2026 AI 활용 사례 공모전 · 업무 생산성 개선을 위한 AI 활용

제 앱에는 결제 없이 유료 기능을 켜는 구멍이 있었습니다 — AI에게 품질·보안·재무를 나눠 맡겨 찾아냈습니다

1인 사업자의 교차검증 구조 만들기 · 활용 도구: Claude (Cowork · Claude Code) · 감사 2026-07-14

① 어떤 상황에서 AI를 활용했나요?

제 앱에는 결제하지 않고도 유료 기능을 켤 수 있는 구멍이 있었습니다. 직원이 없는 1인 사업자라, 소상공인용 예약·고객관리 앱을 혼자 만들어 2026년 9월 4일 App Store에 출시할 때까지 그걸 검수해 줄 사람은 저밖에 없었습니다. 만든 사람이 검사하면 자기가 의심하지 않은 곳은 끝까지 안 봅니다.

그래서 제가 바꾼 것은 앱이 아니라 보고 체계입니다. 이 사례의 산출물은 코드가 아니라 문서입니다. 감사를 한 번 돌리면 담당별 감사보고서 3건과 사장님 시나리오 보고서 1건, 이를 수렴한 통합 보고서 1건이 남습니다. AI가 각 관점에서 보고서 초안을 쓰고, 등급 확정과 조치 승인은 제가 합니다. 조직에는 품질·보안·재무를 각각 다른 눈으로 보고 서로의 판단을 교차검증하는 사람이 있습니다. 1인 사업자에게 없는 건 기술이 아니라 그 구조입니다.

② 어떤 AI를 어떻게 활용했나요?

먼저 운영헌장(COMPANY.md)을 한 장 썼습니다. 프롬프트보다 규칙을 문서로 먼저 고정한 것이 핵심입니다. 실제 파일은 영문이라 우리말로 간추려 옮깁니다(원문은 첨부에 그대로 넣었습니다).

· CEO는 사람(저). 출시·가격·범위의 최종 결정은 사람이 한다.
· 담당을 넷으로 나눈다 — 품질(COO), 인프라·보안·데이터(CIO), 가격·구독 무결성(CFO), 스토어·마케팅(CMO).
· 각 담당은 자기 영역에서 P0·P1이 나오면 CEO에게 보고한다.
· 담당은 비가역적 결정을 하지 않는다 — 심사 제출, 결제, 데이터 삭제, 실제 고객 발송은 사람이 클릭한다.
· 모든 과제는 문서로 끝낸다. 심각도는 P0(블로커)·P1(주요)·P2(경미)·P3(다듬기)로 표기한다.
· 운영 데이터 보호선: 심사용 데모 계정 훼손 금지, 테스트 데이터에는 UAT_ 접두사를 붙인다, 실제 고객 번호로는 발송 금지.

이번 감사에는 헌장의 네 자리 중 앞의 셋(품질·보안·재무)이 참여했습니다. 마케팅 담당은 스토어 자료를 맡고 있어 감사에는 들어오지 않았습니다. Claude(Cowork·Claude Code)에 역할을 하나씩 지정해 같은 앱을 서로 다른 관점으로 보게 했습니다. 흔히 보는 'AI 역할극'과 다른 점은 세 가지입니다. 첫째, 문서만 읽히지 않고 운영 중인 시스템을 직접 조회하게 했습니다. 둘째, 한 담당의 지적을 다른 담당이 직접 확인해 반박하게 했습니다. 셋째, 하면 안 되는 일을 프롬프트가 아니라 문서로 먼저 못 박았습니다. 여기에 더해 "비개발자 식당 사장님" 역할을 하나 두어, 기능 목록이 아니라 하루 영업 시나리오대로 심사용 데모 계정을 써 보게 했습니다(데모 데이터가 미용실 기준이라는 한계는 보고서에 그대로 적혀 있습니다).

③ 활용 결과 어떤 변화가 있었나요?

2026년 7월 14일, 약 3시간 동안 감사를 돌렸습니다. 네 관점의 보고를 교차검증해 수렴한 결함이 9건이었습니다. P0 2건, P1 2건, P2 4건, P3 1건입니다.

가장 큰 건 P0-1이었습니다. 구독 결제 검증 자체는 정상이었는데, 서버가 검증한 결과가 사용자 권한으로 고쳐 쓸 수 있는 위치에 저장되고 있었습니다. 즉 로그인한 사람이 자기 요금제 등급을 스스로 올려 결제 없이 유료 기능을 열 수 있는 상태였습니다. 발견에서 끝내지 않고, 그 공격이 실제로 통하는 것까지 재현해 확인한 뒤 DB 계층의 쓰기 차단과 권한 회수로 막았고, 같은 시도가 차단되는 것까지 재검증했습니다. 재현은 제 소유 데모 계정 1건에 한정했고 타인 계정과 실제 고객 데이터는 대상에서 제외했습니다. 구체적인 필드명·키 종류·요청 형태는 지금도 운영 중인 서비스라 공개하지 않습니다. 두 번째 P0은 Apple 결제 키가 운영 서버에 설정되지 않아 결제가 서버 오류(500)로 실패하는 문제로, 같은 경로를 밟는 심사자도 같은 오류를 만나는 상태였습니다.

