BANK MANAGEMENT SYSTEM

GROUP CODE: C

GROUP MEMBERS:

Sufiyaan Usmani (21K-3195)

Qasim Hasan (21K-3210)

Ahsan Ashraf (21K-3186)

PROBLEM STATEMENT:

A bank management system is a program that monitors a person's bank account. The system enables the customer to create an account, deposit funds into it, and withdraw funds from it. It also provides the administrator with the ability to control the financial system.

PROJECT SCOPE:

This Bank Management System is not the exact replication of a real-world banking system. However, it provides several features which one can also find in a real banking system.

Boundaries of the System:

- This system does not include all the attributes and methods related to the bank's employees except an admin, which is given some controls to regulate the bank.
- It does not include any mortgage system.
- This banking system is not centralized, i.e., it's neither connected to any other banking software nor has any other connections with any other bank.
- It is wholly based on the console for the user interface and does not include GUI.

FEATURES:

We are going to set up a banking system. The project will be divided into three sections: administration portal, customer portal, applications, and some extra features such as:

- Customer: Transactions (withdraw, deposit, transfer), view transaction history, delete the account, edit account info.
- Admin: View all accounts, manage accounts, delete accounts, data backup
- Application: Password security, Currencies rates worldwide, create a new account with proper error checking, data in binary form, only program can access it.
- Bank Policy: Proper bank policies will be shown to the customers for which they have to agree before creating their account.
- Exceptional Handling: Efficient error checking and Exception handling methods.
- Email Notification: An email is sent when the customer makes a new account.
- Extra Features (OPTIONAL):
    1. Data Science (Python: pandas).
    2. Sending email: socket.h or poco library.
    3. User manual.
    4. Documentation and explanation of code.
    5. Using TextColour() and gotoxy() to make the interface of console look good.

Covering all Concepts of OOP:

Inheritance: Abstract and child classes

Example: There is an abstract class called user. Classes of admin and customer will be child classes of user class and inherit its properties and methods.

Polymorphism: Abstract class with method overloading.

Example: There will be a function called deleteAccount() for both admin and customer. The customer can only delete his account while the admin can delete accounts of both admin and customers. This will reflect polymorphism.

Abstraction: Hiding unnecessary information from both customers and admin.

Encapsulation: Grouping data and methods together and restricting direct access to data members.

Example:

LANGUAGE AND TOOLS:

C++ LANGUAGE

DEV C++

VISUAL CODE

GITHUB/GIT