

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：Q

實習單位：智易科技

輔導老師：陸承志

姓 名：吳佳慧、楊昀庭、劉妍敏

學 號：1111762、1111846、1111850

目錄

1111846 楊昀庭、1111850 劉妍敏

壹、工作內容

一、工作環境介紹	3
二、工作詳述	4
三、實習期間完成之進度	15

貳、自我評估與實習心得

一、自我評估與實習心得--楊昀庭	16
二、自我評估與實習心得--劉妍敏	18

1111762 吳佳慧

壹、工作內容

一、工作環境介紹	20
二、工作詳述	21
三、實習期間完成進度	27

貳、自我評估與實習心得

自我評估與實習心得	28
-----------	----

1111846 楊昀庭、1111850 劉妍敏

壹、工作內容

一、工作環境介紹

智易科技股份有限公司成立於 2003 年，隸屬於仁寶集團，為臺灣網通產業具影響力的企業之一。新竹總部為主要研發的基地，鄰近交流道出口，與交通大學及清華大學比鄰而居。雖然地處新竹科學園區範圍內，但避開了上下班尖峰時段最為擁擠的園區出入口，使得員工通勤更加便捷。外部建築設計風格鮮明，具有高度辨識度，周邊生活機能完善，是一間在光復路上十分顯眼的科技企業。

身為國內首家整合寬頻、多媒體、無線與網際網路通訊協定的智慧型網路終端設備公司，智易科技專注於為全球電信業者、ISP 服務提供商及通訊設備公司提供整體解決方案與代工服務。產品以使用者為出發點，致力於開發操作直覺、人性化的使用介面，滿足終端用戶的實際需求。且具備高度的設計規劃能力與技術支援，協助客戶推出具備市場差異化的產品與服務，提升競爭優勢。此外，智易在全球多地設有銷售據點，拓展品牌的國際能見度，也能深入了解各地的市場需求。

我們所實習的部門位於六樓的 EIT 實驗室，隸屬於研發團隊，主要負責產品的軟體整合測試。實驗室為獨立空間，別於一般可容納數十人的開放式辦公區，有助提升專注力與團隊協作效率。由於測試工作涉及多項產品與設備，實驗室內的辦公桌也特別寬敞，方便同仁進行操作。公司內部亦設有多樣設施，如員工食堂、健身房與咖啡廳等，營造出良好的生活與工作平衡環境。整體裝潢以綠色為主，營造舒適輕鬆的氛圍，也有助於緩解長時間工作下的眼壓與疲勞。

部門文化方面，同事間互動融洽、氣氛和諧，我們有時也會一同享用輕鬆的下午茶時光。這樣的日常交流不僅讓工作氣氛更愉快，更提升了團隊凝聚

力。智易科技擁有專業的技術實力與國際化視野，並提供了良好的工作環境與企業文化，是個兼具學習成長與職涯發展潛力的理想工作場所。



圖 1-1 智易科技公司外觀

二、工作詳述

1. 手動連接至 GUI，並使用 Selenium 操作介面

實習初期，指導員個別提供我們一台 Gateway，透過網路線連接至電腦。我們嘗試手動登入裝置的 GUI（圖形化使用者介面），進行 SSID（網路名稱）與密碼的修改。首先在瀏覽器輸入預設 IP，進入 Gateway 管理頁面，接著依序操作找到無線網路設定欄位，修改 SSID 及密碼後儲存變更。確認設定成功後，我們斷線並重新搜尋剛修改的無線網路，進行連線測試以驗證設定是否生效。透過這樣的操作流程，可掌握 Gateway 裝置的網頁介面結構及各功能配置，熟悉無線參數設定的邏輯。

此外，也使用 Selenium 模擬登入與點擊操作，自動化 GUI 進入、欄位填寫、提交及資料擷取流程。該工具結合了 ChromeDriver 元件，針對 GUI 特定元素進行識別與操作。配合 CSS Selector 實作自動點擊與文字擷取，成功建立一套登入後自動讀取頁面資訊的流程。除了能提升操作效率，還有助於後續測試流程的自動化整合。Selenium 程式中也處理了等待元素出現的問題，避免

操作過快導致元素尚未載入而錯誤終止。整體過程涵蓋 GUI 介面理解、網路參數配置、自動化工具整合三部分。



圖 2-1 Gateway 設備

2. 主機伺服器設定與 IP 配置

為了加深網路相關的知識，指導員要求我們將一台主機設置為區域網路內的伺服器，需手動分配固定 IP 並驗證通訊狀態。首先將主機透過實體網路線接入交換器，再進入作業系統的網路設定頁面，指定靜態 IP、子網路遮罩、預設閘道與 DNS。設定完成後，使用 Windows 的 `ipconfig` 指令，確認 IP 設定正確套用，並以 `ping` 測試其他裝置連線能力。若能成功往返封包，代表該主機已正常加入區域網路，具備提供服務的基礎。我們接著在該主機啟動簡易

HTTP Server，供其他電腦透過瀏覽器存取指定頁面，並進一步驗證資料傳遞是否穩定。

3. 網路基礎實作與線材製作

對網路設定有基礎了解後，接著進入實作環節，我們實際參與了網路線的組裝與測試工作。使用壓線鉗、打線器等工具，依照 T568A 與 T568B 的標準進行 RJ45 網路線的壓接，並使用測線器確認線序與連線品質是否正確。這看似簡單的工作，其實非常考驗手眼協調與細心程度，線序只要有錯誤，就可能導致整條網路線無法使用。完成組裝後，我們將其用於實機連接主機與交換器，並透過靜態 IP 分配、ping 測試與瀏覽器連線，驗證網路是否暢通。

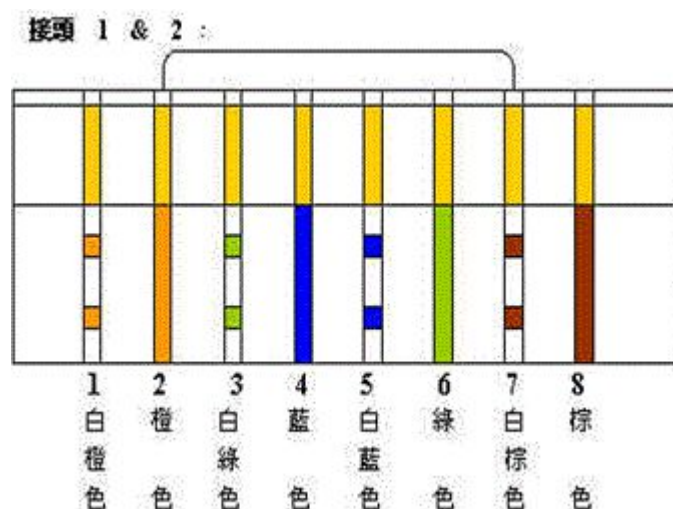


圖 2-2 來源: <https://finalfrank.pixnet.net/blog/post/28492531>
(網路線接頭接法圖解—Frank's 資訊科技潮流站 | 痞客邦)

