Project Report

Pharmacy Management System

Prepared by:

Dehya Khurraim Siddiqui 20K-0128

Tayyab Sahazad 20K-1043

Muhammad Osama 20K-1725

DB Fall 2022

FAST NUCES Karachi

Table of Contents

1. Introduction

Due to the size and quality of service of the pharmacy, pharmacy has a very large customer base. These clients typically visit the drugstore for services after they leave work. The number of consumers visiting the pharmacy is growing currently, which adds to the pharmacists' already arduous workload.

It is challenging for the pharmacist in this situation to serve consumers quickly. To retain consumers, the pharmacist must make sure that the services are satisfactory. The aforementioned factors cause customers to wait longer for services, which slows down sales and increases the chance of losing important clients in the long run.

The pharmacy management system is a type of management that aims to increase accuracy while also boosting efficiency and safety in the drugstore. It is a computer-based system that assists the pharmacist in managing inventory, costs, medical safety, and other factors. During opening stock and sales operations, the system allows the user to enter the manufacturing and expiry dates for a specific product or drug. Additionally, it requires human entry whenever new batches of medications are received.

1.1 Purpose

The goal of this project is to provide software for the efficient management of a drugstore that can accomplish the following goals: By giving statistics on the drugs in stock and a way to update the database, ensuring effective service and activity monitoring in order to increase system efficiency.

1. *To manage the pharmacy's drug inventory as best as possible by keeping an eye on the flow of medications there.*
2. *To make sure that different functions and roles have different levels of limited access.*
3. *To make sure the interface is simple to use.*
4. *To be able to create a view of the customer's ordered medication list.*
5. *To be able to generate bills after a successful purchase.*

1.2 Product Scope

The scope of this project is limited to the activities of a pharmaceutical store which includes will improving health outcomes, reducing hospital and long-term care admissions, enhancing access and care in the Estate and surrounding communities, and ensuring the best use of resources, the use of a computer-based management system for improving the efficiency of a pharmacy is needed and it is an essential part of any modern continuously evolving society. The project's focus is only on the activities of a pharmacy. Drug prescriptions and drug interactions won't be handled by the system. Contraindication and polypharmacy in a prescription will not be handled by the system; this suggests that the pharmacist will need to manually finish these services.

2. Overall Description

2.1 Product Perspective

The Pharmacy Management System serves many purposes, including the safe and effective dispensing of pharmaceutical drugs. Along with this storing the medication data and updating it to update the inventory. Storing the necessary information of clients and vendors to make the process smooth.

2.2 Product Functions

The Pharmacy Management System is a computer software system that is programmed with the ability to perform different tasks required to operate a pharmacy. They make working easier and allow for the digital record stage and fast retrieval of information. Medical technology is increasing day by day.

The developer uses this opportunity to help the customers in selecting and purchasing the right medicine for the patient. A pharmacy management system is used to buy medicines easily and store the client's and vendor's data smoothly. The system is used in superstores and online pharmacies.

2.3 Operating Environment

This software is expected to run efficiently in the following atmosphere

1. *Operating System: Windows 7 or higher*
2. *Platform: XAMPP*
3. *Database: MySQL*
4. *Technologies used: Java, JSP*

2.4 Design and Implementation Constraints

For the system to operate properly, certain rules and restrictions must be followed. Nobody can display their products on this website unless they have a legitimate profile that has been approved and validated by the administrator. Similarly, no one is permitted to use the services of this website without a valid customer account. The product may be returned for modifications, to rectify the damage, or to be replaced if presumed damage (expired medications) is notified at the time of purchase. Any delays in returning the item must be communicated in advance to the vendor. The customer will be invoiced for any damage discovered during the damage control inspection upon return.

3. External Interface Requirements

3.1 User Interfaces

System is designed in such a way that the vendor can add medicine information which they are selling and customer searches for the medicine he/she wants and if it is available the customer orders the medicine. Other than that vendor can also restock and delete the medicine.

Home Page
It is the first interface that appears on the screen when the application is being loaded. This interface gives 3 options: Login, Register as a Customer & Register as a Vendor.

Register as a Customer
This interface receives information from the customer, such as Name, ID, Contact No, Address, Email, and Password to be set. And stores the information into database.

Register as a Vendor
This interface receives information from the vendor, such as Name, ID, Contact No, and Password to be set. And stores the information into database.

Login
This interface first asks user whether to login as a Customer or Vendor. Then ask their respective ID and password.

Seller Home Page
This interface displays a picture and information of Vendor. There is also a navigation bar to help user navigate through the system.

Add Product
This interface receives information from the vendor regarding medicines, such as Medicine Name, ID, Formula, Company Name, MFG date, EXP date, quantity, and price. And stores the information into database.

Restock
This interface display several boxes, each box display a picture and information about respective medicine. Each box also has 2 buttons for restocking the medicine and deleting the medicine.

Order
This displays order information (such as medicine ID, customer ID, quantity, total price) of order received by the respective vendor.

Customer Home Page
This interface displays a picture and information of Customer. There is also a navigation bar to help user navigate through the system.

Buy
This interface display several boxes, each box display a picture and information about respective medicine. Each box also has buttons for ordering the medicine.

Order
This displays order information (such as medicine ID, vendor ID, quantity, total price) of order respective customer made.

