逢 甲 大 學

資訊工程學系

|  |
| --- |
| 碩 士 論 文 （初稿） |
| **Git Education Game - 程式碼版本控制學習遊戲之研發**  **Git Education Game - Game development for learning code version control technology** |
|  |
| 指導教授：陳錫民    研 究 生：張佑瑋 |
|  |

中華民國一百一十一年六月

誌謝

摘要

　　版本控制系統對於軟體行業是不可或缺的工具，身為工程師必然需要具備使用版控工具的能力，然而在大多數大學教育中對於此項技術的教育著墨不多，傳統的教學方式也使得學生對於Git容易有概念上的混淆以及使用上的障礙。本研究提出了一個名為GEG的嚴肅遊戲用於教授Git的概念與使用方法，目的是改善學生的學習動機，並實現相對傳統授課更深入的學習。本研究設計了一個實驗，在實驗中同一門課程的兩個班級被分為實驗組與控制組，為了測量遊戲對學生的影響，設計了一個測驗驗證遊戲是否能夠幫助學生獲得更好的學習成果，並設計了一個問卷調查遊戲是否對學生的動機有積極的影響。結果顯示，遊戲在學生的動機方面有積極的影響，在學習成果方面，儘管在最困難的課題中通過率並不高，但實驗組在每一項目中仍舊比控制組擁有更高的通過率。

**關鍵詞:** **遊戲式學習、版本控制、Git、軟體工程、教育遊戲、嚴肅遊戲**

# Abstract

# 目錄

[誌謝 i](#_Toc101791941)

[摘要 ii](#_Toc101791942)

[Abstract iii](#_Toc101791943)

[目錄 iv](#_Toc101791944)

[圖目錄 vii](#_Toc101791945)

[表目錄 viii](#_Toc101791946)

[第一章 緒論 1](#_Toc101791947)

[1.1 研究背景 1](#_Toc101791948)

[1.2 研究目的 1](#_Toc101791949)

[1.3 研究問題 2](#_Toc101791950)

[1.4 研究原創性 2](#_Toc101791951)

[1.5 重要性 3](#_Toc101791952)

[1.6 論文結構 4](#_Toc101791953)

[第二章 文獻回顧 5](#_Toc101791954)

[2.1 嚴肅遊戲 5](#_Toc101791955)

[2.2 遊戲元素 5](#_Toc101791956)

[2.3 軟體工程中的教育遊戲 6](#_Toc101791957)

[2.4 業界期望與計算機教育 6](#_Toc101791958)

[2.5 延伸整合科技接受模型 7](#_Toc101791959)

[第三章 系統設計與方法 8](#_Toc101791960)

[3.1 遊戲設計原則 8](#_Toc101791961)

[3.1.1 布魯姆分類原則 9](#_Toc101791962)

[3.2 遊戲總體設計 10](#_Toc101791963)

[3.2.1 系統架構 10](#_Toc101791964)

[3.2.2 遊戲流程 11](#_Toc101791965)

[3.2.3 教學CLI 15](#_Toc101791966)

[3.3 遊戲機制 15](#_Toc101791967)

[3.3.1 點數 15](#_Toc101791968)

[3.3.2 排行榜 15](#_Toc101791969)

[3.3.3 成就與獎章 17](#_Toc101791970)

[3.4 關卡主題與數據監控 19](#_Toc101791971)

[第四章 系統展示 21](#_Toc101791972)

[4.1 遊戲關卡 21](#_Toc101791973)

[4.2 遊戲系統 21](#_Toc101791974)

[第五章 系統實驗 22](#_Toc101791975)

[5.1 實驗環境 22](#_Toc101791976)

[5.2 實驗流程 22](#_Toc101791977)

[5.3 研究分析方法 22](#_Toc101791978)

[5.4 實驗數據 22](#_Toc101791979)

[5.5 實驗前置測驗結果與分析 22](#_Toc101791980)

[5.6 實驗後置測驗結果與分析 23](#_Toc101791981)

[5.7 問卷調查結果與分析 23](#_Toc101791982)

[5.8 **模型可靠性和有效性測試** 23](#_Toc101791983)

[**5.9** **結構模型之結果分析** 23](#_Toc101791984)

[**5.10** **研究問題之結果分析** 23](#_Toc101791985)

[5.10.1 研究問題1 23](#_Toc101791986)

[5.10.2 研究問題2 24](#_Toc101791987)

[5.10.3 研究問題3 24](#_Toc101791988)

[5.10.4 研究問題4 24](#_Toc101791989)

[第六章 結論與未來研究 25](#_Toc101791990)

[參考文獻 26](#_Toc101791991)

# 圖目錄

[**圖 3.1 系統架構**](#_Toc70327610) **10**

[**圖 3.2 關卡選單**](#_Toc70327611) **11**

[**圖 3.3 開始選單**](#_Toc70327611) **12**

[**圖 3.4 遊戲的六大區塊**](#_Toc70327610) **13**

[**圖 3.5 遊戲主畫面**](#_Toc70327611) **14**

[**圖 3.6 關卡通過畫面**](#_Toc70327610) **14**

[**圖 3.7 關卡四的排行榜**](#_Toc70327611) **16**

[**圖 3.8 總排行榜**](#_Toc70327610) **17**

[**圖 3.9 獲得成就**](#_Toc70327611) **18**

[**圖 3.10 成就閱覽畫面**](#_Toc70327610) **19**

[**圖 3.11 關卡八學生通關資料**](#_Toc70327611) **20**

# 表目錄

[**表 1.1相似系統比較**](#_Toc70327610) **3**

[**表 2.1 傳統遊戲元素的術語**](#_Toc70327611) **5**

[**表 5.1 實驗組與控制組的前置測驗結果**](#_Toc70327611) **23**

# 第一章 緒論

## 研究背景

版本控制工具的能力是軟體工程師必不可缺的技能[1]，然而這項必備的技能並不一定被當作計算機科學課程的一部分，有教學的課程中也通常使用很短的時間來教學這向技術，這使得學生對於版本控制工具的概念及使用方式等等混淆不清，傳統的教學方式也進一步導致這個現象更為嚴重，學生難以具有學習的動力、也難以理解教學內容。

