사람과 현장에 묻고 아이디어를 연결해 기획으로 실현합니다.

질문을 통해 문제의 본질을 파악하고,
핵심을 통해 확장성까지 생각합니다.

포트폴리오를 웹사이트로 경험하고 싶다면? 웹 포트폴리오 바로가기 ↗

01

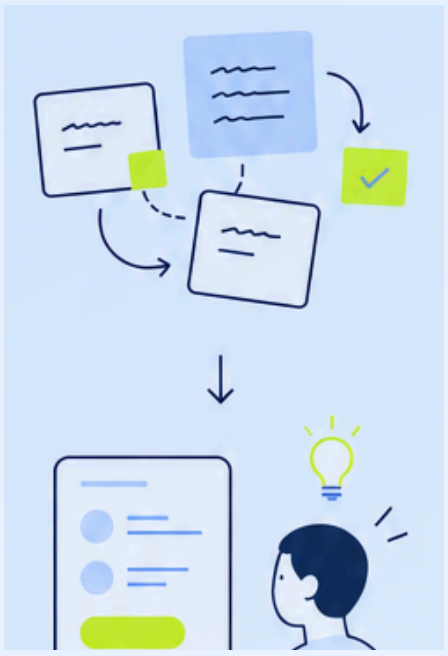
질문과 재검토



사람과 현장의 이야기를 듣고, 문제를 다양한 관점에서 다시 검토합니다.

02

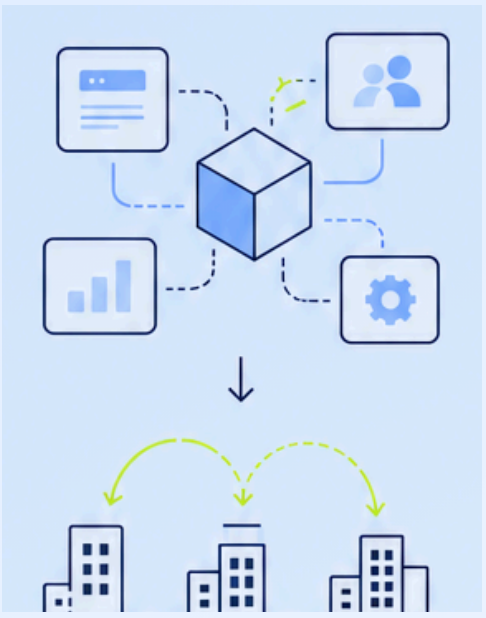
가설과 실험



핵심 가설을 세우고, 작은 실행으로 결과를 확인합니다.

03

구조와 확장



확인한 규칙을 구조로 남겨, 다음 사람이 발전시킬 수 있게 합니다.

주요 성과

업무를 개선하고,
다음 개선의 기반을 남겼습니다.



입출고 체계 개선

세미 ERP · 스프레드시트

배송 이슈 **70%↓**

커뮤니케이션 리소스 **50%↓**

SKU·입출고 기록과 규칙 → 팀원의 대시보드·실시간 확인으로 확장

기획서 ↗ 구현 예시 시트 ↗



물류 자동화

프로젝트당 평균

150분 → 10분

업무 규칙을 코드·웹으로 → 사용하며 문제를 세분화·수정

기획서 ↗ 구현 예시 웹 ↗



물류비 분석 개선

연평균 실절감

3.5억 원 이상

분석 기준을 반복 활용할 계산·예측 도구로 연결

기획서 ↗ 구현 예시 웹 ↗

‘기획서’는 당시 경험을 바탕으로 재구성한 개발 전달 문서입니다.
체험본은 원본 구조·규칙을 참고해 현재 제작한 가상 데이터 예시입니다.

개별 개선을 연결한
통합 서비스

상품·요청·일정·이력을 하나의 흐름으로 연결해, 다음 개선이 이어질 기반을 설계했습니다.

