VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELGAUM-590018

A Mini-Project Report

On

"PHARMACY MANAGEMENT SYSTEM"

A Mini-project report submitted in partial fulfillment of the requirements for the award of the degree of ***Bachelor of Engineering in Computer Science and Engineering*** *of*
Visvesvaraya Technological University, Belgaum.

Submitted by:

Mohammed Firdous Mehdi	**1AM18CS104**
Md Asad	**1AM18CS100**
Mohammed Haseeb Khan	**1AM18CS105**
Mohammed Uzair Dawalji	**1AM18CS108**

Under the Guidance of:
Mrs. V.R. Srividhya
(Asst. Prof., Dept. of CSE)

Department of Computer Science and Engineering

AMC Engineering College,
18th K.M, Bannerghatta Main Road, Bangalore-560 083

2020-2021

AMC Engineering College,

18th K.M, Bannerghatta Main Road, Bangalore-560 083

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the mini-project work entitled **"PHARMACY MANAGEMENT SYSTEM"** has been successfully carried out by Mohammed Firdous Mehdi(1AM18CS104), Md Asad (1AM18CS100), Mohammed Haseeb khan(1AM18CS105), Mohammed Uzair Dawalji (1AM18CS108), the Bonafide students of **AMC Engineering College** in partial fulfilment of the requirements for the award of degree in **Bachelor of Engineering in Computer Science and Engineering** of **Visvesvaraya Technological University, Belgaum** during academic year 2020-2021. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements in respect of project work for the said degree.

Guide:

Mrs. SRIVIDHYA V.R.	**Dr. LATHA C A**	**Dr. A.G. NATARAJ**
Asst. Prof., Dept. of CSE, AMCEC	**HOD, Dept. of CSE, AMCEC**	**Principal, AMCEC**

Examiners: **Signature with Date**

1. ____________________

2. ____________________

ACKNOWLEDGEMENT

The joy and satisfaction that accompany the successful completion of any task would be incomplete without the mention of those who made it possible. We are gladto express our gratitude towards our prestigious institution **AMC ENGINEERING COLLEGE** for providing us with utmost knowledge, encouragement and the maximum facilities in undertaking this project.

We wish to express sincere thanks to our respected chairman **Dr.K.R.Paramahamsa** and beloved principal **Dr. A.G. Nataraj** for all their support.

We express our deepest gratitude and special thanks to **Dr. Latha C.A.,H.O.D, Dept. Of Computer Science And Engineering**, for all her guidance and encouragement.

We sincerely acknowledge the guidance and constant encouragement of our mini- project guide, **Mrs. V. R. Srividhya, Assistant Prof., Dept. Of Computer Science And Engineering.**

NAME	**USN**
Mohammed Firdous Mehdi	**1AM18CS104**
Md Asad	**1AM18CS100**
Mohammed Haseeb Khan	**1AM18CS105**
Mohammed Uzair Dawalji	**1AM18CS108**

ABSTRACT

The purpose of "PHARMACY MANAGEMENT SYSTEM" is to automate the existing manual system by the help of computerized equipments and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same.The required software and hardware are easily available and easy to work with.

Pharmacy Management System, as described above, can lead to error free, secure,reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in betterutilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information t hat is not relevant, while being able to reach the information.

The aim is to automate its existing manual system by the help of computerized euipmentsand full-fledged computer software, fulfilling their requirements. so that their valuable datainformation can be stored for a longer period with easy accessing and manipulation of the same. Basically the project describes how to manage for good performance and better services for the clients.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user-friendly.

Every organization, whether big or small, has challenges to overcome and managing the informations of Medicines, Pharmacy, Company, Sells, Inventory. Every Pharmacy Management System has different Pharmacy needs, therefore we design exclusive employee management systems that are adapted to your managerial requirements.

CONTENTS

Chapter 1

INTRODUCTION

This application is used by two users:

1) Admin
2) Employees

The "PHARMACY MANAGEMENT SYSTEM" has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. Moreover this system is designed for the particular need of the company to carry out operations in a smooth and effective manner

Pharmacy Management System has the modules.

List of Modules:

- Login Page
- Home Page
- Company
- Messaging
- Purchase
- Drugs
- Sales
- User/Setting

1.1 OBJECTIVES OF THE PROJECT:

The main objective of the Project on Pharmacy Management System is to manage the details of Pharmacy, Medicines, Stocks, Company, Inventory. It manages all the information about Pharmacy, Sells, Inventory, Pharmacy. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the Pharmacy, Medicines, Sells, Stocks. It tracks all the details about the Stocks, Company, Inventory.

Functionalities provided by Pharmacy Management System are as follows:

- Provides the searching facilities based on various factors. Such as Pharmacy, StocksCompany, inventory.
- Pharmacy Mangement System also manage the sell details for company, inventory and Pharmacy
- Tracks all the information of Drugs ,Sales and comapany Details.
- Shows the information and description of the Pharmacy and stocks.
- Editing, Adding and Updating records is easy and improved, Which results in proper resource management of Pharmacy data.
- Manage the information of Company.
- Integration of all records of inventory.
- To increaese efficiency of managing the drugs of Pharmacy store.

1.2 SCOPE OF PHARMACY MANAGEMENT SYSTEM:

Manual handling of the record is time consuming and highly
It may help collecting perfect management in details.

In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the management of passed year perfectly and vividly.

It also helps in current all works relative to Pharmacy Management System. It will be also reduced the cost of collecting the management & collection procedure will go on smoothly.

Our project aims at Business process automation, i.e. we have tried to computerize various processes of Pharmacy Management System.

In computer system the person has to fill the various forms & number of copies of the forms can be easily generated at a time.

In computer system, it is not necessary to create the manifest but we can directly print it, which saves our time.

To assist the employees in capturing the effort spent on their respective working areas.

To utilize resources in an efficient manner by increasing their productivity through automation.

The system generates types of information that can be used for various purposes.

It satisfy the user requirement

Be easy to understand by the user and operator

Be easy to operate

Have a good user interface

Be expandable

Delivered on schedule within the budget.

Chapter 2

SYSTEM SPECIFICATION

2.1 HARDWARE REQUIREMENTS

Minimum:

PROCESSOR: i3 4th gen

RAM: 4GB

HARD DISK: 40MB

Recommended:

PROCESSOR: i9 10th gen

1GB+DDR RAM

40GB HARD DISK DRIVE

2.2 SOFTWARE REQUIREMENTS

OS: Windows/MacOs/Linux, 64 bit system

JRE and JDK

MySQL server (WAMP or XAMPP or any)

FRONTEND:
JAVA PROGRAMMING LANGUAGE

IDE:
APACHE NETBEANS (latest vs 12.2)

CONNECTOR:
MySQL connector jar file (vs 5.1.4)

- BACK END: MYSQL

It is a relational database whose components (tables, forms, queries) are linked (related).

The linkages between database components are created by making

relationship links between them.

The relationship can be:

- ♦ One component and another (one-one relationship)
- ♦ One component related to several other components(one-many)
- ♦ Several database components(many-many)

Creation of relationships between the database components reduces data redundancies and enhances ease of access of the information.

The design of a DBMS depends on its architecture. It can be centralized or decentralized or hierarchical. The architecture of a DBMS can be seen as either single tier or multi-tier.

Chapter 3

DESIGN

3.1 Pharmacy Management Database System

- The Database design is a collection of processess that facilitate the designing, development, implementation and maintenance of enterprise data management systems.
- It helps to produce database systems:
  That meet the requirements of the users and have high performance.

SYSTEM ANALYSIS

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information about the Pharmacy Management System to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system.The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action. A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system.

