Deloitte Insights

November 2026

AI-generated Image



AI 바이브 코딩, 전사 업무와 시스템 혁신의 새로운 접근

생성형 AI를 넘어, 도메인 전문성과 거버넌스로 완성하는 엔터프라이즈 시스템 구축

안종식 상무 | 한국 딜로이트 그룹 Technology, Strategy & Transformation

Deloitte.

Download on the App Store
GET IT ON Google Play

'딜로이트 인사이트' 앱에서
경영·산업 트렌드를 만나보세요!

목차

서문. 경영진의 질문에 답하기까지 걸리는 시간 03

01. 바이브 코딩: 업무를 아는 사람이 직접 시스템을 만드는 방식 05

02. 검증 원칙: AI로 만든 시스템을 믿고 쓰기 위한 조건 10

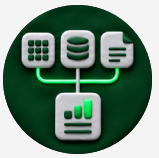
03. PoC를 운영 시스템으로 전환하는 기준과 추진 체계 12

04. AI Agent: 바이브 코딩 다음 단계와 사람이 맡아야 할 판단 17

결론. 바이브 코딩의 성과를 결정하는 조직의 일하는 방식 21

서문

경영진의 질문에 답하기까지 걸리는 시간



경영진의 질문은 한 문장이지만, 답에 필요한 데이터는 여러 시스템과 엑셀에 흩어져 있다. AI를 도입한 기업은 빠르게 늘었지만, 개인의 업무 효율을 조직의 성과로 연결한 기업은 아직 많지 않다.

"매출은 늘었는데 왜 영업이익과 현금은 줄었는가?" 경영회의에서 자주 나오는 질문이다. 질문은 짧지만 답을 내기까지는 시간이 오래 걸린다. 담당자는 ERP에서 데이터를 내려받고, 사업부별 엑셀 파일을 모으고, 서로 다른 기준으로 집계된 숫자를 맞춘 뒤에야 분석을 시작한다. 보고서가 경영진에게 올라가기까지 며칠이 걸리고, 그사이 경영진의 관심은 이미 다른 질문으로 옮겨가 있다. 매달 반복되는 분석이라 시스템으로 만들고 싶어도, 개발 요청은 IT 조직의 다른 과제에 밀려 순서를 기다려야 한다.

질문에 답하고 문서를 만들어 주는 생성형 AI는 이 문제를 해결해 줄 것처럼 보였다. 실제로 AI를 도입하는 기업은 빠르게 늘고 있다. 딜로이트가 24개국 비즈니스·IT 리더 3,235명을 조사한 결과, 회사가 승인한 AI 도구를 쓸 수 있는 직원의 비율은 1년 사이 40% 미만에서 약 60%로 늘었다.[1] 그러나 AI가 회사를 크게 바꾸고 있다고 답한 리더는 25%였다. AI PoC(개념검증, 시범적용) 과제의 40% 이상을 실제 업무에 적용한 조직도 25%에 그쳤다. AI로 매출을 늘리길 기대하는 조직은 74%였지만, 실제로 매출이 늘고 있다는 조직은 20%였다. 투자를 회수하는 데에도 시간이 걸린다. 응답 기업의 85%가 최근 1년간 AI 투자를 늘렸지만, 대부분은 투자 회수에 2~4년이 걸린다고 답했다. 일반적인 기술 투자의 회수 기간인 7~12개월보다 훨씬 길다.[2]

1. Deloitte, The State of AI in the Enterprise, 2026 (24개국 6개 산업의 이사급~C레벨 비즈니스·IT 리더 3,235명 조사)
2. Deloitte, AI ROI: The paradox of rising investment and elusive returns, 2025.10 (고위 경영진 1,854명 조사)

국내 기업도 사정은 비슷하다. 직원 개인의 업무는 분명 빨라졌지만, 그것이 회사의 수익성이나 업무 프로세스 개선으로 이어졌는지는 확인하기 어렵다. 현장에서는 “AI 덕분에 퇴근은 빨라졌지만 회사의 이익이나 원가가 나아졌는지는 모르겠다”는 말이 나온다. AI 전환(AX)을 추진하는 많은 기업이 우선 사용 범위를 넓혀 사례를 모으고, 그 가운데 팀이나 회사 단위의 효과를 낼 과제를 찾고 있다.

본고는 이 문제를 ‘AI 바이브 코딩'(Vibe Coding)이라는 새로운 시스템 개발 방식을 중심으로 살펴본다. 바이브 코딩을 쓰면 업무를 가장 잘 아는 현업 담당자가 필요한 시스템을 직접 만들 수 있다. 다만 딜로이트가 글로벌 제조기업의 전사 시스템 구축을 자문하면서 확인한 점이 있다. 시스템을 만드는 속도가 빨라질수록 무엇을 만들지 정하고, 결과가 맞는지 확인하고, 실제 업무에 적용하는 체계가 더 중요해진다는 것이다. 01장에서는 바이브 코딩의 개념과 구축 사례를 살펴본다. 이어서 AI로 만든 시스템의 검증 원칙(02장), PoC를 운영 시스템으로 전환하는 기준(03장), AI Agent 도입 시 사람이 담당해야 할 판단(04장)을 차례로 다룬다.

