

PRODUCT
COLLECTIVE

INDUSTRY VIRTUAL ▶

NOTES

October 26, 2023



THANK YOU SPONSORS



Bloomfilter



×

GIGANTIC





TABLE OF CONTENTS

- 01 **BEYOND DISCOVERY: THE ART OF POTENT
CUSTOMER LISTENING FOR PRODUCT TEAMS**
Oji Udezue, Chief Product Officer, Typeform
Ezinne Udezue, Chief Product Officer, WP Engine
- 06 **LESSONS FROM 15,000 EXPERIMENTS**
Andres Glusman, CEO, DoWhatWorks
- 08 **BEST PRACTICES IN ENTERPRISE
PRODUCT MANAGEMENT**
Dan Olsen, Author, The Lean Product Playbook
- 12 **MOVING TO THE PRODUCT OPERATING MODEL**
Marty Cagan, Partner
Silicon Valley Product Group
- 17 **THE SUPERPOWER OF PLAY**
Diana Tobey, Senior Design Director, Play Lab IDEO
- 20 **BETTER TOGETHER: HOW COLLABORATION
BETWEEN PRODUCT AND ENGINEERING TEAMS
CAN FUEL GROWTH**
Shadi Rostami, Senior Executive VP of Engineering,
Amplitude

▼ **BEYOND DISCOVERY: THE ART OF POTENT CUSTOMER LISTENING FOR PRODUCT TEAMS**

Oji Udezue, Chief Product Officer, Typeform

Ezinne Udezue, Chief Product Officer, WP Engine

Session Overview:

Join product leaders Oji and Ezinne Udezue as they explore the untapped power of 'Customer Listening'—the ongoing art of tuning into real-time customer feedback from diverse channels beyond the initial discovery phase. Drawing from their experiences leading product at high-growth technology companies, and sprinkled with insights from their upcoming book, Oji and Ezinne will unveil strategies to harness continual customer chatter effectively, offering a fresh perspective on driving product growth in today's tech landscape.

Key Points:

- Customers are talking whether or not you want to listen.
- Listening is learning
- Listen, Synthesize, Act
- Make it easy, make it a habit
- Drive ownership across the organization
- Listening drives delight - you want your customers know you're paying attention, and acting on it.

Notes:

What is Customer Listening?

Customer listening is understanding and analyzing the feedback and sentiments of your customers to improve products, services, customer experience, and overall strategy.

Think of customer listening as the ongoing art and science of tuning into real-time customer feedback from diverse channels beyond the initial discovery phase.

Why does customer listening matter?

At every stage of the product life cycle, people are talking about your product.

Once you've found decent product market fit, people will chatter about your product in productive ways. If you can efficiently listen to what your customers are saying, you'll get great information that will help you improve your product.

- ▼ This is especially true for early stage products that are driven by feedback from your target customers. In fact, you could say that for early stage products, product strategy is entirely based on customer feedback.

If you're in B2B SaaS, your product is a key Go To Market (GTM) asset. The customer journey, onboarding, even support are all often accounted for in your product. To help your product evolve as a GTM asset, you need to listen to your customers.

Listening is learning, and it's one of the most important things you do as a product manager.

Contrasting Customer Discovery and Customer Learning

Customer discovery is forward looking and helps you learn more about strategic opportunities. It is often mapped against known problems and drives Product-Market Fit.

You use customer discovery when you have a tactical or strategic problem space you want to map and understand to improve its workflows.

Customer listening is diagnostic of your existing solution. When you listen to people share what they feel about your product, it exposes opportunities for improvement. The information you get is high resolution, and it drives customer delight.

In order to have effective customer listening, you need to find a high signal-to-noise ratio and process the information you get quickly.

Things to consider when customer listening:

Hearing what is not said

A quote from Peter Drucker embodies the Art of Product management: "The most important thing in communication is hearing what isn't said."

