One Knight in Product - E123 - Jim Morris

Sun, 5/8 8:26PM 37:28

SUMMARY KEYWORDS

product, people, users, discovery, prototypes, company, interviews, teams, silicon valley, solution, test, bit, executives, techniques, build, text message, business, talk, data, engineering

SPEAKERS

Jason Knight, Jim Morris

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. Now, when it comes to inspiration, sometimes we will need a little bit of help. If you follow me on Twitter, you'll know I'm a passionate advocate for mentorship. The first quarter of this year I've mentored 76 different people, but I've realised I just don't scale. Because of this, I've teamed up with a buddy to help more mentors and mentees find each other. So if you want to find out more about that, check out https://oneknightinproduct.com/mentor, where you can sign up to be a mentor, mentee, or both. That's https://oneknightinproduct.com/mentor. Check the show notes for details. On tonight's episode, we talk about discovery and experimentation. No, not creating our own product management Frankenstein's monster, but making the right moves and doing a solution test to understand whether people actually want to use our products. We talk about why some companies struggle with the very concept of discovery, how we might win sceptics around how to run good experiments and how to analyse the data you get back and turn it into action. We also ponder whether all these cool Silicon Valley startups we hear about are any good at this sort of stuff, or they're just stuck in the same boat as the rest of us. So there's so much more please join us on One Knight in Product.

Jason Knight 01:20

So my guest tonight is Jim Morris. Jim's a data driven sports obsessed product discovery coach who says he has been in over 600 user interviews in his time. I'm hoping he didn't come away from those with 600 different feature requests on his backlog. Jim says he's a convicted monopolist, which I definitely have questions about, as well as a former restaurant host and keen gardener who says he hates his cat staying out past dinner time. When he's not anxiously looking at the cat flap, he's busy getting teams to work together and helping product managers be successful, which he's doing with his own consultancy Product Discovery Group. Hi, Jim. How are you tonight?

Jim Morris 01:49

Jim Morris 01:49

Great, Jason, thanks for having me.

Jason Knight 01:51

No problem. It's good to have you here. I'm looking forward to what we're going to discover tonight. But before any of that I do have to ask you describe yourself as a convicted monopolist. So does that literally mean you were playing Monopoly? Got the go to jail card? Or do you mean something a little bit more interesting than that?

Jim Morris 02:07

No jail. But my company was number two in the market. The number one company in the market bought us, the US federal government decided that the combined company was a monopoly... monopolistic player in the market. And they took the combined company to trial, and they convicted the combined company mostly on the emails of the acquiring company. You really don't want to write down "If we do this acquisition, it will reduce barriers to entry. It'll increase barriers to entry and reduce pricing pressure on our sales deals".

Jason Knight 02:43

This is the sort of thing that gets brought up on that tech emails site that you see all the old Apple exposes and stuff. So you're there with all these other malfeasance that has been caused by these other companies. But did you get in trouble for that? Or slap on the wrist? Or had they treat you?

Jim Morris 03:00

For us as employees? I mean, obviously, founder of the acquired company doesn't necessarily have any individual impact. We did convert into, like shares and money into the the acquiring company. It just meant that we had to split out. Yeah, as a product leader was one day I was leader of a large group. And the next day, I was in a very small spin out company, rebuilding everything from scratch, because the acquiring company had let go most of the employees of my company in the intervening two years.

Jason Knight 03:32

Wow, there you go. What a story. That's a limited Netflix series right there. So let's get back down to business and pass go and collect our $200, or whatever it is these days. So you're the founder of Product Discovery Group out there in Silicon Valley. And I can probably take a guess at what you do with Product Discovery Group. But just in your own words, specifically, what problem are you solving with Product Discovery Group?

Jim Morris 03:56

Jim Morris 03:50

Sure. So I coach product leaders to create and grow successful product organisations mostly focused on connecting with their customers, and connecting with their data. And then as part of that, I work also with cross functional teams, as a group to teach them, like product discovery techniques that are setting like outcome goals, as well as recruiting and interviewing users on a regular basis. And then kind of putting that together to create solutions and test them so that they don't just make stuff up and give it to engineers.

