

元智大學資訊管理學系

第三十屆專業實習報告

公司代號：T2

實習單位：精誠集團-宜誠資訊

輔導老師：陸承志 教授

姓 名：陳品安

學 號：1112026

內容

壹、工作內容.....	2
● 一、工作環境介紹.....	2
● 二、工作詳述.....	2
● 三、實習期間完成進度.....	2
● 四、工作當中扮演的角色.....	3
貳、學習.....	4
● 一、撰寫前台網頁.....	4
● 二、預存程序—新增、刪除、查詢、修改(CRUD).....	6
● 三、系統規格書與測試報告.....	11
● 四、如何偵錯.....	12
● 五、自動化測試流程.....	12
參、自我評估與心得感想.....	14

壹、 工作內容

一、工作環境介紹

工作地點是**宜誠資訊股份有限公司**，地理位置位在台北市松山區復興北路 369 號 3 樓，距離捷運中山國中站僅五分鐘路程，鄰近公車站牌，在交通方面有多元化的選擇，除此之外，附近也有許多小餐廳，在飲食方面也非常方便。公司實施彈性上下班制度，上班時間為 8:30~9:30，可依自己的需求彈性調整上班時間。

二、工作詳述

我實習的部門為**系統整合處**，主要負責各類網站系統的開發與維護，從專案標案初期、系統開發、測試到後續維護皆有參與。部門承接的專案類型多元，包括網站建置、數位文件影像管理系統、數位典藏系統、應用系統及文物普查系統等。

而我參與**網頁前端界面的設計與開發**，實作不同功能的頁面，累積畫面切版、樣式設計與互動邏輯處理，培養依據需求調整與重構程式碼的能力；以及**後端負責撰寫預存程序**，實作新增、修改、刪除與查詢等功能，順利整合前後端並串接資料庫，確保整體系統功能順利運作。此外，也協助撰寫 SA 流程文件、操作手冊，逐步熟悉系統開發的整體流程與交付標準。

三、實習期間完成進度

- 1 月中~2 月中：練習

剛進來的時候其實對什麼程式都很不熟悉，會在 FreeCodeCamp、CodeWars 和 CodePen 等網站上進行練習，透過閱讀相關文件來增進理解程式及架構等基本知識，例如 Bootstrap 的官方文件，以及有關 Stored Procedure 的程式碼說明。除此之外，我也會練習前端畫面的設計，例如參考台灣國家公園的網站介面，來提升自己對於畫面開發與排版的熟悉度。

- 2 月中~3 月中：基礎系統開發

對程式有一點基礎的了解後，開始參與實際的系統開發，一開始都是負責一些比較簡單的畫面撰寫，接著開始嘗試學習如何和後端資料庫串接。主要協助開發「台中住宅補助」的兩個前端頁面，以及「建築影像管理系統」的一個畫面，同時也接觸到 oracle 的預存程序設計。正在學習要如何撰寫前端頁面並搭配預存程序，以塞入假資料的方式，模擬頁面完整運作。

- 3 月中~5 月底：「僑生管理系統」

到了三月中，我開始接觸一個新的專案「僑生管理系統」，負責協助翻新舊網站的整體畫面設計、修改一些原有的系統功能，以符合客戶的需求。這個專案中，在撰寫前端網頁畫面時，學到不少實用技巧，例如如果畫面都是同一個彈跳視窗的話可以共用，再根據需求按鈕去做調整就好，除了前端外，也開始實際撰寫 SQL Server 的預存程序，進行更貼近實務的資料操作，要根據資料庫中現有的資料，進行新增、刪除、查詢、修改的功能，確保頁面功能可以和資料庫進行串接和運作。除此之外，我也協助撰寫系統開發相關的 SA 流程文件，進一步瞭解系統的架構、功能設計以及流程規劃。因為這是第一個完整的跟著一個系統開始開發的經歷，這段經歷讓我對系統開發的整體流程有了更清楚也更實際的認識。

- 6 月~8 月：文件產出與整理、C#預存程序

這段期間，我主要參與了「台中育兒平台」的相關文件撰寫工作，包括測試報告與系統規劃書的撰寫，讓我更加熟悉專案文件在系統開發流程中的重要性。除此之外，我也協助「行政院主計總處（SBA）」的前期作業，運用 AI 技術將舊平台的畫面轉換成新平台的樣貌，體驗 AI 在系統重構與介面轉換上的應用。在這期間，我也嘗試以不同於以往的方式進行資料庫操作練習，例如在 C# 中直接撰寫新增、刪除、查詢、修改（CRUD）的預存程序，而非像過去一樣在 SQL 中撰寫，進一步加深了我對後端程式與資料庫整合的理解與實作經驗。

- 9 月~至今：「行政院主計總處」

九月後至今，我主要參與「行政院主計總處（SBA）」系統的測試與報告撰寫工作，負責執行系統功能測試、撰寫測試案例與測試報告，確保系統各項功能的正確性與穩定性。同時，我運用 AI 工具（Gemini）輔助開發，根據 HTML 結構自動生成對應的 C# 程式碼，並結合 Visual Studio 測試環境，透過 PowerShell 指令自動執行頁面新增、刪除、查詢、修改（CRUD）操作與截圖，實現測試過程自動化。在測試過程中，會根據自動化測試結果進行分析，若發現異常，會進一步檢查並修正相關的預存程序（Stored Procedure），或調整前端 JavaScript 傳入後端的參數設定，以確保資料能正確地與後端進行互動。

