

元智大學資訊管理學系
第三十屆專業實習報告
1141 學期

公司代號：Z

實習單位：神達電腦股份有限公司

輔導老師：劉俞志

姓 名：張景勛、何家佑、黃英綺

學 號：1111705、1111739、1111744

目錄

1111705 張景勛

一、工作內容	3
二、學習內容	6
三、自我評估及心得感想	10

1111739 何家佑

一、工作內容	11
二、學習內容	13
三、自我評估及心得感想	17

1111744 黃英綺

一、工作內容	18
二、學習內容	20
三、自我評估及心得感想	30

1111705 張景勛

一、工作內容

1. 工作環境

在神達電腦實習期間，我對公司的工作環境有著非常好的體驗。辦公室整潔明亮，茶水間設有冰箱、微波爐，還有固定補充的零食櫃，讓人在專注工作之餘也能隨時補充能量。中午用餐選擇也相當多元，除了有提供中式與西式餐點的員工餐廳，還設有販售 7-11 鮮食的販賣機，節省了外出覓食的時間。

健身房與淋浴間的設置也令人印象深刻，讓員工可以下班後直接運動放鬆，維持健康生活型態。公司也採取彈性上班制度，早上 7:30 到 9:30 之間可自由選擇上班時間，對於通勤族來說是非常貼心的安排，也讓我更有效率地規劃每天的節奏。整體來說，神達電腦提供了一個舒適、便利且專業的工作場域，讓我能在穩定的環境中專心學習與實踐技能。

2. 工作內容與任務執行

在神達實習的過程中，我主要負責以下幾項核心任務：

1. 測試案例與測試報告撰寫：

根據專案的功能需求，將具體的測試情境、測試結果與執行步驟，依標準格式撰寫至測試案例與測試報告，明確記錄測試內容、預期結果與實際執行狀況。

測試案例附圖：

Description		測試目標		測試結果		測試日期	
				xunchang		2025/6/20	
用例編號	前置/測試條件	測試步驟	輸入數據	預期結果	實際輸出1	實際輸出2	實際輸出3
1	1. 測試環境：Windows 10, Java 8, Maven 3.6.3, JUnit 4.12.0, Selenium 3.141.0, Chrome 80.0.4012.101	1. 點擊首頁按鈕，進入首頁		顯示首頁內容	OK		
2	2. 測試數據：用戶名：admin, 密碼：123456	2. 輸入用戶名和密碼，點擊登錄按鈕		顯示登錄成功消息	OK		
		3. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		4. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		5. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		6. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		7. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		8. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		9. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		10. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		11. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		12. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		13. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		14. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		15. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		16. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		17. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		18. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		19. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		20. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		21. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		22. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		23. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		24. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		25. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		26. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		27. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		28. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		29. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		30. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		31. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		32. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		33. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		34. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		35. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		36. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		37. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		38. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		39. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		40. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		41. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		42. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		43. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		44. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		45. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		46. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		47. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		48. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		49. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		50. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		51. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		52. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		53. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		54. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		55. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		56. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		57. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		58. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		59. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		60. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		61. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		62. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		63. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		64. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		65. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		66. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		67. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		68. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		69. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		70. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		71. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		72. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		73. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		74. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		75. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		76. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		77. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		78. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		79. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		80. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		81. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		82. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		83. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		84. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		85. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		86. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		87. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		88. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		89. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		90. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		91. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		92. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		93. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		94. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		95. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		96. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		97. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		98. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		99. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		
		100. 點擊登錄按鈕，進入登錄頁面		顯示登錄成功消息	OK		

測試標準		測試用例		測試情境	預期結果	實際結果
案例	編號	描述	說明			
測試標準：測試環境配置	1.1	1. 測試環境配置：測試環境配置	1. 測試環境配置：測試環境配置	1. 測試環境配置：測試環境配置	1. 測試環境配置：測試環境配置	1. 測試環境配置：測試環境配置
	1.2	1. 測試環境配置：測試環境配置	1. 測試環境配置：測試環境配置	1. 測試環境配置：測試環境配置	1. 測試環境配置：測試環境配置	1. 測試環境配置：測試環境配置
測試標準：測試數據	2.1	1. 測試數據：測試數據	1. 測試數據：測試數據	1. 測試數據：測試數據	1. 測試數據：測試數據	1. 測試數據：測試數據
	3.1	1. 測試數據：測試數據	1. 測試數據：測試數據	1. 測試數據：測試數據	1. 測試數據：測試數據	1. 測試數據：測試數據

2. 軟體測試實施：

實際執行系統的功能測試與使用者驗收測試（UAT），不論是手動操作或使用工具進行測試，皆須確保系統行為符合需求規格。針對發現的 Bug 進行回報與追蹤，並與開發人員溝通修正方式。

3. 系統操作手冊撰寫與維護：

針對測試案例撰寫簡明易懂的操作手冊，內容涵蓋登入、模組操作與常見問題處理等。當系統更新時，也會即時同步修改操作手冊內容，確保內部使用者獲得正確資訊。

4. 基礎程式學習：

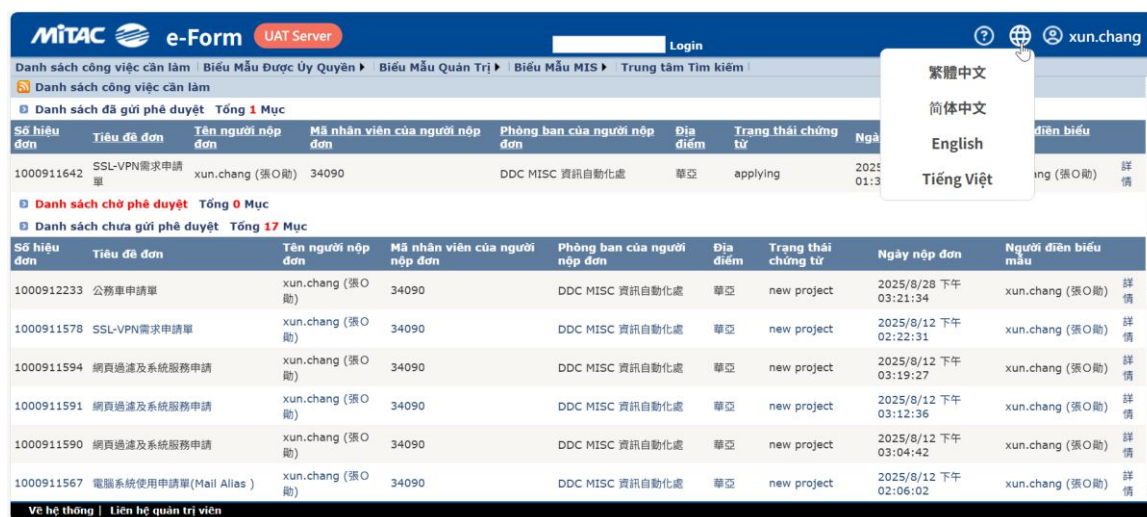
在閒暇時間，我也進行了 Web API 的基礎概念學習。

5. Copilot Studio：

學習並實作 Copilot Studio 搭配 Power Automate 的 AI 自動化流程設計。

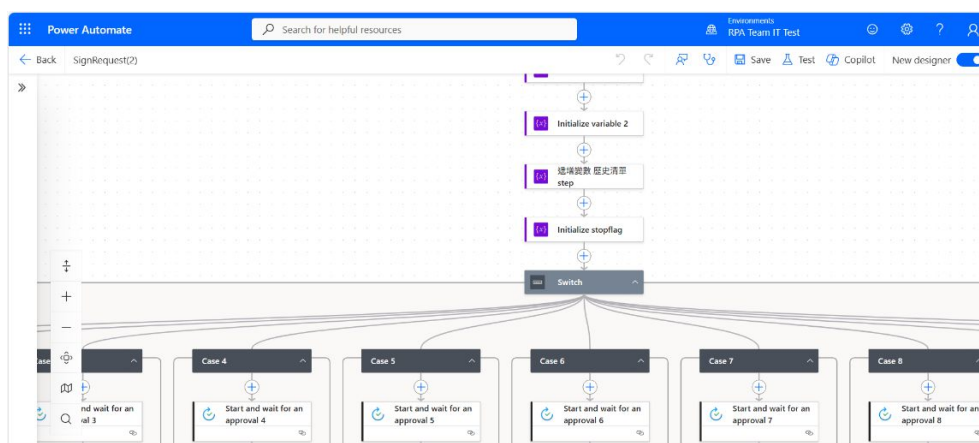
6. Visual Studio 2022 Web Form E-Form 多語系修改：

我後續參與了公司 E-Form 系統（Web Form）的多語系更新與版面統一工作。



7. Power Automate 財務簽核流程開發：

目前我正在協助財務部門設計一套簽核自動化流程，以 Power Automate 為核心實作工具。



3. 實習成果與學習回顧

實習期間，我參與了多項與測試及文件撰寫相關的工作，尤其以 Open PLM 系統的測試案例、測試報告與操作手冊製作為主要任務。透過實作，我成功導入標準化測試文件撰寫流程，使測試品質與一致性顯著提升；並透過多次 UAT 測試，協助開發團隊發現並解決了多項潛在問題。此外，我也參與了公司導入 ChatGPT 的測試工作，了解生成式 AI 在企業場域的應用潛力。

在技術層面，我透過實作初步掌握了 .NET Core Web API 的基礎概念與開發流程，了解如何串接資料庫並使用 Postman 進行 API 測試。這段期間的累積，不僅強化了我的技術基礎，也提升了我在文件撰寫、系統使用者支援及問題追蹤方面的能力。

此外，我進一步接觸 Copilot Studio 串接 Power Automate 的應用，學習如何設計 AI 助理與自動化機制連動，讓我對 AI 在企業應用上的潛力有了更直觀的理解，也提升了我對系統整合與資訊流程自動化的認知。

