

My belief is that like any domain or industry, the more it matters, the more the design matters, what happens is like whenever there's like a new paradigm, I don't know, it's like the mobile or the web or something. The first iterations of those products existing there, they don't have to be like super well designed necessarily because they are the first. But then like as as you built the hundred, like 1000, like different email clients, any email client now has to be like pretty good to be even considered like reason of like an email client, it's like that the bar is so high. So I think like today, it's almost like a very basic thing now, pretty much from the very beginning, you need like pretty high level design to people to even like pay attention or consider you seriously.

Today, my guest is Kari Saranen. Kari was the founding designer coinbase principal designer at Airbnb, co founder of two previous startups. And most recently is the co founder and CEO of linear, which I am fairly confident is the fastest growing and most beloved issue tracking tool in the world, and something that a growing number of product teams are using to build their own product. Kari and his team are building their company and their product in a really unique way, with a huge focus on craft and quality, no AB tests, no metrics based goals instead a focus on taste and opinions, also no durable cross functional teams, instead teams assemble around a project and then disperse once it's done. Also, they have just one product manager as the head of product. And that's it. In our conversation, Kari shares how he built a culture around quality and craft, how he makes tradeoffs, and how he operationalizes quality and thoughtfulness, where design can be differentiator and competing against incumbents. We talk about something called the linear method of building product, which is big on building opinionated software, working in consistent cycles amongst other principles. We also get into linear is unique hiring approach, which involves a paid work trial where candidates work alongside a team for a number of days, instead of just having an interview. Also, a glimpse into how linear got their first 10 customers, found product market fit and scaled their growth engine. There is so much Golden's episode, I'm very excited for you to hear it. With that, I bring you Kari Saranen after a short word from our sponsors. This episode is brought to you by Mercury, who I also happen to use for my business checking account. I've tried a lot of business

banks, and there is nothing even close to the experience you get with Mercury. I moved cash over from another bank, and it literally took less than half an hour to set up the account and wire money over at no cost. They kind of make you want to use the site more often, which I've never felt with another banking site. Mercury is banking engineered for the start of journey, a modern solution to help your company become the best version of itself. And Mercury isn't just a place to hold and money. It's software built to help you scale with safety and stability, whether you're team of two or team of 1000. Mercury also goes beyond banking to provide you with access to the foremost investors, operators and tools. Visit mercury.com to join over 100,000 startups on Mercury, the powerful and intuitive way for ambitious companies to bank. Mercury is a financial technology company, not a bank, banking services provided by Choice Financial Group and Evolve Bank and Trust members FDIC. This episode is brought to you by Composer, the AI powered trading platform now with retirement accounts. Algorithmic trading has historically been reserved for the hedge fund elite. Now with Composer, you can automate your trading with a library of over 1000 strategies that are easy to understand and tweak using an AI assistant and visual editor. Composer is the first ever algorithmic trading platform where you don't need any coding experience. It includes a full range of trading indicators for you to get creative and a discord community of 2500 traders to discuss your ideas with. Composer also has a powerful back tester to see the historical performance of your strategies and you can then invest with a single click. Once you invest, Composer will automatically trade for you based on the logic of your strategy with \$1 billion in trading volume and over 1 million trades executed. Composer already has many big time investors using the platform regularly. Head to composer.trade and use the code Lenny for an extra week of free trial on your Composer membership. That's [Composer.trade](https://composer.trade).

Kari, thank you so much for being here. Welcome to the podcast. Thanks Lenny. It's great to be here.

Maybe to start set a little context. Can you just explain what is linear? What does linear do? And then share maybe a few stats of just the scale of linear at this point.

So linear is the project and issue tracking system that software companies and technical teams love to use. We help like software companies to build software. We started 2019.

Today, some of the top growth companies like Block, Versailles, Ramp, Retool, Mercury and Substack are building with linear. We also additionally have lots of other companies like thousands of other companies using linear. These companies can be like very early stage team like some some companies just graduated from YC or a public company. And just like briefly why we created linear is that like, like you said, like we I work with you at the Airbnb and before that, I work at Coinbase and before that, I had my own startup and all of us founders, there's three of us, we had like a similar base where we worked in multiple different companies in different stages. And what we saw often is that like the tools available for managing software projects weren't that like really cutting it. I think a lot of them felt very conky or complex or or just like they had this kind of legacy way of thinking about software development. And so we just felt like we should do something about it. Then that's so with linear we we set out to do like build this like most frictionless and streamlined system for for modern software development. I'm also happy to share that we we've been profitable the last two years. And we also have this thing where we actually have this net negative lifetime burn rate, which means that like we just have more cash in the bank today than we have raised. I think a lot of startups like usually the normal ways is that like you you spend you raise money and then you you need to spend it to build it. But I think we since we were able to build a business pretty early on, we've been able to be in this position that actually we haven't spent any money on building the business.

That is insane. I didn't even know that. Okay, so for all those reasons, a lot of founders and a lot of product leaders look up to the way when you're built product and the way you think about product. And so to kind of frame this conversation, there's three areas I want to dive into. One is just how you approach building product. Two is how you go about building the team and the business in general. And then three is just how you grow linear. And to start, I want to talk about craft. Clearly, one of the biggest reasons that people look up to linear and use linear is the quality of user experience and the product. And I know your team puts a lot of emphasis on craft and user experience. I imagine that also comes at the cost of some tradeoffs, like takes probably longer to get stuff out the dairy, probably losing sales, because people are waiting for a feature and you're not ready, ready to launch yet because you

want to make it better. What have you learned about creating space for craft and building product that is really, really great?

Yeah, and I think it's interesting that those things you mentioned, like hiring, building business and building product and craft is like, I think that all of those are somewhat related to each other. But so what I can say about the product craft per se, it's like, it's definitely like starts with the people that like on the on the hiring front, we always look for people that care about it. As a business, like, why we really care about it is, is that, like, we see that cooperation only happens if people use the product and like our product, which is supposed to help the core collaboration coordination, if there's friction, or the experience isn't that great, or there's this little paper cuts, I think it gets really annoying for people to use. And so we think like, for the business to be successful, the quality is, and the craft is very important. There's definitely like trade off sometimes, like, there can be like, for example, timelines, like we, we are about to launch something. And then, for example, I or someone else goes to look at it, then and says, like, Oh, this, this doesn't like feel right. And we just like should fix it. So I don't think we should launch this now. So sometimes, like, it doesn't definitely push the timelines. But this, this might be like days. It's not like we need to like, redo everything. The other way we think about it's like, we are actually very okay, like pushing things out to ourselves and into for like a small group of customers, like if they opt into that. So whenever we build a new feature, one of the things is like, we don't want to spend tons of time up front, just like designing it and polishing it perfectly. Because we actually believe that when when you start building the thing, you actually start realizing more like how we could how it should work and how it should be better. So a lot of times with the teams, we tell them like, just, just put it there and not like, I don't know, the first week almost like, after you have some kind of designs in place or some kind of design ideas, just like put it into the app and like ship it to production, but only visible to us. So we internally can test it out. And then I think the next stage is like, we look for like, a customer that could be interested in this feature or, or we just like, ask people to opt in to like some kind of better program. And in those stages, the experience can be

a little like janky, or it's not that like polished, but we're okay with it because we are saying it's not finished, like we just want to get your feedback early so we can make it better. But like once we get to the kind of like it, the full general release, then we like, pay more attention to the actual like polish or the or the craft.

So that is so interesting. I didn't know you do that. So you actually go ahead and launch things really early to select a group of people that want the early stuff. And then did you say that you find one customer to kind of co create and help evolve the feature and change with?

Yeah, usually it can be one or it's like three or five or 10.

So it's often, especially with the larger company, like larger company facing features, we usually do try to find a large company because it's sometimes it's hard to like imagine these things how they should work. And so it's better if someone is willing to work with us to like explain, okay, this is how we do something and something like, for example, we work this way with Vercel that there was some some changes they wanted to see in the in the roadmap feature. So we work with them to to like improve it, then and then they could like give us feedback along the way.

That is so interesting. Because I think people seeing linear from the outside, it feels like you just take the time it need you need to build something awesome. And then it launches and it's amazing. And it's great. But turns out that isn't exactly a build. It's you actually do launch things really early. And people don't necessarily see it until it's done. But there's this whole process behind the scenes.

Yeah, I think like sometimes people can believe like think of that craft is is about perfecting things and like perfecting them in a in a very like, organized way or like very early on the downside with this like perfection mindset is that it can be sometimes hard to like put anything out because nothing is ever like fully perfect. And so we try to like balance this thinking with the fact that we should be always pushing things out very quickly. But then like also fixing them like improving them very quickly. So it's almost like the opposite ideas. But then we we try to like kind of combine that. And I think it's been it's been working well that generally in the company, there's not necessarily like a lot of reviews or something that we we always like review everything that like gets shipped this way. Because we do want people to like feel that they

can like put something in the app. And then we can try it out. So it's more like, let's just like try it out. And then but then like, yeah, we do need to look at it again, when before we release it to everyone.

A lot of founders, a lot of product leaders, a lot of designers definitely want to create space for craft and making products really great, something they're really proud of. But in practice, it's really hard. Very few teams and companies do this. Is there anything else you've learned about creating space for this sort of thing and prioritizing it for founders that are listening that are trying to instill more and more of this? Or do you have to be a designer CEO like Kari and there's really it's hard to do otherwise?

