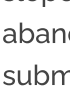
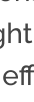


[Home](#) > [Crown car](#) > Most slopcode projects are abandoned and deleted within months of release

Most slopcode projects are abandoned and deleted within months of release

Thom Holwerda

2026-07-07

Crown car

38 Comments

About a month ago, Flathub [announced](#) a ban on slopcoded applications. Evangelos "GeopJr" Paterakis, developer of a number of popular Linux applications and ton of other things, did some research into just how many applications tagged with "AI slop", a tag Flathub reviewers used to keep track of slopcoded applications submitted to Flathub, actually survived the test of time. The results are exactly what you'd expect.

Of the 120 unique repos, 32 were maintained and 88 were abandoned. No seriously, a big portion of them was completely deleted, nowhere to be found, others stopped 6 months ago, right after submitting to Flathub.

[↔ Evangelos "GeopJr" Paterakis](#)

That's absolutely soul-crushing. Why should Flathub's reviewers spend their precious, limited time talking to lazy slopcoders' "AI" agents to get their slopcoded applications into Flathub, when 70% of these applications are abandoned or outright deleted from existence within mere months of being submitted? Minimal effort for the slopcoders, maximum effort for the reviewers. Just dump a bunch of shitty code over the fence, let a chatbot handle the interactions with the reviewers, and pretend you made a valuable contribution.

This is the contradiction slopcode enthusiasts really don't want to talk about. If these "AI" tools are so great, where is all the amazing new software? Where's the massive gains in software quality? Isn't the story that "AI" tools do the menial work, giving programmers more time to focus on improving their software? Reality does not seem to match the story we're being sold. Despite these slopcode tools being out and available for years now, there's no influx of great applications and other software, there's no rise in software quality, nothing.

What we mostly seem to be getting are slopcoded projects nobody, not even their "creators" care about, so they just get abandoned and deleted as quickly as they were dredged up from the bottom of the programming barrel. These aren't applications created because someone wanted them to exist; these are applications created because some mid programmer got high on their "AI" supply and fancied themselves better at programming than they really are – only to realise once the comedown hits they've got crappy, barely working, entirely unmaintainable gibberish vaguely looking like code nobody can make head nor tails of.

And then they abandon the project, ready for the next high – leaving everyone else to clean up their mess.

What a miserable workflow.

DONATE TO OSNEWS

YOU CAN ALSO:

- [Become a Patron](#)
- [Buy OSNews merch](#)

SEARCH OSNEWS

Search

OSNEWS ACCOUNTS

- [Create Account](#)
- [Login](#)

TOPICS

Select Category

RELATED POSTS

[Google's new 'AI' Health Coach started making shit up right away](#)
May 15, 2026 · 4 Comments

[They're putting blue food coloring in everything](#)
July 20, 2025 · 19 Comments

[Musk's Tesla warns Trump's tariffs and trade wars will harm Tesla](#)
March 14, 2025 · 6 Comments

[Vibecoding: nothing more than meowing nuns](#)
September 5, 2025 · 12 Comments

[Humane is shutting down the AI Pin and selling its remnants to HP](#)
February 18, 2025 · 6 Comments

POPULAR POSTS

[You paid me, a long-time Linux user, to use Windows 11 exclusively for a month: here's how it went](#)
July 8, 2026 · 56 Comments

[What was nice about the UI of Windows 2000](#)
June 19, 2026 · 57 Comments

[NVIDIA unveils RTX Spark chip for laptops and desktop PCs](#)
June 1, 2026 · 42 Comments

[Most slopcode projects are abandoned and deleted within months of release](#)
July 7, 2026 · 38 Comments

[Microsoft capitulates again, extends Windows 10 support by another year](#)
June 27, 2026 · 31 Comments

POPULAR POSTS (LAST 3 YEARS)

[Is 2024 the year of Windows on the desktop?](#)
June 20, 2024

[Apple intentionally kills web applications for EU users in iOS 17.4 onward to spite its EU users](#)
February 15, 2024

[Made O'Meter helps you easily and quickly avoid American products](#)
March 6, 2025

[SHOME, not so sweet SHOME](#)
August 19, 2023

[Some personal news](#)
March 23, 2024

About The Author



Thom Holwerda

Follow me on Mastodon [@athomholwerda@exquisite.social](mailto:athomholwerda@exquisite.social)

38 Comments

2026-07-07 8:18 PM

Andreas Reichel

How is your whale doing, Thom?