통합 프로토타입 ↗ 기획서 ↗

신규 설계·예시 체험 / 실제 운영 미도입



상품 · 요청 · 일정 · 이력을 하나의 흐름으로

문제 핵심 · 솔루션으로 읽는 사례

무엇을 문제로 보고, 어떻게 풀었을까요?

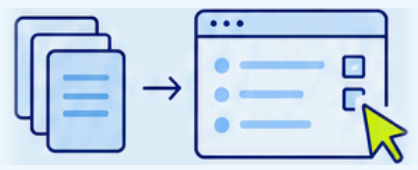
01 ERP 도입 전에, 무엇부터 같아져야 했을까? 05-07

문제 핵심 전달 오류와 공통 기준·공유 경험의 부재
솔루션 스프레드시트 - 세미 ERP 구현



02 불가능한 업무였을까, 규칙이 숨어 있었을까? 08-10

문제 핵심 복잡한 주문 가공 규칙이 개인 경험에 의존
솔루션 주문서 가공-검수 자동화 웹 개발



03 분석이 끝나도, 왜 같은 판단을 반복할까? 11-12

문제 핵심 분석 기준을 프로젝트마다 다시 적용
솔루션 물류비 비교-계산-청구무게 예측 도구 웹 개발 후 연동



04 작은 문제도 설계를 고민하면, 추후 응용이 가능하지 않을까? 13

운영 자동화



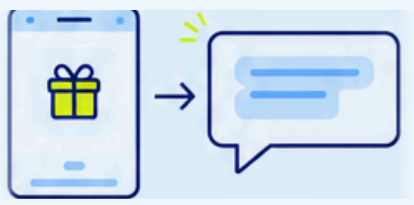
05 도구는 좋아졌는데, 왜 사람이 사이를 연결할까? 14-15

문제 핵심 요청·일정·이력의 분리
솔루션 통합 물류서비스



06 작은 기능 추가로 선물 경험을 연결할 수 있을까? 16

문제 핵심 기존 선물 행동의 서비스 연결 가능성
솔루션 fromm 선물하기 기능 추가 기획



문제 접근 방식 04 →
QA 프로세스 17 →
배움과 회고 18 →

본질이 무엇인지 질문합니다.

들은 이야기를 곧바로 해답으로 삼지 않고, 실행에서 다시 확인합니다.

출발점 실제 문제와 피해 확인

01 묻고, 관찰하고, 직접 해봅니다.

현장의 목소리를 듣고,
실제 과정을 경험하며 파악합니다.

02 정보를 꺼내 비교하고 연결합니다.

여러 출처의 사실관계를 비교하고
맥락과 관계를 드러냅니다.

03 본질이 무엇인지 질문합니다.

- 이것은 현상인가, 원인인가?
- 당연하게 받아들인 전제는 무엇인가?
- 다른 관점에서는 무엇이 달라지는가?

ERP를 원하는 이유는? → 시스템 부재인가, 다른 문제인가? → 지금 진정 필요한 것은?

현재 상황

- 무엇이 실제로 일어나는가?
- 누가 어떻게 겪고 있는가?

제약 조건

- 시간·예산·인력의 한계는?
- 어떤 의존 조건이 있는가?

필요한 것

- 지금 어떤 가치가 필요한가?
- 무엇부터 실행·확인할까?

04

문제 재정의·가설

무엇을 바꾸면 달라질지 정의

→

05

도구 응용·작은 실행

제약 안에서 시도할 방법 선택

→

06

결과·예외 확인·수정

배운 내용으로 다시 질문

공통 기준 · 명시된 규칙 · 재사용 도구 · 다음 사람의 개선

실행 결과에 따라 문제 해석과 가설도 다시 검토합니다.

CASE 01 · 입출고 체계 개선

배송 문제인데,
ERP를 기다리는 것이 답이었을까?