Listening goes beyond what your customers say. You can't take customer feedback word for word. Listen for the needs behind the words.

That means look for the patterns that indicate real needs. Those needs can exist at multiple levels from the entire product down to a single feature. Needs can also exist across multiple gaps and segments.

As You Listen

Remember that customer listening is something anyone can do. To be successful consider the following things:

Identify core needs as you listen. Try to find out what your customers are really trying to say.



Look for customer sentiment. How your customers feel about your product and company can be an indicator of problems down the road. Listen for the gaps that customers identify - things missing from your product.

Chatter about your product can be high resolution. You'll hear sentiment about your entire product and you'll also hear sentiment about a feature you just released. You'll have to decide how you want to deal with that varying level of detail.

As you scale your customer listening efforts, velocity becomes important. As you scale, your listening channels expand and can have high volume. You'll have to figure out a way to process all that information or resort to sampling. The good news is with that increased volume, you're able to react quicker if you're prepared. You can ship something one day, and the high volume of feedback provides you with sufficient information to react the next day.

Feedback can be noisy so you need to synthesize the customer chatter to get a high signal-to-noise ratio. This helps you pull the useful insights out from the information that may not be as helpful.

What to Listen To:

Most software companies have two kinds of channels: direct channels and proxy channels.

Direct Channels

Direct channels allow you to hear directly from customers. Some examples of these channels are:

- Churn feedback - why people stopped using your product
- NPS Verbatims - when you ask people for an NPS score, ask why they gave you that score. That information can be helpful.
- Reviews from places like G2, the App store, or other markets where your software is available.
- Social Media Chatter - what people are saying about your product on the socials such as X (the network formerly known as Twitter), Facebook, or TikTok
- Community - Bring frequent users and early adopters together in a virtual community to get an insight into their view of your product's good/bad points
- Bug Reports - good for companies whose customers are developers.

Proxy Channels:

Proxy channels filter customer feedback through other teams such as customer support, sales engineers, account managers, or partners if you have a platform.



Keep in mind that feedback coming from proxy channels is synthesized and interpreted by the people you're getting the information from. Sample some calls people in these channels are having with clients so you can get a sense of how they interpret the feedback they hear. You don't have to do 10 channels at once. Decide which channels of feedback are most helpful, and most important and focus on those as a starting point.

How to operationalize Customer Learning:

You can sum up the three core steps to customer learning as LISA:

Listen - What you're listening to

Synthesize - How you're synthesizing

Act - What action you want to take

Here's a bit more detail behind that:

Collect

Harvest data from your sources to a centralized system so that you can deal with it.

Clean

Extract a sample that is relevant and actionable. You need to do this to bump up the signal-to-noise ratio. For example, if you use Gong, set it up so that it identifies calls with certain words (i.e. "let me take that to product") which will send links to those calls to Slack.

Weight

Order by the impact the information has on your users. Decide which feedback to pay more attention to. Pick the sources that are more reliable, useful, and trustworthy.

For example, if you have an overbearing customer, first realize that the fact there is a loud voice is a signal of something. Then remember that successful companies have more customers in their future than in their past. With that in mind, decide whether you should listen to your past at the cost of the future.

Listen to the people who represent your future, which are typically people who fit your Ideal Customer Profile (ICP).

Triage

Regularly process feedback to get down to the key info. Bring feedback in and categorize it into different categories (positive, negative) then a team could monitor and deal with it accordingly. Prioritize actions based on capacity. You may find it helpful to decentralize triage so that it doesn't all fall on one person or team.

- ▼ Another approach is to schedule specific triage discussions (meetings) to take specific actions.

Assign

Decide who is going to address the feedback and signals you're getting. Give actions to members of your team.

Report

Gather feedback reactions to the changes.

New Opportunities with listening:

As technology evolves, there are several new opportunities to make listening better, easier and more effective.

