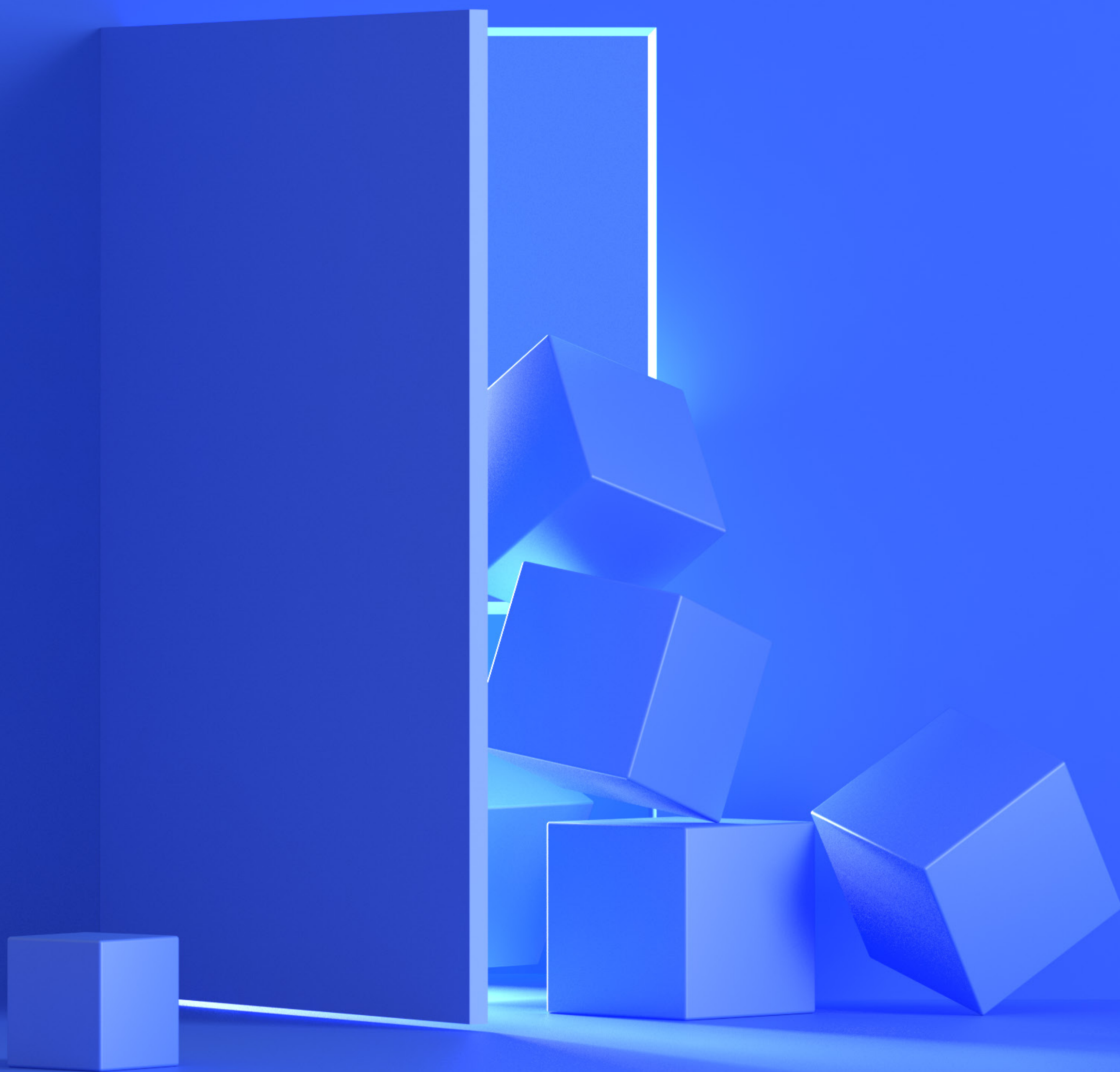


The Ultimate Guide on
**Avoiding
the Feature Factory**



by **airfocus**

Table of contents

- Why we wrote this ultimate guide 03
- What does it mean to work in a feature factory? 06
- Why do product managers or companies fall into the feature factory trap? 18
- How to transition from a feature factory to a value generation factory 25
- Going from feature factory to continuous discovery in 1 year 40
- How to know if you’re working in a feature factory..... 46

PS: If you are in a rush and would like to try airfocus for free today then click here to [get your 14-day trial version](#).

Why we wrote this ultimate guide



If you've come across the phrase "feature factory" you may not have known what it meant at the time. However, one thing is for sure, it's not a praiseworthy one.

When you look at companies like Apple, Facebook, Netflix, though their products have a multitude of features, their features were well thought out prior to release. And even for products with multiple features, product teams sunset features when they are either no longer performing as expected nor align with strategic objectives.

Features aren't simply placed into products for the sake of it, to match competitors, or because customers asked. They are placed in products because they help solve a key goal that customers and users want to accomplish.

A feature factory is a company that consistently releases products, features, enhancements, etc., and predominantly focuses on quantity over quality.

Companies don't become feature factories overnight, however, for companies that are feature factories this speaks to a broader problem related to their strategy and product development process.

How can you know if you work within a feature factory? Ask yourself the following questions or check the [blog post from John Cutler](#).

- Can you tie your features to outputs?
- Do you actively collect customer feedback?
- Can you create roadmaps at the output level, and tie your features to it?

If the answer to any of these questions is "no" then this is a sign that you may be working within a feature factory.

If you find similarities with this definition and your work realize that your company is on the path to becoming one (and want to transition away from it), or simply want to ensure that your company does not become one in the future, then read this guide to completion.

Tons of valuable information on this topic you can also find in Melissa Perri's book "[Escaping the build trap](#)".



Some of the topics that are covered in this detailed guide include:

- Identifying the feature factory and why it's a problem
- Understanding how and why companies fall into this trap
- Signs to understand when you're working in a feature factory
- How to transition from a feature factory to a value generation factory
- Communicating effectively with stakeholders with the right tools
- Using continuous discovery to deliver value to customers and users

What does it mean to work in a feature factory?

The term “feature factory” was developed by [product expert John Cutler](#) after listening to the woes of a friend of his, a developer, complaining about the practices of his software company.

While mentioning his issues he said that he felt that he was working on an assembly line at a factory rather than a software company.

He felt this way because his company was continuously shipping out numerous features that were not well thought out.

“Feature factory” is not a praiseworthy adjective to describe a company. And skilled product managers who want to hone their product practice, along with product people in general (designers, developers, etc.) who take their role seriously, generally avoid them with a ten-foot pole.



A feature factory is a company that consistently releases products, features, enhancements, etc., and predominantly focuses on quantity over quality.

They focus on output rather than outcomes. And as they continue to release features, they encounter further problems because their solutions were not well thought out.

Rather than taking the time to understand [what problems need to be solved](#), why, for whom, testing with hypotheses, and constantly testing their solutions to ensure that they’re accomplishing the right goals, the focus instead is on [output](#).

Besides the fact that this is not how great products and companies are built, why is this bad? Can a company not release multiple features until they figure out which ones stick and then focus on those?

No. While this strategy can be employed, besides the fact that this wastes time and resources, being a feature factory leads to **a multitude of other problems** as well.

Here are some of them:

Poorly implemented features

Feature factories focus on [output, not outcome](#). Rather than taking the time to ensure that features have been validated with customers prior to release and rigorously tested, the focus is on getting the features out as soon as possible. So customer development is put to the side.

Customer development is the practice of building with your customers, constantly returning to your customer base to validate your assumptions, presenting your work to date, while **gathering and implementing feedback** for improvement.

One key benefit of customer development is that by the time features are launched not only will you have ensured that they meet the needs of your key stakeholders, you will also ensure that they work as expected. Which will only further support adoption upon release.

Products riddled with bugs

When the focus is on output, shortcuts are made in the **QA process**.

Whether this is via beta testing, QA testing, or more, these steps are often skipped. And if performed, perhaps quickly to ensure that features are released on time.

Quality takes the back seat.

