

國立臺灣體育運動大學

教學實踐研究

技術報告

「以大學體育相關科系學生適性發展為導向融入多元教學法之資訊技術實務課程教學」

作者：許育嘉

所屬科系：國立臺灣體育運動大學
運動資訊與傳播學系

中華民國 112 年 2 月 15 日

摘要

本技術報告係彙整送審人過去四個年度進行教育部教學實踐研究計畫之成果，闡述針對主修運動、休憩領域的學生，以融入運動場景為核心理念並結合多元化教學方式，所進行的一系列資訊技術實務課程的創新與改革。對於學習興趣低落、學習成效不佳等問題，提出創新教學方式並進行實證。實踐研究的課程為程式設計、網頁設計、資料庫、資料科學四門課程，採用的多元教學方式分別有：講述法、程序部署結果改進 (Approach, Deployed, Result, Improvement, ADRI) 教學模式、成對編程、自動程式評測平台、合作學習、做中學、體驗式學習、情境式學習、逆向設計教學順序、專題導向式學習、遊戲式角色扮演、漸進釋放責任模式、專業服務學習等概念。

技術報告的內容包含五個部分。首先第一章「教學實踐研究動機與主題」，為送審人分析非資訊相關科系學生學習資訊技術實務課程之教學現場問題後，提出以運動場景串接各個課程所構成之研究主題架構。第二章「相關文獻探討」，為基於特定課程教學文獻回顧所進行之擴增與收斂過程，在探索各課程之教學案例文獻後，從多種教學理論中收斂出適合的教學方式。第三章「教學設計與研究方法」說明了各課程採用的多元教學方式，包括：採用 ADRI 教學模式及自動評測平台之程式設計教學；體驗式、遊戲化專題導向、服務學習之網頁設計教學；棒球主題之資料庫教學；漸進釋放責任模式之資料科學教學，也展示了融入運動場景的課程內容設計，以及在符合教學目標及確保學習成效下進行的行動研究。學生學習興趣及學習成效提升，以所蒐集到的學習體驗和自我效能量表進行量化分析，並將個別訪談紀錄進行質性分析，結果依課程分別呈現於第四章「研究成果與學生學習成效」。最後，第五章「方法或應用之創新與貢獻」，分享了送審人發展多元創新教學模式之雙鑽石流程，與跨課程整合之特色。以及這一系列教學實踐研究對高等教育改革研究社群之貢獻，及對培育學生之影響。

目錄

摘要.....	I
目錄.....	II
1. 教學實踐研究動機與主題.....	1
1.1 教學現場之問題及其重要性.....	1
1.2 研究動機.....	2
1.3 研究主題.....	10
2. 相關文獻探討.....	14
2.1 程式設計教學相關文獻.....	14
2.2 網頁設計教學相關文獻.....	17
2.3 資料庫課程教學相關文獻.....	20
2.4 資料科學教學相關文獻.....	23
3. 教學設計與研究方法.....	27
3.1 運用 ADRI 教學模式結合成對編程與線上評測之程式設計課程.....	27
3.2 融入體驗式學習與遊戲化專題導向學習之網頁設計課程.....	33
3.3 從棒球學資料庫管理.....	38
3.4 融入運動分析實務問題之資料科學課程.....	43
3.5 研究方法與工具.....	50
4. 研究成果與學生學習成效.....	53
4.1 程式設計課程(108年教育部教學實踐研究計畫).....	53
4.2 網頁設計課程(111年教育部教學實踐研究計畫).....	58
4.3 資料庫管理課程(110年教育部教學實踐研究計畫).....	63
4.4 資料科學課程(109年教育部教學實踐研究計畫).....	66
5. 方法或應用之創新與貢獻.....	27
5.1 多元創新教學設計的行動流程—雙鑽石模型.....	73
5.2 貢獻.....	75
參考文獻.....	77

1. 教學實踐研究動機與主題

1.1 教學現場之問題及其重要性

送審人於所任教之大學體育相關科系中，教授資訊技術領域課程多年，包含了一系列的程式設計、資料庫管理、網頁設計、資料科學、大數據分析、網頁程式設計等課程，培養非資訊相關科系的學生能具備基礎之運算思維與資訊技術實務操作的能力，進而到如何運用這些資訊技術的素養，發揮到所屬的學科領域及相關產業，解決特定領域的問題。

然而對於非資訊科系的學生而言，學習這些源自於計算機科學學門的資訊技術相關課程是相當大的挑戰，且存在著嚴重數位落差的問題。依送審人過去歷年於第一堂課進行之例行調查，學生在大學入學之前所接觸到的資訊技術，不論是 Word、Excel、Access 之 Office 應用軟體，或是 Visual Basic、JAVA、Python、C/C++、Scratch、App Inventor 這些程式語言，各個學生接觸的機會不一，存在極大的數位落差和學生學習落差。以最基礎的程式設計為例，課堂中見到許多學習落差的樣態：有英語、數學不好，不懂程式語法的；有不知整個檔案結構找不到存檔位置的；有不知道如何 Debug 的；有回家沒安裝程式開發環境的；有程式考試是用看、用背程式碼的，沒動手寫。

學生在大一基礎程式設計與運算思維之能力養成不易，進而也影響了後續大學高年級進階課程之學習。以送審人針對大二、大三接續開設的網頁設計、資料庫管理、資料科學等一系列的課程而言，這些資訊技術課程的教育目標，不是要培養這些非資訊科系的學生個個成為程式設計師、資料庫管理師、網頁工程師、資料科學家，而是要讓學生能將這些資訊技術應用於運動產業中，解決運動產業的問題。以當今產業工作環境而言，各行各業都無可避免地需利用資訊科技來解決問題，積極地善用官方網站、社群媒體、行動 App 來創造商機，但是該如何導入應用、佈建、實施這些系統，對於非資訊科系的學生大都避之唯恐不及，往往都認為以後只要找的到委外的資訊廠商就能解決問題，殊不知某些特定領域的資訊應用，跨領域人才會是影響成敗的關鍵。

程式設計對於大多數非資訊科系的學生來說，都認為學習困難。然而對於一門課中涉及了 Html、CSS、jQuery、Bootstrap... 等，這麼多的資訊技術詞彙的網頁設計課程，更容易產生學習焦慮。MS SQL/MySQL 這類常見的資料庫，也幾乎沒有學生在修課前有使用過的經驗。當基礎資訊技術課程的學習成效不甚理想時，間接也影響到一些進階應用課程的學習動機。學生在選課時一看到課名有「程式」、「系統」等資訊課程時，常常顯示出很難懂、很難通過、很無聊、以後用不到，這類的刻板印象。甚至也有學生反映說要建網站時“用臉書不就好了”、“上 Google/Weebly/Wix 的雲端申請一個免費網站，是不是就不用學這麼多”，說出了在現今各式淺碟式工具逐漸普及下，對於專業的知識體學習動機不足的心裡話。

目前已有許多大學系所本位朝向多元化的發展，強調跨領域、跨學科整合應

用的能力，也發展出一些大學不分系的學位學程。以送審人服務的科系而言，要求學生要具備體育專業素養，主力在發展新聞傳播，同時也需兼備資訊技術實務應用的能力，目的是要能讓學生將資訊技術與傳播結合，應用於運動產業中。當學生有多元選課的機會時，一旦大一修課即遭遇程式設計的學習挫折時，會在未來學習路徑上做出選擇。大部分學生會想辦法規避學習挫折高的課程，而另選擇容易得到高學習成就的課程。而程式設計學習成效不佳，會影響到後續銜接課程的學習成效，舉凡資料庫管理、資料科學、網頁設計等，且會影響到學生在畢業專題作品上的成果。程式設計、網頁設計、資料庫、資料科學等這類資訊技術實務的課程，在教學實務研究文獻上，相對於其他課程，一致被認定學習難度較高，如何將課程重新設計，引導學生願意花更多時間學好，而不放棄，對教師而言是一大挑戰。

本研究就前述教學現場的問題，從教材內容、教學目標、教學方法來進一步思考，依據學生運動主修背景來進行課程變革。以所任教的「程式設計」、「網頁設計」、「資料庫管理」、「資料科學」，四門課分四個年度逐一進行一系列課程的教學實踐研究。

1.2 研究動機

基於前述思維，送審人即以在大學一年級到三年級所開設四門一系列且具連貫性的資訊技術實務課程—「程式設計」、「網頁設計」、「資料庫管理」、「資料科學」為四階段教學實踐研究實施場域，針對不同課程主題內容，融合運用不同教學方法，藉以提升學生學習興趣、學習成效，達成有效教學目的。四門課之教學實踐研究，所對應的研究動機分述如下。

1.2.1 運算思維之建立—程式設計

程式設計的能力，就如同過去所謂數學為科學之母般，是發展創新行動 App 開發、新型態的網路社群行銷、大數據分析、物聯網、人工智慧等於各領域應用之未來發展重要關鍵。在順應世界程式教育向下扎根的發展潮流之下，並深植發展人工智慧的基礎實力，教育部於 108 學年度十二年國民基本教育課程綱要，開始將「程式設計」納入科技領域成為必修課程。而在高等教育部分，則於 106 年度起開始推動的「教學創新試辦計畫」、「技專校院教學創新先導計畫」，以及後續「高等教育深耕計畫」的目標中，推動讓大學程式設計教育普及化，以經費挹注各校開設程式設計相關課程，在確保教學品質下，使各大專校院修讀過程式設計相關課程的學生比例能超過百分之五十之量化目標，而這個目標於 109 年 6 月統計已達到了百分之六十達到目標值(教育部電子報，2020 年 7 月 2 日)。

然而，「程式設計」對於學生而言，是一門挑戰性極高的課程。目前臺灣的大學入學新生，極大多數都是屬於程式設計的初學者，普遍都是在進入大學後，才開始接觸到程式語言。儘管有少部分學校在 12 年國教階段已開始導入運算思維、創客自造、積木式程式設計、機器人等基礎程式設計教學，然而學生在不同

的國、高中所接受到的程式運算、邏輯訓練、問題解決等課程差異極大，呈現出顯著的數位落差(高嘉峻、蔡銘修，2018)，使得高等教育程式設計的教學充滿著挑戰。

程式設計在世界各國大學實施的狀況，不管是什麼科系背景的學生，學生學習的情況大致相同，大都維持著低通過率、重修人數比例高、學習動機低落等問題。Mederiros et. al (2018)為了探討程式設計課程學生學習成就低落的問題，採用文獻回顧法，系統化地經由檢索、探索引文等方式，篩選出 100 篇的研究文獻進行分析，來解答三個研究問題：程式初學者需具備哪些關鍵學前技能與背景知識？初學程式的學生遇到那些學習困難？教師課堂上的挑戰為何？研究結果指出，要學好程式設計需具備的相關基礎技能，依序為：問題解決能力、數學能力、學過程式設計運算思維、摘要能力。而學生學習的困難可歸納成：問題的界定與形成、解決方案表達、解決方案的執行與評估、學習行為四大類，而各類中最常見的困難分別是問題解答、程式語言語法、除錯技巧、動機與投入。而教師所面臨的諸多挑戰，最常見的幾個依序為：教學方法與工具、維持學生的動機投入和堅持、師生的溝通與回饋。

同樣對於學生在程式設計課程的通過率不高，提升學習成效的問題上，也有學者針對所採用之教學介入的形式，以文獻分析法進行系統化的回顧(Vihavainen et. al, 2014)。相關程式設計教學的文獻，經過檢索、過濾、篩選後，剩下 32 篇來分析教學介入的形式，將教學介入的形式，以詞彙方式來編碼，之後進行統計分析。最後歸納出，有下列十種常見的活動形式，依頻率最常見的依序為：教材變革、同儕支持、協同合作、情境化、配分比重、媒體運算、遊戲主題、協助支援、先修課程、團隊合作學習。而其中有些是採取多種教學介入之混合方式，從教材建構、工具的使用、評分比重的調整、成對編程、團隊專案製作等，對傳統課程的教學模式進行改革，可提供學生課程的通過率。

本研究旨在探討程式設計課程，如何在教學時數不足、學習動機不高下，引入新的教學方法，設計更符合參與課程學生特性的教材，以提升學生的學習成效。程式設計教學的目的，並不是要讓人人都能成為專業的程式設計師，也不見得在各行各業裏都須要專業的程式設計師，但是當遇到能用程式輕易解決問題時，能有使用程式解決問題的能力。

1.2.2 標記式語言與開發技術框架之認識—網頁設計

網頁技術的興起與演進相當迅速，自 1990 年代開啟了 Web 的年代以後，接著 2000 年代進入了 Web 2.0 的年代，之後又演進到了行動化網頁的年代，不僅創造了許多新的商業模式，在設計、開發網頁的技術也推陳出新。除了最基礎的 Html、CSS、JavaScript 可組成靜態網頁外，互動式網頁熱門的技術，例如 JQuery、W3.CSS、Bootstrap、React.js、Node.js、Ruby on Rail、Python Django、PHP Laravel，從前端到後端的各種開發框架、程式庫一時之間湧現。而 2018 年後，單頁式的應用(single-page applications)大量出現逐漸替代傳統的輕量化網站，網站(web

site)亦可改稱為(web application)，許多開發團隊前端採用 Facebook 開發的 React Framework (Connolly, 2019)。對應於目前高等教育的課程，這些技術僅部分在資訊相關科系的「網頁設計」、「網站開發」、「互動式網頁程式」、「後端伺服器程式設計」等這一系列的課程中提及。而令人訝異的是北美大學的資訊科系課程中，並沒有特別針對網頁技術開課，而是散佈在其他的程式設計課中(Miller & Connolly, 2015)。與教學相關的文獻自 2013 年後也相對地減少，推測可能是與現實世界中網頁開發實務不斷改變的本質有關(Connolly, 2019)。由於網頁技術變化太快，目前學生在校所學的，到三、四年畢業後在實務上所採用的主流技術，將自然產生一些落差，因此培養能與時俱進的自學能力顯得格外重要。

對於非資訊相關科系而言，最基礎的「網頁設計」課程，可以讓學生認識每日生活都會接觸到的網站技術，如同現階段教育部所推動的大學生在畢業前都須修過程式設計課(教育部電子報，2020 年 7 月 2 日)以培養運算思維，了解網站背後製作開發的原理也有助於運算思維的培養。依據教育部「大學課程資源網」¹的資料顯示，在 109 整學年的課程中，課名包含「網頁設計」這四個字的課程數共有 573 門，分布在 113 所學校的 212 個科系中。然而同樣條件查詢 105 整學年的開課情形，則有 757 門課程，分布在 123 校的 247 個科系中；99 整學年，則更高達 1710 門課程，分布在 273 校的 330 個科系中。近十年內網頁設計課程的數量呈現大幅衰減，尤其是在資訊相關科系，與北美高等教育的發展趨勢一致。然而不可忽視的是，最早來自 Html 以及 CSS 偏向設計觀點的網頁設計課，現今仍普遍見於大學各個非資訊科系以及通識課程中，教學現場所遭遇的問題，仍值得深入探討。

不同科系對於所培養的網頁技術素養，有不同的深度與廣度。「網頁設計」做為網路技術的入門基礎課程，可做為「計算機概論」、「資訊概論」等課程的延伸，或「程式設計」、「大數據分析」的先修課程。入門的「網頁設計」課程，包含 Html/CSS 這類標記式語言(markup language)的使用，會用到程式編碼和除錯，有助於程式設計的學習(Jin, 2017)。也有助於大數據分析、資料科學等相關課程的學習。因其所使用的數據，有許多是由網站擷取而來。較進階的「網頁設計」課程，則考量了現實世界中網頁技術的複雜性，網站建置需要很多技術的合成。過去許多科目都是講授單一的觀念或技術，如資料庫、程式設計、網路通訊等，缺乏整合再一起的實作機會。而網頁技術正是這類訓練學生整合多項資訊技術能力的機會，將 Html、CSS、JavaScript、Bootstrap 等，也包了 Photoshop 影像處理等設計方面，以及 Web 伺服器，整合在一起，是一種資訊技術整合能力素養的訓練。

然而，若僅有一門課讓初學者學習網頁設計的話，課程的重點會著重在 Html、CSS、JavaScript 等門檻觀念(Threshold Concepts)的建立。或是著重在門檻技能(Threshold Skills)的養能，具備善用開發工具動手做的能力。不論課程目標是門檻觀念或是門檻技能，教材所涵蓋的內容、教學教法等，都存在著極大的彈性，未

¹ 教育部大學校院課程網: <http://ucourse-tvc.yuntech.edu.tw/>

能有一致適合各類學生的課程。而非資訊科系的學生，對於與程式相關課程的學習興趣通常不高，也影響到了學習成效。網頁設計通常包含了前端程式設計以及網頁美工的部份，大多數工程相關科系的學生，對於與程式相關的前後端互動設計會比較感興趣，而設計相關科系的學生，則比較喜歡美工的部份，對程式編碼的部分覺得較困難(Guo et al., 2021)。這使得課程所涵蓋的範圍難以決定，那些需要忽略?那些要納入?這類互動科技的課程，Shaffer & Resnick (1999)建議應考量到四項需求平衡，包含學生的興趣、真實世界的需求、所屬學科的特性、學習成效評量，而這個平衡，也是造成學生學習困難的主因(Connolly, 2019)。而這個平衡在送審人所任教的科系相當關鍵，因為學生入學管道即分成兩組，一組為偏向設計的傳播組，學生大多來自高中的社會組;另一組則為偏向程式編碼的資訊組，學生來自高中的自然組。這兩組的學生會在同一個班級內修讀網頁設計課程。因此，送審人過去收到的學生回饋也呈現兩極化，最常抱怨的是”學網頁設計以後又用不到”、”為何要考網頁設計認證”，而也有學生興奮地開玩笑說，”老師!我做出一頁式購物詐騙網頁了”。為了兼顧學生兩極化的特性，課程內容涵蓋範圍的取捨，需進一步的研究與探討。

過去送審人在所任教的運動資訊與傳播學系中，教授「網頁設計」的課程較著重於門檻技能的養成，側重於網頁設計工具的使用，一學期中約有三分之一(6週)的課堂時間用於輔導學生通過 Dreamweaver 證照考試。同時，系所畢業門檻亦要求學生畢業前須取得資訊類的證照，而證照的取得數也使得教學績效指標呈現上有著亮眼的成果。然而，傳統填鴨式的證照考試引導教學，侷限在一再練習題庫以熟悉某一套編輯工具的教學方法，限制了學生未來的發展。學生也反映機械式的解題操作讓課程變的枯燥乏味，使得原本就不高的學習興趣更加低落。在 108 課綱實施後第一屆即將入學的大學新生上，以畢業門檻要求學生考過證照，再結合課程輔導的高教生態體系對本系而言已不合時宜，故開始進行大規模的 111 學年課程改革，廢除畢業門檻的限制，讓證照考試與課程脫鉤，課程設計與教學法因此獲得了釋放，而有更大的改革空間。完整的網頁開發生態系非常複雜，許多技術無法在一兩門課程中涵蓋到，很大的部分需靠自學(McCartney et al., 2016)。那些部分的觀念與技巧是重要地，可參考畢業生流向調查來進行調整，依本系之調查結果以擔任媒體、行銷、資訊等工作居多。做為網頁技術第一門起步課程的網頁設計，其目的應當是要建立學生自學的基礎知識，也就是網頁技術之基礎素養養成，具備一定程度專業能力以符合就業門檻的需求，而不再強調工具的操作。

有鑑於此，本研究將採用「體驗式學習」、「遊戲化專題導向學習」、「漸進釋放責任模式」來發展「網頁設計」課程的教學與學習活動。在課程第一個部份，以體驗式學習之主動實驗、具體的經驗、反射觀察、抽象概念化，四個階段所構成的循環，經由設計過的動手做活動來學習 Html 和 CSS 的基本語法。首先，引導學生使用瀏覽器的開發模式，一窺網頁背後的原始碼組成，探索 Html 元素的階層化結構、DOM 結構化摘要，繼而再動手修改，觀察結果。接著讓學生使用

subline text、visual studio code 這類免費的編碼工具，以 flexbox model 結合 CSS 呈現的網頁，進行 CSS 選擇器的變換、CSS 樣式的套用，來領略 CSS 的運作與語法。課程的第二個部分，則藉由真實專案的需求由的引入，以開發行動化網站所需具備的能力，與實際所學的落差，來進行專題導向學習，除了補充響應式網頁設計(responsive web design, RWD)，以及常用的 W3C.CSS、bootstrap 等程式庫，和基礎之 JavaScript 語法外，也鼓勵從專題的需求出發，主動學習前端程式的開發框架，以及了解網站實際運營時之伺服器設定和相關雲端服務。過程中也同時持續進行體驗式的學習，選擇幾類的典型網站範本讓學生學習如何將樣板修改成自己所需。而真實專案需求的導入，為延伸體驗式學習到校園內之專業服務學習，讓學生以小組合作學習的方式，扮演各專案團隊角色，實踐遊戲化專題導向學習，為校園內公共服務的實際需求者開發所需的網站，做為期末專題之成果。

1.2.3 資料素養培育的根基—資料庫

「資料庫」相關的課程，例如「資料庫管理」、「資料庫設計」、「資料庫應用」、「資料庫系統」等，是計算機科學及資訊技術中重要的一環，不僅是資訊電機類系所之必修課程，也在一些非資訊科系中開設。大多數學生修讀過程式設計有基礎之後，接下來就會接觸到資料庫的課程。資料庫之課程領域儘管在學術屬性上比較偏向應用科學，但其理論與應用核心基礎來自數學之「抽象化」(abstraction)觀念，對於主修資訊或非資訊學生而言，與程式設計課程同樣存在著學習動機低落、學習困難和學習成效不佳等問題。不論是機器學習或是大數據分析，其基礎都是來自資料庫中的資料集，學生除了具備運算思維外，也應深化建立「資料庫思維」，以符合萬物皆數據(every is data)的時代發展現況。

資料庫相關課程依教育部「大學課程資源網」顯示，臺灣的大專校院於 108 整學年在 63 校 206 個系所中共開設了 535 門課，涵蓋了各領域學門，共有 20487 人修課。可見「資料庫」在各個領域都受到相當程度的重視。除了傳統資料庫課程外，隨著各領域不同的應用特性，課程的廣度、深度和授課重點也適度調整為理論、實作、管理、分析及應用等。例如，有系所結合其領域知識開設「生物資料庫理論與實作」、「人文資料庫管理」、「不動產資料庫管理」、「商業資料庫管理」、「健康資料庫分析與處理」、「金融資料庫分析與應用」、「新聞資料庫應用」、「水產資料庫應用」等有特色的課程。

傳統資料庫的課程是以資料庫管理人員為目標職稱建置與實施職能，使其具備產業所需的資料庫管理專業實務能力，然此並不全然適用於非主修資訊領域的學生。而大多數現有資料庫的教材也是如此，未能考量到不同領域的學生的需求。不僅教科書的內容千篇一律地依軟體專案流程講述，從需求分析、資料模型、資料庫設計、SQL 語法、資料庫維運最後教到應用程式之開發，內容雖十分完整，但在有限的教學時數內僅能有初步的了解，能達熟悉、精通者甚少。課堂上若以傳統講述法教學更會使得學生被過度餵食爆炸式的資訊而無法消化，而若以翻轉教室、合作學習、專案導向式學習、問題導向學習、做中學，則需有足夠的資源

來提升學習動機及學生主動學習的意願才容易成功。此外，資料庫教學上常採用的範例資料庫，通常也是採用教科書所附，一般常見應用領域之資料庫模擬簡化版本，如：員工/部門/專案、學生/課程、註冊、活動、書/影音儲存資料、訂單/訂單項目/組成部分等資料。採用這類範例資料庫的好處是，教師只需花一點時間來解釋這些領域，學生也容易聽懂，但缺點是與現實世界的資料庫實例落差極大。而對於學習動機較低落的學生，這類的範例資料庫更讓學習顯得乏味，以進銷存系統、校務系統、圖書系統的資料庫來學習，使得非主修資訊的學生對於為何而學易感到困惑，失去學習的興趣。

學生對於傳統資料庫的教材教法無法感到興趣，使得通過率低、重修人數比例高、學習動機低落等問題，在送審人於運動資訊傳播學系所任教的資料庫管理課程中也觀察到同樣的現象。故近幾年透過親自面談輔導期中預警學生的機會，來蒐集學生的意見回饋，以滾雪球式的漸進方式，逐年持續在教材與教法上進行部分新的教學模組開發、試教與改進。學生的回饋意見顯示，較易出現學習困難的單元依序為正規化、SQL 語法、實體關聯模型(E-R model)，實務操作使用資料庫範例也傾向於與學生主修領域相近的運動休閒較能提升學習興趣，對於曾在美國職棒大聯盟(Major League Baseball, MLB)打棒球的臺灣人有哪些?去年度 MLB 各隊球員薪資總和最高的是哪一隊，與戰績有關嗎?Moneyball 的效應看的出來嗎?休士頓太空人偷暗號拿冠軍，戰績差多少?這類有趣的檢索來學 SQL 語法，是能有效提升學習興趣的方法之一。

依資料庫系統之專案流程順序進行教學，也容易讓非主修資訊的學生提早遭遇到學習的困難(Dobesova, 2016)。學習的途徑應該先從貼近學生所屬主修領域中的實務使用情境優先，先接觸實例資料庫了解整體輪廓與應用後，再帶入容易產生學習困難之基礎理論之推理與思考。因此，本研究的動機是希望提升學生主動學習的意願，以真實世界中應用之資料庫為核心基礎來設計教學內容，並翻轉傳統以專案流程學習資料庫之順序，重新建構以情境式引導和實務應用之操作型態為主的學習進程。

棒球運動在臺灣被稱為國球，與棒球有關的話題非常親民而且接地氣，對於大多數的大學生而言十分具有話題性與吸引力。尤其是主修運動休閒領域的大學生，對於棒球的熱情更勝於其他的專業學科。因此若能以棒球資料庫為實務操作範例，來學習資料庫的課程，預期將能激發起學習的熱情與火花。同樣以棒球為主題場景用於跨域之情境引導教學上，已有從看棒球學英語(王啟恩等人, 2016; 王曉寒, 2008)、棒球學統計(Albert, 2002; Rothman, 2012)、從棒球學物理(李中傑, 2016; Adair, 2002)等教材，但尚未有完整且全面之從棒球學資料庫的大學課程教材與教學案例發表。

Sean Lahman's Baseball Database²(以下簡稱 Lahman DB)是棒球界知名的資料庫，是由一位目前任職於今日美國網(USA Today Network)的記者 Sean Lahman，收集了 MLB 歷史上每支球隊和球員的棒球統計數據所建立的，於 1995 年開始

² 網址：<http://www.seanlahman.com/baseball-archive/statistics/>

在網際網路上免費提供給大眾下載。本研究將採用 Lahman DB 之實例來發展資料庫的教學情境與實作，主要有下列三點理由：

- 一、資料全面且架構完整，規模適中：近年來每場比賽的記錄與統計資料幾乎沒有短少，且涵蓋的資料不僅有球員、隊伍、比賽結果、戰績表現，還包含了球員薪資、球員獲獎、球場等，全部共 27 個資料表。其資料記錄的年代最早可追溯自 1871 年，且歷年來皆維護良好，每年持續更新資料中，其規模大小非常適合教學使用。
- 二、資料開放且可支援各類資料庫格式：Lahman DB 採用了創用公眾授權 (Creative Commons Attribution-ShareAlike 3.0 Unported License)，下載處並提供了微軟 Access、逗點分隔(comma-delimited version)、R Package、MySQL、SQL 備份檔等各類資料庫系統之檔案。不論教學上採用哪一種資料庫系統，皆可直接應用於資料庫之安裝、還原、備份、資料匯入、實體檔案結構等教學。
- 三、可結合使用者互動查詢網站，進行實務操作之驗證：現有可公開存取的棒球資訊查詢網站，如：Baseball-Reference.com、Retrosheet、Baseball Savant，提供了網站使用者互動的資料查詢介面，可用於發展資料庫課程之所見即所得(what you see is what you get, WYSIWYG)創新教學。在網站上查到什麼資料(what you see)，在資料庫介面中同樣用 SQL 也要能獲得(what you get)。同樣地在 Lahman DB 使用 SQL 檢索的結果，也可以透過上述網站介面之查詢結果進行驗證。
- 四、領域層次的知識(domain-level knowledge)容易融入：棒球運動是一項普遍被接受的休閒活動，而在運動觀賞行為中，當觀眾涉入程度高時對球隊或球員易產生忠誠度，而出現愛好者(fans)的行為。除了熱衷查閱各類運動數據的需求可移轉到對資料庫存取的學習外，往往對於數據想要進一步地分析的渴望，也可移轉到對於進階 SQL 語法和建立個人客製化資料庫的學習熱誠上。

