



Vladimir

Software Engineer | Scala, Haskell, Elm, TypeScript

7 years of software engineering experience,
3 years in fintech.

Worked on **7** backends, **10** web apps,
and **1** mobile app.

Skills

Technical Expertise

Responsibilities	Research and analysis, system design, engineering, code review, mentorship, technical interviews, running meetups and reading groups
Scala Backend	Scala, ZIO, Cats, Akka, scodec, fs2, decline, zio-test, scalatest, scalacheck, zio-http, akka-http, circe, chimney, flyway, enumeratum, Finagle
Haskell Backend	Haskell, mtl, lens, conduit, servant, postgresql-typed, haskell-to-elm, relude, hspect, QuickCheck
Elm Frontend	Elm, elm-ui, elm-charts, remotedata, elm-review, elm-test
TypeScript Frontend	TypeScript, fp-ts, io-ts, Apollo, Emotion, Jest, Redux, Saga, RxJS, socket.io, Ramda, Sanctuary, React, SCSS, Next.js, Angular
Blockchain	Bitcoin, Ethereum, Cardano, Plutus
Databases	PostgreSQL, MySQL, Redis, MongoDB, S3
Streaming	Pulsar, Kafka
Testing	Cypress, K6, Percy, testcontainers
Authentication	Zitadel, Keycloak
Infrastructure	Terraform, Kubernetes, Docker, Nginx, Airflow, GitHub Actions, GitHub Container Registry, GitLab CI, DigitalOcean
Monitoring	Grafana, Loki
API Design	REST, GraphQL

Communication

Responsibilities	Team leadership, presenting, planning, issue resolution, stakeholder communication, providing feedback, release management, documentation
Agile software development	Scrum, Kanban, Shape Up
Tools	GitHub Projects, Jira, Confluence, Figma, Draw.io, Notion, Trello
Values	Collaboration, transparency, accountability, proactivity, empathy

Specializations

Software Engineering
Acceptance Testing
Load Testing
DevOps
Mentoring
Team Leadership

Languages

English (C1)
Russian (native)

Experience

Data Engineer

[Samokat.tech](#)

09/2023 — 04/2024

Project

Samokat.tech develops real-time retail solutions, including the Samokat food delivery app and other marketplaces.

Responsibilities

Engineering, code review

Key achievements

I developed data pipelines using platform tools (near real-time), set up monitoring and alerting systems to ensure the reliability and availability. Improved coding standards for a Python codebase, improved the documentation. I also organized a weekly book reading club.

Results

Empowered business teams with reliable and timely data, leading to more informed and effective decision-making.

Engineering Team Lead

[Wolf](#)

08/2021 — 03/2024

Project

A full-stack web application designed for trading on the Binance crypto exchange. It incorporates a trend-following trading algorithm, an exchange simulation engine for backtesting, an exchange API client for live trading, and a web application for inspecting, debugging the trading algorithm, and displaying statistics. The project consists of approximately 17,000 lines of Haskell code and 10,000 lines of Elm code. During a technical interview, I can provide a demonstration of the project running and highlight relevant sections of its source code.

Responsibilities

Design, technology strategy, project management, leading, mentoring, engineering, testing, code review, devops.

Key achievements

The team employed various testing techniques, including snapshot testing, property-based testing, and case-based testing. Our approach included schema-first typesafe SQL queries to PostgreSQL. Additionally, we utilize the Haskell-to-Elm library to generate Elm code from Haskell. Also, we have a strong focus on the absence of partial functions. This technological combination enables us to catch most regressions at compile time.

Results

The application was developed, and the trading algorithm outperforms the "buy and hold" strategy on average, on high market cap asset pairs.

Fullstack Engineer

[Tian](#)

07/2023 — 02/2024

Project

A real-time flight tracker with flight history, on a 3D Earth map, with elevations combined with textures.

Responsibilities

Engineering, code review, devops

Key achievements

Helped the project team in various ways: to develop an Elm application, set up development tooling, develop build scripts, improve documentation, set up aggressive caching for the client, and suggested performance improvements for loading of large amounts of the elevation data, which was the major blocker for the project.

Results

A web application was developed and available for users. The combination of its features positioned the application as a unique offering in the market.

Backend Team Lead

Swift Invention

08/2022 — 05/2023

Project

Swift Invention is an outsourcing company, And I was involved in 2 projects: 1. An application for internal users in the compliance industry. 2. Corporate learning software tailored to reduce operational risk for high-performance teams.

Responsibilities

Leading, mentoring, technical interviews, engineering, testing, code review

Key achievements

As a team lead, I managed two projects, overseeing the implementation of new features, writing tests, and addressing bugs. Additionally, I mentored new developers, predominantly with a Java background, to familiarize them with Scala. I also organized and led a weekly Scala book reading club.

Results

A web application was developed and is used by users. Team efficiency was improved through effective mentoring, resulting in timely delivery of projects.

