INVENTORY MANAGEMENT SYSTEM

A PROJECT REPORT

Submitted by

SANTHOSH SHARMA M (2303811710421136)

in partial fulfillment of requirements for the award of the course

CGB1201 - JAVA PROGRAMMING

In

COMPUTER SCIENCE AND ENGINEERING

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

NOVEMBER- 2024

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report on **" Inventory Management System"** is the bonafide work of **M SANTHOSH SHARMA(2303811710421132)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

CGB1201-JAVA PROGRAMMING Dr.A.DELPHIN CAROLINA RANI, M.E.,Ph.D., HEAD OF THE DEPARTMENT PROFESSOR	CGB1201-JAVA PROGRAMMING Mr.MALARMANNAN A, M.E., SUPERVISOR ASSISTANTP ROFESSOR
SIGNATURE	**SIGNATURE**
Mrs.A.Delphin Carolina Rani, M.E.,Ph.D.,	Mr. A. Malarmannan, M.E.,
HEAD OF THE DEPARTMENT	**SUPERVISOR**
PROFESSOR	ASSISTANT PROFESSOR
Department of CSE	Department of CSE
K.Ramakrishnan College of Technology (Autonomous)	K.Ramakrishnan College of Technology (Autonomous)
Samayapuram–621112.	Samayapuram–621112.

Submitted for the viva-voce examination held on ……06/12/2024………

CGB1201-JAVA PROGRAMMING Mr. M.SARAVANAN, M.E., INTERNAL EXAMINER ASSISTANT PROFESSOR	CGB1201-JAVA PROGRAMMING Mr.R. KARTHIK, M.E., EXTERNAL EXAMINER ASSISTANT PROFESSOR 8138-SCE, TRICHY. 06/12/2024
INTERNAL EXAMINER	EXTERNAL EXAMINER

DECLARATION

I declare that the project report on **"Inventory Management System"** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **CGB1201 - JAVA PROGRAMMING.**

.

Signature

M.

M SANTHOSH SHARMA

Place: Samayapuram

Date:06-12-2024

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and in-debt to our institution "**K.Ramakrishnan College of Technology (Autonomous)**", for providing us with the opportunity to do this project.

I glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequate duration in completing our project.

I would like to thank **Dr. N. VASUDEVAN, M.Tech., Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

I whole heartily thanks to **Dr. A. DELPHIN CAROLINA RANI, M.E.,Ph.D.,** Head of the department, **COMPUTER SCIENCE AND ENGINEERING** for providing her encourage pursuing this project.

I express our deep expression and sincere gratitude to our project supervisor **MR. A. MALARMANNAN, M.E.,** Department of **COMPUTER SCIENCE AND ENGINEERING,** for his incalculable suggestions, creativity, assistance and patience which motivated us to carry out this project.

I render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

VISION OF THE INSTITUTION

To serve the society by offering top-notch technical education on par with global standards

MISSION OF THE INSTITUTION

- Be a center of excellence for technical education in emerging technologies by exceeding the needs of the industry and society.
- Be an institute with world class research facilities
- Be an institute nurturing talent and enhancing the competency of students to transform them as all-round personality respecting moral and ethical values

VISION OF DEPARTMENT

To be a center of eminence in creating competent software professionals with research and innovative skills.

MISSION OF DEPARTMENT

M1: Industry Specific: To nurture students in working with various hardware and software platforms inclined with the best practices of industry.

M2: Research: To prepare students for research-oriented activities.

M3: Society: To empower students with the required skills to solve complex technological problems of society.

PROGRAM EDUCATIONAL OBJECTIVES

1. PEO1: Domain Knowledge

To produce graduates who have strong foundation of knowledge and skills in the field of Computer Science and Engineering.

2. PEO2: Employability Skills and Research

To produce graduates who are employable in industries/public sector/research organizations or work as an entrepreneur.

3. PEO3: Ethics and Values

To develop leadership skills and ethically collaborate with society to tackle real-world challenges.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO 1: Domain Knowledge

To analyze, design and develop computing solutions by applying foundational concepts of Computer Science and Engineering.

