



杜雨涵

系 別 / 年 級：資訊管理學系/四年級

學 校 輔 導 教 師：辛信興老師

實 習 機 構：國立臺東大學圖書資訊館

實 習 部 門：系統發展組

實習機構督導人員：陳怡如

國立臺東大學圖書資訊館

印象最深刻的經驗

印象最深刻的經驗莫過於3/24的Code Review了，在那次的Code Review中看了我做的匯出PDF功能，我用的方式是做一個PDF的模板，再用程式讀入PDF，並將使用者輸入的資訊替換掉並產出新的PDF，雖然功能是有達到使用者的需求，但匯出的PDF排版會跑掉，以及使用者希望PDF模板的格式是可以自行更改的，想要程式可以讀入Word，再替換掉使用者輸入的資訊，再匯出PDF。由於使用者更改了需求，導致我原先花了很多時間寫的程式要整個打掉重寫，當時是我第一次遇到這樣的狀況所以讓我很崩潰，因為這意味著我又要花不少的時間重新寫這個功能，而且還要用更複雜的方式來開發，我可能又需要更多時間去研究，再加上我還有其他的進度需要趕，所以我當時直接想要終止學分，我覺得壓力真的很大，那陣子我每天都是哭著回家的，每天都在自我懷疑我是不是真的很笨，抗壓性又很低，不適合這個行業。但後來經過多人開導以及跟主管協調過進度，才解決了這些問題，另外我也意識到在系統開發中，需求的變更是一件很常見的事情，就算是很資深的工程師遇到這樣的問題也能去接受並去解決它，所以我後來也就漸漸適應了會更改需求這件事，接受這一點，並做好應對準備，在要開發功能前也會更謹慎的與使用者確認需求，愈謹慎的進行溝通和討論愈能降低更改需求的機率。適當地處理需求變更，可以幫助我在開發過程中成長，提高技能，並創建出更好的程式。

經過這次的經驗我學到了要保持冷靜的態度面對問題以及在遇到問題時要適當地提出，讓團隊或主管知道，這樣才能有人幫助你解決問題，如果都不提出的話反而會讓自己的壓力過大，最重要的是不要放棄，持續學習和成長，並且相信自己能夠解決問題。

最大的自我突破

在專業技術方面，我認為我成長了非常多，在實習以前，我對於我寫的程式的要求是可以編譯，可以執行，可以跑的出來就好，但是在實習的時候，被要求要使用MVC的架構，也就是在專案裡要有Controller去處理使用者互動、使用Model並且在最後選擇可以轉譯要顯示UI的View，去開發就學貸款的專案，並且搭配Dependency Injection 的設計模式，這個設計模式是的用法主要是需要先有一個Interface去定義方法，然後透過Class去繼承這個Interface去實作它所定義方法，最後再透過在Controller宣告、註冊這個Interface，並去使用裡面的方法。雖然我在實習前有大概接觸過這些設計模式，但也只是瞭解這些設計模式的用法以及為什麼要這樣用，沒有特別去使用這些設計模式去寫程式，但在實習後被要求要用這些用法去開發專案，用了這些

設計模式去開發專案後，我的專案變得更符合物件導向的原則，不再像以前一樣，所有的程式碼像是麵條雜亂交錯，完全無法看出程式脈絡，雖然專案可以跑得動，但如果之後需要給其他人接手去優化，思考下一步要如何擴充功能的時候，其他人可能看不懂我的程式碼的邏輯，可能會造成他人的麻煩，以及之後維護上的困難。除此之外，由於以前都是在本機上開發一些小型的專案，所以不太需要用到資料庫，再加上製作專題的時候我也不是負責資料庫的部分，因此我的SQL語法以及有關資料庫的能力非常弱，連基本的CRUD是什麼都不知道，但實習後開發就學貸款這種大型的系統，會需要使用到很多資料，所以我不得不強迫自己去精進SQL以及Dapper語法，從開始實習到現在我資料庫的能力成長非常多，現在在程式裡寫搭配Dapper語法的CRUD方法對我來說已經不是什麼大問題了。

