



Frontend development
Web application



**[https://github.com/up1/
workshop-frontend-web-2025](https://github.com/up1/workshop-frontend-web-2025)**



Frontend Development

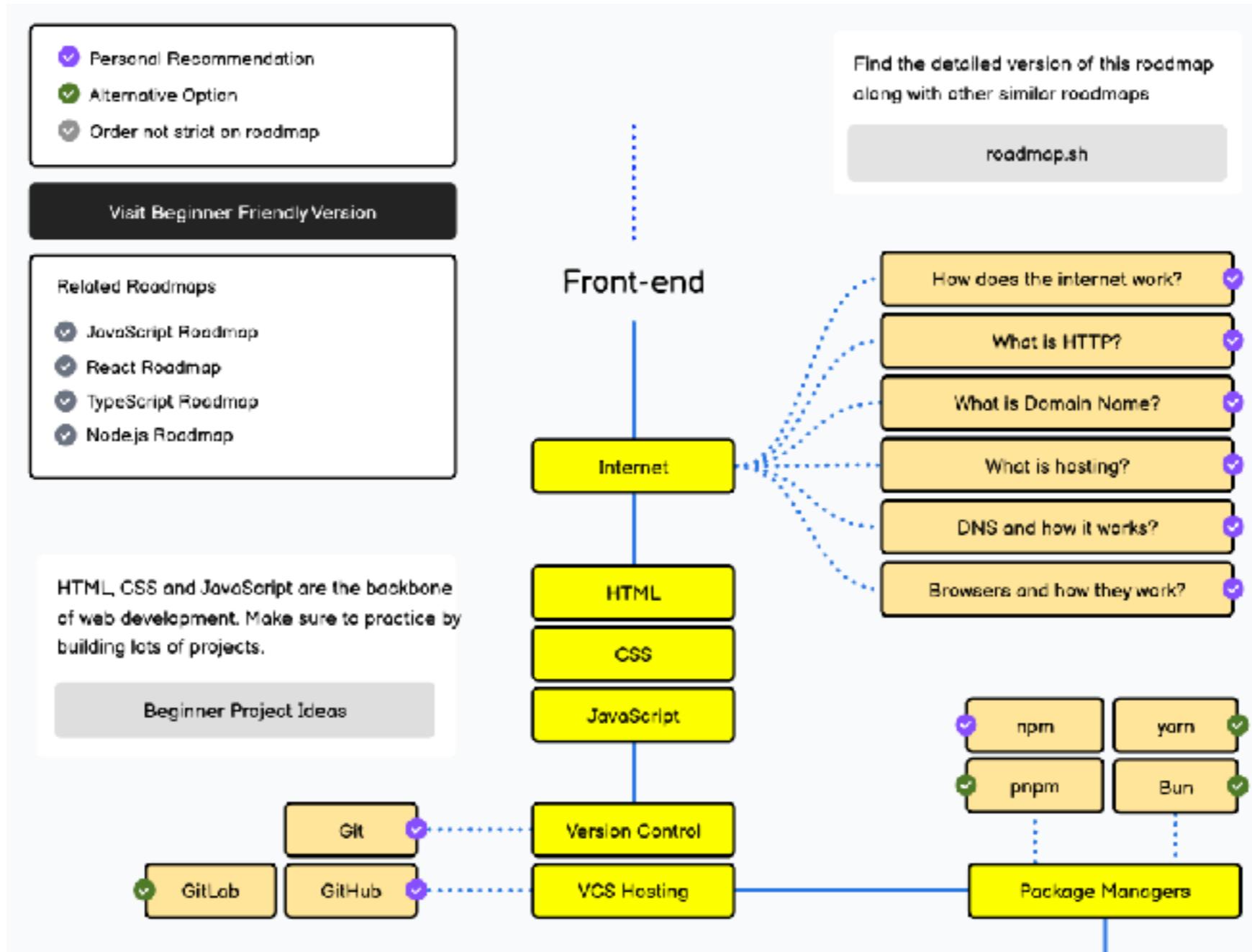
Web Application



Frontend Developer Roadmap



Frontend Developer Roadmap



<https://roadmap.sh/frontend>



Frontend Development

3. Trends and Tools

2. Frameworks and Libraries

1. Core Foundations



Core Foundations



Web Application Properties

SEO
Accessibility
Speed and performance
User Experience (UX)



Core Foundations

HTML, CSS and JavaScript

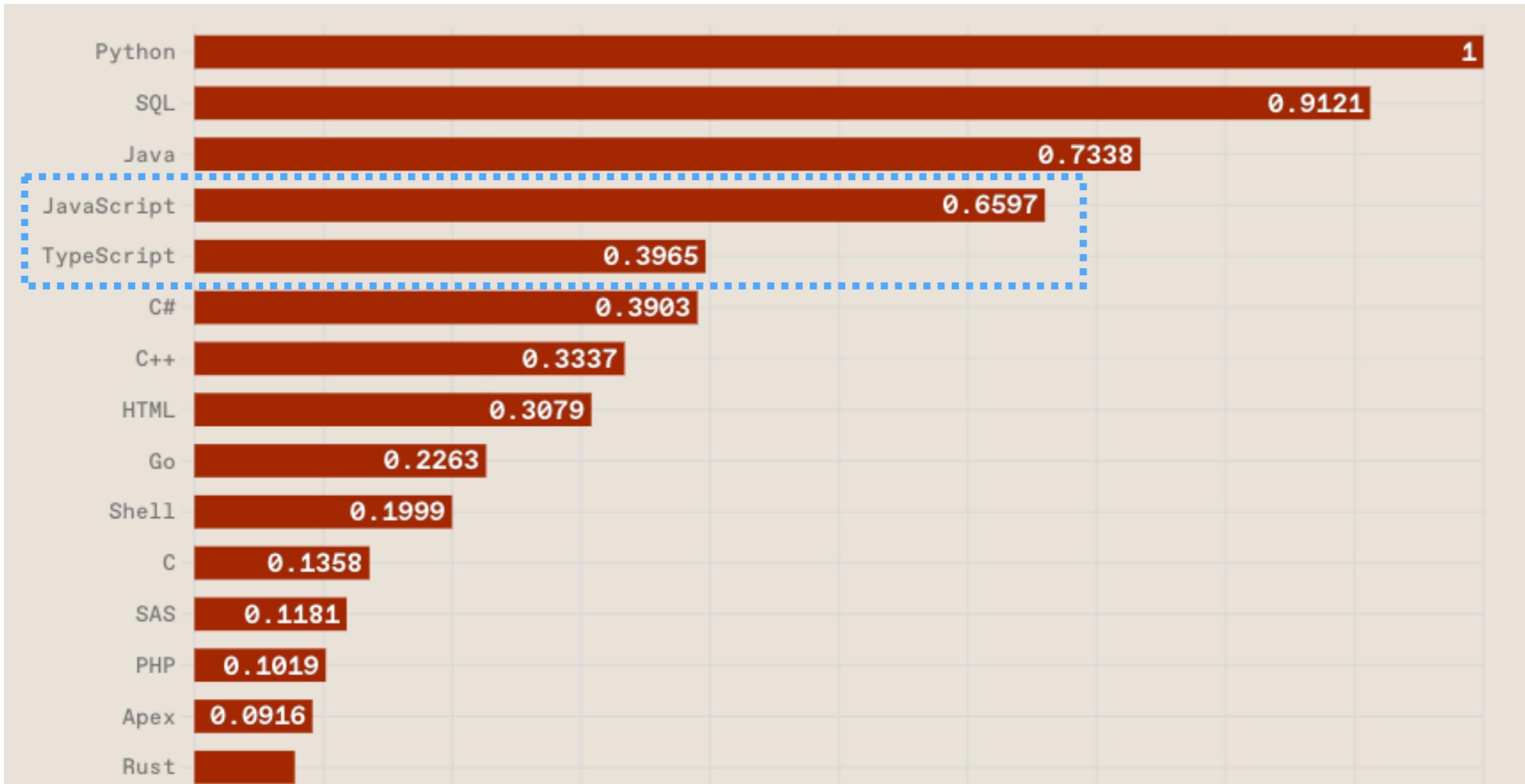
TypeScript

Version Control (git)

