## \*\*Cloud AI Fabric

## ACE is the Automation Center of Excellence:

## De otra Fuente

## **Job Description** The Product Owner works with the Product Manager, Solution Analyst and Development Team in generating user stories Responsible for the analysis and documentation of user stories for features to ensure that they satisfy desired business outcomes and that the development team know what to build They work with the Product Manager to understand the product vision and product strategy goals They organize the breakdown of work for UX and Engineering and then drive the delivery of the outcomes into Clients hands Key Responsibilities: Develop a prioritized backlog of work with Product Manager SA and Engineering Ensuring that user stories are manageable and if necessary sliced into smaller user stories Work with the Product Manager to understand the product strategy and the roadmap that delivers to the strategic outcomes Develop negotiate and finalize user stories Define acceptance criteria along with the team clarity of definition of done Be available to the team to ensure clarity on business requirements are provided as needed Gather feedback to improve features and the requirements gathering process Work with client services product and engineering to reproduce and help resolve bugs Monitor customer acceptance and engagement Proactively contribute to ongoing process improvement Requirement: Experience delivering cloud/web applications in the role of product owner, business analyst or scrum master Excellent organizational analytical written and oral presentation skills Familiarity with requirements design engineering and testing methodologies Solid facilitation skills that draw the most out of all participants and converge rapidly on an outcome Ability to identify potential issues and facilitate development of good solutions Ability to listen and communicate clearly with different areas of the business Superior creative and innovative problem analysis and problem solving skills Ability to prioritize among competing opportunities balance consumer needs with business priorities and articulate the rationale behind decisions Diligent individual who love their team and are driven detail oriented and dedicated **Qualifications**

The days when you could derive a competitive benefit by simply automating some process are long gone. It’s now a question of how effective the automation will be, rather than whether you should adopt it. In other words, the competitive advantage is realized through a holistic and strategic management of all bits and pieces of automation within an organization.

To achieve that, enterprises should implement an automation center of excellence (ACOE) as part of an overall strategy to create new business opportunities. It centralizes and manages automation capabilities, rather than simply streamlining a particular task for an organization. It also treats automation as an ongoing project requiring regular planning, testing, and evaluation.

Today, we are going to take a deeper dive into what an Automation Center of Excellence is, what its benefits are, and who the best candidates are to adopt this approach of strategically managing automation.

## **What Is An Automation Center of Excellence**

Automation technology provides you with a variety of options for meeting your unique business needs, but an ACOE isn’t tied to any particular application. It can manage many types of tools, including job scheduling, [robotic process automation (RPA)](https://blog.accessitautomation.com/robotic-processing-automation) or automation tools built into existing applications.

An ACOE — the right combination of people, processes, and technology — must have strong, policy-driven governance to be effective. It also needs to be highly scalable to future proof it against changes in the size of the organization’s operations.

Furthermore, an ACOE must be integrated into all business processes and driven by business needs by providing a measurable return on investment (ROI) with an acceptable total cost of ownership (TCO). Business process management (BPM) is a similar concept that’s closely linked to ACOEs. BPM attempts to optimize business processes, generally for the purpose of making operations more agile and improving business performance. An ACOE has similar goals, although it focuses on the technological aspects of a successful automation strategy.

Finally, comprehensive reporting capabilities are an essential requirement for ACOEs.

**Benefits**

Organizations have been using the center of excellence (COE) model to transform their operations for decades, but the trend towards ACOEs is relatively new.

Common uses of ACOEs include simplifying complex processes like loan origination, which require the processing of loans by multiple parties in sequential order. Human resources (HR) onboarding and offboarding also lends itself to automation, allowing enterprises to scale their operations quickly without adding full-time employees (FTEs).

Business process outsourcing (BPO) has historically been a popular way to gain the benefits of automation without increasing the workload of their IT staff. However, many organizations are now looking to perform automation in-house once again.

An ACOE lets you take control of your business processes in an efficient manner that will help you maintain evergreen IT processes. It also gives you greater agility since you can configure a robot without being restricted by the terms of a BPO contract.

Technological advances have also facilitated automation, particularly the increasing capabilities of business software applications. Organizations routinely use these applications to help them meet business objectives, including niche applications and in-house tools.

Advances in automation software have been especially great in recent years, as tools are now able to manage many individual tasks. These changes are [driving the trend towards digital transformation](https://blog.accessitautomation.com/resources/automate-your-windows-10-servicing-guide), allowing organizations to integrate their automation solutions with other applications. This capability provides a number of benefits such as maximizing the ROI of existing software, enabling DevOps, increasing information security and data analytics.

Many organizations use an ACOE to meet these objectives, typically by implementing automation in stages. This practice gets everyone involved with technology, which can blur the lines between IT, operations, and business departments.

A solid automation strategy is thus essential for avoiding confusion over governance and security issues. An ACOE helps different roles work together in a structured manner, allowing people in each group to contribute their unique skills towards furthering business objectives.

## **Who Is A Good Candidate For This?**

Large organizations commonly experience the challenge of complex operating environments with individual areas that use automation in isolation from each other. An ACOE can help tie these islands of automation together by identifying and prioritizing new opportunities for automation, without needing to reinvent the wheel each time.

You may think that only large enterprises need an ACOE, but it can also benefit smaller organizations. An unstructured approach to automation may work well when you need to automate a few workflows, but failing to implement best practices can restrict future growth as your need for automation increases. The limited budgets that small companies typically have also make it more important for them to maximize the ROI of their investment in automation software.

## **How Should Your ACOE Be Organized?**

An ACOE’s structure may be classified into people and process, and systems and infrastructure. Both of these pillars require a solid foundation before you can begin an automation project for your organization that will make the most of your automation investment.

Putting the right people and processes together is the first step in achieving an effective ACOE. Your IT department doesn’t necessarily lead an automation project, but they should be involved from the beginning. The core group will need a variety of skill sets to perform diverse tasks such as gathering business requirements, documenting existing processes, researching automation solutions and developing workflows.

Your systems and infrastructure provide the technological foundation for your ACOE and help prepare your organization for the rapid implementation of automation. DevOps best practices are just as important in IT automation as they are for other systems, so it’s critical for you to test, monitor and evaluate when changing them. This approach helps minimize downtime, especially when running automated processes across an entire enterprise.

## **Summary**

Implementing an ACOE will require you to change some mindsets in your organization, since not everyone will see the [benefits of automation](https://blog.accessitautomation.com/benefits-of-automated-app-packaging) when you begin this project. Some employees will need to see positive results before you’ll be able to sell them on your new approach to automation.

It is crucial to the success of your ACOE that all involved teams adopt an [Agile development and testing approach](https://blog.accessitautomation.com/agile-it-planning-deployments-to-drive-digital-transformation) to thinking and implementation as this isn't a static, one-off project, but rather a holistic way to constantly evolve, improve, and perfect your automation.

Start this process by identifying people who already support automation initiatives, so they can help you promote support for automation throughout your organization. Appointing people with the right skills to lead your core team is also a key requirement for a successful ACOE implementation.

RDA:

Business citizen developers

The Need for Citizen Development

Citizen developers are business users with little to no coding experience that build applications with IT-approved technology.

The concept of end-users creating their own solutions might not be new, but organizations understanding the potential impact of empowering citizen developers with more powerful tools is. Organizations are noticing the value of citizen development. According to a recent [Gartner report](https://www.gartner.com/en/documents/3970067/the-future-of-apps-must-include-citizen-development), 61% of organizations either have or plan to have active citizen development initiatives.

**Here’s why there’s a growing need for citizen development:**

#### The IT need

Enterprise digitization efforts are more mature – [40% of CIOs](https://www.gartner.com/en/newsroom/press-releases/2019-10-22-gartner-survey-of-more-than-1000-cios-show-that-fit-enterprises-will-win-when-business-conditions-turn) have reached scale for their digital endeavors, more than double the proportion of digitally-transformed enterprises from 2018.

The result of a more mature digital business model is a never-ending IT backlog. [Research](https://www.mendix.com/press/business-in-crisis-due-to-it-resource-shortages-self-medicating-with-shadow-it/) has found that 77% of IT leaders and 71% of business leaders agree that IT teams have a huge pipeline of new IT solution requests, which aren’t being built. In addition to keeping an organization up and running from a technology standpoint, IT is under immense pressure to modernize the existing IT infrastructure and advance their organization’s digital agenda by building new business-critical apps.

#### The business need

On the other side of the house, business users are fixing their problems by leveraging dated methods such as spreadsheets, desktop databases, etc. Overtime, rampant unchecked development by business users, magnified by the boom of SaaS applications, has resulted in a complex network of solutions that IT has no visibility into.

## Citizen DevelopmentChanges the Game

When you look at the business impact of IT’s inability to deliver new solutions on time, it puts the rise of citizen development into perspective.

**How is the business impacted when IT is not ableto deliver new IT solutions in a timely way?**

Graphical user interface, diagram, application

Description automatically generated

## The Benefits of Citizen Development

Through citizen development, IT can partner with the business to build solutions the organization needs - faster and with less rework. The business users know their domain better than anyone else and working collaboratively will ensure that the right processes and technology are in place. When done right, citizen development can help you:

### [Meet the growing need for apps›](https://www.mendix.com/citizen-developers/)

Today every company is a software company and needs more applications to improve operational efficiency, drive innovation, and generate revenue. IDC predicts that by 2023, over 500 million apps will be developed. That’s the same number of apps developed in the last 40 years! IT, with its limited resources, is never going to be able to build the apps business needs in time. Citizen Development represents the best bet to overcome this challenge.

### [Address shortage of skilled developers›](https://www.mendix.com/citizen-developers/)

With the software developer [unemployment rate below 1.9%](https://insights.dice.com/2018/05/23/tech-unemployment-rate-19-percent-good-news-tech-pros/), there’s a limited pool of skilled developers in the market. The stark shortage of talent puts additional pressure on IT. Organizations that adopt a formal citizen development program address the challenges of talent shortage head-on by empowering non-technical, highly motivated business users to build apps that solve their immediate problems.

### [Govern shadow IT›](https://www.mendix.com/citizen-developers/)

Our recent [study](https://www.mendix.com/resources/digital-disconnect-a-study-of-business-and-it-alignment-in-2019/) confirms that 78% of IT and business leaders agree shadow IT has increased greatly over the past 5 years. While the existence of shadow IT is mostly good ([yes, it really is](https://www.mendix.com/blog/why-shadow-it-is-great/)), when left unchecked it can create a technical wild west for IT. Instead, IT should govern it using a program like citizen development. When a recognized citizen development program is in place, all those must-have apps are built on IT-approved platforms, giving IT the visibility and control they need to maintain and monitor quality and security.

