

STOCK PREDICTION SYSYTEM

A MINI-PROJECT REPORT

Submitted by

K M PADMASRI (230701223)

In partial fulfillment of the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING



RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

An Autonomous Institute

CHENNAI

NOVEMBER 2023

BONAFIDE CERTIFICATE

Certified that this project “STOCKHUB – Inventory Management System ” is the bonafide work of “K M PADMASRI ” who carried out the project work under my supervision.

This mini project report is submitted for the viva voce examination to be held on _____

SIGNATURE

ASSISTANT PROFESSOR

Mrs.ASWANA LAL

Dept. of Computer Science and Engg,
Computer Science and Engg,
Rajalakshmi Engineering College
Chennai

INTERNAL EXAMINER

EXTERNAL EXAMINER

TABLE OF CONTENTS

S.NO	TITLE	PAGE
1	Introduction	05
2	Scope of Project	06
3	UML Diagrams	09
4	Code Implementation	22
5	Conclusion	27
6	Reference	29

ABSTRACT:

An inventory management system is an essential tool in modern supply chain management, providing a comprehensive solution for tracking and managing stock levels, orders, sales, and deliveries. This system facilitates the automation of inventory control processes, ensuring accurate and real-time data visibility, which helps in reducing excess stock and avoiding stockouts. By employing advanced forecasting and analytics, an inventory management system enables businesses to optimize their inventory turnover rates, streamline operations, and enhance customer satisfaction. This leads to improved efficiency, cost savings, and better resource allocation, ultimately driving the organization's growth and competitiveness in the market.

INTRODUCTION

An inventory management system is an essential tool in modern supply chain management, providing a comprehensive solution for tracking and managing stock levels, orders, sales, and deliveries. This system facilitates the automation of inventory control processes, ensuring accurate and real-time data visibility, which helps in reducing excess stock and avoiding stockouts. By employing advanced forecasting and analytics, an inventory management system enables businesses to optimize their inventory turnover rates, streamline operations, and enhance customer satisfaction. This leads to improved efficiency, cost savings, and better resource allocation, ultimately driving the organization's growth and competitiveness in the market.

PROBLEM STATEMENT

The existing manual processes within inventory management systems often result in inefficiencies, inaccuracies, and delays in managing stock levels. Without a streamlined system for inventory tracking, order management, and distribution, businesses struggle to respond effectively to dynamic market demands. This leads to issues such as stockouts, overstocking, and increased operational costs. Improved automation and integration are necessary to enhance efficiency, accuracy, and responsiveness, ultimately leading to better customer satisfaction and competitive advantage. Developing a comprehensive Inventory Management System that automates and integrates key processes, including stock monitoring, order management, inventory tracking, and distribution, is crucial for modern businesses. This system enhances operational efficiency, reduces manual errors, and ensures real-time visibility into inventory levels. It streamlines order management, preventing stockouts and overstocking, while integration

with other business systems ensures seamless operations. Advanced analytics provide valuable insights for informed decision-making, helping businesses respond effectively to market demands. Ultimately, this leads to cost savings, improved customer satisfaction, and enhanced competitiveness.

OBJECTIVE

An Inventory Management System is designed to efficiently oversee various aspects of stock management within a business. Its objectives encompass tracking inventory levels, optimizing order processes, ensuring timely stock replenishment, and managing distribution channels effectively. The system also focuses on maintaining product quality through rigorous quality control measures and efficient supplier management. Reporting and analytics functionalities provide valuable insights into inventory performance, aiding in informed decision-making. Accessibility and security features ensure authorized access to sensitive inventory data while safeguarding against unauthorized breaches. Compliance and documentation protocols help businesses adhere to regulatory requirements, minimizing legal risks. By achieving these objectives, an Inventory Management System enhances operational efficiency, reduces costs, improves customer satisfaction, and strengthens the overall competitiveness of the business in the market.

