

The new browser space



All the links: aka.ms/berlin0220



\$ whoami

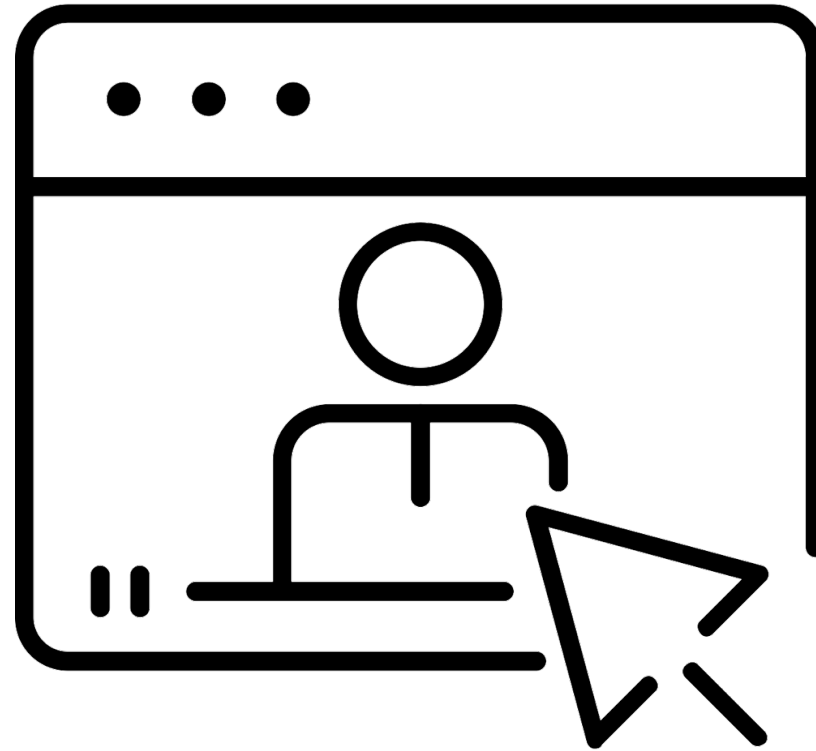


- Chris Heilmann @codepo8 – he/him/his
- Principal Program Manager Microsoft Developer Experience (VSCode, Devtools...)
- W3C Member
- 20 years web developer
- eToys, Agilisys, Yahoo, Mozilla...
- Wrote 3 books, contributed to 10(?)...
- Blogged since 2004, loud on Twitter

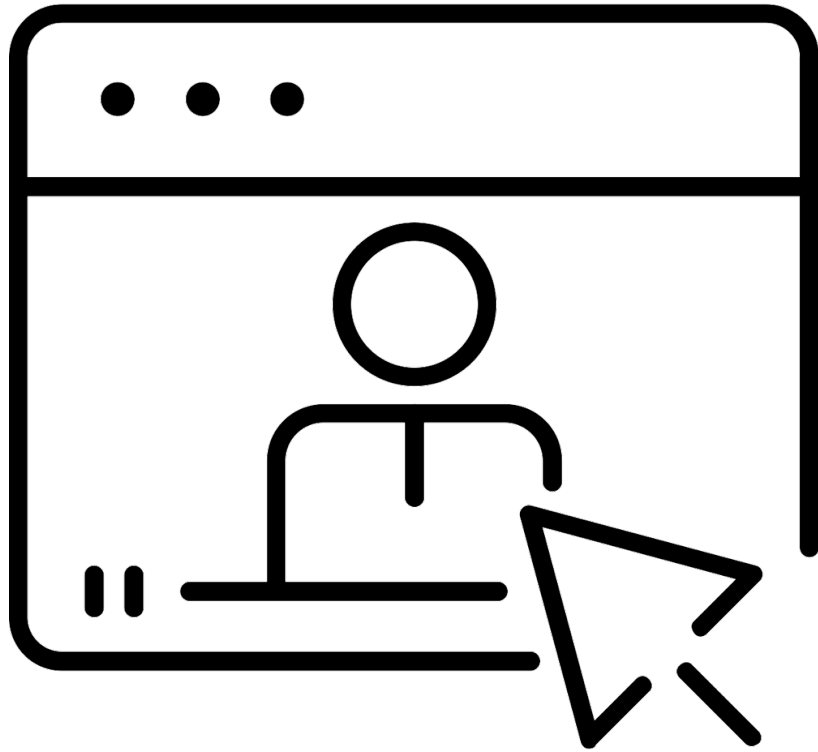
I've been working on a lot of things...



What is a browser?