While the deadline may be reached for release, the side effect of this is that the product continues to be riddled with bugs.

This leads to your development team spending even more time fixing bugs on the product post-release. Time that could have

been better spent had a focus been placed on quality prior.

Lack of understanding of one's target market

One of the key symptoms of a feature factory is the lack of understanding of its target market.

Market segmentation is useful for understanding the needs of your target audience.

Your target audience is the specific group of people that your products are catered towards. This does not mean that others can not use your products, rather it helps you focus and prioritize which problems to solve and for whom.

Not understanding your target market leads to wasted time and resources.

While you may assume that you are providing the right features and services, in fact, you are not. And what will eventually happen is your customer base will start to look elsewhere (at competitor products).

Take the required time to understand who your target customers are and then build according to their needs.

[Jobs To Be Done \(JTBD\)](#) and the [Value Proposition Canvas](#) are great tools to help accomplish this.

Enhancements that don't add much value to the product

When features are poorly implemented, riddled with bugs, customer development is not practiced, and you don't understand your target market, this only leads to enhancements that do not add much value to your product(s).

A company may brag about the many features within their product, however, it can be possible that it's failing to make their customers' lives incrementally better with each release.

If anything, it may be evoking increasing negative emotions; no one enjoys using products that are riddled with issues.

Limited oversight on management of features

Since it's often the case that feature factories tend to implement multiple features at once, in their desire to get out as much as they can with the limited time available there is often limited oversight on managing these features.

Not only is there no formal crystalized process on determining that the right features are built, but even post-launch, who reviews the released features? Who reviews the metrics? Who [collects customer feedback](#) to ensure that features are meeting customer needs? Who determines the next steps for these features?

Rather than ensuring that the recently released features are managed effectively the focus is directly placed on getting the next set of features out the door. And the cycle continues.

Lack of retrospectives

Though popular in Agile but also used in other software development methodologies, [retrospectives](#) are a great way for teams to band together and discover opportunities for improvement.

Teams meet at the end of a sprint, post-release of a solution, or at any other time they deem beneficial, to discuss 3 key things:

1

What went well?

2

What went poorly?

3

What are the areas of improvement?

“Product managers do not conduct regular retrospectives on the quality of their product decisions and compare expected benefits to actual benefits. Developers have “passing tests”, but product managers do not. Product managers view velocity and output as their key performance indicator”

John Cutler

While teams are constantly focused on shipping features, retrospectives get missed which only leads to poor communication, recurring problems, and eventually lower team morale.

Ignoring metrics

If you want to understand how your customers and users are using your product then look at your [metrics](#).

Paired with qualitative data, product leaders use metrics to understand how their products are performing, roadblocks, and make objective decisions.

In many cases, companies that are feature factories ignore their metrics. If they didn't they would be more conscious about the features they built.

Lost market share

When the focus is simply on output and not value created for customers and users, then eventually you will lose to your competition that does focus on deeply understanding the problems that customers and users face, prioritizing them, and releasing the right solution(s).



A product is not successful based on the number of features it has, but rather based on whether it can solve the right problems for the right people.

As your team is focused on shipping new functionality quickly, functionality which may not even be desired or needed, other companies are focused on shipping the right functionality.

Feature bloat

[Feature bloat](#) is when a company builds more features into a product than needed.

The result of this is that the software can become too large, slow, confusing to use by customers and users, and ultimately more difficult to manage in general.

This is definitely not wise for companies that want to continue to be innovative and remain competitive.

And maintaining the product will continue to be challenging as new members join the team (unless there is documentation and other collateral to support new hires).

With this list of problems that feature factories face, why would business leaders choose this path? They don't, it's not a conscious decision that they make.

They fall into this trap and continue on because they may not know any other way exists and/or how to get out of it.

The good news is that there are specific steps that can be taken to escape. However, more on this later. Before we get there let's dive into another very important related topic, predictability in software development.

Predictability in software development

Productivity in software development is how much a team can get done, whereas predictability is knowing what the team will get done and when.

Both productivity and predictability are challenges that many software companies face.

Knowing what will be done, how, and when; predictability, and getting things done faster; productivity.

Why is predictability important?

As much as you may plan every item of work to be accomplished for an initiative, unintended events still occur. However, as the saying goes: proper preparation prevents poor performance.

Not simply due to a project-management mindset of knowing when to deliver, but predictability ties directly into product strategy.

Software companies release products at specific times, sometimes based on market conditions and events, and knowing when products and features will be released is also important for strategic planning.

Predictability helps teams:

- Hire when needed
- Better coordinate their teams
- Prepare their marketing plans
- Make budgeting considerations
- Align initiatives with business strategy

While teams struggle with how to go fast, productivity, before you go fast you need to understand where you are going and why;

know what needs to get done first and then accelerate.

How exactly can teams gain more predictability in their software development practices?

Here are some steps you can take to accomplish this:

Forecasts rather than estimations

Leverage forecasting rather than specific dates for when the new functionality will be available.

While you can plan, unintentional events may occur which can cause items to take longer than planned. For this reason, providing stakeholders with a rough estimate of when they will receive new features and enhancements may be a better approach than a specific date.

This approach can improve stakeholder communication. While there may be negative consequences if internal deadlines are missed, trust is eroded when deadlines communicated to customers are missed.

We spoke about how to manage customer communications and expectations around releases in our detailed guide [Roadmapping from A to Z](#). Check it out!

Leverage Agile

Whether Scrum or Kanban or another methodology, leverage [Agile](#).

Agile assists teams with being more predictable while responding to changing market conditions, ensuring that customer needs are at the forefront, and regularly delivering value.

With the amount of time taken initially to understand a project scope wouldn't [Waterfall](#) be a better option?

Not necessarily. As we mentioned in [The Ultimate Guide to Agile](#), one downside of Waterfall software development is that **it does not enable you to make quick changes based on market conditions.**

Break down large items into smaller tasks

Smaller tasks that can be worked on and tracked regularly to understand how the team is tracking towards completion. This is where [user stories](#) and [acceptance criteria](#) come into play.

User stories are used to communicate value that should be delivered to users and acceptance criteria are used to test and ensure that the delivered work is working as intended.

Sprint planning meetings

What specifically needs to be completed in specific intervals? And how much work can your team actually get done based on their availability?

Utilize sprint planning meetings to plan and understand what your team can accomplish within given sprints. And if you don't follow Scrum then leverage a similar meeting where your teams meet regularly to plan upcoming work.

Daily standups

Daily standups are not limited to Scrum, they can be used for any [software development methodology](#).

Take advantage of them.

In this meeting members of the implementation team get together every morning to discuss 3 key things:

- 1 What did you work on yesterday?
- 2 What are you working on today?
- 3 What work are you blocked on?

These meetings give team members visibility of the work being performed, measure progress, and address blockers that may impact stated deadlines as soon as they can.

Again, making the team aware of what's been planned and ensuring that they can accomplish stated goals.

Estimations

When Agile is leveraged, work is estimated to understand how long it will take.

Rather than long detailed project plans in the initial phase, used in Waterfall, the team meets to understand the scope, but also regularly throughout the project duration, [story time](#) and [sprint planning](#), to estimate the upcoming work.

This clarifies to teams how much they can get done and can provide a more accurate timeline on when new versions can be released.

Product demos

What do product demos have to do with predictability? Simply the fact that it makes teams and internal stakeholders aware of what will be released.

In these meetings, they will directly see work coming to life. This aids in avoiding any surprises upon release.

Leverage the right product tools

Leverage the right product management tools for an overview of what exactly is happening with your products.

Many teams want a 360-degree view when development begins, however for more predictability you should view the entire process.

From the feedback provided to prioritized initiatives, all the way through to completion of development, have a complete view of every step of the process to understand what is happening, why, when, and the expected release date.

[airfocus](#) is an easy to use modular product management platform. With airfocus you can manage your product strategy, understand customers' needs, prioritize what to build next, and create clear roadmaps to communicate priorities with your teams and stakeholders.

Output vs. outcome

A feature factory is a company that consistently releases products, features, enhancements, etc., and predominantly focuses on quantity over quality.



