**全速衝線**

****

中華民國 109 年6 月6 日

[**摘要** 4](#_Toc42798016)

[**1.簡介** 4](#_Toc42798017)

[**2.系統功能需求** 5](#_Toc42798018)

[**3.系統功能需求** 7](#_Toc42798019)

[**4.相關技術** 8](#_Toc42798020)

[**4.1遊戲式學習** 8](#_Toc42798021)

[**4.1.1 遊戲式學習的特性** 8](#_Toc42798022)

[**4.1.2遊戲式學習的優點** 9](#_Toc42798023)

[**4.1.3遊戲式學習的教材設計與應用** 10](#_Toc42798024)

[**4.2程式積木** 10](#_Toc42798025)

[**5.實作成果** 13](#_Toc42798026)

[**6.結論** 19](#_Toc42798027)

[**7.相關文獻** 19](#_Toc42798028)

[**附錄** 21](#_Toc42798029)

[**功能說明、畫面命名** 22](#_Toc42798030)

[**系統安裝** 37](#_Toc42798031)

[**1-1** **取得檔案：** 37](#_Toc42798032)

[**1-2** **安裝編譯器Atom(有其他編譯器可跳過此步驟，如：Visual Studio code)：** 37](#_Toc42798033)

[**1-3** **安裝Node.js：** 38](#_Toc42798034)

[**1-4** **安裝mongo DB：** 43](#_Toc42798035)

[**1-5** **安裝Robo 3T：** 47](#_Toc42798036)

[**1-6** **執行專案：** 50](#_Toc42798037)

[**1-7** **建立資料庫：** 52](#_Toc42798038)

[**1-8** **建立管理員帳號：** 52](#_Toc42798039)

[**玩家操作說明** 57](#_Toc42798040)

**全速衝線**

指導老師：江傳文 副教授

參賽組員：劉晉豪、謝正德、劉豐銘、林浚誠

**摘要**

本專題旨在實作一款名為《全速衝線》的程式教學平台，用以解決使用者在學習程式時所遭遇（1）學習時的枯燥感；（2）因無法得到好的回饋感而失去興趣；（3）程式過於抽象等問題。藉由遊戲式學習的方式，讓使用者可以邊玩遊戲邊學習程式的基本觀念，透過將程式碼的提交給系統後，遊戲畫面給予該段程式碼應有的回饋，讓使用者對撰寫程式有更好的體驗；並透過將程式語言轉變成積木的形式，讓使用者可以更直觀的了解程式整體的架構，圖形化的程式可以解決純文字程式看起來較為枯燥乏味的問題。

**1.簡介**

近年來，由於科技發達、資訊時代逐漸起步的牽引下，資訊科技已成為如今正火紅的話題

，並在教育制度的變遷下資訊科技也已納入了12年國教的課程大綱裡頭，而資訊科技首先要教的就是程式，為了要讓軟硬體有便利的服務，其背後也都必有一支能讓其運作的程式，無論是低階的程式語言還是高階的程式語言都需要理解其對應的程式語言，因此在這資訊科技的洪流裡，要學好程式不再只是為了獲得學業成績的卓越，也為了快速上手其他從程式的發展歷史中逐漸衍生出越來越多種的程式語言而奠定的基礎。傳統觀念上程式往往是枯燥的代名詞，對於一般民眾而言一提到程式相關的話題，都會直接聯想到長期坐在電腦桌前面對著冰冷的螢幕與清脆敲打鍵盤的聲音或是只有01的數字世界來代表著艱深的學問，那如此重要的科目該如何為學生打下良好的基礎就是一個值得思索的項目。

所幸，拜網際網路技術快速發展與各式應用蓬勃興起之所賜，學習不再是只能在課堂上抄著黑板的筆記、自習時看著課本、閒暇時看著額外的教材書，近期在「遊戲式學習[x]」上也開始有著熱烈討論，為了解決學習時經常會遇到的幾個問題，其中包含了學習初期階段往往是最迷惘的，不知道學了能用在哪裡、能有何作為而開始迷惘、造成學習怠惰，最後變成是被迫學習加上興趣被削減將會使得學習的路變得更加難以堅持，故遊戲式學習的方式開始被廣為人知，在程式學習的部分近期也發展出了以「程式積木[1]」來代替程式語言中各項指令彼此交錯的組合以及指令的純文字表式，但在實際的運作過程中可知，程式積木的重點變相的被著重於程式的可讀性以及簡單的積木組合，間接導致學習程式的過程缺少了與程式的互動，以及程式積木在遊戲式學習上往往都是各自獨立的積木語言，不能直接用於任何一種程式語言上，故本團隊以創新且好玩的方式將學習程式與遊戲融為一體。

