

Maksimov Danil

SOFTWARE ENGINEER · RUBY ENGINEER

✉ danil.maximov2000@gmail.com | 📱 DanilMaximov | 📺 nerbyk

Summary

Starting my software journey in 2019 as a student at a Czech university, I quickly shifted to a self-taught Ruby developer upon entering the commercial sector. Over the years, I've embraced a T-shaped development approach. I've contributed to key applications across startups and international enterprise environments. For the past 2.5 years, my focus has been in R&D engineering for the enterprise GRC product. Here, I've overseen large-scale Rails monoliths, managed their associated AWS Lambda microservices and microfrontends, and held end-to-end responsibility from production monitoring and deployments to incident resolutions and new feature development.

Work Experience

Flatstack LLC

SOFTWARE ENGINEER

Oct. 2021 - Present

Warsaw, Poland

As a Software Engineer in a dynamic consulting team integrated into a 200+ SE's R&D department of Diligent Corporation, contributed to the development and support of Governance, Risk, and Compliance (GRC) product.

Initially, our team collaborated closely with the product owner company, contributing to all Ruby solutions as part of several R&D brigades on client side. Later, during the company's acquisition and ownership transition between R&D departments, our team was granted product ownership. Leveraging our extensive expertise, we handled the smooth transfer of ownership to the new R&D department with a to new brigades.

Demonstrated flexibility and agility by adapting to a different R&D office under changing company structure and processes.

- **Upgrade EOL RoR**
 - Successfully executed the End of Life (EOL) upgrade of the core Rails monolith (>1kk LOC just in the ./app/**/*.rb).
 - Resolved dependencies & deprecations, addressed flaky test issues.
 - Transitioned from Ruby 2.5 to 2.6 and Rails 5.2 to 6.0.
 - Seamlessly updated internal gems to version 2.7.
 - Later was achieved smooth EOL transitions in other departments through that work.
- **Year API Restructure Initiative**
 - With team of 4, refactored over 15 database resources with dozen of implicit dependencies.
 - Delivered new UI and API functionality across GRC Audit submodules.
 - Worked on database restructuring to improve normalization followed by data migration with a focus on backward compatibility for multiple API versions and microservices.
 - Participated in post-release monitoring and debugging.
 - Led after a smaller-scale initiative focused on restructuring the several resource.
- **Service & Technical Debt Work**
 - Addressed production bugs related to extended SQL queries for specific clients, achieving latency reductions from 40-50s to <1s.
 - Successfully mitigated a critical end-of-day memory overflow issue, ensuring continued application stability.
- **Monitoring and Deployment Duties**
 - Conducted weekly monitoring duties across multiple roles (staging, production, secondary production) utilizing tools such as DataDog, CloudWatch, and AirBrake.
 - Orchestrated the 2-day-cycle production deployment of multiple applications across almost all public and US GOV AWS regions.
 - Contributed to deployment, resource management, and proactive measures, collaborated with the DevOps brigade regarding infrastructure resources bumps. Additionally, was reducing errors and alerts in DataDog and Airbrake.
- **Continued & Championed CKEditor Migration**
 - Continued the migration of CKEditor following the downsizing of the original team of 4, navigating through its EOL phase.
 - Managed the update and integration of a new custom Ruby gem, introducing an npm package to applications.
 - Orchestrated a phased UI migration plan using feature flags, directed comprehensive front-end test coverage & custom CoffeeScript plugins overwriting.
- **UI/UX Migrations**
 - Collaborated with a 4-member team to overhaul a Rails HTML ERB module, migrating it to a new React micro-frontend application.
 - Markedly accelerated and enhanced the UI/UX experience, resulting in improved overall usability.
- **DSL Implementation and Code Enhancements**
 - Implemented a lightweight DSL within an internal shared gem to optimize the lazy loading of ENV variables for Doctor rake tasks, ensuring compatibility with various triggering AWS services & invocation methods.
 - Continuously contributed to refactoring Ruby codebase, improving both gems and portions of the Rails monolith sections.
- **Technology Stack**
 - Primarily worked with the monolithic backend using Ruby and Rails, and utilized AWS-CDK & Terraform for infrastructure needs when necessary. Also had experience with front-end technologies including SSR/CSR React, and occasionally worked with CoffeeScript and jQuery.

