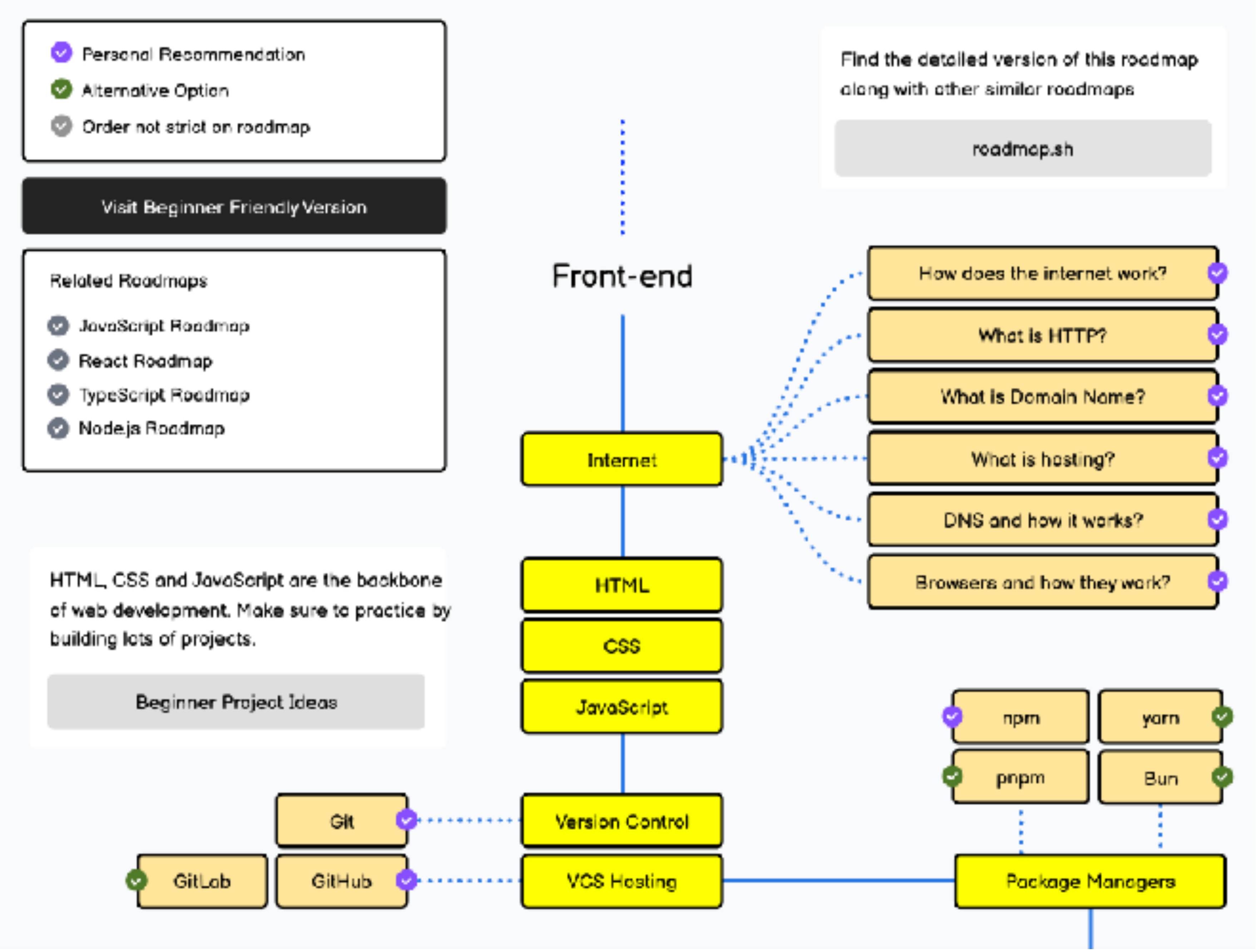
Frontend development
Web application

https://github.com/up1/workshop-frontend-web-2025

Frontend Development
Web Application

Frontend Developer Roadmap

Frontend Developer Roadmap



Personal Recommendation
Alternative Option
Order not strict on roadmap

Find the detailed version of this roadmap along with other similar roadmaps

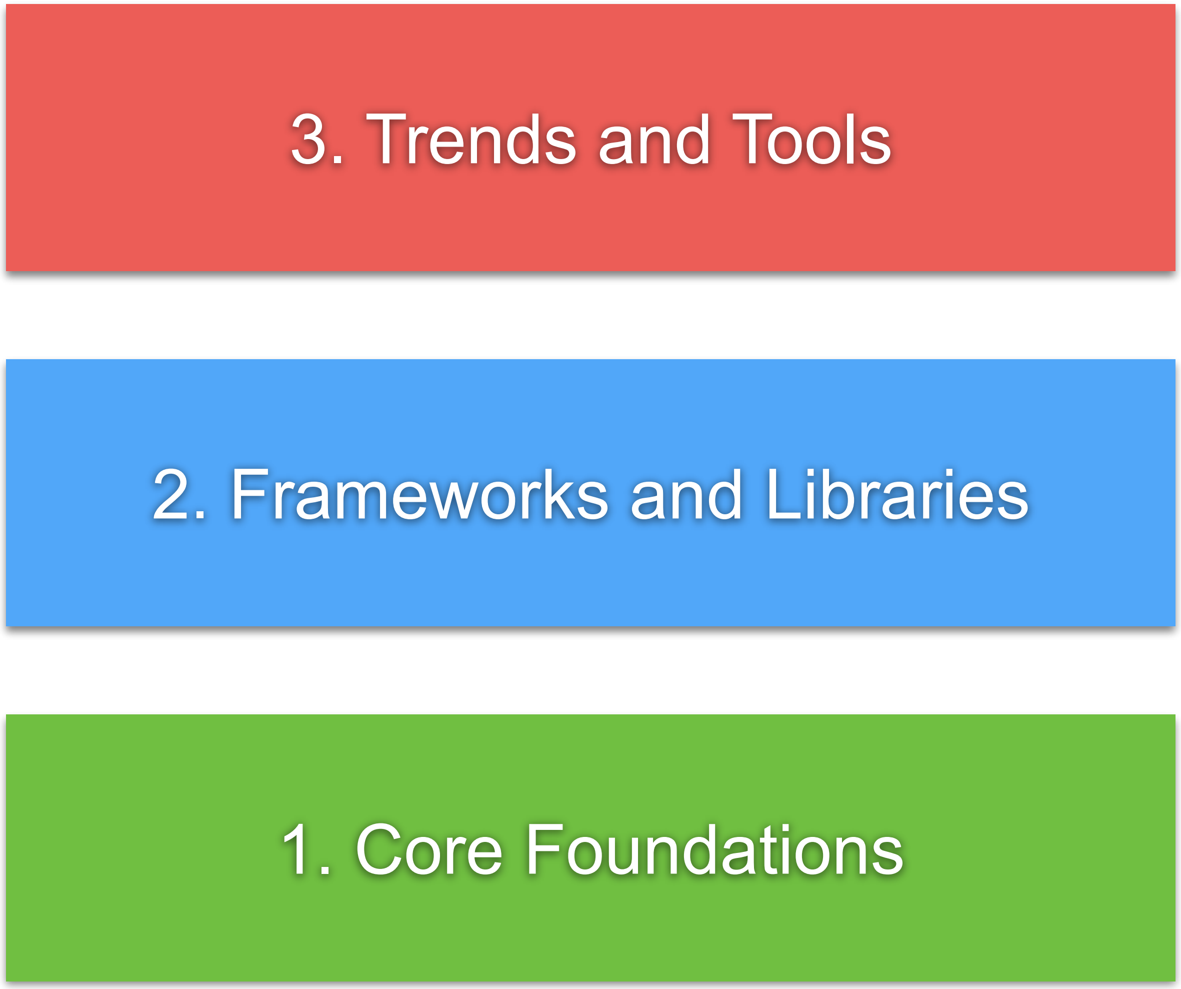
HTML, CSS and JavaScript are the backbone of web development. Make sure to practice by building lots of projects.

Related Roadmaps

- JavaScript Roadmap
- React Roadmap
- TypeScript Roadmap
- Nodejs 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

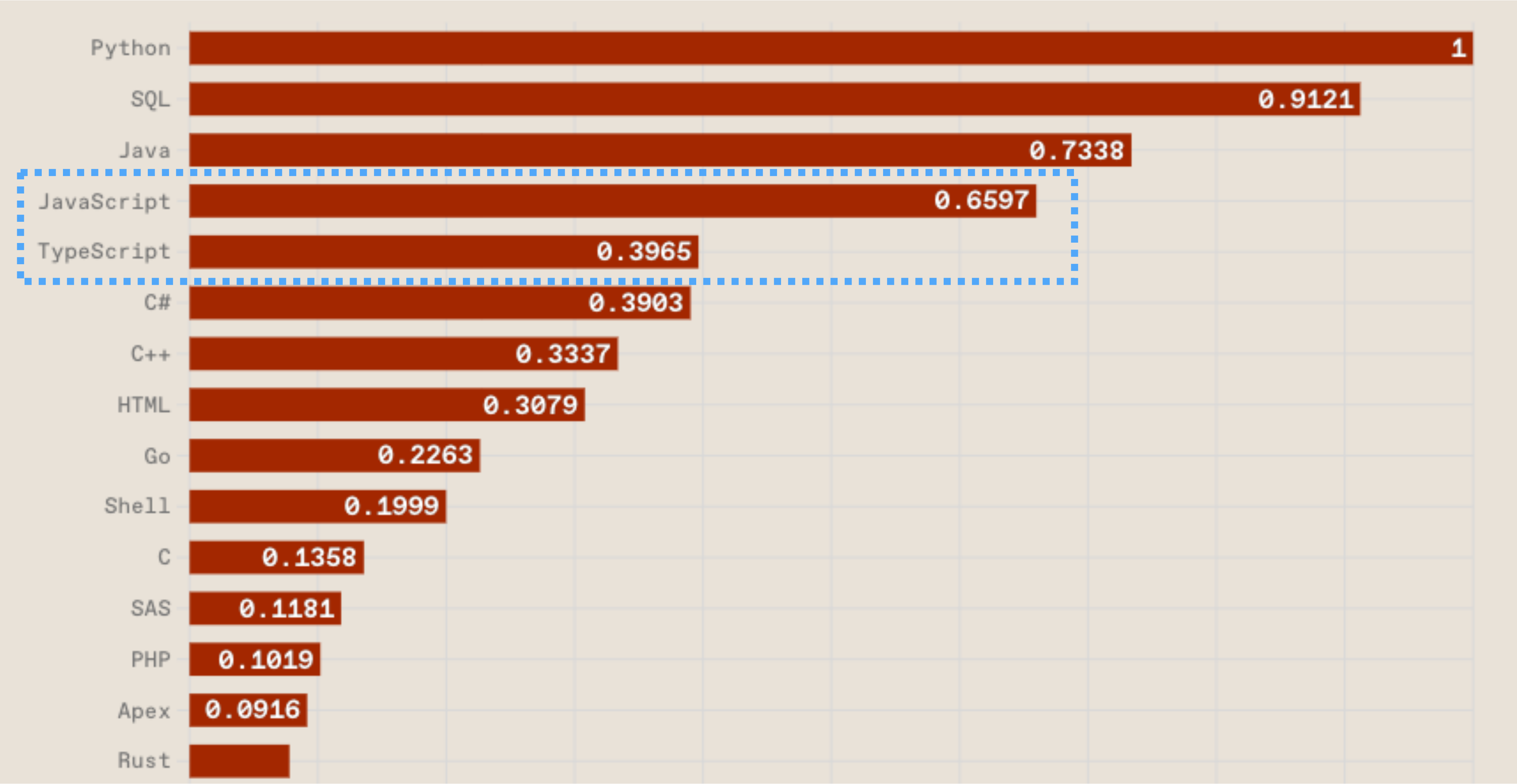
HTML, CSS and JavaScript

TypeScript

Version Control (git)

Responsive design and **accessibility**

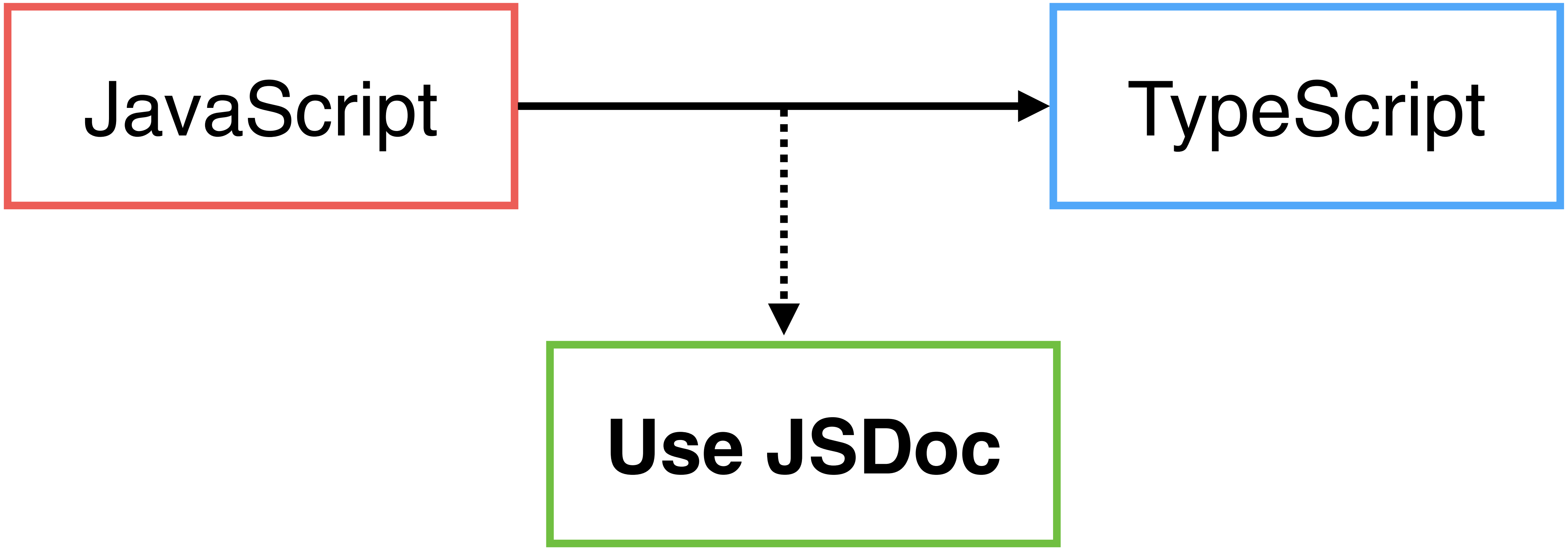
Top Programming 2025



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

Migration with incremental !!

