

Siddharth Nayak

siddharth97nayak@gmail.com | [linkedin.com/in/siddharth1297](https://www.linkedin.com/in/siddharth1297) | github.com/siddharth1297 | Noida, India

Experience

Qualcomm, Noida | *Backend Engineer* June 2024 – Present
Building microservices to improve Qualcomm's location services.

- Developing microservices for processing 5G network data to achieve 1-meter accuracy of device location using Go, Java, Redis and S3.

Open Futures, New Delhi | *Software Developer* Aug. 2019 – Sep. 2021
Designed and delivered micro-second features and algorithms for in-house low-latency trading system using C++ and Python.

- Increased profit potential by 10%** for *high-frequency automated arbitrage trading* algorithms by revamping trade execution algorithms (in C++ and Python) in collaboration with a team of 2.
- Reduced app startup time to 1/3rd** by porting sequential C++ code to *multithreaded* code.
- Independently, built a web-based *real-time* risk monitoring system that **slashed traders' decision-making time by 95%** using Django, WebSocket, and Redis. Wrote *asynchronous* Python HTTP and WebSocket clients for multiple crypto exchanges (**Full ownership**).

Education

Indraprastha Institute of Information Technology, Delhi Aug. 2022 – June 2024
M.Tech in Computer Science and Engineering *CGPA: 9.0/10*

Institute of Technical Education and Research, Bhubaneswar Aug. 2015 – June 2019
B.Tech in Computer Science and Engineering *CGPA: 9.3/10*

Skills

Areas of Interest: Backend Engineering, Distributed Systems, Databases, Operating Systems

Languages: C/C++, Go, Python, Java, CPython, Shell Scripting, JavaScript, SQL, P4, Dafny

Tools: Git/GitHub, gdb, Valgrind, clang-tools, Docker, Kubernetes, eBPF

Frameworks: gRPC, LLVM, DPDK, Django, Flask, C++ QT

Databases: PostgreSQL, Redis

Cloud Platforms: AWS, GCP

Projects

Fault Tolerant Distributed Key-Value Store | *Distributed Systems* March 2024

- Built a distributed key-value store from scratch using **Python and gRPC**, deployed over **Google Cloud Platform**. It achieved a **significantly low latency of sub-1ms for reads and sub-150ms for writes**, utilising the *Raft consensus algorithm and leader-lease technique*.

Kanva: Lock Free Search | *Concurrent Data Structures* Jan. 2023 – May 2023

- Significantly extended **Kanva**, a non-blocking linearizable learned lock-free search data structure written in C++, by implementing a *lock-free range search* that offers a **throughput of 12MOPS/128 cores**, using a memory-efficient *constant-time snapshot algorithm*.

Argolib: A Parallel Scheduler | *Parallel Programming* Sept. 2022 – Dec. 2022

- Developed a *Fork-Join style parallel programming library and scheduler* for C/C++ programs, offering a variety of work-stealing scheduling algorithms. Additionally, **minimised runtime performance overhead up to 30%** by implementing *trace and replay* mechanism.

SafeC: A Memory Safe C Language | *Compilers* Sept. 2022 – Dec. 2022

- Enhanced memory safety of C programs** by implementing an **automatic memory manager** with a **conservative garbage collector** using the *mark-and-sweep* algorithm. Also, implemented an *LLVM* pass to catch null pointer access.

Publication

Kanva: A Lock-free Learned Search Data Structure [Paper]

Gaurav Bhardwaj, Bapi Chatterjee, Abhinav Sharma, Sathya Peri, and **Siddharth Nayak**
In proceedings of the 53rd International Conference on Parallel Processing – 2024 (ICPP '24)