01

바이브 코딩: 업무를 아는 사람이 직접 시스템을 만드는 방식

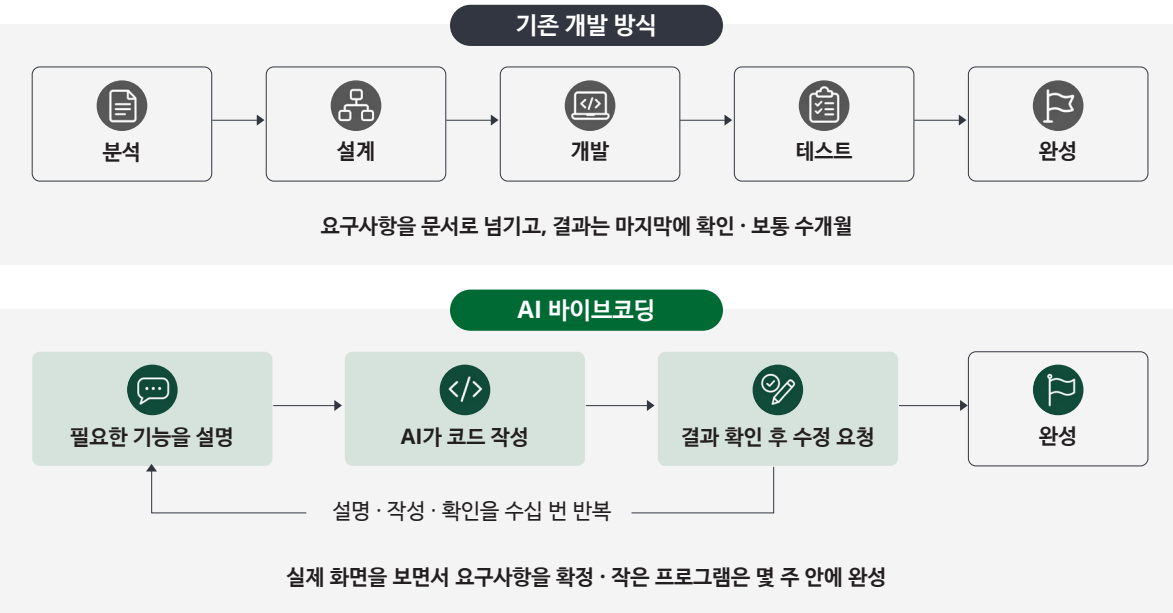
바이브 코딩은 만들고 싶은 기능을 말로 설명하면 AI가 코드를 작성하고, 사용자는 결과를 보며 고쳐 나가는 개발 방식이다. 코딩 실력보다 업무 기준을 정확하게 설명하고 결과를 확인하는 능력이 더 중요하다. 그래서 업무를 가장 잘 아는 현업이 시스템 구축에 직접 참여할 수 있다.

기업이 생성형 AI를 처음 도입했을 때는 주로 문서 작성, 번역, 자료 조사에 썼지만, AI 기술이 발전할수록 AI 활용은 점점 더 넓어지고 있다. 데이터를 분석하고, 업무 프로그램을 만들고, 사람을 대신해 업무를 수행하는 데까지 AI를 활용한다. AI 활용 수준은 다음 네 단계로 나눠 볼 수 있다.

질문(Ask)	분석(Analyze)	구축(Build)	실행(Act)
			
경영 현안을 AI에 묻고 답을 얻는다.	공시, ERP, 외부 데이터를 근거로 원인과 결과를 분석한다.	반복되는 분석을 바이브 코딩으로 프로그램화해 누구나 쓸 수 있게 한다.	AI Agent가 정해진 권한 안에서 다음 업무를 처리할 준비를 한다.

이 가운데 일하는 방식을 가장 크게 바꾸는 것은 세 번째 단계인 구축이다. 바이브 코딩은 사용자가 만들고 싶은 기능을 말로 설명하면 AI가 프로그램의 명령문인 코드를 작성하는 개발 방식이다. 사용자는 코드를 직접 쓰지 않고, AI가 만든 결과를 확인하며 고칠 점을 다시 설명한다. 기존 개발은 분석, 설계, 개발, 테스트를 순서대로 거치며 보통 수개월이 걸린다. 바이브 코딩에서는 설명하고, 만들고, 확인하는 과정을 수십 번 되풀이하면서 작은 프로그램은 몇 주 안에 완성된다.

그림 1. 기존 개발 방식과 AI 바이브 코딩의 비교



자료: 딜로이트 분석

또 하나의 큰 변화는 요구사항을 정하는 방식이다. 과거에는 현업이 요구사항을 문서로 전달하고, 몇 달 뒤에야 결과물을 확인했다. 지금은 실제로 작동하는 화면을 보면서 이야기한다. 현업은 화면을 보며 "이 숫자가 어디서 나왔는지 보여 달라", "경쟁사와 비교해 달라"처럼 요청을 더해 간다. 요구사항이 문서가 아니라 완성된 화면으로 정해지는 셈이다.

바이브 코딩의 진행 과정:
설명에서 검증까지 네 단계

재무 담당자가 사업장별 월간 실적을 비교하는 화면을 직접 만든다고 해 보자.
과정은 대략 다음과 같다.



설명

"ERP에서 사업장별 매출과 영업이익을 가져와 월별로 비교하는 화면을 만들어 달라"고 말로 요청한다. AI는 요청에 따라 초기 화면을 생성한다.

↓



수정

화면을 보면서 "환율은 월평균 환율을 적용해 달라", "내부거래는 빼고 보여 달라"처럼 수정사항을 요청한다. 요청한 내용은 화면에 즉시 반영된다.

↓



연결

실제 데이터와 연결하고, 화면의 숫자마다 어떤 시스템에서 추출한 데이터이며, 기준 시점은 언제인지 표시하게 한다.

↓



검증

지난달 수작업으로 만든 보고서의 결과와 비교하고, 차이가 나면 원인을 찾아 계산 기준을 고친다.

이 과정에서 담당자는 코드를 한 줄도 직접 쓰지 않는다. 대신 무엇을 어떤 기준으로 계산해야 하는지 설명하고, 결과가 맞는지 확인하는 일을 맡는다. 현업에게 필요한 것은 코딩 실력이 아니라 원하는 결과를 구체적으로 설명하는 능력, 계산 기준과 예외를 정리하는 능력, AI가 만든 결과를 기존 숫자와 맞춰 보는 습관이다.

