

The Product Manager's Complete Guide To

PRI RITIZATION

How to Choose the Right Features
and Ship Better Products

“Without clear priorities, the development team lacks direction, and there is only a slim chance of creating a successful product.”

Roman Pichler

Product Management Consultant



Table of Contents

Introduction	4
But First... Product Strategy	9
Prioritization Methodologies Proven to Work	11
How to Find the Right Framework for Your Team	28
4 Best Practices for Rolling Out Your Framework	31
Overcoming Common Challenges to Implementing Your Framework	36
How a Prioritization Framework Improves Your Roadmap	39
Yes, This Works for Real Companies	42
Conclusion: It's All About Making an Impact	45

Introduction

Why You Need a Prioritization Framework

Although every product team's circumstances are unique, we can make one assumption about you with a high degree of confidence if you're a product manager. At any given time, your product backlog is filled with more ideas, suggestions, requests, and demands than you can handle.

Because your resources will always be limited, you'll never be able to accomplish everything you want for your product in the timeframe you'd like. Nor will you be able to satisfy everyone else's requests for new features, fixes, and enhancements in the timeframe they'd like.

This brings us to the critical question. If your team can work on only a few initiatives simultaneously, how will you decide which things to prioritize? You can't choose items at random. And you can't just prioritize the demands of the strongest voice on your team or your most vocal customer. You need to prioritize the projects that support your product vision and strategy. But how do you do that?

Focusing on What Matters

If you find yourself regularly fending off requests for new product functionality or staring at your backlog trying to decide what to focus on next, **we wrote this guide for you.**



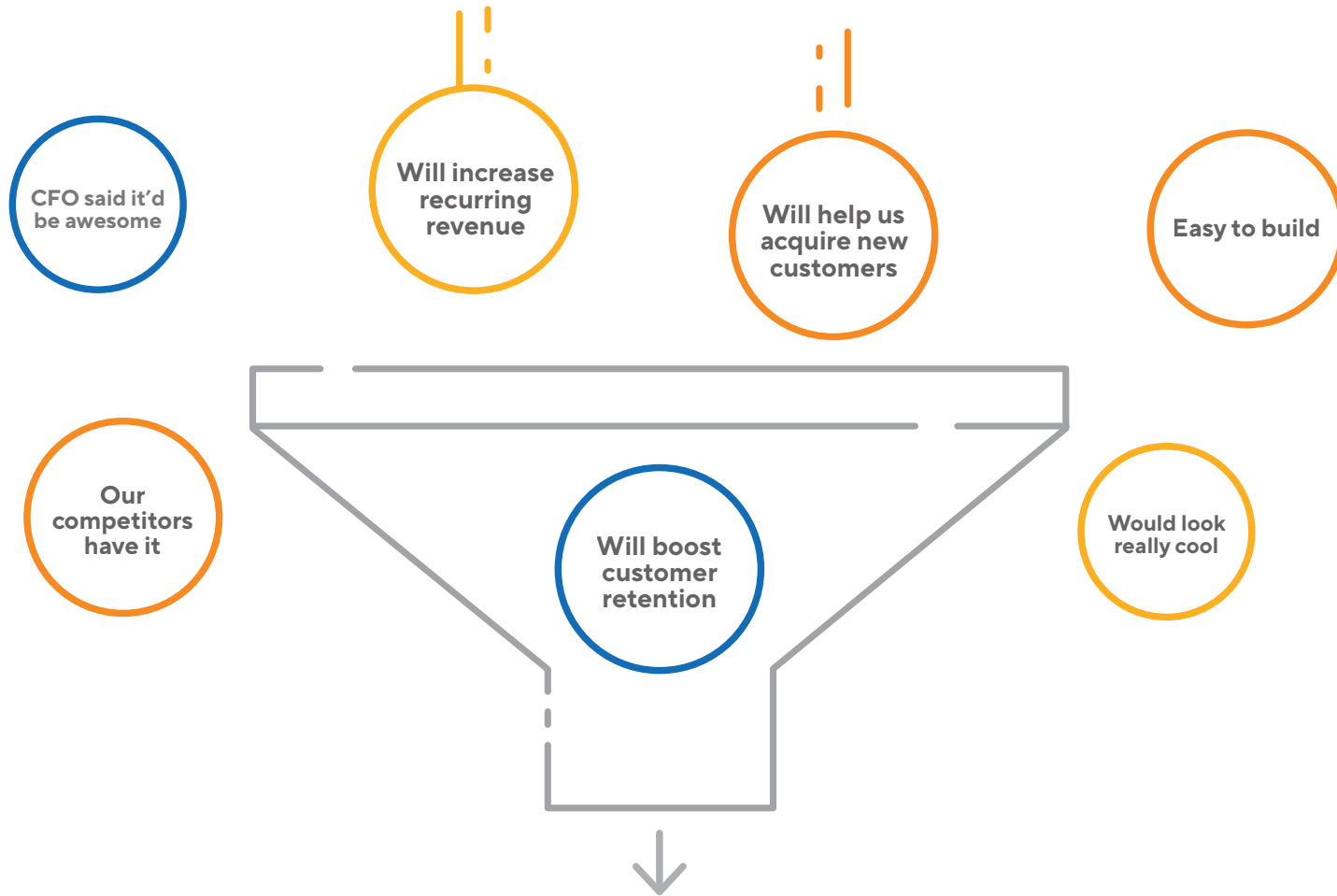
You need to make sure your team is always working on the most strategically beneficial initiatives. These items will have the most significant impact on your product, your users, and your company's bottom line. But to identify and prioritize those items, you need a process and a framework.

That's what we hope to do with this guide: help you find the right prioritization framework for your team and show you how to use it to create more successful products.

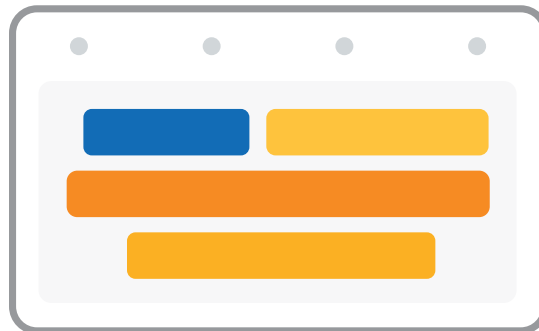
How This Guide Can Help You

Even if you have no experience with prioritization frameworks, you don't need to worry. We're going to take it from the very beginning and walk you through:

- Some of the most effective, proven prioritization frameworks for product teams
- How to select the right framework for your team depending on your goals and resources
- Best practices for getting buy-in on your prioritization framework
- Overcoming the common challenges of implementing your framework
- How to use prioritization to build an impactful product roadmap
- A case study: how one product team leveraged the right prioritization to improve their product strategy



Add to product roadmap



Did You Know

Of the 2,200 product professionals, we surveyed for our 2021 State of Product Management Report, the skill that respondents said their product peers lacked most was the ability to prioritize.



“In many cases, the struggle to prioritize features is just a symptom of something bigger. The real problem is usually a lack of strategic vision or direction.”

Suzanne Abate

CEO at The Development Factory



But First... Product Strategy

Steps to a Product Strategy

1. Develop a high-level vision—the “why”—for creating your product.
2. Talk with your ideal users, your market, to learn what challenges they need to be solved.
3. Define your product’s strategic objectives.
4. Translate these objectives into themes for your product roadmap.
5. Take one more look at your product vision to make sure your plan supports it.

For a deeper dive, read our [Ultimate Guide to Product Strategy](#).

Now, assuming your team has a clear product vision and strategy, let’s look at some of the best prioritization frameworks.



