



김학재

프론트엔드 개발자

🏠 hakjae.dev
🌐 linkedin.com/in/hakjae
✉ hakjae.dev@gmail.com
🐙 github.com/gouz7514

최대한의 사용자에게 최고의 가치를 제공하는 것을 우선으로 생각합니다.
프론트엔드를 넘어 다양한 도메인에 걸쳐 주도적으로 일하고 성과를 내는 것을 좋아합니다.
같이 일하고 싶은 팀을 만들고 그 안에서 모두가 같이 일하고 싶은 개발자로 성장하는 데에 관심이 있습니다.

경력

에이비제트

프로덕트팀 2024.01 ~ 현재

크리에이터들이 겪는 문제를 해결하고, 하나의 플랫폼으로 성장할 수 있도록 돕는 서비스를 개발하고 있습니다.
개발팀 문서의 전반적인 개선, 새로운 개발 문화의 도입을 통해 함께 성장하는 방법을 고민하고 실행하고 있습니다.

Next.js 배포 시간 단축 (Blog)

Next.js 서비스의 CI/CD 파이프라인을 점검하고 전체 배포 시간을 획기적으로 줄여 팀의 생산성 향상에 기여했습니다.

- Next.js 서비스 빌드, Github Action, AWS Codedeploy까지 이어지는 파이프라인 분석 및 병목 파악
- Yarn Berry와 PnP, Next.js의 standalone 옵션을 적용, 서비스 빌드 산출물의 경량화
- Dockerfile을 멀티 스테이지 구조로 개선, 서비스 실행에 필요한 최소 파일만의 빌드를 통해 도커 이미지 용량 90% 감축
- **전체 배포 시간 50% 단축** (Github Action, Codedeploy 합 약 10분 내외 → 5분 내외)

Typescript, Next.js, Yarn, Docker, AWS

인포크딜 개발

AI 추천, 실시간 채팅을 통해 브랜드와 크리에이터가 협업 상대를 탐색하고 소통할 수 있는 서비스를 개발했습니다.

- 샌드버드 SDK의 customType, metadata를 활용해 다양한 타입의 채팅을 정의하고 관련 기능을 개발
- react-hook-form과 zod를 활용해 schema, hook을 정의하고 복잡한 form을 효율적으로 관리
- react-query의 useQuery, useMutation을 활용해 모든 api 호출을 훅으로 구현, 캐시 invalidate를 통해 api 호출 최적화
- DDD 패턴을 도입, 관심사에 따라 컴포넌트/훅/유틸 함수를 분리해 코드의 응집도, 가독성 향상
- 대면/전화 고객 인터뷰를 통해 요구사항을 빠르게 파악하고 프로덕트에 반영

Typescript, Next.js, react-query, react-hook-form, zod, Emotion, AWS

이벤트 수집 로직 안정성 개선 (Blog)

인앱 브라우저에서 네트워크 환경에 따라 영향을 받던 이벤트 수집 로직의 안정성을 개선했습니다.

- axios를 활용한 비동기 요청 방식의 지연 문제를 개선하기 위해 주도적으로 리서치를 진행
- 요청의 응답 여부를 확인하지 않고 요청 전송 여부만 확인하는 sendBeacon API를 활용해 로직 개선
- 기존 비동기 요청 방식의 지연 문제 해결, 이벤트 유실률 100% 감소, 이전보다 더 빠른 페이지 이동을 통해 개선된 UX 제공

Javascript, Nuxt, Vue.js

인포크마켓 개발

공동구매 이해관계자 (브랜드, 크리에이터, 벤더) 누구나 활용 가능한 자사몰 구축 솔루션을 개발했습니다.

- 토스페이먼츠 SDK 연동, 상품의 구매부터 환불까지 상품 생애주기 전체에 걸친 비즈니스 로직 개발
- react-hook-form을 활용해 상품의 정보를 입력받고 여러 컴포넌트에서 하나의 form을 참조하는 구조 개발, 재사용성 · 가독성 향상
- turborepo를 활용한 모노레포 환경 내에서 프로젝트 생성부터 AWS를 활용한 배포까지 개발 전 과정 100% 기여
- Next.js의 standalone 옵션을 활용, 빌드된 docker 이미지의 용량을 약 40% 감축

Typescript, Next.js, react-query, react-hook-form, recoil, Emotion, AWS

에이비제트

프로덕트팀 2021.06 ~ 2023.04

국내 1위 링크 인 바이오 서비스, 인포크링크의 개발/유지/보수를 담당했습니다.