Jason Knight 04:32

Never happens, right? But who's hiring you to come into these companies then? Are you being hired by the product teams? Are you being hired by the leaders? The execs way of the business? Are you kind of being hired by individuals to kind of come in via the backdoor, like, how do you actually get in there in the first place?

Jim Morris 04:46

Yeah, it's usually Heads of Product or CEOs. So there's no.. if there's no head of product, it's often the CEO that's usually in the smaller companies. Sometimes they're passing off product to their their employees for the first time. Right? So helping manage that transition. And we actually see those groups.

Jason Knight 05:04

Now, that's been going on for a while. I think that's been going for six or so years now. So you've been at it for a bit. But before that you've had an illustrious career across tech product, even spent a bit of time as a CTO from time to time as well. So you've obviously got that engineering and tech background as well. But you've now decided to double down on product product discovery product teams, as we just said, so I guess the question there is, what was it that made you have all of those things you could have chosen decide that it was product discovery and coaching your product discovery that you really wanted to focus on?

Jim Morris 05:35

I was lucky enough to be an engineer in the beginning of what we know is the internet and actually met Jeff Bezos, he sat on our couch in our engineering area one time

Jason Knight 05:47

Did he bring two pizzas with him?

Jim Morris 05:48

Yeah, you know, I'm not sure he wasn't really that famous back then. Amazon was only selling books, my company was selling sporting goods. So we were considering a peanut butter meets books, my company was selling sporting goods. So we were considering a peanut butter meets chocolate arrangement, I assume I was not a executive in this company. But what I learned as an engineer, and leading engineering groups, and product, because before there were product managers, it was people like us who were leading product is that we now can build anything, the tools to build them are so much easier than before I can rent servers, I can often find resources all around the world to help me become a team and build something. But the ideas that we use in this process, the ideas that we build, are just as naive and raw as they were 25 years ago. And this is the sad state of the industry is that we are continuing to build off and what comes to mind without really checking with reflecting these ideas off of the world, or checking if they're successful, even after we launch them.

Jason Knight 06:42

Yeah, again, I'm sure that never happens, but sounds like the sort of thing that would be terrifying if it did. But what sort of companies you're working with mainly then like you're out in Silicon Valley, so are you mainly doing a Silicon Valley tour and working with a Silicon Valley startups or scaleups, or you kind of working internationally and working with a bunch of different companies around the world, like who's your customer base, or your ideal customer base?

Jim Morris 07:03

Yeah, I'm the product leader coaching. It varies. It's quite geographically diverse. I have folks in England or Scotland all the way to East Coast and West Coast. Those hours tend to work nicely all around the clock, in terms of the teams they tend to be in... I mean, they vary from startups to corporations. Startups don't have execution problems, they typically have this two pizza team, but they often have an idea problem, that they take funding of the startup as validation, when really it should be validated by the market. Corporations often have both problems where it can be hard to execute, and going to be hard for product teams to actually have a voice in corporations typically have very strong sales, marketing and executives. And so product is growing and is new. And they don't quite know where to put it and how it forms a part of the like, successful part of the company. And so often what I'm doing in these companies is helping them helping product, make its place. And then of course, earn its place. We as product, people should always be earning our keep checking with customers, following our data, creating great deliverables for engineering that are based on reality, not just our opinions.

Jason Knight 08:15

But do you findm and based on what you just described... obviously, you're working with small and large companies ... do you find... I mean, I've worked for large companies before and I can definitely speak to some of those execution problems and the, kind of, top down management and, of course, all the different ways that big companies can suck at doing that kind of stuff. Do you find that that's something that you can make a very big impact on, like when you go in now, or is it just really kind of a case of almost incremental change, and just trying to make them a little bit better to try and push them a little bit down a better path, but there's only certain amounts that you can do in the kind of time that you get with these people?