“In my opinion, it doesn't really matter what framework you use for prioritization, so long as you use a framework. What I've discovered in working with teams is that nothing deflates morale more than a seemingly arbitrary process of prioritization.”

Suzanne Abate

CEO at The Development Factory



Prioritization Methodologies Proven to Work

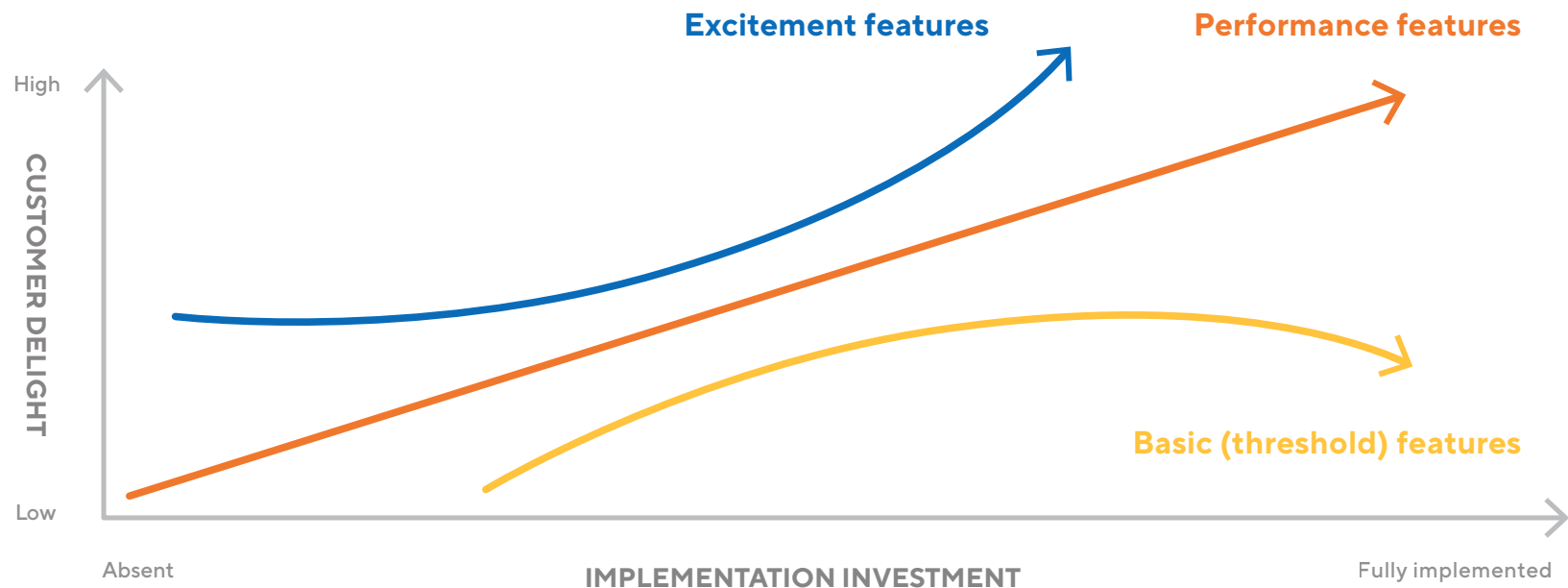
There are hundreds of product prioritization frameworks to choose from. Here, we'll review what we've found to be the best ones for different types of teams, businesses, and product strategies.



Frameworks for Customer-Centric Product Strategies

The Kano Model

Using the Kano model (pronounced “Kah-no”), a product team analyzes and ranks feature ideas according to a single consideration: how much each feature is likely to delight customers, measured against its cost to build. This is one of the most widely used product prioritization frameworks, and it is especially popular among customer-centric companies.



Design Thinking

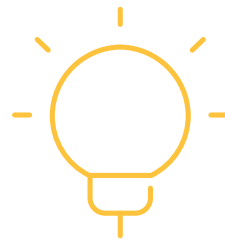
IDEO, the company often credited with creating the design thinking methodology, describes it as a “human-centered approach to building products.”

This methodology turns the standard product-development process on its head. Where most companies start by thinking about the product they hope to build, design thinking asks product teams to focus first on their users. Thus, the first step in the design thinking process is to empathize with users, understand their needs, wants, and challenges. Only then should the team begin generating ideas to solve those challenges.

Empathize



Ideate



Test



Define



Prototype

Jobs to Be Done (JTBD)

Jobs-to-be-done (JTBD) is another customer-centric framework. This methodology helps reorient teams from thinking about the product to focusing on the person who might buy it. JTBD asks the product team to try to understand what a customer is trying to accomplish when they “hire” a product to help them.





Frameworks for Smaller Companies

AARRR (Pirate Metrics Framework)

AARRR, also called the pirate metrics framework, can help startups tune out vanity metrics, such as social media attention, and prioritize their work around five key metrics that affect the company's bottom line. The name is an acronym for those metrics:

- Acquisition (or Awareness)
- Activation
- Retention
- Referral
- Revenue

Category	User Action	Conversation %	Estimated Value
Acquisition	Visits app	100%	\$.01
Acquisition	Stays 1 min+	75%	\$.10
Activation	Subscribes to newsletter	25%	\$.25
Activation	Signs up for free trial	5%	\$1.00
Retention	Opens email, clicks through	8%	\$2.00
Retention	Visits app 3 times in 30 days	3%	\$5.00
Referral	Sends 1+ users to visit site	2%	\$3.00
Referral	Refers 1+ users who activate	1%	\$10.00
Revenue	Generates minimum revenue	2%	\$5.00

The AARRR approach can help companies of all types prioritize their product development around what matters most to users and their bottom line at all stages of the business lifecycle. But the framework works particularly well for startups because these companies are typically cash-strapped and only have a limited time window to prove they have a viable product.



Frameworks for Larger Companies

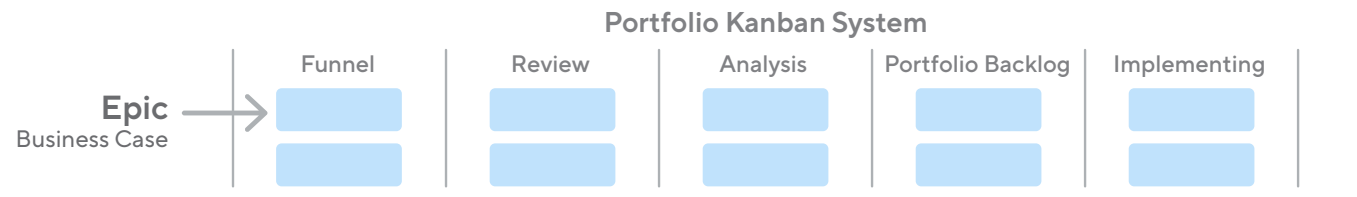
Scaled Agile Framework (SAFe)

The Scaled Agile Framework (SAFe) can help manage some of the common challenges large organizations face when practicing agile development. It takes a top-down approach to decision-making focused on three key aspects of development: team, program, and product portfolio.