本專題中，以遊戲式學習之精神為基礎，我們將研製一款名為「全速衝線」之視覺化編程環境供學生們藉由課程搭配以及自主學習方式精進自身程式設計解題能力。此一線上遊戲式學習環境具備有多項特色，包括：

* 透過遊戲方式進行學習：以計算機程式設計課程之教學內容為基礎，吾人將在「全速衝線」遊戲式學習環境中規劃至少 50 道程式設計任務關卡供學生們進行挑戰，此不僅可以增進學生們的學習興趣，使學生們有更多機會練習編寫程式，也能讓學生們學習到完整的教學內容素材。
* 運用積木組合以及文字編輯相互參考方式創建程式：在「全速衝線」遊戲式學習環境中，學生們可透過積木組合之視覺化方式創建其程式。對於程式設計初學者而言，此有助於減輕學習程式設計語法所遭遇到的困難。另一方面，對於已精熟程式語法的學生而言，則是可運用文字編輯方式創建其程式，以利於將學習焦點專注在問題解決層面。
* 藉由動畫方式呈現程式執行結果：學生們於闖關挑戰過程中所創建程式的執行結果可透過動畫即時反饋，也因此，程式執行時的抽象意念得以轉換為具體方式呈現，學生們將更能體會並理解其所創建程式的運作過程。
* 鼓勵自行設計挑戰關卡：「全速衝線」遊戲式學習環境提供任務編輯進階功能，鼓勵已獲致較佳學習成就的學生們自行設計任務關卡並接受其他同學們的挑戰，持續精進程式設計解題技巧。
* 精準預測學生學習成效：「全速衝線」遊戲式學習環境整合了倒傳遞神經網路

(Backpropagation Neural Networks) 與粒子群最佳化 (Particle Swarm Optimization, PSO) 技術，藉以發展用於預測學生學習成效之演算方法，協助授課教師調整教學策略與活動，進而具體而有效地落實教學創新並提升教師教學品質。

基於前述所提之各項特色，「全速衝線」遊戲式學習環境之發展概念模型如圖 3 規劃所示。



圖1.本專題所研製「全速衝線」遊戲式學習環境之概念模型心智圖



圖1.本專題所研製「全速衝線」遊戲式學習環境之概念模型心智圖(續)

《全速衝線》遊戲的遊玩方式是透過遊玩不同的關卡或遊玩不同玩家所設計的地圖來達到學習增長的效果，故本團隊在關卡設計上下足了功夫，不但將Ｃ語言程式的要點融入各關卡中還在後半部分的關卡中設計了特殊的遊戲機制讓玩家擁有獨立解決問題的能力，並在全破完本團隊所設計的關卡後玩家將可以活用Ｃ語言中的變數、函式、判斷式、迴圈、指標以及擁有解決問題的能力，且不論是在教育方面還是程式解題的基礎都會獲得良好的成效。

**2.相關技術**

**2.1遊戲式學習**

數位遊戲式學習(Digital game based learning)是指一種用數位遊戲的方式來輔助學習[2]，並利用電腦、手機、數位遊戲等相關媒介，讓學習者在遊戲中解決問題並挑戰通關，因而提升學習成效，也增加在解決問題的成就感，此方法同時兼具了教育性與娛樂性，相較於傳統較為枯燥乏味的教育方式，遊戲是一種能讓人放鬆舒壓媒體，而學習要能成功，最重要的一個關鍵就是學習動機，傳統的教育方式較容易讓學習者產生壓力，進而產生學習焦慮及降低學習意願，而將遊戲結合教育，藉由遊戲來降低學習壓力，是一種新的教學方式。

**2.1.1 遊戲式學習的特性**

Prensky(2007)[2]提到了幾個遊戲式學習吸引人的關鍵因素：

1. 娛樂性：以有趣的遊戲替代文字，讓使用者在邊玩邊學習的過程中覺得有趣
2. 遊戲性：有趣的遊戲內容，讓使用者有想玩遊戲的慾望及動機。
3. 規則性：遊戲的內容具有結構性，讓使用者在進行遊戲時，容易組織遊戲內容。
4. 目標性：在遊戲中設立明確的目標及指引，逐步引導使用者。
5. 人機互動性：使用者經由與電腦的互動及操作於遊戲介面來進行遊戲。
6. 結果與回饋：根據遊玩遊戲的過程及結果，獲得學習的機會及成效。
7. 適性化：依學習者能力不同，設立難易度不同的關卡或任務。
8. 勝利感：在遊戲過程中，透過完成任務讓學習者獲得滿足感。
9. 衝突競爭性與挑戰性：在遊戲中，與自我或其他玩家競爭以獲得刺激感及動力
10. 問題解決：在遊戲中設立問題，讓使用者在找尋答案的過程中，激發創意及增進自我
11. 社會互動：讓使用者自主組成遊戲社群，增進社會互動性。
12. 圖像與情節性：以圖畫及故事情節替代純文字的教學，讓使用者印象更加深刻。

