STRATEGIZE

Product Strategy and Product Roadmap Practices for the Digital Age

SECOND EDITION

ROMAN PICHLER

Strategize: Product Strategy and Product Roadmap Practices for the Digital Age, 2nd Edition

Roman Pichler

Copyright © 2022 Roman Pichler. All rights reserved.

Published by Pichler Consulting.

ISBN 978-0-9934992-4-1

Design: Ole H. Størksen
Layout: Booknook
Cover photo: Shutterstock

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book and the publisher was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

The author and publisher have taken care in the preparation of this book but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, please write to info@romanpichler.com.

TABLE OF CONTENTS

Preface 9
- Why You Should Read this Book 9
- A Brief Guide to this Book 10
  - Book Structure and Contents 10
  - Sources, Examples and Terms Used 10
  - Changes in the Second Edition 11

Introduction 13
- A Product Strategy and Roadmap Model 13
  - The Elements of the Model 15
  - Integrating Different Plans and Planning Levels 16
  - Strategic Planning, Agility, and OKRs 19
- Empowerment, Collaboration, and Strategic Decisions 20
- Four Product Success Factors 21
- A Product Strategy Process 24
  - Timeboxed Strategizing 24
  - Continuous Strategizing 26
- Digital Products and Product Strategy 29

Product Strategy Foundations 31
- Understand What a Product Strategy Is 31
  - The Product Strategy Defined 32
  - Bootstrapping a Product Strategy 33
- Think Big and Describe Your Vision 34
- Use the Business Strategy to Guide Strategic Product Decisions 37

Understand Your Product's Innovation Type 39
Core Innovations 40
Adjacent Innovations 41
Disruptive Innovations 41
Summary 43
Take Advantage of the Product Life Cycle Model 45
Development 47
Introduction 49
Growth 52
Maturity 54
Decline 57
Summary 58
Collaborate with the Stakeholders and Development Teams 59
Stakeholder Identification 59
Stakeholder Analysis and Engagement 59
Collaborative Workshops 62
Product Strategy Development 65
Segment the Market 65
Segmenting by Customer Properties and Benefits 66
Choosing the Right Segment 68
Find an Itch That's Worth Scratching 70
Establishing the Right Mindset 70
Discovering a Need 71
Using Empathy and Consumption Chain Maps 73
Making the Need Specific 76
Selecting a Primary Need 78
Describe Users and Customers with Personas 79
A Persona Template 79
Tips for Creating Effective Personas 80
Make Your Product Stand Out 82
The Strategy Canvas 82
The Kano Model 85
The Eliminate-Reduce-Raise-Create Grid 87

Capture Your Strategy with the Product Vision Board 89
Let the Vision Guide You 90
Focus on a Specific Market or Market Segment 90
Clearly State the Main Problem or Benefit 91
Describe What Makes Your Product Special 91
Capture the Desired Benefits for the Business 92
Make your Product Vision Board Testable 93
Use a Collaborative Workshop to Create the Board 93
Complement Your Strategy with a Business Model 93
Sample Business Models 94
Capturing the Business Model 94
Using the Business Model to Create a Business Case 97
Consider Your Product's Ethicality 97
Users First 97
Fair Business Model 98
Right Design and Technology Choices 98
Environmental Impact 99
Develop Variants and Unbundle Your Product 100
Benefits 101
Pitfalls to Avoid 102
Product Strategy for Variants und Unbundled Products 102
Take Advantage of Software Platforms 103
Benefits and Drawbacks of Software Platforms 103
Platform Tips 104
Create a Product Bundle 106
Benefits 106
Pitfalls to Avoid 107
Product Strategy Validation 108
Iteratively Test and Correct Your Strategy 108
Carry Out the Minimal Upfront Discovery Work 110
Involve the Right People 111
Use Data to Make Decisions 112
Turn Failure into Opportunity 114

Get Out of the Building 116
Identify the Biggest Risk 117
Determine the Key Risks 117
Select the Biggest Risk 119
Wash, Rinse, and Repeat 120
Choose the Right Validation Techniques 120
Directly Observe Users and Customers 120
Carry Out Problem Interviews 121
Use Product Fakes 124
Build Spikes to Assess Technical Feasibility 125
Don't Rely on a Single Method and Separate Data Analysis from Data Collection 126
Pivot, Persevere, or Stop 127
Review and Analyse the Data 127
Take the Right Action 128
Plan and Track the Validation Work 129
Timebox the Work 129
Use a Kanban Board 130
Use Stand-up and Weekly Review Meetings 131
Product Strategy Reviews 133
Choose the Right Key Performance Indicators (KPIs) 133
Focus on Needs, Business, and Product Goals 134
Use Health Indicators 134
Set Realistic Targets 135
Combine Quantitative and Qualitative KPIs 136
Take Advantage of Trends 136
Leverage Lagging and Leading Indicators 137
Regularly Review and Adapt the KPIs 137
Avoid These Common KPI Mistakes 138
A List of Sample KPIs 139
Regularly Review and Update the Product Strategy 141
Five Factors 141
Review Frequency 142
Four Choices 143
Combined Strategy and Roadmap Reviews 144