四、工作當中扮演的角色

- 前後端網頁開發者

在專案中，負責撰寫前端網頁，包括響應式網頁設計、互動功能等，也需要連接後端資料庫確保資料能正確地讀取與呈現，主要透過撰寫預存程序進行資料的新增、修

改、刪除與查詢，確保整體系統功能穩定運作。

- 持續學習者

除了從實做中累積經驗外，也必須要因應技術快速變化與 AI 等新興工具的應用，持續吸收新知，在工作中查閱相關文件、參考其他系統設計與實作方式，並嘗試使用新技術優化開發流程。透過這樣的學習模式，不僅提升了解決問題的能力，也更能適應不同專案需求與技術挑戰。

貳、學習

一、撰寫前台網頁

以 ASP.NET 架構做開發，使用 HTML、Bootstrap 製作響應式網頁，搭配 JavaScript、jQuery 和使用者做互動，由 C# 控制器做前後端資料串接。

我參考「台灣國家公園」的網站，練習撰寫前端畫面，與學校課堂上的開發方式不同，在實務中更強調程式碼的結構與維護性，像是 style 不直接寫在每個 HTML 檔案中，而是統一整理在 CSS 檔案中集中管理，這樣能提升效率，避免重複撰寫，後續也較容易維護與修改。也參與了這樣的開發經驗不僅讓我學會更有系統地管理前端程式碼，也讓我對整體網頁開發有了更深入的認識，我未來進一步開發 web 系統的基礎。

在「臺灣銀行教育訓練平台」的專案中，我負責撰寫前端登入與登出功能。由於此階段主要以功能模擬為主，因此在驗證部分採用最簡單的方式，直接於前端設定 value=""，無需連接資料庫。當使用者登入後，系統會跳轉至基本資料頁面，顯示相關帳號資訊，以驗證頁面是否正確運作。透過實作，我熟悉了使用者驗證的基本流程與頁面導向控制的概念，也理解了在系統開發初期，如何利用模擬方式快速驗證功能架構與互動邏輯，為後續串接真實資料與擴充功能奠定基礎。

每月主題
MONTHLY THEME



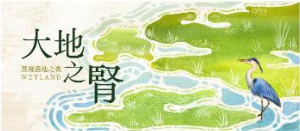
金包里大路（魚路古道）
「離山近，竹子埔南，金包里大路」向來是形勢險要山景的羣峰得勝，金包里大路（魚路古道）最初為根據地變遷金包里...

主題學習包
MTHEME LEARNING



海岸管理法
海岸是陸地與海洋的交界，而臺灣四面環海，擁有軟灘、泥沙、斷崖、珊瑚礁等豐富且多樣的海岸景觀，本主題學習包將...

主題特展
EXHIBITION







相關連結
RELATED LINK








THEME
主題特展





COLLECTION
藏品特展



年會
臺灣大特展教授方偉建致詞

RETROSPECT
歷史回顧



2022線上特展 - 50周年特展

▲「台灣國家公園」練習網站



臺灣銀行
教育訓練資源平台



帳號

密碼

登入



▲「臺灣銀行教育訓練平台」網站

二、預存程序—新增、刪除、查詢、修改(CRUD)

在接觸「僑生管理系統」時，學到比較常見的是**預存程序**（Stored Procedure）是一種將多筆 SQL 指令事先寫好並封裝起來的做法，可以大幅提升資料處理的效率與一致性。在實務應用上，通常會針對每個 CRUD 操作，撰寫一個對應的預存程序，前端只需透過 JavaScript 設計對應的 function 來呼叫後端的預存程序，使用者一旦觸發動作，就會自動執行對應的資料處理流程。SQL 的大致流程是，例如 `DECLARE @SDSTno NVARCHAR(8) = @inputXml.value(N'(/SDSTno)[1]', 'NVARCHAR(8)')`；從前端傳入的 XML 結構中解析出 SDSTno 的值，宣告變數 @SDSTno 儲存傳入參數，這樣就可以在接下來的 SQL 查詢或資料操作中使用這個值，系統會根據這些條件執行對應的 SELECT、INSERT、UPDATE 或 DELETE 動作，完成所需的資料處理。在進行編輯（Update）或刪除（Delete）操作時，系統會先透過 SELECT 指令查詢要修改或刪除的指定資料，並將內容帶入前端表單，讓使用者能夠檢視、確認或進行修改。使用者送出修改或確認刪除後，系統再依據該筆資料的主鍵條件執行 UPDATE 或 DELETE 動作為了系統的穩定與錯誤處理。預存程序也會使用 TRY CATCH 包住整個操作流程，確保在發生異常時能進行錯誤處理與紀錄，維持資料一致性與系統穩定性，讓**程式碼更簡潔、提升系統的安全性與維護效率**。此外，預存程序具備效能優化、權限控管、錯誤處理集中化等優點。在這個實作經驗中，讓我更深入理解前後端資料互動的流程，也強化了我對資料庫設計與系統整合的實務能力。

