

손영진

프론트엔드 개발자(신입)



✉️ ods04139@gmail.com

📞 010-3975-5037

🔗 <https://sonyoungjin.netlify.app/>

💻 github.com/sonprogrammer

GRADUATED

한서대학교

- 항공소프트웨어 공학과

SKILLS

Front-end

- javascript
- typescript
- react
- next.js
- html
- css

Back-end

- node.js
- mongoDB
- Express.js

etc

- git
- notion
- figma

CERTIFICATES

- 정보처리기사

- GTQ 1급

- 토익 835점

PROFILE

저는 일상생활에 가치를 얹어주고 싶은 일을 하고 싶습니다.

다양한 경험과 전공을 통해 가능성을 발견하면서 프론트엔드 분야에 집중적인 관심을 가지게 되었고, 단순히 '보이는 것'을 만드는 것을 넘어 사용자의 행동과 경험에 직접적인 영향을 줄 수 있다는 점에서 큰 매력을 느꼈습니다.

국비지원 교육을 수료하며 부족했던 부분을 보완했고, 일상에서 느꼈던 불편함을 바탕으로 여러 개인 프로젝트를 통해 아이디어를 구현해왔습니다.

앞으로는 사용자에게 더 나은 경험을 제공하는 서비스를 만들고 싶습니다.

또한, 일상에 긍정적인 변화를 주는 개발자가 되고 싶습니다.

EDUCATION

엘리스

웹 개발자 트랙 6기

• 협업경험

- Git Flow기반 브랜치 전략과 코드 리뷰를 적용하여 이슈별 기능 개발
- 코딩 컨벤션 준수를 통해 코드 품질 유지
- 애자일 개발 프로세스를 적용하여 주기적 스프린트 및 효율 협업 경험
- 프론트엔드 & 백엔드 개발 경험
- 필요한 역량을 갖추고, 지속해서 새로운 기술과 트렌드에 적극적으로 도전하며 성장하고 있습니다

PROJECTS

VC(Virtual Coin) - 개인 프로젝트

실시간 모의 코인 투자 서비스

기술스택: React, TypeScript, Node.js, MongoDB, React Query, Styled-components, Tailwind CSS, Chart.js, Upbit WebSocket

주요 구현 사항

- Figma를 활용한 전체 서비스 UI/UX 설계 및 와이어프레임 제작
- 하이브리드 시세 수신 전략 설계 : 초기 진입시 수백개의 코인 실시간 데이터를 WebSocket으로 받으면서 발생하는 지연 시간 문제를 UX의 치명적 결함으로 정의
• REST API로 현재가를 즉시 호출하여 화면을 우선 렌더링하고 백그라운드에서 WebSocket을 동기화하는 하이브리드 방식을 구현하여 초기 로딩 속도를 약 80% 개선
- Upbit WebSocket을 활용한 실시간 코인 시세 데이터 수신 및 업데이트 기능 구현
- 서버 상태 관리 및 UX 고도화
• React Query와 커스텀 헤더 도입으로 서버 데이터 캐싱 및 상태 관리 최적화
- Chart.js를 활용한 코인 가격 차트 시각화 구현
- 안정적인 백엔드 및 보안 설계
• Node.js와 MongoDB기반의 REST API 구축 및 미들웨어를 통한 사용자 권한 검증 및 접근 제어 시스템 구축
- Styled-components와 Tailwind CSS를 조합한 컴포넌트 스타일링 및 PWA 적용으로 크로스 플랫폼 사용자 경험 향상
- Skeleton UI 및 로딩 상태 관리를 통한 사용자 경험 개선

BNTY(Be Next To You) - 개인 프로젝트

트레이너와 회원간 소통 플랫폼

기술스택: React, TypeScript, JavaScript, Node.js, MongoDB, WebSocket, Styled-components, Tailwind CSS, PWA

주요 구현 사항

- 경험 기반의 문제 해결 및 서비스 설계
 - 트레이너 근무 시절 경험한 수업 횟수 관리의 불투명성과 개인정보 관리의 허점을 발견하여 서비스 기획부터 DB설계 진행
 - QR코드 출석 체크 시스템을 도입하여 불필요한 행정 절차 간소화하고 사용 편의성 극대화
- 백엔드 아키텍처
 - Node.js와 MongoDB 기반 MVC 패턴 REST API 설계 및 CRUD 기능 구현

- 실시간 소통을 위한 데이터 동기화
 - WebSocket을 활용한 실시간 채팅 시스템을 구축하고, 편의를 위해 읽음/안읽음 상태 관리 시스템 개발
- 사용자 인터페이스 및 경험 최적화
 - Styled-components와 Tailwind CSS를 활용한 반응형 UI 구현 및 PWA 적용으로 모바일 최적화
 - 자동 로그인 기능 및 Skeleton UI 적용으로 사용자 경험 개선

Fuelly - 개인 프로젝트

AI기반 개인 맞춤 영양 관리 웹 서비스

기술스택: Next.js, TypeScript, Zustand, React Query, Tailwind CSS, Material UI, Recharts, Grok AI API, Lucide React

개요

- AI를 활용하여 사용자 목표 기반 일일 권장 칼로리 및 단백질 섭취량을 제공하는 개인 맞춤 영양 관리 서비스

주요 구현 사항

- AI 연동 및 비정형 데이터 구조화
 - Grok AI API를 연동하여 사용자 설문 데이터 기반의 맞춤형 음식 추천 및 영양 상담 기능 구현
 - AI의 비정형 응답을 JSON 형태로 파싱하여 시스템 데이터와 동기화함으로써 분석 결과를 식단 관리 기능에 즉각적으로 반영하는 파이프라인 구축
- 보안 중심의 인증 아키텍처 설계
 - jose 라이브러리를 활용하여 Access Token(메모리 관리)과 Refresh Token(HttpOnly Cookie)의 이중 보안 인증 시스템 구축
 - Next.js 미들웨어를 활용한 라우트 보호 및 토큰 재발급 로직을 설계하여 서비스 전반의 보안성 및 안정성 확보
- 효율적인 상태관리 및 네트워크 계층 최적화
 - Zustand와 React Query를 활용한 클라이언트 및 서버 상태 관리 구조화
 - Axios 커스텀 인스턴스와 인터셉터를 통한 인증 및 토큰 재발급 로직 중앙 집중화
- 데이터 시각화 및 UI/UX
 - Recharts를 활용한 영양 섭취 현황 시각화 및 Tailwind CSS, Material UI 기반 반응형 UI 구현

Mingle - 팀프로젝트

플레이리스트 공유 SNS 서비스

기술스택: React, TypeScript, Recoil, React Query, Tailwind CSS, Twin.macro, Axios, OpenWeatherMap API

역할

- 프론트엔드 개발 담당
 - React 기반 재사용 가능한 컴포넌트 설계 및 구현
 - REST API를 통한 백엔드 연동 및 데이터 흐름 구축
 - 사용자 입력 처리 및 서버 응답 기반 동적 UI 업데이트 기능 구현
 - 상태 관리 라이브러리를 활용한 클라이언트 상태 관리 및 API 요청 처리