

PORTFOLIO

권기영.



Project Manager

ttu445@gmail.com · 010-4953-9209

BUILT WITH AI

이 포트폴리오 웹사이트 역시 AI를 통해 직접 제작했습니다.

portfolio-giyeongkwon.pages.dev

국회 정책비서, 게임 기획자, PM을 거치며 쌓은 다각도의 도메인 지식을 바탕으로, AI를 통해 '아이디어를 내는 사람'에서 '솔루션을 직접 구현해 변화를 만드는 사람'으로 진화했습니다.

회의실 예약 봇부터 게임 전용 번역 시스템까지, 현장의 비효율을 기술로 정면 돌파하며 AI 에이전트의 실질적 가치를 증명해왔습니다.

"안 된다"는 한계 대신 "되는 구조"를 설계하는 PM으로서, AI를 단순한 도구가 아닌 동료 및 저의 역량을 극대화하는 강력한 '제품'으로 정의하고자 합니다.

INCLUDED PROJECTS

01 이터널 리턴 전용 AI 번역 시스템

02 Slack 기반 업무 완결형 AI 회의실 봇

03 성우 녹음 일정 자동 등록 봇

04 Jira 스웸레인 자동 생성 봇

05 콜라보 종합상황판

이터널 리턴 전용 AI 번역 시스템.

우리 게임의 컨텍스트를 이해하는 전용 시스템으로 번역 파이프라인을 혁신하다

ROLE	PERIOD	TYPE
기획설계 · 도입 추진 · 운영 설계 · 시스템 개선	2026.01 – Present	사내 시스템 도입 및 운영 프로세스 혁신

01 · PROBLEM

업무 폭증과 대응 속도의 물리적 한계

대형 콜라보레이션이나 시즌 업데이트 시기에는 번역 업무량이 평시의 임계치를 상회하게 됩니다. 이 과정에서 다음과 같은 잠재적 리스크와 실질적인 페인 포인트가 발생할 수 있습니다.

물리적 대응의 한계 — 업무량이 인게임 반영 지연을 초래할 수 있을 만큼 방대해지면서, 실무자들이 매번 야근을 불사하며 일정을 맞추는 상황이 반복되었습니다.

글로벌 대응 속도 저하 — 팀에서 최선을 다해 대응했음에도 불구하고, 긴박하게 돌아가는 라이브 서비스 특성상 글로벌 공지 등이 유저 기대치보다 늦게 나가는 경우가 발생하며 속도감에 아쉬움이 남기도 했습니다.

시중의 범용 LLM이 가지는 한계

범용 AI는 게임 도메인 고유의 '약속된 언어'를 알지 못합니다. 예를 들어, 캐릭터 스킬명인 "최후의 발악"을 단순 LLM에 넘기면 문자 그대로 "Last Struggle"로 직역합니다. 하지만 우리 게임의 컨텍스트를 반영한 의도된 번역은 "Last-ditch Effort"여야 합니다. 이러한 고유 명사와 뉘앙스의 오역이 빌드 전체에 누적되면 글로벌 서비스의 몰입도가 근본적으로 흔들리게 됩니다.

02 · APPROACH

'이터널 리턴'만을 위한 전용 로컬라이제이션 시스템

단순한 텍스트 변환기를 넘어, 우리 게임에 완벽하게 최적화된 다층 구조의 전용 시스템을 설계하는 것을 목표로 삼았습니다.

전용 엔진 설계 — 범용 LLM의 자연스러움 위에 16만 개 이상의 사내 마스터 용어집을 '강제 치환 레이어'로 얹어, 우리 게임만의 고유한 톤앤매너를 유지하도록 했습니다.

하이브리드 워크플로우 — 번역가가 '기준'을 세우고 AI가 '반복'을 해결하는 구조를 통해, 사람이 물리적 한계에 부딪히지 않고도 퀄리티를 유지할 수 있는 환경을 구축했습니다.

사용자 피드백 루프 운영 — 현장에서 발견된 문제를 빠르게 시스템 개선으로 연결하기 위해, 전용 피드백 채널을 개설해 운영했습니다.

