國立臺北商業大學

資 訊 管 理 系

114’資訊系統專案設計

**系統手冊**

****

**組 別：第114510組**

**題 目：智能校事專家系統**

**指導老師：蔡文隆老師**

**組 長：11056030 蔡依恬**

**組 員：11056006 林郁軒 11056009 宋繼平**

**11056019 陳宜佳 11056020 許宸熙**

**中華民國114年5月26日**

**目錄**

第1章 前言 6

**1-1 背景介紹** 6

**1-2 動機** 6

**1-3 系統目的與目標** 7

**1-4 預期成果** 8

第2章 營運計畫 10

**2-1 可行性分析** 10

**2-2 商業模式－Business model** 10

**2-3 市場分析－STP** 12

**2-4 競爭力分析SWOT-TOWS** 13

第3章 系統規格 15

**3-1 系統架構** 15

**3-2 系統軟、硬體需求與技術平台** 15

**3-3 開發標準與使用工具** 16

第4章 專案時程與組織分工 18

**4-1 專案時程** 18

**4-2 專案組織與分工** 19

**4-3 上傳GitHub紀錄** 21

第5章 需求模型 22

**5-1 功能分解圖（Functional decomposition diagram）** 22

**5-2 需求清單** 22

**圖目錄**

圖3-1‑1 系統架構圖 15

圖 4-1‑1專案時程甘特圖 18

圖 5-1‑1功能分解圖 22

**表目錄**

表2-4‑1 SWOT分析 13

表2-4‑2 TOWS分析 14

表3-2‑1 軟體需求 15

表3-2‑2 硬體需求 16

表 3-3‑1開發環境與程式編寫工具 16

表 3-3‑2 使用工具 16

表 4-2‑1專案組織與分工 19

表 4-2‑2專題成果工作內容與貢獻度表 21

表 5-2‑1功能性需求 22

# **前言**

## **背景介紹**

隨著生成式人工智慧（Generative AI）與大型語言模型（Large Language Model, LLM）的技術迅速演進，人工智慧應用已從單純的關鍵字搜尋，逐步轉型為具備理解語意與生成能力的對話式系統。其中，檢索增強生成（Retrieval-Augmented Generation, RAG）架構的興起，成為結合資料檢索與語言生成的重要趨勢，廣泛應用於知識問答、客服輔助、教育科技等領域。

在校園環境中，資訊數位化雖已普遍，但資訊的呈現形式與使用者互動方式仍以靜態公告、PDF檔案與分散平台為主，查詢效率與整合程度仍有進步空間。校規條文、課程資料與學生事務等資訊大多散見於不同系統或網頁，缺乏統一搜尋機制，使得學生與教職員在資料查找過程中，經常面臨操作不便與資訊碎片化等問題。

除此之外，校內外各類系統雖各自具備特定功能，例如校園官網、學生資訊系統、Dcard校園討論區等，但彼此間缺乏串接與互通機制，導致使用者在日常操作上需要多次登入、切換平台，降低整體體驗的一致性與便利性。隨著人工智慧能力的提升，導入智慧化查詢與資訊整合技術，已成為未來校園資訊服務發展的重要方向。

## **動機**

在實際校園生活中，學生經常需要查詢像是畢業條件、選課限制、學分要求等資訊，卻常因資訊分散或格式不友善而花費大量時間翻閱PDF檔、公告或向他人求助。這樣的查詢過程不僅繁瑣，還容易出現資訊錯誤或混淆不同版本內容，增加理解與判斷成本。而在選課時，也缺乏一個可信、整理過的評論平台，大部分的意見都過於情緒化及主觀，讓學生無法做出合適的選擇。

平常也有很多瑣事缺乏一個統一的平台來管理，像是想確認自己已修多少學分、畢業門檻還缺什麼時，都需要反覆翻閱資料並進行人工比對，既繁瑣又容易出錯。有時想找人一起修課、參加活動，也只能在群組中四處詢問，一旦訊息刷過就很難再找到。至於課本買賣等需求，也得跑去不同平台貼文或私訊。

這些問題讓學習與生活充滿不必要的困擾。因此，我們希望能做出一個真正站在學生立場的系統，透過AI技術，幫助學生快速且準確地查詢所需資訊，減少查找時間與理解成本。同時，我們也將個人中心、課程交流、活動揪團及二手書交換等功能整合於同一平台中，希望能打造一個方便、直覺的智慧校園系統，讓學生能更輕鬆地掌握資訊、參與互動，進而提升校園中的學習與生活體驗。

