

壹、功能需求說明

本模組提供使用者於管理畫面中，查詢現有所有人員資料，並可新增一筆包含基本資料與聯絡資訊的完整人員資料。整體操作流程簡單直覺，支援查詢、顯示、輸入與儲存功能。

當使用者進入系統畫面時，系統將自動從資料庫中讀取所有人員清單，包含基本資料與聯絡方式（透過資料表關聯查詢）。資料讀取後將於畫面中呈現。

使用者可透過表單新增一筆人員資料，輸入內容包含姓名、身分證號、部門、Email、電話。當使用者填寫完畢並點擊「新增人員」按鈕後，系統會將此筆資料送至後端，同時寫入兩個資料表（人員主檔與聯絡資訊表），過程中使用 Transaction 確保資料一致性。

本模組支援反覆使用與資料同步，即使用者新增資料後，畫面將自動重新載入查詢結果，呈現最新內容。

貳、開發環境說明

- .NET Framework : 4.7.2
- 程式語言 : C# (搭配 ASP.NET Web Forms)
- 資料庫操作方式 : Dapper 套件操作 MySQL 資料庫

後端請使用指定 MySQL 連線字串：

```
Server=hopper.proxy.rlwy.net;Port=54714;Database=railway;Uid=root;Pwd=zN  
nvReIUJgZMqDKIS1GOMCoQEwhXpJ1c;
```

- 三層式架構說明：

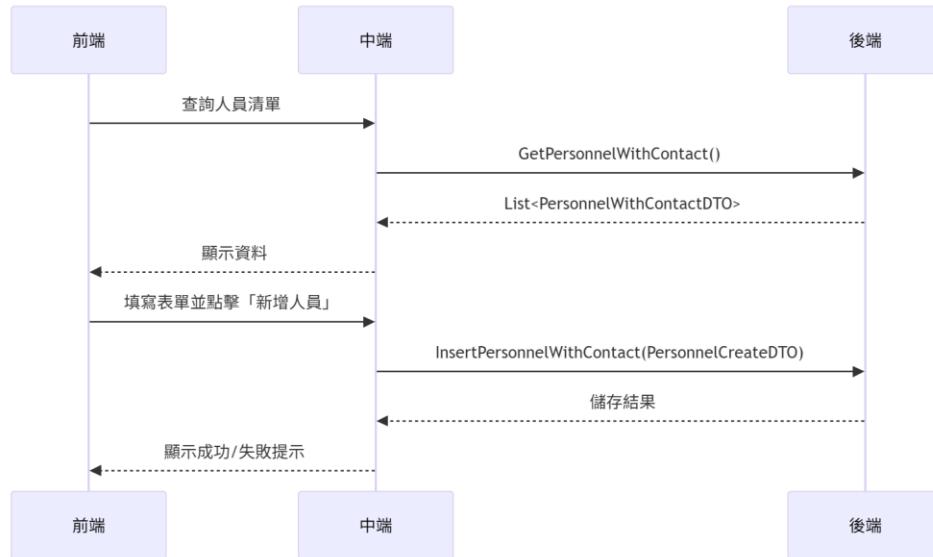
1. 前端 (Frontend)：呈現資料清單與表單輸入介面，觸發查詢與新增操作。
2. 中端 (Service Layer)：處理業務邏輯，封裝查詢與新增流程，呼叫後端函式並處理回傳資料。
3. 後端 (Repository Layer)：透過 Dapper 執行資料庫查詢與儲存，使用 Transaction 控制資料一致性。

- 開發補充說明：

- 請於 Visual Studio 開發環境中，使用 NuGet 管理工具安裝 Dapper 套件，可透過套件管理主控台執行：

```
Install-Package Dapper
```

參、 程式架構



肆、 中端功能模組說明

模組一：取得人員清單（中端）

- 模組用途：從資料庫中查詢現有所有人員資料，包含基本資料與聯絡資訊。
- 觸發條件：使用者進入管理畫面時系統自動呼叫。
- 資料處理流程：
 1. 中端初始化時，呼叫後端函式 `GetPersonnelWithContact()`。
 2. 後端將人員與聯絡資料 Join 查詢後回傳。
 3. 中端將結果暫存為 `List<PersonnelWithContactDTO>`，並轉呈畫面顯示。
- AI 串接說明：無
- 呼叫後端函式：`GetPersonnelWithContact()`
- 回傳格式與錯誤處理：`List<PersonnelWithContactDTO>`，若查無資料則回傳空清單。
- 是否為可重複使用模組：是

模組二：新增人員資料（中端）

- 模組用途：使用者填寫表單新增一筆人員資料（包含基本資訊與聯絡方式）。
- 觸發條件：使用者點擊「新增人員」按鈕
- 資料處理流程：
 1. 使用者於前端輸入 Name、IdentityNumber、Department、Email、Phone。
 2. 中端封裝為 `PersonnelCreateDTO`（含 `ContactInfoDTO`）。
 3. 呼叫後端函式 `InsertPersonnelWithContact()` 將資料寫入。

4. 後端回傳成功/失敗訊息。
 5. 若成功，中端更新畫面；若失敗，顯示錯誤。
- AI 串接說明：無
 - 呼叫後端函式：InsertPersonnelWithContact(PersonnelCreateDTO)
 - 回傳格式與錯誤處理：布林值與錯誤訊息（string）；失敗時前端顯示錯誤
 - 是否為可重複使用模組：否

伍、後端功能模組說明

函式一：GetPersonnelWithContact()

- 用途：從資料庫中查詢所有人員基本資料與聯絡資訊（Join 查詢）
- 使用資料表：
 - 人員主檔（Personnel）
 - 人員聯絡資訊（ContactInfo）
- 執行邏輯與處理流程：
 1. 使用 INNER JOIN 方式，將 ContactInfo.PersonnelId 對應至 Personnel.PersonnelId。
 2. 將符合條件之欄位組成 DTO 回傳。
- 是否使用 Transaction：否
- 回傳格式與錯誤處理：
 - 成功：List<PersonnelWithContactDTO>
 - 失敗：回傳空清單
- 對應之中端模組名稱：取得人員清單

函式二：InsertPersonnelWithContact()

- 用途：將一筆人員主檔資料與聯絡資訊寫入資料庫
- 使用資料表：
 - 人員主檔（Personnel）
 - 人員聯絡資訊（ContactInfo）
- 執行邏輯與處理流程：
 1. 開啟 Transaction。
 2. 插入 Personnel 表並取得自動產生的 PersonnelId。
 3. 以該 PersonnelId 插入對應的聯絡資料至 ContactInfo 表。
 4. Commit Transaction。
- 是否使用 Transaction：是
- 回傳格式與錯誤處理：
 - 成功：true

- 失敗：false + 錯誤訊息（例：Email 重複、欄位遺漏）
- 對應之中端模組名稱：新增人員資料