



CasDoc - 基於生成式 AI 代理人技術的可追溯性軟體文件協作系統

指導教授：馬尚彬 教授
組員：劉元楷、林彥丞、蘇東毅



https://casdoc.io

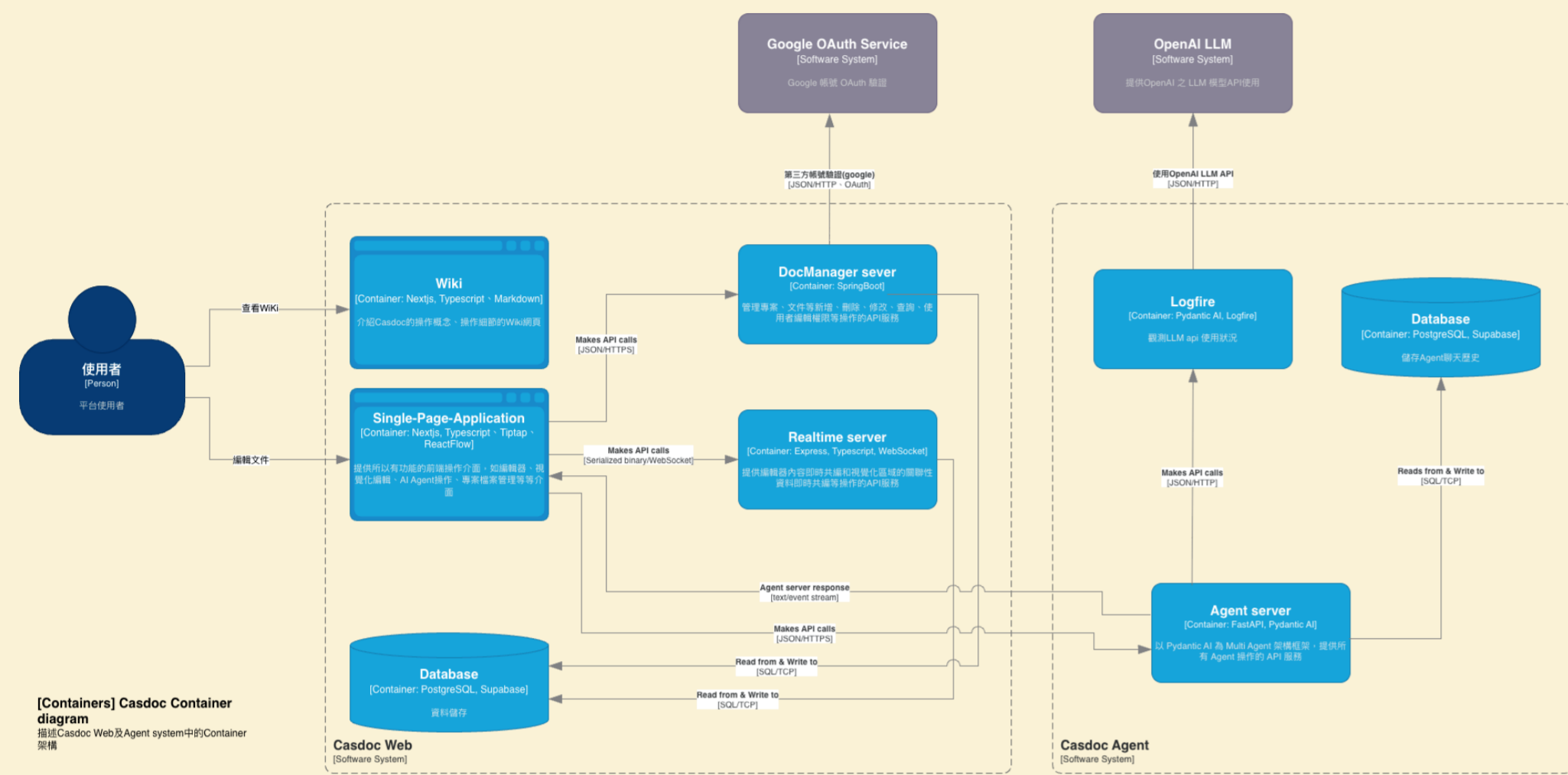


介紹影片

系統簡介

CasDoc 是一套基於 AI 代理人技術的智慧型文件管理平台，用於解決軟體開發中需求、設計與測試文件之間缺乏可追溯性與版本同步等問題。系統整合大型語言模型 (LLM)、RAG 技術與 Multi-Agent 協作框架，可自動生成並維護高品質的軟體文件，確保內容一致、可追蹤且具備可驗證性，大幅降低文件撰寫成本並提升團隊協作效率