Product Roadmap Foundations 146
Understand the Benefits a Product Roadmap Can Offer 146
Take Advantage of Goal-oriented, Outcome-based Product Roadmaps 148
Practise Collaborative Product Roadmapping 150
Base Your Roadmap on a Validated Product Strategy 151
Get the Relationship Between the Roadmap and the Product Backlog Right 152
Distinguish Internal and Public Product Roadmaps 153
Avoid These Common Roadmapping Mistakes 154
Stakeholders Determine the Roadmap Contents 154
The Roadmap is Seen as a Fixed Plan 154
The Roadmap is Speculative 155
The Roadmap Results in a Death March 155
The Roadmap Contains Epics and User Stories 156
The Roadmap is Mistaken for a Release Plan 156
Product Roadmap Development 158
Take the Right Steps 158
Capture Your Roadmap with the GO Template 160
Set the Right Product Goals 161
Derive the Product Goals from the Needs and Business Goals 161
Use KPIs to Determine the Product Goals 162
Employ Single Product Goals 163
Don't Mistake Features for Goals 163
Make the Goals Measurable 164
Prioritise the Product Goals 166
Semantic Dependencies 166
Cost of Delay 167
Inter-Product Dependencies 169
Get the Features on the Roadmap Right 170
Determine Dates 171
Window of Opportunity 172
Steady Release Cadence 173

Estimate Cost Top-Down 174
Bottom-up vs Top-down 174
Flexible Budget 175
Fixed Budget/Development Team 175
Balance Product Goals, Dates, and Cost 176
Iron Triangle 176
Primary Success Factor 177
Secondary Success Factor 178
Fixed vs Changing Success Factors 179
Derive the Product Backlog from the Roadmap 179
Align Related Products with a Portfolio Roadmap 181
Leverage Collaborative Workshops 183
Product Roadmap Reviews 185
Track the Development Progress 185
Inspect and Adapt the Roadmap 187
Review Factors 188
Review Frequency 189
Involve the Right People and Hold the Individuals Accountable 190

Epilogue 193
Acknowledgments 195
About the Author 197
References 199

PREFACE

Progress is impossible without change, and those who cannot change their minds cannot change anything.
George Bernard Shaw

Developing a successful product is not down to luck, a stroke of genius, or just trying hard enough. While these factors are undoubtedly helpful, product success starts with making the right strategic decisions. A challenge for many product managers, product owners, and other product people is that we are often so preoccupied with the tactics—be it writing new user stories or dealing with an urgent support request—that we sometimes no longer see the wood for the trees. In the worst case, we take our product down the wrong path and end up in the wrong forest; we've perfectly executed the wrong strategy and are left with a product that underperforms or even bombs.

Why You Should Read this Book

This book will help you play a proactive game, make the right strategic decisions, use them to guide execution, and help you maximise the chances of creating a successful product. It explains how to create an inspiring product vision, a winning product strategy, and an actionable product roadmap for digital products that are developed using agile practices. The book offers a wide range of carefully selected, proven techniques and tools that complement and support each other. If you work as a product manager, Scrum product owner, product

portfolio manager, head of product, or product coach, then this book is for you.

A Brief Guide to this Book

Book Structure and Contents

This book begins with an introduction followed by four chapters on product strategy and three chapters on product roadmaps. While you might be tempted to go straight to the sections you are most interested in, I recommend reading the introduction first. It presents central ideas, and it truly forms the foundation for the remainder of the book. Reading it will provide you with context to correctly understand the individual chapters and sections. While I have tried to write the book so that the chapters are loosely coupled, there are dependencies between them. Most notably, the roadmap chapters build on the strategy ones, as a realistic and actionable product roadmap should be based on a validated product strategy.

Please note that this book is neither an introduction to product management nor to agile practices—there are other great books that equip you with this knowledge. Instead, I assume that you are already familiar with central product management concepts and an agile framework like Scrum.

Sources, Examples and Terms Used

This book draws on a range of approaches including Lean Startup, Customer Development, Design Thinking, the Blue Ocean Theory, Business Modelling, Jobs to be Done, User-Centred Design, Lean Management, Scrum, and Kanban—in addition to my own methods and templates. I have taken great care, though, to combine the different concepts, techniques, and tools in such a way that they fit together and complement each other.

Most of the examples in this book are taken from the consumer space. The reason for this is simple: I have tried to use products that I hope you have heard of. But the practices covered are applicable to virtually any digital product, including B2B offerings. You can also apply most of my advice to non-digital products. You might have to adjust some of the practices, though, and ignore the software-specific recommendations.

Finally, I refer to individuals who manage a product as *product person* and *product people* rather than *product manager* or *Scrum product owner* to avoid any negative connotations some readers might associate with these terms.

Changes in the Second Edition

This edition of *Strategize* updates and enhances the first one, which was published in 2016. It offers new concepts, adds new materials, shares more tips, provides new examples and illustrations, expands and details existing content, and simplifies the book structure. I believe that this has made the book even better and its advice easier to apply.

INTRODUCTION

If you don't know where you are going, you'll end up someplace else.
Yogi Berra

My first product management experience wasn't exactly what you call a success: I was working as part of a team who was called in to help with a new product development effort, and I ended up working with the product manager leading the product. While I learnt a lot in the process, the resulting product sadly failed. But this taught me an important lesson: There is no point in worrying about the product details and writing use cases and user stories if a sound product strategy is missing and an actionable product roadmap is not available. Strategy and roadmap are truly fundamental, as I explain in this chapter.