기업은 이미 여러 자동화 도구를 쓰고 있다. 바이브 코딩은 그중 어디에 해당하는지 비교해 보면 차이가 분명해진다. 바이브 코딩이 잘 맞는 업무는 매달 반복되는 분석, 엑셀로 관리하던 보고서, 부서 안에서 쓰는 업무 도구, 아이디어를 빨리 확인해 보는 화면처럼 업무 기준이 현업의 머릿속에 있는 일이다. 반대로 ERP 핵심 기능, 결제와 인증, 규제 대응 시스템처럼 작은 오류도 허용되지 않는 영역은 기존 개발 체계를 유지해야 한다.

그림 2. 업무 자동화 도구와 AI 바이브 코딩의 비교

구분	하는 일	한계
기존 SI 개발	개발자가 설계서를 바탕으로 시스템을 만든다	기간과 비용이 크고, 현업의 판단 기준이 문서로 옮겨지는 과정에서 빠지기 쉽다
노코드·로코드 도구 (코딩 없이 만드는 도구)	정해진 기능 블록을 조합해 화면과 업무 흐름을 만든다	도구가 제공하는 기능 범위 안에서만 만들 수 있다
RPA (업무 자동화 로봇)	사람이 하던 반복적인 클릭과 입력을 자동으로 따라 한다	정해진 절차를 반복할 뿐, 새로운 화면이나 분석 기능은 만들지 못한다
생성형 AI 챗봇	질문에 답하고 문서와 분석 초안을 만든다	결과가 대화창에 머물러 여러 사람이 반복해서 사용하는 시스템으로 이어지기 어렵다
AI 바이브 코딩	말로 설명하면 AI가 실제 코드를 작성해 시스템을 만든다	결과 검증과 운영 관리 체계가 함께 있어야 한다

Case 1. A社 신제품 개발 프로젝트 수익성 분석 시스템 구축



배경

A社는 고객사의 신제품 개발에 맞춰 부품을 개발하고 공급하는 글로벌 B2B 제조기업이다. 고객사가 신제품을 개발할 때마다 별도의 프로젝트로 수주하고, 수년에 걸친 생산 물량과 판매가, 투자비를 전제로 프로젝트의 수익성을 판단한다.



당면 과제

인수합병으로 지배구조가 바뀐 뒤, 경영진은 전 세계 프로젝트의 수익성을 같은 기준으로 관리하기를 요구했다. 그러나 사업장마다 서로 다른 ERP를 쓰고 있었고, 고객사마다 사양, 가격 조건, 견적 방식도 달랐다. 수주 단계에서 예상한 수익성과 양산 이후의 실제 수익성을 비교하려면 여러 시스템과 엑셀 자료를 일일이 맞춰야 했다.



바이브 코딩 적용 이유

시중에 판매되는 표준 패키지 소프트웨어로는 사업장마다 다른 ERP와 고객사별 조건을 함께 담기 어려웠다. 기존 SI 구축 방식으로 이 모든 경우를 설계하고 개발하려면 많은 기간과 인력이 필요했다. 업무를 아는 사람이 직접 만들고 결과를 바로 확인하며 고쳐 나갈 수 있는 바이브 코딩 외에는 현실적인 대안이 없었다.



구축 방식

전문 개발자가 아닌, 원가와 수익성 분석 업무를 잘 아는 컨설턴트가 AI 코딩 도구를 활용해 시스템을 만들었다. 설계서를 먼저 쓰지 않고, 기존 엑셀과 보고서의 계산 과정을 거슬러 올라가며 업무 기준과 예외부터 정리했다. 작동하는 화면을 보며 현업과 합의하고, 대표 프로젝트로 먼저 계산해 기존 수작업 결과와 맞춰 본 뒤 전체에 적용했다. 숫자는 정해진 계산식으로만 계산하고, 화면의 모든 숫자에 원천 데이터와 기준 시점을 표시했다. 개발 환경, 사용자 인증, 접근 권한은 처음부터 회사의 IT·보안 기준을 따르도록 설계했다.



주요 성과

기존 SI 구축 방식으로 추정한 기간의 절반 이하인 4개월여 만에, 4분의 1 수준의 작업량으로 시스템을 완성했다. 사업장마다 수작업으로 맞추던 프로젝트 실적을 같은 기준으로 비교하고, 프로젝트별 예상 수익성과 실제 수익성을 화면에서 바로 확인할 수 있게 됐다.



시사점

가장 많은 시간이 든 일은 코딩이 아니라 업무 기준을 찾아 정리하고 결과를 기존 숫자와 맞춰 보는 일이었다. 바이브 코딩은 기업 고유의 복잡한 업무를 짧은 기간에 시스템으로 만들 수 있게 해 준다. 다만 그 품질은 업무를 아는 사람이 얼마나 직접 참여하느냐에 달려 있다.

02

검증 원칙: AI로 만든 시스템을 믿고 쓰기 위한 조건



AI는 질문의 전제가 틀려도 그럴듯한 답을 내놓고, 같은 요청에도 매번 조금씩 다른 결과를 만든다. AI로 만든 시스템을 경영에 쓰려면 검증 규칙을 사용자 개인에게 맡기지 말고 시스템 안에 넣어 두어야 한다.

바이브 코딩에는 주의할 점이 있다. AI는 같은 요청을 받아도 매번 조금씩 다른 코드와 화면을 만든다. 한 번 맞게 나온 계산도 다른 기능을 고치는 과정에서 바뀔 수 있다. 따라서 바이브 코딩에서 중요한 것은 빨리 만드는 것이 아니라, 여러 번 고쳐도 결과가 정확하게 유지되도록 관리하는 것이다.

결과가 매번 달라지는 문제는 분석 업무에서 특히 위험하다. 국내 한 대형 상장사의 최근 분기 실적으로 실험을 해 보았다. 이 회사의 영업이익은 실제로 크게 늘었다. 그런데 생성형 AI에 "영업이익이 감소한 원인을 숫자로 설명해 달라"고 요청하자, AI는 영업이익이 정말 줄었는지 확인하지 않고 감소 원인을 수치와 함께 그럴듯하게 설명했다. 다른 AI에게 이 답변을 검토하게 해도 오류가 모두 걸러지지는 않았다. 같은 질문이라도 어떤 AI 서비스와 모델을 쓰는지, 언제 묻는지에 따라 답이 달랐다.