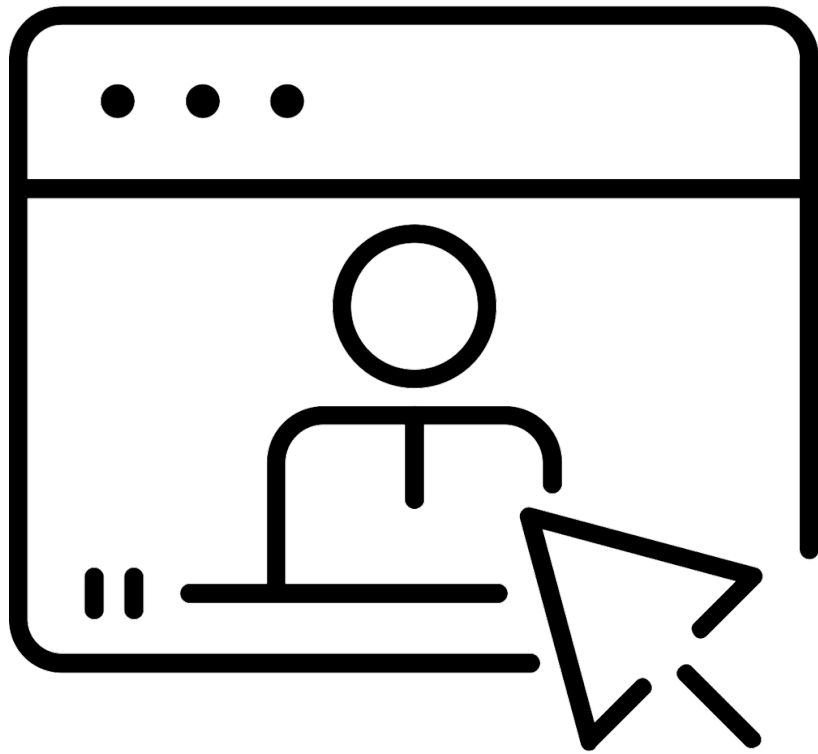


What is a browser?

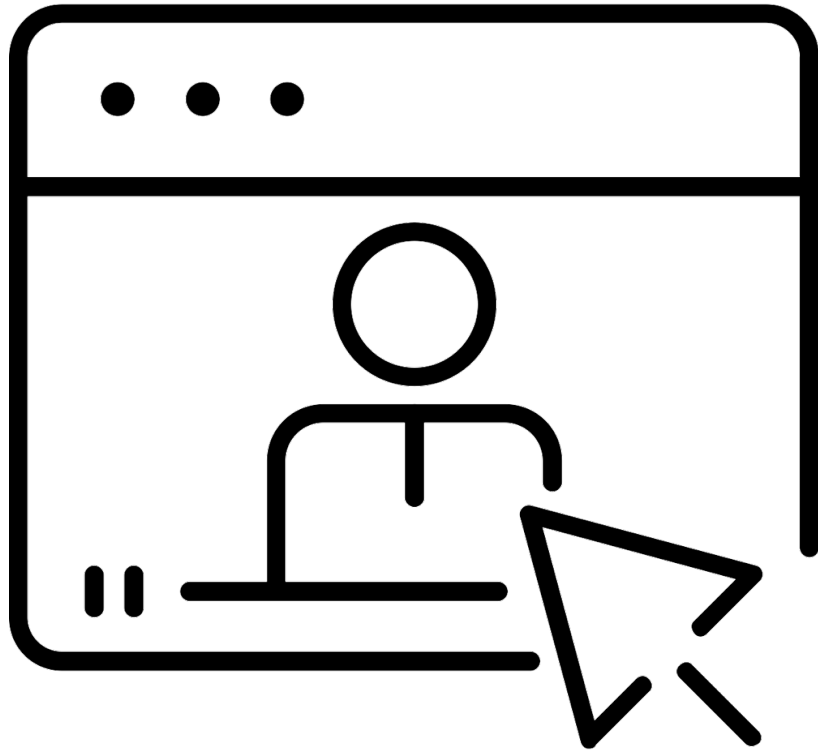


- Gateway to the internet
- Displays web content
- The last resort app to use on a phone?

Browsers for end users

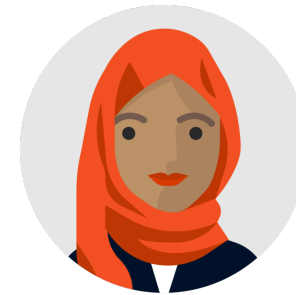
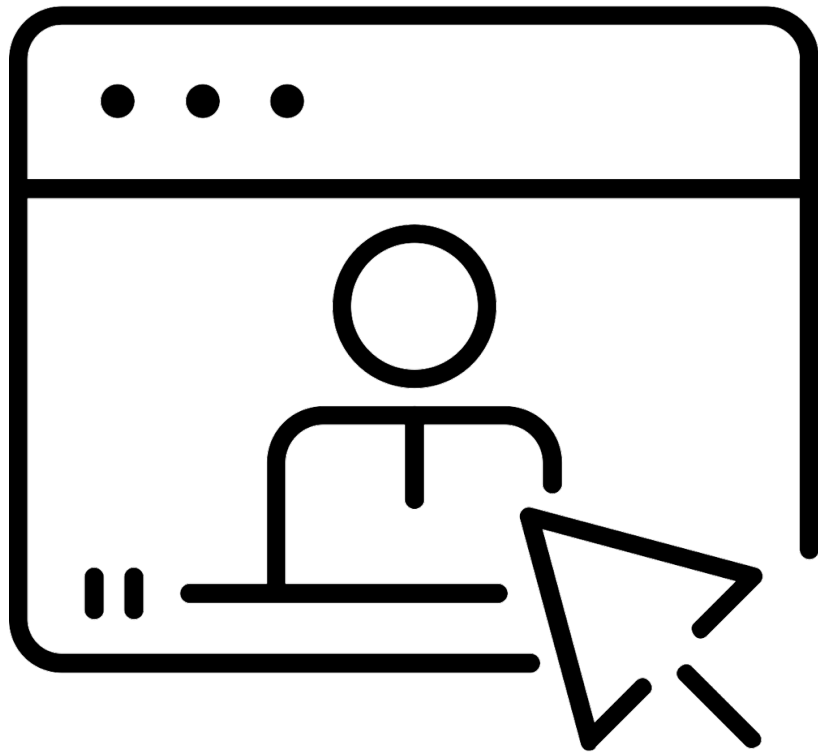


