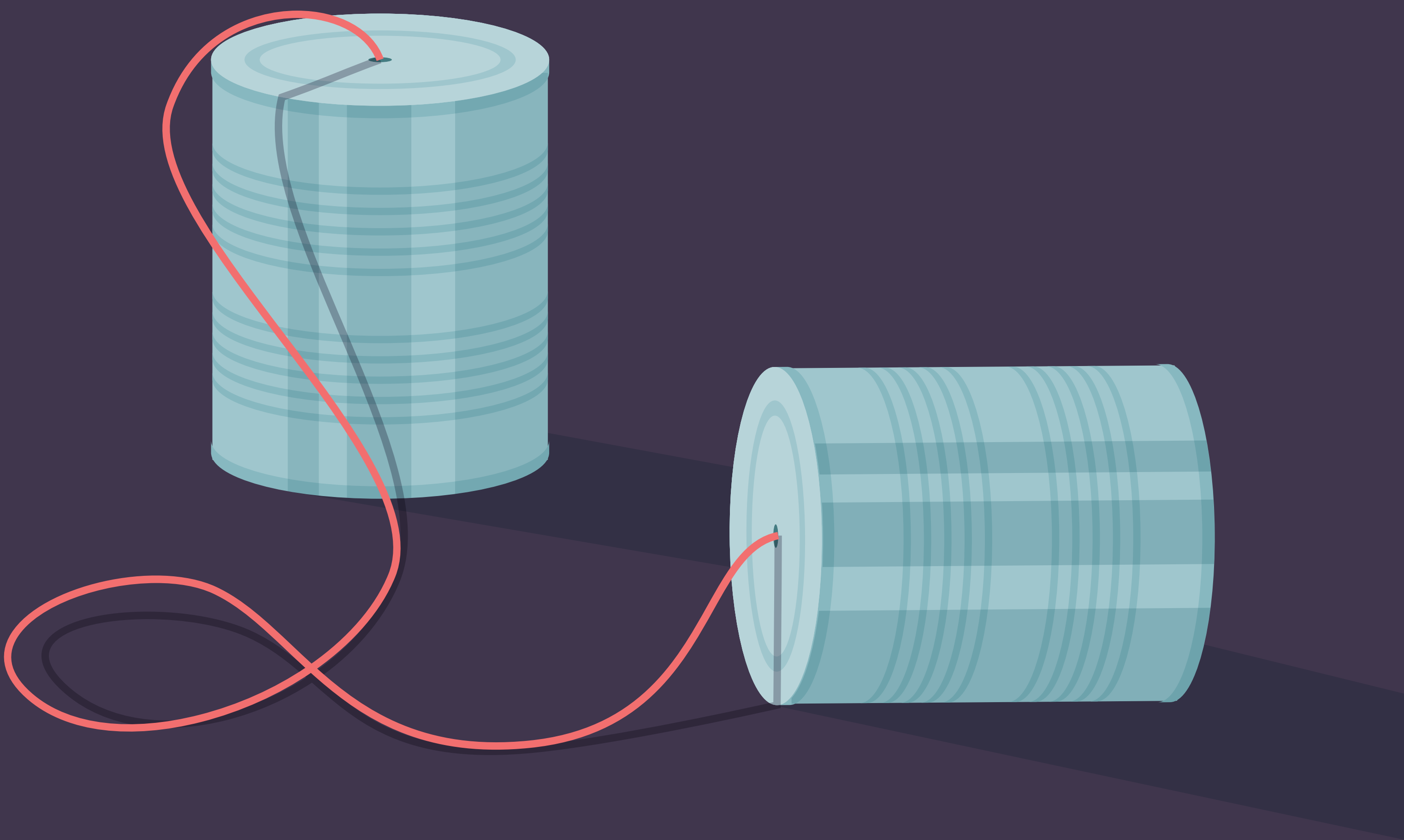
The Ultimate Guide to Communicating Product Feedback



A handy e-book for product-driven support teams

CONTENTS

Intro: Get into the Mindset of a Product Manager

Product Managers have a lot of demands on them: develop new features, be innovative, fix and improve existing features and do it all with limited development resources available. For every feature suggestion that gets integrated into the product, there's five more that won't be.

Support Managers have dramatically different metrics to focus on. It's easy to get into a transactional state of mind, thinking about each 'ticket' or customer individually, reacting to incoming feedback with empathy and concern for the customer's experience. Often times, this focus directly contradicts the business priorities that the product manager focuses on. This means sending feedback and bug reports across to product management can feel like you're banging your head against a brick wall.