🔒 rnd_ai번역피드백

고객님이 이 채널을 생성한 날짜는 날짜: 2025년 12월 18일입니다. 🔒 rnd_ai번역피드백 채널의 맨 첫 부분입니다.

👤 채널에 사람 추가

📝 설명 추가

✉ 채널에 이메일 보내기

📄 템플릿 선택

2025년 12월 18일 ▾



권기영 오전 11:12

rnd_ai번역피드백에 참여했습니다. 또한 초대를 통해 [redacted] 명의 다른 사용자가 참여했습니다.



권기영 오전 11:14

안녕하세요 구두로 말씀드렸던

ai번역 관련 피드백을 위한 채널입니다.

날것의 피드백도 좋으니 사용해하시고 많은 피드백 부탁드립니다.

👍 3 🗨️ 1 😊

04 · IMPACT

현장의 문제를 해결하고 공식 프로세스가 되다

효율과 퀄리티가 현장에서 즉각적으로 증명되며, 본 시스템은 이터널 리턴의 정규 워크플로우로 완전히 자리 잡았습니다.

처음에는 번역팀 캐파를 보완하기 위한 **임시 협조 요청**으로 출발했지만,



권기영 오전 11:44

@channel

AI 번역 직접 실행 협조 요청

안녕하세요 현재 번역팀 공식 확보가 어려워 부득이하게 AI 번역을 병행하고 있습니다

이후 효율과 퀄리티가 검증되며 **상시 운영 프로세스로 정식 전환**됐습니다.



권기영 오후 5:14

[공지] L10N AI 번역 프로세스 상시 운영 전환 안내 @channel

안녕하세요. 지난 [redacted] AI 번역 프로세스에 적극 협조해 주셔서 감사합니다.

해당 기간 운영 결과 AI 번역 방식을 정식 프로세스로 도입하여 상시 운영하고자 합니다.

앞으로 진행되는 번역 건에 대해서는 아래 가이드를 참고하시어 직접 진행해 주시면 감사하겠습니다.

1. 적용 대상 및 예외 사항

- **적용:** 금일부터 진행되는 일반적인 번역 건 전체 (상시 적용)
- **예외:** 신규 그로서리(Glossary) 등록이 필수적인 항목은 기존과 같이 별도 요청 바랍니다.
 - 예시: 콜라보레이션, 신규 아이템 네임, 기존 존재하지 않는 신규 그로서리가 필요한 고유 명사 등

2. 진행 방법

- **프로그램 링크:** [L10N 자동번역 시트 링크]
- **사용법:** 시트 내 '사용법' 탭 확인 후, '템플릿' 탭을 복사하여 본인용 탭을 별도로 생성하여 사용 (동시 작업 가능)

3. 시트 업데이트 (필수)

- 번역 완료 후, 원본 L10N 시트의 # 번역 필요 텍스트 열에 상태값 입력해 주시기 바랍니다.
- **상태값:** VERIFY_REQUEST_EN / VERIFY_REQUEST_JP / VERIFY_REQUEST_CN

4. **피드백 및 개선 안내** 업무를 진행하시면서 번역 품질이나 시트 사용성 등 개선이 필요한 부분이 있다면 언제든지 의견 주시기 바랍니다. 전달해 주신 피드백은 즉각적인 반영이 어려울 수 있으나, 놓치지 않고 꾸준히 업데이트하여 더 나은 환경을 제공할 수 있도록 하겠습니다. 원활한 업무 진행을 위해 기획자분들의 많은 협조 부탁드립니다. 감사합니다. 🙏

👍 6 ✅ 1 😊

05 · LEARNINGS

PM의 도메인 지식이 곧 AI의 품질이다

이 프로젝트는 “기술보다 중요한 것은 기술을 어디에, 어떻게 붙일 것인가를 아는 안목”이라는 것을 증명한 사례였습니다.