Notion 기반 요청·정보 전달에서 누락·혼재·커뮤니케이션 오류 발생

배송 이슈
월 0.2~0.3건

추가 배송비
월 20~30만 원

비즈니스 위기
월 0.5건

당시 의견 요약

CFO·재무팀

ERP가 도입되면 해결된다.
승인까지 기다려야 한다.

상품기획팀

시스템을 사면 될 것 같은데,
알아서 잘 해주세요.

그래서 질문을 더했습니다

01
ERP 도입을 원하는 이유는 무엇일까?
각 부서가 해결하려는 불편을 확인

02
시스템 부재가 문제일까, 아니면 다른 것이 문제일까?
상품 식별·요청 전달 인터뷰로 공통 기준과 공유 문제 확인

03
공통 기준과 공유만 해결하면 충분할까?
ERP가 왜 필요한지에 대한 공통 이해도 필요함을 확인

04
지금 당장 필요한 것은 무엇일까?
공통 기준을 경험하고 필요성을 이해하며, 당장 오류도 줄일 방식

다시 정의한 문제

**상품을 같은 기준으로 식별하고,
필요한 정보를 잘 전달할 구조와 사용 경험이 필요했습니다.**

공통 품번 시트 생성 ↔ 입출고 시트 연결

입출고 체계 개선 · 가설과 실행

완성된 답을 기다리기보다, 바꾸며 확인할 구조를 만들었습니다.

현재 상황

전달 오류가 계속되고, 필요한 정보와 예외도 더 알아가야 했습니다.

제약 조건

개발팀 리소스와 외주 승인을 확보하기 어려웠습니다.

필요한 것

전사가 함께 쓰며 항목과 규칙을 빠르게 수정할 방식이 필요했습니다.

가설

공통 항목과 기능을 유연하게 바꿀 수 있는 시트라면,
전사가 공통 기준에 적응하면서 문제를 발견하고 해결하기도 쉬울 것이다.

첫 구현에서 확인한 것

DB와 웹의 역할 구분을 응용해, 기준 정보와 입력·조회 시트를 나눴습니다.

01

기준 정보 시트

상품 이름·품번의 공통 기준

상품명	품번

→

02

입출고 요청 시트

필수·선택 정보로 요청 전달

요청일	수량

→

03

내역의 누적·조회

상품과 입출고 기록을 찾아 확인

입출고	내역

일부 반복 작업에는 Apps Script 활용

완벽한 소프트웨어란 없었습니다. 발견된 문제는 계속해서 수정했습니다.

누가 확인했는가?

승인 B열에
물류 담당자 이름

누가 수정할 수 있는가?

승인 후 요청자는
조회만 가능

끝난 업무는 무엇인가?

완료된 행을
회색으로 표시

최초부터 승인·권한·완료 규칙이 모두 있었던 것은 아닙니다.

입출고 체계 개선 · 결과와 남은 기반

공통 기준을 쓰는 경험이, 팀의 다음 개선으로 이어졌습니다.

배송 이슈

70% 감소

커뮤니케이션 리소스

50% 감소

상품 기준

SKU 정보의 DB화

공통 품번·상품 정보를 관리하고, 입출고 시트에서 같은 기준을 참조했습니다.

기록과 조회

입출고 내역의 누적

상품과 요청을 함께 찾고, 수불 추적·월말 정산에 활용했습니다.

업무 규칙

승인·완료 패턴

확인 주체, 수정 범위, 완료 표시를 운영 중 보완했습니다.

요청 부서

상품과 과거 요청을 찾고,
같은 기준으로 정보를 전달

물류팀

미완료 요청과 수불을 추적하고,
누가 확인했는지 파악

재무팀

입고 속성·출고 사유를 통해
정산에 필요한 맥락 확인

반복되는 확인을 한곳에서 볼 수 있다면?

상품·요청·상태의 관계를 먼저 정리하자, 팀원은 이를 바탕으로 간이 대시보드와 입출고 실시간 확인 기능을 더했습니다.

시트 기반 정보 구조화입니다. 별도 서버 DB를 구축했다는 의미는 아닙니다. 후속 대시보드는 팀원이 구현했습니다.