Feature factories have one key problem among others, that is their heavy focus on outcome over output.

Their main concern is on the number of features that are released rather than the value generated for their business, and for their customers and users.

Are there times when companies can focus solely on output? Yes, when they have tested and validated their assumptions and need to get defined features out of the door as quickly as possible.

What is the difference between output and outcome? And why is focusing on outcome so important?

Output is an activity taken whereas outcome is the result of that activity.

For example, releasing a new feature is an output, whereas customers being able to accomplish their key tasks within a product 50% faster is an outcome, it's a result of a set of improvements that were released.

How does a company make the shift? To begin, start with the end in mind.

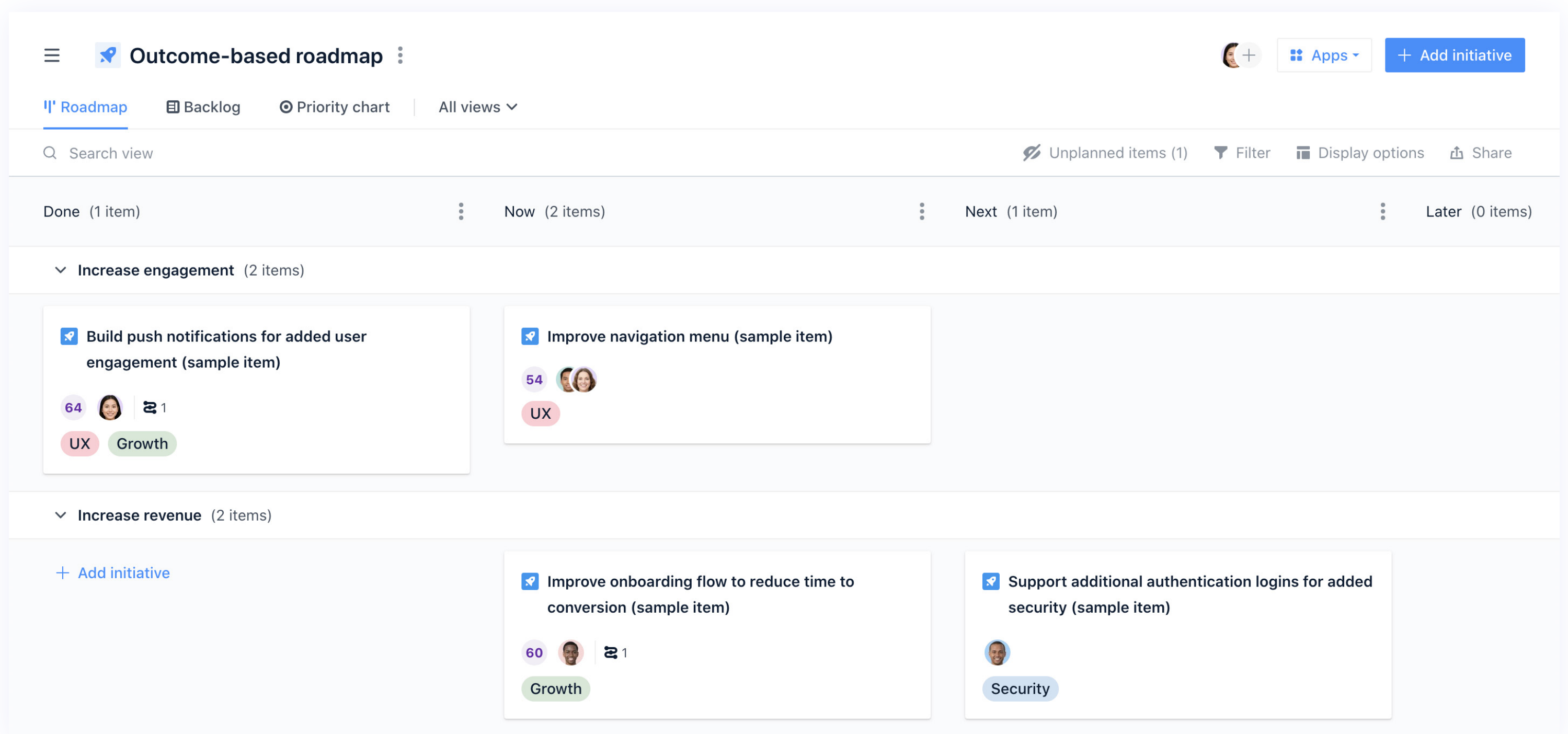
Starting with outcomes is easier when you understand what goals your business should accomplish and who your target customers are.

Start by identifying the objectives of your business and the key results (OKRs).

This is why OKRs are so powerful. They provide a framework for companies to ensure that they are focusing on the right things at the right time.

And once the OKRs have been set, everyone, yes everyone, in the company can feel free to question the work performed to ensure that they align with the OKRs. Especially if there is full visibility among team members on what OKRs are.

The majority of product management tools in the market continue to work with features at the center, providing a tool to collect feedback and create roadmaps at the feature level without providing the ability to tie it to outputs.



Set the company OKRs > set the departmental OKRs > set the product OKRs

This is the old approach of product management. If you want to escape the feature factory we suggest avoiding these tools.

Use airfocus to create an [objective based roadmap](#) and give everyone in your company full visibility into the goals to be accomplished.

And from these OKRs come the initiatives that product teams will focus on to accomplish their goals.

You need a product management tool that provides this level of hierarchy. When speaking to vendors ask them to show you how everything can be linked together.

Successful companies base their decisions on outcomes, not on outputs. Basing decisions on outcomes will enable you to modify your processes and approach to accomplish stated goals.

For example, is the concern the number of prospects your sales team has called, or rather how many deals they have closed?

Based on their close rate, a team may modify and enhance their approach to support these outcomes.

This is why it's important to measure key results, but more on OKRs later in this guide.

If there's one thing you should take away from this section it's the following:



To transition from a feature factory to a value-generating factory start by focusing on outcomes rather than outputs.

Why do product managers or companies fall into the feature factory trap?

If being a feature factory is so bad, why do so many companies fall into the trap?

This happens because they either started without understanding their specific goals and understanding their target market, or they initially did and lost sight of what their objectives were along the way.

Some product management tools also provide a feature approach and usually force the teams to work at the feature level. If your solution forces you to work at the feature level, maybe it's time to consider shifting to a more modern product management platform.

Falling into the feature factory trap can happen to any product team or company, and this is why it's important to stay focused and understand the signs that this is occurring.

However, if your company has become a feature factory or is on a path to becoming one, there are ways to remedy this.

How to avoid the feature factory?

Avoid the feature factory rut with the following steps.

Focus on problems, not features

It's extremely important to gather customer feedback.

However, when customers provide feedback, not all feedback should directly turn into new features.

Features are functionality within products that help customers and users accomplish specific goals. However, when feedback is received you should not go from feedback to feature right away.

Rather, aim to understand the specific problem that customers and users are facing. And from this determine how many other customers have mentioned the same, why, and determine if solving this problem aligns with your company and product goals.

Imagine you are a Business-to-Business (B2B). Your customers make a request for specific functionality that they desire.

At first glance, you may think that they simply want one feature that meets their request. However, when you speak with them and dig deeper to understand the problem it will become clear that they're looking for something else.

With this information, you can then modify your product accordingly while considering all of the meaningful changes that need to be made to accurately solve this problem.

This is why it's imperative to perform further research and validation before your team gets to the drawing board.

These can include:

- 1 Analyzing existing data
- 2 Customer interviews
- 3 Focus groups
- 4 Surveys
- 5 Usability tests

Move away from what customers want a feature to do and instead focus on the problems they need solving.

It's imperative to have a product management platform that provides the ability to tie feedback to problems, not features. If this is not the case you might want to consider a change.

One way to accomplish this is by using air-focus. With the [Insights app](#) you can centralize all feedback in one place, be it chat, email, or manually, to uncover actionable insights. Link feedback to ideas and opportunities to inform your discovery while keeping customers at the center of your product.

Focus on goals before determining features

As your team thinks about the problems worth solving, also consider the goals to be achieved as well.

