

# Vibe Coding: The Wild West of Programming

1. november 2025 | 12 min

---

## What is Vibe Coding?

Vibe coding is when you describe what you want in natural language to an AI, it generates the code for you, and you can build a website or an app.

"Forget that the code even exists," as Andrej Karpathy says.

The crucial twist is that vibe coding isn't just about using AI as a helper. Simon Willison explains the difference precisely:

"If an LLM wrote every line of your code, but you reviewed, tested, and understood all of it, that's not vibe coding, that's just using AI as a typing assistant."

In short, there's a difference between vibe coding-where the AI writes, you run it, test it, and ship it-and AI-assisted development, where the AI suggests code, you understand it, you edit it, and only then do you ship it.

## The Origin: Andrej Karpathy's Tweet

February 2025. Andrej Karpathy, OpenAI co-founder and former head at Tesla, tweets:

"There's a new kind of coding I call 'vibe coding', where you fully give in to the vibes, embrace exponentials, and forget that the code even exists."

### He continues:

"I just see stuff, say stuff, run stuff, and copy paste stuff, and it mostly works... I 'Accept All' always, I don't read the diffs anymore."

As a test case, he built MenuGen, an app that takes photos of restaurant menus and generates images of all the dishes. He didn't know the difference between "Tagine," "Cavatappi," or "Sweetbread." He let the AI build everything...

A weekend later, the app was done.

Merriam-Webster added "vibe coding" as slang and a trending term the very next month. The rest of the tech world quickly followed.

## Who Are the Most Influential Vibe Coders?

### Andrej Karpathy, the Godfather

Stanford Ph.D., formerly OpenAI, Tesla's Director of AI. The man who already in 2023 said: "The hottest new programming language is English."

His philosophy is pragmatic: "Vibe coding full web apps today is kind of messy and not a good idea for anything of actual importance. But there are clear hints of greatness."

### Simon Willison, the Critical Voice

AI researcher and open-source experimenter who has built dozens of vibe-coding projects while simultaneously warning against using it in production.

He proposes the term "vibe engineering" as a more professional version of the phenomenon-where you still experiment but take responsibility for code quality and security.

### Kevin Roose, the Non-Coder Evangelist

New York Times journalist without a technical background who used vibe coding to build "LunchBox Buddy," an app that analyzes your fridge and suggests lunch.

He calls it "software for one": small tools built by non-developers for their own needs. One experiment did go wrong, though, and an AI started generating fake product reviews on a test shop. That's how it is with anything new.

## Commercial Pioneers

Y Combinator reported in March 2025 that a quarter of their Winter 2025 startups had codebases that were over 95% AI-generated.

Sweden's Lovable.dev reached around 100 million dollars in annual revenue in just eight months-perhaps one of the fastest-growing software startups ever. Replit.com went from 10 to 100 million dollars in revenue in nine months after launching their AI Agent. Maybe there's real money behind this trend too?

## The Most Well-Known Vibe Coding Tools (as of October 2025)

## **Lovable.dev (~\$25/mo)**

Builds full web apps from a single prompt. Integrates with Supabase for backend, database, and auth. Loved by founders without a technical background. Fastest path to a prototype, but also known for security issues if you're not attentive.

## **Replit.com (from \$25/mo)**

Browser-based all-in-one platform. Zero setup, collaboration-friendly, and mobile-friendly. Suitable for both hobby projects and learning. A bit pricey if you hack through the night-there's an extra meter running.

## **GitHub Copilot Pro (\$10/mo)**

The AI assistant that lives in your editor and writes code, documentation, and tests for you. It's strong because it understands the context in your repository-but you must be able to tell when it's wrong. Copilot is less of a "vibe-builder" and more a turbo for those who already think like developers.

## **Cursor.com (\$20/mo)**

An AI-native IDE based on VS Code. Preferred by experienced developers who want control. Deep repo context, pull requests, and precise code understanding. Requires technical experience. Less useful for a "non-coder" like me.

## **Windsurf.com (\$15/mo)**

A lighter variant with professional features at a lower price. Good balance between control and convenience.

## **Bolt.new (Free-\$100/mo)**

Lightweight platform for building Next.js apps directly in the browser. A favorite among beginners.

## **v0.app (Vercel)**

Focuses on beautiful React components from text prompts. And you can combine inspiration from it with some of the other vibe-coding tools to get even better front ends.

## **Claude Code**

Claude's terminal-based development tool with strengths in understanding and modifying existing codebases. Includes the ability to set up AI agents and orchestrate sub-agents. The newest feature is "skills," which you can define.

## **The Culture Behind Vibe Coding**

There's a new energy in the developer community. Not just technological, but cultural. Vibe coding represents a break with the classic idea that development is about full understanding of every code element.

For the first time, it's possible to build software without actually "coding"-at least not in the traditional sense. It's reminiscent of the early Web 2.0 years, when "move fast and break things" was the mantra.

But this time you're not just breaking interfaces, you're breaking the discipline itself.

Where development used to be a craft, it is increasingly orchestration. The role shifts from "code-smith" to "AI orchestrator." You define the vision; the AI does the work.

Karpathy compares it to a new era in creative production: "We're moving from craftsmanship to concept design. You no longer need to understand the engine to drive the car."

## Why It's Taking Off Now

### Three factors have made vibe coding possible:

- Agentic AIs: AI systems like Claude, GPT-5, Mistral, and Gemini can remember context, navigate projects, and fix their own errors. I've seen it run software tests "live" in my terminal. A pretty trippy experience.

The combination means that a person with an idea and a text description can go from thought to running web app in under an hour.

## The Professional Controversy

The question that splits the industry isn't whether AI can write code-it's whether it should.

For some developers, vibe coding is a creative free space where innovation happens faster than ever. For others, it's a flattening of the software craft and a threat to code quality, security, and comprehensibility.

Simon Willison sums up the conflict: "Most of the work we do as engineers is evolving existing systems. If you don't understand your code, you can't evolve it."

Vibe coding is therefore both a promise and a warning. It shows how far AI has come in 2025, but also how quickly control can slip out of our hands when speed and "flow" are prioritized over understanding, ethics, and security.

## What It Means for You

If you work in software development, vibe coding is a more-than-obvious sign of how your role will change.

For developers, it means you need to master prompting, context engineering, reviewing, and debugging AI-generated code-not necessarily writing everything yourself.

For leaders, it means software production is accelerating, but governance, security, and compliance risk being left behind on the platform. When code is generated faster than it can be reviewed, the risk shifts from the technician to the infrastructure-and to the entire organization.

## A New Paradigm

Vibe coding is an experiment. The pioneer spirit of the digital age. Fast, imperfect, and full of potential. We like it that way.

As Karpathy said: "The wild west of programming." And as with any pioneer movement, both gold rush and lawlessness follow.

But one thing is certain: vibe coding is here to stay. It will mature and find its place-perhaps as "vibe software engineering," perhaps as something entirely different.

The first rule of the new era is simple: vibe all you want, but you're still responsible for what you build.

---

Part 2: "The Hangover. When AI codes faster than we can think." is now available.

---

Har du spørgsmål om vibe coding, eller vil du høre, hvordan det kunne se ud i din organisation? Kontakt mig - jeg hjælper gerne med at komme i gang.

---