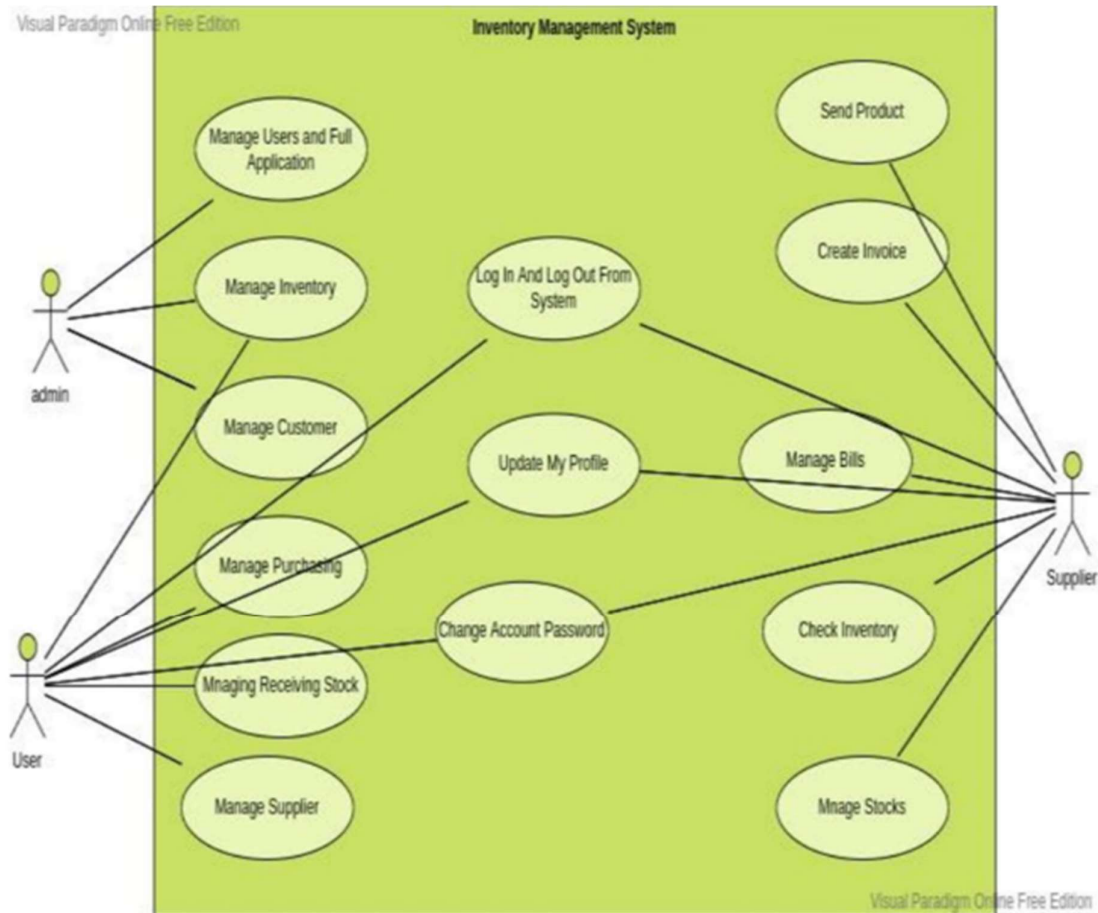
MODULES

- Admin Module: Facilitates system administration tasks such as user management, access control, and configuration settings for the inventory management system.
- Employee Module: Manages employee accounts, permissions, and roles within the system, ensuring appropriate access levels and accountability.
- Category Module: Organizes products into hierarchical categories, simplifying navigation and facilitating efficient inventory management and product grouping.

- **Product Module:** Stores and manages detailed information about products, including descriptions, pricing, quantities, and related data for effective inventory tracking and management.
- **Customer Module:** Handles customer-related data such as profiles, contact information, purchase history, and preferences, enabling personalized interactions and targeted marketing strategies.
- **Order Details Module:** Tracks and manages the specifics of orders placed by customers, including item details, quantities, pricing, shipment information, and order statuses, ensuring smooth order processing and fulfillment.

UML DIAGRAMS

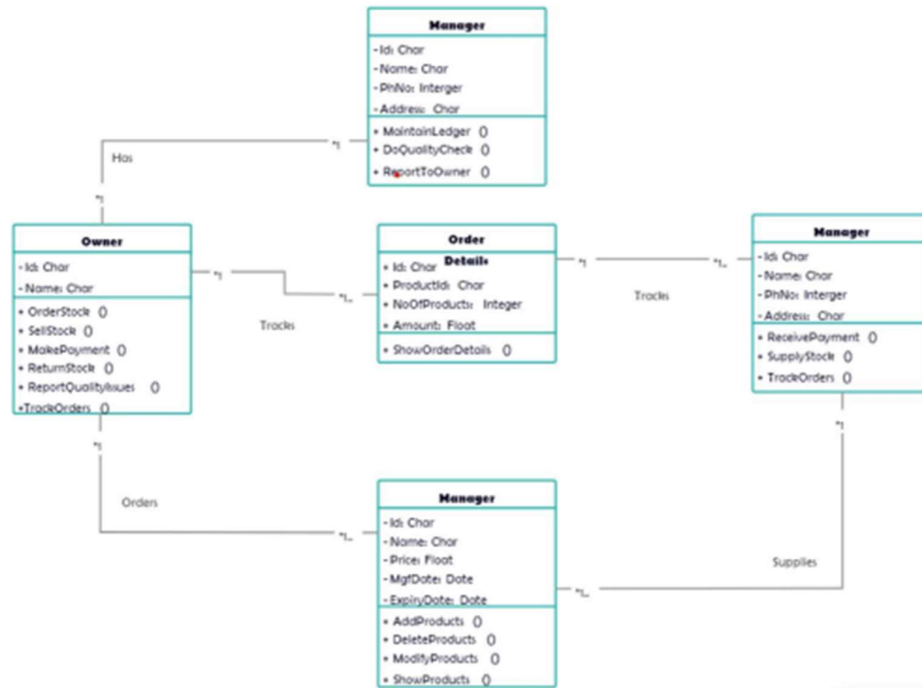
CLASS DIAGRAM



A use case diagram for an inventory management system illustrates the interactions between various actors and the system's functional requirements. The primary actors typically include Inventory Managers, Warehouse Staff, Suppliers, and Customers. Key use cases for Inventory Managers include Manage Inventory Levels, Generate Reports, and Monitor Stock Movements. Warehouse Staff interact with use cases such as Receive Shipments, Update Stock Levels, and Process Orders. Suppliers are involved in Provide Products and Receive Purchase Orders, while Customers are linked to Place Orders and Track Order Status. This diagram highlights the system's core functionalities,

ensuring all user roles and their interactions are clearly defined for efficient system design and implementation

USE CASE :



A class diagram for an inventory management system includes key classes such as Product, Category, Supplier, Customer, Order, Inventory, and User. The Product class, with attributes like productID, name, and price, is associated with Category. Supplier links to multiple products and contains supplierID and contactInfo. Customer connects to Order, which includes orderID, orderDate, and totalAmount. Inventory manages stock details with attributes like inventoryID and quantityAvailable. User includes userID, username, and roles. This diagram captures the essential classes, their attributes, and relationships, providing a clear system structures.

