

Bursting the customer success bubble, balancing customer and product-led success

It is generally accepted in SaaS that customer adoption and retention can be driven through 'Customer Success' (CS) functions: those that improve a customer's experience of a product to both benefit the user and bolster the long term prospects of the business.

Temps de lecture : minute

31 July 2023

At Notion Capital however, we believe that whilst this function is valuable, there is perhaps a tendency to spend excessive amounts on CS when you consider that its application and associated benefits are not scalable. In this instance, CS often becomes a band-aid solution for products that are not yet intuitive enough for consumers to confidently adopt.

Instead, we believe that companies would be better served by putting their product first, and people secondary to solve customers' needs. How your team iterates a product in response to pain points can be incorporated into a work-flow that reduces the need for a middle-man and enables scalable growth and future-proofing.

The need for Customer Success teams

Customer Success has become a huge industry, originating at cloud giant, Salesforce. At the time they were facing an acute pain point in churn, experiencing up to 8% per month which is a big risk to business. The solution, as is often the case with acute pain points, was to throw people

at the problem to address the churn, and the CS industry was born. Nowadays, the typical investment SaaS cloud businesses make into CS is around 10% of revenue, a really substantial investment which does compress important KPIs like gross margins. Our assessment is that the cost is probably worth the outcome; CS does address churn issues and support growth.

But is CS efficient?

One argument is that Customer Success is not scalable, because with more customers, you need more CS agents. As technologists, our instinct is to use technology to address issues of scalability. When considering Salesforce's solution to the problem of churn, one thought experiment leads us to go back in time and question, what if instead they'd asked the product lead to interrogate the software and consider whether usage and adoption might be their problem, something that could be resolved at product level. Should we instead look at the product to solve customer pain points, drive CS and be more accountable to product issues, in a way that is scalable and adds customers?

Another angle for success

Where we've landed at *Notion* is the concept of Product-led Success (PLS), where product teams take more ownership and accountability for driving customer success. We're certainly not arguing for the removal of CS, because there are valuable activities that people carry out uniquely well. But Product-led Success results in bigger investments into product; issues being tackled directly by the product team, which in turn allows you to unlock more potential in the CS function because they are concentrating on the activities they do best. PLS also saves cost, because it reduces the need for an army of people, who can instead focus on fewer, higher impact activities.

There are a number of interesting case studies to take inspiration from. For example at Profitwell acquired by Paddle, one of our portfolio companies, they required product managers to do most of the first-line non-technical support, as they were the ones who were responsible for the customer experience. This created tight alignment between the product team and customer outcomes. Similarly at Snowflake, not a portfolio company, but a business we admire and have studied, does not have a CS function per se. Many of the activities typically found in CS are distributed throughout the organisation. Most importantly it puts tremendous effort into solving issues for customers in ways that are scalable, making sure customers and colleagues have the information they need at their fingertips to act quickly, and to try and minimise complexity and friction for the customer wherever possible.

Finding a balance

The reason that CS exists at all is that tech companies are solving complex problems, therefore their solutions are complex. They are frequently feature rich, difficult to use, and may require a change of behaviour in their users including the need for training. It's a complex world in which CS is essential to help navigate complexity and teach their users. So whilst the product team are developing really feature-rich, capable products, customers are actually only using a small portion of it. They're not getting full value out of the product they're paying for, which frequently results in churn.

PLS should focus minds to create products that are intuitive, easy to use, that expose capabilities over time which are matched to a user's learning curve, and deal with the issue of consumption more effectively through the product. This requires an organisation to develop a product mindset in which product teams are empowered to drive revenues. With smarter design and better UX, you can start to deal with one of the key challenges which drives CS: if you fix the product, you don't need people to step in.

CS functions will continue to play an important role, but they should focus on the things which people do uniquely well such as identifying and tracking customer outcomes and coordinating across large complex customer organisations. But by placing greater accountability on the product function, and addressing root cause issues in the product, we believe software businesses are better placed to drive better outcomes more efficiently.

Itxaso Del Palacio, General Partner and Andreas Panayiotou Director of Pricing & Monetisation at *Notion Capital*.

Article by Itxaso Del Palacio & Andreas Panayiotou