eCorner

Stanford eCorner

Building Startups, Fast and Slow [Entire Talk]

Howie Liu, *Airtable*

05-05-2021

URL: https://ecorner.stanford.edu/videos/building-startups-fast-and-slow-entire-talk/

Howie Liu is the co-founder and CEO of Airtable. Inspired by his own experiences learning to code and building customized business apps, he co-founded the company in 2013 to democratize software creation. Prior to that, he was the founder of Etacts, an intelligent CRM tool that was acquired by Salesforce. While Etacts was a furious one-year sprint to acquisition, Liu followed a very deliberate, long-term approach with his second startup. In this conversation with Stanford lecturer Ravi Belani, he offers advice to innovators who seek to create complex products that can't be prototyped in a week-long hacking session.

Transcript

Announcer Who you are defines how you build. 00:00:07,160 - Today, we are delighted 00:00:10,090 to welcome Howie Liu to ETL. Howie is the CEO of Airtable. Inspired by his own experiences learning to code and inventing his first job by building customized business apps, he co-founded Airtable to democratize software creation. Airtable's mission is to enable anyone to build the software tools that they need to meet their needs. And I should say that Howie is another powerful example of how first-generation Americans are at the forefront of entrepreneurship. He was born in Texas to Korean parents who came to the United States from China. And Howie went on to study at Duke, where he was a double threat. He double majored in mechanical engineering and public policy, for all of you that like to straddle fences, both on the social sciences and the engineering sciences. And Howie jumped into entrepreneurship fairly early after Duke.

He spent a year on his first venture, which was a next-gen CRM company called Etacts. Then he spent a year stint in product at Salesforce. And then he started Airtable in January, 2013 with fellow Blue Devil classmates, Andrew Ofstad and Emmett Nicholas. Airtable has raised over $600 million in funding and has several hundred employees. So with that, please welcome Howie Liu to ETL. Howie, lots of virtual ETL and Stanford love for you. - Thanks so much for having me. 00:01:31,010 - Howie I wanted to start off, 00:01:33,730 because it is rare for us to have a founder that jumped into entrepreneurship right after school. We oftentimes see these stories of people who finished their degrees, and then they pay their dues, so to speak, and they become entrepreneurs in their late 20s or their early 30s. Or we see people who drop out of college early and just jump into entrepreneurship before even finishing their degrees.

And I know a lot of students wrestle with this question of, how do I know when is the right time that I should be a founder or to start a company? What advice do you have for students that are wrestling to understand when they should start a company? - Yeah, I don't think there's a one-size-fits-all answer, 00:02:08,950 but in my experience, I actually really wanted to find an idea that I could drop out from school to work on. And so throughout high school, end of high school, and even throughout college, I really pivoted or iterated through a lot of different ideas. I prototyped some of them. I was learning how to build web apps. And at the end of the day I never found one or I never ended up building an idea out to a point where I felt very high conviction, that this was not only something I could build a product around, but also could build a business around. So I was always sort of interested around not just building, if you want, product, but thinking through a few more chess moves, head on the board and saying, how can you get scale with this? What would the revenue model for this business or product look like long-term? Are there any reasons why this product might get early traction, but then get shut down or asymptote early on? So I went through lots of ideas ranging from some stuff in real estate and finding apartments, to apps that were more consumer and internet social. And long story short, it was very important to me that if I was going to drop out, which felt like a pretty big decision and commitment, I wanted it to be for something that I had really really high conviction could be a viable long-term business. And frankly, never was able to find that thing during college. So somewhat disappointingly, graduated, got a proper degree, and then headed out after college and was about to go in and actually take a job doing basically consulting for tech companies and getting to prototype apps as part of that job. But then at the very last minute had this impulse to just instead go and go full force at that point in starting my first company, which you mentioned.

So I don't think there is a right or wrong answer, but for me, it came down to finding an idea and validating it enough to where I felt like the risk-reward ratio was worthwhile. - What I find interesting though, 00:04:20,400 is that the idea of validation that you're speaking of is different than what a lot of founders speak about. 'Cause a lot of founders speak about product-market fit, finding something that's a painkiller, not a vitamin, and something where people tell other people about your product. But it sounds like the gating factor for you was not that short-term product development or product-market fit, but it was the standard of something that was gonna become a long-term success. If that's true, how do you know? What's the equivalent to know when you should jump in, if there's a long-term fit? - Yeah, no, that's a great question. 00:04:56,577 I think there are two ways you can build a company. And one is to just try to get that initial product-market fit and then ask the longer term questions later. And obviously, sometimes that works. I think Twitter is probably an example of a company that maybe didn't have a ton of long-term thought fleshed out around, well, how are we gonna monetize this thing, and what does this look like in five years, and so on. And I think it just emerged from effectively a weekend hack project out of Odeo, from which it's found, and it just sort of took off.