PROGRAM CODE

```
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

class InventoryManagementSystem extends Frame implements ActionListener {
    private TextField productIdField, productNameField, productPriceField,
    customerIdField,
    customerNameField;

    private Button addProductButton, addCustomerButton, addToCartButton,
    displayBillButton;

    private TextArea displayArea;
    private Connection connection;

    public InventoryManagementSystem() {
        setTitle("Inventory Management System");
        setSize(600, 400);
        setLayout(new FlowLayout());
        addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        });
        // Initialize database connection

        try {
            15 | Page
            connection
            DriverManager.getConnection("jdbc:mysql://localhost:3306/inventory",
            "username", "password");
        } catch (SQLException e) {
```

```

e.printStackTrace();
}
// Product details
Label productIdLabel = new Label("Product ID:");
productIdField = new TextField(10);
Label productNameLabel = new Label("Product Name:");
productNameField = new TextField(20);
Label productPriceLabel = new Label("Product Price:");
productPriceField = new TextField(10);
addProductButton = new Button("Add Product");
addProductButton.addActionListener(this);
// Customer details
Label customerIdLabel = new Label("Customer ID:");
customerIdField = new TextField(10);
Label customerNameLabel = new Label("Customer Name:");
customerNameField = new TextField(20);
addCustomerButton = new Button("Add Customer");
addCustomerButton.addActionListener(this);
// Cart operations
addToCartButton = new Button("Add to Cart");
addToCartButton.addActionListener(this);
displayBillButton = new Button("Display Bill");
displayBillButton.addActionListener(this);
// Display area
displayArea = new TextArea(15, 50);
// Add components to frame
add(productIdLabel);

```

```

add(productIdField);
add(productNameLabel);
add(productNameField);
add(productPriceLabel);
add(productPriceField);
add(addProductButton);
add(customerIdLabel);
add(customerIdField);
add(customerNameLabel);
add(customerNameField);
add(addCustomerButton);
add(addToCartButton);
add(displayBillButton);
add(displayArea);
setVisible(true);
}

public void actionPerformed(ActionEvent e) {
if (e.getSource() == addProductButton) { 17 | Page
addProduct();
} else if (e.getSource() == addCustomerButton) {
addCustomer();
} else if (e.getSource() == addToCartButton) {
addToCart();
} else if (e.getSource() == displayBillButton) {
displayBill();
}
}
}

```

```

private void addProduct() {
    String productId = productIdField.getText();
    String productName = productNameField.getText();
    double productPrice = Double.parseDouble(productPriceField.getText());
    try {
        String query = "INSERT INTO products (product_id, product_name, price)
VALUES (?,
?, ?)";
        PreparedStatement statement = connection.prepareStatement(query);
        statement.setString(1, productId);
        statement.setString(2, productName);
        statement.setDouble(3, productPrice);
        statement.executeUpdate();
        displayArea.append("Product added successfully!\n");
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

private void addCustomer() {
    String customerId = customerIdField.getText();
    String customerName = customerNameField.getText();
    try {
        String query = "INSERT INTO customers (customer_id, customer_name)
VALUES (?,
?)";
        PreparedStatement statement = connection.prepareStatement(query);
        statement.setString(1, customerId);
        statement.setString(2, customerName);
    }
}