Browsers for end users

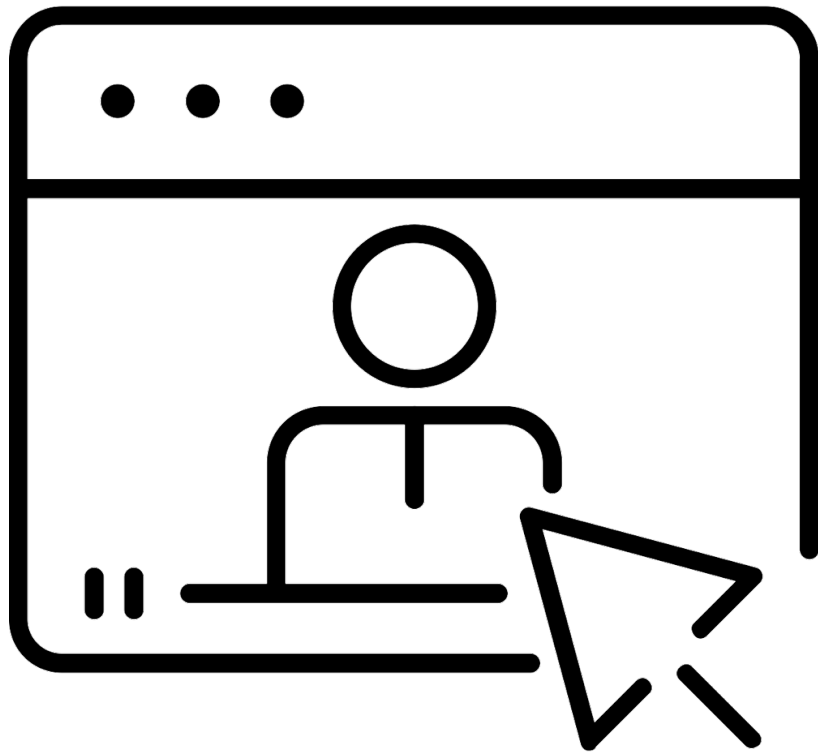


- Firewall for annoying/dangerous web content (Ad Blockers, Tracking Protection, Malware Filters)
- High performance media playback solution
- Record keeper (browsing history)
- Convenience method (autofill)
- Online identity keeper
- Conversion tool