기능 개발, 성능 개선부터 도커를 활용한 서버 컨테이너화까지 다양한 도메인에 대해 학습하고 팀에 기여했습니다.

경영 악화로 인해 퇴사하게 되었습니다.

vue-query 도입 (Blog)

인포크링크에서 페이지 이동마다 api를 요청하던 문제를 해결하고 서비스 속도 및 UX를 개선하기 위해 vue-query를 도입했습니다.

- vue2 환경에서 vue-query 사용을 위해 composition-api를 활용한 커스텀 플러그인 구현
- 기존의 asyncData, mounted 등 여러 라이프사이클의 로직을 하나의 setup 라이프사이클로 이전하고 캐싱을 적용
- 다양한 직군의 동료(PM, 디자이너, 개발자)와 논의를 통해 캐싱이 필요한 로직에 대한 정의 및 전략 수립. 로딩, 스켈레톤 UI 적용
- API 호출 횟수 약 15% 절감, core web vital 개선 (FCP 50%, LCP 80% 이상)

Javascript, Nuxt, Vue.js, vue-query

도커를 활용한 서버 컨테이너화 (Blog)

더 빠른 오케스트레이션과 배포 시간 단축을 위해 docker를 활용해 서버를 컨테이너화하고 CI/CD 파이프라인을 구축했습니다.

- Django 기반의 서버 아키텍처에 nginx를 추가하고 docker를 활용해 하나의 이미지로 구성
- AWS Elastic Beanstalk 기반 서버를 ECS로 마이그레이션해 오케스트레이션 속도 향상 및 배포 안정성 개선
- AWS CDK와 Github Actions를 활용해 누구나 콘솔 접근 없이 코드만으로 배포 가능한 파이프라인 구축
- ECS 지표를 기반으로 한 오토 스케일링과 CloudWatch를 활용해 트래픽이 몰려도 더 안정된 서버 아키텍처 개발

Docker, Nginx, AWS ECS, AWS CodeDeploy, AWS CDK

디자인 시스템 개발

통일된 디자인, 톤앤매너를 위해 서비스 전체에 걸쳐 사용 가능한 디자인 시스템을 개발했습니다.

- 디자이너, 동료 개발자와 지속적인 의사결정을 통해 컴포넌트 네이밍 규칙, 배리에이션, 사용 방법에 대한 정의
- 아토믹 디자인 패턴을 기반으로 각 컴포넌트를 분리하고 재사용과 확장이 용이한 컴포넌트 개발
- 스토리북 배포를 통해 다양한 컴포넌트 테스트, 모두가 하나의 화면을 보고 소통함으로써 협업 효율성 향상
- 비즈니스 로직의 고도화에 따른 디자인 시스템의 개선을 지속적으로 진행

Javascript, Vue.js, Storybook

기술.

- 언어 : Typescript, Javascript
- 프레임워크 및 라이브러리 : React, Next.js, Vue.js, Nuxt, React Hook Form, Zod
- 상태 관리 : React Query, Recoil, Vue Query, Vuex
- 클라우드 서비스 : AWS, Docker
- 기타 도구 : Github, Slack, Notion, Figma

경험.

오픈소스 기여 (Blog)

- 리액트 공식문서 한글 번역 및 CI 오류 해결
- Next.js 한글 번역

단국대 응용컴퓨터공학과

2014.03 ~ 2021.02

캡스톤 경진대회 금상 (Youtube)

졸업과제로 단국대 시간표와 지도를 기반으로 한, 알람 & 내비게이션 애플리케이션을 개발해 금상을 수상했습니다.

- 지도 데이터 구축을 위해 Mapbox사와 메일로 소통하며 교내 지도의 정확도 향상
- 팀장으로서 각 구성원의 참여를 끌어내고 협의를 통해 문제를 해결한 경험 보유
- 해당 프로젝트에 대해 문의해 온 대학 후배를 비롯한 많은 사람들에게 도움을 준 경험 보유