4. 產品韌體升級作業

我們還須負責協助公司產品進行韌體更新，將設備的系統版本升級。首先把電腦連接到這台設備的 Wi-Fi，接著打開瀏覽器進入設備的操作介面 (GUI)。登入系統後，查看畫面上的 Hotspot Information，確認這台設備的相關資訊，例如目前的韌體版本和設備型號，確認無誤後就可以開始進行更新。

更新過程中，需上傳新的韌體檔案，系統會自動完成升級的步驟。完成後將設備重啟，使新的韌體正式生效。再登入 GUI 介面檢查韌體版本是否更新成功。

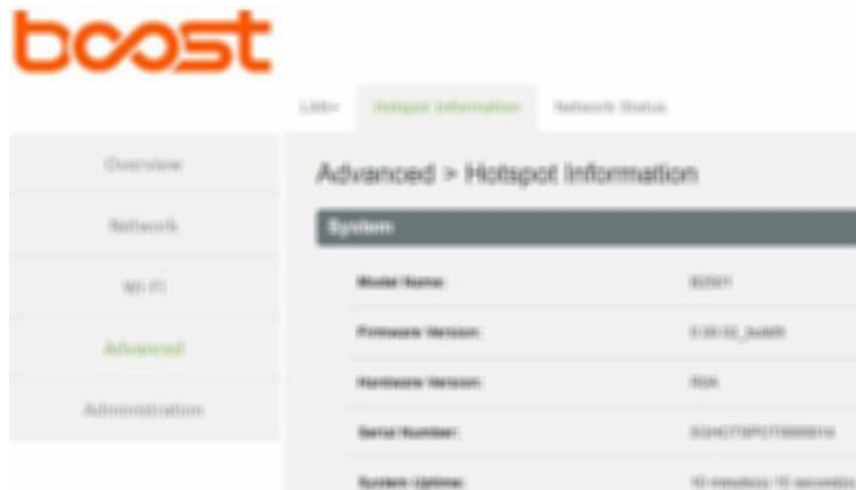


圖 2-3 GUI 介面

5. 驗證碼圖片文字辨識

以 Tesseract 為核心的驗證碼辨識系統建置，目標為辨識圖像中的驗證碼內容並驗證準確度。首先從網路上存取多張驗證碼圖片，以 JTessBoxEditor 蒐集並標註多張驗證碼圖像，訓練自定語言模型檔（如 name_724.traineddata）。資料整理後，執行字型訓練產生語言檔，將其放入 Tesseract 語言資料夾中供辨識使用。辨識流程使用 Python 撰寫，透過 OpenCV 對圖片進行前處理，包括灰階轉換與 threshold 二值化，以強化圖像對比與清晰度，再調用 Tesseract 進行文字辨識。

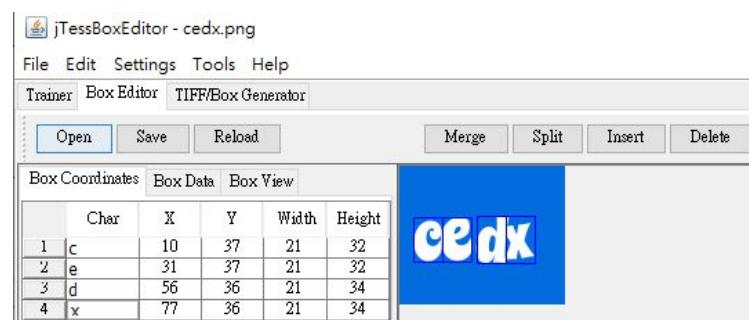
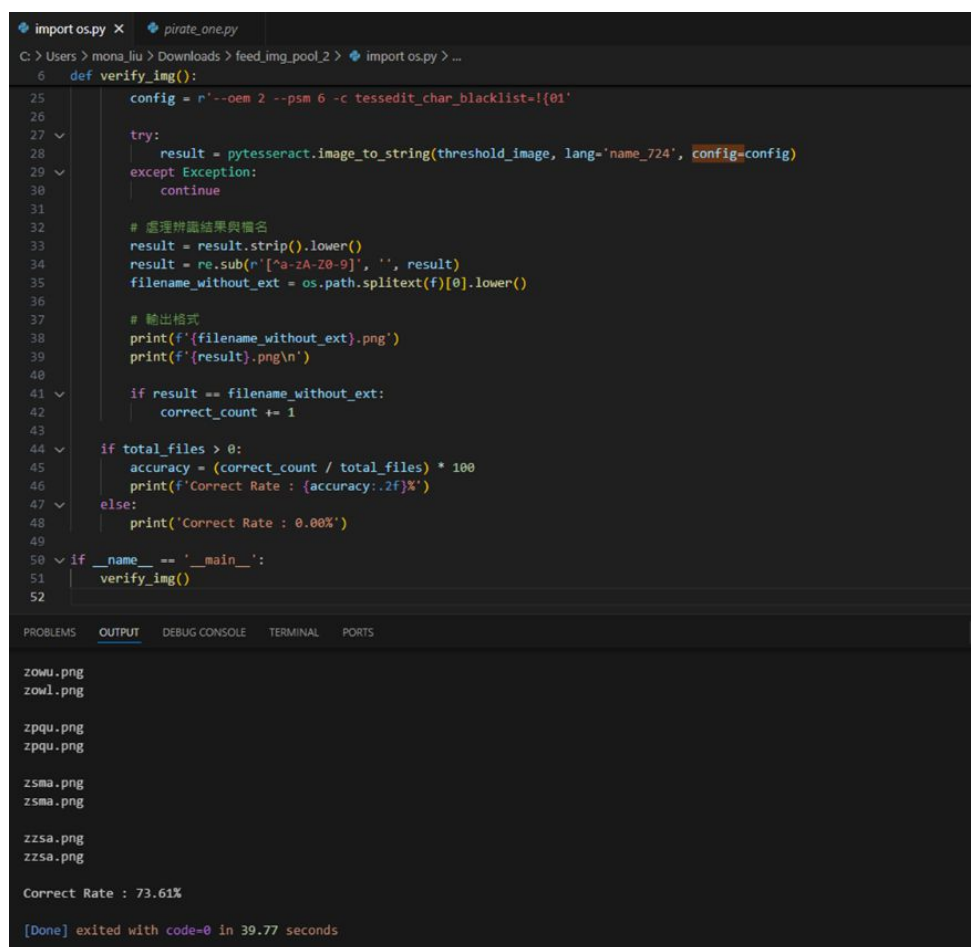


