert into purchase values('" + ij + "','" + kl + "','" + ab +
"','" + cd + "','" + ef + "','" + mn + "','" + gh + "')";
            com = new SqlCommand(sql, con);
            com.ExecuteNonQuery();

            con.Close();
            MessageBox.Show("Bill is Finalised", "Add", MessageBoxButtons.OK,
MessageBoxIcon.Information);

            //dataGridView1.Rows.Clear();
        }
    }

    catch (Exception ex1)
    {
        MessageBox.Show("this Bill No is already available");
    }
}

private void textBox7_TextChanged(object sender, EventArgs e)
{
}

private void button4_Click(object sender, EventArgs e)
{
    Login_Admin la = new Login_Admin();
    la.Show();
    this.Hide();
}

private void button3_Click(object sender, EventArgs e)
{
    Application.Exit();
}

private void button6_Click(object sender, EventArgs e)
{
    string x, ab, cd, ef, gh, ij, kl, mn, op, qr, st, uv, xy, mk, im, br, gn;

    ab = textBox1.Text;
    cd = textBox13.Text;
    ef = textBox12.Text;
    gh = textBox2.Text;
    ij = textBox3.Text;
    kl = textBox14.Text;
    mn = textBox4.Text;
    op = textBox5.Text;
    qr = textBox6.Text;

    if (textBox1.Text != "" && ((ab != "") || (cd != "") || (ef != "") || (gh != ""))
|| (ij != "") || (kl != "") || (mn != "") || (op != "") || (qr != ""))
    {
        if (IsValidated())
        {
            Retail item = new Retail()
            {

```

```

        mName = textBox2.Text,
        quantity = textBox3.Text.Trim(),
        dosage = textBox4.Text.Trim(),
        mID = textBox12.Text.Trim(),
        expiryDate = textBox14.Text.Trim(),
        unitPrice = textBox5.Text.Trim(),
        amount = textBox6.Text.Trim(),
    };

    shoppingCart.Add(item);
    dataGridView1.DataSource = null;

    dataGridView1.DataSource = shoppingCart;

}
try
{
    // string x, ab, cd, ef, gh, ij, kl, mn, op, qr, st, uv, xy, mk, im, br, gn;

    ab = textBox1.Text;
    cd = textBox13.Text;
    ef = textBox12.Text;
    gh = textBox2.Text;
    ij = textBox3.Text;
    kl = textBox14.Text;
    mn = textBox4.Text;
    op = textBox5.Text;
    qr = textBox6.Text;

    if ((ab == "") || (cd == "") || (ef == "") || (gh == "") || (ij == "") || (kl ==
    "") || (mn == "") || (op == "") || (qr == ""))
    {
        MessageBox.Show("please add the full details", "Warning");
    }
    else
    {
        con.Open();
        com = new SqlCommand("select SUM(quantity) from details_M where mID = '" +
textBox12.Text + "'", con);

        SqlDataReader dr2;
        dr2 = com.ExecuteReader();
        String x1 = "";
        while (dr2.Read())
        {
            x1 = dr2[0].ToString();

        }
        dr2.Close();

        double anDouble;
        anDouble = Convert.ToInt32(x1);
        anDouble = Double.Parse(x1);

        double anDouble1;
        anDouble1 = Convert.ToInt32(ij);
        anDouble1 = Double.Parse(ij);
        if (anDouble <= anDouble1)
        {

```

```

        MessageBox.Show("Your available quantity is low ..... ");
    }

    else
    {
        sql = "insert into retails values('"+ ab + "','" + cd + "','" + ef +
        "','" + gh + "','" + ij + "','" + kl + "','" + mn + "','" + op + "','" + qr + "')";
        com1 = new SqlCommand(sql, con);
        com1.ExecuteNonQuery();

        //MessageBox.Show("successfully added", "Add", MessageBoxButtons.OK,
        MessageBoxIcon.Information);

        com = new SqlCommand("select SUM(quantity) from details_M where mID = '" +
        + textBox12.Text + "'", con);

