



Podcast Episode #63 – Your (hardware) product is not your product - think in terms of business models, with your host Balint Horvath, Switzerland

RAW TRANSCRIPT OF INTERVIEW

Hi. Welcome to the show. Your product is not the product. Many of us in high-tech entrepreneurship or entrepreneurship in general come from a technical background. Many founders have an academic track record where we've been trained about all the technicalities of how to build products. We know how to calculate mechanical properties, strain under certain stresses, how to create CAD drawings, how to calculate signal to noise ratio when detecting RF signals and so on. But what is entrepreneurship about, specifically hardware entrepreneurship? In this episode I want to highlight some points and talk about this question that I just posed.

First, if it's not the product that matters in a business, what matters then more? That's the first topic. Second, once you have the pillars of your business should you rush to come out with a fully-fledged product that is the perfect reflection of your vision? Lastly, in this episode we're going to talk about a situation, specifically about structures, framework of how you can think about hardware startups and their products with some common examples.

To start with, because of our technical background we think in terms of technologies, solutions, products so almost automatically we can't help coming up with ideas, potential solutions when we hear about a problem. We fall into the trap of falling in love with the solution. To use an analogy starting out with this solution is like building a high-tech hammer, an IoT connected hammer, if I might say. Not sure if it makes sense at all though. And then you start looking for the nail, the wall and the picture that you would like to hang up later and then using the hammer to drive the nail into the wall. In other words, you have the hammer to solve a problem with it - how to hang up pictures to the wall.

As you might know we as humans are not perfect. We have many biases and this innovator's bias is one of them which is especially strong and potentially harmful for entrepreneurs. Customers don't buy your product because they love your product but because they love the fact that it solves their problem. So what is the solution then to this problem-centered thinking? I believe that instead we should search for the problem and the matching customers, early adopters in a systematic way. There are a



number of ways for doing it. You can talk to customers, interview them, this means face-to-face meetings or via Skype, to interact with them with a short, very fast feedback loop. You can also scratch your own itch so you could solve your own problem. That actually is one of the most powerful motives I believe. It's a driving force to later overcome some problems. And however this should not be exercised on its own in a singular way to come up with a great idea since solving your problem doesn't mean that you'd also solve this way many other people's problem. You have to check if there's a big enough market for it.

Another way to come up with great ideas and corresponding problems is you can immerse yourself in the environment of your customers you want to serve later so you want to find out what the ecosystem is, how the whole system they are in works, what their workflow is like and if you see a problem, you might be able to innovate on it with higher chance of success than otherwise. Another way to come up with a solution is like the fourth, is a bit similar to the last one that I just mentioned but it's not as active is observing customers, what they do, how they use certain products, if they're happy with it or if they're not happy where you can improve on the situation on their journey, customer journey. During this problem search we can run around in circles endlessly constantly looking for a problem.

So how can we break this pattern? Many times we think constraints are bad like we don't have a big enough team or a good enough team, we live in a remote country outside Silicon Valley, we don't have the right funding yet so no money but constraints can actually be super helpful. There's so helpful that if you don't have a constraint, well, first think about it actually, rethink if you really don't have a constraint because most of the time there is. But if you cannot come up with it, you might want to force yourself to come up with one even if it's an artificial one because I believe that's how most of innovation happens. As an example in hardware the Audi team wanted to win the Le Mans. I am not sure how to pronounce it in French. I think Le Mans, competition in France. It's a race that's extra-long. It takes 24 hours to finish it. They didn't have the fastest car though, nor did they care much to get one or build one. But how can you avoid following this obvious route to victory? So not winning the race without having the fastest car. Before Audi, BMW and Mercedes were winning the race so the stakes were higher, even higher and even better looking for Audi. These are the three big German automakers, as you might know.

Audi's chief engineer asked the question, "How can we win in spite of not having the fastest car?" The answer was focusing on a single thing, property, efficiency. This is efficiency. What do you have to do when you have a race for 24 hours? You have to put back some fuel. You have to refuel the car by reducing the number of pit stops Audi could win the race. The technology was the means to it. They used diesel technology. This is I think a good example for working with constraints.



Continuing with the hammer example 3M, the company 3M had a constraint too. How can we fix a picture on the wall which is the job to be done without having holes in the wall, without drilling holes or putting nails into the wall using a hammer? This is how they came up with the mountain tapes to hang up objects on a wall up to a whopping 30 pounds or 13 kilograms. Not bad.

If it's not your product that really matters, what matters then most for hardware entrepreneurs? The job of an entrepreneur or an entrepreneurial team should be to create a repeatable business that scales. This is done via a business model which is a story about how an organization creates, delivers and captures value. This latter quote about creating value, delivering and capturing value is actually a quote by Saul Kaplan, founder of the Business Innovation Factory. You might want to look into that and his work. Business model has different elements. Using Lean Startup methodology, terminology which is one of the most prevalent one today, innovation framework, you can capture the elements of your business on a Lean Canvas which is a one-page representation of how your business works or will work instead of having a long 20-page business plan. There are different elements of it. I will not go too much into details in that. You might want to look into the work of Ash Maurya. Now he has a really, really useful framework on Lean Stack for thinking about it. Just some of the elements are like problem, existing alternatives, which answers such questions as "What problem do I want to solve? What are some existing alternatives which solve the problem?" which can come from even other categories. In case of a car the problem it solves is that you want to get from point A to B. An alternative existing solution could be just going on foot or using some public transportation.

