學習心得

Bootstrap我是在課程才第一次知道有這麼一個網頁套件，這我也才發現原來市面上的網頁有不少都是使用此套件作成的，使用Bootstrap的好處就在於有很多的網頁特效它都幫我們作好了，我們只需要在想要使用的區塊套上相應的Class就可以實現，而網路也有不少現有的模板可供使用，對不擅長設計版面的人來說可以說是相當方便。

Git這也是我在課程中才知道的軟體，雖然老師於課程中只有教我們一些Git的基本概念以及如何使用，而在當時我並沒有充分了解Git的好處和使用方法，但在之後的專題製作我實際使用Git進行版本控制之後發現，它不但能讓自己掌控整個專案的進程，也能在更改程式後出現錯誤時去與之前的版本作比對，了解是哪個部分出問題，加快錯誤修正的時間。

對於PHP我只有之前在學校時社團學長出作業的時候有接觸過，當時我是與同學一起撰寫一個圖書管理的網站，那時只想著把網站寫出來就好，並沒有深入了PHP的特性，在這次課程中我才知道PHP程式與其他程式有哪些不同，其中我個人覺得最大的差異之一就是在變數的宣告上了，之前我在撰寫JAVA的時候，在使用變數之前都必須先進行宣告，而且必須規定其型態之後則無法進行變更，但在PHP不一樣你可以直接使用未宣告的變數，而變數的型態會隨著變數的值進行改變，這讓我在起初並不是很習慣，只不過在使用一段時間之後就覺得這其實也挺方便的。

雖然我之前是學JAVA的，在學校也學過物件導向的課程，但在這次的課程中我才知道要如何去撰寫一個物件導向的程式，此外我也透過這個課程知道物件導向在團隊的程式撰寫中所能帶來的好處，透過將程式物件化可以在我們之後要重複使用該程式時更加方便，並且降低其他人使用該物件時發生錯誤或與原本撰寫的程式發生衝突的情況產生。

MVC是一個物件導向的軟體架構，在課程中老師帶著我們作了一遍簡單的MVC架構，其實當時並不是很了解MVC的運作方式，但多虧老師在上課時所教導的MVC的概念，對我在之後專題製作的MVC架構的實作上有很大幫助。

在這一個月的課程中，我接觸了需多之前所不知道的程式撰寫的概念，對我來說這次課程雖然讓我在程式撰寫的功力上有不少進步，但最大的收穫是在程式撰寫的觀念，以及對其他軟體與技術認識，畢竟要學習寫程式是可以自己在網路上學習，但觀念在透過老師的教導之後可以讓我們更加清楚明白。

在課程結束之後的專題製作，起先我原本打算在一開始就使用MVC的架構去製作，但在嘗試了一兩次之後發現並不是很順利，所以我決定先進行網頁功能的撰寫，在完成初步功能之後，我才再次挑戰使用MVC架構，這次我重新釐清MVC的概念與運作方式，成功在預期的時間內把自己的網站更改為MVC架構。在專題製作一開始我並沒有很頻繁的使用Git，是在主管的提醒之後才開始提高自己Git的提交次數，並試著讓自己養成提交Git的習慣。

在課程之後我們首先接到的是研二所出的作業，第一個是找出陣列中最大的相鄰區域，我選擇使用遞迴的方式去撰寫，並在撰寫完成之後嘗試將其物件化和提高可用性，而在與其他同事討論之後得知有其他的解法，了解其他解法的概念與實作方法。第二個作業是撰寫一個活動報名系統，這次作業的重點是當面對多人進行資料庫存取時，要如何確保資料的正確，這次我所嘗試的方法是鎖資料庫，只不過我是在之後研三的作業時，才清楚了解它的運作方式及所會帶來的影響。

研三所出的作業是撰寫一個簡易的銀行系統，主要的課題除了多人存取資料庫之外，還有的就是程式撰寫的格式和Git的使用狀況，資料庫的問題我除了去找解決的方法之外，我也去了解各個方法的所會產生的影響，及其原理和專有名詞，像是有悲觀鎖和樂觀鎖，以及除了鎖資料庫之外還有版本號控制和原子操作這兩個方法去確保資料正確。另外，主管在這次作業也讓我們明白在團隊撰寫程式的情況之下，統一的程式格式和一目了然的commit有多重要，這些不但可以幫助其他人更加容易看懂自己的程式碼，也能讓自己在未來回頭查看自己程式時能更清楚明白。

研一的作業跟之前的處所出的作業很不一樣，一開始他要求我們去研究簡報技巧，之後則出了兩個題目讓我們二選一，我選擇的是撰寫一個程式去抓取網頁的資料，並寫出一個提供資料庫資料的API，