Banking System Report

**Presented to :** ENG. Amr

**Done by :**

-Mostafa Khafaga

-Marwan Abdelaziz

-Belal Mohamed

**Introduction**

This report details a Java-based banking system implementation that provides essential banking operations through a command-line interface. The system allows users to perform common banking transactions such as creating accounts, depositing and withdrawing funds, transferring money between accounts, and handling stolen card scenarios.

The primary objectives of this project are:

-Implement a banking system with transaction handling.

-Demonstrate object-oriented programming principles.

-Provide a user-friendly interface for banking operations.

-Ensure data integrity in banking transactions.

**System Design**

The system follows an object-oriented architecture.

The system is built using the following key classes:

**Class Structure**

**Account (Abstract Class)**

-Base class for all account types

-Manages common account operations

-Defines abstract withdraw method

**SavingsAccount**

-Extends Account

-Implements minimum balance requirement

-Includes interest calculation functionality

**CurrentAccount**

-Extends Account

-Provides standard withdrawal functionality

**Bank**

-Manages account creation and operations

-Handles inter-account transfers

-Maintains transaction history

**BankingSystem**

-Provides user interface and input handling

-Coordinates between user input and bank operations

**AccountType**

-Defines characteristics of different bank account types

-Manages interest rates and minimum balance requirements

**UML Class Diagram**

A diagram of a computer program

Description automatically generated with medium confidence

**Implementation Details**

Object-Oriented Principles Implementation:

**Abstraction**

The Account class serves as an abstract base class

-Defines common account behavior while leaving specific withdrawal implementation to subclasses

-Encapsulates core banking operations

**Inheritance**

-SavingsAccount and CurrentAccount extend the Account class

-Each account type inherits basic account functionality

-Specialized behavior is implemented in respective subclasses

**Encapsulation**

-Account details (balance, holder name) are protected

-Access is controlled through public methods

-Data validation is performed within the classes

**Polymorphism**

-Account withdrawal behavior varies by account type

-Bank class handles different account types uniformly

-Runtime polymorphism in transfer operations

**User Guide**

**System Requirements**

-Java Development Kit (JDK) 8 or higher

-Command-line interface

**Compilation Instructions**

-Navigate to the project directory

-Compile all Java files:

javac twels/\*.java

**Running the Application**

-Execute the main class:

java twels.BankingSystem

**Using the Application**

**Account Creation**

1) Select option 1 from the main menu

2)Choose account type (Savings/Current)

3)Enter account holder name

4)Note the generated account number

**Fund Operations**

-Deposit: Option 3

-Withdraw: Option 4

-Transfer: Option 5

**Security Operations**

-Report stolen card: Option 6

-View transaction history: Option 7

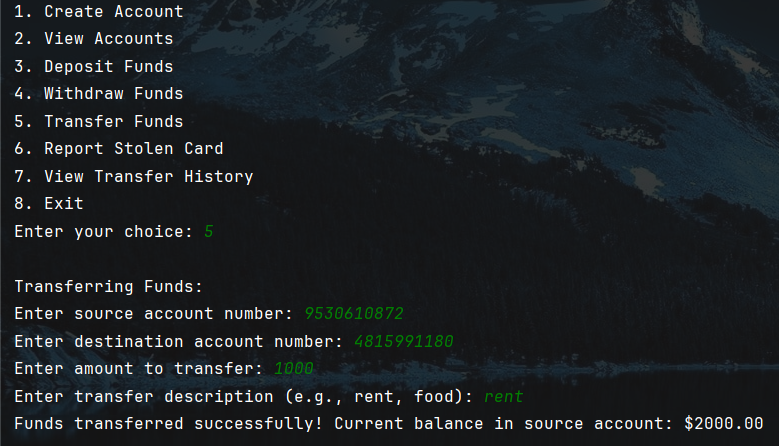
**Testing**

**Unit testing:**

**A screenshot of a computer

Description automatically generated**

**Integration testing:**

A screenshot of a computer

Description automatically generatedA screenshot of a computer screen

Description automatically generatedA screenshot of a computer

Description automatically generated

**A screen shot of a computer

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

**Trello link:**

This is the link of our board.

[**https://trello.com/invite/b/67645d243531dd9f217f203d/ATTI13a811b9c77e7c990bbc9892852789609F3C5330/12th-oop-project**](https://trello.com/invite/b/67645d243531dd9f217f203d/ATTI13a811b9c77e7c990bbc9892852789609F3C5330/12th-oop-project)