번역팀과 보이스 녹음 업무 전반을 매니징하며 수많은 자료가 외주 업체와 오가는 복잡한 파이프라인을 직접 관리했습니다. 타이트한 스케줄 속에서 발생하는 병목 구간을 피부로 느꼈고, 수만 건의 데이터를 검토하는 과정에서 우리 게임만의 고유한 컨텍스트와 도메인 지식을 깊게 쌓을 수 있었습니다.

이러한 현장의 경험이 있었기에 16만 개의 용어집을 시스템에 촘촘히 엮어낼 수 있었고, 결과적으로 AI가 단순한 실험적 툴을 넘어 실제 조직의 임팩트로 작동하게 만들었습니다. PM이 실무의 흐름과 아픔을 깊이 이해하고 있을 때, AI 기술은 비로소 조직의 진정한 해결책이 된다는 확신을 얻었습니다.

Slack 기반 업무 완결형 AI 회의실 봇.

조율부터 예약까지, Slack을 벗어나지 않는 업무 완결형 Agent

ROLE	PERIOD	TYPE
기획 · 개발	2026.03 – Present	사내 커뮤니케이션 흐름에 결합된 업무 자동화

01 · PROBLEM

조율과 예약 사이에서 끊기는 워크플로우

님블뉴런은 NAVER Works Calendar를 통해 회의실을 예약합니다. 하지만 실제 회의 조율은 대부분 **Slack 채널**에서 이루어집니다. 특히 개발리뷰 시즌에 많은 회의로 일정이 바쁜 **개발 리더**들이 포함된 회의의 경우, [리더들의 빈 시간 확인 → 회의실 가용 상태 대조 → 참석자 전원 일정 확인] 과정을 수동으로 반복해야 했습니다. 슬랙 스레드에서 논의하다가 예약을 위해 외부 툴로 이동하는 과정 자체가 맥락을 끊는 업무 비용이자 병목이었습니다.

02 · APPROACH

Slack 안에서 업무가 종결되는 Agent 구조

단순히 질문에 답하는 챗봇을 넘어, 논의가 일어나는 **Slack 안에서 즉시 업무가 종결되는 Agent 구조**를 설계했습니다.

사용자 중심의 인터페이스 — 새로운 툴에 적응할 필요 없이, 사용자가 가장 익숙한 **Slack에서 평소 대화하듯 자연어로 요청**하는 방식을 채택했습니다. “이번 주 금요일 오후에 아트디렉터, 프로그래머1, 기획자1님 모두 가능한 시간에 회의실 어디야?”라는 한 문장이면 충분합니다.

지능형 교차 분석 — AI가 요청을 해석해 특정 참석자들의 개인 캘린더와 회의실 현황을 실시간으로 교차 분석하고, 모두가 만족하는 최적의 옵션을 도출합니다.

시스템 자동화의 완결 — 슬랙에서의 대화로만 끝나지 않고, 최종 확정 시 **NAVER Works Calendar에 실제 일정 등록**까지 자동으로 수행합니다. 이를 통해 ‘커뮤니케이션’과 ‘행정 입력’ 사이의 간극을 완전히 제거하는 것을 목표로 했습니다.

03 · BUILD

실제 문제를 해결하는 핵심 기능

- 자연어 기반 컨텍스트 파싱** — 슬랙 스레드 내 날짜, 시간, 참석자 정보 추출
- 교차 일정 최적화** — 특정 참석자(개발 리더 등)의 빈 시간과 회의실 가용 상태를 실시간 대조
- 인터랙티브 UI 제공** — 분석된 가용 시간대 중 사용자가 선택할 수 있는 슬랙 UI 버튼 제공
- API 통합 및 정합성 유지** — NAVER Works API를 통한 일정 등록 및 외부 수정 사항 실시간 반영
- 연속성 있는 대화 처리** — 슬랙 스레드 문맥을 유지하여 “그때 말고 한 시간 뒤로 변경해줘” 같은 후속 요청 처리



권기영 3월 19일 오전 11:00

회의실봇 안내

회의실 예약/조회를 도와주는 봇입니다.

