

# 류기현 Frontend Developer

v1.0 단계의 신규 제품에서 프론트엔드 구조 설계와 구현을 맡았으며, type-safe UI abstraction과 ECG 그래프 기반 인터랙션 구현에 강점이 있습니다.

khryu0610@gmail.com GitHub Blog

## 경력

메디컬에이아이 서울

2024.11 - 현재

소프트웨어그룹 프로덕트본부 FE팀 · Frontend Developer

## 핵심 역량

### 초기 제품 구조 설계

신규 제품의 v1.0 단계에서 기술 스택, 구현 구조, 공통 패턴을 정하는 데 참여했습니다.

### 복합 입력 인터랙션 설계

ECG 그래프 위에서 터치·마우스·키보드 입력이 함께 동작하는 인터랙션 레이어를 설계하고 구현했습니다.

## 대표 프로젝트

### 판매관리 시스템 메디컬에이아이

2025.09 ~ 2025.12

Frontend Developer · v1.0

회사의 자산, 매출, 거래처를 관리하는 사내 경영관리 시스템

React TypeScript TanStack Query Zustand Zod React Hook Form

- v1.0 초기 개발 참여
- 대형 Form/Filter 패턴 표준화
- 파일 다운로드 및 클라이언트 암호화 처리

### 주요 기여

- 대형 Form과 Filter 구현이 화면마다 반복되었고, 검증/에러 처리 방식이 분산될 수 있었습니다.  
→ 사내 디자인 시스템 위에 react-hook-form 래퍼와 제네릭 Form Field 컴포넌트를 설계했습니다.  
✓ 폼 구현의 재사용성과 타입 안정성을 높이고, 화면 간 검증/에러 처리 방식의 일관성을 확보했습니다.  
사내 디자인 시스템(maiui) 기반으로 Form abstraction 설계 Type-safe 인터페이스 제공
- 대용량 파일 다운로드와 암호화 처리 과정에서 서버 부하와 운영 환경 차이를 고려해야 했습니다.  
→ 스트리밍 다운로드 로직을 구현하고, zip 파일 암호화를 클라이언트에서 수행하도록 설계했습니다.  
✓ 서버 부하를 일부 클라이언트로 분산하고, 윈도우/맥에서 동작하는 다운로드 흐름을 구현했습니다.  
대용량 스트리밍 파일 다운로드 로직 구현 윈도우/맥 기본 압축풀기 기능 지원
- 변경이 잦은 개발 환경에서 API 상태에 따라 프론트엔드 개발과 코드 리뷰가 자주 끊길 수 있었습니다.  
→ API mocking coverage를 높은 수준으로 유지해 프론트엔드 검증 경로를 확보했습니다.  
✓ 백엔드 변경과 무관하게 개발과 코드 리뷰를 이어갈 수 있는 환경을 유지했습니다.

API mocking coverage 90%+

## 고객지원 시스템 메디컬에이아이

2025.07 ~ 2025.08, 2026.02 ~ 2026.03

Frontend Developer · v1.0, v1.1

고객이 카카오톡 채널로 문의를 등록하고, CS 담당자가 답변할 수 있는 웹 애플리케이션

React TypeScript TanStack Query Zustand

- v1.0 초기 구조 설계 및 구현
- v1.1 개선 요구사항 검토 및 일정 산정
- 권한 체계와 멀티 인터페이스 구조 설계

### 주요 기여

- 1 고객용/직원용, 모바일/데스크톱용 인터페이스를 하나의 제품 안에서 함께 관리해야 했습니다.  
→ 하나의 레포에서 여러 인터페이스를 관리할 수 있도록 공통 구조와 분리 기준을 설계했습니다.  
✓ 제품 변형을 별도 프로젝트로 분리하지 않고도 일관된 개발 구조를 유지할 수 있었습니다.  
v1.0 단계부터 초기 구조 설계 참여 모노레포 구조 설계
- 2 직원용 기능에는 조직 정보 기반 접근 제어가 필요했고, 별도 조직관리 시스템과의 연동이 요구되었습니다.  
→ RBAC를 구현하며 기획/Auth/Backend 팀과 권한 정책 및 조직관리 시스템 연동 방식을 조율했습니다.  
✓ 조직 정보 기반 접근 제어가 가능한 직원용 기능 구조를 마련했습니다.  
기획/Auth/BE와 권한 체계 조율
- 3 변경 요구사항이 늘어나면서 개발 범위와 일정의 불확실성이 커졌습니다.  
→ v1.1 개선 단계에서 요구사항을 개발 단위로 구체화하고 일정 산정에 참여했습니다.  
✓ 개선 요구사항을 우선순위와 구현 범위 중심으로 정리할 수 있었습니다.
- 4 빠른 모바일 프로토타이핑을 위한 적절한 UI 기반이 필요했습니다.  
→ Headless UI 라이브러리를 검토하고 Vault 기반 Drawer 도입 가능성을 리서치·공유했습니다.  
✓ 모바일 UI 프로토타입 제작 속도를 높이고, 동시에 탁월한 Drawer UX를 제공했습니다.