PSO 2: Quality Software

To apply software engineering principles and practices for developing quality software for scientific and business applications.

PSO 3: Innovation Ideas

To adapt to emerging Information and Communication Technologies (ICT) to innovate ideas and solutions to existing/novel problems

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences

3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations

4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions

5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations

6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice

7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development

8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

ABSTRACT

The **Inventory Management System (IMS)** is designed to automate and streamline the process of managing inventory in a business. The system provides a centralized platform for tracking products, managing stock levels, processing orders, and generating reports. By utilizing Java programming and relational databases, the system efficiently handles the addition, update, and deletion of products, as well as real-time updates to inventory levels when products are sold or restocked. The IMS also allows for easy monitoring of low stock levels and generates alerts for timely reordering. It supports different user roles with varying access levels to ensure security and data integrity. Through a user-friendly interface built with JavaFX or Swing, the system offers both administrative and operational functionality, making it easier for businesses to manage their stock, track sales, and analyze inventory performance. With the ability to generate detailed reports, the IMS aids in decision-making and inventory forecasting. This project aims to improve the efficiency, accuracy, and reliability of inventory management processes, reducing manual errors and optimizing resource management.

ABSTRACT WITH POs AND PSOs MAPPING

CO 5 : BUILD JAVA APPLICATIONS FOR SOLVING REAL-TIME PROBLEMS.

ABSTRACT	POs MAPPED	PSOs MAPPED
An **Inventory Management System** is a software application designed to efficiently track and manage inventory levels, orders, sales, and deliveries. The system provides tools for adding, updating, deleting, and displaying product records while ensuring real-time inventory monitoring. Developed in **Java**, this application leverages a graphical user interface (GUI) for user-friendly operations and enhances productivity by automating inventory tasks. This system can be used in retail, warehouses, or production industries to minimize inventory errors, optimize stock levels, and streamline operations.	**PO1 -3** **PO2 -3** **PO3 -3** **PO4 -3** **PO5 -3** **PO6 -3** **PO7 -3** **PO8 -3** **PO9 -3** **PO10 -3** **PO11-3** **PO12 -3**	**PSO1 -3** **PSO2 -3** **PSO3 -3**

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	viii
1	**INTRODUCTION**	1
	1.1 Objective	1
	1.2 Overview	1
	1.3 Java Programming concepts	1
2	**PROJECT METHODOLOGY**	2
	2.1 Proposed Work	2
	2.2 Block Diagram	2
3	**MODULE DESCRIPTION**	3
	3.1 User Authentication and Role Management	3
	3.2 Product Management	3
	3.3 Order and Transaction Management	3
	3.4 Reporting and Data Export	3
4	**CONCLUSION & FUTURE SCOPE**	4
	4.1 Conclusion	4
	4.2 Future Scope	4
	APPENDIX A (SOURCE CODE)	5
	APPENDIX B (SCREENSHOTS)	12
	REFERENCES	15

CHAPTER 1

INTRODUCTION

1.1Objective

The objective of the **Inventory Management System** is to provide an efficient, automated solution for businesses to track and manage their inventory. The system aims to streamline processes such as product management, stock level monitoring, order processing, and reporting

1.2Overview

The **Inventory Management System** is a desktop-based application built using **Java** that automates key inventory management tasks for businesses. The system is designed with a **Graphical User Interface (GUI)** using JavaFX or Swing for ease of use, providing features such as product management, sales order processing, inventory updates, and report generation.

1.3 Java Programming Concepts

1) Event Handling: JavaFX or Swing will be used to build the user interface, with event handling mechanisms for user actions like button clicks, text field inputs, and menu selections.

2) Database Connectivity: JDBC (Java Database Connectivity) is used to connect the Inventory Management System to the database.

3) File Handling: The system will allow users to export reports (e.g., sales or inventory data) to **CSV** or **PDF** files.

4) Control Structures: Loops and conditional statements to handle user inputs and system functionality.

