

PRODUCT MANAGEMENT FRAMEWORK

The RICE Prioritization Framework

$$Reach \times Impact \times Confidence \div Effort$$

A Practical Guide to Data-Informed Roadmap Prioritization

Prepared by	Harsh - Product Manager
Use case	Roadmap prioritization & stakeholder alignment
Category	Product Strategy - Prioritization Frameworks

Table of Contents

1. What Is the RICE Framework?.....	3
2. The Formula	3
3. The Four Components.....	4
4. Worked Example	5
5. How to Apply RICE - Step by Step	6
6. When to Use RICE - and Its Limitations.....	6
7. Reusable Scoring Template	7

1. What Is the RICE Framework?

RICE is a quantitative prioritization model that scores competing product initiatives features, experiments, bug fixes, or strategic bets on a common scale so that Product, Engineering, and business stakeholders can debate priorities using shared evidence rather than opinion or seniority. It was popularized by the growth team at Intercom and has since become a standard tool in B2B SaaS, insurtech, and healthtech product organizations where roadmap capacity is scarce and every team has a compelling case for why their initiative should go first.

The model breaks a prioritization decision into four independently estimable components Reach, Impact, Confidence, and Effort and combines them into a single comparable score. The output is not a final decision; it is a structured input that makes trade-offs visible and defensible in a roadmap review.

Why PMs use it

- Forces explicit assumptions instead of gut-feel ranking.
- Creates a shared vocabulary across Product, Engineering, Design, and Leadership.
- Surfaces low-effort, high-impact “quick wins” that are easy to overlook.
- Produces an auditable trail for why an initiative was sequenced the way it was.

2. The Formula

$$\text{RICE Score} = (\text{Reach} \times \text{Impact} \times \text{Confidence}) \div \text{Effort}$$

Each component is estimated independently, then multiplied and divided as shown above. Reach, Impact, and Confidence sit in the numerator because they represent the upside of doing the work, Effort sits in the denominator because it represents the cost of doing it. A higher score means more expected value per unit of team capacity.

3. The Four Components

3.1 Reach

How many people or accounts will this initiative touch within a defined time period? Reach should always be expressed as a concrete number over a fixed window (e.g., “users per quarter,” “new accounts per month”), pulled from product analytics, CRM data, or a defensible estimate - never a vague qualitative guess.

Typical scoring inputs:

- Monthly or quarterly active users affected
- New customers or accounts touched per launch cycle
- % of the customer base in the affected segment × total base size

3.2 Impact

How much will this move the needle for each person it reaches on conversion, retention, satisfaction, or a north-star metric? Because impact is inherently harder to measure precisely than reach, most teams use a discrete multiple-choice scale rather than a raw number.

Score	Impact Level	Description
3	Massive	Transforms the core experience or unlocks a new revenue line
2	High	Clear, material improvement to a key metric
1	Medium	Noticeable but incremental improvement
0.5	Low	Minor improvement, easy to miss in the data
0.25	Minimal	Marginal effect, mostly qualitative

3.3 Confidence

How sure is the team about the Reach and Impact estimates? Confidence exists to prevent optimistic guessing from masquerading as data. A team with a strong analogous case study or a completed A/B test should score higher than a team relying on intuition.

Score	Confidence Level	Description
100%	High	Backed by data experiment results, funnel analytics, or direct customer research
80%	Medium	Reasonable qualitative evidence customer interviews, support ticket trends
50%	Low	Largely a hypothesis; proceed only if effort is low

3.4 Effort

How many “person-months” of combined Product, Design, and Engineering time will this consume, start to finish? Effort is estimated in whole or half units (e.g., 0.5, 1, 2, 3) at a rough order-of-magnitude level not a detailed sprint estimate because RICE is a prioritization pass, not a delivery plan.

- Include Design, Engineering, QA, and Data/Analytics time, not just Engineering.
- Use the same unit (person-months is standard) across every initiative you compare.
- Re-estimate before execution RICE effort is directional, not a commitment.

4. Worked Example

The table below applies RICE to five candidate initiatives for a B2B healthtech/insurtech platform team choosing what to build next quarter. Reach is in thousands of users per quarter; Impact uses the 3 / 2 / 1 / 0.5 / 0.25 scale, Confidence is a percentage, Effort is in person-months.

Initiative	Reach (users/qtr)	Impact	Confidence	Effort (person-mo)	RICE Score
AI-powered claims triage assistant	8,000	3	80%	4	4,800
Self-service benefits enrollment wizard	12,000	2	80%	2	9,600
Provider network search redesign	5,000	1	100%	1	5,000
Predictive churn-risk dashboard for CSMs	1,500	2	50%	3	500
Dark-mode UI	6,000	0.5	100%	0.5	6,000

Reading the result: the self-service enrollment wizard scores highest (9,600) not because it has the biggest single impact, but because it pairs strong reach and confidence with comparatively low effort exactly the kind of trade-off RICE is designed to surface. The predictive churn dashboard, despite being valuable in principle, scores lowest once its narrow reach and unproven assumptions are made explicit; it may still be worth pursuing, but as a validated experiment rather than a full build.

5. How to Apply RICE - Step by Step

1. List every candidate initiative from the backlog, discovery pipeline, and stakeholder requests in a single sheet.
2. Pull a Reach number for each from analytics or CRM data; avoid estimating from memory.
3. Score Impact using the 3 / 2 / 1 / 0.5 / 0.25 scale, calibrated against a past launch with known results.
4. Assign Confidence honestly if the team is arguing about the number, that disagreement is itself a signal to lower it.
5. Estimate Effort in person-months across every discipline involved, not just engineering.
6. Calculate the RICE score and sort descending.
7. Sanity-check the ranking against strategic constraints compliance deadlines, platform dependencies, executive commitments that RICE does not capture.
8. Present the scored backlog to stakeholders as a starting point for discussion, not a final verdict.

6. When to Use RICE and Its Limitations

Best suited for

- Comparing a broad backlog of features, experiments, or fixes competing for the same team capacity
- Quarterly or sprint-cycle roadmap planning where multiple stakeholders each advocate for different work
- Creating a defensible, revisitable record of why a sequence of initiatives was chosen

Known limitations

- Scores can feel falsely precise multiplying estimates does not remove the uncertainty in each input
- Doesn't natively account for technical dependencies, regulatory deadlines, or platform debt
- Vulnerable to "score gaming" if teams inflate Reach or Confidence to win prioritization debates
- Works best as one input alongside strategic judgment, not as an autopilot decision engine

PM Practice Tip

Recalibrate the Impact and Confidence scales every 1-2 quarters against actual launch outcomes. This keeps the framework honest and prevents score inflation from creeping in over time.

7. Reusable Scoring Template

Copy the structure below into your own roadmap tracker. Keep the unit definitions (Reach window, Impact scale, Effort unit) visible at the top of the sheet so every contributor scores consistently.

Initiative	Reach	Impact	Confidence	Effort	RICE Score	Owner