

Migrate a React application to server-side with Next.js

Nataliya Karatkova

 [@meuwka](https://twitter.com/meuwka)



Based on a true story

Get £25 off your 1st 10+ hours. Code: WOW Ends in 06:52:07



[Our services](#)

[Driving instructors](#)

[About us](#)

[Support](#)

[Prices](#)

[Locations](#)

[Sign in](#)

Learn to drive the smart way.

Save 35% compared to traditional driving schools.

See lesson prices near you

[GO](#)



Trustpilot Midrive is rated 8.9 out of 10 by Trustpilot based on over 500 reviews.

#1 national provider



Top-rated national driving school on Trustpilot.

Top-rated instructors

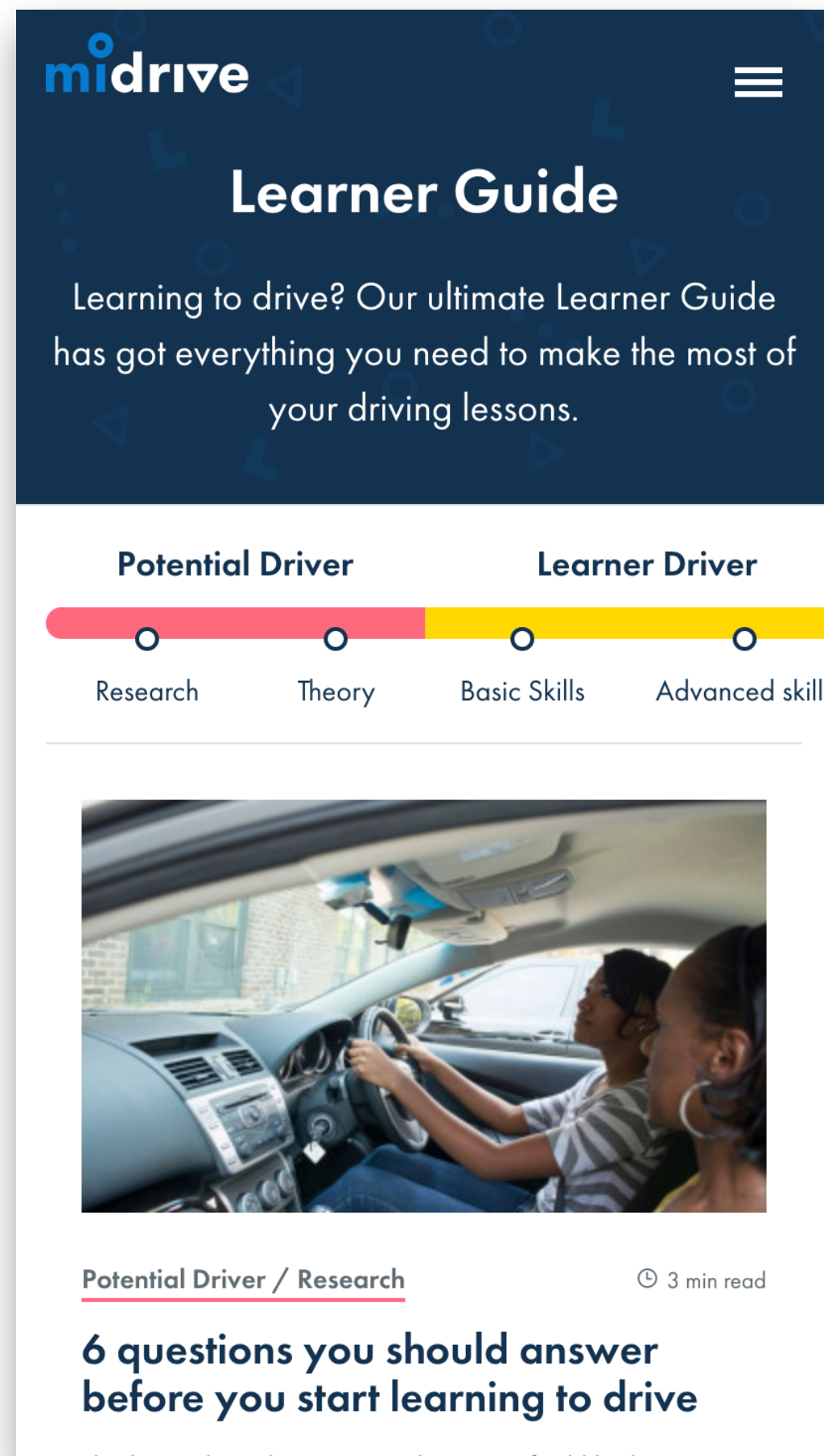


All our instructors are rated at least 4 out of 5 by learners.

