

이상민, Frontend Developer

✉ sangmin.tsx@gmail.com

☎ 010-5604-0978

🌐 <https://github.com/sangminlee98>



Summary

- React.js, Next.js, React Native를 사용해 반응형 클라이언트를 개발 할 수 있으며, 반복적인 배포 작업을 줄이기 위해 CI/CD 파이프라인을 구축하고 Docker를 사용하여 컨테이너 빌드와 클라우드에 서비스를 배포 할 수 있습니다.
- Lighthouse와 같은 분석 도구를 통해 성능 문제를 파악하고 Next.js의 서버 컴포넌트와 React-Query를 사용하여 효율적으로 렌더링을 최적화한 경험이 있습니다.
- 네트워킹과 지식 공유를 중요하게 여기며, 개발자 커뮤니티에서 멘토링, 기술 컨퍼런스를 주도적으로 주최한 경험이 있습니다.
- End-to-End 프로젝트 리드 경험이 있어 전체 개발 프로세스 이해를 기반으로 백엔드, 디자이너, 데브옵스와의 커뮤니케이션에 능숙한 프론트엔드 개발자 입니다.



Waffle - C2C 래플/공구 서비스

2024.07 ~ 진행 중 | FE Leader

Github : [Link](#) | Tech Blog : [Link](#)

비인증 결제를 통해 소비자간에 래플 및 상품을 거래할 수 있는 Webview 기반 앱 서비스

기술 스택 | Next.js, React Native, React, Turborepo, pnpm, Next-Auth, React Query, Zod, Storybook, Shadcn, TailwindCSS

[회원가입 스텝 및 인증 시스템 개발]

- Next-Auth의 인증 세션과 Server Action을 사용하여 **재사용성과 확장성을 고려한 공통 Fetch Client** 구현 [\[Blog\]](#)
 - 민감한 데이터 보호와 검증 우회 방지를 위해 useTransition을 활용하여 **react-hook-form**과 **Server Action**을 결합해 Form 양식 제출을 구현
 - **middleware**를 이용하여 사용자 권한별 페이지 제어 관리
- Suspense와 Server Action을 결합하여 무한스크롤 구현 중 초기 렌더링 시 Server Action 함수를 호출할 수 없는 문제 발생
 - HydrationBoundary를 통해 Query Client를 dehydrate하여 **서버 컴포넌트에서 prefetch된 데이터를 사전 렌더링함으로써 Fetch Waterfall 문제 해결** [\[Blog\]](#)
- 복잡한 회원가입 절차를 개선하기 위해 **Funnel 패턴**과 **zod, react-hook-form**을 도입하여 단계별로 프로세스를 세분화하고, 각 스텝에서 zod schema를 기반으로 한 react-hook-form 검증을 위해 Input과 Form 컴포넌트 추상화 [\[Blog\]](#)

[DX 향상을 위한 모노레포 개발 환경 구성]

- Frontend 환경에서 Webview와 React Native, 어드민 페이지를 구성하기 위해 반복적인 셋업이 요구되는 문제 발생
 - **Turborepo**를 사용하여 공통 패키지 파일을 생성하고, 다른 워크스페이스에서 이를 참조함으로써 단 한 번의 구성으로 내부 프로젝트가 동일한 환경을 공유하도록 구현 [\[Blog\]](#)
- 모듈의 재사용성과 관리 용이성을 극대화하기 위해 **Barrel exports** 패턴 사용
 - 불필요한 다른 모듈을 가져오기 때문에 성능 저하가 발생 → Next.js의 **optimizePackageImports** 옵션을 도입하여 성능 개선

[Webview Bridge 통신]

- react-native-webview의 단방향 통신 문제를 해결하고 Promise 기반의 선언형 흐름을 보장하기 위해 **webview-bridge**를 도입하고, Turborepo 모듈로 분리해 재사용성을 향상 [\[Blog\]](#)

[Storybook 기반 디자인 시스템 구축]

- 중복 컴포넌트 개발과 팀원간의 디자인 소통비용 증가로 **디자인 시스템 도입 건의**
- 개발 생산성 향상을 위해 소스 코드 레벨에서 커스터마이징이 가능한 **shadcn/ui** 기반으로 디자인 시스템 컴포넌트 개발 및 MDX 문서화



주말내집 - 듀얼라이프 커뮤니티 플랫폼

2022.10 ~ 2025.02

Github : [Link](#) | WEB : [Link](#)

오도이촌을 소재로 커뮤니티 및 농가 거래 중개 웹 서비스

기술 스택 | React, Vite, React Query, Zustand, Sass, AWS EC2, AWS S3, AWS Code-Deploy, Github Actions

[자유, 소개, 홍보 등 카테고리별 게시판 서비스 개발]

