第三章 系統規格

3-1 系統架構

* 本系統採用分層式架構，劃分為前端應用層、後端邏輯層與資料處理層，各層職責明確，利於維護與擴充：
* 前端應用層：使用 Flutter 開發，支援 Android 與 iOS 雙平台。主要負責提供使用者與系統互動的界面，包含情緒紀錄、圖像操作、任務提醒等功能，設計上強調視覺引導與簡易操作。
* 後端邏輯層：採用 Django 框架進行使用者驗證、情緒紀錄管理與日記資料處理，並透過 API 與前端進行溝通。
* 資料儲存與分析層：資料庫使用 MySQL，儲存使用者資料、日記紀錄、情緒分析結果與成就系統資訊。情緒分析模組支援透過關鍵詞辨識情緒傾向，未來可擴充至語意分析與風險評估功能。
* AI 模組設計考量：為避免個資外洩，本系統預設採用匿名化或本地模型執行策略，必要時可連結雲端模型進行回饋生成。

架構圖詳如 3-1-1 所示。

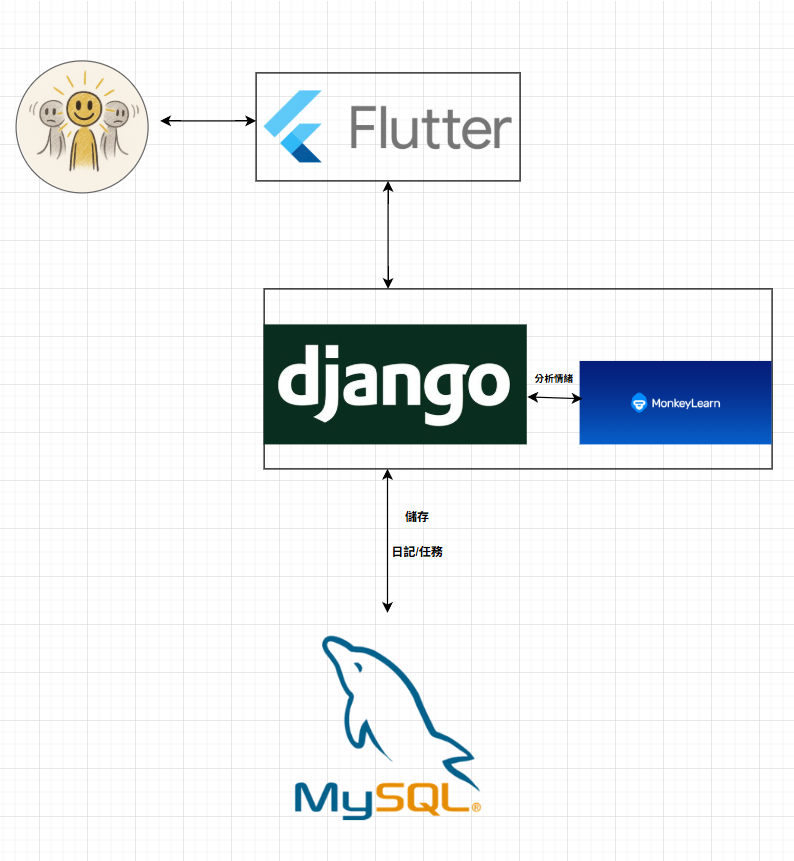


圖 3-1-1、系統架構

* 3-2 系統軟、硬體需求與技術平台

|  |  |
| --- | --- |
| 軟、硬體需求 | |
| 作業系統 | Windows 10 / macOS |
| 裝置建議 | 記憶體 4GB 以上，  螢幕寬 5 吋以上 |
| 網路需求 | 穩定網路連線 |

|  |  |
| --- | --- |
| 系統開發環境 | |
| 作業系統 | Windows 10 / macOS |
| 資料庫 | MySQL |
| 程式開發環境 | Visual Studio Code |
| 前端開發語言 | |  | | --- | |  |  |  | | --- | | Flutter 3.x + Dart | |
| 後端開發語言 | Django 4.x + Python 3.11 |
| 專案管理平台 | |
| 專案管理 | Git + GitHub |
| 版本控制 | GitHub |
| 設計工具 | |
| 介面設計 | Figma |
| 圖示與簡報 | Draw.io、Canva、PPT |