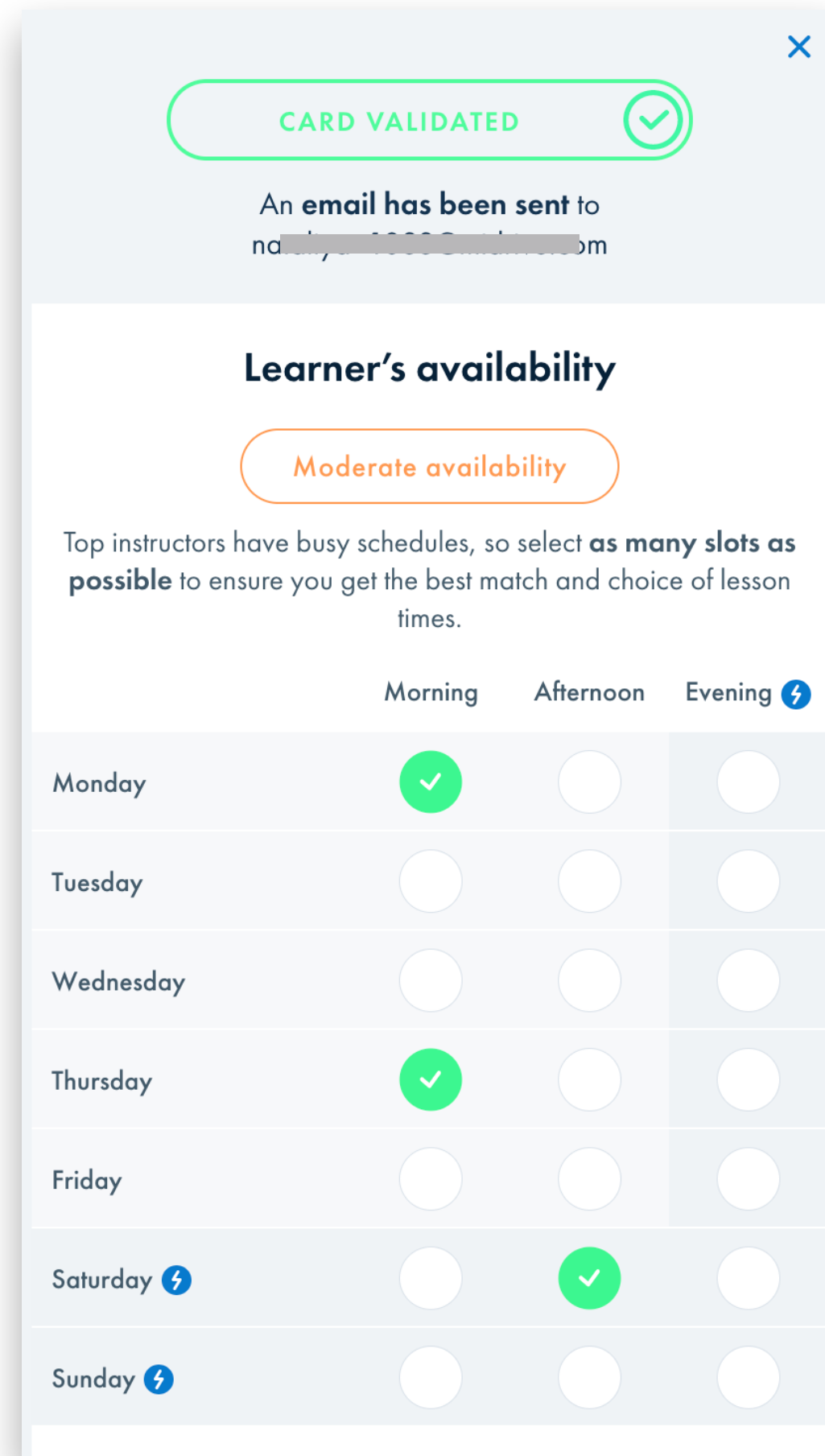
Modern tools



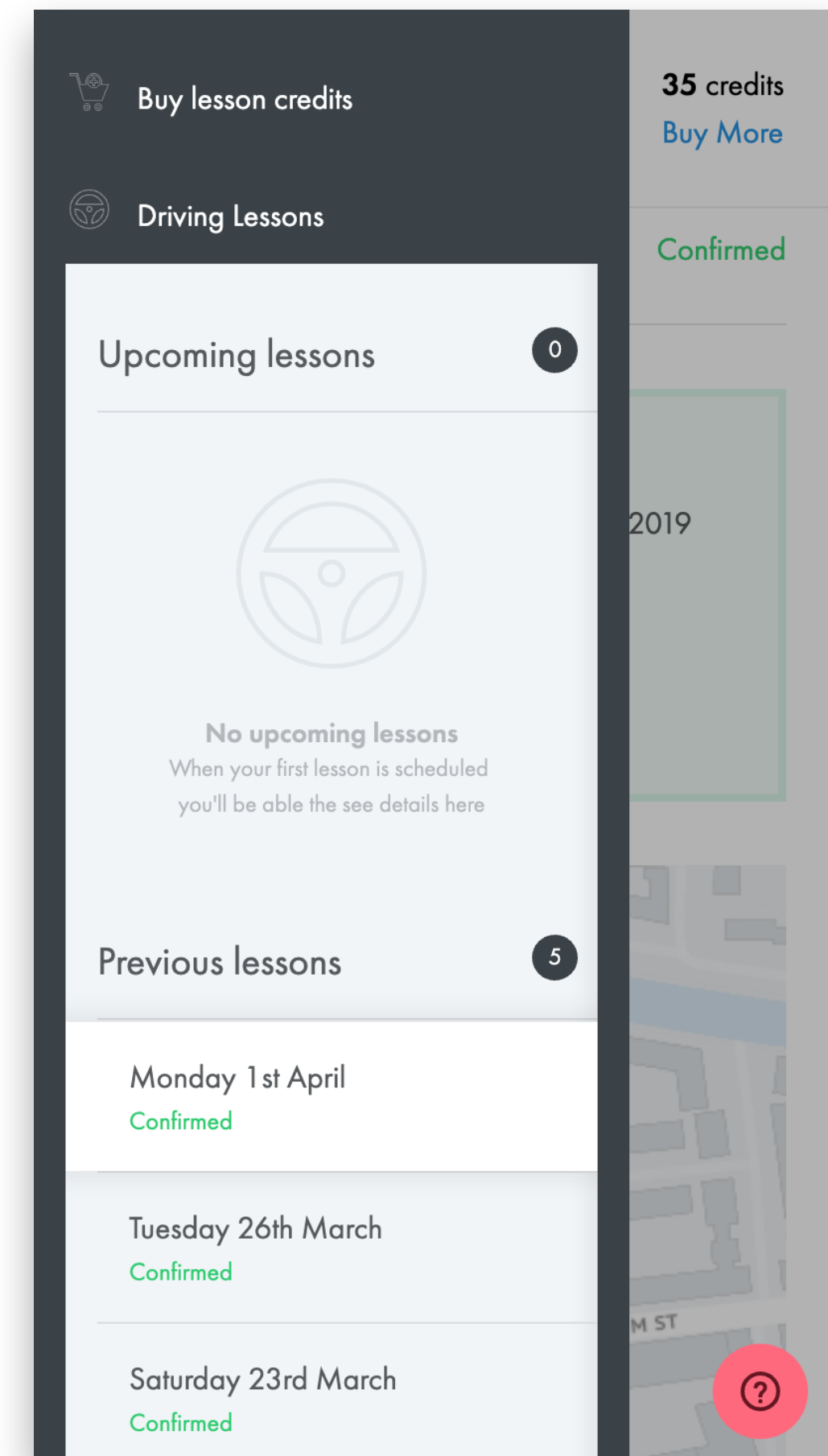
Learn faster with the #1 app for passing the practical test.



**Content
Headless CMS**

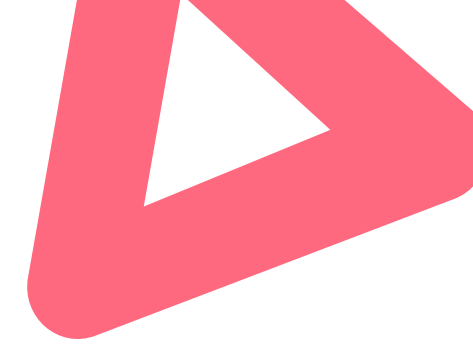


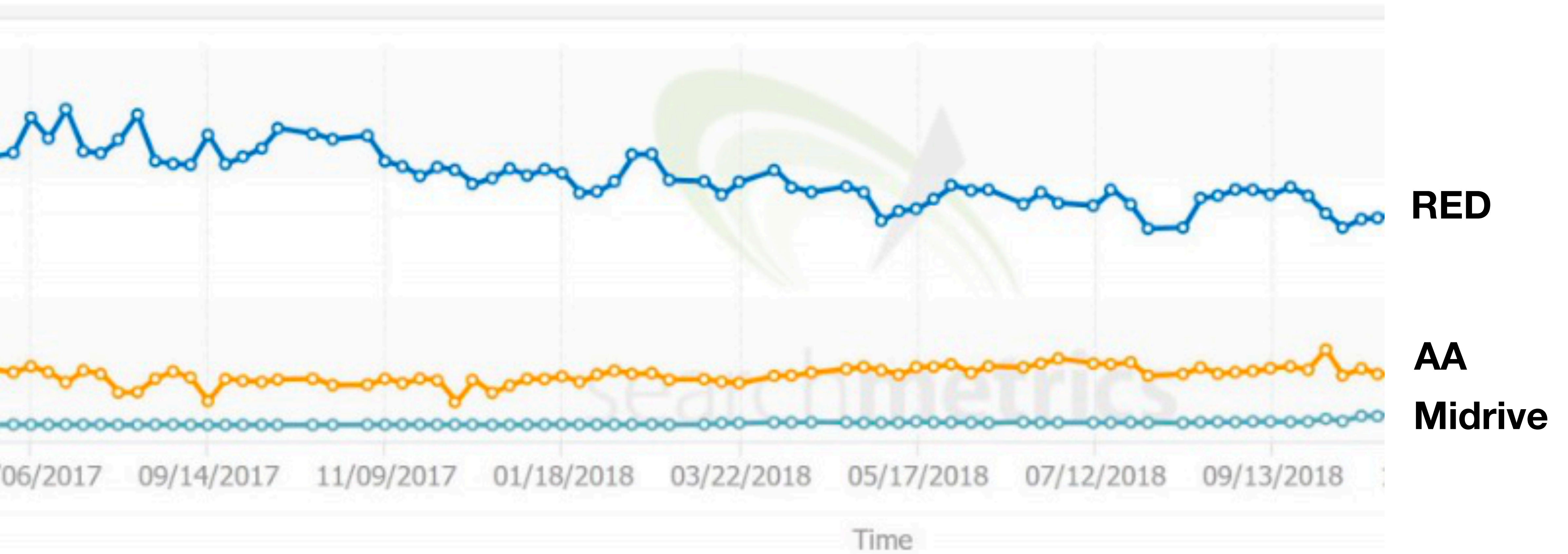
Checkout



Dashboard

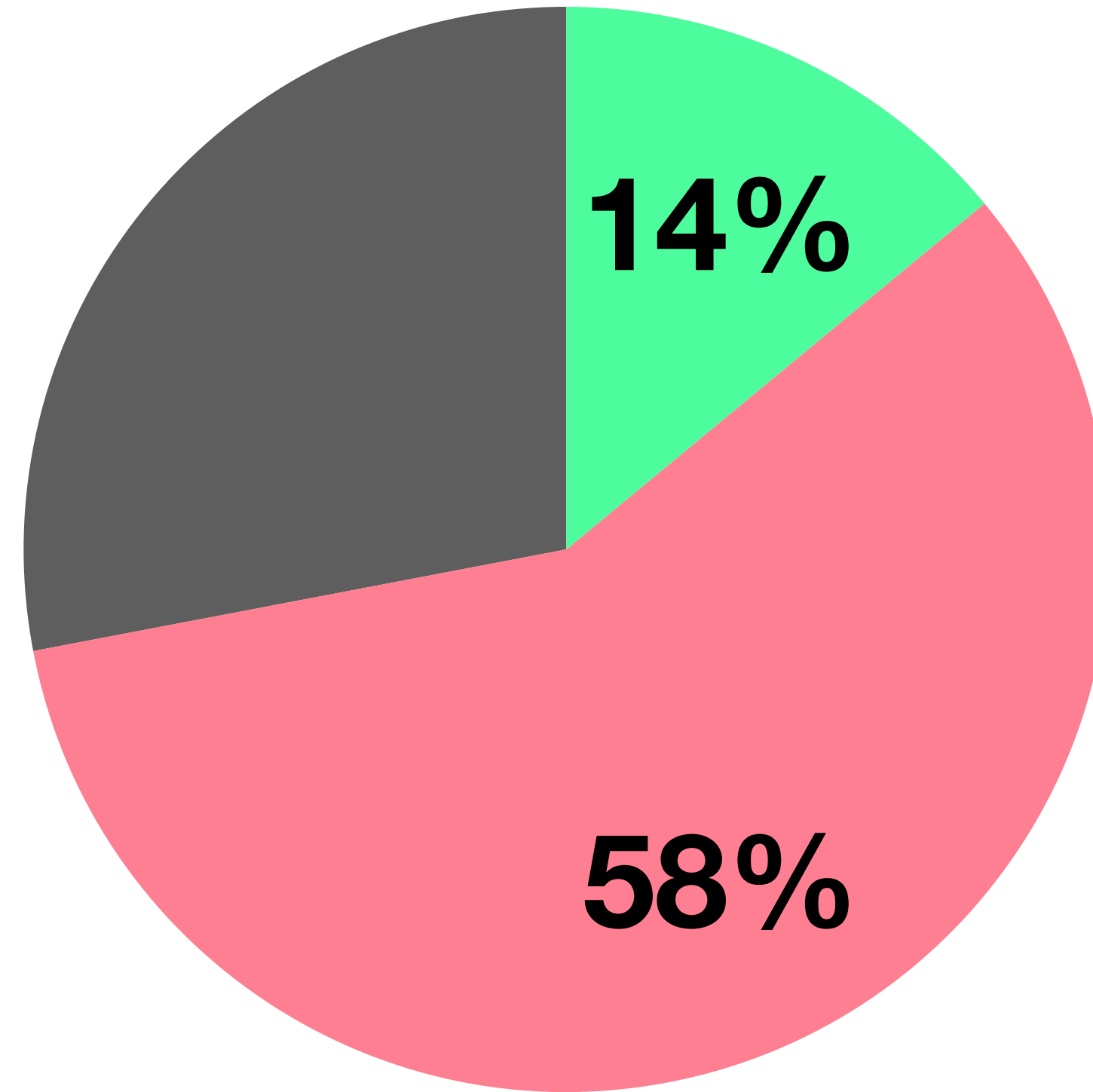
**Before
(2018)**



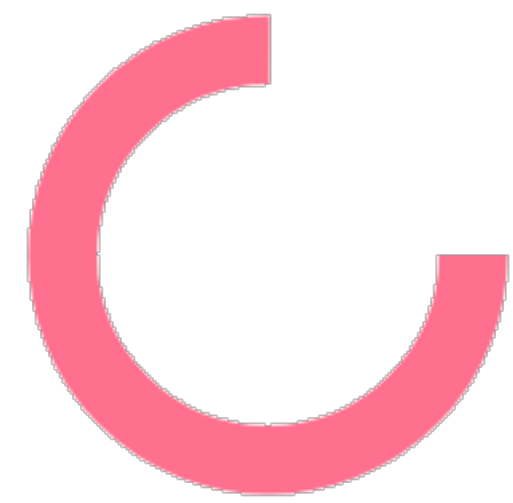


SEO Visibility

● Organic search ● Paid search ● Other



Channels



and moreover

The image shows a browser window displaying the Google Webmaster Central Blog. The page title is "Using page speed in mobile search ranking" with a date of "Wednesday, January 17, 2018". The article text discusses the "Speed Update" and its impact on mobile search ranking. On the right side, there is a sidebar with a search bar, navigation links for "Labels", "Archive", and "Feed", and a social media section for "Google Webmasters" with a YouTube channel link and a "Follow @googlewmc" button.