기획서 ↗ 구현 예시 시트 ↗

CASE 02 · 물류 자동화

주문 한 건을 보내는 데도, 여러 번의 판단이 필요했습니다.

출고 이슈
월 0.4~0.5건

팀 하루 평균 투입공수
200분 이상

월평균 손해
20~30만 원 이상

한 사람이 3개 버전의 앨범을 39개 구매한다면?

박스당 최대 20개. 먼저 배송 단위를 나누고, 각 박스에 보낼 버전을 배분해야 했습니다.

원본 주문 · 1행
앨범 39개
A / B / C 버전
수령인·주소 유지
박스 한도: 20개

→

첫 번째 박스
20개

두 번째 박스
19개

→

물류사용 주문서 · 6행
박스 1 · A 7 / B 7 / C 6
각 버전이 별도 상품 행
박스 2 · A 6 / B 6 / C 7
전체 합계: A·B·C 각 13개

이 판단을 주문 1,000건마다 수동으로 반복해야 한다면? 가공 방식 설명용 예시

이외에도… 당첨 수량 차감·추가, 사은품 동봉, 주문번호 연결, 물류사별 입력 형식까지 함께 맞춰야 했습니다.

담당자 인터뷰
“자동화는 불가능해요.”
조건이 겹치고, 상황에 따라 다르게 처리해야 한다는 설명이었습니다.

더 묻고, 직접 처리하며
어떤 판단이 반복될까?
주문을 함께 가공하며 처리 순서와 예외, 맞아야 할 결과를 구분했습니다.

처리 규칙
분할·버전·수량 계산

예외 조건
조건별 처리·수동 확인

검수 결과
수량·배송비·주소

물류 자동화 · 가설과 반복 확인

한 번에 완성하기보다,
큰 규칙부터 만들고 비교했습니다.

웹 전체를 먼저 만들기에는 시간이 부족했고, 복잡한 예외는 실제 결과를 비교해야 드러났습니다.

가설

반복 규칙을 하나씩 명시하고 수작업 결과와 비교하면, 시간을 줄이면서 주문서 가공의 정확도를 높일 수 있을 것이다.

01

큰 규칙부터 코드화

박스 분할과 버전 배분 등 반복 작업부터 구현

02

기존 담당자의 결과와 비교

수량, 배송비, 주소가 맞는지 함께 검수

03

발견한 오류 수정

예외를 추가하고 같은 주문으로 다시 확인

04

다음 규칙으로 확장

확보한 시간으로 더 복잡한 조건을 확인

비교 검수에서 발견한 오류

사은품이 별도 배송으로 처리되면 박스가 하나 더 생기고, 배송비가 두 배로 늘어날 수 있었습니다.

오류 위험 · 따로 보냄

앨범 박스 + 사은품 박스

추가 박스·배송비 발생 가능

개선 · 첫 앨범 박스에 연결

앨범 + 사은품을 함께 포장

코드 수정 후 같은 주문으로 재검수

수량
주문·가공 결과 대조

배송비
불필요한 추가 박스 확인

오배송
주소·연락처 확인

사은품 사례는 실제 지출이 아니라 비교 검수로 막아낸 오류입니다. 주요 오류와 예외가 정리된 뒤 웹 화면으로 확장했습니다.

물류 자동화 · 확장과 결과

직접 처리하던 순서를,
팀이 따라갈 수 있는 화면으로 옮겼습니다.

이전 · 사람이 반복

주문마다 판단하고 수정

수량 확인 → 박스 분할 → 버전 배분

당첨 수량 처리 → 사은품 연결 → 재검수

이후 · 도구와 사람이 분담

조건을 설정하고 결과를 검수

파일 선택 → 조건 설정 → 가공 실행

수량·박스·오류 확인 → 국내/해외 XLSX

프로젝트당 평균

150분 → 10분

팀 하루 평균 투입공수

200분 → 15분

당시 도구는 동료들이 각자 PC에서 Streamlit으로 실행했습니다.

