銀行系統專案目標：完整詳述

本專案旨在開發一個基於 **C# Windows Forms** 的桌面銀行應用程式。此系統將模擬一個具備多層級使用者權限、基礎交易處理，以及嚴格內部控制的銀行運營環境。

### I. 核心使用者與權限管理

* **多角色使用者系統：** 系統將定義並支援四種明確的角色，每種角色擁有不同的功能權限：
  + **客戶 (Customer)：** 銀行服務的最終使用者，進行個人帳戶查詢與交易。
  + **職員 (Clerk)：** 負責客戶帳戶管理，是與客戶互動的主要角色。
  + **副行長 (VicePresident)：** 銀行管理層，監督營運，並具備職員所有權限。
  + **行長 (President)：** 銀行最高管理層，擁有最高控制權限，包括對銀行營運資金的調整。
* **使用者認證：**
  + **安全登入：** 所有使用者都必須透過帳號與密碼登入系統。系統將驗證憑證的有效性與使用者角色，並導向至對應的功能介面。
  + **密碼處理：** 密碼將採用基本的雜湊處理（而非明文）儲存，以提升安全性。
* **嚴格的使用者管理 (非客戶註冊)：**
  + **客戶無法自行註冊。** 所有新帳號的建立、刪除與權限設定，都將由具備足夠權限的銀行內部人員操作。
  + **職員權限：** 能夠新增、查詢和刪除**客戶帳號**，並設定其基本資訊（如客戶姓名、初始餘額等）。
  + **副行長權限：** 擁有職員的所有權限，並能夠新增、查詢和刪除**職員帳號**，同時可設定職員的權限。
  + **行長權限：** 擁有副行長的所有權限，並能夠新增、查詢和刪除**所有類型的帳號**（包括其他行長帳號），並可設定其角色權限。

### II. 資料模型與儲存 (簡化實現)

* **記憶體內資料儲存：** 為了簡化專案與開發，所有資料（使用者資訊、帳戶餘額、交易記錄等）將儲存在應用程式的記憶體中（使用 List<T> 或 Dictionary<K, V> 來模擬資料庫）。這表示當應用程式關閉時，資料將不會持久化。
* **核心資料模型：**
  + User **物件：** 儲存所有使用者（客戶、職員、副行長、行長）的通用資訊，如 UserID、Username、PasswordHash、Role、Name 等。
  + Account **物件：** 儲存客戶帳戶的詳細資訊，如 AccountNumber、OwnerUserID（關聯到客戶）、Balance、CreationDate 等。
  + Transaction **物件：** 記錄每次資金異動，如 TransactionID、AccountID、Type (存款/取款/轉帳/調整)、Amount、Timestamp、Description 等。
  + BankOperationalFunds**：** 一個獨立的數值，代表銀行自有資金，與客戶存款餘額總和分開管理。
  + AuditLog **物件：** 記錄重要的管理操作，如餘額修改、使用者管理、營運資金修改等，包含操作者、時間、內容等。

### III. 銀行業務功能

1. **客戶 (Customer) 功能：**
   * **個人帳戶查詢：** 登入後，客戶可清晰查看自己帳戶的當前餘額。
   * **存款操作：** 允許客戶向自己的帳戶存入資金，並自動更新帳戶餘額和銀行總資產。
   * **取款操作：** 允許客戶從自己的帳戶提取資金。系統會自動檢查餘額是否充足，若不足則拒絕交易並給予提示。
   * **交易歷史查詢：** 可查看自己帳戶的簡要歷史交易記錄（包括日期、交易類型、金額和餘額變化）。
2. **職員 (Clerk) 功能 (包含客戶所有功能)：**
   * **客戶帳戶列表與查詢：** 能夠查看所有客戶的帳戶列表及詳細資訊。
   * **新增/刪除客戶帳戶：** 能夠創建新的客戶帳號，並設定初始餘額；也能夠刪除現有客戶帳戶。
   * **有條件修改客戶餘額：** 職員可在預設的條件下（例如：單次修改金額的上下限）修改客戶的帳戶餘額，所有修改都將被記錄在 AuditLog 中。
3. **副行長 (VicePresident) 功能 (包含職員所有功能)：**
   * **銀行總資產查看：** 能夠查看當前銀行的**總資產**。
     + **總資產計算：** 總資產 = BankOperationalFunds + 所有客戶帳戶餘額的總和。
   * **新增/刪除職員帳號：** 能夠創建新的職員帳號，並設定其權限；也能夠刪除現有職員帳號。
4. **行長 (President) 功能 (包含副行長所有功能)：**
   * **有條件修改銀行營運資金：** 行長可對**銀行營運資金**進行調整。
     + **安全機制 (雙重驗證)：** 執行此高敏感操作時，系統將強制要求**同時輸入行長本人和一位副行長的密碼**進行雙重驗證，以確保操作的嚴謹性與安全性。
     + **操作說明強制要求：** 每次修改 BankOperationalFunds 時，系統將強制要求行長提供一個簡短的**操作說明**（例如：「銀行增資」、「內部資金調撥」），該說明將與修改記錄一併儲存。
     + **詳細記錄追溯：** 所有對 BankOperationalFunds 的修改都將被詳細記錄在 AuditLog 中，包括操作者（行長）、審批者（副行長）、操作時間、修改前數值、修改後數值，以及操作說明。

### IV. 使用者介面 (Windows Forms)

* **模組化設計：** 介面將根據使用者角色權限進行清晰劃分，確保不同角色只能看到並操作其權限範圍內的功能，減少混淆。
* **直觀友善：** 介面設計將力求簡潔、直觀，提供良好的使用者體驗。
* **即時回饋機制：** 所有操作（成功、失敗、錯誤）都將透過清晰的文字提示或訊息框，提供即時、明確的使用者回饋。