****

**“VP PROJECT”**

**“Shopping Cart”**



**PREPARED By**

**Muhammad Maauz Mansoor (233599)**

**Zahid Zafar (233579)**

**Muhammad Bilal Talib (233543)**

**SUBMITTED TO**

**Aatka Ali**

**DEPARTMENT OF COMPUTER SCIENCE**

**AIR UNIVERSITY MULTAN CAMPUS**

**October 2024**

**“Shopping cart OVERVIEW”**

* **Project purpose :**

The Shopping Cart Application was developed using C# as part of the third semester BSCS course project. This application simulates an online shopping platform where users can browse, add products to their cart, receive product recommendations, and proceed to checkout. The project emphasizes the implementation of object-oriented programming principles and file handling in C#. The system ensures a user-friendly experience and provides various functionalities such as managing cart items, calculating total costs with discounts and sales tax, and tracking previous purchases.

**“DECLARATION”**

**EXPLANATION:**

We declare that this report is an accurate and comprehensive representation of the work done by our team in developing this project using C#. Each member contributed significantly to the development and completion of the project. The report reflects our efforts, from initial conceptualization to final implementation, and we affirm that the contents are original and aligned with the project’s objectives.

**“ACKNOWLEDGMENT”**

**EXPLANATION:**

This message expresses gratitude to various people who helped with the project's success.

* **Gratitude to Allah:** Thanks God for strength and perseverance to complete the project.
* **Appreciation for Parents:** Thanks parents for their unwavering support.
* **Team Acknowledgement:** Expresses gratitude to all team members for their contributions and hard work.
* **Instructor Thanks:** Thanks instructors and mentors for guidance and feedback.
* **General Acknowledgement:** Extends appreciation to friends, classmates, and anyone who indirectly supported the project.

In the name of ALLAH, The Most Beneficent, The Merciful. Alhamdulillah, I am thankful to Allah for granting me the strength and perseverance to complete this project. I would like to extend my heartfelt thanks to my beloved parents for their unwavering support and encouragement throughout this journey. I would also like to express my deepest gratitude to all my team members, who played their roles exceptionally well and contributed immensely to the success of this project. Their dedication, hard work, and collaboration were vital in overcoming the challenges we faced. Furthermore, I am grateful to our instructors and mentors for their invaluable guidance and feedback, which helped shape the direction of our project. Their insights and expertise were instrumental in our learning and development. Additionally, I would like to acknowledge the support of our friends and classmates, who provided motivation and assistance when needed. To those who indirectly contributed to this project, your kindness and support mean a lot to me.

**Thank you very much!**

**“TEAM MEMBERS”**

**(1)**

**MUHAMMAD MAAUZ MANSOOR (233599)**

**WORKING**

* Designed the user interface in C#, focusing on an intuitive layout, clear navigation, and user-friendly design.
* Project Detailed Video Preparation with Voiceover.
* Add products to Cart where user can easily see all products and user can buy with discount.
* Validations, ensuring correct data to enter, retrieve or save in files.
* PPT Slide Contribution in Whole project.

**(2)**

**Zahid Zafar (233579)**

**WORKING**

* PPT Slide Contribution in Whole project.
* View Cart where Users can easily see his own Products that he wanted to buy.
* Delete Cart where User Can Delete Product that he didn’t want to buy in future or now.
* Created user purchase history tracking, allowing users to view their transaction records with product details, quantity, and purchase date.
* Report creation of whole project including Class Diagram and Flowchart.

**(3)**

**Muhammad Bilal Talib (233543)**

**WORKING**

* Developed the login and registration systems, including secure authentication mechanisms and new user creation.
* Added personalized product recommendation logic based on cart contents to enhance user engagement.
* Quantity Management Function creation with all remaining products will show after checkout.
* Integrated file handling for all user operations, ensuring data persistence and record-keeping.
* Implemented checkout functionality, calculating total costs, discounts, and sales tax for a streamlined payment process also get data from user for shipment and payment.
* PPT Slide Contribution in Whole project.

**“ Class structureS (functionalities)”**

### The Shopping Cart Application is built using C# and follows object-oriented programming principles to

### provide a modular and efficient structure.

### **1. PRODUCT CLASS**

The User class manages user-related data and basic operations such as account balance, transaction management, and user authentication.

#### **Attributes:**

* Getprice, setprice, getname, setname getter setters
* Name, price

#### **Methods: (FUNCTIONS)**

* **Setters and Getters:** Manage the setting and retrieving of user data.
* **Login and Verification:** Ensures secure user login by verifying ID and PIN using C# string handling and file reading.

### **2. Shopping Cart Class**

#### The Shopping Cart class handles the overall functionality of managing products, cart operations, and checkout processes, using C#'s file handling and object-oriented capabilities.

#### **Methods: (FUNCTIONS)**

* **Add Product to Cart:** Add items to the cart after validating stock availability using C#.
* **View Cart:** Displays the current items in the user’s cart.
* **Remove Expired Items:** Automatically removes products that have expired in the cart.
* **Checkout:** Manages the checkout process, applying discounts and taxes, and collects payment information.
* **Show Recommended Products:** Suggests products based on the user’s cart contents.