STEP 1:

Entities are:

1. Company
2. Drug
3. Purchase
4. Sale
5. User
6. Login

STEP 2:

RELATIONS:

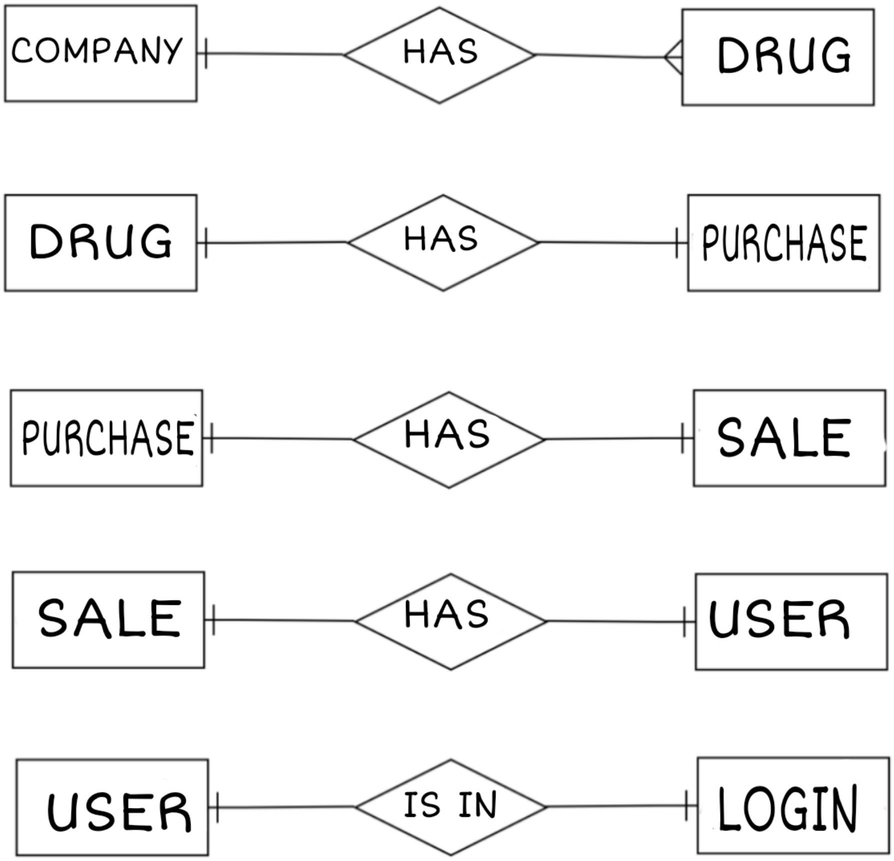


Figure 3.1: Relation of Entities

STEP 3:

Key attributes:

- COMPANY - NAME
- DRUG - BARCODE
- PURCHASE - COMPANY_NAME
- SALE - BARCODE
- USER - ID
- LOGIN - ID
- INBOX - SENDER_ID

STEP 4:

Other attributes

- COMPANY - NAME, ADDRESS, PHONE
- DRUG - TNAME, TYPE, BARCODE , DOSE, CODE, COST-PRICE, SELL-PRICE, EXPIRY, COMPANY_NAME, PRODUCTION-DATE, EXPIRATION-DATE, PLACE, QUANTITY
- HISTORY_SALE - RUSER_NAME, BARCODE, DOSE, TYPE, PRICE, AMOUNT, DATE, TIME, NAME, QUANTITY
- PURCHASE - COMPANY_NAME, BARCODE, TYPE, PRICE, AMOUNT, NAME, QUANTITY
- SALE - BARCODE, DOSE, TYPE, PRICE, AMOUNT, NAME, QUANTITY, DATE
- USER - ID, NAME, DOB, PHONE, ADDRESS, SALARY, PASSWORD
- LOGIN - NAME, TYPE, DATE, TIME, ID
- INBOX - MESSAGE-FROM, MESSAGE-TO, TEXT, SENDER_ID

RELATIONAL SCHEMA

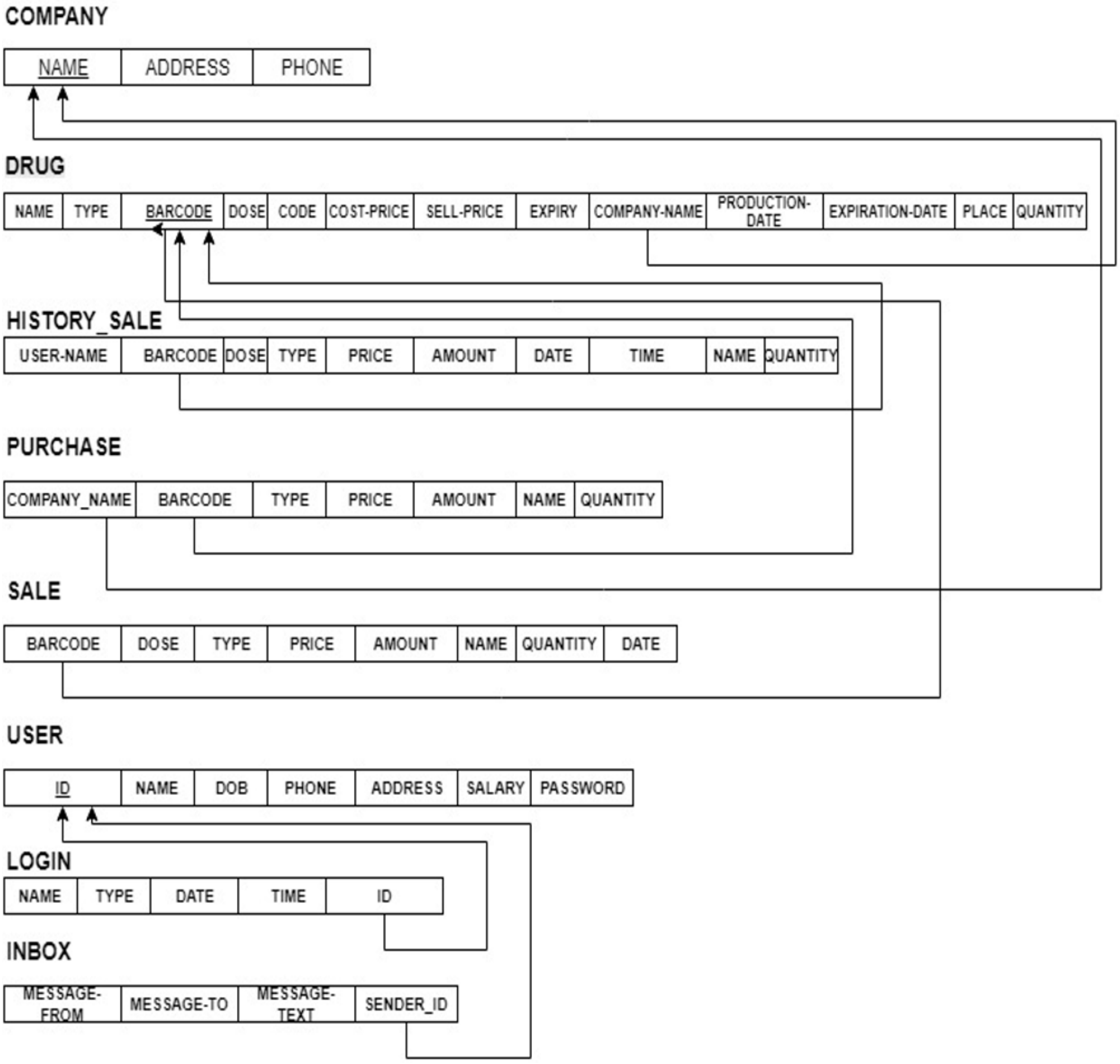

COMPANY
NAME | ADDRESS | PHONE
DRUG
NAME | TYPE | BARCODE | DOSE | CODE | COST-PRICE | SELL-PRICE | EXPIRY | COMPANY-NAME | PRODUCTION-DATE | EXPIRATION-DATE | PLACE | QUANTITY
HISTORY_SALE
USER-NAME | BARCODE | DOSE | TYPE | PRICE | AMOUNT | DATE | TIME | NAME | QUANTITY
PURCHASE
COMPANY_NAME | BARCODE | TYPE | PRICE | AMOUNT | NAME | QUANTITY
SALE
BARCODE | DOSE | TYPE | PRICE | AMOUNT | NAME | QUANTITY | DATE
USER
ID | NAME | DOB | PHONE | ADDRESS | SALARY | PASSWORD
LOGIN
NAME | TYPE | DATE | TIME | ID
INBOX
MESSAGE-FROM | MESSAGE-TO | MESSAGE-TEXT | SENDER_ID