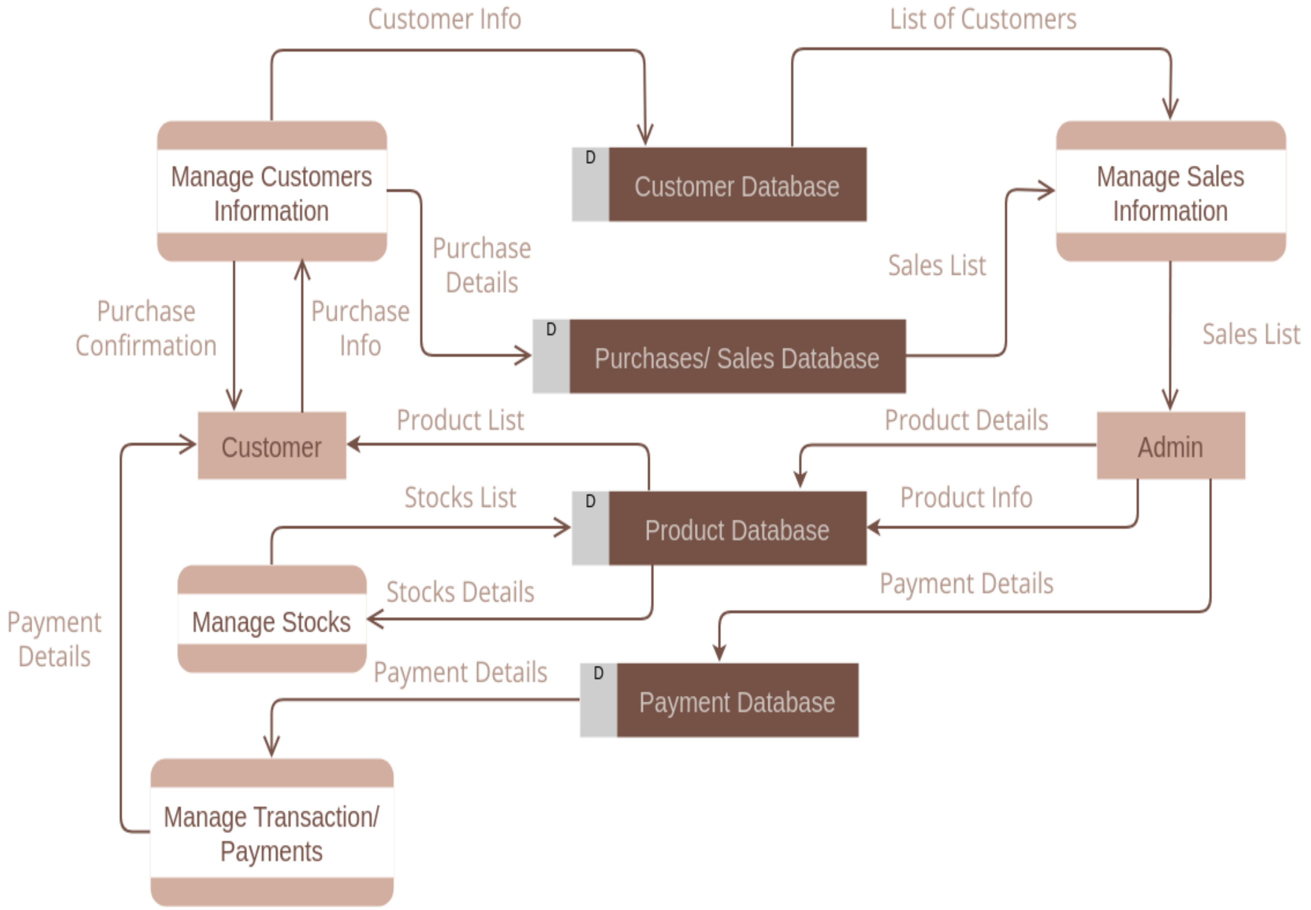
5) Scanner Class: For reading user inputs. 6)Encapsulation: To manage notice data securely through object properties. 7)Method Overriding (toString): To display notice details in a readable format