Yeah, I don't I don't think actually, it's it's it's not like purely coming from me. So I think they'll of the office founders, like, Jory and Thomas and additionally, their background is engineering. But I would say like, they actually have almost like, I think they spend even more time on the details than than sometimes I do. Like, I think, like, very early on, when it was three of us, I would be the one like kind of get doing the broad strokes designs, like this is how the UI works. And this is how the some of the things work. And that they were the ones that like, Oh, there should be animation here. And there should be like, this kind of thing here. So I think it's it's kind of like that DNA, I think comes from comes from all of us. And I think like with with the craft, it's always starts with like, people need to care about it. Like, if it's not valued in the company, then it's very hard to do for anyone to do because not it is people don't feel like it's it's valued. And I had I gave this advice to some founder, he was asking me about it. And like, in their case, like their founders were coming from different companies, like, maybe this one founder came from a European B and then the other ones came from my Facebook and and Amazon. And I think like these, like, for example, I think Facebook and Amazon have a very different culture on quality or craft or, or shipping. And I think like, where I said to them, like, you just like, you need to align on it, like, it's like, you cannot run a company with multiple different kinds of cultures. I think I made some points like why the quality is important for certain kind of products. And like, you should all kind of believe in that. And then like instill that with everyone you hire. The other thing I would say that what we like to do is, we actually don't

have much bms in a company, we only have one. And like, we can talk about more about it. But the one of the things I think that happens is like, when when you build a team, and you start creating this very specific roles for everything, where like, I think that often the PM can be the one that like figuring things out and like making decisions and guiding the team. But they're not the ones like building the feature. So they're not there, like, looking at it, like the whole day, it's like, how, how is this, where does this button go? Or how does it work? And I think like a lot of that like this graph for us happens when we give the project team this like ownership. And the project team is just engineering and design. And then when they start building that feature, they start seeing this opportunity is that this thing could be better. Like a good example of this is like we, one of our engineers, Andreas, like when we were building this right click menu in the app, like so you can right click different things. And under that you have sub menus, like Mac OS does this well, where it's like when you menu open that menu, you hover on the menu, and then you want to go to the sub menu. So you hover to the right, you don't have to go like exactly like horizontally to get into that menu, you can actually go kind of like diagonally, or like you don't actually have to hit the menu exactly, there's like this kind of safe zone. But like a lot of software like just implements like, Hey, let's do this menu, let's make the sub menu. It only works if you exactly hover over the menu. And then like what happens is the user often like misses like with few pixels, what they were trying to like do. So what the address did like, which we didn't tell him to do is like, yeah, this kind of sucks. And like, we should make this better. And so he figured out the way to like create those like safe areas that are dynamic based on like, like the sub menu positioning and everything. So it's much easier now, like you can go diagonally to the actual thing you want to go to. So I think like these kind of things like happen when you give people like more of the ownership of the project and also, like the space to do that. And then you also have like, leadership or or generally the company culture that that values the quality or the craft.

Alright, well, I got to follow this thread. There's a couple questions I want to ask. So you have one product manager, would you call him the head of products?

Yeah, non you who is who is the yeah, he's the head of product.

Awesome. So what made you decide to hire him and even have any PMs?

We started to see that. Okay, we have enough features and and like, like areas of the product. And also like the team is bigger, that it's hard to keep kind of like, align on all of these things or like even keep track of things. And like, initially, we actually hired none as a contractor to help us with this like insights data tool. So we have this like data to tool feature built in linear so you can get like data on what's happening in the workspace. And for us, like founders, we realized like none of us are like, we are not super experienced in in data tools. So we need someone to help out and luckily none was like we knew him and like he actually worked at mode, which is a data tool. And so we initially hired him like, can you help us figure out like, what exactly should this data tool and how should it work? Because I think there's different ways of doing that. And I think, always the easiest way is like, let's just copy what some other company is doing. But we didn't want to do that. So we wanted to like figure out like, what is actually like, useful way to to use this data or like get this data. So he helped us with that. And then we kind of saw like, yeah, this could be like useful in other larger areas or overall with the whole product is like, we might have this kind of questions, like, what should we exactly be thinking around here? Like, why and like, how would we like, define this direction and then like, help the teams to also like, align on it. So like, it to us, it's more like he's kind of like the figures out the direction of the product and and and like steers the some of the efforts and not like he's there in every meeting and like, making every decision or writing every spec or, or something like that.

Another question along the line, because there's a lot of PMs listening, they're going to be like, Oh, shit, these guys don't need PMs or PMs is over product management it. And so just another question along these lines. Somebody needs to do the work that a PM does. Basically, right, there's all these things that is on the plate of a product manager when they're at a company. And if they're not there, other people have to do those things. And what I'm hearing is basically, you give those responsibilities to the engineer designer and maybe other functions within team. Is that right?

Yeah, so I mean, definitely, I think what it means is, and there's definitely like trade off. So like, I think sometimes

when companies specialize roles a lot, it's like, because of it's more efficient, like if the engineer just spends 100% of their time like coding something, then it's like, they're using their skill set to the max. But then we just think that in order to like, build quality things or build things in a certain way, it's actually better if people actually also spend some time thinking about things and not just like executing. So yes, like for every project, there's a project lead that lead can be engineer or designer, it's not like a formal role, or it's not based on it, like, like your whatever level or or it's just like, you need to be certain title that to you can be that it's more like an assignment that like, okay, now, you're responsible, like getting the project started and working with together with the team, like figuring it out. And then like communicating when changes or like communicating how the progress happens. And so it's definitely like, it means that the engineers and or designers also have to like do these things. And like, then do you need to communicate? And then do you need to think about like, the scope or things? It's a different way of doing things. And, and also not everyone wants to do that, which is fine. But we wouldn't in the hiring from try to hire people that are kind of interested in the broader scope than just like the that they're like, specific skillset that they have.

Yeah, I think those last two points are really essential is one, people often don't want to do this work. And they kind of are happy to offload it to a potential product manager. So the fact that, say, engineers have to do all these boring PM things like communicate timelines and keep pure specs aligned and make sure timelines are hidden all that stuff and run meetings. A lot of people don't want to do that. A lot of times they do. And I think in this case, people seem to really want to be doing that. The other is I think you need a really high caliber team. That's very product minded. And the hiring bar needs to be very, very high for say, engineers and designers to want to do all these things and be good at them. And so I think I think those are two necessary ingredients for this to work out.

Yeah, sure.

For teams that want to try this sort of approach, especially startups that are kind of starting out and maybe not excited about hiring product managers. Is there anything else that you think is essential or important to functioning well without any product managers and you're at around 50 people at this point?

Yeah, we're around 15. Probably the only thing I would say it's like the hiring front that like you really need to spend more time on it. And like, basically, you cannot really interview engineers only for the engineering skills, you also have to interview them for like the product skills. And it's like, it's like, you cannot, I think expect that people have some like, if you do some kind of PM interview for them, it's not they're not going to have the same kind of skill set or the same understanding of the concepts or something. But like the way we have done it in the past is like, basically, I might interview them about the product. I'm not a technical person, per se. So I will just like ask them questions about like, how did they do something? Or like, how do they think about something? And it's similar to other roles to this, like we just look for like, do this person have like opinions about, about products and how they work and, and like, can they form opinions? And can they kind of use their own judgment at times? And then, like, can they communicate or articulate those things as well?

Awesome. I was just thinking it's interesting that a tool that I don't think is designed specifically for product managers, but essentially for building products, like the infrastructure for building product in a team is built by a company that has one PM and very few PMs.

Yeah, I mean, I think like, in some ways, I would say that like, with linear, we're trying to help the whole company. And like, I think engineers is probably like the largest user user group of the of the product. And like, I think in some ways, we want to make the BMS drop easier, like, they have to spend less time like managing everything, like where they're like the day to day, because like the engineers actually using the product, and they're like updating the things, it's okay, for the BMS, it would be much more easier to like get this, like, what is the state of things and, and like, maybe trust that much more because people actually use the product. So in some ways, like, I think we, we're trying to like, kind of like make it easier for everyone easier for the engineers, like they, they can focus more on their work. And then for BMS, I think, like, we're trying to make it so that that they can also focus on more on other things than it's like, I don't know, managing the tool that they use. Like, I think that's not the most important job of a PM, I think they should be like thinking more of the, I don't know, bigger picture or like other problems or like figure out like the next, like features or or something.

Just one more question along these lines. There's other companies like Stripe, I think waited till 200 employees to hire, I think their first product manager, Snapchat, I think is famous for something around that. Do you have any sense of if you think this might change when you think you might hire more product managers, their kind of plan here, or is it just kind of see how it goes and as you grow?

I think we will definitely hire more. It's like, I think like what I said before, it's like, I think we like to see in the PM smarts, like operating on a higher level, like the whole company, like, I think the way we try to build it is like, we have less people, but people who are more high caliber and can think about like, certainly like larger scope than what their current role is. So I think like, it's, we're just like trying to build this like a smaller units, but more effective units, which I think like where the PMs go and that they, they would be also less of them, like, and they're not there at every level. But I think in the future, as the company grows and the team grows and the product grows, we might have like several PMs that are focusing on or looking at specific areas or specific types of things of the product or specific customers or something like that.