A Product Strategy and Roadmap Model

An effective product strategy and roadmap are crucial to successfully create, enhance, and manage a product. But what exactly is a product strategy and what is a product roadmap? How do the two plans relate? And what's their relationship to the product vision and the product backlog? To answer these questions, I have developed the model shown in Figure 1 (on the following page).

Let's take a closer look at the model and explore its elements and connections in the next few paragraphs. Please note that I'll discuss the detailed practices to work with the artefacts in Figure 1 in the remainder of this book. The following sections therefore provide an overview.

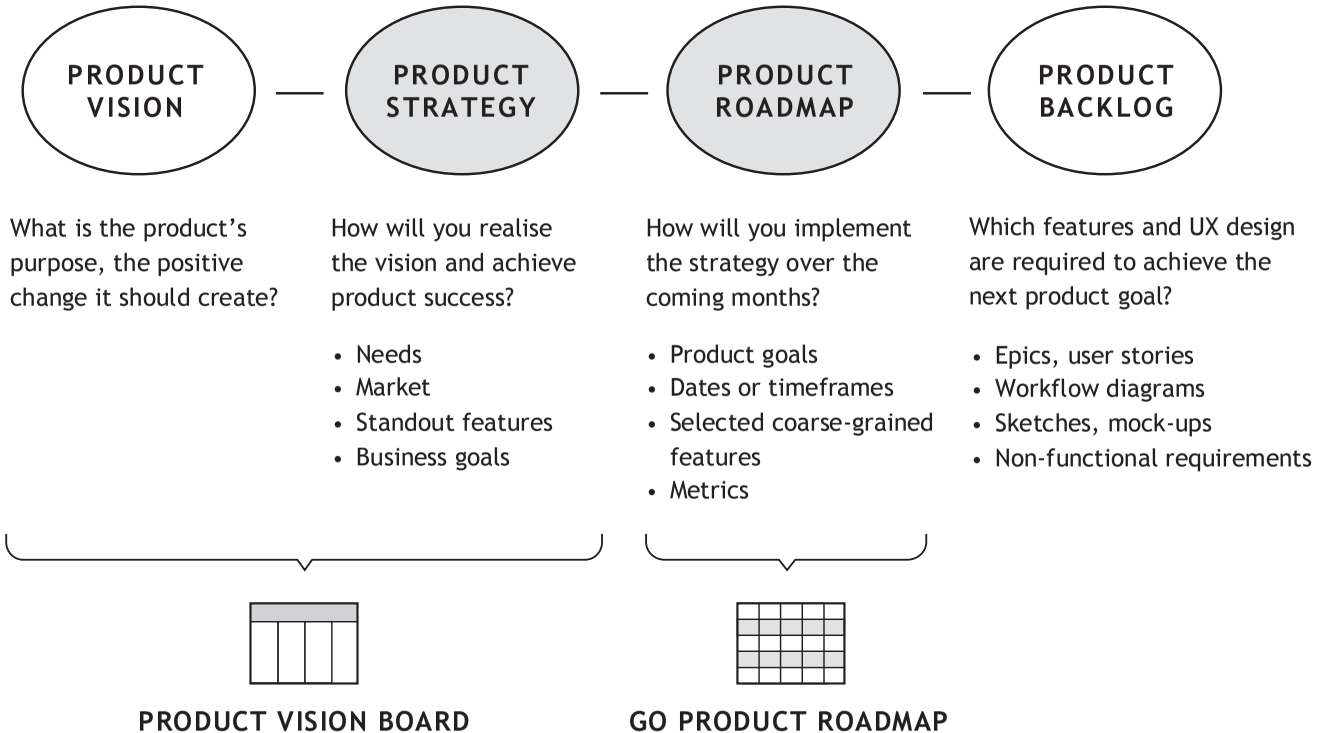


Figure 1: A Product Strategy and Roadmap Model

The Elements of the Model

At the heart of the model in Figure 1 are four artefacts: the product vision, the product strategy, the product roadmap, and the product backlog. The *product vision* describes the product's purpose, the ultimate reason for creating it, and the positive change it should bring about. You can think of the vision as the product's true north that guides and aligns everyone involved in achieving product success. This includes the stakeholders, the management sponsor, and development teams.[1]

The *product strategy* communicates the approach chosen to realise the vision and to make the product successful. Coming up with a strategy requires you to make four important choices: selecting the needs the product should address; determining the market or market segment—the users and customers who should benefit from the product; choosing standout features that set it apart from competing offerings; and setting realistic business goals that describe the benefits the product will create for the business. Making these choices requires you to say no to ideas and suggestions. While this can be tough, it is a necessary part of strategic decision-making. A product that tries to please everyone risks not doing a good job for anybody. What's more, a new or significantly changed strategy must be *validated* to maximise the chances of achieving product success. This is best done by systematically addressing its biggest assumptions and risks.

