FEATURE-LESS
ROADMAPS:

Unlock Your Product's Strategic Potential

ProductPlan

FEATURE-LESS ROADMAPS:
Unlock Your Product's Strategic Potential

TABLE OF **CONTENTS**

INTRODUCTION

Why we wrote this book

The goal of any product roadmap is to capture and communicate a high-level strategy and create alignment. The roadmap shares the vision and objectives for the products and shows how those objectives support the broader strategic goals of the company.

As the Director of Product at a roadmap software company, I've been lucky to work with thousands of product managers. I've seen many product managers struggle to stay focused on their product's strategy and vision while managing a deluge of feature requests. Commonly, this results in a roadmap that is merely a long list of product features as opposed to a strategic plan to deliver value. Feature-laden roadmaps encourage premature commitments that limit the potential of the product team's work. The roadmap needs to be at a higher level to tell your product's story.

Enter: **feature-less roadmaps.**

In the chapters ahead, I've broken down feature-less roadmaps into five parts:

> **Part 1: A product roadmap is not a list of features**
>
> **Part 2: Types of feature-less roadmaps**
>
> **Part 3: Feature-less roadmaps empower product managers**
>
> **Part 4: Rally your team around a feature-less roadmap**
>
> **Part 5: Unlock your product's strategic potential**

This book isn't about ditching features all together; it's about finding the right time and place for them. *Feature-less Roadmaps: Unlock Your Product's Potential* is your guide to understanding, planning, and building a feature-less roadmap that puts your product strategy front and center.

Annie Dunham
Director of Product, ProductPlan
www.productplan.com

A PRODUCT ROADMAP IS
NOT A LIST OF FEATURES

A PRODUCT ROADMAP IS
NOT A LIST OF FEATURES

It's Time to Update Your Roadmap

Roadmaps are most effective when their primary purpose is to communicate the product strategy and its value proposition to both customers and the organization.

When it works as intended, a roadmap helps the product owner earn stakeholder approval to proceed with development and ensures all teams involved are working towards the same goal. As such, product roadmaps are the key to unlocking your product's strategic potential.

So why do some product roadmaps fail to support a product in reaching its full potential? It all comes down to the contents on the roadmap. The way you frame information communicates the story you want to tell.

As our latest report showed, lots of product managers are not completely satisfied with how they communicate their product strategy. Additionally, 33% of product managers said features are the primary content on their roadmap. I draw a line from one of those data points to the other. There is no story with features. A roadmap crammed with ground-level details loses sight of the big-picture strategy.

If you're wondering whether or not the alignment around the content on your roadmap needs an update, see if you relate to one of these red flags:

- **Your organization doesn't have a consensus of your product's vision (your big-picture plan for what the product will accomplish in the market and for your company).**

- **You and your team can't identify a high-level strategy to make that product vision a reality.**

- **There are features on your roadmap where there shouldn't be.**

Each of these red flags trickles down and affects the next. That's why in this book, we will review how to become empowered with company alignment through a feature-less roadmap. You will learn how a feature-less roadmap helps you discover the outcomes that matter to your organization and your customers, ultimately unlocking your product's strategic potential.

Why Features are Still on Your Roadmap

Somewhere along the way, the line between backlog and product roadmap blurred. A backlog in the product development context is a prioritized list of items that the team agrees to work on. Typical items on a product backlog include user stories, changes to existing functionality, bug fixes, and features. The features on your backlog are the tactical elements that enable you to deliver your product roadmap.

However, despite our best intentions, features still can find a way of sneaking onto roadmaps—even if they're disguised as a goal or an outcome. Some product managers like the sense of accountability that they provide. Typically, the features appear as task lists, arranged on a timeline (albeit a vague one). Beware of this format. It creates premature commitments and delivery risk. Features will get a job done, but they shouldn't be the focus at the roadmapping stage. Send them to the backlog. Feature-laden roadmaps create external pressure to build the things on the list without ensuring they're solving a real customer problem or asking why the problem is happening in the first place.

Whereas a feature-less roadmap will help you get both your backlog and your roadmap into a problem statement—that way, your backlog complies with your product roadmap's objectives. The conversation will be "Is this a problem we want to solve now?" not "Here's a bug, we should fix it."

The only conditions that might require a feature-based roadmap are:

1. **Your product is just getting started. The founders come up with what to do; then, you immediately begin to create a list of features.**

2. **Requirements are firm and necessitate a product development schedule.**

3. **Feature releases are time-bounded.**

So, does this mean features will never again appear on your roadmap? Not exactly. Features will still be in your roadmap. However, they will only appear within the appropriate context and as sub-elements of your feature-less roadmap. Product managers already have a variety of tools for cataloging small details behind product development, such as the product backlog, project management apps, and even handwritten to-do lists. The roadmap is not where these details should live at the risk of stifling product innovation.

The Risks of a Feature-Laden Roadmap

Becoming a Feature-Factory

Your customers and CEO are savvy about hearing a running list of features in your roadmap presentation, and they are more comfortable with requesting features to be added to the product. But this is a slippery slope. The value of a single feature isn't guaranteed in an ever-changing market. Depending on the newest technological innovations and most prestigious customer's needs, how you will create value is always susceptible to change. A feature-laden roadmap assumes your strategy is locked in place and that you've figured out the execution as well. Not to mention, there is an implied delivery commitment to features that appear on a product roadmap.

Do you want to hold your promise to a specific functionality two years from now? You face the risk of your product rapidly shifting from "solving customer problems" to becoming a feature-factory. When teams adopt a feature-factory mindset, a term coined by John Cutler, they ship far faster than they can learn and create value. There is a pressure to treat features on a roadmap like a to-do list that you can cross off as you deliver. You might start to feel like you're only succeeding at work as long as you keep adding to the feature-factory. Feature-stuffing over complicates a product and can make it worse, rather than better, if the product doesn't work well with the currently existing features. It creates opportunities for endless debate, confusion, and disagreement amongst stakeholders because while you may understand "how," what's often obscured with feature-laden roadmaps is the "why." Ultimately, your team will miss the forest for the trees.