Awesome. Okay. That was a tangent because I could not go in that direction. But I want to come back to design again and craft. So it feels like linear. One of the reasons you guys have been successful is design and experience is basically a huge differentiator from other products. And there's always this question of can design be enough of a differentiator in specific markets? Is there always an opportunity to build like a significantly better product experience and have a real shot at disrupting an incumbent? Do you have any sense of when design can be enough of a differentiator? And this is coming from like a founder trend aside, should we go big on design and experience or should we invest in like distribution or new technology or something along those lines? Any thoughts there? My belief is that like any domain or industry, the more it matters, the more the design matters. I think like it's kind of like it's fairly easy to see in different like if even in software or in other industries, it's like, what happens is like whenever there's like a new paradigm, I don't know, it's like the mobile or the web or something. The first iterations of those products existing there, they don't have to be like super well designed necessarily because they are the first. But

then like as as you built the hundreds, like 1000, like different email clients, any email client now has to be like pretty good to be be even considered like reason of like an email client, it's like that the bar is so high. So I think like, today, the startups like, I don't know if you look at like the web page Google launched with or like the web page YouTube launched with or some of these like older companies, it's they were very basic. If you launch that kind of website today, it's no one really pay attention. So I think like the design is always like, it's almost like a very basic thing now that like you from pretty much from the very beginning, you need like pretty high level design that people to even like pay attention or consider you seriously. And I think it's like it's not necessarily fair because like sometimes it's like, maybe the product is really good, but they didn't have a designer or they didn't have like time to do it. And then people just kind of like dismiss it because it doesn't doesn't seem like something that in the interest them. So I think like that that's the first thing is like, I think it's it is. And it's going to be more and more important. But I would also say like design is never going to be like the reason or like the reason why a company is successful. It's like, it's like the company still has to have some other things that like that the product still needs to be something it needs to be better in some ways or needs to be different in some ways. And then like a design is just like enabling some of these things like enable and similar to technology is like, if you have good technology, it's like easier to do certain things. And like, if the product works better in some ways, then you having like a bad technology or bad bad like infrastructure. So similar to design, I think like if you're if you have a good design and like people are or even like a good brand, like people are like drawn into it. And then it's like makes like some like user acquisition or user retention or just like even people perception of the product better. I think like an example is like packaging and products like Apple or a lot of companies spend a lot of time like effort into the packaging because it's kind of like already sets the expectations for the user who is receiving the product is like, even before you use the product, you like start thinking that this is a high quality product. And like, I'm gonna love it. And then like when you actually have it, then you actually like might feel that way, unless the product is really bad. And then like, you

wouldn't feel that way. So I think similar with startups or like SAS is like your landing page or some of the other things is like, they are already like communicating something to the user. And they setting setting the expectations. And then I think that that can be like very useful thing, especially early on when no one really knows you or or knows the product or cares about you. So I think like, especially, I feel like design can be very good leverage in the in the beginning. I think that's such an interesting insight, especially that the first thing you said around how the more often and the more crowded the space is, the more opportunity there is for design to be a differentiator that roughly how you think about Yeah, so I mean, just think about like, I don't know, any any product category, it's like, basically, people, people have then like lower choices. And then they like how do they make choices? Like maybe there's a specific thing they want, but like a lot of people don't necessarily know what what is the specific feature I want from this software. So it's more like, Well, what is the best one? Like what is the highest quality one? And if you put things side by side, and you just people see things people are visual. So like then like the design can be one of those things is like stands out is like, well, that looks the best or that looks the most like quality product to me. So I'm going to use that one. Like when people have a lot of choices, they probably will pick the one that like looks more, more interesting. And then I think there is the second part is the brand, which is something that you if you can build a brand, then I think it's like, it doesn't really even the product almost doesn't matter, like it's it's it's it becomes this kind of default like, I don't know, like, again, like Apple or Nike is like, yeah, there's all kinds of shoes you can buy. But there's a reason for someone to buy Nike shoes, other than some some random brand, even if the random brand would be actually a better shoe, they still buy Nike because they like, I don't know, like the brand. So then I think like both the design of the product, but also the design of the brand can be like, like very strong, like kind of things that like pull people to your company or to the product.

Is there anything you've learned about just building a brand over the course of building linear, something you find to be really important in actually building that perception of linear is really great and amazing.

To me, I think like the brand should be always like authentic and like it's just kind of like, I think even people if people can't articulate it, it people start to feel like something is off. Like, I think there can be like companies or startups, they like think about brands like old brand is the local or the colors of the website or something. And then they like do the same thing and some some other company does. And then they like think like, okay, now we have a brand. But like, you actually didn't like think about like what's your brand, like what is the message or voice you you want to talk about. And it doesn't have also like the brand doesn't happen overnight. So it's basically just you start in the beginning, like and and like when you start a company, you don't have no brand. And so you have to create it. So and you you create it over time by the the things you do, the things you say, like how you say them and like what what kind of how do you approach things? How do you treat customers? How do you build the website or product? Like, all of these things start to like build this like idea, like, what does this company mean to me in people's heads? I know, like, we both work with Airbnb and like, I think Francesca is like, I think the brand was probably the most important thing for him. And I don't know how many hours or meetings or conversations there was about the brand. And it's like the brand was always like, it's like part of everything the company does. Because it's true. It's like, yeah, you can you could book things like places to stay a lot of places on the web. But when people think about like, Oh, I want to like stay in some cool place. They're gonna think about Airbnb. It's like, they're not gonna like think about those other places. So that's like the power of the of the brand to like people stop thinking about the other things or or they start understanding like, okay, this is the thing for this.

And it's part of the reason Airbnb has been able to build a direct destination where people aren't like Googling, I want to stay in a home that are like Airbnb.com, which gives Airbnb such a massive advantage, not having to run ads on, you know, Facebook and Google or SEO, it's just like people know Airbnb, and I just go straight there. And there's very few sites where people go, I'm going to go straight there and look for some knowing that they can also compare hotels and all these other sites. Coming back to design briefly, just like very practically, how do you guys do design reviews? Just like, how do you actually go about reviewing what's going on? And

then to maybe to pick a question, but just whatever you can share is just like, what do you how do you know when it's done? How do you know when it's ready and approved? Kari sealed checkbox ready to go?

We've been doing like exploring different ways of doing this. I think like today, I still run the design team. So I do see some of the designs, like on a weekly basis. And and then like I, or one of the other co founders, or one of the, or the head of product, we are basically the sponsors for the projects. So then like, we are kind of like responsible, like check, like reviewing the reviewing the work. And so we might just have a meeting where we go through, okay, let's go through the demo and like people can explain what's what's going on and like how they think about it and why. And then we might have like feedback, okay, this seems like strange or or something. And then I might just after that, I might just like go into the product myself and like try it out. And then what happens sometimes, like, it's like, in the initial stages, like, obviously, we're not gonna like, start fixing everything. It's more like, let's try to get the like the main concept there and like figure out how it works. But then like before we are launching it, I might just go in like try it out and like try the different states and click it, click it around. And sometimes I find things like, like, we were building this threading to comments. And, and then like when it looked all good in the demos and stuff. And then, then I went to try it and like try different lengths of messages and stuff. And then I started to see like, oh, sometimes in animations are kind of janky, or it's just like, off, like, they don't go the right way. The, you know, on the screen doesn't scroll exactly, right? So then I just like, kind of like captured those things and like, send it to the team. And so we had to like, kind of pull back the release a little bit until those things were fixed. That one was like, it's very like, I think, like a simple concept. And it's like very known concept like, okay, this is how trading comments works. So that that was mostly about like, okay, what's the execution of this? But then we have projects where we are like, not sure exactly how this should work. And like, we can't really like, like, we can try it ourselves. We also have to see how how companies use it. So something like, we built this feature project updates. And it's like a common thing companies do is like, you need to write an update on a project. Is it yellow, green, red? And

like companies have very different ways of doing this in different tools. And we just thought like, well, I think it would be like really, really nice if it's like inside linear, and you can the team when they work on a project, they can write the update, linear can also capture some of the stats, like what actually happened. I think with that feature, it's like, it's been working well, but then also now we'll be like exploring. It's like, after using it a while, we think like, oh, actually, that it could be like more robust way of following these updates, maybe people could maybe the leadership could just like get this updates over email, or like, maybe when you have a lot of updates, you should have a search or like a filtering system or something. So I think a lot of times we just think like, okay, this is like the scope of it for now, and we're okay, like launching this, and the execution is good. But we know that like, this is not the like, fully figured out version, and we just need to see people trying it out and like, see that see the feedback.

So it sounds like on the decision of whether it goes out or not, it's kind of this intuitive feeling from your actual experience, trying it out, feeling gut level, this is ready, or this needs a little work.

Yeah, I would say like a lot of things that we do is is more like that, but we don't do AP testing, or we don't do, especially we go follow like certain metrics or something, we might sometimes we do have telemetry, or like, we can look at like how people use certain things. And we sometimes do that. But like, that's not usually the goal we have in mind. It's like, yeah, we should move this number as much. So it's more about like, based on the understanding of the problem we have, and based on the way what we think is right, is this the right solution? And is this a good enough solution to be released to to the customers?

