

ALEXUS KARL JOAQUIN

Senior Full Stack Lead Engineer · AI-Native Development · Published iOS App Developer
+63 930 623 6460 · alexuskarljoaquin@gmail.com · alexuskarljoaquin.com (portfolio with live app demos)

PROFILE

Senior Full Stack Lead Engineer with 8 years of experience taking complex platforms from blank page to production — and 4+ years leading 5-engineer teams at three consecutive companies while staying deeply hands-on. One of the earliest adopters of AI-native development in his organizations: introduced Model Context Protocol (MCP), Claude Code, and Cursor AI company-wide and retrained entire teams around AI-powered workflows, fundamentally changing how software gets shipped. Designed, built, and launched a commercial iOS finance app — Cashn't — solo, from product design through App Store release, and has since shipped seven more products end to end.

SELECTED PROJECTS — BUILT SOLO, ZERO TO DEPLOYMENT

All architected, built, tested and deployed by me alone, with an AI-native workflow (Claude Code and Cursor driven by per-repo skill files; decisions, runbooks and checklists written down as the source of truth). Live demos: alexuskarljoaquin.com.

Cashn't — Budget Tracker | Paid iOS app, live on the App Store (apps.apple.com/app/id6789244062) 2026 – Present
React 19 · TypeScript · Vite · Capacitor 8 · Firebase Auth · Express + PostgreSQL on Railway · Xcode / App Store Connect API

- **Architecture:** local-first — state lives on the device and syncs as versioned snapshots to a small Express + Postgres API (GET/PUT/DELETE per user), so the app works offline and as a guest, and account deletion is one endpoint (App Store 5.1.1). Firebase handles Apple, Google and email sign-in; guests sync under a device id and migrate on sign-in.
- **Design system first:** wrote DESIGN.md ("Banknote After Dark") before any UI — colour tokens, a CVD-checked chart palette with contrast ratios, type scale, motion rules — then built every screen from those tokens; a custom service worker makes the web build an installable PWA.
- **Release engineering:** headless pipeline — capacitor-assets for icons/splash, xcodebuild archive/export with automatic provisioning via an App Store Connect API key, altool upload — and an App Review demo account reset through the Firebase Admin SDK. Passed review first time; v1.1 shipped days later.

LOCA — hyperlocal give / ask / borrow / help app · pnpm monorepo · React + Vite · NestJS + Prisma · PostgreSQL 18 · Capacitor iOS · Caddy on Railway

- **How it was built:** monorepo with shared types/values packages; a 34-table Prisma schema evolved through 18 migrations; the repo runs its own embedded PostgreSQL and seed data, so a new machine is productive in three commands; environment validated at boot.
- **Security and quality:** cookie sessions for web, token sessions for the native app, origin-checked writes, per-client rate limits, CSP/HSTS; 17 end-to-end suites (auth, transactions, messaging, push, moderation...) run against the local database. Deployed with Caddy serving the static app and proxying /api over Railway's private network, so the API, Postgres and storage never get a public address — all from a written DEPLOYMENT.md checklist.

Ghostn't — dating app open one hour a night · React 19 · Capacitor 8 · Express 5 · PostgreSQL · WebSockets · CallKit (Swift) · Railway

- **Performance by design:** one pg pool with statement timeouts; every endpoint a single round-trip (CTEs, LEFT JOIN LATERAL, no N+1); keyset pagination; partial indexes; advisory locks serialising competing line-opens; WebSocket push with heartbeat reaping instead of polling. Client: route-level code splitting (1–5 KB screens) and server-authoritative countdowns with measured clock skew.
- **Shipping:** SQL schema migrations, a scripted end-to-end smoke test that walks the reviewer's path (profile → room → open a line → chat → hold → delete account), seeded demo personas to keep the room alive for App Review, a health-checked Railway service, and the same headless TestFlight pipeline as Cashn't.

Studyverse — student productivity & social learning · React 19 · TypeScript (strict) · Vite 8 · Tailwind 4 tokens · Node + PostgreSQL · WebSockets · OpenAI · Capacitor

- Built mock-first: a one-file service switch runs the app on in-memory mocks or the real API, so UI and backend progressed in parallel. Stale-while-revalidate cache with prefetched tab roots; WebSocket client (reconnect, backoff, heartbeat) applies events straight into the cache. Brand assets generated from geometry by a scripted SVG/sharp pipeline; every route verified at iPhone size in both themes with zero console errors and WCAG AA contrast. API containerised with migrations and seeds on Railway.

Cinemi — AI photo studio · React 19 · TypeScript 6 · Vite 8 · Tailwind 4 · Radix · CVA · MediaPipe WASM

- Token-driven design system (primitives → semantic tokens → Tailwind theme) with Tailwind's default palette disabled so only semantic utilities compile; the reference page reads token values live from CSS so docs cannot drift. Face retouching runs MediaPipe's face landmarker in WebAssembly, lazy-loaded on first use, so photos never leave the device.