## **系統目的與目標**

本專題旨在開發一套以本校為起點，結合AI技術應用的智慧校園平台。具體目標如下：

1. **智慧法規查詢**

建構以RAG為核心的智能問答系統，將校內法規與重要公告轉換為向量（Embeddings），並使用向量資料庫進行相似度檢索，結合大型語言模型（LLM）進行自然語言回答，提供快速且精準的法規查詢服務。

1. **AI優化課程評論**

建立課程評論區，透過AI技術，對學生所提供的課程評論進行情緒分析與語意重寫，過濾情緒化或偏頗的內容，將其轉換為更具建設性、客觀與理性的信息。幫助未來選課的學生能夠更容易地從中獲取實用、清晰的課程資訊。

1. **個人中心**

提供每位使用者專屬的儀表板，整合課程提醒、學分統計、近期待辦等資訊，提升學習效率與自我管理能力。

1. **揪團交流功能**

提供揪團系統，協助學生尋找共同參與的活動、課程或學習主題，促進校內社群互動與合作。

1. **二手書交易平台**

建立簡單易用的二手書刊登與搜尋系統，讓學生可便利地進行教材資源交換，提升書籍再利用率與使用價值。

1. **系統可拓展性**

設計具備彈性的系統架構，未來可擴展至其他學校或新增模組，例如考試資源分享、課程配對推薦等，逐步建構完整的智慧校園平台。

## **預期成果**

本專題預期達成以下具體成果：

* 成功實作基於RAG架構之智慧問答系統，能即時回應學生對校內法規、學程設置、選課規則等問題。
* 完成向量資料庫建置，並將校內規章轉換為向量資料以提升檢索效率與語意匹配準確度。
* 開發具備AI優化功能之課程評論平台，提升課程評價資料的建設性與參考價值。
* 建立個人中心、揪團與二手書交換等模組，強化學習生活整合與使用者互動體驗。
* 打造介面友善、功能完善的智慧校園平台，為未來商品化或跨校推廣打下基礎。
* 展現AI技術於校園應用領域的可行性與創新潛力，提升本專題的技術深度與實務價值。

# **營運計畫**

## **可行性分析**

**技術可行性**

本系統為一套專為大專校院打造的智慧校園整合平台，整合智慧法規查詢、課程評論區、學習進度追蹤、活動揪團及二手書交換等功能。技術上採用前後端分離架構、關聯式資料庫與AI技術以實現即時互動與資料精準查詢，目前已完成主要模組的初步開發與測試，具備實際可行性。

**經濟可行性**

本系統以學生團隊開發為主，利用免費工具、開源資源與低成本雲端服務，有效控制開發與維運成本。除AI API可能產生部分費用外，其餘大多為一次性投入，初期預算可控制在新台幣10,000元以下，具高度經濟效益。

**實務可行性**

系統設計基於本校學生實際需求，包括學分查詢不便、課程資訊分散、缺乏集中討論平台等問題。功能規劃與設計經實際使用者（學生、老師）訪談回饋修正，具備落地實施可能性。若導入成功，未來亦可依架構擴充應用至其他學校。

## **商業模式－Business model**

**目標客群**

本校大專學生、教職員，未來可拓展至其他學校

**價值主張**

整合校園常用資訊與AI查詢服務，簡化查詢流程、提升校內互動與學習效率

**通路**

透過校內官方網站入口、課務系統導入、社群媒體宣傳、校園海報與說明會等方式進行推廣。

**顧客關係**

提供個人帳號登入機制、常見問題教學、使用回饋系統與持續更新維護，提升使用體驗與忠誠度。

**收益來源**

* 初期以非營利模式推行，供本校免費使用以累積使用經驗與回饋。
* 後期可向學校端（如教務處或資訊中心）收取技術維運費或功能授權費，作為平台持續營運與更新的資金來源。
* 同時，可推出進階功能（如個人化選課建議、學習分析報告）並採訂閱制供學生自主選購。
* 若拓展至其他學校，則可採用B2B 技術授權模式，依據學校規模與功能模組收費。

