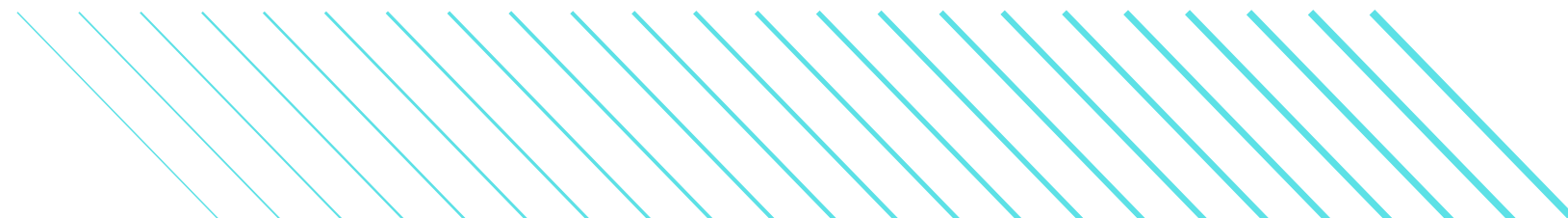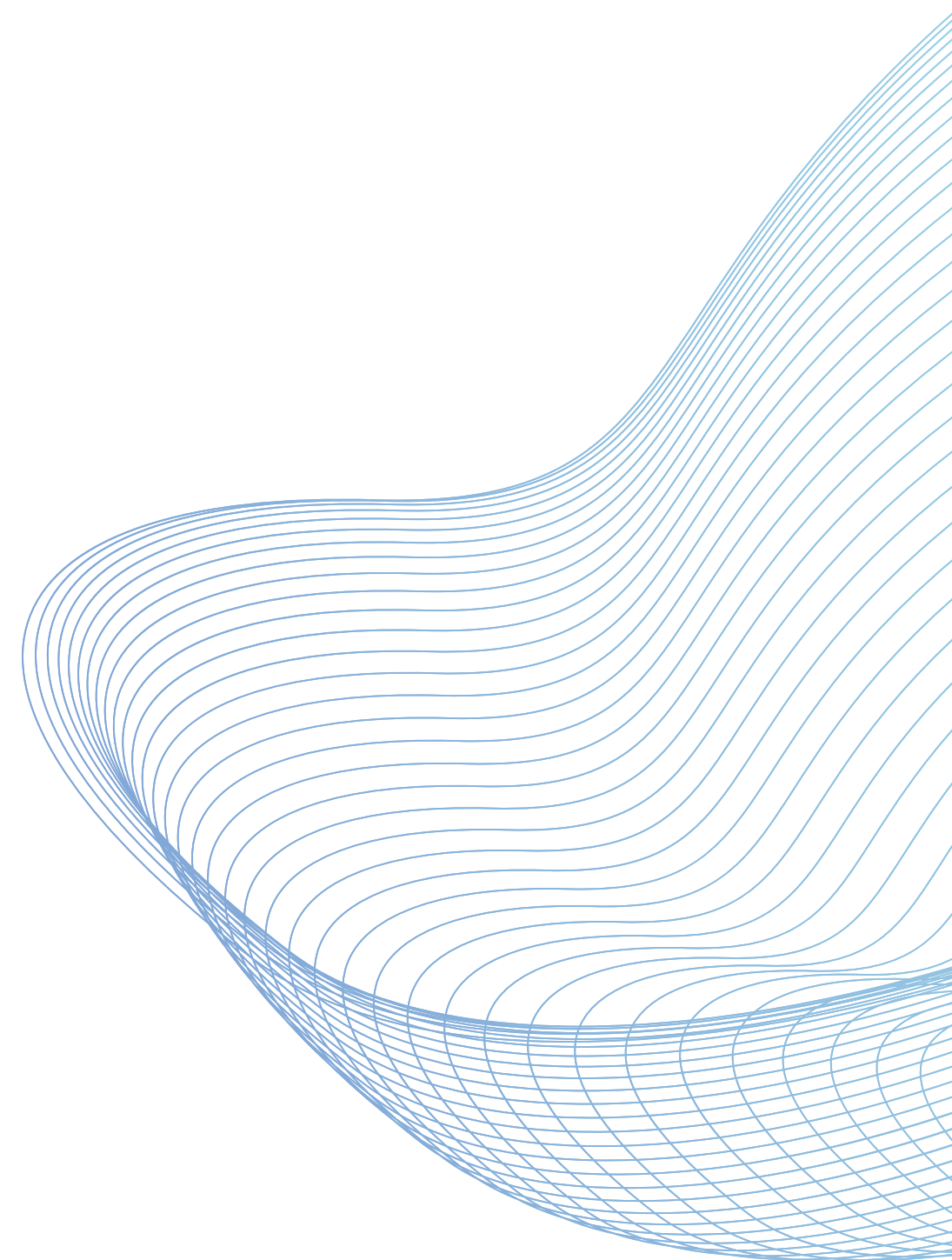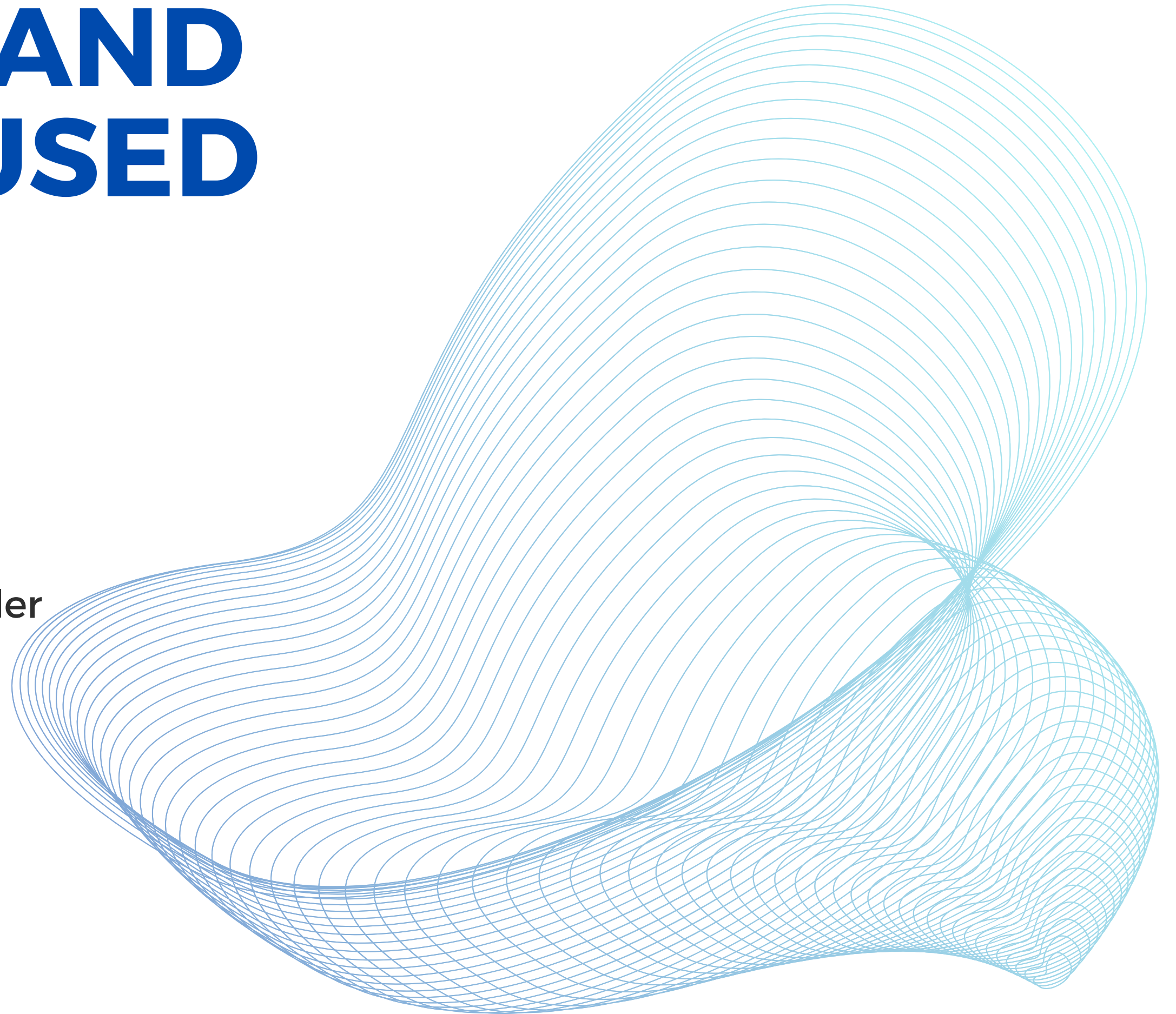# BANK
MANAGEMENT
PROJECT

