

AI 시대의 혁신, 바이브코딩이 가져올 기업의 변화

저자 : 최광일 이한빈



IT 직원과 현업 직원들이 코드를 보며 협업하는 모습 (Nano Banana로 생성)

- 변화① 현업 부서의 디지털 역량 각성
- 변화② IT 개발자의 근본적 역할 진화
- 변화③ 현업-IT 협업 프로세스의 변화
- 빠른 개발속도에 따른 바이브코딩의 부작용
- 바이브코딩이 바꾸는 디지털 전환 공식

“최근 6개월간 월별 매출을 차트로 보여주고, 전월 대비 증감률을 표시하는 대시보드를 만들어줘.”

이 한 문장을 입력한 지 불과 5분 만에, 잘 작동하는 매출 분석 대시보드가 바이브코딩 앱화면에 나타났다. 이는 최근 화제가 되고 있는 ‘바이브코딩’을 단편적으로 보여주는 사례이다. 복잡한 프로그래밍 언어를 몰라도 “이런 기능이 필요해”라고 일상의 언어로 입력하면 AI가 즉시 작동하는 프로그램을 만들어준다. 기존의 AI가 지원해주는 코딩은 개발자들을 위한 보조도구로서 주로 IT 개발 부서의 생산성 향상에 쓰였다. 반면 바이브코딩은 비개발자까지 범위를 확장한 개발 방식이라 할 수 있다.