3.2 Hardware Interfaces

- *Microcontroller: Uno Board*
- *Humidity, Air Quality, Pressure*
- *Processor: Core i3 or higher*
- *Processor speed: 2.6GHz*
- *RAM: 2 GB*
- *Disk Space: 2 GB or higher*
- *It should have a proper internet connection*

3.3 Software Interfaces

- *Operating System: Windows 7 or higher*
- *Platform: XAMPP, Tomcat Server*
- *Database: MySQL*
- *Technologies used: HTML, CSS, Java, JSP, Java Servlet, SQL.*

4. System Requirements

The pharmacy monument system provides practicality for managing medication inventory, minimizing work and human errors concerning handling the stocks of medicines, and facilitating the accessibility of medicines. The system can solve the matter of the present system by minimizing time wastage and reducing resources.

4.1 Functional Requirements

There are functions done by the system like storing the required data of medicines, simply looking medicines, updating, deleting & saving knowledge of medicines, storing necessary data of client and seller, manage order of medicines.

- ***Store the required data on medications:*** *Davlo stores the elaborate data regarding every medicine as well as Name, ID, name, producing Date, Expiring Date, Quantity, and Price.*
- ***Easily looking medications:*** *Davlo provides practicality to look data regarding medicine by looking the medicine's name.*
- ***Update & delete information of medication:*** *Davlo provides practicality to stock the number of medicine, and delete the data of the medication.*
- ***Store necessary data of client:*** *Davlo stores elaborate data regarding every customer as well as Name, Contact variety, Address, Email Address and etc.*
- ***Store necessary data of vendor:*** *Davlo stores elaborate data regarding every merchant as well as Name, Contact variety, Address, Email Address and etc.*
- ***Manage order of medications****: Davlo show the client data a few medicines they ordered as well as the medication Name, Vendor ID, Quantity, Unit Price, and Total worth. Davlo conjointly shows the seller data of drugs they need received the order as well as medication Name, client ID, client Address, Quantity, Unit Price, and Total worth.*

The functionality of system is:

- *Login.*
- *Register as client.*
- *Register as seller.*
- *Add medication.*
- *Restock medication.*
- *Delete data of drugs.*
- *View the orders a selected seller receives.*
- *Buy medication.*
- *View the list of medicines you ordered.*

4.2 Non-Functional Requirements

- *This pharmacy management system will operate within the following characteristics:*
- *The system ought to be simple to use and learn in each facet. The new users ought to get wont to the system quick as potential.*
- *The system ought to give the choice to increase hardware and different services within the future.*
- *The info shall be out there ninety nine of the time throughout business hours.*

- *The info shall be out there ninety fifth of the time throughout the non- business hours.*
- *No info question shall take quite five seconds with just one info affiliation active.*
- *No page shall take quite a pair of seconds to access with just one user on-line.*
- *The system ought to store up to forty GB of information.*
- *The pharmacy management system ought to be operated and controlled by the admin for security functions.*

5. Normalization

oid, datetime, quantity, price, uid, fname, lname, email, add, phone, pid, pname, manufacture, mfg, exp, price, sid, sname, s_pass, s_add, s_phone

PRIMARY KEY:

oid, uid, pid, sid

FUNTIONAL DEPENDACYIES:

oid -> {datetime, quantity, price}
uid -> {pass, fname, lname, email, add, phone}
pid -> {pname, manufacture, exp, price}
sid -> {same, pass}
pid -> quantity

1st NF:

Already in 1st NF

2nd NF:

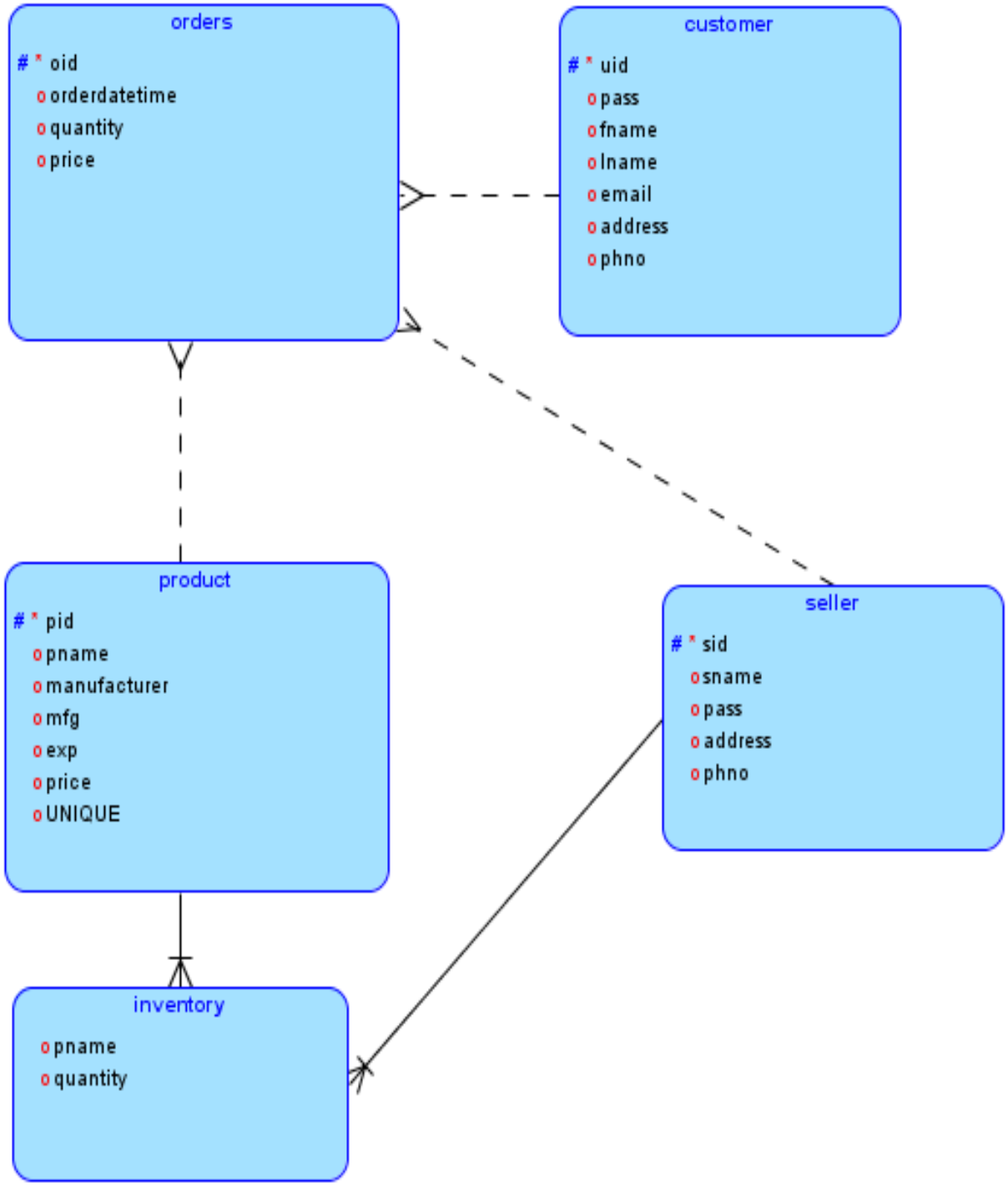
oid, datetime, quantity, price
uid, fname, lname, email, add, phone
pid, pname, manufacture, mfg, exp, price
sid, sname, s_pass, s_add, s_phone