Rather than a single methodology, SAFe acts as a broad knowledge base of proven best practices that teams have used to deliver successful software products. Among the framework's key attributes, it:

- Helps cross-functional teams collaborate more effectively
- Helps organizations improve transparency among teams
- Aligns all aspects of a project to the broader business goals

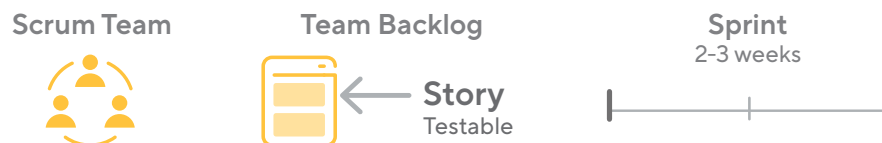
Portfolio Level



Program Backlog



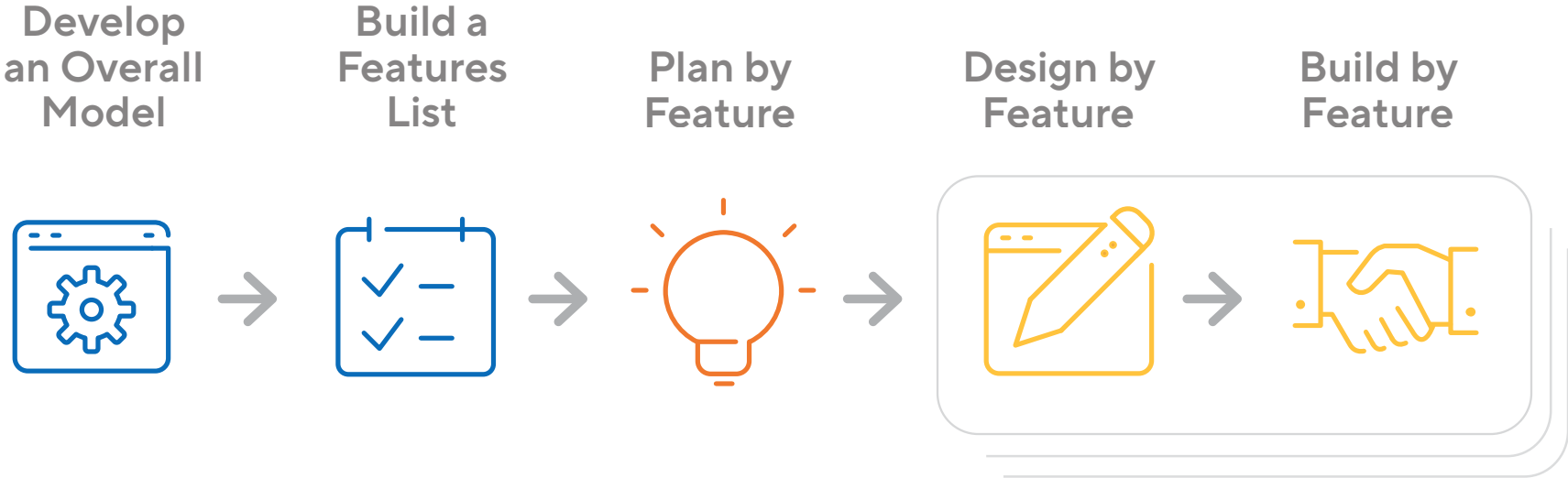
Team Level



Feature-Driven Development (FDD)

Feature-driven development (FDD) is an agile framework that organizes software development around making progress on features. However, it's important to point out that in the FDD context, a feature will, in many cases, resemble a user story in the agile methodology.

Because the FDD model takes a top-down approach to decision-making, it mainly serves large enterprises with a clear personnel chain of command.

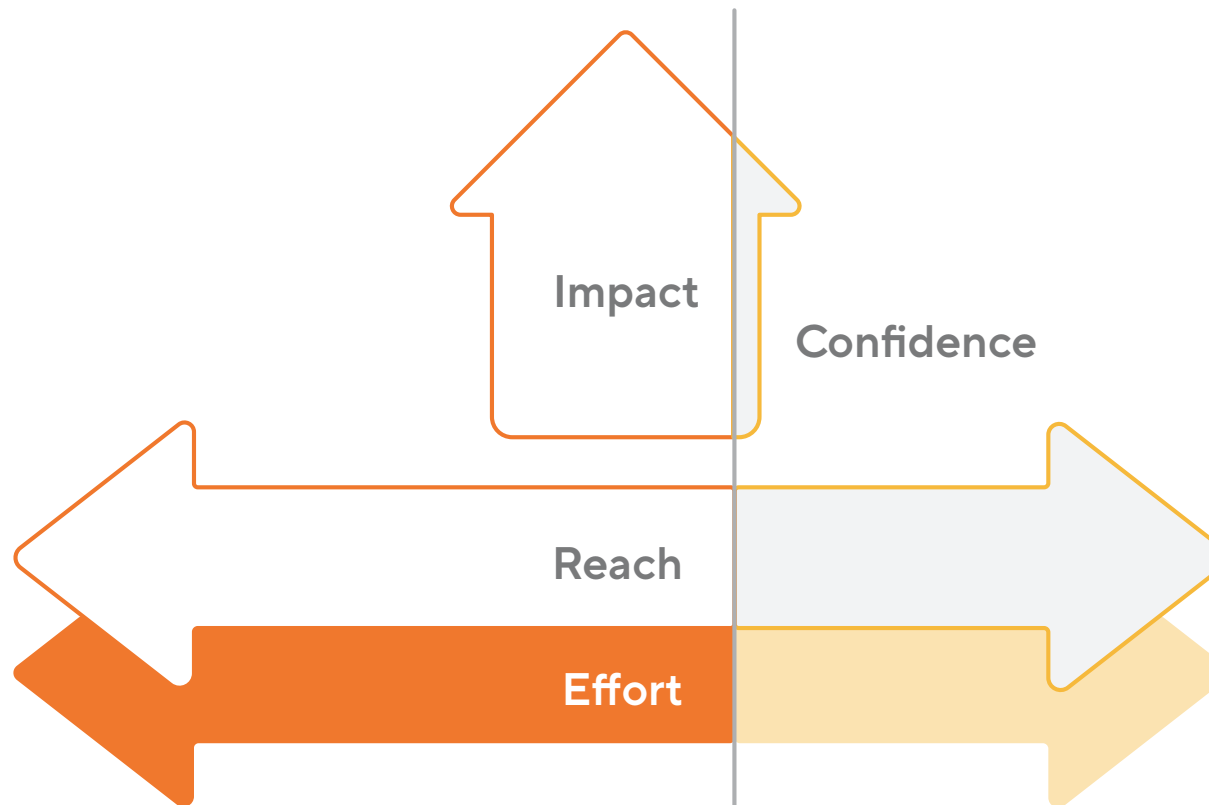




Frameworks for Weighing Value Against Resource Levels

RICE Scoring

The RICE scoring framework takes a quantitative approach helping product teams prioritize their roadmaps by allowing them to score items according to four factors: reach, impact, confidence, and effort. When the team combines its total scores for all initiatives across all four criteria, each initiative will have a single score to weigh its relative value to the product and the company.