        SqlDataReader dr1;
        dr1 = com.ExecuteReader();
        String x5 = "";
        String Quant = "";
        while (dr1.Read())
        {
            x5 = dr1[0].ToString();

        }
        dr1.Close();

        double anDouble5;
        anDouble5 = Convert.ToInt32(x5);
        anDouble5 = Double.Parse(x5);
        if (!string.IsNullOrEmpty(x5) && !string.IsNullOrEmpty(textBox3.Text))
            Quant = (anDouble5 - Double.Parse(textBox3.Text)).ToString();

        SqlCommand cmd = new SqlCommand("UPDATE details_M SET quantity = '" +
        Quant + "' WHERE mID ='" + textBox12.Text + "'", con);
        cmd.ExecuteNonQuery();
        MessageBox.Show("Details are modified");



        sql1 = "select mID,mName,quantity,expiryDate,dosage,unitPrice,amount from
retails where billNo ='"+ textBox1.Text + "'";

        SqlDataAdapter da = new SqlDataAdapter(sql1, con);

        DataTable di = new DataTable();
        da.Fill(di);
        // dataGridView1.DataSource = di;
    }
    con.Close();

```

```

        }

    }

    catch (Exception ex)
    {
        MessageBox.Show("this Bill No is already available");
    }

}

private bool IsValidated()
{
    return true;
}

private void button7_Click(object sender, EventArgs e)
{

}

private void printPreviewDialog1_Load(object sender, EventArgs e)
{

}

private void printDocument1_PrintPage(object sender,
System.Drawing.Printing.PrintPageEventArgs e)
{
    Image image = Resources.atlanta;
    e.Graphics.DrawImage(image, 125, 20, image.Width, image.Height);

    e.Graphics.DrawString("Date: " + DateTime.Now, new Font("Arial", 12,
FontStyle.Regular), Brushes.Black, new Point(25, 160));

    e.Graphics.DrawString("Invoice No: " + textBox1.Text.Trim(), new Font("Arial",
12, FontStyle.Regular), Brushes.Black, new Point(25, 190));
    e.Graphics.DrawString("-----", new
Font("Arial", 12, FontStyle.Regular), Brushes.Black, new Point(25, 235));

    e.Graphics.DrawString("Medicine Name ", new Font("Arial", 12,
FontStyle.Regular), Brushes.Black, new Point(30, 255));
    e.Graphics.DrawString("Quantity ", new Font("Arial", 12, FontStyle.Regular),
Brushes.Black, new Point(240, 255));
    e.Graphics.DrawString("Dosage ", new Font("Arial", 12, FontStyle.Regular),
Brushes.Black, new Point(410, 255));
    e.Graphics.DrawString("Unit Price ", new Font("Arial", 12, FontStyle.Regular),
Brushes.Black, new Point(560, 255));
    e.Graphics.DrawString("Total Price ", new Font("Arial", 12, FontStyle.Regular),
Brushes.Black, new Point(710, 255));
    e.Graphics.DrawString("-----", new
Font("Times and Newroman", 12, FontStyle.Regular), Brushes.Black, new Point(25, 270));

    int y = 295;

    foreach (var i in shoppingCart)

```

```

    {

        e.Graphics.DrawString(i.mName, new Font("Arial", 12, FontStyle.Regular),
Brushes.Black, new Point(30, y));
        e.Graphics.DrawString(i.quantity.ToString(), new Font("Arial", 12,
FontStyle.Regular), Brushes.Black, new Point(240, y));
        e.Graphics.DrawString(i.dosage.ToString(), new Font("Arial", 12,
FontStyle.Regular), Brushes.Black, new Point(430, y));
        e.Graphics.DrawString(i.unitPrice.ToString(), new Font("Arial", 12,
FontStyle.Regular), Brushes.Black, new Point(560, y));
        e.Graphics.DrawString(i.mID.ToString(), new Font("Arial", 12,
FontStyle.Regular), Brushes.Black, new Point(710, y));

