**一、技術主題**

**1. AngularJS**

**概要**

**AngularJS**是一款[開源](https://zh.wikipedia.org/wiki/%E5%BC%80%E6%BA%90%E8%BD%AF%E4%BB%B6)[JavaScript](https://zh.wikipedia.org/wiki/JavaScript)函式庫，由[Google](https://zh.wikipedia.org/wiki/Google)維護，用來協助[單一頁面應用程式](https://zh.wikipedia.org/wiki/%E5%96%AE%E4%B8%80%E9%A0%81%E9%9D%A2%E6%87%89%E7%94%A8%E7%A8%8B%E5%BC%8F)運行的。它的目標是透過[MVC模式](https://zh.wikipedia.org/wiki/MVC%E6%A8%A1%E5%BC%8F)（MVC）功能增強基於瀏覽器的應用，使開發和測試變得更加容易。

**甚麼是 AngularJS？**

AngularJS 是一個使用 HTML、JavaScript 和 CSS 來建立 Web 應用程式的框架（Framework），其由 Google 所創建，用來協助單一頁面應用程式的運行。

它可以擴展應用程式中的 HTML 詞彙，從而在 Web 應用程式中使用 HTML 聲明動態內容，並擴展 HTML 的語法，以便清晰、簡潔地表示應用程式中的組件，並允許將標準的 HTML 作為你的模板語言（Template Language）。

函式庫讀取包含附加自定義（標籤屬性）的 HTML，遵從這些自定義屬性中的指令，可將頁面中的輸入或輸出與 JavaScript 的變量表示模型綁定起來。這些 JavaScript 變量的值可以手動設置，或從靜態／動態的 JSON 資源中獲取。

AngularJS 可以通過雙向資料綁定（Two-way Data Binding），自動從擁有 JavaScript 對象（模型）的 UI（視圖）中同步數據。AngularJS 提供了強大的資料綁定、依賴注入（Dependency Injection）...等，使你可以測試和維護 Webapp。  
1、 有別於傳統的函式庫或框架，AngularJS 擁有屬於自己的一套：

* 直接以 HTML（DOM）當成 Template 來使用
* 將 HTML 變成可複用的元件
* 資料綁定在 {{}} 裡面
* 支援表單操作與表單驗證
* 將程式碼（Code-behind）綁定在 DOM 元素上

2、 AngularJS 完整的前端解決方案：

* 適合用於 CRUE 類型的網站專案／後台（資料連結、基本範本指令、表單驗證...）
* 不適合用於需要大量 DOM 操作的網站
* 完整的單元測試與 E2E（End to End）測試架構

3、 AngularJS 表達式（Angular Expressions）：

* 使用「JavaScript」語法表達模型的程式片段
* 通常用來輸出 Model 的資料或執行控制器的函式
* 範例：{{9\*9}}、{{sum()}}、{{'hello'}}

**AngularJS 框架介紹：**

一個由 Google 打造的前端 JavaScript 框架，與其他 JS 框架最大的不同在於，它直接延伸現有的 HTML 架構，透過宣告式語法（Directives Syntax）直接賦予 HTML 額外的能力，並通過雙向的資料綁定來適應動態內容。雙向資料綁定允許模型和視圖之間自動同步。因此，AngularJS 提升了可測試性，也讓 Web 應用程式在元件化的過程變得極其簡潔有力。

1、 AngularJS 的強項：

* 「關注點分離」：

控制器（Controllers）與檢視（Views）之間切割的非常乾淨，再搭配模組（Module）與依賴性注入相關實作。將商業邏輯從 HTML 中抽離，簡化複雜度（從 DOM 操作中分離思考），以及將網站的前／後端乾淨分離，並簡化後端開發。

* 「以習慣取代配置」：

AngularJS 所設計的宣告式語法直接延伸 HTML 的能力，讓許多 AngularJS 自訂的 HTML 屬性能自然地融入其中，並賦予其意義，而這就是所謂的習慣（Convention）。這是大多數 JS 框架不敢做的嘗試，也因為如此，這樣的設計觀點並不是所有人都能接受。

