Shengran (Kevin) Jin

kevinjin0420@gmail.com
linkedin.com/in/kevinjin0420 · github.com/kevinjin0420 · kevinjin.dev +1-240-756-8098

Education

University of Michigan, Ann Arbor Ann Arbor, MI
B.S.E. in Computer Engineering; GPA: 3.87 / 4.0 *Sep 2024 - May 2028 (Expected)*

Relevant Coursework: Data Structures & Algorithms (EECS 281), Operating Systems (EECS 482), Computer Organization (EECS 370), Web Systems (EECS 485), Machine Learning (EECS 445), Computer Security (EECS 388)

Experience

TikTok San Jose, CA
Backend Software Engineer Intern *May 2026 - July 2026*

- Raised a Go and Flink data-processing pipeline from 55% to 98% success rate at up to 160,000 records per day by hardening retry, timeout, and failure-recovery logic across stages
- Rebuilt the ingestion stage with schema validation, deduplication, and normalization, added per-stage anomaly detection. Backfilled three weeks of silently dropped data and validated using a Grafana dashboard
- Decomposed a monolithic Python prototype into 12 Go microservices behind a cache-first serving layer, defining service boundaries and request contracts, cutting repeat-analysis latency from 10 seconds to under 20 milliseconds
- Shipped features across an unfamiliar large-scale backend, clearing blockers by consulting service owners directly and negotiating API contracts with the frontend team

Michigan Mars Rover Student Project Team Ann Arbor, MI
Teleoperations Lead *July 2025 - July 2026*

- Redesigned the teleoperation backend on an async FastAPI server with per-subsystem WebSocket endpoints and MessagePack binary framing replacing JSON, cutting round-trip command latency by 40%
- Marshalled multi-threaded ROS2 callbacks onto a single asyncio event loop with lock-guarded scheduling and per-topic rate limiting, giving ordered WebSocket delivery that eliminated dropped commands
- Prototyped a MuJoCo-based C++ physics simulator intended to replace the team's GLFW/Bullet simulation stack for hardware-in-the-loop testing of the control and autonomy software stacks

Projects

Embedded Telescope github.com/eecs373-w26-telescope
STM32 and Raspberry Pi telescope that guides users to deep-sky objects *C / C++*

- Owned the Raspberry Pi HUD and the STM32 encoder, IMU, and GPS drivers on a 4-person team, designing a C++ hardware abstraction layer that wraps each sensor in a uniform peripheral class with RAII-managed bus handles
- Wrote a templated running-average filter over angular sensor readings that stays accurate across wraparound
- Designed a framed binary UART protocol with CRC-16 integrity checks and a streaming decoder that resynchronizes after corruption, giving ordered, validated, and reliable telemetry between the two processors
- Architected the Pi heads-up display around a finite state machine, utilizing a background serial decoding thread integrated with a mutex-guarded shared state that drives a 60 fps Raylib render loop

Job Alerts
Personalized job-posting alert service built on a serverless, event-driven scraping pipeline *Python*

- Architected a serverless pipeline that offloads enrichment and per-user classification to independently scaling SQS workers, keeping signup-driven classification load off the scan Lambda's fixed 5-minute budget
- Re-architected the read path from multiple NoSQL lookups to a single Aurora DSQL query behind a thread-safe connection pool with optimistic-concurrency retries, cutting request latency from 5 seconds to under 1
- Designed a URL-derived canonical key that resolves the same posting seen from multiple sources under different URLs, tuned to prefer occasional duplicate alerts over missed listings

Technical Skills

Programming Languages: C++, Go, Python, C, TypeScript, Java, SQL

Frameworks and Technologies: Linux, Docker, gRPC, Protobuf, MessagePack, Kafka, Flink, Spark, Redis, WebSockets, FastAPI, Flask, Spring Boot, PostgreSQL, MySQL, SQLite, Amazon Aurora DSQL, Three.js, ROS2, STM32 bare-metal

Concepts: Object-Oriented Design, Distributed Systems, Concurrency, RESTful and WebSocket APIs

Tools and Platforms: Git, GitHub Actions, CMake, AWS (Lambda, SQS), Google Cloud, Grafana, Shell, LaTeX