NORTH SOUTH UNIVERSITY

PROJECT REPORT

Programming Language I (CSE 115L)

Section: 03

Submitted to

Rejuana Islam - Lab Instructor

North South University

Department of ECE

Submitted by

Name	ID
Abrar Wadud	2531214042
Farhad Hossain Jisan	2533789042
Md. Samsul Arafin	2533949042

Date of Submission: *14/12/2025*

Inventory Management System

Introduction

Managing inventory is essential for small shops. It means keeping track of items, how many there are, and their availability. Effective inventory management helps prevent mistakes and ensures that shops maintain a balanced stock level, avoiding both stockouts and overstocking. This way, resources are used wisely.

Many small businesses use manual methods or simple digital records to keep track of their inventory, but this can lead to confusion and inefficiency. As technology advances, small businesses are seeking user-friendly software solutions. However, many applications are too complicated for their needs. This creates a need for simple systems that focus on basic features without requiring advanced technical skills.

One practical option is a command-line inventory management system. This system utilizes a text-based interface that is compatible with any device and requires minimal system resources. By separating administrative tasks from user browsing, it is simpler to use.

This project aims to create a straightforward command-line Inventory Management System. It will enable both administrators and users to manage and view inventory items easily. This system will be ideal for small businesses, as it demonstrates basic programming and design concepts.

Problem Statement

Many small shops and organizations continue to manage their inventory using outdated methods, such as manual record-keeping or unorganized digital files. These methods often do not provide a straightforward way to organize inventory data, making it challenging to browse, update, and maintain item information efficiently.

Current inventory solutions can be too complex for small businesses, often requiring advanced features that are not user-friendly or intuitive. This highlights the need for a straightforward and user-friendly inventory management system that anyone can utilize without requiring specialized technical skills.

Additionally, having no clear roles in the system can make it more challenging to use. Administrative tasks and basic item browsing are often combined, which can lead to confusion. A well-structured system that separates administrative tasks from general browsing can make inventory handling more transparent and more efficient.

The goal of this project is to create a command-line-based Inventory Management System. This system will have an easy interface for managing and browsing inventory in a small shop. It will allow administrators and users to view, search, add, update, and manage item details effectively. This will help make inventory management more organized, accessible, and efficient.

Objectives of the Program

The main goals of the Inventory Management System project are as follows:

1. To create a simple, text-based system that helps small shops keep track of their inventory.
2. To provide an easy-to-use interface that allows users to manage and browse their inventory items effectively.
3. To allow users to perform key tasks, such as adding new items, updating information, removing items, and viewing details about the products they have.
4. To make searching for items and checking their availability quick and efficient.
5. To organize inventory information in a way that is easy to access and understand.
6. To show how basic programming ideas and logical design can be applied using the C programming language.

Scopes and Limitations

Scopes

- The system is designed to manage the inventory of small shops efficiently using a command-line interface.
- It supports two types of users: administrators and general users, with clearly defined roles.
- Administrators can perform essential inventory operations, including adding, updating, deleting, and viewing items.
- Users can browse and search for items, check availability, and interact with the system through an intuitive menu-driven interface.
- The system provides immediate feedback for incorrect inputs, ensuring a smooth user interaction experience.
- It demonstrates the application of fundamental programming concepts, including file handling, conditional statements, loops, and structured programming, in the C programming language.

Limitations

- The system does not include a graphical user interface; all interactions are text-based.
- It is intended for small-scale inventory and may not be suitable for large businesses or extensive stock management.
- Advanced features such as automated reporting, multi-user concurrency, or networked access are not included.

Project Design

Overview of the Inventory Management System

The Inventory Management System is a user-friendly command-line application created using the C programming language. It is designed to help manage and keep track of inventory items effectively, breaking down complex tasks into simple steps for better understanding and use.

1. System Overview

The system is designed for two types of users: Administrators and General Users. Each user type has different abilities:

- **Administrator:** This user has complete control and can add, update, delete, and view details about inventory items. They can also manage the amount of stock available.
- **User:** This type can browse through items, search for specific products, and check if those products are in stock.

Both user types interact with the system using a simple text menu that guides them through the available options, making it easy to find what they need.

2. Key Components

The system is organized into different parts, making it easier to understand:

- **Main Menu:** This is the first screen users see, where they choose whether they are an Administrator or a General User.
- **Authentication (Optional for Admin):** This feature ensures only authorized users can access administrative functions.
- **Inventory Management:** This section handles all actions related to inventory, including adding new items, viewing existing ones, making updates, and deleting items.
- **Search and Display:** This section enables users to browse, search for, and view details about various items.
- **Feedback and Validation:** This module helps by providing clear messages and prompts, ensuring users enter the correct information, and enhancing their overall

experience.

3. How It Works

The system utilizes simple file handling to save inventory data, ensuring it remains unchanged even after the program is closed. When users select options from the menu, the system processes their commands, performing actions based on their choices. The interaction flows smoothly, allowing users to navigate through tasks until they finish their session.

