VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belgaum-590018, Karnataka State, India

Mini Project Report
on

“Pharmacy Management System”

Submitted in Partial fulfillment of the Requirements for the award of Degree

BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE & ENGINEERING

Submitted by

NAME : Preetham A
USN : 1CD15CS114

Under the Guidance of

Ms. Bhavana P
Assistant Professor,
Department of CSE,CITech

Ms. Priyadarshini M
Assistant Professor,
Department of CSE,CITech

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CAMBRIDGE INSTITUTE OF TECHNOLOGY
K R Puram, Bangalore-560036, Karnataka, India

CAMBRIDGE INSTITUTE OF TECHNOLOGY

CERTIFICATE

This is to certify that the **Database Management System** mini project entitled "**Pharmacy Management System"** has been carried out by **Preetham A** (1CD15CS114) is bonafide student of **Cambridge Institute of Technology** in partial fulfillment for **Fifth Semester** of **Bachelor of Engineering** in **Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year **2019-2020**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report. The Database Project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said Semester.

Internal Guides:

1) Ms. Bhavana P
Dept. of CSE, CITech

2) Ms. Priyadarshini M
Dept. of CSE, CITech

HEAD OF DEPARTMENT

Dr. Shashikumar D R
Dept. of CSE, CITech

Examiners:

1)

2)

ACKNOWLEDGEMENT

I take great pleasure in expressing my sincere gratitude to **Dr. Suresh L**, Principal of **Cambridge Institute of Technology** for giving me an opportunity to computerize this project.

I wish to express my gratitude to **Dr. Shashikumar D R,** Professor and Head of Department of Computer Science and Engineering for providing such a healthy environment for the successful completion of Project work.

My sincere thanks to **Ms. Bhavana P**, Assistant Professor, Department of Computer Science and Engineering and **Ms. Priyadarshini M,** Assistant Professor**,** Department of Computer Science and Engineering for their expert guidance, initiative and providing a good working environment and for their constant support and encouragement throughout the project.

I would also like to thank all other teaching and technical staffs of Department of **Computer Science and Engineering**, who have directly or indirectly helped me in the completion of this Project Work.

And lastly I would hereby acknowledge and thank my parents who have been a source of inspiration and also instrumental in the successful completion of this project.

Preetham A (1CD15CS114)

ABSTRACT

The project titled as “Pharmacy Management System” is developed in java swings as front end oracle 11g as our database backend .the main objective of the project is to control the total pharmacy administration system online so that we maintain the records easily and also the entire activity can be automated by day to day process activities. In this project the different modules are admin login, doctor, patient, medicine, pharmacy and prescribes the administrator has to login to the project to view and manage the details of the pharmacy. By using this software module we can maintain stock and keep track of the available medicine

Contents

Chapter 1

INTRODUCTION

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 reduced the hardships faced by this existing system. Moreover, this system is designed for particular need of the pharmacy to carry out operations in the smooth and effective manner.

This pharmacy is still doing their whole work manually. Therefore the pharmacy needs many workers to monitor all the process and to check the presence of each drug in Pharmacy. So, when the batches of the drug arrive in the pharmacy the manual entry is done in the register. When the month is completed the workers in the pharmacy have to generate the list or report manually of the drugs in the pharmacy. This work is done to maintain required stock in the pharmacy. This kind of work may lead to mistake by workers and lead to a major problem.

Therefore, to solve this kind of problem the urgent need is to develop a Pharmacy Management System that will prove the beneficial for the pharmacy. By using this software module, we can maintain stock and generate sales report and keep track of availability of medicines.

There are two main requirement of the software that is it must manage the stock of the pharmacy and another is that it must produce the report for sales and stock.

The application is reduced as much as possible to avoid errors while entering data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use the system. Thus, by this all it proves it is user friendly. 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 than to concentrate on record keeping. Thus, it will help organization in better utilization and resources

There are two main requirement of the software that is it must manage the stock of the pharmacy and another is that it must produce the report for sales and stock.

The application is reduced as much as possible to avoid errors while entering data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use the system. Thus, by this all it proves it is user friendly. 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 than to concentrate on record keeping. Thus, it will help organization in better utilization and resources.

Every organization whether big or small has challenges to overcome and maintaining the information of Medicine, Pharmacy, Company and Sales. Every Pharmacy Management System has different pharmacy needs therefore we decide exclusive employee management systems that are adapted to your managerial requirement. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level information and detail for the future goals.

By using this management system we can check record instantly which is not possible by manual methods. As the system gives information of the expired medicine we can discard them and replace them with new stocks. Thus, we conclude that Pharmacy Management System is helpful for handling the task efficiently.

