

# Marek Kacprzak

BACKEND ENGINEER | NODE.JS, TYPESCRIPT & NEST.JS

Warsaw, Poland

+48 601 199 008 | [kacprzakmarek92@gmail.com](mailto:kacprzakmarek92@gmail.com) | [navitelogger](#) | <https://www.linkedin.com/in/marek-kacprzak-238053218/>

## Personal Profile

Backend-leaning full-stack engineer with 3+ years of hands-on production experience across Node.js, TypeScript, and Nest.js - delivering services that talk to relational databases, third-party APIs, and event pipelines under real load. Comfortable owning the full cycle: from architecture discussion and code, through the test pyramid (unit / integration / end-to-end), to deployment on Kubernetes and AWS Lambda, to investigating production incidents and shipping the fix. Security is not an afterthought - a cybersecurity specialisation at Warsaw University of Life Sciences keeps it as the lens I review my own code through. Track record across Nest.js, Express, Python (FastAPI), Go (Huma, Gin), PostgreSQL, MySQL, Redis, Kafka, and BigQuery, with observability via Prometheus, Grafana, Sentry, and ELK. Currently Head of Software Development and Security for a confidential automation platform and Brand Technical Specialist at IBM. Looking for a backend role with a clear product, strong engineering peers, and room to keep raising the security and quality baseline.

## Education

### Warsaw University of Life Sciences

Warsaw, Poland

Computer Science Engineering Degree, specialisation: cybersecurity

October 2023 - Current

Graduation year: 2027

## Work Experience

### Confidential Client (under NDA)

Warsaw, Poland - Remote

Head of Software Development and Security

January 2026 - Current

- Hybrid engineering and security ownership for an automation platform - ship Node.js / TypeScript backend code (Nest.js, Express) while owning the deployment, observability, and security posture.
- Build and maintain backend services that interact with multiple data stores (PostgreSQL, Redis) and third-party APIs; deploy on Kubernetes across AWS, GCP, and Azure, with AWS Lambda functions for event-driven and batch workloads.
- Tune the observability stack - Grafana / Prometheus for metrics, Sentry for error tracking, ELK for log aggregation - so production issues can be traced from a customer report back to a specific request, deploy, or upstream dependency.
- Run event pipelines on Kafka for high-throughput inter-service messaging; integrate AI-driven code and infrastructure analysis into the CI flow for proactive vulnerability triage and architectural governance across backend, frontend, and platform layers.
- Technologies:** Node.js, TypeScript, Nest.js, Python (FastAPI), Kubernetes, AWS (Lambda), GCP, Azure, Kafka, Redis, PostgreSQL, Grafana, Prometheus, Sentry, ELK, Docker, CI/CD.

### IBM Polska

Warsaw, Poland

Brand Technical Specialist (promoted from Pre-sales Engineer)

October 2025 - Current

- Pre-sales architect across IBM's automation and platform portfolio - translate client requirements into deployable architectures and feed product teams the refinement opportunities surfaced during real engagements.
- Deliver technical accelerators - PoCs, PoTs, workshops, demos, MVPs, and RFP / RFI responses - across automation, integration, and platform offerings.
- Work directly with customers to investigate technical fit, position solutions against competitors, and articulate value, ROI, and trade-offs aligned to their decision criteria.
- Co-lead the technical workstream on a Business Partner engagement (AirTrade) - solution architecture, integration patterns, and joint client conversations.

### OSEC sp. z o.o.

Warsaw, Poland

Full Stack Developer

March 2024 - April 2025

- Owned backend services for an Allegro-seller analytics product - shipped Node.js / TypeScript APIs (Nest.js, Express) alongside Go services (Huma, Gin), all talking to PostgreSQL and Redis under high concurrency on the seller-facing data path.
- Deployed services on Kubernetes (Oracle Cloud) with GitHub Actions CI/CD; ran AWS Lambda functions for scheduled batch jobs and event-driven workloads outside the main request path.
- Built a fully-featured browser extension in TypeScript using the WXT framework, integrating with the platform's APIs to extend analytics directly into the seller's browser.
- Wrote the test pyramid end-to-end: unit tests for services and pure logic, integration tests against ephemeral PostgreSQL containers, and Playwright-driven end-to-end tests for the frontend - PRs blocked on green CI and at least one reviewer.
- Worked directly with sellers when their data looked wrong, traced root causes back into the pipeline, and fed those findings back into product priorities.
- Technologies:** Node.js, TypeScript, Nest.js, Express, Go (Huma, Gin), Angular, PostgreSQL, Redis, AWS Lambda, Kubernetes, Docker, GitHub Actions, Playwright, Jest.

