DevGeekWeek 2020
INSPIRED BY INNOVATION

React, Server Side Rendering and Next.js

JOHN BRYCE
תלמד הייטק. זה עובד!
a matrix company

About Me

- Yariv Katz - CEO, Nerdeez LTD
- FullStack expert, consulting and teaching
- My lectures are available on YOUTUBE
  - https://www.youtube.com/user/ywarezk

About this lecture

- All the code examples are available here:
  - https://github.com/ywarezk/react-dgw
- I'm assuming you know these react topics:
  - Function components
  - Hooks

We have these Goals from this lecture

- You should understand what is Server Side Rendering (SSR)
- Understand if you need it in your site
- Understand the development consequences of working with SSR
- Know how to easily implement SSR using React and Next.js

Comparing 2 identical sites

Comparing 2 sites

- both of the sites are created with React
- The code to create them is almost identical
- The site urls we will compare:
  - http://cdn.nerdeez.com/todo-dgw/index4.html
  - https://todo-ssr.vercel.app/tasks

- The sites we are comparing are very simple, in real life examples every result of the comparison is multiplied

How are we going to do the comparison?

- Loading Time
  - We will use chrome dev tools for this measurement
  - We will set the test on slow connection
  - We will examine the resources load of the two apps
  - We will examine the DomContentLoadad
  - We will examine using snapshots

We can significantly improve the initial load time of our React app

According to google statistics if the site takes more than 3 seconds to load we lose more than 50% of our users

How is SEO for single page application?

Post on the topic:

- https://rubygarage.org/blog/seo-for-react-websites
- tldr; not good
  - indexing on react app is slow
  - Errors
  - if the site is slow and starving the bot it won't all the site

SEO of react sites is not so good and can be improved

After this lecture, the load speed and SEO of your React site will greatly improve

Now you check!

How long did it take you React site to load?

Did you know…

A react component can run on a Node Server...

What will be the output of running a react component on Node ?
and what lifecycle and hooks will node run?

React components on Node

- React component can run on Node
- Node can produce an HTML string from a React component
- Node will only run the constructor (if not a function component) and the render (or the function if function component)
- Node will not run useEffect hook
- not will not run class lifecycle hooks (except constructor and render)

Let’s do another component in Node
TodoList that is grabbed from a server

TodoList Component

- This component will send a request to the following REST server
- https://nztodo.herokuapp.com/api/task/?format=json
- The component will display the items from the server in a list

TodoList Component

- How can we initiate the component with no useEffect and no lifecycle hooks on node?
- can we use the fetch we are using on the browser?
- If we load the server content on the server the loading will be faster and will be produced in the HTML
- if we load on the server maybe we can skip the loading in the client as well

From these ex. we can learn that our React app potentially can run on the browser and on Node

What will be the caveats of creating a React app that runs on both platforms?

How can running my app on Node can help me solve the initial problems we presented?

Slow initial load, bad SEO?

What is Server Side Rendering (SSR)?

SSR

- Node server will run our React app
- Server will produce html from our components
- Browser will load the html and display the screen while the js files load
- After the js files load they hydrate the screen and the app will turn to SPA

Do I need to worry about SSR on my react app?

- Are your users arriving from google? if the answer is YES then you need SSR
- Are you users bouncing from the side due to slow load (can be easily checked with google analytics)? if the answer is YES then you need SSR

Not all sites need SSR

- It’s important to note that not all sites need SSR
- Sites where the users are captives, like site for doctors to enter details about the patience, the doctors are captivated and will wait more if needed
- Note that SSR makes the development process harder.

How can I make the SSR development as easy as possible

What is Next.js?

What is Next.js?

- React Framework
- Provides extra tools for React to be super fast
- Easy SSR development
- Easy deployment
- Easy routing
- Very easy to learn

Our first Next.js app

create-next-app

- We can use **create-next-app** to bootstrap a new react and next project
- Let’s start a new next app and examine our project

Next Routing

Next Routing

- automatic routing based on the files in the **pages** directory
- you can use the hook useRouter
- you can use the **Link** component for navigation

SSR with Next.js

SSR with Next.js

- For each page there is 2 modes for SSR
    - Static
    - Server side rendering

- Prefer the first one
- For the second one Next.js can help us create a serverless deployment
- getStaticProps
- getStaticPaths
- getServerSideProps

Easy serverless deployment

Serverless / Static

- In the next ex we will create 2 kind of pages in our app
- a static server side page
- and a dynamic one
- we will deploy our app using vercel

Styling our app

Summary

- SSR is node ability to create html from our react app and then the client side hydrate that html and it turns back to SPA
- this cause a fast initial load and a better SEO
- Next.js is really easy to learn and can help us achieve SSR more easily

DevGeekWeek 2020
INSPIRED BY INNOVATION

Thanks :)

See you next year

JOHN BRYCE
תלמד הייטק. זה עובד!
a matrix company