圖 2-4 JTessBoxEditor 訓練畫面

辨識完成後對結果進行調整，移除非英數字元與換行符號，標準化輸出結果格式。驗證部分，將每張圖片檔名視為正確答案，與 OCR 辨識結果比對是否一致，若相符則計入正確筆數，最後統計整體辨識正確率。為提升準確度，我們多次調整語言模型訓練字型組合，分析影響結果的圖像雜訊與字型變形因素，最終將成功率提升到約 74%。整體系統實作涵蓋圖像處理、模型訓練、辨識與比對四大環節，成功建置自定義 OCR 解決方案並完成準確度統計驗證。



```
import os.py x  pirate_one.py
C: > Users > mona_liu > Downloads > feed_img_pool_2 > import os.py > ...

6 def verify_img():
    config = r'--oem 2 --psm 6 -c tesseract_char_blacklist=!{01}'
    try:
        result = pytesseract.image_to_string(threshold_image, lang='name_724', config=config)
    except Exception:
        continue

    # 處理辨識結果與檔名
    result = result.strip().lower()
    result = re.sub(r'[^a-zA-Z0-9]', '', result)
    filename_without_ext = os.path.splitext(f)[0].lower()

    # 輸出格式
    print(f'{filename_without_ext}.png')
    print(f'{result}.png\n')

    if result == filename_without_ext:
        correct_count += 1

if total_files > 0:
    accuracy = (correct_count / total_files) * 100
    print(f'Correct Rate : {accuracy:.2f}%')
else:
    print('Correct Rate : 0.00%')

if __name__ == '__main__':
    verify_img()

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

zowu.png
zowl.png

zpuq.png
zpuq.png

zsma.png
zsma.png

zsza.png
zsza.png

Correct Rate : 73.61%

[Done] exited with code=0 in 39.77 seconds
```

圖 2-5 Python 圖像文字辨識驗證程式與執行結果

6. 主機板模組安裝檢測

此作業為無線性能測試，主要負責 WiFi 模組天線安裝與硬體組裝的步驟，提供各頻段天線連接至主機板時穩定的射頻環境。需具備基本的硬體裝配與 RF 模組知識，並理解 WiFi 頻段特性與測試環境要求。

開始作業前，需確認主機板已完成基本硬體組裝並固定於鋁製散熱基座上。接著依據設計圖與標籤，將對應標示之天線模組（2.4GHz、5GHz、6GHz）分別安裝至主機板上的射頻連接端口。安裝過程中需特別注意接觸狀態，保持連接穩定且不損壞 RF 端子，並於完成後進行基本的連通性確認。檢查天線與模組之間的連線阻抗是否正常，系統啟動後各頻段能正確被偵測與啟用，確保訊號傳輸的隔離與穩定性，具備無線功能測試的硬體環境。



圖 2-6 MTK WiFi 7 無線模組



圖 2-7 2.4/5GHz 與 6GHz 天線組

7. G5AR/G4AR WiFi 初始化效能測試

在系統開機流程中，WiFi 模組會經歷韌體載入、驅動初始化、網路介面啟動等階段。本測試的目的是透過 WiFi 從啟動開始到完成初始化（可掃描到 SSID）的實際時間，評估裝置開機時的網路準備效率。



圖 2-8 FWA 5G 通訊家用路由器

測試過程中，會透過系統 log 判斷 WiFi 初始化完成的時點。當 log 顯示出特定字時，代表 WiFi 初始化程序已完成，可開始進行 SSID 掃描與網路連線。記錄下此時間戳記與 WiFi 模組啟動的初始時間，並計算兩者之差(即為 WiFi ready time)，該區間反映 WiFi 功能的延遲情況，是系統啟動效能的重要指標。

此外，測試內容亦包含 Mesh 網路初始化時間的觀察。Mesh 功能啟動後，裝置會自動搜尋並與其他節點建立連線，形成網狀網路架構。記錄到網路拓撲建立完成的時間，藉此分析網路初始化的穩定性與速度。為確保資料穩定與一致性，需反覆進行並彙整各次測試的平均時間及變異範圍，作為後續效能調整與韌體版本比對的依據。

		G4AR (MR5)	G5AR (MR0)	G5AR (MR1)
System Boot until WiFi ready	WiFi ready & beacon broadcast	第一次1:30	第一次2:18	第一次2:23
		第二次1:36	第二次2:14	第二次2:23
		第三次1:30	第三次2:14	第三次2:21
		第四次1:30	第四次2:14	第四次2:23
		第五次1:29	第五次2:14	第五次2:22
	WiFi Mesh ready	第一次1:42	第一次2:44	第一次2:50
		第二次1:46	第二次2:42	第二次2:50
		第三次1:40	第三次2:42	第三次2:44
		第四次1:41	第四次2:41	第四次2:49
		第五次1:40	第五次2:48	第五次2:49

圖 2-9 測試結果

8. ODU 硬體識別資料與憑證導入

本工作旨在為 ODU 設備寫入 Board Data 和安全憑證，以確保設備具備正確的身份識別碼、網路配置參數及加密驗證能力。透過 ADB shell 執行指令，依據每台設備的獨立數值，寫入其對應序列號 (SN)，以及基頻 MAC 位址 (basemac)。設定設備的型號，並配置初始的使用者密碼。

在核心參數寫入後，進行 IMEI 碼的處理。先讀取模組的 IMEI 資訊，再透過指令將其寫入板級資料中。完成後，必須執行一輪 arc-board get 指令

獲取資料，確認參數值均已正確寫入。接著部署安全憑證，將與設備 SN 相對應的格式憑證檔案推送到設備的目錄下，將此檔案寫入設備的永久儲存區，並以 certutil verify 檢查憑證的有效性與完整性。最後執行 sync 指令確保所有資料操作均已寫回磁碟，完成板級資料與安全配置程序。

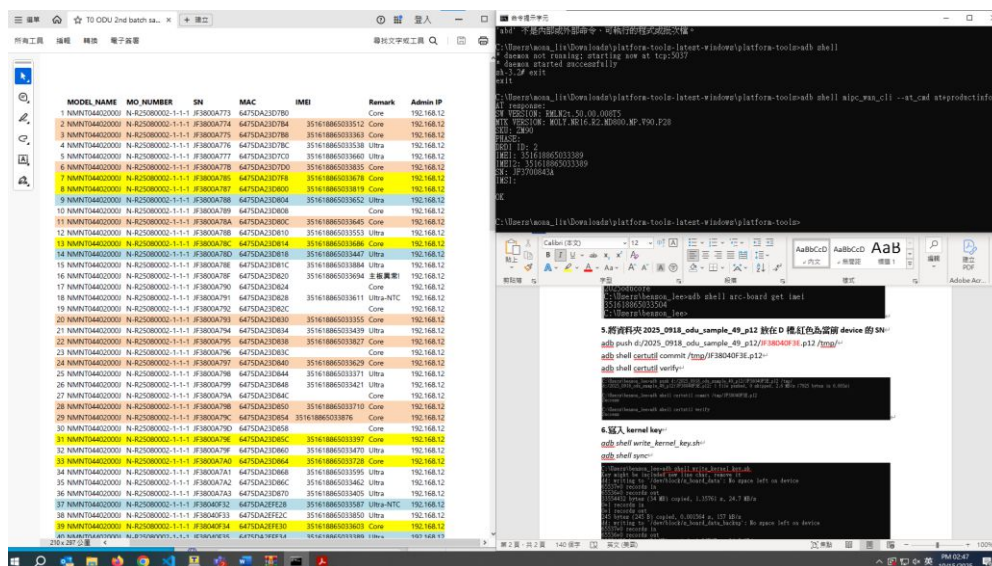


圖 2-10 設備參數資料表與終端機操作畫面

9. 使用 Flash Tool 重新燒錄 ODU 韌體及驗證

操作流程始於準備一條 USB 線連接設備到 PC，接著開啟 Flash Tool 軟體。在工具介面中，載入指定的燒錄設定檔 flash.xml。確認載入無誤後，燒錄模式需選取 Firmware Upgrade，這是保證系統所有分區完整更新的關鍵步驟。配置完成後，點擊介面上的 Download 按鈕 啟動燒錄程序，並立即對板子執行斷電再上電的操作，使設備進入下載模式。燒錄成功後，工具介面會顯示 Download OK 的提示訊息，標誌著韌體更新作業的完成。