When the goals are set this will help you determine which problems to tackle and then inform the potential features to build.

Again, this will help shift the mindset from simply building and launching features to being outcome focused; setting the right goals and ensuring that they're achieved via the initiatives pursued.

Foster a culture of asking the question “why?”

When objectives and key results are set for the company and made transparently it enables everyone in the company to continuously ask the question “why?”.

When teams begin working on new features, they should always be challenged on the why. This includes:

- 1 Why are we working on this?**
- 2 Who asked for this?**
- 3 How does working on this help us reach our objectives for the quarter/year?**
- 4 Does this initiative raise any risks that we should be aware of?**
- 5 Is this the best thing to do at this moment based on other priorities and how have we validated this?**

Product leaders will be more conscious of the decisions they make when they know that they may be challenged on their decisions.

This isn't just about the product team, however, but the entire company culture.

The moment team members start to challenge features, respectfully, and they are aware they will be challenged, then things will change.

Celebrate outcomes

If your team has the practice of celebrating outputs, change this to celebrate outcomes. Of course, this can only be done when outcomes are clear.

This is why it's imperative to set goals for products and features. Set the goals, track the metrics, and as they are met communicate and celebrate them with your team.

For example, if the goal is to increase your [Monthly Recurring Revenue \(MRR\)](#) by 20% within six months of launching a new feature, celebrate with your team when it's accomplished!

Your team will be happy that they met the goal, understand the steps they took to meet it, and this will help shift the culture towards accomplishing outcomes and not outputs.

Have visibility of the hierarchy

Having an overview of objectives, KPIs, initiatives, and features within the product management stack is crucial to ensure that the team maintains the right mindset (outcome-focused).

Your product management solution must give you the ability to tie different levels together and provide different views of your hierarchy.

This is also a tremendous aid when speaking to stakeholders at different levels. This way you can ensure that your team keeps the objectives in mind even when building features.

Project-oriented mindset vs. product-oriented mindset

There is a shift in mindset taking place among business leaders from being project-oriented to product-oriented. A trend visible among IT teams especially.



A solely project-oriented mindset can lead to a feature factory.

Organizations that have historically been output-driven are now leveraging a product-oriented mindset approach to their work.

With a project-oriented mindset, the main concern for teams is delivering their scope of work according to set timelines. This is often seen in feature factories; there are a list of features that need to be shipped within certain time periods.

A product-oriented mindset is more output focused, however. While timelines are considered, the key concern teams have is delivering the right capabilities to their customers and users. If this means that timelines have to be modified as a result, release dates pushed further out, then they are.

The outcomes to accomplish drive decisions rather than solely ensuring that projects are completed according to a defined project plan.

Here are some of the other key differences between a project-oriented vs. a product-oriented mindset:

Success is solely based on value generation

Teams with a project-oriented mindset define success by completing a project according to time and resources allocated. With a product-oriented mindset, success is solely defined by how much value is generated. Nothing else.

Teams and companies that typically have a project-oriented mindset often overuse the timeline-based roadmap which communicates what is going to be delivered and when.

This leads to the issue of stakeholders interpreting this roadmap as a project plan, which they shouldn't. The timeline-based roadmap is not inherently bad in and of itself, rather it's the way in which it's used and its frequency which can lead to problems.

Focus on the entire product lifecycle

With project management, there are defined timelines that teams focus on accomplishing work. This leads to teams focusing less on performance once initiatives are complete.

With a product-oriented mindset, however, there is a focus on the entire **product lifecycle**, which includes regular upkeep and improvement of the product over its entire life.

Product-oriented mindset leads to more effective prioritization

With project management the focus remains on delivering projects according to the defined scope and requirements, a product-oriented approach employs constant hypothesis testing, roadmaps, and focuses on value provided by the various features to key stakeholders.

This means that the scope can and only will change when needed, as long as there is a clear and beneficial reason why.

More stable team structure with a product-oriented mindset

With a project-oriented approach, individuals work on many projects at once and there is a likelihood of reassigning them mid-project.

A product-oriented approach utilizes cross-functional teams to manage one product at a time with permanent teams. This leads to more focus on the product they are working on as well as stability.

More aligned with business users with a product-oriented approach

When project management defines the scope, builds it, and then delivers it months later without any further interactions with business users, this leads to a poor understanding of their needs and a solution that is less likely to be adopted.

A product-oriented approach calls for continuous interactions with business users allowing them to refine the product to their specific needs and helping shape better results. Frequent interactions with stakeholders support success.

Leverage internal product management practices to truly align with business users and deliver solutions that accomplish key business and stakeholder objectives.

To learn more about this topic check out our detailed guide [Internal Product Management in Digital Transformation](#).



A project-oriented approach focuses on outputs, whereas a product-oriented approach focuses on outcomes.

Focusing on outcomes is what enables businesses to thrive, as they focus on modifying their plans to ensure that their goals are met.

If you want to see more information we would like to recommend you to watch the video from this article where John Cutler presents the topic [“How to Break Free of the Feature Factory”](#).

How to transition from a feature factory to a value generation factory

Curious to know whether you work within a feature factory?

We'll dive deeper into this later in the guide but here's a checklist to assess whether your company potentially fits the definition.

If you answer yes to one or more of these items then it's a sign that you work within a feature factory.

- 1 Productivity worship**
- 2 Zero retrospectives**
- 3 Not leveraging Objectives and Key Results (OKRs)**
- 4 Sprint goals are nowhere to be found**
- 5 Rarely sunsetted features**
- 6 Features are not enhanced upon release**
- 7 Minimal, if not any, customer discussions**
- 8 Lack of rigorous prioritization**
- 9 Feature bragging**

- 10 Solely leveraging feature-based roadmaps with a timeline plan**

- 11 Handing off work to teams**

- 12 Prioritization of upfront revenue**

Do one or many of these apply to your company and team?

If so, don't worry. We would not speak about this topic in detail without providing solutions to create teams that deliver impact and consistently generate value for customers, users, and key stakeholders.

We spoke a bit about this earlier in this guide, but let's discuss in more detail how to transition from a feature factory to a company that delivers consistent value, a value generation factory.

Six practical steps to avoid the feature factory

Here's a checklist of practical steps that you can start acting upon to escape the feature factory.

1

Shift the focus to outcomes over outputs

Feature factories focus on output, and how much gets done. The amount that gets done isn't always relevant if the right initiatives are not being prioritized and the right objectives achieved. So shift your focus from speaking and reporting on the amount of work done to rather what has been/will be achieved once the right work is done.

2

Leverage Objectives and Key Results

This goal setting framework enables teams and companies to understand what specifically they are going to work on, why, and expected outcomes.

Once OKRs are in place everyone in the company should be aware of what they are and can respectfully question how proposed initiatives tie back to the established OKRs.

This is one of the key pieces of advice we can provide to avoid the trap so more details on this topic soon.

3

Leverage outcome-oriented roadmaps

Outcome-oriented roadmaps are strategic tools that not only define what initiatives a company will pursue to improve its products, but also map the initiatives to the OKRs of the business.

So to get started with this you first need to know what the OKRs are. Rather than a feature-based roadmap with timelines that communicate what will get done and when, use a roadmap that helps you think and communicate expected achievements from prioritized initiatives.

So make sure that your product management platform allows you to build an outcome-based roadmap. If this isn't the case then reconsider your solution. You shouldn't use a generic tool such as Powerpoint or Miro. We explain this in our 5-minute Guide on [Why you should use a PM tool for your roadmaps](#).

4

Establish a culture of asking “why?”

One benefit of everyone being aware of the OKRs is that they can now start respectfully challenging proposed initiatives. Is releasing a new proposed feature really going to assist with a defined objective, or does a different path make more sense?

Establish a culture among your team of asking these types of questions. And this starts by including key stakeholders in the OKR process and also challenging initiatives yourself, when needed.

5

Ruthless prioritization

Not everything can be done, nor should every interesting idea be pursued.

Companies and teams have multiple ways in which they prioritize initiatives. However, despite whichever method you follow, you should leverage a product management tool that enables you to gather feedback from key stakeholders and also customize frameworks to fit your team's needs.