필요한 채널에서 회의실 봇을 초대 후 @회의실봇 멘션하거나

@회의실봇 DM으로 자연어 질문하면 됩니다.

사용 예시

- 오늘 회의실 현황 → 시간대별 예약되어있는 회의실 확인
- 내일 오후 2시 빈 회의실 → 빈 회의실 목록 + 예약 버튼 표시
- 오늘 홍길동 김철수 이영희 전부 가능한 시간 → 여러 사람 공통 가능 시간 확인
- 이번주 목요일 전체 예약 현황 → 누가 어디 예약했는지 확인

예약 방법

빈 회의실 조회하면 각 회의실별 예약 버튼이 나옵니다. 버튼 누르고 회의 제목/참석자만 입력하면 네이버 워크플레이스 캘린더에 자동 등록됩니다.

시간/날짜는 "오늘", "내일", "이번주 금요일", "오후 3시" 같이 자연어로 말하면 알아서 인식합니다.

혹시나 작동 상에 문제가 생기거나, 혹은 개선 피드백 혹은 문의사항 있으면 @권기영 멘션 주세요!

04 · IMPACT

조율 단계가 사라지고 실행만 남다

여러 명의 캘린더를 일일이 대조하며 회의실을 찾던 번거로움을 없앴습니다. 사용자는 슬랙을 벗어나지 않고도 '가장 합리적인 예약 가능 시간'을 추천받음으로써, 의사결정에서 실행(예약)까지의 리드타임을 획기적으로 줄였습니다. AI를 단순한 정보 제공자가 아닌, 복잡한 운영 예외(충돌/수정)를 처리하는 효율적인 **운영 보조 인력**으로 포지셔닝한 사례입니다.

05 · LEARNINGS

AI의 품질은 '모델'보다 '문제 정의와 예외 설계'에서 온다

이번 프로젝트를 통해 생성형 AI의 가치는 화려한 모델 성능보다, 실제 업무 현장의 복잡한 예외 처리(데이터 정합성, 일정 충돌 로직 등)를 얼마나 꼼꼼하게 설계하느냐에 달려 있다는 것을 체감했습니다. 이후 자동화 도구를 기획할 때도 '어떤 모델을 쓸까'보다 '사용자의 흐름 속에서 어떤 예외가 발생할까'를 먼저 고민하는 습관을 얻게 되었습니다.

성우 녹음 일정 자동 등록 봇.

Slack 확정이 곧 Google Calendar 등록과 리마인드로 이어지는 외주 협업 자동화

ROLE	PERIOD	TYPE
기획 · 개발	2026.03 - Present	외주 스튜디오와의 협업 비효율 개선

01 · PROBLEM

대화 속에 흘러가 버리는 일정과 행정 부채

성우 녹음 일정은 외부 스튜디오와 가안 조율 후 Slack에서 사내 실무자들의 최종 확인을 받아 확정됩니다. 문제는 확정 이후, 제가 직접 Google Calendar를 켜서 수동으로 입력해야 외주 스튜디오 PM과 싱크가 맞는다는 점이었습니다. 여기에 대본 전달 같은 내용도 일정에 따라 직접 챙겨야 했기에, 이러한 행정 업무의 연쇄는 늘 심리적 부채이자 업무 흐름을 끊는 병목이었습니다.

02 · APPROACH

이모지 반응만으로 등록되는 마찰 없는 흐름

제가 수행하던 반복 업무를 시스템에 이관해, "확정 선언"이 곧 "데이터 등록"이 되는 직관적인 흐름을 설계했습니다. 슬랙 메시지에 이모지 반응 하나만 달면 날짜·캐릭터·시간 정보를 파싱해 Google Calendar에 자동 등록합니다. 또한, 일정에 맞춰 대본 등 필수 자료 전달을 개인 DM으로 재촉하는 리마인드 시스템을 구축하되, 버튼 토글식으로 보낸 자료는 다시 리마인드 오지 않게 하여 알림 피로도를 낮췄습니다.