在非傳統教育方式當中，遊戲化被認為是一種有潛力的教學方式，提高使用者的積極性被認為是遊戲化的一個關鍵特徵[2]，基於遊戲的學習做為一種教學方式可以強化學生的內在動機[3]，增加主動學習的意願，虛擬的環境中可以模擬各種課堂上難以立即重現的情境，並透過互動的方式使教學抽象概念更加容易，遊戲可以提供學生不停嘗試的機會，不必擔憂操作失敗可能帶來的風險，即時回饋也增進了學習的效率，因此遊戲化能夠有效改善學生的學習動機、學習效率，由於遊戲的內在特性，例如競爭、挑戰、互動，他們能將學習過程轉變為有趣的體驗，並可接受的教學時間和教師負擔範圍內實現深度學習[3]。

## 研究目的

基於前段所講述之議題，本研究提出系統Git Education Game（以下簡稱GEG），GEG是一款基於Web的嚴肅遊戲，它是由Unity結合JSP開發而成的，用於教學Git的概念與指令，目的是為了有效改善學生對於Git的學習動機、學習效率，並補足學校課程中所不足的部份。此遊戲將Git的概念與指令分為數道關卡，並引入遊戲元素如：點數、獎章、排行榜等等用以激勵學生參與學習，從學習中獲得成就感，鼓勵學生在模擬的環境中不斷嘗試，而學生的學習行為則經由API發送至後台的資料庫當中，教師可以即時監控學生的學習狀況。

## 研究問題

為了評估系統的有效性，本研究設計了一個教育研究實驗，在實驗中將同一門課程的兩個班級分為實驗組及控制組，分別以基於遊戲的方式授課以及以傳統的方式授課，並評估學習效果，同時我們提出以下研究問題:

* Research Question 1 (RQ1): 加入GEG作為教學輔助工具是否比傳統的授課方式具有更高的學習成果？
* Research Question 2 (RQ2): GEG作為教學輔助工具是否為學生對Git的態度及行為帶來正面影響？
* Research Question 3 (RQ3): 會主動以GEG進行學習的學生比例是多少？主動進行學習的學生是否具有較高的學習成果？
* Research Question 4 (RQ4): 學生認為GEG有何優點與缺失？

為了回答這些研究問題，本研究應用並擴展了部分UTAUT2[4]和PLS-SEM[5]，考察使用遊戲化學習方式是否正面影響學生對Git的態度，並考察遊戲化對自我效能、表現預期及享樂主意動機等等因素的影響，本研究引入了績效預期、努力期望、享樂主義動機[6]、自我效能感[7]、態度[7]、行為意向、實際行為，並加入了遊戲化有用性、遊戲化動機檢測我們所設計的機制帶來的影響。

## 研究原創性

專門用於教學程式碼控制技術的遊戲並不多，更多以文章的形式存在，而既有的系統則不見得具有足夠的娛樂性能激起學生的學習動機，它們大多為互動式教學軟體，缺乏遊戲元素。除此之外，也有系統模擬的指令過於簡化，在實際使用時無法作為參考，僅能學習工作流程或概念，表1.1列出了相似系統的比較。

表1.1 相似系統比較

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Learning Git Branching** | **Oh My Git!** | **Git Education Game** |
| 採用模擬Git指令方式 | O | X | O |
| Git指令關卡指導 | O | O | O |
| 完整指令用法學習過程 | X | O | O |
| 額外的指令講解機制（指令卡片說明） | X | O | O |
| 可互動的提示系統 | O | O | X |
| 遊戲元素或機制 | X | X | O |
| 紀錄使用者活動供評估學習狀況 | X | X | O |
| 網頁上可直接使用 | O | X | O |

我們所提出的系統Git Education Game作為輔助學習工具相較其他模擬Git指令的教具在實際使用上具更高的可參考性，而相對讓玩家直接使用Git指令的系統執行更加快速，並對某些具有較複雜參數的指令可以限縮在學習曲線較可令學生吸收的範圍內。

因為引入了遊戲元素，我們提出的系統遊戲機制更為豐富，能更佳促進學生的學習動機，這些機制也作為教具的一部分給予學習時的幫助，改善學生的學習效率。老師方面則可以藉由後台的數據觀察學生的學習狀況，即時加強教學不足的部份，促使學習的成果更高。

## 重要性

如研究背景所述，無論是資訊工程的學生還是業界人士，版本控制工具的使用能力是必備的，在軟體業界工作必然需要使用版本控制軟體，除此之外，參與開源專案、開發開源專案、與開源社群互動，最基本的門檻也都是具備使用Git版本控制工具的能力，甚至現在熱門的DevOps技術所需要的持續整合、持續部署技術也同樣必須透過Git來觸發。

