

Mateusz ZŁOŚNIK

ABOUT

Backend developer, also into DevOps. Currently building a real-time orbital collision detection system for my bachelor's thesis, scanning 30,000+ tracked objects. I administer a Hetzner VPS hosting my portfolio - Docker, nginx, GitHub Actions CI, and a Prometheus/Grafana monitoring stack.

CONTACT

mateuz204@gmail.com

+48 535 898 860

zlosnik.dev

github.com/Salad109

LANGUAGES

- Polish - Native
- English - C2 (Cambridge certified)

TECHNICAL SKILLS

Languages

Java, Kotlin, Python

Backend & APIs

Spring Boot, Spring Modulith, REST, Protocol Buffers

Databases & Cache

PostgreSQL, MySQL, Redis, Caffeine, Flyway

Infrastructure

Docker, Linux, nginx, AWS RDS, Cloudflare

CI & Observability

GitHub Actions, Prometheus, Grafana

Testing

JUnit 5, Mockito, Testcontainers

EDUCATION

B.ENG. COMPUTER ENGINEERING

6th semester

Politechnika Wroclawska (Wrocław University of Science and Technology)

PROJECTS

SATELLITE CONJUNCTION DETECTOR

Bachelor's Thesis

◇ Live Demo | GitHub

◇ Built a system that checks **all ~30,000 tracked objects in Earth orbit** against each other for potential collisions - **~450 million pairs in under 30 seconds**, roughly **1,200x faster** than the public industry reference system (CelesTrak SOCRATES)

◇ **Validated against the industry reference** (>99% accuracy) and correctly predicts the **2009 Iridium-Cosmos collision** - the largest satellite collision on record - from archived pre-impact data, pinpointing impact time to within 10 milliseconds.

◇ Custom algorithms for spatial lookup and trajectory math bring the full scan from hours down to seconds. Orbit propagation via the Orekit astrodynamics library, automated catalog ingestion from Space-Track.org

◇ **Fully automated CI/CD** - every push runs the full test suite, mutation testing, and static analysis (SpotBugs, Error Prone); merges to master publish a multi-arch Docker image and **auto-deploy to my VPS** over SSH

LIBRARY MANAGEMENT SYSTEM

◇ Live Demo (librarian1 / password123) | GitHub

◇ Full library workflow - book/author/copy catalogs, reservations, checkouts, role-separated interfaces for public, customers, and librarians with Spring Security

◇ **Race-condition-proof** reservation flow using optimistic locking with retry logic, backed by a concurrent integration test. **Testcontainers** for real-PostgreSQL integration tests, Codecov-tracked coverage, **Flyway** migrations, **Caffeine** caching, Swagger docs, and Prometheus/Grafana dashboards

HIGH-PERFORMANCE URL SHORTENER

◇ Live Demo | GitHub

◇ **Performance optimization sandbox** on a deliberately simple problem. Handles **10,000+ req/s at ~2ms redirect latency**, with a benchmark-driven README showing the tradeoffs of each choice

◇ Reversible ID scrambling via modulo arithmetic keeps short codes compact without exposing sequential DB IDs. Java 21, Spring Boot, Redis, **Protocol Buffers**, with Prometheus/Grafana dashboards during load tests

INFRASTRUCTURE

◇ I run a Hetzner VPS hosting multiple portfolio apps behind an **nginx reverse proxy** with Let's Encrypt SSL, Cloudflare DNS, and UFW/fail2ban hardening

◇ Built a **Prometheus + Grafana** monitoring stack tracking uptime, latency, host metrics, and per-container metrics across all services