



UTM
UNIVERSITI TEKNOLOGI MALAYSIA

FACULTY OF ENGINEERING
SCHOOL OF COMPUTING
SECJ3303 INTERNET PROGRAMMING
GROUP PROJECT - SECTION 01

Project Report

Project Title: Online Pharmacy System - PharmOnline

Lecturer : Dr Nor Azizah Binti Sa'adon

Prepared by:

Team Net or Trap

NO.	NAME	MATRICES NUMBER
1.	See Wen Xiang	A19EC0206
2.	Cheong Chien Li	A19EC0186
3.	Kong Hao Yang	A19EC0065
4.	Loo Zhi Xuen	A19EC0078

1.0 Introduction

PharmOnline is an online pharmacy system whereby it aims to be used by the pharmacist and the customers. Currently, many pharmacies manage their inventory management manually and the customers need to visit the pharmacies or contact the pharmacy to get their medicine or service needs. Hence, there is a **need** for the pharmacy to own their exclusive online tools to digitalize the inventory management and ordering process. Therefore our **approach** to solve this issue is to create an online pharmacy system whereby the pharmacy is able to record and keep track of the inventory details online. Besides, the customers are also able to choose the medical products and place their order online. The proposed system definitely would **benefit** the pharmacy whereby the pharmacy would be able to handle the inventory management and ordering module with a more efficient and systematic approach. There are some **competitions** to the system we are proposing such as Shopee and Lazada whereby Shopee and Lazada are e-commerce websites that are well-recognized and often used by the community. Therefore, many pharmacies are using Shopee and Lazada services as their platform to build an online store. But, PharmOnline is different compared to both e-commerce websites as our proposed system is mainly built specialized for the pharmacy as their own product. Therefore, the pharmacy has full control to the system instead of relying on external platforms and it leads to more profitable and convenient to the pharmacy.

2.0 Problem Statement

Nowadays, many pharmacies have expanded their businesses online in order to generate more sales beyond the boundaries of their physical store. Online stores have their own records of inventory and products available for sale, which has to be maintained and regularly updated by the administrators in charge of the online store. Besides managing the inventory records of their online store, pharmacies also have an inventory system present in their physical store used to keep track of the availability of their products in-store, which is independent of the inventory records of their online store. Thereby lies an issue where the online store records might not always be exactly aligned with the records in the inventory system of the physical store. Much time and effort has to be spent by the accounting department of the pharmacies to keep track of their inventories properly, and ensure that the two separate inventory records are always aligned with each other. The process of aligning

and keeping track of both of the inventory records can be extremely tedious and might end up costing the pharmacies many precious working hours.

Hence, the system which we are proposing, PharmOnline aims to solve this issue by providing a single centralised inventory system that has been integrated with an online store platform. PharmOnline shall allow pharmacies to only have a single inventory system to manage both their physical store as well as online store inventory records, which will be able to greatly reduce the effort required by the accounting department of the pharmacies to keep track of their inventories.

In short, PharmOnline is a pharmacy inventory system that is integrated with an online store platform. Pharmacies shall be able to utilize a centralized inventory system to keep track of their inventories for both their physical and online stores. PharmOnline shall be able to greatly reduce the effort required by pharmacies to manage their inventories, as well as help them to save potentially hundreds of precious working hours.

3.0 Module Functionality and Actor



Figure 3.1 : Use Case Diagram for Online Pharmacy System (Pharm Online)

Table 3.1 : List of modules for Online Pharmacy System (Pharm Online)

Module	Actor	Functionality
Module 1: Login	<ul style="list-style-type: none"> · Customer · Pharmacist 	<ul style="list-style-type: none"> · Authenticate users with username and password
Module 2: Register	<ul style="list-style-type: none"> · Customer · Pharmacist 	<ul style="list-style-type: none"> · Register new user account by filling in the required information of user
Module 3: Manage Profile	<ul style="list-style-type: none"> · Customer · Pharmacist 	<ul style="list-style-type: none"> · Edit the profile of users including the personal information and password of users and also involve the deletion of user account
Module 4: View Product	<ul style="list-style-type: none"> · Customer 	<ul style="list-style-type: none"> · View products available for sale
Module 5: Place Order	<ul style="list-style-type: none"> · Customer 	<ul style="list-style-type: none"> · Ability to place orders for medicine that will be paid and collected physically at the pharmacy.

Module 6: Manage order	<ul style="list-style-type: none"> · Customer · Pharmacist 	<p><u>Customer View</u></p> <ul style="list-style-type: none"> · Perform operations such as modifying orders or cancelling orders to manage orders that have been placed. · Accept customer's orders and update the status of order to "prepared" once the orders are prepared and ready to be collected by customers. · Reject orders. · Modify the items inside the orders. · Complete an order after customers made a payment for the order and collected the items of the order. · Print bills for the completed orders.
Module 7: Manage Product	<ul style="list-style-type: none"> · Pharmacist 	<p><u>Pharmacist View</u></p> <ul style="list-style-type: none"> · Perform CRUD operations to medicine items · Categorize each medicine · Set selling price for each medicine
Module 8: Manage Category	<ul style="list-style-type: none"> · Pharmacist 	<ul style="list-style-type: none"> · Perform CRUD operations to the category of medicine

Module 9: Manage Stock	· Pharmacist	· Perform the operations of adding stock and returning stock · Edit and delete the transactions of stock
Module 10: Manage suppliers	· Pharmacist	· Perform CRUD operations to suppliers
Module 11: Generate report	· Pharmacist	· Generate sales report and stock report