Browsers for developers



Browsers for developers

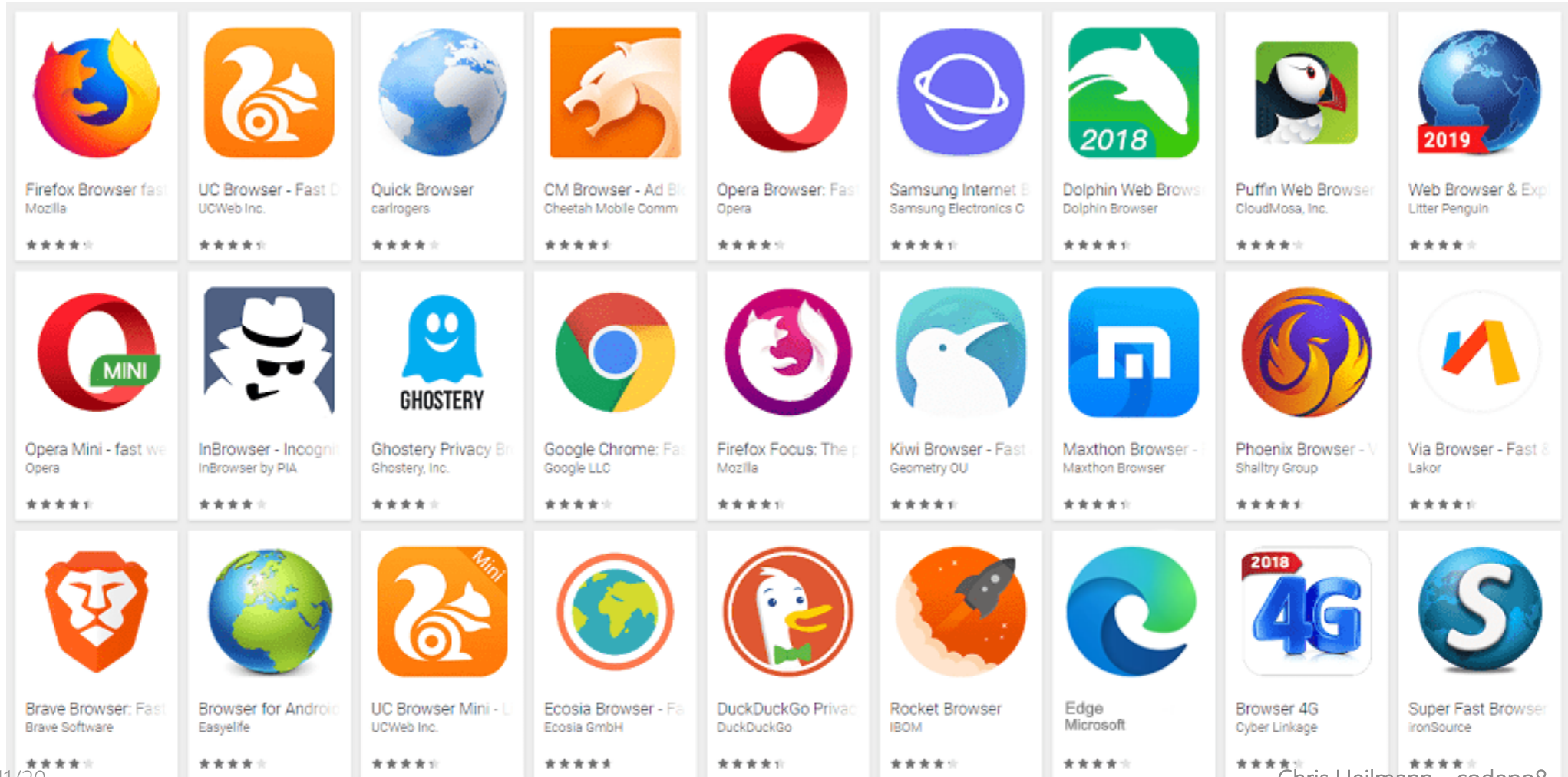


- Consumption and creation platform
- App platform (Progressive Web Apps, Electron...)
- Test platform (Automation with Puppeteer / Playwright)
- Preview of technology of the web to come (Beta, Canary, Developer builds, flags, origin trials)

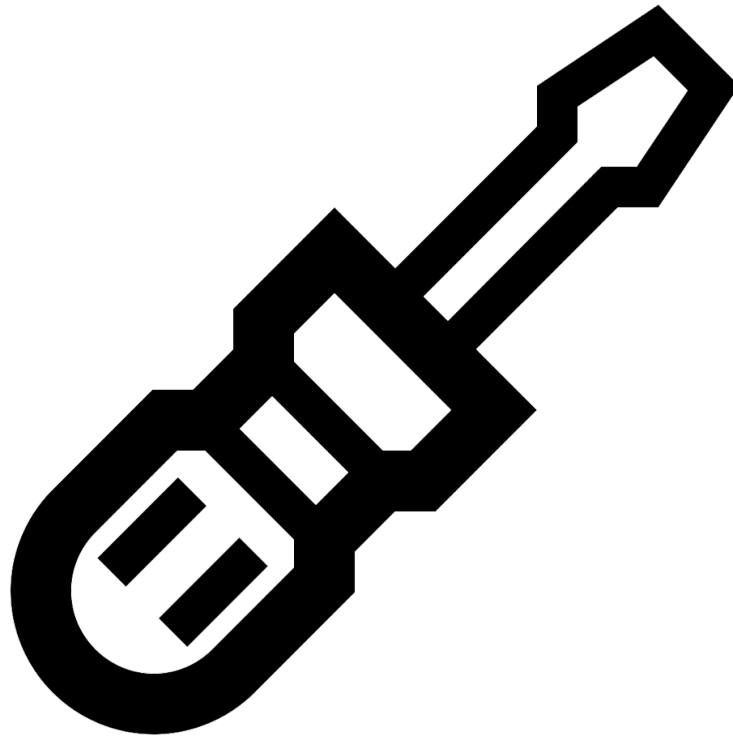
Spoilt for choice (Desktop)



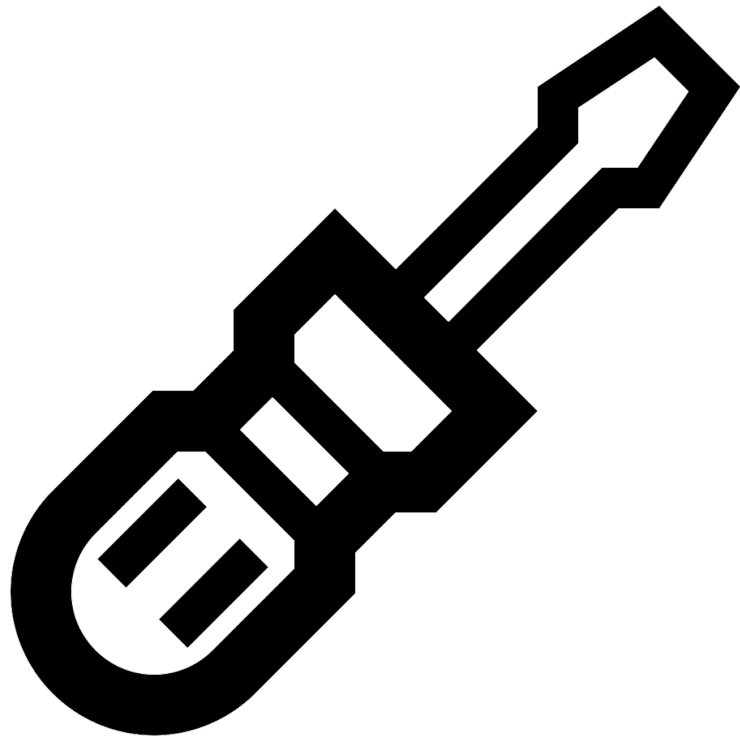
No way to predict (mobile)



What are web standards?