        y += 30;
    }
    e.Graphics.DrawString("-----", new
Font("Times New Roman", 12, FontStyle.Regular), Brushes.Black, new Point(25, y));

    e.Graphics.DrawString("Total Amount : " + textBox7.Text.Trim(), new
Font("Arial", 12, FontStyle.Regular), Brushes.Black, new Point(550, y + 30));
    e.Graphics.DrawString("Discount(%) : " + textBox8.Text.Trim(), new
Font("Arial", 12, FontStyle.Regular), Brushes.Black, new Point(550, y + 60));
    e.Graphics.DrawString("Total Pay : " + textBox9.Text.Trim(), new
Font("Arial", 12, FontStyle.Regular), Brushes.Black, new Point(550, y + 90));
}

public string quantity { get; set; }

public string mName { get; set; }

private void button7_Click_1(object sender, EventArgs e)
{
    printPreviewDialog1.Document = printDocument1;
    printPreviewDialog1.ShowDialog();

    textBox2.Text = "";
    textBox3.Text = "";
    textBox4.Text = "";
    textBox5.Text = "";
    textBox6.Text = "";
    textBox7.Text = "";
    textBox8.Text = "";
    textBox9.Text = "";
    textBox10.Text = "";
    textBox11.Text = "";
    textBox12.Text = "";
    textBox14.Text = "";

    textBox13.Text = dateTimePicker1.Value.ToShortDateString();
    dataGridView1.DataSource = null;

    con.Open();

    com = new SqlCommand("select max(convert(int,billNo)) from retails", con);
}

```

```

SqlDataReader dr2;
dr2 = com.ExecuteReader();
String x1 = "";
while (dr2.Read())
{
    x1 = dr2[0].ToString();

}
dr2.Close();

double anDouble;
anDouble = Convert.ToInt32(x1);
anDouble = Double.Parse(x1);

anDouble = anDouble + 1;
textBox1.Text = anDouble.ToString();
con.Close();
}

private void textBox3_KeyPress(object sender, KeyPressEventArgs e)
{
    e.Handled = char.IsNumber(e.KeyChar) || e.KeyChar == 8 ? false : true;
}

private void textBox8_KeyPress(object sender, KeyPressEventArgs e)
{
    e.Handled = char.IsNumber(e.KeyChar) || e.KeyChar == 8 ? false : true;
}

private void textBox10_KeyPress(object sender, KeyPressEventArgs e)
{
    e.Handled = char.IsNumber(e.KeyChar) || e.KeyChar == 8 ? false : true;
}

private void dateTimePicker1_ValueChanged(object sender, EventArgs e)
{

}

private void textBox12_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        textBox3.Focus();
    }
}

private void textBox3_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        button6.PerformClick();
    }
}

private void textBox8_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        button1.PerformClick();
    }
}

```

```

private void textBox10_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        textBox11.Focus();
    }
}

private void textBox11_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        button2.PerformClick();
    }
}

private void textBox1_TextChanged(object sender, EventArgs e)
{

}

private void dataGridView1_MouseDown(object sender, MouseEventArgs e)
{
    try
    {
        if (e.Button == System.Windows.Forms.MouseButtons.Right)
        {
            var hit = dataGridView1.HitTest(e.X, e.Y);
            dataGridView1.Rows[hit.RowIndex].Selected = true;
            contextMenuStrip1.Show(dataGridView1, e.X, e.Y);
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("There are some problem..!!!");
    }
}

private void deletItemToolStripMenuItem_Click(object sender, EventArgs e)
{
    try
    {
        if ((textBox1.Text == ""))
        {
            MessageBox.Show("Sorry,please add the full details", "Warning");
        }
        else
        {

            con.Open();