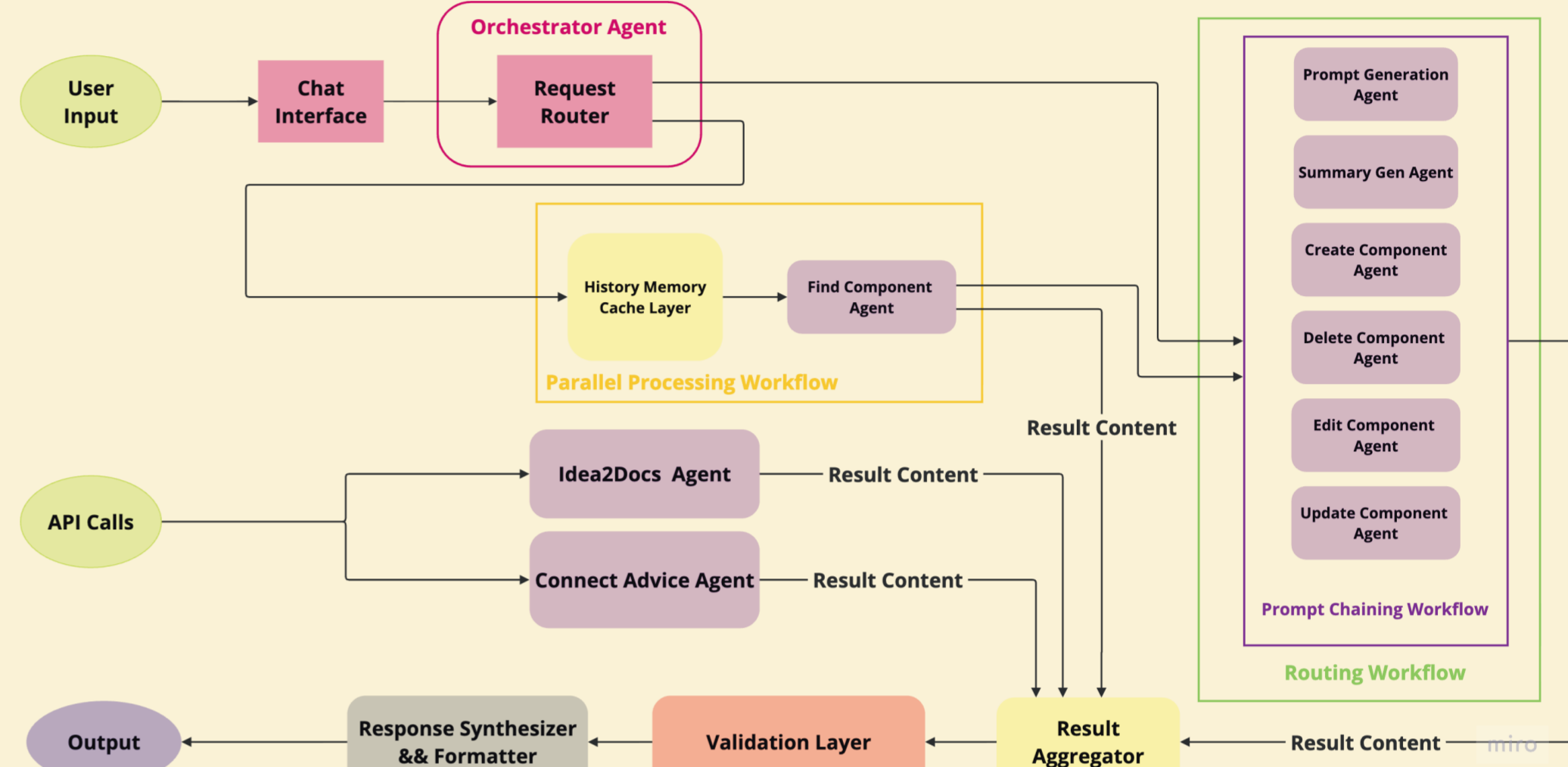
系統架構



