Boost's Guide to being a Kick-ass Product Owner

Making a bigger impact and a better world as a Product Owner

Contents

» *Links and Tool links* — Throughout this document you will find boxes with links like this. These take you to other useful resources and practical tools to help you on your journey as a Product Owner.

The Guide shows Product Owners how to use Scrum to make a bigger impact and a better world.

Introduction

A practical guide to maximising your impact

The Guide shows Product Owners how to use Scrum to make a bigger impact and a better world. It gives you the skills you need to become a kick-ass Product Owner.

You'll learn how Scrum works, and how to leverage the roles and rules of Scrum to deliver the most value, most efficiently. Find out how you can engage your stakeholders and inspire your Scrum Team to build a product that your customers will love, and love to use.

The Guide is based on the experiences of working Product Owners. It uses real-world examples to illustrate how to apply the Agile mindset and the Scrum framework in your day-to-day work. You'll get templates you can follow and practical exercises you can run.

This is Boost's approach. Agile is about learning and adapting as you go, and this is what we've found works in the eleven years we've been delivering Agile projects. We'll explain where and why we sometimes vary from standard Scrum practice.

A Guide for all Product Owners... and their colleagues

The Guide explains Scrum from the Product Owner's perspective. While it's written with Product Owners in mind, it'll be useful for anyone with an interest in Scrum. Stakeholders, Scrum Masters and members of Development Teams will get a Product Owner's view of Scrum. This will help you support your Product Owner to deliver the most value.

The Guide is for all Product Owners, regardless of experience. If you're new to, or considering moving into the role, you can use the Guide without any knowledge of Scrum. On the other hand, if you're an established Product Owner, you'll get new perspectives and the chance to learn from the experiences of your peers.

While Scrum was born in the world of software, the Guide is for owners of any product, working in any sector.

The Guide shows how to build your product. To discover what to build and why in a one-day workshop, check out our Agile Project Kick-off Kit.

» *Download your Kick-off kit*

What you'll learn

Find out:

- how Scrum works and how to make it work best for you
- the responsibilities of the Product Owner role
- how to meet these responsibilities
- the most effective ways to work with your Scrum Team
- how to get the support from stakeholders you need to succeed
- why the Agile mindset is key to successfully using Scrum.

You'll learn how to:

- regularly deliver working solutions to your customers
- get the right people making the right decisions at the right time
- inspire and empower your Scrum Team
- get buy-in from and set expectations for your stakeholders
- discover how your product can make your customers' lives better
- utilise user stories to turn what you discover into customer-centred solutions
- constantly improve your product and your processes.

The role of the Product Owner

As Product Owner you are responsible for maximising the value delivered. You do this by working with stakeholders to develop a vision and strategy for a product that will achieve your organisation's goals by making the lives of your customers better.

This vision helps you inspire your Scrum Team by showing how they can make a difference. It also helps you discover what your customers need and the solution that will meet these needs. The Product Owner is the voice of the customer.

You are part of the Scrum Team and work on a day-to-day basis with the people building your product. You prioritise and communicate to the team the benefits you want to give your customers. The team consider how they'll deliver these benefits, and you all discuss and confirm the approach. The team work in short iterations to build working solutions that you can get in front of your customers for testing. To keep these iterations moving efficiently, you are available to clarify requirements, answer questions and check work as it's completed.

Throughout this cycle you collaborate with your Scrum Team to inspect and adapt your product and processes, boosting effectiveness and customer satisfaction.

About Boost

We've spent eleven years refining our Agile practice while designing and developing web and mobile apps, delivering our popular ICAgile-certified courses and providing Scrum and Agile consultancy services.

Boost's purpose as a company is to support others to create a positive and lasting impact.

