

Jai Sinha

München, BY, DE | me@jsinha.com | linkedin.com/in/jai-sinha | github.com/jai-sinha

EDUCATION

University of Wisconsin–Madison

Aug 2021 – May 2025

Bachelor of Science in Computer Science

Madison, WI

Relevant Coursework: Linear Algebra, Applied Statistics, User Interfaces, Computer Engineering, Computer Graphics, Machine Organization, Algorithms, Artificial Intelligence, Big Data, Database Implementations

EXPERIENCE

Software Engineering Intern

May 2024 – Aug 2024

Tesla

Palo Alto, CA

- Redesigned tester data upload pipeline with concurrent Go routines, reducing cycle time by up to 30 seconds per test run on critical manufacturing lines yielding 10%+ throughput gains and eliminating data losses with retries.
- Led Go development for a new vehicle's main battery controller, coordinating with Battery and Firmware teams to deliver 11 test states ahead of schedule; contributing to shared libraries used in 10+ other products.
- Built hardware validation tool with Go backend and React frontend, using WebSocket + channels to individually verify 14 functions across off-the-shelf tester components, catching integration issues pre-deployment.
- Extended shared CAN messaging toolset with 20+ new functionalities for parsing bit-level payloads, collaborating with Firmware team to abstract low-level protocols into Go and Python for widespread use.

Software Engineering Intern

May 2023 – Aug 2023

Tesla

Palo Alto, CA

- Developed LabVIEW software for high-volume tester, shortening tester duplication process by 30+ minutes, and supported debugging and error analysis across a 15-hour time zone difference.
- Supported 50+ hours of tester pre-ship software validation and hardware troubleshooting; learned on-the-fly to solder, read wiring diagrams, use multimeters and thermocouples, and effectively communicate cross-functionally.
- Created template dashboard using internal SQL-based tools to visualize tester performance based on metrics such as first-pass yield, Pareto failure analysis and more, with process duplicated across 1000s of testers.

PROJECTS

Lebensmittel | *Swift, SwiftUI, Go, Gin, Nginx, PostgreSQL, GCP* | ls.jsinha.com

- Designed and shipped a full-stack shared grocery + meal-planning app with live sync through WebSockets; working proof-of-concept completed in 5 days, current product open-sourced and downloadable on iOS App Store.
- Built Swift client with ViewModel architecture using Observable models paired to modular SwiftUI components for efficient re-renders and business logic flow; created singleton WebSocket and auth managers for single-flight JWT refresh using Keychain-backed token caching.
- Deployed a lightweight Go/Gin + PostgreSQL backend on a free-tier GCP e2-micro instance, manually configuring code and infrastructure to handle 50+ concurrent users with \$0 in server costs and <1ms average server latency.

Open World | *TypeScript, MapLibre, Bun, OpenStreetMaps, Cloudflare R2* | openworld.jsinha.com

- Built a fully client-side SPA that visualizes a user's Strava activities on an interactive map and computes global explored road coverage asynchronously, maintaining smooth canvas rendering with a <75ms INP.
- Engineered a self-hosted geodata pipeline to efficiently compress global city boundaries, reverse-geocoding lookup data, and road tiles into a 25GB Cloudflare R2 bucket at a <\$1 monthly spend, avoiding costly 3rd-party APIs.
- Implemented a dual Web Worker pipeline to decode, privacy-trim, sample, and grid-assign polylines off the main thread, and use the global PMTiles lookup to compute per-city and per-viewport exploration percentages.
- Designed LRU IndexedDB caching for visited cell sets, processed activity IDs, and geodata, minimizing recompute and re-fetching from the R2 bucket, while also making page reloads up to 25x faster.

TECHNICAL SKILLS

Languages: Go, Python, Swift, JavaScript, SQL, C/C++, C#, Java, R

Frameworks: React, SwiftUI, Flask, Gin, Bootstrap, Playwright, RMarkdown, SQLAlchemy, Apache Spark

Developer Tools: Git, Docker, Google Cloud Platform, VS Code, Vite, JupyterLab, Linux, Bun