Front-end frameworks

Which front-end framework is best suited for building a webshop?

Literature study / Trend analysis

Date	:	5-4-2024
Version	:	1.0
Status	:	Draft
Author	:	Luc Swinkels

Version history

Version	Date	Author(s)	Amendments	Status
0.1	4-4-2024	Luc Swinkels	First draft	Draft
1.0	5-4-2024	Luc Swinkels	First definitive version	Definitive

Table of contents

Context **4**

Methods **5**

Trend analysis 5

Literature study 5

Results **6**

Trend analysis 6

Literature study 7

Conclusion **10**

Sources **11**

Context

The goal of this research is to answer the following research question:

“Which front-end framework is best suited for building a webshop?”

Answering this question means I need to figure out which options are available, and which option is best suited for my needs.

To do this, I will be using the literature study (Vogel, n.d.) and trend analysis (*CMD Methods*, n.d.) methods.

Methods

Trend analysis

I’m going to look at trends in the web development space by using community platforms in the webdevelopment space such as StackOverflow and analysing which technologies are often used by professional developers.

Literature study

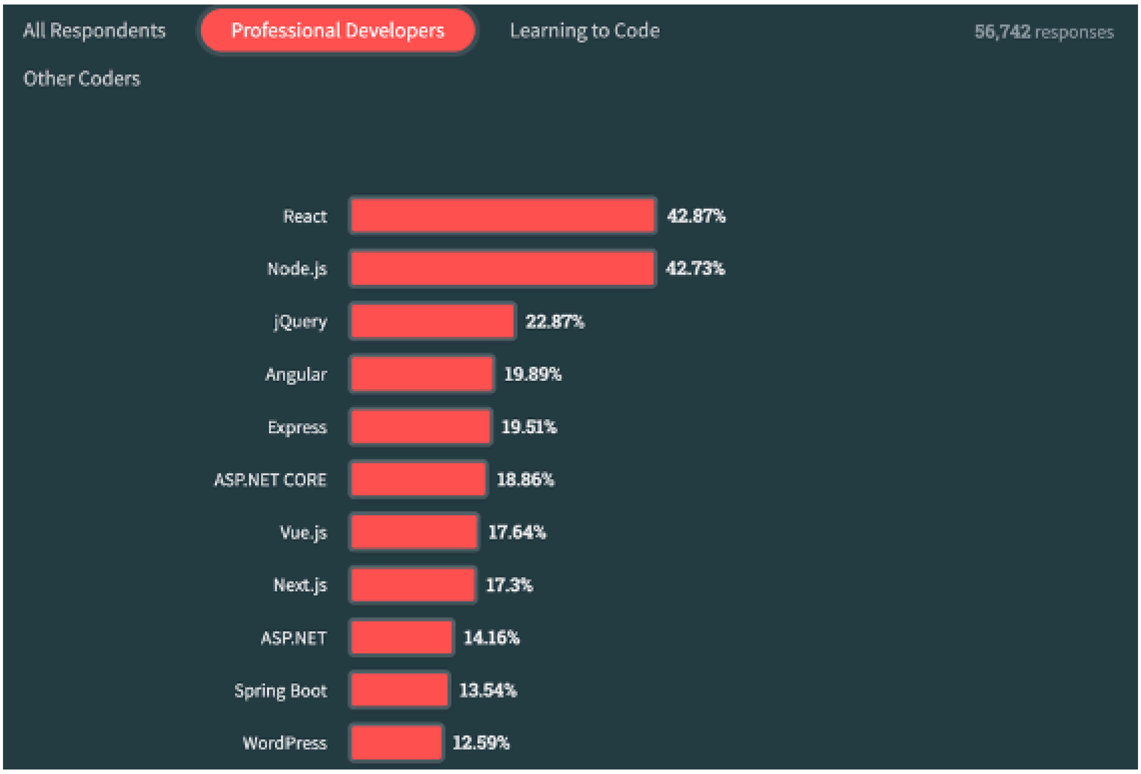
I will be browsing the internet for resources on popular frameworks, what their pros and cons are, and ultimately what would fit best with my type of project (building a webshop).

