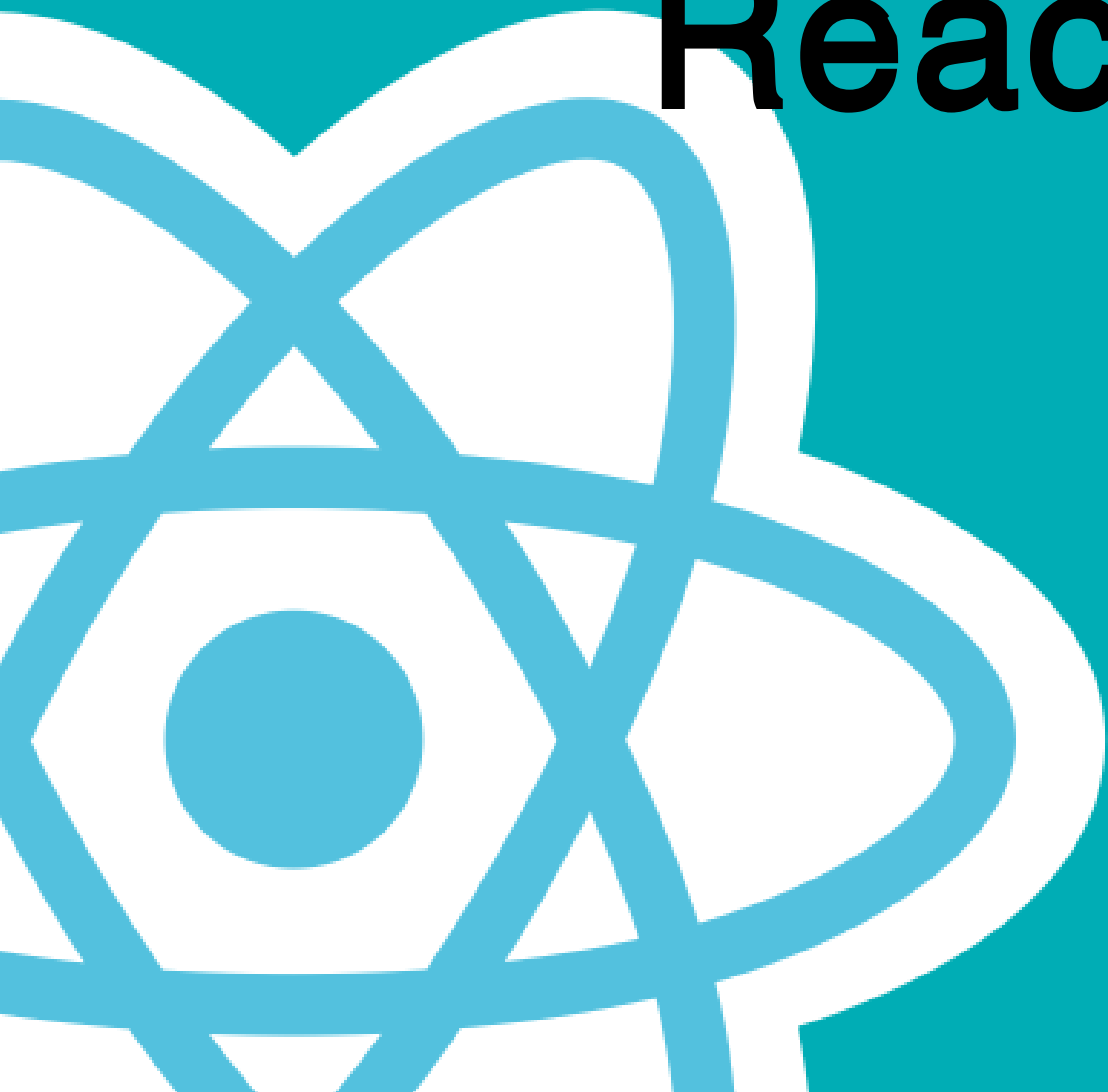


استراتژی‌های بهینه‌سازی

سئو در React



خلاصه ای کوتاه از رندر در React

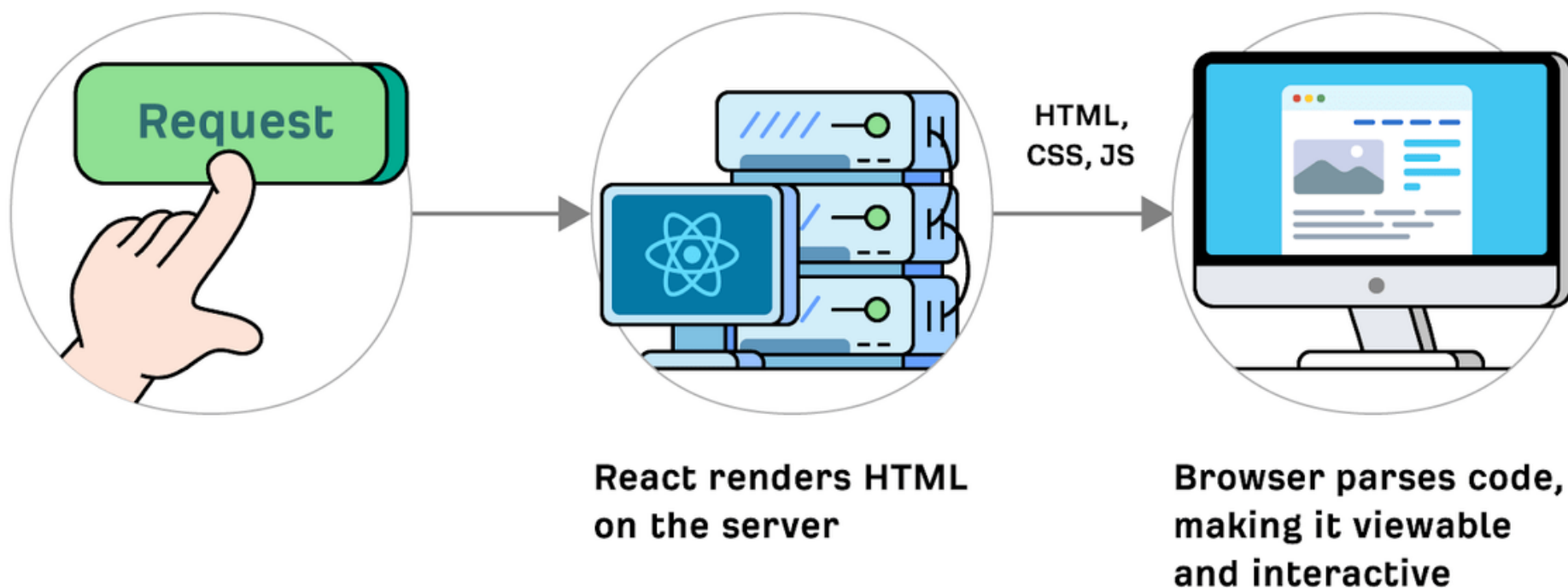
React یک App Shell Model را پیاده‌سازی می‌کند، به این معنی که اکثریت قریب به اتفاق محتوا، اگر نه همه، به طور پیش‌فرض رندر سمت مشتری (CSR) خواهند بود. CSR به این معنی است که HTML عمدتاً حاوی کتابخانه React JS است نه اینکه سروری که کل محتوای صفحه را در پاسخ HTTP اولیه از سرور (منبع HTML) ارسال کند. همچنین شامل جاوا اسکریپت متفرقه حاوی داده‌های JSON یا پیوندهایی به فایل‌های JS است که حاوی اجزای React هستند. با بررسی منبع HTML می‌توانید به سرعت متوجه شوید که یک سایت در سمت مشتری ارائه شده است.

