### AN INTERVIEW WITH MR. ERIK TROAN



From left, Jason Li, BSS Editor-In-Chief; Eleanor Xiao, BSS Publication Editor-In-Chief; Dr. Jonathan Bennett, BSS Faculty Advisor; Olivia Fugikawa, BSS Editor-In-Chief; Megan Mou, 2020 BSS Essay Contest Winner; and Mr. Erik Troan '91, founder and CTO of Pendo.io.

#### To get us started, you are a self-proclaimed technology geek. When was it and what was it that first got you interested in computer programming and engineering?

I've been doing [CS and Eng] since I was probably five or six years old. Going back a few decades ago, people didn't have computers at home, but my father was at IBM so he was always around them. He put me in a programming class at the library when I was probably six or seven. We also had a little computer at school so those two things made me kind of interested in exploring the world. I know what I enjoyed about it then is probably what I still enjoy about it now; it's a combination of problem solving while doing it in an abstract environment where you really can control a lot of things. Many of you probably enjoy math. Math is fun because there is a right and a wrong, correct? You get to solve a problem. You're solving a puzzle, but you're not constrained by inconveniences like physics, gravity, forces, or having to go build things and buy equipment. For me, growing up, it was always a way I could build and explore things. I could do things that didn't have the constraints of the physical world or have to get materials and put them together. Physical things break and then

you have to fix them. So I think that level of control over computer software is really what made me interested in it.

### You've highlighted NCSSM as one of the most valuable parts of your education. What was your favorite part of your experience there?

The other people. It's all about the kids. It's about the social scene—and I don't necessarily mean social like parties, but social in terms of, you're in a peer group that's smart. I've probably said something like that at Convocation: you're not going to sit in a group of people as smart and as energetic and as intellectual and as curious as you are ever again and that leaves a mark. You really do realize there are like-minded people out there; you have interesting conversations and you're solving interesting problems. I remember I took a fractal class senior year and just the excitement of the kids getting together to solve the problem, and to really learn about it, is what really made an impact on me.

I learned the importance of communication, so I learned to write. I had some very good instructors. I learned from them that writing was something that helped you communicate, and if you want to become effective in this world, you're not going to do it mostly one-on-one; you're going to do it one-to-many. Even in the world of the Internet, Youtube and everything else, writing is probably the most basic way that you're going to communicate to a lot of people.

I also really fell in love with history, of all things. I had a great U.S. History teacher. That's followed me my whole life. I'm still a history buff; I read history books all the time. I was so fortunate, just like you, to have instructors who pushed me and made me better all the time. I have a son who is [in NCSSM Online], and it's great watching him online and how he kicks me out of his room so he can do his classes because he's getting that extra push and really interesting problem-solving material. He just started his new Bioinformatics class this week and you can just see the lights back on in his eyes.

### What do you do at Pendo now? What does a day look like for you?

There are two different parts to that question. My official job at Pendo is that I'm Chief Technology Officer, and that I run all of Product Engineering. All the products that we build come out of my team; it's a team of about 110 people now. [Our product] runs in the cloud (out on the internet) because it's a software product. The team deploys that product, keeps it running, manages automated testing, and does 24/7 monitoring. All of that is all under me. So [my job is to] get the product, work with the team that defines what we should build, figure out how to architect it, build it, get it shipped to customers, and keep it up and running.

What do I do every day? I go to meetings pretty much all day-the most important part of my job is helping with communication within my team or across teams. I don't make the big decisions. If I'm making a decision, there's probably something wrong. It's much more about getting the right people in the room, enabling them to make the decision and make sure they feel supported and empowered to make the decision. Asking questions is also a big part of it. I actually sat in your chair a few hours ago going through whether or not we should use a third-party partner for something new we're rolling out. There was a team of three people that told me two or three weeks ago that they were really excited about it. Now they all changed their mind and all I did was ask them questions about their decision. "Why?" "Have you thought about this?" "What about that?" I didn't make the decision though. We were going with their recommendation, but I helped to validate and make sure they had the confidence that they thought through everything. So I do that with both Engineering and Product. The other thing I'll do is work with Sales,

Product and Marketing to try to understand the market. I'll talk to customers about what Pendo does for them and where they are having a problem. I think I have three or four customers that I'm an executive sponsor for, so another part of my job is making sure that they are successful. It's a lot of talking. I joke that the CTO is actually Chief Talking Officer; some technical in it, but not much.

