

프로젝트 상세 경력기술서

제갈민혁 / 프론트엔드 개발자 / 2024.09 ~ 2026.06

(주)방배동밸리 / 프론트엔드 개발 팀원 (2026.03 - 2026.06)	
Frontend Developer	ASDF / 사용자 온보딩 및 관리자 운영 기능 고도화 개발환경 Next.js, React, TypeScript, Zustand, next-intl, axios, ExcelJS, react-toastify, Tailwind CSS 사용 기술 • Next.js • React • TypeScript • Zustand • next-intl • axios • ExcelJS • react-toastify • Tailwind CSS Situation 사용자 온보딩 과정에 회원가입, 약관 동의, 아이디·비밀번호 찾기, 관심사 선택, 회원탈퇴, 다국어 처리, 보호 라우트 등 여러 흐름이 포함되어 있었습니다. 또한 관리자 화면에서는 로그, 회원 정보, 회원 로그 등 운영 데이터를 반복적으로 다운로드해야 했고, 프로필 수정, 댓글, 페이지 편집 등 여러 화면에서 유사한 사이드바 UI가 필요했습니다. Task 프론트엔드 개발 팀원으로서 온보딩 흐름을 안정적으로 구현하고, 반복적으로 필요한 사이드바와 엑셀 다운로드 기능을 공통 구조로 정리하는 역할을 맡았습니다. 화면별 개별 구현을 줄이고, 신규 화면에서도 동일한 구조를 재사용할 수 있도록 만드는 것이 주요 과제였습니다. Action Next.js, React, TypeScript 기반으로 회원가입, 약관, 아이디·비밀번호 찾기, 관심사 선택, 회원탈퇴, i18n, 보호 라우트 기반 온보딩 흐름을 구현했습니다. Zustand 기반 공통 사이드바 상태 구조와 useSidebar 훅을 설계해 프로필 수정, 댓글, 페이지 편집 등 여러 화면에서 동일한 방식으로 사이드바를 제어할 수 있도록 구성했습니다. 또한 공통 엑셀 다운로드 버튼, 다운로드 훅, API 모듈, 시트 생성 유틸을 분리해 다운로드 대상 데이터 정의만 추가하면 동작하는 구조로 표준화했습니다. Result 공통 사이드바 구조 도입으로 유사 사이드바 기능 개발 시간을 약 30% 단축했습니다. 공통 엑셀 다운로드 구조를 관리자 로그, 회원 정보, 회원 로그 화면에 적용해 유사 다운로드 기능 개발 시간을 약 40% 단축했고, 운영 데이터 추출 흐름을 일관되게 개선했습니다.

(주)방배동밸리 / 프론트엔드 개발 팀원 (2025.10 - 2026.05)

Frontend
Developer

FLOBEE User / 구매자 주문·탐색 흐름 개선

서비스 설명

FLOBEE는 꽃집 탐색, AI 오더, 무료견적, 입점사 운영, 관리자 백오피스 기능을 포함한 플랫폼입니다. 프론트엔드 개발 팀원으로서 구매자 서비스, 판매자 서비스, 관리자 백오피스의 주요 화면과 데이터 처리 흐름을 구현했습니다.

개발환경

Next.js, React, TypeScript, Zustand, React Hook Form, axios, ExcelJS, Toss Payments SDK, react-daum-postcode, react-toastify, date-fns, Swiper, Tailwind CSS

사용 기술

• Next.js • React • TypeScript • Zustand • React Hook Form • axios • ExcelJS • Toss Payments SDK • react-daum-postcode • react-toastify • date-fns • Swiper • Tailwind CSS

Situation

구매자 서비스에서는 사용자가 주변 꽃집을 지도와 리스트로 탐색하고, AI 오더 또는 무료견적을 통해 주문 흐름으로 진입해야 했습니다. 지도, 리스트, 필터, 정렬, 거리 정보, 마커, 무한스크롤 등 여러 UI 요소가 함께 동작해야 했고, 지도 재조회 과정에서 API 중복 호출이 발생할 수 있는 상황이었습니다. 또한 비회원 사용자의 주문 진입 과정에서는 배송지 입력, 지오코딩, 로그인 예외 처리가 필요했습니다.

