최재희

UX 엔지니어 · 프론트엔드

메일	jenn0.6n@gmail.com
GitHub	github.com/0jenn0
포트폴리오	작업 인덱스 · 라이브 데모
현재	사인팟 프론트엔드 개발자 (2025.10 — 현재)

1 소개

디자이너 없이 디자인과 프론트엔드를 한 흐름으로 끝내는 개발자.

병원 사이트 20개를 1인이 운영할 수 있는 구조를 만들었고 교육 SaaS의 5개 역할 정보구조를 기능 하나 빠뜨리지 않고 전면 교체했습니다.

화면을 설계할 때는 데이터의 위계부터 정합니다. 무엇을 가장 먼저 읽게 하고 무엇을 접어 두고 무엇을 끝까지 남길지 정한 뒤에 시각 요소를 배치합니다. 화면은 멈춘 상태가 아니라 움직이는 상태로 설계합니다. 완성도는 정적 시안이 아니라 실제로 쓰는 흐름에서 갈린다고 생각합니다.

2 기술

구현	TypeScript · React · Next.js(App Router) · Tailwind CSS v4
설계	데이터 위계에서 출발하는 정보구조 · 디자인 토큰 3계층 · 컴포넌트 라이브러리 · Storybook
인터랙션	모션 토큰(ease · duration · stagger) · 스크롤 반응 UI · GSAP · Lenis · prefers-reduced-motion
품질	명도 대비 · 키보드 동선 감사 · 데이터 상태 명세 · Lighthouse로 성능 측정 · 개선
협업 도구	Figma(MCP 연동) · Turborepo 모노레포 · Claude Code 에이전트 워크플로

3 경력

사인팟 (Signpod) · 프론트엔드 개발자

2025.10 — 현재 · 병원 브랜딩 · 간판 업체. 병원 웹사이트와 사내 운영 도구 담당

병원 랜딩사이트 모노레포

설계 · 운영 주도 · 사이트 20개

문제

사이트마다 따로 구현하면 수정 1건이 작업 20번으로 늘어남. 화면 구간(모바일 · 태블릿 768 · PC 992)은 디자이너와 처음부터 정했지만 일정상 모바일 시안이 늦거나 없는 섹션이 더 많아 사이트마다 모바일 품질이 달라질 위험.

대안 검토

① 화면마다 그때그때 판단: 기준이 사람마다 달라짐 ② 판단 기준 문서화 + 화면 폭별 측정 자동화 + 검증된 배치를 공용 컴포넌트로 재사용: 시안 없이도 같은 기준 · 숫자로 확인 → ② 채택.

실행

공통 토큰 · 컴포넌트를 공유 패키지 9개로 나누고 앱에는 오버라이드만 남김. 모바일 시안이 없을 때의 점검 기준을 스킬 문서로 정하고 375 · 768 · 1440 폭 측정을 스크립트로 자동화. 모바일 시안이 있던 병원의 브레드크럼 배치를 공용 컴포넌트 옵션(stackOnMobile)으로 올려 시안 없는 병원에 재사용. clamp() fluid 스케일과 컨테이너 단위로 타이포 · 간격이 끊김 없이 줄어들게 함.

결과

새 사이트는 공유 패키지 위에서 시작하고 모바일 시안이 없는 섹션도 같은 기준으로 점검. 43개 파일에서 prefers-reduced-motion 대응.

성능

LCP를 늦추던 1.68MB 팝업 이미지를 찾아 업로드할 때 webp로 압축하는 단계를 넣음. 51MB 히어로 영상은 523KB 프리뷰를 먼저 띄우고 2.1MB 루프로 나중에 교체. 모바일 FCP 2.4초 → 1.2초 (자체 측정).

병원 홈페이지 빌더

2026.10.29 — 11.04 · 기획 · 디자인 · 구현 단독 · 7일

문제 병원 앱 16곳에 같은 역할의 섹션이 따로 구현돼 약 187개 파일로 흩어짐. 같은 이름의 섹션도 컨테이너 폭 39가지 · 세로 여백 9가지 · 제목 규칙 10가지로 서로 달랐음.

대안 검토 ① 병원별 코드 유지: 원장님이 고를 수 있는 단위가 안 생김 ② 섹션 단위로 통합하고 색 · 문구는 변수로 → ② 채택. 컴파운드 컴포넌트는 편집 축이 여러 개인 메인 배너에만 적용(나머지 11종은 축이 하나라 API만 늘어남).