One thing in a fast-growing organization, you'll find, is the problems don't change; the problems change scale. When you solve a problem for a team of 20 with the right solution for 20, when you get to a [team of] 100, that solution probably doesn't work any more. For example, you might realize that you didn't have a process for something at 20. That didn't matter at the time because two people went to lunch once a week, saw each other informally at lunch and communicated, "That's not working," or "Fine, I'll fix it for you." Then they go back and they fix it. When you get to 100 people, those two people could now sit on different floors and don't go to lunch any more. All of a sudden, you have a hole where things that were solved informally can't be solved informally any more because the roles that you need to have around the table drift apart. That doesn't mean you screwed up, or that your process is wrong, it just means that you've grown a lot, and you have to make sure you're constantly looking for things where you missed or where you can do better. So as you hire specialists, they can thrive without there being gaps between all the specialists.

### Do you miss the coding part of your job or the engineering part of your job?

No, because I still do it. This is one of those things where it's either the best thing about me or the worst thing about me as the CTO. I'm still technical and still want to code. It means I have to hire and surround myself with people who complement that aspect of me. I have two VP's of Engineering who don't code. So, I brought people in to take things that gave me a little more room to still be technical. [But] I have to be very careful with what I code. I tend to code things that are longer term, things that aren't going to be blocking a customer, or things that are not a promise that we are going to get it done by a certain day. I've been told that most of the team loves that I [code], because it makes me relatable and I can understand what they're going through a little more. But it also does mean that there are other parts of my job that I delegate and I do differently because I insist on continuing to code.

### What is your advice to students who want to learn more about coding?

Go to school. Go to college. I'm a huge believer in formal computer science education. I hire people who don't have it, so I do hire coding academies and people who have just picked it up, but the easy path to success if you want to be in a computer software job is to go to college in computer and software. You will learn things in school that you won't learn anywhere else. I took a lot of graduate classes as an undergraduate and I still use those lessons. Some of the best engineers have Master's degrees. I haven't hired a Master's degree who's not a good engineer. These things go together. People get smart in graduate school and undergraduate for a reason, so don't let that go.

While you're [in school], do as many internships, real world projects, volunteer projects, and capstone projects as you can. You've got to go solve problems. You've got to exercise the muscles in your brain to let you try different ways of approaching problems, because there are all sorts of different ways to do it. Unless you've done it 20 times, you're not going to know which [solution] is the best. College and graduate school absolutely accelerate that process.

## What is your advice to students who hope to start their own business?

To start a company, you've got to find a problem that you are passionate about, not a solution you are passionate about. I tried a [starting a company with a] solution that I really liked and that company fizzled and failed because it wasn't a problem. Find something you think is worth solving. Find customers who think it's worth solving-ideally, before you build it. You can't always do that, but if you can find some way to get people who will validate this as a problem worth paying for, there's nothing to replace that. One of the reasons I got really attracted to Pendo was, our CEO gave me a problem that I understood, was excited about and resonated with my career. He started talking to potential customers who said, "Oh yeah! Talk to this guy, they said this." So he could clearly come in with a head start towards, "Was this a valuable problem to solve? What was important about the problem? How is this really going to help customers?"

Another thing I would say is: it is very easy, and I've been guilty of this in my career, to take customers for granted. They are going to make you successful. The only things that make you successful are their willingness to pay and your willingness to make them successful. So don't take them for granted, especially if you're talking about low prices and products. It's very easy to say, "Yeah, we have to make it up in volume." or "If any one customer is unhappy, we can kind of blow it off." If one customer is unhappy, it's probably a thousand customers who are unhappy. Make your customers happy. Your customers will carry you everywhere. We have been very lucky here. We've done five venture-funding rounds. Every one of them was preempted, so we never went to look for money. Investors came to us, wanting to put money in. The reason that happened was our customers. The first one that found us was because we had already sold to six other portfolio companies, and they started hearing about Pendo, Pendo, Pendo. "What is this thing? Why are you guys all using it?" They were so happy with the value we were delivering that they told their investors about it, who told their partners about it. It happened again and again. It wasn't any magic. We have seven core values, and one of them is "maniacal focus on the customer." We are absolutely dedicated to them. I think out of everything I've learned, that is the one thing that I have discounted in my career, and I wish I hadn't. It's so important.

### Many of the students here at NCSSM view failure as something shameful. How did you overcome the failures that you encountered in your life, and how did you learn from them?