            DialogResult result = MessageBox.Show("Do you want to Remove the
Member ?", "Warning", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

            if (result == DialogResult.Yes)
            {
                int index = dataGridView1.CurrentCell.RowIndex;

                shoppingCart.RemoveAt(index);
            }
        }
    }
}

```

```

        dataGridView1.DataSource = null;
        dataGridView1.DataSource = shoppingCart;

        sql = ("delete from retails where billNo ='" + textBox1.Text +
        "'");

        com = new SqlCommand(sql, con);

        com.ExecuteNonQuery();

        MessageBox.Show("Successfully deleted", "Delete",
        MessageBoxButtons.OK, MessageBoxIcon.Information);

        // textBox1.Text = "";

        sql1 = "select
mID,mName,quantity,expiryDate,dosage,unitPrice,amount from retails where billNo ='" +
textBox1.Text + "'";

        SqlDataAdapter da = new SqlDataAdapter(sql1, con);

        DataTable di = new DataTable();
        da.Fill(di);
        //dataGridView1.DataSource = di;

        com = new SqlCommand("select SUM(quantity) from details_M where
mID = '" + textBox12.Text + "'", con);

        SqlDataReader dr1;
        dr1 = com.ExecuteReader();
        String x6 = "";
        String Quant1 = "";
        while (dr1.Read())
        {
            x6 = dr1[0].ToString();

        }
        dr1.Close();

        double anDouble6;
        anDouble6 = Convert.ToInt32(x6);
        anDouble6 = Double.Parse(x6);
        if (!string.IsNullOrEmpty(x6) &&
!string.IsNullOrEmpty(textBox3.Text))
            Quant1 = (anDouble6 +
Double.Parse(textBox3.Text)).ToString();

        SqlCommand cmd = new SqlCommand("UPDATE details_M SET quantity =
'" + Quant1 + "' WHERE mID ='" + textBox12.Text + "'", con);
        cmd.ExecuteNonQuery();
        MessageBox.Show("Details are modified");
    }
    else
    {
        MessageBox.Show("not deleted", "Warning", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        // textBox1.Text = "";

    }
}

```

```
        con.Close();
    }

    catch (Exception ex)
    {
        MessageBox.Show("There are some problem..!!!");
    }
}

private void textBox9_TextChanged(object sender, EventArgs e)
{
}

private void button6_KeyDown(object sender, KeyEventArgs e)
{
    if(e.KeyCode==Keys.Enter)
    {
        textBox12.Focus();
    }
}

