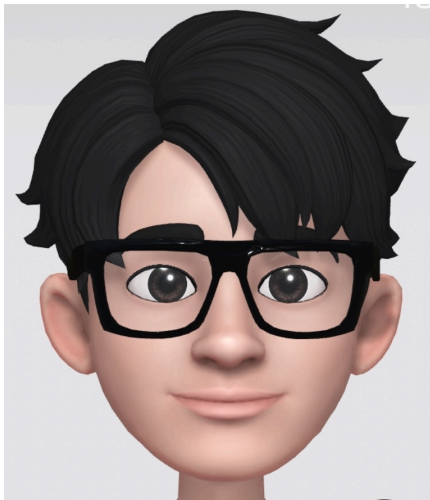


윤수호 | 간단명료함을 지향하는 개발자



Email

7552678@naver.com



phone

010-4953-2679



블로그



기록창

- ✓ 문제 해결력이 뛰어난 프론트엔드 개발자를 꿈꿉니다.
- ✓ 기술적인 주제에 대해 글 쓰는 것을 좋아하고 잘 합니다.
- ✓ 간단한 용어로 설명해 사람들이 쉽게 이해하도록 하는 것에 보람을 느낍니다.



경력

어반엘티 (2023.02.01 ~ 2024.11.30)



Projects

사내 자체 프로젝트 아파트모아 진행

구현한 부분 :

- **HOC를 활용한 각 페이지의 로그인 검사 기능 추가**

- 도입배경 : 각페이지 접근시 accessToken 여부 확인해야 되는 절차 발생
- 사용이유 : HOC (Higher Order Component)는 컴포넌트를 가져와 새로운 컴포넌트를 반환하는 함수로 토큰 여부 확인 및 재발급과 같은 반복적인 코드가 컴포넌트에 필요하다 판단해 도입하게 되었습니다.
- 성과 : HOC를 활용한 accessToken 여부 및 미확인시 accessToken 재발급 코드 작성 후 재발급 실패시 로그인 페이지로 redirect 하여 코드의 가독성을 높였습니다.

- **초기 렌더링 향상을 위한 code split 진행**

- 도입배경 : 프로젝트 규모가 커지면서 초기 렌더링 속도가 느려져 사용자 경험이 저하됨을 확인 했습니다.
- 사용이유 : 렌더링 최적화를 시도했지만 효과가 미미하여, 초기 렌더링 속도를 개선할 방법을 찾던 중 Lazy Loading을 활용한 code split이 불필요한 컴포넌트 렌더링을 방지하고 현재 페이지에 필요한 컴포넌트만 로드할 수 있다는 점을 확인 하였습니다.
- 성과 : 불필요한 초기 렌더링을 방지하기 위해 Lazy Loading을 적용하여 현재 페이지에서 필요한 컴포넌트만 로드 → 초기 번들 크기 **20% 감소**

- **모듈화를 이용한 전역상태관리 렌더링 문제해결**

- 도입배경 : 컴포넌트 리렌더링 시 불필요한 리렌더링이 과도하게 발생하는 문제 발견 했습니다.
- 사용이유 : 전역 상태 관리 객체의 크기가 커짐에 따라, 특정 값이 변경될 때 모든 관련 컴포넌트가 리렌더링되는 현상을 확인. 또한, 각 컴포넌트에서 store를 사용할 때 상위 객체를 직접 불러와 사용하고 있는 문제점 발견 했습니다.
- 성과 : 전역 상태 객체를 모듈화하고 더 작은 단위의 객체로 분리. 또한, 각 컴포넌트에서 store 사용 시 필요한 범위만 참조하도록 변경하여 리렌더링 횟수를 **최대 8회 → 2회로 감소**

- **컴포넌트 재사용을 위한 커스텀 훅 생성**

- 도입배경 : UI 컴포넌트와 비즈니스 로직이 혼재되어 유지보수성이 저하되고, 반복적인 상태 관리 코드가 다수 발생하는 문제를 확인
- 사용이유 : 코드의 일관성을 유지하고 가독성을 향상시키며, UI와 비즈니스 로직을 명확히 분리하여 재사용성을 극대화하기 위함
- 성과 : 기존에 서버와 연동 중인 비즈니스 로직을 커스텀 훅으로 변환하여, UI 컴포넌트는 렌더링에 집중하고 로직은 별도의 훅에서 관리하도록 구조를 개선하여 비즈니스 로직과 ui로직을 분리해 코드 가독성 증가

사용기술 및 언어 :

js, react, react-query, css-module, redux-toolkit, 기타등등 라이브러리(ex: ck editier, axios..)