JavaScript to TypeScript



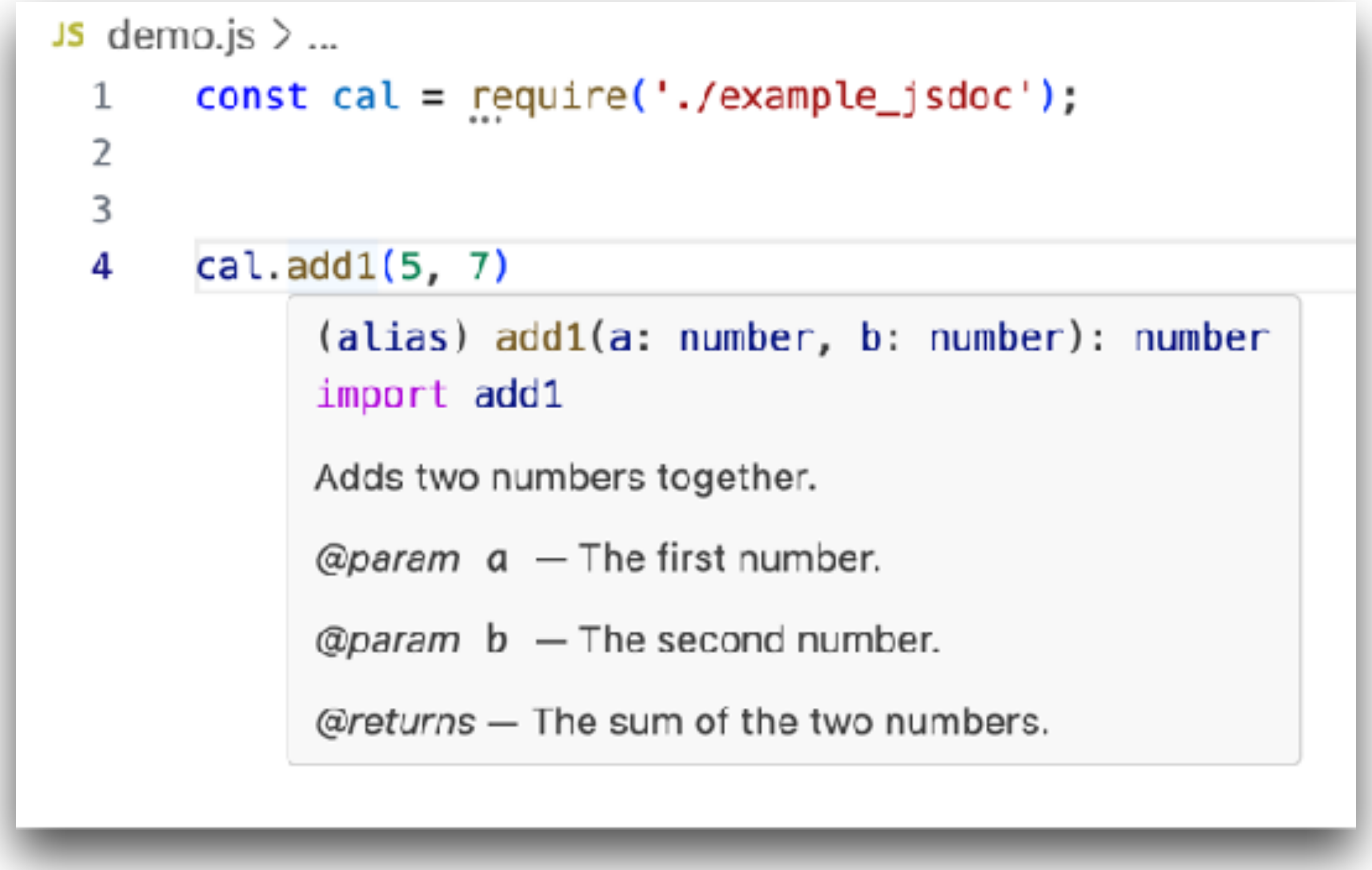
https://www.typescriptlang.org/docs/handbook/migrating-from-javascript.html

JSDoc and TSDoc

Documenting your source code

Improve readability and maintainability

IDE integration



```
const cal = require('./example_jsdoc');


cal.add1(5, 7)
```

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

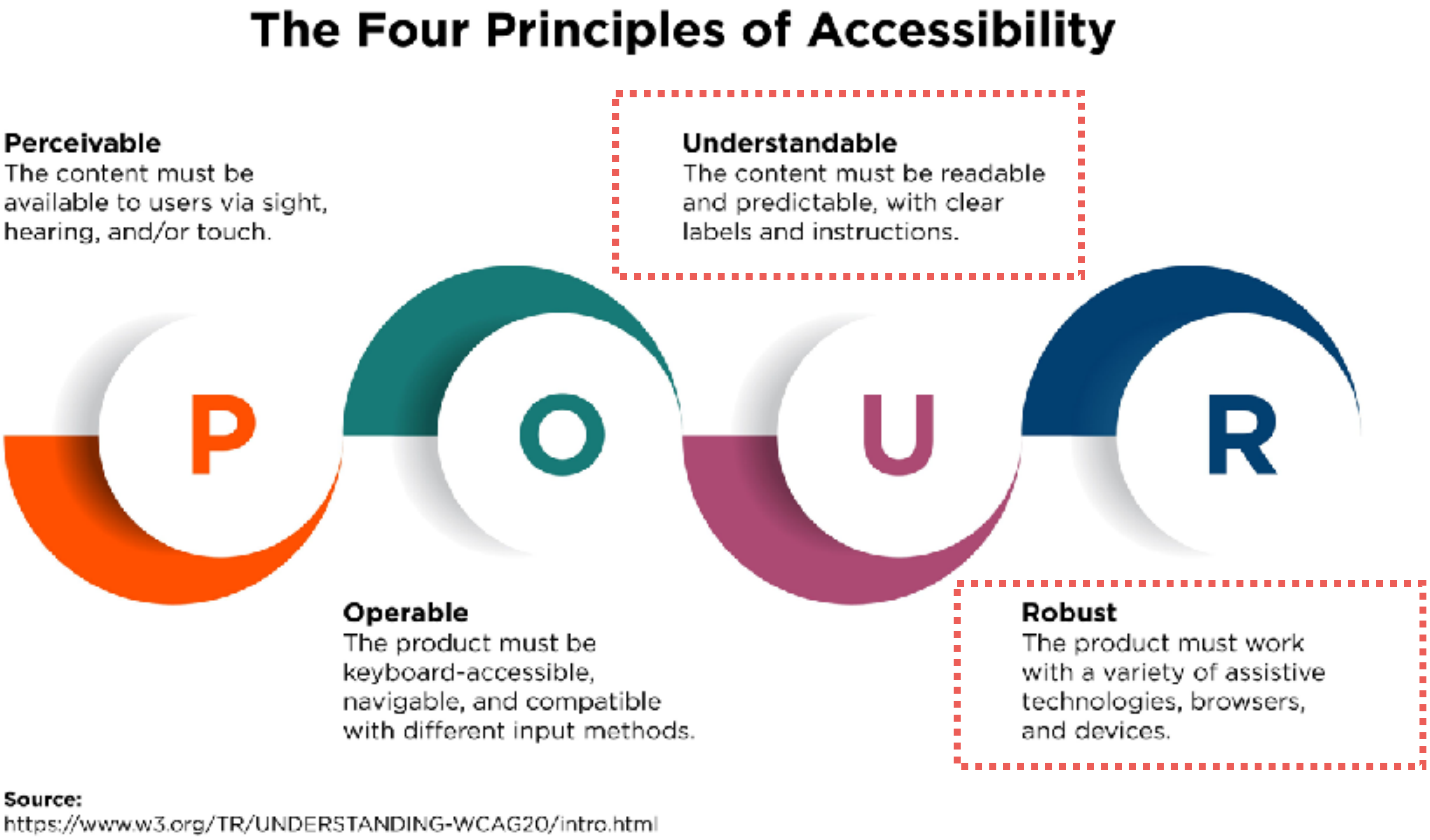
Adds two numbers together.

@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.

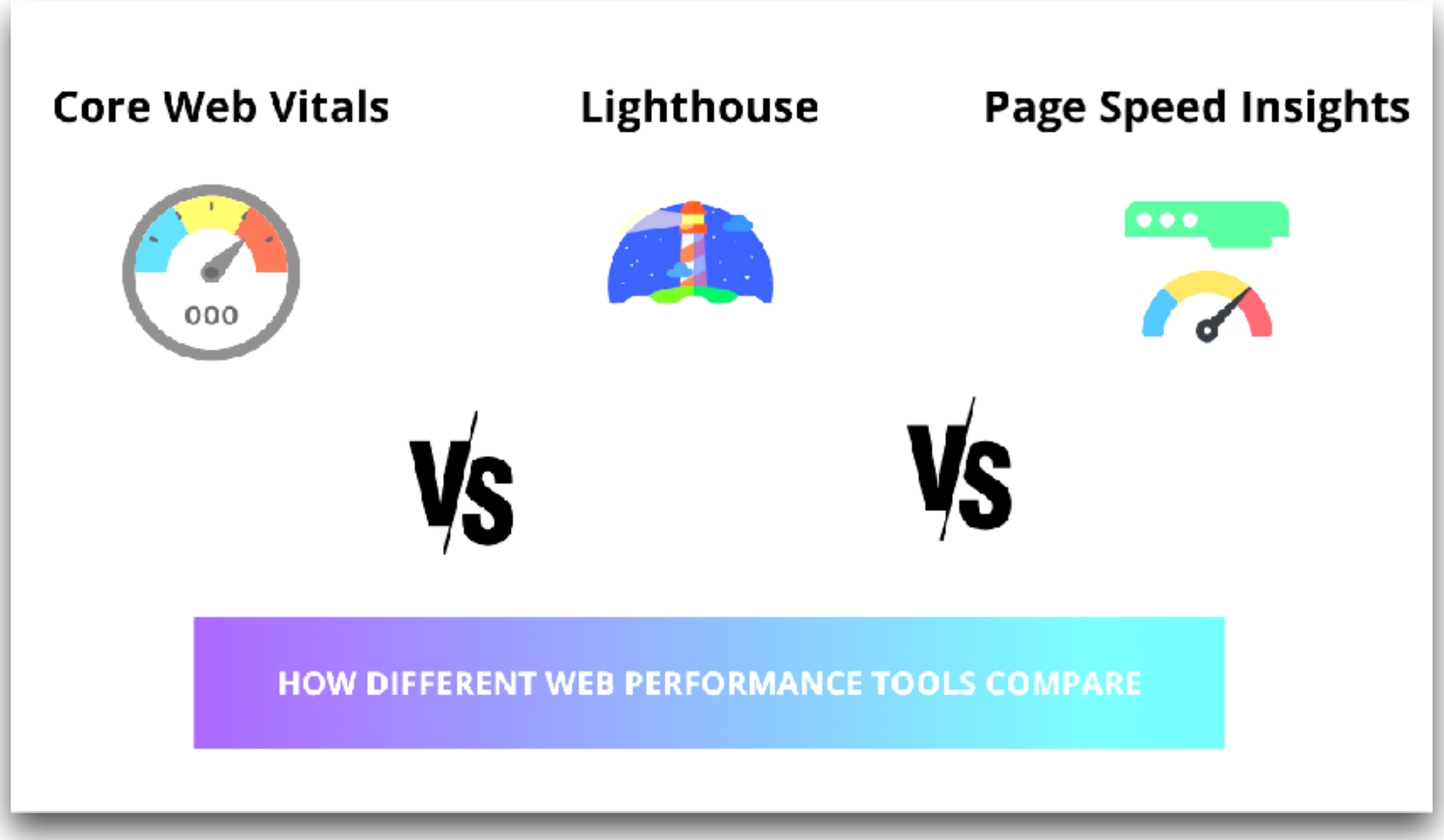
Operable
The product must be keyboard-accessible, navigable, and compatible with different input methods.

