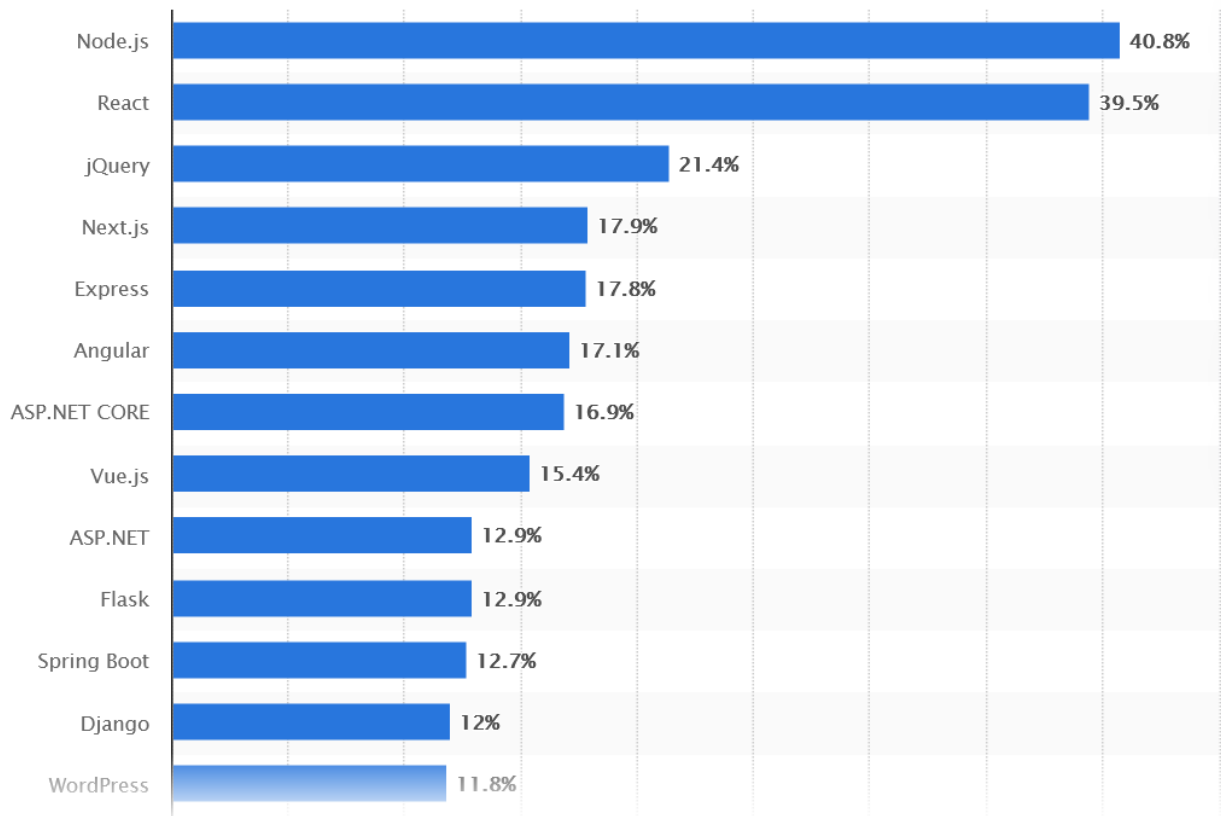


React vs Next.js

Malos Alexandru group 924

When discussing web frameworks, React.js often comes to mind first, as it remains one of the most widely used frameworks among developers worldwide in 2024.



Next.js, built on top of the React library, extends its capabilities with additional features. However, because Next.js builds upon React, it is often considered more challenging to learn. In this document, I will explore the improvements Next.js brings to React and assess whether the added complexity is worth the benefits.

Next.js advantages

1. Speed

Next supports static site generation and server-side rendering, which increases the performance. Due to static generation it pre-renders and caches all web application pages and serving them efficiently. It builds pages on the server that reduces client-side processing time.

2. Less Setup

Next minimizes setup complexities because it provides many features that requiring zero configuration. For example – Page routing doesn't require any setup or code, that simplifies the process of creating routes within your application.

3. Built-in CSS Support

Next provides built-in CSS support, that helps you to include stylesheets without additional libraries or tooling. This is very beneficial for smaller projects in which helps them by avoiding unnecessary complexity while adding a CSS preprocessor.

Head-to-Head Comparison

Feature	React	Next.js
Performance	Faster client-side rendering but slower for first load	Optimized with SSR, SSG, and automatic static optimization
Routing	Requires React Router	Built-in routing system
Data Fetching	Uses hooks like <code>useEffect()</code>	Multiple options: SSR, SSG, ISR, CSR
API Support	Requires external backend API	API routers available in Next.js
Image Optimization	Manual optimization required	Automatic image optimization
Deployment	Requires additional setup for backend & hosting	Easy deployment with Vercel, supports various hosting platforms
SEO	Not SEO-friendly by default	Excellent SEO support with SSR & SSG

Conclusion

Taking everything into account, I will choose Next.js for my project's frontend development.

Documentation:

<https://medium.com/@bektasburakk/react-vs-next-js-which-one-should-you-choose-in-2025-7c36a60f456a>

<https://www.geeksforgeeks.org/nextjs-vs-reactjs-which-one-to-choose/#next-vs-react-comparison>

<https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/>