Responsive design and **accessibility**



Top Programming 2025

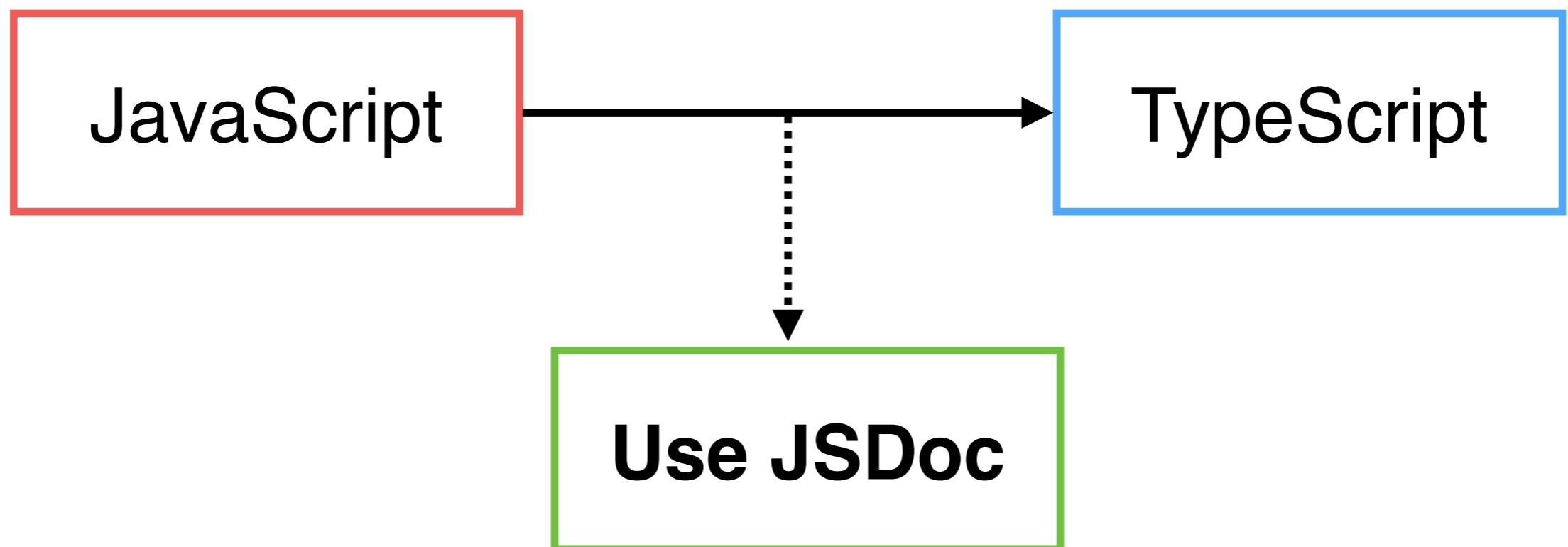


<https://spectrum.ieee.org/top-programming-languages-2025>



Migration with incremental !!

JavaScript to TypeScript

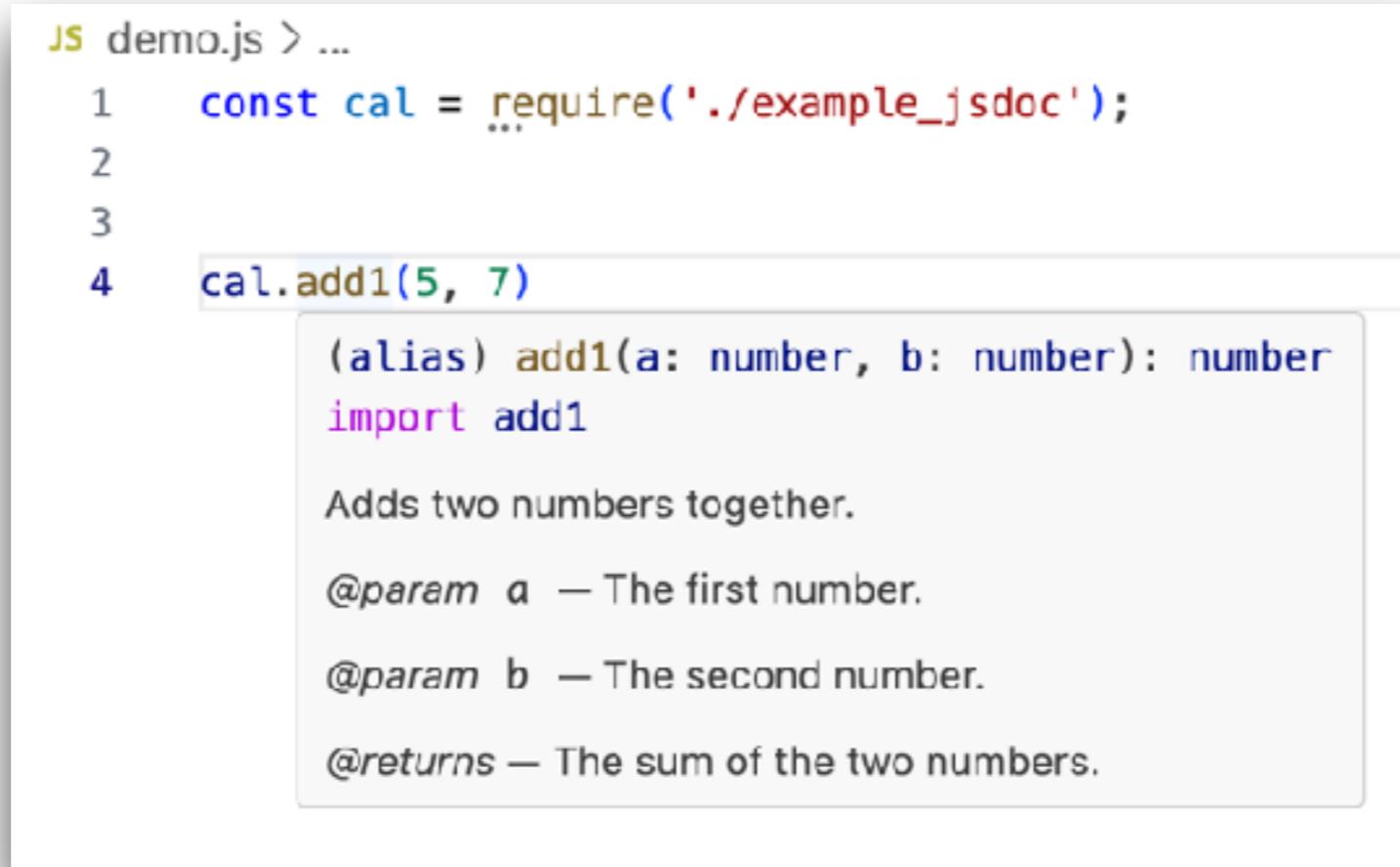


<https://www.typescriptlang.org/docs/handbook/migrating-from-javascript.html>



JSDoc and TSDoc

Documenting your source code
Improve readability and maintainability
IDE integration



The screenshot shows a code editor window with the following content:

```
JS demo.js > ...
1  const cal = require('../example_jsdoc');
2
3
4  cal.add1(5, 7)
```

A tooltip is displayed over the call to `cal.add1(5, 7)`, showing the following documentation:

(alias) `add1(a: number, b: number): number`
`import add1`

Adds two numbers together.

`@param a` — The first number.

`@param b` — The second number.

`@returns` — The sum of the two numbers.



Web Accessibility

The Four Principles of Accessibility

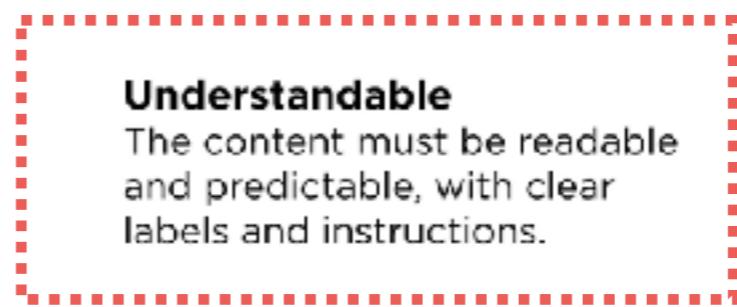
Perceivable

The content must be available to users via sight, hearing, and/or touch.



Understandable

The content must be readable and predictable, with clear labels and instructions.



Operable

The product must be keyboard-accessible, navigable, and compatible with different input methods.



Robust

The product must work with a variety of assistive technologies, browsers, and devices.



Source:

<https://www.w3.org/TR/UNDERSTANDING-WCAG20/intro.html>

<https://www.w3.org/TR/wcag-3.0/>



Check your web app !!

