Ludvík Prokopec

Highly skilled and innovative Fullstack Web Developer with advanced expertise in utilizing modern technologies. Seeking challenging opportunities to contribute my technical skills and creative problem-solving abilities in a dynamic team environment.

Work Experiences

SignageOS - Fullstack developer (2022 - present)

Fullstack development of a CMS system used to manage content running on screens such as digital signage. Building GraphQL APIs using tools like Nexus and Pothos. Creating a complex user interface in React and TypeScript. Creation of modules on the backend side, CMS and the module itself displayed on the screen. Orientation in a large code base structured as a monorepo. Knowledge of development tools such as Turborepo, Babel, Jest, Yarn and PNPM. Advanced knowledge of Prisma ORM, integration tests and working with larger PostgreSQL databases. Working with multiple environments such as development, stage and production.

GraphQL, Nexus, Pothos, React, Next.js, Codegen, Urql, Prisma, TypeScript, Yup, Tailwind, Styled Components, Formik, BullMQ, Turborepo, GIT

NETVAREC - Frontend developer (2020 - 2022)

Frontend development for complex websites with an administrative interface created in Contember. Creating websites with authentication or Stripe payments. Development primarily with the Svelte/Sveltekit framework and TypeScript. Fetching data using GraphQL on the client side. Creating styles using the Tailwind framework and preprocessors like SCSS and SASS.

Svelte, Sveltekit, GraphQL, GraphQL houdiny, TypeScript, SCSS, SASS, CSS, Tailwind, Contember, GIT

Education

Faculty of information technology - CTU (Bachelor’s degree) - (2021 - 2024)

Study of web engineering, especially semantic web, modern data formats such as RDF or JSON-LD or advanced algorithms and data structures. Knowledge of distributed systems and databases such as ElasticSearch and ElasticStack or Hadoop. Fundamentals of web and multimedia search algorithms and structures such as page rank. Knowledge of various databases such as PostgreSQL, Redis, MongoDB, Neo4j or Apache Cassandra. Graph theory and graph related algorithms such as BFS, DFS, Dijkstra or Bellman ford. Practical and theoretical knowledge of optimizations and algorithms and their proofs, time and memory complexity. Writing code mostly in C/C++ in combination with multi-threaded programming. Experiences with modeling domain and database models using UML and OntoUML.

Algorithms, Graph problems, Modern data formats, Semantic web, Dynamic programming, Multi-threaded programming, Data structures, Distributed systems, PostgreSQL, MongoDB, C/C++, UML/OntoUML

Faculty of information technology - CTU (Master’s degree) - (2024 - present)

Study of web engineering. Knowledge of middleware architecture and microservices. API design and composition of web services by Enterprise Service Bus, API Gateway, Enterprise Application Integration. Experiences with messaging services like Apache Kafka or Rabbitmq. Knowledge of multimedia database indexing structures like M-Tree, AESA, LAESA, GNAT or D-index and searching in multimedia database based on similarity. Multimedia databases querying with range query and KNN query.

APIs, Middleware architecture, Microservices, Messaging services, Multimedia database indexing, Multimedia database querying

Projects

Monorepo full-stack setup

Setup for a medium-sized web application. Repository structured as a monorepo using Turborepo. All code is transpiled and transformed using the TSUP bundler. PostgreSQL database with Drizzle ORM for easy administration and type-safe database access. For authentication user used Authjs framework connected using Drizzle ORM adapter. Strongly typed API created using tRPC, which is configured not only for calls from the browser but also from the Next.js server for instant data retrieval during rendering. Shadcn tool was used to create the components. The web application has implemented translations that can also be used in the package to create reusable components. The whole repository is designed in a very modular way which supports clear code management and organization.

TypeScript, tRPC, Next.js, React, Shadcn, Tanstack table, Tanstack query, Drizzle ORM, Authjs, TSUP, Translations, Zod, Turborepo, Vitest, Dotenvx, Github Actions

PTSQ

tRPC technology is a frequently used tool for creating strongly typed APIs. A typed API guarantee gives the client an assumption about the data types it can send and receive. Unlike alternative technologies like GraphQL, tRPC is simpler and doesn't require code generation, but it only works for very narrow project settings. It is designed for a mono repository structure and proprietary type-safe API only. This work seeks to overcome these boundaries by designing and implementing a custom library that offers a strongly typed API and schema introspection to create an open strongly typed API while maintaining the simplicity of project creation supported by tRPC.

TypeScript, tRPC, GraphQL, Zod, Typebox, Tanstack query, JSON schemas, Typesafe API, REST, Prisma, Changesets, Turborepo, Vitest, Github Actions

Kiq

The HTML code of the web page is converted by the browser into the JavaScript object model, which uses methods and functions to reverse the page editing. The edits often cascade through the complex page model, degrading page performance. This is partially offset by a number of optimizations that make most of these objects non-reactive. This means that changing variables within JavaScript does not directly translate into changes, and redrawing the page and interface must be updated explicitly. Kiq eliminates these drawbacks by creating a reactive proxy model.

React, JavaScript, TypeScript, DOM, Reactivity, Declarative programming, Virtual DOM, Babel, JSX, Hydration, SSR

lewis-wow Ludvík Prokopec ludvik.prokopec@gmail.com ☎ +420 732 520 250