**關鍵資源**

AI查詢模組、語意優化引擎、學分與課程資料庫、課程論壇系統、開發與維運人力。

**關鍵活動**

系統開發與測試、資料整合與優化、推廣與教育訓練、平台維護與版本更新。

**關鍵夥伴**

校內教務單位、資訊中心、學生會等，未來可與教育科技公司合作

**成本結構**

系統開發與維運費用、伺服器與儲存空間、API與第三方工具使用費、人工智慧訓練與人力支出。

## **市場分析－STP**

**Segmentation（市場區隔）**

校內不同系所學生、教學與行政單位使用者、具導入潛力的其他大專校院

**Targeting（目標市場）**

初期將專注於校內學生群體，進行推廣並收集使用者回饋，從中建立穩定的核心用戶群並收集真實的使用數據，為系統優化與後續擴展奠定基礎。後期則會拓展至全校的教職員，並進一步推向其他大專院校市場。

**Positioning（市場定位）**

本系統定位為專為校園生活設計的智慧整合平台，強調資訊即時查詢、跨模組整合與互動性，是傳統教務系統與通訊公告工具的升級版本，強化學習效率與校園資訊透明度。

## **競爭力分析SWOT-TOWS**

表2-4‑1 SWOT分析

|  |  |
| --- | --- |
| **優勢 Strengths** | **劣勢 Weaknesses** |
| * 結合 RAG 與 LLM 技術，查詢回應快速且準確 * AI優化評論幫助學生做出適合的選課決定 * 整合度高、功能實用 * 介面設計以學生為核心，操作簡便 * 架構具彈性，未來可擴充與跨校應用 | * 技術與人力有限，開發與維護壓力大 * 初期法規資料處理需花大量時間 * 課程評論仰賴用戶累積，初期內容少，參考性有限 * 多模組整合增加系統複雜度 |
| **機會 Opportunities** | **威脅 Threats** |
| * 生成式 AI 應用熱潮持續上升，校園端也逐漸重視數位轉型 * 學生對一站式整合平台需求高，有利於快速擴大使用族群 * 校內行政單位具合作潛力，利於導入與推廣 * 有潛力拓展至其他學校或轉為商業化產品，市場潛力大 | * 他校或外部平台可能推出相似產品，形成競爭 * 若回答錯誤或評論失準，易引發信任問題與負評 * 學生已習慣使用 Dcard、LINE 等平台，轉移成本高 * 校規內容經常更新，若維護不及時將導致資訊過時 |

表2-4‑2 TOWS分析

|  |  |
| --- | --- |
| **S-O 優勢/機會** | **W-O 劣勢/機會** |
| * 發揮技術整合優勢，搭上教育數位轉型風潮，快速推廣至校園內外 * 強化系統特色與UI體驗，透過 RAG 與評論優化功能，創造有別於傳統查詢系統的價值 * 採模組化設計，利於後續導入其他學校或教育單位 | * 導入開源工具以降低技術門檻與成本 * 善用學生回饋優化使用體驗，逐步擴大資料規模與功能 |
| **S-T 優勢/威脅** | **W-T 劣勢/威脅** |
| * 與教務處等單位合作，確保資料正確性與即時更新 * 建立使用者回饋與錯誤回報機制，提升系統透明度與信任感 * 持續AI優化處理技術，讓課程評論更具參考價值 | * 初期以核心功能為主，漸進式推廣，降低開發與維運負擔 * 建立定期資料審查與更新流程，減少內容過時問題 * 以學生使用習慣為導向，加入提醒、互動通知等功能提升黏著度 |

