What’s Your Problem?

Working Subtitle: Break into product management and build your career through mentorship.

Aaron Kesler

Disclaimer:

Opinions expressed are solely my own and do not express the views or opinions of my employer or anyone else.

Table of contents

Introduction

Just because you *can* do something doesn't mean you *should*.

You can write code, fix bugs, and solve minor customer issues, but in your heart what you really want to be doing is solving the bigger, more impactful problems, or setting the vision or the strategy for the products you're working on. You've spent years in the weeds of getting the exact technical specifications to work, while all the time you've been thinking about if this is actually something that needs to be made or if it should be made in this way.

I was where you are: building applications, taking orders, and focusing on the minutiae. It took some deep soul-searching and stepping back from a company I helped found, but eventually, I found my way. I've lead teams of developers, managing the production of multiple applications every year for massive healthcare organization, small startups, and recently acquired companies.

In this book, I'll share how I went from working at Starbucks to earning over $100K. I'll give you the process I've established for the entire concept to launch of a product. I'll provide the repeatable method I've developed for how to find and solve problems down to the root of them. And I'll show you how I product manage my career in order to reach higher and constantly improve. I'll teach you how you, too, can leverage your technical background in software development to become a product manager, getting out of the weeds, handling the big picture, and finding and solving the bigger problems.

My Development Journey

I'll be honest with you—I was an idiot out of college. I was twenty-two when I graduated. I thought I knew everything about everything and thought I knew exactly how to forge my career. While in college I started a company with my roommate, Scott, to develop applications for local businesses. Scott and I were both studying computer science at the time and wanted to put what we were learning into practice, beyond the boring projects we were being assigned in school.

We developed an app for the University of Delaware campus, went to some pitch competitions, and made some connections with people wanting app development. Through those connections, we got hooked up with another college startup company, now called Carvertise, that had just gotten some funding. Carvertise is a company that pays people to drive their cars while displaying advertisements.

Carvertise came to us with a problem. They were collecting so much GPS data and ad impressions but had no way of sharing it with their customers to demonstrate the value of their service. When we sat down with Carvertise's CEO, Mac, we asked him a ton of questions to discover exactly what problem he was trying to solve. We drew up a contract and got to work. Shortly after that contract was completed, Mac wanted Scott and me to come on full-time. When we graduated college, the company Scott and I created was acquired by Carvertise, and we were now the third and fourth members of Delaware's fastest-growing startup.

We were moving up, successful in our chosen field, and yet I hated it.

I began to realize I couldn't stand the nitty-gritty of software development. I loved our initial meetings, where Mac was a customer and I was gathering requirements and

empathizing with the problems he and his customers were having. I loved figuring out how we could solve his problems. After I joined Carvertise, I turned into a straight-up software developer. I went from a 50/50 balance of strategizing new business development and coding to being a code monkey with, at most, only 10% strategy. While I was buried in the weeds, I couldn't quite understand what I didn't like about this job. I thought it was that I now had a boss, or that Mac and I clashed heads so much on strategy. In actuality, I loved those difficult conversations. Mac and I both wanted what was best for the company, but we had different ideas of how to get there from a technical perspective.

What I realized after three years of working with Carvertise was that I just didn't like coding. I was sitting silently with my best friend, Scott, in front of a computer all day. We only talked when we were pair programming, or when one of us needed help with something. Spending hours debugging, with only the slightest reward of solving a pesky little error in the functionality, just wasn't for me. I was surrounded by people at a quickly growing startup, but I felt so incredibly alone. It wasn't the company; it was that I was doing the wrong thing. I loved having conversations with customers, hearing problems that we could solve for them, crafting roadmaps, or pitching which direction we should head in to our CEO. I just didn't want to have to code the solutions. I felt so isolated when I was fiddling with the code. I craved interaction. I missed seeing the reactions of people experiencing a cool feature we created for the first time. I needed connection, and I knew I would never get that as a programmer.

After that realization, I left Carvertise. I had spent three years with them and didn't have anything else lined up. I knew that I didn't want to code but had no idea what

to do next. I wanted to start my own company again. I wanted to get out of Delaware, and I just needed change. I interviewed at a few places for software development jobs and other technical positions but ultimately decided to work at Starbucks while I figured out what I truly wanted to do with my life.

Barista to Product Manager

I worked at Starbucks for a year, and it was without a doubt one of the most important years of my life. I learned a lot of the communication skills there that I still use to this day as a product manager. I also learned that I have a huge passion for coffee (certified Starbucks Coffee Master, thank you very much). The skills you need in retail are the same skills you can use in any other industry, solving any other problem.

Actually, I think Starbucks has pulled a lot of its training material ideas from *How to Win Friends and Influence People* by Dale Carnegie. The company wants you to connect with the customers by showing a genuine interest in them, remembering their names and drink, smiling, and saying “thank you” at the end of every interaction. While at Starbucks I was able to connect with a lot of my regular customers who in turn became my mentors. I built those relationships by using exactly the principles above. I think one of the things that allowed me to connect with these people so quickly was that I was genuinely interested in them, their lives, and, most importantly, how I could add value to their lives beyond a cup of coffee.

While I was working at Starbucks, I was also reading a ton of business books, listening to any entrepreneurial business podcast I could find, and talking to many entrepreneurs. Luckily for me, many of these entrepreneurs happened to be my

regulars. For my list of top business reads, podcasts, and audiobooks check my website (https://kesler.io/).

I had a very naive view of entrepreneurship until I started doing all this research. I had taken entrepreneurship courses in college and tried to start a company then too. I thought that if I just created some product I thought was cool, then I'd be the next Mark Zuckerberg and build a billion-dollar company.