**2.1.2遊戲式學習的優點**

數位遊戲在學習過程中具有趣味與好玩兩個重要的學習元素，因此數位遊戲可以讓學習者處在一個輕鬆且學習動機較強的學習環境，讓學習者可以透過遊戲式學習發展出數位科技時代所必須的基本技能及專門領域的知識。遊戲式學習可以帶入新的知識，增強技能，發現新的想法、概念和發展成果，而且數位遊戲還能刺激學習者腦力激盪，並且增加學習者的邏輯判斷能力來發展創新的方式以加強解決問題的能力。而在對青少年訪問的研究結果也指出，青少年認為遊戲式學習對於中學的科學及數學教育是有一定的幫助。根據Selnow與Reynolds（1984）的研究結果[3]指出，玩數位遊戲能夠讓青少年遠離日常生活上與學業上的問題，甚至獲得在現實生活中所無法獲得的主導權與成就感。綜合許多研究人員的研究發現遊戲式學習相較於傳統的學習方式或是一般的數位學習，數位遊戲所能提供給教育的優勢，可以分別從教與學兩個面向來加以探究，在教學方面，數位遊戲的學習教材不再只是傳統的純文字教材，遊戲式學習的教材可能包括單純的靜態圖片、動態影像、以及其他的聲光特效，讓學習者不僅僅是視覺的刺激，還有聽覺的刺激甚至是互動式體感刺激，提供了學習者較豐富的學習體驗，此外遊戲式學習能夠在不同的教育階段實施，不管是在老師講課時、體驗活動時、實作練習或者課外延伸都能加以適用，讓學習過程更為彈性，再者，雖然遊戲式學習在教材上的製作時間與心力花費相對傳統的上課教材來說都較高，但遊戲式學習的教材具有重複利用性，且能輕易透過修改內容細項來符合不同課堂章節所需要的教材，而且遊戲式學習不像是傳統教學較為呆板且枯燥乏味讓學生較無反應，遊戲式學習可以使得學生與老師有更多的互動，讓老師更能掌握及了解學生的學習狀況，進而增加學生的學習成效；而在學習方面，數位遊戲能讓學習者想起享受在學習中的樂趣，因為學習者在遊玩數位遊戲的過程中，相較於傳統學習方式，學習者是主動方，因此學生能自己掌握自己的學習狀況，從而降低學習焦慮，並提升學習動機，善用遊戲式學習可以促進「學習的典範轉移」。數位遊戲的設計可與真實情境作結合，所學習到的知識或技能較容易轉移至實際情況，這表示接受遊戲式學習的玩家在遇到新情境問題時，能更有效的、更有能力的在短時間內學習以前所沒有教導過的問題，且更正確的解決問題。且如前面所提到，學習者在遊戲式學習中，可以照自己的學習狀況調整學習進度，如在較不熟悉的章節或單元可以進行不斷的練習，因此遊戲式學習可以讓學習者達到個人化的學習，且學習成效可較傳統學習高。學習者從數位遊戲中學到對學習最有益處的能力就是解決問題。在遊戲中解決問題的過程，一般會經過以下的四個循環狀態：理解事件的前因後果，腦力發想或許可行的解決方案，實現解決方法，查驗結果並重複。因為在數位遊戲中多變的情境需要擬定不同的策略來解決問題，因此學習者在玩遊戲的過程中將會面臨到要自己思考解決方案，藉此訓練到學習者透過問題的癥結來擬定相對應的策略，這個能力讓學習者可以在真實世界遇到問題時有更明智的解決方法。此外，現今許多數位遊戲皆已打破遊戲中既有的規則、慣例、目標，其不僅允許玩家依照自己的想法及遊戲社群的討論結果來設計屬於自己的遊戲。學習者在設計自己的遊戲內容和虛擬系統時，將激發出創造能力及批判性思考能力，這也就是所謂的創新發揮的學習[4]。

