

고경표 Kyoungpyo Koh

Work. Build. Taste.

Product Designer

2026



Greetings!

AI와 함께 논리와 구조를 세우고
감도 높은 디테일로 완성합니다

Work.

맥락을 이해하고 진짜 문제를 정의합니다

Build.

직접 구현하고 아이디어를 검증합니다

Taste.

다양한 경험과 디테일로 완성도를 끌어올립니다

고경표
Kyoungpyo Koh

LG CNS 2024 - Now
Product Designer
Certified AX Expert

Korea University 2018 - 2024
Civil & Architectural Engineering
Software Technology & Entrepreneurship

Skills
Design System with Figma, Miro, Jira & Confluence
React, Flutter, SwiftUI
Codex, Claude Code
GitHub, Vercel, Supabase

pyospect@gmail.com
<https://www.linkedin.com/in/pyospect/>

- 1. 개인의 AI 활용을 팀의 협업 방식으로 연결하기
- 2. 엔터프라이즈 AI 에이전트 플랫폼 설계
- 3. LIG D&A 출장관리시스템 리뉴얼
- 4. Personal Projects

개인의 AI 활용을 팀의 협업 방식으로 연결하기

A Type A 고경표, 10:24

필터를 상단에 고정 : 목록을 탐색하는 중에도 조건을 바로 바꿀 수 있도록

영향 : 목록 화면 1개 · 필터 컴포넌트 2개

B Type B 고경표, 10:31

필터를 사이드로 이동 : 목록 영역을 넓혀 한 화면에 더 많은 결과를 표시

영향 : 목록 화면 1개 · 레이아웃 그리드 변경

A Type A - 필터를 상단에 고정

B Type B - 필터를 사이드로 이동

C 직접 조정

+ 질문 또는 작업 내용을 알려주세요

반가워요, 경표님! Commerce 리뉴얼 프로젝트의 디자이너로 확인했어요. 작업에 필요한 도구 연결 상태부터 점검할게요.

• 사내 도구 검색하는 중

- Jira 연결됨
- Confluence 연결됨
- Git 연결됨 (Bitbucket)

도메인 용어 ▾ 업무 규칙 ▾ 이전 결정 ▾ 사용자 목표 ▾

프로젝트 맥락을 불러왔어요. 작업 중 언제든 다시 물어볼 수 있어요.

업무를 시작하기 전에 확인할 게 있는데, 주문 취소 화면의 권한 분기와 빈 목록 상태가 아직 정의되지 않았어요. 지금 정할까요, 아니면 이슈로 남길까요?

2분 전

지금 정해야겠다. 초안 보여주면 검토해볼게.

90%

TF 주도 · 워크플로우 설계 · 프로토타입 구현

개발팀이 Jira·Confluence·Git으로 일감과 결정, 변경을 추적하듯, 디자이너와 기획자도 같은 구조로 일할 수 있도록 설계부터 구현까지 직접 진행하고 있습니다. 프로젝트 온보딩부터 일감과 맥락 확인, 버전 관리와 충돌 해결까지 하나의 흐름으로 연결했습니다.

AI로 화면을 만드는 속도는 빨라졌지만, 팀의 작업은 여전히 문서와 전달 순서에 묶여 있었습니다

PRD나 화면 정의서 같은 문서는 인수인계의 기준이 되지만, 검토에서 나온 결정 이유와 변경 과정은 회의록이나 메신저, 각자 보관하는 파일에 흩어집니다. 고객이 핵심 화면을 더 빨리 확인하려는 상황에서 디자이너는 맥락과 최신 버전을 다시 찾는 뒤에야 수정을 시작할 수 있습니다.

요구사항

기능 정의

화면 정의

화면 검토

반복 수정

인수인계

회의록

공유 파일

메신저

회의록

메신저

공유 파일

회의록

공유 파일

메신저

메신저

회의록

회의록

메신저

의사결정과 변경 내용을 다음 사람도 이어받을 수 있어야 했습니다

이미 이런 구조로 일하는 곳, 바로 개발팀입니다.

할 일은 Jira에, 배경과 결정은 Confluence에, 모든 변경은 Git에 남깁니다.

누가, 언제, 무엇을, 왜 바꿨는지가 자연스럽게 쌓이는 구조이지만,
이 도구들은 디자이너와 기획자에게는 진입 장벽이 높습니다.

그래서 구조는 유지하고 어려운 부분은 에이전트에게 맡기면, 디자이너와 기획자도
이 업무 방식을 함께 쓸 수 있다고 생각했습니다.



이 파트는 지금 누가 얼마나 진행했어요?

지난번에 왜 그렇게 결정했었나요?

이 화면은 누가 어떻게 바꾼 걸까요?

Agent

* Thinking...

Connected with  Figma

Connected with  Jira

Connected with  Confluence

Connected with  git

업무 시작부터 마무리까지 결정과 기록이 흩어지지 않도록 연결했습니다

일감을 선택하면 업무 관련 맥락을 함께 확인하고, 에이전트가 검토에 필요한 핵심 화면을 먼저 만들어 보여줍니다.
사람은 결과 검토에만 집중하고, 수정 내용이 겹치면 어느 쪽을 반영할지만 결정합니다.

