

# Hello, I'm Ian

---

## A Frontend Engineer in Taiwan

---

### About Me

我是一名前端工程師，擁有多年以上開發經驗，專注於建構可擴展的前端架構、實作高互動性的使用者介面，以及優化複雜系統的效能。我曾參與開發前台網站、後台系統與 Android TV App 等，並熟悉跨平台產品的設計與實作。我熱衷於解決 UI/UX 挑戰，並透 AI 工具輔助，加速開發流程，持續交付乾淨、可維護且以使用者為核心的解決方案。

I am a frontend engineer with extensive experience in building scalable architectures, developing highly interactive user interfaces, and optimizing performance for complex systems. I have contributed to projects ranging from customer-facing websites and admin dashboards to Android TV apps, with solid expertise in cross-platform product development. Passionate about solving UI/UX challenges, I leverage AI-assisted tools to accelerate workflows and consistently deliver clean, maintainable, and user-centric solutions.

---

### Autobiography

我並非資訊相關科系出身，偶然間看到一個動畫效果精美的網站啟發而開始自學前端。利用工作之餘的時間，我扎實學習 HTML、CSS、JavaScript 等基礎技術，並持續挑戰更進階的框架與工具，開發了口罩地圖、電商平台等實際作品，並公開於 GitHub 作為技術展示。這段歷程不僅培養了我獨立解決問題的能力，也讓我深刻體會到程式架構設計與最佳實務的重要性。

在職涯中，我主要負責後台系統的前端開發，熟悉 API 串接、資料處理與表單互動邏輯設計。同時，我也參與多元專案，包括利用 PixiJS 建構互動式平面圖系統、使用 Kotlin/Jetpack Compose 開發 Android TV 應用程式，以及透過 Python、Docker 與 AWS 建立爬蟲服務。在這些專案中，我不斷累積跨領域經驗，並展現快速學習與應用的能力。

目前我最擅長的是 React 與相關前端技術，並將前端作為長期職涯方向。未來，我希望持續深化前端專業，同時逐步拓展後端能力，成為兼具廣度與深度的 T 型工程師，以更全面的視角推動產品開發與優化。

I did not come from a computer science background, but my journey into frontend development began when I was inspired by a visually stunning website with engaging animations. Fascinated by the experience, I started self-learning frontend technologies such as HTML, CSS, JavaScript, and jQuery during my spare time while working full-time. Over time, I built and published projects like a mask map application and a simple e-commerce site on GitHub, which not only strengthened my technical foundation but also taught me the importance of clean architecture and best practices.

In my professional career, I have mainly worked on admin dashboards, where I specialized in API integration, data handling, and designing interactive form logic. Beyond that, I have also contributed to diverse projects, including developing an interactive floor plan editor using PixiJS, building an Android TV application with Kotlin and Jetpack Compose, and creating crawler infrastructure with Python, Docker, and AWS. These experiences allowed me to expand my technical versatility and sharpen my ability to quickly learn and apply new tools.

Currently, my core expertise lies in React and modern frontend technologies, and I am committed to pursuing frontend development as my long-term career path. Moving forward, I aim to deepen my frontend expertise while progressively exploring backend development, striving to grow into a T-shaped engineer capable of tackling product challenges with both depth and breadth.

---

## Skills

- **Languages & Fundamentals:** HTML5, CSS3, JavaScript (ES6+), TypeScript
  - **Frameworks & Libraries:** React, React Router, Next.js 15, MobX
  - **Styling & UI:** SCSS, Ant Design, Bootstrap
  - **Build Tools & Tooling:** Vite, Vitest, Git
  - **Localization:** i18n
  - **Package Managers:** Npm, Yarn
- 

## Currently Learning

- **Tailwind CSS:** 以 Utility-first 方式快速建立響應式 UI
  - **Next.js:** 支援 SSR 與 SSG 的全端 React 框架
  - **BFF (Backend for Frontend):** 設計針對前端優化的 API Layer
- 

