

# Austin Cripe

904-469-1213 | [austincripe.business@gmail.com](mailto:austincripe.business@gmail.com) | Jacksonville, FL | [linkedin.com/in/austin-cripe/](https://linkedin.com/in/austin-cripe/) | [Portfolio](#)

---

Software Engineer with production experience building **customer-facing planning systems** in Clojure and ClojureScript, focused on clean domain modeling, real-time workflows, and reliable distributed systems. Strong emphasis on **clean design, testability, and operational reliability**.

## PROFESSIONAL EXPERIENCE

### Software Engineer

Clean Coders | April 2025 - Present

- Took ownership of development for **customer-facing planning and estimation features**, collaborating with product stakeholders to clarify ambiguous requirements and ship production-ready solutions.
- Designed and implemented **distributed, auto-scaling infrastructure** and **real-time communication systems** using WebSockets, Webhooks, AWS SNS/SQS, and ElastiCache (Redis).
- Implemented **end-to-end observability** using **OpenTelemetry** with **SigNoz**, deployed via **Docker**, enabling distributed tracing, metrics, and service-level visibility across production systems.

### Audio Engineer and Application Developer

Sight & Sound Productions | April 2024 - April 2025

- Developed custom reports using Jaspersoft Studio and advanced SQL prompts.
- Built web application connecting warehouse and driver workflows using AWS Amplify, DynamoDB, and React.

## SKILLS

**Languages:** Java, Javascript, Typescript, Clojure, ClojureScript, Rust

**Infrastructure & Data:** Datomic, Amazon Web Services, Docker, PostgreSQL, ElastiCache (Redis), SNS, SQS, Pub/Sub

**Architecture:** Distributed Systems, Pub/Sub architectures, Event-driven messaging

**Backend & Real-Time:** WebSockets, REST APIs, Express.js, Socket.IO

**Practices:** Test-Driven Development, Pair Programming, SOLID Design

## PERSONAL PROJECTS

### Custom HTTP Server (Java) | [GitHub Repo](#)

- Implemented a **framework-free HTTP/1.1 server** in Java, including routing, connection lifecycle management, and concurrency.
- Designed a clean separation between **HTTP protocol concerns** and **application logic**, enabling multiple applications to run on the same server.

### Tic-Tac-Toe Platform | Clojure | [GitHub Repo](#)

- Modeled game state transitions and outcomes as pure functions, enabling deterministic simulation, replay, and AI evaluation.
- Deployed with CLI, Quil, and Hiccup for Command Line, stand-alone, and web GUIs.
- Emphasized clean separation between game logic, state management, and I/O.

## EDUCATION

### University of North Florida Development Course

October 2023

Full-time, Immersive Program

Ubuntu, Postbird, VS Code, Agile, Application Development

### Full Sail University

September 2015

Bachelor of Science Audio Arts, Full-Time, Accelerated Program