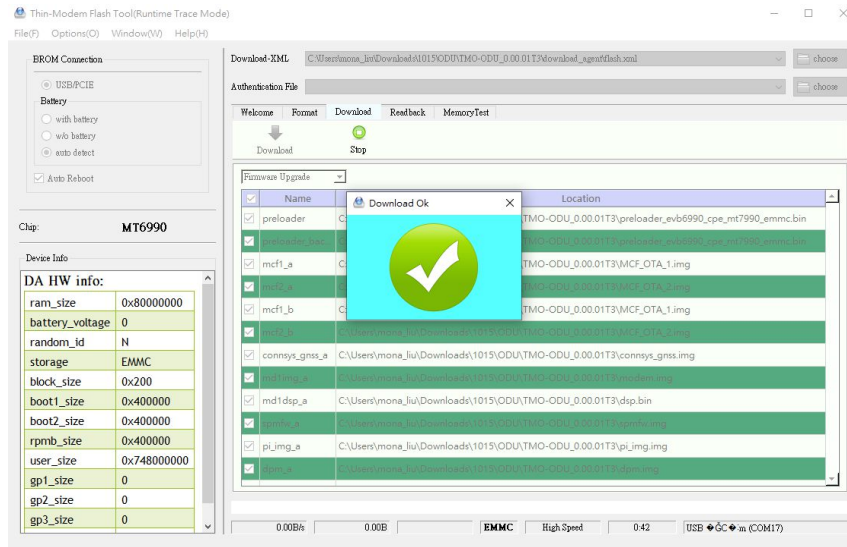


圖 2-11 燒錄成功畫面

韌體燒錄完成後，需先等候設備重新啟動完成，再進行參數驗證以確保燒錄成功。驗證流程是透過 ADB shell 連線至設備，執行一系列指令來確認設備的關鍵資訊：依序檢查設備的序號 (SN)、基頻 MAC 位址、型號，以及 IMEI 碼，核對其與出廠資訊的一致性。此外，亦需檢查設備的預設使用者密碼是否正確寫入，並透過讀取檔案確認當前運行的韌體版本號與燒錄目標相符。最後，執行 certutil verify 檢查設備的數位憑證狀態與完整性。

```

命令提示字元 - adb shell

C:\Users\mona_liu\Downloads\platform-tools-latest-windows\platform-tools>adb shell
sh-3.2# arc-board get sn;
get basemac;
arc-board get moJF3800A79Esh-3.2# arc-board get basemac;
del;
arc-board get user-pwd;
arc-board get imei;
cat /versio64:75:D8:5Csh-3.2# arc-board get model;
n.txt;
certutil verifyTMO-G5Coresh-3.2# arc-board get user-pwd;
20250ducoresh-3.2# arc-board get imei;
351618865033397sh-3.2# cat /version.txt;
imagename:rdk-generic-broadband-image-mt6990-odu-20251003050520
BRANCH=rdk-2024q1-dunfell
TOCTO_VERSION=dunfell
VERSION=0.00.01T3
SPIN=0
BUILD_TIME="2025-10-03 05:05:20"
Generated on Fri Oct 03 05:05:20 UTC 2025
sh-3.2# certutil verify

```

圖 2-12 ADB Shell 執行設備參數讀取與韌體版本驗證畫面

10. ZAP 網頁安全性掃描與弱點檢測

使用 ZAP (Zed Attack Proxy) 工具進行管理、設定介面與內部 API 等資料交換路徑的安全性測試，導入環境流量模擬各類資料與瀏覽器行為，逐項攔截修改分析伺服器回應，檢測系統後台介面潛在弱點與安全風險。

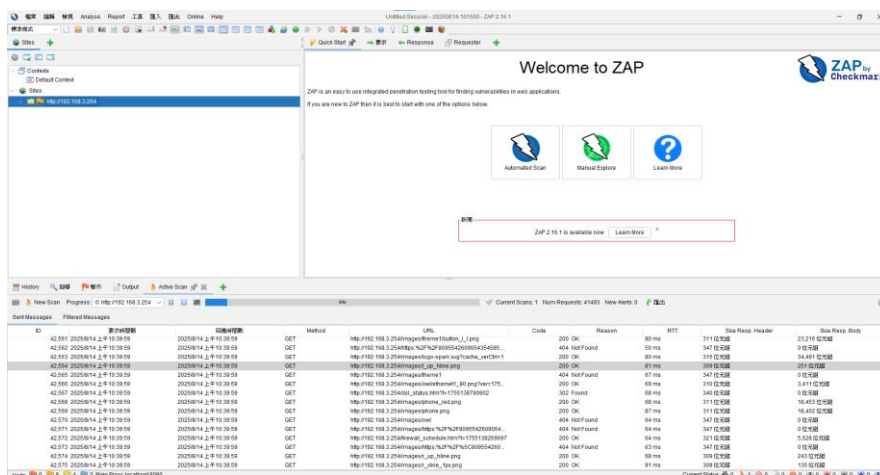


圖 2-13 Zed Attack Proxy

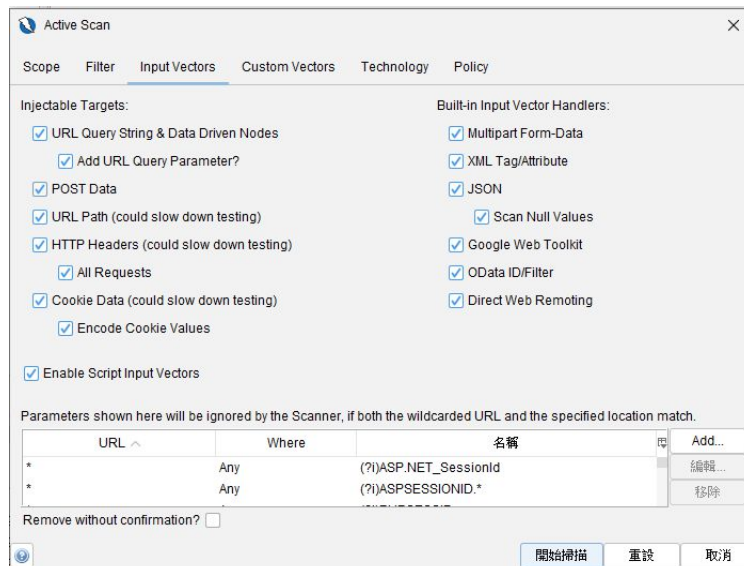


圖 2-14 安全性測試項目

掃描過程中，ZAP 會自動檢測常見漏洞類型，如敏感資訊外洩、Cookie 設定不安全等。針對掃描報告中顯示的可疑項目，逐一比對並驗證，排除誤報及記錄實際存在的風險。透過觀察伺服器回應來確認漏洞的利用性與影響範圍，