Practice. I used to be a private pilot. There's a saying around that, "The only way to avoid mistakes is through experience, and the only way to get experience is by making mistakes." Failure is a side effect of taking a chance, taking a risk, pushing your zone. If you don't fail, you're not trying very hard. And I get it-I watched my kids go through middle school and high school. I get that we built a society for students where failure can be very, very expensive. You guys get that one "C" in a class, and there's colleges you might not get into or scholarships you might not get. You're in a weird place in high school. Don't mistake how you succeed in that microcosm for how you're going to be successful in life. You're going to be successful in life by growing, learning, and trying new things. One of the reasons that [the US] is still one of the premiere economies in the world is that we don't punish failure. I failed in my last company. It was a big crater in the ground, burned through tens of millions of dollars of other people's money. When I helped raise money for Pendo, people saw that as a plus. They didn't say, "Well, your last one failed, why would I invest in you now?" They said, "Hey, what'd you learn?" and I said "This and this and this, oh my god I learned so much," and they'd say, "Great, let's do this." You're lucky to be in a country that, as you move out of that narrow, high-achieving academic track that you're on, failure won't always be punished. So embrace it and try things. You have to try new things or you won't get better. I have a really smart PhD [employee] here, who worked on a project for 6 months. He spent 6 months of engineering on something. It didn't work. At the end of the day, we had to shelve it. But if we didn't try, we would always be wondering "What if?" We had to figure out other ways to solve it, and that approach was a bad one. So, I get why kids at Science and Math and in high school are scared of failure; I think you've been taught your whole lives to be scared of failure. Just try and get over it when you're in college. It's going to be very hard to do anything special unless

you take some gambles. You can't win every gamble. But when you fail, look at why. I learned more from the company that didn't work than from the companies that did. I've hired people who have been fired, by the way. The question is, "What did you learn from your failure? Why did it not work? What was it about you that had to change?" and if they say "it was the company, there were bad people there, they didn't listen to my ideas, they did everything I told them not to" — run away. That's not learning. If they say "Yeah, the company had this culture, and I brought this culture, and I had trouble adapting," or, "I really thought this was gonna work out, and I pushed hard for it. I think I alienated some people" — those are the kind of people you want, because they'll probably be better next time.

#### What do you enjoy doing outside of work?

I scuba dive. I'm actually a scuba instructor, so I teach one or two classes a year. I'd like to get back into flying; I'm looking into doing that in the next year or two. I'm a pretty avid snowskier; I try and go out west a couple times a year. I run hundreds of miles a year—that's not something I enjoy, I do it because I'm old. I like to cook. It's the highlight whenever we can get the whole family around the table. I have 3 teenage boys, so it doesn't happen that often, but when we can all sit down to a meal, our meals tend to last one and a half, two hours. They're very long, drawn-out meals after the food's all gone. We don't see each other that much so we enjoy that. I like to travel. Whenever I can, I'll take a trip overseas—not for work, but if I can travel personally, it's great to just be in a new place and wake up somewhere else. I also like to read. I read a ton.

### As a scientist who works in the entrepreneurial side of things, what do you have to say about the role of entrepreneurship in advancing science?

First of all, I'm not a scientist, I'm an engineer. I consider those really, really different. When I didn't go to grad school, I turned away from being a scientist. Scientists discover new things. Engineers try and apply science to problems. I think engineering and entrepreneurship are really tightly coupled. My personal belief is that most of the time if you engineer something new, you probably need a new company to really make it successful. The iPhone is a huge counterexample to that, but most truly novel engineering projects have a company to go off of because [they're] bringing something new to the market.

Science is harder. A lot of science has around 30 or 40 year payback periods, and it's not that they're not interesting topics and that they're not important. If you're lucky enough to be a scientist, and you're able to do it in an entrepreneurial sense, I think that's great. But for most science, I think, you need to figure out how to do it through research grants and universities where you don't need to have that immediate payback period.

Some of the most interesting things that come out of science start as an intellectual curiosity. They didn't start off as solving a real-world problem, but they all feed into real-world problems, into the hands of people like me, who are engineers. When you start crossing those wires too much, you get into bad positions like you have in medicine, where a lot of research in medicine is being privatized. If it isn't for a disease that has enough payback, then it gets shelved, so you start getting the wrong incentives around basic research.

# How important is entrepreneurship for applying the knowledge that comes out of science, or for doing engineering?

I think for building real solutions to real problems, it's critical. If all we had in the world was fifty Fortune 50 companies, then we wouldn't get innovation, and we wouldn't get new products to solve new problems out there. The easiest example is alternate energy. It's not Exxon and BP that are bringing alternate energy and clean energy to the world. They have a business—they dig up oil, refine it, burn it, and make a lot of money doing it. It's hard to take a business like that and go do something that's new and different. It's a risk. It's a lot easier to take a risk as an entrepreneur because you have a lot to gain, but if you're running a big company, how do you value that risk? You have a lot to lose.

[Entrepreneurship] is critical. I think universities are normally pretty bad at it. Stanford is uniquely good at it, and I don't know if it's [because they're] Stanford or if it's because they're sitting in the middle of Silicon Valley and have hundreds of billions of investment capital sniffing around looking for the next opportunity all the time. Mostly it's disappointing when you look at universities that have so much pressure to raise money and own part of their inventions that they aren't good at getting these companies up or bringing the technology up. Which is too bad because I think there's a huge amount of symbiosis between those two.