Job Finder · FastAPI · httpx · PostgreSQL · React · TanStack Query & **Alexus Legal AI** · Next.js 14 · OpenAI streaming · PostgreSQL · PWA

- Job Finder: a provider module per job board — free direct scrapers (SEEK JSON API, LinkedIn guest endpoint, Indeed mosaic JSON) with RapidAPI fallback — fanned out concurrently with asyncio, deduplicated and cached six hours; weighted scoring engine (skills 45 %, title 30 %, recency 15 %, location 10 %); SQLite locally, Postgres on Railway. Alexis Legal AI: Next.js route handlers streaming OpenAI responses, a prompt engineered for empathetic case-specific answers, per-user conversations in Postgres.

WORK EXPERIENCE

Senior Full Stack Engineer | Cloud Employee

Apr 2025 – Present

- Lead development of scalable production applications on Ruby on Rails, ReactJS, and PostgreSQL — owning features end to end from database schema and API design through frontend delivery and post-release monitoring.
- Pioneered the company's shift to AI-native engineering: introduced Model Context Protocol (MCP) integrations that connect AI assistants directly to internal tools and data sources, turning AI from a chat toy into a working part of the development pipeline.
- Spearheaded company-wide rollout of Claude Code and Cursor AI — evaluated the tools, made the business case, wrote the usage standards, and personally trained engineers — resulting in visibly faster feature delivery and a team that now builds AI-first by default.
- Authored the organization's AI-assisted development standards and best practices: prompt patterns, code-review rules for AI-generated code, and guardrails that let the team move fast without sacrificing quality.
- Designed and implemented the observability stack with Grafana dashboards and Sentry error tracking — production issues are now caught by alerts, not by customers. Built and maintain the engineering knowledge base in Notion (onboarding guides, architecture decisions, runbooks).

Senior Full Stack Lead Engineer | Capital Source Group

Jan 2024 – Apr 2025

- Trusted to design, build, and launch the company's core Financing System entirely from scratch — a system processing real financial transactions where correctness and uptime directly affect the business — using Python (Flask, Alembic) on the backend and ReactJS on the frontend.
- Architected the full AWS production environment: RDS for transactional data, Elasticsearch for fast search across financial records, and load balancers for high availability — a resilient architecture built to survive traffic spikes and component failures.
- Engineered Selenium-based web scrapers that automated data collection previously done by hand, removing a manual bottleneck and feeding the platform with fresh, reliable data around the clock.
- Owned the delivery pipeline end to end: CI/CD via GitHub Actions, frontend distribution through S3 + CloudFront, and domain management — one engineer accountable for the entire path from commit to customer.
- Led a team of 5 engineers in a daily-sprint Agile/Scrum environment: ran plannings and retros, unblocked engineers daily, reviewed every significant change, and kept a from-scratch build shipping on schedule.

Senior Full Stack Lead Engineer | Vertere Global Solutions

May 2021 – Jan 2024

- Spearheaded full-stack development of financing systems over a nearly three-year tenure — Python (Flask, Alembic) backends and ReactJS frontends — carrying products through multiple major iterations while keeping them stable in production.
- Fully integrated the AWS stack — Elasticsearch, RDS, load balancers, S3, and CloudFront — designing for both performance and fault tolerance so the platform stayed responsive as usage grew.
- Implemented Selenium-based data pipelines and stood up CI/CD on GitHub Actions, replacing risky manual deployments with repeatable, automated releases; owned end-to-end system stability — monitoring, tuning, and hardening.
- Led and mentored a 5-member engineering team, instilling Agile/Scrum discipline with daily sprints and growing junior engineers into confident, independent contributors.

Software Engineer | FurtherEdge Co.

May 2018 – Jan 2021

- Recruited on the strength of freelance work delivered while still a second-year college student — client-facing web applications built with Angular frontends and Python backends, shipped to real users on real deadlines.
- Designed and built RESTful APIs, database-backed features, and user-friendly interfaces across a wide variety of client projects; deployed and hosted production applications on AWS (S3, EC2, RDS).
- Contributed to larger-scale team projects in a structured professional environment — all while completing a BS in Information Technology, graduating Magna Cum Laude.

TECHNICAL SKILLS

Languages & Frameworks: Python (Flask, Django, FastAPI), Ruby on Rails, JavaScript/TypeScript, ReactJS, Next.js, NestJS, Angular, Tailwind CSS

AI & Automation: LLM Integration, Model Context Protocol (MCP), AI Agents, Claude Code, Cursor AI, Prompt Engineering, Workflow Automation

Cloud & DevOps: AWS (EC2, S3, RDS, CloudFront, Load Balancers), Docker, GitHub Actions, CI/CD, Railway

Data & Observability: PostgreSQL, MySQL, MongoDB, Elasticsearch, Prisma, Grafana, Sentry

Mobile & Product: iOS App Development (Capacitor, CallKit), App Store Publishing & Review, Product Design, Design Systems, Release Management

Practices: Software Architecture, System Design, RESTful APIs, Performance Optimization, Agile/Scrum, Technical Documentation (Notion)

EDUCATION

BS Information Technology, Magna Cum Laude

2016 – 2020

Nueva Ecija University of Science and Technology