처리 흐름 예시

01

입력

주문서와 당첨자 파일 확인

02

설정

앨범 버전·박스 용량 등 조건 지정

03

실행

분할·차감·추가 규칙 적용

04

검수·출력

사람이 대조한 뒤 국내/해외 XLSX 저장

도구에 남은 업무 지식

처리 순서, 발견한 예외, 수량 대조 기준을 코드와 화면의 단계로 남겼습니다. 웹 사용 중에도 문제가 생긴 단계를 더 쪼개 고쳤습니다.

AI와 코드 작업을 진행했고, 수량 대조와 최종 배송 검수는 사람이 맡았습니다.

기획서 ↗ | 구현 예시 웹 ↗ | 수작업·자동화 비교 영상 ↗

현재 체험본·영상은 당시 규칙을 바탕으로 재구성한 가상 데이터 예시입니다.

물류비 분석 개선 · 문제정의와 코드 실험

우리가 쓰던 A업체는, 실제로 더 저렴했을까?

A·B 두 3PL을 이용했지만, 프로젝트를 대부분 A업체에 맡겼습니다. 같은 조건으로 비교하면 선택이 달라질지 궁금했습니다.

분석 프로젝트

물류를 한눈에 보기

운임·작업비·국가·청구무게 등 흩어진 정보를 수치로 정리하고 비교했습니다.

여기서 얻은 아이디어

매번 다시 계산할 필요가 있을까?

정리된 기준을 코드로 만들면, 다음 프로젝트에도 바로 적용할 수 있겠다고 봤습니다.

가설

분석에서 정리한 비교값을 계산 도구로 만들면, 관행에 의존하던 선택을 실제 프로젝트 조건에 맞춰 검토할 수 있을 것이다.

비교 항목	A·B에 공통으로 적용한 기준
배송비	국가·청구무게 구간
작업비·부자재	상품 수량·포장 조건
실행 가능성	일정·품질·업체의 작업 가능 여부

먼저 일부 계산을 코드로 실험하고, 실제 배송 조건과 비교했습니다. 이후 Streamlit에 기능으로 붙였습니다. 현장 방문·인터뷰·직접 포장은 박스 공간과 작업 조건을 보완하는 아이디어로 이어졌습니다.

분석 개선 전체 과정 ↗ 구현 예시 웹 ↗

물류비 분석 개선 · 확장과 결과

분석을 웹 도구로,
청구무게 예측까지 연결했습니다.

01
분석·코드 실험

→

02
Streamlit 계산·비교

→

03
예측값 수동 반영

비교 도구 · 기능 구성 개념도

입력	결과
프로젝트·상품 구성	업체별 운임·작업비
국가·박스 조건	동일 기준으로 비교
일정·품질 조건	선택 가능한 업체 확인

가장 저렴한 업체가 항상 가능한 선택은
아니었습니다.

청구무게 예측 · 데이터 흐름

3PL이 실제 청구한 박스 무게

여러 상품이 담긴 배송 데이터 전처리

랜덤 포레스트 학습

예측 결과를 수동으로 도구에 반영

상품 실측 무게가 아닌, 포장이 포함된 청구무게를 예측했습니다.

비용 분석을 통한 연평균 실절감

3.5억 원 이상

도구를 통한 연평균 실절감

1.5억 원 이상

별도 성과이며 합산하지 않음 · 모델 단독 효과로 제시하지 않음

다음 프로젝트에서도 쓸 수 있는 판단 도구로 남겼습니다.

실제 청구와 오차를 기록하며 비교했고,
서비스 내 실시간 학습과는 구분합니다.



기획서 ↗

개선 분석 전체 자료 ↗

운영 자동화 · 입력 구조의 재사용

입력을 자동화한 방식은, QA 준비에도 응용할 수 있었습니다.

