



이은빈

프론트엔드 개발자

전화번호

010-2337-9516

이메일

dmsqlsdl54@gmail.com

포트폴리오

<https://eunbin-portfolio.vercel.app/>

GitHub

<https://github.com/eunbinnie>

간단 소개

UI 구현에 강점을 가진 프론트엔드 개발자 이은빈입니다. 퍼블리셔 경험을 기반으로 빠르고 정확한 퍼블리싱 능력을 React와 결합해, 컴포넌트 기반으로 재사용성과 유지보수성이 뛰어난 UI를 구현해왔습니다.

팀 프로젝트에서는 담당 작업 영역의 개발 마감 기한을 평균 23% 단축하며 효율성을 높였고, 이를 통해 추가 기능과 애니메이션을 구현해 프로젝트의 완성도를 강화했습니다.

프로젝트

board-app

[배포 링크](#)

[GitHub Repository](#)

쿠키 기반 인증과 Supabase DB 연동을 구현한 게시판 프로젝트

개인 프로젝트 (FE) | FE 1명

역할 및 기여 | 회원가입, 로그인, 게시글 목록/작성/상세 페이지, Supabase 기반 인증/DB 연동

기술 스택 | React, TypeScript, Supabase, TailwindCSS

| 로그인, 회원가입 및 인증 흐름 구현

- Supabase Auth의 기본 localStorage 토큰 저장 방식을 JS 쿠키 기반으로 변경하여 인증 흐름을 직접 구현
- 회원 인증 여부에 따라 로그인 페이지 또는 홈으로 리다이렉트하는 **페이지 가드** 구현

| 게시판 기능 구현

- 게시글 목록, 상세 조회, 작성, 삭제 기능을 Supabase DB와 연동하여 구현
- 회원만 게시글을 작성할 수 있도록 처리하고, **작성자와 로그인한 사용자의 UUID를 비교해** 본인만 게시글을 삭제할 수 있도록 권한 제어
- 게시글 작성 시 유효성 검사 적용 및 등록 후 목록 페이지로 리다이렉트 처리

FANDOM-K

[배포 링크](#)

[GitHub Repository](#)

관심 있는 아이들을 추가하고 후원할 수 있는 팬 참여 서비스

マイグ레이션 프로젝트 (FE) | 단독 진행

역할 및 기여 | 기존 마이페이지, 사이트 전체 마이그레이션 및 리팩토링

기술 스택 | Next.js, React, TypeScript, React Query, Zustand, TailwindCSS

| React 환경에서 과도한 이미지 네트워크 요청 발생

- Swiper에서 고해상도 이미지 로딩 시 슬라이드 전환 지연 및 버벅임 발생
- **Next.js**로 마이그레이션하며 next/image를 활용해 WebP 변환 및 이미지 최적화 적용
- **네트워크 리소스 사용량 90% 절감**하여 슬라이드 전환 속도 개선

| useState와 fetch 기반 상태 관리의 복잡성 문제 발생

- **API 호출 단순화** 위해 React Query의 **infinite queries** 적용
- 관심 아이템 상태 관리를 위해 Zustand로 구조를 개선하고, Persist 미들웨어로 localStorage를 연동해 불필요한 상태 및 로직 제거
- 상태 관리 코드 **LOC 약 30%** 감소, 로직 복잡성을 줄여 가독성 향상

GlobalNomad

[배포 링크](#)

[GitHub Repository](#)

여행 체험을 등록하고 예약하며, 리뷰를 남길 수 있는 플랫폼

팀프로젝트 (FE) | FE 5명

역할 및 기여 | 팀장, 내 정보 수정, 예약 내역, 체험 등록·수정, 랜딩 페이지 디자인 및 퍼블리싱

기술 스택 | Next.js, React, TypeScript, React Query, TailwindCSS

| 협업 효율성 증대 및 작업 효율화

- 전체 **유저 플로우를 시각화**하여, 중복되는 UI 컴포넌트 설정 및 작업 우선순위 설정
- **칸반 보드**로 작업 상태를 공유하고, 사용 빈도가 높은 컴포넌트를 우선 개발하여 팀원 간 대기 시간 최소화

| ReactQuill 에디터의 SSR 미지원으로 dynamic import 시 layout shift 발생

- 에디터 위치에 동일 크기의 textarea 공통 컴포넌트를 초기 렌더링하여 **스켈레톤처럼 처리**, 자연스러운 전환 유도 및 layout shift 방지

| 후기 작성, 예약 취소 등 데이터 변경 이후 화면에 즉시 반영되지 않는 문제

- 새로고침 없이 변경 사항을 반영하기 위해 React Query의 **invalidateQueries**를 활용하여 **데이터 동기화** 처리
- 확정된 서버 응답 이후 상태 반영 방식을 사용해, 실제 예약 상태와의 불일치 가능성 방지