4.0 Flow Of Project

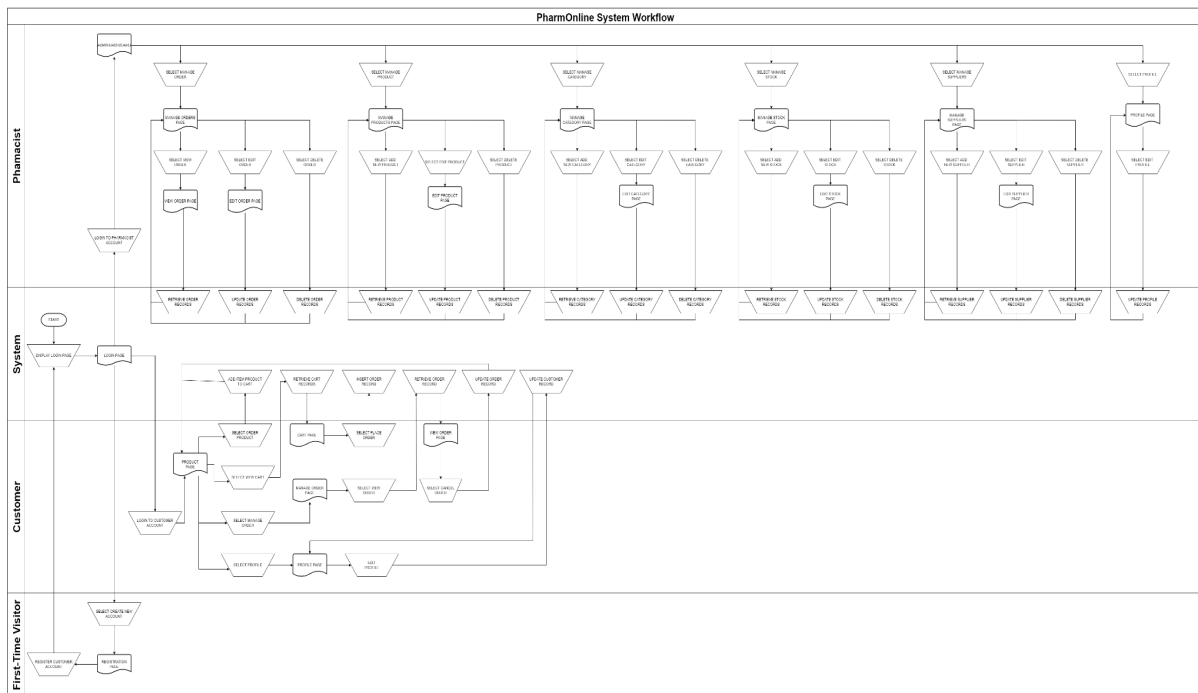


Figure 4.1: Swimlane Diagram of PharmOnline System

5.0 Implementation Of Elements

i. Register

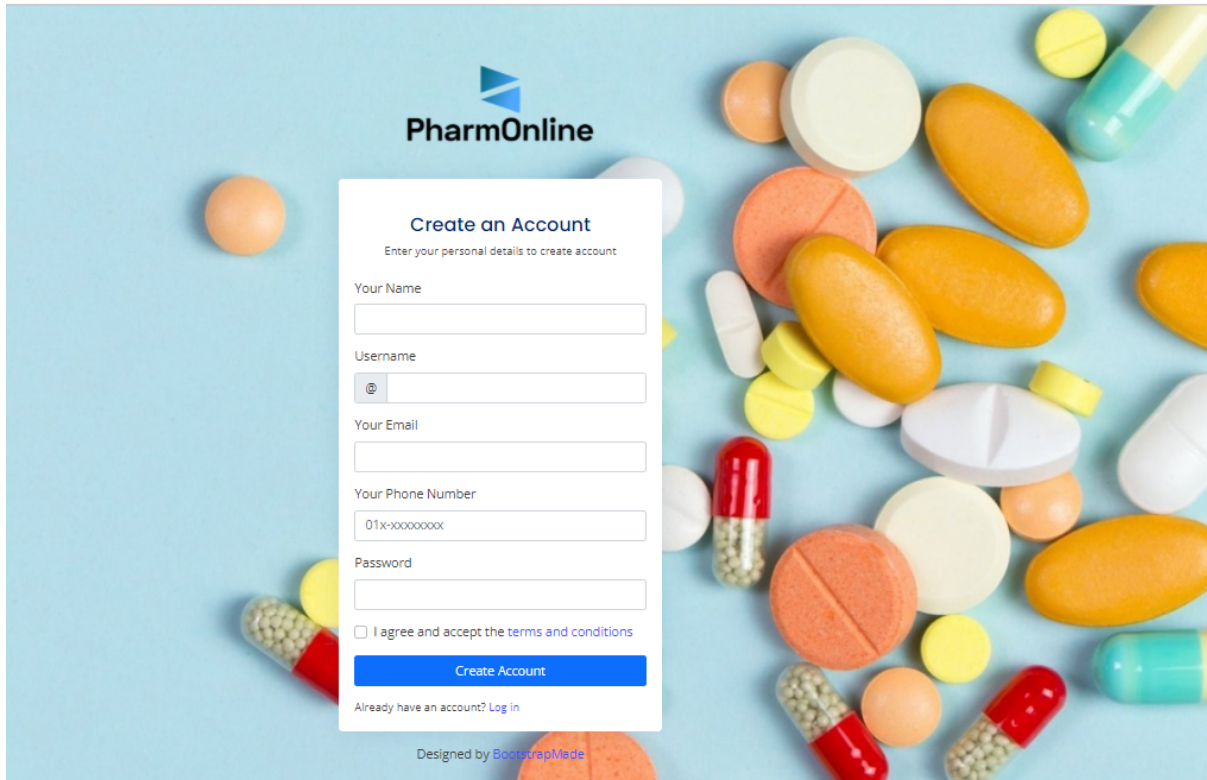


Figure 5.1: Screenshot of registration page

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException, ClassNotFoundException, SQLException {
    response.setContentType("text/html;charset=UTF-8");

    PrintWriter out = response.getWriter();

    String fullname = request.getParameter("fullname");
    String email = request.getParameter("email");
    String username = request.getParameter("username");
    String password = request.getParameter("password");
    String phoneNum = request.getParameter("phoneNum");

    String driver = "com.mysql.jdbc.Driver";
    String dbName = "pharmonline";
    String url = "jdbc:mysql://localhost/" + dbName + "?";
    String userName = "root";
    String pw = "";
    Class.forName(driver); //Load Driver
    Connection con = DriverManager.getConnection(url, userName, pw); // Set Connection

    String query = "SELECT * FROM user WHERE username='" + username + "'";

    Statement st = con.createStatement(); // create query
    ResultSet rs = st.executeQuery(query); // Execute query

    if(rs.next()){
        out.println("<script type='text/javascript'>");
        out.println("alert('The username is already taken.');
```

Figure 5.2: Screenshot 1 of code in Regitser.java

```
else{
    query = "SELECT * FROM user WHERE userType=\"Customer\"";

    st = con.createStatement();
    rs = st.executeQuery(query);

    int size=0;

    if(rs != null){
        rs.last();
        size = rs.getRow();
    }

    st.close();
    con.close();

    String userID = "C" + String.valueOf(size + 1);

    User newUser = new User();

    newUser.setUserID(userID);
    newUser.setFullname(fullname);
    newUser.setEmail(email);
    newUser.setUsername(username);
    newUser.setPassword(password);
    newUser.setPhoneNum(phoneNum);
    newUser.setUserType("Customer");

    newUser.insertIntoDB();

    out.println("<script type=\"text/javascript\">");
    out.println("alert('Registration Successful!');");
    out.println("location='login.jsp'");
    out.println("</script>");
}
```

Figure 5.3: Screenshot 2 of code in Regitser.java

```
public void insertIntoDB() throws ClassNotFoundException, SQLException{

    String driver = "com.mysql.jdbc.Driver";
    String dbName = "pharmonline";
    String url = "jdbc:mysql://localhost/" + dbName + "?";
    String userName = "root";
    String pw = "";
    String query = "INSERT INTO `pharmonline`.`user`(`username`, `fullname`, `phoneNum`, `email`, `userType`, `userID`, `password`)

    Class.forName(driver); //Load Driver
    Connection con = DriverManager.getConnection(url, userName, pw); // Set Connection
    Statement st = con.createStatement(); // create query
    st.executeUpdate(query); // Execute query

    st.close();
    con.close();
}
```

Figure 5.4: Screenshot 1 of insertIntoDB method in User.java

ii. Session/Page Validation

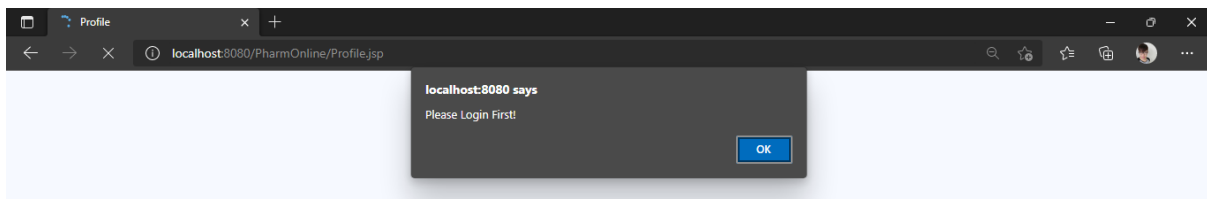


Figure 5.5: Screenshot of interface for session validation

```
<%  
    if(request.getMethod().equals("GET") && request.getSession().getAttribute("loggedIn") == null){  
        // String redirectURL = "login.jsp";  
        // response.sendRedirect(redirectURL);  
  
        out.println("<script type='text/javascript'>");  
        out.println("alert('Please Login First!');");  
        out.println("location='login.jsp';");  
        out.println("</script>");  
    }  
%>
```

Figure 5.6: Screenshot of code for session validation

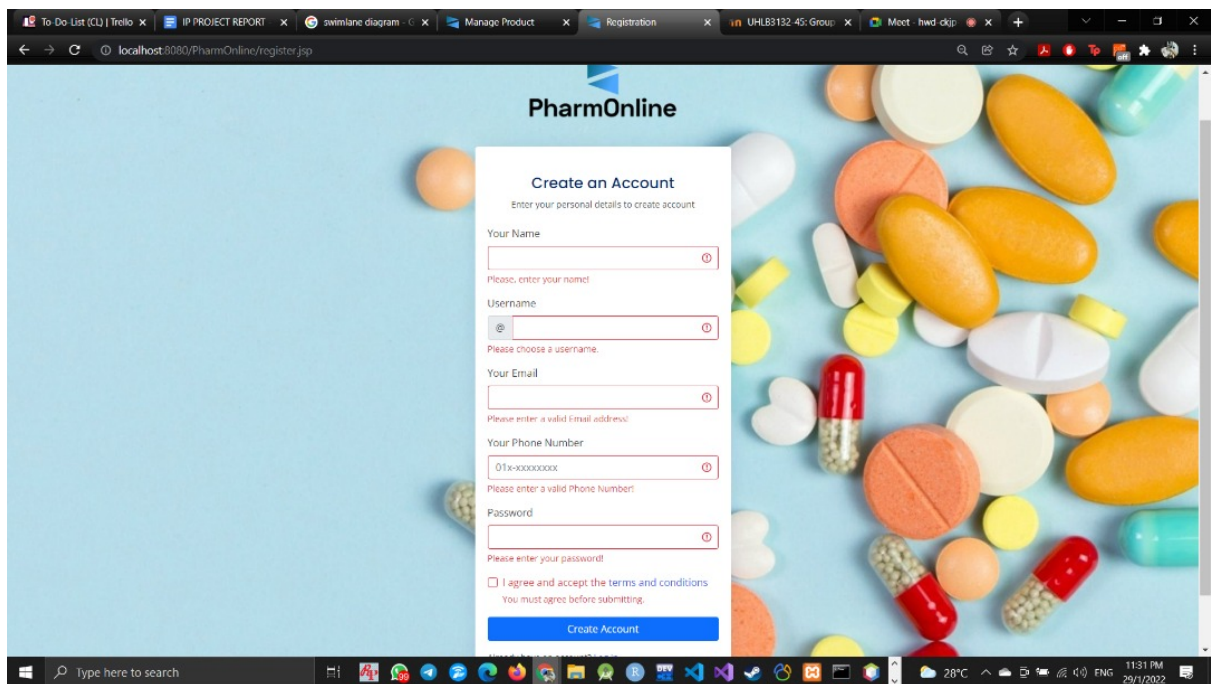


Figure 5.7: Screenshot of interface for page validation

