

# Ronnie Scott

Whispering Pines, NC

| [rscott910@outlook.com](mailto:rscott910@outlook.com) | [linkedin.com/in/ronnie-d-scott](https://www.linkedin.com/in/ronnie-d-scott) | [github.com/rdscott910](https://github.com/rdscott910)

Full Stack Software Engineer with 6 years of experience building and shipping production software at startups. Experienced across the full stack — from relational database design and backend services to customer-facing UI — with a focus on third-party integrations, internal tooling, and modernizing legacy systems. I use AI daily as a development multiplier and have built the infrastructure to make it effective: persistent rule systems, MCP tool integrations, structured implementation plans, and context pipelines that keep agents aligned with project architecture. I care about code quality, I pick up new stacks fast, and I own my work end to end.

## CORE SKILLS

**Languages:** TypeScript, JavaScript, Perl, Python, SQL

**Frontend:** React, Vue, Vuex, Redux, Tailwind CSS, CSS, Bootstrap, Storybook

**Backend:** Node.js, Express, REST APIs, ORPC, Zod, API Integrations

**Databases:** PostgreSQL, MySQL, MongoDB

**AI & Tooling:** Cursor, MCP (Model Context Protocol), Prompt Engineering, Context Engineering, AI Agent Orchestration

**Infrastructure:** Git, GitHub Actions, Docker, GCP, AWS, Electron, Webpack

**Testing:** Vitest, Playwright, React Testing Library, Mocha, Sinon

## PROFESSIONAL EXPERIENCE

### Full Stack Software Engineer

March 2021 – Present

#### *Takeout Central — Remote*

Report directly to company leadership and serve as a key technical contributor across Takeout Central's platform and architecture. Own full-stack projects end to end, collaborate with the mobile team through custom APIs and backend support, and integrate the platform with external systems and services.

- Architected and launched a Square payments integration bridging a legacy Perl monolith with a Node.js service layer, enabling customer profiles, card-on-file workflows, authorization, delayed capture, cancellation, refunds, webhook handling, and audit-friendly transaction flows.
- Reduced payment processing costs by implementing Square as a new payment processor and tailoring payment workflows to Takeout Central's ordering platform.
- Built database-backed idempotency controls for payment, refund, and card operations to reduce duplicate transaction risk and improve retry safety in checkout and back-office workflows.
- Implemented secure card tokenization and vaulting flows using Square card-on-file APIs so reusable payment methods could be supported without storing raw card data.
- Added HMAC-verified webhook ingestion, structured logging, and Prometheus metrics to improve observability, operational monitoring, and incident troubleshooting.
- Delivered integrations with major point-of-sale and order-management platforms, helping unblock important business initiatives under tight deadlines.
- Led end-to-end development of multiple customer-facing features from ideation through deployment, including work critical to outside platform integrations.
- Integrated a headless CMS that allowed marketing stakeholders to manage homepage and static-site content without engineering support.
- Improved delivery-time accuracy by integrating routing-based distance and travel-time calculations into operational workflows.
- Served as a technical lead on platform initiatives by consulting with customers, scoping custom solutions, and estimating delivery effort.

**Tech Stack:** JavaScript, Vue, Vuex, Webpack, CSS, SASS, Perl, MySQL, GCP, AWS, Mocha, Sinon, Node.js

### Software Engineer (Contract)

March 2024 – Present

#### *Virnika — Remote*

Worked directly with founders at an AI-powered restaurant phone-ordering startup to help deliver technical solutions on schedule in a fast-paced startup environment.

- Integrated Virnika with 20+ point-of-sale systems through Deliverect's channel integration, significantly accelerating platform connectivity.
- Assisted in development of the admin dashboard for user account management, billing, payments, and operational data retrieval.

- Contributed technical guidance around production readiness and delivery strategy in support of important investor-facing milestones.

**Tech Stack:** Node.js, Express, MongoDB Atlas, Docker, Retool, AWS, ngrok

## Junior Software Developer

October 2019 – March 2021

### *Takeout Central — Chapel Hill, NC / Hybrid*

Worked directly under the VP of Engineering to contribute to existing systems, build new platform features, and modernize legacy software while rapidly learning the company's stack.

