

CUSTOMER ENGAGEMENT: THE CASE OF THE MISSING FIANCÉ



"A developer must be engaged in the customer experience." I heard that statement in a company town hall meeting once and I could not help but get stuck on the three main words: *developer, engaged, customer!*

Over lunch that day, I created a funny visual in my head about a developer actually being engaged to marry a customer. And I thought, "Wow it must be an old-timey arranged-marriage type scenario!" My Indian grandparents often told us stories about a time when it was entirely possible to be engaged to someone and yet have no contact with them!

And I thought, "How true.... This is perhaps the culture some teams have, expecting the developer to be engaged with the customer, but with hardly any forums to establish direct contact! Sometimes not even during sprint review!"

Then I started to wonder, how often is the developer having a true interaction with the customer? Are they able to ask questions about the intended use of the feature that is currently being developed, able to seek feedback on a previously delivered feature, able to gauge whether what is being developed is truly what the customer needs to satisfy their needs?

While it began as nothing more than just an amusing thought, I have often wondered about the value of establishing direct contact between the developer and the customer. Is such contact beneficial or detrimental?

Just like my old-timey couple, why is there always a layer of product management chaperoning the happy couple? How about some alone time?

Direct contact can provide many benefits:

1. **Nothing will be lost in translation.** In fact, nothing is "translated" at all! The developer gets to hear exactly what the customer says.
2. **A chance to help the customer understand the product better.** There is always a chance that a particular customer representative is not fully aware of the complete set of features/capabilities of the product. Perhaps the customer has been through some organizational change and the new set of people in charge are not as familiar with the product as the initial group. Direct contact with the developer offers an opportunity to strengthen the customer's awareness of the product and its features.
3. **A chance to understand the customer psyche.** Are they asking for a particular feature simply because they were used to it being in their former product of choice? Teams could potentially help educate them on other features that deliver similar results.
4. **Realtime customer feedback!** The main reason we do things the way we do! That precious feedback! Fresh and firsthand! And it could help develop a relationship which could help future communications.

Of course, there are some potential risks to consider:

1. **Is the Team 'polished' enough for this interaction?** When meeting a customer, we are essentially representing our entire organization. Will the team be capable of maintaining allegiance to the company's vision?
2. **Does the team understand the language of presentations?** Some presentations are made focused on what the future could bring, some are focused on what is currently available, some could be, as I like to say, "lightly kissed by Sales & Marketing!" An engineer might say, "this will kill you," while the sales officer may say, "this will cause some discomfort."
3. **Does the team understand that these meetings might not be the best place to make promises about when/which features can be delivered?** Also, that they're not the best place to share frustrations they themselves might be having about the company's current managerial/development styles?
4. **Is the team on board with presenting a "united" front to the customer?** Or is someone likely to roll their eyes when another person promises a picture slightly rosier than they would themselves prefer?

Each of these potential risks underscores the need for preparation before meeting with customers. Establishing a unified message and consistent tone will help portray your team and your company in the best light and avoid confusing the customer with mixed signals.

Product Management could act as a much-needed buffer between the hardworking engineering team and the always-demanding customer. They can save the team's time by distilling the essence of long meetings and presenting the team with a concise set of requirements that can go straight to refinement. The product manager can set the overall vision and the [product owner](#) can help fulfill that vision by helping the team remain focused on what is truly the core deliverable.

In conclusion, both team and customer benefit by direct interaction, even when facilitated by product management. This interaction makes for happy engagement and, in the long term, a healthy relationship.