Vitold Radkevich

Senior Backend & SRE Engineer | Cloud & Platform Systems

7+ years of professional experience

Warsaw, Poland
+48571091688 (Mobile)
radkevichvit@gmail.com
Linkedin: Vitold Radkevich
Website: https://vitold-sys.github.io/

- Certifications: AWS Certified, Microsoft Certified
- Backend & Architecture:
  Java, Kotlin, Spring (Core, Boot, MVC, Data, Security, Cloud); REST, GraphQL, Microservices.
  Designed system and solution architectures, led technical decisions, and delivered multiple projects from scratch to production.
- Frontend:
  React, Vue.js, TypeScript, JavaScript, Backstage, HTML, CSS, Bootstrap — integrated complex frontend applications with backend services.
- Databases & Cloud:
  PostgreSQL, MySQL, MsSQL, MongoDB, OracleDB;
  AWS (EC2, ECS, RDS, S3, Lambda, CloudWatch), Azure, GCP (BigQuery, Cloud Storage, Compute Engine) — designed, deployed, monitored, and optimized production cloud systems.
- DevOps & Reliability / SRE:
  CI/CD (Jenkins, GitHub Actions), Docker, Kubernetes, RabbitMQ, Kafka.
  On-call and incident management (Incident Commander at Allegro), SLO/SLI monitoring, alerting, performance and stress testing, automation of internal platforms.
- Leadership & Delivery:
  Led technical decisions, mentored engineers, coordinated cross-functional teams, improved technical standards, and ensured delivery of production-ready systems.
- Agile & International Experience:
  Worked in Agile teams up to 12 engineers; collaborated with PMs, QAs, and international stakeholders to deliver business-critical solutions.
- Languages:
  English — B2
  Polish — B2

Work experience

> Allegro
> Site Reliability Engineer/Software Engineer
> 1 Oct 2024–up to present (1,4 years)

> EffectiveSoft
> Java Software Developer
> 5 Apr 2022– 1 Oct 2024 (2,7 years)

- ➢ VironIT
  Java Web Developer
  25 Nov 2019–20 Feb 2022 (2,4 years)

- ➢ JSC Bank
  Software Engineer
  15 October 2018–30 June 2019 (9 months)

Education

- ➢ Belarusian State University
  Bachelor's degree, Management information resources
  September 2015 - June 2019

Certificates

- Upper Intermediate English (EF SET 57/100 (B2 Upper Intermediate))
- AWS Certified Cloud Practitioner
- AWS Parther: Accreditation (Technical)
- Apache Kafka Series - KSQL on ksqlDB for Stream Processing!
- AWS Architecting Serverless Solutions
- Stepik Linux

Commercial projects

Project #13: Gatling Enterprise

Roles: Backend & SRE Engineer

Technologies: Kotlin, Spring Boot, Gatling, Kubernetes, CI/CD (GitHub Actions), Testing Tools (ADHOC, SharedGE), Jira, Git

Project description: Gatling Enterprise is an internal platform for performance and load testing of Allegro services. The platform provides scalable execution of simulation scenarios, manages system capacity for multiple teams, and validates critical business workflows, including mobile purchase flows and data center failover scenarios.

The solution also includes backend services responsible for preparing and delivering test data used by Gatling feeders, enabling realistic and repeatable load tests across different environments.

Team size:
4 members (3 programmers, 1 project manager)
Duration: 16 months

Responsibilities:
- Managed user access and simulation scheduling, ensuring fair distribution of

system resources.

- Designed and optimized load test scenarios for system resilience and failover validation.
- Reviewed and validated simulation scenarios from cross-functional teams (Pull Request review).
- Made architectural decisions to improve system efficiency, scalability, and reliability.
- Monitored system performance, configured alerts, and analyzed test results to propose improvements.

Project #12: Internal platform to simulate failures across Allegro's production and staging environments

Roles:	Backend & SRE Engineer
Technologies:	Kotlin, Hexagonal Architecture, Chaos Mesh, Kubernetes, CI/CD (GitHub Actions), REST, MongoDB, Monitoring Tools, Backstage, React, TypeScript, Jira
Project description:	This project is a chaos engineering platform used to simulate failures across Allegro's production and staging environments. The system enables controlled experiments on clusters, services, and databases to assess system resilience and reliability. It supports automated selection of services for random failure injection, network latency simulation, database delay emulation, and Canary env. The architecture leverages a highly decoupled, event-driven approach to enable safe execution of chaos experiments and automatic rollbacks in case of incidents.