而透過設計遊戲來教學Git的優勢有以下兩點：

1. 電腦可以扮演成員與學生互動，給予問題環境，模擬現實中無法短時間立刻重現的情境
2. 真實的工作環境會害怕犯錯，遊戲可以避免這種情況盡可能鼓勵學生嘗試

這些優勢是採用其他教學方式難以實現的優勢，因此本研究論文所提出之系統具有其必要性。並且此系統可以幫助老師即時了解學生的學習狀況，針對學生無法即時理解的單元做加強，而引入遊戲元素的方式也能大大增加學生學習的參與度，刺激學生學習，以實現相對傳統教育更深入的學習。

## 論文結構

　　本文的第二章對相關的論文做回顧；第三章詳細介紹本論文提出的系統設計與方法；第四章為系統展示；第五章為系統實驗；第六章為結論與未來研究。

# 第二章 文獻回顧

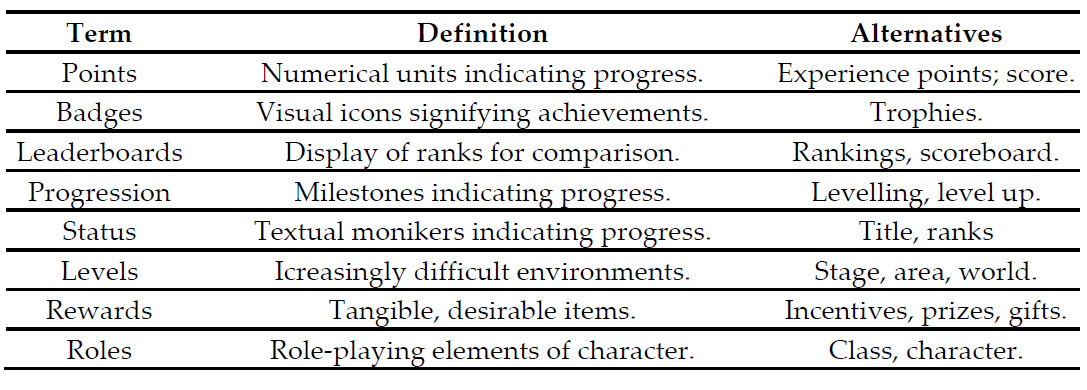
## ****嚴肅遊戲****

嚴肅遊戲是應用於嚴肅目的的電腦遊戲，近幾十年來嚴肅遊戲與遊戲化都被用於開發用於嚴肅目的，兩者的定義不同，嚴肅遊戲以完整的遊戲為基礎，將娛樂作為次要，以教育為中心[8][9]，遊戲化則是將遊戲元素加入到非遊戲的環境之中[10]，但都試圖使用遊戲或遊戲元素來教育和改變行為模式[11]，在教育、健康等環境中教育、鼓勵、說服使用者[12][13]。本研究的主要目的，即是通過嚴肅遊戲，來教育、改變學生的版本控制軟體的認知，藉由引進遊戲化元素，建立遊戲機制等等來使嚴肅遊戲能夠獲得不同於普通遊戲的有益成果[14]。

## ****遊戲元素****

遊戲元素為吸引使用者的重要動力，在[15]中提到了數個遊戲化元素，並且遊戲元素往往是相互關聯的，例如”級別”和”等級”可以指通過經驗獲得的等級，並統整了遊戲化元素，如表2.1所示。

表2.1 傳統遊戲元素的術語



本研究引入其中的點數、獎章、排行榜等元素作為系統主要的遊戲機制。

## ****軟體工程中的教育遊戲****

　　傳統授課只允許被動學習，這導致教育軟體工程的過程難以提供足夠實用的知識，在[16]中，開發了一款模擬軟體工程過程的教育紙牌遊戲，該研究描述了他們如何設計遊戲機制以令學生充分參與、了解軟體工程的流程與可能遭遇的問題，結果顯示該遊戲在引領學生入門軟體工程流程方面取得了一定的成功，學生一致認為將該由納入軟體工程課程中有助於他們理解軟體工程概念。[3]則描述了他們如何運用遊戲的內在特性，將教授Scrum開發方法，將學生的學習過程轉變為有趣的體驗，並提到遊戲可以在可接受的教學時間和教師負擔範圍內實現深度學習，結果顯示由於遊戲的競爭性，激發了學生在學習的參與度，對學生的學習體驗及學習動機造成了積極的影響。因此本研究也選擇加入競爭元素，如排行榜機制來刺激學生產生學習動機。

## ****業界期望與計算機教育****

在畢業生進入勞動力市場的能力研究中，團隊技能、協作與這些技能的工具被提到需要改進，[17]中提到了行業期望與畢業生能力的差異，文獻中提到的能力與團隊如何管理軟體有相當的關係。[18]提到了Git是一個成熟的、廣受好評的程式碼版本控制系統，並提到對業界標準工具經驗缺乏的學生來說，被充分限制了參與實習的能力，並講述了他如何引入Git作為一種機制，用於發布課程練習，[19]中介紹了他們在計算機科學中使用Git作為他們課程平台的經驗，並認為從課程的角度來看因此本研究提出以Git為主題的遊戲化教育實驗，用以增進學生對Git的掌握與學習動機、態度。

