

The University of Azad Jammu and Kashmir, Muzaffarabad



Project Report programming Fundamentals

Name	Feeza Kulsoom
Roll No	2024-SE-03
Class	SE
Semester	First
Title	Restaurant Management System
Submitted to	Engr. Muhammad Awais
Submission Date	22nd March,2025

Department of SE

Restaurant Management System

Project Report

1. Introduction

The **Restaurant Management System** is a C++-based project designed to streamline restaurant operations, including order management, billing, stock tracking, and discount handling. This system provides an efficient and interactive interface for both customers and restaurant staff. The project focuses on utilizing **functions, arrays, structures, and loops** while maintaining a well-structured and elegant UI with ALT characters.

2. Objectives

The primary objectives of this project are:

- To automate restaurant order processing.
- To manage stock effectively.
- To calculate bills with discounts and taxes.
- To enhance user experience with an interactive UI.
- To ensure efficiency through modular programming.

3. Features

3.1 Menu Management

- Categorized menu with six items each in **Breakfast, Lunch, Dinner, Tea, Seasonal, and Special Menu**.
- Displays menu options for easy selection.

3.2 Order Processing

- Customers can select multiple items.
- Real-time order summary is displayed.

- Receipt generation after order completion.

3.3 Discount & Tax Calculations

- Special discounts for students and regular customers.
- GST applied to final bill.
- Special offer days with additional discounts.

3.4 Stock Management

- Tracks stock levels for each menu item.
- Notifies staff when items are out of stock.

3.5 UI & User Interaction

- Elegant formatting with ALT characters.
- Color-coded interface using **SetConsoleTextAttribute** for better visualization.
- Interactive order management with easy navigation.

3.6 Additional Features

- Order history tracking.
- Customizable orders.
- Real-time order tracking.
- Daily budget tracker.

4. Technical Implementation

The project follows a structured approach with:

- **Data Handling:** Arrays for storing menu items, prices, stock levels, and order history.
- **Control Structures:** Loops (for, while, do-while) and decision-making (if-else, switch).

- **Modularization:** Functions for order processing, billing, discount application, stock updates.
- **Error Handling:** Input validation to ensure correct data entry.
- **Console UI:** ALT characters for structured display and colored text for better readability.

5. Code Structure

- **Main Function:** Controls program flow.
- **Menu Display Function:** Shows available items and prices.
- **Order Processing Function:** Takes user input, updates stock, and calculates total.
- **Billing Function:** Applies discounts, adds taxes, and generates final receipt.
- **Stock Management Function:** Updates and checks available stock.

6. Future Enhancements

- **File Handling:** Save bills for future reference.
- **Menu Modification:** Allow admin to dynamically add or remove menu items.
- **Staff Login:** Secure section for restaurant staff to manage stock and sales reports.
- **Graphical Interface:** Upgrading from console-based UI to a graphical interface.

7. Conclusion

This **Restaurant Management System** provides an efficient solution for managing restaurant operations using **C++ programming fundamentals**. The project successfully integrates order processing, billing, stock tracking, and an interactive user interface. Future enhancements can further improve its functionality, making it an ideal system for small and medium-sized restaurants.

8. Snapshots of project

```
#####  
#           ? THE GREAT FEAST ?           #  
#          *-----*                      #  
#           ? RESTAURANT ?                 #  
#####  
  
Welcome to The Great Feast Restaurant!  
  
%/%/%/%/%/%/%/%/%/%/%/%/%/%/%/%/%&*%$$$$$$#####  
**                                **      MAIN MENU  
%/%/%/%/%/%/%/%/%/%/%/%/%/%/%/%/%&*%$$$$$$#####  
  
1. Display Menu  
2. Order Something  
3. Save Bill  
4. View All Orders History  
5. Exit  
Enter your choice:
```