One more question along this thread is, how do you actually structure these reviews? It sounds like you go straight to a prototype? Is there like a design review phase? Is it all kind of informal and people just review here's what we need your feedback on?

Yes, well, there's like projects don't necessarily have like specific states to them. But like, I would say like, roughly, usually we do start with designs, or there's like that, some explorations on the design, like, okay, there's like different ways of we could like approach this or sometimes

there's just one way because it's like pretty clear, clear. But then like, what I said before, is that like, we do try to like get into the building phase as quickly as possible. Because then we can also like see is this this direction actually like reasonable? And like, what what else does it call? Like, is there some problems it causes? Or how does it just generally feel here? So I think like that, that there isn't like specific like, review stages, it's more like, yeah, let's check like, if let's check on this project, like every week or every two weeks. And then before releasing, let's also make a like a review of it and like really test it out that like, is it that is it the quality we want?

Awesome. So that's a good segue to another area. I want to spend some time, which is the linear method. You espouse this way of building product that you call the linear method, which you publish online and willing to in the show notes. And I just want to ask a few questions around this way of building product. One is you are big on this idea of building opinionated software. Can you talk about just what does that mean? And then maybe give an example or two of how you actually have done that a linear.

So first, like with the linear method, what we why did we create in the first place is like, we just believe that there is more of like this modern ways of building software and thinking about it. And we wanted to like share some of our thinking on it. And that's kind of like also, it relates to how we built linear as well. Like so you might like, understand why we make some choices, because this is like the way we think about making these choices. So we're trying to like share our thinking behind the product and not just like, here's the product and like, figure it out. Yeah, so like the opinionated piece, like, I personally have like this belief that productivity software should be, and especially company software should be opinionated. I think that like, what the productivity software is trying to do is, is like, make people productive. And I think like what what productive means is like, you actually do something that matters for the company, which is, which is like, I don't know, build some new feature or like fix something or design something. Like all of those things are like, eventually, they provide some kind of value for the customer. I think there is this like ideas or notions in the world that like, flexible software is great. I'm like, I think it can be great sometimes. But like what happens is like

people start spending a lot of time like, figuring things out, like how does how does this feature work? Like you can use it in 10 different ways. And then everyone, every team or everyone figures out the different way of doing it. So our thinking is like, we, we like to provide this like good defaults or good opinions, like this is how the feature works. And this is how the workflow works. So you as a user or as a team don't have to think about it. And you can focus on the work you do. And the other thing is like, my design mantra is always like, design something for someone, like it's very hard to design everything for everyone, because there's you just end up with a very generalized solution. So then what we're trying to do with the opinionated solution is that like, that's the best solution or the most optimized solution we think of. And then like when you use it, like hopefully you agree and like you can feel that it's the most optimized. So being opinionated, it's like, I think the value it provides people is like, you don't have to think too much like, or spend some more time on the tool than you do on your actual work.

And then another core element of the linear method is something called cycles. And I know linear is all around this idea of creating cycles and working in cycles. You talk about what is a cycle, and how it works at linear.

So for example, like the cycles is it's optional, like not every team has to use it or not they have the whole company has to use it, but it's there as you can turn it on or off. But basically, I think the why we created cycles is that I think any team that works on software or some other products like you always have almost like infinite list of things to do. And that list gets longer every day. And it can be sometimes very distracting for the individual or for the team to like decide like what, like there's a new thing coming in like should we work on that or should we work on this other thing we've decided in the past. So the cycles is just a way to say that like, like for the next week or the next two weeks or whatever timeframe, we're going to work on these things. And like these are the things we think like are the priority or the focus for this timeframe. And then the team can kind of try to focus on those things. Now, if something happens, like, like, I don't know, we really need to jump on this other thing. At least there was like some kind of initial state that like we decided before we want to do these things. And then now something else happened. And so now we have to go on this thing. So so you have like a answer

when someone comes to you to ask like, why didn't you do this other thing before? Then you can say, well, we did decide to do that. But then something happened and we had to do this other thing. So the cycles is like, it's very similar to sprints. But we like to call it cycles, because we are not really sprinting anywhere. The cycles also run on like automated schedule. So it's like, you don't have to think about like, which day does it start? Or, or like, every, every time like set it up manually. So it's just like runs automatically. And so it's just meant to like, to help the team to focus on like, let's just like focus on this few things. And forget about the infinite list of other things that are in the background.

You mentioned earlier that you don't set metrics goals. And so let me take into that a little bit. Is that true? You don't really have number goals for features for launches and things like that. And so let me let me start there and I have a follow question.

Yeah, so we might have like a company level goal sometimes like, for example, like weekly active users, like that's like a metric we want to want to increase or something. But in terms of like specific features, we don't have goals for those. And the reason is that I think like product like us or like a system that is used by different kinds of companies and kind of it's like a system made of multiple different parts. And it's not like a very, it's not necessarily like you want to optimize any specific thing about it. It's like, also companies are a little bit different. So like their usage of different features can like differ because they just operate slightly differently, or their team size is different, or the setup of the team is different, or the culture is different. So there's like, I think, for example, I don't know, some like, Instagram or some of these apps is like, yeah, we need to drive engagement. And that's like the main feature, like that's the main metric for every feature. Like, we don't actually, we don't have that. Like we, we just think that like, there should be features that help companies. And sometimes we can look at the metrics before we start working on it, like, let's see what's the state of things are. But we don't necessarily want to set like, oh, we need to increase the specific metric by x. It's more like, we want to solve this problem. And ideally, the success way it looks like the problem like customers agree that the problem is solved, or

they enjoy the solution. And it's not like that the metrics went up.

So just to summarize so far, if no metrics, you have no experiments, you have essentially no PMs, just one product leader, you spend a lot of time on design and craft and making things awesome. I'm curious just what you think it takes to make a company work in that way, because this is pretty different from how a lot of other founders think and a lot of other product teams work.

Yeah, we like to like, talk about this internally, like this like a mixture of like, magic and science and like, like how we described is like, there's always some level of science that we do. And I think like, some companies are very scientific on their product management that like, they like to measure everything and they do do a lot of tests and things. But like, we just decided like, we don't think that's necessary, or that's that's a good for us. So the science for us means that like, we do talk to users a lot. And like the whole, whole like the any project we start with, we do some like level of user research. And as founders, like different people on the team, like, we might have like weekly calls with customers or users, we also encourage everyone in the team, like go to the customer Slack, they answer people questions, like we have shared Slack channels with customers, we go anyone like I sometimes go answer the questions there. I also see when they complain about something. I think so. I think the first part is like, the whole team kind of has to be really understanding the product and the customers and the problems people are facing. And I'm kind of have that like, empathy, and as well as like the understanding like, what is the state of things today? And then we talk about that. And then sometimes we might pull up stats and see like, oh, wonder like, is there some kind of patterns we see like, okay, these kind of companies are using this thing more? And what do we think about it? But usually we have some kind of question we want to answer. It's like, I wonder what what is going on? And then we look at it versus like, let's us pull some metrics and then decide that we should increase this metric. And then the magic part is like, what happens when you kind of build this understanding, like everyone in the company builds, it's not like everyone has the same understanding, but like everyone builds more of that customer and product understanding, then like, we have discussions like, what should we be doing? Or like, what decision we want to make

here, then everyone is much more like kind of informed of the actual reality of the, the customers or the, or the product.

And then we think like, you can much more like use your intuition or, or thinking to do those decisions that you don't have to use data or metrics to back those things up. So that's like, I think the main thing is like, the whole company kind of has to like, be with the customers or like, like talk to them, and then like understand like where the product might work well or where it might fall short.

That's what I imagined you were gonna say. And I love hearing that. For someone that wants to create a similar culture. Is there tactically anything you find just understand if your employees and engineers, designers have enough of that context and really understand the problem?

I mean, I think it's always like a different people, like different people in a company will have different understandings. It's not like you can expect like everyone, like will every day like go to see everything and like has has this. But like we, we do sometimes like sessions with that with the team, or we do record videos with the customers, we kind of write notes, and we share this with people. I feel like again, it's like fairly apparent, like if people, like, if you know your customers or the product, like it's very different way you can talk about it versus like, if you don't have any idea, like, I think like, if you don't have any idea, you probably don't even know what to say. So I think it's like, kind of apparently people have that. And it's not like, every project like we need like everyone to have this like, understanding, it probably usually enough is one or two people have that understanding or have different understanding of different things. So I think it's again, as I feel like it's like a culture thing. And I think the other thing is like, you kind of have to have the, you just have to kind of believe in it. Like, I think sometimes people use data a lot or too much because states are, they're worrying or they're afraid that will I make the wrong choice? And, and like, I'm using data to like, make the choice for me. But then like, you might still feel like this is not the right choice, but the data is selling me is the right choice. And then turns out maybe it was the right choice or not, but it's more like, again, like a practice thing, like you need to be, I think the company and you need to be okay that like, sometimes we make mistakes. And like, we've made the wrong choice. And then we just can fix

it. But at least we made that choice and the data didn't make that choice for us.