Shubham Kaushik

AIM OF THE PROJECT

The aim of the Bank Management System is to develop a console-based application that provides users with basic banking functionalities such as logging in, signing up, checking account balance, depositing and withdrawing funds, taking loans, and transferring money between accounts. The project aims to ensure secure and efficient banking operations, user privacy, and convenience by delivering a user-friendly interface for managing bank accounts.

HARDWARE AND SOFTWARE USED

Hardware Component - Laptop

Software Component - Any C++ compiler

STRUCTURE OF CODE

User Class:

- This class represents a bank user and holds information such as name, password, account number, balance, and loan amount.
- Methods:
  - **GetName()**: Returns the name of the user.
  - **CheckPassword()**: Checks if the provided password matches the user's password.
  - **GetBalance()**: Retrieves the current account balance of the user.
  - **GetAccountNumber()**: Retrieves the account number associated with the user.
  - **Deposit()**: Adds the specified amount to the user's account balance.
  - **Withdraw()**: Subtracts the specified amount from the user's account balance, if sufficient funds are available.
  - **TakeLoan()**: Increases the user's account balance by the specified loan amount.
  - **Transfer()**: Transfers a specified amount from the user's account to another user's account.

STRUCTURE OF CODE

Bank Class:

- This class manages the collection of users and their associated operations.
- Methods:
    - **SignUp()**: Allows a new user to sign up by providing their name, password, and account number.
    - **LogIn()**: Authenticates a user by matching their provided name and password with existing user records.
    - **ShowBalance()**: Displays the account number and current balance of the logged-in user.
    - **Deposit()**: Allows the logged-in user to deposit funds into their account.
    - **Withdraw()**: Allows the logged-in user to withdraw funds from their account, if sufficient balance is available.
    - **TakeLoan()**: Allows the logged-in user to take a loan, increasing their account balance by the specified loan amount.
    - **Transfer()**: Allows the logged-in user to transfer funds to another user's account by specifying the recipient's account number.
    - **IsUserLoggedIn()**: Checks if a user is currently logged in.

STRUCTURE OF CODE

1. DisplayInitialMenu() Function:
   - Displays the initial menu options to the user, allowing them to choose between logging in, signing up, or exiting the program.
2. DisplayBankMenu() Function:
   - Displays the bank menu options to the logged-in user, providing various banking operations to choose from, including checking balance, depositing, withdrawing, taking a loan, transferring money, and logging out.
3. main() Function:
   - The entry point of the program.
   - Manages the flow of the program by presenting the initial menu, handling user choices, and calling corresponding class methods based on user actions.