Core Web Vitals



Lighthouse



Page Speed Insights



VS

VS

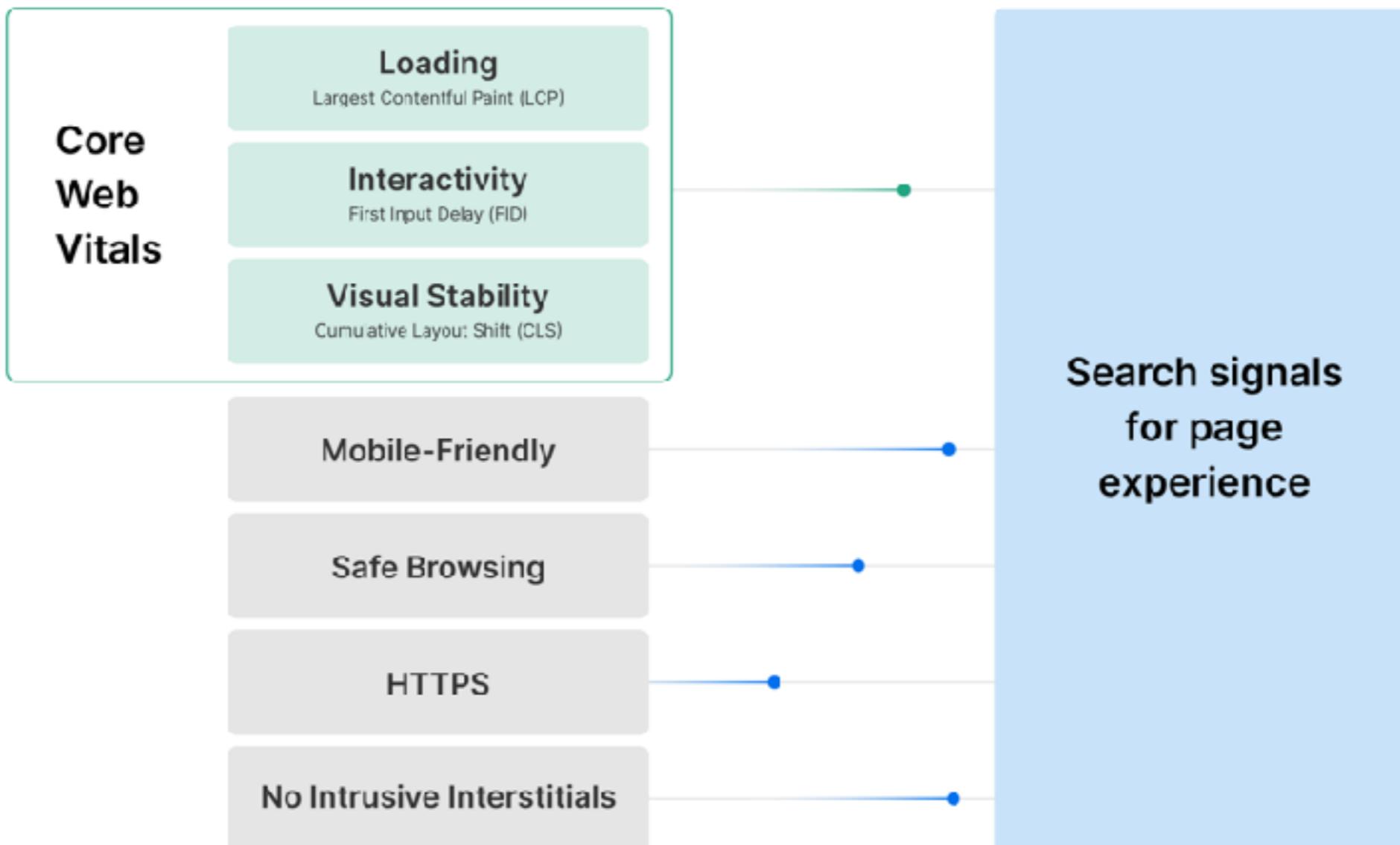
HOW DIFFERENT WEB PERFORMANCE TOOLS COMPARE

<https://learn.microsoft.com/en-us/microsoft-edge/devtools/accessibility/reference>

<https://www.marketingaid.io/understanding-different-performance-measurement-tools/>



Core Web Vitals



<https://web.dev/explore/learn-core-web-vitals>

<https://developer.chrome.com/docs/crux/>



Lighthouse

The screenshot shows the Lighthouse audit interface. At the top, the tab bar includes Elements, Recorder, Console, Sources, Network, Performance, Memory, Application, and **Lighthouse**. Below the tab bar, the URL is listed as <https://untestable.site/>. The audit results are displayed with four red circular icons, each containing a white exclamation mark, corresponding to the categories: Performance, Accessibility, Best Practices, and SEO.

Performance: 3-45 (red)

Accessibility: 58-89 (orange)

Best Practices: 98-100 (green)

SEO: 98-100 (green)

A message box states: "There were issues affecting this run of Lighthouse:" with two bullet points:

- There may be stored data affecting loading performance in this location: IndexedDB. Audit this page in an incognito window to prevent those resources from affecting your scores.
- The page did not paint any content. Please ensure you keep the browser window in the foreground during the load and try again. (NO_FCP)

Performance

Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)

METRICS Collapse view

Metric	Description
First Contentful Paint	Error! The page did not paint any content. Please ensure you keep the browser window in the foreground during the load and try again. (NO_FCP)
Large First Contentful Paint	Error! The page did not paint any content. Please ensure you keep the browser window in the foreground during the load and try again. (NO_FCP)
Total Blocking Time	Error! The page did not paint any content. Please ensure you keep the browser window in the foreground during the load and try again. (NO_FCP)
Cumulative Layout Shift	Error! The page did not paint any content. Please ensure you keep the browser window in the foreground during the load and try again. (NO_FCP)
Speed Index	Error! The page did not paint any content. Please ensure you keep the browser window in the foreground during the load and try again. (NO_FCP)



Lighthouse

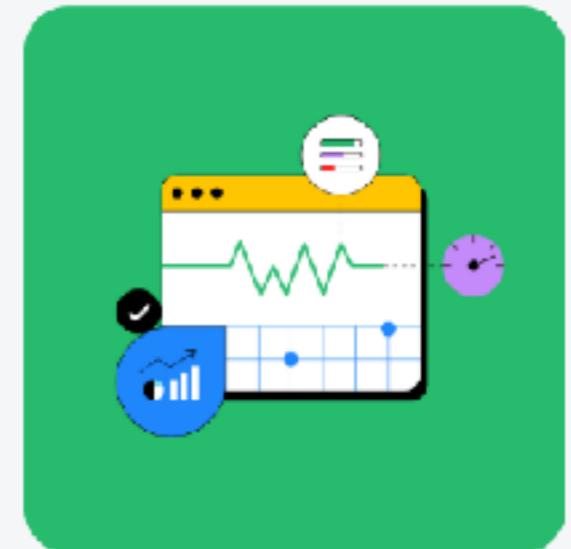
The image shows two side-by-side screenshots. On the left is the homepage of the 'Automation Exercise' website, which is a practice platform for automation engineers. It features a banner with a woman holding shopping bags, a sidebar with categories like Women, Men, Kids, and Brands, and a main area displaying product cards for items like a Blue Top and a Men Tshirt. On the right is the Lighthouse audit results page for the same website. The top navigation bar includes tabs for Home, Products, Cart, Signup / Login, Test Cases, API Testing, Video Tutorials, and Contact us, with 'Lighthouse' selected. The audit summary shows scores of 45 for Performance, 83 for Accessibility, 54 for Best Practices, and 83 for SEO. A note indicates issues with IndexedDB and slow loading times. Below this is a large circular 'Performance' score of 45, with a detailed breakdown of metrics including First Contentful Paint (2.3s), Largest Contentful Paint (2.6s), Total Blocking Time (0ms), Cumulative Layout Shift (1.08), and Speed Index (3.1s). A screenshot of the website's mobile view is also shown.



Core Web Vitals

Core Web Vitals

Essential metrics for a healthy site.



Learn Core Web Vitals

An initiative to provide unified guidance for quality signals that are essential to delivering a great user experience on the web.

Overview

Web Vitals

The business impact of Core Web Vitals

Optimizing Core Web Vitals
for business decision makers

Core Web Vitals Metrics

Largest Contentful Paint (LCP)

Cumulative Layout Shift (CLS)

Interaction to Next Paint (INP)

Defining the Core Web
Vitals metrics thresholds

<https://web.dev/explore/learn-core-web-vitals>



Recommend Metrics

Metric name	Good Threshold
First Contentful Paint (FCP)	<= 1.8 secs
Largest Contentful Paint (LCP)	<= 2.5 secs
Cumulative Layout Shift (CLS)	<= 0.1 secs
First Input Delay (FID)	<= 100 ms
Interaction to Next Paint (INP)	<= 200 ms