At this point, I don't even know what you are trying to achieve here. On the one side, SME software companies, who realign their complete workflow and business model around good LLM models. On the other side, people (who don't even write code) condemning everything that has anything to do with LLMs/AI (not matter what).

Why is it so hard to accept, that some people (who actually serve their customers and are successful in their business for decades) find those new tools useful, and others do not?

I have actually no problem, to acknowledge that you don't like LLM/AI. I see not need to continuously tell you how dumb you are in my eyes for not using it or to call you work any names, because I don't like your tools. (To make it clear, I don't hold you for dumb at all. I know that you are a very intelligent, just hurt because reality pulled the rug under your feet.)

So why are those continuous indirect and direct insults needed? What about you accepting our world same as you expect we accepting yours?

Log in to Reply

2026-07-08 12:53 PM

FlyingJester

I think that's a very false dichotomy. There are plenty of professional developers who dislike AI for very real reasons. For instance, the person who wrote the linked article.

I'm also a professional software developer of over 10 years now. I've witnessed in their business for decades critical thinking skills, tempts people to just skim over problems because AI can just solve it instead, and leads to a codebase filled with things people don't really understand. It's like all the tropes of what would happen if you just copied everything from stack overflow, but now it's automated.

I think that developers' time is better spent writing scripts and understanding problems than using chatbots. And becoming dependent on them not just for writing code, but for understanding it, won't be a very profitable endeavor when the AI vendors finally need to actually become profitable.

Log in to Reply

2026-07-08 4:16 PM

CaptainN

Most of the software people are paid to "generate" is derivative, disposable commercial software. It's always been disposable trash. And AI is exceedingly good at generating this kind of slop, and it's genuinely useful.

What the AI enthusiasts don't want to (or don't have expertise to) admit, is that it basically cannot solve a single novel problem. LLMs are simply not good at that. They generate tokens to an average, and don't ever understand anything.

The thing the AI skeptics (and haters) don't want to admit, is that LLMs don't have the ability to solve every novel problem to be useful.

The same arguments apply to anything AI can generate, including non-software writers (though we disrespect software writers the hardest by suggesting their work is somehow less creative than other forms of expression, but I digress.) It doesn't solve all problems, but it's still useful. Not complicated, but very disruptive, and there's a high degree of utter hyperbole built up around this tech, on both sides.

Around and around we go.

Log in to Reply

2026-07-08 5:29 PM

Alfman verbose-1

CaptainN-

The thing the AI skeptics (and haters) don't want to admit, is that LLMs don't have to be able to solve every novel problem to be useful.

The same arguments apply to anything AI can generate, including non-software things (though we disrespect software writers the hardest by suggesting their work is somehow less creative than other forms of expression, but I digress.) It doesn't solve all problems, but it's still useful.

I've also been saying this. Critics often berate LLMs for being imperfect, which while true, ignores the fact that humans are also imperfect. The value proposition for using AI doesn't just exist when an AI achieves perfection. I have no interest in selling anyone on AI, but I do think people need to understand that even just a mediocre/average AI can be potentially disruptive next to average workers.

It is my belief that long term AI will ultimately displace lots of human jobs, but we're also likely to see AI take on roles that are simply going unfilled by humans – not because humans can't do it but because corporations are cheap bastards that don't pay developers to work on house cleaning projects.

