

威摩科技股份有限公司

- 履歷使用公司: 威摩科技股份有限公司
- 使用人員: 廖昱凱
- 使用時間: 2024/02/12 19:20
- 履歷僅提供徵才目的使用，請勿違法蒐集、公開或轉其他用途；若招募作業完成請立即銷毀或刪除，以避免觸犯個資法。

徵試歷程

2024/01/26 17:56 新增備註 | 廖昱凱 | 行動裝置_iOS

2024/01/25 15:32 應徵履歷 | Backend | Sr. Backend Engineer



陳鍵銓 28歲 男 代碼: 20000000162568

最高學歷 大學畢業 國立聯合大學 資訊工程學系

希望職稱 後端工程師

總年資 5~6年工作經驗

最近工作 銓鼎科技股份有限公司 資深後端工程師

居住地 112 台北市北投區石牌路***

E-mail a8130777@gmail.com

聯絡電話 0975-970-087

聯絡方式 手機 12:00~22:00

應徵日期: 2024/01/25 15:32

應徵資訊

應徵職務 Sr. Backend Engineer

自我推薦 您好，我叫陳鍵銓(Harc)，在後端開發和nodeJS ecosystem上都有充足的經驗，近日得知貴公司在徵人，希望能有參加面試的機會，謝謝！

符合分析

履歷符合

根據此份履歷的學歷、經歷、技能等指標，與您的職務所需條件進行比對分析結果。

學歷	非常符合
符合項目	大學
經歷	非常符合
符合項目	[全職] 後端工程師 - 資深後端工程師 (6年...
技能	符合
符合項目	node.js c# db api docker aws python 工具 koa nosql git ci 軟體程式設計 微服務 操作 restful mysql postgresql javascript mongodb redis rdbms golang web_framework typescript c++ graphql elasticsearch grpc microsoft_sql_server gin flask cloud orm pytest queue mqtt dart sdk lua udp windows apollo gateway 軟體開發 scrum message_broker open_source api開發 tcp framework golang開發 通訊協定 自動化 獨立開發 硬體

不符合	linux	repository	mvc	access	kubernetes	java	spring	linux	資料庫	gcp	nestjs	php
	維護系統	scooter	asp.net_mvc	unix	laravel	django	json	口語表達能力	文件撰寫	網站維運		
	coaching	best_practices	系統規劃	工作流程	training	重構	英文-聽(略懂)/說(略懂)/讀(略懂)/寫(略懂)					
期望	非常符合											
符合項目	全職	後端工程師	後端工程師	台北市								
不符合	sr	其它軟體及網路相關業										

性格適合 未施測

透過適性測驗結果比對出性格、職能的關鍵項目，可針對職務需求檢視充分展現或略有不足的內容。

性格
資料不足無法分析
職能
資料不足無法分析

工作經歷

總年資 5~6年工作經驗
後端工程師 (6~7年)

銓鼎科技股份有限公司 資深後端工程師 2017/02~2023/04 (6年3個月)

產業類別	其他電信及通訊相關業
職務類別	後端工程師
公司規模	30~100人
工作技能	Node.js、Python、Git
工作內容	<ul style="list-style-type: none">• IOT資料收發系統 以NodeJS為基礎開發IPC上的client以及配套的server，並獨立設計溝通用的UDP protocol，另外透過自製工具進行維運期的百餘台設備管理、OTA更新。• 幫助公司轉換主力API架構到微服務上 將原有的RESTful API根據公司需求轉換成GraphQL API，並且梳理不同客戶的商業需求將原有服務切割成核心服務及客製服務；接著在此基礎上開發出一個微服務化的GraphQL API並支援hot reload等功能，增加部署的靈活性。• 從零建立公司內部CI/CD環境、流程 和主管共同制定公司內部使用的publish flow、commit style，並搭建stage system、docker registry、drone以及串通整套部署流程，建立從commit到image進入registry的全自動開發流程。• 高效的基礎資料ORM 以Golang、gRPC為基礎開發用於公司大部分服務的基礎資料ORM以增進存取效率。• 車上資料(canbus)收集系統 因應合作廠商的限制，以python開發高擴展性的資料收集client並透過MQTT回傳，經過NodeJS開發的gateway將資料normalize後上傳到message queue以供後續服務介接，並設計平台提供操作硬體設定、車上硬體OTA功能。• 國內外電動巴士車機軟體開發 獨立開發以elecrtion為基礎的windows車機使其具備資料回報、到離站播報、出勤、監視畫面切換等功能，並以NodeJS native module為基礎介接合作廠商的SDK以操作車上週邊硬體，且在其中設計HAL(Hardware Abstraction Layer)以提高維護效率降低硬體迭代帶來的衝擊，以及設計local db、資料收集功能以合乎當地法律規範的車機協定。• IOT車輛藍牙收集設備 使用Golang開發以收集藍牙資料為主的都3(都市交通控制通訊協定3.0)協定server，以供客戶查詢相關資料，並部署在IPC上。• 公司主力產品通用框架的設計及協作 曾協助其他專案進行產品的抽象化設計及語意定義，並將其應用在維護中的其他產品。