- S3에 이미지 데이터가 무분별하게 축적되어 불필요한 용량을 차지하는 문제가 발생
 - 게시글 작성 중 페이지 이탈 시 업로드된 이미지를 삭제하고, 게시글 업로드 시 작성된 이미지와 Image Store를 비교하여 **실제로 사용된 이미지들만 S3에 업로드** 하도록 구현해 불필요한 데이터 삭제 자동화
- Lighthouse와 Performance 패널을 분석한 결과 이미지가 실제 렌더링되는 사이즈 비해 원본 이미지의 크기가 큰 것을 확인
 - **CloudFront + Lambda@Edge**를 사용한 On The Fly **이미지 리사이징 및 캐싱, webp 포맷 변환**을 통해 이미지 용량을 **5MB → 316KB 95% 최적화** 및 동일한 이미지 요청 처리 속도를 **2s → 43ms 97% 감소**시켜 LCP와 TBT 지표 개선
- 프론트엔드에서 aws-sdk로 S3에 직접적으로 이미지를 업로드 및 삭제 처리하는 방식으로 인해 랜섬웨어 공격 발생
 - AWS 환경변수를 클라이언트에서 관리해야 하는 aws-sdk를 사용하지 않고 **Presigned-URL** 방식을 통해 서버와 통신하도록 구현하여 보안 강화
- '좋아요' 기능에 **Optimistic UI 업데이트**를 구현하여 서버 응답을 기다리지 않고 인터페이스 반응을 즉각적으로 반영하도록 구현

[클라우드 서비스 배포 및 CD 적용]

- Github Actions을 사용하여 개발서버와 운영서버를 분리한 프론트엔드 CD 파이프라인 구축

[QA 및 유저 테스트를 통한 서비스 지속 개선]

- 실제 사용자 테스트와 인터뷰를 통해 얻은 피드백을 바탕으로 사용성 개선
- 주기적인 자체 QA를 통한 이슈 파악 및 서비스 향상



Hands Up! - 영상 기반 Q&A 놀이터

2023.07 ~ 2025.02

IOS : [Link](#)

짧은 영상을 통해 쉽고 재미있게 질문하고 답변하며 소통할 수 있는 앱 서비스

기술 스택 | React Native, Next.js, Ky, Expo, Turborepo, Firebase, React Query

[유저 사용성 개선]

- 대량의 데이터를 효율적으로 로드하기 위해 **커서 기반 페이지네이션** 도입
 - 실시간 데이터 변경이 빈번한 메인 스크린에서 커서를 통한 데이터 추적으로 데이터 불일치를 최소화하고, 안정적인 페이지네이션으로 사용자 경험을 향상
- WebView와 Native 간의 데이터 동기화를 위해 **브릿지 모듈을 활용한 Query Client 즉시 업데이트 통신 함수 구현**
- Suspense, Error Boundary를 적용하여 Skeleton Loading 처리 및 에러 핸들링을 선언적으로 처리

[딥링크 및 메타 데이터 관리]

- RN 기반 서비스에서 콘텐츠 공유하기 기능을 도입하기 위해 부분적으로 Webview 페이지 분리
 - URL Scheme을 활용한 **딥링크 기능 구현**: 웹뷰 페이지에서 앱 미설치 유저는 스토어 링크로, 설치된 유저는 앱으로 이동하도록 설계
 - GenerateMetaData를 활용한 **동적 OG 태그 생성**: 콘텐츠 공유 시 링크에 노출되는 내용을 커스터마이징하여 브랜드 경험 강화
 - Webview 배포 전략을 통한 **빠른 서비스 업데이트 대응**: 앱 심사 없이 콘텐츠 변경 및 기능 개선 가능

[콘텐츠 미디어 리스트 관리]

- Swiper 기반의 인스타그램 스토리 형식 미디어 스와이프 컴포넌트 개발
- ScrollView를 사용한 Masonry Layout에서 무한스크롤 데이터 렌더링 성능 저하 발생
 - DOM 가상화가 적용된 RecyclerView 기반의 **FlashList**를 사용하여 렌더링 성능 최적화

Skills

- **Language** : TypeScript, JavaScript
 - **State Management** : Zustand, Recoil, Redux
 - **Testing** : Jest, Storybook, Chromatic
 - **Framework** : React.js, Next.js, React Native, Nest.js
 - **Styling** : Sass, Styled-Components, TailwindCSS
 - **ETC** : Docker, Turborepo, AWS, Github Actions
-

Activities

Techer [테커] / 시니어 멘토 2022.09 ~ 진행중 [Link](#)

- 실리콘밸리 개발자의 **SW 개발자 커리어 그룹**
- 실리콘밸리 한달 살기, 기술 컨퍼런스, 해커톤, 네트워킹 행사 등 다수의 활동 주최 및 경험
- 2024 국회의원 표창장 수상

SW 부트캠프 멘토링

- [테커] 실리콘밸리 SW 부트캠프 멘토 2024.06 ~ 2024.08
 - [디캠프 x 테커] 실리콘밸리 SW 부트캠프 멘토 2023.12 ~ 2024.02
 - [티타임즈 x 테커] 실리콘밸리 SW 부트캠프 멘토 2023.06 ~ 2023.08
-

Education

성결대학교 컴퓨터 공학과
2017.03 ~ 2024.02(졸업)