綜合資料維護作業(A300)

人員編號：	00107511		
中文姓名：		身分證號：	
☐ 精確比對 ☐ 模糊比對			
英文姓名：			
出生日期：	年/月/日	性別：	☐ 男 ☒ 女
僑居地洲別：		僑居地：	
i備卡			

[新增](#) [查詢](#) [清除畫面](#) 下方清單，若為 紅體字，表示為列管名單中的僑生。

顯示 10 項結果 搜尋：

修改	人員編號	中文姓名	英文姓名	出生日期	性別	身分證字號	學校名稱	僑居地
修改	人員編號	中文姓名	英文姓名	出生日期	性別	身分證字號	學校名稱	僑居地
編輯	00107511	葉蕭依	YAP HUE YEE	1967-11-29	F	RgAxADIAMw	國立臺灣大學	01

顯示第 1 至 1 項結果，共 1 項 [首頁](#) [上一頁](#) [1](#) [下一頁](#) [尾頁](#)

```
ALTER PROCEDURE [dbo].[A30001]
    @inputXml XML,
    @code INT OUTPUT,
    @message NVARCHAR(200) OUTPUT
AS
BEGIN
    BEGIN TRY
        DECLARE @SDSTno NVARCHAR(8) = @inputXml.value('(/Q_S DSTno)[1]', 'NVARCHAR(8)');
        DECLARE @Name NVARCHAR(120) = @inputXml.value('(/Q_Name)[1]', 'NVARCHAR(120)');
        DECLARE @Ename NVARCHAR(400) = @inputXml.value('(/Q_Ename)[1]', 'NVARCHAR(400)');
        DECLARE @Birthday DATETIME = @inputXml.value('(/Q_Birthday)[1]', 'DATETIME');
        DECLARE @Sex NVARCHAR(1) = @inputXml.value('(/Q_Sex)[1]', 'NVARCHAR(1)');
        DECLARE @IdentityId NVARCHAR(40) = @inputXml.value('(/Q_IdentityId)[1]', 'NVARCHAR(40)');
        DECLARE @DocUnit NVARCHAR(200) = @inputXml.value('(/Q_DocUnit)[1]', 'NVARCHAR(200)');
        DECLARE @colonyArea NVARCHAR(5) = @inputXml.value('(/Q_colonyArea)[1]', 'NVARCHAR(5)');

        SELECT SDSTno SDSTno_01,
            *,
            '國立臺灣大學' DocUnit, --學校名稱
            CONVERT(VARCHAR(10), Birthday, 120) Birthday_01
        from Personal
        WHERE(@SDSTno = '' OR SDSTno LIKE @SDSTno + '%') AND
            (@Name = '' OR Name LIKE @Name + '%') AND
            (@Ename = '' OR Ename LIKE @Ename + '%') AND
            (@IdentityId = '' OR IdentityId LIKE @IdentityId + '%') AND
            (@Birthday = '' OR CONVERT(VARCHAR(10), Birthday, 120) = CONVERT(VARCHAR(10), @Birthday, 120)) AND
            (@Sex = '' OR Sex LIKE @Sex + '%') AND
            (@colonyArea = '' OR colonyArea LIKE @colonyArea + '%')

        SET @code = 1;
        SET @message = 'Success';
    END TRY
    BEGIN CATCH
        SET @code = 99;
        SET @message = ERROR_MESSAGE();
    END CATCH
END;
```

▲查詢功能

學籍資料編號:	141928		
人員編號:	00107511	申請編號:	00150401
申請日期:	2025/04/29	來文單位:	國立高雄科技大學
來文字號:	12345	備輔文號:	1
獎(補)助日期(起):	2025/05/06	獎(補)助日期(迄):	2025/06/06
獎(補)助類別:	僑務委員會補助僑生	獎(補)助項目:	趙資香先生獎學金
金額:	1111	次數:	3
說明:	test		
備註:	test		
特殊身分:	學生		

儲存 刪除 關閉

```

ALTER PROCEDURE [dbo].[A30402]
    @inputXml xml,
    @code int OUTPUT,
    @message nvarchar(100) = '' OUTPUT
AS
BEGIN
    SET NOCOUNT ON;
    BEGIN
        DECLARE @KeyId int = @inputXml.value('(/KeyId)[1]', 'int');
        DECLARE @ApplyDate DATETIME = @inputXml.value('(/ApplyDate)[1]', 'DATETIME');
        DECLARE @SupportSDate DATETIME = @inputXml.value('(/SupportSDate)[1]', 'DATETIME');
        DECLARE @SupportEDate DATETIME = @inputXml.value('(/SupportEDate)[1]', 'DATETIME');

        SELECT *,
            CONVERT(char(10), [ApplyDate], 120) [ApplyDate_01],
            CONVERT(char(10), [SupportSDate], 120) [SupportSDate_01],
            CONVERT(char(10), [SupportEDate], 120) [SupportEDate_01]
        FROM Support
        WHERE KeyId = @KeyId;
        SET @code = 1;
    END
END

