

YIAN GE

☎ 254-400-6327 ✉ yiange131@gmail.com [in Yian Ge](#) [AlanYlan](#) [MyWeb](#)

Education

Northeastern University

Sep. 2024 – May 2026

Master of Science, Software Engineering Systems | **GPA: 3.9/4.0**

Boston, MA

- Coursework: Algorithms, AI Generative Models, Prompt Engineering for Generative AI, Operating Systems

University of Texas at Austin

Sep. 2022 – May 2024

Bachelor of Science, Applied Mathematics | **GPA: 3.9/4.0**

Austin, TX

- Honors: Honoree Listing 2024, Honors Graduate

Technical Skills

AI / ML: LLM application engineering (Anthropic, OpenAI, Gemini), RAG, vector search (FAISS, sqlite-vec, pgvector), embeddings & retrieval scoring, agent orchestration (LangGraph, MCP), prompt caching, PyTorch

Languages: Python, TypeScript, Swift, Kotlin, Java, C++, SQL

Frameworks & Infra: React, Next.js, FastAPI, NestJS, GraphQL, PostgreSQL, Redis, Supabase, Docker, AWS, Cloudflare, Linux, CI/CD (GitHub Actions), Playwright

Experience

OptiSigns Inc.

Jun. 2025 – Present

Software Engineer (intern through May 2026)

Houston, TX (On-site)

- **Co-authored** the internal tooling AI coding agents use across the company's ~40 repositories, including cross-repo context loaders and a reusable skill library with a **meta-skill for onboarding any third-party API**; designed a three-session agent workflow (plan / code / verify) where **the verifier cannot edit what it checks**; measured where agent time actually goes, and the findings were adopted as engineering practice.
- **Integrated** the product across **21 device platforms** (Samsung Tizen, LG webOS, Amazon Fire TV, TCL, Roku, ChromeOS, Android, embedded Linux), each with its own SDK, signing chain and capability limits, and took **4 of them** from vendor SDK to full portal integration.
- **Shipped** device management for a fleet of **200,000+ screens** across a React/TypeScript portal, a NestJS/GraphQL backend, device agents and embedded Linux firmware, replacing three hand-maintained lists of what each device model supports, which had drifted apart, with one generated from what the hardware actually reports and guarded by a CI check.
- **Built** the on-device operating-system interface customers see on the screen, from first-boot setup through diagnostics, fully operable from a phone for a display that has no keyboard, shipping enabled in the default product image.

Projects

EchoVessel: Local-First Digital Persona Engine | Python, FastAPI, sqlite-vec, React 19

Apr. 2026

- **Designed** a **four-layer memory architecture** that distills conversation into durable memory, then retrieves it by **RAG over a local vector store** (sqlite-vec + sentence-transformer embeddings) scored on relevance, recency and importance.
- **Shipped** it as a local-first daemon with pluggable LLM providers, two-way streaming voice (**Whisper ASR**, **Fish Audio** cloned-voice TTS), and Web and Discord channels sharing one persona state over SSE.

EchoTwin: Full-Duplex Voice Agent | Python, sherpa-onnx, Silero VAD, Fish Audio TTS, asyncio

Jun. 2026

- **Engineered** a full-duplex voice agent holding **first sound under a second**, chaining **sherpa-onnx** streaming ASR, **Silero VAD** and **Fish Audio** streaming TTS over asyncio, interruptible mid-sentence.
- **Ran tool calling inside the streaming turn** with **prompt caching** across rounds, so the agent can look something up mid-sentence without breaking the latency budget.

Roomet: AI Interior Redesign | Next.js 15, FastAPI, Gemini, YOLO-World, arq/Redis

Feb. 2026

- **Built** a photo-to-redesign pipeline on **Gemini** image generation, producing three variants in parallel, each conditioned on a different sample of style reference images rather than a different seed, then editable **one piece of furniture at a time** by pointing at it, using **YOLO-World** open-vocabulary detection to isolate that object for the model to regenerate.
- **Moved** the expensive work off the request path: generation onto an **arq + Redis** worker queue and segmentation into a cached lazy pass, with the detector on CPU so the deployment needs **no GPU**.

Adversarial Safety Evaluation Pipeline | Python, vLLM, LLM-as-judge, QLoRA/PEFT

Jul. 2026

- **Built** the measurement half of an adversarial testing pipeline: it generates attack prompts, runs them against a target LLM on **vLLM**, and has a **judge model** rule on whether each answer broke policy across **16 risk categories**, with cases the two judges disagree on routed to human review.
- **Made the judge trustworthy first**, raising its agreement with human labels from **Cohen's κ 0.15 to 0.44** and requiring judge and target to come from different model families so a model never grades its own kind; also benchmarked few-shot prompting against **QLoRA fine-tuning** for generating the attack prompts.