

Michal Ráček

Senior Software Engineer | Java, TypeScript, Python | 10+ years experience

Email: michal.racek@twfrbr.com | Social: [@krablak](#) | Phone: +420 608 526 904

GitHub:

- <https://github.com/krablak>
- <https://gitlab.com/holly-prototypes>

Profile

Senior software engineer with 10+ years of experience delivering production systems across energy, automotive, healthcare, logistics, and finance. I usually work as a senior developer or architect, taking ownership of projects end-to-end — from requirements and system design to implementation and production rollout.

I build software as part of a team, staying hands-on while helping shape robust solutions. I value open discussion and see every role — from engineering to QA and product — as essential to delivering quality systems.

Alongside commercial work, I build side projects in Swift, TypeScript, and React Native to keep learning, experimenting, and refining how I solve problems.

Core Technologies

- Backend
 - Java (Spring Boot, WebFlux, Reactor, JPA/Hibernate), Python, reactive systems, messaging
- Front-end
 - TypeScript, React, Redux, Zustand, MobX
- Databases
 - MySQL, MongoDB, MapDB, SQL Server
- VCS
 - Git, Mercurial, CVS
- Build & CI
 - Gradle, Vite, Maven, Ant, Grunt, Gulp
 - GitHub Actions, Teamcity, Jira, Azure DevOps etc.

(Comfortable switching tools when the project requires it.)

Selected Projects

2025-2026 - Software Engineer, Avenga s.r.o.

Developed an online banker application for Future Lab, an experimental branch concept by Česká spořitelna focused on new forms of bank–client interaction.

The platform enabled remote consultations with advanced interaction features such as screen sharing and expert call-ins. Built on Azure Communication Services with a Java backend and React frontend, the project involved complex real-time workflows and represented an unusually advanced use of the ACS ecosystem.

Media coverage:

- [Future Lab](#)
- [Česká spořitelna má první nonstop pobočku](#)
- [Podcast: Banka, která nikdy nezavírá. Jak vypadá pobočka, kde bankéřům pomáhá umělá inteligence?](#)

Stack: Java, Spring, Azure Communication Services, React, TypeScript, Zustand

2024-2026 - Software Engineer, Avenga s.r.o.

Developed a feedback and evaluation platform for assessing banker–client interactions. The UI was designed iPad-first to deliver a near-native user experience while remaining fully web-based.

Stack: Java, Spring, Oracle, React, TypeScript, Zustand

2023-2026 - Software Engineer, Avenga s.r.o.

Developed and maintained a task distribution platform used by the largest bank in the Czech Republic to reliably route back-office workloads with high availability and fault tolerance.

The system required extensive integrations across banking systems and custom workflows tailored to individual departments, making reliability and consistency key design concerns.

Stack: Java 21, Spring, Oracle, Elasticsearch, Kafka, React, Redux

2021–2023 — Co-Founder & Developer, Holly Solutions

Designed and built a desktop editor for a proprietary slide format. Started as a technical proof-of-concept to validate porting a mobile app to desktop; now evolving into a production content editor with localization workflows.

Stack: TypeScript, Electron, Cordova

2020–2022 — Co-Founder & Developer, Holly Solutions

Built a real-time scoring and event management platform for tag rugby tournaments running globally. Designed for high concurrency, distributed deployment, and replicated data storage. Backend implemented using reactive Java stack to handle large traffic bursts.

Stack: Java, Spring Boot, WebFlux, Reactor, AWS Aurora, TypeScript, React

2019–2020 — Co-Founder & Developer, Holly Solutions

Designed and implemented a discount-card management platform integrated into an existing CMS. Extended the CMS with a React SPA and custom services implementing domain logic and integrations.

Stack: ASP.NET Core, TypeScript, React, SQL Server

2017–2021 - Developer & Co-Founder - Holly Solutions s.r.o.

Designed and implemented a GDPR metadata management system based on microservices and messaging APIs. Integrated with legacy systems and modernized the UX and frontend stack while preserving compatibility with existing infrastructure.

Stack: TypeScript, React, .NET Core, Node.js, RabbitMQ, SQL Server

2016–2017 — Software Architect, Unicorn Systems

Designed next-generation validation and visualization platform for European energy network data. Handled large-scale datasets (tens of GB daily), heterogeneous inputs, and complex validation workflows across multiple teams and countries.

Stack: Java, Spring Boot, MongoDB, MySQL, messaging

2016–2017 — Software Architect, Unicorn Systems

Maintained and extended an IoT system for remote monitoring and localization of hospital beds. Worked with low-level protocols and Python bindings, improving reliability and maintainability of the system.

Stack: Python, MySQL, embedded communication protocols

2015–2016 — Developer / Architect, Unicorn Systems

Worked on multiple enterprise systems across energy and automotive sectors:

- pricing and configuration engine for automotive models
- messaging and integration layers for enterprise data exchange
- prototypes integrating scientific computation modules into enterprise systems

Stack: Java, Spring ecosystem, Oracle, messaging, integration frameworks

2013 — Developer, Institute of Information Theory and Automation

Developed a decision-support simulation system modeling radioactive pollution propagation.

Stack: Python, CherryPy, JS frontend visualization

2010–2013 — Senior Developer / Architect, Unicorn Systems

Built secure distributed communication platforms for European energy market operators.

Focused on reliability, security, and long-running process orchestration.

Stack: Java enterprise stack, BPM systems, Oracle, Tomcat

Own Projects

- Experiments and prototypes at [my GitHub](#)
- [Micro Garden](#) app for Apple Watch
- Messaging application for OS X
 - Written in Swift 3 and communication based on WebSockets
 - [Server side](#) is pure Java 8 using Undertow server
 - Frontend done using riot.js and server side rendered mustache templates
 - Sources are available [here](#)

Competitions / Awards

- First place in Openshift Winter of Code 2014 with: On-line atmospheric dispersion modelling laboratory demonstrates that running of complex scientific models can be conducted also from the web environment using a small home-made cluster powered by multiple OpenShift gears.
 - Source code with description is available here:
https://bitbucket.org/radekhofman/dss_os-git
- In Top 100 at Koding Global Hackaton 2014 as [PyJunkies Team](#)
 - Result was interactive map of nuclear explosions history. Source code is available at <https://github.com/krablak/global.hackathon>