```

```

statement.executeUpdate();
displayArea.append("Customer added successfully!\n");
} catch (SQLException e) {
e.printStackTrace();
}
}

private void addToCart() {
String customerId = customerIdField.getText();
String productId = productIdField.getText();
try {
String query = "INSERT INTO cart (customer_id, product_id) VALUES (?, ?)";
PreparedStatement statement = connection.prepareStatement(query);
statement.setString(1, customerId);
statement.setString(2, productId);
statement.executeUpdate();
displayArea.append("Item added to cart successfully!\n");
} catch (SQLException e) {
e.printStackTrace();
}
}

private void displayBill() {
String customerId = customerIdField.getText();
double totalBill = 0;
try {
String query = "SELECT p.product_name, p.price FROM products p JOIN cart
c ON
p.product_id = c.product_id WHERE c.customer_id = ?";
PreparedStatement statement = connection.prepareStatement(query);

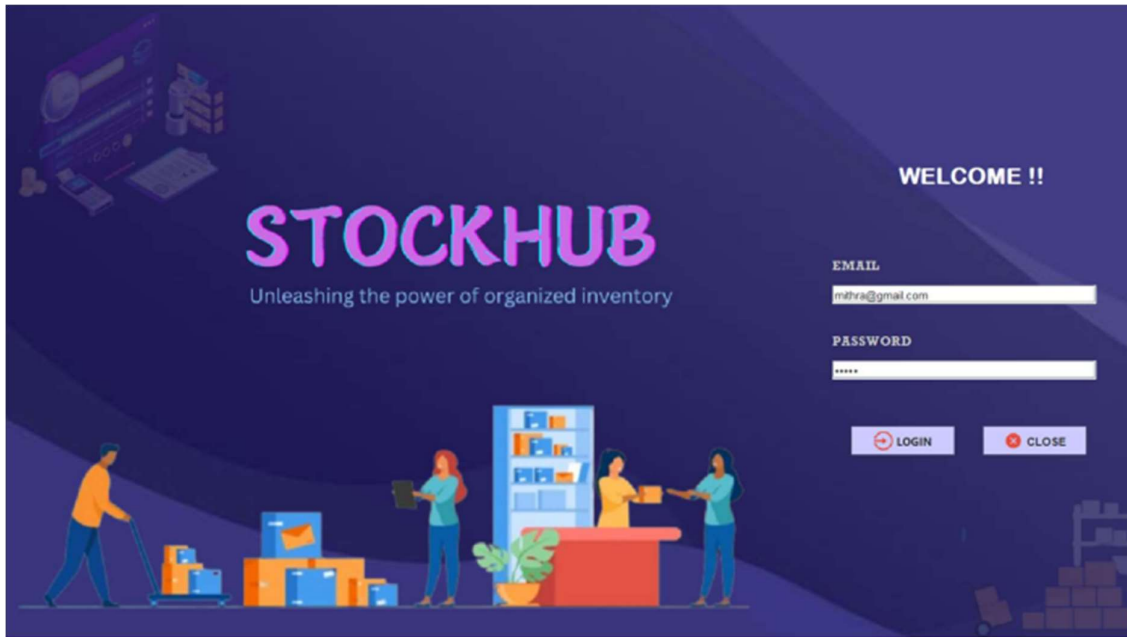
```

```

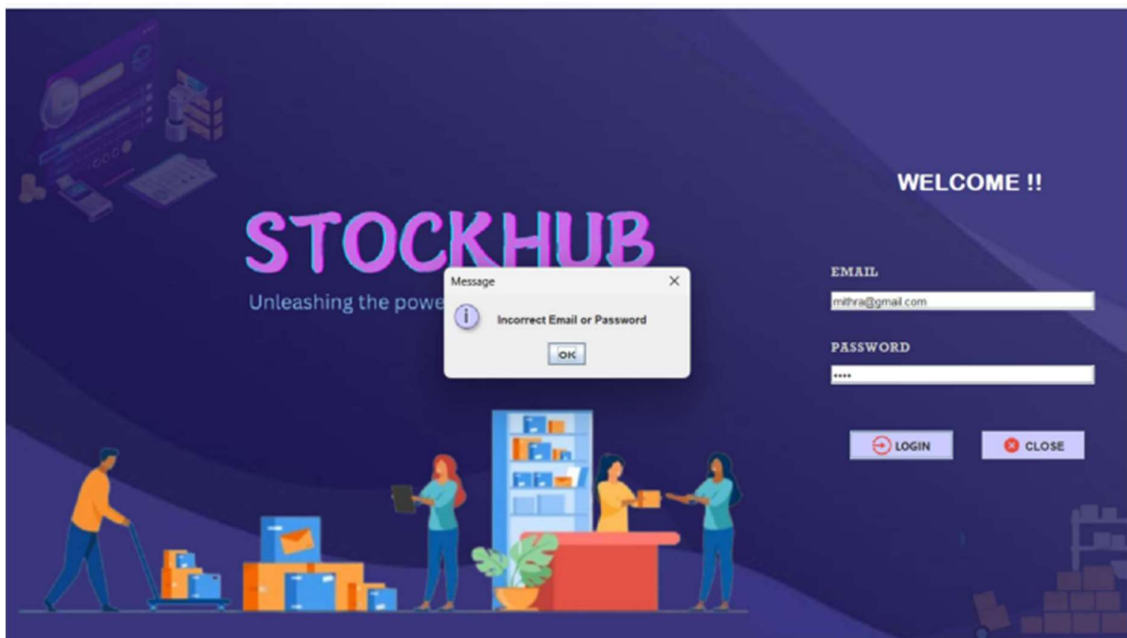
statement.setString(1, customerId);
ResultSet resultSet = statement.executeQuery();
displayArea.append("Customer Bill:\n");
displayArea.append("Product\t\tPrice\n");
while (resultSet.next()) {
String productName = resultSet.getString("product_name");
double price = resultSet.getDouble("price");
totalBill += price;
displayArea.append(productName + "\t\t" + price + "\n");
} 20 | Page
displayArea.append("Total Bill: " + totalBill + "\n");
} catch (SQLException e) {
e.printStackTrace();
}
}
public static void main(String[] args) {
new InventoryManagementSystem();
}
}

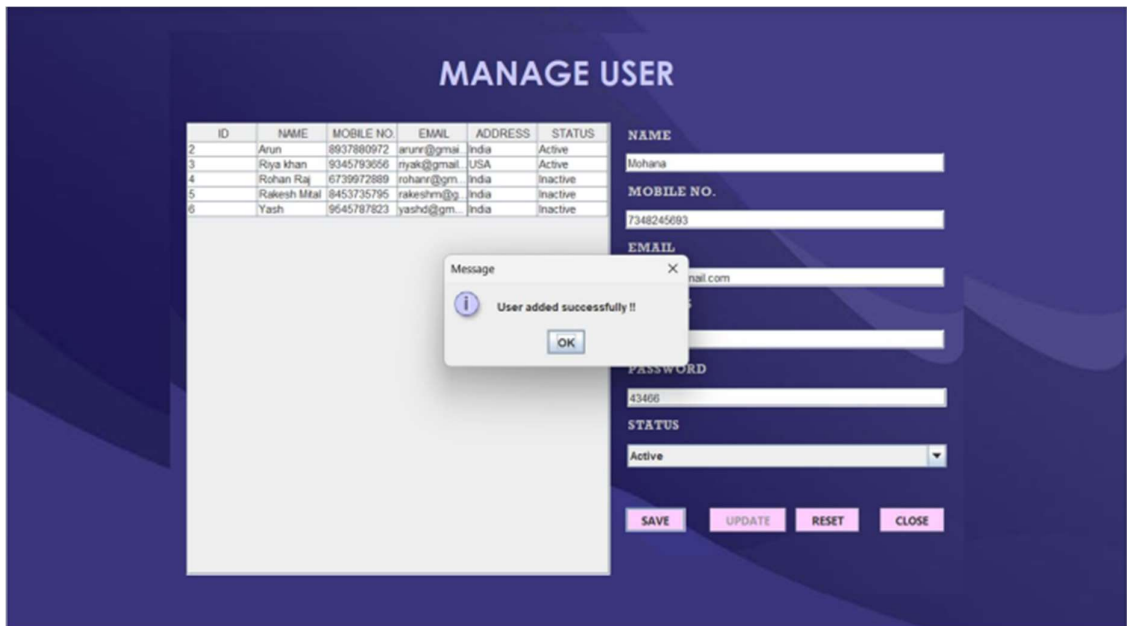
```

OUTPUT:



The details for the login page need to be entered.