透過下半年參與的 E-Form 多語系開發與 Power Automate 財務部門自動化簽核流程專案，我從原本偏向測試與文件維護的工作逐步擴展到前端開發與流程設計的領域，並實際操作 GitLab、SonarQube、Power Automate、SharePoint、Adobe 等多項企業工具，對專案管理與程式品質控管有了更完整的體驗。

4. 工作當中扮演的角色

在這段實習旅程中，我所扮演的角色橫跨測試、開發與系統支援等多個層面。我的核心任務是針對 Open PLM 系統進行測試規劃與執行，並撰寫測試報告與操作文件，確保系統功能穩定且使用者能順利操作。除了日常的測試工作，我也參與了公司導入 AI 工具的觀察與測試，這讓我有機會實際接觸企業級 AI 解決方案，並深入理解這類工具在數位轉型中的角色與潛力。

我主動學習如何使用 Copilot Studio 設計 AI 助理，並與 Power Automate 整合以建構自動化流程。這樣的經驗不僅提升了我在系統整合上的理解，也拓展了我對 AI 自動化應用的實際操作能力。在技術學習方面，我也從原本的觀察者逐漸轉變為開發的初學者，嘗試撰寫簡單的 Web API 程式碼與修改 Web Form 程式。這段時間不只是單純地完成任務，而是透過實際參與，讓我一步步建立起對資訊專案流程的整體認識與實務敏感度，也更加確立了我未來從事系統分析與開發工作的興趣與方向。

二、學習內容

一、Project Apply / PLM（產品生命週期管理系統）

瞭解系統作業流程：

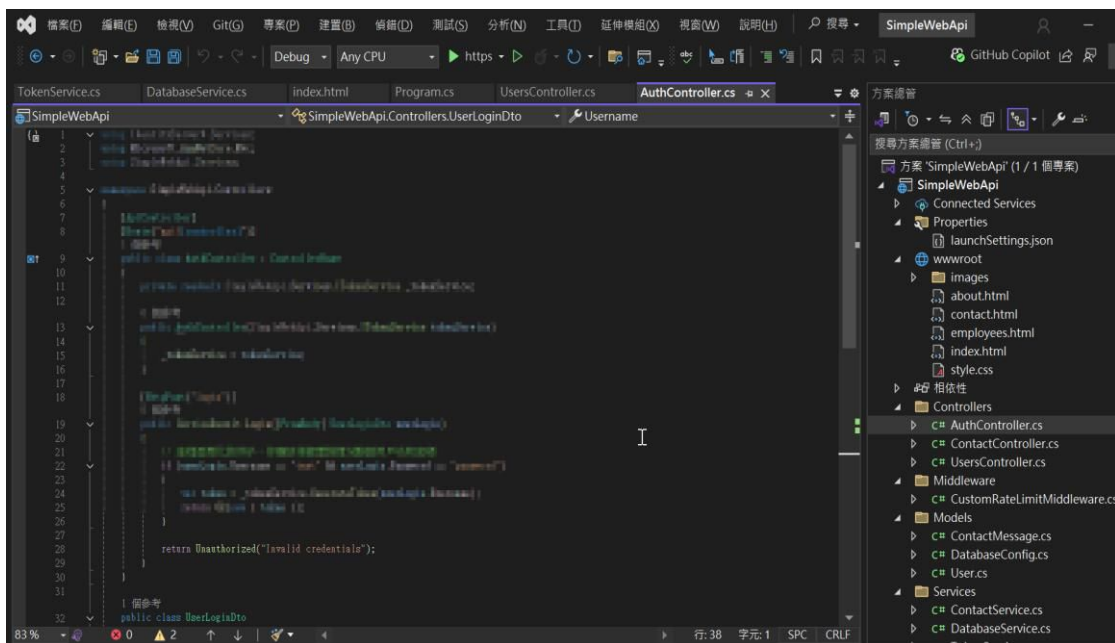
- 透過系統操作與文件閱讀，熟悉了 Project Apply 或 PLM 系統的核心作業流程，例如：專案啟動、需求管理、設計開發、測試驗證、變更管理、版本發布等階段。
- 理解了不同角色（如：專案經理、開發人員、測試人員）在系統中的權限與操作職掌。
- 學習了系統如何支援跨部門協作與資訊共享。



二、Net Core 6 WebAPI

瞭解現行專案架構：

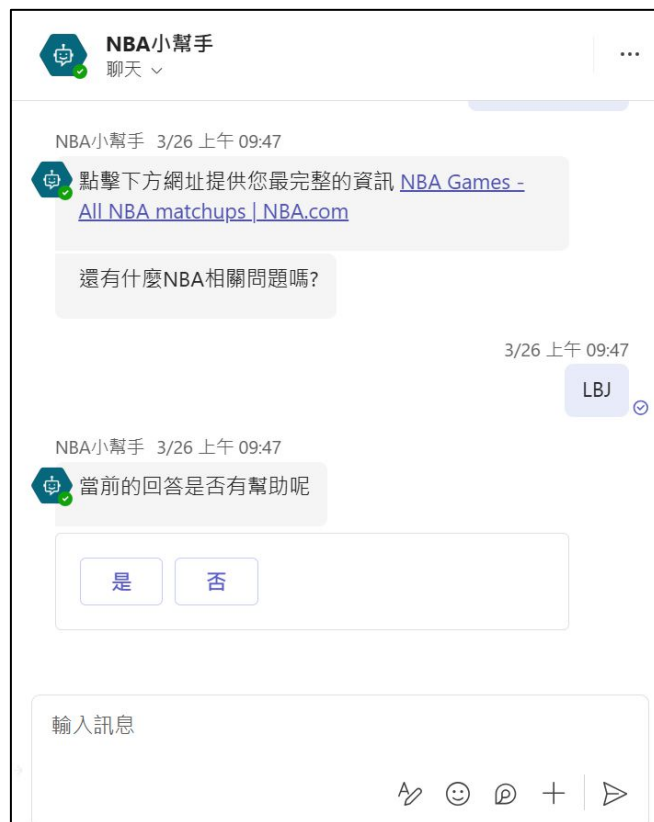
- 深入研究了公司現行專案所採用的 .Net Core 6 WebAPI 架構，理解其分層設計（如：Controller 層、Service 層、Repository 層、Domain 層）的職責與互動方式。
- 學習了依賴注入(Dependency Injection)核心概念在專案中的實際應用。
- 練習使用 Postman 等工具對部署後的 API 進行基本連線與功能測試。



三、Copilot Studio 聊天機器人建置

建立與訓練 AI 自動化對答技術的支援 Bot：

- 學習使用 Copilot Studio 建立自訂聊天機器人，並設計其回應邏輯、觸發條件與使用者意圖辨識流程。
- 實作針對特定內容（如 NBA）的問答流程，並建立相關知識庫，提供使用者快速獲得常見問題的解決方案。
- 熟悉如何整合多個資料來源（如網站內容、PDF 手冊、內部 FAQ 文件）並轉化為可供機器人檢索的內容。
- 測試不同對話流程，並根據實際使用情況進行調整，提升 Bot 的準確率與回應效率。



四、E-Form 多語系修改與前端調整

熟悉多語系系統架構與資源檔維護：

- 使用 Visual Studio 2022 對公司內部 E-Form 系統進行多語系化改版，熟悉 .resx 資源檔的維護流程。
- 新增越南語語系內容，並校對繁體中文、簡體中文與英文翻譯的正確性，確保介面顯示與文意一致。
- 修改 E-Form Header UI，統一各頁面語言選單設計，讓不同模組間的使用體驗一致。
- 利用 SonarQube 進程式碼品質掃描與缺陷檢查，確保修改後無潛在錯誤，再將更新版本推送至 GitLab 分支供主管審閱。

The image displays two screenshots related to the E-Form system's multilingualization and code quality.

The top screenshot shows the Visual Studio 2022 interface with the `SearchCenter.aspx.resx` file open. The file contains a table of resource names and their translations for different languages.

名稱	中性值	vi	zh-cn	zh-tw
apply_date	ApplyDate	Ngày nộp đơn	申請日期	申請日期
btnReset.Text	Reset	Xóa bỏ	清除	清除
btnSearchById.Text	Search	Tra cứu	查詢	查詢
id	Form No.	Số hóa đơn	单据号码	單據號碼
lblAppDate.Text	ApplyDate	Ngày nộp đơn	申請日期	申請日期
lblApplyList.Text	Search List	Danh sách tra cứu	查詢清單	查詢清單
lblApplyTotal.Text	total	cùng	共	共
lblApplyUnit.Text	service(s)	bút	笔	筆
lblCompany.Text	Company	công ty	公司	公司
lblFormNo.Text	Form Type	Số hóa đơn	单据号码	單據號碼
lblFormType.Text	Event Type	Loại biểu mẫu	表单类型	表單類型
lblLocation.Text	Location	Địa điểm	地点	地點
lblProposer.Text	Applicant	Người nộp đơn thực tế	实际申请人	實際申請人
lblScore.Text	Score	Đánh giá	评分	評分
lblStatus.Text	Status	Trạng thái	状态	狀態
lblWriter.Text	Creator	Người điền đơn	填單人	填單人
lbtnCNLangResource1	English		简体	簡體
location	Location	Địa điểm	地点	地點
pnlSearchByDept	Department	Bộ phận	部門	部門
proposer_dept	ApplicantDept	Bộ phận người nộp đơn	申請人部门	申請人部門

The bottom screenshot shows the SonarQube interface for the project `OA-EFORM`. The `QUALITY GATE STATUS` is **Passed**. The `MEASURES` section shows the following metrics:

- New Code: Since last scan ITIL20... Started 17 days ago
- Overall Code: 0 New Bugs, 0 New Vulnerabilities, 0 New Security Hotspots, 0 Added Debt, 0 New Code Smells
- Reliability: A
- Security: A
- Security Review: A
- Maintainability: A

The bottom of the SonarQube interface shows the `Coverage` on 58 New Lines to cover (0.0%) and `Duplications` on 420 New Lines (7.14%).