並依嚴重程度分類，提出修補建議（如輸入驗證與輸出編碼、參數化查詢、強化認證與 session 管理、加密敏感資料、設定安全 cookie 屬性等）。最終將測試結果彙整成技術細節、修補建議的掃描報告，提供維運團隊開發與維護參考。

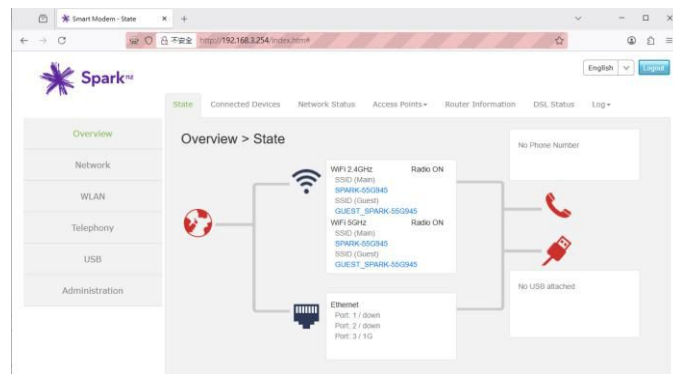


圖 2-15 GUI 介面

11. RG/CM 設備韌體燒錄、預設值配置

在進行設備韌體燒錄前，需先準備環境(USB 隨身碟、電纜線、console 線)並接上電源。在 RG (Router Gateway) console 輸入指令掛載 USB 隨身碟，將韌體檔案燒錄進設備。韌體燒錄後，需進行預設值配置，在 CM (Cable Modem) console 輸入指令，清除所有動態配置參數。完成配置後，設定 Upstream Amp Type 並寫入配置，重啟設備以套用新設定。



圖 2-16 Cable modem 家用路由器

待設備重啟後，在 CM console 執行檢查步驟，驗證韌體版本與硬體參數是否正確寫入。最後跳轉至日誌頁面，確認設備顯示的 log 訊息，代表流程完成，CM 模組已成功啟動並進入可操作狀態。

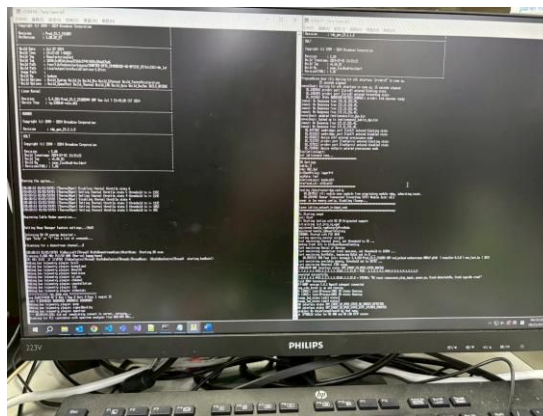


圖 2-17 Tera Term 操作介面

三、實習期間完成之進度

月份	項目
2 月	1. 建立網路基礎概念 2. 手動連接至 GUI
3 月	1. 訓練驗證碼模型 2. 主機伺服器設定與 IP 配置 3. 產品韌體升級
4 月	1. 撰寫自動化腳本(Selenium) 2. 基礎網路佈線與實作
5 月	1. 撰寫自動化腳本(Selenium) 2. 撰寫報告
6 月	1. 撰寫報告
7 月	1. 主機板模組安裝檢測
8 月	1. ZAP 網頁安全性掃描與弱點檢測
9 月	1. RG/CM 設備韌體燒錄、預設值配置
10 月	1. G5AR/G4AR WiFi 初始化效能測試 2. ODU 硬體識別資料與憑證導入 3. 燒錄 ODU 韌體 (FW) 及驗證
11 月	1. 撰寫實習報告

貳、自我評估與實習心得

一、自我評估與實習心得——楊昀庭

學習與成長

1. 網路通訊基礎與自動化實務

在實習初期，我掌握了基礎網路設備的部署與配置技能，包括網路線的標準化組裝、天線安裝，以及 IP 位址的精確設定。更進一步，我學會了透過 GUI 介面對設備進行操作與管理。為了提升效率，使用了 Selenium 自動化測試框架，成功實現設備登入與介面操作流程的自動化，有效優化重複性工作流程，並為後續的系統測試奠定了穩固的實務基礎。

2. 設備維護與韌體管理

我參與了網路設備的關鍵維護工作，尤其專注於韌體升級管理。這項經歷培養了我極高的細心度與嚴謹的標準作業流程（SOP）執行能力。在執行更新前，必須仔細核對設備型號及編號，以確保韌體版本準確無誤。透過嚴格遵循流程，我能夠保障設備運行的穩定性與安全性，展現了對維護作業的責任感。

3. 進階技能應用與專案成果

在指導者的引導下，我具備了應用進階技能解決實務問題的能力：

- Selenium：我透過對 Selenium 的深入掌握，開發自動化腳本來處理網頁操作，大幅提高了資料擷取與重複性任務的效率，展現了利用程式設計提升生產力的能力。
- 影像辨識與優化：在驗證碼辨識專案中，我應用 Tesseract-OCR 模型的訓練與優化技術（利用 JTessBoxEditor），成功地將經過

圖像處理後的文字辨識正確率提升至 74%，增加了我在影像處理和模型調校方面的實戰能力。

- 自然語言處理（NLP）應用：我們在實習期間完成了一個基於 LSTM 的情感分析系統專案。透過大規模資料（超過 30,000 筆）的處理與模型建構（使用 Keras、NLTK 進行前處理），最終模型的測試集準確率高達 93.4%。此專案提升了我從數據蒐集、模型開發到高效能系統建構的能力。

實習心得

在智易科技實習的這段時間，我所獲得的遠不止於技術能力的提升，更培養了未來投入職場時不可或缺的職場軟實力。

我非常感謝團隊中熱心且專業的同事們，他們在指導我進行工作內容時，展現了極高的耐心和細緻度。無論是基礎網路配置還是進階的程式實作，他們總是願意花費額外的時間，一步一步地講解，這不僅讓我快速掌握了複雜的工作流程和技術細節，更重要的是，我從他們的言行中學習到了寶貴的職場溝通與協作的藝術。這種專業指導與積極的團隊氛圍，讓我深刻體會到跨部門溝通的順暢性對於專案成功的關鍵作用。我學會了如何主動提問以獲得最大效率的解答，以及如何以積極、正向的態度面對工作挑戰。