Team size:
4 members (3 programmers, 1 project manager)
Duration: 16 months

Responsibilities:	- Designed and implemented core backend functionality for orchestrating chaos experiments across multiple environments. - Developed mechanisms for Canary testing and automated rollback in case of failures. - Ensured data consistency and accurate logging for all maintenance activities. - Collaborated with SRE and development teams to define safe failure scenarios and assess business impact. - Ensured the system could safely run randomized chaos experiments without affecting critical services. - Managed and reduced technical debt, improving code quality, maintainability, and reliability of the system.

Project #11: OutageBot (Incident Management Platform)

Roles:	Backend & SRE Engineer
Technologies:	Kotlin, Spring Boot, REST APIs, Slack API, Kubernetes, CI/CD (GitHub Actions), Monitoring & Alerting, MongoDB, Jira, AI, n8n
Project description:	OutageBot is an internal incident management platform integrated with Slack, designed to streamline incident handling, communication, and post-incident analysis at Allegro. The system automates incident lifecycle management — from detection and

coordinated response to postmortem creation — improving operational efficiency, transparency, and system reliability.

Additionally, the platform supports incident trend analysis and early anomaly detection to proactively identify potential incidents.

Team size:
4 members (3 programmers, 1 project manager)
Duration: 16 months

Responsibilities:

- Designed and developed Slack-based workflows for automated incident creation, coordination, and status tracking.
- Implemented mechanisms for automatic creation of dedicated Slack channels for incidents, enabling structured communication and faster response.
- Integrated the platform with internal monitoring and alerting systems to trigger incident workflows automatically.
- Built functionality for generating postmortems, ensuring consistent incident documentation and knowledge sharing.
- Implemented incident data collection and analysis to identify recurring patterns and improve system reliability.
- Contributed to early incident detection and anomaly analysis based on historical data and system signals.
- Improved incident response processes by reducing manual actions and human error during high-pressure situations.
- Collaborated closely with SREs, engineers, and on-call teams to refine incident handling practices and tooling.
- Ensured high availability, reliability, and maintainability of the platform used during critical system failures.

Project #10: Internal Status Page & Reliability Reporting Platform

Roles: Backend & SRE Engineer

Technologies: Java, Spring Boot, REST APIs, Kubernetes, Monitoring & Alerting Systems, Metrics (Prometheus / internal metrics), MongoDB, CI/CD (GitHub Actions), React, Jira

Project description: An internal Status Page platform providing real-time visibility into the health and availability of Allegro's core systems and services.

The platform aggregates data from monitoring, alerting, and event systems to present a unified, real-time view of system status.

It also serves as a reliability reporting tool, enabling engineering leadership and executives to analyze incidents, outages, and system stability over time.

Team size:
9 members (7 programmers, tech lead, 1 project manager)
Duration: 16 months

Responsibilities:

- Designed and developed backend services powering a real-time status dashboard for core Allegro components.
- Integrated the platform with monitoring, alerting, and event systems to collect and correlate live system health data.
- Implemented logic for tracking outages, incidents, alerts, and system events with historical context.
- Built functionality to browse incidents and outages by week, month, and year, supporting operational reviews.

- Designed and generated reliability reports and summaries for engineering leadership and executive stakeholders (GMV).
- Enabled data-driven discussions around system stability, reliability trends, and operational risks.
- Ensured high performance and reliability of the platform used during incidents and operational reviews.
- Collaborated with SREs, engineers, and stakeholders to define meaningful reliability metrics and reporting formats.
- Improved observability and transparency across the organization by centralizing system health information.

Project #9: Internal platform to manage and track system maintenance windows and change schedules

Roles:	Backend & SRE Engineer
Technologies:	Java, Spring (MVC, Boot, Data), Angular (frontend), REST, Kubernetes, CI/CD (GitHub Actions), MongoDB, Gradle, Git, GitHub, Jira
Project description:	Internal platform to manage and track system maintenance windows and change schedules. Helps SRE teams coordinate changes, reduce incidents, and communicate downtime to stakeholders. Ensures visibility into system operations and planned updates.

Team size:
5 members (3 programmers, 1 project manager, team lead)
Duration: 16 months

Responsibilities:
- Developed backend and frontend for application to track scheduled changes.
- Implemented reliable monitoring and alerting for planned system changes.
- Integrated with internal tools for automated notifications and incident tracking.
- Ensured data consistency and accurate logging for all maintenance activities.
- Collaborated with cross-functional teams to optimize workflows for SRE operations.
- Managed and reduced technical debt, improving code quality, maintainability, and reliability of the system.

Project #8: Platform for purchase flowers (from scratch)

Roles:	Backend developer
Technologies:	Spring, Microservices, Hibernate, Git, JUnit, MongoDB, DynamoDb, MySQL, Stripe, AWS, Docker, Maven, Git, GitHub, Jira, Confluence
Project description:	This project is centered on creating an online platform catering to corporate clients globally, facilitating the seamless purchase of flowers. Situated in the United Arab Emirates, the platform aims to provide a convenient and efficient solution for businesses worldwide to access and acquire floral arrangements for various corporate needs.

