

No.
79

NOV 2022

發行人：周天穎 總編輯：葉美伶
編輯委員：辜文元、劉若瑜、孔祥芝、林方芸
美術編輯：許育綸 執行編輯：汪盈秀
發行所：逢甲大學GIS中心

GIS MAGAZINE

溝通 · 交流 · 分享

技術專欄

01 » 技術類【陳家泓】
Vue從入門到應用

02 » 技術類【姚佩純】
雛型演進導向之系統分析方法

03 » 專案類【蔡承諺】
守護山民—一窺水保防災應變

人物專訪

04 » 客服類【謝紅玫】
客服二三事

3WA問題解決專家—何宗翰
【汪盈秀-採訪編輯】

05 » 設計類【廖志勝】
利用CSS格線快速布局儀表板配置



葉美伶 處長

猶記今年初的第一期月刊在三月份出刊，我用「是技術也是藝術」、「是科技也是玩具」以及「是傳統也是前衛」三個標題說明六篇文章的概要，似乎也就是前一陣的事而已，但現在卻已經是年末將至節氣小雪的季節。是哪一首詩寫的這個季節應該是”一片飛來一片寒”，但因為極端氣候的影響，台灣這裡的氣候只有早晚微涼的舒適，也適合靜心的閱讀 2022 年的最後一期月刊。

本期月刊同仁投稿共 5 篇，包含技術、專案、客服與設計等領域。**科管處家泓**以台中社會住宅管理系統為例，介紹 Vue 從入門到應用的方法；Vue(音同 View)，Wiki 的解釋是”一個用於建立使用者介面的開源 MVVM 前端 JavaScript 框架，也是一個建立單頁應用的 Web 應用框架”，且 2016 年一項針對 JavaScript 框架的調查表明，Vue 有著 89% 的開發者滿意度；是一個較為新穎的技術，值得深入應用與探討。**統合處佩純**列舉三種系統分析方法，說明其定義及範疇，最後以雛型設計工具及方法用於智慧水網管理資訊系統開發，以降低系統使用需求的認知差異。**空資處承謙**其工作業務主要以水保局防災整備及應變計畫為主，故此次以”守護山民”為題，讓大家了解水保防災應變之程序，中心集團有許多與防災應變相關的計畫，如承謙所述”…默默地在每一次的地震、颱風事件發生期間夙夜匪懈，不分晝夜地緊盯各地區狀況，隨時應變災害發生”，也為辦理防災應變計劃而驕傲。

中心集團每年完成數百個空間資訊相關系統，每個系統的規模、服務機制及使用者皆不相同，所以資訊系統的應用與使用者需求及回饋相當重要。**統合處紅玫**以客服二三事分享客服同仁的應對技巧，文中插圖引用特別生動有趣，句句都是出於工作的”客服之於我”的內心對話。最後也是**統合處志勝**的 CSS 格線快速布局儀表板配置的方法，從前言開始點出專案任務所需解決的問題，在結語還用爸爸與小孩做比喻，讓讀者可以了解 CSS-Grid 的內涵及未來的發展趨勢。

月刊的人物專訪是在今年新增的報導，旨在透過專訪了解同仁的專長養成過程及其工作外的生活點滴，本期專訪**統合處宗翰**，記得宗翰第一年來中心工作，參加 GI Day 的新人報告，熱情開朗的聲音讓大家印象深刻，轉眼也成為已服務十年的資深同仁了，精彩的專訪不容錯過 !!

在接近年末的這一期月刊出版前，感謝集團同仁踴躍投稿，發揮中心集團 I²C+S 的精神，我代表月刊編輯小組感謝同仁的支持！期待明年各位同仁也能發揮各種創意及研究的精神支持投稿，另一方面也希望同仁能詳閱每一篇文章，因為閱讀可以加強文字敘事能力且可以讓簡報及企劃案品質提升，好處多多！

葉美伶

2022/11/21

人物專訪

3WA 問題解決專家—何宗翰

■ 管理處 / 汪盈秀 - 採訪編輯



何宗翰 John 小檔案

南台科技大學資訊管理系學士

中心經歷

2020/01 - 今	副總工程師
2013/6-2019/12	資深工程師
2012/4-2013/5	系統工程師

專長

.Net、JavaScript、C、PHP、Python、MySQL、PostgreSQL、API、ORACLE、Visual Studio、ArcGIS、GIS 程式開發、GUI 單機程式、OCR 影像文字辨識、多電腦分流處理、行動裝置系統開發、系統建置維護、架構分析設計、單機排程程式、資料庫分析與設計、資料庫建置與管理、資料庫程式開發、資訊系統開發、電腦硬體裝修、網站伺服器、網站程式設計、網路地圖功能開發、網路系統配置、應用系統建置

獎勵紀錄 嘉獎 3 支

網路上常說「高手在民間」，在集團內就有這麼一位神人高手：統合處何宗翰 -John，翻開 John 個人專長簡歷，洋洋灑灑一大排的程式語言、資料庫處理、系統開發，軟硬體兼具，讓人忍不住讚嘆：「哇！有什麼程式是你不會的嗎？」肥米輸入法、Netflix 雙字幕與字體設定功能、因應疫情期間居家辦公的遠端開機、LineBot 機器人群組通知、網站與主機運作狀態偵測機、本田 NSR-150 經典二行程機車自製程式 CDI 與 RC 伺服電腦……都是 John 在工作之餘的個人傑作，有些幫助了部門，有些則幫助了廣大鄉民們；靠著對程式的熱愛，埋首於解決問題，也成了 John 生活中最大樂趣。

談到 John 對程式開發啟蒙的起點，是在國小 2 年級。第一次接觸到電腦，是 MS-DOS 3.2 的年代，John 就開始玩起 PE2、倚天中文 ET3、PcTools 編輯器，恰巧當時家裡樓上的房客是中興大學學生，大哥哥們熱心地當起家教老師，開始教 John 簡單的程式小遊戲，透過破解遊戲程式，修改儲存檔中的經驗值、等級，享受到快速晉級（偷吃步？）的樂趣，也燃起了這個小學生對程式的愛好的火苗。

爾後高職進入資訊科，二專進入計算機工程組，也是奠定 John 電子電路專業的關鍵時期，而看似與程式開發無關的電子學，這樣的背景對程式開發有什麼幫助或影響嗎？「當然有啊！」John 眼神發亮地說。接著詳細解釋了電子電路原理，而在單晶片進程演變中最大的關鍵點就是：「用程式控制」，其中影響最具關連性的「控制成本」便是 John 在開發時的主要思維，「若程式撰寫的方式可以讓你少放幾顆電容幾顆電阻，電路變簡單的同時也縮小了體積。」

戰績輝煌的專屬資料庫，動手做才有用！

喜歡研究新事物新技術的 John 在進入集團之前，經歷過非常多不同產業的工作，從補教業教學與評測系統、新創自由軟體開發、物聯網電子相框、產線物流 RFID E-Tag 條碼機、串流平台聲紋辨識等等，每個經歷幾乎都是在當時最前端的技術應用，如此豐富的經驗累積也成了 John 實戰筆記的最佳紀錄，John 認為他現今的能力是堆疊式成長而來，把每一次的經驗存下來，逐漸豐富了自己的資料庫。John 特別提到「資料」與「資訊」的差別，將這些資料消化後轉換成有用的資訊，才是有意義的，也才能真正成為自己的東西。

榮獲集團 GI DAY 第一屆（2012）「洛基新人獎」最佳發表人的 John，當年有點無厘頭的簡報方式令人印象深刻，而另一個記憶點則是「修車高手」，熱愛摩托車的他笑說因為想省錢所以開始學自己維修，從維修手冊開始研究，進而了解制動系統、供油系統、傳動系統、點火系統……聊起摩托車滔滔不決，除了修自己的車，也幫親朋好友修車，齊全的設備簡直可以自己開車行了！John 喜歡透過科技的力量解決問題，在摩托車維修上亦是如此，也因此認識了不少同好車友；因為喜歡，所以玩不膩，「動手做自己喜歡的事」，可以破解每一台車的問題，成就解鎖就是最好玩的地方。



圖 2. 圖 3. John 的車庫也是工作室，玩車就是要先拆座墊



圖 4 車友同好

自己動手做的習慣也貫徹在 John 的日常生活中，喫蝦米輸入法是不少人慣用的輸入法，John 也是其中使用者，但因為喫蝦米桌面版長久以來有視窗重複出現的 Bug，官方遲遲沒有修復，也曾遇到無法正常出字的視窗等問題；John 某天在練習 Python 程式時，想嘗試作個簡單的作品，就決定自己來試試，利用 Python 寫了一套仿喫蝦米 windows95 介面的「肥米輸入法」，從 2017 年開發後使用至今，原始碼也完全免費開放在網路上與網友們共享。在使用 Netflix 影音平台時，只因為投影至電視螢幕觀看的距離，覺得字幕實在太小了，索性就自己研究瀏覽器外掛的開發方式，重新建字幕，把字幕放大之外，還連帶做了雙語字幕，起初只在 PTT 上分享，其他網友使用了覺得很棒，建議 John 上架到 Google 商店，分享給更多人使用，於是從 0.1 版次不斷更新至 2.6 版次，也在今年初正式上架 Google Chrome 擴充應用程式了。

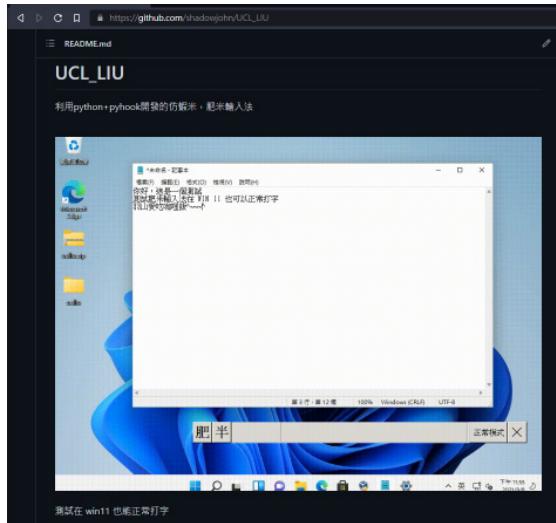


圖 5 肥米輸入法



圖 6 Netflix 雙字幕與字幕放大外框顏色、粗細效果

自發性的熱血行動力，屢屢建功

這樣「解決問題」的行動力，John 也同樣投注到部門中。在幾次資安事件中，John 察覺到電腦未關機的外部入侵風險，但礙於專案機動性，同仁們時常在下班後仍需遠端連回本機處理公事，須維持電腦開機狀態，加上疫情期間居家辦公的開關機需求，為了能同時解決「節約能源」與「下班開機」的問題，John 自主開發了遠端開機的程式，讓同仁可以放心的在下班關閉電腦主機，若有工作需求再透過系統遠端啟動電腦。還有善用 Line bot，創建「小山」機器人，成為統合處部門群組的小幫手，舉凡請假、開會、訂餐、掃地、小天使消毒等大小事，都可透過小山來管理及通知。甚至為了協助部門更清楚掌握手上專案系統的主機及網站狀態，開發了網站服務偵測機，即時監測網站主機使用情況，監控系統可以主動回報電腦基本資料狀態、硬碟空間使用歷程記錄、網站 Domain、SSL 憑證是否即將到期、夜間異常登錄、異常程式入侵等等，除了可預防專案主機無預警的故障，更是加強了在資安層面的保護。

