비즈니스

사내 업무 도구를 직접 만들면서, 무엇을 만들지 않기로 했나

방동걸 (주)웨이스 대표이사
2026년 8월 26일

저희는 제조 현장의 AI를 만드는 회사인데, 정작 사내 업무는 오랫동안 폴더와 문서로 돌아갔습니다. 제안서, 사업계획서, 회의록, 회사 수치가 전부 파일이고, 그 파일들을 AI 도구로 만들고 고칩니다. 팀 규모가 크지 않아 시스템을 도입하기도 애매했습니다. 고객 공장에는 VEXPLOR를 넣는 웨이스가 사내 업무 도구를 직접 만들면서 내린 선택을 적은 글입니다.

그래서 지난 7월 사내용 도구를 하나 만들기 시작했습니다. 대단한 발상이 있어서가 아니라 하루에 수십 번 반복하는 동작이 눈에 거슬려서였습니다. 파일 탐색기에서 문서를 찾고, 미리보기로 열어 보고, 터미널로 옮겨 가 작업을 시키고, 결과를 확인하러 다시 미리보기로 돌아옵니다. 창 세 개를 계속 오갑니다.

만들자고 정하기는 쉬웠습니다. 어려웠던 쪽은 **무엇을 만들지 않을지** 정하는 일이었습니다. 사내 도구는 요구가 끝없이 붙고, 붙는 요구마다 그럴듯합니다. 일곱 주 남짓 운영하면서 지킨 원칙 세 가지와, 그 원칙 때문에 포기한 것들을 적습니다.

01

첫째, 공유 서버를 두지 않았습니다

가장 자연스러운 설계는 서버 한 대를 띄우고 팀원이 브라우저로 접속하는 방식입니다. 설치가 한 번이고 업데이트도 한 번입니다. 저희는 이 안을 접었습니다.

이유는 셋입니다. 하나, AI 도구는 각자의 계정으로 로그인해 사용하는데 그것을 서버 한 곳에 모으면 계정을 공유하는 모양이 됩니다. 둘, 업무 폴더가 클라우드 동기화로 각자 컴퓨터에 이미 복제돼 있어서, 서버가 같은 파일을 따로 건드리면 충돌 사본이 생깁니다. 셋, 서버를 띄우는 순간 그것은 도구가 아니라 서비스가 되고 관리자와 장애 대응이 따라붙습니다.

그래서 각자 자기 노트북에서, 자기 컴퓨터에 이미 동기화된 사본을 대상으로 도구를 실행합니다. 설치와 업데이트는 번거로워졌지만 위 세 가지가 전부 사라졌습니다.

대가도 있습니다. **팀원마다 도구 버전이 다를 수 있습니다.** 이건 실제로 문제를 일으켰고, 지금은 자동 업데이트와 설치 버전 표시로 관리합니다.

02

둘째, 대화 화면을 새로 만들지 않았습니다

AI 도구를 감싸는 앱을 만들 때 가장 흔한 선택은 채팅 화면을 직접 만드는 것입니다. 디자인을 마음대로 할 수 있고 기능도 붙이기 쉽습니다.

저희는 그러지 않고 **이미 사용하던 명령줄 도구를 그대로 화면 안에 태웠습니다.** 겉보기에는 투박합니다. 대신 그 도구에 이미 붙어 있던 규칙과 절차가 하나도 빠짐없이 그대로 따라옵니다.

이 차이는 나중에 커졌습니다. 채팅 화면을 자작했다면 업무 규칙을 두 곳에 구현해야 했을 것입니다. 명령줄에서 돌 때 한 번, 앱에서 돌 때 또 한 번. 두 구현은 반드시 갈라지고, 갈라진 뒤에는 어느 쪽이 맞는지 아무도 모릅니다. 같은 형태의 문제를 다른 자리에서 실제로 겪었기 때문에 이 판단은 지금도 옳았다고 봅니다.

포기한 것은 화면의 자유도입니다. 대화 이력을 예쁘게 보여 주거나 답변에 버튼을 붙이는 기능은 만들 수 없습니다. 그런 요청이 들어올 때마다 같은 답을 합니다. 그 기능의 값보다 규칙이 둘로 갈라지는 비용이 큽니다.

