Scrutinize

Give a problem with ambiguity

At Apsona, we made a major shift from product-based pricing to bundled pricing, which created a lot of ambiguity around how existing SMB customers would react. Leadership had decided to keep existing customers on the old model, but there was no clear strategy for upselling without risking trust.

I took ownership of figuring out how we should approach this from the customer side. Coming from a non-sales background, I knew I needed to be very intentional. Rather than pushing a blanket upsell, I started by segmenting our SMB customers based on usage patterns and license behavior to understand where real value already existed.

My reasoning was that if customers weren't already seeing strong value, forcing a pricing conversation would damage long-term relationships. So I focused on identifying power users and customers already close to the bundled threshold, and worked with the purchasing team to frame the bundle as increased value rather than higher cost.

Found and opportunity

We experimented with targeted messaging and tailored pitches, and over time we successfully moved a meaningful number of accounts onto the bundled model without negative customer fallout. More importantly, it taught me how to navigate high-stakes ambiguity by balancing business goals with customer trust — something I now apply whenever I'm operating without a clear playbook.

While rolling out a pricing update at Apsona, I was responsible for updating our Salesforce AppExchange listing. As I went through it, I realized something felt off — the way we were positioning our product and value proposition didn't match how customers were actually using it, and the listing hadn't been meaningfully updated in quite some time.

I saw this as an opportunity because AppExchange is a high-intent channel, and misaligned messaging there doesn't just hurt conversion — it attracts the wrong leads. Since this wasn't part of my original scope, I first validated the impact with my manager and then took the initiative to research how AppExchange discoverability and ranking actually work.

Based on that, I put together a proposal for leadership outlining gaps in messaging, visuals, and testing strategy, along with how we could measure improvement through partner analytics. Once approved, I worked cross-functionally to update the listing, refine messaging, and improve the overall presentation.

Over the following months, we saw a significant increase in inbound leads and better alignment with customers who were genuinely interested in our product. Increased product discovery. More importantly, it reinforced how I approach opportunities — when I see a leverage point that affects growth or customer quality, I'm proactive about owning it, even if it falls outside my defined role.

Tell about the time you failed

In my previous role, I worked closely with enterprise customers to understand their needs and translate them into product improvements. During a sales enablement session, we identified a customer with nearly 800 licenses who needed different products assigned to different users.

At the time, license assignment was a highly manual process. While there was an internal workaround where developers could bulk-assign licenses using organization credentials, it wasn't scalable or customer-facing.

I identified this as a potential product opportunity and took ownership of proposing a bulk license assignment tool. The goal was to reduce operational effort and support anticipated growth in enterprise customers with complex licensing needs.

I drafted a product proposal outlining why a bulk licensing feature could be valuable and aligned with future customer growth. The idea received buy-in from the product team and leadership, including the CTO, and I led the initiative through development.

However, I made two critical mistakes:

- I **underestimated development complexity**, assuming it could be delivered in two weeks.
- I **did not validate the business impact rigorously enough** — I relied on anecdotal customer cases rather than concrete data around frequency, revenue impact, or time savings.

As development progressed, edge cases and technical dependencies surfaced, extending the project to over a month. Testing and bug fixes further delayed release.

he feature did not meaningfully increase revenue, reduce time spent, or unlock new sales, and it introduced ongoing maintenance overhead due to bugs.

In retrospect, this represented a poor return on engineering investment and a missed opportunity to allocate resources to higher-impact initiatives

I took full responsibility for this outcome. While the idea had leadership support, the failure was in my judgment and planning.

This experience fundamentally changed how I evaluate product ideas. I now pressure-test every initiative against three questions:

1. Will this drive revenue?
2. Will it meaningfully reduce time or operational effort?
3. Will it unlock new customer adoption or sales?

If the answer is no to all three, it doesn't justify the build. Since then, I've been much more disciplined about validating impact before advocating for development work.

how you broke a messy problem into parts

how you identified constraints

how you aligned people with different incentives

"How do you decide when you *don't* know?"

"Do you freeze or move?"

"Can you explain your assumptions?"

I was asked to implement a way to track CSAT and map responses back to specific customer support cases, but I had very limited information upfront. There was no defined budget, and while leadership said they were open to paying, there was no clarity on how much or whether this would be a long-term investment.

Given that ambiguity, I first clarified the non-negotiable constraint — responses had to be reliably tied to individual cases. From there, I evaluated a range of options across paid and unpaid tools. Rather than making assumptions about cost alone, I focused on three criteria: functional fit, speed of implementation, and reversibility if the approach didn't scale.

Based on that, I recommended starting with SurveyForce, an open-source tool that met the core requirement without locking us into a high upfront cost. I presented this as a deliberate first step rather than a permanent decision, and aligned with stakeholders before moving forward.

That approach allowed us to move forward despite incomplete information, meet the immediate need, and preserve flexibility to invest further once we had more clarity.