J Jim Morris 08:48

Depends on the commitment of the leaders of the company. So when I work with cross functional teams, they soak up the techniques and the experience about working with customers and pulling data. And using data in conversations to convince people, whether it's qualitative, or quantitative leaders often have a harder time giving up their micro control of decision making. So as we talk about actually using data to make decisions, most people are trying to put in objective and key results frameworks, we have data as goals. And so that's really the, if the company is committed from the top to change, not just discovery is not something that is an isolated technique, like I can teach you to interview users. But if you're not learning from interviewing users, and you're not changing your mind from interviewing users, you should not be doing it. But when you change your mind, that means that a roadmap item might change, but that roadmap item might feel sacred in your company. And so it starts at the top by not making sacred roadmaps.

Jason Knight 09:54

Yeah, and almost combating the cognitive biases as well like the sunk cost fallacy is the fact that people have gone out there haven't maybe either fallen in love with an idea, or they've maybe even worse still made a commitment either to the board that they're gonna do this new thing, or they've got to go out and make a commitment to a client that they're going to sell this new thing or that they're going to do this new thing so that they can get the contract signed and stuff like that. I mean, all things that again, I'm sure never happen, but if they did, would be absolutely terrifying. But yeah, so you're... so you're a big advocate for experimentation and product discovery. You've talked about it just now. And obviously, that's the whole point of the product discovery group as well. But I guess, these days, it's not exactly controversial. Certainly a product circles, like it's very much in fashion, both through the efforts of coaches like yourself, other coaches that we're all aware of, as well as some of the popular books that are out there. Like it's, it's out there, and people are talking about this a lot. And obviously, in many cases, they've been talking about it for some time as well. But it definitely seems to be cresting a wave at the moment. But you've touched on it yourself as well, not everyone's doing it enough. Some people aren't doing it at all, across all of the types of companies that you work for, do you find that even if they're not doing it, and maybe don't really have a great idea about how they're going to do it, that there's at least an appetite to go out and do it? Or do you think that there are some companies out there that almost on a point of principle, don't do it because they think that they know everything, and they don't need that sort of thing at all.

J Jim Morris 11:14

There are definitely companies who believe they don't need that sort of thing at all. And these might be the companies that adopt a lot of these fake agile frameworks.

Jason Knight 11:24

Hey, here we go. Let's do SAFe.

Jim Morris 11:25

I know, it's a rabbit hole of discussion. But what it does is it takes something like agile, which is an inherently messy, incremental process. And with this inherently messy and incremental process, it doesn't lend itself well to leaders who want to predict the future, and who wants some control over the future. So when I say control over the future, I think this is impossible. Yeah, but people don't believe that. And so they want this fate control over the future. And so they take agile, and they put it in a framework, and it no longer becomes agile, but you still call it agile. And so it's a little bit of gaslighting, especially for the employees who thought they were going to be able to do incremental improvement, and come up with innovations at their level. But in reality, all the resources are there stage eight decision making by heads of engineering who won't let you control the backlog of the engineers, because they won't spend the time or don't spend the time with their colleagues to agree on important metrics for the company, which is the way you delegate without freaking out the way you macro manage rather than micromanage.

Jason Knight 12:31

Yeah, for sure. And obviously, that touches on another potential rabbit hole there, the whole horror of shadow roadmaps and engineering roadmaps and everything being kind of hidden. I was once called a transparency Nazi for suggesting that that might be a bit of a bad idea to do. But you know...

Jim Morris 12:45

Some people don't like work in progress, right? Some people expect you to be polished when you walk into any meeting. And they feel like it's a waste of their time to discuss the intricacies of whether you should or shouldn't do something. And I'm not a big fan of like wasting time. But I am a fan of if we're going to talk about whether we should do something. Ideally, it's based in some form of reality. And as we're doing it, why don't I show you so that you have some influence in the beginning, as opposed to surprisingly, at the end? This is like why we talk to users, you know, we can do discovery with our executives and with our users. And, you know, it's... that's one of Teresa Torres, his big thing is to be managing stakeholders. Just talk to them more often.

