

!important

Chris Heilmann

@codepo8
March 2019

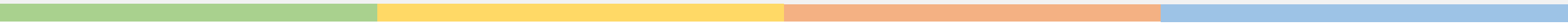


Chris Heilmann @codepo8



- Principal Software Development Engineer
- Microsoft Edge and Developer Experience (VSCode, Devtools...)
- W3C Member WebML Discussion Group
- 20 years web developer
- eToys, Agilisys, Yahoo, Mozilla...
- Wrote 3 books, contributed to 10(?)...

!important

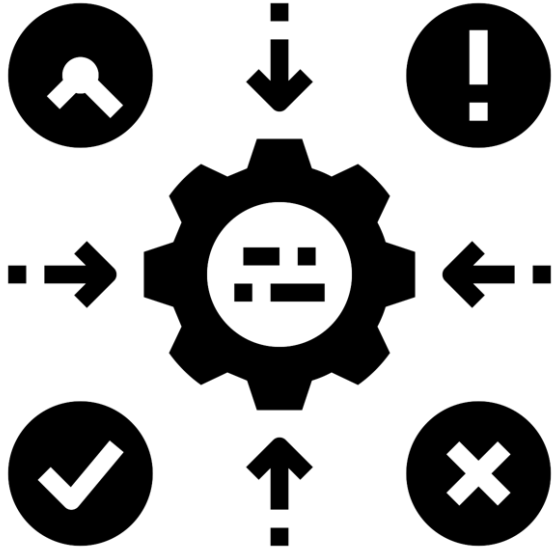




!important



- For a CSS developer: this is so important, it should override all other inherited settings
- For a CSS developer finding it in production: someone was here who shouldn't have been
- For a JavaScript developer writing CSS: that's how I get it to do what I want
- For a JavaScript developer not knowing CSS: not important



Best practices need
insight and context



- It is not HTML vs. CSS vs. JavaScript
- Each on their own are probably not enough to create a great product
- What was enough in the beginning of the web is probably not enough now that expectations changed
- Just because we can do everything with JavaScript, it may not be the most effective and performant way



We all crave
importance and
recognition

- Being visible is much easier these days than it used to be
- A lot more code is open and available in the public
- We could collaborate and build great things together
- But it is much easier to criticize the work of others, isn't it?

Drop code, receive
assumptions and
well-intentioned
advice...



Pretty wild how you can avoid loops these days in JavaScript

"create a dropdown from the values of an object and remove all duplicates"

```
    'e44': 'serif',  
    'e45': 'ligature',  
    'e46': 'beak',  
    'e47': 'spine',  
    'e48': 'head serif',  
    'e49': 'head serif'  
  }  
  const createdrop = () => {  
    let droptions = new Set(  
      Object.values(information)  
    );  
    let str = `  
      <option>${[...droptions].sort().join('</option><option>')}</option>  
      </select>`;  
    return str;  
  }
```



Probably not safe to
assume clean data...

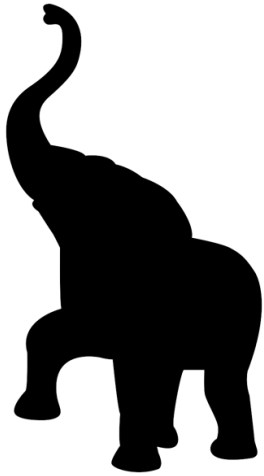


Loïc Giraudel @l_giraudel · Feb 11

Replying to @codepo8

I would rely on map to prevent an empty `<option></option>` when information object is empty.

```
const information = {
  'e44': 'serif',
  'e45': 'ligature',
  'e46': 'beak'
}
const createdrop = () => {
  const dropValues = new Set(Object.values(information))
  const sortedOptions = [...dropValues].sort().map(v => `<option>${v}</option>`)
  return `<select>${sortedOptions.join('')}</select>`
}
```

Not safe to assume
performance with
lots of data...



James Socol @jamessocol · Feb 11

Replying to @codepo8

Avoid *writing loops. There are a bunch of (5ish?) list iterations in there, and depending on the workload may need more optimizing



Happy accidents can happen...



Michael Bruyninckx @modul8com · Feb 11
Replying to @codepo8
how does this filter out duplicates ? this is only sorting, right ?

Jim Liu @jimcalliu · Feb 11
Using Set allow you to store _only_ unique values. Learn more here 🧠💡

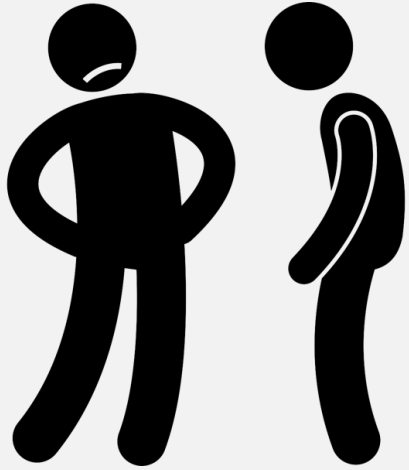
Set
The Set object lets you store unique values of any type, whether primitive values or object references.
developer.mozilla.org

Brian Wynn @bwynn_dev · Feb 11
+1
Set is your friend, add it to your tool-belt!

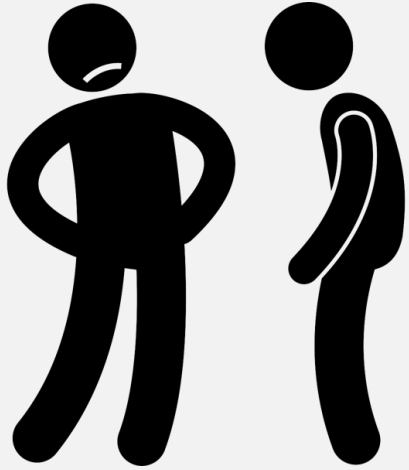
Michael Bruyninckx @modul8com · Feb 11
cool, didn't know that one yet. Thanks !

Things we say to
each other without
considering the
context...

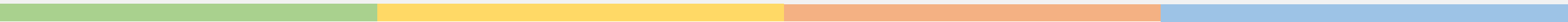


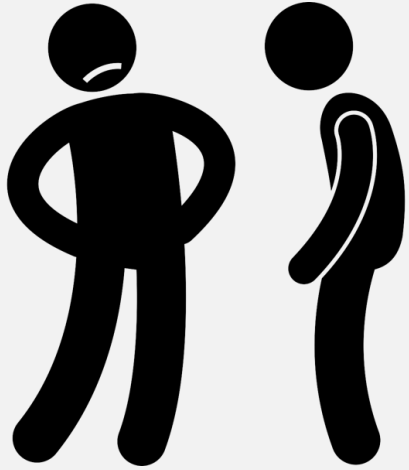


THIS DOES NOT WORK
WHEN THE DATA YOU PUT
IN IS IMPURE!

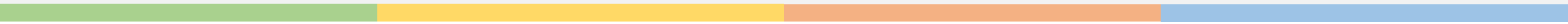


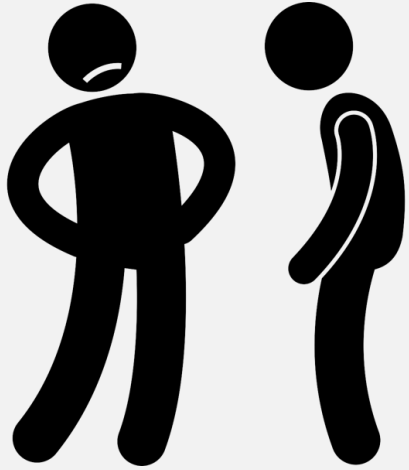
THIS DOESN'T PERFORM
WHEN YOU HAVE A LOT
OF DATA!