GIS 二辦神祕遠端開機系統						
Computer ID	部門	誰的電腦	最後登錄時間	IP	MAC	電腦狀況
2	統合處-審查	Freakeng	2022/09/01 11:46			OFFLINE
3	統合處-審查	Pinky	21 分鐘前			OFFLINE
4	統合處-審查	Akira				OFFLINE
5	統合處-系統	Sky	59 分鐘前			ONLINE
6	統合處-審查	Kan	59 分鐘前			ONLINE
7	統合處-系統	Jeffrey	59 分鐘前			ONLINE
8	統合處-系統	Phoenix	59 分鐘前			ONLINE
9	統合處-系統	Karen	59 分鐘前			ONLINE
10	統合處-系統	Eric	59 分鐘前			ONLINE
11	統合處-系統	Mountain	59 分鐘前			ONLINE
15	準備	Andy	59 分鐘前			ONLINE
16	準備	Carl	59 分鐘前			ONLINE
18	準備	Odem	6 分鐘前			OFFLINE
21	統合處-系統	Shadow	54 分鐘前			ONLINE
22	統合處-系統	Sam	54 分鐘前			ONLINE

圖 7 遠端開機系統

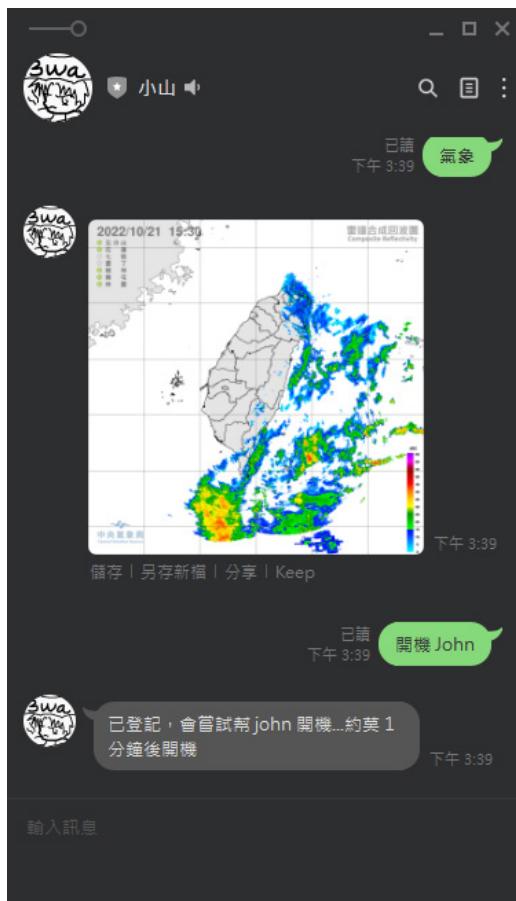


圖 8 透過小山機器人即可協助同仁遠端開機

集團在 2013-2015 期間，與日本產業技術綜合研究所（AIST）因專案合作關係，進行了工程師外派交流，當時還算是新人的 John 獲得這個難得的機會，跟著資深工程師一同到了日本進行 2 個月的專案工作。為了解決分析日本東北大地震福島核災造成的輻射問題，進駐到日本筑波 AIST，計畫彙整相關數據並進行資料視覺化，John 的任務需將 2000 種不同格式的 Excel 檔案、13 多億筆的資料想辦法匯入 OGC Sensor Observation Service (SOS) 平台並完整顯示，找出夏季與冬季福島輻射量差異是否適合居民返鄉，艱難的任務在與前輩共同協作下順利完成，也學習到了如何解讀資料的不同面向，是很寶貴的經驗。喜歡日本神社的 John 笑著說，短短 2 個月的日本生活，充分體會日本職場與享受日本文化，很希望未來還能有機會到其他國家交流。



圖 11 John 於 AIST 合影（最右為技術部主任 - 田中良夫博士）

網站服務偵測機										
序號	網址	狀態	最後更新	HTTP 儲存時間	最佳存取時間	最佳存取地點	最佳存取時間	回報時間	回報地點	回報狀態
1	臺灣氣象局	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	7 分	世界	正常
2	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	8 分	世界	正常
3	台灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	9 分	世界	正常
4	中央氣象局	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	14 分	世界	正常
5	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	145 分	世界	正常
6	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	147 分	世界	正常
7	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	148 分	世界	正常
8	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	149 分	世界	正常
9	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	150 分	世界	正常
10	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	151 分	世界	正常
11	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	152 分	世界	正常
12	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	153 分	世界	正常
13	臺灣地圖	正常	2020-01-08	2020-01-01	2.5 小時前	綠色	正常	154 分	世界	正常

圖 9 網站服務偵測機



圖 10 即時偵測系統狀態



圖 12 AIST 福島輻射監測平台

練好基本功才是成為武林高人的最佳秘笈

前陣子 ITD Group 舉辦的「第一屆工程師速度與激情」競賽，榮獲冠軍的 John 最快秒數與最慢者的差距有 20 倍之遙，看起來對程式游刃有餘的 John，有沒有讓他感到挑戰的專案？他歪著頭想了很久，談起最近正在做的水利署專案，需要同時介接六、七千隻攝影機、警政署五萬多支攝影機，包含單張與影像串連，短時間內達成下載並判斷與處理大量資料，在多執行緒的模式下解析影像正確性及確保系統穩定可正常運作，針對不同攝影機的客製化，還要克服水利署內建防毒軟體帶來的問題，整個過程困難重重，也花費了 John 不少時間，但當然，他還是完成這項挑戰任務，也在 .Net C# 的多執行緒關卡升級了不少經驗值！

這麼厲害的高手，問他有沒有什麼撇步可以分享給大家，John 馬上回答：「看書！」看似簡單卻是最扎實的訓練方式，John 說他很喜歡逛圖書館與舊書攤（剛好再配上一塊隔壁攤的雞排！），在書堆裡挖寶，常有意想不到的收穫；但現在科技進步這麼快，有“時差性”的紙本書籍對程式開發真的還有幫助嗎？「有！」John 非常肯定地說道，從根本的基礎中學習才能再衍伸應用，例如在這本書上看到某個撰寫程式的方式、技巧，一個生活小創意或是一張電子電路圖，當下或許不會用到，但腦中記下書中索引，好用就抄寫到自己的資料庫，日後一定能派上用場。John 打趣的說，蹲廁所時就是看書的最好時機！（工程師們！把廁所的漫畫書換成工具書吧！）

習慣用 Blog 紀錄的 John，期許自己是個「問題解決專家」，所以把 Blog 命名為 3WA (What/Why/When/Answer)，想盡辦法「自動化」，希望透過程式把反覆做的事用聰明且快速的方式去完成，簡單的初衷伴隨對程式的熱愛，造就了今日的神人高手～

「痛苦麻煩的事，困難一、二次就好，再次遇到相同的問題，就是專家了……其實就是懶啦！」John 大笑後謙虛下了這個結論。

John 的熱血精神可說是充分落實了集團 I²C+S 理念，「3WA 問題解決專家工作室」 (<https://3wa.tw>) 也歡迎大家一同交流！



圖 13 「第一屆工程師速度與激情」競賽發表



圖 14 John 獲頒競賽冠軍獎金

圖 15 水利署水情影像平台管理後台

01 Vue 從入門到應用

科管處 / 陳家泓



前言

大家好，這次很榮幸可以投稿 GIS 月刊，本次的投稿內容為介紹使用 Vue 應用於專案上，本次是筆者首次將 Vue 應用在專案上，不足之處也請多包涵和指教，本篇後續會將台中社會住宅稱為社宅，由於社宅專案內容豐富，故會擷取社宅部分內容分享出來，主要介紹 Vue 所使用的技術，以專案開發的角度來介紹如何使用 Vue，比如 Vue 常見關鍵字、頁面渲染、函式方法呼叫…等，內容雖精簡但絕對紮實。

一. 前後端分離架構

說明架構

首先說明社宅專案所應用技術組成（圖 1），前端部分以 JQuery、Vue[1] 等工具來實作內容，UI 部分則是以 Bootstrap[2] 為主，後端是以微軟的 .Net 為主，配合 SQL Server 作為資料的存取。

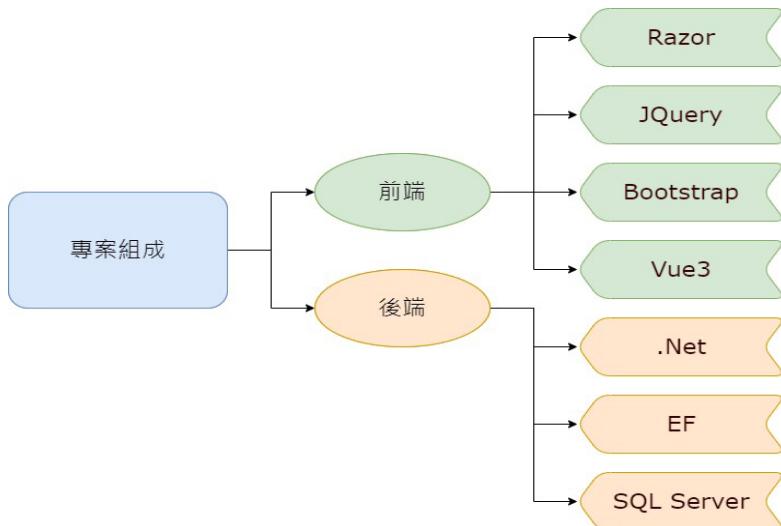


圖 1 社宅專案所應用技術組成

運行架構是由使用者操作前端站台，前端再以 Web API 方式來呼叫後端站台，配合使用 EF(Entity Framework Core) 進行 DB 的存取，以處理該次要求資料。（圖 2）

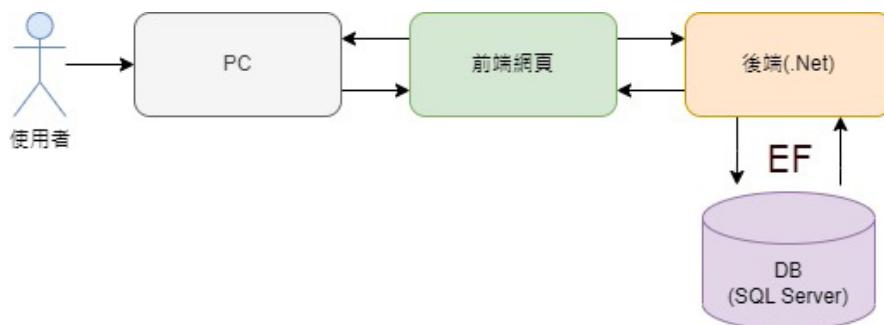


圖 2 社宅專案架構

二. Vue 介紹

前端實作上使用到多種技術框架，本篇內容會以 Vue 為主來介紹，且建立在讀者們都已經有一定程度的認識 JS、JQuery。Vue[1] 是由尤雨溪 (Evan You) 於 2013 年工作時的業餘專案，結合了過去開發 AngularJS 的經驗，開發出 Vue.js 這一套框架，並於 2014 年 2 月在 Hacker News 與 Reddit 等網站對外正式發表，Vue 目前有 2 和 3 的版本，此次應用都是以 Vue3 為基準，且 Vue 有高度向下相容性，所以不太需要擔心會有過多的學習成本在不同版本。