## ****延伸整合科技接受模型****

UTAUT模型由六個主結構組成，UTAUT模型整合了八種主要的技術接受理論[20]，[21]研究了五所大學的學生對數位學習的接受度，UTAUT2模型則整合了技術接受模型（TAM）[22]，結果發現計算機自我效能、計算機遊戲性對電子學習系統的感知易用性有顯著影響， [23]研究了遊戲化對對行動銀行服務的影響，研究結果顯示遊戲化與使用手機銀行服務的意向之間存在直接和強烈的關係。因此對本研究而言，它是最適合我們發展與驗證理論的模型。

# 第三章 系統設計與方法

本章節中將會介紹3.1遊戲設計原則、3.2遊戲總體設計、3.3遊戲機制以及3.4關卡主題與學生數據監控。

## ****遊戲設計原則****

　　本研究採用了[32]所提出的設計原則，將flow experience與教育設計結合，試圖創造出具有高度沈浸感的遊戲體驗，進而使學習過程的效益更高，因此本研究提出之系統以以下步驟進行設計:

1. 分析學習者: 我們的學習者以computer science科系的大二學生為主，具有基本的編程能力，但大多還不具備有版本控制工具的使用能力，也大多沒有長期維護專案或大型專案的開發經驗，因此需要讓他們能夠具備基本的版本控制知識與版本控制工具使用能力。

2. 設立明確的教育目標，選擇適合的遊戲內容: 本研究的教育目標設立為令學生具備Git指令的使用能力，這些指令包含建立git repository, commit, push等等，並且能夠理解使用版本控制工具的原因與Git的工作流程。因此本研究選擇採用模擬開發情境的方式，讓學生在遊戲中操作模擬的Git工具，完成我們所設立的任務目標。

3. 根據教學目標和遊戲內容設計教學方法: 本研究將Git的指令設計成數道關卡，要求學生在模擬的情境中完成我們所設立的任務目標，學生在學習過程可以獲得遊戲內的機制作為學習輔助，並且由於我們所設計的競爭性的遊戲機制與遊戲獎勵，可以激勵學生在學習過程中的參與程度，增加學生的學習動機。

4. 將教學視為主要目標，遊戲作為輔助工具: 本研究將GEG作為課程的一部分，在後續的課堂中仍以傳統的講座方式教授Git的概念與操作方式，對學習遇到困難的學生可以以遊戲進行自主學習，選擇適合他們的學習方式。

5. 良好運用電腦遊戲的特徵: 我們的遊戲屬於模擬遊戲，同時加入了一些競爭遊戲的特徵，藉由電腦遊戲的特性，帶來重複嘗試的機會與動力，遊戲的提示說明、指令卡牌作為輔助學生理解的工具。

6. 將學生置於教學過程的中心，幫助他們享受學習: 學生在學習過程中基於遊戲的積分機制、成就機制等等，主動嘗試各種可能，增加主動參與學習的動機，使學生能夠學習到更多關於Git的知識與使用方法。

7. 定期評估學生的學習情況，不斷改進教學: 學生的activity被紀錄並發送至伺服器後台，從關卡通過數量、學生通關花費的指令數、學生通關花費的時間等等資訊可以判斷學生的學習狀況，並即時調整可能過於困難的關卡，或增加更多關卡來幫助學生釐清混淆不清的概念。除此之外，也進行測驗與問卷調查來了解學生的學習狀況。

1. **布魯姆分類原則**

在設計原則的第二步設計教育目標當中，本研究根據Bloom’s taxonomy[25]，將學生在知識、理解和應用上等等獲得的Git能力分為以下幾點:

1. Remember: 學生可以記得跟Git關聯的概念名詞與部份指令，比如:repository, commit, branch, merge等等。

2. Understand: 學生可以在操作、提示與視覺化的Git工作流程概念中理解local與remote repository的差別、Git與Git server的差別、分散式版本控制系統的作用、branch的作用與功能、merge的作用及conflict是什麼。

