

# Victor Alvarez

Caracas, Distrito Capital, VE | +58 (412) 558-1218 | [alvarez3197@gmail.com](mailto:alvarez3197@gmail.com)

[github.com/vhalvarez](https://github.com/vhalvarez) | [linkedin.com/in/vhalvarez](https://linkedin.com/in/vhalvarez) | [vhalvarez.com](http://vhalvarez.com)

## EXPERIENCIA LABORAL

### Ingeniero de Software - Arquitecto Backend, FIIIDT

Jun 2024 – Ene 2026

- Arquitecté una plataforma de microservicios soportando 1,000-100,000 usuarios diarios usando Clean Architecture, DDD y CQRS, logrando 40% mayor rendimiento y 60% mejor mantenibilidad del código
- Construí 8 microservicios en producción en dos sistemas: 4 para gestión empresarial (Auth, Users, Reports, Notifications) y 4 para infraestructura PKI nacional (Certificate, User, Audit, API Gateway) permitiendo escalabilidad horizontal con latencia API <100ms
- Desplegué una infraestructura contenerizada en servidores institucionales con Docker Compose orquestando más de 15 servicios (PostgreSQL, Redis, Vault, NATS, Kong, MinIO, Prometheus, Grafana) con balanceador NGINX, automatizando pipeline de despliegue
- Implementé una suite completa de testing con más de 20 tests unitarios Jest, más de 17 tests E2E Playwright y tests de integración logrando más de 60% de cobertura de código en microservicios
- Desarrollé un sistema de generación asíncrona de reportes procesando 200-300 reportes/minuto con colas Bull y almacenamiento MinIO compatible S3, reduciendo tiempo de generación en 75%
- Tecnologías: NestJS, TypeScript, PostgreSQL, Redis, NATS, Bull, HashiCorp Vault, MinIO, Kong, NGINX, Docker, Jest, Playwright, OpenTelemetry, Jaeger, Prometheus, Grafana, Next.js 15, React 19

### Desarrollador Backend y Especialista BI, Bancamiga Banco Universal

Nov 2018 – Jul 2021

- Migré modelos Excel legacy a APIs RESTful Laravel con dashboards interactivos Vue.js sirviendo más de 30 stakeholders
- Ingeniería de pipelines ETL procesando más de 1M de transacciones diarias con actualizaciones WebSocket en tiempo real y visualizaciones Chart.js
- Optimicé consultas SQL reduciendo generación de reportes de 4 horas a 15 minutos (94% de mejora)
- Implementé integración de base de datos Oracle para cuentas de clientes, transacciones y logs de auditoría
- Tecnologías: Laravel, PHP, JavaScript, Vue.js, Oracle, Chart.js, WebSockets, Docker

## EDUCACIÓN

### Licenciatura en Ciencias de la Computación, Universidad Nueva Esparta

Nov 2025

- Cum Laude (17.88/20.0)
- Cursos Relevantes: Ingeniería de Software, Estructuras de Datos y Algoritmos, Sistemas de Bases de Datos, Sistemas Distribuidos

## PROYECTOS

### Ingeniero Full Stack, Lunch Microservices ([github.com/vhalvarez/lunch-microservices](https://github.com/vhalvarez/lunch-microservices))

Sep 2025 – Oct 2025

- Construí una plataforma de microservicios para eventos masivos de donación de comida gestionando más de 200 órdenes simultáneas con 6 servicios independientes comunicándose vía RabbitMQ
- Integré un sistema de predicción IA usando Groq/Llama 3.1 analizando 100 órdenes cada 5 minutos con 85% de precisión para optimización de inventario
- Desplegué en AWS EC2 con Docker Compose automatizando pipeline de despliegue, implementando monitoreo de salud y logging JSON estructurado con Pino
- Implementé tests unitarios y de integración con Vitest cubriendo lógica de negocio principal y endpoints API
- Tecnologías: Node.js, TypeScript, Fastify, Vue 3, PostgreSQL, RabbitMQ, Redis, Groq API, Docker, AWS EC2, Vitest, Pino

### Full Stack Developer, Wan Shi Tong ([github.com/vhalvarez/wan-shi-tong](https://github.com/vhalvarez/wan-shi-tong))

Abr 2024 – May 2024

- Construí una plataforma full-stack de biblioteca gestionando catalogación de libros, seguimiento de préstamos y cálculo automatizado de multas con autenticación JWT
- Diseñé un esquema PostgreSQL con Prisma ORM implementando consultas eficientes para búsqueda y gestión de usuarios con RBAC
- Creé UI Vue 3 responsive con TypeScript, Tailwind CSS y notificaciones en tiempo real
- Tecnologías: Node.js, Express, Vue 3, TypeScript, PostgreSQL, Prisma, JWT, Swagger, Docker