# **系統規格**

## **系統架構**

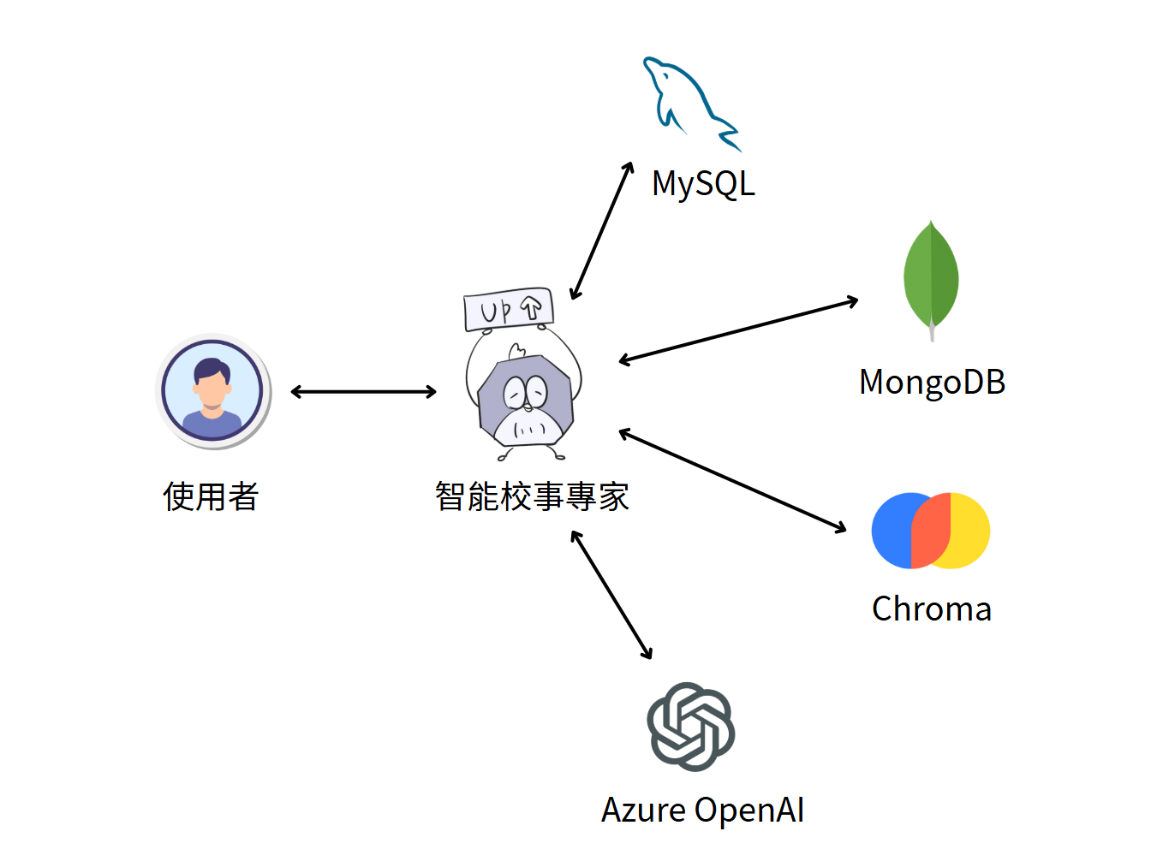


圖3-1‑1 系統架構圖

## **系統軟、硬體需求與技術平台**

本系統為 Web 應用平台，使用者僅需透過瀏覽器即可使用，無須額外安裝任何軟體。以下列出一般使用者所需的軟、硬體環境建議：

表3-2‑1 軟體需求

|  |  |
| --- | --- |
| 軟體需求 | |
| 作業系統 | Windows 10以上、macOS、Android、iOS |
| 瀏覽器 | Google Chrome（建議）、Microsoft Edge、Firefox、Safari |
| 網路環境 | 穩定的網際網路連線（建議 Wi-Fi 或 4G 以上） |
| 螢幕解析度 | 1280×720（HD）以上，支援手機與平板裝置自適應顯示 |

表3-2‑2 硬體需求

|  |  |
| --- | --- |
| 硬體需求 | |
| 裝置類型 | 桌上型電腦 / 筆記型電腦 / 智慧型手機 / 平板 |
| 處理器 | 雙核心 1.5GHz 以上（建議四核心以上） |
| 記憶體 | 4GB RAM 以上 |
| 顯示設備 | 一般彩色顯示器或行動裝置螢幕 |
| 輸入裝置 | 滑鼠、鍵盤、觸控螢幕（視裝置而定） |

## **開發標準與使用工具**

表 3-3‑1開發環境與程式編寫工具

|  |  |
| --- | --- |
| 開發環境與程式編寫工具 | |
| 系統開發環境 | Windows 10/11 |
| 程式編寫工具 | Visual Studio Code |