Google Webmaster Central Blog
Official news on crawling and indexing sites for the Google index

Using page speed in mobile search ranking

Wednesday, January 17, 2018

Update July 9, 2018: The Speed Update is now rolling out for all users.

People want to be able to find answers to their questions as fast as possible – [studies](#) show that people really care about the speed of a page. Although speed has been used in ranking for some time, that [signal](#) was focused on desktop searches. Today we're announcing that starting in July 2018, page speed will be a ranking factor for mobile searches.

The "Speed Update," as we're calling it, will only affect pages that deliver the slowest experience to users and will only affect a small percentage of queries. It applies the same standard to all pages, regardless of the technology used to build the page. The intent of the search query is still a very strong signal, so a slow page may still rank highly if it has great, relevant content.

We encourage developers to think broadly about how [performance](#) affects a user's

Hey! Check here if your site is mobile-friendly.

Search blog ...

Labels

Archive

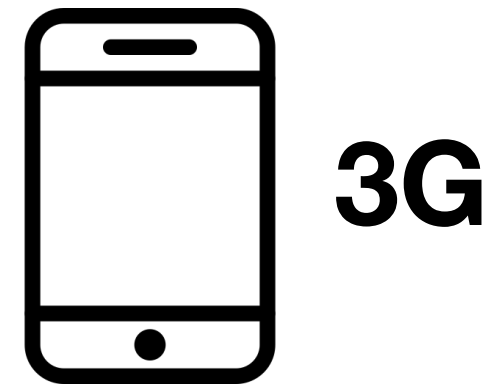
Feed

Google Webmasters

YouTube 294K

Follow @googlewmc

<https://webmasters.googleblog.com/2018/01/using-page-speed-in-mobile-search.html>



Performance



Accessibility



Best Practices



SEO

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

Performance

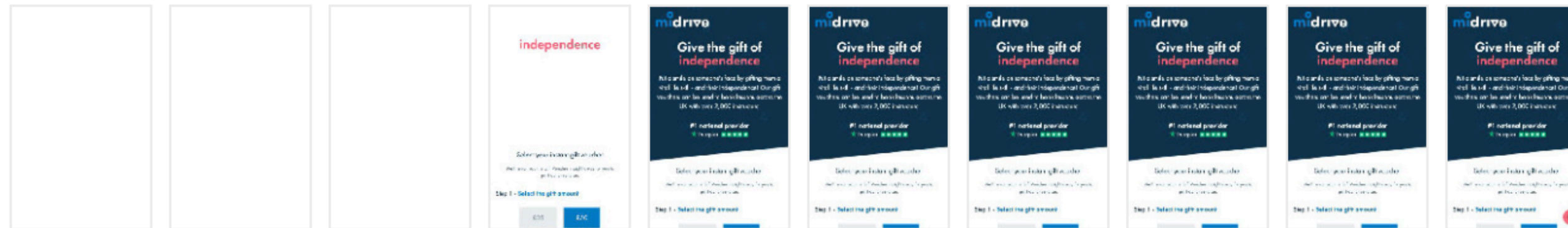


Metrics

First Contentful Paint	5.7 s	First Meaningful Paint	5.7 s
Speed Index	6.9 s	First CPU Idle	14.2 s
Time to Interactive	14.2 s	Estimated Input Latency	300 ms

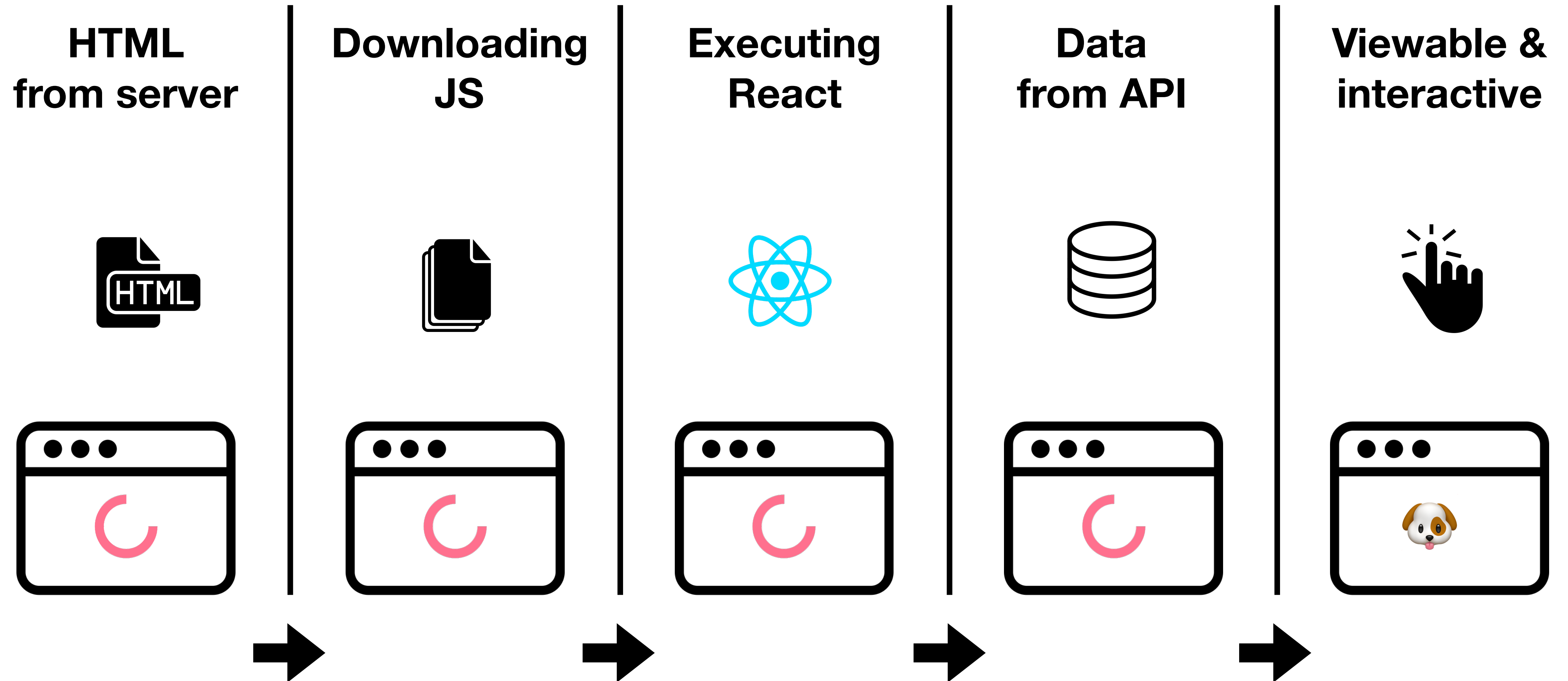
[View Trace](#)

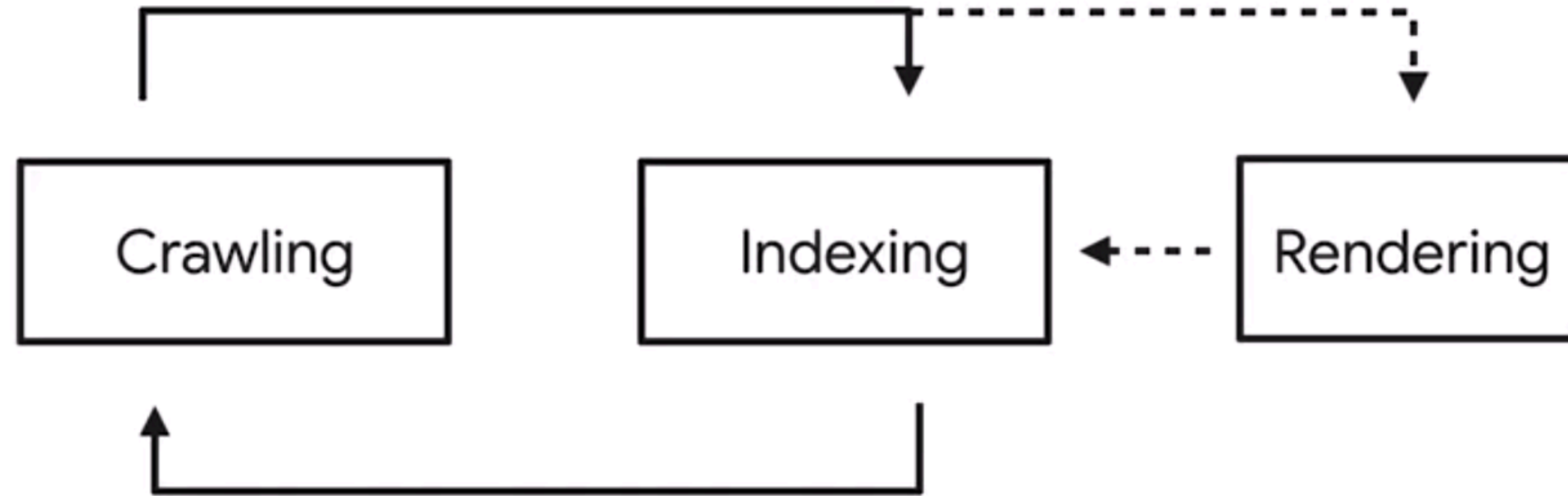
Values are estimated and may vary.