<https://developers.google.com/speed/docs/insights/v5/about>



PageSpeed Insights

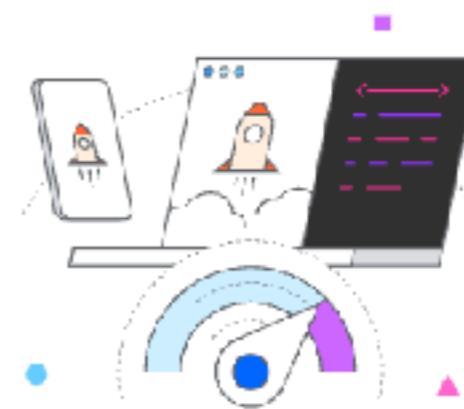
 PageSpeed Insights

Enter a web page URL Analyze

Enter a valid URL

Make your web pages fast on all devices

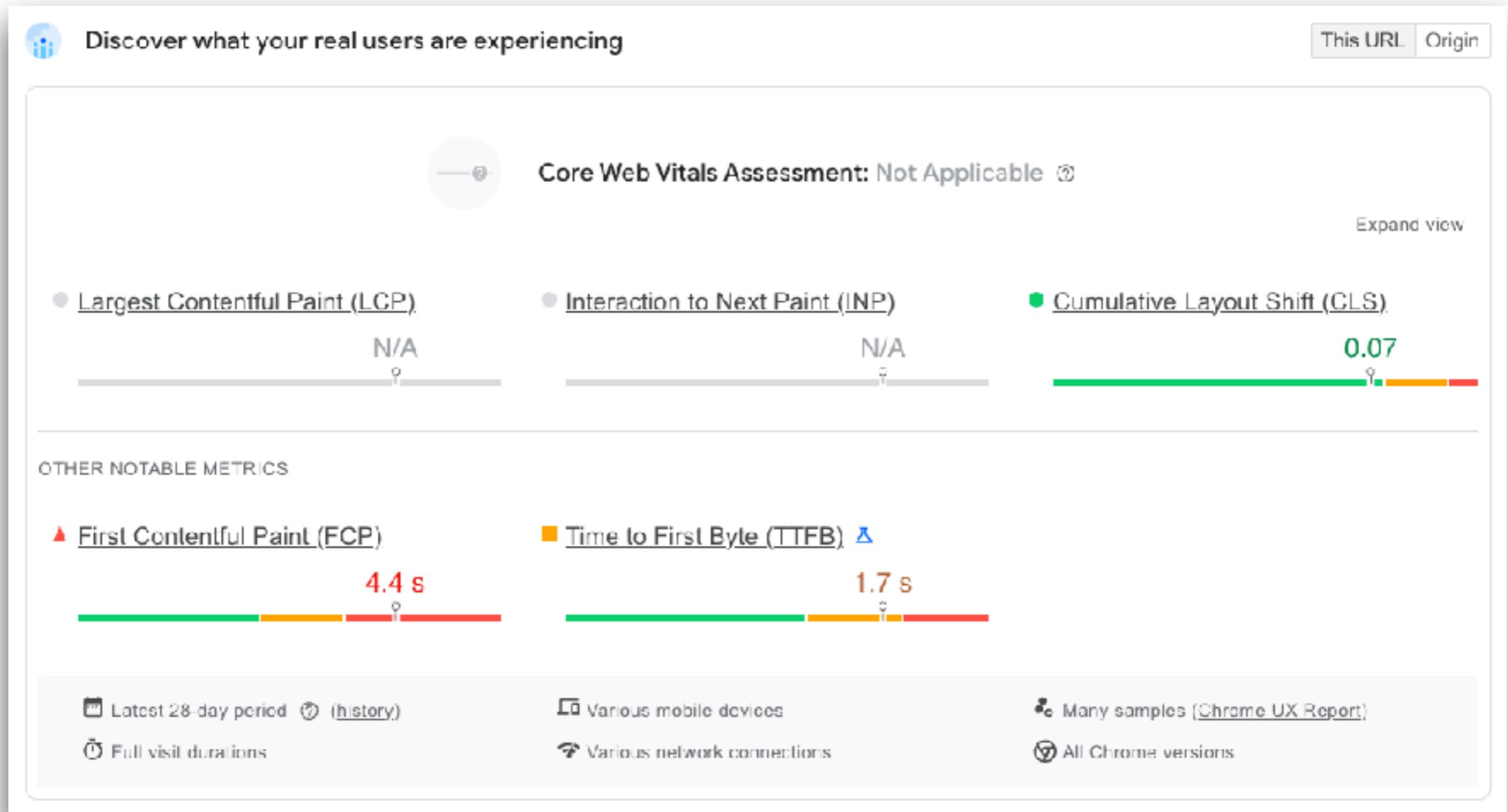
CHECK OUT
[What's new](#)
[Documentation](#)
[Learn about Web Performance](#)



<https://pagespeed.web.dev/>



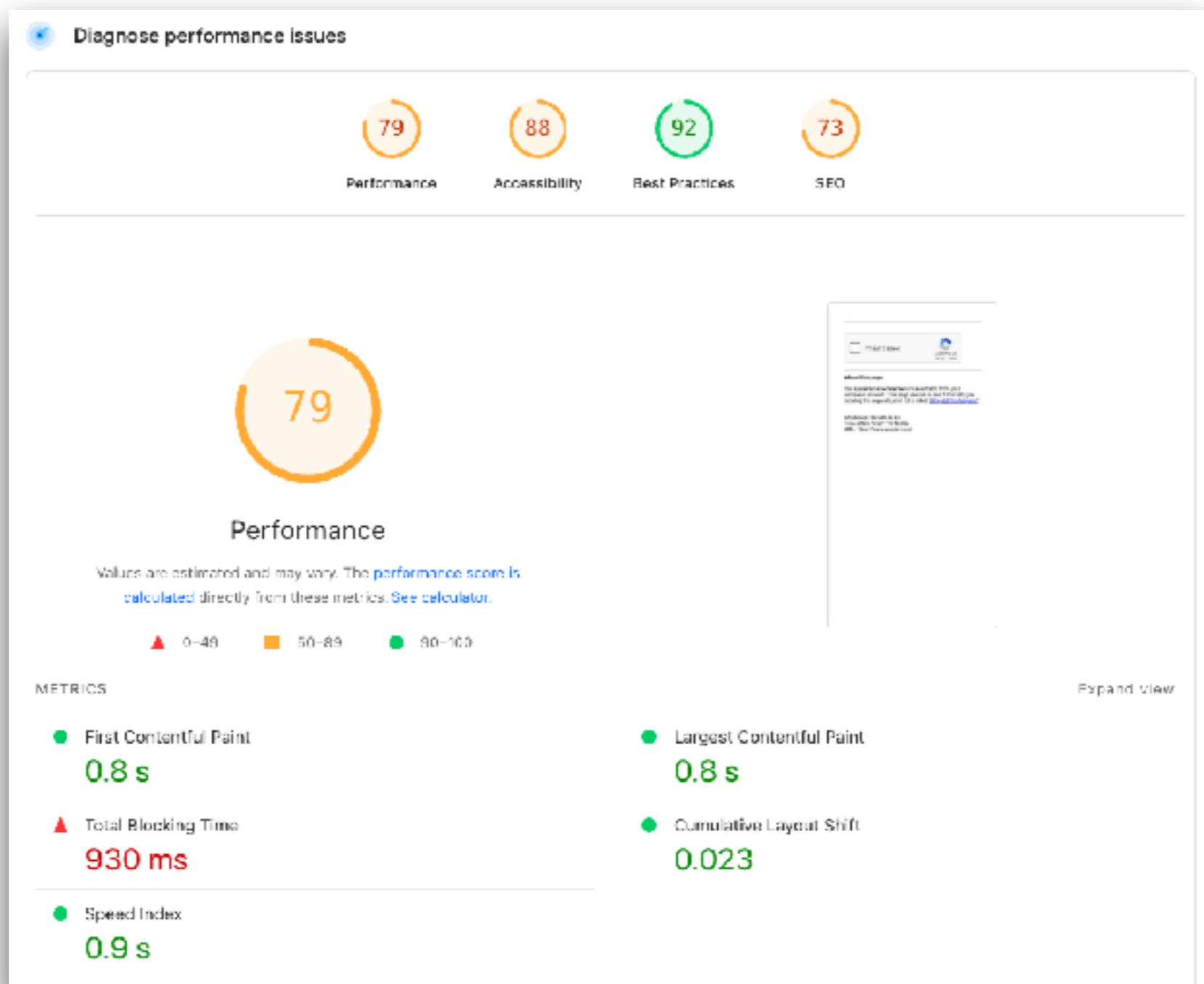
Result !!



<https://pagespeed.web.dev/>



Result !!

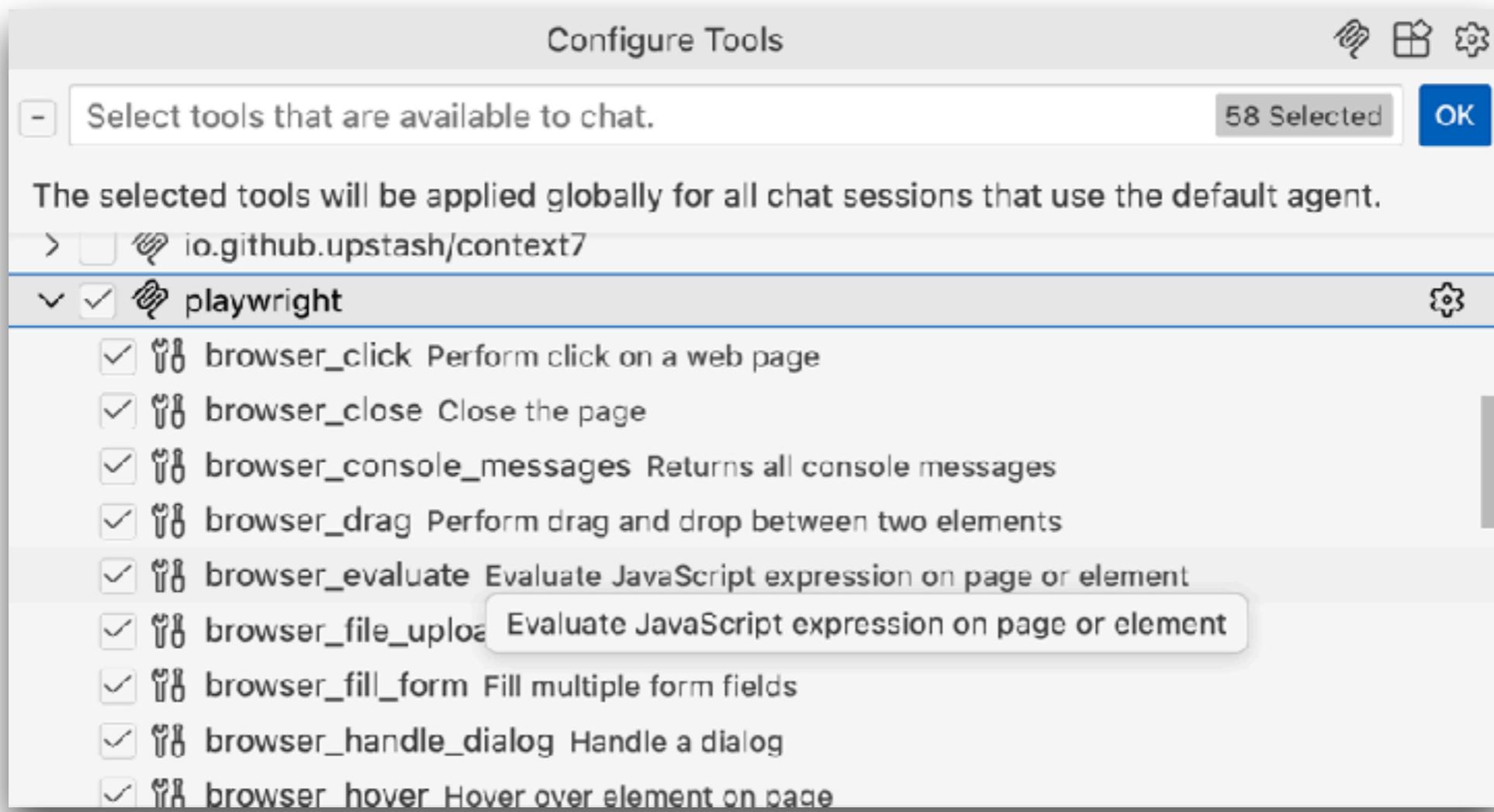


<https://pagespeed.web.dev/>



Try to testing with AI !!