## Career Timeline

- 🕒 **2021/11 - Present**  
**Software Engineer at Miracle Mobile**
- 

## Work Experience

### Nearyou - 房屋租賃平台

- **Tech Stack:** Next.js 15, React 19, TypeScript 5.8, MobX 6, SCSS Modules, Ant Design 5, Google Maps API, Git
- **Role:**
  - 前端架構設計：採用 MVVM 架構模式，建立從 View (React Components) → ViewModel (MobX Store) → Service → API 的分層結構
  - 混合渲染策略：整合 SSR (物件詳情頁)、SSG (靜態頁面)、CSR (互動功能) 以提升 SEO 與載入速度
  - 功能開發：實作房屋搜尋系統 (Google Maps 整合)、多步驟刊登表單、LINE OAuth 登入、後台管理系統
  - 效能優化：使用 React.memo、@dnd-kit 拖曳排序等技術
  - AI 輔助開發：使用 Claude AI 加速開發流程，開發者專注於架構與程式品質
- **Challenges:**
  - 複雜狀態管理：設計 MobX ViewModel 層處理跨頁共享狀態 (搜尋條件、使用者驗證、物件管理邏輯)
  - 地圖整合效能優化：大量地圖標記的效能優化 (clustering 與地圖範圍同步查詢)
  - 動態 SEO：依據物件內容生成標題、描述、og:image 等中繼資料

- **Development Date:** 2025/07 ~ 2025/08

### 亞洲指標 **Crawler System** - 爬蟲系統基礎建設

- **Tech Stack:** Python, Scrapy, Playwright, Docker, AWS EC2/S3, Cronjob, Linux, Git, CI/CD
- **Role:**
  - AWS 雲端基礎建設：自零建立 EC2、Bastion Host、安全群組與網路安全設定
  - Docker 容器化管理：設計並實作多台爬蟲伺服器的標準化容器架構，確保服務穩定運行
  - 爬蟲維護與優化：修正並優化不同網站的爬蟲，涵蓋靜態頁面抓取、動態渲染、模擬使用者操作
  - CI/CD 自動化部署：透過 AWS S3 與 SSM 實作跨多台 EC2 的自動部署流程
- **Challenges:**
  - 大量資料標準化：定義統一的資料格式，處理文章、留言與統計資料等複雜結構
  - 技術債管理：重構舊有程式碼，移除未使用模組、統一命名規範、檢查安全設定
  - AI 輔助維護：使用 Claude AI 自動分析網站結構變更，加速爬蟲維護效率與準確性
- **Development Date:** 2025/04 ~ Present

### **PTS TV App** - 公共電視 **Android TV App**

- **Tech Stack:** Kotlin, Jetpack Compose
- **Role:** 參與 UI 功能與播放控制功能開發
- **Challenges:** 學習全新生態系 ( Android/Kotlin/Jetpack Compose )
- **Development Date:** 2024/09 ~ 2025/01

### **展昭國際** - 展覽場地平面圖系統

- **Tech Stack:** React, React Router, MobX, Ant Design, PixiJS
- **Role:** 開發可互動的平面圖系統
- **Challenges:** 銜接 DOM 與 PixiJS 渲染層，並實作複雜的狀態與復原邏輯
- **Development Date:** 2023/05 ~ 2023/08

### **柯南國際** - 後台系統

- **Tech Stack:** React, React Router, MobX, Ant Design
- **Role:** 實作大量表單與驗證邏輯
- **Challenges:** 使用 MobX computed values 高效管理複雜表單狀態
- **Development Date:** 2023/01 ~ 2023/04

### **KAVAVA** - 美國土地租賃平台

- **Tech Stack:** React, React Router, MobX, Ant Design, Google Maps API
- **Role:** 開發與維護地主後台功能
- **Challenges:** Google Maps API 整合與 UI 資料同步
- **Development Date V1:** 2022/06 ~ 2022/09
- **Development Date V2:** 2023/10 ~ 2024/01
- **Development Date V3:** 2024/03 ~ 2024/05

以下為練習與探索新技術的個人專案

### **Blacknegative**

以 jQuery 重現 Blacknegative 網站動畫

### **Pokemon Store**

使用 Vue 2 與 Vuex 製作的簡易電商網站

### **Mask Map**

使用 Vue 2 與本地資料建立口罩地圖

### **Blogzine**

以 Bootstrap 5 製作的 Blogzine 模板複製版

### **TodoMVC React**

使用 React Hooks、MobX、TypeScript、Vite、Vitest 製作的 TodoMVC