



박찬정 | 사용자 피드백을 중요하게 생각하는 개발자

Frontend



Contact.

Email. ckswjd9595@gmail.com

Phone. **010-9595-3465**

Channel.

Blog. [개인 개발 블로그](#)

GitHub. [Chanjeong](#)

Side-Project. [Projects](#)

[Portfolio](#)

Introduce.

저는 Next.js와 TypeScript 기반의 프론트엔드 개발경험을 중심으로 성능을 개선하는 일에 강점이 있습니다. 특히 Coloso에서 인턴경험을 하면서 백오피스 Vue 시스템을 Next.js로 완전히 마이그레이션하며, 비동기 처리와 상태 관리 구조를 새로 설계해 PDF 생성 및 정산 프로세스 자동화로 업무 효율을 10분 이상 단축시켰습니다. 또한, 중고거래 프로젝트에서는 최신 React/Next API를 활용해 중복 API 호출을 20% 줄이고 초기 로딩 속도를 300ms 개선하는 등, 사용자 경험 개선에 집중했습니다. 새로운 기술을 빠르게 습득하고 실무에 녹여내는 능력은 저의 가장 큰 강점입니다. 공식 문서를 통해 직접 학습하고, 사이드 프로젝트에 즉시 적용하며 기술을 체득하는 습관을 가지고 있습니다. 또한, 개인 기술 블로그도 만들어 운영중입니다. 이러한 역량을 바탕으로 성능 최적화와 사용자 중심 UI/UX 개선에 실질적인 기여를 할 자신이 있습니다.

Education.

 **RMIT University**

Information Technology (컴퓨터공학과) — (2017. 4 ~ 2022. 2 졸업)

Work Experience.

데이원컴퍼니(Coloso)

2025.06 ~ 2025.08 |  [근무 일지](#)

[수료증](#)

역할: 내부 정산 업무를 위한 백오피스 시스템을 리뉴얼하는 프로젝트에 프론트엔드 인턴으로 참여

기술: Next.js, TypeScript, Apollo Client, GraphQL, NestJS, Ant Design, styled-components, Cursor AI

담당 역할

1. 레거시 시스템 마이그레이션

- Vue 기반의 기존 정산 시스템을 Next.js 기반, Apollo Client와 GraphQL을 활용한 데이터 패칭 구조 재설계로 완전 이관
- **기술적 도전:** Vue의 반응형 시스템을 React의 상태 관리로 100% 변환
- **해결 과정:** Vue의 기반 컴포넌트 통신을 React의 콜백 함수와 상태 끌어올리기 패턴으로 마이그레이션
- **복잡한 폼 상태 관리:** 엑셀 업로드 시 실시간으로 폼 데이터를 업데이트하는 비동기 처리 로직 구현

2. 정산서 관리 기능 개발

- 정산서 신규 추가/편집 기능 구현
- 엑셀 파일 업로드 → 정산 아이템 자동 분류 및 저장 시스템 구축
- 정산 내역서 생성 및 PDF 일괄 생성, 미리보기, 회수 기능 개발
 - **기술적 도전:** Vue 기반의 PDF 생성 로직을 Next.js 환경에서 동작하도록 변환
 - **해결 과정:** 서버사이드에서 Promise.all을 활용한 병렬 PDF 생성 처리로 성능 최적화
 - **성과:** PDF 일괄 생성 및 이메일 발송 자동화로 직원들 작업시간 10분 이상 단축

3. 백엔드 아키텍처 이해 및 프론트엔드 연동

- **기술적 도전:** 프론트엔드 개발자로서 NestJS의 resolver, service, module 관계를 이해
- **해결 과정:** NestJS 모듈과 의존성 주입 구조를 학습, GraphQL resolver와 service 간 관계를 이해
- **성과:** TypeORM 엔티티와 GraphQL 스키마 간의 매핑을 이해, 효율적인 쿼리 최적화로 불필요한 데이터 요청 제거

Project Experience.