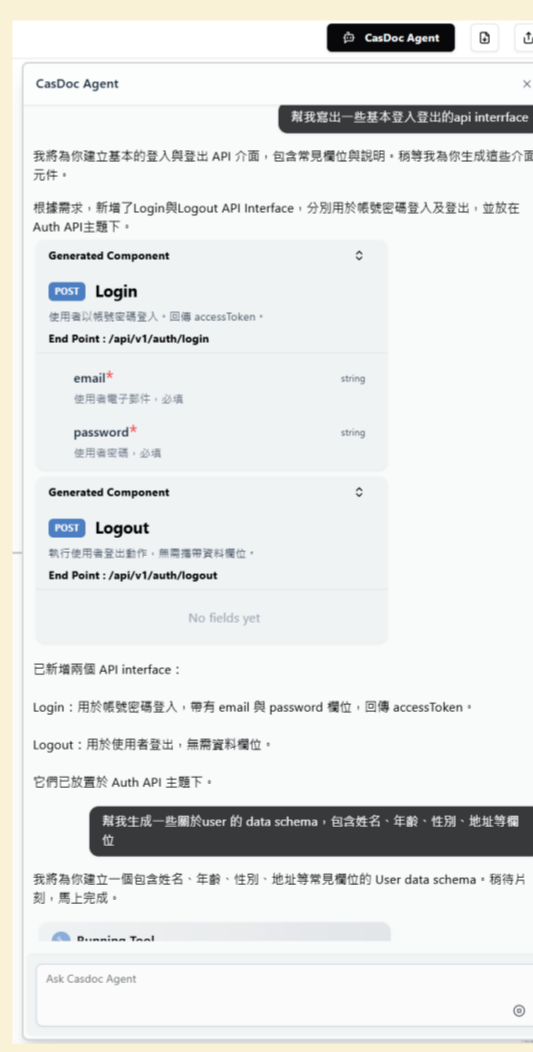
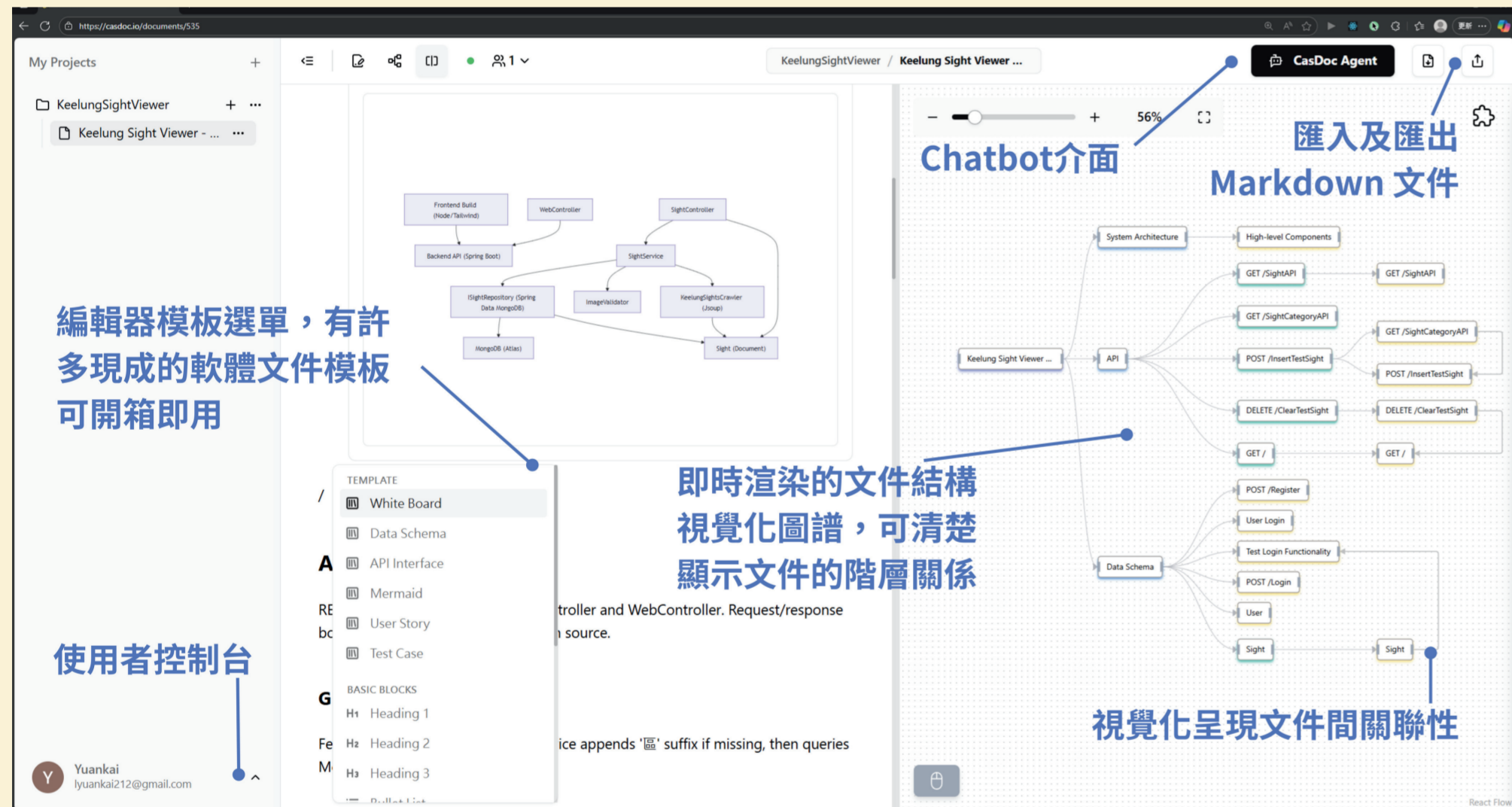
研究動機