Leverage AI - analyze sentiment more effectively

Segmentation - A more detailed audience picture

Localization - Listen in all your markets consistently

Connect qual to quant - Marry up chatter with analytics. It's becoming possible to tie together information about what customers are doing (quantitative) with why they did it (qualitative).

Engage at scale - Treat users as individuals, personalize responses. We've been listening, but have not been replying, but there are capabilities coming.

For more information:

Visit Oji and Ezinne's website [Productmind.co](https://productmind.co)

Find out more about their new book about Product Management for product led growth (yet to be titled) at <http://howtoplg.com>

▼ LESSONS FROM 15,000 EXPERIMENTS

Andres Glusman, CEO, DoWhatWorks

Session Overview:

It's hard to believe but 80% of tests do not move the needle. So how do you set yourself up to beat the odds and drive growth? Andres Glusman will share insights gleaned from analyzing over 15,000 split tests from the world's top brands to help you confidently run experiments. Andres is the CEO and Cofounder of DoWhatWorks and was formerly led Product and Growth at Meetup.

Key Points:

Only 10 - 20% of AB tests provide results that move the needle.

To make the best use of experiments when your win rate is so low, you need to:

- Set expectations and goals thoughtfully.
- Measure speed to win, not speed to launch.
- Gather data before you test to beat the 20% benchmark.

Before you use AI with your tests, gather data on what works.

Notes:

The problem with A/B Testing

50% of people actually are above average, but 80% of people think they are above average. This indicates that people are bad at estimating. This mindset extends to when people try to optimize their website they overestimate their effectiveness.

In fact most people just guess.

Rigours Product Managers run A/B experiments, unfortunately the percentage of experiments that actually move the needle is between 11 - 20%.

That means that growth teams waste most of their AB testing time and effort. They keep doing it because when they do get it right, the results are worth the effort.

So if you know that only 1 in 5 (at best) of your experiments produce results, what do you do? You could ramp up volume to overcome the low win rate, Of course you can only run so many experiments.

The hard part is not getting a test out the door (except for some teams), it's getting enough data to get statistically significant results. Especially if you're in B2B SaaS.

▼ So you need to work on your win rate.

You also need to gather data on what works. In other words, talk to other companies who are doing experiments and find out what they're trying and what worked.

3 Lessons You Can Use Today

Lesson 1: Set goals & expectations thoughtfully.

When you decide to test, realize only 1 - 2 experiments out of 10 will win and set your expectations accordingly

That means you need to think about the timing of your tests and how many tests you're prepared to run.

Lesson 2: Velocity helps, but measure it correctly.

Speed to launch was the initial focus, but the better focus is speed to learn. Even better is speed to win.

Average speed to win is going to be 3 - 4 months. Think about how to bring that to smaller chunks.

Lesson 3: Beat the 20% benchmark

Gather data before you test

Add & subtract Sometimes the best optimizations are talking something away. If you're in the business of removing features you'll win more than when you add.

Context matters - it depends on where in the experience you are, customers you're talking to, industry you're in. There will be patterns, but there is no universal truth.

Final thoughts

The path of experimentation is a series of stages:

- Guess
- Test
- Velocity <- This is where the industry is at currently.
- Get data
- Know what works <- You should focus here.
- AI

You can't do AI without data on what works. Right now AI is garbage in, garbage out.

You should focus on knowing what works prior to trying AI.

▼ BEST PRACTICES IN ENTERPRISE PRODUCT MANAGEMENT

Dan Olsen, Author, The Lean Product Playbook

Session Overview:

Product management has experienced tremendous growth over the last several years, but much of the product advice available is from companies building products for consumers (Business-to-Consumer or B2C). There are important differences between creating B2C products and creating Enterprise products, also known as Business-to-Business or B2B. B2B2C companies face similar challenges.