五、Power Automate 財務簽核流程自動化開發

1)設計與實作自動化簽核系統：

- 依據財務部門需求，規劃一套完整的簽核流程自動化系統，建立三個 SharePoint 表單（固定簽核設定、使用者填寫簽核資訊、簽核紀錄追蹤）。
- 使用 Power Automate 實作簽核流程邏輯，讓系統能自動發送 Microsoft Teams 通知並即時更新 SharePoint 狀態。
- 設計條件分支（Condition）與自動化路徑，使簽核層級能自動指派給正確主管，實現流程智慧化。
- 正在開發 Adobe 浮水印自動嵌入簽核文件與「簽核被拒絕及取消後可重新送審」機制，建立流程閉環與容錯設計。

+ 新增

頁面詳細資料

預覽

沉浸式閱讀程式

草稿儲存日期 2025/10/31

SignRequest

查看全部

+ 新增項目

在方格檢視中編輯

共用

複製連結

匯出至 Excel

所有項目

Ticket ID	申請人	申請主旨	申請內容	會簽	簽核類別	簽核狀態
RFP20251031-0007	xun.chang (張景勛 - MIC)	電腦購買申請	asdf		A	已拒絕

SignRequest_history

查看全部

+ 新增項目

在方格檢視中編輯

共用

複製連結

匯出至 Excel

所有項目

Ticket ID	Step	Role	Responsibility	Arrival date	Processing date	Execute Result	Remark
RFP20251031-0007	1	Requestor	xun.chang (張景勛 - MIC)	2025/10/31 PM 03:47	2025/10/31 PM 03:47	單據已成功建立	

核准

核准要求詳細資料

已拒絕

電腦購買申請

asdf

最後狀態: 已拒絕

步驟 3:拒絕者

shelly.huang (黃英綺 - MIC)

2025/10/31 下午 03:15:10

步驟 2:核准者

neo.he (何家佑 - MIC)

2025/10/31 下午 03:15:00

步驟 1:核准者

xun.chang (張景勛 - MIC)

2025/10/31 下午 03:14:39

要求者

xun.chang (張景勛 - MIC)

2025/10/31 下午 03:14:26

另存新檔為 PDF

三、自我評估及心得感想

這次在神達電腦的實習，是我深入了解企業實務運作與自我成長的重要歷程。前期主要參與 Open PLM 系統的測試與文件撰寫，學會根據需求文件撰寫測試案例，進行單元測試、整合測試與使用者驗收測試（UAT），並撰寫清晰的測試報告以確保系統穩定與可追蹤性。透過這些任務，我培養了對系統功能驗證的邏輯思維與細節觀察力。

在中期階段，我開始接觸 .NET Core WebAPI 架構，了解控制層與服務層間的互動機制，並利用 Postman 進行 API 測試，為日後的開發打下基礎。同時，我也嘗試使用 Copilot Studio 建立簡易 AI 聊天機器人，體驗企業導入生成式 AI 的可能性與應用方式。

進入下半年度後，我的學習重心逐漸轉向開發與流程自動化。參與 E-Form 系統多語系新增及 UI 的修改，使我熟悉了實際專案中的程式碼修改與程式碼品質控管；而在 Power Automate 財務簽核專案中，我更從零規劃出自動化邏輯、表單架構與流程串接，實際落實跨系統整合。這不僅強化了我的邏輯設計與問題解決能力，也讓我體會到如何將企業實際需求轉化為技術實踐。

在這段實習期間，我不僅累積了軟體測試與系統開發的專業技能，也學會了職場中最重要的軟實力——包括時間管理、跨部門溝通、任務回報與主動學習的態度。神達電腦的工作環境與團隊氛圍，讓我在穩定中成長，也讓我確立了未來想朝向系統分析、AI 自動化與企業流程整合方向發展的目標。這段實習經驗不只是技術的磨練，更是我邁向專業工程師之路的重要起點。

1111739 何家佑

一、工作內容

1. 工作環境介紹：

神達電腦（MiTAC）是台灣知名的科技公司，主要聚焦於伺服器、車用電子、智慧裝置等領域的研發與製造，具備強大的系統整合與硬體開發能力，在國際市場上也有一定的能見度。公司以 B2B 為核心業務，客戶多為大型企業，因此對產品的穩定性與客製化程度要求相當高。身為實習生，能在這樣以技術為主體、重視效能與創新的環境中學習，是很難得的機會。

辦公室設施方面，神達電腦的工作環境非很好，中午用餐也相當方便，員工餐廳提供中式、西式餐點，今年公司內也新開了 7-11 可供選擇，。健身房設施很齊全，下班可以直接運動，還有淋浴間，對注重健康的人來說很加分。另外，茶水間和廁所都非常乾淨整齊，茶水間還有冰箱方便員工存放食物與飲品，零食櫃也會定期補充，累的時候隨手就能補充點能量。公司的彈性工時制度也讓生活跟工作更容易平衡，早上 7:30 到 9:30 之間彈性上班，只要工作滿八小時就能準時下班，如此一來就可以避開上下班的車流尖峰。

而在團隊氛圍方面，同事之間相處融洽，樂於互助，當遇到不懂的地方，總能獲得耐心且具啟發性的指導，不僅解決當下問題，也幫助我理解背後的原理並培養獨立處理的能力，而主管也非常親切並樂於指導，整體的工作氣氛相當良好。

2. 工作詳述：

1. 測試計劃及測試報告書撰寫：

根據專案需求與規格，規劃詳細的測試案例與步驟，編寫標準化的測試計劃書，明確測試範圍、資源、時程及預期成果。執行測試後，彙整結果並撰寫詳盡的測試報告書，記錄發現與結論，確保品質參考的可靠性。

2. 軟體測試：

執行功能測試、整合測試及使用者驗收測試（UAT）等階段的測試活動，透過測試工具或手動方式驗證軟體是否符合需求規格。發現缺陷（Bug）時進行回報與追蹤，與開發團隊協作解決問題，以確保軟體品質並提升使用者體驗。

3. 系統操作手冊撰寫與更新：

為現有系統或新功能編寫清晰易懂的操作手冊，包含系統登入、功能模組操作步驟及常見問題排除方法。隨著系統更新或功能調整，即時維護並更新手冊，確保內容正確且具時效性，為使用者提供準確指引。

4. 程式撰寫：

- Web API 撰寫與系統開發：

參與後端 Web API 的設計與開發，使用 C# 搭配 Dapper 進行資料存取與邏輯實作，確保資料傳輸正確與效能穩定。依據專案需求進行 API 測試與除錯。

- WebForm 修改與系統維護：

針對現有 WebForm 系統進行多語系、功能調整、UI 優化等，依據實際使用情境修正前端介面與後端邏輯，確保系統穩定運作並提升操作體驗。

3. 實習期間完成之進度：

在實習過程中，我逐步熟悉並完成了以下主要任務與學習目標：

- I. 測試文件標準化：成功導入並執行了標準化的測試計劃與報告書撰寫流程，提升了測試文件的品質與一致性。
- II. 軟體品質確保：透過嚴謹的軟體測試流程，有效地發現並協助解決了多項系統潛在問題，確保了數個專案版本的順利上線。
- III. 操作文件完善：完成了許多流程的操作手冊撰寫，並針對內部系統進行了文件同步更新，提高了使用者的操作效率。
- IV. 程式開發參與：協助完成公司內部的後端程式開發，並參與多個內部系統的修改與系統維護維護與調試，並且運用 AI 大幅提升開發效率，
- V. 技術能力提升：透過實際操作與學習，掌握了 .Net Core WebAPI、.Net Web Forms、Gitlab 版本控制、自動化流程等專業技能。

4. 工作當中扮演的角色：

在這次實習期間，我扮演著多重角色，深入參與團隊的各項任務，並在實務中累積寶貴經驗。身為測試執行者，我負責執行功能測試、整合測試及使用者驗收測試（UAT）等工作，透過手動測試與工具操作，確認系統功能是否符合需求規格，確保產品在交付前具備良好品質與穩定性。

除了實際執行測試，我同時負責撰寫並維護測試計劃書、測試報告及系統操作手冊，確保各階段的資訊能被準確傳遞與保存，讓團隊與使用者都能清楚掌握系統功能與操作方式。

在開發方面，我積極參與 Web API 的撰寫與既有 WebForm 系統的修改與維護，透過實作強化對系統架構與資料流程的理解。同時，我也主動運用 AI 工具輔助開發流程，例如利用 AI 協助程式撰寫與除錯、提升文件產出效率，並在多語系功能的調整中協助公司導入資源檔管理工具，大幅提升了整體翻譯與維護的效率。

身為一位實習生，我持續主動吸收新知，善用每一次與同事協作的機會，加深對專案背景與技術細節的理解，並積極參與討論與任務執行，與團隊共同推進專案進度，確保目標如期達成。