info@boost.co.nz

+64 4 939 0062

twitter.com/boostnz

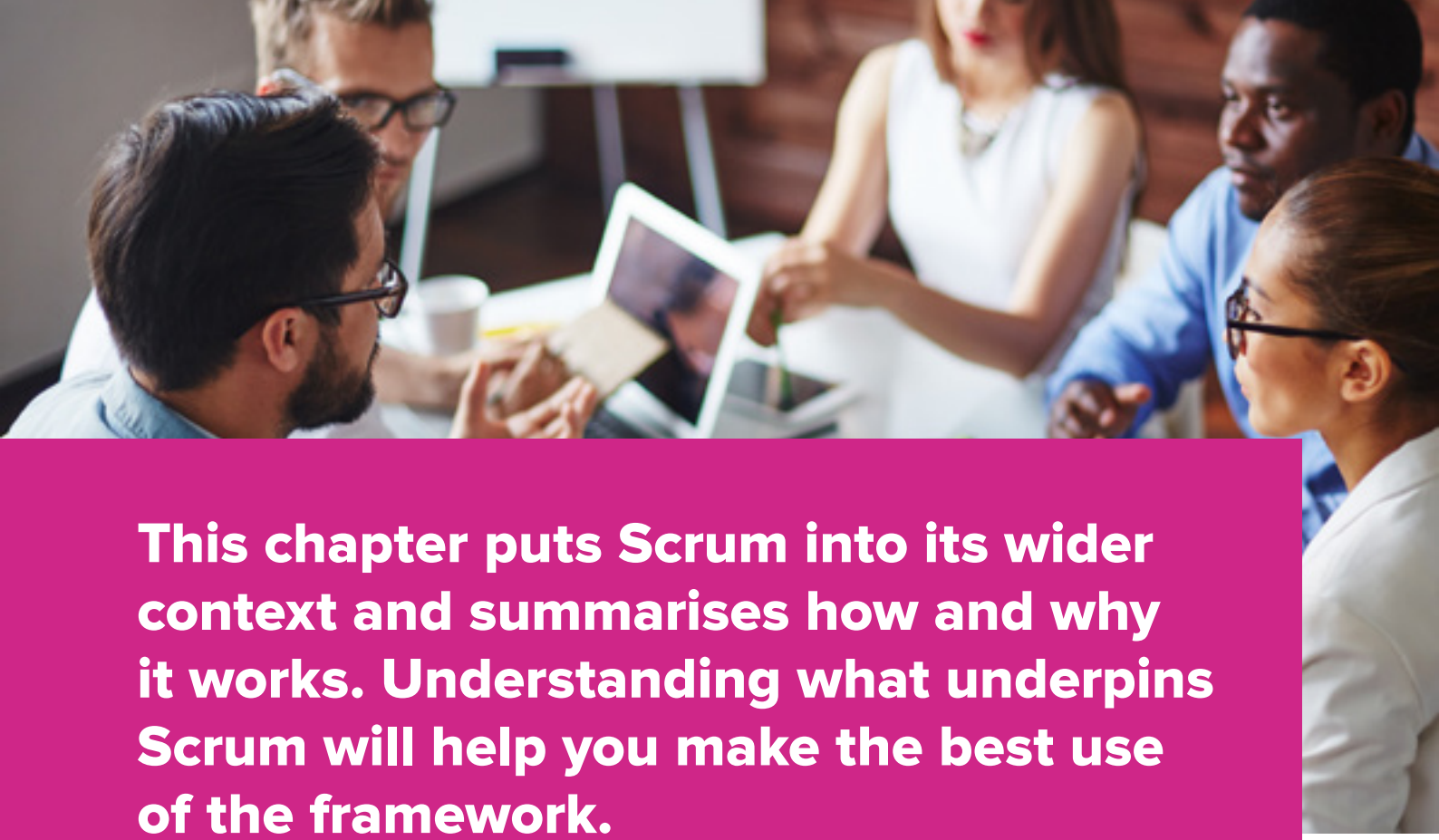
facebook.com/BoostNewZealand

Level 5, 57-59 Courtenay Place, Wellington

What is Scrum?

Scrum is a way for teams to work together so they deliver the most value, most efficiently.

Section contents



This chapter puts Scrum into its wider context and summarises how and why it works. Understanding what underpins Scrum will help you make the best use of the framework.

Scrum is an Agile framework in which self-organising, self-contained teams work collaboratively and transparently in regular iterations, inspecting and adapting as they go, in order to sustainably deliver maximum value.

Scrum uses a set of roles, rules and tools to deliver working solutions in regular stages or iterations, with regular check-ins to inspect and adapt the work. These look and learn loops mean decisions are made with the best available information, and the product and processes can be refined and improved. Collaborative, self-contained and self-organising teams give Scrum projects a sustainable rhythm that avoids the damaging end-of-project crescendos common to many traditional Waterfall projects.

The Product Owner within a Scrum project shares the same goal as the Scrum framework itself: maximising the value delivered to the customer.

While Scrum is easy to learn, it's hard to master.