React CSR





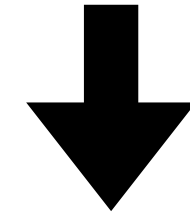
How Google Search indexes JS sites

<https://youtu.be/LXF8bM4g-J4>

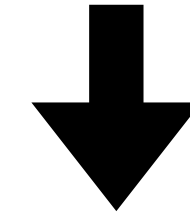
DX + UX + SEO = ?

Do you need an SEO?

YES

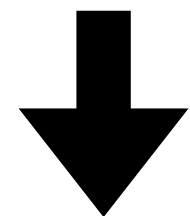


NO

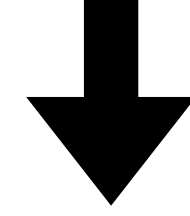


**Does your data
update frequently?**

YES



NO



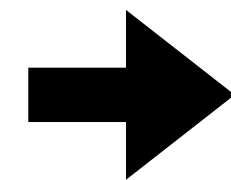
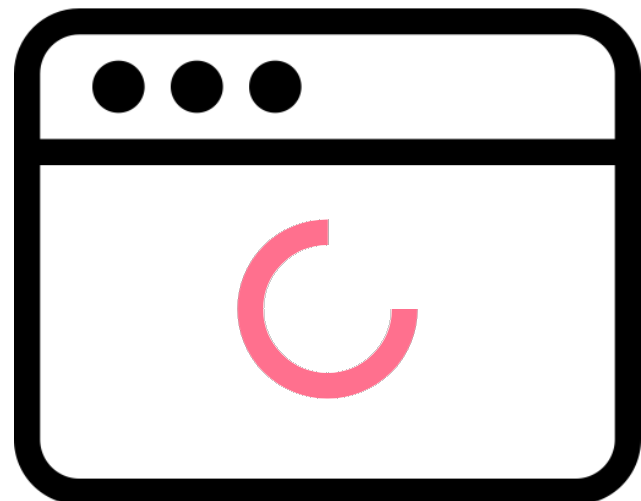
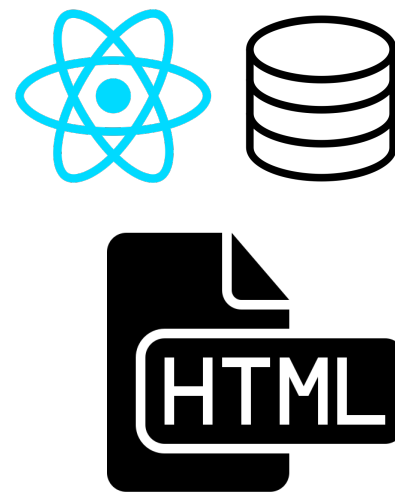
CSR

SSR

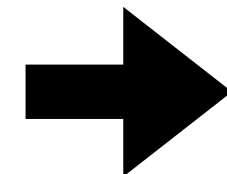
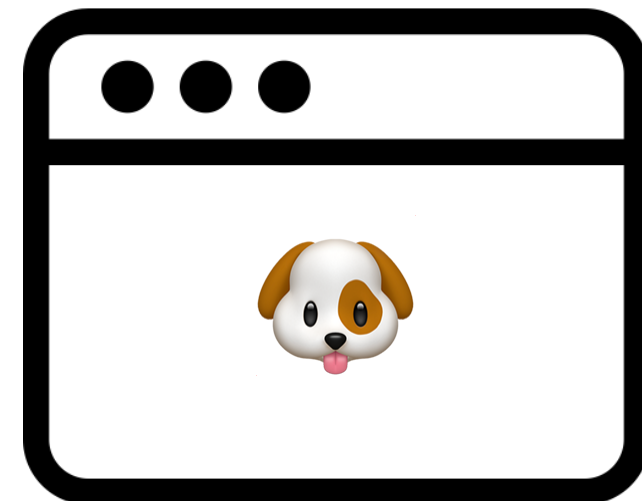
**Static Site
Generator**

React SSR

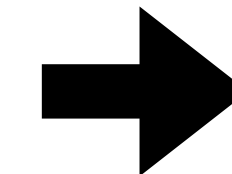
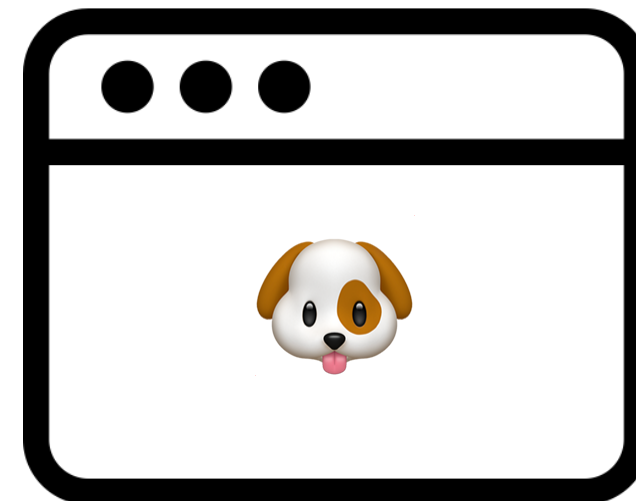
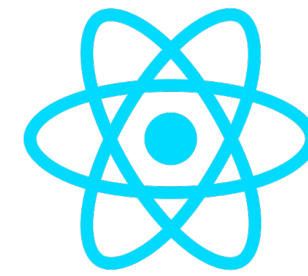
HTML
from server



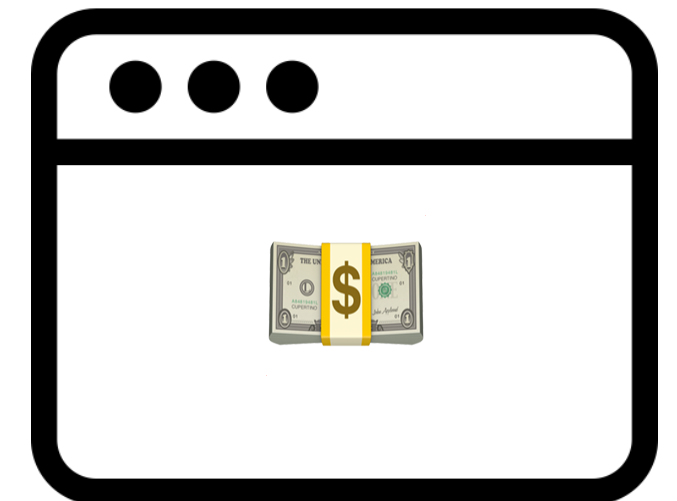
Viewable &
downloading JS



Executing
React



Interactive



SSR 

✘ SSR is hard enough

✘ High cost of rendering

✘ SSR is hard

Next.js

The image shows a browser window displaying the Next.js website. The browser's address bar shows 'nextjs.org'. The website's navigation bar includes 'Features', 'Learn', 'Docs', 'NEXT.js', 'Showcase', and 'Blog'. The main heading is 'Server-Side Rendering Done Right', followed by the subheading 'Take the pain out of creating Universal React apps with Next.js'. A blue 'Learn More' button is centered below the subheading. Below this, four features are listed: 'Exceptional Performance', 'SEO Ready', 'Deploy Anywhere', and 'Zero Configuration'. A diagram illustrates the server-side rendering process: a code editor on the left shows a React component using the `<Toast>` component to render 'This site is **Server Side Rendered**'. This code is processed by a server stack (represented by three server icons) and the resulting HTML is rendered in a browser window on the right, showing 'This site is **Server Side Rendered**'.

Features

Learn Docs

NEXT.js

Showcase Blog

Server-Side Rendering Done Right

Take the pain out of creating Universal React apps with **Next.js**

[Learn More](#)

- ✓ Exceptional Performance
- ✓ SEO Ready
- ✓ Deploy Anywhere
- ✓ Zero Configuration

```
import Toast from './react-components/toast'

export default function HomePage() {
  return (
    <Toast>
      This site is
      <strong>Server Side Rendered</strong>
    </Toast>
  )
}
```

http://nextjs-site.now.sh

This site is **Server Side Rendered**

Starter kit

- **194+ examples** <https://github.com/zeit/next.js/tree/canary/examples/>
- **create-next-app** <https://open.segment.com/create-next-app/>

Demo 1

- Router, prefetch, code splitting
- Data fetching with `getInitialProps`

 **SSR is hard**

 **Next.js**

The image shows a YouTube video player interface. The main content area displays a presentation slide with the text "Regret 2: Not using Next.js" in the center. The slide is framed by a bright green border. In the top right corner of the slide, there is a React logo and the text "REACT AMSTERDAM". In the bottom right corner of the slide, there is a Twitter handle "@mxstbr". To the right of the slide, a video thumbnail shows a man, Max Stoiber, speaking at a podium. He is wearing a grey NASA-branded sweatshirt. The background of the video is green with the word "AMSTERDAM" visible. Below the video, the name "Max Stoiber" and the title "Tech Regrets at Spectrum" are displayed. At the bottom of the player, there are logos for "ag Grid" and "FOCUS REACTIVE TRUSTED REACT EXPERTS". The video progress bar shows the video is at 7:10 out of 17:27. Standard video player controls like play, volume, and full screen are visible.

