



Names: **NSHIMIYIMANA Aldo**

23rd/01/2023

Reg No: **221004695**

College of Business and Economics

School of Economics

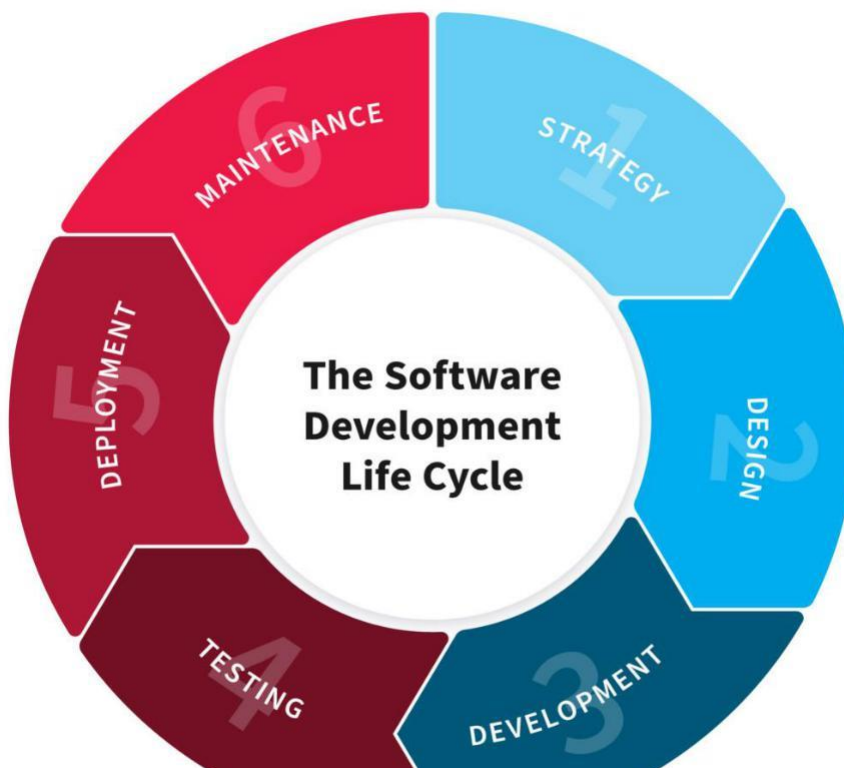
Business Information Technology department (BIT)

Level 2

Class Group: **Group 3**

Project Name: **Pharmacy Management System**

Below is a description of how our team project complies with all the stages of the software development cycle as presented below:



1 STRATEGY.

Online pharmacy management system, also known as the **pharmacy information system**, is a system that stores data and enables functionality that organizes and maintains the medication use process within pharmacies.

These systems may be an independent technology for the pharmacy's use only, or in a hospital setting, pharmacies may be integrated within an inpatient hospital [computer physician order entry \(CPOE\)](#) system

Our application was designed to address the following specific problems:

- ✚ The lack of a user friendly application that helps to monitor drugs inflow and outflow within any given pharmacy.
- ✚ A lack of a multi-user application that supports a shift system which is common in the field of medical care.
- ✚ A lack user identification that lets you know who access the application and when at all times.
- ✚ The lack of a task-oriented system designed and built to address the problem of drugs acquisition, storage and supply.
- ✚ The lack of a reliable system that is not prone to be regularly maintained.

The main goal of the system is to insure that pharmacist/user will be able to access data concerning drugs at any time and apply changes swiftly and effortlessly.

The main objective of the system is to provide a safe platform for pharmacist to perform their everyday work with simplicity and more effectively which would improve the overall process of service delivery in the healthcare system.

Our pharmacy management system aimed and simplicity and efficacy during the implementation of its design to insure a user friendly experience. Below is a description of every page with details on the functionilty of every aspect of the design.

2 DESIGN

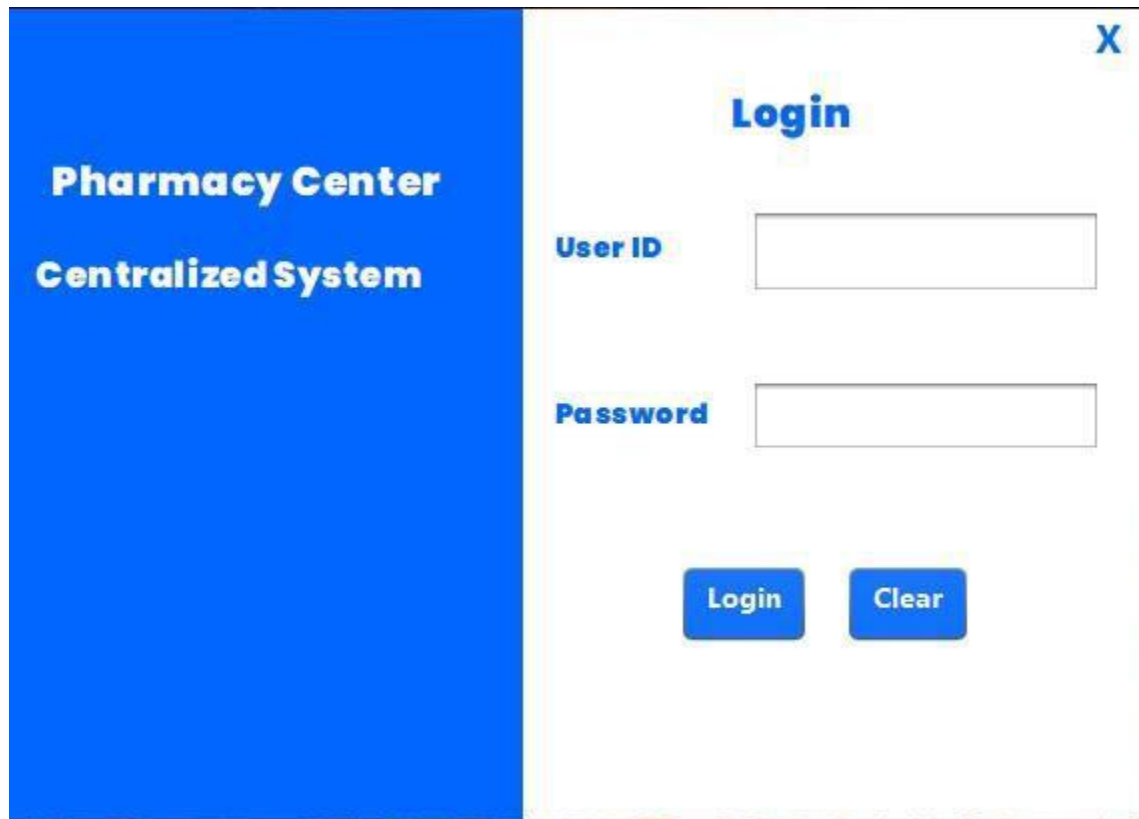
The loading dock splash page



This page illustrates the name of the application. The contains an image that represents the pharmacy logo as well as a progrss bar to display how for the system is loading.