CHAPTER 2

HARDWARE AND SOFTWARE REQUIREMENTS

2.1 Software Requirement Specifications

Operating System	:	Windows 8
Front End	:	Swings
Back End	:	Oracle 11g
Documentation	:	Microsoft Office 2007

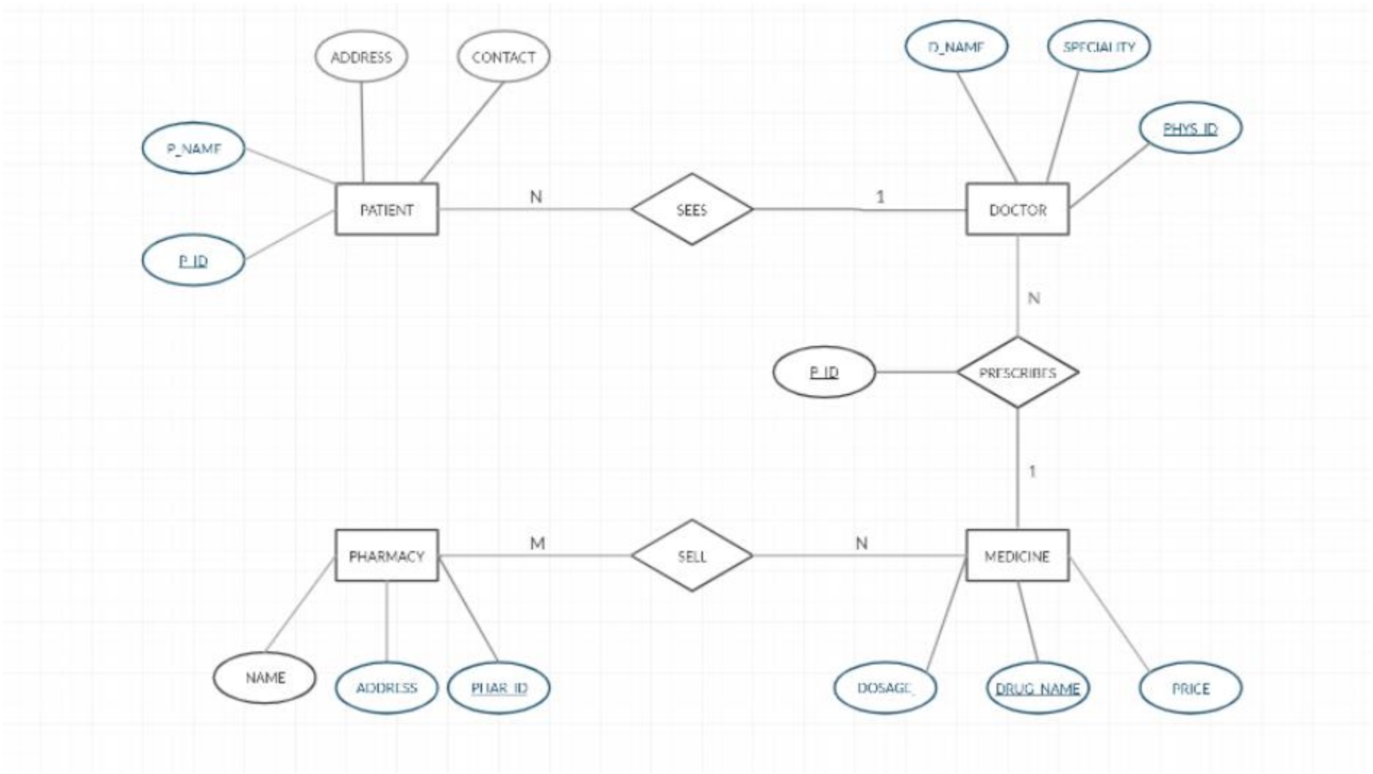
2.2 Hardware Requirement Specifications

Computer Processor	:	Core i3
Processor Speed	:	2.3 GHz Processor
Hard Disk	:	400 GB or more
RAM	:	Min 2GB

CHAPTER 3

ENTITY RELATIONSHIP DIAGRAM

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. In other words, ER diagrams illustrate the logical structure of databases.