2、 在 AngularJS 裡，當瀏覽器將 HTML 與這些 AngularJS 自訂的語法解析成 DOM 物件之後，AngularJS 會直接將原生的 DOM 物件當作網頁片段的範本，然後直接以 DOM 物件（原生的 JavaScript 物件）進行操作。這大幅減少了轉換型別的成本，但相對在範本操作的過程，其效能也比其他框架高出許多，這也就是 AngularJS 所謂的 DOM Templates 特性。

3、 AngularJS 所提供的雙向資料綁定特性，可以說是最實用的功能之一，雖然這個特性並不是 AngularJS 獨有（其他 JS 框架也大多都有此特性），不過對於沒有接觸過其他框架的人來說，這確實是一個非常有趣的功能之一。

**AngularJS 的設計與目標：**

AngularJS 信奉的是，當組建視圖（UI）與寫軟件邏輯同時進行時，聲明式的代碼會比命令式的代碼來的好，儘管命令式的代碼非常適合用來表達業務邏輯。

AngularJS 透過 MVC（Model-View-Controller）模式功能增強瀏覽器的應用，使開發和測試變得容易，並通過依賴注入為客戶端的 Web 應用提供了傳統服務端的服務，例如獨立於視圖的控制。因此，後端減少了許多負擔，產生了更簡單的 Web 應用。

1、 AngularJS 的理念：

* 將應用邏輯與對 DOM 的操作分開處理（提高代碼的可測試性）
* 應用程序的測試與應用程序的編寫一樣重要（代碼的構成方式對測試的難度有巨大的影響）
* 將應用程序的客戶端與服務器端分開（這允許客戶端和服務器端的開發可以齊頭並進，並且讓雙方的複用成為可能）
* 指導開發者完成建立應用程序的整個歷程（從用戶界面的設計，到編寫業務邏輯，再到測試）
* 化繁為簡

