Hardik Singh

972-330-1278 | hardikhssingh@gmail.com | linkedin.com/in/hardiksingh-hs/ | github.com/Hardik-Singh | hardiksingh.com

Education

University of Texas at Austin — Austin, TX
Bachelor of Science in Computer Science, Minor in Business — *Aug. 2023 – May 2027*

Experience

Software Engineering Intern — Aug 2025 – Nov 2025
Google — *Seattle, WA*

- Developed a Python-based Gemini debugging assistant for the Fuchsia (fx test) pipeline, adding an intelligent artifact-discovery subsystem that automatically locates, loads, and pre-processes required logs/manifests/traces to enable accurate model-driven failure analysis.
- Built an intelligent log-analysis system that heuristically chunks large logs, extracts stack traces, and summarizes build failures across 10+ programming languages, drastically reducing manual triage time.
- Performed performance analysis of FIDL communication in Rust for artifact transfer and created a reusable benchmarking framework with Python-based visualization tools to compare latency across runs and identify bottlenecks.

Software Development Intern — May 2025 – Aug 2025
Amazon Web Services — *Seattle, WA*

- Added raw query string support to CloudFront (CF) Functions by updating NGINX/Quickstep servers with FlatBuffer schema and serialization changes to meet Fortune 100 client requirements for migration.
- Enhanced Rust-based string parsing for CloudFront Functions using byte-level libraries, achieving 46% faster processing across 700+ edge locations with sub-millisecond latency improvements.

Software Engineering Intern — May 2024 – Aug 2024
John Deere Financial — *Des Moines, IA*

- Designed a high-performance REST API with Python, FastAPI, and GraphQL to stream over 12 million telemetry data points from New Relic via AWS Kinesis; deployed using Terraform with 99.9% uptime.
- Developed lightweight Java classes using Spring and Maven to implement structured JSON logging and traceability in AWS Lambda and ECS pipelines, efficiently capturing over 1M daily logs with 0.32% performance overhead.

Software Engineering Intern — May 2022 – August 2022
Crestron Electronics — *Plano, TX*

- Developed C++ libraries for automated control of Crestron DSP and control systems, enabling direct hardware interfacing and eliminating repetitive command execution across 30+ devices.
- Built a TensorFlow-based object detection pipeline to automate audiovisual device testing, improving validation speed by 330% and reducing manual testing overhead.

Projects

Visetta | *Rust, Tauri, TypeScript, AWS (Lambda, API Gateway, RDS, ElastiCache), Pinecone* — Jan 2025 – Present

- Built a cross-platform AI assistant in Rust + Tauri with OS-native APIs, global hotkeys, and automated MCP server generation from data blobs into customizable domain-specific endpoints.
- Integrated a Pinecone-powered Retrieval-Augmented Generation (RAG) pipeline enabling multi-session memory.

Quantitative Poker Bot | *Go, Python, NumPy, PyTorch* — Feb 2024 – Present

- Built a Go poker engine with Monte Carlo rollouts and concurrent simulation workers achieving 8K+ rollouts/sec.
- Integrated PyTorch RL agents via gRPC for self-play and adversarial strategy evaluation.

Low-Latency Trading Exchange | *C++, Python, Boost.Asio, pybind11, QuantLib, spdlog* — Nov 2023 – Present

- Developed a C++ trading exchange with Boost.Asio async networking, lock-free queues, and a cache-friendly binary heap order book, achieving 5K orders/sec with consistent sub-millisecond latency on an 8-core Apple M2.

Technical Skills

Languages: Rust, C++, C, Python, Java, SQL, Go, JavaScript, TypeScript, R, MATLAB
Frameworks: Spring, React, Node.js, React-Native, FastAPI, Terraform, Django, Express, PyTorch, TensorFlow
Developer Tools: AWS, Azure, Apache, Docker, Kubernetes, PyCharm, IntelliJ, Git, Jira
Libraries: OpenCV, Pandas, NumPy, Matplotlib, SciPy, Keras, QuantLib, Boost.Asio, FIX Protocol