03 · BUILD

실제 문제를 해결하는 핵심 기능

- Slack 메시지 이모지 반응 기반의 등록 트리거
- 자유형 텍스트에서 일정 데이터 파싱 — 성우·시간·캐릭터 등
- 외주 스튜디오 공유용 Google Calendar 등록·삭제 동기화
- D-Day 기반 리마인드 DM — 자료별 버튼 토글 체크리스트



권기영 3월 11일 오후 4:21

[나자봇 도입 했습니다.]

voice-over 채널에 일정비서 나자를 추가했습니다.

도입 배경

- Slack 채널에 공유되는 일정을 매번 수동으로 Google Calendar에 등록하는 번거로움을 줄이기 위해 제작했습니다.(제가 하던 일)

작동 방식

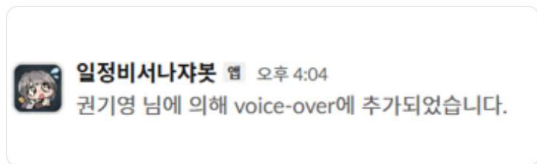
1. Slack 채널에 일정 내용이 담긴 메시지를 작성합니다.
2. 해당 메시지에 이모지 반응을 달면 자동으로 Google Calendar에 등록됩니다.
3. 메시지 내용이 수정되면 캘린더 일정도 자동으로 업데이트됩니다.
4. 이모지를 제거하면 캘린더 일정도 삭제됩니다.

AI 기반 자연어 파싱

- 고정된 입력 형식 없이 자유롭게 작성한 메시지에서 Google Gemini AI가 날짜, 시간, 제목, 참석자를 자동으로 인식합니다.
- 형식에 맞춰 작성할 필요 없이 평소대로 일정을 공유하면 됩니다.

등록 완료 시 봇이 스레드에 캘린더 링크를 답장합니다. (편집됨)

image.png ▾



04 · IMPACT

비정형 텍스트가 구조화 데이터로, 그 위에 후속 액션이 흐르다

슬랙에서의 “내부 확정”이 별도 작업 없이 “대외 공지”와 “사후 관리”로 직결되는 체계를 만들었습니다. 캘린더 복사-붙여넣기 같은 단순 반복 행정을 제거하고, 시스템이 마감 기한을 먼저 챙겨주어 에셋 누락 리스크를 차단했습니다. 도구 간 이동에 따르는 인지적 비용을 없애고, 본질적인 업무에 더 집중할 수 있는 환경을 스스로 구축했습니다.

05 · LEARNINGS

마찰 줄이기는 입력만이 아니라 후속 흐름까지의 문제

입력의 마찰만 줄여서는 사용자가 도구 안에 머물지 않는다는 걸 확인했습니다. 등록 이후의 리마인드와 체크리스트까지 같은 흐름 안에서 처리되어야 비로소 자동화의 가치가 완성됩니다. 서로 다른 도구를 사용하는 조직 간의 ‘브릿지 역할’을 자동화함으로써, 외부 조직과 제 개인의 리드타임을 함께 단축할 수 있었습니다.

Jira 스웸레인 자동 생성 봇.

새 버전 발행을 감지해 Jira Kanban 보드에 스웸레인을 자동 생성하는 운영 자동화

ROLE	PERIOD	TYPE
기획·개발	2026.03	Jira 운영 반복 업무 자동화

01 · PROBLEM

버전이 생길 때마다 반복되던 보드 운영의 수작업 병목

라이브 서비스 운영에서는 새 버전이 생길 때마다 Jira Kanban 보드에 해당 버전용 스웸레인을 추가해야 했습니다. 이 작업은 단순해 보이지만, 여러 보드에 반복 적용해야 하고 누락 시 실무자들이 새 버전 기준으로 업무를 정렬하지 못하는 문제가 있었습니다. 특히 버전 추가는 주기적으로 발생하는데, 매번 Jira 보드 설정에 직접 들어가 이름을 만들고 JQL을 넣고 순서를 맞추는 과정은 운영 집중도를 떨어뜨리는 소모적 관리 업무였습니다.