國內外大學尚未具有完整性且全面性採用棒球進行資料庫教學之課程與教材，僅有少數的線上平台如 Udemy 提供了” Baseball Database Queries with SQL and dplyr” 3 這類 3 小時的短課程。因此本研究將秉持著因材施教適性發展的原則，以真實世界的 Lahman DB 和棒球網站為實務操作案例，以其為核心基礎來客製化發展適合運動領域和喜愛棒球之學生修習的課程與教材，並實踐於教學課程中進行行動研究。

1.2.4 運動大數據時代下的資料科學

「資料科學」(data science) 亦稱為「數據科學」，與「大數據分析」、「巨量資料分析」意義相近，是近年來興起的一門新的課程，目標是要學生能從資料中，利用一些所學得的理論與技術，包括包括應用數學、統計、圖型識別、機器學習、

³ 課程網址：<https://www.udemy.com/course/baseball1/>

資料視覺化、資料倉儲以及高效能計算等，來提取出有價值的知識而加以運用，協助讓非專業人士也能理解問題，幫助正確的處理大量資料，得以進行研究調查，或應用於商業競爭上。而在運動產業中，各式各樣的數據也出現巨量的成長，從過去一直保留至今的各場次比賽紀錄資料庫，到近年來在運動科學上邁向科技化的應用，利用各式感測器(sensor)量測的生心理數據、生物力學數據、空間數據、各式運動測試數據等，這些大量數據後所隱藏的知識，需有足夠的運動領域知識並精通數據分析方法才有機會發掘。然而在主修體育之相關科系中，有關資訊科技應用的課程仍略顯不足(Barneva & Hite, 2017)。

而國內大學「資料科學」的課程，依據教育部「大學課程資源網」的資料顯示，最早應始於 103 學年東吳大學的巨量資料分析學分學程開設「資料科學與巨量資料導論」開始，從 103 學年全國大學(不含技職)開設課程數 1 門開始，隔年 104 學年度共開設 27 門，而到了 107 學年更出現爆炸式的成長到有 38 校共 61 個系所開設了 120 門課程。而這些開設課程的系所，大都是資訊相關科系，而少數非資訊科系開課的則有教育、經濟、金融、企管、心理、社會、應數、醫管、土木等系所，可見「資料科學」在各個領域都受到相當程度的重視，且隨著各領域不同的應用特性，課程內容也應隨之調整，如某些系所結合其領域知識開設「健康資料科學概論」、「教育資料科學概論」、「資料科學與社會研究」等課程。

反觀運動產業相關科系，在面臨大數據資訊時代到來之時，運動大數據分析的教學大多分散在碩士班或專題類的課程中以部份時數教授，尚未能有整學期「資料科學」完整課程的開設。但對於運動產業，資料科學並非不是不重要，而是資料科學須結合許多基礎領域的知識，對於數學、統計、程式設計、機器學習等需有一定程度的基礎，而這些基礎領域通常也不在體育相關科系的開課範圍內，因此對體育相關科系形成很高的開課門檻。

在國內體育相關科系中，結合運動產業的特性開設「資料科學」課程實已刻不容緩。而對於這些未能有足夠基礎知識主修體育的學生，一個良好課程的規劃與設計，包含教材內容、教學方法、學習活動、技術實作等，會是影響學習成效最重要的關鍵。

另一方面，「運動分析」這門學科，在 Moneyball 這類題材近幾年開始受到注目後，以往僅於運動愛好者間普遍討論的棒球統計學(又稱賽伯計量學、Sabermetrics)、籃球計量學(APBRMetrics)等，逐漸擴展至大學課程中，並結合運動情蒐、運動科學、運動感測器等既有運動學術領域之發展，在國外的大學中逐漸演變發展出「運動分析」課程及學位學程，如美國西北大學設有 Master's in Sports Administration Specialization in Sports Analytics⁴，英國羅浮堡大學則設有 MSc Sport Analytics and Technologies programs⁵。

運動分析所探討的實務問題，非常適合發展「資料科學」課程教學之情境以

⁴ 網址 <https://sps.northwestern.edu/masters/sports-administration/analytics-specialization.php>

⁵ 網址 <https://www.lboro.ac.uk/study/postgraduate/masters-degrees/a-z/sport-analytics-and-technologies/>

及用於設計課程實作，主要有下列三點理由：

- 一、現有的運動資料集十分完整：各項運動每場比賽的記錄與敘述統計資料幾乎沒有短少。尤其是棒球比賽的數據，歷年來皆維護良好，堪稱是目前世界上最豐富的運動資料庫。而這些資料通常都是免費開放的，可以輕易取得，在進行分析前不太需要進行資料清理與調整，且每場比賽的規則明確，使得資料的因果關係容易確認。
- 二、運動分析所要探討的問題定義明確：所有的運動都遵守一項明確的規則：誰贏了？誰輸了？而大多數運動分析主要要解決的問題，就是預測出比賽輸贏。其次的問題，則是要去說明為何會造成輸贏，如探討某位運動員場上的表現，或是因為採用某種策略而導致輸贏。
- 三、領域層次的知識(domain-level knowledge)容易融入：運動是一項普為被接受的休閒活動，而在運動觀賞行為中，觀眾涉入的程度高且對於球隊或球員會產生忠誠度，而出現愛好者(fans)的行為，除了能對運動競賽規則有清楚的了解外，往往也會對於競賽的結果預測有一些個人的初步分析與看法。而其他領域中的資料分析，分析者較難有相類似的環境，去深入了解這些領域的知識規則。

目前國內的大學，尚未有適合運動相關科系學生修習之「資料科學」課程與教材，因此本研究將秉持著因材施教適性發展的原則，以「運動分析」之實務問題為基礎來客製化發展適合運動領域學生修習的課程與教材，並實踐於教學課程中進行行動研究。

1.3 研究主題

本研究旨在研究開發出一系列適合主修運動休憩管理領域學生的資訊技術實務課程，包含了程式設計、網頁設計、資料庫管理、資料科學四門課。課程的設計需要兼顧學生的學習動機與興趣、符合學生的知識技能與程度，並克服學生對於計算機科學的恐懼，建立學生學習的成就感與自信，並達到一定程度的學習成效。依據不同課程採用多元的教學方法下，研究主題為：(1)學習動機與興趣之提升；(2)教材範圍與學生知識領域及技能程度之符合度；(3)學習成就感與自信心之建立；(4)學習成效確保與評量方式之探討。各課程所對應的研究主題詳如表 1。所採用的多元教學方法與各課程間的關聯，如圖 1 所示。

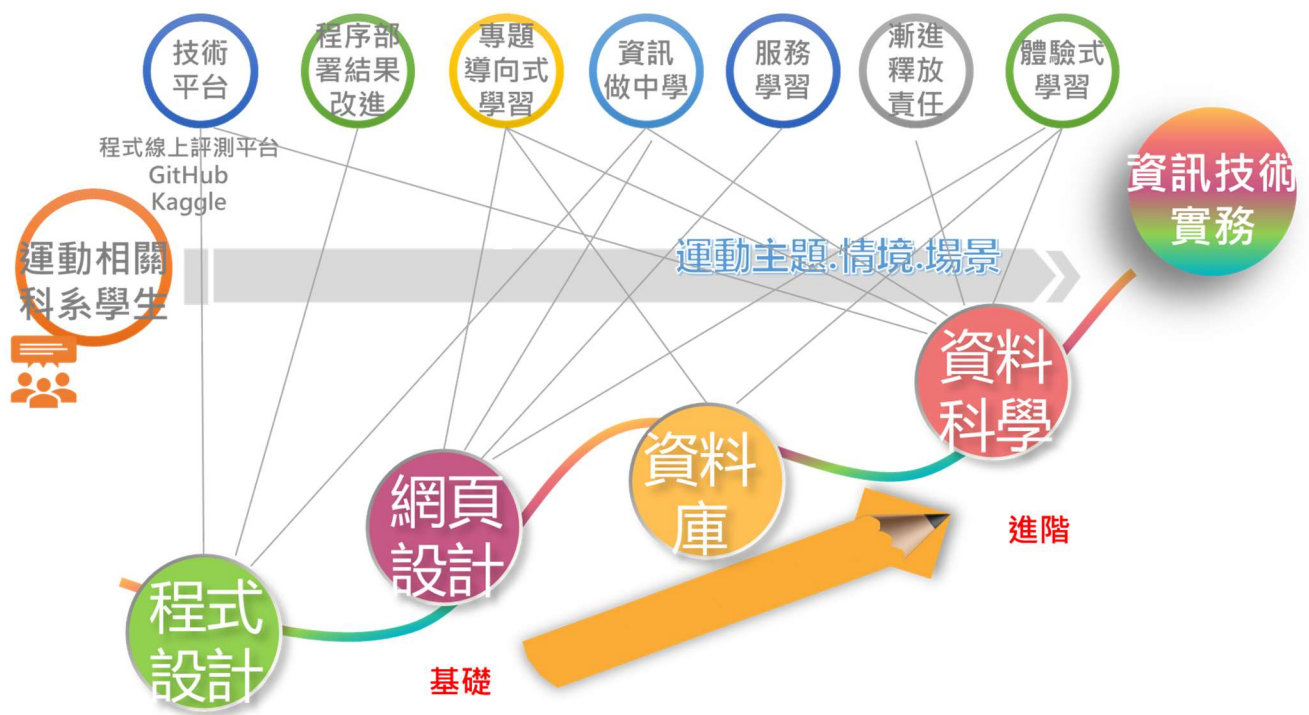


圖 1 資訊技術實務課程研究架構與發展歷程

所設計出來的四門課程，於 108 學年至 111 學年於國立臺灣體育運動大學運動資訊與傳播學系大一到大三的課程中實踐。以行動研究法來了解學生對於課程的反應，探討課程內容設計、教學教法、技術實作成果，對於學習成就感與自信心、學習成效、學習滿意度之影響，並加以強化與改進課程，期望能達到以下的研究目的：

- 一、探討本研究所提出的課程設計，是否能符合體育相關科系學生之客製化學習需求。
- 二、利用各課程中所導入「運動分析」、「棒球」、「奧林匹克」之等運動場景，先經真實案例之觀摩與實務操作後，再學習基礎原理時，探究學生在概念學習與技術實作過程間之問題與解決策略。
- 三、探討情境式引導、問題導向式學習、專案導向式學習、合作學習策略、做中學之技術實作，對學生學習投入之影響。
- 四、綜合研究成果提出意見，做為相關課程改進之參考。同時透過行動研究歷程，幫助教學研究者自身的專業成長。
- 五、結合學校特性，發展具運動主題特色之資訊技術實務課程，以與其他大學系所相關課程有所差異。

表 1 課程及研究主題

	程式設計	網頁設計	資料庫	資料科學
多元教學方法	ADRI、做中學、線上評測、成對編程	體驗式學習、專題導向式學習	體驗式學習、逆向設計教學順序	體驗式學習、漸進釋放責任
學習動機與興趣之提升	導入程序部署結果改進模式(Approach, Deployed, Result, Improvement, ADRI)之教學模式，並加強學生「做中學」(learning by doing)的經驗來設計符合目前任教科系學生能力的教材。	藉由角色扮演的遊戲化專題導向學習，讓課堂活動變得有活力、更有趣。並藉由專題小組內合作學習，透過同儕的互動來提升學習的動機與興趣。	藉由對於棒球運動中的球隊、球星、比賽戰績等大眾化運動話題來包裝資料庫管理課程，讓學生能有著球迷般的熱忱，透過資料庫技術一起探討棒球數據。以降低學生對於資訊科學和數理基礎自信心不足的恐懼。	藉由運動領域之實務問題來包裝資料科學課程，降低學生對於數理統計和程式設計的恐懼。課程的將參考現有運動領域中「運動分析」(sports analytics)課程常見的分析主題，挑選出適合發展運動產業資料分析的主題，將傳統以統計方法所進行的運動分析，嘗試以資料科學的方法重新進行另一種面向的分析與闡釋。
教材與學生知識領域及技能程度之符合度	TQC Python 程式設計題庫、以及大學程式教育推動計畫之教材	做為基礎網頁技術素養的建立，兼顧專業門檻觀念與門檻技能之課程內容，是否能符合本系學生背景與未來就業所需，課程之廣度與深度是否造成了學習負荷與學習困難，有待進一步研究與驗證。	參考社群網站和媒體中有關國內外棒球及 MLB 普遍被討論到的熱門話題，挑選出適合發展資料庫課程之主題，重新建構教學單元的內容與順序。並以 Lahman DB 之真實世界案例取代將傳統的資料庫課程使用的模擬案例。	參酌目前資料分析之競賽平台如 Kaggle，以及 Github 上之程式分享來編製課程實作之教材。

學習成就感與自信心之建立	程式成對編程之合作學習策略。期末以通過認證考試來建立學習自信心與成就感。	學生可將實作結果透過自己的手機呈現，並藉由校園內實際服務學習對象對於專題實作成果的肯定，來建立成就感與自信心。	讓學生親手透過資料庫的實務操作，完成指定的任務，提出棒球數據之報告，讓學生實際感受到「I can do it by myself」來建立成就感與自信心。	讓學生親手透過軟體或程式之操作，完成指定的任務，提出真實世界資料解析之報告，讓學生實際感受到「I do It by myself」來建立成就感與自信心。
學習成效之確保	導入之自動化線上程式評測系統，探究學生在程式編碼實作過程之問題與解決策略。	利用服務學習以團隊合作和遊戲化專題導向式學習，引導學生以小組方式開發滿足現實世界需求的網站。除了專業知識與技術實作層面外，也培養了學生的溝通能力、責任感，和各方面自我效能的提升。學習成效的衡量需採用多元化的設計，哪一種問卷與量表適合使用，有待進一步研究。	以團隊合作和專案導向式學習，引導學生以小組方式發掘現實世界的資料需求和資料庫實務操作上兩者間的缺口，並依循課程中所學的各個實作任務，綜整成為一個完整的期末專題來解決問題缺口。	以團隊合作和專案導向式學習，讓學生完成小組獨立研究，引導學生發掘實務問題，並依循課程中所學的各個實作任務，綜整成為一個完整的專題研究，提出專業的分析報告與決策建議。

2. 相關文獻探討

2.1 程式設計教學相關文獻

2.1.1 ADRI 教學模式

程序部署結果改進模式(Approach, Deployed, Result, Improvement, ADRI)為一種廣被採用品質確保模型，最早是由 Walter Shewhart 以 Plan-Do-Check-Act(PDCA)模型為基礎，於 1939 年發展出來的(Johnson, 2016)，之後也經常被延伸應用到教育和商業領域中(Razvi et al., 2012)。在商業上，澳洲籍紐西蘭的企業評比機構，採用 ADRI 模型來對企業進行品質獎的評選，在教育產業中，澳洲與紐西蘭的大學，以及臺灣的大學評鑑中，也採用了 ADRI 模型來建立教學品質的保證機制。

ADRI 在程式設計的教學應用，最早是 Malik 等學者應用 ADRI 模型發展創新的系統化教學模式，於基礎程式設計課程中實施，以及用來提升學生的學習成效，並進行一系列的延續性研究(Malik & Harsh, 2013; Malik & Coldwell-Neilson, 2017a; Malik & Coldwell-Neilson, 2017b); Malik, 2018)。這些研究結果，均能反映以 ADRI 的程序進程式設計教學，可以獲得學生正向的回應，並提高了修課的通過率，同時教師也可由研究中來分析所採用的教材和評量工具，調整教學的方式。

ADRI 模型的教學建議，主張程式初學者在課程學習的過程中，應該同樣地重視解答問題的策略和程式語法語意的知識上，透過 ADRI 模型的四個階段設計，可以實踐上述目標。四個階段的程式教學設計如圖 2 所示，說明如下：

程序(Approach)：著重於問題解答策略，例如運用 pseudo-code 和流程圖，針對問題敘述提出解答。學生了解問題敘述與限制後，可提出一般解和演算法，而不需考量是否具備程式句法和語意的知識。

部署(Deployment)：著重於程式語言的句法和語意，了解程式語言的規則和結構。學生須將前一階段提出的演算法，開發為程式。並且了解直譯、編譯的過程，以及測試和除錯的技巧。

結果(Result)：著重於程式的輸入、輸出、處理程序，以及共通的程式撰寫錯誤。學生必須了解解答問題的程序，了解程式輸出輸入格式的規定和資料型態，一些程式共同慣用的語法，以及了解當程式邏輯錯誤時會如何呈現。

改進(Improvement)：於原本的問題上，增加新的問題描述與要求，用來增加實作的經驗，並且嘗試以另外一種程式語言結構，來解答同樣的問題。學生須著重於分析新增的需求，並且這些需求以原先的程式進行擴充，提供更深入的程式實作機會。

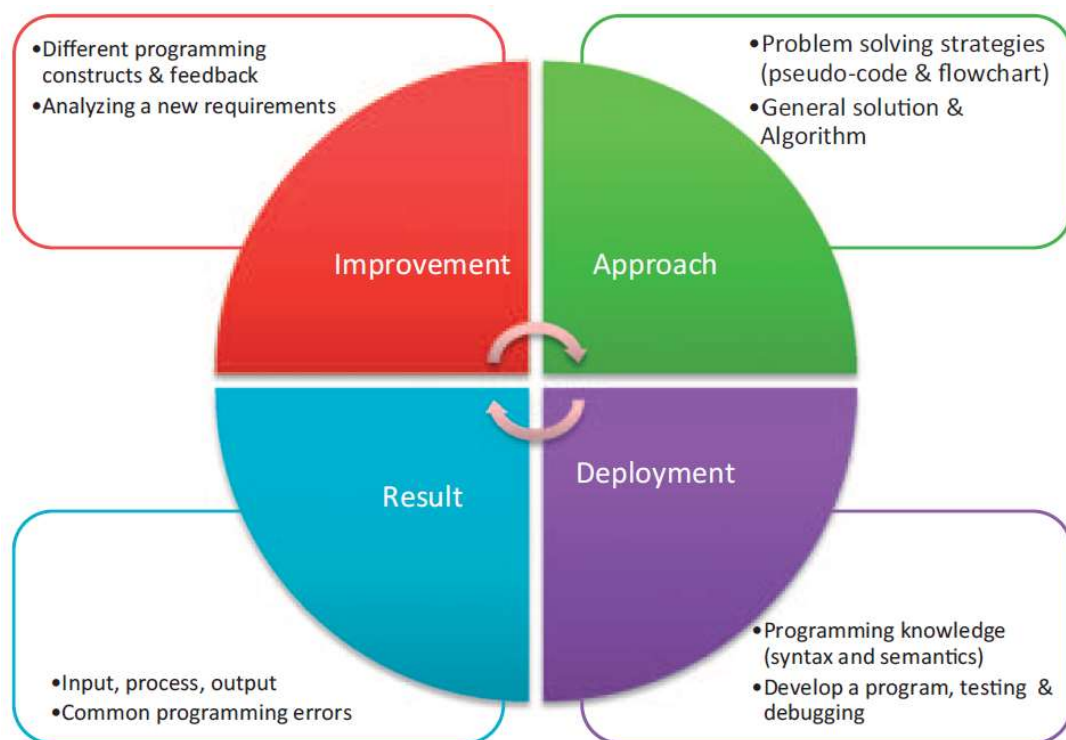


圖 2 ADRI 四階段的程式教學模式(資料來源：Malik & Coldwell-Neilson, 2017b)

2.1.2 自動化程式線上評測(online judge)

以程式設計而言，了解程式設計的原理，與了解如何寫出程式，仍有一段距離。以程式的撰寫而言，一開始都是接觸到片段的幾行程式碼，範圍大約是該次上課的章節。當需要利用寫程式來解決問題時，通常都是到了下半個學期，教學進度累積到一定程度時，而學生此時才會深深體會到，由於先前的片段程式碼基礎學的不夠好，而無法利用程式來解答問題。因此有研究者提倡(Brito & De Sá-Soares, 2014)，基於建構式教學主義的理念，知識的獲得來自自身經驗與體驗，而不是經由某種類型的資訊傳遞，因此在程式實作能力的提升上，應當增加學習成效評量的頻率，並結合系統化的教學模式。評量的頻率最好是一週一次，因為一週的時間尚可讓學生去改變學習的方法，並且能讓教師能適時提醒那些需要關注的學生。而當要實施每週評量時，沒有額外的資源很難達成，此時自動化的評測系統可做為程式設計教學輔助的選擇。

程式實作能力的提升，可藉由具自動化程式線上評測功能之教學輔助平台的導入，讓學生有更多實作練習的機會。傳統程式上機實作的教學流程，是在課堂上實施，在教師講述該課程教學內容與範例後，安排實例讓學生實際練習程式編碼，再輔以課堂的答疑與實驗指導。但受限於課堂時間及生師比的問題，課堂實作往往變成了打字練習，練習輸入程式碼。而一位教師要面對全班數十位的學生，很難在課堂上個別一一去解答學生的問題。而不論是課堂上的實作，或是安排課後作業實作，學生是否有確實進行實作，最大的關鍵仍在於主動性。

自動化程式線上評測系統，在許多教學研究中被提出作為輔助教學的工具。教師可在教學平台上，依據課程的內容設計程式實作的題目，學生則可於平台上作答，實作編碼或提交已完成的程式碼，平台會自動依據作答自動進行語法剖析與輸入輸出測試，給予分數。並記錄學生的學習次數、分析各題目之答對狀態、學生學習的弱點等。

Gou et al., (2017) 引入了線上 PTA (Programming Teaching Assistance) 教學在線判題系統，實施於兩個計算機類科系的大一班級，每班均為 40 人，於程式設計基礎課程中教授 C 語言。其中一個班級採傳統的程式實作練習，另一個班級採新型態的線上線下程式實作練習。研究結果顯示，採用新型態的實作練習，在期末考的表現上，學生獨立完成題組作答的能力有很大的提升，而平均答題與練習數量也有顯著的提升。

而不僅是線上評測，進一步也有學者以自動線上評測的平台為基礎，進一步發展自適性的程式設計學習平台(紹陽林等人, 2017)，平台會依據學生目前實作練習的評分結果，依學生程度自動分派下一個練習的題目，針對不熟、不懂、答錯的部分加強練習。

除了線上評測、自適性學習的功能外，有的平台更加強化程式實作時，在答題及更正錯誤時的立即回饋功能(Palominos et al., 2018)。傳統課堂上進程式實作時，當學生有編碼的問題時，教師須到學生電腦前，一起看著螢幕進程式碼除錯。而透過平台學生直接反映實作時的困難，而教師可以透過平台查閱程式碼與錯誤訊息，快速的了解學生的問題，給予指導，強化了傳統課堂上，師生溝通反饋的機制。

本研究為了更容易掌握，學生是否確實進行課後作業的練習，了解學生的學習困難，故將導入自動化程式線上評測平台，並設計合適的教學內容，搭配課堂實作與課後練習的增加，期望能提升學生的能力與學習成效。

2.1.3 極致軟體編程與成對編程(pair programming)

極致軟體編程(extreme programming, XP)，是在軟體工程中敏捷軟體開發(Agile Software Development)的方法論的其中之一，XP 一共列出了 4 項範圍及 12 項核心的最佳實務做法，並且將之推向極限。使得敏捷軟體開發的四個價值觀：個人及相互交流勝過軟體工程和工具、可運行的軟體勝過完整廣博的文檔資料、與用戶合作勝過合約談判、回應變化勝過服從計畫，得以實踐，因此這個方法，廣受軟體產業中軟體開發團隊採用。

學生程式開發能力的教學與訓練，其實可視為是在軟體開發實務工作上的一小部分程序，因此 XP 所提出的最佳實務做法，也經常被引入教學中，發展出創新教學模式。D'Souza, & Rodrigues (2015) 參酌了 XP 的精神，應用在教學場域中而提出了以學生為中心的教學架構，稱為極致教育學(extreme pedagogy)，用以提升工程教育的品質，建構於生師互動、工作知識、學生協同作業和回應變化四個

核心價值上。極致教育學包含了三個特色：由連續性的實作中學習、由連續性的協同合作中學習、由連續性的測驗中學習。而協同合作學習，也採用了XP中成對編程(pair programming)的最佳實務，由兩個學生為一組，共同學習、解答問題。此外也提出，透過持續地評量學習成效，將有助於透過學生評量的回饋來支援學生學習，並且提高學習動機。

成對編程，在許多資訊相關課程中，也被融入做為教學的模式。Hanks et al., (2011)在回顧許多文獻後，歸納出成對編程應在資訊相關課程上有許多好處，可以提高學生修習入門課程的通過率、加深主要內容的記憶、提高軟體品質、提高學生解題的信心、改善學習成效。而這些好處，對於女同學會更顯著。但是實施成對編程最大的挑戰在於，時程的考量以及學習夥伴的合適性。

為了讓學生在課後也可以進行成對編程的學習法，Tsompanoudi et al., (2016)利用網頁化的程式套件(web-based plug-in)，結合了整合開發環境，實施了分散式成對編程，並於物件導向設計與程式課程中進行教學研究。學生可以不受地理位置限制，連上網頁來完成程式的作業，成對的學生之一擔任駕駛者的角色，負責編寫程式，另一位則擔任觀察者的角色，負責審查程式碼。兩人之間的溝通，可透過平台上的聊天功能來進行，並且提供程式碼標註的功能，有利於討論。研究結果指出，採用分散式的成對編程，提高了學生的課程通過率，並且也獲得了學生正向的回饋。

本研究將參考XP的方法論，引入成對編程的學習模式，同時於課堂中及課後實施。期望能解決學生獨自練習編碼實作時，苦於找不出bug而延宕的困境，課堂中也較容易跟上進度，此外透過同儕的相互激勵，也能更確實的完成課後程式作業，減少作業相互複製交差了事的敷衍情事產生，實踐合作學習策略來提升學習動機。

2.2 網頁設計教學相關文獻

2.2.1 網頁技術發展與課程改革

要開發出一個網站，涉及了前端介面設計、後端平台技術、資料庫、程式設計、網路與伺服器、雲端服務等多項議題，而使得「網頁設計」課程，在不同的科系中針對不同的學生背景，所涵蓋的範圍而有所不同。一些相關的教學研究，針對容易出現學習困難的程式編碼，來加以改良，例如常見的合作學習、成對編程，以及社會文化法(sociocultural approach)之協同編程(Faraon et al., 2020)。

對於在有限的教學時間內，如何盡量讓學生的學習能達到網頁設計所需具備的門檻知識(Threshold Concepts)或門檻技能(Threshold Skills)，翻轉教育為解決方案之一。Vellappan & Lim (2021)採用了翻轉教室，進行Html、CSS、JavaScript的教學。學生可以自己選擇空檔，在課前觀看教學動手做的示範影片，並跟著一步一步地親手操作這些基礎的網頁編碼。而在面對面的課堂中，則進行重點觀念的複習，以及分組方式進行進階編碼任務的練習。