二、學習內容

1.NET Core 6 WebAPI 專案架構與開發實作

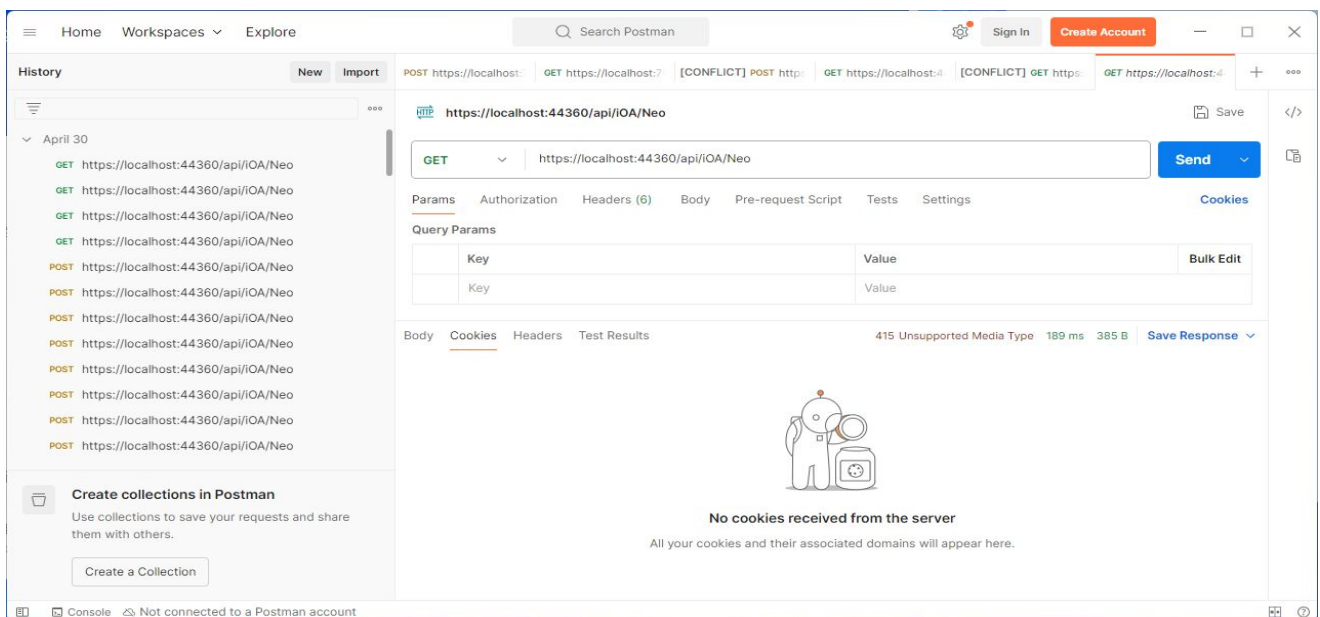
在實習期間，我深入瞭解公司現行使用的 .NET Core 6 WebAPI 架構，透過實際參與專案開發與維護，我逐步掌握了分層設計的概念，包含 Controller 層負責接收與回應請求、Service 層負責業務邏輯處理、Repository 層進行資料存取、以及 Domain 層定義資料模型與邏輯規則。這些層之間透過依賴注入（Dependency Injection）方式降低耦合，提升維護性與可擴充性。

我也學會了如何使用 Swagger 工具自動產出 API 文件，方便團隊成員理解並測試 API 的使用方式。在開發與測試過程中，我使用 Postman 工具進行 API 測試，檢查各項功能是否如預期運作，並能正確處理錯誤回應。此外，我也熟悉了如何將專案部署至本地及測試伺服器，並透過 appsettings.json 管理不同環境的設定，例如資料庫連線字串與第三方服務金鑰。

資料庫部分，專案採用了 Oracle SQL 作為後端資料庫，我搭配 Dapper 作為 ORM 工具，進行資料查詢與對應資料模型的設計與撰寫，並學習撰寫複雜查詢語句與效能優化方式。

```
[HttpGet]
public async Task<ActionResult> GetNews([FromHeader] HeaderParameters headers)
//可在調整成可以前後端共用，可以過濾部分欄EX 公告過期後，前端不顯示
var result = new ReturnObject<EncryptViewModel>();
try
{
    將來源參數，依api需求，進行相關處理
    回傳值
}
catch (Exception ex)
```

(WebAPI 撰寫)



(使用 Postman 進行 API 測試)

2. PLM 系統（產品生命週期管理）流程理解

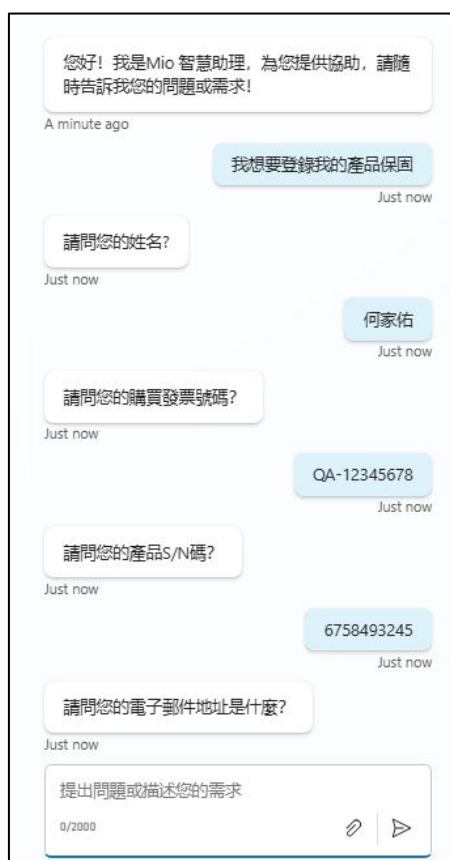
在 PLM 系統（Product Lifecycle Management）的學習中，我透過實際操作與文件閱讀，熟悉了公司內部的專案作業流程。從專案啟動、需求收集、設計開發、測試驗證，到後續的變更管理與版本發布，每一個階段都有明確的流程與審核機制。系統中不同角色（如專案經理、開發人員、測試人員）依據各自職責進行操作與資料更新，形成一套完整且可追蹤的作業模式。

我也觀察到 PLM 系統如何協助跨部門進行協作與資訊共享，透過權限控管與階段性文件管理，確保所有參與人員皆能掌握最新的專案狀態與版本內容，進而提升整體效率與品質控管能力。

3. Copilot Studio 聊天機器人建置與知識整合

在數位轉型的背景下，我嘗試學習建置一套企業內部的技術支援聊天機器人，使用 Microsoft 的 Copilot Studio 平台進行設計與訓練。這套聊天機器人專門用來回應與公司產品相關的技術問題，並提供使用者快速獲得解答的管道。

我學會如何建立與管理 Bot 的回應邏輯，包含意圖辨識、條件判斷與多輪對話流程的設計。此外，我也整合了多種資料來源，包括內部 FAQ 文件、PDF 手冊與公司網站內容，將其轉化為機器人可讀取的知識庫。



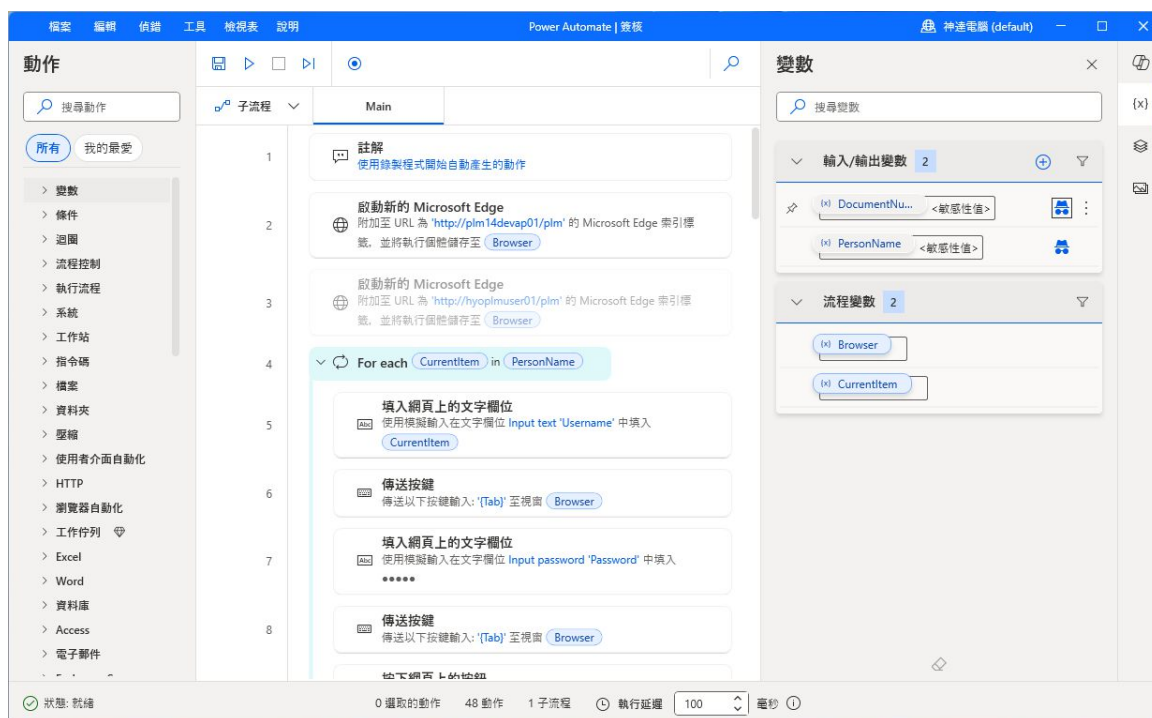
(ChatBot 對話範例)

4. Power Automate 自動化流程設計與測試簽核整合

在參與 PLM 系統測試的過程中，我觀察到測試人員在執行流程驗證時，經常需要手動切換不同角色帳號來進行簽核操作，這不僅耗時，也容易出現操作遺漏或錯誤，造成測試效率低落。為了解決這個問題，因公司有提供學習資源，我主動額外學習來導入 Power Automate 設計自動化簽核流程，成功簡化原本繁瑣的測試步驟。

我針對 PLM 系統的測試環境設計了一套流程，當某個簽核條件達成後，Power Automate 便會自動模擬對應角色的簽核行為，依照順序完成簽核流程，無需人工登入切換帳號。透過這種方式，我能快速模擬多角色簽核場景，大幅提升測試效率。

這項應用不僅讓 PLM 測試作業更加流暢，也展現了 Power Automate 在實際業務流程中的靈活性與實用性。透過這次實作，我更深刻理解了如何運用自動化工具來解決實際問題，並且具備設計、串接與優化自動流程的能力，為日後導入更完整的測試自動化架構奠定基礎。



(測試自動化流程)

5. WebForm 系統多語系&UI 優化

在實習期間，我參與了多個以 ASP.NET WebForm 為基礎的系統維護與改版專案。根據公司需求，我負責將既有系統導入多語系功能，包括在原有語系基礎上擴充，或針對完全未支援多語系的系統進行全新開發。除了多語系功能，我也依照公司最新的 UI 設計規範，對舊系統進行介面更新與調整，使整體操作體驗更加一致、現代化，並兼顧使用者便利性與視覺美觀。