Playwright MCP from Microsoft



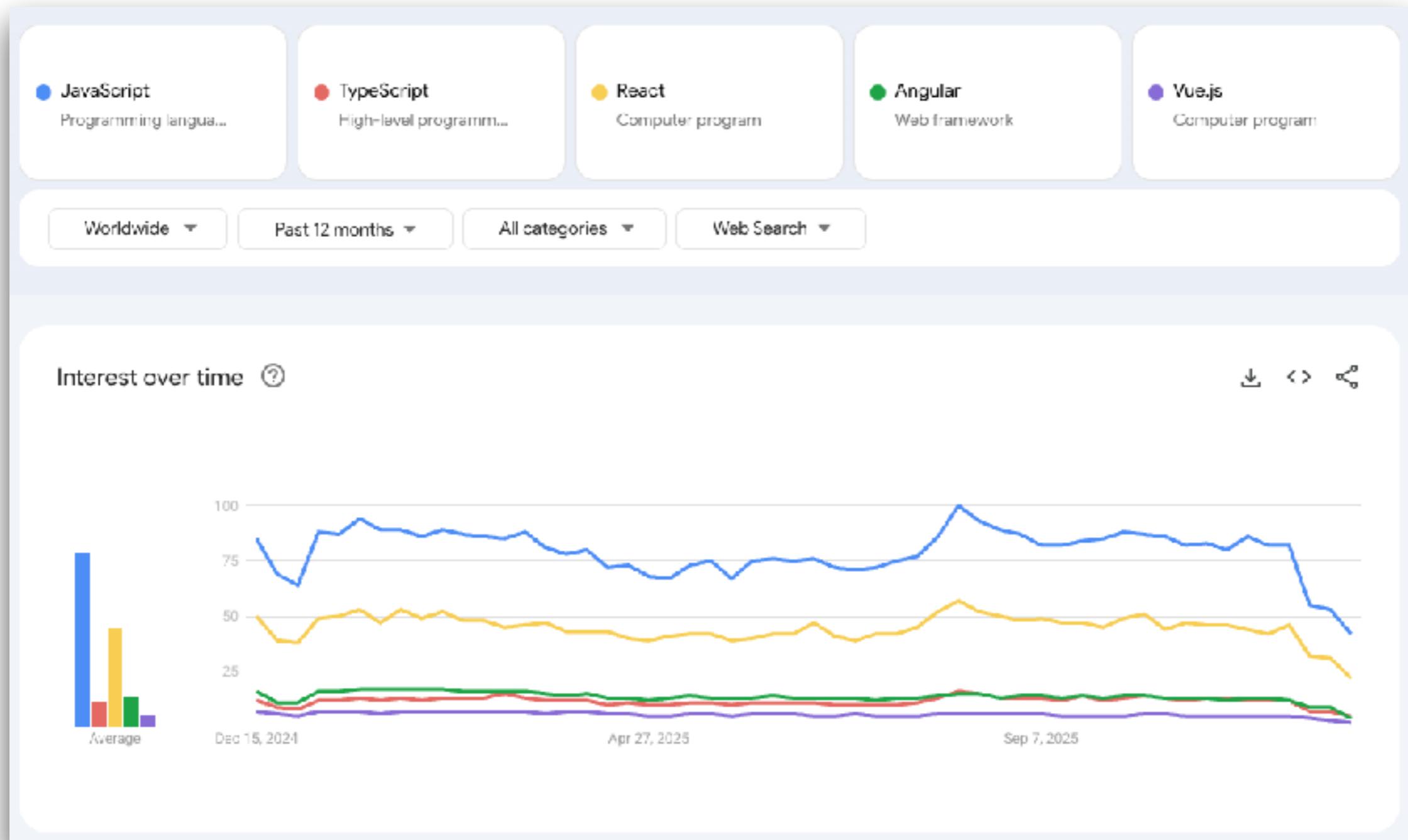
<https://github.com/microsoft/playwright-mcp>



Key Frameworks and Libraries



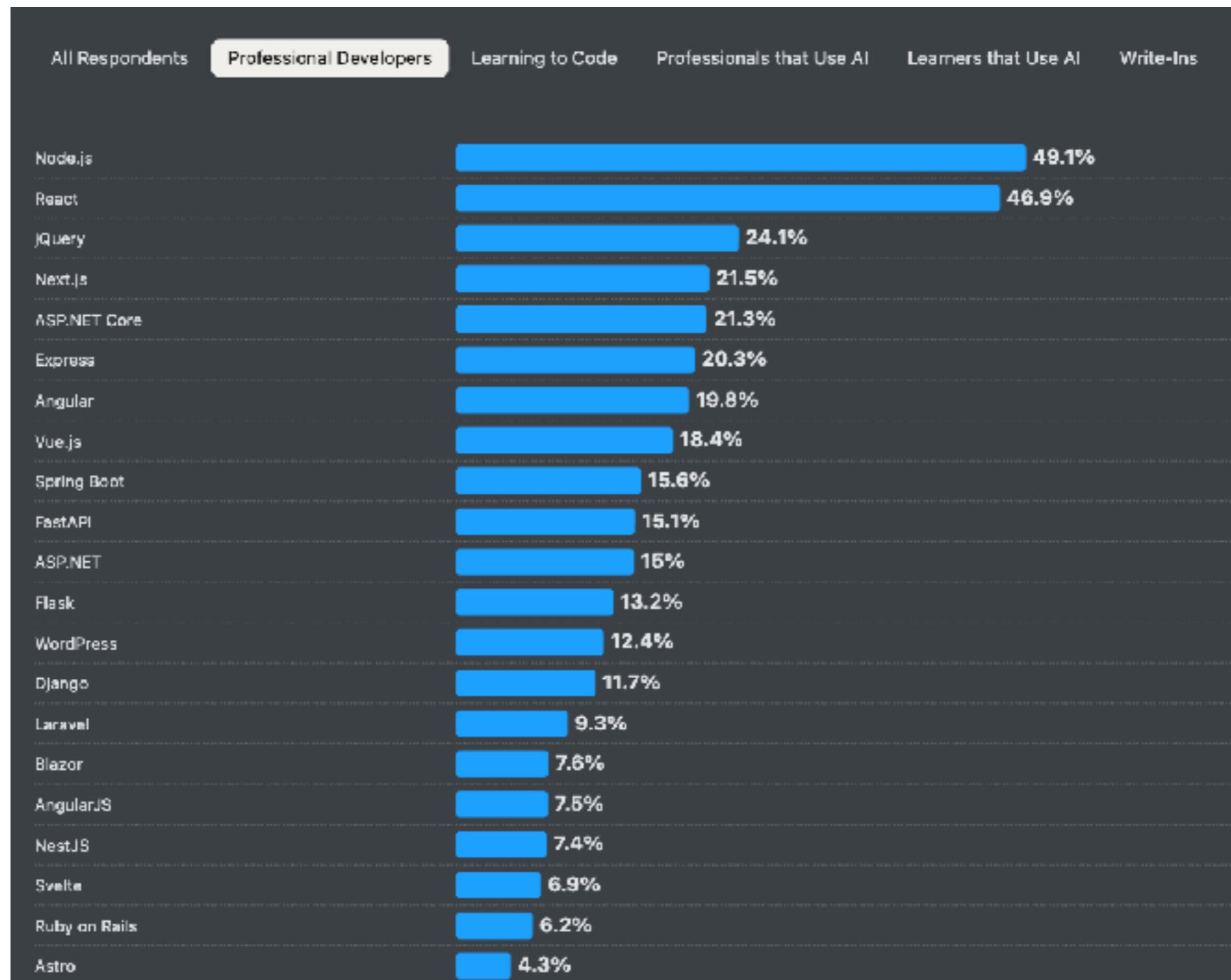
Google Trends !!



