

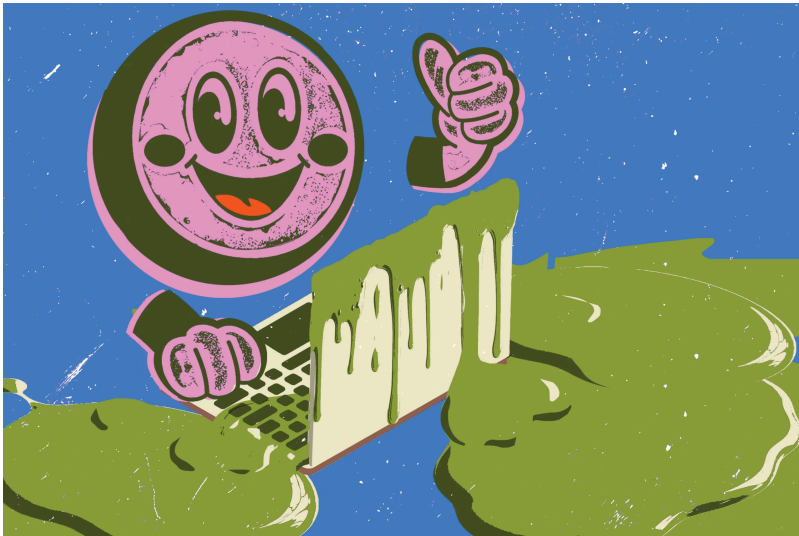
This copy is for your personal, non-commercial use only. Distribution and use of this material are governed by our Subscriber Agreement and by copyright law. For non-personal use or to order multiple copies, please contact Dow Jones Reprints at 1-800-843-0008 or visit [www.djreprints.com](http://www.djreprints.com).

<https://www.wsj.com/tech/ai/vibe-coding-slop-ai-tools-e6a99394>

TECHNOLOGY • ARTIFICIAL INTELLIGENCE • KEYWORDS: CHRISTOPHER MIMS

# The AI Superstars Who Say a 'Vibe Slop' Crisis Is Coming

A pair who helped launch the agentic-AI craze worry that their creations are pumping out bad—even dangerous—code



JOE MAGEE FOR WSJ



By Christopher Mims [Follow](#)

May 22, 2026 5:30 am ET

Two engineers who built the core of the massively popular OpenClaw AI agent have a stark warning: The artificial intelligence supposedly capable of replacing well-paid software developers is flooding the world with bad, potentially even dangerous, code.

It's a phenomenon they call "vibe slop"—a combination of "vibe coding," creating software with AI tools by describing it in plain English, and "AI slop," the endless, low-value AI-generated content all over social media.

Vibe slop happens when coders replace the hard work of designing and testing a system with the shortcut of prompting AI to whip it up, they say, and the resulting software won't stand the test of time. It's become a big enough problem that the world's main repository of open-source code—GitHub—has instituted new policies and features to combat it.

“You have infrastructure that’s falling apart, and you have software that’s now very, very buggy compared to before,” says Mario Zechner, creator of Pi, the agentic harness inside OpenClaw. “We can play this game for a couple more months, or maybe even years, but eventually it will catch up to us.”

Zechner and his partner in making Pi, Armin Ronacher, aren’t saying AI is never useful. Both use it to handle drudge work in their own projects. And they believe in it enough to have crafted an AI coding tool now used by millions.

Their core message: These systems are supposed to make senior engineers so productive that companies can lay off junior engineers, but in reality, many companies are trading near-term productivity for long-term woes. Not only does the pipeline of junior talent dry up, but residual effects include buggy software, service outages, security vulnerabilities and mounting technical debt.

## Humans out of the loop

As independent programmers with stature but no dependencies on Big Tech, the pair are sounding off in a debate now raging across the software universe—right at a time when two giants, OpenAI and Anthropic, are looking ahead to their IPOs.