Enterprise product managers and product teams face distinct challenges such as

- The loud voice of Sales influencing product priorities
- Important clients demanding specific solutions
- Struggles in applying an MVP (minimum viable product) mindset
- Having to meet the needs of both buyers and end users
- Inflexible product release deadlines
- Prioritizing product work and creating a product roadmap given all the above

Do one or more of those symptoms sound familiar? What is an Enterprise product team to do?

Come hear Product Management author, trainer, and consultant Dan Olsen share his advice on Best Practices in Enterprise Product Management.

Dan will share relevant advice from his bestselling book The Lean Product Playbook.

Key Points:

- Enterprise Product Management differs from B2C product management. The top 4 Product Management challenges specific to enterprise product and how to solve them are:
- Clients dictating specific solutions - start with problems before solutions.
- Sales influencing product priorities - be strategic: say now or not now.
- Inflexible product release deadlines - be realistic about scope and time tradeoffs
- Struggling to apply an MVP mindset - prioritize the puppies

Notes:

Most materials about product management assume a B2C context. Building products in a B2B setting, also known as enterprises, have some key differences.

Those differences are best described by a set of enterprise Product Management challenges.

▼ **Clients dictating specific solutions:**

It's quite common for B2B software companies to have a big customer say "we need you to build feature x for us." Those types of requests usually don't end well.

When this happens, your product team starts in the solution space instead of starting in the problem space.

Problem Space vs Solution Space:

The problem space defines a customer problem, need or benefit that the product should address. For example, "It's too hard to share photos with my friends".

The solution space defines a specific implementation, or design to address the customer needs. For example, a mockup for an app or building the actual app.

The idea is to separate the problem space from the solution space so that you can make sure you understand the problem before you create a solution.

One reason 80 - 90% of products fail is because people jumped into solution space without trying to figure out the problem first.

This happens because most of the product team lives in the solution space. Developers develop and designers design. It falls to Product Managers to define which means they should make sure that the team spends the appropriate time in the problem space.

When you're handed a solution to deliver, use 5 why's technique to bring discussion of solution back to the problem. To get that conversation started ask "what problem will that feature help us solve?"

Sales influencing product priorities:

In a B2C setting the main roles working on the product are Product, UX, and Developers. The team also has to work with business and marketing.

All of those folks are present in a B2B setting, but you also have Sales. Sales has a lot of influence because they're trying to sell your product.

Sales will always say your product has a feature when asked because they are incented to do that.

Unfortunately, saying "yes" to every request from customers and prospects is NOT strategic. You're not applying a strategic filter.

To address this, you need to apply a product strategy, which at its core means saying No to bad ideas and even some good ones that aren't directly aligned with your product strategy.

▼ Inflexible Product Release Deadlines

Product development has a “Iron triangle” of variables comprising scope, time, quality, and resources

If you try to lock down all four variables, things will get out of balance, so you need to flex at least one variable.

No one will cut quality. Resources are constrained in the short term since no one is just sitting around. Even if you found someone, they need time to get up to speed.

That means In the short run, the only levers you have are Time and Scope.

On top of all that, B2B organizations focus on the Magical Dates of B2B Software - which are the ends of the quarter. The end of quarter is magical because that's how teams have structured their roadmap.

This fixed date mindset is a terrible mix with the uncertainty inherent in software development.

So that leaves scope as the main lever you have, and an MVP Mindset will help you make the most out of the scope lever.

Struggling to apply an MVP Mindset:

Minimum Viable Product (MVP) is a controversial topic because there are so many perspectives of what a MVP is.

MVP can be describe along this scale:

- Something a customer can read to learn about the product (landing page)
- Something a customer can see what the product will look like (Mockups)
- Something a customer can interact with (Interactive prototype)
- Something a customer can use (Alpha or private beta product)
- Something a customer can buy (Publicly available product)

To help get agreement onto what people are talking about, Dan identified three different interpretations of MVP:

- MVP - Live working product
- MVP Prototype - representative of what product would be
- MVP Scope of Functionality - Thoughts/Documents - real purpose of MVP determine what is in and what is out.