Figure 3.2: Relation Schema for Pharmacy Database

STEP 5: **ER DIAGRAM**

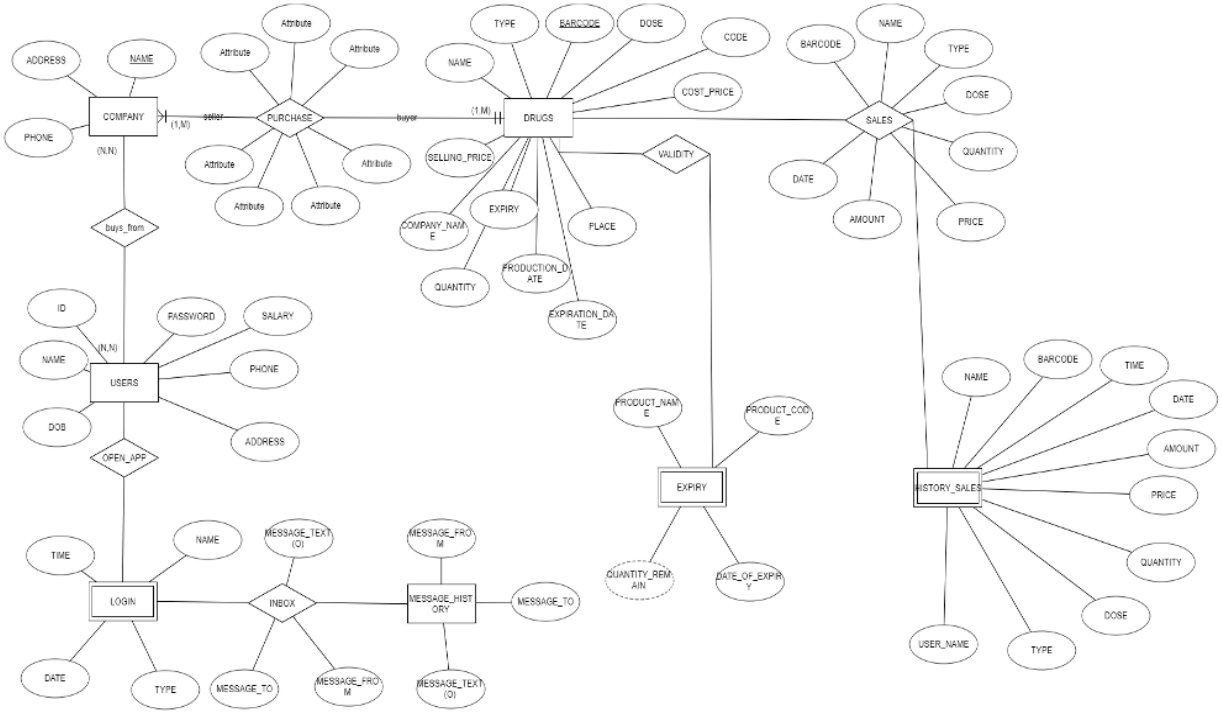


Figure 3.3: ER Diagram

3.2 NORMALISATION

The It is a process of Analyzing given Relational Schema based on their Functional Dependency and primary key to achieve desired properties of minimizing redundancy and also minimizing the insertion , deletion and update Anomalies.Storing information several times leads to wastage of storage space and increase in the total size of data stored. The type of alterations normally needed for relation is:

- Insertion of new data values to relation. This should be possible without being forced to leave blank fields for some attributes.
- Deletion of a tuple, namely, a row of a relation. This should be possible without losing vital information unknowingly.

Functional Dependency:

As the concept of dependency is very important, it is essential that it should be understood first and then proceeded to the idea of normalization. There is no fool-proof algorithmic method of identifying dependency.

Properties of normalized relations:

Ideal relations after normalization should have the following properties:

- No data values should be duplicated in different rows unnecessarily.
- A value must be specified (and required) for every attribute in a row.
- Each relation should be self-contained. In other words, if a row from a relation is deleted, important information should not be accidentally lost.
- When a row is added to a relation, other relations in the database should not be affected.
- A value of an attribute in a tuple may be changed independent of other tuples in the relation and other relations.

Consider the COMPANY table

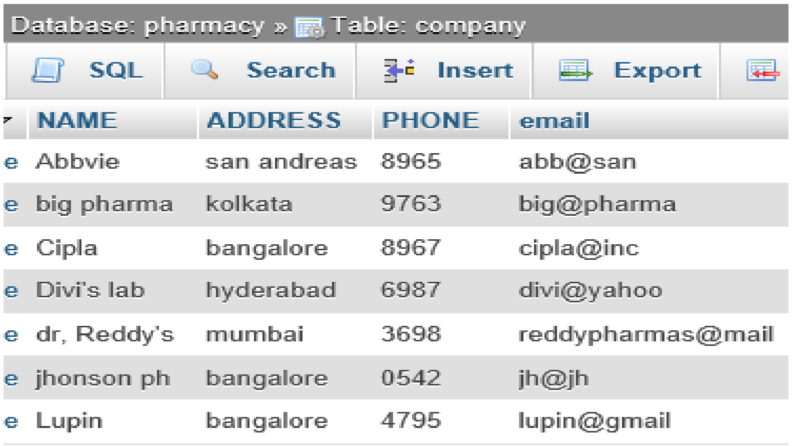
The prime attributes identified are the attributes which is part of candidate key.

The non-prime attributes are not part of primary key.

The COMPANY table has extra column as 'email' as the main details is in the 'phone' column, so the column email is been dropped.

Before Normalization-

Figure shown below is before undergoing normalization the email column is extra as the phone column is the parent and primary key for it.



	NAME	ADDRESS	PHONE	email
e	Abbvie	san andreas	8965	abb@san
e	big pharma	kolkata	9763	big@pharma
e	Cipla	bangalore	8967	cipla@inc
e	Divi's lab	hyderabad	6987	divi@yahoo
e	dr, Reddy's	mumbai	3698	reddypharmas@mail
e	jhonson ph	bangalore	0542	jh@jh
e	Lupin	bangalore	4795	lupin@gmail

Figure 3.4: Company table(Before Normailzation)

1NF

- Each table cell should contain a single value.
- Each record needs to be unique.

After normalization, the extra table is been dropped as only primar key value is stored and unique,on running the sql script

ALTER TABLE COMPNAY DROP COLUMN email.

Therefore, the given tables are in 1NF (1st Normal Form)

2NF

- Be in 1NF
- Single Column Primary Key

Each column will be having single column primary key.Therefore, the given tables are in 2NF (2nd Normal Form)

3NF

- Be in 2NF
- Rule 2- Has no transitive functional dependencies

Two tables does not have transitive functional dependencies.Therefore, the given tables are in 3NF (3rd Normal Form)

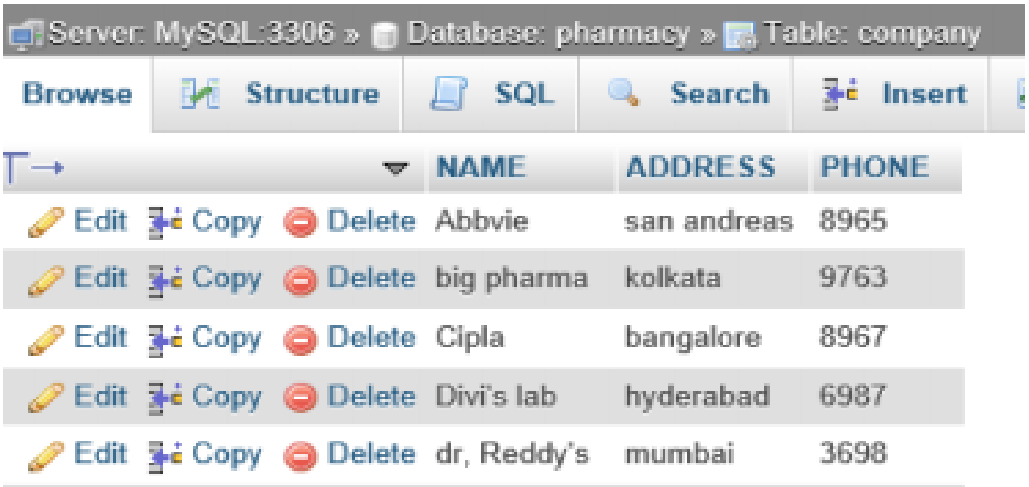
BCNF

- Be in 3NF
- Should not have more than one Candidate Key

When there are no multiple candidate keys. There are no other attributes in company table, which can be a candidate key.

Therefore, the given tables are in BCNF (BCNF Normal Form)

AFTER NORMALIZATION:



	NAME	ADDRESS	PHONE
Edit Copy Delete	Abbvie	san andreas	8965
Edit Copy Delete	big pharma	kolkata	9763
Edit Copy Delete	Cipla	bangalore	8967
Edit Copy Delete	Divi's lab	hyderabad	6987
Edit Copy Delete	dr, Reddy's	mumbai	3698

Figure 3.5: Company table(After Normailzation)

3.3 TRIGGERS

A trigger is a stored procedure in database which automatically invokes whenever a special event in the database occurs. For example, a trigger can be invoked when a row is inserted into a specified table or when certain table columns are being updated.

BEFORE and AFTER :

BEFORE triggers run the trigger action before the triggering statement is run.
AFTER triggers run the trigger action after the triggering statement is run.
I have used 3 triggers with insert, update and delete as actions.