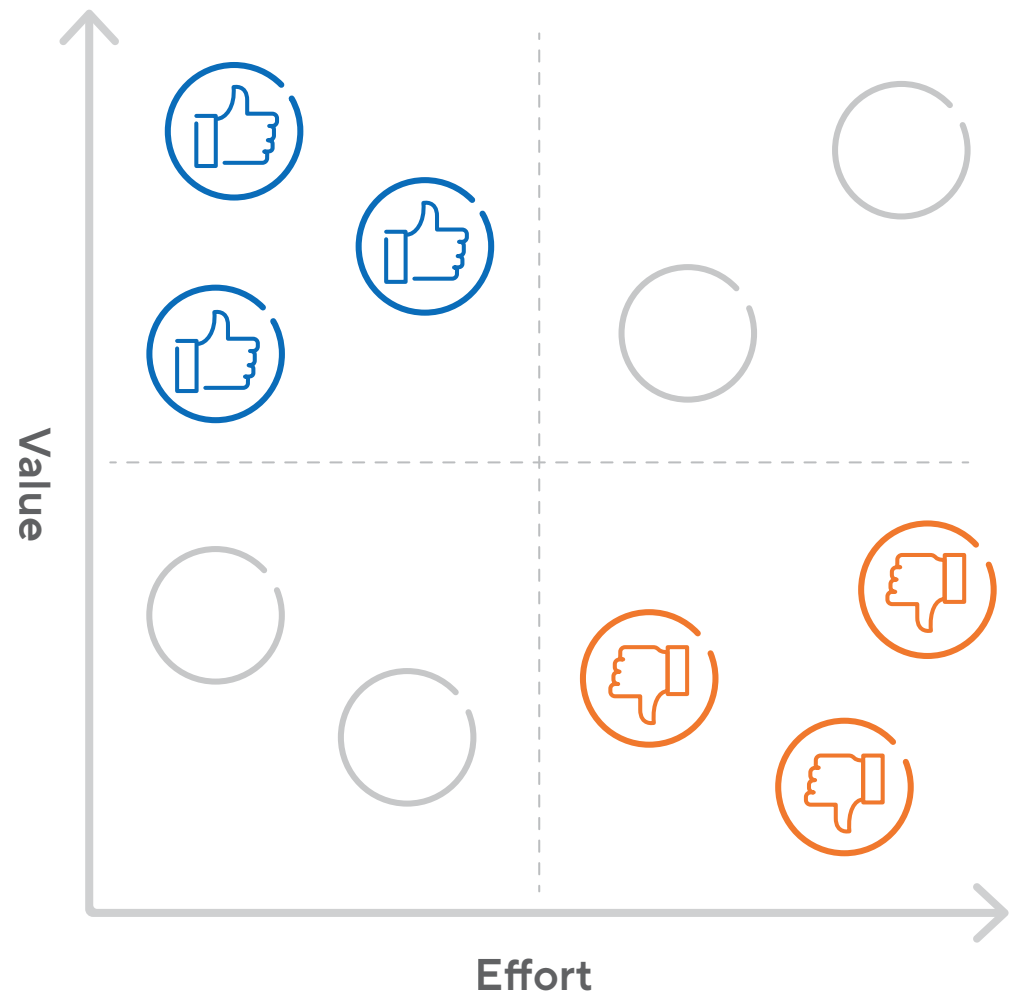


Value vs. Complexity

The value vs. complexity approach allows a product team to evaluate each initiative according to how much value the initiative will bring and how difficult it will be to implement. After scoring an initiative on those two criteria, the team then plots it on a quadrant and prioritizes it on the roadmap accordingly.

The matrix is simple. The initiatives that deliver the highest value and require the least effort will be the low-hanging fruits for your roadmap. Meanwhile, you should probably avoid the initiatives that offer the lowest value and require the highest effort. Initiatives that fall into the other two quadrants may merit further discussion with your team and stakeholders.

Value vs. complexity is a popular method among product teams looking for an objective way to allocate time and finite development resources to initiatives based on their perceived or potential benefit.



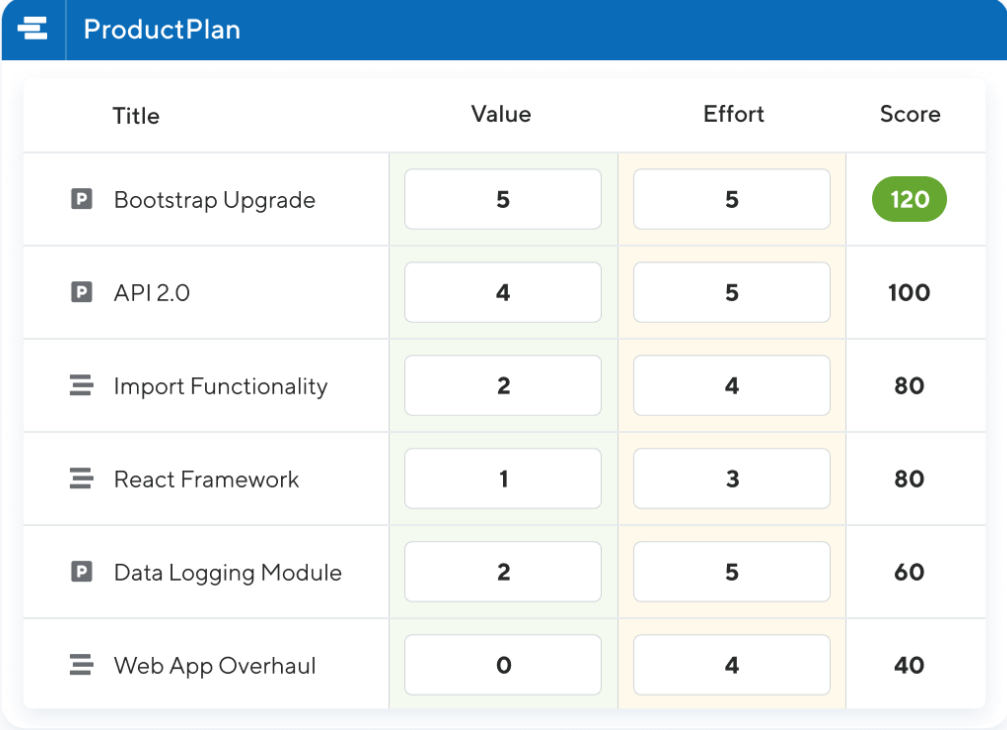
Weighted Scoring







Weighted scoring prioritization (sometimes called the Benefit vs. Cost approach) uses numerical scoring to rank initiatives according to their benefits and costs. It can help product teams looking for objective prioritization techniques that factor in multiple layers of data.

[The ProductPlan Planning Board](#), pictured here, is one example of weighted scoring in action.

Using a scoring method to rank their strategic initiatives against benefit and cost categories, product managers can facilitate a more objective (and productive) discussion about what to include on the product roadmap.

In the case of the Planning Board, scoring is on a 1 through 5 basis. A 1 means either low benefit or low cost, and a 5 means high benefit or high cost. The score you choose determines what percentage of a category's weighting each item will receive.