**“Interaction and Workflow”**

### **User Authentication:**

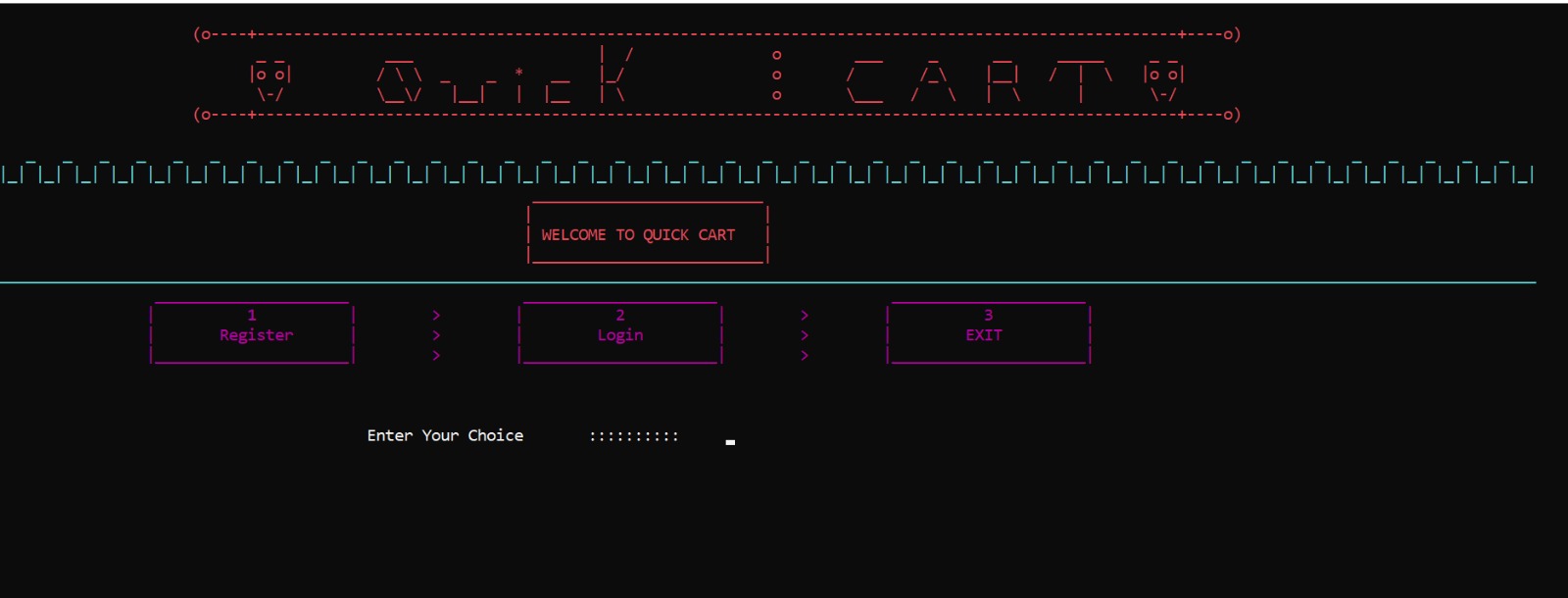
**Login and Authentication:** Users log in by entering their ID and PIN, which are verified against the stored data using C#'s string comparison methods.

### **User Operations:**

Authenticated users can:

* **Cart Management:** Users can browse products, add them to their cart, view recommendations, and remove items using the methods implemented in C#.
* **Checkout:** Users proceed to checkout, where discounts and sales tax are applied to the total amount.
* **Transaction Records:** Users can view their checkout history and review past purchases, all managed through C#'s file handling operations.

**“MAIN MENU (OUTPUT)”**

****

**EXPLANATION:**

When the program starts, it displays a Main Menu with three primary options: Register, Login, and Exit. The user can choose to register if they are a new user, log in if they are already registered, or exit the application. The system remains in this menu until the user selects either login or exit. If an invalid option is selected, the program prompts the user to try again. This interface is built using C# to handle user input and display options in a loop.