MANAGE USER

ID	NAME	MOBILE NO.	EMAIL	ADDRESS	STATUS
2	Arun	8937880972	arunr@gmail	India	Active
3	Riya Khan	9345793656	riyak@gmail	USA	Active
4	Rohan Raj	6739972889	rohanr@gm	India	Inactive
5	Rakesh Mital	8453735795	rakeshm@g	India	Inactive
6	Yash	9545787823	yashd@gm	India	Active
7	Mohana	7348245693	mohana@g	India	Active

Message

User updated successfully !!

OK

NAME

Yash

MOBILE NO.

9545787823

EMAIL

il.com

PASSWORD

STATUS

Active

SAVE

UPDATE

RESET

CLOSE

MANAGE USER

ID	NAME	MOBILE NO.	EMAIL	ADDRESS	STATUS
2	Arun	8937880972	arunr@gmail	India	Active
3	Riya Khan	9345793656	riyak@gmail	USA	Active
4	Rohan Raj	6739972889	rohanr@gm	India	Inactive
5	Rakesh Mital	8453735795	rakeshm@g	India	Inactive
6	Yash	9545787823	yashd@gm	India	Active
7	Mohana	7348245693	mohana@g	India	Active

NAME

MOBILE NO.

EMAIL

ADDRESS

PASSWORD

STATUS

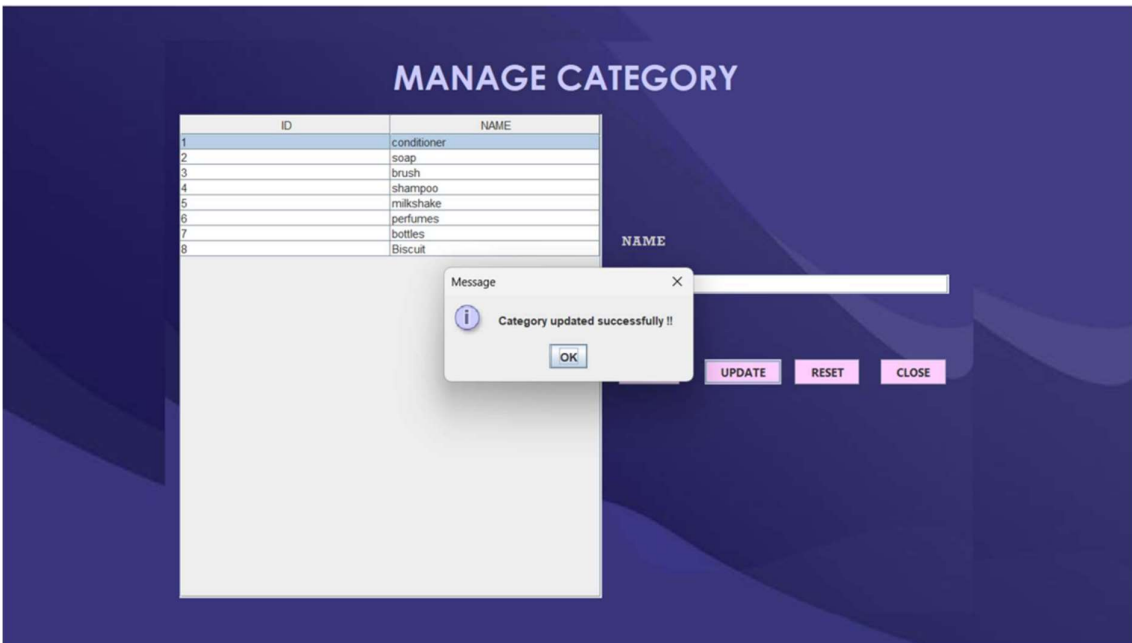
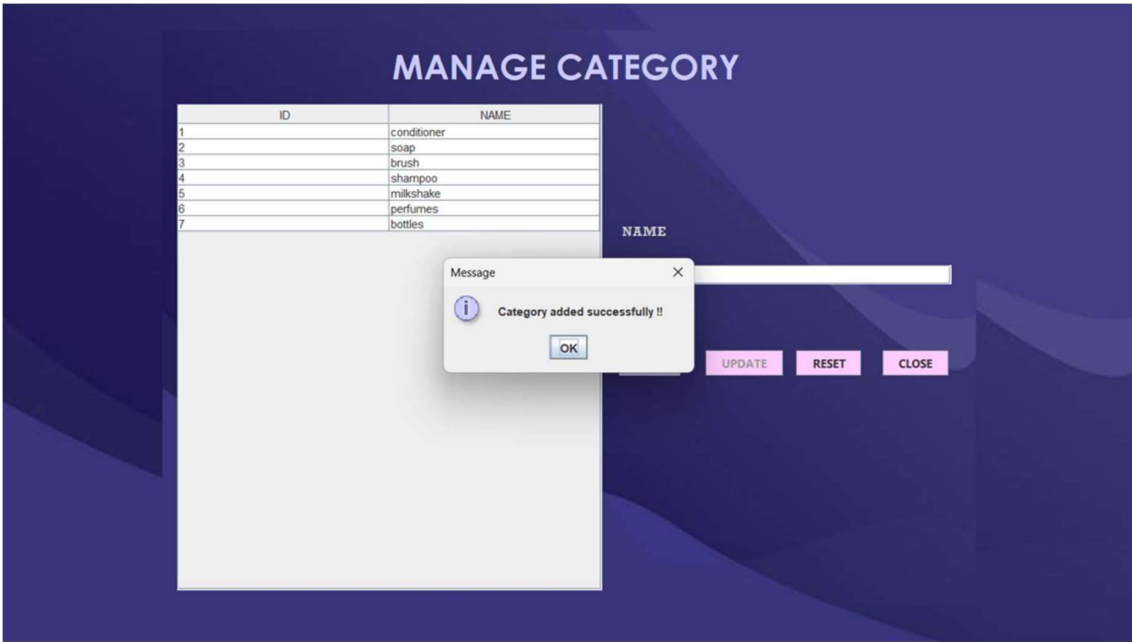
Active