### [Boost IT and business productivity›](https://www.mendix.com/citizen-developers/)

Citizen development provides business users with the approved technology and required IT support to build the basic productivity apps they need. When citizen developers build their own apps, they can perform their job more efficiently, help the company save on operating costs, and help reduce IT backlog. IT is then free to focus on more technically complex projects like legacy modernization, core system extensions, and enterprise-wide applications.

### [Break down silos›](https://www.mendix.com/citizen-developers/)

As companies grow, so too do the silos within the organization. There’s already a [disconnect between IT and the business](https://www.mendix.com/resources/digital-disconnect-a-study-of-business-and-it-alignment-in-2019/). Citizen development can help repair this disconnect by creating an opportunity for business users and professional developers to work collaboratively. For instance, when the app a citizen developer is building on the IT-approved tool gets complex, they can invite a professional developer to help build the complex parts in the same tool.

Today every company is a software company and needs more applications to improve operational efficiency, drive innovation, and generate revenue. IDC predicts that by 2023, over 500 million apps will be developed. That’s the same number of apps developed in the last 40 years! IT, with its limited resources, is never going to be able to build the apps business needs in time. Citizen Development represents the best bet to overcome this challenge.

Defining the vision for the team's product and analyzing tradeoffs.

<https://www.christianstrunk.com/blog/product-vision>

# How to define a product vision (with examples)

There’s an assumption that the “average” person has about 10 ideas in their life that could/would change the world. Very successful people have about 15-20 of these ideas. That’s interesting, isn’t it?

How do you really know that an idea could become a life-changing billion-dollar product or service?

I don’t have the answer to this question. What I do know is that continuous product discovery and a clear product vision helped many successful companies to get there. Ok, some companies made it with shit loads of money… but that’s a different topic altogether. 😉

In this article, I’ll focus on how to define a product vision. I want to highlight that creating a vision for your product is one of the hardest challenges. It has a lot of implications for the organization and the teams within it. I’ll break it all down into simple segments from the definition to the creation process underlined with real-world examples.

**Important note:** There is/can be a difference between a product and company vision. In this article, I’ll focus on creating a **product** vision.

## **What is a product vision?**

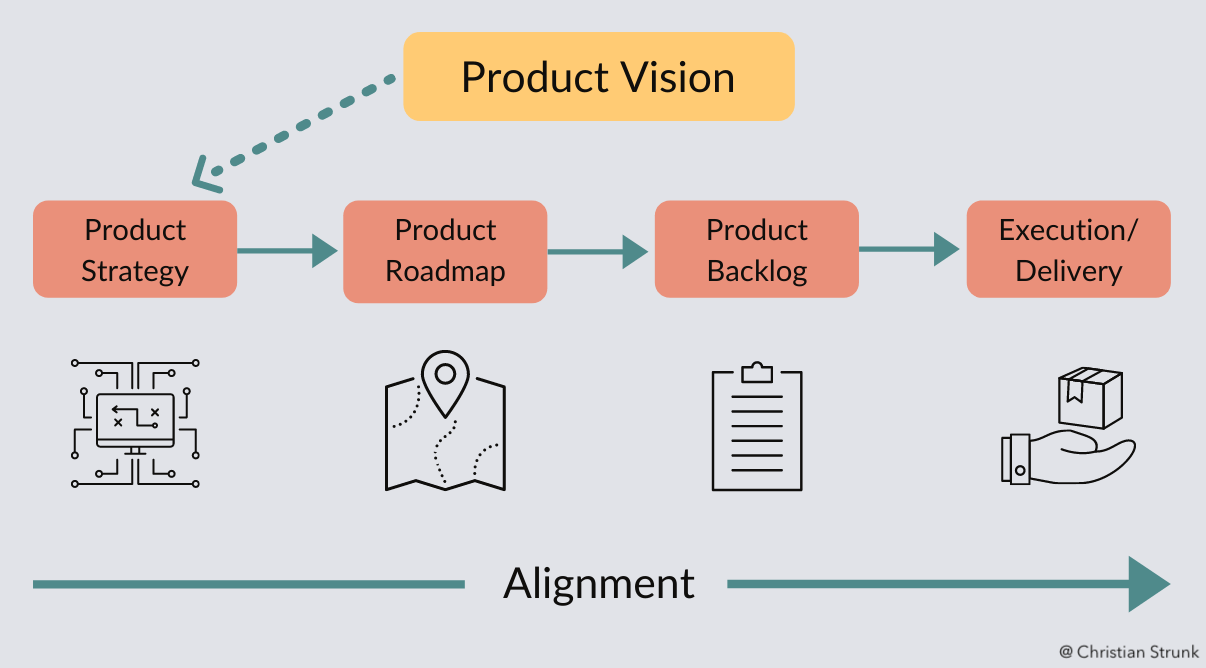
A product vision describes the future state of a product that a company or team desires to achieve. You can also define that future state as: **a goal**.

But, the product vision isn’t just that! It’s a **foundation** for other big topics such as the

# Anexo Home

* [product strategy](https://www.christianstrunk.com/writing/product-strategy) [anexo 1](#_product_strategy)
* [product development roadmap](https://www.christianstrunk.com/blog/product-development-roadmap) [anexo 2](#_2.__product)
* [backlog & planning](https://www.christianstrunk.com/writing/5-steps-to-make-backlog-groomings-more-engaging) [anexo 3](#_3._backlog_&)
* [execution & product launch](https://www.christianstrunk.com/writing/product-launch-checklists) [anexo 4](#_4._execution_&)

A great product vision will ease up the definition and alignment process for all of the above topics.



Before we jump into the definition process, let’s clarify a few pieces of core information around that surround the product vision.

### What information does a product vision contain?

1. The reason every company exists is because of its customers. Whether they’re end-customers (B2C) or businesses (B2B). That’s why the most important characteristic of a product vision is that it’s focused on the customer!

2. It’s looking into the future and outlining a clear state of the product/goal that the company and team(s) want to achieve. I recommend setting a **big** and **bold** goal! As I mentioned in [5 Best practices to make your OKRs awesome](https://www.christianstrunk.com/blog/5-okr-best-practices):

“We don't fail because we aim too high and miss. We fail because we aim too low and hit." – Les Brown

3. This goal should be underlined with the motivation behind it. This motivation should answer the “why” (not how!).

4. The art of defining a great product vision that people want to follow is to make it catchy. Something that makes you want to pitch it to other people no matter if you’re the founder or an employee.

5. (optional) A product vision can have a time constraint, although, it’s better to define it as “timeless.”

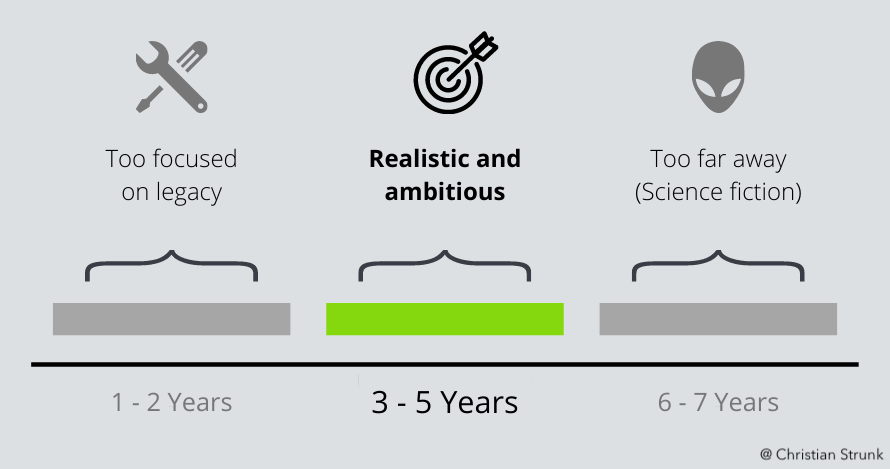
### Product vision time frame

As mentioned, the product vision defines the future state of the product that you want to achieve. Bold and achievable.

What is the best time frame though?

I believe a timeframe of 1-2 years is too short because people tempt to focus too much on legacy issues. A product vision that looks 6-7 years into the future might be too much “science fiction.”

Therefore, I recommend going with something in between. A timeframe of **3-5 years** isn’t too close and not too far away.



### What a product vision isn’t!

A product vision isn’t a detailed plan that explains how to reach the goal. It’s important to leave the goal kind of “open” with some room for interpretation.

Let’s have a look at the product vision for Google Calendar. The team realized in about 2006 that no other company offered a great calendar product and experience. Therefore, the team came up with a simple 4 point vision:

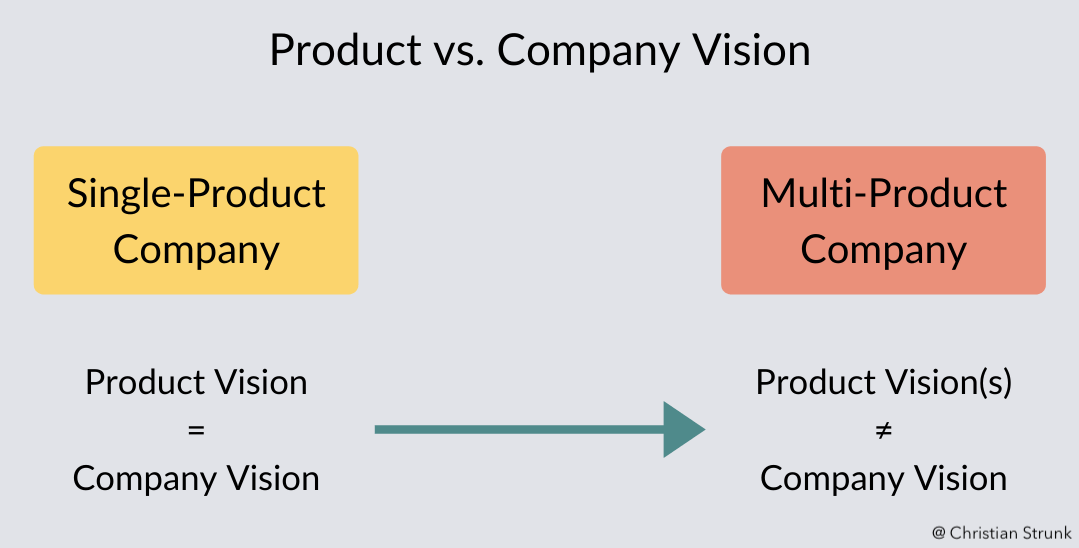
* Fast, visually appealing, and joyous to use
* Drop-dead simple to get information into the calendar
* More than boxes on a screen (reminders, invitations, etc.)
* Easy to share so you can see your whole life in one place

It contains no detailed explanation on a feature level, although the goal is clear. It may not be sexy, but it helped the team to build the most popular calendar application on the planet.