At its core, MVP is a mindset. For every milestone, what is the minimum required functionality that is enough. In other words, what can we say “no” to?

If no sounds too harsh, say “not now” or “not for this release”.

- ▼ To get alignment about what's in scope, use a one page product roadmap that shows what features fit into particular versions lined up with your products key benefits.

Without the clarity that comes from this one page artifact, it's hard to get alignment.

Teams often make two mistakes with MVPs.

- Over scope MVP - This happens when influential people see that something was left out, and get irate about the feature being left out. So team gives in and includes it, thus increasing the scope and endangering the timeline. This is like throwing spaghetti at the wall and seeing what sticks.
- The non viable MVP - Team includes a subset of functionality that doesn't result in a delightful, usable, reliable solution. This is a symptom of "Just an MVP" being used as an excuse to cut corners.

To avoid these mistakes, you need to make sure you include necessary features in each version and only those features that result in a functional, reliable, usable, and delightful product.

▼ MOVING TO THE PRODUCT OPERATING MODEL

Marty Cagan, Partner, Silicon Valley Product Group

Session Overview:

With more new competitors than ever, with more new disruptive technologies than ever, and with continually increasing expectations from customers, most companies today know they need to learn to work like the best tech-powered product companies. But what does that really mean? Following some specific process? Moving to Agile? Implementing OKR's? Embracing Lean Startup techniques? In this talk we'll start by defining what the product operating model is, and what's involved in moving to this model, and most importantly, what can a company do once they've moved to this model, that they could never have done before?

Key Points:

The Product Operating Model describes how your company can change to be more like the best tech companies.

The Product Operating Model has 3 Product Dimensions: How you build, how you solve problems, and how you determine what problems to solve.

The Product Operating Model has 4 Competencies: Product Manager, Product Designer, Tech Lead, and Product Leadership.

The Product Operating Model has 5 Product Concepts: Product Strategy, Product Teams, Product Discovery, Product Delivery, and Product Culture.

Notes:

The Product Operating Model - is a new name for the difference between the best product companies and the rest.

The most common question Marty gets is "how can my company change to be more like "the best? How can they move to the product operating model?

What is the Product Operating Model?

An operating model is not a process. It's a conceptual model.

If you look at how strong product companies implement their product development, you see a common set of principles.

Tough to differentiate between product principles and good stuff to do.

The Product Operating Model comprises 3 dimensions, 4 competencies, 5 concepts and 20 principles.

▼ 3 Product Model Dimensions:

Dimension 1 - How You Build (continuous delivery)

How often does each product team release?

If you're not releasing every two weeks, you can't take care of your customers like you need to, and you can't do the rest of the dimensions.

Companies worked on this for several years ago as they adopted agile.

The key to this dimension is how you build, test, and deploy solutions in consistent, frequent, small release vehicles

Dimension 2 - How You Solve Problems (Product Discovery)

Two common types of teams are feature teams and empowered teams.

Feature teams implement solutions defined by others

Empowered teams are given a problem to solve. The team comes up with the best solution to solve the problem.

Empowered teams building the skills to do continuous discovery: try out ideas, experiment, find a solution that addresses four risks (viable, valuable, usable, feasible)

It's easy to take a feature request and have a discussion to reverse engineer the root problem to solve and the measure to success. If you can do this, you can get to an outcome-based roadmap.

The next question is are those the most important problems to solve?

A company is going to live/die based on the choices it makes. Problems it wants to solve; challenges it chooses to address.

Dimension 3 - How You Decide Which Problems to Solve (Product Strategy)

Feature team companies don't have a product strategy. Rather, development staff are allocated to different business units. This is sometimes referred to as "peanut butter" product strategy.

Best product companies are good at deciding the few things that will make a difference to a company.

This is the most impactful dimension of the product model.

4 Competencies (roles):

There are 4 Competencies that will be new for companies moving to the product operating model. The biggest reason for transformation failure is not taking these competencies seriously.