軟體文件雖是開發流程的核心，但在實務上常面臨撰寫成本高昂且品質不一的挑戰，過度依賴人工維護容易導致內容結構不完整或難以更新。此外，現有文件往往散落於不同工具平台，缺乏統一管理機制，造成可追溯性不足與版本控制困難。更關鍵的是缺乏自動化同步機制，當需求變更時，相關的設計與測試文件無法即時連動更新，導致文件與系統不一致，大幅增加了開發團隊的後續溝通與維護成本

AI Agent Workflow



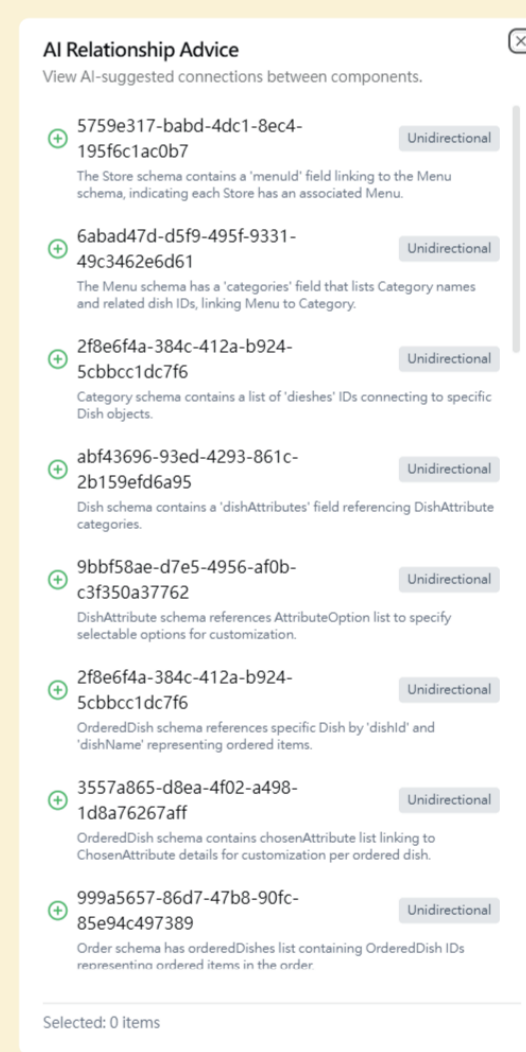
系統功能展示



AI Chatbot

自動化生成：Agent 能讀取 GitHub 程式碼或根據自然語言，自動生成對應的軟體文件草稿

智慧協作：使用者可透過與 Agent 聊天互動，要求其修改文件內容、生成程式碼片段，或彙整專案資訊，充當開發者的智慧助理



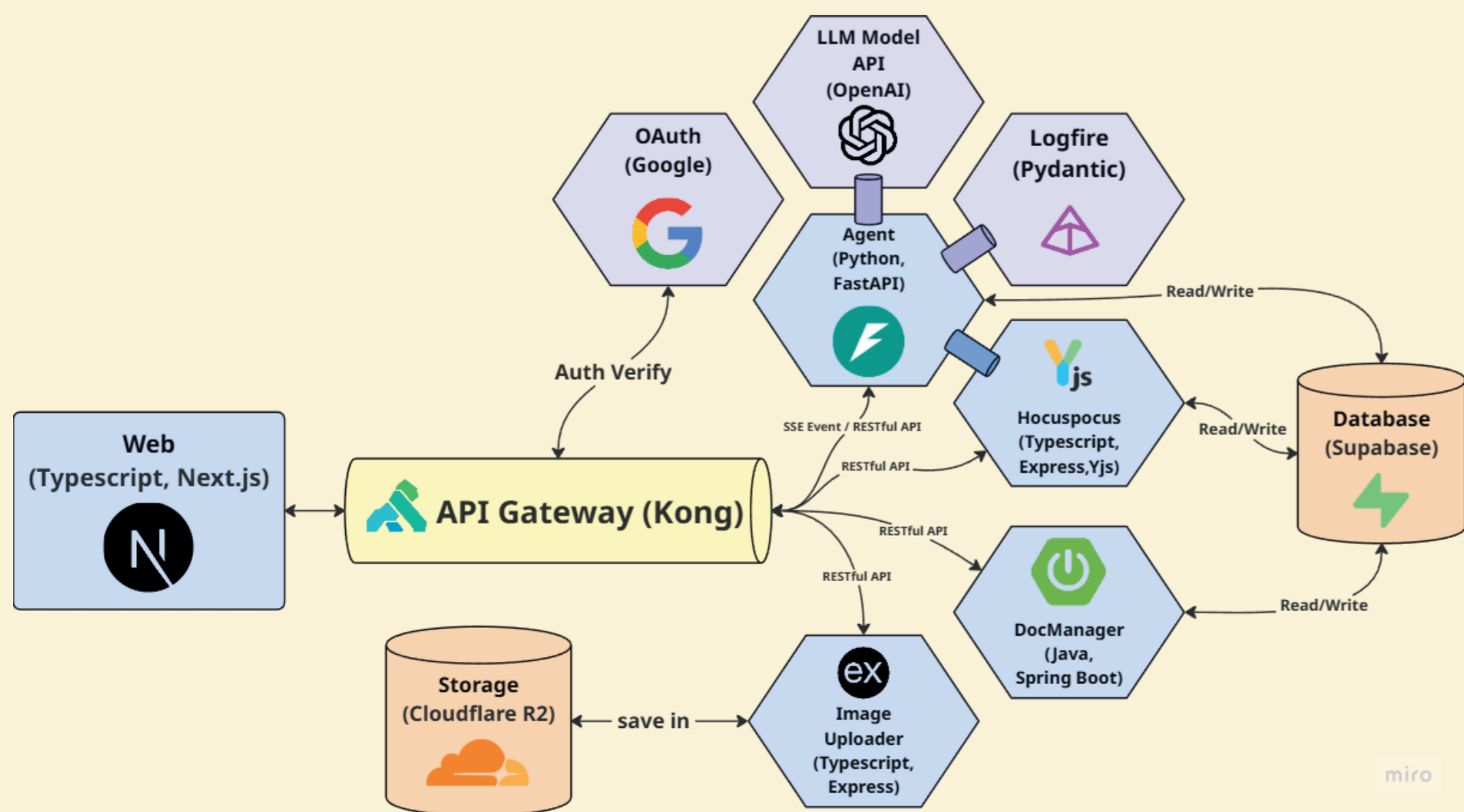
AI 關聯性建議

Auto Relate：能自動分析不同文件的語意關聯，給出建立連結之建議與原因

變更影響分析：當某份文件內容變更時，系統會根據以建立的關聯性自動計算並標示受影響的文件內容，協助開發者快速掌握維護範圍與文件國時風險

開發流程與實作框架