<https://youtu.be/UC-bUkbuvZs>

Migration

Option 1: Proxying

Option 2: Moving from CRA and React Router to Next

<https://spectrum.chat/next-js/general/moving-from-cra-and-react-router-to-next~01a66b8e-496d-47a5-8cf3-8bfb2138289e>

```
const proxy = require("http-proxy-middleware");

const pageProxy = proxy(checkIfLegacyRoute, {
  target: LEGACY_URL,
  changeOrigin: true
});

const server = express();

server.use(pageProxy);
```

Proxying

Dynamic routes

```
server.get("/example/:id", (req, res) => {  
    return app.render(req, res, "/example", { id: req.params.id });  
});
```

Custom server

```
static async getInitialProps({ query }) {  
  const response = await fetch(  
    `http://jsonplaceholder.typicode.com/posts/${query.id}`  
  );  
  const example = await response.json();  
  
  return { example };  
}
```

Page

**Non SSR components
(window is not defined)**

```
import dynamic from 'next/dynamic';  
const ReactJson = dynamic(  
  () => import('react-json-view'),  
  { ssr: false }  
);  
  
...  
  
<ReactJson src={data} />
```

Dynamic Import

Authentication

```
static async getInitialProps({ req, res, store, query }) {  
  if (req) {  
    store.dispatch(setToken(req));  
  }  
}
```

page


```
export const setToken = req => async dispatch => {  
  const authToken = getCookie("authToken", req);
```

```
  ...
```

redux/actions

```
try {
  const payload = await accountApi.setToken(authToken);

  return dispatch({
    type: actionTypes.AUTHORIZED,
    payload
  });
}
catch (err) {
  return dispatch({
    type: actionTypes.NOT_AUTHORIZED
  });
}
```

redux/actions

```
import jsCookie from "js-cookie";

export const getCookie = (name, req) => {
  const isServer = req;

  if (isServer) return getCookieServerSide(name, req);
  return jsCookie.get(name);
};
```

services/cookies

```
import httpCookie from "cookie";

const getCookieServerSide = (name, req) => {
  const { cookie } = req.headers;
  const parsedCookie = httpCookie.parse(cookie || "");

  return parsedCookie[name];
};
```

services/cookies


```
const payload = await accountApi.login(data);
```

```
setCookie("authToken", payload.authToken);
```

login

github.com/zeit/next.js/tree/canary/examples/with-cookie-auth

Cookie-based authentication

 **Next.js**

✘ High cost of rendering

✘ Unnecessary re-rendering

✘ `renderToString()` ↑ TTFB ↑ CPU

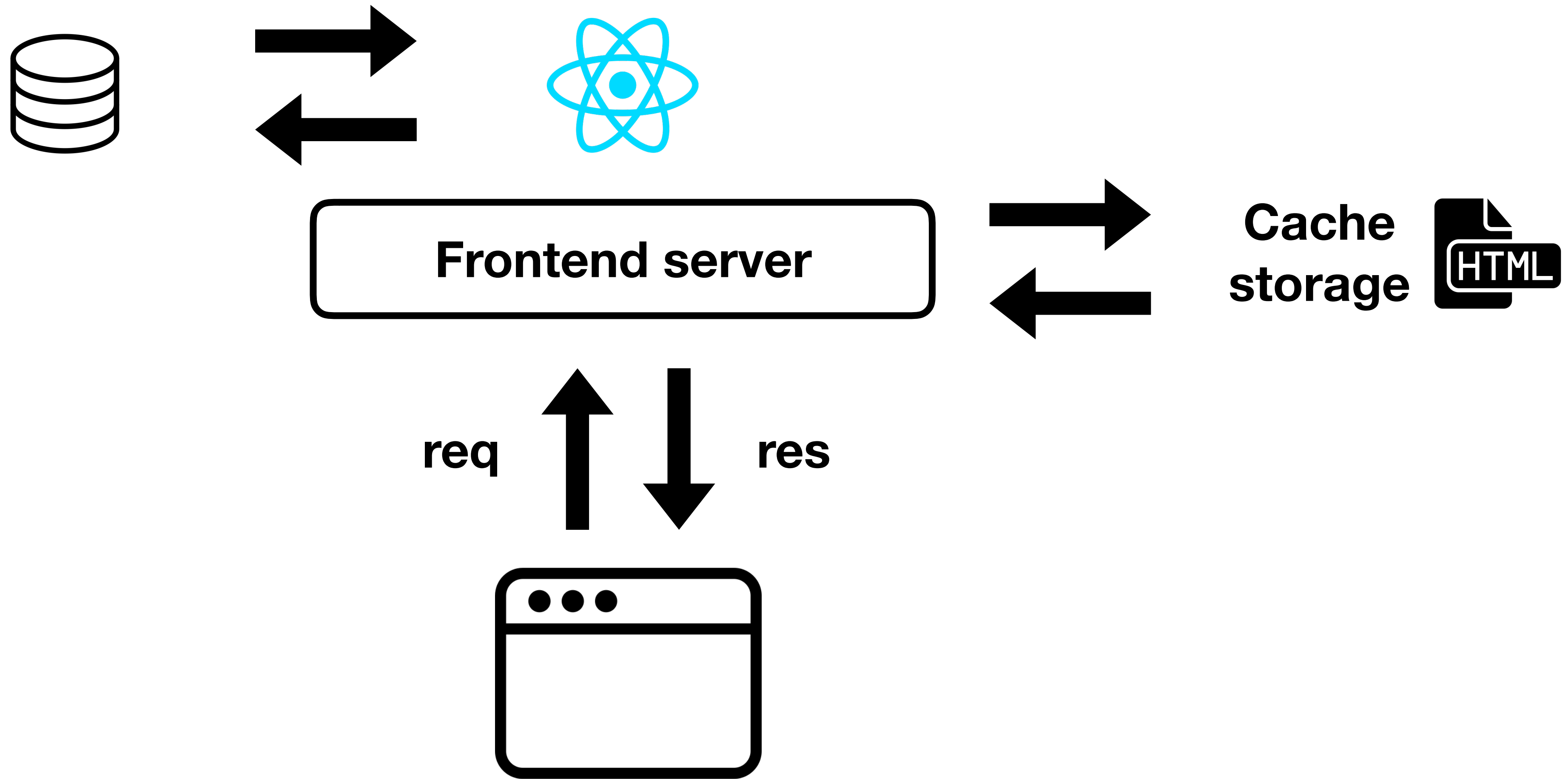
✔ Server-side caching

90K visitors in a month

The image shows a screenshot of the Heroku pricing page. A blue rectangular box highlights the 'Standard 1x' plan, which is the selected option. The page displays various pricing tiers: Free (\$0), Hobby (\$7/dyno per month), Standard 1x (\$25/dyno per month), Standard 2x (\$50/dyno per month), Performance M (\$250/dyno per month), and Performance L (\$500/dyno per month). A 'PROFESSIONAL' section is also visible, which is partially obscured by the blue box. The table below details the features and specifications for each plan.

	Free \$0	Hobby \$7/dyno per month	Standard 1x \$25/dyno per month	Standard 2x \$50/dyno per month	Performance M \$250/dyno per month	Performance L \$500/dyno per month
What is it good for?	Ideal for experimenting with cloud applications in a limited sandbox.	Perfect for small scale personal projects and hobby apps.	Enhanced visibility, performance, and availability for powering your production applications.		Superior performance when it's most critical for your super scale, high traffic apps.	
RAM	512MB	512MB	512MB	1GB	2.5GB	14GB
Deploy from Git	●	●	●	●	●	●
Automated OS patching	●	●	●	●	●	●
Unified logs	●	●	●	●	●	●
Number of process types	2	10	Unlimited	Unlimited	Unlimited	Unlimited
Always on	Sleeps after 30 mins of inactivity, otherwise always on depending on your remaining monthly free dyno hours.	●	●	●	●	●
Custom domains	●	●	●	●	●	●
Free SSL on custom domains		●	●	●	●	●

Our frontend server



Cache already rendered pages

```
const storage = new Storage({ url: REDIS_URL });  
const ssrCacheMaxAge = 60 * 60; // 1 hour in seconds
```

Storage


```
const cachedHtml = await storage.get({ key });  
  
if (cachedHtml) {  
  res.setHeader("x-cache", "HIT");  
  res.send(cachedHtml);  
  return;  
}
```

take from cache

```
const html = await app.renderToHTML(req, res, route, query);

if (res.statusCode === 200) {
  await storage.set({
    key,
    maxAge: ssrCacheMaxAge,
    value: html
  });
  res.setHeader("x-cache", "MISS");
}

res.send(html);
```

