

정혜원 | Jeong Hyewon

언어 공부에 대한 집착, 깔끔한 코딩에 대한 갈망, 최상의 결과물을 주는 팀워크



Contact.

- ✓ Email. jaake0704@gmail.com
- ✓ Phone. 010-8954-4608

Channel.

- ✓ Portfolio . [개발하는 너굴맨 포트폴리오](#)👏
- ✓ GitHub . <https://github.com/wonny-won>
- ✓ Blog . <https://dev-raccoon-man.tistory.com/>

저는 _____ 한 개발자입니다.

- 개발시 유지보수, 성능, 확장성, 안정성, 보안을 고려합니다.
- 기술의 동향을 파악하고 새로운 기술을 공부하는 것을 즐깁니다.
- 기존의 기술을 다양한 측면으로 깊게 공부하는 것을 즐깁니다.

저는 이렇게 일해요!

📌 사용자 경험 향상 측면에 많은 열정을 가지고 있어요.

사용자가 경험하기 전에 직접 테스트하며 어색한 사용 흐름이나 불편함을 발견합니다.
해당 지점이 기획 혹은 디자인 의도와 일치하는지 확인 후 필요시 개선점을 제안합니다.
개선점 제안을 통해 더 나은 사용자 경험으로 이어질 수 있도록 협업합니다.

📌 성능 최적화는 늘 작은 곳에서 시작되죠.

저의 성능 최적화는 컴포넌트 단위 리렌더 관리에서 시작합니다.
조건부 렌더링 등을 통해 불필요한 렌더를 최소화 하며 전체 렌더 성능을 높여왔습니다.
또한 렌더링 이외에도 네트워크, 메모리, 레이아웃 등 다양한 성능 이슈를 주도적으로 탐구하고 개선해온 경험이 있습니다.
이런 경험은 앞으로도 실질적인 성능 개선 아이디어로 기여 할 수 있을 것 입니다.

📌 러닝 커브가 빠르며, 프로젝트의 핵심 구조를 빠르게 파악합니다.

프로젝트에 완벽히 녹아들기 위해 가장 필요한 능력은 '빠른 학습 속도와 프로젝트의 핵심 파악'이라고 생각합니다.
저는 프로젝트의 핵심을 파악해 낯선 환경 속에서 단기간 내 프로젝트 맥락에 적응해 실질적 기여를 만들어냅니다.
또한 빠른 러닝커브는 핵심 파악을 좀 더 빠르게 할 수 있도록 합니다.

보유 기술

Frontend

현재 업무에 사용중 혹은 사용했던 기술들입니다.

- HTML / CSS
- JavaScript / TypeScript
- React.js / Next.js / svelte
- apollo-client / axios / websocket / webworker / react-query
- recoil / context-api
- emotion / styled-component / tailwind

회사 프로젝트

를 클릭하시면 상세한 프로젝트 설명 또는 제작한 프로젝트를 보실 수 있습니다.

포페런츠

엔지니어 | FE팀 개발자

2025.04.10 ~ 2025.09.30

✓ Tech stack

- typescript
- next.js
- tailwind
- emotion
- context-api
- ...

액티브 시니어를 위한 여행 이커머스 서비스 

✓ 개발 시스템 및 문서화 체계 구축 및 서비스, 어드민 제작 및 런칭

- 아너드 자체 웹 서비스 프론트엔드 E2E 단독 개발 및 런칭
 - 신한 X 아너드 제휴 서비스 프론트엔드 E2E 단독 개발 및 런칭
 - 모노레포 구축
 - Turborepo + pnpm 기반 워크스페이스 설계
 - 서비스/공용 모듈을 분리해 구조화
 - 유지보수, 재사용성을 고려한 데이터/UI 컴포넌트 분리
 - 확장성을 고려한 컴포넌트 설계
 - 전체 삭제 데이터 프로바이더 → 다중 프로세스 처리 프로바이더로 확장
 - 서비스 런칭 전 과정(기획, 디자인, 개발)에 참여해 E2E 개발 프로세싱 리딩
 - 개발 시스템 및 문서화 도입
 - 자체 웹 서비스 릴리즈 및 문서화 체계 구축
 - 개발팀 주요 관리 문서 및 프로세스 도입으로 업무 공유를 원활히 할 수 있도록 기여
 - 회의록 / 릴리즈 문서 / 프로젝트 별 진행 상황 / QA 문서 / 온보딩 및 인수인계 문서 등등
- 신규 어드민 서비스 제작
 - 실무진 니즈를 반영한 핵심 기능 기획 및 구현
 - 기존 리소스를 활용한 MVP 빠른 제작 및 배포로 업무 효율성 극대화
 - 개발 일정 리드 및 협업 조율
- 기존 어드민 유지보수 및 신규 기능 추가
 - 실무진 니즈를 반영한 신규 기능 기획 및 구현
 - 각종 핫픽스 수정

메이아이

엔지니어 | FE팀 개발자

2025.02.10 ~ 2025.04.09

✓ Tech stack

- typescript
- react.js
- styled-component
- vega-chart
- context-api
- ...