本研究採用 Scrum 敏捷開發模式，運用 GitHub Projects 進行專案管理，並透過用戶訪談與後台數據回饋持續迭代最小可行產品 (MVP)。系統設計採微服務 (Microservices) 結合 Clean Architecture 分層架構；技術框架上，前端選用 Next.js，後端整合 Express、FastAPI 與 Spring Boot 以發揮各語言優勢。開發流程落實版本控制，並藉由 GitHub Actions 與 ArgoCD 實現 CI/CD 自動化部署至 GCP，同時導入 Grafana、Prometheus 與 Posthog 構建完整的服務可觀測性



結論與未來規劃

實證結果顯示，CasDoc 可有效作為敏捷團隊文件與追溯管理支援工具，能在不顯著增加開發者負擔的情況下，讓團隊成員更容易地尋找相關文件與進行關聯性設定，提升變更影響分析與關聯查找的效率。

未來本研究將往三個方向推進：

- 依循成熟度模型的概念擴充量測指標，建立與追溯性直接對應的定量觀察
- 於不同規模與產業場域進行實驗與評估，以提升外在效度
- 和 GitHub 深度整合，落實文件和程式碼同步的工作流

獲獎紀錄

- 114 學年度國科會大專生計畫 通過
- 2025 大專校院資訊應用服務創新競賽 鈦坦敏捷組 金獎
- 2025 資訊應用服務創新創業新秀選拔 潛力產品組 銀牌
- 2025 TCSE 軟體工程競賽 銀獎
- 2025 國科會 FITI 入圍初選
- 2025 數發部 GenAI Stars 入圍複審
- 2025 智慧創新大賞 決賽入圍



產品操作展示