And so I think you can sort of build first, get traction, and then ask questions later. I think there's also a different way, which is you try to go and de-risk the long-term outcome for the business by saying, what is the business plan, basically? And you don't literally have to write a business plan. I don't think I ever wrote a formal business plan. But it asks us lot of the same questions that one might while writing one. Like, how do we beat competition? Like, if this is such a great idea, why aren't there other people doing? And even if there aren't any right now, if we go in and build this product and get some traction, what's to stop a competitor from coming in and doing the same thing better? Or beating us to win the market? So I think it is important to, if you're gonna pick the latter path and de-risk more of the opportunity upfront, to ask the right sorts of questions and probably there are some great resources out there around how do you write a business plan and what are the questions you wanna ask? I'm personally a big fan of the Peter Thiel class lecture essays that talk about these various questions or ways of thinking about sort of ideas. - The "Zero to One." 00:06:40,880 So I just wanna really have people grok this, because I think everybody right now, the mantra is always around having some proxy for how much people love your product, either by a net promoter score, or Sean Ellis has this growth hacking methodology. But this methodology is a long-term lens, and it sounds like the checklist that you're talking about is the questions that Peter Thiel put forth on thinking about network effects sort of in the long-term. And it sounds like maybe it's a combination of doing something differently in the short-term that leads to lock in the long-term. Are there any other specifics that, and I say this because I know our students always have this question about, they're gonna have jobs to join. And they're gonna think about, what is compelling enough for me to say no to a six-figure or a seven-figure, or not seven-figure, but six-figure salaries that are being thrown at young students these days to jump in and start a company? And I think the wisdom that you're giving is fresh and different.

And so I just wanna make sure that that gets heard. - Yeah, for sure, I think in a way it's sort of, 00:07:47,820 you're shifting risk between TAM risk. If you build this product and even if you get traction, will you have a large enough TAM and will you be able to capture enough of that TAM to make it all worthwhile and build a large and enduring business versus initial product execution risk? Like, can you even get the traction in the first place? And I think both are valid forms of risk that you should think about, but in a way, the worst case scenario is if you haven't thought about either. And so neither do you have a product that's taking off in the short term, nor have you gone in and made sure that once you get traction, that there is a light at the end of the tunnel that that will be fruitful. So I think you kind of have to at least bet one of the two, and if you're going the build product-market fit first approach, then I think it's really just about prototyping things, building them, and then being able to get early customer feedback as soon as possible. And I think this is something that probably the (indistinct) methodology really ascribes to. You just go out and try to get customer validation, show rather than tell that this is gonna work, versus the, hey, let's plot out what this business looks like long-term, what's the five-year roadmap. So if we build this first, how are we gonna sequence that to get to a winning outcome in the long-term? - And I will say, people have revisionist history here. 00:09:15,940 They will say, when we started, we had this big vision. But usually, when we dive into the details, the reality is different.

It's a lot of fumbling towards victory and co-creating things through luck. But are you saying that you actually did, when you started, you knew where you were gonna go? You have this vision and you were playing four-dimensional chess in terms of understanding the different stages? - I don't know if we were playing four-dimensional chess, 00:09:36,919 but we definitely had a clear idea of where we want it to go. In 2012, we made an internal vision deck that also had a rough business plan around it. And it basically described more or less like our five-plus-year roadmap. And I would say in rough terms, we have been executing against that. Like we want it to start with effectively a relational database that anyone can use, sort of like a Microsoft Access or a FileMaker Pro reincarnated. But then we would eventually build the ability, layers of greater customizability around logic. So now we have something called automations in Airtable, which only just came out about a year ago, where you can say, if this, then that type logic inside Airtable. At some point, we're gonna allow for interface customization on Airtable. And so these were all parts of the original vision that we had set out to execute against.

And I think we also had some idea of how we thought the market would evolve around us. So we described our position against project management tools and we've watched over the past five years, or really since 2013, so almost eight years now, how the project management tools like Asana or Smartsheet, et cetera, have evolved in the landscape. So I think we did sort of do a lot of that legwork upfront. We ended up spending almost three years building the product before even launching it. And so just the amount of time and investment that we put in before getting real, at scale customer feedback loops. I think was the function or was feasible because we had done more of this upfront TAM and market validation. - So we need to dig into this, 00:11:24,670 because I think this is a singularity in terms of what we've seen. And I think it's very, very refreshing too. And I think, it's almost like old school entrepreneurship of actually having a vision, a long-term vision, and then executing against that. And I should say, everybody these days calls themselves a platform.