```

```

ALTER PROCEDURE [dbo].[A30403]
    @inputXml XML,
    @code INT OUTPUT,
    @message NVARCHAR(200) OUTPUT
AS
BEGIN
    BEGIN TRY
        DECLARE @KeyId          int = @inputXml.value('(/*KeyId)[1]', 'int');
        DECLARE @SDSTno         NVARCHAR(256) = @inputXml.value('(/*SDSTno)[1]', 'NVARCHAR(256)');
        DECLARE @SDSTApplyNo    NVARCHAR(256) = @inputXml.value('(/*SDSTApplyNo)[1]', 'NVARCHAR(256)');
        DECLARE @ApplyDate_01   DATETIME     = @inputXml.value('(/*ApplyDate_01)[1]', 'DATETIME');
        DECLARE @DocUnit        NVARCHAR(256) = @inputXml.value('(/*DocUnit)[1]', 'NVARCHAR(256)');
        DECLARE @DocNo          NVARCHAR(256) = @inputXml.value('(/*DocNo)[1]', 'NVARCHAR(256)');
        DECLARE @SDDocNo        NVARCHAR(256) = @inputXml.value('(/*SDDocNo)[1]', 'NVARCHAR(256)');
        DECLARE @SupportSDate   DATETIME     = @inputXml.value('(/*SupportSDate_01)[1]', 'DATETIME');
        DECLARE @SupportEDate   DATETIME     = @inputXml.value('(/*SupportEDate_01)[1]', 'DATETIME');
        DECLARE @SupportKind     NVARCHAR(256) = @inputXml.value('(/*SupportKind)[1]', 'NVARCHAR(256)');
        DECLARE @SupportItem     NVARCHAR(256) = @inputXml.value('(/*SupportItem)[1]', 'NVARCHAR(256)');
        DECLARE @SupportAmount   NUMERIC(18,2) = TRY_CAST(@inputXml.value('(/*SupportAmount)[1]', 'NVARCHAR(256)') AS NUMERIC(18,2));
        DECLARE @SupportCount    NVARCHAR(256) = @inputXml.value('(/*SupportCount)[1]', 'NVARCHAR(256)');
        DECLARE @Decription     NVARCHAR(256) = @inputXml.value('(/*Decription)[1]', 'NVARCHAR(256)');
        DECLARE @Note            NVARCHAR(256) = @inputXml.value('(/*Note)[1]', 'NVARCHAR(256)');
        DECLARE @Specail        NVARCHAR(256) = @inputXml.value('(/*Specail)[1]', 'NVARCHAR(256)');

        UPDATE Support
        SET
            SDSTno         = @SDSTno,          SDSTApplyNo    = @SDSTApplyNo,
            ApplyDate      = @ApplyDate_01,    DocUnit        = @DocUnit,
            DocNo          = @DocNo,           SDDocNo        = @SDDocNo,
            SupportSDate   = @SupportSDate,    SupportEDate   = @SupportEDate,
            SupportKind     = @SupportKind,     SupportItem     = @SupportItem,
            SupportAmount   = @SupportAmount,   SupportCount    = @SupportCount,
            Decription     = @Decription,      Note           = @Note,          Specail = @Specail
        WHERE KeyId = @KeyId and SDSTno=@SDSTno and SDSTApplyNo=@SDSTApplyNo;
        SET @code = 1;
        SET @message = 'Success';
    END TRY
    BEGIN CATCH
        SET @code = 99;
        SET @message = ERROR_MESSAGE();
    END CATCH
END;

```

▲編輯功能，利用查詢將指定資料帶進表單，修改後再對資料進行更新

```

ALTER PROCEDURE [dbo].[A30405]
    @inputXml xml,
    @code int OUTPUT,
    @message nvarchar(100) = '' OUTPUT
AS
BEGIN
    SET NOCOUNT ON;
    BEGIN
        DECLARE @KeyId nvarchar(50) = @inputXml.value('(/*KeyId)[1]', 'nvarchar(50)');

        DELETE FROM Support
        WHERE KeyId = @KeyId

        SET @code = 1;
    END
END

```

▲刪除功能

學籍資料編號:			
人員編號:	00107511	申請編號:	00150401
申請日期:	年/月/日	來文單位:	
來文字號:		僑輔文號:	
獎(補)助日期(起):	年/月/日	獎(補)助日期(迄):	年/月/日
獎(補)助類別:		獎(補)助項目:	
金額:		次數:	
說明:			
備註:			
特殊身分:			