Team size:
9 members (7 programmers, tech lead, 1 project manager)

Duration: 4 months

Responsibilities:	• Designed and developed a module for integrating with the Odoo system using microservices architecture.

- Designed and developed a module for integrating with the Odoo system using microservices architecture.
- Built and maintained microservices with Java.
- Configured and managed MongoDB with AWS MongoDB Atlas.
- Integrated AWS SQS for messaging and AWS SNS for notifications.
- Developed RESTful APIs.
- Tested and debugged the integration module.
- Coordinated with cross-functional teams for deployment.

Project #7: Project for assess business risks (from scratch)

Roles: Backend developer/Tech Lead/DevOps

Technologies: Spring, Hibernate, Git, JUnit, MySQL, MailJet, Stripe, AWS, Google API, JasperReports, GoDaddy, Docker, Linux, Liquibase, Maven, Git, GitHub, Jira, Slack

Project description: This project aims to assess business risk and Business Interruption Value (BIV) by incorporating key business parameters. Utilizing the NAICS code, the system employs specific calculation logic and formulas to generate comprehensive reports, offering valuable insights into risk evaluation for businesses.

The platform was developed for the US market, following regulatory and business requirements of American customers.

Team size:
6 members (3 programmers, 1 qa, 1 project manager, 1 ba)
Duration: 20 months

Responsibilities:

- Designed and developed a comprehensive project from scratch, including detailed UML diagrams and database schemas to outline the system's structure and data flow.
- Created and implemented APIs for calculating business parameters, including processing large Excel datasets with over 140,000 records and efficiently storing data in the database.
- Set up and configured AWS services including EC2, ASG, ELB, S3, Route 53, CloudFront, Beanstalk, VPC, and RDS for full project deployment and management.
- Oversaw the setup, maintenance, and support of test and production environments to ensure reliable deployment and operation of the application.
- Led technical discussions and decisions, guiding the development team through architectural and operational challenges.
- Collaborated with developers, QA engineers, and other stakeholders to align on project goals and deliverables.
- Monitored system performance, making necessary adjustments to optimize reliability and efficiency.

Project #6: Cleaning platform (from scratch)

Roles: Backend developer/DevOps

Technologies:	Spring MVC, Spring, JPA, Hibernate, Git, JUnit, MySQL, MailJet, Gradle, Google API, JasperReports, Zapier API, Ionos, Tomcat, Linux, Liquibase, Gradle, Git, GitHub, Jira, Slack
Project description:	The project is a comprehensive platform designed to streamline the process of finding and managing cleaning services for flats/houses in England. Users, both cleaners, and clients, can seamlessly connect through the platform, facilitating the creation and management of cleaning jobs. The system incorporates a franchise model, allowing for scalability and regional expansion. An intuitive admin panel provides centralized control, ensuring efficient oversight and management of the entire operation. Additionally, the implementation of mail notifications enhances communication, keeping users informed and engaged throughout the cleaning service process. The platform was developed for the UK market in collaboration with English stakeholders.

Team size:
6 members (4 programmers, 1 qa, 1 project manager)
Duration: 28 months

Responsibilities:	- Rewrote the project from scratch, focusing on designing the new architecture and hosting it, transitioning from the old system that used Spring MVC with an embedded React app to a separate Spring REST API and a standalone React front end. - Designed and implemented a new architecture, ensuring clear separation between the backend API and the frontend application to improve maintainability and scalability. - Managed the hosting and deployment of the application, including setting up and configuring servers and environments to ensure stable and reliable operation. - Oversaw the setup, maintenance, and support of test and production environments, ensuring that the application runs smoothly in both environments and handling any issues that arise. - Collaborated with frontend developers to ensure seamless integration between the backend API and the frontend application, providing technical support and resolving integration issues. - Provided technical leadership and guidance throughout the project, ensuring adherence to best practices and supporting the development team in achieving project goals.

Project #5: Security in bank

Roles:	Backend developer
Technologies:	Kotlin, Spring, JPA, Hibernate, Rabbit MQ, BitBucket, JUnit, MySQL, Microservices, Jenkins, Sonar, Liquibase, Maven, Git, GitHub, Jira, Slack
Project description:	This project focuses on verifying PKCS#12 certificates for all users and managing access to various systems. The goal is to ensure secure access and system integrity by validating key certificates. The project involved developing a robust backend system to support these functionalities, utilizing modern technologies to ensure reliability and scalability.