Boosters say AI can revolutionize how companies handle decades of legacy code, and it has in some cases. And even developers who aren’t unleashing AI on their company’s core software report finding a happy medium, using AI for creating software tests or quick prototypes. Many in top AI labs insist that AI itself is a solution for evaluating and improving AI-generated code, without humans going over every last line.

“If you assume it will work out of the box, it probably won’t,” says Rohan Varma, who leads the team at OpenAI that builds its coding tool, Codex.

To make AI-generated code into something that can be rolled out to actual customers, engineers can now use Codex itself to check that code, he adds. Already, the system can view and test websites like a human would, check that code was written with company-specific best practices, and prod it for security issues.

Yet when it comes to critical infrastructure serving millions of people, even at OpenAI, human engineers are ultimately responsible for reviewing any code generated by AI, and for the success or failure of those systems, he says.

## The limits of AI coding

In a recent post, Alphabet Chief Executive Sundar Pichai wrote that 75% of all new code at Google is generated by AI, up from 50% last fall. A year ago, Meta CEO Mark Zuckerberg predicted that before 2026 ends, AI would be writing and reviewing most of the code from the company's internal AI development team.

It's hard to reconcile such statements with the misgivings of critics. But that just speaks to confusion about what today's AI agents can and can't do, says Zechner.

For instance, AI coding tools are better at generating new code than assessing and upgrading existing software, especially the massive reams of it that exist inside of established companies.

Startups using AI agents to vibe code their products can rapidly create new programs. But once their systems reach a certain level of size and complexity, Zechner adds, they find themselves in the same boat as larger enterprises, which have found AI agents to be of limited use.

Anthropic, with its Claude Code tool, is a case study in these tensions.



Armin Ronacher, left, and Mario Zechner, whose software contributions led to the AI agent craze, worry about 'vibe slop.'

ARMIN RONACHER; MARIO ZECHNER

Zechner lauds the company for “dogfooding” its own software—that is, using it to build its own internal systems. But he doesn't like the tool.

“Claude Code is one of the most broken pieces of software I've ever used in my entire life,” he says, citing flickering on-screen graphics, feature creep and a prodigious

appetite for memory. Zechner blames these issues on its developers' use of AI to build it.

Catherine Wu, head of product for Claude Code at Anthropic, says the visual flicker was a side effect of a software team moving at a rapid pace, and prioritizing the rollout of new features. In the past year, the median user of Claude Code went from using it 20 minutes a day to 20 hours a week. The flicker has largely been fixed, she adds.

AI tools enable junior engineers to take more responsibility for the features they develop, from concept through delivery to customers, says Wu. While Anthropic is constantly updating the instructions it gives AI to help it avoid its past mistakes, she adds, "The human still has the end responsibility."

Anthropic has some of the best AI engineers on the planet, so an AI-heavy approach can work for them, but it might not work for all of the company's customers, says Timothy B. Lee, a computer scientist who writes the newsletter Understanding AI. When dealing with their in-house software systems, many firms depend on the tacit knowledge staff programmers have accumulated over years—none of which would be in training data for AI agents.

"These models can very easily go the wrong direction, and somebody has to notice that," says Lee.

## **When the bill comes due**

Zechner believes a reckoning is coming. He thinks big companies will soon realize that their overemphasis on AI-produced code is driving up costs and leading to subpar software. He thinks many smaller startups that depend on vibe coding will fold. He also thinks cloud-based repositories of useful software tools, like GitHub, will continue to fill up with AI-generated coding garbage.

Just before Zechner and Ronacher dialed into our video interview from their home offices in Austria, Zechner had to ban a (human) programmer from contributing to one of his code repositories on GitHub. Without the programmer's knowledge, his AI agent was filing one bogus error report after another.

Zechner, shaking his head, said, "I mean, what are we even doing?"

Christopher Mims is a columnist who writes about technology for The Wall Street Journal's tech bureau in San Francisco. The subjects of his columns vary widely from one week to the next. He has written about bidets, brain implants, the cult of the founder, the history of technology, innovation, venture capital, robotics, batteries, energy,...

