D. Y. Patil College of College of Engineering and Technology, Kolhapur

Department of Computer Science & Engineering

Class: SY-A **Subject: AOOC**

Experiment no: 15

Group No. G29 **Mini Project**

Title of Mini-Project: Fruit Shop Management System

Problem Statement:

In today's digitally-driven world, even small-scale businesses like local fruit shops seek automation and simplified transaction handling to remain competitive and efficient. Manual billing, stock mismanagement, and lack of proper customer tracking often lead to delays, errors, and customer dissatisfaction in traditional fruit stores. These challenges underline the necessity of a lightweight yet robust system that can automate daily operations without requiring advanced technical knowledge from shopkeepers.

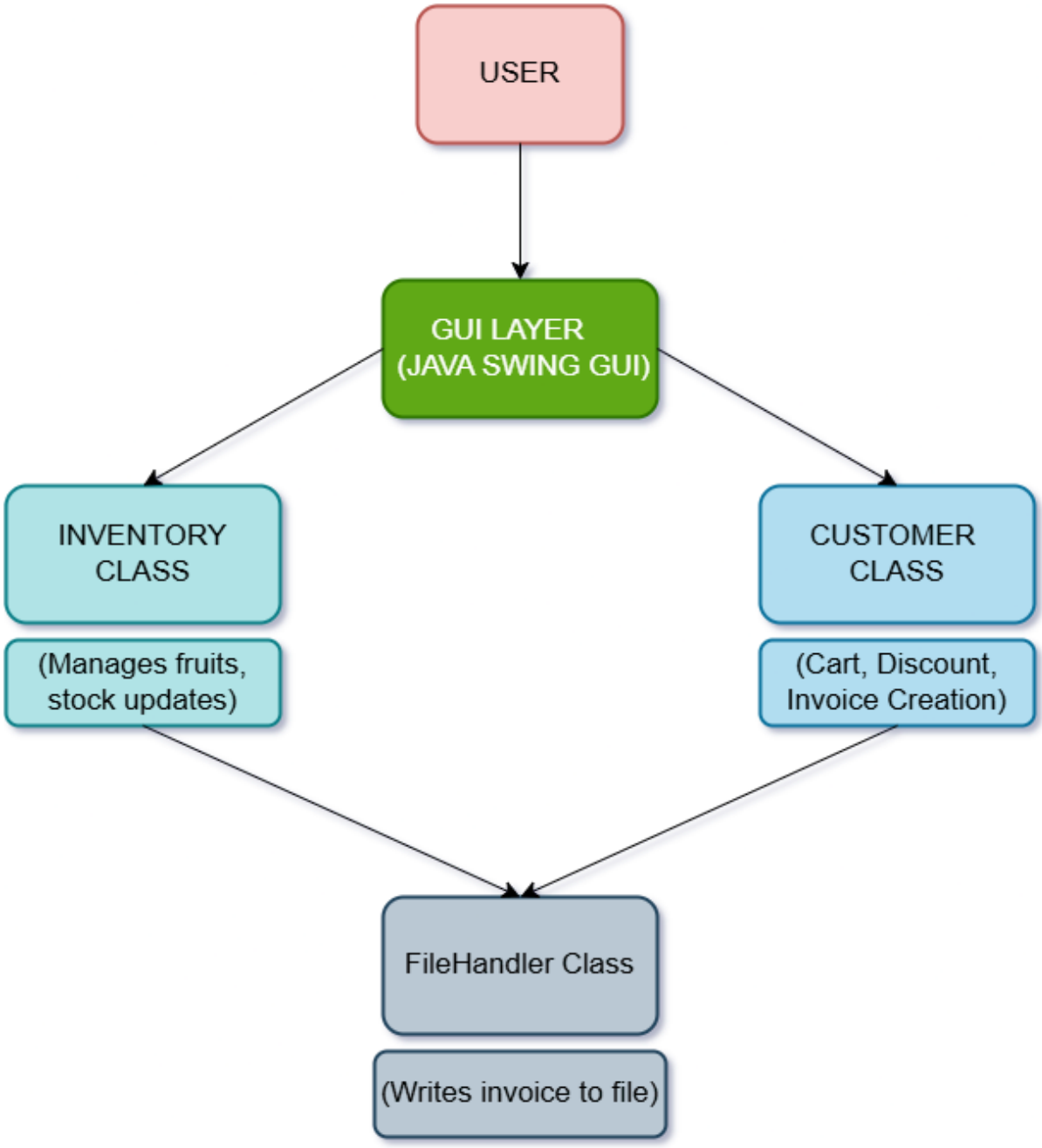
The problem lies in the absence of an affordable and user-friendly application tailored for basic inventory management, customer billing, and sales tracking in a typical fruit shop environment. Shop owners need a solution that reduces human error, manages stock in real-time, applies discounts, generates invoices, and optionally logs transaction data for future analysis — all while being intuitive and easily operable on a basic computer system.

Introduction:

we have developed a Java-based Fruit Shop Management System using object-oriented programming principles, integrated with a GUI built using Java Swing. This system allows shopkeepers to add and sell fruits, track inventory, calculate totals with discounts, generate formatted invoices, and store billing records in a text file. The GUI ensures a user-friendly experience, minimizing the need for command-line interaction. Custom exception handling prevents invalid operations such as over-purchasing items out of stock. Furthermore, the system design promotes modularity and scalability for future enhancements like database integration or mobile app extensions.

This project not only solves a real-world retail issue but also serves as a comprehensive implementation of core Java concepts such as classes and objects, interfaces, exception handling, file I/O, and GUI development — making it a valuable learning experience as well as a practical solution.

