Vladimir Logachev

Fullstack developer, FP enthusiast.
Remote (Novosibirsk, Russia)

logachev.dev@ya.ru

Site: https://logachev.dev
GitHub: https://github.com/vladimirlogachev
Telegram: https://t.me/vladimirlogachev
Twitter: https://twitter.com/logachev_dev
LinkedIn: https://www.linkedin.com/in/vladimirlogachev

Bio

I prefer functional languages that implement strict static typing. I actively use Scala, Haskell, and Elm, but am also ready to tackle any other typed functional language.

I associate the success in my career with FP, so I devote a lot of time and attention not only to code but also to people. My mission as a developer is to make functional programming deliver value to both companies and individual specialists.

I'm a big fan of meetups and reading groups, which I run at my workplaces from time to time, and also I consider pair programming and pair testing to be an effective practice.

In programming, I prefer not to rely on intuition (which, I believe, is usually based on previous experiences and tends to fail in unprecedented situations), but instead, read books well in advance.

Although most of my commercial experience is in front-end development, I am looking for an opportunity to specialize in the backend and am not interested in job offers that involve doing tasks using JS / TS.

Skills and usage experience

Haskell

Concepts: Monads, applicatives, monad transformers
Libraries: mu-hakell, postgres-typed, aeson, parsec, transformers
Language options I'm familiar with 😄: TypeApplications, TypeOperators, PartialTypeSignatures, TupleSections, DeriveFunctor, StandaloneDeriving, QuasiQuotes, OverloadedStrings, LambdaCase, MonomorphismRestriction

Scala

FP: cats-core, cats-effect, fs2, scala-parser-combinators
Testing: scalatest, scalacheck, specs2
Other libraries: scodec, akka, akka-http, akka-stream, scala-parser-combinators

Elm

Concepts: Tasks, Ports, JSON encoding/decoding, Browser API Interop (Websockets)
Libraries: elm-css, elm-graphql, elm-ordering, elm-units, elm-dropbox, elm-crypto-string

Other

Databases: PostgreSQL, Redis, Clickhouse, MongoDB
Infrastructure and tooling: Docker, Dhall, GitHub Actions
APIs: GraphQL

Notable contributions

FP Specialty

https://t.me/fpspecialty_ru

I run this FP reading group for russian-speaking users. During COVID lockdown we discuss PF books and courses remotely, on a weekly basis, but in the past it used to be an offline group.
Reading group

Russian translation of the Mostly Adequate Guide to Functional Programming in JavaScript

https://github.com/MostlyAdequate/mostly-adequate-guide-ru

The book introduces the reader to the functional programming paradigm and describes a functional approach to developing JavaScript applications. The translation was initiated by Maxim Filippov and stopped at 60%. Then me and Sakayama joined the translation, refactored every chapter translated before us and then finished the translation.
JavaScript

higherkindness/mu-graphql-example-elm

https://github.com/higherkindness/mu-graphql-example-elm

I rebuilt an Elm example, which serves as an illustrative frontend for the mu-haskell library (demonstrating its GraphQL capabilities).
Elm, GraphQL

higherkindness/mu-haskell

https://github.com/higherkindness/mu-haskell

I made minor changes to the example project (which should be treated as a part of the documentation), and also helped to discover couple of bugs.
Haskell, GraphQL

Education and courses

Mastering Haskell Programming

https://www.udemy.com/certificate/UC-DRMAMOQ5
Packt, 2019

Functional Programming in Haskell, part 2 (certificate with honors)

https://stepik.org/cert/207739
Computer Science Center, 2019

Functional Programming in Haskell, part 1 (certificate with honors)

https://stepik.org/cert/196007
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

Experience

Pamir, frontend developer 05/2020 — 02/2021

Developed a web application, which utilizes server-side rendering and covered it with unit tests. Packaged everything in Docker and set up CI. I also mentored the second frontend developer who joined the team later.
Frontend: TypeScript, React, Next.js, GraphQL, Apollo, FP-TS, Emotion, Jest Infrastructure: Nginx, Docker, GitHub Actions

Eldis, software engineer 10/2019 — 12/2019

https://eldis.ru
I developed a declarative decoder for the internal binary document format, covered it with tests, including property-based testing.
Scala, scodec, cats, fs2, decline, specs2, scalacheck

Neolab-Nsk, fullstack developer 01/2019 — 09/2019

I implemented new functionality in existing web applications, fixed defects and developed new applications, and microservices, covered them with unit tests and integration tests.
Frontend: TypeScript, React, Redux, Saga, RxJS, FP-TS Backend: TypeScript, Node, Redux, Saga, RxJS, Redis, Lua, Mongo, PostgreSQL, Clickhouse, Docker

SocialSweet Inc, frontend developer 08/2018 — 01/2019

https://sweet.io
Sweet's product is a loyalty platform, social network and online store. I performed tasks related to business logic at the front end and was engaged in covering the existing code with unit tests and tuning them, thanks to which the tests were launched using CI pipeline, and the defects associated with an unsuccessful merging of Git branches in a huge codebase really began to be prevented.
TypeScript, Angular, RxJS

Allmax, frontend developer 11/2017 — 08/2018

https://apps.apple.com/ru/app/savl/id1369912925
I worked in the Savl project — this is a mobile application, wallet with support for 6 cryptocurrencies. I was responsible for the data layer in the mobile application. I applied everything that I learned from books about functional programming and software design, and also completely covered the business logic with tests, as a result of which the application became fault-tolerant and modular, that is, it stopped crashing due to exceptions or unexpected behavior of external services, and allowed to enable and disable support for individual cryptocurrencies at any time. Also inside the company, I made several presentations on functional programming.
JavaScript, Flow, React Native, Redux, Saga, Ramda, Sanctuary, Socket.io