GTM 사이트(점주,프로,매니저의 레슨 및 근무시간 등록 사이트) 유지보수 및 신규개발

구현한 부분 :

- **vuetyfy를 활용한 drag and drop을 활용한 레슨 시간 변경**
 - 도입배경 : 현재 등록된 레슨의 이동을 쉽게 변경할 수 있는 기능을 원하는 점주의 요청 발생
 - 사용이유 : 기존의 레슨 변경은 날짜를 drag and drop이 아닌 삭제 후 재등록을 하여 레슨을 변경, 하지만 사용중인 라이브러리(vuetyfy)에 스케줄기능중 drag and drop 기능이 있는것을 발견하여 의견 제시 후 도입 결정
 - 성과 : 기존의 레슨 변경 기능이 아닌 drag and drop을 도입해 마우스로 drag 후 drop하면 레슨 날짜가 변경되는 기능 추가
- **근무시간 등록과 레슨등록 페이지의 일원화 진행 및 주 단위 레슨 등록 기능 추가**
 - 도입배경 : 기존의 강사의 근무시간 등록 페이지와 레슨등록 페이지가 나누어져 있어 불편하다는 의견 발생
 - 사용이유 : 근무시간과 레슨등록 페이지가 굳이 2개일 필요 없이 하나로 통일하고 레슨등록을 모달로 진행하는 의견 제시
 - 성과 : 기존의 레슨등록 페이지를 제거 후 근무시간 페이지에 레슨 등록 버튼 생성 후 버튼 클릭시 레슨 등록 모달 생성 및 일주일 단위로 레슨을 등록할 수 있는 기능 추가
- **오퍼레이터 와 GTM사이트 공지사항 기능 추가**
 - 도입배경 : 기존 오퍼레이터 홈페이지에 공지사항을 등록하길 원하는 의견 발생
 - 성과 : 오퍼레이터 홈페이지에 에디터를 활용하여 공지사항을 등록하고 오퍼레이터에서 공지사항 등록시 관리자 페이지에서 공지사항을 일주일간 볼 수 있으며 "다시 보지 않기" 버튼 클릭시 local storage에 버튼 클릭 날짜를 저장하여 해당 공지사항이 일주일간 보이지 않는 기능 추가

사용 기술 및 언어 :

vue, vuex, typescript, rxjs, bootstrap, vuetyfy



side Projects

EMS(기기 사용 등록 사이트)

url : <https://ems-dev.vercel.app/>

id : admin

pw : 1

구현한 부분 :

- **미들웨어를 활용한 acces,refresh Token 관리 진행**

- 도입배경 : 토큰이 필요한 페이지 이동시 해당 토큰이 유효한지 또는 존재하는지 확인하는 기능 필요
- 사용이유 : 미들웨어는 페이지가 렌더링 되기전에 실행되는 함수이기 때문에 토큰을 검사하게 적합하다는 판단을 내렸습니다.
- 성과 : 미들웨어에서 accessToken 여부를 확인후 없을 경우 refreshToken을 사용해 재발급을 받습니다. 만약 refresh 토큰이 없을 경우 로그인페이지로 redirect합니다.

- **fetch를 활용한 api 함수 생성**

- 도입배경 : nextjs는 fetch의 함수에서 여러기능이 추가되어 있는 상태인데 예로 cache를 사용해 서버측에 해당 데이터를 일정시간 유지해 사용할 수 있는 기능이 있습니다. 하지만 axios의 여러 편의기능(예:interceptor)을 사용할 수 없기 때문에 직접 만들기로 하였습니다.
- 사용이유 : nextjs에서는 interceptor같은 기능을 사용하기 위해 공용함수를 만들지 않을 경우 fetch를 사용할때마다 interceptor에 사용할 기능을 적어주어야 되기 때문에 이를 공용함수로 만들기로 결정하였습니다.
- 성과 : nextjs는 서버에서 동작할때와 클라이언트에서 동작할 경우가 있기때문에 이를 window객체의 여부로 판단하여 window 객체가 있을 경우 클라이언트 없을 경우 서버로 판단하였습니다. 이에 따라 여러 기능을 추가하였는데 baseUrl, Authorization, 401에러 발생시 토큰 재발급 기능을 추가하여 하나의 api함수로 여러 편의 기능을 추가하여 코드의 가독성과 유지보수성을 높였습니다.

