FENI UNIVERSITY
Center for Learning and Development

BACHELOR OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING

PROJECT REPORT

Course Code: CSE-218

Course Title: Data Structure Lab

PROJECT NAME HERE

Bank Management System

SUBMITTED BY		SUBMITTED TO
Name	Fahariya Yesmin	Abdur Rahman Riad Lab Technical Officer (LTO) Department of CSE Feni University
ID	222031014	
Batch	28th	
Department of CSE Feni University		

Date of Submission: 23 August 2023

1. INTRODUCTION

The Bank Management System is a software application that simulates the basic operations and functionalities of a bank. It involves managing customer accounts, performing transactions, and maintaining account records. This project report provides an overview of the Bank Management System's design, implementation, and the data structures used to achieve its functionality.

2. OBJECTIVE

The main objectives of the Bank Management System project are:

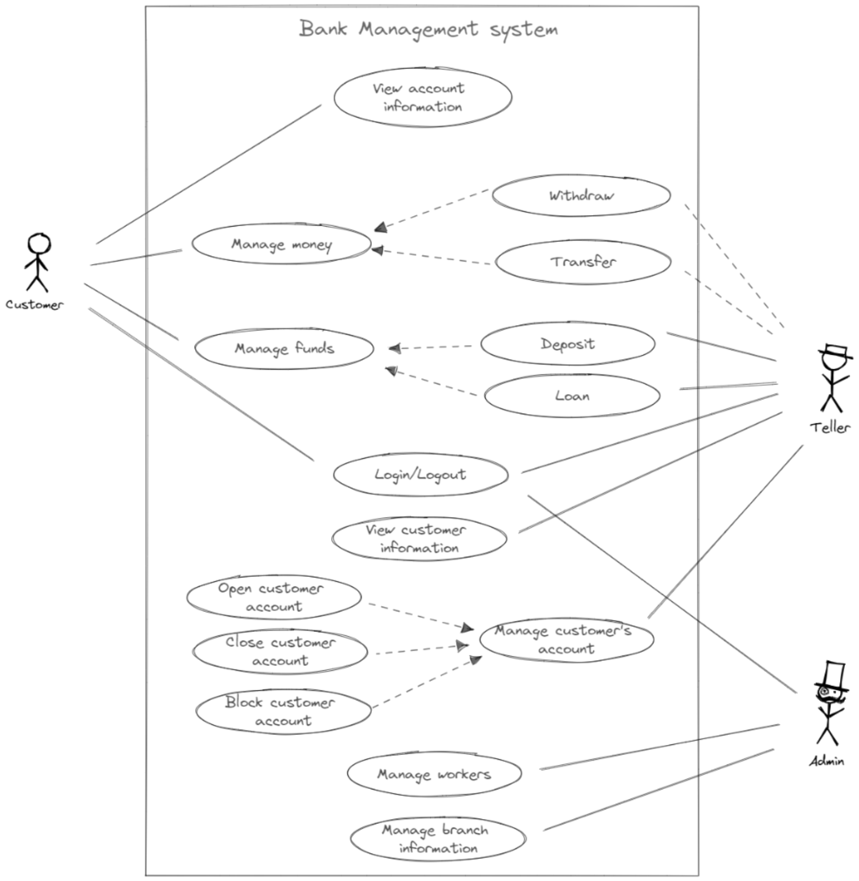
- Create and manage customer accounts.
- Allow deposit and withdrawal transactions.
- Calculate interest on savings accounts.
- Provide balance inquiry and transaction history.
- Demonstrate the use of various data structures for efficient operations.

3. OUTCOME

```
#include <stdio.h>

#include <stdlib.h>

#include <string.h>



// Define structures

struct Account {

    int accountNumber;

    char accountHolder[100];

    float balance;

};



// Function prototypes

void createAccount(struct Account accounts[], int *numAccounts);

void deposit(struct Account accounts[], int numAccounts);

void withdraw(struct Account accounts[], int numAccounts);

void displayAccounts(struct Account accounts[], int numAccounts);
```

```
int main() {
    struct Account accounts[100];
    int numAccounts = 0;
    int choice;

    while (1) {
        printf("Bank Management System\n");
        printf("1. Create Account\n");
        printf("2. Deposit\n");
        printf("3. Withdraw\n");
        printf("4. Display Accounts\n");
        printf("5. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                createAccount(accounts, &numAccounts);
                break;
            case 2:
                deposit(accounts, numAccounts);
                break;
            case 3:
                withdraw(accounts, numAccounts);
                break;
            case 4:
                displayAccounts(accounts, numAccounts);
                break;
            case 5:
                printf("Exiting...\n");
                exit(0);
            default:
                printf("Invalid choice. Please try again.\n");
        }
    }

    return 0;
}

void createAccount(struct Account accounts[], int *numAccounts) {
    // Implementation for creating an account
}

void deposit(struct Account accounts[], int numAccounts) {
    // Implementation for depositing money
}

void withdraw(struct Account accounts[], int numAccounts) {
    // Implementation for withdrawing money
}

void displayAccounts(struct Account accounts[], int numAccounts) {
    // Implementation for displaying account details
}
```

4. PROJECT DIAGRAM



5. IMPLEMENTATION PROCEDURE

Create Account: The bank employee can create a new customer account using the information given by the customer.

Deposit: This feature is used to add money to an existing customer account. The added money will be updated in the database.

Withdraw: This feature is used to withdraw money from an account.

Display Accounts: An option for checking account.

6. RESULT AND DISCUSSION

```
Bank Management System
1. Create Account
2. Deposit
3. Withdraw
4. Display Accounts
5. Exit
Enter your choice:
```

1. **Create Account:**

This option allows a bank employee or administrator to create a new account for a customer.
Information needed might include the customer's name, address, contact details, and initial deposit amount.
Each account would be assigned a unique account number and possibly a PIN for security.

2. **Deposit:**

Customers can deposit money into their accounts using this option.
The system should prompt the user to enter their account number and the amount they want to deposit.
The balance of the account should be updated accordingly.

3. **Withdraw:**

Customers can withdraw money from their accounts.
Similar to deposit, the system should ask for the account number and the withdrawal amount.
The system should ensure that the withdrawal amount does not exceed the account balance.

4. **Display Accounts:**

This option is for the bank staff to view a list of all accounts along with their details.
It could display information such as account number, account holder's name, balance, and possibly other details.

5. **Exit:**

This allows the user to exit the banking management system.

7. CONCLUSION

After finishing the Bank Management System project using the C programming language, it is clear that this system is a useful tool for managing the transactions and operations of a bank. Users can do many things with the system, like set up accounts, deposit and withdraw money, and check their account balances. This project shows how powerful and flexible the C language is, as well as how it can handle complicated tasks in a clear and efficient way. Overall, putting the Bank Management System into place using C was a success, and it is expected to be a useful tool for managing how a bank works.

Github Link: https://github.com/fahaariyayesmin/bank_management_system