Vue 的特色

- 漸進式 JavaScript 框架 (The Progressive JavaScript Framework) [3]
- model–view–viewmodel [4]
- 元件系統 (Components System)
- Vue.js Command-Line Interface(Vue CLI)
- Vue Router
- **官方文件詳細**

Vue 最主要的架構就是 model–view–viewmodel[4]，可說想學會 Vue 就必須要了解其架構，至於什麼是 model–view–viewmodel 呢？簡單來說就是用 data 綁定的方式來渲染頁面，我們只需要注重在如何把資料整理好，就可以把想像中的頁面呈現出來，對於第一次看到這概念的我看的也是迷糊，但經過幾次實作之後就會開始體會到此架構所帶來的好處。

Vue 相較於 JQuery，其主要的差異在於 Vue 是以資料的方式來管理頁面節點 (DOM)，不同於 JQuery 指令渲染的方式來管理頁面節點 (DOM)，再來 Vue 相較於 JQuery 來說減少了更多的 DOM 操作，雙向資料綁定讓我們可針對資料做改變後，頁面就可以即時更新資料，可自行組合成 Component，透過 Component 可達到重複利用也可以分享給其他人使用。Vue 其中主要的關鍵字有以下 data、mounted、updated、computed、watch、methods。接下來會在成果展示部分來解析實作過程。

三. 成果

這章節將介紹專案使用 Vue 的成果社宅介紹頁（圖 3），由上方大圖、基本資料、四個小區塊組成。



圖 3 社宅介紹頁

本次使用官方可用的 CDN[5] 的方式來引用 Vue(圖 4)，圖 4 中 @ 後的 next 可以改成指定版本，接下來在需要使用的頁面上新增 Vue 會用到的 js 檔(圖 4)。

```
<!-- Vue 引入 -->
<script src="https://unpkg.com/vue@next"></script>
<script src="./vue.js"></script>
```

圖 4 CDN 引入 Vue

初始化 Vue 實體(圖 5)，Vue.createApp 就是 Vue 用來新建 Vue 物件的呼叫方法，括號中可以用 object 的方式來傳遞 options，最後用 mount 方法將 Vue 與 DOM 做綁定，綁定方式可參照 Js 的 querySelector()，而為何會需要綁定動作？是因為 Vue 核心概念之一為元件系統，所以會把 HTML 切成獨立個體，讓功能單一化，方便組成元件。

```
1  var VueComponent = Vue.createApp({
2    +  data(){...},
27   +  mounted(){...},
62   +  updated(){...},
77   +  methods:{...}
154 }).mount('.housing');
```

圖 5 Vue 初始化語法

那我們如何在頁面上顯示資料呢？先設定好 Vue 的 data(圖 6)，data 為 Vue 獨立個體內部使用的資料，然後使用 {{}} 這個模板語法把 data 安插在 HTML 裡就好了(圖 7)，頁面執行時的結果就如圖 8 ，仔細看才發現 data 裡面沒有地址資料啊？先不要急，往下面看將會為你解惑。

```
<div class="pin">{{BuildingInfo.Traffic}}</div>
```

圖 7 模板語法 HTML

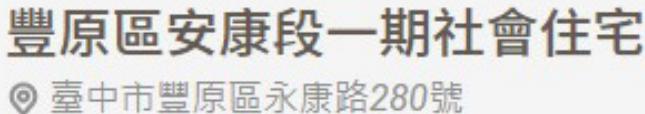


圖 8 Vue 模板語法結果

```
data() {
  return {
    originUrl: env.FrontEnd,
    HouseDetailOnline: '',
    HouseData: null,
    FirstPhoto:[],
    SecondPhoto:[],
    BuildingInfo: {
      Traffic: "",
      BuildingName: "",
      BuildingNo: "",
      Room: ""
    },
    LayoutNo: "",
    NowChooseBuild: "",
    Layouts: [],
    defaultUrl: '', //線上看屋url
  }
},
```

圖 6 Vue 資料內容

下一步介紹 mounted，此關鍵字為 Vue 實體掛載完成時，最簡單的理解為等同於 JQuery 的 Ready。首先在 Vue 初始化的 options 裡面新增 mounted 語法(圖 9)，在裡面新增呼叫 API 抓取資料的程式，再把回傳的 response Data 傳到 Vue 的 data 裡面，圖 9 中 this 指的是 Vue 實體本身，需要注意這部分有使用到箭頭涵式。這樣頁面上只要是該 Vue 實體有綁定到的範圍都會根據 data 的變化即時顯示，不需要再額外寫事件去處理 DOM，所以圖 7 的 BuildingInfo.Traffic 這時就會從圖 6 的空字串被覆蓋成新的資料。

```
mounted() {//DOM全載完 執行
  Repo.Run({
    Request: Svc.GetBuildingInfo(request.BuildingNo)
  }).then(response => {
    this.BuildingInfo.Traffic = response.Data.Building.Addr;
    this.BuildingInfo.BuildingName = response.Data.Building.BuildingName;
  });
},
```

圖 9 mounted 語法

同理我們只要其他地方也用 {{}} 模板語法綁好資料，就可以呈現出資料（圖 10）。

基本資訊

■基地簡介：本基地建築為1棟地下2層地上13層之電梯住宅，為本市首座社會住宅，已於107年3月完工，提供200戶予市民。
■基地特色：基地一樓設有健康關懷服務站、社會服務站、共好實踐基地等設施，其中健康服務站委由中國醫學大學附設醫院進駐，提供社宅及附近居民相關醫療諮詢、成人預防保健、健康講座等等多元醫療服務。另社會服務站則委由伊甸社會福利基金會營運管理，提供個案工作服務、福利諮詢、長輩共餐及老少共學等服務，借助其豐沛社福量能提供多元化社會服務。

生活機能

基地中心點600公尺服務半徑範圍而言，設施有學校(豐田國小、豐南國中)、市場(南陽市場、永康黃昏市場)、公園綠地(南陽公園、田心公園)、機關設施(陽明大樓、署立豐原醫院、豐原區公所、田心里活動中心)、廣場停車空間等，生活機能完善。

週邊交通設施

豐原安康共好社宅緊鄰18公尺永康路，門牌號為280號，車行3分鐘內可接台3縣，10分鐘內可接中山高速公路，到台中市區(臺中火車站)車程約30分鐘。距離豐原火車站約2公里，步程範圍內可搭乘共8號路線的公車，通往豐原在地商圈或本市中心商業區。

圖 10 其他 模板語法成績

下一步教你如何把 HTML 檔案整理簡短乾淨，仔細觀察會發現頁面總是有些部分是不斷重複一樣的 HTML（圖 11），從圖 11 中看出來一格格圖片部分都是重複區塊，所以可利用其重複的部分整理一下 HTML（圖 12），再用 v-for 方式配合模板語法重複寫入 List 資料渲染 HTML，v-for 使用方式也很簡單，其中只要在重複地方的最外層 DOM 掛上 v-for 然後用 in 來區分 List 就 OK 了，而 v-for 後面的 :key 則是 Vue 為了有效率的渲染頁面和不造成渲染錯誤所規定的語法，圖 12 中 v-for 裡 index 就是目前是 List 當中第幾個（起始為 0），圖 12 還可以看到叫做 template 的元素，這元素用法類似大括號，不會在頁面上實際渲染出 DOM，而是根據裡面模板內容去執行，可以當作邏輯處理時的利器，此時眼尖的讀者也一定有看出 v-if 這有點熟悉卻陌生的語法，其實也就是我們常見的 if 判斷式，分別還有對應的 v-else、v-else-if 搭配使用。

房型資訊



圖 11 明顯重複的 HTML

```
<h4 class="big-title text-center">瀏覽其他房型</h4>
<div class="housing-body-house-types ">
<template v-if="HouseData.OtherRooms != null">
    <template v-for="(OtherRoom,index) in HouseData.OtherRooms" :key="index">
        <div class="item">
            <div class="img-cover">
                <img :src=`${originUrl}/${OtherRoom.Photos.Src}` :alt=`${OtherRoom.Room}`>
            </div>
            <div class="info flex-between-wrap">
                <div class="dividers fz-14">
                    <span>{{OtherRoom.Room}}</span>
                    <span>{{OtherRoom.MinArea}}~{{OtherRoom.MaxArea}}坪</span>
                </div>
                <div class="price">NTD <span>{{OtherRoom.Rents[0]}}</span>起/月</div>
            </div>
            <div class="overlay">
            </div>
            <a :href="`${originUrl}/House/HouseDetail?BuildingNo=${OtherRoom.BuildingNo}&Room=${OtherRoom.RoomKind}`" class="stretched-link"></a>
        </div>
    </template>
</template>
</div>
```

圖 12 整理重複性 HTML

接下來把重複性的作法一樣用在接下來的房型表格部分（圖 13），結構（圖 14）部分一樣是用 v-for 來重複染畫面，但 v-for 的部分包進來的話畫面就會過於狹小，為了讓讀者看得清楚，只針對重點 HTML 部分做顯示，圖 14 的 Layout 就是 v-for 的陣列個體。

租金與設備					
房型資訊			租金與配備		
房型	總戶數	出租面積(坪)	第一年月租金 (含管理費)為市價之7折。	第二年月租金 (含管理費)為市價之7折。	第三年月租金 (含管理費)為市價之7折。
一房型	120	11.82~13.41	5,100~5,700/4,700~5,400	5,100~5,700/4,700~5,400	5,100~5,700/4,700~5,700
二房型	60	23.58~23.79	7,900~8,000/7,400~7,500	7,900~8,000/7,400~7,500	7,900~8,000/7,400~8,000
三房型	20	37.41~37.41	12,300~12,300/11,600~11,600	12,300~12,300/11,600~11,600	12,300~12,300/11,600~12,300

圖 13 房型表格資訊

```

<!-- 房型資訊 -->
<div class="basic-info">
    <h5>房型資訊</h5>
    <table class="table table-striped">
        <tr>...</tr>
        <tr>...</tr>
        <tr>...</tr>
        <tr>
            <th>
                <div class="flex-left-center">
                    {{changeRentDes(BuildingInfo.BuildingNo,2)}}

                    <a href="#moreInfoModal_2" class="more-info-btn" data-bs-toggle="modal">
                        <img class="icon-info" :src='${originUrl}/assets/img/icon-infomation.svg' />
                    </a>
                </div>
            </th>
            <td>{{CombineSlash(toCurrency(Layout.RoomInfo.MinRents[1]), toCurrency(Layout.RoomInfo.MaxRents[1]))}}</td>
        </tr>
        <tr>...</tr>
    </table>
</div>