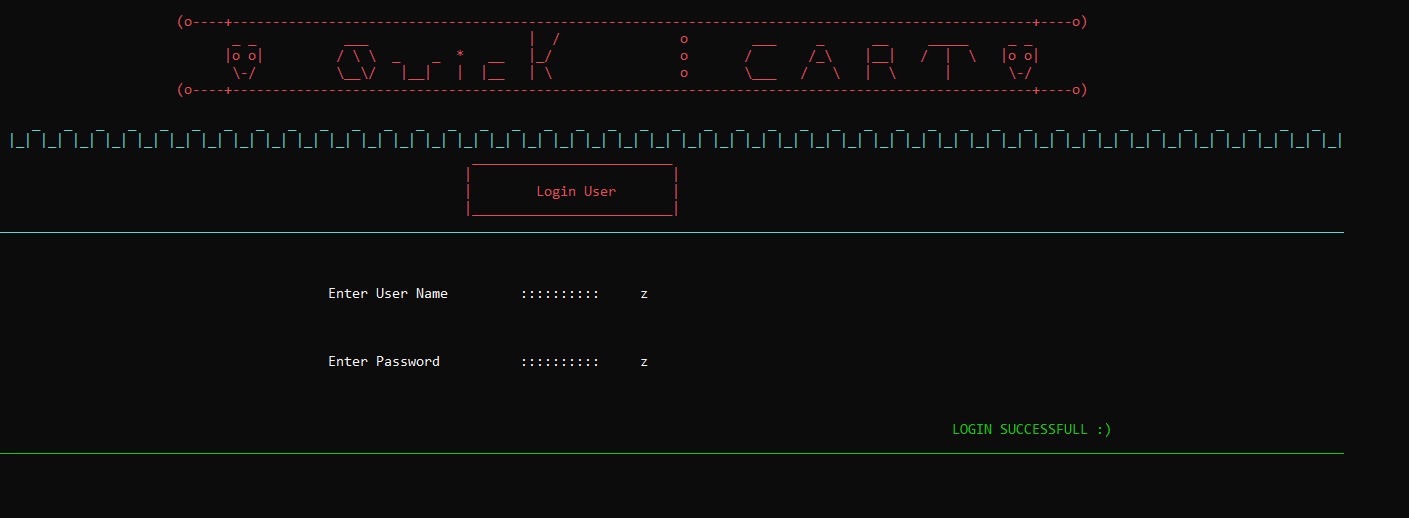
**Key Features:**

**New User Registration:** Takes the user to the registration form to create a new account.

**Login:** Takes the user to the login form where they can access their cart.

**Exit:** Terminates the program and clears the screen.

**“USER login (OUTPUT)”**



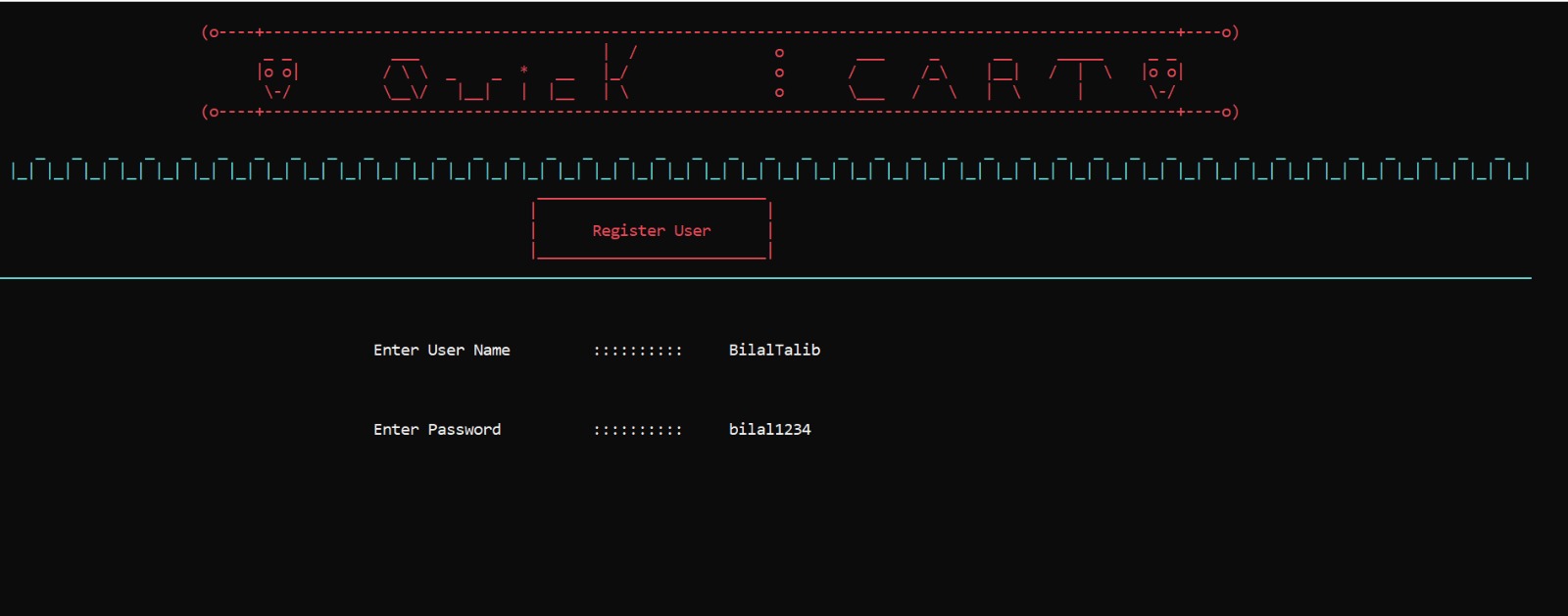
**EXPLANATION:**

**When the user selects the login option, they are prompted to enter their username and password. If the username and password match the stored data, the system grants access to the user's session. If the credentials are incorrect, the system informs the user with an "Invalid username or password" message. After three failed login attempts, the system will lock the account and prompt the user to recover their account using their CNIC.**

**Key Features:**

* **Login Verification:** Compares the entered credentials against the stored user data.
* **Account Locking:** After three failed attempts, the user is prompted for their CNIC for recovery.
* **Masked Input:** Passwords are hidden while being typed for security purposes.

