

私立中原大學
資訊管理學系 專題計畫書

紅樓慧讀：
紅樓夢整本書閱讀 AI 學習系統

組員：

11144209 資管三乙 潘驄杰

11144224 資管三乙 蕭帛昊

11144155 資管三甲 鍾子傑

目錄

壹、緒論.....	3
一、專案動機.....	3
二、目標與功能.....	3
三、系統特色.....	3
貳、需求與市場分析.....	3
一、概述：NABCD 需求與市場分析模型.....	3
二、需求分析.....	3
三、市場競品分析.....	3
四、商業模式.....	3
五、推廣模式.....	3
參、系統分析與設計.....	4
一、軟體組織架構流程圖綜觀.....	4
二、UML 系統架構圖.....	4
三、DFD 資料流程圖.....	4
四、資料庫設計.....	4
五、使用者介面設計.....	4
六、軟體元件設計與技術棧.....	4
七、API 接口規範.....	4
八、測試規範.....	4
九、例外處理.....	8
十、時程規劃.....	8
肆、成本估計.....	10
伍、專案組織.....	10
陸、未來展望與結論.....	10
柒、參考文獻.....	11

壹、緒論

一、專案動機

二、目標與功能

三、系統特色

貳、需求與市場分析

一、概述：NABCD 需求與市場分析模型

二、需求分析

三、市場競品分析

四、商業模式

五、推廣模式

參、系統分析與設計

一、軟體組織架構流程圖綜觀

二、UML 系統架構圖

三、DFD 資料流程圖

四、資料庫設計

五、使用者介面設計

六、軟體元件設計與技術棧

七、API 接口規範

八、測試規範

3. 檔案結構

高山文津/

|

|—— frontend/

前端應用

| |—— web/

Web 應用

```
    |   |   └── components/          # Vue 元件
    |   |       └── QuestionInput/    # 問題輸入組件
    |   |       └── AnswerDisplay/    # 答案顯示組件
    |   |       └── ImageUpload/      # 圖片上傳組件
    |   |       └── LearningProgress/ # 學習進度組件
    |   └── views/                  # 頁面視圖
    |   └── store/                  # 狀態管理
    |   └── assets/                 # 靜態資源
    |
    └── mobile/                   # 移動應用
        ├── lib/                   # Flutter 代碼
        |   ├── screens/            # 移動端頁面
        |   ├── widgets/            # 自定義組件
        |   ├── services/           # API 服務
        |   └── utils/              # 工具函數
        └── assets/                 # 應用資源
    |
    └── backend/                  # 後端服務
        ├── api/                  # API 定義
        |   ├── routers/            # API 路由
        |   ├── schemas/            # 請求/響應模型
        |   └── middleware/         # API 中間件
    |
    └── core/                     # 核心邏輯
        ├── reasoning/           # 推理邏輯
        |   ├── agent.py           # 推理代理
        |   ├── pipeline.py         # 推理管道
        |   ├── cot.py              # 思維鏈推理
        |   ├── models.py           # 推理模型定義
        |   ├── parallel.py         # 並行處理
        |   ├── optimizer.py         # 查詢優化
        |   ├── visualization.py    # 推理可視化
        |   └── tests/               # 推理測試
```

```
|   |   |
|   |   └── retrieval          # 檢索邏輯
|   |       ├── engine.py      # 檢索引擎
|   |       ├── preprocessor.py # 查詢預處理
|   |       ├── vector_search.py # 向量檢索
|   |       ├── hybrid_search.py # 混合檢索
|   |       └── knowledge_graph.py # 知識圖譜檢索
|   |       └── reranker.py     # 結果重排序
|
|   |   └── rag/               # RAG 整合
|   |       ├── rag_agent.py    # RAG 代理
|   |       └── controller.py    # RAG 控制器
|   |       └── adaptive_concurrency.py # 自適應並發
|
|   └── services/            # 業務服務
|       ├── user_service.py    # 用戶服務
|       ├── ocr_service.py     # OCR 服務
|       ├── learning_service.py # 學習記錄服務
|       └── feedback_service.py # 反饋收集服務
|
|   └── models/              # 資料模型
|       ├── user.py           # 用戶模型
|       ├── question.py        # 問題模型
|       ├── learning_record.py # 學習記錄模型
|       └── feedback.py        # 反饋模型
|
|   └── utils/                # 工具函數
|       ├── text_processor.py  # 文本處理
|       ├── config_manager.py  # 配置管理
|       └── error_handler.py   # 錯誤處理
|
|   └── data/                 # 數據相關
|       └── knowledge_base/    # 知識庫資料
```

```
|   |   └── ancient_texts/          # 古代文獻
|   |   └── annotations/           # 文本註釋
|   |   └── dictionaries/          # 古漢語詞典
|
|   └── vector_store/             # 向量存儲
|       └── embeddings/           # 文本嵌入
|           └── indexes/          # 檢索索引
|
|   └── knowledge_graph/          # 知識圖譜
|       └── entities/            # 實體數據
|           └── relationships/    # 關係數據
|
|   └── scripts/                 # 資料處理腳本
|       ├── data_import.py        # 數據導入
|       ├── vectorization.py     # 向量化處理
|       └── graph_builder.py      # 圖譜構建
|
|   └── logs/                     # 日誌
|       ├── api/                  # API 日誌
|       ├── reasoning/            # 推理日誌
|       └── errors/                # 錯誤日誌
|
|   └── tests/                    # 測試
|       ├── unit/                 # 單元測試
|           ├── reasoning/         # 推理測試
|           ├── retrieval/          # 檢索測試
|           └── api/                  # API 測試
|
|       └── integration/          # 整合測試
|           ├── rag_pipeline/       # RAG 管道測試
|           └── user_flow/          # 用戶流程測試
|
|   └── e2e/                      # 端到端測試
```

```
|   └── web_interface/          # Web 界面測試
|   └── mobile_interface/       # 移動界面測試
|
|   └── docs/                  # 文檔
|       └── api/              # API 文檔
|       └── user/             # 用戶指南
|       └── dev/              # 開發者文檔
|
└── config/                 # 配置文件
    └── dev/                  # 開發環境配置
    └── prod/                # 生產環境配置
    └── settings.py          # 全局設置
```

九、例外處理

十、時程規劃

肆、成本估計

伍、專案組織

陸、未來展望與結論

柒、參考文獻

- [Revenue of the Education Apps segment in China from 2023 to 2027](#)
- [Challenges and strategies in teaching classical Chinese to high school students](#)
- [Retrieval-augmented language models for open-domain question answering](#)
- [Global Digital Education Content Market 2025-2029](#)
- <https://www.metaedu.org.tw/zh-hant/2022-taiwan-edtech-report/>