**2.1.3遊戲式學習的教材設計與應用**

本專題以提升學習程式設計之初學者的學習意願並降低初學者之學習焦慮為研究方向，經調查與研究發現，以遊玩遊戲的方式來學習程式設計，對大部分的修習C++程式設計的大ㄧ學生來說，能有效降低對於學習程式設計的學習焦慮，並且在低成就組(期中考成績在全班四分之ㄧ以下)效果更為顯著[5]，因此本專題決定設計一款以輔助教學為目的之程式設計遊戲，在遊戲中主要的編譯環境為C語言，學習者在遊戲中必須以載具的型式來闖關，並且搭配C語言與本系統的指令來控制載具抵達終點，遊戲的內容進度將搭配大ㄧ學生所修習的計算機導論課程，依據課程章節內容依序設計出相對應的遊戲關卡，遊戲關卡總共50關，所輔助的課程內容依序為：基礎邏輯、變數宣告、基本輸入輸出、條件式判斷、流程控制、副函式、陣列、指標以及字串等，此外根據研究指出[2]，學習者在設計屬於自己的遊戲內容或虛擬系統時，能激發其創造力以及批判性思考的能力，對此本專題也設計了能讓玩家創造屬於自己關卡地圖的系統，在這個系統當中，玩家可以透過自己在關卡闖關過程中所獲得的元素來設計屬於自己的關卡地圖，且這些獲得的元素亦與本遊戲的主軸­­─程式設計有密切關係，因此玩家在體驗這個系統時，同時能激發其創造能力並提升C語言的程式設計能力。

**2.2程式積木**

一談到程式領域的遊戲式學習絕大部份非程式積木莫屬，而程式積木是指將複雜的程式碼都變成積木的形式，此舉不僅可以避免在寫程式時因打錯字而造成程式無法正常編譯，而將程式碼以積木的方式表現出來，更可以讓程式撰寫變得較為簡單，也較容易閱讀，從而減輕程式學習者在學習時因過多的程式碼需要記憶或較不直觀的文字表達而造成學習上的困擾[6] 。根據Piaget於1960年代提出，程式積木結構具有以下三種特徵：

1. 整體性：結構中的各個部分，其實是以一定的規則進行組合而形成一個整體。
2. 轉換性：只要依照一定的規則，此整體不會因為順序的更動而改變結構本身。
3. 自律性：組成結構的各個部分可劃分為一個個的整體，不受外在因素影響。

　　在程式積木的發展中，Blockly 是一款運行在網頁客戶端的 JavaScript 程式庫，目的是用來打造視覺化程式設計語言，它是一項由Google根據 Apache2.0 許可的開放源碼專案。常見被應用在網頁瀏覽器中，視覺上與 Scratch 差不多。其發展始於 2011 年夏天，首次發佈是在 2012 年的 Maker Faire 公開發佈。Blockly最初是為了替代 App Inventor 的 OpenBlocks 所設計的。而後來也被運用在於遊戲上，其遊戲名稱為Blockly Game，如（圖2），他擁有多種的遊戲型態，我們選擇其中的「迷宮」來進行說明；如（圖3）所示，其遊戲分為「遊戲區塊」與「程式區塊」，其遊戲區塊是用來顯示程式區塊中程式所執行出來的結果，讓玩家可以了解程式如何實際的運作。在上文所提及的Scratch是由麻省理工學院媒體實驗室（MIT Media Lab）所開發的一套圖像化程式設計工具，可以用來創造互動式故事、遊戲、多媒體動畫與美術。

　　根據Summer Games: Learn to Program [7]所說：「尋找方法讓孩子參與有建設性且有意義的學習？」，藉由這個目的，他們開發了Blockly這個應用，該文章也指出：「通過最大限度地減少語法的使用，用戶可以專注於計算機科學家使用的邏輯和概念，在他們冒險通過迷宮和更高級的競技​​場時，按照自己的進度前進。」。沿用上述的想法，我們在本專題內也用Blockly實作出了程式積木的環境供玩家來使用，希望玩家可以透過積木的方式減少因為打程式碼而造成的拼字錯誤而造成遊玩上的困難[8]，另外因為Blockly積木的形式能完整的將函式、迴圈與判斷式的範圍框定出來，故可以令玩家更清楚的了解該程式段所作用的範圍。

　　而Google所開發出的Blockly積木雖然原本就具有將積木轉成 「JavaScript、Python以及PHP」等程式碼之功能，但Google所提供的積木內容並不符合我們的需求，因此Google也提供了Blockly的開發人員工具（圖4）讓我們可以創建屬於自己的積木，雖然它能將積木轉譯成多種程式語言，但是卻沒有我們所需要的「C語言」，所以本團隊另外開發出將我們自製的積木轉譯成C語言的工具。

圖2 Blockly Game首頁



圖3迷宮遊戲頁面



圖4 Blockly開發人員工具

**3.系統功能需求**

本專題《全速衝線》藉由線上編譯器以及Blockly積木將學習程式與網頁遊戲結合，讓使用者能夠透過邊玩遊戲邊學習的方式來循序漸進的學習程式C語言，有效的幫助使用者提起學習程式的熱情以及降低學習程式的門檻，而且在程式碼編譯失敗後以及程式執行的結果不如預期的情況下，將會告知使用者失敗原因，提升學習成效。此遊戲除了系統設置的主關卡外，還能讓使用者以自己的想法創造地圖，發揮每個人的想像力，並且上架至伺服器上供其他玩家遊玩、評價，並且可以修改地圖及刪除地圖。且系統會對每一位玩家的遊戲資訊進行記錄，包含帳號、使用者名稱、信箱、詳細資訊……等，系統管理員可根據上述對玩家記錄的遊戲資訊判斷是否使用非法程式或者是修改遊戲資料做出作弊行為，而對玩家進行封鎖及解除封鎖的動作，以及將統計各關卡通關人數及通關率，分析玩家的通關情況，並做出修正。基於以上之功能描述，本專題《全速衝線》的使用案例圖與功能架構圖如圖5與圖6所示，各主要項目之功能說明則簡述於表1。