Task

주변 꽃집 탐색 화면의 사용성을 개선하고, API 호출 구조를 정리해 불필요한 중복 요청을 줄이는 역할을 맡았습니다. 동시에 AI 오더와 무료견적에서 비회원 사용자도 주문 정보 입력 흐름에 진입할 수 있도록 예외 처리와 입력 과정을 안정화해야 했습니다.

Action

주변 꽃집 지도·리스트 화면에 필터, 정렬, 거리 정보, 마커, 리스트·지도 토글, 무한스크롤을 구현했습니다. 지도 재조회와 주변 꽃집 탐색 흐름을 정리해 API 중복 호출을 줄였고, 사용자가 지도와 리스트를 오가며 탐색할 수 있도록 화면 상태를 관리했습니다. AI 오더와 무료견적에서는 비회원 배송지 입력, 지오코딩, 로그인 예외 처리, 샘플 이미지, 용도 선택 흐름을 구현했습니다.

Result

주변 꽃집 탐색 API 중복 호출을 줄여 API 호출량을 약 35% 감소시켰습니다. 지도 재조회 및 주변 꽃집 탐색 흐름 개선을 통해 Lighthouse 점수를 45점에서 78점으로 향상시켰습니다. 비회원 주문 진입 이탈률을 약 20% 낮추고, 주문 정보 접근 시간을 약 25% 단축했습니다.

(주)방배동밸리 / 프론트엔드 개발 팀원 (2025.10 - 2026.05)

Frontend
Developer

FLOBEE Vendor / 판매자 주문·상품 운영 화면 개선
개발환경

Next.js, React, TypeScript, Zustand, React Hook Form, axios, ExcelJS, Toss Payments SDK, react-daum-postcode, react-toastify, date-fns, Swiper, Tailwind CSS

사용 기술

• Next.js • React • TypeScript • Zustand • React Hook Form • axios • ExcelJS • Toss Payments SDK • react-daum-postcode • react-toastify • date-fns • Swiper • Tailwind CSS

Situation

판매자 서비스에서는 입점사가 영업 상태 변경, 꽃집 정보 수정, 주문 확인, 정산 내역 조회, 상품 관리 등 여러 운영 업무를 처리해야 했습니다. 각 기능에서 API 연동, formData 처리, 탭별 화면 전환, 예외 처리가 필요했고, 판매자가 운영 상태를 빠르게 확인하고 수정할 수 있는 화면 구조가 필요했습니다.

Task

입점사 운영 화면에서 영업 상태 변경과 꽃집 정보 수정 흐름을 안정화하고, 주문·정산·상품 관련 화면을 탭 구조로 정리해 탐색성을 개선하는 역할을 맡았습니다.

Action

메인페이지 영업 상태를 API로 제어하고, 영업중·영업종료 상태 표시 및 변경 예외 처리를 구현했습니다. 꽃집 정보 수정 화면에서는 update 흐름, 은행명, 좌표, 대표 이미지 formData 처리 구조를 정리했습니다. 이지오더, 맞춤구매, 입점관리, 월별 정산내역, 상품 목록은 탭별 컴포넌트로 구성해 판매자가 필요한 정보를 빠르게 확인할 수 있도록 했습니다.

Result

영업 상태 변경 처리 시간을 약 30% 단축했습니다. 입점사 정보 수정 오류를 약 25% 감소시켰고, 주문·정산 화면을 탭 구조로 정리해 탐색 시간을 약 20% 단축했습니다.