Jason Knight 13:25

Oh, yeah, for sure. Yeah. I mean, I kind of started to label that. And I don't know if it was me, we labelled it that or someone else who labelled it that and I just copied them, which is almost certainly what really happened. The kind of idea of full stack discovery and like, just not like you get some people to sit there. And they're so precious about the fact that they have to talk to users. And that's really the only type of discovery as valid. And if it's not coming through their product lens, then it's completely invalid. And just, you know, point in time, just one deal or whatever like that. And it's like, yeah, sure, it kind of is that. But of course, all conversations basically contribute to a bigger picture. And I think it's really incumbent on all product teams to to listen to everyone, right? They can wait the evidence accordingly. They can wait the discussions accordingly. But they can't just not listen to them.

J Jim Morris 14:05

Well, product folks. Often discovery gets a bad name, because it takes on a life of its own party discovery is only relevant. If it solves a business problem. We mean, you solve customer problems to the benefit of the business. Yeah, so what's the benefit of the business? Executives will say it's a stage gate decision making and we'll say... the executives will say... "I'm going to make money, more revenue or profit by doing the math in my head and assigning features to a product team". When in reality, they should take their top level metrics, break them down one or two levels, and start to dole those out to teams. Instead of these, these features. And then, if teams adopt this, this business attitude, their stakeholders will respect them more. Because they're speaking two languages, executives are terrified of not getting your paycheck in the bank by raising investment or hitting revenue numbers, and you get the luxury of getting a paycheck all the time, but it does owe it to the business to be related to your work has to be tied to the business. And I think that's where not only are we not we're not selling stakeholders. We're not we're doing discovery, but we're also trying to relate and make relevant our work.

Jason Knight 15:18

Yeah, no, absolutely. I think it's important. You know, we want those people to come towards us, of course, and start to think a little bit more like us and accept the ways that we think. But I think to do that the bargain has to be that we try and think a little bit how they think as well, right? And not necessarily in the middle, but not at one end or the other. But I'd like to think that Silicon Valley tech startups have all drank the Kool Aid in this regard, nor that they're doing product stuff properly. Like, that's the basis, you're shaking your head already. So I'm sure this isn't the case. But like, a lot of the classic books are based on the practices of people that have worked for some of the biggest Silicon Valley tech companies in the world. Right. So is it the case that those really good by the book, product discovery or product practice companies? are even in a minority in Silicon Valley? Or is there like just a strong minority of not that type of company in Silicon Valley as well?

J Jim Morris 16:08

Well, I'll first say that the the sad truth in product is that there are successful teams that have bad product managers, and no product discipline at all. And there are unsuccessful teams that have highly disciplined very good product people on them. It sometimes relates to whether you're in a rocket ship business or not, were you there at the right place at the right time. And what we do is to increase the probability of success and to turn ordinary employees into extraordinary employees, right through doing these techniques. So the issue of Silicon Valley is that you get a lot of external validation, when you receive funding, when you have free lunch, when you have a great environment. And really, you can get isolated from the fact that there are customers out there who need to pay for your services so that your company is viable in the long term. So I don't, I don't see it as a widespread concept in Silicon Valley that we're operating with a data first qualitative or quantitative data first approach. Sometimes we're just operating fast, hoping to see what sticks. And so there's a lot of great execution in Silicon Valley, that's what I would say is probably the number one asset is well execution. And wealthy people who give money to people they don't know, in risky areas. In the rest of the world, wealthy people give money to people they know, in non risky areas.

Jason Knight 17:27

Yeah, that makes a lot of sense. But it's also slightly disheartening that even in the heartland of tech, people are still struggling to do this. So what hope the rest of us have|? But speaking about that hope. I mean, we've talked about there are some companies that don't like to do this stuff, either on purpose, or they just don't know how to do it yet. Or they think that's a bit of a bind, or it takes too long, or both of those things that we just mentioned. But you're obviously going into these companies, and I guess at least in some situations, persuading them that that is the way to go. Now, you could argue, and maybe should argue that, of course, if people are coming to you that there's almost a natural appetite for that anyway, because someone who didn't want to do this probably isn't going to get you it in the first place. But presuming that you've gone into at least some sceptical companies, or ones where the product team will maybe up for this, but the leadership team weren't and you need to help them embed that in and actually get the buy in to do that. What are some of the approaches that have worked best for you to kind of crest that hill and go down the other side into the wonderful meadows of discovery?