It's important to realize and overcome the differences in priorities. If your organization isn't taking feedback from the customer support team, the frontline team becomes a shield instead of a filter. Support agents will feel ineffective and they are more likely to experience job fatigue and burnout if they aren't able to effect change. Product isn't as informed as they could be and the needs of the customer get left out of the roadmap.

As a product manager, you're constantly having to consider how feedback fits in with the bigger picture. From the highest level, considering how feedback affects the product vision and where you've said your product is going, right down to the detail, thinking about how feedback fits in with what's already in the pipeline for development or in short term plans.

Anything we build needs to be considered holistically: If we build it, will it make the product more complicated for someone else? Will we need to maintain it, support it, sell it and market it?

If we just built what customers asked for with no further consideration, we'd end up with just a pile of features... which doesn't make for a compelling, useful product.

Janna Bastow

Co-founder, ProdPad

80% of Product Managers feel feedback from customer teams are important, but only 15% of them have an effective feedback process in place.

uservoice

By painting a compelling picture for your Product team, your Support team can push through customer-driven feedback more effectively. This guide will show you how to:

- Create a business case for change
- Humanize customer needs
- Provide organizational context
- Format feedback packages more impactfully
- Create alignment between Support and Product

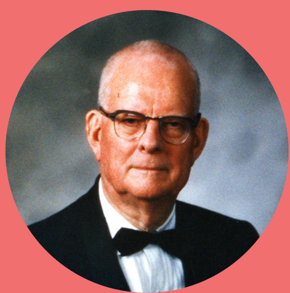
1. Creating a Business Case

Product roadmaps are driven by money: what keeps money (retention) and what makes money (acquisition).

It is easy for a sales team to impact the roadmap in the context of acquisition, since it is relatively easy to connect the dots between a feature that didn't exist and a number of dollars missed.

It is somewhat more difficult for the Support team to connect the dots between missed requests and un-actioned feedback and retention. Possibly because the value of retention to a business is the sum of subtle inputs that happens over a longer time frame. Fortunately, Support teams can also show the business value of requests.

"In God we trust. all others must bring data"



W Edwards Deming

There’s often multiple ways a feature request impacts the bottom line:

- Happier customers, who stick around for longer
- A reduced cost of support from fewer questions
- Additional sales revenue from new customers

Let’s break these down individually to paint a picture of feedback no Product Manager could refuse. Here are a few tricks that you can use to create a business case that will have your Product Manager begging for more.

Happier Customers

If we don’t make this change, will customers be more likely to cancel? How many customers are “at risk” of cancelling? For our purposes, let’s consider any customers who complain are increasingly more likely to churn. Secondly, it’s well established that for every customer that does contact you, there’s many more that don’t.

You can either use the industry standard of 1:25 (TARP 1999), or survey customers to determine your “non-contact” rate (see side panel). This will show the revenue impact not implementing the suggested functionality will have. This process works for bugs or feature requests.

Customers who have complained or made a request ____________________

Multiply by non-contact rate ____________________

Multiply by LTV, or add total contract values ____________________

Total anticipated revenue at risk ____________________

If you have had customers that churned specifically due to this issue, note them separately as “realised lost revenue”.

To back up this number, you can also include supplementary data:

Satisfaction of tickets related to this issue

- *if the CSAT is low, it’s a good indicator that customers are unhappy.*

Feedback forum votes

- *If you allow users to publicly vote up feature requests, include the total number of votes and comments customers have left for this request.*

Visits to support articles

- *If the most searched article is “How to Login” - we should probably make it easier for users to find.*

Non-contact Rate:

Some companies are better than others at soliciting feedback from their customers. The standard ratio is that for every 1 customer that contacts you about an issue, 25 won’t have.

You can calculate your own non-contact rate by randomly surveying a few customers with the following simple questions.

“Have you encountered a problem with X company?”
”Did you contact support about the issue?”

If the answer is no, definitely reach out to see if you can still help!

Your non-contact rate is:

Customers who had a problem and reached out : Customers who had a problem and did not

Reduced Support Costs

How much time does your team spend dealing with requests for this piece of feedback? Your team's time is the most expensive cost of running a business - so make sure you include this in your calculations. You may need to do the calculations stated below for each of the channels you support (ie. phone complaints are most costly than email complaints).

Number of tickets received X ____________________

Average time to resolve tickets ____________________

Total amount of time spent on tickets X ____________________

Average hourly cost of support ____________________

Total cost of support ____________________

“ Get data informed.