**“New User Registration (Output) ”**



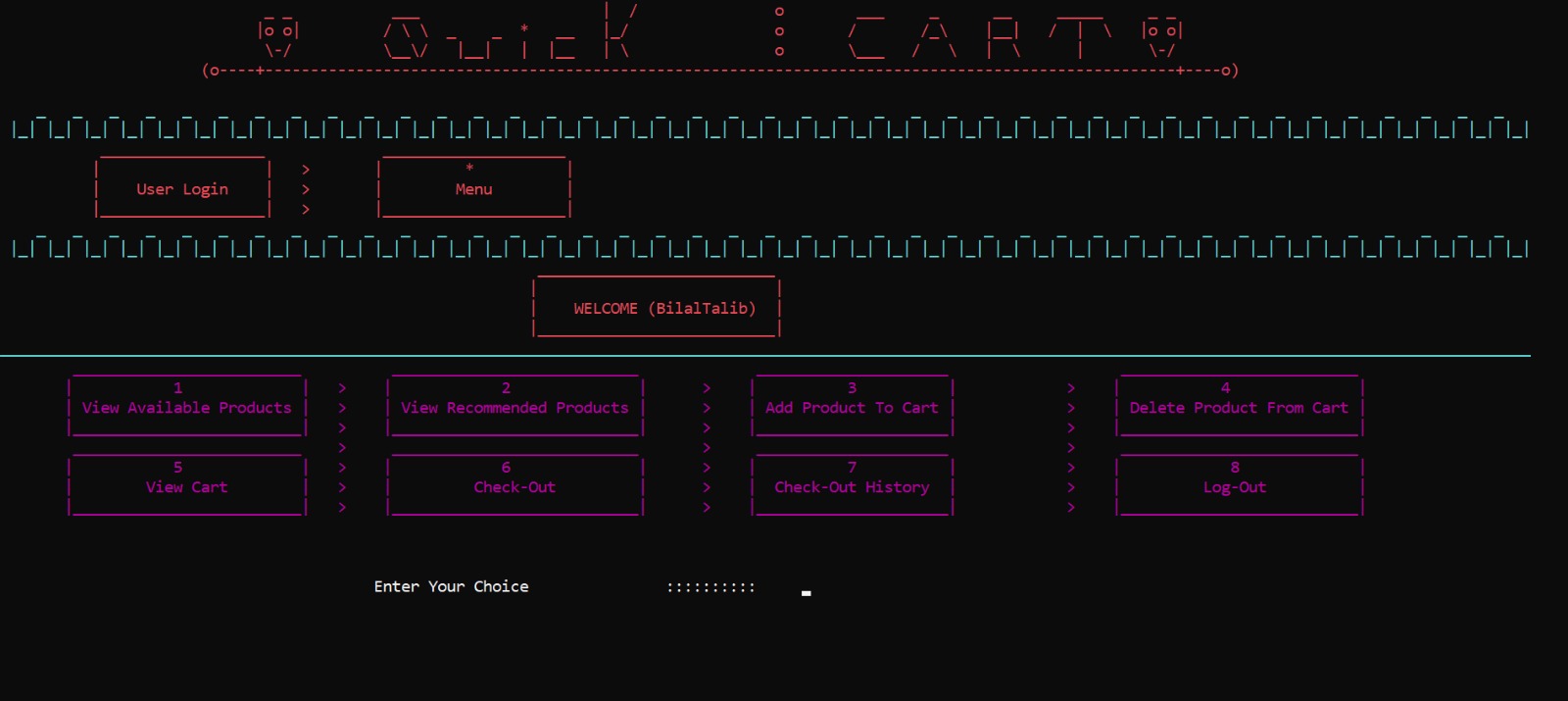
**EXPLANATION:**

**When an admin selects the "New User Registration" option, they are prompted to enter the new user's personal and account details, including username, password, CNIC, phone number, and initial deposit. After all details are entered, the system creates the account and confirms the successful registration.**

**Key Features:**

* **User Registration: Admins can register new users by entering required details.**
* **Account Creation Confirmation: After successful registration, the system displays a confirmation message.**

**“USER functions (OUTPUT)”**

****

**EXPLANATION:**

**Once logged in, the user is presented with a menu that provides various shopping-related functions. The user can:**

* + **View Available Products: Displays a list of products with their prices and available quantities.**
  + **View Cart: Shows the current items in the user’s cart.**
  + **Add Product to Cart: Allows users to add products to their cart by specifying the product name and quantity.**
  + **Delete Product from Cart: Lets users remove products from their cart.**
  + **Checkout: Proceeds to the checkout process where total price, discounts, and taxes are calculated.**
  + **View Purchase History: Displays the user’s past purchases.**
  + **Logout: Logs the user out and returns them to the main menu.**

