



# 김민준

## Frontend Developer



Phone

010-9798-3108



Email

ckx9798@gmail.com



Git Hub

ckx9798



Blog

velog.io/@ckx9798

## Skills & Tools

HTML

CSS/SCSS

JavaScript

TypeScript

React

Next.js

Tailwind CSS

Styled Components

Bootstrap

React-Query

Recoil

Zustand

Jotai

## About me

- 재사용성은 효율을 넘어, 협업자를 배려하는 방식이라고 생각합니다.

반복되는 UI 요소를 줄이고 유지보수를 용이하게 하기 위해 재사용 가능한 구조와 공통 설정에 대해 고민 및 학습하며, React 기반의 UI 컴포넌트 라이브러리를 직접 설계·구현해보았습니다.

- 작은 불편을 기술로 해결하고, 더 나은 제품을 만들어가는 과정에 관심이 많습니다.

단순 링크 저장 웹앱에서 사용자의 불편을 발견하고, 브라우저 환경에 더 적합한 경험을 고민한 끝에 직접 크롬 익스텐션으로 확장했습니다.

- ‘처음이라서 못 한다’는 말보다, ‘필요하니 해본다’는 태도를 지향합니다.

React를 처음 배운 지 6주 만에 Zustand 기반 상태 설계를 구현했고, SaveLinks 프로젝트에서는 Google Gemini API를 2일 만에 학습해 도입했습니다.

- 만든 것을 써보게 하는 과정까지가 개발이라고 생각합니다.

직접 개발한 프로젝트를 홍보하여 크롬 익스텐션 유저 10명, UI 컴포넌트 라이브러리 200+ 다운로드를 기록하며 초기 사용자 확보에 성공했습니다.

## Recent Project



SaveLinks

📅 2025.01 - 진행 중

[Extension Link](#)

[Github Link](#)

[Project Link](#)

유튜브, 인스타 등 숏폼 영상을 저장하여 한 페이지에서 관리하는 크롬 익스텐션

- 주요 기능



- 크롬 익스텐션을 통해 비로그인으로도 숏폼 영상을 즉시 저장하고, 저장된 영상을 한 페이지에서 관리 및 시청할 수 있는 기능 제공

- Google Gemini API 기반의 채팅 기능을 통해 링크와 관련된 질문에 실시간으로 응답

- 추가적인 회원가입을 통해 일반 링크, 메모, 숏폼 영상을 저장하고, 멀티 디바이스 간 연동 및 동기화 기능 제공

- 성능 최적화 진행

- 최종 LightHouse 성능 점수 향상 (58 → 98점) / 최대 페인트 요소 LCP 7s → 1.1s 단축

- 코드 스플리팅 기법을 통한 초기 페이지 렌더링 컴포넌트를 선별적 로드

- WebP 포맷 이미지 압축, lazy/eager 로딩 전략, preload 적용

- 텍스트 압축(gzip 방식) 적용으로 2KB 이상 파일 약 30개를 압축, 전송 최적화 달성

- chrome.storage 활용하여 기존 프로젝트와 소통

- 현재 url을 크롬 저장소에 저장 → content.js를 통해 웹으로 메시지 형태로 전달 → 메시지를 읽어 데이터로 활용하는 로직 구현

- API응답이 느리고 백엔드 서버와의 소통이 불가능한 상황 극복

- React Query의 캐싱 기능과 persistQueryClient를 통한 네트워크 요청 최소화를 통해 첫 렌더링 이후 렌더링 속도 개선 (캐싱 후 LCP 3.14s → 0.28s 단축)

- 반복적인 API 요청으로 코드가 중복되는 문제 해결

- Axios Instance와 Interceptor를 구축, 공통 로직을 중앙에서 처리하도록 개선

- 도메인 구매를 통해 사용자 접근성 향상 (xyz 도메인 구매)



## 사용성과 재사용성을 고려한 모달과 버튼 컴포넌트를 제공하는 React UI 라이브러리

### • 주요 기능



- 기본 모달 / 디자인 모달 / 토스트 알림 모달 / 다양한 버튼 등을 제공하는 라이브러리
- 환경 구성 및 배포
  - 단순히 코드 작성에 그치지 않고, **아이디어 발굴부터 설계, 개발, 문서화, 배포**까지 전체 라이프사이클을 주도적으로 관리해보며 **오픈소스 라이브러리의 전 주기 경험**
  - 컴포넌트 외부 노출 구조를 index.ts에서 통합 관리, package.json의 module, exports 설정을 통한 안정적인 사용 환경 구성
- 성능 및 개발자 경험 개선
  - 타 개발자가 사용할 때의 편의성과 재사용성을 고려하여 라이브러리를 설계, **개발자 경험을 중심으로 설계하는 시야**를 갖추게 되었고, **협업 시에도 명확하고 재사용 가능한 코드**를 작성하는 기준체득
  - 공통된 UI 패턴을 컴포넌트 단위로 분리하고, 스타일은 inline style을 유지하여 재사용성과 유지보수성을 높이는 동시에 렌더링 성능 및 번들 크기를 최적화
  - 사용자 입장에서 라이브러리를 더 쉽게 활용할 수 있도록 상세한 사용 가이드와 예제 코드를 제작하여 GitHub에 공개했고, 이를 통해 다른 개발자들의 피드백을 반영해 지속적으로 개선



## 원데이클래스, 액티비티 등 각종 체험을 등록하고 예약기능을 제공하는 웹사이트

### • 실시간 데이터 반영을 통한 사용자 혼란 해소



- 예약 및 체험 목록이 즉시 반영되지 않아 발생하는 혼란을 해결하기 위해 React Query의 **invalidateQueries**를 활용하여 **항상 최신 데이터 유지**, 사용자 경험 개선
- 끊임 없는 콘텐츠 제공
  - 페이지 전환 시 콘텐츠 로딩 지연으로 인한 이탈율 문제 해결을 위해 **무한 스크롤 및 로딩 스피너** 도입, 사용자 이탈 감소 및 UX 개선
- 사용자 피드백 강화
  - 서버 에러 및 페이지 이동 시 피드백 부족으로 인한 불편을 해결하기 위해 **React-Toastify**를 도입, 실시간 알림 제공 및 사용자 반응성 향상
  - 관련 로직을 공통 커스텀 훅으로 추출하여 중복 코드 제거 및 유지보수성 향상
- 네트워크 안정성 확보
  - **CORS 문제** 및 통신 시 성능 저하 이슈를 해결하기 위해 **프록시 서버 도입**, 네트워크 부하 감소 및 안정적인 통신 환경 확보
- 전역 상태 관리를 통한 렌더링 최적화
  - 프로필 이미지 변경 시 불필요한 리렌더링 발생 문제를 해결하기 위해 **Zustand**를 도입, 전역 상태 효율적으로 관리하여 렌더링 성능 최적화

## Education

가톨릭대학교 국제학부 | 졸업 📅 2017.03 - 2021.02

코드잇 부트캠프 프론트엔드 과정 | 수료 📅 2024.06 - 2024.12

모던 자바스크립트 Deep Dive 스터디 📅 2024.12 - 진행중

코드잇 단기 인턴 📅 2025.05 - 2025.06.15

## Certification

정보처리기사 필기 합격 | 한국산업인력공단 📅 2025.03

컴퓨터 활용능력 1급 | 대한상공회의소 📅 2020.08

GTQ 2급 | 한국생산성본부 📅 2020.07