Why are your CEO and customers requesting a particular feature? A feature-less roadmap requires you to give the customer and CEO feature requests a proper sniff test.

Too Much Focus on Outputs Rather than Outcomes

One of the challenges of shaking features off your roadmap is getting accustomed to thinking about the bigger picture. That's why many product managers favor thinking about the outputs they're delivering rather than the outcomes. The output is anything that your team delivers. Output mindsets are an easy, shortsighted trap to fall into—frequently, for agile teams. Whereas an outcomes mindset is working to create a change in the behavior of your customers, users, and stakeholders as a result of your product.

In big companies, a top-down culture often encourages stakeholders to hand down features for their teams to deliver. By focusing solely on the top-down feature output, you run the risk of falling victim to what Josh Wexler calls solution sickness. Solution sickness is all about fixating on a solution and ignoring any alternative ways of solving a problem by trying to address the problem better.

Solution sickness doesn't allow any creativity to emerge when trying to reach a legitimate desirable outcome, since you're working on only pumping out a deliverable. You'll start to feel like an order-taker, whose role is to provide these features in the best way possible. Then comes the day when someone asks, "Why did this feature get prioritized?" and it will become apparent that the features fixed a short-term problem without addressing a long-term outcome. An outcomes mindset ensures you focus on the quality of improvements over quantity. No matter how brilliantly you plan your product's release or how well your teams promote it, your product is likely to fail if it doesn't solve the right problems, which will only happen with an outcomes mindset.

Eyeing the Competition

Features leave you susceptible to the tempting allure of playing catch up with competitors. With a competitive feature strategy, your focus is not on addressing the customer's problem; it's on eyeing the competition. Ultimately, your product will come up short for your customers because it doesn't address their needs.

If you move your roadmap to one that answers the customers' problem in a feature-less way, you will have a headstart on the competition. Your solutions will be innovative and accurately address the issues, while your competitors are working on something less critical.

2

TYPES OF
FEATURE-LESS ROADMAPS

TYPES OF FEATURE-LESS ROADMAPS

2

There's a variety of ways you can structure your feature-less roadmap. Here's an overview of five popular feature-less roadmap formats.

Themes:

"You're going to get distracted. You're going to get distracted by a loud customer. You're going to get distracted by the next shiny object. Someone's going to come to you with a fantastic idea, and they phrase the idea as a feature. With a theme, though, it helps you stay on track. You're always tying back these things that you're delivering, these features and enhancements, to a specific theme."
- **Jim Semick**, ProductPlan

The real advantage of having a feature-less roadmap formatted around themes is how closely themes are tied together with your company's objectives. Themes enable you to observe and reconcile what your users need with what's viable for your company. They elevate conversations and gather momentum around more significant ideas. Themes are a powerful approach to "ground" your roadmap, so every team at your company is aligned to work towards that outcome.

Most theme-based roadmaps have a few themes presented on the product roadmap. Ideally, themes describe customer value—what customers will receive or the job you'll help them accomplish. The themes answer these questions: What problem are we looking to solve for our customers? Why does our team feel this problem is worth solving? Why should we prioritize this problem over others? Each theme should have a measurable goal and expectation that can be tied up to your company goal. Then, each theme will have supporting features prioritized underneath them. The themes help you stay strategically

on track because each initiative that you're delivering (features and enhancements) must be tied back to a specific theme. Collectively, these themes will have a more significant impact than any one-off feature.

Group Features into Themes

Tenant leasing software—a case study

Feature-laden roadmap example

1. **Move in workflow**
2. **Lease info**
3. **Move-in checklist**
4. **Tenant screening**
5. **Online rental app**

↑

The problems:
Overly optimistic, not highest customer value, hard to shift priorities, presumed we knew all the answers

Feature-less roadmap example

▼ | **Move in renters 50% faster**

Lease info

Online Rental App

Tenant Screening

Move-in Workflow

???

Many product leaders have realized that tying themed roadmaps to product strategy is the best way to get stakeholder alignment. If you can get executive alignment on the goals first, it's easier to create themes that align with those goals. As part of the process, it's essential to discuss the metrics and KPIs that define whether the goals are met. Rather than telling your stakeholders, "We're going to create innovative software for the property management industry," a good theme would be "Reduce shopping cart abandonment— which in turn will keep 22% more customers on our site." That way, it's tied together with the outcomes that you want to create for customers and your company. It also is something that improves the product manager's job. It changes the conversation from why a stakeholder's pet feature isn't prioritized to what the most important outcome is for the business.

There's also no perfect timing for a theme. Some themes can be knocked out in a single release or span multiple quarters, depending on its scope and scale. Themes will take as long as is deemed necessary.
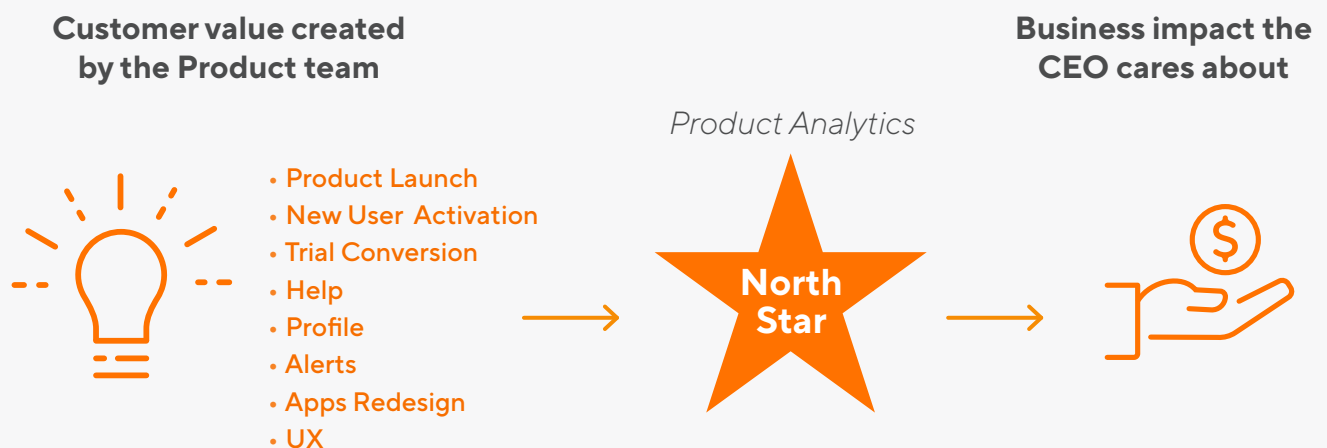
North Star:

"The North Star Metric (NSM) is the single metric that best captures the core value that your product delivers to customers. Optimizing your efforts to grow this metric is key to driving sustainable growth across your full customer base."