○ 사람이 결정하는 지점 ● 에이전트가 처리하는 지점

1

온보딩

에이전트가 역할, 참여 프로젝트,
도구 연결 상태를 확인하고
빠진 조건을 먼저 질문

LLM

2

일감 & 맥락

일감을 선택하면 목적, 업무 규칙,
이전 결정을 일감 내용과 함께
바로 확인

Jira

Confluence

3

생성 & 검토

결정에 필요한 핵심 화면부터
만들어 검토하고, 확정과 수정을
사람이 결정

LLM

4

버전 & 충돌

변경은 버전으로 남기고,
겹치면 두 변경의 의도를 비교해
함께 선택

Git

5

팀 자산

확정된 결정과 변경 이유를
프로젝트 메모리로 축적해
다음 작업의 기준으로

Project Memory

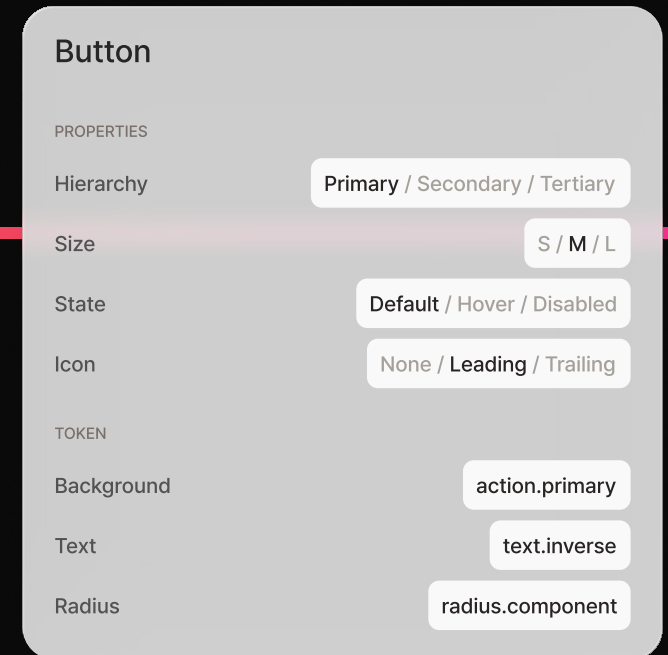
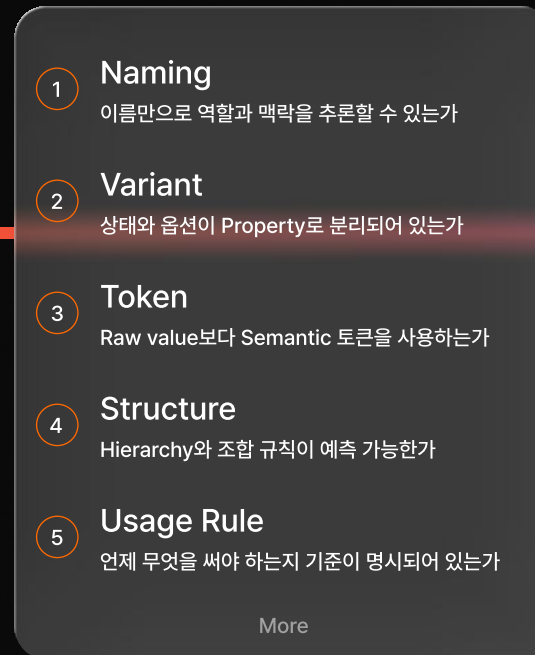
AI가 화면을 만들기 전에는 디자인 시스템부터 읽을 수 있어야 합니다

여러 고객사의 디자인 시스템을 점검하며 사람이 이해하는 규칙과 AI가 일관되게 해석할 수 있는 구조 사이의 간극을 정리했습니다.
Naming, Variant, Token, Component 구조와 사용 규칙을 기준으로 시스템을 보완하고 있습니다.

Figma Design System

→ AI-Ready Criteria

→ Structured for AI



기준은 Skill로 만들고 생성 결과까지 반복해서 검증했습니다

디자인 시스템의 규칙을 하네스로 만들어 화면 생성 과정에 적용하고 있습니다.
같은 조건에서 생성된 화면이 디자인 시스템, 맥락, 접근성을 지키고 있는지 LLM과 스크립트를 활용하며 반복 검증합니다.



Component Rules

- Use existing components before creating new ones.

- Use semantic tokens only.

- Do not introduce arbitrary spacing values.

- Follow defined component hierarchy and interaction patterns.

Component Usage

기존 패턴 대신 신규 CTA 스타일 생성

NEED REVIEW

Semantic Token

Raw color / spacing value 사용

FAIL

Layout Pattern

목록 필터 규칙과 다른 위치 사용

NEED REVIEW

Component Usage

기존 Button / Filter component 재사용

PASS

Semantic Token

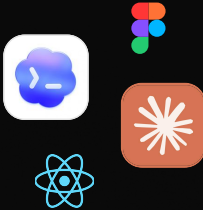
정의된 색상 · 간격 Token 준수

PASS

Layout Pattern

기존 목록형 화면 패턴 유지

PASS



온보딩 에이전트

에이전트가 도구 연결을 점검하고, 프로젝트의 용어와 규칙, 사용자의 목표와 제약을 미리 불러옵니다. 로딩이나 권한, 빈 화면처럼 설계에서 놓치기 쉬운 조건은 먼저 질문합니다.

반가워요, 경표님! Commerce 리뉴얼 프로젝트의 디자이너로 확인했어요.
작업에 필요한 도구 연결 상태부터 점검할게요.

● 사내 도구 검색하는 중

-  Figma 연결됨
-  Jira 연결됨
-  Confluence 연결됨
-  Git 연결됨 (Bitbucket)

도메인 용어 ▾

업무 규칙 ▾

이전 결정 ▾

사용자 목표 ▾

프로젝트 맥락을 불러왔어요. 작업 중 언제든 다시 물어볼 수 있어요.

업무를 시작하기 전에 확인할 게 있는데, 주문 취소 화면의 권한 분기와 빈 목록 상태가 아직 정의되지 않았어요.
지금 정할까요, 아니면 이슈로 남길까요?

2분 전

지금 정해야겠다. 초안 보여주면 검토해볼게.

Jira & Confluence

Jira 일감에는 이미 프로젝트 맥락이 담겨 있습니다.
어떤 규칙을 지켜야 하는지, 지난번 검토에서 무엇이 결정되었는지는 Confluence 문서에 함께 기록하도록 했습니다.
검토에서 확정된 화면과 변경 사유는 메모리로 남아 다음 작업의 출발점이 됩니다.

MyWork

주문 취소 플로우 개선
PROJ-214 · In Progress

부분 취소 사유 입력 화면
PROJ-218 · To Do

환불 상태 알림 배너
PROJ-221 · To Do

주문 취소 플로우 개선

PROJ-214 · In Progress

Summarized by AI Agents

목적 ▾

취소·환불 관련 CS 문의를 줄이는 것이 이번 분기의 핵심 목표

✕ ABC 2026 Q1 OKR

업무 규칙 ▾

- 부분 취소는 결제수단별로 환불 규칙이 다름 → 카드는 승인 취소, 계좌는 익일 입금
- 취소 완료 전에는 환불 예정 금액을 노출하지 않음
- 프로모션 적용 주문은 취소 시 할인 반환 조건 안내 필수

✕ PROJ Ground Rule v.1

이전 결정 ▾

취소 사유는 자유 입력 대신 선택식으로 확정 (데이터 축적을 위해)

✕ [PROJ-214] 의사결정 로그

연결된 핵심 화면 ▾

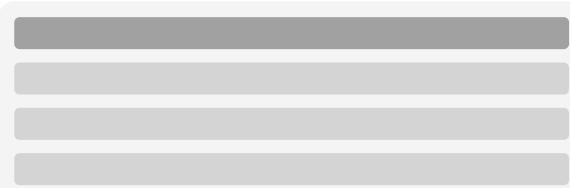
Git 기반 협업 & 충돌 해결

수정이 겹치면 에이전트가 두 변경을 화면 기준으로 나란히 놓고 의도와 영향을 설명하고, 팀원들이 직접 선택합니다. 복잡한 명령어 없이 판단할 수 있고, 결정된 내용은 바로 코드 레벨로 반영됩니다.

같은 화면을 수정한 두 변경이 충돌했어요. 어느 쪽을 기준으로 할까요?

A Type A

홍길동, 10:24

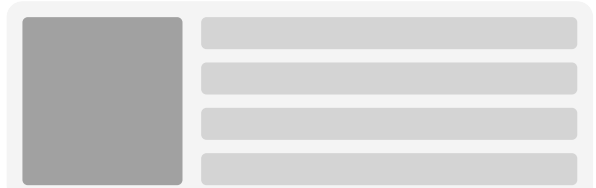


필터를 상단에 고정 : 목록을 탐색하는 중에도 조건을 바로 바꿀 수 있도록

영향 : 목록 화면 1개 · 필터 컴포넌트 2개

B Type B

고경표, 10:31



필터를 사이드로 이동 : 목록 영역을 넓혀 한 화면에 더 많은 결과를 표시

영향 : 목록 화면 1개 · 레이아웃 그리드 변경

A Type A - 필터를 상단에 고정

Recommended

B Type B - 필터를 사이드로 이동

C 직접 조정

+ 질문 또는 작업 내용을 알려주세요



팀이 함께 사용할 수 있는 협업 구조를 만들고 실제 적용 범위를 확인하고 있습니다

온보딩부터 충돌 해결까지 모든 과정을 사내 솔루션으로 구현하여 파트 구성원들과 함께 다듬고 있습니다.
혼자 일할 때는 사라지기 쉬웠던 판단과 변경 사유가, 팀 전체가 공유할 수 있는 형태로 기록되고 있습니다.