private void textBox6_TextChanged(object sender, EventArgs e)
{
}

}
}
```

Special features

In visual studio 2010

- In built, designing and implementation tools
- Debugger
- Inbuilt C# library for implementation
 - using System.Security.Cryptography;(for hashing password).
 - using System.Text.RegularExpressions;(for email, phone no and NIC no validation).
 - using Pharmacy_Management_System.Properties;(for using the properties of project).
 - using System.Collections.Specialized;(for suggestion append of combo box).
 - using System.Configuration;(for system configuration).
- SQL server to connect the database.

Above special features of the visual studio 2010, we have used for our project.

In our application

- According to the special features this application will check all the validation of particular fields for example e-mail, telephone number, NIC No, string and character validations of each textbox and number of inputs of each textbox.
- In billing section, we automatically find the medicine name, dosage, unit price with typing the medicine id and it will automatically calculate the total amount with the discount and balance, finally keep the records of sales.

Challenges in implementation

For the implementation of this application, we already decided to do a standalone system, but we have two choices to choose which programming language we want to use that are java and C#. After our group discussion, we finalised to go for C# programming language. We interested to learn a new programming language. But, some members of our group not ready to learn and two of them learned this language.

Here, we mentioned that the programming language we used for this project is new to us. But, we have already study and familiar with java and C# is little different

from java, we have used object-oriented concept to build our application this is our first challenge.

Another thing was version problem of our platform, in early stage we used visual studio 2015 to implementation. Some of issues raised during the implementation that are difficult to connect database, sometimes connected database was not working properly, SQL server problem, it got load to start and some other little issues. Finally, we used visual studio 2010.

Real data of the stock was the challenge to us. We got the requirements to implement but did not get the real data to check.

And many conflicts raised between group members based on coding work during the implementation in finding correct methods to improve our application.

Testing

Testing is done in order to align the product with the user. The functionalities and functions are tested in order to point out the defects and errors that were made during the development phases so that the quality of the product and the functionalities of the product are improved. The testing plan defines the items to be tested and the functions are selected bases on the importance of the functions, and the risk of the functions on the user's viewpoint.

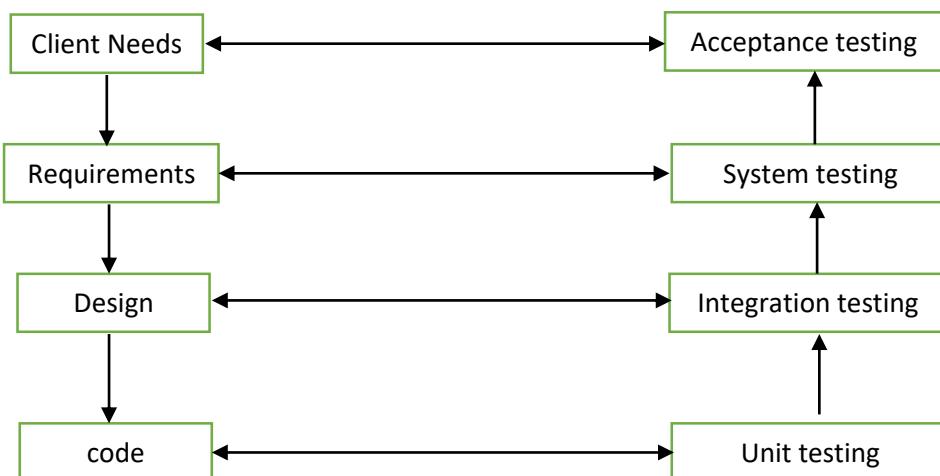
The test cases were designed corresponding with the use case descriptions. They were executed manually and the results were recorded. The bugs identified were corrected and tested again.

In the testing process of this system both black box testing and white box testing is being used.

- **White box testing** is mostly carried out while the coding is taking place. Each statement, conditions and decision structures test soon after finishing the coded. The purpose of using white box testing for this system is that White box testing enables to identify the small bugs that could be created by the mistakes of coding and due to the errors in the decision logics, and it enables the developers to prevent those errors before integrating the recently developed contents together with the system.

- **Black box testing** has used once after a particular functionality is completely developed. A set of random values used for the testing and the errors and bugs which are generated is identified and fixed. The focus over user error can be tested at this stage.

In order to uncover the errors, present in different phases we have the concept of levels of testing. The basic levels of testing are:



A series of testing is done for the proposed system before the system is ready for the user acceptance testing.

Acceptance testing

Acceptance testing is often done by the customers to ensure that the delivered product meets the requirements and works as the customer expected. It falls under the class of black box testing.

System testing

System testing is the testing to ensure that by putting the software in different environments (eg. Operating system) it still works. System testing is done with full system implementation and environment. It falls under the class of black box testing.

Integration testing