03

셋째, 방식은 폴더에 두고 앱에는 진입점만 두었습니다

업무하는 방식 자체는 앱 안에 넣지 않습니다. 제안서를 어떤 순서로 쓰는지, 회사 수치를 어디서 확인하는지, 문서를 어디에 저장하는지 같은 규약은 전부 업무 폴더 안의 문서로 두고, 앱은 그것을 불러오는 자리만 제공합니다.

이유는 단순합니다. **도구는 바뀌고 방식은 남기 때문입니다.** 방식을 앱 코드에 심으면 앱을 사용할 수 없게 되는 순간 방식도 함께 사라집니다. 문서로 두면 앱이 없는 사람도 그 문서를 읽고 같은 방식으로 일할 수 있습니다.

이 원칙은 실제로 쓸모가 있었습니다. 저희 팀에서 이 도구를 사용하는 일곱 명 중 여섯 명이 다른 운영체제를 사용하는데, 한동안 앱이 그쪽에서 제대로 동작하지 않는 기간이 있었습니다. 그동안에도 업무는 돌았습니다. 방식이 폴더에 있었기 때문입니다.



폴더 트리·작업 창·문서 보기를 한 화면에 모은 형태

04

그래서 만들지 않은 것들

안 만든 것	대신 한 것	이유
공유 서버	각자 노트북 실행	계정·충돌·운영 부담
자체 채팅 화면	쓰던 도구를 태움	규칙 중복 방지
앱 안의 업무 규칙	폴더 문서로 분리	도구는 바뀐다
문서 편집 기능	보기 전용	저장 충돌 책임
발표자료 화면 렌더	기본 앱에 위임	서식 재현이 끝없다

마지막 두 줄이 가장 자주 요청받는 기능입니다. 앱 안에서 문서를 고치고 싶고, 앱 안에서 발표자료를 보고 싶은 것은 당연한 요구입니다. 그래도 열지 않았습니다. 편집을 열면 저장 충돌과 서식 손상을 저희가 책임져야 하고, 발표자료를 화면에 그리려면 서식 재현이라는 끝없는 작업이 시작됩니다. 둘 다 이미 잘하는 프로그램이 있는 영역입니다.

05

도구가 실제로 해결한 문제는 창 왕복이 아니었습니다

만들 때 세운 목표는 창 왕복을 없애는 것이었습니다. 그건 초기에 달성됐고, 그 뒤에 더 값나가는 쓸모가 따로 나타났습니다.

업무 규칙이 팀원 컴퓨터까지 실제로 가 닿는 길이 됐습니다. 위험한 작업 전에 확인을 받는 규칙, 파일을 고치기 전에 사본을 떠 두는 규칙 같은 것을 문서로 적어 두는 것과 그것이 팀원 컴퓨터에서 실제로 도는 것은 전혀 다른 일입니다. 저희는 오랫동안 적어 두기만 하고 그게 도는지는 보지 않았습니다. 안 보고 있었다는 것조차 몇 주 뒤에 알았습니다.

그 실패의 자세한 경위와 거기서 만든 확인 질문은 별도 글에 적었습니다. 요약하면 규칙 파일이 동기화 필터에 걸려 팀원 컴퓨터에 아예 도착하지 않고 있었고, 아무 오류도 나지 않았습니다. 지금은 앱 설치본이 그 배달을 맡습니다.