Product Manager

The Product Manager is responsible for the value and viability of what the product team builds, and is ultimately accountable for the results of the product team. Product Managers make sure what the team builds is worth building.

A Product Manager in feature teams is usually closer to project management.

A Product Manager in an empowered team focuses on solving problems.

The product owner is a subset of the overall Product Manager job related to delivery. Being a true Product Manager is a hard job.

Learn enough about the rest of the business and related constraints to tell when something you build is viable for the entire business.

You also need to know about the customer so you can determine if your product is valuable. That means the customer believes the thing you built is sufficiently better than what they have that they will buy it.

Behind every great product, you'll find someone that knows the customers, the data, the business, and the industry, working tirelessly to create value for the customer and for the business.

If you've done your job in product discovery, backlog refinement is fairly easy

If you're in a feature team company and want to move to a product operating model adopt the skills to know customer, data, the business, and the industry trying to create value for customers and business

Product Designer

The Product Designer is responsible making sure whatever is built is usable, and is concerned about how customers experience the value of the product.

Design is not just what it looks like and feels like; design is how the product works. This role focuses on the entire experience (not just user interface)

In a feature team, you most likely won't see a designer so the product manager may act as a designer also.

Ideally, your team has both a strong product manager and a strong product designers. If you try to do both yourself, you're signing up for a lot of work.

Engineers

The tech lead ensures the feasibility of what the team builds, and has overall responsibility for delivery.

- ▼ If a company has all of its engineers as contractors, it's probably a sign that the company doesn't understand the role of engineers.

Engineers need to engage in discovery work with product designer and product manager. They need to be in the room where it happens. Nothing is more important than an empowered engineer. Engineers are not just there to code. They're there to come up with the right solution.

Product Leadership

Product Leaders are the manager of Product Managers, manager of Product Designers, and manager of Engineers.

Product Leaders enable and empower the product teams with coaching and strategy.

The product operating model depends on product leadership. Product Leaders address the challenges that Andy Grove pointed out "What gets in the way of good work? There are only two possibilities: The first is that people don't know how to do good work. The second is that they know how, but they aren't motivated."

Empowered product teams need to have a strategic context in which they make informed product decisions. Netflix understood this when they established the principle of Lead with context, not control.

5 Product Concepts:

Product Strategy

Product Strategy is the most impactful concept to a company. This is where you decide which problems to go do discovery on.

To create a proper product strategy you need to Immerse yourself in insights from:

- Customers
- Data
- Technology - things that are just now possible.
- Industry - what's going on in your particular part of the world that you can use to help solve customer problems

Product sense - recognition that you've immersed yourself in insights.

Product Teams

- Small
- Durable
- Cross Functional
- Empowered with problems to solve
- Accountable with results

The nature of how people work together is important. Innovation relies on Product Manager, Product Designer, and Tech Lead working to solve a problem together.

- ▼ Product, Designer and Tech Lead bring different skill sets to the table. All those skill sets are necessary,

The PM is looking at implications for marketing, sales, finance, etc.

Product Discovery

Be able to quickly and inexpensively determine whether a product idea is worth building

Principle: assess product risks.

Product Delivery

If it hurts, do it more often.

Deliver at least once every two weeks.

Principle: small, frequent, uncoupled releases

Continuous deployment is the key to quality and velocity - best way to get your product in the hands of your customers.

Companies should instrument your product (telemetry). Product model depends on collecting data about your product.

Product Culture

See Jeff Bezos - fail early and iterate until you get it right.

Principle Learning over failure

What does transformation really mean?

Transformation means changing how you build, how you solve problems, and how you decide which problems to solve.

If you want to compete against the strong product companies, you have to do transformation as described above.

▼ THE SUPERPOWER OF PLAY

Diana Tobey, Senior Design Director, Play Lab IDEO

Session Overview:

Play is a superpower that empowers us to redesign the world. Learn why play matters, the mindsets and behaviors that create the conditions for play, and ideas for bringing more play into product teams to fuel creative thinking.