這段經歷讓我對未來畢業後融入職場環境、建立有效的人際網絡有了更清晰的認知和更充分的準備，我確信在這段實習中所學到的能力，將會是我未來職業生涯中一筆寶貴的財富。

二、自我評估與實習心得——劉妍敏

近一年的實習經驗，讓我深刻體會到工作與學習之間的差距。透過不同專案任務，我學會在工作環境中運用理論、面對問題並解決挑戰，熟悉技術操作的細節。此外，也培養了我在分析、思考與溝通上的能力，更清楚地了解個人定位與未來方向。

剛踏入職場時，其實還隱約有些不安，認為自己對公司的專業領域見識淺薄，無法做好負責的項目或適應環境轉變。但在前輩及學長姊的鼓勵下，逐漸建立起我對專業知識的熱忱與好奇。前期幾個月，公司讓實習生先學習基本觀念，從較單純的工作開始實作練習，因此安排我協助 WiFi 模組的基本測試。啟動過程中涉及多個階段，如韌體載入、模組啟動、介面初始化等，透過逐次比對時長記錄，判讀日誌的設定與環境，辨識關鍵的延遲原因，我對網路模組架構有了更深入的理解。隨著經驗的累積，工作範圍也逐漸擴大，開始負責硬體層面的組裝與測試準備。在天線模組安裝任務中，需要理解射頻的特性，並使用量測工具檢查阻抗與連通性，待組裝完成後初步驗證各頻段都能被系統正確偵測與啟用。無論何種設備、步驟的調整皆與測試有著密切關聯，讓我在後續分析分配到相關工作時，能以更全面的角度看待各項要素。

進入實習後半段，我接觸更偏向系統與安全性的測試任務。其中印象最深的是使用 OWASP ZAP 進行網頁與 API 的安全性檢測，這項工作應具備一定的網路協定知識與攻防思維，才能正確解讀工具回傳的結果。雖然流程相當耗時，但我學會如何從攻擊者角度審視系統架構，以及資訊安全在產品開發中扮演的重要角色。關注資安與測試工程的交集，使我對未來職涯也產生新的想法。除了技術層面的累積外，這一年最大的成長在於工作態度與思維模式的轉變。起初面對複雜的系統與陌生的流程時，常會感到手足無措，但部門的前輩都很願意指導與解答疑問，因此我開始會主動參與討論、協助尋找問題根源，更有信心去嘗試新的挑戰。同時，團隊合作也是我學到的重要課題，與不同部門的同事合作時，能以更有條理的方式表達邏輯，並吸收學習他人的觀點，藉由溝通協調，勇於跨出自己的舒適圈。

實習不僅是專業技能的提升，更是一種持續學習與成長的態度。在態度與思維上，我試著調整心態，將過程中遇到的瓶頸視為養分，於是理解結果並非唯一的關鍵，面對困難時不斷嘗試與思考，無形中也在引領我們進步。回顧這段歷程，無論在專業知識、問題解決或職場心態上，都獲得了寶貴的成長。未來希望能延續這份經驗的收穫，在往後學習與工作中持續精進、突破，挑戰更具深度的任務。

1111762 吳佳慧

壹、工作內容

一、工作環境介紹



圖 1-1 智易科技

智易科技股份有限公司的台灣總公司位於新竹科學園區，靠近新竹交流道出口，並鄰近清華大學。成立於 2003 年，智易科技專注於寬頻、多媒體和無線網路通訊技術的產品研發，是國內首家整合寬頻、多媒體、無線網路及網際網路通訊協定的專業、智慧型網路終端設備公司。

智易科技的產品行銷全球，並在多個國家設有研發及銷售據點，致力於提供創新科技解決方案，推動全球數位化發展。公司秉持以使用者為核心的研發理念，專注於滿足客戶需求，並設計人性化操作介面，以提升使用者體驗，確保客戶滿意。

我所屬部門是多媒體研發三部，EIT 測試課，部門辦公室位於公司的五樓，在這辦公室中，大部分同事是負責研發，而測試課人數只占少數，我們測

試課主要負責機上盒的軟體測試，包括例行性測試、Google/Netflix/Prime 認證測試等測試，而我們實習生的工作主要負責手動測試的部分。公司有為每位實習生提供一台筆電，在我的座位上還有一台桌機屏幕，我會把這台桌機屏幕作延伸投影，把要記錄測試結果的線上檔案與用作參考或是筆記的檔案，分別投影在不同屏幕上以避免改錯檔案。當進行測試時，通常會另外搬一台電視，以模擬客戶實際使用機上盒的操作情境。



圖 1-2 工作環境

二、 工作詳述

1. 測試產品：機上盒 Set-top box (STB)



圖 2-1 專案產品

目前主要負責測試的專案依上圖(圖 2-1)順序，分別為中華電信的 HamiVideo、Swisscom 的 IP1800、IP2000 及 IP2300。

2. 測試環境架設

機上盒測試環境架設

在進行測試前，我們會先把機上盒恢復至出廠預設狀態(Out-of-the-box，OOB)，以確保出廠設置正確並排除可能的干擾。然後透過 ADB connect 連接機上盒與測試設備（圖 2-2），為後續的測試指令和操作做準備。

```
C:\Users\tobby_ng>adb connect 192.168.1.19
failed to authenticate to 192.168.1.19:5555

C:\Users\tobby_ng>adb connect 192.168.1.19
already connected to 192.168.1.19:5555

C:\Users\tobby_ng>adb devices
List of devices attached
192.168.1.19:5555    device
```

圖 2-2 連接裝置

在恢復原廠預設狀態後的第一次連接必定為 failed to authenticate，畫面會跳出是否同意裝置的連接的提示，點擊 allow 後才能成功連接。除了透過 Ethernet/WIFI 連接，還可以用 USB 傳輸線連線。

Ubuntu 測試環境架設

在入職後，我們會先架設 Ubuntu 環境。Ubuntu 是業界很常用的 Linux 作業系統，許多測試流程、平台建置與軟體開發都以它為基礎。完成 Ubuntu 虛擬環境架設後，會先安裝 adb、zip、SSH、JAVA、node、samba、TFTP 等必要套件。

後續的測試作業，如 YTS 手動測試、xTS 自動化測試、產測與 OTA 更新，都會在 Ubuntu 上進行，因為這些測試套件需要穩定且一致的 Linux 執行環境才能正常運作。為確保測試結果的正確性與一致性，每次測試前都需要將 YTS、xTS、eMMC 等測試套件或映像版本更新到最新版本。此外，在韌體與網路測試中，也會使用到 TFTP 來傳輸映像檔或進行系統燒錄，因此 Ubuntu 是整個測試流程中的關鍵基礎平台。