4. User Interaction

- **Input:** Users make selections by typing numbers on their keyboard.
- **Output:** The system presents information in a clear, formatted way with helpful visuals to make reading easier and more engaging.
- **Error Handling:** If a user enters incorrect information, the system detects the error and displays a message, rather than closing the program.

5. Design Guidelines

The system is built on a few essential principles:

- **Modularity:** Each function is organized by its purpose, making it more straightforward to maintain and test.
- **User-Centric:** The menus and instructions are easy to understand, ensuring that users can navigate without confusion.
- **Resource-Efficient:** As a command-line application, it is designed to use minimal computer resources while effectively demonstrating essential programming concepts.

Overall, the Inventory Management System aims to provide a clear, engaging, and practical approach to managing inventory.

Implementation Details

The Inventory Management System is a software tool created using the C programming language and designed to run on Windows computers. It operates through a simple command-line interface, which means users interact with it by typing commands rather than using a graphical interface. The system is designed to save and retrieve inventory information, allowing users to access their data even after closing the program.

Main Features

The system has different functions depending on the type of user:

- **Administrator:** This role has full access and can add new items, view existing ones,

update details, or remove items from the inventory. Administrators can also set alerts for low stock, manage stock levels, and generate basic reports.
- **User:** Regular users can browse through the inventory, search for specific items, and check if items are in stock. Their options are limited to viewing data only, making it a simple tool to use.

User Interaction

The program is designed to be user-friendly, with menus that provide clear choices based on the tasks users need to complete. Each option in the menu corresponds to a specific job, making navigation straightforward.

Data Management

Inventory details, like the types of items and their quantities, are stored in files. This setup ensures that all entries are saved, allowing administrators to update them easily and users to access the information as needed.

Error Handling and Input Checking

The system includes built-in checks to prevent errors when users enter information. If a user makes an error, the program gives clear feedback, helping them correct it without disrupting their session.

Information Display

To facilitate easy reading and interpretation of inventory data, the program presents information in a clear, organized table format. This organization enables a quick understanding of inventory status at a glance, thereby enhancing overall usability and efficiency.

Results and Discussion

The Inventory Management System was thoroughly tested to ensure it is user-friendly and functions properly. It includes essential features like tracking items, sending alerts, generating reports, and browsing through inventory, all of which operate smoothly. The menu-based setup allows users to navigate easily.

Some minor issues were identified, including errors that occurred when users entered incorrect information. These problems were addressed by implementing checks to prevent mistakes, such as attempting to delete non-existent items or updating information incorrectly.

The system is effective for managing small inventories, helping users complete tasks quickly and accurately. Clearly defining the roles of administrators and regular users made access to different features easier to understand and manage. Overall, the project successfully created a straightforward and user-friendly tool for managing inventory in small shops.

Conclusion

The Inventory Management System project showcases a simple and effective command-line tool designed to help small shops keep track of their inventory. It caters to two primary user types: administrators and regular users. Administrators can easily add, update, or remove items, manage stock levels, and set up alerts for low inventory. Meanwhile, regular users can browse and search for products with ease.

During its development and testing, the system proved to be reliable and user-friendly. It includes features that prevent incorrect entries and handle errors effectively, ensuring a smooth user experience. The application also utilizes file storage to track inventory data, making it easy to save and retrieve information later. The clear formatting of the information displayed helps users read and understand it better.

In summary, this project not only meets its goals of providing a practical and user-friendly inventory management solution but also highlights the importance of effective software design. It demonstrates how careful planning can lead to a program that is both easy to use and maintain.

Future Enhancements

Here are some ideas to make the Inventory Management System even better in future versions:

1. **Better Organization of Data**: Utilizing more efficient methods to organize information can help the system operate more quickly, especially when managing a large number of inventory items.
2. **Compatibility Across Devices**: Ensuring the system works well on various types of computers and operating systems would make it easier for more people to use.
3. **Simplified Data Storage**: Storing inventory information in easy-to-handle formats, such as CSV files, can make management more straightforward. Adding features like automatic saving and backups with timestamps can help prevent the loss of essential data.
4. **User-Friendly Design**: Incorporating a graphical user interface (GUI) would enhance the system's appeal and make it easier for users to navigate.
5. **Inventory Management for Multiple Stores**: Expanding the system to support

inventory tracking for different locations would be beneficial for larger businesses that need to manage stock across multiple shops.

These improvements would not only enhance the system's performance but also make it more user-friendly and versatile for managing inventory.

Reference

- ChatGPT, “Assistance with report writing and structuring,”
- Grammarly, “Grammar and phrasing correction for report content,”
- Google Antigravity, “AI-assisted coding platform for faster programming,”
- GitHub Copilot, “AI-assisted code suggestions and completion,”