상품 정보 7개 이상, 속성 3개, 옵션 4개 이상을 서로 다른 단계에서 입력하고 검수했습니다. 이름·가격·재고·판매 시간의 실수는 비즈니스 문제로 이어졌습니다.

PO

운영 리소스를 줄이고
QA·안정성에 집중

CEO

입력 오류가 만드는
비즈니스 위험 감소

개발

백오피스를 개선할
리소스 부족

입력과 검수를 모두 사람이 해야 할까?

상품기획팀이 전달한 정보를 기계가 입력하고, 사람은 원본과 결과를 비교하는 데 집중할 수 있다고 봤습니다.

01

Excel 입력

상품기획팀이 등록 정보 작성

02

코드 변환

Apps Script로 Playwright 코드 생성

03

자동 입력

기존 백오피스의 등록 동작 실행

04

사람 검수

원본·백오피스·실제 사이트 비교

1인 하루

180분 → 30분

상품 등록

5일 → 1일

비즈니스 오류

월 0.2건

이전 월 0.3~0.4건

백오피스가 복잡하다는 같은 문제를 다시 봤습니다.

QA용 상품도 가격·재고·판매 시간을 바꿔야 했습니다. Excel 값만 바꿔 입력 동작을 실행하는 방식을 테스트 준비에 응용했습니다.

내가 응용한 것	테스트 상품·조건 설정
다음 사람이 발전시킨 것	PO의 운영 자동화 고도화

통합 물류서비스 · 문제 재검토와 가설

도구를 모으기 전에,
정보가 바뀌는 순간을 연결했습니다.

AS-IS

스프레드시트	Slack	캘린더	각 PC 자동화
상품·입출고 요청	논의·변경	일정	주문서 가공

요청과 일정을 따로 입력하고, 수정 내역과 상품 이력은 사람이 찾아 연결했습니다.

01 같은 화면에 모으기만 하면 반복 확인이 사라질까?

02 같은 정보를 여러 곳에서 고치면 오류는 계속되지 않을까?

03 승인·변경·완료가 어디까지 함께 바뀌어야 할까?

가설

상품과 요청을 같은 데이터로 연결하고, 승인·변경·완료 시 일정·재고·이력을 함께 반영하면 재입력과 확인 부담을 줄일 수 있을 것이다.