Frontend Engineer

Pamir

05/2020 — 12/2020

Project

Real estate portal and marketplace.

Responsibilities

Mentoring, engineering, testing, code review

Key achievements

I developed a web application that employs server-side rendering and implemented comprehensive unit test coverage. I containerized the application using Docker and established continuous integration (CI). Additionally, I provided mentorship to the second frontend developer who joined the team later. Also I organized and led a weekly reading club.

Results

Developed a web application, improved operational efficiency, improved the team's capacity and expertise.

Backend Engineer

Eldis

10/2019 — 12/2019

Project

Eldis develops software for maintaining registers of shareholders, mutual funds, holding meetings of shareholders and other related areas. The company has their own binary document format containing instructions for filling out table documents, originally developed in C++. I was working on a modern and reliable Scala implementation of this format.

Responsibilities

Engineering, testing.

Key achievements

I developed a declarative decoder for the internal binary document format, ensuring comprehensive test coverage, including property-based testing, and streaming (for expectedly large files). Effectively collaborated with other teams to ensure the correctness of the Scala implementation.

Results

The Scala implementation of the binary document format was successfully integrated into a new larger project, expanding the company's technological capabilities.

Fullstack Engineer

Neolab-Nsk

01/2019 — 09/2019

Project

NDA.

Responsibilities

Engineering, testing.

Key achievements

I implemented new functionality in existing web applications, addressed defects, and developed new applications and microservices, ensuring coverage with unit and integration tests.

Results

Contributed to the successful delivery of projects and achieving business objectives. Comprehensive testing and defect resolution resulted in more stable and reliable applications, maintenance costs.

Frontend Engineer

SocialSweet Inc

08/2018 — 01/2019

Project	Sweet app, a combination of a loyalty platform, a social network, and an online store.
Responsibilities	Engineering, testing, code review
Key achievements	I focused on tasks related to front-end business logic. I actively contributed to enhancing code quality by covering existing code with unit tests and fine-tuning them. This effort resulted in the successful integration of tests into the CI pipeline, effectively preventing defects associated with unsuccessful Git branch mergers in the extensive codebase. Also I shared knowledge and best practices with the team, contributing to the overall improvement of the development process.
Results	Improved operational efficiency, reduced manual effort and errors, increased the reliability, enhanced user satisfaction, boosted team productivity.

Frontend Engineer

Allmax

11/2017 — 08/2018

Project	Savl, a mobile application, cryptocurrency wallet with support for several cryptocurrencies.
Responsibilities	Engineering, testing, code review, mentoring, meetups.
Key achievements	My primary focus was on the data layer within the mobile application. Applying principles from functional programming and software design, I implemented changes that enhanced the application's modularity and fault tolerance, effectively preventing crashes due to exceptions or unexpected behavior from external services. Furthermore, these modifications allowed for the seamless enabling and disabling of support for individual cryptocurrencies, coupled with the implementation of comprehensive unit testing. Also, within the company, I conducted several presentations on functional programming, and helped new developers to onboard.
Results	Improved stability, reduced crashes, enhanced user satisfaction, improved development efficiency.

Education

Mastering Haskell Programming

Packt, 2019

Functional Programming in Haskell, part 2

Stepik, Computer Science Center, 2019

Functional Programming in Haskell, part 1

Stepik, Computer Science Center, 2019

Computer Science Summer School, Theory of Programming Languages

Novosibirsk State University, 2019

Maintenance of computer equipment and computer networks

Novosibirsk Aviation Technical College, 2004 — 2008

Open Source Contributions

elm-modular-grid

An Elm library. Responsive modular grid layouts for Elm.

Elm, elm-ui, elm-land

elm-package-insights

A CLI that scans the Elm package registry and detects broken packages.

Haskell, mtl

servant-to-elm-example

An example full-stack web app, constructed in a code-first, typesafe, and functional way. The servant-to-elm tool generates Elm types, decoders/encoders from Haskell types, and Servant definitions. This not only detects regressions during compile time but also provides customizable Elm functions for fetching data from the server.

Haskell, Elm, servant-server, haskell-to-elm, servant-to-elm

mu-graphql-example-elm

An example full-stack web app, developed with a schema-first, typesafe, and functional approach. Leveraging the mu-haskell library, it performs type checking of Haskell code against GraphQL schema during compile time. Additionally, the elm-graphql library generates Elm types, decoders/encoders from GraphQL schema. I revamped the Elm frontend and made minor adjustments to the Haskell backend, and uncovered some bugs in the mu-haskell library itself in the process.

Haskell, Elm, GraphQL, mu-haskell, elm-graphql

Translation of the Mostly Adequate Guide to Functional Programming in JavaScript

Translated the Mostly Adequate Guide to Functional Programming in JavaScript into Russian. The book introduces the functional programming paradigm and outlines a functional approach to JavaScript application development. I joined the translation effort when progress was at 60%. Collaborating with a friend, we refactored each previously translated chapter before completing the translation.

JavaScript, Ramda