```

圖 14 房型表格 HTML

圖 14 可見 {{}} 模板語法內有疑似 function 的出現，CombineSlash、toCurrency 就是 Vue 可自定義的 methods(圖 15)，用法就跟 HTML 的事件綁定非常類似，且還可放在模板當中作為資料面的處理。

```

methods: {
    toCurrency: function (num) { // 數字加千分位
        if (num === undefined)
            return "0";
        var parts = num.toString().split('.');
        parts[0] = parts[0].replace(/\B(?=(\d{3})+)(?!\d))/g, ',');
        return parts.join('.');
    },
    CombineSlash: function (str1, str2) { // 合併租金顯示字樣
        let result = '';
        if (str1 === str2)
            result = str1;
        else
            result = ` ${str1}~${str2} `;
        return result;
    },
}

```

圖 15 methods 使用

有些人會問說圖 12 、圖 14 為何有些屬性前方會加上冒號，這也是 Vue 的語法，冒號是 v-bind 的意思，代表可以將 data 的資料綁到屬性，只用冒號就是 Vue 的語法糖 [6]，更可搭配 js 的樣板字串 [7] 組成想要的字串。

更多 Vue 的使用方法與技巧，將在下次的分享中為大家做更詳盡的介紹。

四． 結論

使用新技術免不了會遇到問題，其中最大的問題莫過於概念理解與實作經驗，比如我在使用 Vue 開發的過程中還是多少會用 JQuery 等方式來直接操作頁面，導致頁面一些 DOM 發生不可預期的問題，造成與 Vue 資源衝突，但由於 Vue 方便上手、擴充性佳，只要習慣資料控制的方式，很快會發現比原本的開發速度還要快上許多，且因為有其架構特性，所以在維護方面也更容易找出問題加以排除，但前提就是需要學習新的操作模式，而是這次選擇使用 Vue 來開發就相對放棄 JQuery 傳統的開發工具，如今前端框架多元的情況下 (Vue、React、Angular)，我們需要根據開發情境來使用工具。

參考文獻

- [1] Vue 官方網站
<https://vuejs.org/>
- [2] Bootstrap 官方網站
<https://getbootstrap.com/>
- [3] Vue.js，何謂漸進式框架？
<https://gotraveltoworld.medium.com/vue-js-%E4%BD%95%E8%AC%82%E6%BC%B8%E9%80%B2%E5%BC%8F%E6%A1%86%E6%9E%B6-7d0281a7efa9>
- [4] Modelviewviewmodel
<https://book.vue.tw/CH1/1-1-introduction.html#vue-js-%E8%88%87-mvvm-%E6%A8%A1%E5%BC%8F>
- [5] 什麼是 CDN？
<https://www.cloudflare.com/zh-tw/learning/cdn/what-is-a-cdn/>
- [6] 語法糖
<https://zh.wikipedia.org/wiki/%E8%AF%AD%E6%B3%95%E7%B3%96>
- [7] 樣板字串
https://developer.mozilla.org/zh-TW/docs/Web/JavaScript/Reference/Template_literals

02 雜型演進導向之系統分析方法



統合處 姚佩純

前言

近年來由於網際網路的蓬勃發展，政府或民營單位也由傳統的紙本化作業，逐漸轉型為資訊系統化管理。在轉換過程中，勢必會經過一段陣痛期，例如雙方在需求的溝通或認知上有落差，導致一直無法了解實際需求，進而影響交付的資訊系統無法達到應有的效益。因此，如何能降低上述所說的需求認知差異並快速了解雙方的需求，「系統分析」所扮演的角色非常重要。

一. 常見的系統分析及建置方法

資訊系統的開發是依循系統化及邏輯化的步驟逐一進行，若沒有經過規劃設計且與客戶或甲方確認需求前，直接跳至系統開發，將造成修改的頻率大幅增加，進而影響程式碼的邏輯變得難以理解且後續維護不易，最終交付物亦無法達成客戶需求，而造成專案的損失。現行網路中也逐漸出現多種系統分析方法及理論，不同專案屬性也有各自適合的分析方法，下列將介紹三種較常見的分析方法。

(一) 漸增系統分析模式

漸增系統分析模式主要將系統需求區分為不同功能或子系統，每個部份需求則定義為一個週期，每一週期只循環一次，每一次循環將於系統中再增加新功能或模組，循序漸進直到完成系統開發。此分析方法較適合於需求較明確且有專案將有足夠的預算可投入進行每一週期的系統的功能開發。

(二) 敏捷軟體開發模式

敏捷式開發模式為現行最普遍被使用的分析方法，可以建立較短的開發循環，透過漸進式的方式開發軟體，並且可逐步依照客戶的想發或是市場的趨勢進行快速的調整，以符合現行需求，相較於其他分析方法更為彈性且不必將先前所有的開發項目砍掉重練。此方法適用於大型專案，可將其切割為較小的功能或模組，並隨著專案的進行，逐步依據用戶及市場的反應回饋修正開發的路線。

(三) 雜型系統分析模式

雛形系統分析模式主要先針對使用者較明確的需求部分，透過分析、設計及規劃等步驟快速完成雛形系統，在將此雛形系統與客戶進行討論，藉由雙方實際操作雛形系統，釐清及修改細節功能，反覆動作直到符合合約需求為止。

雛形系統分析模式最大優點是透過雛形展示，提高使用者會議討論的參與率，快速釐清需求以利後續進行系統開發。因此，本篇文章將介紹雛形系統分析模式的實務案例，讓讀者進一步瞭解實務上的應用方式。在進行雛形系統規劃時，必須使用到一些設計工具，下個章節將介紹幾個雛形設計工具，提供給讀者參閱。

二. 雛型設計工具

近年來網路上出現多種系統設計工具，不同的工具有其適合運用的地方或專案。系統分析人員可透過運用適合的設計工具快速進行系統分析，以加快整體專案開發速度，以下將介紹兩個簡單、快速上手且筆者較常使用的雛型系統工具與讀者分享。

● Axure

Axure RP 是一套擬真的網站建構工具，此工具提供多樣的功能 icon 或是小圖示，可以快速將使用者的需求透過此工具勾勒出雛形畫面，完成設計後可發布於網頁中，與客戶或甲方進行需求的規劃及設計確認，亦可直接將網址提供給對方查看。

● Prott

Prott 能設計各種的行動裝置介面原型，包含平板、手機與可自由設定大小的網頁介面，主要先把所有的系統雛型畫面依照先後順序排放後，僅需拉一條線將各個畫面串聯起來即為轉場，其操作部分沒有太過於繁瑣的流程，且最大的優點為各雛形頁面的轉場流暢度極高且無須付費即可使用，因此 Prott 為筆者最為推薦的一個設計軟體。

三. 實作案例

本篇文章將以一個實際案例進行說明—【澎湖地區智慧水網管理資訊系統開發】，此案為台灣自來水公司（簡稱：台水公司）的智慧水表系統化試辦案，整合機關既有系統建置智慧水網整合平台，輔助機關決策及提升用戶服務之效益，下列將說明如何透過流程化導向進行系統分析。

(一) 心智圖重點歸納法

「化繁為簡」為專案管理最重要且首要的環節。將甲方或客戶所提供的需求文件，把繁多的文字敘述，透過分類、歸納及收斂，轉換並產出模組化功能心智圖，以利後續可依各模組的逐項功能進行系統分析及深入探討，如圖 1 所示。

戶抄見量/系統淨供水量）。本模組資料來源介接提供水監測平台、抄表機傳輸系統及全區用戶表系統，各功能規劃詳述如表 2：	
表 2、產銷差績效管理模組功能說明	
功能	說明
即時數據查詢	1. 為本模組之主畫面。 2. 顯示之數據包括各類水源水量、各淨水場出水量、分區供配水量、售水量及售水率。 3. 主畫面顯示各淨水場出水量、供水量之即時監測數據、分區前售水量及售水率（三多、城北、民生、文澳、民族及案山等已換裝智慧水表之小區）、前期售水量及售水率（其餘尚未換裝智慧水表之分區）。
供售水量分析	1. 各類水源供水量分析：以圖表呈現各類水源之出水量及占比。 2. 分區售水率分析：利用 GIS 呈現各系統別及分區別之本期售水率及去年同期售水率，若有突增或突減者（差異百分比±5%），以顏色、圓形等方式提醒。 3. 動態售水率分析：針對已換裝智慧水表之三多、城北、民生、文澳、民族及案山小區，提供下拉式選單供使用者選擇欲查詢之分區，並顯示分區動態售水率及前日各小時售水率變化。
歷史數據查詢	1. 包括各類水源水量、各淨水場出水量、分區供配水量、售水量及售水率。 2. 透過點擊主畫面之各監測數據，顯示該數值之歷史數據趨勢圖，並可利用選擇功能，針對淨水場、分區、時間區間（起迄日）、時間頻率等，同步展示變化趨勢圖，並由使用者定義輸出報表格式與內容。 3. 漏水率之趨勢圖應與環評承諾之漏水率降低目標值進行對疊與比對。 4. 變化趨勢圖使用之圖表類型（如折線圖等）於需求訪談時確認。
報表產生	1. 包含區處及營運所日常所需之表單，如分區抄見量報表、分區售水率報表（當期售水率及去年同期售水率比較）、降漏月控報表、或自訂格式之報表。由使用者依選單方式選擇欲輸出之報表名稱。



圖 1 需求說明書與心智圖比對圖

(二)Axure 及 Prototype 軟體應用

完成心智圖的歸納後，可再依循甲方偏好 的設計風格需求，利用相關軟體工具（本篇文章專案使用的工具為 Axure RP）先行勾勒出功能雛形畫面，再由 UI 設計人員，利用他們的專業技術，製作出擬真化的系統雛形畫面，提升整體視覺化的效果，相關畫面如圖 2 所示。

圖 2 系統功能雛形介面

於需求訪談討論時，為能有效地降低功能範疇的發散，建議可先透過 Prototype 工具（本篇文章所介紹的 Prototype 工具為 Prott 所開發，此軟體的優勢為每張系統雛形畫面的切換流暢度高，且亦提供協助編輯），整合所有系統雛形畫面，並透過流程化的方式將所有畫面進行串接為虛擬網站，在與甲方或客戶進行討論，讓雙方能更聚焦於功能的應用及優化部分，提升整體的討論效益，範例虛擬網站如圖 3 所示。



圖 3 虛擬網站範例

(三)資料統整

以本文的實際案例來說，其宗旨主要為整合現行機關既有系統，如圖 4 紅框所示，各系統所提供的資料格式截然不同，例如有描述空間資料的 kml 檔、檔案匯出之 excel 檔或是直接透過資料庫讀取資料等方式，如何將各系統所提供的資料做系統化的統整，亦為系統分析重要的一環。