구현 완료
온보딩부터 충돌 해결까지 활용 가능한 단계



팀과 검증 중
파트 구성원들과 함께 사용하며 다듬는 단계

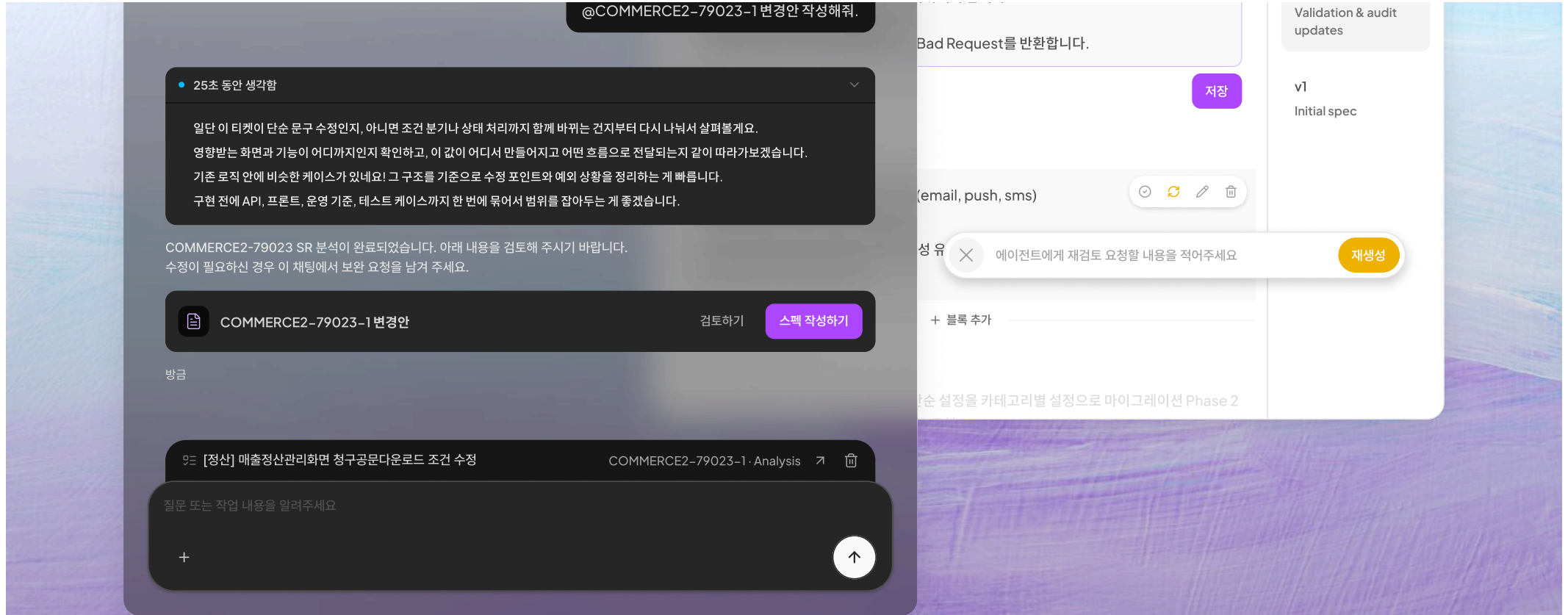


파일럿 적용 및 성과 측정
프로젝트 선정 후 실제 업무에서 확인

협업 도구의 구조만큼 사용자가 마주하는 언어도 중요했습니다

모든 결정과 변경을 기록하는 개발팀의 방식을 디자인과 기획 업무에서도 유효하도록 용어를 정리했습니다.
어려운 조작을 에이전트에게 맡기면, 사람은 문제 정의와 검토, 결정처럼 판단이 필요한 일에 집중할 수 있습니다.

엔터프라이즈 AI 에이전트 플랫폼 설계



50%

리서치 · UX설계 · 화면 및 프로토타입 구현

요구사항 분석부터 개발까지 지원하는 솔루션 “Cline Spec Driven”을 미국 스타트업 Cline과 함께 설계했습니다. 기업이 AI에게 일을 맡기면서도 과정과 결과를 통제할 수 있도록 협업 구조를 정의했고, 디자인 프로토타입이 실제 제품 개발 기반으로 이어졌습니다.

사람 사이의 소통도 쉽지 않은데, 사람과 AI 사이는 어떨까요?

처음 마주한 과제

사람이 읽는 문서와 AI가 처리하기 좋은 데이터는 형태가 다릅니다.

이렇게 형식이 다르면 변환하는 과정에서 정보가 누락되거나 맥락이 어긋나는 문제가 발생합니다.

이 간극을 줄이기 위해 둘 다 읽을 수 있는 Markdown 기반 Spec 문서를 공통 기준으로 삼는 솔루션에서 출발했습니다.

진짜 병목을 확인하기로 했습니다

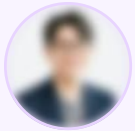
“문서 형식의 간극”이 정말 사용자가 겪는 가장 큰 병목인지 직접 확인하고 싶었고,

이 솔루션으로 프로젝트를 수행하게 될 동료 엔지니어 3명을 인터뷰했습니다.

그 결과 표면 아래 숨어 있던 두 가지 문제를 발견했습니다.



요즘 AI와 협업하면서 느낀 가장 큰 병목은 무엇이었나요?

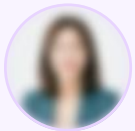


결과만 나오니까 어디서부터 잘못됐는지 모르겠어요...

AI가 아무리 빠르게 병렬로 작업해도 결과물의 최종 검수는 결국 사람의 몫이잖아요?
그런데 작업 과정이 보이지 않으면 중간에 개입할 수 없고, 결과가 나온 뒤에야 오류를 발견하게 됩니다.



사람이 과정을 지켜보다가 언제든 중단할 수 있어야겠네요



저 혼자 AI로 코딩을 잘한다고 해서 조직 모두가 잘 쓰는 건 아니더라고요.

Codex, Claude Code 같은 범용 코딩 에이전트는 이미 충분히 강력합니다.
하지만 개인이 능숙해지는 것과, 대규모 개발 조직이 같은 표준으로 일하는 것은 별개의 문제인 것 같아요.