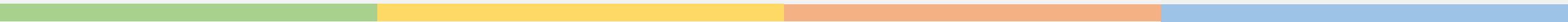


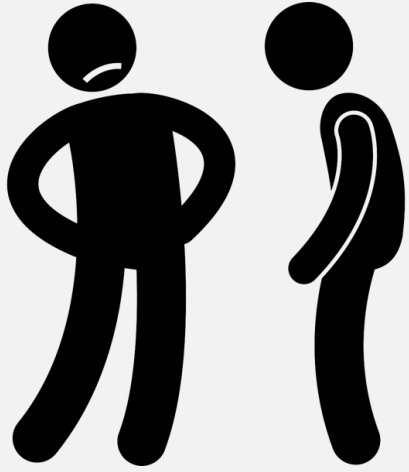
THiS coULD BE wrITten
mUch SHORTER!



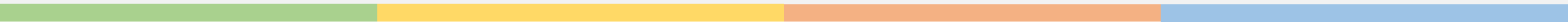


THIS dOeS NOt WORK iN
oLd BrOwSErS!





THIS IS nOT ACCeSSIBLe!



Derpify

Enter text, receive derp...

codepo8.github.io/not-that-important/derpify-js.html



Chris Heilmann

@codepo8

```
function derpify(str) {
  return str.toLowerCase().split('').map(
    c => (Math.random() < 0.3) ? c :
    c.toUpperCase()
  ).join('');
}
derpify('hello there')
"HELLO TheRE"
```



Jonas Raoni Soares da Silva @jonasraoni · Feb 14

Replying to @codepo8

```
Less verbose: s.replace(/./g, c => Math.random() < .3 ? c.toLowerCase() :
c.toUpperCase()) =]
```



Chris Heilmann

@codepo8

mOaR rEAdAble

```
function derpify(str) {
  return str.toLowerCase().split('').map(
    c => (Math.random() < 0.3) ? c : c.toUpperCase()
  ).join('');
};
```



Comfortable
rebellion...





Each of these is a
sure-fire win...

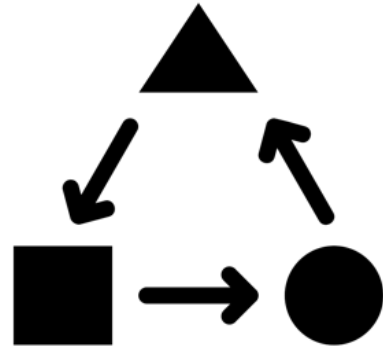


- If you zoom out far enough, you can make accusations that can't be argued against
- This means you look clever and caring for the web
- It does, however, also mean that we discourage people
- And is it really the right place to argue for quality? When the product is done?

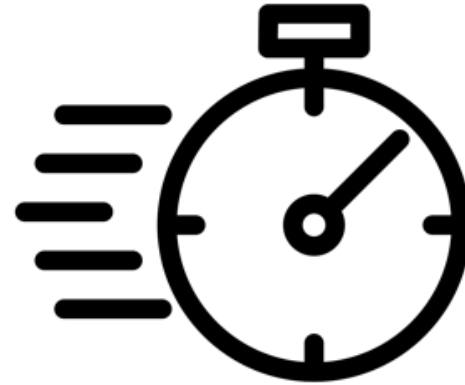
As quality focused developers, we rightfully care about...



Accessibility



Interoperability

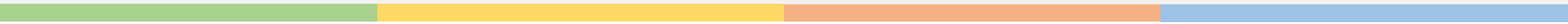


Performance



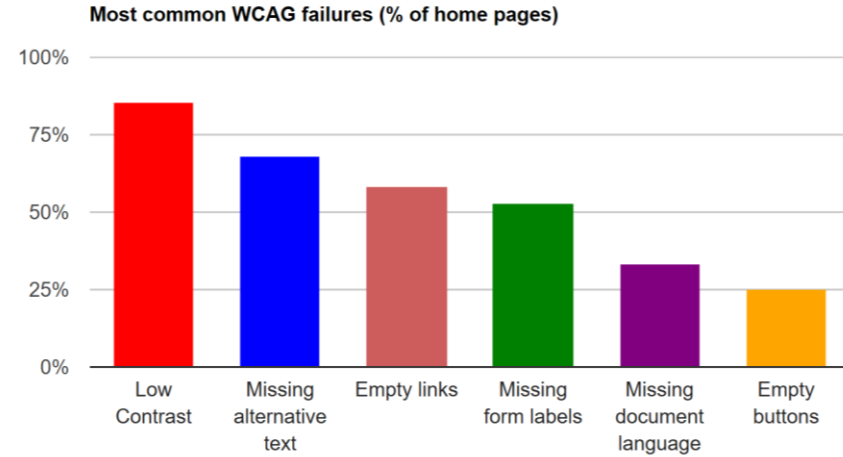
Security

Alas, the state of the web is
!good





Rampant accessibility failures...

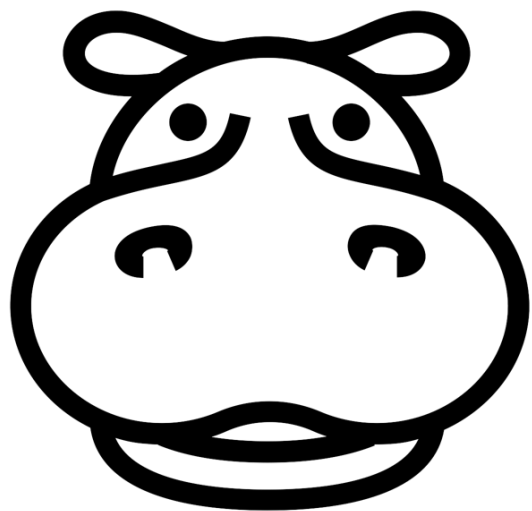


Most common types of WCAG 2 failures

WCAG Failure Type	# of home pages	% of home pages
Low contrast text	852,868	85.3%
Missing alternative text for images	679,964	68%
Empty links	581,408	58.1%
Missing form input labels	528,482	52.8%
Missing document language	329,612	33.1%
Empty buttons	250,367	25%

<https://majestic.com/reports/majestic-million>

<https://webaim.org/projects/million/>



Huge and not optimised...