06

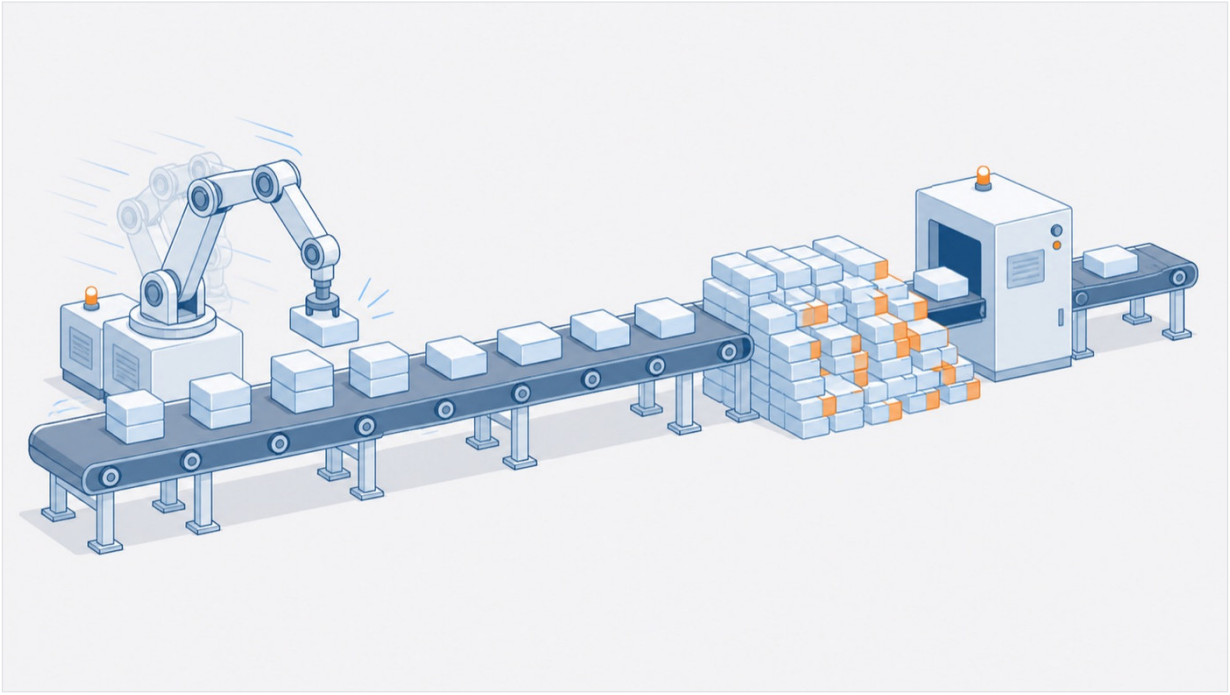
만들 시간은 어디서 났나

여기서 이런 의문이 드실 수 있습니다. 스무 명이 안 되는 회사에 사내 도구까지 만들 여력이 어디 있느냐는 것입니다.

저희가 고객에게 파는 그 방식으로 저희 도구를 만들었습니다. 이 도구의 코드는 대부분 AI 도구로 작성했고, 사람은 무엇을 만들지 정하고 결과를 검수하는 일을 합니다. 그래서 첫 기획 문서를 쓴 것이 7월 초인데 그 사이 버전이 0.2대에서 0.28대까지 올라갔습니다. 예약 작업, 사용량 측정, 문서 관계망 같은 기능이 그 안에 들어갔습니다.

이 속도에는 함정도 붙어 있습니다. **만드는 속도가 검증하는 속도를 넘어서면 안 잡힌 결함이 쌓입니다.** 저희가 겪은 사고 대부분이 기능이 없어서가 아니라 만든 것을 확인하지 않아서 났습니다. 그래서 지금은 새 기능을 넣을 때 그것을 확인할 방법을 먼저 정하고, 확인 방법이 안 서면 기능을 미룹니다.

짧은 예가 하나 있습니다. 팀의 AI 사용량을 집계하는 화면을 만들었는데, 원본 기록이 한 번의 작업을 여러 조각으로 나눠 남기는 구조였습니다. 조각을 그대로 더하면 사용량이 실제의 세 배로 나옵니다. 화면은 아주 그럴듯했고 숫자만 틀렸습니다. 만들기 전에 "이 숫자가 맞는지 무엇과 대조할 것인가"를 정해 두지 않았다면 그 세 배를 그대로 믿었을 것입니다. 사내 도구에서 가장 위험한 것은 안 되는 기능이 아니라 **틀린 숫자를 잘 보여 주는 기능**입니다.



만드는 속도가 확인하는 속도를 앞지르면 확인 안 된 것이 검사 앞에 쌓인다

07

맥에서 만들면 맥에서만 확인하게 됩니다

한 가지 사건을 그대로 적겠습니다.

어느 버전에서 팀원들이 앱을 켤 때마다 오류 대화상자를 보기 시작했습니다. 만든 사람 컴퓨터에서는 아무 증상이 없었습니다. 원인은 기능이 아니라 **자동 업데이트가 실패했을 때 그 실패를 처리하는 코드가 다시 실패하는 것**이었고, 두 운영체제가 그 자리에서 서로 다른 경로를 타기 때문에 한쪽에서만 드러났습니다.