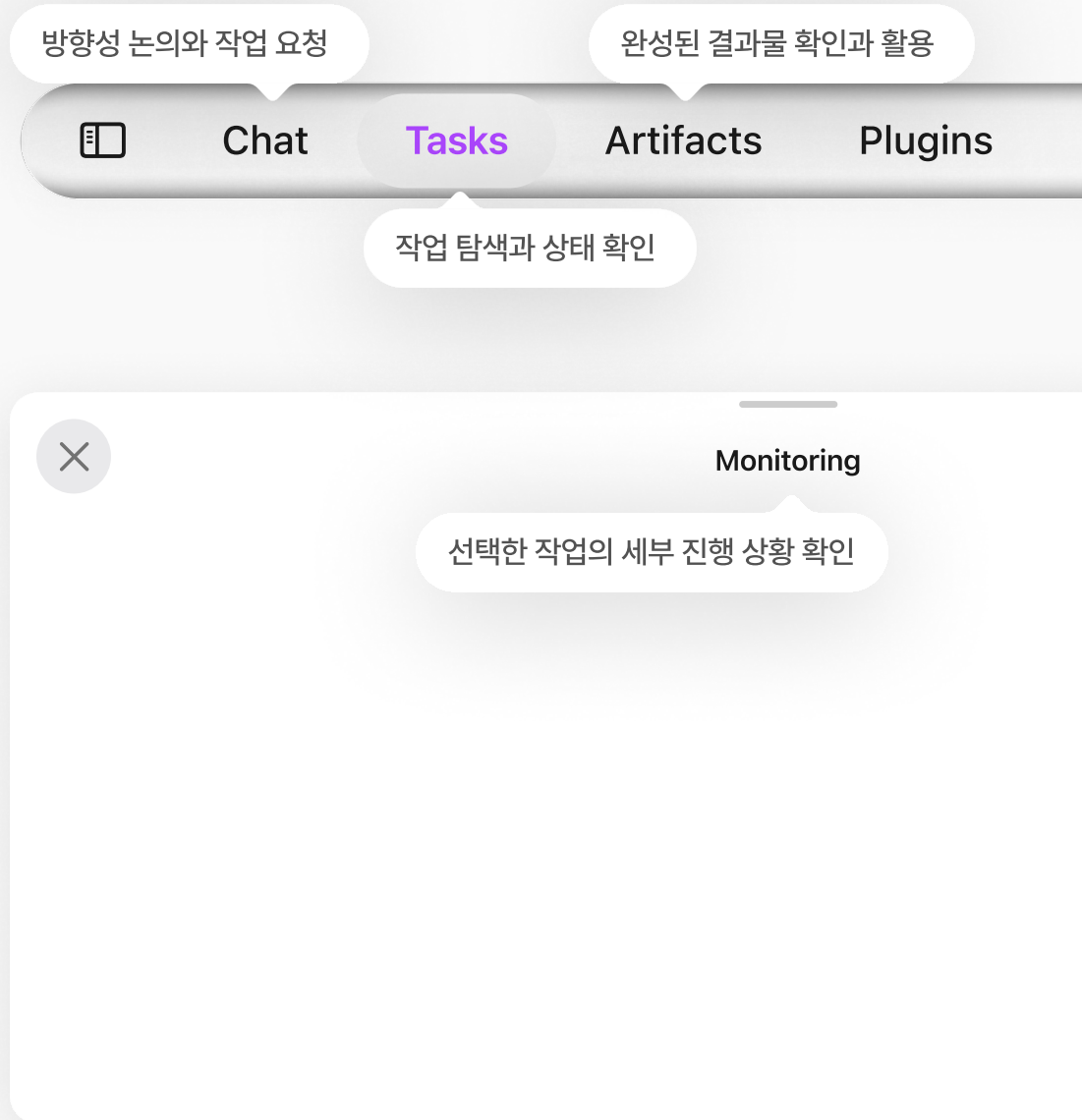
개인용 툴은 있어도 팀 전체가 같은 방식으로 AI와 협업할 수 있는 체계가 없군요

채팅이 모든 걸 책임지지 않도록 협업의 흐름을 다시 설계했습니다

AI와 협업하는 과정을 대화, 진행, 결과 세 단계로 나누었습니다.
진행 단계는 Tasks 목록과 Monitoring 패널로 분리하여
전체 작업과 개별 작업을 각각 확인할 수 있습니다.



항상 같은 형식으로 확인하는 정보까지 매번 채팅으로 물어볼
필요는 없습니다. 정보의 성격에 맞는 UI에서 더 빠르고
명확하게 확인할 수 있으니까요.



Ask in Chat

AI가 다음 단계로 넘어가기 전에 사용자의 확인을 받습니다.
분석 시작, 변경안 작성, 구현 여부처럼 방향이 갈리는 순간은 사용자가 직접 결정합니다.

@COMMERCE2-79023-1 변경안 작성해줘.

25초 동안 생각함

- 일단 이 티켓이 단순 문구 수정인지, 아니면 조건 분기나 상태 처리까지 함께 바뀌는 건지부터 다시 나눠서 살펴볼게요.
- 영향받는 화면과 기능이 어디까지인지 확인하고, 이 값이 어디서 만들어지고 어떤 흐름으로 전달되는지 같이 따라가보겠습니다.
- 기존 로직 안에 비슷한 케이스가 있네요! 그 구조를 기준으로 수정 포인트와 예외 상황을 정리하는 게 빠릅니다.
- 구현 전에 API, 프론트, 운영 기준, 테스트 케이스까지 한 번에 묶어서 범위를 잡아두는 게 좋겠습니다.

COMMERCE2-79023 SR 분석이 완료되었습니다. 아래 내용을 검토해 주시기 바랍니다.
수정이 필요하신 경우 이 채팅에서 보완 요청을 남겨 주세요.

COMMERCE2-79023-1 변경안

검토하기

스펙 작성하기

방금

☰ [정산] 매출정산관리화면 청구공문다운로드 조건 수정

COMMERCE2-79023-1 · Analysis

↗

🗑

질문 또는 작업 내용을 알려주세요

+

↑

Monitoring in Tasks

에이전트가 어디까지 작업했는지 일감 단위로 한눈에 볼 수 있습니다.
Jira에서 가져온 일감별 패널에서 진행 단계, 체크리스트, diff, 로그를 순서대로 확인합니다.
잘못된 방향은 결과가 나오기 전에 발견하고, 언제든지 멈출 수 있습니다.

● PROJ-1770-1 · 결제 수단 변경 시 쿠폰 적용 초기화 문의58%

3단계 진행 중

Explored 12s

PaymentMethodService에서 changePaymentMethod()가 호출될 때 CartStore의 coupon 상태가 리셋되는 흐름을 먼저 확인하겠습니다.

Read src/cart/CartStore.tsx for 25s

reset되는 부분을 Payment만 있을 때와 쿠폰을 적용했을 때를 나누어 적용하도록 수정하겠습니다.

src/payment/PaymentMethodService.tsx+4 -1

- CartStore.reset();

+ CartStore.resetPaymentOnly();

+ CartStore.recalculateWithCoupon();

Thought 5s

npm test

PASS src/payment/__tests__/PaymentMethodService.test.ts

✓ should preserve coupons when payment method changes

✓ should recalculate total discount correctly

Task Checklist

✓ PaymentMethodService 분석 완료

✓ CartStore 상태 분석 완료

☐ changePaymentMethod() 수정 예정

Artifact Editor

팀 전체가 같은 Spec 레이아웃을 기준으로 검토하고 수정할 수 있어야 합니다.
결과물을 블록 단위의 초안으로 다루고, 누구든 수정하고 확정할 수 있습니다.

확정 재생성 직접 수정 삭제 + 추가

PROJ-1773 Spec

3/32 blocks

1. Validation Rules

- category는 email, push, sms 중 하나여야 합니다.

