

國立臺東大學 資訊管理學系 115級畢業專題

數網知來-數學教學網站建置 × 數位遊戲學習法

A Mathematics Teaching Website

× Digital Game-Based Learning Approach

專題生：

資管三 詹秉諺 11112118

資管三 許雁欣 11112147

指導教授：

謝明哲 博士

中華民國 114 年 6 月

摘要

本專題《數網知來－數學教學網站建置 × 數位遊戲學習法》旨在結合網站設計與遊戲化學習理念，打造一個能讓國小學生在遊戲中學習數學的互動平台。

研究起因於傳統數學教學缺乏互動與趣味性，導致學生學習動機低落。為解決此問題，本研究以「寵物養成遊戲」與「答題獎勵」為核心機制，開發出具備教材學習、題庫練習、遊戲養成及數據分析等功能的教學網站，讓學生能在輕鬆有趣的環境中學習數學概念，同時提供教師即時掌握學生學習進度的工具。

研究以台東富岡國小二年級學生為對象，透過學習動機量表與前後測驗進行評估，結果顯示學生學習動機明顯提升，平均分數由2.46上升至3.78，證明此系統具有效果與實用性。整體而言，「數網知來」不僅能提升學生的學習興趣與成效，也能協助教師進行精準教學，展現數位遊戲式學習在數學教育中的創新價值與應用潛力。

第一章 緒論

1.1 背景與動機

在台灣的教学體系下，數學作為一門核心學科，其教學方法的現代化已成為教育改革的重要內容。然而，在傳統教學模式下，數學往往被認為枯燥難懂，導致學生對學習的興趣降低，學習成效欠佳。傳統課堂通常以教師講解和紙筆練習為主，俗稱「填鴨式教育」，缺乏互動性與視覺化工具的支持，難以滿足現代學生多樣化的學習需求。同時，教師在教學過程中，缺少能夠即時監控學生學習進度、了解學習困難點的數據支持工具，這使得新型的教學模式難以實現。

在科技的迅猛發展下，數位教學系統為改變這一現狀提供了可能性。多媒體技術、互動學習模組以及人工智慧的引入，使得數學教學能夠更加生動有趣，並提供個性化的學習體驗。然而，目前市面上大多數的數學教學平台功能設計單一，或偏重練習而忽略互動性，往往只是作為「電子版的教科書」，無法真正解決學生與教師的痛點。因此，本研究希望結合數位技術與數學教學需求，設計一個創新且實用的數學教學網站，讓學生能在遊玩中學習知識，作為解決上述問題的具體對策。

1.2 研究目的

本研究的主要目的在於開發一個能夠提升數學教學效率和學習成效的數學教學網站，並結合互動式學習內容、數據分析工具及遊戲，全面滿足學生與教師的需求。透過網站的設計與應用，期望能達成以下具體目標：

1. 提升學生學習興趣與成效：通過互動式模組與遊戲化練習設計，激發學生的學習動力，並幫助學生更加直觀的理解數學概念。
2. 支援學生個性化學習：網站將根據學生的學習進度與表現，自動調整學習內容的難度，提供更加符合其需求的學習路徑。
3. 提供教師教學輔助工具：通過數據分析功能，教師能夠即時掌握學生的學習狀況，針對薄弱環節進行精準教學，提升教學效率。

1.3 研究方法

為實現研究目的，本研究採取以下設計科學研究法(DSRM)進行開發與驗證：

1. 定義問題 (Identify Problem & Motivate)：

傳統的教學模式缺乏互動性，學生較難理解抽象的數學概念，單一填鴨式教學無法讓學生提起興趣，學習成效不佳，現今網上學習資源雖然豐富但缺乏針對性及適應性，導致學習效率低下。

2. 解決方案與目標 (Define Objectives Of A solution)：

進行需求分析，確定學生和教師在數學教學過程中的痛點及需求。設計網站的功能架構，包括互動學習模組、遊戲化設計及個性化學習。

3. 設計與開發 (Design & Development)：

此網站功能模組須具備教學、題庫以及數據分析模組，前端技術採用HTML、JavaScript，支持動態可視化與響應式設計。後端技術使用Python用於數據處理與個性化算法。綜合以上功能並同時確保網站介面友善、操作直觀。

4. 展示 (Demonstration)：

邀請學生和教師作為測試使用者，模擬真實使用場景，驗證系統功能的有效性與穩定性。

5. 評估 (Evaluation)：

蒐集測試數據與使用者反饋，分析網站在教學應用中的實際表現。根據測試結果與反饋進行功能優化與錯誤修復。重複測試與優化，確保網站能夠滿足預期的教育需求。

6. 討論 (Communication)：

目前研究範圍僅限於國小生，無法覆蓋所有年齡層的學習需求，未來期望能將部分功能優化，並將教學模組及題庫逐一增加至其餘年齡層皆可適用；或與AI深度學習技術結合，提升個性化學習的精準度。

1.4 章節架構

第一章 緒論

說明本研究的背景與動機、研究目的、研究方法及章節架構，為後續研究奠定基礎。

第二章 文獻探討

探討與本研究相關的數學教學方法、數位學習平台設計及相關技術應用的現況，分析其優缺點，並確立本研究的創新點與研究價值。

第三章 研究方法

詳細描述數學教學網站的設計理念與開發過程，包括功能架構、系統實作及測試環境。

第四章 系統開發測試與評估

進行系統開發，展示測試過程與結果，評估網站在學生學習成效提升及教師教學支持方面的表現，並討論優化空間。

第五章 結論與建議

總結本研究的主要成果與貢獻，說明系統在數學教學中的應用價值，並針對未來研究方向提出建議。

第二章 文獻探討

2.1 數學教學

數學是什麼？就字面上的意義及原始的定義來說，數學就是數量和空間的科學（Davis & Hersh, 1981）。透過抽象化和邏輯推理，數學成為科學研究的重要基石。其歷史可追溯至古代，起源於實際問題的解決，例如測量土地與記錄時間。隨著文明的進步，數學逐漸發展成為一門獨立且具有高度抽象性的學科。

同時，數學也是現代科學與技術的基礎，是培養邏輯思維和問題解決能力的重要工具。數學教育不僅能促進個人能力的發展，還能推動社會進步。研究指出，數學教育對學生的邏輯思考能力、抽象推理和空間感有重要影響（謝佳鈺，2009）。因此，提升數學教學質量一直是教育研究的重要議題。

在台灣的教育中，數學教學領域的核心素養主要注重於觀察問題、運用工具解決問題、邏輯推理及應用能力。（教育部，2018）在小學階段，重點在於基本數學概念的建立，特別是數量感、運算能力以及幾何的基本認知。課程設計會以生活經驗為基礎，引導學生認識數量與運算的基本概念，加深數學對日常的應用。這些內容不僅有助於建立數學思維，還能促進學生的邏輯推理能力。

數學領域與人們生活息息相關，因此學生對於數學的知識基礎必須得建立好，為了能讓學生能吸收更多知識和提升學習動機，所以本研究也將數位遊戲融入數學領域教學中，來激發學生的學習動機以及提升學習成效。

2.2 數位遊戲式學習

現今教學現場大多為講述式的教學法，以教師為中心的課程，因學童為被動的接收知識，造成學童對於學習數學態度呈現消極的狀況（Bernstein, 1964）由於缺乏自主參與的機會，學生對於學習內容的內在動機往往不高，特別是在數學學科中，這種被動學習模式容易導致學生產生學習倦怠或消極的態度。因此，改善教學方法以提升學生的學習態度，是當前教育中的一項重要課題。

然而，遊戲是兒童生活中的最愛，如果能把數學的學習遊戲化，將可讓學生在一個愉快、自然的情境中獲得數學知識的成長（葉盛昌，2003）。遊戲化學習不僅能激發學生的學習興趣，還能增強其對數學概念的理解與應用能力。兒童大多是主動的參與遊戲活動，不會因為對遊戲沒經驗而有害怕、緊張的心理壓力，在遊戲中更可以不受拘束，自然地將內在想法表現出來（廖育汝，2021）。

現行講述式教學法在數學教學中存在明顯的局限性，遊戲化學習作為一種創新的教學策略，提供一個動態且真實的學習環境，讓學生能夠將抽象的數學概念具體化。能夠有效激發學生的學習興趣，促進其數學概念的理解與應用能力。

