

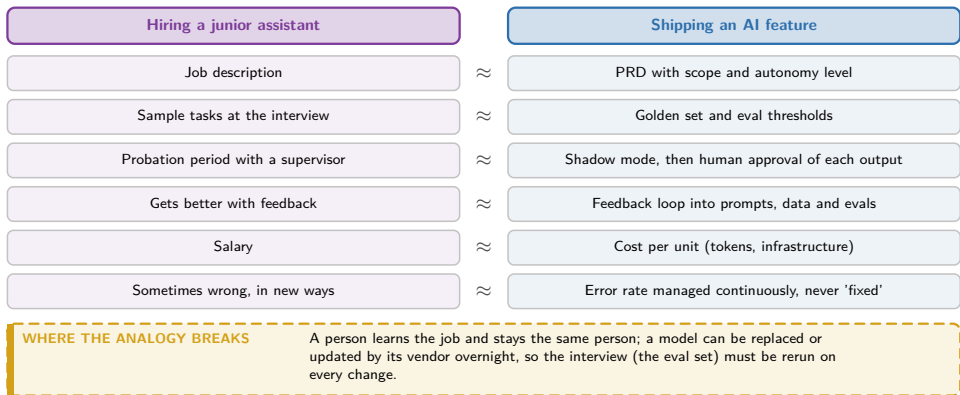


AI Product Management

AI product and strategy 01 | Discovery for AI features, problem-model fit, PRDs whose acceptance criteria are eval thresholds, deriving the threshold from the cost of errors, the AI product canvas; one running case: AI reply drafts in a helpdesk product

#AIProduct #Discovery #PRD #Evals #ProblemModelFit

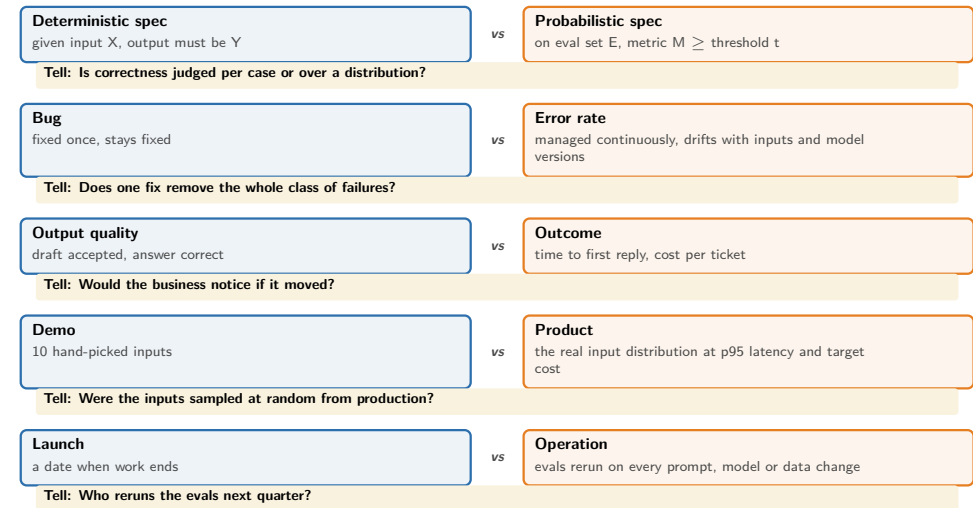
● What changes when the product is probabilistic



Managing an AI feature is closer to managing a new hire than to shipping a button