What's interesting about this is, if you've heard the episode on ramp and how ramp builds product with Jeff Charles, there's such a different ways of building product ramp is all about velocity, shipping, all the time, metrics, measuring everything. And your approach is almost the opposite. And I think what's interesting there is as a takeaway is just there's many ways to do it. You just have to do it almost fully. And you have to have really specific people, it feels like the people want to work in a certain way. And a lot of it, I think also is the founder has to be like natural to the way the founder operates and thinks of building and building a company. Yeah, and for sure. And then like, if you look at like, successful companies, and like Amazon is very different than Apple, and like how they operate, and I think both of them are successful, but not in the same way. So I think it's again, it's like, yeah, it's a decision you make as a company or as a founder, like what kind of company you want to build. I do think there is like some aspects of like the domain that you're in. Like what what does that domain and the problem space require from from the company. And for us, if they think it's like, we are in the, I think we are in the retention business, like it's like, and the trust business kind of that. Ideally, we have a company starting use linear very early on, and then they stay with us forever. And I think the only way we can do that is like, we need to continuously kind of deliver them like good quality product and like kind of maintain that trust that we are that that we don't like fail them or or somehow otherwise like mistreat them. And I think like some businesses are much more like transactional, where it's like, yeah, we just need to make this e-commerce sale. And then once it's done, like we don't care what happens. So in our case, it's more like we really need to like build this relationship over time. And then that's why like, I think some of the choices we make are also like, kind of like more about respecting the customer versus like, we're just wanting to drive the revenue of the company.

Awesome, such an important point.

This episode is brought to you by Pendo, the all in one platform for product like companies building breakthrough digital experiences. With all the tools you need all in one simple to use platform, Pendo makes it easy to answer critical

questions about how users are engaging with your product, and then turn those insights into action. With product analytics, low code in app guides, user feedback and session replay, customizable roadmaps and AI generated insights and campaigns. Pendo is the only solution you need to build, ship and optimize a successful product led motion. But don't take my word for it. Create your free Pendo account today and start building better experiences across every corner of your product. Yes, want to take your product led know how a step further, check out Pendo's lineup of certification courses led by top PLG experts and designed to help you grow and advance in your career. Learn more and experience the power of the Pendo platform today at [pendo.io slash Lenny](https://pendo.io/slash/Lenny). That's P E N D O dot IO slash Lenny. Something you're really good at personally is focus. I find that just trying to get you on this podcast was a lot of like, Hey, Gary, Hey, have you thought about this yet? And I know that a lot of VCs are just like reaching out to you all the time, all these really fan CVCs that are just like trying to talk to you and get close to you. And I just know you're really good at avoiding shiny objects and staying really focused and really heads down. And I've always wanted to just ask you, what do you how do you do that? Do you have any tricks systems processes approaches to staying focused other than just ignore ignore the inbox mostly?

Yeah, I don't think there's any like complicated like processes. And so I think like one of the things like I was in YC in 2012 and one of the things the main thing they say there is like what you should be focusing on when you build a startup is like talk to customers, build the product exercise. And like if you if you're not like if you find yourself doing something else than those three things, it's probably like the wrong thing to do.

And everybody said exercise or yeah. Yeah. And the exercise is that like, it's important for you to be healthy, or not to like burn yourself out. So I think there is like, it was like a balance, like advice to to that.

Love it.

So, so I'm doing those three things. And, but as so, so I think the thinking there is like, I think we often as a company also talk about this and, and like very early on, and I use this the same way. And like, I think the company can use it the same way. It's like, I think there's always things that

you're supposed to do. Or like, it sounds like a good idea to do. And like, it could be like, yeah, like come to this podcast. And I actually think like, before I wasn't like, or like, I was at this question is like, is this important to do now? Or is it important to maybe do later? So I think like, for example, the question on this podcast is like, I didn't feel like it was important to do it earlier, because we weren't at the stage or scale or something that it I think would be like as interesting or or or something. So I think like it was a better timing to do it later. Similarly, like when we built the product, well, like initially, we were just very like, focused on like, is this really important thing to do? There's always like, yeah, like, you could get like, sock to like security kind of certificate. But and we know that like, eventually, we need to get it, but we don't need needed today. So we just say no to that. And like, if customers ask for it, it's always like, we don't have it. And we will have it one day, but not now. And see like, and a lot of times people are like, Okay, like, that's fine. And then like, internal, we also talk about this, like, you know, like, RPG games, you have the main quest lines. And then you have the side quest lines. And we often talk about the company is like, avoid the side quests. Like, like, there's always like ideas people have. And it's a good thing. And it's like, people have ideas. But then it might be like, Yeah, let's make like this T shirt. So like, let's make this thing. And then we're like, Well, is it does it help the customers? Does it help the product? Like, this sounds like a side quest to me. And like, basically means like, we shouldn't do it. Like this doesn't progress the main quest line, which is like building this product and like making it awesome for for these customers. So similar to me, it's like, I operate this way personally too that I think about like, is this important for the main quest line in in building this company for me? Or, or is this something that I can ignore for now, or something I can do later? And it makes more sense then.

What is such incredibly good advice? Basically, ask yourself how important is this to do now? And is this the main quest or is this a side quest?

Amazing. Okay, so let's talk about hiring. As with most areas, you're very, very, very deliberate about hiring. The bars so insanely high at linear and you also hire very few people. So just a few questions along these lines. Just one is when you're hiring people, what do you look for that you think

maybe other people are not looking for enough? And where do you spend a lot of time?

I think one of the things we all of us founders kind of saw in this high growth companies that sometimes like the high growth is like, especially on the employee side, it's not that great. It can create a lot of kind of chaos or just messiness or we're just generally like in my his past on working in companies, it's almost never it was always easier to work with a smaller team, very high quality people than with a very large team of like more average people. It's like almost like it's always faster and better output when you have like more much more like smaller team. So that was kind of like the thing with linear too is like we just believe that you can actually build better with less people than you can with more people. So that's like this is the basic belief we have. So then when it goes to hiring, we've been like taking very like, kind of like slow steps on it that like in the almost like the first year, we didn't hire anyone. Then like the second year, we hire like couple people and then the second year, we hired more few more. We never top more than doubled in a year. And that's kind of like, you know, like guideline that like we shouldn't more than double. And this might be something we change in the future that we actually might do less than that. But like what we look into hiring is is like a couple things like one is also obviously depends on the role. But basically, I would say like every with every role, we often talk about like the there needs to be some taste or some like, this kind of understanding of the, like how things are done or how how what's the like, people have more like a broader perspective than like whatever their role is. So like, we talked about the engineering before that like, they do need to do some of this BM type of stuff. And and so what we look for in them, like, is that like if they have some of this like skill set or product thinking, or they can articulate why some some choices are better than some others, or like in their past, like, did they disagree with some of the company's choices or the team's choices? Or like, so we want to have this like, obviously, they need to be good developers, but also like, do they have this like a product sensibility or, or can do they have like a judgment around that? And this goes similar to like, like, for example, like a marketing hire is like, we, we think about like, yeah, we do need the marketing skill sets. But then we also want to see that this person also like, like, maybe it's a good

storyteller, or like, they have like this, like, kind of appreciation for, for writing or stories or like, they have a taste of what's, what's like, interesting and what's not. So I think like, with or like, when we hire like operations person, we also like to see that they, they maybe like have understanding on HR, and maybe it's not their role, but they understand it. And what happens is like, when you have this people that are a little bit more than their title, it's like the company is, I think, much more easier to manage, because it's like, people can like pick up things more easily, or they can like, work together more easily, because everyone has more like a shared like, more like a shared areas, or, or it's not like, it's not, you rarely get to the point, like people say, it's not my job, it's, it's more like, people understand, okay, yeah, I'm, I'm kind of in operations. But today, I kind of need to help on this HR thing. So which is okay. And so that's kind of like, what we look for people is like, they are more than there, they can take more scope than their skill set would assume or like, what normally is expected from them. So essentially, you're looking for kind of these Venn diagram overlappings across different functions and teammates. Yeah. And I think it's the other thing is, like I said before, it's like, we want to build much, I think, like a company that has less employees, which means that it's like, like I said before, we don't want that many like specialized roles or like two specific areas of ownership or something. We just think that we could build this like, we could have less people and those people can take on more scope and they can own more scope. I think like traditionally, I feel like in companies, like how do you get more scope is that you advance in the levels of the company, because there's like a lot of different teams and different levels. And then to get any kind of scope, you need to like rising into this like higher levels. And what we try to do is like, you don't actually, you don't have to have that many levels. But like people can just like, already when they, when they start, they can start like owning more areas. And I think that can be like much more like, also like interesting, not to everyone, but like, I think interesting to many people. And it's kind of like how I also like always felt about as being a designer is like, I don't, I didn't feel like my job is purely like just looking at the designs. I also thought like, I actually need to be helping this business or helping this other area as well. So I think it's

just kind of like also like natural to me.

Awesome. So one thing you didn't mention is you have a really unique way of interviewing, which is a paid work trial. Can you just talk about what that is? And also just while you're in that area, you talked about testing for product sensibility. So whatever you can share or how you actually do that would be awesome.