Katie Hempenius

@katiehempenius

Following



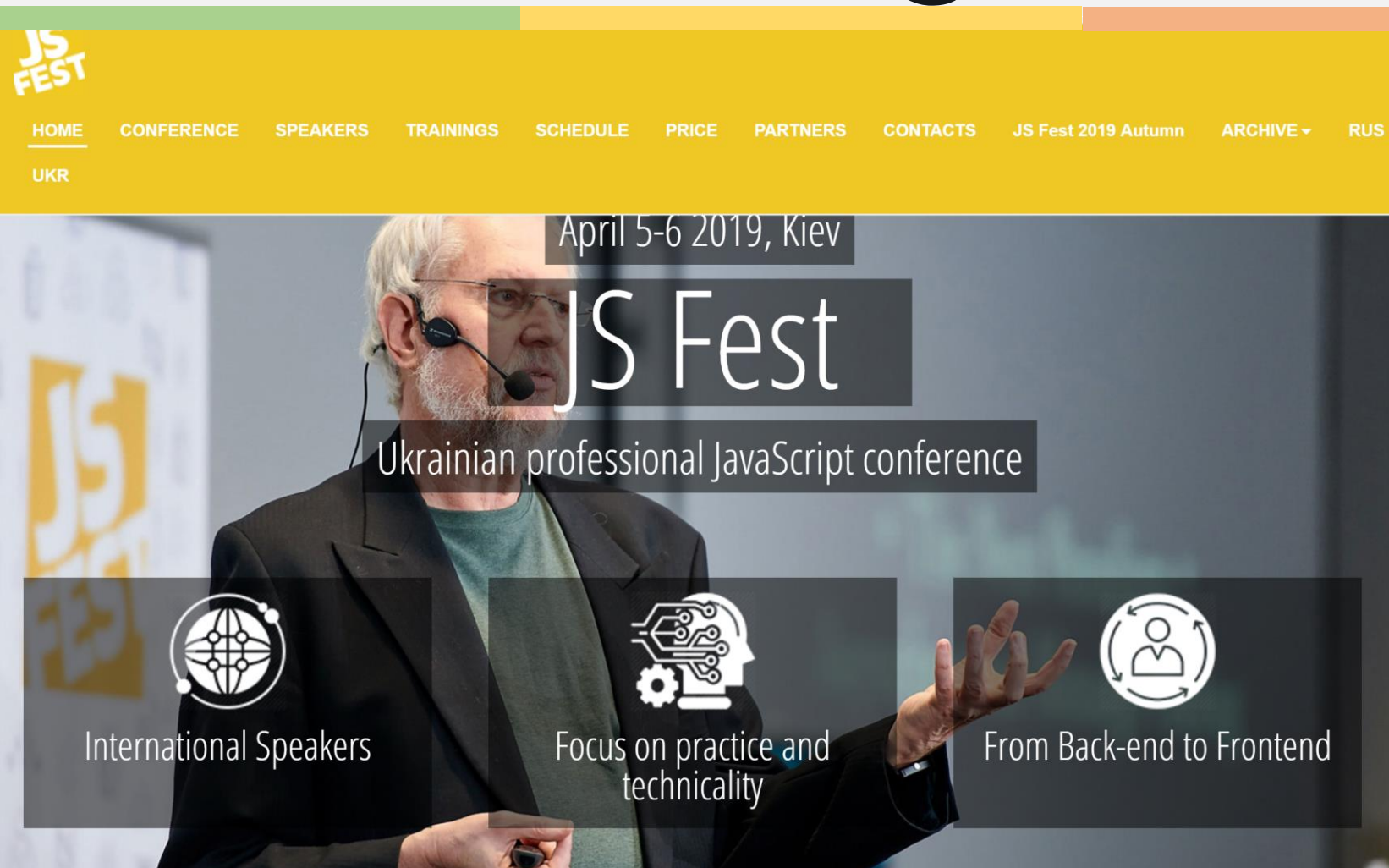
Page Weight Percentiles: overall & by resource type

If one of these categories is super high for a site, it's often an indicator that there's "low hanging fruit" performance wins to be had by optimizing that resource type.

(Data source: [@HTTPArchive](#), Desktop Oct'18 data)

	Overall	Images	JS	Font	CSS	HTML
p90	5.8 MB	4 MB	1.2 MB	290 KB	200 KB	130 KB
p75	3 MB	1.8 MB	780 KB	170 KB	100 KB	60 KB
p50	1.5 MB	650 KB	420 KB	100 KB	50 KB	30 KB
p25	700 KB	180 KB	200 KB	50 KB	20 KB	10 KB
p10	280 KB	40 KB	90 KB	30 KB	10 KB	10 KB

How is that possible with all the insights we get?



JS FEST

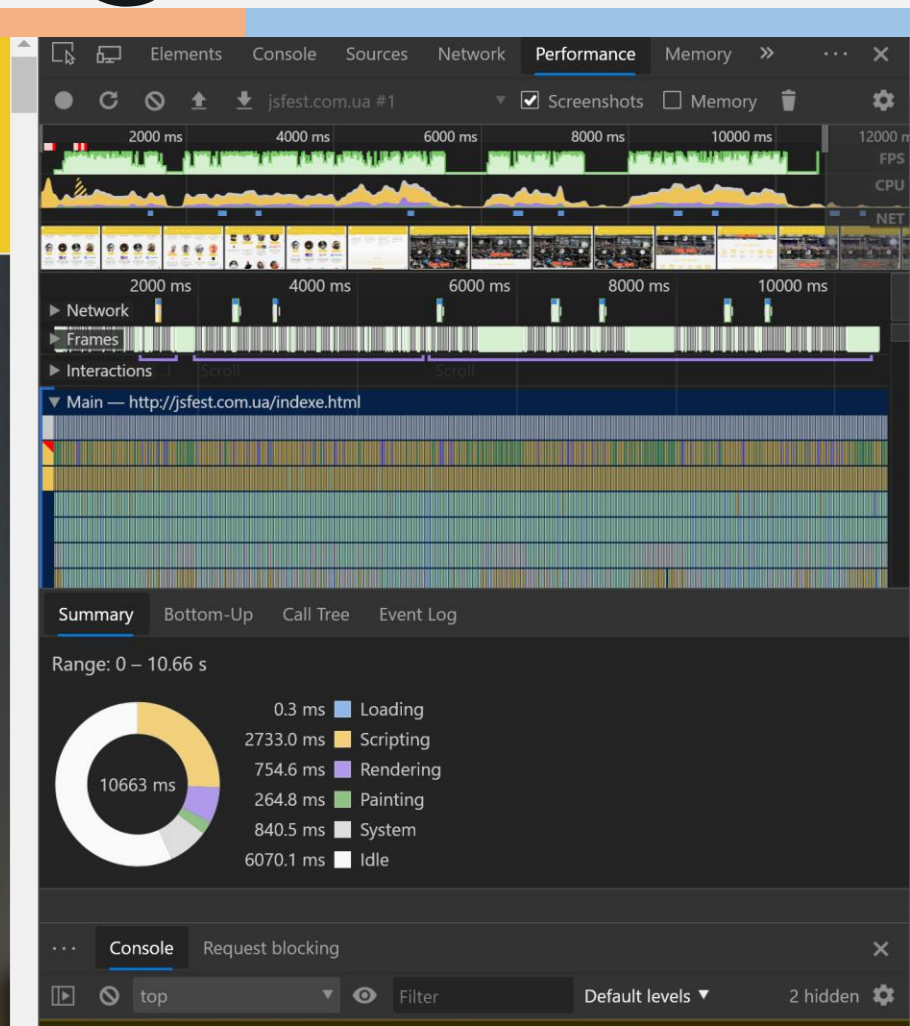
HOME CONFERENCE SPEAKERS TRAININGS SCHEDULE PRICE PARTNERS CONTACTS JS Fest 2019 Autumn ARCHIVE RUS UKR

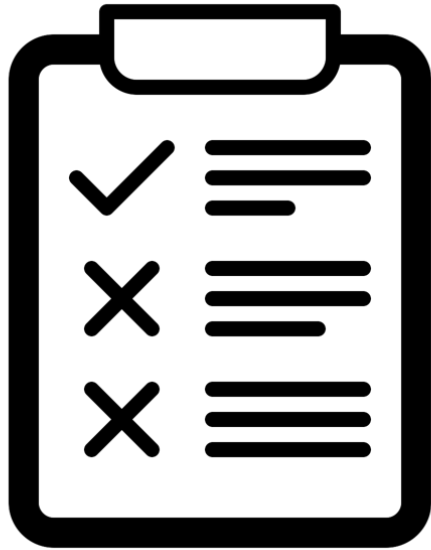
April 5-6 2019, Kiev

JS Fest

Ukrainian professional JavaScript conference

- International Speakers
- Focus on practice and technicality
- From Back-end to Frontend





We have audits in
browser tools...