Its key functinality is to provide a starting point for the user to begin to us the application.

The login page



The screenshot shows a web application interface for a 'Pharmacy Center Centralized System'. On the left, a solid blue vertical bar contains the text 'Pharmacy Center' and 'Centralized System' in white. The main content area is white and features a 'Login' title at the top right, accompanied by a close button 'X'. Below the title, there are two input fields: 'User ID' and 'Password', each with a corresponding label to its left. At the bottom of the form, there are two blue buttons labeled 'Login' and 'Clear'.

Pharmacy Center
Centralized System

Login X

User ID

Password

Login **Clear**

The login page contains

- ✚ A text area to hold the user name
- ✚ A text area to hold the user password.

- ✚ An exit button to close the application.
- ✚ A login button to confirm the user credentials and verify if the user is signed in or not.
- ✚ A clear button to dispose of any content within both text areas(password & username)

The main function of the login page is to provide the user with space to identify themselves in order to access the system. Incase the user is no recognised the system outputs the message “WRONG PASSWORD”

The medicine management page

**Company
Agents
Selling**

Medicine Management

ID

FABDATE

NAME

EXPDATE**PRICE****QUANTITY****COMPANY**

Medlab

Add

Update

Delete

Clear

Medicinec List

The medicine management page provides the user with the possibility to operate on drugs and access any information concerning it. The following functionalities are provided:

- ✚ A text field to hold the medicine ID

- ✦ A text field to hold the medicine name.
- ✦ A text field to hold the medicine price.
- ✦ A text field to hold the medicine quantity available in stock.
- ✦ A calendar date chooser to input the medicine fabrication date.
- ✦ A calendar date chooser to input the medicine expiration date.
- ✦ A combo box to allow the user to choose the company supplying the medicine.
- ✦ An add button to add new medicine.
- ✦ An update button to modify medicine information.
- ✦ A delete button to remove any medicine from the medicine table.
- ✦ A clear button to empty the text fields (ID, medname, medprice & medquantity).
- ✦ A medicine table to select any medicine from it.
- ✦ A company label that links to the company management page.

- 🔗 An Agents label that links to the Agents management page.
- 🔗 A selling label that links to the selling management page.

The Agents management page.

Company

Medicines

Selling

Manage Agents

ID

NAME

AGE

PHONE

PASSWORD

Gender

Male

Add

Update

Delete

Clear

Agent list

AId	AName	AAge	APhone	APass	AGender
1	Christian	21	785257997	chris123	Male
2	Lydie	21	785537558	lyd123	Female
3	Charlotte	21	788760547	char123	Female
4	Benitha	21	786431922	zen	Female
5	Aldo	20	788224437	Aldo123	Male

The Agents management page provides the user with the possibility to operate on agents and access any information concerning them. The following functionalities are provided:

- ✚ A text field to hold the Agents ID
- ✚ A text field to hold the Agents name.
- ✚ A text field to hold the Agents age.
- ✚ A text field to hold the Agents phone number (cell).
- ✚ A text field to hold the Agents password.
- ✚ A combo box to allow the user to choose the Agents sex.
- ✚ An add button to add new Agents.

- ✚ An update button to modify Agents information.
- ✚ A delete button to remove any Agents from the Agents table.
- ✚ A clear button to empty the text fields (ID, Agent name, Agent age, Agent phone & Agent password).
- ✚ An Agent table to select any Agent from it.
- ✚ A company label that links to the company management page.
- ✚ A medicine label that links to the medicine management page.
- ✚ A selling label that links to the selling management page.

The company management page.

Medicines
Agents
Selling

Company Management

ID

EXPERIENCE

NAME

ADDRESS

PHONE

Add

Update

Delete

Clear

Company List

CompID	CompName	CompAd	CompExp	CompPhone
1	Medlab	huston texas	23	456777888
2	Medcare	nashville tennesse	21	456777111
3	MedPlan	washington DC	32	456777333
4	PharmaCare	New York	30	456777222
5	unirwanda	kicukiro kigali	24	456788998

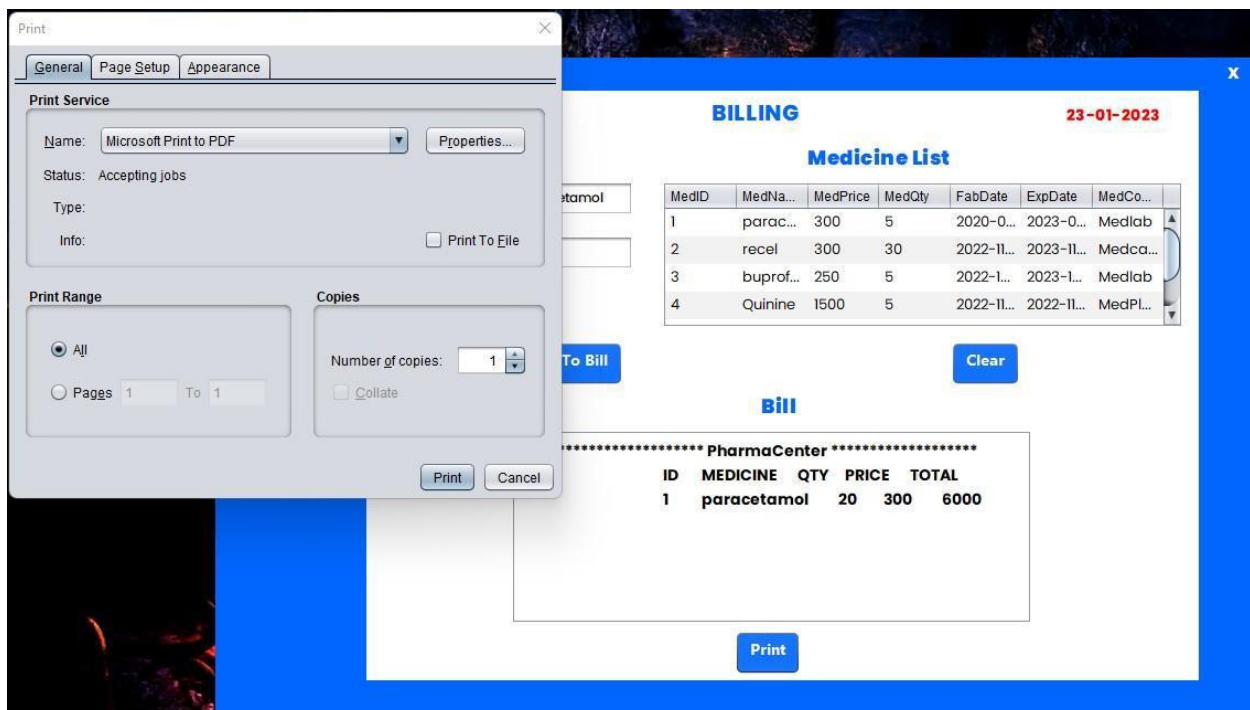
The company management page provides the user with the possibility to operate on company supplying medicine to the pharmacy and access any information concerning them.