- Sean Ellis, GrowthHackers

Another feature-less roadmap format is a North Star roadmap. In business, the North Star Metric was coined to give organizations a singular focus on a particular goal. Once you have an active North Star in place, it has an exponential impact on all decision making around your roadmap. With a North Star Metric, every roadmap activity can be judged based on whether or not it is advancing this metric. It is your unique product strategy. Instead of being distracted by day-to-day matters, everyone can always define success by whether or not they are advancing the company using this metric.

North Star Framework

Customer value created by the Product team

Product Analytics

Business impact the CEO cares about

• Product Launch
• New User Activation
• Trial Conversion
• Help
• Profile
• Alerts
• Apps Redesign
• UX

North Star

A useful North Star focuses on distilling down customer value to a single measurable number and the exchange of that value. It connects the customer value you are trying to create as a product team with the business impact that the executive team in your company ultimately cares about. If a project, feature, or initiative doesn't improve that metric, then its value must be questioned for its lack of relevance.

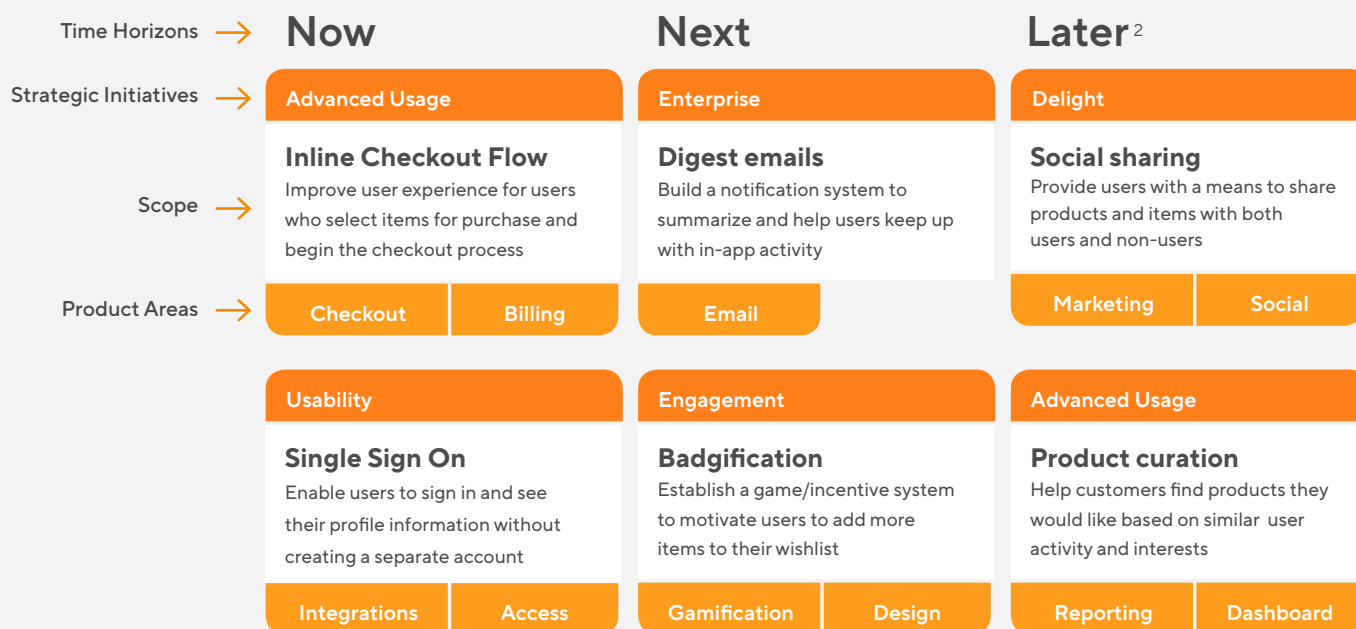A North Star roadmap looks like this with corresponding examples:

North Star (name of roadmap)	The social site is tracking # of users who add 10 friends in the first week.
The most important high-level thing that you can do to influence the North Star	Update Invitation flow
What will be looked at first?	• Suggest friends • Integration opportunities • Content

At this point, if the alignment is built on these being the most influential areas to advance the North Star, the team can begin experimenting and getting feedback on specific ideas and features. This may even be in the roadmap (identify top 3 ways to improve invitation flow)–from that, you'll build a backlog.

Now, Next, Later Columns:

A 'Now, Next, Later' column roadmap focuses on the status of identifying priority columns of a product's problem areas.[1] The priority columns are "Now," "Next," and "Later." The Now and Next columns are effectively the goals for the quarter and Later is the longer-term roadmap.