INSERT TRIGGER

```
CREATE TRIGGER `insert_user` AFTER INSERT ON `users`
 FOR EACH ROW INSERT INTO users_log VALUES(NEW.ID, NEW.NAME,
NEW.ADDRESS, NEW.PHONE, "INSERTED", NOW())
```

when a user has been added to the databse using the software or the server,this trigger is invoked and it's stored in separate users_log table with exact date and time and this table can be accessed through the database.

UPDATE TRIGGER

```
CREATE TRIGGER `update_user` AFTER UPDATE ON `users`
 FOR EACH ROW INSERT INTO users_log VALUES(NEW.ID, NEW.NAME,
NEW.ADDRESS, NEW.PHONE, "UPDATED", NOW())
```

whenever the user Details is been updated through the software or the server,the update trigger is invoked and it's updated value is been stored in separate table namely users_log with exact date and time. and this table can be accessed through the database.

DELETE TRIGGER

```
CREATE TRIGGER `delete_user` BEFORE DELETE ON `users`
 FOR EACH ROW INSERT INTO users_log VALUES(OLD.ID, OLD.NAME,
OLD.ADDRESS, OLD.PHONE, "DELETED", NOW())
```

whenever a user is been deleted through the software or the server,the delete trigger is invoked and it's old deleted value is been stored in separate table namely users_log with exact date and time and this table can be accessed through the database.

3.4 Procedures

Stored Procedures are created to perform one or more DML operations on Database. It is nothing but the group of SQL statements that accepts some input in the form of parameters and performs some task and may or may not returns a value. The most important part is parameters. Parameters are used to pass values to the Procedure. There are 3 different types of parameters, they are as follows:

IN:

This is the Default Parameter for the procedure. It always receives the values from calling program.

OUT:

This parameter always sends the values to the calling program.

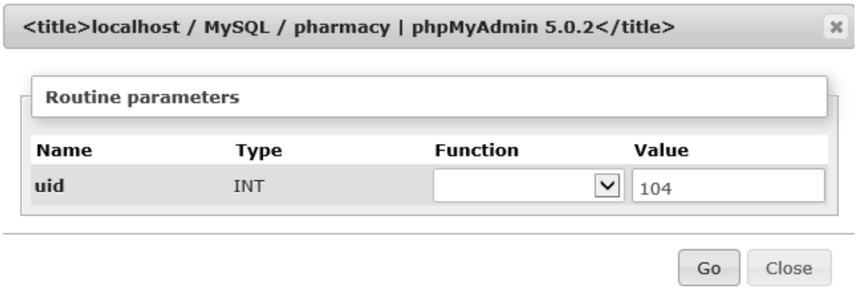
IN OUT:

This parameter performs both the operations. It Receives value from as well as sends the values to the calling program.

Here I am using both **IN** and **OUT** Parameter for our stored procedure,the result of the query is shown next and also figure while giving the input for IN parameter.

```
DELIMITER $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `getusers`(IN `uid`
INT, OUT `users_name` VARCHAR(50))
   NO SQL
SELECT NAME INTO users_name FROM users WHERE ID = uid$$
DELIMITER ;
```

USING the above sql script a stored procedure is been created in the PHARMACY database, which requires IN parameters.



<title>localhost / MySQL / pharmacy | phpMyAdmin 5.0.2</title>

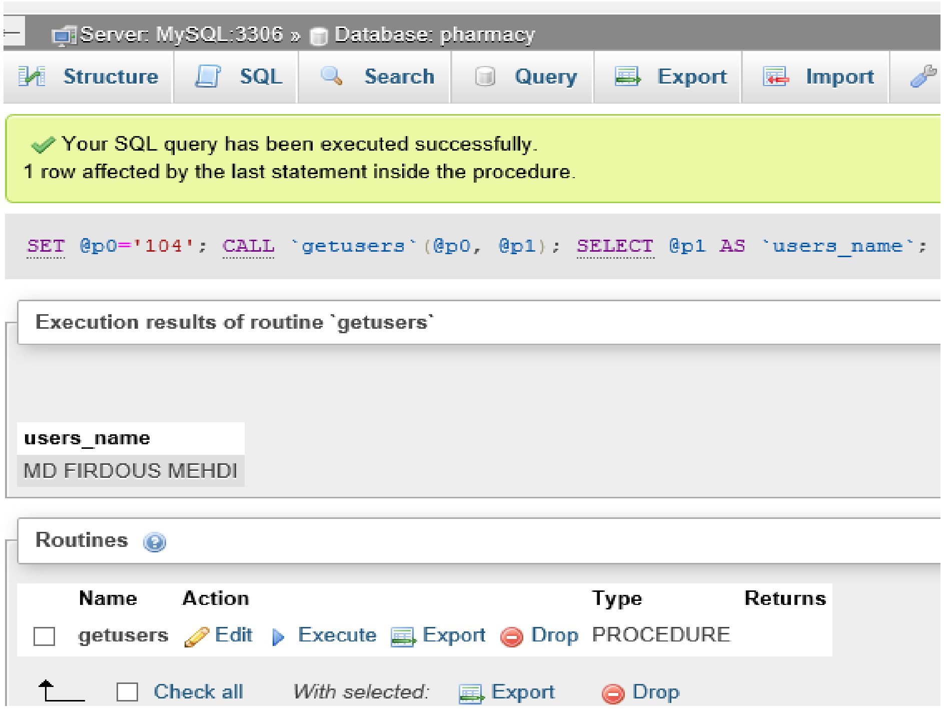
Routine parameters

Name	Type	Function	Value
uid	INT		104

Figure 3.6: input value for the Procedure

After giving the Value for the query, NAME for the respective UID is shown.

RESULT FOR STORED PROCEDURE:



✓ Your SQL query has been executed successfully.
1 row affected by the last statement inside the procedure.

```
SET @p0='104'; CALL `getusers`(@p0, @p1); SELECT @p1 AS `users_name`;
```

Execution results of routine `getusers`

users_name
MD FIRDOUS MEHDI

Routines

	Name	Action	Type	Returns
□	getusers	Edit Execute Export Drop	PROCEDURE	

Figure 3.7: Result for the Procedure

Chapter 4

IMPLEMENTATION AND CODING

4.1 SOURCE CODE:

FRONTEND

login.java

```
public class Login extends javax.swing.JFrame
{Connection con = null;
PreparedStatement pre= null;
ResultSet res= null;
staticsetrionglmaNeF;oolkit.getDefaultToolkit().getImage(Login.class.
getResource("/img/Untitled.    public Login() {
png")));
        initComponents();
        con=Connect.connect();
        new changestyle().chabgelock();
        SwingUtilities.updateComponentTreeUI(this);
        login_Hold();
    }
 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{
if(id.getText().equals("")||pass.getText().equals("")){
    JOptionPane.showMessageDialog(null,"Complete Your Login
Information","Missing Information",2);
}else {
    String sql = "select ID,NAME,PASSWORD from users where
ID='"+id.getText()+"' ";
    try{
        pre=con.prepareStatement(sql);
        res=pre.executeQuery();
        if(res.next()){
        NAME=res.getString("NAME");
 }else{
            JOptionPane.showMessageDialog(null,"Wrong Password","
Failed Access",2);          }
      }else{
      JOptionPane.showMessageDialog(null,"Wrong ID","Failed Access",
2);
      }
   }catch(Exception e){
      JOptionPane.showMessageDialog(null,e.getMessage(),"Error",2);
   }
 }
   }
   private void passKeyReleased(java.awt.event.KeyEvent evt)
{
if(pass.getForeground().equals(Color.GRAY)){
   char passs = pass.getText().charAt(0);
   pass.setText("");//
   pass.setText(String.valueOf(passs));
   pass.setForeground(Color.BLACK);
   pass.setEchoChar('*');
}
   }
    private void passMouseClicked(java.awt.event.MouseEvent evt)
 {                          if(pass.getForeground().equals(Color.GRAY)){
    pass.setText("");
    pass.setForeground(Color.BLACK);
    pass.setEchoChar('*');
 }
    }
```