3rd NF:

oid, datetime, quantity, price
uid, fname, lname, email, add, phone
pid, pname, manufacture, mfg, exp, price
sid, sname, s_pass, s_add, s_phone
pid, pname, quanity

6. System Design

6.1 ER Model



7. Code Screenshot

7.1 Triggers

```
64      *Triggers*
65
66
67      DELIMITER //
68
69      CREATE TRIGGER updatetime BEFORE INSERT ON orders FOR EACH ROW
70      BEGIN
71          SET NEW.orderdatetime = NOW();
72      END//
73
74      DELIMITER ;
78      DELIMITER //
79      CREATE TRIGGER inventorytrigger AFTER INSERT ON orders
80      FOR EACH ROW
81      begin
82
83      DECLARE qnty int;
84      DECLARE productid varchar(20);
85
86      SELECT   pid INTO productid
87      FROM      orders
88      ORDER BY  oid DESC
89      LIMIT     1;
90
91      SELECT   quantity INTO qnty
92      FROM      orders
93      ORDER BY  oid DESC
94      LIMIT     1;
95
96      UPDATE inventory
97      SET quantity=quantity-qnty
98      WHERE pid=productid;
99      END//
100
101     DELIMITER ;
```

7.2 Procedures

```
105     *Stored Procedures*
106
107
108     DELIMITER //
109
110 •   CREATE PROCEDURE getsellerorders(IN param1 VARCHAR(20))
111     BEGIN
112         SELECT *  FROM orders where sid=param1;
113     END //
114
115     DELIMITER ;
```

```
119     DELIMITER //
120
121 •   CREATE PROCEDURE getorders
122     (IN param1 VARCHAR(20))
123     BEGIN
124        SELECT * FROM orders WHERE uid=param1;
125     END //
126
127     DELIMITER ;
```

7.3 Views

```
75      -- *View*
76
77 •   CREATE VIEW orderdetails AS SELECT * FROM orders;
78
79 •   CREATE VIEW customerdetails AS SELECT  * FROM customer;
80
81 •   CREATE VIEW inventorydetails AS SELECT  * FROM inventory;
82
83 •   CREATE VIEW productdetails AS SELECT * FROM product;
84
85 •   CREATE VIEW sellerdetails AS SELECT  * FROM seller;
```