在心理素質方面，因為在開發過程中常常會面臨各種挑戰和壓力，像在上述有提到我有一陣子壓力真的很大，但現在我學會了如何調適自己，首先我會保持冷靜，面對壓力時，保持冷靜是非常重要的。可以透過深呼吸、暫停幾秒鐘，讓自己冷靜下來。再來就是我會先試著先分析問題，找出問題的根本原因，透過分析，可以讓自己更有條理地思考，並找出解決問題的方法。如果有時候是因為進度的壓力，我會把進度條列出來，並訂定明確的目標，並把注意力集中在達成目標上。透過訂定目標，可以讓自己更有方向感和動力，並減輕壓力。我認為我在心理素質方面成長了不少，從一開始的無所適從到現在能夠適當的調適自己，在實習期間我不只使學到許多新的技術和工具，提高自己的技術能力，也學到自己獨立思考和解決問題的能力，以及如何幫助自己調整狀態和心理，以應對壓力、焦慮、情緒波動，在實習期間，我也學會了如何與同事進行有效的溝通和合作，並且在不斷的實踐中養成了良好的工作習慣和態度。這些成長讓我更加自信和堅定地走向自己的職業道路，並且希望能夠在未來的工作中持續發揮這些能力，不斷學習和進步。

總體來說，實習是一個非常寶貴的機會，能夠讓我在實際的工作環境中學習新知識和技能，同時也為未來的職業生涯打下良好的基礎。希望我能夠繼續保持積極的態度和努力，並在實習中獲得更多的收穫和成長。

檢討與建議

在正式開始實習的前十天才傳郵件通知實習所用的工具是.net core，並且希望我們在進組前能自行學習並了解，雖然有提前通知，但提前十天通知並且要先自行學習感覺有點太趕，更何況那十天還是寒假期間，我早就有安排打工及出遊的行程，因此我只能用非常零碎的時間去學.net core，我認為可以在面試或錄取的時候就告知實習會用的是什麼開發工具，當時面試只有說會是用C#開發，所以在實習前我只有針對C#的部份去學，當我在實習前十天接到上班後會用.net core、MVC架構、DI設計，並且要先自行學習的消息後，真的讓我很措手不及。雖然我在過去學過用Java的Spring Boot開發，已有相關概念的開發經驗，但跨框架的開發中仍有許多特性和語法需要熟悉，因此在熟悉框架的階段也是吃足苦頭。

另外，對於工程師這個職業和我所想的其實差蠻大的，一般我們想像工程師就是上班一直打code，跟人沒什麼互動，後來發現完全不是，實習後我才發現人與人的協作比原本想像的多很多。在開發系統時，除了跟User溝通需求，內部協作也需要溝通及討論等，得知彼此的開發狀態，知道需不需要協助或何時能夠求助對方。此外，組內還有各式各樣的會議，每週三有組內會議，報告組內進度，週五有Code Review，都需要精準的資訊傳遞能力才能確保開發順暢運作。

以我目前實習的這段日子裡學到的東西來說，我認為超出我的期待太多了，當初在面試的時候談到的工作內容主要是以維護校內現有的系統並且是以學習為主，所以我原本還擔心在實習的時候沒辦法學到什麼東西，但隨著2/20開始實習的時候得知我以及另一位實習生要共同開發就學貸款的系統，以及有一位外聘的業界工程師指導下，我在實習的時候學的東西越來越多，完全超乎我的預期。尤其是能夠有外聘的業界工程師指導也是很棒的體驗，因為我可以從他豐富的經驗中學習到很多寶貴的知識和技能，和他合作也可以讓我更深入地了解業界的最新趨勢和技術，以及學習實際的工作方式和流程。這些都是從課本和教學中難以獲得的經驗，能夠為我日後的職業生涯打下堅實的基礎。

同時，與同事和導師良好的互動和溝通也是實習中不可或缺的部分。透過與團隊合作，我可以學習如何有效地溝通和協作，以及如何解決問題和應對挑戰。這些都是職業生涯中非常重要的技能，能夠幫助我在未來的工作中更好地與他人合作和建立良好的工作關係。不僅能夠學習新技能和知識，這次實習也讓我體驗到了真正的團隊合作。和同事們一起開發新系統需要良好的溝通和協調能力，同時也需要承擔一定的責任和壓力。通過這次實習，我不僅可以學習新技能，還可以進一步提高我團隊合作能力和職業素養，這些都是在未來職場上非常寶貴的資產。

此外，這次實習也讓您更深入地了解了我在公司或組織的運作方式和文化。通過與同事們的交流和合作，可以更好地了解公司的目標和價值觀，並且在實踐中體驗到這些價值觀的重要性。這些都是在未來職業生涯中需要的基本素養和技能，並且能夠幫助我成為一位出色的專業人士。