Most production code in C/C++ contains hidden vulnerabilities, and yet managers and executives often deprive development teams of resources needed to proactively weed these out. It's not the fault of developers, but still code quality suffers. I think LLMs could start filling more of these important yet unfilled roles for less than the cost of hiring minimum wage interns.

Log in to Reply

2026-07-08 10:55 PM

FlyingJester

The thing is, that's kind of what makes it slop. That it is no better than the worst code a human can write that is shipped in secret only. That's also the worst part of AI, that it enables the worst tendencies of development, only now at scale.

I don't agree with this. I think it's biased and believe humans are even less consistent. But it's ok, we don't really have to agree.

2026-07-08 11:24 PM

Alfman verbose-1

FlyingJester.

The thing is, that's kind of what makes it slop.

The problem is that if we judge human code by a similar standard, we'd also have to call that "slop". That is CaptainN-'s point.

That it is no better than the worst code a human can write that is shipped in secret only. That's also the worst part of AI, that it enables the worst tendencies of development, only now at scale.

I don't agree with this. I think it's biased and believe humans are even less consistent. But it's ok, we don't really have to agree.

2026-07-09 11:42 AM

Serafean

I have no problem accepting that some people find it useful. What I have a problem with is when LLM-generated MRs waste review time because the code is crap. And this is exactly the same situation : reviewers and repo maintainers get burdened with the abandonware in flathub. Creating more work for everyone else.

Why am I OK with crap code from a human? Because over time, the human will likely get better from the feedback we give. Also the bad human likely wouldn't create that much code.

I care about LLM being used and uselessly increasing complexity in the software commons, meaning in FLOSS code. What happens in closed source I really don't care.

Log in to Reply

2026-07-07 8:36 PM

Andreas Reichel

To discuss the matter at hand objectively: Please look at the study is Coelho, Valente, Silva & Shihab. "Is this GitHub Project Maintained? Measuring the Level of Maintenance Activity of Open-Source Projects." <https://arxiv.org/pdf/2003.04755>

Summary of the language finding: The authors compared survival curves for the top-5 languages in their set of unmaintained projects and found the distributions were statistically different (Kruskal-Wallis, $p \leq 0.05$). The sharpest contrast was Java versus Ruby — at the four-year mark, Java projects had roughly a 38% survival probability against about 79% for Ruby, with the other major languages falling in between. So Java showed the lowest survivability of the group.

Done in 2020, "slopcode" has changed anything on the question, how many projects would survive.

Looking at my own repos <https://github.com/manticore-projects/flat-repositories> I can see at least 2 abandoned indeed (they just did not stand the test of time) while half of them look "unmaintained" (simply because there is nothing to do) — leaving half of them active. Welcome to the 1/10 rule of success in real business.

Log in to Reply

2026-07-07 9:34 PM

Morgan

I was excited to see a new video from Youtuber Action Retro (Sean), where he created a simple writing app for the original compact Macintosh line of computers, so those enthusiasts would have a somewhat modern, Markdown format, full screen distraction-free editor for those old computers.

Sadly (though props to him for being upfront about it), he revealed that it's slopcoded via Claude. I had to stop watching right there. I have zero interest in following a project that is likely full of substandard and mis-licensed code. That Sean himself probably doesn't understand (again to his credit, he's not a developer so it's probably about the same quality as if I had tried to use Claude to write something). I mean, I truly appreciate his effort to do this; his "Frogfin" website is phenomenal and a valuable resource for retrocomputing enthusiasts like me. I just wish he didn't feel the need to consult a hallucinatory, plagiarist, autocomplete toy rather than lean on the talented and diverse group of friends he's made over the years in the retrocomputing scene, many of whom are seasoned developers who would probably love to collaborate on such a neat little idea.

I'm not going to stop watching his channel, despite this incident and the general downward trend it's been on — he seems to be going through some health issues but I'm not going to speculate further as it's not my business — but he's still an awesome guy and a genuine boon to the retrocomputing scene.