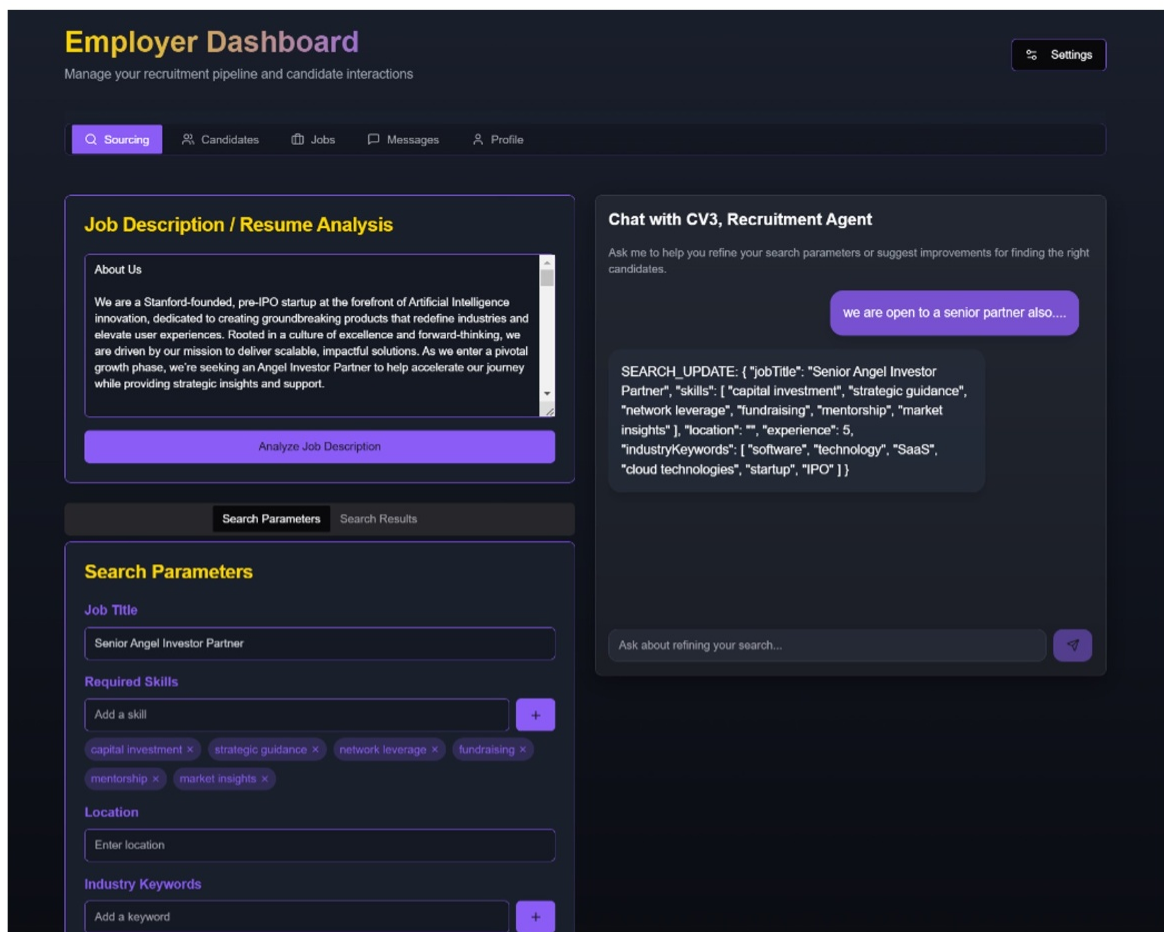
바이브코딩을 활용한 대시보드 작성

2025년 2월에 오픈AI의 공동 창업자이자 수석 AI 개발자 안드레 카파시가 “느낌만 말하면 AI가 코드를 완성한다”는 의미로 개념을 제시한 이래, 바이브코딩은 전 세계로 빠르게 확산되고 있다. ‘에어비엔비(Airbnb)’와 ‘코인베이스(Coinbase)’ 등을 발굴한 유명한 스타트업 엑셀러레이터 ‘와이콤비네이터(Y Combinator)’가 투자한 스타트업 중 25%가 전체 코드의 95% 이상을 AI로 생성하고 있을 정도이다. 이런 AI 코딩 방식의 확산은 인력 구조 변화로까지 이어지고 있다. 마이크로소프트는 2025년 AI 코딩 방식 도입을 가속화하면서 7월 한 달 동안만 9,000명의 개발자를 포함한 직원을 해고했다. 이런 추세는 더욱 가속화되어, 가트너는 2028년에 프로그램의 40%가 바이브코딩으로 개발될 것으로 전망했다. 바이브코딩은 단순한 개발 도구를 넘어 기업의 운영 방식을 근본적으로 변화시킬 것이다. 특히 제조, 금융, 유통 등 업력이 오래된 전통 대기업들이 레거시 시스템과 복잡한 조직 구조 속에서도 빠르게 혁신할 수 있는 돌파구가 될 가능성이 높다.

변화① 현업 부서의 디지털 역량 각성

바이브코딩이 만들어낸 가장 큰 변화는 코딩 지식이 없는 현업 직원들이 직접 프로그램을 개발할 수 있게 된 점이다. 2024년 BCG는 내부 컨설턴트들을 대상으로 바이브코딩의 효과를 측정하는 연구 결과를 발표했다. 코딩을 모르는 컨설턴트들에게 바이브코딩 교육을 제공한 뒤, 데이터 분석을 수행하게 했다. 데이터 처리, 통계 모델의 정확도 등을 바탕으로 성과를 측정한 결과, 코딩 경험이 전혀 없는 컨설턴트들도 전문 데이터 과학자 대비 평균 84% 수준을 기록할 정도로 효과는 상당했다.

업무에 필요한 프로그램을 직접 개발할 수 있게 된 것도 바이브코딩의 주요 이점이다. 구글 등 테크 기업에서 채용 담당자로 일했던 카일러는 바이브코딩 툴 ‘러버블(Lovable)’을 활용해 채용 어플리케이션을 개발한 사례를 공유했다. 직무 기술서와 지원자의 이력서를 비교하여 평가하는 기능을 구현했으며, 추가로 채용 요건을 입력하면 채용 플랫폼인 링크드인을 통해 적합한 후보를 찾아주는 기능도 바이브코딩으로 만들었다. 현업 실무자들은 바이브코딩을 통해 정형화된 업무들을 자동화하고, 절약된 시간을 비정형, 복잡한 업무에 집중해 업무 생산성을 향상시킬 수 있다.



바이브코딩으로 만든 채용 어플리케이션 화면

변화② IT 개발자의 근본적 역할 진화

현업 직원들이 개발 역량을 갖게 되면서, IT 개발자들의 역할도 자연스럽게 변화하고 있다. 과거의 개발자가 코드를 작성하는 실행자였다면, 바이트코딩 시대에는 설계자로 역할이 전환될 것이다. 즉, 개발자들은 ‘무엇을 만들지’ 비즈니스 상황과 연결해서 고민하고, AI가 개발자들의 요구사항을 코드로 작성하면, 개발자는 이를 검토하는 작업을 반복할 것이다. 이러한 과정을 통해 코딩 시간의 단축은 물론 프로그램의 완성도도 높일 수 있다.

