

AI Feature Velocity Raises the PM Bar for Evidence

PM Daily Digest

2026-09-23

AI Feature Velocity Raises the PM Bar for Evidence

By PM Daily Digest • September 23, 2026

Current AI-product discussions show pressure to match competitors colliding with half-tested releases. Practical eval and discovery methods make both product failures and deliberate no-build decisions more visible.

Big Ideas

AI feature velocity is exposing a product-selection problem. A PM at a funded startup says competitor-chasing is pushing out half-tested workflows and questionable AI outputs; a commenter argues that as building gets easier, choosing what to build becomes the bottleneck. Use model capability for user understanding, experiments, and prototypes—not simply more features. That shift needs quality discipline: a prompt, model, or code change can improve one behavior while breaking another, so evaluation should be repeatable and happen before shipping. [1, 2, 3, 4]

Tactical Playbook

Define failures before metrics. The eval guide warns that teams often jump to metrics before studying actual failures, risking measurement of the wrong outcome. Its leasing-assistant example makes the point: when a prospect says the rent is out of budget, a polite goodbye sounds fine but misses the sales goal—the assistant should offer cheaper units or other properties. Start with the user-facing job, then decide what counts as failure. [4]

1. **Capture complete traces:** user input, system prompt, retrieval, tool calls, intermediate model calls, and final output. [4]
2. **Review before automating:** sample diverse traces; annotate the first 10 in actionable, user-facing terms and focus on the first upstream error.

Then let an agent propose more annotations for a human to accept or reject. Continue until learning plateaus—the guide’s rule of thumb is about 100 traces. [4]

3. **Prioritize recurring failures:** cluster and count failure modes across at least 100 diverse annotated traces. In a study of 100 production traces, automated tools caught obvious trace-level errors but missed issues requiring product judgment or outside context, and sometimes flagged good responses. Use automation with human review, not instead of it. [4]

Track what you deliberately reject. Petra Wille and Teresa Torres discuss adding trash-can markers to discovery and delivery boards to record customer problems and solutions the team chooses not to pursue. Use the record to ask whether the team is comparing solution options and whether discovery is working. An empty solution-space bin is a warning; an empty problem-space bin could indicate an innovation or culture problem, but may also reflect a strong strategy filter. Check whether people can safely raise problems, and use the record to retire “zombie” opportunities that keep resurfacing. [5]

Case Studies & Lessons

Lenny cites company-reported outcomes associated with eval investments: Ramp’s automatic receipt-collection accuracy rose from 35% to 83%; Shopify’s AI workflow builder was 2.2× faster and 68% cheaper than the frontier-model setup it replaced; Harvey nearly doubled its contract reviewer’s internal quality score; and Cursor reported higher user satisfaction at 41% lower cost after tuning Auto Balance. These are different measures, but show evals being applied to both product quality and operating cost. [6]

Career Corner

Eval-writing is a hiring signal: Lenny says nearly half of 25 PM openings he shared asked for experience writing evals. Candidates can demonstrate the skill by showing how they identified a user-facing failure, turned it into a repeatable test, and used recurring failures to guide product work—not just by listing AI tools. [6, 4]

Tools & Resources

Try the evals skill from Hamel Husain and Shreya, which Lenny linked as a way to save time and avoid mistakes; pair it with the human review process above. [7, 4]

Sources

1. r/ProductManagement post by u/Difficult-Rich-7302

2. r/ProductManagement comment by u/lily__de__valley
3. X post by @trq212
4. Advanced evals: How to find (and fix) hidden AI failures in your product
5. X post by @ttorres
6. X post by @lennysan
7. X post by @lennysan