

Mohammad Talim

Software Engineer

Faridabad, Haryana | hello@mdtalim.me | +91 8929203487 | mdtalim.me | linkedin.com/in/md-talim
github.com/md-talim

Skills

- **Languages:** Java, Go, TypeScript/JavaScript, Python, SQL
- **Backend:** Spring Boot, REST APIs, Redis, PostgreSQL, MySQL, JUnit
- **Frontend:** React, Next.js, Tailwind CSS
- **Tools:** Git, Docker, Linux, GitHub Actions, IntelliJ, VS Code
- **Concepts:** OOP, Data Structures, Concurrency, Distributed Systems, Caching, API Design

Experience

Software Engineering Intern

Jun 2025 – Dec 2025

Codeva

- Built and iterated on core product surfaces (dashboard, profile, landing) for an AI STEM learning platform using **React + TypeScript**, shipping changes via PRs in a multi-engineer codebase
- Improved code quality and team consistency by configuring **ESLint + Prettier** (rules/scripts) and standardizing formatting and linting workflows across the repo
- Collaborated asynchronously with a 15–20 person remote team: resolved merge conflicts, participated in code reviews, and unblocked integration issues during active development

Personal Projects

Redis Server — Built from Scratch ([CodeCrafters](#), #3 Java Globally)

[\[code\]](#)

Java | Virtual Threads | TCP | RESP Protocol | Concurrency

- Hand-implemented the **RESP wire-protocol** parser and **36 commands** (strings, lists, sorted sets, streams, pub/sub, transactions, geo), interoperable with official redis-cli and redis-benchmark
- Profiled with **async-profiler flame graphs**, identified **5 bottlenecks** (unbuffered per-byte syscalls, synchronized console logging, excessive allocations in serialization), and applied targeted fixes — achieving **84K ops/s (70× improvement, within 1.6× of C Redis)**
- Designed a **multi-client concurrency model** using Java virtual threads (one per connection), ConcurrentHashMap-based storage with lazy TTL expiration, and ReentrantLock-driven blocking commands (BLPOP, XREAD)
- Built **master–replica replication** end-to-end (PSYNC, RDB v11 snapshot parsing, command propagation, offset-tracked WAIT) with one-command Docker Compose demo (master + 2 replicas); added **JUnit 5 tests** and **GitHub Actions CI**

Gate — Full-Stack URL Shortener

[\[code\]](#)

Java | Spring Boot | React | TypeScript | MySQL | Docker

- Built a URL shortener with **Spring Boot** and **React** — **JWT authentication**, per-user click analytics with date-range filtering, and **one-command Docker Compose** deployment
- Added an **in-memory cache** on redirects (Caffeine, 10k entries, 10-min TTL) — reduced latency from **~5 ms to <1 ms**; scheduled hit/miss logging for observability
- Built a **sliding-window rate limiter** from scratch — **100 req/min per IP** on redirects, **10 creations/hour** per user, with automatic stale-entry cleanup
- Moved click-event recording to a **background thread** so redirects return instantly without waiting on database writes

Education

B.Tech in Computer Science Engineering

Sept 2024 – July 2027

Rawal Institute of Engineering & Technology

Diploma in Computer Science Engineering

Sept 2021 – July 2024

Govt. Polytechnic College, Indri