新增

關閉

```

ALTER PROCEDURE [dbo].[A30404]
    @inputXml XML,
    @code INT OUTPUT,
    @message NVARCHAR(200) OUTPUT
AS
BEGIN
    BEGIN TRY
        DECLARE @SDSTno NVARCHAR(256) = @inputXml.value('(/SDSTno)[1]', 'NVARCHAR(256)');
        DECLARE @SDSTApplyNo NVARCHAR(256) = @inputXml.value('(/SDSTApplyNo)[1]', 'NVARCHAR(256)');
        DECLARE @ApplyDate_01 DATETIME = @inputXml.value('(/ApplyDate_01)[1]', 'DATETIME');
        DECLARE @DocUnit NVARCHAR(256) = @inputXml.value('(/DocUnit)[1]', 'NVARCHAR(256)');
        DECLARE @DocNo NVARCHAR(256) = @inputXml.value('(/DocNo)[1]', 'NVARCHAR(256)');
        DECLARE @SDDocNo NVARCHAR(256) = @inputXml.value('(/SDDocNo)[1]', 'NVARCHAR(256)');
        DECLARE @SupportSDate_01 DATETIME = @inputXml.value('(/SupportSDate_01)[1]', 'DATETIME');
        DECLARE @SupportEDate_01 DATETIME = @inputXml.value('(/SupportEDate_01)[1]', 'DATETIME');
        DECLARE @SupportKind NVARCHAR(256) = @inputXml.value('(/SupportKind)[1]', 'NVARCHAR(256)');
        DECLARE @SupportItem NVARCHAR(256) = @inputXml.value('(/SupportItem)[1]', 'NVARCHAR(256)');
        DECLARE @SupportAmount NVARCHAR(256) = @inputXml.value('(/SupportAmount)[1]', 'NVARCHAR(256)');
        DECLARE @SupportCount NVARCHAR(256) = @inputXml.value('(/SupportCount)[1]', 'NVARCHAR(256)');
        DECLARE @Decription NVARCHAR(256) = @inputXml.value('(/Decription)[1]', 'NVARCHAR(256)');
        DECLARE @Note NVARCHAR(256) = @inputXml.value('(/Note)[1]', 'NVARCHAR(256)');
        DECLARE @Specail NVARCHAR(256) = @inputXml.value('(/Specail)[1]', 'NVARCHAR(256)');

        INSERT INTO
        Support (SDSTno,SDSTApplyNo,ApplyDate,DocUnit,DocNo,SDDocNo,SupportSDate,SupportEDate,
        SupportKind, SupportItem,SupportAmount,SupportCount,Decription,Note,Specail)
        VALUES (@SDSTno,@SDSTApplyNo,@ApplyDate_01,@DocUnit,@DocNo,@SDDocNo,
        @SupportSDate_01,@SupportEDate_01,@SupportKind,@SupportItem,
        TRY_CONVERT(NUMERIC(18, 2), @SupportAmount), -- 將字串轉換為數字
        TRY_CONVERT(NUMERIC(18, 0), @SupportCount), -- 將字串轉換為整數
        @Decription,@Note,@Specail);

        SET @code = 1;
        SET @message = 'Success';
    END TRY
    BEGIN CATCH
        SET @code = 99;
        SET @message = ERROR_MESSAGE();
    END CATCH
END;

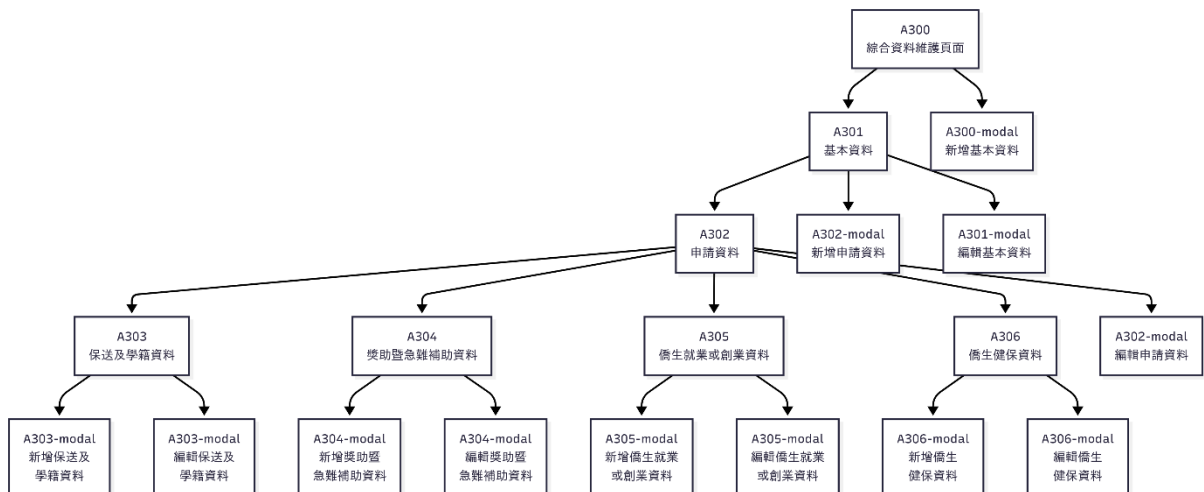
```