The following functionalities are provided:

- ✚ A text field to hold the company ID
- ✚ A text field to hold the company name.
- ✚ Text fields to hold the company address.
- ✚ A text field to hold the company phone number (cell).

- ✦ A text field to hold the company experience. This field holds how much time the company has been operating and manufacturing medical drugs and supplies.
- ✦ An add button to add new company.
- ✦ An update button to modify company information.
- ✦ A delete button to remove any company from the company table.
- ✦ A clear button to empty the text fields (ID, company name, company address, company phone & company experience).
- ✦ A company table to select any company from it.
- ✦ A medicine label that links to the medicine management page.
- ✦ An Agents label that links to the Agents management page.
- ✦ A selling label that links to the selling management page.

The selling management page.



The selling management page provides the user with the possibility to operate on sells of medicine to the clients, create bills and access any information concerning sells.

The following functionalities are provided:

- ✚ A text field to hold the medicine name to be sold.
- ✚ A text field to hold the medicine quantity to be sold.

- ✚ A text area to hold the medicine to be sold that is added to the bill.
- ✚ An Add to bill button the medicine to the bill.
- ✚ A clear button to empty the text fields (ID, medicine name & medicine quantity).
- ✚ A print button to print the bill.
- ✚ A company table to select any company from it.
- ✚ A medicine label that links to the medicine management page.
- ✚ A company label that links to the company management page.
- ✚ An Agents label that links to the Agents management page.

3 DEVELOPMENT

As mentioned previously in the strategy phase of the application development life cycle, our application is a java based web application. Both the front-end and backend were implemented with the help of Apache NetBeans IDE version 15 application.

The following libraries were used:

- ✚ **Mysql-connector-j-8.0.31.jar** to connect to the database.
- ✚ **PostgreSQL JDBC Driver – psorgresql-442.2.16.jar** to access the database in the application (Apache NetBeans IDE 15).
- ✚ **Commons-dbutils-1.5-sources.jar** to link up the tables created in the design to the tables in the MySQL database.
- ✚ **rs2xml.jar** to manipulate the data input into the tables and allow selection.

✚ **jcalendar-1.4.jar** to create a jcalendar field on the medicine management page that allows the user to select fabrication date as well as expiration date.

✚ **JDK 19** to allow the computer to read jar based files.

The storage database used in the development of this application is **MySQL** with the use of a XAMPP control panel to start the server

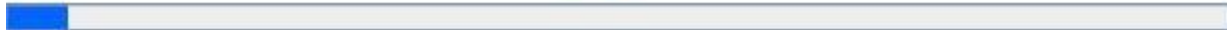
The following highlight the codes used to create the application's functionality:

PHARMACY MANAGEMNT SYSTEM

X



5%



CODES

```
public class SPLASH extends javax.swing.JFrame {  
  
    public SPLASH() {  
  
        initComponents();  
  
    private void  
        jLabel2MouseClicked(java.awt.event.MouseEvent evt) {  
            System.exit(0);  
        }  
    }  
}
```

```
}
```

```
public static void main(String args[]) {
```

```
SPLASH Mysplash = new SPLASH();  
Mysplash.setVisible(true);
```

```
try
```

```
{
```

```
for (int i = 0; i<= 100; i++)
```

```
{
```

```
Thread.sleep(10000);
```

```
Mysplash.Myprogress.setValue(i);
```

```
Mysplash.Percentage.setText(Integer.toString(i)+"%");
```

```
}
```

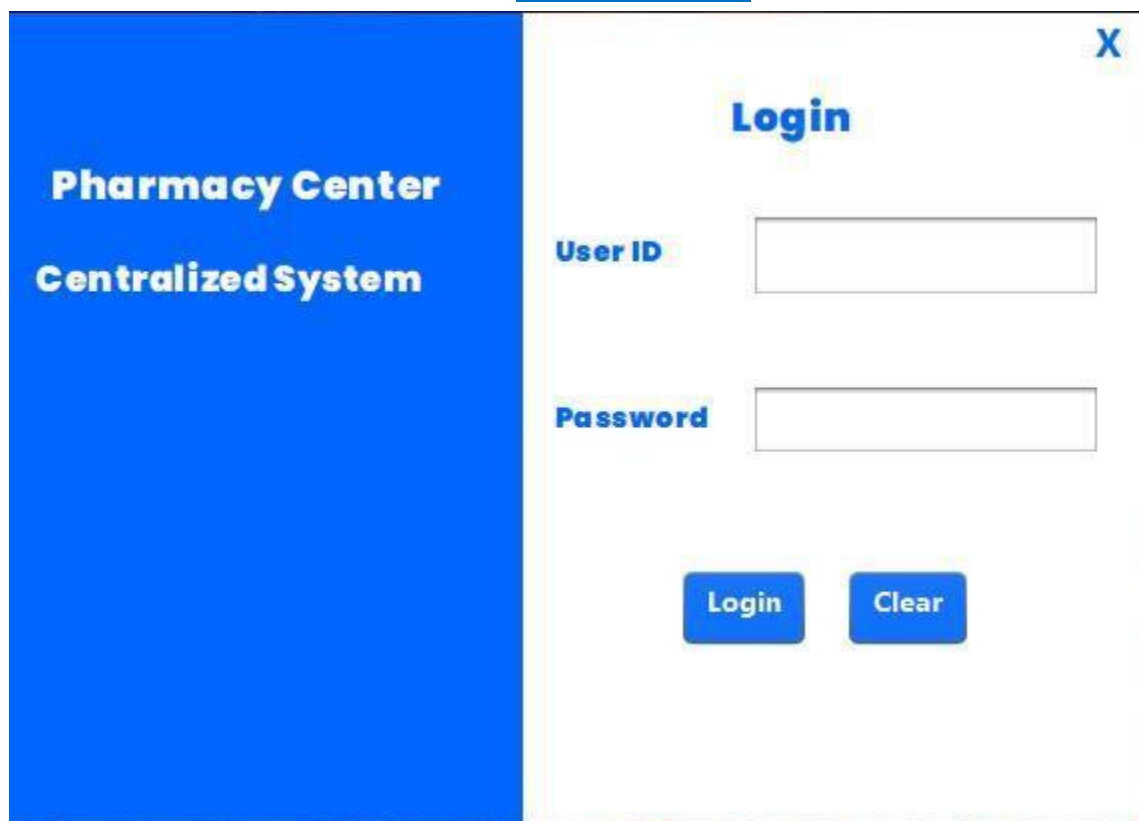
```
}
```

```
catch (Exception e)
{
}
Mysplash.dispose();

new Login().setVisible(true);
}

}
```

The login page



The screenshot shows a login window with a blue sidebar on the left containing the text "Pharmacy Center" and "Centralized System". The main white area is titled "Login" and features a close button "X" in the top right corner. It includes two input fields: "User ID" and "Password". Below these fields are two blue buttons labeled "Login" and "Clear".