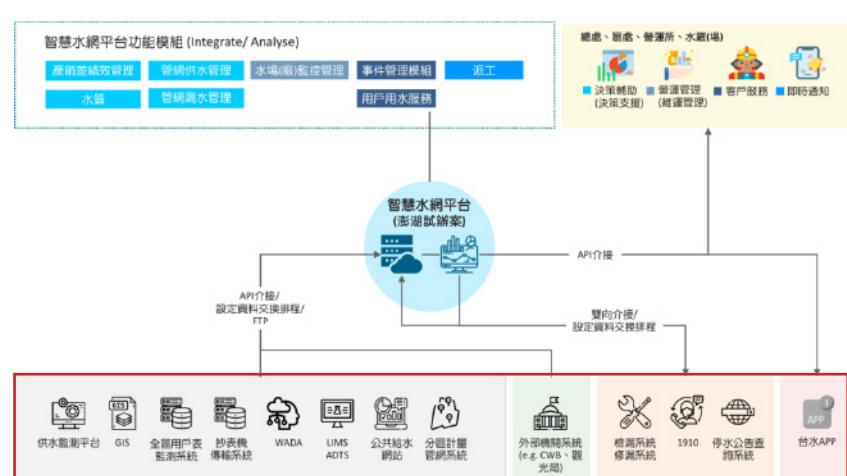


圖 4 專案系統整合架構圖

因此，在完成與甲方或客戶需求訪談及資料蒐集後，透過「資料轉換」將資料從一種格式轉換為另一種格式以符合現行專案所需，以利後續進行資料庫的規劃。而資料正規化管理，則能更有效率地再將轉換後的格式資料作分類，以降低資料重複性，避免後續有新增資料需求時，而無法進行整合，進而提升後續系統的整合力，資料表正規化關聯如圖 5 所示。

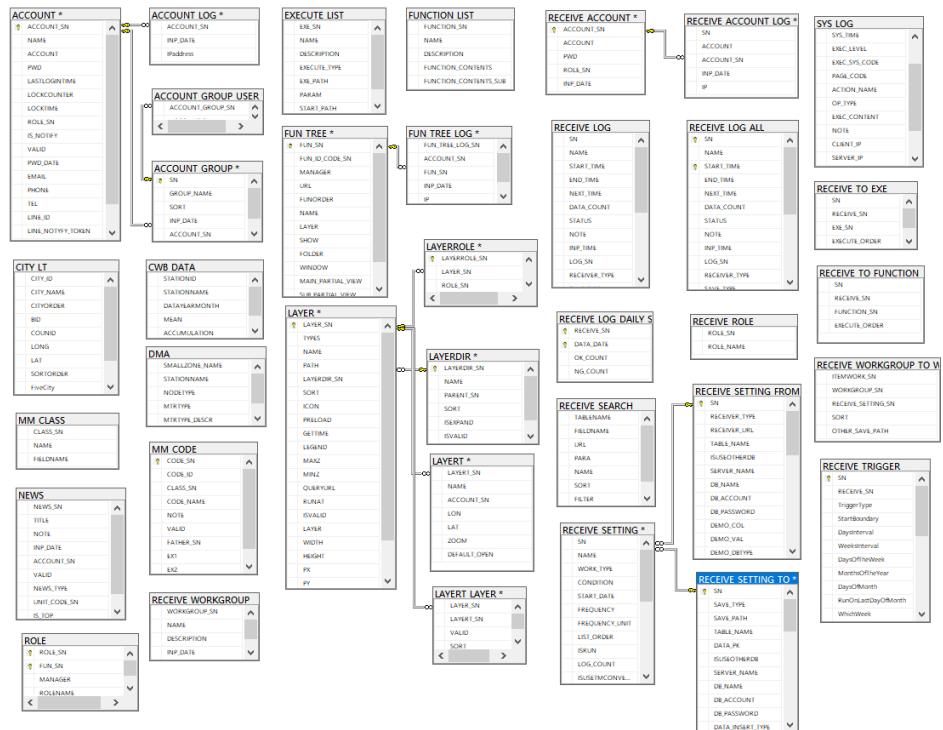


圖 5 專案資料表關聯圖

四．結論

縱然網路上有眾多系統分析的理論，如漸增式分析、瀑布式分析或是敏捷式開發…等等，但尚須與經驗進行結合。在實務需求訪談中，總是會有許多意想不到的議題出現，例如客戶總會將過去的操作思維套入專案中，或無法清楚描述他們的需求，種種問題皆需有利用實際經驗進行排除。因此，在完成每一專案後，可以將所學到的成功及失敗經驗，轉化成文件化記錄下來，作為後續專案的借鏡，長久累積後即可成為自己的資源資料庫。

參考文獻

- Axure RP 特色
<https://axure.userxper.com/feature/>
- 設計師必懂（一）WIREFRAME 與 PROTOTYPE 的不同
<https://designtongue.me/%E8%A8%AD%E8%A8%88%E5%B8%AB%E5%BF%85%E6%87%82-wireframe-prototype-%E7%9A%84%E4%B8%8D%E5%90%8C/>
- 什麼是資料移轉？
<https://www.netapp.com/zh-hant/data-management/what-is-data-migration/>
- 敏捷式開發？
<https://aws.amazon.com/tw/events/taiwan/techblogs/agile-development/>
- 資訊系統開發模式
<http://www.bestwise.com.tw/File/TeachApp/52WIT00705/ch02.pdf>
- 漸增式系統分析
<https://quizlet.com/tw/546067618/23%E6%BC%B8%E5%A2%9E%E6%A8%A1%E5%BC%8F-flash-cards/>

03 守護山民 - 一窺水保防災應變

空資處 蔡承諺



前言

台灣，是一個地震、颱風豪雨事件頻繁的島嶼，曾經的我們面對災害發生，只能懾於大自然的力量下束手無策。但隨著時代的變遷，科技的進步，有那麼一群人挺身而出，默默地在每一次的地震、颱風事件發生期間夙夜匪懈，不分晝夜地緊盯各地區狀況，隨時應變災害發生，也因為他們堅守著各自的崗位，才能使得我們有了更安全的家園。

一．水土保持局防災權責

為因應不同的自然災害類型，台灣在民國 89 年頒布了災害防救法，明定各項災害權責所屬，並且藉由各單位的分工，更加健全災害防救的體制，強化災害防救功能，以確保民眾的生命財產安全。

於災害防救法第三條便明定寒害、土石流及大規模崩塌災害、森林火災、動植物疫災為行政院農業委員會權責，而其中土石流及大規模崩塌災害便屬水土保持局防災權責範圍（圖 1）。



圖 1 土石流及大規模崩塌災害（圖片來源：土石流防災資訊網）

二．坡地災害管理

根據今(111)年度統計，全台目前有 1,729 條土石流潛勢溪流，238 處可能影響民眾生命財產安全的大規模崩塌潛勢區，面對眾多的潛勢災害區，應該要怎麼進行防災便是一大課題。也因此水土保持局藉由災害管理的方式，以減災、整備、應變、復原等四個階段（圖 2），分別進行消除災害發生因子、事前準備因應災害發生、災害發生時的緊急應變以及災害發生後的復原工作等，藉由各層面的防災應變作為，盡可能降低災害發生的風險。



圖 2 災害管理四階段

● 災害管理－減災

在災害發生前，透過各種軟硬體的減災措施，達到預防勝於治療之效，硬體減災措施如進行各項邊坡穩定工程、野溪整治、清淤工程、水土保持處理工程、土石流防治工程等（圖 3），其目的為盡可能減少土砂來源、降低土砂下移量體以及水流速度，透過消除可能造成土石流及大規模崩塌現象發生的因子，以降低災害發生的風險。



圖 3 土石流防治工程 - 梳子壩

硬體減災措施除了施作各項工程以外，亦包含建置現地儀器監測系統、快速掌握現地狀況，藉由長期監測現地資料分析、研究觸發災害因子與預警條件，以利災害發生時，可爭取應變時間，採取相關應變措施。

而軟體減災措施方面，則是藉由山坡地利用管理與特定水保區劃設，以降低山坡地開發，並透過公開土石流潛勢溪流調查的地區及影響範圍，盡可能讓民眾了解哪些區域是高風險地區，以及針對該處居民進行防災疏散避難規劃與訓練，透過災前訓練（圖 4），盡可能降低災害發生所造成的損害。



圖 4 南投縣國姓鄉南港村土石流防災實作演練

● 災害管理－整備

雖然藉由第一階段的減災，可盡量降低災害的發生，但仍然無法完全消除災害發生的風險，因此災前的防災準備工作不可或缺。而整備階段可分為人員與物資上的準備，防災人員的整備為各級防災人員的縱橫向聯繫、防災系統教育訓練以及村里的防災訓練等（圖 5），以確保災害發生期間由下而上的防災應變作為和資訊傳遞無虞。



圖 5 緊急應變小組教育訓練

而物資上的整備則是藉由盤點各式防災設備測試、各項緊急物資的盤點整備、所需的防災通訊設備檢查、防汛地區巡檢等，以確保可即時因應災害的發生；另外也會與各級單位會同召開聯繫會議，掌握各級單位所擁有的各項防災資源，以利於災害發生時，可以進行靈活調度使用，增加即時應變的防災量能。

● 災害管理－應變

當颱風豪雨或地震事件發生時，各級單位便會成立緊急應變中心或緊急應變小組，緊急應變小組開設情形（圖 6），以隨時掌握各地區現場情況。



圖 6 緊急應變小組開設示意圖

水土保持局在此期間，則會根據氣象預報與實際降雨量進行評估，若預估降雨量將達發生土石流或大規模崩塌的警戒雨量值，則會發布黃色警戒，針對發布黃色警戒區域的當地居民進行疏散避難勸告；若實際雨量已超過警戒雨量值，則針對發布紅色警戒區域的當地居民進行強制疏散避難撤離（圖 7）。

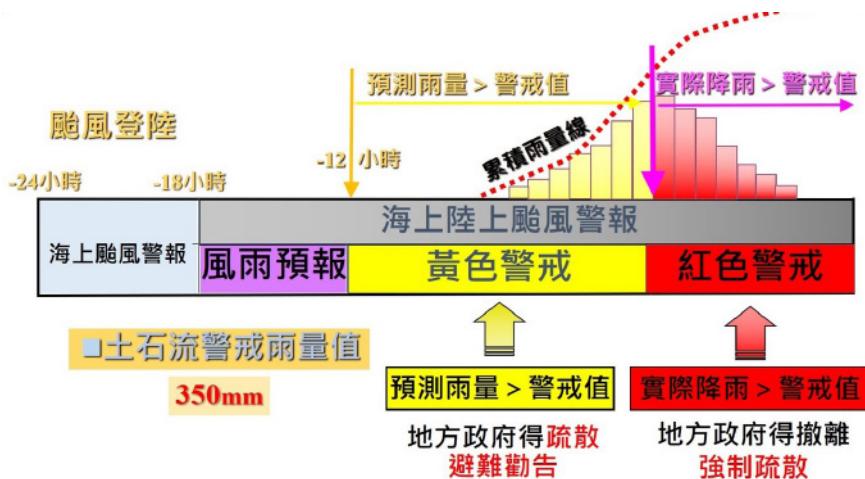


圖 7 土石流發布警戒流程

因此當氣候事件開始影響各地區時，應變中心便會透過不同管道隨時掌握轄內的各種情況，如新聞報導、當地防災專員或村里長回報、各級單位回報狀況以及現地儀器數值監測，藉此進行緊急應變與防災資源調度，通報流程如圖 8(下頁)所示。

