Liam Richard Lewis

Full Stack Developer

liamricharddev7@gmail.com | +1 724 804 8849 | Mesa, AZ 85213 | US Citizen

Linkedin | Portfolio | 11/07/1996

Skills

Backend Technologies

Node.js (TypeScript, JavaScript, Express.js), Java (Spring Boot), PHP (Laravel, CodeIgniter), C# (ASP.NET), Python (Django, Flask, FastAPI), Go (Golang), gRPC, NestJS

Frontend Technologies

React.js, Next.js, Angular, Vue.js, React Native, Tailwind CSS, MUI, JavaScript, HTML, CSS

Cloud & DevOps

AWS (EC2, Lambda, S3, CloudWatch, ECS, Fargate), Google Firebase, Azure DevOps, Docker, Kubernetes, Jenkins, GitHub Actions, CI/CD, Terraform, Helm

Data Management

PostgreSQL, MySQL, MongoDB, Redis, DynamoDB, CassandraDB, Elasticsearch, Snowflake, BigQuery

API & Protocols

REST APIs, GraphQL, gRPC, JWT/OAuth2 Authentication, WebSockets, OpenAPI, Swagger

Professional Experience

Senior Full Stack Engineer, *HCLTech* Mar 2022 – Mar 2025 | Remote

Project: AI-Driven Government Procurement & Bidding Platform

- Engineered a scalable microservices architecture using **Python (Django, FastAPI, Flask)** for government procurement workflows, achieving 99.99% uptime and processing 500+ daily requests.
- Built a **GraphQL** layer that allowed fine-grained queries over large sets of solicitation data and agency documents, reducing over-fetching and boosting performance by 30%.
- Implemented **WebSocket**-based alerts for newly published opportunities, approaching deadlines, and updates on proposal evaluations, providing users with instant, actionable insights.
- Centralized government contracting data by migrating 100,000+ records into **MongoDB**, and automated real-time updates from government sources via **Apache Kafka**, ensuring the system displayed the most current information.
- Established role-based access control (**RBAC**) utilizing **OAuth2** and **JWT authentication** protocols; mitigated potential data breaches by 65% and ensured compliance with industry security standards.
- Implemented **rate limiting** and **API throttling** using **Redis**, supporting 50+ client applications concurrently while preventing denial-of-service attacks, improving overall system reliability and availability.

Senior Software Engineer, *Netflix, Inc* Oct 2018 – Feb 2022

Los Gatos, CA, United States

Project: Fintech Application

- Pioneered high-throughput backend services in **Java** (**Spring Boot**) incorporating **gRPC** and **Apache Kafka**, processing 5,000+ transactions per second while shrinking average latency to 85ms.
- Optimized **Java Spring Boot microservices**, decreasing latency in critical transaction workflows to an average of 85ms while handling peak loads of 7,000 transactions per second.
- Integrated **AWS Lambda serverless functions** into application architecture, reducing infrastructure costs by 30% and decreasing average function execution time by 50ms.
- Integrated **MongoDB** and **Oracle** with read replicas for high-availability storage.
- Engineered real-time dashboards within **Splunk** that visualized system performance, pinpointing bottlenecks, and improved incident response times by 40% while monitoring 50+ transaction pipelines.
- Automated infrastructure provisioning with **Terraform**, reducing setup time by 60%.
- Fortified system security by enforcing role-based access control across 5 key applications and enabled **AES-256** encryption, exceeding regulatory requirements for financial data protection and compliance.
- Spearheaded the creation of a **fraud detection** system using **event sourcing** that reduced false positives for customers flagged for fraud, decreasing manual review workload by 40 hours a week.

Project: Media-Sharing & Game Platform (Scalable, Cloud-Native Architecture)

- Orchestrated the development of 10+ high-performance backend microservices using **Go** (**Golang**), improving API response times by 30% and enabling support for 2 million concurrent users.
- Leveraged **Apache Kafka** and **Redis** to deliver real-time media updates and game data with sub-100ms latency.
- Refined scalable data infrastructure utilizing **DynamoDB** for efficient storage of media metadata and game states, achieving a 20% reduction in database query latency due to optimized schema.
- Orchestrated a global content delivery network utilizing **AWS CloudFront**, which decreased average media loading times by 60ms for international users and supported 2M+ concurrent streams globally.
- Orchestrated serverless media transformation pipelines via **AWS Lambda**, processing 5M+ images monthly with 99.99% uptime, enabling faster content delivery and improved user experience across platforms.
- Initiated centralized logging using the **ELK stack**, capturing 100% of application logs and reducing debugging time by 40% by providing visibility into application performance, exceeding team goals.
- Instituted a comprehensive disaster recovery strategy employing **AWS Backup** and cross-region replication, achieving a **Recovery Time Objective** of under 15 minutes.
- Orchestrated performance tuning strategies with tools like **Prometheus**, decreasing average game latency by 150ms during peak hours, resulting in a superior user experience and higher player retention.

Software Developer, *The Walt Disney* Jun 2017 – Sep 2018

Burbank, CA, United States

Project: Microservices SaaS Dashboard

- Accelerated core API enhancements with **Node.js (NestJS)** following senior engineer guidance; elevated **API test** coverage scores by 40% and decreased server response times by 150ms, as measured by **Datadog**.
- Fashioned responsive UIs with **Tailwind CSS** and **React**, resolving 80% of cross-browser compatibility issues and boosting average page load speed by 40% on mobile devices.
- Forged scalable data architecture using **SQLite** for local development environments and **Snowflake** for analytical workloads, enabling 10+ engineers to conduct efficient data analysis and reporting.

- Orchestrated microservice communication using **Apache Kafka**, achieving 99.99% uptime, and enhanced application responsiveness by optimizing **Redis** cache invalidation policies during peak traffic.
- Orchestrated infrastructure-as-code using **Terraform**, automating provisioning of 50+ virtual machines, and slashing deployment times by 60%, while ensuring adherence to security protocols and infrastructure compliance.
- Actively influenced in code reviews with senior developers and contributed to documentation and best practice sharing sessions.

Education

Bachelor's Degree in Computer Science, May 2013 – Apr 2017
University of Hong Kong