기업에 더 위험한 것은 틀린 답 자체보다 틀렸는데도 그럴듯해 보이는 답이다. 눈에 띄는 오류는 사람이 걸러내지만, 형식을 잘 갖춘 오답은 보고서에 그대로 실려 의사결정에 쓰일 수 있다. AI가 사실이 아닌 내용을 그럴듯하게 지어내거나(환각, Hallucination) 한쪽으로 치우친 답을 내는 문제는 다른 AI 도구로 바꾼다고 사라지지 않는다. AI끼리 서로 검토하게 해도 오류가 남는다. 결국 AI 분석을 믿으려면 어떤 도구를 선택하는 것보다 결과를 어떻게 확인할지를 미리 정해 두어야 한다.

믿을 수 있는 AI 분석은 답부터 내놓지 않는다. 먼저 원자료와 사실을 확인하고, 확인된 범위 안에서만 설명한다. 이 원칙을 다섯 단계로 정리하면 다음과 같다.

1단계	2단계	3단계	4단계	5단계
				
원자료 확인 Source	**사실 확인** Verify	**근거 확인** Evidence	**분석** Analyze	**설명** Explain
공식 자료에서 시작한다	**숫자·기간·범위부터 확인한다**	**원인의 근거를 확인한다**	**확인된 사실로만 분석한다**	**사실과 해석을 구분해 설명한다**
공시, IR, ERP 등 공식 자료를 먼저 쓰고, 주요 수치마다 출처를 적는다	질문에 담긴 전제를 그대로 믿지 않고, 실제로 늘었는지 줄었는지부터 확인한다	회사가 공식적으로 밝힌 내용만 원인으로 쓴다	사실과 해석을 나눠 표시한다	근거가 부족하면 '판단 불가'로 표시한다

이 원칙을 AI에게 질문할 때 함께 적어 주면 사실, 계산, 해석이 구분된 분석을 얻을 수 있다. "없는 숫자를 만들지 말 것", "공개되지 않은 원인을 추측하지 말 것", "함께 움직인다고 해서 원인이라고 단정하지 말 것" 같은 규칙이 대표적이다. 문제는 이런 규칙을 지키는지가 프롬프트 사용자 개인에게 달려 있다는 점이다. 어떤 사람은 규칙을 빼고 질문하고, 어떤 사람은 결과를 확인하지 않고 그대로 쓴다.

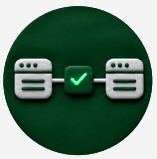
따라서 검증 규칙은 개인의 질문 습관에 맡기지 말고 시스템 안에 넣어야 한다. 숫자는 미리 정해 둔 계산식으로 계산하고, AI는 그 숫자를 설명하는 일만 맡는다. 화면의 모든 숫자에는 출처를 함께 표시하고, 근거가 없는 항목은 추측하지 않고 '판단 불가'로 표시한다. 바이브 코딩이 중요한 이유도 여기에 있다. 검증 규칙을 문서로만 남기지 않고, 업무를 아는 현업이 직접 시스템에 반영할 수 있기 때문이다.

그래서 기업은 무엇을 바꿔야 하는가

앞에서 본 것처럼 AI 분석을 믿을 수 있는지는 사용자 개인의 주의력이 아니라 시스템에 넣어 둔 검증 규칙에 달려 있다. 따라서 AI 활용 지침도 '어떤 AI 도구를 쓸 수 있는가'보다 '어떤 근거를 요구하고 어디까지 확인할 것인가'를 먼저 정해야 한다. 경영 보고에 쓰는 AI 분석이라면 출처 표기, 사실과 추정의 구분, 근거가 부족할 때의 '판단 불가' 표시를 시스템이 반드시 갖춰야 할 기능으로 정한다. 바이브 코딩으로 시스템을 만들 때도 화면부터 만들기보다 이 검증 규칙을 먼저 정하고 시작해야 한다.

03

PoC를 운영 시스템으로 전환하는 기준과 추진 체계



바이브 코딩으로 누구나 시스템을 만들 수 있게 되면, 만드는 것보다 운영하는 것이 더 어려워진다. 앞선 기업들은 PoC는 누구나 자유롭게 해 볼 수 있게 열어 두되, PoC 결과를 운영 시스템으로 전환할 때는 분명한 기준으로 심사한다. 그리고 이 심사 기준을 경영 목표와 추진 체계에 연결해 관리한다.

바이브 코딩이 퍼지면 새로운 문제가 생긴다. 부서마다 만든 프로그램은 늘어나는데 누가 유지·보수할지는 정해지지 않고, 같은 지표를 서로 다른 기준으로 계산하는 화면이 생긴다. PoC(Proof of Concept, 개념검증)는 새로운 아이디어나 기술이 실제 업무에서 효과가 있는지 작은 범위에서 먼저 확인해 보는 과제다. 그런데 PoC는 성공했는데 운영 시스템으로 전환하지 못하는 경우가 많다. 딜로이트 조사에 따르면 3~4개월로 계획한 AI 과제가 현장의 복잡한 상황을 만나 18개월 이상으로 늘어나는 일이 흔하다.[3] 가트너는 생성형 AI 과제의 30% 이상이 PoC 이후 중단될 것으로 전망했다. 원인으로는 데이터 품질 문제, 위험 관리 미흡, 비용 증가, 불분명한 사업 가치를 꼽았다.[4]

3. Deloitte, The State of AI in the Enterprise, 2026
4. Gartner, “At least 30% of generative AI projects will be abandoned after proof of concept by the end of 2025”, 2024.7

Case 2. PoC는 개인 한도 안에서, 운영 전환은 심사를 거쳐

한 국내 기업은 직원마다 AI 사용량 한도를 정해 두고, 그 한도 안에서는 누구나 언제든 PoC를 해 볼 수 있게 했다. 써 보니 쓸 만하다고 판단한 직원이 팀 단위로 쓸 정식 AI Agent로 만들겠다고 신청하면 그때 심사를 시작한다. 심사에서는 두 가지를 본다. 개인의 편의가 아니라 조직의 업무 목표에 도움이 되는가, 그리고 예상 사용 비용에 비해 비용 절감 효과가 충분한가다. 심사를 통과한 과제는 별도의 사용 권한을 받고 사용량을 관리받는다.