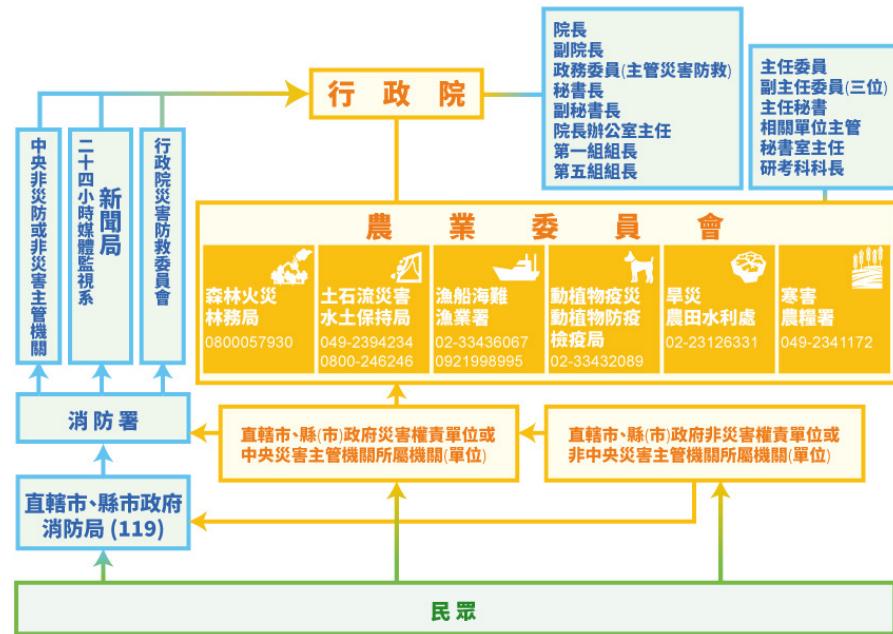


圖 8 災害緊急通報流程圖 (圖片來源：土石流防災資訊網)

當接獲轄區內發生災害事件時，應變小組便會根據實際狀況，與各單位進行縱向與橫向聯繫，迅速傳遞災害資訊，並協同相關權責單位進行緊急救災，土石流災害通報流程架構圖如圖 9 所示。

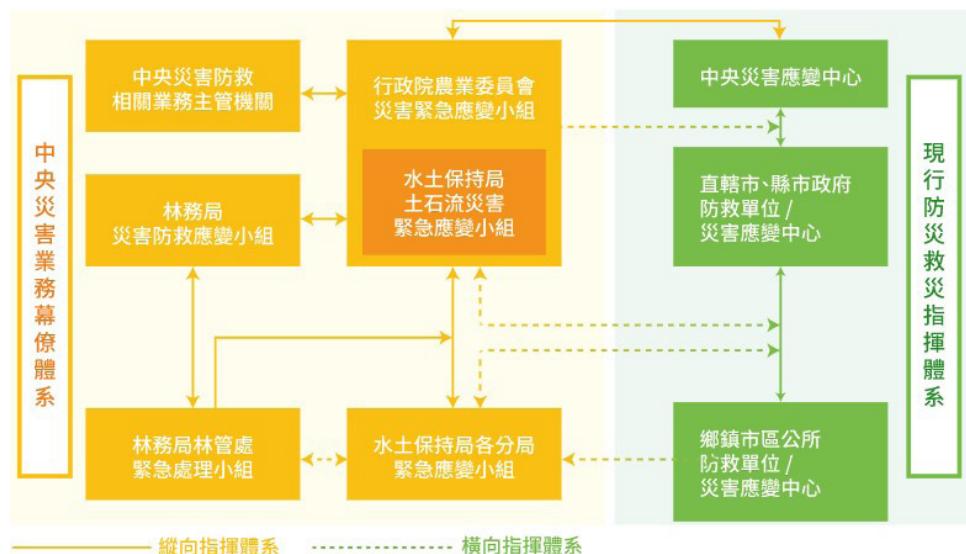


圖 9 土石流災害通報流程架構圖 (圖片來源：土石流防災資訊網)

根據災害事件的規模與類型不同，各單位進行不同方式的緊急應變處理，處理流程概略如下：

(一) 動員

根據災害事件規模程度不同，動員救災人力啟動救災作業，一般首先由警消人員以及民間救災團體協助當地居民撤離與緊急救災；當遇到大規模災害事件發生時，則會視情況動員軍方人力協助救災。

(二) 災民收容與撤離

在土石流潛勢溪流與大規模崩塌潛勢區於每年皆有更新規劃緊急疏散避難圖（圖 10），故當動員救災人員後，即依據疏散避難圖協助災民撤離至規劃好的避難場所進行避難，而當遇到無法撤離到避難場所，如道路中斷等情況時，則協助撤離至附近安全場所進行避難。



圖 10 疏散避難圖（圖片來源：國姓鄉公所）

(三) 緊急醫療救護系統運作

動員救災人員進入災區進行救災的同時，避難處所與醫療救護系統亦會於同一時間啟動，依據當地居民的醫療需求以及現場的醫療資源進行準備相應緊急救護準備。

(四) 實施交通管制

於接獲災害事件通報後，即動員附近的救災人力，如警消人員、防災專員或村里長等，於災害事件現場進行緊急交通管制，設置封鎖警示黃帶，或進行封路，以避免造成二次災害發生，亦有利於救災人員可迅速進入災區。

(五) 防止二次災害發生

救災人員抵達現場後，根據災害事件類型進行不同的緊急處置措施，如道路若有落石倒木，進行緊急搶通並設立警示管制道路；若有發生邊坡崩塌情形，除管制現場道路通行外，先行使用帆布覆蓋並於坡頂施作簡易導水等緊急處置（圖 11），藉由緊急處理措施防止災害擴大或二次災害發生。



圖 11 緊急處理措施示意圖（帆布覆蓋）

● 災害管理－復原

氣候事件結束後，應變中心將持續彙整相關災情訊息並追蹤現場處理情形，以利進行救災與復原，而針對現場災害發生地點亦會進行評估，若仍有造成立即性危險的疑慮，則會進行緊急處理工程，建立臨時防災設施（圖 12），以防止災情再擴大；而後再逐步進行復原重建工程等，以減低災害再次發生之風險。



圖 12 緊急處理工程示意圖（鋪設加勁格網）

三．結語

近年來，受到氣候變遷的影響，颱風豪雨事件更加走向極端化，短延時強降雨成為了常態，因此也更加考驗大家對於災害管理的應變能力。也因為有著一群人不眠不休的默默付出，藉由四階段的災害管理，方能多次在災害發生前，即時警示並進行相應的措施，大大降低災害發生的機率與損害程度。

然而在防災的課題上，僅靠一群人的努力是無法完全阻止災害的發生，唯有藉由降低對於環境的破壞，並於平日便熟練防災演練與作好充足的防災準備，於根本上建立起全民的防災意識，透過大家的努力，方可有效降低災害事件造成之損害。

參考文獻

- 土石流防災資訊網
<https://246.swcb.gov.tw/>
- 南投國姓鄉公所
<https://www.guoshing.gov.tw/1-1.asp?prokid=20200919091323&proid=20220613100924>

04 客服二三事

統合處 謝紅玫



前言

我們不是對話機器人，解決問題是最根源的出發點，而服務只是其中的基本條件。

要如何從客戶服務中得到雙贏，解決問題完成服務達到客戶的目的，同時獲得我們所需的資訊，進而精進我們的系統、產品，從中獲得養分內化為自己的一部份？

客服之路，還很漫長呢。

一．客服在做些什麼？

在 GIS 集團中絕大多數承接的計畫專案都有協助甲方建置資訊系統，而客服人員的存在主要目的就是為使用者指引操作與解決其所面臨的問題。

客服人員是第一線人員，每一個系統都有其專屬的客戶服務流程，如圖 1 所示，讓客服人員得以依循這個 SOP 來完成每一次任務，盡力排除使用者的問題。

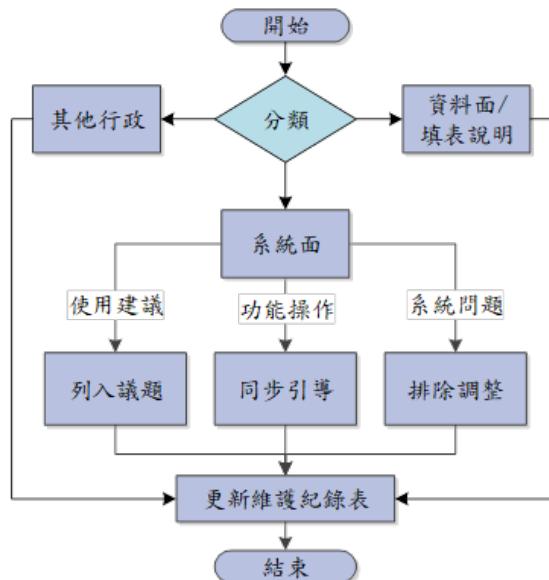


圖 1 水權用水範圍管理系統客服作業流程圖 ►

然而事情往往不會如你所預期的道路前進，身為第一線客服所要面對或處理的，不會僅僅是你所以為的系統問題，如同圖 2 的客服問題型態，客服問題種類與情況五花八門百百種。

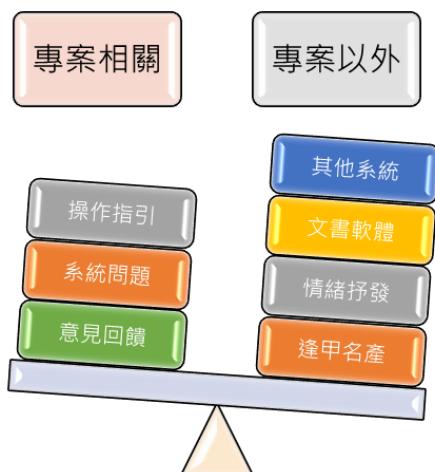


圖 2 客服問題型態 ►

有時候使用者來電明知道他要問的系統不是我們承接的，使用者卻回饋說「因為你比較了解，那個系統的客服都不懂我在說什麼！」又或者出現像圖 3 的情境，使用者來電結果罵了半天或是講不出所以然，最後發現他要問的根本不是我們系統！但我們依然只能保持理性面對、禮貌應對，提供我們所能給予的其他協助，例如提供協助查詢其他諮詢管道，通常使用者也會給予正面的回應而結束這次的客服。