Integration testing is testing in which a group of components are combined to produce output, also the interaction between software and hardware is tested in integration testing. If software and hardware components have any relation. It may fall under both white box and black box testing.

Unit testing

Unit testing is the testing of an individual unit or group of related units. It falls under the class of white box testing. It is often done by the programmer to test that unit programmer has implemented is producing expected output against given input.

Operating Systems	Windows 10 Pro
Hardware	Intel Core i5 2.3GHz 4GB RAM
Software	Windows 10 Operating System (64 bit)
Debugger and testing tools	visual Studio Debugger

Add medicine field			
Test case descriptor	Input data	Expected output	status
Category	Medicine category	Display the successfully added message	pass
Medicine Id	Unique Id		
Medicine name	Medicine name		
Dosage	Dosage of medicine		
Manufacturer	Manufacturer name		
Imported by	Company name		
Brand name	Brand name		
Supplier Id	Unique Id		
Generic name	Unique name		
Quantity	Amount of stock		
Entry date	Date		
Manufacturer date	Date		
Expiry date	Date		
Buying price	Price rupees		
Selling price	Price rupees		

Validate medicine details Steps: 1. Enter all the required fields 2. Unique inputs	Incorrect data	Display the error type	pass
---	----------------	------------------------	------

View medicine field			
Test case descriptor	Input data	Expected output	status
Choose type	Category of medicine	Display the medicine details	pass
Medicine name	Medicine name		
Validate view medicine details Steps: 1. Enter all the required fields	Incorrect data	Display error type	pass

View expiry of medicine field			
Test case descriptor	Input data	Expected output	status
Search date	date	Display the medicine details	pass
Validate view medicine details Steps: 1. Enter all the required fields	Incorrect data	Display error type	pass

modify medicine field			
Test case descriptor	Input data	Expected output	status
Choose type	Category of medicine		
Medicine name	Medicine name		pass

Medicine Id	Unique Id	Display the medicine details	
Category	Medicine category		
Medicine Id	Unique Id		
Medicine name	Medicine name		
Dosage	Dosage of medicine		
Manufacturer	Manufacturer name		
Imported by	Company name		
Brand name	Brand name		
Supplier Id	Unique Id		
Generic name	Unique name		
Quantity	Amount of stock		
Entry date	Date		
Manufacturer date	Date		
Expiry date	Date		
Buying price	Price rupees		
Selling price	Price rupees		
Validate view medicine details Steps: 1. Enter all the required fields	Incorrect data	Display error type	pass

Delete medicine field			
Test case descriptor	Input data	Expected output	status
Choose type	Category of medicine		
Medicine name	Medicine name	Display the medicine details	pass
Medicine Id	Unique Id		
Validate view medicine details Steps: 1. Enter all the required fields 2. Unique input	Incorrect data	Display error type	pass

Add staff field

Test case descriptor	Input data	Expected output	status
Staff name	Full name	Display the successfully added message	pass
Date of birth	date		
Gender	Male/ Female		
Address	Address		
Contact no	number		
NIC No	Unique no		
Age	Age		
Salary	Salary rupees		
Joining date	date		
E-mail	Unique e-mail id		
username	name	Display the error type	pass
Password	password		
role	Job of staff		
Validate staff details Steps: 1. Enter all the required fields 2. Unique inputs	Incorrect data	Display the error type	pass

View staff field			
Test case descriptor	Input data	Expected output	status
Staff name	Full name	Display the staff details	pass
NIC No	Unique Id		
Validate view staff details Steps: 1. Enter all the required fields	Incorrect data	Display error type	pass

Modify staff field			
Test case descriptor	Input data	Expected output	status

Staff name	Full name	Display the staff details Display the successfully updated message	pass pass
NIC No	Unique Id		
Staff name	Full name		
Date of birth	date		
Gender	Male/ Female		
Address	Address		
Contact no	number		
NIC No	Unique no		
Age	Age		
Salary	Salary rupees		
Joining date	date		
E-mail	Unique e-mail id		
username	name		
Password	password		
role	Job of staff		
Validate staff details Steps: 1. Enter all the required fields 2. Unique inputs	Incorrect data	Display the error type	pass

Delete staff field			
Test case descriptor	Input data	Expected output	status
Staff name	Full name	Display the staff details	pass
NIC No	Unique Id		
Validate view staff details Steps: 1. Enter all the required fields	Incorrect data	Display error type	pass

Add supplier field			
Test case descriptor	Input data	Expected output	status
Supplier name	Full name	Display the successfully added message	pass
Supplier Id	Unique no		
Company name	Company name		
License number	Unique no		
E-mail	Unique mail id		
Contact number	number		
address	address		
Validate supplier details Steps: 1. Enter all the required fields 2. Unique inputs	Incorrect data	Display the error type	pass