Time Horizons →	**Now**	**Next**	**Later** [2]
Strategic Initiatives →	Advanced Usage	Enterprise	Delight
Scope →	**Inline Checkout Flow** Improve user experience for users who select items for purchase and begin the checkout process	**Digest emails** Build a notification system to summarize and help users keep up with in-app activity	**Social sharing** Provide users with a means to share products and items with both users and non-users
Product Areas →	Checkout / Billing	Email	Marketing / Social
	Usability	Engagement	Advanced Usage
	Single Sign On Enable users to sign in and see their profile information without creating a separate account	**Badgification** Establish a game/incentive system to motivate users to add more items to their wishlist	**Product curation** Help customers find products they would like based on similar user activity and interests
	Integrations / Access	Gamification / Design	Reporting / Dashboard

If something is going to fall under Now, then it is in progress and has a relatively firm delivery time frame. This planning cadence works well for teams that already use bi-weekly sprints. Next is anything that you're 90% confident you will do but haven't started yet. It's anything that has been scoped and prioritized but does not have a set timeframe. The Later column is a useful place to park your ideas and plant some seeds around whatever it is your team is passionate about. The beauty of this column is that it saves organizations from long debates so they can get back to building their Now and Next initiatives. The prioritization within this bucket is less critical to the other two. You'll revisit them in a few months, so don't worry about being as specific in this column as the other two. Cards sitting on the Later column don't have to have all those answers yet, but as a card moves closer to the Now column, they should become a lot more detailed.
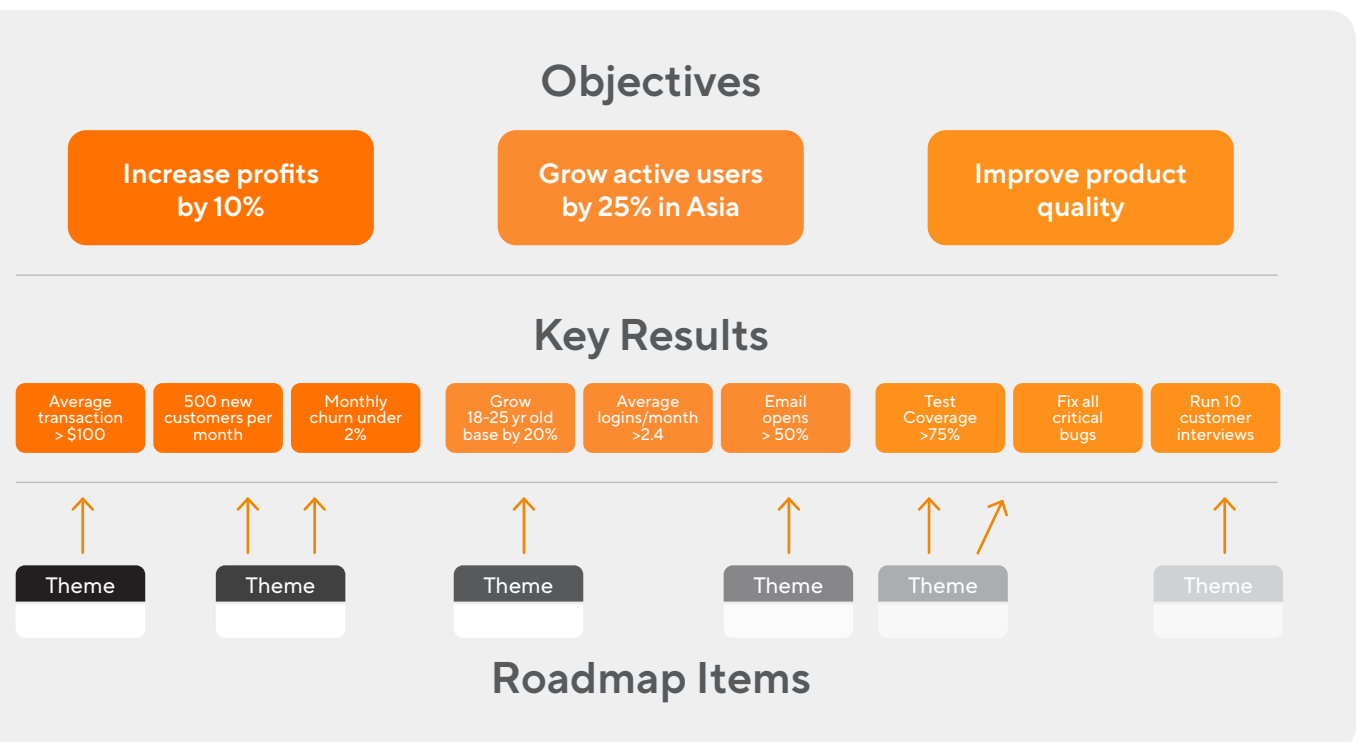
---

[1] https://www.prodpad.com/blog/creating-product-roadmaps/

[2] https://www.prodpad.com/blog/how-to-build-a-product-roadmap-everyone-understands/

This type of feature-less roadmap enables you to communicate with your company that you're aware of the problem, but you don't have to provide anyone with the exact solution at this stage. You'll be able to incorporate customer feedback into your product planning dynamically. Once you have your initiatives finalized, you can attach more supporting details for anyone who wants to drill further down. Keep in mind, the idea around each roadmap cards should always be strategic, not tactical. For instance, "rewriting transactional emails" is too specific to be a Now, Next, Later roadmap strategy. The card should inform your team what you're doing, what goal it ties back to, and why it's on your roadmap.

OKRs:

Objectives and Key Results, or OKRs, are a model for setting business goals and trackable outcomes. The OKR framework was popularized by Google, which attributes much of its success to this goal-setting model. The "O" represents objectives (high-level goals), and the "KR" represents measurable key results. While OKRs differ from company to company, they are always tied back to larger company goals.[3]

Objectives

| Increase profits by 10% | Grow active users by 25% in Asia | Improve product quality |

Key Results

| Average transaction > $100 | 500 new customers per month | Monthly churn under 2% | Grow 18-25 yr old base by 20% | Average logins/month >2.4 | Email opens > 50% | Test Coverage >75% | Fix all critical bugs | Run 10 customer interviews |

| Theme | Theme | Theme | Theme | Theme | Theme |

Roadmap Items

One of the significant advantages of the OKR methodology is that it helps to keep cross-functional teams aligned and accountable for the organization's priorities. Typically, these OKRs are ambitious stretch goals phrased in clear and memorable statements.

Inherently, product OKRs are tied to larger company goals. It's critical to use actual numbers to make the key results portion of your OKRs measurable. State your goals in simple words as "Increase user engagement by X%" or "Make registration process faster by X%." Then, each of these goals determines the reason for every feature's existence.

