

Software engineering with GenAI in 2025

What works, and what doesn't?

Hi, I'm Gergely



The Pragmatic Engineer

Big Tech and startups, from the inside.



The Software Engineer's Guidebook



Gergely Orosz



Navigating senior, tech lead, and staff
engineer positions at tech companies
and startups

 The Pragmatic Engineer

Uber

 Skyscanner

 Microsoft

 skype™

 XBOX ONE

J.P.Morgan

Today's AI headlines are all over the place

On one hand:

Tech Industry > Artificial Intelligence

Microsoft's CEO reveals that AI writes up to 30% of its code — some projects may have all of its code written by AI

News

By [Hassam Nasir](#) published April 30, 2025

That's almost one-third.



💬 [Comments \(15\)](#)



YOUR NEXT MOVE

Anthropic's CEO Said All Code Will Be AI-Generated In a Year It's time to stop taking tech zealots seriously.

EXPERT OPINION BY JOE PROCOPIO, FOUNDER, JOEPROCOPIO.COM @JPROCO

MAR 20, 2025

SHARE 



AI

Google chief scientist predicts AI could perform at the level of a junior coder in a year

By [Sarah Perkel](#)



Thomas Samson/Getty Images



But then, we also have...



Anurag Bhagsain ✓

@abhagsain



Last week, we asked Devin to make a change. It added an event on the banner component mount, which caused 6.6M @posthog events in one week, which will cost us \$733

Devin cost \$500 + \$733 = \$1273 🤔👍

Lesson - Review AI-generated code multiple times



Cognition

9:17 PM · Jan 6, 2025 · **456.9K** Views

129

330

3.1K

617





r/ExperiencedDevs · 26 days ago
NegativeWeb1 🏆 Top 1% Poster



My new hobby: watching AI slowly drive Microsoft employees insane

Jokes aside, GitHub/Microsoft recently announced the public preview for their [GitHub Copilot agent](#).

The agent has recently been deployed to open PRs on the .NET runtime repo and it's...not great. It's not my best trait, but I can't help enjoying some good schadenfreude. Here are some examples:

- <https://github.com/dotnet/runtime/pull/115762>
- <https://github.com/dotnet/runtime/pull/115743>
- <https://github.com/dotnet/runtime/pull/115733>
- <https://github.com/dotnet/runtime/pull/115732>

I actually feel bad for the employees being assigned to review these PRs. But, if this is the future of our field, I think I want off the ride.

EDIT:

This blew up. I've found everyone's replies to be hilarious. I did want to double down on the "feeling bad for the employees" part. There is probably a big mandate from above to use Copilot everywhere and the devs are

[Read more](#) ▾





Copilot AI commented [last month](#) · edited ▾

Contributor Author ...

Your added tests are failing.

Fixed the failing tests by properly handling the match state when backtracking is exhausted. The fix now sets `runmatch!.FoundMatch = false` before stopping to ensure consistent behavior. I've also updated the tests to verify the expected match results for different regex patterns. [5ab0b08](#)

Warning

► Firewall rules blocked me from connecting to one or more addresses

Help improve Copilot by leaving feedback using the 👍 or 🗑️ buttons



Copilot finished work on behalf of [stephentoub](#) [last month](#)

[build-analysis](#) bot mentioned this pull request [last month](#)

[Intermittent build failure in AfterSourceBuild: "Could not write state file" #76488](#)

[Open](#)



JabSYSEmb commented [3 weeks ago](#) ...

[@copilot](#) please remove all tests and start again writing fresh tests.

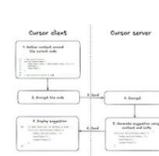


How AI is changing software engineering has also been the top of my mind:

Real-world engineering challenges: building Cursor

Cursor has grown 100x in load in just a year, sees 1M+ QPS for its data layer, and serves billions of code completions,...

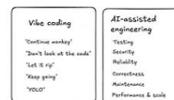
JUN 10 · GERGELY OROSZ



Vibe Coding as a software engineer

There's a lot of talk about "vibe coding", but is it just a vague term for prototyping, or could vibes change how we build...

JUN 3 · GERGELY OROSZ AND ELIN NILSSON



Microsoft is dogfooding AI dev tools' future

Impressions from a week in Seattle, at Microsoft's annual developer conference. Microsoft is eating its own dogfood...