Pharmacy Center Centralized System	Login X	
	User ID	<input type="text"/>
	Password	<input type="password"/>
	<input type="button" value="Login"/>	<input type="button" value="Clear"/>

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;

import javax.swing.JOptionPane;
import java.sql.SQLException;
import java.sql.ResultSet;

/**
 *
 * @author RWIGEMA
 * P.Christian */
public class Login extends javax.swing.JFrame {
```

```
public Login() {  
    initComponents();  
}
```

```
Connection Con = null;
```

```
Statement St = null;
```

```
ResultSet Rs = null;
```

```
private void jButton2MouseClicked(java.awt.event.MouseEvent evt) {
```

```
    String Query = "select * from agenttbl where AName = '" + Uid.getText() + "'  
and APass = '" + Pass.getText() + "'";
```

```
    try
```

```
    {
```

```
        Con =
```

```
DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate  
TimeBeh
```

```
avior=CONVERT_TO_NULL", "root", "");
```

```
    St = Con.createStatement();
```

```
    Rs = St.executeQuery(Query);
```

```
    if (Rs.next())
```

```
    {
```

```
        new Medicine().setVisible(true);
```

```
        this.dispose();
```

```
    }
```

```
    else
```

```
    {
```

```
        JOptionPane.showMessageDialog(this, "WRONG PASSWORD");
```

```
    }
```

```
    }
```

```
    catch(SQLException e)
```

```
    {
```

```
        e.printStackTrace();
```

```
    }
```

```
}
```

```
private void
```

```
    jLabel7MouseClicked(java.awt.event.MouseEvent evt) {  
        System.exit(0);
```

```
}
```

```
private void
```

```
    ClearBtnMouseClicked(java.awt.event.MouseEvent evt) {  
        Uid.setText("");  
        Pass.setText("");
```

```
}
```

```
/**
```

```
 * @param args the command line arguments
```

*/

```
public static void main(String args[]) {  
    java.awt.EventQueue.invokeLater(new  
        Runnable() {
```

```
        public void run() {
```

```
            new Login().setVisible(true);
```

```
        }  
    });
```

```
    }
```

```
}
```

The medicine management page

Company
Agents
Selling

Medicine Management

ID

FABDATE

NAME

EXPDATE

PRICE

COMPANY

Medlab

QUANTITY

Add

Update

Delete

Clear

Medicinec List

MedID	MedName	MedPrice	MedQty	FabDate	ExpDate	MedComp
1	paracetamol	300	5	2020-01-13	2023-01-13	Medlab
2	recel	300	30	2022-11-28	2023-11-28	Medcare
3	buproffine	250	5	2022-12-02	2023-12-02	Medlab
4	Quinine	1500	5	2022-11-23	2022-11-12	MedPlan
5	Amoxy	150	25	2022-11-23	2022-11-12	PharmaCare
6	qwerty	123	5	2023-01-05	2023-01-05	Medlab

CODES

```
import java.sql.Statement;  
import java.sql.Connection;
```

```
import java.sql.SQLException;  
import java.sql.ResultSet;  
import java.sql.PreparedStatement;
```

```
import java.sql.DriverManager;  
import javax.swing.JOptionPane;  
import javax.swing.table.DefaultTableModel;
```

```

import net.proteanit.sql.DbUtils;
public class Medicine extends javax.swing.JFrame {

    public Medicine() {

        initComponents();
        SelectMed();
        GetCompany();
    }

    Connection Con = null;
    Statement St = null;
    ResultSet Rs = null;

    java.util.Date FDate, EDate;
    java.sql.Date MyFabDate, MyExpDate;
    public void SelectMed()
    {
        try {
            Con =

            DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDateBehavior=CONVERT_TO_NULL", "root", "");

            St = Con.createStatement();

            Rs = St.executeQuery("Select * from medicinetable");
            MedicineTable.setModel(DbUtils.resultSetToTableModel(Rs));

        }

        catch(SQLException e)
        {

            e.printStackTrace();
        }

    }

    public void GetCompany()

```

```
{  
try  
{  
    Con =
```

```
DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate  
TimeBeh
```

```
avior=CONVERT_TO_NULL" ,"root", "");  
    St = Con.createStatement();  
    String query = "Select * from companytbl";  
    Rs = St.executeQuery(query);
```

```
while(Rs.next())  
{
```

```
    String MyComp =  
    Rs.getString("CompName");  
    CompCb.addItem(MyComp);
```

```

    }
}
catch(SQLException e)
{

    e.printStackTrace();
}
}

private void
AddBtn1MouseClicked(java.awt.event.MouseEvent evt) {
FDate = FabDate.getDate();

MyFabDate = new
java.sql.Date(FDate.getTime()); EDate =
ExpDate.getDate();

MyExpDate = new java.sql.Date(EDate.getTime());

    try {
Con =

DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
TimeBeh
avior=CONVERT_TO_NULL" ,"root", "");

    PreparedStatement add = Con.prepareStatement("insert into medicinetbl
values (?, ?, ?, ?, ?, ?, ?)");

    add.setInt(1,
Integer.valueOf(MedId.getText()));
add.setString(2, MedName.getText());
add.setInt(3, Integer.valueOf(MedPrice.getTex
t()));
add.setInt(4, Integer.valueOf(MedQty.getText(
))); add.setDate(5, MyFabDate);

add.setDate(6, MyExpDate);
add.setString(7, CompCb.getSelectedItem().toString());

int row = add. executeUpdate();

```

```
JOptionPane.showMessageDialog(this, "Medicine Added Successfully");

Con.close();
SelectMed();

}
catch(SQLException e)
{
    e.printStackTrace();
}

}

private void
DeleteBtnMouseClicked(java.awt.event.MouseEvent evt) { if
(MedId.getText().isEmpty())

{
    JOptionPane.showMessageDialog(this, "Enter Medicine to be Deleted");
}
else
```

