

A day of a product manager with AI

When to use what, how to connect it, and where your judgment stays.

a day of a product manager with ai

workflows, prompts and tools



n8n



Productboard



Jira



Amplitude



Dovetail



Linear



ChatGPT



Claude



Lovable



Replit

A practical companion to the post. The prompts stay beside the steps where you use them.

09:00	Start with one useful digest
10:45	Keep some of the saved time
12:00	You do the interview. AI cleans the notes.
15:00	Draft the PRD from what you know
End of day	Keep the learning and the follow-ups

10:00	Give the backlog evidence
11:30	Watch competitors without starting over
14:00	Use AI to choose the next check
16:30	Make one interaction visible

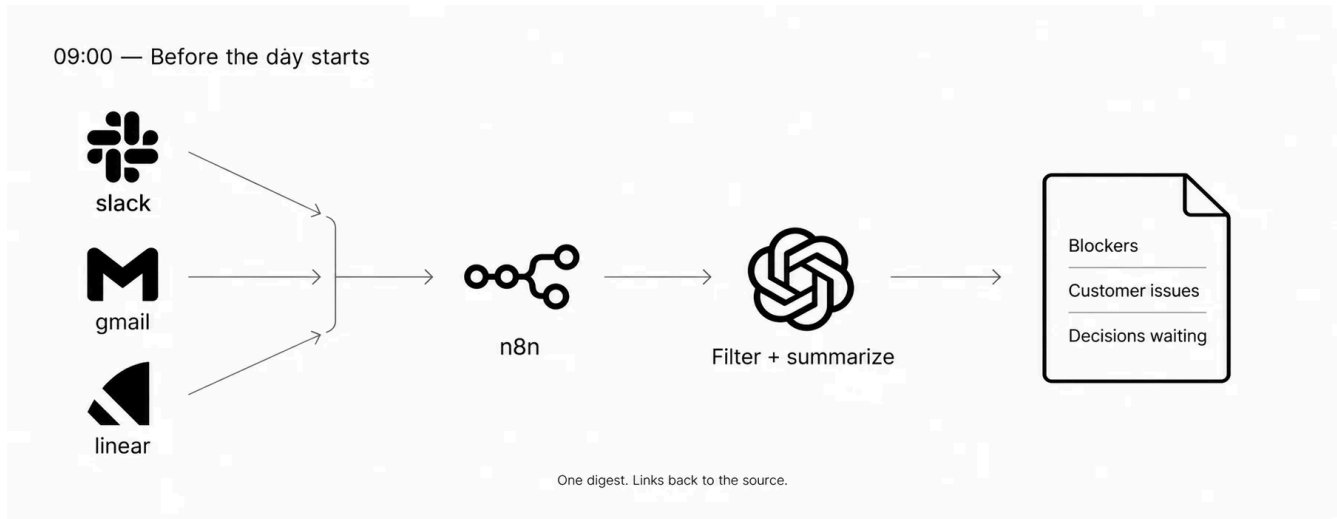
The whole day is a deliberately busy example, not a schedule you need to follow every day. Start with one workflow.

09:00

Start with one useful digest

Slack · Gmail · Linear or Jira · n8n · an LLM

When Before the working day, when messages and issues are scattered across several places.



Illustrative workflow. Set up the connections for your own workspace.

How to use it

1. Run a Schedule Trigger before work. Fetch recent messages and issues from the channels and projects you actually follow.
2. Normalize the records: source, ID, timestamp, text and original link. Deduplicate by source + ID; combine the records into one batch.
3. Send that batch to the model with the digest prompt. Deliver one short note to Slack or email. Keep source links beside each item.

Use this prompt for this step

Summarize only what can affect product decisions today. Ignore FYI noise. Show blockers, customer issues, decisions already made and questions waiting for me.

What you get

A short list of blockers, customer issues, decisions already made and questions waiting for you.

Your call Read the originals before making a consequential decision. Start with read-only collection; the digest does not send customer replies or change tickets.

[n8n Schedule Trigger ↗](#) · [Gmail node ↗](#) · [Slack node ↗](#) · [Linear node ↗](#)

10:00

Give the backlog evidence

Productboard or Jira Product Discovery · Linear or Jira · n8n

When When support tickets, sales notes and interview feedback need to be connected to roadmap items.

How to use it

1. Collect the feedback with its source link, customer and date. Load the candidate roadmap items and their descriptions.
2. Ask the model to propose a match, possible duplicates and missing information. Keep the supporting quote and source beside each suggestion.
3. Review uncertain matches before attaching them. Mark customer impact UNKNOWN when the evidence is missing; approve priority yourself.

What you get

Roadmap items with traceable evidence; delivery tasks with clearer context and visible gaps.

Your call A stakeholder repeating an idea is not proof of customer demand. Do not let a match score become an automatic roadmap priority.