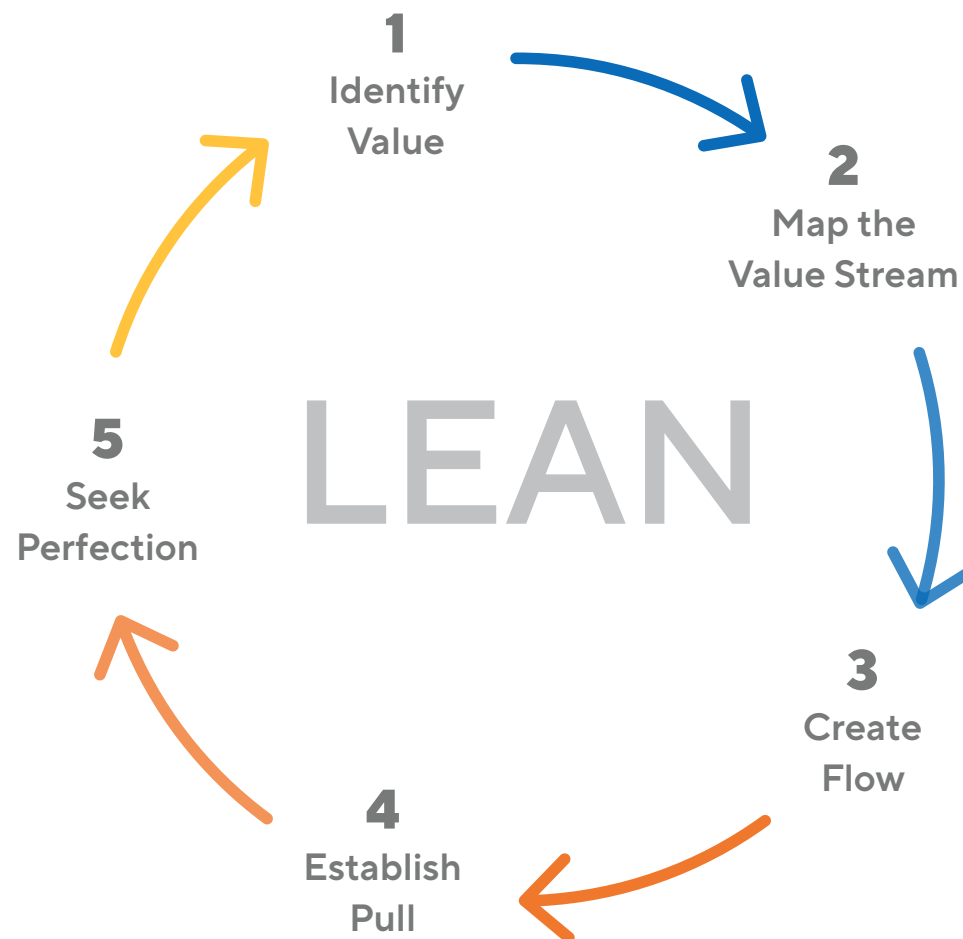
Title	Value	Effort	Score
 Bootstrap Upgrade	5	5	120
 API 2.0	4	5	100
 Import Functionality	2	4	80
 React Framework	1	3	80
 Data Logging Module	2	5	60
 Web App Overhaul	0	4	40



Frameworks Specifically for Agile Software Teams

Lean Software Development (LSD)

The lean software development (LSD) model focuses on eliminating from development everything but what the product needs. This is also sometimes called the Minimum Viable Product (MVP) strategy because the team's goal is to deliver a bare-bones product to users, then apply their feedback to improve the product over time.

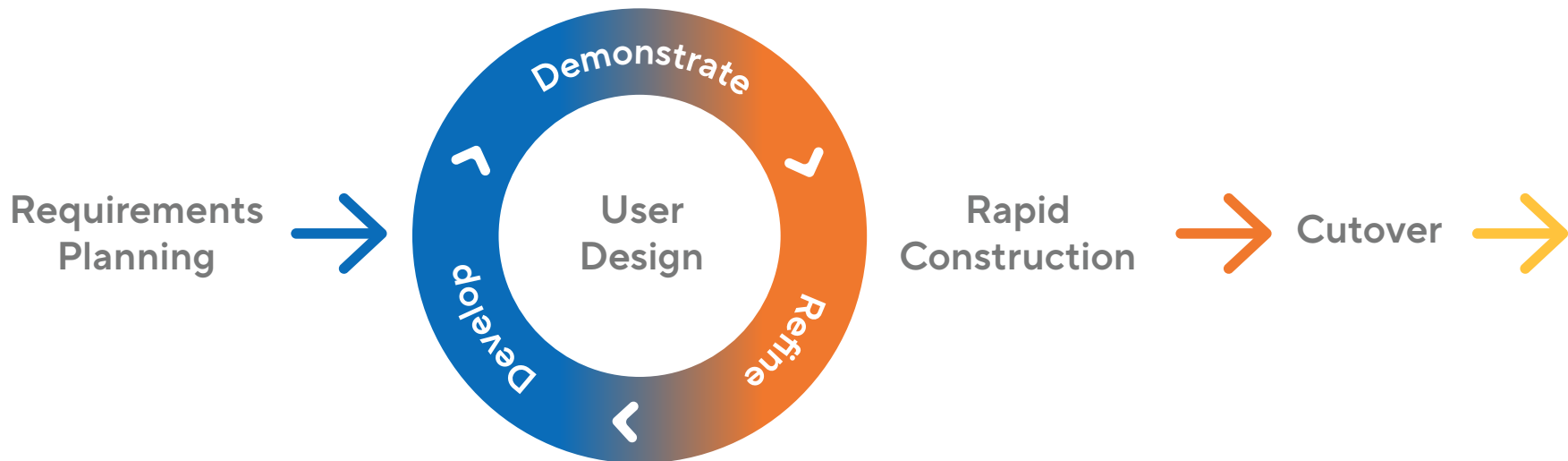


Rapid Application Development (RAD)

The rapid application development (RAD) is an agile prioritization framework focused on rapid prototyping software products, updating them frequently based on feedback, and continuously releasing newer versions to the market.

Using the RAD framework, product teams cycle regularly and quickly through the following four stages of product development:

- **Phase 1:** Requirements planning
- **Phase 2:** User design
- **Phase 3:** Rapid construction
- **Phase 4:** Cutover (pushing the new version live to users)



Crystal Agile Framework

Like the agile development methodology itself, the Crystal Agile framework focuses on interactions among people and teams rather than processes and tools. The framework gives product teams the freedom to develop and improve their workflows.

Because it is so flexible, the Crystal Agile framework is a popular methodology with organizations that want to empower their teams to work however they deem most effective.



Strengths include:	Weaknesses include:
Allows teams to work the way they deem most effective	Lack of pre-defined plans can lead to scope creep
Facilitates direct team communication, transparency, and accountability	Lack of documentation can lead to confusion
The adaptive approach lets teams respond well to changing requirements	



Frameworks Developed by Great Companies

Google's HEART Framework

Although Google developed the HEART framework for its user experience (UX) designers, the methodology also works well for product prioritization. The framework can help a product team evaluate each aspect of a product according to five user-centered metrics, which form the acronym HEART: happiness, engagement, adoption, retention, and task success.

	Goals	Signals	Metrics
Happiness			
Engagement			
Adoption			
Retention			
Task Success			

By forcing the product team to quantify each of these criteria for each of its feature ideas, HEART can help the team prioritize the most strategically impactful initiatives when they have more ideas or requests than they can work on.

Working Backward (the Amazon Method)

Amazon's working backward framework takes a novel approach to helping a product team determine which initiatives will give the product its most significant impact on users. Using this framework, the product team starts its work by pretending the product is complete and ready for launch. Then, the team writes a press release announcing the new product—which describes the product's target customers and the problems it solves for them.

The thinking behind Amazon's approach goes as follows: When a product team writes this mock press announcement, they get a gut-check on whether they're enthusiastic enough about the product to pursue the idea further. If they find themselves uninspired by the draft, that is a good indicator the product idea lacks something. It also indicates that the team has not thoroughly thought through the problems or the solution.