02 · APPROACH

"새 버전 생성"이 곧 "보드 운영 반영"으로 이어지도록 설계

반복되는 보드 관리 작업을 사람의 기억이나 수동 대응에 맡기지 않고, 새 버전의 생성 자체를 트리거로 삼아 자동 반영되도록 만들었습니다. Jira의 버전 목록을 주기적으로 감시해 새 버전이 확인되면, 지정된 여러 Kanban 보드에 해당 버전용 스웸레인을 일괄 생성하는 흐름을 설계했습니다. 단순 생성에 그치지 않고, 이미 존재하는 스웸레인과 비교해 더 높은 버전만 추가되도록 하여 중복 생성이나 과거 버전 오염도 막았습니다.

03 · BUILD

실제 운영에 맞춘 핵심 기능 구현

- Jira 버전 목록 주기 감시 — 새 버전 자동 탐지
- 여러 보드 대상 스웸레인 일괄 생성
- 기존 스웸레인 조회 후 최신 버전보다 높은 경우에만 추가
- Puppeteer 기반 Jira UI 자동화 및 로그인 세션 유지
- 수동 실행 (`node index.js 12.0.0`) + 자동 감시 (`npm run watch`) 이중 지원

04 · IMPACT

버전 관리가 개인 작업이 아니라 시스템 반응이 되다

새 버전이 나올 때마다 사람이 Jira 보드 설정을 반복적으로 손보던 흐름을 제거했습니다. 그 결과 보드 운영 반영이 더 빨라지고, 여러 보드 간 적용 누락 가능성도 줄였습니다. 특히 업무툴의 관리성 문제를 사람의 꼼꼼함으로 해결하는 대신, 시스템이 먼저 반응하도록 바꾸면서 버전 단위 업무 정렬의 일관성을 확보했습니다. 작지만 자주 발생하는 관리 업무를 줄여 팀의 운영 리듬을 더 매끄럽게 만들었습니다.

05 · LEARNINGS

운영 자동화의 핵심은 기능 구현보다 '누락 가능성 제거'에 있다

이 작업을 통해 운영 자동화는 시간을 몇 분 아끼는 문제를 넘어서, 반복 업무에서 발생하는 누락과 편차를 구조적으로 제거하는 일이라는 점을 확인했습니다. 또한 Jira처럼 완전한 공개 API만으로 해결되지 않는 도구는 UI 자동화와 내부 API 활용을 함께 고려해야 실무적인 해법이 된다는 것도 배웠습니다. 결국 중요한 것은 거창한 시스템이 아니라, 팀이 반복적으로 겪는 작은 불편을 정확히 제거하는 자동화였습니다.

콜라보 종합상황판.

시트·슬랙·Jira·드라이브 사이의 공백을 메우는 협업 허브 — 산재된 정보가 단일 화면으로 응집되는 게임 콜라보 운영 도구

ROLE	PERIOD	TYPE
기획 · 개발	2026.05 - Present	콜라보 포스트모템 액션 아이템 실행 — 외주·내부·IP 홀더 협업 비효율 개선

01 · PROBLEM

정립된 프로세스와 실제 현장 사이의 깊은 괴리

이전 콜라보 운영 회고에서 가장 두드러진 부채는 **기존 프로세스와 실제 업무 흐름 사이의 깊은 괴리**였습니다. 명문화된 절차는 있었지만 실제 작업물은 개인 로컬에, 피드백은 슬랙에, 일감은 Jira에 파편화되어 있었습니다. “최신 버전이 어디인가”라는 단순한 질문에도 도구 4개를 횡단해야 했고, 그 사이 IP 홀더의 핵심 피드백은 빠른 대화 흐름 속으로 휘발되곤 했습니다. 프로세스가 실제 업무 속도를 따라가지 못해 정보가 겹도는 상황이었습니다.