AI 코딩 플랫폼 ‘깃허브(Github)’에서 발표한 연구에 따르면 AI 코딩 도구를 사용한 개발자들의 개발 속도가 55% 가량 빨라진 것으로 조사됐다. 실제 구체적인 성과도 나타나고 있다. 미국 금융회사 ‘벡가드(Vanguard)’는 내부 개발팀에 AI 도구를 도입해 코드 작성 시간을 기존 대비 27% 단축시켰다. 또한 초기 프로그램 개발부터 유지보수까지 전체 시스템 구축 과정에서 10~15%의 생산성 향상을 보였다고 밝혔다.

월마트는 여러 부서에서 다양한 AI 코딩 도구를 실험하고, 효과가 검증된 것만 전사로 확산하는 방식으로 바이트코딩을 도입했다. 그 결과 내부 개발자 업무 시간을 한 해 동안 400만 시간이나 절약했다. 이는 개발자 2,000명이 1년간 일하는 것과 같은 규모다. 이 시간을 고객 경험 개선과 새로운 비즈니스 모델 설계에 재투입하고 있다. 월마트 CTO는 이에 대해 “AI가 코드를 작성하는 동안, 우리 개발자들은 고객이 진짜 원하는 게 무엇인지 고민하는 데 더 많은 시간을 쓸 수 있게 됐다”라고 밝혔다.

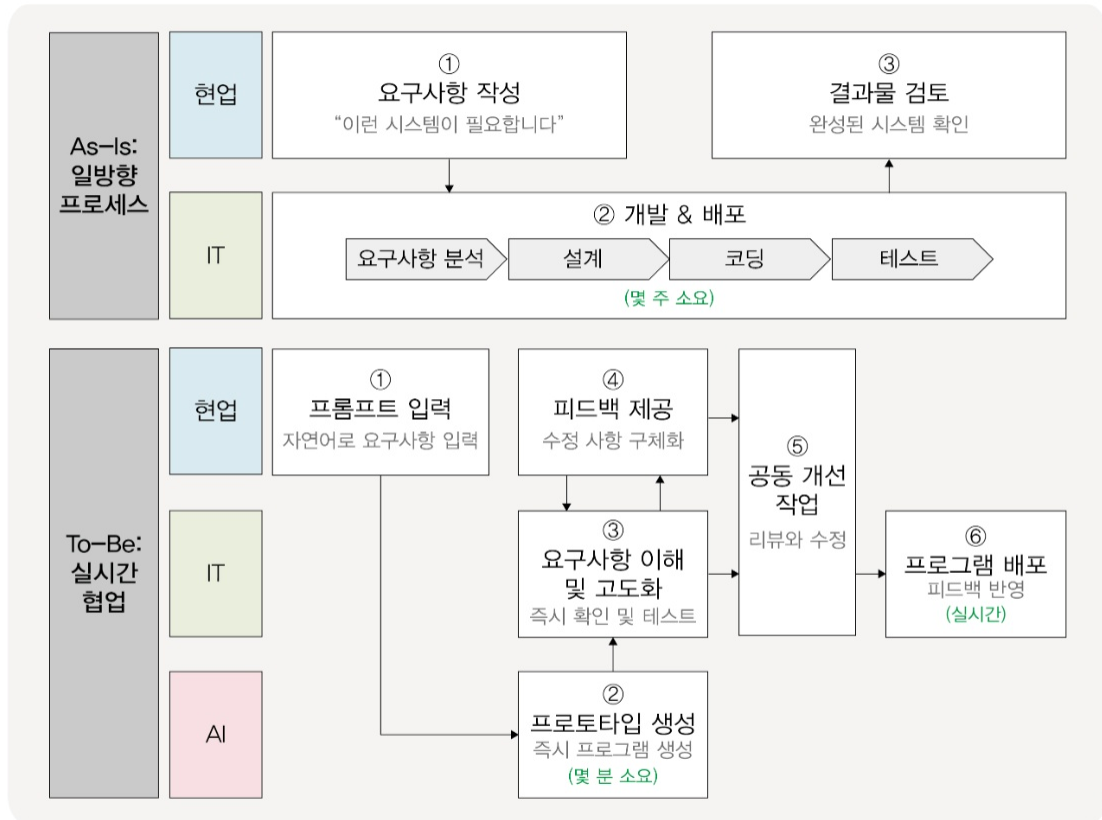
전통 대기업은 스타트업과 달리 오래된 ERP 같은 레거시 시스템, 여러 SI 벤더와의 복잡한 계약, 엄격한 보안 거버넌스 등의 제약이 있다. 바이트코딩은 새로운 개발 방식이지만 월마트 사례가 보여주듯, 작은 부서에서 먼저 시작해 검증된 방식만 확산하는 단계적 접근으로 충분히 안착 가능하다. 핵심은 속도가 아니라 체계적 준비와 점진적 확산이라 할 수 있다.