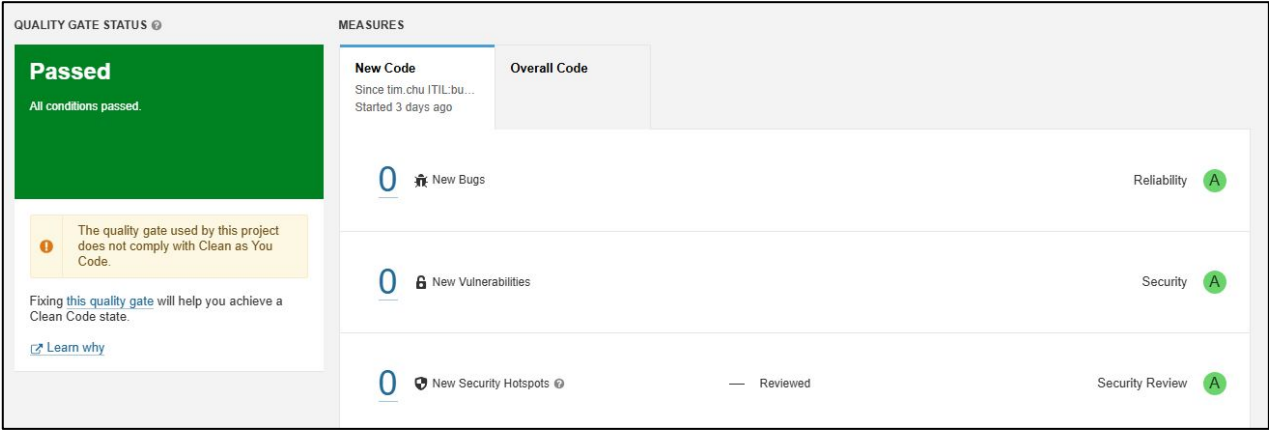
在實際開發過程中，我發現過往建立與維護多語系資源檔的流程效率低落，因此主動尋求改進方案。透過詢問 AI 工具，我引入了 ResX Manager 以及快速建立本機資源檔的方法，大幅簡化多語系作業流程。經過改進後，原本需約十個工作天的多語系開發作業，現在只需兩個工作天即可完成整個系統的語系建置。同時，我將此流程整理成使用手冊，提供團隊成員參考，確保整體開發流程能被延續並維護品質。

在開發過程中，我不僅撰寫與修改程式碼，也負責測試與除錯，並使用 SonarQube 進程式碼品質檢測，確保程式符合統一規範、維護性高且易於後續擴充。此外，我也學習了與資料庫的互動方式，包括查詢多語系相關資料與維護資料一致性，確保前端顯示的內容能正確對應後端資料。

這段經驗不僅讓我熟悉 WebForm 系統的架構與開發流程，也強化了我在多語系功能、UI 改版、效率改善與程式碼品質管理等方面的實務能力。透過實作與團隊協作，我逐步建立了完整的開發思維，並學會如何將技術知識與工作流程相結合，提升專案效率與整體品質，為公司舊系統的更新與功能優化做出實質貢獻。



(系統多語系&UI 優化範例)



(SonarQube 掃描結果範例)

三、自我評估及心得感想

在神達電腦的實習過程中，我一開始主要負責學習與撰寫 Web API。剛開始接觸時，學習進度算是蠻快的，但隨著深入了解公司實際的程式架構，像是 依賴注入 (Dependency Injection) 這類較進階的技術出現後，我才發現企業端的開發遠比學校教的複雜許多。這是我第一次接觸完整的企業程式碼，也因此深刻體會到學校所學只是基本功，要在職場中真正站穩腳步，還需要不斷磨練與提升。透過實習，我也首次接觸到開發規範，學會在團隊合作中遵守既定標準的重要性，這對協作與維護都非常關鍵。

隨著實習進展，我逐步熟悉了專案流程。在參與 WebForm 系統的多語系開發時，我發現既有的建立流程繁瑣且效率低落。但我沒有只是被動地接受任務，而是主動思考如何提升效率，在研究後引入了 ResX Manager 並找到了自動建立本機資源檔的方法。這個嘗試非常成功，成功將原本預估需要十個工作天的流程，大幅縮短到兩天內即可完成。這個經驗讓我意識到，實習生的價值不僅是完成任務，更是能主動為團隊的流程帶來優化。

在實務操作中，我也逐漸了解到許多開發需要注意的細節。例如，應該更加逐步理解程式碼脈絡與執行順序，當程式出現問題時，不應過度糾結於單一點，而是重新檢視整體流程；在工作上，若想嘗試新的方法，也需要先設定止損點，避免因新方法失敗而耽誤整體進度。結合以上讓我意識到時間管理的重要性，過多耗費在非重點或瑣碎細節上，可能會拉低整體效率，即便原本效率不錯也會受到影響。

除了開發工作，我也多次參與 PLM 系統的測試，撰寫測試報告與使用者手冊。這讓我體會到，從使用者角度撰寫文件並不容易，需要換位思考，才能真正提供有價值的參考。同時，我觀察到許多測試流程需要手動切換帳號進行簽核，不僅耗時也容易出錯。因此，我主動利用 Power Automate 建立自動化程序，使測試與操作更加高效。這讓我體會到，主動將所學應用於解決團隊的實際痛點，能帶來巨大的價值。

這段實習經驗讓我在專業技能、專案流程理解、開發效率、程式碼思維、時間管理等方面都獲得實務鍛鍊，也清楚看見自身的不足與待加強之處。但我最大的收穫，是心態上的轉變：我學會了不僅是被動地完成被交辦的任務，主動去『發現問題、尋找工具、並解決問題』是非常重要的。

我真心覺得神達是一個很好的實習環境，不僅提供實作機會，也會給予學習與思考的空間，鼓勵我主動尋找更有效率的方法、改善工作流程，並將創新想法應用於實務，也讓我學會如何在團隊中成長、協作與主動貢獻。

1111744 黃英綺

一、工作內容

1. 工作環境介紹：

神達電腦為專注伺服器、車用電子與智慧應用領域的科技公司，以 B2B 解決方案為核心，客戶多為國際大型企業，因此對品質與客製化有高度要求。能在這樣注重效能與技術整合的環境中實習，是非常寶貴的學習機會。公司提供完善且友善的工作環境，包含員工餐廳、販賣機與乾淨舒適的休憩空間；同時設有健身房與淋浴設備，讓員工在工作之餘也能兼顧生活品質。茶水間及公共區域維持整潔，並提供冰箱與零食補給，為日常工作帶來便利。彈性工時制度讓上下班時間更具彈性，除了提升工作效率，也能有效平衡生活節奏。團隊間氣氛融洽且樂於協作，同仁對問題詢問與技術指導都非常有耐心，從系統操作到實務觀念皆願意分享，使初入職場的我能快速適應與學習，對我的職場發展有很大的幫助。

2. 工作內容詳述：

- **軟體測試執行：**

依據產品生命週期，參與功能測試、整合測試與使用者驗收測試等階段。透過手動測試或輔助工具檢查系統功能的完整性，並對問題進行記錄與追蹤。在測試過程中，與開發人員合作排除錯誤，協助提升產品穩定性與使用者體驗。

- **測試案例與報告撰寫：**

根據各項專案開發需求，撰寫測試情境與細項流程。包括制定測試目標、測試各種情境，預期結果等。測試執行完成後，透過測試結果，撰寫報告並提供最後測試結果，作為品質保證的依據。

- **文件撰寫與維護：**

負責撰寫系統操作手冊，包括操作流程、常見問題與解決步驟等，內容圖文並茂，利於使用者快速上手。針對每次系統更新，皆即時同步修訂文件，確保資訊準確性與可用性。

- **多語系功能開發：**

負責新增 E-Sop 系統的多語系功能以及部分 UI 介面調整。

使用 Git clone 將專案儲存至本機，再開啟 VS2022 進程式開發，先建立翻譯內容的資源檔，並將原本程式碼替換成根據所選語言對應的資源檔的內容，並統一網頁介面方便後續與其他系統方便管理，修改完後在使用 SonarQube 進程式碼掃描，確認沒問題之後再 Git 回分支。

- **使用者介面優化：**

透過與使用者訪談，了解使用者操作習慣，以使用者角度去調整搜尋欄位的順序以及表單介面的配置。並撰寫管理員操作手冊，將調整的步驟轉寫成一份檔案，可以供後續同仁參考。減少使用者操作上的步驟，讓管理員也可以降低後續調整的時間成本。

撰寫匯入匯出工具的操作手冊，根據操作步驟將開發環境調整好的內容匯出並匯入至 UAT 環境，該分文件可以提供給其他同仁參考，幫助後續環境部署。

3. 實習成果與進度：

- **測試流程制度化與品質提升**

協助建立測試標準與報告格式，讓測試流程更一致、可追蹤。實際參與多項測試執行與缺陷修正，提升系統穩定度與整體產品品質。

- **文件與知識管理完善**

撰寫並維護多份系統操作手冊與測試文件，內容清晰易懂，協助團隊成員快速熟悉新系統，提升文件管理效率。

- **多語系與介面優化成果**

參與 E-Sop 系統多語系功能開發與 UI 介面統整，成功導入多語支援，優化操作流程並強化系統一致性與使用體驗。

- **系統整合與流程驗證經驗**

參與 PLM 系統更換測試，撰寫測試案例並追蹤問題修正，確保新系統功能與舊系統一致，為專案上線品質把關。

4. 實習中的角色定位：

- **軟體測試與品質驗證人員**

執行各階段測試，驗證功能正確性與穩定性，並提交測試報告提供決策依據。

- **技術文件與知識撰寫者**

撰寫系統操作手冊與測試報告，確保內容完整一致，協助團隊於系統升級與轉換期間順利銜接。

- **使用者支援與介面優化協作者**

根據使用者回饋進行介面調整與改善，優化欄位順序與表單配置，提升使用便利性。

- **跨部門協作與學習型成員**

主動與開發人員、測試團隊溝通協作，吸收實務經驗，培養問題分析與解決能力。

二、學習內容

一、PLM 系統：

在實習期間學習並實際操作 PLM (Product Lifecycle Management) 系統，了解其在企業研發與產品管理中的關鍵角色。此系統支援產品從設計構想到量產、維護與淘汰的完整生命週期，具備資料集中、流程可控、跨部門協作等優點，對產品品質與時效性有顯著提升。