02 · APPROACH

한 화면에서 모든 컨텍스트가 종합되도록

도구를 늘리는 대신 **도구 간의 다리를 놓아 괴리를 좁히는 방향**을 택했습니다. ‘한 콜라보 = 한 화면’이라는 원칙 아래, 마스터 폴더(드라이브), 일감 필터(Jira JQL), 추적 채널(슬랙)을 단일 뷰로 응집시켰습니다. 리소스 행 옆에는 피드백, 슬랙 대화, Jira 하위 티켓을 한 클릭으로 호출하는 사이드 패널을 두어 컨텍스트 스위칭을 최소화했습니다.

03 · BUILD

실제 문제를 해결하는 핵심 기능

- 콜라보별 통합 셋업 — 탭마다 드라이브 URL / 슬랙 채널 / Jira JQL이 프리셋으로 묶여 자동 트래킹
- 실시간 협업 동기화 — Supabase Realtime을 활용해 동료의 변경 사항을 새로고침 없이 즉시 반영
- Jira 양방향 지능형 동기화 — 5분 폴링으로 상태·담당자 등을 동기화하되, 사용자가 편집 중인 항목은 충돌 방지를 위해 자동 스킵
- Jira 계층형 다중 매칭 — Epic-Task-Subtask 구조를 인식해 하나의 리소스를 여러 담당자(그래픽·사운드 등)와 연결하고 진행률을 자동 종합
- 슬랙 AI 자동 분류 — CF Worker와 Gemini가 30분마다 대화를 분석해 관련 리소스에 매칭. 오분류는 휴지통-복원 흐름으로 간편하게 정정
- IP 홀더 피드백 패널 — 피드백 발생-반영-최종 확인 단계를 체크박스로 추적. 듀얼 모니터 워크플로우를 고려해 별도 창 분리 지원
- 통합 대시보드 — 진행률, QA 검수 현황, 마감 임박 건, 미해결 피드백을 한눈에 조망

04 · IMPACT

흩어진 정보가 하나의 흐름으로 묶이다

도구 이동에 따르는 인지 비용이 제거되어, 운영자가 화면을 벗어나지 않고도 콜라보의 전체 맥락을 파악할 수 있게 되었습니다. 슬랙에서 휘발되던 IP 홀더 피드백은 영구 기록되어 누락이 사라졌고, AI가 관련 대화를 자동으로 행 옆에 모아주어 검색 비용을 획기적으로 낮췄습니다. 포스트모템이 지정한 4대 액션 아이템(마스터시트 / 드라이브 / 히스토리 / 피드백 트래킹)을 단일 도구로 해결하여, 차기 콜라보에서 동일한 부채가 반복되지 않는 기반을 마련했습니다.

05 · LEARNINGS

도구는 데이터가 아니라 사람의 흐름을 따라야 한다

이번 작업은 “새 도구를 만든다”보다 “사람이 이미 일하는 방식 위에 데이터를 흐르게 한다”가 훨씬 중요하다는 점을 다시 확인시켜 주었습니다. 운영자는 한 콜라보를 운영할 뿐, 슬랙·드라이브·Jira·시트로 나누어 사고하지 않습니다. 그래서 도구를 더 추가하기보다 도구 사이의 다리를 놓아, 한 화면에서 컨텍스트가 자연스럽게 종합되도록 만드는 방향이 옳았습니다.

또한 AI 자동화는 **통제 가능성**이 핵심이라는 점도 체감했습니다. 모든 걸 한 번에 끝내는 일괄 처리보다, “**사용자 트리거 + 결과 캐시**” 흐름이 운영 안정성 면에서 훨씬 오래 살아남습니다. 오분류가 발생하면 사용자가 휴지통-복원으로 즉시 정정할 수 있는 안전장치를 함께 뒤야 자동화가 무너지지 않는다는 점도 함께 배웠습니다.

결국 자동화의 진정한 가치는 데이터를 한곳에 모으는 것이 아니라, **사람이 도구를 의식하지 않고 본연의 업무에 몰입하게 만드는 것**에서 완성된다는 점을 깨달았습니다.