

When AI Can Build Anything, Your Job Is Deciding What's Worth Building

Yuri Cataldo (RavenRock VC)

KEYWORDS: Product Development, AI, entrepreneurship, strategy, management.

Generative AI removes many barriers to execution, putting greater pressure on founders to choose the right customers, products, and ideas.

At Yale Drama School, where I trained as a costume designer, we were assigned a new play or opera to design every other week. Every Saturday at 9 a.m., we presented the work: costume sketches, scale models of the set, and the visual research behind them. We pinned it all to the wall and stood beside it while the faculty took our ideas apart. The critiques often ran past dark.

My main professor was Ming Cho Lee, who had taught there since 1969 and trained two generations of American designers. Ming could find the one choice you'd made out of habit rather than intent, and he'd point at it and ask you why. "I liked it" was not an answer. You had to say what the choice was doing and what you'd rejected to get there.

I hated these "crits" for most of my first year. Then I noticed that the classmates pulling ahead weren't the most talented people in the room. They were the ones who had stopped defending their work and started judging it. I copied them, and my designs started winning awards.

That skill, judging your own ideas before customers do, is the one AI just made more valuable. AI now lets one person build in days what used to take a team months. But speed doesn't create demand. When building gets cheap, the bottleneck moves from execution to judgment: deciding what's worth building at all. Art school has a method for training that judgment, and it transfers to business.

The Bottleneck Moved from Execution to Judgment

Early-stage AI prototypes keep getting better. The answers to "who asked for this and why now?" don't. Thanks to coding agents, building software is no longer the hard part. A landing page that used to take two weeks takes an afternoon. A prototype that used to occupy a small team for a quarter now takes a long weekend.

Everybody loves the speed part, especially founders. But when you can build 10 things in the time it used to take to build one, you don't get 10 times the progress. You get 10 things nobody asked for. The constraint that was quietly protecting you all those years was the sheer difficulty of making anything at all, and difficulty forced you to choose.

Now that pressure is gone, and it exposes something uncomfortable. Most of us were never any good at deciding what to build. We were just slow enough that nobody found out.

I know because I did it. In 2023, before coding agents, I worked with a small team for months building an app so my MIT Sloan classmates could stay connected. Within minutes of launch, 177 people signed up across four schools. Nobody used it. The first lesson was obvious: Nobody wants another app on their phone. The second lesson took longer. People who signed up spent a few minutes looking around and had no idea what to do next. I'd confirmed that people thought the problem was real. I never once asked them what would make them open it on a Tuesday. Those are different questions, and I only answered the first.

I rebuilt it this year. The fix wasn't a better app. It was meeting people where they already are: WhatsApp. The setup is simple, and it makes weekly intros to the 254 members from 14 different business schools and the



MIT founder program. It works because opening it isn't a decision members have to make. It's built into something they already do every day: Open WhatsApp. The tools got better in three years, but that didn't solve my problem. What changed is that I finally asked what the thing was for instead of what it could do.

AI will build whatever you want without complaint and will never once ask whether the thing is worth building in the first place. That question belongs to you. It always did. Now it's the only one left.

Where the Work Went, and Where the Judgment Stayed

Ideation. AI will hand you 50 ideas in 10 seconds, and every one of them will sound reasonable. This is the problem. These models are trained on existing data, so what comes back is the consensus of everything already built. Generating ideas is nearly free now. Knowing which idea people will still pay for in 18 months costs everything.

Prototyping. A good prototype is like a good argument. If it contains everything, it proves nothing. AI will build everything you ask, and it will never tell you that the build list is too long. Deciding what to leave out is the whole design act, and it's the part that feels like doing nothing, which is why people skip it.

Validation. AI will take 100 interview transcripts, cluster them by theme, surface the outliers, and hand it all back in a minute. It'll also give you the answer your question was fishing for, because you wrote the prompt and your assumptions rode in with it. Actual customer interviews are messier than that. Customers describe symptoms. Diagnosing the cause is your job. That translation, from what they said to what's broken, is the step that doesn't survive being handed off. One question does most of the work here: what did you do the last time this happened? Past behavior is checkable. Stated intentions aren't.

Positioning. AI writes clean, competent copy that works for everybody. Positioning is the opposite of working for everybody. It's knowing who this matters most to and who you're willing to lose. Michael Porter said it in 1996, and it came up often when I was in business school: Strategy is making trade-offs, and the core of strategy is choosing what not to do, because without trade-offs there's no choice to make and therefore nothing to call a strategy.^[1] Ming was asking the same question at the

model table with a different vocabulary. Not "Is this good?" but "What did this beat, and what did it cost you to choose it?" AI has no stake in your project, so it will never think to ask what you said no to.

AI absorbs the labor and hands the judgment back to you. Your judgment is now the ceiling on your product.

So How Do You Get Better at It?

Judgment isn't something you're born with. You train it by exposing yourself, on purpose and repeatedly, to your own bad calls and advice from people who have trained to give it. Art schools have been running that drill for centuries. Business school teaches judgment through cases, which means judging somebody else's decisions from a safe distance then going home. Art school makes you pin your own work to the wall and stand next to it while people in the room talk about it. Here are the three practices that transfer.

1. Run an Actual Critique, Not a Status Meeting

Most teams believe they are already doing this. They aren't. What they do is a biweekly demo followed by supportive nodding and questions. That is meeting attendance, not constructive feedback.