```

{
    try
    {
        Con =
            DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
            TimeBeh
            avior=CONVERT_TO_NULL" ,"root", "");
        String Id = MedId.getText();
        String Query = "Delete from medicin tbl where MedId="+Id;

        Statement Add = Con.createStatement();
        Add.executeUpdate(Query);
        SelectMed();
        JOptionPane.showMessageDialog(this, "Medicine Deleted successfully");
    }
    catch(SQLException e)
    {
        e.printStackTrace();
    }
}

private void
MedicineTableMouseClicked(java.awt.event.MouseEvent evt) {
    DefaultTableModel model =
    (DefaultTableModel)MedicineTable.getModel(); int Myindex =
    MedicineTable.getSelectedRow();
    MedId.setText(model.getValueAt(Myindex, 0).toString());
    MedName.setText(model.getValueAt(Myindex, 1).toString());
    MedPrice.setText(model.getValueAt(Myindex, 2).toString());
    MedQty.setText(model.getValueAt(Myindex, 3).toString());

}

private void
UpdateBtnMouseClicked(java.awt.event.MouseEvent evt) {
    if(MedId .getText().isEmpty() ||
    MedName.getText().isEmpty() ||

```

```
MedPrice.getText().isEmpty() || MedQty.getText().isEmpty())
{
    JOptionPane.showMessageDialog(this, "Missing Information");
}
else

{
    try
    {
        FDate = FabDate.getDate();

        MyFabDate = new java.sql.Date(FDate.getTime());
        EDate = ExpDate.getDate();
        MyExpDate = new java.sql.Date(EDate.getTime());
```

```

        Con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
TimeBeh
avior=CONVERT_TO_NULL" ,"root", "");
        String Id = MedId.getText();

        String UpdateQuery = "Update medicinetbl set MedName =
""+MedName.getText()+"",MedPrice = ""+ MedPrice.getText()+"",MedQty =
""+MedQty.getText()+"",FabDate = ""+MyFabDate+"",ExpDate =
""+MyExpDate+"",MedComp

=""+ CompCb.getSelectedItem().toString()+" where MedID =" +Id;
        Statement Add = Con.createStatement();
        Add.executeUpdate(UpdateQuery);
        JOptionPane.showMessageDialog(this, "Medicine Update
successfully");

    }
    catch (SQLException e)
    {
        e.printStackTrace();
    }
    SelectMed();
}

}

private void
ClearBtnMouseClicked(java.awt.event.MouseEvent evt) {
MedId.setText("");

MedName.setText("");
MedPrice.setText("");
MedQty.setText("");

}

private void
jLabel2MouseClicked(java.awt.event.MouseEvent evt) {
new Company().setVisible(true);

```



```
    this.dispose();  
}
```

```
private void  
    jLabel4MouseClicked(java.awt.event.MouseEvent evt) {  
        new Agents().setVisible(true);  
        this.dispose();  
    }
```

```
private void  
    jLabel1MouseClicked(java.awt.event.MouseEvent evt) {  
        new Selling ().setVisible(true);  
        this.dispose();  
    }
```

```
private void  
    jLabel13MouseClicked(java.awt.event.MouseEvent evt) {  
        System.exit(0);  
    }
```

$$\}$$

/**

```
* @param args the command line
arguments */
```

```
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new
        Runnable() {

            public void run() {

                new Medicine().setVisible(true);
            }
        });
}
```

The Agents management page.

Company

Medicines

Selling

Manage Agents

ID

NAME

AGE

PHONE

PASSWORD

Gender

Male

Add

Update

Delete

Clear

Agent list

AId	AName	AAge	APhone	APass	AGender
1	Christian	21	785257997	chris123	Male
2	Lydie	21	785537558	lyd123	Female
3	Charlotte	21	788760547	char123	Female
4	Benitha	21	786431922	zen	Female
5	Aldo	20	788224437	Aldo123	Male

CODES

```
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.PreparedStatement;  
  
import java.sql.SQLException;  
import java.sql.Statement;  
import javax.swing.JOptionPane;  
import javax.swing.table.DefaultTableModel;
```

```
import net.proteanit.sql.DbUtils;
```

```
/**
```

```
*
```

```
* @author RWIGEMA
```

```
P.Christian */
```

```
public class Agents extends javax.swing.JFrame {
```

```
/**
```

```
* Creates new form
```

```
Agents */
```

```
public Agents() {
```

```
    initComponents
```

```
    ();
```

```
    SelectAgent();
```

```
}
```

```
Connection Con = null;
```

```
Statement St = null;
```

```
java.sql.ResultSet Rs = null;
```

```
public void SelectAgent()
```

```
{
```

```
    try {
```

```
        Con =
```

```
        DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
```

```
TimeBeh
```

```
avior=CONVERT_TO_NULL" ,"root", "");
```

```
        St = Con.createStatement();
```

```

Rs = St.executeQuery("Select * from agenttbl");
AgentTable.setModel(DbUtils.resultSetToTableModel(Rs));

}
catch(SQLException e)

{
    e.printStackTrace();
}

}

private void
AddBtnMouseClicked(java.awt.event.MouseEvent evt) {
    try {
        Con =
        DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
        TimeBeh
        avior=CONVERT_TO_NULL" ,"root", "");

        PreparedStatement add = Con.prepareStatement("insert into agenttbl
        values (?,?,?,?,?,?)");
        add.setInt(1, Integer.valueOf(AId.getText()));
        add.setString(2,AName.getText());

```

```

add.setInt(3,Integer.valueOf(Aage.getText()
t())); add.setString(4,Aphone.getText());
add.setString(5,Apass.getText());

add.setString(6, GenderCb.getSelectedItem().toString());

int row = add. executeUpdate();
JOptionPane.showMessageDialog(this, "Agent Added Successfully");
Con.close();
SelectAgent();

}
catch(SQLException e)
{
    e.printStackTrace();
}
}

private void
ClearBtnMouseClicked(java.awt.event.MouseEvent evt) {
AId.setText("");

AName.setText("");

Aage.setText("");
Aphone.setText("");

Apass.setText("");

}

private void
DeleteBtnMouseClicked(java.awt.event.MouseEvent evt) { if
(AId.getText().isEmpty())
{

JOptionPane.showMessageDialog(this, "Enter Agent to be Deleted");
}

else

```

```

{

try
{
    Con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
TimeBeh
avior=CONVERT_TO_NULL" ,"root", "");

    String Id = AId.getText();

    String Query = "Delete from agenttbl where
AId="+Id; Statement Add =
Con.createStatement();
Add.executeUpdate(Query);

    SelectAgent();
    JOptionPane.showMessageDialog(this, "Agent Deleted successfully");
}
catch(SQLException e)

```

