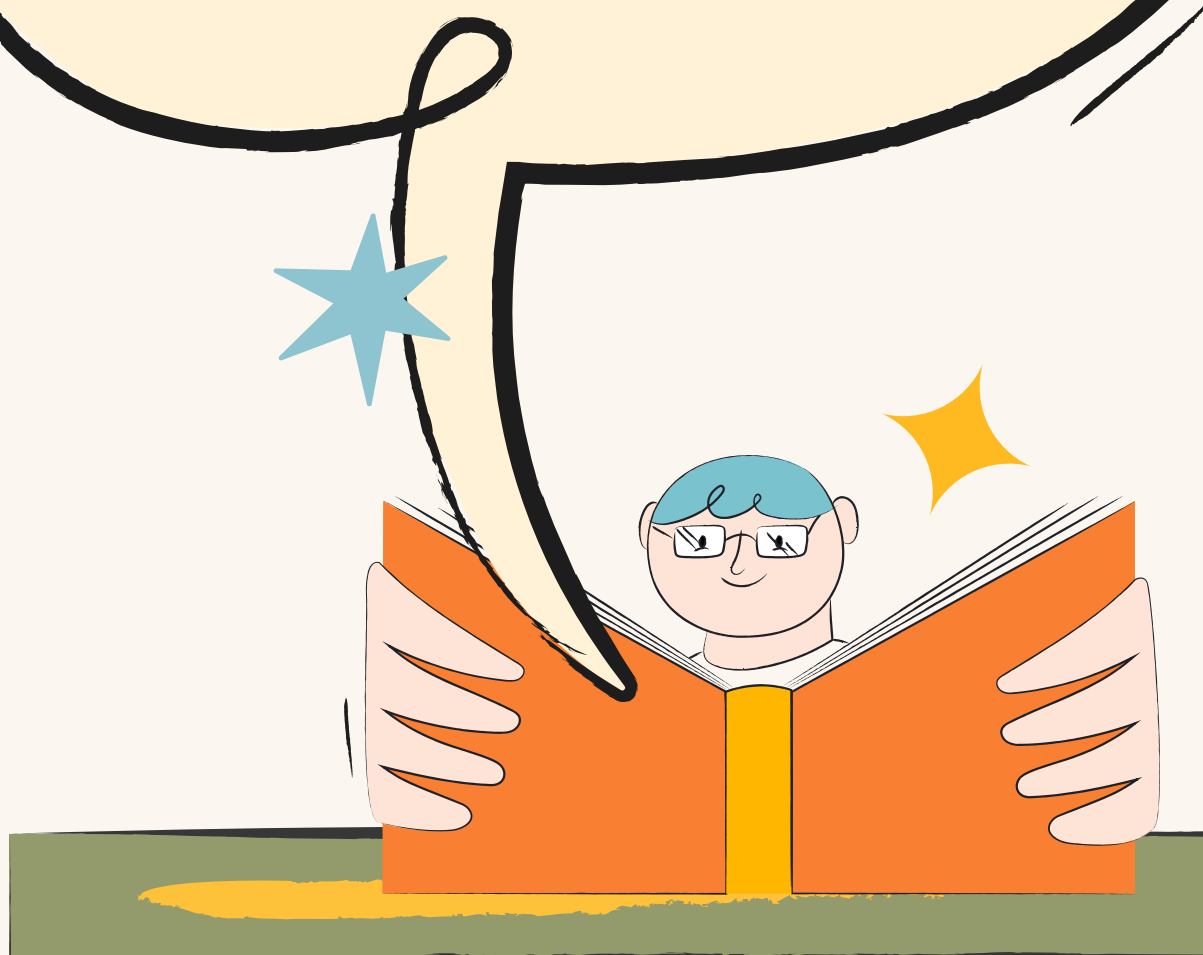


The Beginner Product Manager's Handbook



Hi, friend!

First things first, congratulations! You're about to embark on a journey that will be exciting, challenging, rewarding, and take your career places that today might seem unimaginable. But, of course, you're going to need a little bit of help along the way.

We created this resource to help you kick off your journey in product management. Think of this ebook as a portable cheerleader, mentor, and field guide—all rolled into one. Here's a preview of what's inside:

- ✦ First up, we've got **Lily Smith** with **"10 Tips for New Product Managers to Work Smarter and Advance Faster."** Lily's insights will become the playbook that'll have you cruising through your first 100 days.
- ✦ Next, **Becki Hyde's "Leading Your Team from Idea to Success"** will give you a solid foundation for working with product teams—which will come in handy from pretty much day one. From ideation to execution, learn how to rally your team, align goals, and make the magic happen.
- ✦ Ever heard the phrase 'the customer is king?' Well, **Jeanette Fuccella** is about to make you the king's confidant with **"The PM Guide to User Research: Listen to Your Customers to Build Better Products."** Jeanette shows you how to master the art of user research so you can build stuff people actually want to use.

✦ Last but not least is **Andy Smith's "A Crash Course in the 3 Need-to-Know Product Management Methodologies."** Methodologies and frameworks are big in this line of work, and even if you eventually decide to veer away from these systems, you need to know the rules before you can break them. Andy's guide is a great starting point to get you up to speed.

Alright, enough from me. Go ahead, dive in, and make waves! And remember, every product that changed the world started with someone who dared to learn something new.

Keep on shipping,



Hannah Clark
Editor, The Product Manager

Table of Contents

01

10 Tips for New Product Managers to Work Smarter and Advance Faster

– by Lily Smith

p.4-20

02

Leading Your Team from Idea to Success

– by Becki Hyde

p.21-32

03

The PM Guide to User Research: Listen to Your Customers to Build Better Products

– by Jeanette Fuccella

p.33-45

04

A Crash Course in the 3 Need-to-Know Product Management Methodologies

– by Andy Smith

p.46-59

01

10 Tips for New Product Managers to Work Smarter and Advance Faster

– by Lily Smith

The life of a product manager tends to be fraught with chaos, and juggling priorities that are competing for your time is often the norm. So, how can we work smarter? And how can we make time for learning so that we can advance our own careers, deliver better work and ultimately better business outcomes?

As a product manager who has risen through the ranks mainly in small startups, there is a certain mentality to pace and a work ethic that I've adopted which I believe has really helped me in those tougher times. This article shares some of the learnings from my journey developing into a Chief Product Officer role.

The following advice is provided in 'no particular order', and I hope that you find some nuggets of inspiration to support you on your own, unique, product journey. Here's a preview of what's inside:

- 1) Ask yourself, "What's the most important thing right now?"
- 2) Establish a culture of sharing
- 3) Develop your empathy beyond customers
- 4) Learn to be honest with yourself about your own assumptions and biases
- 5) Get really good at asking questions - and listening well
- 6) Get used to giving, receiving, and sharing feedback
- 7) Understand what good vs good enough looks like
- 8) Be brave
- 9) Practice
- 10) Network

1. Ask yourself, “What’s the most important thing right now?”



As PMs we’re often very busy, and it’s easy to get caught up in ‘busy work’. I’ve found myself in this state a lot—and still do! ‘Busy work’ is essentially work that is keeping you busy but isn’t necessarily helping you to achieve the goal that you and your product team are aiming for.

At a macro level, you need to make time on a weekly basis to step back, reflect, and ask yourself, “Is what I’m doing today the most important thing to achieve my goals / the business goals”?

Sometimes the answer will be “yes”, sometimes the answer will be “no”, and you can take corrective action. But sometimes (and probably often) you won’t be certain, and at this point you should absolutely lean on your team and your manager to reflect with you and help you understand if you’ve prioritized correctly.

Frequently, we end up prioritizing work that has been identified as important at a more strategic level (e.g. the business decides that we need this feature or to go after this customer segment). If you're in this situation you can approach it two ways:

1. Do the work, deliver the desired output, and see if you get the outcome.
2. Question the logic / research that resulted in that strategic decision, dial in on the outcome, and flex the output depending on what you find out along the way.

Most PMs will feel that point 2 is the 'right' way to respond, but if you haven't gained the trust of the business or if you don't have the experience to question the strategy, then I would recommend to stick with 1—because this will earn you trust and gain you experience and confidence in your product methodology.

At a more micro level, when I'm working in a product team, I re-prioritize on an almost daily basis. I'll review the progress in a sprint and re-prioritize the work within the sprint according to information that I've gained. When doing this, it's important to communicate changes to the development team and why you've made them (if they weren't already involved in the decision-making).

It won't happen every day, but on a regular basis you should be able to talk about the value that you've created for the team, the product, the customers, and the business. Keep a note of this so that you can easily reflect on the impact that you're having—it's a great motivator and reassures you that you're on the right path.

2. Establish a culture of sharing



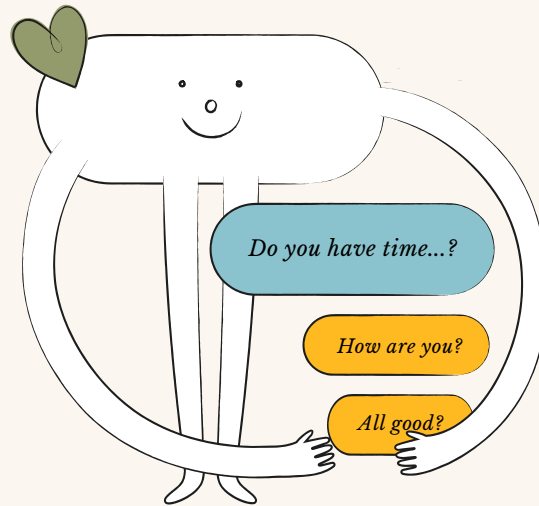
There's a very good reason why so many people talk about culture when discussing how to create high-performing product teams. The culture of the team can have a huge impact on performance, and being able to create and nurture that culture is a huge plus for a product manager.