Audits


Identify and fix common problems that affect your site's performance, accessibility, and user experience. [Learn more](#)

 Device

- Mobile
- Desktop

Audits

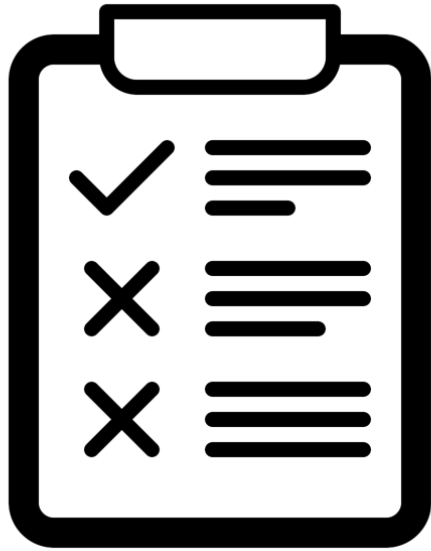
- Performance
- Progressive Web App
- Best practices
- Accessibility
- SEO

 Throttling

- Simulated Fast 3G, 4x CPU Slowdown
- Applied Fast 3G, 4x CPU Slowdown
- No throttling

Clear storage

Run audits



We have audits in browser tools...



Elements Console Sources Network Performance Audits

10:45:20 PM - fwdays.com

84 Performance 63 Accessibility 79 Best Practices 100 SEO PWA Progressive Web App

Score scale: 90-100 50-89 0-49

Performance

84

Metrics

First Contentful Paint	1.5 s	✓	First Meaningful Paint	1.8 s	✓
Speed Index	2.2 s	✓	First CPU Idle	5.6 s	!
Time to Interactive	5.7 s	!	Estimated Input Latency	30 ms	✓

[View Trace](#) Values are estimated and may vary.

Opportunities

These optimizations can speed up your page load.

	Opportunity	Estimated Savings
1	Serve images in next-gen formats	1.95 s
2	Properly size images	1.5 s
3	Eliminate render-blocking resources	0.75 s
4	Preconnect to required origins	0.4 s
5	Defer unused CSS	0.15 s



Accessibility Insights



Accessibility Insights for Web



[Watch 3-minute video introduction](#)

Launch pad



FastPass

Run two tests to find the most common accessibility issues in less than 5 minutes.



Assessment

Walk through a guided process for assessing accessibility compliance.



Ad hoc tools

Get quick access to visualizations that help you identify accessibility issues.

<https://accessibilityinsights.io/>

- Overview
0% Completed
- 1 Automated checks
- 2 Keyboard
- 3 Focus
- 4 Landmarks
- 5 Headings
- 6 Repetitive content
- 7 Links
- 8 Native widgets
- 9 Custom widgets
- 10 Timed events
- 11 Errors
- 12 Page navigation
- 13 Parsing
- 14 Images
- 15 Language
- 16 Sensory
- 17 Text legibility
- 18 Audio / video

OVERVIEW

This page contains a summary that indicates the progress of your assessment. An assessment is a manual experience in which you navigate through a set of tests that cover all WCAG 2.0 AA success criteria. Each test has one or more requirements that can be:

- Automated
- Assisted
- Manual

Where possible, Accessibility Insights for Web "assists" the test process by generating a list of instances to evaluate and highlighting them on the screen. Accessibility Insights for Web also allows you to manually record failure instances.

Summary



Test details

Automated checks	<input checked="" type="checkbox"/> 48	Page navigation	○○○
Keyboard	○○○○○	Parsing	○
Focus	○○○○○	Images	○○○○
Landmarks	○○○	Language	○○○
Headings	○○○	Sensory	○○○○
Repetitive content	○○○	Text legibility	○○○
Links	○○	Audio / video	○○
Native widgets	○○○○	Multimedia	○○○○○
Custom widgets	○○○○○○	Live multimedia	○

Help

- [Getting started](#)
- [How to complete a test](#)
- [Ask a question](#)

Target page: [Fwdays. IT tech conferences in Ukraine, Kyiv](#)

1 Automated checks

2 Tab stops

Automated checks

Show failures On [↗ Export result](#)

✖ 20 failures were detected. When the Automated Checks toggle is on, failure instances selected in this page are highlighted in the target page.

<input checked="" type="checkbox"/>	>	Path	Snippet	How to fix
<input checked="" type="checkbox"/>	>	color-contrast: Elements must have sufficient color contrast ✖ 9 WCAG 1.4.3		
<input checked="" type="checkbox"/>	>	frame-title: Frames must have title attribute ✖ 1 WCAG 2.4.1		
<input checked="" type="checkbox"/>	>	link-name: Links must have discernible text ✖ 7 WCAG 2.4.4 , WCAG 4.1.2		
<input checked="" type="checkbox"/>	>	meta-viewport: Zooming and scaling must not be disabled ✖ 1 WCAG 1.4.4		
<input checked="" type="checkbox"/>	>	unique-landmark: Landmarks must have a unique role or role/label combination (aria-label OR aria-labelledby) ✖ 2 WCAG 2.4.1		

Failure details

Select a single failure instance from a group in the table above to see more details here.

SCANNING 100%

SCAN TIME: 03:00

HINTS

URL: <https://fwdays.com/en/>

DATE: 2019-03-13 22:46

409

YOUR SCAN RESULT LINK: <https://webhint.io/scan/186276>

webhint version: 4.4.1 Configuration JSON

Security

- disown-opener: 13 hints
- no-protocol-relative-urls: 3 hints
- sri: 22 hints
- strict-transport-security: 51 hints
- validate-set-cookie-header: 7 hints
- x-content-type-options: 41 hints
- ssllabs: 1 hints



webhint.io

	ACCESSIBILITY	HINTS 1	PASSED 0/1
	COMPATIBILITY	HINTS 53	PASSED 4/7
	PWA	HINTS 1	PASSED 3/4
	PERFORMANCE	HINTS 216	PASSED 1/7
	PITFALLS	HINTS 0	PASSED 0/0
	SECURITY	HINTS 138	PASSED 3/10

Well, all that is...
!important





Learn once, repeat
forever...