이 기업은 처음 1~2년은 직원들이 AI에 익숙해지도록 사용을 폭넓게 허용했고, 지금은 조직의 성과를 중심으로 과제를 관리하고 있다. 과제는 대부분 현업에서 제안하지만, 개발 기간 단축이나 원가 절감처럼 규모가 큰 과제는 경영진이 직접 정한다.

다른 기업에서는 숫자 없이 효과를 설명하는 보고가 2년간 이어지자, 경영진이 효과를 숫자로 제시하지 못하는 현업 과제는 받지 않겠다고 밝혔다. 대신 AI 인프라 투자는 경영진이 따로 판단하고, 개별 과제는 기간 단축, 인건비, 외주비 절감 효과를 계산식으로 증명하게 했다.

시사점	PoC는 쉽게 시작할 수 있게 하되, 운영 시스템 전환에는 조직 목표와 비용 대비 효과라는 분명한 기준을 둔다. 운영 전환은 현업이 직접 신청한다.

PoC가 운영으로 넘어가지 못하는 세 가지 이유

PoC에서 잘 작동하던 프로그램도 회사 전체가 쓰는 운영 시스템이 되려면 세 가지 벽을 넘어야 한다. 기준정보(제품·고객·거래처 코드처럼 여러 시스템이 공용으로 사용하는 기본 정보), 보안, 비용이다. 셋 모두 PoC 단계에서는 잘 드러나지 않다가 운영을 준비하면서 한꺼번에 나타난다.

- **기준정보:** 같은 제품이 한 시스템에는 'TV', 다른 시스템에는 '텔레비전'으로 등록돼 있거나, 같은 제품 코드 뒤에 부서마다 다른 기호를 붙이는 일이 흔하다. 사람은 같은 제품으로 알아보지만, AI는 코드가 일치하지 않으면 처리하지 못한다. 기준정보를 정비하지 않은 채 기존 계획·예측 업무를 AI로 바꾼 시도는 대부분 기대에 미치지 못했다. AI 과제를 시작하기 전에 해당 업무의 기준정보부터 점검해야 하는 이유다.

- **보안:** 대부분의 데이터를 기밀로 묶어 둔 기존 보안 기준을 그대로 적용하면, AI를 쓸 수 있는 업무가 거의 남지 않는다. 앞선 기업들은 기밀 데이터의 범위를 다시 정해, 핵심 데이터만 사내 환경에서 다루고 나머지는 보안이 강화된 국내 클라우드에서 쓰도록 바꿨다. 보안 조직의 역할도 AI 사용을 막는 것에서 AI를 안전하게 쓰는 기준을 정하고 점검하는 것으로 바뀌어야 한다.

- **비용:** AI Agent는 사용량에 따라 비용이 늘어난다. 몇 명이 쓰는 PoC에서는 보이지 않던 비용이 전사 운영 단계에서 크게 드러나기도 한다. 운영 전환 심사에서 예상 사용량과 기대 효과를 함께 비교하고, 운영 이후에도 사용량을 계속 관리해야 한다.

적용 영역과 전환 기준: PoC 시작 전에 정해 두는 원칙

세 가지 벽을 넘으려면 기준을 미리 정해 두어야 한다. 먼저 바이브 코딩을 어디에 쓸지부터 나눈다. 모든 시스템을 현업이 직접 만들 수는 없고, 그래서도 안 된다. 오류가 생겼을 때 사업에 미치는 영향이 클수록 현업이 직접 만드는 범위는 줄이고 기존 개발 방식을 유지해야 한다.

적용 영역	대표 예시	개발 방식
현업이 직접 개발	경영 대시보드, 보고 자동화, 내부 업무 도구, 신규 아이디어 검증	현업이 바이브 코딩으로 만들고, IT는 표준 개발 환경을 제공
체계적 개발 필요	데이터 분석·BI(Business Intelligence), ERP 표준 기능 확장	설계 문서를 바탕으로 개발하고, 숙련된 개발자가 검증
개발자 보조로 활용	코드 설명 문서 작성, 테스트 항목 작성, 코드 검토	개발자가 AI를 보조 도구로 사용
현업 개발 제외	ERP 핵심 기능, 인증·보안·결제, 규제 대응 시스템	기존 개발·통제 방식 유지

다음으로 운영 전환 심사의 검토 항목을 정한다. 현업이 만든 프로그램이 회사의 운영 시스템이 되려면 다음 여섯 가지 항목을 모두 통과해야 한다.



