

Sunbird-dcTrack Frontend

專題編號：109-CSIE-S027

執行期限：108 年第 1 學期至 109 年第 1 學期

指導教授：劉建宏

專題參與人員： 106590010 林孜頤

106590030 陳姍文

106590043 高宗佑

一、摘要

dcTrack 為 Sunbird 開發的 DCIM 軟體，功能包含管理資料中心伺服器、提供機房設施概況瀏覽、電力設備。本專題主要負責 dcTrack 的 UI 與資料的處理、使用者互動、國際化以及視覺化部分的實作。整個專題採用 scrum 流程和 GitHub 來進行開發，每 2 週向 Sunbird 展示實作結果，為確保開發的品質，我們並撰寫相對應的測試案例，觀察程式覆蓋率是否有達標，驗證功能是否正確，測試所使用的框架為 Jasmine，測試工具為 Karma。

關鍵詞：DCIM、angularJS、javaScript、Jasmine

二、緣由與目的

本專題為跟業界合作的專案，參加這個專案的動機是想從中獲得學習與他人共同開發專案，實際去參與業界開發的流程，幫助未來在工作時能有更好的開發流程及實作經驗。

透過此次專題，我們希望達到以下目的，以此讓我們相關技術及能力能夠提升：

- (一) 使用 AnuglarJS 框架完成 Frontend 開發。
- (二) 完成公司給予開發的需求，實現公司想要的功能。
- (三) 針對其撰寫程式設計對應的測試案例並實測，確認邏輯正確。
- (四) 了解業界開發的流程。
- (五) 將開發的程式以報告方式呈現給公司負責人，獲取經驗及建議。
- (六) 學習公司的工程師與我們撰寫

的程式碼有何差異進而去學習。

三、研究範圍

利用 AngularJS 及 javaScript 實作公司所給的需求，例：機櫃內項目的拖拉、跨機櫃內的項目拖拉…等，並使用 Jasmine 撰寫相對應的測試，幫助確認邏輯、行為是否正確。

四、使用技術方法

- (一) 使用 jQuery UI 的 draggable、droppable 確認物件是否可被拖拉或放置。在拉起物件後，由於會受限於 HTML 的層級範圍，造成物件無法順利拖拉到其他機櫃內，因此必須先將物件用 clone 的方式產生另一個不受範圍限制的物件，以確保拖拉範圍不被限制。
- (二) 使用 AngularJS 中的\$validator，將項目的名字欄位在輸入時，會與後端確認是否在此機櫃中已存在相同名字的項目，並設置後續驗證失敗或正確的行為。
- (三) Scrum
Scrum 是一種敏捷軟體開發的方法。藉著此方法，可以讓我們在開發時會有更好的效率。[1]

五、開發流程

我們每個禮拜都會利用 10 或 20 小時的時間來開發這個專題，開發流程則是照著 Scrum 的方式進行。

根據 Scrum 流程，我們會以 2 周為單位稱一個 Sprint 去進行開發，在 Sprint 的一開始先討論這 2 周要做哪

些東西，並每天進行 Daily Scrum，在 Sprint 的最後進行 Review，藉此來達到敏捷開發的效果。

六、系統架構

圖 1 為 dcTrack 前端之系統架構圖，包含 10 個主要的元件，說明如下：

- (一)itemList：顯示全部項目，若項目類別為 cabinet 或 chassis 則會顯示該項目的視圖。
- (二)itemDetail：顯示項目的詳細資料，若項目類別為 cabinet 或 chassis 則會顯示該項目的視圖。
- (三)cabinetElevation：在 itemList 及 itemDetail 顯示 cabinet 視圖及處理視圖相關操作。
- (四)chassisElevation：在 itemList 顯示 chassis 視圖。
- (五)itemBlade：在 itemDetail 顯示 chassis 視圖。
- (六)chassis-slot-layout：顯示 chassis 視圖及處理視圖相關操作。
- (七)jsChannelService：能發送進行跳轉至其他頁面的 API。
- (八)elevationService：儲存 cabinetElevation 的資料、相關的運算邏輯及 API。
- (九)chassisElevationService：儲存 chassis-slot-layout 的資料、相關的運算邏輯及 API。
- (十)itemDataService：儲存全部項目的資料。能獲取全部資料給 itemList 或抓取單筆資料給 itemDetail、cabinetElevation 及 chassis-slot-layout 使用。

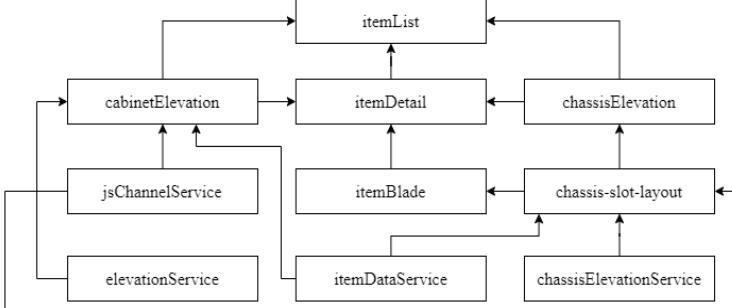


圖 1. dcTrack 前端系統架構圖

七、實作結果

依照公司所給的需求進行開發，此部份我們舉其中一份需求做為說明實作結果。

需求：可拖拉機櫃內的項目、可拖拉項目不同機櫃放置

(一) 拖拉時：

1. 可放置：目標機櫃的項目位置底下顯示綠色背景，拖拉項目上的箭頭為綠色。
2. 不可放置：目標機櫃的項目位置底下顯示紅色背景，拖拉項目上的箭頭為紅色。

(二) 目標機櫃已有重複的項目

若目標機櫃內項目與拖拉的項目名字重複，跳出相同重複名字的警告，提醒使用者須改名才可完成拖拉。

(三) 設計測試案例

針對撰寫的函式進行測試，例如：拖拉時，需判斷放下的目標位置是否已有項目、是否有足夠的高度給拖拉的項目存放。根據此前提去設計相對應的測試案例。

八、結論

AngularJS 的雙向資料繫結在傳遞資料上較輕鬆，不必像 jQuery 需要先獲取該元素在做處理，並可直接透過綁定的方式，針對各個資料做相對應的處理。

測試上，Jasmine 的 spy 是一個很強大的功能也很好上手，只要事先將要攔截的函式確認好，便可以在開始測試時去決定是否忽略掉或是回傳特定的值，將需要連接後端的地方利用 spy 去攔截去回傳固定的資料，避免對後端的資料進行更動。

九、參考文獻

- [1] "Scrum,"
[https://en.wikipedia.org/wiki/Scrum_\(software_development\)](https://en.wikipedia.org/wiki/Scrum_(software_development))