 **Sebastian McKenzie** 
@sebmck Following 

This has been my debugging strategy for the past 10 years and I don't see it changing

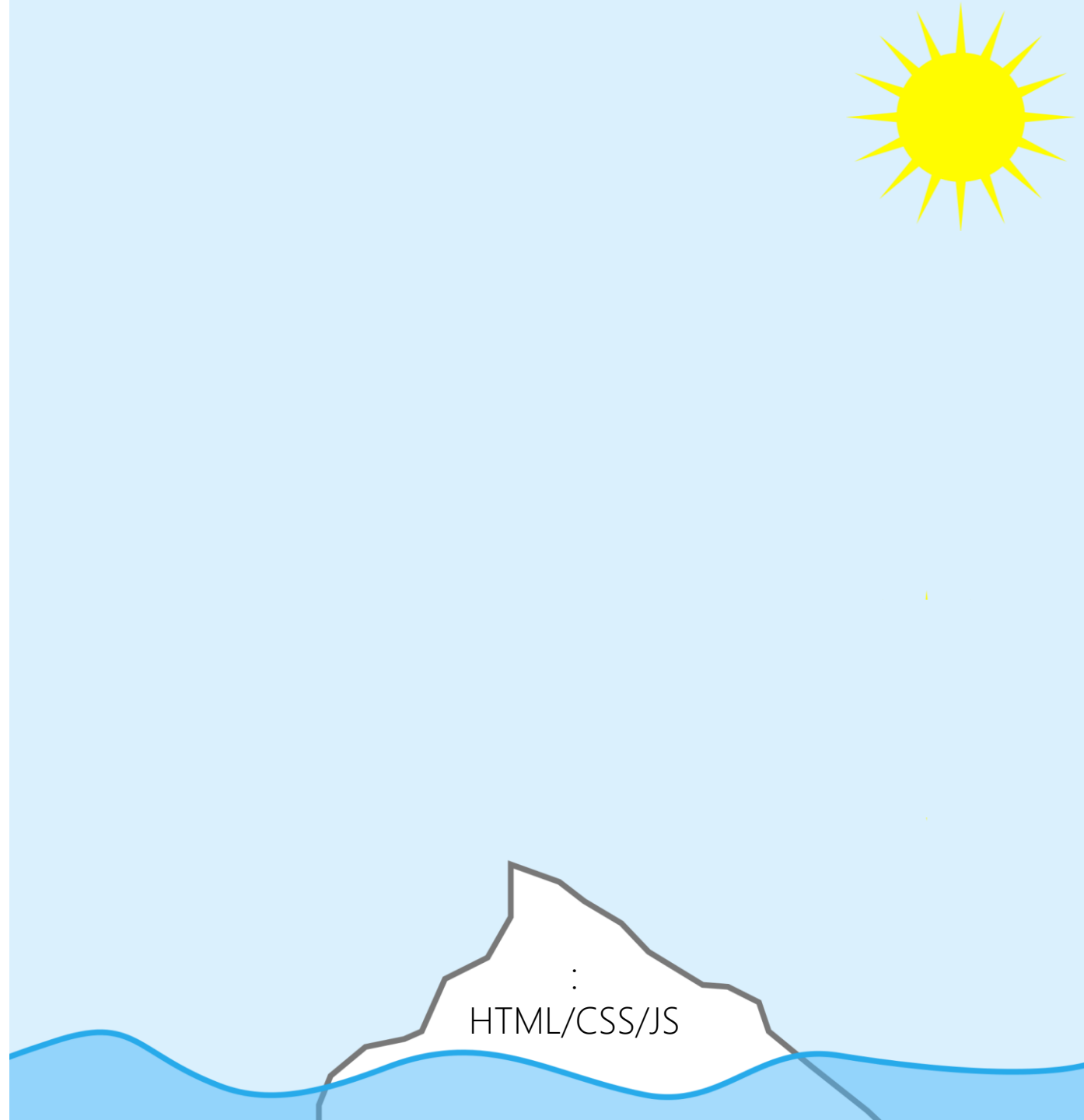
```
console.log('hello');  
const res: AnalyzeDeper  
  filename,  
);  
console.log('bye');
```

