

박민선 | 프론트엔드 개발자

Info.

Birth. 95.03.09 (만, 29)

Phone. 010.8810.6145

Mail. pminsun309@gmail.com

Channel.

Blog. <https://minsunblog.com/>

GitHub. <https://github.com/pminsun>

안녕하세요. 데이터를 효율적으로 처리하고 사용자 친화적인 UI/UX를 구현하는 프론트엔드 엔지니어 박민선입니다. React의 재 사용성을 고려한 컴포넌트 설계와 라이브러리 및 프레임워크의 안정성을 고려하여 개발합니다. Next.js의 서버사이드 렌더링(SSR)을 활용해 초기 로딩 속도를 개선하고 SEO 성능을 향상시키는 웹 구현을 실천하고 있습니다.

프로젝트 개요

회사 프로젝트

- 벨로가 사이클 자전거 친화 도시 1010 관제 시스템 WEB

개요 : 11개 지역 라이딩 데이터를 차트와 Mapbox로 직관적 시각화하여 사용자가 기간 별 검색과 지도 상에서 경로 확인 가능하며 관제 시스템 페이지를 개발

Tech Stack : Next.js, React.js, emotion, TypeScript

- Pandora App / Pandora Admin WEB

개요 : 판교 자율주행 대중교통 예약 서비스 앱에는 QR 티켓 발급, 노선도 제공, 티켓 조회와 같은 기능을 개발, 이용자 관리는 별도의 웹 기반 관리자 페이지에서 처리하여 개발

Tech Stack(App) : React-Native, styled-components

Tech Stack(WEB) : React.js, TypeScript, react-query, styled-components

- VelogaAuto Admin Page WEB

개요 : 기획서에 따른 자사 앱과 연동되어 관리자가 스킨, 배너 등록이 가능한 기능이 있는 관리자페이지를 구축

Tech Stack : ejs, JavaScript, Html, Css

사이드 프로젝트

- 개인 블로그 WEB (<https://minsunblog.com/>)

개요 : Notion API를 활용한 반응형 블로그 WEB

Github 링크 : <https://github.com/pminsun/MinsunBlog>

Tech Stack : Next.js, TypeScript, Tailwind, AWS EC2, Zustand

- 레시피 정보 검색, 저장 서비스 WEB (<https://www.myallfood.com/>)

개요 : Edamam API와 Firebase를 활용한 레시피 정보 검색, 저장 서비스 WEB

Github 링크 : <https://github.com/pminsun/AllFood>

Tech Stack : Next.js, JavaScript, React.js, Firebase, React-Query

기술 스택

Next.js, React, TypeScript, React-Query, Zustand, AWS, Emotion, Tailwind

경력

엘로나이프

팀원(퍼블 & 프론트 담당) | 연구개발센터 | 재직 중

2021.10.01 - 현재 (2년 8개월 2024.05.30기준)