J Jim Morris 18:27

I get them to testing with users within 30 days. And sometimes that requires me to write the recruiting screener for the users. Like I don't like to do work for my clients, but I have longer engagements. And so what that allows me to do is to have them do some things, have me do some things and get to this point of talking to users. And once they talk to five or six users, they start to understand the approach, which is I'm not going to show one solution, I'm showing multiple solutions, which is a required part of my approach. And then I'm going to talk to users in a way that allows them to react to things so that I get more honest feedback. And when they see the honest feedback engineers who say what can we learn from five users start to understand what they can learn from five users can't learn everything. But Jacob Nielsen's quote is zero interviews gives zero insight. So I do that. And then of course, the rest of my engagement, it becomes everything they do, they start to do on their own, so that by the time I leave, they're not really hooked on it. Because I've talked to users, because that's one of the hooking points, they're actually able to do all the various steps. The second thing is I convince management that as you grow, it's going to be hard to tell everybody what to do, and that you're actually better off spending 10 to 20 hours in your roadmap discussions. What if we took half of that and talked about metrics? Right? So the idea that there's a different way to manage? That's the hardest part is we teach discovery, we don't teach the different way to manage. So the managers, even though we criticise them for not allowing discovery, we don't actually work with them to understand that to teach them there's a different way to manage that can work to their advantage. Because even if you as a manager can convey business reality to your employees, you will be able to scale and become a great executive. But if you as a manager still want to be a pm working through other people, it won't scale. We are not Steve Jobs. We are not Elon Musk, these are not the examples to follow.

Jason Knight 20:20

Yeah, well, there's a number of reasons why we shouldn't follow Elon Musk, for example, although obviously, we'd own Twitter by now, if we were I guess, did it properly. Okay, so that sounds fair enough. And obviously, trying to make that more cultural change, I guess, sometimes works. And sometimes it doesn't, it really depends on the company and the buyer. And you can get, as you say, but assuming that you do get that buy in, and that there is scope, then to go out and do some proper discovery or get your five users or whatever it is that you want to achieve. If you're working for a company, that's never done it before working with a team, that's never done it before, it's quite possible that these, like, if we're talking to people, we're talking about people that have maybe been in a more of a feature factory, top down delivery type of product management role, sort of project manager, really not really a product manager, if they don't have the skills, they're not just going to be able to go and ask good questions of the people that they want to do discovery interviews with, because they'll probably go in there and reaffirm all their biases and lead the respondents and do all of that stuff that we're told not to do when we talk about discovery. So I guess the question that comes out of that is what techniques or methods do you go into these people with then to try and get them up to speed as quickly as possible to make sure that when they do go out and talk to these customers, especially if it's the first time and you're trying to validate the approach, that they're actually getting good results out of those interviews? And that they're asking the right questions, and getting something useful at the back of it?