圖5《全速衝線》使用案例圖



圖6《全速衝線》功能架構圖

表1.《全速衝線》功能項目說明

|  |  |  |
| --- | --- | --- |
| 功能項目 | 功能項目操作 | 說明 |
| 會員管理 | 註冊會員 | 訪客可於註冊頁面填寫個人資料（帳號、名稱、信箱、密碼）來註冊成為會員。 |
| 修改密碼 | 會員可將舊有的密碼修改為新密碼抑或是找回密碼。 |
| 查看成就 | 會員可點擊遊戲畫面中成就按鈕來查看獲得的成就。 |
| 關卡遊玩 | 檢視關卡 | 玩家可以透過點擊關卡按鈕檢視關卡內容或遊玩關卡。 |
| 解鎖關卡 | 玩家須通過指定關卡才能開啟後續關卡。 |
| 查看紀錄 | 玩家可查看遊玩後的關卡紀錄，包括程式碼、指令積木、最短程式碼。 |
| 設定環境 | 玩家可設定遊戲環境（遊戲音樂開關、音量大小、遊戲動畫速度）。 |
| 程式編輯 | 撰寫指令 | 玩家可用關卡提供的指令來撰寫程式碼。 |
| 組合積木 | 玩家可用關卡提供的Blockly積木來編輯積木組合。 |
| 提交程式 | 無論玩家是自行撰寫或透過積木來完成程式的組合，皆可透過提交程式來完成遊戲。 |
| 重置地圖 | 玩家可透過關卡內重置地圖的功能來達成一鍵將地圖區的元素初始化。 |
| 重置關卡 | 玩家可透過關卡內重置關卡的功能來達成一鍵將程式區的內容及地圖區的元素全部初始化。 |
| 轉譯積木 | 玩家可透過轉譯積木，將當前積木的組合轉換成C語言來對照當前積木的組合。 |
| 地圖設置 | 創建地圖 | 玩家可進入失落帝國的創建及修改地圖頁面，以各種物件及設定來創建自己的地圖。 |
| 修改地圖 | 玩家可透過修改地圖按鈕進入創建及修改地圖介面來修改已創建的地圖。 |
| 上架地圖 | 玩家可透過上架地圖按鈕將已創建並已通過檢測的地圖上架供其他使用者遊玩。 |
| 下架地圖 | 玩家可透過下架地圖按鈕將已上架的地圖下架，使其可進行修改或刪除。 |
| 刪除地圖 | 玩家可透過刪除地圖按鈕將已創建好的未檢測、已檢測、已下架地圖刪除掉。 |
| 後台管理 | 管理會員 | 管理員可以查看所有玩家之帳號、使用者名稱、信箱、星星數、可遊玩的最高關卡、使用者帳號狀態，並可封鎖或解除封鎖會員。 |
| 統計資料 | 管理員可以查看已遊玩人數、玩家的通關率及平均失敗次數等資訊來做出的統整圖表。 |
| 管理關卡 | 管理員可以管理員身分進入遊戲並修改關卡設定。 |

**4.系統資料需求**

**5.系統架構**

　　《全速衝線》平台主要採用使客戶／伺服式架構，客戶端與伺服端之間透過網際網路進行連接，客戶只要在有電腦與網際網路的地方就能進行遊戲，客戶端與伺服端的簡述如下：

* 客戶端：

－採用網頁瀏覽器作為操作環境，使用者可以在瀏覽器上進行遊玩關卡、

　檢視紀錄、創作地圖以及遊玩其他使用者所創造之地圖。

* 伺服端：

本系統的資料核心，將Windows作為伺服器的作業系統，並提供以下功能：

－管理端：

　　　管理者可以透過平台所提供之管理介面對使用者進行帳號的封鎖／

　解除封鎖，或通過統計資訊觀察使用者的遊玩情形，並進行評估。

－資料庫：

　　　使用MongoDB，存放使用者基本資訊、通關資料以及使用者所創

立地圖資訊。



圖7 系統架構

**6.實作成果**

如前所述，本專題所發展之《全速衝線》網頁遊戲是以遊戲式學習程式設計為主要發想，在遊戲內教使用者程式C語言的語法以及各種實際的應用，且開發出一套程式積木（如圖8）供使用者遊玩，積木的顏色代表了分類，本團隊設置了原始的程式指令積木以及遊戲遊玩的指令積木共8種族群。