It's almost an overused term. But you guys truly have built a platform. You're building software that builds software. It's very, very meta. But it's true, and you've legitimately done it. And so I want the prospective founders to really grok this or understand this, because the conventional wisdom these days is launch yesterday or launch today. And then the weapon that you have as a founder is to iterate quickly. And you guys took three, how long was it before you launched? - Almost three years before we launched publicly. 00:12:09,540 Now, we were getting some form of external validation throughout, so early on, it was prototypes or mock-ups and we would get feedback on it, and then we'd built semi-functional betas or alphas, et cetera, but we didn't (indistinct) in earnest launched the product for almost three years. - And so then what is the minimally viable product then 00:12:33,690 if you take this tact? Or is that even the wrong conception? Is that the wrong way to think about it? - Yeah, I think- 00:12:41,990 - And what is the right way to think about it? 00:12:43,870 - I'll contrast it also with my prior experience.

00:12:46,580 My first company Etacts was around for literally a year from start to finish, from when we founded it and started working on it to when we sold it to Salesforce as an acquisition. And I think we probably only spent 10 to 12 weeks building the product full-time before we launched it publicly and got a TechCrunch article and did the whole thing in NYC. And so just very, very different approaches Between the first company and the latter. So I think I've at least with one data point each experience both approaches. And I will say, first of all, emotionally, they're very different journeys. In the first case, it just felt like this constant furious sprint to try to get to the next milestone. And we didn't really know what the future would hold. There was never really any clear or a certainty around what the next five years look like. But the only thing that mattered was getting some form of progress. It was almost like we were these simple vector machines that were just sort of pivoting around and trying to find what is the right direction to go to descendt the gradient.

And then with Airtable, we had just a lot more patience, maybe even arguably to a fault where, obviously, time is precious. We all would rather get to where we wanna be in terms of impact, success, product growth sooner rather than later, all else equal. And so I think we almost had this level of complacency or serenity around knowing that, okay, here's where we're gonna get to. We kind of know what the mid game, end game looks like. And we just have to methodically build the product that we think will unlock that, or will be a great starting point. So to answer the question more specifically for Airtable, in that case, we were fighting against a few things that made that MVP a much higher threshold to pass. What is that? We were trying to build a product that individuals and hopefully businesses would actually entrust with valuable data and important workflows. And so this is different from being for instance, a consumer social product where the stakes are fairly low. Somebody can just go and try out a new product like that with little ramifications if it doesn't work out. Whereas with Airtable, we're trying to get companies, people and companies to actually put important business process into Airtable.

And if we went down, we went bankrupt, if we lost their data, that would have serious consequences for them. So that was already one reason why the stakes were higher. I think the second was that we were also trying to replace existing behavior. So we weren't just trying to add a new thing into people's lives. As sometimes, again, like consumer internet products, they're trying to add a net new thing into the suite of products that you use. TikTok isn't necessarily directly replacing your usage of Instagram, YouTube, et cetera, it's kind of adding a new thing. And so you don't really have to worry about replacing one for one, all the functionality of the existing product. Whereas for Airtable, we were trying to get people to displace other products with Airtable, whether they be project management tools that were more task-centric, Trello, et cetera, or even Spreadsheets. And Spreadsheets in particular were a challenging one to replace because, think about Excel or Google Sheets. They're pretty sophisticated products with a ton of complexity.

The number of lines of code and countless, probably millions of person hours of engineering effort, if not more that have gone into Microsoft Excel over the past three decades to form the product that it is today. It's gargantuan and there's no way that we could get to perfect feature parity against that product. And yet, we wanted people to replace their existing usage of Excel, especially for non-numeric workflow use cases where they were really using it as almost like an app or an app builder as opposed to a number crunching tool. We were trying to get them to replace that existing behavior with Airtable. And so I think there was a higher threshold. Like we couldn't just have a weekend hack project version of Airtable that people would use instead of a traditional Spreadsheet. So I think we just, the nature of our product, and the bet that we were making around why people would use us over the prior art necessitated a much higher MVP threshold across. - And the reason why the shift occurred 00:17:24,433 is because previously, you would have people that would hunker down for years, work really, really hard, pop their heads up, and then realize that this beautiful product had no match. There was no product-market fit. How did you still, were able to, was the difference here because you knew what the target was that you were dealing with these antiquated, old, existing incumbents, and so the specs were really clear? Or how did you avoid that pitfall that has pushed the whole community in this other direction now? - Yeah, for sure.