J Jim Morris 21:42

Yeah, I've concentrated on one product discovery technique, which I call the Solution Test. And that's before you do usability, because you don't quite know if it's the solution you want. Usability is just as a confused people. Not do they want to use it. My teams don't ask the hard question often. Does anybody want this? And so I go straight for that question. You can read the Sprint book, it's not in there. So do you really want it? As we approach that existential question? We're also looking at the interviewer as a way to, well, I'll say that I, for the solution test, I have people create variety of solutions, usually fake prototypes, as a way to get reactions from users. When I asked you, Hey, would you use a concept and I describe it to you verbally? Maybe I show you a screenshot of it, you're not really participating in that concept. You're like in a movie theatre watching it? So the distinction is quite real when I send you the link, and have you interact with it and share me your screen. As I get users to interact with things, the reactions become more honest. And so the solution test for me, hits a very sweet spot in terms of authenticity. That is, in some ways, hard to mess up. Now you can sort of lead people through, but I do the first couple of interviews, right? So I give them the theory and model it. I watch them interview, and then I give them advice. It's typical didactic technique. And I get QA people, engineers, I get designers used to tell me they were terrified of talking to users. And it's sort of a script. Although once we get to the prototype itself, there's not much to say, except set up the mindset, give them the reason they're there, you need to buy an iPhone accessory, you need to pick a primary care physician, and then just stop talking. And so if we get to awkward silence, that I know we're getting somewhere, and when they get us are struggling, and they learn not to talk during that struggle, they realise that like making good software is hard. And that they should really step back and just have a humility type moment. So that's the in terms of avoiding leading questions. There's a structure I give them. The nice thing about a solution test interview, is it also helps validate? Is that the problem? Is the problem correct? Is the opportunity interesting? Do you have the right users that you've recruited? Are your personas good or bad? Is your job to be done, actually good or bad? And we put all this stuff together in the sprint book, they do it in a week, I usually do it in a couple of weeks. Because I'm not within you know, 24/7. But once they create this prototype, and I call them experiments, the experiment is meant to learn something not meant to get them to say yes, yeah. And so that's the, if they can get that approach, I turn everybody in the interviewers, I don't bother trying to make non designers make prototypes. We do if it's a Sprint week, then we have to do that. But otherwise it most teams, just the designer, but we will get into that whole flow, and then pass it to the designer, whereas most people will skip making that experience and they don't actually learn how to make a good experience. They'll just sort of make a bunch of bullet points and hand it over.

Jason Knight 24:48

Yeah, that makes a lot of sense. I was gonna talk a bit about experimentation as well, because I know that you've got a fiery passion for it. And we've obviously already touched on some of the elements of it. So I guess, if we then assume that some of these prototypes or mock ups, or whatever we want to call them are basically mini experiments that you're going to run with users to test the desirability, usability, feasibility, all of the stuff that we might want to test with these people. What are some of the key things to look out for? I was going to ask about a good experiment. But let's flip that and say, what are some of the key things not to do when you're putting one of these experiments in front of people if you want to get good results out of it?

Jim Morris 25:26

One, don't build it in software. Right? So people, when they hear about experimentation and discovery, they think immediately to a B testing, or multivariate testing and say, no, no, no, no, I've done a lot of that. That requires you to build two solutions, or five or six, in order to get one winner. If you release that to the public, the two to five solutions have to be privacy, compliant, secure, disaster recovery, compliant, it's just a lot of engineering time, that doesn't go into the value to the user. Once you know the value, sure, invest in that stuff. So don't build it. As much as possible. Most of my teams are making fake prototypes, whether it's a report API documentation, clickable prototypes, that's one thing is don't build it. The other is, don't just show it, there has to be an experiential element. We're not holding in our hands. But we're looking at it our own screen, we can interact with it, the fake report might be a bunch of Excel data that works in an Excel graph. Doesn't have to be a web UI that's fully functional. Don't make one long solution. This just happened with the team. I said, we're gonna make these experiments, we're gonna do multiple solutions. Everyone did sketching. We said to the designer, like, here's the sketches, let's make some prototypes. And designer sort of fused them into one long experience, which is what people do, because they're always asked not to make works in progress. But to make a final experience, if they were to go to a design review and say, Here's my three ideas. That would be frustrating for leaders who want to give a yes, no decision. Yep. So the ideas from the team had been incorporated. So the designer was listening. But as an artefact to show you, you would click through it, you would experience somebody to be like, "Yeah, okay". That's the response that you just have. That's interesting. I like it, I'd use that. And those are all lies. So what we do is we, I want you to slice it up, take one piece of the experience, if it's ecommerce, don't do, search and navigate, filter, pick something, read the reviews, add to cart, billing, shipping address, credit card checkout, give me add to cart, to me three variations of add to cart. What that means is I'm getting the variation from my designer, if I've got a subject matter expert, who doesn't build experiences, but has a great idea for Add to Cart, they're talking to me about how they can cross sell on the Add to Cart page or add to cart pages have been, you know, bloated and overdone at this point. But you get the idea that at some point, they thought, oh, let's innovate this page. for better for worse.

Jason Knight 27:50

Gamify!

