

# **Bank Management System**

(A functional banking system demonstrating core C concepts)

**Department of Electronics and Computer Engineering  
IOE, Thapathali Campus**

## **TEAM:**

**THA081BEI009      Bidhan Shrestha**

**THA081BEI012      Kabindra Panta**

**THA081BEI022      Nirajan Chaudhary**

Date: 16<sup>th</sup> March, 2025

# OBJECTIVES

- Demonstrate use of C Programming concepts.
- Design & Build a functional banking system

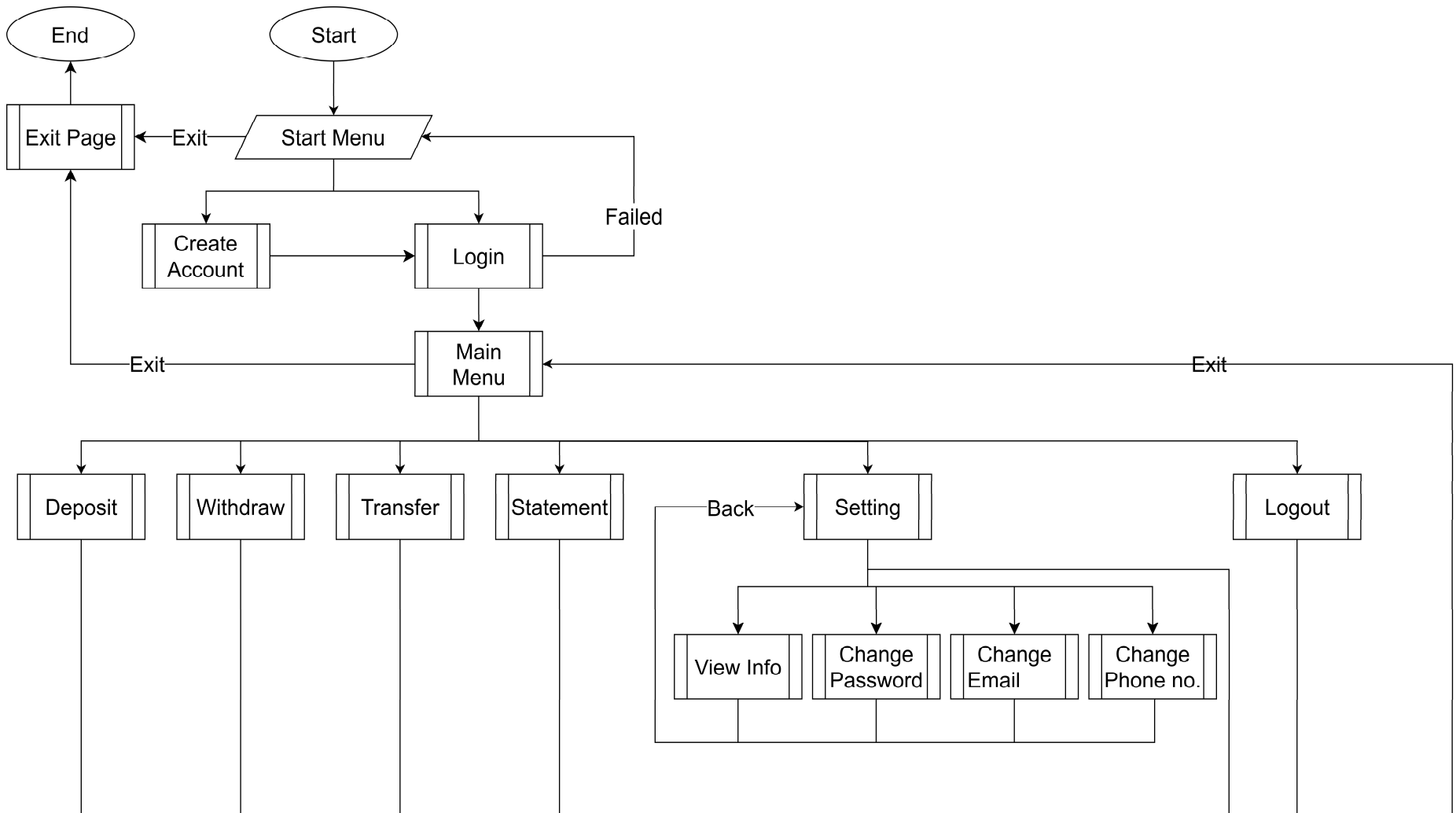
# Features

- Account creation and login
- Deposits, withdrawals transfers and transaction history
- User-info management (password, email, phone updates).
- Input validation and Masked Password Input
- Smooth and Intuitive UI

# System Architecture

- Structure for User data handling
  - ✓ struct User – 8 members
- File handling for data storage
  - ✓ userdetails.txt
  - ✓ transaction\_log.txt
- 20+ functions for modular design

# System Flowchart



# Security

- Accepts strong password
  - ✓ 8 characters minimum
  - ✓ At least 1 uppercase, 1 lowercase, 1 digit and 1 special character
- Masked Password Input
  - ✓ Shows '\*' for characters entered
- Very low data redundancy
- Unique phone number
- Buffer flushing to prevent buffer overflow

# Input Validation

- Email Validation
  - ✓ Checks for @ and . in email address
- Phone Number validation
  - ✓ Ensures 10 digits starting with 97/98
  - ✓ Ensures uniqueness.
- Date validation
  - ✓ Ensures YYYY-MM-DD format

# Banking Operations

- Deposit/ Withdraw
  - ✓ Takes and validates input amount
  - ✓ Updates balance
  - ✓ Logs transaction
- Transfer
  - ✓ Validates receiver account
  - ✓ Updates balance and Logs transaction
- Statement
  - ✓ Shows the record of transactions
  - ✓ Timestamp, transaction type, amount, A/C no., final balance



# UI/UX Design

- Intuitive control flow
  - ✓ Allows to go to previous menu
  - ✓ Doesn't end program on completing an operation
- Smooth UI
  - ✓ Subtle loading bar
  - ✓ Displays current operation

# Challenges

- Concurrent File Access
  - ✓ Risk of data corruption
- Date and Email Validation
  - ✓ Accepts any date in specified format
  - ✓ Doesn't ensure email's existence
- Data Integrity
  - ✓ Program Interruptions may affect integrity

# Future improvements

- Feature Upgrades
  - ✓ Admin Panel
  - ✓ Loan facilities
  - ✓ Develop GUI
- Technical Upgrades:
  - ✓ Use DBMS instead of files
  - ✓ Use hashing algorithms and encryption

# Thank You