

Matej Smitala

Senior Backend Engineer — Java · Kotlin · Spring Boot · Kafka · PostgreSQL

matej.smitala@gmail.com · +421 903 740 392 · Slovakia (EU) · Remote · B2B contract · smitala.dev · github.com/smitalm

SUMMARY

Senior backend engineer with 15 years in production. The last 8 on event-driven microservices carrying high-volume real-time data for a regulated online betting operator in the EU. Java and Kotlin on Spring Boot, Kafka, PostgreSQL. I design, build, maintain and run the services I work on, including production on-call.

TECHNICAL SKILLS

Languages: Java 17–25, Kotlin, TypeScript, SQL, C#

Backend: Spring Boot 3/4, Spring Kafka, Spring MVC, Kotlin coroutines, Node.js, .NET

Messaging & streaming: Apache Kafka, Avro, Confluent Schema Registry, RabbitMQ, ZeroMQ, WebSocket

Data: PostgreSQL — schema and query design for mixed read/write workloads, index and query tuning, table partitioning, JSONB; jOOQ, Flyway

Architecture: event-driven microservices, layered architecture, CQRS read models, transactional outbox, at-least-once delivery, idempotency, high availability, backpressure, monolith decomposition

Observability: Micrometer, OpenTelemetry, Prometheus, Grafana, ELK / Kibana

Testing: automated integration tests in CI against real dependencies — Testcontainers, Grafana k6, Gatling

Delivery: Maven, GitLab CI, Docker, SaltStack

Practice: ADRs, OpenAPI, C4 / Structurizr, PlantUML, docs-as-code

EXPERIENCE

Senior Software Engineer (contract)

2018 – Present

Client: a regulated online betting operator in the EU · Bratislava, Slovakia / remote

- Over the engagement: **several production microservices designed, built, released and run** — JVM and Node, Kafka-backed, most still live and on my on-call rotation from their first deploy. The work below is representative rather than exhaustive.

Monolith decomposition into event-driven microservices (2025 – present)

- On the programme breaking a betting monolith into event-driven Kotlin services on Kafka, with exactly one owning service per entity and every boundary a versioned Avro or OpenAPI contract. Design goes through reviewed architecture decision records.
- Core domain service:** per-entity aggregate in PostgreSQL under ordering guarantees derived from Kafka partitioning rather than distributed locking; transactional outbox for at-least-once publication; per-partition failure isolation and fail-stop retry instead of a dead-letter queue. NFRs: 5 000 msg/s at 100 ms p99.
- Provider feed ingestion:** two Kotlin services ingesting a third-party live data feed over RabbitMQ and normalising it onto Kafka. FSM-driven feed recovery re-syncs from a snapshot on any sequence gap, HA is active/passive via RabbitMQ single-active-consumer, and a declarative rule engine makes onboarding a new market configuration rather than a release.

Real-time push platform — platform owner and main maintainer (2018 – 2025)

- Took over a business-critical real-time platform that had no active maintainer and became its main maintainer. At its centre is a Node.js / TypeScript routing application speaking WebSocket to clients and ZeroMQ to the backend, fanning several classes of live data from ~9 backend systems out to web, mobile web and native clients; a TypeScript client library and a Java + ZeroMQ publisher library sit on either side of it.
- Sustains **~20 000 concurrent WebSocket connections** and **~100 000 messages/s outbound** at production peak, from 2 000–5 000 msg/s inbound — a 20–50x fan-out.

- Reduced cumulative delivery delay under load spikes from **several seconds to tens of milliseconds**, after proposing and building the per-hop metrics and distributed tracing that attributed the delay to specific hops.
- Rewrote most of the routing application into typed TypeScript, which made the performance and reliability work that followed reviewable.
- Built the end-to-end test suite, the load generator and a scriptable test publisher, so failover and failure scenarios run in CI against real containers on every build.
- Ran it as an internal platform for ~9 consuming systems: 20 Prometheus metric families, Grafana dashboards and written runbooks on the routing application, and both integration libraries maintained as versioned public contracts — semantic versioning, changelog, deliberately migrated breaking changes.

Live betting — real-time data path from Kafka to client (2020 – present)

- Part of the programme that redesigned the entire live section of the customer-facing web and mobile clients. I designed how those clients integrate with the real-time push platform: the subscription model for the live offer, the snapshot-plus-delta contract, and the reconnect handling that keeps a client consistent after a dropped connection.
- Implemented most of the code on that real-time path, on both sides of it — the publishing side of a Java / Spring Boot **CQRS read model** that materialises the entire offer from five Kafka / Avro streams into its own PostgreSQL and an in-memory cache tier, and the client-side state engine that applies its snapshots and deltas.
- The read model serves every customer-facing surface over 100 REST endpoints and 30 real-time push channels, and runs shared-nothing: each instance consumes the whole stream and owns a full private copy of the data — no shared database, no leader election, no distributed lock.
- Operated the live path — **thousands of live events a day** — for **several years in production**.

Software Engineer — Softec

2011 – 2018

Bratislava, Slovakia

- Greenfield loan and mortgage process automation for **Tatra Banka** — long-running approval workflows in Java / Spring integrating core banking over SOAP, with an AngularJS frontend. Started as a frontend developer and moved into the backend.
- Led the loan-automation component of a greenfield .NET and AngularJS branch application for **Poštová Banka**.
- Built the Android proof of concept for mobile onboarding with biometric client verification.
- Built an Android app for the non-profit odkazprestarostu.sk.

EDUCATION

Mgr., Applied Informatics — Comenius University, Bratislava

2007 – 2012

LANGUAGES

Slovak (native) · English (professional working proficiency)