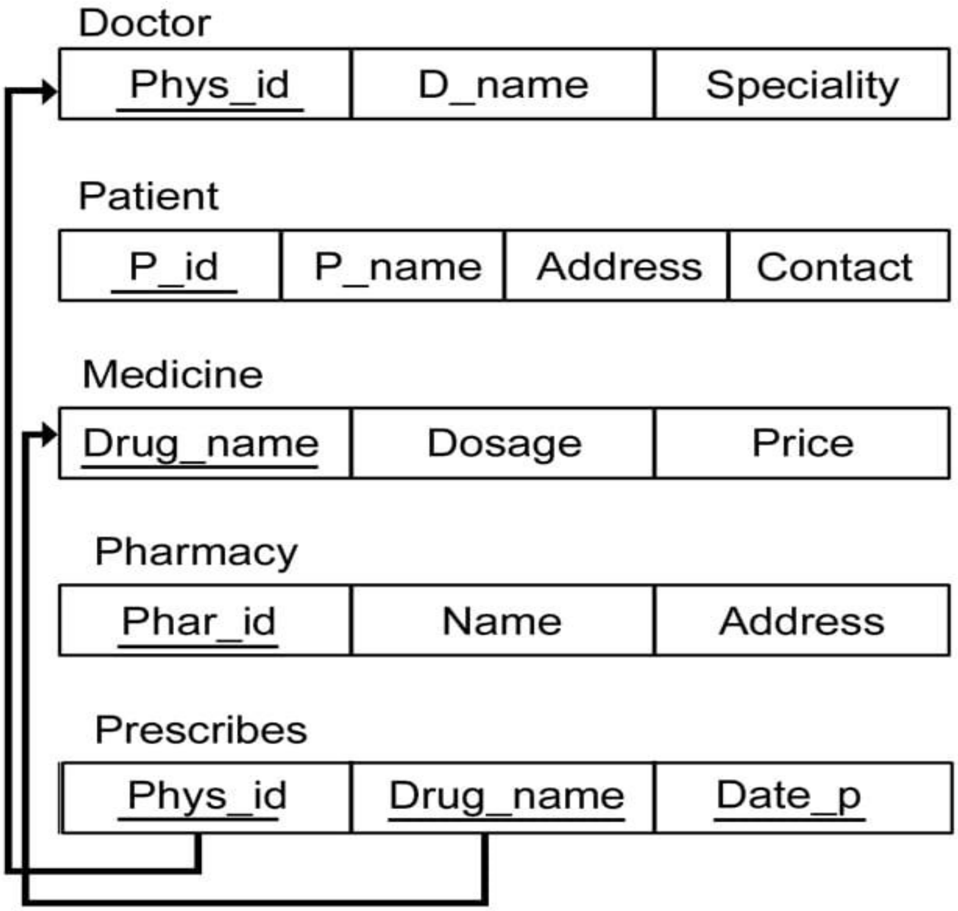


CHAPTER 4

SCHEMA DIAGRAM

A database schema is the skeleton structure that represents the logical view of the entire database. It defines how the data is organized and how the relations among them are associated. It formulates all the constraints that are to be applied on the data.

A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams.



CHAPTER 5

IMPLEMENTATION

5.1 Backend Implementation

5.1.1 Create Statement

The CREATE TABLE statement is used to create a new table in a database.

SYNTAX

```
CREATE TABLE table_name (
    column1 datatype,
    column2 datatype,
    column3 datatype,

   ....
  );
```

The column parameters specify the names of the columns of the table.

The data type parameter specifies the type of data the column can hold (e.g. varchar, integer, date, etc.).

DOCTOR .

```
create table doctor
(phys_id number(10) primary key,
d_name varchar2(20),
speciality varchar(20));
```

MEDICINE

```
create table drug(
drug_name varchar2(20) primary key,
dosage varchar2(20),
price number(10));
```

PATIENT

```
create table patient(
p_id number(10) primary key,
p_name varchar2(20),
address varchar2(20),
contact number(10));
```

PHARMACY

```
create table pharmacy(
phar_id number(10) primary key,
name varchar2(20),
address varchar2(20));
```

PRESCRIBES

```
create table prescribes(
phys_id number(10) ,
drug_name varchar2(20),
date_p date,
primary key(phys_id,drug_name,date_p),
foreign key(phys_id) references doctor(phys_id) on delete cascade,
foreign key(drug_name) references drug(drug_name) on delete cascade);
```

5.1.2 Describe Statement

The desc statement is used to give detailed description about the tables created.

```
SQL> desc doctor;
Name                                      Null?                    Type
----------------------------------------- ------------------------ ----------------------
PHYS_ID                                   NOT NULL                 NUMBER(10)
D_NAME                                                             VARCHAR2(20)
SPECIALITY                                                         VARCHAR2(20)
```

```
SQL> desc drug;
Name                                      Null?                    Type
----------------------------------------- ------------------------ ----------------------
DRUG_NAME                                 NOT NULL                 VARCHAR2(20)
DOSAGE                                                             VARCHAR2(20)
PRICE                                                              NUMBER(10)
```

```
SQL> desc prescribes;
Name                                      Null?                    Type
----------------------------------------- ------------------------ ----------------------
 PHYS_ID                                  NOT NULL                 NUMBER(10)
DRUG_NAME                                 NOT NULL                 VARCHAR2(20)
DATE_P                                    NOT NULL                 DATE
```

```
SQL> desc patient;
Name                                      Null?                       Type
----------------------------------------- --------------------------- -----------------------
P_ID                                      NOT NULL                    NUMBER(10)
P_NAME                                                                VARCHAR2(20)
ADDRESS                                                               VARCHAR2(20)
CONTACT                                                               NUMBER(10)
```

```
SQL> desc pharmacy;
Name                                      Null?                       Type
----------------------------------------- --------------------------- -----------------------
 PHAR_ID                                  NOT NULL                    NUMBER(10)
 NAME                                                                 VARCHAR2(20)
ADDRESS                                                               VARCHAR2(20)
```

5.1.3 Insert Statement:

The INSERT statement adds one or more new rows of data to a database table.