not in the cache

```
app.render(req, res, route, query);
```

all other cases

More detailed

- **Stream rendering to reduce TTFB and CPU load**

<https://github.com/zeit/next.js/issues/1209>

- **Component-based server-side caching:**

1. <https://github.com/walmartlabs/react-ssr-optimization>

2. <https://github.com/FormidableLabs/rapsallion>

**CSR better for servers,
SSR better for users**

 **High cost of rendering**

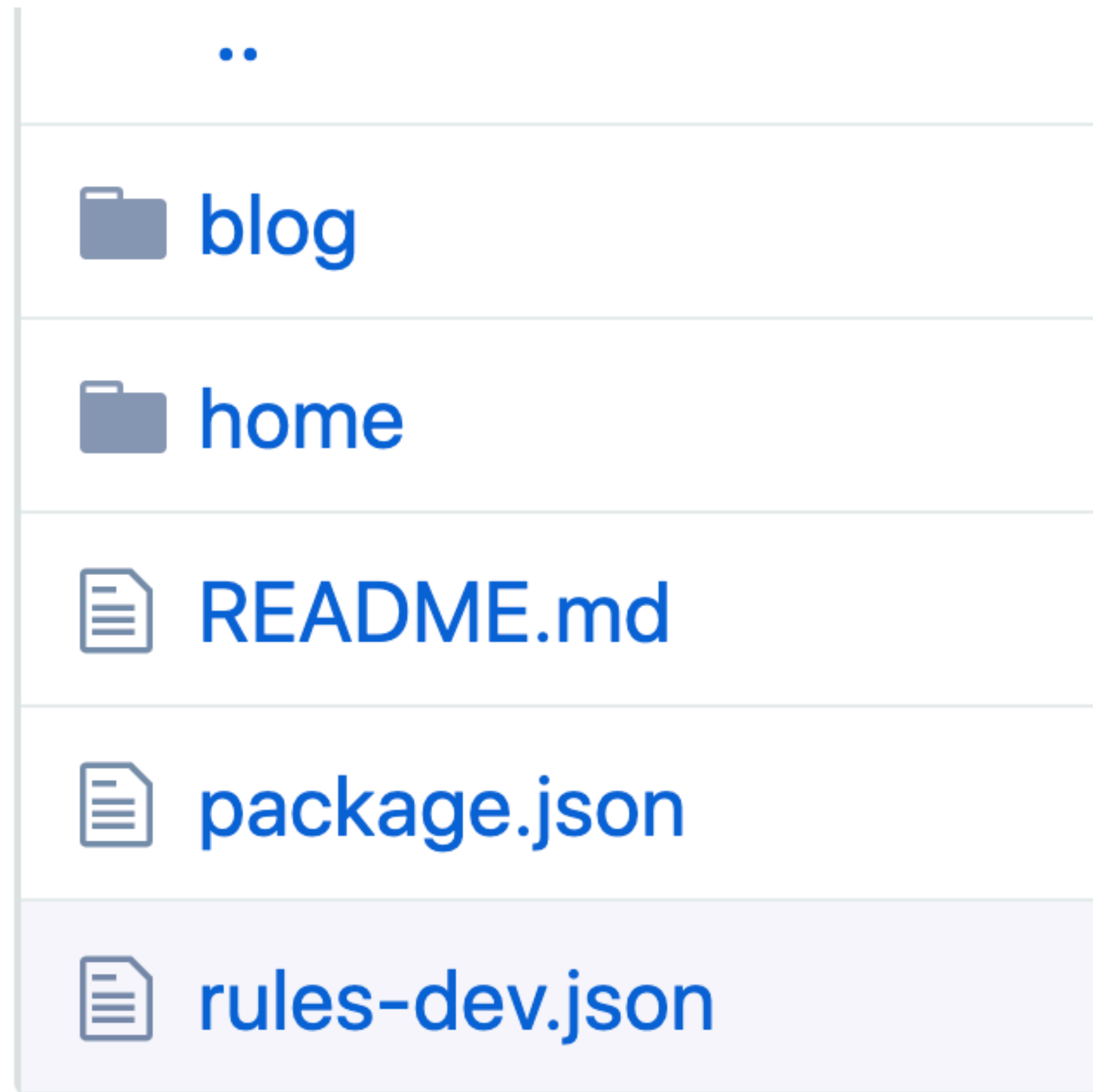
 **Caching**

Caching + *A/B* testing

```
static async getInitialProps({ Component, ctx }) {  
  if (ctx.isServer) {  
    const variant = defineVariant(["A", "B"]);  
    ctx.store.dispatch({ type: "SETVARIANT", payload: variant });  
  }  
}
```

_app.js

One more thing



Multi Zones

<https://nextjs.org/docs#multi-zones>



Tim @timneutkens · 15h

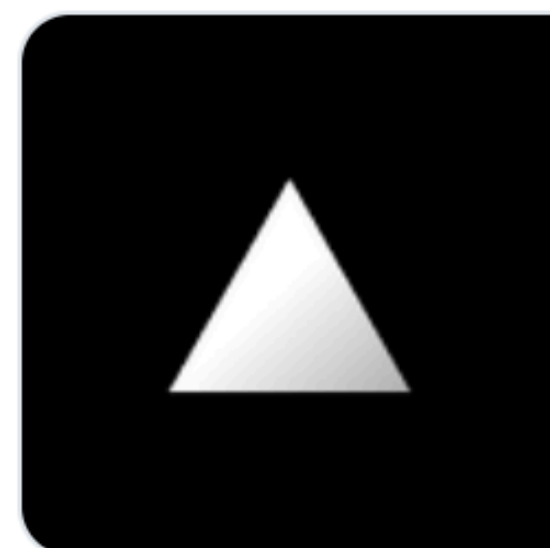


Wrote a proposal for:

- Dynamic (SSR)
- Static (Pre-rendering)
- Client-side only (CSR/SPA)

On a per-page basis in Next.js.

This allows building truly hybrid web applications 🌟



[RFC] Rendering in Next.js (SSR, Pre-rendering, CS...

Next.js currently has 2 modes of rendering: Dynamic rendering means render on demand when a request comes in. Pre-rendering means rendering to html at b...

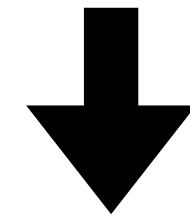
github.com

[RFC] Hybrid app

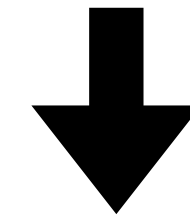
<https://github.com/zeit/next.js/issues/7355>

Do you need an SEO?