**“Add Product to Cart (Output)”**

****

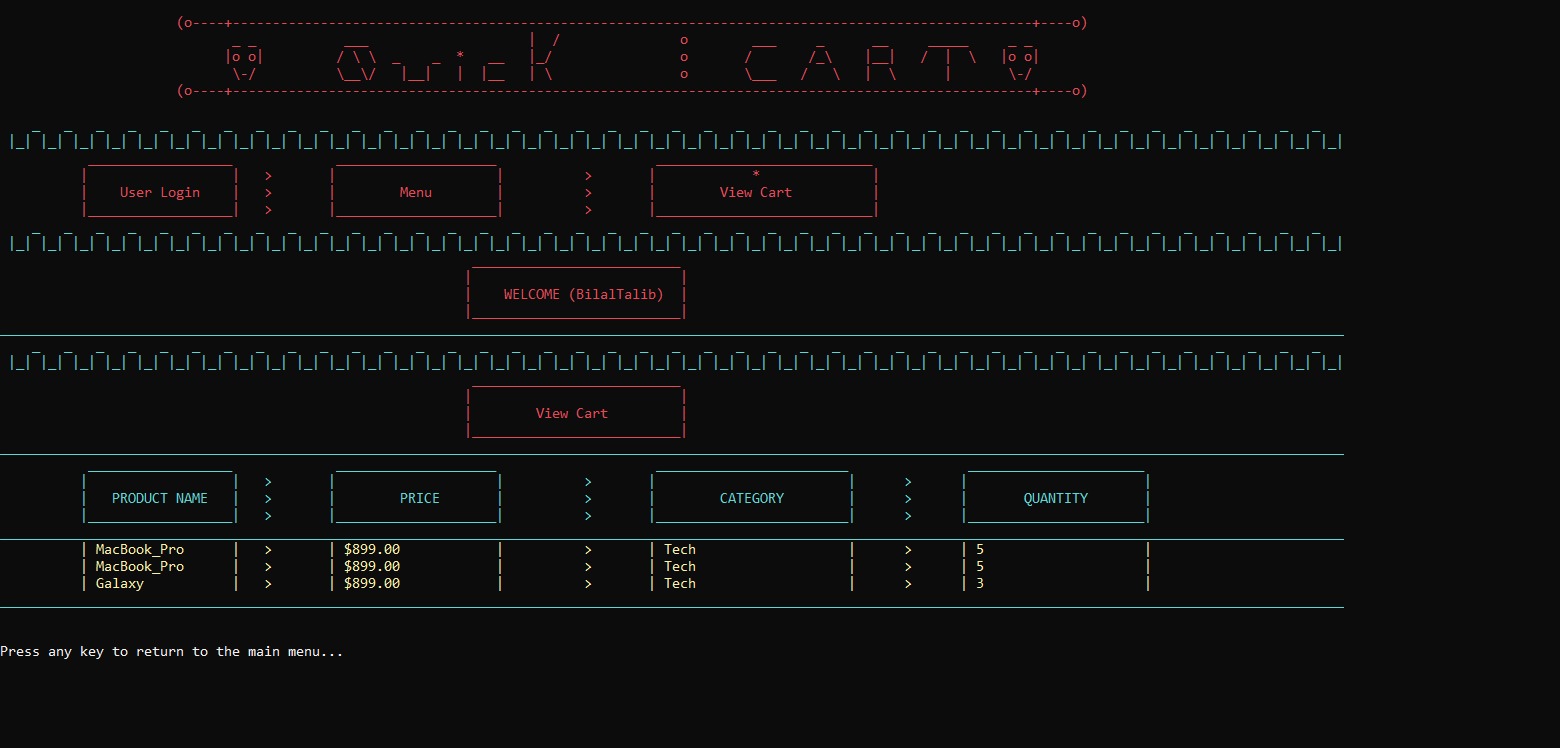
**EXPLANATION:**

**When the user selects the "Add Product to Cart" option, they are prompted to enter the product name and quantity. The system checks the availability of the product and verifies if the requested quantity is in stock. If available, the product is added to the cart, and the user is notified. If the requested quantity exceeds stock availability, the user is informed of the available quantity.**

**Key Features:**

* **Product Availability Check:** Verifies if the requested product is in stock.
* **Stock Validation:** Ensures that users cannot add more products than what is available.

**“View Cart (Output)”**

****

**EXPLANATION:**

**When the user selects the "View Cart" option, the system displays all items currently in the cart, along with their prices, quantities, and the total amount for each product. If the cart is empty, the system informs the user that there are no items in the cart.**

**Key Features:**

* **Cart Summary: Displays the list of products, their prices, and total amounts.**
* **Empty Cart Handling: If the cart is empty, a message is shown to inform the user.**

**“Recommended Products (Output)”**

****

**EXPLANATION:**

**Selecting "Recommended Product" allows the system to generate personalized product suggestions for each user based on their interactions and preferences. Each recommendation includes product ID, name, category, and reason for recommendation. If no recommendations are available, a message indicates this.**