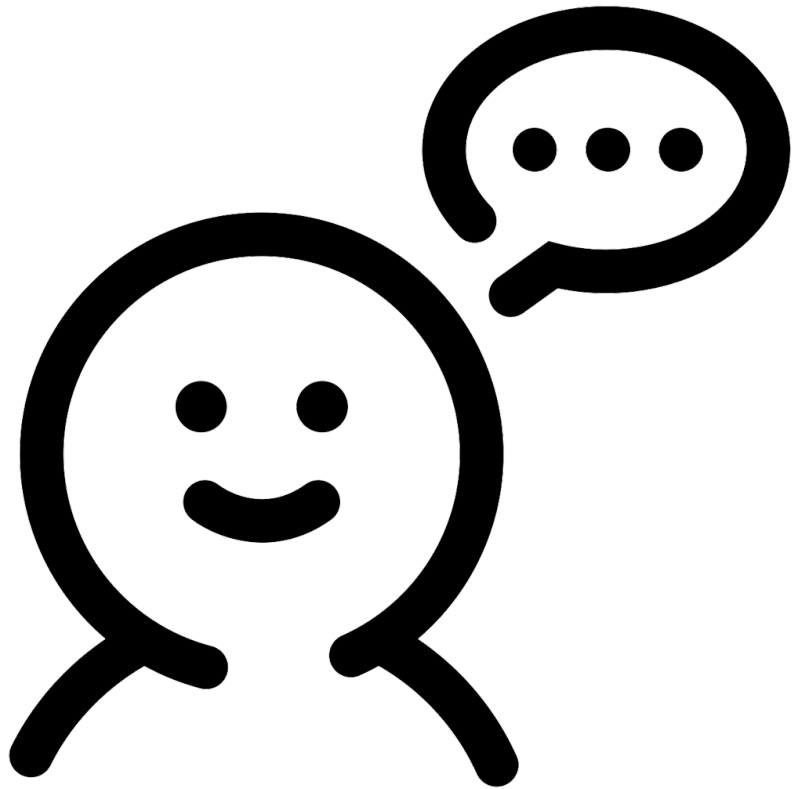


Why do we need web standards?



- Predictability of code outcome
- Prevention of monopolies
- Quality control – if your code validates the issue must be somewhere else
- Maintainability, backwards compatibility and future readiness.

Web standards are an agreement



- They used to be pretty academic
- Nowadays they are informed by use and needs of developers and users collected by browser makers and developers
- Discussions happen in the open and you are welcome to join



The web is a
roaring success...



- It is hard to imagine a world without the world wide web
- Web technologies are standardised and browser differences are negligible
- Open source has matured from a niche to the technology that runs the web
- Documentation is plentiful, free and interactive

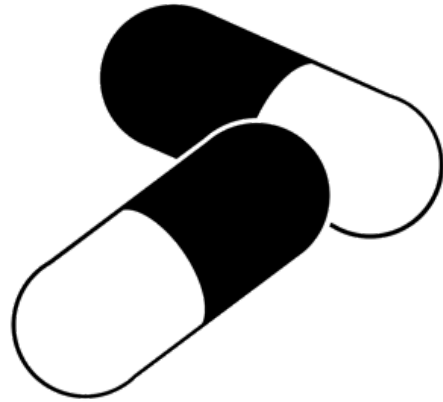


So, why is it all
kind of a mess?

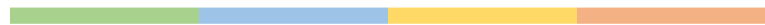
- Blatant lack of accessibility
- Security problems
- Performance issues
- A need for blockers to keep the web safe and a lot less annoying

Different users, different needs





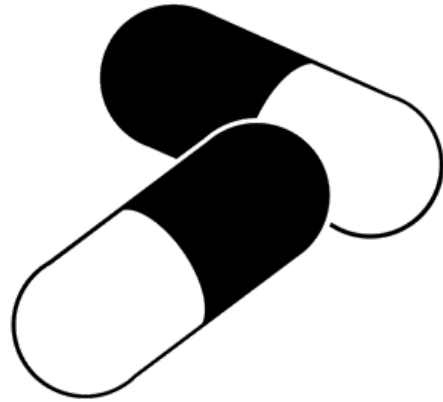
Somehow bitter
pills to swallow



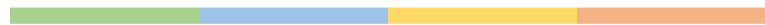
- To different users, a browser is
 - something they can't control
 - something they cherish and highly customize
 - irrelevant, as all of them work OK enough
 - nothing worth changing their habits for
- Security and privacy aren't things that people control, but trust in the platform to do for them

Different developers, different needs





Somehow bitter pills to swallow (2)



- The caring developer is a tiny fraction of the market
- For many, this is a job to make a living
- New developers don't find an empathic, open world, but an avalanche of choice and opinions
- Satisfying developer needs has no direct correlation to creating better user products



Why learn the basics?



- Browsers can't break the web – their forgiveness knows no bounds
- If it works, why bother learning why?
- If it doesn't work, there's ample of resources to learn quick fixes.



Abstraction attraction

- Learning “the web” isn’t a desirable skill
- Instead, abstractions get you eyeballs on job boards
- Abstractions promise faster turnaround and great products – big companies use them!