The main concepts I missed, even though they were being hammered into my head by every entrepreneur I talked to, listened to, and read were:

- You don't have to think of a product idea. Instead, have your potential customers give you problems they're facing and create a product out of it.
- You don't want a billion-dollar company. What you really want is financial freedom.
- Focus on a niche you know or want to learn about instead of finding your passion.

So, I went back to square one.

While I was still working at Starbucks, I took a stab at starting another company. One of my best friends Doug Mullens and I chose a niche: landscaping. We started cold calling and cold emailing every company in the area to ask them about their problems. In these emails, we specifically said we weren't trying to sell them anything, since we didn't have anything to sell, and that we just wanted to talk about their problems to hopefully come up with some type of solution.

We got a ton of responses. Most people would get back to us, eager to share their problems. While this was an amazing exercise, we shortly realized that, while landscapers have many complex problems that need products, they weren't willing to pay all that much money for them to be solved. We also did competitor research and tried to see if the people we had talked to tried out our potential competition. Turns out most of these landscapers hadn't done any searches to see what software solutions already existed. After all that, we decided we needed a different niche.

Next, Doug and I met in a brewery called Lonerider, in Raleigh, NC, and racked our brains trying to figure out what an ideal customer for some type of software solution would look like. We came up with: "a small business that has employees and uses some kind of software already built for that business." While I was looking around the brewery, I saw the high-tech register and all the fancy automated brewing equipment and said, "What about breweries?"

We reached out to every brewery we possibly could in the Triangle (Raleigh, Durham, Chapel Hill), had a ton of meetings, and came up with a few ideas based on problems breweries currently had. Then we went back to try and validate the ideas with our core breweries. After we came up with the idea of a rewards system for breweries and heard from multiple breweries that they were interested, I designed some wireframes to better illustrate our idea to the breweries. Once they signed off on those, we hired our first developer. Six months later, we had an initial prototype on the App Store of Perky Pints, a brewery rewards platform the breweries didn't have to manage.

At this point, I knew I needed more money than the nine dollars an hour I was making as a barista. I reached out to my network of regulars at Starbucks and asked

them to sit down with me individually. I pitched them the idea and told them I needed to leave Starbucks. They all jumped at the opportunity to help me. They connected me with recruiters, contacts at local companies, and anyone who might help. Within two weeks, I was offered a Technical Support Manager position by a local startup in the data governance space. It checked all the boxes I was looking for at the time: it was technical, it was customer-facing, and I didn't have to code.

On my first day at the new job, I met everyone in the office including their sales engineer and their new product manager, who joined about two weeks before I did. I was super curious about what they did, so I asked them question after question. This product manager didn't last too long, but after learning briefly about the responsibilities a product manager held I knew I had found something I wanted to do. After the product manager left the company, I went for a walk with my boss and discussed taking on the responsibilities of product management. We quickly decided it was a good idea. She wanted to hook me up with their previous product manager who did the same exact thing, combined technical support management and product management. This product manager, Ankur, and I met on another walk with my boss. The second he opened his mouth, I could tell he was an expert in his field. I needed to learn as much as possible from him.

Shortly after this, I started contacting other product managers in the area to grab coffee with them and learn more about the job. I met a lot of good friends and mentors this way.

After diving deep into learning about product management with these new mentors one thing became apparent to me. While I liked the space and the subject

matter of the field I was in, I couldn't be a full-time technical support manager as well as a product manager. I needed to focus on product management as my full-time job. This small startup I was at had the CEO, CTO, and development manager driving the product. There wasn't all that much room for a true product manager. I did what I could to validate features with customers, gather requirements, and help craft a roadmap. However, I knew after a year I either needed to do the job fully or go somewhere where I could.

I quickly got a job offer at a massive healthcare organization for a senior product manager position in the data and analytics department. My manager and I hit it off in the first minute of the phone interview—he actually offered me the job on the phone. I was bewildered! Our passion for product management oozed out of the conversation. He started with the problems they were facing in the department: setting up product management best practices and having a ton of assets with no way to properly organize them. I knew I could get the department to where it needed to be, and it was done.

I stayed at this healthcare organization for a few years and it was one of the best jobs I'd ever had. I'd been a part of building the department's first full-stack software product in use by the entire organization where the estimated return on investment (ROI) on that product was more than $1.5 million per year. I'd established new standards for product management and software development in that organization, and was chosen to present at one of the largest healthcare conferences in the country. It had been an incredible couple of years.

Now I’m a product manager at a startup valued at over a billion dollars where I’ve been responsible for bringing multiple products from zero to one, and building out the strategy and vision for the company's future in a brand-new space.

I landed these incredible jobs in product management through a series of missteps, breakthroughs, and occasional luck. I am so grateful to be out of the weeds of coding every little thing, and I’m happy to provide you with the resources to do the same without taking the drastic measures that I took.

In the rest of the book, I will share the things I had to learn on the fly. Things that I wished I had a reference manual to go back to. While this book is not intended to be a textbook, I hope it’s something you’ll refer back to for quick refreshers. I will also share with you how to build out your network, find mentorship and build faster, more meaningful connections with people than you could possibly hope to from a networking event or meetup.

I’ll lay out a path for you to use your technical skills and find a more fulfilling job if that’s what you’re struggling with. I was so desperate to get out of the weeds that I was willing to do just about anything other than code. You don’t have to let it get to that point. See what you’re good at, lift and shift those skills to add value, and enjoy doing it.

Whether it’s utilizing your technical background for something other than coding, breaking into product management, or starting your own company with some type of software product. I’ll give you the templates and steps you need to figure out what kind of job to seek or what product to build. Learn from my mistakes and give yourself a leg up on the competition before ever spending a dime hiring a developer.

And finally, once you have a product you are managing—be it for your own company or someone else's—I'll give you the best practices for keeping that product going. Knowing how to manage both your customers and your team will be essential for any product or application you manage.

With that said, let's go!

Chapter One: What's Your Problem?

This isn't your typical product management book. I'm not going to go too in depth on things like agile, lean startup methodologies or feature prioritization etc. If you want to learn more about those things, check out the Additional Resources section in the back of this book. These additional resources will help you get up and running with product ownership and management methodologies. They will teach you how to be a great traditional product manager.

Right now, though, I'm going to teach you how to just start. We need to figure out what your problem is.

I find that a lot of people with technical backgrounds feel they can only be a developer, or be technical, and are afraid of product management. Usually, they think there is a lot of training or certifications, like an MBA, that they need first. Truthfully, you don't need any of it. You just need the mindset, which you already have, you just need to unlock it. This book will teach you how to get up and start doing product management, without needing to worry about being "by the book." How? One word: empathy. If you can empathize with your customer and their needs, then you can go build a product for them. And, if you can get others to empathize with you, your product will be greenlit.

If you can figure out the customers' problem and offer them a solution with your product, you are a product manager.

This book is meant to enable people to be product managers without having to be so worried about the "rules" that may or may not exist, or the credentials they don't necessarily need. I will also go over some of the credentials I have, and recommend some, just in case you're interested in getting one of them.

What makes you qualified to be a Product Manager?

Anyone could be qualified to be a product manager.

Many people have a fear of getting into product management because they feel like imposters, their ideas aren't good enough, or there is a lot to it. From my experience, these people fail to always put the customer first. Once you start putting the customer first every time, no one can argue with your ideas.

There are many different forms of product management and many different products, but all of them have the same core beliefs and concepts. The criteria you need to follow to be a great product manager are:

- Be the voice of your customer.
- Be extremely passionate about solving problems, or at least the problem you're trying to solve.
- Be able to ask very good questions—remember the five W's from elementary school: who, what, when, where, and why (and how).

- Be able to say “no.”

Let’s break this down.

Be the voice of your customer.

During my last week at that small company in the data governance space, I was nervous. I met up with their previous product manager Ankur one more time to ask his advice and guidance, as he just made the shift from a small company to a large corporation too. The piece of advice he left me with was, to this day, the most important piece of guidance anyone has given me as a product manager: “If you always argue from the customers' perspective, you will never lose an argument.”

This outlook takes you and emotion out of the equation in any argument. It’s much easier to argue on someone else’s behalf than your own. When you argue on someone’s behalf you’re likely not to get emotional. Defending your customer can get a little heated from time to time, but people tend to get more emotional when they’re arguing for themselves than for someone else.

Think about how a lawyer handles a case. Lawyers aren’t defending themselves; they’re defending someone else. Be a lawyer for your customer and keep emotions out of it. Emotions can ruin arguments because they aren’t logical, or fact-based. Once I started arguing on my customer’s behalf, I found that nobody could argue with me just for their feature, or what they think is “cool.” This approach also saves you from getting involved in any office politics. Because you're arguing for your customer, who likely isn’t in the room, you’ve shown that you aren’t self-interested or trying to kiss up to anyone except for your customers. I’ll go into this later on.

Be extremely passionate about solving problems, or at least the problem you're trying to solve.

When I first came across Mac, the CEO of Carvertise, at a pitch competition at the University of Delaware, he was the most passionate person I'd ever seen speak about their idea, their business, their vision. I even went to one of my professors and asked him how I could get that passionate about anything I'm trying to build. He tried to hammer into my head that I needed to be solving a problem for a customer, or potential customer. It also needed to be a problem the customer was willing to pay to have solved. For some reason, I didn't get it until I listened to a podcast called *Smart Passive Income*, by Pat Flynn, about a year later. One of the guests of the podcast, Dane Maxwell, said "So I don't really have a passion for real estate companies. I don't really have a passion for transaction management. But I have a passion for solving painful problems. And now that I have a passion for solving painful problems, I can take that and go ANYWHERE with it." (see the Additional Resources section for a link) It finally clicked. I don't have to be passionate about an idea, I just need to be passionate about solving people's problems.

Upon hearing that podcast episode, I realized solving problems is precisely what I'm passionate about. It felt like an entire world just opened up to me. Good products don't come from random "ideas" you think can be the next billion-dollar unicorn. They come from your customers or potential customers, and the problems they're having. This is how Doug and I went on to create Perky Pints. This is how my team at the massive healthcare organization built our department's first full-stack web application, and this is why I'm writing this book. I'm extremely passionate about solving problems. I'm extremely passionate about solving **your** problems! Whether it's that you're

technical and don't know what to use that background for besides development, or that you're a product manager and you're struggling with imposter syndrome, or you are struggling to find your next product idea. I want to help you solve these, and I will throughout the rest of this book.

Be able to ask very good questions.

In order to find good problems to solve you need to be able to ask good questions. Think of the way a good friend confides in you. Think of the way you listen. You probably listen intently with the deepest care and respect for them. You ask questions, you actively listen, and you empathize with them. I've been told by many of my friends it feels like they're talking to a therapist (in a good way) whenever they're going through a problem and choose to talk it through with me. I take what good therapists do and use those tactics on my friends. Good therapists listen actively, and keep you talking until you get to the core issue. It is only then that they offer advice or guidance. This is how I also approach things when I discuss customer problems to get to the root of the issue. It's only at the root where you can begin to try and solve.

Be able to say "no."

Sometimes solving problems means taking certain options off the table. A customer doesn't always know what is technologically possible and might not be able to identify what is best for them. A customer might not also know if a feature they propose is best for our business or our strategy. Why would they? That's our job, not theirs. Now, I know the word "no" can come across as harsh sometimes, and fortunately there are many ways to say it creatively without coming across like a jerk. I employ the, "I love

you, and no," mindset a mentor of mine, fellow author and business coach, Jason Goldeberg (https://thejasongoldberg.com/) taught me.

One way to do this is to tell your customer or stakeholder you'll put their idea on the backlog for discovery. This allows you some time to validate it, or test it, to see just how badly they want this feature. If they never ask about it again, chances are they didn't really care about the feature they just wanted to feel heard. If they do ask about it, then I've had enough time to validate with other customers, see if it's fiscally responsible and if it fits with the vision.

This brings me to the next, best way to say "no." Just say it. Kindly explain what exactly is wrong with their idea. If you haven't had any other customers request that problem be solved, just say so. If you can't figure out how exactly you'll get the return on investment from that problem being solved, say it. This takes you out of the equation. Even if you don't think it's the best problem to solve, let your customers be the guide. The problem they had may change or expand so that you get what you need, and they get what they need.

Even if you know you're able to be the voice of your customer, be passionate about the problems you're solving, ask very good questions, and say no, you still might not *feel* like you are qualified to be a product manager. In fact, even if you land a job as a product manager, you still might feel like you fooled them somehow. I've been there! This is called imposter syndrome.

Imposter Syndrome Sucks, Be an Actor!

Though it may not feel as though you are qualified to be a product manager, there is a good chance that it is imposter syndrome talking. Throughout the course of

my career, imposter syndrome has always crept in, in unusual ways. It can be like a snake that you don't detect until it's wrapped its way around your thoughts and paralyzed you. Imposter syndrome is, as the name describes, when one feels as if they are an imposter, a fraud, or a fake because they feel they don't know what they're doing or talking about. It is the feeling of not being enough. I'll tell you this right now, you are enough. Common things that I ask myself when I notice I have imposter syndrome are: "Why would they ever listen to me?" or "Why would they want to talk to me or answer my call?" or, before I held the title of product manager, "I can't be a product manager, because I don't have xyz years of experience or xyz certification." A lot of imposter syndrome thoughts are limiting beliefs that aren't real or aren't true.

You could see imposter syndrome show up in your life in three different ways: playing the part, memory blanking, and lack of qualification.

When I was much younger, maybe ten years old, as many young kids do, I played all of the sports. Basketball, soccer, baseball, you name it—I played it. I was pretty decent at soccer, and I was OK at baseball. Basketball though, I absolutely sucked at. I asked my mom what I was doing differently than the kids who were actually good. I especially asked about one kid, who seemed so good and ran up and down the court with so much confidence and poise. She looked at me and said, "Aaron, to tell you the truth, he's not very good either. He doesn't make baskets that often. He just acts like he's good. You need to play the part of a good basketball player. Act like one of your favorites, like Jason Kidd or MJ." After that, I had a couple of decent plays. I faked out one of my neighbors and made a layup. My neighbor's dad even went up to my mom during the game and asked where I learned to do some of that stuff. It's all because I

was acting like Jason Kidd. This has stuck with me ever since. We're all just playing parts. Acting.

I've noticed that the same feeling of "acting the part" applies in an office environment. Sure, some people know exactly what they're supposed to do all the time, most, however, don't. I was nervous about going to a 5,000-person organization, having never done product management outside of a startup environment. I thought there was no way I would be on my team's level. But I was wrong, and we all had things to learn from each other. Imposter syndrome really makes itself known when you think other people are so beyond better than you, or when you think you don't know as much as the rest. When I encounter these feelings, all I think to myself is what would a great product manager do? What would Marty Cagan do? What would Steve Jobs do? What would one of my mentors do? I play the role of a great product manager when my confidence goes down. I then find that it gets boosted, because it almost always turns out that I make good decisions when I play the part.

Another thing that could feel like imposter syndrome is when your memory blanks when trying to recognize your own work. You may find yourself looking back on your year and thinking, "What did I accomplish?" or "What do product managers even do?" A friend of mine asked me this about her own past, even though she accomplished an incredible amount of work that year. She got a product funded, wireframed the entire product and every flow in it, conducted design thinking exercises with her customers, hired a dev team, and even got an MVP out the door. That is a ton of stuff! When I listed all of her accomplishments back to her, she said, "Oh, I guess you're right…but none of that really seemed like work." This is the goal. If you love your job enough that none of it

actually seems like work, then it's the right job for you. Sometimes it helps to ask a friend or mentor what they think you've accomplished if you're having trouble seeing it yourself, or when your memory blanks on you.

One last sign of imposter syndrome you might experience is the feeling of being underqualified, or that you lack the necessary qualifications. Whether it's the lack of certifications, education, or experience let me ask you this, how does not having a certification or however many years of experience mean you're not enough? One question I get asked by people who want to be product managers and sometimes who are currently product managers is, "What certification should I get?" or "How many certifications should I have before I apply for a role?" The answer really is whichever one or however many would give you the confidence to apply for the job. A lot of job applications have "requirements" which, to quote Pirates of the Caribbean, "are more like guidelines, really." So, if it would make you feel better to get a certification, then go get a SAFe (Scaled Agile Framework) or Pragmatic Institute certification.

This is exactly what I did. I spent my own money on both of these certifications. One of them I got before I went to the massive healthcare organization the other I got while working there. The first one, Pragmatic Marketing (offered by Pragmatic Institute) was in person, and I made a lot of wonderful connections. I learned a little, but really all it did was validate just how much I had learned on my own. Pragmatic Marketing, offered by Pragmatic Institute, seems to focus more on what a Product Manager can do individually. Like meeting customers, tracking success metrics via OKRs, and coming up with pricing models, etc. The second one, SAFe, taught me a different way to go about product management, which was pretty valuable. The organization (or at least the

teams you interact with) has to be following SAFe in order for it to really work, though, which is why it might not be useful to get before you have the job. So, if you feel you must get a certification, but the company you're at doesn't employ SAFe principles already, then I'd go with Pragmatic Institute and pick the Foundations course and one or two other courses in the subject you'd like to improve on. If you just want general product management knowledge and to see where you stack up, I'd stick with the Foundations and Focus class and add the others in as you see fit.

Overall, I don't believe most people need certifications in order to apply for a product management job. If you've done the job or know how to do the job and have experience with it, just apply! The worst case that can happen is you get turned down, but I guarantee you'll learn a ton in the interview. This is exactly what happened when I started interviewing for product positions. I got rejected the first few times, researched all the questions they asked me that I wasn't fully sure of, and sure enough, eventually got an offer.

The same friend who asked me about what she had accomplished also asked me about the certifications she needed. My response, of course, was that she just built an entire product from the ground up, what more certifications could she need other than her actual experience? Most product managers don't get to do that, because they work on already existing products. End-to-end product management is a skill in and of itself. She could teach what she did. So, ultimately, I told her the same thing I'm telling you; "get whatever certification would give you the confidence you need."

Another part of feeling underqualified is when you may not have the exact title of product manager, but you're playing the role of one. You may think you can't put that title on your resume, or that you're not really a product manager because you have the title of Analyst or Developer. Don't be afraid to put the title Product Manager or whatever job you want, even if you didn't have the "official" title when you were at that company. One of my mentors gave me this tip. If you did the job and you know your field, who cares? Give yourself whatever title makes your resume look best. If you played that role, you can have the title. I give you permission! And, if you do list an employee there as a reference, just be sure to coach them on what title you said on your resume and what you want that reference to highlight before they call.

I'll let you in on a secret, none of my titles at any of the companies I have worked for were product manager before I moved to the massive healthcare organization. I've included a resume on my website for you to use as a template (https://kesler.io/). At Carvertise, I list myself as Technical Product Manager and Lead Developer. My official title was actually Co-CTO, but I found that this title hindered me rather than helped, because everyone I would interview with would assume that product manager would be a step down. At the data governance startup, my title was Technical Support Manager, but after our product manager left I started playing the role which is reflected in my resume, even though I never got the official title.

Don't be afraid to give yourself permission to be an actor or change a title on your resume. I'm not saying lie, by any means! I'm saying, if you can do the job or have done the job, then don't be afraid to show it and put the best version of yourself forward.

The bottom line is if you have worked on a product and have a passion for solving problems for customers, then you can be a product manager. Don't let the lack of a title or official certifications deter you. You can absolutely be qualified to do this job and get out of the weeds of software development without titles and certifications.

The most important thing you need to both land this job and perform this job is connections.

Chapter Two: Connections are Everything

When you start on your journey to becoming a product manager, it may seem like the easiest step would be getting a certification or taking a position adjacent to product management, but you can do without all that. Whether you are looking for a position with a company or looking for a problem to solve, the most important factor in building your career as a product manager is the connections you make with other people. You never know when someone you message on LinkedIn to grab a coffee with, out of the blue, can get you to your next job, or when some hallway conversation at the office can lead you to your next product.

Creating Connections Through Service

It isn't always easy to create these all-important connections, though. A lot of people go to networking events and such to try to expand their networks and create connections. They meet a bunch of people one evening and then nothing comes of it. A lot of these networking events fail at fostering meaningful connections. The best networking event I went to was a local product management meetup, and the reason it was so good was that they had us do a fun guided activity together to help break the

ice. They had us come up with a product idea and prioritize features using a methodology the organizers selected.

One of the people on my team and I connected immediately. She was just so full of energy, wanted to learn about product management, and had really creative ideas for features of the fake product we were building. She got up, presented, made everyone laugh, and was so excited about our hypothetical product. Best presentation out of the group, by far. Maybe I'm biased, but she absolutely killed it. We spoke a little after the meetup and she told me she was looking for a job. I offered to help her with her resume, and we pretty quickly became friends. We met up a few times to work on her resume, and we got to know each other better and created a meaningful connection.

I'm not telling you to become friends with everyone you meet at networking events; I'm saying the basis on which we network is flawed. It's transactional, "What can you do for me? How can you advance my career?" When was the last time you went to a networking event and actually kept in touch with any of the people there? You may have connected on LinkedIn, you like their posts, and you see them at other networking events, but you're not building anything with them.

I've found the quickest way to create a meaningful connection is to serve others. You need to provide value to the other person with zero expectation of getting anything back. This may sound obvious but let me elaborate. I offered to help that woman with her resume, and we met up a few times. After learning more about her I invited her to join my volleyball team, since she played throughout college and beyond. She invited me to her Halloween party and we actually started to have a real friendship. This wouldn't have been possible if I didn't first volunteer to help her with whatever she

needed. I provided value to her with no expectation of getting anything out of that and a real friendship blossomed.

I don't want to say treat people you don't know like products or customers, but the product management principles really do apply here. If you treat potential connections as if they're your customers, then you will make really good connections and possibly friendships. That value-first mindset is key. You then learn about them, ask them the series of questions we will talk about in this chapter and chapter three. What is the biggest problem they're facing right now? What is one thing they wish they could just wave a magic wand and make it disappear?

You don't have to find a solution for them, just listen. People really like to talk about their problems. Obviously, you don't want to become their therapist, and you don't need to. Showing a genuine interest in them, their wants, needs, and desires is a great way to build a connection. Most people don't want to go there when it comes to networking. Most people stick with the same standard few questions (where do you work? what do you do? How long have you been doing it?) when it comes to networking, so they can quickly work a room.

If you go to enough of these events and hear the same questions over and over, then you'll probably notice most of those questions are pretty self-interested. The person speaking is just asking for their own benefit. Those questions aren't really productive. They don't create a memorable conversation or connection. I've had those types of conversations at networking events, I've seen the same people at other networking events, and they will ask the questions again as if we were meeting for the first time! If you're not going to remember who you meet by asking these questions, why

ask them? Unless you truly are just waiting for the person to answer your set of questions “correctly” so you can figure out how they can help you. This is fine, that’s kind of the point of networking events, I suppose, but to create a genuine connection it needs to get deeper. After I went to enough networking events, it became pretty clear to me who was trying to work a room versus who was trying to genuinely connect.

When you talk to people about the problems they’re having or trying to solve, it slowly breaks down the hesitation or barriers people have when they initially meet someone. Once they realize you’re genuinely interested in what they have to say, that you’re listening, and paying attention they will continue to talk to you. What’s more, they will remember you. If you can also relate to their current problems on a personal level, in other words, empathize, this will deepen that quick connection. What I also noticed by taking this approach is that other people get curious about the conversation we’re having and then start to flock over to join. Which in turn creates even more connections than those standard boring few questions we’re used to. See, this is also somewhat selfish if we get down to that philosophical level again, but it isn’t a bad thing to be selfish if it means you’re creating a meaningful bond with someone. Just focus on truly trying to serve that person, without any expectations other than something meaningful.

So I encourage you at your next networking event or the next time you meet someone to ask them about the problems they’re experiencing. See what comes of it.

Learn Their Name

It sounds simple, but you’d be surprised how many people do not bother to learn the names of people they meet. One of the most important books I’ve ever read is *How to Win Friends and Influence People* by Dale Carnegie. It teaches so much about

human psychology and how to forge deeper connections with people. I read it right after I started working at Starbucks and it completely reinforced and partially changed how I interacted and communicated with customers. Carnegie's book helped me understand the reasoning behind the quick connections I was making with my most valued customers. A few of the main strategies I took away are learning customer names, being genuinely interested in them and their lives, wanting to serve them, and not being afraid to share about myself.

I have a few techniques to learn names. If it's a difficult-to-pronounce name I ask how it's spelled, and I write it down. If I don't have a pen and paper with me at the time I practice writing it mentally, where I actually visualize myself writing each letter. If it isn't pronounced how it's spelled, I write out the actual way I would say it. For instance, a friend of mine is named Yoelvis. I asked him how he actually pronounced his name and he said, "Don't worry, it's no big deal." So, when I met his wife, I asked her how she pronounced it. We all laughed, and she pronounced it like Jo-El-Vees. So, I wrote it down exactly like that.

Dale Carnegie says a person's name is the most beautiful sounding thing to that person in any language. If that's the case, it's incredibly important to pronounce peoples' names correctly, especially since most people won't take the time to actually learn how to pronounce difficult names correctly. This gives you a leg up on forging a quicker connection with them because it really doesn't take that much effort. However, what you get back for that effort is a high reward, a genuine connection.

Now some people use the excuse that they are "terrible with names" to avoid having to learn. No, you're not, you're focusing on too many things at the same time.

Think about when you normally learn someone's name. It's typically when you're being introduced to them. You're focused on making a good impression, what you're going to say, and trying to figure out how to connect with them. It's stressful, so their name slips in one ear and out the other. You might think, "Well I can just get their name some other way." or, "I'll just ask the person next to me later." If you remember their name and then proceed to use it in that first interaction though, that helps forge a quicker connection. Most people won't do even that bare minimum.

Actually Care

Showing a genuine interest in people is another quick way to build a connection. The problem is it can't be faked. You have to be genuinely interested in them, what they have to say, teach, do etc. After this you can relate to them if you have similar anecdotes that show common interest.

While at Starbucks it was common to take an order, get behind the Espresso Bar, and continue talking to the customer as you made their drink. We had an opportunity to converse because they had nowhere else to go while they were waiting. I'd ask them about their plans for the day, weekend, what they did for work, etc. If this customer became a regular, I began to learn about their passions, hobbies, and what they do in their spare time. What's more, some of my co-workers even got jealous of me, because they didn't understand exactly how I was able to be so well-liked by our regulars.

At my Starbucks location, we had this one regular customer who was kind of a jerk to almost everyone. He had a scowl on his face every time he came in and seemed to always be in a bad mood. A lot of my co-workers were intimidated by him, and so they never asked him about himself. He ordered the same thing every time. So, I began

by remembering his name, his order, and then, while I had him waiting at the bar, I would ask him questions about himself and what he did. The next time he would come back in I'd tell him I was thinking about our last conversation and find something to relate to, to share with him. Eventually, during days I wasn't working he began to ask about me. When I was working, some of my co-workers even went to the back room and told me to take care of him, since he was actually respectful to me.

Some people treat other people as "less than" until you serve them, connect with them, and show them you're a person too. I'm not saying it's easy to connect with a difficult person or show genuine interest in them, especially if they're rude. I treat it like a puzzle to put together. Why are they rude? Are they going through something? They must have friends or someone who likes them out there. If they're married, their spouse tolerates them enough to want to spend the rest of their life with them, so they must have *some* redeeming qualities. Show them that you actually care about the things they care about. Dealing with difficult people isn't always easy and treating it like a puzzle turns it into a game which can be fun to figure out.

Show How You Can Help

Creating a connection with service is far easier when the other person knows how you can serve them. I used to be pretty terrified to share about myself if people didn't specifically ask me about myself. I realized though through reading Carnegie's book, I needed to get over that and open up. Don't be afraid to talk about yourself when it's appropriate. Tell the other person about your hobbies, or what you enjoy doing on the weekends. Maybe you're learning a new language or you like to play volleyballl every weekend. When you share about yourself or can relate to other people it tends to

make that person like you more. It shows you're not just an order taker or some random person, you have hobbies, goals, passions, etc. For example, I had a few regulars who ran their own marketing companies and also did a little web development. So, I shared that I also used to do web development and studied computer science. They started asking my advice on tools, website development practices, and such. A few times on my 10-minute breaks they even needed some coding help. I wasn't hoping to gain anything out of this, I just wanted to serve them. However, this eventually led to my next job. One of my regulars who became one of my mentors said she had a client who was looking for a technical support manager. This all happened because I wanted to serve while expecting nothing in return and I wasn't afraid to share about myself when the time was right.

Service is a Mindset

Forging connections through serving others can come naturally since you already have a product mindset. Just like product, service is a mindset, and they are similar. Very similar. The product mindset is about wanting to add value to others in a repeatable and scalable way. The service mindset is more about wanting to serve others regardless of getting anything in return. They are both about solving problems for customers, friends, or people in general. Service typically solves an immediate need, whereas product solves a long-term need.

I took this service mentality over to my next job my regular helped me get, at a company in the data governance space when I was their technical support manager. It was especially difficult in the beginning because most of what we were doing was answering support tickets. To do this, I had to show my personality and connect with

people via email. I used a lot of exclamation points, asked people about their days, and let them know how excited I was to hear from them. Then when I eventually hopped on calls with them, I portrayed this level of excitement to them and continued to ask them about themselves. Just like my captive audience at Starbucks, when my customers were waiting for their espresso drinks when we had to wait for a server to reboot or the system to load up, I'd take the opportunity to ask them about their weekends, families, etc. Just like Starbucks as well, even the people who could be a little rude or abrupt on emails or tickets turned around, because they knew it was going to be me on the other side of the ticket, and now they knew me. A few of them even asked to connect on LinkedIn when I left, because they were just genuinely interested in me.

It was difficult for me to bring the product mindset to this company. Development resources were limited, and the product was trying to solve a lot of problems at once. This led to a very support/service-focused organization. We would get tickets and let feature requests come from customers, without really trying to dig deep into and solve the root problems. When we'd get a request, I tried to implement some design thinking principles and talk to the customer in order to figure out what their root problem was. I started reading every product management book I could get my hands on. Met with product managers at local product-oriented companies and got connected my predecessor at this company, a fellow support manager turned product manager, Ankur. He is the one who said, "If you argue from the customer's perspective you will never lose an argument."

Through all of this mentorship, and knowledge seeking I was really starting to make the connection between how a service mindset could scale into a product mindset. And I needed to learn more.

Learn From Everyone

Just as being of service can forge a connection with someone, so too can showing that person how they are of service to you. The best way to do this without asking too much of them is to show the other person *how* you have learned from them. Everyone has something to share or knows something you don't know. It's easy to learn from everyone you meet and build connections with them through that.

Learning is one of the main things that motivates me. When I first discovered product management I learned as much as I could about it through books, people, meetup groups, conferences, and trial and error of building my own product. When I was trying to improve our product at this company data governance company I read data governance books, attended conferences, and spoke to customers. Same thing when I went to a massive healthcare company, I took online classes about healthcare, talked to a lot of folks around the company, and learned about the internal and external problems. When I used to encounter members of ours, I'd ask them what they think of their health plan, our service, etc. They always had some pain point about money and costs, or how a claim of theirs was handled, and it was always really interesting to hear.

One of the things I've realized through my journey is that everyone, no matter their walk of life, has something to teach. Personally, I feel like people look down on others who are "less than" them. Whether "less than" in terms of status, money they

make, the job they have, or education/degree they have, people look down for a lot of different reasons. I've never really understood why that is, because everyone has a story. Everyone has something to share.

Earlier, we discussed being able to ask good questions when connecting with someone. If you know what questions to ask to get someone to open up, you can learn all manner of things. Be a therapist trying to extract the root of someone's problem by listening actively and by keeping your customer talking.

The five things you can do or say to get to the root are:

- "Tell me more..."
- "Go on..."
- "What have you tried to do to solve x?"
- The Five Whys
- Remain Silent

"Tell me more…"

This will make the person think about more details and issues related to the problem they're facing. Hopefully, they'll dive much deeper into the initial problem they brought to you. Therapists do this to try to get you to work through your own issues. I usually use this one in conjunction with another question. "Tell me more about that, why do you like apricot flavored ice cream." Or "tell me more about that, what did you use your smoker for over the weekend?"

“Go on…”

This signifies that you’re listening, and you just want to hear more about the topic. You know there is something more. I usually use this one if there is an extended pause in the conversation. This is especially useful on the phone if it isn’t possible to meet in person.

“What have you tried to do to solve x?”

This lets you know if they’ve looked into other solutions, potential competitors of yours, and how desperate they are to solve this problem. If they haven’t done much research about the particular problem, how would they find your solution in the first place? Remember the landscaping scenario? None of them even attempted to find a solution to their problem so how would they find mine, and why would they use it once they did?

The Five Whys

This method came out in the 1930’s by Sakichi Toyoda, the founder of Toyota. Today, this is a tactic used mainly for interrogation, as *why* can sometimes come across as harsh. Nevertheless, it’s a fantastic way to understand why someone has the problem they have. In this tactic you sort of act like a child asking, “Why is the sky blue?” You ask, “Why?” five times after someone tells you they’re having a problem. For instance:

“Brewery rewards systems suck!”

“Why?”

“Because I have to use pen and paper or spreadsheets to keep track.”

“Why?”

“Because the point-of-sale systems only have very generic non-customizable rewards.”

“Why?”

“Because they weren’t built specifically for breweries and restaurants.”

“Why?”

“Because they never asked us what rewards systems we’d like to have.”

“Why?”

“Because they built their systems for retailers and not us.”

There is so much to unpack there! Feel free to pair this tactic with a “tell me more,” or “go on.” This can lead to learning a lot about each *why*, until you get to that root cause or root why.

Remain Silent

This tactic is so incredibly simple and totally invaluable. When there is silence people begin to get uncomfortable. If you refuse to break this silence, then they will keep talking. The more they talk about their issue the more information you have and the more you learn.

From my customers at Starbucks to the consultants I worked with at the massive healthcare company, there is always something to learn from other people. Through their experiences and the way they carry themselves, there is an incredible amount of things people have to offer. For example, when I was working with those consultants I learned from the way they defended the features they needed to be implemented into a product I managed and the way they presented to large groups of stakeholders to get them all on the same page and for them to buy-in to the idea. The crafting of their

presentations was so well done and succinct, they understood that most people would be looking at the slides on their computers after the presentation, so all the information was there in a clear way.

These interview tactics and the skill of being able to learn from everyone are crucial in product management. I use these tactics almost every day. They are difficult to master, but with some practice, your customer interviewing skills will be unstoppable.

Selling Through Genuine Connections

At Starbucks, my customer Jason Goldberg, a business coach, fellow author and all-around wonderful human being, taught me about servant leadership and entrepreneurship in a way that no one had before. He is a huge promoter of serving others unconditionally, personal growth, and playful creativity being a part of work. He taught me so much about sales and negotiation.

During one of our first conversations I was talking to him about the beer app, that eventually became Perky Pints, which I was thinking about making at the time. I asked him about sales and told him how much I hated feeling like I was “selling” to someone. He had a similar experience with sales, until he realized it wasn’t actually sales he really hated. It was the concept of rushing a relationship. A good example is a slimy, used-car salesman. They want to get you in a car as soon as possible and get you out the door. When you think about a good salesman though, you probably think about how he made you feel, how he connected with you, what he talked about other than the sale, and how comfortable or personable he was. He also probably believed wholly in his product.

This helped me with all of my customer interviews, and eventually when we had a product. I went from trying to see if something would interest people and “fit” into their

lives, to solely trying to create a relationship with them. The goal became finding out what I could do to add value, whether it was my product or just information I'd found. For instance, a lot of breweries thought it was illegal to run a reward program in general. Regardless of whether they went with Perky Pints to manage their reward system I wanted them to be informed, so they knew they could run their own reward system. Some in fact did end up trying to manage their own reward system and realized how hard it was on their own. I didn't need them to be on Perky Pints. I wanted them to ultimately have a better business. Now, a lot of breweries don't use Perky Pints, but I have pretty good relationships with the owners I've met.

The thing about selling that I've taken away the most is that you need to add value to the other person's life. And not necessarily just with the product you're selling. Sure, the product you're selling needs to add value, but really, it's you. If you can surprise and delight them or give them something to talk about at their family dinner or next dinner party, you've added even more value. The thing about long sales is exactly what Jason said. It's about the relationship—the connection—rather than the quick yes.

Also, not all customers are worth the time. When you think about sales being relationships, you're not going to want to be in a relationship with everyone. You can choose who you sell to. If you think a customer is going to be extremely needy and not worth the time it would take to maintain them as a customer, why do you want them in the first place? Just as customers can pick and choose their products, salesmen can, and should, also pick and choose their customers. Now if you're selling a new product, you don't necessarily have as much of a choice. I was lucky with Perky Pints that the customers we had wanted us to succeed, not all products are so lucky.

What Jason ultimately taught me was not to rush building those genuine connections and to sell people on value delivered, not a product. If someone isn't ready to buy at a specific time, that means they don't perceive the value as worth it to them. It doesn't mean they will never find what you're selling valuable.

Be able to say "I love you, and no."

This is where another one of Jason's teachings comes into play. The concept of, "I love you, and no." Just as we discussed earlier, being able to say "no" effectively is a crucial skill you need as a product manager. No is a very important word in a product manager's vocabulary. In the same way that not every customer should be sold to, not every feature or product should be developed. As a product manager, get used to getting a lot of feature requests. A lot of the time my customers don't know just how much effort the feature they're requesting will take. If a feature is going to take two or three months of a developer's time and the ROI is very minimal, or it's only valuable to that one customer, "I love you, and no." comes in handy. Just because I tell someone no doesn't mean I don't like that person or there is anything wrong. It just means I can't do something for them, no matter how much I like them.

During my first few months at that healthcare company, I noticed quite a few of my stakeholders were emailing or walking up to the developers on my team to report issues. I noticed a lot of these issues were just user errors or something I could easily help with. My manager at my previous employer was ruthless with her developers' time. So much so that sometimes I couldn't even email them with questions, and rightfully so. I respected this because she understood that the time spent troubleshooting and dealing with users was time they could spend coding. I employed this tactic with my

stakeholders. I wrote an email to them and asked them to let me relay information to the developers if there's an issue. In the meantime, however, I'd be happy to try and help first and if I can't, it will be fixed when we have time to get it into our sprint cycle. In other words, "I love you, and no!"

Another good way I use this tactic is during roadmap reviews. Customers always want their features moved up, but we have a vision and a strategy to achieve that vision. So I bring up the roadmap with them and share all the things we're working on, with all of the dependencies mapped out in order to ship the feature they're asking for. Then I ask them to help me figure out what we can do to move their feature up. Spoiler alert: they usually can't figure it out. In other words "I love you, and no."

Through Jason, I learned how to establish effective boundaries by employing an occasional "I love you, and no." and I encourage you to do the same.

No matter what you're trying to do—get hired for a job, manage a product, build a product, anything—the connections that you make along the way will make all of that easier. There are so many ways you can help create and strengthen connections from being of service to learning their name, to learning from them. The bottom line, however, is that people like to work with people they like. Building interpersonal connections is the cornerstone of success in all endeavors.