변화③ 현업-IT 협업 프로세스의 변화

현업 부서가 직접 개발 역량을 갖추고, IT 개발자의 역할이 고도화되면서 두 부서 간의 협업 방식도 근본적으로 변화할 가능성이 크다. 과거에는 현업 부서가 “이런 시스템이 필요합니다”라고 요청서를 작성하면, IT 부서가 몇 주 후 결과물을 제공하는 일방향적 프로세스였다. 하지만 이제는 현업 담당자가 직접 “최근 3개월간 고객 만족도를 지역별로 분석하는 대시보드 만들어줘”라고 자연어로 입력해도 빠른 시간 내에 작동하는 시제품이 완성되는 상황이 되었다.

이는 기존 문서 중심의 순차적 소통에서 실시간 협업으로 업무 방식을 근본적으로 변화시킬 수 있다. 현업은 더 이상 요구사항을 말이나 글로 세세히 풀어 설명할 필요가 없다. 대신 실제 작동하는 시제품으로 의도를 전달할 수 있게 된 것이다.

바이브코딩 도입 전과 후 비교



글로벌 IT 서비스 기업 ‘코그니전트(Cognizant)’가 2025년 진행한 ‘바이브코딩 워크’는 이러한 협업 변화를 극명하게 보여준다. 이 행사에는 전 세계 25만 명 이상의 직원이 참여했는데, 엔지니어뿐만 아니라 HR, 마케팅, 운영 등 비기술 부서 직원들도 포함됐다. 참가자들은 러버블, ‘커서(Cursor)’, ‘클로드코드(Cloude Code)’ 같은 바이브코딩 플랫폼을 활용해 단 며칠 만에 아이디어를 작동하는 프로그램과 시제품으로 구현했다.

결과물은 워크플로 자동화 도구부터 고객 대면 앱 컨셉까지 다양했으며, 심지어 행사를 관리하는 내부 허브조차 바이브코딩으로 제작됐다. 이 프로젝트는 ‘AI와 대화만 하면 구성원 누구나 소프트웨어를 만들 수 있다’는 목표 하에 추진됐다. 단순한 실험이 아닌, 직원 주도 업무혁신의 미래에 대한 코그니전트의 비전 제시라 할 수 있다.

이러한 협업 방식의 변화는 여러 긍정적 효과를 가져온다. 마케팅 팀이 만든 대시보드를 IT팀이 즉시 검토하고 개선하면서 개발 속도가 증가할 수 있다. “이런 보고서가 필요해요”라는 말 대신 작동하는 프로토타입으로 소통하면서 부서 간 오해가 대폭 감소할 수 있다. 현업이 코드를, IT가 현업 프로세스를 이해하게 되면서 조직 IT 기술 역량의 상향 평준화도 가능하다.

빠른 개발속도에 따른 바이브코딩의 부작용

하지만 바이브코딩의 빠른 개발 속도는 필연적으로 새로운 문제를 만들 수 있다. 이것을 ‘AI Speed Trap’이라고 하는데, 여기에는 두 가지 위험이 있다.

