PROJECT REPORT

INVENTORY MANAGEMENT SYSTEM

Title Page

Project Title: Inventory Management System

Course: C Programming

Submitted By: Lavish Gupta

GitHub Repository: https://github.com/Lg2411g/Inventoryprojects.git

Submitted To: [Teacher Name]

Department of Computer Science

Academic Year: 2024–2025

Abstract

This project, Inventory Management System, is a menu-driven application developed in the C programming language.

It helps a small shop or store maintain records of products, including their ID, name, quantity, and price.

The system supports adding, updating, deleting, listing, and searching inventory items. Data is stored permanently

using binary file handling. The project follows modular programming concepts, using separate .c and .h files.

This report explains the problem definition, system design, algorithms used, implementation details, test cases, and results.

1. Introduction

Inventory management is essential for every business to track available products, their quantities, and pricing.

Manual maintenance of inventory is time-consuming and error-prone. This project offers a simple automated solution.

2. Problem Definition

A local shopkeeper wants a system that can:

- Store item details

- Add new items

- Modify existing items

- Delete items

- Search any item quickly

- Display all inventory items

- Preserve data even after closing the program

The system must be simple, fast, user■friendly, and reliable.

3. Objectives

- To design an efficient inventory tracking system using C.

- To demonstrate file handling using binary files.

- To implement modular and reusable code using header and source files.

- To show proper documentation and maintainable software practices.

4. Scope of the Project

The system is suitable for:

- Small shops

- Retail stores

- Student mini-projects

The system does not support:

- Multi-user login

- Barcode scanning

- Database integration

5. System Design

Architecture Diagram (Text Representation):

MAIN MENU → Inventory Functions (Add, List, Search, Update, Delete) → Binary File Storage

6. Flowcharts

Add Item Flowchart (Text Form):

Start

→ Enter Item Details

→ Check Duplicate ID

→ If Yes → Show Error

→ If No → Add Item

→ Save Updated List

→ End

7. Algorithms

Add Item Algorithm:

1. Input item details

2. Check if ID exists

3. If yes, display error

4. Else append to list

5. Save to file

Search Algorithm:

1. Input ID

2. Loop list

3. If match, show item

4. Else not found

Update Algorithm:

1. Input ID

2. Search

3. Update fields

4. Save

Delete Algorithm:

1. Input ID

2. Search

3. Shift elements

4. Save

8. Implementation Details

Language: C

Compiler: GCC

Paradigm: Modular Programming

Files Used:

- main.c — Menus

- inventory.c — Logic

- inventory.h — Structure + functions

- data.bin — Binary storage

- Makefile — Build automation

Data Structure:

Item { id, name, quantity, price }

9. Testing & Results

Test Cases:

1. Add Item — Pass

2. Duplicate ID — Pass

3. Search Valid — Pass

4. Search Invalid — Pass

5. Update — Pass

6. Delete — Pass

7. Exit — Pass

10. Output Screens (Text Description)

Example Add Item:

Enter ID: 101

Name: Rice Bag

Quantity: 10

Price: 450

Item Added Successfully.

11. Conclusion

The system successfully manages adding, searching, updating, deleting, and listing inventory items.

Binary file handling ensures data persistence. The modular approach makes the project maintainable and readable.

12. Future Enhancements

- Admin login

- Categories

- Export to Excel/CSV

- GUI support

- Expiry alerts

13. References

- Let Us C — Yashwant Kanetkar

- Online C documentation

- Class lecture notes