There’s a difference between a company vision and a product vision. The company vision isn’t necessarily the product vision. Especially in multi-product companies. Google, for example, offers many different products, such as their search engine, Google Adwords, Google Calendar, GSuite, and more. Each product has its own product vision. Ideally, the product vision should be connected to the company vision. Google’s company vision statement is: “to provide access to the world's information in one click.” because that’s Google’s core business.

In single product companies, the product vision is usually the company vision and vice versa.

**Note:** The whole topic about company vision is a bit more complex. I’ll write about it in more depth in a separate article soon.



To summarize, it’s important for us (especially Product Managers) to understand the definition and core information a product vision needs to contain. With that in hand, we’ll always have a good foundation to start defining our own product vision. 

Bottom of Form

.

## **How to define a product vision (example)**

To me, product vision creation is a two-part creation process

1. Defining key product information.
2. Phrasing the product vision in one inspiring sentence.

Let’s take a deeper look into each step.

### 1. Defining key product information

Before defining a product vision, it’s important to have some valid data. The whole product discovery process is obviously a great way to get data and find answers to open questions. Without the data, it doesn’t make sense to define a vision for the next couple of years. It’s, however, important to remind ourselves that a product vision needs to be regularly challenged, evaluated, and updated based on new data.

According to **Roman Pichler’s** product vision board, it’s important to answer 4 key questions:

1. What’s the target group?
2. What are the customer needs?
3. What is and will be the product and it’s USP(s)?
4. What are the business goals?

Alternatively, you can work with other models, e.g. the lean business model canvas focusing on the product. What’s most important is gaining a clear picture of your customer, your market, the problems you want to solve, and your business goals. You can build a product vision based on that.

**Let’s look at the data we have for our example product (simplified example):**

* A single product company producing card readers (credit & debit cards)
* The buying persona/target customer is between 35-45 years old, owning a small business (max 20.000€ revenue p.a.),
  + They live in Germany, UK, Italy, or France
  + They want to be able to accept card payments
  + They want to have as little administrative efforts as possible (billing, tax declaration, etc.)
  + Payments have to be super fast (faster than cash)
* The business goal is to move up to a 10% market share across the 4 mentioned countries (**bold**)
* Provide the fastest and easiest onboarding (for business owner) and payment experience (for customers)

Depending on the state of your product and research you may have more or less information. For the product vision statement/sentence it’s important to focus on the most important ones.

### 2. Phrasing the product vision in one inspiring sentence

Before we think about a final sentence it’s important to ask ourselves two important questions:

**1. What are the key values/attributes of our product?**

Our card readers enable merchants and customers to make fast and safe cashless payments.

Looking at the Google Calendar application the main category is time management or self-organization.

For the Spotify app, it’s music streaming. We could go deeper and connect music with creativity, joy, freedom, and many other things. I’m pretty sure you and the Spotify Product Managers know more than I do. 😉

The reason I’m looking at this question is to identify the key value and connect it with strong verbs and adjectives that we can use for our vision statement.

**2. What will our product do better/make outstanding in the future?**

I look at topics such as:

* additional value/problem-solving features
* improved performance/quality
* competitive advantage
* better usability/experience
* … and more

Are you familiar with the typical bold CEO product requirements that ignore all technical difficulties, legal requirements, or other important things :

* “It must be a super-fast transaction, less than a second!”
* “We need a great one-click checkout like Amazon!”
* “The data must be accessible at any time, even without an internet connection!”?

Great! These statements can be perfectly implemented into your product vision.

**Phrasing the inspiring product vision statement**

Going back to our card reader example, we do have the following key information in place:

* Our customers are owners of small/medium-sized businesses (SMBs)
* We empower the owners to accept card payments
* End-customers will have a great, fast, and safe payment experience (faster than cash)
* It’s easy to use and has a minimal administrative effort

A way to phrase the vision statement could be:

“We believe in a world where small businesses can offer a super fast and safe payment experience to their customers, for minimal costs with no administrative efforts.“

Obviously, that’s one way to phrase a vision. The way you phrase it depends on the length, complexity of the sentence, and your personal preferences.

I’d like to share my three favorite vision statement templates:

1. We believe (in) a [**noun:** world, time, state, etc.] where [**persona**] can [**verb:** do, make, offer, etc.], for/by/with [**benefit/goal**].
2. To [**verb:** empower, unlock, enable, create, etc.] [**persona**] to [**benefit, goal, future state**].
3. Our vision is to [**verb:** build, design, provide], the [**goal, future state**], to [**verb:** help, enable, make it easier to...] [**persona**].

## **Frequently asked product vision question**

I regularly get asked questions by Product Managers or Product Teams I coach around the product vision. Therefore, I summarized the main questions and answers below. If you have more questions don’t hesitate to contact me.

### Why is having a product vision important?

I’ve worked for companies with and without a product and company vision. There’s always a big difference in companies that work with visions. It gives teams a clear direction and fosters a common understanding of where they want to be in the future. Product Managers and their teams can make better decisions and prioritize based on the product vision.

### Who creates and owns the product vision?

That’s a tricky question and it depends. In early-stage startups, the product vision is defined by the founders. That doesn’t mean that the first employees are excluded from the definition process.

In growing and bigger companies, the Product Manager and Teams define the vision together with their Development Team. Nevertheless, it’s not owned by them in my opinion. Neither by the founder(s) nor by the Product Managers. Every employee should feel responsible when contributing to and challenging the vision at any time.

As an employee, if you don't contribute towards the vision, then don't be surprised if it turns out to be something you don't agree with. The same concept applies to a Founder or Product Manager. If you don't engage your colleagues in the vision, don't expect them to follow it.

### Is there a process for creating a product vision?

That depends on how you work as a Founder/Product Manager. I’m always a fan of going the “formal” way and inviting people to a meeting. Sharing an agenda and documents with the attendees beforehand will help them to better prepare upfront.

If you want to learn more about the process I recommend listening to this podcast episode:

### Is working with a product vision agile?

The product vision is definitely compatible with the agile mindset and methodologies such as Scrum, Kanban, and any other. The vision shows a clear direction and teams can decide how they get there. They can adjust the plan and the vision at any time **if needed**.

### How to use the product vision after its creation?

It’s crucial to regularly share and repeat the product vision process. Especially in backlog grooming sessions and sprint planning sessions. I always tell product people to regularly align with the Leadership Team and challenge if

1. the company is on track and heading in the right direction
2. the vision is still valid or needs to be adjusted.

Syncing on this topic once a quarter for an hour isn’t too much.

Last but not least I’ve got one more important product vision secret to share.

## **The most important product vision secret**

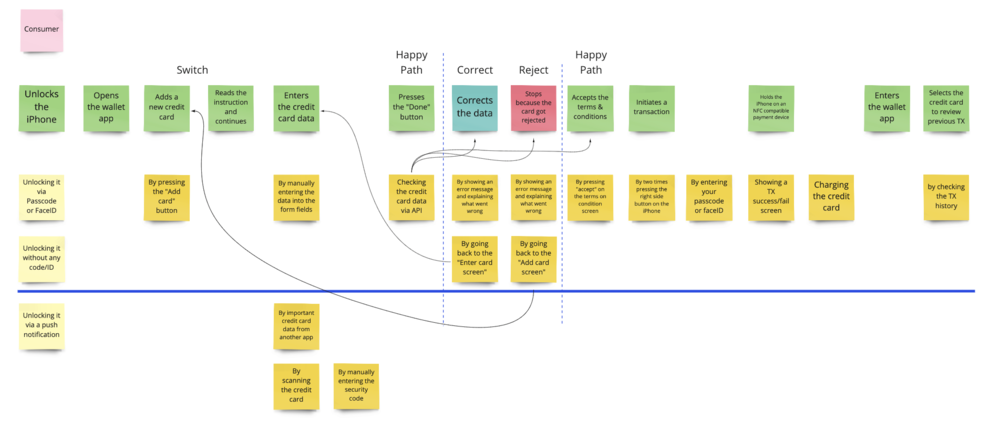
What’s most important is to not copy any fancy product vision one by one! It’s important to define a vision everyone buys into, believes in, and works toward. Look at successful companies:

They spend a lot of time and money on offsites, meetings, fights, time with coaches and consultants, etc. to come up with their own unique vision. Alignment and a clear direction is the key! Make sure your product vision has an individual touch to it. I can tell you from my experience, people love that most. That’s it. **That’s all the magic. ✨**

If you think I should mention this on top of the article and delete the rest of it please tell me on Linkedin [@christianstrunk](https://www.linkedin.com/in/christianstrunk/).

### Do you want to master product & feature planning?

## **Check out the video course!**

[[](https://www.christianstrunk.com/story-mapping-video-course)](https://www.christianstrunk.com/story-mapping-video-course" \t "_blank)

**ANEXOS**

## [product strategy](https://www.christianstrunk.com/writing/product-strategy" \t "_blank)

[Back](#_Anexo_Home)

# Important elements every new product strategy needs to have

About eight years ago I created my first business plan for my first startup. I recently found the business plan and decided to have a nostalgic look at it. While flicking through the pages, I came to an interesting conclusion:

A business plan and a product strategy have many things in common.

They both require market research, numbers, roadmaps, forecasts, and more.

That’s when I asked myself what the difference between the two documents was? In this article, I’d like to elaborate on the key elements a new product strategy needs to have. I’ll explain how you can create a product strategy, what time frame, and the difference between it and other forms of documentation like a business plan.

At the end of the article, you can download my product strategy slide deck with many more examples, templates, and tips.

## **The product strategy definition**

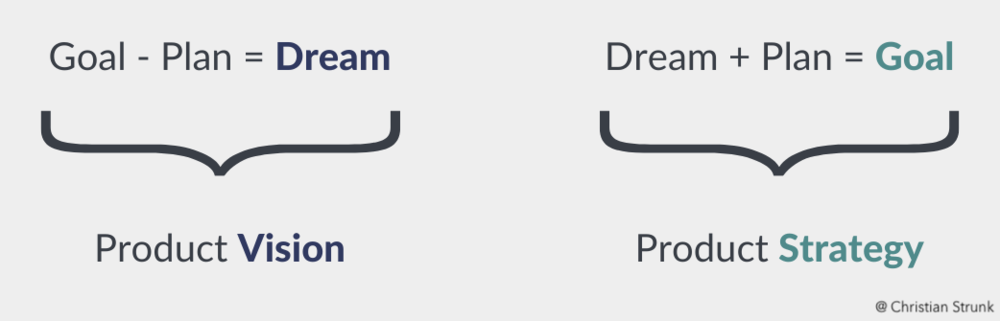
A product strategy is a high-level plan for a team/company to achieve a certain goal or goals with a product/service. These goals are derived from the product vision which plays a key role in the product strategy. Using the product vision I can clearly define the following my product strategy:

* for whom we’re building the product/service
* which problem it solves
* how the product will look like
* what the business goals are
* etc.