Another box on the Canvas is the customers together with the early adopters below it. Who has a problem you want to solve? And who has the biggest pain? This last group so people who have the biggest pain forms the early adopter group who would respond to your solution the most intensely. Then comes another box, unique value proposition. How does your solution differ from others and what value do you create uniquely? This is what it answers, this box. Still another box is the channels which is about how you can reach your customers. Then you have in the bottom right corner, you have revenue streams which answers the question "How can you monetize the customers? What revenue can you generate?" And then there are also other elements to it like a cost field, solution, unfair advantage, key metrics. I'll put a link in this episode to some resources.

The second part of this episode is about a situation when you have a business model at least with assumptions and the question is now how do you go about attacking, tackling your assumptions on the Lean Canvas. Every wise entrepreneur looks at the biggest risks associated with the assumptions that he made and addresses those especially starting with the biggest risks. What could have happened to Richard



Branson if he had actually bought his planes when he started out Virgin Airlines in case the idea had not worked out? He would have been in trouble, in big trouble, in great debt. Instead he got a guarantee from the owner that in case the business doesn't take off he could return his planes without a problem. So he was renting the planes and testing his business idea.

In a similar way, when Tesla started out what do you think was the biggest risk to tackle? Was it factory related, so manufacturing, profit or the fact that they didn't have a brand yet? No. It is actually related to the fact that Tesla had to make people switch away from using combustion, combustion-based cars to electric cars. What's one of the big advantages of combustion cars? It's their range, how long they can go without refueling them. At that time when Tesla started out after 30 miles, the cars, the electric cars had to be recharged. So to beat that critical number, 30 miles, they had to create a car that can run 200 miles on a single charge. So if then that's your big goal, why would you want to build the car from scratch? This is how they started out to work with Lotus and they actually license their body parts to come out with a Tesla as a big first step, to come out with the Tesla Roadster in a very short time. What would have happened if they had instead started out with the series 3, the latest model, the more affordable Tesla that is being sold these days? No matter how clever Elon Musk is they would have fallen flat as there are way too many risks to build a car from scratch. Take a look at Elon Musk's secret plan for Tesla from that time. It says, "First, we want to build a sports car. Then use that money to build an affordable car. Then use that money to build even more affordable cars." The message is tackle the biggest risk first and then the next risks so allow time for your big vision to materialize, roll out your ideas in stages and phases.

Lastly, on this episode I wanted to touch on the topic of typical business models for hardware products. These give life to your product on the market place. The way they are labeled are based on the way they generate revenue so if they have a short or longer cost of a lifetime. Traditionally before software came around we had hardware, we built hardware physical products and sold them if possible in big quantities so the cost of producing them could be kept low. Nowadays such a model, especially for high-tech products is failing, it's going obsolete. One of the most famous examples is Jawbone which went bankrupt last year after raising more than \$900 million in total. But there are exceptions like I had on my podcast Sproutworld, Michael Stausholm who bootstrapped the company so started out from scratch knowing investors and he has a pencil as a product and is still thriving his business. Well, still, the old model is dying.

The other that is more and more dominant is hardware as a service, HaaS, similar to SaaS software as a service where you pay for the hardware and afterwards you pay for the usage on a regular basis monthly or based on the data transfer. As a result,



the customer lifetime value goes up which is opposed to be backed on cash flow or revenue. Just one example is DipJar a connected tip jar or donation box. Another model that is becoming prevalent is the hardware-enabled services. It's again similar to HaaS but the additional services is optional. This however presupposes that you make enough money on the actual sale of each unit. One example is Fitbit or Runtastic which I use their app where you have a watch and with a premium model, premium plan you can pay for extra services like I pay for a marathon training plan for Runtastic.

Lastly, there's the consumables model where you pay little for the hardware but then you pay later for the consumables. What you use up, the consumables you use up when you make use of the hardware. This is also called the Keurig model which is originating from the company that sells coffee, coffee machine and capsules where you pay for the capsules as consumables. This is also what Nespresso does. A key component of this model is that consumable should not cost very, very much. So something like a max 15-20 percent of the original hardware as otherwise customers would become not so happy customers since they would smell your intention of using more like Gillette blade model where you Gillette charges a lot for each new blade, roughly 60 percent of the price of a razor which is outrageous.

To summarize, in this episode I discussed three main topics. First, I elaborated on a common trap that a lot of founders run into that they build the product first without thinking deeply about the problem, without validating with the customers that there's a problem solving for them. Your product is not the product but it's the business model. Second, I discussed the need for looking at the risks and attacking the biggest risks first. This led also to the staged rollout used by Tesla and also actually other companies like Apple with the iPhone when they came out in 2007 and they came out in stages introducing new features. The last topic was business models for hardware startups with some examples where a higher customer lifetime value is targeted. Feel free to let me know what you think about this episode and if there are other business models you see for hardware startups or let me know how your hardware startup operates. How do you tackle your biggest risks in your business?