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

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

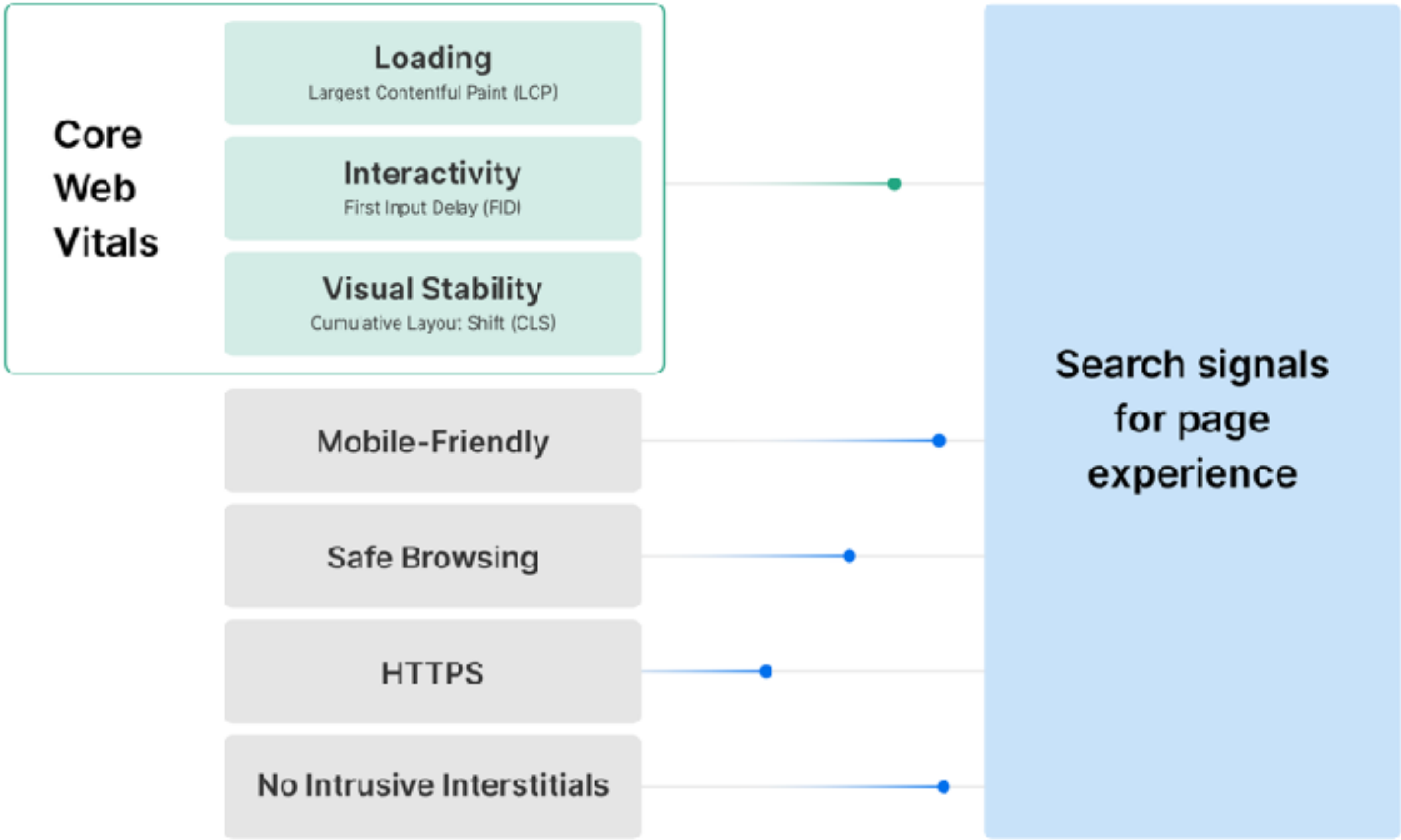
https://www.w3.org/TR/wcag-3.0/

Check your web app !!



https://learn.microsoft.com/en-us/microsoft-edge/devtools/accessibility/reference
https://www.marketingaid.io/understanding-different-performance-measurement-tools/

Core Web Vitals



Core Web Vitals

Loading
Largest Contentful Paint (LCP)

Interactivity
First Input Delay (FID)

Visual Stability
Cumulative Layout Shift (CLS)

Mobile-Friendly

Safe Browsing

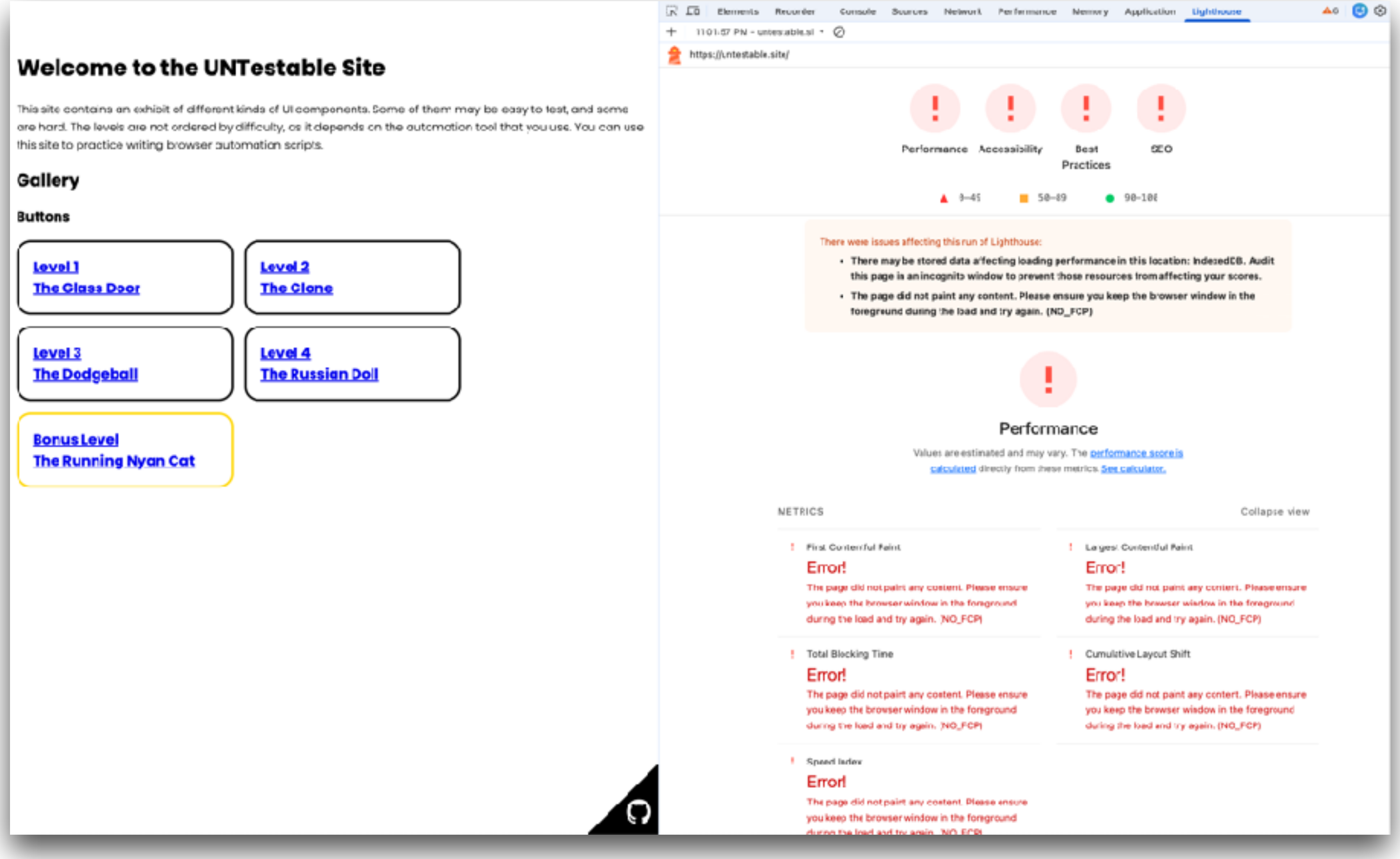
HTTPS

No Intrusive Interstitials

Search signals for page experience

https://web.dev/explore/learn-core-web-vitals
https://developer.chrome.com/docs/crux/

Lighthouse



11:01:57 PM - untestable.si

https://untestable.site/

Welcome to the UNTestable Site

This site contains an exhibit of different kinds of UI components. Some of them may be easy to test, and some are hard. The levels are not ordered by difficulty, as it depends on the automation tool that you use. You can use this site to practice writing browser automation scripts.

Gallery

Buttons

Level 1
The Glass Door

Level 2
The Clone

Level 3
The Dodgeball

Level 4
The Russian Doll

Bonus Level
The Running Nyan Cat

▲ 0–49 ■ 50–89 ● 90–100

There were issues affecting this run of Lighthouse:

- 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

! 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)

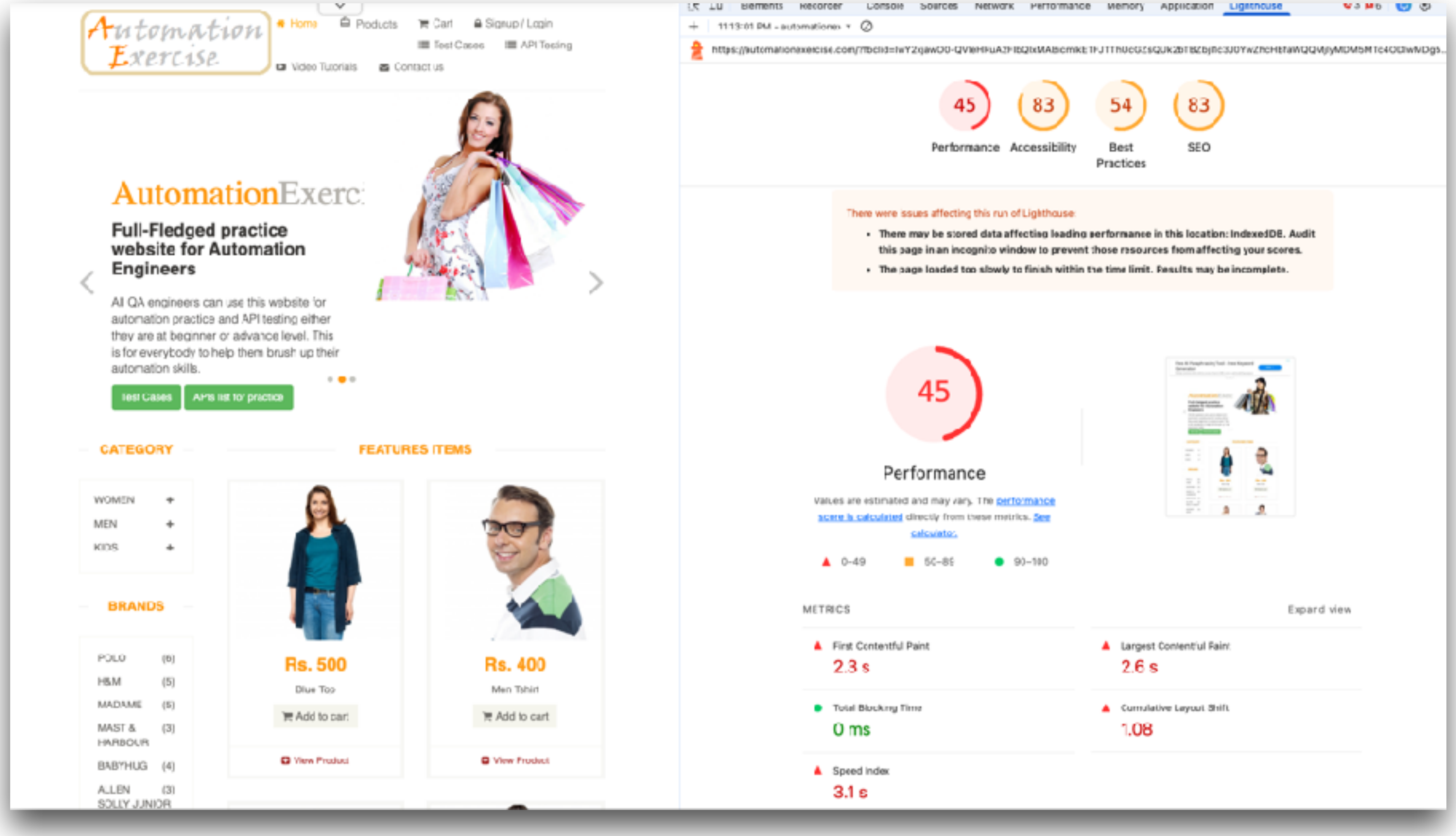
! Largest Contentful Paint
Error!
The page did not paint any content. Please ensure you keep the browser window in the foreground during the load and try again. (NO_FCP)

! 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



AutomationExerc

Full-Fledged practice website for Automation Engineers

All QA engineers can use this website for automation practice and API testing either they are at beginner or advance level. This is for everybody to help them brush up their automation skills.

11:13:01 PM - automationex

https://automationexercise.com/?fbclid=IwY2qawO0-QVleHFuA7FlBQIxMABcmkE1FJTThUcGZsQUk2bTBZbjlic3J0YWZhcHBfaWQQMjlyMDM5MTc4ODIwNDgs...

45	83	54	83
Performance	Accessibility	Best Practices	SEO

There were issues affecting this run of Lighthouse:

- 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 loaded too slowly to finish within the time limit. Results may be incomplete.

