Technically working: Shomik Ghosh (Boldstart VC)

We talked to Shomik about how being technical helps at work

Justin
Apr 30, 2020

11 2

Shomik is a principal at Boldstart, a VC fund in New York that invests in enterprise software. He's not an engineer, but being technically literate has helped him through years in banking, some angel investing on the side, and now, a shiny new role in VC. Here's a stock photo of a random person I found on the internet.

Just kidding, it's him.

How "technical" are you?

Depends on the definition. I don't think of myself as technical as the hardest thing I've ever coded is tetris 😂. I've also done a fair amount of data visualization in R but nothing too complex. That being said, I'm one of those strange people that freaking loves diving deep into software infrastructure and after having done that over the past 6 years or so, I'm "technical" enough to be dangerous in conversations with folks who are *actually* technical 😂.

What do you do for work?

I'm currently a VC at Boldstart Ventures, a seed stage enterprise software focused VC fund. My day to day job includes meeting with entrepreneurs that have an idea for a new product, deep diving into technology to understand why the product may be useful, talking with Fortune 500 technical executives to understand their software architectures and pain points, and talking with potential talent to either join our companies or learn about new technology/use cases that I hadn't considered before. Basically, I just have to be a very good listener 😀.

At Boldstart, we invest in companies both pre and post-product and strive to be the first check into our companies, helping partner closely with our founders to help them scale thoughtfully. We love technical entrepreneurs solving F500 enterprise pain points or approaching modern workflows/applications with beautiful design and an opinionated view of how work should evolve. A few portfolio companies of ours are Superhuman, Front, Snyk, BigID, Kustomer, and Security Scorecard.

How has being "technically literate" helped you get better at your job? Any specific examples?

I could not do my job without being technically literate. The founders that we talk to on a daily basis are software engineers, product managers, data scientists, designers, and generally highly technical people. Not only would I lose their respect, but I would not be able to actually help any of our companies scale without being literate enough to describe what they do to potential customers, talent, or other investors.

Recently one of our portfolio companies, Spectro Cloud, publicly launched. Investors and intrigued customers reached out asking why they were different from other Kubernetes management software companies. I wouldn't be able to have a conversation without knowing what k8s is, what containers are, what virtual machines are, clusters, nodes, etc. It's not enough to just know the words either: you need to know how the systems and products interact and why those are important in the context of the person you're talking to.

It's frankly why I was so excited when Technically started and loved the Developer tooling post. This sort of information needs to be out there so we can all be more literate on the software infrastructure that we increasingly rely on every day.

Have you ever tried learning how to code? How did it go?

In college we had to take a mandatory intro to programming course. I learned how to code in basic Java (sadly before the rise of Python in intro classes). The analytical thinking behind understanding how logic works actually helped me a lot as I moved into banking and finance. I was basically able to "code" in excel. I don't mean in visual basic (although constructing macros was important) but in terms of nesting multiple functions and visualizing how the logic would flow through as one function would be represented by like 20 functions within each.

I recently tried to learn Python in a Coursera course but life and work got in the way of taking it to completion. It's still bookmarked though and I still have my IDE ready to go when I pick it up again!

What's your go-to (aside from Technically, of course) for leveling up your technical knowledge?

So seriously first, I think Technically is one of the most underrated newsletters out there. It's crazy how much stuff we take for granted in software. There's literally millions of lines of cables connecting data centers to route information around the world. We're talking cable on the seafloor lower beneath us than Mt Everest above us! Some people may not care but for those of us who get lost on random wikipedia deep dives, Technically is saving us days of time by condensing all this info in an easily understandable manner.

Aside from that, I rely heavily on friends that are software engineers, data scientists, product managers, and designers. I love reading engineering and product blogs. Technical blogs from companies like Segment, Superhuman, Netflix, etc. Podcasts like: Software Engineering Daily and Enterprise Ready. Newsletters like Technically, The New Stack, API Evangelist, DevOps Weekly, and What's Hot In Enterprise.

Also, Twitter is invaluable. I try to post stuff on my twitter – @shomikghosh21 – but rely on many others to surface great things for me to read. Please hit me up on Twitter if you have any interesting articles or thoughts to share or just want to connect.

2 Comments

Write a comment...

Wannabebuysidebro May 11, 2020

Not to be rude but in my humble opinion I think this interview should be a bonus to your 2x monthly short primers.

Not much 'technical' value to be gleaned from this interview and seems kind of like a lazy filler post.

♡ Reply Gift a subscription Collapse •••

1 reply by Justin

1 more comment...