00:17:56,200 And we certainly didn't want to just be so arrogant as to believe that we could stick our heads, huddle down, and like Steve Jobs just come up with a magical new product that for sure everybody in the world would want. So I would say that we were pretty iterative in terms of de-risking the product market fit, even if it didn't entail actually launching the product. So early on, in the earliest days, we were going out and just talking to lots of other people who had worked on similar products. Whether it's Steven Sinofsky from Microsoft, who had overseen Office and Windows, and getting his feedback on our idea. Or other people who had worked on similar products. There were a number of low-code companies that existed 10 years ago and never quite made it. And so we kind of wanted to go and talk to these founders and other operators and figure out, why isn't there already a product like this that's mainstream, that's successful? Why did your company not ever quite hit that ubiquity. And so we did a lot of that early research, just trying to figure out what was the prior art, who else has thought about this idea in the space. Subsequently, we kind of develop our own thesis or it kind of corroborated our thesis that, none of these products were good enough. Like I think there was a certain threshold we believed where if the product was hard to use, if it was too complex, if it just wasn't a viable replacement for Excel or Google Sheets, then people wouldn't use it.

So there was almost like, we believe that there was a binary of either your product is better enough and intuitive enough to replace existing player, or it wasn't. And then if it's not, then you just die this very slow death, where you have a few people who are enthusiasts of your product, but it never quite catches on in a mainstream way. So that's what the conclusion that we came to. We had some validating conversations with people who did work on products like this, and actually felt pretty strongly that this could be a thing, that we were onto something. I think Steven actually had pretty positive feedback for us. And we also talked to somebody who had led Microsoft Access within Microsoft and shared their perspective on this product. So that was kind of the first step. I think thereafter, we just realized, okay, it's gonna really come down to our ability to distill a lot of the complexity behind a relational database low-code app platform into something that feels effortless, intuitive, even delightful to use. And has to be as easy or arguably even easier to use than the status quo. So the friction to adopt Airtable had to be minimal relative to using a Spreadsheet.

So we started working a lot of client-side or front-end only prototypes. We were just prototyping the design of the product and some of the interactions and the UX. And we would build these initially just mocks and then later, functional prototypes, but fake the back-end. So it wouldn't actually persist data, but you could kind of click around. And then we started testing those with different types of knowledge workers in different use cases, industries, saw how they use their existing tools, and then ask them to kind of replicate a workflow or see if they could understand how our product work. And there were just these subsequent milestones like that where we would go and validate the next big risk. Like, can we design a product that actually is intuitive enough? Okay, if we can do that, then can we actually get people, even if the product, even if the UX is theoretically intuitive enough, can we actually get people to shift their data over to our product? So then build a backend that's functional enough to actually persist data and get people even on small scale to see if they can actually start using Airtable. And if not, then why? So I would say, every three to five months, we were probably de-risking a different question about the viability of this product-market fit leading up to actually launching it, and then just crossing our fingers and seeing if people, now that anybody could sign up for the product and it was out there, whether people would actually come in and sign up for and hear about and talk about our product. - So there was still these user interaction 00:22:17,500 feedback loops that you're doing constantly. It's just that it took a lot longer to get to the MVP or the actual- - Yeah, exactly.

00:22:26,840 And the form of customer feedback was not necessarily just putting out a production-ready product that anybody could self-serve sign up for, but really kind of testing specific hypotheses or bending specific questions in sequence with customers or users. - And it was built sequentially, 00:22:45,020 or was it also built in parallel where you're doing the front end and then the back end with a different team, and then combining the two at a point in the future? - It was, early on, so it was just Andrew and myself, 00:22:58,340 and we were just working on the front end. So I think there was a long stretch where we didn't even touch back end code. I think at one point, we even tried faking a back end by using local storage or cookies to persist the data. So it was completely hacked. Like you would open up the browser and it would literally, if you open up a different browser or a different computer, your data wouldn't be there, and there was not even an authentication system. So it really was just kind of a hack to test the front end UX. And then at some point, we built a back end, but it was very deliberately just a minimal back end. We didn't over-invest into the infrastructure, even though we knew we would need to go and at some point, for instance, once we validated that, we thought this product would work from a UX standpoint, then we invested into building, for instance, the real time architecture that would allow multiple people to collaborate on Airtable for it to handle merge conflicts, et cetera. But we didn't wanna work on that up front, because we knew it'd be hard.