TO-BE · 사건과 결과를 함께 설계

요청 승인·날짜 변경	전사 캘린더 반영
입출고 완료	현재고·상품 수불 이력 반영
수정·논의	이전 값·사유·담당자 기록

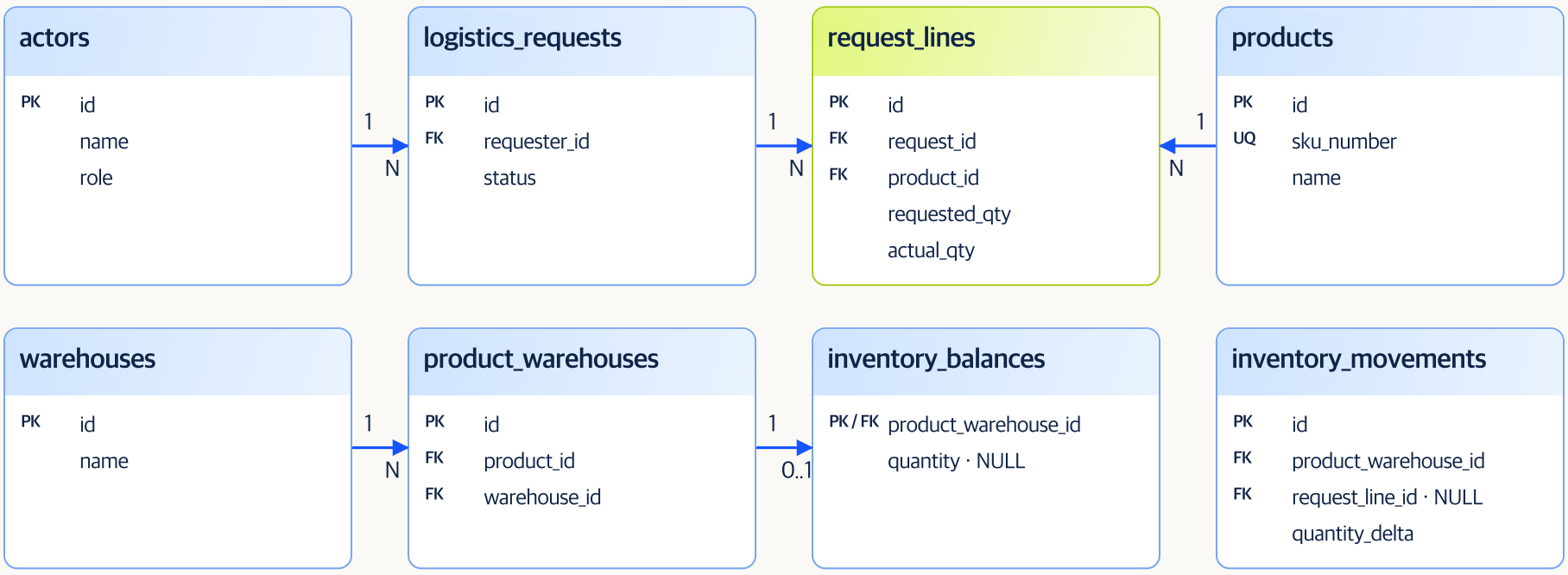
통합 서비스 체험 ↗ 기획서 ↗

신규 설계·프로토타입입니다. 시연에서는 요청 상태와 배송 일정을 한눈에 볼 수 있다는 피드백을 받았으며, 실제 도입 성과와 구분합니다.

통합 물류서비스 · 설계 및 시현

화면을 넘어,
데이터와 정책까지 설계했습니다.

핵심 테이블 ERD · 논리 설계



추가 FK: 상품·창고 매핑 → 상품 / 수불 이력 → 상품·창고 매핑 및 요청 품목

요청 1건·상품 2종 → 요청 1행 + 요청 품목 2행

핵심 테이블 발췌 · JSON 구현과 구분 / PK 기본키 · FK 참조키 · UQ 고유값

C-0 서비스 개요·아키텍처
전체 관계·역할·공통 정책



기획서 ↗

C-1 상품 등록·재고 관리
공통 상품 기준과 변경 이력



기획서 ↗

C-2 입출고 요청·일정 관리
요청·승인·완료와 캘린더



기획서 ↗

C-3 주문서 자동화·비용 예측
앨범·MD 가공과 국내·해외 XLSX



기획서 ↗

C-4 3PL·운영 기준 관리
업체·요금·상시판매 규칙·계정



기획서 ↗

통합 프로토타입 ↗ | 기획서 ↗ | 전체 DB·정책 ↗ | 개발 범위·일정 ↗

저장·요청·승인 분리 / 권한·취소·재고 예외 / Slack 봇 게시 스펙

fromm 선물하기 · 기능 추가 기획

채팅에 선물하기 기능을 더해, 기존 구매 경험으로 연결합니다.

팬에게 외부 쿠폰을 공유하던 아티스트의 행동에 주목했습니다. 채팅의 + 메뉴에 선물하기 진입점을 추가하고, 기존 스토어 웹뷰·결제·주문·환불 흐름을 활용하는 기능을 기획했습니다.

fromm 채팅

아티스트의 + 메뉴

팬들에게 선물을 보내고 싶어요.

\+ **선물하기**

추가할 최소 기능

스토어 웹뷰로 연결

비공개 상품 진입

가격·수량 옵션 선택

구매 후 쿠폰 발급 연결

01

채팅 + 메뉴

아티스트가 선물하기 선택

02

기존 스토어

상품·수량 선택 후 아티스트 결제

03

쿠폰 전달

발급된 쿠폰을 아티스트에게 제공

04

직접 공유

아티스트가 팬 채팅방에 공유

최소 기능으로 확인할 가설

구매와 공유가 간편해지면 선물 행동이 서비스 안으로 이어지고, 팬 경험과 수수료 수익을 함께 확인할 수 있을 것이다.