3. Apply: 學生可以在反覆練習中獲得操作各種指令或解決跟Git有關問題的能力，比如:clone, commit, push, pull, create branch, merge, conflict solve。

4. Analyze: 藉由循序漸進的關卡，學生必須在了解前面關卡的基礎上才得以通過後續的關卡，這能使學生了解每一個指令或Git概念的功能與區別，最終學會整體的版本控制工具及Git的工作流程與概念。

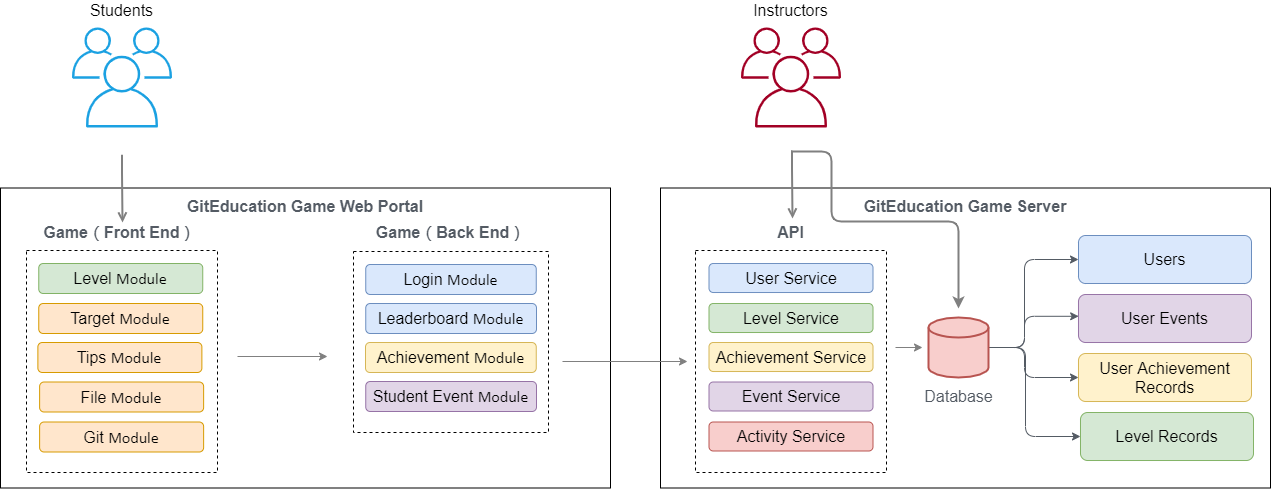
## ****遊戲總體設計****

GEG模擬了開發情境以及Git Command Line（Git CLI），玩家在遊戲中必須操作遊戲模擬的Git CLI，遊戲內分為十個關卡，除了第一關與第二關屬於遊戲的介紹章節以外，其餘每道關卡各代表了一個或數個Git的指令或概念，學生必須在每一個關卡中藉由遊戲內的提示來學習並嘗試操作，最終達成關卡內的任務目標並通過關卡。

遊戲內的關卡難度是循序漸進的，最初的關卡只需要完全跟隨提示輸入指令即可完成，後續的關卡則需要學會應用前面關卡的概念才能得到正確答案，當玩家通過關卡後便能獲得積分，達成某些特殊條件後也能獲得徽章與積分，這些結果將呈現在排行榜上，用以激勵學生。

### ****系統架構****

**遊戲的系統分為前景與後台，前景除了負責使用者與遊戲介面的互動，還會發送通知給遊戲的後台，後台會去與遊戲的伺服器進行請求，包含使用者註冊、登入服務，使用者的事件紀錄 以及相關的資料請求（學生的積分、成就狀態），系統的架構如圖3.1所示。**



**圖 3.1 系統架構**

**老師方面可以透過API獲取資料了解學生的學習狀況，包含活動紀錄、關卡通過紀錄等等，也可以直接從資料庫查詢來獲得更詳細的資訊。**

**後端伺服器採用了Spring Boot框架，使用RESTful API作為與遊戲系統溝通的橋樑，並負責處理系統的資料邏輯，比如：檢查帳號是否存在以拒絕重複註冊、對已存在關卡紀錄的資料予以更新而非建立一筆新資料，資料庫則採用MongoDB以紀錄各種學生的事件**

### ****遊戲流程****

本研究的遊戲是基於web的，學生必須進入遊戲的網址才可以開始進行遊戲。圖3.2顯示了遊戲的開始選單，我們要求學生必須以學號註冊才能開始進行遊戲，圖3.3則顯示了遊戲的章節選單，我們所要教授的Git概念與指令被包含在這些關卡當中，而學生必須通過相應的關卡才能解鎖後續的關卡。



圖3.2 開始選單: 學生必須登入註冊並登入帳號才可以開始進行遊****

圖3.3 關卡選單: 學生必須通過關卡才會解鎖之

在學生正式開始第一關之前，學生首先會進入第零關，如圖3，這個關卡會向學生進行解釋遊戲如何進行。

圖3.4中顯示了的遊戲介面被分為六大區塊，下列的號碼與圖片中的識別號碼相對應：

1. 關卡的簡述以及關卡的指示開啟按鈕

2. 關卡的任務目標，完成的目標會以綠色顯示，否則顯示紅色

3. 檔案區塊，學生有時必須操作檔案進行修改及儲存

4. Git console介面，學生必須在此輸入相應的Git指令

5. Git的視覺化顯示區塊，包含遠端（上半部份）與本地（下半部份），右上角有關卡重啟按鈕，當學生使用不可逆操作時可以使用

6. 與關卡相關的Git指令提示卡片，將鼠標移動至卡片上方時會放大

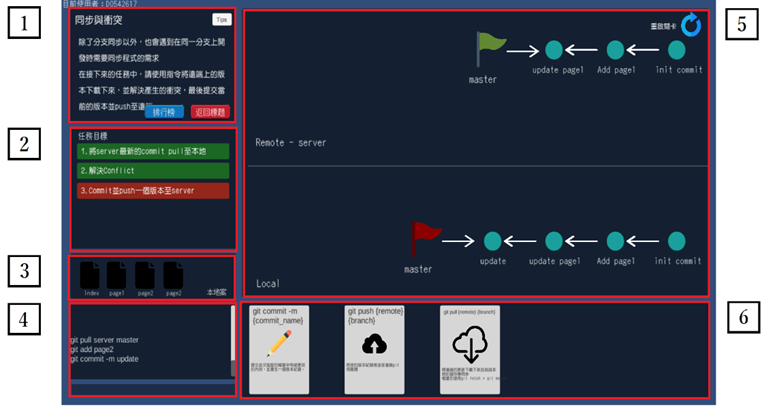


圖3.4 遊戲的六大區塊

學生進入關卡後，關卡的指示會自動被開啟，如圖3.5，閱覽至最後一頁時再按下一頁即會關閉，也可以手動立即關閉，而學生必須照著指示完成目標。



圖3.5 遊戲主畫面



圖3.6 關卡通過畫面

圖3.6顯示了當關卡完成後，除了基本的恭賀訊息外，也向學生顯示在關卡中耗費的時間與輸入指令的次數，這些紀錄都會被傳送至伺服器資料庫，除了可用於後續分析外，還作為遊戲內排行榜依據的一部分，同時章節選單會解鎖相應的關卡，使其成為可自由進出的關卡，以利學生複習。