Even if the business has a strong existing (and perhaps not ideal) culture, it's possible to influence your product team to create a micro-culture that reflects the way you want to work.

Here are some basic rules I use myself that are key to creating a culture that fosters an accountable, supportive, high-performing product team:

- **Be helpful to others:** If your developers are feeling the pressure of a deadline, ask if you can help. Maybe you can do some testing, provide some useful assets for them, or just talk through a challenge they're facing. Or perhaps you can give them some space and clear meetings from their diary. It's always good to offer assistance—it helps the rest of the team feel that they are being supported by you.
- **Ask for help:** On the flipside, if you are feeling the pressure, let your team know and share what's bothering you. This helps people see that you are human and demonstrates that you value the views and opinions of your peers. It will also help you get a more diverse perspective on challenges that you are facing.
- **Share the blame when things go wrong:** If something goes wrong, it can be so easy and natural to try and avoid the shame and find the culprit. But if you're truly acting as a product team leader, you'll accept that everyone has a part to play in mistakes. You must develop your reflective skills, so that in each moment you're able to answer these questions: "What was my contribution to this, and how could I have helped to avoid this situation?"
- **Share the success when things go right:** And similarly, when things go right, make sure the whole team can bask in the glow—even if one person stood out as the main superstar, it's unlikely you could have achieved the success without the support of others (maybe they were even making space for that person?). One person's success should be the team's success.

3. Develop your empathy beyond customers

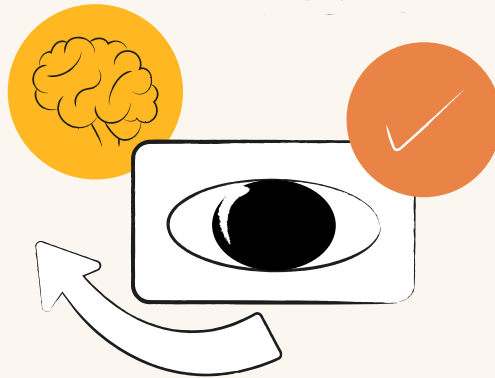


It's super important to empathize with your team, your manager, the leadership and other peers, as well as your customers.

We can end up feeling that other people's behaviors are a reflection of how they feel about us (for example, "no one paid attention in my product demo because they don't value my work"). This is usually not the case, and often people can have other things going on—whether in work or in their personal life.

Learn to ask "how are you?" with intention, and "do you have time for this?", or "do you have other priorities right now?". All of these questions will invite the team member to share a little bit more, which might help you understand their own personal context. You can then adjust your ways of working to accommodate them—which they will hugely appreciate. You'll also develop a much deeper understanding of the different challenges across the different roles and teams and become a better product leader as a result.

4. Learn to be honest with yourself about your own assumptions and biases



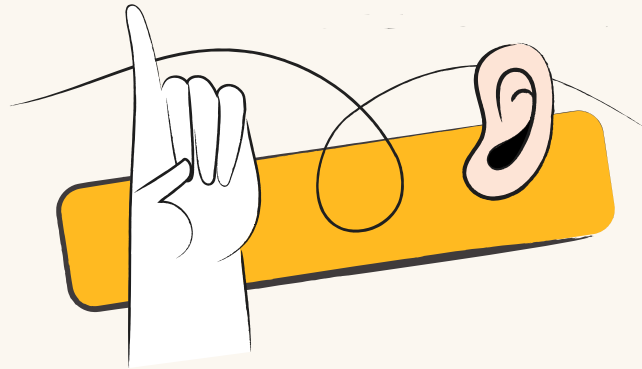
For PMs there can be an expectation that we should have all the answers. But one thing I've learned is that it's okay to say that you don't know. It's okay to say, "based on this research my assumption is this... but until we go live, we won't know for sure". Or "I'm biased towards this idea because it really solves my own problems, but I don't yet know if it's a priority for our customers".

I have a constant filter to check what I'm saying. Do I really know something, or have I assumed it? If it's an assumption, then I make that clear. I'll also try to find out if it is really true.

Are you biased toward a particular decision? If you feel strongly about something, check yourself on why you feel like that—do you have a good reason, or are your biases skewing your thinking?

Never present your opinion as fact. My kids do this a lot, and it really gets my goat.

5. Become better at asking questions—and listening well

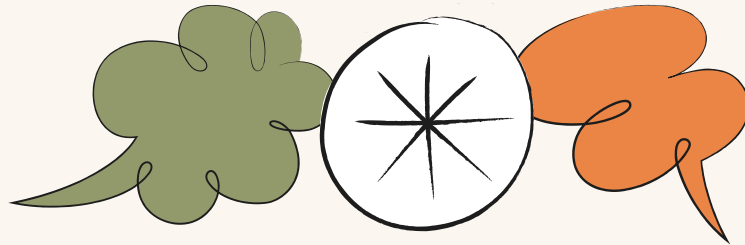


Practice asking questions! If you're going to a BBQ with family, a wedding, or a similar social occasion, get really good at asking questions, digging into the details, and listening to the answers.

People generally love to talk about the things that interest them, so it's pretty easy to practice this. Then take this skill into work. If the developers start talking about something that you don't understand, pull one of them aside later, and ask them to help you understand it. Can they draw you a diagram? Can they tell you the history and the future ideal state?

And don't be shy to ask the stupid questions. So many times I've felt like everyone was nodding along in a meeting and I've asked 'the stupid question' only to have half the room say "yes, I was unsure about that too!". If you don't want to ask in front of others, follow up later and make sure that you understand. It's your responsibility to have at least a surface level understanding of everything going on with your product (from technical design to market research to financial modeling).

6. Get used to giving, receiving, and sharing feedback



Feedback can be scary—you don't want someone to tell you that you're not perfect. But the alternative is that they think it but don't tell you, and you miss out on an opportunity to improve. Or even worse, they tell other people, and you develop a reputation for being a certain way without having the opportunity to address the critique.

A piece of feedback I have received often is along the lines of “you're always pointing out the problems in other people's areas, it's very demotivating”. I have to be careful and remember that lots of people aren't used to direct feedback, and they need to get to know me better before I can be so direct. On the flipside, one of the best compliments I've had was from an engineer (whom I worked with closely for a long time) who said to me “you're really good at telling me when my work isn't good enough, but not making me feel bad about it”.

Some people are better at giving feedback than others, so you may have to coax someone into sharing their thoughts with you. My advice is to regularly ask for feedback, and if someone looks pensive or frustrated

during a meeting with you, ask them for two minutes of their time and say something like, “I noticed that you didn’t seem happy in that meeting, can I ask if I have said something that you weren’t happy with?”

Then tell them you would really appreciate it if they shared their thoughts because you want to have a good working relationship. Hopefully, this will give them the space to share and reassure them that you are open to listening. If they give you feedback, you must respond considerately (even if you don’t agree with the critique!). Otherwise you risk never getting feedback from them again. Try saying something like, “I really appreciate you sharing your thoughts with me, your opinion really matters and I’m going to think about what you’ve said”.

Conversely, it’s really good practice to share feedback with others. This can be really difficult if you tend to avoid conflict. But it’s so important in developing a culture of trust and a high-performing team. You want your team members to be thinking, “if [insert your name here] thought I could be working better, I’m sure they would say something directly, because they always do”.

If you struggle with giving others feedback, I find being upfront about it really helps. For example, “I’d like to give you some feedback but I find this really hard because I want you to like me and feel good about your work. Do you mind if I share something that I think is stopping you from being amazing at your job?”

Creating a culture of regularly giving and receiving feedback normalizes this exchange and allows the whole team to grow and develop. In my current role we have it as an agenda point on all 1:1 meetings (whether with managers or peers).

7. Understand what good vs good enough looks like



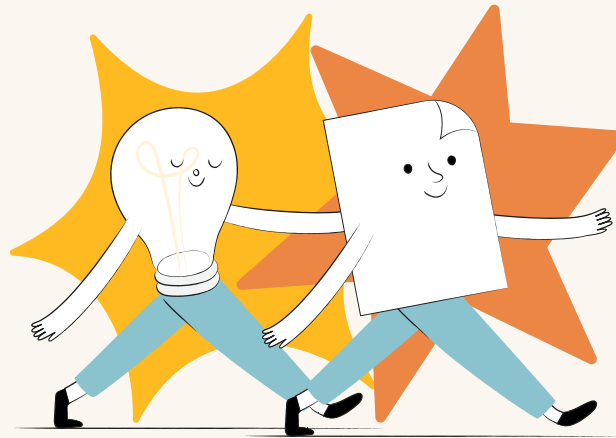
In my early career I worked with a CEO who constantly said to me “you need to know what good looks like”. Honestly, I was kind of confused by this at the time, but now I find myself having the same conversations with product managers.