45

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

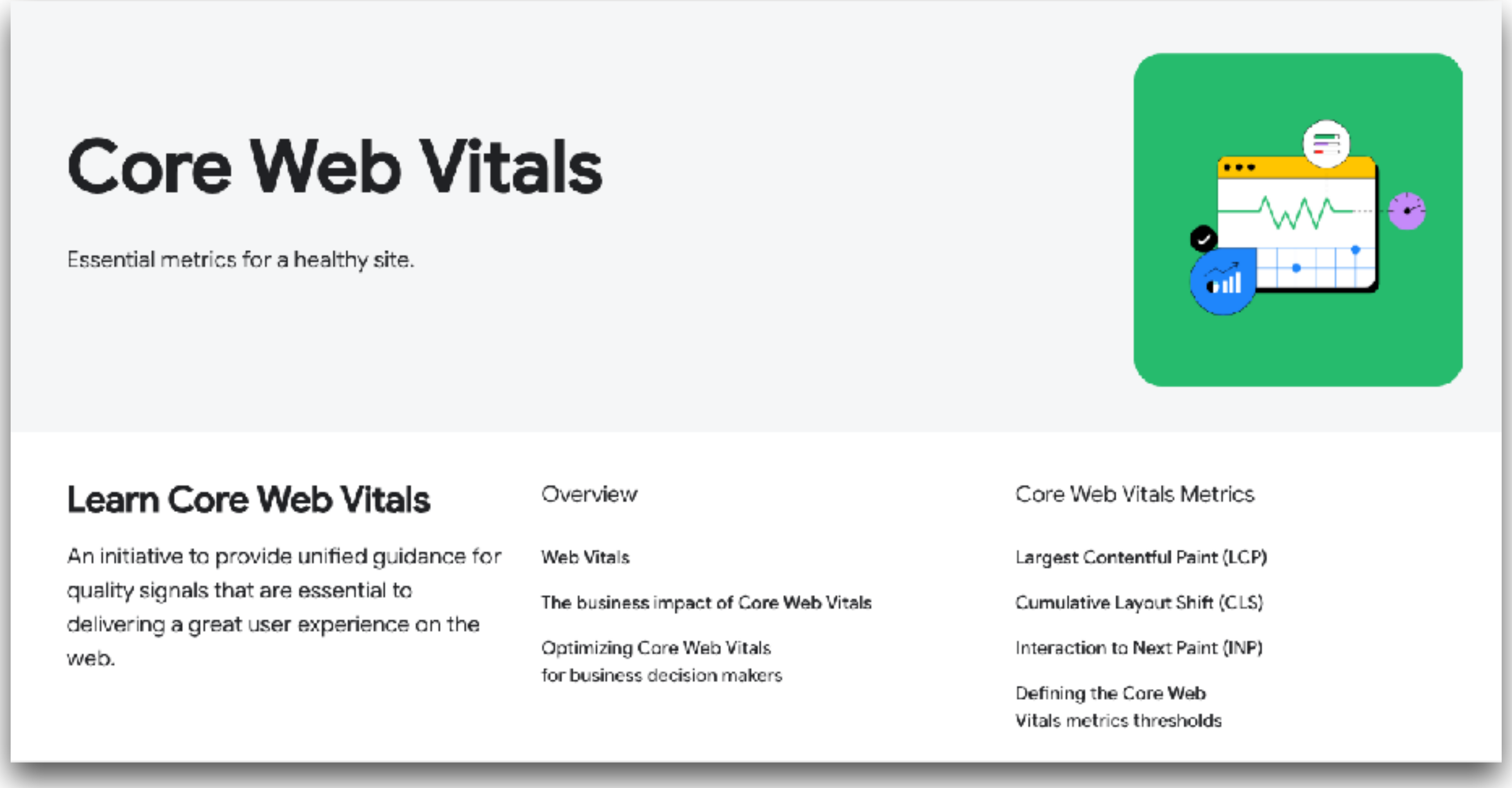
▲ 0–49 ■ 50–89 ● 90–100

METRICS

Expand view

▲ First Contentful Paint 2.3 s	▲ Largest Contentful Paint 2.6 s
● Total Blocking Time 0 ms	▲ Cumulative Layout Shift 1.08
▲ Speed Index 3.1 s	

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

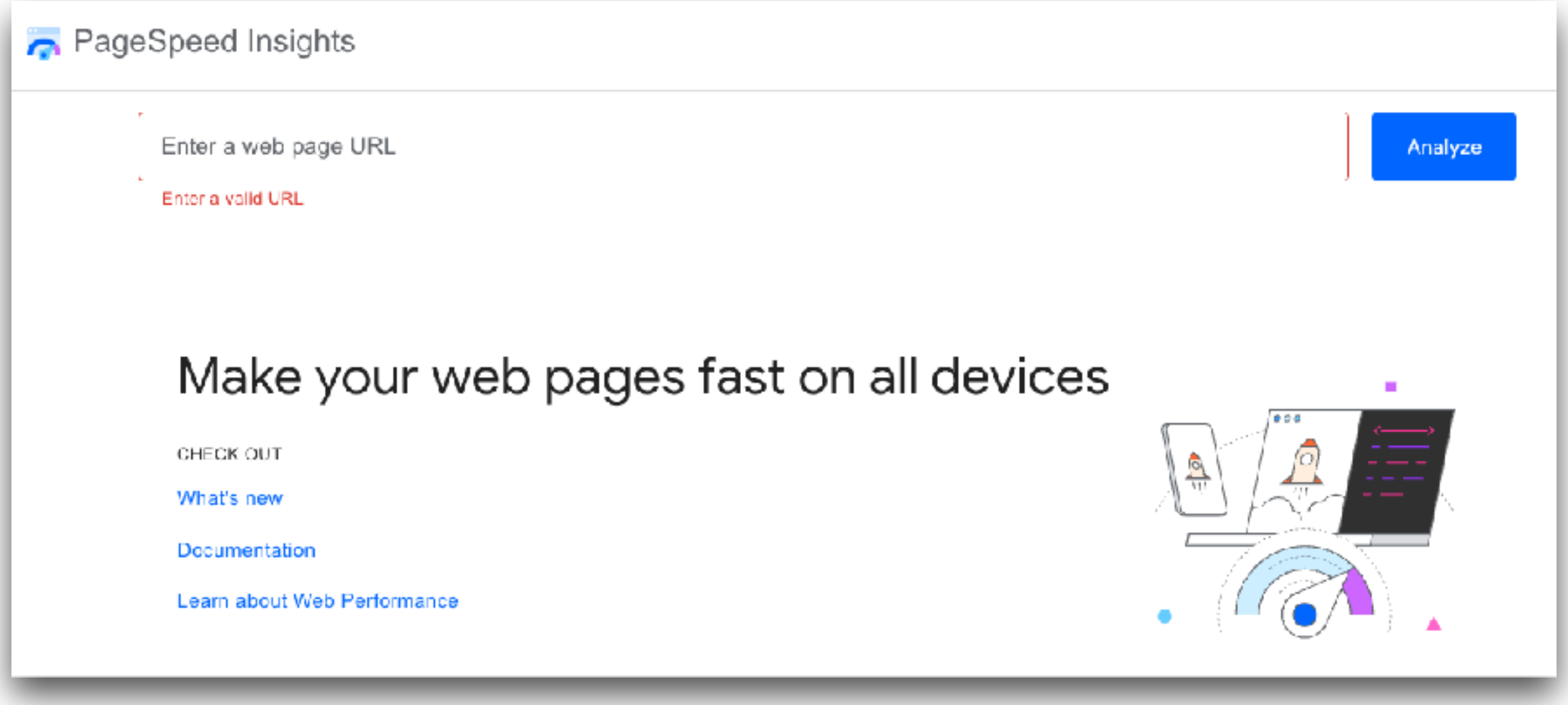
https://web.dev/explore/learn-core-web-vitals

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

Enter a valid URL

Make your web pages fast on all devices

CHECK OUT

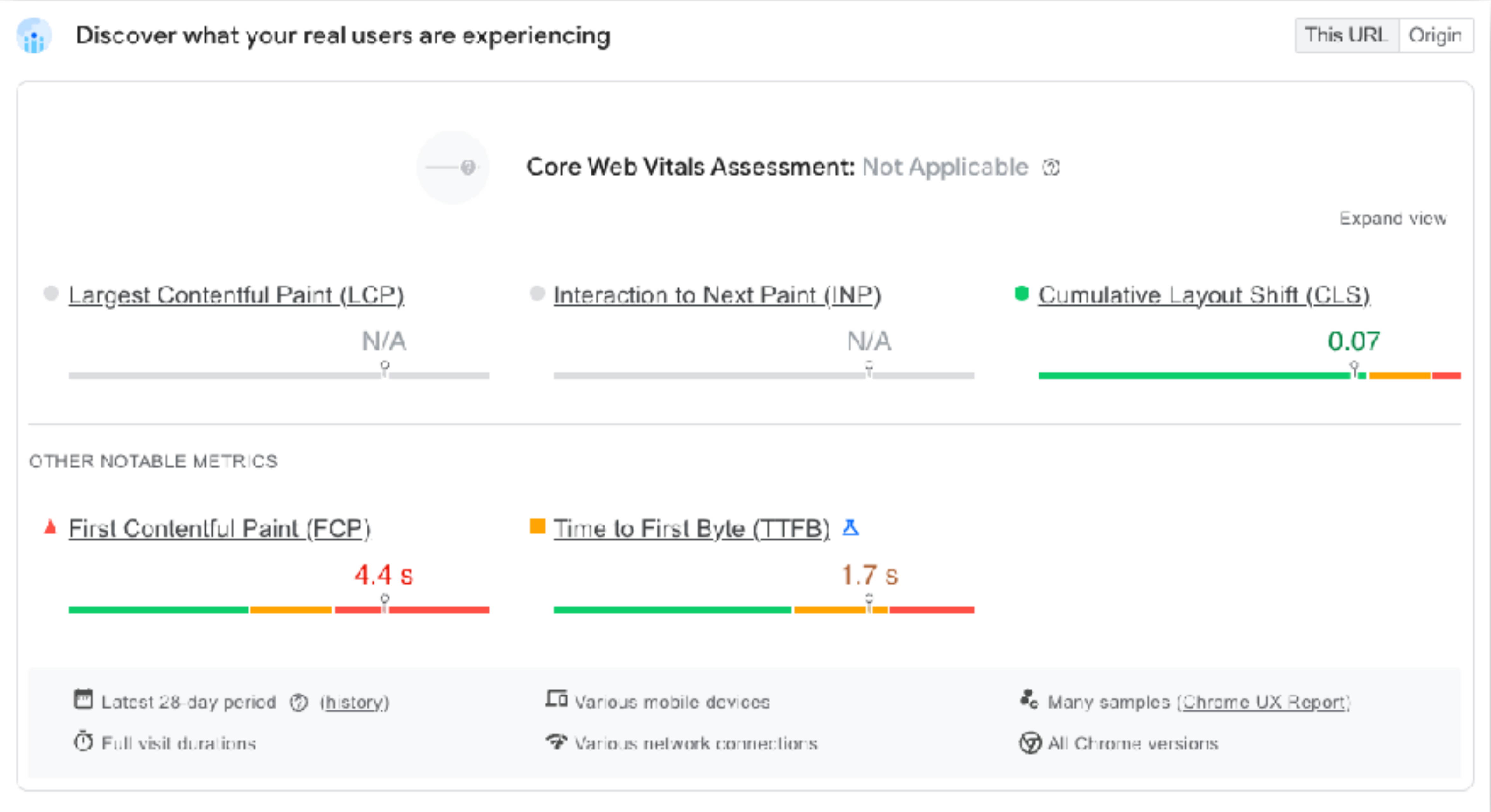
What's new

Documentation

Learn about Web Performance

https://pagespeed.web.dev/

Result !!



Discover what your real users are experiencing

Core Web Vitals Assessment: Not Applicable

Largest Contentful Paint (LCP)
N/A

Interaction to Next Paint (INP)
N/A

Cumulative Layout Shift (CLS)
0.07

OTHER NOTABLE METRICS

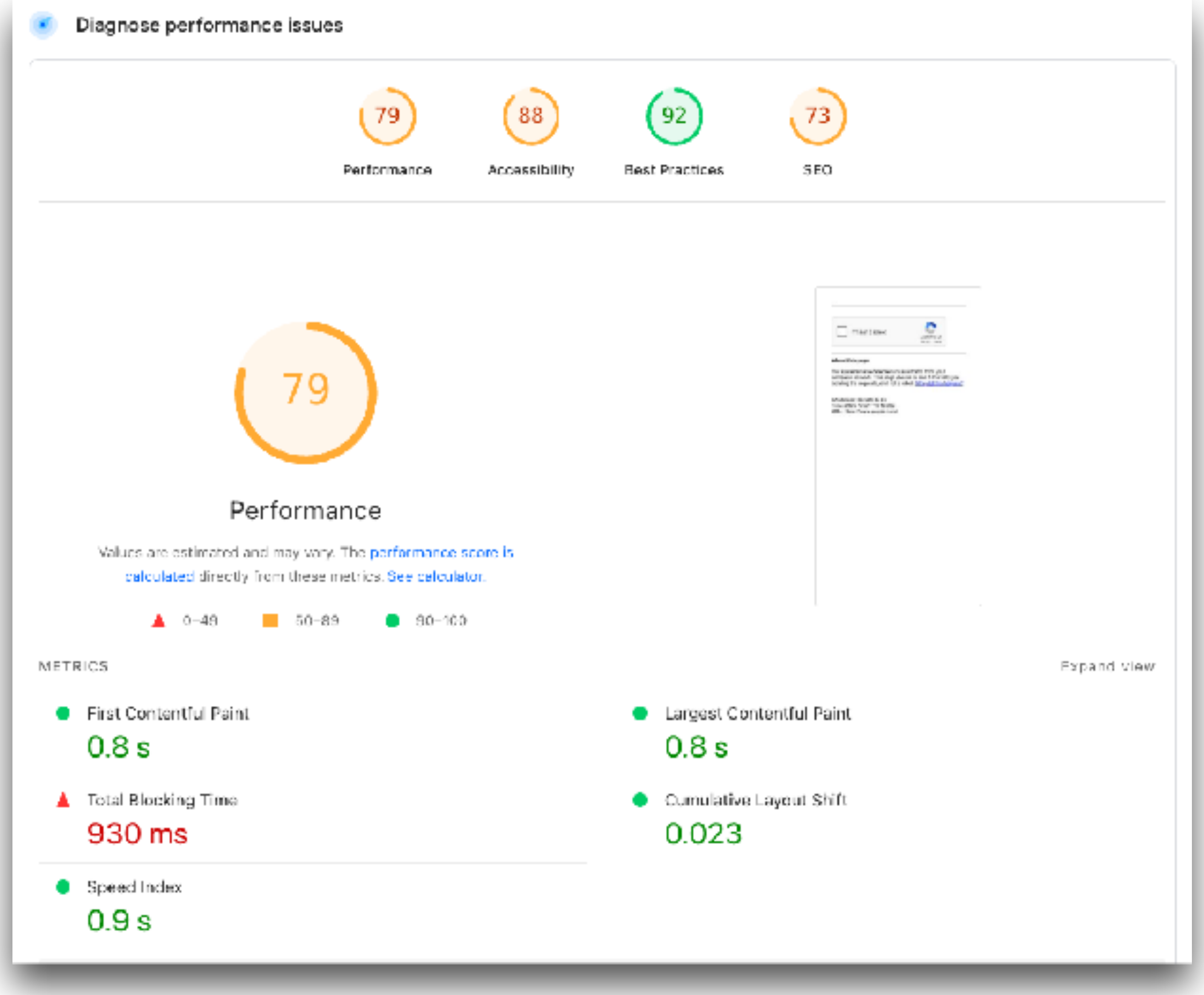
First Contentful Paint (FCP)
4.4 s

Time to First Byte (TTFB)
1.7 s

Latest 28-day period (history)
Full visit durations
Various mobile devices
Various network connections
Many samples (Chrome UX Report)
All Chrome versions

https://pagespeed.web.dev/

Result !!