WekitBucket

[배포 링크](#)

[GitHub Repository](#)

위키 페이지를 작성·공유하고, 게시판을 통해 소통할 수 있는 협업 플랫폼

팀프로젝트 (FE) | FE 5명

역할 및 기여 | 자유게시판 리스트 · 상세 페이지, 공통 모달 컴포넌트

기술 스택 | Next.js, React, TypeScript, TailwindCSS

| 자유게시판 이미지 로드 지연으로 LCP 평균 6.9초 측정

- 이미지 포맷을 AVIF와 WebP로 최적화하여 로딩 성능 개선
- new Image()를 활용한 별도 이미지 유효성 검증으로 발생한 **중복 네트워크 요청**을 onError 이벤트 통합 처리로 방지
- LCP를 평균 1.9초로 약 72% 단축**

| SSR 전환 시 자유게시판에서 발생한 hydration 오류

- 서버는 UTC, 클라이언트는 KST를 사용해 시간 불일치로 인한 렌더링 차이 발생
- UTC를 KST로 변환하는 로직을 적용하여 SSR 환경에서도 일관된 시간 출력 및 hydration 오류 제거

업무 경험

아웃스탠더스 (2025.02 ~ 2025.03)

대학원생을 위한 정보 교류 플랫폼 '가방끈'을 운영하는 스타트업

직무 | 프론트엔드 인턴 (1개월 체험형 인턴십)

기술 스택 | Next.js, React, TypeScript, Emotion, MobX

| 가방끈 웹 서비스 리뉴얼

- 홈페이지 리뉴얼에 따라 새로운 디자인 시스템을 반영, 마크업 구조 및 UI 퍼블리싱 개선
- 회원가입 방식을 계정 정보와 추가 정보를 함께 입력하는 구조로 변경하면서, 비회원 상태에서의 추가 정보 입력 제약을 해결하기 위해 전역 상태 관리 적용
- 두 정보를 한 번에 전송할 수 있도록 흐름을 구성하고, API 미완성 단계에서도 안정적으로 대응

엔트위즈소프트 (2023.04 ~ 2024.01)

쇼핑몰과 홈페이지 개발을 전문으로 하는 SI 회사

직무 | 웹 퍼블리셔

기술 스택 | HTML, CSS, JavaScript

| 그라운드시소

- Date 객체를 활용해 잘못된 날짜 선택을 검증하고, 오류 데이터의 장바구니 추가를 차단하는 로직 개발

| Feelkeen

- 페이지 진입 시 회원 등급을 기반으로 적합한 카테고리를 판별하고, URL 쿼리에 해당 카테고리 번호를 동적으로 생성 및 변경하여 리다이렉션 처리 로직 구현
- 회원 등급 검증 로직을 모듈화하여 재사용 가능하도록 설계

| 워클로

- iframe 내부의 DOM에 접근하여 게시판 카테고리별 맞춤 입력 양식을 동적으로 제공하는 로직 구현

학력

순천향대학교 컴퓨터소프트웨어공학과 | 졸업

2018.03 ~ 2022.02

학점 3.5 / 4.5

교육

코드잇 스프린트 프론트엔드 트랙 6기 | 수료

2024.03 ~ 2024.09

- JavaScript, React, Next.js를 중심으로 최신 웹 개발 기술을 학습하고, 팀 프로젝트를 통해 협업 및 문제 해결 역량을 강화함.
- React Query를 활용한 서버 데이터 관리와 TypeScript를 적용해 코드의 안정성과 유지보수성을 강화하는 방법을 학습함.