Jim Morris 27:51

Yeah, so, so don't make big long prototypes, one because they make for long, boring user tests that don't give you results and don't do it. So again, if you're testing with users, and you're not changing your mind, or pivoting, just stop doing it, you're wasting time. Now, usability testing has its place like you don't want to confuse people. So it might take a final prototype that's almost ready for engineering, it's pixel perfect, at the very end of discovery, as we've been refining it, and run it through one more time. Sure, usability does matter. But not unless people want to use it. Okay, and then don't make a prototype that is like you would use it in real life. I coached designers to make prototypes that are ugly, because it may not follow the company style guide, if I have a call to action in the company style guide is really subtle. When a screen shows up, I don't want to tell the user that don't do anything, would you click this button, I want the button or the link to be so obvious that they have to tell me they're not going to click it or they're going to click it. So as we lead people through an experiment, I often do things to the thing that you're interacting with. So they don't have to talk to you. They have to talk to you, Jason. I've again, it's not a leading question. But I might want to explain myself, it's a natural thing to want to explain myself. And so I need to take myself out of it. And part of it is making the solution test that does this. Those are a couple of the things I don't overdo it. Don't over build. People will put a lot of buttons like they said, make it obvious. And they'll build the resulting pages for all those buttons. Well, if no one clicks on buttons 1, 2, 3 and four, and they only click on button five and six while next user test, build the buttons for six and seven, because they often think they don't believe or they don't understand that user testing is continuous that I'm going to come back to this topic. So I can talk I can actually tackle a subset of what's in my brain. Yeah, so the other you know, they make long prototypes, but that's because they're always trying to build long projects. Yeah. So if we take agile, the one thing that was successful in software is that we just build things and release them faster. Like engineers can't tell you why. We can't tell you like we just were bad at software. But if we do it faster, we're better at software. Actually in product, it works like that to our ideas, our dreams are big. But if we start to write them down and realise how actually bad they are, but if we break them down and do discovery in this, like, Let's do three versions of add to cart, add to cart becomes awesome, then we do three versions of billing and shipping. That looks great. And the users actually they can handle this. They can handle experience a on this page, and then like slightly different experience B and experience C, users are not timid or afraid. I mean, they put up with a lot. I mean, just go to Craigslist, eBay, Amazon, these are frustrating sites to us. They love these sites.

Jason Knight 30:38

Yeah. So when you get out of the back of the testing that you're talking about the experimentation, obviously, the interviews that you get to do as well, you're coming out of that, as we discussed earlier with some qualitative and quantitative data that you want to do something with. Now, obviously, we could probably talk all day about what to do with that data and some of the cool techniques you could use. But I guess, if you were working with a less than data savvy team, and you wanted to teach them some of the basics about what to do with that data, so they can actually turn what they've done into insight, like the only frameworks or techniques that you kind of start them off with to help them to do that.