اگر خطوط زیادی از HTML را در آنجا نمی‌بینید، برنامه احتمالاً رندر سمت مشتری است. با این حال، وقتی عنصر را با کلیک راست و انتخاب «Inspect element» (یا F12) می‌توانید بررسی کنید...

انواع رندر در React

وبسایت‌هایی که با استفاده از React ایجاد می‌شوند، با رویکرد سنتی‌تر برای گذاشتن کارهای سنگین رندر محتوا بر روی سرور با استفاده از زبان‌هایی مانند PHP که رندر سمت سرور (SSR) نامیده می‌شود، متفاوت است.

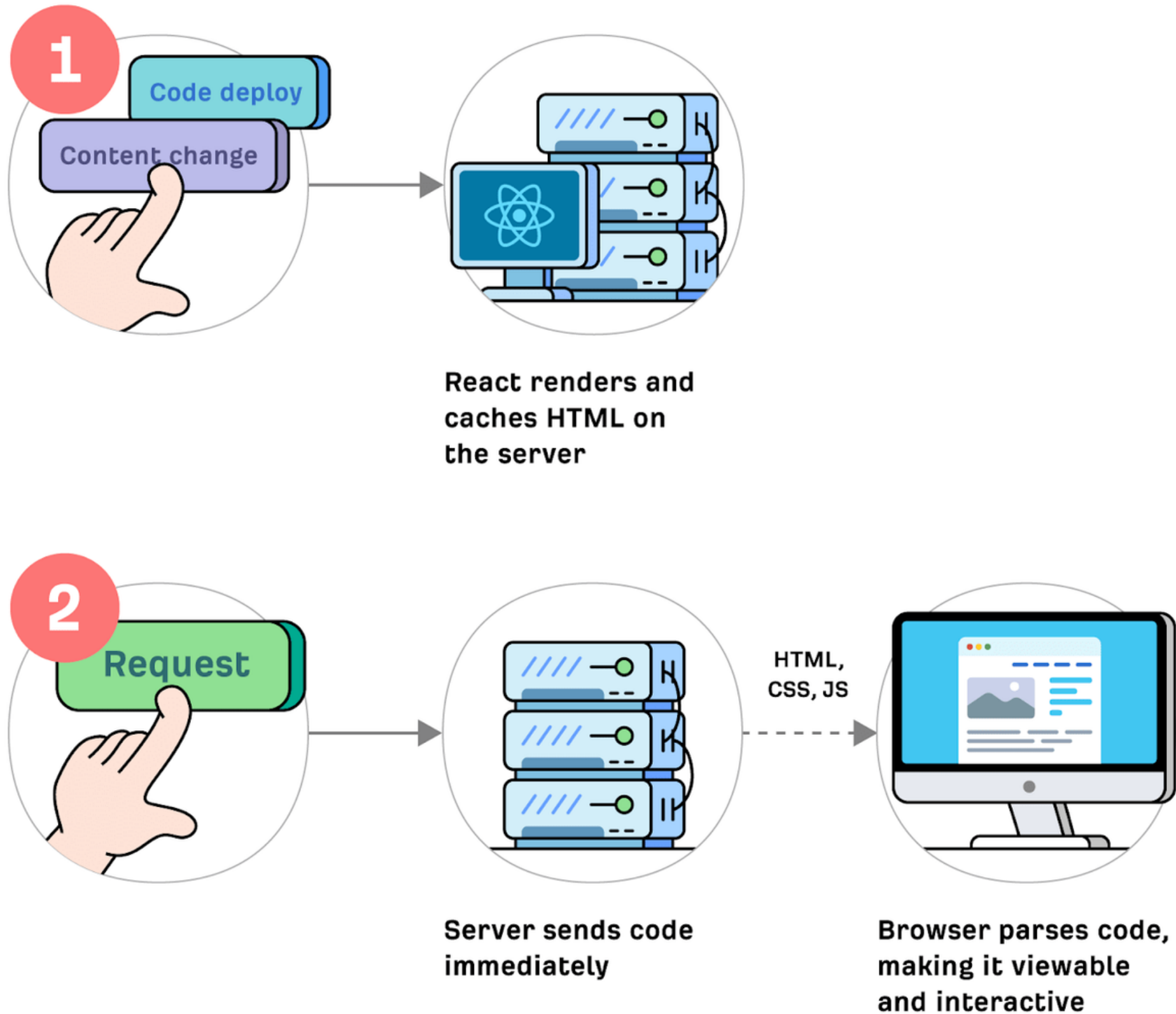
SSR



SSG

قبل از SSR، توسعه دهندگان آن را حتی ساده تر نگه داشتند. آنها اسناد HTML ایستا را ایجاد می کردند که تغییر نمی کرد، آنها را روی سرور میزبانی می کردند و بلافاصله آنها را ارسال می کردند. سرور نیازی به رندر کردن چیزی نداشت و مرورگر غالباً برای ارائه بسیار کمی داشت. SPA ها (از جمله آنهایی که از React استفاده می کنند) اکنون به این رویکرد ثابت باز می گردند. آنها اکنون قبل از اینکه مرورگر URL را درخواست کند، جاوا اسکریپت را از قبل به HTML ارائه می کنند. این رویکرد، تولید سایت استاتیک (SSG) نامیده می شود که به عنوان رندر استاتیک نیز شناخته می شود.