CHAPTER 2
PROJECT METHODOLOGY

2.1Proposed Work

The **proposed work** for the **Inventory Management System** involves developing a software solution that automates inventory tracking, product management, order processing, and reporting for businesses. The system will be designed using Java with a **Model-View-Controller (MVC)** architecture, where the **frontend** will be developed using JavaFX or Swing for a user-friendly interface, and the **backend** will handle business logic, database operations, and security features. The database will store information related to products, orders, users, and transactions, with CRUD operations performed via **JDBC**.

2.2 Block Diagram



CHAPTER 3
MODULE DESCRIPTION

3.1 User Authentication and Role Management

This module is responsible for managing user access to the system. It includes the user login functionality, where users must enter valid credentials (username and password) to access the system.

3.2 **Product Management**

The **Product Management** module allows users to add, update, view, and delete product information in the system. This includes product details such as name, category, price, quantity

3.3 **Order and Transaction Management**

The **Order and Transaction Management** module handles the creation, tracking, and processing of sales orders. It allows users to create new orders, select products, and calculate total costs based on quantity and price.

3.4 **Reporting and Data Export**

The **Reporting and Data Export** module generates detailed reports based on inventory and sales data. Users can generate reports on stock levels, sales performance, order history, and transactions. Reports can be filtered by date ranges, categories, or product types.

CHAPTER 4
CONCLUSION & FUTURE SCOPE

4.1 CONCLUSION

The **Inventory Management System** successfully automates and streamlines the process of tracking and managing inventory, sales, and orders. By integrating features like real-time stock updates, order processing, product management, and reporting, the system helps businesses improve efficiency, reduce errors, and maintain optimal inventory levels. The user-friendly interface and role-based access ensure secure and easy management for users at various levels, from administrators to salespersons. Overall, this project enhances operational workflows and supports better decision-making through data-driven insights

4.2 FUTURE SCOPE

The future scope of the **Inventory Management System** includes adding advanced features such as **predictive analytics** for inventory forecasting, **integration with e-commerce platforms** for online sales synchronization, and **multi-location support** for businesses with multiple stores or warehouses. Enhancements can also be made to the **mobile interface** to allow users to manage inventory on-the-go. Additionally, incorporating **machine learning** algorithms for demand prediction and dynamic pricing could further optimize inventory management and sales strategies.

APPENDIX A
(SOURCE CODE)

```
import java.awt.*;
import java.awt.event.*;
import java.util.ArrayList;

class Item {
    int id;
    String name;
    int quantity;
    double price;

    public Item(int id, String name, int quantity, double price) {
        this.id = id;
        this.name = name;
        this.quantity = quantity;
        this.price = price;
    }

    @Override
    public String toString() {
        return "ID: " + id + ", Name: " + name + ", Quantity: " + quantity + ",
Price: " + price;
    }
}

public class InventoryManagementSystemAWT extends Frame {
    ArrayList<Item> inventory = new ArrayList<>();
```

```
TextArea displayArea;
TextField idField, nameField, quantityField, priceField;
Label messageLabel;

public InventoryManagementSystemAWT() {
    // Layout setup
    setLayout(new FlowLayout());

    // Title
    Label title = new Label("Inventory Management System");
    title.setFont(new Font("Arial", Font.BOLD, 16));
    add(title);

    // Input fields
    add(new Label("ID:"));
    idField = new TextField(10);
    add(idField);

    add(new Label("Name:"));
    nameField = new TextField(15);
    add(nameField);

    add(new Label("Quantity:"));
    quantityField = new TextField(5);
    add(quantityField);

    add(new Label("Price:"));
    priceField = new TextField(7);
    add(priceField);
```