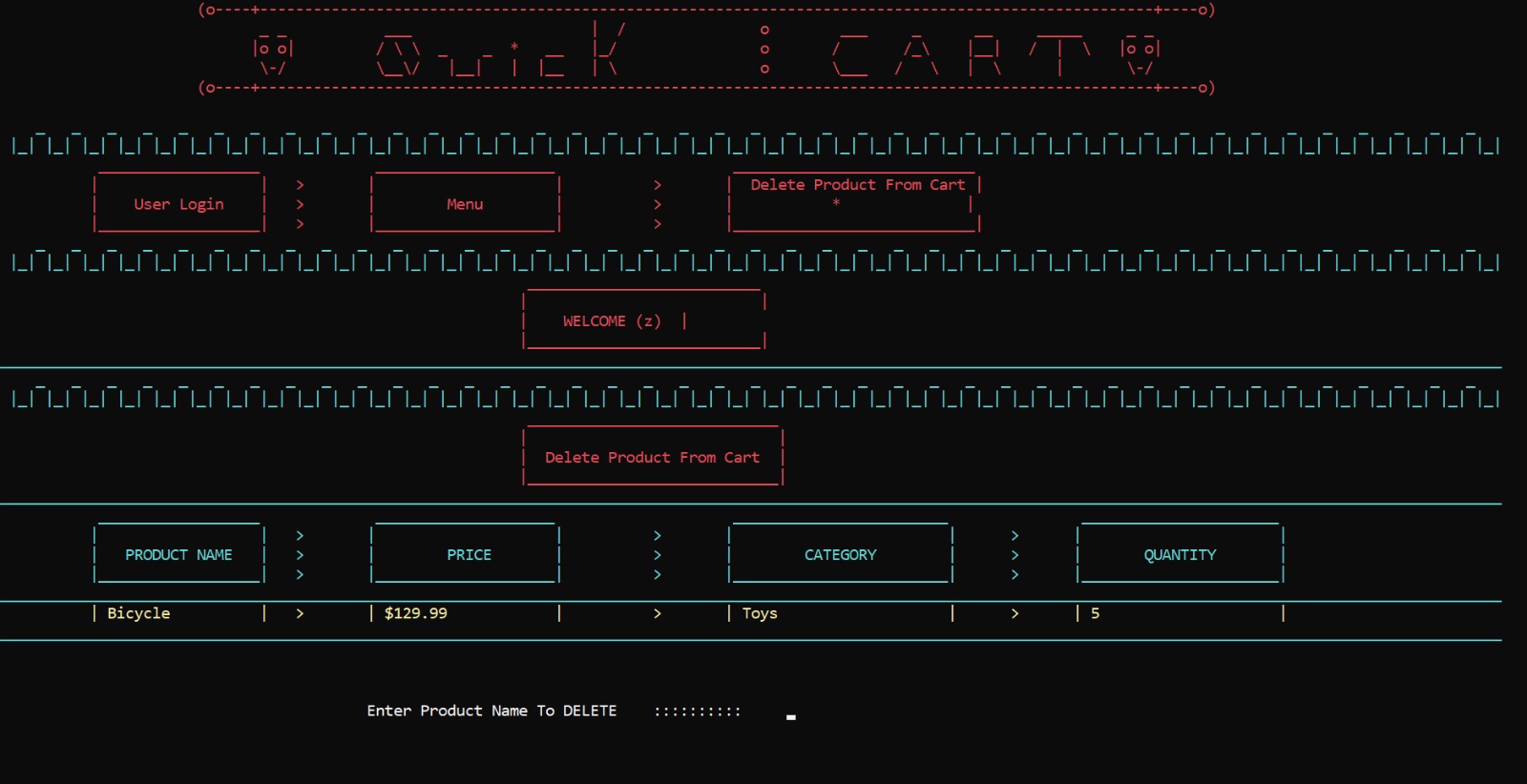
**Key Features:**

**Personalized Recommendations: Customized suggestions based on user data.**

**Detailed Info Display: Shows product ID, name, category, and recommendation reason.**

**No Recommendation Handling: Displays a message if no recommendations are found.**

**“Delete Product from Cart (Output)”**

****

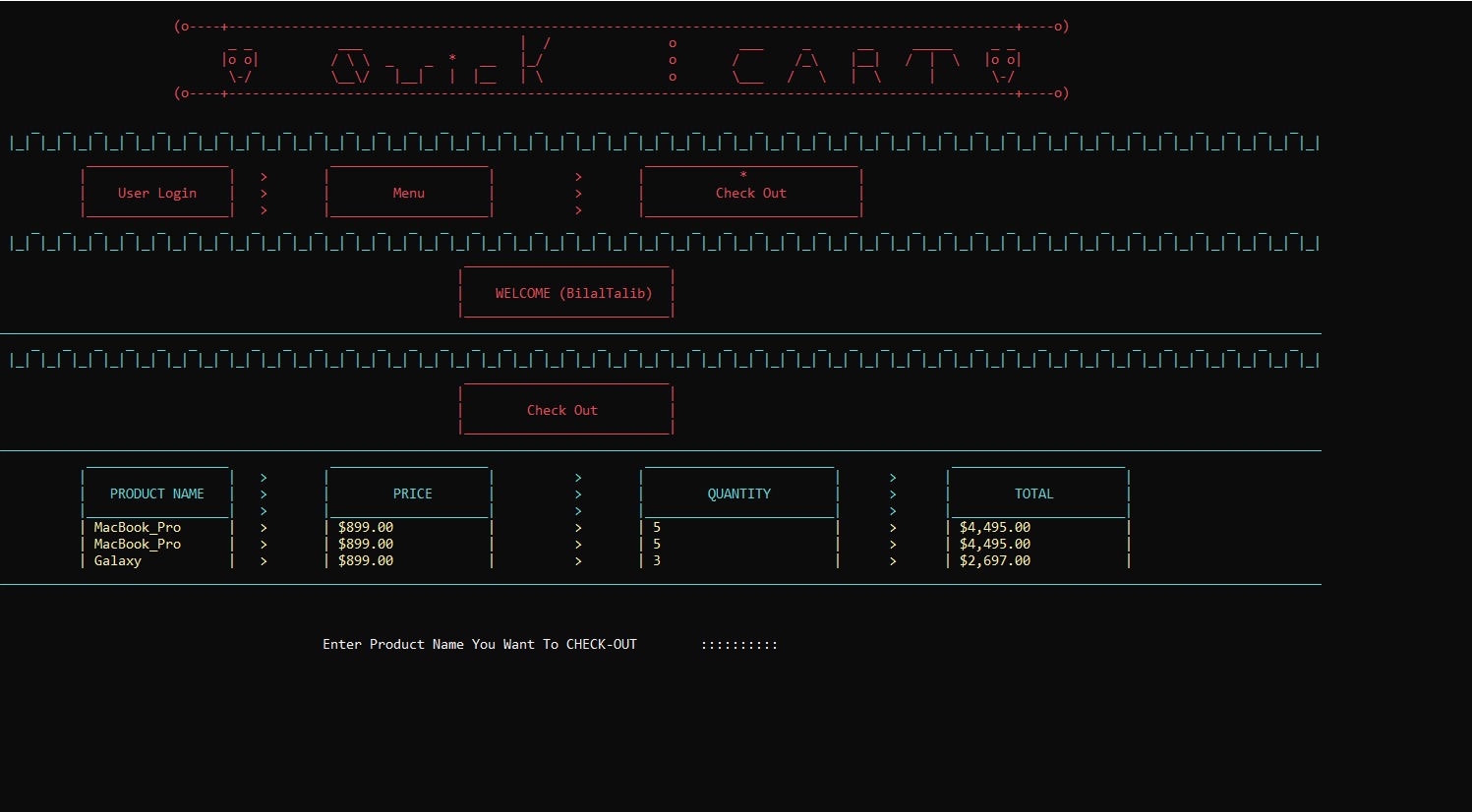
**EXPLANATION:**

**When the user selects the "Delete Product from Cart" option, they are prompted to enter the name of the product they wish to remove. The system checks if the product is in the cart, and if found, it removes the item. If the product is not in the cart, the user is informed.**

**Key Features:**

* **Product Removal: Allows users to remove specific products from their cart.**
* **Error Handling:** **If the product is not found, the system displays an appropriate message.**