[전자상거래 및 커뮤니티] 다락마켓 (중고거래 플랫폼)

기간: 2025.01 ~ 2025.02 | 역할: 풀스택 개발

기술: Next.js, TypeScript, Prisma, Supabase, Tailwind CSS | [🔗 프로젝트 보기](#)

담당 역할

1. Next.js 15 최신 렌더링 최적화 구현

- **기술적 도전:** useActionState, unstable_cache 등 최신 React/Next.js API를 활용한 복잡한 상태 관리
- **해결 과정:**
 - 폼 제출 시 서버와 클라이언트를 동기화해 로딩 상태를 자동 관리
 - 게시글, 좋아요, 댓글을 독립적으로 캐싱, 태그 기반 무효화로 데이터 일관성 유지
- **성과:** 캐싱으로 중복 DB 요청 제거, API 호출 횟수 약 20% 감소 및 초기 로딩 시간 200~300ms 단축

2. 고성능 무한 스크롤링 시스템

- **기술적 도전:** Intersection Observer API와 서버 액션을 결합한 메모리 효율적 무한 스크롤링
- **해결 과정:**
 - useRef와 IntersectionObserver로 정밀한 스크롤 감지 (threshold 1.0)
 - 서버 페이지네이션과 클라이언트 상태 동기화
- **성과:** 화면에 보이는 데이터만 렌더링하여 5000개 이상 데이터 사용시에도 부드러운 스크롤링, 메모리 누수 방지

3. 실시간 채팅 아키텍처 설계

- **기술적 도전:** Supabase Realtime과 Prisma ORM 연동한 안정적인 실시간 메시징
- **해결 과정:**
 - RealtimeChannel을 활용한 실시간 메시징과 useOptimistic훅 패턴
 - 서버 액션과 실시간 브로드캐스트 분리로 데이터 일관성 확보
- **성과:** 먼저 메시지 UI를 갱신하고 서버에서 데이터 검증, Supabase Realtime 연결 안정성을 확보

4. 보안 시스템 강화

- **기술적 도전:** 보안 확인을 여러 단계로 수행, 세션 관리를 통한 안전한 인증 시스템
- **해결 과정:**
 - iron-session 서버사이드 세션 관리로 JWT 취약점 회피, bcrypt 해싱(salt rounds: 12) + zod 스키마 검증
- **성과:** 서버사이드 세션으로 토큰 탈취 위험 제거, 입력 데이터 스키마 검증으로 악성 요청 차단

[헬스케어 및 커뮤니티] 러닝 메이트 매치 서비스

기간: 2024.09 ~ 2024.11 | 역할: 프론트엔드 개발 (4인 팀)

기술: Next.js, TypeScript, Zustand, React Query, Tailwind CSS | [🔗 프로젝트 보기](#)

담당 역할

1. Zustand Persist를 활용한 상태 관리 및 인증 시스템

- **기술적 도전:** 클라이언트 사이드 상태 관리와 서버 인증 토큰 동기화
- **해결 과정:**
 - persist로 사용자 정보 영구 저장 및 partialize로 민감하지 않은 데이터만 선택적 저장
 - js-cookie로 accessToken, refreshToken 쿠키 관리
- **성과:** 브라우저 새로고침 후에도 로그인 상태 유지, 토큰 자동 갱신으로 재로그인 횟수 최소화

2. React Hook Form을 활용한 모집글 CRUD 시스템

- **기술적 도전:** 복잡한 모집글 작성/수정 폼의 상태 관리와 유효성 검사
- **해결 과정:**
 - Controller 컴포넌트를 사용해 성별, 제한 인원, 날짜 등 입력 필드 제어
 - rules 객체를 통한 실시간 유효성 검사 및 isSubmitting 상태로 폼 중복 제출 방지
- **성과:** 실시간 유효성 검사로 제출 전 오류 사전 차단, 중복 제출 방지

3. TMAP API를 활용한 고급 지도 기능 구현

- **기술적 도전:** 복잡한 지도 상호작용과 경로 시각화 시스템 구축
- **해결 과정:**
 - Tmapv2.Polyline을 통한 경로 시각화 및 LatLngBounds로 지도 범위 자동 조정
 - 마커 클릭 시 scrollIntoView와 fitBounds로 사용자 경험 개선
- **성과:** 경로 시각화로 러닝 코스 직관적 파악, 자동 지도 범위 조정으로 사용자 조작 최소화

Other Experience.

프론트엔드 교육과정 수료

교육기관: 제로베이스 | 2024.04 ~ 2024.11

1. **핵심 기술 학습**
HTML/CSS, JavaScript, React 등 프론트엔드 핵심 기술을 단계적으로 학습
2. **실무 역량 강화**
주차별 강의, 과제, 블로그 포스팅, 코딩테스트를 통해 이론과 실무 역량을 동시에 강화
3. **프로젝트 경험**
개인 및 그룹 프로젝트를 성공적으로 수행

육군 병장 만기 전역 (2022.05 ~ 2023.11)

Skill

- Front-End: TypeScript, React.js, Next.js, TailwindCSS, Zustand, React-Query
 - Back-End: Node.js, Prisma, Supabase, GraphQL
 - Database: PostgreSQL
 - Tools: Git, Notion, Slack, Jira
-