### ****教學CLI****

本研究採用教學CLI，相關原因有以下幾點：

1. 初學者直接使用GUI容易混淆對Git運作原理的理解，並可能誤將GUI設計的功能當作Git的直接功能

2. Git CLI在所有環境與機器上都是相同的，使用GUI則可能因作業系統不同而無法使用

3. CLI相對GUI更為完整，Git的所有功能都被包含，GUI則不一定

4. 當使用遇到困難時要尋求幫助更為容易，GUI不一定存在完整且良好的文檔，而 CLI則更容易在網路上獲得幫助

J. Lawrance等人提到在圖形界面（GUI）中學習Git被觀察到雖然可以避免學生不適應命令行界面，但容易使學生感到混淆，因此使用CLI仍可能是學生的首選[15]。

## ****遊戲機制****

遊戲元素作為本研究吸引學生主動學習Git的方式，在GEG中引入了以下幾項遊戲設計元素：點數、排行榜、徽章。

1. **點數**

作為最常見的遊戲元素之一，Star[26]發現僅僅採用積分就能增加對任務表現的量化指標，當學生每通過一個關卡時，便會獲得分數，為了配合循序漸進的關卡難度，通過越後期的關卡，學生所獲得的分數也會更多，使學生可以檢視自己的學習進度，而分數作為排行榜的排序依據之一，玩家可以在起始選單中開啟總排行榜查看自己的分數。

1. **排行榜**

排行榜也是最常見的遊戲元素之一，同時也是GEG主要的遊戲性所在，GEG中的排行榜分為兩類，第一種是關卡排行榜，它根據學生通關時的數據進行排名，花費時間較少的學生會在排行榜的前方，花費時間相同時則比較花費的指令行數，並且會列出學生的完成時間，如圖3.7，顯示了關卡4的排行榜紀錄。

一張含有 文字, 計分板 的圖片

自動產生的描述

圖3.7 關卡四的排行榜

第二種則是總排行榜，如圖3.8所示，根據學生在整個遊戲關卡獲得的點數進行排名，同時列出學生獲得的成就數量。

一張含有 文字 的圖片

自動產生的描述

圖3.8 總排行榜

考慮到排行榜可能對於較低名次學生造成反效果[27]，GEG設置了一個開關用以控制是否顯示全部排名，放在排行榜介面的右下角，預設狀態下只會顯示部份排名（前十位），若學生自願開啟則可以查看全部的名次。

1. **成就與獎章**

獎章通常用於象徵玩家的功績，視覺化的成就本身就代表一種獎勵，為成就提供一個獨特的標誌[28]。GEG中設置了十項成就，當學生完成特定的任務時便可以獲得，獲得成就的學生除了可以蒐集到獎章以外也能獲得一定量的點數，鼓勵學生藉由完成特定的操作以在排行榜的競爭中獲得更前面的排名。成就也被分為較基礎且易於獲得的成就與較難獲得的成就，比如：當學生通過了遊戲中的第一關時，可以獲得一個名為入門的成就，圖3.9顯示了學生獲得該成就時跳出提示的畫面。

圖3.9 獲得成就

此外當學生在某個關卡排行榜中成為第一時也可以獲得一個成就，又或是當學生在三十秒內就通關卡時也能夠獲得一個成就，並且GEG還設置了一些趣味性的成就，比如學生將遊戲指示一頁都不閱覽直接關閉時可以獲得一個名為我不需要提示的成就；又或是當學生閱覽一張卡片超過二十五次時可以獲得一個名為卡排之王的成就，這些成就可以增進學生的參與度，同時鼓勵學生在遊戲過程中盡可能的嘗試，以及增加可重玩性。

圖3.10顯示了成就閱覽器，學生可以在成就閱覽器中查看自己目前解鎖的成就，也可以查看那些未被解鎖的成就，在成就閱覽畫面中那些成就的圖示被隱藏起來，但是仍可透過對成就的敘述來了解需要達成什麼目標才能獲得，藉以鼓勵學生嘗試完成這些目標。



圖3.10 成就閱覽畫面

## ****關卡主題與數據監控****

在GEG中，遊戲的第一關（遊戲介紹在第零關）介紹為什麼要使用版本控制軟體，其餘總共有八個關卡直接教授Git指令與概念，分別介紹以下主題：

1. 建立Git倉儲庫（git init）

2. Commit操作（git commit）

3. Push操作（git push）

4. 創立分支（git branch）

5. 合併與刪除分支（git merge and git branch -D）

6. 同步與衝突（git pull and conflict solve）

7. 分支合併與衝突（git merge and conflict solve）

學生在GEG中的活動會被紀錄下來發送至資料庫，教師可以根據狀況即時了解學生的學習狀況，這些活動包含：

1. 學生開始進行關卡

2. 學生完成關卡中的某項目標

3. 學生完成關卡（包含花費時間與行數）

4. 學生開啟遊戲中的提示

5. 學生關閉遊戲中的提示

6. 學生在CLI中輸入的指令

7. 學生目前的獎章數量

8. 每個關卡通過的學生人數

教師可以查看每個關卡通過的人數來判斷學生是否在某些環節遇到困難，比如在圖4.11當中，查閱了關卡8在實驗日時的通過紀錄，我們觀察到只有大約5成的學生有通過關卡，關卡9也同樣只有約一半的學生有通過，教師可以由此得知學生對於分支合併、衝突解決等等並不能即時充分理解與掌握。

