



이은빈

✉ dmsqlsdl54@gmail.com

☎ 01023379516 | 📅 1999년생

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.

개발 직무

프론트엔드 개발자

기술 스택

HTML5

CSS 3

JavaScript

TypeScript

React

Next.js

TailwindCSS

styled-components

React Query

Git

GitHub

학력

○ 2022.02 졸업

대학교(4년) | **순천향대학교**

컴퓨터소프트웨어공학과

3.5 / 4.5

경력(업무경험)

개발경력 0년 10개월

○ 2023.04 - 2024.01

0년 10개월

주식회사엔트위즈소프트

쇼핑몰과 홈페이지 개발을 전문으로 하는 SI 회사

웹 에이전시팀 / 퍼블리셔

HTML5

CSS 3

JavaScript

그라운드시소 (2023.09 ~ 2023.10)

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

Feelkeen (2023.07 ~ 2023.11)

• 회원 등급에 따라 URL 쿼리 매개변수를 변경하고 적합한 페이지로 리다이렉션 처리하는 로직 구현

- 회원 등급 검증 로직을 모듈화하여 재사용 가능하도록 설계

워크로 (2023.05 ~ 2023.06)

- JavaScript로 iframe 내부에 접근하여 게시판 카테고리별 맞춤 입력 양식을 동적으로 제공하는 로직 구현

프로젝트

○ 2024.07 - 2024.08

GlobalNomad

여행 체험을 등록·예약하고 리뷰를 남길 수 있는 플랫폼

Next.js

React

TypeScript

TailwindCSS

axios

React Query

- 팀 구성: FE 5명

[문제 상황 및 해결 방법]

1. 초기 데이터 로딩 지연 및 CLS 0.617초 측정
 - React Query의 SSR prefetching 도입해 CLS를 약 58% 단축
2. 사용자 인터렉션 중 데이터 갱신 문제
 - React Query의 invalidate 기능을 활용해 데이터 변동 시 최신 상태를 자동 동기화
3. ReactQuill 에디터의 SSR 미지원으로 인해 어색한 UI 전환 발생
 - 초기 상태에서 textarea 컴포넌트를 활용해 로딩 중 UI의 이질감을 해소
4. 후기 작성 후 결과 피드백 부족
 - Lottie 애니메이션을 적용해 성공/실패 여부를 직관적으로 전달

웹/마켓/스토어

· <https://sp-globalnomad-6-9.vercel.app/>

저장소 링크

· <https://github.com/part4-team9/global-nomad>

○ 2024.06 - 2024.07

WekitBucket

지인들과 위키 페이지를 작성·공유하며 게시판으로 소통할 수 있는 협업 플랫폼

Next.js

React

TypeScript

TailwindCSS

○ 2024.04 - 2024.05

• 팀 구성 : FE 5명

1. SSR 전환 시 자유게시판에서 hydration 오류 발생

• 시간 포매팅 문제를 커스텀 hook으로 분리해 서버와 클라이언트 간 시간 불일치를 해결

2. 새로고침 시 상태 초기화 문제

• useSearchParams를 활용해 정렬, 페이지네이션 등 상태 정보를 쿼리 문자열에 저장해 데이터 복원 구현

3. react-hook-form 사용 중 register 전달 에러 발생

• React.forwardRef()로 자식 컴포넌트에서 ref를 참조하며 재사용 가능한 입력 컴포넌트 구현

웹/마켓/스토어

· <https://wekitbucket.vercel.app/>

저장소 링크

· <https://github.com/eunbinnie/WeKitBucket>

Fandom-K

관심 있는 아이돌에게 투표하고 후원할 수 있는 서비스 플랫폼

React

JavaScript

styled-components

• 팀 구성: FE 5명

[문제 상황 및 해결 방법]