With a validated product strategy in place, you are in a great position to build an actionable product roadmap. The roadmap describes how the product strategy will be implemented in the next six to twelve months; it communicates the specific benefits the product will achieve; and it aligns and guides the stakeholders and development teams. An effective product roadmap is built on *product goals*, which describe the outcomes the product should create, for example, acquiring new users

---

1 I use the term *development team* in this book to refer to a cross-functional, self-managing group whose members design, architect, implement, test, document, and deploy a digital product.

and increasing engagement.[2] The roadmap may also contain additional elements like *dates* or *time frames*, selected coarse-grained *features*, and *metrics*. A date or time frame states when a goal should be met, the features sketch the output required to achieve a goal, and the metrics help you understand if a goal has been accomplished.

A product roadmap that is built on product goals provides a great basis for deriving a product backlog and making the right tactical product decisions. You can simply copy the next product goal into the backlog together with its features. Then add further items that are required to meet the goal, such as epics, user stories, workflow diagrams, sketches, mock-ups, and non-functional requirements (NFRs).

In addition to the four elements described above, my model offers two templates that I have created, the *product vision board* and the *GO product roadmap*. The former helps you capture the product vision and strategy; the latter helps you communicate a product roadmap that is based on product goals and outcomes. Note that you don't have to use the templates to take advantage of the model in Figure 1. You can happily use alternatives, provided that you clearly state the needs and business goals in your strategy and that you work with an outcome-based, goal-oriented roadmap.

Integrating Different Plans and Planning Levels

Having a product strategy and roadmap is great. But how do you ensure that the two plans are systematically connected with each other as well as the product vision and the product backlog? The model in Figure 1 achieves this in the following way: The vision guides the strategy; the strategy forms the basis for creating an effective roadmap; and the roadmap finally directs the product backlog.

---

2 In the first edition of this book, I used the term *release goals* instead of product goals. But as not every release necessarily creates tangible value for the users and the business, for instance, a smaller bug-fixing release, I have changed the name. This brings the term in line with my more recent work (Pichler 2020) and the Scrum Guide 2020. The latter recognises product goals as part of the Scrum framework.

With each step, the decisions you take become more specific; and higher-level goals are translated into increasingly detailed and focused ones. The vision is transformed into needs and business goals. From these, product goals are derived, which, in turn, guide the discovery of the right product backlog items. Additionally, the time frames become progressively shorter—from five to ten years covered by the vision to a product backlog that contains items for the next few months.

While the model suggests that product planning starts with vision setting, the connections between its elements are *bidirectional*. This means that bigger product backlog changes can trigger product roadmap modifications. For example, one or more goals on the roadmap might have to be adjusted, or the dates and time frames might have to be corrected. Similarly, larger product roadmap updates can lead to a product strategy change. You might find, for instance, that one of the business goals is unrealistic, that the needs have to be adjusted, or that a standout feature has to be reworked. Finally, if you can't find a validated product strategy, then you'll have to change or abandon the product vision.

Strategy and execution are therefore systematically connected in the model in Figure 1. Strategic decisions guide the implementation of product backlog items, and insights from the tactical work lead to changes in the product roadmap and strategy. This ensures consistent decisions, and it avoids a *strategy-execution chasm* where strategic and tactical decisions are disjointed. In the worst case, such a chasm results in the development teams doing a great job at building the wrong product. Figure 2 (on the following page) illustrates the connection between strategy and execution.

In Figure 2, the strategic work guides the tactical product decisions, and insights from building software help you adapt the product strategy and roadmap. A validated product strategy directs the product roadmap, which helps discover the right product backlog items. These are transformed into product increments and eventually a product. The former allows you to determine the development progress using, for

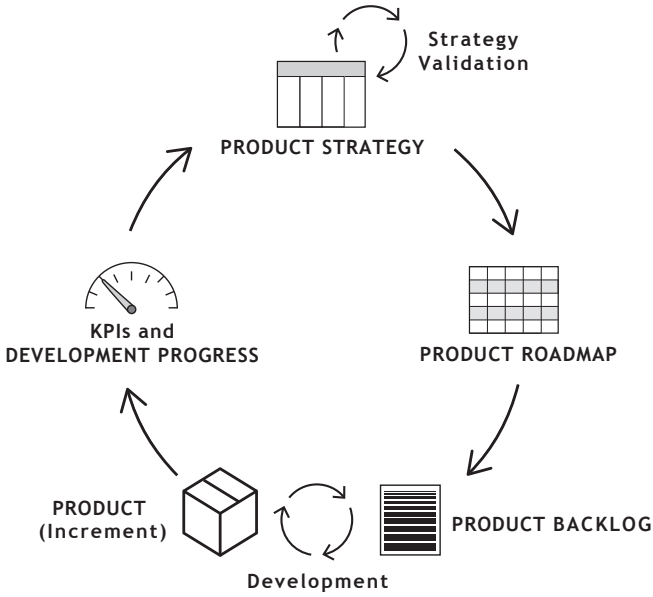


Figure 2: The Product Strategy Cycle

example, a release burndown chart.[3] The latter enables you to measure the product performance using key performance indicators (KPIs) like engagement, revenue, and customer satisfaction. The data you collect helps you inspect the strategy and roadmap and adapt and evolve the plans.[4]

Note that there are also two smaller circles in the diagram: a validation one that indicates that the product strategy is best tested in

3 A *product increment* is the output of a sprint. It is commonly defined as working software that has been tested and documented and that might be released. You can think of an increment as a reusable prototype, a stepping stone towards a new product or a new product version.