2.3 遊戲設計模式

在數位遊戲中，養成式遊戲以其高度沉浸性、目標驅動性及持續回饋機制。養成式遊戲的核心在於讓玩家透過長期的投入與規劃，逐步達成階段性目標，並最終完成一個完整的角色或系統養成過程。這一設計理念為學習平台提供了啟發，特別是在學習內容需要逐步深化與個性化進程的情境下。

首先，目標設定與成長階段設計是養成遊戲的重要特色。遊戲通過設置多層次的目標，讓玩家從簡單的任務開始，逐步挑戰更高難度的任務，最終完成長期目標（新竹師院學報，1993）。這一過程與教育理論中的分層學習一致。在「數網知來」數學教學網站中，可將數學技能劃分為多個養成階段，例如從基礎運算到高階解題，學生需要完成每個階段的挑戰以解鎖下一階段內容。這樣的設計有助於降低學生的挫敗感，並通過每個階段的成功經驗增強學習動機。

其次，資源管理與策略性養成在養成遊戲中也佔據重要地位。玩家需要通過完成任務獲得資源，例如金幣、經驗值或特定道具，並合理分配資源以促進角色或系統的成長（陳俊吉，2013）。在數學學習網站的設計中，資源管理的理念可以轉化為學習資源的積累與應用，例如學生通過完成題目或挑戰獲得「養成道具」，並使用道具來養成屬於自己的角色。這不僅能增加學習的趣味性，還能培養學生的規劃能力。

再者，即時回饋與成就系統是養成遊戲吸引玩家的重要因素。研究表明，即時反饋機制能顯著提升玩家的參與感與沉浸感（林凱胤，2014）。在教育應用中，當學生拿到較高的成績時，系統可立即提供視覺化的回饋，例如升級動畫、積分增長或獲得獎章等，幫助學生即刻感受到努力的成果。此外，成就系統可以設計多樣化的獎勵，例如解鎖新關卡、獲取個性化的角色裝飾，進一步增強學習過程中的趣味性與個人化體驗。

養成式遊戲的設計方式強調持續性學習、階段性目標、即時回饋以及社群互動，為數位學習網站的遊戲化設計提供了重要參考。「數網知來」將採用這些養成式遊戲的設計理念，為學生創造一個富有挑戰性、趣味性和成就感的學習環境。

2.4 資訊系統開發

數學教學網站的設計與實現依賴於專業的資訊系統開發技術與方法。系統架構設計需採用模組化方法，結合教學模組、遊戲模組與數據分析模組，以確保系統的可擴展性與靈活性（黃昱翔，2021）。同時，為了提升跨平台的使用體驗，網站前端設計應採用HTML5和JavaScript等技術，確保網站能在不同設備（如電腦、平板與手機）上順暢運行。

用戶體驗設計（User Experience Design, UX）是數學教學網站成功的關鍵之一。研究表明，清晰直觀的操作介面能顯著降低用戶的學習負擔，進一步提升學習效果（陳俊瑋，2006）。此外，網站還應整合數據分析功能，通過學習記錄追蹤學生的學習進度，為教師提供有價值的教學建議，並為學生提供個性化學習方案。

2.5 均一教育平台

均一教育平台，透過數位平台，讓孩子享有免費優質學習資源、用適合自己的方式學習，「均一」的教學影片和練習題可以因應學童個別差異，提供學童課前預習或課後複習，學童可以依照自己的程度作為補充教材使用（謝婉瑛，2018），透過數學答題練習讓學生理解各單元之概念，答對題數越多，分數、等級越高，若答錯也會有詳細的題解說明，讓學生可以跳脫教室、補習班的學習環境，在家使用電子設備也能學習數學知識。

有別於一般枯燥的答題練習，「均一」有國小部分的完整教學影片和互動練習題，也可以因應各個不同出版的課本，讓使用者容易上手（李依蓉，2018），在答題需要幫忙時，有小題示可以逐步引導協助作答，並搭配點數徽章、提升等級等獎勵方式，激勵學生學習，提升數學練習的互動性和趣味性。

教師亦可以在平台上直接看到學生的學習進度，「技能進展」根據學生各單元的熟練度分為「開始、熟悉、掙扎」等學習成效分級，藉此找出學生較難以理解的單元介入指導透過此功能，教師可以更有效率地分配教學資源，並針對特定學生的需求進行個別化輔導，進一步提升學習效果。此外，該平台也提供數據分析功能，協助教師評估整體教學成效，進行課程規劃上的調整。

第三章 研究方法

3.1 研究架構

本研究主要是在探討，數學教學遊戲對國小學生之學習動機與學習成效之影響，根據研究目的與文獻建立本研究之研究架構，並說明控制變因、操作變因與依變項敘述：

一、教學時間

學習時間為一個學期。

二、教學內容

學習之內容為113學年度三年級下學期之數學教材。

三、地區年級

成員為台東縣富岡國小二年級之學生。

四、評量工具

（一）學習動機：

以學習動機量表(LMS)之前測與後測進行檢測

（二）學習成效：

以教師自定之測驗考卷進行前測與後測來進行檢測

五、解決方案

規劃能提升學習成效與學習動機的數學教學遊戲，開發完成後實際讓國小生使用，進行使用後測驗，以評估學習成效與動機是否有達到提升的效果，若沒達成則找出問題修正重新規劃，直到學習成效與動機達到預期目標。

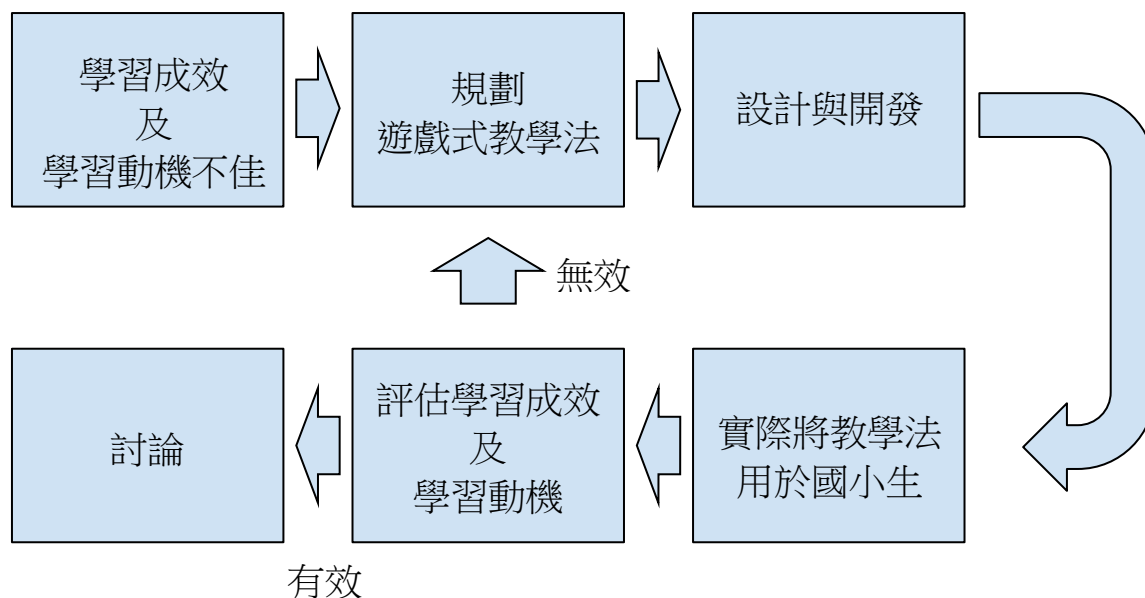


圖3-1：解決方案模型圖

3.2 研究對象

本研究主要在探討數位遊戲式教學對於國小生數學領域之學習動機與學習成效之影響。本研究為台東縣富岡國小，以其中二年級作為研究對象，一班人數為5人，使用本研究設計之數位遊戲進行教學並藉此比較教學前與對教學後的學習動機與學習成效是否有顯著差異。