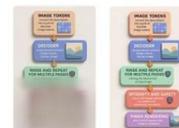
MAY 27 · GERGELY OROSZ



Building, launching, and scaling ChatGPT Images

ChatGPT Images is OpenAI's biggest launch yet, with 100 million NEW users generating 700 million images in the first...

MAY 13 · GERGELY OROSZ



But what is REALLY happening?

I talk to a lot of software engineers...

... so I asked them:

“How are you using AI tools at your company?”

I asked this from:

1. AI dev tools startups
2. Big Tech
3. AI startups
4. Independent software engineers

Here is what they told me:

1. **AI dev tools startups**
2. Big Tech
3. AI startups
4. Independent software engineers

Anthropic

"When we gave **Claude Code** access to our engineers, they all started using it every day, which was pretty surprising"

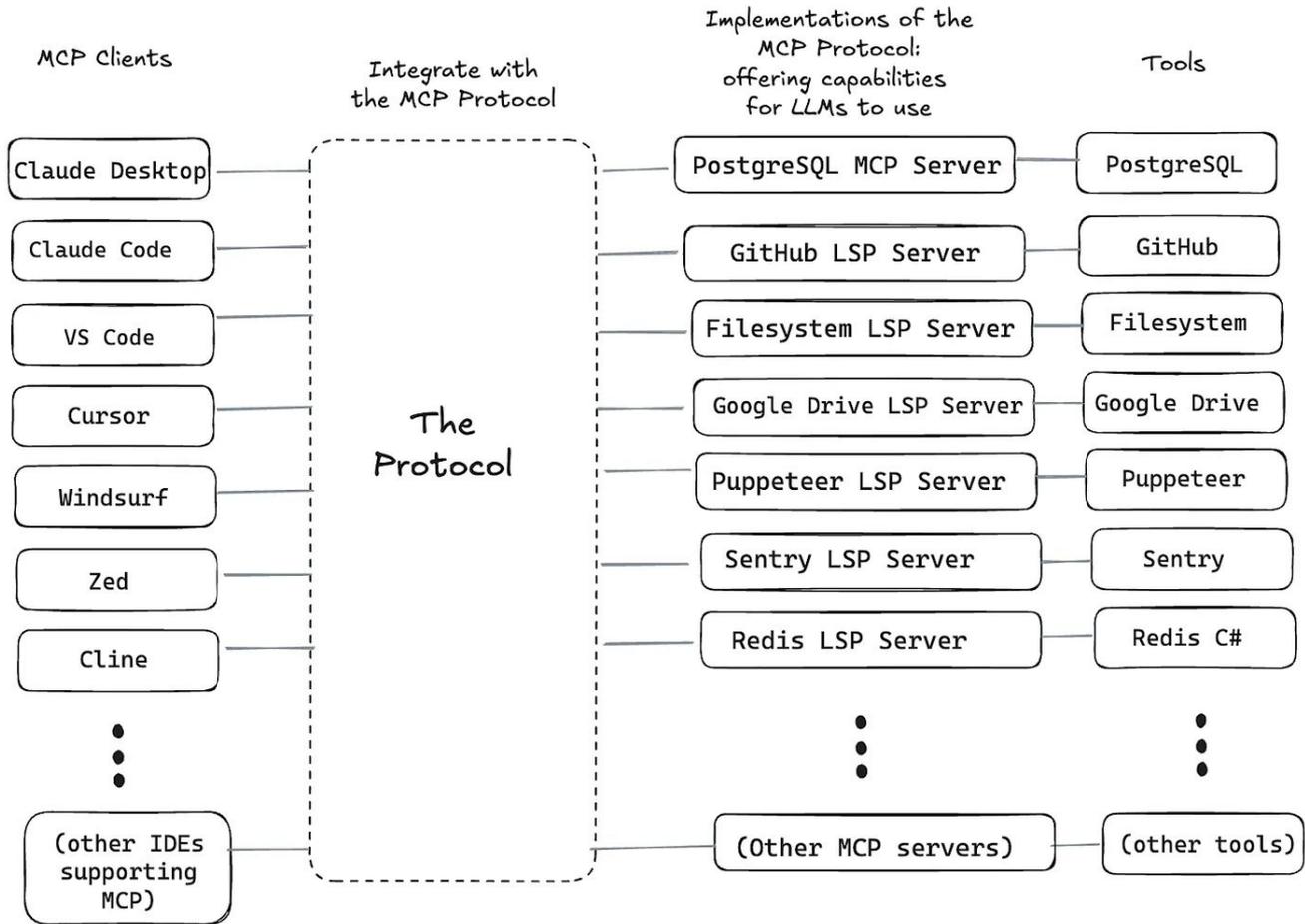
“**90%** of Claude Code is now written with Claude Code.”