And it also wasn't the crux of whether people would ultimately adopt Airtable. We knew that real-time collaboration was a solvable problem, a hard problem. And yet, the difference of building that up front versus punting it till as late as possible before we launched, that wasn't gonna give us useful signal, having real-time or not in this prototype. - And your investors were on board with, 00:24:27,750 did they sign up knowing it was gonna be three years before there was gonna be a product release? Or was that- - I don't think we ever 00:24:36,130 committed to a specific timeline. And truth be told, I don't think any of our investors expected it. I don't think we expected it to take three years from start to launch. Now, to be fair, I think the first year, over a year of that three years, we hadn't raised money. So it was just two people bootstrapping, burning our own savings on a ramen diet and working ourselves on the product. - And can you talk about that? 00:25:01,960 Can you talk about the role of co-founders when you're building these Howie Liu endeavors, these big ambitious endeavors? Is there a difference in terms of the role that co-founders play? Did they, and was everybody on board with, was it one of these things where people were thinking it's gonna be a year and it ended up being three? Or was everybody on board from the beginning? Saying we're gonna go invest three years of our life before- - Yeah, so to answer to that last question first, 00:25:25,070 I think it was somewhere in between. I think we were very committed to building a long-term company.

We had gotten fully on board with this vision of, hey, if we pull this off, this isn't just a little marginal product in the world. We might actually build a really meaningful platform that enables people around the world to create useful software applications in a way to open up the power of software as a medium for building useful value to an order of magnitude more people than it's currently exposed to. And so I think we were excited enough about the long-term possibility and then financial upside, believing that the ultimate success of this company could imply building something on the level of another Microsoft. If we are the platform that every company, every individual out there in the world uses to build useful applications. So we were excited enough about the end game to be very patient, I think, in kind of the short term, but I also don't think we went in and ever thought it would take three years to actually launch this product. If you had asked me, I probably would've said very naively, as engineering timelines are certain to always end up inflating with the fractal of complexity that you don't see up front, we probably assumed we would launch the thing in like six to nine months. And so it definitely ended up expanding far beyond what we expected. I think if I could go back and do it again, I would also probably hire more aggressively up front, because hiring great people, first, co-founders, but then later, employees, is one of the greatest forms of leverage to quickly building a successful business. As it turns out, hiring great people isn't just a way to execute on what you already know you need to do, but those people help you figure out what you need to do in the first place. And they kind of helped shape the product itself, not just implement the details of the product.

So I think it's critical to hire and hire really well early on. It's not easy, especially when you have no brand as a company and you have no real traction, and maybe at best, you have some funding that you've raised, but it is hard to attract the very best talent. So I think hiring co-founder or finding the right co-founders is really important because the co-founders are the ones who are gonna be fully bought in to the vision, to the company, even if there's no clear indication that it's gonna work. I think that's almost definitionally what makes somebody a founder as opposed to somebody who's an executer. Both of which are important, but the founder is willing to take this giant leap of faith and almost stretch their imagination or stretch their disbelief against all odds that this thing is gonna work. So I think that's critical. And part of the value of having co-founders beyond, obviously, having people who can help you think through all the hard problems and actually execute, just be another pair of hands doing all of, you're gonna wear so many hats in building the company in the early days. And if you wear 30 hats, well, if you have co-founders, maybe you can divvy those 30 hats up so each only has to wear 20 or 10. But I also think another part of it is just the emotional, the sharing of the emotional toil of constantly being in this uncertain phase of the company before you've actually hit that real growth curve and product-market fit. It's a very nerve-wracking experience.

Even if you have conviction about the TAM, you wake up day-to-day and you do sometimes wonder, what have I spent a year of my life, or even six months of my life on? And so I think having founders that are really there as your partners-in-crime and emotional support throughout that process is also critical. - And when was the moment when you knew 00:29:42,530 that Airtable was gonna be big? Beyond just your own conviction in the vision? When did that occur? Did that occur before you launched? - Yeah, no, I think there were like, 00:29:53,653 I'll share a few of the key milestones, and I don't think, I think there's always still questions of how big is it gonna be. I think at this point, obviously, we have a fairly scaled business. We're doing real revenue. And we have larger enterprises than we probably could have hoped for in the earliest days, this soon using the product in really, really substantive ways. Like in the case of Netflix, for instance, it's wall-to-wall effectively, and we power a lot of really important business processes for them. But even now, I think we don't take it for granted. I don't think we sit on our laurels and say, hey, we're for sure on the path to become a Microsoft-sized company. I think there's a lot that we're gonna have to figure out and fight for over the coming year. So I think it's always, I think you kind of, at every phase, there is a moment where you're like, oh, okay, I think finally, I feel conviction that we've gotten to this phase of the company.