```
<div class="col-12">  
    <label for="phoneNum" class="form-label">Your Phone Number</label>  
    <input type="tel" name="phoneNum" class="form-control" id="phoneNum" placeholder="01x-xxxxxxx" pattern="01[0-9]{1}-[0-9]{7,8}" required />  
    <div class="invalid-feedback">Please enter a valid Phone Number!</div>  
</div>
```

Figure 5.8: Screenshot of code for page validation

iii. View/Edit Profile

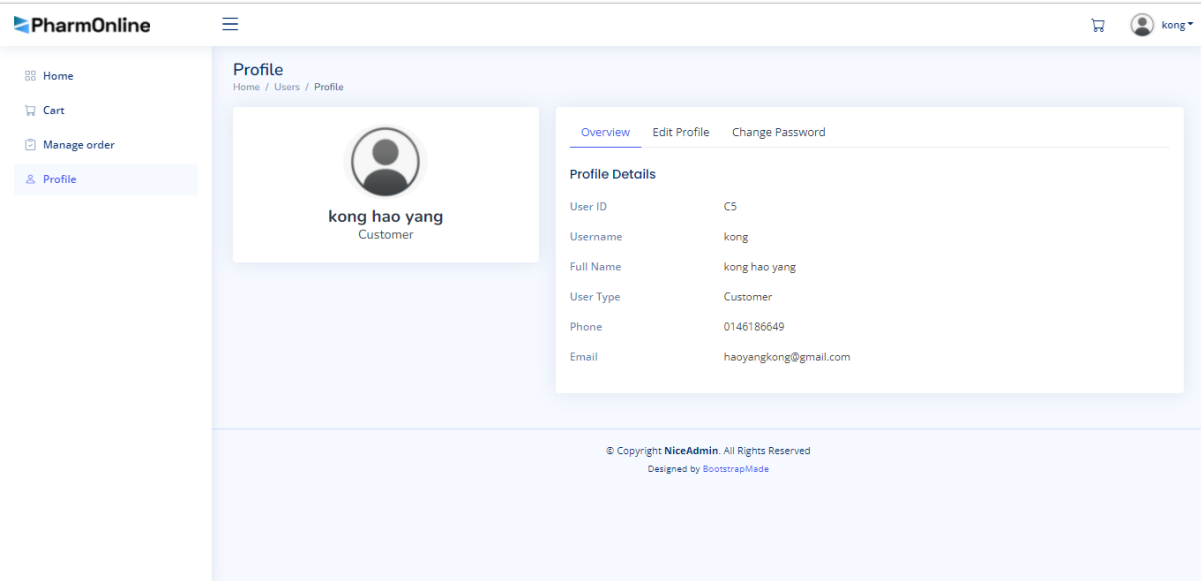


Figure 5.9: Screenshot of interface for view profile

```
<div class="row">
  <div class="col-lg-3 col-md-4 label">User ID</div>
  <div class="col-lg-9 col-md-8"><jsp:getProperty name="user" property="userID"/></div>
</div>

<div class="row">
  <div class="col-lg-3 col-md-4 label">Username</div>
  <div class="col-lg-9 col-md-8"><jsp:getProperty name="user" property="username"/></div>
</div>

<div class="row">
  <div class="col-lg-3 col-md-4 label">Full Name</div>
  <div class="col-lg-9 col-md-8"><jsp:getProperty name="user" property="fullname"/></div>
</div>

<div class="row">
  <div class="col-lg-3 col-md-4 label">User Type</div>
  <div class="col-lg-9 col-md-8"><jsp:getProperty name="user" property="userType"/></div>
</div>

<div class="row">
  <div class="col-lg-3 col-md-4 label">Phone</div>
  <div class="col-lg-9 col-md-8"><jsp:getProperty name="user" property="phoneNum"/></div>
</div>

<div class="row">
  <div class="col-lg-3 col-md-4 label">Email</div>
  <div class="col-lg-9 col-md-8"><jsp:getProperty name="user" property="email"/></div>
</div>
```

Figure 5.10: Screenshot of code for view profile

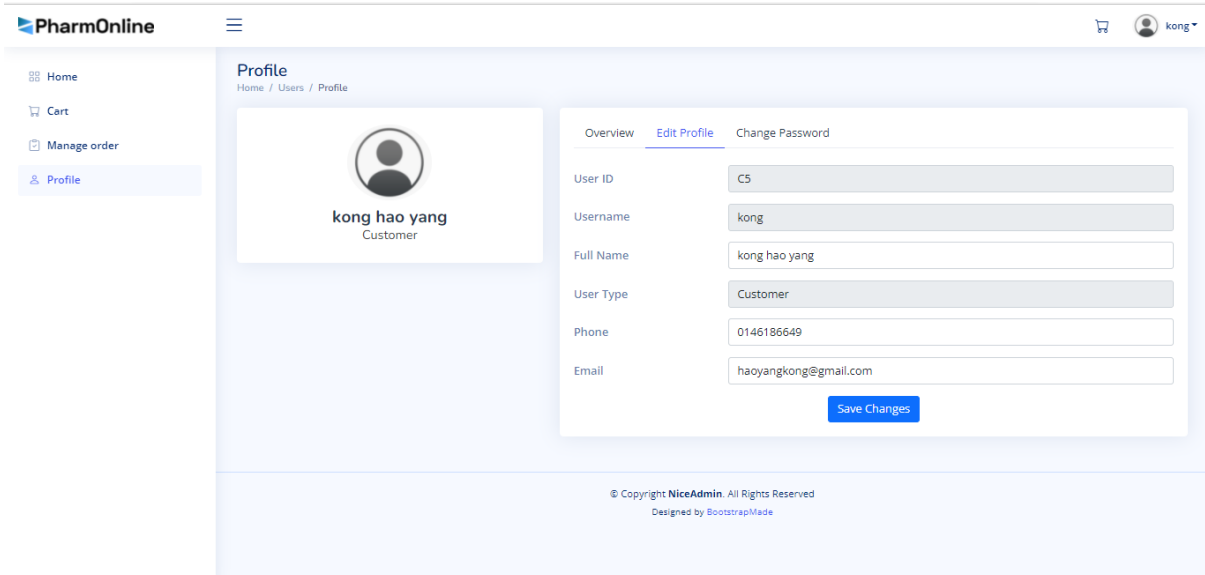


Figure 5.11: Screenshot of interface for edit profile

```

public void updateInDB() throws ClassNotFoundException, SQLException{
    String driver = "com.mysql.jdbc.Driver";
    String dbName = "pharmonline";
    String url = "jdbc:mysql://localhost/" + dbName + "?";
    String userName = "root";
    String pw = "";
    String query = "UPDATE `pharmonline`.`user` SET `fullname` = '"+fullname+"', `phoneNum` = '"+phoneNum+"', `email` = '"+email+"' WHERE `username` = C";

