

Kotov Semen

github.com/andude10 | [linkedin.com/semyon](https://www.linkedin.com/in/semyon) | [+36 20 535 0225](tel:+36205350225)
kotow.dev | formalkotov@gmail.com

Education

Eötvös Loránd University (ELTE)
BSc in Computer Science (GPA 4.6/5.0)

Budapest, Hungary
2023 – July 2026

Skills

Languages Rust, C/C++, Go, C#, Java, TypeScript, Python, Haskell, Prolog
Technologies Git, Svelte, Tokio, Tauri, SQL Server, PostgreSQL, Prometheus, Spring, Hibernate
Infrastructure Azure, Docker, Linux, CI/CD
Other FP, OOP, Distributed Systems, Agile, REST API

Work Experience

BlackRock, Java Software Developer | *Azure, Java, Spring* June 2024 – May 2026

- Engineered high-scale Java backend services within the Aladdin Trading division (IBOR), servicing 200+ clients and handling a daily volume of 1M+ transactions
- Improved latency and fault tolerance of in-house networking protocol (BMS).
- Optimized cache replication in microservices, reducing network messages by 95% (49M to 2.5M)
- Rolled out new telemetry across core trading services, paired with Grafana dashboards.
- Migrated two trading services from Sybase to SQL Server.

Projects

Tonic – Cross-platform spreadsheet ([Releases](#)) ([Blog](#)) | *Rust, Tokio, TypeScript* April 2026

- Desktop spreadsheet app built with Tauri (Rust backend, Svelte frontend), featuring a fully custom spreadsheet engine and UI: formulas, filtering and sorting, undo/redo, tables, and more.
- Implemented and wrote a technical blog post about a parallel formula calculation algorithm in async Rust, achieving a 4x performance increase over the synchronous baseline ([see benchmarks](#)).
- Implemented the TACO graph compression algorithm from a 2023 research paper
- Designed a formula parser using the Chumsky crate, utilizing Flat AST for cache locality.
- Added support for custom scripting, allowing users to write their own formulas in JavaScript

Low-overhead caching library with high hit rates ([GitHub](#)) | *Rust* August 2025

- Implemented S3-FIFO eviction strategy with adaptive sharding for parallelization.
- Achieved up to 73% hit rate and 130 μ s latency with a cache of 1,500 entries over 10,000 elements.

Desktop app for data logger analytics ([GitHub](#)) | *C#* July 2023

- Built a desktop data analysis application to parse and visualize logs captured by company data loggers.
- Implemented the Douglas-Peucker algorithm, reducing number of points on the graph by 360%.

Hobbies

Hiking, films and music: kotow.dev/hobbies