曾經也有使用者一早打來「喂！紅玫瑰早安，我今天要去水利署開會，請問你知道要怎麼從清泉崗機場到黎明新村嗎…然後我晚上要住逢甲，你可以跟我說要怎麼搭車嗎？然後明天我還要搭高鐵去高雄…」因為對方是澎湖縣政府的承辦，難得回一趟台灣本島開會，但是不知道該如何搭車，所以這時候的我們就變成了他的導航。

有時候使用者只是需要一個可以討論議題的同伴、又只是需要一個當下得以抒發情緒 murmur 的對象，也因為這一層信賴，身為客服人員的我們也必須不斷的充實自我才不會被考倒！



圖 3 客服人員的奇聞怪事（圖片來源：Cherng）

二．客服的應對進退

從專案上的角度而言，系統的使用者 End-User、計畫的甲方，是我們主要的客戶服務對象。但實際上來說，周圍的每個人都是客服對象，因為每個人都可以是 End-User 的角色，每個人都可以對專案的相關事務提出自己的問題，當然有人可以給予說明解釋，這些過程都是客服的一種。

可是依據服務對象的不同，客服人員也需要學會用不同的方式來應對，例如當使用者來電已經是明顯情緒不佳的時候，絕對不要急著搶話回答，即便你的系統沒有任何問題，因為多半那個時間點對方已經不是要聽你的說明，急著回答可能只會加深對方的憤怒值，只要試著理性、平緩不帶語氣的回應「是」、「嗯」、「了解」等讓他知道你有在聽，有時候使用者的怒意就會下降，因為他影響不了你最後也只能回歸理性。

也有些時候因為使用者反應的問題不是客服人員可以直接回答，會需要進一步由系統人員協助確認，這時候內部橫向的溝通詢問問題的方式也是絕對的重要，例如：

- **使用者說 OO 功能不能用，請協助確認。**
- **使用者反應 OO 功能執行沒有反應，瀏覽器是 xxx，時間點大約是幾分幾秒，請協助確認。**

上述兩種拋出問題的描述，前者有可能會造成「你系統壞了」的誤解，而後者因為提供的線索比較多，系統人員就會知道可能是程式有例外的情況需要進一步追蹤確認，比較不容易造成字面理解上的誤會。所謂「說者無心，聽者有意」就是這種狀況了，所以，客服人員面對內外都必須持續學習適宜的應對進退！

三．客服存在的必要性？

在部分的職場文化中，客服人員的角色往往被視為是誰都可以勝任，彷彿能夠輕而易舉地取而代之，然而實際上確實是如此嗎？姑且不提透過取得相關證照展現自我價值的學術或技術面向的正面肯定，但如同過去在考試得高分的人才又一定能夠流利的口說應對嗎？我想大家都知道這不是絕對肯定的答案。

如果客服人員不存在，技術同仁是否還能心無旁騖的 Coding，時間是否會浪費在可能從來不熟悉的系統本身？主管們是否必須跳下來應對那些可能只是芝麻般大小的事情？在集團中主要以規劃師的職系來擔任系統客服，因為有第一線人員的存在，消化了前述可能面臨的窘境，讓系統、專案、公司得以順利運行。

當新進同仁要進入一個運行中的專案，要如何能夠讓快速上手系統，那就來當客服吧！在客服的過程中會面臨到各種問題會是督促你快速成長的良方（笑）

四．客服的收穫

平時總是在電話中接觸的使用者們，在辦理教育訓練的時候有機會碰面，所以也有幸在教育訓練的時候收到使用者們心意滿滿的小禮物，從圖 4 可以看出成果豐碩（笑，從生活用品、咖啡、水果、點心都有呢！衷心感謝使用者們的愛護！



圖 4 歷年收穫的小禮物們

五．客服之於我

客服人員需要強大的心理素質，需要能夠正向面對、需要吸收負面能量、需要維持情緒不易受影響、需要聆聽與理解、需要引導協助與支持、需要耐心貼心同理心。就像以前校園裡都會聽到有張老師信箱、張老師專線的存在般，我們也許很微小，但會有需要我們的一群人在。

「請、謝謝、對不起」，在台灣人人從小就知道的基本應對禮儀，可是在這個資訊爆炸的年代，似乎也已經不是那麼重要而被漸漸遺忘。在客戶服務的過程中，得到的一句簡單「謝謝」，也許只是一個習慣的應對，卻不是每一次的服務都能得到這樣一個簡單的回應。但簡單的兩個字，對我而言就是可以持續下去的一股動力。圖 5 的標籤雲，是客服的經驗中留在我腦海的記憶。

也許我們會遭遇客訴

也許我們有時候應對會不小心被牽著走

也許我們沒辦法第一時間給予最正確的解答

也許我們不能滿足你的期待

但我們還是一塊柔軟的海綿

在能力所及的範圍內，提供需要的人一點點點點的協助

我是不專業的客服

謝謝一直以來的客服過程中給予養分的每一位使用者

有你在，真好！



圖 5 客服之於我

參考文獻

- 每一通電話都是商機
<https://www.cw.com.tw/article/5005440>
- 客服中心 Call 住顧客的心
<https://www.managertoday.com.tw/articles/view/774>

05 利用 CSS 格線快速布局儀表板配置

統合處 廖志勝



前言

今年接到一份需要開發儀表板 (dashboard) 的網頁案子 (圖 1)，筆者看了設計稿之後，沉思了好一陣子，畢竟很少接觸到像儀表板這樣的網頁，不知如何著手進行切版。儀表板網頁的特性在於一頁式滿版區塊設計不能有上下卷軸，可以讓使用者一眼就看到全部資訊等…，所以在切版上跟一般網頁方式有點不同。因為 Html 的高度設定子層需要繼承父層的關係，對於儀表板高度也要剛好符合滿版的設計來說著實不容易撰寫。

之後便運用 display 中的 table 屬性完成了版面的切版，但是種巢狀包覆式的寫法有點不太清爽，心想是不是能有甚麼更好的寫法呢？突然想起了好幾年前研究過 Grid 屬性寫法，如果能以 Grid 屬性來撰寫，Html 碼應該會簡潔許多。以前 Grid 屬性在市場上非主流寫法，主要是過去 IE 瀏覽器並不支援，所以一般很少會利用 Grid 屬性來撰寫網頁版面。隨著近幾年 IE 已經終止服務了，目前 Grid 屬性各大主流瀏覽器都已經能完整支援了，Grid 屬性開始越來越多人使用。本篇文章以實際範例來介紹 Grid 屬性如何快速撰寫好一個儀表板的版面。



圖 1 儀表板設計稿

```
</div><div class="dashboardInfoArea">
<div class="grid_layout GL_01 alertnumBox" data-label="預警數量看板">
<div class="tableBox" > <h...<!-- END tableBox -->
</div><!-- END 預警數量看板 -->
<div class="grid_layout GL_02 alert2ListBox" data-label="第二級警戒值公司明細">
<div class="tableBox" > <h...<!-- END tableBox -->
</div><!-- END 第二級警戒值公司明細 -->
<div class="grid_layout GL_03 alert3ListBox" data-label="第三級警戒值公司明細">
<div class="tableBox" > <h...<!-- END tableBox -->
</div><!-- END 第三級警戒值公司明細 -->
<div class="grid_layout GL_04 unusualMAPBox" data-label="林園工業區異常事件圖台">
<div class="tableBox" > <h...<!-- END tableBox -->
</div><!-- END 林園工業區異常事件圖台 -->
<div class="grid_layout GL_05 CCTVBox" data-label="CCTV影像輪播">
<div class="tableBox" > <h...<!-- END tableBox -->
</div><!-- END CCTV影像輪播 -->
<div class="grid_layout GL_06 statisticsBox" data-label="統計圖表">
<div class="tableBox" > <h...<!-- END tableBox -->
</div><!-- END 統計圖表 -->
<div class="grid_layout GL_07 monitorBox" data-label="監測資料分析">
<div class="tableBox" > <h...<!-- END tableBox -->
</div><!-- END 監測資料分析 -->
</div><!-- END dashboardArea -->
```

圖 3 以 display 中 grid 屬性寫出來的 html 版面，簡潔許多

```
<div class="dashboardArea">
<div class="dashboardAreaLayoutBox">
<div class="dashboardAreaLayoutALL">
<div class="dashboardAreaLayoutL" data-label="左區">
<ul class="row">
<li class="col-12 alertnumBox">
<div class="tableBox" > <h...<!-- END tableBox alertnumBox-->
</li>
<li class="col-12 alert2ListBox">
<div class="tableBox" > <h...<!-- END tableBox -->
</li>
<li class="col-12 alert3ListBox">
<div class="tableBox" > <h...<!-- END tableBox -->
</li>
</ul>
</div><!-- END dashboardAreaLayout 左區L -->
<div class="dashboardAreaLayoutC" data-label="中區C">
<ul class="row">
<li class="col-12 unusualMAPBox">
<div class="tableBox" > <h2...<!-- END tableBox -->
</li>
</ul>
</div><!-- END dashboardAreaLayout 中區C -->
<div class="dashboardAreaLayoutR" data-label="右區R">
<ul class="row">
<li class="col-12 CCTVBox">
<div class="tableBox" > <h2...<!-- END tableBox -->
</li>
<li class="col-12 col-md-6 statisticsBox">
<div class="tableBox" > <h2...<!-- END tableBox -->
</li>
<li class="col-12 col-md-6 monitorBox">
<div class="tableBox" > <h2...<!-- END tableBox -->
</li>
</ul>
</div><!-- END dashboardAreaLayout 右區R -->
</div>
</div>
</div><!-- END dashboardArea -->
```

圖 2 以 display 中 table 屬性寫出來的 html 版面

一． CSS 格線布局

有 CSS Grid 是甚麼東西呢？它和網頁的表格一樣有行 (**Column**) 跟列 (**Row**) 的概念，CSS Grid 比一般的表格更靈活的定位自己的子元素。一般網頁表格在 RWD 的設計上很難達到排版的效果，利用 CSS Grid 能輕易實現較多樣變化的排版。CSS Grid 在 2012 年左右大概存在了，它默默發展了好幾年，國外也很多文章介紹與教學。礙於瀏覽器 IE 的限制，一直沒有被廣泛的使用。由於近年來瀏覽器的推陳出新，目前大多主流的瀏覽器都已經能夠支援 Grid 屬性了。