## ECG-GRAPH Ruler for Tablet 메디컬에이아이

2025.02 ~ 2025.04

Frontend Developer · tablet adaptation

심전도 그래프에 특정 지점을 표시하여 시간·진폭 간격을 측정하는 태블릿용 도구

React TypeScript D3

- 데스크톱 중심 기능의 태블릿 환경 재설계
- 터치 기반 제스처 판별 로직 설계
- 측정 UI 및 캘리퍼 기능 구현

### 주요 기여

- 1 태블릿에서는 모든 입력이 손가락 터치 하나로 시작되어, 점 생성 / 줌 이동 / 정밀 측정의 의도를 시작 시점에 구분할 수 없었습니다.

→ 터치 시작 시 타이머(800ms)와 시작 좌표를 기록하고, 이후 이벤트의 시간·이동량으로 의도를 판별하는 구조를 설계했습니다.

✓ 서로 다른 동작이 간섭하지 않는 터치 기반 측정 경험을 구현했습니다.

시간 × 이동량 기반 제스처 판별 구조 설계 임계값 튜닝

2 d3-zoom이 자체 이벤트 시스템으로 핀치 줌과 팬을 처리하고 있어 커스텀 인터랙션과 충돌이 발생했습니다.

→ d3-zoom filter에 현재 상태를 주입해 커스텀 인터랙션 중에는 d3가 이벤트를 소비하지 않도록 제어했습니다. 또한 정밀 측정 중 d3 View Point가 밀리는 문제를 발견해 측정 후 좌표를 기존 상태로 복원하여 충돌을 해소했습니다.

✓ 줌/팬과 정밀 측정 기능을 동시에 유지하면서도 입력 충돌을 줄였습니다.

d3-zoom filter에 현재 상태 주입 정밀 측정 중 transform 적용 차단 및 롤백 처리

3 사용자가 파형이 아닌 영역을 터치해 오입력이 발생할 수 있었습니다.

→ 파형 근방에만 유효한 터치 범위를 두는 hit testing 로직을 적용했습니다.

✓ 사용자 입력이 의도한 요소에 더 정확히 반응하도록 개선했습니다.

4 드래그 가능한 측정 요소들이 늘어나면서 동일한 동작 패턴을 여러 곳에서 반복 구현할 위험이 있었습니다.

→ 드래그 라이프사이클을 흑으로 모듈화해 캘리퍼와 버튼 그룹 등 여러 요소가 재사용할 수 있도록 했습니다.

✓ 드래그 처리 방식의 일관성을 높이고 재사용 가능한 인터랙션 패턴을 만들었습니다.

## 개인 프로젝트

### 지도 API 컴포넌트화 라이브러리 수상

6인

#### SSAFY 전국 지역 대상 발표

카카오맵 API를 Vue.js 컴포넌트로 추상화한 오픈소스 NPM 패키지 라이브러리

- 패키지 빌드/배포, 컴포넌트 설계/개발, Docs 개발/작성
- ESLint 커스텀 룰 개발
- 라이브러리 로컬 테스트 환경 구성 방법을 연구하고 팀 세션으로 공유

오픈소스 패키지 설계, 배포, 문서화, 개발 프로세스 정착 경험을 보여주는 사례

### 총학생회를 위한 투표 시스템

2인

투표/회의가 필요한 집단에서 사용할 수 있는 웹 애플리케이션

- 실시간 이벤트(SSE) 및 Broadcast 처리
- SSE 데이터와 실제 데이터 정합성 유지 로직 구현

실시간 상호작용과 의사결정 흐름 중심 제품에 대한 관심을 보여주는 프로젝트

## 학력

---

### 연세대학교

문과대학 중어중문학과  
2018.03 - 2023.08

서울

## 교육

---

### SSAFY 삼성청년소프트웨어 아카데미

비전공 Java반  
2023.07 - 2024.06

서울

### 멋쟁이사자처럼 웹 프론트엔드 스쿨

프론트엔드 중점 과정  
2022.07 - 2023.01

온라인

## 기술

---

Frontend [React](#) [TypeScript](#) [Next.js](#) [Zustand](#) [TanStack Query](#) [React Hook Form](#) [Zod](#) [D3](#)  
Testing [Vitest](#) [Playwright](#) [Mock Service Worker](#)  
Tooling [Vite](#) [Yarn](#) [pnpm](#) [GitLab CI/CD](#)  
디자인 [Figma](#)  
협업 [Jira](#) [Notion](#)  
자격증 [SQLD](#)  
영어 [OPIc IM2](#)