pharmacy.java

```
public class Pharmacy extends javax.swing.JFrame
{    Connection con = null;
    PreparedStatement pre= null;
    ResultSet res = null;
    static int ex;
    User user ;
    Company comp;
    static Drug drug ;
    static Move_Drug move_drug;
    Buy_Drug buy;
    Date d;
    SimpleDateFormat dd;
    static String to;
    static String from ;
    static String text ;
    static String almost_expired_bar ;
    static String expired_bar ;

    public Pharmacy() {
    setIconImage(Toolkit.getDefaultToolkit().getImage(Pharmacy.class.
         getResource("/img/Untitled.png")));
        initComponents();
        con=Connect.connect();
        showDate();
        buttonvis();
        user = new User();
        drug = new Drug();
        comp = new Company();
        lert_message();
    }
```

```
 private void purchaseActionPerformed(java.awt.event.ActionEvent evt)
{
        if(!username.getText().equals("Employee")){
            enterpurchase();
        }else {
            String pass = JOptionPane.showInputDialog("You are not allowed
to check user
Inforamtion\nTo get in please confirm Admin Password");
            String sql = "select NAME,PASSWORD from users where NAME
='Ebrahem Samer' ";
                try{
                    pre=con.prepareStatement(sql);
                    res=pre.executeQuery();
                    if(res.next()){
                        if(res.getString("PASSWORD").equals(pass)){
                            enterpurchase();
                        }else if(pass.isEmpty()){
                            JOptionPane.showMessageDialog(null,"You must write
 admin Password","Failed Access",2);
                    }else {
                        JOptionPane.showMessageDialog(null,"Wrong Password","
Failed Access",2);                        }}

            private void endwithActionPerformed(java.awt.event.ActionEvent
        evt) {
                comp.setVisible(true);
                comp.save.setEnabled(false);
                comp.update.setEnabled(false);
                comp.delete.setEnabled(true);
            }
```

drug.java

```
public class Drug extends javax.swing.JFrame
{Connection con = null;
PreparedStatement pre = null;
ResultSet res= null;
    public Drug() {
        initComponents();
        con=Connect.connect();
        company();
    }
 private void deletedrugActionPerformed(java.awt.event.ActionEvent evt)
{
if(barcode.getText().equals("")){
 JOptionPane.showMessageDialog(null,"Enter Drug_Name You want to
delete","Missing Information",2);
}else {
    if(checkdrug_name()){
        String sql = "delete from drugs where BARCODE='"+barcode.getText
()+"' ";
        try{
        pre=con.prepareStatement(sql);
        pre.execute();
       }catch(Exception e){
      JOptionPane.showMessageDialog(null,e.getMessage(),"Error",2);
        }return found;
    }
    private void company(){
        String sql = "select NAME from company";
        try{
            pre=con.prepareStatement(sql);
            res=pre.executeQuery();
            while(res.next()){
                company_name.addItem(res.getString("NAME"));
            }
        }
```

4.2 DATABASE DESIGN

TABLE CREATION (BACKEND):

company

```
CREATE TABLE  `company` (
  `NAME` varchar(50) NOT NULL,
  `ADDRESS` varchar(50) NOT NULL,
  `PHONE` varchar(20) NOT NULL,
  PRIMARY KEY (`NAME`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

drugs

```
CREATE TABLE `drugs` (
  `NAME` varchar(50) NOT NULL,
  `TYPE` varchar(20) NOT NULL,
  `BARCODE` varchar(20) NOT NULL,
  `DOSE` varchar(10) NOT NULL,
  `CODE` varchar(10) NOT NULL,
  `COST_PRICE` double NOT NULL,
  `SELLING_PRICE` double NOT NULL,
  `EXPIRY` varchar(20) NOT NULL,
  `COMPANY_NAME` varchar(50) NOT NULL,
  `PRODUCTION_DATE` date NOT NULL,
  `EXPIRATION_DATE` date NOT NULL,
  `PLACE` varchar(20) NOT NULL,
  `QUANTITY` int(11) NOT NULL,
  PRIMARY KEY (`BARCODE`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

history_sales

```
CREATE TABLE `history_sales` (
  `USER_NAME` varchar(20) NOT NULL,
  `BARCODE` varchar(20) NOT NULL,
  `NAME` varchar(50) NOT NULL,
  `TYPE` varchar(10) NOT NULL,
  `DOSE` varchar(10) NOT NULL,
  `QUANTITY` int(11) NOT NULL,
  `PRICE` double NOT NULL,
  `AMOUNT` double NOT NULL,
  `DATE` varchar(15) NOT NULL,
  `TIME` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

inbox

```
CREATE TABLE 'inbox' (
  `MESSAGE_FROM` varchar(20) NOT NULL,
  `MESSAGE_TO` varchar(20) NOT NULL,
  `MESSAGE_TEXT` varchar(200) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

login

```
CREATE TABLE  `login` (
  `NAME` varchar(50) NOT NULL,
  `TYPE` varchar(20) NOT NULL,
  `DATE` varchar(20) NOT NULL,
  `TIME` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

purchase

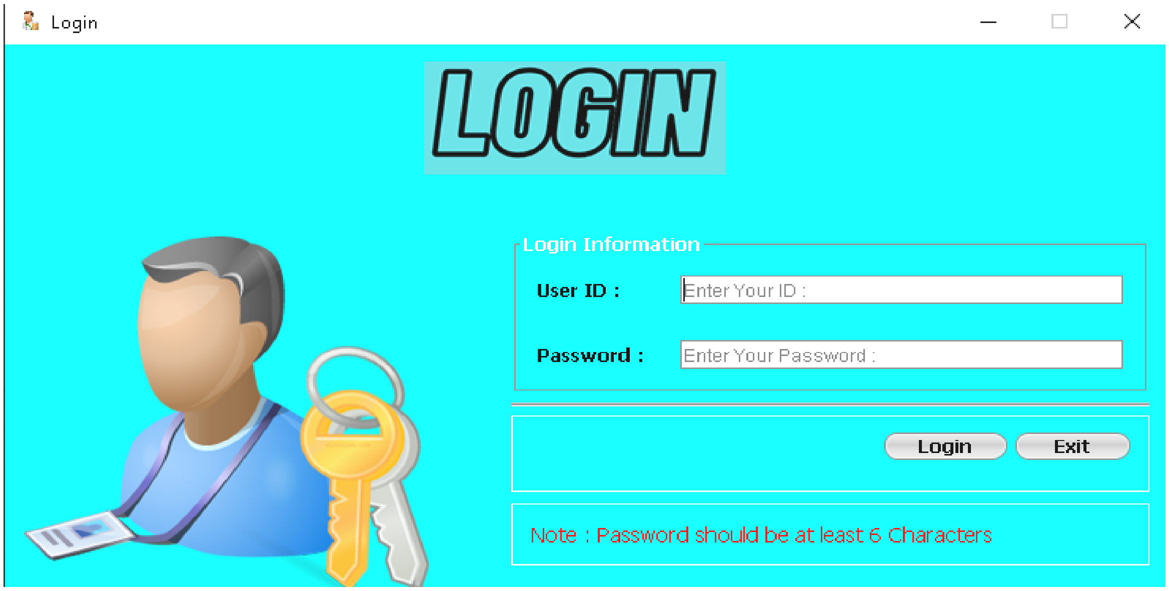
```
CREATE TABLE `purchase` (
  `BARCODE` varchar(20) NOT NULL,
  `NAME` varchar(50) NOT NULL,
  `TYPE` varchar(20) NOT NULL,
  `COMPANY_NAME` varchar(20) NOT NULL,
  `QUANTITY` int(11) NOT NULL,
  `PRICE` double NOT NULL,
  `AMOUNT` double NOT NULL,
  PRIMARY KEY (`BARCODE`),
  KEY `fkr3` (`COMPANY_NAME`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