Diagnose performance issues

79	88	92	73
Performance	Accessibility	Best Practices	SEO

79

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 ■ 50–89 ● 90–100

METRICS

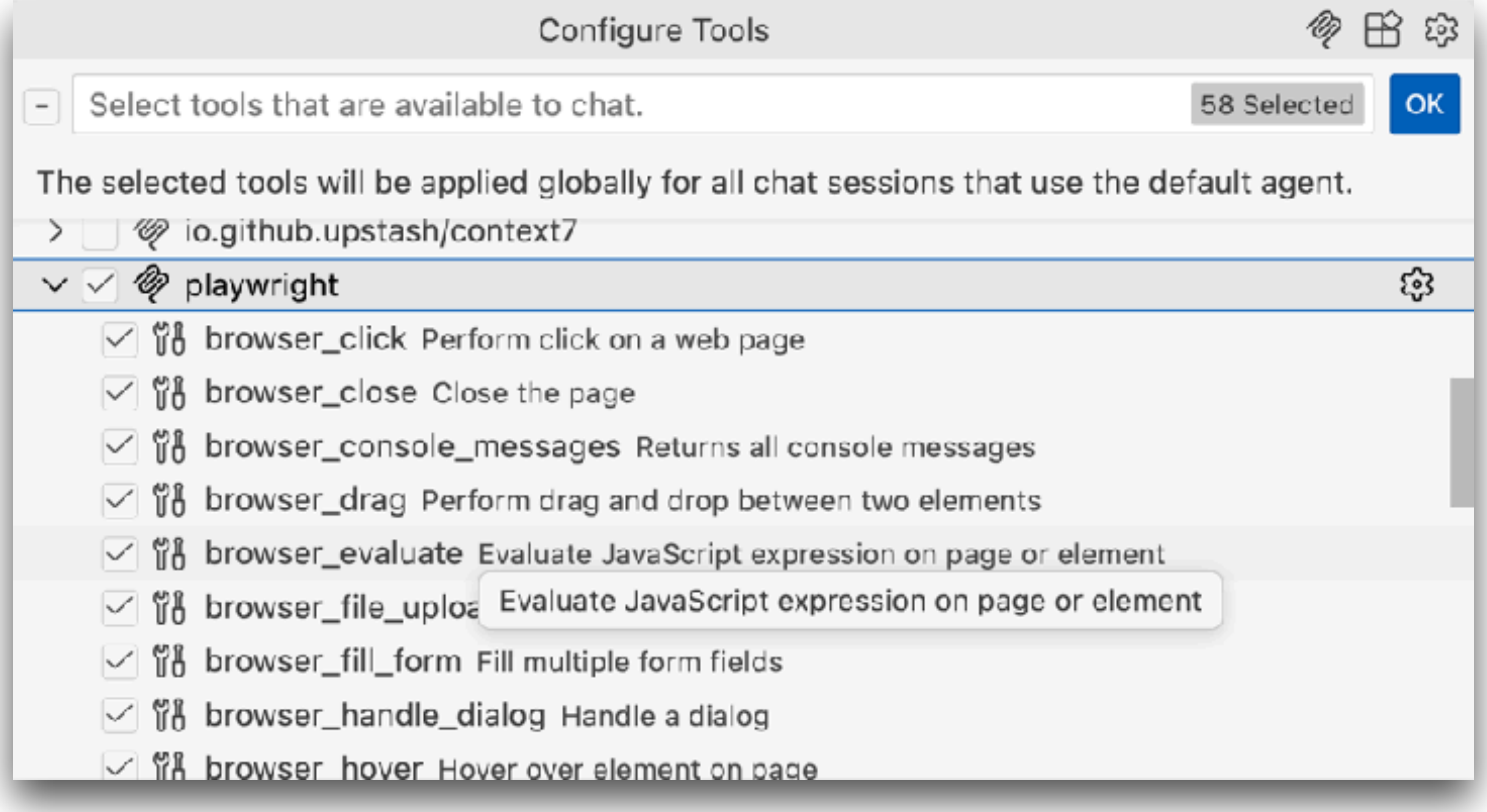
Expand view

● First Contentful Paint 0.8 s	● Largest Contentful Paint 0.8 s
▲ Total Blocking Time 930 ms	● Cumulative Layout Shift 0.023
● Speed Index 0.9 s	

https://pagespeed.web.dev/

Try to testing with AI !!

Playwright MCP from Microsoft



Select tools that are available to chat.

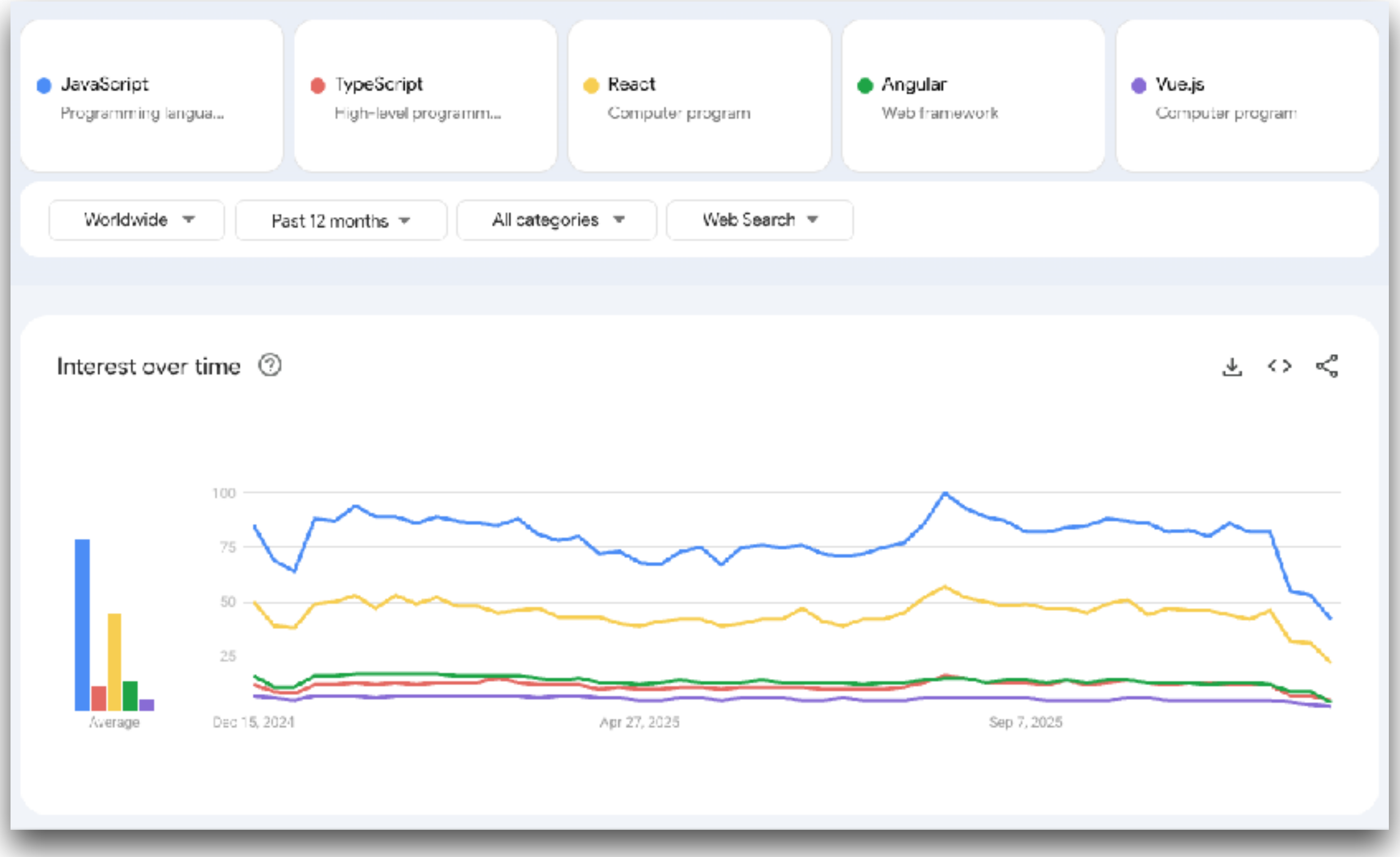
The selected tools will be applied globally for all chat sessions that use the default agent.

- □ io.github.upstash/context7
- ✓ playwright
  - ✓ browser_click Perform click on a web page
  - ✓ browser_close Close the page
  - ✓ browser_console_messages Returns all console messages
  - ✓ browser_drag Perform drag and drop between two elements
  - ✓ browser_evaluate Evaluate JavaScript expression on page or element
  - ✓ browser_file_uploa Evaluate JavaScript expression on page or element
  - ✓ browser_fill_form Fill multiple form fields
  - ✓ browser_handle_dialog Handle a dialog
  - ✓ browser_hover Hover over element on page

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

Key Frameworks and Libraries

Google Trends !!



● JavaScript
Programming langua...

● TypeScript
High-level programm...

