DIVYANSH KHARE

San Jose, CA • (831) 419-3248 • dkhare1998@gmail.com • GitHub • LinkedIn

Full-stack engineer and multi-time founder who takes systems from 0→1 and owns them end to end across frontend, backend, infrastructure, and the agent layer. Most recently, sole architect and tech lead of SCM at Teaching Lab Studio: an AI-agent governance platform that makes autonomous agents safe to deploy against sensitive, regulated production data. Earlier, reversed an **89% revenue-loss trajectory** at Union Pacific Railroad and locked in **$8M+/year**. Known for shipping fast and defending every architectural decision from first principles.

EXPERIENCE

Teaching Lab Studio **Mar 1, 2026 – May 29, 2026**

Software Engineer III • Tech Lead, SCM *Remote*

- Sole architect and tech lead of SCM, taking it from **0→1** to a live production pilot with a large customer while owning engineering, product strategy, and forward-deployed customer work in parallel.
- Re-architected the sensitive-data boundary for FERPA-regulated data from in-flight scrubbing to **structural isolation**: the model operates only on opaque per-run references and numeric metrics, and student PII rehydrates solely at the authorized user's browser behind a per-request authorization check, so names never enter the AI region.
- Built the capability registry, skill launcher, and per-invocation authorization model end-to-end (NestJS, Prisma, Postgres, Next.js, React). Time-bound grants scope to a staff/capability/classroom/material tuple, resolve server-side, and back 7 FERPA-mapped audit events.
- Built SCM into an identity provider for connected AI agents on a Python/FastAPI + LangGraph runtime, where single-use grants read scope from the database row, never the agent's request, and a per-failure-class retry policy bounds each workflow to ≤2 LLM calls.
- Engineered the Temporal runtime with human-in-the-loop approval gates. Deterministic, replayable workflows include a re-validation closure that prevents an authorization expired during human review from ever writing to the database after approval lands hours later.

Union Pacific Railroad **Mar 2023 – Feb 2026**

Software Development Engineer II • Mechanical Systems *Omaha, NE*

- Led the freezer-car modernization that reversed an **89% revenue-loss trajectory** and locked in **$8M+/year** with a **45% efficiency gain**. Built a **deterministic correlation engine** joining wayside, AEI, and CAD signals on trainID over Kafka, reconciling sensor-coverage gaps (humped-car detection from AEI count mismatch, recovered via Yard Systems APIs) and applying an ML-derived health threshold to surface derailment-risk bearings for crew intervention.
- Built an internal, management-scoped MCP-based AI assistant over the reconciled dataset, collapsing a day-to-week analyst SQL lookup (across segregated AEI, Wayside, and CAD databases) into a real-time answer, so field staff locate a humped freezer car and confirm its reefer unit fast enough to save perishable cargo that previously spoiled unlocated.
- Owned the CADX bulletins and authorities features in a **2M-line legacy C++ codebase** on Linux with no AI assistant, then took ownership of the CADX web applications end to end. These systems push compliance-critical operating directives to train crews, from track-blockage and subdivision-closure bulletins to severe-weather and track-change authorities.
- Established a **backend-for-frontend (BFF) auth architecture** for the CADX web apps, which previously exposed SSO session cookies to the browser. Routed the frontend through a server-side proxy holding the session, moved auth to httpOnly cookies, and added CSRF double-submit protection, so tokens never reach client JavaScript.
- Shipped field inspection and reporting tooling as an offline-first hybrid web container on regulated Zebra devices, chosen over native to ship fast to internal field teams. Edge caching and async pipelines cut inspection and reporting time **60%**.
- Decomposed a legacy C++ monolith into modular services, a **70% improvement** in real-time data responsiveness across three enterprise applications.
- Stood up structured telemetry at >90% coverage and used it to pinpoint failures across multi-service systems. Production downtime cut **50%**, deployment cycles **40% faster**, with automated regression testing reducing pre-release defects.
- Built high-availability frontend platforms for Automatic Equipment Identification and Wayside Detection, replacing log-driven workflows with consistency-aware UIs, intelligent prefetching, and structured observability; enterprise data accuracy up **85%**.

VortexeAI **Jun 2024 – Oct 2024**

Co-founder & CEO *San Jose, CA*

- Took VortexeAI **0→1 in four months** to **200+ daily active users** and a **TechStars invitation**, building a low-code, AI-driven workflow editor for product and engineering teams.
- Engineered a deterministic multi-agent orchestration engine (graph traversal + dependency resolution) at **99.9% execution reliability**.
- Built the React/Next.js node editor and LLM tool runtime with memory routing; 50+ automation patterns deployed, 40% faster onboarding.

AetherisVis **Nov 2024**

Co-founder & CTO *San Diego, CA*

- Shipped a browser-based 3D equipment configurator for a luxury gym brand in **one week**, with real-time 4K/8K customization, delivered as a commercial client product.
- Built a modular React + TypeScript visualization framework (35+ reusable components) that cut UI prototyping time 60% across client projects.
- Held real-time rendering responsiveness under concurrent load via lazy loading and interaction-loop tuning.

Overwhelming Studios **Nov 2022 – Mar 2023**

Founder & CTO *Campbell, CA*

- Earned direct recognition from Ant Rogers (Head of UI/UX, PlayStation) for independently developed games, reaching **promotion-support stage with PlayStation Studios** before visa constraints halted progression.
- Shipped two Unreal Engine 5 mobile titles with cloud auth and cross-device sync for **1,500+ players**, with 30% faster load times and 25% better retention.

SELECTED PROJECTS

Sense UI **Dec 2024 – May 2025**

Solo Developer • AI-comprehensible component architecture

- Built a metadata-driven component architecture that lets AI agents comprehend and operate UI, a **40% lift** in accurate agent UI interactions and 30% less integration debugging.
- Schema extraction + introspection pipeline standardized component behavior so LLMs reliably execute multi-step tasks; deployed across 15+ projects, 50% fewer user-guided corrections.

Atlas Admin Panel **Aug 2025**

Solo Developer • NestJS, PostgreSQL, Elasticsearch, Supabase, Docker

- Built an admin dashboard for **10,000+ concurrent users** with sub-second prefix and fuzzy search via tuned Elasticsearch indexing.
- Role-based invite system, cursor-based pagination, and Redis caching cut data-retrieval latency 35%; zero-flicker hydration model on a Next.js + React 18 frontend.

EDUCATION

University of California, Santa Cruz, B.S. Computer Science: Computer Game Design **Aug 2022**

SKILLS

Languages: TypeScript, Python, Java, C++, C, C#

Frameworks: React, Next.js, Node.js, NestJS, FastAPI, Spring Boot, Tailwind

AI / Agentic: LangGraph, Anthropic API (Claude), MCP, Temporal workflows, multi-agent orchestration, eval pipelines

Infrastructure: PostgreSQL, Prisma, Temporal, Docker, Kubernetes, AWS, Elasticsearch, Redis, Supabase