SAVE

UPDATE

RESET

CLOSE



MANAGE CATEGORY

ID	NAME
1	chips
2	soap
3	brush
4	shampoo
5	milkshake
6	perfumes
7	bottles
8	Biscuit

NAME

SAVE

UPDATE

RESET

CLOSE

MANAGE ORDER

CUSTOMER LIST

ID	NAME	MOBILE NO.	EMAIL
1	Raam	9548656937	raams@gmail
2	Sundar	6964792854	sundarm@gm
3	Malini	7520456586	malini@gmail
4	Rocky	8538572899	roc@gmail.com

PRODUCT LIST

ID	NAME	PRICE	QTY	DESC	CAT_ID	CAT_NAME
1	Pears	34	13	Bathing so	2	soap
2	Pepsodent	24	24	Toothpaste	3	brush
3	Loreal sha	185	54	Bathing sh	4	shampoo
4	hersheys	20	29	drink	5	milkshake
5	lux	45	31	Bathing so	2	soap
6	tupperware	40	38	Drinking w	7	bottles
7	lays	10	25	Food	1	chips

CART

PRO_ID	NAME	QTY	PRICE	DESC	SUB_TO
7	lays	2	10	Food	20

SELECTED CUSTOMER :

NAME : Rocky
MOBILE NO. : 8538572899
EMAIL : roc@gmail.com

SELECTED PRODUCT :

PRODUCT NAME : lays
PRODUCT PRICE : 10
PRODUCT DESCRIPTION : Food
ORDER QUANTITY : 2

ADD TO CART

TOTAL AMOUNT : Rs. 20

SAVE ORDER DETAILS

RESET

CLOSE

Message

Added Successfully

OK

MANAGE ORDER

CUSTOMER LIST

ID	NAME	MOBILE NO.	EMAIL
1	Raam	954856937	raams@gmail
2	Sundar	6964792854	sundarm@gmail
3	Malini	7526458586	malina@gmail
4	Rocky	8538572899	roc@gmail.com

PRODUCT LIST

ID	NAME	PRICE	QTY	DESC	CAT_ID	CAT_NAME
1	Pears	34	13	Bathing so	2	soap
2	Pepsodent	24	24	Toothpaste	3	brush
3	Loreal sha	185	54	Bathing sh	4	shampoo
4	hersheys	20	29	drink	5	milkshake
5	lux	45	31	Bathing so	2	soap
6	tupperware	40	38	Drinking w	7	bottles
7	lays	10	25	Food	1	chips

CART

PRO_ID	NAME	QTY	PRICE	DESC	SUB_TO
7	lays	2	10	Food	20
4	hersheys	1	20	drink	20
2	Pepsodent	2	24	Toothpaste	48
1	Pears	3	34	Bathing s	102
3	Loreal sh	1	185	Bathing s	185

Message X
 Product is out of stock. Only 25 Left
 OK

SELECTED CUSTOMER :

NAME :

MOBILE NO. :

EMAIL :

SELECTED PRODUCT :

PRODUCT NAME :

PRODUCT PRICE :

PRODUCT DESCRIPTION :

ORDER QUANTITY :

TOTAL AMOUNT : Rs. 375

MANAGE ORDER

CUSTOMER LIST

ID	NAME	MOBILE NO.	EMAIL
1	Raam	954856937	raams@gmail
2	Sundar	6964792854	sundarm@gmail
3	Malini	7526458586	malina@gmail
4	Rocky	8538572899	roc@gmail.com

PRODUCT LIST

ID	NAME	PRICE	QTY	DESC	CAT_ID	CAT_NAME
1	Pears	34	13	Bathing so	2	soap
2	Pepsodent	24	24	Toothpaste	3	brush
3	Loreal sha	185	54	Bathing sh	4	shampoo
4	hersheys	20	29	drink	5	milkshake
5	lux	45	31	Bathing so	2	soap
6	tupperware	40	38	Drinking w	7	bottles
7	lays	10	25	Food	1	chips

CART

PRO_ID	NAME	QTY	PRICE	DESC	SUB_TO
7	lays	2	10	Food	20
4	hersheys	1	20	drink	20
2	Pepsodent	2	24	Toothpaste	48
1	Pears	3	34	Bathing s	102
3	Loreal sh	1	185	Bathing s	185

Message X
 Product already exist in cart !!
 OK

SELECTED CUSTOMER :

NAME :

MOBILE NO. :

EMAIL :

SELECTED PRODUCT :

PRODUCT NAME :

PRODUCT PRICE :

PRODUCT DESCRIPTION :

ORDER QUANTITY :

TOTAL AMOUNT : Rs. 375

MANAGE ORDER

CUSTOMER LIST

ID	NAME	MOBILE NO.	EMAIL
1	Raam	9548856937	raams@gmail
2	Sundar	6964792854	sundarm@gmail
3	Malini	7526458586	malinia@gmail
4	Rocky	8538572899	roc@gmail.com