LG 전자, 현대 백화점에서 사용하는 방문객 데이터 분석 서비스 

- ✓ 온보딩 기간동안 새로운 대시보드와 신기능을 릴리즈
- ✓ 타회사에 스카웃 되어 이직
 - 모노레포 사용경험 및 깃버전 관리 경험
 - 코호트 분석 페이지 제작
 - 데이터 스키마 프로토타이핑 및 신규 페이지 개발
 - 고객사 커스텀 기능 개발
 - 버그 추적 및 핫픽스 배포
 - 알림 팝업 신규 개발 및 배포
 - 각종 신기능 릴리즈 및 핫픽스 배포
 - 대시보드 퍼포먼스 안정화
 - 단일 책임 원리 적용으로 데이터/UI 컴포넌트 분리
 - 렌더 성능최적화
 - 렌더 조건과 visibility를 이용한 reflow / repaint 제거
 - 네트워크 데이터 요청 최적화 진행
 - 세마포어 기반으로 동시 요청 수 5개 제한으로 API 서버 과부하 방지
 - 요청 조건 제어를 통해 불필요한 네트워크 요청을 제거하고, 네트워크 요청 비용을 절감
 - 신기능 개발시 확장성을 고려한 컴포넌트 설계
 - 전체 데이터 삭제 프로세스 프로바이더 → 다중 삭제 프로세스로 역할 확장

팀스톤

주임 | FE 개발자

2023.08 ~ 2024.08

✓ Tech stack

- typescript
- svelte
- websocket
- webworker
- tailwind
- echart
- svelte-i18n
- tabulator
- xlsx
- sheetjs
- yup
- vite
- ...

삼성, 메이저 은행권에서 사용하는 시스템 모니터링 서비스 

- ✓ 기존 ontune v4를 웹버전으로 컨버팅해 v5로 출시
- ✓ 성과를 인정받아 최단기 주임 승진
 - 시스템 성능 분석 위젯 개발
 - 총 8개의 성능 분석 위젯을 설계 및 개발 및 커스텀 레이어 팝업 기능 구현
 - 위커 스레드를 활용한 데이터 연산과 렌더 분리로 성능 최적화
 - 반응형 UI 적용 및 유지보수성 향상을 위한 구조 개선
 - 테스트를 기반으로 사용자 친화적인 UI/UX 개선 의견 제시 및 반영
 - 위젯 성능 개선
 - 렌더 조건과 visibility를 이용해 리플로우 최소화를 통한 렌더 최적화
 - requestAnimationFrame 적용으로 애니메이션 메모리 누수 해결
 - 메모리 지속 증가(50,256K) → 안정화 후 28,416K로 개선
 - CPU 사용률 10.9~80.4% → 3.9~5.1%로 감소
 - CSS 최적화를 통한 CPU 사용량 감소 및 개선
 - 워커 스레드를 활용한 렌더 최적화 진행
 - 차트 모듈 트리 쉐이킹으로 번들 크기 축소

- 이벤트 디바운싱 적용으로 불필요한 연산 최소화
- Host 및 제품 튜닝 페이지 개발
 - Host 및 제품 튜닝 관련 총 6개 페이지 개발 및 기본 데이터 세팅 구현
 - 렌더 범위 최적화
 - 기존 전체 리렌더 구조 → 변경된 데이터만 다시 렌더링하도록 리렌더 범위 최소화
 - 데이터 관련 함수 안정성 검증
- 시스템 성능 분석 페이지 개발
 - 시스템 성능 분석 관련 총 2개 페이지 신규 개발
 - 위젯 데이터 지표 선택을 위한 커스텀 레이어 팝업 기능 설계 및 개발
 - 사용자 편의성을 고려한 인터랙션 설계
 - 메트릭 데이터 스케일 처리 공통 유тиль 함수 개발로 코드 일관성 확보
- 공통 컴포넌트 및 글로벌 함수 개발
 - ReportBaseChart
 - 트리쉐이킹을 위해 기존 BaseChart를 활용해 개발
 - default상태를 지정해 데이터와 차트타입만 넘겨줘도 그려지도록 개발
 - 높은 자유도를 보장해 다양한 커스텀이 가능하도록 개발
 - WidgetMideaQuery