▲新增功能

三、系統規格書與測試報告

在這次實習中，我首次接觸到**系統規格書**，學習從中**解讀功能需求與系統邏輯**，這對我建立開發前的系統分析與規劃能力有很大幫助。規格書中會詳細列出系統功能、作業流程，以及各頁面的輸入項目與作業，讓我能更清楚掌握專案的整體架構與開發方向，避免在實作時產生誤解或偏離需求的情況。除此之外，我也運用了 mermaid 工具繪製流程圖，這對於視覺化系統邏輯與溝通需求流程非常有幫助，先利用 AI 工具，根據系統畫面與功能的操作流程撰寫描述，再由 mermaid 自動產生對應的流程圖，這可以大幅提升繪製的效率，也讓流程邏輯更明瞭，方便與團隊成員討論與確認。同時，我也學習撰寫**測試報告**，透過實際執行測試案例檢驗系統運作是否順暢，並比對預期結果與實際執行結果是否一致，以**確認系統功能的正確性與穩定性**，這不僅有助於發現潛在問題，也能提升整體系統品質。這樣的實作經驗讓我體會到系統規劃書在開發流程中的價值，讓我在日後面對複雜系統時，能更有條理地進行需求拆解與流程設計。

```
1 ---
2 config:
3   theme: redux
4   layout: dagre
5 ---
6 graph TD
7   A300[A300<br>綜合資料維護頁面]
8   A301[A301<br>基本資料]
9   A302[A302<br>申請資料]
10  A303[A303<br>保送及學籍資料]
11  A304[A304<br>獎助暨急難補助資料]
12  A305[A305<br>僑生就業或創業資料]
13  A306[A306<br>僑生健保資料]
14
15  A300_add[A301-modal<br>新增基本資料]
16
17  A301_add[A302-modal<br>新增申請資料]
18  A301_edit[A301-modal<br>編輯基本資料]
19
20  A302_edit[A302-modal<br>編輯申請資料]
21
22  A303_add[A303-modal<br>新增保送及<br>學籍資料]
23  A303_edit[A303-modal<br>編輯保送及<br>學籍資料]
24
25  A304_add[A304-modal<br>新增獎助暨<br>急難補助資料]
26  A304_edit[A304-modal<br>編輯獎助暨<br>急難補助資料]
27
28  A305_add[A305-modal<br>新增僑生就業<br>或創業資料]
29  A305_edit[A305-modal<br>編輯僑生就業<br>或創業資料]
30
31  A306_add[A306-modal<br>新增僑生<br>健保資料]
32  A306_edit[A306-modal<br>編輯僑生<br>健保資料]
33
34  A300 --> A301
35  A301 --> A302
36  A302 --> A303
37  A302 --> A304
38  A302 --> A305
39  A302 --> A306
40  A300 --> A300_add
41  A301 --> A301_add
42  A301 --> A301_edit
43  A302 --> A302_edit
44
45  A303 --> A303_add
46  A303 --> A303_edit
47
48  A304 --> A304_add
49  A304 --> A304_edit
50
51  A305 --> A305_add
52  A305 --> A305_edit
53
54  A306 --> A306_add
55  A306 --> A306_edit
```



▲使用 mermaid 繪製的流程圖

四、如何偵錯

相較於學校作業，公司專案的規模通常更為龐大，無法像在學校時一樣，透過逐行檢查找出錯誤，逐一檢查不僅耗費大量時間，也會嚴重影響工作效率。因此，培養有效的偵錯能力，是在職場上不可或缺的重要技能。其中，學會閱讀與理解錯誤訊息也很重要，當程式發生錯誤時，編譯器會提供詳細的錯誤訊息與堆疊追蹤資訊，能善加分析這些訊息，就能快速找到錯誤來源。在職場實務中，具備從錯誤訊息中提取關鍵資訊的能力，將有助於大幅縮短除錯時間，熟練運用各種偵錯方法與技巧，不僅能有效提升開發效率，更能確保專案如期順利完成。

五、自動化測試流程

在實習過程中，我也嘗試結合 AI 技術與自動化測試流程，運用 Google Gemini 來輔助程式開發與測試。只要在 Gemini 中輸入指令，並提供參考的 HTML 結構，它就能自動產生對應的 C# 程式碼，實現網頁中新增、刪除、查詢、修改（CRUD）功能的操作模擬。接著，將這些程式整合至 Visual Studio 的測試環境中，只需在 PowerShell 輸入指令，例如：`dotnet test sba-sfund-tests --filter "name~CM_B13" --logger "console;verbosity=detailed"`，系統就會自動根據該頁面的功能執行測試，並在過程中自動進行詳細的截圖與記錄，這樣的流程讓整個測試過程更有效率，不僅能節省人工操作的時間，也能減少人為疏漏。透過這個過程，我深入了解了 AI 在自動化測試中的應用方式，並體驗到如何結合程式開發、AI 模型與測試工具，達到高效率的系統驗證。

```
public class TESTB13_CM_B13 : ReportingTest
{
    private const string testFunction = "CM-B13-機構代碼";

    private const string TargetGov = "0"; // 0. 中央
    private const string NewTargetCode = "A1T0001";
    private const string NewTargetName = "自動化測試機構";
    private const string UpdatedTargetName = "自動化測試機構-已修改";
    private const string TargetFlag = "1"; // 1. 目錄對象

    3 個參考
    private async Task NavigateAndFilter()
    {
        var uri = new Uri(Page.Url);
        var baseUrl = uri.GetLeftPart(UriPartial.Authority);
        await Page.GotoAsync(baseUrl + "/CM/B13");
        await Expect(Page.Locator("#func-pane")).ToContainTextAsync("機構代碼");
        await Page.Locator("#table-filter").SelectOptionAsync(new SelectOptionValue { Value = TargetGov });
        await Page.WaitForLoadStateAsync(LoadState.NetworkIdle);
        await Page.WaitForTimeoutAsync(500);
    }
}
```

```

public async Task CM_B13_Create()
{
    _stepCounter = 1;
    await EnsureLoggedInAsAdmin();
    await NavigateAndFilter();
    await TakeStepScreenshot($"{testFunction}_導航至頁面");

    await Page.Locator("#btn-new").ClickAsync();
    await TakeStepScreenshot($"{testFunction}_點擊新增按鈕");

    await Page.Locator("[name=target_code]").FillAsync(NewTargetCode);
    await Page.Locator("[name=target_name]").FillAsync(NewTargetName);
    await Page.Locator($"[name=target_flag][value='{TargetFlag}']").CheckAsync();
    await TakeStepScreenshot($"{testFunction}_填寫必填欄位");

    await Page.Locator("#btn-save").ClickAsync();
    await Page.WaitForTimeoutAsync(1000);
    await Expect(Page.Locator("body")).ToContainTextAsync("新增成功");
    await TakeStepScreenshot($"{testFunction}_新增成功訊息");

    await Page.GetByRole(AriaRole.Button, new() { Name = "確認" }).ClickAsync();
    await Page.WaitForTimeoutAsync(300);

    await Page.Locator("#table-search").FillAsync(NewTargetCode);
    await Page.WaitForTimeoutAsync(500);
    await Expect(Page.Locator("#main-table")).ToContainTextAsync(NewTargetCode);
    await TakeStepScreenshot($"{testFunction}_驗證新增結果");
}

