

AI-Ready PM Execution: Shared Codebases and Decision Discipline

PM Daily Digest

2026-07-22

AI-Ready PM Execution: Shared Codebases and Decision Discipline

By PM Daily Digest • July 22, 2026

This brief covers the operating context PMs need to move AI work from prototypes to production, a five-step competitive-intelligence decision system, and a retention-diagnosis approach that avoids premature feature building. It also outlines a practical AI PM learning path.

Big Ideas

AI execution is moving from prototypes into the codebase

Aakash Gupta argues that leading teams can now make codebase changes through prompts, with PMs moving from AI-assisted documentation (2023) to codebase prototypes (2025) and production-ready changes (2026). [1] The differentiator is not prompting alone; it is the operating context available to the agent.

“Every team should be writing `CLAUDE.md`’s, `REVIEW.md`’s, skills, and docs that enable agents to work productively in the codebase with zero additional context from the prompt.” [1]

Apply it: Treat this as product infrastructure. Create a `CLAUDE.md` for project memory, a `REVIEW.md` that makes the quality bar explicit, and reusable “skills” for recurring workflows. [1] This gives PMs a concrete path to contribute through AI while keeping team standards visible.

A related product strategy is to build vertical interfaces around proprietary workflow graphs, using routers across competing—and increasingly interchangeable—models. [2] **Why it matters:** differentiation can reside in the workflow and interface rather than dependence on a single model.

Tactical Playbook

Turn competitor monitoring into a decision record

Competitive intelligence is most useful as a decision system, not a collection of competitor features, pricing, or announcements. [3] For each meaningful change, use this five-step record:

1. **State the observed signal** without interpretation: what actually happened? [3]
2. **Attach evidence**, prioritizing official product pages, release notes, or documentation over social posts and directory listings. [3]
3. **Add an interpretation—but label it as uncertain.** A change *may* signal something; it is not proof. [3]
4. **Identify the business implication:** enterprise pipeline, positioning, roadmap, sales objections, partners, or retention. [3]
5. **Assign an action owner and timing.** [3]

Why it matters: most competitor changes should be monitored or tested in customer conversations rather than triggering an immediate feature response. This structure helps prevent reactive copying. [3]

Case Studies & Lessons

Shared codebases can broaden who ships

Laurel CPO Jiaona Zhang built a shared repository intended to let anyone on the team ship to production—including customer success managers. [1] **Lesson:** broader AI-enabled execution requires shared access and documented operating context, not merely a new generation tool.

Zero retention calls for diagnosis before more features

One startup founder reported that 70 people tried a product and none stayed. They identified several competing explanations: weak need, positioning, target audience, onboarding, or distribution—and worried that adding an MCP integration could become feature work before the core problem was understood. [4]

Apply it: When follow-up emails fail, test the user flow directly. A community recommendation is to use a user-testing platform with a product URL and pointed flow questions, then review video feedback to locate the leak. [5]

Career Corner

Build AI PM fluency across four layers

A useful learning sequence for PMs is:

- **Foundations:** transformers, LLM training, and when to use prompting, RAG, or fine-tuning. [6]
- **Agents:** architectures, tools, and distribution. [6]
- **Production:** evaluation, observability, and testing—the gap between a demo and a working product. [6]
- **Strategy:** product sense, pricing, and operating models. [6]

Why it matters: this combines technical judgment with the commercial and operational decisions PMs must make around AI products.

Tools & Resources

- AI Foundations for PMs — a starting point for LLM basics and the prompt/RAG/fine-tuning decision. [6]
- AI Agents for PMs — covers agent architectures and tools. [6]
- AI Product Strategy — a resource focused on AI strategy, pricing, and operating-model questions. [6]

Sources

1. substack
2. X post by @scottbelsky
3. r/ProductMarketing post by u/linapisom
4. r/startups post by u/CreativeGPT
5. r/startups comment by u/No-Sink1088
6. substack