Boost's Culture Lead Gavin Coughlan has filled all the roles in Scrum, so he has a good understanding of what works.

"It's a pretty lightweight framework," he says. "There's not a lot to it, but it takes a lot of rigor and discipline to really make it work."

Scrum theory

Scrum is based on the idea that decisions are better based on experience than guesswork. To this end, Scrum maximises opportunities for:

- transparency
- inspection
- adaptation

Working in iterations lets you regularly inspect what you're doing and adapt if necessary. When you combine this with a collaborative and transparent approach, you make sure the right people have the right information at the right time so they can make the right decisions.

> **To get Scrum working well, you need to have an Agile mindset.**

- *The Scrum Guide* — the official Scrum manual
- *The Agile Manifesto* — the Agile values and principles
- *Agile Product Ownership in a Nutshell* — 15 minute YouTube intro from Henrik Kniberg

Scrum and Agile

Scrum is just one set of tools and techniques that help you work in an Agile way. Others include Extreme Programming, Crystal, Lean and Kanban.

Agile is a philosophy or a mindset, Scrum is a framework. Agile has values and principles, Scrum has roles and rules. You are Agile, you use Scrum.

You can think of it a bit like vegetarianism. A vegetarian might have values (animals are people too) and principles (don't eat anything that moves). That's like Agile. But when dinner time approaches they'll rely on a recipe book. That's like Scrum.

A great cook doesn't need a recipe book. So it is with Agile and Scrum. Once the Agile mindset has become ingrained in the way you work, you can start to experiment.

Gavin puts it like this:

"You can think of Scrum as a set of training wheels for people who are starting to work in an Agile way. It really forces a lot of the values and principles to get embedded in how you work on a day-to-day basis.

"It's a good idea for people starting out to do it pretty much by the book. Once you're a little bit deeper into it, then you can start to see what's working really well for you or what's not."

Interestingly, while Agile encompasses Scrum, Scrum came first.

The History of Scrum and Agile

Former Top Gun, doctor and IT boffin Jeff Sutherland, and developer, product manager and consultant Ken Schwaber unveiled Scrum in 1995.

The name comes from the game of rugby. In 1986, Hirotaka Takeuchi and Ikujiro Nonaka used the Scrum analogy to describe new fast and flexible approaches to development.

The nineties saw a range of these approaches appearing. In 2001, 17 software gurus met in a ski lodge in Utah. They didn't agree on much, but they did agree on what became the Agile Manifesto.

The Agile Manifesto

To get Scrum working well you need to have an Agile mindset: you need to embody the values and principles of Agile that were set out in The Agile Manifesto.

"You can use Scrum without being aware of the Agile values and principles but it would be a huge mistake," says Gavin. "You'll probably see some benefits, but you won't get the massive benefits that you should get if you actually focus more on the values and principles."

Agile Values

If you're not involved in development, each time you read the word "software", replace it with "solutions".

We are uncovering better ways of developing software by doing it and helping others do it. That is, while there is value in the items on the bottom, we value the items on the top more. Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

Principles behind the Agile Manifesto

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity – the art of maximising the amount of work not done – is essential.
- The best architectures, requirements, and designs emerge from self-organising teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour.

The Scrum Guide

What is Scrum? Well, Scrum is what the Scrum Guide says it is. The Guide is Scrum's evolving gospel. If what you're doing doesn't match what's in the guide, it's not Scrum. It doesn't mean it won't work though.