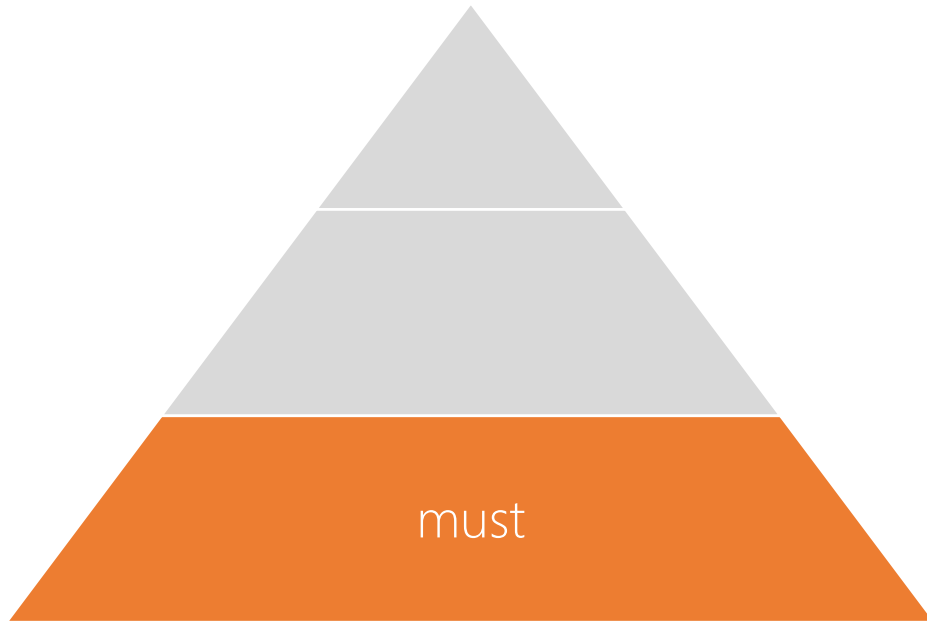
Tech innovation
isn't the answer



- Constant change in the developer space is a double-edged sword
- We empower a small group of highly motivated people at the cost of overwhelming others

Focusing on delivery

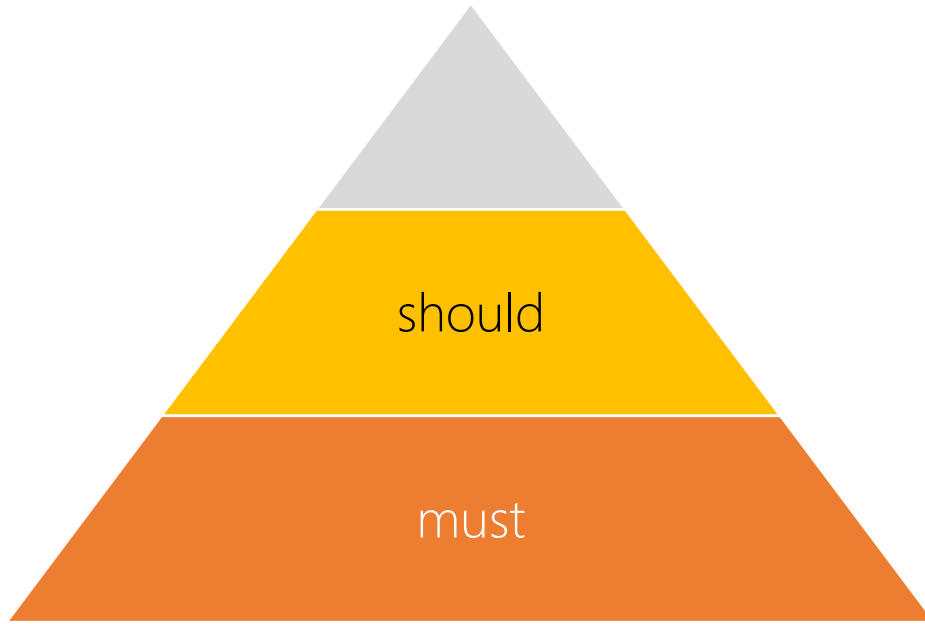




Filling the needs



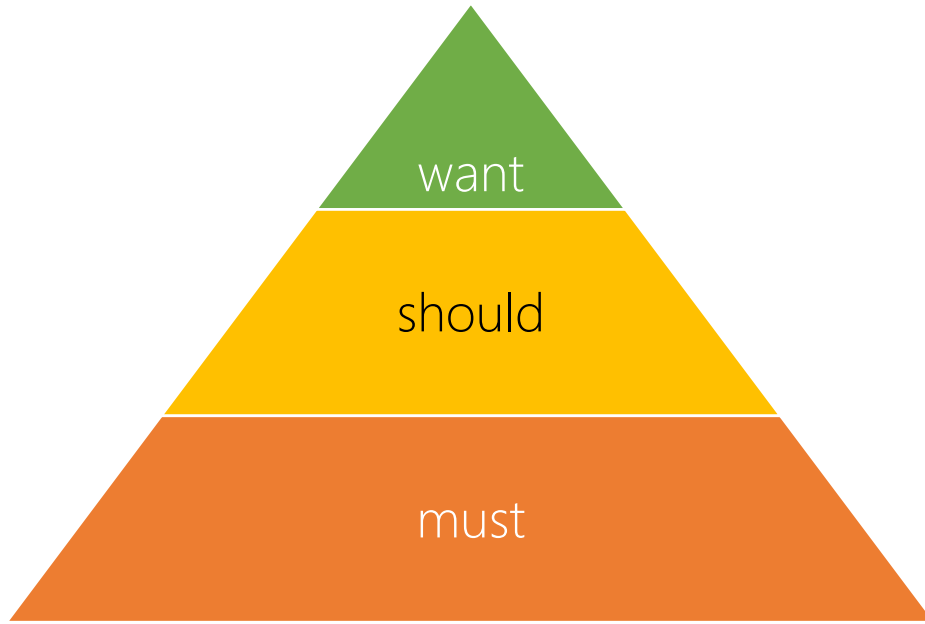
- Availability
- Accessibility
- Security
- Mutability to user needs



Adding quality



- Enhanced performance
- Extended availability (offline)
- Customisability



Creating delight



- Make full use of the platform
- Predict use cases and prevent simple mistakes
- Offer extended functionality



We <3 tools!