자기소개서

프론트엔드 개발자로 전향한 이유

소통 창구를 설계하는 프론트엔드 개발자로 성장

제가 프론트엔드 개발자로 나아가게 된 계기는 퍼블리셔로 일하며 느낀 데이터 통신 지식의 중요성에서 비롯되었습니다. 특히 그라운드시소 프로젝트를 진행하던 중, "사용자가 티켓 상품을 장바구니에 추가할 때, 이미 다른 상품이 존재하면 추가를 제한하는 기능"을 구현해야했습니다. 그러나 카페24 플랫폼의 제약으로 인해 외부 페이지에서 장바구니 데이터를 확인할 수 없었고, 이로 인해 해당 요구사항을 충족하기 어려운 상황에 놓였습니다.

결국 티켓 상품은 단독 구매만 가능하다는 알림을 띄우고, 장바구니 내 결제를 제한하는 방식으로 해결했습니다. 이 과정에서 데이터 통신과 API 처리에 대한 이해의 부족이 기능 구현에 큰 제약이 된다는 점을 깨달았습니다. 이 경험을 통해 디자인 요소를 구현하는 퍼블리셔의 역할을 넘어, 데이터와 UI를 연결하여 기능의 완성도를 높이고 다양한 대안을 제시할 수 있는 프론트엔드 개발자가 되어야겠다고 결심했습니다.

이후 프론트엔드 부트캠프에 참여해 React를 활용한 동적 UI 렌더링, Next.js의 서버사이드 렌더링, 그리고 React Query를 이용한 서버 상태 관리 등을 학습하며 실질적인 기술 역량을 쌓았습니다. 팀 프로젝트에서는 이미지 최적화를 통한 렌더링 지연을 개선하고, 데이터 변경 후 UI가 새로고침 없이 즉시 반영되도록 처리했습니다. 사용자 경험을 고려한 구조를 직접 구현하며, 실제 문제에 기술적으로 대응하는 역량을 키울 수 있었습니다.

프론트엔드 개발은 UI를 구현하는 것을 넘어, 사용자와의 인터랙션을 설계하고 직접적으로 소통할 수 있는 창구를 만들어주는 역할이라고 생각합니다. 이러한 과정에서 사용자 경험을 개선할 뿐만 아니라 기술적인 한계를 극복하는데 필요한 창의적인 접근법을 활용할 수 있다는 점에 매력을 느낍니다. 앞으로도 데이터를 기반으로 문제를 해결하고, 사용자와의 소통 창구를 설계하며 만족도가 높은 서비스를 제공할 수 있는 프론트엔드 개발자로 성장하고자 합니다.

성격의 장점과 단점

체계적 실행과 소통 능력의 발전

저는 목표를 설정하고 체계적으로 실행하는 능력이 강점입니다. 퍼블리셔로 근무하던 시절, 제한된 기간 내에 20개 이상의 페이지를 제작해야 하는 상황에서 직접 WBS를 작성해 작업 단계를 세분화하고, 페이지별 작업 시간을 산출하며 진행 상황을 철저히 관리했습니다. 그 결과, 마감 기한을 2일 앞당기며 모든 페이지를 완성도 높게 제출할 수 있었습니다.

프론트엔드 부트캠프 팀 프로젝트에서도 요구 사항을 분석한 뒤 기능 우선순위를 설정하고, Notion을 활용해 역할을 체계적으로 분담하며 협업을 효과적으로 이끌어냈습니다. 이러한 경험을 통해 계획 수립 능력과 팀워크의 중요성을 더욱 체감하며, 제 강점을 협업 과정에서도 발휘할 수 있음을 확인했습니다.

하지만 처음에는 팀 논의 과정에서 여러 의견을 빠르게 정리하고 최적의 결론을 도출하는 데 시간이 걸리는 단점이 있었습니다. 이는 협업 경험 부족에서 비롯된 것으로, 초기 프로젝트에서 API 설계나 코드 컨벤션 논의 시 적극적으로 의견을 제시하지 못했던 경험이 있습니다.

이러한 단점을 극복하기 위해 Notion과 시각화 도구를 활용해 의견을 구조적으로 정리하고, 팀원들과의 논의를 통해 다양한 관점을 이해하는 연습을 했습니다. 마지막 프로젝트에서는 팀장으로서 팀원들의 의견을 빠르게 조율하고, 효율적으로 결론을 도출할 수 있었습니다. 이를 통해 프로젝트 완성도를 높이는 데 기여했습니다. 이러한 경험은 제가 의사결정 능력과 소통 능력을 지속적으로 발전시키는 개발자로 성장하는 계기가 되었습니다.