Focus on the right things rather than everything. So once the objectives are defined, and as you work with your team to establish a culture of asking “why?”, the next step is to ruthlessly prioritize; only focus on initiatives that make the most sense for your business to reach stated objectives.

6

Gather feedback to understand which problems to solve

This should be an ongoing activity, continuously gathering feedback from key stakeholders, including customers and users, with the purpose of understanding which problem to tackle.

The key here to transition away from being a feature factory is to ensure that the feedback is directly tied to objectives to achieve, and not to features to be delivered.

Using Objectives and Key Results (OKRs) to avoid the feature factory

Objectives and Key Results (OKRs) is a goal-setting framework used by businesses to align performance with business objectives.

Among its many benefits, they are great for ensuring that every department, team, product, and person in the company is aligned with the strategic goals of the business and the measurable results to work towards achieving these goals.

OKRs have two main parts: objectives and key results

Objectives are broad ambitious goals to be accomplished.

An example of this is becoming the market leading solution for x. This is the clearly identified goal to be reached.

While setting goals is beneficial, as the popular saying goes: “what gets measured gets done”.

Key results do just that, they are measurements of progress towards meeting the objective.

For example, a 30% increase in customer retention within Q2.

It's best practice to tie OKRs to a specific time frame, for example for the quarter.

Some advise that there should be 3-5 key results per objective. A minimum of two at the very least, while more than five may be too much.

Key results as they are measured are also graded by some, where 1 (100%) means that you have fully achieved the key result and a .7 (70%) is still good.

If one finds that they are constantly hitting their key results (constant scores of 1) then this means that they are not being challenged enough to accomplish their goals.

To learn more about this topic check out [How Google Sets OKRs](#).

How exactly do OKRs help companies get out of the feature factory trap?

At the beginning of the year business, executives set the overall OKRs for the entire business. Once complete department man-

agers then set OKRs for each department which ties into the business OKRs. This then factors into the product OKRs.

Think of it like a pyramid, the overall business OKRs affect all business activities, while all business activities should aggregate to the business OKRs.

Once the OKRs are set, only those initiatives which directly align with the OKRs should be pursued.

If an initiative does not assist a company or team with reaching objectives then someone can comfortably challenge why it's being pursued.

Feature factories are output-driven and companies that focus on OKRs are outcome-driven.

The objectives are set, key results defined, and as initiatives, projects, products and features are built and released, throughout the year the key results are measured to determine whether the objectives are being reached or not.

The focus is on outcomes, not on output.

So as your company thinks about the strategic initiatives for the next quarter, give good thought to the OKRs you want to

achieve. And as feature suggestions and ideas are gathered only work on the ones that relate to the OKRs.

airfocus makes it easy to develop and gain an overview of company wide OKRs with its ability to craft OKR roadmaps.

An [OKR roadmap](#) is a timeline based roadmap that lists the objectives and key results for specific time periods. It does not have to be limited to the OKR for one product or department, but can also list the OKRs of multiple products or departments.

When gathered in one view it acts as a central source for teammates to understand and track the OKRs for various products, team members, or departments.

Earlier this year our team gathered some of the industry's most renowned thinkers and writers on product OKRs to consider what it takes to set OKRs properly and use them effectively.

This resulted in a roundtable debate that covered the benefits of using OKRs for organizations, why using OKRs can help you escape the "feature factory" mindset, whether to use individual team member OKRs, and much more. Take advantage of the insights [here](#).

Using Outcome-Oriented Roadmaps to avoid the feature factory

Another key way to escape the feature factory is via Outcome-Oriented Roadmaps.

A [product roadmap](#) is a tool that outlines the strategic objectives of a business and how the product team and business plan to evolve their product to reach set goals.

Roadmaps have many benefits including:

- Aligning teams
- Positioning products
- Improving communication and creating transparency with stakeholders
- Setting the vision for the future
- Planning resources
- Providing an overview of the product portfolio

Roadmaps are not crafted in silos by product teams. Rather they are crafted with feedback from other stakeholders within the company.

This is because others in the company may have information and inputs that can be factored into decision making.

While internal and external roadmaps are two of the most common categories of roadmaps, there are multiple types of roadmaps including:

- Timeline roadmap
- Technology roadmap
- Product launch roadmap
- Experiment based roadmap
- ... and many more

One of the most impactful and useful types of roadmaps is the [outcome-based roadmap](#).

If you're trying to avoid the feature factory this is one roadmap you need to leverage because this roadmap enables you to perform the key action to escape the trap: mapping your work to outcomes and not outputs.

Outcome-based roadmaps clarify the expected results of roadmap initiatives by focusing mainly on Key Performance Indicators (KPIs).

A Key Performance Indicator (KPI), is a measurable value that aligns directly with the overall outcomes that a business hopes to achieve.

KPIs are used to evaluate how well a business and its members are achieving its targets. Check out our article on how to [set, measure, and track KPIs](#) to learn more about this topic, though we'll shed some more light on this in the next chapter as well.

An outcome-based roadmap starts with the product vision and then works its way down focusing on the specific initiatives to complete to see the vision through; start by understanding the outcomes and then what needs to be done to achieve it.

In sequential order, on an outcome-based roadmap, you will clearly see the product vision, followed by the intended outcomes with metrics to track, and what will be delivered as a result.

In between this, from outcome to features, you may see the opportunities to be pursued, product ideas, planned solutions, followed by features and [user stories](#).



This roadmap is helpful because it clearly communicates to stakeholders how delivered items tie directly to the product and company vision, and the outcomes to be reached.

Before you can get started with an outcome-based roadmap you need to be aware of your Objectives and Key Results (OKRs).

If you want to start trying to build an outcome-based roadmap, start with [this template](#). It will give you the foundation to transition from an output-focused feature factory to an outcome-focused value generation factory.

With airfocus roadmap templates, you can have different views of your outcome-based roadmap with a Table, Chart, Board, or Timeline view.

How to communicate during this transition with stakeholders

Stakeholders are key to success, both internal and external.

External stakeholders include your customers who pay for your products, and internal stakeholders include the people and teams that build products and provide services for your customers and users.

Transitioning out of being a feature factory is not a one person task, it involves a shift in mindset and culture from the entire company. This can be facilitated with the right frequency and means of communication.

Here are some things to keep in mind when communicating with stakeholders to transition from a feature factory to a value generating outcome-based factory.

Know who your key stakeholders are

A **stakeholder** is anyone who has a direct or indirect influence on a product.

Internal stakeholders are the individuals inside of a business. For example members of the senior management team, internal users, designers, and developers.

External stakeholders are those who can influence the product though they do not work within the business. These include customers, industry analysts, trade unions, and development agencies.

Both of these groups are important and will require different pieces of information and frequency of communication depending on who they are.

For example, internal stakeholders will require more information and details on plans, especially if they will be the ones implementing the work, while external stakeholders will want to know your company's key initiatives and when new features and solutions will be available.

This is why it's important to have multiple views within your product management platform, so you can communicate the right information to the right stakeholders.

airfocus supports multiple views when working in a workspace. Not only for an overview of your backlog, prioritization, initiatives, and more, but also for the ability to create and share internal and external views for various stakeholders.

This leads to more meaningful and impactful communication.

Step into the shoes of your stakeholders

As you communicate with stakeholders, put yourself in their shoes. For each stakeholder consider:

- What key information do they need?
- Is there a best time and place to communicate required information?
- Does the documentation that I am providing contain the information they need?
- Is there anything that I can do to make the key message clearer?
- Have I communicated this enough times for them to understand

Pay attention to and focus on the stakeholder(s) that you communicate with, not all are created equal. The method and pieces of information you share with business executives will differ from those shared with customers.

With customers, you want to share the vision of where your products are going and tie this into how it will make their lives better. With business executives however you want to help them understand the strategic initiatives and how they tie into their department and the company goals.

It's important to engage with customers to avoid the feature factory because regular engagement with customers enables you to understand their problems. And when you tie their feedback to problems, and not features, then you are in a better position to provide them with value.