    Class.forName(driver); //Load Driver
    Connection con = DriverManager.getConnection(url, userName, pw); // Set Connection
    Statement st = con.createStatement(); // create query
    st.executeUpdate(query); // Execute query

    st.close();
    con.close();
}

```

Figure 5.12: Screenshot of code for edit profile

iv. Consistent UI

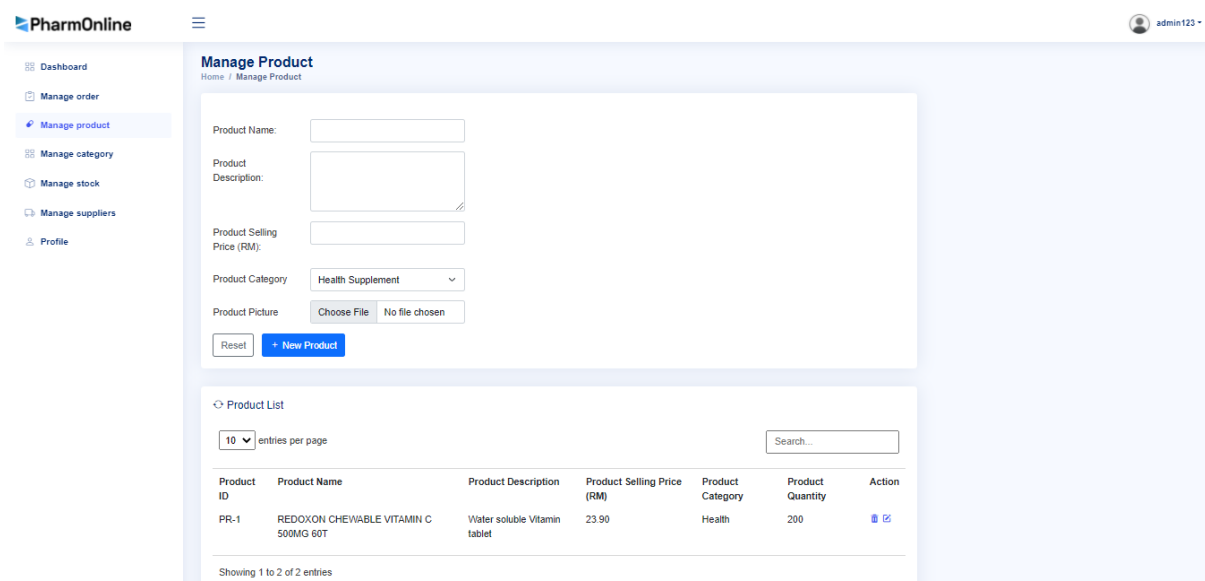


Figure 5.13: Screenshot of interface for manage product

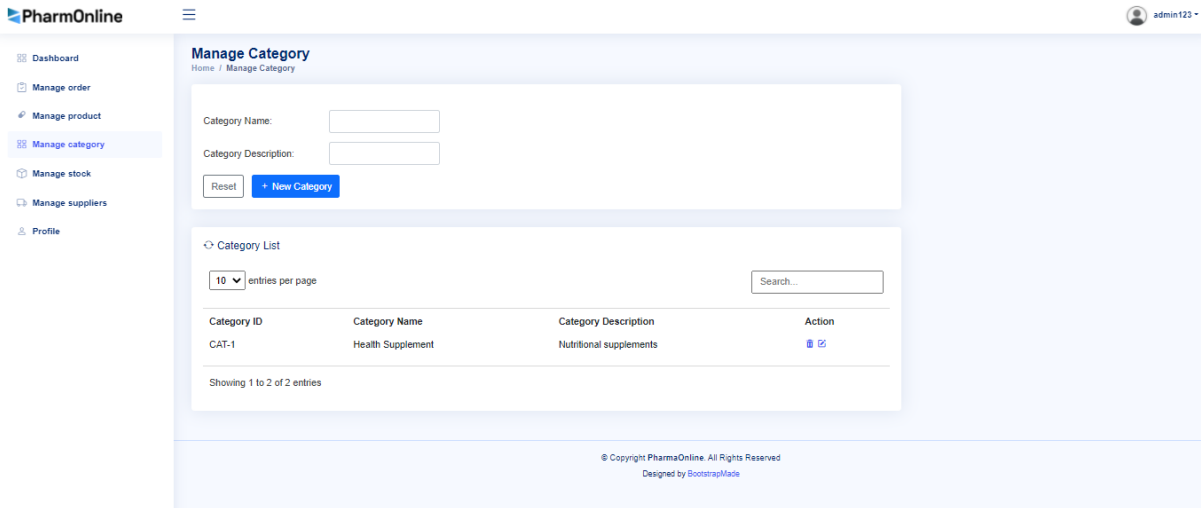


Figure 5.14: Screenshot of interface for manage category

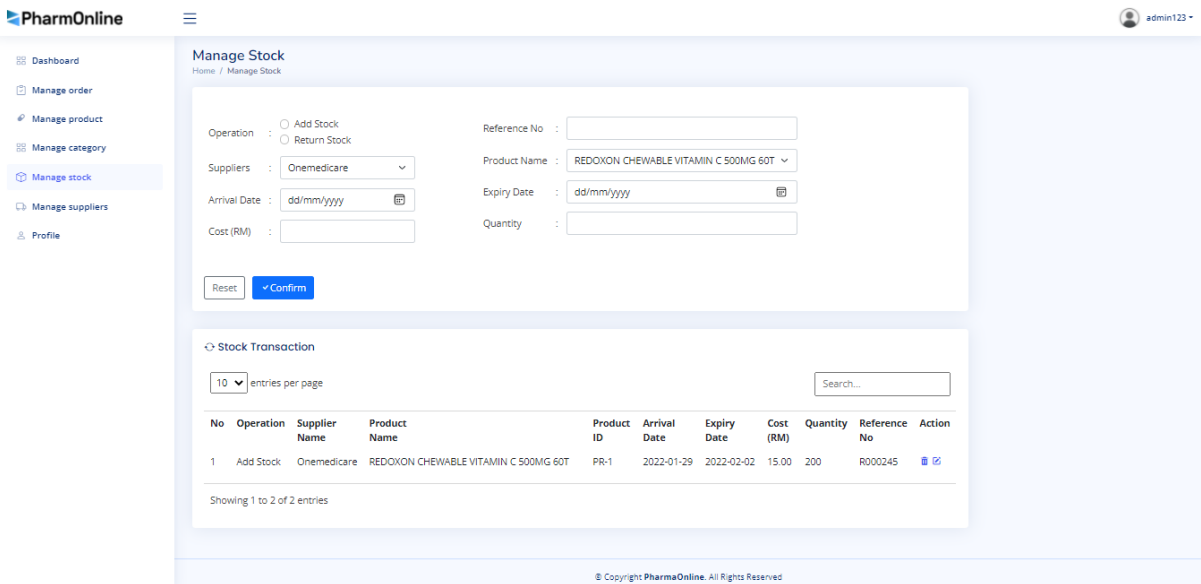


Figure 5.15: Screenshot of interface for manage stock

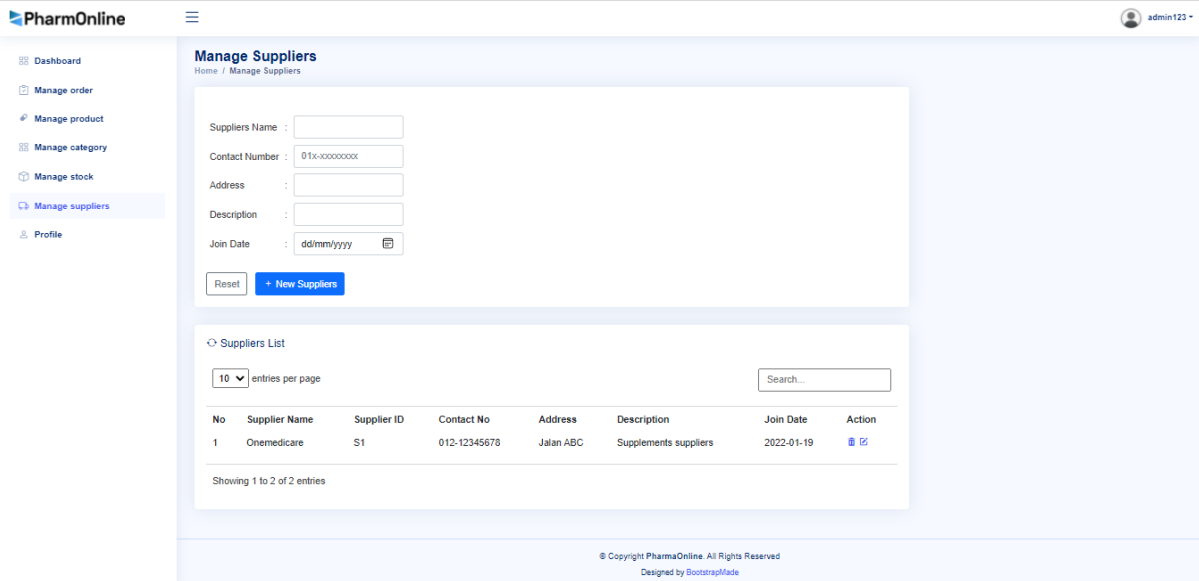


Figure 5.16: Screenshot of interface for manage suppliers