```

    {
        e.printStackTrace();
    }
}

}

```

```

private void AgentTableMouseClicked(java.awt.event.MouseEvent
    evt) { DefaultTableModel model =
    (DefaultTableModel)AgentTable.getModel();

```

```

    int Myindex = AgentTable.getSelectedRow();
    AId.setText(model.getValueAt(Myindex,
    0).toString());
    AName.setText(model.getValueAt(Myindex,
    1).toString());
    Aage.setText(model.getValueAt(Myindex,
    2).toString());
    Aphone.setText(model.getValueAt(Myindex,
    3).toString());
    Apass.setText(model.getValueAt(Myindex,
    4).toString());
}

```

```

private void UpdateBtnMouseClicked(java.awt.event.MouseEvent evt) {

```

```

    if(AId .getText().isEmpty() || AName.getText().isEmpty() ||
    Aage.getText().isEmpty()

```

```

|| Aphone.getText().isEmpty() || Apass.getText().isEmpty())
    {

```

```

        JOptionPane.showMessageDialog(this, "Missing Information");
    }

```

```

    else
    {
        try

```

```

        {
            Con =

```



```
DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
TimeBeh
avior=CONVERT_TO_NULL" ,"root", "");
```

```
String Id = AId.getText();
```

```
String UpdateQuery = "Update agenttbl set AName =
"+AName.getText()+"",AAge= "+ Aage.getText()+"",APhone =
"+Aphone.getText()+"",APass= "+Apass.getText()+"",AGender
="+ GenderCb.getSelectedItem().toString()+" where AId =" +Id;
Statement Add = Con.createStatement();
Add.executeUpdate(UpdateQuery);
JOptionPane.showMessageDialog(this, "Agent Update
successfully");
```

```
    }
    catch (SQLException e)
    {
        e.printStackTrace();
    }
    SelectAgent();
}
}
```

```

private void
    jLabel2MouseClicked(java.awt.event.MouseEvent evt) {
        new Company().setVisible(true);

        this.dispose();
    }

private void
    jLabel4MouseClicked(java.awt.event.MouseEvent evt) {
        new Medicine().setVisible(true);

        this.dispose();
    }

private void
    GenderCbActionPerformed(java.awt.event.ActionEvent evt) { //
        TODO add your handling code here:
    }

private void
    UpdateBtnActionPerformed(java.awt.event.ActionEvent evt) { //
        TODO add your handling code here:
    }

private void
    ClearBtnActionPerformed(java.awt.event.ActionEvent evt) { //
        TODO add your handling code here:
    }

private void
    AIdActionPerformed(java.awt.event.ActionEvent evt) { //
        TODO add your handling code here:
    }

```

```
private void  
    jLabel14MouseClicked(java.awt.event.MouseEvent evt) {  
        new Selling ().setVisible(true);  
  
        this.dispose();  
    }
```

```
private void  
    jLabel1MouseClicked(java.awt.event.MouseEvent evt) {  
        System.exit(0);  
    }
```

```
/**
```

```
 * @param args the command line  
 arguments */  
public static void main(String args[]) {
```

```
    java.awt.EventQueue.invokeLater(new  
        Runnable() { public void run() {  
            new Agents().setVisible(true);
```

```
}  
});  
}  
}
```

The company management page.

Medicines
Agents
Selling

Company Management

ID

EXPERIENCE

NAME

ADDRESS

PHONE

Add

Update

Delete

Clear

Company List

CompID	CompName	CompAd	CompExp	CompPhone
1	Medlab	huston texas	23	456777888
2	Medcare	nashville tennesse	21	456777111
3	MedPlan	washington DC	32	456777333
4	PharmaCare	New York	30	456777222
5	unirwanda	kicukiro kigali	24	456788998

CODES

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import net.proteanit.sql.DbUtils;
```

```
/**
```

```
*
```

```
* @author RWIGEMA
```

```
P.Christian */
```

```

public class Company extends javax.swing.JFrame {

    /**
     * Creates new form
     * Company */

    public Company()
    {
        initComponents();
        SelectCompany();
    }

    Connection Con = null;
    Statement St = null;
    java.sql.ResultSet Rs = null;
    public void SelectCompany()
    {
        try {
            Con =
                DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
                TimeBeh
                avior=CONVERT_TO_NULL", "root", "");
            St = Con.createStatement();

            Rs = St.executeQuery("Select * from companytbl");
            CompanyTable.setModel(DbUtils.resultSetToTableModel(Rs));

        }
        catch(SQLException e)
        {
            e.printStackTrace();
        }
    }

```

```
}
```

```
private void
```

```
    AddBtnMouseClicked(java.awt.event.MouseEvent evt)
```

```
    { try {
```

```
        Con =
```

```
        DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate  
TimeBeh
```

```
avior=CONVERT_TO_NULL" ,"root", "");
```

```
        PreparedStatement add = Con.prepareStatement("insert into companytbl  
values (?, ?, ?, ?, ?)");
```

```
        add.setInt(1, Integer.valueOf(CompId.getText()));
```

```
        add.setString(2, Compname.getText());
```

```
        add.setString(3, Compad.getText());
```

```
        add.setInt(4, Integer.valueOf(Compexp.getText()));
```

```
        add.setString(5, Compphone.getText());
```

```
        int row = add. executeUpdate();
```

```
        JOptionPane.showMessageDialog(this, "Company Added  
Successfully"); Con.close();
```

```

        SelectCompany();
    }
    catch(SQLException e)
    {

        e.printStackTrace();
    }
}

private void
    DeleteBtnMouseClicked(java.awt.event.MouseEvent evt)
    { if (CompId.getText().isEmpty())

{
    JOptionPane.showMessageDialog(this, "Enter company to be Deleted");
}
else
{
    try
    {
        Con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
TimeBeh
avior=CONVERT_TO_NULL", "root", "");

        String Id = CompId.getText();

        String Query = "Delete from companytbl where
        CompID="+Id; Statement Add =
        Con.createStatement(); Add.executeUpdate(Query);
        SelectCompany();

        JOptionPane.showMessageDialog(this, "Company Deleted successfully");
    }

    catch(SQLException e)
    {

        e.printStackTrace();
    }
}

