**-PROJECT REPORT**

**Name:**  Aman Kumar

Samir thakur

**Registration No.:** 23BCA20051

**Title:** ATM Machine

**Course Name:** Data Structures

**Semester:** 3

**-INDEX**

1. Problem Statement
2. Software Used
3. Functions / Modules
4. List of Errors Encountered while coding the project
5. Key or challenging logic in the project
6. Project Code
7. Elaborate Sample Input and Output Screenshots

**-PROBLEM STATEMENT**

To simulate the basic functionalities of an Automated Teller Machine (ATM) system. This application should allow a user to interact with the ATM through various options such as viewing account information, withdrawing cash, depositing cash, and checking the account balance.

Data structures used include an array, linked list, file handling, stacks and queues.

**-SOFTWARE REQUIREMENT**

CODE BLOCKS 13.12

or

DEV C++ 5.11

**-FUNCTIONS**

1**deposit()**

* Purpose: Allows the user to add money to their balance.
* Process: It prompts the user to enter an amount, adds that amount to balance amount, and displays the updated balance.

2. **withdraw()**

* Purpose: Allows the user to withdraw money from their balance.
* Process: It checks if the requested withdrawal amount is less than or equal to balance amount. If sufficient funds are available, it deducts the amount and displays the new balance; otherwise, it displays an error message.

3. **check balance()**

* Purpose: Displays the user’s current balance.
* Process: It simply outputs the value of balance amount to the user.

4. **atm machine transaction()**

* Purpose: Provides the main interface for users to select different ATM operations.
* Process: This function displays a menu of options (Deposit, Withdraw, Check Balance) and allows the user to select an action by entering a number. After each transaction, it asks the user if they would like to perform another transaction. If the user chooses to exit, the loop stops, ending the function.

5 **main()**

* Purpose: Entry point of the program.
* Process: It calls atm machine transaction() to start the ATM functionality and thanks the user when they exit the program.

**-LIST OF ERRORS**

**1. Input Validation Issues**

**2. Negative Deposit or Withdrawal**

**3. Incorrect Option Selection**

**4. Insufficient Funds Handling**

**5. Exit Confirmation Error**

**-KEY OR CHALLENGING LOGIC**

The major part of the challenge included making a doubly linked list that would accept string data and store it efficiently. Involving the usage of file handling was another challenge. To retrieve songs from a pre-made list and add it to the linked list. It also required us to be able to write the new input songs to the file. All changes in the program required to be reflected on the text file.

**-PROJECT CODE**

#include <iostream>

using namespace std;

float balance\_amount = 0.0;

void deposit() {

float deposit\_amount;

cout << "\nEnter the amount to deposit: ";

cin >> deposit\_amount;

balance\_amount += deposit\_amount;

cout << "Your current balance is Rs. " << balance\_amount << ". Thanks for depositing!" << endl;

}

void withdraw() {

float withdraw\_amount;

cout << "\nPlease enter amount to withdraw: ";

cin >> withdraw\_amount;

if (withdraw\_amount > balance\_amount) {

cout << "Insufficient fund! Please proceed to deposit money." << endl;

} else {

balance\_amount -= withdraw\_amount;

cout << "You have withdrawn Rs. " << withdraw\_amount << " and your balance is Rs. " << balance\_amount << endl;

}

}

void check\_balance() {

cout << "Your current bank balance is: Rs. " << balance\_amount << endl;

}

void atm\_machine\_transaction() {

int option, new\_transaction;

do {

cout << "Choices Available in the ATM Machine" << endl;

cout << "1. Deposit Money" << endl;

cout << "2. Withdraw Money" << endl;

cout << "3. Balance Amount" << endl;

cout << "\nYour option: ";

cin >> option;

switch(option) {

case 1: deposit(); break;

case 2: withdraw(); break;

case 3: check\_balance(); break;

default: cout << "Invalid option!" << endl; break;

}

cout << "Do you want a new transaction?\nPress 1 to 'proceed' and 2 to 'exit' from here\n";

cout << "Your option: ";

cin >> new\_transaction;

} while (new\_transaction == 1);

}

int main() {

atm\_machine\_transaction();

cout << "Thank you for the visit!" << endl;

return 0;

}**-SAMPLE INPUT AND OUTPUT**