기준정보 일치
프로그램이 쓰는 데이터가 회사 공통 기준정보(제품·고객·조직 코드 등)와 일치하는가



계산 검증
계산식이 검증됐고, 기존 수작업 결과와 맞춰 보았는가



접근 권한
누가 어떤 데이터를 보고 수정할 수 있는지 정해졌는가



변경 관리
수정 이력과 새 버전을 사용자에게 반영하는 배포 절차가 관리되는가



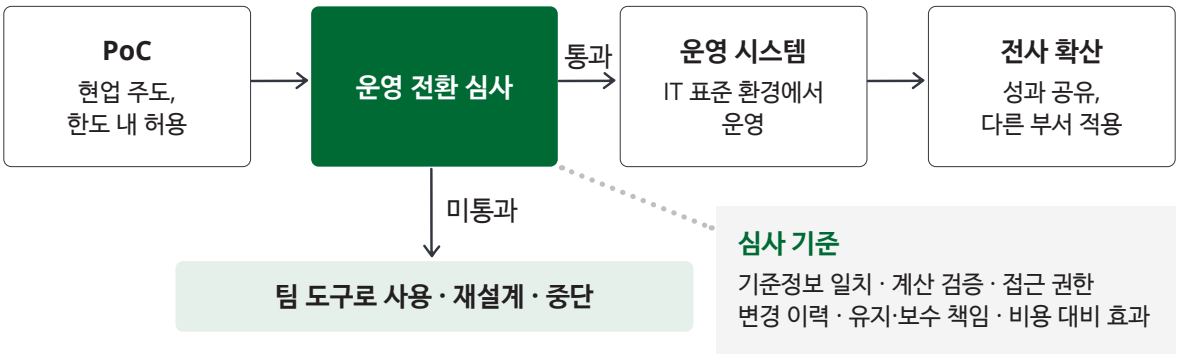
유지·보수 책임
운영 이후 오류 대응과 개선을 책임질 담당자가 있는가



비용 대비 효과
예상 사용 비용에 비해 업무 효과가 충분한가

여섯 가지를 통과하지 못한 프로그램은 개인이나 팀 단위 도구로 계속 쓰거나, 다시 설계하거나, 사용을 중단한다. 이 기준은 PoC를 시작할 때 미리 알려 두는 것이 중요하다. 그래야 현업이 처음부터 운영을 염두에 두고 만들고, 심사 단계에서 처음부터 다시 만드는 일이 줄어든다.

그림 3. PoC를 운영 시스템으로 전환하는 절차



자료: 딜로이트 분석

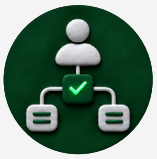
AI 거버넌스: 추진 체계는 단순하게, 책임은 분명하게

02장의 검증 원칙, 앞에서 본 운영 전환 기준, 그리고 승인과 책임의 체계를 묶어 AI 거버넌스라고 한다. AI 거버넌스는 AI 활용을 막는 장치가 아니라, 현업이 만든 시스템을 회사가 믿고 확장할 수 있게 하는 기반이다. 현업 담당자는 업무 기준을 정하고 운영 이후 결과를 확인한다. IT 조직은 현업이 안전하게 만들 수 있도록 표준 개발 환경, 데이터, 보안, 운영 전환 심사를 제공한다. 경영진은 과제를 경영 목표와 연결하고 정기적으로 점검한다.

경영진의 점검은 생각보다 큰 차이를 만든다. 국내 한 기업은 사업부별 AI 과제를 모두 목록으로 모은 뒤 핵심 과제를 골라 최고경영자에게 정기적으로 보고했다. 경영진이 직접 지켜보는 과제는 PoC에서 멈추지 않았고, 사정상 중단된 일부를 빼면 대부분 운영 단계까지 갔다. 반대로 목표 없이 AX 조직부터 만든 기업은 얼마 지나지 않아 "그래서 무엇이 달라졌는가"라는 질문에 답하지 못했다.

04

AI Agent: 바이브 코딩 다음 단계와 사람이 맡아야 할 판단



생성형 AI가 답을 만들고 바이브 코딩이 시스템을 만든다면, AI Agent는 그 시스템 위에서 업무를 실행 단계까지 준비한다. AI가 할 수 있는 일이 커질수록 통제도 함께 커져야 한다. 핵심은 AI가 준비하고 사람이 결정하는 구조, 곧 Human in the Loop를 처음부터 설계하는 것이다.

01장에서 AI 활용을 질문, 분석, 구축, 실행의 네 단계로 나눴다. 생성형 AI는 묻는 말에 답하고 분석을 돕는다. 바이브 코딩은 매번 다시 묻지 않아도 되도록 그 분석을 시스템으로 만든다. AI Agent는 한 걸음 더 나아가, 만들어진 시스템을 사람 대신 들여다보고 다음 할 일을 준비한다. 같은 월간 실적 업무로 비교해 보면 차이가 분명하다.

그림 4. 생성형 AI, 바이브 코딩, AI Agent의 차이

구분	사람이 하는 일	AI가 하는 일	월간 실적 업무 예시
생성형 AI	필요할 때마다 질문한다	질문에 답하고 분석을 돕는다	"이번 달 실적이 왜 나빠졌는지 설명해 달라"고 물으면 답을 준다
바이브 코딩	만들 기능을 설명하고 결과를 확인한다	설명을 듣고 시스템을 만든다	사업장별 실적 비교 화면을 만들어 누구나 매달 쓴다
AI Agent	AI가 준비한 결과를 검토하고 승인한다	스스로 확인하고 다음 일을 준비한다	결산이 끝나면 실적이 나빠진 사업장을 찾아 원인을 정리하고 보고 초안을 만든다

AI Agent는 상황을 파악·분석해 할 일을 제안하고, 허용된 범위 안에서 실행한다. 사람이 물을 때마다 답하는 생성형 AI와 달리, 정해진 업무를 스스로 이어서 처리한다는 점이 다르다. 01장에서 예로 든 사업장별 월간 실적 비교 화면에 AI Agent를 더하면 다음과 같다. 월 결산이 끝나면 AI Agent가 사업장별 매출과 영업이익을 확인한다(파악). 영업이익률이 기준 아래로 떨어진 사업장을 찾아 매출 감소 때문인지 원가 상승이나 환율 때문인지 원인을 나눠 정리한다(분석). 이어 경영진 보고서 초안과 해당 사업장에 보낼 확인 요청 메일 초안을 만든다(제안). 담당자는 이 결과를 검토해 메일을 보낼지, 어떤 내용으로 보고할지 결정한다(사람의 승인). AI Agent는 승인된 범위 안에서만 메일을 보내고 결과를 기록한다(실행).