Results

Trend analysis

Stack Overflow 2023 survey

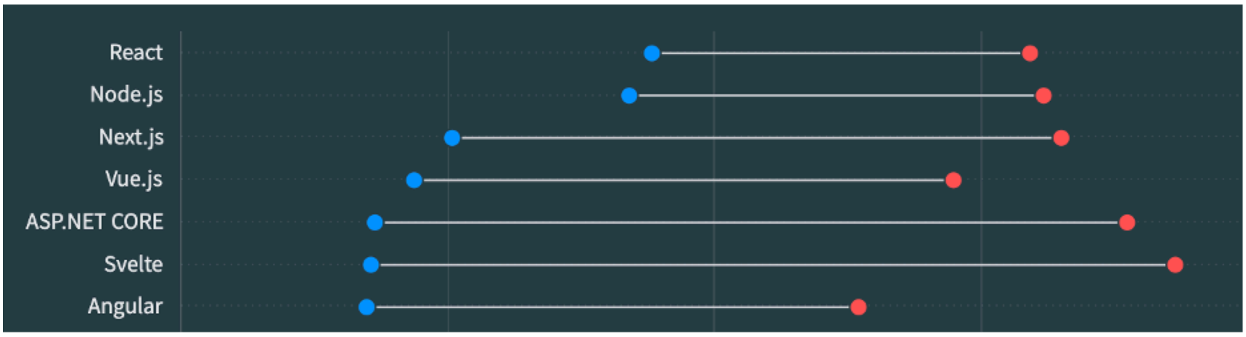
My first point of reference is going to be the 2023 Stack Overflow Developer survey (*Stack Overflow Developer Survey 2023*, n.d.). This survey, answered by 90,000 developers, gives tons of insight in modern web development practices. I will be basing my findings on the professional developers survey results, as I believe the “learning to code” results are worse to go off when looking for a professional solution.



Most popular web frameworks – Stack Overflow Developer Survey 2023

For web frameworks, we also see a clear favourite in React, but we also see JavaScript has completely taken over the world of front-end web frameworks with React, Angular, Vue and Next.js.

Since I also care about development experience and not just what the “most popular” framework is, I also want to look at how much developers prefer working with a specific framework.



Most desired and admired technologies – Stack Overflow Developer Survey 2023 (Blue = desired, red = admired)

This shows that although frameworks like Svelte and Next.js aren't as popular as React, people that do use them, often enjoy working with them.

Literature study

Since I already know from my previous e-commerce platforms research that I will be using Shopify as my platform, my options are somewhat limited. I already know based on my personal experience, as well as the StackOverflow survey results that I want to use TypeScript (JavaScript) as my main programming language, therefore the framework should be a JavaScript based framework.

In the JavaScript framework space, there are a lot options that could be used for this (Shah, 2024) (*Stack Overflow Developer Survey 2023*, n.d.) with each having their pros and cons:

React (*React*, n.d.)
React is the most popular web framework at this time, offering a very wide community of developers which means if you run into an issue, chances are there are 1000 other developers who have run into the same issue, and there will be information on the web to solve it.

Pros include:

- Large community of developers behind it
- Backed by Facebook (and built by)
- Relatively simple to understand in its basic nature
- Good for beginners

Cons include:

- Lack of official documentation (although this is counteracted by the massive community of developers)
- Lots of updates, which means lots of new features to get used to
- Lacks SEO features

Next.js (*Next.Js by Vercel - the React Framework*, n.d.)
Next.js is based on React, which means it offers everything React has, but with some new additional features like server side rendering, image optimization, and other full-stack features for data management.

Pros include:

- Includes anything React has to offer
- Adds lots of useful features to React for data handling, performance, and SEO
- Well-documented

Cons include:

- Can be quite complicated at times due to it being a full-stack framework
- Not as popular as React
- Often tied to the Vercel ecosystem

Remix (*Remix - Build Better Websites*, n.d.)
Remix is very similar to Next.js, also being built on top of React. It also classifies as a full-stack framework and has full-stack features for data handling like Next.js.