A real critique has a shape to it. The work goes up and the person who made it says nothing, no context, no "so what I was going for here was." The urge is to narrate and explain your work. You have to sit on it, because the work must survive without a tour guide. Then everyone else describes what they see. Then the hard part, which is three questions: What is this doing, what does it cost, what would you cut? The maker takes notes and doesn't defend.

The room also gets no vote. Feedback, not verdicts. The second a room can overrule the maker, you're doing politics instead of criticism.

That last rule isn't mine, and it isn't an arts-school eccentricity. Pixar runs this meeting and calls it the Braintrust, and Ed Catmull is explicit: Candor alone isn't what makes it work; the absence of authority is. The director doesn't have to follow a single note. Whatever gets said in that room, it's on the director afterward to decide what to do with it. Catmull learned why that matters by getting it wrong: When Pixar tried the same format with its technical groups, it didn't work nearly as

well, and the reason turned out to be power. Once a room can overrule you, you walk in defending, and everything you say after that is positioning.^[iii]

Amazon runs a colder version of the same idea. Bezos banned PowerPoint in 2004 and replaced it with a six-page narrative memo that everyone reads in silence for the first 20 to 30 minutes of the meeting, before anybody says anything.^[iii] Different medium, same rule. The work has to stand up on its own without the author talking over it.

Ninety minutes, every two weeks, no defending and no verdicts. It's genuinely unpleasant for about a month. Then it becomes the most useful meeting on your calendar. And when you've spent three years having your work taken apart every Saturday morning, something else happens: you stop confusing the work with yourself. That separation is the actual skill here. Everything else is technique.

2. Put the Walls Back Up on Purpose

The worst thing anyone can tell you is "do whatever you want." Infinite options don't produce better work. They produce work that's inoffensive to everybody and urgent to nobody, which is an expensive way to fail.

AI hands you infinite options by default. Build the walls yourself before you start prompting. Pick one idea and commit. Build it for one customer you can name and describe in detail. Ship in five days or don't ship. That's the deadline on the test, not the company.

Yale gave us two weeks per production, and that deadline did more for the work than any amount of extra time would have. The constraint doesn't have to be the right one. It has to be real. Its job isn't to find the correct answer; it's to make the wrong ones show up fast enough to be cheap.

3. Keep a Decision Log and Actually Read It Back

Almost nobody does this. It compounds faster than anything else on the list.

Every time you make a real decision, write four lines. What you decided. What you rejected. What you expect to happen. By when. Thirty seconds, and you can do it in a notes app while you sit in a parking lot.

Then once a quarter, sit down with the last three months

of it.

The first read hurt. What I found in my own log was that my confidence and my accuracy had almost no relationship to each other. The calls I'd agonized over turned out just fine. The ones I made on autopilot, in a hallway, while balancing two other things, were the expensive ones. I'm one person with one log and I'd take that as an observation, not a finding.

Controlled research has found the same gap. In a 2025 randomized trial by the research nonprofit METR, 16 experienced open-source developers took 19 percent longer to finish real tasks in their own codebases when they used early-2025 AI tools. Afterward, they estimated, on average, that AI had made them 20 percent faster. The tools have improved since, and METR now treats that speed result as historical. The gap between how fast the developers felt and how fast they actually were is the part worth remembering.^[iv] That gap is why the critique alone isn't enough. A critique will make you feel more certain. Only the log will tell you whether the certainty was worth anything.

To put this into practice, here are three questions worth asking this week:

- What did I decide in the last seven days that I couldn't defend in 90 seconds?
- Who is this product explicitly **not** for?
- What's the cheapest piece of evidence that would change my mind, and when will I have it?

What It Comes Down To

The uncomfortable version of the AI story isn't that it replaces you. It's that it removes your cover. Every excuse you had for not testing the thing before you built it is gone. No more "we didn't have the resources." No more "that would have taken six months." You have the resources and it takes a weekend. So, what are you going to build, and why?

AI moved the bottleneck from execution to judgment, and art school has been training judgment for a long time. The three practices are how you borrow that training. The critique makes you explain a choice without hiding behind the effort it took. The walls make you choose before the tools let you build everything. The log tells you, months later, whether your confidence was worth anything. None of it requires new software. It requires doing the uncomfortable thing on a schedule.

Ming had opinions about everything and never once told us the right answer. He asked us what we had decided, and what it was based on, and he made us say it out loud in front of everyone until we could.

[i] Porter, M. E. "What Is Strategy?" *Harvard Business Review*, November-December 1996, p. 70.

[ii] Catmull, E., with Wallace, A. *Creativity, Inc.* Random House, 2014. See also Catmull, E. "Inside the Braintrust." Stanford eCorner, April 30, 2014. <https://stvp.stanford.edu/wp-content/uploads/sites/3/2024/09/inside-the-braintrust-transcript.pdf>

[iii] Bezos, J. Internal email banning PowerPoint, June 9, 2004. CNBC, October 14, 2019. <https://www.cnbc.com/2019/10/14/jeff-bezos-this-is-the-smartest-thing-we-ever-did-at-amazon.html>

[iv] Becker, J., Rush, N., Barnes, E., and Rein, D. "Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity." METR, July 10, 2025. <https://metr.org/blog/2025-07-10-early-2025-ai-experienced-os-dev-study/> See also METR, "We are Changing our Developer Productivity Experiment Design," February 24, 2026. <https://metr.org/blog/2026-02-24-uplift-update/>