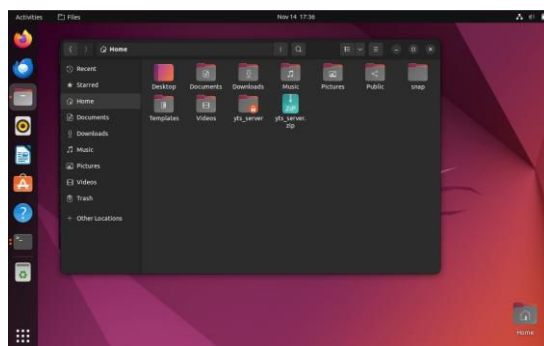


圖 2-3 Ubuntu

3. 例行性測試工作

在剛開始實習時，我最先學到的手動測試項目包括 CTS Verifier、Smoke Test 和 YouTube Test 這三個，在實習中後期還有一份 Clean SDK Test Plan 的手動測試項目，這些測試是我們在實習期間會經常進行的工作。隨著產品軟體的更新，我們會在每次軟體新版本發布前，負責對這些專案進行認證測試，並把測試結果記錄到線上 excel 表格，此表格由 Google 提供，並會定期更新測試項目，所以每一次進行認證測試之前都必須要先仔細檢查測試項目，確保新發布的軟體可以符合 Google 認證標準及穩定性，並及早發現可能的問題，避免用戶在使用過程中遇到錯誤或故障。

CTS Verifier



Android 兼容性測試套件 (Compatibility Test Suite, CTS) 是一套商業級的測試工具，用於驗證裝置是否符合 Android 系統的相容性標準，測試項目包括 Audio、Camera、Bluetooth、Wi-fi、TV 等…。在進行 CTS Verifier 測試前，需根據被測機上盒所搭載的 Android TV 作業系統版本，下載對應版本的 CTS 套件 (Android 12 或 14) 至本機，然後透過命令提示字元 (cmd) 下指令，將測試套件中的 APK 安裝到 STB (圖 2-4)，並進行後續的相容性測試。

CTS Verifier 需要手動操作界面，並根據各項測試項目的描述架設測試環境與操作結果進行判斷，以評估該項測試是 “PASS” 還是 “FAIL”。

```
C:\Users\tobby_ng>adb -s 192.168.1.40:5555 install -r -g "C:\adb\android-cts-verifier\CtsVerifier.apk"  
Performing Streamed Install  
Success
```

圖 2-4 在 cmd 下指令安裝 CTS Verifier 套件到機上盒

Smoke Test

這是一種基本的軟體功能測試方法，主要用來快速檢查應用程式或系統在更新或安裝新版本後，核心功能是否能正常運作。這類測試涵蓋系統中最基本且關鍵的功能，測試範圍包括：OoB 設置、Launcher 與使用者介面功能、音效輸出、Google 投影功能、待機與喚醒模式、裝置連線能力，以及系統的基本穩定性等。

YouTube SDR Playback Test (YTS)

YTS 是一款用於機上盒中 YouTube 應用程式的測試工具，主要驗證 YouTube 影片播放功能系統是否符合 GTVS 和 CDD 中對視訊解碼的要求。測試透過指令播放影片，期間依說明觀察影片播放狀況，並在畫面提示後選擇“PASS”或“FAIL”。部分測試會在影片播放結束，或完成指定測試指示後，自動判定並直接顯示結果，無需人工操作，但是如果出現 FAIL 的結果就必須反覆測試分析是否是軟體的問題或是測試環境的問題。

```
tobby@tobby-VirtualBox: $ yts
18:02:38.714 YTS: CLI: ...
18:02:40.104 ...
18:02:40.104 ...
18:02:49.905 ...
18:02:49.907 ...
06:02:50.885 ...
06:02:50.886 ...
06:02:50.904 ✓ PASSED
06:02:50.909 ...
06:02:51.689 Executed 1 of 1 test SUCCESS.
```

圖 2-5 yts test

Clean SDK Test Plan

相較另外三個例行性測試不同的是 Clean SDK Test Plan 只有在 SDK 進版的時候才需要做測試，SDK release 通常是一季或是半年一次。此 Test Plan 測試項目包含常見的測試項目與以往測試中經常遇到的問題，以確認 SDK 版本更新，功能是否能正常運作，並針對不同規格與設備進行更深入的驗證。例如，Bluetooth 測試會針對各型藍牙裝置進行開關機及壓力測試；Power Mode 測試則涵蓋三種不同的 Standby Mode。由於測試範圍更為全面且細緻，完成整份 Test Plan 所需的時間也相對較長。

Android Pure SDK Test Plan_v1.1 - 100% Calibri

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

12

1

圖 2-6 Clean SDK test plan

4. Netflix 手動測試 (ADHOC) & 認證測試

Netflix 認證測試是一種互動式測試，測試內容包含 Auto, Manual, EP 及 HA 等多項測試，因裝置支援多種規格，因此會針對每個規格進行多次類似的測試。目前，我只有協助測試 Manual Test 中較簡單且重複的項目，如觀察畫面大小與畫質等。而當同事完成所有認證測試後，可能會請我們協助將測試結果記錄到線上 excel 表單上，透過這些記錄可以很清楚地了解每個版本的測試結果。

在進行 Netflix 認證測試時，會同時進行 Ad-Hoc 手動測試，範圍包括：OOB、Playback、Audio、LaunchExit、Security、Performance、MDX Casting、Network 和 Miscellaneous。手動測試主要目的是確保使用者透過數位機上盒在觀看及操作 Netflix 應用程式時有絕佳的使用者體驗，避免出現非預期問題。

5. 產測與 OTA 更新

在進行一系列的測試前，會先完成產測和 OTA 更新，這部分通常由同事們處理，再交給我們做測試，但偶爾需要我們協助燒錄多台機上盒。

產測：裝置接上 u-boot 和 console(圖 2-7)，使用 Tera Term 連接裝置並下指令進行燒錄，主要通過兩種方式燒錄資料到機上盒的嵌入式多媒體卡(eMMC)。

- usb2emmc：將資料存到 USB 隨身碟，再透過 USB 燒錄至 eMMC。
- tftp2emmc：先在 Ubuntu 架設 TFTP server，再將資料儲存到 Ubuntu 的 TFTP 資料夾，經由 TFTP 協定從遠端伺服器下載資料並進行燒錄。

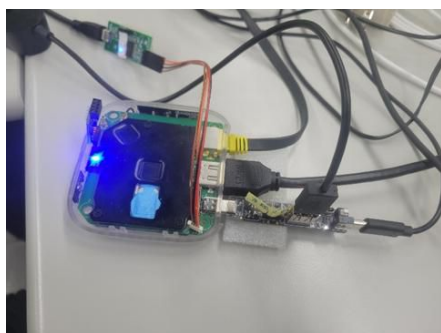


圖 2-7 裝置接 u-boot 和 console

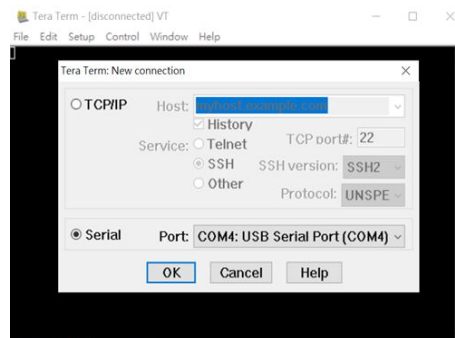


圖 2-8 Tera Term 連接 UI

7. AI Labling 標註校正

這項工作是針對系統已自動標註的圖片進行人工審查與修正。由於 AI 影像辨識判斷的準確度需要透過非常大量的資料進行訓練，透過人工校正可協助模型學習、影像分析得更準確。我的主要任務是將系統預設標為 person 的小孩改為 child。這樣能提高資料的準確性，幫助模型更準確地學習人類分類。而這是在沒有其他工作時進行，工作內容相對單調但輕鬆，且沒有急迫完成時限。