[Productboard ↗](#) · [Jira Product Discovery ↗](#)

10:45

Keep some of the saved time

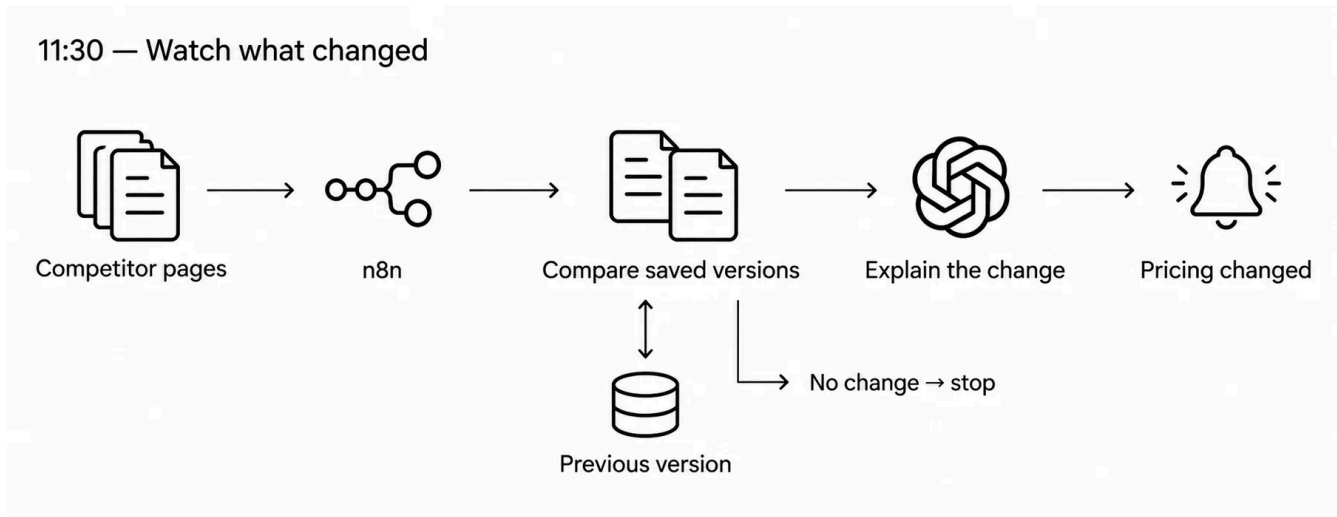
Take the break. Do not replace every saved minute with another dashboard.

11:30

Watch competitors without starting over

n8n · HTTP Request · Data Table or database · an LLM

When When a known set of competitor pricing pages, product pages and changelogs needs regular monitoring.



Illustrative workflow. Set up the connections for your own workspace.

How to use it

1. Keep a small list of permitted public URLs. Fetch on a schedule and extract the page content, excluding navigation and other noisy elements.
2. Store normalized text, a content hash and the fetch time for each URL. Compare with the previous version before paying for an AI call.
3. If there is a real change, send the old and new text to the model. Ask for what changed, supporting excerpts and the source URL; notify only on material changes.

What you get

A concrete update: pricing changed, enterprise launched, a feature disappeared, or onboarding changed.

Your call A timeout or blocked request is not a product change. Log fetch failures separately. Watch Reddit through a permitted feed or API, not an assumed free connector.

[HTTP Request node](#) · [n8n Data Tables](#)

12:00

You do the interview. AI cleans the notes.

Granola · Dovetail · Claude or ChatGPT, if needed

When After an interview, when the conversation needs to become reusable research evidence.

How to use it

1. Do the interview yourself. Use your approved note-taking setup and keep the transcript or original notes.
2. Run the extraction prompt against the actual notes and your current product beliefs. Check that quoted customer wording is real.
3. Export or copy the checked notes into Dovetail. Tag the problem, workaround and contradictory evidence; link back to the interview.

Use this prompt for this step

Extract what actually hurt the customer, what workaround they use now and anything they said that contradicts what we currently believe about the product. Keep their actual wording where useful.

What you get

Pain, current workarounds and evidence that challenges your assumptions, ready for the research repository.

Your call Granola → Dovetail describes a handoff. Use export, copy/import or an approved connector; do not assume a native integration exists.

[Granola integrations](#) ↗ · [Dovetail](#) ↗

14:00

Use AI to choose the next check

Amplitude or Mixpanel · an LLM with the actual analysis

When When an observed metric moves and you need to narrow the investigation.

How to use it

1. Verify the event definition, denominator, date range and tracking health first. A 42% → 31% activation drop here is an illustrative example.
2. Give the model the available breakdowns: source, device, geography, release and sample sizes. Ask which comparisons are worth checking first.
3. Run those comparisons in the analytics tool. Confirm or reject the hypotheses using the underlying data.

What you get

A short investigation plan and the segments to inspect, rather than an invented explanation.

Your call Analytics is the source of truth. A model suggestion or correlated segment does not prove what caused the drop.

[Amplitude Root Cause Analysis](#) ↗ · [Mixpanel](#) ↗

15:00

Draft the PRD from what you know

Claude or ChatGPT · roadmap context · customer evidence · engineering constraints

When When a feature is understood well enough to document, with source material available.