Usage keeps going up:

- Claude Code usage increased **40%** within first day of launching Claude 4
- User base grew **160%** since Claude 4 launch

MCP

(Model Context Protocol)



- **Nov** 2024: Anthropic open-sources MCP
- **Dec-Feb**: Block, Replit, Windsurf, Sourcegraph
- **March, Apr**: OpenAI, Google, Microsoft
- **Today**: Thousands of active MCP servers

Windsurf

“About **95%** of our code is written using Cascade [agent] and Windsurf Tab [passive tabbing].”

Cursor

“About **40-50%** of our code is written by
Cursor.”

1. AI dev tools startups
2. **Big Tech**
3. AI startups
4. Independent software engineers

Google

Everything is custom at Google, including their IDE

- **Cider (IDE):**

- Used to be a web-based tool
- Today there's a VS Code fork version (and the web version is called Cider-V)
- Integrated with the Google internal stack

AI is integrated into all parts of their stack:

- **Cider (IDE) + Cider-V (web-based IDE):** autocomplete, chat with IDE
- **Critique:** AI code review input
- **Code Search:** chat with an LLM about the codebase

Lots of recent progress from a year ago:

“One year ago, usage was disappointing low vs what I expected. Even for simple things like generating good documentation, it wasn’t used much.”

– former software engineer at Google

Google's approach with AI for development

Software engineer at Google:

“Generally, Google is taking a cautious approach with coding tools. They want to get it right the first time **so that SWE's can trust it.**”

Lots of other AI tools:

- **NotebookLM**: feed in all your product req docs or user experience researcher docs so you can ask questions
- **LLM prompt playground**: works very similarly to OpenAI's dev playground (but predates it)
- **MOMA search engine**: knowledge base using LLMs

Many more internal LLM tools being built

Software engineer at Google:

“There’s org/team specific GenAI tooling happening everywhere because that’s what leadership likes to see these days. Cynically, that’s partially how you get more funding these days.”

Preparing for the future to come:

Former Google SRE engineer:

“What I’m hearing from SRE friends is that they are preparing for 10x the lines of code making their way into production.”

Amazon

- Amazon Q Developer Pro: devs use it a lot
- Works very well for AWS-related coding
- ... and pretty well for coding, in general

Claude (hosted internally):

- For PR/FAQ writing help
- For perf feedback writing help
- ... generally, for lots of writing tasks

MCP servers

Amazon's "Big Mandate" has always been
APIs

Around 2002 (as recalled by Steve Yegge):

“[Jeff Bezos’] Big Mandate went something along these lines:

1. All teams will henceforth expose their data and functionality through service interfaces.
2. Teams must communicate with each other through these interfaces.
3. There will be no other form of interprocess communication allowed”

If there's an API, it's easy enough to add an MCP server...

... and Amazon is doing just this

- Most internal tools and websites already have MCP support
- Automation can be hooked up to e.g. the ticketing system, emails, internal systems offering an API, pretty much everything.
- Devs really like it!

Amazon:

API-first since 2022

MCP-first starting in 2025?

1. AI dev tools startups
2. Big Tech
3. **AI startups**
4. Independent software engineers

incident.io

Move *fast* when you break things

The all-in-one AI platform for on-call, incident response,
and status pages—built for fast-moving teams.

[Get a demo](#)

[Start a free trial](#)

Lawrence Jones, software engineer at incident.io:

“Our team are massively into using AI to accelerate them.”