一張含有 文字 的圖片

自動產生的描述

圖3.11 關卡八學生通關資料

# 第四章 ****系統展示****

　　此章節展示本研究所提出之系統的各個關卡，並解說我們在各個關卡中如何教學Git的概念，學生應該獲取什麼樣的知識。

## ****遊戲關卡****

## ****遊戲系統****

# 第五章 ****系統實驗****

## ****實驗環境****

**本研究在台灣逢甲大學110學年度上學期的物件導向設計課程中實施，這堂課程分為兩個班級，為了設計與驗證這項系統，隨機選擇一組作為實驗組，另一組作為控制組，分別有54位學生與59位學生，參與課程的學生多為大二的計算機科學學生，此外這堂物件導向設計課程使用的學習平台需要使用Git上傳作業程式碼（包含但不限於CLI）。**

## ****實驗流程****

。

## ****研究分析方法****

## ****實驗數據****

**在此課程中我們利用**偏最小平方結構方程模型設計出問卷給學生填寫以得到問卷的數據。

## ****實驗前置測驗結果與分析****

關於前置測驗的結果如表5.1，本研究使用獨立樣本t檢定來比較控制組與實驗組的平均數是否有所差異，控制組抽樣54個，平均數為72.593；實驗組抽樣59個，平均數為76.271，在變異數同質性檢定中，檢定統計量f值為1.1761，機率值p值為1.4487，未達α=0.05的顯著水準，表示兩組樣本的變異數並無顯著差異，因此獨立樣本t檢定採用變異數相同的檢定統計量t值計算方式，在獨立樣本t檢定中，檢定統計量t值為-1.1918，機率值p值為0.2359，未達α=0.05的顯著水準，因此無法拒絕虛無假設，表示控制組與實驗組兩組的平均數並沒有顯著差異。

表5.1 實驗組與控制組的前置測驗結果

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Group** | **Count** | **Mean** | **Median** | **Minimum** | **Maximum** | **Std. dev.** |
| Control group | 54 | 72.5926 | 70 | 40 | 100 | 15.6838 |
| Experiment group | 59 | 76.2712 | 80 | 30 | 100 | 17.0090 |

## ****實驗後置測驗結果與分析****

## ****問卷調查結果與分析****

1. **模型可靠性和有效性測試**
2. **結構模型之結果分析**
3. **研究問題之結果分析**

上一節當中已經介紹了實驗的結果並完成結構模型的分析與驗證，本節將依照實驗與結構模型的結果回應所述的研究問題。

1. **研究問題1**
2. **研究問題2**
3. **研究問題3**
4. **研究問題4**

# 第六章 結論與未來研究

版本控制軟體的使用能力對軟體工程師而言至關重要，然而學校教育對其著墨甚少，相關的研究也大多是結合版本控制的應用軟體，而遊戲化的學習方式具有相當潛力，因此本研究開發了一個用於教育Git的嚴肅遊戲，並設計與進行一個教育研究實驗，用以評估系統的教育效果，同時本研究基於UTAUT2模型設計了一個研究模型，通過PLS-SEM對問卷調查中的數據進行擬合測試。

根據研究結果我們發現加入GEG作為教育輔助工具的組別相比單純傳統授課的組別有更高的學習成果，並且在這些學生當中，會進行主動學習的學生超過了一半，而這些學生相比整體實驗組學生具有更高的學習成果。此外模型中的因素除了享樂動機以外皆對學生對於Git的態度產生了明顯的直接影響，而遊戲化相關的因素也使學生對於Git的態度產生了間接影響，進而影響學生的行為意圖與後續行為，因此可以認為本研究所提出之系統使學生產生對學習、使用Git的正面影響。

關於未來研究，我們打算更改遊戲的內容，使其在使用時更接近於真實的Git指令，並修正回饋中提到的當機、指示不清楚等等問題。此外我們打算增加更多關卡，這些關卡可以統稱為”挑戰”，提供學生更多練習機會，藉由將學生較不容易理解的部份整合成各種情境挑戰，使遊戲從單純的入門工具成長為可以兼具入門與熟練的工具，遊戲性的部份則要為點數機制設計更多可應用的場景，並增加更多成就、獎章。

# 參考文獻

[1] Lassi Haaranen and Teemu Lehtinen. 2015. Teaching Git on the Side: Version Control System as a Course Platform. In Pro-ceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '15).

[2] Zichermann, G., & Linder, J. 2010. Game-based marketing: inspire customer loyalty through rewards, challenges, and con-tests. John Wiley & Sons.

[3] Christiane Gresse von Wangenheim, Rafael Savi, Adriano Ferreti Borgatto, SCRUMIA—An educational game for teaching SCRUM in computing courses, Journal of Systems and Software, Volume 86, Issue 10, 2013, Pages 2675-2687, ISSN 0164-1212, https://doi.org/10.1016/j.jss.2013.05.030.

[4] V. Venkatesh, J. Y. L. Thong, and X. Xu, ‘‘Consumer acceptance and use of information technology: Extending the unified theory of acceptance and use of technology,’’ MIS Quart., vol. 36, no. 1, pp. 157–178, 2012.