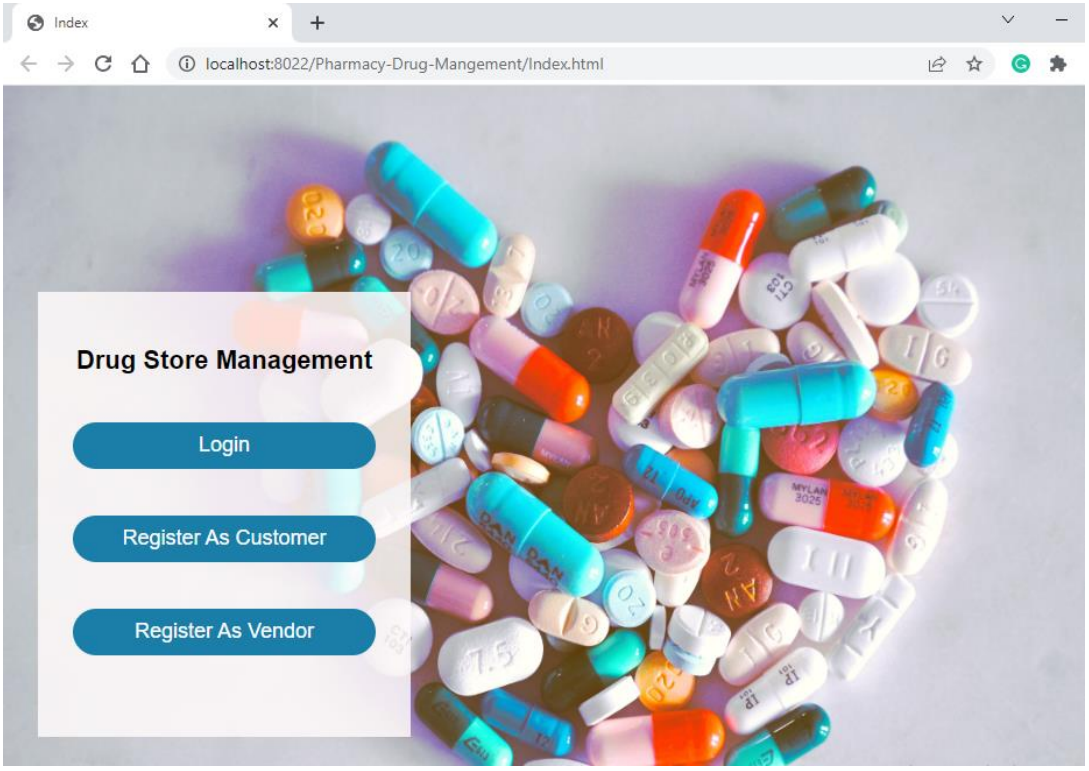
7.4 Functions

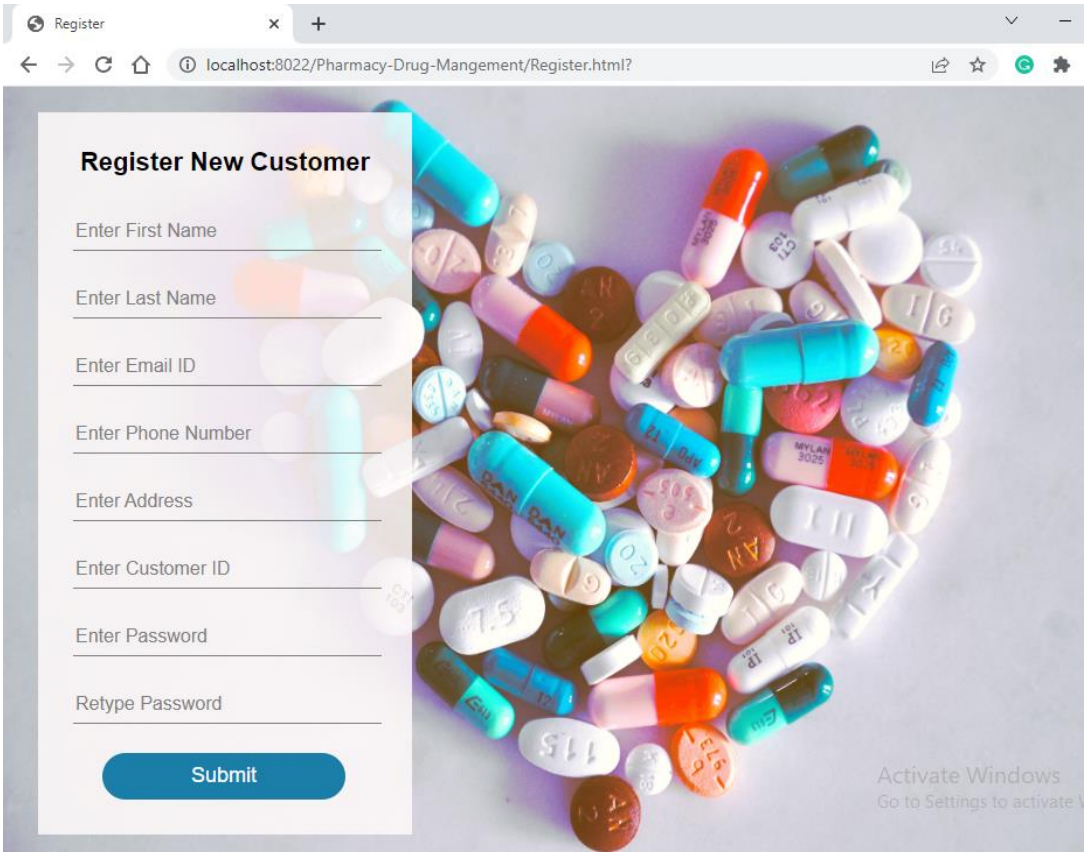
7.5 Connectivity

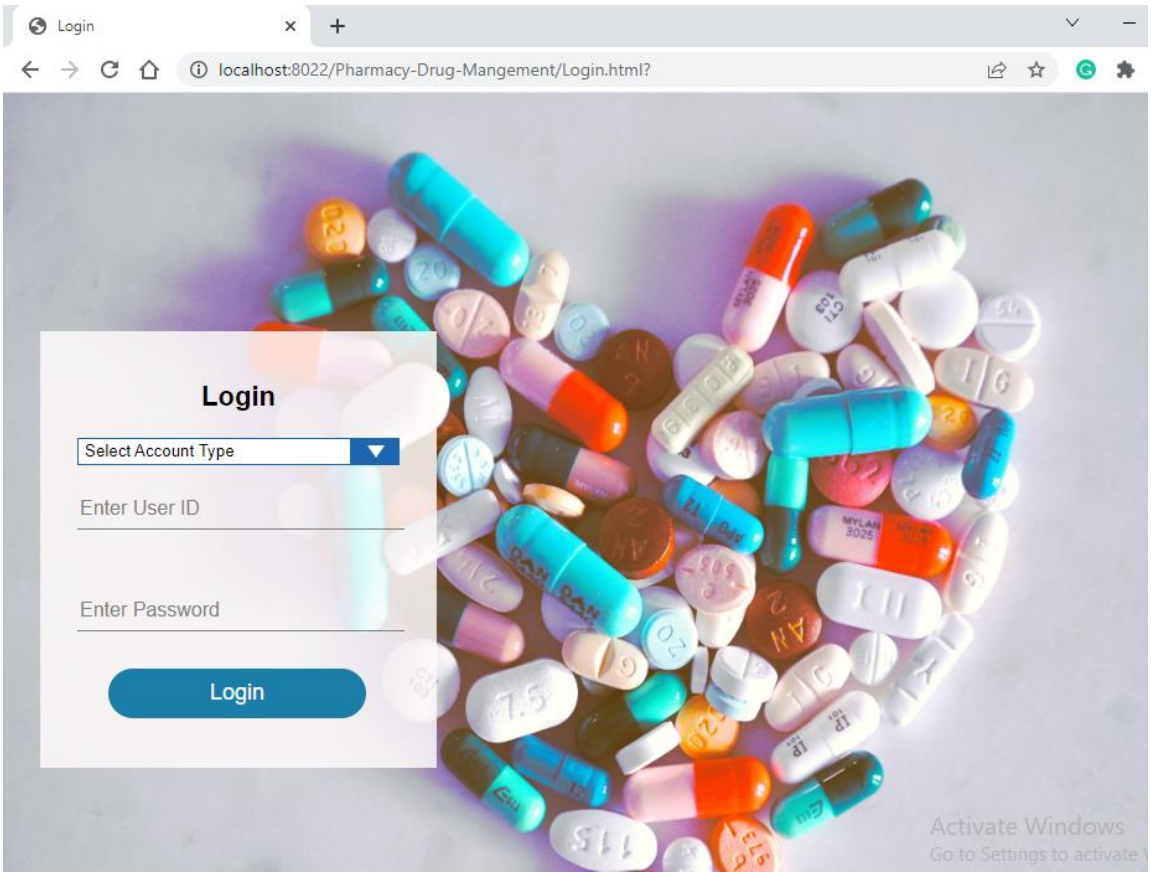
```
ResultSet rs=null;
Connection conn=null;
PreparedStatement ps=null;

String query2="SELECT sid,pass from Seller WHERE sid=?";
String query1="SELECT uid,pass from customer WHERE uid=?";
String query3="SELECT aid,pass from admin WHERE aid=?";

try{
Class.forName("com.mysql.jdbc.Driver");
conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/drugdatabase","root","fast1234");
```

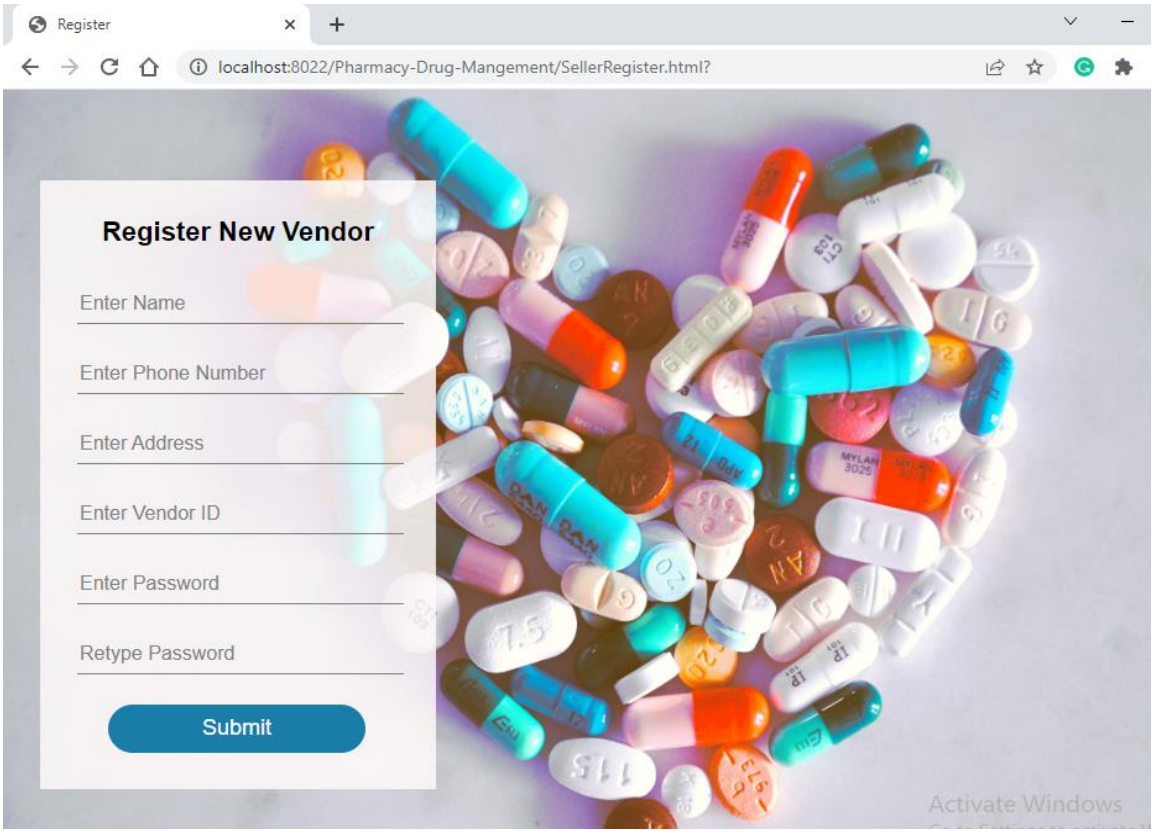
```
ResultSet rs=null;
Connection conn=null;
PreparedStatement ps1=null;
PreparedStatement ps2=null;

String query1="delete from inventory where  pid=?";
String query2="delete from product where   pid=?";

try{
    Class.forName("com.mysql.jdbc.Driver");
    conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/drugdatabase","root","fast1234");
    ps1=conn.prepareStatement(query1);
    ps2=conn.prepareStatement(query2);

    ps1.setString(1,prod);
    int i=ps1.executeUpdate();
    ps2.setString(1,prod);


    int j=ps2.executeUpdate();

    response.sendRedirect("AddInventory.jsp");
}
catch(Exception e)
{
    out.println(e);
}
finally {
    try { if (rs != null) rs.close(); } catch (Exception e) {};
    try { if (ps1 != null) ps1.close(); } catch (Exception e) {};
    try { if (ps2 != null) ps2.close(); } catch (Exception e) {};
    try { if (conn != null) conn.close(); } catch (Exception e) {};
}
```

