ALEXANDER KRIVOSHLYK

Java Backend Developer

Applied systems · image processing · equipment and AI/ML integration

dravonosec@yandex.ru | Moscow · Open to remote work
+7 (996) 591-21-72 (Russia / Telegram) | +7 (777) 635-07-63 (Kazakhstan / WhatsApp)
krivoshlyk.com | linkedin.com/in/alexander-krivoshlyk

SUMMARY

Java Backend Developer experienced in building backend and desktop applications. I worked with **Java, Spring Boot, and PostgreSQL**, asynchronous workflows, integrations, and large-image processing. I contributed across the full product lifecycle: **from clarifying customer requirements and designing solutions to deployment and operation**.

CORE STACK

Backend	Java, Spring Boot, Spring MVC, Spring Data JPA, Spring Security, REST, WebSocket/STOMP	**Data**	PostgreSQL, Hibernate, QueryDSL, Liquibase, pgvector, SQLite, S3/local storage
Architecture	modular monolith, event-driven workflows, DDD, package by feature/domain	**Integrations**	OpenFeign, AI/ML services, JNA, JNI
Quality	JUnit 5, Mockito, MockMvc, Testcontainers, integration testing, code review	**Delivery**	Gradle, Docker, GitLab CI/CD
Desktop	JavaFX, OpenCV, Apache POI, Apache Commons Math, ChartFX, JFreeChart	**Observability**	Spring Boot Actuator, Micrometer, Prometheus, Grafana
Additional	microservices architecture, Redis, RabbitMQ, gRPC, React, TypeScript, HTML/CSS	**AI tooling**	Codex

EXPERIENCE

Java Developer May 2023 - Present
Axalit

At Axalit, I worked on the backend of a web platform that managed the full core-scanning workflow: image and measurement acquisition, storage, processing, analytics, visualization, and export. I designed the **REST API and domain model** for projects, scanners, layers, samples, measurements, and analysis results.

The persistence layer used **PostgreSQL, Spring Data JPA, and Hibernate**. I managed schema changes with Liquibase and implemented complex filters and queries with QueryDSL. Scanning statuses and other live updates were delivered through **WebSocket/STOMP**, while background operations and task states were organized with Spring Modulith.

I integrated an external AI/ML analysis service through **OpenFeign**, including image submission, retry logic, conflict handling, and result persistence. I also built configurable tables with dynamic columns, filtering, sorting, and Excel export through Apache POI.

One of the key optimizations was separating scanner acquisition from long-running persistence. Running these stages independently **reduced scanner occupancy from approximately 4-5 minutes to about one minute** without a major rewrite of the existing pipeline.

I covered the code with unit and integration tests using JUnit 5, Mockito, MockMvc, and Testcontainers, and participated in code reviews, task decomposition, and two-week Agile sprints.

Technologies: Java · Spring Boot · PostgreSQL · Hibernate · QueryDSL · Liquibase · Spring Security · WebSocket/STOMP · OpenFeign · Spring Modulith · Docker · GitLab CI/CD · Testcontainers

Java Desktop Developer / Technical Lead
Axalit

Before moving into backend development, I spent approximately a year and a half working on several engineering desktop applications built with JavaFX. I developed interfaces, mathematical algorithms, image-processing workflows, and integrations with specialized equipment through JNA/JNI.

On a laboratory instrument project, I led a team of three developers, worked directly with the customer, clarified requirements, designed the application from scratch, and coordinated tasks. I introduced **Spring and Hibernate** to separate responsibilities and selected **SQLite** for embedded storage. Other teams later adopted similar approaches.

While optimizing charts, I found that every JavaFX Popup tooltip created a separate scene: a chart with 8,000 points produced thousands of scenes and excessive memory usage. I replaced them with a single reusable tooltip.

I also designed a shared **Apache POI** architecture for generating DOCX and XLSX reports. The Builder-based approach made new configurations faster to implement and exposed selected settings to users; the reporting system was later adopted by other departments.

Technologies: Java · JavaFX · Spring · SQLite · Hibernate · OpenCV · JNA/JNI · Apache POI · Apache Commons Math · Netty

Detailed engineering case studies and diagrams: **krivoshlyk.com**

ADDITIONAL PRACTICE

- **Insight Platform - pet project.** Designing service boundaries between a Java/Spring application and a Python collector with a protobuf/gRPC contract. Redis and RabbitMQ are configured in the local infrastructure; application workflows are still being implemented.
- **Frontend exposure.** Implemented and maintained individual React modules, layout, and styling. Much of this work was AI-assisted; I do not position frontend as my primary specialization.
- **Observability.** Worked with Spring Boot Actuator/Micrometer and Prometheus metrics in a commercial backend and have practical familiarity with Grafana dashboards.

CODEX

- Use **Codex** for code analysis, task decomposition, implementation, refactoring, tests, and documentation.
- Personally review proposed solutions and final code, with particular attention to architecture, security, and tests.

EDUCATION AND LANGUAGES

- **M.Sc. in Artificial Intelligence Engineering** - Ural Federal University, 2025; data analytics specialization.
- **B.Sc. in Software Engineering** - Ural Federal University, 2022.
- **Languages:** Russian - native; English - B2 upper-intermediate.