表 3- 2
研究樣本人數分配

組別	年級	參與實驗之樣本	總計
實驗對象	二年級	5人	5人

3.3 研究假設

根據本研究架構與研究目的，提出以下幾項假設：

假設一：經過教學實驗後，國小學生使用數學教學遊戲之後，學習成效優於傳統式教學之前。

假設二：經過教學實驗後，國小學生使用數學教學遊戲之後，學習動機優於傳統式教學之前。

3.4 研究工具

一、即時反饋式教材

根據教育部編訂之數學教材為範本，進行網站教材製作，並在其中小章節之間多加入小問答，增加即時教學成效反饋，讓教師能更掌握學生分別的學習狀況，並參考學習狀況加強不同學生較弱的章節，達到「因材施教」的效果。

二、數位遊戲學習法

本研究使用的數位遊戲是由研究者自行開發，並讓學生使用電腦等設備進行操作，遊戲教學內容包含國小三年級之課程，學生可自行選擇進度章節來進行自主學習。

1. 養成式遊戲：

遊戲採用養成式遊戲的方式，學生進行自主學習後，可以點擊該單元之小測驗進行養成道具獲取挑戰，完成後測驗，可獲得不同完成度之養成道具，如：100分：特大經驗瓶；90分以上：大經驗瓶；80分以上：中經驗瓶.....以此類推，不同養成道具給予不同「經驗值」，幫助學生自己的角色（寵物）進行養成。

遊戲中也加入圖鑑蒐集功能，裡面記載每種不同物種不同型態的成長完全體，可提起學生的蒐集慾望，達到對遊戲產生興趣的效果。

藉由遊玩遊戲，提起學生興趣，讓學生自發性地進行學習；且養成式遊戲的遊玩具有「長久性」，促使學生不斷的進行遊玩學習，達到寓教於樂的效果

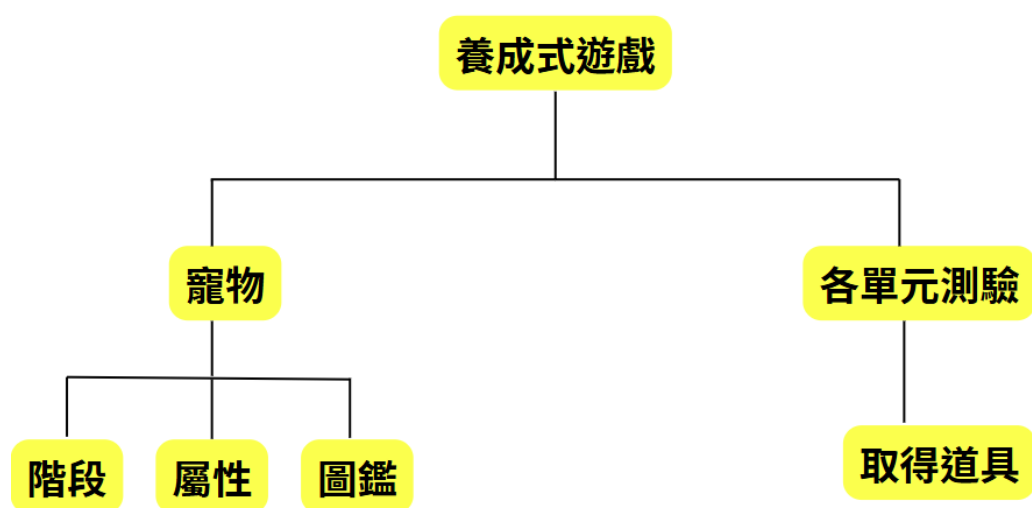


圖3-2：遊戲設計架構

3.5 研究流程

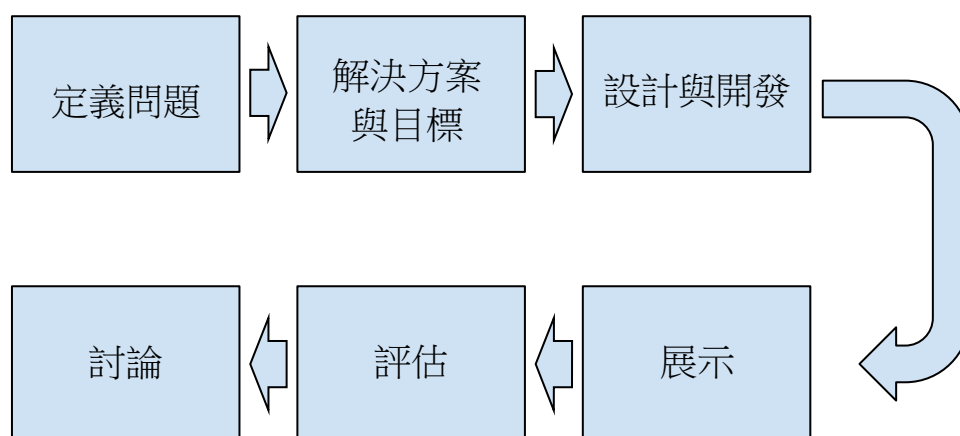


圖3- 3：研究流程圖

（一）定義問題

根據研究背景與動機，查找相關文獻後，與指導教授討論，定義研究的問題與主題，利用數位化工具提升學生的學習效果與參與度，同時減輕教師在課程設計與管理中的負擔。

（二）解決方案與目標

針對上述問題，設計網站的功能架構，包含以下三大核心模組：

- **互動學習模組：**提供分層教學內容，結合動態可視化技術，讓學生能夠以直觀方式學習複雜概念。
- **遊戲化設計：**融入獎勵機制、排行榜與學習挑戰，提升學生的學習積極性與參與度。
- **個性化學習：**藉由數據分析模組，根據學生的學習行為與進度，提供專屬的學習建議與內容推薦，幫助其針對弱項進行強化學習。

（三）設計與開發

此網站功能模組須具備教學、題庫以及數據分析模組，以滿足用戶多元化的學習需求。教學模組將採用分層式學習法，根據學生的能力與學習進度提供動態調整的課程內容，從而實現因材施教的效果。題庫模組將引入基於AI的智能生成與自適應難度調整技術，確保題目內容的多樣化與挑戰性，並通過算法提升學習效果。數據分析模組利用深度學習與數據挖掘技術，分析用戶行為數據，生成個性化學習報告與優化建議，幫助學生更高效地學習。

（四）展示

為了驗證系統功能的有效性與穩定性，將邀請學生和教師作為測試使用者，模擬真實使用場景進行全面測試。例如，學生可透過教學模組完成分層學習內容，並於設計的遊戲裡進行小測驗，記錄其學習進度與正確率。教師則可在數據分析模組中查看班級整體學習表現，並使用個性化推薦功能設計專屬的補充教材。此外，學生與教師之間的互動（如即時答疑功能）也將在測試中被情境化地模擬。

（五）評估

蒐集測試數據與使用者反饋，分析網站在教學應用中的實際表現，測試將涵蓋多種場景，包括學生使用移動設備進行課程學習、教師通過桌面端批改作業與提供反饋，以驗證系統在不同情境下的穩定性與用戶體驗，根據測試結果與反饋進行功能優化與錯誤修復。重複測試與優化，確保網站能夠滿足預期的教育需求。

（六）討論

目前研究範圍、系統的功能與內容設計僅針對國小學生，無法涵蓋所有年齡層的學習需求，未來期望能通過持續的改善與調整，逐步優化網站功能，擴展適用範圍、使教學模組和題庫內容能涵蓋更多年齡和學習階段，滿足所有需要學習數學的學生。

第四章 系統開發測試與評估

本章節將說明系統的開發與測試過程，並呈現學生在使用本系統進行學習後的學習成效，以及教師對於系統作為教學輔助工具的回饋與評估。

4.1 系統開發過程

本系統以 HTML、CSS、JavaScript 為基礎進行開發，搭配 Tailwind CSS 美化介面。系統功能包含使用者登入驗證、學習活動互動（答題、道具蒐集、寵物養成）與題目編輯功能（限教師）。

前端技術將採用HTML與JavaScript，支持動態可視化（例如互動式圖表、動態知識樹），確保在不同設備上的流暢體驗。後端技術使用Python，結合自然語言處理、機器學習算法以及推薦系統，實現數據處理與個性化學習體驗。同時，網站將深度結合AI技術，提供智能化輔助功能，如即時答疑、學習進度追蹤以及專屬學習計劃生成。綜合以上功能，網站將確保介面友善、操作直觀，並成為一個以用戶為中心的學習平台。