» *The Scrum Guide*

Scrum Values

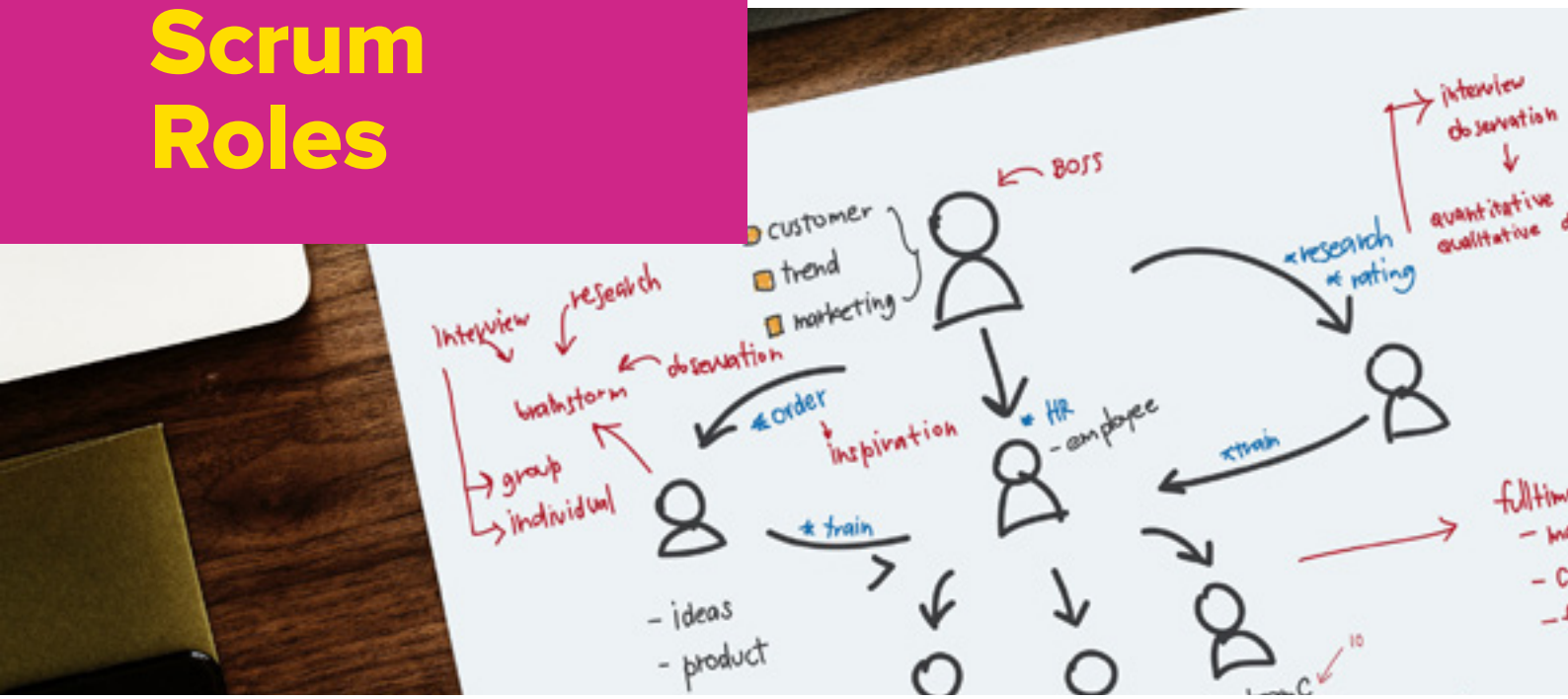
> **When the values of commitment, courage, focus, openness and respect are embodied and lived by the Scrum Team, the Scrum pillars of transparency, inspection, and adaptation come to life and build trust for everyone.**
>
> – The Scrum Guide

Commitment **Courage** **Focus** **Openness** **Respect**

Because Scrum is strongly collaborative, it works best when everyone has a shared set of Values guiding the way you work. You can expand on these by developing your own Team Charter.

» *Develop your Team Charter*

Scrum Roles



The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master.

– The Scrum Guide

Product ownership expert Jeff Patton sums up the aims of each role like this:

- **Product Owner:** Build the right product
- **Development Team:** Build the product right
- **Scrum Master:** Help everyone keep the process working effectively

It can take a while to get to grips with the fact there's two teams: the Development Team and the wider Scrum Team including the Product Owner and the Scrum Master. Both teams are self-organising and cross-functional. It's not unusual for people to move from one role to another (in established teams with a strong Agile mindset anyway).

So as Product Owner you are an integral part of the Scrum team, not just a client writing cheques and checking Gantt charts.

Product Owner

> **The Product Owner is responsible for maximizing the value of the product resulting from work of the Development Team.**
>
> – The Scrum Guide

As Product Owner, you're the voice of the business and voice of the customer. You bring the product vision and strategy. You show the team 'why', they show you 'how'.

The Product Owner manages the Product Backlog, the prioritised list of outcomes we need the product to achieve.

You're available to answer questions and clarify requirements and you check the work in progress to ensure it's getting the project closer to achieving its outcomes.

To do this you need your organisation at your back. You need the time and authority to make prompt decisions.