<https://trends.google.com/>



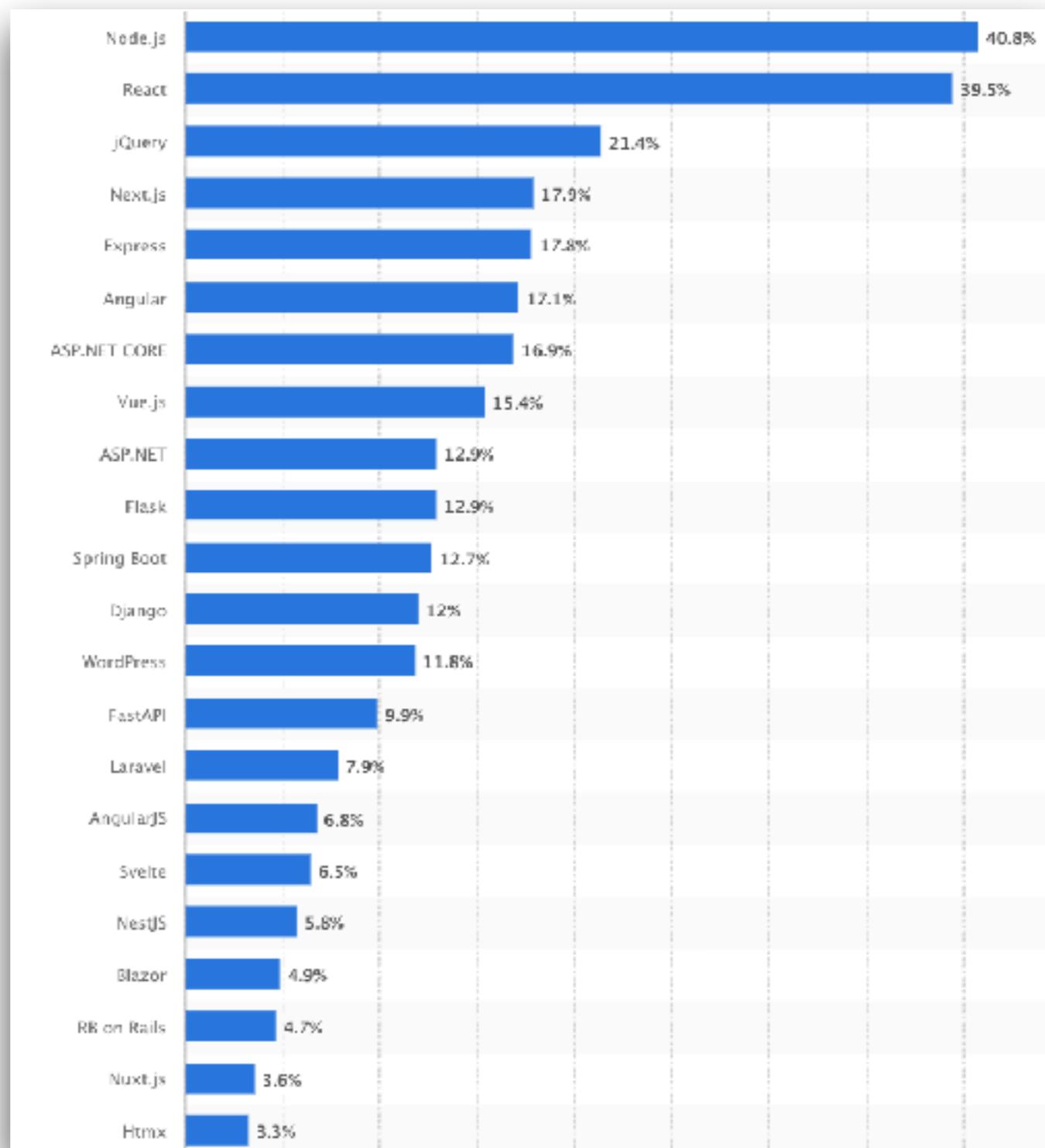
Top Web Frameworks



<https://survey.stackoverflow.co/2025/technology#most-popular-technologies-webframe-prof>



Most used web framework 2024



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



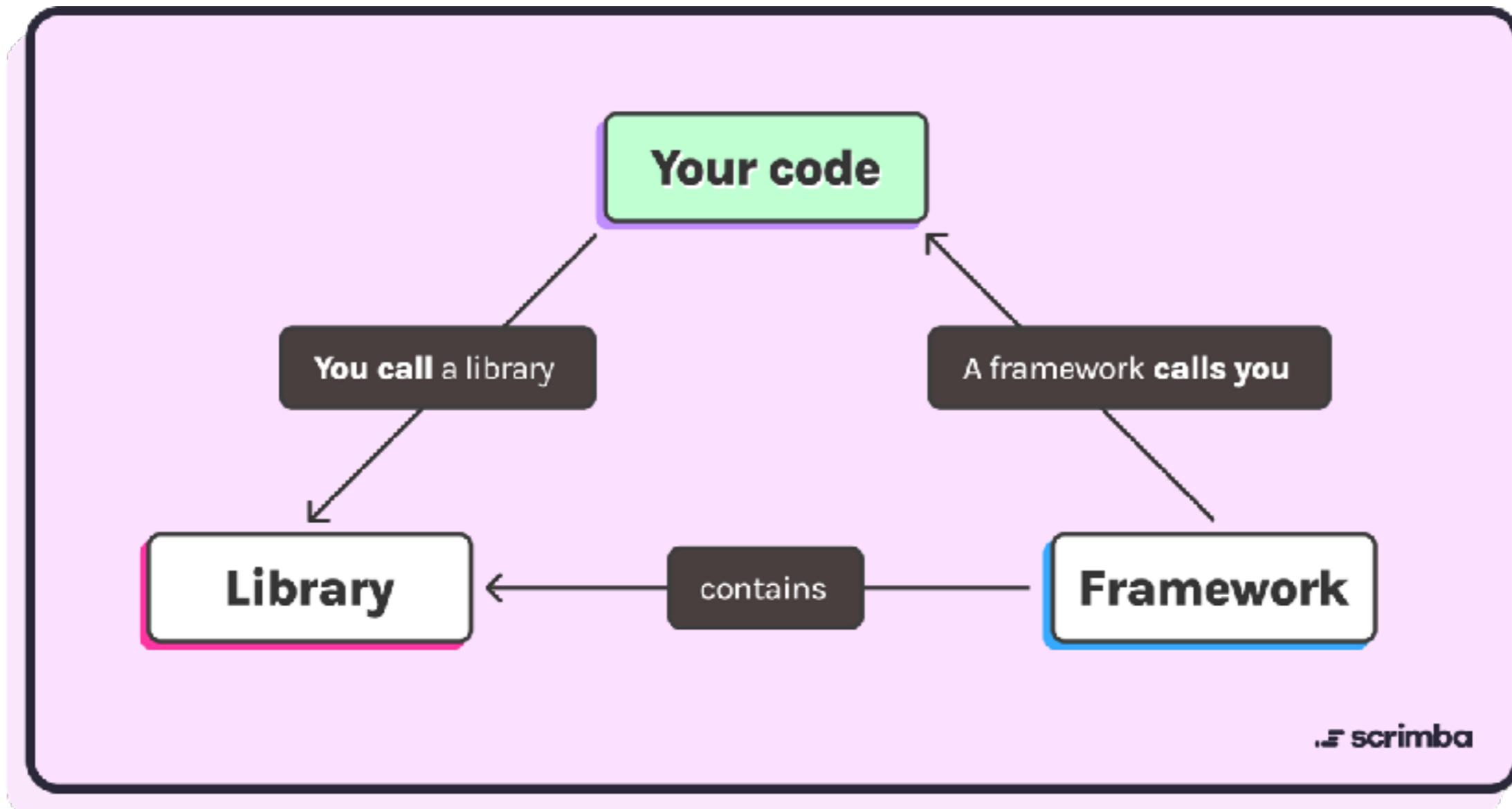
Web Frameworks ?

	React / Next.js	Vue / Nuxt.js	Angular
Core technology	React (library) Next.js (framework)	Vue (framework) Nuxt.js (framework)	Angular (framework)
Language	JavaScript (TypeScript support)	JavaScript (TypeScript support)	TypeScript
Architecture	Component-based	Component-based (Vue) Convention-driven (Nuxt)	Structured Dependency Injection
Rendering	CSR, SSR, SSG, ISR	CSR, SSR, SSG, ISR	CSR, SSR