表 3-3‑2 使用工具

|  |  |
| --- | --- |
| 前端 | |
| 開發語言 | HTML5、CSS3、JavaScript |
| 視覺設計工具 | Figma |
| 後端 | |
| 開發語言 | Python |
| 框架 | Django |
| 智慧法規查詢工具 | |
| 文件處理工具 | PyPDFLoader |
| 文本分割工具 | RecursiveCharacterTextSplitter |
| Embedding 技術 | Azure OpenAI Embeddings |
| 語言模型 API | Azure ChatGPT（Azure OpenAI） |
| 問答架構 | RAG（Retrieval-Augmented Generation） |
| 整合工具框架 | LangChain |
| 資料庫 | |
| 關聯式資料庫 | MySQL |
| 非關聯式資料庫 | MongoDB（用於儲存對話紀錄） |
| 向量資料庫 | Chroma（儲存嵌入式向量） |
| 管理工具 | MySQL Workbench、MongoDB Compass |
| 版本控制工具 | |
| 版本控制 | Git、GitHub |
| 應用程式 | Fork |
| 其他工具 | |
| 文件編輯 | Microsoft Word、Excel |
| 封面與簡報設計 | Canva、PowerPoint |

# **專案時程與組織分工**

## **專案時程**



圖 4-1‑1專案時程甘特圖

## **專案組織與分工**

表 4-2‑1專案組織與分工

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 項目/組員 | | 11056030/  蔡依恬 | 11056006/  林郁軒 | 11056009/  宋繼平 | 11056019/  陳宜佳 | 11056020/  許宸熙 |
| 後端開發 | 資料庫建置 |  | ● |  |  |  |
| 伺服器架設 | ● | 〇 |  |  |  |
| 首頁 | ● |  |  |  |  |
| 登入 |  | ● |  |  |  |
| RAG校規查詢 | ● |  |  |  |  |
| AI優化課程評論 |  | ● |  |  |  |
| 揪團 | ● | 〇 |  |  |  |
| 二手書交換平台 | 〇 | ● |  |  |  |
| 個人中心 | 〇 | ● |  |  |  |
| 後端統整 | ● |  |  |  |  |
| 前端開發 | 首頁 |  |  | 〇 |  | ● |
| 登入 |  |  | ● |  |  |
| RAG校規查詢 |  |  |  | ● |  |
| AI優化課程評論 |  |  | ● |  |  |
| 揪團 |  |  |  | ● |  |
| 二手書交換平台 |  |  |  | ● |  |
| 個人中心 |  |  |  |  | ● |
| 前端統整 |  |  | ● |  |  |
| 美術設計 | UI/ UX |  |  |  |  | ● |
| Web/APP介面設計 |  |  | ● | 〇 | 〇 |
| 色彩設計 |  |  | 〇 |  | ● |
| Logo設計 |  |  | 〇 |  | ● |
| 素材設計 |  |  | ● | 〇 | 〇 |
| 首頁 |  |  | 〇 | 〇 | ● |
| 登入 |  |  | ● |  |  |
| RAG校規查詢 |  |  |  | ● |  |
| AI優化課程評論 |  |  | ● |  |  |
| 揪團 |  |  |  | ● |  |
| 二手書交換平台 |  |  |  | ● |  |
| 個人中心 |  |  |  | ● |  |
| 文件撰寫 | 統整 | ● |  |  |  |  |
| 第1章 前言 | ● |  |  |  |  |
| 第2章 營運計畫 | ● |  |  |  |  |
| 第3章 系統規格 | ● |  |  |  |  |
| 第4章 專題時程與組織分工 | ● |  | 〇 |  |  |
| 第5章 需求模型 |  |  |  | ● |  |
| 第6章 程序模型 |  |  |  |  | ● |
| 第7章 資料模型 |  |  |  |  |  |
| 第8章 資料庫設計 |  | ● |  |  |  |
| 第9章 程式 |  |  |  |  |  |
| 第10章 測試模型 |  |  |  |  |  |
| 第11章 操作手冊 |  |  |  |  |  |
| 第12章 使用手冊 |  |  |  |  |  |
| 報告 | 簡報製作 |  |  |  |  |  |

表 4-2‑2專題成果工作內容與貢獻度表

|  |  |  |  |
| --- | --- | --- | --- |
| 序號 | 姓名 | 工作內容<各限100字以內> | 貢獻度 |
| 1 | 組長  蔡依恬 |  | 20 % |
| 2 | 組員  林郁軒 |  | 20 % |
| 3 | 組員  宋繼平 |  | 20 % |
| 4 | 組員  陳宜佳 |  | 20 % |
| 5 | 組員  許宸熙 |  | 20 % |
|  | | | 總計:100% |