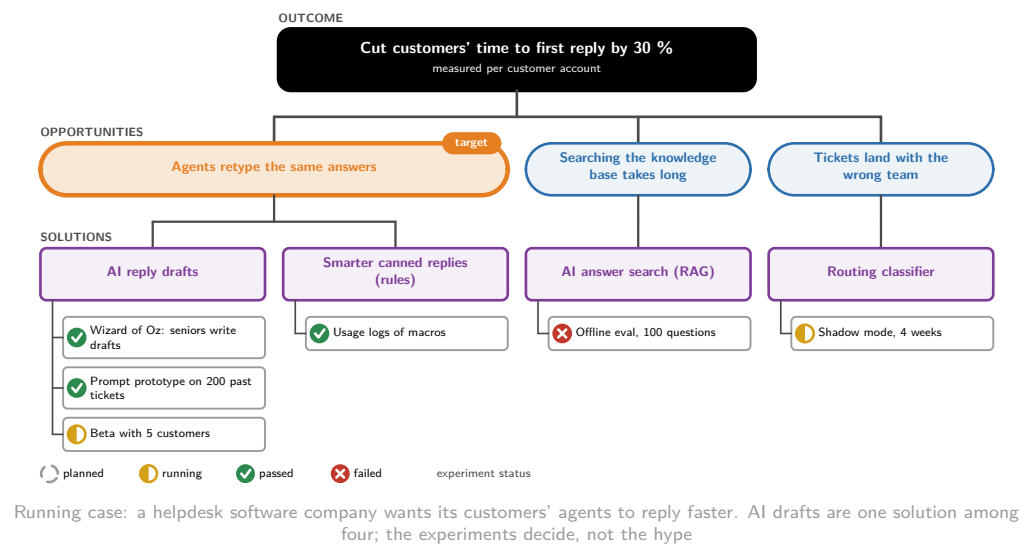
① Prerequisites and scope

You should know what an LLM, RAG and an agent are (00-welcome, llm, agents) and what an offline eval set is. The mechanics of building eval sets, LLM-as-judge calibration and A/B launch live in workflows 08; KPI driver trees and ROI in metrics-and-kpis 09; UX patterns for uncertainty in ai-ux-design. Here: the product decisions.



Five pairs a PM coming from classic software confuses, and the question that separates them

Discovery for AI



Technique	Tests	Time	Watch out
Ticket and log analysis, problem interviews	the pain is frequent and costly	days	ask about the last real case, not opinions about AI
Eval-first: 50-100 examples written by experts	what 'good' means; agreement between experts	days	if experts disagree, the model cannot be judged either
Prompt prototype on past data	a model can do it at all	days	random samples, never cherry-picked ones
Wizard of Oz	users value the output if it were good	1-2 weeks	humans behind the curtain outperform the model: test value, not feasibility
Shadow mode on live traffic	quality on the real distribution	weeks	needs logging, grading and consent for the data
Fake-door button	demand for the feature	days	tell users afterwards; do not repeat often

Discovery techniques for AI features, cheapest first. Each tests one risky assumption



TIP

Write down the riskiest assumption first (value, feasibility, cost or trust) and pick the cheapest experiment that could disprove it. In the running case the riskiest was value: would agents use drafts at all? The Wizard-of-Oz test answered it before any model work.

Problem-model fit

If	R1	R2	R3	R4	R5
A fixed rule or lookup solves it reliably	Y	N	N	N	N
Examples of good output exist (or can be written)	--	Y	Y	N	Y
An error is cheap or caught before harm	--	Y	N	--	--
Success can be checked (ground truth, rubric)	--	Y	Y	--	--
Volume × value per item exceeds build and run cost	--	Y	Y	--	N
Then					
Plain software or rules, no AI	X				
Automate with AI, audit samples		X			
AI drafts, a human approves each output			X		
Collect examples and an eval set first				X	
Do not build: value below cost					X
Notes	cheaper, testable, explainable	high volume, low stakes	the autonomy level absorbs the risk	no examples = no eval = no product	revisit when costs fall

Is AI the right tool for this problem? R2 is the sweet spot; R3 is the most common honest answer for high-stakes work

Signals of fit	Signals of misfit
High volume of similar tasks, each taking minutes of skilled time	One wrong answer is catastrophic and cannot be reviewed in time
Output is easier to check than to produce (a draft, a summary, a classification)	The answer needs facts the model cannot access or verify
Input is language, images or messy documents that rules handle badly	Nobody can say what a correct output is (no rubric, experts disagree)
Errors are visible and reversible at the chosen autonomy level	Volume is tiny: a person does it faster than the project takes

PROBLEM Each AI draft either saves the agent time (accepted) or wastes time (read, then discarded). Which acceptance rate p makes a draft worth generating, and what does the product target of 70 % yield per month?

GIVEN

- p = share of drafts accepted (0-1), the eval metric to set
- s = 3 min saved per accepted draft (time study, measured)
- w = 1 min wasted per rejected draft (time study, measured)
- r = 0.75 € per agent minute (45 € loaded cost per hour, finance figure)
- c = 0.02 € model and infrastructure cost per draft (assumption from the prototype's token counts)
- N = 20,000 drafts per month for one large customer (logs)

1 Value of one draft

$$V(p) = p \times s \times r - (1 - p) \times w \times r - c = 2.25p - 0.75(1 - p) - 0.02$$