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



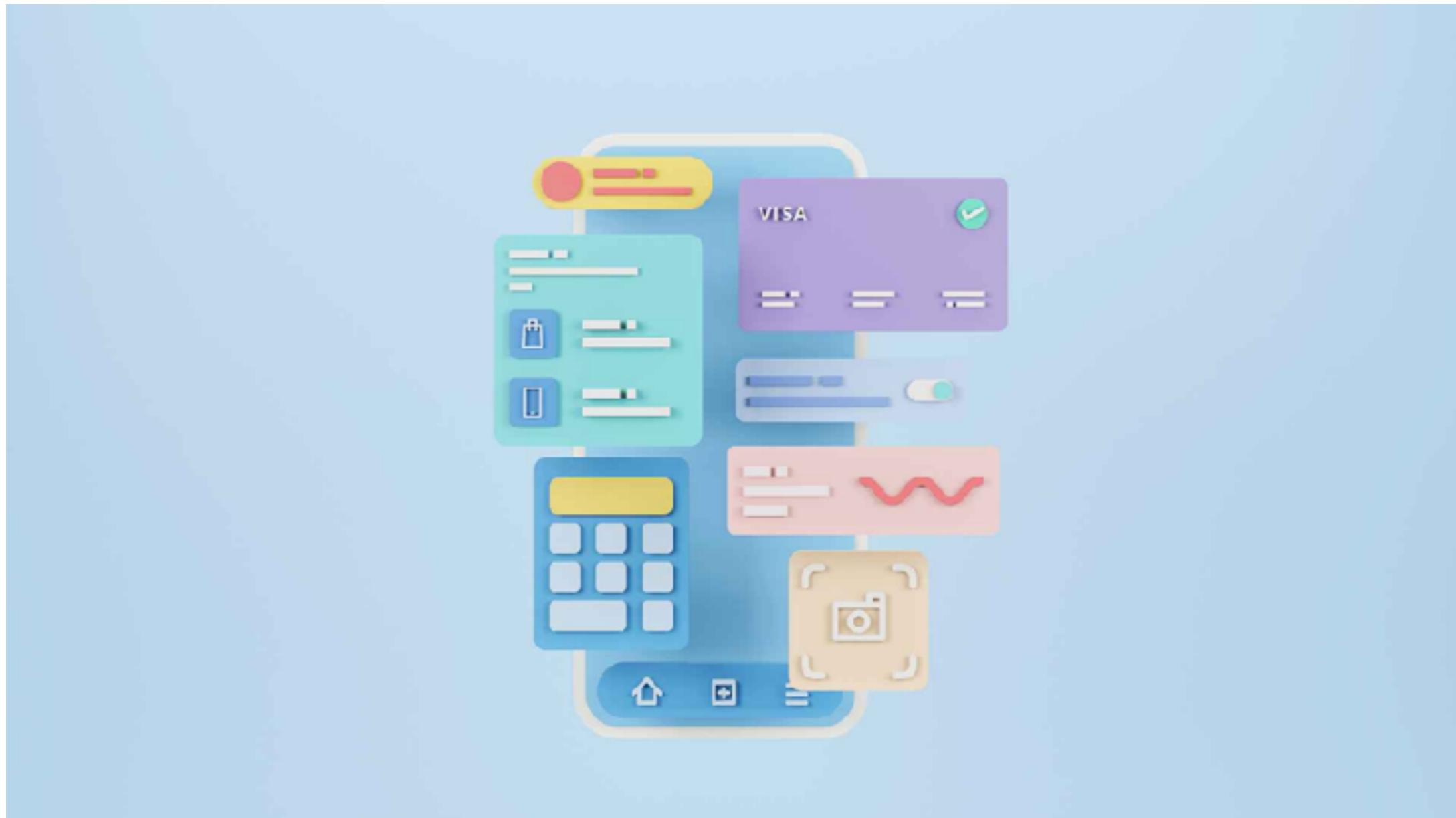
Library vs Framework ?



<https://v1.scrimba.com/articles/difference-between-framework-and-library/>



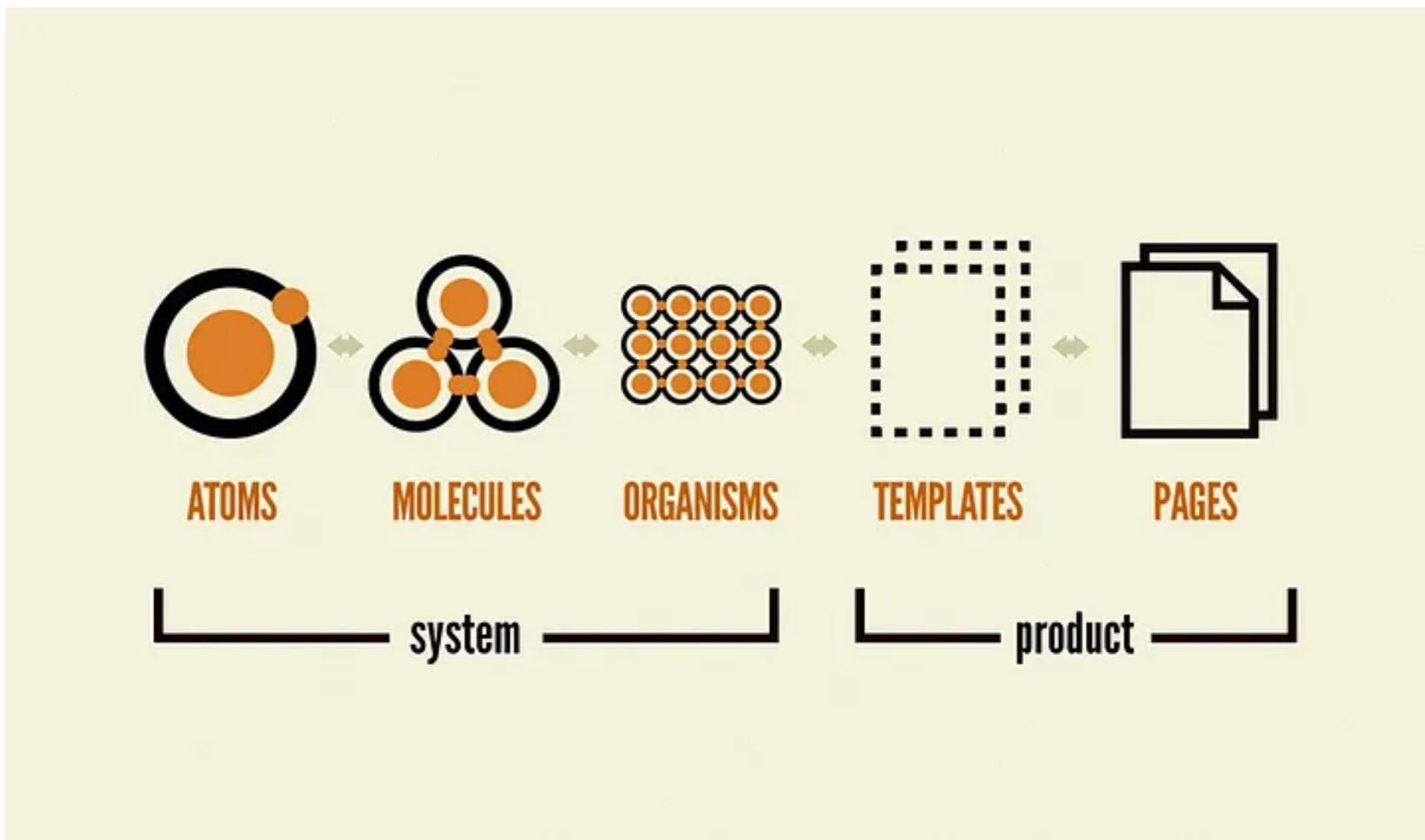
Component-Driven



<https://www.componentdriven.org/>



Atomic Design



<https://atomicdesign.bradfrost.com/>



Focusing on performance !!

Framework	Startup time	Bundle size	Memory use
Qwik	Very fast	~1KB	Very Low
Svelte	Fast	~60-70% smaller	Low
SolidJS	Fast	Compact	Very low
Astro	Static with SSG	Minimal (zero-js) Pure HTML	Tiny

https://krausest.github.io/js-framework-benchmark/2025/table_chrome_138.0.7204.50.html



Architect Your WebApps ?

Improve how you architect webapps

Patterns.dev is a free online resource on design, rendering, and performance patterns for building powerful web apps with vanilla JavaScript or modern frameworks.

[Download eBook or PDF](#)

[Read online](#)

and Addy Osmani

with Lydia Hallie



<https://www.patterns.dev/>



Trends and Tools



Trends and Tools

AI Integration
WebAssembly (Wasm)
Performance optimization
Design system (maintain consistency)



Ant Design

Open source UI design system

The screenshot shows the official website for Ant Design. At the top is a navigation bar with the Ant Design logo, a search bar containing "Type keywords...", and links for Design, Development, Components, Blog, and Resources. A dropdown menu is open over the "Design" link, showing a list item for "Ant Design". Below the navigation is a large hero section with the title "Ant Design" and a subtitle "Help designers/developers building beautiful products more flexible and working with happiness!". Two buttons are visible: "Getting Started" (highlighted in blue) and "Design Language". The main content area features three cards: "Ant Design is here!" (with a "HOT" badge), "CSS in V6", and "Ant Design X 2.0". The "CSS in V6" card contains text about the development of v6 and the use of CSS variables. The "Ant Design X 2.0" card discusses the planned release of Ant Design X 2.0, mentioning MarkdownX, PC components, Mobile components, and X SDK. A banner at the bottom right indicates "Ant Design 6.0 is released!".

<https://ant.design/>



Pain Point of Web application ?



Pain Point of Web application ?

User Experience (UX)
Technical and Development
Performance optimization
Business and process



Common Solutions ?

