Stanford eCorner

Skin in the Game

13-03-2019

URL: https://stvp.stanford.edu/clips/skin-in-the-game-2

Forget market surveys, urges Alberto Savoia, Google Innovation Agitator Emeritus. He explains how to generate YODA – "Your Own DAta" – by building simple "pretotypes." He demonstrates how IBM was able to test market interest in a speech-to-text program before building such a program was technically possible.

Transcript

- So I urge all my entrepreneurs to collect your own market data.. Firsthand, fresh, local, recent, recently collected, and most importantly your data needs to have skin in the game.. What do I mean by skin in the game? If I asked you, "what do you think of my idea?" And you tell me, "well, Bert, it's good." I say, "should I leave my job to pursue it?" And you say, "sure, go, go for it." Right, you have nothing at risk.. Skin in the game means that as an entrepreneur you're putting, you're risking, right? You're risking your time, your reputation, your money, to go and start a new venture.. You're putting your own skin in the game, you wanna get skin in the game back from the market, right? And skin in the game can be the market's time, money, commitment, information, reputation, something of value and at risk.. Let me give you a very simple example.. Suzie came up with a great idea.. A smart hammer, so you hit the nail instead of hitting your fingers.. She goes and asks people, "hey, you know I'm thinking of this hammer, "would you buy it?" And some people said yes, good idea, other people say bad idea.. Does this count as data? No, right? This is opinions..

Now, in another scenario, she says, "well, I'm planning to build this hammer, "and if you give me a $50 deposit, "I will make sure that you get one of the first ones." So, some people say it's a lame idea, they're dead to me.. You know, just like in Shark Tank, "you're dead to me!" But the other people, instead of saying it's a good idea, if they actually open their wallet and give you some money, you have the first indication that the market really is interested in your idea, right? You fix an asymmetry.. Before you put in your skin in the game, by quitting your job or getting this the funding, get some from the market.. So, that's what we call "YODA." You can quote me on this, "it's much easier to get people "to open their mouth than to open their wallet." The hardest thing an entrepreneur can do is to get companies and people to open their wallet.. I've been told so many times, "Alberto, it's a fantastic idea, go build it," and then when I went to sell it to them, "well, yeah not tonight, we're having a bad quarter, "forget about it," right? So it's very important, you need to change the way you approach market research from this traditional model of doing market surveys, or asking people this question, "If we build it, will you buy it?" Which is how most market research is done, by the way, right? You ask these questionnaires.. You flip it around, completely, 180 degrees.. Think about this, "if you buy it, we will build it." Now this seems very counterintuitive, but I will give you an example of exactly how to do it.. Because now you're asking, "well, don't I have to have a product built "before I can see if people want to buy?" And the answer is no.. You don't build it, you pretotype it.. This is not a typo, this is a word I invented, pretotyping, because it had to be invented..

So, strategy number five is pretotype it.. What is a pretotype? Well, it's the simplest artifact or technique you can use to collect YODA very quickly and very inexpensively.. So there is a big range between having an idea and the final product.. That could be years, months, you know, millions of dollars, and the way I see it, pretotype comes very very early.. A pretotype is something that you can build in minutes, hours, maybe a couple of days, and it should cost anywhere from zero to maybe a couple of hundred bucks if you really feel like splurging, right? A prototype is something that actually works, can do something, and can take a couple of days, couple of weeks, couple of months.. I've worked on prototypes for software development tools that took a year and a half, just to prove that the thing would actually work, and of course the products take a long time.. So, pretotypes are things that you can build very very quickly.. Literally in a couple of hours.. Let me give you an example of pretotyping.. Something that actually got my thinking about this process..

About 34 years ago, IBM wanted everybody to have personal computers.. But you know, 40 years ago, most people didn't know how to type, right? Those of us that were around there will remember, this is how people typed.. So they figured, no

way that people can use a computer if they have to learn how to type.. Who types? Programmers, writers, and secretaries, right? Nobody else knows how to type, nobody wants to take typing lessons.. But they wanted to know, "well, if we solve this problem of speech to text, "so you can just speak to a computer, "will people actually buy our product?" So they did something very clever, an experiment.. They brought people in the room, they gave them a microphone, a monitor, no keyboard, and told them, "look, speak into this computer, "we have a prototype of speech to text." And people spoke into it, and magically the computer did whatever people told them to do.. Of course this was not possible.. Those days, even the fastest computer couldn't handle this.. So what was happening? Well, in a room next door, one of those amazing people that can type as fast as you can talk, was actually transcribing everything that was being heard through the computer.. And I tell you as an engineer, this example kind of really messed up with my mind..

'Cause if you go and.... How many engineers here? How many, yeah that use soldering iron, etc.. So if you come to an engineer and said, "Alberto we need to build a prototype," I say, "great, fire up the compounder, "or the soldering iron," right? I thought, "this is not a prototype, "they're just pretending that they have something to work." So in fact I created this first word, "pretendotype." 'Cause I knew, this is not a prototype, it's not like IBM was planning to breed a race of small typists that fit inside boxes and you feed them cheese and crackers through the floppy drive, right? So it's not a prototype, it's something completely different from the end product.. And then I shrunk the word to "pretotype," because it's easier to pronounce.. But remember, pretotype means "before a prototype," but also, "use your imagination to pretend." So, what IBM learned is that before the pretotype test, a lot of people thought, "of course we want a speech to text computer." Now, if you try to actually use a speech to text computer all day without a keyboard, your throat gets sore, the room gets very loud, and you cannot dictate, or work on confidential things like, "fire Bob." Oh, sorry Bob, didn't know you were around here, right? So, in thought land, speech to text was a sure winner, when people actually tried it, realized it was not.. So, this is an example of what I call a Mechanical Turk pretotype.. You know, and these days everybody's talking about robotics and A.I.. Before you spend four years and 40 million dollars to create some automated pizza maker or something, something else, you can use human beings to simulate that behavior.. It doesn't scale, but you can learn if people would actually use it.. A prototype, we don't know, is something typically built to figure out, can we build it? How long will the battery last? How will it work, etc..

etc.. It basically asks the question, "how do we build it right?" And here's the secret, most of the time you can build it.. Now if you tell me, "I have an idea for a time machine." You know, I call the guys with the white coats, and tell you you probably cannot build it.. But most of the ideas people have are buildable.. A pretotype asks a different question.. It asks you, "should we build it?" No, will I use it? What will I use it for? In other words, pretotype asks the idea, it asks, "is this idea the right it?" Something that if I build competently will succeed...