**“Checkout Process (Output)”**

****

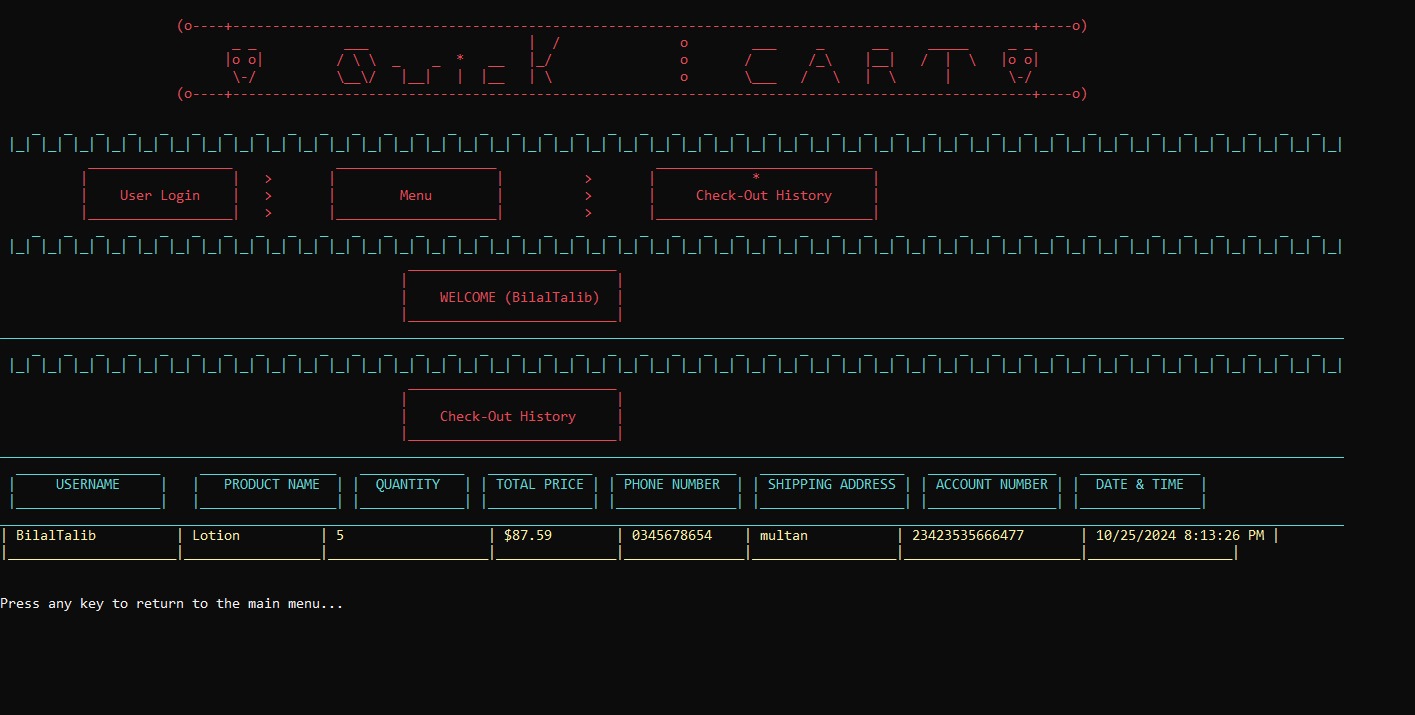
**EXPLANATION:**

**When the user proceeds to checkout, the system calculates the total price of the items in the cart. If the user is purchasing 5 or more units of a product, a 10% discount is applied. A 2.5% sales tax is then added to the final total. The user is prompted to enter their phone number, shipping address, and payment information. After successful payment, the order is confirmed, and the user’s cart is cleared.**

**Key Features:**

* **Discount Calculation:** **A 10% discount is applied for bulk purchases (5 or more units).**
* **Sales Tax Addition:** **A 2.5% sales tax is applied to the total amount.**
* **Order Confirmation:** After successful payment, the user is shown an order confirmation message.

**“Purchase History (Output)”**

****

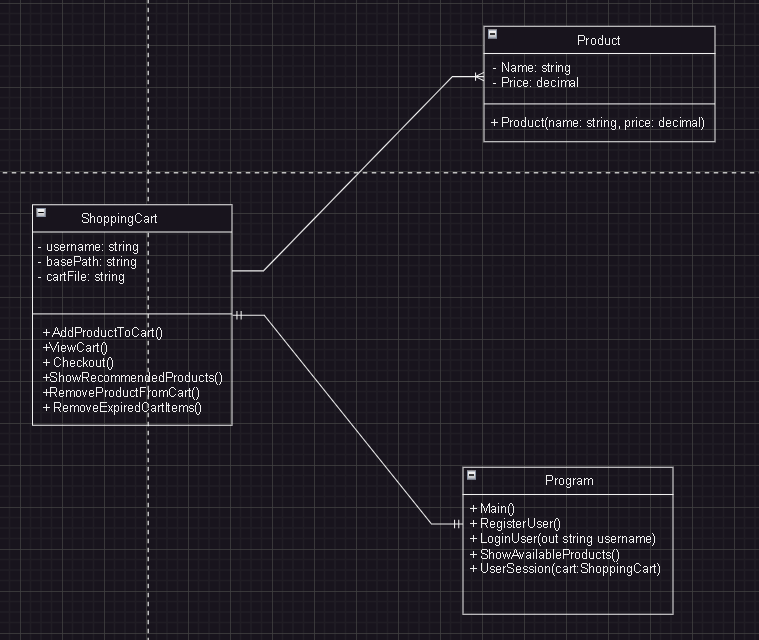
**EXPLANATION:**

**When the user selects the "View Purchase History" option, the system retrieves and displays all previous purchases. Each record includes the product name, quantity, total amount, and the date of the transaction. If no previous purchases are found, the system informs the user.**

**Key Features:**

* **Transaction Retrieval:** **Displays all past purchases.**
* **Empty History Handling:** **If no purchases are found, an appropriate message is shown.**

**“Uml diagram”**



**“Flow Chart”**

**USER**

**REGISTER**

**LOGIN**

**Tech**

**View all Products**

**Cosmetics**

**Fashion**

**View Recommended Products**

**Toys**

**Product Name**

**Add to Cart**

**Quantity**

**Remove Products From Cart**

**Checkout**

**Checkout History**

**Checkout Detail View**

**“Conclusion”**

**EXPLANATION:**

This project successfully demonstrates the use of C# for developing an e-commerce-style shopping cart system. It implements object-oriented programming principles, file handling, and secure user authentication. The Shopping Cart Application simulates a basic e-commerce platform, providing users with functionalities like cart management, product recommendations, and a streamlined checkout process. Admin functionalities such as user management and record viewing enhance the system's administrative control.

The project achieves its objectives of practicing C#-based file handling, object-oriented design, and the development of a user-friendly interface. Moreover, the project can be further expanded to include additional features like:

**Advanced Search Filters:** Allow users to filter products based on categories, price ranges, and popularity

**Encrypted User Authentication:** Use encryption for secure user credentials and sensitive data management.

**Database Integration:** Replace file handling with database management systems to improve scalability and data integrity.

**Mobile App Integration:** A mobile version of the shopping system can enhance accessibility for users on different devices.

The implementation of this project using C# has allowed us to develop a robust and functional shopping platform while also reinforcing important software development concepts like encapsulation, inheritance, and modular design. The final system provides an excellent foundation for further enhancements and can be adapted for larger-scale e-commerce applications.

The End !