실행 12종 52개 레이아웃으로 정리하고 메타 카탈로그와 컴포넌트 등록부를 분리해 편집기 옵션 · 입력칸 · 갤러리가 한 곳에서 나오도록 구성. Figma MCP(use_figma)로 104프레임을 덤프해 Playwright로 실측 대조(±2px).

랜딩 '이미 만들어져 있어요'를 첫 문장으로 기획 · 디자인. 매직링크를 열면 원장님 병원 이름의 사이트가 느낌 4종으로 바로 보이도록 스크린샷 · iframe 대신 서버에서 그리고(SSR) 기기 전환 스켈레톤을 제거해 로딩 기색을 없앰.

결과 52개 중 42개 시안 일치(나머지 10개는 사유 기록). 메인 배너는 프리셋 4개에서 축 5개 조합 432가지. 공개 페이지는 요청마다 저장소를 읽는 서버 렌더(SSR)로 승인 즉시 반영.

비주얼캠프 · 웹 프론트엔드 인턴

2024.12 — 2025.06

교육 관리 시스템의 원장 전용 어드민 사이트 UI/UX 설계 및 프론트엔드 개발 담당.

사용자(원장) 니즈를 파악하기 위한 직접적인 커뮤니케이션 및 요구사항 분석 수행.

실사용자 피드백을 바탕으로 한 UI 개선 및 사용성 최적화 작업 진행.

디자인부터 개발 · QA 업무 경험을 통한 프로덕트 전체 생명주기 이해.

카카오 이모티콘 작가 · 후즈

2022 — 2024

카카오 이모티콘샵에 6종 정식 출시(움직이는 이모티콘 포함). 기획부터 캐릭터 · 프레임 애니메이션까지 개인 작업.

출시 너는 내 몽몽이 · 댕댕미 폭발! 댕댕커플 · 쮸롱이는 귀엽쮸롱 · 공부할 시간이야! 곰도리쌤티콘 등 6종.

움직임 정해진 프레임 수 안에서 한 컷씩 그리며 애니메이션 원칙을 손으로 익힘. ease-in · ease-out · 타이밍과 간격 · 예비동작(anticipation) · 늘이고 누르기(squash & stretch) · 후속동작(follow-through).

연결 같은 동작도 프레임 간격 하나로 경쾌해지거나 무거워진다는 감각을 얻음. 지금은 UI 모션 토큰(ease · duration · stagger)을 정하고 상태가 바뀔 때 무엇을 먼저 움직일지 정하는 기준으로 씀.

UX로 확장 채팅창의 작은 크기에서도 감정이 바로 읽혀야 했기에 사람이 화면에서 무엇을 먼저 보는지 고민하기 시작함. 거절돼도 이유를 알 수 없던 작가 시절의 문제가 사이드 프로젝트 치즈콘으로 이어짐.

4 프로젝트

SolBridge · 수학 학원 AI 채점 플랫폼

외주 · 2026.06 — 2026.10

㈜교육공간. 3인 팀에서 디자인을 단독으로 맡고 리디자인 주도. 서비스 2개 계열과 웹 · 태블릿 앱.

문제 기능을 쓸 때마다 그 기능이 어디 있는지 다시 찾아야 하는 구조. 기본 UI 관례를 지키지 않은 게 원인이라 색 · 컴포넌트 교체로는 풀리지 않음.

대안 검토 ① 비주얼만 교체: 탐색 문제가 그대로 남음 ② 구조 재설계: 기능이 소리 없이 빠질 위험 → ②를 택하되 시작 전에 기존 기능 명세 470개를 모두 뽑아 재설계 기준으로 삼음.

실행 5개 역할(학생 · 강사 · 기관 관리자 · 마스터 관리자 · 인증) 53페이지를 역할 중심 정보구조로 재설계. 공통 컴포넌트 225개로 라이브러리 구축.

결과 기능 누락 QA 0건(AI 투입 3차 QA 포함). 전면 교체 뒤 기능 위치 문의 0건.

상태 · 접근성 로딩 · 빈 목록 · 조회 실패 같은 데이터 상태와 복구 흐름을 섹션 단위로 명세. 좁은 화면과 키보드 동선까지 다루고 WCAG AA 명도 대비 18쌍을 모두 통과한 감사 기록을 남김.

디자인 · 코드 연결 원시 · 의미 2계층 Figma 변수를 CSS 변수명과 맞춤. Figma MCP를 양방향으로 써서 에이전트로 시안을 캔버스에 만들고 승인된 시안을 코드로 구현함.