Good vs fast is a trade-off that product managers are very familiar with. It usually refers to design and engineering work but it equally applies to PM work. For example, if you’re writing up user stories for developers that are currently blocked by you, they may just need the important details and not so much the context.

But sometimes you really need to impress, and to learn how to do this is not often done in a silo. You’ll need to spend time practicing or finessing your work and asking for feedback to improve. You have to do this on a regular basis—if you don’t, then you’re probably not getting better at your job. This is very closely related to point 6 and asking for feedback, but it’s also about learning to judge for yourself whether your work is good.

Look at how people respond to others’ work—do they think it’s good or good enough? See the people rising through the ranks, getting attention from the CEO and other stakeholders, what does their contribution look like and how can you match that?

8. Be brave!



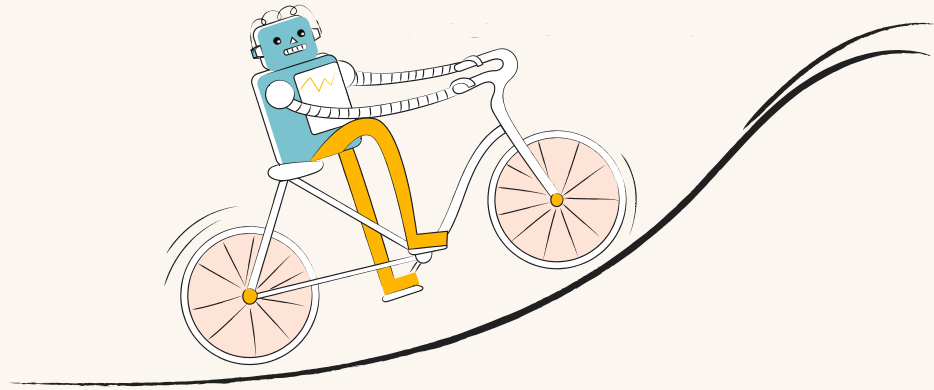
If you want to develop yourself as a PM, you will need to stretch out of your comfort zone regularly. The best possible outcome of this is getting used to being out of your comfort zone!

I regularly challenge myself to do things that are new, scary, and require me to push myself beyond my current capabilities—not just at work but in my home life too.

This has helped me learn how to deal with stress and to embrace and relax into ‘the unknown’. Positive affirmations really help me personally. I tell myself “you can do this, it will be great, your intentions are good, you’ve done your homework, and that matters the most”.

The feeling of fear is very similar to excitement. So when I feel afraid (even if it’s just presenting the roadmap to the business), I tell myself that it’s exciting, and honestly, I think a lot of the time it’s actually true!

9. Practice



Often when we talk about learning product management, we neglect to talk about practice. But this is fundamental to becoming better at your job. Again, if you are in your comfort zone, you probably aren't learning. You need to adapt what you're doing and practice new and different ways in order to figure out what works best.

You might not get things right first time—e.g. the first time you do a user interview—but if you conduct 30 user interviews and make sure to reflect on each one after, then I can guarantee you will be better after the last one. And, guess what, ask for feedback!

If it's daunting because it's very new or very important, establish who can help you practice beforehand. Perhaps you need to do a test run of a presentation or talk through the exercises you're going to run in a workshop. If you don't have time for practice, then set clear expectations while trying the thing (e.g. "this is the first time running this exercise, this is what it is, this is why I think it's a good idea, and I'd love for you to try it with me and see how it goes").

10. Network with other product managers



I've cultivated a network of product people over my many years in tech. These are people that I've worked with, people that have come to talk at events I run or that I've interviewed on the podcast that I co-host, and people that I've met at events.

Throughout my career I've found my product management network crucial for my development for a number of reasons. These people have been very supportive when I've needed advice or a pep talk. I've been inspired by some of their stories and learnings and different ways of working that they've experienced. And nearly all of my product management roles have come through personal recommendations and contacts.

If you're settled in a role and you already have the support of product people, that's awesome. But don't neglect getting out there and hanging out with other product people. Other opportunities (to learn, connect, and—if it's part of your path—to build your personal brand) are likely to present themselves if you expand your network. Try to create some diversity in your network too—talk to people that look different to you and come with a different perspective.

It's not just about product people too. I would encourage you to connect meaningfully with the people inside your business. Get to know the leadership team, for example. What drives them, what are their stories, and what are their challenges? These deeper connections open you up to a large support network when you need it and greater opportunities along the way.

Don't neglect your mental health

I hope you've found this set of tips helpful. My last piece of parting advice is to take care of yourself. Product is hard but it's also amazingly fun and rewarding. Like a new parent looking after their newborn child, you won't be doing your best work if you've had endless sleepless nights, so make sure you take moments for self-care.

By looking after yourself you also set the very best example to your peers and your team. You're able to think more clearly and creatively, and with more energy and enthusiasm.

I wish you the very best in your own product journey.



Lily Smith

Lily Smith is CPO at BBC Maestro, co-host of The Product Experience Podcast and founder of ProductTank Bristol. In her 15 years in tech, she's worked with dozens of start-ups and specializes in the early stage—discovering product-market-fit and building the team.

Lily has worked on products for GoCompare, BTVision, Future Publishing, Travis Perkins, and Bower Collective and has a wide range of experience across D2C and B2B .

02

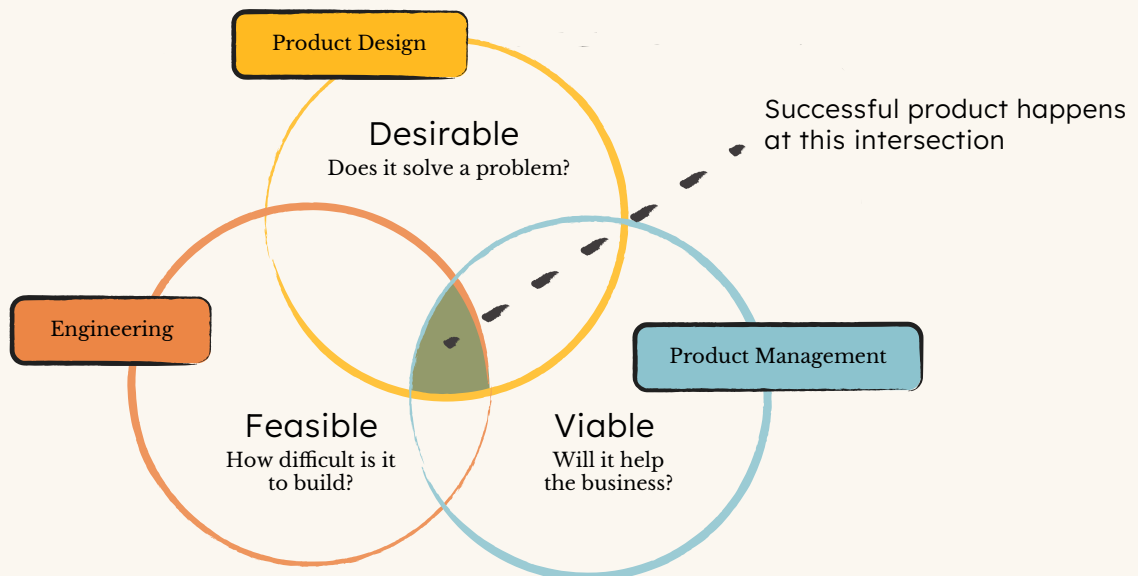
Leading Your Team from Idea to Success

– by Becki Hyde

Good product management requires leadership without authority. While there are product managers who lead by telling others what to do, dictator behavior isn't the best way to create products that drive the business forward, meet the needs of users, and balance myriad technical considerations. Being the idea person, who hands down tasks to a team of specialists, not only limits the effectiveness and efficiency of the people who build and deliver the product, but is also incredibly risky to the product itself.

Product development is a team sport, and while someone has to call the plays, that person must do so by synthesizing the best possible information and ideas from a diverse group of teammates. So how does a product manager lead without authority, manage all the inputs to the development process, and help a team come out the other side with something awesome? Let's start at the beginning.

Keep the team balanced



First, we must assemble a team. While a product manager generally doesn't have direct control over who is on the team, it helps to know what team makeup is going to help you be most successful. At minimum you can ask and advocate within your organization for the right ingredients for success.

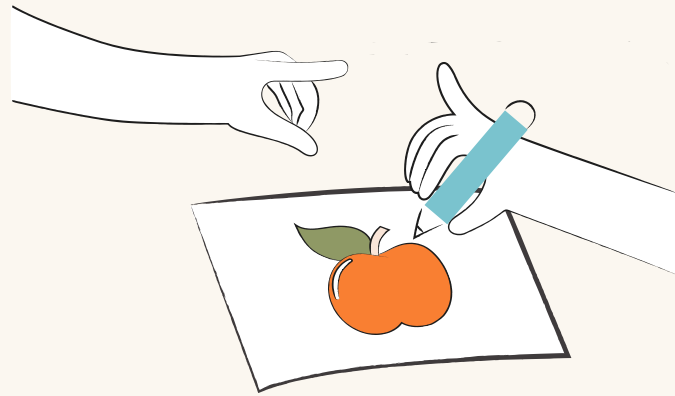
A high-performing product team consists of every specialty or discipline needed to make day-to-day decisions regarding the product. Generally speaking this means two to six developers with varying expertise and one to two product designers. The important thing, regardless of who is on the team, is that you all start together. This means from kickoff, through research and ideation, into building and shipping, you have all disciplines represented. Starting together has many benefits.

