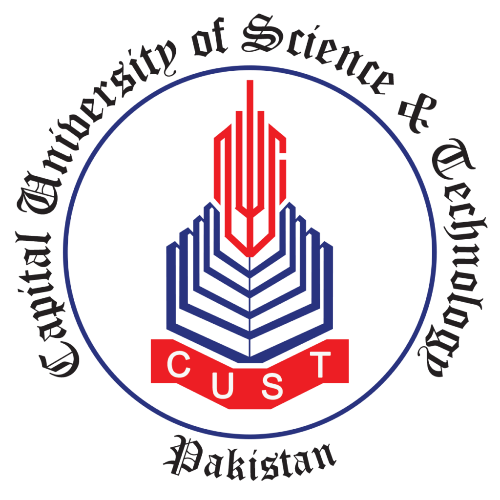
****

**Shahsawar (BSE243164)**

**Sultan Khan (BSE243176)**

**Najeeb Sultan (BSE243162)**

**Huzaifa (BSE243156)**

**ITP PROJECT: ATM CASH WITHDRAWL SYSTEM**

**Introduction**

The **NATIONAL BANK ATM System** is a C++-based console application designed to simulate basic ATM functionalities. This system ensures secure access, provides user-friendly navigation, and performs essential banking operations such as money transfer, cash withdrawal, and balance inquiry.

**Objectives**

The primary objectives of this project are:

* To create a functional and secure ATM system for demonstration purposes.
* To provide a clear and structured interface for users to perform banking tasks.
* To implement error handling for invalid inputs and insufficient funds.

**Features**

1. **Password Protection**:  
   Ensures only authorized users can access the system by requiring a valid password at login.
2. **Menu Options**:  
   A well-structured menu allows users to select from the following options:
   * Transfer Money
   * Withdraw Money
   * Balance Inquiry
   * Exit
3. **Transaction Management**:
   * **Transfer Money**: Enables users to send money to a specified account by selecting a predefined or custom amount. The system validates the recipient's account number and checks available balance before proceeding.
   * **Withdraw Money**: Users can withdraw cash by selecting predefined denominations or entering a custom amount.
   * **Balance Inquiry**: Displays the current account balance in real-time.

**System Workflow**

1. **Login Process**:
   * The user is prompted to enter a password.
   * Access is granted upon entering the correct password (1234); otherwise, the user is prompted to try again.
2. **Main Menu**:  
   After successful login, the main menu is displayed, allowing users to choose their desired operation.
3. **Transactions**:
   * The system dynamically updates the total balance after each transaction.
   * For transfers and withdrawals, the system ensures sufficient funds before processing.
4. **Exit**:  
   Selecting the exit option terminates the session and displays a farewell message.

**Technical Specifications**

* **Language**: C++
* **Core Components**:
  + Conditional statements for flow control.
  + Functions for modularity (e.g., menu display, withdrawal options).
  + Loops for password validation and menu navigation.
  + Input/output handling for user interaction.

**Benefits**

* **Educational Value**: Demonstrates programming concepts such as modular design, loops, and decision-making.
* **Scalability**: The system can be extended to include additional banking features like deposit, mini-statements, or multi-language support.
* **User-Focused**: Provides clear guidance and feedback, ensuring a seamless user experience.

**Conclusion**

The NATIONAL BANK ATM System serves as a simplified yet effective simulation of an ATM interface. It provides a secure, interactive, and reliable platform for users to perform basic banking operations. The project demonstrates a practical application of C++ programming and can serve as a foundation for more advanced systems.

## Code

## #include<iostream>

## using namespace std;

## // Function to display the menu

## void displayMenu() {

## cout << "\t\t\tWelcome to NATIONAL BANK\n";

## cout << "\t\t\tAccess Granted\n";

## cout << "\t\t\t\*Menu\*\n";

## cout << "\t\t\t1. Transfer Money\n";

## cout << "\t\t\t2. Withdraw\n";

## cout << "\t\t\t3. Balance Inquiry\n";

## cout << "\t\t\t4. Exit\n";

## cout << "\tPlease choose an option: ";

## }

## // Function to display withdrawal options

## void displayWithdrawalOptions() {

## cout << "How much money do you want to withdraw?\n";

## cout << "1. 500\n";

## cout << "2. 1000\n";

## cout << "3. 2000\n";

## cout << "4. 5000\n";

## cout << "5. Other amount\n";

## }

## int main() {

## int option, withdrawalOption, amount;

## int totalAmount = 10000;

## int password;

## cout << "Enter the password: ";

## cin >> password;

## // Password verification

## while (password != 1234) {

## cout << "Invalid password. Try again: ";

## cin >> password;

## }

## displayMenu();

## do {

## cin >> option;

## switch (option) {

## case 1: {

## cout << "\*Transaction Statement\*\n";

## int accountNumber;

## cout << "Enter the account number: ";

## cin >> accountNumber;

## if (accountNumber == 123456789) {

## int bankOption;

## cout << "Please select the bank:\n";

## cout << "1. HBL\n2. Meezan\n3. MCB\n4. Allied\n";

## cin >> bankOption;

## cout << "How much money do you want to transfer?\n";

## displayWithdrawalOptions();

## cin >> withdrawalOption;

## switch (withdrawalOption) {

## case 1:

## amount = 500;

## break;

## case 2:

## amount = 1000;

## break;

## case 3:

## amount = 2000;

## break;

## case 4:

## amount = 5000;

## break;

## case 5:

## cout << "Enter the amount to transfer: ";

## cin >> amount;

## break;

## default:

## cout << "Invalid option!\n";

## continue;

## }

## if (amount <= totalAmount) {

## totalAmount -= amount;

## cout << "You transferred Rs. " << amount << ".\nRemaining balance: Rs. " << totalAmount << "\n";

## } else {

## cout << "Not enough balance.\n";

## }

## } else {

## cout << "Invalid account number.\n";

## }

## break;

## }

## case 2: {

## displayWithdrawalOptions();

## cin >> withdrawalOption;

## switch (withdrawalOption) {

## case 1:

## amount = 500;

## break;

## case 2:

## amount = 1000;

## break;

## case 3:

## amount = 2000;

## break;

## case 4:

## amount = 5000;

## break;

## case 5:

## cout << "Enter the amount to withdraw: ";

## cin >> amount;

## break;

## default:

## cout << "Invalid option!\n";

## continue;

## }

## if (amount <= totalAmount) {

## totalAmount -= amount;

## cout << "You withdrew Rs. " << amount << ".\nRemaining balance: Rs. " << totalAmount << "\n";

## } else {

## cout << "Not enough balance.\n";

## }

## break;

## }

## case 3:

## cout << "Your total balance is: Rs. " << totalAmount << "\n";

## break;

## case 4:

## cout << "Exiting. Thank you for using our service!\n";

## break;

## default:

## cout << "Invalid option. Please try again.\n";

## displayMenu();

## }

## } while (option != 4);

## return 0;

## }

## Output

## 