AARIZ SHEIKH

Mumbai, India | arizsheikh17@gmail.com | **+91 9137871445** Portfolio: aariz.vercel.app | GitHub: github.com/arino08 | LinkedIn

EDUCATION

M.H. Saboo Siddik College of Engineering — *Mumbai, India Bachelor of Engineering in Information Technology | Expected Graduation: 2027*

TECHNICAL SKILLS

- **Languages: Rust** (Tokio, Async), **Java** (Core), TypeScript, JavaScript, SQL.
- **Systems Engineering:** Multithreading, TCP/IP Networking, Memory Management, Linux (Nobara/Arch).
- **Web Technologies:** Next.js 14, React.js, Tailwind CSS, D3.js, WebSockets, Docker.
- **Tools & Cloud:** Git, GCP (Cloud Run), GitHub Actions, Postman, MongoDB.

KEY ENGINEERING PROJECTS

FlashKV | High-Performance In-Memory Database

(Tech Stack: Rust, Tokio, Bytes, TCP) GitHub Repository

- Architected a multi-threaded key-value store in **Rust**, implementing the **RESP protocol** to function as a drop-in replacement for Redis.
- Engineered a **sharded `RwLock` architecture** (64 shards) to minimize lock contention, handling **10,000+ concurrent connections** via Tokio's asynchronous runtime.
- Implemented "Active Expiry" algorithms using background threads to efficiently evict TTL-expired keys without blocking main thread operations.
- Optimized memory usage using zero-copy byte manipulation (`Bytes` crate), reducing heap allocation overhead by 40%.

RustyLoad | Async HTTP & TCP Load Tester

(Tech Stack: Rust, Clap, Hyper, Ratatui) GitHub Repository

- Built a CLI-based distributed load testing tool capable of generating **50,000+ RPS** (Requests Per Second) to stress-test web servers and databases.
- Implemented **statistical latency analysis** to calculate P95 and P99 percentiles, identifying tail-latency bottlenecks in distributed systems.
- Designed a dual-protocol engine supporting both **HTTP/1.1** for web servers and raw **TCP** packets for database benchmarking.
- Integrated a TUI (Terminal User Interface) using `ratatui` for real-time throughput visualization and error rate monitoring.

SketchFlow | Infinite Canvas Whiteboard

(Tech Stack: Next.js 14, Rough.js, FlashKV, MongoDB)

- Developed a collaborative cloud whiteboard featuring an infinite canvas engine using **Rough.js** and **2D matrix transformations** for smooth pan/zoom.
- Integrated the custom-built **FlashKV** database as a high-speed caching layer, reducing diagram retrieval latency by **300ms** compared to direct DB hits.
- Implemented robust cloud persistence using **MongoDB** with optimistic UI updates, ensuring zero-latency user interactions during network syncing.
- Engineered collision detection algorithms for efficient shape manipulation (resize, drag, rotate) within the React rendering lifecycle.

OPEN SOURCE & CERTIFICATIONS

- **Google Cloud Skills Boost:** Earned 9+ Badges including *GenAI*, *Cloud Infrastructure*, and *Secure Networking* (2023).
- **Learning in Public:** Publishing a weekly engineering series on LinkedIn documenting the transition from High-Level Web Dev to Low-Level Systems Programming in Rust.