And sometimes there's a lag where at least, I've always been a little bit, not a pessimist or a cynic, but just somebody who is more conservative in terms of how I think about the company, how I describe the risks to investors, for instance. And so for me, it always takes a little bit of time to let sink in that we've actually for sure achieved this level of success as a business. But one of the earliest real unlocks for us, I think was literally just, we didn't even have a working billing system in the earliest days of being a public product. We'd launched the thing, and we kind of faked a payment entry form. We wanted to convey to the market that we were a real business, we got a real pricing model, and therefore you could trust us, but we didn't actually bother to build the billing logic, to actually start collecting payments and charging people on a recurring basis. And so we basically faked this payment entry form. You actually could put in your credit card and we would collect those details in Stripe, but we weren't actually charging people's credit cards. And I think there was this big moment of truth when we started seeing people actually put in their credit cards, even though we weren't, there was no reason to upgrade. We didn't actually gate any functionality in that super early time, but people and real businesses, I remember seeing real company names pop up and being amazed that people would voluntarily enter in their credit cards, thinking that they were gonna pay real money for a product. So I think that moment of not only can we get people to sign up for the product and use the product, but also to pay for it, I think that was a big leap.

And then once we started actually doing real billing and then started seeing the revenue actually grow in a significant way, I think hitting the million dollar ARR threshold was probably a significant milestone. Obviously, hitting 10 or 20 million ARR was an even bigger one and I think made us feel like, oh, wow, this thing is scaling. There's a real business that we're building here. It's not, again, just a product that has engagement, but there's also a business, there's also monetizability to this product. Even though some might argue that, why would somebody pay for this product when the status quo is Spreadsheets, which are free. So I think we definitely, and we still feel like there's a lot to prove going forward. Can we actually scale this to become a billion dollar-revenue business? That's kind of our next big target that we wanna achieve in the next five years. Can we go and actually also deepen the capabilities of the platform and start serving even more complex use cases? And can we build an ecosystem around the business? - I'm gonna go to questions in a minute, 00:33:45,790 but all these examples, Howie, are after the product launched, but for those three years, you never had a moment where you said, oh, this is gonna be big and I should stick it out? Because I do think founders also wrestle with whether to give up or not. And I want them to get a cue as to when not to give up if you're pre-product launch. - I think the user feedback was pretty critical.

00:34:09,060 I would say the biggest moment pre-launch was we actually got, so we had kind of a very janky backend system working. And I think it was so janky that literally we would deploy a separate instance of Airtable onto a different server for each customer. That's how we isolated different customers instead of having a multi-tenant system. And so we would deploy Airtable to a different server, it's kind of a nightmare to manage, 'cause we'd have to go in and SSH into each server independently and make updates or changes whenever we have updated the code base. But I think we were actually able to get a few real customers, they weren't paying us, but the fact that they were able to actually shift really valuable workflows onto Airtable. One of them was a nonprofit that started using Airtable as an alternative. They were otherwise gonna consider using Salesforce to manage their donors and their volunteers and their programs, and they really didn't wanna do that because it would have been really expensive to hire a Salesforce admin to come in and build out their setup. And they would've been helpless to make changes themselves. That UX would have been slower and less intuitive for their employees than they would have wanted. And so they actually really wanted to use Airtable, despite our fair warnings that the product was super alpha.

And they actually started using it. And honestly, it was probably pretty buggy. I don't think we ever lost data, but they were using it in a very, very raw state. And the fact that despite the bugginess, despite rawness, they were actually getting real value of it, and they actually got really excited about committing to use it. I think that was this really exciting early milestone. And then we probably got other customers like that before we launched publicly. And then after that, I think just seeing more customers come in very organically and do the same was really heartening. - That's terrific. 00:36:04,713 Thanks, Howie. I'm gonna go to the questions from the audience.

So this first question is, to what extent do you think that Airtable's ongoing growth and success thus far can be attributed to the quality of the product itself rather than an ability to effectively sell the product? How do you balance these separate but important components of growth? - Yeah, so I think until very recently, 00:36:29,790 it should all be attributed to the product. In the sense that we didn't have a very, so I think we hired our first salesperson in probably late 2017, if I recall correctly. So it was well into having launched the product and we were actually doing real revenue, I think. We didn't hire that person until I believe we were already at around 10 million in ARR. And 10 million, while still early days as a business, I think it's a sizable amount of revenue. And it was growing organically. So even if we hadn't hired that salesperson, it would have continued to grow. And even to this day, I think half of our revenue is still completely self-served. So people are just signing up for the product organically, figuring out how to use it on their own, pulling out credit cards, and not really the result of sales. And even some of our sale, or a lot of our sales revenue today is also the beneficiary of organic adoption of the product, because these users, especially within companies or enterprises are organically adopting the product.