```

<!-- Sidebar -->
<aside id="sidebar" class="sidebar">

<ul class="sidebar-nav" id="sidebar-nav">

<li class="nav-item">
<a class="nav-link collapsed" href="indexAdmin.jsp">
<i class="bi bi-grid"></i>
<span>Dashboard</span>
</a>
</li>!-- End Dashboard Nav -->
<li class="nav-item">
<a class="nav-link collapsed" href="ManageOrder.jsp">
<i class="bi bi-clipboard-check"></i>
<span>Manage order</span>
</a>
</li>!-- End Profile Page Nav -->
<li class="nav-item">
<a class="nav-link" href="Manage product.jsp">
<i class="bx bxs-capsule"></i>
<span>Manage product</span>
</a>
</li>!-- End Profile Page Nav -->
<li class="nav-item">
<a class="nav-link collapsed" href="Manage category.jsp">
<i class="bi bi-grid"></i>
<span>Manage category</span>
</a>
</li>!-- End Profile Page Nav -->
<li class="nav-item">
<a class="nav-link collapsed" href="Manage stock.jsp">
<i class="bi bi-box"></i>
<span>Manage stock</span>
</a>
</li>!-- End Profile Page Nav -->

<li class="nav-item">
<a class="nav-link collapsed" href="Manage suppliers.jsp">
<i class="bi bi-truck"></i>
<span>Manage suppliers</span>
</a>
</li>!-- End Profile Page Nav -->
<li class="nav-item">
<a class="nav-link collapsed" href="Profile.jsp">
<i class="bi bi-person"></i>
<span>Profile</span>
</a>
</li>!-- End Profile Page Nav -->

</ul>

</aside>!-- End Sidebar-->

```

Figure 5.17: Screenshot of code for sidebar showing consistent UI

v. Data Management - CRUD +Activation

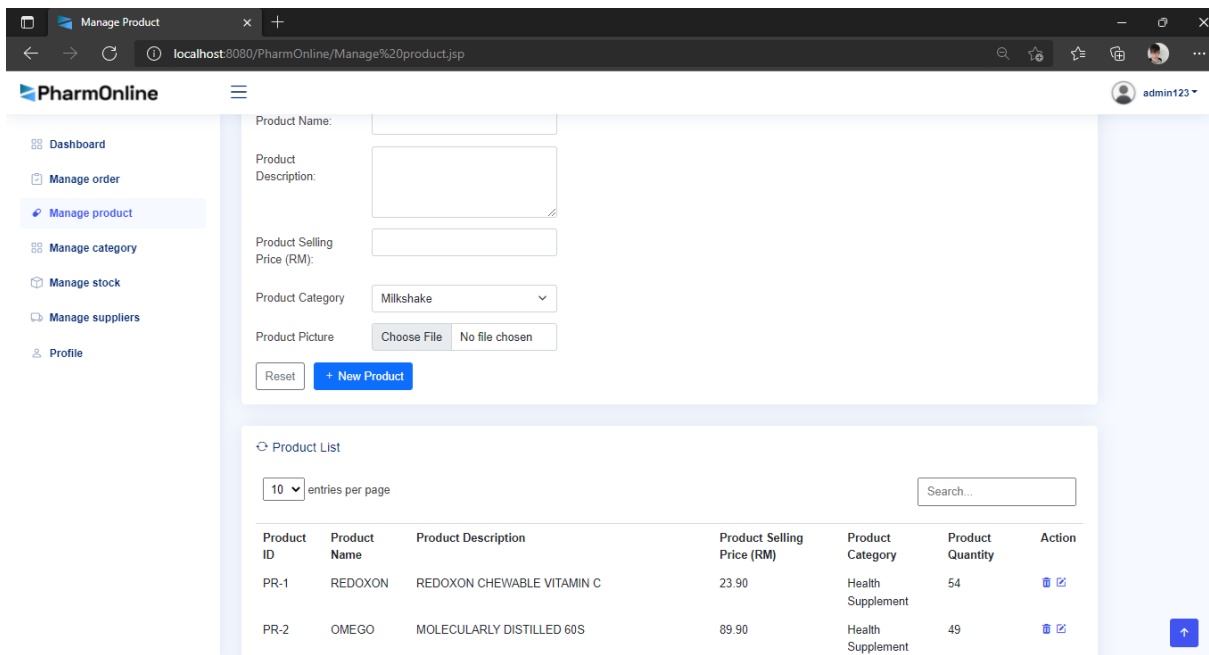


Figure 5.18: Screenshot of interface for manage product

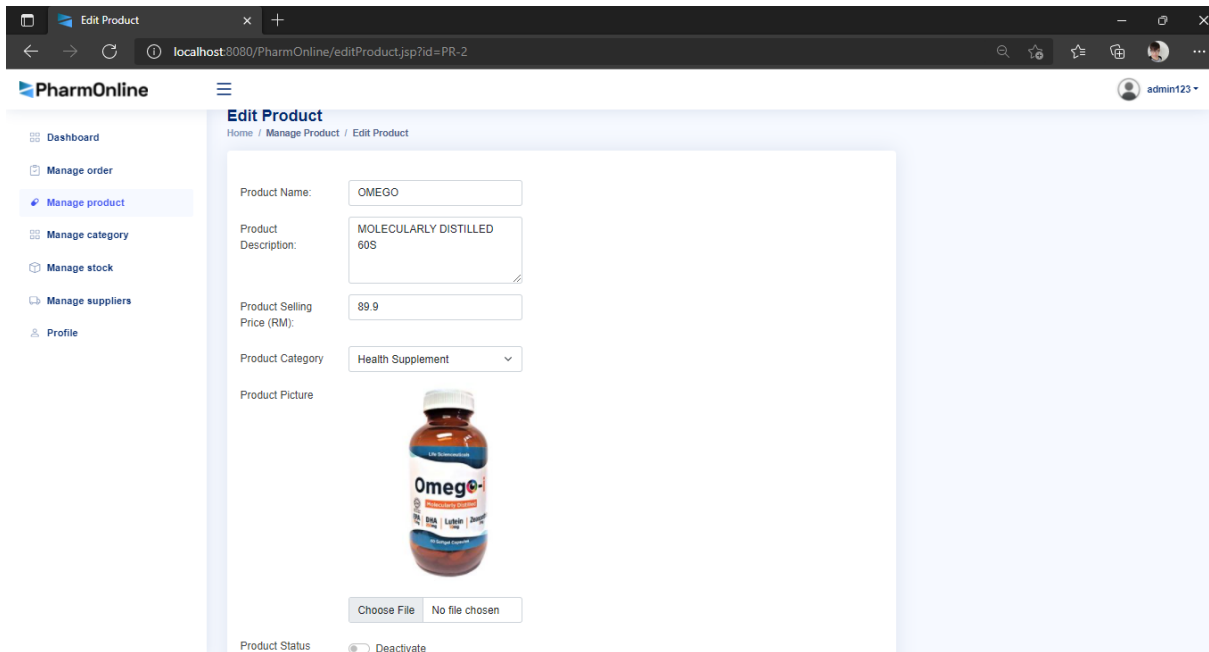


Figure 5.19: Screenshot of interface for edit product + activation

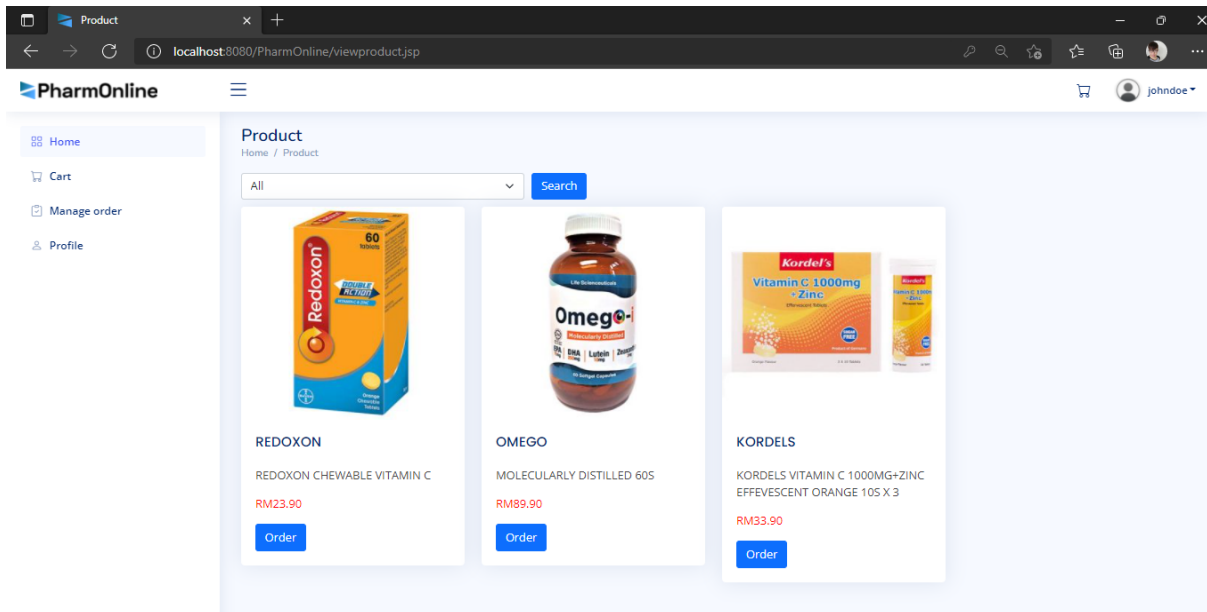


Figure 5.19: Screenshot of interface for view product