SYNTAX:

```
Insert into tablename values (value1,value2....);
```

DOCTOR

```
SQL> insert into doctor values(101,'swaroop',’ent’);

SQL> insert into doctor values(102,'danial',’dental’);

SQL> insert into doctor values(103,'kumar',’cardio’):
```

DRUG

```
SQL> insert into drug values(crocin,100ml,30);

SQL> insert into drug values(fedex,20ml,10);
```

PRESCRIBES

```
SQL> insert into prescribes values(101,'crocin',’01-oct-10’);

SQL> insert into prescribes values(102,'fedex',’14-feb-16’);
```

PHARMACY

```
SQL> insert into pharmacy values(1111,'omkar',’kolar’);

SQL> insert into pharmacy values(1112,'cvs',’manglore’);
```

PATIENT

```
SQL> insert into patient values(1,'kedar',’banglore’,9787545621);

SQL> insert into patient values(2,'narayan',’kgf’,7209631574);
```

5.1.4 Triggers

Triggers are stored programs, which are automatically excecuted or fired when some events occur. Triggers are, in fact, written to be executed in response to any of the following events-

- a database manipulation (DML) statement (DELETE, INSERT, OR UPDATE)
- A database definition (DDL) statement (CREATE, ALTER or DROP)
- A database operation (SERVER ERROR,LOGON, LOGOFF, STARTUP, or SHUTDOWN)

```
CREATE OR REPLACE TRIGGER DRUG_PRICE
Before insert or update on drug
For each row
when(new.price>70)
begin
:new.price:=50;
end;
/
```

```
Trigger created.
```

}

5.2 Front End Implementation

5.2.1 Login page

This is the page enables the existing customer to login their account using their registered username and password.

```
String url="jdbc:oracle:thin:@localhost:1521:XE";
        String user="preetham";
        String pwd1="Preetham";

        String us=new String();
        String pw=new String();

        String u=new String("PLEASE ENTER CORRECT DETAILS");
        String p=new String("LOGIN SUCCESSFUL");

        us=usr.getText();
        pw=pwd.getText();

        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
            Connection con=DriverManager.getConnection(url, user, pwd1);
            Statement st=con.createStatement();
            String sql="select user_name,password from register where user_name='"+us+"' and
password='"+pw+"'";
            ResultSet rs=st.executeQuery(sql);
            if(rs.next())
            {
                jLabel1.setText(p);
                new Menud().setVisible(true);
                setVisible(false);
            }
            else
      jLabel1.setText(u);
        }
        catch(ClassNotFoundException | SQLException e)
        {
            System.out.println(e);

         }
        }
```

5.2.2 Registration page

This is the page where a new customer can register their account and can utilize the service.

```
String url="jdbc:oracle:thin:@localhost:1521:XE";
String user="preetham";
String pwd="Preetham";

String us=new String();
String pass=new String();
String cp=new String();

us=usr.getText();
pass=pwd1.getText();
cp=cpwd.getText();

try
{
   Class.forName("oracle.jdbc.driver.OracleDriver");
   Connection con=DriverManager.getConnection(url, user, pwd);
   Statement st=con.createStatement();
   String s="insert into register values('"+us+"','"+pass+"','"+cp+"')";
   //JOptionPane.showMessageDialog(null, "Register Successful");
   ResultSet rs=st.executeQuery(s);
   if(rs.next())
      {
         JOptionPane.showMessageDialog(null, "Register Successful");
      }
}
catch(Exception e)
{
   System.out.println(e);
}
```

5.2.3 Insert

```
 String url="jdbc:oracle:thin:@localhost:1521:XE";
String user="preetham";
String pwd="Preetham";

String physid=new String();
String dname=new String();
String spl=new String();

physid=Phys.getText();
dname=doc.getText();
spl=spec.getText();
try
{
   Class.forName("oracle.jdbc.driver.OracleDriver");
   Connection con=DriverManager.getConnection(url, user, pwd);
   Statement st=con.createStatement();
   String sel ="select * from doctor where phys_id='"+physid+"'";
   ResultSet rs=st.executeQuery(sel);
   if(rs.next())
   {
      JOptionPane.showMessageDialog(null, "Record is present already");
   }
   else
   {
      Statement st1=con.createStatement();
      String s="insert into doctor values('"+physid+"','"+dname+"','"+spl+"')";
      JOptionPane.showMessageDialog(null, "Insert Successful");
      ResultSet rs1=st1.executeQuery(s);
   }
}
catch(Exception e)
{
   System.out.println(e);
}






 }
```

5.2.4 Search

```
String url="jdbc:oracle:thin:@localhost:1521:XE";
String user="preetham";
String pwd="Preetham";

String id=new String();
id=phys.getText();

try
{
    Class.forName("oracle.jdbc.driver.OracleDriver");
    Connection con=DriverManager.getConnection(url, user, pwd);
    Statement st=con.createStatement();
    Statement st1=con.createStatement();
    String s="select * from doctor where phys_id='"+id+"'";
    ResultSet rs=st.executeQuery(s);
    ResultSet rs1=st1.executeQuery(s);
    if (rs.next())
    {
        JOptionPane.showMessageDialog(null, "Search Successful");
        jTable1.setModel(DbUtils.resultSetToTableModel(rs1));
    }
    else
    {
        JOptionPane.showMessageDialog(null, "Physician does not exist");
    }
}
catch(ClassNotFoundException | SQLException e)
{
    System.out.println(e);
}
```