Development Team

> **The Development Team consists of professionals who do the work of delivering a potentially releasable Increment of "Done" product at the end of each Sprint.**
>
> – The Scrum Guide

The Development Team is:

- **Self-organising:** No one assigns tasks to team members and only they decide what is achievable each Sprint, and how it is achieved.
- **Cross-functional:** It has all the skills needed to do the work. Ideally team members are also cross-functional. This helps stop specialists becoming bottlenecks.
- **A single unit:** There are no sub-teams and even if individuals have specialities, the whole team shares accountability for all work.

Scrum Development Teams are motivated to deliver because they're empowered to make a shared commitment to the work they'll do.

Development Teams work best when there's between five and nine people, all working in the same location. Otherwise communication can suffer.

They also work best when the Product Owner knows the product well, and communicates this knowledge responsively and decisively.

Scrum Master

The Scrum Master is responsible for promoting and supporting Scrum as defined in the Scrum Guide. Scrum Masters do this by helping everyone understand Scrum theory, practices, rules, and values.

– The Scrum Guide

The Guide calls the Scrum Master a servant-leader. Other frameworks call them Agile Coaches. They help the whole Scrum Team by making sure they follow the Scrum framework and are embedding the Agile values and principles in their process. Additionally, they help those outside the Scrum Team understand how they can most effectively work with the team (code for keeping their sticky beaks out of the Sprint).

They make sure everything that's needed is in place, clear blocks to progress and help all roles work together well.

The Scrum Master is the Scrum expert, so as Product Owner if you need help with Scrum, turn to the master.

Scrum Events

The events (sometimes called ceremonies) of Scrum are designed to maximise the benefits of face-to-face communication, maintain transparency and lock in regular opportunities to inspect and adapt.

The Sprint

> **A time-box of one month or less during which a "Done", usable, and potentially releasable product Increment is created.**
>
> – The Scrum Guide

In Scrum your work loops round in a repeating pattern of iterations called Sprints. Each Sprint is the same length and is structured in the same way, with the same events occurring each time.

Sprint Planning

> **The work to be performed in the Sprint... This plan is created by the collaborative work of the entire Scrum Team.**
>
> – The Scrum Guide

Who: Development Team, Product Owner and Scrum Master.

Sprint Planning decides what can be delivered this Sprint and how this will be done.

The Development Team runs through items from the Product Backlog in priority order, choosing those they forecast that they'll complete in the upcoming Sprint.

Neither Product Owner nor Scrum Master can tell the team what goes in. The Scrum Team fleshes the items out (they become the Sprint Backlog) and agree on a Sprint Goal (the overall objective of the Sprint).

At Sprint Planning the team will quiz the Product Owner if they need any clarification. Making the requirements clear is a key way Product Owners ensure the right product is built.

Daily Scrum

> **The Daily Scrum is a 15-minute time-boxed event for the Development Team…At it, the Development Team plans work for the next 24 hours.**
>
> – The Scrum Guide

Who: Development Team (at Boost we aim to have the Product Owner and Scrum Master there too, keeping them in the loop).

The Development Team coordinate the upcoming day's work at the Daily Scrum (a.k.a. daily stand-up).

It's a time to see who needs help and advice. It can help to encourage the Development Team to answer — in a conversational way — three "inspect and adapt" questions:

- What did I do yesterday that helped us meet the Sprint Goal?
- What will I do today to help us meet the Sprint Goal?
- Do I see anything that blocks me or the team from meeting the Sprint Goal?

For a Product Owner this is a great chance to see the progress that's being made.

Sprint Review

> **A Sprint Review is held at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed.**
>
> – The Scrum Guide

Who: The Scrum Team and anyone who is interested in seeing the progress of the project, such as stakeholders, subject matter experts or actual users.

The Sprint Review lets the Product Owner test that the product meets both business and user needs, and is a chance to build internal buy-in.

The Sprint Review starts with a demo of the working features created during the Sprint. It's not a slideshow presentation, it's the actual product in action. This allows for hands-on interaction and meaningful feedback.

Based on what the Product Owner learns in the Review, they'll be able to adjust the features and priorities in the Product Backlog.

Sprint Retrospective

> **The Sprint Retrospective is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint.**
>
> – The Scrum Guide

Who: Development Team, Product Owner and Scrum Master.

At the end of each Sprint, the retro is a chance to take a look at what is working well and what can be improved. The Scrum Master usually runs the retro, making sure the meeting stays within its timebox and results in actionable ways the team can work better.