첫째로, 프로그램의 유지보수의 어려움이다. AI가 생성한 코드들은 여러 담당자가 개별로 만들어 체계적이지 않은 경우가 많다. 이런 경우 시스템 오류 대응, 개선을 위한 수정, 향후 확장 제한 등의 문제가 발생할 가능성이 높다.

둘째로, 프로그램의 보안 약화이다. 현업에서 쉽게 프로그램을 개발하다 보니, IT팀의 통제에서 벗어나게 된다. 이를 ‘그림자 IT’라고도 부르는데, 보안 취약점이 노출되어 해커에게 좋은 먹이감이 된다. 2021년 NYU 연구팀의 분석에 따르면, AI 코딩이 생성한 코드의 40% 이상에서 보안 취약점이 발견되었다. 이후 지속적인 개선 노력에도 불구하고 최근에도 여전히 30% 이상에서 보안에 취약한 것으로 조사됐다.

이런 위험에 대응할 수 있는 방안으로, AI를 맹신하지 않고 숙련된 개발자가 AI가 생성한 코드를 꼼꼼히 검토하는 ‘Vibe-but-Verify’ 원칙이 확산되고 있으며 여기에 더해 개발 초기부터 보안을 고려하는 ‘DevSecOps(개발, 보안, 운영)’ 개발 문화도 조성되고 있다. 또한 단순 업무의 경우는 직접 코딩이나 No-Code/Low-Code 플랫폼을 활용해 더 안전한 프로그램 개발을 하도록 유도하고 있다. 바이브코딩의 성공적 활용은 속도만을 추구하는 것이 아니다. 안전성과 지속가능성을 동시에 확보하는 균형잡힌 접근이 중요하다.

바이브코딩이 바꾸는 디지털 전환 공식

바이브코딩은 AI를 활용해 모든 구성원이 IT 역량을 갖추 수 있게 하는 ‘디지털 민주화’를 실현한다. 이는 기존에 전문 IT 시스템 도입과 프로세스 자동화에 집중되어 소수의 전유물이었던 디지털 전환을 모든 구성원의 것으로 탈바꿈시킨다. 그렇다면 이런 바이브코딩을 성공적으로 정착시키려면 어떻게 해야 할까?

첫째, 바이브코딩은 도구가 아닌 조직 사고방식의 전환이다. 바이브코딩의 본질은 특정 플랫폼 도입이 아니라, 조직 전체가 개발자처럼 문제를 논리적으로 분해하고 데이터로 검증하는 방식으로 일하게 만드는 것이다. 월마트에서 바이브코딩을 통해 시간을 절약한 것보다 중요한 건, 그 시간을 고객 경험 개선이라는 핵심 문제를 푸는 데 더 집중한 것이다. 앞으로 경쟁력은 누가 더 많은 직원을 문제 해결자로 만들었느냐에 달려 있다.

둘째, 바이브코딩 기반 업무를 수행할 수 있도록 구성원 역량을 확장시켜야 한다. 현업 직원들은 바이브코딩이라는 새로운 도구를 다룰 줄 알아야 하고, IT 인력들은 단순 ‘코드 작성자’에서 ‘설계자’로 역할을 전환해야 한다. 현업 담당자들에게는 프롬프트 설계 및 작성법, 오류 수정 방법 등을 실습하는 교육이 필요하다. IT 인력들에게는 코드 자체에 집중하던 기존 업무 방식을 탈피하여, 코드 작성은 AI에 맡기고 프로그램 설계에 집중할 수 있도록 관련 역량을 강화하도록 훈련시켜야 한다.

셋째, 확보된 역량을 활용해 작은 성공 사례를 확산시켜야 한다. 바이브코딩 역량을 갖춘 직원들을 중심으로 작은 프로젝트부터 시도해야 한다. 성공 사례가 축적되어야 노하우가 생기고 실패를 줄일 수 있다. 이를 바탕으로 더 큰 프로젝트를 진행할 기반이 마련된다.

바이브코딩은 조직 운영 철학을 변화시키고 있다. 기존의 ‘요청-제공’의 수직적 관계에서 ‘공동 창작’의 수평적 파트너십으로, ‘누가 디지털 솔루션을 만드느냐’가 아니라 ‘어떻게 조직 전체가 디지털 사고방식으로 문제를 해결하느냐’가 관건이 될 전망이다.