Therefore, by organizing the information around the goals, you will keep your feature-less roadmap high-level and make your strategy and vision easy to understand. A set of shared OKRs can help everyone understand how their day-to-day to-do items support the key results the company has set for itself. OKRs gets people out of the weeds of, "Why that feature and not my feature?" and back to the broader, strategic vision.
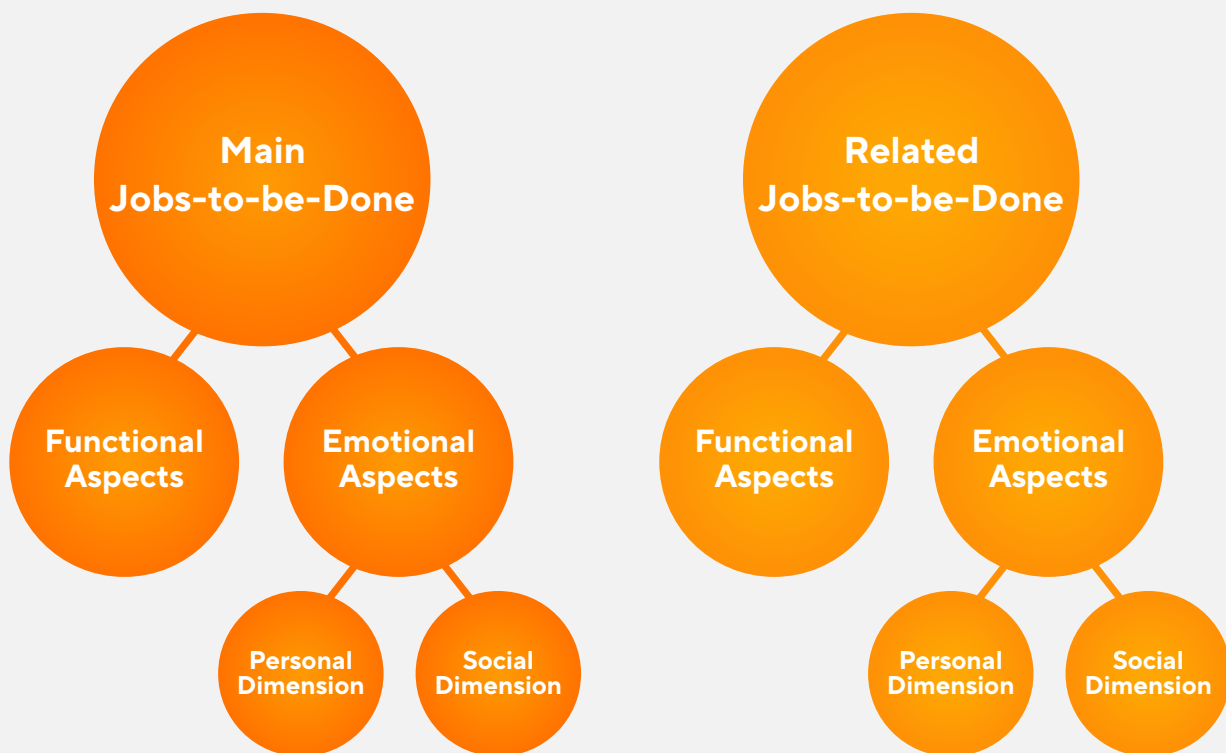
Because they involve high-level strategic objectives that often require the contributions of several departments, OKRs can help create a greater sense of teamwork and support across the company. Objectives and key results can make cross-functional teams more agile. Many people describe OKRs as an agile framework, which makes sense. By setting short-term objectives and measuring the outcomes each quarter, a company can more quickly adjust priorities and reallocate resources if needed.

An OKR's objective section should read like a roadmap theme or epic. It should be a short, clear statement of a high-level goal such as, "Improve the trial process for users." Turn user needs or problems into OKRs. After your team has decided on a set of OKRs for your roadmap, pull everyone together regularly to check the progress of each objective's key results.

This check-in would help make sure everyone continues working toward your agreed-upon priorities and give you early warning signs if the goals you've set were too ambitious.

Jobs-to-Be-Done:

The jobs-to-be-done framework (JTBD) is an approach to developing products based on understanding both the customer's specific goal or "job," and the thought processes that would lead that customer to "hire" a product to complete the job.[4] JTBD was developed by Tony Ulwick, founder of the innovation consulting firm Strategyn. When using this feature-less framework theory, you're discovering what users are trying to accomplish when they buy your product and how you can make them better. Therefore, the focus is not on the product but the customer. Particularly on the higher purpose for which a customer buys a product. A JTBD roadmap can help focus product development on solving problems as opposed to building features.

Main
Jobs-to-be-Done

Functional
Aspects

Emotional
Aspects

Personal
Dimension

Social
Dimension

Related
Jobs-to-be-Done

Functional
Aspects

Emotional
Aspects

Personal
Dimension

Social
Dimension

in the often-used example, the feature request you might hear is, "I need a drill." But then after a little probing, you'll discover what the customer needs a well-drilled hole. JTBD understands that all people buy any product or service to get something done or solve

---

[4] https://www.productplan.com/glossary/jobs-to-be-done-framework/

a problem. What is the emotional state they're hoping your product will give them? As a result, it forces the product team to uncover the underlying goal that users are trying to achieve. It's not really about a well-drilled hole; it's about the enjoyment of seeing a picture hanging in their living room. This discovery frequently results in a compelling customer experience that better understands the needs of its markets.

With many approaches to product development, organizations ask their target user personas what they want—and then build what their users tell them to. Something to be wary of with this approach is that your users often don't have the vision or vocabulary to explain what they want, especially if nothing like it has reached the market yet. By applying the jobs-to-be-done framework, you can help uncover not just what your users think they want, but what your users really want—and why.

Note: You can have a combination of all of these roadmaps. For instance, you might have a Now/Next/Later roadmap that is based on Jobs-to-be-Done or, you might have a North Star roadmap map that uses Themes.