Where the Sprint Review inspects the product, the Retrospective inspects the process.

When the Product Owner models the Scrum Values of commitment, courage, focus, openness and respect, they help make retros as effective as possible.

Scrum Artifacts

> **Artifacts defined by Scrum are specifically designed to maximise transparency of key information so that everybody has the same understanding of the artifact.**
>
> – The Scrum Guide

Scrum has three artifacts:

Product Backlog

Sprint Backlog

The Increment

Product Backlog

> **The Product Backlog is an ordered list of everything that is known to be needed in the product. It is the single source of requirements for any changes to be made to the product.**
>
> – The Scrum Guide

As Product Owner, this is your baby. You choose what goes in, order it with the highest priorities — and the most detail — at the top, update it as new information comes in, and make sure the whole team always has access to it. You do this with constant input from stakeholders, subject matter experts, customers and the Scrum Team. However, the final decisions are yours.

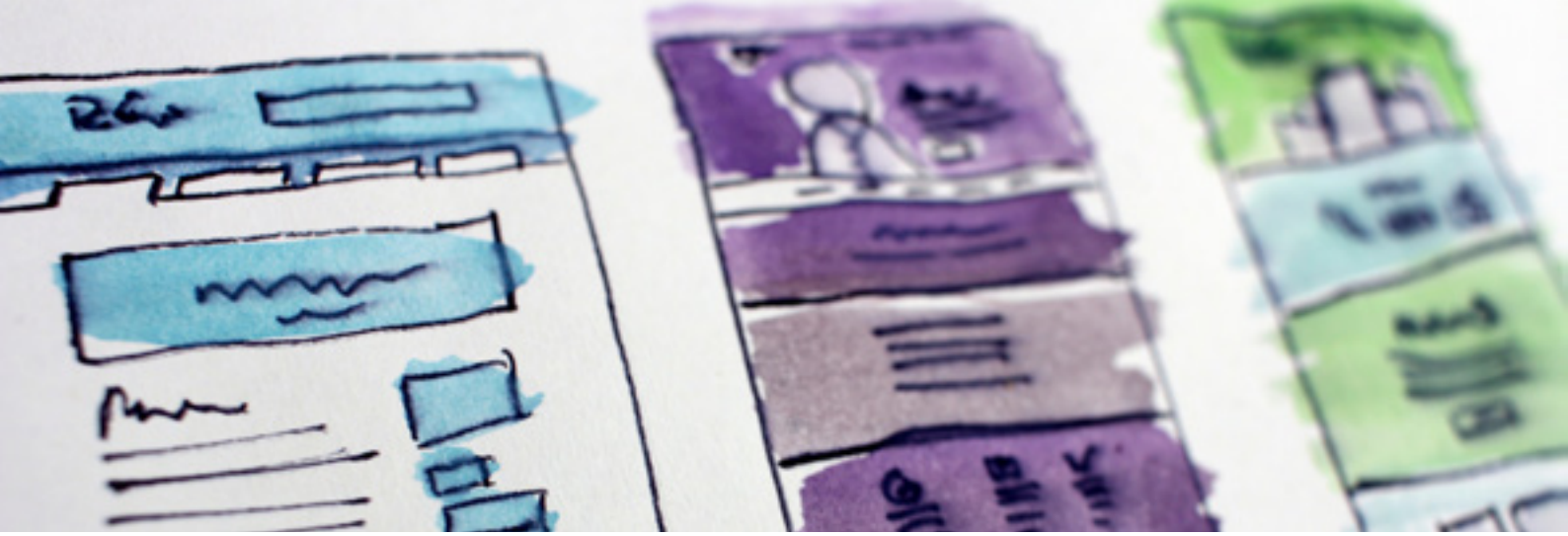
Sprint Backlog

> **The Sprint Backlog is the set of Product Backlog items selected for the Sprint, plus a plan for delivering the product Increment and realising the Sprint Goal.**
>
> – The Scrum Guide

Created at Sprint Planning, the Sprint Backlog is the team's trackable to-do list for that Sprint. Following the planning meeting, everyone has a shared understanding of all the work involved. The plan includes enough detail that you can track progress, and more detail emerges as tasks are completed. You track progress on a board or tool that is visible to the whole Scrum Team.

At Boost we usually detail the Sprint Backlog as user stories.

As Product Owner, you can't assign Development Team members to particular pieces of work. They make their own choices about what they work on.