[5] Fleming, T.M.; Bavin, L.; Stasiak, K.; Hermansson-Webb, E.; Merry, S.N.; Cheek, C.; Lucassen, M.; Lau, H.M.; Pollmuller, B.; Hetrick, S. Serious Games and Gamification for Mental Health: Current Status and Promising Directions. Front. Psychiatry 2017, 7, 215. [CrossRef] [PubMed]

[6] F.-H. Huang, ‘‘Adapting UTAUT2 to assess user acceptance of an e- scooter virtual reality service,’’ Virtual Reality, vol. 24, no. 4, pp. 635–643, Dec. 2020.

[7] V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis, ‘‘User acceptance of information technology toward a unified view,’’ MIS Quart., vol. 27, pp. 425–478, Jul. 2003.

[8] Uskov, A.; Sekar, B. Serious Games, Gamification and Game Engines to Support Framework Activities in Engineering: Case Studies, Analysis, Classifications and Outcomes. In Proceedings of the IEEE International Conference on Elec-tro/Information Technology, Milwaukee, WI, USA, 5–7 June 2014; pp. 618–623.

[9] Fleming T, Cheek C, Merry S, Thabrew H, Bridgman H, Stasiak K, et al. Serious games for the treatment or prevention of

depression: a systematic review. Revista de Psicopatología y Psicología Clínica

(2014) 19(3):227-2.10.5944/rppc.vol.19.num.3.2014.13904 [CrossRef] [Google Scholar]

[10] Deterding S, Dixon D, Khaled R, Nacke L. From game design elements to gamefulness: defining gamification. Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments. Tampere: ACM; (2011). p. 9–15. [Google Scholar]

[11] H. M. Lin, M. H. Lee, J. C. Liang, H. Y. Chang, P. Huang, and C. C. Tsai, ‘‘A review of using partial least square structural equation modeling in E- learning research,’’ Brit. J. Educ. Technol., vol. 51, no. 4, pp. 1354–1372, 2019.

[12] Burke JW, McNeill MDJ, Charles DK, Morrow PJ, Crosbie JH, McDonough SM. Optimising engagement for stroke rehabili-tation using serious games. Vis Comput (2009) 25(12):1085–99.10.1007/s00371-009-0387-4 [CrossRef] [Google Scholar]

[13] Chatham RE. Games for training. Commun ACM (2007) 50(7):36–43.10.1145/1272516.1272537 [CrossRef] [Google Scholar]

[14] Mayer, I. Playful Organisations & Learning Systems; Breda University of Applied Sciences: Breda, The Netherlands, 2016.

[15] Katie Seaborn and Deborah I. Fels. Gamification in theory and action. Int. J. Hum.-Comput. Stud., 74(C):14–31, February 2015.

[16] A. Baker, E. O. Navarro and A. van der Hoek, "An experimental card game for teaching software engineering," Proceedings 16th Conference on Software Engineering Education and Training, 2003. (CSEE&T 2003)., 2003, pp. 216-223, doi: 10.1109/CSEE.2003.1191379.

[17] A. Radermacher and G. Walia. Gaps between industry expectations and the abilities of graduates. In Proceeding of the 44th ACM technical symposium on Computer science education, pages 525{530. ACM, 2013.

[18] J. Kelleher. Employing git in the classroom. In Computer Applications and Information Systems (WCCAIS), 2014 World Congress on, pages 1{4. IEEE, 2014.

[19] J. Lawrance and S. Jung. Git on the cloud. Journal of Computing Sciences in Colleges, 28(6):14{15, 2013.

[20] V. Venkatesh, M. G. Morris, G. B. Davis, and F. D. Davis, ‘‘User acceptance of information technology toward a unified view,’’ MIS Quart., vol. 27, pp. 425–478, Jul. 2003.

[21] S. A. Salloum, M. Al-Emran, A. A. Monem, and K. Shaalan, ‘‘Explor- ing students’ acceptance of E-learning through the de-velopment of a comprehensive technology acceptance model,’’ IEEE Access, vol. 7, pp. 128445–128462, 2019.

[22] K. Tamilmani, N. P. Rana, and Y. K. Dwivedi, ‘‘Consumer acceptance and use of information technology: A meta-analytic evaluation of UTAUT2,’’ Inf. Syst. Frontiers, vol. 23, no. 4, pp. 987–1005, Aug. 2021.

[23] Baptista, Goncalo & Oliveira, Tiago. (2017). Why so serious? Gamification impact in the acceptance of mobile banking ser-vices. Internet Research. 27. 118-139. 10.1108/IntR-10-2015-0295.

[24] LIU, Eric Zhi Feng. Avoiding internet addiction when integrating digital games into teaching. Social Behavior and Person-ality: an international journal, 2011, 39.10: 1325-1335.

[25] Anderson, Lorin W., and David R. Krathwohl. A taxonomy for learning, teaching, and assessing: A revision of Bloom's tax-onomy of educational objectives. Longman,, 2001.

[26] Star, K. 2016. Gamification, Interdependence, and the Moderating Effect of Personality on Performance. Doctoral Thesis. University of Coventry.

[27] Michael D. Hanus and Jesse Fox. Assessing the effects of gamification in the classroom: A longitudinal study on intrinsic motivation, social comparison, satisfaction, effort, and academic performance. Computers & Education, 80:152–161, 2015

[28] Juho Hamari, Jonna Koivisto, and Harri Sarsa. Does gamification work? – a literature review of empirical studies on gami-fication. In Proceedings of the 2014 47th Hawaii International Conference on System Sciences, HICSS ’14, pages 3025–3034, Washington, DC, USA, 2014. IEEE Computer Society.