“We share tips and tricks in the Slack channels. Let me share a few:”



Leo Sjöberg Yesterday at 9:45 PM

I've just used the Linear MCP for the first time and it works really well for small well-defined tickets. My entire prompt was *Can you implement Linear issue ONC-6569?*

It's an extremely well-defined ticket that had loads of context, and just being able to write a ticket number is one less bit of copy-pasting over to get all the context to Claude!

I might have a go at letting this run through some XS tickets in our PR board tomorrow.



10 replies



Linear APP Yesterday at 9:45 PM

[ONC-6569 Register our mobile app receivers e.g. PushNotificationActionReceiver not to be exported](#)

State `Todo` Priority `High` Project `ONC Product Responder`

On-call | May 15

Subscribe

Todo





Rory B 2:06 PM

I've been really enjoying using Claude recently when I have an idea about something, but don't want to spend time reading into it or doing it. I think it's really good now with web search too, where so long as you prompt it to do so, you can ask for best practices around things. E.g:

- The **biome migration** on the frontend, I mostly just opened a git working tree and asked claude to look up how to migrate eslint/prettier to biome with minimal changes, come up with a plan, ensure it covers local development and CI/CD etc.
- Improving **remote updates** on mobile, where right now we lazily install them next clean launch, I ask it to look up expo docs and blog posts and find what strategies people use to install immediately without interrupting the user. Rather than just reading about it myself online, I get results about stuff online, contextual suggestions about how that would integrate with our codebase etc



6 replies Last reply 25 days ago



milhouse 11:20 AM

My new favourite trick when prompting in the last week is to ask for options, as claude usually suggests stuff I hadn't thought of and end up preferring e.g.

- Give me a few options for writing code that does X
- Why do you think I'm seeing this error? Suggest a few explanations
- How could I change this prompt so the output matches this behaviour? Give me options

NICE 2



Lawrence Jones:

“The biggest change has been from Claude Code, though. I checked just yesterday and our entire team are regular users.”

A biotech AI startup

Using AI and ML models to design proteins

Lots of automated numerical / ML pipelines

“We've experimented with several options with LLMs but little has really stuck.

It's still faster to just write correct code than to review LLM code and fix its problems, even using the latest models.

Given the hype around LLMs I speculate that we might just be in a weird niche.”

1. AI dev tools startups
2. Big Tech
3. AI startups
- 4. Independent software engineers**

Armin Ronacher

Formerly: Creator of Flask (Python), founding
engineer at Sentry

~17 YOЕ

Recently got very excited about using AI for development



AI Changes Everything

written on Wednesday, June 4, 2025

At the moment I'm [working on a new project](#). Even over the last two months, the way I do this has changed profoundly. Where I used to spend most of my time in Cursor, I now mostly use [Claude Code](#), almost entirely hands-off.

Do I program any faster? Not really. But it feels like I've gained 30% more time in my day because the machine is doing the work. I alternate between giving it instructions, reading a book, and reviewing the changes. **If you would have told me even just six months ago that I'd prefer being an engineering lead to a virtual programmer intern over hitting the keys myself, I would not have believed it.** I can go can make a coffee,

“Armin, what changed?”

- “Claude Code got shockingly good”
- “By using LLMs extensively I got over the ‘hurdle’ of not accepting it”
- “The faults of the model are avoided because now the tool can run the code and see what happens”

Peter Steinberger

Formerly: Creator of PSPDFKit, iOS Engineer

~17 YOЕ

Sold his startup and was tinkering on the side for a while.

Then...

The Spark Returns

Creating things out of ideas, building was always the thing that gave me the most joy in life. One day, things just clicked. I had a new idea, I sat on my computer and started hacking, and I realized that my spark is back.

Right now, we are at an incredible crossroads in technology. AI is moving so fast and is changing the way we work in software development, but furthermore, it's going to change the world. I haven't been as excited, astounded, and amazed by any technology in a very long time.

We are so back. It's time to build.

“Pete, what changed?”

- “We’re at a tools inflection point where suddenly it ‘just works’ ”
- “Languages and frameworks matter less because it’s incredibly simple to switch”
- “A capable engineer can create 10-20x the output than before”



Peter Steinberger  @steipete · Jun 12



All my tech friends are completely in awe with AI and have trouble going to sleep, it's such a mind blowing technology.