4 Figure 2 offers a process-oriented view of the model shown in figure 1. It complements the strategy and roadmap model with additional elements like product increment, product, development progress, and KPIs. Please note, though, that Figure 2 simplifies some of the connections to clearly illustrate the relationship between strategy and execution. For example, the figure that does not show that bigger product backlog changes can trigger a roadmap modification or that development progress data can lead to a roadmap update.

an iterative fashion, a development one showing that a digital product may be developed using sprints.

Strategic Planning, Agility, and OKRs

When I explain the model in Figure 1, I am often asked two questions: Does it support an agile way of working? And can it be used with objectives and key results (OKRs)? The short answer to both questions is *yes*. But let me share the long answers with you.

The model facilitates agility in two ways: First, the product strategy and roadmap state higher-level goals that guide the detailed product decisions without anticipating or prescribing them. Instead, the product functionality and detailed user experience design are discovered while the product is developed using an agile framework like Scrum. Second, the product strategy and roadmap are regularly reviewed and updated. They are not static, fixed plans, but they evolve based on changing market conditions, technologies, and regulatory requirements. Additionally, insights from demoing and releasing product increments to users and customers are integrated into the strategy and roadmap, as Figure 2 illustrates. As mentioned above, product backlog changes can trigger a product roadmap update, which, in turn, might lead to a product strategy change. In other words, the strategic planning approach I suggest supports empirical management and adaptive planning.

If you want to use OKRs, then you can view the goals in Figure 1 as objectives. There are four goals: *vision*, user and customer goals captured as *needs*, *business goals*, and *product goals*. These goals form a cascading chain: The vision guides the needs and business goals, and the latter direct the product goals.[5] You might, for example, formulate the product goals as objectives and the corresponding dates, metrics,

---

5 The model in Figure 1 implements the hierarchy of product-related goals I propose in Pichler (2020). I explain in more detail how you can discover the right needs and business goals in the chapter *Product Strategy* Development, and I'll discuss how you can derive product goals from the needs and business goals in the chapter *Product Roadmap* Development.

and features as the key results. But I personally find OKRs not well suited to manage products and set product-specific goals. Here is why: Using OKRs tends to lead to a rather text-heavy approach. It can result in documents that are comparatively lengthy and hard to understand. This feels like turning back the clock to the 1970ies when Andy Grove invented OKRs at Intel. It ignores the progress made in recent years by developing dedicated visual product management tools like my product vision board and GO product roadmap or Alexander Osterwalder's business model canvas (Osterwalder and Pigneur 2014).

Empowerment, Collaboration, and Strategic Decisions

Understanding what the product strategy and roadmap are and how they relate is important to effectively use the plans, align the stakeholders, and direct the work of the development teams. But it is not enough. Without the right level of empowerment, you risk that your strategy and roadmap are driven by senior stakeholders and that the plans primarily cater for their needs rather than the users' and customers'. But such an approach is hardly a recipe for achieving product success. In the worst case you end up with a *Frankenstein product*—a product that consists of an odd collection of features, has a weak value proposition, and offers a poor user experience. As the person in charge of the product, *you* should own the product strategy and the product roadmap, and you should be empowered to have the final say on strategic decisions.

But this does not mean that you should create the strategy and roadmap on your own, hand the finished plans to the stakeholders and development teams, and expect them to put them into action. No matter how well thought-out your product strategy and roadmap are, they are worthless if the stakeholders and development teams do not buy into them. You should therefore involve them in creating and updating the plans, preferably in the form of collaborative workshops, as I explain in more detail later in this book. A collaborative approach offers the following three benefits:

First, it leads to better decisions, as it allows you to leverage the expertise of the stakeholders and development team members. Second, it creates transparency and a shared understanding—people know what needs to be done and why a decision was made. Third, it achieves greater buy-in. If you give people the opportunity to contribute to strategic decisions, they are more likely to support and implement them.

When you decide together with the stakeholders and development teams, cultivate an open mind, and don't cling to preconceived ideas. Instead, attentively listen to the individuals, and appreciate their ideas and concerns if even you disagree. But don't make the mistake of trying to please people or allow individuals to dominate. Instead, search for a decision that maximises the value your product creates and that attracts as much support as possible.

Overcoming a Lack of Empowerment

If you currently don't own the product strategy and roadmap and if you are not empowered to make strategic product decisions, then follow these three tips: First, increase your product strategy and roadmap expertise. This includes being able to create and validate a product strategy and derive a product roadmap from it—something this book will teach you. Otherwise, it will be hard for you to earn the stakeholders' trust and guide them. Second, strengthen your leadership skills and your ability to influence others. This includes practising active listening and constructively dealing with disagreements and conflicts, as I explain in more detail in my book How to Lead in Product Management (Pichler 2020). Third, lobby for more authority and engage with the senior decision-makers in the organisation. Explain to the individuals why empowerment is a perquisite for achieving product success. Your Scrum Master or agile coach may be able to help you with this, as facilitating organisational change is part of their job. This includes establishing an effective product management function.