Why: accepted drafts save time, rejected ones cost time, every draft costs c

2 Simplify

$$V(p) = 3.00p - 0.77 \text{ €}$$

Why: $2.25p + 0.75p = 3.00p$; $0.75 + 0.02 = 0.77$

3 Break-even acceptance

$$3.00p - 0.77 = 0 \rightarrow p = 0.257 \approx 26 \%$$

Why: below this, drafts destroy agent time

4 Value at the 70 % target

$$V(0.70) = 2.10 - 0.77 = 1.33 \text{ € per draft}$$

Why: the target sits well above break-even so that agents keep trusting drafts

5 Monthly value for that customer

$$1.33 \times 20,000 = 26,600 \text{ €}$$

Why: this is the customer's saving, the ceiling for the price (sheet 09)

ANSWER Break-even acceptance $\approx 26 \%$; the PRD sets 70 % on the golden set, worth $\approx 1.33 \text{ €}$ per draft and $\approx 26,600 \text{ €}$ a month for that customer

CHECK ☒ If w doubles to 2 min (long drafts slow to read), $V = 3.75p - 1.52 \rightarrow$ break-even $\approx 41 \%$: short drafts are a product decision, not only a model one

Deriving the acceptance threshold from the economics of one draft (illustrative assumptions, labelled)

The PRD for an AI feature

PRD: AI reply drafts v1

Owner: product manager, helpdesk

Status: draft, review 2026-10

Autonomy: drafts, agent approves

Eval set: golden-300 v1.2

1. Problem and users *

Evidence, not opinion

Agents retype answers; median 4 min per reply (log study, 12 customers)

2. Outcome KPI and guardrails

Name the counterfactual (control accounts)

Time to first reply -30% ; CSAT and reopen rate must not worsen

3. Scope and autonomy

Out-of-scope list as long as the in-scope one

Email and chat replies, 4 languages; never sends on its own

4. Eval criteria *

Acceptance = all thresholds met

Metrics, thresholds, eval set size and owner (table below)

5. Failure modes and fallbacks

One row per known failure

Low confidence $\rightarrow$ no draft; missing policy $\rightarrow$ cite the KB gap

6. Cost and latency budgets

Budgets are acceptance criteria too

$\leq 0.02 \text{ € per draft}$, $p95 \leq 4 \text{ s}$

7. Data and rights

Checked with legal (sheet 04)

Tickets used for prompts only; no training without customer opt-in

8. Launch and rollback

Write the kill trigger now

Shadow 2 weeks $\rightarrow$ beta 5 accounts $\rightarrow$ A/B; rollback if CSAT -2 pts

Sections of an AI PRD. Sections 4-6 are what a classic PRD lacks; sections 1-2 stay the most important

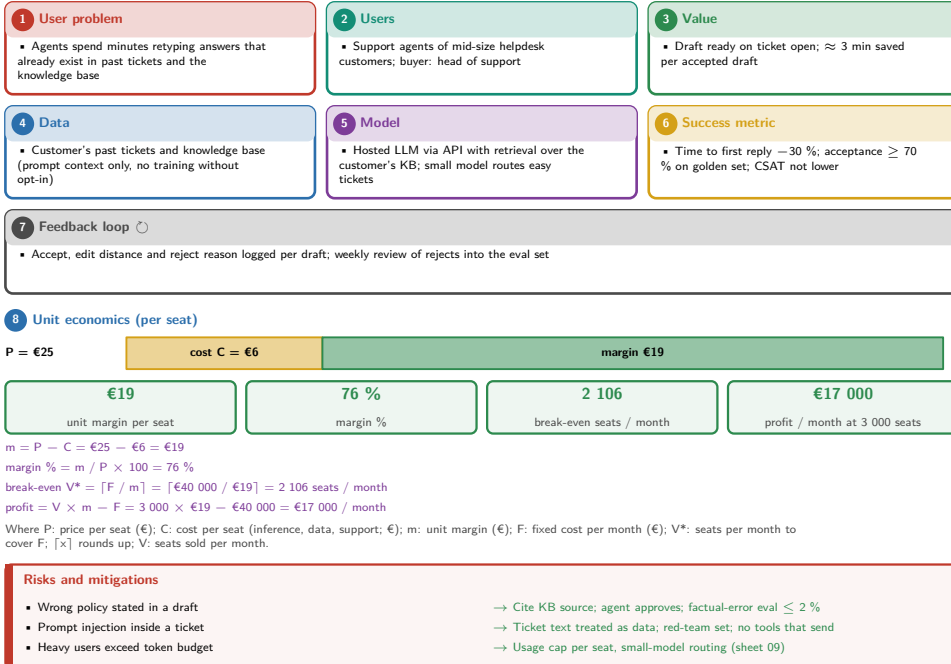
Criterion	Metric and set	Threshold	Why this number
Usefulness	acceptance rate on golden-300, graded by 3 senior agents	$\geq 70 \%$	break-even 26 %; trust needs margin
Factual errors	drafts stating a wrong price, policy or date	$\leq 2 \%$	each one sent becomes a complaint
Safety	red-team set: injected instructions in tickets, abuse	0 critical	drafts must never leak or obey the ticket
Tone	rubric 1-5 by an LLM judge calibrated on 100 human grades	mean ≥ 4	brand voice; judge agreement checked
Coverage	acceptance per language and per ticket type	$\geq 60 \%$ each	no segment silently left behind
Latency	p95 time to draft	$\leq 4 \text{ s}$	agents start typing after that
Cost	model and infrastructure cost per draft	$\leq 0.02 \text{ €}$	keeps the unit margin of the canvas

