

Case-study for SvelteKit in a modern business application

Hannah Lappe

April 25, 2023

Abstract

Over the past decade, single page applications (SPAs) have become the primary approach for developing websites. This development was fueled by JavaScript frameworks such as React, Angular and Vue which made it easy to develop rich web applications. But modern web applications need to satisfy a growing number of technical requirements. This has led to a point where it has become difficult to start new projects from scratch.

To address this issue, in recent years the JavaScript ecosystem saw the emergence of so-called meta frameworks. Meta frameworks are JavaScript libraries that try to fit the role of opinionated "batteries included" frameworks, known from other programming languages, such as Ruby on Rails or Django. They provide out of the box support for many of the features required for building modern web applications. Thus, developers need to spend less time thinking about software architecture and can spend more time thinking about business requirements. Furthermore, meta frameworks use JavaScript for both frontend and backend, Which allows for a tighter integration between the two and therefore enables a more streamlined development experience, as well as reducing overall application complexity.

SvelteKit is an up-and-coming meta framework that is rapidly gaining popularity. It has multiple advantages over traditional SPA frameworks. SvelteKit allows per page control over the rendering strategy. This allows to render different parts of an application wit server side rendering (SSR), client side rendering (CSR) or at build time with pre-rendering. Furthermore, SvelteKit uses various compile time optimizations to produce a more efficient and performant JavaScript bundle. Moreover, it provides functionality, such as reactive state management, component composition, and streamlined data fetching, which helps in writing concise and readable code. Overall, SvelteKit promises to make development of modern web applications faster, while simultaneously improving developer experience.

In this study, we investigate if SvelteKit's claimed benefits translate to the development of business applications. To this end we reimplement parts of an application that is used to schedule rides for a chauffeur service provided by a federal agency. We compare the resulting application with the original implementation in terms of performance and development effort. Furthermore, we want to generalize the results from this project to typical business applications and try to provide a blueprint for future projects.