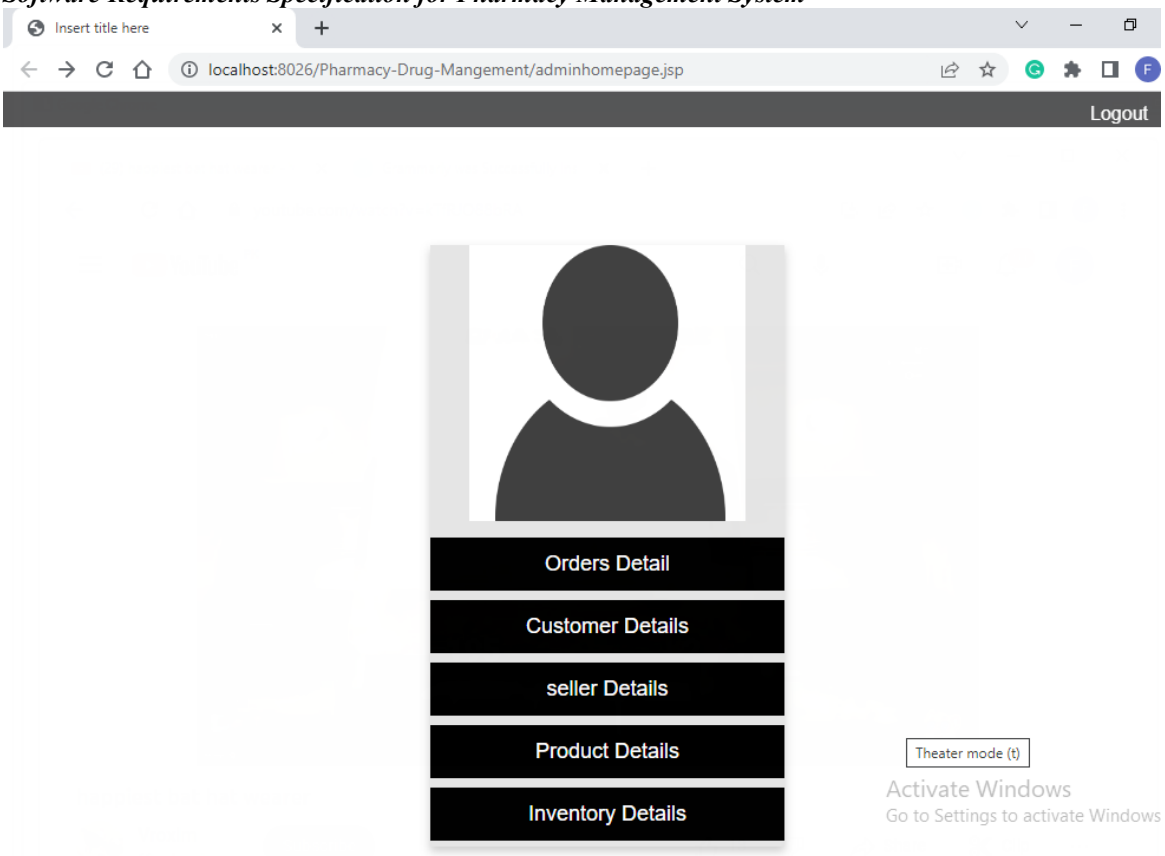
8. Project Screenshot

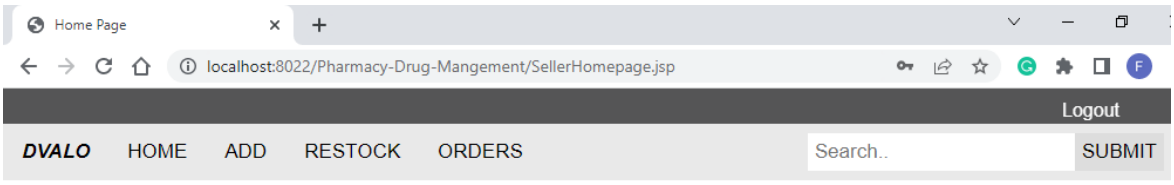


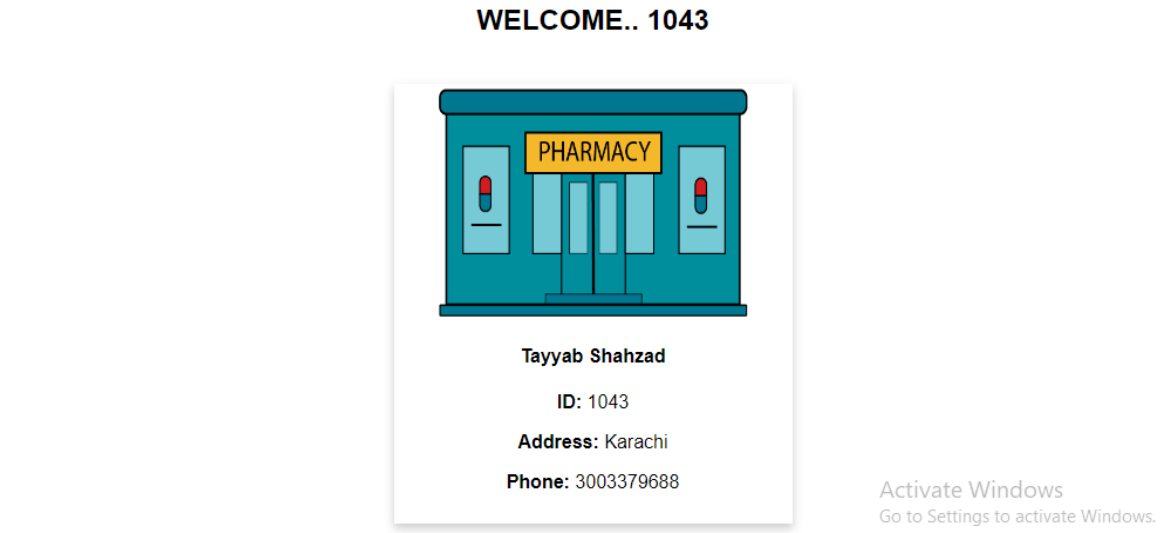


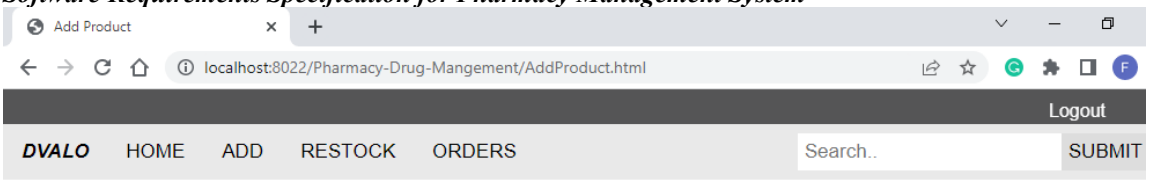

Login
localhost:8022/Pharmacy-Drug-Mangement/Login.html?
Login
Select Account Type
Enter User ID