- Rebuilt critical customer-facing pages from Perl and jQuery to Vue.js, translating Storybook and Figma designs into reusable components while improving supporting backend APIs.
- Integrated external APIs into an established Python/Django backend despite no prior production Python experience, delivering new business-critical functionality.
- Built new platform features from concept through deployment, including gift cards, SMS delivery tracking updates, and a referral code program.
- Improved and maintained internal applications used for delivery operations, reporting, sales, and account management.

## SELECTED PROJECT EXPERIENCE

---

### First Quality Books (FQB)

Ongoing

#### *Co-Founder / Software Engineer*

Co-founder and active engineer on a GAAP-aligned desktop accounting application built as a full-stack TypeScript monorepo (Electron, React, PostgreSQL, Node.js). Responsible for contributions spanning database schema design, backend architecture, UI component systems, automated testing, and AI-assisted development workflows.

#### Full-Stack Development

- Designed and implemented a contract-first API layer using ORPC and Zod, enforcing end-to-end type safety from backend procedures to frontend forms across 10+ business entities.
- Built complete CRUD and workflow support for core accounting entities including accounts, checks, invoices, customers, vendors, employees, items, and tax rates.
- Implemented a double-entry accounting ledger system with balanced-entry assertions, reversal flows, transactional writes, and void protections to preserve financial correctness and auditability.
- Designed PostgreSQL schemas for 20+ relational tables using Drizzle ORM, including hierarchical accounts, polymorphic party relationships, domain enums, and money-safe integer storage patterns.
- Created a reusable 50+ component UI library with editable data grids, entity form workflows, fuzzy-search comboboxes, and an IDE-like multi-panel desktop workspace built on Dockview.
- Established automated testing and CI/CD workflows using Vitest, Playwright, GitHub Actions, and cross-platform build validation across macOS, Ubuntu, and Windows.

#### AI-Assisted Development & Agent Orchestration

- Use Cursor as a primary development environment, with AI agents contributing directly to the commit history for refactoring, debugging, testing, and code generation tasks.
- Authored 6 domain-specific rule files that serve as persistent system prompts, constraining AI agent output to match project architecture, coding standards, and business logic requirements.
- Configured 5 MCP (Model Context Protocol) servers providing 75+ tools for Git operations, Figma design-to-code, AWS documentation, and browser-based testing — enabling agents to operate across the full development lifecycle within a single session.
- Enabled 4 Cursor plugins with 17+ agent skills covering CI monitoring, PR workflows, merge conflict resolution, smoke testing, code style cleanup, and design implementation.
- Built a context export pipeline that produces curated codebase snapshots for external AI tools, solving the context window problem while preserving architectural rules and constraints.
- Authored a 1,700-line structured implementation plan that decomposes complex accounting features into dependency-ordered, AI-executable tasks with embedded architecture documentation and GAAP compliance constraints.
- Wrote product requirements that serve dual purpose as human documentation and AI agent context, with explicit scope boundaries so agents avoid over-engineering beyond MVP.
- Maintain clear boundaries on what gets delegated to AI (boilerplate, refactoring, test generation) versus what stays human-driven (architectural decisions, domain correctness, security implications).

## LEADERSHIP, WRITING & COMMUNITY

---

- Published Medium articles on Vue and Webpack topics.

- Mentored junior developers and interns, including supporting an intern who shipped production code during onboarding.
- Organized and led monthly Lunch and Learn sessions to support knowledge sharing and junior developer growth.
- Speaker at Learn to Code RDU / Project Shift meetup events.

## **EDUCATION & CERTIFICATIONS**

---

**Project Shift Software Engineering Fellowship**, Full Stack Web Development — Durham, NC (2019)

**Durham Tech**, Software Development (24 credits completed) — Durham, NC (2018 – 2019)

**Appalachian State University**, Studied Music Education — Boone, NC (2010 – 2014)

**UNC Pembroke**, Studied Music Education (Full Tuition Scholarship) — Pembroke, NC (2009 – 2010)

**Certifications:** Microsoft Certified: MTA Software Development Fundamentals