Your product management tool should not only allow you to gather customer feedback and tie it directly to problems, which is imperative, but also provide an easy way to engage with customers.

The airfocus [Portal](#) is a branded customer-facing website that helps product teams:

- Share roadmaps, plans, and ideas from your workspaces and celebrate recent launches
- Gather feedback and new ideas seamlessly and make customer-centric product decisions
- Test and validate ideas and concepts with the crowd and improve your discovery process

Use the Portal to build products people want and close the feedback loop with your audiences, regardless of whether they're customers, teammates, or business stakeholders.

Provide collateral for reference

Once you've gotten around to creating an OKR roadmap, don't just talk about it, show and prove.

Share collateral for reference.

This does not simply include a link to a shared roadmap (which can easily be done with [airfocus](#)) but can also include summaries of customer discovery, design prototypes, presentation decks, and more. Especially when sharing the roadmap for the quarter (or year), it helps to craft and share a presentation deck as an aid.

Paint the picture for the future

Inspire your stakeholders (internal and external) by constantly reminding them of the mission you are working towards.

Go past simply releasing products and features, but why does your product exist? Why does your company exist? How do you see the world changing based on the fact that your product is available?

The reason this is vital for communicating with stakeholders is because it helps them see the future along with motivating and inspiring them to work towards it and support your products along your journey.

Frequent communication around intended outcomes of initiatives

When transitioning from a feature factory to a value generation factory when communicating with internal stakeholders the communication should be centered around intended outcomes.

While senior management is generally concerned about the progress of strategic initiatives, other internal team members require information on what is being worked on and why.

When speaking on initiatives to be pursued and progress constantly drive the point home on the problems that are being solved for customers, users, and the business and the intended outcomes that these initiatives should drive.

Frequent communication along these lines will lead to a shift in your teams' mindset to focus more on outcomes, rather than output. And hopefully, they will begin communicating their work based on intended outcomes as well.

Bring data to the table

Using data is important for escaping the feature factory because tracking and reporting on meaningful data will better enable you to generate value for your customers, users and business.

Feature factories do not pay heavy emphasis to their metrics, the main focus is on releasing features on time. When key metrics to generate value for customers, users, and the business are defined, understood, tracked, and constantly revisited, it becomes another tool that can be used to prioritize which initiatives to address, when, and why.

For any key decisions that are made it also helps to share the data that was used to arrive at the decision. This helps stakeholders understand why the decisions were made, that decisions weren't simply made based on opinions, and be bought in.

Using better solutions to collect customer feedback centrally

If you want to avoid the feature factory then hone into your customer feedback.

Gather it and start tying feedback received to customer and user problems to determine what to action upon.

Customer feedback is the most pivotal tool that product teams can use to help them prioritize, build, and sustain their products while working towards business growth.



Customer feedback includes suggestions on improvements, thoughts, experience and general sentiment.

Feedback can come in various forms, from different channels, and from many stakeholders. Among the many stakeholders that feedback comes from, the feedback from customers and users is vital.

Customers are stakeholders who trade their cash for a solution in the hopes that it will assist them in solving their problems (accomplishing specific goals). Users engage with products regularly and support businesses by sharing these products with their network when they are pleased with their experience.

Here are the reasons why customer feedback is vital to product success and business growth.

1

Customer feedback informs you of customer sentiments about your product

Rather than shipping features that you think your customers and users want, rather hone in on the feedback that customers are providing and tie this feedback to problems that they are facing and you want to solve. This will ensure that released features are offering value and creating impact.

The easiest way to know if a product is accomplishing customer and user goals is via their feedback.

It's important to tie feedback received to the problems that need to be solved and not features. Move away from this by using

a product management tool that gives you the ability to tie customer feedback directly to identified problems (not features).

2

Customer feedback leads to newly uncovered problems to tackle

Assumptions can only get you so far. Even after uncovering problems to solve for your target customers and users, when you speak to them for initial discovery or even post-launch of a new feature you will uncover further problems to solve.

This is what will keep you in business and ensure that you're constantly delivering value.

3

Customer feedback helps inform prioritization

Successful product companies don't work on everything at once, rather they prioritize.

One of the ways they prioritize is by understanding value vs. effort. When many of your customers and users are facing a particular problem then this gives you an indication of which problems to tackle.

One of the pieces of advice we gave in our guide [The Ultimate Guide to Customer Feedback](#) is to strive to understand the “why”.

Customers and users will ask for many things. When they provide feedback related to feature ideas, always perform further analysis to understand why they are making their request. This is how you get to the root of the problem they are facing.

And this is why it’s imperative to tie customer feedback to problems to solve, not features

Ask “why” to understand why what they are asking for will benefit them. Ask “why” to understand whether there is value for the rest of your customer base and business.

Understand the underlying reason why requests are being made. Try the [Five Whys](#) technique to get to the underlying cause of their problem.

The goal is to deliver value, not output. Value can be delivered by having outcomes related to solving their problem.

Product feedback can come from multiple channels and may be provided to different people within your company. This is why it’s imperative to have a simple tool to capture and centralize feedback and tie them to problems to solve. Otherwise, it will be a messy process to keep track of all of this information.

It’s difficult to make the right product decisions without having your feedback centralized in one location with supporting information.

These can include:

- Customer name
- Total ARR (Annual Recurring Revenue)
- Target customer
- Module
- Opportunity for additional revenue
- etc.

Collect this supporting information so that your team can make more informed decisions on what to act upon and how. This is important to be able to define what criteria makes sense to your business and be able to modify the information asked when recording feedback accordingly.

When looking for product management tools as the vendor if the platform is flexible enough to adapt to your specificities

Other teams (sales, customer success, any other internal stakeholder) should also document this information if feedback is channeled through them. This helps them understand the level of information needed by the product management team to make informed decisions.

While you might be tempted to record customer feedback manually (in a spreadsheet for example), as the amount of feedback that you receive increases it will become harder to manage the sheer amount of feedback you will receive from various sources while managing your feedback loop.

This is why a product management platform can help. They provide many benefits including documenting, centralizing, and sharing feedback with key stakeholders.

For example airfocus' [Insights app](#) allows product teams to centralize feedback from multiple sources in a flexible way. With it teams can uncover insights, tie requests to problems, and craft their product strategy.

Check out this [article](#) to learn more.

Going from feature factory to continuous discovery in 1 year

Leverage continuous discovery to transition away from being a feature factory.

And once this becomes a staple within your company and teams, maintain this practice once you have fully become a value generation factory.

Transitioning from a feature factory to a company that practices continuous discovery may be challenging, but it's possible.

Introducing process changes involves risks, new learnings, can be time consuming, and mistakes will be made along the way. However, nothing valuable comes easy.

Why is product discovery important?

Product discovery is the customer-centric process of determining whether or not a product or feature should be developed.

The discovery process can differ from company to company, or team to team, however, the goals are always the same.

Product discovery is performed for many reasons which include:

- Surfacing and documenting problems faced by customers and users
- Gaining the ability to step into the shoes of customers while having a 360 degree view of their experience with interviews and feedback from internal stakeholders
- Taking the required amount of time to understand customer needs (not rushing solutions)
- Verifying your own assumptions before crafting solutions to problems
- Gathering feedback from multiple channels
- Prioritizing which problems to solve and solutions to offer to deliver value and create impact

There are many benefits to product discovery, however here are two of the key ones:

Product discovery leads to better products

Product discovery enables you to understand specifically what customers and users are asking for and why.

Go past simply recording feature requests and understand the underlying reason why customers are making their requests, the severity of their problems, their willingness to pay for a solution, their experiences with other solutions, and more.

This additional information helps product teams build better products because they know the specific problems they are solving and how to craft their solution based on detailed information provided by their target customer base. This is only achieved when you focus on the problem rather than the feature.

Product discovery improves relationships with customers

Customers and users enjoy being a part of the product development process. They enjoy being reached out to, especially your most engaged customers.

Your most engaged customers are the most invested in your product. They have used your product for a long time and probably know your product in and out. They are pleased with your product and happy to continue to provide feedback as you continue to solve their problems.

