migrate react application to next.js

http://nathankrasney.com/

motivation

Knowing next.js you will want to enjoy its features :

- Zero config page code splitting
- Zero config SSG by default
- Image component
- data fetching SSG / SSR / ISR
- ….

also for react project

But how to do this

http://nathankrasney.com/

react->next.js top level migration steps

Step 1 : Next.js of the box:

- Routing
- page code splitting
- ssg by default

Step 2 : Next.js specific features :

- Image
- SSG/SSR/ISR

Step 3 : Next.js further optimization

- Analize using core web vitals and lighthouse
- next.js Script component
- More here

react->next.js detailed migration steps 1/x

- Check performance with lighthouse in chrome dev tools mobile and desktop. this is not must but nice to compare before and after (google might use this for ranking your website)
- Create a new next.js project with typescript. Remove all css file and use only hello world in index.tsx
- Handle package.json :
  - carefully move dependencies from react project to next project
  - at this stage you can ignore react specific package like react-router and react-helmet and react project packages like vite
  - Accepting the next.js react and react-dom version is optional
  - npm install , npm run dev

react->next.js detailed migration steps 2/x

- Create _document.tsx according to this (you might need to recompile) and carefully copy index.html into it if required. You can use html element link and script (next.js Script is more powerful)
- CSS
  - Remove all css files from next.js styles directory
  - All css file must be css module files so copy form react project and change file name. you can put them in the styles folder or not
- Carefully copy public information (excluding project stuff like favicon.ico \ manifest.json) from react project to next project
- Carefully copy src information from react project to next project src excluding react project stuff e.g. index.tsx but including index.css. Take the content of index.css and put in globals.css (globals.css is imported by_app.tsx to be shared between pages)
  - Move pages components to pages directory
  - Rename the home component page file name to index.tsx

http://nathankrasney.com/

react->next.js detailed migration steps 3/x

- Go over ALL components in next.js (start with Home):
  - Remove react-router
  - Remove react-helmet and replace with next/head
  - Import correct css path probably using absolute import path
  - use styles of css modules
- Routing :
  - replace react-router code with next.js router file system semantics
  - Use next.js Link
  - Remove page code splitting if exist because it is done by next.js out of the box
- NavBar : share it by using _app.tsx
- Replace img with Image component

http://nathankrasney.com/

react->next.js detailed migration steps 4/x

- Use in this order ssg \ isr \ ssr \ csr per page
- Consider replacing your server with API route
- Check performance with lighthouse in chrome dev tools mobile and desktop. this is not must but nice to compare before and after

References

Migrating from Create React App - next.js docs

Converting Create-React-App to Next.js from Vercel - june 2020