產品生命週期管理 (PLM) 系統涵蓋零件管理、物料清單 (BOM) 管理、變更管理、文件管理以及 Creo 協同合作功能。此系統將產品設計、開發、生產等相關資訊同步整合至其他系統，例如專案管理系統 (Project Apply)、團隊管理系統 (TMS)、企業資源規劃系統 (ERP)、以及綠色產品系統 (GP)，以促進後續的產品研發、開發、量產及變更過程。

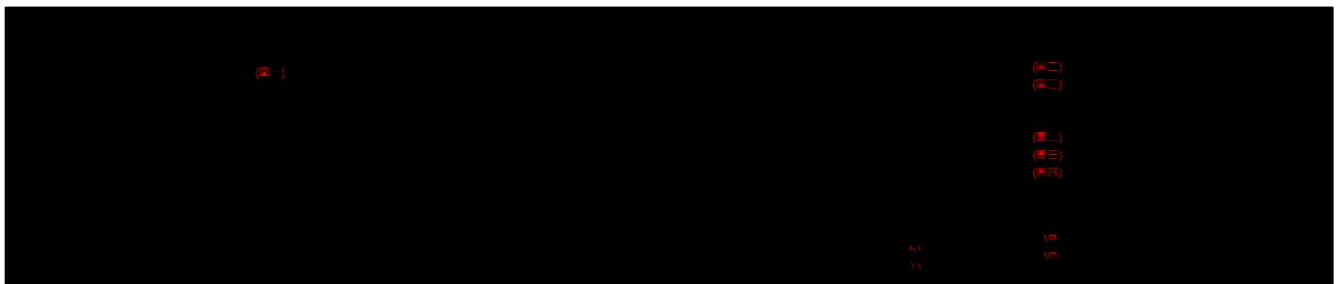
1. 零件結構與生命週期管理：在 PLM 系統中建立零件、查詢零件、管理零件狀態及零件資料等。
2. 客製功能：在公司 PLM 系統中有一些專屬客製流程可以提升工作效率，像是可以快速複製零件資料及屬性；或是利用 Excel 建立 BOM，透過系統的檢查機制完成批量操作的繁瑣流程。
3. 變更管理與簽核流程：熟悉 PLM 系統中各種變更單的操作，透過不同單據啟動不同簽核流程，每張變更單需依層級流程進行簽核。此外，系統整合 TMS，根據 TMS 設定檔設定的專案或分類節點，在送審單開始簽核時，會自動抓取對應的簽核角色及人員。
4. 綠色產品：為了響應 ESG，PLM 系統結合 GP 檢驗，製作綠色產品。
5. 歷史追蹤與流程檢視：學會查看該零件的歷史異動紀錄，或是透過工作流程圖查看目前簽核流程到哪個階段、有哪些父階料號等等。
6. 額外支援任務：

- **更換系統測試：**

由於公司要更換 PLM 系統，因此協助參與各項流程測試，確保流程與原系統皆相同。

- **撰寫測試案例與測試報告：**

寫出所有操作情境及預期結果，協助測試流程是否正確，再將測試結果整理成測試報告，確保每項流程功能皆沒問題。



▲附圖為測試案例的檔案，會有各種情境及對應的預期結果，根據這份檔案去進行相關測試，每項情境都會測試三次，確保相關功能沒有問題

測試編號		測試案例		測試情境	預期結果	實際結果	測試日期	測試時間	整合測試編號	原系統版本	角色人員	測試人
編號	案例	編號	案例									
						無法建立EC，無法上傳、查閱			通過-測試			
						無法查看問題狀態、無法移動及刪除問題			通過-測試			
						無法選擇問題狀態			通過-測試			
						無法建立問題狀態、無法刪除及上傳			通過-測試			
						無法建立文件、無法刪除及上傳			通過-測試			
						無法刪除文件、無法刪除及上傳			通過-測試			
						無法刪除文件、無法刪除及上傳			通過-測試			
						無法刪除文件、無法刪除及上傳			通過-測試			
						無法刪除文件、無法刪除及上傳			通過-測試			
						無法刪除文件、無法刪除及上傳			通過-測試			

▲附圖為測試報告，該報告展示了透過測試案例執行的最終結果

● 協助測試問題追蹤：

將測試問題列入至問題清單，定期確認問題是否已修復完成，若完成，則需要進行複測，覆測通過才能確保所有功能正確。

功能模組	問題分類	優先級	描述	提出人	提出日期	狀態	責任人	確認	完成日期	MITAC覆測
						In Progress		目前已在Submit		OK
						Completed		後續優化的方式		
						Pending		已調整。		OK 已設定預
						Completed		已調整，確認是		覆測NG,並
						In Progress		已調整。		OK
						Completed		已調整。		OK

▲附圖為問題列表的檔案，說明問題內容、建立問題人員、提日期、問題修改狀態、修改後覆測結果等資訊

● 撰寫使用者操作手冊：

由於公司更換系統，公司成員需要重新適應新系統，撰寫相關操作手冊，可以幫助員工適應系統轉換。

建立 QVL Apply Form

- 在目錄中尋找ECO-工程變更命令
- 點選 新增ECO-工程變更命令
- 分類選擇 QVL Apply Form
- Change Type選 QVL Apply
- 輸入其他必填欄位：Project Code、Model Name、Spec. Check、Control Run、System Level Test、Phase In Type

▲附圖為建立變更單的操作流程

- 使用者介面優化：

- 調整查詢欄位順序

透過使用者訪談，了解使用者操作系統的習慣，並根據欄位重要性以及使用者常用欄位調整查詢順序，讓使用者可以漸少查詢的時間成本，快速找到資料。



▲附圖為調整搜尋欄位順序的前後比較圖

- 調整表單介面的配置

以使用者的角度，將重要欄位以及必填欄位集中，並排列整齊，讓使用者可以操作更流暢，讓介面更清晰，避免需要頻繁捲動畫面查看重要欄位。



▲附圖為調整表單介面的前後比較圖

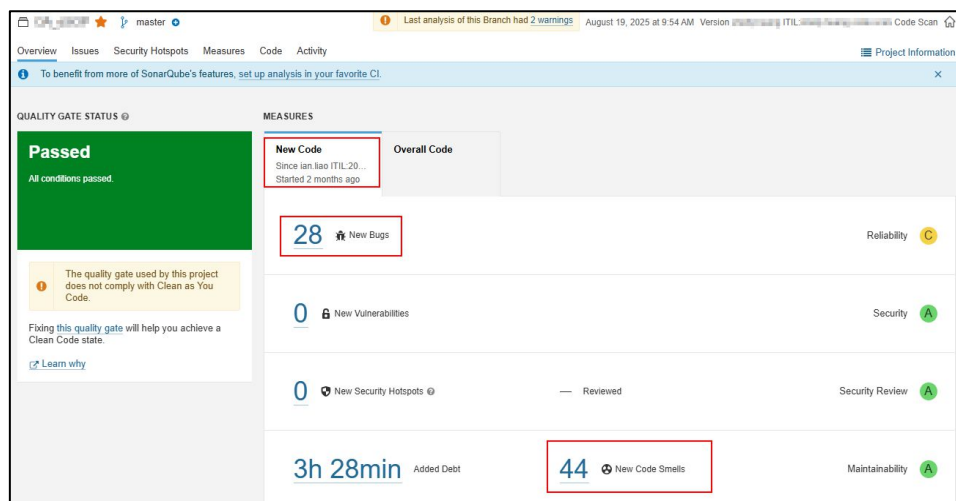
二、E-Sop 系統多語系功能開發：

使用 Clone 將程式碼 Git 到本機並建立新分支，接著進行多語系功能開發。首先，建立和配置專案所需的資源檔，將所有介面上的固定文字內容存入其中。隨後，修改原始程式碼，將原始的文字內容替換為對應資源檔中的 Key，確保在切換語言時，能依據 Key 抓取當前語言的內容。完成多語系功能之後，也同步調整 UI 介面，將 E-Sop 介面格式與其他系統統一，以提升使用者體驗並方便後續的管理。



▲附圖為 UI 介面調整的前後比較圖

所有開發和介面調整完成後，為符合資安標準，需使用 SonarQube 進程式碼掃描，檢測潛在的 Bug、Code Smells。若掃描結果發現任何高風險或需要優化的程式碼，將立即進行修正。最終確認程式碼品質與安全性無虞後，再將程式碼 Git 回分支，提交給主管進行後續的審核與合併流程。



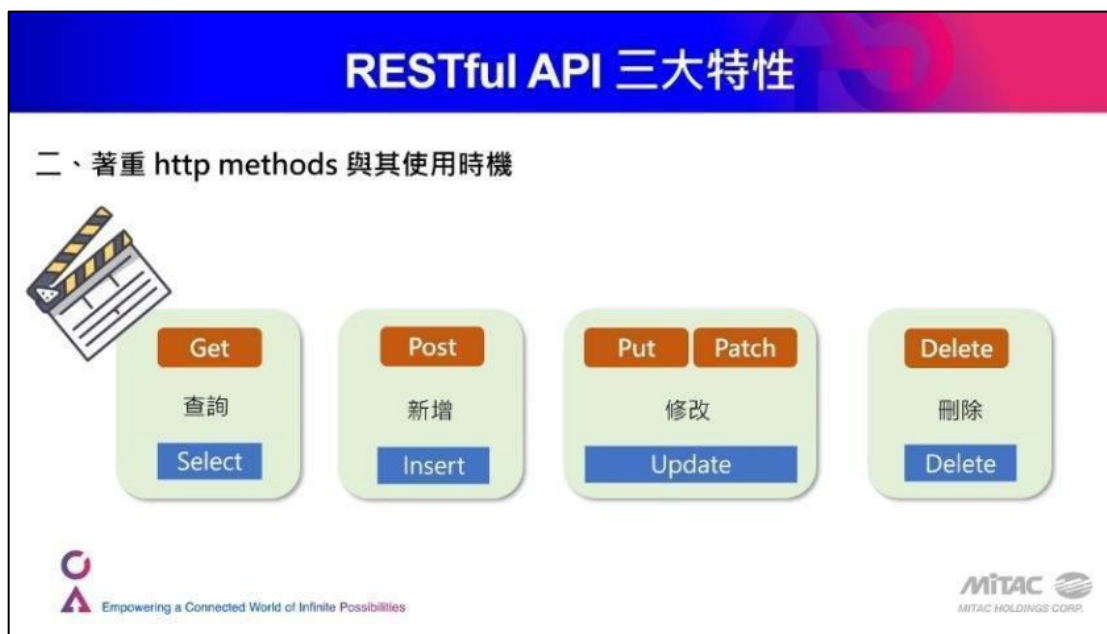
▲附圖為 SonarQube 掃描結果，需將 Bug、Code Smells 修正為 0