57.0	16.0	52.0	10	44

圖 4 CSS Grid 目前支援的瀏覽器版本

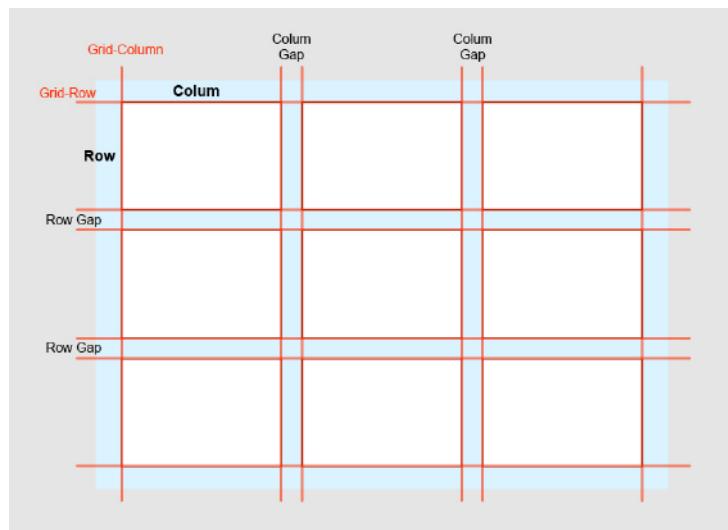


圖 5 CSS 格線布局類似網頁表格的概念

二． 儀表板 CSS 撰寫

以圖 1 的設計稿來說我們會有七個容器在版面上。Html 程式碼很簡單只要一個大 DIV 裡面有七個小 DIV 如圖 6

```
▼ <div class="grid-container">
  <div class="item1">1</div>
  <div class="item2">2</div>
  <div class="item3">3</div>
  <div class="item4">4</div>
  <div class="item5">5</div>
  <div class="item6">6</div>
  <div class="item7">7</div>
</div>
```

圖 6 Html 程式碼

然後我們先將設計稿簡化成如圖 7 也就是我們要完成的排版並幫它畫上格線。

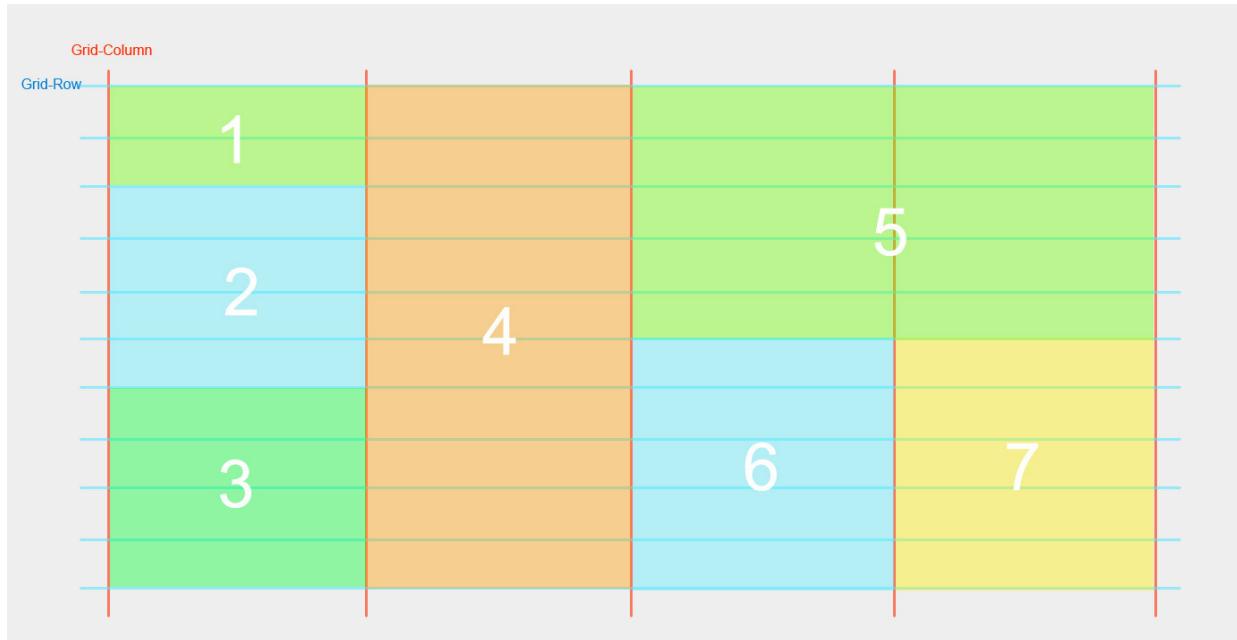


圖 7 將版面格線化

利用格線來看很清楚我們可以看到 Grid-Column(紅色) 的 Column 有 5 條，Grid-Row(藍色) 的 Row 有 11 條。布局出來的有 10 個 Row 四個 Column 總共 40 個小區塊。

.grid-container 為父層 DV 用來宣告 Grid 以及旗下子元素的布局。

grid-template-columns:1fr 1fr 1fr 1fr 1fr; 代表產生四個 Column 元素，

grid-template-rows: 1fr 1fr 1fr 1fr 1fr 1fr 1fr 1fr 1fr 1fr; 代表產生 10 個 Row 元素。

如果列數和行數太多用這寫法會寫很長所以也可以用 repeat(X , 1fr) 這樣的寫法。這邊也可以簡寫成 grid-template-columns:repeat(4, 1fr) 和 grid-template-rows: repeat(10, 1fr) 。

而 1fr 為 Grid 這邊專有的單位，代表其平均布局，例如在五個區塊中的 2fr 便是 $2/5$ 。Gap 屬性為行列的間距這邊設定為 10px。最後產生出的畫面與 css 如圖 8 。

The screenshot shows the CSS code for the grid container and the resulting layout preview. The CSS code is as follows:

```
grid-container {  
    box-sizing: border-box;  
    width: 100%;  
    height: 100%;  
    background-color: #f0f0f0;  
    padding: 10px;  
    display: grid;  
    grid-template-columns: 1fr 1fr 1fr 1fr;  
    grid-template-rows: 1fr 1fr 1fr 1fr 1fr 1fr 1fr 1fr 1fr 1fr;  
    gap: 10px;  
}
```

The layout preview shows a 2x5 grid of cells. The first row contains cells 1, 2, 3, 4, and 5. The second row contains cells 6, 7, and 8. The third row contains cells 9 and 10. The fourth row is empty. The fifth row contains cells 11 and 12. The sixth row contains cells 13 and 14. The seventh row contains cells 15 and 16. The eighth row contains cells 17 and 18. The ninth row contains cells 19 and 20. The tenth row is empty.

圖 8 父層的 css 與預覽頁面

父層宣告完可以發現 7 個元素很有順序地排列在 4X10 的欄位中並且每個元素間距為 10px。

接下來我們就要為每個子元素定義它們該何去何從了。以元素 1(.item1) 來說它介於 Grid-Column 的 1 和 2 之間 Grid-Row 的 1 和 3 之間，

CSS 寫法為 grid-column-start: 1;grid-column-end: 2;grid-row-start: 1;grid-row-end: 3; 這寫法也是落落長我們可以簡寫為 grid-column:1/2(1 到 2);grid-row:1/3(1 到 3);

再以元素 4(.item4) 為例 Grid-Row 它佔滿了全部所以為 grid-column:2/3;grid-row:1/11; 最後兩個子元素寫完以後如圖 9 所示。



圖 9 定義完元素 1 和元素 4 的預覽頁面

在圖 9 可以很清楚看到我們定義好的元素 1 和元素 4 已經到了自己的定位了。其他元素也自動地不去佔有元素 1 和元素 4 的區塊而自動排列。這也是 Grid 屬性的特性，只要定義好父層，其子層就會按照父層定義好的樣式去自動排列，不需要一個一個去各別定義子層。例如這案例我已經定義好父層寬度為 100%，所以你會看到元素 1、2、3、4 的寬度不用刻意設定它們會自己去佔滿這 100%。接下來我們就再去定義剩下來的幾個子元素，最後如圖 10 所示。

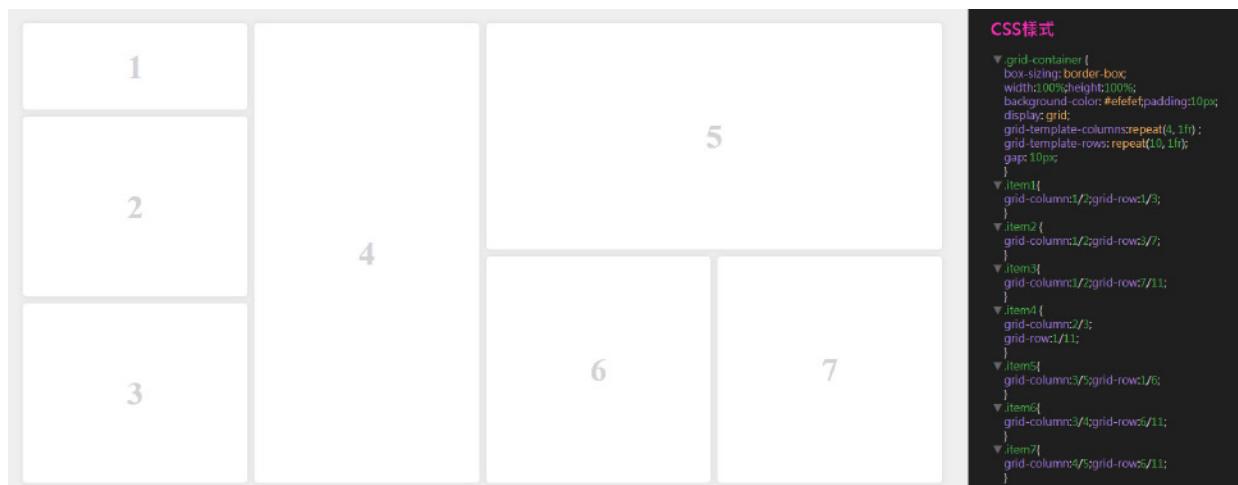


圖 10 定位完其他元素後，儀表板的版面框架已完成

三． 結語

相對於 Grid 屬性大家對 Flex 屬性比較不陌生，因為有些設定兩者都可以共用的關係，也常常被拿來做比較。以通俗的話語來解釋，Grid 這個爸爸控制慾比較強，他自己就決定了小孩們未來的發展，而 Flex 剛好相反，小孩們自主性比較強點。所以雖然兩者都是排版用的屬性，也沒有說這個版只能用 Grid 排出來而不能使用 Flex 屬性來排版。依個人的使用上來說，大方向的布局，譬如主版面的框架、儀表板、攝影機瀏覽畫面等這些用 Grid 屬性來撰寫的話會比較容易達成。細部的編排就可以用 Flex 屬性來撰寫。兩者應該相輔相成讓你的網頁撰寫上達到最佳的效果。近來主流的 UI 框架 Bootstrap 的最新版本也開始導入 CSS-Grid 的概念了，雖然目前最新版本對 Grid 的支援還沒有很完善，但是不久的將來應該也可以使用 Bootstrap 的 Grid 專用 Class 來進行排版。相信在未來使用 Grid 屬性撰寫版面的人也會越來越多了。

參考文獻

- Mdn web docs - CSS 格線布局
https://developer.mozilla.org/zh-TW/docs/Web/CSS/CSS_Grid_Layout
- w3 schools - CSS Grid Layout Module
https://www.w3schools.com/css/css_grid.asp



GIS MAGAZINE

溝通 · 交流 · 分享



線上閱讀平台
magazine.gis.tw