SSG



تفاوت SSR و SSG

در عمل، SSR و SSG مشابه هستند. تفاوت اصلی این است که رندر با SSR زمانی اتفاق می‌افتد که مرورگر در زمان ساخت با SSG یک URL را در مقابل یک چارچوب پیش‌فرم محتوایی درخواست می‌کند (زمانی که توسعه‌دهندگان کد جدید را مستقر می‌کنند یا یک سرپرست وب محتوای سایت را تغییر می‌دهد). SSR می‌تواند پویاتر باشد اما به دلیل تأخیر بیشتر، در حالی که سرور محتوا را قبل از ارسال آن به مرورگر کاربر، ارائه می‌کند. SSG سریع‌تر است، زیرا محتوا قبلاً ارائه شده است، به این معنی که می‌توان آن را فوراً به کاربر ارائه کرد (به معنی TTFB سریع‌تر).

چگونه گوگل صفحات را پردازش می کند؟

برای درک اینکه چرا رویکرد رندر سمت مشتری پیش فرض React باعث ایجاد مشکلات سئو می شود، ابتدا باید بدانید که Google چگونه صفحات را خزیده، پردازش می کند و فهرست بندی می کند.

ما می توانیم اصول اولیه نحوه عملکرد این کار را در مراحل زیر خلاصه کنیم:

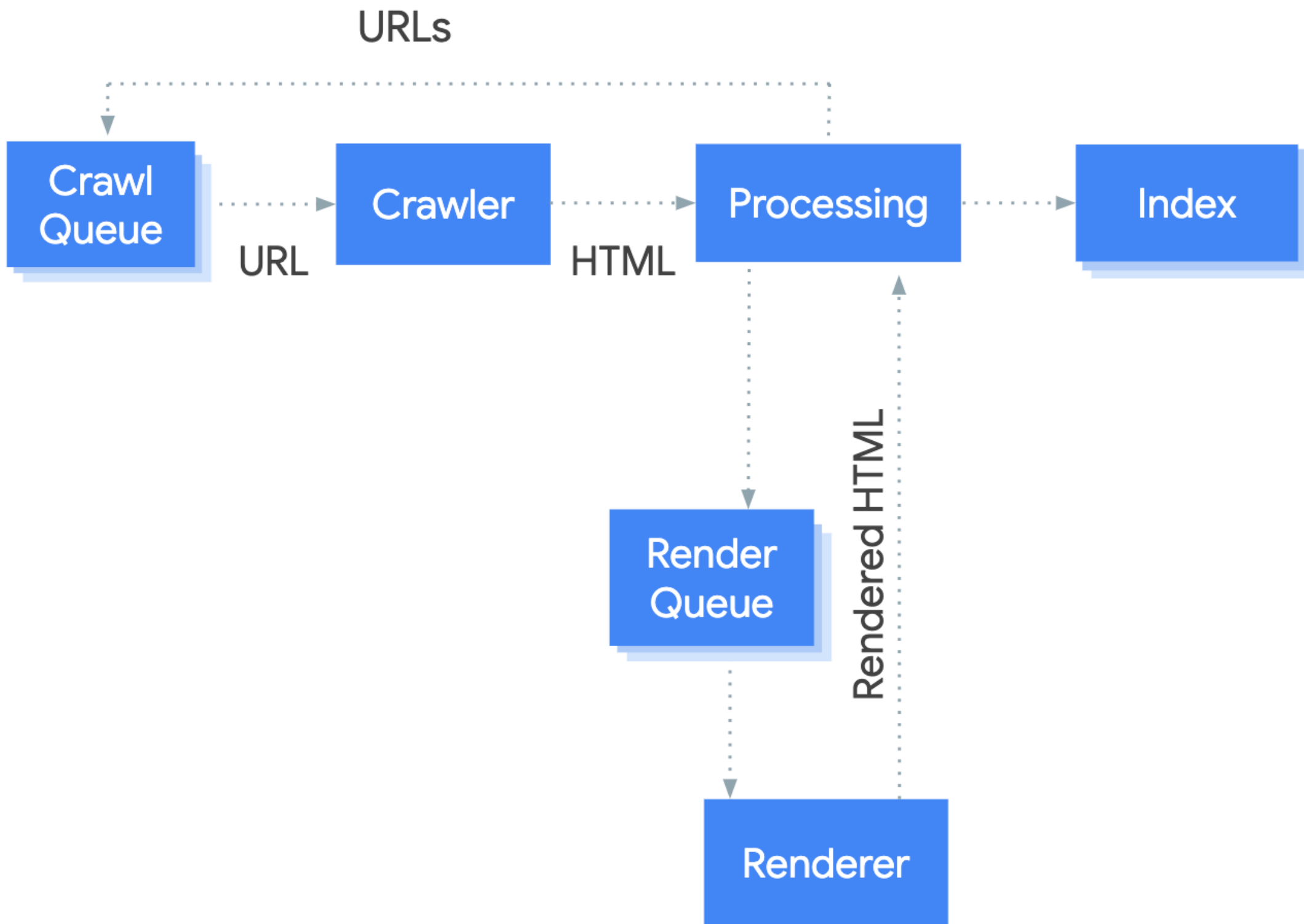
1. خزیدن – Googlebot درخواست های GET را برای نشانی های وب در صف خزیدن به سرور ارسال می کند و محتوای پاسخ را ذخیره می کند. Googlebot این کار را برای CSS، JS، HTML، فایل های تصویری و موارد دیگر انجام می دهد

2. پردازش – این شامل افزودن URL ها به صف خزیدن موجود در پیوندهای `<a href>` در HTML است. همچنین شامل آدرس‌های اینترنتی منابع صف (CSS/JS) است که در تگ‌های `<link>` یا تصاویر درون تگ‌های `` یافت می‌شوند. اگر Googlebot در این مرحله یک برچسب `noindex` پیدا کند، فرآیند متوقف می‌شود، Googlebot محتوا را ارائه نمی‌کند، و Caffeine (نمایه‌گر Google) آن را ایندکس نمی‌کند.

3. ارائه – Googlebot کد جاوا اسکریپت را با مرورگر Chromium بدون هد اجرا می‌کند تا محتوای اضافی را در DOM بیابد، اما منبع HTML را نه. این کار را برای همه URL های HTML انجام می‌دهد.