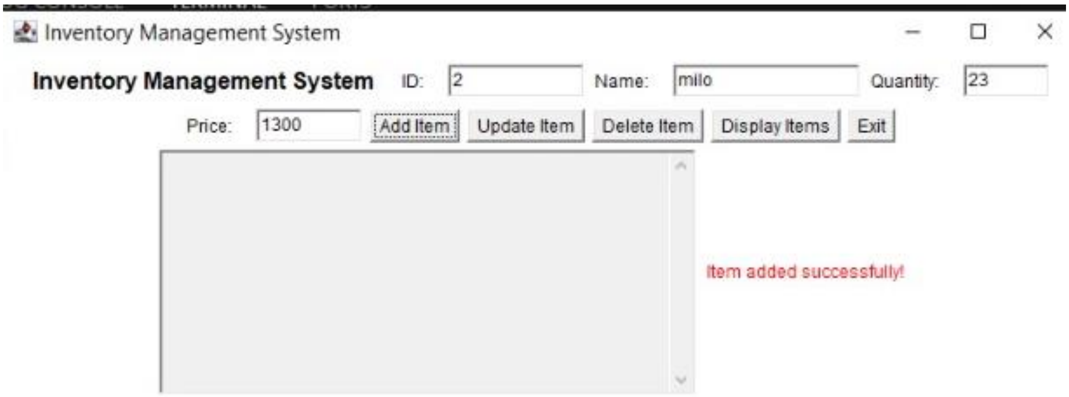
```
// Buttons
Button addButton = new Button("Add Item");
Button updateButton = new Button("Update Item");
Button deleteButton = new Button("Delete Item");
Button displayButton = new Button("Display Items");
Button exitButton = new Button("Exit");

add(addButton);
add(updateButton);
add(deleteButton);
add(displayButton);
add(exitButton);

// Display area
displayArea = new TextArea(10, 50);
displayArea.setEditable(false);
add(displayArea);

// Message label
messageLabel = new Label(" ");
messageLabel.setForeground(Color.RED);
add(messageLabel);

// Button actions
addButton.addActionListener(e -> addItem());
updateButton.addActionListener(e -> updateItem());
deleteButton.addActionListener(e -> deleteItem());
displayButton.addActionListener(e -> displayItems());
```

```
    exitButton.addActionListener(e -> System.exit(0));

    // Frame settings
    setSize(600, 400);
    setTitle("Inventory Management System");
    setVisible(true);

    // Close window action
    addWindowListener(new WindowAdapter() {
        public void windowClosing(WindowEvent e) {
            System.exit(0);
        }
    });
}

private void addItem() {
    try {
        int id = Integer.parseInt(idField.getText());
        String name = nameField.getText();
        int quantity = Integer.parseInt(quantityField.getText());
        double price = Double.parseDouble(priceField.getText());

        inventory.add(new Item(id, name, quantity, price));
        messageLabel.setText("Item added successfully!");
        clearFields();
    } catch (Exception ex) {
        messageLabel.setText("Error: Invalid input!");
    }
}
```

```
private void updateItem() {
    try {
        int id = Integer.parseInt(idField.getText());
        boolean found = false;

        for (Item item : inventory) {
            if (item.id == id) {
                int quantity = Integer.parseInt(quantityField.getText());
                double price = Double.parseDouble(priceField.getText());

                item.quantity = quantity;
                item.price = price;
                messageLabel.setText("Item updated successfully!");
                found = true;
                break;
            }
        }
        if (!found) {
            messageLabel.setText("Item not found!");
        }
        clearFields();
    } catch (Exception ex) {
        messageLabel.setText("Error: Invalid input!");
    }
}

private void deleteItem() {
    try {
        int id = Integer.parseInt(idField.getText());
        boolean found = false;

        for (int i = 0; i < inventory.size(); i++) {
            if (inventory.get(i).id == id) {
                inventory.remove(i);
                messageLabel.setText("Item deleted successfully!");
                found = true;
                break;
            }
        }
        if (!found) {
            messageLabel.setText("Item not found!");
        }
        clearFields();
    } catch (Exception ex) {
        messageLabel.setText("Error: Invalid input!");
    }
}

private void displayItems() {
    if (inventory.isEmpty()) {
        displayArea.setText("No items in the inventory.");
    } else {
        StringBuilder builder = new StringBuilder();
        for (Item item : inventory) {
            builder.append(item).append("\n");
        }
        displayArea.setText(builder.toString());
        }
    }

    private void clearFields() {
        idField.setText("");
        nameField.setText("");
        quantityField.setText("");
        priceField.setText("");
    }

    public static void main(String[] args) {
        new InventoryManagementSystemAWT();
    }
}
```