How to use it

1. Bring the problem, user evidence, analytics and engineering constraints into one context pack. Include the scope and explicit non-goals.
2. Draft from that pack only. Separate confirmed requirements, assumptions and open questions. Check each requirement against the source material.
3. Run the engineering-review prompt. Resolve ambiguity with engineering before calling the PRD ready.

Use this prompt for the PRD draft

Here is the customer problem, analytics and engineering constraints. Draft the PRD only from this. Do not invent requirements. Mark assumptions clearly.

Use this prompt for the engineering review

Read this as an engineer. What is unclear or can be interpreted in two different ways?

What you get

A focused draft plus a list of unresolved questions, ready for human review.

Your call The model must not turn missing information into confident requirements.

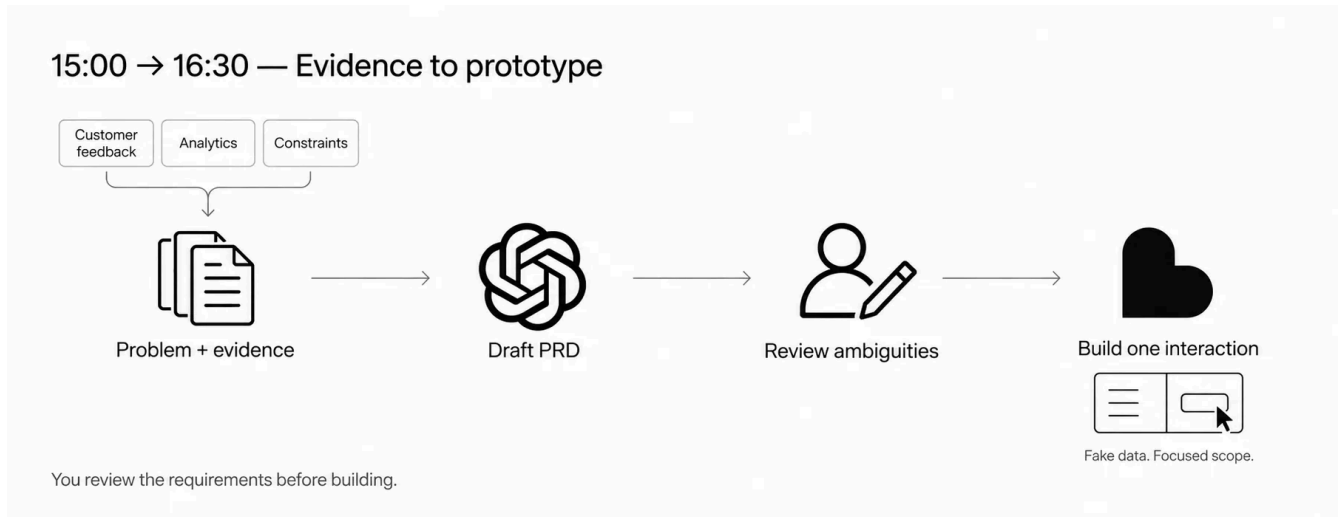
[Claude ↗](#) · [ChatGPT ↗](#)

16:30

Make one interaction visible

Lovable, Replit or v0 · fake data

When When a short clickable example will communicate the idea better than another long explanation.



Illustrative workflow. Set up the connections for your own workspace.

How to use it

1. Choose one workflow, one user and the moment you want to show. Pick one builder, not all three.
2. Replace [WORKFLOW] in the prompt with the interaction. Build with fake data; leave payments, production credentials and unrelated features out of the prototype.
3. Click through it yourself, then show it to design, engineering or a customer. Capture the questions it exposes and iterate on that interaction.

Use this prompt for this step

Build a simple clickable prototype for this workflow: [WORKFLOW]. Use fake data. Focus only on the main interaction. Do not add random extra features.

What you get

A shareable prototype that tests an interaction, not an accidental production app.

Your call Fake data and incomplete internals are fine for this demonstration. A prototype still needs engineering review before production use.

[Lovable for prototypes ↗](#) · [Lovable Plan mode ↗](#) · [Replit ↗](#) · [v0 ↗](#)

End of day

Keep the learning and the follow-ups

n8n · Notion or a private product log · calendar or task manager

When Before the day ends, while decisions, interviews and open questions are still fresh.

How to use it

1. Collect today's approved decisions, new tasks, interview notes and relevant metric changes. Keep the original links.
2. Run the reflection prompt. Save what got stronger, what got weaker and what is still undecided in a short dated note.
3. Review the follow-ups, confirm owners and dates, then put them into the task manager or calendar.

Use this prompt for this step

What did we actually learn today? What assumptions got stronger, what got weaker and what decisions should still NOT be made?

What you get

A useful product memory and a short list of follow-ups you have actually confirmed.

Your call Unapproved ideas stay ideas. Check owners and dates before sending reminders or creating external commitments.

[n8n Notion node ↗](#)

Start small. Keep your judgment.

Automate the movement, cleaning and first pass of information. Keep the interviews, priorities, decisions and important approvals with people.

[Read the full post ↗](#)