4. نمایه‌سازی – کافئین اطلاعات را از Googlebot می‌گیرد، آن‌ها را عادی می‌کند (HTML شکسته را اصلاح می‌کند)، و سپس سعی می‌کند همه آن را معنا کند، و برخی از سیگنال‌های رتبه‌بندی آماده برای ارائه در یک نتیجه جستجو را از قبل محاسبه می‌کند.

نحوه پردازش!



ضعف ها React در پردازش!

1. رندر نشدن جاوا اسکریپت – این یک مشکل قدیمی تر است، اما Google فقط در سال 2008 شروع به رندر کردن جاوا اسکریپت به صورت محدود کرد. با این حال، این هنوز متکی به یک طرح خزیدن برای سایت های جاوا اسکریپت بود که در سال 2009 ایجاد شد. (Google از زمانی که این طرح را منسوخ کرده است).

2. موتور رندر (Chromium) قدیمی است – این منجر به عدم پشتیبانی از آخرین مرورگر و ویژگی های جاوا اسکریپت شد. اگر از ویژگی جاوا اسکریپتی استفاده می کنید که Googlebot از آن پشتیبانی نمی کند، ممکن است صفحه شما به درستی نمایش داده نشود، که می تواند بر نمایه سازی محتوای شما تأثیر منفی بگذارد.

3. Google تأخیر رندر داشت - در برخی موارد، این می‌تواند به معنای تأخیر تا چند هفته باشد و زمان تغییرات در محتوا برای رسیدن به مرحله نمایه‌سازی را کاهش دهد. این امر تکیه بر Google برای ارائه محتوا برای اکثر سایت‌ها را رد می‌کند.

خوشبختانه، گوگل اکنون اکثر این مشکلات را حل کرده است. Googlebot اکنون همیشه سبز است، به این معنی که همیشه از آخرین ویژگی‌های Chromium پشتیبانی می‌کند.

مشکلات رایج سئو با React

در پنج سال گذشته، Google مدیریت خود را با محتوای جاوا اسکریپت نوآوری کرده است، اما سایت‌های کاملاً ارائه‌شده توسط مشتری مسائل دیگری را معرفی می‌کنند که باید در نظر بگیرید. توجه به این نکته مهم است که می‌توانید با React و SEO بر همه مشکلات غلبه کنید. React JS یک ابزار توسعه است. React با هیچ ابزار دیگری در پشته توسعه تفاوتی ندارد، خواه این افزونه WordPress باشد یا CDN که انتخاب می‌کنید. نحوه پیکربندی آن تعیین می‌کند که آیا سئو را کاهش می‌دهد یا افزایش می‌دهد. در نهایت، React برای SEO خوب است، زیرا تجربه کاربر را بهبود می‌بخشد. فقط باید مطمئن شوید که مسائل رایج زیر را در نظر می‌گیرید.


چطوری این مشکلات رو حل کنیم؟

1. استراتژی رندر مناسب را انتخاب کنید – مهم‌ترین مشکلی که باید با React مقابله کنید، نحوه ارائه محتوا است. همانطور که گفته شد، گوگل امروزه در ارائه جاوا اسکریپت عالی است. اما متأسفانه، این مورد در مورد سایر موتورهای جستجو صادق نیست. Bing تا حدودی از رندر جاوا اسکریپت پشتیبانی می‌کند، اگرچه کارایی آن ناشناخته است. سایر موتورهای جستجو مانند Baidu، Yandex، و دیگران پشتیبانی محدودی ارائه می‌دهند.

- زمان بارگذاری
- رندر سمت کلاینت باعث به‌روزرسانی تمام صفحه نمی‌شود، به این معنی که کتابخانه فقط یک بار نیاز به بارگیری دارد.
- الگوریتم «تفاوت» React فقط HTML را در DOM تغییر وضعیت می‌دهد و در نتیجه مرورگر فقط محتوایی را که تغییر کرده است دوباره ارائه می‌کند.