By starting together, you build shared context. Everyone is able to understand user needs, because they're around for early research readouts. Everyone builds a growing understanding of the technical considerations on the product, so they're able to discuss tradeoffs in implementation. Everyone knows what the business value of the product is, so they can connect their work to the larger mission. Shared context makes teams and products more durable, and more able to pivot on a dime when new information is uncovered.

By starting together, you build psychological safety. Psychological safety is the belief that you will not be punished or humiliated for speaking up with ideas, questions, concerns, or mistakes. In teams, it refers to team members believing that they can take risks without being shamed by other team members. Psychological safety is now widely understood to be the greatest contributor to better performance on product teams. There are many ways to create and maintain psychological safety, but a great way to begin is by having equal participation in early decision making for the product. When people are handed tasks without understanding the motivation or decisions behind the plan, they will have blinders on at best. At worst, they'll be disengaged and lack motivation for their role on the product team altogether.

When the team starts together, there can be thrash or lack of direction as the various specialties on the team each pull in their own direction. This is where a product manager can create balance, focus, and lead without dictating. Strong facilitation skills are a must for product managers, especially in the early stages of product development. Use workshops and inclusive facilitation techniques to help the team explore where they should, and drive toward decisions when it's time to move forward.

Collaborate with designers



Knowing when you know enough to start building something is an art, not a science. There's an adage in product development—if you're not nervous about your first release, you've waited too long. The same applies for moving from research and discovery into development. If you feel confident, you've waited too long. You may have some personalities among your design or development folks that are eager to push forward—and if so, awesome!—but more likely the team will either look to the product manager for that “let's get on with it” momentum, or they'll keep digging and researching until you get everyone together and facilitate a decision.

Before you build out a backlog and start delivering stories, designers are going to be spending the majority of their time doing user research and other discovery-related activities. This is awesome! However, we can't delay jumping into development indefinitely. It's also not a good idea to make decisions about what needs research and when to move forward without the understanding and commitment of the designers on the team.

Maintain a healthy relationship with design by collaborating on a learning backlog, to provide focus for your research efforts. Gather the whole team and do an assumptions exercise, to generate assumptions about your product that, if proven false, could hamper your success. Also do a risks and mitigations exercise in addition to the assumptions exercise. Each workshop tends to generate different ideas and action items.

Once you have a prioritized list of assumptions to validate and risks to mitigate, sit down with the designers and identify which items involve research, and how much the team needs to know to balance risk with moving forward. Then, check in on that list regularly, at least weekly, and discuss if you know enough to get started, or if there's something out there we have to hunt down before we know what to build.

Collaborate with developers

In much the same way you collaborate with designers on research related to your users and domain, you'll collaborate with developers on technical risks, assumptions, and general system design. Again, the role of the product manager in these conversations is to offer another perspective, focus, and prioritization in the context of business need, so that we don't simply go off researching every possible implementation before we start trying some things out. Collaboratively building a list of technical chores will also help you as the PM to balance trade offs later on, and speak to your stakeholders about major blockers that might be coming your way.

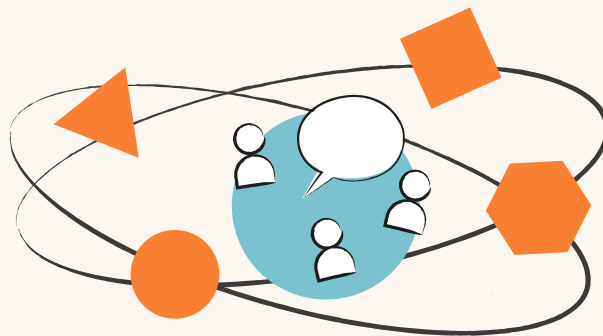
Define and measure success

A significant role of the product manager during early discovery and solution framing is to help the team focus and define what success looks like. Set clear goals at the very start, and make them measurable.

“Build a mobile app for kids learning piano” is something people know how to do, but the definition of success is binary (we either did or didn’t) and doesn’t relate to any larger business goal (how will this app generate revenue or reduce cost?).

“Increase subscriptions to our online piano classes by 25 percent” is a much clearer goal. It gives the team something to grab onto (I can make that number go up!) and provides focus (our app doesn’t need a social component, as research shows that doesn’t impact subscription signups).

Create tight feedback loops



One of the best ways to mitigate risk in product development is by getting feedback early and often. A great product leader seeks out feedback from the team, from users, and from stakeholders, and incorporates that feedback throughout the product’s development. Feedback enables the product team to iterate and adjust to new information, ensuring the product is viable, valuable, and feasible.

- **How to create feedback loops within your team**

The first and best way to lead the team through the day-to-day is to encourage and support processes and practices that create tight feedback loops. There's new information to respond to every day, and being able to iterate through things quickly is vital to success.

Lead by example—build your backlog with small, full-stack stories that encourage cross-discipline conversations. Bring the whole team together for planning meetings. Involve designers and developers in acceptance testing. Invite feedback and conversation about prioritization. In general, think of yourself as more of a steward of the vision than someone who writes out tasks for others to complete.

- **How to create feedback loops with users**

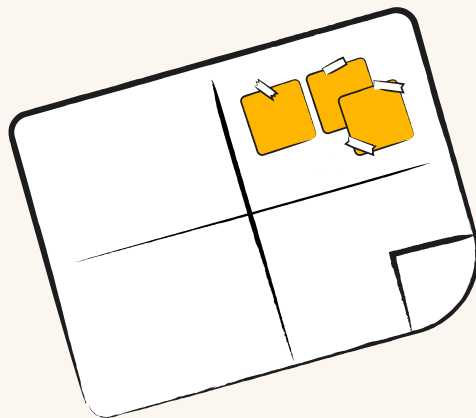
Advocate for user feedback at all stages of product development, and support the designers in getting it done. This can mean helping out with logistics or networking, or it can be as simple as sitting in on research sessions and taking notes. While it's not your role to lead research, being supportive and collaborative with designers on research and testing efforts will help you understand what's important to them, and incorporate their valuable perspective into the product strategy.

- **With your stakeholders**

While it may seem like you are serving your team by acting as a buffer between them and outside stakeholders, in reality this turns you into a translator, increasing not just your communication overhead but also the risk that

something gets lost in translation. While it is helpful for you to manage communication with stakeholders, invite the team to present, and invite stakeholders to speak to the team directly when they have questions about users, design, implementation, or technical concerns.

Maintain focus post-launch



Once your team has launched a product to users, the real chaos begins. If the team thought they were inundated with information and options before, now the floodgates are truly open. You'll still have stakeholder requests, changes in technology that impact your product, and major features you want to deliver that align to your strategy, but now you'll be adding in requests and feedback from users and customers—at scale.

- **Pivot or persevere**

Once initial user and customer feedback starts rolling in, you'll be faced with a key decision: pivot, or persevere? If you're having the impact you hoped (e.g. moving the metrics you defined at the start) then it's likely you'll persevere. If things aren't going the way you hoped, you have a more difficult choice before you.

Regardless, the decision should leverage the expertise and perspective of everyone on your team, as well as key stakeholders. This isn't a decision you should make alone. Someone else may have key information you missed or forgot, or may see a path forward that wasn't obvious to you. By facilitating this key decision, you reduce the risk of a bad call, and keep the team engaged in whatever decision is made—especially important if you're going to pivot with the same group of people.

- **Say no**

As you're moving forward with building out new features, you'll have an ever-increasing volume of requests, suggestions, nice-to-haves, and immovable requirements. At this point, product leadership becomes as much about what you say "no" to as what you prioritize in your backlog and roadmap.

The continued success of your product is dependent on many factors, but focus is key. It's always an option to alter the strategy (see "Pivot or persevere" above), but once you've got one, maintaining focus on that strategy and the vision it supports will ensure you're not diluting your efforts.

Be a servant leader

Product development is too complex for one person to drive alone—it takes a team of experts. The best leaders are the ones who are able to bring their ideas to the table, while also welcoming the ideas, feedback, and questions of others. They see themselves as support for their team. They respect and value the contributions of everyone, and work to ensure everyone's voice is heard. This is called "servant leadership," and it's crucial to healthy teams, and healthy teams build the most successful products.

- **Help the team get unstuck**

Sometimes the best thing a servant leader can do for their team is to help them get unstuck. Product development is a risky endeavor, full of unknowns and surprises. It can be easy to get lost in the information overload and find yourself unable to make the call. It can be even easier to feel like you don't have enough information to make a decision, and you have no clear path to finding what you need.

This is where a great product manager steps in and leads through ambiguity. Sometimes it's to facilitate a workshop or conversation that brings clarity to the team. Sometimes it's to make the call when no one on the team feels comfortable. Sometimes it's as simple as breaking a tie on an implementation detail, when either solution will work just fine.

Give away the spotlight

Recognition, success, accolades—all the benefits of a job well done and a product that solves user problems and benefits the business—are not a finite resource. It's easy for product managers to steal the spotlight, often unintentionally. Stakeholders see you as the point of contact, or the face of the product, and will attribute all the successes and failures to you. Be willing to take on the latter, and be sure to give credit for the former to the team. They deserve it, and by sharing it with them, they'll see you as a team member, and not a dictator. Nothing is more important when it comes time to make tough decisions.