When support is seen as a separate part of your business, there isn't much of a reason for tracking granular data too closely. If you know total volume, and how fast it is growing, then you can predict how many team members you need. But if support conversation data is driving things like bug fix priorities, updating of existing features, and tracking of previous release goals, then clean and consistent support tracking data is critical.

Get deep on customer problems.

I wish I had 40 hours per week to talk to our customers and understand their issues more deeply. Unfortunately, I don't. But our support team has almost 360 hours per week to do this! If they are getting a better understanding of how we can align with our customers interests, then they can help me view our product strategy through that lens. Extremely helpful! ”

Jeff Vincent

Head of Product Management, Wistia

2. Humanize Feedback

The biggest benefit of passing feedback through from frontline teams is that it's straight from the horse's mouth. You can include great soundbites or quotes to really show the pain customers are experiencing. This kind of unfiltered emotion can, when backed up with data and business, capture the imagination of the organization.

Use the Words of Your Customers

Rather than trying to mince words and decipher what customers want, dig deeper and use their own words to support the feedback. Don't try and make requests easier to understand or better fit the picture you're trying to paint. Take direct snippets of emails and recordings of phone calls.

By using the words of the customer, it's easier to really walk a mile in their shoes and gain empathy for their situation. Also, those !!!!! matter. Let the frustration in the customer's voice shine through - it might just make the feedback more compelling for your Product Manager.

If a customer is willing to share their thoughts and speak with the product team, it can be helpful to facilitate that conversation. In your feedback package, include the contact information and background details of a customer that has interesting feedback or use cases.

Develop Storytime

use case: a series of related interactions between a user and your product that enables the user to achieve a goal.

Break down customer actions into a story of the steps they are taking and what they hope to accomplish. Instead of repeating the customer's stated request (for example, 'The Customer Wants to Sort by Date"),you can frame the request as a story.

As a <type of user>, I want to do <something> so that <some goal>.

Example: As a power user, I can specify files or folders to backup based on file size, date created and date modified so that I can save time and storage space.

Context is king, and by providing more information about the customer and what they want to accomplish, you can make the feedback more compelling.

"Product Managers think (and likely dream) in use-cases. When reviewing feedback, be sure to communicate what kind of customer the request is representing (not just the opportunity). The more specific the use-case, the better. Sharing real-world details helps the Product team understand the problem and maximizes the impact of your feedback."

Alex Grant

Senior Product Manager, Hootsuite

3. Paint a Bigger Picture

At the end of the day, everyone in your organisation should be working towards the same ultimate goal: attracting many loyal, happy customers. Anything that encourages progress towards this goal is valuable. Connecting small scale feedback to a bigger picture helps inspire action.

To drive this change, it’s crucial to showcase the why, instead of the what. As illustrated in Simon Sinek’s Start with Why, people don’t get excited about the what. They need a reason to inspire them: "It doesn't matter What you do, it matters Why you do it."

Whether your company has a set of values, or each department uses quarterly goals, you can better position customer feedback if you tie it back to the bigger picture.

“These “departments” are all accomplishing the same ends. They each exist to tell people about a Thing, get them excited about the Thing, give them the tools to use the Thing, and empower them to spread the word so more people use the Thing, too. They also exist to find out what people are saying about the Thing (be it good or bad) and tell the teams making the Thing so they they can fix, improve, and iterate.”

Elizabeth Tobey

Director of Support at tumblr

Align with Department Goals

Make a list of the top-line goals of each department and keep them on hand for communicating feedback. For example, if the Product team is tasked with increasing the number of active users per month, a request to fix broken email notifications for inactive users is directly connected to their success. If Finance is tasked with reducing refunds, resolving a confusing invoice system may make their lives easier, as well as reduce support requests.

If you can make an advocate in each department, it's easy to rally the troops around a needed change by pulling in stakeholders outside of the support team.

“Understand, align with, and inform product goals.
If CS folks are aware what Product is focused on, their feedback will always be timely and extremely informative. Conversely, if you think something else needs to be focused on, bring that feedback in that context: “Hey, I know we’re not working on this during Q2, but based on these numbers, I think [x feature] is really valuable to our customers, and isn’t working as it should."

Be open and available for participation.
I get that some cultures separate groups into silos. That sucks. But for more open cultures, make sure there is an ongoing conversation about how CS can participate in product development. While we don’t give out timelines for features, we do loop in CS and Documentation teammates as early as possible, so they can approach their conversations within that context. When things are closer to ready
- CS is the best QA team I know.”