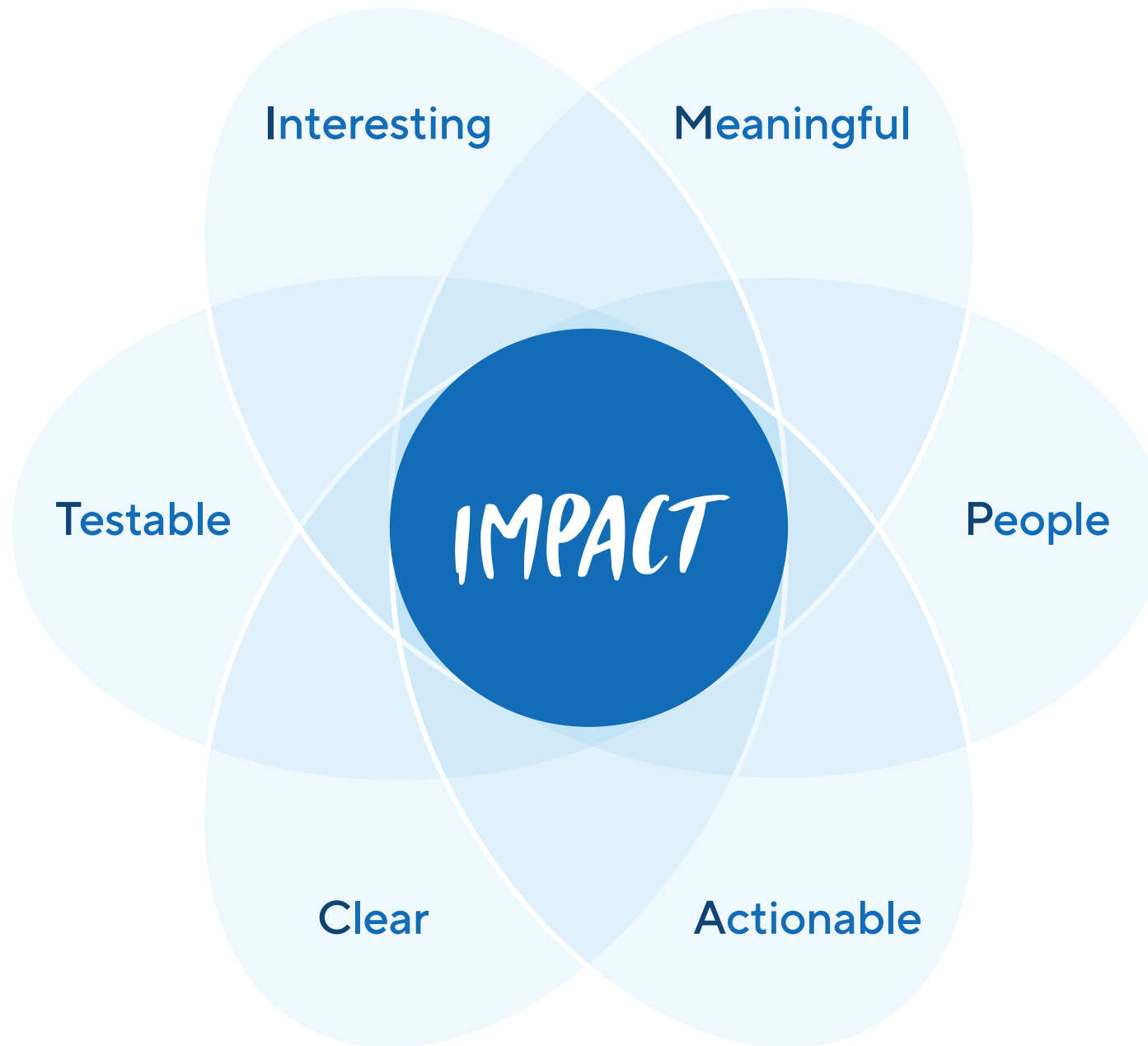
Bonus: A New Framework Developed by ProductPlan's VP of Product

The IMPACT Method

Developed by Annie Dunham, ProductPlan's VP of Product, [IMPACT](#) is as much a mindset as it is a framework. Annie designed it to help keep ProductPlan's customers and business at the center of everything we do. IMPACT ensures the product team is focusing their energy and resources on the most meaningful strategic work.

With this framework, the team can prioritize all items competing for space on the product roadmap by evaluating each against the following six criteria (which form the acronym IMPACT):

- 1. Interesting:** Does the product address the things our customers really care about? Can we tell a story of how the product creates positive change for them?
- 2. Meaningful:** Are we moving the business forward toward measurably reaching its goals? Are we providing real value to our users?
- 3. People:** Who is impacted by our product? Who uses it, who sees the benefits, and who pays for it?
- 4. Actionable:** Are we coming up with ideas that can be implemented and realized? Do we have the resources, budget, and expertise to execute?
- 5. Clear:** Do we truly understand what we're trying to do? Can it be concisely articulated so even a child could understand?
- 6. Testable:** Can you try things out before making a significant commitment? Are there ways to experiment and measure success on an ongoing basis?



“The prioritization method you choose will depend on your product, your organization’s culture, and your product management style.”

Jim Semick

ProductPlan's Co-Founder and Chief Strategist



How to Find the Right Framework for Your Team

As you saw from the summaries in the previous section, prioritization does not have a one-size-fits-all approach. Different frameworks will be more valuable and relevant for different types of companies. Even within your team, the most effective framework today might not be the right choice in a year or two.

To choose the best prioritization framework for your team's needs, you'll want to ask some fundamental strategic questions.

What success metrics will we prioritize?

Let's say your company plans to take the product-led growth approach. Your strategy is to build a product that essentially sells itself. Instead of growing a sales department, you will make your product easy for customers to find, try, and buy later. With this approach, you will focus entirely on delighting customers with your product—and you will not worry about growing revenue in the short term.

Under these circumstances, you would opt for one of the customer-centric frameworks, such as Design Thinking or the Kano Model. Your priority would be making happy customers.

If, on the other hand, you were a small startup with a minimal operating budget and impatient investors, you might instead choose a prioritization framework focused on



revenue-generating metrics. One example would be the AARRR framework because this approach prioritizes only those items that directly affect a business's bottom line—such as customer acquisition and revenue.

Do we want flexibility in our framework or a more predictable process?

For some teams, particularly in more prominent companies, a strict set of guidelines can help make processes more effective. For example, if your team would benefit from clearly defined rules in your prioritization approach, you might choose a framework such as feature-driven development (FDD).

But suppose you want to create more freedom to be creative in your prioritization decisions. In that case, you might instead opt for the Crystal Agile Framework, one of the most flexible approaches for agile organizations.

How much will our resource levels affect our priorities?

If your product team can secure only a small number of developers and a limited budget for your product, that should play a role in which prioritization framework you select.

For example, you might want to use the Value vs. Complexity model, which lets your team score each feature idea on how much value it will contribute to the product and how much effort it will take to build.

There are many other considerations when selecting a prioritization framework. For example, the size of your company, the amount of data your team can compile and analyze, and how well your team collaborates on strategy—these can all influence which framework will be right for you at the time. The great news is that if you try out one of these frameworks and it doesn't seem like a fit for your needs, you can move to another.

“Even the best prioritization strategy won’t be as effective as it could be for your product unless your entire cross-functional team is on board with it.”

Julie Hyman

Senior Product Manager, Quest Software



4 Best Practices for Rolling Out Your Framework

Prioritizing your product roadmap is a team sport. Whichever prioritization framework you select, the first step to making it successful will be rolling it out to your cross-functional team. Here are some best practices for a successful implementation.

1. Seek team-wide buy-in.

When you've decided on a framework to try, you'll want to make a case for it with your product team. Walk the team through the framework's formulas and processes, and communicate to them why you believe it will help you make better strategic decisions for your product.

You might even want to run through a mock prioritization session. Use a hypothetical product with a list of competing feature ideas, and ask the team to score or rank them all based on your chosen framework's approach. The idea here is to make it fun so your team learns how the approach works—without the pressure of making real decisions with your actual product.

The more you can earn your team's buy-in for this framework, the more effective your prioritization sessions will be going forward, and the more varied perspectives you'll bring to those decisions.



2. Limit the number of items you prioritize at a time.

Although each of these prioritization frameworks can help you narrow your list of competing ideas for product features and functionality, none of them can tell you how many to place on your product roadmap.

One of the risks of rolling out a new prioritization framework is that product managers find too many items on their backlog scored at a “priority” level. It leaves the team with a similar problem as before—deciding what to prioritize and what to shelve for later.