(주)방배동밸리 / 프론트엔드 개발 팀원 (2025.10 - 2026.05)

Frontend
Developer

FLOBEE Admin / 백오피스 운영 데이터 처리 흐름 표준화
개발환경

Next.js, React, TypeScript, Zustand, React Hook Form, axios, ExcelJS, Toss Payments SDK, react-daum-postcode, react-toastify, date-fns, Swiper, Tailwind CSS

사용 기술

• Next.js • React • TypeScript • Zustand • React Hook Form • axios • ExcelJS • Toss Payments SDK • react-daum-postcode • react-toastify • date-fns • Swiper • Tailwind CSS

Situation

관리자 백오피스에서는 회원 상태별 조회, 쿠폰 생성 및 발급, 마일리지, 활동 내역, 입점업체 데이터 등 다양한 운영 데이터 처리가 필요했습니다. 기능별로 검색, 필터, 페이지네이션, 엑셀 다운로드 요구사항이 반복되었기 때문에 화면마다 별도로 구현하면 개발 효율과 유지보수성이 떨어질 수 있는 상황이었습니다.

Task

회원 관리와 쿠폰 운영 화면을 기능별로 분리하고, 반복적으로 필요한 조회·다운로드 흐름을 재사용 가능한 구조로 정리해야 했습니다. 관리자가 필요한 데이터를 빠르게 조회하고 추출할 수 있도록 UI와 데이터 처리 흐름을 개선하는 것이 주요 과제였습니다.

Action

회원 관리 페이지를 목록, 활동, 휴면, 탈퇴, 대상 회원 탭으로 분리하고 검색, 필터, 페이지네이션 구조를 설계했습니다. 온라인·오프라인 쿠폰 생성, 발급 목록, 즉시 발급, 엑셀 다운로드 기능을 구현했습니다. 또한 공통 엑셀 버튼과 다운로드 혹은 회원, 활동, 마일리지, 쿠폰, 입점업체 화면에서 재사용할 수 있도록 구성했습니다.

Result

회원 관리 화면의 조회·처리 흐름을 개선해 회원 조회 및 관리 처리 시간을 약 30% 단축했습니다. 쿠폰 운영 데이터 추출 시간을 약 40% 단축했으며, 공통 엑셀 다운로드 구조 재사용으로 유사 다운로드 기능 개발 시간을 약 35% 단축했습니다.

(주)방배동밸리 / 프론트엔드 개발 팀원 (2025.07 - 2026.06)

Frontend
Developer

WOWCRUISE / 크루즈 예약 전환 플로우 구축

개발환경

React, react-router-dom, react-bootstrap, styled-components, axios, Swiper, Highcharts

사용 기술

• React • react-router-dom • react-bootstrap • styled-components •
axios • Swiper • Highcharts

Situation

크루즈 예약 과정은 객실 선택, 승객 정보 입력, 결제, 예약 완료까지 여러 단계로 구성되어 있었고, 단계가 길어질수록 필수 정보 누락, 세션 만료, 재진입 실패, 모달 중복 구현 등으로 사용자가 이탈할 수 있는 상황이었습니다.

Task

예약하기 depth 1~5 플로우를 구축하고, 예약 단계별 입력 검증과 세션 복원 흐름을 안정화해야 했습니다. 또한 예약, 검색, 마이페이지, 회원가입, 네비게이션 등 여러 화면에서 반복되는 모달을 공통 구조로 정리하는 역할을 맡았습니다.

Action

예약하기 depth 1~5 플로우를 구성하고 객실 선택, 승객 정보 입력 검증, 결제, 예약 완료 화면을 구현했습니다. 승객 정보 입력 검증과 버튼 활성화 조건을 정리해 필수 정보 누락 시 다음 단계로 진행되지 않도록 처리했습니다. 예약 세션 복원, 로그아웃 후 복원 차단, TTL 예외 처리를 적용해 예약 재진입 흐름을 개선했습니다. 공통 모달 구조를 구성해 예약, 검색, 마이페이지, 회원가입, 네비게이션 화면에서 재사용할 수 있도록 했습니다.

Result

예약 단계 이탈률을 약 18% 감소시켰습니다. 예약 진행 오류를 약 25% 줄이고, 예약 재진입 성공률을 약 20% 개선했습니다. 공통 모달 구조를 통해 유사 모달 기능 개발 시간을 약 30% 단축했습니다.

