

# 2026 Q1 智慧型履歷解析系統：整合測試報告 (Sample)

檔案編號：TST-20260127-V3 | 機密等級：內部測試用 | 日期：2026/01/27

測試目的：驗證 OCR 文字提取準確度、關鍵字權重分析演算法 (TF-IDF/BERT)、以及 JSON 格式化輸出的穩定性。

## 1. 專案背景與技術架構摘要

本測試文檔旨在模擬真實世界中多樣化的履歷與技術文件格式。系統後端採用 **Python 3.12** 開發，核心架構基於 **FastAPI** 實現高併發的 RESTful API 服務。在資料處理層面，我們使用 **Pandas** 進行結構化數據清洗，並整合 **OpenCV** 進行影像前處理（如：二值化、去噪、歪斜校正），以提升 Tesseract 或 Google Cloud Vision 的 OCR 辨識率。對於非結構化文本，則透過 LangChain 介接 LLM 進行語意分析與實體識別 (NER)，目標是從混雜的文本中精準提取「候選人技能」、「工作年資」與「教育背景」。

## 2. 模擬候選人資料 (Resume Mockup Data)

以下區塊用於測試鍵值對 (Key-Value) 提取能力：

- 姓名 (Name): 陳測試 (Chen, Test-User)
- 期望職位 (Target Role): 後端工程師 (Backend Engineer) / 系統架構師
- 聯絡資訊: email: test\_dev@chunghua.edu.tw | Mobile: +886-912-345-678
- 教育程度: 中華科技大學 (Chung Hua University of Science and Technology) - 資訊管理系
- 語言能力: 中文 (Native), 英文 (TOEIC 790, Target: 900+)

## 3. 技能關鍵字矩陣 (Skill Keywords Extraction Test)

請測試系統是否能從以下密集文字中識別出特定技術名詞：

候選人熟悉 **Linux (Ubuntu/CentOS)** 環境部署，具備 **Docker** 與 **Kubernetes (K8s)** 容器化管理經驗。在 Web 開發方面，精通 **Django** 與 **Flask** 框架，並會使用 **Supabase** 進行資料庫管理。熟悉 **Redis** 快取機制以優化 API 回應速度，並使用 **Celery** 處理非同步任務（如爬蟲與郵件發送）。版本控制使用 **Git/GitHub**，並了解 CI/CD 流程 (GitHub Actions)。前端基礎包括 **HTML5, CSS3, JavaScript (Vue.js/React)**。

## 4. 專案經驗：電商數據爬蟲與自動化機器人

- 專案名稱: Discord 價格監控機器人 (Discord Bot for E-commerce)
- 技術棧: Python, Discord.py, BeautifulSoup, Selenium, SQLite
- 專案描述: 開發一隻全自動機器人，能夠定時爬取 PChome 與 MOMO 等電商平台的產品價格。當目標商品價格低於設定閾值時，透過 Discord Webhook 發送即時通知。解決了動態網頁加載 (AJAX) 的爬取難題，並實作了 User-Agent 輪詢機制以規避反爬蟲策略。
- 量化成果: 每日處理超過 5,000 筆商品數據，通知延遲低於 3 秒。