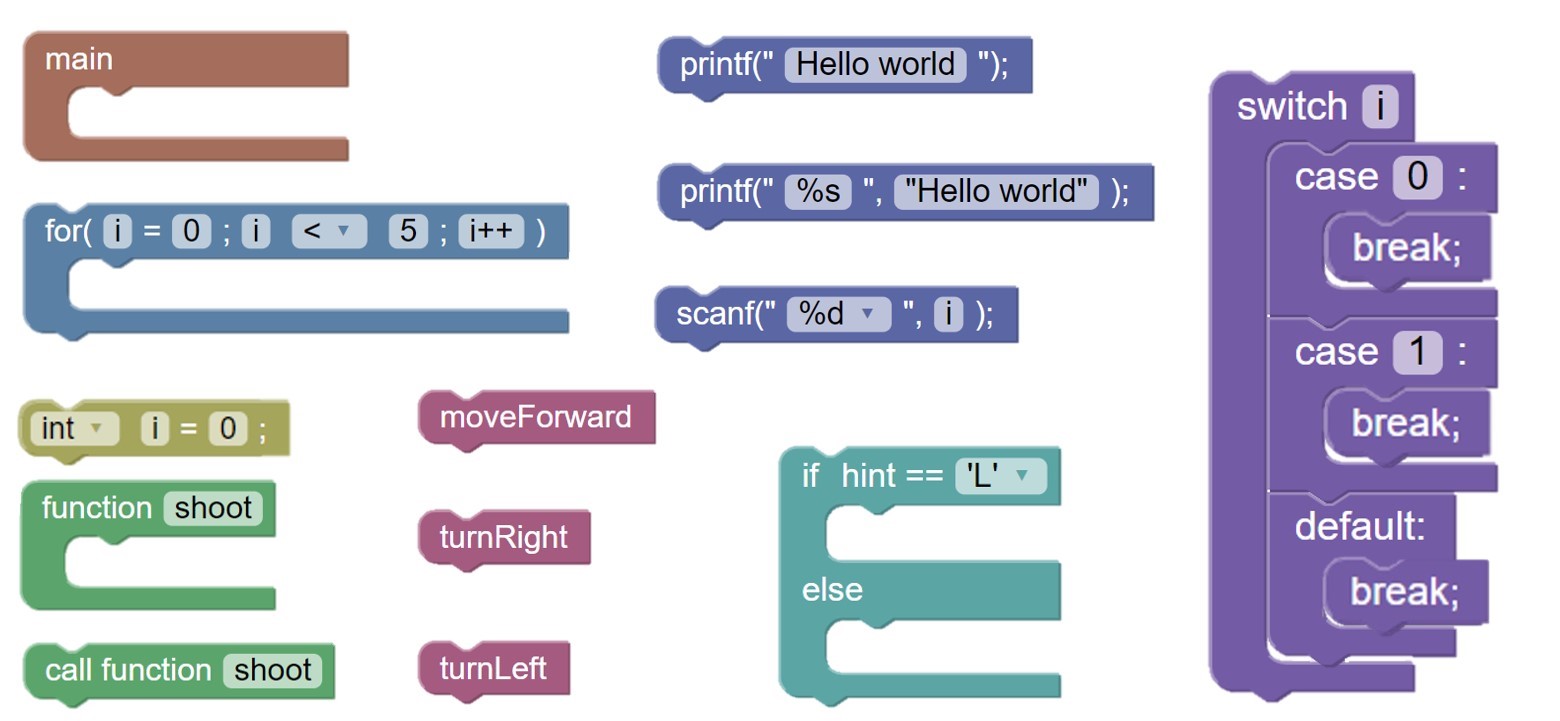


圖8程式積木

玩家一開始進入《全速衝線》網頁遊戲時，最先出現的是登入頁面，使用者將從此畫面進行註冊帳號、登入遊戲。接著進入到遊戲首頁頁面（如圖9），此畫面分成了三個部份分別為：畫面中間的普魯斯帝國、庫魯瑪帝國、失落帝國、畫面右下角的功能區（裝備、指令大全、成就、設定）以及右上方的個人資料區（使用者名稱、獲得星星數、個人資料、登出），可在此畫面選擇帝國進行遊玩或是點選右下角功能區進行裝備升級、查看指令大全、查看成就、調整設定。選擇帝國後，將進入該帝國地圖接著點擊任一可遊玩關卡進入開啟關卡確認畫面（如圖10），接著點擊進入關卡，進入遊玩關卡頁面（如圖11），此畫面將呈現此專題的重點，以左方程式區及右方地圖區的搭配來編輯程式碼操縱遊戲主角通過各式各樣的地圖抵達終點（如圖12），並且除了打程式碼的方式外還能夠切換成Blockly積木模式進行遊戲（如圖13），當以Blockly積木模式進行遊戲時可點擊轉譯積木按鈕，開啟轉譯視窗將目前組裝的程式積木轉換為C語言並顯示於視窗中（如圖14）。第一帝國-普魯斯帝國及第二帝國-庫魯瑪帝國為主要教學關卡，總共50關，將由基本邏輯教起，依序教學基本邏輯→基本輸入輸出→判斷式→迴圈→自訂函式→陣列→指標→字串，並且在各帝國的最後幾個關卡皆設置魔王關，讓使用者需要融會貫通已教學的程式碼才能通關。