PRODUCT LIST

ID	NAME	PRICE	QTY	DESC	CAT_ID	CAT_NAME
1	Pears	34	13	Bathing so	2	soap
2	Pepsodent	24	24	Toothpaste	3	brush
3	Loreal sha	185	54	Bathing sh	4	shampoo
4	hersheys	20	29	drink	5	milkshake
5	lux	45	31	Bathing so	2	soap
6	tupperware	40	38	Drinking w	7	bottles
7	lays	10	25	Food	1	chips

CART

PRO_ID	NAME	QTY	PRICE	DESC	SUB_TO
7	lays	2	10	Food	20
4	hersheys	1	20	drink	20
2	Pepsodent	2	24	Toothpaste	48
1	Pears	3	34	Bathing s	102
3	Loreal sh	1	185	Bathing s	185
5	lux	1	45	Bathing s	45

SELECTED CUSTOMER :

NAME :

MOBILE NO. :

EMAIL :

SELECTED PRODUCT :

PRODUCT NAME :

PRODUCT PRICE :

PRODUCT DESCRIPTION :

ORDER QUANTITY :

[ADD TO CART](#)

TOTAL AMOUNT : Rs. 420

[SAVE ORDER DETAILS](#)

[RESET](#)

[CLOSE](#)

Menu
Inventory_bill-17792... [+ Create](#)

All tools Edit Convert E-Sign
Find text or tools

STOCKHUB

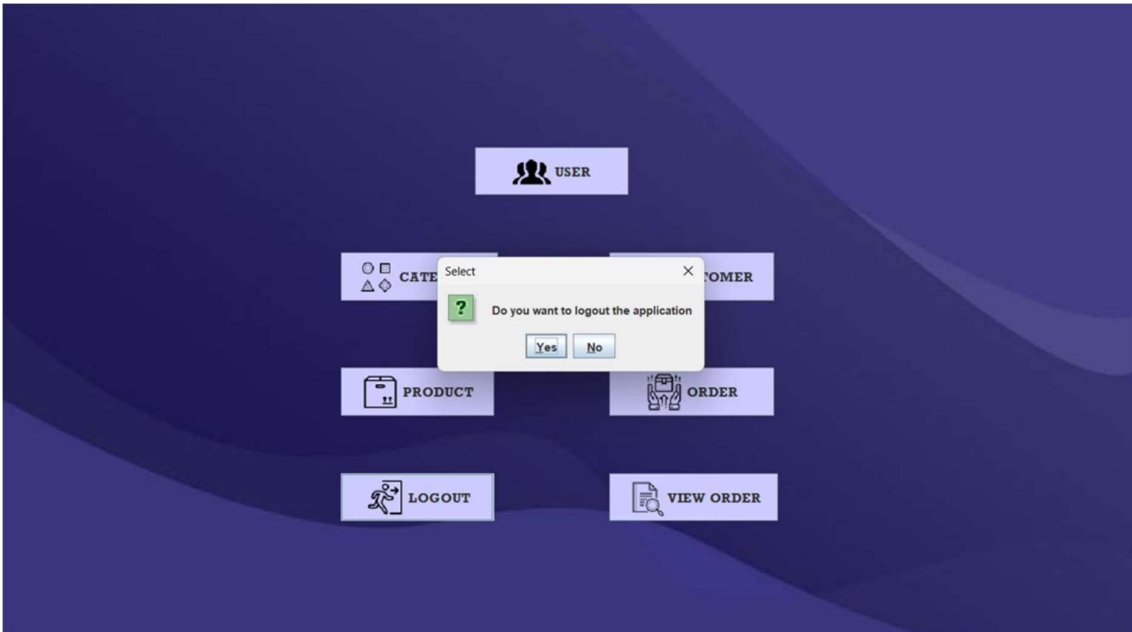
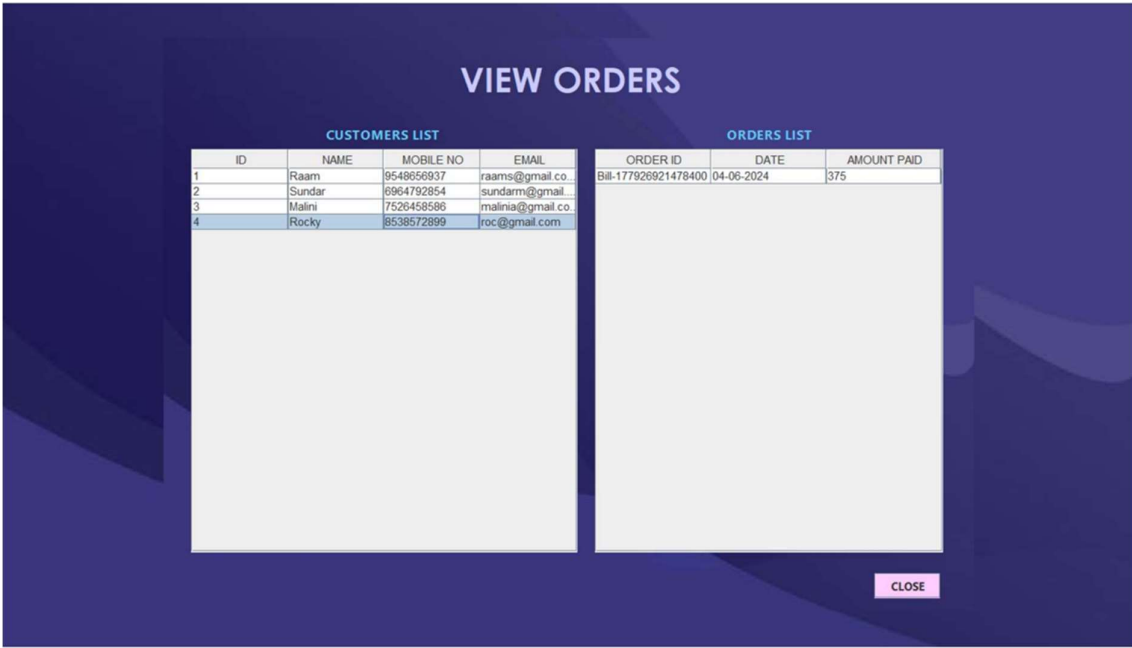
Order Id : Bill-177926921478400