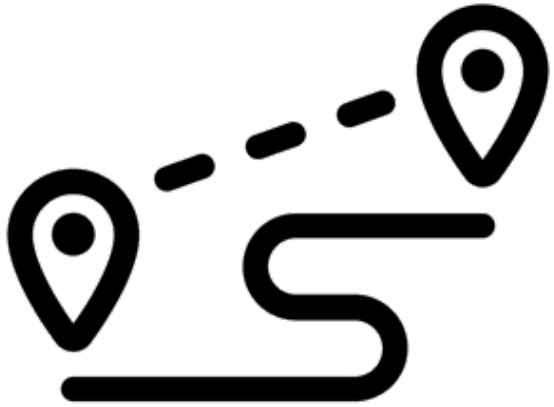
- Browser developer tools
- Editors, extensions and themes
- Terminals and their configuration
- Deployment and optimization toolchains
- Abstractions, Components and libraries



It is all a bit
much...



- We overwhelm people who want to start with the web – or just want to do their job
- By demanding people to set up complex dev environments we inhibit diverse contribution



Tools to help
simplify webdev





Node: the glue



- JavaScript on the server
- Enables a highly customisable web infrastructure (servers, build tools...)
- Elevated JavaScript from “how does browser \$x break” to a solution for all kind of web tasks



Chromium: the engine



- Open Source
- Multi Platform
- Powers various browsers and App engines (Electron)
- Can be automated and scripted (Puppeteer, Playwright)



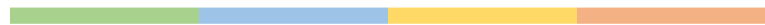
Visual Studio Code: the “hackable” editor



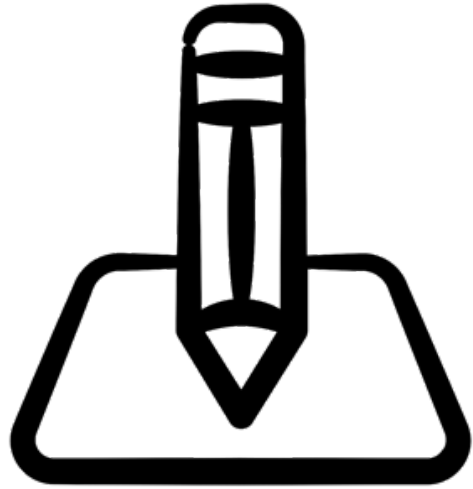
- Open Source
- Multi Platform
- Extensible / Customisable
- Based on Electron/Chromium
- Includes Git version control
- Has built-in terminal
- Lightweight, code focused



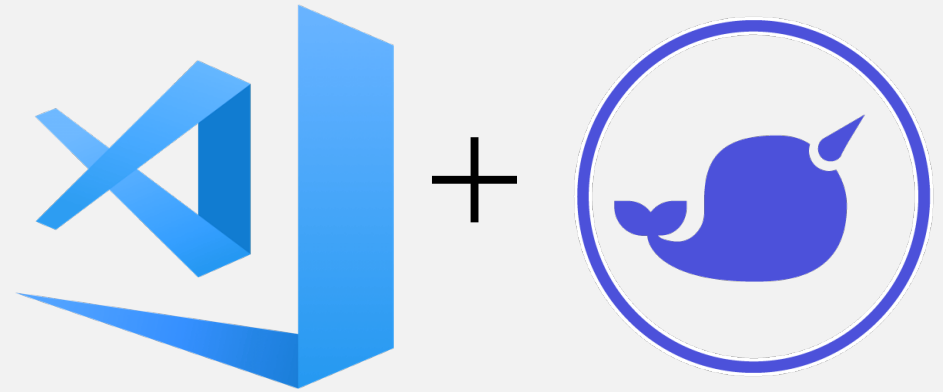
Webhint: testing and best practices



- Testing tool to check web projects for:
 - Accessibility issues
 - Performance problems
 - App readiness
 - Compatibility with standards
 - Common pitfalls to avoid
 - Security issues
- Online service (webhint.io) or node package to integrate in other tools
- Highly customisable

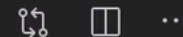


Show problems with
your code while you
write it...