Enter Password
Login
Activate Windows

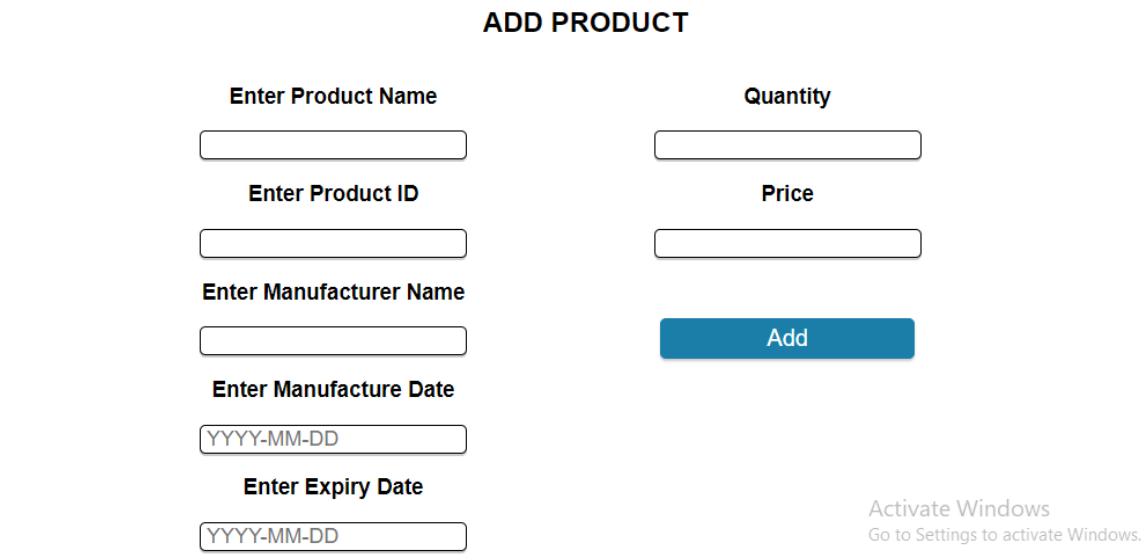

Register
localhost:8022/Pharmacy-Drug-Mangement/SellerRegister.html?
Register New Vendor
Enter Name
Enter Phone Number
Enter Address
Enter Vendor ID
Enter Password
Retype Password
Submit
Activate Windows