Yeah, so we do with all of the employees, we've done like a paid work trial and depends on the role, what it looks like, how long it is, and depends on also sometimes on the person. But basically, like we do fairly like standard like in the V loops where, where we test, like, we have some hiring manager interviews and then like skill interviews and or tests and then the last step of the process is the work trial. And basically, yeah, we, we basically come as like a mini contractor to the company. And we give them a very like, usually fairly vague problem statements, like, if you're engineers, like, Hey, there's this feature that needs to be built, like how would you build it and like, go build it. And so basically, they need to first like understand the problem, then they need to scope it down to something that they can do in the time frame that they have. And then they actually like go, they get the access to a code base, they can actually go and go and like build a version of it. And then like, at the end, they can present the work they did. And why we do this is that like, we just seen that like, it's a very good way to see like, for both us, like both for the company and the candidates to see like how we work together. And like, for I think, for the candidate, what they can see is that, like, what kind of company are joining like, what is it like to work here? And how are the, what is the my ownership or like, how do I approach this? Like, I think a lot of engineers also like that they see the code base and they're like, Oh, wow, this is like really clean. And like, it's not like some kind of spaghetti code type of thing situation. So I think like it helps the candidates as well understand like, what, what are they signing up for? Which I think can be like very risky sometimes with, especially with startups, like it's, it's really hard to tell like how the startup is operating just from the interviews. And in a large companies, I think things are more standardized. So it's like, I think they're more similar on it's easier to make that choice. But with startups, it can be like very like different how companies operate.

Yeah, that is so unique. And I rarely hear of a company being able to hire that way. I imagine one of the reasons you can get away with that where people are like, don't have a full time job for a while while we're doing a pay trial is because linear such a enticing place to work. Imagine for a lot of companies, they can't really do that. But I guess any thoughts on just maybe more companies can actually pull this off.

Yeah, I mean, I think it's always like, if you don't ask, like, you don't know, like, I think like, in our case, we that's just been the standard. And we try to work with the candidate, like, let's figure out maybe we do it on the weekend, or maybe we do it some other, like a vacation holiday or something. So there can be ways we can like schedule it so that it's it causes as little kind of problems to the candidates as possible. And I think we only have like, only a few people probably have ever declined it. Like it's it's not like, I think everyone else has been at least after the fact they've been happy that they did it, because they felt like they had a much better sense of the company they're joining. And then also, like doing that work trial, they can actually join our meetings, they they get access to our slack and notion. And they also have one on one chats with the rest of the, like some of the other people on the team. So they already get to know people. So it's a good way for them to like evaluate us as well. And then for us, it's obviously we can see like, what is important for us to see is like, how does this person operate in this kind of environment? And like, how do they approach problems? Like, how do they think? And like, are they able to make progress in a very short timeframe, which I always think it's like, very important for startups, like, in a large companies, you have maybe all time in the world to do stuff. But I think like, in any kind of startup, even even with us, when we like take our time doing things sometimes, it's still important, like, we can do things quickly if we have to. Super cool. Just to close the thread on product sensibilities, or anything you could share of just how you actually help understand someone's strengths and inability there?

Yeah, I wouldn't say like, we have like, some kind of very scientific or some, some like, special way, figure it out for this. So I think it's a lot of it's like, it's like a discussion of, of, and I often think, like, ask people that, like, like, asked about their projects. And I try to go deeper. It's like, why was this decision made? Like, why do you think the decision

was made? And like, I might ask, like, do you think it was the right decision? And like, or do you, did you agree on it? Or asked about, like, what, what do you think you would have done differently? Or something? So I think it's more like, I'm trying to see if they do, do they have thoughts in this area? And like, what their answers is? And people's answers can be very like, different levels, like some people might be, yeah, just like, I didn't like it, which I don't like, yeah, it's an opinion, but it's not based on anything. It's just like, you didn't like it, you should be able to expand on it saying like, why don't like it? Because in this case, like, it would not work well for this kind of users or in this kind of context or for this kind of purposes. So they, they have like more of this like reasoning, or some kind of rational, like, why they think this way. And they can articulate that. So I think that's like, kind of like what I'm, we often like testing for us, is like, can they, can they do this and how well they can do it? And it's, it's gonna be like very, yeah, they're gonna be like, very wide ranges of how people do it. And when when you see someone who really thinks about this stuff, it's very clear to see that they can just like talk about it forever. And they can go deeper and deeper. And then some people that maybe don't haven't had the experience or don't think this way, they're like, yeah, I know, I don't really know, like, I just, I just build it and then seem fine.

Let's transition to the third area I wanted to spend some time on, which is growth. And basically, I love to just understand how linear grows and what you figured out around growth, especially in B2B, SAS. So first question here is just, how long did it take from starting to work on linear to launching, say V1, something that a number of people can use. So we started, like, official in 2019, some, I think, months before that, we were already exploring and prototyping the product. So it wasn't. So I think we prototype different kinds of designs a little bit. And then we also, one of the things we really wanted to solve is like, we wanted to make the application really fast. And the way we figured out we do that is like, we have more of this, like a local base data structure, where all the data lives in the client and then and then it gets synced on the on the back ends, like with this delta packets. And, and back then, we were just exploring like different off the shelf solutions and systems, but there wasn't nothing really there. So we ended up like building our

own. And so we, we spent some time like prototyping that. And then once we we've officially, I think started working on the company in April 2019, and then we announced the company, like roughly meet April. And we have this like little website up with with the waitlist. And, and then I think by May, we could use it ourselves. And then we, we already we started inviting some friends like try it out. But then I think in June, I think we started more like inviting people from the waitlist and around June, July, I think we had about, I don't know, 100, 200 users on it, and then maybe like about 10 companies or something. And, and then we were in this private beta stage for almost a year. And the way we did it was just like, we had this waitlist of people on the waitlist. There was like few survey questions, like what kind of tools you use today? And, and then, like, why do you want to use linear? And then we just, and what's the company size? And we invited people based on like we invited more like smaller companies, using the tools we currently support it. And then also, like, I was trying to see like, who is more like interested versus us. And I don't know, I just want to try it out type of purple. And then a year later in June, we, we launched it publicly. And back then, maybe we already had like, I don't know, several hundred of companies using it. And then we also launched the pricing. And I think like, almost all of them, maybe one company didn't subscribe, but everyone else subscribed to that pay plan. Okay, there's a number of really interesting things here. So one is you're in private beta for a year. And then a year later, you launched. How long was that period between starting to like incubate and starting to build to that private beta milestone? Yeah, I think just like few, few months like just a few months of building the V1. Yeah. Wow. Okay. I thought it was a lot longer. That is so interesting. Okay. What a team you've got over there. Okay. And then this survey piece is really interesting. I've heard a little bit about the story. So essentially, you launched it on Twitter, you had kind of a following, your founders had a bit of a following. So I think that helped build up the initial waitlist. But what you did there wasn't just like, hey, go sign up for a waitlist, and you just add email addresses. It's a survey asking them what tools they use, like whether it's GitHub, or something else, and then also the size of the company and their interest. And that helped you basically prioritize who to go after and who to onboard. Is that right?

Yeah. And the reason we did it, because we know that like, we didn't support everything. And like, what was that before? And the focus is like, we want to like, also like, be focused on like, let's just like, build a version that can work for some people, or some companies, we don't have to try to address everyone in the world in the in the first months of the of the business. And even before after that. So it was very like selective process. And I think we yeah, we were fortunate that like, we were able to get people sign up on a waitlist. And I think after, after a month or so, we had like, maybe 4,000 people on the waitlist. And then we had this like, internal, I think initially it was just like a very manual process. But eventually we built this in like an invite tool that we could just send invites. But in the beginning, I would go read the actual surveys, like in a spreadsheet, then I copy the email, and then I email them, they invite link from my personal email. And then like, I would just like, email them like, after like, few days or a week, and it's like, Hey, what do you think? And the reason we and so we would invite only like, in the beginning, we may be invited to like 10 people a week. And like, eventually, we increase those amounts. But the reason we did it that way was that we thought that, like, if you just invite everyone at once, or a lot of people at once, the all of those people are gonna probably hit the same problems in this kind of software that is very early stage. Like, I don't know, they, they hit the same bug or the same problem in the software. So then they will all send us feedback, like, Hey, there's this problem. And then we felt like it was kind of like a waste of effort. So we would just do this cohorts, like, let's invite these people and then they say, like, like, Hey, this is a problem, like, I don't, this doesn't work or something, then we go fix that. Then after we fix that, we invite the next cohort of people, then they say, like, Well, there's this thing that is needed, or this doesn't work. And then we fix that. So for that year, we did this cohorts, and then always get the feedback from the cohort, saying, like, this is like, wrong, or this doesn't work. And then we will fix that. So, so eventually, I think it was much more like a, I think, like, effective way of doing the initial development than just like inviting or letting everyone to use the product right, right at the beginning. There's so many interesting lessons from this. I wanted to ask how you got your first 10 customers. And what I'm hearing

essentially was from this waitlist, you launched it on Twitter, people signed up, you pick people to let on board, you worked with them over the course of a year to make it what they needed, and then eventually started charging.

Yeah, I think like the first 10, like, people companies using it, I think maybe a little over a half, maybe there was like three friends that like friends that have startups and they used it. And then I think the majority of them were just from this waitlist, but they didn't like pay us anything like we didn't have pricing in the beginning, or during the private beta. At some point, we start building the payments function. So we just like added a added page in the settings that like you can optionally pay. And then we just give you a slider that like how much do you want to pay for seats? And then we just like see like if some people paid like \$28 per seat, and some people paid like \$1. So so we buy like it doesn't really matter, we just wanted to test the functionality and see like what people people think. And then like, yeah, like after a year, when we launched, we already had like, I don't know, first week of launching, like we have probably like some hundreds of customers.