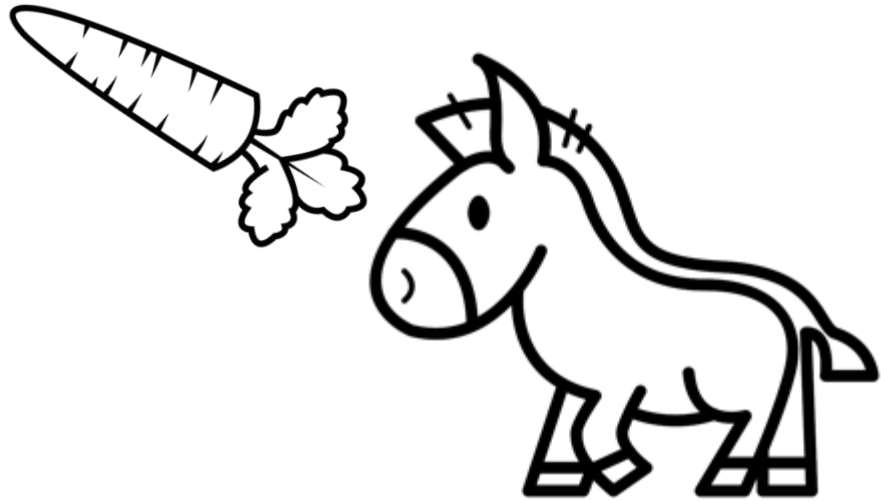
2:34 AM - 24 Mar 2019

252 Retweets 1,540 Likes 

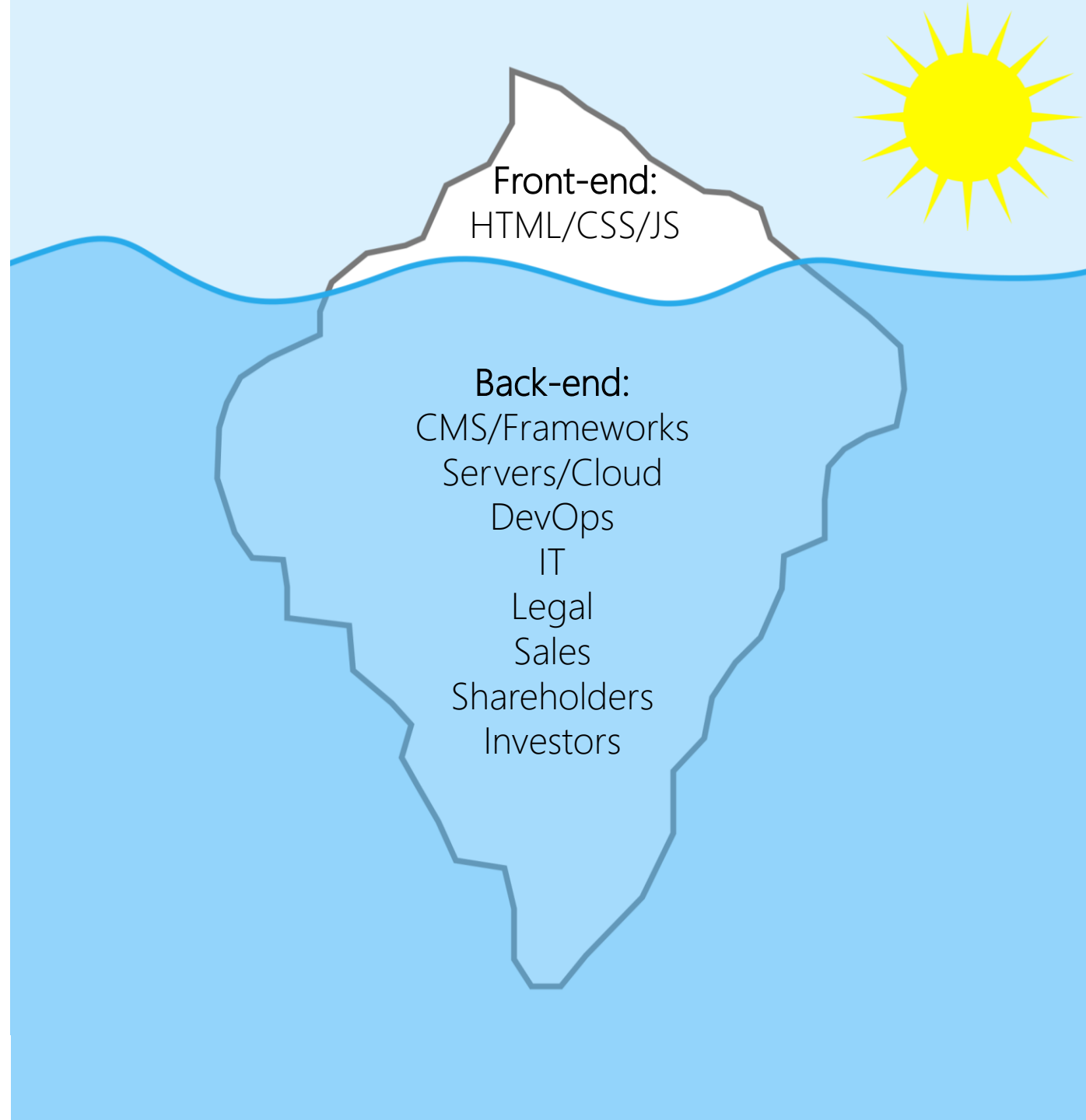


Web products stack





Realities of web products stack...



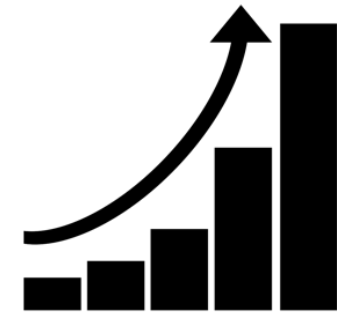
We're competing without knowing it...



Staying in budget
(time + money)



First to Market
(FTM)

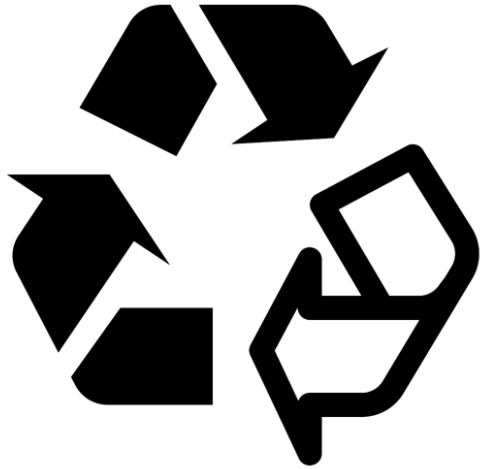


Continuous
growth

Budget = reckless re-use



Staying in budget
(time + money)



Let's start with a framework...



You Retweeted



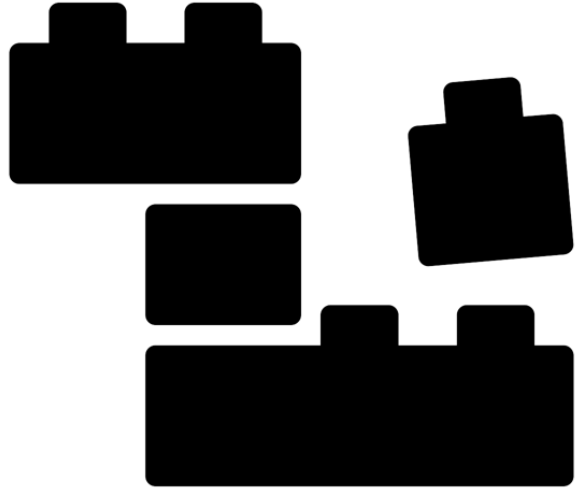
Dylan Beattie @dylanbeattie · Feb 20

```
$ create-react-app my-react-app  
$ cd my-react-app  
$ cloc .
```

Before you even open a text editor, your project contains 1.5 million lines of code. Most of it contributed by volunteers and enthusiasts. No formal review or release process.

This is 'normal' in modern web development.

Language	files	blank	comment	code
JavaScript	12193	204003	223944	1127134
Markdown	1755	71061	2	183288
JSON	1767	523	0	132948
TypeScript	1089	4499	24967	28653
HTML	55	108	30	8198
C/C++ Header	21	1148	347	5959
YAML	162	146	135	5121
XML	12	241	13	2271
CoffeeScript	27	591	51	1513
CSS	23	76	24	1412
Bourne Shell	8	173	129	902
make	23	209	197	670
C++	10	100	54	474
Windows Module Definition	5	88	0	442
Python	1	81	175	355
D	5	0	0	71
Handlebars	2	10	0	54
Bourne Again Shell	2	11	1	43
TOML	1	2	0	39
Lisp	1	0	0	6
DOS Batch	1	0	0	2
SUM:	17163	283070	250069	1499555



The web is
assembled, not
authored...



Katie Hempenius @katiehempenius · Feb 17
"27% of sites are 90-99% third party content"

By @paulcalvano w/ @HTTPArchive data

What is the distribution of 1st party vs 3rd party res...

The last time we looked at this query, we had 470K sites in the HTTP Archive. Now that we're upwards of 1.3 million, I was wondering whether this changed much. S...

discuss.httparchive.org

30% 35% 40% 45% 50% 55% 60% 65%
% of Third Party Content

38% of sites have between 75% and 99% 3rd party content

30 61

<https://discuss.httparchive.org/t/what-is-the-distribution-of-1st-party-vs-3rd-party-resources/100/14>

!sinkable



First to market – get it out!



First to Market
(FTM)

FTM = FML for quality



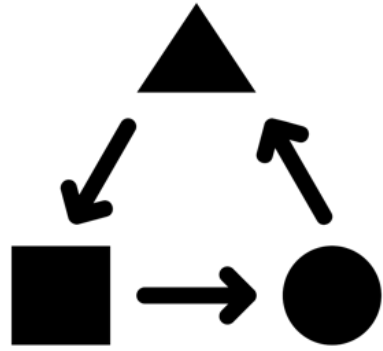
First to Market
(FTM)