教育背景

最高學歷 大學畢業

最高 國立聯合大學 資訊工程學系 (2013/06~2017/06 畢業)

個人資料

就業狀態 待業中

英文姓名 Harc

基本資料 84年次

兵役狀況 未役

持有駕照 普通小型車駕照

求職條件

工作性質 全職

希望職稱 後端工程師

希望職類 後端工程師

希望產業 產業不拘

希望地點 台北市

遠端工作 對遠端工作有意願

希望待遇 面議

可上班日 錄取後二週可上班

上班時段 日班

技能專長

專長 NodeJS

具備多年node開發經驗，可以獨立開發各種API server、TCP/UDP server、gateway、ORM、排程系統等，並且有豐富open source專案閱讀經驗，可以在一定時間內熟練並應用新的ecosystem。

Python

有部分開發經驗，主要聚焦在python的asynchronous方案及狀態機控制以便開發注重擴充性及狀態機管理的IOT軟體，並有少許API開發經驗(使用flask)，此外有深入探討過python的打包方案、開發環境配置方案(tox, pytest, black等)。

Golang

有ORM、gRPC、IOT程式開發經驗，並針對goroutine、SOLID in Golang進行過研究。

過去文章：https://hackmd.io/@HARCHHI/rkf_r4F1m

DevOps

有scrum開發經驗，並熟悉容器化技術(docker, docker compose)。

曾主導建立公司的CI/CD流程(基於Drone實作)，有完整的自動化發布開發經驗。

其他技能樹

其他使用過的/服務語言大致如下

(皆依熟練程度排序，不包含各語言開發環境相關的套件)

- language: javascript/typescript, python, golang, c/c++, c#, lua, dart
- API framework
 - nodeJS: koa, connect, Apollo GraphQL, strapi, express
 - python: flask
 - golang: gin
- nodeJS web framework: react
- database
 - RDBMS: postgresql, mysql, mssql

- noSQL: redis, mongoDB, elasticsearch
- message broker: nats
- API protocol: GraphQL, RESTful http, ws, mqtt, gRPC
- Base protocol: TCP, UDP
- CI/CD related
 - docker
 - drone
- cloud platform: AWS
- others
 - OpenAPI(swagger)
 - markdown
 - git

自傳

- 關於我

我從國中開始接觸軟體開發，早早選定職涯得到的優勢是對技術選擇/熟悉有相當的靈敏度，擅長接觸各種不同的開發體系/語言，並總結出屬於自己的學習方法/開發流程。除了本職的後端開發外，還經手過IOT/VOT軟體的開發，對於硬體SDK串接、HAL設計、軟硬體邏輯隔離及測試也有一定了解。

- 小規模團隊磨練出來的技能廣度

由於前公司開發團隊的規模問題，除了SA/SD以外還需要和PM一起討論產品功能、決定產品方向，此外也經常必須負責包含IOT前端在內的所有系統開發、功能發想、protocol設計，雖然沒有太多的團隊協作經驗但對於前端合作依然有不少的理解。

- 把握時間成本，制定合理的開發時程

因為時常進行一條龍開發的緣故，開發時間永遠是工作時不可忽視的一個環節，此外還需要考量臨時插件/troubleshooting的處理，使得我對於開發時程的掌控有著更多的經驗；在時間允許時我會稍加"奢侈"一下，試著建立更多彈性的核心模組以減少後續的開發/擴展時間，時間不足時也能在儘量降低技術債的前提下快速建立一個符合驗收需求的版本，並在後續協調時程來補完功能/修復技術債。

- 對於可讀性、可維護性、可測試性的追求

經過多年的開發，我熟知可讀、可維護、可測試的重要性，所以習慣性在開發時遵守《Clean Code》的各種建議。並以SOLID為核心進行高可測軟體的開發，且熟悉設計單元測試、整合測試的情境/流程，也熟悉Test Double的應用，致力於開發無論整合還是單元測試都能做到零相依的自動化測試服務。