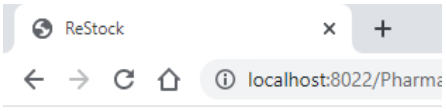

Insert title here
localhost:8026/Pharmacy-Drug-Mangement/adminhomepage.jsp
Logout
Orders Detail
Customer Details
seller Details
Product Details
Inventory Details
Theater mode (t)
Activate Windows
Go to Settings to activate Windows

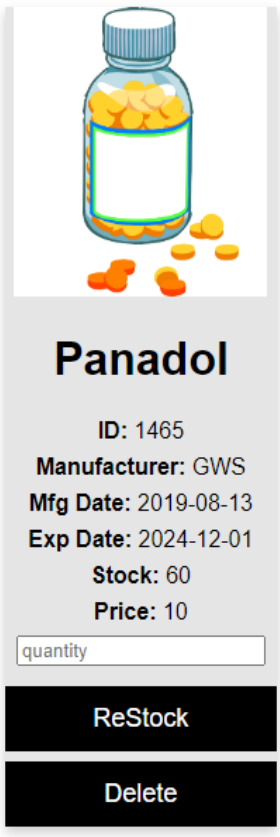

Home Page
localhost:8022/Pharmacy-Drug-Mangement/SellerHomepage.jsp
Logout
DVALO
HOME
ADD
RESTOCK
ORDERS
Search..
SUBMIT

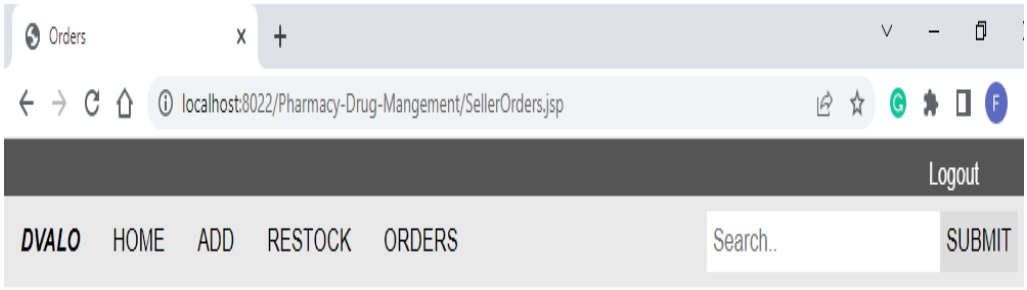

WELCOME.. 1043
PHARMACY
Tayyab Shahzad
ID: 1043
Address: Karachi
Phone: 3003379688
Activate Windows
Go to Settings to activate Windows.



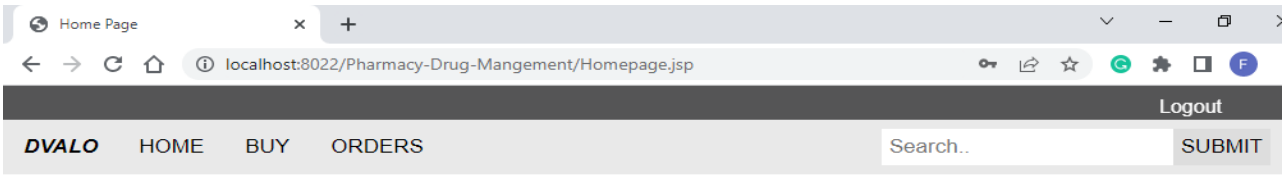




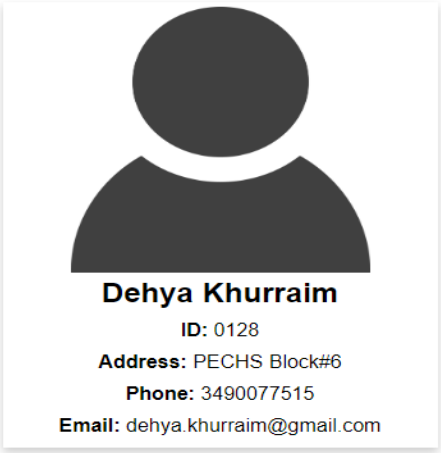


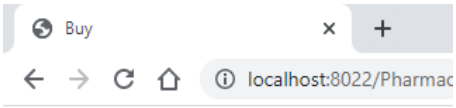


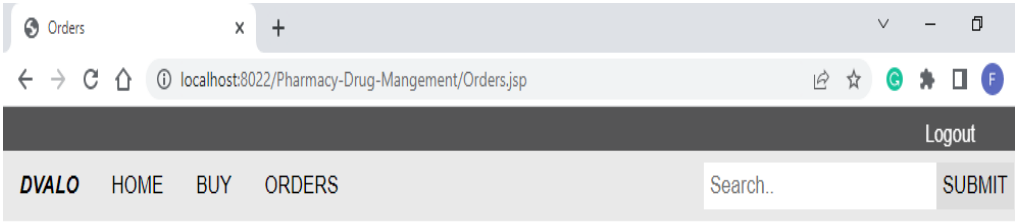
Order ID	Product ID	Price	Quantity	CUSTOMER ID	Order Date and Time
1000	1465	200	20	0128	2022-12-04 17:25:14.0



Welcome 0128







Order ID	Product ID	Price	Quantity	Seller ID	Order Date and Time
1000	1465	200	20	1043	2022-12-04 17:25:14.0