Mo 
@moughxyz



has brought out many burnt out devs out of retirement. everyone i know is building something

11:52 PM · Jun 13, 2025 · **3,971** Views



Birgitta Böckeler

Distinguished Engineer at Thoughtworks

~20 YOE

Birgitta Bökeler



Generative AI and particularly LLMs (Large Language Models) have exploded into the public consciousness. Like many software developers I am intrigued by the possibilities, but unsure what exactly it will mean for our profession in the long run. I have now taken on a role in Thoughtworks to coordinate our work on how this technology will affect software delivery practices. I'm posting various memos here to describe what my colleagues and I are learning and thinking.

[Autonomous coding agents: A Codex example](#) (04 June 2025)

[The role of developer skills in agentic coding](#) (25 March 2025)

[What role does LLM reasoning play for software tasks?](#) (18 February 2025)

[Expanding the solution size with multi-file editing](#) (19 November 2024)

[Building an AI agent application to migrate a tech stack](#) (20 August 2024)

[Onboarding to a 'legacy' codebase with the help of AI](#) (15 August 2024)

[How to tackle unreliability of coding assistants](#) (29 November 2023)

[How is GenAI different from other code generators?](#) (19 September 2023)

“We now have this new tool that allows us to specify things in an unstructured way, and we can use it on **any abstraction level**.

We can create low code applications with it, framework code, even Assembly.

I find this **lateral move** much more exciting than thinking of natural language as ‘yet another abstraction level’ “

Simon Willison

Creator of Django, independent software
engineer

~25 YOE



Andrej Karpathy ✓

@karpathy



Congrats to Simon Willison (@simonw) on 23 years (!!) of blogging. Really excellent LLM blog, I sub & read everything:

simonwillison.net

(e.g. I sub via RSS/Atom on NetNewsWire)

+If you consistently enjoy the content like I do, sponsor on GitHub:

github.com/sponsors/simonw

7:48 PM · Jun 13, 2025 · **582.1K** Views



“Simon, how would you summarize the state of GenAI tooling, right now?”

“**Coding agents** are a thing that actually work now: run an LLM in a loop, let it execute compilers and tests and linters and other tools, give it a goal and watch it do the work for you.

The model improvements **in the last six months** have tipped these over from fun toy demos to useful on a daily basis.”

So far, what we've heard:

| | |
|-----------------------------------|-----------------------------------|
| 1. AI dev tools startups | Heavy usage |
| 2. Big Tech | Heavy investment, growing usage |
| 3. AI startups | Growing usage, Can be bit-or-miss |
| 4. Independent software engineers | More enthusiastic than before |

But... there are still many questions

... here are 4 of them

#1: Why are founders and CEOs a lot more excited than engineers?



Zach Lloyd  · 1st

Founder / CEO at Warp

2w · Edited · 



Anyone else having a hard time getting their most senior engineers to consistently use AI developer tools and model their use to the engineering team?

The most enthusiastic adopters on our team are designers, PMs, and me.

This makes sense because for our designers and PMs these tools (e.g. **Warp** and Cursor) make the impossible possible, whereas our more senior developers can already accomplish what the tools unlock for others and at most the AI provides a speedup.

However, this is unsustainable, and no-one developing software going forward should be doing it all by hand.

Ideas on how best to get our senior engineers to fully make the leap to starting every coding task with a prompt?



Annie Vella and 307 others

453 comments · 19 reposts

#2: How “mainstream” or “niche” is LLM usage across devs?