8. 金鑰配置 (Key injection)

這項工作是協助將機上盒設備從開發或未授權狀態，轉換為可合法播放受保護內容的狀態。由於新取得的設備尚未通過 DRM 授權，因此無法播放受保護內容（如 Netflix、YouTube）。為完成授權，需要依照 SOP 操作，將密鑰安全地嵌入設備，並記錄裝置相關資訊，接著下指令取得 Widevine Key 並上傳至 Google，以完成設備註冊並取得播放授權。

三、實習期間完成進度

月份	項目
2 月	學習多媒體串流與 Android 生態系統概念 學習例行性測試流程 學習產測及 OTA 更新 測試環境架設(在 Window 作業上架設 Linux 測試環境)
3 月	正式例行性測試工作 學習 Netflix 手動測試
4 月	問題複製-開關機測試 例行性測試工作 完成 AI Label
5 月	學習 Netflix 認證測試中部分 Manual test 測試 例行性測試工作

6 月	例行性測試工作 Key injection
7 月	例行性測試工作 Clean SDK test plan 測試工作
8 月	例行性測試工作 問題複製-Wifi 問題複製-Bluetooth
9 月	Clean SDK test plan 測試工作 問題複製-Wifi 問題複製-Bluetooth
10 月	例行性測試工作 Key injection
11 月	例行性測試工作 問題複製-Bluetooth

貳、自我評估與實習心得

實習過程中學到的知識

1. 多媒體串流與 Android 生態系統概念

在剛進入公司的第一週，主管向我介紹了實習期間的工作內容，以及多媒體與 Android 系統的基礎概念。

多媒體串流指的是透過網路即時傳送影音資料，讓使用者可以邊看邊接收，不必等待全部資料下載完成。而我們的工作便是測試串流技術的即時性與穩定性，確保使用者在使用機上盒播放影音過程中，不會有停頓或延遲，提升觀看品質。

而 Google Android 認證測試主要包括 CTS、GTS、VTS、STS 等，由於縮寫都為 *TS，故統稱為 XTS 測試，這些測試涵蓋不同層面的相容性與功能驗證，確保裝置符合 Android TV 的標準與性能要求。

2. 鍛煉英文能力與記錄習慣

大部分測試的說明與操作指引都是以英文描述，尤其是公司的機上盒專案主要面向海外市場，因此在測試時多以英文介面進行測試。雖然我偶爾會使用翻譯工具來輔助理解，但翻譯常無法完全準確傳達技術用語與操作流程，加上部分說明看似簡單，卻未必完全理解操作步驟，容易造成誤解。

為了更順利完成測試，我養成了記錄筆記的習慣。現在我會先嘗試閱讀原文指引，遇到不懂的再查閱筆記或使用翻譯工具。每當學習到新的流程或遇到不熟悉的操作或問題，我會記下步驟、操作以及解決方法，這不僅能提升我的工作效率，也能避免未來重複犯下同樣的小錯誤，減少對同事的依賴。

3. 時間分配

每個測試的項目其實不算多，有些項目的內容相似，甚至在執行某一測試時，就能順便涵蓋到其他項目，因此較快的測試在半天內就能完成。但這不包括需要長時間待機、整晚播放的測試，這類測試相對耗時。

為了避免時間浪費，我會優先完成可以同時進行，且不需要長時間等待的測試項目。而像是需要待機兩小時、或整晚播放直播的測試，則會安排在午飯時間或下班前才執行。為了避免跳過的測試被遺漏，我會將尚未完成的 test case 記下來，方便後續補測。透過這樣的時間規劃，不僅提升整體工作效率，也幫助我更好地掌握測試進度，盡可能在當天完成所負責的項目。

4. 認識技術術語與工具操作

在實習期間，我接觸到許多與機上盒測試相關的技術術語與工具，逐漸建立起對專業名詞與實務操作的理解。例如 STB(Set-Top Box)、OTA (Over-The-Air)更新、ADB (Android Debug Bridge)、OOB (Out-of-the-box)、eMMC、HDMI CEC 等，這些都是測試說明文件中常見且必須理解的縮寫。由於大多數測試說明僅以簡寫呈現，而同事在交付任務時也常直接使用這些縮寫，因此若不了解其含義，會影響對測試目的與操作步驟的掌握。

除了術語外，我也實際操作了多種測試工具，包括使用 ADB 連接裝置、Tera Term 進行 u-boot 指令操作、使用 cmd 安裝 APK、使用 Ubuntu 虛擬機的 TFTP 傳輸文件，以及利用 logcat 抓取錯誤日誌。透過持續的練習與記錄，我有較為了解這些工具的使用方法，並理解相關語法。

心得感想

在這次實習工作前，我沒有任何資訊類相關工作經驗。這一年的實習機會讓我累積了寶貴的工作經驗，雖然我所做的工作主要是手動測試，不用寫程式，但透過這次的實習，我深入了解了產品驗證的流程，也學會了如何依據測試案例逐步驗證功能、記錄異常狀況並回報問題。

在測試過程中我經常遇到無法判斷是否為錯誤的情況，常常需要向同事請教，也擔心自己頻繁向同事發問會造成打擾。但我很感謝主管與同事們都願意耐心指導我，幫助我釐清測試流程，讓我逐漸了解測試邏輯與產品規格，也學會了如何在遇到問題時先嘗試思考、查找資料，再有條理地詢問。這不僅提升了我的獨立解決問題的能力，也讓我在溝通與表達方面更加進步。

這次的實習經歷讓我體會到細心與耐心在工作中的價值，也讓我理解測試在整個產品品質控管中扮演的重要角色。剛入職時，主管曾提醒我們，測試工作是產品開發流程中重要的一環，若測試結果送出後被發現測項有異常，不僅需要重測，還會造成時間與人力的浪費。這些實務經驗讓我更深刻地認識到責任心與專業態度在資訊相關職位中的重要性，也讓我對未來的職涯方向有了更真實的體悟。

總體而言，這次實習不僅讓我累積了專業技能，也培養了我的工作態度、問題解決能力與溝通協作能力。我相信，這些經驗將成為我職涯中的重要基礎，支持我在資訊領域持續成長，並在未來面對更具挑戰性的工作時，能更自信、有效率地應對。