A product strategy should always be related to a product vision. There’s a quote from Dave Ramsey that says:

“A **goal without** a **plan** is **just a dream**."

Let’s do the math!



Everything starts with a dream. To make it come true, it’s important to develop a plan.

### The timeframe of a product strategy

I’ve had multiple discussions on the time constraints of building a product strategy. These days you see more and more Product Managers and Leads categorizing everything that’s longer than 6-12 months as “product strategy.” In my opinion, the strategy shouldn’t be longer than 2 years, especially in these very uncertain times.

If goals and milestones are shorter, they fall into the category of a roadmap. I think it’s not easy to nail it down in 6, 7, or more months. It depends on many factors such as the:

* industry (e.g. healthcare vs. online gaming)
* customer relationship (B2B vs. B2C)
* release cycles
* promised delivery date/ETAs
* and more

The problem with "roadmaps” that are longer than 12 months is that you likely won’t be able to hit them.

## **Key elements a product strategy needs to contain**



The product strategy needs to cover three different areas:

* Business/financial goals
* Market & competition
* Product features & USP.

Each of them needs to be further broken down. That’s the tricky part! I’ve seen many product strategies that contain great information but are simply horribly structured. In 2017 I decided to create a “**product strategy agenda**” for myself. This agenda helped me

1. to better structure the content & information
2. to not forget important topics.

Not all of the agenda points are needed or mandatory. If you’ve read some of my [articles](https://www.christianstrunk.com/blog), you may have noticed that I like to share real examples that you can use as a template and for inspiration.

Let’s have a look at each agenda point. Additionally, you can listen to my podcast episode about product strategy:

### #1 Product vision

The product vision describes the future state of a product that a company or team desires to achieve. The product strategy should always start with the product vision for two reasons:

1. You set a clear frame and goal
2. Based on that you can derive and justify the strategy

Keep in mind that a product vision should be a short and clear sentence and not a long piece of text. But that’s a whole other topic in itself. You can read more on it in my article “[product vision (with examples)](https://www.christianstrunk.com/blog/product-vision).”

**Note:** If you don’t have a product vision defined yet, I recommend reading this article first.

### #2 Business goals

Every business has its goals. Without goals and priorities, companies aren’t able to survive. Product goals are always coupled with business goals and sometimes it can be hard to distinguish the difference between the two. Let’s look at a simplified example:

Let’s say we’re a bakery in New York that only bakes whole-grain bread in different sizes, but always the same sort. Product-wise, we have a lot of options:

* Start baking other types of bread such as wheat-baguettes, croissants, etc.
* Start selling sandwiches and coffee to go.
* Start baking cakes, donuts, etc…

That all sounds great, nevertheless, the main business goal is to “conquer” the whole-grain bread market and to become the biggest and best in that niche. That’s a much clearer business goal. Find your niche, and own it. .

Having such a clear business focus and tailored business goals need to be highlighted at the beginning of a product strategy. It’s important for Product Managers to know this and to align with the key stakeholders/leadership team from the outset.

### #3 Market analysis

To truly understand what the product is and where you want to be in the future, it’s important to look at the market. Every product strategy needs to provide clear information about the market. There’re many techniques and ways you can define the market size. I don’t think there’s a one-size-fits-all solution. The deeper you go, the better. Here are a couple of methods, you can use to analyze the market:

* Top-down or bottom-up market analysis
* Competitive analysis
  + Comparison landscape & matrix (feature/product level)
  + SWOT analysis
  + PEST analysis
* Market needs & trends
* Market positioning (e.g. longtail vs. upmarket)
* More…

On top of that, it can be helpful to share the market history or the evolution of certain customer behaviors or trends. That helps people to understand where we’re coming from. 😉

#### **Free Product Strategy Template**

**Access** the free product strategy **slides**with many more in-depth examples & templates.

Top of Form

Get the template

Bottom of Form

.

### #4 Persona(s)

Depending on your product and its complexity it can have more than 1 (product) persona. Nevertheless, I’d try to keep things clear and simple with the main information about the persona(s) being something along the lines of:

* Name
* Age
* Gender
* Education/job
* Income
* Interests/motivation
* Technology they use
* …

It’s, by the way, never bad to link to e.g. a Google Drive where people can find all the research, interviews, and more information. I like presenting the persona slide after the market analysis slide. However, I’ve seen product strategies where this was reversed.

### #5 Development team/squad

A slide or slides about the team are optional. This all depends on the size of your company and whether or not it’s worth listing every member of your team. For bigger organizations, I recommend adding a team slide. Presenting a picture, name, and title is more than enough. If you’re hiring new team members, it’s worth mentioning the hiring plans and open positions as well.

For bigger product strategies with a cross-functional team impact, it can even be helpful to present the outcomes of “skill mapping” in combination with the hiring plan. I’ll write more about this in the future.

### #6 Product: features, priorities, and USP

This part of the product strategy outlines key features and priorities as well as the unique selling proposition (USP) of the product/service. Depending on the state of the product (new vs. existing) I always define the key features/priorities.

**Example:** Let’s imagine Facebook doesn’t exist and we decide to build the very first social media platform in 2020 from scratch:

1. Since most people use their smartphones these days, one of the highest priorities would be having a mobile-first approach.
2. Having a mobile-first approach means that user experience would be very important. Therefore, we would focus on the best user experience.
3. Based on the MVP we would then map out the [user story mapping](https://www.christianstrunk.com/writing/how-to-define-an-mvp-with-user-story-mapping) and focus on account creation, a timeline, and connecting to other people.

I’ll leave it at just those three points because I’m not an expert when it comes to building a social media platform. 😉

It’s important to share with stakeholders and with your team the “frame”. Something I forgot to measure multiple times in my product strategies was explaining how we actually measure success! I recommend adding at least one slide that highlights the key KPIs you use to measure success/failure. You can learn more about this in my article about “[Product KPIs that make all the difference](https://www.christianstrunk.com/blog/7-product-kpis).”

### #7 The product roadmap

The whole “art of road mapping” is worth an article in itself (see below). For the product strategy, it’s best to keep it simple though.

I present the roadmap in two parts:

* The long term outlook (2-3 years)
* The short term roadmap (6-12 months)

I create e.g. a Gantt-chart and highlight different features or steps in different colors. This can be very helpful to show the planned high-level progress. Visualizing the roadmap on a higher level helps to avoid discussions on all the nitty-gritty details with your stakeholders. The short term roadmap is what my team and I commit to. After that, it becomes vaguer and less specific to give people an outlook into the future.

In my product strategy template, you can find a couple of examples of how to present a roadmap. Here you can read more about “[the agile way of creating a product development roadmap.](https://www.christianstrunk.com/blog/product-development-roadmap)”

### #8 Numbers and data

Despite the product strategy not being a business plan, it’s good to equip it with numbers and data. Depending on your product or service you can present:

* Revenue forecast
* Profit & loss calculation
* Conversion rate prediction for
  + Signup
  + Sales
  + Monthly active users (MAU)
* and many more others

When I get asked how long the numbers should look into the future I reply:

“How far are you confident in predicting this?”

Make sure the data is as accurate as possible. You’ll be measured by the outcome.

## **Product strategy template/framework**

To summarise, the product strategy should focus on business goals, market needs, and key product features and priorities. To create a well-structured strategy paper you can use the following agenda:

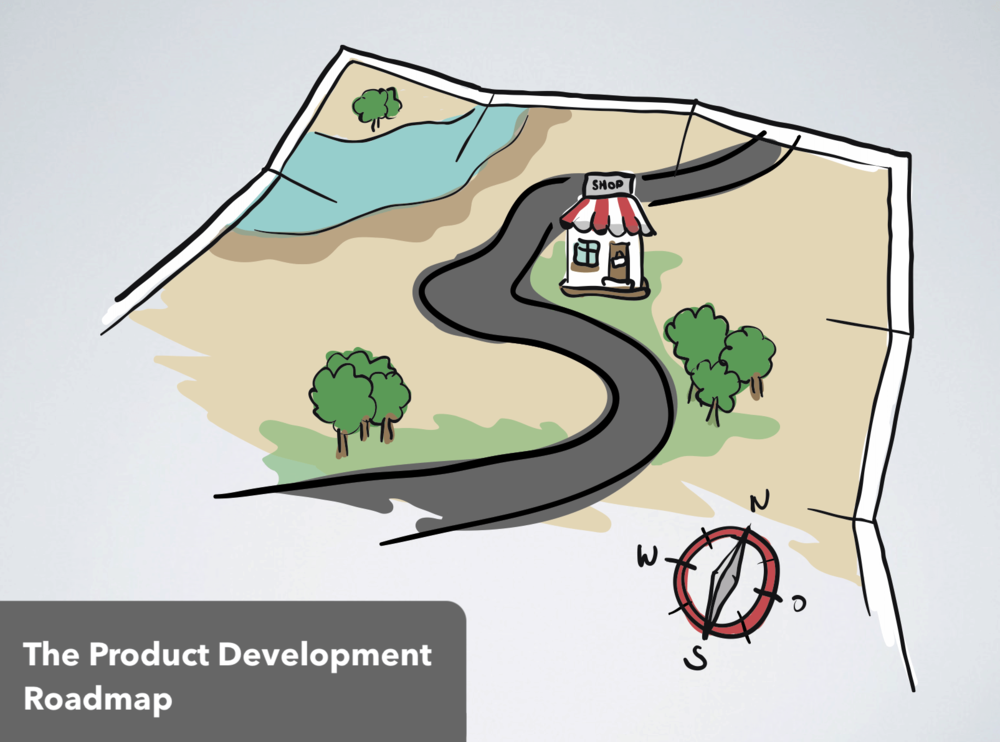
1. Product vision
2. Business goals
3. Market analysis
4. Persona(s)
5. Development team
6. Product features & priorities
7. The roadmap
8. Numbers & data

If you’d like to get some inspiration on important elements of a new product strategy, then feel free to download my product strategy template. I’ve created many different examples, versions, and templates of the mentioned agenda points which you can use as a framework on more than 40 slides. On top of that, I’ve added many more useful links if you want to dig deeper into certain topics.

# [product development roadmap](https://www.christianstrunk.com/blog/product-development-roadmap" \t "_blank)

1. [Back](#_Anexo_Home)

# The agile way to create a product development roadmap



In my second job as a Product Manager, the VP of Product loved roadmaps. He requested a roadmap and slides for literary everything. He always said:

“Give me an exec-summary and 1-3 nice slides that I can share with the Management Team.”