윈도우 미디어 쿼리를 이용 할 수 없는 상황에서 반응형 UI를 위해 제작하게 된 공통 컴포넌트

 - ResizeObserver api 이용
 - 스타일 자유도를 높여 언제든 모드 추가가 가능하도록 개발
 - BarChart / BarChartTooltip

Common Grid 내부에서 사용되는 BarChart 및 BarChartTooltip.

 - 라이브러리 없이 직접 구현
- 다국어 처리 경험
- 사원으로 입사해 성과를 인정받아 24년도 하반기 주임으로 승진
- 모든 프로젝트 1차 버전 배포 후 1차 QA 참여

딩코

FE 멘토, 개발자
2021.12 ~ 2023.04

✓ Tech stack

- typescript
- Next.js
- apollo-client
- emotion
- yup
- recoil
...

인프런에도 올라온 온/오프라인 부트캠프 ☺

- 코드캠프 온/오프라인 부트캠프 학습자료 제작
 - 제작해온 학습자료
 - 각 주제별 핵심 내용이 꼼꼼히 정리 되어있는 학습자료
 - 수강생에게 노션 장인의 자료로 평가받은 학습자료
(링크에 자료평 캡쳐본 있습니다.)
- [인프런 X 코드캠프] & 코드캠프 온/오프라인 부트캠프 프론트엔드 멘토
 - 프론트엔드 수강생 일정 관리 및 질의 응답
 - 수강생 과제 및 포트폴리오 로직 피드백

- 로직 피드백은 성능상 혹은 유지보수상 조금 더 나은 방향으로 제시
- 코드캠프 광고 페이지 개발
 - SEO를 위한 시멘틱 태그 적용
 - 반응형 웹 제작
- 딩코 사이트 어드민 & 유저 백업 개발
 - 유저 로그인 로그 데이터 개발

개발 문제 해결 사례

 를 클릭하시면 상세한 개선 사례 설명을 보실 수 있습니다.

1. 모노레포 배포 에러 해결 사례

✓ [문제]

Vercel 배포 실패 → 빌드 단계에서 모듈을 찾지 못해 에러 발생

✓ [원인]

1. Vercel은 빌드 타겟 디렉터리 기준으로 의존성 해석
2. 모노레포 + npm 워크스페이스 환경에서 Vercel 배포시 모듈 호이스팅으로 인한 패키지 경로 인식 실패
 - 모듈 호이스팅으로 node_modules가 상위 루트 경로에서만 생성
 - 빌드 할 대상은 하위 폴더이기 때문에 루트의 모듈을 읽어올 수 없음

✓ [해결]

가장 확실한 방법인 패키지 매니저를 pnpm으로 변경 후 워크스페이스를 pnpm으로 변경

- npm은 호이스팅 제어 설정 복잡도 증가해 직접적으로 제어가 어려운 반면 pnpm은 패키지 단위로 node_modules 관리

✓ [결과]

패키지 호이스팅 문제를 제거해 빌드 에러를 근본적으로 해결 후 배포 성공

- 패키지 단위로 node_modules가 분리되며 하위 루트에 node_modules를 생성해 빌드에러 해결

✓ [비고]

추후 동일한 이슈 재발 방지 및 사이드 이펙트에 도움이 될 수 있을 것 같아 트러블 슈팅 문서화 완료

1. 위젯 메모리 누수를 개선한 사례

✓ 특이사항

- 데이터 2초 주기 변경
- 화면 2초 주기 리렌더

✓ [문제]

특정 위젯의 자바스크립트 메모리가 지속적으로 증가하는 메모리 누수 상황 발견

- CPU 사용률 10.9~80.4%
- 자바스크립트 메모리 지속 증가(50,256K)

✓ [원인]

애니메이션 구현을 위해 사용한 interval이 2초에 한번씩 호출되어 동시에 여러 interval 생성 후 1:1 clear 되지못해 발생한 문제