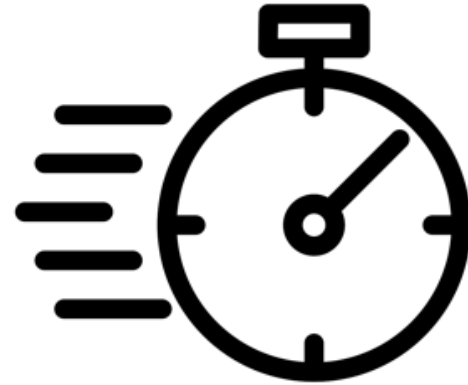
!given



Accessibility



Interoperability



Performance



Security

MVP



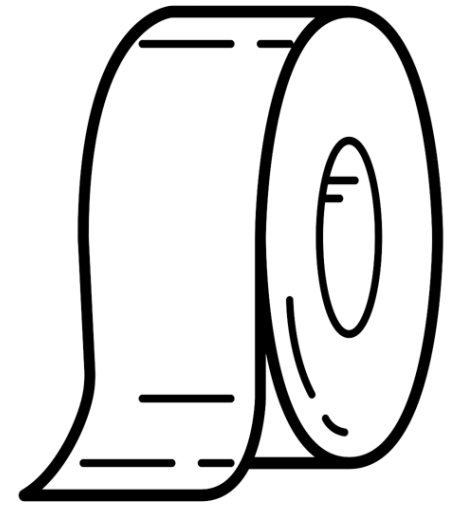
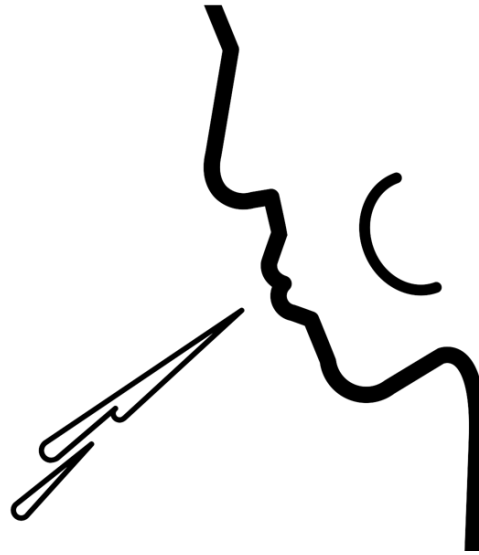
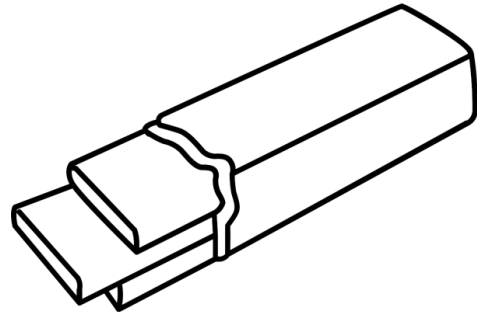
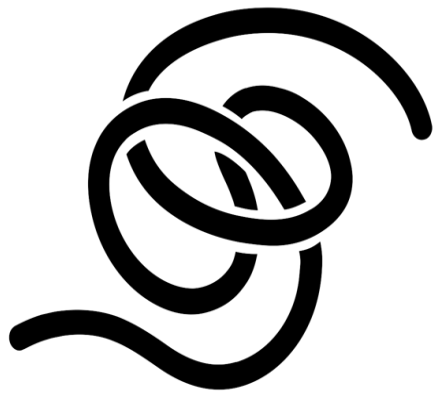
First to Market
(FTM)

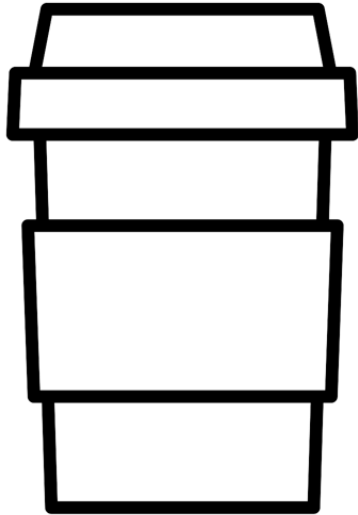
MVP = Minimum Viable Product



First to Market
(FTM)

MVP = Most Validation Pending





Understandable
throwaway culture...



I am John Cooney (jmtb02)

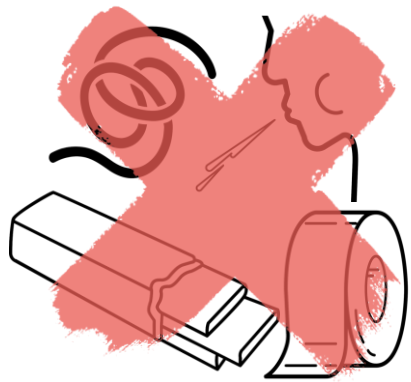
- 15-year Flash game developer
- Developed over a 100 flash games, culminating in plays into the billions... including Achievement Unlocked, This is the Only Level, Hedgehog Launch, Dark Cut, Exit Path, Corporation Inc
- Creator of **jmtb02 Studios**, my poorly named Flash games company, and **Flash Rights**, a resource for Flash developers protecting their work through copyright and fair use
- Head of Game Development at **Armor Games** for 5 years
- Director of **Launchpad** and **Premium Games** at **Kongregate** for 4 years

The Flash Games Postmortem

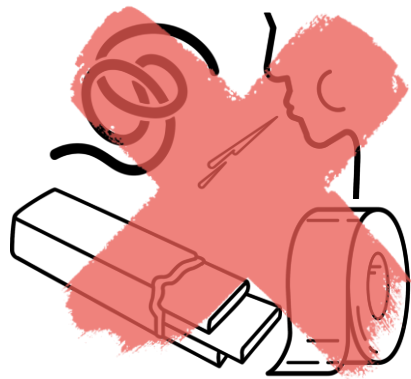
114,207 views

2.4K 94 SHARE SAVE ...

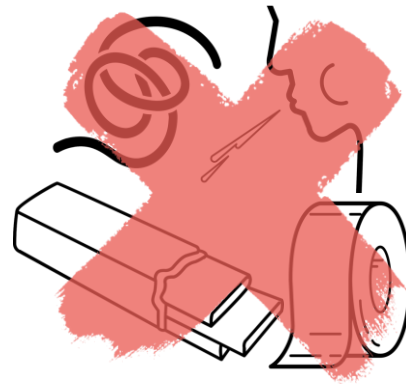
A series of quick wins...



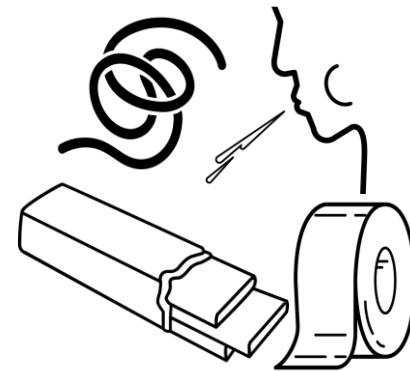
MVP#1



MVP#2



MVP#3



MVP#4

...

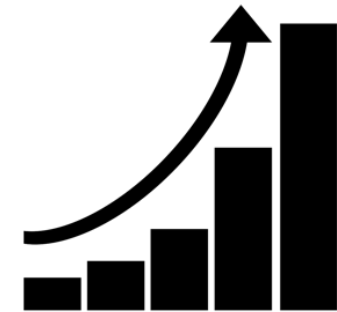
Alas, on the web...



Staying in budget
(time + money)

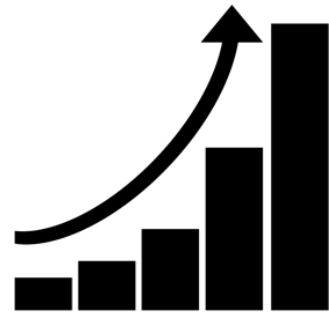


First to Market
(FTM)