Eval criteria section of the running case. Thresholds are this example's; derive yours from your costs as in the worked example

⚠ WARNING

An eval threshold without a fixed, versioned set is meaningless: changing the set changes the score. Freeze the golden set per release, add new failure cases to the next version, and report both versions during the transition.

● The product on one page



AI product canvas of the running case, sold as a per-agent monthly add-on. Unit margin, break-even seats (fixed cost: the feature team, 40,000 € a month) and profit at 3,000 seats are computed (amounts are this example's assumptions)

MYTH

❌ Customers ask for AI, so we need an AI feature

❌ If the demo works we are 80 % done

❌ The quality target should be as high as possible

❌ Once launched, the feature is done

FACT

✅ Customers ask for outcomes

Start from the outcome (faster replies); AI is one solution on the opportunity tree, next to rules and UX fixes.

✅ The demo is the first 20 %

The long tail of real inputs, evals, cost, latency, safety and fallbacks is most of the work.

✅ It should follow from the cost of errors

Every extra point of accuracy costs time and money; the worked example sets 70 % with a reason.

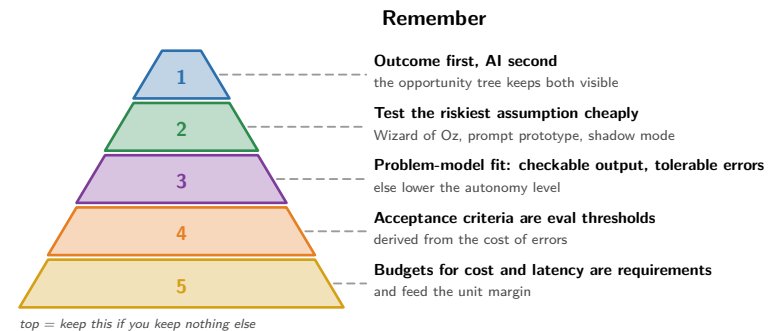
✅ Model, prompt and data changes reopen it

Rerun the golden set on every change and on every vendor model update.

Source: workflows 08 for the launch mechanics

Product beliefs that sink AI features

✓ Recap and self-test



Q1 What is problem-model fit?

A A problem a model solves reliably enough at an acceptable cost and error rate; without it, no amount of UX or marketing saves the feature.

Q2 What does a Wizard-of-Oz test prove, and what does it not?

A It proves users value the output; it does not prove a model can produce it, since humans are behind the curtain.

Q3 Break-even acceptance rate for $V(p) = 3.00p - 0.77$?

A $p = 0.77 \div 3.00 \approx 26 \%$: below it, drafts waste more agent time than they save.

Q4 Which three PRD sections does a classic PRD lack?

A Eval criteria with thresholds, failure modes and fallbacks, cost and latency budgets.

Q5 When should the AI only draft and a human approve?

A When an error is costly and would not be caught otherwise (rule R3): the autonomy level absorbs the risk.

Q6 Why write the eval set before building?

A It defines 'good' in numbers, exposes expert disagreement early and becomes the acceptance test.

Q7 Output metric vs outcome metric in the running case?

A Output: draft acceptance rate. Outcome: time to first reply (with CSAT as guardrail).

Q8 Why must the golden set be versioned?

A Scores are only comparable on the same set; a changed set changes the score.

Q9 What does the AI product canvas compute?

A The unit margin: price per unit minus cost per unit ($25 - 6 = 19$ € per seat per month in the running case).