5.2.5 Update

```
String url="jdbc:oracle:thin:@localhost:1521:XE";
String user="preetham";
String pwd="Preetham";

String phy=new String();
String spl=new String();
phy=phys.getText();
spl=spll.getText();

try
{
   Class.forName("oracle.jdbc.driver.OracleDriver");
   Connection con=DriverManager.getConnection(url, user, pwd);
   Statement st=con.createStatement();
   Statement dst=con.createStatement();
   String sel ="select * from doctor where phys_id='"+phy+"'";
   String upd="update doctor set speciality='"+spl+"' where phys_id='"+phy+"'";
   ResultSet rs=st.executeQuery(sel);
   if(rs.next())
   {
      ResultSet rs1=dst.executeQuery(upd);
      JOptionPane.showMessageDialog(null, "Update Successful");
   }
   else
   {
      JOptionPane.showMessageDialog(null, "Physician does not exist");
   }
}
catch(ClassNotFoundException | SQLException e)
{
   System.out.println(e);
}









}
```

5.2.6 Delete

```
String url="jdbc:oracle:thin:@localhost:1521:XE";
String user="preetham";
String pwd="Preetham";

String did=new String();

did=phys.getText();

try
{
   Class.forName("oracle.jdbc.driver.OracleDriver");
   Connection con=DriverManager.getConnection(url, user, pwd);
   Statement st=con.createStatement();
   Statement dst=con.createStatement();
   String sel ="select * from doctor where phys_id='"+did+"'";
   String del="delete from doctor where phys_id='"+did+"'";
   ResultSet rs=st.executeQuery(sel);
   if(rs.next())
   {
      ResultSet rs1=dst.executeQuery(del);
      JOptionPane.showMessageDialog(null, "Delete Successful");
   }
   else
   {
      JOptionPane.showMessageDialog(null, "Physician does not exist");
   }
}
catch(ClassNotFoundException | SQLException e)
{
   System.out.println(e);
}
```

5.2.6 login code

```
String url="jdbc:oracle:thin:@localhost:1521:XE";
 String user="preetham";
 String pwd1="Preetham";

 String us=new String();
 String pw=new String();

 String u=new String("PLEASE ENTER CORRECT DETAILS");
 String p=new String("LOGIN SUCCESSFUL");

 us=usr.getText();
 pw=pwd.getText();

 try
 {
     Class.forName("oracle.jdbc.driver.OracleDriver");
     Connection con=DriverManager.getConnection(url, user, pwd1);
     Statement st=con.createStatement();
     String sql="select user_name,password from register where user_name='"+us+"' and
password='"+pw+"'";
     ResultSet rs=st.executeQuery(sql);
     if(rs.next())
     {
         jLabel1.setText(p);
         new Menud().setVisible(true);


        setVisible(false);
    }
    else
     jLabel1.setText(u);
}
catch(ClassNotFoundException | SQLException e)
{
   System.out.println(e);
}
```

5.2.6 Connection code

Connection code

```
package miniproject;
import static java.lang.Class.forName;
import java.sql.Connection;
import java.sql.DriverManager;


import javax.swing.JOptionPane;
public class Conn
{
   public static Connection connectdb() {
    try{
        Class.forName("oracle.jdbc.OracleDriver");
        Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","preetham","Preetham");
        return con;
    }
    catch(Exception e){
        JOptionPane.showMessageDialog(null, e);
    }
    return null;
}
  public static void main(String args[]){
      new Login().setVisible(true);
  }


}
```

5.2. Registration code

```
String url="jdbc:oracle:thin:@localhost:1521:XE";
String user="preetham";
String pwd="Preetham";
String us=new String();
String pass=new String();
String cp=new String();
us=usr.getText();
pass=pwd1.getText();
cp=cpwd.getText();

try
{
   Class.forName("oracle.jdbc.driver.OracleDriver");
   Connection con=DriverManager.getConnection(url, user, pwd);
   Statement st=con.createStatement();
   String s="insert into register values('"+us+"','"+pass+"','"+cp+"')";
   ResultSet rs=st.executeQuery(s);
   if(rs.next())
      {
         JOptionPane.showMessageDialog(null, "Register Successful");
      }
}
catch(Exception e)
{
   System.out.println(e);
}
```

5.2.7 Table Content

```
String url="jdbc:oracle:thin:@localhost:1521:XE";
String user="preetham";
String pwd="Preetham";
try
{
   Class.forName("oracle.jdbc.driver.OracleDriver");
   Connection con=DriverManager.getConnection(url, user, pwd);
   Statement st=con.createStatement();
   String s="select * from doctor";
   ResultSet rs=st.executeQuery(s);
   jTable1.setModel(DbUtils.resultSetToTableModel(rs));
}
catch(ClassNotFoundException | SQLException e)
{
   System.out.println(e);
}
```