（一）Photoshop

Photoshop 是由 Adobe 所開發的影像處理軟體，該軟體發布在 Windows 和 macOS 平台上。Photoshop 主要針對像素所構成的數位影像進行編輯，內建多種繪圖與調整工具，能夠有效地完成圖片的修飾與創作工作。此外，其圖層編輯功能讓使用者能夠靈活地分層操作影像內容，並搭配歷史記錄功能進行多次測試與比較。該軟體支援各種濾鏡效果，可以實現豐富多樣的視覺創意，適合影像設計與處理需求。

（二）HTML

HTML 是一種程式語言，主要用於描述網頁的結構與內容，適用於所有支援瀏覽器的平台上。HTML 以標籤的形式定義各種網頁元素，例如文字段落、圖片和超連結等，能夠清晰地組織網頁資訊。HTML 的層級結構設計，讓開發者能方便地管理內容的層次，並與 CSS 和 JavaScript 一同實現完整的網頁設計。作為前端開發的基石，HTML 提供了一個簡單且易於擴展的框架，適合用於各種網站的建構與維護。

(三) Javascript

JavaScript 是一種用於網頁的程式語言，適用於多種作業系統的瀏覽器環境。JavaScript 主要用於實現網頁中的互動功能，例如按鈕觸發、內容更新和動畫效果等，使網頁不僅僅是靜態的內容展示。JavaScript 的動態特性讓開發者能夠即時改變頁面元素的狀態，並與使用者互動。透過與 HTML 和 CSS 的結合，JavaScript 能夠構建出具有豐富功能與靈活性的網頁應用，是前端開發中不可或缺的工具之一。

(四) MySQL

MySQL 是一種開源的關聯式資料庫管理系統 (RDBMS)，廣泛應用於網站開發中，用於儲存和管理大量的結構化資料。它支援多使用者、多執行緒的環境，具有高效能、穩定性高和易於使用的特點。本系統使用 MySQL 儲存使用者資訊、測驗題目、寵物養成數據以及學生的學習歷程等關鍵數據。透過標準的 SQL (Structured Query Language) 語法，系統能夠對資料庫進行查詢、新增、修改和刪除操作，確保數據的完整性與一致性。MySQL 的優勢在於其可靠性、擴展性、安全性以及豐富的社群支援，能夠為本系統提供穩健的後端資料支援。

4.2 重要程式碼功能說明

本節將針對系統中一些關鍵的程式碼檔案與功能進行說明，以展示系統的核心運作邏輯，特別是前後端資料互動的部分。

(一) server.js - 後端 API 與 MySQL 資料庫互動

server.js 檔案是本系統的後端核心，使用 Node.js 搭配 Express 框架搭建 API 服務，並透過 mysql2/promise 模組與 MySQL 資料庫進行互動。它負責處理前端發送的資料請求，並對資料庫中的測驗題目進行增、刪、改、查的操作。

重要功能片段：

1. 列出所有題目

```
app.get('/api/questions', async (req, res) => {
  try {
    const [rows] = await pool.query(
      'SELECT id, question, options, correct_answer FROM questions'
    );
    const questions = rows.map(r => {
      let raw = r.options == null
        ? ''
        : (typeof r.options === 'string'
          ? r.options
          : r.options.toString());
      if (raw.charCodeAt(0) === 0xFEFF) raw = raw.slice(1);
      raw = raw.trim();

      let opts;
      try {
        opts = JSON.parse(raw);
        if (!Array.isArray(opts)) throw new Error('not array');
      } catch {
        opts = raw === ''
          ? []
          : raw.split(',').map(s => s.trim()).filter(s => s);
      }

      return {
        id:    r.id,
        q:     r.question,
        opts,
        correct: r.correct_answer
      };
    });
    res.json(questions);
  } catch (err) {
    console.error(err);
    res.status(500).json({ error: err.message });
  }
});
```

2. 取單題（編輯時用）

```
app.get('/api/questions/:id', async (req, res) => {
  try {
    const [rows] = await pool.query(
      'SELECT id, question, options, correct_answer FROM questions WHERE id = ?',
      [req.params.id]
    );
    if (!rows.length) return res.status(404).json({ error: 'not found' });
    const r = rows[0];

    let raw = r.options == null
      ? ''
      : (typeof r.options === 'string'
        ? r.options
        : r.options.toString());
    if (raw.charCodeAt(0) === 0xFEFF) raw = raw.slice(1);
    raw = raw.trim();

    let opts;
    try {
      opts = JSON.parse(raw);
      if (!Array.isArray(opts)) throw new Error('not array');
    } catch {
      opts = raw === ''
        ? []
        : raw.split(',').map(s => s.trim()).filter(s => s);
    }

    res.json({
      id:      r.id,
      q:       r.question,
      opts,
      correct: r.correct_answer
    });
  } catch (err) {
    console.error(err);
    res.status(500).json({ error: err.message });
  }
});
```

3. 新增、更新、刪除題目

```
app.post('/api/questions', async (req, res) => {
  const { question, options, correct_answer } = req.body;
  await pool.query(
    'INSERT INTO questions (question, options, correct_answer) VALUES (?, ?, ?)',
    [question, JSON.stringify(options), correct_answer]
  );
  res.status(201).json({ message: 'created' });
});

app.put('/api/questions/:id', async (req, res) => {
  const { question, options, correct_answer } = req.body;
  await pool.query(
    'UPDATE questions SET question=?, options=?, correct_answer=? WHERE id=?',
    [question, JSON.stringify(options), correct_answer, req.params.id]
  );
  res.json({ message: 'updated' });
});

app.delete('/api/questions/:id', async (req, res) => {
  await pool.query('DELETE FROM questions WHERE id=?', [req.params.id]);
  res.json({ message: 'deleted' });
});
```

功能解釋：

- **資料庫連線池 (mysql.createPool)：** 後端伺服器啟動時，會建立一個 MySQL 連線池。連線池能夠有效管理資料庫連線，避免頻繁建立和關閉連線所帶來的效能開銷，提高應用程式的響應速度和穩定性。
- **API 實作：** server.js 定義了多個 RESTful API 端點，用於對測驗題目進行 CRUD (Create, Read, Update, Delete) 操作：
 - GET /api/questions：獲取所有題目列表。程式碼中特別處理了 options 欄位，它在資料庫中可能以 JSON 字串或逗號分隔字串的形式儲存，後端會將其解析為 JavaScript 陣列，方便前端使用。
 - GET /api/questions/:id：根據題目 ID 獲取單一題目。
 - POST /api/questions：新增題目。接收前端傳來的題目內容、選項陣列和正確答案，其中選項陣列會被轉換為 JSON 字串儲存到資料庫中。
 - PUT /api/questions/:id：更新題目。根據題目 ID 修改現有題目資料。
 - DELETE /api/questions/:id：刪除題目。根據題目 ID 刪除指定的題目。
- **錯誤處理：** 每個 API 請求都包含了 try...catch 區塊，用於捕獲資料庫操作可能發生的錯誤，並返回適當的錯誤狀態碼和訊息給前端，提升系統的健壯性。
- **CORS 支援：** app.use(cors()) 啟用跨域資源共享 (CORS)，確保前端（運行在不同的埠或網域）能夠正常訪問後端 API。

(二) login.js - 使用者登入驗證 (前端)

login.js 檔案負責前端的使用者登入介面邏輯。儘管此處的程式碼是基於預設的靜態帳號進行驗證，但其結構清晰，可輕鬆擴展以與上述的後端 API 進行整合，實現動態的使用者身份驗證。

重要功能片段：

1. 帳號與登入跳轉：

```
const accounts = {
  student: [
    { username: 'student1', password: 'password1' },
    { username: 'student2', password: 'password2' },
  ],
  teacher: [
    { username: 'teacher1', password: 'password1' },
    { username: 'teacher2', password: 'password2' },
  ]
};

const handleLoginSuccess = (role) => {
  loginMessage.textContent = `${role === 'teacher' ? '教師' : '學生'}登入成功！即將前往主頁...`;
  loginMessage.style.color = 'green';
  loginBtn.disabled = true;
  localStorage.setItem('token', `${role}_token`);
  localStorage.setItem('role', role);

  setTimeout(() => {
    window.location.href = 'index.html';
  }, 1500);
};
```