Four Product Success Factors

The ultimate goal of creating a product strategy and roadmap is to achieve *sustained* product success. You want to ensure that your product does a great job for its users and customers and that it generates value for the business—on a continuous basis. While realising product success cannot be reduced to a simple formula, there are four factors

that have a profound impact on it. These are: desirability, feasibility, viability, and ethicality, as shown in Figure 3.

Desirability is probably the most important factor. If people don't need or want to use a product, it will fail. To be desirable, a product must solve a specific problem or offer a tangible benefit. An example of the former are Apple's AirTags, which help people find their keys and other misplaced items. Contrast this with Sonos, a wireless hi-fi system that allows people to enjoy music by providing easy access to a range of streaming services. But no matter if a product solves a problem or provides a benefit, it must create value for its users—or it is doomed.

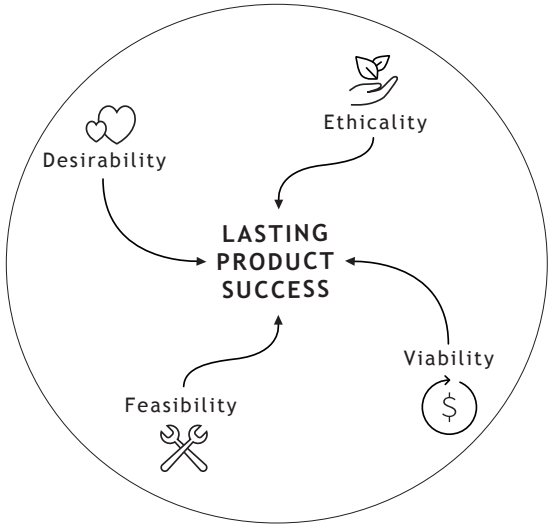


Figure 3: The Four Product Success Factors

Feasibility implies that it is possible to design and build the product; the technologies required exist or can be developed. Additionally, enough people with the right skills are available or can be recruited. If, for instance, developing the product requires advanced machine-learning algorithms, then you'd have to ensure that appropriate machine-learning frameworks exist or that they can be created.

Viability means that developing and providing the product is vi-

able from a business perspective. The product therefore must create enough business benefits to justify spending money on it. These may include generating revenue, reducing cost, increasing productivity, and strengthening the brand. Examples of revenue-generating products are Google Search, which makes money through ads, and Microsoft Word, which is monetised through subscriptions (as part of the Office suite). Contrast them with products like the Google Chrome browser and Microsoft Edge, which offer different business benefits. They allow their companies to control how users access the Internet, tie them to their respective ecosystems, and make it more likely that they interact with revenue-generating offerings.

Ethicality, finally, states that a product must not cause any harm to people and the planet. It must not affect people's mental wellbeing, for instance, by getting them hooked or by offering content that promotes misinformation, self-harm, or violence. Additionally, an ethical product does not contribute to climate change and does not damage the environment by how it is developed, provided, and if it includes hardware, manufactured, delivered, and disposed of. To achieve this, you will benefit from a business model that is fair to all parties and from making ethically sound design and programming decisions, for example, by avoiding the use of dark patterns and mitigating machine learning biases, as I explain in more detail later in this book.

A Brief History of the Four Success Factors

The first three factors—desirability, feasibility, and viability—were originally suggested by Tim Brown in his book *Change by Design*. He describes them as competing constraints that must be balanced to innovate successfully. But as Brown notes, this could result in "dreaming up alluring but essentially meaningless products destined for the local landfill—persuading people … 'to buy things they don't need with money they don't have to impress neighbours who don't care.'" (Brown 2009, pp. 20) What's more, the impact digital products can have on people's mental health makes it compulsory in my mind to add ethicality as a fourth success factor. On the positive side, an ethical product increases the chances of achieving lasting product success, and it reduces the risk that the reputation of the company suffers.

A Product Strategy Process

Making the right strategic product decisions usually requires you to collect data and acquire new knowledge. You might have to talk to prospective users, perform a competitive analysis, and investigate suitable technologies—to name just a few common tasks. This work needs to be planned and managed, and to do this, you will benefit from establishing the right process. This process comes in two flavours, timeboxed and continuous strategizing.

Timeboxed Strategizing

Whenever you develop a brand-new product or make a bigger change to an existing one, I recommend using an innovation process like the one shown in Figure 4.

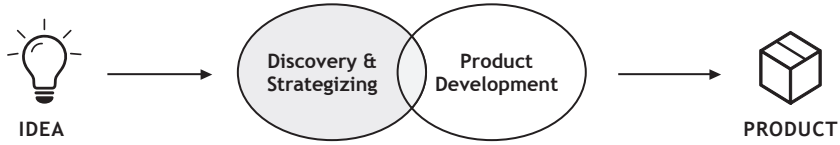


Figure 4: Innovation Process with Timeboxed Strategizing

The process in Figure 4 consists of two sets of activities, product discovery/strategizing or and product development.[6] The first set determines if and why the product should be developed. This might require observing and interviewing users, developing and testing prototypes, and exploring the product's underlying business model. The results include a validated product strategy, an actionable product roadmap, and an initial product backlog, as well as a valid business model, high-level user experience (UX) design concept and coarse-grained software architecture.