Becki Hyde

Becki Hyde is a product manager and design leader who has worked with companies large (200k+) and small (5); in agencies, startups, midsize tech, government, and enterprises operating in manufacturing, healthcare, finance, and more. The common theme through all her work has been creating a path forward through ambiguity, working across disciplines to solve problems.

Becki is currently a Manager of Product Managers and Designers at VMware Tanzu Labs.

03

The PM Guide to User Research: Listen to Your Customers to Build Better Products

– by Jeanette Fuccella

In today's competitive market, a deep understanding of user behaviors, needs, motivations, and pain points is essential for product managers to create successful products. User research plays a vital role in this process as it involves a systematic approach to gather and analyze data, which leads to valuable insights.

The ultimate objective of user research is to inform and guide product development decisions, ensuring that the resulting products or services effectively meet the needs and expectations of the target users. Crucially, user research is most effective when it's an ongoing and iterative process, allowing for the accumulation of knowledge about users over time.

This chapter will explore the significance of user research for product managers, highlighting the benefits it offers, and exploring the different types of user research methods.

Why is user research important for product managers?

Being a product manager can be a daunting role! Foundationally, the role of the product manager is to find the intersection between the needs of the user and the needs of the business, all while managing the various risks associated with building a product or service. The data and assets derived from user research activities provide the product manager with the necessary foundation to navigate these challenges successfully.

Overseeing the development and success of products and services requires the ability to make decisions quickly while simultaneously shepherding stakeholders through the process. User research plays a pivotal role in a product manager's

success by providing the data needed to drive stakeholder alignment around both tactical and strategic decisions.

In the fast-paced world of product management, making decisions quickly is imperative. User research provides valuable insights that enable product managers to make informed decisions efficiently, leading to faster iterations and improved products. User research equips product managers with robust data and insights that instill confidence in their decision-making process. By basing decisions on user feedback and real-world observations, product managers can align their product strategy with users' needs and desires.

In the absence of user research, product managers may find themselves paralyzed by analysis and uncertainty. User research provides a structured approach, offering a clear path forward and preventing product managers from being overwhelmed by endless deliberation.

When working at such a fast pace, it can become easy for product managers to fall into the trap of making false assumptions or having biases about their users. A strong foundation of user research can unveil blind spots and challenge preconceived notions, enabling product managers to gain fresh perspectives and uncover insights that may have otherwise been missed.

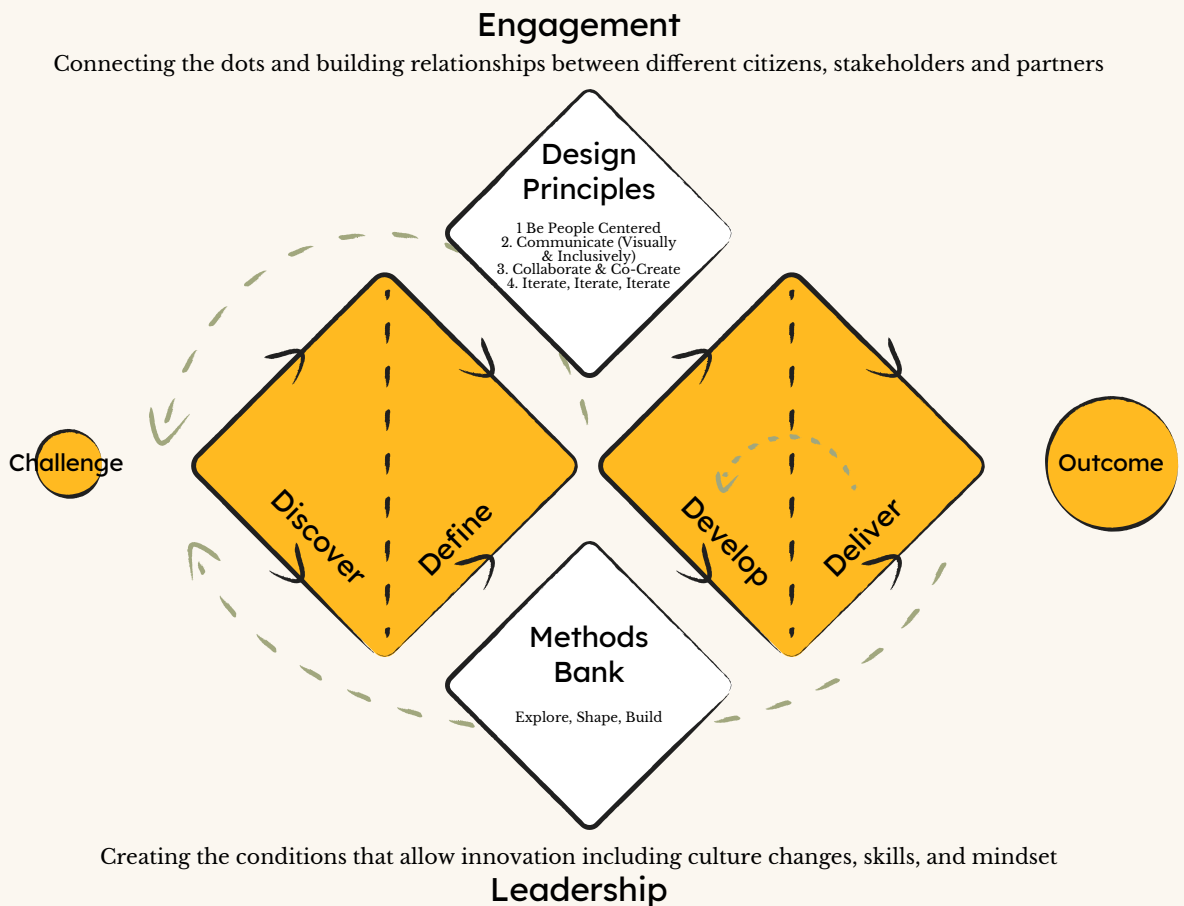
Last but not least, user research helps mitigate the inherent risks associated with product development. User research provides an opportunity to test assumptions and verify if a product concept aligns with users' expectations. It helps product managers identify areas where their initial ideas may have been flawed, allowing for adjustments and improvements along the way. By understanding users' needs, pain points, and preferences, product managers can

make informed decisions and minimize the possibility of building products that don't achieve product market fit.

User research frameworks

There are any number of published frameworks to guide product managers in approaching what types of user research to conduct throughout the product development lifecycle. Two of the most common are the Double Diamond and Continuous Discovery. Both methods aim to mitigate risk through proper exploration and experimentation prior to investing heavily in engineering. An additional 2x2 framework is useful in prioritizing research resources.

- **Double Diamond**

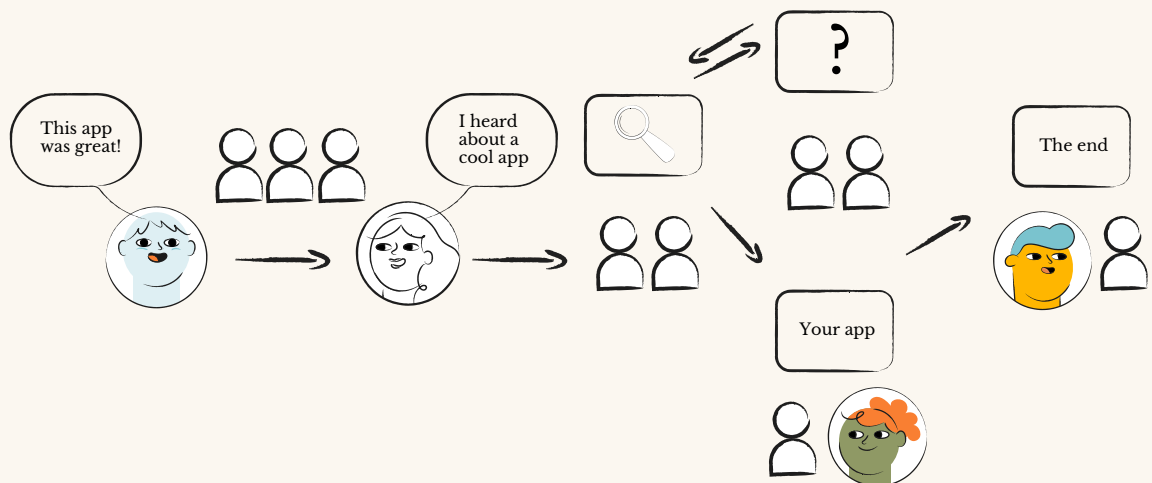


The Double Diamond framework consists of two phases:

1. Solve the right problem
2. Solve the problem right

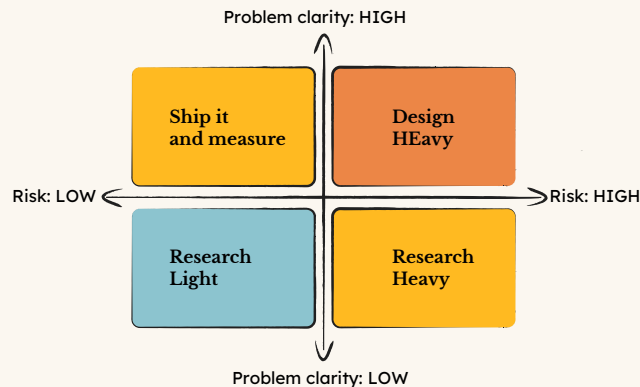
Each of these phases begins first with a motion of divergence of thought and exploration, followed by convergence around the “right” problem or solution. The Double Diamond framework is particularly helpful when researching new product opportunities or when the problem space is particularly opaque.