Log in to Reply

2026-07-07 9:45 PM

Andreas Reichel

I had to stop watching right there, I have zero interest in following a project that is likely full of substandard and mis-licensed code.

This is your good right to do, although I would have loved to see a rational example of such sub-standard code to back your claim. Otherwise, it sounds a bit like "don't buy from Jews", or if you prefer a more modern German context "don't buy Chinese cars". We both know, how this ended.

Log in to Reply

2026-07-07 10:32 PM

FlyingBoats

'Otherwise, it sounds a bit like "don't buy from Jews", or if you prefer a more modern German context "don't buy Chinese cars". We both know, how this ended.'

Because avoiding certain production methods and racism are totally the same thing. Parrotting this argument isn't doing you any favours.

Log in to Reply

2026-07-07 10:39 PM

Andreas Reichel
Luckily I am not depending on any favours, but lets try another example then:

Because avoiding production methods...

I am sure, you know who invented "made in Germany" and why? But this one ended well for sure!

Log in to Reply

2026-07-08 3:18 PM

Morgan

You can't help but troll can you? And this time somehow equating a personal preference for a video about software development with anti-Semitism of all things? This is a new low for you, literally calling me a Nazi because I don't like a particular developer practice. What an infantile and ignorant take. What's next, you going to call me a serial killer because I use Nano instead of V?

Log in to Reply

2026-07-08 12:42 PM

FlyingJester

Yeah, I've had similar experiences with channels making things I am interested in, but using chatbots. It's mostly the lack of understanding. Using a chatbot very often means the person doesn't really have a deep understanding of the problem space or the thing they built, and it becomes very apparent when they go to make a video explaining it.

Log in to Reply

2026-07-08 2:07 AM

cheemosabe

Water is wet. It's so easy to build an app now that non-developers now write apps as test prompts to see what will come out. Some don't even know what maintenance means. The future is here now, it's just very unevenly distributed: <https://blog.exe.dev/building-software-from-your-phone>

That's just one company where most coding is done by LLMs.

Log in to Reply

2026-07-08 3:24 AM

Alfman verbose-1

cheemosabe.

Water is wet. It's so easy to build an app now that non-developers now write apps as test prompts to see what will come out. Some don't even know what maintenance means.

That seems logical to me. Whereas hiring traditional software developers used to require a large commitment, the *relatively* low cost and easy accessibility of using LLM "slop code" as everybody likes to call it makes it far easier for unprofessional amateurs to create one-off projects.

I have mixed feelings about this. On the one hand it's understandable why some are against these cheap & unprofessional software methods..but at the same time it would be hard for me to say that software development should only be accessible to elites.

There's two sides to every argument. Take a look at generative AI videos... there's a ton of slop, which I hate with a passion. Furthermore it's also clear how AI tools threaten professionals, but on the other side there's a ton of opportunity to democratize high quality VFX and CGI in the hands of everyone. In general FLOSS is about the democratization of software and tools, so arguably FLOSS advocates should be in favor of training models on open data, empowering users while respecting rights.

Log in to Reply

2026-07-08 4:25 AM

cheemosabe

Yes, I have mixed and even plenty of negative feelings about AI in general. While for science and coding there are and will continue to be clear benefits (clear downsides too), with video it's difficult to see the positive side, at least at the moment. I don't think I've yet seen an AI generated video that I felt brought something positive for me. We have the historical example of synthesizers, which didn't kill regular musical instruments, but AI makes it possible to create music/video with little input. I guess humanity will get to decide what it finds valuable in whatever emerges. Maybe there will be some interesting audio/visual techniques that will be discovered with AI that people will then be able to apply using traditional methods (similar to how new chess strategies were discovered by AIs since AlphaZero).

Log in to Reply

2026-07-08 2:54 AM

Alfman verbose-1

Thom Holwerda,

This is the contradiction slopcode enthusiasts really don't want to talk about. If these "AI" tools are so great, where is all the amazing new software? Where's the massive gains in software quality? Isn't the story that "AI" tools do the menial work, giving programmers more time to focus on improving their software?