Fig 1: starting program

```
1. Display Menu
2. Order Something
3. Save Bill
4. View All Orders History
5. Exit
Enter your choice: 1

-----
                Starters
-----
1. Spring Rolls - Rs5.5 (Stock: 10)
2. Garlic Bread - Rs3 (Stock: 15)
3. Nachos - Rs4.5 (Stock: 12)
-----

                Main Course
-----
1. Biryani - Rs8 (Stock: 10)
2. Pasta - Rs7.5 (Stock: 12)
3. Grilled Chicken - Rs10 (Stock: 8)
-----

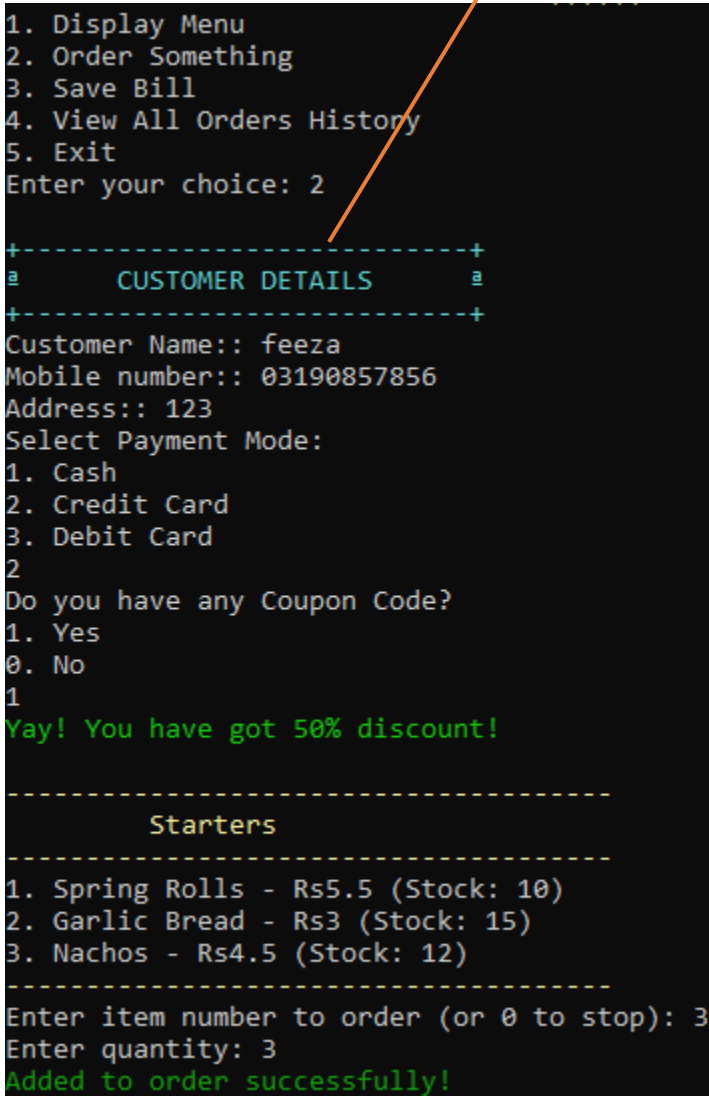
                Desserts
-----
1. Brownie - Rs4 (Stock: 10)
2. Ice Cream - Rs3.5 (Stock: 12)
3. Pudding - Rs4.5 (Stock: 9)
-----

                Beverages
-----
1. Tea - Rs2 (Stock: 20)
2. Coffee - Rs3 (Stock: 18)
```



Fig 2: Displaying menu

Asking customer for details before order



```
1. Display Menu
2. Order Something
3. Save Bill
4. View All Orders History
5. Exit
Enter your choice: 2

+-----+
|          CUSTOMER DETAILS          |
+-----+
Customer Name:: feeza
Mobile number:: 03190857856
Address:: 123
Select Payment Mode:
1. Cash
2. Credit Card
3. Debit Card
2
Do you have any Coupon Code?
1. Yes
0. No
1
Yay! You have got 50% discount!

-----
|          Starters          |
-----
1. Spring Rolls - Rs5.5 (Stock: 10)
2. Garlic Bread - Rs3 (Stock: 15)
3. Nachos - Rs4.5 (Stock: 12)
-----
Enter item number to order (or 0 to stop): 3
Enter quantity: 3
Added to order successfully!
```

Fig 3: Order processing

This is because there is no item no.8

```
Starters
-----
1. Spring Rolls - Rs5.5 (Stock: 10)
2. Garlic Bread - Rs3 (Stock: 15)
3. Nachos - Rs4.5 (Stock: 9)
-----
Enter item number to order (or 0 to stop): 8
Invalid choice! Please select a valid item.
```

Fig 4: Handling invalid inputs

previous stock 10

```
-----
Main Course
-----
1. Biryani - Rs8 (Stock: 10)
2. Pasta - Rs7.5 (Stock: 12)
3. Grilled Chicken - Rs10 (Stock: 8)
-----
Enter item number to order (or 0 to stop): 1
Enter quantity: 10
Added to order successfully!

-----
Main Course
-----
1. Biryani - Rs8 (Stock: 0)
2. Pasta - Rs7.5 (Stock: 12)
3. Grilled Chicken - Rs10 (Stock: 8)
-----
Enter item number to order (or 0 to stop): 0
```

stock updated to 0

Fig 5: Updating stock After each order


```

+-----+
|          CUSTOMER RECEIPT          |
+-----+
Name:: feeza
Mob no:: 03190857856
Address:: 123
Mode of Payment:: Credit Card
Subtotal:: Rs93.5
Discount Applied:: Rs46.75
Tax (8%):: Rs3.74
Total Payable Price:: Rs54.23
Current time:: Sat Mar 22 22:39:30 2025

*****
*          THANK YOU!          *
*    FOR VISITING US!          *
*    SEE YOU AGAIN :)          *
*****

```

Fig 6: Bill generation

Store all orders as a record

```

1. Display Menu
2. Order Something
3. Save Bill
4. View All Orders History
5. Exit
Enter your choice: 4

+-----+
|          ORDER HISTORY          |
+-----+

Order #1 | Date: Sat Mar 22 22:39:30 2025
Customer: feeza
Total Amount: Rs93.5

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%*%*%$$$$$$#####
**                                MAIN MENU                                **
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%*%*%$$$$$$#####

1. Display Menu
2. Order Something
3. Save Bill
4. View All Orders History
5. Exit
Enter your choice: 5
Exiting the program.

```

Fig 7: storing all orders and exiting

Project Type: First-Semester Programming Fundamentals

Programming Language: C++

Development Tool: Dev-C++