And at some point, they may ask to talk to sales to negotiate a larger scale license, or to ask questions about enterprise functionality that's only available on the sales touch plans. So I think it's been still the primary reason why we've grown. Going forward, we do wanna put more thought around driving growth more proactively through sales and marketing. And so I think there's a huge market opportunity for us of, every company, every enterprise could be using Airtable. And we can either wait for them to organically discover it and adopt it on their own, or we can find ways to profitably go and target them through sales and marketing. So that's definitely something we're gonna do. But I also think in the long-term, having a really great product matters a lot. We're not gonna become a really large business unless A, we maintain differentiation via product innovation. So at some point, we're gonna have competition that tries to catch up to where the product is today. They can replicate everything that we have to date.

But the key is gonna be for us to be a moving target and to continue bolstering the functionalities of products so that at any given point, we are the best product, we do have this edge and this lead against competitors. And then I think the second thing is, that's kind of how we're going to unlock more TAM ultimately. So we're gonna unlock new use cases, new customer types going forward by continuing to invest in the product. And for us, there's this interesting challenge of, it's not just about making the product delightful. It's also that we have to build real functionality that actually enables those other use cases, while at the same time, I do think usability, delight, et cetera, is also critical, or else we kind of lose that binary threshold clearing ability that we talked about earlier, which is if the product isn't intuitive enough for people to adopt and figure out and want to use on their own without having to read a drive manual or hire an administrator or a consultant to set up, then we've instantly switched into a different category of product and the adoption models can be completely different. And frankly, we lose a lot of the magic and the growth engine of the business. So balancing those two, especially in this genre of bottoms-up adopted, but B2B SaaS, I think is challenging and also really important. - That's excellent, thank you. 00:40:06,690 So there's a question about how essential or non-essential do you think an MBA is for an entrepreneur? I'm not sure if you want, do you wanna comment on that? - Yeah, I'm happy 00:40:12,520 to take that one. I personally didn't get an MBA.

My co-founders Andrew and Emmett did not either. So data point of one for us, at least, it wasn't an essential factor. That being said, I do think there is a lot of value to the MBA-style thinking in a way. I think what I described earlier in terms of trying to vet out the risk factors of the business, kind of fleshing out the business plan instead of just coding up a product and focusing on the UX. I think that is a really important skillset. I don't know that, I think you can gain it from actually getting an MBA. I think you could probably also find a lot of materials, read a lot of the business books, et cetera, that teach you that same thing without necessarily going through the MBA program. But I do think that that perspective of business strategy is crucial. And maybe, I think there was a point where during the web 1.0 bubble, there were a ton of MBAs who were flooding into the server environment. And then I think after the web bubble collapsed, it almost went full circle the other way.

And I think there was almost too much of anathema against MBAs. There is all this disdain for MBAs and this reverence towards engineers or pure engineering talent. And I feel like now, especially with the rise of B2B SaaS companies, fintech companies, et cetera, these companies that are not just about building a beautiful product or a usable, kind of a viral product, which doesn't require as much MBA thinking, but instead, you're building a real business and you have to think about strategy when you build something like a payments or a fintech company. I think that the pendulum has once again swung to a balance where both are important. And so ultimately, I think it probably doesn't hurt to get an MBA. And the skillset underneath it is crucial, for sure. - How significant was your one-year stint at Salesforce? 00:42:08,530 Would you recommend people get a stint like that before they do something entrepreneurial? - It was critical for the formation of Airtable. 00:42:15,990 So I think I was in a very fortunate position within Salesforce where I got to learn a ton. I think partly, that was a result of coming in through an acquisition, having great mentors while there, but I also just think, going to Salesforce and we were sort of intentional about selling our company to Salesforce as opposed to being acquired by, for instance, a consumer internet company. Because I just had this excitement or wanted to learn about B2B software, SaaS, cloud, kind of cloud in general.