對學弟妹的建議

在實習的這段期間我深深的認知到，如果資管系的學生未來要往工程師這個行業去發展，在學校所學的東西在工作的時候完全不夠用，以資管系來說，系上的課大多都是管理的課，程式相關的課程占比不多，而且一個學期也學不多，如果未來真的有意要往程式這方面去發展，我建議除了要修系上開的課之外，還要自己額外花時間去研習程式語言，學習的程式語言種類不用多，但一定要專精一種，這樣將來碰到其他程式語言的時候才會比較後上手。

以我的經驗來說，我很早就決定要朝程式這方面去發展，所以我在大學二年級下學期時就在外面進修，當時我是以學校教過的程式語言Java再延伸下去學，雖然大學一年級下學期修的物件導向程式語言是教Java，但當時是我第一次接觸物件導向的概念，我自認當時我學的不是很好，物件導向的觀念不是很熟悉，所以我一開始是以搞懂物件導向為主。我認為物件導向是一個很重要的觀念，現今主流的程式語言幾乎都是物件導向程式語言像Python、C++、Java、C#等等都是物件導向程式語言，所以我認為在學程式語言之前，先搞懂物件導向的觀念很重要，基本概念像功能或方法、類別、建構子、物件、類別物件變數等等可以先熟悉，這些基本概念都熟悉後可以再去學進階的像繼承、抽象、封裝、多型。這些概念在實際開發的過程中真的常常都會使用到，而且懂了這些觀念之後，從原本我熟悉的Java跳脫到其他程式語言，像C#。雖然我沒有學過C#，但我很容易就能看懂它的語法，只差有些用法不同像Java的method在C#是Function，但其實這些很快就能調適過來，所以我建議學程式語言之前首先要先懂物件導向的觀念。

除了物件導向的觀念之外，還有一些常用的設計模式（design pattern），像Singleton 單例模式、DAO 設計模式、Factory 設計模式，以及實習時用到的Dependency Injection，我學的這些其實都是學校不會特別教到，但是在實務上常常會用到的，尤其是設計模式，我覺得練習寫程式先不用照著設計模式去開發沒關係，但一定要去理解、思考這些設計模式的意義及概念，因為設計模式是在開發上非常重要的參考工具，了解設計模式後未來在接觸到相似的軟體架構或是程式碼時，可以快速的理解別人編寫程式碼的概念，我認為設計模式絕對是一門主修科目，你可以不完全使用，但卻不能不知道。

再來就是工程師這個行業，它是一個快速變化的領域。要跟上行業的變化，需要不斷學習和掌握新的技術，還需要自學的能力，開發一些新功能的時候，可能你從沒使用過某項技術，但需要用到它，這種時候只能自己去學習，像我在開發的時候就遇到過這種難題，第一個是需要在下拉式選單增加搜尋的功能，第二個是要製作匯出PDF的功能，這兩個問題都需要用到第三方的套件，雖然有套件給我使用，但如何去使用這些套件解決問題就要自己研究，可能你是第一次用這些套件，那你就得自己去讀懂說明文件並使用，很多時候不會有人在你旁邊教導你，你只能夠靠自己去想辦法解決問題，這就需要有自學的能力。

最後給學弟妹的建議就是要勇於跨出第一步，如果你對於工程師、寫程式這個行業真的很有興趣的話，與其躊躇不前，不如去找相關書籍或課程，學下去就對了，程式設計是一門技能，需要不斷練習才能掌握，越早開始學對你有益無害，你可以嘗試先從編寫一些簡單的程式開始，這有助於你瞭解基本語法和概念，在學習程式設計的過程中，犯錯是很常見的，不要害怕犯錯。相反，從錯誤中學習並改進。每個工程師都曾經犯過錯，重要的是如何從錯誤中學習並進步。

```
<div class="modal fade" id="delModal" tabindex="-1" aria-hidden="true">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="exampleModalLabel">提示信息</h5>
        <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
      </div>
      <div class="modal-body">
        請認是否刪除？
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" id="confirmNo">取消</button>
        <button type="button" class="btn btn-primary" id="confirmYes">確認</button>
      </div>
    </div>
  </div>
</div>

<script src="/js/ApplyDateSetting/index.js" async="async" asp-append-version="true"></script>
</script>
```

