Jayson Virissimo

Staff Software Engineer

Prescott Valley, AZ · Remote · jayson.virissimo@asu.edu
virissimo.info · github.com/jaysonvirissimo · linkedin.com/in/jaysonvirissimo

SUMMARY

Staff software engineer with 11 years in Ruby on Rails, nearly four of them on a regulated alternative-investments platform, owning partner integrations end to end: architecture, compliance rules, data pipelines, incident response, and the partner relationship. Former engineering manager for checkout and payments. Builds the boring, correct parts of financial software: idempotent pipelines, advisory locks and database constraints, authenticated webhooks, and test-first pull requests (600+ merged since 2023).

SKILLS

Languages & frameworks: Ruby, Ruby on Rails, SQL, GraphQL, TypeScript, React, JavaScript, Swift

Data & infrastructure: PostgreSQL, Redis, Sidekiq, AWS, Kubernetes, Argo Workflows, Docker, Sentry

Security & auth: SAML SSO, OAuth 2.0, JWT, HMAC webhook verification, multi-tenant authorization, rate limiting, secrets rotation

Fintech domain: broker-dealer integrations, FINRA licensing rules, AML/KYC onboarding, pre-trade compliance checks, idempotent payment flows, Stripe, Affirm, PayPal

Practices: test-first development (RSpec), architecture decision records, incident response, AI-assisted engineering, mentoring

EXPERIENCE

Staff Software Engineer, SUBSCRIBE — Feb 2023 – Present

Engineer on a regulated platform for private-fund subscriptions, orders, and compliance (Ruby on Rails and React on AWS). Hired as Senior, promoted to Staff.

- Sole engineer and technical owner of the flagship integration with one of the largest US independent broker-dealers, from requirements and API design through launch and support: automated account and team imports, hierarchical advisor teams, and real-time licensing and training checks.
- Owned a second partner integration end to end, from requirements through launch, on a configuration-driven import pipeline built so later partners onboard with minimal custom code.
- Built pre-trade compliance checks that gate order entry on advisor registration and licensing, completed product training, and product concentration limits, covering tens of thousands of advisors across organizations.
- Hardened concurrency handling and data integrity across the integrations with advisory locks, unique background jobs, database unique constraints, idempotent migrations, and retries with backoff; fixed race conditions in account, team, and contact creation.
- Strengthened tenant isolation and organization-scoped authorization across the integrations; delivered HMAC-authenticated webhooks with constant-time verification, support for multiple SAML identity providers, and OAuth token caching.
- Led research, design, and delivery of an advisor training platform (React and Rails) that gates investment-product access on completed training.
- Owned incident response for the integration surface, from alert triage through mitigation, root-cause fixes, and coordination with partner engineers.
- Primary technical contact for partner firms: ran requirements and design sessions with their engineers and managers, and proposed API shapes the broker-dealer adopted.
- Ran zero-downtime data migrations that removed redundant association records and restructured team associations; profiled and optimized the nightly import jobs, cutting query counts by a third to two-thirds per job in profiling against production-shaped data.
- Onboarded and mentored the engineer who now owns the flagship integration, enabling a clean handoff; interviewed engineering candidates.

Senior Software Engineer, Handshake — May 2022 – Feb 2023

Senior engineer on the messaging team of a student career platform (Rails, GraphQL, React).

- Wrote architecture decision records with product managers and designers before building.

- Replaced the student and employer inbox's REST endpoints with GraphQL mutations.
- Moved expensive inbox operations out of the monolith into a messaging service.
- Introduced command objects in legacy controller actions to reduce feature rollout risk.

Engineering Manager, Tuft & Needle Jan 2019 – Apr 2022

Managed a team of up to five engineers and four contractors that owned checkout, payments (Stripe, Affirm, PayPal), promotions, customer-experience tooling, and the retail point-of-sale system for a direct-to-consumer mattress brand, while staying hands-on in the Rails monolith.

- Prevented double charges by making payment capture transactional with order creation, holding a pessimistic lock until capture, and centralizing idempotency-key generation outside the Stripe gateway.
- Extended fraud screening from Stripe-only orders to Affirm and PayPal and reported confirmed fraud, with customer IP, to a third-party fraud service.
- Launched e-commerce sites for two additional mattress brands on the same monolith, including brand-specific configuration and CI auto-deploy pipelines.
- Built career plans with direct reports and negotiated their promotions and raises; several were promoted to Senior and have since become Staff engineers and engineering managers.

Software Engineer, then Senior Software Engineer, Tuft & Needle Sep 2015 – Dec 2018

Built the checkout API, customer-experience tooling, and fulfillment automation in a Rails monolith; second-highest contributor in the codebase's history with 1,000+ commits.

- Upgraded the monolith from Rails 4.2 to 5.1 and Ruby 2.3 to 2.5, later continuing through Rails 6.0 as engineering manager.
- Designed the fallback that let orders complete during sales-tax provider outages.
- Found a SQL-injection exposure in the operations portal's search code after spotting an anomalous raw-SQL error, and closed it by sanitizing user input in the searcher classes.
- Designed key-based device authentication for retail point-of-sale terminals, with a private key provisioned over USB in place of passwords.
- Added caching and per-IP rate limiting to the public Grape API after a 2017 scraping incident degraded the site.
- Introduced automated performance testing and cut product-table queries on the cart endpoint from 22 per item to 1 per item.

PROJECTS

LedgerLore: iPhone and iPad app on the App Store that teaches the engineering side of fintech (double-entry ledgers, payment rails, reconciliation, idempotency, webhooks) through 686 spaced-repetition exercises scheduled by FSRS.

Altkata: Native iOS and macOS app on the App Store with 1,728 spaced-repetition exercises on alternative investments, built on recito, my Swift and Ruby framework for exercise-driven learning apps.

ActiveRecall: Co-maintainer of a Ruby gem that adds spaced-repetition scheduling, from Leitner to FSRS, to ActiveRecord models; 42,000+ downloads.

CERTIFICATIONS

Securities Industry Essentials (SIE), FINRA, Jan 2025

Fundamentals of Alternative Investments, CAIA Association, Aug 2023

Anti Money Laundering and Transaction Compliance Specialization, SkillUp via Coursera, Aug 2026

Graph Developer, Associate, Apollo GraphQL, Sep 2025

EDUCATION

Bachelor of Science, Political Science, Arizona State University 2009