users

```
CREATE TABLE `users` (
  `ID` int(11) NOT NULL,
  `NAME` varchar(50) NOT NULL,
  `DOB` varchar(20) NOT NULL,
  `ADDRESS` varchar(100) NOT NULL,
  `PHONE` varchar(20) NOT NULL,
  `SALARY` double NOT NULL,
  `PASSWORD` varchar(20) NOT NULL,
  PRIMARY KEY (`ID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Chapter 5

SCREENSHOTS

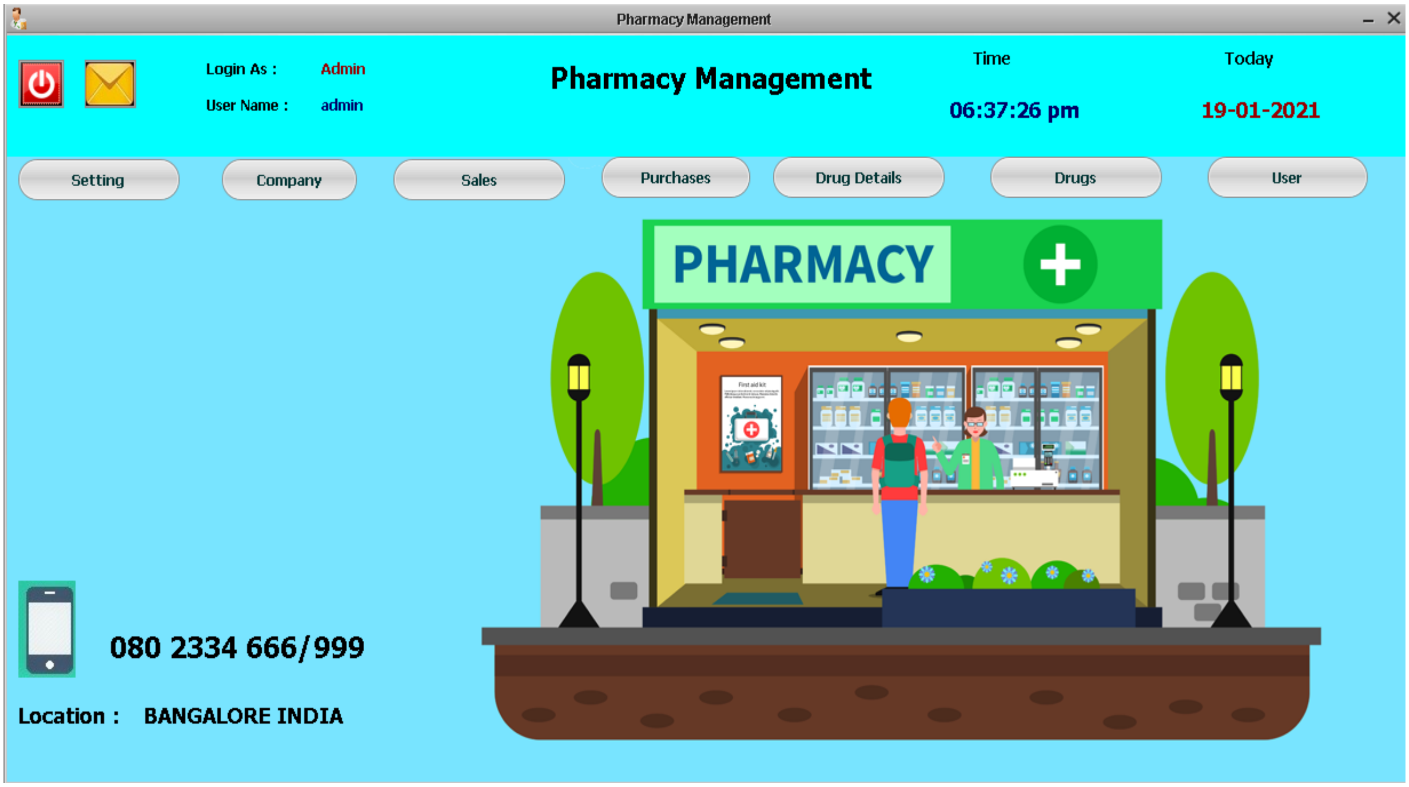


Enter Your ID :

Enter Your Password :

Note : Password should be at least 6 Characters

Figure 5.1: LOGIN PAGE



Login As : Admin

User Name : admin

Pharmacy Management

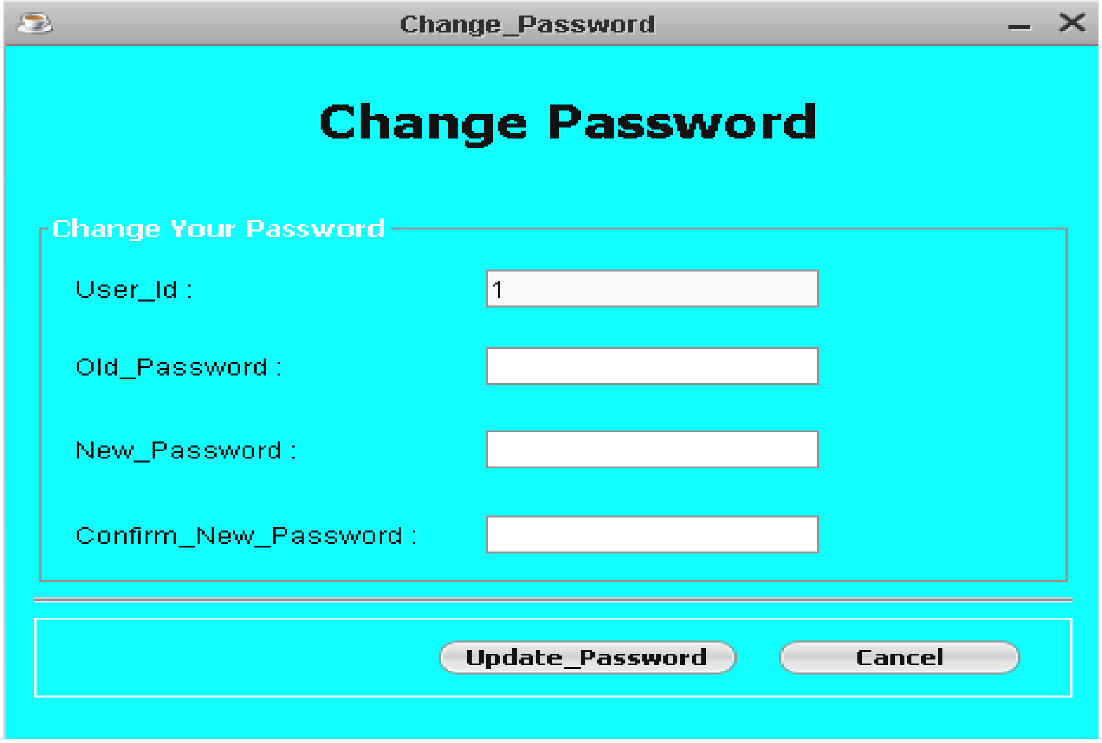
Time 06:37:26 pm

Today 19-01-2021

080 2334 666/999

Location : BANGALORE INDIA

Figure 5.2:PHARMACY MANAGEMENT AS ADMIN



Change Password

Change Your Password

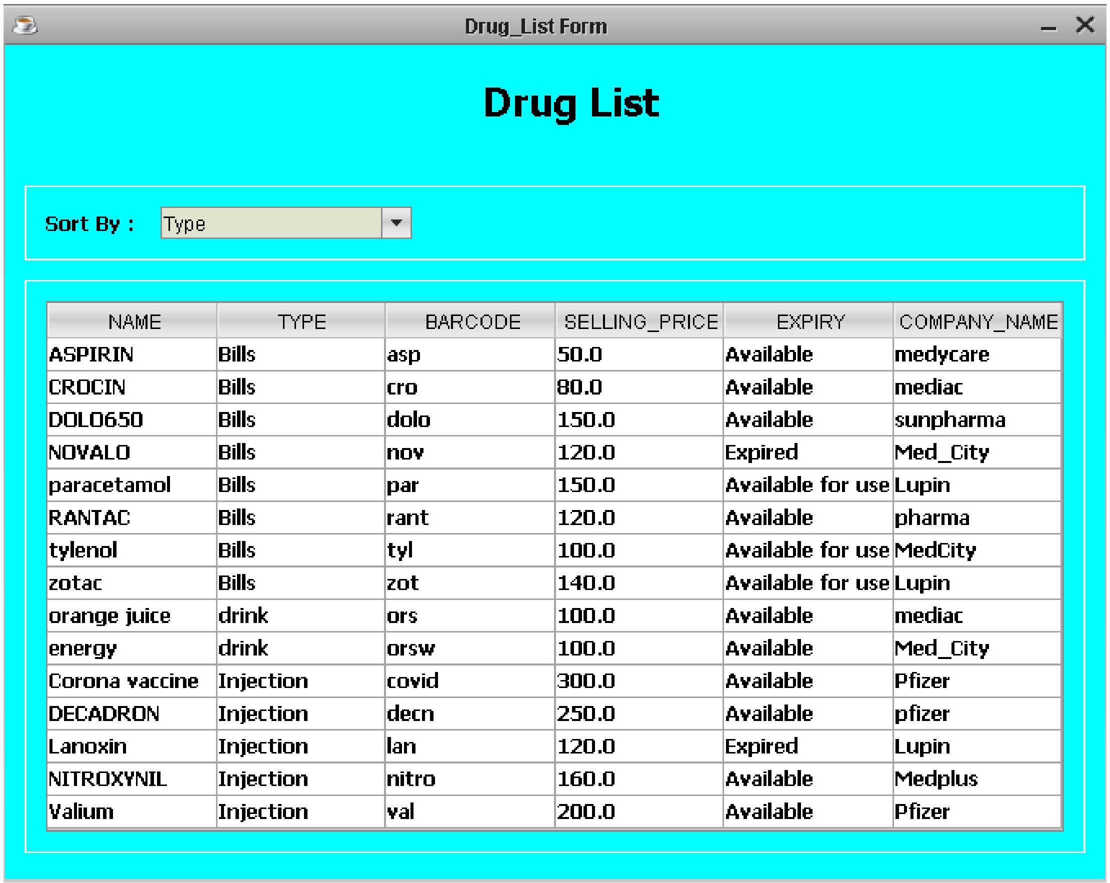
User_Id : 1

Old_Password :

New_Password :

Confirm_New_Password :

Figure 5.3: CHANGE PASSWORD

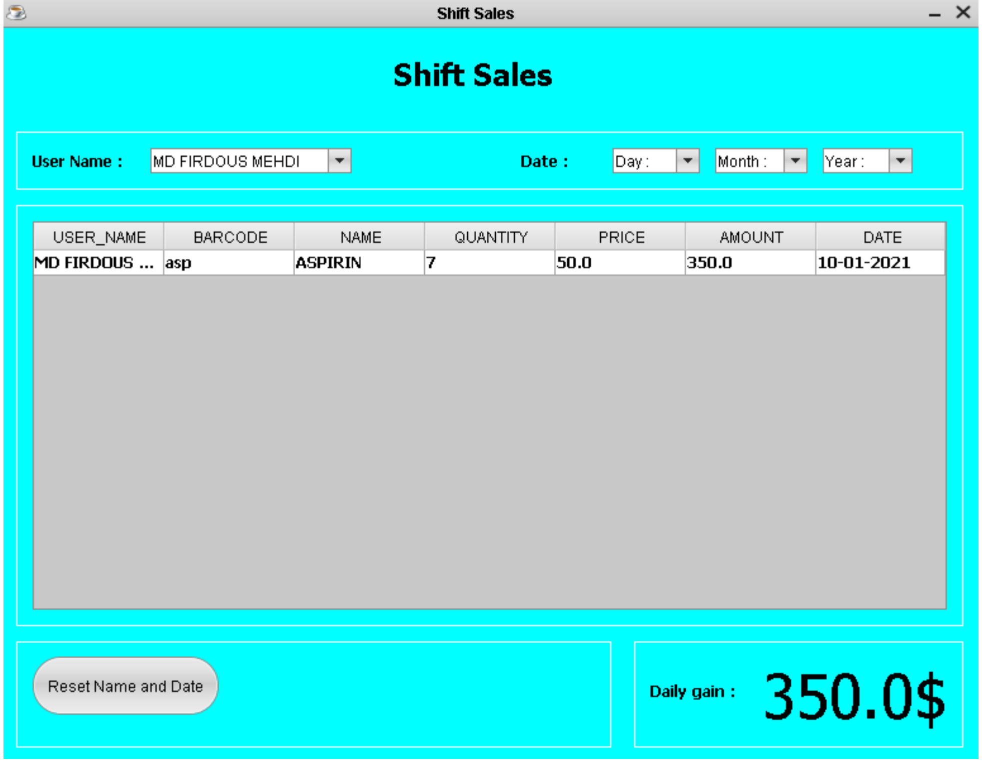


Drug List

Sort By : Type

NAME	TYPE	BARCODE	SELLING_PRICE	EXPIRY	COMPANY_NAME
ASPIRIN	Bills	asp	50.0	Available	medycare
CROCIN	Bills	cro	80.0	Available	mediac
DOLO650	Bills	dolo	150.0	Available	sunpharma
NOVALO	Bills	nov	120.0	Expired	Med_City
paracetamol	Bills	par	150.0	Available for use	Lupin
RANTAC	Bills	rant	120.0	Available	pharma
tylenol	Bills	tyl	100.0	Available for use	MedCity
zotac	Bills	zot	140.0	Available for use	Lupin
orange juice	drink	ors	100.0	Available	mediac
energy	drink	orsw	100.0	Available	Med_City
Corona vaccine	Injection	covid	300.0	Available	Pfizer
DECADRON	Injection	decn	250.0	Available	pfizer
Lanoxin	Injection	lan	120.0	Expired	Lupin
NITROXYNIL	Injection	nitro	160.0	Available	Medplus
Valium	Injection	val	200.0	Available	Pfizer

Figure 5.4: DRUG LIST (search)



Shift Sales

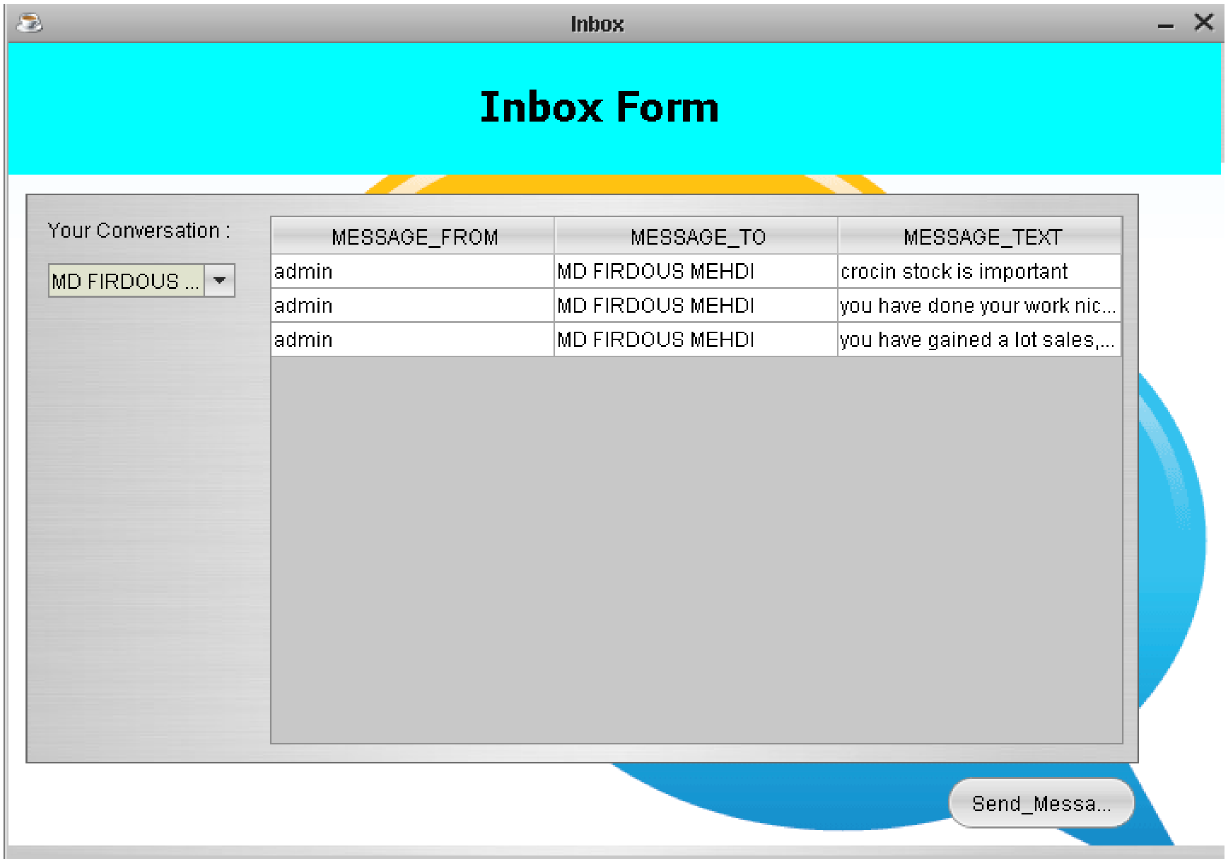
User Name : MD FIRDOUS MEHDI

Date : Day : Month : Year :

USER_NAME	BARCODE	NAME	QUANTITY	PRICE	AMOUNT	DATE
MD FIRDOUS ...	asp	ASPIRIN	7	50.0	350.0	10-01-2021

Daily gain : 350.0$

Figure 5.5: SALES PAGE

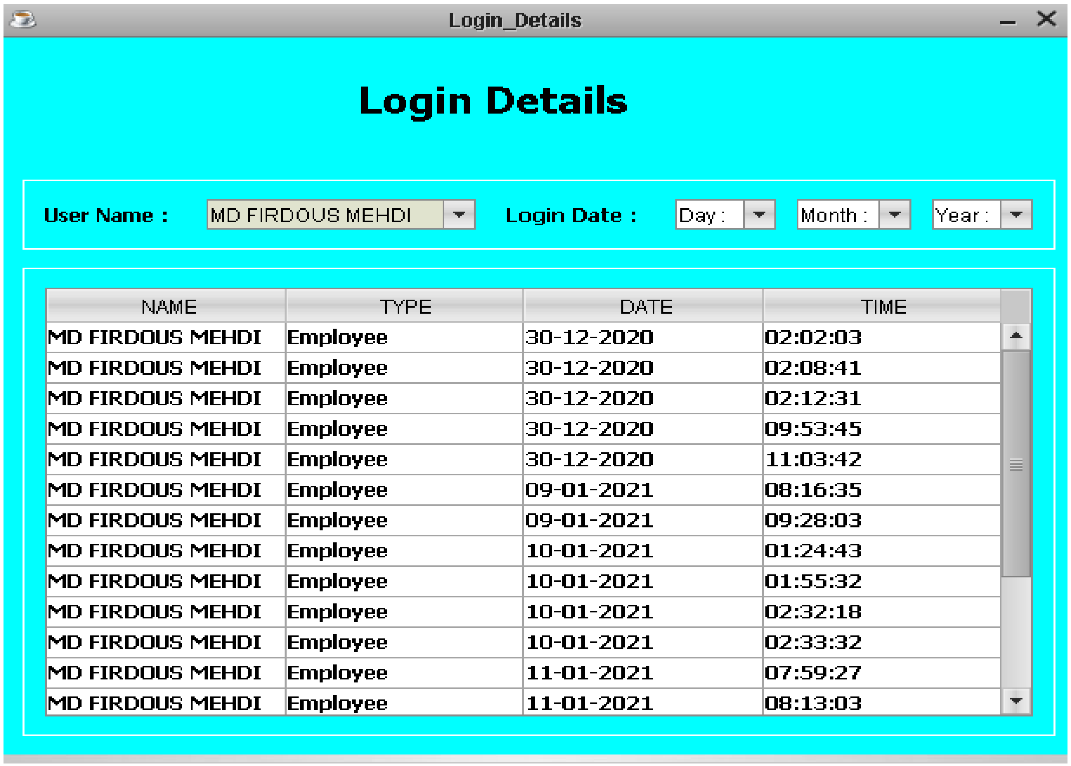


Inbox Form

Your Conversation : MD FIRDOUS ...

MESSAGE_FROM	MESSAGE_TO	MESSAGE_TEXT
admin	MD FIRDOUS MEHDI	crocin stock is important