然而，翻轉教室的成功與否，有賴於學生願意撥出時間於課前預習的學習紀律。在一些教學研究中，與翻轉教室息息相關的學習管理平台 (learning management system, LMS)，結合了行動化學習的概念，很適合被用來支援這類範圍廣泛的課程中。Tezer & Çimşir (2018) 使用行動版的 Moodle 平台於網頁設計課程中，學生可以透過學習平台的幫助，使用行動裝置來閱讀教材以及觀看解說的影片。研究結果顯示，與傳統教學使用紙本教科書的班級做比較，採用行動化 Moodle 平台的班級學生學習成效顯著優於傳統教學法，且可增加學生的興趣。Guo et al. (2021) 在網頁設計之課程設計，建議應強調動手做的能力，採應用導向和任務導向來設計教學活動，並透過 LMS 讓學生可於課前瀏覽線上教材、講解影片，而課堂中的時間則用於問題討論、典型網頁設計案例分析、實務操作、合作學習。此外未能在課程內提及的相關網路延伸教材，可透過 LMS 提供學生自主學習。

在行動化的概念延伸上，(Jin, 2017) 將課程聚焦於行動網頁的開發，也就是教程式初學者，設計單頁靜態之 RWD 網頁，讓學生能使用自己的手機，看到學習的成果，以激勵他們的學習動機，創造成就感。而在整合 RWD 開發技術框架 W3.CSS 的經驗，可強化問題解決能力，讓初學程式者能了解軟體的開發原理。研究的結果顯示，學生對於自我效能的信心、由網路資源自我學習 W3.CSS 的信心，皆是正向的。行動網頁設計的課程，也有學者融入專題導向式的學習 (project-based learning, PBL)，引領學生在課程中透過「做中學」引發學習動機、興趣，以及解決問題之能力 (蔡慧貞、任曉晶，2019)。

上述這些與課程教學實踐有關的研究，將做為本研究之參考。課程的範圍將設定為基礎 Html、CSS、JavaScript，以及 RWD 及現有開發框架和程式庫的使用。對於容易造成學習困難的程式部分，除了加強動手做的活動外，也將把程式編碼操作示範與講解錄製成影片，並將影片與延伸補充之線上教材一起放到 LMS 中，供學生隨時存取。

2.2.2 遊戲化之專題導向學習 (project-based learning)

將遊戲化的元素融入到教學活動中，可為原本枯燥乏味、互動性與參與度不高的教室學習氛圍，注入一股新的活力與吸引力，來提升學生的學習興趣。在網頁設計課程的實踐上，Song et al. (2017) 將遊戲化與群眾外包 (crowdsourcing)，融入了 CDIO 的教學模式中，將「Conceive (構思)」、「Design (設計)」、「Implement (實施)」和「Operate (操作)」四個階段對照到軟體工程的瀑布模型，設計出教學與學習的活動，並讓學生透過遊戲化專案團隊角色的扮演，如程式設計師、介面設計師、產品經理，融入到網站開發專案中。

Ying (2021) 在網頁設計與製作課程中，先進行 Html、CSS、JavaScript 等基礎理論知識的學習，並做為後續的網站建置課程之先備知識。後半部的課程則採用遊戲化來進行專題的教學，邀請業師扮演業界客戶，教師擔任公司經理，學生則分組扮演專案開發團隊，擔任專案經理、網頁設計師等，依客戶所提出的需求

來實作一個包含表單的網站。這個教學方法，提供了學生沉浸式的學習環境，藉由所扮演的角戶互動，激勵了學生的自主學習，並且增進了學生的組織及領導能力。透過遊戲的過程，也使得上課變得有趣，也更容易記住課程內容的重點，學習成效也顯著較實驗組佳。

送審人任教之網頁設計課程，修課學生為招收高中自然組之「資訊組」和社會組之「傳播組」各占一半所組成。一半自然組背景的學生對於設計方面的課程單元較不感興趣；另一半社會組背景的學生對於程式碼方面的課程單元較不感興趣。這情況非常適合導入遊戲化之專題導向學習，由教師扮演企業經理來引導專案團隊，而同學們可依專長扮演專案經理、介面設計師、程式設計師來組成團隊，由淺入深進行課程中的各項親手操作活動，最終建置一個完整的網站作為期末專題之成果。

2.2.3 引入真實世界專案需求之體驗式學習(experiential learning)

依據 David Kolb 所提出的體驗式學習模型(experiential learning cycle)，包含了主動實驗、具體的經驗、反射觀察、抽象概念化，這四個階段所構成的循環。為了從經驗中獲得真正的知識，學習者必須具備四種能力：1、學習者必須願意積極參與體驗；2、學習者必須能夠反思經驗；3、學習者必須具備並使用分析技能來概念化體驗；和 4、學習者必須具備決策和解決問題的技能，才能使用從經驗中獲得的新想法。

為了提升學生的學習興趣，充實與課程相連結的相關技能，並建立學習的成就感，Tzafilkou et al. (2020)大學部和碩士班的電子商務課程，並探討實施體驗式教學下，學生之感知易用性、有用性和接受度，以及和自我效能、學習意願及學習表現之關聯性。研究結果顯示，體驗式學習應用在資訊技術的課程中，可加深學生對課程任務活動的參與，建立起成就感。利用體驗式學習循環的框架，所設計動手做的活動能提升學生的學習興趣與能力。

服務學習是體驗式學習的形式之一，以學生領域專業知識的學習，與社區中的真實需求現況結合，提供了發展融滲式服務學習的機會。而在網頁設計課程方面，Lee et al. (2018)於課程中融入專業的服務學習，藉由服務學習對象所提出的真實世界需求，以小組的方式來建置滿足這些需求的網站作為課程的期末專題，實踐體驗式的學習。網頁設計與服務學習結合，除了幫助學生學習課程內容外，也幫助他們增進溝通和問題解決技巧，有助於未來職涯的發展。透過服務學習的參與，體驗式學習的反思階段是不可少的，可以讓學習變得有意義，並從經驗中建構知識。然而，也遇到了幾個挑戰：1、需要額外的小組討論時間；2、團隊合作上意見不一的困難；3、與真實客戶溝通的困難；4、受限於課程中所使用的網站建置工具(如：Wix、Weebly)無法滿足客戶之功能需求。

以滿足服務學習對象為需求的團體專案作為期末專題，開發社區中客戶所需的網站，融入於網頁設計的課程中，與目前國內在高等教育積極推動的大學社會責任亦可相互結合。楊婉秀(2020)在互動式網頁設計課程中，融入了協助彰化縣

社頭地區的織襪產業永續經營之社會責任教育，在課程中有織襪業者的參與提出真實世界的需求，修課的學生則以小組方式以所學到的網頁技術建構業者所需的網站。然而，這類課程的做法，需考量到業者的參與度，業者與學生溝通上的落差，以及對於學生作品的滿意度，以及學生需較傳統課程花費更多的時間。

本研究為了解決課程實踐時可能遭遇的挑戰並降低不確定性，將體驗式學習中的服務學習對象範圍，以校園取代了社區，提出改良式的校園公共服務學習之網頁設計課程，以校園中存在的，可模擬企業真實客戶為對象，開發所需的網站，並與本校之體育屬性作為特色。例如本校目前眾多的專項運動代表隊，需要類似中華職棒球隊官網來揭露資訊進行球團行銷及球迷互動。或是與其他課程的創新教學結合，例如：體育課程在教學創新上要求學生以某項運動主題編撰雜誌做為成果；本校奧林匹克教育的課程需辦理實體的成果展。原本這些課程的實體成果，可於本課程中以網站的形式呈現，做為校園內專業服務學習的對象。

基於上述相關文獻，本研究將採用動手做、體驗式學習、遊戲化之專題式學習等策略，對「網頁設計」課程進行改革。

2.3 資料庫教學相關文獻

2.3.1 資料庫教學改良之系統性回顧(systematic review)

Hamzah et al. (2019)為了解資料庫課程採用的教學方式和學生的學習困難，從學術期刊資料庫中檢索後篩選出相關的 35 篇論文進行系統性文獻回顧，歸納出了在實施傳統教學法的資料庫課程教學現場，學生反映出的問題主要有：1. 有關資料庫及相關程式語言同樣都非常抽象難以了解，以至於失去學習興趣與信心(Ying, 2016)。2. 教師無法個別關注學生間程度的落差與心理需求，以致在學習上產生沮喪與被動。3. 傳統以教師為中心或以教室為中心之教學，無法提供學生創新與主動學習的能量，缺乏不斷成長的創新力與創造力(Wang & Chen, 2014)。而教學上常採用的範例，不外乎是侷限於教科書所提供的圖書管理系統和學生管理系統，這使得學生變得被動且失去了學習熱情與創新性(Wang & Ma, 2017; Abid et. al., 2015)。而學生學習成效表現較差的學習單元，為 SQL 語法(Mason et. al., 2016; Folorunso & Akinwale, 2010)、資料正規化(Pošćić, et. al., 2012)、能滿足實務市場的需求水準之專題製作(Podeschi, 2016)。而為了改善上述實務現場之問題與情況，被採用的教學方法有：混成教學法(blended learning approach)、建構法(constructivism approach)、翻轉教室、認知學習理論、主動學習技巧、以電腦為基礎之教學等。

而對於資料庫教學中，許多學生深感困擾之 SQL 語法學習單元，Taipalus & Seppänen (2020)進行了 SQL 教育之系統性文獻回顧，從近 30 年發表的 89 篇研究論文中，分析研究的類型與主題，提出可分成六個議題(topic)以及六種研究類型(type)，並以此進行兩者之交叉分析。文獻回顧中共整理出涵蓋了 66 種的教

學方法用於 SQL 的教學上，而這些文獻的研究主題為：學生錯誤、練習資料庫、特定教學法、非特定教學法、樣式與視覺化、教學負荷。其中，練習資料庫的採用也屬一個重要的研究議題，有許多學者探討如何以視覺化方式呈現資料庫之綱要 (scheme)、解釋資料的需求、使用哪一類的商業領域資料等，可做為本研究預計採用真實棒球資料庫做為練習資料庫之參考。

2.3.2 結合學習輔助工具之教學方法改良

資料庫課程對於主修資訊的學生而言，學習的深度與廣度都較非主修資訊的學生之要求要高。因此在相關教學研究上，會更強調學生之學習成果能呈現到貼近實務水準要求的專案製作上。為了追蹤導入強調主動學習之專題式學習 (project-based learning) 之成效，有學者使用具專案管理流程之課程管理工具，進行五個學年實施成效之縱斷面比較研究 (Domínguez & Jaime, 2010)。前三年採傳統教學法，後兩年採主動學習。教學活動的資料都保存在課程管理系統中，以進行長期追蹤，並分析學生之考試退考率 (dropt rate)、通過率、得分和課堂出席率，來進行驗證。

而同樣在強調專題式的教學中，有學者建議可以依循專案開發的工作流程來教學 (Jain, 2017)，以解決傳統的資料庫課程過於太重理論而輕操作，以致難以提高學習興趣，且難以評量學生實務能力之問題，故強調實作的重要性。Cai 等人於教授 Oracle 資料庫之課程中 (Cai et. al., 2020)，倡導要進行專題實作，並依照工作流程來分階段安排專案實作任務，依序設計了：資料庫的分析與設計、資料庫安裝、建立資料庫、更新、查詢、備份之實作任務等學習單元，而實務操作練習的資料庫為學生透過各階段任務自行建立，並於期末完成一個「學生管理系統」之專題。

近年來興起之「翻轉教室」(flipped class) 也被用於美國密蘇里大學工程資訊技術學院的資料庫原理課程中 (Fang, 2017)，採用的是多維度混成學習翻轉教室模型。研究者指出在教學上遭遇到多數的實務操作，是簡單驗證的實驗，缺少設計與全面性。以及實作的時間不足，且內容相對簡單，與真實實務環境非常不同，難以符合工程資料庫應用之工業標準。研究者同時也整理出，資料庫課程教學在其他國家，如中國的大學 (Fang et. al., 2017)，普遍常見到三個教學現場的問題：1. 重理論而輕實務，實務性技巧的操作練習時數不足 (Rashid, T, 2015)。2. 擠滿式 (cramming model) 的教學，學生從粉筆加黑板轉變為電腦加投影布幕，學生在課堂上被灌輸了爆炸式的資訊，失去了學習熱情。3. 實務操作練習的案例與真實世界的實務環境差距過大。而這三個問題也是送審人於數年前初任教資料庫課程所遭遇的問題。

而在強化學生學習的學習成效上，Sakibayev et. al. (2019) 採取了行動學習的策略來強化學習。他們在哈薩克一所大學的資訊科學系資料庫課程中，使用行動科技來提升學生的學習成效。除了資料的概念的講解外，也在 android-based 手機

環境中使用 SQLite 進行行動學習，讓學生可隨身利用手機練習用 SQL 語法來新增、刪除資料庫及資料表、以及建立鍵值和索引，查詢與編輯資料。並設計了 9 個任務讓學生來完成，所使用的情境為：學生、教師、課程等資料的管理，以及個人 DVD 收藏、圖書編目、商品型錄，以及足球隊的基本資料與比賽的紀錄。

2.3.3 課程內容的變革與真實資料庫的使用

隨著資料庫技術應用之普及，在各個國家的大學中，資料庫課程也被列入非資訊電機相關科系系所的必修課或是通識課程，安排在大一的一門入門資訊課之後。因此，如何提升這些學生的學習興趣，成為教學上一個重要的研究議題(Lin, 2020)。有學者提出於教材中融入動畫，用來解說資料庫之技術和原理，提供非主修資訊的學生一個更容易接受和理解的課程內容(Dietrich, 2014)。

而在課程的設計上，教學單元順序的安排，也與資訊相關科系強調依專案工作流程安排有所不同。由於這些非主修資訊相關科系的學生對於抽象化概念接觸不多，直接透過具體地實例來觀摩學習，會是較能提升學習興趣的作法。Wang 和 Wang 在其所任教的大學中，提出主修商業的學生也應該需修讀資料庫課程(Wang & Wang, 2016)，而針對商學院所開設的資料庫課程，他們採用了逆向設計方法(“backward design” approach)來重新設計課程，也就是先看實例談操作，再談設計、需求和理論。Chen 在提供給資訊研究所新生，大學非主修資訊之先修課中，資料庫課程的模組設計也採用了與傳統相反的順序，由資料庫的操作、SQL 語法與操作，再進到資料庫設計之基礎理論，最後接續到資料庫程式設計(Chen, 2019)。

而對於資料庫課程教學單元的解構中，Mason et. al. (2016)認為對於非主修資訊的學生和初學者而言，傳統資料庫課程涵蓋的內容太過複雜，因此以認知負荷理論(cognitive load theory)為基礎，移除掉了容易產生注意力分散和重複效應(split attention and redundancy effect)的課程單元，並將內容重新排序，提供學生學習上適切的工作範本，並設定各學習任務之子目標。對資料庫的初學者而言，強調類似的工作範本會比強調問題解答的學習有效。而子目標的設定有助於資訊技術課程之學習表現。研究結果顯示，學生的學習成就和滿意度均有提升，尤其對於學習成就處在中後段的學生。

將課程內容解構，依傳統課程順序反向重新安排的革新思維，在捷克的帕拉茨基大學(Palacky University Olomouc)地理資訊系所開設資料庫系統課程中也被使用(Dobesova, 2016)，同時也強調採用一個實際的植物園 GIS 系統(BotanGIS)之完整資料庫做為案例來教學。這門課開在大二上學期，每年約有 30 位修課的學生。教師首先會帶學生去參觀學校的植物園和溫室，接著於電腦教室，讓學生於 BotanGIS 之 Web 平台上進行檢索，去找出一些植物的資訊。最後，再說明使用者的需求及分析，詳細解釋關聯式資料庫的模型。這種結合實際完整資料庫案例的教學方式，能吸引學生的注意，提高學習興趣與意願，且容易理解課程所傳達的知識內容。研究者也提到了，於傳統的資料庫教學課程中所採用的資料庫案例

大都從教科書來的，有：圖書館資料庫(書籍編目資料)、學校的學生與修課資料，商店的訂單與發票等。然而，若採用能讓學生親自體驗的真實案例資料庫，將更好解釋案例中的資料。這個研究也進行了使用真實案例之教學法之成效驗證，以問卷來蒐集學生對於使用真實案例的好處(滿意度、了解程度、清晰程度)，以及從學生考試成績之學習成就來評量。在期末考的表現上，學生對於有關真實案例的題目，正確率比與教科書案例有關的題目來的高。代表透過真實案例的講解比教科書案例的內容更容易記憶。而研究者同時也比較了四年的學生學習成績，第一年未採用真實案例，之後三年則有採用。採用後學習的成績有提升，且學生在學習中使用案例資料庫的比例也提高。

本研究將參考上述文獻，進行課程內容的解構與順序重組的變革，以及採用真實資料庫做為設計觀摩和實務操作教學之理念，來發展「從棒球學資料庫管理」之特色課程。

2.4 資料科學教學相關文獻

2.4.1 資料科學課程之教學實踐研究案例

與本研究研究主題相關之國內外文獻中，篩選出幾個有關「資料科學」課程設計及實務現場之代表性案例進行評析，依所實施的學生主修領域不同，分述如下。

一、於資訊相關科系之案例

Asamoah et al. (2015)參酌業界顧問的意見和自身資料分析研究團隊的經驗，發展了一套「資料科學導論」(introduction to data science)的課程，並於美國萊特州立大學(Wright State University)開設，教學的目標主要是培養數據科學家，包括了三個方面：

- 1、數學和統計知識：學生應熟悉一套工具，用於匯總、分類和對(非)結構化數據進行預測分析；能夠構建的基本模型分析數據；通過模式識別和知識發現技術獲得經驗，以建立數據關係；展示數據分析技術和模擬現象。
- 2、資訊技術：學生應該熟悉現代的開源編程語言，以進行數據分析；學習資料視覺化工具的使用，以深入了解數據關係；了解支持基本數據分析任務，包括提取，清洗，分類或預測以及解釋。
- 3、實務上的專業：學生應能提出潛在的分析方法在當今數據豐富的環境中解決實務上的問題；了解如何解釋分析結果；運用科學和商業原則進行分析和使用基於計算機的分析技術和應用程序解釋數據。

課程中共設計了8個於課堂中動手實驗的任務，分析的案例非常生活化，分別為股市報酬、通勤車輛、超市購物、博弈模擬、體適能、運動賽事、電影評論、程式犯罪報告，和一個期末小組專題。而不是個人書面報告和程式作業。課堂動手實驗的任務讓學生以小組合作學習的方式，在協作中學習資料科學工具的操作，

並讓學生接觸真實世界中的實務資料以實踐資料分析結果報告和解釋。每一個實驗室作業包括使用 R 或 Python 處理實際數據。而課程主要授課的對象，有各種不同主修學科的學生參與，為了推廣跨領域的學習，小組的成員至少有兩名需來自不同主修領域，藉助學生對於特定領域實務的認知來深度解讀資料，共同完成期末的專題。而實際上在實施時，共有 14 位學生修讀，其中主修資訊管理的學生 4 位，電腦科學 9 位，1 位是非資訊相關科系。

他們所設計發展的課程，其品質最後是以跨領域小組成員間協同合作達成的程度之學習經驗，和個人對於學習成果與滿意度來進行衡量，由修過課的學生進行問卷調查。調查的結果顯示，在小組協同部份，學生對這門課程跨領域的特性、共同學習共同教學的模式感到非常滿意，且在學習經驗方面呈現出正向的態度。而在學習成果部份則反映出學生對於資料科學之知識與技能自認可以達到精通的程度。

二、於非資訊相關科系之案例

Chung(2018)認為應鼓勵非資訊相關科系的學生，主動積極參與學習資料分析。為了達到這個目標，他以醫學臨床學習常用的實證實務(evidence-based practice, EBP)來引導，並以學生為導向，開發設計了一個暑期短期共 10 週的課程，以工作坊的形式於美國南伊利諾州大學開課，提供非資訊科系的學生修讀。

課程的實施，包含了 EBP 的研討，以網路建立點對點的支持和讀書會來進行社交活動。而教材則採用主動學習，將教材放置於學習管理系統上讓學生在課前先進行閱讀，並準備了五個測驗題讓學生作答，而課堂上則是依據學生作答的情況，加以強化解決學習的困惑，確保關鍵概念的建立，課堂的最後則讓學生進行數據分析的實驗，以 R 語言進程式編碼進行實作。而課程學習成果的驗證，以學生是否能運用所學得的知識與技能，解答主修領域中的問題。以最後兩週的時間，完成團隊專題。

研究結果顯示，以 EBP 支持科學實證有助於發展應用資料分析的課程。同時採用主動學習法(active learning)使得學生在教師因資源有限而較少介入的情況下，仍可使學習同儕凝聚在一起。而課程也激勵了非資訊科系學生的參與，藉著影響學生去尋找在他所屬的領域中如何應用資料分析。最後課程的專題實作使得學生得以體驗大學階段的研究。

三、於體育相關科系之案例

Barneva & Brimkov (2019)於美國紐約州立大學(State University of New York)中，針對運動管理系所大學部的學生，開設了「資料分析」的課程。修課的學生僅需具備高中數學的程度，且大學修過 Excel、Access 軟體應用及基礎統計學，且不要求學生修過程式設計的課程。

他們認為主修體育的學生，通常都對於計量(quantitative)與推理(reasoning)的課程感到困難，且偏好動手實作、服務學習、專題導向、實習的課程，而不是典型的講授課程。與計量有關的課程，大都是在通識教育中修習，且大都僅到達入門的程度。而對於資料科學家不可或缺的 Python 和 R 等程式知識，大都數體育

相關科系的學生都沒有修過相關的課程。因此，「資料科學」的課程在體育相關科系中實施的難度，往往比電腦科學或數學相關科系來的高。此外，「資料科學」的課程，須於電腦教室中實施，以提供學生電腦動手實作，而不僅是講授理論。

課程的內容包含了資料的蒐集、資料視覺化、各種決策分析的方法，以及如何使用軟體去衡量運動的表現，競技體育如何透過資料分析來獲益等。並使學生能利用資料科學解決運動管理中常見的問題，如：預測隊伍的表現、制定獲勝策略、球員的分析、球員或隊伍的排名等等。課程的進行，是以專題導向的方式來進行，分階段一步一步地在作業中提出如何應用資料分析於某個運動項目，以提升表現、創造收益、預測趨勢、引導進步，最後期末完成一整個專業的正式報告。教材的部分，因「資料科學」涵蓋範圍很廣，尚未有合適的教科書，因此捨棄傳統的教科書，而採用免費的開放教育資源(Open Education Resource, OER)，包括免費的線上教學課程、免費開放書籍、短片等。教師花費許多時間整理了適合教學的資源，讓學生在家中事先閱讀並完成指派的進度，而後於課堂中討論。為了克服學生在程式設計方面能力之不足，在軟體部份採用了 Knime 和 Excel 的 Solver 附加套件，使學生能使用軟體解答線性與非線性的問題，求解最佳化問題，且進一步可使用機器學習的模型，來找出致勝策略。

這樣的課程設計，經實施後驗證是非常可行的，最後修課的學生均能在運動的案例探討中，利用資料分析的模型來完成不同的階段任務，並且在獨立個人的專題中，能使用有關的分析模型，以動手做的方式實際進行運動資料分析後提出建議，最後完成專業的書面報告與公開演說發表。

2.4.2 漸進釋放責任(gradual release of responsibility)

漸進釋放責任(gradual release of responsibility, GRR) 的教學設計(Fisher & Frey, 2008)，本質上是鷹架學習法的一種，在學習的過程中，學生可以依照所掌握的經驗逐步地加深學習，同時在過程中提供回饋支持。在漸進釋放責任與校準的學習過程中，可提升學生的參與度與自我效能感(Ortlieb & Schatz, 2020)。GRR 方法有兩項關鍵的特徵，強調教師指導學生與學生依賴教師兩者間的互動性，以及學習任務的難度逐漸增加。學生一開始的學習活動，是建立在教師引導上，而隨著學習任務越來越難，教師的依賴與明確指導的支持也越深。而後逐漸發展到學生不再需要教師的協助，即可完成任務，到最後能完全獨立，完成自己想做的任務。GRR 模型由四個互動的部分組成，包括了重點課程、引導式教學、合作學習、獨立作業(Fisher & Frey, 2008)。而在實務上，這四部分分別被對應”我做 (I do)”、“我們做 (We do)”、“你們一起做 (You do it together)”、“你單獨做 (You do it alone)”四個簡單且廣受歡迎的口號(Archer & Hughes, 2010; Hollingsworth & Ybarra, 2017)。GRR 已被應用於廣泛的教育情境中，包含了閱讀理解力(Webb et al., 2019; Duffy, 2014)、引導寫作(Fisher & Frey, 2008)和數學上(Saligumba & Tan, 2018)，以及針對主修運動科學學生的程式設計課程教學(Chuang et al., 2022)。

2.4.3 開放教育資源(open educational resource)

廣義的開放教育資源(open educational resource, OER),除了傳統所指的開放式課程 (open courseware, OCW) 與大規模開放線上課程 (massive open online courses, MOOCs) 外,也涵蓋了部落格、官方網站、影片、軟體庫等。隨著自媒體的發展,越來越多的內容創作者於個人(如:medium.com)及 YouTube 等網站分享具專業化、有參考價值的內容,近年來許多課程也善用了開放教育資源進行教學。資料科學課程是一個新興的教學領域,教科書的選擇不多且需隨著技術變化的速度而經常更新,常見的數據科學、機器學習、數據挖掘等教科書,往往過於技術性及進階,不適合運動科系的學生。而網路上可公開存取的開放教育資源,是另一個不錯的選擇。使用這類新形態的開放教育資源教學時,學生的學習方式與傳統不同。以專注軟體版本控制和原始碼代管服務的平台 GitHub (<https://github.com>),以及數據建模和數據分析競賽的平台 Kaggle (<https://www.kaggle.com>)為例,其學習方式是把資料下載,建立程式碼副本,然後換成自己實務的問題,類似建築師利用計算機輔助設計軟體從開放式建築繪圖數據網絡中下載案例來學習一樣(Downes, 2019)。這類開放教育資源提供了許多程式資源庫,網頁中伴隨著程式碼可利用 markdown 標記式語言加入對應的文字解說,清楚瞭解大數據分析步驟,並可讓使用者複製並操作程式,以自己的數據集和目的來實作。Kaggle 則提供許多有趣的分析競賽和數據集,其中不乏常在教科書出現的鐵達尼號逃生(Tabbakh et al., 2021)、樂高玩具的偏好(Bahad et al., 2021)、冰山偵測 (Yang & Ding, 2020)等案例。而與運動相關的有足球勝負預測 (Carpita & Golia, 2021)、籃球勝負預測(Thabtah et al., 2019)等案例,非常適合用於與體育運動專業知識結合之教學。

本研究以修課對象為體育運動領域學生的運動資料科學分析課程,進行課程變革與實踐的行動研究,採用漸進釋放責任(gradual release of responsibility, GRR)的教學設計(Fisher & Frey, 2008),並參考前述文獻之課程實踐經驗,融入與領域專業核心融合、專題導向學習、做中學、利用開放教育資源等策略。

3. 教學設計與研究方法

3.1 運用 ADRI 教學模式結合成對編程與線上評測之程式設計課程

3.1.1 教學規劃與課程設計

本研究程式設計的教學，程式語言將選用 Python，可降低程式語法和語意的難度，並有利於未來人工智慧與大數據分析相關課程之接軌。在學生核心能力的部分，所設定的教學目標為：一、具備團隊合作精神，利用運算思維分析、解決問題之能力；二、熟悉程式設計之語法與語意，具備獨立程式編碼能力；三、解決實際問題的程式設計能力。而為了提升學生學習成效，針對上述之教學目標，擬訂教學改善之活動。

教學方式採用的是 ADRI 教學模式，於大一新生之程式設計課程中，進行兩學期，各 18 週的教學。教材除了採用系統性知識體系的單一書籍外，增加了自編之 ADRI 程式範例教材，並輔以線上學習協作平台、自動化程式線上評測平台等教學資源之導入等。教材選用、教學資源應用、評量方式等，說明如表 2。具體的課程進度、課程主題、課程內容、教學方式與活動，整理如表 3 所示。