2. 登入功能：

```
const login = () => {
  const username = loginUsername.value.trim();
  const password = loginPassword.value;

  if (!username || !password) {
    handleLoginFailure('請輸入使用者名稱和密碼。');
    return;
  }

  loginBtn.disabled = true;
  loginMessage.textContent = '登入中...';
  loginMessage.style.color = 'gray';

  setTimeout(() => {
    let foundUserRole = null;

    const student = accounts.student.find(
      (account) => account.username === username && account.password === password
    );
    if (student) {
      foundUserRole = 'student';
    }

    if (!foundUserRole) {
      const teacher = accounts.teacher.find(
        (account) => account.username === username && account.password === password
      );
      if (teacher) {
        foundUserRole = 'teacher';
      }
    }

    if (foundUserRole) {
      handleLoginSuccess(foundUserRole);
    } else {
      handleLoginFailure('使用者名稱或密碼錯誤。');
    }
  }, 500);
};
```

功能解釋：

- **靜態帳號驗證與角色判斷：** 目前程式碼中直接編碼了學生和教師的帳號資訊。當使用者嘗試登入時，系統會比對輸入的帳號密碼是否與這些預設值匹配，並判斷其角色。在實際使用環境中，此部分建議修改為向後端 API (server.js) 發送登入請求，由後端進行資料庫驗證。
- **本地儲存角色信息：** 登入成功後，使用者的角色 (student 或 teacher) 會被儲存在瀏覽器的 localStorage 中。這使得前端頁面可以在不同頁面間共享使用者角色資訊，例如在主頁面根據角色顯示不同的功能按鈕。
- **前端狀態回饋與頁面跳轉：** 登入過程中，會透過 loginMessage 元素向使用者提供即時的回饋，如「登入中...」、「登入成功」或「錯誤訊息」。登入成功後，系統會自動跳轉至主頁 (index.html)。

(三) index.html (內嵌 script) - 主頁面功能控制

主頁面 index.html 內嵌的 JavaScript 負責根據登入的使用者角色動態調整頁面內容，特別是控制教師專用功能的顯示。

重要功能片段：

1. 身分登入時的不同介面

```
<script>
  document.addEventListener('DOMContentLoaded', () => {
    const editQuizBtn = document.getElementById('editQuizBtn');
    const logoutBtn = document.getElementById('logoutBtn');
    const userInfo = document.getElementById('userInfo');

    const role = localStorage.getItem('role');

    if (!role) {
      userInfo.textContent = '目前身分：尚未登入';
      editQuizBtn.classList.add('hidden');
      logoutBtn.classList.add('hidden');
    } else {
      logoutBtn.classList.remove('hidden');
      if (role === 'teacher') {
        userInfo.textContent = '目前身分：教師';
        editQuizBtn.classList.remove('hidden');
        editQuizBtn.style.display = 'inline-block';
        editQuizBtn.onclick = () => {
          window.location.href = 'edit_quiz.html';
        };
      } else if (role === 'student') {
        userInfo.textContent = '目前身分：學生';
        editQuizBtn.classList.add('hidden');
      }
    }

    logoutBtn.addEventListener('click', () => {
      localStorage.removeItem('token');
      localStorage.removeItem('role');
      window.location.reload();
    });
  });
</script>
```

功能解釋：

- **角色判斷與動態顯示：** 在頁面載入時，腳本會從 `localStorage` 中讀取儲存的使用者角色 (`role`)。如果角色為 `teacher`，則會顯示「編輯題目」按鈕 (`editQuizBtn`)，並設定其點擊事件，導向 `edit_quiz.html` 頁面；如果是 `student` 或未登入，則該按鈕會被隱藏。
- **登出功能：** 「登出」按鈕 (`logoutBtn`) 的事件監聽器負責清除 `localStorage` 中儲存的 `token` 和 `role` 資訊，並將使用者導向登入頁面，實現完整的登出流程。

(四) edit_quiz.html (內嵌 script) - 題目編輯功能 (前端)

edit_quiz.html 頁面提供了教師編輯和新增測驗題目的前端介面。其內嵌的 JavaScript 負責與後端 server.js 中定義的 API 進行互動，實現題目的增刪改查功能。

重要功能片段：

1. 讀取所有題目並顯示

```
async function loadQuestions() {
  const res = await fetch(apiBase);
  const list = await res.json();
  const tbody = document.getElementById("questionList");
  tbody.innerHTML = "";
  list.forEach(q => {
    const tr = document.createElement("tr");
    tr.innerHTML = `
      <td class="p-2 border text-center">${q.id}</td>
      <td class="p-2 border">${q.q}</td>
      <td class="p-2 border">${q.opts.join(" / ")}</td>
      <td class="p-2 border">${q.correct}</td>
      <td class="p-2 border text-center space-x-2">
        <button data-id="${q.id}" class="editBtn px-2 py-1 bg-yellow-400 rounded">編輯</button>
        <button data-id="${q.id}" class="delBtn px-2 py-1 bg-red-500 text-white rounded">刪除</button>
      </td>
    `;
    tbody.appendChild(tr);
  });
  document.querySelectorAll(".editBtn").forEach(b =>
    b.onclick = () => editQuestion(b.dataset.id)
  );
  document.querySelectorAll(".delBtn").forEach(b =>
    b.onclick = () => deleteQuestion(b.dataset.id)
  );
}
```

2. 新增或更新題目

```
async function handleSave() {
  const id = document.getElementById("questionId").value;
  const qText = document.getElementById("questionText").value.trim();
  const nodes = Array.from(
    document.querySelectorAll("#choicesContainer input[type=text]")
  );
  const opts = nodes.map(n => n.value.trim());
  const correct = document.querySelector(
    'input[name="correctAnswer"]:checked'
  )?.value;

  if (!qText || opts.some(o => !o) || correct == null) {
    alert("題目、選項、正確答案都必填!");
    return;
  }

  const payload = {
    question: qText,
    options: opts,
    correct_answer: opts[correct]
  };

  const url = id ? `${apiBase}/${id}` : apiBase;
  const method = id ? "PUT" : "POST";

  await fetch(url, {
    method,
    headers: { "Content-Type": "application/json" },
    body: JSON.stringify(payload)
  });
  resetForm();
  await loadQuestions();
}
```

功能解釋：

- **前端與後端 API 互動：** 頁面中的 JavaScript 使用 fetch API 與 server.js 中定義的 /api/questions 端點(MySQL)進行互動。
 - loadQuestions() 函式發送 GET 請求獲取所有題目，並將其動態渲染到表格中。
 - saveQuestion() 函式根據是否有 questionId 判斷是新增 (POST) 還是更新 (PUT) 題目。它將表單資料組織成 JSON 格式發送給後端。

- editQuestion() 函式發送 GET 請求獲取特定題目 ID 的資料，並將其填充到表單中，方便教師進行編輯。
- deleteQuestion() 函式發送 DELETE 請求到後端，實現刪除功能。
- **資料格式轉換：** 前端在發送數據到後端時，會將選項陣列透過 `JSON.stringify(payload)` 轉換為 JSON 字串，以便後端能夠正確儲存到 MySQL 資料庫的 options 欄位。同樣地，從後端獲取資料時，也會將 options 字串解析回陣列。
- **使用者介面回饋：** 透過 alert 函式提供操作提示（如驗證錯誤、刪除確認等），並在操作完成後重新載入題目列表，確保介面數據的即時更新。

(五) tutorial.js - 教材學習與測驗 (前端)

tutorial.js 管理教材學習頁面的動態內容和測驗互動。它包含了教材的文字內容和隨附的練習題目。

重要功能片段：

1. 載入教材內容

```
function loadLesson() {  
  lessonSlide.classList.add('fade-out');  
  setTimeout(() => {  
    lessonSlide.innerHTML = lessons[currentLessonIndex].content;  
    loadQuiz();  
    updateButtonVisibility();  
    lessonSlide.classList.remove('fade-out');  
    lessonSlide.classList.add('fade-in');  
  }, 300);  
}
```

2. 載入測驗題目

```
function loadQuiz() {  
  const quiz = lessons[currentLessonIndex].quiz;  
  quizQuestion.textContent = quiz.question;  
  quizOptions.innerHTML = quiz.options  
    .map(  
      (option, index) => `  
        <label>  
          <input type="radio" name="quizOption" value="${option}">  
            ${option}  
        </label>  
      `  
    )  
    .join('');  
  quizResult.textContent = '';  
}
```