확인

팬 반응·CS

아티스트 피드백·구매

수정·확대

상품 구성·동선 보완

운영 가능한 범위 확장

중단

강매·구매 유도 인식 등

부정적 반응 확인

기능 추가 기획서 ↗ 화면 예시 ↗

이번 포트폴리오를 위한 신규 기능 제안입니다. 쿠폰 발급 연동과 중복 사용 제한은 추가 확인이 필요합니다.

보조 사례 · QA 프로세스 개선

QA가 시작되는 시점을 앞당겨, 검증의 우선순위를 함께 맞췄습니다.

운영 자동화로 시간을 확보한 뒤, 기획 단계에서 핵심 흐름과 위험을 먼저 파악하고 QA 준비·검수에 연결했습니다.

> **질문** 검수를 더 많이 하면 충분할까?
>
> **재검토** 무엇이 중요한지 늦게 알면, 준비와 우선순위부터 흔들리지 않을까?
>
> **가설** 기획부터 검증 기준을 맞추면 중요한 위험을 더 일찍 확인할 수 있을 것이다.

01

기획 단계 참여

이용 흐름·일정·우선순위 파악

02

간이 TC

핵심 분기와 확인할 위험 정리

03

QA 브리핑

기획·개발·QA가 검수 범위 공유

04

세부 검수·재확인

구체적 TC와 수정 결과 확인

준비시간 감소로 확보한 검수 여력

약 2배

운영 전후 관찰한 QA 이슈

20~30% 감소

상황에 따라 검수의 깊이와 순서를 조절했습니다.

변경이 잦은 기능	주요 분기와 영향부터 확인
급한 배포	일정·비즈니스 영향·우선순위 공유
반복되는 환경 준비	상품 입력 자동화를 QA 설정에 응용

인원을 두 배로 늘린 결과가 아닙니다. 이슈 감소는 운영 전후 관찰값이며 단일 조치의 인과 효과로 단정하지 않습니다. 책모임 등 비공식 소통 시도도 별도로 진행했습니다.

QA 상세 사례 ↗ QA 원본 자료 ↗

배움과 회고

본질을 다시 묻고,
실행으로 가설을 확인했습니다.

01 주어진 해답도, 내 첫 판단도 다시 확인하기

"ERP면 해결된다", "자동화는 불가능하다"는 설명을 그대로 받아들이지 않았습니다. 인터뷰·관찰·직접 처리로 근거를 모으고, 진짜 문제가 무엇인지 다시 질문했습니다.

02 가설을 검증하며 문제의 본질에 더 가까워지기

작은 단위의 구현과 결과 비교는 완성도를 높이는 동시에, 무엇이 핵심이고 어떤 제약이 중요한지 알아가는 과정이었습니다. 실행 결과에 따라 문제 정의와 가설도 고쳤습니다.

03 반복되는 문제의 원리를 찾아 다음 개선으로 연결하기

상품·요청·상태가 어떻게 연결되는지, 입력과 검수는 무엇이 다른지 살폈습니다. 이 관계를 구현으로 남기자 팀원은 대시보드를, PO는 자동화의 후속 고도화를 이어갈 수 있었습니다.

다음에는 더 잘하고 싶은 것

측정의 기준	개선 전후의 기간·비교 조건·집계 기준을 더 명확히 정하고 기록하기
필요를 앞서 발견하는 기획	요구사항을 듣고 구현하는 데서 더 나아가, 사람들이 다음에 필요로 할 것을 예상하고 제안·검증하기

사람과 현장에 묻고, 아이디어를 연결해
기획으로 실현합니다.

웹 포트폴리오 ↗ 통합 서비스 ↗