- 값은 boolean 타입만 허용합니다.

- 유효하지 않은 카테고리 요청은 400 Bad Request를 반환합니다.

취소

저장

2. Scope & Objectives

알림 카테고리별 ON/OFF 설정 지원 (email, push, sms)

사용자별 설정 수정 및 조회 API 확장

기존 단순 ON/OFF 설정과의 하위 호환성 유

설정 변경 이력 추적 (감사 로그)

×

에이전트에게 재검토 요청할 내용을 적어주세요

재생성

+ 블록 추가

3. Migration Plan

Phase 1 (Week 1): 스키마 확장 및 기존 단순 설정을 카테고리별 설정으로 마이그레이션

Phase 2

2 Versions

Diff

v2

latest

Validation & audit updates

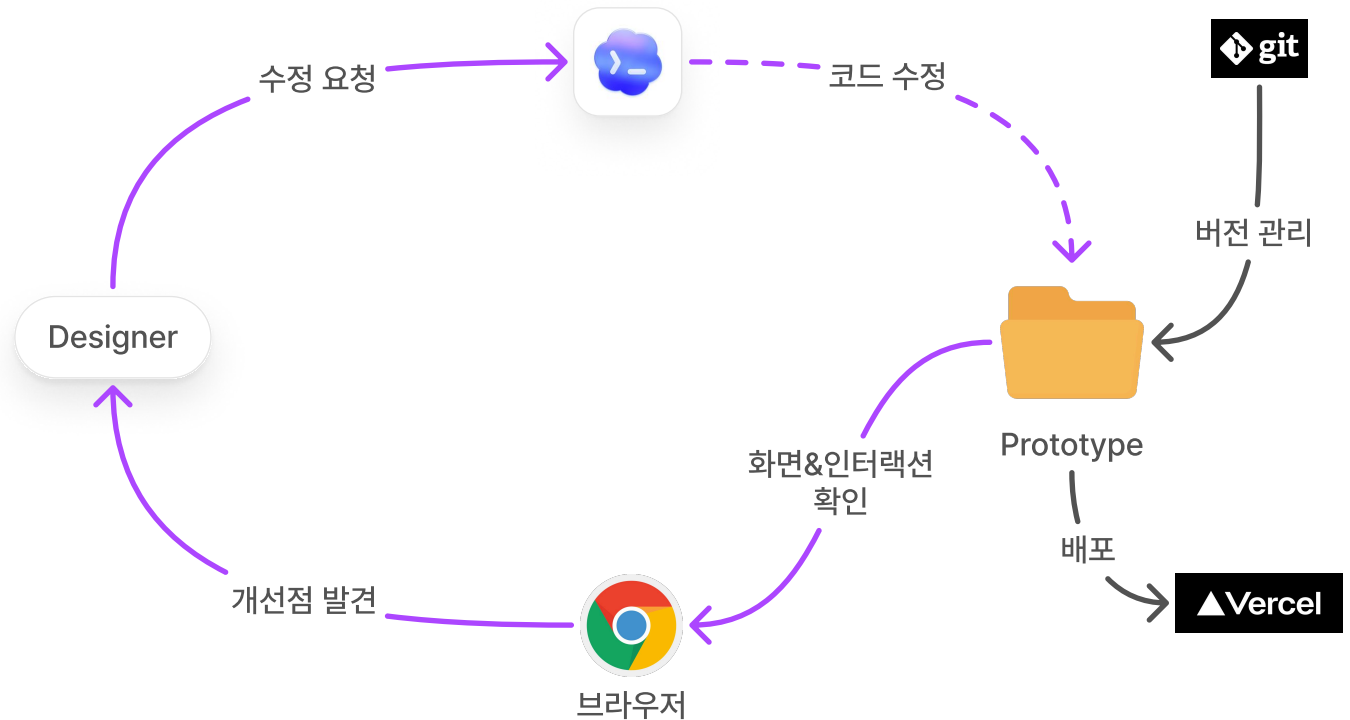
v1

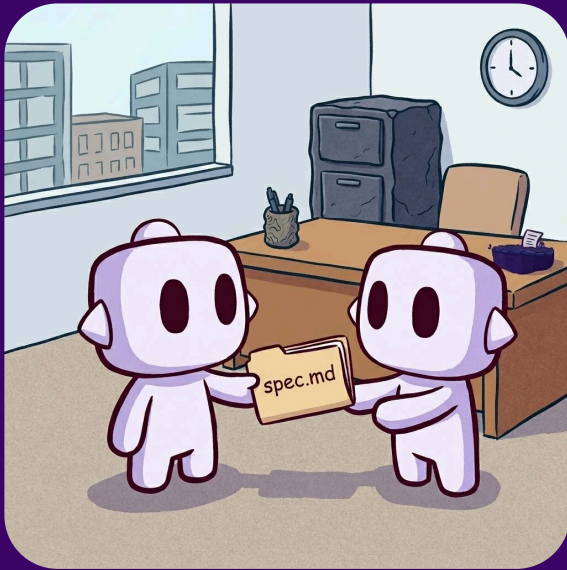
Initial spec

디자이너의 프로토타입이 실제 제품의 기반이 되기까지

AI Literacy를 바탕으로 프로토타입 코드를 실제 제품의 핵심 구조로 발전시켰습니다.

같은 화면과 코드를 엔지니어와 함께 보며 아키텍처를 구체화했고, 이 방식은 지금 사내 프로젝트의 논의와 검증에 활용되고 있습니다.





AI 시대가 찾아와도 UX는 결국 사람의 여정을 설계하는 일입니다

사용자가 지금 무슨 일이 일어나는지 이해하고 통제할 수 있어야, AI는 협업 도구로 받아들여집니다. 엔터프라이즈에서는 이 통제감이 곧 신뢰와 사업 기회로 이어졌고, 개인의 능숙함을 조직의 표준으로 만드는 일이 AI 시대 UX의 새로운 역할이라고 느꼈습니다.

이 협업 방식을 팀 전체로 넓히려는 고민이, 첫 번째 케이스의 출발점이 되었습니다.

LIG D&A 출장관리시스템 리뉴얼

60%

데이터 분석 · 아키텍처 및 화면 설계

20년 넘은 레거시 시스템의 데이터 구조와 유저 플로우를 바로잡고, 점진적 공개(Progressive Disclosure) 기반 워크플로우를 처음 도입하여 화면을 개선했습니다. 그 결과 필수 입력 항목을 47개에서 16개로 줄이고 시스템 학습 시간을 4시간에서 20분으로 단축했습니다.