YES

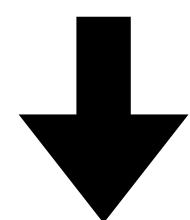


NO

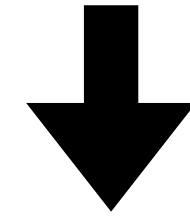


**Does your data
update frequently?**

YES



NO

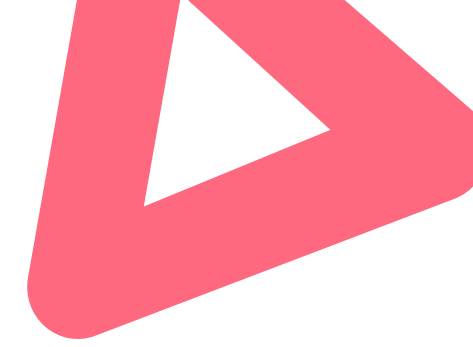


CSR

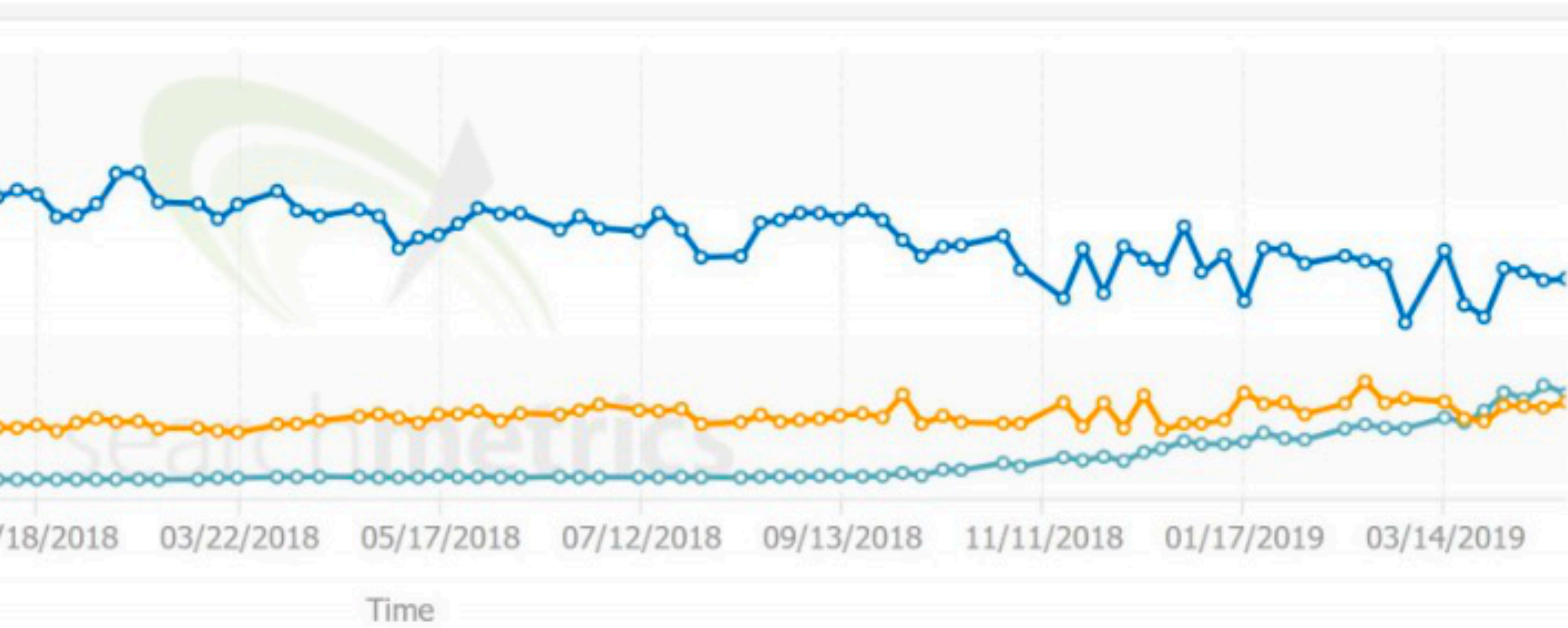
SSR

**Static Site
Generator**

Results



SSR + SEO Marketing = ❤️



RED
Midrive
AA

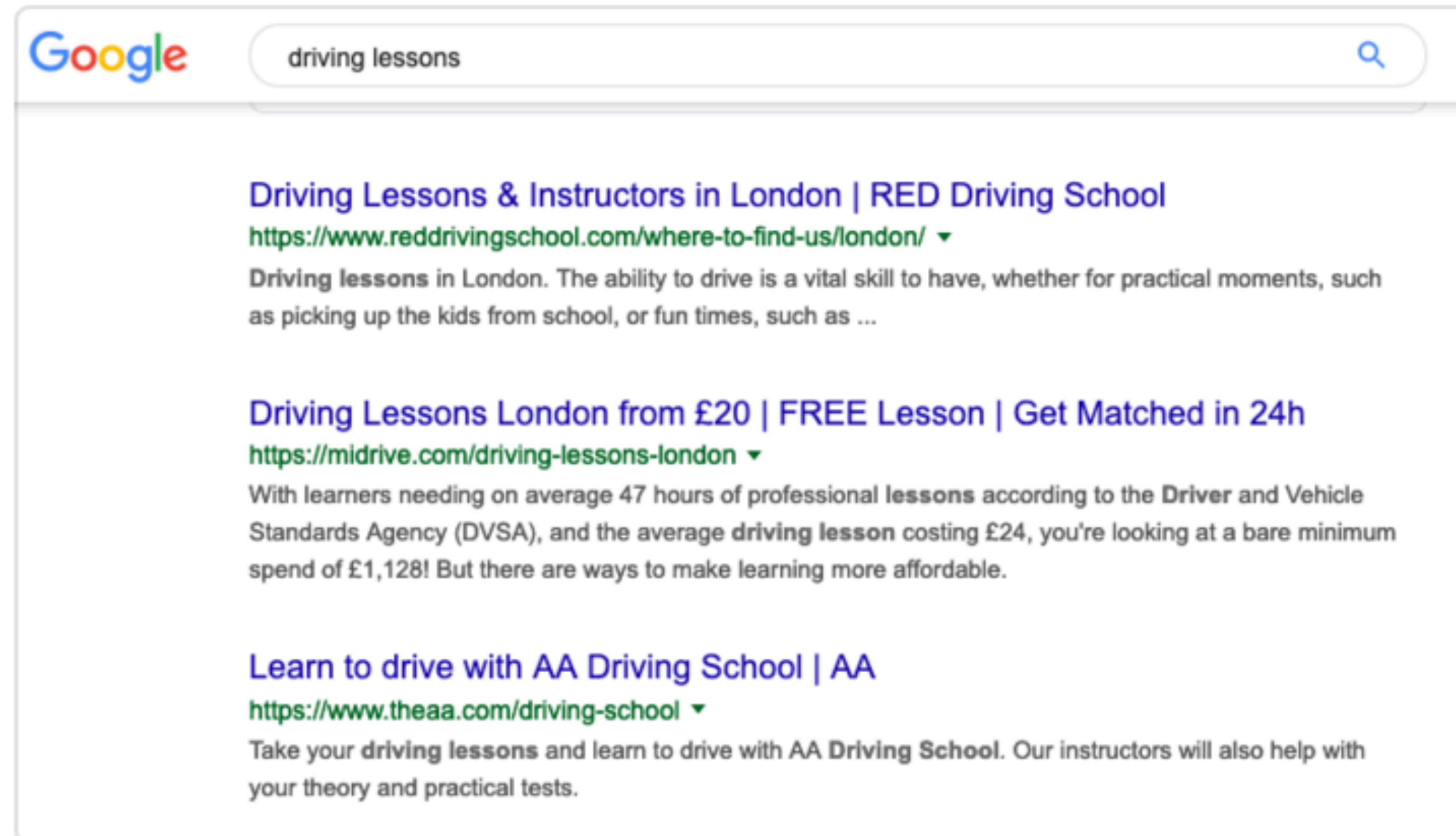
SEO Visibility



Cormac 1:08 PM

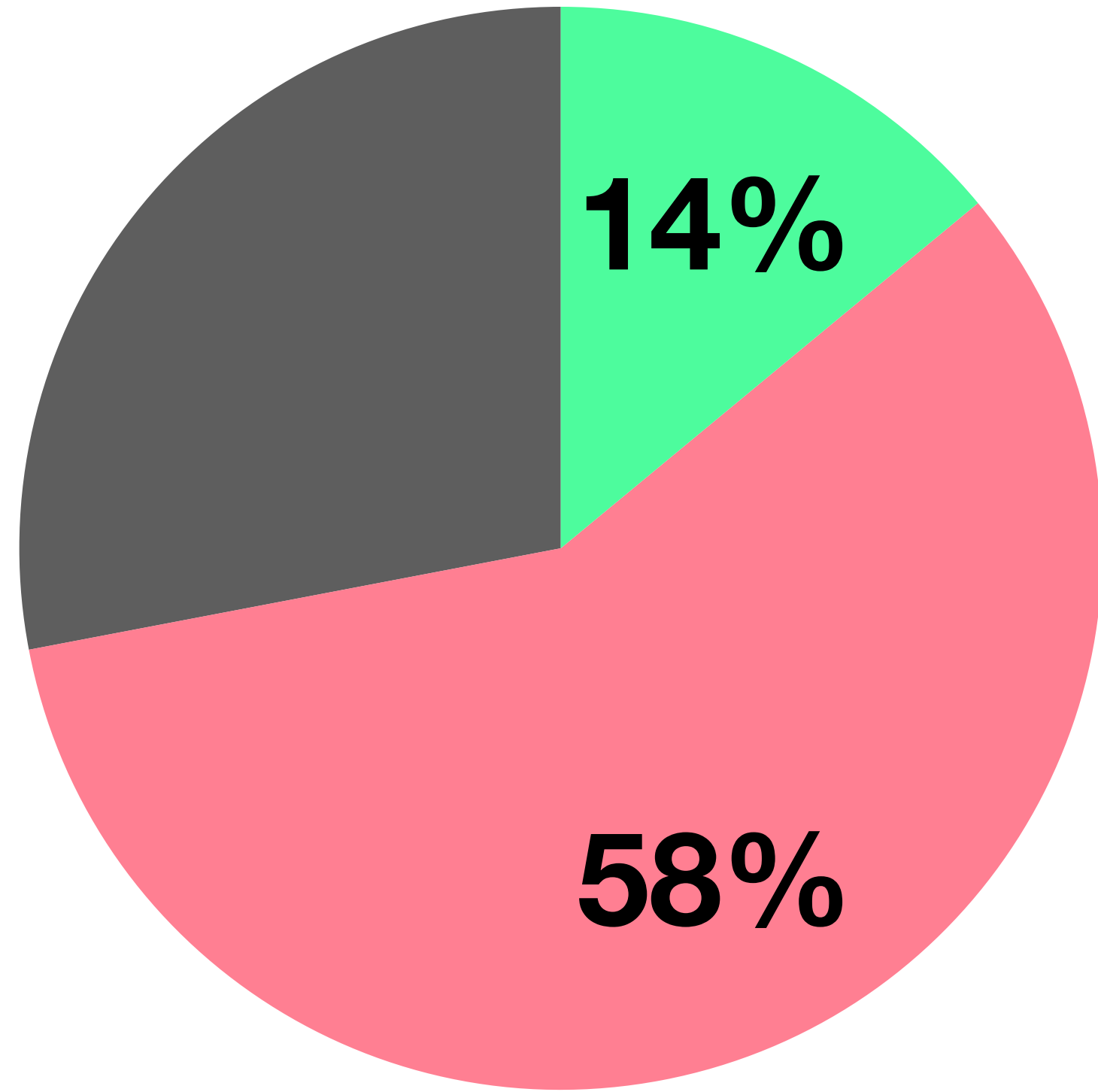
Good to see we're ranking #2 today on 'driving lessons'