表 2 程式設計教學範疇

課程範疇	課程名稱	基礎程式設計	進階程式設計
	開課學期	大一上	大一下
	選修別	系選修	系分組必修
教材選用	選用程式語言	以 Python 語言進行教學	
	教科書	自編講義、選用書籍、線上教學網站、TQC+ Python 基礎程式設計認證題庫	
	程式教學範例、課堂實作、課後作業	依 ADRI 模式，挑選適合之程式範例設計教材	
教學資源應用	1. 學習及協作平台：學校之 Moodle 學習平台、Github、微軟 Azure Notebook 等。 2. 程式自學網站：如 Code.org、Codecademy、Free Code Camp、Khan Academy、Udemy 等線上學習資源。		

	3. 程式自動評量平台：Code Judger。	
教學目標	1. 具備團隊合作精神，利用運算思維分析、解決問題之能力。 2. 熟悉程式設計之語法與語意，具備獨立程式編碼能力。 3. 解決實際問題的程式設計能力。	
教學方法	1. 講述法 2. 動手實作 3. 利用程式自動評測平台強化練習 4. 兩人一組進行成對編程，完成作業與專題。	
評量方式採用	1. 評測平台之程式練習情形。 2. 程式作業。 3. 程式設計上機實測。	1. 評測平台之程式練習情形 2. 程式作業 3. 程式設計上機實測。 4. 與科系發展特色相關之專題製作。

表 3 程式設計課程設計

基礎程式設計			
週次	課程主題	內容說明	教學方式/活動
1	課程介紹	說明課程的內容、進度、上課方式、學習資源、評分要求等	1. 採用講述法，快速系統化地介紹 Python 之基本語法。 2. 實作以課本範例練習為主。 3. 兩人一組進行成對編程，同儕合作學習。
2	Python 開發環境建置	介紹 Anaconda、jupyter notebook 之安裝與操作	
3	程式設計基本語法	變數、常數、資料型態、運算式、運算子	
4	基本輸入輸出	輸入指令、格式化輸出說明	
5	程式的流程控制	IF 敘述	
6	程式的流程控制	迴圈	
7	資料型態	串列(List)	
8	資料型態	元組(Tuple)、字典(Dict)	
9	期中考	課本重要範例測驗	
10	TQC+認證考試解題實務教學	基本程式設計	1. 採做中學，逐題一步一步引導講解例題後，由學生自行類提之解題。
11		選擇敘述	
12			
13			

14		迴圈敘述	2. 依 ADRI 之教學模式，編製教學講義。 3. 使用 code judger 評測平台。
15			
16		綜合演練	
17			
18	期末考	認證題庫測驗	上機考
進階程式設計			
週次	課程主題	內容說明	教學方式/活動
1	課程介紹	說明課程的內容、進度、上課方式、學習資源、評分要求等	1. 採做中學，逐題一步一步引導講解例題後，由學生自行類提之解題。 2. 依 ADRI 之教學模式，編製教學講義。 3. 使用 code judger 評測平台。
2	TQC+認證考試解題實務教學	進階控制流程	
3			
4			
5		函式 (Function)	
6			
7			
8		綜合演練	
9	期中考	TQC+認證考試	上機考試
10	第三方模組及套件安裝	pip install、import、GitHub 介紹	1. 採用講述法，快速系統化地介紹 Python 之基本語法。 2. 實作以課本範例練習為主。 3. 兩人一組進行成對編程，同儕合作學習。
11	陣列資料處理	Numpy	
12	繪圖套件	Matplotlib	
13	科學計算套件	Scipy	
14	圖形使用者介面	tkinter	
15	影像和視訊處理	OpenCV	
16	網路爬蟲框架	Scrapy	
17	專題製作規劃	範例參考、題目定義、構想規劃	
18	期末專題	擇一應用，進行小型專題開發。	報告及展示

本研究配合兩個學期之課程，共分成四個階段實施，以每學期第 9 週期中考

做為分界，說明如表 4:

表 4 四階段之教學過程與內容

階段	期間	配合課程	修課人數	教材	研究活動
一	108 上 1~9 週	基礎程式	24	教科書:Python 初學特訓班 [基礎篇]	學習評量(期中考)
二	108 上 10~18 週	設計		TQC+基礎程式設計題庫偶數題、自動評測平台	學習評量(期末考)、問卷、自動評測平台使用統計
三	108 下 1~9 週	進階程式設計	20	TQC+基礎程式設計題庫奇數題搭配流程圖、自動評測平台、認證考試	學習評量(期中考)、自動評測平台使用統計、質性訪談
四	108 下 10~18 週			教科書:Python 初學特訓班 [應用篇]	問卷

實施教學的第一階段，採用的傳統的講述法，針對教科書上系統化的程式基本語法，進行講解與範例操作，目的是要讓學生在短時間內快速了解程式設計的內涵。在教學階段二時，開始引入了 Code Judge 平台(如圖 3)，做為自動化程式線上評測系統，並搭配中華民國電腦技能基金會所開發的 TQC 程式設計的題庫，來進行教學。學生可以上課中及上課後，於平台中進行練習與評測，並會記錄練習的狀況。而在教學階段二和三，同時實施改良 ADRI 之教學模式，引入了流程圖來加強解說邏輯判別式、迴圈等，這些學生反應比較難的教學單元，並以學校的 Moodle 教學平台(如圖 4)來輔助教學，將相關的流程圖、程式碼、及部分上課錄製的影音解說置放於教學平台上。

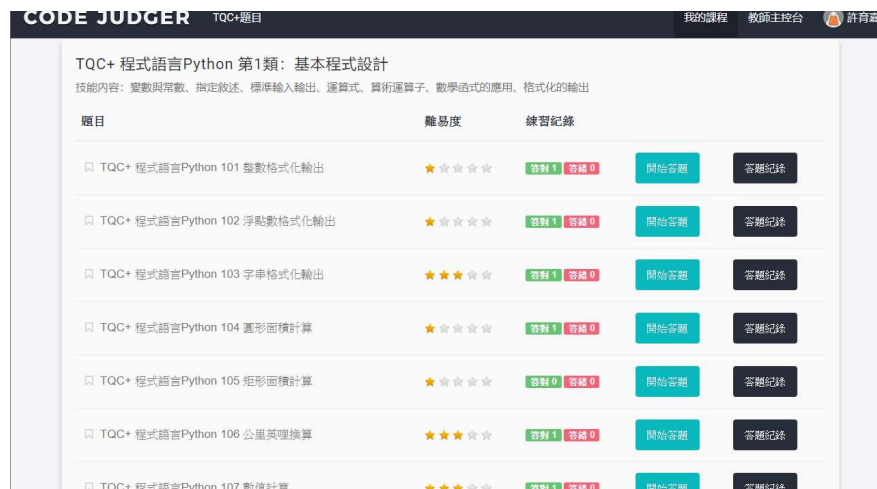


圖 3 Code Judger 平台中的 TQC 程式設計題庫

The screenshot displays a Moodle course page for '108-2-運傳1A-進階程式設計'. The main content area features a flowchart titled '506 一元二次方程式解流程圖' (506 Quadratic Equation Solution Flowchart). The flowchart starts with '開始' (Start), followed by '輸入係數 a, b, c' (Input coefficients a, b, c), then '判斷式: d=b^2-4*a*c' (Decision: d=b^2-4*a*c). A decision diamond checks 'd < 0'. If '否' (No), it leads to 'x1=無解, x2=無解' (x1=No solution, x2=No solution). If '是' (Yes), it leads to 'x1=(-b+d*0.5)/(2*a), x2=(-b-d*0.5)/(2*a)' (x1=(-b+d*0.5)/(2*a), x2=(-b-d*0.5)/(2*a)). Both paths lead to '輸出解 x1, x2' (Output solutions x1, x2) and finally '結束' (End).

Below the flowchart is a code editor showing Python code for solving quadratic equations:

```

12 # return None
13 return "Your equation has no root."
14
15 elif delta == 0:
16     return -b / (2 * a)
17 else:
18     res1 = (-b + delta ** 0.5) / (2 * a)
19     res2 = (-b - delta ** 0.5) / (2 * a)
20     return str(res1) + ", " + str(res2)
21
22 a = eval(input())
23 b = eval(input())
24 c = eval(input())
25 result = compute(a, b, c)
26 print(result)
27

```

The interface also shows a course overview on the left with dates and activity counts, and a sidebar with navigation options like '課程資訊', '學生名單', and '作業測驗'.

圖 4 輔助教學使用之 Moodle 平台

3.1.2 研究架構與程序

整體研究架構設計，如圖 5 所示，共包含四個部分：教學內容設計、課堂活動、合作學習與回饋反思，如圖六所示。第一部分為教師依 ADRI 模式，將解答問題能力與程式語法語意兩者並重之教學策略，融合在一起規劃教學內容。第二部分課程實作活動，為配合課程進度所實施之課堂程式實作及課後程式作業，利用教學平台提供自動評分之功能，讓學生能確實並多加練習，並透過平台之互動溝通功能，解答程式除錯的問題。第三部分合作學習希望透過 XP 的軟體開發模型成對編程的概念，採用合作學習的策略，讓學生兩個人一組相互激勵、一起腦力激盪、增加互動、共同解決問題。第四部份反思與回饋，透過教學反思日誌、學習回饋單、問卷與訪談的方式、歸納分析問題與建議，並對課程進行修正。

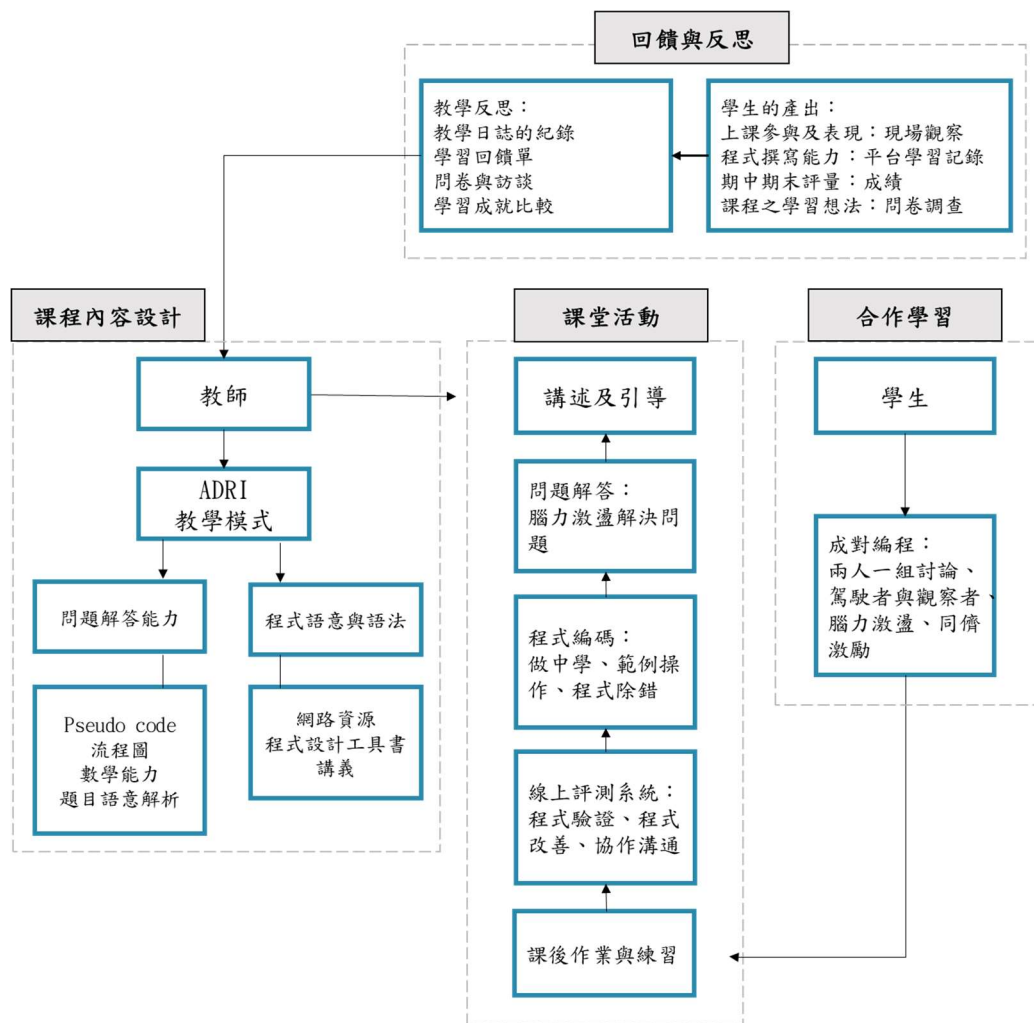


圖 5 研究架構

研究進行的程序，說明如下：

一、教學資料收集與教材編製

研究者依據教學主題與單元目標，廣泛蒐集程式設計的各种題目，由目前出版社所出版的程式設計書籍中，挑選與改編為適合研究對象於課堂實作與課後作業，並編製教學講義。本研究依 ADRI 模型，來發展程式設計教材，內容包括四個部分，首先讓學生以 psuedo-code 或流程圖為工具，來解決問題；其次是程式語言及語意的說明；接著讓學生依據程式的輸入、輸出要求進行編碼，並程式測試；最後，要求學生以其他的程式語法，重新解答同樣的問題。

二、網路教學平台之設置

研究中將採用學校現有之 Moodle 平台，做為教材下載、問卷等教學輔助使用。此外，針對程式設計部分，則採用外部的工具及平台，來提供學習參考資源、自動化程式線上評測、程式碼標註與討論、生師互動等功能，目前已有 repl.it、code.org、Microsoft Azure Notebook、Code Judger、UVa Online Judge、URI Online

Judge、Code Marshal Online Judge、Spoz Online Judge、GitHub 等平台，但各項功能支援的情形不一，需一一測試後選擇最合適的，或個別使用單一功能，做為本研究使用。當決定採用之教學工具平台後，將相關教材、範例程式碼等，建置於平台上。

三、整合教學活動設計與教學平台

本研究在程式編碼時，將採用兩個人一組的成對編碼方式，來實踐合作學習的理念。而在教學實務上，分組實作、課後作業、專題開發之進行，需與教學平台的功能整合。

四、實施教學

本研究預計實施教學的時間，配合計畫執行的時間一年，從 108 學年度第一學期開始，一直到學期結束，連續實施於兩個學期的連貫課程。約每 2 到 3 週為一個教學單元，每個單元均包含問題解決的教學、程式的語法及語意、課堂實作練習、課後程式作業。每階段進行一個單元的教學及作業，每階段完成後則進行反省與修正，再進入下一階段，如此循環。而在第 9 週及第 18 週時，則為成就測驗的階段，實施期中與期末考試。於各階段教學結束後，再依據相關資料進行研究。

五、教學回饋

教學過程中，蒐集學生的課堂練習作業、課後作業、回饋單、教學平台之互動內容、學生訪談記錄、問卷填寫結果、以及研究者之教學省思日誌等，做為教學回饋。

六、修正教學設計

依據上一階段之教學回饋呈現之缺失或不足處，調整各部分講解與練習的時間、教材的難易度、練習與作業的頻率等，修正後於下一階段的教學活動實施。

七、資料整理與分析

於每階段完成後之教學回饋資料進行彙整，待學期結束後，針對所蒐集的多方資料進行統計分析、統整，最後進行詮釋與分析。

八、提出結論與建議

綜整兩個學期循環的資料，進行整理並歸納出結論，對結果提出相關建議。

3.2 融入體驗式學習與遊戲化專題導向學習之網頁設計課程

3.2.1 教學規劃與課程設計

為解決學生學習動機、興趣與成效等問題，送審人依據任教科系之學生特性與就業發展，規劃出教學主題與單元目標。再從相關的電腦書籍、開發技術之官網、網路開放教學資源中，廣泛蒐集網頁設計之案例素材，經挑選並改編為適合於課堂有限時間內實作的規模，編製成基礎能力、進階能力、實務能力三個部分的教材，和課堂動手做活動之任務。

研究中採用學校現有之 Moodle 平台，做為教材建置、問卷等教學輔助使用。針對課程實務實作部分，將採用免費授權的 Visual Studio Code 或是 Sublime Text 為主要開發工具，並補充 BlueGriffon、Dreamweaver 這類所見即所得(WYSIWYG)的網頁編輯器。這些工具需配合教材與活動設計，一一測試後挑選出要講解的操作部份。

教學設計之概念圖如圖 6 所示，融入了「體驗式學習」與「遊戲化專題導向學習」之教學策略，前半個學期(第 1 週到第 9 週)，依規劃之教學單元主題，採漸進式地由淺入深依序安排教學順序。而後半個學期於第 10 週，引入校內專業服務學習的真實需求，邀請服務學習的對象說明其網站建置之需求，做為期末專題的主題。學生組成專案團隊，以遊戲化方式扮演不同的專案角色，並鼓勵主動學習延伸的相關網頁技術，完成期末專題的開發。

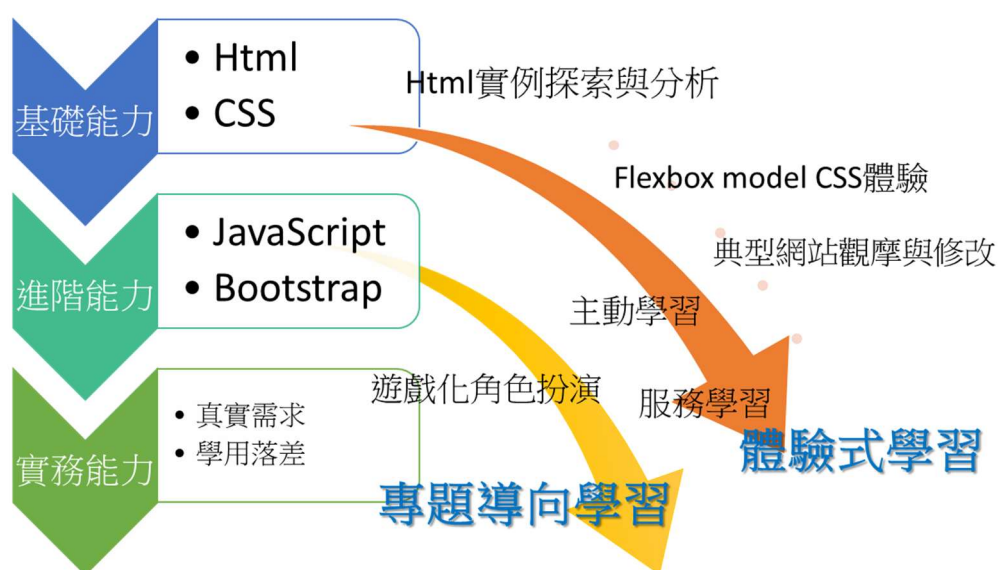


圖 6 教學設計與規劃

設計的課程，為一個學期共 18 週，每週 3 學時之課程，為運動資訊與傳播學系之大二全體學生必修。課程的實踐將驗證課程設計，是否能符合學生的學習需求，能夠提升學生的學習動機，使學生達到一定程度的學習成效，並滿足學生的學習成就。課程實施的同時將進行行動研究，以了解學生之學習情形及回饋，以做為課程設計改進之參考。課程範疇、教材選用、教學資源應用、評量方式等，說明如表 5。具體的課程進度、課程主題、課程內容、教學方式與活動，整理如

表 6 所示。

表 5 網頁設計教學範疇

課程 範疇	課程名稱	網頁設計
	開課學期	大二上
	選修別	專業必修(資訊組及傳播組共同必修)
教材 選用	工具使用	Visual Studio Code/Sublime Text、線上網站建置器 (Wix、Weebly)、Dreamweaver。
	教科書	自編講義、線上教學網站、網路開放教育資源
	實驗、實作、作業	課中於電腦教室中讓學生動手操作，完成指定任務。期末專題，則於課後時間完成。
教學資源應用		<ol style="list-style-type: none"> 1. 學習管理平台：學校之 Moodle 平台。 2. 線上教學資源：W3Schools.com、getbootstrap.com、Github.com。
教學目標		<ol style="list-style-type: none"> 1. 能瞭解網頁及網站背後之運作原理，並看得懂網頁原始碼，具編修之能力。 2. 能熟悉網頁製作工具之實務操作，具備技術框架、程式庫之應用能力。 3. 技術整合能力、溝通能力、責任感、自信心之建立。 4. 具備網站建置、網頁技術應用之規劃能力，以滿足網路行銷、網路傳播之職場需求。
教學方法		<ol style="list-style-type: none"> 1. 體驗式學習：範例解說，引導修改範例進行操作，觀察試驗結果，摘要結果並記錄心得。 2. 遊戲化專題導向學習：模擬真實環境中，公司經理、專案經理、網頁美工、程式設計師、業主等角色扮演。並藉由一步一步完成網站建置專題的過程中，引導延伸學習能力不足之處，以克服網站建置過程中的技術問題。 3. 主動學習：延伸教材建構於 Moodle 學習管理平台上，補足專題製作所需能力之落差。
評量方式採用		<ol style="list-style-type: none"> 1. 課堂活動動手做實務操作: 30% 2. 學習評量測驗(期中考筆試): 30% 3. 期末專題製作(書面及上台報告): 30% 4. 團隊合作、主動學習、課堂參與: 10%

表 6 網頁設計課程設計

週次	課程主題	內容說明	教學方法/活動
1	課程介紹	說明課程的內容、進度、上課方式、學習資源、評分要求等。	
2	網頁原始碼實例探索、網頁運作原理	使用 Chrome 之開發工具，探索幾個實際案例網站之原始碼，了解組成網頁之各個成分及相關語言。網站伺服器運作原理。	【課堂實作】探討網頁之組成
3	Html DOM 之階層式架構與結構化摘要	標記式語言介紹、樹狀結構探索、Html 的組成、DOM 方法與屬性。	【課堂實作】利用記事本製作第一個網頁
4	HTML 基礎語法	網站伺服器運作原理、與圖文有關的 Html 常用語法、Html Editor	【課堂實作】圖片及超連結製作：絕對路徑、相對路徑
5	Html 進階語法	表單、多媒體、RWD 等進階語法	【課堂實作】影像地圖網頁製作
6	CSS 基礎介紹	CSS 運作原理與基本語法	
7	CSS 選擇器	Flexbox model 與 CSS 之變換	【課堂實作】利用 CSS 進行網頁外觀快速切換
8	WYSIWYG 編輯器介紹	Dreamweaver/BlueGriffon 之 Html+CSS 操作	【課堂實作】操作 Dreamweaver/BlueGriffon，運用 CSS 美化頁面
9	期中考	檢核 Html、CSS 之基本之知識	
10	期末專題起始會議	真實網站設計需求說明、專題進行方式說明。邀請運動教練、數位展覽等需求單位共同參與，模擬招商。	
11	程式開發工具介紹	Visual Studio Code、Sublime Text 之使用	
12	JavaScript、jQuery	基礎原理、常用範例介紹	
13	W3.CSS、Bootstrap	基礎原理、常用範例介紹	
14	網站規劃與專題進度報告	報告專題訪談結果，以及初步網站架構規劃	
15	網站建置器、網站範本、經典網站案例說明	Wix、Weebly 簡介、免費 CSS、RWD 範本、10 大設計網站觀摩	【課堂實作】下載網站範本，快速建置網站

16	網頁伺服器與雲端服務	IIS/Apache 伺服器介紹、FTP、虛擬主機、Azure/AWS 雲端服務	
17	網站技術與開發實務	當今大型網站開發採用之前後端技術與工具介紹	【業師協同教學】
18	專題成果報告與展示	邀請原需求單位參與成果發表，共同講評。並模擬業主驗收。	

3.2.2 研究架構與程序

研究架構共包含四個部分：課程設計、課堂活動、服務學習、回饋反思，如圖 7 所示。第一部分為教師參照相關文獻之教學策略，以學生特性及必備的網頁技術素養培養來進行教材範圍的取捨，決定教學的主題和重點。第二部分為課堂活動，為體驗式學習之親身動手做活動之設計思考，以 Html、CSS、JavaScript、Bootstrap 等網頁技術做為所要傳達知識領域的核心，兩者融合在一起規劃動手實作的活動，以加深學習的成效。第三部分服務學習則是為了引入真實世界的專案需求，培養學生自我主動學習以解決問題之能力，並訓練溝通能力與責任感，及建立自信心。第四部份反思與回饋，透過教學反思日誌、學習回饋單、問卷與訪談的方式、歸納分析問題與建議，並對課程進行修正。

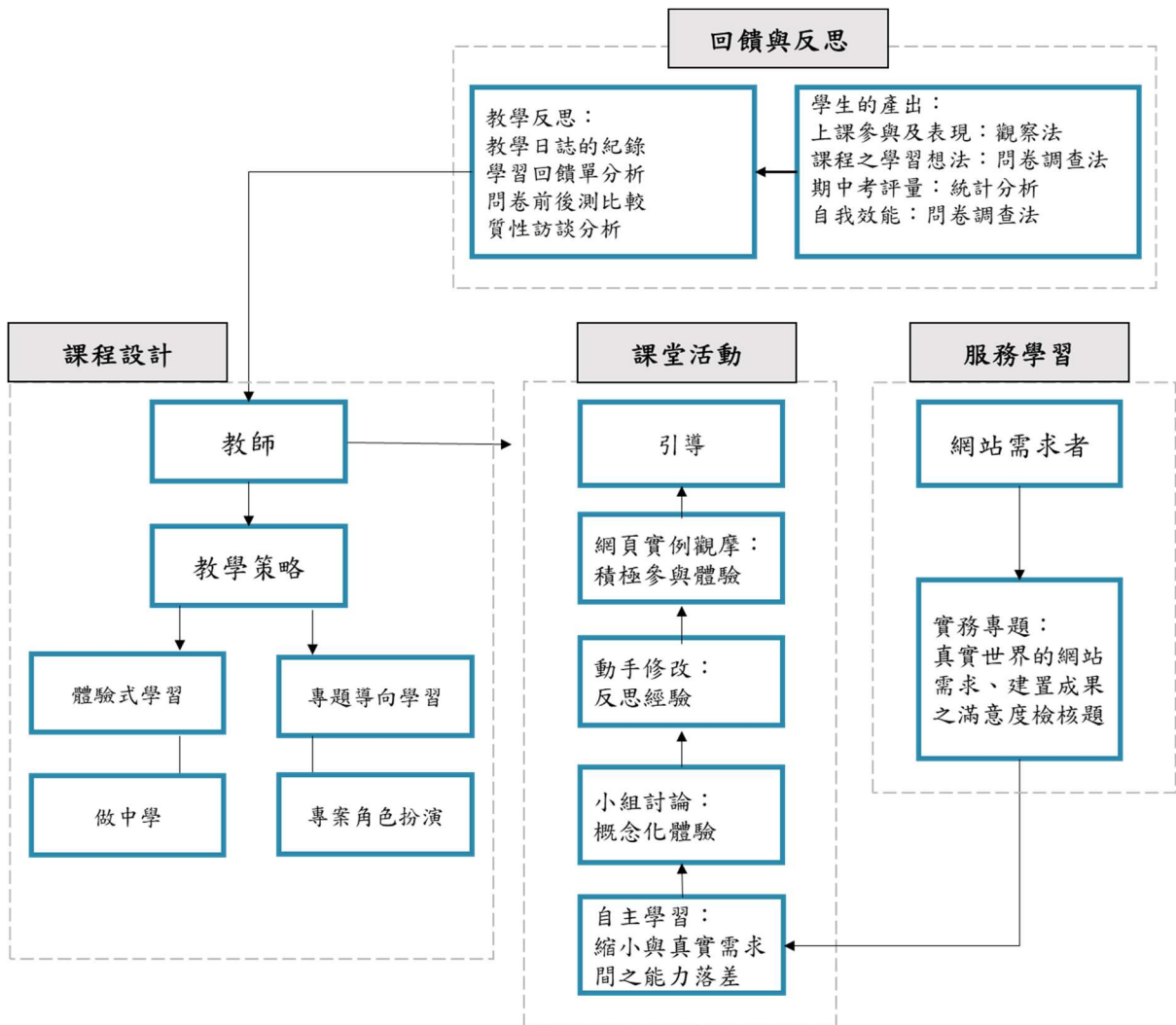


圖 7 研究架構

3.3 從棒球學資料庫管理

3.3.1 教學規劃與課程設計

本研究設計的課程，旨在介紹資料庫管理系統之技術及用途，如何設計資料庫來記錄資料，以及查詢、分析在資料庫中的資料。課程中將以棒球做為情境引導之場景，融入各種熱門與數據有關之棒球話題，再搭配真實之 Lahman 資料庫進行實務案例之觀摩與操作，使學生能具備運用資料庫技術，來解決實務上問題之能力。研究目的是要發展出適合運動領域學生之資料庫課程教材，提升學生之學習成效，使得能利用資料庫技術的學生比例增加，建立非資訊科系學生對於資料庫應用方面之自信心，並養成自我主動學習之習慣。

