

ELAN NASH

Software Engineer

Los Angeles, CA

✉ elanmnash@gmail.com  linkedin.com/in/elannash  github.com/elannash  elannash.com

SUMMARY

Software engineer building member-facing web apps, **Python** backends, and AI-powered job-matching tools. Replaced spreadsheet workflows with member-facing job approval and automated resume generation.

SKILLS

Languages: Python, TypeScript/JavaScript, SQL, C++

Backend & APIs: FastAPI (async, streaming), Pydantic, SQLAlchemy, Alembic, REST APIs, JWT auth, Node.js, Next.js Edge Runtime

Frontend: React, Next.js, React Native (Expo), Zustand, React Query, TanStack Router, Tailwind

Databases & Data: PostgreSQL (RLS, RPCs, triggers, trigram similarity), Supabase, Airtable

Cloud & Testing: Docker, Google Cloud Run, Cloud Scheduler, Render, Vercel, GitHub Actions, pytest, Jest, Vitest, Playwright, Maestro

AI / LLM: Anthropic API (Managed Agents, custom tools), OpenAI API (embeddings, structured outputs), LiteLLM, source checks, per-run cost tracking and session budgets. Daily Claude Code and Codex with custom skills and CI gates

EXPERIENCE

GHN Career Academy – Software Engineer

Apr 2024–Present

Online career-coaching academy. First engineer, initially solo, now on a small team under a CTO. About 150 members.

Technologies: Python, FastAPI, PostgreSQL, Next.js, React, Google Cloud Run, Docker, Anthropic API, OpenAI API

- Shipped **CareerHub**, the member-facing match approval app on Next.js with the Edge Runtime. Its Zustand store tracks each decision as pending, confirmed, or failed, so server hydration never overwrites an in-flight write and a failed save blocks further decisions
- Built **Resume Crafter**, which generates a base resume for every member on intake from GPT structured output validated by Pydantic schemas, rendering ATS-compliant DOCX templates and skipping repeat runs through SHA-256 payload hashing
- Reordered the **job-matching pipeline**, which sources and scores job postings for every member, after the first week's OpenAI bill ran about 30x budget and exposed a filter-ordering regression. Deterministic rules and an embedding gate now run before LLM scoring, holding measured cost to **about \$1.50 per run**
- Designed the **member-network search backend** in FastAPI, ranking about **74,000 contact records** by relevance and connection degree with stable pagination in PostgreSQL and a contact-ID tiebreaker for equal scores
- Deployed the pipeline as **Cloud Run** functions released from source with pinned revisions, Cloud Scheduler jobs, Secret Manager checks, and OIDC auth between functions, with an offline cost estimator run against live data before each release

NXTMode – Full-Stack Developer (Capstone)

Aug 2024–Dec 2024

Technologies: React, Node.js, Express, TypeScript, MongoDB, Firebase

- Delivered a full-stack **React + Express** app on a **5-person team** for an external stakeholder, owning the REST API over MongoDB and Firebase and taking every change through peer code review and CI under a semester deadline

TECHNICAL PROJECTS

not'd – Social Album-Logging Mobile App (Full-Stack Focus)

Technologies: React Native, Expo, TypeScript, PostgreSQL, Supabase, Edge Functions (Deno), React Query, Zustand, Jest, Maestro

- Own app engineering and UI/UX, from a **React Native + Expo** client with gesture-driven interactions to a Supabase/PostgreSQL backend with Apple Sign-In and a Deno edge layer that signs ES256 JWTs for Apple Music
- Designed **PostgreSQL access controls** for album feeds and follow state through **RLS and RPC-controlled writes**, with append-only consent history, age-gate immutability, and server-owned policy version stamping
- Distinguished **unconfirmed writes** after transport failures from definite rejections, refreshing cached data before reporting the outcome. Quarantined corrupt local list-operation records so unrelated operations can continue

Custom Relational Database Engine

Technologies: C++, SQL Parser, B+ Trees, Binary-File Storage

- Wrote a relational database engine in **C++** with hand-built data structures, parsing SQL with a custom tokenizer and evaluating boolean WHERE clauses through Shunting Yard over **B+ tree indexes** stored in binary files

ADDITIONAL EXPERIENCE

Pasadena City College – Teaching Assistant

Jan 2022–May 2022

- Taught data structures, algorithms, and computer science fundamentals to about **40 students**

EDUCATION

University of Southern California – B.S. Computer Science

Graduated Dec 2024

Pasadena City College – A.S. Engineering and Technology

May 2022