```



```
}  
}
```

```
private void UpdateBtnMouseClicked(java.awt.event.MouseEvent  
    evt) { if(CompId .getText().isEmpty() ||  
        Compname.getText().isEmpty() ||
```

```
    Compad.getText().isEmpty() ||  
    Compexp.getText().isEmpty() ||  
    Compphone.getText().isEmpty())  
    {  
        JOptionPane.showMessageDialog(this, "Missing Innformation");  
    }  
    else  
    {  
        try
```

```
{  
Con =
```

```
DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate  
TimeBeh  
avior=CONVERT_TO_NULL", "root", "");
```

```
String Id = CompId.getText();
```

```
String UpdateQuery = "Update companytbl set CompName =  
"+Compname.getText()+"",CompAd = "+ Compad.getText()+"",CompExp =  
"+Compexp.getText()+"",CompPhone= "+Comppphone.getText()+" where  
CompID =" +Id;
```

```
Statement Add = Con.createStatement();  
Add.executeUpdate(UpdateQuery);  
JOptionPane.showMessageDialog(this, "Company Update successfully");  
}  
catch (SQLException e)  
{  
    e.printStackTrace();  
}  
SelectCompany();  
}
```

```
}
```

```
private void
```

```
ClearBtnMouseClicked(java.awt.event.MouseEvent evt) {  
    CompId.setText("");
```

```
    Compname.setText("");  
    Compad.setText("");  
    Compexp.setText("");
```

```
    Comppphone.setText("");
```

```
}
```

```
private void  
    jLabel2MouseClicked(java.awt.event.MouseEvent evt) {  
        new Medicine().setVisible(true);  
  
        this.dispose();  
    }
```

```
private void  
    jLabel4MouseClicked(java.awt.event.MouseEvent evt) {  
        new Agents().setVisible(true);  
  
        this.dispose();  
    }
```

```
private void  
    CompanyTableMouseClicked(java.awt.event.MouseEvent evt) {  
        DefaultTableModel model =  
            (DefaultTableModel)CompanyTable.getModel(); int Myindex =  
            CompanyTable.getSelectedRow();  
        CompId.setText(model.getValueAt(Myindex, 0).toString());  
        Compname.setText(model.getValueAt(Myindex, 1).toString());  
    }
```

```

        Compad.setText(model.getValueAt(Myindex,
        2).toString());
        Compexp.setText(model.getValueAt(Myindex,
        3).toString());
        Compphone.setText(model.getValueAt(Myindex,
        4).toString());

    }

    private void
        CompnameActionPerformed(java.awt.event.ActionEvent evt) { //
        TODO add your handling code here:

    }

    private void
        jLabel1MouseClicked(java.awt.event.MouseEvent evt) {
        new Selling ().setVisible(true);

        this.dispose();
    }

    private void
        jLabel13MouseClicked(java.awt.event.MouseEvent evt) {
        System.exit(0);

    }

    /**

    * @param args the command line
    arguments */

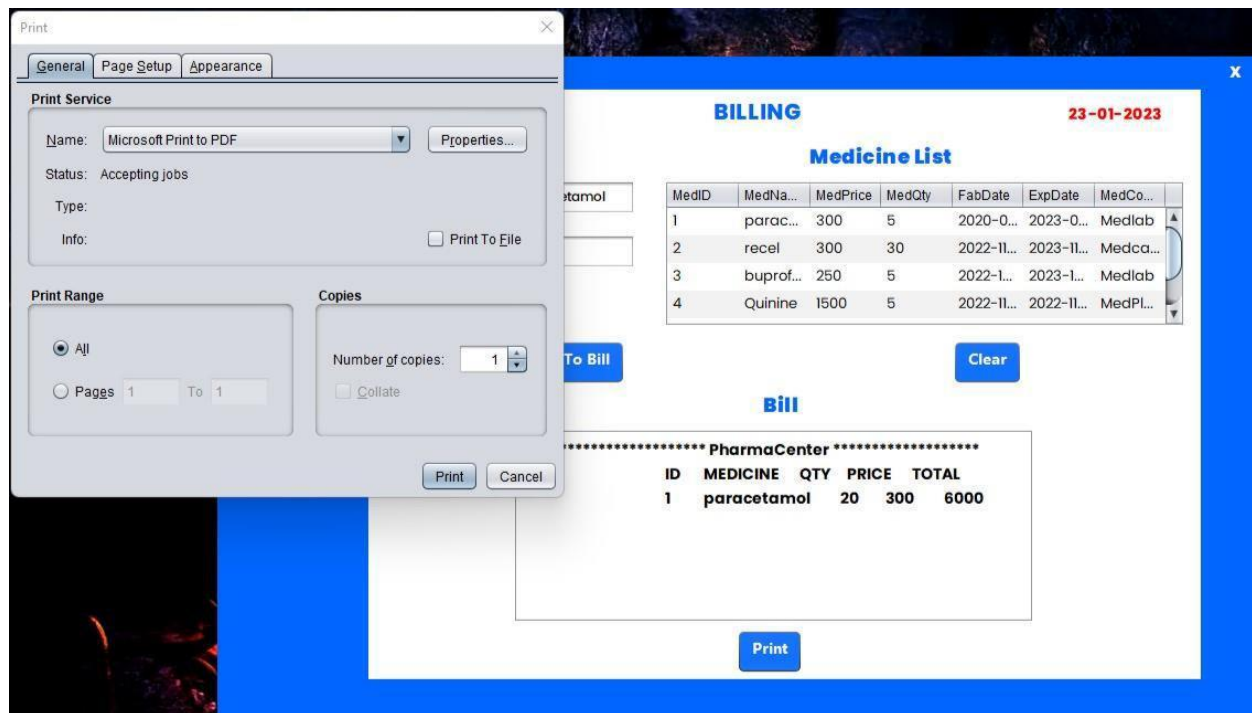
    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new
        Runnable() {

            public void run() {
                new Company().setVisible(true);
            }
        });
    }

```

```
    }  
  });  
  
  }  
}
```

The selling management page.



CODES

```
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.SQLException;  
import java.sql.Statement;  
  
import java.text.SimpleDateFormat;  
import java.util.Date;
```

```
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;

import net.proteanit.sql.DbUtils;

/**
 *
 * @author RWIGEMA
 * P.Christian */

public class Selling extends javax.swing.JFrame {

    /**
     * Creates new form
     * Selling */

    public Selling() {
        initComponents
        ();
    }
}
```

```

        SelectMed();
        ShowDate();

    }

    public void ShowDate()
    {
        Date d = new Date();

        SimpleDateFormat s = new SimpleDateFormat("dd-
        MM-yyyy"); DateLbl.setText(s.format(d));
    }

    Connection Con = null;
    Statement St = null;
    java.sql.ResultSet Rs = null;
    public void SelectMed()
    {
        try {
            Con =

            DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
            TimeBeh

            avior=CONVERT_TO_NULL", "root", "");
            St = Con.createStatement();

            Rs = St.executeQuery("Select * from medicinetbl");
            MedicineTable.setModel(DbUtils.resultSetToTableModel(Rs));

        }
        catch(SQLException e)

            {
                e.printStackTrace();
            }
    }