此一個學期共 18 週的課程，提供主修運動資訊與傳播學系之大二學生修讀。課程實施的同時將進行行動研究，以了解學生之學習情形及回饋，以做為課程設

計改進之參考。研究擬投入之課程範疇、教材選用、教學資源應用、評量方式等，說明如表 7。具體的課程進度、課程主題、課程內容、教學方式與活動，整理如表 8 所示。

表 7 資料庫管理教學範疇

課程 範疇	課程名稱	資料庫管理
	開課學期	大二上
	選修別	系分組必選修
教材 選用	工具使用	MS SQL、SQLite
	教科書	自編講義、線上教學網站、網路開放教育資源
	實驗、實 作、作業	課中於電腦教室中讓學生動手操作，完成指定任務。課後則由學生透過智慧型手機行動學習。
教學資源應用		<ol style="list-style-type: none"> 1. 學習管理平台：學校之 Moodle 平台。 2. 棒球網站：MLB.com, Baseball-reference.com, Retrosheet.org, ESPN.com。
教學目標		<ol style="list-style-type: none"> 1. 透過實作對資料庫有基礎之認識，已具備資料庫之實務操作能力。 2. 能利用資料庫技術解決競賽成績、運動訓練、競賽策略、運動表現與績效記錄等問題。
教學方法		<ol style="list-style-type: none"> 1. 講述法：資料庫理論與技術解說。 2. 主動學習(Active Learning)：教材建構於學習管理平台(LMS)上，請學生依進度研讀。 3. 實務實作(Learning by Doing)：真實資料庫範例解說與操作，並於課堂完成指定操作任務，課後以行動學習加強練習。 4. 合作學習：以小組方式進行資料庫之設計。 5. 專題導向式學習(Project-based Learning)：引導學生利用各課堂之操作任務，綜整後完成期末專題研究。
評量方式採用		<ol style="list-style-type: none"> 1. 各主題課堂任務實作成果:30% 2. 期中考: 40% 3. 期末專題研究(書面及上台報告): 30%

表 8 網頁設計課程設計

週次	課程主題	內容說明	教學方法/活動
1	課程介紹	說明課程的內容、進度、上課方式、學習資源、評分要求等。	
2	棒球數據之來源與網站	介紹國內外之棒球網站，如	【體驗式】 資料庫

	查詢簡介	Baseball-Reference.com、了解資料庫之應用。	於網站介面之呈現
3	Lahman 資料庫及其架構介紹	觀摩資料庫之架構，說明如何下載與安裝。	【體驗式】資料庫之安裝、備份、還原之實務操作
4	Lahman 資料庫之安裝與備份還原	介紹 MS SQL 資料庫之安裝及各種備份與還原方式。	
5	主題情境資料需求	熱門棒球話題與相關資料之提供。	【課堂作業】小組合作學習
6	基礎 SQL 語法介紹	新增、刪除、修改、查詢等語法	【講述法】【動手做】
7	進階 SQL 語法介紹、	Join、group、函數、欄位運算等語法	
8	SQL 語法強化練習		
9	棒球數據檢索	分析球員薪資、球員表現、球隊排名之 SQL 檢索	【講述法】【動手做】
10	資料模型	關聯式資料庫、E-R model	【課堂作業】小組合作學習
11	資料正規化	正規化之目的、方法	【講述法】
12	資料庫之需求收集與分析	棒球情境之需求描述與解析	【講述法】
13	資料庫設計	依棒球之情境需求設計資料表	【課堂作業】小組合作學習
14	期中考	以非選題評測對於資料庫相關原理的了解、選擇題測驗 SQL 語法	【學習評量】
15	資料庫程式設計	資料庫連線程式工作範例講解	【講述法】【動手做】
16	棒球情蒐資料庫實務應用	介紹國際比賽之棒球情蒐與分析。	【業師協同教學】
17	賽會資料庫實務應用	介紹賽會資料庫之應用。	【業師協同教學】
18	期末專題研究報告	專題研究成果上台報告。	【期末專題】

本研究所建構之教材，融合了可供下載觀摩、操作之 Lahman DB，以及提供 Web 查詢之棒球資料庫(BaseballReference.com)，設計了教學主題。相關的教材與教學活動等資料，建置於 Moodle 教學平台上，包括了棒球資料庫實例、主題情境融入實作的作業、網路學習資源的整理、資料庫原理與之式講述的投影片、學習單/回饋單/問卷、技術實作課堂示範和課後練習之影片說明。個人學習單的內容為運動數據網站和 Lahman DB 的探索，而課堂的小組討論，則模擬資料庫的

需求提出棒球資料查詢的情境。課堂的活動則讓學生以小組方式，進行 MS SQL Server 檢索以及 E-R Model Diagram 設計的實作練習。在學習評量上，學生之學期成績由課堂活動參與及作業(30%)、考試(40%)、學期專題(30%)所組成，考試為採個人筆試成績，而學期專題與作業則以分組團隊合作方式進行。為了讓學生了解資料庫在運動領域的運用，課程中安排了校外專家的演講，以協同教學方式讓學生認識了體育署全國各級學校運動人才資料庫。

3.3.2 研究架構與程序

整體研究設計共包含四個部分：課程設計、課堂活動、主動學習與回饋反思，如圖 8 所示。第一部分為教師參照相關文獻之教學策略，以「棒球」做為情境需求引導的發想，而「資料庫」做為所要傳達知識領域的核心，兩者跨領域融合在一起規劃教學內容。第二部分課堂活動，為依課程主題所實施之資料庫實務操作，將棒球情境中的需求運用資料庫的技術來達成，最後從理論知識來解說。第三部分主動學習則是為了解決課堂授課時數的限制，希望藉由學生自我主動學習，於課前研讀相關教材，並養成未來可利用網路開放教學資源自我主動學習之能力。第四部份反思與回饋，透過教學反思日誌、學習回饋單、問卷與訪談的方式、歸納分析問題與建議，並對課程進行修正。

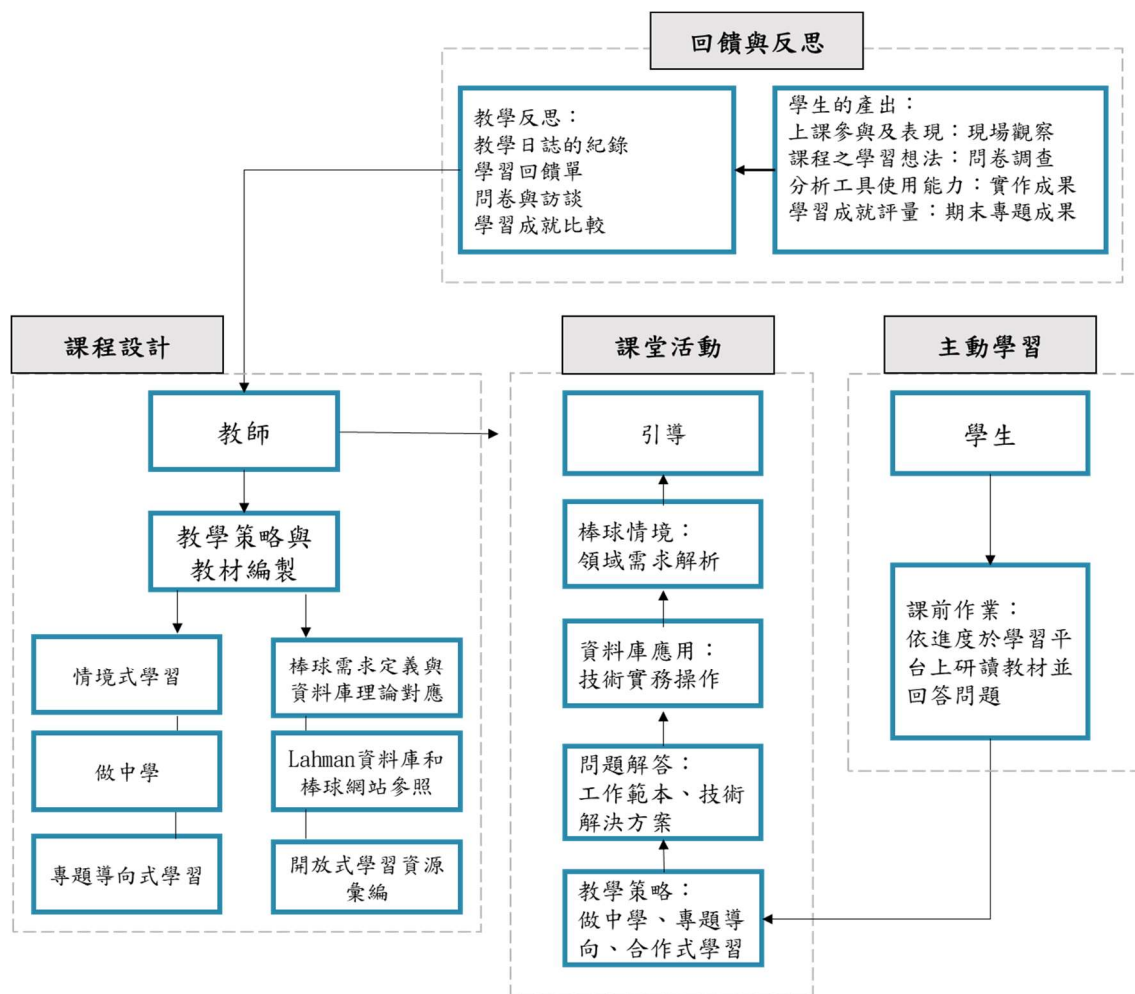


圖 8 研究設計

研究進行的程序，說明如下：

一、教學資料收集與教材編製

送審人依據教學主題與單元目標，於社群平台、新聞傳媒中搜尋，廣泛蒐集可能與資料庫有關之棒球熱門話題當案例，如 MLB.com、ESPN.com 等網站上之專家評論、Youtuber 熱門的短片、Facebook 粉絲團等，經挑選並改編為適合研究對象能自我主動學習的程度，且能於課堂中實作，編製成教材。

二、資料庫實務操作環境與學習平台建構

研究中將採用學校現有之 Moodle 平台，做為教材建置、問卷等教學輔助使用。針對課程實務實作部分，教室中資料庫將採用免費授權的 MS SQL (Express 或 Developer 版本)。而課後為了強化學生之練習，將於 Android 及 iOS 平台之智慧型手機上採用 SQLite 來進行行動學習。

三、融入「棒球」之「資料庫管理」課程設計

依規劃之教學單元主題，採用逆向設計方法重新安排順序，結合棒球熱門話題設計引導情境，以融入資料庫的教學中。並借助底層為棒球資料庫之棒球數據網站為互動查詢介面，做為實務操作成果之對照，讓學生達到「做中學」，最後引導學生以團隊合作學習方式完成期末專題。

四、實施教學

本研究預計實施教學的時間，為一學期的課程共 18 週。約每 4 到 6 週為一個教學主題，每個主題均包含棒球領域需求做為情境問題引導、資料庫技術應用與原理解說、課堂實作練習。課程行動研究以第 9 週為分隔，分為兩個階段。每階段完成後則進行教學反省與修正，再進入下一階段，如此循環。而在第 18 週時，則為期末專題成果報告，做為學習成就評量的一部份。於兩階段教學結束後，再依據相關資料進行研究。

五、教學回饋

教學過程中，蒐集學生的自我主動學習情形、課程實作之完成度、回饋單、學生訪談記錄、問卷填寫結果、以及研究者之教學省思日誌等，做為教學回饋。

六、修正教學設計

依據上一階段之教學回饋呈現之缺失或不足處，調整各部分講解重點、實作時間、教材的難易度等，修正後於下一階段的教學活動實施。

七、資料整理與分析

於每階段完成後之教學回饋資料進行彙整，待學期結束後，針對所蒐集的多方資料進行統計分析、三角檢證與統整，最後進行詮釋與分析。

八、提出結論與建議

綜整一整個學期課程兩個循環的資料，進行整理並歸納出結論，對結果提出相關建議。

3.4 融入運動分析實務問題之資料科學課程

3.4.1 教學規劃與課程設計

本研究設計的課程，旨在介紹與運動相關的各種資料及其用途，以及如何利用資料科學技術，來整理、分析這些運動資訊。課程中將以各種比賽的記錄與戰績統計、各職業運動聯盟定期公布之運動資訊為元素，融入程式語言、資料庫管理、資料視覺化介面等資訊技術的元素，使學生在實務上能具備運用資料科學的方法與技術，來解決實務上的問題。為一個學期共 18 週之課程，提供主修運動

資訊與傳播學系之大三學生選修。以驗證課程之設計，是否能符合學生的學習需求，能夠提升學生的學習動機，使學生達到一定程度的學習成效，並滿足學生的學習成就。學生在修課前，於大一階段應該已修過基礎的 Python 程式設計兩個學期共四個學分，以及基本統計學一個學期共兩個學分，大多數的學生應對於程式編碼及數理統計已具備基礎入門的程度，但僅非常少數能達到專精之程度。課程實施的同時將進行行動研究，以了解學生之學習情形及回饋，以作為課程設計改進之參考。研究擬投入之課程範疇、教材選用、教學資源應用、評量方式等，說明如表 9：

表 9 資料科學教學範疇

課程 範疇	課程名稱	運動資訊管理 (資料科學)
	開課學期	大三下
	選修別	系選修
教材 選用	工具使用	WEKA、KNIME、Python 相關套件
	教科書	自編講義、線上教學網站、網路開放資源
	實驗、實作、作業	於電腦教室中讓學生動手操作，完成指定任務。
教學資源應用	<ol style="list-style-type: none"> 1. 學習管理平台：學校之 Moodle 平台、Github 網站、Kaggle 網站等。 2. 資料科學線上自學課程：如 Youtube 相關影片及 Coursera、edX.org、Udemy 等線上學習課程網站。 	
教學目標	<ol style="list-style-type: none"> 1. 透過實作對資料科學與大數據分析有基礎之認識。 2. 具備大數據分析軟體與套件之實務操作能力。 3. 能利用資料科學的方法與技術解決運動訓練、競賽策略制定、運動表現與績效評估等問題。 	
教學方法	<ol style="list-style-type: none"> 1. 講述法：資料科學方法學與技術強化解說。 2. 實務實作(Learning by Doing)：資料分析工具與程式範例解說與操作，並於課堂完成指定操作任務。 3. 漸進釋放責任模式(GRRM)：逐步引導臨摹 GitHub 上之大數據分析範例程式，在逐漸發展至自己要分析的問題上。 4. 合作學習：分組進行運動分析實務問題研討 5. 專題導向式學習(Project-based Learning)：引導學生利用資料科學方法與技術解決運動領域之問題，完成期末專題研究。 	
評量方式採用	<ol style="list-style-type: none"> 1. 各主題上機實作成果: 60% 2. 期末專題研究(書面及上台報告): 30% 3. 主動學習之作業及課堂討論: 10% 	

具體的課程課程設計，整理如表 10 所示。教材之編撰是以開放式教育資源(OER)做為核心。在課程開課前的準備階段，研究者首先設定好授課的主題，再依據主題進行網路教育資源的瀏覽，這些教育資源在經過一系列的評估程序後，將符合條件的教材列入候選，之後配合課程的教學進度，重新組織發展成為正式授課的教材。於課程開始前，建構於學校之 Moodle 學習平台中，包含了許多網頁資料的連結，以及 YouTube 的影音短片，可供學生隨時瀏覽。

本研究採用的開放教育資源教材評估準則，分成五個面向，說明如下：

- 完整性: 需有逐步的分析說明，並提供範例之程式碼。程式碼可搭配網頁示範或提供操作介面者為佳。
- 可讀性: 能於 30 分鐘內供教師快速瀏覽完畢，了解大意者為佳。適合學生的英語程度或以中文撰寫為佳。
- 易懂性: 以入門程度之體驗為主，學生以學習過入門程度之 Python，具備基本運動比賽(籃球、棒球)知識，未曾接觸過資料科學者為對象。
- 權威性: 創作者具備相關學識與精力，網站主機具長久性或官方網站者為佳，內容為原創而非抄襲。
- 普遍性: 內容為一般常見之主題，具有一定的討論熱度，有一定的瀏覽數、討論度、參與度。

由於網路上的資料在近幾年隨著線上課程的推廣，以及部落客、Youtuber 等內容創作者的興起，充斥著各式各樣良莠不齊的內容，要找到一個有組織化、系統化，內容適合大學生學習，需經過教材開發者一一審視內容，方能列入教材。最終所選用應用於課程中的教材，共分成三個主題，相關資源與說明如附件一所示。在提升學生的學習興趣上，我們也列出了運動分析之問題場景和數據科學教學內容的對照。

課程設計上，第一個主題「運動數據庫的應用」的教學目標，是複習資料分析會使用到的 Python 語法，包含資料型態、以及統計、資料框架、繪圖等套件，以最入門的運動統計為場景。接著，介紹如何取得分析所需的資料，使用了 Python 的網路爬蟲套件或是可直接從運動網站上下載資料的 API 套件。

第二個主題為「NBA 籃球數據分析」，教學目標是要讓初學者了解「資料科學家」這個職業在做甚麼，以及在 NBA 的發展現況。之後則開始導入入門的應用案例，從資料的整理、到視覺化的呈現，進行初步的多變量相關性分析。而在這個主題中，也開始引入 GitHub 和 Kaggle 這類的開放資源，以引導學生於日後能具備自主學習之能力。

第三個主題「MLB 棒球數據分析」的教學目標，則聚焦於機器學習模型之應用，介紹模型建構流程之資料清理與過濾、特徵工程、模型之選擇與誤差測量等程序，以及一些演算法用來解決分類、預測等問題。考量學生的程度，教學中有涉及到機器學習演算的部分，並未詳加介紹，而是將模型當作是黑

盒子帶過。

而在作業設計部分，每個主題的結束，均搭配一個以組為單位的作業，共有三個。這三個作業的設計，是隨著對於資料科學的認知，而有不同的目的。第一個作業，主要是讓學生能夠複習 Python 的程式操作，並學會安裝運動分析所需的套件。以課程中所介紹的 sportsipy 套件和範例程式，在眾多數據中找出勝率最高的隊伍、薪資最高的球員、計算某隊某年度之得分等，這類資料過濾與篩選的操作。第二個作業，是讓學生於 GitHub 與 Kaggle 中檢索，報告與 NBA 有關之應用，並要求同學實際依照所提供的程式碼，實際於個人電腦的 Python 環境中，實作過一遍。第三個作業，同樣是讓學生從 GitHub 和 Kaggle 中檢索與 MLB 有關的應用，但檢索的關鍵字需同時加入機器學習有關的關鍵字，如人工智慧、類神經網路、決策樹、叢集分析等。而報告的重點不再要求程式實作，而是讀懂要解決的運動分析問題、解決的方法、使用的資料與欄位、結果呈現、學習心得。

表 10 資料科學課程設計

單元名稱	運動分析之問題與目的	資料科學所用到的技能	OER 網址	OER 說明
運動數據庫之應用	1. 認識運動科學中會用到的統計計算 2. 認識運動數據分析的資料來源，以及可用的數據類型 3. 查詢並擷取 reference 網站的運動數據	1. 資料型態 List、Dictionary 2. 資料框架 Pandas 3. 網站爬取 Beautiful soap 套件 4. 套件的使用: sportsPI	Python for Sport Scientists: Descriptive Statistics https://towardsdatascience.com/python-for-sport-scientists-descriptive-statistics-96ed7e66ab3c	這是 Towards Data Science Inc，發布於 Medium 線上部落格發布平台的一系列文章。可提供讀者免費存取。在運動科學家的 python 文章中，教導了常被使用到敘述統計之平均值、中位數、標準差、變異數，以及相關係數之 python 程式案例，使用了虛構的跑步距離與速度當資料集。
			sportsipy https://pypi.org/project/sportsipy/ https://sportsreference.readthedocs.io/en/stable/	sportsipy 是一個免費的 python API，可直接從 www.sports-reference.com 擷取所需要的資料，以便利 Python 應用程式之開發，特別是資料分析、機器學習有關之應用。
籃球數據分析	1. 數據分析時代的 NBA 2. APBRMetrics 簡介 3. 分析 Mapping Kobe Bryant's shots. 4. 籃球數據分析的程序 5. 球員 RPM 分析、進攻與防守、薪水的分析	1. 資料科學家的職業 2. 資料地圖的視覺化分析 3. NBA Stats API 4. 資料分析流程 5. Git 與 Kaggle 資源與介面的介紹 6. seaborn 中的 heatmap 函數於相關性分析 7. seaborn 的多變量分析及圖表	The NBA Data Scientist https://youtu.be/MpLHMKTolVw	介紹一位女性數學博士，於 NBA 費城 76 人隊擔任資料科學家。說明工作的性質與日常工作
			SportsVU NBA https://youtu.be/jOQE1_tkEwE	介紹 SportsVU 以攝影機為基礎之球員追蹤技術
			The math behind basketball's wildest moves https://www.ted.com/talks/rajiv_mahe_swaran_the_math_behind_basketball_s_wildest_moves	介紹可運用機器來計算籃球比賽中，距離、位置、命中率、投籃熱區等數值
			How data transformed the NBA The Economist https://youtu.be/oUvvhkXyOA	NBA 在數據分析發展下的轉變
			Every shot Kobe Bryant ever took. https://wikieducator.org/Sport_Informatics_and_Analytics/Pattern_Recognition/Python , https://graphics.latimes.com/kobe-every-shot-ever/ , https://github.com/datadesk/kobe-every-shot-ever	Kobe Bryant 曾經於球場上所投過的球，以於球場地圖上以視覺化的方式呈現。提供了可供互動查詢的網站，以及背後運作的程式碼。而這個資源也被 Wikieducator 的課程引用做為案例
			NBA Stats API. http://danielwelch.github.io/documenting-the-nba-stats-api.html , https://danielwelch.github.io/little-pynny/	供 python 讀取 stats.nba.com 網站數據之免費 API，開發應用程式使用

			NBA Analytics With Python. https://princetonsportsanalytics.wordpress.com/2017/09/30/nba-analytics-with-python-a-tutorial/	Princeton Sports Analytics club 所提供之 NBA 分析簡單入門範例，涵蓋了資料的取得、處理、視覺化、學習模型之應用。
			NBA Player RPM Prediction Defense vs Offense. https://www.kaggle.com/dhamlett/nba-player-rpm-prediction-defense-vs-offense/notebook	分析球員的效能
			球員效率與薪資的分析。 https://zhuanlan.zhihu.com/p/42625197	中文分析球員效率與薪水
棒球分析	1. Sabmetrics 及常用統計數值介紹 2. 以歷史對戰紀錄進行蒙地卡羅計算勝率(非 OER) 3. 不同年代之每場比賽得分、勝場數 4. 球員入選棒球名人堂之預測 5. 打擊角度、速度與結果之分類	1. Excel 之數據整理、統計工具箱(非 OER) 2. 蒙地卡羅模擬範例(非 OER) 3. 視覺化之資料探索(Exploring and Visualizing The Data) 4. 機器學習模型之流程: 資料清洗與前處理、特徵工程、模型選擇與誤差度量、 5. sklearn 套件之使用-K mean、logistic regression、random forest 6. 機器學習分類模型之介紹:KNN、SVC、gBDT	GLOSSARY https://www.mlb.com/glossary	MLB 官網上所提供與棒球相關之專有詞彙說明，包含許多賽博計量學所使用之表現統計指標
			Scikit-Learn Tutorial: Baseball Analytics Pt 1. https://www.datacamp.com/community/tutorials/scikit-learn-tutorial-baseball-1	datacamp 主要是提供付費的資料科學課程，而有部分為免費的入門教學。此資源為一步一步以 Python 程式碼介紹如何利用機器學習模型來進行資料分析與預測
			體育分析與機器學習 https://www.zhihu.com/column/c_1005779975458443264	知乎上的專欄，約有 20 多篇利用機器學習在運動分析上應用的短篇文字，主要在說明運動分析的步驟，專有名詞的解釋，部分有提供程式碼。
			Classifying MLB Hit Outcomes. http://tylerjamesburch.com/blog/baseball/hit-classifier-1 https://github.com/tjburch/mlb-hit-classifier/blob/master/notebooks/1-model-selection.ipynb	波士頓紅襪隊分析師 Tyler James Burch 個人的部落格，說明打擊結果之分類，並提供 github 之程式碼

3.4.2 研究架構與程序

整體研究設計共包含四個部分：課程設計、課堂活動、主動學習與回饋反思，如圖 9 所示。第一部分為教師參照相關文獻之教學策略，以「運動分析」做為知識領域問題的發想，而「資料科學」做為問題的解答方法，兩者跨領域融合在一起規劃教學內容。第二部分課堂活動，為依課程主題所實施之資料分析實務操作，並將分析結果運用運動領域之知識做解析。第三部分主動學習則是為了解決課堂授課時數的限制，希望藉由學生自我主動學習，於課前研讀相關教材，並養成未來可利用網路開放教學資源自我主動學習之能力。第四部份反思與回饋，透過教學反思日誌、學習回饋單、問卷與訪談的方式、歸納分析問題與建議，並對課程進行修正。

