

STAR CLOUD

2026 年 4 月 工作報告

從基礎建設到業務邏輯深耕

報告人：技術應用部 許家偉



開發產能與量化成果

Star Cloud | 月度工作報告

30+

Git Commits

完成超過 30 次以上的程式碼提交
與版本迭代

15+

模組功能實作

涵蓋機台、倉儲、取貨碼等核心功能

100%

設計規範導入

全面落實 Minimal Luxury 極簡奢華風設計標準

B013

異常自動化處理

實作獨立 Job 處理機台異常，降低人工排除時間

< 1s

遠端指令延遲

導入 MQTT 取代輪詢，指令下發延遲降至 1 秒內

High

高併發吞吐量

透過 Redis Queue 解耦，保障 Laravel 資料庫安全



4 月核心開發概覽

Star Cloud | 月度工作報告

承接 3 月的基礎架構，4 月份專注於機台分佈與補貨機制、銷售紀錄與取貨碼等核心模組的深度開發與架構優化，以及關鍵的 MQTT 架構導入。



機台分佈與補貨機制

機台地圖展示與一鍵遠端補貨



商品管理與庫存

商品管理實作與動態庫存矩陣



銷售紀錄與取貨碼

精確追蹤交易歷程與取貨編輯



機台分佈地圖優化

- 強化行動端 (RWD) 瀏覽體驗，提升外勤人員查詢效率

遠端補貨管理邏輯

- 實作遠端一鍵補貨功能，結合前端 UI 與後端資料庫同步

[?] 插入機台分佈地圖實機截圖]



[?] 插入庫存矩陣實機截圖]

實作商品管理

- 建立商品管理的 CRUD 功能，方便庫存盤點與資料更新

庫存矩陣功能實作

- 動態展示各貨道庫存狀態，視覺化管理商品存量



通行碼與取貨碼編輯功能

- 實作完整的 CRUD 流程，結合 QR Code 標準化生成

操作紀錄與日誌

- 修正異常顯示問題，精確追蹤取貨與交易歷程



為什麼要換成 MQTT？

解決高併發瓶頸

- 傳統 HTTP Polling 浪費資源，MQTT 可應付海量機台同時連線

即時性與穩定性

- 支援雙向溝通，遠端指令 (B055) 延遲更低

架構解耦

- 透過 EMQX + Go Gateway + Redis List 非同步處理寫入，保護 Laravel 不被拖垮

4 月實作重點

- 溫度回報邏輯優化
- 錯誤日誌 (B013) 佇列處理 (ProcessErrorLog Job)
- 交易紀錄 (B600/B602) 業務邏輯拆分
- 完善 API 文件中心，制定全域通訊規範

五月展望

Phase 1: 系統正式上線

部署全新 Star Cloud 後台，取代舊有系統，並針對線上機台進行大規模連線壓力測試。

Phase 2: 第二階段開發

拓展營銷模組（如廣告排程、折價券系統）與進階報表分析，提升客戶營運效益。