```

```

public async Task CM_B13_Update()
{
    _stepCounter = 1;
    await EnsureLoggedInAsAdmin();
    await NavigateAndFilter();

    await Page.Locator("#table-search").FillAsync(NewTargetCode);
    await Page.WaitForTimeoutAsync(500);
    await TakeStepScreenshot($"{testFunction}_搜尋更新目標");

    var targetRow = Page.Locator("#main-table tr", new() { HasText = NewTargetCode });
    await targetRow.ClickAsync();
    await TakeStepScreenshot($"{testFunction}_選擇資料列進行更新");

    await Page.Locator("[name=target_name]").FillAsync(UpdatedTargetName);
    await TakeStepScreenshot($"{testFunction}_填寫更新後名稱");

    await Page.Locator("#btn-save").ClickAsync();
    await Page.WaitForLoadStateAsync(LoadState.NetworkIdle);

    await Expect(Page.Locator("body")).ToContainTextAsync("紀錄已成功更新");
    await Page.WaitForTimeoutAsync(300);
    await TakeStepScreenshot($"{testFunction}_更新成功訊息");

    await Page.GetByRole(AriaRole.Button, new() { Name = "確認" }).ClickAsync();
    await Page.WaitForTimeoutAsync(300);

    await Page.Locator("#table-search").FillAsync(UpdatedTargetName);
    await Page.WaitForTimeoutAsync(500);

    var updatedRow = Page.Locator("#main-table tr", new() { HasText = UpdatedTargetName });
    await Expect(updatedRow).ToContainTextAsync(UpdatedTargetName);
    await TakeStepScreenshot($"{testFunction}_驗證更新結果");
}

```

```

public async Task CM_B13_Delete()
{
    _stepCounter = 1;
    await EnsureLoggedInAsAdmin();
    await NavigateAndFilter();

    await Page.Locator("#table-search").FillAsync(UpdatedTargetName);
    await Page.WaitForTimeoutAsync(500);
    await TakeStepScreenshot($"{testFunction}_搜尋刪除目標");

    var targetRow = Page.Locator("#main-table tr", new() { HasText = UpdatedTargetName });
    await Expect(targetRow).ToBeVisibleAsync();
    await targetRow.ClickAsync();
    await TakeStepScreenshot($"{testFunction}_選擇資料列進行刪除");

    Page.Dialog += async (_, dialog) =>
    {
        await dialog.AcceptAsync();
    };

    await Page.Locator("#btn-delete").ClickAsync();
    await TakeStepScreenshot($"{testFunction}_點擊刪除按鈕");

    await Expect(Page.Locator("body")).ToContainTextAsync("資料已成功刪除");
    await Page.WaitForTimeoutAsync(300);
    await TakeStepScreenshot($"{testFunction}_刪除成功訊息");

    var confirmButton = Page.GetByRole(AriaRole.Button, new() { Name = "確認" });
    if (await confirmButton.CountAsync() > 0)
    {
        await confirmButton.ClickAsync();
        await Page.WaitForTimeoutAsync(300);
    }

    await Page.Locator("#table-search").FillAsync(UpdatedTargetName);
    await Page.WaitForTimeoutAsync(500);
    await Expect(Page.Locator("#main-table")).Not.ToContainTextAsync(UpdatedTargetName);
    await TakeStepScreenshot($"{testFunction}_驗證刪除結果");
}