I understand that was rhetorical and that you'd rather ignore actual examples of this, but the fact is AI tools are genuinely making progress at tackling real problems.

<https://thehackernews.com/2026/06/ai-agent-uncovers-21-zero-days-in-html>

I do believe AI should be debated because there are costs to society and IMHO these costs could be very significant. However I do think that for better or worse these assumptions that AI can't do anything useful are becoming dated. AI is proving to be capable at taking on tasks that, for one reason or another, humans developers haven't been doing to a high standard themselves.

Log in to Reply

2026-07-08 3:05 AM

Andreas Reichel

I do believe AI should be debated because there are costs to society and IMHO these costs could be very significant.

Absolutely, I have been talking to some really talented software engineers and SMEs — and all of us a genuinely scared.

The typewriter replaced the scribe. The automobile replaced the horse, and the steam-hammer replaced the blacksmith. But that took decades for each niche — while now LLMs compete in every aspect within less than a year and accelerating. We struggle just to keep up.

Example: I have a fairly complex Frontend for running ARIMA or Holt/Winters models on time series. I gave Claude/Fable just a screenshot of this UI not even explaining the underlying mathematical context — and it mocked a much!!!! better UI and design including ACF, PACF, Residuals and Ljung-Box test! I understood what I was doing, just from showing a screenshot!

Log in to Reply

2026-07-08 12:11 PM

Kochise

We shouldn't be afraid but humbled. Keep in mind that without humans first, there would be no AI and knowledge base to learn from. We as a specie just have to adapt to the new tools and stop pretending we should return back to weeks long development processes when we can have it just as good in hours. We should now focus on something else still left unattended : peace over the world.

Log in to Reply

2026-07-08 1:53 PM

Alfman verbose-1
Kochise,

We shouldn't be afraid but humbled. Keep in mind that without humans first, there would be no AI and knowledge base to learn from. We as a specie just have to adapt to the new tools and stop pretending we should return back to weeks long development processes when we can have it just as good in hours. We should now focus on something else still left unattended : peace over the world.

That's true when talking about LLMs that rely extensively on training data. However other branches of AI can build up their own knowledge without having to extract it from "human" sources.

For example AI learned to create novel strategies for playing chess better than the humans who taught it. I believe this will happen for programming too, but it takes time. I suspect what will ultimately prevail is a hybrid approach. LLMs will still provide human interfaces, which they are good at, but instead of having an LLM to actually solve technical problems, the it will be able to build domain specific AIs that perform better than the LLM in the same sense that humans have built domain specific AIs that perform better than humans.

We're not there yet, but I have a feeling that this and not LLMs is going to be the next step in super human AI. I think it's all technically doable, but whether we should or not is a whole other debate.

Log in to Reply

2026-07-08 5:29 PM

Kochise

Quite frankly I'm rather as ecstatic as Dr. Károly Zsolnai-Fehér from <https://www.youtube.com/@TwoM> about AI and image generation.

<https://users.cg.tuwien.ac.at/zsolnai>.

We just have to adapt like in 2001 a Space Odyssey, from monkeys to stellar people.

"Any sufficiently advanced technology is indistinguishable from magic" – Arthur C. Clarke

2026-07-08 5:50 PM

Alfman verbose-1

Kochise,

Quite frankly I'm rather as ecstatic as Dr. Károly Zsolnai-Fehér from <https://www.youtube.com/@TwoM> about AI and image generation.

We just have to adapt like in 2001 a Space Odyssey, from monkeys to stellar people.

"Any sufficiently advanced technology is indistinguishable from magic" – Arthur C. Clarke

Indeed. I think that AIs of this type, i
with physics accurate simulators, c
start solving real world physical pro
is so much happening in the field of
LLMs..but for whatever reason near
media coverage sticks to LLM chat

2026-07-08 3:14 AM

Tuxie

I've been programming since the mid 1980s and I've been a professional UNIX/Linux sysadmin for almost 30 years. I know how to write good code and manage servers but I just don't think it's fun anymore.