Pros include:

- Includes anything React has to offer
- Adds lots of useful features to React for data handling, performance, and SEO
- Well-documented

Cons include:

- Can be quite complicated or bloated at times due to it being a full-stack framework
- Not as popular as Next/React

Hydrogen (*Hydrogen: Shopify's Headless Commerce Framework*, n.d.)
Shopify also has their own headless framework, Hydrogen, which is based on Remix (React). Hydrogen offers everything that React and Remix have to offer, as well as already having integrated Shopify API's to make things easier when building front-ends for Shopify.

Pros include:

- Includes anything React/Remix have to offer
- Adds useful integrations for Shopify API's, speeding up Shopify-related development
- Backed by Shopify

Cons include:

- Niche solution for Shopify specifically, which means the development community behind it is a lot smaller
- Dependant on Remix

Vue.js (*Vue.js*, n.d.)
Vue.js is a direct competitor to React. It uses a different syntax but offers the same functionality throughout. While it is not as popular as React, it is still a very well-documented and popular option.

Pros include:

- Very well-documented
- Good for beginners
- Lightweight

Cons include:

- Not backed by a tech giant
- Not a massive development community behind it
- Lack of scalability features

Nuxt (*NUXT: The Intuitive Vue Framework*, n.d.)
Nuxt is based on Vue.js, similar to how Next.js is based on React. Nuxt aims to solve data handling issues and provide useful add-ons to the already existing Vue framework.

Pros include:

- Includes everything Vue.js has to offer

- Adds useful features to Vue.js for performance, SEO and data handling

Cons include:

- Not backed by a tech giant
- Not a massive development community behind it
- Can be quite complicated or bloated at times due to being a full-stack framework

The reason I didn’t include Angular is because I feel like it is only used for very large-scale organisational applications, and quite outdated in terms of supported technologies. This is also further supported by the StackOverflow survey, where it has a very low “admired” rating.

Conclusion

As a conclusion, I have chosen to opt for Shopify's Hydrogen framework (based on Remix), for a number of reasons:

- It is based on Remix (React), which means I have access to all of React/Remix's features and broad community of developers.
- It has integrated tools for Shopify API's, as it was made by Shopify. Instead of having to write a lot of custom code to accomplish simple Shopify connections, this is already done for me.
- It is literally designed for this exact purpose. That means my development experience will be catered towards what I want to achieve, and should mean that my development experience is significantly better than when using a framework that was meant for something such as static websites. This also means that it should be the easiest or fastest way to build a headless front-end for Shopify websites.
- It is a fairly new technology, and I have never worked with either Remix or Hydrogen. I do already have experience with React and Next.js, so using Hydrogen will not only expand my knowledge about Hydrogen and Shopify, but also about the Remix framework, which is knowledge that I value very much.

Sources

1. Vogel, J. (n.d.). ICT Research Methods — Methods Pack for research in ICT. ICT Research Methods. https://ictresearchmethods.nl/library/literature-study/
2. CMD methods. (n.d.). https://cmdmethods.nl/cards/library/trend-analysis
3. Stack Overflow Developer Survey 2023. (n.d.). Stack Overflow. https://survey.stackoverflow.co/2023/#section-most-popular-technologies-web-frameworks-and-technologies
4. Remix - Build better websites. (n.d.). https://remix.run/
5. React. (n.d.). https://react.dev/
6. Next.Js by Vercel - the React framework. (n.d.). https://nextjs.org/
7. Hydrogen: Shopify's headless commerce framework. (n.d.). Hydrogen: Shopify's Headless Commerce Framework. https://hydrogen.shopify.dev/
8. NUXT: The Intuitive Vue Framework. (n.d.). Nuxt. https://nuxt.com/
9. Vue.js. (n.d.). The Progressive JavaScript Framework | Vue.js. https://vuejs.org/
10. Shah, H. (2024, February 1). 7 Frontend JavaScript frameworks loved by developers in 2024. Simform - Product Engineering Company. https://www.simform.com/blog/javascript-frontend-frameworks/