CHAPTER 6

SNAPSHOTS

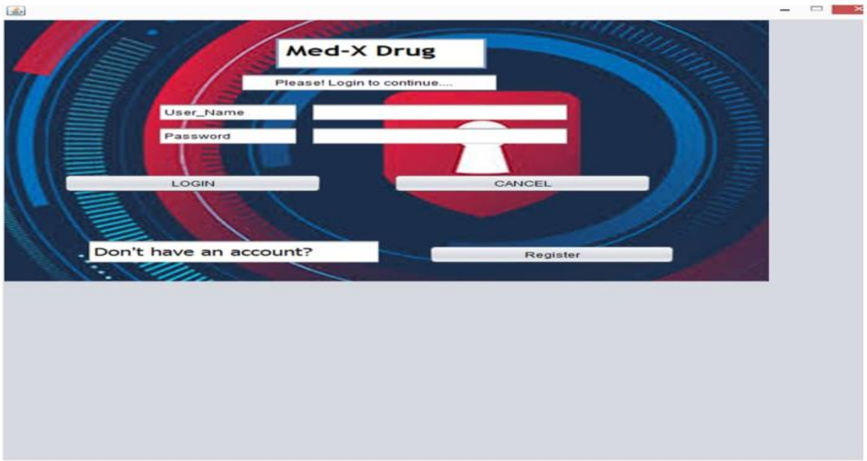


Fig 6.1.login page

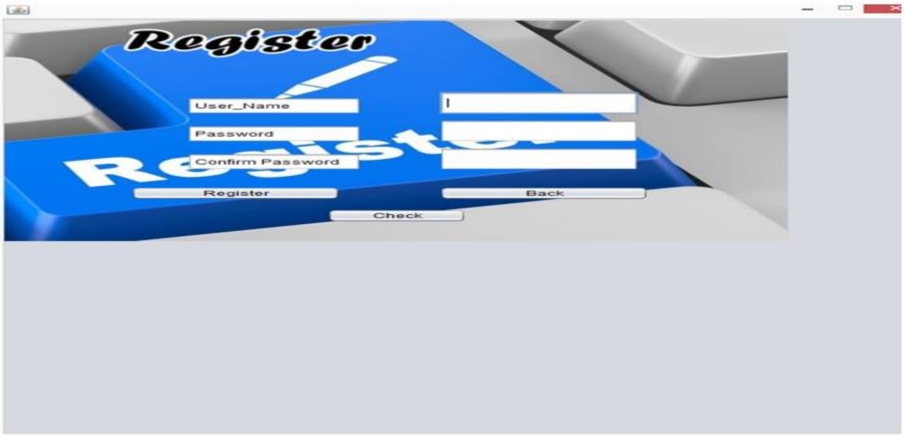
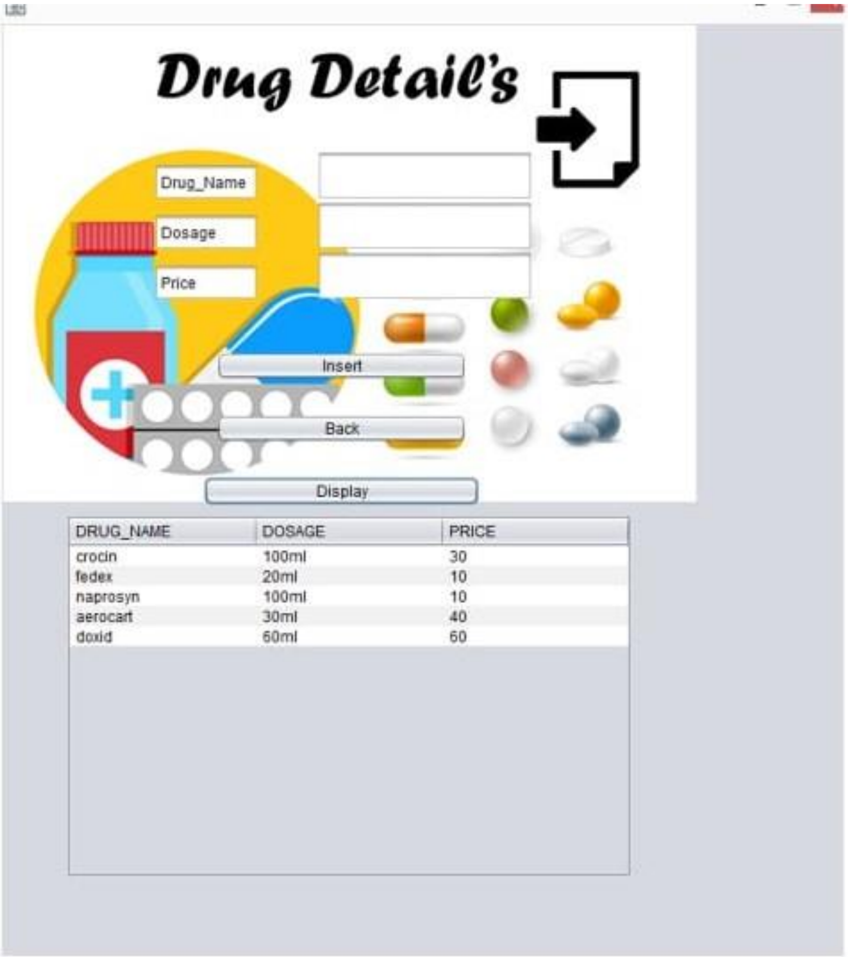
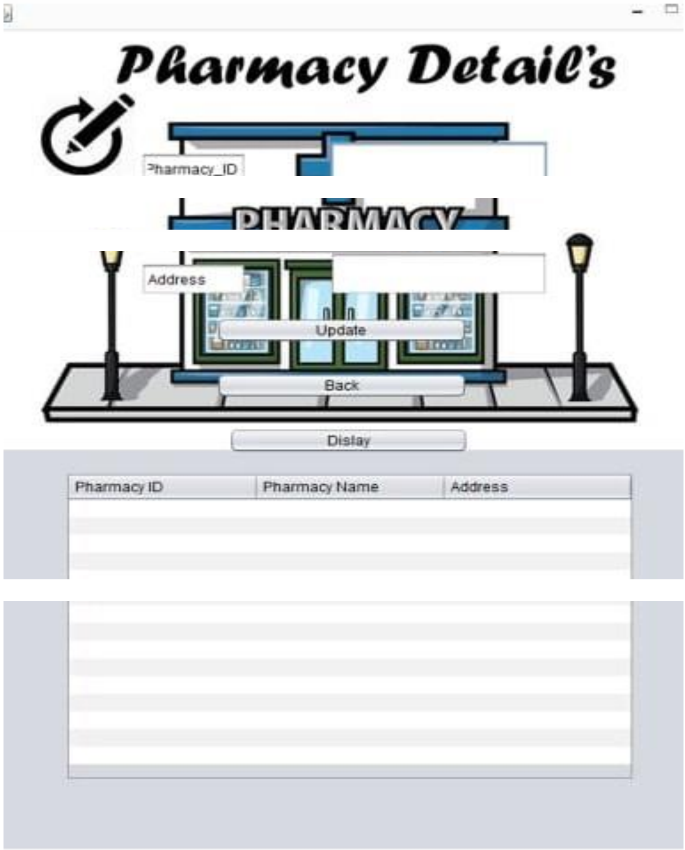