圖9遊戲首頁頁面



圖10關卡確認頁面



圖11遊玩關卡頁面



圖12遊戲執行畫面



圖13Blockly積木模式遊玩關卡頁面



圖14轉譯積木視窗

除了主要教學關卡，還設置了第三帝國-失落帝國，其內容與其他兩個帝國不同，遊玩方式為玩家創造地圖上傳至伺服器端，讓其他玩家能夠遊玩該地圖，其目的是為了讓玩家經由發揮想像力創造地圖的過程中，更加熟悉對程式碼的應用，以及增加此遊戲的可玩性及多變性。點擊失落帝國進入玩家自訂地圖頁面，此頁面將顯示其他玩家所創建的地圖以及該地圖的名稱、遊玩條件、製作者、評價、上架日期、地圖簡介，可在此頁面點擊想遊玩的地圖進行遊玩，也可進入我的自訂地圖頁面（如圖15）管理自己的地圖，對地圖做上/下架、檢測、刪除的動作，接著點擊左下創建地圖按鈕進入創建及修改地圖頁面（如圖16）對地圖進行各種設定，包括新增物件、刪除物件、調整地圖大小、設定地形…等，由玩家發揮想像力創造各式各樣的關卡。



圖15我的自訂地圖頁面



圖16創建及修改地圖頁面

關於後臺管理的部分，管理員能夠經由登入頁面輸入管理員帳號密碼以管理員身分進入至後臺管理首頁，可在此頁面選擇管理會員、進入遊戲、查看統計資料，點擊管理會員將進入管理會員頁面（如圖17），此頁面將顯示所有玩家的個人資料（帳號、名稱、信箱、星星數、已達到的最高關卡、使用者狀態），管理員可透過此頁面查看是否有玩家舞弊修改遊戲資料，並且可選擇玩家將其封鎖或解除封鎖。管理員於管理員首頁點擊進入遊戲將以管理員身分進入至遊戲中，並有著管理員的自動輸入指令功能方便檢測各關卡。管理員於管理員首頁點擊統計資料即可進入統計資料頁面，該頁面將顯示1-50關關卡的通關情況包括已遊玩人數、通關率、平均失敗次數，並以折線圖（如圖18）及長條圖（如圖19）表示，管理員即可藉由這些資料分析關卡難易度是否恰當及使用者通關情形。