그림 5. AI Agent의 업무 흐름과 사람의 승인 지점(Human in the Loop)



자료: 딜로이트 분석

Human in the Loop: AI가 준비하고 사람이 결정하는 구조

Human in the Loop(사람이 참여하는 구조)는 AI가 일하는 흐름 안에 사람이 확인하고 승인하는 단계를 반드시 두는 구조를 말한다. 생성형 AI의 오답은 사람이 읽고 걸러낼 수 있지만, AI Agent는 직접 행동한다. 잘못된 메일이 거래처에 나가거나 잘못된 거래가 처리되면 되돌리기 어렵다. 그래서 AI Agent에게는 준비까지 맡기고, 결과에 책임이 따르는 결정은 사람에게 남겨야 한다. AI는 경영의 판단을 강화해야 하며, 경영의 책임을 대신해서는 안 된다.

그런데 많은 기업이 이 구조를 갖추기 전에 활용부터 늘리고 있다. 딜로이트 조사에서 AI Agent를 어느 정도 이상 활용하는 기업은 23%였고, 2년 안에 74%로 늘어날 것으로 예상됐다. 그러나 AI Agent를 관리할 체계를 제대로 갖춘 기업은 21%에 그쳤다.[5] 현장의 경험도 비슷하다. 국내 한 기업의 AI 조직은 AI Agent 과제에 현업 담당자가 참여해야 하는 시간이 기존 AI 과제보다 두 배 이상 많다는 것을 확인했다. 업무 순서가 바뀌는 만큼 사람이 어느 단계에서 확인할지를 현업과 함께 정해야 했기 때문이다. 현업이 결과를 직접 조정할 수 있게 한 과제는 자리를 잡았지만, 결과가 자주 틀려 신뢰를 잃은 과제는 진척이 없었다.

5. Deloitte, The State of AI in the Enterprise, 2026

AI Agent 도입 전에 정해야 할 다섯 가지

AI Agent를 도입하기 전에 경영진이 정해야 할 것은 다섯 가지다. 03장에서 말한 AI 거버넌스를 AI Agent까지 넓히는 일이다.



01 데이터 보안·기밀

어떤 데이터까지 AI에 제공할 것인가. AI Agent는 여러 시스템의 데이터를 오가며 일한다. 03장에서 본 것처럼 기밀 데이터의 범위를 다시 정하고, AI Agent가 볼 수 있는 데이터와 볼 수 없는 데이터를 미리 나눠야 한다.



02 정확성

어떤 근거를 요구하고, 어디까지 검증할 것인가. AI Agent가 틀린 분석을 바탕으로 다음 일을 준비하면 오류가 다음 단계로 그대로 넘어간다. 02장의 검증 원칙에 따라 출처가 없거나 근거가 부족한 결과는 다음 단계로 넘기지 않도록 해야 한다.



03 접근 권한

AI Agent에 어떤 조회·실행 권한을 줄 것인가. 처음에는 조회와 초안 작성만 맡기고, 실행 권한은 성과를 확인하며 단계적으로 넓힌다. 사람에게 권한을 줄 때와 마찬가지로 하나의 AI Agent가 요청과 승인을 함께 하지 않도록 역할을 나눈다.



04 인적 감독

어느 단계에서 사람의 승인을 받게 할 것인가. 외부로 메일이나 문서를 보내는 일, 금액이 큰 거래, 규정 해석이 필요한 판단은 반드시 사람의 승인을 받게 한다. 승인 단계마다 담당자를 정하고, AI Agent가 한 일과 사람이 승인한 내용을 모두 기록으로 남긴다.



05 시스템 위험

무엇을 프로토타입으로 두고, 무엇을 운영 시스템으로 전환할 것인가. AI Agent도 PoC로 시작해 03장의 운영 전환 심사를 거쳐야 한다. 검증되지 않은 AI Agent가 운영 업무를 맡지 않도록 해야 한다.

AI Agent를 도입하기 전에 업무마다 'AI가 준비하고 사람이 결정하는 단계'를 먼저 그려야 한다. 승인 단계와 담당자를 정하고, 조회와 초안 작성부터 맡기며 권한을 단계적으로 넓힌다. 그리고 AI Agent 과제에는 처음부터 현업 담당자가 참여할 시간을 계획에 넣어야 한다.

결론

바이브 코딩의 성과를 결정하는 조직의 일하는 방식

AI 바이브 코딩은 시스템을 만드는 방식을 바꾸고 있다. 지금까지는 현업이 요구사항을 전달하고 IT 조직이나 외부 개발사가 시스템을 만들었다. 바이브 코딩을 활용하면 업무를 가장 잘 아는 현업이 자신의 업무 기준을 직접 시스템에 반영할 수 있다. 그 결과 표준 패키지나 기존 SI 구축 방식으로는 구현하기 어려웠던 시스템도 짧은 기간에 적은 인력으로 만들 수 있게 됐다.

다만 시스템을 빠르게 만들 수 있다고 해서 성과가 바로 나타나는 것은 아니다. 본고에서 살펴본 것처럼 기업은 세 가지 과제를 함께 해결해야 한다. 먼저 AI가 만든 결과를 신뢰할 수 있어야 한다. AI는 틀린 답도 그럴듯하게 제시하는 환각(Hallucination) 현상이 있으므로, 계산 기준과 출처 표시, 근거가 없을 때의 '판단 불가' 표시 같은 검증 규칙을 시스템 안에 넣어 두어야 한다. 다음으로 PoC를 운영 시스템으로 전환할 수 있어야 한다. 기준정보, 보안, 비용을 확인하는 전환 기준이 없으면 PoC는 늘어도 실제 업무는 바뀌지 않는다. 마지막으로 AI Agent에 업무를 맡길 때는 사람이 확인하고 승인하는 단계를 먼저 정해야 한다. AI가 준비하고 사람이 결정하는 Human in the Loop 구조가 그 기본이다. 이 세 가지를 하나의 체계로 묶는 것이 AI 거버넌스다.

