

Srikanth K N

6382004136 | srikanthkn10@gmail.com | github.com/Srikanthkn0 | srikanthkn.dev | Bangalore, India

EDUCATION

Alliance University

Bangalore, India

B.Tech, Computer Science and Engineering — CGPA: 8.7 / 10.0

2024 – 2028

- Coursework: Data Structures & Algorithms, Operating Systems, DBMS, Computer Networks, OOP

OPEN SOURCE CONTRIBUTIONS

OpenMS — C++/Python Scientific Computing Library

Feb 2026 – Present

500+ research publications · Used in pharmaceutical & proteomics research globally

- **PR #8825** — Implemented zero-copy NumPy structured array for MSChromatogram via nanobind, eliminating Python/C++ allocation overhead and achieving a **30× speedup at 1M peaks**
- **PR #8728** — Extracted reusable SequenceCoverage API from inline logic; added C++ implementation and pyOpenMS bindings enabling downstream Python analysis without raw C++ access
- **PR #8842** — Extended zero-copy get_peaks_struct() to Mobilogram for consistent no-copy NumPy output across ion mobility data
- Audited Cython 3.5 vs nanobind 3.6 API surface: identified **42 missing classes, 134 regressed methods, 278 classes with differences** — directly shaped the pyOpenMS 3.6.0 migration roadmap
- Discovered pre-release breaking change in EasyPQP (**1,100+ weekly PyPI downloads**) before pyopenms 3.6.0 shipped — traced po.String rejection in convert.py and documented the upstream fix

PROJECTS

Guardrail Gateway | *Python · FastAPI · Claude API · React · PostgreSQL · Scikit-learn*

[Live](#) | [GitHub](#)

- Built a **production LLM safety gateway** that intercepts every prompt before the model runs and every response before the user sees it — blocked content is **redacted, not just flagged**, enforcing safety at both input and output layers
- Engineered a **1,667-pattern rule set** covering jailbreaks, injection, credential leaks, and exfiltration; augmented with an **ML injection classifier** (Grok in prod, sklearn fallback) — validated by **305 tests** including adversarial strings
- Shipped a live dashboard with health monitor, analytics, rule tester, chat inspection panel, and audit logs; **multi-provider routing** (Grok, Gemini, OpenAI, Mock) makes any block or redaction demoable in under 30 seconds

LLM Eval Pipeline | *Python · FastAPI · PostgreSQL · Gemini · GPT-4o · React*

[Live](#) | [GitHub](#)

- Built an **async LLM evaluation pipeline** — upload a CSV of prompts and expected answers, pick a model and prompt template; backend runs **async jobs** scoring each row and persisting full run history in PostgreSQL
- Implemented a **CI quality gate**: regression test suites must achieve $\geq 80\%$ pass rate or the build fails, catching prompt and model regressions before they reach production users
- Wired **four inference paths** (Gemini, GPT-4o mini, Groq, Mock) so local dev and CI work without API keys while production runs real models — same UI and scoring logic across all providers

TECHNICAL SKILLS

Languages: C++ (templates, STL, memory layout, RAII), Python, JavaScript, SQL

AI / LLM: Claude API, Gemini, GPT-4o, Groq — streaming, multi-provider routing, prompt safety, ML classifiers

Bindings: nanobind, pybind11, Cython — zero-copy NumPy, Python/C++ interop, buffer protocol

Backend: Node.js, Express, FastAPI, REST APIs, JWT, PostgreSQL, Socket.io

Tools: Git, GitHub Actions, CMake, Docker, Linux, Sphinx

Concepts: Zero-copy memory, LRU cache, LLM safety & eval, async job queues, API migration, concurrency, DSA

ACHIEVEMENTS

- Merged **3 pull requests in OpenMS** as a 2nd-year undergrad — C++ API design and nanobind zero-copy bindings with a **30× speedup**
- Discovered pre-release API breakage in EasyPQP (**1,100+ weekly PyPI downloads**) before pyopenms 3.6.0 shipped — found by testing real downstream tools against the dev wheel
- GSoC 2026 applicant — OpenMS org, Full Python Bindings Using Nanobind (350-hour Medium–Advanced project)
- **LeetCode Contest Rating: 1500**