✓ [해결]

requestAnimationFrame api를 이용한 애니메이션으로 재구현

- 메모리 누수 판단 과정
 - 브라우저 작업 관리자로 JS 메모리 및 CPU 사용률 추적
 - JS 메모리가 GC 이후에도 안정화 되지 않고 지속적으로 증가 시 메모리 누수 판단
- 누수지점 파악 과정
 - 브라우저 DevTools → Performance 탭에서 메모리 타임라인 녹화
 - JS 힙 메모리가 증가후 제대로 해제되지 않은 지점 확인
 - 브라우저 DevTools → Memory 탭에서 Allocation instrumentation on timeline 실행
 - 메모리 막대가 해제되지 않은 구간의 타겟 오브젝트 확인
 - 브라우저 DevTools → Memory 탭에서 프로그램 시작 초기 힙스냅샷 녹화
 - 일정 시간 이후 힙스냅샷을 한번 더 녹화 후 두 스냅샷에서 타겟 오브젝트 비교
 - Retained Size 증가되어 있다면 누수 포인트로 판단

✓ [결과]

1. 메모리 사용량이 개선되어 누수가 개선되고 더불어 CPU 사용량도 개선되며 화면이 많이 부드러워짐
 - CPU 사용률 10.9~80.4% → 3.9~5.1%로 감소
 - JS 메모리 지속 증가(50,256K) → 안정화 후 28,416K로 개선
2. GPU가 다소 많이 사용되는 것 같다는 일부의 평가가 존재

✓ [비고]

현재 본인 지식선에선 리소스를 더이상 줄이는 것이 무리라 판단해 애니메이션 제거를 요청

3. 불필요한 전체 리렌더로 인한 페이지 성능저하 개선 사례 ☺

✓ [문제]

1. 사이드 패널의 active를 수정 시 불필요한 반응성 전파로 광범위한 DOM 업데이트되어 성능 저하 발생
2. 그리드에 데이터가 많아질수록 버벅임과 메모리 사용량이 크게 증가

✓ [원인]

1. 반응성 부여 시 리렌더를 실행 할 조건을 부여하지 않아 변수 중 하나라도 변경되면 무조건 전체가 업데이트되는 구조
2. 페이지 내부에 필요한 대부분의 변수(상태)가 외부 파일에 하나의 객체로 묶여 있다
 - 변수 객체중 하나의 프로퍼티라도 변경되면 해당 객체를 참조하고 있던 모든 컴포넌트에서 반응성이 전파되어 DOM 업데이트
 - grid 생성에 필요한 인스턴스 객체도 해당 객체에 포함되어있어 패널 밖 그리드까지 불필요한 DOM 업데이트 발생

✓ [해결]

1. 반응성(\$) 부여 시 필요한 값만 참조 할 수 있도록 조건을 설정해 렌더 시점 분리
2. 하나의 객체로 관리되던 변수(상태) 중 반응성이 필요 없는 값은 컴포넌트로 퀘 변수로 이동해 반응성 전파 범위 축소

✓ [결과]

대량 데이터 환경에서도 버벅임과 메모리 증가 완화, 변수(상태)의 선언을 컴포넌트 내부로 일부 옮겨 가독성 개선

✓ [비고]

사이드 프로젝트



를 클릭하시면 상세한 프로젝트 설명을 보실 수 있습니다.

Huggie

side | 팀
프론트엔드 개발자

나만의 맛집을 컴팩하게 모아볼 수 있도록 만든 웹앱

- ✓ 팀원분들의 각자의 개인사정으로 중단된 프로젝트

- 🍎 **Huggie 소개 노션**

- 지도 페이지 구현
- 카카오 소셜 로그인 구현

- **Tech stack**

typescript, React.js, vite, axios, styled-components, react-query, kakao-map, jest ...

Fitable

side | 개인
프론트엔드 개발자

이직을 위한 포트폴리오와 이력서를 첨삭 받을 수 있는 사이트

- ✓ 배포 후 일정기간 이후 비용 문제로 인스턴스 삭제

- 👩‍💻 **Fitable 소개 노션**

- dompurify를 이용한 XSS 보안 강화
- 검색엔진 최적화를 위한 시멘틱 태그 적용
- 배포환경 및 운영체제 통일을 위한 Docker 사용
- FileReader, react-hook-form, hooks 분리, 캐시등을 이용한 프로젝트 성능 고려

- **Tech stack**

firebase, typescript, Next.js, react-query, react-hook-form, react-quill, dompurify, react-slick, antd, portone, emotion, yup, AWS, docker ...

여기까지 읽어주셔서 감사합니다.

추가로 궁금하신 점이나 필요하신 서류가 있으시다면

jaake0704@gmail.com 또는 010-8954-4608로 연락 주시기 바랍니다.

좋은 하루 보내세요. ☀