Fig 6.2.register



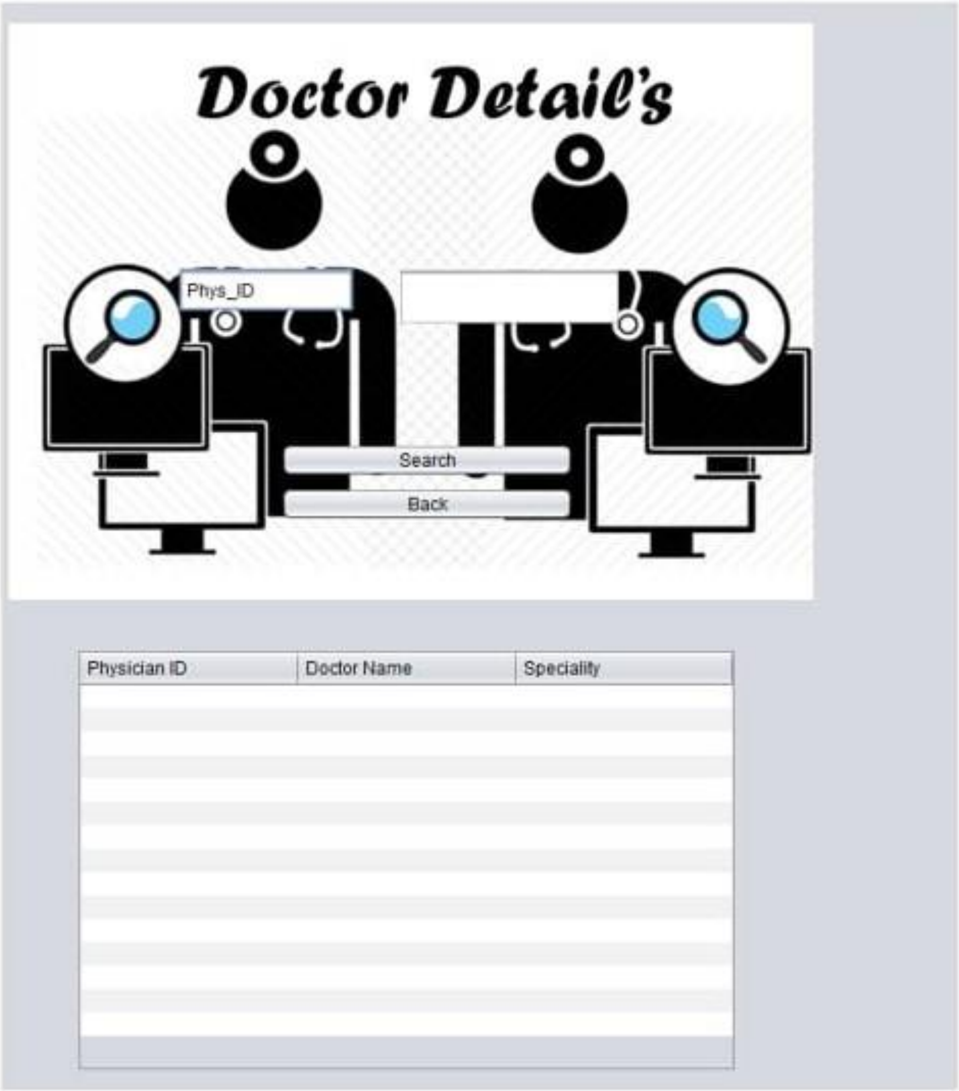
DRUG_NAME	DOSAGE	PRICE
crocin	100ml	30
fedex	20ml	10
naprosyn	100ml	10
aerocart	30ml	40
doxid	60ml	60