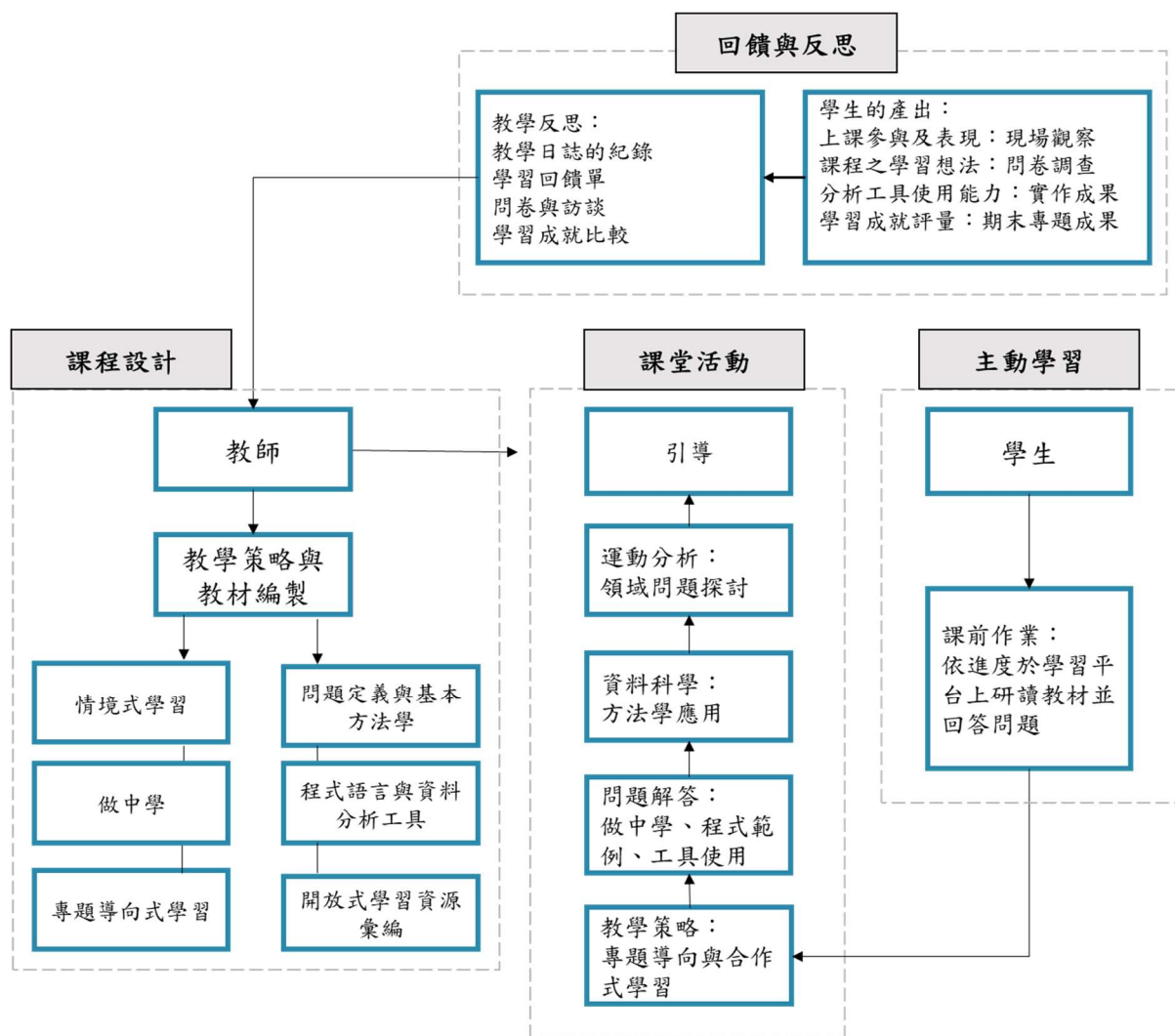


圖 9 研究設計

研究進行的程序，說明如下：

一、教學資料收集與教材編製

研究者依據教學主題與單元目標，於開放式教育資源中搜尋，廣泛蒐集運動分析領域的各類分析問題與案例，如 Kaggle 平台上與運動有關之資料分析競賽題目，或是 Github 上與運動有關之大數據分析程式，經挑選並改編為適合研究對象能自我主動學習的程度，且能於課堂中實作，編製成教材。

二、軟體工具與學習平台建構

研究中將採用學校現有之 Moodle 平台，做為教材建置、問卷等教學輔助使用。此外，針對課程實務實作部分，則採用免費的開源軟體，如 WEKA、KNIME、Python Scikit-learn 套件等資料分析工具。這些軟體功能十分強大且涵蓋眾多資料科學之方法學，受限於學生程度及授課時數，課程中無法逐一詳述。故需配合教材，一一測試後選擇出最合適的操作的部份，或僅使用單一功能，做為本研究使

用。當決定採用之軟體工具後，將相關教材、範例程式碼等，建置於平台上。

三、整合「運動分析」及「資料科學」之課程設計

依所設計之運動分析主題，如與棒球分析、籃球分析相關之 Sabermetric、APBRmetrics 等領域知識，融入敘述統計、資料視覺化、蒙地卡羅模擬、機器學習演算法等資料科學的教學中。並以運動闡場景發展情境式教學，運動分析之案例實作讓學生能「做中學」，並引導學生以團隊合作學習方式完成期末專題。

四、實施教學

本研究預計實施教學的時間，為一學期的課程共 18 週。約每 4 到 6 週為一個教學主題，每個主題均包含運動分析之領域知識、資料科學方法學應用、課堂實作練習、業師協同教學講座。課程行動研究以第 9 週為分隔，分為兩個階段。每階段完成後則進行教學反省與修正，再進入下一階段，如此循環。而在第 18 週時，則為期末專題成果報告，做為學習成就評量的一部份。於兩階段教學結束後，再依據相關資料進行研究。

五、教學回饋

教學過程中，蒐集學生的自我主動學習情形、課程實作之完成度、回饋單、學生訪談記錄、問卷填寫結果、以及研究者之教學省思日誌等，做為教學回饋。

六、修正教學設計

依據上一階段之教學回饋呈現之缺失或不足處，調整各部分講解重點、實作時間、教材的難易度等，修正後於下一階段的教學活動實施。

七、資料整理與分析

於每階段完成後之教學回饋資料進行彙整，待學期結束後，針對所蒐集的多方資料進行統計分析、三角檢證與統整，最後進行詮釋與分析。

八、提出結論與建議

綜整一整個學期課程兩個循環的資料，進行整理並歸納出結論，對結果提出相關建議。

3.5 研究方法及工具

本研究採用行動研究(action research)模式來進行，在引進創新教學教法和重新設計教材的同時，執行經過規劃的教學行動方案、持續改進實務行動、監控並評鑑行動的實施歷程與結果，過程中透過反覆的循環，來改進教學的品質，提升學生學習的成效。

行動研究的概念，最早是由 Lewin 於 1948 提出，而後由 Corey 於 1953 年將其應用在教育研究上，認為教師即為研究者，讓教師參與有關的研究過程，以改進教育的現況，而後在教育研究上逐漸且普遍受到重視。行動研究後續有許多學者，如 Elliot、Ebutt、Kemmis 和 McTaggart 等，針對細節稍作修正，提出不同的模式。本研究採 Elliot 的模式，使用螺旋式循環的過程，每個循環包含了五個步驟：界定初始想法、提出一般性計畫、執行行動步驟、監控執行與效果、偵查。每一個循環的初始想法會在行動過程中產生改變，而這些改變會促使行動者再次發生反省，致使計畫修正，再次進入第二循環。

行動研究法在方法論上是採用折衷統合的方法，為質性與量化之綜合研究。其優點是可以運用許多不同的方法，蒐集資料。量化的部分採用問卷調查，調查參與課程的學生對於教材內容各個知識點之學習困難，以及對教師課程進行方面之滿意度，以進行統計分析。而針對學習成效的部分，則以學習評量成績，進行統計分析。質性的部分，則採用深度訪談法、學習回饋單，蒐集學生對於教學與自我學習成效之反饋意見，而對於教師則以觀察法，將教學省思記錄到教學日誌中，最後彙整多方資料進行分析，以概念圖、原因和影響分析、解釋性論述，來闡釋資料。各研究工具，詳細說明如下。

一、質性資料研究工具：

1. 學生學習回饋單

配合課程單元設計，用來以了解學生的學習情況。將於學校的 Moodle 學習平台上設置回饋表單，於教學過後讓學生填寫，可藉此讓學生在回顧上課的過程與心得、想法，提供教師教學上的回饋，並讓研究者瞭解，所設計的教學內容與活動，於實際教學環境的適用性。

2. 教師教學省思日誌

教師於每課程單元結束後，記錄課堂上發生的事件與遭遇困難，尤其是實作時，學生程式語法除錯上的困難，或是資料庫課程遭遇到的正規化問題，以及提問的問題。並且觀察學生動手實作程式、網頁、資料庫、數據分析等，所花的時間，及上課態度等，以分析課程進行之優缺點與待改進之處，進一步地進行自我省思、改善教學。

3. 學生訪談紀錄

針對學習表現較佳和較差的學生，以立意抽樣方式進行深度訪談，以瞭解學生對於課程的想法、心得感想以及建議與回饋。

二、量化資料研究工具：

1. 自我學習成效態度量表與學習體驗量表

本研究將參考相關文獻，編製適用於程式設計課程之問卷，設置於學校 Moodle 學平台上讓學生於學期末填寫，問卷內容包含學習態度、學習動機、自我效能三個部分，以李克氏五點量表(five-point Likert scale)來計分。

2. 實作與測驗評量

本研究所使用的資料，為各項上課課堂實作練習與課後實作作業的之完成比例，以及實施上機撰寫程式答題之期中考和期末考，或是所習得之專業知識測驗成績，以及期末專題報告的成績。

4. 研究成果與學生學習成效

4.1 程式設計課程(108 年教育部教學實踐研究計畫)

4.1.1 學生學習回饋量化分析

一、自動化程式線上評測系統使用情形

兩個學期的研究教學中，學生使用 Code Judger 平台之情形，全體學生類計次數如圖 10 所示。我們可以看到，在 108、309、310、409 這些題目上，正確率比較低，但是相對的這些題目，學生的答題次數也會比較多。這隱含的意義可解釋成，學生在遇到難題時，會盡力的去解決它、克服它，而不是放棄，這對於學習動機而言，呈現出正向。

而進一步我們對照學生之學習成就，將期中考與期末考成績與平台使用情形比對(如圖 11)，大致上可看出解題數、正確率與成績具有正相關。但因學生之樣本數(<30)不足，未進行統計相關係數計算及統計檢定驗證顯著性。

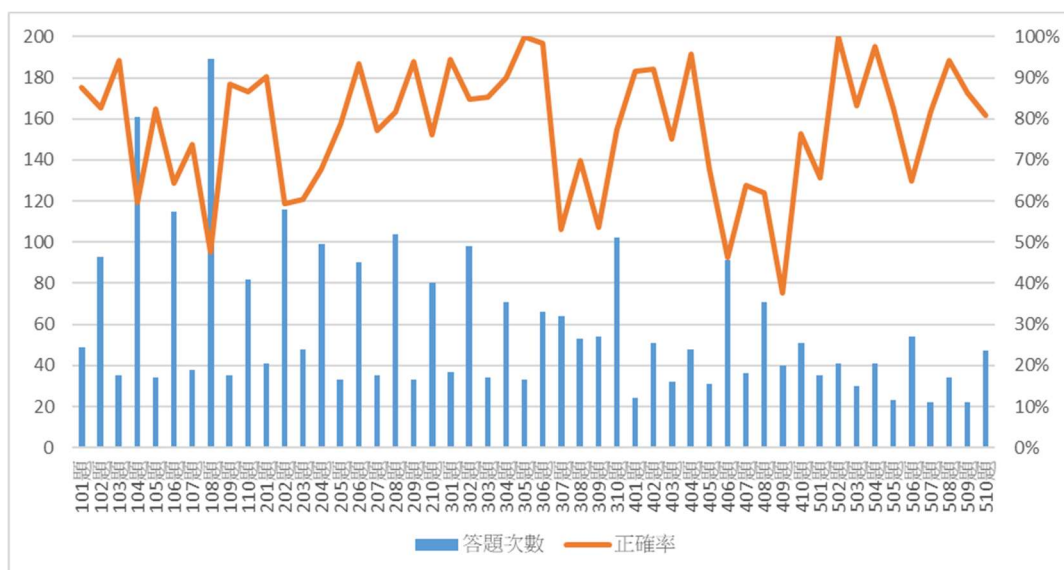


圖 10 程式自動評測平台使用統計

編號	系級	學號	姓名	108-1					108-2				
				期中考	解題數	正確率	平時成績	期末考	期中考	解題數	正確率	本學期解題數	本學期正確率
001	運動資計	40409001	林奕廷	100	83	92.77	22.16	100	60	183	92.35	100	92
002	運動資計	40509002	林奕廷	98	91	84.62	21.53	100					
003	運動資計	40509003	林奕廷	93	33	78.79	15.58	65					
004	運動資計	40509004	林奕廷	100	44	81.82	17.02	100					
005	運動資計	40509005	林奕廷	85	57	49.12	12.86	60					
006	運動資計	40509006	林奕廷						100	67	91.04	67	91.04
007	運動資計	40509007	林奕廷						80	5	60	5	60
008	運動資計	40609008	林奕廷						0	23	43.48	23	43.48
009	運動資計	40809009	林奕廷	60	74	64.86	16.87	100	40	241	58.09	167	55.09
010	運動資計	40809010	林奕廷	100	111	75.68	21.81	60	80	244	81.15	133	69.17
011	運動資計	40809011	林奕廷	48	82	78.05	19.69	100	80	249	64.26	167	57.49
012	運動資計	40809012	林奕廷	68	50	58	13.69	30	40	70	64.29	20	80
013	運動資計	40809013	林奕廷	98	56	71.43	16.38	100	40	129	63.57	73	57.53
014	運動資計	40809014	林奕廷	38	93	46.24	15.49	100	40	129	48.84	36	55.56
015	運動資計	40809015	林奕廷	30	69	79.71	18.84	80	20	158	71.52	89	65.17
016	運動資計	40809016	林奕廷	23	36	83.33	16.58	70	0	98	78.57	62	75.81
017	運動資計	40809017	林奕廷	45	58	34.48	10.58	85	60	248	45.16	190	48.42
018	運動資計	40809018	林奕廷	95	73	52.05	14.71	70	40	156	49.36	83	46.99
019	運動資計	40809019	林奕廷	100	56	75	16.95	100	60	184	72.28	128	71.09
020	運動資計	40809020	林奕廷	100	67	79.1	18.57	100	100	166	80.12	99	80.81
021	運動資計	40809021	林奕廷	88	51	70.59	15.81	100	100	81	72.84	30	76.67
022	運動資計	40809022	林奕廷	100	51	70.59	15.81	100	100	112	78.57	61	85.25
023	運動資計	40809023	林奕廷	71	68	60.29	15.61	50	60	128	59.38	60	58.33
024	運動資計	40809024	林奕廷	100	61	83.61	18.78	100					
025	運動資計	40809025	林奕廷	0	0	0	0.00	0					
026	運動資計	40809026	林奕廷	100	87	39.08	13.82	80	40	98	38.78	11	36.36
027	運動資計	40809027	林奕廷	0	116	56.03	19.06	30					

圖 11 平台使用與期中考、期末考成績之比較

二、學生學習成就評量

第一個學期之期中考，主要是以教科書為主的內容進行上機考試，而期末考則是以 TQC 之題庫 101~201 之奇數題為範圍，進行上機考。考試的結果，期中考有 8 位滿分，期末考共有 12 位滿分，整體而言因此階段教學範圍小且簡單，學生學習成就評量成果普遍為佳。

而第二個學期之期中考，是以 TQC 之檢定考試範圍(101~510)，採用認證考試的考場環境進行評測。考試結果，共 20 人應考，通過(>70 分)7 人(35%)，而未通過 13 人(65%)。對比於上一學期之學習成就相對為差，其原因由學生深度訪談之回饋得知，多數學生反映，與上學期相比考試範圍太大以致成績較差。也有些學生提到，面對認證考試上機實測環境太緊張，以致失常。

三、學生問卷回饋

研究中，我們分別在第一個學期末和第二個學期末，以問卷調查的方式來蒐集學生的學習意見回饋，問卷的內容分別有：學習動機、教材教法、學習程效；以及針對本研究所採用之「程式評測平台」和「流程圖」、兩人一組之合作學習策略、學習成效。而部分的問題是以反向題方式呈現。因問卷的樣本數不足 30 分，屬小樣本因此不進行信度、效度檢定，僅統計平均分數與標準差，結果如表 11、表 12，並做為研究中深度訪談時的參考。

從表 11 中可看到，題號 A3 的平均分數極高，顯示出學生大都認同學習程式設計的重要，但 A1、A2 也說明學生興趣與主動性屬普通。而題號 B1~B4 的分數，反映出學生對於採用線上評測系統有極正面的回饋。而 C1、C6 可看出，學生對於課程的評量表現不慎滿意，而學習興趣未更提升。

表 11 第一學期問卷之內容與結果統計

	問題	平均	標準差
A.學習動機			
1	我對於學習「程式設計」很有興趣。	3.75	0.97
2	我是自己主動要選修「程式設計」的課程。	3.85	1.18
3	我覺得學習「程式設計」能建立專業，對自己的將來很有幫助。	4.45	0.67
4	未來我有興趣想從事資訊相關工作	3.30	1.08
5	我選修「程式設計」課程，只是為了畢業學分，興趣其實不高。	2.50	1.19
B.教材教法			
1	「教科書」(期中考前)與「線上評測系統」(期中考後)，我比較喜歡用「線上評測系統」來上課。	4.30	0.66
2	我覺得「線上評測系統」對於學習成效很有幫助。	4.45	0.60
3	我覺得「線上評測系統」可以激勵我，一題接著一題練習，獲得答對時的成就感。	4.60	0.50
4	我很喜歡使用「線上評測系統」這類強迫練習的上課方式，可督促自己認真上課。	4.30	0.66
5	我覺得「線上評測系統」應該是課後回家練習就好，上課應多講一些「教科書」的內容。	2.90	0.91
6	老師上課內容教太多、太快，不易吸收。	2.35	0.67
7	上課講的內容，我有很多聽不懂。	2.85	1.04
8	課堂上練習的時間不夠。	2.65	0.99
9	上課所遇到的程式問題，都有問老師、同學或靠自己研究，最後都有解決。	4.20	0.83
C.學習成效			
1	我「期中考」得到的分數符合我應有的程度，覺得很滿意。	3.65	1.46
2	我「期末考」得到的分數符合我應有的程度，覺得很滿意。	4.15	0.81
3	修了一學期的課後，我覺得程式設計對我來說很困難。	2.90	1.41
4	我都很認真的在上課。	4.0	0.92
5	我下課後有再花時間做練習，準備考試。	4.1	0.91
6	上了一學期的程式設計課程後，我對於程式設計其他相關課程的興趣是「更提高」的。	3.65	0.88

在表 12 中，我們看到學生對於題號 A1~A5 和 B1~B5 的平均分數均超過 4 分，可顯示出採用 ADRI 的教學模式，更加深化講解程式的流程與語法，並採合作學習策略是非常被學生接受與認同的。但在 C1~C5 以及 D1~D6，則一般分數

都在 4 分以下，顯示學生對於「程式設計」仍普遍覺得自己學的不夠好。但在 D2、D4、D5 上則有超過 4 分，說明學生在第二學期的課程上，已能掌握學習的方法與重點，並有認為程式設計能力有相當多的進步。

表 12 第二學期間卷之內容與結果統計

問題	平均	標準差
A.使用「程式評測平台」結合「流程圖」的講解：		
1 能讓我節省學習的時間	4.29	0.67
2 更快了解學習內容	4.53	0.62
3 改善我的學習效能	4.29	0.77
4 改善學習的品質	4.41	0.51
5 使的學習程式語言變得容易	4.35	0.79
B.兩個人一組的學習方式：		
1 能讓我節省學習的時間	4.12	0.86
2 更快了解學習內容	4.06	0.90
3 改善我的學習效能	4.06	0.90
4 改善學習的品質	4.06	0.83
5 使的學習程式語言變得容易	4.24	0.83
C.本學期學習程式設計		
1 很少有不會寫的問題	2.47	1.00
2 很少出現程式的 bug	2.00	0.87
3 很方便隨時可以自學寫程式	3.47	0.94
4 更清楚知道要如何學習寫程式	3.82	0.95
5 教學的活動(如:課後練習、期末專題)，都很容易上手	3.35	1.17
D.學習成效		
1 我「期中考」得到的分數符合我應有的程度，覺得很滿意	3.35	1.46
2 我「期末考」得到的分數符合我應有的程度，覺得很滿意	4.00	0.79
3 修了兩個學期的課後，我覺得程式設計對我來說還是很困難	3.47	0.94
4 與上學期相比，我更認真、更投入在課程上	4.00	0.87
5 與上學期相比，我自認為學習成果有「更好」	4.12	0.70
6 與上學期相比，我對於程式設計的課程的興趣是「更高」的	3.94	0.66

4.1.2 學生訪談意見質性分析

在實施 ADRI 教學的同時，教師於課堂上透過與學生的互動中，也了解到學生的反應及回饋，記錄於教學省思日誌中，同時也在實訪問卷調查同學的回饋意

見後，對於全體的學生進行深度訪談，回答內容經編碼分類後依出現的頻率彙整如表 13，以了解學生學習的情況，。

表 13 學生訪談意見回饋

訪談問題	常見回答重點節錄
1.上學期與下學期上機考試成績差異的原因為何?	<ol style="list-style-type: none"> 1. 考試太緊張，加上考試準備範圍比較大一點。 2. 這次因為有牽扯到證照的部分，得失心會更重一點 3. 這次太緊張。 4. 上學期就是本身是沒有在打工，現在是有在打工。
2.使用評測平台是否有助於課後練習?	<ol style="list-style-type: none"> 1. 練習的次數跟上課的態度應該沒什麼變。 2. 可增加熟悉度。 3. 就常常練，上學期期末我練習很多次。 4. 就惡補這樣。
3.對證照考試結果滿不滿意?	<ol style="list-style-type: none"> 1. 想說應該，感覺憑印象應該可以考得還不錯，結果就破功。 2. 不滿意，至少要有 60。 3. 我覺得還可以，因為另外兩題自己沒有把握好，然後練習，就難度比較高的自己沒有把握好，自己沒有多練一點。
4.採用 ADRI 教學方式是，是否有助解決學習的困難?	<ol style="list-style-type: none"> 1. 上課也聽不懂，有一些啦。 2. 有幾題比較...，像那個函式跟迴圈有時候邏輯還是，就會忘記說接下來要怎麼辦。 3. 看不太懂流程圖，因為它都是一直分分支出去，所以我覺得寫得有點太複雜。 4. 比較好理解。 5. 後面因為有加那個流程圖就是比較容易能夠了解，大概那個程式要怎樣子寫這樣子。 6. 我覺得多少是有幫助，就是他比較讓我順下來的順序會比較順。

4.1.3 教師教學省思與教學建議

研究者於過程中透過教學省思日誌的紀錄、學生問卷回饋、學生個別訪談、專家意見諮詢後，反思教學改進之道，歸納發現如下：

一、採用程式自動評測系統題庫式的練習，能提升學習興趣與成就感

與傳統採用教科書之練習題相較，學生於程式自動評測平台上練習，更容易獲得成就感。主要是因為平台上的題目普遍較簡短也較容易，雖然題目比較多，但按部就班一題接著一題練習，會比教科書上的題目，更有練習解題的意願。

二、改變考試的方式，易影響到學生學習評量的表現

採用真實的認證考試的上機考試環境，容易造成學生臨場緊張，容易失誤。

而若採用與平時練習相同的環境進行上機考，則較無失誤發生。這點亦可從問卷中對於期中考的表現度，從第一學期 3.65 降到第二學期 3.35 的情況看出。

三、學生願意於課後花多少時間學習，影響學習成效甚鉅

雖然說學習動機與學習興趣，對於學習成效影響很大。但依研究中之質性訪談發現，學生的學習成效，與願意學習投入的時間影響更大。有學生提到對於程式設計有高的學習動機與興趣，且上課時也都非常認真學習和做練習，但在課後則因打工，反而於考前疏於練習，已至學習評量分數不高。

四、引入流程圖對於 ADRI 的教學模式有幫助

許多同學對於能引入流程圖，加強程式邏輯的講解十分認同。然而在質性訪談中，對於學習成就較差的學生，其回應為有些困難的題目，流程圖也是看不懂。

五、兩人一組的學習方式，接受度很高

本研究參考成對編程概念，於第四階段讓學生以兩人一組來進行學習，由問卷中得到學生極佳的回饋。

本研究在實施兩學期課程後，提出之建議如下：

1. 成對編程，不容易於課堂上實施。(如同組學生均不擅長程式時。)
2. 學生人數少之課程，計畫成效難以用量化驗證，而以質性方式呈現。
3. 各科系(本校其他運動相關科系)學生，須具備的程式設計能力不同。
4. 不同學生、不同時間的學生狀態都不同，處在一個變動環境中，成效不容易驗證。

而對於未來教學上，提出之省思如下：

5. 藉由計畫之執行，可讓教師放下腳步，思考教材教法，仔細了解學生的學習情形。
6. 計畫拉近了老師與學生的距離，教學評量分數有提升。
7. 學生學習成就易受外部因素影響。
8. 學習成就(認證考試)佳的學生，是否就具備利用撰寫程式解決問題(如：專題製作之呈現)。
9. 本研究之教學模式，是否適合擴展應用到全校所有科系(如全校性的通識課程)。

4.2 網頁設計課程 (111 年教育部教學實踐研究計畫)

4.2.1 學生學習回饋量化分析

本研究設計了課程學習之自我效能量表、學習興趣量表、與奧林匹克教育專業服務學習結合的感知量表，共三份問卷，均採李克氏五點量表。自我效能量表分別於第 4 週及第 18 週進行前後測，以了解學生之學習成效。而學習興趣量表、

與奧林匹克教育專業服務學習結合的感知量表，僅於期末的第 18 週施測，以了解學生的學習體驗感知。在完成修課的 48 名學生中，共有 40 名學生(83%)完整提供了三份問卷的回應，其中 19 名為男性，21 名女性，除了一位是大三重修生以外，其餘均為運動資訊傳播學系大二學生。

一、學生課程學習自我效能量表

自我效能量表(self-efficacy scale, SE)用來讓學生自我評估對於較以往更深的理解、信心的提升、對自我專業的發展有幫助，三方面之感受，問卷的內容修改自 Kearney-Volpe et al. (2021)的研究，共有 11 個題目，以李克氏五點量表來計分。問卷內容及前後測結果的分析，如表 14。

40 份有效問卷之前後測差異，採成對樣本 t 檢定的結果顯示，在網頁設計有關的各項專業學習上，課後均較課前有顯著提升。其中，信心程度增加最多的是有關開發框架的部分(SE6)，其次是關於 CSS(SE5)。而改變幅度最小的是團隊溝通、協調(SE11)，其次是學習網站前端開發技術的能力(SE10)。

表 14 自我效能量表與施測結果 (N=40)

問題	前測		後測		前後測 差異	顯著性 (雙尾)
	平均值	標準差	平均值	標準差		
SE1. 我可以逐步解釋我對一個網站專案的想法	2.45	1.061	3.75	0.840	-1.300	0.000***
SE2. 我知道創建網站所涉及的各個階段	2.15	1.051	3.75	0.742	-1.600	0.000***
SE3. 我能把我的網站上傳到一個伺服器上	1.90	0.955	3.48	0.784	-1.575	0.000***
SE4. 我知道如何構建 HTML 元素	2.33	0.859	3.78	0.800	-1.450	0.000***
SE5. 我知道如何使用 CSS 來影響網頁的風格	1.95	0.986	3.58	0.844	-1.625	0.000***
SE6. 我能套用並編寫一個簡單的開發框架程式庫 (例如: JavaScript、Bootstrap、W3.CSS 等任一個)	1.73	0.877	3.48	0.933	-1.750	0.000***
SE7. 我能使用套裝軟體工具來製作網頁 (如: Dreamweaver、VS Code、BlueGriffon 等任一種)	1.95	1.011	3.55	0.986	-1.600	0.000***
SE8. 我有信心可以有效地處理網頁開發時所遭遇的突發問題	2.00	1.013	3.18	0.844	-1.175	0.000***
SE9. 如果我投入必要的精力，我可以解決在創建網站中遇到的大多數問題	2.48	1.198	3.43	0.903	-0.950	0.000***
SE10. 我有信心具備學習網站前端開發技術的能力。	2.58	1.083	3.28	0.847	-0.700	0.000***
SE11. 我有信心具備團隊成員相互溝通、協調，協力完成資訊專案的能力。	3.45	0.876	3.90	0.928	-0.450	0.030**

二、學生學習興趣量表

學習興趣量表，係改編自 Hsu et al. (2021)課程期望量表，共有六個問題。

問卷內容及結果統計如表 15。結果顯示，學生對於教材設計(LI4, M=4.03)及課堂中實作活動(LI3, M=4.00)的體驗式學習，是非常認同的。但由於這門課是必修課程，不是每一位學生都是有興趣修課的，因此課程的學習興趣(LI1, M=3.48)和對於期末專題的興趣(LI6, M=3.53)的平均值是最低的。

表 15 學生學習興趣量表與施測結果 (N=40)