三、.NET Core 6 WebAPI 學習應用：

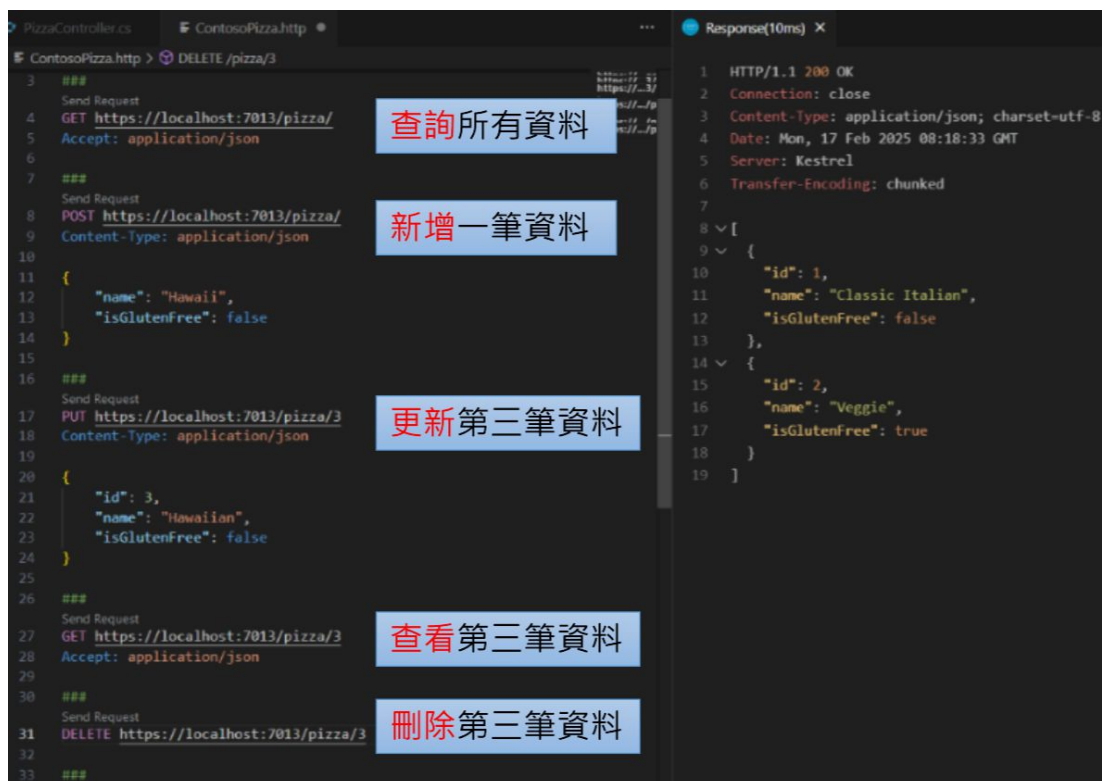
1. 理解 WebAPI 基本概念

RESTful API 強調使用標準的 HTTP 方法（verbs）來執行對資源的操作。這些方法不僅僅是傳輸資料，它們各自代表了特定的語義，使得 API 的操作更加清晰、可預測和符合標準。這有助於客戶端和伺服器之間更好地理解 and 協作。

- Get（查詢 / Select）：用於獲取資料。
- Post（新增 / Insert）：用於創建新的資料。
- Put / Patch（修改 / Update）：用於更新伺服器上已存在的資源。
 - Put（更新全部資源 / Replace Resource）：用於需要完全替換一個資料的內容時。
 - Patch（部分更新資源 / Modify Resource）：用於只需要修改資源的某幾個屬性，而不是整個資料時。两者的主要區別：Put 是「替換」資料，Patch 是「修改」資料。
- Delete（刪除 / Delete）：用於從伺服器端刪除指定的資料。



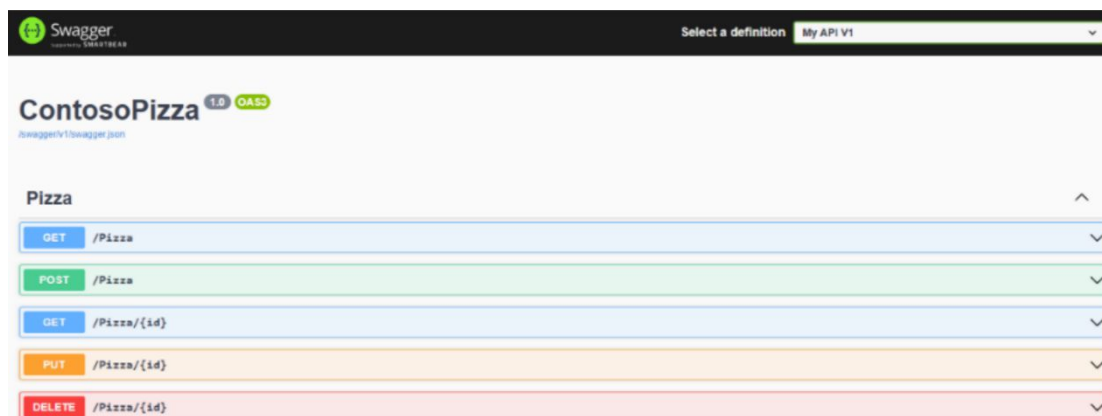
2. 使用 ASP.NET Core 控制器建立 Web API



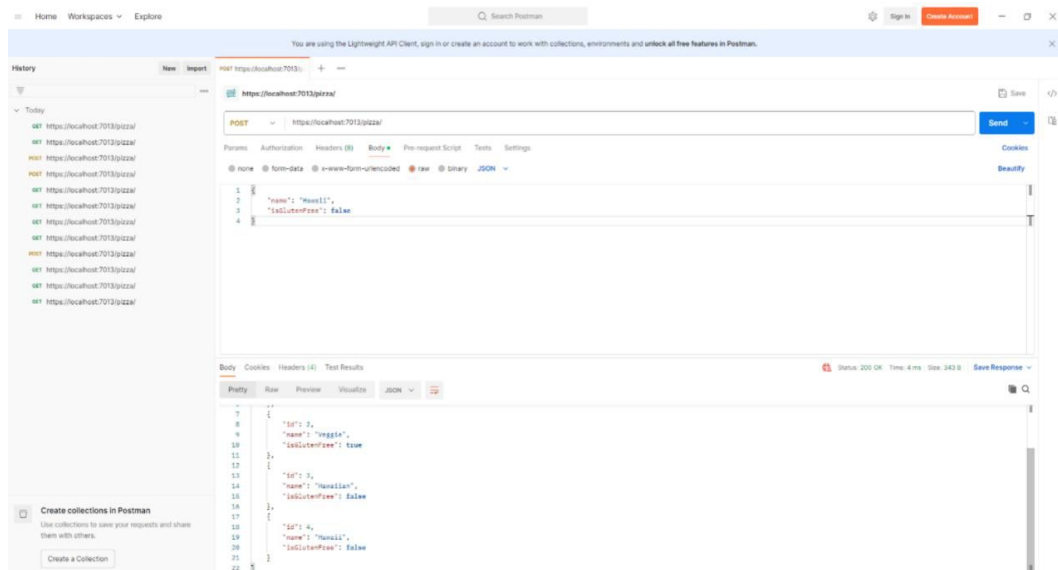
圖片左側是發送的 HTTP 請求 (Request)，右側是伺服器返回的 HTTP 回應 (Response)。這張圖清晰地展示了如何透過 RESTful API 的標準 HTTP 方法 (GET, POST, PUT, DELETE) 對後端資料進行完整的 CRUD 操作。每個請求都指向特定的資源 URI，並使用對應的 HTTP 方法來表達操作的意圖。

3. 使用 Swagger、Postman 測試 Web API 網址、功能

- Swagger UI 著重於 "API 文件的自動化生成和互動式展示"，讓使用者可以輕鬆地了解 and 嘗試 API。它通常是作為後端服務的一部分部署，提供給前端開發者和測試人員查閱。