System Architecture:



Module description or working of system:

The Fruit Shop Management System is divided into several logically distinct and cohesive modules, each responsible for handling a specific set of functionalities. Below is a detailed overview of each module:

1. GUI Module (User Interface)
   - Technology Used: Java Swing
   - Description: This module serves as the front-end of the application, allowing users (shopkeepers) to interact with the system through buttons, text fields, and selection lists. It collects customer name, fruit choice, and quantity, displays the invoice, and triggers backend operations.
   - Key Features:
     - Dropdown menu for selecting fruits
     - Input field for quantity
     - Display area for invoice
     - Customer name entry
     - Buttons to add items, generate invoice

2. Inventory Module
   - Class: Inventory
   - Description: Manages the stock of fruits available in the shop. It keeps track of fruit quantities, adds new fruits, and updates stock levels after each purchase.
   - Key Functions:
     - addFruit(Fruit fruit)
     - getFruit(String name)
     - updateStock(String name, int quantity)
     - getAllFruits()

3. Customer Module
   - Class: Customer
   - Description: Handles customer-specific actions, such as storing the customer's name and their selected items in a cart, calculating total cost, applying discounts, and generating formatted invoices.
   - Key Functions:
     - addToCart(Fruit fruit, int quantity)
     - generateInvoice()
     - getCart()
   - Implements: Discountable, Billable interfaces

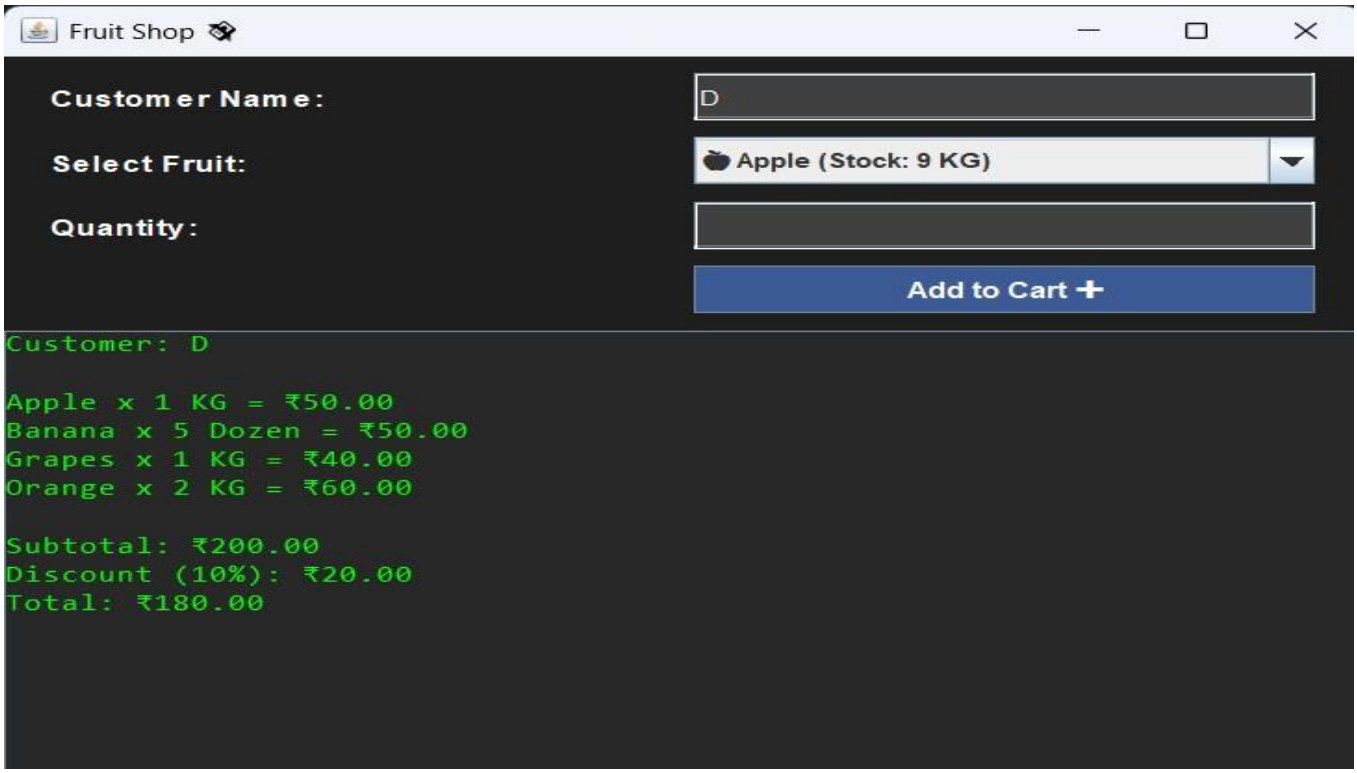
4. Billing & Discount Module
   - Interfaces: Discountable, Billable

- Description: This module provides rules for calculating total amounts and applying discounts based on conditions (e.g., bulk purchase). It's implemented by the Customer class to decouple logic from the UI.
- Key Method:
    - applyDiscount(double total): double
    - generateInvoice(): String

5. Exception Handling Module
    - Class: OutOfStockException
    - Description: Ensures system stability by handling situations where users request more fruits than available in stock. Prevents runtime crashes and improves user feedback.
    - Key Feature:
        - Custom exception triggered on invalid stock requests

6. File Handling Module
    - Class: FileHandler
    - Description: Responsible for logging all generated invoices to a text file (sales.txt) for future reference and analysis. It demonstrates file I/O in Java.
    - Key Method:
        - logSale(String invoice)

Screenshots:



Group Members:

UNIQUE ID	Roll No	Name of Student	Sign
EN23220466	141	Abhishek Devkatte	
EN23217963	143	Anuj Powar	
EN23268977	145	Vijay Kore	
EN23174019	146	Ranjit Dhapate	