

From Personal AI Hacks to Shared Product Learning Systems

PM Daily Digest

2026-08-24

From Personal AI Hacks to Shared Product Learning Systems

By PM Daily Digest • August 24, 2026

The strongest signal is a move from individual AI experimentation toward shared operating systems and prompt-led discovery, paired with tighter evidence loops for adoption and enterprise pilots.

Big Ideas

AI leverage is becoming an organizational design problem. Sachin Rekhi argues that the highest-leverage way to improve a team’s AI productivity is a team-wide “Compounding OS”—a company brain—not chasing the latest model releases. His three building blocks are an organization-wide agentic platform, a shared skill library, and machine-legible company context. [1] For PMs, the implication is to treat reusable skills and accessible context as shared product infrastructure, not isolated individual hacks.

Prompts are a new discovery surface. Hiten Shah’s “empty box” thesis is that not knowing what users will ask is the point: every unexpected prompt is product research. Clicks reveal how people use what you built; prompts reveal what product they wish existed. [2] Instrument prompt themes alongside funnel metrics, then turn recurring, high-value surprises into roadmap hypotheses.

Agentic UX needs a trust layer. Scott Belsky argues that the products that win will combine trust and inspectability with unusually strong hospitality and personalization. [3] For PMs, inspectability is a core experience requirement alongside delight—not a late compliance add-on.

Tactical Playbook

Diagnose adoption in two passes. First, instrument behavior before interviewing: analytics shows where, when, and for whom adoption breaks; conversations explain why. [4] Then split users at the first completed run: dropping before it points to discoverability, onboarding, or friction; completing once and not returning points toward value or workflow mismatch. Ask what they did instead. [5] In B2B, look for the shadow workflow—an old screen, spreadsheet, or email—and compare its step count, fit, and value. Also ask Customer Success which accounts received a rollout versus merely logged in. [5]

Treat enterprise pilots as evidence, not demos. Jen Abel’s enterprise-sales playbook argues that the familiar five-stage view hides a process closer to 15 steps, and that buyers are highly sensitive to the sales motion. [6] Adapt the sequence as follows:

1. Make the intro call a 30-minute, one-to-one conversation with no demo or slides; let the buyer speak first, then ask what needs to change, why now, and how success would be measured. [6]
2. Before a group demo, work with the champion to choose attendees and build the storyline; avoid becoming an unprepared checkbox presentation. [6]
3. Run a two- or three-day pilot with three or four power users, three explicit tasks, and co-authored success criteria. Work backward from signature and involve procurement, security, and legal early. [6]

Case Studies & Lessons

A 20-user beta exposed an activation problem, not a feature gap. A PM’s Claude Code-built language-learning app drew 20 immediate users from a relevant subreddit, but most left because they did not know what to do or find the product compelling. The next move was an immediate exercise to diagnose skill level and orient users. [7] The lesson: after acquisition, make the first action both useful and diagnostic before expanding scope.

Career Corner

Make the job search operate like a targeted funnel. A current profile of Basia Kubicka reports five inbound PM offers a week—including from PlayStation, Comcast, and Dialpad—and growth from zero to 70K followers in 18 months. [8] She treats LinkedIn as a landing page, uses Claude Code to compare five target job descriptions and requires at least 75% keyword overlap before rewriting her profile, maintains a story bank of real accomplishments and numbers, and adapts structures from posts that outperform a creator’s 30-day average by 10×. [8] Apply the same PM discipline: choose five target roles, test focus before positioning yourself, maintain an evidence bank, and iterate from observed response rather than generic AI output.

Tools & Resources

Use Claude Code for evidence hygiene, not PM judgment. One PM points it at exported CS notes, call summaries, and backlog items to cluster repeated needs, flag duplicates, and identify items thin on evidence; the PM still makes the prioritization calls. It stuck because incorrect outputs could be corrected in five minutes or less. [9] This is a useful boundary: automate reconciliation and synthesis, retain human ownership of framing and trade-offs.

Sources

1. X post by @sachinrekhi
2. X post by @hnshah
3. X post by @scottbelsky
4. r/ProductMgmt comment by u/DeanOnDelivery
5. r/ProductMgmt comment by u/Parking-Stress-3041
6. 84 minutes of enterprise sales alpha | Jen Abel
7. r/ProductManagement post by u/jabo0o
8. substack
9. r/ProductManagement comment by u/Parking-Stress-3041