## **上傳GitHub紀錄**

內文撰寫…

# **需求模型**

## **功能分解圖（Functional decomposition diagram）**

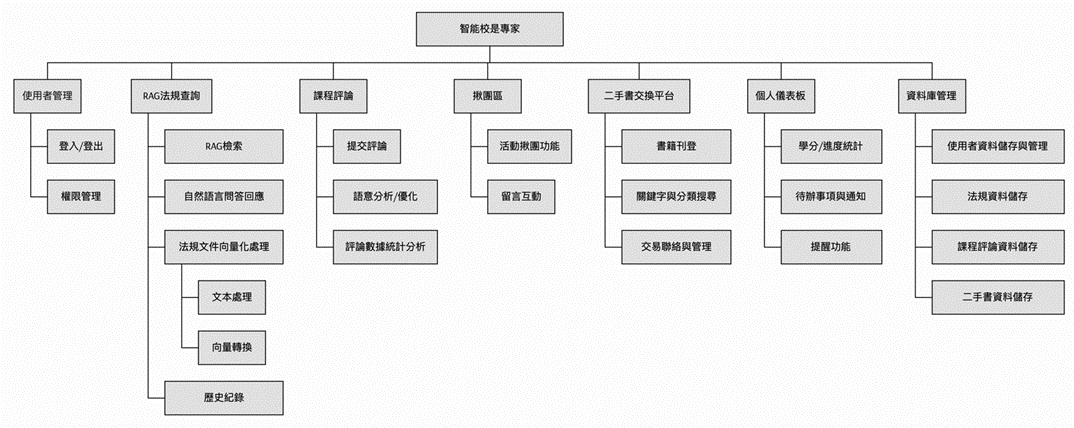


圖 5-1‑1功能分解圖

## **需求清單**

**功能需求**

表 5-2‑1功能性需求

|  |  |
| --- | --- |
| **使用者登入功能** | 1. 使用者可透過學號與密碼登入系統。 2. 系統需進行帳號密碼驗證，正確者方可登入。 3. 若登入失敗（例如密碼錯誤），需顯示錯誤訊息提示。 |
| **RAG法規查詢** | 1. 使用者可通過關鍵字或語句查詢校內法規。 2. 系統使用RAG（Retrieval-Augmented Generation）技術將文檔轉換為向量，並使用向量資料庫進行相似度檢索。 3. 結合大型語言模型（LLM），生成精準的自然語言回答。 |
| **AI優化課程評論平台** | 1. 提供課程評論區，並引入AI語意分析技術，檢測評論中的情緒化或偏頗內容。 2. 自動將不具建設性的評論進行優化，提升評論的參考價值與可信度。 |
| **揪團交流功能** | 1. 提供一個活動或課程揪團系統，協助學生尋找共同參與的學習或課外活動。 2. 促進校內社群的互動與合作，加強學生之間的交流。 |
| **二手書交換平台** | 1. 設計一個二手書的交換平台，讓學生可以便捷地上傳和搜尋教材。 2. 提高教材資源的再利用率，並提升書籍的使用價值。 |
| **個人儀表板模組** | 1. 為每位使用者提供專屬的儀表板，顯示課程提醒、學分統計、待辦事項等資訊。 2. 提升使用者的學習效率與自我管理能力，讓學生可以輕鬆掌握自己的學習進度。 |

**非功能需求**

1. **系統性能**

* 系統應能夠處理大量用戶並發請求，保證系統穩定性與響應速度。
* 在高負載情況下，依然保持順暢的使用體驗，並提供可靠的數據檢索與處理能力。

1. **可維護性**

* 系統應具備良好的代碼結構與文檔，便於開發團隊後續維護與擴展。
* 定期進行代碼重構和測試，保證系統的可維護性與穩定性。

1. **可用性**

* 系統應提供簡單直觀的用戶介面，保證用戶無需過多學習即可輕鬆使用。
* 提供詳細的幫助文檔與在線支持，確保用戶問題能夠及時解決。

1. **兼容性**

* 系統應該兼容各種常見的瀏覽器（如Chrome、Firefox、Safari等）。
* 移動端與桌面端應具備良好的適配，並支持多設備使用。

1. **系統可擴展性**

* 設計系統架構時應考慮未來的功能擴展，保證新模塊或功能能夠無縫集成。
* 系統應支持在多個學校間擴展，並能夠根據不同需求進行定制。