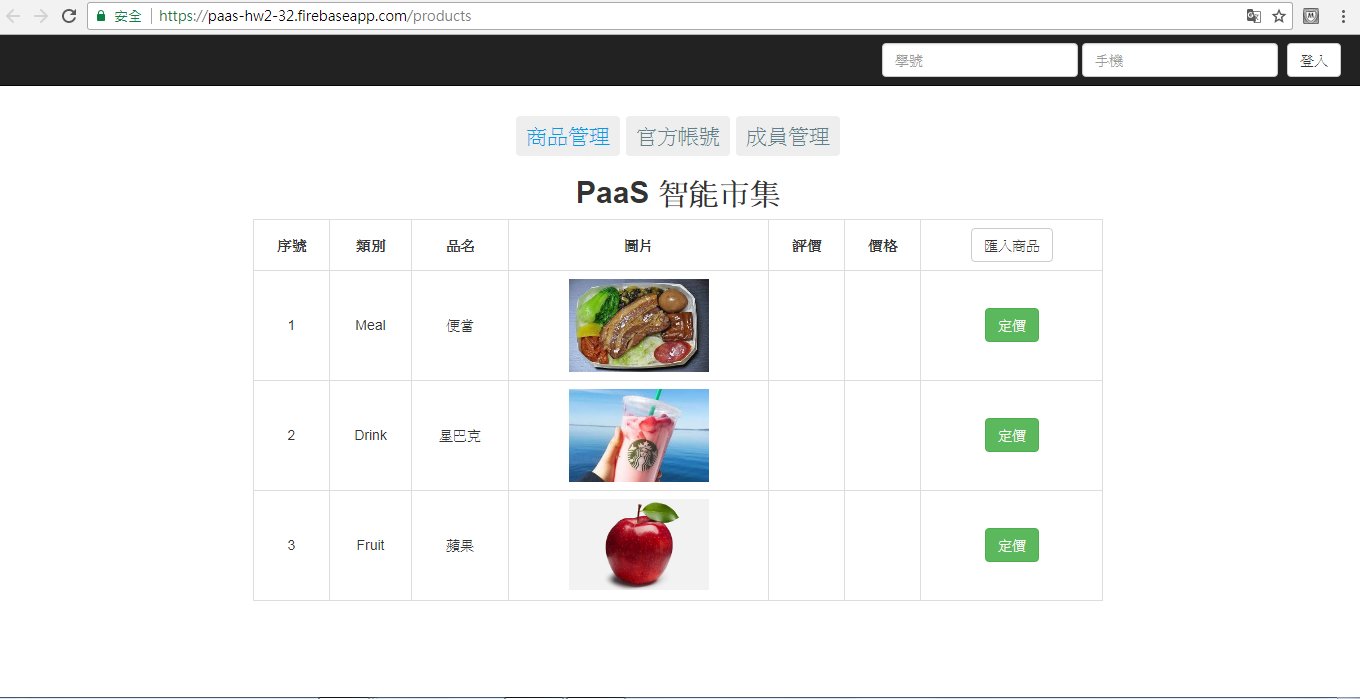
2、 可以免除以下痛苦：

* 註冊大量的 Callback 函式（回調容易打亂代碼的可讀性）
* 大量操作 DOM 物件（DOM 很笨重且容易出錯）
* 從 UI 中擷取與資料處理（非具體的業務邏輯和業務細節）
* 需要撰寫大量的初始化程式才能使用框架