Screenshot 2019-03-15 at 13.04.42.png ▼

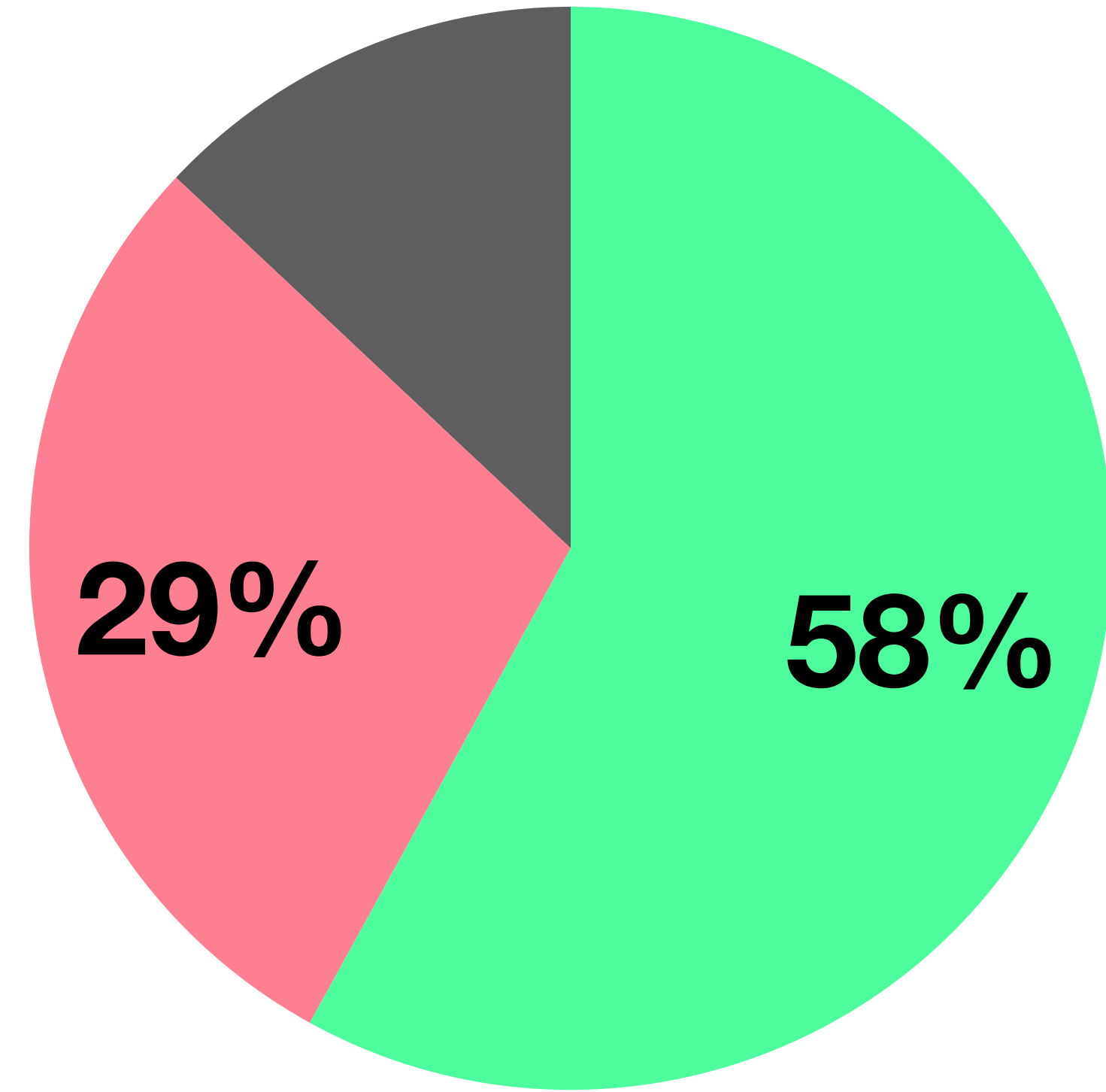


SEO Visibility

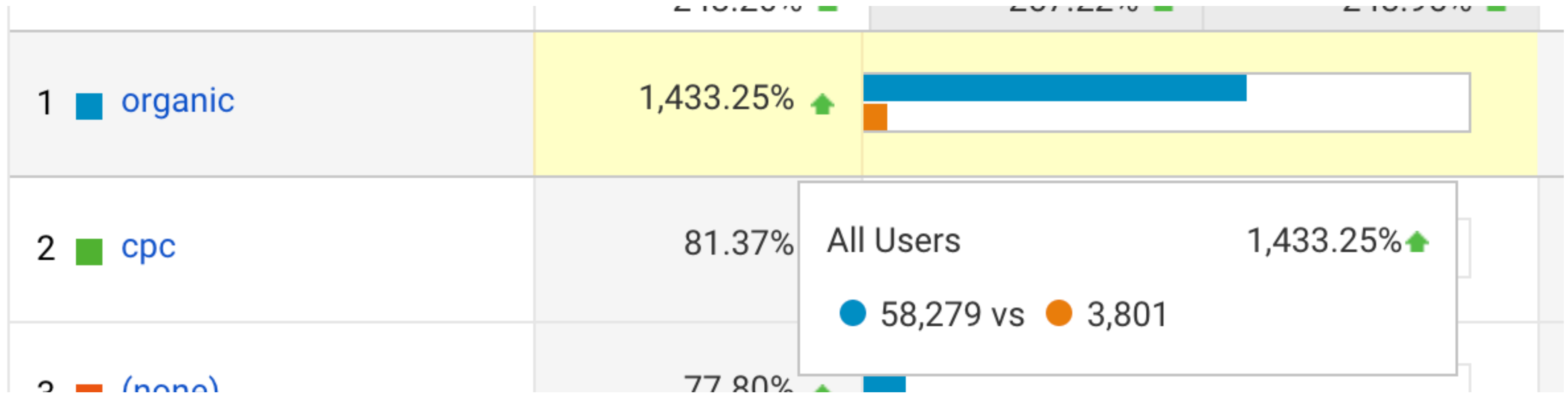
● Organic search ● Paid search ● Other



Before

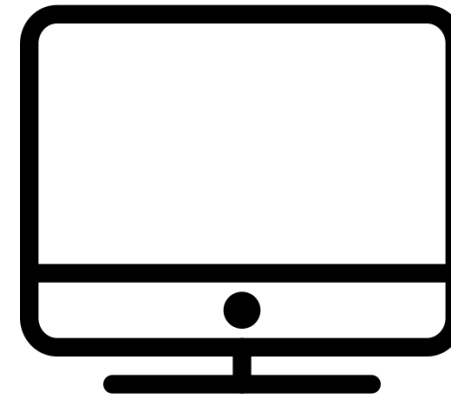


After



Before: 4K

After: 60K



WiFi

100

Performance

100

Accessibility

100

Best Practices

100

SEO

Score scale: ■ 90-100 ■ 50-89 ■ 0-49

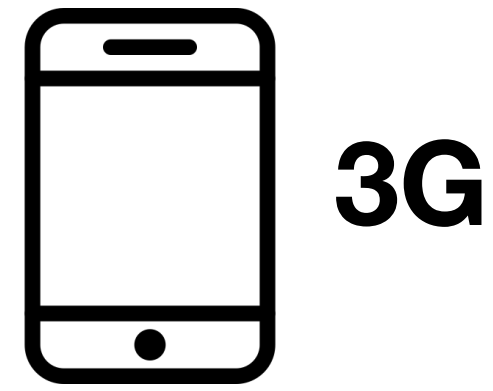
Performance

100

⌚ Metrics

First Contentful Paint	0.1 s ✓	First Meaningful Paint	0.2 s ✓
Speed Index	0.5 s ✓	First CPU Idle	0.9 s ✓
Time to Interactive	0.9 s ✓	Estimated Input Latency	40 ms ✓





Performance



Accessibility



Best Practices



SEO

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

Performance



Metrics

First Contentful Paint	0.9 s ✓	First Meaningful Paint	1.0 s ✓
Speed Index	2.4 s ✓	First CPU Idle	7.9 s ⚠
Time to Interactive	7.9 s ⚠	Estimated Input Latency	560 ms ⚠

[View Trace](#)



Values are estimated and may vary.



App

 index.js	39.47 KB
 _app.js	16.25 KB
 webpack-27d8b076b2d67411a805.js	1.46 KB
 commons.04e2f8c0b39a794128ad.js	158.49 KB
 main-f2689a6d59ae86fd6b17.js	40.22 KB

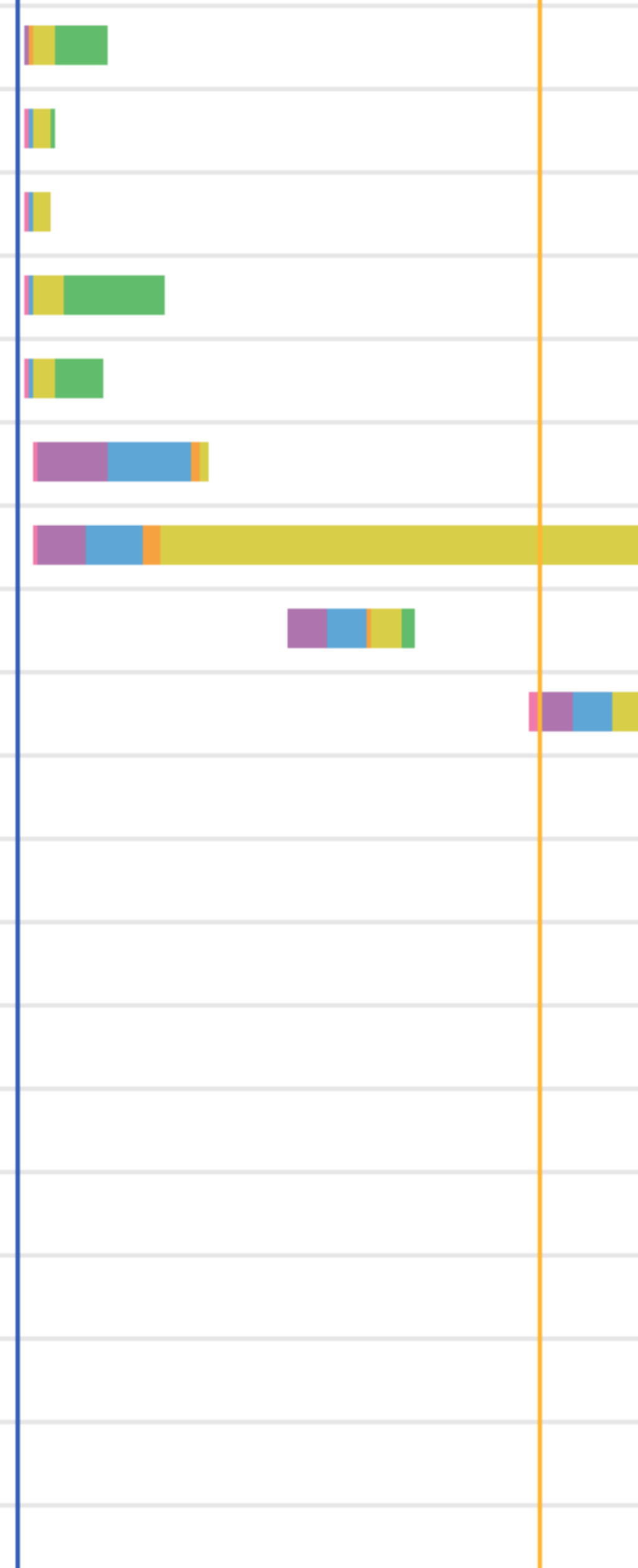
Analytics

 uno1kwh.js	7.51 KB
 analytics.min.js	62.59 KB
 snippet.js?key=99cd56ee-cfbc-4f6d-8a40-bf5522062c1f	1.20 KB
 get-loader.js?v=1&user_id=32188	7.39 KB

Prefetch

 page.js	39.97 KB
 packages.js	36.73 KB

 hotjar-521787.js?sv=5	2.52 KB
 gtm.js?id=GTM-MSZSB3P&l=dataLayer	25.99 KB
 analytics.js	17.42 KB
 bat.js	7.23 KB
 fbevents.js	15.84 KB
 conversion_async.js	9.06 KB
 mixpanel-2-latest.min.js	23.56 KB
 prum.min.js	3.04 KB





Lloyd Alex 12:14 PM

Holy wow! Great work @Nat

This time last year this was always 9-10 seconds! 😄

Pasted image at 2019-01-29, 12:13 PM ▾

https://midrive.com

2s

Loading time on 3G: **Excellent**

Low

Est. Visitor loss
(Due to loading time)

Looking good. Find out what went into your scores.



The screenshot shows a web browser window with the URL `spectrum.chat`. The page is the Next.js community chat interface. On the left, there is a sidebar with navigation icons and a community card for Next.js. The community card includes the Next.js logo, the text "Next.js", a description "A place to chat about Next.js and answer questions. For all other ZEIT products, check this out.", the website `nextjs.org`, 7,211 members, and 263 members online. A purple "Join community" button is at the bottom of the card. Below the card is a "Channels" section with links for "Chat", "# All channels", "# General", and "# Jobs". The main content area shows a list of chat messages. The first message is from Tim, pinned, titled "Companies / Sites using Next.js", listing ZEIT, Netflix, and Trulia. The second message is from David Calhoun, titled "Next.js 8 and RNW Transpilation Issues". The third message is from Santosh Kumar, titled "Expert's opinion required to sharing code between two Next.js projects". The fourth message is from Cyril, titled "Same module being included in multiple chunks".

<https://spectrum.chat/next-js>

Thanks!

 [@meuwka](https://twitter.com/meuwka)