

Owen Eldridge

Resume

Full Stack Software Engineer

owen.eldridge@pm.me · github.com/oweneldridge · linkedin.com/in/oweneldridge

SKILLS

Languages and frameworks

Go, GORM, Python, PyTest, TypeScript, Node.js, React, Next.js, GraphQL (gqlgen), gRPC/Protobuf, Django, Ember.js, Redux, OAuth 2.0

Infrastructure and data

AWS (Lambda, S3, SQS, SNS, Kinesis, EKS, EventBridge, Step Functions, ECS, CDK, IAM, Secrets Manager), PostgreSQL, Redis, DynamoDB

Tools and practices

GitHub Actions, Docker, Podman, Datadog APM/Logs, Statsig, Playwright, Vitest, SonarCloud, CI/CD, microservices, feature-flag rollouts, Agile/Scrum

EXPERIENCE

SmithRx

2025 to present

Software Engineer · Healthcare technology, pharmacy benefits management

- Led cross-system data quality initiatives across the claims pipeline, finding and fixing systemic issues that affected hundreds of thousands of records, with production corrections delivered through formal change control
- Investigated and resolved escalated production reporting incidents through root cause analysis, and wrote the correction runbooks so the fixes outlived the incidents
- Built a CLI validation tool with dozens of automated checks and a thorough unit test suite, providing quantitative evidence of data integrity during a legacy-to-modern claims platform migration

- Leading the Historical Claims Import initiative as engineering lead, coordinating delivery across product, engineering, and data teams
- Built Python automation for production claims analysis that turned hours of manual batch work into minutes, adopted as the operations team's standard workflow
- Designed GraphQL APIs bringing cost calculation transparency to pharmacy claims, with Relay-compliant patterns, graceful degradation for older data, and shared fragments that cut frontend duplication
- Performed 100+ code reviews per quarter across 8 repositories, catching security vulnerabilities, data integrity bugs, and race conditions before they shipped

Indigo Tech Group | Mindbody

2023 to 2025

Technical Lead, Full Stack Software Engineer · Payments

- Authored Python coding standards based on PEP 8/257 that became the team baseline, covering layout, naming, error handling, and documentation while keeping the payments platform PCI compliant
- Engineered a webhook subscription management system on AWS Lambda and DynamoDB, enabling real-time payment event notifications to external systems
- Built card-present transaction processing by integrating the TriPOS Cloud API with Payrix, from POS terminal communication through transaction storage
- Architected a secure, responsive iframe payment component in Next.js and TypeScript for credit card and bank account management inside the core application
- Restructured monolithic Lambdas into modular microservices with PyTest suites at 80%+ coverage, surfacing 60+ bugs along the way
- Designed the Datadog tagging schema and monitoring strategy across payment services, cutting incident response time through structured, contextual logging
- Mentored junior engineers through reviews and pairing, on Pythonic idioms, data structures, and test-driven development

Mariana Tek | Xplor Technologies

2022 to 2023

Full Stack Software Engineer · Boutique fitness studio management SaaS

- Developed the Ember.js admin application and Django REST APIs for a multi-tenant platform serving Barry's, barre3, and Pvolve
- Implemented cart and checkout features including line-item discounts, quick-sale modals, and payment flows for in-studio point of sale
- Built automated end-to-end suites in Selenium and Python covering check-in, checkout, and class reservation flows
- Engineered a multi-tenant customization framework letting franchise clients ship white-labeled apps with their own styling and feature sets

Spearmint

2021 to 2022

Full Stack Engineer · Accessibility-focused GUI for generating JavaScript tests, 1,000+ GitHub stars

- Improved application performance roughly 300% and cut code editor memory use 60% by replacing heavyweight dependencies while preserving full terminal emulation
- Extended TypeScript coverage through the React-Redux layer so action creators, reducers, and state were checked at compile time
- Implemented Redux for immutable, predictable state, eliminating unnecessary re-renders and prop drilling

EDUCATION

Florida Atlantic University, Computer Science. Johnson & Wales University, BS.

OFF HOURS

German wine, classical guitar, adventure cycling, woodworking, ice carving, travel, and cooking most of the world's cuisines at home.

owen.eldridge@pm.me [GitHub](#) [LinkedIn](#)