비주얼캠프 · 천지문 앱 UI 디자인 · Flutter 적용

외주 · iOS · Android 출시 · 디자인과 디자인 적용 코드

문제 iOS · Android 앱을 동시에 출시. 두 OS는 글자 크기 단위(pt · sp)와 기본 크기(본문 17pt · 16sp), 사용자가 시스템에서 글자를 키우는 방식(Dynamic Type · 글꼴 배율)이 모두 다름. 한 벌의 디자인 값으로 두 앱을 만들면 한쪽은 OS 기준과 어긋나고 큰 글자 설정에서는 본문이 잘릴 위험.

대안 검토 ① 두 앱 공통 값 한 벌: 관리는 쉽지만 한쪽 OS 기준을 어김 ② 글꼴(Pretendard)만 공통으로 두고 글자 크기 · 줄 간격 · 단위는 OS별 두 벌 → ② 채택.

실행 Figma에 글자 스타일 16종을 iOS · Android 두 벌로 만들어 변수로 연결. 개발에서는 토큰처럼 OS 모드만 바꾸면 맞는 값이 나옴. 큰 글자 설정에서도 본문을 고정 높이로 자르지 않고 줄바꿈 · 스크롤하도록 규칙으로 적음. 버튼처럼 누르는 영역은 iOS 44×44pt · Android 48×48dp 이상.

모달 같은 상황(학습 중단 확인 · 지문 필터 · 카메라 권한 등) 24개를 iOS · Android 화면 관례에 맞춰 따로 만들어 48개. 하단 시트 · 확인 창 · 상태 안내 세 종류로 정리.

Flutter 적용 디자인 토큰을 Flutter 테마로 옮기고 홈 · 읽기 · 문항 풀이 · 결과 · 로그인 화면과 학습 시트 · 피드백 모달을 시안대로 적용. iOS에서 Liquid Glass를 지원하면 네이티브 탭 바를 쓰고 아니면 Figma로 만든 대체 하단 바를 그리도록 판정. 스플래시 애니메이션 · 시즌(할로윈) 테마 · 앱 아이콘 작업.

개발 전달 개발자가 헷갈리지 않게 OS 공식 규칙과 이 앱이 정한 값을 나눠 적음. 예: 상단 바 좌우 여백 iOS 24 · Android 16은 OS 규칙이 아니라 이 앱의 결정이라고 표시하고 Apple · Android 공식 문서 링크를 붙임.

GrowingWise · 학원 운영 AI SaaS

외주 · 2026.07 — 2026.10

㈜그로잉와이즈. 2인 팀. 제품 화면 5종의 디자인을 단독으로 맡고 원장 대시보드 · 랜딩 · 회사 사이트는 프론트엔드 구현까지.

문제 AI가 학원 홍보용 블로그 글을 대신 써 주지만 원장님 화면에서는 글마다 지금 어느 단계인지(AI가 쓰는 중 · 검토가 필요함 · 올려도 됨)가 보이지 않아 그대로 올려도 되는지 판단하기 어려움.

대안 검토 ① AI가 쓴 글을 바로 자동 발행: 틀린 글이 그대로 올라감 ② 원장님이 모든 글을 처음부터 다시 검토: AI를 쓰는 의미가 사라짐 ③ 글마다 단계를 화면 맨 위에 보여 주고 마지막 발행만 원장님이 누름 → ③ 채택.

실행 대시보드 맨 위에 글의 단계(AI 작성 중 → 검토 필요 → 승인 → 발행)를 막대로 보여 주고 단계를 누르면 그 단계의 글만 모아 보이게 함. 4일 만에 원장 대시보드를 6개 화면으로 다시 나누고 버튼 · 입력칸 같은 공통 UI를 새로 만듦.

AI 비서 물방울처럼 일렁이는 AI 비서 버튼을 SVG 필터로 만들고 누르면 원장님이 AI에게 묻는 대화 창이 열리게 함.

홈페이지 학원와이즈 홍보 페이지 섹션 19종과 스크롤 연출(GSAP · Lenis)을 디자인하고 구현.

치즈콘 · 이모티콘 작가 피드백 플랫폼

사이드 · 2025.06 — 진행 중

문제 카카오 이모티콘 심사는 승인 · 거절만 알려줘서 작가가 무엇을 고쳐야 할지 알 수 없음. 작가로 활동하며 직접 겪은 문제.