---

6 Product discovery refers to the activities required to determine if and why a product should be developed. Carrying out this work makes it more likely to create a product that users want and need. Cagan (2018, 27) suggests that this involves answering the following four questions: Will people use or buy the product? Can the users figure out how to use it? Can the developers build it? Can the stakeholders support the product? As this definition shows, product discovery and strategy work can be viewed as largely identical.

The second set of activities determines what the product should look like and do and how the product should be developed. This involves discovering and implementing the right UX design and functionality as well as making the right technical decisions. If you use a Scrum-based development process, you will test product increments with selected users early and often, for example, by demoing and releasing the software to them. This helps you learn from their feedback, adapt the product while it is being developed, and maximise the chances of creating a product that does a great job for its users.

Note that Figure 4 shows discovery/strategizing and development as overlapping activities rather than distinct phases. The reason: Some development tasks like high-level user experience and architecture design are often part of the initial work. This allows you to address risks that are related to the desirability and feasibility of the product. Additionally, it sets up the development work and removes the need for a preparation sprint or sprint 0. Be careful, though, not to make the mistake of carrying out big design upfront (BDUF). Only address the major UX and architecture risks and make most of the design and implementation decisions when the actual product is developed.

As it's hard to correctly predict how much time will be required to complete the necessary work and create a validated product strategy and actionable roadmap, I recommend using a time box, a fixed period during which the work has to be completed. Your goal should be to carry out *just enough* product discovery and strategizing work to create the plans and be able to make an informed go/no-go decision about developing the actual product. If you are unsure how big the time box should be, then start with one month and hold weekly review meetings where you assess the progress, as I discuss in more detail in the chapter *Product Strategy Validation*.

Finally, don't forget to involve the key stakeholders and development team members in the strategizing work. While it can be helpful that the entire development team participates in the activities, it is often sufficient to ask the team to nominate three people: a member with the right UX skills—assuming that the product is end-user facing, a

member with the relevant architecture and technologies expertise, and a member who has the right quality assurance capabilities. This will enable you to address the key user experience and technology risks, including having the right development and test tools available.

Design Thinking and Timeboxed Strategy Work

If you are interested in design thinking, you might be wondering if you can use design-thinking concepts to carry out the strategizing work shown in Figure 4. Before I answer this question, let's briefly explore what design thinking is. The approach can be described as five sets of activities, which are commonly called *empathise*, *define*, *ideate*, *prototype*, and *test*.

The first set helps you understand who your users are and what they need, for instance, by observing and interviewing them. The second set defines the specific user problem that you want to solve. The third set generates ideas for possible solutions, which are turned into simple prototypes in the fourth stage. The final set tests the prototypes with selected users to find an approach that is likely to result in a successful product.

Note that the activities are usually not carried out sequentially. Instead, you may find yourself looping back to a previous stage. For example, you might discover in the prototype stage that none of the solution ideas receive positive feedback from prospective users. Consequently, you'd have to go back to the ideate stage and come up with new ones.

Can you use the design-thinking stages to structure and organise the strategizing work shown in Figure 4? Yes, you can. What's more, the stages are compatible with the approach of creating and validating a product strategy discussed in this book. You should therefore be able to apply the practices in a design-thinking context.

Continuous Strategizing

As helpful as timeboxed strategizing is to create a new product strategy and roadmap, it is not enough. The world doesn't stand still; markets and technologies change; new competitors emerge. It is therefore important that you practise continuous strategizing and regularly review and adapt your strategy and roadmap. This includes the following five activities: First, determining the product performance using key performance indicators (KPIs), such as engagement, conversion, and customer satisfaction. Second, reviewing market trends that might impact your product like machine learning and Internet of Things (IoT). Third, keeping an eye on the competition and monitoring the changes

you competitors make to their product portfolios and individual offerings. Fourth, following developments at your own company that might affect the product strategy and roadmap, for instance, a change in the business strategy. Fifth, meeting selected users and customers at least once per quarter, be it online or onsite. This will help you empathise with them and discover new ideas for product enhancements and new offerings.

You can think of continuous strategizing as an ongoing workflow. This workflow directs the tactical work and informs the detailed product decisions that are captured in the product backlog.[7] At the same time, insights from the tactical work influence the strategic decisions and help you adapt the product roadmap and strategy. As mentioned before, the strategic and tactical work should be connected, and strategic and tactical product decisions should be closely aligned.[8]

To help you effectively carry out the continuous strategizing work, I recommend the following two measures: First, make sure that you allocate enough time, at least half a day per week as a rule of thumb. Some product people like to spend one hour per day on continuous strategizing, others prefer to carry out the necessary work once or twice per week. Note that the amount of uncertainty and change present will impact the strategizing effort: A young product requires a higher effort compared to an older, mature one. Second, schedule regular strategy and roadmap reviews—roughly once per quarter—and invite stakeholders and development team members to them.

---

7 Torres (2021, 21) suggests that continuous discovery involves carrying out research activities at least once per week, which include interacting with customers.