APPENDIX B

(SCREENSHOTS)



Inventory Management System

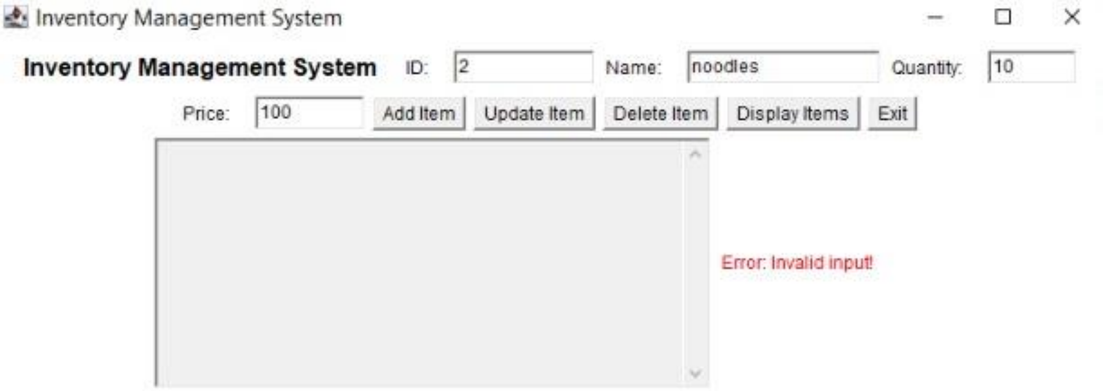
ID: 2

Name: milo

Quantity: 23

Price: 1300

Item added successfully!



