Productize Not Customize

Video Transcript:

Paul: Hi, it's Paul Clifford from Distrupware and today's message is **never ever customize your product**, always productize and I'll talk about that in just a second.

Okay, so here's the situation...you have your product out there, you're an early stage startup and you're starting to talk to a lot of customers and **some really big enterprises are really interested in your solution.** So you're getting closer to getting them on board. They're a great name. It's **going to do wonders for your business** and so you're really excited to have that and when it comes to the crunch then they come out with that phrase, "I'll buy it if you can do X and Y for me".

So basically what **they want is customization** and this is where the alarm bells need to start ringing. Okay, you've got to remember that your product is a marathon. It's not a sprint and the quick wins like this, **they look great on the surface but they have a huge pain behind them**.

So they want customizations, you don't want that because **you want to stay as a product**. The thing is customizing any solution even though the ideas are probably quite good, it leads down a rabbit hole. Okay, a complete mess, you know, you have to do things like create branches and at the end of the day you're not a software development house, **you're a SaaS company or a product company** and in my experience it always goes wrong. It's a lose-lose for you and the customer.

You lose because **you have to commit resources to features that aren't necessarily on your road map.** You then have to maintain more than one branch of code. You need to spend more time with them in account management and their bug fixes are probably not going to be necessarily aligned with the bug fixes for the core product. And sometimes, I've seen it, **where clients want to claim IP or intellectual property over their ideas** and of course, you can't then reuse those ideas for anything else so it's a complete nightmare.

And of course **the client loses as well**. Although they don't see that straightaway and you need to explain this to them. For them it's going to get more and more expensive to maintain and of course, you have to charge them. **You have to charge them maintenance on top of the customization**, because it's not a SaaS anymore. And they get behind your core road map because they're on a separate branch.

You're product is moving in this direction and every time they want some new features or on your core you've got to merge those in, which **creates another project, another stream of customizations, another stream of bug fixing** and everything and it gets more and more expensive.

And one of the worst things that I've seen is at the deal stage, when the customer says, "yeah, I want this but I want X and Y", what normally happens is once they go live and they get implemented and they start adopting it, **they actually come up with a whole load of other priorities that are actually more important than the initial things that they wanted** at the start of the project.

So then **you're in a huge mess because you're trying to shuffle around and re-prioritize stuff** that you've committed to and then they want other stuff because now they're starting to use it and they're starting to see things that they want to change.

So basically never go down that road. Okay, but I understand the pressure, I've been there. So this is how I've solved that previously in some of my previous software companies and specifically one big SaaS startup, where you had a lot of pressure like this.

So what I do is I **create what's called a 'Customer Expedited Road map'**. Okay. And the way to handle that is that essentially, first of all, when a customer asks you for the customization, the **blanket answer is "no"** and you outline the reasons that I've already gone through. Secondly, you explain to them that you do have an expedited road map process and this is how you go into a bit more detail.

What you do is you take their requirements, you look at your road map and you work out how you can create product features to deal and provide a solution for those specific requirements. So basically **you're productizing their requirements.**

Now, the cost is higher because to build their requirements into more of a product you have to make it more flexible. **You have to make it work for everyone.** You have to put in more platforms, structure and everything like that so the product piece becomes more expensive to build as opposed to just customizing it for one customer, but it **really does make the whole thing win**.

Then **the way you get them to pay for that** is you go back to them and say, "right, we've got these features on our road map, they're down here next year sometime, whatever, this is what we plan to do and this is how we can provide a solution to your requirement but for us to move it forward, so that you get it much, much sooner **we have this expedite process where basically we can charge you a fee** for moving it towards the product road map, towards the beginning of the product road map".

That way they get the features early, which is when they actually want them and you get your development funded, which is what you want and also, you're able to stay aligned with your core SaaS product or whatever product it is you're building, but the point is **you want to stay on one line of code**, or one branch of code as we call it in the industry.

So to sum up, **say no to customizations**, productize any customization request or any feature request that you get. **Build it into the core** and make sure you

charge the customer to move it forward, to move it up your road map so you get that cash, you get that revenue coming in which helps fund the rest of your road map and that makes that a win-win for everyone.

This is Paul Clifford, leave a comment below, let me know what you think about that and I'll speak to you soon.

Are you ready to become a software entrepreneur?

Get everything you need at www.disruptware.com/.