Prioritize speed (code, image, data)

Simplify UX

Mobile-First design

Use analytics

Regular audits

Invest in **security**



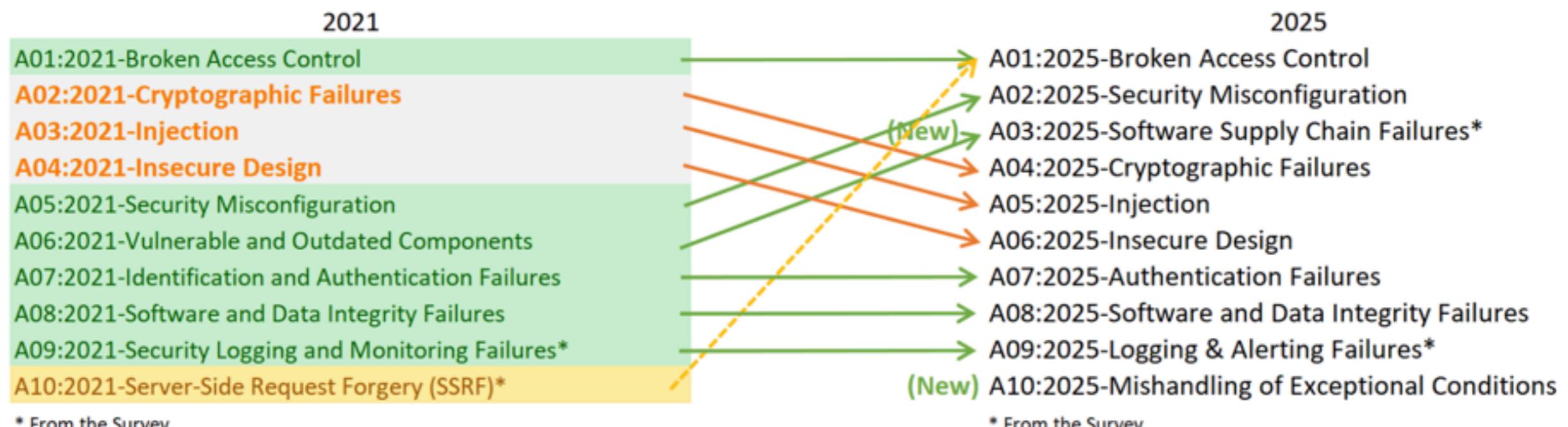
Security of Web application ?



OWASP Top 10 Web

What's changed in the Top 10 for 2025

There are two new categories and one consolidation in the Top Ten for 2025. We've worked to maintain our focus on the root cause over the symptoms as much as possible. With the complexity of software engineering and software security, it's basically impossible to create ten categories without some level of overlap.



https://owasp.org/Top10/2025/0x00_2025-Introduction/



React2shell ?



<https://react.dev/blog/2025/12/03/critical-security-vulnerability-in-react-server-components>



React2shell ?

CVE-2025-55182

Remote Code Execution (RCE)

React Server Components (RSC) in React and Next.js



<https://www.wiz.io/blog/critical-vulnerability-in-react-cve-2025-55182>



React2shell ?

CVE-2025-55182

Remote Code Execution (RCE)

React Server Components (RSC) in **React** and **Next.js**

React 19.0.0, 19.1.0, 19.1.1, 19.2.0
Next.js 15.x, 16.x

<https://www.wiz.io/blog/critical-vulnerability-in-react-cve-2025-55182>



Solutions ?

Update React

Update Next.js

Detection with react2shell-scanner

Network defense (WAF rules)

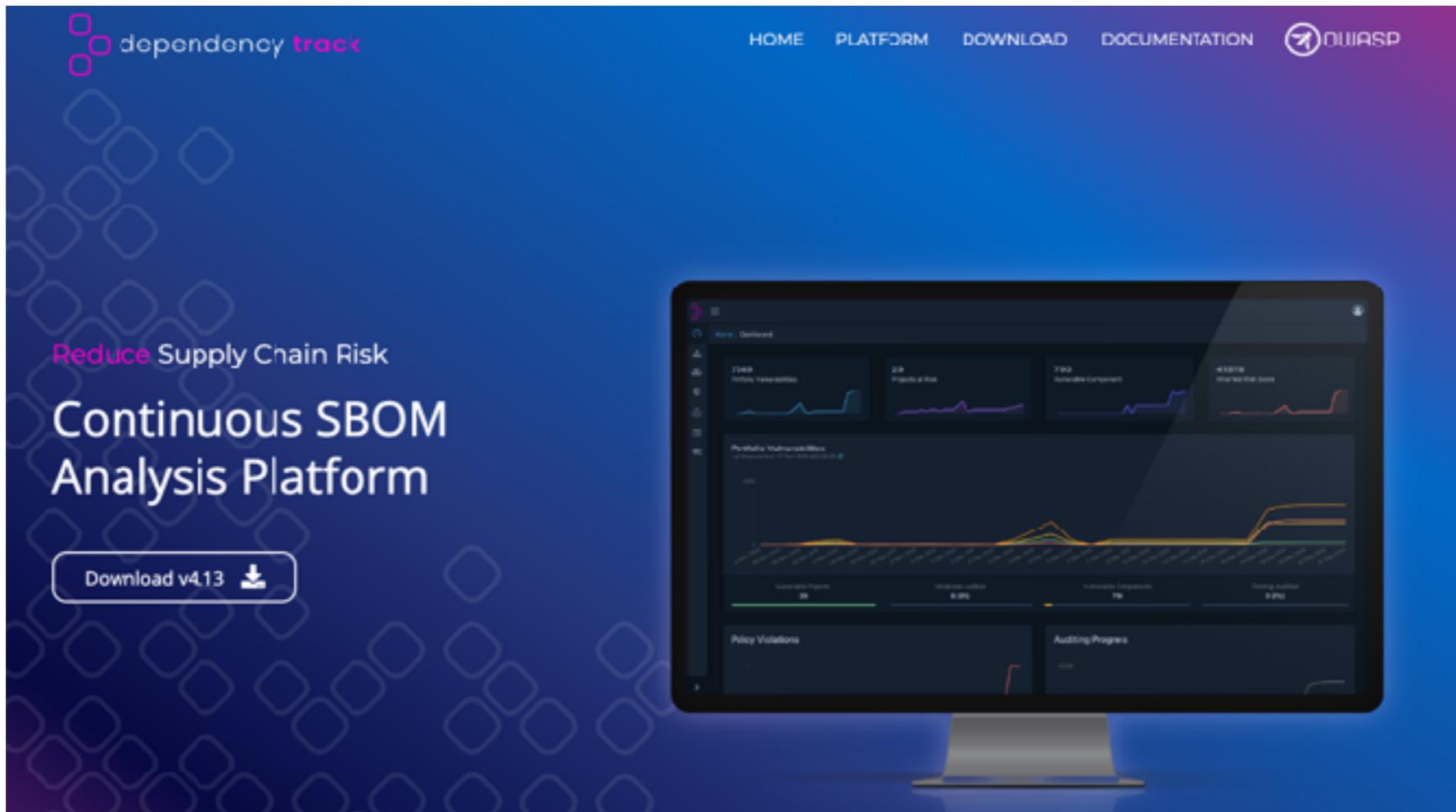
<https://github.com/assetnote/react2shell-scanner>

<https://cloud.google.com/blog/topics/threat-intelligence/threat-actors-exploit-react2shell-cve-2025-55182>



Dependency Track

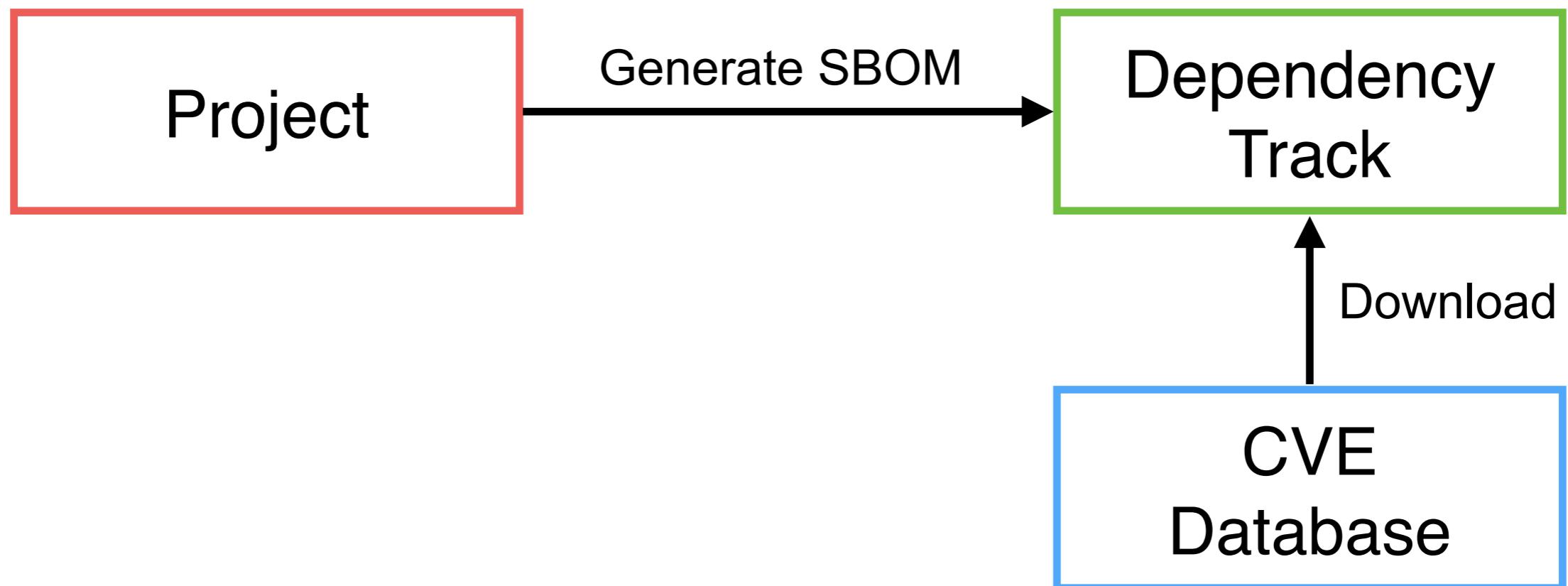
Reduce supply chain risk
Continuous SBM (Software Bill of Material)



<https://dependencytrack.org/>



Process ?



https://en.wikipedia.org/wiki/Software_supply_chain



Q/A

