

오성은

Junior Front-end Developer

주소: 서울특별시 관악구 신림동 10-380

email: nnmmaz95@gmail.com

phone: 010-5688-0927

github: <https://github.com/ose1012>

Blog : <https://velog.io/@haesoo>

About me

웹 브라우저를 통해 유저가 직접 마주하는 부분이며 시각적으로 바로 확인이 가능한 것에 대해 매력을 느낀 저는 프론트엔드에 관한 기본 지식에 대해 공부하고 있습니다. 웹 접근성을 기초로 한 UI/UX를 공부하고 있으며 다양한 라이브러리를 활용하여 클라이언트가 요구하는 서비스를 만들기 위해 노력하고 있습니다. 또한 기초를 다지기 위해 방송 통신 대학교를 수료할 계획이 있습니다. 앞으로 프론트엔드 개발자로서 다양한 경험과 실무 프로젝트를 통해 하이브리드 앱에 대한 전반적인 이해도를 높이고 전문가가 되어 한 프로젝트의 PL이 되는 것이 현재 저의 목표입니다.

Skills

Language

기본적인 Native Javascript 언어 문법에 대한 지식이 있습니다.

Python을 사용해 크롤러를 작성할 수 있습니다.

java를 사용하여 CRUD를 구현할 수 있습니다.

간단한 데이터를 가져오는 쿼리(oracle)를 작성할 수 있습니다.

Framework / library

react : react를 메인으로 사용하여 프론트 인터페이스를 구축할 수 있습니다.

vue.js와 angular에 대해서 기초 수준으로 다룰 수 있습니다.

react의 life-cycle을 이해하여 api 호출에 대한 지식이 있고

기본적으로 사용되는 API를 활용할 수 있으며, 검색으로 시간을 들여서라도 어려운 문제를 해결할 수 있습니다.

flask 파이썬 백엔드 프레임워크를 사용해 본 경험이 있습니다.

Tools

vscode와 eclipse를 사용할 수 있습니다.

기본적인 터미널 사용 방법을 숙지하고 있습니다.

Git을 사용할 수 있고 컨플릭트를 해결해 본 경험이 있습니다.

Slack과 Notion을 사용해 본 경험이 있으며 figma를 사용할 수 있습니다.

Project

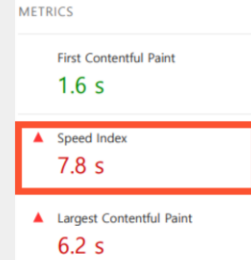
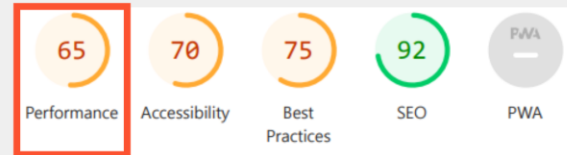
DO!BLOCK (2022.11 ~ 2022. 12) - <https://github.com/Hanghae99-DoBlock/FE>

- Do!Block은 사용자의 입장에서 “내가 한 일, 해야할 일을 TodoList로 관리하면서 동시에 공유할 수 있는 서비스가 있을까?” 하는 질문에서 시작한 프로젝트입니다.
- react + spring을 사용하였고 calendar 라이브러리를 사용하여 메인 화면에서 날짜를 선택할 수 있게 도와줍니다. 또한 drag & drop 라이브러리를 사용하여 자유롭게 Todo를 이동할 수 있게 구현하였고 블록(피드) 형태의 SNS 부분에서 본인이 팔로우하고 있는 사람과 추천 피드를 나누어 볼 수 있도록 구현하였습니다. 배지 기능을 도입하여 해당 서비스에 더욱 재미를 주기 위해 사람들이 일정한 목표를 달성 할 때마다 배지를 획득할 수 있는 서비스도 구현하였습니다.
- 서버는 AWS EC2 인스턴스를 사용하여 유저 수에 맞춰서 유저가 많아지면 서버가 원활하게 돌아가도록 유지하고, 유저가 적을 때는 최소한의 비용으로 서버가 유지되게 만들어 비용 관리적 측면과 서버 관리적 측면 양쪽을 둘 다 잡았습니다. 또한 인스턴스에서는 Docker를 사용하여 개발자 간의 환경 배포를 용이하게 만들었습니다.

- 사용중인 오프스크린 이미지 용량이 크고 사용자가 첫 페이지를 로드하는 즉시 대규모 단일 JavaScript 번들이 전송되어 해당 이슈를 해결하기 위해 WebP로 이미지 변환 및 지연로딩(Lazy-loading)을 통해 바로 로딩하지 않고 보이지 않는 페이지만 로드한 후, 다른 페이지에 접속했을 때 그곳의 데이터를 로드해오는 작업을 진행하여 웹 페이지 성능을 최적화 하였습니다.

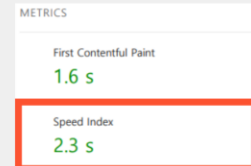
문제 상황

웹페이지 성능 최적화가 이루어지지 않아 로딩이 길어짐
사용 중인 오프스크린 이미지 용량이 큼
첫 페이지 로드 시 대규모 JavaScript 번들이 전송됨



개선 결과

WebP로 이미지 변환하여 오프스크린 이미지 용량 감소
Lazy-Loading 을 통해 보이는 페이지만 로드함



239% 개선