● React
Computer program

● Angular
Web framework

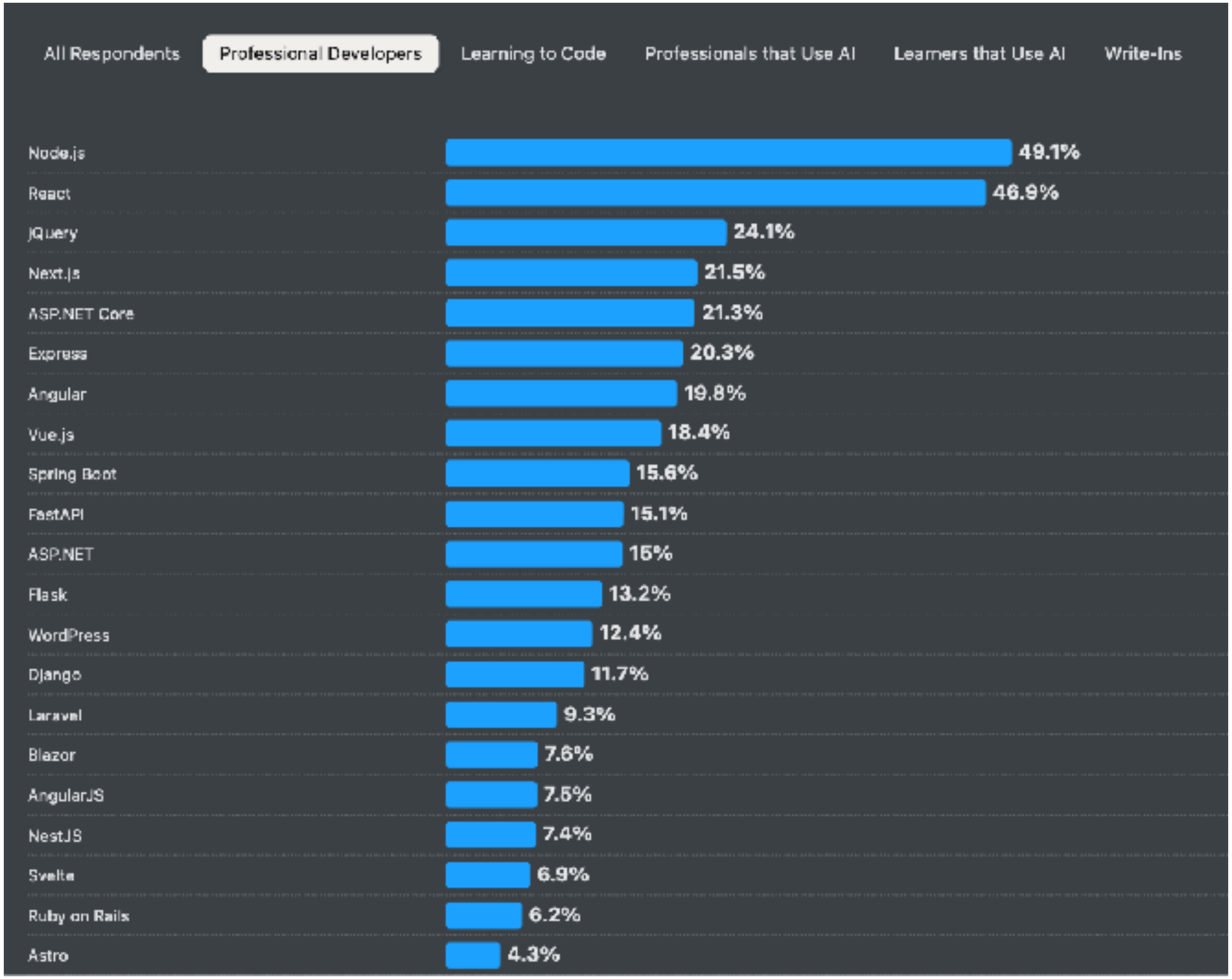
● Vue.js
Computer program

Worldwide ▾ | Past 12 months ▾ | All categories ▾ | Web Search ▾

Interest over time

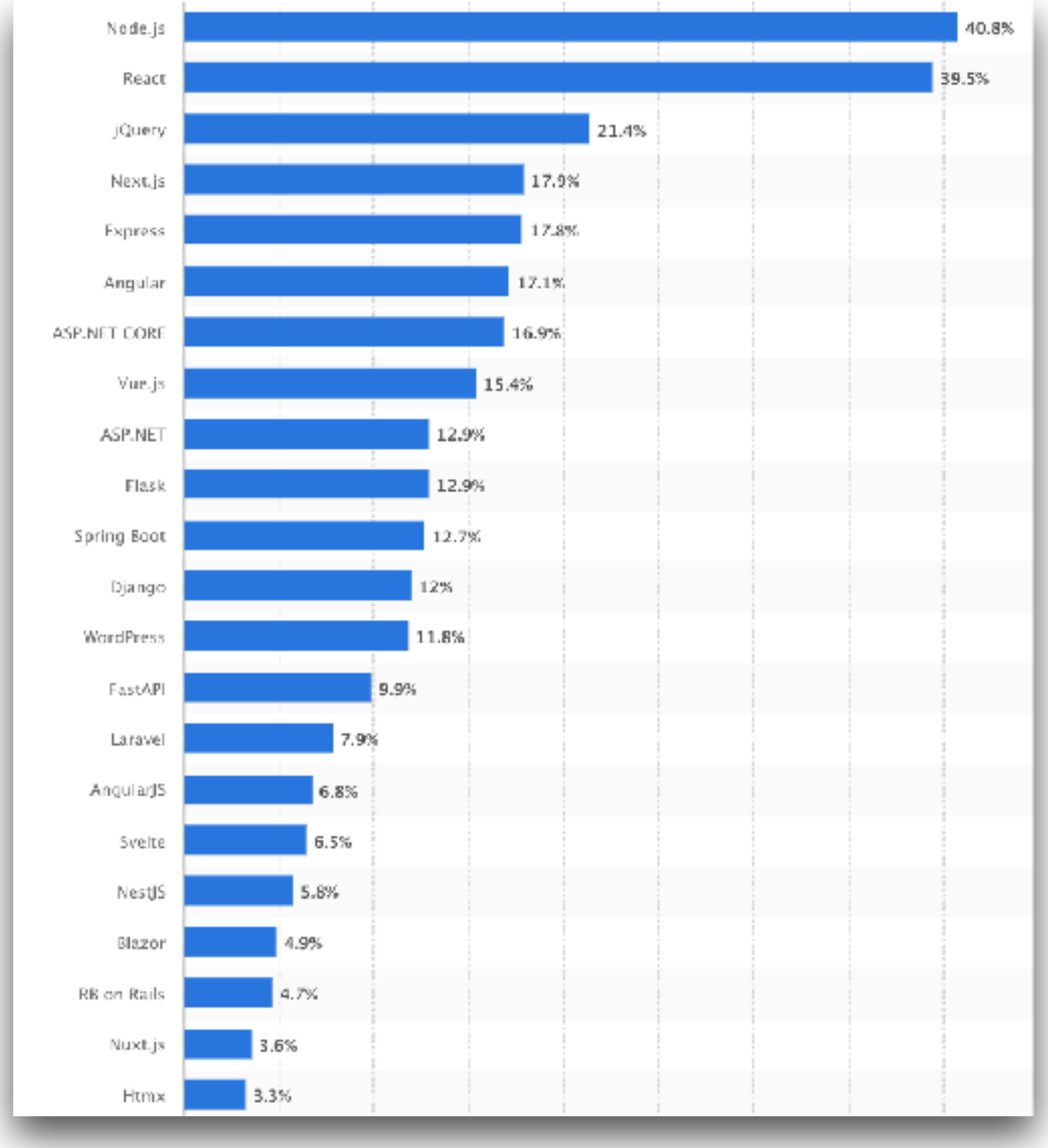
https://trends.google.com/

Top Web Frameworks



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

Most used web framework 2024



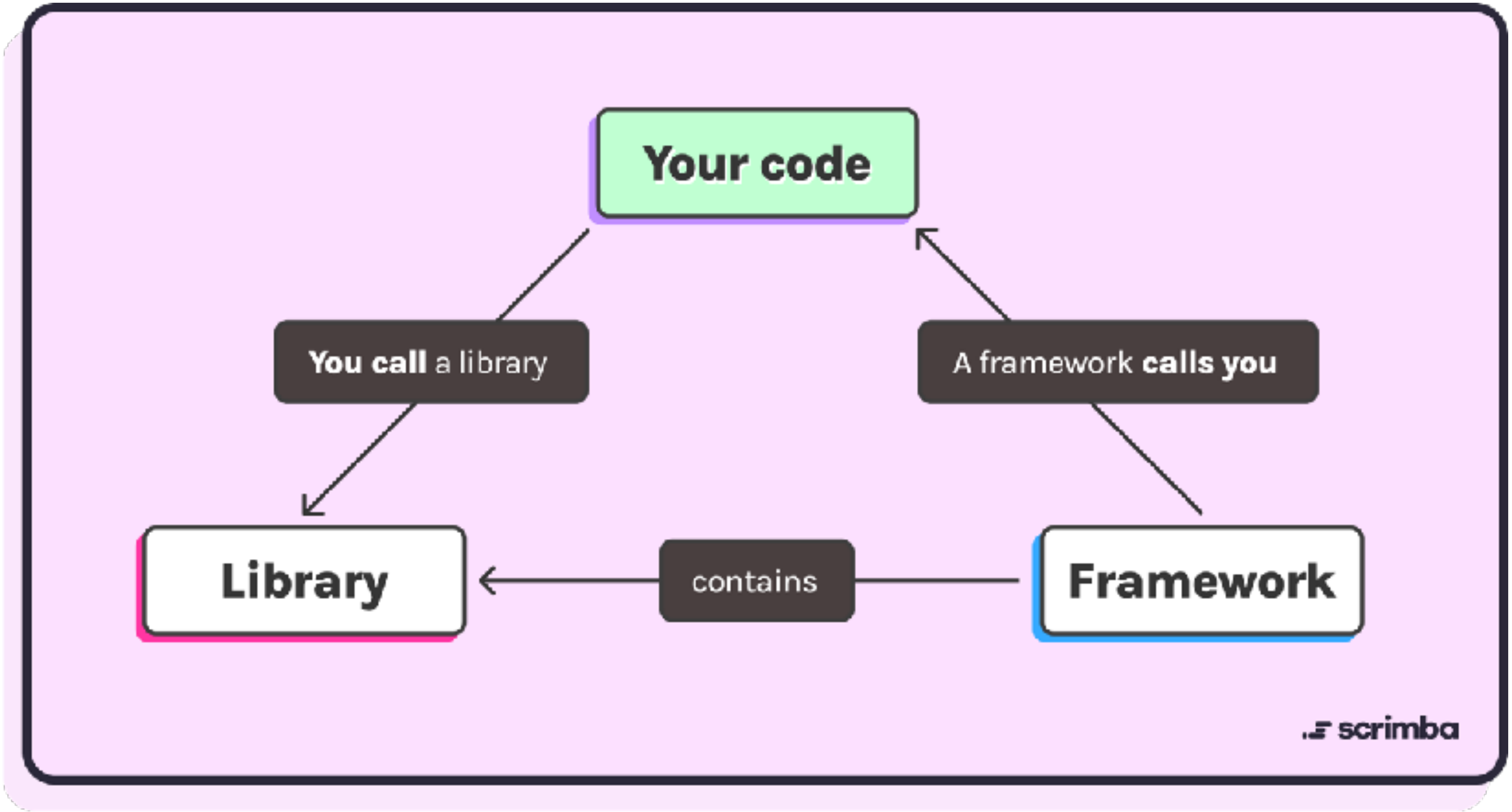
https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/

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 ?