View supplier field			
Test case descriptor	Input data	Expected output	status
Supplier name	Full name	Display the staff details	pass
Supplier Id	Unique Id		
Validate view supplier details Steps: 1. Enter all the required fields	Incorrect data		

Modify supplier field			
Test case descriptor	Input data	Expected output	status
Supplier name	Full name	Display the supplier details	pass
Supplier Id	Unique Id		
Supplier name	Full name		
Supplier Id	Unique no		
Company name	Company name		
License number	Unique no		
E-mail	Unique mail id		

Contact number address	number address		
Validate supplier details Steps: 1. Enter all the required fields 2. Unique inputs	Incorrect data	Display the error type	pass

Delete supplier field			
Test case descriptor	Input data	Expected output	status
Supplier name	Full name	Display the staff details	pass
Supplier Id	Unique Id		
Validate view supplier details Steps: 1. Enter all the required fields	Incorrect data	Display error type	pass

Retail sales field			
Test case descriptor	Input data	Expected output	status
Bill no	number		
Date	date		
Medicine Id	Unique Id		
Quantity	number		
Discount	Number in percentage		
Paid amount	Amount in rupees		
Validate sales details Steps: 1. Enter all the required fields 2. Unique inputs	Incorrect data	Display the error type	pass

View sales field

Test case descriptor	Input data	Expected output	status
Bill no	number	Display the sales details	pass
Date	date		
Validate view sales details Steps: 1. Enter all the required fields	Incorrect data	Display error type	pass

View purchase field

Test case descriptor	Input data	Expected output	status
Medicine name	name		
Medicine Id	Unique Id	Display the purchase details	pass
Supplier Id	Unique Id		
Validate view purchase details Steps: 1. Enter all the required fields	Incorrect data	Display error type	pass

Login field

Test case descriptor	Input data	Expected output	status
Input username	Username	Accept the username and password, display the main field	
Input password	password		pass
Validate login details Steps: 1. Enter all the required fields	Incorrect data	Display error message and stay login page	pass

Change password field			
Test case descriptor	Input data	Expected output	status
Input username	Username	Accept the username and password, display updated message	pass
Input old password	Old password		
Input new password	New password		
Input confirm password	New password		
Validate login details Steps: 1. Enter all the required fields	Incorrect data	Display error message and stay change password page	pass

In this section, we have mentioned the test types, which we have followed when implementing the testing procedure and test cases of the system. We have further described test strategies used, test plan, sample test cases and presents the report of the overall test phase.

Discussions and challenges

In the starting point of our project, the members of group 3S discussed about the topic of the project. But, we could not get a stable decision on project topic and we suggested some project topics to our mentor, that was the application for CD/DVD shop management, health care centre of University of Jaffna and school management system, these topics were already done by our seniors for their group project and finally we got the permission to do a “Pharmacy Management System”, from our mentor.

Our members have already decided to do a standalone system and chosen visual C# as a programming language in the platform of visual studio. And also we planned to do an efficient complete application for our client.

Next challenge raised to find the suitable client for us. We have faced many problems during our project time. Because we needed the requirement to prepare the project proposal, through this we wanted to get approval letter from our university, for start our project work.

Already we included in previous sections, the challenges were faced by us, during the requirement gathering, system designing, and implementation works.

Another way of problem was team unity and cooperation, some of the team members not interested and do not care about the group project in the designing and implantation work. Interested people could not contact other members on needed time to meet client and mentor of our group. This type of problem made us frustrated. But, we manage these problems in better way.

Software engineering techniques used in our project and we learned these techniques in short time of period. In daily basis we have learned new important things of software engineering techniques, how to take of the project and how to guide a group.

And also some issues during the implementation, sometimes our coding is correct by some bugs occurred.

Finally, project report is preparation was the challenge to us. Because, member of group 3S are in their semester vacation at their home town, they could not meet each other, by telephone calls, we managed that problem also.