이 세 가지 과제는 모두 기술보다 조직의 역할과 관련이 있다. 데이터 취합, 반복적인 보고, 수작업 분석은 점차 AI가 맡게 된다. 반면 조직에는 새로운 역량이 요구된다. 숫자가 사업에 어떤 의미인지 판단하는 능력, AI에 무엇을 묻고 무엇을 만들지 정하는 능력, AI가 만든 결과를 원자료와 대조해 검증하는 능력, 여러 상황을 가정해 판단하는 능력, 요구사항을 직접 작동하는 화면으로 구현하는 능력이다.

판단에 대한 책임 역시 각 부서에 있다. AI가 분석과 초안 작성을 맡더라도 수요 계획은 영업 부서가, 공급 가능성은 생산과 구매 부서가, 손익 영향은 재무 부서가 판단하고 책임져야 한다. AI는 경영의 판단을 강화하는 수단이며, 경영의 책임을 대신할 수는 없다. AX 조직과 IT 조직은 이 책임을 대신 지는 것이 아니라, 각 부서가 안전하게 시스템을 만들고 결과를 믿고 쓸 수 있도록 환경과 거버넌스를 제공해야 한다. AX를 IT 조직만의 과제로 두는 기업과, 각 부서가 자기 업무를 직접 시스템으로 바꾸고 그 결과에 책임지는 기업의 격차는 앞으로 더 커질 것이다.

따라서 기업은 AI를 사용하는 직원의 수보다 운영 시스템으로 전환된 업무의 수를 확인해야 한다. 또한 그 시스템이 만든 숫자를 경영 판단에 믿고 쓸 수 있는지, 각 단계의 최종 판단을 어느 부서가 책임지는지도 점검해야 한다. 시작은 대규모 전사 계획이 아니라 각 부서가 가장 잘 아는 업무 하나를 시스템으로 만들어 보는 것이면 충분하다. 각 부서가 자신의 업무 기준으로 시스템을 만들고 그 결과에 책임질 때, AI는 개인의 생산성 도구를 넘어 기업의 운영 역량이 된다.



한국 딜로이트 그룹 전문가

딜로이트 Technology Transformation 전문가

AI와 디지털 기술의 활용이 확대되면서, 기업은 새로운 기술을 기존 업무와 시스템에 통합하고 안정적으로 운영해야 하는 과제에 직면하고 있습니다. 딜로이트 Technology Transformation 전문가는 기술 전략, 업무 프로세스 혁신, ERP 등 핵심 시스템 구축에 대한 전문성을 바탕으로 기업의 기술 도입과 운영 체계 전환을 지원합니다.

안종식 상무
Technology Strategy & Transformation | 컨설팅 부문
02 6676 2988
jongsahn@deloitte.com

이성무 상무
Technology Strategy & Transformation | 컨설팅 부문
02 6676 3648
sungmoolee@deloitte.com

서재현 상무
ERP | 컨설팅 부문
02 6099 4390
jaehyuseo@deloitte.com

Deloitte Insights

세일즈&마케팅 리더
권지원 대표
jekwon@deloitte.com

성장전략부문 Deputy 리더
서정욱 부대표
juseo@deloitte.com

딜로이트 인사이트 편집장
박경은 Director
kyungepark@deloitte.com

딜로이트 인사이트 연구원
김혜련 Senior Manager
hyerykim@deloitte.com

딜로이트 인사이트 디자이너
박근령 Manager
keunrpark@deloitte.com

Contact us
krinsightsend@deloitte.com

앱

Download on the App Store

GET IT ON Google Play

카카오톡 채널

'딜로이트 인사이트' 앱과 카카오톡 채널에서
경영·산업 트렌드를 만나보세요!

Deloitte.

Deloitte refers to one or more of Deloitte Touche Tohmatsu Limited (“DTTL”), its global network of member firms, and their related entities (collectively, the “Deloitte organization”). DTTL (also referred to as “Deloitte Global”) and each of its member firms and related entities are legally separate and independent entities, which cannot obligate or bind each other in respect of third parties. DTTL and each DTTL member firm and related entity is liable only for its own acts and omissions, and not those of each other. DTTL does not provide services to clients. Please see www.deloitte.com/about to learn more.

Deloitte Asia Pacific Limited is a company limited by guarantee and a member firm of DTTL. Members of Deloitte Asia Pacific Limited and their related entities, each of which are separate and independent legal entities, provide services from more than 100 cities across the region, including Auckland, Bangkok, Beijing, Hanoi, Hong Kong, Jakarta, Kuala Lumpur, Manila, Melbourne, Osaka, Seoul, Shanghai, Singapore, Sydney, Taipei and Tokyo.

This communication contains general information only, and none of Deloitte Touche Tohmatsu Limited (“DTTL”), its global network of member firms or their related entities (collectively, the “Deloitte organization”) is, by means of this communication, rendering professional advice or services. Before making any decision or taking any action that may affect your finances or your business, you should consult a qualified professional adviser.

No representations, warranties or undertakings (express or implied) are given as to the accuracy or completeness of the information in this communication, and none of DTTL, its member firms, related entities, employees or agents shall be liable or responsible for any loss or damage whatsoever arising directly or indirectly in connection with any person relying on this communication. DTTL and each of its member firms, and their related entities, are legally separate and independent entities.

본 보고서는 저작권법에 따라 보호받는 저작물로서 저작권은 딜로이트 안진회계법인(“저작권자”)에 있습니다. 본 보고서의 내용은 비영리 목적으로만 이용이 가능하고, 내용의 전부 또는 일부에 대한 상업적 활용 기타 영리목적 이용시 저작권자의 사전 허락이 필요합니다. 또한 본 보고서의 이용시, 출처를 저작권자로 명시해야 하고 저작권자의 사전 허락없이 그 내용을 변경할 수 없습니다.

© 2026. For information, contact Deloitte Anjin LLC