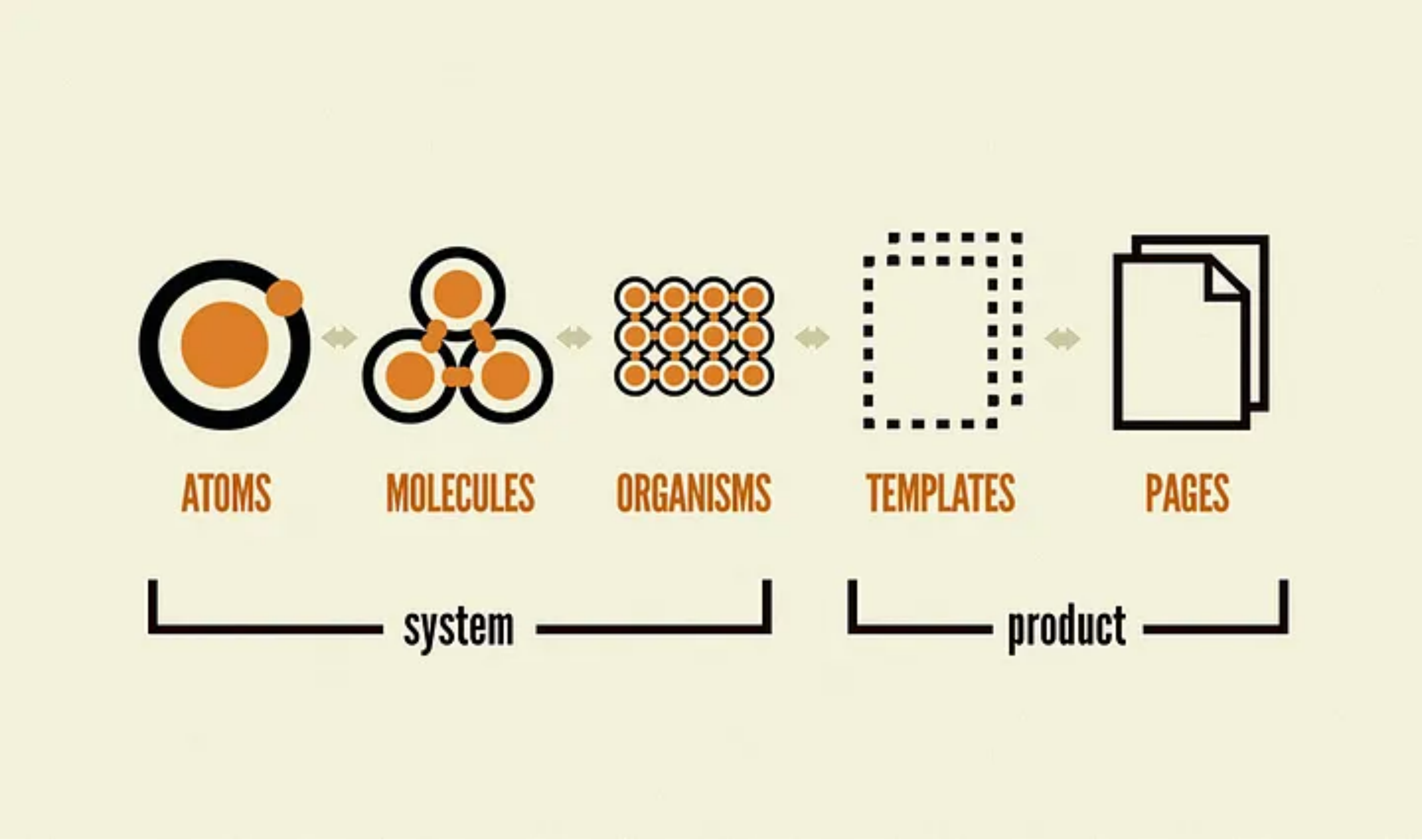


https://v1.scrimba.com/articles/difference-between-framework-and-library/

Component-Driven

https://www.componentdriven.org/

Atomic Design



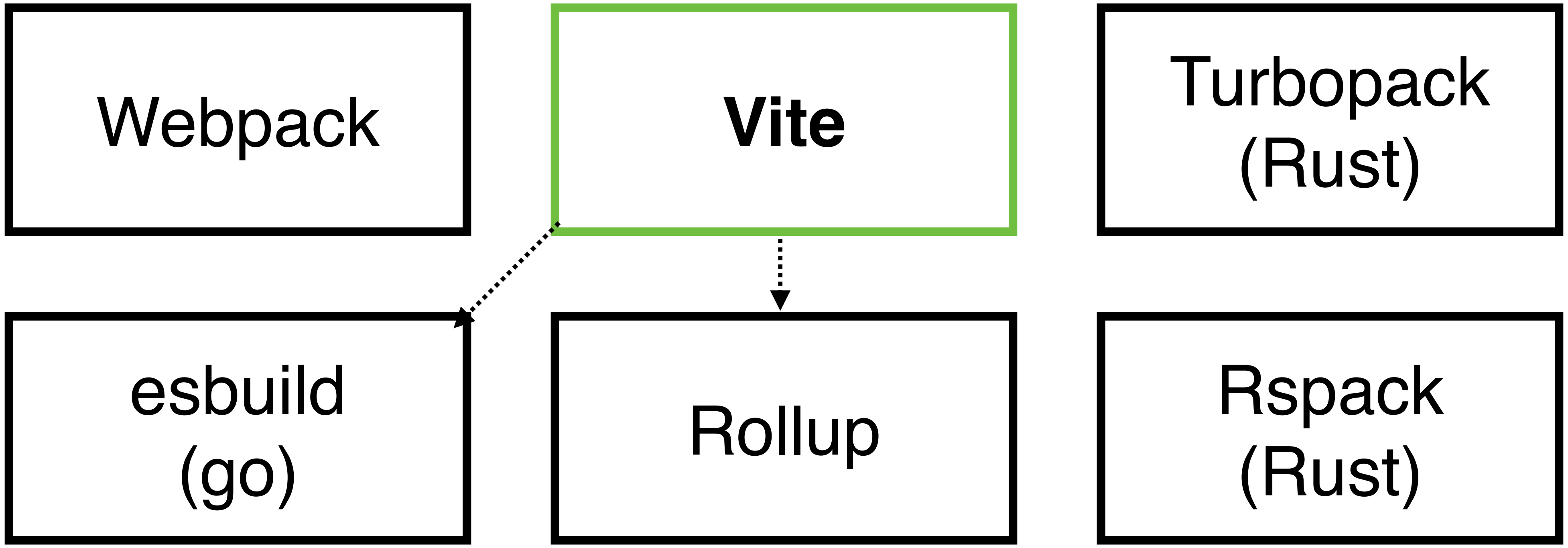
https://atomicdesign.bradfrost.com/

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

Build Tools for web framework ?



Webpack

Vite

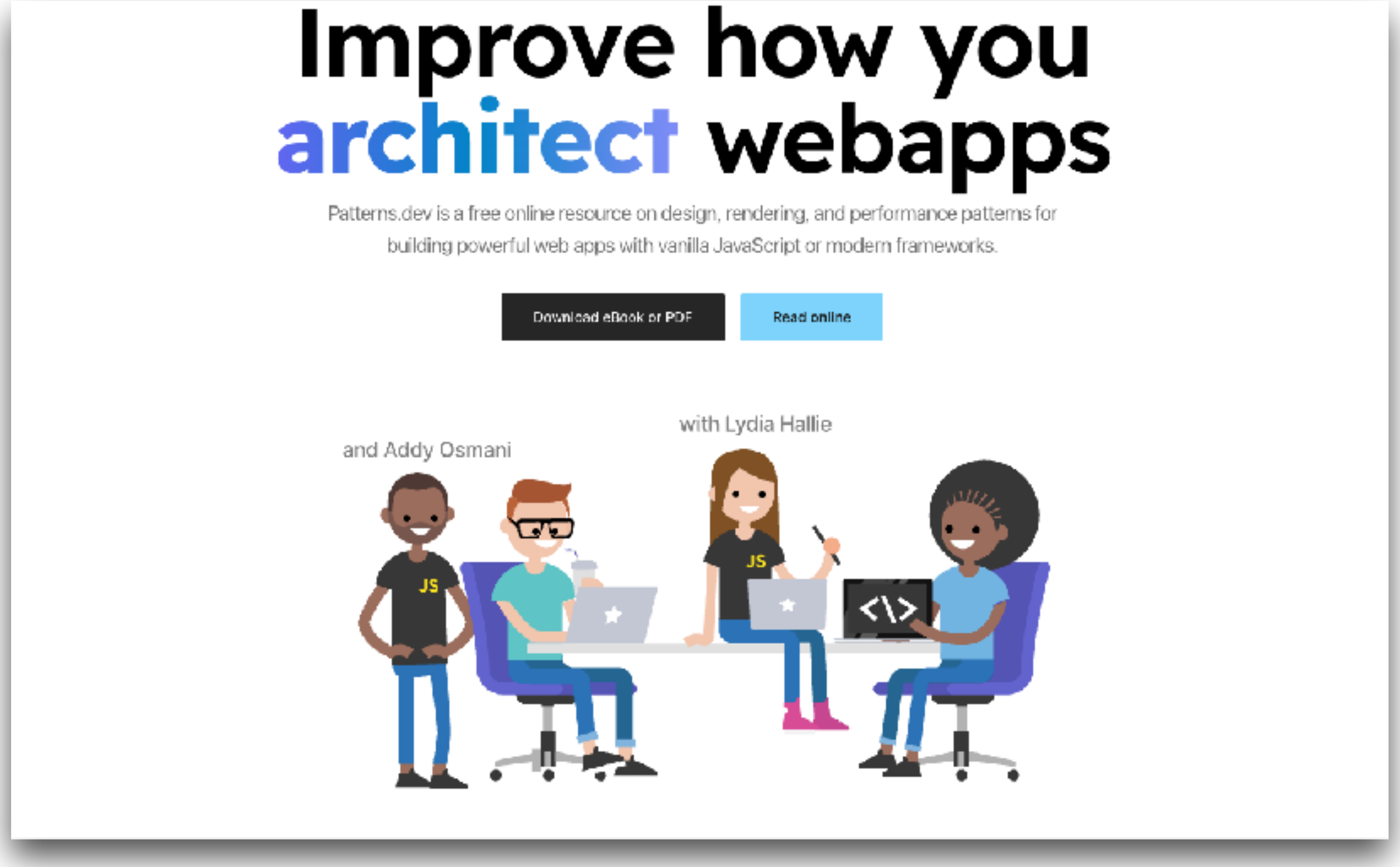
Turbopack (Rust)

esbuild (go)

Rollup

Rspack (Rust)

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.

https://www.patterns.dev/

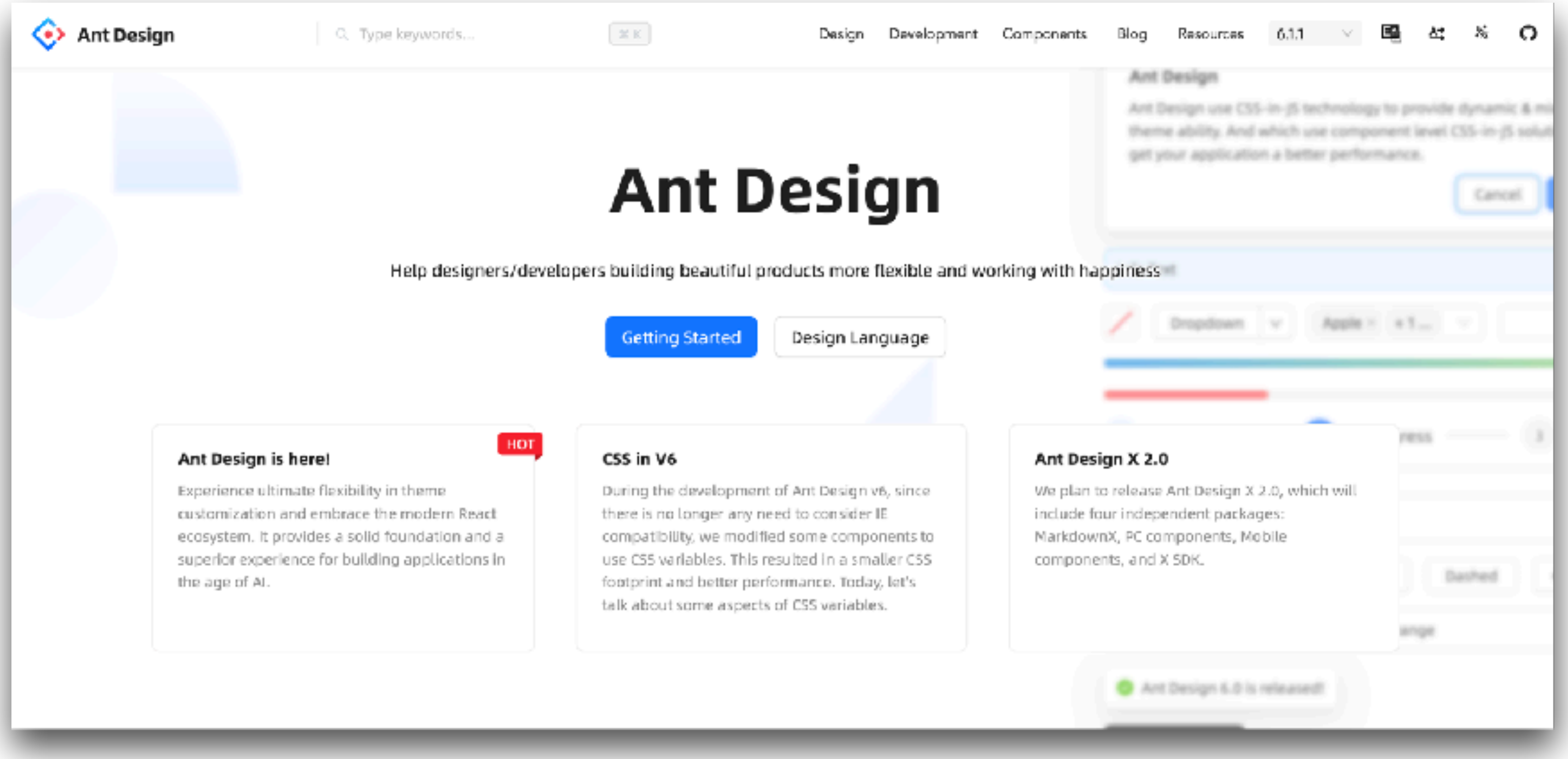
Trends and Tools

Trends and Tools

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

Ant Design

Open source UI design system



Ant Design

Help designers/developers building beautiful products more flexible and working with happiness

Ant Design is here!

Experience ultimate flexibility in theme customization and embrace the modern React ecosystem. It provides a solid foundation and a superior experience for building applications in the age of AI.

CSS in V6

During the development of Ant Design v6, since there is no longer any need to consider IE compatibility, we modified some components to use CSS variables. This resulted in a smaller CSS footprint and better performance. Today, let's talk about some aspects of CSS variables.

Ant Design X 2.0

We plan to release Ant Design X 2.0, which will include four independent packages: MarkdownX, PC components, Mobile components, and X SDK.

Ant Design use CSS-in-JS technology to provide dynamic & mi theme ability. And which use component level CSS-in-JS solut get your application a better performance.

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

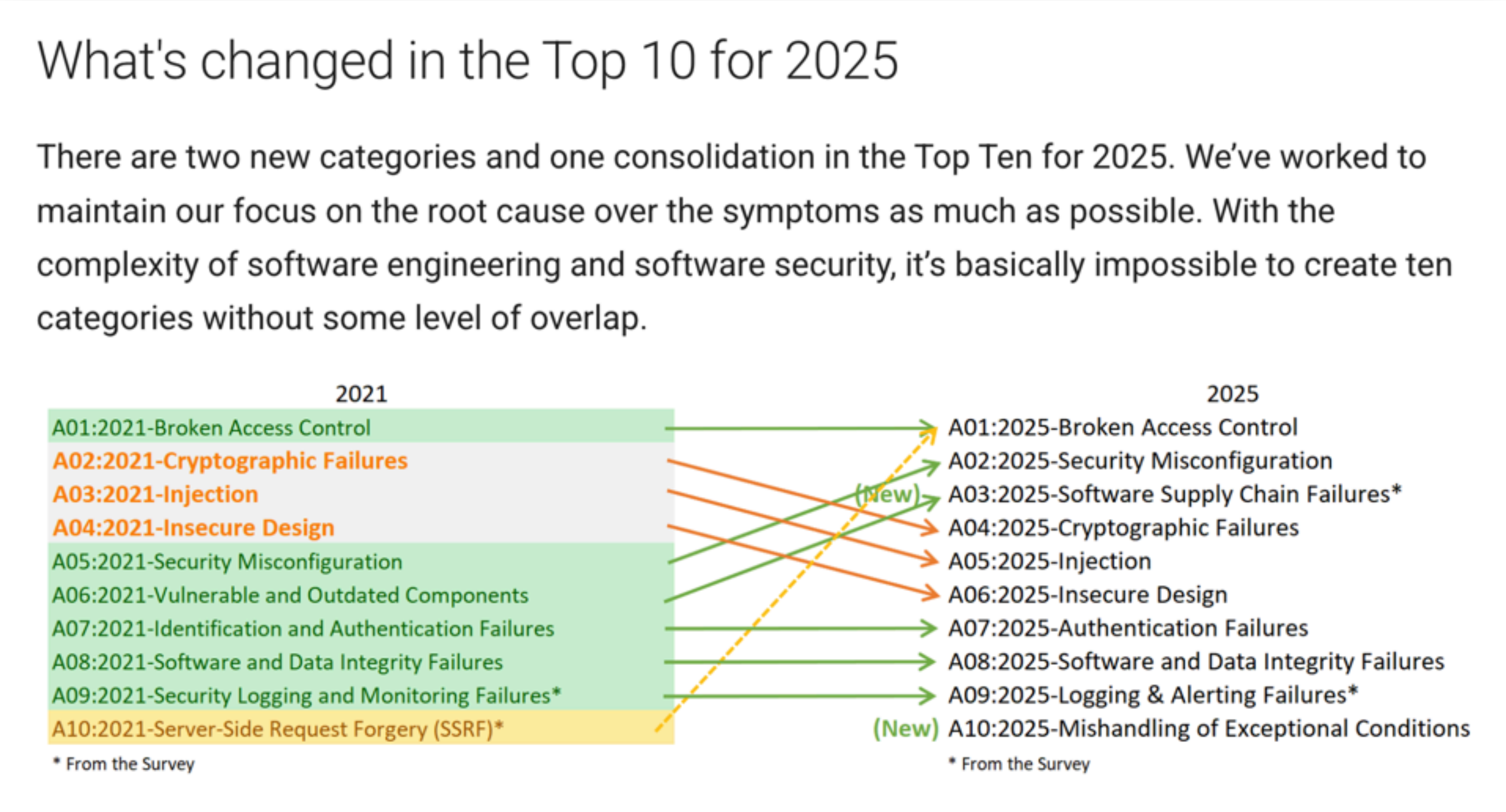
Use analytics

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.