Inventory Management System

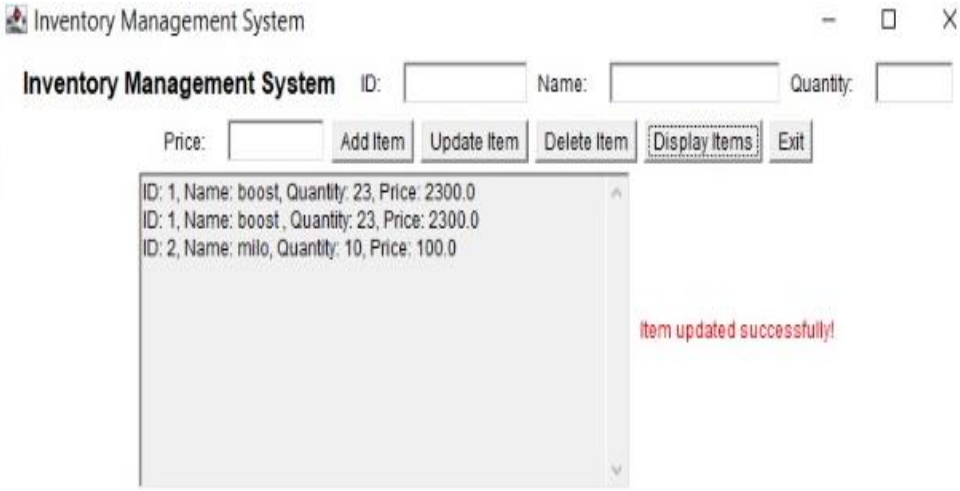
ID: 2

Name: noodles

Quantity: 10

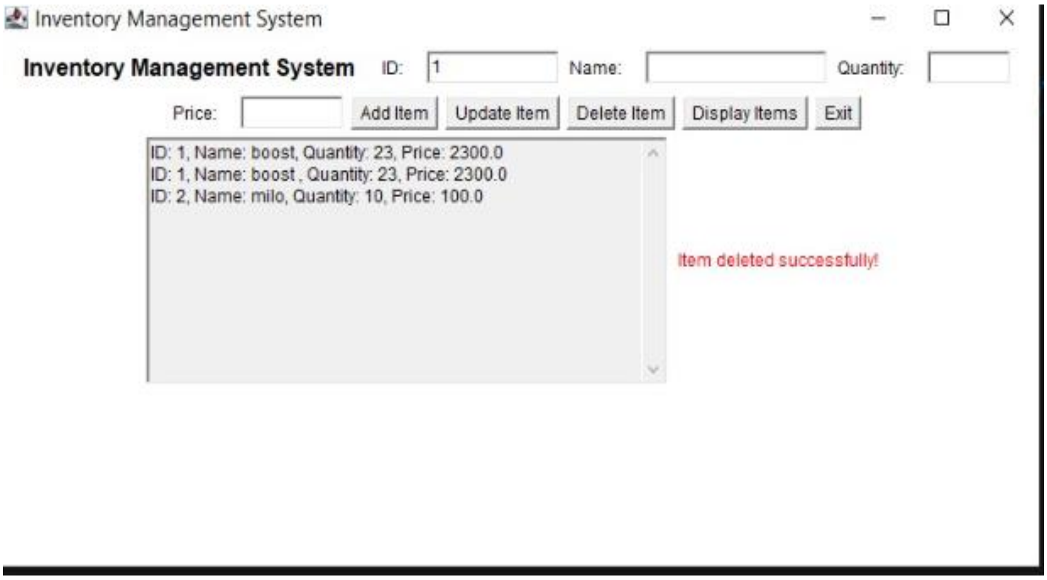
Price: 100

Error: Invalid input!



```
ID: 1, Name: boost, Quantity: 23, Price: 2300.0
ID: 1, Name: boost , Quantity: 23, Price: 2300.0
ID: 2, Name: milo, Quantity: 10, Price: 100.0
```

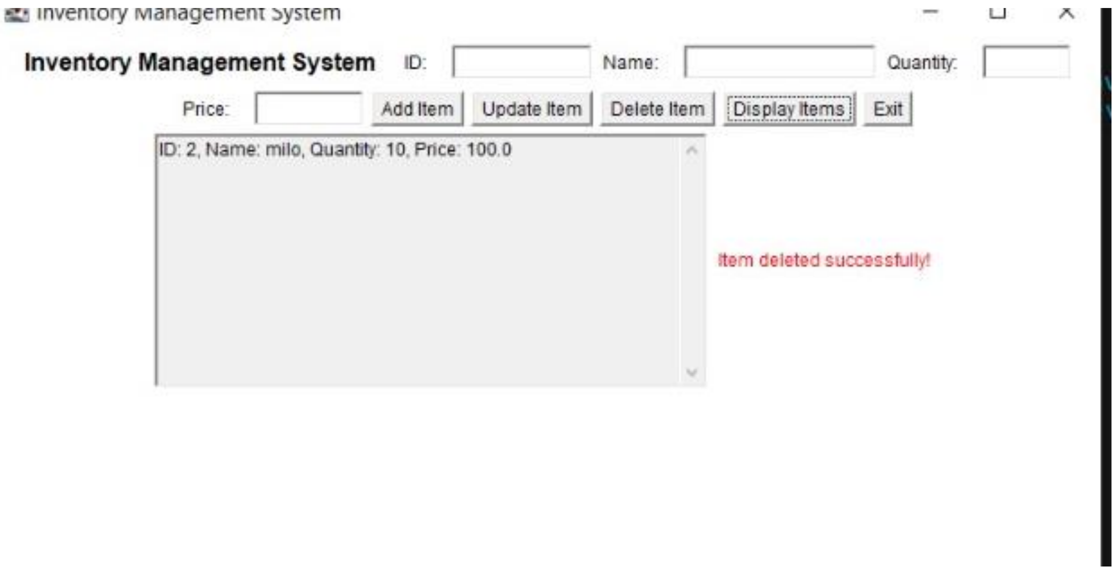
Item updated successfully!