And if you have customers with low engagement, speaking to them will help you understand why they are not as engaged and work towards improving their engagement. Either way, customers love being heard.

While product discovery is the process of determining whether something should be built, continuous discovery is doing this on a continuous basis. Continuously asking the question “what should we build” and performing research to inform this decision.

Making continuous discovery a habit within your company and teams

Now that we've clarified the benefits of continuous discovery (building better products and improving relationships with customers) let's outline how you can make continuous discovery a habit within your company and teams.

Some of this advice may be new and take time to fully implement, however, if followed it will lead to more impactful products, more value for your users and business, and an improved relationship with your customers and team members.

Have a discovery plan

Always make a plan for discovery work to be performed. In the same way that a development team plans their work into sprints, create a plan for the discovery work your team will perform with customers and other stakeholders.

This plan should outline the goals of the discovery, the number of customers you want to speak to, when, and the next steps. This will force your team to really think about the steps they want to take to obtain insights.

Make discovery a collaborative team effort

If there's only one person within your product team who focuses on customer discovery and has all of the information and context then there are improvements that need to be made.

Make discovery a collaborative effort. Rather than solely one product manager that manages and participates in discovery research, calls, and other activities, add a designer and an engineer to take part in these activities as well.

Feel free to also add members from sales and customer support every once in a while, or other team members. Don't add the entire team, they have important work to do. However other members can add further insights and support during discovery activities.

Build customer empathy

Make it a point to understand your customers and users to the best of your ability, so much so that you have a good picture of their daily routines, their key goals, the products they use, and their daily challenges.

This is what makes the Value Proposition Canvas so powerful. The Customer Profile gives you a great understanding of your customers while the Value Proposition helps you align your solutions to your customers needs, pains, and desired gains (focus on [Jobs To Be Done](#)).

If you can get to the point where you can step into the shoes of your customers and users to feel what they're feeling, then it will make customer discovery more impactful and the solutions you build much more valuable. And help you avoid the feature factory trap.

Build along with customers

When a customer makes a request do not record it, build the solution in a silo, and then release it.

Rather, build an MVP, consider a [beta test](#) with select customers, and then [release it to the world](#).

How does this tie into continuous discovery? When beta testing solutions you will discover further areas of improvement, whether issues to fix or further areas to enhance.

Go a step further and build with your internal team, consider an [alpha test](#) before your beta test.

Practice customer development

Customer development is the act of improving your product with your customers involved in the product development process.

This involves constantly returning to your customers and users to obtain their feedback and validate your assumptions. Rather than building a product from end-to-end, releasing it, and getting feedback, share the progress of your work across multiple milestones for feedback.

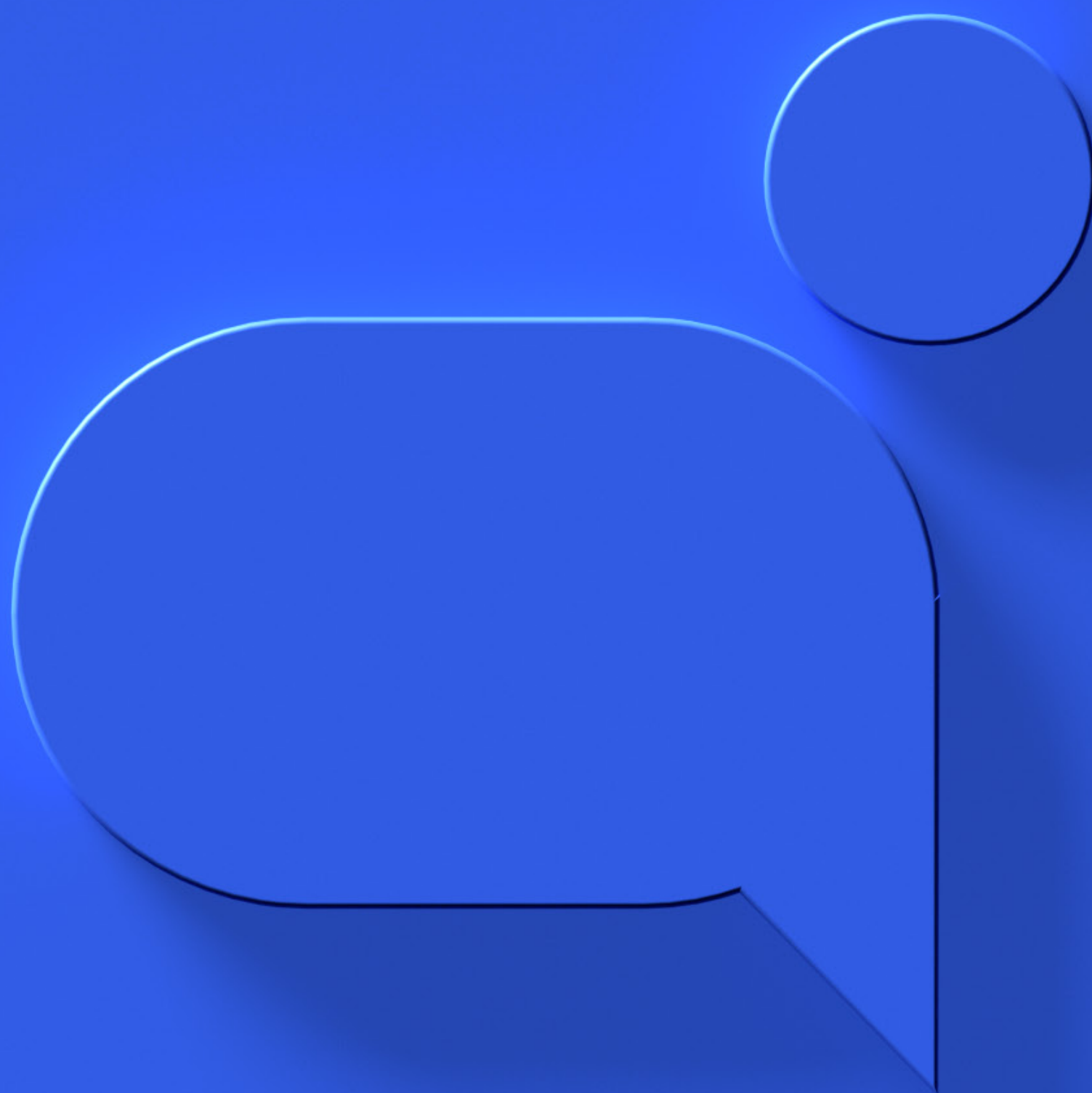
For example, once the wireframes are complete, meet with customers to obtain feedback. Once the hi-fidelity designs are complete, meet with customers to obtain feedback. Once a working version of the product is available, even if rudimentary, meet with customers to obtain feedback.

Along with improving your relationship with your customers, who in many cases will be happy to be a part of the process, this will help you validate your assumptions as you go from discovery to delivery.

Along with the fact that you will have validated that your product works the way customers intend upon launching your product, and gain more supporters to market your product as well.

Check out [The Role of Customer Development in Building a Winning MVP](#) for more on this topic.

How to know if you're working in a feature factory



Here are a few signs that you can use to identify if you are working in a feature factory.

John Cutler wrote a similar article about this topic and you can find his [original blog here](#). There are several other telltale signs you could be working in a feature factory. Some of them include:

1

Productivity worship

If you work within a company or team that is constantly focused on increasing productivity (or velocity). Then this is a sign that you may be working within a feature factory.

Don't get us wrong, productivity is important. However, when the focus remains on solely getting more done rather than ensuring that the right things are done then this is a cause for concern.

There's no benefit in moving fast if you're moving in the wrong direction.

2

Not leveraging Objectives and Key Results (OKRs)

Feature factories do not leverage [OKRs](#). If they did they wouldn't be feature factories.

OKRs is a goal setting framework used to align performance with overall goals in a measurable way. When using OKRs the process begins by setting broad ambitious goals and then key results to track against these goals.

For companies that leverage OKRs, prior to working on any feature or initiative they give thought to how it will assist them in accomplishing their objectives.

They also leverage a product management tool that allows them to tie their OKRs with initiatives and features so that they always have objectives in mind.