FEATURE-LESS ROADMAPS
EMPOWER PRODUCT
MANAGERS

EMPOWER PRODUCT MANAGERS

The risks of a feature-laden roadmap have been established, and we've explored some feature-less format options that will unlock your product's strategic potential. Now let's dive into how a feature-less roadmap will specifically empower you, the product manager.

Say No to Your Team Better.

As product managers, we talk a lot about the power of "no" when it comes to our roadmaps and the content that is added to them. The truth is, a one-off feature has a much better chance of slipping through the cracks and being added to the roadmap if there's no theme or strategy for it to be held up against. Sometimes the executive team has a bad case of shiny object syndrome and provides their own set of feature marching orders based on the outcomes they desire. That order might not translate well into your product and falls short of your customer's expectations.

Feature-less roadmaps speak to what stakeholders ultimately care about, the goals and objectives of the company. Now, you can say no, and give people the evidence and reasoning behind your product decisions that everyone understands. You'll eliminate any chance of company vision, and product vision alignment is lost. Does the request support your North Star metric? Does it fall under one of your roadmap's three themes for the quarter? All the while, you can discuss customer feedback and ideas through the lens of your roadmap without exacerbating the problem and being the bad guy batting down ideas.

Reactive vs. Proactive Strategy.

In our recent report, 37% of product managers wished their company had a clearer strategy.[5] Product managers are struggling to align the outcome of their product roadmap with the company strategy. Due to the fact that there can be much top-down

---

[5] https://go.productplan.com/2020-product-management-report/

miscommunication around the company vision or strategy, it can be challenging to question the company strategy status quo. As a product manager, the feature-laden can trap you into having the wrong conversation or story. Especially when product managers lack access to decision points along the way. It's effortless to get fixated on features and forget about the underlying problems you're looking to solve when you don't know the bigger picture goal you're working towards.

Feature-laden roadmaps are often a result of top-down management and are reactive. It's looking at your backlog and being a non-stop machine to churn out features. But if you have a feature-less roadmap guided by OKRs or solving a JTBD problem, then you are empowered to be proactive. A feature-laden roadmap also puts you in a position where you can win the battle but lose the war. Fast forward to the end of the year. What happens when you're standing in front of your management team explaining that rather than building a product that works toward the company vision, you focused on delivering every single feature on time. Leaving aside how implausible that is, they will not be happy.

Give yourself a chance at wowing your team and focus on the problem you addressed in your feature-less roadmap, not the solution. This reframing is the fundamental idea behind a feature-less product roadmap–and its benefits are enormous and immediate. No longer will you be locked into prescriptive, solution-based plans too far in advance. This agility enables nimble execution to coexist alongside long-term planning. It prevents preemptive and unnecessary feature commitments. You will be more adaptive to changing market conditions without straying from your product strategy.

Hold Yourself Accountable.

A product manager has to be wary of the effect their ego might have on the product. No matter how smart you are, how much experience you have, and how much subject matter expertise your team possesses, product teams can't rely on previous experience, hunches, and instincts to make plans. For instance, some product managers risk feeling inclined to release something 'cool' every month. Or they get too in love with their solution or a feature even after that particular service's need becomes obsolete. Then the product development team spends too much time trying to figure out how to square this declining feature with the rest of the product. Forget it. Drop it and move on and away from that type of ego-thinking with a feature-less roadmap.

Once you know you're going to solve the problem, you need to figure out how. Have technologies changed? Are there other parts of your portfolio that should influence the solution? If you can stay focused on themes, you can optimize the detail work. That said, some detail work helps you better identify customer problems, so there's a balance.

You have to know how you'll measure the roadmap. By tying everything back to measurable outcomes, product teams move beyond checking boxes and crossing things off lists to moving the needle in a big way. "Ensure that every goal is measurable," says Roman Pichler of Pichler Consulting. "This allows you to tell if you have met the goal or not. If your goal is to acquire customers, for example, then ask yourself how many new customers should be acquired, or if your goal is to reduce technical debt, determine how much of the bad code should be removed or rewritten. If you don't state a target, it will be hard to tell if you have met the goal or not."

Creating a winning outcome-based feature-less roadmap is all about setting achievable and desirable goals, so organizations must work carefully to define those intended results and ground them in reality. The benefit of a feature-less roadmap is that it challenges you to think about how you will measure the success of your efforts. However, if you don't have all the data you need to make fully qualified plans, then that can lead to problematic expectations.

To succeed, feature-less roadmaps depend on identifying the outcomes that will really have a significant impact on the product and not just appease a complaining customer or satisfy the ransom demand of a stakeholder.

Understand Your Customers.

"For long-term success–yours and the customers'–you need not just to help your customers achieve their required outcome, but you need to help them do that in a way that is appropriate for them,"
- **Lincoln Murphy,** author of Customer Success[6]

---

[6] https://docs.google.com/document/d/1K4QHfc2cqq0IOCbKTJnqyvb7s3godWwpw4GY4KxdfyY/

Most importantly, feature-less roadmaps require deep and thorough customer insights in order to determine how they work into your roadmap. You can't fill a roadmap with customer problems if you don't understand your customer's needs or desired outcomes. If a customer hasn't thought about why they're asking for a particular feature—rather than just adding it to the backlog, help them describe their objective. Figure out their "Why?" Every time a customer's feedback is, "We need feature X." ask them, "What do you want to achieve?"

"If it doesn't help your customer achieve that required outcome in the right way–the way they want or need to achieve it–then you failed to deliver the appropriate experience, and the customer won't see the experience as one that was successful. Even if they achieved their required outcome!" said Lincoln Murphy.

With a feature-less roadmap, you're setting yourself and your product up for a proactive strategy, a customer-focused solution, and story to better work with your team and stakeholders.[7]

[7] https://www.productplan.com/roadmap-themes-north-star/

RALLY YOUR TEAM
AROUND A FEATURE-LESS ROADMAP

RALLY YOUR TEAM AROUND A FEATURE-LESS ROADMAP

With your team on board, it's time to transfer from a feature-laden roadmap to a feature-less roadmap. This transfer can be challenging to think through initially, so we've outlined five key steps and a few extra tactical best practices for your feature-less roadmap.