TRIP MANAGEMENT SYSTEM

검색어를 입력하세요

10분 전 저장됨

임시저장

결제 요청하기

국내출장 정산하기

출장 기본정보

출장목적
고경표 (12345)

출장기간
2025-06-10 ~ 2025-06-13 (4일)

출장목적
업무협의

정산구분
실비형

정산정보

법인카드 사용금액
423,400원

개인카드 사용금액
14,800원

현금 사용금액
40,900원

기타 지급액 ①
52,200원

총 비용 ①
479,100원

법인카드 사용금액을 제외한 지급 예정액 ①
55,700원

남은 예산
16,891,634원 / 20,000,000원

결제정보

결제자 미정

2025-06-27 지급 예정

출장정보 입력 수정

2 정산정보 입력

교통비

등록이트비

활동비

숙박비

차량유지비

3 결제정보 입력 수정

들어오는 경로 추가하기

미조회 지역거리기준 추가 요청하기

법인카드로 일괄 추가하기

교통비 전체

109,100원

법인카드

53,400원

개인카드

14,800원

현금

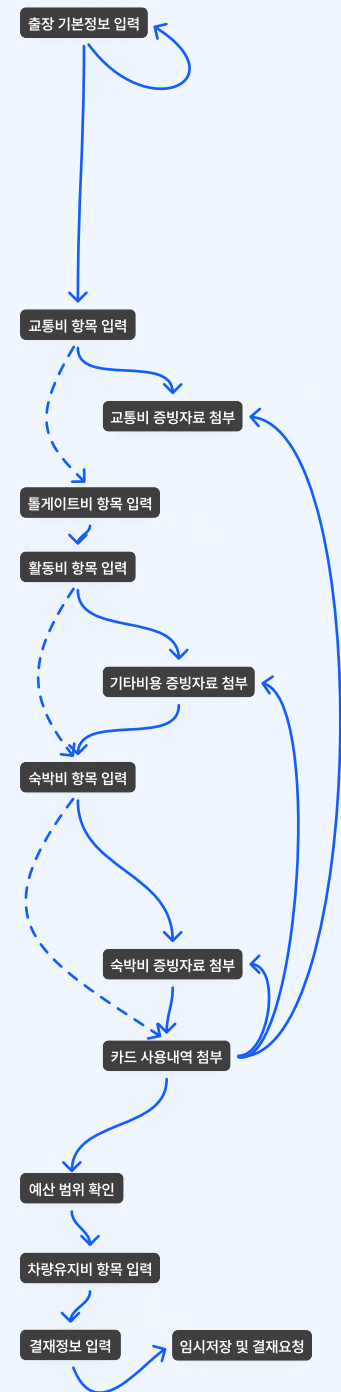
40,900원

날짜	출발지 → 도착지	교통편	거리(km) ①	금액(원)	결제수단	증빙내역 ①
☒ 2025-06-10	판교HQ → 센트럴시티(서울)	자차	19	26,100	① 정산 후 해당 금액만큼 급여계좌로 입금될 예정입니다	
☐ 2025-06-10	센트럴시티(서울) → 전주고속터미널	고속버스	--	22000	법인카드 개인카드 현금	증빙 추가하기
☐ 2025-06-10	전주고속터미널 → 제1연구소	택시	--	14800	법인카드 개인카드 현금	증빙 추가하기
☐ 2025-06-12	제1연구소 → 전주역	업무용차	24	0	① 소속부서 비용으로 청구되므로 정산 대상이 아닙니다	
☐ 2025-06-12	전주역 → 오송역	고속철도(일반실)	--	14800	법인카드 개인카드 현금	IMG_20250612_105228.jpg ×
☐ 2025-06-12	오송역 → 제3공장	동승	32	0	① 동승하는 경우에는 운전자 계정으로만 정산 진행 바랍니다	
☐ 2025-06-13	제3공장 → 대전복합	기타 교통편	12	40900	법인카드 개인카드 현금	증빙 추가하기

복잡한 화면보다
오래된 업무 구조 자체가 문제였습니다

오랫동안 기능이 덧붙여지면서 실제 출장 절차와 데이터 구조가 어긋나 있었습니다. 사용자는 긴 화면을 오르락내리락하면서 똑같은 항목을 반복해서 입력하거나, 화면 곳곳에 숨겨진 입력 조건을 비교해 가면서 하나씩 직접 판단해야 했습니다.

시스템이 처리해야 할 복잡한 계산 방식이 사용자에게 떠넘겨진 셈입니다.



출장구분

국내일반출장

출장자정보

사번

12345

성명

고영표

부서

Design팀

직책

사원

출장내용

출장기간

년-월-일
--선택--
년-월-일

Product

--선택-- 공통

출장목적

--선택-- 출장지 도착지

출장지역

출발지

--선택--

도착지

--선택--

* 출발지는 소속사업장명, 도착지는 출장목적지를 입력하시기 바랍니다.

출장내용

첨부파일

파일 선택

 신박진 파일 업로드

구름비 (20대 15인)

* 출장비용 검토도 모두 입력후시작이 가능합니다.(출장, 업무휴가비용 포함)

구름비 출장파워블로그(법안카드 증명 첨부X)

증빙내역

첨삭보기

출장기록부 (20대 15인)

출발

년-월-일

입

출격여부(입/출)

년-월-일

출

* 차량통 출격기록의 경우, 입·출구통을 출납하게 입력하여야합니다.

함두버

* 출석을 미검토한 경우, 반드시 미검토 처리박스에 체크 후 일수를 입력하십시오.
* 미검토 일수만큼 직접작성한 경우, 통틀에 기간 시 이상 처리됩니다.

기타 증명파워블로그(법안카드 증명 첨부X)

증빙내역

첨삭보기

속박비

속박비 증명파워블로그(법안카드 증명 첨부X)

증빙내역

첨삭보기

카드 사용내역 (20대 15인)

구분

사용일자

카드회사

카드번호

승인번호

--선택-- 년-월-일 --선택--

* 사업장으로 등록된 카드가 없는 경우 사립카드번호로 조회해 입력합니다.

출장차계산내역

기간

활동비(원)

숙박비(원)

년-월-일 - 년-월-일 0 0

비용 & 예산

비용부서

기용예산

계정구분

기용예산

0 기용예산조회

차량유지비

비용부서

금액(원)