```

public void insertIntoDB() throws ClassNotFoundException, SQLException{
    String driver = "com.mysql.jdbc.Driver";
    String dbName = "pharmonline";
    String url = "jdbc:mysql://localhost/" + dbName + "?";
    String userName = "root";
    String pw = "";
    String query = "INSERT INTO 'pharmonline'. 'product' ('product_ID', 'product_Name', 'product_Description', 'product_Selling_Price', 'product_Category', 'product_Quantity', 'product_Picture', 'product_Status') VALUES
    Class.forName(driver); //Load Driver
    Connection con = DriverManager.getConnection(url, userName, pw); // Set Connection
    PreparedStatement st=con.prepareStatement(query);
    st.setString(1, this.ID);
    st.setString(2, this.name);
    st.setString(3, this.description);
    st.setDouble(4, this.price);
    st.setString(5, this.category);
    st.setInt(6, this.quantity);
    st.setBlob(7, this.picture);
    st.setString(8, this.status);
    st.executeUpdate();
    st.close();
    con.close();
}

public void updateInDB() throws ClassNotFoundException, SQLException{
    String driver = "com.mysql.jdbc.Driver";
    String dbName = "pharmonline";
    String url = "jdbc:mysql://localhost/" + dbName + "?";
    String userName = "root";
    String pw = "";
    String query = "UPDATE 'pharmonline'. 'product' SET 'product_Name' = '"+name+"', 'product_Description' = '"+description+"', 'product_Selling_Price' = '"+price+"', 'product_Category' = '"+category+'', 'product_Statu
    Class.forName(driver); //Load Driver
    Connection con = DriverManager.getConnection(url, userName, pw); // Set Connection
    Statement st = con.createStatement(); // create query
    st.executeUpdate(query); // Execute query

    st.close();
    con.close();
}

```

Figure 5.20: Screenshot of code for product CRUD

```

<<
String query = "";
if (abc.equals(""))
    query = "SELECT * FROM product WHERE product_Status='Activate' AND product_Quantity>0";
else{
    query="SELECT * FROM product WHERE product_Status='Activate' AND product_Category='"+abc+"' AND product_Quantity>0";
}

int i = 1;
Class.forName(driver);
Connection con=DriverManager.getConnection(url,userName,password);
Statement st=con.createStatement();
ResultSet rs=st.executeQuery(query);

int counter=0;
while (rs.next())
{
    counter++;
}
>>

```

Figure 5.21: Screenshot of code for activation

vi. User Transaction + history

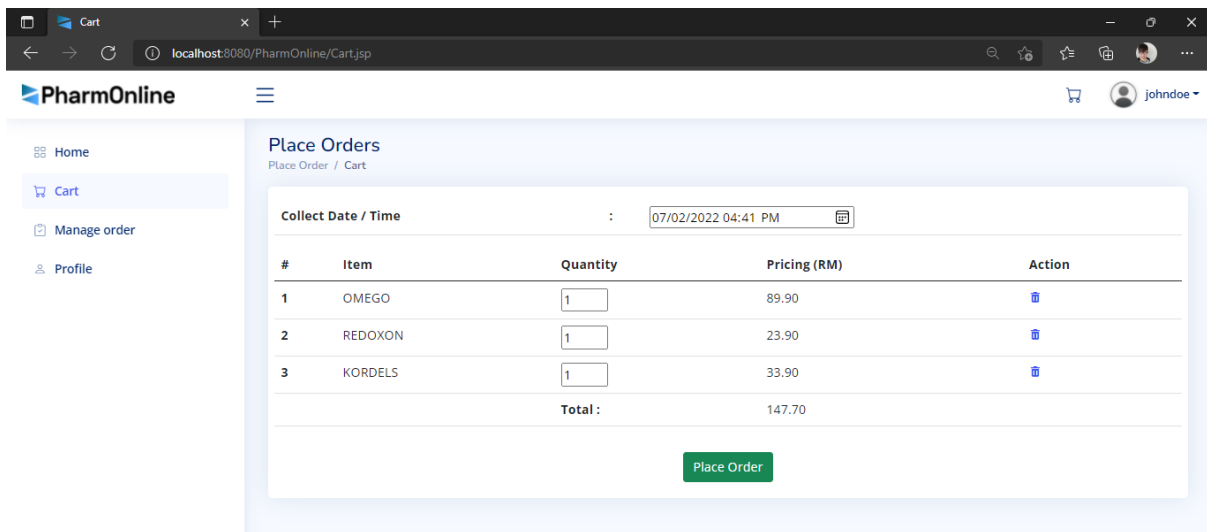


Figure 5.22: Screenshot of interface for Place Order

```
public void addOrderToDB() throws SQLException, ClassNotFoundException{
    String driver = "com.mysql.jdbc.Driver";
    String dbName="PharmOnline";
    String url="jdbc:mysql://localhost/"+dbName+"?";
    String userName="root";
    String password="";
    String query = "SELECT count(*) FROM `order`";
    String querycheck = "SELECT * FROM `order`";
    String queryx="INSERT INTO `order` (`orderId`, `orderDateTime`, `collectDateTime`, `totalprice`, `status`, `customerID`) VALUES (?, ?, ?, ?, ?, ?)";
    Class.forName(driver);
    Connection con=DriverManager.getConnection(url,userName,password);
    Statement st=con.createStatement();
    Statement stcheck=con.createStatement();
    PreparedStatement statement=con.prepareStatement(queryx);
    ResultSet rs=st.executeQuery(query);
    ResultSet rsCheck=stcheck.executeQuery(querycheck);
    rs.next();
    int count=rs.getInt(1);
    count++;
    String ID="OR"+String.valueOf(count);
    while (rsCheck.next())
    {
        String []temp=rsCheck.getString(1).split("OR");
        int number=Integer.parseInt(temp[1]);
        if (count<=number)
        {
            count=number+1;
        }
    }
    orderID="OR"+String.valueOf(count);
}
```

Figure 5.23: Screenshot of code addOrderToDB method to Place Order in Order.java

