AP ASSIGNMENT-4

Byte Me! - Enhanced Food Ordering System

ARYAN VASHISHTHA (2023148)

Overview:

This assignment is an enhancement of the previously developed Byte Me! food ordering system, incorporating additional features like GUI, advanced file handling, and JUnit testing. The main goal is to implement user-friendly features, ensure seamless functionality through I/O stream management, and validate operations using unit tests.

Features Implemented:

1. GUI Enhancements

Pages Created:

Menu Page: Displays all available food items with details like name, price, category, and stock and allows adding items to the cart.
Orders Page: Displays all pending orders along with their IDs, statuses, and ordered items.

Components Used:

JFrame, JTable, JScrollPane, JButton, and JLabel from Swing for GUI development. Interactive elements such as buttons for navigation and tables to display data.

Additional Functionalities:

Cart Integration: A dedicated cart page is available to view and modify the cart's contents, and checkout functionality is provided.
Stock Management: Dynamically updates the stock in real-time when items are added to the cart.

2. I/O Stream Management

Files Used:

menu.txt: Stores details of all menu items.
orders.txt: Maintains the record of all orders, including items, quantities, statuses, and order IDs.
cart.txt: Temporarily stores cart details for real-time updates during a session.

Functionalities:

Save Order Histories: Each user's order history is saved to a file, enabling session persistence.
Temporary Cart Storage: The cart.txt file tracks cart updates in real-time and is cleared after successful checkout.
Dynamic Updates: Any menu, cart, or order status modifications are reflected in the respective files immediately.

3. JUnit Testing

Tests Implemented:

Cart Operations:

1. Add items to the cart and verify the total price updates correctly.
2. Modify item quantities and validate the total price recalculations.
3. Prevent setting negative quantities in the cart.

Out-of-Stock Items:

Ensure users cannot add unavailable items to the cart, and an appropriate error message is displayed.

Usage Instructions

Setup:

Extract the zip file and navigate to the project directory.
Compile the project using IntelliJ IDEA or the command line (javac).

Execution:

Run the CLI interface first (ByteMe.main()), followed by the GUI (MainMenuGUI).
The menu.txt and orders.txt files are pre-populated with initialized data.

File Management:

Any operations in the CLI or GUI automatically update the corresponding files.
The cart.txt file is cleared after successful checkout.

JUnit Tests:

Navigate to the test folder.
Run the JUnit tests using IntelliJ IDEA or via the command line with java org.junit.platform.console.ConsoleLauncher.

Key Components:

Feature	Files Involved
Menu Management	MenuPage.java, menu.txt
Cart Operations	CartPage.java, cart.txt
Order Management	OrdersPage.java, orders.txt
I/O Stream Handling	FileHandler.java
JUnit Testing	Testing.java

Assumptions:

1. Either the GUI or the CLI will work at a time.
2. We need to create a text file to store the order history of a customer, and it will store the customer's order history. The predefined naming convention is order_history_username.
3. One can add some items from the GUI to the cart, and the order can be checked out. It will display on the Orders Page as well. However, the order will only be displayed and not created from the code.