Continuous
growth

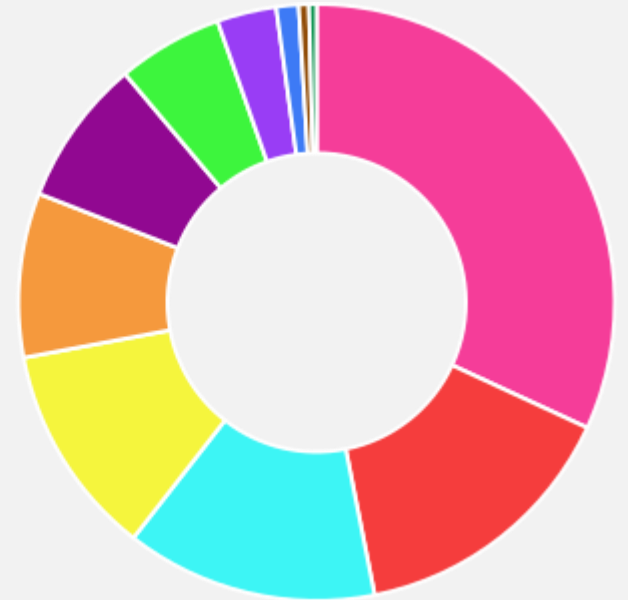
Growth = stickiness + analytics



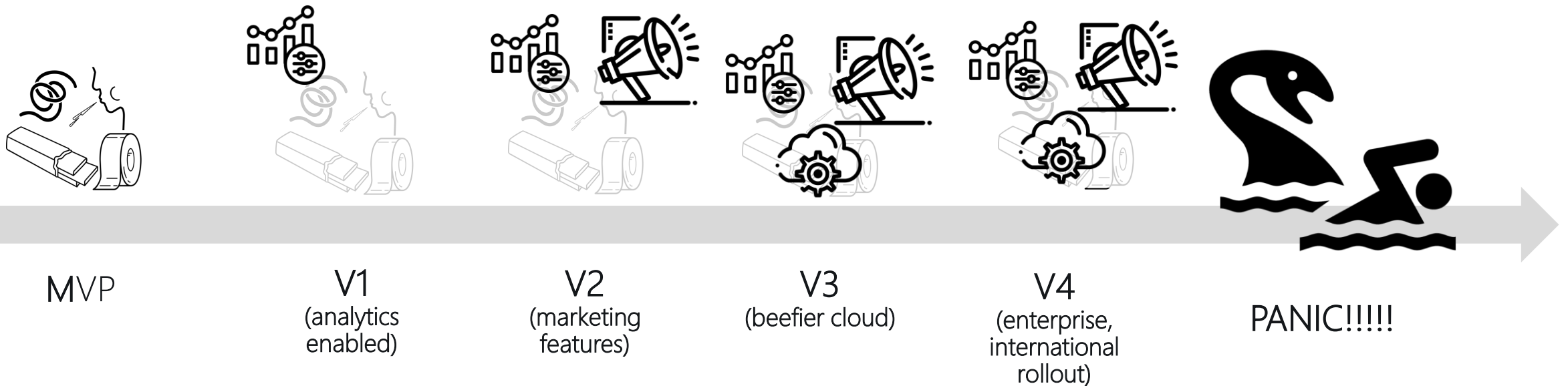
Continuous
growth



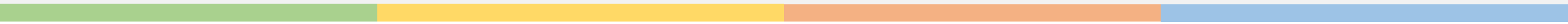
Third party slowness and cruft...



Reuse keeps budgets in check



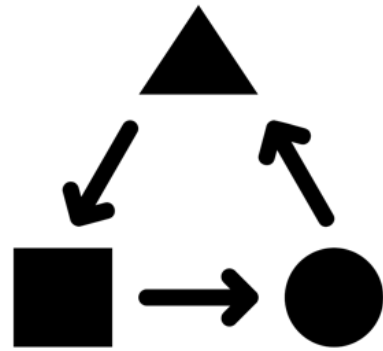
This is the moment when things
happen...



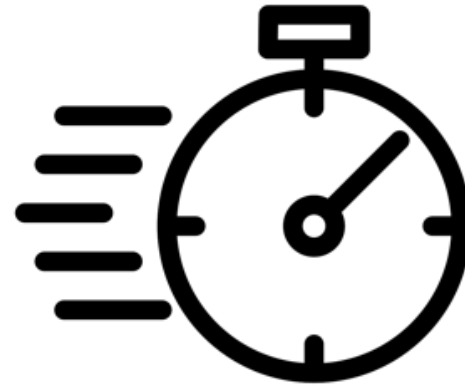
Let's hire some consultants to help us with...



Accessibility



Interoperability



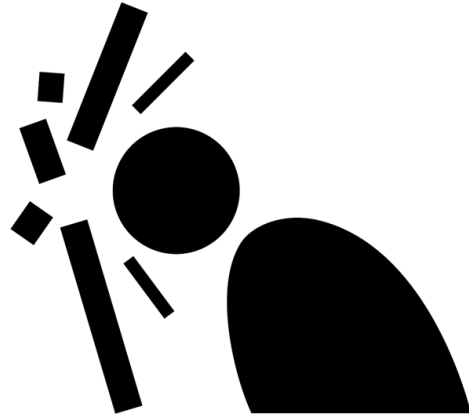
Performance



Security

Because when it costs money, it is
>important

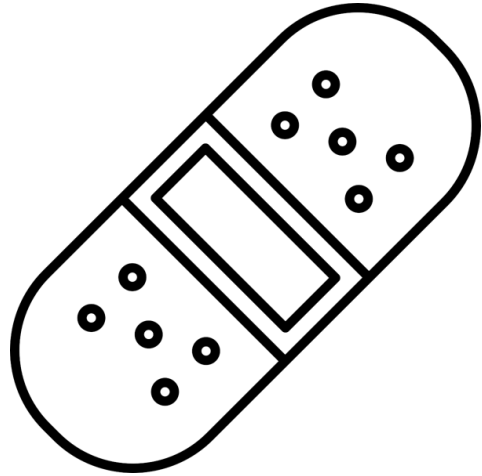




!easy



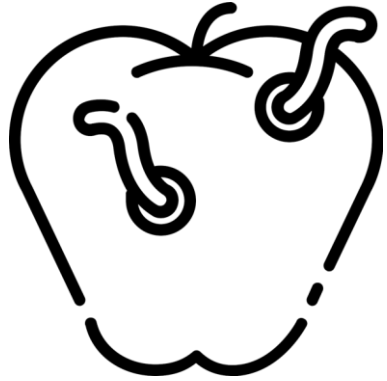
- Fixing a non-performing, inaccessible, insecure product at this stage of the process is a lot of work
- There are ways to do it, but it involves changes in the process – and people hate change
- There's always a lot of blame going around, which doesn't help either
- Often, we pile on more products and processes, adding to the cruft



Patches and "solutions"...



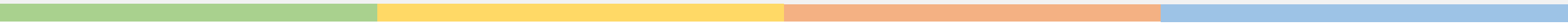
- AMP
- Safari limiting JavaScript
- Chrome's "Never Slow Mode"
- All kind of blockers



The web is a
platform ripe for
abuse.

- The web is defined as forgiving to developer error
- Parsers and browsers need to swallow a lot of terrible code and still make something useful of it
- Years of competition in browsers and platforms created legacy non-standard code that needs support
- This means not only cruft, but also security attack vectors
- We can't stop developers from being sloppy...

Well,  ...





What can *we* do?



Front-end:
HTML/CSS/JS

Back-end
CMS/Frameworks
Servers/Cloud
DevOps
IT
Legal
Sales
Shareholders
Investors



What can **we** do?



- Stop arguing amongst ourselves which technology is better
- Contribute to and clean up products that people use without thinking about them (CMS, build tools, frameworks...)
- Take our time to understand why people do things wrong before telling them off that they do
- Earn our seat at the table where stupid decisions are being made



What can **we** do?



- Optimise our workflows to include required quality control as soon as possible
 - npmjs.com/package/hint
 - css-tricks.com/make-it-hard-to-screw-up-driven-development/
- Brush up our knowledge of development tools and give feedback to them – we are their customers

The way we work now is becoming
less and less **important**



Most of the products created now
use other people's code...



So, let's build **accessible, secure**
and **performant solutions** for
people to re-use!





Power to the next generation!



Thanks!
Chris Heilmann
@codepo8