1. 슬라이드로 API 데이터 동적 로딩

• Swiper의 onSlideChange와 navigation 옵션을 사용해 슬라이드 이동 및 버튼 클릭 시 API 데이터를 동적으로 로드

2. 반응형 디자인에 따른 아이템 개수 동적 변경

• 2차원 배열을 reduce와 concat으로 1차원 배열로 변환 후, 반응형 조건에 따라 다시 2차원 배열로 재구성하여 데이터 렌더링

3. Next 슬라이드 이동 시 딜레이 문제

• 데이터 로딩 중 스켈레톤 UI를 표시해 슬라이드 이동과 데이터 로드 프로세스를 분리하여 즉시 이동 가능하도록 설계

웹/마켓/스토어

· <https://sprint11fandom-k.netlify.app/>

저장소 링크

· <https://github.com/Sprint6-Fandom-K/Fandom-K>

○ 2024.03 - 2024.09

코드잇 스프린트 프론트엔드 트랙 6기

HTML5

CSS 3

JavaScript

Git

GitHub

React

styled-components

TypeScript

Next.js

React Query

- 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. 예를 들어, 팀 프로젝트에 Next.js의 SSR을 적용해 CLS를 약 58% 단축시켰고, react-query로 API 호출을 최적화하며 불필요한 데이터 요청을 줄이는 구조를 설계했습니다. 이 경험을 통해 기술적 과제를 해결하며 성과를 만들어내는 방법을 체득했습니다.

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 극복하는 데 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.

○ 기억에 남는 프로젝트

협업과 UI 개선으로 완성한 GlobalNomad

부트캠프에서 진행한 마지막 팀 프로젝트 'GlobalNomad'는 제가 프론트엔드 개발자로서의 역량을 확실히 다질 수 있었던 경험이자, 성장의 전환점이 된 프로젝트입니다. 이전 두 번의 프로젝트는 짧은 기간(2~3주) 동안 진행되었기에 빠른 기능 구현과 일정 준수에 초점이 맞춰졌고, 코드 리뷰 부족과 공통 컴포넌트 정리 미흡으로 중복 코드가 다수 발생하는 아쉬움이 있었습니다.

이러한 경험을 바탕으로 'GlobalNomad' 프로젝트에서는 팀장으로서 코드 품질과 협업 방식을 초기 단계부터 체계적으로 계획했습니다. 팀원들과 함께 유저 플로우를 시각화하고 실시간으로 소통하며 요구 사항을 보완했고, 기능 우선순위를 설정해 효율적이고 재사용 가능한 컴포넌트 중심의 개발 방향을 명확히 정리했습니다. 이 과정에서 팀의 적극적인 소통이 다양한 대안을 도출하고, 결과적으로 더 완성도 높은 결과물을 만들어낸다는 것을 체감했습니다.

팀원들과 협업을 원활히 하기 위해 공통 컴포넌트를 각각 담당하도록 분배했고, 저는 input 컴포넌트를 공통 컴포넌트로 제작하여 재사용성을 극대화했습니다. 또한, JSDoc을 활용해 컴포넌트의 사용 방법과 데이터 흐름을 문서화함으로써 팀원들이 빠르게 이해하고 활용할 수 있도록 유지보수성을 높였습니다.

사용자 경험 개선에도 적극적으로 기여했습니다. 기존 후기 작성 모달은 등록 성공 여부를 명확히 전달하지 못해 QA 과정에서 혼란을 초래하는 문제가 있었습니다. 이에 사용자 입장에서 프로세스를 재검토했고, 후기 작성 모달 리팩토링 작업을 제가 맡았습니다. 이를 해결하기 위해 Bottom Sheet를 추가하여 후기 등록 여부를 재확인하는 절차를 도입했으며, 등록 후 삭제가 불가하다는 안내 메시지를 명확히 제공하도록 수정했습니다. 또한, 성공 또는 실패 여부를 직관적으로 전달하기 위해 Lottie 애니메이션을 활용했습니다. 작업 이후, 팀원들에게 "후기 작성 과정과 결과 피드백이 명확해졌다"는 긍정적인 평가를 받은 점이 가장 기억에 남습니다.

'GlobalNomad' 프로젝트는 프론트엔드 개발자로 전향하며 목표로 삼았던 "데이터와 UI를 연결해 기능 완성도를 높이고 사용자와의 소통 창구를 설계"한다는 목표를 실현한 경험이었습니다. 이 프로젝트를 통해 단순히 기능을 구현하는 것을 넘어, 팀원들과 협업하며 함께 문제를 해결하고 더 나은 서비스를 만들어내는 개발자로 성장할 수 있었습니다.

○ 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.

링크

Github

<https://github.com/eunbinnie>

포트폴리오

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