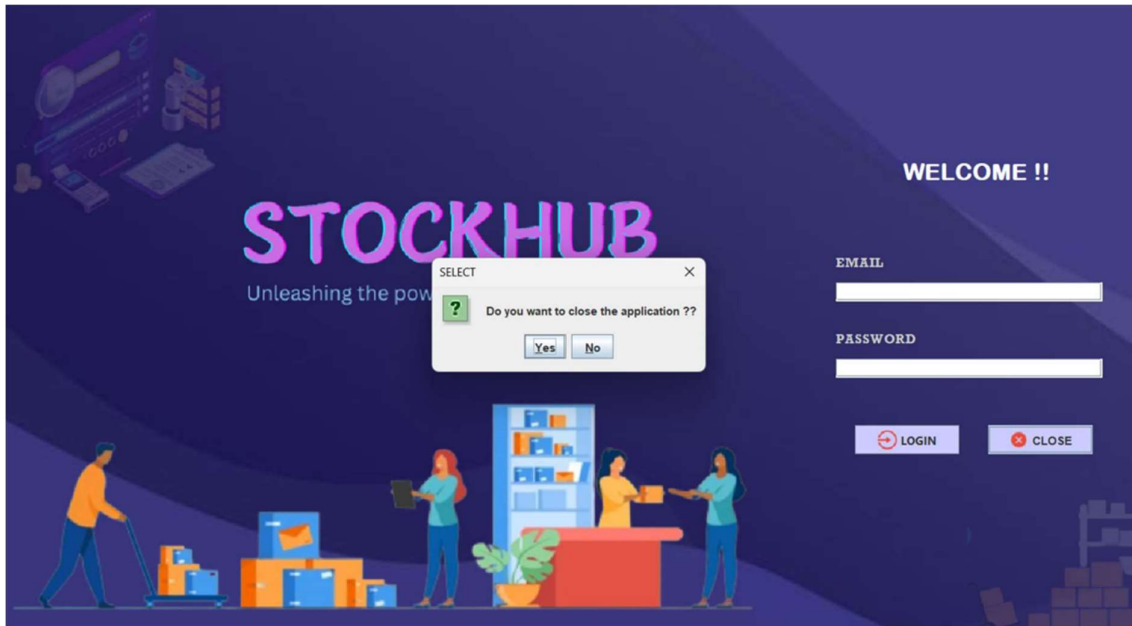
Date : 04-06-2024

Total Amount Paid : Rs.375

Name	Description	Price per unit	Quantity	Sub Total Price
lays	Food	10	2	20
hersheys milkshake	drink	20	1	20
Pepsodent	Toothpaste	24	2	48
Pears	Bathing soap	34	3	102
Loreal shampoo	Bathing shampoo	185	1	185

Thank you , Please visit again.





CONCLUSION

StockHub - Inventory Management System, has successfully addressed the challenges of manual inventory control. Some key takeaways includes :

- **Enhanced Functionalities:** StockHub successfully implemented functionalities like employee and customer management, a user-friendly cart system for customer purchases, and real-time bill generation, leading to a more efficient and streamlined workflow.
- **Positive Impact:** The system's adoption has resulted in cost savings through optimized inventory control and reduced manual labor. Additionally, improved accuracy and faster processing have likely led to enhanced customer service experiences.
- **Future Development:** StockHub presents exciting opportunities for further development. Integration with existing business systems like accounting software could streamline financial processes. Advanced reporting features providing insights into sales trends and purchase patterns can empower data-driven decision-making.
- **Improved Efficiency:** StockHub streamlines inventory management processes, reducing time spent on manual tasks like data entry and stock checks.
- **Enhanced Accuracy:** Real-time data tracking minimizes errors and discrepancies often associated with manual methods.
- **Optimized Inventory Levels:** StockHub facilitates informed decisions on stock levels, preventing stockouts and overstocking.
- **Streamlined Operations:** The system provides a centralized platform for managing employees, customers, products, and cart functionalities, promoting smoother workflows.
- **Increased Transparency:** StockHub offers clear visibility into inventory levels, facilitating better decision-making and improved communication across departments. In summary, StockHub has proven to be a valuable tool for streamlining inventory management. By enhancing efficiency, accuracy, and overall transparency, the system empowers informed decision-making and paves the way for continued business growth. Future development can focus on further integration and advanced reporting capabilities to unlock StockHub's full potential.

REFERENCES

WEBSITES :

1. NetbeanIDE – <https://netbeans.apache.org/front/main/index.html>
2. Java Swing Tutorial – <https://www.javatpoint.com/java-swing>
3. NetbeansTutorial – <https://netbeans.apache.org/tutorial/main/kb/>
4. MySQLTutorial – <https://www.w3schools.com/MySQL/default.asp>
5. JDBC Tutorial – <https://www.javatpoint.com/java-jdb>

BOOKS :

1. "Java: The Complete Reference" by Herbert Schildt: Comprehensive guide to Java programming, covering core features, APIs, and frameworks.
2. "Java Swing" by Marc Loy, Robert Eckstein, and Dave Wood: Detailed guide to creating GUIs with Java Swing toolkit.
3. "NetBeans: The Definitive Guide" by Tim Boudreau, Jesse Glick, Simeon Greene, and Jack J. Woehr: Complete guide to using NetBeans IDE for Java development.