問題	平均	標準差
LI1. 我對於網頁設計課程非常有興趣學習	3.48	0.933
LI2. 我認為學習網頁設計對於我未來職能非常有幫助	3.63	0.838
LI3. 我認為老師上課所設計的實做體驗非常符合課程的目標	4.00	0.784
LI4. 我認為老師上課所選用的教材非常符合課程的目標	4.03	0.733
LI5. 對於結合課程實做來學習網頁設計課程，我是非常有興趣的	3.78	0.698
LI6. 對於期末結合奧林匹克主題的專題，我是非常有興趣的	3.53	0.716
LI7. 我對於這門課整體而言感到很滿意	3.83	0.781

三、期末專題與奧林匹克教育專業服務學習結合的感知

為了了解以跨課程所學習到的知識，以奧林匹克教育作為期末專題網站開發的主題，課程的最後，進行了專業服務學習感知的問卷調查，問卷編製係修改自 Lee et al. (2018)之研究，以蒐集以下之看法：(a)學習感知(perceived learning, PL)；(b)服務涉入(service involvement, SI)；(c)對專業的態度(attitude toward profession, AP)；(d)服務學習感知(Perception of service learning, PS)。問卷的內容，及結果統計如表 16。

學生對於與專業服務學習之整體感知，所有問題回饋的平均值為 3.86。在 5 分量表中，平均 3.86 分表明，學生平均認為他們在服務學習體驗方面接近積極。而對於四個不同類別的感知上，表二顯示問卷中各類別的內部一致性信度(Cronbach's Alpha)均達 0.8，而各類別平均值界於 3.742 至 3.955。這表明學生對於每一個類別的感知都是接近積極的，但均未達到非常積極的程度，他們同意期末專題與專業服務學習結合。這有助於課程的學習(PL, M=3.955)，可連結到另一門課程的專業服務學習(SI, M=3.803)，並且改善溝通與問題解決技巧(AP, M=3.742)，有機會尋找更多能提供專業服務學習的機會(PS, M=3.932)。

從各題目來看，平均值對高的為”奧林匹克主題的服務學習(期末專題)，強化了我在這門課中所涵蓋主題的知識”(PL3, M=4.00)，最低的是”我所完成的期末專題，幫助我明確化未來的職業規劃”(AP1, M=3.61)，顯示出學生感知最正向積極的對於學習的幫助，但是否有助於未來職業規劃，較持保留態度。

表 16 專業服務學習的感知量表與施測結果

類別	問題	平均值	標準差	類別平均	Cronbach's Alpha
學習感知	PL1. 我所做的奧林匹克教室的專題幫助我更了解網頁設計課程中所教授的內容	3.91	0.741	3.955	0.868
	PL2. 專題中與組員的合作與互動，幫助我在這門課的學習	3.95	0.939		
	PL3. 奧林匹克主題的服務學習(期末專題)，強化了我在這門課中所涵蓋主題的知識	4.00	0.747		
服務涉入	SI1. 使用網頁設計課程所學到的技能，創作出奧林匹克教室的網站，對於能善用自己的能力幫助他人，我感受到了個人的責任感。	3.89	0.813	3.803	0.801
	SI2. 由於這門課的專題經驗，我會期待有更多的機會能參與奧林匹克教室(或奧林匹克教育)的活動。	3.73	0.788		
	SI3. 這門課以製作奧林匹克教室網站的專業服務結果，我感覺到與奧林匹克教育有更加緊密的連結。	3.80	0.795		
對專業的態度	AP1. 我所完成的期末專題，幫助我明確化未來的職業規劃	3.61	0.784	3.742	0.887
	AP2. 我所完成的期末專題，幫助我改善“真實世界”的溝通技巧	3.75	0.892		
	AP3. 我所完成的期末專題，幫助我改善問題解決的技巧	3.86	0.824		
服務學習感知	PS1. 透過這次專題的過程，幫助我看見了我在網頁設計課程中所學，能於日常生活中被應用。	3.95	0.645	3.932	0.923
	PS2. 這類結合跨課程、專業服務學習、真實需求的課程，應該在更多的大學課程中被實踐。	3.91	0.709		
	PS3. 我會推薦這類專業服務學習的課程給其他學生	3.93	0.759		

4.2.2 學生訪談意見質性分析

此外，也於第 16、17 週時，於課後一對一訪談，蒐集學生對於體驗式學習中各項實作的意見，共有 23 位學生受訪。

本研究於第 16、17 週時，分批於課後對學生進行一對一訪談，以蒐集學生對於體驗式學習的課堂實作的意見回饋，共有 23 位學生受訪。此外，也於第 18 週施測的問卷中，加入了開放性的問題，蒐集對於課程專題方面的回饋，共收到 40 份回應。這些回饋的意見經編碼後採內容分析法進行分析，問題與最常出現的回答類型，摘錄於表 17。

表 17 學生訪談意見回饋

訪談問題	回饋意見節錄
體驗式學習之課堂實作	
1. 從這些實作中，你學到了什麼？	1. 能使用記事本就可以製作出網頁。 2. 是學習模式，會用 W3School，自主上網學習。 3. 細心程度有成長。 4. 自己做出網站提升成就感。
2. 樂意用這種做實作的方式學習嗎	1. 在課堂上跟著老師的步驟實作，我很喜歡。 2. 有實作可以加深印象。 3. 課堂上要有時間給我們練習、做作業。 4. 蠻好的，優點是知道如何製作出網站，缺點是耗時、做不出來。
3. 這些實作對於後面的考試幫助為何？考試考得如何？	1. 比較沒有，在實作上不足以記住考試的內容。 2. 有，潛移默化的感覺。 3. 實作可加深印象，但考試不一定寫得出來。 4. 沒有幫助，考試是比較偏向死記。
4. 這些實作對於後面的期末專題幫助為何？	1. 我可以透過實作的範例，去具體銜接專題。 2. 以循序漸進的方式製作專題。 3. 我比較知道我可以做的分工是甚麼。 4. 具體會知道怎麼做，有方向可以學習。
期末專題跨課程服務學習	
1. 對於專題以團隊工作的方式進行，你的感受為何？	1. 幾乎都由我一人處理。 2. 剛開始其實大家都偏向不想做事，但在我們好好溝通的情況下，每個人都有達到自己最大的貢獻 3. 分工的狀態還算可以，工作都會互相幫忙。 4. 以團隊分工的方式進行比一個人做壓力相對少。
2. 對於以奧林匹克教室為期末專題，你的感受為何？	1. 優點是能夠將兩門課程進行結合使同學能夠將課程更加熟練。 2. 能將自己所學的課程進行融合成網頁這讓我有非常大的成就感。 3. 讓我不會去糾結要選什麼主題。 4. 跨課程結合的專題，讓我覺得很有意思，一半是我擅長的領域，一半是我不擅長的。
3. 對於專業性的提升，思考有關職涯與主修，你的感受為何？	1. 這堂課我學到最多的就是與人的相處。 2. 可以在學校就可以先模擬以後如果可能要去其他大公司招標你自己的案子。 3. 這些技能應該是以後出社會都用得到的。 4. 我覺得我的電腦程式能力提升相當多。
4. 這門課及期末專題，你遇到的挑戰為何？	1. 對網頁設計基本概念不太深入的瞭解，所以在做作業的時候比較容易出現發生錯誤的問題。 2. 十分具有挑戰性，因此每次的作業都搞到很晚。 3. 程式碼與範例的相同但自己做的時候會跑不出來 這方面就花蠻多時間才處理問題的。

	<p>4. 是個困難的領域，原本應該不會去碰，但因為課堂需要，需要去學習。</p> <p>5. 課程一開始真的很難懂，是到實作才對老師說的東西有一點理解。</p>
--	-----------------------------------------------------------------------------------

4.2.3 教師教學省思與教學建議

本研究之建議與省思如下：

1. 強調實作的體驗式學習，有助於提升學生學習效能。
2. 分組進行期末專題的製作，因每位學生的程度與能力不同，雖然對於某一些資訊技術較強的學生會有不公平的現象，但經歷良好的溝通與協調後，有助於朝向互補性的分工，對於未來職場的團隊工作是正向的。
3. 與其他課程結合，進行指定主題之專業服務學習作為期末專題，與未指定期末專題主題相較下，更能讓學生有成就感與自信心。
4. 融入奧林匹克教育的期末專題，能提升學生之學習興趣與學習成就。

4.3 資料庫管理課程(110 年教育部教學實踐研究計畫)

4.3.1 學生學習回饋量化分析

一、學生課程學習自我效能量表

本研究設計了課程學習之自我效能量表，分別於第 8 週及第 18 週進行前後測，以了解學生之學習成效，問卷之問題與結果統計如表 18。

因為修課人數的關係，取得之樣本數僅有 18，考量樣本數過少且可能不符常態分布，因此採用 bootstrap 1000 的方式，進行成對樣本 t 檢定。統計結果呈現前後測之自我成效均有達到顯著。其中，各問題之前後測平均值改變最小的是資料庫的設計文件方面(Q13)，而信心程度增加最多的是有關 SQL 語法的部分(Q9)。也就是說，利用 Lahman DB 的確有提升自我成效達顯著，且對於未曾學過的專業技能，如 SQL(Q9、Q10)、Access(Q3)、資料庫管理系統(Q4)，自我效能改變最大。而對於設計文件之閱讀理解(Q13)、操作(Q8)、，所需的專業度較低，改變幅度最小。

表 18 學生課程學習自我效能量表與施測結果

問題	前測		後測		前後測 差異	顯著性 (雙尾)
	平均值	標準差	平均值	標準差		
1. 在一些棒球資料的 Web 網站查詢資料，對我來說不難。	3.00	1.237	3.83	0.786	-0.833	0.002**
2. 我知道如何使用 Excel 來管理大量資料。	3.33	1.029	4.11	0.832	-0.778	0.006**
3. 我知道如何使用 Access 來管理大量資料。	2.28	1.018	3.72	0.752	-1.444	0.000***
4. 我知道如何使用資料庫管理系統來管理大量資料。	2.50	1.043	3.94	0.802	-1.444	0.000***
5. 如果有其他資訊人員要幫我建置資料庫時，我能在參與資料庫建置專案的過程中，一步一步地清楚說明我的想法。	2.89	1.023	3.67	0.686	-0.778	0.001**
6. 我知道要建構資料庫時，大略可分成哪些階段。	3.06	0.998	3.94	0.725	-0.889	0.002**
7. 我知道如何將建置好的資料庫，提供給其他使用者連線存取、使用。	2.22	1.114	3.50	0.786	-1.278	0.000***
8. 我知道怎麼在資料庫中，新增、刪除、修改、查詢資料。	3.28	1.018	3.83	0.707	-0.556	0.020**
9. 我能使用 SQL 語法，新增、刪除、修改、查詢資料。	1.94	1.056	3.61	0.850	-1.667	0.000***
10. 我能修改、更正查詢資料庫時的 SQL 語法。	1.94	1.056	3.50	0.924	-1.556	0.000***
11. 我了解資料庫設計時如何進行正規化。	2.22	1.003	3.61	0.916	-1.389	0.000***
12. 我能夠依據需求，設計資料庫的表格(欄位)。	3.00	1.188	3.94	0.639	-0.944	0.001**
13. 我能看懂資料庫的設計文件(圖或表)。	3.33	1.188	3.83	0.707	-0.500	0.058
14. 我有信心能有效率地管理大量的資料。	2.67	1.138	3.44	0.784	-0.778	0.003**
15. 我能利用目前具備技能，解決一些資料處理或使用上的問題。	2.94	1.056	3.67	0.767	-0.722	0.008**

二、學生學習困難量表

為了知道學生對於各個教學單元的反應，本研究設計了學習困難量表，於第 18 週進行施測，與前一學年(2020)之同一課程做前後期比較，進行獨立樣本 t 檢定，問卷之內容及施測結果統計，如表 19。

前一學年(2020 年)之資料庫管理課程，教學單元之順序安排未採用逆向設計，雖然部分教學實例也採用 Lahman DB，以及作業、專題採用了運動的情境，但在講解上亦有採用校園教務選課作為案例。施測結果顯示，與前學年之資料庫管理課程，各題之前後期差異均未達顯著。前後期各題之平均值，改變最大的是 ER-model(Q2)。依各題平均值排序，較易遭遇到學習困難依序為：

前期(2020 年): ER-model(2.67)>SQL(2.93)>R-DB(3.07)>正規化(3.13)

後期(2021 年): SQL(2.89)>正規化(3.00)>R-DB(3.06)>ER-model(3.17)

而本研究實施之課程(後期:2021 年)的 SQL 最難，但自我效能最佳!

表 19 學生學習困難量表與施測結果

問題	2020 (N=15)		2021 (N=18)		前後期 差異	顯著性 (雙尾)
	平均 值	標準 差	平均 值	標準 差		
1. 我在學習「關聯式資料庫簡介」上沒有遭遇困難	3.07	0.704	3.06	1.162	0.011	0.974
2. 我在學習「ER model」上沒有遭遇困難	2.67	0.816	3.17	1.098	-0.500	0.155
3. 我在學習「第三正規化」上沒有遭遇困難	3.13	0.990	3.00	1.138	0.133	0.725
4. 我在學習「SQL 語法」上沒有遭遇困難	2.93	1.223	2.89	1.183	0.044	0.916
5. 對於「依體育相關資訊系統之情境規劃需求、繪製 use case 及 ER model」課程實作的安排，能讓我更深入了解學習的內容	3.47	0.915	3.28	0.958	0.189	0.569
6. 對於「使用 lahman 棒球資料庫，規劃情境與檢索需求，安裝並檢索資料」課程實作的安排，能讓我更深入了解學習的內容	3.53	1.060	3.39	1.145	0.144	0.712

三、學生學習評量及滿意度之比較

學生學習評量的結果，本研究實施之課程(2021年)與前一學年(2020年)的課程相較，在使用同一份考題之期中考之基礎上來檢定學生學習成效之差異。考題之題型及配分比例為：簡答題四題(16%)、正規化案例推導(20%)、資料檢索 Join 案例推導(20%)、ER Diagram 案例繪製(20%)、SQL 語法選擇題(24%)。測驗的結果，2020年的成績平均值為 58.69，標準差為 24.367，2021年則為 81.69 和 12.339。兩學期之學生成績，進行了前後期之獨立樣本 t 檢定，得到 $p=0.003$ ，達到顯著。也就是說，兩個學期的成績是有顯著差異的，2021年的成績有顯著的進步。前後期較大的差異在於 2020 年的標準差極大，主要原因是學生在非選題的部分，某些題目完全沒答題留空白的學生數較多，而 2021 年同樣情形的學生數則不多。

而在學生對於課程的滿意度部分，本研究依學校於期末各個課程施測之教師教學評量，來進行探討。在李克式五點量表滿分 5 分下，2020 年與 2021 年的分數分別為 4.61 與 4.01。2021 年實施創新教學之滿意度反而降低，可能原因是因為計畫的關係需填寫一些問卷、受訪，以及一些教學活動強迫學生在課堂上參與，可能使學生覺得學習負荷較大所致。

4.3.2 學生質性訪談意見整理

本研究於第 18 週課程之最後一週，對修課的學生進行訪談，訪談的問題及回答內容經編碼分類後依出現的頻率彙整如表 20：

表 20 學生訪談意見回饋

訪談問題	常見回答重點節錄
1.對於印象最深的單元(或作業)為何?	1. Lahman DB 找資料。 2. 畫 ER-model 的圖，很有成就感。 3. 期末的統整專題。
2.感到最困難的單元?	1. Access、SQL 沒學過，記不住，操作無法立即跟上。 2. 遇到困難都是問同學。 3. Lahman DB 中的技術指標術語。 4. 遇到問題網路找資料，自主學習找答案，最後才是問老師。
3.哪方面成長最多	1. 思考邏輯 2. 團隊溝通
4.未來學習之建議?	1. 不用害怕 2. 認真學 3. 問同學或上網找資料 4. 先備知識 5. 銜接教材

4.4.3 教師教學省思與教學建議

本研究之建議與省思如下：

1. 棒球主題情境、Lahman DB 實例、逆向設計、以專題為基礎之學習，有助於提升學生學習之效能。
2. 對於課程難易度，以及想達到的 Bloom 教育目標分類的層級，拿捏不易且會影響到設計的課程。
3. 個別單一課程的變革，未來可朝向整體串聯學程內每一門課程之改革。
4. 藉由教學實踐研究計畫可鼓勵教師進行教學創新，但要達到被研究期刊接受發表的水準，仍有待努力。

4.4 資料科學課程(109 年教育部教學實踐研究計畫)

4.4.1 學生學習回饋量化分析

本研究為比較自我效能量表的施測結果在前後測的差異，將李克氏五點量表的結果決定為次序尺度 (ordinal scale)，並使用 Kolmogorov-Smirnov 和 Shapiro-Wilk 檢定確認數據的為非常態分配(non-normal distributions)，故採用魏克生符號檢定 (Wilcoxon signed-rank test) 來確認同一個樣本在前後測差異之顯著性。顯著水準 α 定為 0.05，採雙尾檢定，前後測施測的信度 Cronbach's Alpha 值分別為 0.815 和 0.921，個別題項之分析結果整理如表 21。整體來看，除了學習動機於

課程尾聲未再提升外，學習興趣、GRR 教學對知識理解的幫助、GRR 教學對信心程度的增強、開放教育資源介入、專業知識的理解程度、整體能力的信心程度，後測均較前測有所增加。學習動機的降低，可能的解釋為，學生在修課後對於知識與技能已有了基礎，求知慾不再那麼強烈。而從前後測各問項的平均值來看，學生認為採用運動大數據分析實例來教學，可提升學習興趣 (問項 2)，認同程度最高。而前後測平均值差異最大者，為修課後對於運動大數據分析能力提升之信心程度 (問項 7)。前後測的差異，在 7 個題項中有 3 個題項呈現顯著差異，分別為題項 3、5、7，代表著學生對於大數據分析能力的自我效能在課程學習後確有顯著提升。

然而，開放教育資源的介入雖然有助於提升學生的大數據分析能力 (問項 6)，但未達顯著，探究成因可能與學生寫程式的能力有關。GitHub 和 Kaggle 這類的軟體庫雖然提供分析程式案例讓學生可以臨摹、修改、創新試驗，降低學習的焦慮，但程式稍有不慎即有 bug，會降低自我效能的評估。對於學生而言，程式設計仍是學習大數據分析上最大的挑戰。

表 21 學習體驗自我評量問卷統計結果(N=31)

問題	平均數(標準差)		Z 值	P 值
	前測	後測		
1. 我很想瞭解如何運用“資料科學”來進行“運動分析”的探討。	3.71(0.86)	3.68(0.83)	-0.254	0.800
2. 我認為在“資料科學”的教學中，使用“運動分析”做為案例，可提升學習的興趣。	3.84(0.78)	4.03(0.87)	-1.237	0.216
3. 目前，我對“運動分析”了解的程度是清楚的。	3.19(1.05)	3.74(0.82)	-2.112	0.035*
4. 藉由實作練習作業，我對“運動分析”的理解比以前更深入。	3.74(0.82)	4.06(0.85)	-1.699	0.089
5. 在完成實作練習作業之後，我有信心將“運動分析”處理得比以前更好。	3.52(0.85)	3.90(0.79)	-2.049	0.040*
6. 練習處理 GIT/Kaggle 上之開放程式資源，對我的“運動分析”能力發展有幫助。	3.52(0.81)	3.87(0.85)	-1.527	0.127
7. 現在我有信心將“資料科學”應用於“運動分析”上，同時處理完善。	2.84(1.00)	3.48(0.77)	-2.737	0.006*

這類與程式相關的課程，在學習上會因性別的不同而有學習興趣(Sax et. al, 2017)、學習焦慮(Tsai, 2002)，和學習成效的差異(Rubio et. al, 2015)。因此，本研究進一步探討性別之差異，將原始 31 個樣本分成男生 (N=18) 與女生 (N=13) 兩個群組，探討各個題項之前後測在男女兩個群組間的差異，進行魏克生等級和檢定 (Wilcoxon rank-sum test)。兩個群組不論是在前測或後測，七個問項的問卷作答結果其檢定之 p 值均大於 0.05，性別對於本課程之實踐成果並無顯著差異。

學生對於整學期課程的學習滿意度，可從學校統一於期末施測的教學評量問卷結果得知。該問卷共有十個問項，採用的是李克氏五點量表。運動大數據的課

程，於 109 和 110 學年本研究期間的填答率分別為 100%、81%，平均分數分別為 4.54、4.51。對比於本研究進行前 107 和 108 學年的同一門課程，填答率分別為 78%、92%，平均分數分別為 4.21、4.42，數據顯示學生對於課程的滿意度有具體的提升。

4.4.2 學生訪談意見質性分析

對學生單獨個別訪談的紀錄，依循紮根理論之步驟(Strauss & Corbin, 1990; 高麗娟 & 黃光獻, 2014)，以學習挑戰、學習成長、學習建議為基礎進行逐步編碼，最後得到兩大主軸構面如下：

一、引入開放教育資源對學習的影響

網站與部落格：職業運動中與大數據分析相關的開放教育資源非常多，例如：NBA、MLB 官網和提供數據的網站，雖然大多數都曾接觸過這些網站，但在修課時學生才有機會深入瞭解這些網路上的資源，這也讓學生充滿的探究的熱情，達到提升學習動機的目的。

“我自己平常是對 NBA 籃球比較有興趣，數據也都比較看得懂，那如果是 MLB 的話就是要再去了解一些數據方面的東西。”

“以前不知道 NBA 原來這麼多相關數據第一次看到覺得還蠻有趣的，報告就做得很認真。”

影片：本課程中常採用 YouTube 影片，做為課程模組一開始的介紹影片以及引發學習興趣使用，讓學生留下新奇、深刻的印象，有助於後續各教學單元學習動機之提升。

“我覺得比較有印象深刻是老師放一些籃球的研究的影片，比如說可能透過戰術的一些分析或者是某一個球員他很注重在哪一個點進行投射的那種”

軟體庫：許多學生提到是在這門課中首次接觸到 GitHub 與 Kaggle，覺得認識這些資源對於學生將來的發展會很有幫助，願意深入去了解它，能提升學習動機。

“找 NBA 跟 MLB 這個，用 Kaggle 去找。因為之前有聽過這些網站，但是也沒有實際過來查，就是有實際查過就發現其實這邊蠻多資訊可以看的。”

“知道說有這個網站可以去查跟程式跟數據有關的東西，就是多一個管道可以去查資料，因為以前不知道。”

“籃球數據跟棒球數據分析，就是都蠻有趣的，然後也利用 Kaggle 跟 GitHub 有找到蠻多資料，然後分析的內容也蠻感興趣的這樣。”

困難與挑戰：當採用開放教育資源教學時，有一些限制存在。這些資源大多數是英文的，且常有一些專業的術語，雖然可以直接用瀏覽器翻譯成中文來閱讀，但對部分英文不佳的學生，是一個學習上的門檻。

“英文要好一點，因為我們很多網站都是就是是英文的網站，就是你要看得懂英文才比較好理解他的內容。”

“一些比較比較複雜的運動數據專有名詞都會用縮寫，所以其實全英文的話直接翻譯也不太會知道他的那個縮寫到底在講什麼”

二、GRR 對於學習之影響

專題導向學習：採用 GRR 教學方法，在最後 You do it alone 的階段，學生需進行模組專題之發表。這個部分是一種專題導向學習的概念，學生對於這種學習方式的回饋是正向的，樂意用這種方式來學習。

“以前課堂比較少會有這種去研究一個項目然後去跟大家分享的這種型態，所以我覺得之後也可以把這種形態應用在更多的科目上面。”

“之前比較說是老師做一個我們做一個，這堂課就是比較多自己去發揮的空間，學比較多。”

“透過專題就可以去了解一些可能資料管理的方式跟做法，資料整理就是還蠻讓我懂得怎麼收集資料的。”

做中學：採漸進釋放責任的方式引導學生使用 Python 程式進行大數據分析，也讓學生增加許多技術實作的機會，經過一次一次的經驗累積，熟悉這些大數據分析的技術，而學生親手看到實際做出來了也增加了成就感，達到讓學生滿意的學習成效。

“因為接觸這些實作，就是在做專題這幾次這樣慢慢一次一次比較更了解。”

“可以透過這些數據然後用程式的方式顯現出來。”

“程式實際去使用，就是學以致用的感覺，就是學的比較多。”

“有辦法去使用別人寫好的東西，然後甚至是去進一步的修改這樣。”

“能夠看得懂那些分析出來圖表，然後可以去進行分析，然後了解它。”

主動學習：運動大數據分析涉及到的一些運動表現指標、專業術語、程式語

法，不同學生間存在著知識的落差，在有限的課程時數下需藉由課後主動學習來彌補。從學生的訪談中，看到了學生積極主動學習的態度。

“要去找資料，直到找到可以用的那個資料，就是可能會認識了非常多不同的東西，或是那個程式編寫的一些方法。”

“程式要實作的那個部分，一開始總是會失敗，就是有時候會找不到原因。”

“我對程式碼沒有很熟悉到可以一下就可以完成，所以還要上網查資料”

綜合上述，開放教育資源對於學生動機的提升是正向的。而 GRR 的教學模式，能與專題導向學習、做中學、主動學習的概念相互結合，對於提升學習成效是有幫助的。

4.4.3 教師教學省思與教學建議

本研究採用 GRR 結合 OER 的教學方式，逐步帶領學生分析籃球、棒球數據，結果顯示學生於課程學習前後之自我效能感、自信心和專業知識有顯著的提升。學生在過程中因為有機會收集、分析和視覺化資料與他們的體育主修背景相關的資料，自主性和相關性可能會增強學習的成效(Wallace & Kernozek, 2017)。而這些行為也可以解釋為對自己執行特定任務的感知能力之個人信念所導致的行為，以自我效能理論來解釋(Bandura, 1977)。而在兩個年度課程的教學研究過程中，除了對於學生問卷之量化分析和質性訪談外，研究者也從觀察者之角度，以行動研究法(McNiff, 2013)，在教學與研究並進的過程中進行記錄與省思，相關發現及討論如下。

首先，兩個年度的課程教學研究中，均觀察到在第一個模組的專題報告，會有學生要求展延的情形。109 學年的課程，學生共分成了 7 組，其中有 3 組準備不周無法如期報告，展延至下次上課。而 110 學年的課程，學生全部分成 5 組，其中有 1 組提出展延。但在第二和第三個模組，則沒有展延報告的現象。探究原因發現，在第一個模組學習時，大數據分析對於學生來說一開始的學習門檻極高，且仍不習慣這類 GRR 的教學方式，會讓學生無所適從不知專題報告如何做起。因此當專題報告要求學生自行探究 Kaggle、GitHub、部落格等資源來做報告時，當下無法聚焦於特定的主題，需要觀摩當週已報告同學的案例後，方能具體知道怎麼做報告，以致延遲報告。然而，這一個學習門檻通過了，在第二和第三個模組報告時，則不再有延遲的現象。

其次，是學生課程的自由分組，兩次課程均各有一組的人數僅一人，其餘皆為 3 到 4 人。經訪談得知其原因在於學生認為自己一個人做報告比較快，且對自己有信心，想避免與組員分工、溝通協調耗時。而這兩個單人組的學習成就表現，一個是全班最佳，另一個卻是全班最差。此外，從訪談亦得知表現極佳之學生熱

衷於棒球統計，因此對於專題的報告非常投入，肯花時間研究。另一位表現較差者，則在於該生之人格特質使然，有問題時也不主動詢問老師或其他組同學，以致專題報告不完整得分較低。為了避免上述兩個的現象發生，建議當使用 GRR 教學時，應一併強化合作學習的部分，由教師主動介入學生的分組，限制單人一組，並且於報告前安排跨組間觀摩、交流、討論的機會，以更落實合作學習，發揮組內共學、組間互學的效益。而本研究的問卷數據以性別分群進行分析時，顯示男性與女性在學習上並無顯著差異，推測可能是因為課程實施自由分組，進行合作學習有關。