1. از کدهای وضعیت به درستی استفاده کنید – یک مشکل رایج در اکثر SPA ها این است که کدهای وضعیت را به درستی گزارش نمی دهند. این به این دلیل است که سرور صفحه را بارگیری نمی کند – مرورگر در حال بارگیری است. معمولاً مشکلاتی را در موارد زیر مشاهده خواهید کرد:

- هیچ تغییر مسیر 3XX وجود ندارد، به جای آن از جاوا اسکریپت تغییر مسیر استفاده می شود.
- کدهای وضعیت 4XX برای نشانی های اینترنتی «یافت نشد» گزارش نمی شوند.

Request URL	Status codes
 /this-should-404	200

- از React Router استفاده کنید
- صفحه 404 بسازید
- یک برچسب noindex به صفحات «یافت نشد» اضافه کنید
- یک `<h1>` با پیامی مانند "404: صفحه یافت نشد" اضافه کنید.
این ایده آل نیست، زیرا ما کد وضعیت 404 را گزارش نمی کنیم.
اما از ایندکس کردن صفحه توسط Google جلوگیری می کند و به آن کمک می کند صفحه را به عنوان یک 404 نرم افزار تشخیص دهد.
- زمانی که نیاز به تغییر URL دارید از تغییر مسیرهای جاوا اسکریپت استفاده کنید. باز هم ایده آل نیست، اما گوگل از تغییر مسیرهای جاوا اسکریپت پیروی می کند و سیگنال های رتبه بندی را پاس می کند.

2. از Hash کردن URL خودداری کن – این مشکل برای React معمول نیست، اما اجتناب از URL های هش مانند موارد زیر ضروری است:

- <https://reactspa.com/#/shop>
- <https://reactspa.com/#/about>
- <https://reactspa.com/#/contact>

به طور کلی، گوگل پس از هش چیزی را نخواهد دید. همه این صفحات به صورت https دیده می شوند:

راه حل

SPA ها با مسیریابی سمت مشتری باید History API را برای تغییر صفحات پیاده سازی کنند. می توانید این کار را با React Router و Next.js نسبتاً آسان انجام دهید.

2. در صورت لزوم از پیوندهای `<a href>` استفاده کنید – یک اشتباه رایج در SPA ها استفاده از `<div>` یا `<button>` برای تغییر URL است. این یک مشکل با خود React نیست، بلکه نحوه استفاده از کتابخانه است. انجام این کار باعث ایجاد مشکل در موتورهای جستجو می‌شود. همانطور که قبلاً ذکر شد، وقتی Google یک URL را پردازش می‌کند، به دنبال نشانی‌های اینترنتی اضافی برای خزیدن در عناصر `<a href>` می‌گردد. اگر عنصر `<a href>` وجود نداشته باشد، Google URL ها را نمی‌خزد و رتبه صفحه را پاس نمی‌کند.

راه حل

راه حل این است که پیوندهای `<a href>` به URL هایی را که می‌خواهید Google آنها را کشف کند، اضافه کنید. بررسی اینکه آیا به یک URL به درستی پیوند داده شده‌اید آسان است. عنصری را که به صورت داخلی پیوند می‌دهد بررسی کنید و HTML را بررسی کنید تا مطمئن شوید پیوندهای `<a href>` را درج کرده‌اید.