Fig 6.3.Insert



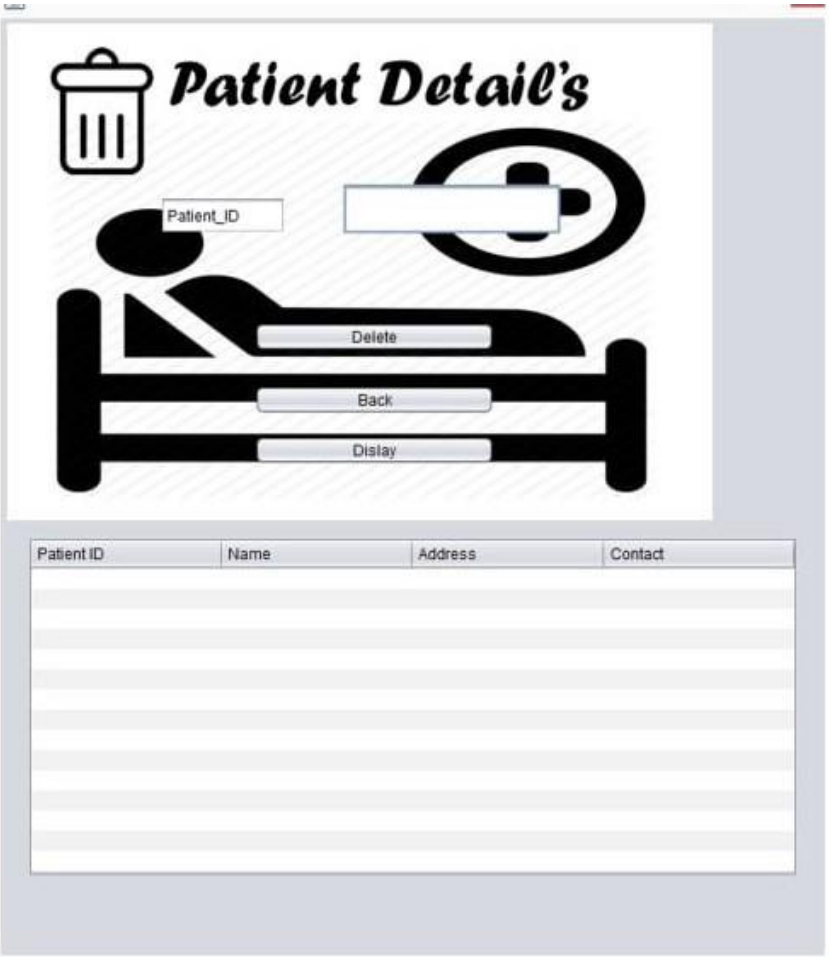
Pharmacy ID	Pharmacy Name	Address

Fig 6.4.Update



Physician ID	Doctor Name	Speciality

Fig 6.5.Search



Patient ID	Name	Address	Contact

Fig 6.6.Delete

CONCLUSION

The following Conclusions can be deduced from the project:
It provides friendly Graphical user interface which proves to be better when compared to the existing Systems. It effectively overcomes the delay of communication. Updating the information becomes so easier. System security, data security and reliability are the striking factors. System has adequate scope for modification in future if necessary.

Advantages

The System is flexible in so far as its branch expansion is concerned. Its very user-friendly, It has the fastest access to database, the errors made manually is more where as using this software we can reduce the errors, it has more storage capacity, it contains better search facility, look and feel environment, the transactions are quick. And it ensures data security.

REFERENCES

[1.] https://stackoverflow.com/

[2.] https://www.w3schools.com/

[3.] https://www.codewithawa.com/

[4.] https://www.hacksmile.com/