It’s important when you roll out a new prioritization framework that you set an internal limit on items you’ll ask your team to work on in a given timeframe. Let them know about this limit beforehand as well. Your developers will appreciate knowing that, no matter how many backlog items rank as “very important” or “very valuable,” you will do the work of further narrowing that list before giving them more tasks than they can handle.

3. Share your new framework with all stakeholders.

A significant benefit of having the right prioritization framework is that it helps your team communicate to stakeholders—executives, sales reps, investors—a clear set of guidelines for the initiatives that make it on the roadmap and when you cannot.

Often product managers find themselves unable to say no to an executive eager to prioritize a new feature. And in many cases, the reason is that they do not have a straightforward process they can articulate for prioritizing an item versus placing it on the backlog.

When you roll out your new prioritization framework to your product team, it’s also a good idea to share it with everyone else in the company. Let them know you have a new methodology for deciding which items to build into your product, and all new requests and suggestions will have to go through that same process.

4. Standardize your prioritization process.

One risk with introducing a prioritization framework is that, although your coworkers might be onboard with the concept of a framework, they don't follow the same process you've laid out.

If you introduce the AARRR method, some members of your product team might focus only on two of the metrics in that framework—say, customer acquisition and activation. When they rate feature ideas, they might ignore customer retention, referrals, and revenue.

For your framework to have the most significant positive effect on your company's strategic decisions, you want everyone reviewing initiatives according to the same criteria. In other words, you want standardization.

Servpro: Using a roadmap app to standardize the prioritization process across franchises

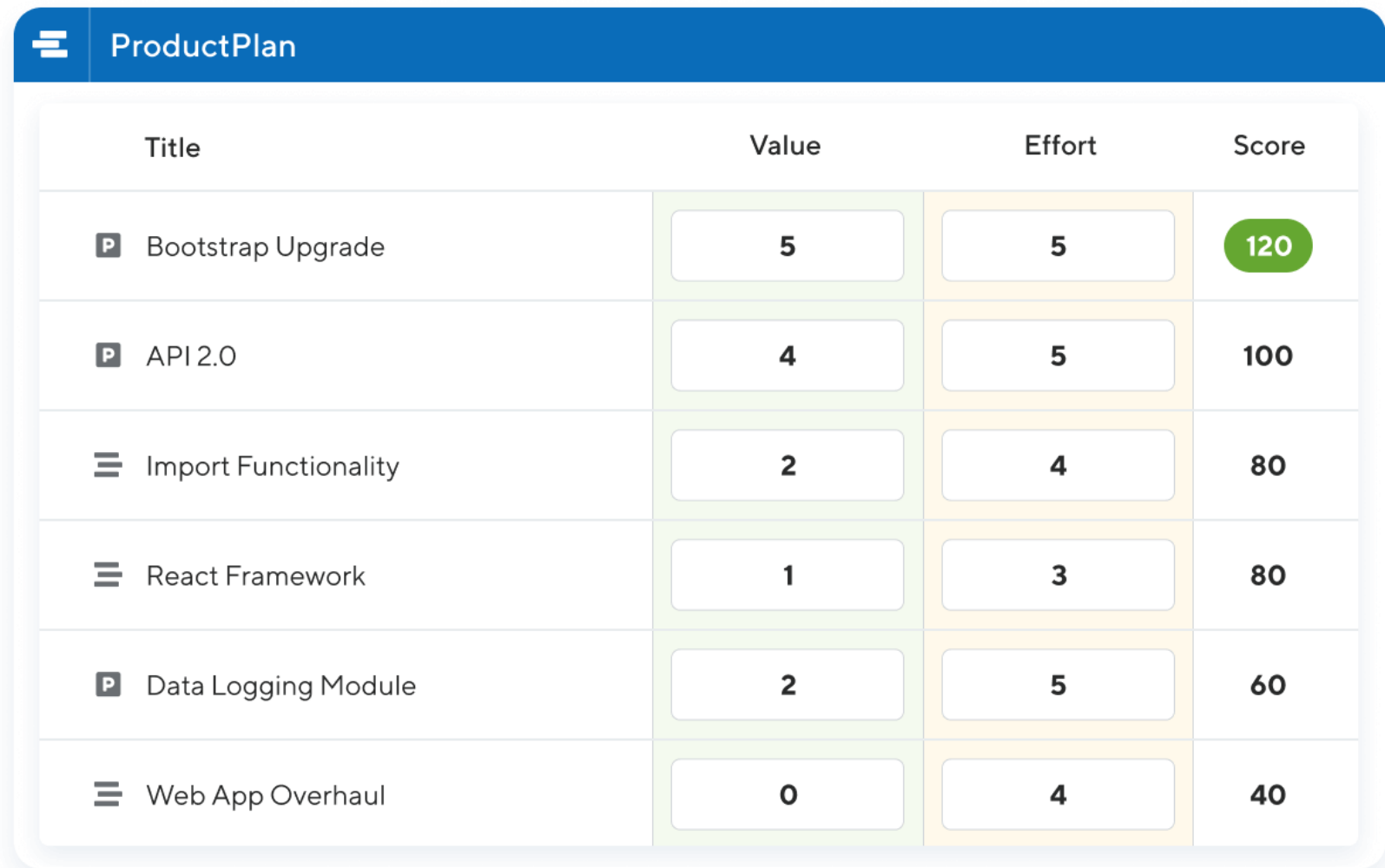
One real-world example is Servpro, a nationwide franchisor of water and fire cleanup companies.







Because the company had grown so rapidly, to more than 450 corporate employees, they found various departments using their processes to define high-priority items that received attention, resources, and budget.

So the company's Director of Program Management, Jason Wix, developed a standard process for any team to request development work. Wix then used ProductPlan's Planning Board feature (pictured on the next page) and a custom version of the Weighted Scoring model to evaluate all requests according to the same criteria.

Because the app makes it easy to share a roadmap with anyone online via a link, Wix could make his team’s strategic plans and priorities transparent to all departments across the company. As he explains, the results were transformational for Servpro’s operations:

“Everyone now knows why their item didn’t get to the top. So we put the onus on them to talk through things as a group and decide what makes the top.”



Title	Value	Effort	Score
 Bootstrap Upgrade	5	5	120
 API 2.0	4	5	100
 Import Functionality	2	4	80
 React Framework	1	3	80
 Data Logging Module	2	5	60
 Web App Overhaul	0	4	40

“Even with all the prioritization frameworks that have gained popularity... product prioritization remains a universal challenge across organizations...”

Austin Yang

Senior Product Manager, Landbot



Overcoming Common Challenges to Implementing Your Framework

After deciding to implement a new prioritization framework, many teams find themselves defaulting to certain bad habits. Here are the ones we at ProductPlan have seen plague product teams most often, along with our advice for preventing your team from falling into the same traps.

1. Don't prioritize based on what your competitors are doing.

It's common for a team to run through its prioritization method—ordering several competing feature ideas according to several criteria—but then still feel the pull of prioritizing the feature a key competitor already offers.

If your team has clear evidence that adding that me-too feature will add to your bottom line, then it can earn a slot on your roadmap. But your product's development should be based on the research, customer feedback, and innovative ideas that you and your team compile—not on what another product is doing.

2. Avoid saying yes to everything.

Even after prioritizing specific initiatives with their new framework, product teams often still struggle to say no to requests from salespeople, C-level executives, and customers.



A key reason for having a prioritization framework in the first place, though, is to prevent this type of development overload. This is why it's essential to let everyone know about your new framework. Then, once you've started using the framework, practice saying no to requests that don't meet the success criteria of your framework.