3. 提交答案

```
submitQuizBtn.addEventListener('click', () => {
  const selectedOption = document.querySelector('input[name="quizOption"]:checked');
  if (selectedOption) {
    const answer = selectedOption.value;
    if (answer === lessons[currentLessonIndex].quiz.correctAnswer) {
      quizResult.textContent = '答對了！你好棒！';
    } else {
      quizResult.textContent = '答錯了，再想想看喔！';
    }
  } else {
    quizResult.textContent = '請選擇一個答案！';
  }
});
```

功能解釋：

- **靜態教材與測驗內容：** 教材和測驗內容以 JavaScript 陣列 (lessons) 的形式直接定義在程式碼中。每個元素包含教材的 HTML 片段和對應的測驗題目（問題、選項、正確答案）。
- **頁面動態渲染：** loadLesson() 函式根據當前的 currentLessonIndex 動態更新 lessonSlide 元素顯示教材內容，並呼叫 loadQuiz() 函式來渲染對應的測驗題目和選項。
- **測驗互動：** 當使用者點擊「提交答案」按鈕時，腳本會檢查選定的答案是否與教材中定義的正確答案匹配，並在 quizResult 元素中顯示答題結果。
- **導航控制：** prevBtn 和 nextBtn 允許使用者在教材頁面之間順序瀏覽。backToMainBtn 則提供便捷的返回主頁面的功能。

(六) game.js - 數學遊戲與寵物養成

game.js 包含了數學養成遊戲的所有邏輯，包括寵物狀態管理、答題機制、道具收集和圖鑑系統。在題庫部分也結合剛才教師可進行的題庫編輯，使答題題庫與MySQL(後端)進行連接，達成題庫的多變化性。

重要功能片段：

1. 載入題庫與題庫抓取

```
// 題庫容器，等 API 載入
this.questions = [];

this.bindEvents();

// 載入題庫，載入完成再啟用「開始答題」
// 一開始先鎖住「開始答題」
this.goToQuizBtn.disabled = true;

// 載題庫，成功後才解鎖
this.loadQuestionsFromAPI()
  .then(() => {
    this.goToQuizBtn.disabled = false;
  });
```

```
async loadQuestionsFromAPI() {
  const res = await fetch('http://localhost:3000/api/questions');
  if (!res.ok) throw new Error(`HTTP ${res.status}`);
  this.questions = await res.json();
}
```

2. 答題的多變化性設計(隨機取題)

```
startQuiz() {
  this.backToMainBtn.style.display = "none";
  this.startQuizBtn.disabled = true;
  this.submitAnswerBtn.disabled = false;
  // 隨機取 5 題
  this.selectedQuestions = this.questions
    .sort(() => 0.5 - Math.random())
    .slice(0, 5);
  this.questionContainer.innerHTML = "";
  this.questionContainer.classList.remove("hidden");
  this.loadQuestion();
}

loadQuestion() {
  const q = this.selectedQuestions[this.currentQuestionIndex];
  let html = `<p>${q.q}</p>`;
  q.opts.forEach(o => {
    html += `<label><input type="radio" name="answer" value="${o}"> ${o}</label>`;
  });
  this.questionContainer.innerHTML = html;
}
```

3. 答題成績計算

```
submitAnswer() {
  const sel = document.querySelector("input[name='answer']:checked");
  if (!sel) {
    alert("請選擇答案！");
    return;
  }
  if (sel.value === this.selectedQuestions[this.currentQuestionIndex].correct) {
    this.score += 20;
  }
  this.currentQuestionIndex++;
  if (this.currentQuestionIndex < this.selectedQuestions.length) {
    this.loadQuestion();
  } else {
    this.endQuiz();
  }
}
```

4. 經驗值養成

```
checkEvolution() {
  if (this.petStage === 0 && this.experience >= 100) {
    this.petStage = 1;
    this.petImage.classList.remove("pet-shake");
    this.updatePetImage();
    alert("恭喜孵化成幼年期！");
  }
  else if (this.petStage === 1 && this.experience >= 400) {
    this.petStage = 2;
    this.attributeSelection.classList.remove("hidden");
    alert("經驗達到400，請選擇進化屬性！");
    this.updatePetImage();
  }
  else if (this.petStage === 3 && this.experience >= 900) {
    this.petStage = 4;
    this.unlocked[this.petSpecies].push(this.petAttribute);
    this.updatePetImage();
    alert("恭喜進化成成年體！");
    alert("可以開始養下一隻寵物囉！");
    this.nextPetBtn.classList.remove("hidden");
  }
}
```

5. 寵物進化養成

```
updatePetImage() {
  if (!this.petSpecies) {
    this.petImage.style.display = "none";
    return;
  }
  this.petImage.classList.remove("youth-stage", "adult-stage");
  let src = `images/eggs/${this.petSpecies}_egg.png`;
  if (this.petStage === 1) {
    src = `images/pets/${this.petSpecies}_baby.png`;
  }
  else if (this.petStage === 2) {
    src = `images/pets/${this.petSpecies}_baby.png`;
  }
  else if (this.petStage === 3) {
    src = `images/pets/${this.petSpecies}_${this.petAttribute}_youth.png`;
    this.petImage.classList.add("youth-stage");
  }
  else if (this.petStage === 4) {
    src = `images/pets/${this.petSpecies}_${this.petAttribute}_adult.png`;
    this.petImage.classList.add("adult-stage");
  }
  this.petImage.src = src;
  this.petImage.style.display = "block";
}
```

功能解釋：

● 寵物養成核心：

- updatePetDisplay() 函式根據寵物的等級動態更新其圖像（幼年期/成年期），並在達到成年等級時觸發圖鑑的解鎖。圖鑑（unlocked 物件）追蹤已獲得的寵物類型。
- updatePetStatus() 函式處理經驗值的累積和寵物的升級邏輯，並更新介面上的經驗條和等級顯示。

● 答題與獎勵機制：

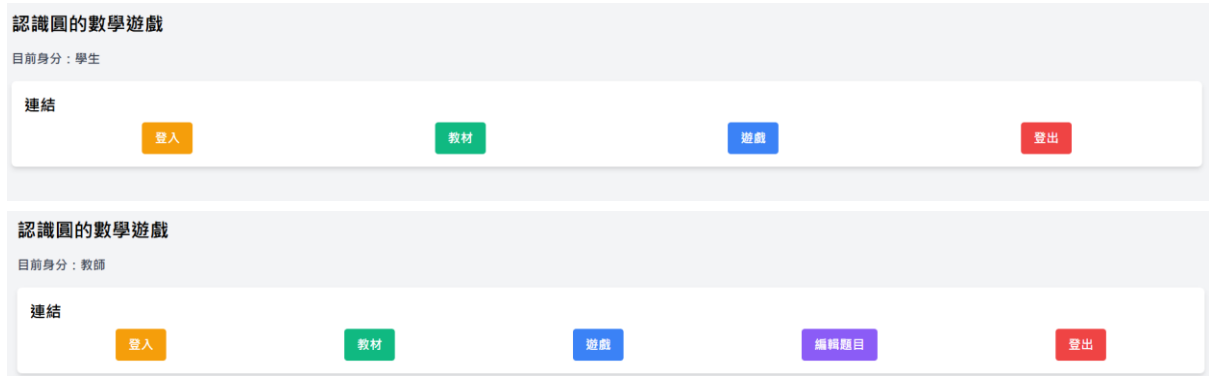
startQuiz() 函式負責準備題目（目前為靜態示範，但可擴展為從後端獲取隨機題目），submitAnswer() 函式判斷答案的正確性，並根據結果增加經驗值、連對次數以及給予道具獎勵（getAward() 函式）。

4.3 測試過程與結果

針對學生與教師進行不同面向的系統測試，並彙整如下表所示，並配合圖片展示：

測試項目	測試對象	結果與說明
1. 登入系統	學生、教師	能成功登入並根據角色進入不同功能區。
2. 答題功能	學生	能依題目作答並獲得對應道具，系統反應正常。
3. 題目編輯功能	教師	教師帳號登入後能成功新增、修改題目內容與選項。
4. 遊戲視覺介面與操作性	學生、教師	介面簡潔，學生操作容易；教師亦能快速上手。