Key Points:

Play is innate, work is a learned behavior. Play matters because it prepares you to face the unexpected in an unpredictable world.

The opposite of play isn't work, it's boredom. If your team is stuck in a rut, change the rules of the game.

Six techniques to unleash the superpower of play.

For Product Strategy - Persona Tarot Cards, System Board Game

For Feature Development - Bad Reviews and AI Roleplay (and Props)

For Team Rituals - Creative Workouts and LEGO Retrospective

Notes:

If necessity is the mother of invention, play is its father.

Why Play Matters

Play is core to the human experience. It's also core to the experience of many animals. Why has play survived millions of years of evolution?

Rewards of play can be the difference between life and death. Brown bear cubs who play more in their first year have a better chance of survival in every subsequent winter..

Play helps survival because it is the key to lifelong learning.

Play prepares the player to face the unexpected in an unpredictable world.

Sticking to the one right answer mentality puts us at a disadvantage. Play opens your mind to take chances. Need more play to face an unpredictable world.

Creating the conditions for play

The power of play - tweaking the rules of play drives people to change behavior.

What are the rules of the game that we're playing and when does it make sense to change those rules?

▼ There are rules you should follow and rules you should break.

You should follow social rules governing how you play together. Physical and psychological safety.

It's ok to break the rules of the game you are playing. Rules of the game themselves are things that should be malleable. Example house rules in UNO,

There is a lot of value in identifying constraints which don't need to be constraints.

What are the stated rules of product management if it were a game? What you're trying to get to (outcome)?

Unstated rules: there isn't a lot written about unstated rules. What is the meta game that you and your team are playing?

Are there unstated rules of engagement that might hold your product team back? How can you be better equipped to handle a changing world?

The opposite of play isn't work; it's boredom.

If your product teams feel like they're stuck in a rut, consider changing the rules of the game.

Unleashing the superpower of play:

When you talk about incorporating play into your product, the first word that may come to your mind is "Gamification" this is not the end all be all of bring play into product

Disney introduced leaderboards - led to increased incidents, culture became competitive.

Possibilities of play example: Glitch - collaborative multi-player game. Game centered on players being able to chat with each other. A couple of years later they pivoted and changed the rules of the game, turning their product into Slack.

Slack has shown there is room for play in product.

Here are some technique to insert play into different aspects of product management.

Product Strategy:

Persona tarot cards - These cards help you understand how a feature makes you feel when you use it. To do that ask people during user research what kind of

▼ help they want from your product. The tarot cards represent different personas they could pick from.

In a world where - In this technique, users tell you they want the product experience to feel more like talking to their best friend, what might you change / do differently?

System Board Games - this technique helps you understand the moving elements of the world today. Games are a great way to map out systems. Think about how you would invite stakeholders to help you draw a picture of the product game you're playing and identify areas for improvement.

Feature Development:

Bad Reviews from the future - Think about the human aspects of your users. What are some things your product may unintentionally do to cause your product to lose trust with your customers? To use this technique, imagine a feature you're working on after you ship it gets low reviews. Consider what the most likely reason is. Do this with your team.

Roleplay - Don't just roleplay the role of devil's advocate. Asked high schoolers to role play as AI. Gave adults "Cringe cards" were trying to learn the line between what is acceptable or not coming from AI. Don't underestimate the power of props. Silly String cans renamed as Creativity in a Can.

Team Rituals:

Creative Workout - Creativity is a muscle, so it's helpful to start any brainstorm with a creative warm-up One such warm-up is the Creative Workout. Take two categories that are completely unrelated and then mash them up. Creative workouts generate laughter.

LEGO retrospective - Instead of listing out went well, do differently have people use Legos to build something that represents the answer for went well, do differently.

