PROJECT REPORT SDA LAB(FOOD ORDERING SYSTEM)-(SE-4B)

BY

ZAUREZ ALAM KHAN(SE-202045)

ARSALAN SADIQ(SE-211075)

AZAN AHMED KHAN(SE-211047)

1. Introduction The Food Ordering System is a Java-based graphical user interface (GUI) application that allows users to order food items from a menu. It provides functionalities for adding items to the order, removing items from the order, displaying the order details, and placing the order. The system incorporates the principles of object-oriented programming and utilizes the Java Swing library for creating the GUI components.
2. Class Structure The system consists of several classes:

* FoodItem: An abstract class representing a food item. It contains attributes such as name and price, as well as methods for displaying item details and removing the item from the order. It also has a reference to the **Order** class.
* Burger and Pizza: Concrete subclasses of **FoodItem** representing specific types of food items (burgers and pizzas, respectively). They override the **displayItemDetails** method to provide specific details for each item.
* Order: A class representing the order. It maintains a list of **FoodItem** objects and provides methods for adding and removing items, calculating the total amount, and displaying the order details.
* FoodOrderingSystemGUI: The main GUI class that extends **JFrame** and implements the **ActionListener** interface. It creates the GUI components, handles user actions, and interacts with the **Order** class.

1. Functionality The Food Ordering System GUI provides the following functionalities:

* Adding Items: Users can select an item from the list and click the "Add Item" button to add it to the order.
* Removing Items: Users can select an item from the order list and click the "Remove Item" button to remove it from the order.
* Displaying Order: Clicking the "Display Order" button shows a dialog box with the order details, including the items, prices, subtotal, tax, and total amount.
* Placing Order: Clicking the "Place Order" button displays the order details and exits the application.

1. **Implementation** The application uses the Java Swing library to create the GUI components. The **JFrame** class is extended to create the main window, and **JList**, **JScrollPane**, **JButton**, and **JOptionPane** are used to create the list of items, scrollable item list, buttons, and dialog boxes, respectively.

The **ActionListener** interface is implemented to handle button click events. Depending on the clicked button, the corresponding action is performed, such as adding an item to the order, removing an item from the order, displaying order details, or placing the order.

The **Order** class maintains a **DefaultListModel** to store the **FoodItem** objects. It provides methods to add and remove items, calculate the total amount, and display the order details using a **StringBuilder**. The order details are shown in a scrollable **JTextArea** within a **JScrollPane**.

1. Sample Execution

Upon running the application, the Food Ordering System GUI window is displayed. The user can select items from the list and add them to the order by clicking the "Add Item" button. The selected item is added to the order, and a confirmation message is shown.

The user can also remove items from the order by selecting an item from the order list and clicking the "Remove Item" button. A confirmation message is displayed upon successful removal.

Clicking the "Display Order" button shows a dialog box containing the order details, including the items, prices, subtotal, tax, and total amount.

Clicking the "Place Order" button displays the order details and exits the application.

1. Conclusion

The Food Ordering System provides a simple and intuitive graphical interface for ordering food items. It demonstrates the use of object-oriented programming concepts such as inheritance, abstraction, and polymorphism. The system can be extended by adding more food items or introducing new features, such as payment processing or user authentication.

OUTPUT