(1) 登入頁面、不同對象登入畫面



(2) 答題功能



(3)教師編輯題目

編輯 / 新增題目

題目內容

請輸入題目

選項 (必填 4 個) & 選擇正確答案：

- ☐ 選項 1
- ☐ 選項 2
- ☐ 選項 3
- ☐ 選項 4

儲存題目

取消編輯 / 清空

現有題庫

#	題目	選項	正確	操作
3	【提示：周長 = $2 \times \text{半徑} \times \pi$ 】有一個圓半徑3公分，周長大約是多少呢？($\pi=3.14$)	9 公分 / 18 公分 / 28 公分 / 36 公分	18 公分	編輯 刪除
4	【提示：面積 = $\text{半徑} \times \text{半徑} \times \pi$ 】有一個圓半徑4公分，面積大約是多少呢？($\pi=3.14$)	12 平方公分 / 25 平方公分 / 48 平方公分 / 100 平方公分	48 平方公分	編輯 刪除
5	圓周率是哪個符號呢？	π / θ / Ω / Δ	π	編輯 刪除
6	下列不是圓形的物體是？	籃球 / 甜甜圈 / 披薩 / 正方形餅乾	正方形餅乾	編輯 刪除
9	【提示：直徑 $\div 2 = \text{半徑}$ 】有一個圓直徑8公分，半徑是多少呢？	2 公分 / 4 公分 / 6 公分 / 8 公分	4 公分	編輯 刪除
10	【提示：直徑 $\div 2 = \text{半徑}$ 】有一個圓直徑20公分，半徑是多少呢？	10 公分 / 12 公分 / 15 公分 / 5 公分	10 公分	編輯 刪除
11	【提示：半徑 $\times 2 = \text{直徑}$ 】有一個圓半徑5公分，直徑是多少呢？	5 公分 / 10 公分 / 15 公分 / 20 公分	10 公分	編輯 刪除
12	【提示：半徑 $\times 2 = \text{直徑}$ 】有一個圓半徑6公分，直徑是多少呢？	20 公分 / 16 公分 / 12 公分 / 3 公分	12 公分	編輯 刪除
	【提示：面積 = $\text{半徑} \times \text{半徑} \times \pi$ 】有一個圓半徑6	6 平方公分 / 27 平方公分 / 100	100 平方	編輯

```

MySQL 8.0 Command Line Client
mysql> SELECT * FROM questions;
+-----+-----+-----+
| id | question | options | correct_answer |
+-----+-----+-----+
| 3 | 【提示：周長 = 2 × 半徑 × π】 有一個圓半徑3公分，周長大約是多少呢？(π=3.14) | ["9 公分", "18 公分", "28 公分", "36 公分"] | 18 公分 |
| 4 | 【提示：面積 = 半徑 × 半徑 × π】 有一個圓半徑4公分，面積大約是多少呢？(π=3.14) | ["12 平方公分", "25 平方公分", "48 平方公分", "100 平方公分"] | 48 平方公分 |
| 5 | 圓周率是哪個符號呢？ | ["π", "θ", "Q", "Δ"] | π |
| 6 | 下列不是圓形的物體是？ | ["籃球", "甜甜圈", "披薩", "正方形餅乾"] | 正方形餅乾 |
| 9 | 【提示：直徑 ÷ 2 = 半徑】 有一個圓直徑8公分，半徑是多少呢？ | ["2 公分", "4 公分", "6 公分", "8 公分"] | 4 公分 |
| 10 | 【提示：直徑 ÷ 2 = 半徑】 有一個圓直徑20公分，半徑是多少呢？ | ["10 公分", "12 公分", "15 公分", "5 公分"] | 10 公分 |
| 11 | 【提示：半徑 × 2 = 直徑】 有一個圓半徑5公分，直徑是多少呢？ | ["5 公分", "10 公分", "15 公分", "20 公分"] | 10 公分 |
| 12 | 【提示：半徑 × 2 = 直徑】 有一個圓半徑6公分，直徑是多少呢？ | ["20 公分", "16 公分", "12 公分", "3 公分"] | 12 公分 |
| 13 | 【提示：面積 = 半徑 × 半徑 × π】 有一個圓半徑6公分，面積大約是多少呢？(π=3.14) | ["6 平方公分", "27 平方公分", "108 平方公分", "432 平方公分"] | 108 平方公分 |
+-----+-----+-----+
9 rows in set (0.00 sec)

```

(4)遊戲視覺介面與操作展示



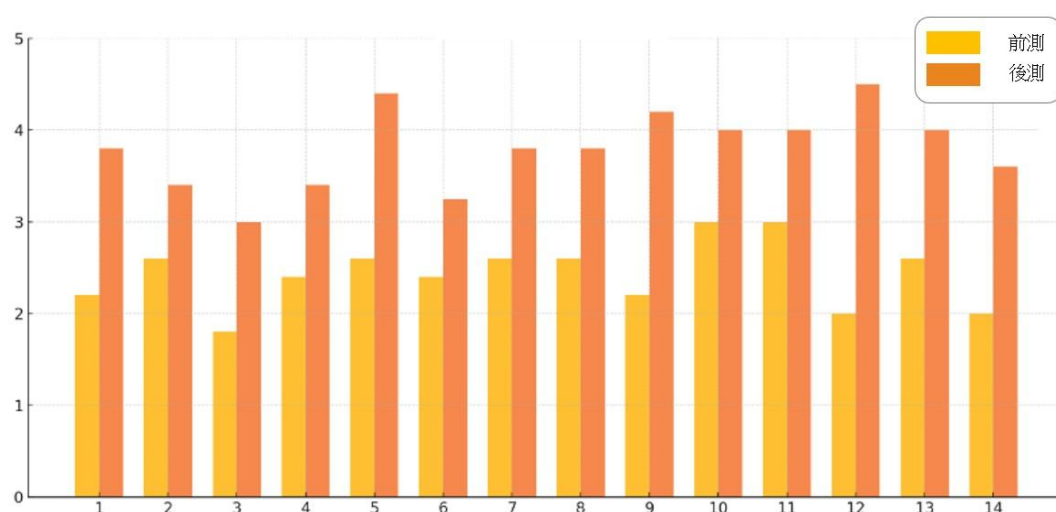
4.4 學習成效與教學支持評估

透過問卷調查學生與教師對系統的回饋，統整如下表所示：

評估面向	學生回饋	教師回饋
學習動機提升	大部分學生表示遊戲機制提升學習興趣。	認為遊戲化可吸引學生參與。
操作便利性	操作簡單易懂。	能快速管理與編輯題目。
內容理解度	題目設計清楚易懂，有助於複習。	可作為課後練習補充教材。

以google問卷作學習動機量表(LMS)前後測，

受測者共5位，皆為富岡國小2年級之學生，測試結果如下圖：



學習動機量表(LMS)前後測對比

整體趨勢

全量表平均分數由 2.46 → 3.78，提升 +1.32（約 +27 % 滿分幅度）。

五位受試者中，人人後測總分皆高於前測，顯示教學活動對「閱讀數學文章動機／興趣」有一致性的正向效果。

4.5 優化空間

部分學生反映希望增加更多寵物或關卡元素以增添趣味，教師則建議加入答題紀錄分析與匯出功能。

第五章 結論與建議

5.1 研究設計哲學

本研究的設計哲學根植於「以學習者為中心」的教育理念，並融合「遊戲化學習」與「數據驅動」的創新思維。傳統的單向式教學已難以滿足當代學生的多樣化學習需求，尤其在抽象且枯燥的數學領域。因此，本系統的設計旨在轉變學習模式，從被動接受轉向主動探索與創造。

1. 遊戲化學習：

我們將遊戲的核心元素～目標設定、即時回饋、成就獎勵和競爭互動～融入數學學習。這不僅僅是將數學題目包裝成遊戲，更是透過寵物養成、道具收集、等級提升、圖鑑蒐集等機制，激發學生內在的學習動機與興趣。透過遊戲，學習不再是枯燥的任務，而是充滿挑戰與樂趣的探索過程。這種設計哲學旨在利用遊戲的引人入勝之處，將學習行為轉化為一種自我驅動的娛樂體驗，從而提升學習投入度和持久性。

2. 個性化與適應性學習：

目前系統在後端 AI 整合方面尚處於規劃階段，但其前端設計已為未來的個性化學習奠定基礎。透過教師可編輯題目以及學生答題數據的收集，系統將能逐步建構學生學習行為模型。未來的目標是利用 AI 算法分析這些數據，自動調整題目難度、推薦合適的學習內容和學習路徑，實現真正的「因材施教」。這強調的是根據每個學生的獨特進度、優勢和弱點，提供量身定制的學習體驗，確保學習資源的最大化利用和學習效果的最優化。