(주)방배동밸리 / QA 및 프론트엔드 개발 팀원 (2025.04 - 2025.07)

Frontend
Developer

ADER / QA 및 신규 디자인 적용
개발환경

PHP, JavaScript, CSS, Swiper

사용 기술

• PHP • JavaScript • CSS • Swiper

Situation

마이페이지 내 주문내역, 취소·환불, 계정 관련 화면은 사용자 유형과 상태에 따라 다양한 분기와 예외 케이스가 존재했습니다. 또한 신규 디자인 적용 과정에서 화면 레이아웃, 버튼 동작, 데이터 노출 상태가 기존 기능과 충돌하지 않도록 검증이 필요했습니다.

Task

마이페이지 유형별 사용자 시나리오를 기준으로 QA를 진행하고, 발견된 UI·기능 이슈를 수정하는 역할을 맡았습니다. Figma 기반 신규 디자인을 반영하면서 주요 사용자 흐름의 안정성을 유지하는 것이 과제였습니다.

Action

마이페이지 유형별 사용자 시나리오를 기준으로 QA를 진행하고, 주문내역, 취소·환불, 계정 관련 화면에서 상태 분기와 예외 케이스를 점검했습니다. 발견된 UI·기능 이슈를 수정했고, Figma 기반 신규 디자인을 반영해 화면 레이아웃과 UI 컴포넌트 구조를 정리했습니다. 디자인 적용 과정에서 화면 이동, 버튼 동작, 데이터 노출 상태를 함께 검증했습니다.

Result

마이페이지 유형별 QA와 버그 수정을 통해 동일 이슈 재발률을 20% 이하로 낮췄습니다. 신규 레이아웃 적용 후 주요 사용자 화면의 디자인 적합성과 화면 안정성을 개선했습니다.

(주)방배동밸리 / 프론트엔드 개발 팀원 (2024.09 - 2025.06)

Frontend
Developer

SIDIZ / 고객센터 및 마이페이지 사용성 개선
개발환경

Shopify Liquid, JavaScript, CSS, jQuery, HTML

사용 기술

• Shopify Liquid • JavaScript • CSS • jQuery • HTML

Situation

고객센터와 마이페이지에서 AS 신청, 1:1 문의, 대량구매 문의, 정품등록, 거래명세서 발급, 주문 조회, 리뷰, 적립금, 배송지 변경 등 다양한 사용자 후속 처리 기능이 필요했습니다. 각 기능은 품 상태, 정책 모달, API 연동, 비동기 처리, 검증, 상태 표기가 함께 요구되는 구조였습니다.

Task

고객센터와 마이페이지의 주요 사용자 흐름을 API와 연동하고, 입력 검증과 상태 표기를 개선해 구매 이후 사용자가 필요한 작업을 안정적으로 처리할 수 있도록 하는 역할을 맡았습니다.

Action

AS 신청, 1:1 문의, 대량구매 문의, 정품등록, 거래명세서 발급 품 상태와 정책 모달을 정리했습니다. 마이페이지 주문, AS, 리뷰, 적립금, 배송지 변경 흐름을 API와 연동하고 비동기 처리, 검증, 상태 표기를 개선했습니다. 장바구니 로그인 flag, 주문 상세 연락처 변경, 주문 취소·조회 조건, 비회원 주문조회 정책 모달을 적용했습니다. 상품 상세 리뷰 필터, 옵션, 데이터 추가, 추천제품 검색, 검색 결과 피드백, 공통 input/modal CSS도 함께 정리했습니다.

Result

주문, 장바구니, 리뷰, 비회원 주문조회 상태 동기화를 개선해 구매 후속 흐름의 사용성을 높였습니다. 고객센터와 마이페이지에서 반복적으로 사용되는 입력, 모달, 상태 처리 흐름을 정리해 사용자 작업 흐름을 안정화했습니다.