- **Continuous Discovery**



Continuous Discovery is a framework created by product discovery coach Teresa Torres. Unlike the Double Diamond, which suggests a more linear process, Continuous Discovery embraces the inherently iterative nature of product development. Continuous Discovery focuses on identifying and exploring opportunities and associated potential solutions. It’s most effectively used when a product already exists and the team’s focus is on making small, incremental improvements.

- **User Research Prioritization Framework**



Regardless of their approach, product teams consistently encounter far more research questions than they are able to explore. I created the User Research Prioritization Framework to help product teams evaluate where to invest their limited time and resources. The framework is most effective when used as part of a planning session. Teams identify open research questions and assumptions and then collaboratively identify the correct placement using the two axes of problem clarity and the risk of getting it wrong. Use of the framework helps identify not only the amount of research necessary, but also the type of research. Additionally, it helps teams avoid “analysis paralysis” by acknowledging that not every question requires in-depth research.

The framework has received widespread adoption across product teams and organizations of every size, shape, and industry. Part of the usefulness of the model is in its flexibility. Organizations that are more risk-tolerant have reported shifting the vertical axis to the right to lean more toward “shipping and measuring”. Other organizations, particularly those with scarce research resources, have reported applying an additional level of prioritization within the “Research Heavy” quadrant to further refine and focus their efforts. The research tool, Notably, leveraged the matrix by aligning research methods with each quadrant. Regardless of application, this 2x2 provides many teams with a solid framework for evaluating research priorities.

Types of user research

User research encompasses various methodologies that can be broadly categorized into two main types: qualitative research and quantitative research. Both types of research activities are necessary for a complete understanding of users and their needs.

Qualitative research

Qualitative research focuses on gathering rich, in-depth insights into user experiences, attitudes, and motivations. This type of research utilizes non-numerical data to gain deep insights into user behavior and preferences. Through methods such as interviews, focus groups, and observations, product managers can delve into the “why” behind user behaviors, uncovering valuable context and uncovering hidden pain points. Qualitative research allows for open-ended exploration and encourages users to express their thoughts and emotions, providing nuanced insights that quantitative data may not capture.

- **Interviews**

While a large portion of a product manager’s role is talking with customers, it’s important to distinguish between customer interviews that are user research activities and other forms of conversations with customers. Intentionally distinguishing the intent of the conversation will yield the best results. Not only will the conversation be more likely to achieve its goals but it will reduce the likelihood of misinterpretation of customer feedback.

In advance of an interview, ask: Is the goal to test, sell, or identify concerns with my prototype / concept?

- If you respond with "all three," you are tackling too much in one interview.
- If you are hoping to learn/test but also find yourself needing to educate a customer, ensure that there is a clear distinction between those two parts of the interview, e.g. "I am not going to give you a direct answer to your question right now because I want to continue to learn from you what you expect and need, but we will preserve 5 minutes at the end of the interview to answer any questions you might have."

In addition to properly identifying the intent of a customer engagement, other factors will also influence data collection and integrity. Word choice, body language, and even small interjections can have major impacts on user responses and reactions. Below are some best practices and examples:

Build trust and rapport: Establishing trust with participants is crucial for obtaining honest and open feedback. Creating a safe and non-judgmental environment encourages participants to share their thoughts and experiences freely.

- Begin with introductions. Introduce yourself first so that participants know who you are and why you want to speak with them. If possible, downplay your role since coming across as an expert may be intimidating to the participant and may prevent them from being candid. Then, ask the participant to introduce themselves in a way that they can easily do and that allows them to feel smart and confident.
- Begin with "easy" questions that allow the participant to maintain a sense of confidence and control. Even light banter is okay if it helps to build trust.

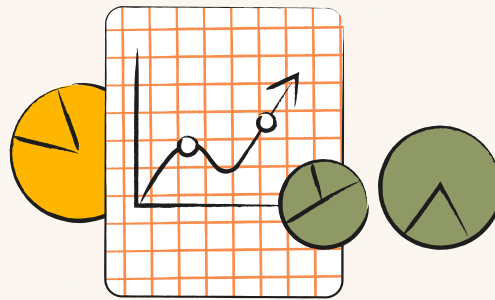
- **Concept testing**

Testing product concepts with users at an early stage helps evaluate ideas and gather feedback. This iterative process allows product managers to refine concepts, identify potential pitfalls, and align the product with user expectations. Getting feedback on a mock press release is a great way to collect early directional feedback.

- **Rapid prototype design iterations**

Creating and testing multiple low-fidelity design prototypes with different solution approaches allows product managers to iterate quickly and gather user feedback. As the product concept becomes more refined, conducting high-fidelity design evaluations becomes crucial. This involves testing interactions, user expectations, and overall usability to ensure that the product aligns with users' needs and provides a seamless experience.

Quantitative research



While qualitative research provides rich insights and a deep understanding of users, quantitative research focuses on collecting and analyzing numerical data to understand user behavior at scale. Because quantitative research involves the use of large datasets, the findings are more generalizable across larger user populations. Behavioral analytics and A/B

testing are common quantitative research methods used by product managers.

Behavioral analytics enables product managers to gain valuable insights into user actions, interactions, and patterns within their products or services. It involves collecting and analyzing data on how users behave, navigate, and engage with the product. By tracking user actions—such as clicks, page views, feature usage, and conversion rates—behavioral analytics provides product managers with a deep understanding of how users interact with their product.

By identifying usage patterns and user preferences, product managers can make data-driven decisions in prioritizing product enhancements and new features. By understanding which features are most popular or underutilized, product managers can allocate resources effectively and focus on areas that will have the greatest impact on user satisfaction and retention.

Behavioral analytics can also help product managers identify bottlenecks and areas of friction within the user experience. By analyzing user paths and conversion funnels, product managers can pinpoint where users may be dropping off or encountering obstacles, enabling them to optimize the user journey and enhance overall usability.

Finally, behavioral analytics empower product managers to measure the effectiveness of product changes and experiments. By comparing user behavior before and after implementing a new feature or design modification, product managers can assess the impact of their decisions and iterate accordingly. This approach leads to continuous improvement and a better understanding of what resonates with users.

What about surveys?

Surveys are also often a part of a product managers' research toolkit and can be used to collect both quantitative and qualitative data. Surveys are helpful for understanding sentiment, preference, and attitudinal data.

Since survey design and question formulation is a discipline unto itself, it's recommended that novice product managers utilize standardized and industry-tested survey devices before creating surveys from scratch, such as the Product/Market Fit survey, NPS, SUS/UX Lite, or Customer Effort Score. Survey data should nearly always be triangulated with other quantitative or qualitative data to ensure the proper understanding of the results.

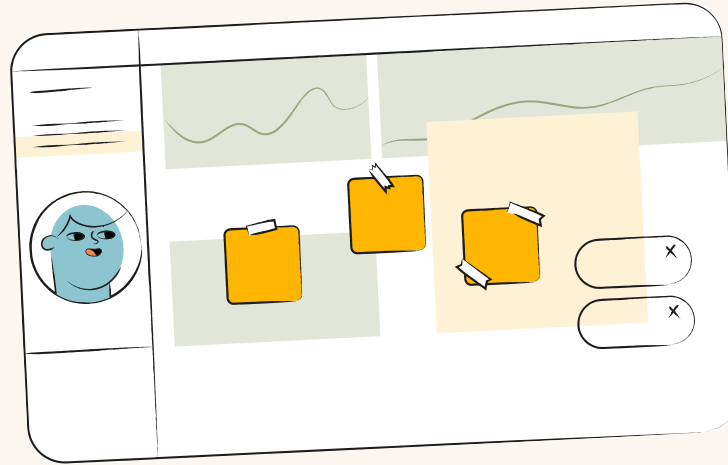
- **Mixed methods**

By leveraging both qualitative and quantitative research, product managers can make well-informed decisions, prioritize product enhancements or new features based on user needs, and design products that truly resonate with their target audience. The synergy between qualitative and quantitative research activities ensures that product managers have a holistic view of users, enabling them to create products that meet user expectations and drive business success.

- **Other user research methods for product management**

In addition to the methods identified above, there are a myriad of other user research methodologies available to product managers. See below for some recommended resources.

User research to build exceptional products



Expertise in user research is essential for product managers to create exceptional products that resonate with users. By embracing the power of qualitative and quantitative research methods, product managers can gain a deep understanding of their users' needs, preferences, and pain points.

With this knowledge, product managers can make informed decisions, minimize risks, and deliver products that captivate and delight their target audience. Incorporating user research techniques such as interviews, concept testing, prototype evaluations, and behavioral analytics empowers product managers to build user-centric products that stand the test of time.