사용 기술 및 언어 :

nextjs14 app router, tailwindcss, shadcn, tanstack-query, zustand, typescript

여정(여행 동행자 모집 웹사이트)

구현한 부분 :

- **Stompjs, Sockjs를 활용할 실시간 댓글 알림 기능 추가**
 - **도입배경** : 사용자 게시물에 댓글이 달릴 경우, 실시간으로 알림을 받을 수 있도록 WebSocket 기반의 알림 시스템을 도입하였습니다.
 - **사용이유** : WebSocket API만 사용할 경우 직접 메시지 라우팅 및 연결 복구를 구현해야 하는 부담이 있어, 이를 간소화하기 위해 STOMP.js를 도입하였습니다. 또한 STOMP.js는 특정 주제를 구독한 클라이언트에게 자동으로 메시지를 전달할 수 있기 때문에 코드가 간결해지는 효과를 기대할 수 있습니다.
- **AWS CloudFront, GitHub Actions을 이용한 ci/cd 진행**
 - **도입배경** : 협업 중인 디자이너 및 백엔드 개발자와 원활한 소통을 위해, 배포된 화면을 보며 회의를 진행할 필요가 있었습니다. 이를 위해 AWS 기반의 CI/CD 환경을 구축하게 되었습니다.
 - **사용이유** : 작업 중 배포가 필요한 상황이 잦았고, 수동 배포로 인해 회의가 중단되는 문제가 발생했습니다. 이를 개선하고자 자동화된 배포 프로세스를 도입했습니다.
 - **해결 방법** : AWS CloudFront와 GitHub Actions을 활용하여 CI/CD 파이프라인을 구축하고, 배포를 자동화하여 원활한 협업 환경을 조성했습니다

사용 기술 및 언어 :

React, TypeScript, TanStack Query, Tailwind CSS

API 문서화 관리 시스템

주요 기능

1. 사용자 인증

- JWT 기반 인증 시스템
- 비밀번호 암호화 (bcrypt)
- 이메일 중복 확인

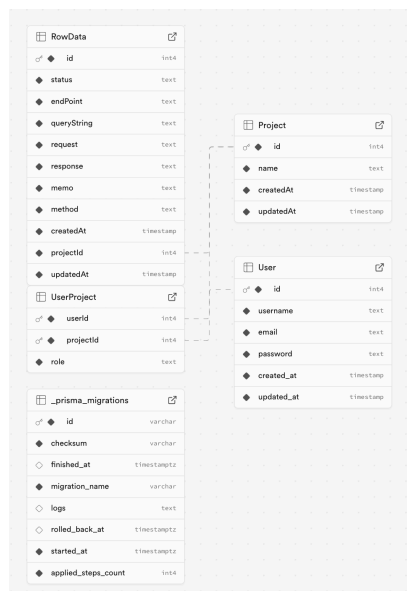
2. 프로젝트 관리

- 프로젝트 생성/수정/삭제
- 프로젝트별 API 목록 관리
- 실시간 상태 업데이트

3. API 문서화

- RESTful API 메소드 지원 (GET, POST, PUT, PATCH, DELETE)
- JSON 요청/응답 포매팅
- API 상태 추적 (시작전, 진행중, 수정완료, 최종완료)

ERD



사용 기술 및 언어 :

Next.js 14, React, TypeScript, TailwindCSS ,PostgreSQL, Prisma ORM , Headless UI, React-Toastify, supabase

Study

공부한 내용을 기록하기 위해 준비했습니다.

공부를 하면 할 수록 새로운 것을 배우는 것도 좋지만 배운 내용을 기록하여

잊지 않는 것도 중요하다는 것을 알게 되었습니다.

이에 기록하기 위해 스터디 내용을 기록해두었습니다.

url : <https://perfect-volcano-440.notion.site/29c3b3d5b34846a7bf6623e535ca7b5e?pvs=4>

 **스터디 내용이 궁금하시다면 눌러주세요**

<https://www.notion.so/29c3b3d5b34846a7bf6623e535ca7b5e>



자격증

자격증 이름	취득한곳	취득일
ITQ 한글엑셀 A등급 (A0012013326000428)	한국생산성본부	2013-12-14
ITQ 한글파워포인트 A등급 (A0012013326000428)	한국생산성본부	2014-08-07
ITQ 아래한글 A등급 (A0012013326000428)	한국생산성본부	2014-05-29
GTQ(그래픽기술자격) (GA02G13312202251)	한국생산성본부	2013-11-28
운전면허증1종보통	경기지방경찰청	2015-04-29
웹디자인기능사	한국산업인력공단	2021-07-09

교육 과정

- 2022.04 ~ 2022.07 웹 프론트엔드 과정 수료 패스트캠퍼스
- 2021.03 ~ 2021.07 웹 ui/ux 퍼블리셔과정수료 이젠아카데미

학력

폴리텍대학교 전기과(2년제)

2016.03~2021.02

한양사이버대학교 응용소프트웨어공학과(편입)

2025.03 ~ 재학 중