BHARAT KATHI

bharat1031@gmail.com
+1 (510) 945-9684
github.com/bk1031
/in/bk1031

SKILLS

Languages: Go, Python, C++, Dart, Swift, Java, Typescript, SQL

Technologies: AWS, Azure, Kubernetes, Terraform, Docker, PyTorch, Kafka, PostgreSQL, Snowflake

EDUCATION

University of California, Santa Barbara **September 2021 – June 2025**

B.S. Computer Engineering

- Relevant Coursework: Circuits and Systems, Computer Architecture, Computer Vision, Data Structures & Algorithms, Deep Learning, Distributed Systems, Embedded Systems, Machine Learning, Operating Systems

EXPERIENCE

Gaucho Racing (UCSB Formula SAE) • Data Lead **September 2021 – Present**

- Founded and led the Data subteam, building Mapache: an intelligent, real-time telemetry and analytics system for our team's Formula-style electric racecar
- Designed an embedded edge stack with Jetson Orin Nano to decode 200+ sensor signals over CAN and execute real-time ML models for thermal prediction and anomaly detection
- Created a robust telemetry pipeline that caches data onboard in the event of connectivity loss, and uploads it post-run using GPU-accelerated Parquet conversion and cuDF, ensuring zero data loss
- Developed cloud infrastructure using MQTT, AWS, and SingleStore to ingest, store, and analyze high-frequency telemetry for performance diagnostics and tuning
- Built a React + TypeScript dashboard with real-time signal visualizations, trip/lap segmentation, and historical comparisons across drivers and configurations
- Integrated a natural language interface powered by LLMs, enabling engineers to query telemetry insights conversationally (e.g., "Compare battery temperature rise across laps")
- Collaborated closely with vehicle dynamics and powertrain teams to validate sensor data and translate telemetry insights into mechanical design improvements
- Implemented continuous monitoring of system health (Jetson resource usage, connectivity, thermal load), with alerts routed back to the driver via CAN during operation
- This system was the first of its kind in the history of Formula SAE, and has collected over 20 GB (over 1B datapoints) of vehicle data

SingleStore • Software Engineering Intern **June 2024 – September 2024**

- SingleStore is a distributed, high-performance SQL database that unifies transactional and analytical workloads (HTAP) in a single engine
- Contributed to SingleStore's integration into Snowflake via Snowpark Container Services, enabling real-time analytics inside Snowflake's data warehouse
- Improved onboarding by adding preset cluster configuration options in the Python based UI (Streamlit), streamlining setup for new users
- Built and presented a demo simulating a real-time ridesharing app, streaming live driver/rider data and visualizing insights in a React dashboard
- Demonstrated up to 50x lower latency and faster queries compared to native Snowflake workloads, highlighting the performance benefits of SingleStore
- Presented the demo to enterprise customers and at SingleStore NOW, supporting product launch and early adoption

Axiamatic (Greylock-backed startup) • Software Engineering Intern **June 2023 – September 2023**

- Built internal tools with React and TypeScript for Admin UI, an internal tool for engineers to manage and monitor the platform
- Developed new backend services in Python (FastAPI) for the Admin UI, and deployed them on AWS Fargate
- Created a Service Registry Browser to visualize 100+ AWS-deployed microservices and their dependencies, significantly improving visibility across teams and accelerating onboarding for new engineers
- Built an On-call Log system to track incidents, document responses, and notify responsible teams via Slack, reducing on-call resolution time and improving cross-team coordination during outages

Pacific Esports League • Software Engineering Intern **June 2022 – September 2022**

- Built core features for the PEL Portal, a Flutter-based web app used by 1,000+ players to form teams, register for tournaments, and view real-time stats
- Designed a custom CMS for league admins to manage tournaments, streamlining team and player management
- Implemented a matchmaking system to auto-seed brackets, notify players, and collect results, reducing manual labor by more than 20 hours/week
- Set up CI/CD pipelines with GitHub Actions to auto-deploy Go-based microservices to Azure Container Apps

PROJECTS

StorkeCentral · 1,500+ Users **storkecentral.app**

- Created a platform for UCSB students to access important campus information, and scaled to over 1,000 monthly active users
- StorkeCentral provide easy access to dining hall menus, campus maps, bus schedules, and more
- Users can add their friends, share class schedule information, and receive notifications when friends are in class
- Flutter frontend, Go + PostgreSQL + Kubernetes backend

SproutSense **github.com/BK1031/SproutSense**

- Designed and built SproutSense as part of my senior capstone project: a distributed LoRa-based telemetry system for real-time environmental monitoring across large agricultural fields
- Sensor Modules trasmit readings to nearby Base Stations, forming a star topology minimizing node complexity/power and simplifying large-field scaling
- Developed custom PCBs for both Sensor Modules and Base Stations, supporting over 100 modules per base station, communicating over 915 MHz LoRa using Listen-Before-Talk (LBT) to reduce collisions
- STM32-based Sensor Modules measure temperature, humidity, soil moisture, soil nutrition, and light levels at configurable intervals (30 min to 24 hrs)
- ESP32-based Base Stations relay LoRa messages to an MQTT broker, which are then ignested by a Python Flask application where messages are parsed, scaled, and stored in a PostgreSQL database
- An exponential weighted moving average (EWMA) is used at each node to estimate channel activity and probabilistically delay transmissions to avoid collisions
- React + Typescript frontend enables farmers to visualize historical trends and detect anomalies across the field
- Designed for ultra-low power operation and long-term deployment in remote, offline environments, powered by solar with a battery backup lasting over a week

Epic Shelter **github.com/gaucho-racing/EpicShelter**

- Created Epic Shelter, a high-performance, distributed data backup/migration system
- Supports PostgreSQL and MySQL wire compliant databases, as well as SingleStore and Snowflake
- Data is backed up from a source database table as a batch of parquet files, backed up to S3, then ingested into a target database table
- Orchestrator service allows distributing data movement across multiple worker nodes, and monitoring the status of the backup/migration
- cuDF is used to acclerate reading/writing parquet files on GPUs
- Parquet files from S3 are directly ingested into the target database when supported (SingleStore, Snowflake)
- React + Typescript frontend, Go orchestrator, Python backend (cuDF + Pandas)

Sentinel **github.com/gaucho-racing/Sentinel**

- Created Sentinel, a central authentication service for Gaucho Racing (UCSB's Formula SAE team)
- Keeps track of an active member directory, interfacing with our Discord server to sync subteams and roles
- Implemented OAuth 2.0 and OpenID Connect protocols to allow internal tools to authenticate members, as well as serve as an identity provider for external tools
- Automated onboarding/offboarding processes for the team, including managing permissions for Google Drive, GitHub, and even SOLIDWORKS CAD Licenses.
- React + Typescript frontend, Go + SingleStore backend

PUBLICATIONS

Mapache - An Intelligent, Real-Time Telemetry Platform for Formula SAE

- Bharat Kathi, Andrey Otvagin, Jacob Jurek, Austin Chan, Colin Bickel
- Accepted to the 2025 International Telemetering Conference (ITC), to be published in their Vol. 60 Proceedings
- https://github.com/Gaucho-Racing/Mapache/blob/main/itc25.pdf

Real-Time Environment Monitoring for Sustainable Agricultural Practices

- Julia Chan, Max Cohn, Bharat Kathi, Pablo Sandoval Rivas, Kriteen Shrestha, Yogananda Isukapalli
- Accepted to the 2025 International Telemetering Conference (ITC), to be published in their Vol. 60 Proceedings
- https://github.com/BK1031/SproutSense/blob/main/itc25.pdf