ID: 1

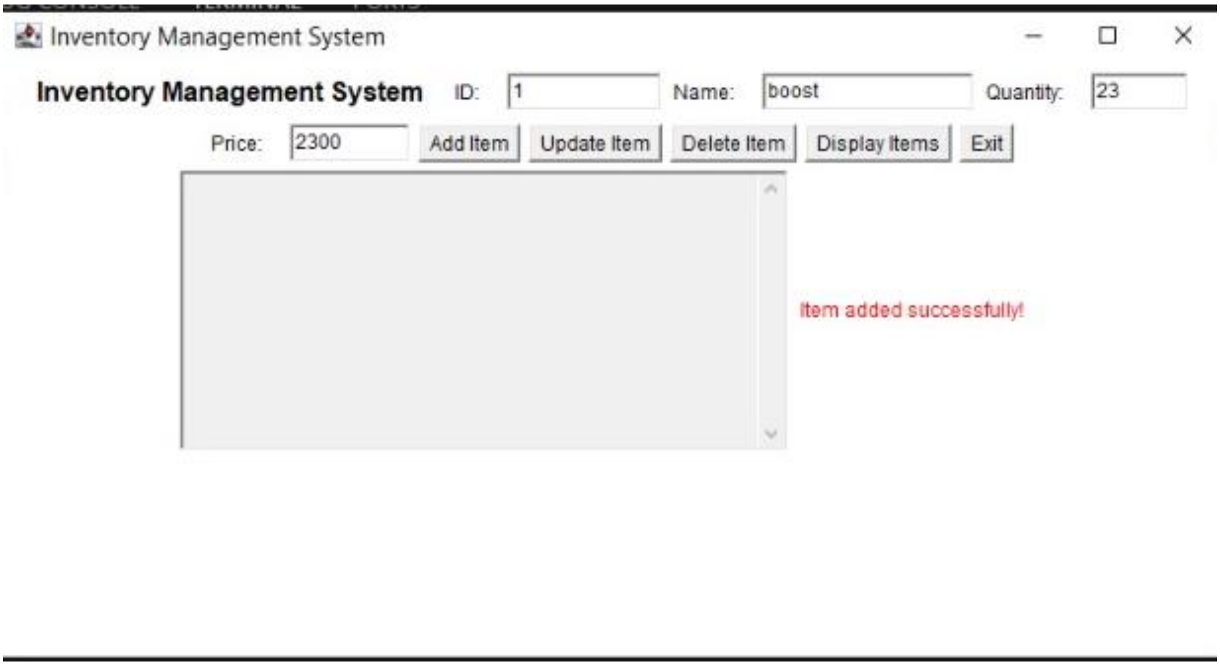
```
ID: 1, Name: boost, Quantity: 23, Price: 2300.0
ID: 1, Name: boost , Quantity: 23, Price: 2300.0
ID: 2, Name: milo, Quantity: 10, Price: 100.0
```

Item deleted successfully!



ID: 2, Name: milo, Quantity: 10, Price: 100.0

Item deleted successfully!



ID: 1 Name: boost Quantity: 23

Price: 2300

Item added successfully!

REFERENCES

1. Kumar, A., & Gupta, A. (2018). *Inventory Management System: A Review*. International Journal of Advanced Research in Computer Science and Software Engineering, 8(4), 120-124.
2. Singh, P., & Sharma, R. (2017). *Design and Implementation of an Efficient Inventory Management System*. Journal of Computer Science and Technology, 32(3), 233-240.
3. Patel, S., & Desai, H. (2019). *A Smart Inventory Management System Using IoT and RFID*. International Journal of Emerging Technologies in Computer Science and Engineering, 6(2), 45-50.
4. Verma, R., & Sharma, P. (2020). *Automated Inventory Control and Management Using Software Application*. International Journal of Advanced Computer Science and Applications, 11(5), 78-82.