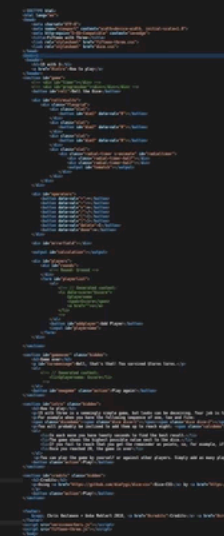
aka.ms/webhint4code

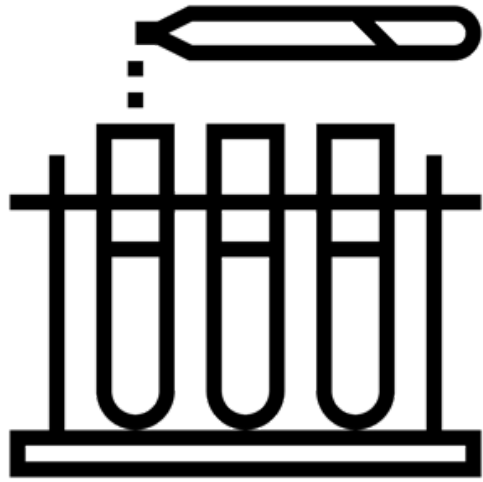
<> index.html ×



<> index.html > html > body

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>Fifteen with Three</title>
8      <link rel="stylesheet" href="fifteen-three.css">
9      <link rel="stylesheet" href="dice.css">
10 </head>
11 <body>
12   <header>
13     <h1>15 with 3</h1>
14     <a href="#intro">How to play</a>
15   </header>
16   <section id="game">
17     <!-- <div id="timer"></div> -->
18     <!-- <div id="progressbar"><div></div></div> -->
19     <button id="roll">Roll the Dice</button>
20
21     <div id="rollresults">
22       <div class="flexgrid">
23         <div class="slot">
24           <button id="die1" data-val="0"></button>
25         </div>
26         <div class="slot">
27           <button id="die2" data-val="0"></button>
```





Test your apps in
the browser...



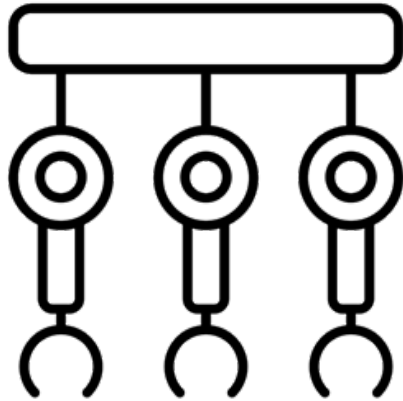
+



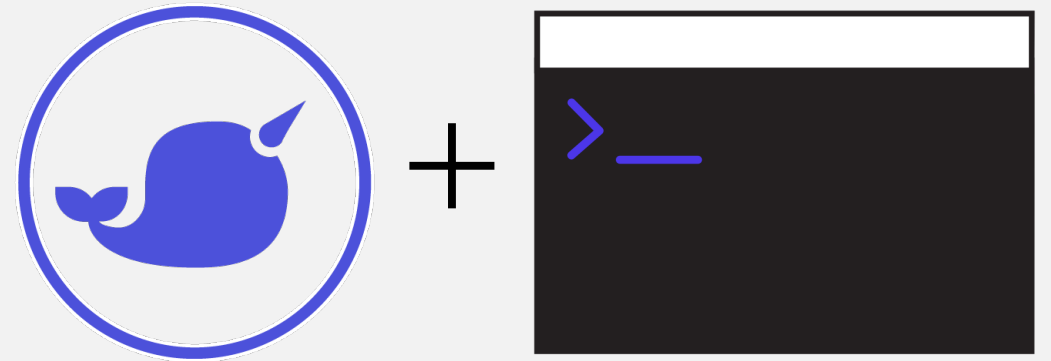
aka.ms/webhint4edge

aka.ms/webhint4firefox

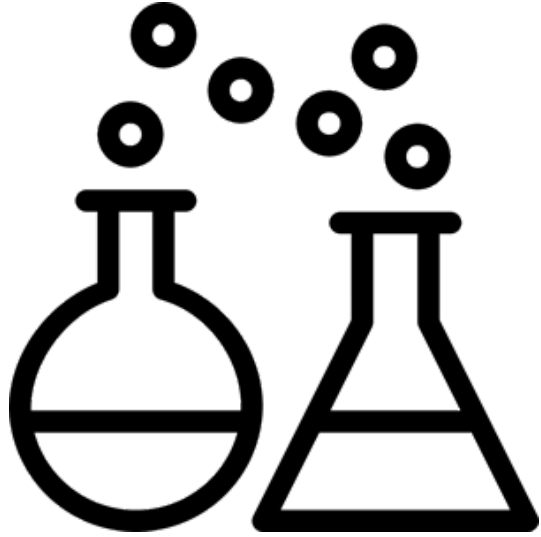
aka.ms/webhint4chrome



Add automated testing to your build process...



aka.ms/webhint4node



Elements for Code



- Edge/Chromium developer tools inside VS Code
- No need to switch to the browser to try a few things out
- No need to replicate the same functionality in both places

aka.ms/elements4code

EXPLORER

- OPEN EDITORS
 - index.html src
- WEBPACK-TYPESCRIPT-SASS-APP-BOILERP...
 - .vscode
 - launch.json
 - node_modules
 - src
 - assets
 - favicon.ico
 - index.html
 - TS index.ts
 - styles.scss
 - .gitignore
 - CHANGELOG.md
 - LICENSE
 - package-lock.json
 - package.json
 - README.md
 - JS serve.js
 - tsconfig.json
 - tslint.json
 - webpack.config.js
 - yarn.lock

```

<> index.html x
src > <> index.html > html > body
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Web App</title>
6 </head>
7 <body>
8   <div class="container">
9     <div class="banner container">
10      <h1>Elements for Microsoft Edge</h1>
11    </div>
12    <button id="addButton">Add a new item</button>
13    <div class="container" id="list"></div>
14  </div>
15 </body>
16 </html>
17

```

What's next?



https://aka.ms/browsers

☰ MENU



text/plain

ericlaw talks about the web and software in general

2/11/20



2020-02-09

Demystifying Browsers

I started building browser extensions more than 22 years ago, and I started building browsers directly just over 16 years ago. At this point, I think it's fair to say that I'm entering the *grizzled veteran* phase of my career.

With the Edge team continuing to grow with bright young minds from college and industry, I'm increasingly often asked "Where do I learn about browsers?" and I haven't had a ready answer for that question.

ABOUT ERICLAW

Impatient optimist. Dad. Author/speaker. Created Fiddler & SlickRun. PM @ MSFT '01-'12, and '18-, presently working on Microsoft Edge. My words are my own.

[View all posts](#)





What's missing on the web platform?



webwewant.fyi

In summary...



- The web is a diverse place
- We have a chance to make it easier for everyone to create quality products for all end users
- We have open source platforms with open communication channels that power the web
- We accumulated a lot of knowledge, why not add it to our tools to teach the creators of the web in context?

Thanks – your voice matters!



Chris Heilmann

@codepo8

@EdgeDevTools

christianheilmann.com

Click the feedback in Edge developer tools!