코드를 잘못 짠 것이 문제의 전부가 아니었습니다. **만드는 사람의 컴퓨터가 팀의 다수와 다르면, 확인은 늘 소수 쪽 환경에서만 돌아갑니다.** 저희 팀은 이 도구를 사용하는 일곱 명 중 여섯 명이 만드는 사람과 다른 운영체제를 사용합니다. 즉 기본값이 뒤집혀 있었습니다.

지금은 두 운영체제 모두에서 실제로 켜 보는 확인을 릴리스 절차에 넣었습니다. 자동 테스트로 대신하지 않는 이유는, 이 사고가 실행되기 전 단계가 아니라 앱을 켜는 순간에 났기 때문입니다.

08

아직 못한 것

이 글이 성공담으로 읽히지 않게 남은 구간을 그대로 적습니다.

대외 판매는 하지 않습니다. 이 도구는 사내 도구이고, 고객에게 파는 제품군에 들어 있지 않습니다. 팔려면 먼저 정리해야 할 보안 항목이 여섯 가지 남아 있고 그 전에는 유상 계약과 고객 실데이터 투입을 하지 않기로 정해 두었습니다.

규칙의 정확도가 만점이 아닙니다. 안전 규칙이 얼마나 잘 잡는지를 서른 가지 상황으로 처음 재 봤을 때 스물한두 개를 맞혔습니다. 못 잡는 다섯 가지는 실제로 보호 대상 파일을 바꾸는 명령입니다. 고치는 중입니다. 적어도 지금은 이 숫자를 알고 있습니다. 그 전까지는 모르면서 믿고 있던 기간이 길었습니다.

모든 컴퓨터에서 확인되지 않았습니다. 최근 배포한 배달 경로가 팀원 컴퓨터에 실제로 도달했는지 확인 중이고, 아직 전원 확인이 끝나지 않았습니다.

09

사내 도구를 만들지 고민하신다면

같은 고민을 하시는 분께 저희 경험에서 건질 만한 것을 세 줄로 줄이면 이렇습니다.

첫째, **기능 목록보다 안 만들 목록을 먼저 정하십시오.** 사내 도구는 요구가 끝없이 붙고 거절할 근거를 미리 만들어 두지 않으면 반년 뒤에 유지보수가 불가능한 물건이 됩니다.

둘째, **이미 잘 도는 것을 감싸되 다시 만들지 마십시오.** 감싸면 그 도구의 개선이 공짜로 따라오고, 다시 만들면 그 도구가 개선될 때마다 따라가야 합니다.

셋째, **업무 방식은 도구 밖에 두십시오.** 도구는 몇 년 안에 바뀝니다. 방식이 문서로 남아 있으면 도구를 갈아도 일이 멈추지 않습니다.

만드는 김에 팔면 어떻겠냐는 이야기도 안에서 나왔습니다. 지금 답은 아직 아니라는 쪽입니다. 사내에서 매일 쓰이는 것과 남에게 팔 수 있는 것 사이에는 보안, 문서, 지원이라는 구간이 남아 있고 그 구간을 건너뛰면 고객이 대신 비용을 치릅니다. 저희가 제조 현장에서 파는 제품에 적용하는 기준을 사내 도구에도 똑같이 적용하기로 했습니다.

WACE(웨이스)는 공장이 스스로 판단하고 멈추지 않게 하는 제조 AI 운영체제 VEXPLOR를 만드는 회사다.

원문 blog.vexplor.com/post/what-we-chose-not-to-build
발행 2026년 8월 26일 · 방동걸 (주)웨이스 대표이사

회사 주식회사 웨이스 (WACE Inc.) · 대표이사 방동걸 · 사업자등록번호 439-87-02852
주소 (18462) 경기 화성시 동탄구 동탄영천로 150, 에이동(A동) 1702-1708호 (영천동, 실리콘앨리동탄)
연락처 전화 070-4730-0629 · contact@wace.me
홈페이지 wace.me · linkedin.com/company/wace-inc

원문 홈페이지 링크드인