Team size:
15 members (10 programmers, 2 qa, 1 ba, tech lead, project manager)
Duration: 6 months

Responsibilities:
- Provided ongoing project support: Addressed and resolved bugs and issues throughout the development and deployment phases, ensuring smooth operation and minimal disruption.
- Implemented new functionalities: Developed and integrated new features to enhance the system's capabilities and meet evolving requirements, improving overall performance and user satisfaction.
- Ensured comprehensive module test coverage: Utilized JUnit to write and execute unit tests, ensuring that all modules are thoroughly tested and reliable before deployment.
- Maintained high code quality: Adhered to best practices in coding, conducted regular code reviews, and used Sonar to monitor and improve code quality.

Project #4: European transportation hub (from scratch)

Roles: Full stack developer

Technologies: Spring, JPA, Hibernate, Rabbit MQ, Git, JUnit, MsSQL, Microservices, Jenkins, AWS, GraphQL, React, Flyway, Maven, Git, GitHub, Jira, Slack

Project description: A logistics and transportation platform developed for the Estonian market, designed to support international cargo flows and cross-border parcel delivery.

The system integrates with external partners (including AliExpress) and optimizes transportation workflows across multiple regions.

Development was performed in an outstaff model in close collaboration with an Estonian-based team.

Team size:
12 members (8 programmers, 1 qa, 2 ba, project manager)
Duration: 4 months

Responsibilities:
- Designed and implemented a parcel delivery module integrated with AliExpress, supporting international cargo workflows.
- Designed and developed a new microservice using Spring and Hibernate as part of a microservice-based architecture.
- Contributed to system architecture decisions, focusing on scalability, performance, and integration reliability.
- Integrated backend services with GraphQL and external systems to enable real-time data exchange.
- Automated build and deployment pipelines using Jenkins, improving release stability and deployment speed.
- Collaborated with backend, frontend, and business teams to align technical solutions with logistics requirements.
- Worked in an outstaff model with an Estonian-based team, aligning architecture and integration requirements with external stakeholders.

Project #3: Social aggregator for condominium

Roles: Backend developer/DevOps

Technologies: Spring, JPA, Hibernate, AWS, Git, JUnit, MySQL, JWT, Stripe, MailJet, Linux, Liquibase, Maven, Git, GitHub, Jira, Slack

Project description: The social aggregator project for condominiums in Dania integrates chats, posts, events, and essential services like cleaning and delivery. It aims to enhance community communication and convenience, providing residents with a unified platform for connecting and accessing necessary services.

Team size:
6 members (4 programmers, project manager, qa)
Duration: 12 months

Responsibilities:

- Migrated the platform to AWS infrastructure, improving system stability and operational reliability.
- Designed and implemented backend features supporting chats, events, payments, and notifications.
- Integrated third-party services (Stripe, MailJet) to support payments and communication workflows.
- Managed and supported production and test environments, ensuring system availability and data integrity.
- Performed database migrations and schema evolution using Liquibase.
- Collaborated with product and QA teams to deliver new features and maintain system quality.

Project #2: Crypto exchanger

Roles: Backend developer

Technologies: Spring, Hibernate, Docker, Git, JUnit, PostgreSQL, JWT, Stripe, AWS, Maven, Git, GitHub, Jira, Slack

Project description: The project involves building a secure cryptocurrency wallet risk assessment system and a crypto exchange platform.
Development was carried out in collaboration with an international team based in Australia, requiring coordination across time zones and close communication during a limited delivery window.
The system focused on transaction security, risk evaluation, and integration with external compliance and payment services.

Team size:
6 members (4 programmers, project manager, qa)
Duration: 4 months

Responsibilities:

- Integrated the platform with external Risk Assessment APIs to ensure secure

cryptocurrency transactions.

- Designed and implemented backend services supporting wallet risk analysis and exchange operations.
- Deployed and operated services on AWS using Docker-based environments.
- Ensured system reliability and security through monitoring, testing, and controlled deployments.
- Collaborated with cross-functional teams to align technical implementation with security and compliance requirements.
- Collaborated with an international team (Australia) across time zones, aligning technical decisions and delivery within tight deadlines.

Project #1: Transportation in bank

Roles: Backend developer

Technologies: Spring, JPA, Hibernate, Git, JUnit, MySQL, Jenkins, Sonar, OracleDB, Python, Java, Bash Scripting

The project focuses on optimizing the internal transportation of documents within a bank's offices. This initiative aims to enhance efficiency, reduce delays, and promote a seamless workflow in handling crucial materials between different departments.

Project description:

Team size:
10 members (6 programmers, 1 qa, 2 ba, project manager)
Duration: 8 months

Responsibilities:
- Migrated the system to a new database, improving data consistency and system performance.
- Reworked core modules to support new business logic and internal workflows.
- Developed and maintained backend components using Spring and Hibernate.
- Automated build and quality checks using Jenkins and Sonar.
- Supported production systems by analyzing incidents and resolving system-level issues.