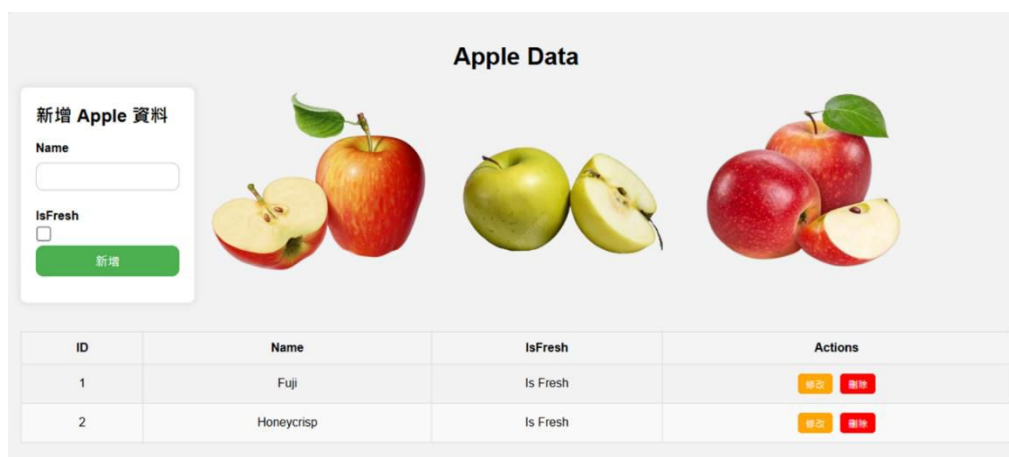
- Postman 則是一個更為通用的 "API 開發、測試和管理工具"。它不僅可以發送任何類型的 HTTP 請求，還提供了強大的測試、自動化和協作功能。開發者常常在開發和測試 API 時使用 Postman 來驗證 API 的行為，並在 Swagger UI 中查閱 API 文件。



4. 自製 Apple Web API 網頁

透過前端頁面去後端呼叫 API 執行 CRUD

- 在新增 Apple 資料可以直接輸入蘋果的資料，點及新增按鈕時，城市會去呼叫 API 執行 Post 新增資料，並且會同步新增到頁面底下欄位
- 點擊修改按鈕可直接在該欄位編輯資料，同時會去呼叫 API 執行 Put 修改資料
- 點擊刪除時，系統會呼叫 API 執行 Delete 刪除該筆資料



總結來說，這次學習涵蓋了 .NET Core Web API 的基本概念、RESTful 設計原則、開發實踐，以及如何利用 Swagger 和 Postman 進行有效的測試。最重要的是，透過「自製 Apple Web API 網頁」的範例，清晰地展示了前端介面如何透過按鈕事件驅動後端 API 呼叫，實現動態資料的 CRUD 操作，最終構建出一個完整的、互動式的 Web 應用程式。

三、 Jira 系統：

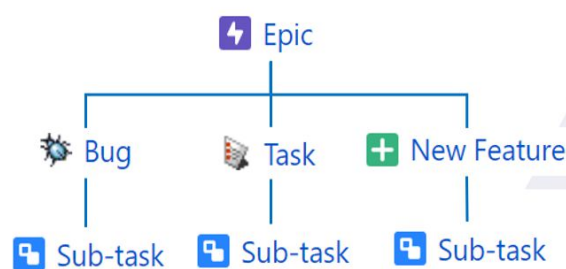
Jira 是一個強大的專案管理與問題追蹤工具，主要用於軟體開發團隊進行敏捷開發（如 Scrum 和 Kanban），幫助團隊規劃、追蹤、發布和支援產品，並能管理各種任務、錯誤、需求，透過可自定義的工作流程和看板，提升團隊協作效率與專案透明度。

1. 在 Jira 中 Issue Type 有以下幾種類型：

- Epic：通常代表一個較大的、長期性的目標或專案。一個 Epic 可以包含多個 Bug、Task 或 New Feature。
- Bug：用於追蹤軟體或其他產品中的缺陷或問題。通常與某個特定的功能或行為不符預期相關。
- Task：用於追蹤一般性的工作事項或活動。可以是開發、測試、設計等任何需要完成的具體任務。
- New Feature：用於追蹤要開發或實現的新功能或改進。通常與產品的增強或擴展相關。
- Sub-task：是 Bug、Task 或 New Feature 的細分，無法單獨存在。它用於將較大的問題或任務分解成更小、更易於管理的工作單元。

Issue Type

可以根據不同工作事項的規模、種類來選擇不同的 Issue Type
JIRA 的 3 層架構：



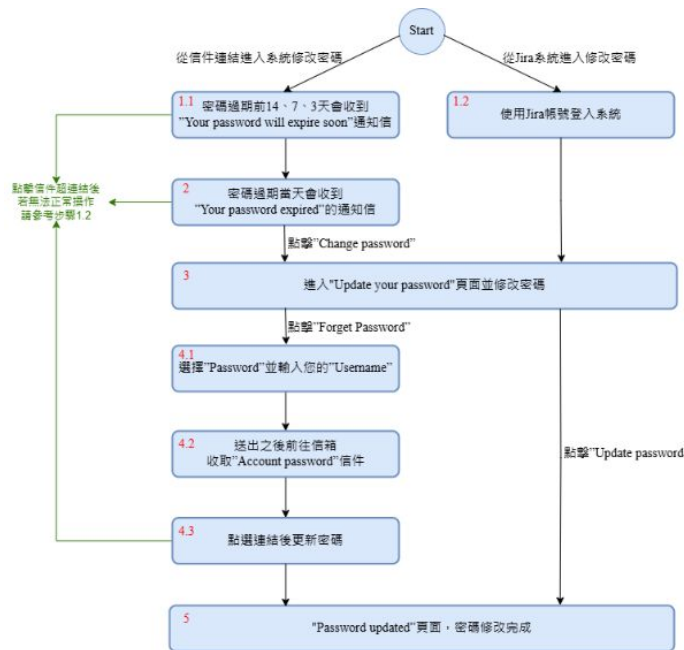
*Sub-task 只能在 ISSUE 下被建立，無法單獨存在

2. 看板操作與進度追蹤：

- 可以透過看板查看目前的專案進度，也能清楚看到每項任務的狀態。此外，還可以利用篩選功能，快速找到目前尚未完成的任務。
- 在每項任務（Issue）中，可以上傳相關檔案或是發表評論，讓所有相關人員能夠即時追蹤最新情況並表達意見。

3. 額外支援任務：

- 版本升級測試：參與 Jira 升級流程，執行功能驗證與回歸測試，確保套件功能正常運作。
- 帳戶與權限管理：協助建立新使用者帳戶，並根據角色分配至專案 Group，控制操作權限與資料可視性。
- 文件製作：撰寫外部用戶操作手冊（如密碼變更流程），採圖文並茂方式呈現，提高文件可讀性與操作便利性。



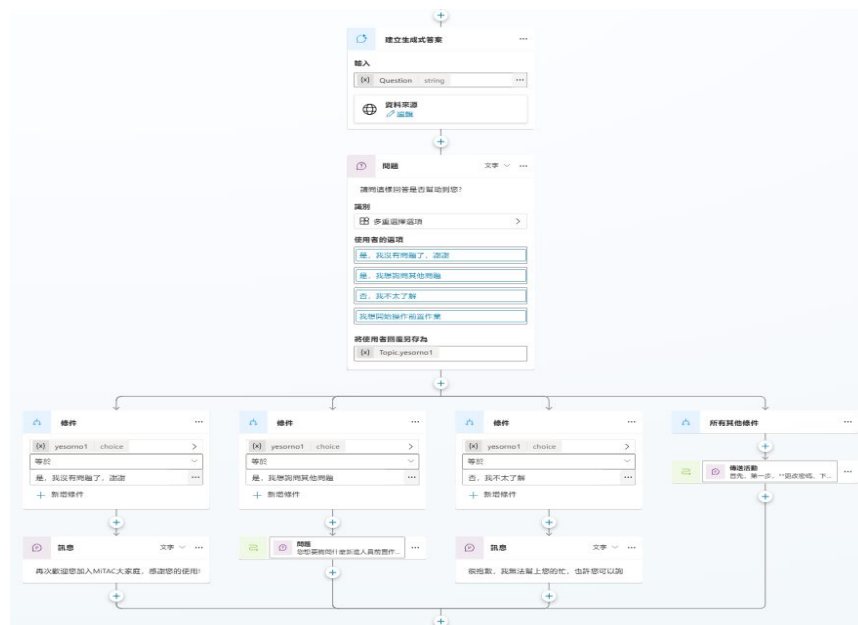
▲附圖為外部用戶操作手冊中的流程圖，用戶可以透過流程圖了解整個操作的流程，再根據操作手冊中的步驟去進行操作

四、Copilot Studio 聊天機器人建置：

透過 Microsoft Copilot Studio 開發企業內部的客服型聊天機器人，設計對話流程與回應邏輯。建立知識庫並導入公司內部 FAQ、文件與產品資訊，讓使用者能夠即時取得答案。



▲附圖為製作簡單智慧機器人所需要使用的知識庫



▲附圖為啟動智慧機器人設定流程，可以根據使用者選擇的條件去執行相對應流程



▲附圖為與機器人對話的介面，包含問題的回覆以及使用者回饋的選項

三、自我評估及心得感想

在神達電腦的實習期間，我獲得了超越課堂學習的實務經驗。從進行系統測試、撰寫技術文件，到實際參與多語系功能開發與介面優化，每一項任務都讓我更加理解軟體專案的完整運作流程。特別是在 PLM 系統更換與 E-Sop 系統開發的過程中，我深刻體會到測試人員不僅是找出問題，更是確保品質與使用者體驗的守門員。

這段期間，我學會如何撰寫結構化的測試案例、規劃測試流程，並透過問題追蹤與覆測，確保功能品質。這不只是技術上的成長，也培養了我面對問題時的細心與責任感。透過實際操作與跨部門合作，我學習到有效的溝通方式，以及在團隊中如何表達想法、提出建議並共同解決問題。

在主管與同事的指導下，我也體會到文件撰寫與知識分享的重要性。從建立使用手冊到撰寫測試報告，每一份文件都代表著專案品質的一部分。透過這些任務，我培養了對細節的敏感度與邏輯思考能力，也更能體會團隊協作與流程標準化對專案成功的關鍵性。

此外，實習期間我也首次接觸到 Web API、Jira、Copilot Studio 等企業級工具，拓展了對軟體開發與專案管理的認識。這些經驗不僅提升了我的技術廣度，也讓我在面對新技術時更有自信與學習方法。

整體而言，這次實習讓我不僅成長為一位能獨立執行任務的學習者，更加確立了未來在資訊領域深耕的方向。未來我希望能持續精進自動化測試與系統整合相關技術，並將所學到的專業態度與實務經驗，轉化為未來職涯持續前進的動力。