2021

- A01:2021-Broken Access Control
- A02:2021-Cryptographic Failures
- A03:2021-Injection
- A04:2021-Insecure Design
- A05:2021-Security Misconfiguration
- A06:2021-Vulnerable and Outdated Components
- A07:2021-Identification and Authentication Failures
- A08:2021-Software and Data Integrity Failures
- A09:2021-Security Logging and Monitoring Failures*
- A10:2021-Server-Side Request Forgery (SSRF)*

* From the Survey

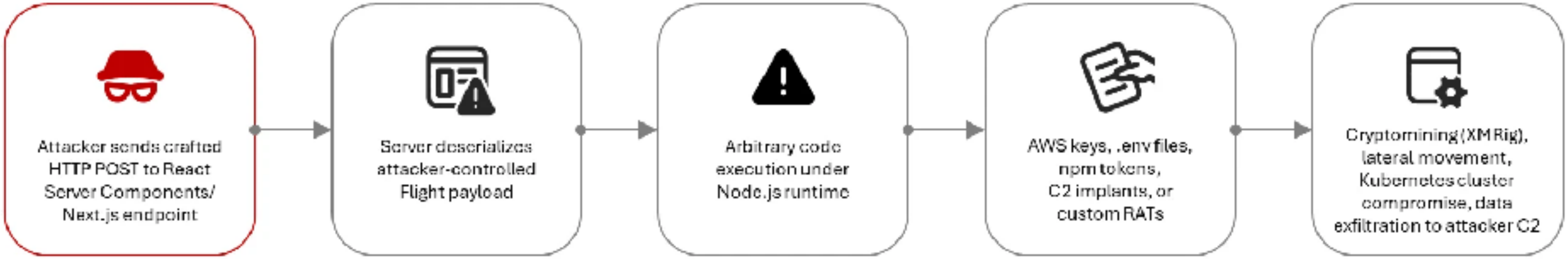
2025

- A01:2025-Broken Access Control
- A02:2025-Security Misconfiguration
- (New) A03:2025-Software Supply Chain Failures*
- A04:2025-Cryptographic Failures
- A05:2025-Injection
- A06:2025-Insecure Design
- A07:2025-Authentication Failures
- A08:2025-Software and Data Integrity Failures
- A09:2025-Logging & Alerting Failures*
- (New) A10:2025-Mishandling of Exceptional Conditions

* From the Survey

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

React2shell ?



Attacker sends crafted HTTP POST to React Server Components/ Next.js endpoint

Server deserializes attacker-controlled Flight payload

Arbitrary code execution under Node.js runtime

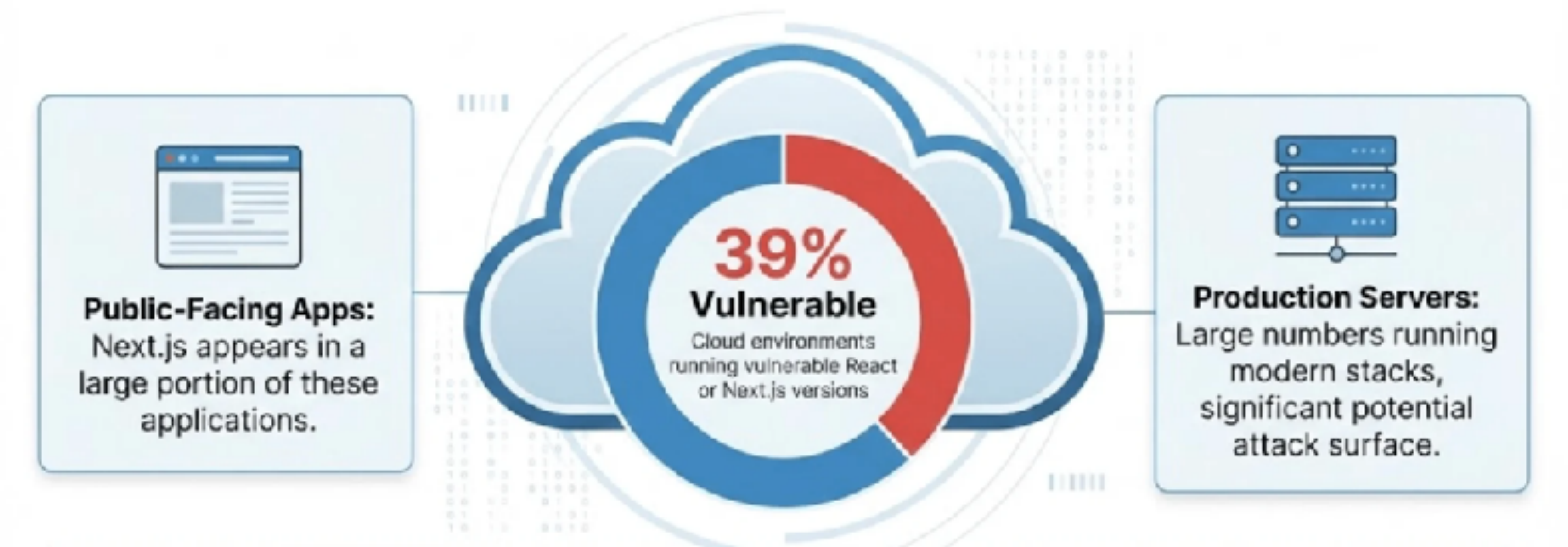
AWS keys, .env files, npm tokens, C2 implants, or custom RATs

Cryptomining (XMRig), lateral movement, Kubernetes cluster compromise, data exfiltration to attacker C2

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**


Public-Facing Apps:
Next.js appears in a large portion of these applications.
39%
Vulnerable
Cloud environments running vulnerable React or Next.js versions
Production Servers:
Large numbers running modern stacks, significant potential attack surface.

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

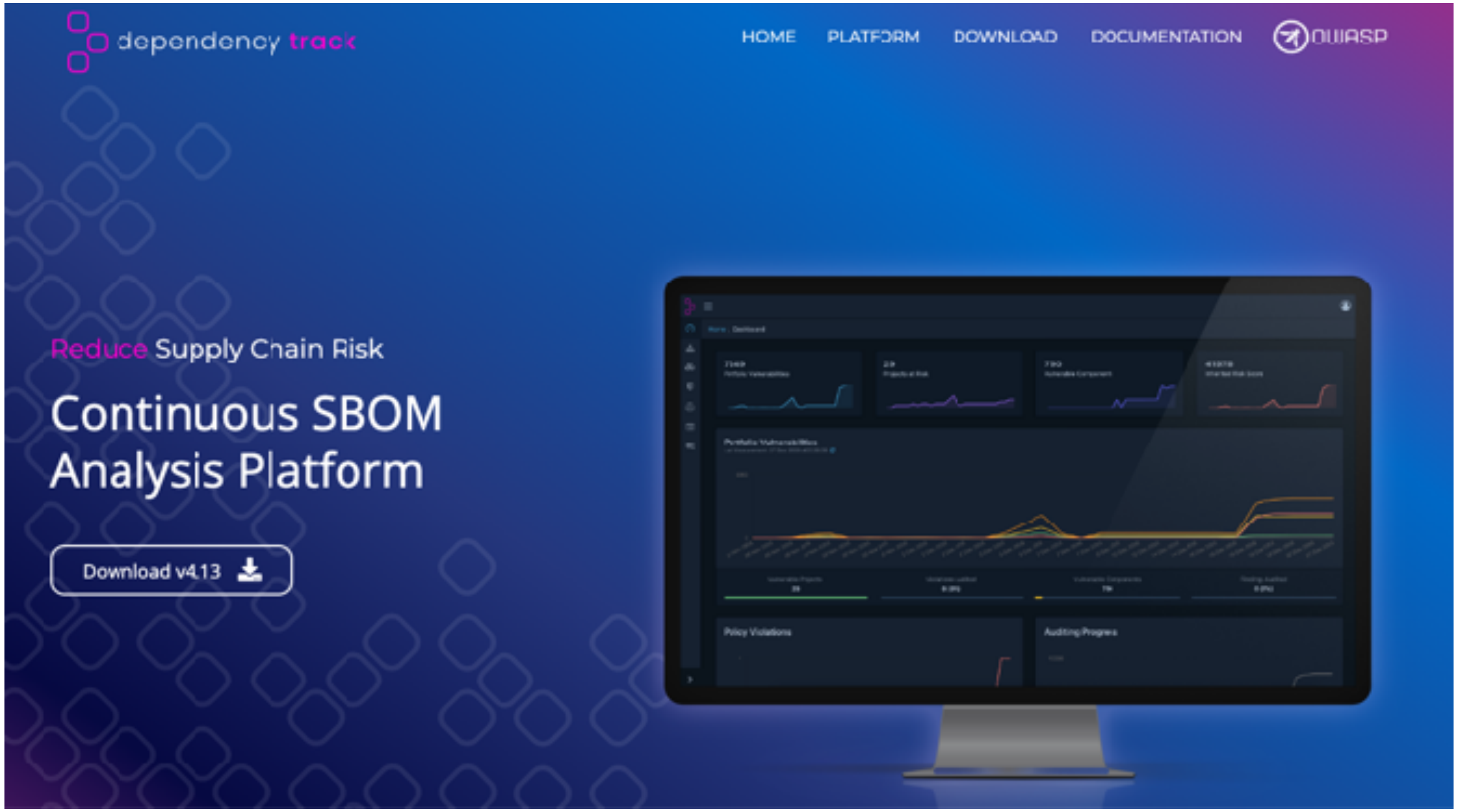
Network defense (WAF rules)

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

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

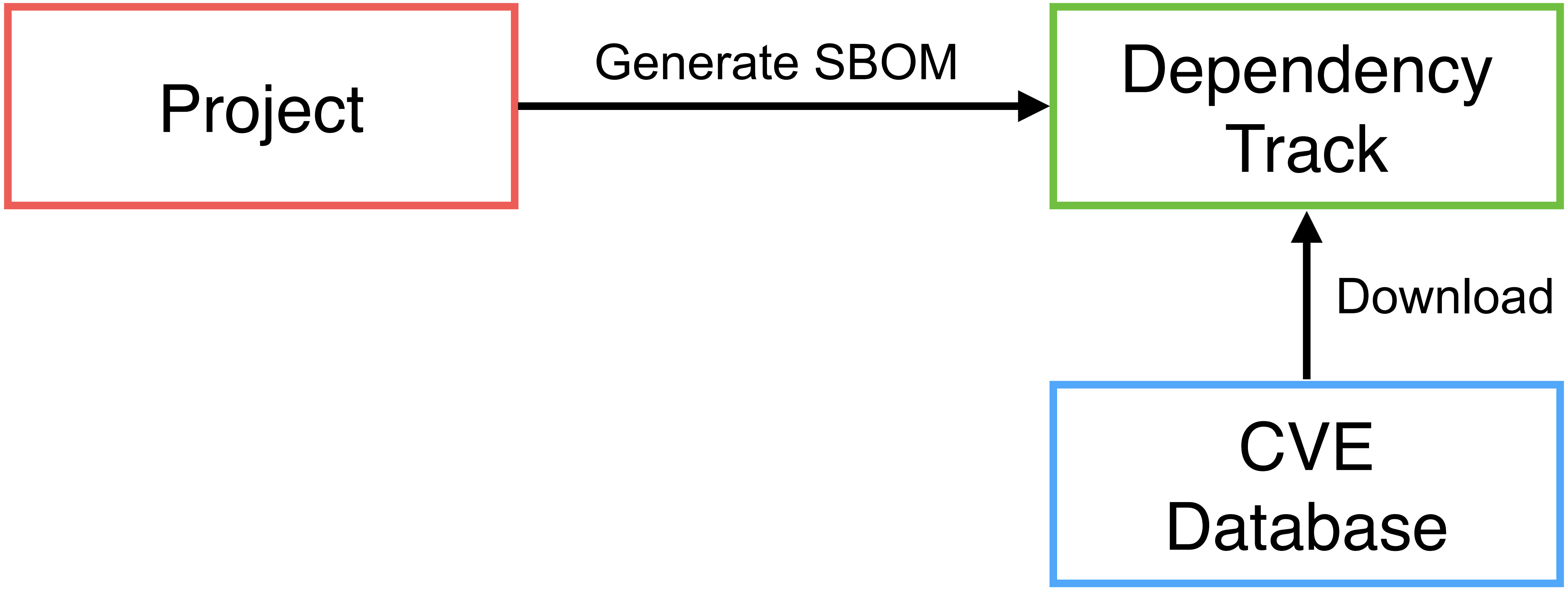
Dependency Track

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



https://dependencytrack.org/

Process ?



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

Q/A