8 One way to achieve this is to ask the person in charge of the product to look after the product strategy and product roadmap as well as the product backlog. This results in what I call full-stack ownership: The individual owns the product top to bottom, so to speak. To apply this concept on larger products and in a scaled environment, use specialised product roles that are responsible for product parts like features and components as I explain in Pichler (2016). Using product goals on the product roadmap and copying the next one into the product backlog is another alignment measure, as I discuss in more detail in the section *Derive the Product Backlog from the Roadmap*.

The first measure helps you avoid nasty surprises such as a competitor offering a new killer feature. It makes it more likely that you notice early warning signs like declining sign-up rates, increasing churn, or growing number of support calls. This allows you to be responsive and take action early so you don't end up firefighting and possibly having to resolve a crisis. The quarterly reviews help you consider longer time frames and bigger trends. By involving stakeholders and development team representatives—ideally the same people who helped you create the current product strategy and roadmap—you leverage people's collective expertise, create alignment, and secure buy-in. While I do recommend that you schedule the reviews in advance, you should, of course, not wait for the next review if there are new developments that need to be urgently addressed. Instead, hold the next collaborative review as soon as possible.

Tips for Making Time for Continuous Strategizing

As the person in charge of the product, you might find that your workload is already so high that practising continuous strategizing seems impossible. If that's the case, then follow these four tips to free up time:

First, don't carry out tasks that are not part of your actual role. This includes taking on Scrum Master duties, which is a common mistake in my experience.[9] Second, delegate and share some of the work. For instance, the development team members might be able to do some product backlog refinement work on their own. Third, don't task-switch. Instead, do one thing at a time. Every time you start a new task, be it reading an email or looking at new data, you need time to remind yourself of what has to be done and how to do it. Fourth, take regular breaks that allow you to recharge. Working too hard for too long is counterproductive. Productivity drops and mistakes increase.

Be aware that de-prioritising or neglecting strategic work will most likely lead to unplanned work in the future. If you don't make time to listen to the users and watch the market, then don't be surprised if engagement and referral rate are down or if a competitor leapfrogs you. In this sense, strategizing is like riding a bicycle or driving a car: If you don't look ahead, you're likely to crash.

---

9 Taking on Scrum Master responsibilities does not only increase your workload. It also hides a systemic problem and organisational impediment: If you seem to be able to take on the Scrum Master role, then there is little need for your company to hire or develop Scrum Masters.

Digital Products and Product Strategy

A product strategy and a product roadmap are plans that describe the value a *product* should create. But what is a product? While this might seem a trivial question, I find that many organisations lack a clear and shared answer to it, especially when it comes to digital products. This, however, can cause the following two issues:

First, you might mistake a product part—like a feature or component—for the actual product. I view a feature as a product capability, as a large piece of functionality, for instance, the ability to save a file. A component is an architecture building block like a service, layer, or subsystem. It might be the code that handles data storage, to stay with the example. This mistake would cause you to create a feature or component strategy and a feature or component roadmap. But these plans would be too detailed and fine grained. They would provide no real value despite increasing the planning and coordination effort.

Second, you might make the mistake to view a product portfolio or a product bundle like Adobe Creative Cloud and Microsoft Office as a product.[10] This can lead to a strategy and a roadmap that is too coarse-grained and high-level. Consequently, it does not offer enough guidance for the stakeholders and development teams working on the individual products.

These mistakes show that it's important to be clear on what a product is and if the asset you manage is a product before you create a product strategy and roadmap. I view a product as an entity that creates specific value for a group of people, for the users and customers, and for the organisation that develops it. The former is achieved by solving a problem—think of Google Search, which addresses the challenge of finding information on the Internet—or by providing a tangible benefit—take Microsoft Word, which allows people to easily create and edit documents. Creating value for the business is accomplished by directly generating revenue, like Microsoft Office does, helping promote or sell

10 For more information on product bundles, please refer to the section *Create a Product Bundle* in the chapter *Product Strategy* Development.

other products or services, think of an online store like Amazon.com, and increasing productivity and reducing cost, as an internal software platform does, for instance.[11] To put it differently, a product is a value-creating vehicle: It's an asset that exists to enhance people's lives and to help an organisation grow and prosper.

If an asset does not create value for its users, customers, and the company, then it is not a product. Take Microsoft Word as an example. As a user, I can open a file, edit it, and save it. These three capabilities represent steps in a common user journey, they might be developed by dedicated teams, and they might require complex technical solutions. But I would view them as features, not products, as they do not provide value to the users and to Microsoft on their own.

Tips for Identifying Digital Products

If you struggle to determine which assets are products and which ones aren't, then follow this approach: Start with the users and customers and identify your products *outside-in*. Ask yourself why people use the capabilities you offer. Which problems do they try to address, which benefits do they want to obtain, or which jobs do they try to get done? You can answer these questions by observing and interviewing users and by using relevant analytics data.

Additionally, begin with revenue-generating products, which are usually comparatively easy to identify: These are the assets people are willing to pay for. An example would be the products sold on an online retailer's website. Then consider the supporting products like the website and mobile apps that allow the customers to find and purchase the right products. Finally, determine any internal technical products like a software platform or a data repository. This should give you a clear understanding of the products in your business and put you in a good position to make the right strategic product decisions.

11 Please see the section *Take Advantage of Software Platforms* in the chapter *Product Strategy* Development for more information on platforms.