I've generated an awful lot of slop for personal use lately and I love it. I fully realize that almost all of it is unreleasable as-is, but it's tailored to my personal wishes and instead of designing configuration UIs I can just fire up opencode and tell it to change the code when I want to tweak something.

The vibe slop workflows are strongly biased towards putting the code on github and making releases, even if they are not even intended for others to use. Eventually I had my LLM install a personal Forgejo server so now I have a personal private Github/DockerHub clone with Github-compatible actions/workflows and the LLM agents can use it to both post and handle issues and create, review and merge pull requests, without anything leaving my home LAN. All running on a personal server in Dockhand behind a Traefik reverse proxy.

I hardly even sysadmin anymore. I just open OpenCode and type things like "Make service X only reachable from the client with MAC addr 00:11:22:33:44:55" or "Create a nightly backup job of service X to the Backblaze B2 bucket foo and verify a full service restore after every backup in a temporary container. Then send me an email with the full backup, restore and verification log."

[Log in to Reply](#)

2026-07-08 3:19 AM

Andreas Reichel

almost all of it is unreleasable as-is

Please allow me to ask: why? Can you elaborate with examples?

[Log in to Reply](#)

2026-07-08 3:40 AM

Tuxie

It's often hardcoded-tailored to my specific system and wishes and the quality of the code is often not something I want to put my name on. But it works for me. It's like having a messy apartment that's cosy to chill in, but you'd want to clean it up before inviting someone.

The biggest reason is that I don't have energy to deal with users with different needs, though. I vibe stuff to solve immediate personal needs, then I want to move on, not deal with bug reports for things I'm not personally affected by. I already do more than enough of that during work hours.

[Log in to Reply](#)

2026-07-08 3:50 AM

Andreas Reichel

Thank you. I really appreciate.
I found LLMs especially helpful for deployment (Gradle build scripts, GitHub Actions, Git commits) and UI mocking. Less useful for Business Object Schemas and data structure, those I still do by hand.
Useful tool for some (users or tasks), not for others. Fair.

[Log in to Reply](#)

2026-07-08 4:02 AM

r_a_trip

Temporary problem. LLM coding is relatively new. In the hands of professionals, it probably speeds things up. In the hands of curious non-coders, it most likely yields software that will be abandoned as soon as it is generated. Once LLMs disappear in the background as just another tool in the arsenal of tools, the curious non-coders will have long since abandoned their sense of novelty and the influx of unmaintained, non-coder software will dry up as well.

[Log in to Reply](#)

2026-07-08 6:13 AM

Adurbe

Honestly, this feels more a Flathub problem than an AI one. Implement a 6 month minimum age for a project and set the expectation that code should be patched a minimum of a monthly basis (really should be more regular). The whole problem effectively goes away.
They have a really low bar atm, including multiple "fan" apps that are mere wrappers of real ones that remain unpatched.

[Log in to Reply](#)

2026-07-08 11:34 AM

NaGERST

What if the app is feature complete when packaged as an flatpack and posted to flathub? Example (and perhaps a bad one) if i make a flatpak of TeaPot (the most complete CLI OSS spreadsheet application afaik), why would it need updates? It was first released in the 90s and has only had minor changes for the last 20 years. Yeah of course, it passes the age test, so this problem might not apply to your solution. Maybe i need newer software in my example.

[Log in to Reply](#)

2026-07-08 4:26 PM

Adurbe

While that's perfectly possible. Very few modern apps are built without dependencies. And dependencies get patched and need patching.
Even a super basic electron wrapper has weekly patches and updates. At a minimum a project should have a release cycle to include them.