If pursuing the initiative will and it's the better course of action to take, among other competing initiatives, they move forward. And if it does not assist them in reaching their objectives they park it to the side and focus on something else that will.

Feature factories do not follow this process. Rather, feature factories simply ship feature after feature without giving much thought to how they tie into the overall strategy and objectives of the business.

3

Sprint goals are nowhere to be found

For companies who follow Scrum, if your teams continuously work without **sprint goals** then this is a sign that you're working in a feature factory.

Sprint goals are clear objectives for what should be delivered by the end of the sprint. They are meant to tie to the overall strategic objectives of the product, team, and company.

When teams don't utilize sprint goals then this is indicative of the mentality that the team should accomplish everything they can in the sprint. This is a sign of a feature factory.

Again, predictability is before quantity.

4

Minimal, if not any, customer discussions

Feature factories do not have regular discussions with their customers and users. Due to this they lack an understanding of their target customers and fail to build the right solutions for them.

Products are built to provide solutions to customer problems.

When product leaders have regular discussions with customers, focus on the why to discover the problems to solve rather than features expressed, and utilize customer development to build with their customers, then great products are built which solve key pain points. Rather than bloated products that may satisfy some customers but not exactly delight them and make products a staple in their daily lives.

5

Feature bragging

It's not about how many features your product has, but rather how impactful these features are.

Customers could not care less if your product has 20+ features. The amount of features within a product is not necessarily a competitive advantage.

If you're working within a company that constantly brags about the number of features they have released, and no one even remembers why these features were built, for who, and how they differentiate your product from competitors, then this is a sign that you're working within a feature factory.

6

Solely leveraging feature-based roadmaps with a timeline plan

A feature-based roadmap lists the features that will be released for a product. On it you will mainly see broad timelines, normally quarters, with lists of features that are expected to be delivered in these quarters to improve the product.

A timeline-based roadmap is similar to a project plan. Along with listing the strategic priorities for products, the timeline view communicates the priorities (first things first) and how long initiatives are expected to take.

These roadmaps serve as the central source for team members to understand what is in progress and when items are expected to be released.

These roadmaps are not inherently bad in and of themselves, however, it depends on how they are used and interpreted. Feature factories focus on these roadmaps, which features are coming and when.

Non-feature factories leverage strategic thinking and strategic roadmaps. For example an [OKR roadmap](#), an outcome-based roadmap, a timeline-based roadmap that lists the objectives and key results for each objective in a specific time period.

It's important to have a product management tool that lets you tie the different levels of hierarchy from objectives to features. If this is not the case then you should reconsider your product management solution.

The OKR roadmap does not have to be limited to the OKR for one product or department, but can also list the OKRs of multiple products or departments. When gathered in one view it acts as a central source for teammates to understand and track the OKRs for various products, teams, or departments.

If the main roadmap that's relied upon in your company is a feature-based roadmap with a timeline view, then this is a sign that you're working in a feature factory.

If you want to try out various product roadmap templates we created a [template library](#). Check them out.



The right product management tool

One important point to mention is to confirm that your product management tool enables you to escape the feature factory, or keeps you stuck in one.

Even if the product management market is new, there are different approaches to product management and you need to confirm that you have the right tool that enables you to accomplish your goals and adapt to changing needs.

If you cannot:

- Easily collect customer feedback from various sources
- Tie customer feedback to problems to solve, not to features
- Display the roadmap at different levels (from objectives to features) without having to change your tool
- Give you a vision of the key objectives
- Create custom prioritization frameworks that fit your business goals
- Involve multiple stakeholders to vote for restraint criteria in your prioritization process

Then this may be a sign that you don't have the right tool to escape the feature factory and it may be a great time to consider a change.

Six ways to identify if you need to practice product management in a modern way

There are a multitude of modern challenges that product teams face on a daily basis as they deliver solutions to customers and users. We know about them in detail because we wrote a [detailed report](#) on this topic.

Solve these problems with a modular flexible product management tool that enables you to build products in a modern way that works for your team.

Here are six signs to look out for to determine whether you need to modernize your product management approach and leverage airfocus.



Need to tie documented feedback directly to product insights

Gathering product feedback is vital, from customers, users, and internal stakeholders.

Customer and user feedback tied to problems enables you to create value for them via your products, features, and enhancements. Gathering internal feedback enables you to sort out which initiatives to prioritize and gain sentiment on how your team feels about your product suite.

It's important to have one central location where feedback is documented to gather the right information and ensure that your feedback is organized.

Take this a step further however to generate insights from this feedback to inform your strategy. If you're currently missing this within your product management tool and process then you should consider modernizing your approach.

2

Need for engagement with your customers and user base

Engagement with customers and users is key. These dedicated stakeholders support your products and want to be aware of what changes are being made to further assist them in accomplishing their goals.

Engagement can take many forms from customer webinars, to enrolling them as beta testers, to customer interviews.

One key method of engagement is frequently sharing your product plans and updates via your product roadmap. If your public facing roadmaps are being built in a word document or PowerPoint slide and then shared with customers when asked, then consider modernizing your approach.

Leverage a product management tool that gives you full visibility into your product development process from gathered feedback, to delivery, and post release of features, with the ability to engage your customer and user base by sharing easily updatable roadmaps with them.

3

Need systematic and transparent prioritization

Not everything can be done, nor should be. It's important to have a systematic approach to determining which initiatives to address.

However, different teams sometimes have different ways to prioritize. Customization is important when using a product management tool to modify the grading criteria towards your needs.

Prioritize opportunities and initiatives based on a defined (and customizable) set of values and cost criteria. Then score the items based on your input and relative criterion, then visually map them out on a prioritization chart to not only help your team make more informed and objective decisions.

This is one approach among others. The key factor is having a tool that supports your approach.

4

Need to focus on outcomes over outputs

It's the shift in focus on outcomes that enables companies to escape the feature factory trap. When focusing on outcomes teams can make strategic product decisions that accomplish customer and business goals.

If your team is mainly focused on outputs then you definitely need to modernize your approach.

Not only with outcome-based roadmaps to clearly identify and communicate how delivered items tie directly to the product and company vision, and the goals to accomplish, but also by tying the different levels of hierarchy from objectives to features.

5

Need better team alignment

Cross-functional teams should be aware of what's being worked on, whose leading initiatives, and expected release dates.

Alignment should not be a challenge within product teams. If it is then perhaps you should consider a modern product management tool that easily facilitates this.

A tool that can align teams with a clear direction. Along with the ability to craft dynamic and strategic lean roadmaps with ease while focusing on outcomes with team objectives to solve the right problems for customers, users, and your business.

6

Need to work your teams' way

Companies and teams often face challenges as they grow. When working with many product management solutions this only further supports the issue as the tool is not modular nor customizable.



Modularity with product management tools enables teams to accomplish the following:

- **Use product-specific apps to accomplish their goals**
- **Take advantage of customization with custom fields, views, and workspaces**
- **Use custom prioritization frameworks**
- **Use multiple customizable templates**

While you're busy with day-to-day activities and priorities, take a step back and look ahead to plan for the future to ensure that your teams are set up for success with the right tool that enables them to perform their best work.

Give your team the unique work environments they require for their work while leveraging a product stack that grows with you.

Now that we've gone through these six points, here's the good news. [airfocus](#) can support you with all of these key needs and more.

What next?

You now have a much better understanding of what feature factories are, the multiple problems they cause, how to communicate with stakeholders, and detailed steps on escaping the feature factory trap.

If you're working within a feature factory, realize that your company is becoming one, or simply want to ensure that your company does not become one in the future, this is the guide that you should regularly return to and follow.

Avoid becoming a feature factory by focusing on outcomes over outputs. Gain full visibility into your product management process from setting objectives, prioritizing initiatives, managing releases, gaining [insights](#) post-release, and more with [airfocus](#).

airfocus is the first modular product management platform that helps product teams discover, learn, plan and deliver solutions in a way that fits their specific needs.

From vision to delivery your way, get started with a [14-day free trial](#), or book a demo [here](#).