admin	MD FIRDOUS MEHDI	you have done your work nic...
admin	MD FIRDOUS MEHDI	you have gained a lot sales,...

Figure 5.6: INBOX (messaging)



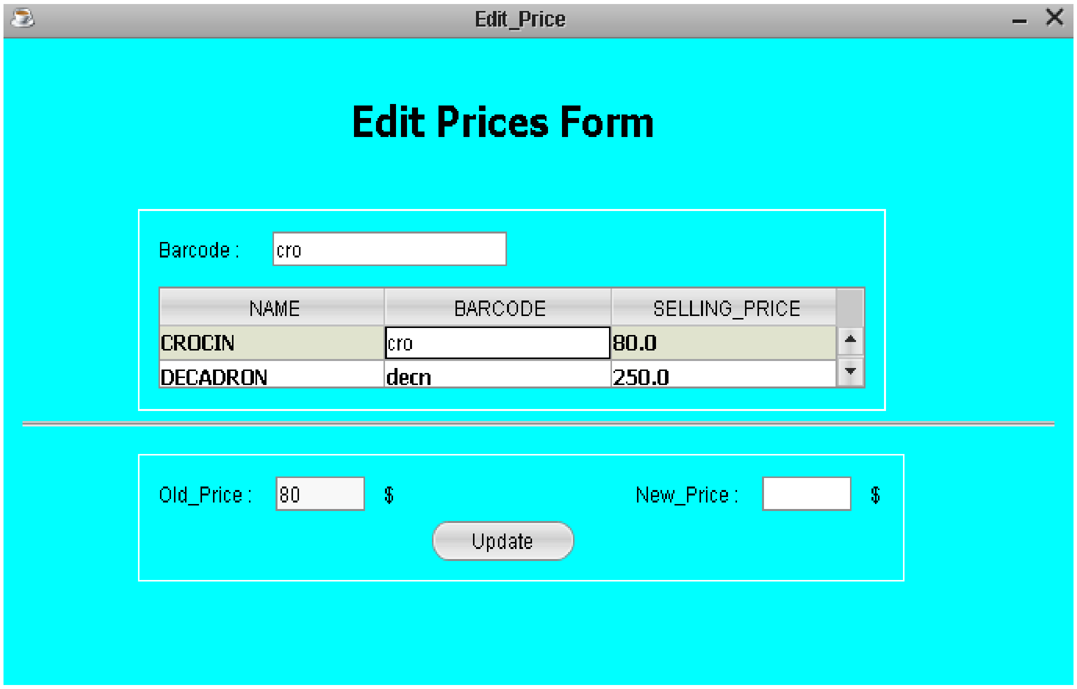
Login Details

User Name : MD FIRDOUS MEHDI

Login Date : Day : Month : Year :

NAME	TYPE	DATE	TIME
MD FIRDOUS MEHDI	Employee	30-12-2020	02:02:03
MD FIRDOUS MEHDI	Employee	30-12-2020	02:08:41
MD FIRDOUS MEHDI	Employee	30-12-2020	02:12:31
MD FIRDOUS MEHDI	Employee	30-12-2020	09:53:45
MD FIRDOUS MEHDI	Employee	30-12-2020	11:03:42
MD FIRDOUS MEHDI	Employee	09-01-2021	08:16:35
MD FIRDOUS MEHDI	Employee	09-01-2021	09:28:03
MD FIRDOUS MEHDI	Employee	10-01-2021	01:24:43
MD FIRDOUS MEHDI	Employee	10-01-2021	01:55:32
MD FIRDOUS MEHDI	Employee	10-01-2021	02:32:18
MD FIRDOUS MEHDI	Employee	10-01-2021	02:33:32
MD FIRDOUS MEHDI	Employee	11-01-2021	07:59:27
MD FIRDOUS MEHDI	Employee	11-01-2021	08:13:03

Figure 5.7: LOGIN DETAILS



Edit Prices Form

Barcode : cro

NAME	BARCODE	SELLING_PRICE
CROCIN	cro	80.0
DECADRON	decn	250.0

Old_Price : 80 $

New_Price : $

Figure 5.8: EDIT PRICES

Chapter 6

CONCLUSION:

Our project is only a humble venture to satisfy the needs to manage their project work. Several user friendly coding have also adopted. This package shall prove to be a powerful package in satisfying all the requirements of the school. The objective of software planning is to provide a frame work that enables the manger to make reasonable estimates made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

At the end it is concluded that we have made effort on following point

- A description of the background and context of the project and its relation to work already done in the area.
- Made statement of the aims and objectives of the project.
- The description of Purpose, Scope, and applicability.
- We define the problem on which we are working in the project.
- We describe the requirement Specifications of the system and the actions that can be done on these things.

- We understand the problem domain and produce a model of the system, which describes operations that can be performed on the system.
- We included features and operations in detail, including screen.
- We designed user interface and security issues related to system.