모빌리티 소프트웨어 서비스 스타트업

학력

서울과학기술대학교

2015.03 - 2020.02 | 학사

프로젝트

[사이드] 개인 블로그 웹사이트

Notion API를 활용한 반응형 블로그 WEB

Tech Stack : Next.js, TypeScript, Tailwind, Zustand, AWS EC2, AWS S3

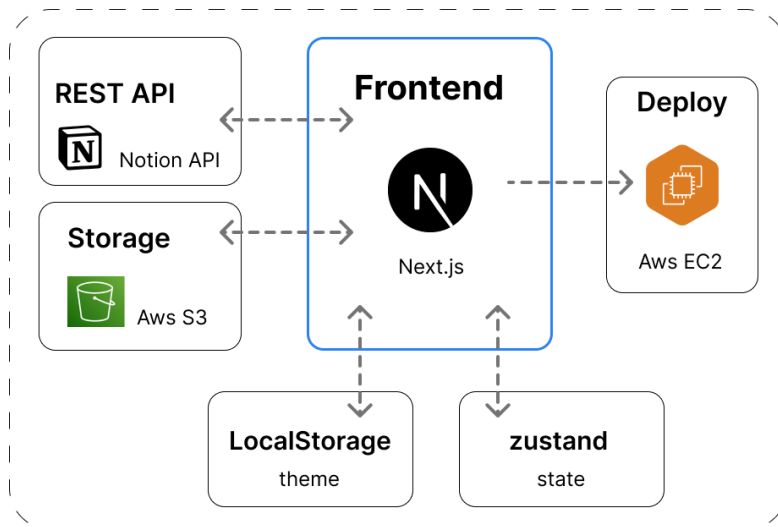
홈페이지 링크

<https://minsunblog.com/>

Github 링크

<https://github.com/pminsun/MinsunBlog>

아키텍처



프로젝트에서 배운 점

문제

1. 블로그는 사용자가 불특정 환경에서 보게 되는 유형으로 모든 디바이스에서 반응형으로 제작될 필요가 있다.
2. 포스팅 된 글에 대한 상세 페이지에서 보여줄 속성, 내용에 대한 API들에 대한 비동기 함수 병렬 실행 필요하다.
3. 초기 리스트 페이지 진입 시 데이터를 가져오는 비동기 처리 과정에서 이미지 로딩이 오래 걸려 이미지 영역의 빈 화면 보다는 해당 영역에 대해 스켈레톤 처리가 필요하다.
4. Blog, Project 페이지의 포스트 리스트에 대한 viewStyle, sort 선택한 값을 효율적으로 관리하기 어려우며 각 페이지마다 상태 변경이 중복되어 유지보수 및 확장이 어렵다.

해결

1. 반응형을 적용하기 위해 CSS의 flex, grid를 활용해 PC, 태블릿, 모바일에 대응되도록 변경하였다.
2. Promise.all() 사용한 병렬 처리로 순차적으로 요청을 보낸 경우보다 두 배 이상 빠르게 요청을 완료했다.
3. Next.js의 <Image /> 컴포넌트에서 blurDataURL 속성을 활용하여 base64로 인코딩된 회색 이미지를 사용하여 이미지 로딩 중 스켈레톤 처리를 구현했다.
4. 상태 관리를 단순화하고, 코드 중복을 줄이기 위해 Zustand 라이브러리를 도입했다. persist 미들웨어를 사용해 상태를 로컬 스토리지에 저장하여 설정이 유지되도록 변경했다.

[사이드] AllFood 웹사이트

레시피 정보 검색, 저장 하는 WEB

Tech Stack : Next.js, Javascript, React.js, firebase, react-query

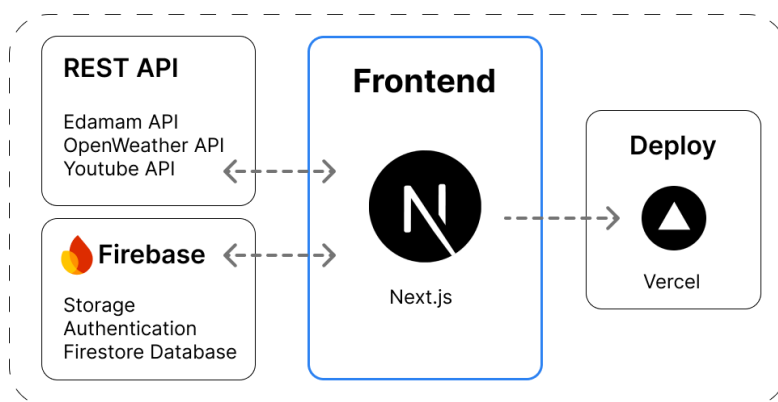
홈페이지 링크

<https://www.myallfood.com/>

Github 링크

<https://github.com/pminsun/AllFood>

아키텍처



프로젝트에서 배운 점

문제

1. 계정 별 레시피 관리가 필요하다.
2. 새로 고침 하면 Dynamic Routing으로 받은 값이 사라져 유지가 필요하다.
3. 같은 api를 사용하는 4개의 youtube 채널을 불러와야 된다.

해결

1. firebase의 auth를 활용해 계정 생성 시 database에 user.uid 명의 문서가 생성되어 하위에 레시피 정보를 저장할 수 있다.
2. getServerSideProps를 활용하여 query에 저장된 값을 서버 측에서 가져와 새로 고침 시에도 데이터를 유지할 수 있다.
3. useQueries 혹은 사용하여 여러 채널의 데이터를 병렬로 가져와 비동기 쿼리를 동시에 실행하며, 여러 쿼리를 하나의 배열로 관리하고 상태와 오류 처리를 일관성 있게 처리할 수 있다.