0

결제

* 결제 ○ 할지

결제자

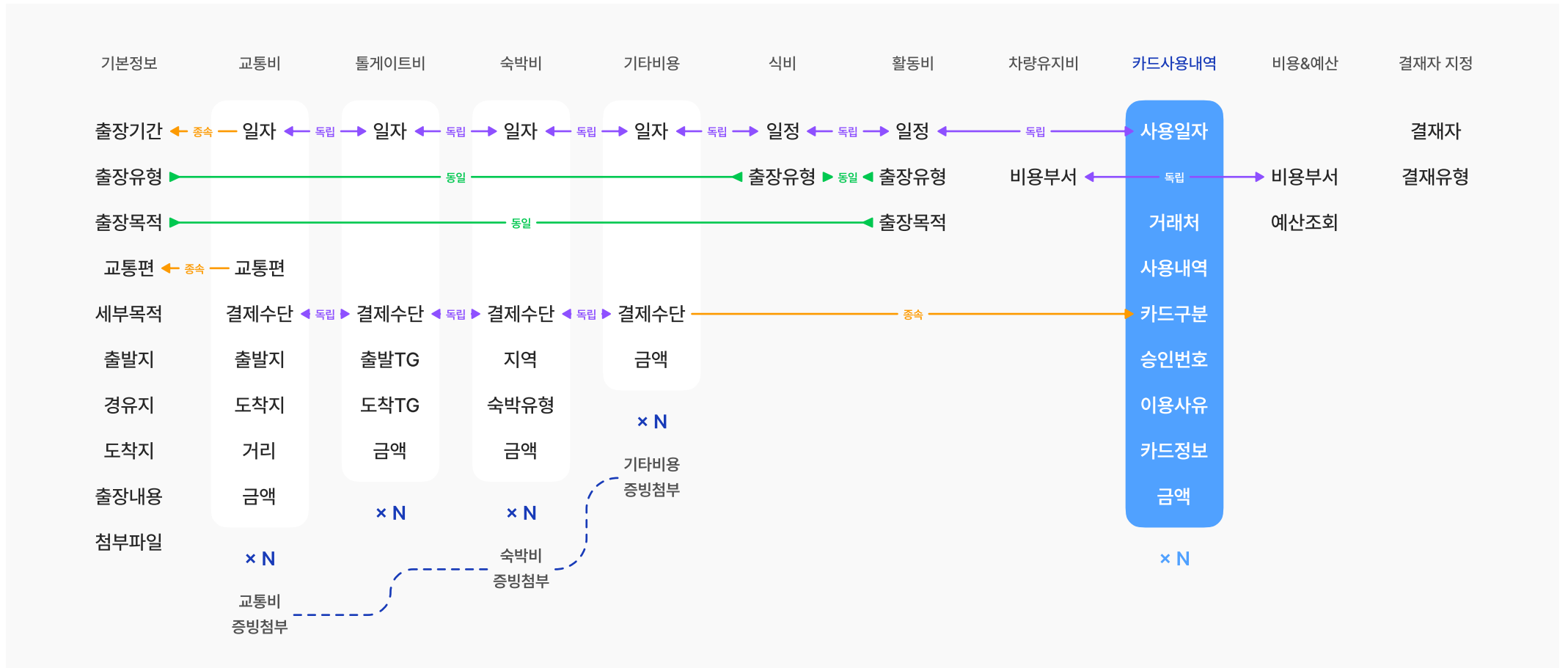
부서

임시차량

결제요청

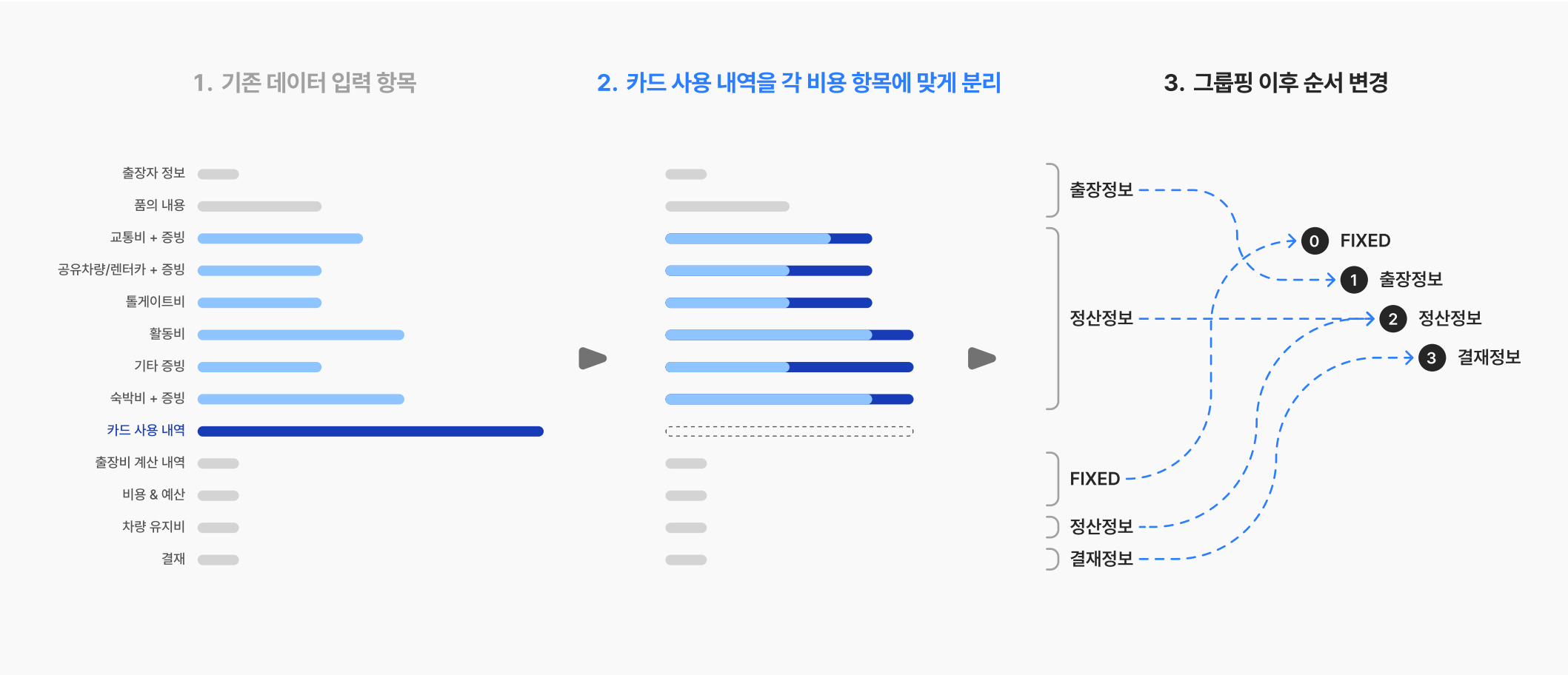
우선 숨겨진 데이터까지 모두 펼쳐 보았습니다

단순히 입력 항목 개수보다 서로 다른 관계의 데이터가 한 화면에 무작위로 섞여 있어서 사용하기 어렵다는 점을 발견했습니다. 동일한 항목명이어도 다르게 입력해야 하고, 서로 다른 입력칸이어도 동일하게 입력해야 하는 항목이 섞여 있었습니다.



업무 절차와 데이터 조건에 맞게 구조를 바꾸었습니다

비용 항목 중심으로 나열된 데이터를 해체하고, 업무 흐름과 데이터 종속 관계를 기준으로 입력 항목을 재구성했습니다.
조건에 따라 달라지는 값과 고정된 값을 분리하여 사용자가 판단해야 했던 구조적 부담을 시스템이 대신 처리합니다.



그룹핑한 항목을 Stepper 형태로 나누었습니다

전체 업무를 출장정보 → 정산정보 → 결재정보로 재구성하여 각 단계에는 필요한 정보만 남기고, 완료된 내용은 요약해서 보여주었습니다.