I've never heard of that approach to pricing is just an actual sliding scale, where people can slide the scale themselves and how much they want to pay. Did that help you figure out what to charge? Or is it mostly just an experiment?

I don't think it's like gave us like enough data to like decide like I think it's but I think it was like good to see that it's like there was some people that went I think that 20 was probably the maximum that people could pay. So I think there were some people that went to it and they felt like, actually like, yeah, I really love the products. I'm happy to pay like \$20. So I think at least it gave us some like confidence that if we charge for this and and it's like something under \$20, it's it's there's going to be like market for it.

I want to hear about the story of how you've started to feel product market fit, whatever that means to you, when did you start to feel like, Oh, wow, this is actually gonna work. And maybe this is going to be a real business.

Yeah, I think like, we always been kind of, I don't know, some like paranoid or skip like, yeah, I guess maybe a paranoid is good way about the product market fit. I think like, it's like a paranoid in a way, like we're always wondering, like, do we really have it? Like, and like, with who do we have it? And I think it's, it's true in our kind of business is that

like, I think, I think we started feeling it very early on. And like, when people first like, started using it, and we could see like, Oh, now the whole company is using it. And they're, they seem like happier using it and the feedback is good. And they might have some additional asks for them for us. But, but we started feeling like there was definitely like product market fit with certain kind of customer. And these were like, more like, smaller, like early stage companies, maybe we're still the founder is still running the, the product then and they care about the speed of the shipping or they, they kind of have like a certain values in a way. So it was like a good fit with them. And then I think we always like, know that we, we want to like address the whole market and not just like these early stage customers. But we knew that like, I don't like if a fortune 500 company came to us then or even like today, we might not be like, I don't think we can like provide them the solution today that works for them. So I don't think the fit is there. So for us, like the way we think about is like, do we have that? Do we have the fit in this specific segments? And like, how strong that fit is? And so like in the, in the company is like journey, I think we the first year, we kind of just focused on like, can we get the fit in the first two years, we focused on like, can we get the fit in the, in the early stage, like startup kind of segment? And like, basically, the goal was like, we want to be the default for, for startups, like the default tool that the startups pick. And I think we were able to accomplish that, but we just purely focused on that segment and getting the product market fit there. And then like after, or at the same time, we started getting some larger companies. And we saw like, yeah, it's not like, really great for you right now. But like, let's work on it, making it better. And so I think the last two years, we've been like focusing on that. It's like, how do we make the software work better? How do we get the product market fit better, like stronger in this larger company segments, like, that are like thousands of people, or like hundreds of people, or like a thousand people.

I think this is such a good way and smart way of thinking about product market fit. A lot of people see product market fit as this like binary, I have it or I don't. And like, when am I going to really feel product market fit? And what you're describing is what I often hear is it's more of this spectrum of like, more and more confidence that there's product

market fit and even more specifically, it's like product market fit with segments of the market, it's kind of like this map of the world, and you're just like slowly acquiring territory in the market with specific elements and then over time, it grows and grows.

Yeah, I think like a spectrum is a good way to think about it too.

I think it's I feel like there's this blog post and like written in the past, where it's like, you know, when you have product market fit, and I think it's it's probably like, it's like that for some, I don't know, social consumer apps, like, you know, like if it's taking off or not, then you don't really have like a lot of different segments, or like, you don't really think about it, like you just have users and you have millions of users. So, and then you see like it's taking off. And so you have a product market fit. But then I think like in a more like a BGP world, I think there's always like, you can have different sizes of customers, you can have different domains the customers are in, or there's different kind of like categories where it's like, you might be doing a little well in one category, and then not that well another. And I think like maybe the countering to do things is that like, actually, if you're doing really well in some category, it's just like, kind of double down on that. This is something like I talked to, to the zoom founder Eric at some some point in the company's lifecycle. And this is also what he said, like, it's like, when they were like, building zoom in the early days, they would get this one type of customer, like, I don't know, maybe it's like a university, and then they like really, it worked really well for them. Then they're like, well, how do we get more of the universities? So they would always like focus on a certain kind of customer, rather than like, let's just try to get everyone like, so let's focus on everything, which is not possible. So again, like, it's about the like, the focus is like, if you have some like, you see that something is working really well, then it's almost like you should focus on doing that more until you hit some kind of points like, okay, now we do have that category, like, captured or or handled as much as we want. And we should like expand to to a new area. Essentially, look for pull and just follow that and pay attention to that.

Yeah, and I think there's like, for us, it's often like there can be sometimes like, for example, now we have most of the AI companies are like, are using us. So I think like it's always and

before I was like a crypto company. So I think there's like, when we see this kind of things happening, then we start to think like, Oh, could we do something differently? Or like should we? Could we like get more of these AI companies on board? Such a great lesson. Just a few more questions. You mentioned that you launched on Twitter, and that led to a large wait list and a growing wait list. Is there anything you did before that to kind of build this following? You know, that sounds like really, like really amazing, cool, we just announced it on Twitter, and we have this large wait list, and then we grow and we get all these customers. Is there anything you did ahead of time, in anticipation of this launch? Would you recommend people work on building some kind of following online before they were going to start it? Was it just like, Hey, we happened to have this kind of following and it worked out anything along those lines, you would recommend to founders these days?

Yeah, I mean, I think definitely if you have a following and, and like, opus depends what kind of following, but I think like my, my background as a designer, I was like, at the Airbnb and Coinbase and other places, and I did some like talks and conferences and, and write some plug posts. So I was definitely like out there and then kind of had that some of that following, which which was helpful. But it wasn't like I have like, 1000s of like 100s of 1000s of followers or millions or something, I had maybe like, 10,000 or something, which is like a significant number. But then I think what the other thing is, is, I think with the announcement, like, one of the things we, we did, I think, well, it's like, I think sometimes startups do, try to like emulate successful large companies too much. And you kind of like do this, like, fancy announcements, where it's like, hey, now we're like, doing this fancy thing. And then it's like, sounds very like corporate or something. And I think with the, with our announcement, we try to like, wrote it more like, direct or authentic to us, like this is like, what are we going to do? And this is why I'm like this, these are some of the things we're going to do. Then on on the Twitter, we did the same thing, like all of us founders, we wrote our own reasons why we're doing this. And I think it was like, just much more like, and I think like, people could dress and people like us could resonate more with it. So we were kind of writing to the right audience. And like, I think that's the probably the first thing you're like, when

you're announcing your company is like, you think about like, who is my first audience? Like, who would be the best users, like early users for this product? And, like, where are they? And then like, how do they think about things? And like, what, what kind of language they use? So for us, it was like, fairly, it came very naturally, because we are these people, we're in building software and these companies. And we've seen, like, other people have seen similar things we have seen. So I think it did that the way we announced it like resonated with a lot of people. And then I think we, we did have some friends. And as it like, we got some data like angel round, where we got some like friends involved. And the main reason we did it was that which is felt like it's, it's in the early days, it's good to have like, you feel like a real company, in a way, that's like you have someone to answer for in a way, even though the investors don't really run your company, or they don't have that much power, it's more like, oh, I took someone's money, so I now need to like make it worth it kind of. But then like, I think with the announcement, again, like, we could use some of those people to like, spread the message as well.

To kind of close out our conversation, just a couple more broad questions. You have a pretty unique culture at linear. And I know you one fun thing that you do is you have this kind of baking competition. Can you, can you talk about that? And what do you do there?

Yeah, so so so since we are like, fully remote and distributed companies, so we have people in Europe and US, like a lot of like, kind of like, group gatherings are kind of challenging, like, remote group sessions are kind of challenging, because the time zones are so different. So like some of the basic things like happy hours, it's not really like, really, like, doesn't really work that well. And also like, assume happy hours is really not that fun anyway. So so what we, I think a lot of people in the company watch their great British baking show. And so we decide like, maybe we do something like that, like where, basically, we would just pick a recipe. Firstly, it was like baking. Now we expanded to cooking recipes to. And so we just pick a recipe that is like somewhat reasonable to do in internet, like in a few hours in a couple hours. And it doesn't like require like tons of equipment or skill or something. And and then we, we just tell people I go by the ingredients use the company card, everyone has a company card. And then, then like hop on zoom on this day.

And for me, it's usually like, since I'm in California, it's like 8am in the morning. So so we start like that baking or cooking then. So so we've made things like, like a roll cake and lemon meringue pie. And we made some, like pastel net data, which is like Portuguese, like pastry. And, and then like, we just hop on the zoom, everyone's like doing their thing, following the recipe. And then like, sometimes people have like questions like, Hey, I'm, I don't know, stuck with this or like, my dough looks weird, like, does your dough look like this? And people can kind of like help each other. And then also like chit chat about whatever random things at the same time. And then like, we, we do the thing. And then we, we everyone takes pictures and posters on this like slack channels, like what they, what they achieve, then I think we have like, kind of friendly competitions, like who did it better and like, or who did it best. So people sometimes put a lot of effort into the, into the decorations and, and visuals. So in a way, it's again, like, a craft thing that we do. It's it's like, I think baking and cooking and these kind of things is also a craft. So we liked it that way. And yeah, we've been basically doing it quarterly, since the beginning of the company. And yeah, the latest thing, we were a little bit like, I think, didn't have that much time. So we decided to do like easier thing, which is like a summer drink recipe. So I think it's it was like people made like matcha drinks and and some like coconut drinks or like iced tea or something. So even that was kind of like interesting to do.