가설 다른 작가에게 이미지 단위로 피드백을 받으면 고칠 곳을 알 수 있다고 보고 작가끼리 피드백을 주고받는 플랫폼을 만듦.

실행 이모티콘은 32장이 한 세트라 세트를 한 번에 올려 피드백을 요청하는 흐름을 성능 중심으로 설계. WebP 변환 · 흐린 미리보기를 CPU 코어 수만큼 띄운 Web Worker(OffscreenCanvas)에서 처리하고(같은 32장을 메인 스레드와 비교한 로컬 측정: 처리 시간 340ms → 95ms) 전처리가 모두 성공해야 업로드해 반쪽짜리 세트가 남지 않게 함. 출시 전 시안이 퍼질까 꺼리는 부담을 줄이려고 원본 대신 흐린 이미지와 만료되는 서명 링크로만 공유.

발견 공유 범위를 좁혀도 작가들은 출시 전 시안을 남에게 보이는 것 자체를 꺼림. 문제는 공개 범위가 아니라 사람에게 보여야 하는 구조였음.

피벗 그래서 사람 대신 AI에게 비공개로 피드백을 받는 방식으로 전환 중.

5 AI 워크플로

역할 분리 에이전트에게 맡기는 일(시안 구현 · PC 시안의 반응형 변환 · 성능 개선 · SEO)과 직접 검증하는 일(토큰 일관성 · 접근성 · 플랫폼 관례 판단)을 나눠 규칙으로 정리.

컨텍스트 설계 반복 작업을 프로젝트별 Skill · 규칙 · 세션 인계 문서로 정리. Figma MCP로 시안을 읽고 캔버스를 만드는 일을 양방향으로 처리.

검증 에이전트 결과물은 그대로 쓰지 않음. 토큰 위반 · 접근성 · 시안 대조로 고친 뒤 반영.

6 교육 · 기타

Phi 인스티튜트 오브 디자인(토스 설립) · Pre-course 1기 Track B(Engineering)

2026.04.21 — 06.17 · 수료

HID 1-Pager로 문제를 정의하고 여러 번 고쳐 쓰며 좁히는 법을 익힘. 처음 주제(AI 코드 구현안 비교)를 중간에 버리고 'AI가 무엇을 기억하는지 채팅 화면이 보여주지 않는다'로 다시 정의해 기억 인스펙터 프로토타입을 만들어 배포.

Engineering 실물 사물을 분해 → 패턴 → 추상화 → 플로우차트(CT 4단계)로 옮긴 뒤 코드 프로토타입으로 만드는 법을 익힘. 모래시계 · 오르골 · 리듬게임(직접 만든 Rive 캐릭터) 제작.

미적 문해력 직접 찍은 사진을 좋은 조형 · 나쁜 조형으로 나누고 '예쁘다' 대신 조형의 말로 이유를 적으며 4주 동안 미감의 기준을 쌓음.

배운 점 핵심 인터랙션은 써보기 전에 확정하지 않음(동기 스크롤을 구현해 써본 뒤 제거). 사용자 테스트 없이 자기 관찰로 문제를 좁힌 것이 한계였음.

우아한테크코스 · 프론트엔드 6기

2024 · 수료

언어 · 협업 JavaScript · TypeScript 심화 학습과 웹 페이지 개발. 페어 프로그래밍으로 소프트 스킬을 키우고 코드 유지보수성을 높이는 실습과 테스트 주도 개발(TDD)을 경험.

React React 기반 프론트엔드 웹 애플리케이션 제작과 컴포넌트 설계. React 모달 라이브러리를 구현해 npm에 배포(npmjs.com/package/river-modal-component).

테코톡 「단일 이미지를 활용한 반응형 이미지 최적화」 10분 테코톡 발표. 이미지 한 장으로 반응형 웹 페이지에 맞는 여러 크기의 이미지를 자동 생성하는 plugin과 웹 페이지를 소개(youtube.com/watch?v=hNov4CJGDBQ).

글쓰기 「리액트에서의 클로저: 개념부터 활용까지」. React의 핵심 개념인 클로저의 동작 원리와 함수형 컴포넌트 · 훅(useState · useEffect · useCallback)에서의 활용 방안을 다루고 오래된 클로저(stale closure) 같은 흔한 실수와 해결 방법까지 정리한 기술 문서(github.com/woowacourse/woowa-writing/blob/0jenn0/technicalWriting.md).