Project Proposal: Cafe Management System

Group Members

Muhammad Usama kayani	Muhammad Zain Ul Islam
FA-20-BSSE-133	FA-20-BSSE-087

Introduction

The cafe management system aims to streamline and automate the daily operations of a cafe through a user-friendly mobile application. This project will leverage Java programming language with NetBeans IDE for development, integrating essential Object-Oriented Programming (OOP) concepts to ensure modularity, scalability, and maintainability.

Project Objectives

- Develop a mobile app for cafe management that allows staff to efficiently manage menu items, orders, inventory, and customer interactions.
- Implement a database backend to store and retrieve data related to menu items, orders, customers, employees, and inventory.
- Provide a user-friendly interface for staff to place and manage orders, update inventory levels, and generate reports.
- Enhance operational efficiency by automating routine tasks such as order processing, billing, and inventory management.

Functional Requirements

1. Menu Management

Add, edit, and delete menu items with attributes like name, price, and category.

Categorize menu items (e.g., beverages, snacks, desserts).

2. Order Management

Place new orders, modify existing orders, and mark orders as completed.

Calculate total order amounts including taxes and discounts.

Track order status (e.g., pending, completed, canceled).

3. Inventory Management

Monitor and update inventory levels automatically based on orders placed.

Notify staff when inventory levels are low or items are out of stock.

Manage ingredient quantities for recipes used in menu items.

4. Customer Management

Capture customer information (name, contact details, preferences).

Maintain customer order history and loyalty program details.

Send notifications or promotional offers to registered customers.

5. User Authentication and Authorization

Secure login for staff members with role-based access control.

Admin privileges for managing system configurations and user roles.

OOP Features to Implement

Inheritance: Define superclass/subclass relationships (e.g., Menu Item superclass with subclasses for different menu categories).

Polymorphism: Handle various operations (e.g., order processing, inventory management) based on the type of menu items or customer preferences.

Interfaces: Define interfaces for behaviors such as managing orders, calculating prices, updating inventory, and generating reports.

Encapsulation: Encapsulate data and methods within classes to ensure data integrity and modularity.

Abstraction: Abstract complex functionalities into manageable components, promoting code reusability and maintenance.

Technology Stack

Development Platform: Java SE, NetBeans IDE.

Database: MySQL or SQLite for data storage.

Frameworks/Libraries: JDBC for database connectivity, JavaFX for GUI development.

Tools: Database management tools MySQL Workbench