```

protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException, SQLException, ClassNotFoundException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {
        String userID=request.getParameter("userID");
        double totalPrice = Double.parseDouble(request.getParameter("totalprice"));
        LocalDateTime collectDateTime1 = LocalDateTime.parse(request.getParameter("collectDateTime"));
        Timestamp collectDateTime = Timestamp.valueOf(collectDateTime1);
        Order addOrder = new Order();
        addOrder.setTotalprice(totalPrice);
        addOrder.setCollectDateTime(collectDateTime);
        addOrder.setCustomerID(userID);
        addOrder.addToDB();
        String id = addOrder.getOrderID();
        int itemNo = Integer.parseInt(request.getParameter("rowNo"));
        for(int i=1;i<=itemNo;i++){
            int quantity = Integer.parseInt(request.getParameter("quantity"+i));
            String productID = request.getParameter("selectedproductID"+i);

            OrderProduct addOrderProduct = new OrderProduct();
            addOrderProduct.setOrderID(id);
            addOrderProduct.setProductID(productID);
            addOrderProduct.setQuantity(quantity);
            addOrderProduct.addToDB();
        }
    }
}

```

Figure 5.24: Screenshot of code Place Order in placeOrder.java

The screenshot shows the 'Manage Orders' interface for a customer. The page title is 'PharmOnline' and the user is logged in as 'Johndoe'. The main content area is titled 'Manage Orders' and shows a table of orders. The table has columns for Order ID, Order Date / Time, Collect Date / Time, Amount, Status, and Action. There are three orders listed:

Order ID	Order Date / Time	Collect Date / Time	Amount	Status	Action
OR1	29/01/2022 04:44:13 PM	07/02/2022 04:41:00 PM	RM147.70	Completed	View
OR2	29/01/2022 04:44:24 PM	03/02/2022 04:44:00 PM	RM179.80	Cancelled	View
OR3	29/01/2022 04:45:51 PM	04/02/2022 04:00:00 PM	RM67.80	Accepted	View

The interface also includes a sidebar with navigation options: Home, Cart, Manage order, and Profile. The top navigation bar shows the PharmOnline logo and the user profile 'Johndoe'. The footer of the page contains the copyright notice: © Copyright NiceAdmin. All Rights Reserved.

Figure 5.25: Screenshot of interface Manage Order for Customer View

```
ManageOrder_C.jsp x
Source History
355 out.println("</thead>");
356
357
358 while(rs1.next()){
359     String orderDateLine=rs1.getString(2).toString();
360     String collectDateLine=rs1.getString(3).toString();
361     DateFormat df = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
362     DateFormat outputformat = new SimpleDateFormat("dd/MM/yyyy hh:mm:ss aa");
363     Date orderdate = null;
364     String orderDateoutput = null;
365     Date collectdate = null;
366     String collectDateoutput = null;
367     orderdate= df.parse(orderDateLine);
368     orderDateoutput = outputformat.format(orderdate);
369     collectdate= df.parse(collectDateLine);
370     collectDateoutput = outputformat.format(collectdate);
371     if(rs1.getString(5).equals("Pending")){
372         out.println("<tr>");
373         out.println("<td>"+rs1.getString(1)+"</td>");
374         out.println("<td>"+orderDateoutput+"</td>");
375         out.println("<td>"+collectDateoutput+"</td>");
376         out.println("<td> RM"+rs1.getString(4).toString()+"</td>");
377
378         out.println("<td><span class='\"+rs1.getString(5)+"</span></td>");
379         out.println("<td><a href='\"+rs1.getString(1)+"\"><button type='\"+rs1.getString(1)+"\"></a></td>");
380
381     }
381 }
```

Figure 5.26: Screenshot of Code Manage Order for Customer View in ManageOrder_C.jsp

vii. Admin Manage User Transaction

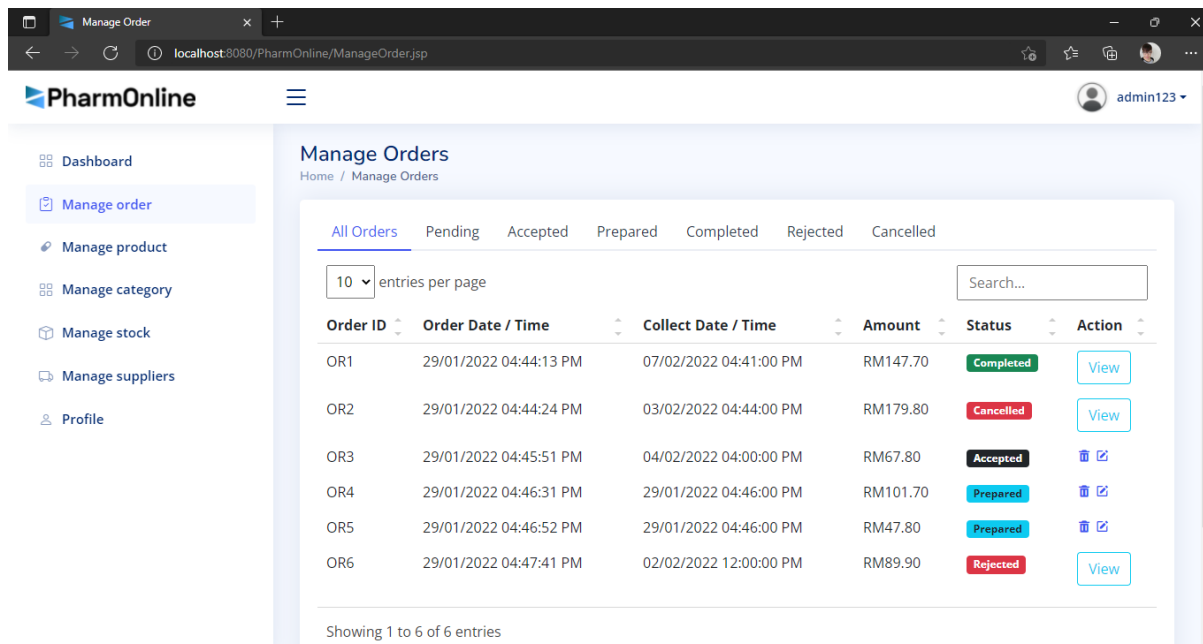


Figure 5.27: Screenshot of interface for Manage Order



Figure 5.28: Screenshot of code Manage Order in ManageOrder.jsp

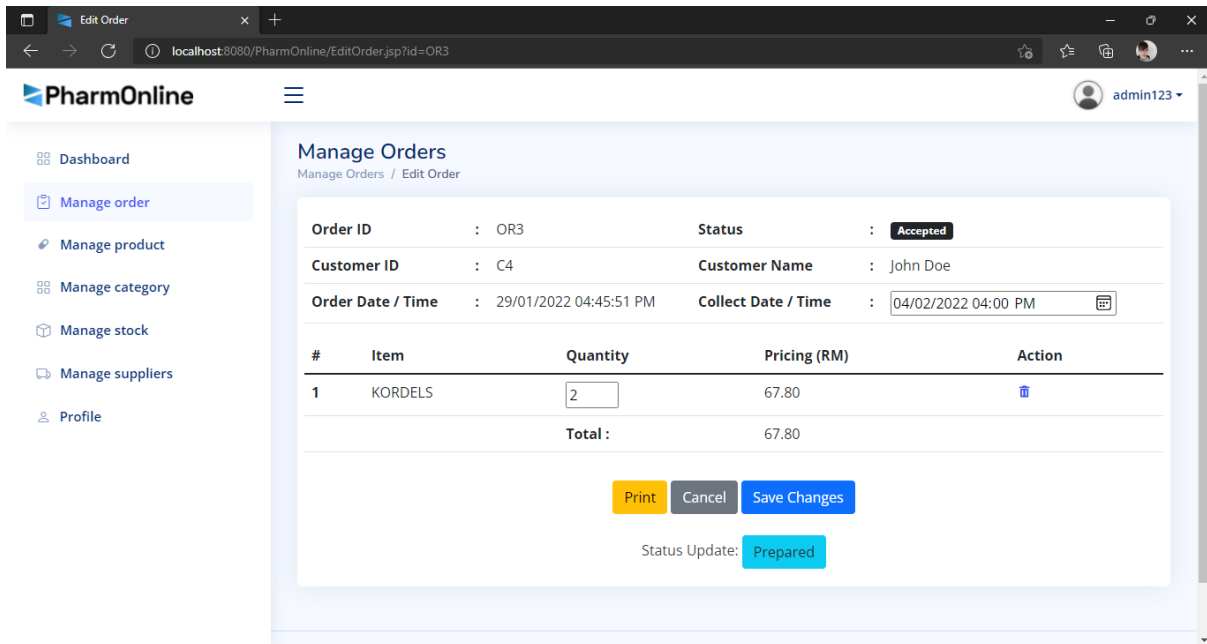


Figure 5.29: Screenshot of interface Edit Order