Jeff Vincent

Head of Product Management, Wistia

Align with Company Values

At a higher level, product decisions are made based on the company mission statement and values. For example, one of Kayako's values is Think like a Freak - we want to be disruptive, experimental and challenge assumptions like it is our first day, every day. This company value frequently gets pulled in to support new feature requests and data experiments: if we you were looking at this with fresh eyes or starting it again, what would your solution look like?.

It goes back to all being on the same team. By providing bigger context to feedback, it reduces the anecdotal, emotional pull of customer feedback and reminds stakeholders of the end goals: happy, profitable customers and a long lasting business.

4. Choosing the best format

Chances are, you've stopped at the Product Manager's desk and had a five minute conversation about a small change to a workflow that would be helpful to customers and Support alike. But if that conversation isn't followed up in writing, the odds of it getting done are slim to none. Some Product teams like emails, some prefer presentations and others would love to see a spreadsheet. So what's the best format to raise feedback?

Surprise - you just need to ask them! While ad-hoc "off the side of the desk" conversations are never the best way to go, every Product Manager has a preferred format of receiving customer feedback. It starts with understanding how the product team prioritises and maintains backlog for product. By understanding this, you're able to choose the best format for feedback. Here's two we really like.

Email

The inbox has become the modern to-do list. Having all of the information in one place allows Product Managers to search and reference it as needed. Some things to think about when considering email as your form of correspondence:

Pros:	Cons:
emails can be done as needed, no set schedule required	context can get lost when there is no conversation
ensures all necessary information is included	email doesn't build as strong of a relationship as in person meetings
archives feedback so that nothing gets lost	emails can (unfortunately) get ignored.

Scheduled Structured Meeting

Note the “structured” modification in this title. It’s not enough to just meet with the Product team and chat. Instead, make the most of this meeting by being prepared.

- Bring suggestions and customer feedback insights backed up with data
- Ask questions to see what would be helpful information to dig up
- Learn what Product Managers are working on, so you can track helpful metrics and keep an eye out for requests
- Walk through use cases together

5. Communication doesn't end there

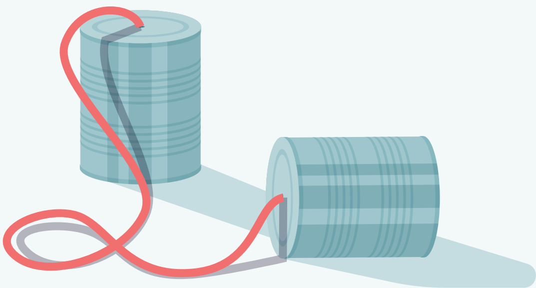
Congrats! You did a great job communicating the needs of the customers, and changes are going to be made. As it turns out, the work is not over. It's called a feedback loop for a reason.

Gathering all of this information at the start means that you have a set benchmark, or starting point. Once the feature is implemented, follow up with the Product team on the results.

Create a Business Case: If you can show fewer complaints or fewer hours spent answering these tickets, put a number on it. Product teams love hearing the ROI of their work!

Humanize Feedback: Developers often ship something and never get to hear the feedback on it. If customers write in, even with a short thanks or "I love it!" copy and paste the interaction into a document or email to share. Hearing direct praise from a happy customer can connect the Product team even closer to the customer experience!

Use the Right Format: Share the love company wide! If a product change resulted in a big win, it's great to spread the news and give credit where due. Not only will everyone want to hear about the changes, it celebrates the great teamwork of Support and Product - which can only lead to more collaboration in the future!



Appendix: The Feedback Template

Feedback: ____

____

____.

This request impacts ______*these customers*______ by ___________

______*intended effect*______.

Business Case

We've had ____*# of complaints*____ complaints about it, putting an estimated $____________ of revenue at risk from churn. Satisfaction scores around these tickets are low at______________. Dealing with this costs our support team $_________ every month.

We've had____*# of churned customers*______customers cancel or leave due to this request.

Other data to include:

- Feedback forum votes
- Visits to Knowledge Base Articles
- Searches in Knowledge Base
- Sales data
- NPS survey responses

Humanize it

Here’s some examples of customers asking for this change:

“ ____ ”

“ ____ ”

“ ____ ”

Use Cases/Stories:

As a <type of user>, I want to do <something> so that <some goal>.

If you’d like to hear more, these customers would be happy to share additional information about their needs:

John Smith - john@smith.com - 123-456-7890

Paint a Bigger Picture

This is important because: ________________________________

____

____

____

____

____

____.

Include the following:

- How it aligns with our values
- How it impacts the goals of product/engineering/sales
- How competitors are approaching this

"thanks for reading"