On some days I felt more like a Slide Designer than a Product Manager. Nevertheless, during that time I learned a lot about prioritization and creating agile product development roadmaps. (+ Google Slides 🎨) Both of these things go hand in hand and are very important when it comes to making a good short-term plan.

In this article, I’ll take you through the step-by-step roadmap creation process which will cover:

1. How to align and prioritize upfront
2. How to create a product development roadmap with your team
3. How to share and pitch it to stakeholders.

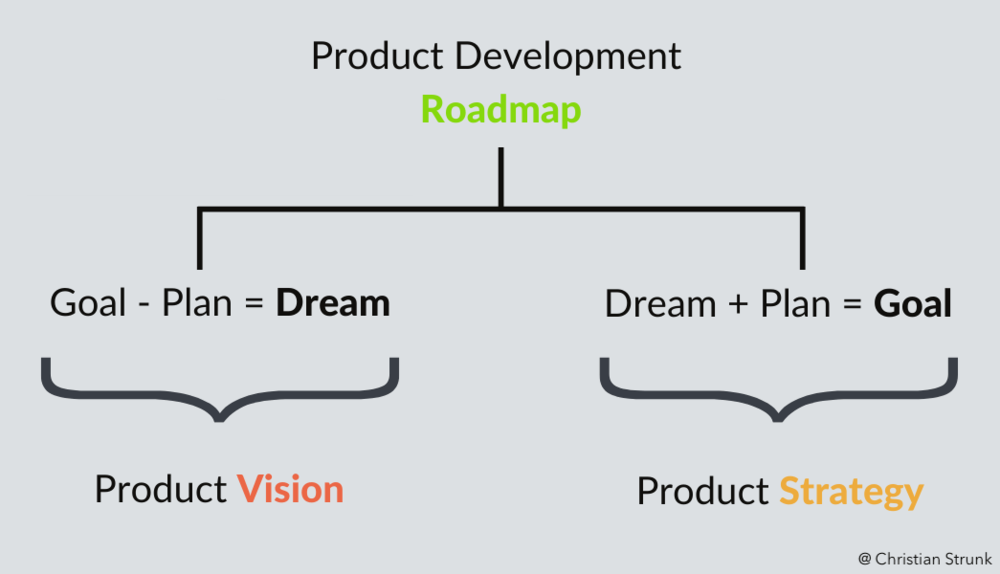
Even if I’m focusing on a “product development” roadmap, you can apply most of the principles to almost every other type of roadmap.

## **What is a product development roadmap?**

I think the best way to explain that is in pictures. In my previous article about the [elements of a product strategy](https://www.christianstrunk.com/blog/product-strategy)﻿ I highlighted this quote:

“A goal without a plan is just a dream.”

That’s what I develop the product roadmap on:



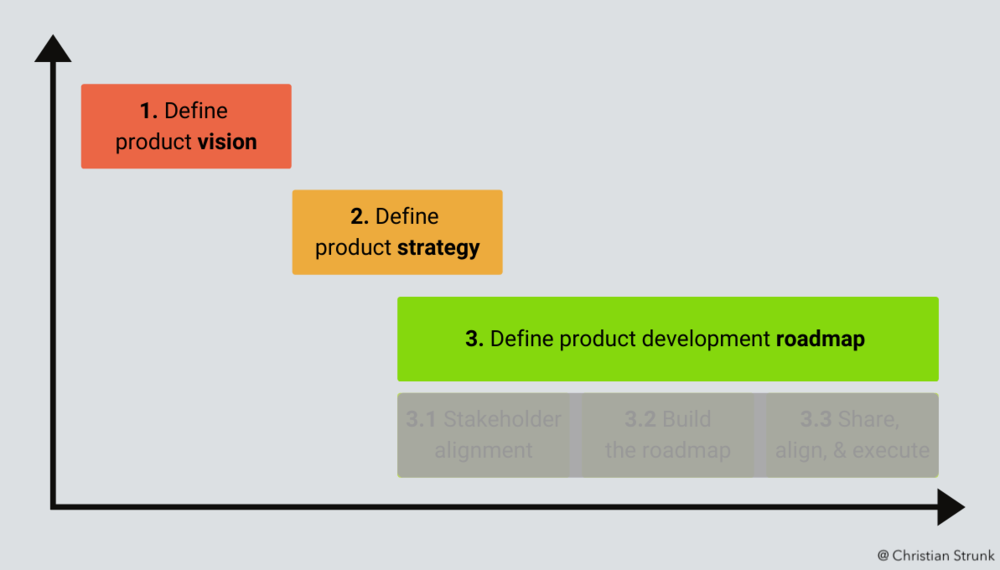
Everything a Development Team works on is subject to a bigger goal. The bigger goal is called “[the product vision](https://www.christianstrunk.com/blog/product-vision)” and the high-level plan to get there is “[the product strategy.](https://www.christianstrunk.com/blog/product-strategy)” The detailed plan is the product development roadmap.

A big mistake I made many times when kicking off my career was to simply “create a roadmap.” Over time I’ve learned and realized it’s way more than that. The roadmap creation process requires preparation, communication, and alignment. The result of that isn’t just a couple of slides that present the planned future. It contains answers to the following questions:

* What’s our goal(s)?
* Why do we want to achieve it?
* Who will build the solution?
* When will milestones be achieved?
  + Where do we stand now?
* How will we get there?

If a product vision and strategy are already in place you can reuse them for your product development roadmap. If not, these questions should be answered separately in the roadmap slides or a tool of your choice.

### Product development roadmap creation process



Abraham Lincoln once said:

“Give me six hours to chop down a tree and I’ll spend the first four sharpening the axe.”

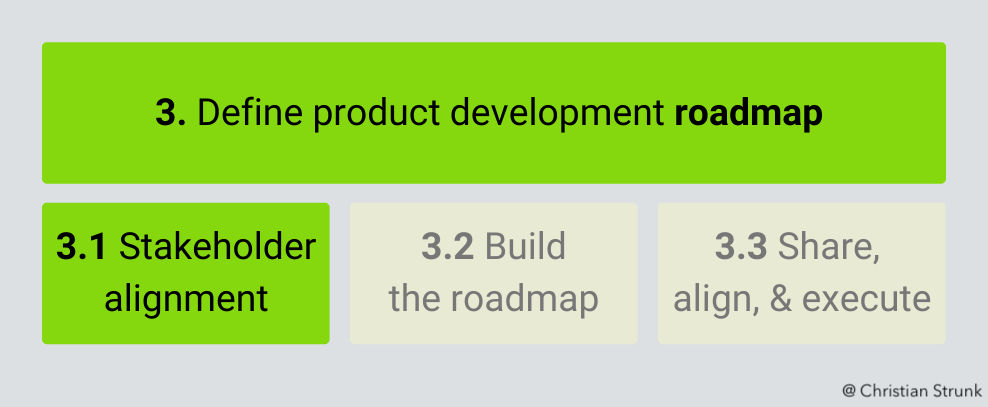
I think that’s a great mindset many people should adapt to have.

There’re 4 crucial things a Product Manager needs to do to be able to create a great product development roadmap:

1. Understanding the product & market
2. Reaching out to stakeholders of all kinds
3. Building the roadmap
4. Sharing and aligning it with the team and stakeholders.

Number 1 is a prerequisite and worth many separate articles. Number 2-4 are necessary in order to start the “real” work. I always initiate point 2 by talking to stakeholders.

## **Stakeholder communication and prioritization**



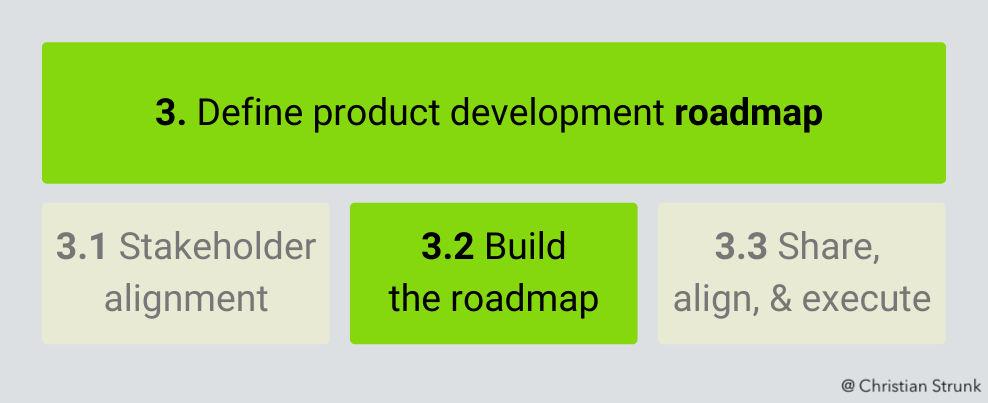
I see the interaction part with stakeholders as the “axe sharpening” part of the quote. Knowing what the market needs doesn’t set the roadmap in stone. Stakeholders have their own needs, wishes, and requests. A Sales Team might want different features to those a Support Team would want. What’s most important, is to listen to people and gather the relevant information that impacts me and my team.

I recommend doing real deep-dives when you talk to stakeholders and asking them what their needs are. There were many times I was just taking note of something they wanted without asking why and for more details. I’ve always found it helpful to invite an Engineer to conversations with technical stakeholders. Not only to better understand everybody’s needs, but also to mutually exchange knowledge. Involving team members helps them to better grasp the bigger picture and to develop a sense of importance and urgency. In his book “Inspired”, Marty Cagan recommends always interviewing customers with a Product Manager, an Engineer, and a Designer. Involving at least one of them in stakeholder discussion won’t hurt too. I wrote a “[How to identify and manage stakeholders](https://www.christianstrunk.com/blog/how-to-manage-stakeholders)” article where you can learn more on the topic.

The second and hardest part I’ve been faced with multiple times was that of prioritization. Obviously, I can’t fulfill all wishes at once. Some of them never. That means, saying **no** is not only important, it’s also crucial.

**Note:** Please keep in mind that the preparation phase requires talking to your team. You can only gauge the bigger picture when you know what they plan to build technically (+ dependencies). And they only know what to plan for the future if they know what’s most important product-wise.   
It’s **not** a chicken and egg problem. The product vision and strategy give everyone a clear direction in the first place.

## **How to build a product roadmap**



The product development process covers many different steps and phases. The outcome (or sometimes just output) of it is always software in the form of a release of a

* feature
* product
* service & migration
* bugfix/improvement
* etc.

There are two more important things you need to be aware of. Does the release include broader customer communication (**external**) and/or training of different departments (**internal**)? In **B2C** that’s usually the case. In **B2B**, it can happen that you have certain (automated) release cycles and your customers want to test the software in e.g. a sandbox environment.