2. از بارگذاری کند HTML ضروری خودداری کنید - برای سایت‌هایی که با React توسعه یافته‌اند، وقتی کاربر روی یک عنصر کلیک می‌کند یا ماوس را نگه می‌دارد، معمولاً محتوا را به DOM تزریق می‌کنند - صرفاً به این دلیل که کتابخانه انجام این کار را آسان می‌کند. این ذاتاً بد نیست، اما محتوای اضافه شده به DOM از این طریق توسط موتورهای جستجو دیده نخواهد شد. اگر محتوای تزریق‌شده حاوی محتوای متنی مهم یا پیوندهای داخلی باشد، ممکن است تأثیر منفی داشته باشد:

- عملکرد صفحه بهتر است (چون گوگل محتوا را نمی‌بیند).
- قابلیت کشف سایر URL ها (زیرا گوگل لینک‌های داخلی را پیدا نمی‌کند).

راه حل

تشخیص این مسائل آسان نیست. و تا آنجا که من می دانم، هیچ ابزاری مستقیماً درباره آنها به شما نمی گوید. در عوض، باید عناصر مشترکی مانند موارد زیر را بررسی کنید:

- Accordions
- Modals
- Tabs
- Mega menus
- Hamburger menus

فرض کنید متوجه شده اید که جاوا اسکریپت HTML را به صفحه اضافه می کند. در این صورت، باید با توسعه دهندگان کار کنید. این به این صورت است که به جای تزریق محتوا به DOM، به طور پیش فرض در HTML گنجانده می شود و از طریق CSS با استفاده از ویژگی هایی مانند پنهان و نشان داده می شود.

پیش نیاز هارو فراموش نکن رفیق!

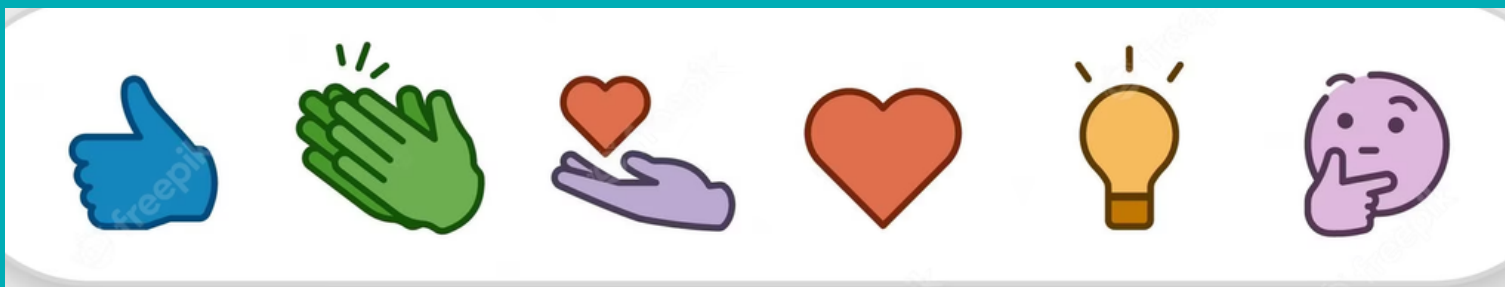
در حالی که ملاحظات سئو اضافی در برنامه های React وجود دارد، این بدان معنا نیست که سایر اصول اساسی اعمال نمی شوند. همچنان باید مطمئن شوید که برنامه های React شما از بهترین روش ها پیروی می کنند:

- Canonicalization
- Structured data
- XML sitemaps
- Mobile-first
- Website structure
- HTTPS
- Title tags
- Semantic HTML

جمع بندی و کلام آخر!

متأسفانه، کار با برنامه های React به لیست طولانی از مشکلاتی که یک سئو فنی باید بررسی کند اضافه می کند. اما به لطف چارچوب‌هایی مانند Next.js، کار یک سئو کار را بسیار ساده‌تر از آنچه در تاریخ بوده است، می‌کند. امیدوارم این پست به شما کمک کرده باشد تا ملاحظات دیگری را که باید به عنوان SEO هنگام کار با برنامه‌های React رعایت کنید، بهتر درک کنید.

ممنون که تا اینجا همراهی کردی رفیق
خوشحال میشم نظرت رو درباره پست
گامنت کنی 😊



از اینا یادت نره :