3. 數據驅動的教學輔助：

本系統的設計不僅關注學生端，也高度重視教師的需求。透過後端資料庫對學生答題紀錄、遊戲進度等數據的儲存和管理，教師能夠獲得客觀、即時的學習成效分析。這種設計哲學旨在將數據轉化為有意義的教學洞察，幫助教師精準識別學生的薄弱環節，調整教學策

略，並進行更有效的差異化教學。這賦予教師更強的教學決策能力，從而提升整體教學效率和品質。

4. 友善且直觀的使用者體驗：

我們深知良好的使用者體驗是數位學習系統成功的基石。因此，在介面設計上，我們採用簡潔直觀的視覺風格，確保操作流程流暢易懂，無論是學生還是教師都能快速上手。從登入、選擇學習模組到遊戲互動，每個環節都力求降低認知負荷，讓使用者能夠專注於學習本身，而不是操作複雜的系統。

總體而言，本研究的設計哲學旨在打破傳統數學教學的束縛，透過融合遊戲化、個性化和數據驅動的設計原則，為學生創造一個引人入勝、高效且富有成就感的學習環境，同時為教師提供強大的教學輔助工具，共同推動數學教育的進步。

5.2 研究成果與貢獻

本研究成功開發了一套融合遊戲元素的國小數學學習網站，實現了以下核心目標，並對數學教育領域做出了具體貢獻：

1. **顯著提升學生學習動機與興趣：** 透過創新性的寵物養成機制、經驗值累積、道具收集、即時回饋以及內建的圖鑑蒐集功能，本系統有效地將枯燥的數學練習轉化為富有樂趣和挑戰性的遊戲體驗。問卷調查結果與質性回饋均顯示，絕大多數學生對學習數學的積極性顯著提高，證明了遊戲化設計在激發學習內在動力方面的巨大潛力。
2. **提供教師高度彈性的教學輔助工具：** 系統後端與 MySQL 資料庫的整合，賦予教師直接在前端介面進行題目新增、編輯、更新與刪除的權限。這大大增強了教學的彈性，教師可以根據學生的即時學習狀況和課程進度，快速調整和客製化題庫內容，實現更精準的「因材施教」。
3. **促進學習的即時回饋與直觀理解：** 教材中的答題功能設計了即時的正確與錯誤反饋，學生能夠立即得知答題結果，並根據回饋進行調整。結合教材學習模組的逐步引導，有助於學生更直觀地理解抽象的數學概念，減少學習過程中的挫敗感。

4. **系統操作便捷且介面友善：** 透過 CSS 的應用，系統介面簡潔且具有良好的使用者體驗。無論是學生進行遊戲互動，還是教師管理題目，操作流程均設計得直觀易懂，降低了使用者學習系統的時間成本。
5. **奠定未來數據分析與個性化學習的基礎：** 雖然當前系統的數據分析功能尚未完全實現，但其資料庫結構已為儲存學習歷程、答題表現等關鍵數據奠定了堅實基礎。這為未來導入 AI 算法，實現更深層次的學習行為分析和個性化學習推薦提供了可能性。

總之，本系統不僅作為一個數學學習平台，更重要的是，它成功地在學生與學習之間建立起了一種積極且愉悅的互動連結。透過實踐證明，將精心設計的遊戲化元素與教學內容相結合，能夠在不犧牲學術嚴謹性的前提下，有效提升學習效能，為未來數位教育的發展提供了寶貴的實踐經驗與應用價值。

5.3 未來研究與發展建議

為使本數學教學網站能夠持續優化並發揮更大的教育潛力，未來研究與發展可從以下幾個方面進行深入探討與實踐：

1. 擴充題型與年級適用性：

- **增加多元題型：** 目前主要以選擇題為主，未來可考慮引入填充題、判斷題、應用題、拖曳題等多元題型，以更全面地評估學生的數學能力。
- **涵蓋更多年級與數學領域：** 將題目和教材內容擴展至國小其他年級，並逐步納入更廣泛的數學領域，如幾何、代數初步、統計與機率、應用問題等，使系統能服務更廣泛的學生群體。

2. 深度導入 AI 技術，實現智能化學習：

- **AI 題庫推薦系統：** 這是未來最重要的發展方向之一。可利用機器學習算法分析學生的歷史答題數據（正確率、答題時間、錯誤類型等），建立個人化的學習模型。基於此模型，系統能智能地推薦適合學生當前知識水平和學習弱

點的題目，實現自適應難度調整，真正做到「因材施教」。

- **AI 即時答疑與輔導：** 整合大型語言模型（LLMs）或特定的數學知識圖譜，開發具備即時答疑能力的 AI 輔導功能。當學生遇到問題時，可以即時獲得解釋、提示或相關概念的連結，模擬真人教師的輔導過程，提高學習效率。
- **學習進度預測與風險預警：** 利用數據分析預測學生在特定知識點或整體學習上的潛在困難，並向教師發出預警，以便教師能及早介入輔導。

3. 完善使用者帳號系統與跨裝置支援：

- **完整帳號與資料儲存：** 目前雖已使用 localStorage 儲存部分狀態，但未來需建立完善的後端使用者帳號管理系統，支援使用者註冊、登入、密碼重置等功能，並將所有學習歷程、寵物養成數據、道具庫存等關鍵資訊儲存在資料庫中。
- **跨裝置進度同步：** 確保使用者無論在電腦、平板或手機上使用系統，其學習進度、遊戲數據都能夠即時同步，提供無縫的學習體驗。
- **個人化儀表板：** 為學生提供更詳細的個人學習報告儀表板，展示其學習進度、強項與弱項、答題趨勢等，幫助學生自我監控和反思學習。

4. 拓展遊戲模式與社群互動：

- **更多寵物與養成元素：** 根據學生的反饋，設計更多種類的寵物、豐富的道具、多元的進化路線和養成事件，持續保持遊戲的新鮮感和吸引力。
- **強化多人競技與合作模式：** 可引入更多即時多人對戰或合作解題模式，增強學生的互動性、協作能力和良性競爭。
- **虛擬學習社區：** 建立一個安全的虛擬社群平台，讓學生可以分享學習心得、討論問題、展示自己的寵物成就，營造積極的學習氛圍。

5. 進行長期學習成效追蹤與實證研究：

- **整合至實際教學課程：** 將本系統作為正式教學工具，與學校合作進行長期、大規模的教學實驗，收集更豐富的量化數據（如前後測成績、學習動機量表）和質性回饋。
- **多校際比較研究：** 擴大研究範圍，與不同地區或學校合作，比較本系統在不同教學環境和學生群體中的學習成效和適用性。
- **效益評估模型：** 建立一套科學的效益評估模型，量化分析本系統對學生學業表現、學習動機、自信心等方面的影響，為教育政策制定提供依據。

透過上述方向的持續努力，本「數網知來」數學教學網站未來有望發展成為一個功能更強大、體驗更優化、影響更深遠的智能學習平台，為推動台灣的數學教育現代化貢獻一份力量。

第六章 相關參考文獻

Davis & Hersh (1981)。The Mathematical Experience

謝佳鈺 (2009)。國小學童邏輯推理能力養成之研究

教育部 (2018)。十二年國民基本教育課程綱要

Bernstein, A. L. (1964)。Motivation in mathematics.

葉盛昌 (2003)。遊戲式數學教學模式對學生數學學習的影響

廖育汝 (2021)。遊戲融入教學對國小數學學習興趣與成效之研究——國小一年級幾何課程

陳俊瑋 (2006)。數位學習系統之設計原則研究-以故宮為例

黃昱翔 (2021)。網頁架構系統開發之研究探討

新竹師院學報(1993)。「遊戲」之定義、理論與發展的文獻探討

陳俊吉 (2013)。策略性人才培育體系探討

林凱胤 (2014)。即時回饋機制對學生學習專注力影響之研究

葉川榮 (2009)。國中階段的學業競爭現象

謝婉瑛 (2018)。運用「均一教育平台」教學對國小五年級學童數學學習態度與學習成就之影響

李依蓉 (2018)。均一教育平台輔助國小三年級數學學習成效之研究——以桃園市某國小為例