Project Report: Banking System using C++ (OOP + DSA)

Project Title

Banking System - A Console-based Application in C++

Developed By

Abhishek Kumar

Galgotias University

 abhishek9504383290@gmail.com

1. Introduction

This project is a Banking System developed using C++, which simulates basic banking operations such as account creation, deposit, withdrawal, and account viewing. It follows Object-Oriented Programming (OOP) principles and utilizes DSA concepts such as the use of STL containers like vector to manage multiple accounts efficiently.

2. Objective

To develop a functional banking application that:

- Demonstrates core OOP principles like encapsulation and abstraction

- Implements a user-friendly command-line interface

- Uses DSA practices for internal data handling

- Simulates real-world banking operations

3. Tools & Technologies Used

C++ - Programming language

Project Report: Banking System using C++ (OOP + DSA)

G++ / GCC - Compiler

STL (vector) - For dynamic account storage

VS Code / Terminal - IDE/Editor and terminal to run

4. Key Features

- Create a new bank account

- Deposit money into the account

- Withdraw money from the account

- View details of a specific account

- Display all customer accounts

- Use of OOP classes and STL containers

5. System Design

Class: Account

Attributes: accountNumber, accountHolder, balance

Methods: deposit(), withdraw(), display()

Class: Bank

Attributes: vector<Account> accounts

Methods: createAccount(), viewAccount(), depositToAccount(), withdrawFromAccount(), listAllAccounts()

6. Future Enhancements

- Add file handling to store account data persistently

- Introduce authentication/login for added security

Project Report: Banking System using C++ (OOP + DSA)

- Implement a GUI version using frameworks like Qt or SFML

- Add transaction history tracking

7. Conclusion

This project showcases a practical implementation of object-oriented design in C++. It simulates real-world banking functionalities, emphasizes code reusability, and promotes modular and clean code structure - a solid example of applying OOP and DSA concepts together.

8. References

- C++ STL Documentation

- GeeksforGeeks - OOPs Concepts

- CPlusPlus.com - Input/Output & STL