Recommended resources:

- [The Complete Guide To Collecting Meaningful User Feedback](#)
- [How to Write UX Research Objectives \(with 14 Examples\)](#)
- [How to Use Jobs To Be Done Framework: A Guide for Product Managers](#)
- [10 Best User Research Tools](#)



Jeanette Fuccella

Jeanette Fuccella is the Director of User Research Insights & Operations at Pendo. She regularly writes and speaks on topics such as product discovery, research operations, democratization of user research, and the value of nurturing curiosity as part of a research practice

Jeanette is deeply interested in the relationship between technology and humans and is passionate about forming diverse and collaborative teams to tackle complex challenges with innovative solutions.

04

A Crash Course in the 3 Need-to-Know Product Development Methodologies

– by Andy Smith

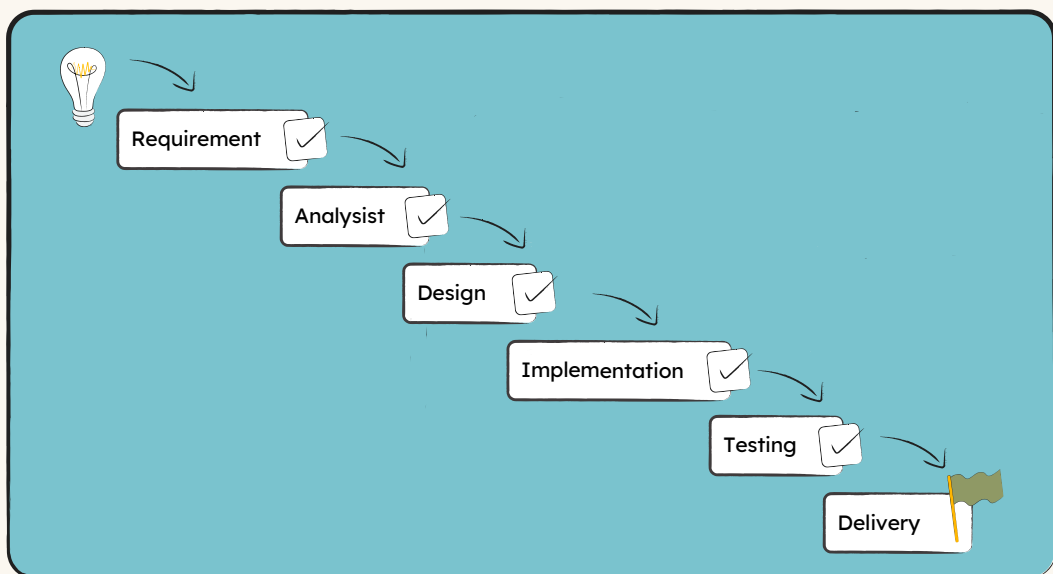
Fewer things can have a bigger impact on how you integrate and interface your team's work with your organization than the process by which you define, deliver, and measure your products. Get it right, and you'll create an environment that accelerates your team, product, and company's successes. Get it wrong, and it won't just hold you back, it can actively work against you.

Waterfall, Lean, and Agile are all popular ways to manage the successful delivery of customer value. But why would you favor one over another?

In this chapter we'll explore what they mean, their advantages and limitations, and where you may typically encounter or want to apply them.

Waterfall

Waterfall is a methodology that breaks down any deliverable into a set of sequential steps or sub-stages, each of which must be considered complete before the next one can start. To work, Waterfall requires clearly documented acceptance criteria at each stage of the process, which protects against wasted effort further down the line.



A typical Waterfall process, depicting the sequential stages of Idea, Requirements, Analysis, Design, Implementation, Testing, and Delivery

The simple, linear nature of a Waterfall process makes it easier to manage, monitor, report, and comprehend throughout. Projects can be readily visualized and mapped against time, so that teams, leadership, and stakeholders know whether everything is on track.

In a Waterfall process, the product manager's role is early on; interfacing with customers to define and document the requirements and articulate the goals. The process itself is usually managed by a project manager or assigned delivery lead.

While the “checkpoints” nature of Waterfall helps track progress, a key challenge is that delivery to the customer—the real test of value—does not happen until the very end of the process. This creates long lead times and a slow feedback cycle. Customers often say “yes” to something on paper but have far more input once they actually use a certain feature, at which point it can be hard to incorporate their feedback. The internal compartmentalization can also be a barrier for more innovative and collaborative problem-solving.

These days Waterfall is often considered outdated by many product teams as more contemporary frameworks have been developed and the types of products being managed have evolved. However, there are still many opportunities where Waterfall is a great fit.

Waterfall originated from—and is therefore highly suited to—physical production processes. It’s effective when the cost of change is high, or where there is low variance. It’s a natural fit for any development towards a fixed deadline, as the upfront planning helps evaluate viability and the formalized reporting keeps everything on track.

Lean

Where Waterfall seeks to create strong predictability, Lean looks to aggressively and continuously drive efficiency, which it does through five key principles:

1. **Value:** Identify what is valuable to the customer. Remove anything else.
2. **Value Stream:** Figure out every step of the process to deliver that value, and that value only.

3. **Flow:** Make the value stream as optimized and efficient as possible.
4. **Pull:** With an optimized production process, reduce the time-to-market such that products can be produced much more “on demand”, removing the need to store inventory or waste effort producing unnecessary units.
5. **Perfection:** Continue to learn and optimize the production process.

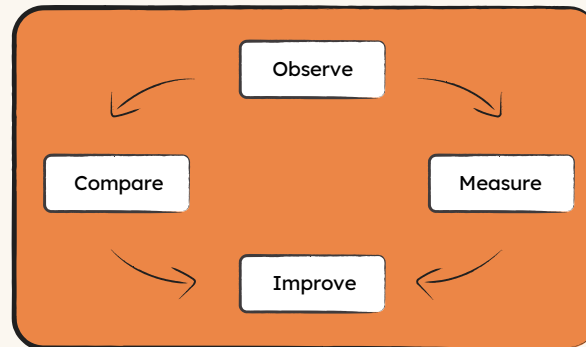
Like Waterfall, Lean comes from and is highly effective in the manufacturing industry. It was created and pioneered by the car manufacturer Toyota, which created and popularized many of the concepts of Lean manufacturing in their Toyota Production System.

A unique feature of Lean is how much it focuses on eliminating waste and inefficiencies, termed Muda, Mura, and Muri. Put simply, anything which does not bring value to the customer should be eliminated. Anything being under-utilized or utilized inappropriately—time, people, machinery, knowledge—should be optimized or scrapped. Lean also actively encourages the automation of any task which doesn't require the specific skills of a person, especially if the task is prone to human error.

But who is responsible for identifying and implementing these optimizations? No-one is better qualified than those doing the work, and Lean incorporates the concept of Kaizen; continuous improvement driven by any employee at any time.

With such distributed and wide-ranging improvement planning, it's critical to know the impact of such efforts, and one prerequisite of successful Lean implementation is good

data. Without it, there is no way to properly identify the real opportunities for improvement, or compare how much things have (or haven't) improved.



Typical phases of continuous process improvement within Lean, a repeating process to Observe, Measure, Improve, and Compare

Heavy optimization towards a laser-focussed value places much more demand on the product manager role to be deliberate and data-backed. Lean does not allow for hedging bets, so up-front confidence is needed that the chosen outcomes are the right ones.

Like Waterfall, Lean originated from the manufacturing industry, although it's also successfully applied in other industries like software. While it provides more flexibility and distributed ownership than Waterfall, it's still largely beneficial to organizations with relatively low-variance work that have a good idea of how they will deliver value and wish to prioritize effectiveness of delivery over more aggressive product discovery.

That said, over the past 15 years the concept of Lean Startup has gained more popularity. Although related to Lean manufacturing, Lean Startup optimizes the fast discovery of new value (instead of the process for delivering known value) in order to develop the right business model. This involves heavy use of build-measure-learn cycles, depending entirely on real feedback in order to check assumptions about the viability and market appetite for the business's ideas pre-launch.

Agile

Undeniably, Agile has become a major force in many organizations, with many varying degrees of success. The complexity and popularity of Agile merits some deeper understanding.

Agile was devised by a group of people from software companies. Realizing (or rather frustrated) that the constraints of popular development methodologies rooted in physical product manufacture did not apply to modern digital products, all of these individuals had become passionate about creating more effective ways to develop better software.

Representing many of the evolving alternatives such as Scrum, Extreme Programming (XP), Dynamic Systems Development Method (DSDM) and more, these individuals came together in 2001 and developed The Agile Manifesto, a set of 4 values and 12 principles that encompass the mindset and priorities an organization should embody in order to embrace these new possibilities and become agile.

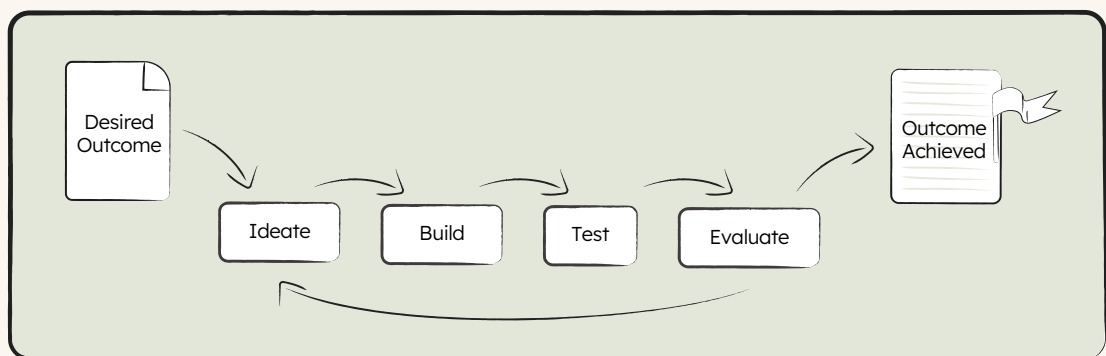
Agile Values

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

Read the **12 principles behind the Agile Manifesto**.