J Jim Morris 31:17

Yeah, one reason teams do not do discovery, again, or why teams don't replicate a design sprint, is because analysing qualitative data, sucks, takes time. Yeah, especially if you're new to it. If I have an unstructured set of interviews, I talked about solution tests, let's talk about generative interviews where they're unstructured. If I have five or six of those, I have to go through half of those interviews. If it's five hours, call it two and a half hours of interviews, to understand what I'm looking for. And then that last part of that, I might find something that I want to go back to the first interview and find that there's a quote or a feeling about. So and I've checked with some qualitative researchers, that it's about one and a half passes through the data, which is five hours at seven half hours, although I do listen to people at a faster clip, just like me when podcasts as long as they don't talk like President Obama, which is slow and then fast and slow and fast. So it helps to be organised upfront before you start the first interview, especially with solution tests, because there's so structured, which is again, the reason I start there, because if you start with unstructured interviews, there's so much everyone can take their own understanding from it. But if we actually think about a couple solutions, and exercise our brain to what the solution might be, we're actually in a better mental state to evaluate someone's reaction to it, rather than these kinds of very raw interviews that are just most common in discovery. You know, hey, what do you think? What do you like what your pain point, let me follow you around for the day, these are hard to analyse. So let's just say that a structured interview is easier with a prototype, they're going to make one primary hypothesis about the study users, they want to, they want to do this specific thing. Then, on each of the screens we're doing, we're also making choices like, in the text message, I'm gonna say, Hi, Jason, in the text message, I'm gonna put the last four digits of a credit card or not, in the text message, I'm gonna put a link says, blah, blah, blah.org/this. And so even a text message might have five hypotheses in it. And we might make three or four different text messages to see which one users feel most comfortable clicking or some other action. So then as as we get to the interviews, for the known hypotheses, we can then just kind of check off yes or no, like personalization didn't like it, not by asking them, do you want your name in there or not just by saying which of these text messages or push notifications makes you feel most confident about proceeding on this task? Right, we might have to read between the lines, but we're gonna find one that works or not, in actually, a lot of people don't want to click on text messages, they will just go around to the website that you've mentioned that your car repairs ready, they're not going to go to this, they're going to click on that link from what is maybe the car repair company, maybe not. But that's okay, the text message had its purpose. So the that hypothesis failed. But the primary hypothesis of actually going to pick up your car when you get the notification worked. Yeah, so I give them the structure that allows them ideally, within half an hour of the last interview, if they're filling it out as they go, to make a distinction, four to five users agree with the primary hypothesis, or at the end of the interview. If I'm like interviewing is a live event. Did I cover that? And my colleagues on the interviewer can say, Look, you didn't know they can chat me. You didn't ask the big question. Would you be disappointed if you didn't get a notification that your car was ready? I asked that user says very disappointed, somewhat disappointed. There's some reaction. And then we note that so that sometimes we get through interviews and we forget to ask the big question. Yeah. And the big question is, wouldn't would you use this or not? It's, it's not that simple, because users often say yes, but they don't mean,

Jason Knight 35:00

The Mom Test, right?

Jim Morris 35:02

Yeah, that's a great, that's my great structures to hypotheses to hang it all off of. And I also have a technique I call the user analysis grid. It happens when I'm in when I used to do in person testing. And you can't take a lot of notes. But I want to make sure that I've covered each of the main points. And so I'd have users as rows. And my main points is columns. And I can usually fit five or six users data on one clipboards worth of stuff. And that grid is now in Excel or Google Sheets, when I'm online, which is most of the testing now.

Jason Knight 35:34

Some excellent techniques there. Where can people find you after this? If they want to speak to you about anything that they've heard on the podcast tonight? Or find out a bit more about discovery or tell you if they've seen your cat staying out late?

Jim Morris 35:46

Yeah. Do you see my cat around the corner? Please call the number on her collar!

Jason Knight 35:51

Have you put a GPS tracker on her?

Jim Morris 35:52

Yeah, I put an air tag on her. Yes, we love our cat and she's a little bit wayward. We'll see if that works for us as passive tracking. It's a pretty amazing device. And it hasn't seemed to annoy her. In terms of finding me, I do not wear a tracker. I guess my phone is a tracker. You can go to https://productdiscoverygroup.com. Or you can just Google product discovery group. And you can find me from there Twitter @sfjmorris, like San Francisco, Jim Morris, SF J. Morris, or LinkedIn. There's a lot of Jim Morris's in the world, pretty generic name, but you put product or product discovery after my name, usually work.

Jason Knight 36:33

I'll go and do that and save everyone the effort by putting it into the show notes. I didn't come and find you. friction free. Well, that's been a really fantastic chat, obviously, really grateful that you spend some of your time talking about some really interesting topics. And hopefully we can inspire some people to think a little bit differently about discovery or at least trying to do it in the first place. Hopefully, we can stay in touch. But yeah, as for now. Thanks for taking the time.

J Jim Morris 36:55

Yeah. Thanks, Jason. I really enjoy your podcast episodes, and I'm super happy to be on it.

Jason Knight 37:02

As always, thanks for listening. I hope you found the episode inspiring and insightful. If you did again, I can only encourage you to pop over to https://oneknightinproduct.com, check out some of my other fantastic guests. Sign up to the mailing list or subscribe on your favourite podcast app and make sure you share your friends so if 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.