And so I thought Salesforce would be a particularly fertile ground to learn from as opposed to one of the big consumer internet companies. And it turned out to be totally true. I just learned so much more than I ever could have. There's no way I would have gotten the insights that were critical to the formation of Airtable if I didn't work at Salesforce. So I think kind of getting, Peter Thiel describes it as secrets, and I think it's very valid. Ultimately, there's gotta be some reason why you're gonna become the founder who creates a company that nobody else could. And it's usually you have some insight, or maybe you have some talent that allows you to have an edge, but one of the best ways to get that insight, to get that secret is to go and work within some other company that especially is aligned with your interests. And you can go really deep into an area and have some imaginative leap that nobody else did. For me, it was seeing the power of the Salesforce platform and then realizing, hey, what if I could go and basically consumerize this? Or, in kind of a Clayton Christensen sense, low-end disrupt it with a product that is massively more accessible, but even if it doesn't have all the same power and bells and whistles as the original did. - And that belief on that thesis, that was a belief? 00:44:07,230 That was a vision that you had internally? It wasn't like a focus group that told you to do that? It came from within the founders, right? - I think often the best, I think, ideas do come 00:44:20,610 from having tangible context or insights.

Just for me to understand Salesforce's app platform and how it worked and how powerful and how valuable it was. But then to have some creative leap of imagination that no focus group, no deterministic process, you can't Spreadsheet your way into a calculation that shows you this is gonna be it. You kind of have to have that leap of faith and imagination. - That's so good. 00:44:49,410 Because you are the poster child of the low-code movement, I'm gonna ask this question, which is, how can aspiring founders who don't necessarily have the skillset to code enterprise-grade software succeed in their entrepreneurial endeavors without funding to hire people? How important was it to have a co-founder that can do more technically beyond prototyping, especially in the early stages? Or at what point did you have to hire, that? - It's a great question. 00:45:12,040 And I think the answer has changed a lot over the past 10 years. When I started, I came to the belief that it would, I actually tried to go and find throughout college, I think, people to team up with who were better coders than me, and I would focus on the idea and the business strategy and somebody else would focus on the coding. And then I just realized, at least for me at that point, I wasn't gonna be able to convince anybody to code on my, why are they gonna believe me? And why do I get to come up with the ideas? And so I realized, it was really essential, at least for where I was, to just go and become proficient at especially web application development. I had learned some C++ programming in high school, just kind of autodidactically, but decided it was really important for me to be able to code up any of my own ideas, at least a prototype form, 'cause otherwise it would just be too hard to convince somebody else to do all the work for me. And to communicate the idea even effectively without a prototype.

So I just committed to becoming a proficient app developer, because I didn't think it would be feasible for me to, it would be an easier path for me to learn how to do that than to go and convince somebody to join me. That being said, today, I think the landscape is a little bit different. And I think, especially for an idea that's not a pure software play. So Airtable is a pure software play in that our entire competency is building this fairly complex piece of software that we sell on a licensed basis. I think there are other types of ideas where the innovation is more from like a business insight. Let's say you wanna create a marketplace for tutors. Or it could be a marketplace for childcare or whatever it may be, the crucial insight there may not actually be in how the app is coded. It may not be in any kind of special software sauce, but more of like some insight into how to attract better the talent on the supply side for tutoring. Or how to better vet them or create a better rating system, et cetera. And so I think in those businesses, there are now rate low-code apps or platforms.

There's, for instance, Bubble to build interactive front ends or Webflow, et cetera. Airtable certainly is used by a lot of these upstart companies as a back end. You can do things, use things like Zapier to create logical workflows so that every time a new tutor signs up on the platform, you can automatically send them an email or do something, push a message into Slack for you to approve their signup, or whatever it may be without having to code that with lines of code yourself. So I think there are more low-code tools now than ever before to kind of get especially a non, or a more business-oriented idea off the ground without necessarily having a full-time programmer. - Terrific. 00:48:17,060 There's a final question on equity dilution on how should an entrepreneur decide on the amount of equity they should give up on pre-seed or seed rounds? - Yeah, I think the short answer is, 00:48:27,790 I think it's all a little bit relative. And the valuations and therefore the amount of dilution you have to absorb in the early rounds has just changed dramatically from the time we started Airtable to now. So to some extent, you're gonna be at the whim of just the market conditions. And even if you were the best company, back 10 years ago, you were just gonna get a different valuation almost by an order of magnitude, maybe more than you do today. I think if I recall correctly, Dropbox raised their seed round at something like a $4 million valuation.

And I think probably today they could have raised even with the exact same level of progress and team and product at 10 times that for the seed round. So it just depends. I do think what actually matters more than dilution is the dynamic of investor expectations and value add. So finding the right investors who are gonna be aligned with your vision and the timeline it's gonna take. The fact that we raise for investors that even though we gave up a lot of the company early on, were patient and didn't put a lot of pressure on us to launch too early or to sell the company when we hadn't launched after two years. I think that actually ended up being more important in the long-term than the marginal, 10, 20, even 30% difference in dilution that we could've taken or not taken during that time. (upbeat music)..