Have you ever won one of these competitions yourself?

We don't know if we like declare winners that much. But I do think like, I do, since I'm a designer, I do have some advantages on the, on the visual presentation. So I think that I generally do well on that. And obviously, that's like, with this remote competition, it's that's the only thing you can really like, look for. It's not not necessarily about the taste or the texture, because you can't really taste it through the zoom.

Maybe it's the last question. Just again, broadly, you've gone from being an IC designer, manager of designers to the CEO of a very fast growing company. What's something that you've learned about leadership over the journey of linear that maybe you didn't expect?

For some reason, it was surprising to me, I think that like, being a CEO or, or some of these like leadership roles is

that you end up doing so many different things. And I think like, even when I was a designer, like, even if I would be like, like some like high level designer in some company, it's still like, you're just mostly focusing on the design. And that's like your job. But then like, when you're a CEO, then it's like every week or every day, there's some some different thing going on. And it's not like, there's sometimes they can be like problems. But a lot of times it's like, hey, we need to like, figure out how are we going to do this? Like, how are we going to do this? Like compensation? Or how are we going to do this marketing plan? Or like, how are we going to do this like offsite thing? And so it's just like, to me, what is like challenge, like, definitely like, challenging for me is, is handling that like, different kinds of things that that come to you. And like staying somewhat focused still on something. So I think it's haven't necessarily fully figured it out. But I also like, learn, figure it out that like, yeah, hire, like hiring and delegation helps with this that like, if you can find other leaders that can like take on certain areas that that's that's helpful. That that's the like the main thing that that like, how what is the it's like a very wide range of things that you may be didn't have experience before. But also, I think it's interesting to me to learn about these things. And like, you learn about financials, and you learn about legal things. And then you start to feel like, Oh, actually, I know something about these things are like over time.

For the actual final question, before we get to a very exciting light and round, what's just the future linear? What's coming? What's happening in the future? Anything can share?

Yeah, I think there's always things we're working on and improving. One, like a newer thing we're working on is this feature called asks. And, and basically, what it is is that we we see that like, in a company, there can be a lot of a lot of different people that needs to interact with the product team or different people that needs to interact with this team, but they're not necessarily like in linear or part of this team. So we would be building this like an ask feature, which which is like integration to Slack, where you can very easily like, go to a Slack channel, then then ask like, so your question, like, you need something from this team, like maybe it's it team that like, you need like a laptop, or maybe it's like the, like infrastructure team, and you need like, help them like you need something from them. Then the team that this is is handling the

request, they can very easily like send it to to linear into this like triage that we have. And then like, they can start like, doing stuff with it. And if they have like questions or discuss like additional questions to the actual person who requested it, we can like send those messages back to the person through Slack so they don't actually have to go to to linear or they don't have to be a linear user to use it. So we think like, this is like just like a good way for the company or the whole company to be more potentially involved in the in the company, like the product operations, without like having to be like, like a power user of linear or or because not every function really uses it or or needs to use it.

Awesome. What a cool peak at something coming out soon. Or maybe all by the time this comes out. And with that, we've reached our very exciting lightning round. I've got a bunch of questions for you. Are you ready?

Yeah, I'm ready.

All right. Well, what are two or three books that you've recommended most to other people?

Timeless way of building by Christopher Alexander, like he he's just like, he wasn't really an architect, but he I think thought in Berkeley. And I think he he has this like interesting thoughts about like, building things. And like, he focuses on buildings and towns and these kind of spaces. But I think there's a lot of like things that are also interesting for building software. The other the other book that I like is like the Zen and the art of motorcycle maintenance, because it's also like talks about the quality of things. And and I think that's one of the main themes of the book. And like, that the thing is also that quality is so hard to define. It is like, if you actually like start thinking about it's like, how do you define it? It's it's like, it's kind of like, it's really hard to pin pin down. But it's all kind of like, when you try something or see it, then you kind of know if it's quality or not.

What are some recent movies or TV shows they've really enjoyed?

I think that the movie is like, probably the John Wick for I think it's, it's, it's like, kind of feels like, I mean, help us say it's like, there's no a story in that movie. But it's like, I think it's very true to its nature. So I like that fact. And then also, recently, I started watching the Silo on Apple TV. And I think I kind of like it. It's like a good mystery. And then also, it's kind of reminds me of the fallout

game. So, so I kind of like it that way too.

I actually read the silo books. And I was really excited for the show to come out. But we mentioned this on a previous podcast, the show is like so little to do with actual books, like the core ideas are the same, but there's always stories that they're just making up on the show. So I kind of stopped watching because I was just like, that's not what I was hoping for. But maybe I need to check the books later, like one side, which the show definitely read the books. But there's three of them. And only the first one is actually good, the other one, they're not actually, we could and I should not have read them, because it just just went off the rails a little bit. Anyway, next question, what is a favorite interview question that you like to ask candidates when you're interviewing them?

I think usually I like to ask, like, what is what is the candidate most proud of and why, like what, like on their professional life or otherwise, like what they're most proud of and why. And then I think we can go deeper on that. But I think it's kind of like, gives you a little bit indication, like what the person values and, and like how they think about things. And, and I also like, I think it's always nice that people can share something like they think they did really well, and we can spend time on it versus just like, asking something like more like negative things.

What are some favorite products you've usually discovered that you really, really like?

Not sure if I discovered them recently, but like recently I have been in this home office, I've been installing some of this hue lights. And I really like them because like, throughout the day, I can like have more like kind of harsh lighting, because it's I'm in meetings or something. And then like in the evening, I can kind of like change the temperature, like I make it much more like red or orange or something. So it's just like, I think it's nice to like, you can kind of transition to space is like, okay, now I'm working and now I'm like, doing something else and you can use the lights to kind of like indicate that.

That is so cool. Do you like automate the schedule or you manually change the color?

Yeah, I just manually change it. So I have like, like on my home app, I have like scenes that like, so there's the night scene, and then there's the day scene, like the morning scene. And so I

just like, click that button and then then it turns changes the lights.

That is extremely cool. I'm gonna try that myself. What is a favorite life motto that you like to repeat yourself or share with people, something you kind of come back to a lot?

Go slow to go fast is, I think for me, it's, it's, it's about that. Sometimes people have tendency to rushing to things and especially in, I think in startups, but other places too, that you kind of have this like, I think urgency is important, but, but then sometimes you, you have like too much urgency and you are rushing things. And what happens is that you, you rushed it and then now you need to come back to fix it. So I think sometimes we, I like to think that like, you should take some time to actually like think about it and like, what are you gonna do and then do it? Because then it's in the end, it's gonna be faster that way than like going back and forth and fixing things.

What is the most valuable lesson that your mom or your dad taught you?

I think it's like respecting people and things. So I think it's, I mean, I think the people respect us is pretty obvious, but I think the, I think with the, the things you have also, I think like you should take good care of them. Like when you, when you use them, you just like, I don't know, clean them or put them away and then they're like ready for the next time. So I think I like that though, like you, rather than like you treating things like that they're a trash or, or kind of like not that valuable, you should kind of like treat things that they are, they are valuable.

Final question, you were born in Finland, I think you grew up in Finland. What is a finished food that people should definitely try to get as soon as they can?

One is like this salmon soup. And it might sound weird, like a fish soup, like maybe it's not gonna be that interesting. But it is like, it's like a creamy soup with some like potatoes, carrots, and, and other things. And it's kind of like almost like a sweet, little bit like sweet flavor to it. So that's one thing like you can, you can make it yourself at home or, or you can like, if you go to Finland, there's probably like a few restaurants that offer it.

Okay, amazing. Is that something we could get here or you have to go to Finland to get it?

I don't think I've never seen it here in in US in any restaurant,

but it's not very hard to make it yourself. If you can probably Google recipe, it's basically you just need some salmon and some some like basic spices and some cream and and some fruit vegetables.

All right, next episode, we're gonna do a cooking show with Kari. Kari, thank you so much for being here. You're building a very special company in a really unique way. And I think many founders and many product builders can learn a ton from watching you operate in the business that you're building. So again, thank you so much for being here. Two final questions where can folks find you online if they want to reach out and maybe ask you some more questions? And how can listeners be useful to you?

Yeah, so I'm on Twitter, my, my name, Kari Sarn, and, and, and then we also have the linear account, which is I think is interesting. So that's just at linear. And then, yeah, I think like, I hope everyone can like, check out, check out linear and like see if it could work for them in their company and like figure out if there's a pilot, like, I think we always happy to assist on those things that like, if you just want to try it out, then I try it with the team, we can help you to set it up and, and like help you to like understand how to use the product.

Awesome. And it's just linear that app, right? Is that the URL?

Yes.

Awesome. Okay. Easy peasy. Amazing. Kari, again, thank you so much for being here. Bye everyone.

Thank you so much for listening. If you found this valuable, you can subscribe to the show on Apple podcasts, Spotify or your favorite podcast app. Also, please consider giving us a rating or leaving a review, as that really helps other listeners find the podcast. You can find all past episodes or learn more about the show at Lenny's podcast.com. See you in the next episode.