圖17管理會員頁面

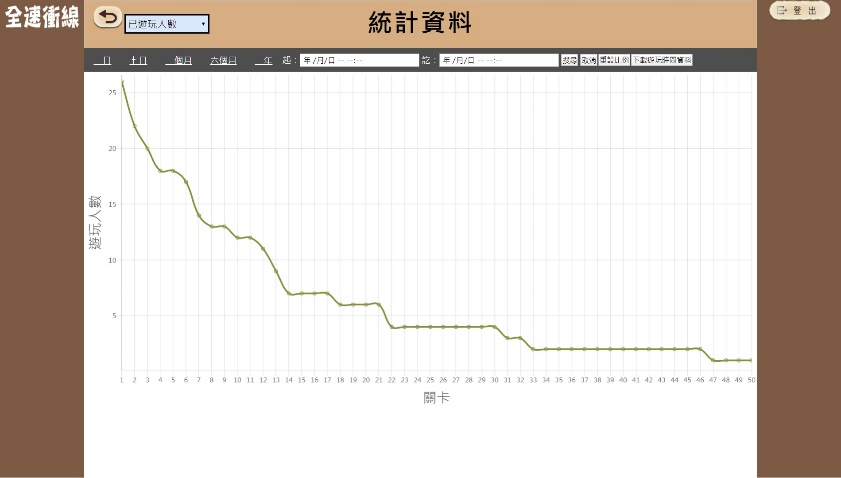


圖18統計資料-已遊玩人數

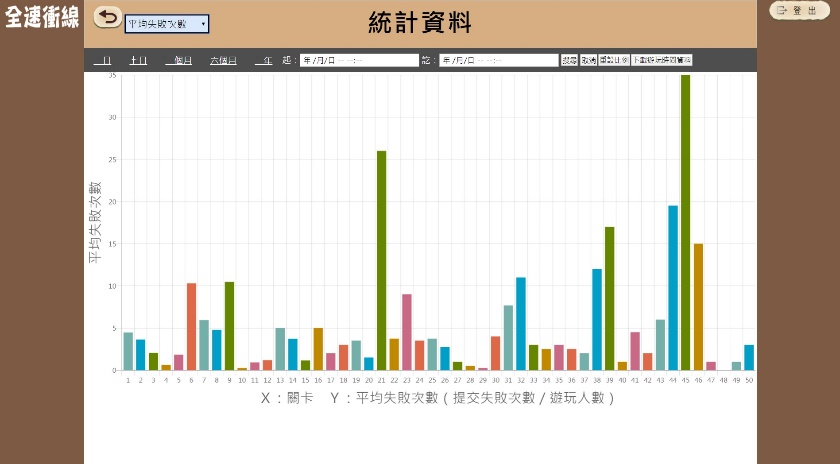


圖19統計資料-平均失敗次數

**7.結論**

本專題中，我們研製了一款名為《全速衝線》的平臺，藉以實現降低使用者學習程式時的焦慮感與讓使用者可以在學習的過程中得到適當的回饋的目的。

　　基於遊戲式學習的原理與Blockly程式積木技術，此一平臺具有多項特色，包括了用圖形化的程式積木以減緩使用者的枯燥感；以及將程式的執行結果，用遊戲的動畫來呈現讓使用者提升學習的樂趣；以及藉著平台所設計的關卡讓使用者可以依照自己的學習進程或相關課程的教學進度來遊玩遊戲，都可以達到自主學習的目的，並且與課堂進度所進行遊戲能達到雙倍學習的成效與基礎，使其提高學習程式的熱情。

此一平臺經驗證後，已能達到改善使用者學習程式時因無法得到實際的回饋而對學習失去興趣的目的，並將本平臺的程式積木部分與市面上類似產品比較（如表2）可以發現我們的更加優異。然而，在教學程式語言的種類方面，此一平臺仍有需要加強之處，因此，未來我們將新增比對功能，將使用者所撰寫的程式與正確解答進行比對，使得平台能夠指出使用者所撰寫的程式有哪裡可以進行改善以及新增此平臺所能對應的程式，例如：python、Java...等，讓使用者可以學習多種程式語言。

表2.類似產品比較表

|  |  |  |  |
| --- | --- | --- | --- |
|  | 全速衝線 | Scratch | Blockly Game |
| 遊玩方式 | 程式積木 | 程式積木 | 程式積木 |
| 可轉換語言 | Ｃ語言 | 無 | JavaScript、Python以及PHP |
| 目的 | 輔助教學，可隨時將積木轉換為Ｃ語言，能讓使用者比較積木與Ｃ語言的對應關係。 | 使用積木來創造一個動畫或小遊戲。 | 透過各個遊戲來傳達各種程式觀念。 |
| 優點 | * 可以隨時將積木轉為C語言對照。 * 能看出完整的程式範圍。 | * 較接近物件導向的觀念，每段程式較像一個「物件」。 | * 依照遊戲形式的不同教導不同的觀念。 |
| 缺點 | * 為了能與C語言相對應，積木樣式較不多樣。 | * 偏向實作應用，沒有自己的教程來教導觀念。 | * 遊戲中不能隨時將積木轉換成程式語言來對照。 |

**7.相關文獻**

**[1]**[**Google Introduces Kids to Coding Through Blockly Games Project**](https://www.eweek.com/it-management/google-introduces-kids-to-coding-through-blockly-games-project)By: [Todd R. Weiss](https://www.eweek.com/Authors/todd-r.-weiss) | August 17, 2014

[2] Prensky, M. (2007). Digital game-based learning. St. Paul, MN: Paragon House

[3] Selnow, G. W., & Reynolds, H. (1984). Some opportunity costs of television viewing. Journal of Broadcasting & Electronic Media, 28, 315-322.

[4] 簡晨卉，數位遊戲式學習在城鄉國小數學加減法學習成效之研究，國立台中教育大學數位內容科技學系碩士班，碩士論文，民國102年。

[5] 何昱穎、張智凱、劉寶鈞（2010）。程式設計課程之學習焦慮降低與學習動機維持–以Scratch為補救教學工具。數位學習科技期刊，2（1），11-32。

[6] 陳冠岑，以Scratch學習程式設計及其與學習者認知風格的關聯性，國立交通大學理學院科技與數位學習課程。碩士論文，民國101年6月。

**[7]** [Summer Games: Learn to Program](http://ai.googleblog.com/2014/08/summer-games-learn-to-program.html) Monday, August 11, 2014

[8] 積木式程式設計 陳婉凌 著