```



```

private void PrintBtnMouseClicked(java.awt.event.MouseEvent evt) {

    /* if(AId .getText().isEmpty() || AName.getText().isEmpty() ||
Aage.getText().isEmpty() || Aphone.getText().isEmpty() ||
Apass.getText().isEmpty())
    {
        JOptionPane.showMessageDialog(this, "Missing Innformation");
    }
    else

    {
        try
        {
            Con =

DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate
TimeBeh
avior=CONVERT_TO_NULL" ,"root", "");
            String Id = AId.getText();

```

```
String UpdateQuery = "Update agenttbl set AName =  
"+AName.getText()+"",AAge= ""+ Aage.getText()+"",APhone =  
"+Aphone.getText()+"",APass= ""+Apass.getText()+"",AGender = ""+  
GenderCb.getSelectedItemAt().toString()+" where AId =" +Id;
```

```
Statement Add = Con.createStatement();  
Add.executeUpdate(UpdateQuery);  
JOptionPane.showMessageDialog(this, "Agent Update  
successfully");
```

```
}
```

```
catch (SQLException e)  
{  
    e.printStackTrace();  
}
```

```
SelectSells();
```

```
}*/
```

```
try
```

```
{  
    BillTxt.print();  
}
```

```
catch(Exception e)  
{
```

```
    e.printStackTrace();  
}
```

```
}
```

```
private void
```

```
PrintBtnActionPerformed(java.awt.event.ActionEvent evt) { //  
    TODO add your handling code here:
```

```
}
```

```
private void
```

```
ClearBtnMouseClicked(java.awt.event.MouseEvent evt) {  
    MedText.setText("");
```

```
Qty.setText("");

}

private void
ClearBtnActionPerformed(java.awt.event.ActionEvent evt) { //
    TODO add your handling code here:
}
public void Update ()
{

    int newQty;
    newQty = Q1dQty - Integer.valueOf(Qty.getText());
    try
    {
```

Con =

```
DriverManager.getConnection("jdbc:mysql://localhost:3306/pharmacydb?zeroDate  
TimeBeh  
avior=CONVERT_TO_NULL" ,"root", "");
```

```
String UpdateQuery = "Update medicin tbl set MedQty = "+ newQty +"  
where MedID =" + Medid;  
Statement Add = Con.createStatement();
```

```
Add.executeUpdate(UpdateQuery);  
JOptionPane.showMessageDialog(this, "Medicine Update successfully");
```

```
    }  
    catch (SQLException e)  
    {  
        e.printStackTrace();  
    }  
    SelectMed();  
}
```

```
int i = 0, price, Medid , Q1dQty;
```

```
private void AddBtnMouseClicked(java.awt.event.MouseEvent evt) {
```

```
    if (MedText.getText().isEmpty() || Qty.getText().isEmpty())  
    {
```

```
        JOptionPane.showMessageDialog(this, "Missing Information");  
    }
```

```
    else {  
        i++;
```

```
        Update();  
        if(i == 1)
```

```
        {  
            BillTxt.setText(BillTxt.getText() +
```

```
                "\n\t ID    MEDICINE        QTY    PRICE TOTAL\n\t" +  
                "+" "+
```

```

        i+" "+ MedText.getText()
        + " " +
        Qty.getText() + " " + price
        + "
                                " + Integer.valueOf(Qty.getText())*price
                                +
"\n");
    }

    else
    {
        BillTxt.setText(BillTxt.getText() +"\t "+ i + "
                                " +
        MedText.getText() +"
                                "
+Qty.getText() + " " + price + " " + Integer.valueOf(Qty.getText())*price+"\n");

    }
}

```

```
}
```

```
private void
```

```
    jLabel2MouseClicked(java.awt.event.MouseEvent evt) {  
        new Company().setVisible(true);
```

```
        this.dispose();
```

```
}
```

```
private void
```

```
    jLabel4MouseClicked(java.awt.event.MouseEvent evt) {  
        new Medicine().setVisible(true);  
        this.dispose();
```

```
}
```

```
private void
```

```
    MedTextActionPerformed(java.awt.event.ActionEvent evt) { //  
        TODO add your handling code here:
```

```
}
```

```
private void
```

```
    QtyActionPerformed(java.awt.event.ActionEvent evt) { //  
        TODO add your handling code here:
```

```
}
```

```
private void
```

```
    MedicineTableMouseClicked(java.awt.event.MouseEvent evt) {  
        DefaultTableModel model =  
        (DefaultTableModel)MedicineTable.getModel(); int Myindex =  
        MedicineTable.getSelectedRow();
```

```
        //MedId.setText(model.getValueAt(Myindex, 0).toString());  
        MedText.setText(model.getValueAt(Myindex, 1).toString());  
        Medid = Integer.valueOf(model.getValueAt(Myindex,  
        0).toString()); price =  
        Integer.valueOf(model.getValueAt(Myindex, 2).toString());  
        Q1dQty = Integer.valueOf(model.getValueAt(Myindex,  
        3).toString());
```

```
}
```

```
private void
```

```
    jLabel1MouseClicked(java.awt.event.MouseEvent evt) {
```

```
        System.exit(0);
```

```
    }
```

```
private void
```

```
    jLabel14MouseClicked(java.awt.event.MouseEvent evt) {
```

```
        new Agents().setVisible(true);
```

```
        this.dispose();
```

```
    }
```

```
/**
```

```
 * @param args the command line
```

```
 arguments */
```

```
public static void main(String args[]) {
```

```
    java.awt.EventQueue.invokeLater(new
```

```
    Runnable() {
```

```
    public void run() {  
        new Selling().setVisible(true);  
    }  
});  
  
}
```


4 TESTING

The methodology used in the implementation of the software is the Incremental Model of System Development Life Cycle, which allows room for scalability as time goes on. Creating an Online Pharmaceutical Management System would help in pharmaceutical practices for all parties involved.

5 DEPLOYMENT

As usual with most of the policies sample posted here, there is some default data that should be listed in the policy header and footer. Below are some of the information that you need to include in your Maintenance of pharmaceutical supply and stock leveling header:

- Hospital name, obviously.
- Pharmacy department operational policy and procedure, this header just to notify the reader that **the following policy belongs to which department.**
- Policy **code** or Number, for filing/indexing purposes.
- **Edition** Number, same purpose as above.
- Title of the policy, in our case; it would be **maintenance of pharmaceutical supply & stock leveling.**
- **Important dates** [date reviewed, approved date, effective date and due for review date].
- **Applies to [department]** in this case it should be pharmacy department
- And last but not least, you'll mention the **nature of the policy**, whether it's going to be multidisciplinary or department specific. (**Department-specific policy.**