```

while(rsorderproduct.next()){
    count++;
    String selectedproductID = rsorderproduct.getString(2);
    out.println("<input type='hidden' id='selectedproductID'+count+' value='"+selectedproductID+"'>");
    int orderProductquantity = rsorderproduct.getInt(3);
    int totalorderproductquantity=0;
    String queryorderproductquantity="";
    if (excludedOrderID.isEmpty()){
        queryorderproductQuantity = "SELECT * FROM orderproduct WHERE productID='"+selectedproductID+"'";
    }
    else{
        queryorderproductQuantity = "SELECT * FROM orderproduct WHERE productID='"+selectedproductID+"' AND orderID NOT IN ('"+excludedOrderID+"')";
    }

    ResultSet rsorderproductQuantity = storderproductRow.executeQuery(queryorderproductQuantity);
    while(rsorderproductQuantity.next()){
        totalorderproductquantity+=rsorderproductQuantity.getInt(3);
    }

    out.println("<tr>");
    out.println("<th scope='row'>"+count+"</th>");
    String queryProduct="SELECT * FROM product WHERE product_ID='"+selectedproductID+"'";
    ResultSet rsProduct=stProduct.executeQuery(queryProduct);
    rsProduct.next();
    int productquantity = rsProduct.getInt(6);
    out.println("<input type='hidden' id='productquantity'+count+' name='productquantity'+count+' value='"+productquantity+"'>");

```

Figure 5.30: Screenshot of code Edit Order in EditOrder.jsp

viii. Reporting

The screenshot displays three summary tables in a reporting interface. Each table has a search bar and a 'Showing 1 to 1 of 1 entries' message.

Products Summary

Product ID	Product Name	Description	Selling Price (RM)	Category	Quantity
PR-1	Softa Man Hand Disinfectant 100ML	Nurturing hand sanitiser	20.00	Hand	20

Showing 1 to 1 of 1 entries

Category Summary

Product ID	Product Name	Description
CAT-1	Hand Disinfectant	Hand Disinfectant related product

Showing 1 to 1 of 1 entries

Supplier Summary

No	Supplier Name	Supplier ID	Contact No	Address	Description	Join Date
1	Jecki Medical Supplier	S1	010-9791823	Jalan Kemayang, Johor	Provide hand disinfectants	2022-01-30

Showing 1 to 1 of 1 entries

Figure 5.31: Screenshot of interface for reporting

6.0 Reflection

See Wen Xiang

After I have finished this project, I feel that I have enhanced not only my software developing skill and also team working skill and communication skills. This is a good experience for me to learn how to develop a web application by solving real world problems. Besides that, this practical project experience has enhanced my understanding about the topic of these lectures such as Java Servlets, JSP, JDBC, JSTL & EL, as well as the MVC architecture. Apart from this, I have also learned how to be a listener by listening to other team members' opinions and understanding it and accepting it as well.

I had contributed to write the proposed solution and do the mock-up interfaces for my assigned use case which are manage order and place order. For the programming part, I have handled the modules of place order (User-View including cart function) and manage order (Admin-View & User-View). Kong Hao Yang is assigned to develop the modules of managing stock and suppliers. Loo Zhi Xuen contributed to manage category and manage product modules. Cheong Chien Li had contributed to login, register, manage profile and generate report features.

In terms of the assessment of my group, I personally think that basically the overall flow of the progress developing the project is going smoothly even though we have met some difficulties during the progress of developing. However, we have successfully solved those difficulties and managed to catch-up the timeline which we had set initially. I am very glad to form a team with my team members. From the progress of developing the project, I can feel that we really were helping each other when we met the problems. I appreciate that I have the chance to learn this course and to develop this project. Without the attentive teaching of the lecturer and help of team members, I think that I cannot successfully develop this project with my team members.

Kong Hao Yang

Throughout this project, my team-working and communication skills are further enhanced since this project is team-based which requires me to cooperate with other team members to develop a web application. Besides that, this project provides me an opportunity to apply the knowledge that I have learned from the course of internet programming in solving real world problems. And, this practical experience has further enhanced my understanding toward the topics taught during lectures such as JSP & JSP tags, MVC architecture, JavaBean, Session, JDBC and so on.

In the words of contribution to the project, my contribution is doing the chapter of module functionality and actor as well as the mockup interfaces for the modules that are assigned to me in project proposal, and developing the modules of managing stock and suppliers. For team members, Cheong Chien Li is responsible for the modules of login, register, manage profile and generate report, while See Wen Xiang handles the modules of place order and manage order. Lastly, Loo Zhi Xuen is handling the modules of manage category and manage product as well as view product.

In the words of assessment of the group, I would say that the web application development activity is going smoothly even though we did face some obstacles in the development progress. However, all the obstacles faced are successfully resolved by our team with the hardworks and determination of our team. Besides that, every member is very responsible for this project as everyone manages to complete the individual tasks assigned with good quality within the time planned. And, the outcome of the application development after integration is satisfying as every single functionality of the application is able to achieve the expected functionality in the proposal. In short, I am really proud of the web application that we have developed and I also appreciate the contribution of each member in this project.

Loo Zhi Xuen

From the project as a collaborative learning interaction, I learned a lot in terms of hard skills and soft skills. In terms of hard skills, implementation of the knowledge gained from the Internet Programming class definitely helps me to have better understanding towards the syllabus. Most of the modules such as servlet, JSP, session, JDBC, MVC architecture are implemented in the project. In terms of soft skills, the project enhances my real-world problem solving skills as the team needs to figure out an idea to solve the issue that can be seen in the real world. Besides, working in a team trains my communication skills to deliver my ideas and thoughts efficiently to the team. Last but not least, respecting, accepting and learning from each other are very important in a team so everyone can work together in a comfortable situation.

For the contribution part, I'm responsible for handling the category and product modules. For the member, Kong Hao Yang is responsible for completing supplier and stock modules. Cheong Chien Li is assigned to handle login, register, manage profile and generate report features. Last but not least, See Wen Xiang is handling the cart and order modules.

For the assessment of the group, the progress of the group meets the expectation of the timeline. Initially, the progress might not be as fast as the team expected, but the team managed to rapid up the progress to make sure all functionality was built before the demonstration. I feel comfortable to work in the team as each team member is actively to share their findings and ideas and show responsibilities in completing the assigned individual part. In conclusion, I appreciate the help and contributions of all members in this project.

Cheong Chien Li

This project has been very helpful to me in terms of polishing my skills as a software developer. In the beginning of this semester, My team members and I had trouble deciding our project title. We all agreed that we never had the chance to work on any medical-related software, so we decided to build an online pharmacy website. At first, I wasn't sure whether or not I would be interested in working on medical-related software. However, throughout the development process, I've come to realise that medical/pharmaceutical software are actually quite fun to work with.

Throughout this project, I was able to learn and gain much experience on developing websites using Java Servlets, JSP, JDBC, JSTL & EL, as well as the MVC architecture. I think that these skills and knowledge are very valuable to me as a future developer because there are many employers out there who are looking for candidates with these skills. I was

able to grasp the concepts of these technologies relatively well after spending the semester working on this project.

This project has also enabled me to improve my team-working and communication skills as my team members and I have faced many difficulties throughout this project. It was only through great teamwork and communication, that we were able to pull through and resolve all of the difficulties that we faced. For example, we had to put our communication skills to the test when discussing how to make all of our interfaces consistent. This is because everybody had been assigned some parts of the system, so the issue of not having a consistent user interface was apparent during the first few weeks of the semester. It was only via great teamwork and communication among each of us that allowed us to complete the project successfully.

Throughout this project, I was assigned to handle the modules of user registration, login, forget password, users profile, as well as generate reports. I mainly focused on CRU of the CRUD functionalities, because my team and I had no plans to allow the registered users to delete their accounts. Even so, I was still able to learn much valuable knowledge throughout the semester, and I am now one step closer to working in the real world outside as a software engineer.

In conclusion, this project has helped me grow a lot as a software engineering student. This is because it forced me to apply the skills that I have learned in class into solving a real-world problem. Apart from just learning more about technical coding skills, this project was also able to allow me to practice and further improve my communication and team-working skills, by highly encouraging us to work together with our team members to produce our final application.