▼ **BETTER TOGETHER: HOW COLLABORATION BETWEEN PRODUCT AND ENGINEERING TEAMS CAN FUEL GROWTH**

Shadi Rostami, Senior Executive VP of Engineering,
Amplitude

Session Overview:

Product and engineering teams are critical to business growth today. The only problem? They don't always see eye to eye. In this fireside chat or keynote session, Amplitude SVP of Engineering Shadi Rostami will discuss why these teams occasionally conflict and how to best bridge the gap to increase the company's capacity to win. The audience will walk away understanding the value of aligning product and engineering teams, how to create this alignment and forge a joint team culture, and what successful growth outcomes look like. Shadi will also share advice and insights on how optimizing your product development teams can boost efficiency in a time when everyone is seeking to increase ROI.

Key Points:

How you structure your team matters.
Prioritize iteration and innovation as you scale.
Prioritize outcomes over outputs.

Notes:

The Challenge

A common belief in the product community says Product managers think about what and why and engineering teams think about how. That's not how it should go. The Product Team, including product, engineering, and design should think about all three as a group.

So how do you go from the silos to a collaboration (empowered) model?

Will explain three lessons for making product and engineering more collaborative.

Lesson 1: How you structure your team matters

Common industry expectation is that the product team has the triad. (Product, design and engineering) and that collaboration happens at all levels.

The Product Manager acts as the general manager of the business. The designer focuses on experience; The engineer focuses on building the solution.

The engineer chimes in with what is possible and what is not. The Designer chimes in about what the workflow should look like.

▼ But there's another dimension. The Product Manager also needs to think about the rest of the organization - marketing, sales, finance, and legal for example. They monitor how aligned their product strategy is with corporate strategy and connectivity with the rest of the organization.

Once you have your product triad, you can form teams that focus on a particular area of the overall product, These teams are customer facing and are vertical. You may also have platform teams.

Engineers need to be on a customer call once a week. They need to understand the customer. Engineers make a lot of tradeoff decisions.

Lesson 2: Prioritize iteration and innovation as you scale

You'd expect that as you add more engineers, your innovation scales linearly. What you actually get is that it's a curve. You reach a point and then as you keep adding more it decreases.

What you want is that it levels out at some point.

Culture of iteration - Think Big, Iterate Often.

Thinking big is important so that you don't fall into an incremental trap.

Instead of thinking about iterative steps alone, look at if there's new ways to solve the problem.

Customer development partners - if you're building a solution and don't have 1 - 3 customers who will be partners to give you feedback. If you can't find that, don't build that feature.

Number of rituals important to maintain a culture of innovation.

Every all hands do a product demo.
Slack channel for feature releases.

Hackathon has resulted in new feature ideas.

Lesson 3: Prioritize outcomes over outputs

How are you moving on your north star

Even though you're focusing on outcomes, output is still important.

Principle - ship fast so you can get fast feedback. The job does not end at shipping. You need to determine if you got the right outcome. What are the measures you're trying to improve?

▼ Every week the Head of Engineering and CPO sit with a team to discuss how metrics are moving. What are you doing to improve?

Couple of mistakes they've made: Quarterly planning and set quarterly metric led to people to get a short-term focus. If you do this, you need to do something else to encourage folks to think longer term.

Another mistake is the approach to define metrics. Picked revenue but that may not be the best thing to focus on because it drove the team to focus on very large organizations. Realized they should have focused on several paying customers instead of revenue.

When you start a new product, you may not want to structure that team the same way as all the other product teams working on established products.

Think about what you're optimizing for and make decisions. Give the new product team the freedom to make decisions that allow them to optimize on speed.

Mix qualitative and quantitative information to understand how you're doing with metrics.

Measuring developer experience:

When you consider the developer experience, it may influence the future speed of your team. Advocate for thinking about how to improve the developer experience.

PRODUCT
COLLECTIVE

INDUSTRY
VIRTUAL ▶

NOTES

October 26, 2023