Keywords:: AWS (ECS, EC2, CloudWatch, IAM, Lambda, S3, RDS, CodePipeline + CodeBuild, API Gateway, SQS, SNS, EventBridge, DynamoDB, SSM Parameter Store, StepFunction), Terraform, TypeScript (AWS-CDK), React.js (SSR/CSR), CoffeeScript, jQuery, Ruby on Rails)

My first professional experience involved working in the office environment of a Czech **B2B startup** specializing in **e-commerce price optimization** (Disivo – Ecommerce price optimization platform).

In this project, I was part of **team of 6 developers**, among whom was a strong leader who made significant contributions to the Ruby open-source community.

The project itself initially consisted of a **Rails monolith with Vue.js 2**, client-side later transitioned to a pre-release version of **Hotwire with TurboLinks**.

- **New Features Development:**
 - The main part of the work consisted of **implementing new business logic** managing the **entire development cycle** across server and client sides.
 - Conducted requirement analysis amidst evolving stakeholder demands, streamlining our development process.
 - Resulted in enhanced efficiency and productivity, contributing to the **team's overall delivery increase of up to 60%**.
- **Microservice Development & API Integration**
 - **Integrated a partner API** within a Ruby microservice for systematic data collection and analysis.
 - Overcame **data retrieval limitations** and **query constraints** by implementing a **caching system** and **organizing parallel API fetching**, achieving speeds of up to 200 RPS (api limit...).
- **Optimization and Bug Resolution:**
 - Promoted to handle service tasks, concentrating on **optimizing servers, databases, and queues** resources utilization.
 - Additionally, was **addressing urgent production bugs**.
 - Successfully **refactored several key modules** that had remained **unchanged since the prototyping stage**.
- **Vue.js Migration to Hotwire**
 - Contributed to migration of client application from **Vue.js 2 to Hotwire**. I was **responsible for migrating 40% of UI modules**.
 - Despite Hotwire's pre-release status, seamlessly transitioned the entire application to the new client by July 2021.
- **Mentorship and Skill Growth:**
 - **Gained invaluable mentorship under Janko Marohnić**, known for projects like shrine, test.vim, and rodauth-rails.
 - Elevated to the role of secondary lead developer, assisting Janko in overseeing project development.

Keywords: Ruby on Rails, Hotwire, Sidekiq, Redis, MariaDB, Elasticsearch, Vue.js

Russian-Speaking Community in Brno

FULL STACK DEVELOPER

Brno, Czech Republic

Oct. 2019 - Nov. 2020

Collaborated closely with the Russian-speaking community in Brno, contributing to the development of various small-scale Ruby on Rails full-stack applications to address community needs.

- **Visa Status Scraper:** Developed an **API service for scraping the website** of the Czech Ministry of the Interior to track visa status updates. This solution aimed to address the issue of the **ministry not notifying foreigners** about visa status changes.
- **Foreigner Support CRM:** Created a comprehensive **CRM system tailored for local organizations offering support** to foreigners. This involved developing a **RESTful Ruby on Rails API** (utilizing ActiveRecord and PostgreSQL) and a **Telegram client in Ruby**.
- **Telegram Bots for Students:** Developed multiple **Telegram bots to assist students and applicants**. These bots employed a hierarchical role system (using tools like cancancan and aasm), **delayed job functionality with Sidekiq**, and integration with diverse APIs, including private university APIs.

Keywords: Ruby, Ruby on Rails, Redis, PostgreSQL, React.js

Education

Brno University of Technology

FACULTY OF INFORMATION TECHNOLOGY

Brno, Czech Republic

Sep. 2019 - Oct. 2021

- Dropped my studies at third semester because of COVID-19

Contributions

In my journey to master Ruby, I invest my free hours in exploring its depths beyond typical business applications. Engaging in personal projects, I immerse myself in the language's complexities, seeking to understand its ultimate capabilities. This often involves delving into open-source code.

Two standout projects exemplify this commitment:

Non-Blocking FibersPool TCP Server: This project explores native asynchronous operations facilitated by Ruby 3+ Fibers scheduler interface, enabling intensive server/client interactions over TCP around in-memory storage. Notably, the server operates single-threaded while handling requests from multiple clients simultaneously. Additionally, it includes benchmarks setup, achieving 3 million read/writes per minute (cpu specific).

anyway).

Playground Gem for Ruby 3.2+: Playground gem for Ruby 3.2+ features initially designed to provide lightweight DSL for FaaS (Function as a Service) computing instances but mostly used as dynamic testing playground for cutting-edge features and experimental techniques within the Ruby ecosystem. Inspired by Jeremy Evens gems, aws-sdk v3 gem structure & rails/activesupport.