#### **Do you like this article so far?**

Get more practical and in-depth guides to product management principles like product discovery, product development, and product leadership.

Top of Form

Subscribe

Bottom of Form

.

### Prioritizing the product development roadmap

In order to know what type of roadmap I need to create, I start with what I call the “pre-prioritization”. That means defining the first draft based on my best knowledge after doing market research, talking to stakeholders, and first conversations with my team. There are many ways and techniques out there to prioritize features such as:

* User story mapping
* GIST planning
* Value vs. costs/risk
* MoSCoW model
* Kano model
* 100 Dollar method
* and many more

It’s not easy to define what the best method is. I tried all of the listed ones and realized it depends on the amount of time and information I have in place. After I know how priorities roughly look, I start visualizing it in some slides or a tool.

### Visualizing the product development roadmap

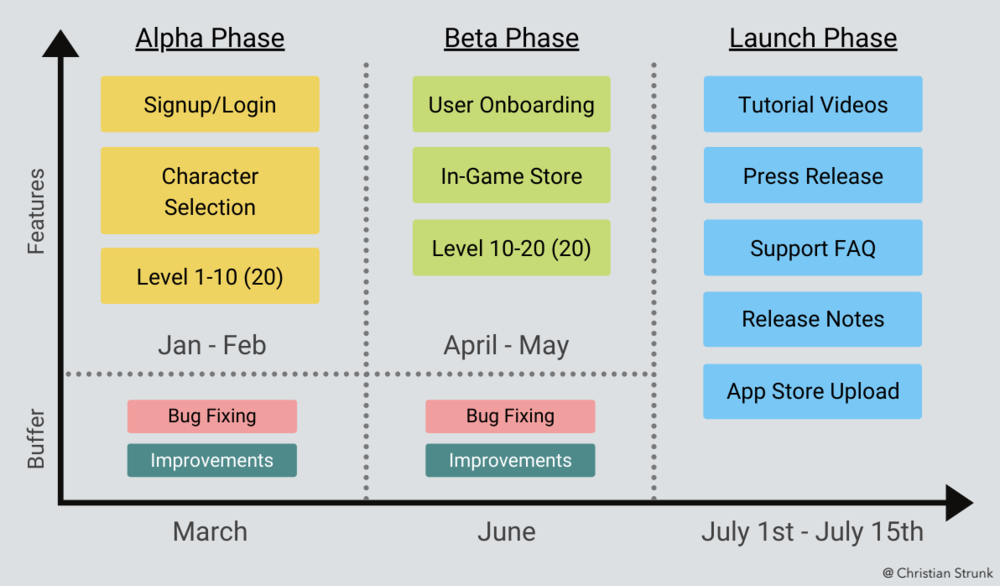
Let’s take a look at two different examples:

**1. Product launch (B2C** )

Let’s imagine we’re about to launch an iPhone game. The roadmap would contain key features (technical and non-technical) that need to be developed to finish the MVP. All features are categorized in different milestones such as an alpha and beta testing phase.

On many occasions I’ve come across Product Managers not visualizing work such as launch preparation, training (e.g. support, sales, etc.) and more. My recommendation is to always add that information as well. It helps to

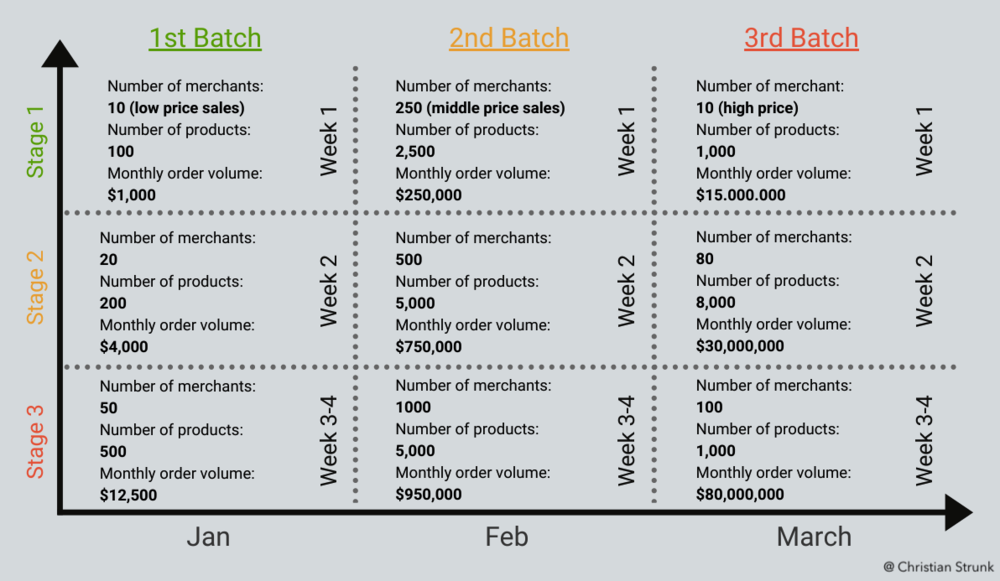
* not forget key launch preparation topics
* see the bigger picture as an Engineering team
* get a clear overview of affected departments (dependencies).



**2. Service deployment + migration (B2B)**

In this example, we’re an eCommerce SaaS platform such as “Shopify”. We’ve decided to replace the existing product catalog service (the service that stores all product data such as name, prices, tax classes, images, and more). The goal of this service is to be more robust and to scale 10x more product data. This new service will affect the whole platform and, therefore, all of its customers.

In such a case, it’s important to have a clear migration plan starting with a batch of customers that have a small number of products and order volume. From there on more and more badges will be migrated to the new service. (simplified example)



**Note:** There’re many ways to design a roadmap or migration. The main goal of this article is to “visualize” the creation process of it.

### Reviewing the roadmap with the Engineering Team

After I’ve created the first draft with a couple of high-level dates it’s necessary to share it with my team. By then my roadmap contains information about the long term goal, (business) goals, and the detailed roadmap. I share this document one day before the meeting happens so that the team can prepare. Depending on how much I’ve involved them beforehand, a meeting will go on for 30-60 minutes.

If my roadmap touches different domains and Engineering Teams, I suggest always syncing with the other Product Managers from the outset. We either pitch the roadmap individually in our team meetings or schedule a bigger session with all those involved.

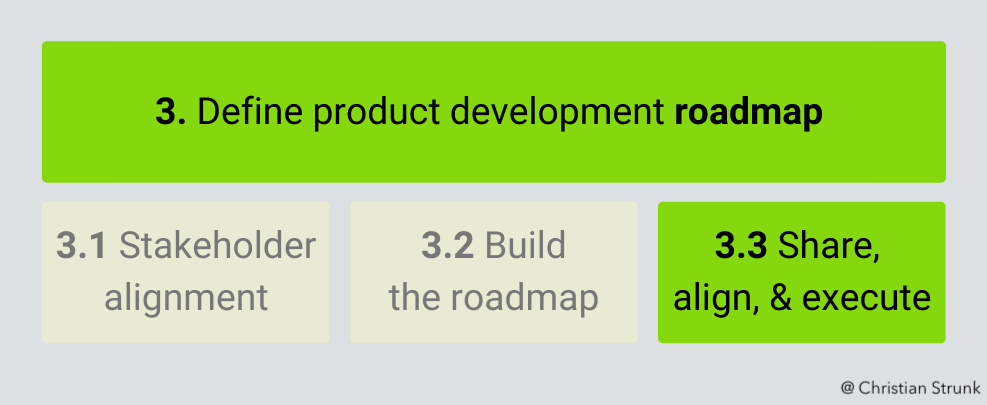
The main topics I try to cover and gain clarity on are:

* What does the team think about the roadmap?
  + Do they like the plan?
* Do the priorities make sense from a technical point of view?
* What did I forget to add (whether it’s technical or not)?
* Can we deliver everything within the proposed timeframe?

I wrote an article on how to [make meetings and grooming more engaging](https://www.christianstrunk.com/blog/backlog-grooming) if you’d like to dive deeper into the topic.

The alignment meeting usually entails adjustments and follow-ups. Therefore, it’s important to keep pushing and driving it as a Product Manager. I keep my team in the loop until we’re all on the same page.

## **Aligning and developing the product roadmap**



After the priorities are clear and additional information has been added, it’s time to let the rest of the company know. At this stage, I would have already spoken to stakeholders so they will be interested in seeing the roadmap. I either share it via email or schedule a meeting(s), but this depends entirely on the size of your company. I personally rarely needed to adjust the roadmap again. This can sometimes occur though, and often for the following reasons

* Legal changes
* Market changes
* High prio customer requests
* Highest paid person’s opinion (HIPPO)
* etc.

If you’d like to make sure that you cover all relevant stakeholders and departments I recommend reading this article which includes a [product launch checklist](https://www.christianstrunk.com/blog/product-launch-checklists).

The next big step for you and your team is to start building what’s on your roadmap. If you haven’t already filled up your backlog, it’s time to do that now. Here are some more useful links that can help you to better prepare your projects:

* [Creating user stories and epics](https://www.christianstrunk.com/blog/user-stories-and-epics-for-the-win)
* [Breaking down features for development](https://www.christianstrunk.com/blog/break-down-features)

## **Keep the product roadmap updated. Be agile!**

One important aspect we haven’t looked at is the **timeframe** of a roadmap. I believe every roadmap that is longer than 6-12 months (depending on your industry/business model) should be a product strategy. The likelihood that a roadmap longer than 12 months will stay the same and be achieved is in fast-changing times close to 0. The agile manifesto describes this very well:

“Responding to change over following a plan.”

Another related thing I’ve learned the hard way about roadmaps is:

The moment they’re created, they’re outdated!

Is that bad though? 🤔

No, it’s not! That’s the nature of it all. It’s, however, crucial to regularly review the plan and to adjust it accordingly. A monthly revisit, whether the plan is still valid or not, will help you stay on track.

How agile do you create product development roadmaps? Let’s chat on Linkedin [@christianstrunk](https://www.linkedin.com/in/christianstrunk/).

# 3. [backlog & planning](https://www.christianstrunk.com/writing/5-steps-to-make-backlog-groomings-more-engaging)

[Back](#_Anexo_Home)

# 5 steps to make product backlog groomings more engaging



Today’s the day! You (a Product Manager) have prepared your product backlog. The goals are clear, features and user stories are defined and your team’s ready for the next big thing. The kick-off usually happens in team meetings where we discuss OKRs, roadmaps, features, stories, etc,. There were times where I left a meeting with more questions and unclarity than before. Despite the fact that the idea behind backlog grooming is to bring clarity to the whole team. I’d like to share 5 steps that help me better facilitate and structure team meetings to gain clarity on imminent projects with an engaged team.