Start with the strategy.

If you're not sure where to begin, your overall corporate strategy is an excellent starting point. Begin by working backward from the ultimate objective. It's tough to build a feature-less roadmap if you don't understand the 'why' in the first place. The first phase in moving to feature-less is to review the features on your roadmap. Ask yourself, "Why are we doing this? To what end? What drove getting those items prioritized in the first place?"

You need that perspective first; then, it becomes just a 'pivot' of the data around your preferred feature-less roadmap format.

Do the deep work upfront.

Without an apparent problem, you will struggle to identify the solution. Think backward before adding the solution (or feature) to your roadmap. What will it be like if your team succeeds in the next year? How will your customers feel about what they are able to do?

Do the deep work upfront and let it carry you, rather than continuously validating something that doesn't have the initial work. If you look at the existing objectives, you can work backward and translate them into one of the feature-less roadmap formats we've reviewed.

Be specific, but not specified.

Set realistic expectations around your roadmap. While themes, OKRs, or whatever format you choose to use are far broader and high-level than particular features, they should still

be defined using a specific language. The intent isn't for them to be a catch-all, but rather one specific bit of customer value you're adding to the product.

That way, your roadmap is free to change to your environment. With features as a secondary element, you can reduce the scope or increase the timeline without needing to address a complete change of direction.

Avoid solutions.

Don't douse creativity by jumping to solutions. Although it might feel like you need to do everything for your team–that's not the way to be the executive of the product. User research isn't just a part of product design but a requirement of product strategy. If you track and synthesize different problems, it's a different way to talk about the product you are building.

Stay away from dates.

When you place firm dates on your roadmap, particularly for items not slated for development until several cycles down the road, you're setting up your company for disappointment. When you decide to include dates on your roadmap, you want to make sure to build the roadmap in such a way that you can change or remove them when circumstances change and those dates are no longer reasonable.

Iterate your roadmap.

Of course, feature-less roadmaps shouldn't be set in stone, and neither should the priority of the desired outcomes. Things will change. Acknowledging that inevitability by frequently revisiting and refreshing the roadmap based on the latest learnings and updated organization-wide goals gives the pursued outcomes a better chance of reaching fruition.

Other Feature-less Roadmap Tips:

- **Use a roadmap software**

  Intentionally designed roadmapping software helps you make a visually compelling case for your product strategy (which can be extremely useful when supporting your team's transition from a feature-based roadmap to a feature-less roadmap).

- **Use color**

  Color is a great way to represent how your roadmap ties to the product vision or strategic objectives. Color-code each item on your roadmap to help people make the connection between each initiative and how it fits into the big picture.

- **Use large fonts**

  People have a limited amount of time to digest your strategy, so use large fonts, especially if you are presenting your roadmap on a projector or in an online meeting.

- **Keep it high-level**

  Remember that you are telling a story about how your strategy fits with the product vision. So say the story in big, bold strokes rather than minute details. If you can, create logical groupings of initiatives to make the roadmap easier to grasp.

- **Add data**

  In a meeting where you're presenting your roadmap, if you don't have the evidence at-the-ready, you can lose that all-important momentum. If you have that data accessible to anyone viewing your roadmap at any time, it can make a significant difference in giving credibility to your strategic plan.

5

UNLOCK YOUR
PRODUCT'S STRATEGIC
POTENTIAL

5 UNLOCK YOUR PRODUCT'S STRATEGIC POTENTIAL

Ultimately, feature-less roadmaps are all about solidifying what the organization is trying to achieve, communicating those objectives, and building consensus and momentum. As you work within your feature-less roadmap, remember the big picture. There needs to be a strategic justification accompanied with every item on the roadmap. Although folks may think they care more about when Feature X will ship, what they're genuinely interested in is how Feature X will help customers save time, save money, or grow the company's bottom line. Fully vetted, socialized, and measurable outcomes will keep everyone's eyes on solving the right problems for your customers and product. Every single element of your feature-less roadmap strategy will reinforce your product vision.
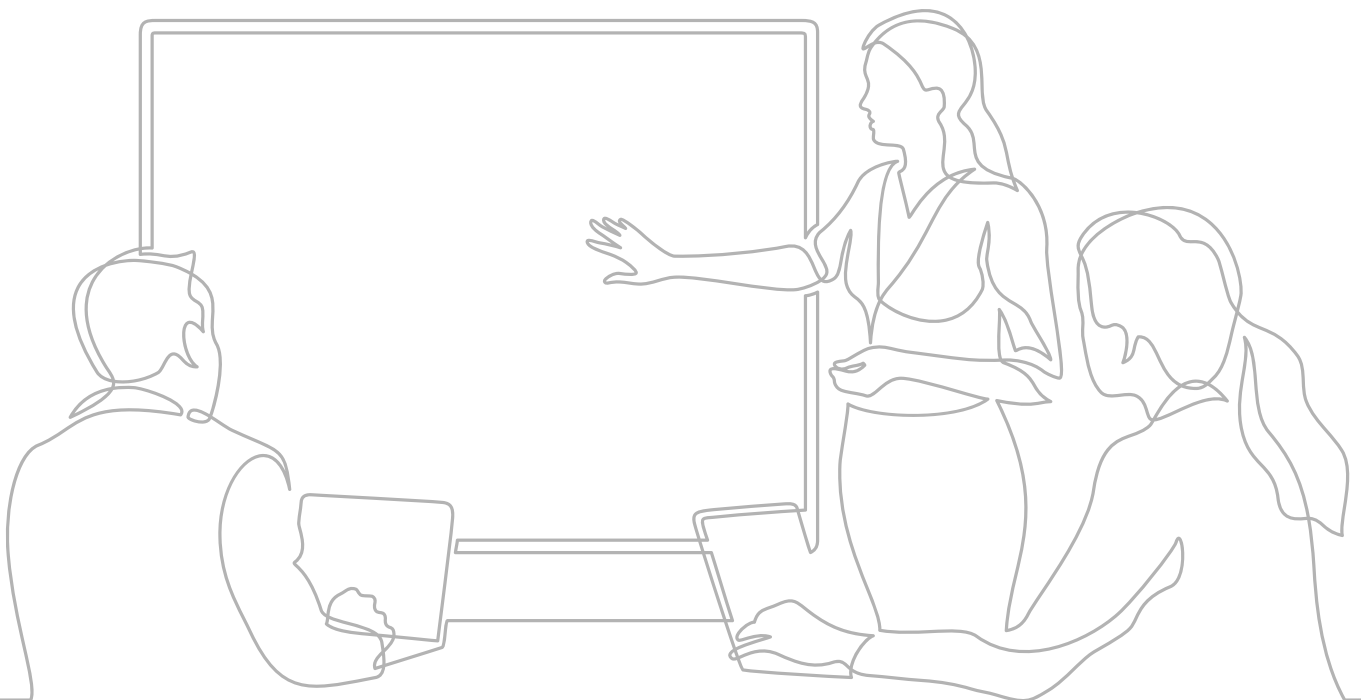
ProductPlan