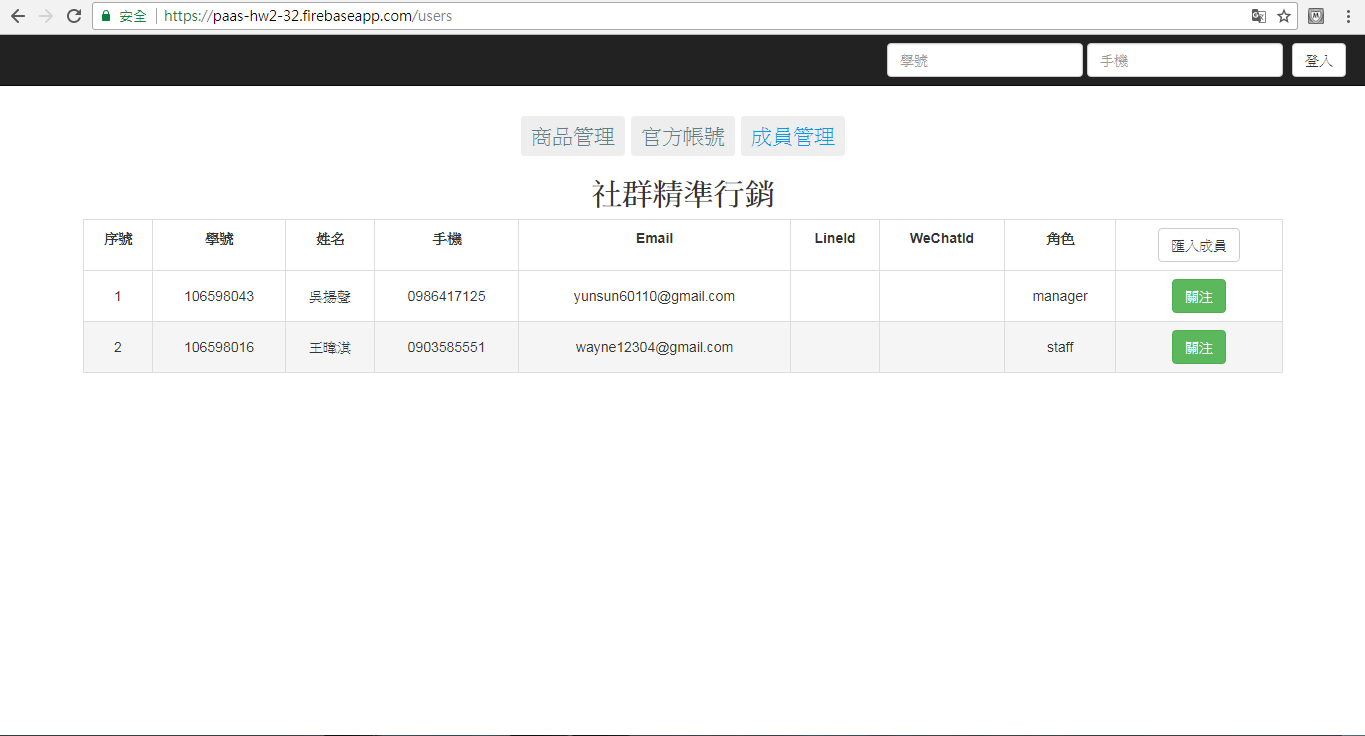
aNGULAR的FEATURES & BENEFITS

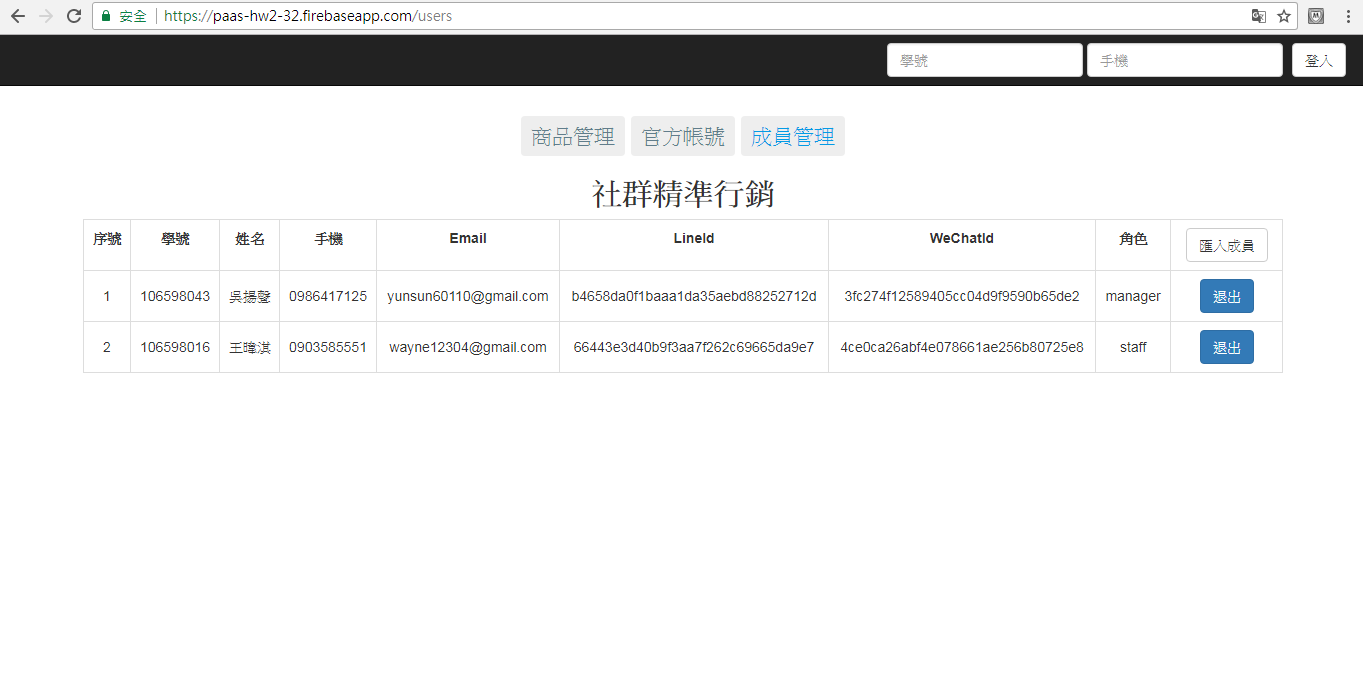
* CROSS PLATFORM
* SPEED AND PERFORMANCE
* PRODUCTIVITY
* FULL DEVELOPMENT STORY

**二、作業輸入、輸出畫面**









**三、參考資料**

1. AngularJS --wiki

<https://zh.wikipedia.org/wiki/AngularJS>

1. 技術專欄--AngularJS簡介

<https://www.openfoundry.org/tw/tech-column/9208-angularjs-1>

1. Angular官網

<https://angular.io/features>

**四、學習心得**

教授好，這禮拜因為去考汽車駕照路考，所以沒有去聽老師的課，也知道老師討厭，一半才進來聽的學生。

還好有一起上的同學有去聽課，他們上完課的感想，都說上老師的課，收穫滿滿，好難得看到那麼充滿熱血、認真上課的老師，雖然上的東西，實在不太容易。

這次的作業，讓我理解到了，Angular的強大之處，也讓我理解到了，凡是靠自己實在太累了，畢竟學也學不完，何不嘗就像老師說的，要爬陽明山，先搭紅5到文化大學的道理。