- 0 FIXED
- 1 출장정보
- 2 정산정보
- 3 결재정보

TRIP MANAGEMENT SYSTEM

검색어를 입력하세요

10분 전 저장됨

임시저장

결제 요청하기

국내출장 정산하기

출장 기본정보

출장자
고경표 (12345)

출장기간
2025-06-10 ~ 2025-06-13 (4일)

출장목적
업무협의

정산구분
실비형

정산정보

법인카드 사용금액
423,400원

개인카드 사용금액
14,800원

현금 사용금액
40,900원

기타 지급액 ①
52,200원

총 비용 ①
479,100원

남은 예산
16,891,634원 / 20,000,000원

법인카드 사용금액을 제외한 지급 예정액 ①
55,700원

2025-06-27 지급 예정

결재정보

결재자 미정

출장정보 입력 수정

정산정보 입력

결재정보 입력 수정

돌아오는 경로 추가하기

미조회 지역거리기준 추가 요청하기

법인카드로 알뜰 추가하기

교통비 전체

109,100원

법인카드

53,400원

개인카드

14,800원

현금

40,900원

날짜	출발지 → 도착지	교통편	거리(km) ①	금액(원)	결제수단	증빙내역 ①
2025-06-10	판교HQ → 센트럴시티(서울)	자차	19	26,100	① 정산 후 해당 금액만큼 급여차로 입금할 예정입니다	
2025-06-10	센트럴시티(서울) → 전주고속터미널	고속버스	--	22000	법인카드 개인카드 현금	증빙 추가하기
2025-06-10	전주고속터미널 → 제1연구소	택시	--	14800	법인카드 개인카드 현금	증빙 추가하기
2025-06-12	제1연구소 → 전주역	업무용차	24	0	① 소속부서 비용으로 청구되므로 정산 대상이 아닙니다	
2025-06-12	전주역 → 오송역	고속철도(일반실)	--	14800	법인카드 개인카드 현금	IMG_20250612_105228.jpg ×
2025-06-12	오송역 → 제3공장	통승	32	0	① 통승하는 경우에는 운전자 계정으로부터 정산 진행 바랍니다	
2025-06-13	제3공장 → 대전복합	기타 교통편	12	40900	법인카드 개인카드 현금	증빙 추가하기
2025-06-13	대전복합 → 서울경부	고속버스	--	16600	법인카드 개인카드 현금	IMG_20250613_192428.jpg ×
2025-06-13	서울경부 → 판교HQ	자차	19	26,100	① 정산 후 해당 금액만큼 급여차로 입금될 예정입니다	

+

 비용 추가하기

10분 전 저장됨

임시저장

장기출장자 대리 신청하는 경우를 위해
출장 기본정보에 출장자 정보 노출

정산을 진행하며 예산 범위 포함 여부를
스크롤 없이도 실시간으로 확인 가능

FIXED 영역과 액션 버튼이
하단 고정인 아닌 상단인 이유

정산항목을 많이 추가하지 않을 경우
시선이 밑으로 가지 않아도 되기 때문

프로세스 탭 UI 활용하여
다음으로 넘어가기 전에 가드레일 역할 수행

출장자가 가장 궁금해하는 정보는
따로 계산하여 별도 표기

정산금액 지급 예정일 표시

출장정보 화면에서 즐겨찾기/최근 출장의 품의복사 기능 제공

돌아오는 경로 추가하기 기능

법인카드 승인내역 팝업에서
복수선택하여 정산항목 여러 개 추가 가능

로직상 또는 일정상 정산이 불필요한 항목은
비활성화 또는 탭을 열지 않아도 이용 가능

빈 화면을 스크롤하며 정산항목 위치를 찾지 않아도 됨

증빙 추가하기 버튼 선택 시
결제수단에 따라 증빙 팝업이 다르게 노출

비용이 발생하지 않은 정산항목일 경우에는
결제수단 옆에 가이드 문구 표시

 비용 추가하기

- 재무팀 정산 담당자의 오픈 첫 주 피드백 -

"많이 고민한 흔적이 보여요. 이제 이전으로는 못 돌아가겠어요."

별도 교육이나 가이드 없이, 오픈 3일 만에 약 3,000건의 정산 데이터가 정상적으로 쌓였습니다.
입력한 데이터가 다른 영역의 액션 때문에 삭제되던 문제도 사라졌습니다.

복잡함은 화면 뒤에 숨겨진 업무 구조에서 온다는 것을 배웠습니다

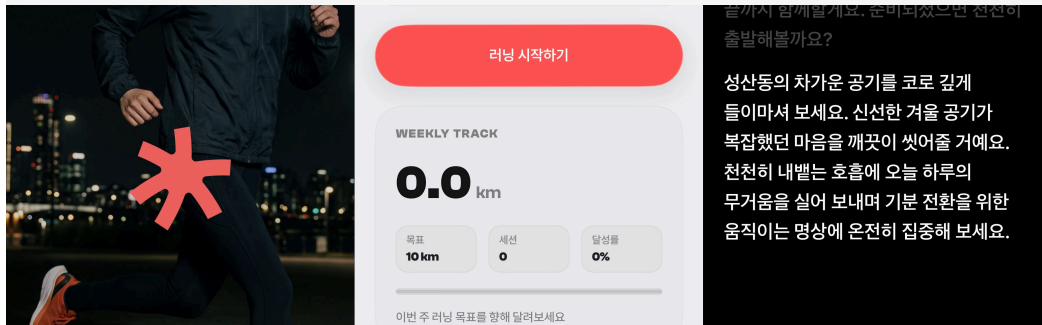
데이터의 관계를 먼저 정리하자 화면은 자연스럽게 단순해졌고, 사용자가 하던 판단을 시스템이 대신할 수 있게 되었습니다.
이렇게 데이터와 업무 흐름을 구조화한 경험은 이후 AI와의 협업 구조를 설계하는 기반이 되었습니다.

Personal Projects

뉴런 Newrun

[App Store →](#)

러닝 목적, 날씨, 위치, 심박수에 맞춰
음성으로 응원하는 AI 러닝 코치 앱



포토리지 Photorage

[App Store →](#)

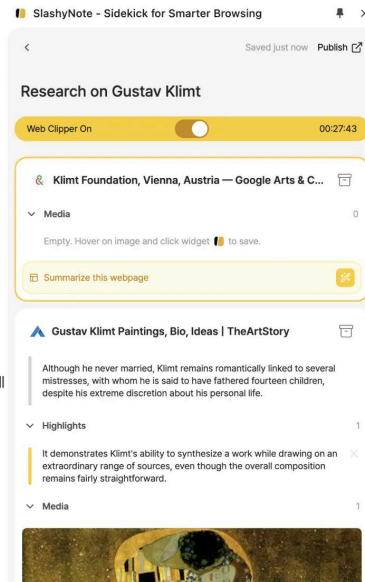
흩어진 네컷사진을 한곳에 모으고
주변 사진관까지 찾아주는 아카이빙 앱



SlashyNote

웹에서 찾은 정보를 표시하고
AI로 요약하는 크롬 익스텐션

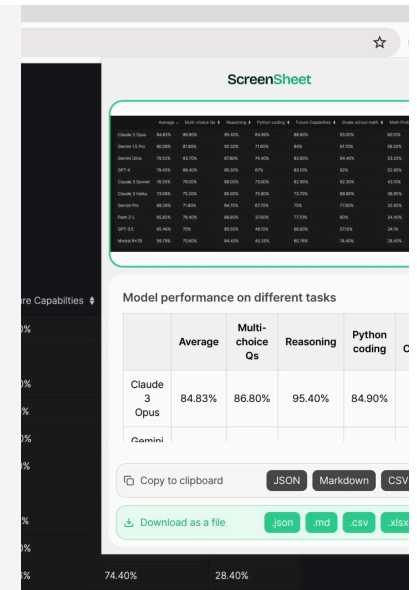
[Chrome Web Store →](#)



ScreenSheet

화면을 캡처하면 이미지 속 표와
차트를 엑셀로 변환하는 앱

[Chrome Web Store →](#)



Work. Build. Taste.

고경표

Kyoungpyo Koh

pyospect@gmail.com

<https://www.linkedin.com/in/pyospect/>