```

▲C#程式碼

```

環境完成 (2.4 秒)
sba-sfund-tests 成功 (0.5 秒) → sba-sfund-tests\bin\Debug\net9.0\sba-sfund-tests.dll
總共有 1 個測試檔案與指定的模式相符。
D:\SBA\sba-sfund-tests\bin\Debug\net9.0\sba-sfund-tests.dll
已通過 CM_B13_Create [9 s]
標準輸出訊息:
[CM_B13_Create] Step 1: CM-B13-機構代碼_導航至頁面
[CM_B13_Create] Step 2: CM-B13-機構代碼_點擊新增按鈕
[CM_B13_Create] Step 3: CM-B13-機構代碼_填寫必填欄位
[CM_B13_Create] Step 4: CM-B13-機構代碼_新增成功訊息
[CM_B13_Create] Step 5: CM-B13-機構代碼_驗證新增結果
已通過 CM_B13_Update [5 s]
標準輸出訊息:
[CM_B13_Update] Step 1: CM-B13-機構代碼_搜尋更新目標
[CM_B13_Update] Step 2: CM-B13-機構代碼_選擇資料列進行更新
[CM_B13_Update] Step 3: CM-B13-機構代碼_填寫更新後名稱
[CM_B13_Update] Step 4: CM-B13-機構代碼_更新成功訊息
[CM_B13_Update] Step 5: CM-B13-機構代碼_驗證更新結果
已通過 CM_B13_Delete [6 s]
標準輸出訊息:
[CM_B13_Delete] Step 1: CM-B13-機構代碼_搜尋刪除目標
[CM_B13_Delete] Step 2: CM-B13-機構代碼_選擇資料列進行刪除
[CM_B13_Delete] Step 3: CM-B13-機構代碼_點擊刪除按鈕
[CM_B13_Delete] Step 4: CM-B13-機構代碼_刪除成功訊息
[CM_B13_Delete] Step 5: CM-B13-機構代碼_驗證刪除結果
測試回合成功。
測試數總計: 3
通過: 3
時間總計: 21.7293 秒
sba-sfund-tests 測試 成功 (22.3 秒)
測試摘要: 總計: 3, 失敗: 0, 成功: 3, 已跳過: 0, 持續時間: 22.2 秒
在 27.3 秒內建置 成功

```

```

TestRun_2025-10-09_15-10-26
└─ Screenshots
  └─ CM_B13_Create
    └─ 01_CM-B13-機構代碼_導航至頁面.png
    └─ 02_CM-B13-機構代碼_點擊新增按鈕.png
    └─ 03_CM-B13-機構代碼_填寫必填欄位.png
    └─ 04_CM-B13-機構代碼_新增成功訊息.png
    └─ 05_CM-B13-機構代碼_驗證新增結果.png
  └─ CM_B13_Delete
    └─ 01_CM-B13-機構代碼_搜尋刪除目標.png
    └─ 02_CM-B13-機構代碼_選擇資料列進行刪除.png
    └─ 03_CM-B13-機構代碼_點擊刪除按鈕.png
    └─ 04_CM-B13-機構代碼_刪除成功訊息.png
    └─ 05_CM-B13-機構代碼_驗證刪除結果.png
  └─ CM_B13_Update
    └─ 01_CM-B13-機構代碼_搜尋更新目標.png
    └─ 02_CM-B13-機構代碼_選擇資料列進行更新.png
    └─ 03_CM-B13-機構代碼_填寫更新後名稱.png
    └─ 04_CM-B13-機構代碼_更新成功訊息.png
    └─ 05_CM-B13-機構代碼_驗證更新結果.png

```

▲測試成功，系統功能詳細截圖

參、自我評估與心得感想

在這次的實習過程中，我從基礎開始學習到後期參與完整專案開發，逐步累積了扎實的實務經驗與技術能力，對我而言是一段充實又具成長性的時光。以實際參與

ASP.NET 架構為基礎的系統開發，從前端畫面設計、互動功能撰寫，到後端資料串接與預存程序設計，讓我能將課堂上學習的程式設計與資料庫知識，真正應用於企業專案的開發過程中。

在開發過程中，我運用 HTML、JavaScript、C# 與 SQL Server 等技術，進行前端介面設計與後端資料串接，實作響應式網頁及 CRUD 功能，透過這些經驗，我更理解前後端資料交換的原理，也學會撰寫具結構性與可維護性的程式碼。同時，我也熟悉 Debug 工具的使用，從最初看不懂錯誤訊息，到能透過中斷點定位找出問題，這過程讓我培養了耐心與邏輯思維，並建立起分析與解決問題的能力。

中期階段，我逐漸熟悉專案的開發流程，學會從需求分析、系統規格書閱讀、功能實作到測試驗證的完整步驟，在學姊與同事的指導下，以及閱讀官方技術文件，我能更快速掌握系統架構與開發邏輯。雖然在學校中也學過相關理論，但實務上所面臨的狀況更加複雜，讓我深刻體會到學校教學與業界應用之間的落差。例如，實務開發更重視程式架構的可維護性與團隊協作、系統功能則以模組化設計來提升後續維護與擴充的效率，這些都是在課堂上較少接觸、但非常關鍵的實務觀念。到了後期，我更進一步嘗試結合 AI 輔助開發與自動化測試流程，運用 Google Gemini，大幅提升測試效率與準確性，讓我實際體驗到 AI 技術在系統開發與品質驗證中的應用價值。

除了技術層面外，這次實習也讓我學到職場中更重要的軟實力，透過與同事討論需求、回報進度與分工協作，我培養了良好的溝通技巧與團隊合作能力。在執行任務時，我學會以專業態度面對挑戰，主動尋求協助並提出建議，讓整個開發流程更順暢，也更能理解團隊協作在專案成功中的重要性。這段實習不僅讓我學到實務開發的技巧，也讓我在態度與思維上有了轉變，是踏入職場前最寶貴的養分。