확인 결과 중에는 좋은 소식도 있었습니다. 19개 테이블 전부에 RLS(행 수준 보안 — 남의 매장 자료는 조회되지 않게 하는 DB 잠금)가 켜져 있고 정책이 모두 소유자 기준으로 걸려 있다는 것을 운영 중인 상태에서 직접 확인했습니다. 다만 같은 감사에서 FAQ 조회용 DB 함수 하나가 그 울타리를 우회할 수 있다는 지적(P2, 조건에 따라 P1)이 함께 나왔습니다. 그래서 정확히 쓰면 "테이블 단위 격리는 확인됐고, 함수 경로 하나는 후속 과제로 남았다"입니다.

④ 나만의 방식 또는 개선 포인트는 무엇인가요?

첫째, AI가 틀린 것을 오탐으로 판정하고 그 판정이 필요했던 원인을 별도 결함으로 등록했습니다. 품질 담당이 P0으로 올린 2건(계정 삭제 기능 없음, AI 응대 고장)은 오탐이었습니다. 제 컴퓨터의 코드 저장소가 운영 서버보다 옛날 버전이었고, 두 기능 모두 운영 서버에서는 멀쩡히 돌고 있었습니다. 다른 담당이 운영 상태를 직접 조회해 걸러냈고, 저장소가 운영과 어긋난다는 사실 자체를 P2 결함으로 등록했습니다. 그 뒤로 저는 AI 보고서를 받으면 코드가 아니라 운영 상태를 먼저 조회하게 합니다. 참고로 담당별 원보고서의 개별 지적은 품질 16건·보안 17건·재무 7건으로 훨씬 많고, 중복·오탐·비차단(출시를 막지 않는) 항목을 걷어낸 뒤 남은 것이 위 9건입니다.

둘째, 권한 경계를 프롬프트가 아니라 문서에 먼저 썼습니다. 헌장에 못 박은 네 가지(심사 제출, 결제, 데이터 삭제, 실제 고객 발송)를 AI가 실행한 기록은 감사 기간 내내 없습니다. 반대로 배포와 DB 마이그레이션(데이터베이스 구조 변경)은 제가 "바로 수정"을 지시한 뒤 진행됐고, 누가 언제 무엇을 했는지가 로그에 그대로 남았습니다.

· 02:35 결제검증 배포
· 02:41 보안잠금 마이그레이션 적용
· 02:48 재검증 중 트리거 버그 발견
· 02:55 수정 및 데모 계정 등급 되돌림
· 03:02 나머지 배포

다만 데모 계정 등급이 재검증 중 일시 변경됐다가 되돌려진 기록이 있습니다. 보호선이 완벽히 지켜졌다는 뜻이 아니라, 어긋난 순간이 로그에 남아 즉시 복구됐다는 뜻입니다. 경계를 문서로 정해 두면 사고가 나도 언제 무엇이 바뀌었는지가 남습니다.

셋째, 자동 기록 장치도 붙였습니다. 다만 첫 장치는 실패했습니다. 근태 기록용 서버를 만들어 2026년 7월 14일부터 8월 8일까지 609건을 쌓았는데, 그중 596건이 예약작업이 자동으로 남긴 생존 신호(heartbeat)였고 실제 작업 기록은 13건이었습니다. 가동률로는 쓸 수 있어도 생산성 지표로는 못 씁니다. 그대로 적어 둡니다.

⑤ 다른 사람도 따라 할 수 있나요?

앱을 만드는 사람만 쓸 수 있는 방법이 아닙니다. 저는 같은 방법을 개발과 전혀 상관없는 업무에 한 번 더 썼습니다. 2026년 9월 9일, 제 사업이 받을 수 있는 정부지원사업을 조사하는 데 조사용 AI 에이전트 23개와 교차검증용 AI 에이전트 1개를 붙여 보고서 20건을 만들었습니다. 검증 담당은 그 20건을 공고 원문과 대조해 확인 43항목·수정 필요 20항목·판단 불가 5항목으로 판정하고, 보고서별 정정 목록 9행을 따로 냈습니다. 대표적으로 "지금 즉시 신청, 1순위"로 올라온 서울 상표출원 지원사업은 실제로는 2026년 8월 25일 예산 소진으로 접수가 중지된 상태였습니다. 센터는 중지 사실을 공고에 정확히 게시하고 있었고, 그 공고를 확인하지 않은 것은 제 AI 보고서였습니다. 검증 담당이 없었으면 저는 그 표만 보고 헛걸음을 했을 겁니다.

