

URL: <https://ecorner.stanford.edu/clips/advice-for-founders-who-cant-code/>

While Airtable co-founder and CEO Howie Liu became proficient in web app development to start his companies, he believes that low-code solutions enable today's founders to launch software-heavy businesses without hiring (or becoming) a full-time programmer.



Transcript

- Because you are the poster child of the low code movement, 00:00:05,467 I'm gonna to 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.. 00:00:26,730 And I think the answer has changed a lot over the past 10 years.. I mean, when I started, I just, I came to the belief that it would, it would be really, I actually try to go and find throughout college.. I think, you know, people to team up with who were better coders than me, and I would focus on the idea and like kind of the business strategy and like, you know, somebody else would focus on the coding.. And then I just realized, at least for me at that point, I wasn't going to be able to convince anybody to like code on my idea.. Like why, you know, why are they going to believe me? And like, why do I get to like come up with the ideas? And, and, and so I realized like, it was really essential, at least for where I was, to just go and like become proficient at especially web application development.. I had learned, you know, some C plus plus programming in high school, just kind of auto didactically.. And, but decided like, it was really important for me to be able to code up any of my own ideas, at least a prototype form, because otherwise it would just be too hard to convince somebody else to do all the work for me, you know? And so, and, and to communicate the idea even effectively without a prototype.. So I just committed to becoming a proficient, you know, kind of app developer, you know, because I, I didn't think it would be feasible for me to, to, you know, it would, it would be an easier path for me to learn how to do that than to like 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 an idea, that's not a pure software play.. So Airtable is a pure software play in that, you know, our entire competency is building this fairly complex piece of software that we sell, you know, on a licensed basis.. Right? You know, I think there are other types of ideas where the innovation is more from like a business insight, right? Like, let's say you want to create a marketplace for tutors, right? You know or it could be a marketplace for childcare or whatever it may be.. The crucial insight there, may not actually be in like how the app is coded, right? It may not be in any kind of special software sauce, but more of like some, you know, insight into how to attract better, you know, talent on the supply side for tutoring, right? Or how to better vet them or create a better rating system, et cetera.. And so I think in those businesses, there are now like great low code apps or kind of platforms.. You know, there's a for instance, Bubble, to build kind of interactive front ends or Webflow, et cetera.. Airtable certainly is used by a lot of these kinds of upstart companies as a backend.. You can do things like use things like Zapier to kind of, you know, create, you know, kind of logical workflows so that, you know, every time, you know, a new tutor signs up on the platform, you can automatically send them an email or, or do something, you know, post a message into Slack for you to approve their, their, you know, their signup or whatever it may, be without having to code that with, with lines of code yourself.. So I think there are more low code tools now than ever before to kind of get, you know, especially a non or, or kind of a more business oriented idea, you know, off the ground without necessarily having a full-time programmer..