# Projects

---

## Nest.js Analytics API with BigQuery Backend

Warsaw, Poland

Personal Project

2024 - Current

- Self-directed Nest.js + TypeScript backend exposing analytics endpoints over a BigQuery data warehouse - the side project where I keep my SQL-at-scale, query-cost, and API-shape muscles in shape outside client work.
- Strict typing end-to-end with DTO validation (class-validator), modular Nest.js architecture (controllers / services / providers), and OpenAPI generated directly from the codebase.
- BigQuery query layer with cost-aware pagination, parameterised queries against injection, partition-pruning by date, and a Redis result cache for hot dashboards.
- Test pyramid: Jest unit tests for services, supertest integration tests against a Dockerised PostgreSQL for metadata, and a small Playwright suite covering the auth and dashboard flows end-to-end.
- **Technologies:** Node.js, TypeScript, Nest.js, BigQuery, PostgreSQL, Redis, Jest, supertest, Playwright, Docker, OpenAPI.

## Build Management System

Warsaw, Poland

Full Stack Developer & DevSecOps Engineer

January 2026 - Current

- Security-first build orchestration platform built on Node.js / TypeScript - centralised dashboard, queueing engine, and hardened build server with secrets handling baked in from day one.
- Engineered a custom management API with atomic configuration writes, automated backups, and path-traversal validation against file system attacks; implemented HMAC-SHA256 request signatures and environment sanitisation to keep master secrets out of build scripts and activity logs.
- Designed the queueing and worker model to scale horizontally as the number of build targets grows; integrated container builds and deployment hooks behind the same dashboard.
- Hardened the build server with UFW network rules, admin-only traffic isolation, and timeout kill-switches for hung processes.
- **Technologies:** Node.js, TypeScript, Python, Nginx, Docker, UFW, HMAC-SHA256, Linux.

# Skills

---

## Backend & Languages

Node.js (Nest.js, Express), TypeScript, JavaScript, Python (FastAPI, SQLAlchemy), Go (Huma, Gin), API design, distributed services, third-party API integration.

## Data & Storage

PostgreSQL / PostGIS, MySQL, BigQuery, Redis, Kafka (producers & consumers), pgvector, schema design for high-concurrency workloads.

## Cloud & Deployment

AWS (Lambda, ECS / Fargate, S3), GCP, Azure, Oracle Cloud, Kubernetes, Docker, Nginx, Traefik, GitHub Actions CI/CD.

## Testing

Jest, supertest, Playwright, test pyramid (unit / integration / end-to-end), ephemeral containers for integration tests, code review & PR gating in CI.

## Observability & Reliability

Prometheus, Grafana, Sentry, ELK (Elasticsearch, Logstash, Kibana), structured logging, on-call instrumentation, production incident triage.

## Security

Threat modelling, secure API design, input validation, HMAC request signing, secrets handling, supply-chain & container security, OWASP awareness.

## Frontend

TypeScript, React, Next.js, Angular, Tailwind CSS, WebSockets, real-time dashboards, browser extensions (WXT framework).

# Interests

---

## Backend Architecture

API design trade-offs (REST vs. RPC vs. event-driven), idempotency, backpressure, and how schema design choices ripple into latency and operability years later.

## Application Security

Reading public post-mortems on auth, injection, and supply-chain incidents; OWASP guidance and how it maps onto a Node.js / TypeScript codebase.

## Testing & Code Quality

Test pyramid economics, mutation testing, property-based testing, and what to mock vs. what to spin up for real.

# Languages

---

## English

Full working proficiency (C2-level certificate)

## Polish

Native proficiency

## Spanish

Limited working proficiency (B2 DELE certificate)