[Log in to Reply](#)

2026-07-09 11:52 AM

Serafean

> While that's perfectly possible. Very few modern apps are built without dependencies. And dependencies get patched and need patching.
That's a flathub problem: dependencies shouldn't be vendored, precisely for this reason.

[Log in to Reply](#)

2026-07-08 11:34 AM

mebarg

I don't see the problem with Slopcode; it's just code—it solves a problem, and that's it. You used to write it yourself, and now the LLM does; is it better than what you wrote? Great. If not? Write it yourself. I enjoy programming—I'm not a professional, but I build the programs I need for the tasks at hand—and with the arrival of LLMs, I can do more in less time and with better results than before. It works for me.

[Log in to Reply](#)

2026-07-08 1:42 PM

Cytor

AI works well as a coding tool in a professional environment. Who cares what is going on with some random Flathub apps. Although I do not like the use of AI in creative use cases... such things really need the human touch.
At this point, I think Thom should just post any news on AI, no matter good or bad. It gets really old.
Dang, got baited into commenting on an AI post again 😞

[Log in to Reply](#)

2026-07-10 2:43 AM

vasilito

For me, AI tools help in my daily work. I am a developer of a free public service app "WayHome" to help addicts overcome addiction habits. AI allowed to speed up development quite a lot. Before I started using AI for coding tasks, we had the assembly meetings functionality nearly stalled. Slopcode allowed not only to finish this feature but make it a premium experience for it's users. The number of installations is growing. So in my experience the motivation and dedication is what makes the product either success or failure and not using or not using AI.

[Log in to Reply](#)

2026-07-10 9:08 AM

mkone

I would draw a parallel between AI coding and digital photography – bear with me here.
A long time ago, to be a good photographer, you had to understand composition, lighting, all the various knobs on a professional camera etc. If you didn't, your SLR photos were worse than rubbish. Honestly, you would have been better off with a point and shoot fixed lens camera. It was very costly to be terrible at those things because you were usually working with a 24 (or 36) exposure film, and you could not afford to be mediocre at your job.
Digital photography changed it. While it is still important to understand all of those things that professional photographers once understood, the nature of digital cameras means your camera can not do a lot more for you. In the hands of a professional, it provides even more benefit. However, digital photography dramatically reduced the cost of taking photos to the point that the world is now almost overwhelmed by photos. What digital photography has done though, is reduced the barrier to entry. The vast majority of people who dabble in digital photography will not become professionals, and their pictures will not be seen by anyone other than their close friends and relatives.

So it is with AI coding. For most unskilled people who are dabbling as a hobby, what they produce will be useless and go no further than themselves and one or two people. For really skilled people and those who are solving actual problems, AI will increase their productivity.
The fact that there is so much slop out there is not important and is not an indictment of AI.

[Log in to Reply](#)

2026-07-10 9:42 PM

Andreas Reichel

This is the nicest and fairest assessment of this emotional debate. Only a minor remark: when you can share your photos w/ family or friends, then they won't be "useless" not matter how bad an amateurish they are (and I am not talking about the pictures here).

[Log in to Reply](#)

2026-07-11 8:15 AM

vasilito

excellent comment

[Log in to Reply](#)

Leave a Reply

You must be [logged in](#) to post a comment.

ABOUT OSNEWS

OSnews Privacy Statement
© OSnews Inc. All Rights Reserved.
OSnews and the OSnews logo are trademarks of OSnews.
Theme Customizations by Adam Scheinberg
Proudly powered by Wordpress
Reader comments are owned by the poster.
We are not responsible for them in any way.
All trademarks, icons, and logos shown or mentioned in this web site are the property of their respective owners.
Reproduction of OSnews stories is permitted only with explicit authorization from OSnews.
Reproductions must be properly credited.

IMPORTANT LINKS

- OSnews Privacy Statement
- Sponsorship
- OSnews on Mastodon
- Notice to Bulk Emailers
- Contact OSNews
- Trans rights are human rights

FROM THE ARCHIVES

Select Month