- Finally the system is implemented and tested according to test cases.

Chapter 7

FUTURE SCOPE OF THE PROJECT:

In a nutshell, it can be summarized that the future scope of the project circles around maintaining information regarding:

- We can add printer in future.
- We can give more advance software for Pharmacy Management System including more facilities.
- We will host the platform on online servers to make it accessible worldwide
- Integrate multiple load balancers to distribute the loads of the system
- Create the master and slave database structure to reduce the overload of the database queries.
- Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers.

The above mentioned points are the enhancements which can be done to increase the applicability and usage of this project. Here we can maintain the records of Pharmacy and Medicines. Also, as it can be seen that now-a-days the players are versatile, i.e. so there is a scope for introducing a method to maintain the Pharmacy Management System. Enhancements can be done to maintain all the Pharmacy, Medicines, Stocks, Company, Inventory.

We have left all the options open so that if there is any other future requirement in the system by the user for the enhancement of the system then it is possible to implement them.In the last we would like to thanks all the persons involved in the development of the system directly or indirectly.We hope that the project will serve its purpose for which it is develop there by underlining success of process.

Chapter 8

BIBLIOGRAPHY AND REFERENCES

- Google for problem solving
- Java and Software Design Concepts by Apress
- Database Programming with JDBC and Java by O'Reilly
- Head First Java 2nd Edition
- http://www.javaworld.com/javaworld/jw-01-1998/jw-01-Credentialreview.html
- http://www.jdbc-tutorial.com/
- https://www.tutorialspoint.com/java/
- http://www.javatpoint.com/java-tutorial
- https://docs.oracle.com/javase/tutorial/
- http://www.wampserver.com/en/
- http://www.JSP.net/
- http://www.tutorialspoint.com/mysql/
- http://apache.org/docs/2.0/misc/tutorials.html
- https://www.wampserver.com/en/
- https://dev.mysql.com/downloads/connector/