정리하면 순서는 이렇습니다. (1) 헌장을 먼저 쓴다 — 특히 "AI가 하면 안 되는 일" 줄부터 쓴다. (2) 역할을 한 번에 하나씩 지정해 같은 대상을 다른 관점으로 보게 한다. (3) 심각도 등급을 강제한다 — 등급이 없으면 AI는 사소한 것과 치명적인 것을 같은 톤으로 보고한다. (4) 조사·감사 담당과 별개로 검증 담당을 반드시 하나 더 둔다. (5) 비가역 작업은 사람이 클릭한다. 특히 (4)의 검증 담당 하나가 제 경우 두 번 다 값을 했습니다.

> **측정 방법**
>
> 결함 9건은 2026년 7월 14일자 통합 UAT(사용자 인수 시험) 보고서 제4절 표에 ID·심각도·발견자·조치상태로 정리된 수렴 항목을 센 값입니다. 네 칸 서식이 적용된 것은 이 통합 표이고, 담당별 원보고서는 각자 다른 서식을 씁니다. 담당별 건수(16·17·7)는

> 각 보고서의 항목 번호를 센 값입니다. 609건은 로그 파일의 줄 수를 세고 상태(status) 값별로 분류한 값이며, 43·20·5는 검증 보고서의 판정 기호를 기계적으로 센 값이라 범례 줄과 절 제목이 포함돼 실제 판정 건수보다 두세 건 많을 수 있습니다. 첨부 PDF에 통합 보고서 제4절 표 9행 전문, 시각별 실행 로그, 검증 보고서 판정표를 원문 그대로 실었습니다.

【출처 · 윤리 · 한계】

· 본문 수치(9건, P0 2·P1 2·P2 4·P3 1, 16·17·7, 19개 테이블, 609건, 596건 대 13건, 43·20·5, 20건, 9행)는 모두 원본 보고서와 로그에서 제가 직접 확인하거나 센 값이며, 예시로 지어낸 숫자가 없습니다.

· 감사 대상이 제 제품이므로 감사자와 피감사자가 같은 사람이라는 한계를 먼저 밝힙니다. 그래서 AI가 틀린 2건과 계측 실패 1건을 지우지 않고 남겼습니다.

· 최종 9건은 품질 담당이 올린 P0 2건이 오탐으로 확인돼 빠진 뒤의 숫자입니다. 표에 남은 P0 2건은 이와 별개 항목입니다. AI 단독 정확도는 측정하지 않았습니다.

· 9건 전부를 운영 환경에서 확인한 것은 아닙니다. 운영 확인은 P0-1과 사장님 시나리오로 잡은 2건이고, 나머지는 코드를 읽어서 확인한 결과(정적 감사)입니다. 조치도 전부 끝난 것은 아니어서, 표의 조치상태 칸 기준으로 후속 과제가 남아 있습니다.

· 권한 상승 결함의 재현은 제 소유 데모 계정 1건에 한정했고 타인 계정·실제 고객 데이터는 제외했습니다. 재현 방법의 구체적인 내용은 공개하지 않습니다.

· 법적 근거: 「개인정보 보호법」 제29조(안전조치의무) — 매장 간 데이터 분리와 권한 회수는 이 조항이 요구하는 접근권한 관리에 해당합니다.

· 과학기술정보통신부 「인공지능(AI) 윤리기준」(2020.12.23.) 10대 핵심요건 중 이 사례가 직접 다룬 것은 '책임성'(비가역 결정은 사람이 하고 모든 판단을 문서로 남김)과 '안전성'(권한 상승 결함을 재현·차단·재검증)입니다.

· 사용자 수·다운로드·평점·매출은 심사위원이 검증할 수 있게 공개된 수치가 아니어서 쓰지 않았습니다. 이 글은 제품 성과가 아니라 검수 방법에 대한 기록입니다.

· 첨부한 표·도표는 원본 보고서·로그의 텍스트를 그대로 옮겨 가독성을 위해 조판한 것이며 원본 화면 캡처가 아닙니다. 내용의 추가·삭제·수정은 없고, 계정 식별자 등 식별 정보는 가렸습니다. 조판에도 Claude를 사용했습니다.

· 같은 제품을 소재로 지역사회 분야에도 응모했으나, 그 편은 개인정보 마스킹 측정 기록이고 이 편은 검수 구조에 대한 기록입니다.

· 이 글의 초안 정리에도 Claude를 사용했고, 모든 수치는 원본 보고서와 로그에 대조해 고쳤습니다.

본문에 인용한 표·로그·판정표 원문은 **02_증빙** 파일에 실려 있습니다. 표·도표는 원본 텍스트를 그대로 옮겨 조판한 것이며 원본 화면 캡처가 아니고, 계정 식별자 등 식별 정보는 가렸습니다.