**What’s product backlog grooming?**A “backlog grooming” or “backlog refinement” is a **team meeting** where the Product Manager and the Software Engineers come together to discuss user stories in the backlog. The goal is to gain clarity on projects, tasks, and [estimate tickets](https://www.christianstrunk.com/blog/how-to-estimate-tickets) and the scope.

Usually, we meet once a week to discuss tickets in our backlog or other topics like the roadmap or quarterly goals. The frequency of these sessions depends on how well-groomed our backlog is. If we kick off a new, bigger project, I spend more time with my team to get everyone aligned as fast as possible. If we’re in the middle of a project that is clearly defined, we meet bi-weekly.

A very important thing I always do one day (at least 24h) before every grooming, is share a short agenda. This contains at minimum a list of stories/tickets I want to discuss. I always ask my team to add additional points or stories to the agenda. It’s very important to not forget about the work that needs to be tackled outside of my little product world 🙃

The better you are aligned on running and upcoming projects, the more time you save in meetings and during implementation. Let’s look at the rules that helped me and my teams to be more structured.

## **#1 Do a check-in**

“Never change a running system!” - Anonymous

Unfortunately, you can’t improve and change if you don’t try out new things. I regularly change the agenda of the backlog grooming sessions. Not in a way that I do completely new crazy stuff or try to reinvent the wheel. It’s more about making tiny changes within a given structure.

We always start with a quick check-in. I ask how everyone is doing and how we are feeling today. Before I start talking about something else I ask if anyone has any particular topics they want to discuss, or any feedback on the agenda or other pressing topics. It’s important for your team members to arrive at the grooming. If you were fixing a bug before or worked on something very complex, you need some time to make the context switch.

There are multiple ways to get started. Another way to do a check-in is to ask person by person what’s on their mind (30sec - 1min). That can help you to better understand what the mood of the team is. Based on that, you can adjust your personal agenda if needed.

You can find many other check-in questions on the internet. The main message is to remind ourselves that we’re all humans. A quick “health check” before starting a meeting helps me to get everyone in the mood to sit together and discuss particular topics.

### What about time 🤔

In my opinion, the time you spend is very well invested. If you figure out during the check-in that someone is struggling with a big issue in the live system which causes, for example, app crashes or breaks the website, you should spend time discussing that instead. If you know that someone isn’t in a good mood or has another issue you can dive into that. The productivity boost when everybody is in meeting mode shouldn’t be underestimated. People are simply more engaged if they have the ability to speak up and if everyone is listening.

If you have people's attention it’s’s time to get started...

## **#2 Kick it off! Seriously!**

After we’ve checked in and everyone’s taken a sip of coffee or Club Mate, we start. Before we talk about any story, I kick things off by sharing the current plan. This can be the

* Product roadmap
* OKRs (Objectives and key results)
* Company goals or vision
* Project status and milestones
* Current sprint goal
* etc…

Sharing the priorities and goals at the beginning is very crucial to (again) help catch everyone up. As mentioned above: If you have fixed an urgent issue or just stopped working on a complex task you might forget about the global priorities. One of the key tasks of every PM is repeating, repeating, repeating. Every day, every week, every month…

If I’m sometimes in a rush and forget to share our OKRs, I get a friendly reminder from my teammates to open the slides and have a look at them.

I present the slides and read out every objective and key result. Before I move to the next objective, I ask for questions and if the team think we’re on track. I recommend asking open questions in order to make people engage with the topic. You can ask things like...

* How do we feel about the current progress?
* Does anyone foresee potential obstacles or delays?
* Where do we stand currently?
* Are our goals still clear?

Instead of asking

* “Any questions? (1.5 seconds later) No? Ok, let’s continue….”

I always take time to answer questions. If I have the feeling we’re losing track or there are more things to discuss, I propose to schedule a follow-up or discuss it “offline” (which means after the meeting).

Checking in and kicking off the grooming should take no longer than 5-10 minutes. From there on, I start presenting the backlog.

#### **Do you like this article so far?**

Get more practical and in-depth guides to product management principles like product discovery, product development, and product leadership.

Top of Form

Subscribe

Bottom of Form

.

## **#3 Present the playground (backlog)**

The playground I’m talking about is the backlog. I used to work with digital backlogs which means I present it on a big screen (if we are all in one room) or I share it via my screen on hangout, zoom or any other tool. Before we start the deep dive, I spend a couple of minutes presenting the stories in the backlog.

Since I’ve prepared my backlog with [epics and user stories](https://www.christianstrunk.com/blog/user-stories-and-epics-for-the-win) I give my team an overview of the tasks that I’d like to discuss. This is usually the time where questions pop up or team members have other tasks they want to discuss. If needed, we rearrange the order of tickets to be groomed and then we start with the first story from the top.

In a perfect world, everyone has read the tickets before and is prepared.

Sometimes people aren’t able to prepare for a grooming for a multitude of reasons. If you notice that this happens on a regular basis with more of your team members, I recommend addressing it and finding a better way to make sure that the team prepares for the grooming. Retrospectives can be a great time to discuss these kind of topics.

Bear in mind that the backlog is the place you all use to plan and work with 🤝

## **#4 Deep dive into every user story**

After picking a story we haven’t discussed, I spend a couple of seconds to reading it out loud to the team. Yes, I do that with every story. Alternatively, you can give everyone a minute to read. I like the old fashion way of reading them out loud. The next action is to ask for feedback and open questions. This is actually the time where you as PM receive questions or people start discussing technical details.

Some tickets need more time and others need less. I always set a timebox of 10 minutes. It can sometimes take longer, although, this is an indicator that more preparation is needed. Based on the feedback and outcome of the conversations, we adjust or break it down into more tickets. By “we” I mean either me or someone else who is sharing the backlog. If we have a backlog with a lot of tech stories, one of the engineers presents the backlog, takes notes and adjusts the tickets.

Sometimes I get asked by teammates to let them facilitate the whole grooming including the steps before the session. I really recommend letting Engineers play the PM role as well. At the end of the day, we’re one team with one goal and everyone plays an equal part. It’s important to change perspectives and as mentioned before, you should constantly try out new things 😉

After a ticket has been enriched with all the relevant information, we estimate it according to the Fibonacci format. That’s the last missing piece of the puzzle that’s needed for a ticket to match our “definition of ready” and to be “ready for development”.

That was the happy part… What if a story needs more specification?

## **#5 Define clear follow-ups**

Even with a timebox of 10 minutes per story, we spend an average of 5 min per task. As mentioned above, longer discussions are an indicator that more clarification and preparation is needed. That doesn’t mean that you need to cut discussion off after 10 minutes. Sometimes it’s very important and healthy to have longer discussions because they can relate to bigger topics or more stories.

Regardless if a story needs more preparation, follow up tasks need to be created. Something else I recommend is always defining clear follow-up actions. This can happen naturally, depending on how experienced and how well your team works together. Sometimes though, teams struggle to define follow ups and assign to do's. In this case, I believe the Product Manager can guide the team.

**Example:**

You’ve discussed a story and identified that it needs to be broken down into 2 more tech-stories. You all agreed that this needs to happen by the next grooming.

And... what’s next?

If no particular person takes over the task (and you know that it might not be prepared for the next grooming) you can always ask who wants to do it.

Instead of asking broad questions, let’s get specific. If one or more people committed to taking over the task you can even ask them when they think it’ll be ready.

In a nutshell I focus on the

1. “What” (Action item)
2. “Who” (Owner)
3. “When” (ETA)

Sounds pushy? 🙊

It does... kind of...

At the end of the day, it is the team’s responsibility (including the PM) to prepare and plan features. As a Product Manager, you always strive for the most granular planning in the shortest time. You’re not assigning tasks to people and giving them deadlines. The proposed method is a collaborative way of achieving results and empowering the team to take more ownership.

What are your tips and tricks and how do you shape your team meetings?

Let’s have a chat on Linkedin [@christianstrunk](https://www.linkedin.com/in/christianstrunk/)﻿.

# 4. [execution & product launch](https://www.christianstrunk.com/writing/product-launch-checklists)

[Back](#_Anexo_Home)

# Product launch checklists: How to better plan your releases



Do you ever get the feeling after releasing a new product or feature, that you may have forgotten something? Do you ever think you’ve thought absolutely everything through, only to realize you haven’t.? If you can relate, then I recommend continuing reading. I wrote this article based on my experience of failed and successful product releases.

With growing experience as a Product Manager, I’ve learned to work more efficiently. Writing a user story as a senior took me way less time after I’d done my homework, research, etc. than as a junior. Ok, this doesn’t count for everyone but I’m sure it does for you. 😉

Formalizing certain processes really helped me to stay on top of things. One example of this is the product launch checklist. In the following, I’ll describe how I prepare and communicate an upcoming product or feature launch. You can use my **free downloadable checklist** as a template and adjust it to your companies needs.

## **Product launch presumptions**

A successful product launch starts with good planning. That’s why this article presumes that the product or feature that’ll be released has been

* researched (market & competition)
* validated based on data
* prototyped and tested
* prioritized accordingly
* and more

This list isn’t complete and not every point needs to be covered. That depends on how your company works.

Next, I’d like to focus on two important steps for a product release:

1. Preparation
2. Communication

**Note:** Both parts are covered in the free product launch checklist.

## **1. Preparing your next product launch**

There are a lot of different products out there. I’ve launched many backend services, web applications, and (my favorite) apps. That’s why I’ll focus on examples using these 3 areas. The “preparation part” of the product launch checklist is also applicable to other products and services such as hardware.

**Example:**

Let’s imagine we’re working in a B2B2C insurance company. This company offers its customers the best insurance deals via an app and over the web. The app was launched in 2008 and design-wise never touched again.

**Guess what:** We’ll redesign it. 🎨

The project has been validated and we know it’s the right thing to do. Now we’re moving close to the launch date (approx. 4-8 weeks).

### Product launch documentation

The product launch documentation covers all relevant information around the product. It’s written for internal use and can be re-used as a template for external material.

**Product documentation (internal)**

I write product documentation in a way that it can be used internally & parts of it externally. The points I cover are:

* Feature description & goals
* Business & customer value
* Research & validation
* Functionality & usage
* Comparison with the old functionality (optional)
* Planned rollout timeline (e.g. staged rollout, customer target groups, beta phase, etc.)
* Links to other sources (e.g. Jira board, design files, feature comparison)

Depending on your work and communication style you can add more or fewer points to the documentation. I believe it’s important to have a **source of truth** that allows everyone to access and find key information.

If you’re interested in “how to write product documentation”, I’ll be writing an article about that in the near future.