Agile recognises that modern software development is flexible, and therefore no longer needs to be heavily process

or document driven. As the costs of development have dropped, more risks can be taken providing the associated overhead costs of managing the process are also low. The fundamental goal is this: deliver working software incrementally that provides customer value as soon as possible in an environment that enables you to learn and make the next iteration even better. It's distinct from Lean Startup in that it's designed to incrementally deliver a product, rather than validate the overall opportunity.



Overview of an Agile development workflow, taking desired outcomes as input, and performing tight iterative loops to ideate, build, test, and evaluate until the outcomes have been achieved

In Agile organizations, the product manager is highly cross-discipline and will work deeply embedded into the development teams, usually alongside technical and design leads. There, they must focus on motivating and communicating the why of the team's contribution, whether it's directly developing the app for the end user, or developing key internal platforms which enable the wider products.

Agile gave rise to the "Product Owner" role, which has informed much of modern product management. However, in some circles this has become distinct from the product manager by being less strategic and more execution focussed. While that may make sense under specific conditions, don't overlook the risks introduced by creating a gap between the strategy owner and the knowledge embedded in the product team.

In Agile organizations, the product manager is highly cross-discipline and will work deeply embedded into the development teams, usually alongside technical and design leads. There, they must focus on motivating and communicating the why of the team's contribution, whether it's directly developing the app for the end user, or developing key internal platforms which enable the wider products.

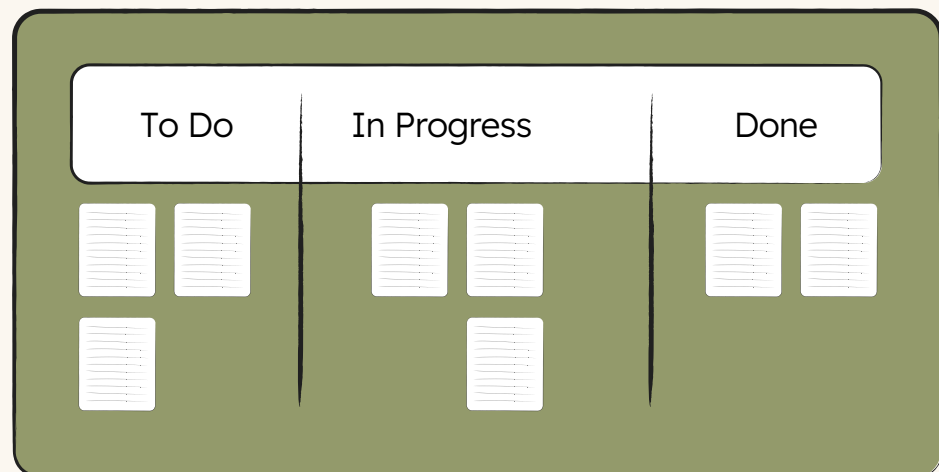
Agile gave rise to the “Product Owner” role, which has informed much of modern product management. However, in some circles this has become distinct from the product manager by being less strategic and more execution focussed. While that may make sense under specific conditions, don't overlook the risks introduced by creating a gap between the strategy owner and the knowledge embedded in the product team.

Agile organizations will most often be employing one of many frameworks. Many of them pre-date (and were input to) the formulation of the Agile Manifesto, and they all exist for the same purpose—to set appropriate boundaries, expectations and practices to realize the benefits in real-world product development.

It's beyond the scope of this chapter to dive into every Agile framework. However, two frameworks popular at the time of writing are Scrum and Kanban, which have some key differences:

- **Scrum** is usually recognisable from a set of rituals such as daily standups, retrospectives, and the existence of formal roles such as Scrum Master. Work is divided into “Sprints”, a fixed period (often two weeks) where the team is expected to ship an increment and hold a retrospective to reflect and incorporate learnings. It's a fairly literal implementation of the Agile principles and has established much of the popular Agile terminology.

- **Kanban** on the other hand prescribes fewer rituals, and instead focuses on visualizing the flow of work and delivering efficiency by limiting the work-in-progress. This is done with a Kanban board—either physical or digital—where every work item is represented by a card and flows from a “to-do” column, through “in progress”, to “done”.



An example of a simple Kanban board, where work items, represented by cards, are visualized by status and flow from left to right. More examples can be found at kanbanize.com.

Organizations and teams usually adapt the columns to suit (for example, by adding an “In Review” column before “Done”), and instead of releasing to a specific timed cadence like Scrum’s sprints, teams are encouraged to release-when-ready. (If this idea of workflow efficiency and release-when-ready sounds familiar, you won’t be surprised to hear that Kanban originated from Lean).

A common mistake when adopting Agile is to dogmatically impose a framework and hope that things will work out. Agile is intended to be highly adaptable, and any chosen rituals should reflect and accelerate the distinct organization’s own working culture. Taking a popular framework is a great way to jump-start and help everyone understand what Agile can mean in practice, but each team should regularly introspect,

To learn fast and iterate successfully, there must also exist a particular dynamic where the team doing the work not only has the authority to make significant decisions about the product, they possess all the expertise to deliver and have a direct and trusted relationship with the users to get timely, unfiltered feedback. More senior members of the organization must be able to set strong, motivating guidance without prescribing solutions, give up a lot of direct control and predictability, and embrace the team coming back with surprising and challenging discoveries.

Although popular and highly adaptable, Agile is not a silver bullet. Even when implemented well it can present challenges:

- **Knowing when you're done**

Continuous improvement through Agile's quick learning cycles and iterative delivery raises the question: How does a team know when they will be done? Agile's "Definition of Done" helps us know what criteria must be fulfilled, but an iterative approach means it can be very hard to tell when something will be done, and planning longer-term roadmaps with any certainty is a challenge.

- **Unnecessary or undesirable experimentation**

Agile teams sometimes forget that not all items need to be discovered and delivered iteratively. There may be well-defined parts of the product with clear value and known cost. Iterating towards what is already known can be a costly distraction and end up taking longer to deliver the desired value.

Iterative delivery is also not desirable in all software products. If your customers are large enterprise companies who need to test, integrate and roll out new versions of your product across multiple sites, it's unlikely that they will appreciate being given a new release every week.

- **Scaling Agile**

One of Agile's principles is to favor communication and collaboration over processes and formal boundaries, something which gets harder and harder to do effectively as a company grows. As successful Agile organizations approach enterprise scale, it becomes exponentially more difficult to manage bi-directional information flow and for everyone to maintain the same level of insight and involvement that was so important to their successes so far.

A human-centric approach must by definition work within the limitations of humans, and there is a limitation to the number of relationships we can reasonably maintain, the conversations we can have in a week (and remain productive) and the amount of information we can hold in our head. This is where things like the Scaled Agile Framework® (SAFe) and Scrum@Scale come in. Designed to create business agility at enterprise scale, these frameworks balance necessary agency for teams to do what works for them, while setting up certain rituals and processes which must be aligned for the overall business to function.

It's a tall order, and these frameworks are sometimes seen as overly prescriptive and complex. The introduction of such a comprehensive framework is a significant change management challenge and mustn't be approached lightly.

- **Going hybrid**

By exploring the three key product development methodologies, we've seen that each comes with different tradeoffs. On one end of the scale, Waterfall produces great predictability and reporting, but in doing so reduces flexibility and creates long lead times. Lean creates efficiencies, but with low margin for error. Agile enables organizations to adapt and

improve quickly by shipping and learning continuously and iteratively, but trades away predictability and more concrete longer-term planning.

The good news is, you don't need to pick just one. By opting for a hybrid approach, you can take elements from different methodologies and combine them into a system tailored to your organization's needs. For example, if the cost of change is high for your product, more considered upfront planning and signoff is essential (Waterfall). However, this does not preclude prototyping, regular customer feedback sessions and frequent retrospectives during development to learn quickly if elements of those bets were wrong (Agile).

Even if you choose to go all-in on Agile, different frameworks can be combined. For example, Scrumban is a combination of Scrum and Kanban which comes from combining the visualization strengths of Kanban with the planning and reflection rituals of Scrum.

Much of what we consider modern product management has come from the world of Agile, but a good product manager will consider all the available options. Which methodology an organization uses is not the responsibility of the product manager, but the cross-cutting perspective of the role can give insight into how the business is working, how it wants to work, and identify antipatterns.

Different methodologies are not either-or, and if you only take one thing away from this chapter, it's that you should not be overly dogmatic about how you enable your teams to deliver great products. Your organization, your stakeholders, your team and the particular set of challenges you are working towards all influence the most effective way to bring the most value to your customers, which should always be your number one priority.



Andy Smith

Andy Smith is a senior product leader with 8 years of experience. Passionate about technology, Andy started his career as a developer before moving into formal product leadership roles to help individuals, teams and organizations understand their purpose, discover opportunities, and unlock the full potential of our increasingly digital world in ways that customers love.



theproductmanager.com