ABOUT **PRODUCTPLAN**

ProductPlan makes it easy for teams of all sizes to build beautiful roadmaps. Thousands of product managers worldwide–including teams from Nike, Microsoft and Spotify– trust ProductPlan to help them visualize and share their strategies across their entire organization. With our intuitive features, product managers spend less time building roadmaps and more time shipping products.
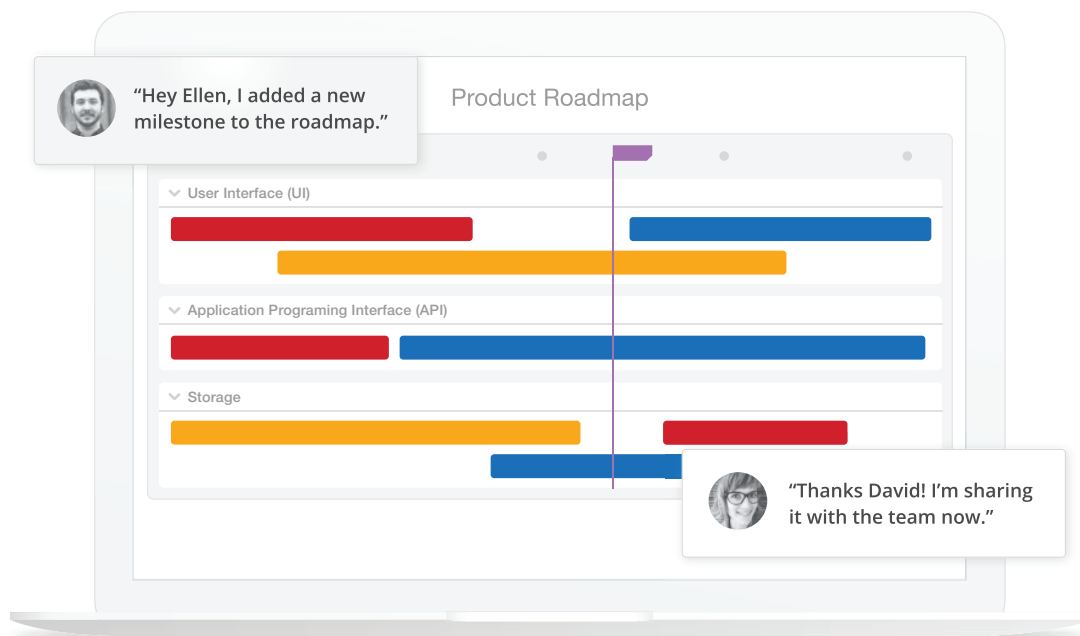
TRY IT FREE →

WORKS **CITED**

Cast, S. (2019, April 4) Putting it all Together—Creating a Lean Objective-Based Product Roadmap
Retrieved December 16, 2019, from https://www.prodpad.com/blog/creating-product-roadmaps/

Cutler, J. (n.d.) Grounding Your Product Roadmap with Themes and a North Star.
Retrieved November 19, 2019, from https://www.productplan.com/roadmap-themes-north-star/

Fitoussi, V. (2019, September 19). Feature-less Roadmap: The Balance Between Delivering Concrete Features vs. Planning with High-level Themes.
Retrieved October 29, 2019 from https://amplitude.com/blog/feature-less-roadmap

Juncal, S. (n.d.) How are North Star Metrics Leading You Astray?
Retrieved November 19, 2019, from https://www.productplan.com/north-star-metrics/

Kirsch, M. (n.d.) Reasons Product Roadmaps Fail.
Retrieved November 18, 2019, from https://www.productplan.com/reasons-product-roadmaps-fail/

McCloskey, H. (n.d.) How to Use Google's OKR Framework to Prioritize Your Roadmap.
Retrieved December 18, 2019, from https://www.productplan.com/prioritize-product-roadmap-with-okrs/

Murphy, L. (2019, November 15) Customer Success: How Innovative Companies are Reducing Churn and Growing Recurring Revenue.
Retrieved December 28, 2019, from https://docs.google.com/document/d/1K4QHfc2cqq0IOCbKTJnqyvb7s3godWwpw4GY4KxdfyY/edit

Rex, A. (2018, October 23). Escape from the Feature Roadmap to Outcome Driven Development.
Retrieved December 20, 2019, from https://www.mindtheproduct.com/escape-from-the-feature-roadmap-to-outcome-driven-development/

Saez, A. (2017, June 10). How to Build a Roadmap Everyone Understands.
Retrieved November 18, 2019, from https://www.prodpad.com/blog/how-to-build-a-product-roadmap-everyone-understands/

Semick, J. (2020, January 7). Product Managers in 2020.
Retrieved January 8, 2020, from https://go.productplan.com/2020-product-management-report/

Unknown. (n.d.) What is a backlog?
Retrieved December 28, 2019, from https://www.productplan.com/glossary/backlog/

Unknown. (n.d.) What is the Jobs to be Done Framework?
Retrieved December 18, 2019, from https://www.productplan.com/glossary/jobs-to-be-done-framework/