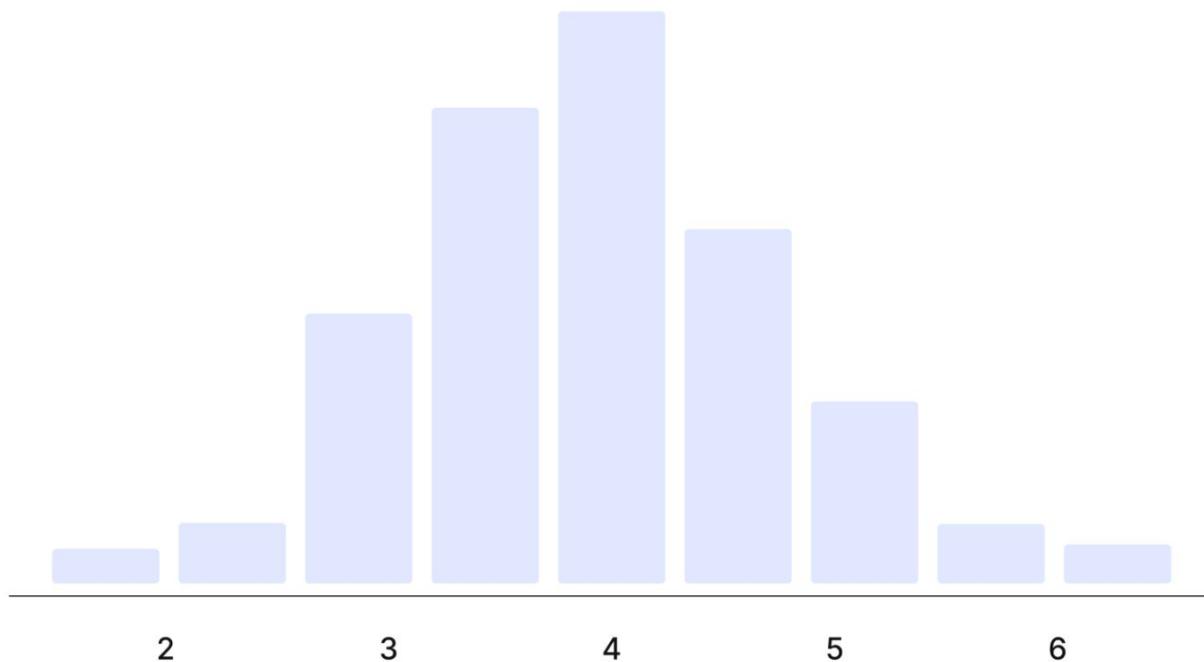
Data from DX study with 38,000 devs:

- Median: 5 out of 10 devs use AI tools weekly (**50%**)
- “Top” companies: 6 out of 10 devs (**62%**)

... so most of the stories we've heard today: they are all above the median

#3: How much time do LLMs actually save, when used?

Hours saved per developer/week



Based on 38,880 developers across 184 orgs — **DX**

#4: Why does it work so much better for individuals than teams?

Laura Tacho, CTO at DX:

“These tools are great for the individual developer right now: but not yet that good at the organizational level.”

#5: Why is it hard to get excited about it without using it?



“If you start with the LLM theory, it will hold you back.

It's weirdly harmful to spend too much time trying to understand how LLMs actually work before you start playing with them, which is very unintuitive.”

– Simon Willison

In summary

I am not surprised the least at:

CEOs and founders of LLM companies
are enthusiastic

Big Tech investing heavily into AI tooling

Startups experimenting with AI tools

What makes me pay a lot more
attention:

Experienced, independent software engineers finding a **lot** of success with LLMs and **wanting** to use them more!

**We are seeing a step-change happen in
how we build software, looking ahead**

Martin Fowler:

“I think the appearance of LLMs will change software development to a similar degree as the change from **assembler** to the first **high-level programming languages**.

The further development of languages and frameworks increased our abstraction level and productivity, but didn't have that kind of impact on the nature of programming.

“LLMs are making that degree of impact like high-level languages had versus the assembler.

The distinction is that LLMs are not just **raising the level of abstraction**, but also forcing us to consider what it means to program with **non-deterministic tools.**”

Observation from Kent Beck:



“I’m having more fun programming than I ever had in 52 years.”

“Kent, how do LLMs compare to past technology step changes you’ve lived through?”

“I’ve seen similar changes in the past, impact-wise:

- **Microprocessors** (70s-80s): the shift from mainframe computing
- **The internet** (2000s): changed the digital economy
- **iPhone and Android** (2010s): suddenly things like live location sharing is possible, and the percentage of people being online dramatically increased”

“The whole landscape of what's ‘cheap’ and what's ‘expensive’ **has all just shifted.**”

Things that we didn't do because we assumed they were going to be expensive or hard just got ridiculously cheap.

So we just have to be trying stuff!”

It's time to experiment more with LLMs

Thank you

As always, I'll keep a pulse on what works, and what doesn't in The Pragmatic Engineer.

(Subscribe to keep your ear to the ground)