A DoorDash PM's shared Claude Code 'Team OS' turns AI from a personal tool into team self-service

PM Daily Digest

2026-11-27

A DoorDash PM's shared Claude Code 'Team OS' turns AI from a personal tool into team self-service

By PM Daily Digest • November 27, 2026

How one PM built a shared AI repo with a launch gate, practical handoff fixes for small teams with no PMs, launch tiering instead of nonstop launch emails, and why a 27–23 blind LLM vote doesn't settle anything.

A Claude Code "Team OS" that removes the PM as bottleneck

Aakash Gupta wrote up a shared Claude Code system built by DoorDash PM Hannah Stulberg (ex-Google APM, 1,500+ hours in Claude Code). He says it supports 20+ people [1]. Most PMs use AI for their own productivity. Stulberg's repo is shared instead: any function on her team can use it. Data scientists pull their own metrics, strategy leads get their own customer-call summaries, and engineers check analytics without filing a ticket [1].

The setup takes about 30 days, according to the post [1]. Its main parts: - A root CLAUDE.md under 500 tokens, plus nested CLAUDE.md files that serve only as navigation indexes. Each folder is owned by a function: the PM owns product, the data scientist owns analytics, and strategy owns customer calls [1]. - An analytics layer of metric definitions, SQL and dashboard links that loads progressively by depth. It also has shared skills (call summaries, PR creation, weekly synthesis) and verified playbooks for recurring analyses such as churn investigations [1]. - **A launch gate.** No feature ships until its metrics, queries, schemas, dashboards and playbooks are checked into the repo [1]. This is the

most reusable idea for PMs: it keeps the team's shared context complete as the product changes.

Seven people, no PMs: who owns the problem?

A developer at a 7-person B2B SaaS company described their setup. The CEO sets the roadmap. The head of product, who comes from design, ships code with vibe coding. Work arrives as one-line tickets, verbatim customer requests or live meetings, so expectations only show up at the demo [2]. Commenters split on what to do. One said everyone owns discovery at this size, so keep process light and break down silos [3]. Another said a single person must own the product [4].

The most practical replies agreed on the handoff: - Write half a page using the same template every time: who has the problem, what they do today, what "done" means, and what's out of scope. Then hold a 15-minute kickoff where the dev explains the work back, and show a rough version at the halfway point [5]. - A verbatim request is "the customer's guess at a solution." Every customer ticket needs three lines before it reaches a dev: what the customer was trying to do, what they did instead, and how you'll know it worked. If CS can't fill those in, the ticket goes back for another customer call [6]. - One founder described the opposite approach, building "upside down": engineers prototype first, product and design shape the result into flows that can ship, and the team relies on explicit decision rights for tie-breaks [7].

Stop launching every feature

A PMM at a regulated B2B financial-services firm says nearly every feature gets its own launch email, yet uptake stays low. Her plan for 2027 is to bundle about 90% of smaller releases into a monthly newsletter and send targeted messages to the customers who would actually use the rest [8]. Commenters suggested a launch-tiering scale, from Tier 1 (1–2 a year) to Tier 4 (bug fixes) [9]. One said only features that bring net-new revenue, open cross-sell or upsell, or reduce churn deserve a full launch motion [10]. The PMM said her company already has tiers, but stakeholders treat everything as Tier 1 [11]. To win over her manager, one reply suggested a digest-vs-email pilot. It would measure adoption, not just opens, for example the share of eligible accounts that run a first export within 14 days [12].

A blind LLM vote that proved nothing

Two cofounders ran 50 real customer questions through a cheap model and an expensive one, hid which was which, and had the team vote. The cheap model won 27–23, then 26–24 in a second batch [13]. Commenters called that result noise. They suggested 300 questions scored on things you can check, such as amounts, dates and whether the model asked for a missing order number, and counting retries before calling a model cheap [14]. "Cheap per call isnt the same

as cheap per ticket" [15]. Another commenter said to stay model-agnostic and build testing into the routine, since new models keep shipping [16].

Quick hits

- **Ghost jobs.** A free tracker that reads 877 US companies' careers sites counted 3,340 open PM roles. 480 are possible ghost postings (open 90+ days, or reposted and still open after 45). Only 103 roles (about 3%) are associate-level, and 28% are remote [17].
- **Onboarding.** About a third of roughly 540 Stackboard installs were uninstalled, mostly on day one. The solo developer removed the setup form, and uninstalls then went from about 3 a day to 10 and 11. He says this "could be noise," and he's asking users what made them want to close it in the first 30 seconds [18].
- **Making a familiar metric more human.** Patrick Collison built HumanFish, a chess eval bar that accounts for clock time, Elo, how obvious moves are, and how narrow the path ahead is. He says it's a "somewhat meaningful improvement" over the standard broadcast bar [19].
- **Forecasting.** Paul Graham: whether or not AIs "really" think, assume they do when predicting the future, "because they'll have all the effects they'd have if they really did" [20].

Sources

1. substack
2. r/ProductManagement post by u/Super_Bourguignon
3. r/ProductManagement comment by u/black_eyed_susan
4. r/ProductManagement comment by u/Bernhard-Welzel
5. r/ProductManagement comment by u/SivanKad
6. r/ProductManagement comment by u/TrulyN
7. r/ProductManagement comment by u/Standard-Feed-9260
8. r/ProductMarketing post by u/Fun-Technician-6015
9. r/ProductMarketing comment by u/No-Coat-6277
10. r/ProductMarketing comment by u/MeesterPositive
11. r/ProductMarketing comment by u/Fun-Technician-6015
12. r/ProductMarketing comment by u/EduFonsecaA
13. r/startups post by u/Powerful-Concept7920
14. r/startups comment by u/Fragrant_Paint3740
15. r/startups comment by u/nmohoni
16. r/startups comment by u/bieker
17. r/ProductManagementJobs post by u/gabethegeek
18. r/startups comment by u/intensityflow
19. X post by @patrickc
20. X post by @paulg