

# Anthony Arellano

201.686.4015 theanthonyarellano@gmail.com Emerson, NJ [Portfolio](#) [LinkedIn](#) [GitHub](#)

---

## SKILLS

---

React.js, Next.js, TypeScript, React Query, Zustand, Jest, JavaScript, Webpack, Rspack, Module Federation, Kafka, Redux, Express, Spring Boot, Python, SQLAlchemy, Protobuf, HTML, CSS, Tailwind, Material UI, SQL, PostgreSQL, Supabase, WebSocket, SSE, OAuth2, Claude API, Tone.js, Unit Testing, AWS, Terraform, AWS CloudFront, Docker, Kotlin, GitHub Actions, CI/CD

## EXPERIENCE

---

### Peloton Interactive

Senior Software Engineer | Feb 2026 - Present

Full Stack Software Engineer III | Jul 2024 - Feb 2026

Full Stack Software Engineer II | Aug 2023 - Jul 2024

Full Stack Software Engineer I | Mar 2022 - Jul 2023

- Authored Peloton's Module Federation contract for 29 MFEs across 5 teams: singleton: true + requiredVersion enforcement for React and routing, error-boundary-wrapped remotes for graceful degradation on load failure, and a scaffolding CLI that auto-registers new scopes in the central remote registry and generates per-app CI workflows from shared templates.
- Designed a schema-driven form runtime unifying 30+ row types under one editor for RMS, Peloton's homescreen layout tool used by the programming team. Types keyed by a ViewType:ContentType:ContentSource composite hash; Redux state composed via additive per-type property reducers so adding a new type touches only its own file, and new types register a React Router branch rather than modifying shared code.
- Designed Lumberjack, Peloton's first persistent audit system for content mutations, collapsing on-call RCA investigation from hours to ~15 minutes for content-mutation incidents. Captures 15 entity types from 6 producer services with idempotent async batch DLT reprocessing for malformed-event replay. Shipped a production MCP server plus the Cosmos investigator dashboard (JSON diff viewer, filters, URL-driven entity history).
- Built per-PR deploy previews for Peloton's content platform with an element-selector commenting overlay that lets PMs, TPMs, QA, and engineers pin review comments to specific HTML elements. Each PR gets an isolated subdomain serving only its changed remotes from ephemeral S3, with unchanged remotes falling through to stage via runtime window-global override and a CloudFront Function parsing the Host header to route without per-PR infrastructure. Now a daily driver for the team's bug-fix, QA, and agentic PR-review workflows.
- Caught a silent Module Federation breakage in Rspack's handling of ...dependencies spreads during phase-one validation of a 29-MFE Webpack-to-Rspack migration, before rolling to the fleet. Shipped the migration in two phases (container + one representative remote first, then the remaining 27 apps) over one week using a Claude Code agent fleet with human review gates per landing, validated by bundle diffing, visual regression, and manual QA. Cut production builds 9x platform-wide.
- Shipped the real-time editing pipeline for Peloton's class metadata editor: SSE for update fan-out, WebSocket for presence, a React hook with circular buffer. The initial version let incoming server state clobber in-progress form edits; resolved with field-level divergence detection that surfaces a conflict UI on disagreement rather than silently resolving.

Peloton Hackathon 2023 Finalist (Top 6 / 100+ Teams) | June 2023 - June 2023

- Contributed in a cross functional team by developing a React web application which fully integrated with the team's API in order to display a user's League advancements.

## PROJECTS

---

### Fretboard Coach

AI-powered guitar coaching for intermediate players (In progress)

Next.js, React 19, Claude API, Tone.js, Supabase, Tailwind

- Built a streaming Claude API integration with custom system prompts that generate adaptive practice plans with structured session steps.
- Implemented interactive fretboard diagrams with interval highlighting and Tone.js-based ear training exercises for interval recognition.
- Tracks progress across 9 learning layers (foundations through modes to integration) with real-time progress persistence and PWA support.

### Coup

A real-time multiplayer board game built with React, TypeScript, and WebSockets

React, TypeScript, Zustand, Express, WebSockets, Tailwind, Vite, Vercel

[live site](#) | [github](#)

- Built a custom GameSocket abstraction with connection state tracking and 3-second reconnection backoff for resilient real-time communication.
- Synchronized game state across concurrent players using Zustand store with lobby management and dynamic player lifecycle.
- Designed TypeScript message protocol and pnpm monorepo architecture with shared types between client and server.

## EDUCATION

---

App Academy | Full Stack Web Development Bootcamp

November 2021 - May 2022 | Remote

Ramapo College of New Jersey | Bachelor of Arts in Audio Engineering

Fall 2017 - Winter 2019 | Mahwah, NJ