**National University of Computer & Emerging Sciences**

**Karachi Campus**



**The Inventory Management System**

**Project Report**

**Programming Fundamentals**

**Section: BCS1-J**

**Group Members**

**24k-0825 Hafsa Asim**

**24k-0914 Shorouq Iqbal**

**24k-0807 Nimra Mehmood**

### **1. Abstract**

The primary objective of this project was to develop a robust **Inventory Management System** capable of handling inventory-related operations like stock management, supplier tracking, and low-stock alerts. However, during the design phase, it became evident that real-world inventory management is closely tied to employee roles and delivery logistics.

To create a more comprehensive and realistic solution, we decided to integrate **Employee Management** and **Delivery Management** modules. This integration allows the system to function as a multi-faceted tool, bridging gaps between inventory, staff, and logistics.

Project link: <https://github.com/rouq-alt/PF-PROJECT>

### **Objectives:**

* Build an efficient **Inventory Management System** as the core module.
* Extend functionality by adding modules for managing employees and deliveries to address interrelated operational needs.
* Provide persistent data storage through file handling.
* Ensure a user-friendly interface for seamless interaction across all modules.

### 

#### **2.1 Inventory Management** (Primary Module)

* **Purpose:** Core functionality of the project, designed to streamline inventory tracking and management.
* **Features:**
  + Display total inventory items.
  + View detailed inventory reports.
  + Search items by supplier or unique item codes.
  + Add new inventory items (admin-only).
  + Update stock levels (admin-only).
  + Highlight low-stock items for replenishment.
* **Rationale for Adding Other Modules:**
  + Inventory operations often involve employees responsible for stock updates and delivery management.
  + Delivery data impacts inventory levels, necessitating tighter integration between these domains.

#### **2.2 Employee Management** (Supporting Module)

* **Purpose:** Track and manage employee details critical for inventory operations.
* **Features:**
  + Add employee records (name, department, salary).
  + View all employee details.
* **Reason for Inclusion:** Employees are integral to inventory management. By maintaining a record of employees, their responsibilities can be streamlined for efficient operations.

#### **2.3 Delivery Management** (Supporting Module)

* **Purpose:** Manage delivery requests and logs to ensure inventory levels are accurately reflected post-delivery.
* **Features:**
  + Log new delivery requests (item name, company, and address).
  + View delivery logs for tracking and audits.
* **Reason for Inclusion:** Deliveries directly impact inventory, whether adding new stock or fulfilling customer orders. The Delivery Management module ensures a seamless flow between inventory and logistics.

### **3. System Architecture**

#### **3.1 Program Structure**

* **Primary Module:** Inventory Management serves as the backbone of the program.
* **Supporting Modules:** Employee and Delivery Management work alongside inventory to provide a holistic operational tool.

#### **3.2 Data Management**

* Persistent storage is achieved using plain text files for each module.
* File operations include reading, writing, and appending data as required.

#### **3.3 Security Features**

* Administrative privileges are enforced for sensitive inventory operations (e.g., adding new items or updating stock).
  + User ID: HSN
  + Password: 990

### **4. Implementation Details**

#### **4.1 Technologies Used**

* **Language:** C Programming
* **File Handling:** Standard I/O functions (fopen, fclose, fprintf, fscanf, etc.).

#### **4.2 Code Highlights**

* Modular design ensures flexibility and maintainability.
* Real-time file synchronization guarantees data consistency.
* Authentication safeguards critical operations.

### **5. Testing and Validation**

The program was tested in the following scenarios:

1. Adding and retrieving inventory items.
2. Searching inventory by supplier and item code.
3. Adding and viewing employee details.
4. Logging and reviewing delivery requests.

**Results:** All modules performed as intended, with smooth transitions and no data loss during file operations.

### **6. Conclusion**

The **Inventory Management System** successfully achieves its intended purpose, with the added **Employee** and **Delivery Management** modules enhancing its functionality. This multi-module approach reflects the interconnected nature of inventory systems in real-world scenarios.

By addressing not just inventory but also the supporting processes of employee oversight and logistics, the system delivers a comprehensive solution.

The functionality of our project is shown below:























