

One Knight in Product - E114 - Dan Olsen

Fri, 3/4 7:34PM 40:33

SUMMARY KEYWORDS

people, product, book, talking, customers, pm, problem, product managers, prioritisation, dice, user, interactive fiction, job, tough, realise, management, point, idea, companies, world

SPEAKERS

Jason Knight, Dan Olsen



Jason Knight 00:00

Hello and welcome to the show. I'm your host Jason Knight and on each episode of this podcast, I'll be having inspiring conversations with passionate product people. If that sounds like the sort of treasure you want to seek out, why not don your magic armour and come on a quest to hear from some of the finest product thought leaders and practitioners in the world... oh, and me. You can GO NORTH to OneKnightInProduct.com, where you can sign up to the mailing list, subscribe on your favourite podcast app or follow the podcast on social media and guarantee you never miss another episode again. On tonight's episode, we take down a classic tome from the product management bookshelf, open it up and see if the treasure map inside is still pointing to product riches. We try to work out what sorts of armour you need to defend against rampaging HIPPOs, what mix of attributes you need to roll to have the best chance of success as a product manager, and how to find the right words to solve the riddle of Product Management. For all this and much more please enter the magical realm of One Knight in Product. So my guest tonight is Dan Olsen. Dan's an accomplished stand up comedian, product coach and author who wants work designing nuclear submarines before going deep into lean product management principles with 2015's *Lean Product Playbook: How to innovate with Minimum Viable Products and rapid customer feedback*. But the books Dan's really passionate about are interactive fiction alongside text adventures, which won't mean anything to anyone these days. But I can assure you, we've been bonding over that and have our collection of multi sided dice and complicated scorecards at the ready. Hi, Dan, how are you tonight?



Dan Olsen 01:32

I'm doing great, man. How are you doing?



Jason Knight 01:35

I am doing fantastic. Actually, I do have to admit that I can't find my multi sided dice tonight. But I'm sure there are some online generators that we can use if it comes to it.

 Dan Olsen 01:44

There you go, I took my kids to... I took my kids to Comic Con here in Silicon Valley. And they're like, "What are those?" So we got them a bag of the dice. So they both have a bag of dice that are ready to go.

 Jason Knight 01:54

Okay, as all children should. So... dice aside, first things first, aside from the book, you're also a coach by day. So what are you up to these days? And what types of companies are you coaching?

 Dan Olsen 02:05

Yeah, I mean, the main thing I do these days is private training workshops. Luckily, our world of Product Management has exploded in the last several years right. Now, he Marc Andreessen famously said, software is eating the world. It took companies a few years to figure it out. But they realise, hey, if you actually want to build successful software, you can't just have engineers, you need product managers. And so, you know, it's a red hot career. And there's not a lot of great places to learn product management. You know, it's kind of like you have to kind of learn by doing and so hands on workshops, like the ones that are offered are really, really popular. And so I basically spend, we're just talking the rest of this month, every week, I'm tied up doing private training workshops, with product teams, which I really love, you know, sometimes it's just product people, but a lot of times it's pm plus design plus n plus any other cross functional groups. And we actually work in the team. So people practice a cross functional collaboration. So there's a lot of fun. That's what I mainly do with a little advising consultant, here and there. And then also, the Lean product community that I've built over eight years now is up to over 11,000 people each month, I host a top speaker, and we get together. So and have fun. I just hosted Marty Cagan last month, hosting Geoffrey Moore on the 15th.

 Jason Knight 03:13

Oh, wow, it's a marketing thing as well. Cross functional. But you say software's eating the world. And that's obviously true. And I guess you could argue that also product management is eating the world because it has to to get the right software built. But would you say that based on the feedback or the experiences that you're having, doing some of this coaching? I mean, you must work quite a bunch of people, right? Is good software eating the world? Or is there quite a lot of bad software? And by extension, bad product management taking over? Let's go to games, right? Like, is there like a Pac Man of bad software, eating the world?

 Dan Olsen 03:49

Well, the problem is, if you think about it, right, how many different tech companies are there in the world, right, there's hundreds of 1000s At this point, maybe millions at this point. Each one has its own code base, right? Each I know some of them are using your common libraries

like jQuery or, or this or that or Node. But then they build on top of that. So everyone's building their own bespoke code base. So unfortunately, it's not like the arcade world where there's one mega hit or so I guess you could consider jQuery maybe, right? It's kind of you get into which languages or frameworks, but then that's not really that's more like a platform, because then you build on top of that. So the way I view it is there's like these hundreds of 1000s of instances of code. And to your point, it's probably a bell curve distribution of how most things in life are a bell curve. So some people are rocking the three sigma awesome code, and some people are rocking the fat part of the bell curve. So that's what you see racy, that's what that's what I see here, as far as the code and the products.



Jason Knight 04:49

But you must see some things like some quite horrible situations out there where you're working with really demotivated teams working on really poor products that are kind of collapsing under the weight of the world. In tech, they're all their own bad product decisions in the past? Do you get to go into some of those companies and turn them around or you work in a lot more with functional organisations or organisations that want to be functional?



Dan Olsen 05:11

That's a great question. It's it's the mix. It's a mix. And basically, yeah, let's talk about tech debt here. You know, none of the people I work with have any tech debt at all. I don't know what you're talking about. Jason? I don't know. So everybody has, it's occurred? You know, the weird thing is, it's a curse, the longer you're successful, the more tech debt you accrue, right? Yeah. You know, I worked on Quicken back in the day to quicken still around, right? Well, Microsoft Word still around Adobe Photoshop, you know, those, probably the longer your products around like you use whatever frameworks are cool back in the day, like prototype, right? Or something else. And now those are all gone, you got to use, you know, something else. So... so it's every once in a while, the longer you're around, it's easy. If you and I were to start a new start right now, we'd be using the freshest code, the most modern stuff, but then it just ages over time, right? And eventually, you got to pay that. So yeah, definitely a lot of my clients, you know, a lot of our clients, the big enterprise company, or big companies, and they have, they've been around for a while. And actually a lot of them as I like to say they existed well before the internet, right? Like, you know, they were around like AT&T, Wells Fargo, these guys were around before the internet. So they were successful companies. And so they have built in patterns and culture and things and processes that serve them well, in the pre digital age. And now they're trying to adapt. So a lot of my clients are like that, I do get us a percentage of people that are more like, Oh, they're actually at the two sigma level, and they want to take it to three sigma, that's always fine. So so I need each team where they're at, you know, and try to get help them get to the next level.



Jason Knight 06:43

Yeah, that makes no sense. I guess the question that begs is whether there's any people for whom it's already Game Over, and there's not actually a way for them to actually make the changes that they need to get there, and they're always gonna be kind of stuck in the past? Like, is that... Or have you ever been in a situation like that?



Dan Olsen 06:59

Well, the problem is, those people will probably not come seek me out like this. Right? If it's hopeless, they're probably not gonna say, Let's spend our training budget on this, right? So it's an interesting thing, because there has to be enough awareness that there's an issue or an opportunity to improve and a willingness to improve the people that come and see me. So it's actually a great self selection thing, you know, but certainly, there are some, in fact, all the time, right? Am I meet up at my talks, I get questions, we'll talk about how pm needs to be empowered, how product teams need to be empowered, and somebody be like, Yeah, but no matter what I do, you know, we're waterfall and it's political and nobody listens. Then I'm like, Well, maybe you should go work somewhere else, you know, at some point, at some point, if you, you should fight the good fight, the one sole pm on a horse against, you know, the whole company culture. At some point, you know, you should probably switch switch teams.



Jason Knight 07:51

Yeah, this is why I always feel conflicted whenever I do my mentoring, because half the time it's always like, yeah, just just kind of get another job. Yeah, okay. Sounds awful. And I guess you're trying to make people... Well, you're trying to help people get the best out of what they have. But yeah, sometimes it feels like there isn't necessarily a best and sometimes I say it's best to just kind of find a different mission, right?



Dan Olsen 08:12

Yeah. Luckily for me, my workshop class usually fighting the good fight, they see the light at the end of the tunnel, they know where they want to go. They need help getting there, basically, right. But back to your thing is the bell curve. Look, what if you're brand new PM, like, you know, we know it's hard to break as catch 22? How do you break into pm if you never done PM? But imagine you break into PM, and you get your first PM? But I'm just so grateful to have that job? You don't know? Are you in a like average situation? Are you in a dysfunctional city? Or you in a highly dysfunctional situation? Is it normal not to have a designer? Is it normal for the engineers to tell you to piss off? Like, if you have no idea, right? So part of it is you gotta like work in a few places to get enough sample size to be like, Oh, wow, man, that place was messed up by this. This is how it should be. Right. So so it's tough, but you certainly, you know, there's plenty of writings and people speaking and writing out there about what good teams look like. They are the minority, obviously, but they do exist. And if you can't have a perfect team, at least you can fix some of those aspects or find a team where those those aspects are good, you know?



Jason Knight 09:11

Yeah, absolutely. So the Lean Product Playbook. Now I'm sure a lot of my listeners have read that it's been around a while. But the book's seven years old now. So would you say that it still holds up? Or do you think there's anything that you'd add to it based on some of the stuff that you know now?

 Dan Olsen 09:24

No, I think it does it. It's selling it just had a record month of sales. And, and it's crazy. I don't know why today's January. Everyone's posting all these books and like, hey, all these posts on LinkedIn and Twitter have the cover and people saying I just discovered this book. So it's funny, and I think that because there's so many new entrants, so many people are coming into the product world that haven't read the books that are out there. They don't know the best books. They haven't read them yet. So there's like people discovering this. Yeah, it's funny, actually, I'm hosting Geoffrey Moore. And he wrote Crossing the Chasm. I'm sure you have it there.... Yeah so... but it's funny because a while ago I was at some university, and we were doing a panel on enterprise property management. And one of the panellists happened to mention Crossing the Chasm. And the whole class, his face was blank. And I stopped for a second like, Wait a minute. Yeah, I'm like, raise your hand. If you've heard across the chasm. No one raised their hand. I'm like, Oh, my gosh, there are these classics out there, kind of, if you will. So it makes sense. You know, people are coming in. But so yeah, it does hold off. It's kind of an evergreen book on how to build successful products. So that's basically what it's meant to do. And obviously, some of the tools, you know, like, Figma wasn't around then. But the general idea of rapid prototyping tools are in there, right.

 Jason Knight 10:33

So I guess on the other hand, it could be things in there that you wouldn't recommend anymore based on some of the experiences that you have, like, maybe you don't have to add anything. But then I remember speaking to Marty Cagan, for example, and he was saying that there was some stuff that he took out the first version of inspired, but he did the second one, because he just didn't recommend that anymore. Is there any stuff like that in the Lean Product Playbook?

 Dan Olsen 10:52

I don't think so. I mean, I basically, I'm very pragmatic, not dogmatic about stuff, right? You know, we all know, like, especially like in the Agile world, people can be very, some people can be very dogmatic. And I think being too dogmatic does not serve you well, because each situation can be different. I actually would add some stuff. There's some stuff. I mean, it's 335 pages, I would add some more case studies on value props. So if you see some of my talks or workshops actually have great studies, case studies from Instagram and Uber on value props using their actual material. And then the the big thing is to prioritise customer needs and how underserved the need is, I have the importance for success factor. I always wanted to kind of create a visualisation, I just didn't have time before the book was out. So actually, in 2018, I finally cracked the nut with like a heatmap visualisation that I think really solidifies it. So things like that

 Jason Knight 11:42

So maybe a version two coming your way. But you're a fan of interactive fiction. And I remember speaking to Christina Wodtke once who is a fan of such fiction, and we floated the idea of a product management game book. So how come you didn't write the Lean Product

Playbook as a Choose Your own Adventure style product? Paper, dice and character sheets and stuff like that? That seems like that could have been a niche you could have gone after?

D

Dan Olsen 12:08

Yeah. No, I'm glad that Christina as a mutual friend, she realised we both are interested in interactive fiction. I mean, I read those books to joint venture, they're great when I was a kid. I mean, my book emerged from my PowerPoints from my talk, basically, I was giving talks and frameworks and things like that. And so it emerged from that, you know, would be, you know, Christina told me every November there's the NaNoWriMo thing, right? So it would be kind of cool. And to write something, and, you know, when I read fiction that I get excited about, like, Ready Player One, or something that makes me want to explore doing some fiction, and Christina has the maths right. So actually, you know, like, I actually got a master's in industrial engineering, and we had to read the goal. Back in the day, I read the goal, which is a novel that tries to teach you, operations techniques. And she's like, the queen of writing these fiction books that teach you business stuff. So I think that would be fun to do. But, but it's funny, because yeah, my main my background rarefaction, just as a kid growing up playing the techspace games. And for me, the other called interactive fiction out, but for me, it was the ultimate in problem solving, right is like figuring out problems. And, and as I like to say, you know, that's the main job of a product manager is to define the problem, right? developers develop designers design. So if we had to sum up what we do in one word, I would use the word define, right. And it's about defining the problems, and then working with your team to come up with the best solutions. And I think, you know, when you were facing that cursor, and just the text, and that's all you had, is you versus that you had to really do good problem solving, you know, to figure stuff out and move along.



Jason Knight 13:39

Yeah, that's interesting. The idea of, I mean, I think I was framing it as like, what's the verb for product management? Because we said before we chatted before about the designers design, the developers developing, like, the product managers managing is just doesn't sound like anything does it is, I always get back to this whole question that I sometimes ask on the podcast, like how you describe product management to people that have no idea about it. And I think that the, you know, trying to define is an interesting one. But the point around problem solving and coming up with solutions is an interesting one, right? Because if you look at a lot of the classic literature, and a lot of the blog posts, and the articles and all the other stuff out there, it's all talking about the importance in a cross functional team of the product, people going out there identifying problems, I guess you could say that comes from the games, but then kind of working more with the designers and developers to actually define how to actually solve those problems. So kind of problem space versus the solution space. And there's a lot about talking how the product managers should stick primarily in the problem space and leave the solution space primarily to the other side. Like do you agree with that?

D

Dan Olsen 14:49

Well, it doesn't have to be so black and white divided. What I like to say is obviously and yeah, there's probably space. There's just space at play here, right? At the end of the day, we're going to ship working software that's in the solution space. That's it. develop. We hope that's

the developers domain. You know, the designers going to create the mock ups and what the experience is going to be with pixels, right? That's a solution space artefact as well. So those are solutions bases covered by those teams, right, those disciplines. The real question is, if to your point is, if the PM is going, yeah, let me do wireframes mockups. And there is there's no problem with them doing that, as long as they're dotting the I's and crossing the t's on the problem space. But if they neglect the process, so far too easy, it's too easy to get wrapped up in the mechanics of agile and things like that. And you're like, well, who's actually talking to customers? Who's figuring out the problems? So the message is not that a PM should not be involved in, in solution design. The message is... well, first and foremost, your job is to define the problem for the team. Right? So that's it. And that's not as simple as that, like we use the word identify. It's not that simple. It's not like the problems are just sitting there. And you have to go find all I found a problem. It's perfectly articulated. Here we go. So I think I like the word problem. Articulation right. Now, the other way I like to think about this is what I call the four Ds, there's discovery, definition, design, develop, right? Those are the verbs, right? Kind of like discover, define, design, develop. And again, designers are gonna design developers gonna develop, so who's really driving the discovery in the definition? And what definition means is the problem or the opportunity, as we were saying, so PMs have to do that. Then what I see happen is PMs, just get a PMs a tough job, right? If it's a very broad broad shoulders job, as I like to say, you know, a good PMs abhors a vacuum, if there's gaps in the team, they're gonna fill it right. Like, if there's no designer who's gonna do the wireframes, probably the PM is gonna step up and try to do it, there's no QA who's gonna do it. The PM is gonna step up and do it, right. So, you know, and all those things on your time and, you know, reporting to execs, and all this just can pull away from the core job of just are you out there talking to customers? Are you trying? Are you able to articulate the problems? Because when you talk to customers, just like the game, the text adventure game, when you're talking to customers, they don't go, Oh, here you go. Here's my perfectly articulated problem on a silver platter, just go solve this, right. In fact, they will lead you astray. They, you know, especially enterprise customers, you're a little startup trying to chase revenue, I call it chasing revenue, because you know, the clock is ticking on how much cash you have in the bank to raise your next round or get profitable. And you know, IBM goes, well, if you build this for us, we'll buy it. And they give you a spec. And then you go Yes, sir. Yes, ma'am. And you go and build it. And then lo and behold, you realise, hey, they didn't even weren't even clear on the problem they were trying to solve, this actually doesn't solve the problem. So you know, that's, that's kind of the whole problem, space solution, solution solution, as I call it. So it's really like, the PMs job to get out there. And just like, just like in the text, eventually trying to figure out what the heck do I have to do here? What is the problem, same thing, get out there and talk to customers come up with hypotheses of what the problem could be the same thing is a game you like I think this might this might be the problem, this might be the solution. And you kind of play around with it, you play around. But usually in a game, there's not much downside, if you do something wrong, it'll just say I don't understand what you're doing, or that won't work, or you can't use it that way or something like that.



Jason Knight 18:02

Right. But I think it's interesting as you extend that analogy even further, if we talk about that, during those text adventure games, sometimes you have to spend quite a lot of time not just going with the North, west, south and eastern stuff, but you'd also spend quite a lot of time trying to find the exact words to use exactly. To ask a question, or the exact words to use to

open the door whatever, like in Lord of the Rings. That speaks a lot then towards in the real world, trying to find that empathy and trying to actually work out a way to empathise with users that maybe you don't have a shared vocabulary of.

 Dan Olsen 18:35
Yeah, totally agree.

 Jason Knight 18:37
So is that something that you feel is kind of covered by all of these books that we have now? And something that product managers can just do? Or do you think there's still a lot of work to do in that area? To get people talking effectively, internally and externally?

 Dan Olsen 18:49
I think that's a great point. I mean, I think that's a great point. You're right. So these old these old games have a parser. And you basically put in a verb and a noun, like get lamp, right or something like that. And just old, if you want to geek out there's actually a documentary called Get lamp because part of these games is if you're in the dark, a monster called a group would come and kill you. And so you can be in the dark for more than a turn or two, it would say, if you were in the dark, you didn't have a match or torch or something, it would say you are likely to be eaten by a group. And actually one of my favourite nerdcore songs is MC front a lot has a song called you are likely to be eaten by a group or it actually the one of the coders of Zork Steven Gretzky's actually in the videos anyway, go down the rabbit hole there, but back to back to so but sometimes, like what do I call this thing? It's I don't know what this is, I don't know, this is you know, whatever word you're gonna use. Same thing with the user, right? It's happens a lot, where a lot of times the first version of your product that comes out, the words don't make sense. The messaging doesn't resonate with customer and the good news is the easiest thing to fix. You don't have to change any code, just change them to ASCII text and you change it right. This happens all the time. I saw this with my startup. And so yeah, getting the vernacular. You know, it really comes down to active listening And I really love how like, user research has become much more popular in the last few years, as people realise, you know, to build a custom box, you got to go out there talk to users, it's, you know, it's not just about showing the mock ups and getting feedback. So that's the funny thing that some companies that are still like, at the bottom half of the bell curve, they think that's enough is like, Well, we made the mock up. And we got feedback most like, Well, how about what did you do before the mock up even to understand the problems are great. And so it's really like active listening, listening to the terms that they're using in people say it a different way echoing back why what I heard you say, Is this Is that right, you know, kind of getting confirmation like that. And, you know, I think that's a good skill for PMs to have. Obviously, there are user researchers who are very talented in that if you're fortunate to have us researcher great. I also, though, don't like PMs just punting and say I've got a researcher, so I don't even talk to customers. As I like to say I've been there where I've seen teams, send the research team out, interview, the users come back, they send the report, so much gets left on the cutting room floor, like all the all the little juicy nuggets of the firsthand. So you want to have firsthand customer experience. And use active listening, for sure.



Jason Knight 21:07

Absolutely. Makes a lot of sense. And obviously times a lot of the other stuff that's been put out there at the moment with regards to doing that discovery by I think an interesting thing that I spoke about with someone who was talking a lot about interview techniques was alongside finding their vocabulary and trying to find a way to bond with them is even going as far as to mispronounce or say things the way that they say them, even if you know that they're wrong to try to make sure that you're not correcting them and that they don't start the close up. So really.



Dan Olsen 21:34

Yeah, I would never correct the user, I would never be like, oh, yeah, no, it's this. And another thing as a quick hack there, too, is show don't tell if a user starts showing you, you know, say you're doing some Salesforce research, you know, I hate it when I have to do a report in Salesforce, they have so many clicks, and it's so confusing. Level one is you go okay, great noted. Level two is, hey, can you actually log into Salesforce and show me what you're talking about? Right? Yeah. Instead of getting a verbal summary, you see the firsthand experience? That's very powerful, too.



Jason Knight 22:06

Yeah, I think he's really interesting thinking about the idea that is the natural or is the most natural thing in the world for people to get defensive, though, when someone says something that's basically wrong about a product that they are showing them, for example, so like, you're going to go in there, do your discovery, or maybe not your discovery, you'll be doing your demo or your prototype reveal to them. And they say something that you know, is wrong, because maybe your product doesn't do that. And you know, it doesn't do that. But they actually show that it does that. And then you're going to get really defensive and basically shut them down. So yeah, I mean, that's a natural human trait, right? So how do we stop product managers doing that kind of thing and shutting people down by mistake?



Dan Olsen 22:44

Look, what I like to say is feedback is a gift. Right? It's like a real gift. You can choose to not accept it if you don't want to, but you don't need to reject it in the moment, right? If somebody says, this is horrible, like, what's the point of everybody can have their own opinion about things? And if that's the reality, that's their perception, you're not going to change their mind, right? I mean, so yeah, I just... You got to have, you know, you obviously, it's tough, the closer you're aligned to the product that your baby, I understand it's natural human tendency to, as they jokingly say, it's like you're rooting when we do a usability test, everyone's rooting for the user to get through. Sometimes you can cross the line and help the user get through, you know, I've seen one time, the person can figure out where to click, and the moderator put their hand on top of the users hand on the mouse, move the mouse and click the button like that doesn't count. It doesn't count. You can't get assisted, assisted usability doesn't count. Right. So, you know, obviously, you want the test to go well, but we should all be seekers of truth, we

should be one objective, we should have like a scientist, kind of mentality, scientific method mentality. Like, we just want to know the truth. And you know, and the reality is, you may be talking to 10 users, there's 1000s, hundreds of 1000s out there that you are not going to be able to sit by and hold their hand and help them use your tool, your your product needs to stand on its own. So you need to know how it behaves standing on its own, not with you giving it a crash.



Jason Knight 24:01

Yeah, I bet that team that did that assistance, though, still voted, I was a success, though. We don't get to use that product. But in those game books we've talked about you have to at the beginning, roll the dice, generate a character with unique attributes, strength, stamina, all that stuff, kind of like Dungeons and Dragons. Yeah. And sometimes you get really good stats and kind of walk through the game. And sometimes you have all terrible stats and you die five pages into book. So if we then to reflect that tortuous analogy on to product management, what are some of the attributes the product manager needs to score highly entered the best chance of succeeding on their quest to develop amazing products?



Dan Olsen 24:37

Sounds good. So we can go through them strength is physical strength is not too important. Intelligence is important. You're not going to be casting any spells or anything, but it's all just wisdom is good. I mean, wisdom is really important. Because you know, you got a lot of times there'll be data, people talk about being data driven, but you always have to interpret the data. So wisdom is you're not going to be doing any healing spells, but you need wisdom. Dexterity is always handy no matter what. And the analogy of that in our world is probably being agile, right? Being up, don't get locked in. We like to say like strong opinions held loosely, right? You got to be respond to new data and things like that. And then charisma, of course, it helps, you know, charisma is always helpful. You don't really need charisma. But because no one reports to us, as I like to say, you know, with great responsibility comes no power, unlike Spider Man, you got to be good at influencing without authority. Right? So it's less about charisma and more about influence, like how can I, you know, how can I get these folks aligned in rowing in the same direction by painting a vision by by having a compelling why, you know, with data and rationale, right? So that's to go through the typical stats. I mean, basically, you need good communication skills, right? You're sitting between all the different functions. So back to the speaking the language, not only language, are the users any speak the language of the engineers? Was it a front end issue? Was it a back end issue, a CSS issue? You know, how big is this thing? You need to speak the language of the designers as far as you know, layout pixels, flows, marketing, execs, everybody, so communication is really important. I think one of the top skills that I always you know, a lot, a lot of times I'll help my clients interview or select. PMS is like prioritisation. By definition, there's way more ideas than we could possibly do. You're going to be bottleneck, you're kind of back to the goal. If you think of a factory line, there's going to be a bottleneck somewhere, you know, we should be bottleneck totally on development. Right? And the question is, what should we send those folks back to the thing? In the early days of describing my job, I would say my product management responsibilities, maximising the ROI on developer resources, like my job, we can ask them to do these 10 things, these 10 things, there's a million combinations of what we could ask them to do. Our job is what we ask them to do we have a high degree of confidence is actually going to create customer value. Right so

prioritisation is always important, because you know, and I like to use a ruthless prioritisation. So those are some skills. And the last one I'll say is, at the end the day actually, when I was like into it, this was the biggest differentiator because all the PMs that would come to interview for us had high stats on all those are things high intelligence, I was ecstatic. The one thing I kind of use the term dynamic range is some people are really good at the big picture. 30,000 foot view, right? Strategic high level, big picture view. Some people are really bad at that. But they're really good. The minutia the details, right? Oh, this is off by pixel, oh, this is misspelt? Those kind of things, right? And dynamic ranges? How what spread of that range? Are you comfortable? And can you connect the dots when the CEO or the GM says here's our strategic goal for the year? And then you turn around to your agile team? Can you connect the dots and say, here's why we're working on this feature in this sprint. Here's how it connects back. That's what I call dynamic range. So that's that's another one that I value a lot.



Jason Knight 27:49

Yeah, I think having that ability to zoom in and zoom out, but also tie a thread. Talking about slightly different games, Theseus and his labyrinth and stuff. But you know, tying a thread from the big picture all the way down to the individual ticket, let's call it I mean, I think it's really, really essential to make sure that engineers and designers feel completely involved in the process and that they're as strategic partners as possible, rather than just people that you just throw in a bag. I mean, you're over the fence every now and then. Because that never work well. But speaking then have more text adventure. hilarity? Yes. Let's imagine the big 500 pound ogre, walking around the corner, the ogre that is the most senior Highest Paid executive in the company that's come there to smash all your plans apart, because they believe that you should be doing something different. All your prioritisation that you've just ruthlessly done is going completely out the window, because they just come in and trampling all over. How can we defend ourselves against such ogres? And what weapons do we need to use against them?



Dan Olsen 28:53

That's tough. I mean, you know, you definitely want to vortal blade by your side. But let's see. So, you know, that's what I would call it. You know, that's one of the top things you see. And usually when I call this out, people realise this is happening. And it's not just exact. A lot of people do this. It's like shiny object syndrome. Yeah, something gets our attention, you know, and it's there's plenty of stuff. It's like, you know, what's our blockchain strategy? What's our chatbot strategy? What's our ML deep learning strategy, ML strategy, right. Like, it's like the it's like, it's like farcical. It's like, yeah, I was in the shower, listening to a podcast about blockchain. And we've got to drop everything and figure out how we're gonna do blockchain. Right? What's our blockchain play going to be? And you know, it's it's like, it's like the Dilbert it's literally like Dilbert. Right. So and it's tough. So the one thing is almost all every time somebody says we need to do X, it's a solution. Not a problem. It's technology is some new tech shiny avec tak. And look, it may have some applications, but you're not starting With the Discover In the Define you're starting with a solution, and then saying, okay, what can we do there? I mean, so many examples of this, and most all of them fail. Because they're not grounded in that. Right. So, you know, it's tough. So what the one thing you know, and it's a broader symptom of a lack of prioritisation or product planning, right? Because basically what happens is, we're already biting too much off, right? If you think about the pie 100. Now, there's a pie 100% of the pie. Most of the times your average product roadmap that you see for a quarter or a year, let's just

do a quarter. It's got like, 120% of the pie is on you're already at the beginning of the quarter, you're already over committed. Yeah. And by the way, people haven't gotten sick or left yet, right, the jobs. So you're going to be understaffed. It's just like D&D, kind of like when you roll the dice, except most of times roll the dice, something negative happens. It's not a it's not an even distribution, right? It's like, it's like with the Agile story. It could be eight straight points. Sure. Could it be five? Could it be 11 is much more likely to be 11. And five, it's a skewed distribution. It's not a symmetric dispute, right? So anyway, you know, basically, and it falls back to nobody ever says no, right? You're sitting around doing your quarterly product planning meeting. And as I like to say, you know, every feature idea, any one feature idea in isolation, it's like a cute little puppy. Oh, look at this feature. Who wouldn't want this feature? Of course! And you put that puppy down, you pick up the next one. Oh, look at this one, right? It's like, I'm sorry, we only have enough food and water for 10 puppies for the quarter. So you have to pick which 10 puppies are the most important, right? It's like, it's unfortunate, but nobody does it. It's so funny. Because it's kind of like the equivalent of what you would do in a sprint with your story points. You need to do it at the quarterly level, right? The units are not story points. The units are like people weeks or something or people sprints. But that's the last the big thing I see is companies are not many companies are not good at that. So they end up over committing 120% Before the quarter starts. Yeah, or 110. Let's just say 110. And then, you know, you roll the dice, and she gets some negative things happening. Since gonna delay and things and people quick, and then that'd be fine. If that was it, you might be okay. But then week two, week three, the quarter, that's when the the ogre comes in and says, what's our blockchain strategy, drop it and like to call these meteors like, because it's like asteroids just come in, and just blow up everything right? And they're gonna happen, they can kind of happen, you can try to minimise it. But the best defence you have is, okay, here's our roadmap with the what do you want to displace? We'll take what guess we're going to take the disruption of this asteroid hitting us. But what are we going to displace? The problem is when people don't displace, and they go, what is added in? Yeah, we're gonna add that in, we're just gonna put that in. And what you've done is, you know, there's like the triangle of like, scope, time, quality, you know, and resources. It's like you just, without explicitly cutting something, you just implicitly cut something. Those those features at the end of the roadmap, they're just gonna finally flip into the next quarter, no one's gonna acknowledge it, right? Or you're gonna cut a corner on quality, especially if you're, you know, the other thing I like to say Jason is like, it's amazing how much enterprise software ships on certain magical dates of the year, March 31, June 30, October 31. There's something magical about these days where all the features launch, right? It's like this, you know, because we got to hit the quarter, we got to do the quarter. And so what happens then you have a bad train wreck happening where you over committed 110% You've got these asteroids coming in every few weeks. You roll the dice, you got some negative outcomes. And then you got to hit the date, though. Still, you still got to hit the date. There's no relief. Right. And that's, that's how we get. That's, that's how we get train wrecks happen.



Jason Knight 33:50

Yeah, absolutely. And talking about rolling the dice, I was just wondering, since there seems to be a bit of a backlash against story points these days that maybe we could just roll the dice for the story points and just see what happens.



Dan Olsen 34:01

There is a backlash against everything, right?



Jason Knight 34:05

Always a trend going one way or the other. But you have to pick up the right equipment to succeed in any quest, but you can't carry too much in your inventory. Right. So part of that is around some of the stuff you've just been talking about around making sure you don't overload yourself. But part of it could also represent the inventory of the things that a product manager should be doing with their day to day job. So there's lots of things that we've talked about, that they should be doing. What shouldn't they be doing? What should they leave at the side of the road or discard because it's not really part of their job?



Dan Olsen 34:35

Interesting. Yeah. Well, first off, you want to get a bag of holding so you can hold as much stuff as you can. And a horn of plenty is good so you don't go hungry,



Jason Knight 34:42

But this has gone against your lean credentials, all these bags of Un-Lean.



Dan Olsen 34:47

Hey, if it doesn't take weight, it's all good, man. Let's see. So basically, you know, it kind of goes back to what we were saying is, the job can be infinite, if you let it right. Never we're not like you ever like, you know, it's funny because I have a friend, you know, they're a doctor, they see patients during the day, they have to follow up on notes. But their work can actually be done for the most part, they can be done, they can be done for the day, close the laptop, and be done. And then just next morning, pick it up. Yeah, our work never ends, right? There's always a customer, a stakeholder, a functional team member that needs something from us information, a task, something, there's always more research you could do. There's always more analytics you could do. There's always more refinement, you could do there's, there's always right. So so it's tough. So what I would say is, and you know, I mean, productivity stuff is cool. I don't necessarily go way deep into it, just do an audit of do a quick audit of your time, you can scan your calendar, right. I mean, it probably looks like Tetris, where, you know, I've seen PMs double and triple booked, and they've had to apply their prioritisation skills to be like, which meaning of these three? Should I go through? Should I go listen to the VP? Should I go? If they've got to decide on the fly? You know? So just do that. And then you know, it's tough, because if you're over booked, if you're feeling very, very stressed, you know, a lot of places have trouble with ratios. And it sounds very simplistic, but it is a good high level macro metric of what's the ratio of PMS to engineers, right? And if you've got kind of 20 engineers, every PM will no wonder they're gonna be frazzled and feel like they can't do their job effectively. There's no way there's just no way, right. And so you know, that's one thing is try to advocate for good ratios is plenty of advice from me and other people, unhealthy ratio ranges, you know. And then the other thing is, if you're doing more work, because there's a function missing, like QA, or UX or research, then you've got to advocate for that. And there's a back to this thing of best

practices. The reason your company doesn't have a UX researcher is because the management team doesn't aware of the importance of it. And so you've got to educate them and show you can show that that right. So the more things you can offload, like, you know, QA and UX design and UX research, then the more you'll be able to focus on the core things that you need to do right and, and you got a balancing act between outward facing to customers, right, and doing research and discovery and talking to customers and inward facing with your dev team. And so you've got to strike the right balance. So that can be tough. I've seen some people, they think the only part of their job is just the scrum stuff. And the ceremonies and they live in JIRA all the time. And it's a never talked to a customer. And so that's, you know, moderation is better than extremes of like, okay, I'm going to spend all my time doing that, and not talk to any customers. You rarely see the opposite. It's theoretically possible. I guess you could be able to talk customers all the time and never ever doing Scrum, but the scrum, because it's every two weeks, it tends to take care of itself. It's like the beast that must be fed every two weeks. It's gonna be okay.



Jason Knight 37:48

Yeah, I think is really interesting, this whole idea that there's always something more. I mean, I've been there in the past working late, always one more task to do and actually, probably at the end of the day, if you didn't do it, I mean, there's this whole 80/20 rule like, yeah, the amount of effort that you expend and the amount of actual value you get out of it. And yeah, I completely agree that we always talked about how we should be able to prioritise customer needs, but we very, we seem to be very bad at prioritising for ourselves as well. So absolutely agree we should be doing that.



Dan Olsen 38:19

Yeah, it's tough. And then one, one book I recommend there is actually Make Time by Jake Knapp.



Jason Knight 38:25

Is it a Choose Your Own Adventure book?



Dan Olsen 38:26

No. It's not. Sadly, it's not it's not. But it is a very practical book. And you know, there's all these different productivity schemes like GTD and all it's just, it's just I found it very helpful. So he, he's the co authors, they wrote the book sprint, so they came from Google and did the design sprint stuff. But anyway, this is focused on productivity. And I, you know, I found it very helpful. I consider myself pretty productive, but I found some good tips to go even more, be more productive in that book. So anyway.



Jason Knight 38:54

Sounds good. I'll make sure to dig that out. And where can people find you and ask you out

sounds good. I'll make sure to dig that out. And where can people find you and seek you out after this? If they want to go on a quest to find out more about you or some of the stuff that you're getting involved in? Or maybe try and persuade you to do version two of the book?

 Dan Olsen 39:07

Yeah, well, I speaking of course, I got to give a shout out You turned me on to the adventure called the hilarious these of these text games it is brilliant. So if you are into interactive fiction, you gotta check out Lenny's adventure calm. You can find me at my website, Dan-Olsen.com. That has links to like my talks on YouTube, my YouTube channel to the Lean product meetup that we host every month, Lean Product Playbook's on Amazon, it's available also in Chinese, Turkish, Polish, and Thai. So yeah.

 Jason Knight 39:40

I assume you translated those all yourself.

 Dan Olsen 39:42

Not at all

 Jason Knight 39:42

With an eye of all seeing power

 Dan Olsen 39:46

There you go. There you go.

 Jason Knight 39:48

I'll make sure to link that all into the show notes anyway, and hopefully you get a few people pop across or maybe you can even have another second record month or book. Well, that's been a fantastic chat. So obviously really glad that you took some time to share some fantasy themed product management titbits with me, obviously will stay in touch but yes for now. Thanks for taking the time.

 Dan Olsen 40:07

Thanks. It was fun.

 Jason Knight 40:10



Jason Knight 40.10

As always, thanks for listening. I hope you found the episode inspiring and insightful. If you did again, I can only encourage you to hop over to OneKnightInProduct.com, check out some of our other fantastic guests, sign up to the mailing list or subscribe on your favourite podcast app. Make sure you share with your friends so you and they can never miss another episode again. I'll be back soon with another inspiring guest. But as for now, thanks and good night.