另一個問題，是開放教育資源的問題，在課程中常發生教學案例中的程式碼無法順利執行。本研究有關大數據分析之程式技能教學，是以 GitHub 以及 Kaggle 的軟體庫當案例，融入到 GRR 教學模式中。學生可以直接在網站的編譯器，直接修改程式碼、參數值之後，按下執行網頁中的送出按鈕即可看到執行的結果。然而，當學生進行到 You Do It Alone 階段時，需移植網站上的程式碼到自己的電腦上來做自己的分析時，常遇到程式碼出錯而無能力除錯的問題。這類 Github、Kaggle 的問題，在一些教育的文獻中也曾被提出 (Zagalsky et al., 2015; Chow, 2019)。在經過幾次協助學生除錯的過程中發現，最大的問題在於這些 Python 的程式碼與套件的使用，有版本上相容的問題。解決方法是要教會學生，如何依照開放軟體之 Python 環境說明，來設置對應版本的套件，但這部分因時間限制，並沒有在課程範圍中，需在先備知識的課程中加強。

最後，在學生的專題報告上，發現到有少數組的報告，報告的重點不同，但是參考自同一個開放教育資源。經訪談後，發現其原因在於開放教育資源雖多，但難易程度不一，能引起興趣、能讀懂的往往是同一項資源。這現象與主動學習的特性有關，學生在主動學習時，會自然地忽略難以理解、較無趣的內容，而專業知識與能力，會使主動學習的效果受限。在施予一段時間課程教學後，範圍廣大的大數據分析技術對學生來說仍是困難的，因此會選擇适合自己程度的基礎入門內容。

本研究在實際實施一學期課程後，提出之結論如下：

1. 運動分析實務問題與資料科學兩者融合為一，可提升學生之學習動機。
2. 利用開放教育資源的學習經驗，有助於提升學習成就上的自信心。
3. 學生之間所具備之程式語言基礎及運動知識存在著顯著差異。
4. 對於利用資料科學進行運動實務問題的分析，學生感到最大的學習困難為程式的實作，以及有關運動統計及其計算公式之英文詞彙。
5. 利用開放教育資源的學習經驗，有助於提升學習成就上的自信心。

而對於未來教學上，提出之建議與省思如下：

1. 在有限教學時數下，融合了兩種學科進行教學，課程須設定為入門程度，無法探討到進階的應用。在課程推廣上，也可考慮定位在通識課程中實施。
2. 利用開放教育資源來開發教材，未來可考慮利用更嚴謹的決策方法(如：

德菲法)加入專家和學生的意見，來挑選更合適的教材。

3. 課程的設計應搭配更完整的課程地圖之配套措施，在先備課程先建立好學生之基礎能力。

5. 方法或應用之創新與貢獻

5.1 多元創新教學設計的行動流程——雙鑽石模型

本技術報告的目的，是描述如何將運動場景結合多元化的教學方式，應用於運動相關科系學生之資訊技術實務課程教學。解決非資訊科系的學生對於資訊技術實務課程學習上，常見之學習興趣低落、學習成效不佳之問題。資訊技術實務課程之教學實務研究，由一系列四門相關課程組成，採用的多元教學方式，依修課順序為：(1) 程式設計；(2) 網頁設計；(3) 資料庫管理；(4) 資料科學，採用的多元教學方式包括：講述法、ADRI 教學模式、成對編程、自動程式評測平台、合作學習、做中學、體驗式學習、情境式學習、逆向設計教學順序、專題導向式學習、遊戲式角色扮演、漸進釋放責任模式等概念、服務學習。研究中所蒐集到的資料顯示，學生的學習興趣，在課程以運動場景和多元教學法重新設計後，因可與自身主修學科背景產生連結，達到提升學習興趣的目的。而在學習成效上，透過動手實作的教學活動設計，可增強學習的自信心與成就感，透過自我效能量表前後測的檢定呈現出有顯著的差異。課程及教學實踐研究創新之處整理如表 22：

表 22 教學實踐研究課程之創新

課程	創新特徵說明
程式設計	ADRI 教學模式結合成對編程與線上評測平台
網頁設計	融合動手做之「體驗式學習」和「遊戲化專題導向學習」，並引入服務學習、主動學習之概念於課程活動設計中，降低學生面對 Html、CSS、jQuery、Bootstrap 等諸多生硬的技術詞彙時之學習焦慮，讓學習活動變的有活力、更有趣。課程將從做中學的體驗學習中建立基礎素養，以專案需求來延伸學習，最終整合應用這些推陳出新的網頁技術，與「奧林匹克教育」課程結合，依服務學習對象的真實世界需求，建置專題網站。
資料庫管理	將大眾日常所熟悉之運動休閒興趣與專業之資料庫技術融合在一起，發展以「棒球」為主題之情境來引導學習，並結合實例觀摩與實務操作
資料科學	將「運動分析」與「資料科學」兩門分屬不同學科的課程進行跨領域融合，以開放式教學資源建構教材，並採取整合「主動學習」、「情境式教學」、「做中學」、「合作式學習」、「專題導向式學習」、「漸進釋放責任模式」概念之教學模式與策略，於課程中實踐

從這一系列的教學實踐研究中，送審人統整出創新教學模式的發想方式，類似於設計思考(design thinking)常用的雙鑽石模型(the Double Diamond model)，適

合用於思考教學現場問題的考解決之道，而有別於一般教育學特定教學理論研究的作法。送審人所處的教學環境，多數課程是修課人數約 20 位的小班上課，且同學期僅授課一個班級。因此教學研究常被採用的準實驗法，或是以高低成就分組統計驗證教學成效等研究方式，並不適用於教學實踐研究。故在進行教學實踐研究的構思時，並沒有預設要採用哪一種教學理論或教學模式，進行該理論的文獻回顧，而是檢索出同樣課程的教學文獻，從文獻案例探討中來思考哪一個學模式較能夠解決自己所面臨的教學現場問題，依雙鑽石模型之探索(Discovery)、定義(Definition)、發展(Development)和執行(Delivery)四階段進行教學實踐研究，如圖 12 所示。

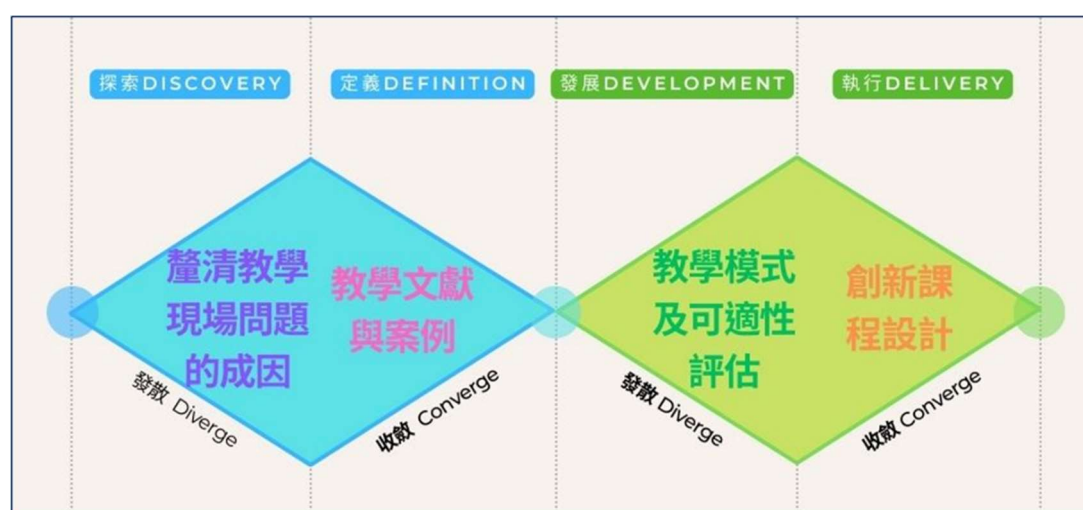


圖 12 課程創新之雙鑽石設計流程

首先在探索階段，送審人以教師觀點透過課堂上的觀察和與學生的互動，釐清教學現場的問題。並在此階段大量閱讀相關課程的研究文獻，以擴展視野與想法，看看其他學者是否有面臨相同的問題。接著，在定義的階段，找出文獻中共同提到的教學現場問題，加上自身經驗來進行聚焦，收斂出教學實踐研究的問題。第三階段的發展，則從先前相關課程的文獻中，發掘可供採用的教學模式清單，逐一擴展文獻的閱讀至其他採用這些教學模式的課程。最後在執行，將眾多教學模式清單依課程屬性及可行性來進行評判，加以收斂，決定該課程所要採用的教學模式，並吸收文獻案例的經驗與特色，來設計 18 週的課程內容。因此本報告各課程的教學實踐研究，解決方案採取的是多元化的教學模式，以符合教學現場的需求。

此外，本研究另一個創新在於跨課程間的整合，不侷限於單一課程和單一的教學模式，如圖 13 所示。本研究教學實例為：「資料科學」與運動的「技戰術分析」課程。前者強調演算法和程式，後者強調統計學，但兩者可在的相同運動場景中進行數據分析上找到共同點而加以整合。教學上同樣是分析 NBA 球員的效率，但使用的方法不同。另一個跨課程的教學實例為：「網頁設計」與本校通識

中心校定必修的「奧林匹克教育」課程相互結合。讓學生使用前者學習到的網頁設計技能，建置後者奧林匹克教室的網站，作為期末專題。學生扮演的不僅是網頁設計師，也需在修過奧林匹克課程後，知道業主要的是什麼，主動蒐集素材，規劃網站內容。



圖 13 跨課程間的整合之概念

5.2 貢獻

送審人自 108 年度起，開始投入教學實踐研究，致力於引入新的教學模式來提升學生的學習興趣與成效，讓更多學生具備符合未來職場實務需求水準的資訊素養和就業力。在教育部教學實踐計畫的支持下，連續四年陸續以「程式設計」、「運動資訊管理」（資料科學）、「資料庫管理」、「網頁設計」課程進行教學實踐研究。於教學上分別導入 ADRI 教學模式結合成對編程與線上評測平台、結合運動場景和運動分析情境之大數據分析、以棒球資料庫實例結合反向教學順序策略、融入角色扮演及專業服務學習專題之網頁設計，在運動與休憩學群中的教學中，進行資訊技術教育之教學革新。近年來也積極參與教育部「推動大學程式設計教學計畫」、「教學實踐研究計畫」各區域基地辦理之研討會、工作坊，以及「苗圃計畫」之設計思考種子教師培訓，與領域相近的同道、先進們共同交流，吸收成功的教學經驗。

在學術上，目前教學實踐研究雖然已有大量的體驗式學習、情境式學習的文獻，但多數場景為商業和管理的應用領域，適合主修運動或休憩領域學生的運動

場景教學課程，仍存在著教學研究的缺口。本研究的主要貢獻主要有三：首先，對於如何以運動場景來設計資訊技術實務課程之教學上，提供了參考。其次，示範了如何整合相近的課程進行跨課程教學，以達到兩門課互補學習的共效。最後，本技術報告一系列課程群組的教學實踐研究中所提供的經驗與見解，可協助其他科系就容易出現學習困難、學習焦慮、學習成效不佳的學科進行整個課群一系列的教學變革。

參考文獻

中文部分

- 王啟恩、李昌霖、陳璽文、劉以樂 (2016)。看 MLB 學英語。希伯崙。
- 王曉寒 (2008)。看棒球學英文-輕鬆學習棒球相關英文。臺灣商務。
- 李中傑 (2016)。球物理大聯盟：王建民也要會的物理學。五南。
- 蔡慧貞、任曉晶 (2019)。行動網頁設計課程融入 PBL 創新研究與實踐。台
北海洋科技大學學報, 10(1), 33-47。
- 高嘉峻、蔡銘修 (2018)。淺談十二年國教在國民中小學推行科技領域課綱資訊
科技學習之障礙與策略。臺灣教育評論月刊, 7 (2), 80-84。
- 高麗娟、黃光獻 (2014)。體育運動質性資料分析方法。體育學報, 47(2), 159-177。
<https://doi.org/10.6222/pej.4702.201406.1201>
- 教育部電子報 (2020 年 7 月 2 日)。六成大學生修過程式設計相關課程 教育部
「推動大學程式設計教學計畫」見成效。取自 [https://epaper.edu.tw/
print.aspx?print_type=topical&print_sn=1154&print_num=928](https://epaper.edu.tw/print.aspx?print_type=topical&print_sn=1154&print_num=928)

英文部分

- Abid, A., Farooq, M. S., Raza, I., Farooq, U., & Abid, K. (2015). Variants of Teaching
First Course in Database Systems. *Bulletin of Education and Research*, 37(2), 9-
25.
- Adair, R. K. (2002). *The physics of baseball*. Harper Collins, New York.
- Albert, J. (2002). A Baseball Statistics Course. *Journal of Statistics Education*, 10(2),
Article 2. <https://doi.org/10.1080/10691898.2002.11910663>
- Archer, A. L., & Hughes, C. A. (2010). *Explicit Instruction: Effective and Efficient
Teaching*. Guilford Publications.
- Asamoah, D. A., Doran, D., & Schiller, S. (2015). Teaching the Foundations of Data
Science: An Interdisciplinary Approach. 2015 Pre-ICIS SIGDSA Workshop, Fort
Worth, Texas.
- Bahad, P., Saxena, P., & Kamal, R. (2021). Exploratory and Predictive Analytics of
User Preferences from Kaggle LEGO-Toys Datasets Using Spark ML. *IOP
Conference Series: Materials Science and Engineering*, 1099(1), Article 1.
<https://doi.org/10.1088/1757-899X/1099/1/012019>
- Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioral change.
Psychological Review, 84(2), 191-215. [https://doi.org/10.1037/0033-
295X.84.2.191](https://doi.org/10.1037/0033-295X.84.2.191)
- Barneva, R. P., & Hite, P. D. (2017). Information technology in sport management
curricula. *Journal of Educational Technology Systems*, 45(3), Article 3.

- Barneva, R. P., & Brimkov, V. E. (2019). Teaching Data Analytics to Sport Management Majors: A Practical Approach. In *International Workshop on Learning Technology for Education in Cloud* (pp. 408-417). Springer, Cham.
- Brito, M. A., & De Sá-Soares, F. (2014). Assessment frequency in introductory computer programming disciplines. *Computers in Human Behavior*, *30*, 623-628.
- Cai, X., Shen, W., Cui, Y., Meng, X., Zhang, W., & Yan, Z. (2020). Application Research of Project-based Teaching Based on Working Process in Oracle Database Teaching. *MS&E*, *750*(1), 012003.
- Carpita, M., & Golia, S. (2021). Discovering associations between players' performance indicators and matches' results in the European Soccer Leagues. *Journal of Applied Statistics*, *48*(9), Article 9. <https://doi.org/10.1080/02664763.2020.1772210>
- Chen, X. (2019). Design of a database graduate course as a leveling class for non-CS major graduate students. *Journal of Computing Sciences in Colleges*, *35*(1), 32–43.
- Chow, W. (2019). A Pedagogy that Uses a Kaggle Competition for Teaching Machine Learning: An Experience Sharing. *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, 1–5. <https://doi.org/10.1109/TALE48000.2019.9226005>
- Chuang, K. L., Kee, Y. H., & Chen, H. H. (2022). Implementation of the gradual release of responsibility informed curriculum and pedagogy for teaching programming: Action research based on a course for sport science students. *Journal of Hospitality, Leisure, Sport & Tourism Education*, *30*, 100367.
- Chung, S. (2018). Data Analytics Workshop Series for Non-Computing Major First-Generation-College-Bound Students. *Information Systems Education Journal*, *16*(5), 37.
- Connolly, R. (2019). Facing Backwards While Stumbling Forwards: The Future of Teaching Web Development. *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 518–523. <https://doi.org/10.1145/3287324.3287433>
- Dobesova, Z. (2016). Teaching database systems using a practical example. *Earth Science Informatics*, *9*(2), Article 2. <https://doi.org/10.1007/s12145-015-0241-3>
- Domínguez, C., & Jaime, A. (2010). Database design learning: A project-based approach organized through a course management system. *Computers & Education*, *55*(3), 1312–1320.
- Downes, S. (2019). A look at the future of open educational resources. *The International Journal of Open Educational Resources*, *2*.
- Duffy, G. G. (2014). *Explaining Reading, Third Edition: A Resource for Explicit Teaching of the Common Core Standards*. Guilford Publications.
- D'Souza, M. J., & Rodrigues, P. (2015). Extreme pedagogy: An agile teaching-learning methodology for engineering education. *Indian Journal of Science and Technology*, *8*(9), 828-833.
- Fang, A.-D., Chen, G.-L., Cai, Z.-R., Cui, L., & Harn, L. (2017). Research on Blending Learning Flipped Class Model in Colleges and Universities Based on

Computational Thinking—"Database Principles" for Example. *Eurasia Journal of Mathematics, Science and Technology Education*, 13(8), 5747–5755.

- Faraon, M., Rönkkö, K., Wiberg, M., & Ramberg, R. (2020). Learning by coding: A sociocultural approach to teaching web development in higher education. *Education and Information Technologies*, 25(3), 1759–1783. <https://doi.org/10.1007/s10639-019-10037-x>
- Fisher, D., & Frey, N. (2008). Better Learning through Structured Teaching: A Framework for the Gradual Release of Responsibility. In *Association for Supervision and Curriculum Development*. Association for Supervision and Curriculum Development.
- Folorunso, O., & Akinwale, A. (2010). Developing visualization support system for teaching/learning database normalization. *Campus-Wide Information Systems*.
- Guo, E., Xu, W., Yu, H., & Zhong, Y. (2021). Reform and Practice of Web Design and Production based on OBE Concept in the Context of New Engineering. *IOP Conference Series: Earth and Environmental Science*, 693(1), 012017. <https://doi.org/10.1088/1755-1315/693/1/012017>
- Hamzah, M. L., Rukun, K., Rizal, F., & Purwati, A. A. (2019). A review of increasing teaching and learning database subjects in computer science. *Revista ESPACIOS*, 40(26), Article 26.
- Hanks, B., Fitzgerald, S., McCauley, R., Murphy, L., & Zander, C. (2011). Pair programming in education: a literature review. *Computer Science Education*, 21(2), 135-173.
- Hollingsworth, J. R., & Ybarra, S. E. (2017). *Explicit direct instruction (EDI): The power of the well-crafted, well-taught lesson*. Corwin Press.
- Hsu, T.-H., Horng, G.-J., & See, A. R. (2021). Change in Learning Motivation Observed through the Introduction of Design Thinking in a Mobile Application Programming Course. *Sustainability*, 13(13), 7492. <https://doi.org/10.3390/su13137492>
- Johnson, C. N. (2016). The benefits of PDCA. *Quality Progress*, 49(1), 45.
- Jin, K. H. (2017). *Teaching Responsive Web Design to Novice Learners*. 6.
- Kearney-Volpe, C., Fleet, C., Ohshiro, K., Arias, V. A., & Hurst, A. (2021). Making the elusive more tangible: Remote tools & techniques for teaching web development to screen reader users. *Proceedings of the 18th International Web for All Conference*, 1–14. <https://doi.org/10.1145/3430263.3452418>
- Lee, S. J., Wilder, C., & Yu, C. (2018). Exploring students' perceptions of service-learning experiences in an undergraduate web design course. *Teaching in Higher Education*, 23(2), 212–226. <https://doi.org/10.1080/13562517.2017.1379486>
- Lin, Y. (2020). Student-Centered Teaching Design and Practice of Database Course. 149–153. <https://doi.org/10.2991/assehr.k.200727.033>
- Mason, R., Seton, C., & Cooper, G. (2016). Applying cognitive load theory to the redesign of a conventional database systems course. *Computer Science Education*, 26(1), 68–87.
- Malik, S. I., & Coldwell-Neilson, J. (2017a). Impact of a new teaching and learning approach in an introductory programming course. *Journal of Educational*

Computing Research, 55(6), 789-819.

- Malik, S. I., & Coldwell-Neilson, J. (2017b). A model for teaching an introductory programming course using ADRI. *Education and Information Technologies*, 22(3), 1089-1120.
- Malik, S. I. (2018). Improvements in introductory programming course: Action research insights and outcomes. *Systemic Practice and Action Research*, 1-20.
- McCartney, R., Boustedt, J., Eckerdal, A., Sanders, K., Thomas, L., & Zander, C. (2016). Why Computing Students Learn on Their Own: Motivation for Self-Directed Learning of Computing. *ACM Transactions on Computing Education*, 16(1), 2:1-2:18. <https://doi.org/10.1145/2747008>
- McNiff, J. (2013). *Action Research* (0 ed.). Routledge. <https://doi.org/10.4324/9780203112755>
- Medeiros, R. P., Ramalho, G. L., & Falcão, T. P. (2018). A Systematic Literature Review on Teaching and Learning Introductory Programming in Higher Education. *IEEE Transactions on Education*, (99), 1-14.
- Miller, C. S., & Connolly, R. (2015). Introduction to the Special Issue on Web Development. *ACM Transactions on Computing Education*, 15(1), 1:1-1:5. <https://doi.org/10.1145/2724759>
- Ortlieb, E., & Schatz, S. (2020). Student's Self-Efficacy in Reading – Connecting Theory to Practice. *Reading Psychology*, 41(7), 735–751. <https://doi.org/10.1080/02702711.2020.1783146>
- Palominos, F. E., Palominos, S. K., Durán, C. A., Córdova, F. M., & Díaz, H. (2018). Challenges in the use of a support tool with automated review in student learning of programming courses. *Procedia computer science*, 139, 424-431.
- Pošćić, P., Subotić, D., & Ivašić-Kos, M. (2012). Developing the course Database systems to respond to market requirements. *2012 Proceedings of the 35th International Convention MIPRO*, 1141–1145.
- Rashid, T. (2015). Investigation of Instructing Reforms in Databases. *International Journal of Scientific & Engineering Research*, 6(8), 64–72.
- Razvi, S., Trevor-Roper, S., Goodliffe, T., Al-Habsi, F., & Al-Rawahi, A. (2012). Evolution of OAAA strategic planning: using ADRI as an analytical tool to review its activities and strategic planning. *Proceedings of Seventh Annual International Conference on Strategic Planning for Quality Assurance and Accreditation of Universities and Educational Arab Institutions*.
- Rothman, S. (2012). *Sandlot stats: Learning statistics with baseball*. JHU Press.
- Rubio, M. A., Romero-Zaliz, R., Mañoso, C., & Angel, P. (2015). Closing the gender gap in an introductory programming course. *Computers & Education*, 82, 409-420. <https://doi.org/10.1016/j.compedu.2014.12.003>
- Sakibayev, S., Sakibayev, R., & Sakibayeva, B. (2019). The educational impact of using mobile technology in a database course in college. *Interactive Technology and Smart Education*.
- Saligumba, I. P. B., & Tan, D. A. (2018). Gradual Release of Responsibility Instructional Model: Its Effects on Students Mathematics Performance and Self-

- Efficacy. *International Journal of Scientific & Technology Research*, 7(8), 276–291.
- Shaffer, D. W., & Resnick, M. (1999). “Thick” authenticity: New media and authentic learning. *Journal of Interactive Learning Research*, 10(2), 195–216.
- Song, D., Tavares, A., Pinto, S., & Xu, H. (2017). Setting Engineering Students Up for Success in the 21st Century: Integrating Gamification and Crowdsourcing into a CDIO-based Web Design Course. *EURASIA Journal of Mathematics, Science and Technology Education*, 13(7). <https://doi.org/10.12973/eurasia.2017.00745a>
- Strauss, A., & Corbin, J. M. (1990). *Basics of qualitative research: Grounded theory procedures and techniques* (p. 270). Sage Publications, Inc.
- Tabbakh, A., Rout, J. K., & Rout, M. (2021). Analysis and Prediction of the Survival of Titanic Passengers Using Machine Learning. In A. K. Tripathy, M. Sarkar, J. P. Sahoo, K.-C. Li, & S. Chinara (Eds.), *Advances in Distributed Computing and Machine Learning* (pp. 297–304). Springer. https://doi.org/10.1007/978-981-15-4218-3_29
- Taipalus, T., & Seppänen, V. (2020). SQL education: A systematic mapping study and future research agenda. *ACM Transactions on Computing Education (TOCE)*, 20(3), 1–33.
- Tezer, M., & Çimşir, B. T. (2018). The impact of using mobile-supported learning management systems in teaching web design on the academic success of students and their opinions on the course. *Interactive Learning Environments*, 26(3), 402–410. <https://doi.org/10.1080/10494820.2017.1337037>
- Thabtah, F., Zhang, L., & Abdelhamid, N. (2019). NBA Game Result Prediction Using Feature Analysis and Machine Learning. *Annals of Data Science*, 6(1), Article 1. <https://doi.org/10.1007/s40745-018-00189-x>
- Tsai, M. J. (2002). Do male students often perform better than female students when learning computers?: A study of taiwanese eighth graders' computer education through strategic and cooperative learning. *Journal of Educational Computing Research*, 26(1), 67-85. <https://doi.org/10.2190/9JW6-VV1P-FAX8-CGEO>
- Tsompanoudi, D., Satratzemi, M., & Xinogalos, S. (2016). Evaluating the effects of scripted distributed pair programming on student performance and participation. *IEEE Transactions on education*, 59(1), 24-31.
- Tzafilkou, K., Protogeros, N., & Chouliara, A. (2020). Experiential learning in web development courses: Examining students’ performance, perception and acceptance. *Education and Information Technologies*, 25(6), 5687–5701. <https://doi.org/10.1007/s10639-020-10211-6>
- Vellappan, U., & Lim, L. (2021). STUDENTS’ LEARNING EXPERIENCE IN A WEB DESIGN COURSE USING FLIPPED CLASSROOM APPROACH IN HIGHER EDUCATION. *International Journal of Education, Psychology and Counseling*, 6(40), 193–206. <https://doi.org/10.35631/IJEPC.640016>
- Vihavainen, A., Airaksinen, J., & Watson, C. (2014). A systematic review of approaches for teaching introductory programming and their influence on success. *Proceedings of the tenth annual conference on International computing education research*, 19-26.
- Wang, J., & Chen, H. (2014). Research and practice on the teaching reform of database course. *2014 International Conference on Education Reform and Modern*

Management (ERMM-14).

- Wang, N., & Ma, C. (2017). Teaching reform on database course for science laboratory class. *2017 12th International Conference on Computer Science and Education (ICCSE)*, 310–313.
- Wang, S., & Wang, H. (2016). Renewal of classics: Database technology for all business majors. *Journal of Computer Information Systems*, *56*(3), 211–217.
- Wallace, B., & Kernozek, T. (2017). Self-efficacy theory applied to undergraduate biomechanics instruction. *Journal of Hospitality, Leisure, Sport & Tourism Education*, *20*, 10–15. <https://doi.org/10.1016/j.jhlste.2016.11.001>
- Webb, S., Massey, D., Goggans, M., & Flajole, K. (2019). Thirty-Five Years of the Gradual Release of Responsibility: Scaffolding Toward Complex and Responsive Teaching. *The Reading Teacher*, *73*(1), 75–83. <https://doi.org/10.1002/trtr.1799>
- Yang, X., & Ding, J. (2020). A Computational Framework for Iceberg and Ship Discrimination: Case Study on Kaggle Competition. *IEEE Access*, *8*, 82320–82327. <https://doi.org/10.1109/ACCESS.2020.2990985>
- Ying, F. (2016). Research on Blended Learning Mode Based on the Micro-Lecture in Database Application. *Review Of Computer Engineering Studies*, *3*(3), 62-66.
- Ying, H. (2021). The Application of Game Teaching Method in the Course of “Web Design and Production” in the Teaching Project. *2021 2nd International Conference on Education Development and Studies*, 55–62. <https://doi.org/10.1145/3459043.3459054>
- Zagalsky, A., Feliciano, J., Storey, M.-A., Zhao, Y., & Wang, W. (2015). The Emergence of GitHub as a Collaborative Platform for Education. *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, 1906–1917. <https://doi.org/10.1145/2675133.2675284>