3. Don't prioritize what's easy.

An ongoing temptation product teams face is to rack up a series of “wins” by cranking out many features quickly. Sometimes, the development team encourages this because they prefer to work on functionality that is easier to build.

Resist these temptations. Remember, your product strategy should be focused only on building solutions that move the needle for your company—that delight customers, grow your market share, and add revenue—not on what you can develop and release in a hurry.

Define your product's vision, then establish a strategy for realizing that vision. And finally, develop a list of ideas for functionality and order those ideas using your prioritization framework. But, again, don't prioritize what's easy.

“The right prioritization strategy helps you keep clutter and ‘shiny things’ off of your product roadmap—so you can reserve the roadmap for things that will have a real impact on your market.”

Annie Dunham

ProductPlan's VP of Product



How a Prioritization Framework Improves Your Roadmap

As Annie's quote above indicates, one of the great advantages of using a prioritization framework for your product decisions is that it helps prevent your roadmap from filling up with the wrong initiatives. Here's how the right framework can improve your product roadmap.

1. It can free up your team's time and creative energy.

Without a framework for identifying the initiatives that will provide your product team the greatest strategic advantages, your team will spend time and energy cycling through your backlog.

A framework can standardize these steps. If you choose the Weighted Scoring model, for example, your team will have a specific series of steps to evaluate each initiative's value against its cost. When you've completed these steps for all competing ideas, you'll have your priority items. No more needing to argue, reconsider and relitigate the list. You will know—based on evidence—which items to add to the product roadmap.

Does this mean your roadmap will have only winning feature ideas? Of course not. No product team can be certain in advance which items will succeed with customers. But because the prioritization framework standardizes and shortens the decision-making process, the team will have more time to develop potentially great ideas.



2. It can prevent ad-hoc requests from claiming unearned spots on the roadmap.

Using a prioritization framework can also help your team deal more effectively with the never-ending stream of requests that can derail their progress. For this tactic to work, though, you'll need to make sure all of your stakeholders know you're using it.

Let's say your team has no fixed stages or guidelines during the development process. You improvise your approach from scratch for every new product or even for every update to an existing product. What's to keep a sales rep or executive from demanding your team stop everything from building something they want to prioritize?

Without a process that you can point to, you will have to negotiate these requests every time. And in many cases—particularly with an executive—you'll lose. Worse, every time they have to shift gears and refocus on a different creative project, your team risks not fully re-engaging in the work they were doing on your product.

Using a framework that allows you to stop accepting new ideas or requests after a certain stage will help you protect your product team from these disruptions and frustrations. It will let them stay focused creatively on the same initiative throughout the development process. That will improve the chances your product will be a success.

3. It can help your team generate more strategically viable ideas.

As your product team gains practice working with your prioritization framework, they'll begin asking themselves the key question: How does this initiative measure up against our prioritization criteria?

Suppose you're using Annie's new IMPACT approach, for example. In that case, your coworkers might quickly ask themselves if their new feature suggestion will prove **I**nteresting and **M**eaningful, if the **P**eople they're designing it for will see its benefits, if it is an **A**ctionable request, if its purpose will be **C**lear to the team (and users), and if the team can create a **T**estable version before committing to develop the full feature.

In other words, the right prioritization framework can improve your roadmap by reducing the number of items your team proposes that will likely fail to meet your success criteria.

“Are we going to win new customers? Sustain existing business? Keep relationships? If we’re looking at a feature and we can’t decide why we’re doing it, then we shouldn’t be doing it at all.”

Chris Dudak

Senior Manager of Product Management, Lionbridge



Yes, This Works for Real Companies

ProductPlan customer Lionbridge uses prioritization within ProductPlan's roadmap platform.

Lionbridge is a leading language localization service, translating content into more than 350 languages for clients worldwide—using a combination of in-house linguistics experts and highly sophisticated machine-learning applications.

But Chris Dudak, the company's Senior Manager of Product Management, explains that keeping up with demand to update the product continually became increasingly difficult. The key challenge was dealing with large volumes of competing priorities from various stakeholders.

Without a standard framework to evaluate those requests, Chris and his product team found it challenging to know which ones should receive the development team's limited resources. They also had trouble explaining their rationale behind the decisions to frustrated stakeholders.

Leveraging ProductPlan's Planning Board—and the weighted scoring framework

Then they started using ProductPlan's roadmap platform—and running competing feature requests through the Planning Board, which uses the weighted scoring approach to product prioritization. Now, Chris's team could create objective scores for each agenda



item according to a consistent set of criteria—including its potential strategic value and ability to delight customers, against its estimated development effort and operational costs.

And as we pointed out in the previous section, this solution has had a positive effect on the Lionbridge product team's workflows and efficiency. Chris notes that the team now has a framework to quickly identify the most strategically viable projects, place those items on the roadmap, share the team's strategic thinking with everyone in the company, and free up the product team's time and creative energy.

As Chris explains, “With ProductPlan, our product management team is spending less time talking about what we're doing and more time actually doing it.”

“With this Impact-first mindset, businesses and teams have an overarching umbrella underneath which they can evaluate frameworks and processes, all with the end goal of identifying their greatest opportunities to make a meaningful difference.”

Annie Dunham

ProductPlan's VP of Product



Conclusion: It's All About Making an Impact

We hope this guide has persuaded you to find a prioritization framework to help guide your product team's work. But we also want to reiterate that the specific framework you choose is less important than establishing some way of vetting ideas and prioritizing the ones that promise to do the most good—for your product, your users, and your company.

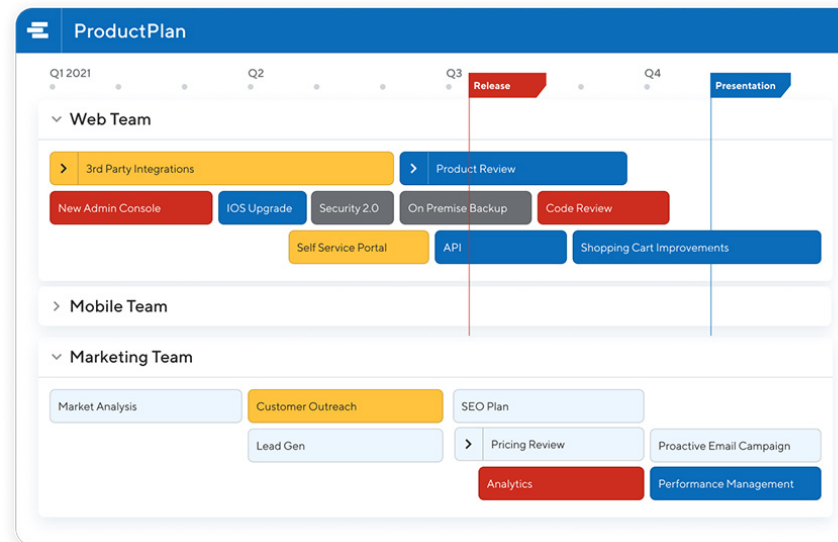
Whatever type of product you're building, whatever industry you're in, your high-level goals should be the same: to solve real problems for your market, to improve people's jobs or lives in some way, and to help your company succeed and grow. In other words, you're developing a product not just to release features but to make an impact.

Selecting a prioritization framework to inform which solutions to build is a great way to start. And the team here at ProductPlan wishes you success.



About ProductPlan

ProductPlan is the most intuitive roadmap platform trusted by thousands of teams including Microsoft, Progressive, and Booking.com. Create collaborative roadmaps that drive strategic decisions so you can build what matters.



[Start Your Free Trial](#)