**Comparison sheet** **& matrix**

Looking at our redesign example, we’ve changed many screens in the existing app. Therefore, we’ve created a comparison sheet (e.g. Confluence document) that shows each screenshot with a before and after. I always highlight the main changes especially when functionality has been moved or changed. Creating a document like this can also be very helpful for the Customer Support department. At the same time, your support and operations department can reuse the document for external communication such as support centre articles.

In the case of developing a completely new feature, it’s very helpful to create, for example, a feature comparison/matrix. This can help the Sales and Marketing teams to better present and sell the new feature or product.

**Frequently Asked Questions (FAQ)**

How can you create FAQs before launching a product? Initially, I build this based on assumptions and questions people ask internally. Sometimes you decide to make trade-offs when you redesign a product. Maybe even functionality changes or moves to other places. And yes… it’s not always 100% self-explanatory (even if it should be). Mentioning things in the FAQ section can be very helpful for the Customer Support Team and later on for your customers.

If you build a new feature or product FAQ’s can be helpful to make customers convert. It’s not only about functionality and how to’s. The better you educate your customer the more likely they are to buy-in.

Besides documentation, it can also be very important to educate and train people internally. I recommend doing this for bigger launches in particular.

**Note:** There are multiple ways to document product launch related material like videos and tutorials. The list isn’t complete. My intention is to show some possible options.

#### **Get the free product launch checklist!**

Less struggle & **more structure** during your next **release**.

Top of Form

Download PDF

Bottom of Form

.

### External launch material

Since we’ve done a lot of work covering the internal materials it’ll now be even easier and faster to prepare the external product launch material. I recommend to not simply copy and paste. The language you use with your customers is usually very different from the language you use internally. Do I need to mention not publishing confidential and competitive data? 😉

**Release notes**

The release notes in the App Stores should contain all relevant changes. A mistake I see very often in release notes is, that they’re written in a very “non-human” way. Please always keep in mind that you’re delivering more/new value to your customers. Instead of calling it “bug-fixes” you can spend 2 more minutes explaining what you’ve fixed and why. Don’t overcomplicate things. Clarity always wins.

**Website, landingpages, emails**

**Marketing!** Do I have your attention?

Depending on your company size and structure you might be more or less involved in the marketing of your product or feature. And as a Product Manager, it’s important to stay on top of all marketing activities. I always highlight the USPs, main “selling” points, and customer value in the product documentation. This document/material can then be shared with the Marketing Team. Alternatively, you can create a separate “marketing/sales” document. You as a Product Manager should know what are good ways to interact with your customers and what aren’t. Don’t be afraid to share that knowledge with your Marketing Team. In the past, I’ve assumed that they’re aware of these insights… and I’ve often been wrong. The bigger the organization the more likely it is that the opposite is true.

**Support center articles**

Based on your product documentation, before/after comparison, and FAQ’s support center articles can be created easier. If you don’t create these yourself, get in touch with e.g. the Operations Team who will create it. Good communication, exchange, and reviews always helped me to make the best out of every customer phasing documentation.

Having good product documentation in place is the first step. Next, we’ll have a look at when and how to communicate the upcoming product launch (redesign).

## **2. Product launch strategy: timing & communication**

What’s most important is to make sure that all key stakeholders are updated and aware of the (in our case) redesign launch. Here you can read more information and best practices around [stakeholder management](https://www.christianstrunk.com/blog/how-to-manage-stakeholders). I’d like to highlight the importance of really updating all of them. Over my career, I’ve seen too many occasions where teams and Product Managers launch features without letting key stakeholders know. That’s a **no go**! 🚫

### Product launch communication: Whom to update when?

Informing all departments a week before the launch is too short term. The bigger the project is the more time in advance I inform the stakeholders. Let’s have a look at some key departments to update.

**Marketing Team**

The Marketing Team needs to know everything about the product and how to market it. Ideally, you provide them with screenshots or give them access to the beta version. If the design isn’t finished, stay closely in touch with them and provide them with regular updates. Since they need to create landing pages, update the website, and more I’d inform them at least **8 weeks** before the launch.

**Note:** I’m talking about product **launch** communication. Every stakeholder should know that you’re working on that big feature from day 1.

**Customer Support Team**

Launching a product without an informed Customer Support Team can be suicidal. At the beginning of my career, we launched (what I thought was) a tiny feature without informing Customer Support. **The** **result:** The customers weren’t able to use the old functionality, we deployed a huge bug, and hundreds of people were calling us. I can tell you my boss was everything other than happy about that. That’s why I always update the CS Team at least **6 weeks** before and start preparing with them the launch.

**Sales Team**

It doesn’t matter if you’re redesigning an app or building a new feature, sales want to know. I’d start communication with the Sales Team around **4 weeks** before launching. It’s important to share the documentation and to do a presentation of the product. Workshops can be very helpful to build a common understanding and talk about product and feature capabilities. A feature matrix and competition comparison can also be very helpful for Sales Teams. Based on the questions and requests from the Sales Team, I recommend summarizing them in an email. I’ve often heard that salespeople like to sell more than the product actually can. 😉

**Technical Business Development Team**

If your Bizdev Team works with partners and your internal SDKs and APIs, I’d get in touch with them at least **4 weeks** in advance too. Sharing technical documentation upfront is very important. For meetings like this, I always add a Senior Developer from my team to talk about the technical changes and do Q&A sessions (questions & answers).

### Internal product launch training

The mentioned departments are some examples of how and who to communicate within the run-up to a product launch. If your company is bigger, you might have more departments and people to communicate with. In smaller companies, you can simply bring all the involved people into one room.

I’d like to share three more ways to update bigger groups of people regardless if they’re all part of one department or different ones.

**Internal launch email**

It’s always helpful to announce an upcoming launch (pre-launch mail) and to send an email after the (successful) launch (post-launch mail). If you use other tools like slack instead of email you can use them instead.

**Product Demo Sessions**

Some companies have monthly product demo sessions where all employees can either test the product(s) live in a room or via a call where teams present what they’re working on. Or even better: what they’ve finished. 🚀

**AMA (Ask me anything)**

These sessions are great for “interest groups”. People and departments who’ll be affected will voluntarily join you. It’s an open meeting without much of an agenda. Simple questions and answers. Be prepared. They ask good and tough questions.

## **Checklist Download & Summary**

I’ve created a checklist for the material preparation as well as the communication for each department.

Managing the product backlog and prioritizing them based on changing requirements

Drive product enhancements that increase customer value, create efficiencies and grow savings and quality.

Scrum

Uipath

-------------------------------------------------------------------------------------------------------------------------------<https://intel.wd1.myworkdayjobs.com/en-US/External/userHome>

# Job Details:

## **Job Description:**

ACE is the Automation Center of Excellence:

and plays a critical role in setting and executing the strategy for Automation in HR Global Services. Through our unattended and attended automation products, and citizen development automation programs, we provide automation services and solutions that meet the highest corporate quality standards of efficiencies and maintain industry best-in-class practices that cover processes for all HRGS product lines and impact all Intel employees worldwide.

The RDA Product Owner is responsible for managing the entire RDA product process within the HRGS organization.

The Product Owner will research and determine the organization's needs and identify which features to include from vendor releases as organizational capabilities.

The Product Owner will use bots to pitch ideas for product features to stakeholders and evangelize solutions to the Intel Automation and HR Automation Champions community.

The Product Owner will collaborate closely with the IT platform engineering team and other organizations to ensure the products meet specific technical and business capabilities requirements as well as drive to continually improve the products and product governance processes.

The Product Owner will be the final escalation level and oversee all HR RDA technical personnel and capabilities.

**In this role you will help Intel with the following, but not limited to:**

* Serves as the liaison between business units, technology teams, and business citizen developers
* Collaborating across the HR organization and with other organizations to understand and anticipate their needs and translate them into technical product requirements and create models to enable them as organizational capabilities
* Defining the vision for the team's product and analyzing tradeoffs between technical usability and business requirements needs
* Creating a product road map based on this vision
* Critically evaluate information gathered from multiple sources, reconcile conflicts, decompose high-level information into details, abstract up from low-level information to a general understanding, and distinguish user requests from the underlying true needs
* Proactively communicate and collaborate with external and internal customers to analyze information needs and functional requirements and deliver artifacts as needed
* Managing the product backlog and prioritizing them based on changing requirements
* Overseeing all stages of product creation including design and development
* Monitoring and evaluating product progress at each stage of the process
* Drive product enhancements that increase customer value, create efficiencies and grow savings and quality
* Understand and advocate best practices in high-level system designs
* Gather customer requirements and advise on product features and strategic direction
* Present and demonstrate the solution to external groups and customers as necessary
* Work with various teams to understand all local and international regulations that impact product requirements

## Qualifications:

You must possess the below minimum qualifications to be initially considered for this position. Preferred qualifications are in addition to the minimum requirements and are considered a plus factor in identifying top candidates.

**Minimum Qualifications:**

* Bachelor's degree in Computer Science, Systems Engineering or related field.
* 3+ years of experience in Robotic Process Automation.
* UiPath / SCRUM/Agile certification
* Fluent English communication skills both oral and in writing.
* Must have permanent-unrestricted right to work in Costa Rica.

## **Job Type:**

Experienced Hire

## **Shift:**

Shift 1 (Costa Rica)

## **Primary Location:**

Costa Rica, San Jose

## **Additional Locations:**

## **Business group:**

Intel's Human Resources group is responsible for hiring, developing and retaining the best and brightest employees while continuing to strengthen the company's culture and values. Intel Human Resources provides first-rate, cost-effective services and support to employees worldwide and is dedicated to advancing Intel's business goals.

## **Posting Statement:**

All qualified applicants will receive consideration for employment without regard to race, color, religion, religious creed, sex, national origin, ancestry, age, physical or mental disability, medical condition, genetic information, military and veteran status, marital status, pregnancy, gender, gender expression, gender identity, sexual orientation, or any other characteristic protected by local law, regulation, or ordinance.

## **Position of Trust**

N/AInside this Business Group  
  
Intel's Human Resources group is responsible for hiring, developing and retaining the best and brightest employees while continuing to strengthen the company's culture and values. Intel Human Resources provides first-rate, cost-effective services and support to employees worldwide and is dedicated to advancing Intel's business goals.

RDA bots can strengthen compliance with the company’s policies and procedures or regulatory directives by prompting employees to read disclaimers and follow process flows. They help ensure that employees fulfil all compliance requirements in a task or process before moving to the next action.

Graphical user interface, application

Description automatically generated

Table

Description automatically generated

COE Centralized,divional, federated

<https://www.youtube.com/watch?v=mcq4GNwt-J8>

The product manager or product marketing manager studies the customer’s wants and needs, whereas the product owner makes sure that product development is following the product roadmap. The product manager decides what is going to be built or adapted and the product owner makes sure the development team does just that.