Increment

> **The Increment is the sum of all the Product Backlog items completed during a Sprint and the value of the Increments of all previous Sprints.**
>
> – The Scrum Guide

The Increment is what you get when you add the results of the Sprint to the product created by all the previous Sprints.

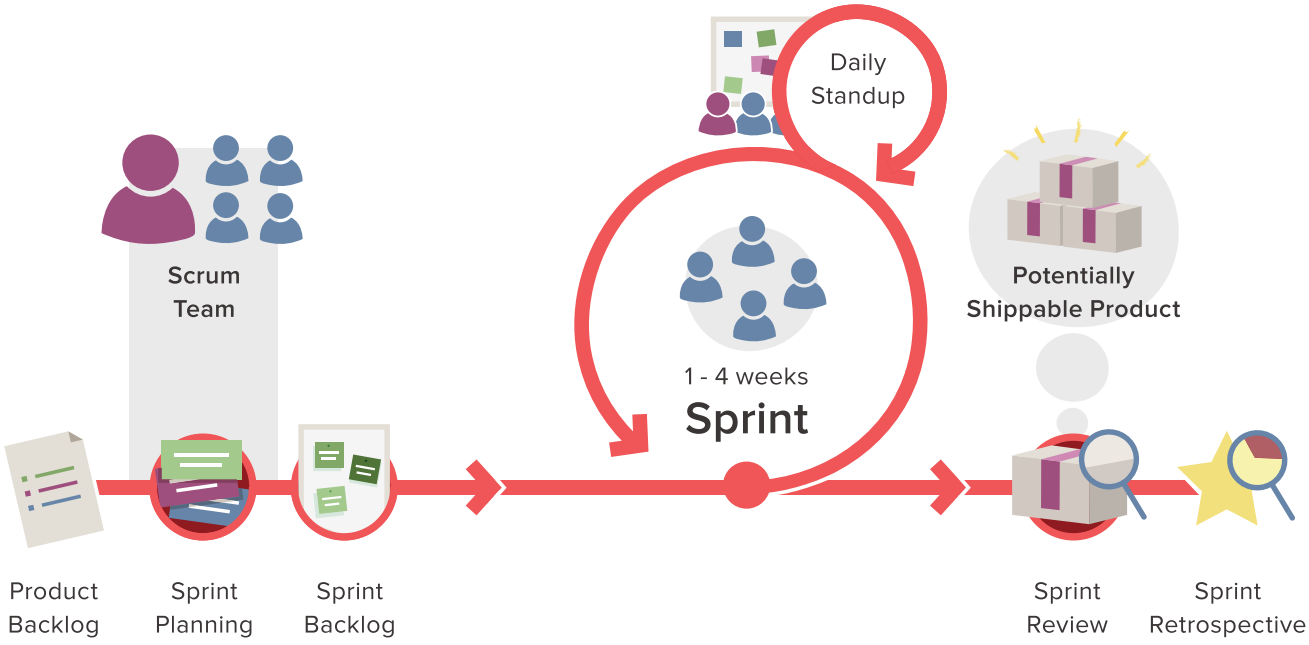
As with concepts like “minimum viable product” or “potentially shippable product”, the idea of a “Done” Increment is to make sure Scrum focuses on delivering working solutions. The Sprint loop needs to deliver something you can look at and learn from.

The Increment must:

- be usable so the Product Owner can check it (even if they aren’t going to release it)
- meet the Scrum Team’s Definition of Done
- be a step toward the product vision.

The Definition of Done spells out the whole Scrum Team’s shared understanding of what it means for a story to be finished.

The Scrum process at a glance



Having looked at the ‘what’ and ‘why’ of Scrum, our next chapter will focus more on ‘how’. First we’ll look at how a Product Owner can best work with the Scrum Team, then we’ll cover how you collaborate with people outside the Scrum, your Stakeholders.

Building a successful Scrum team

See how elements of the Scrum framework that seem abstract can be applied in practice.

Section contents

Get six tips for building a successful Scrum team. They show what you can do as a Product Owner to create a team that delivers maximum value, helping you make the biggest impact in your work.

These tips take elements of the Scrum framework that can initially seem abstract and show how you can apply them in practice in your day-to-day work with your Scrum Team.

- *The Scrum Guide* — The official Scrum manual
- *Definition of Done* — Scrum Guide
- *Scrum.org* — training and certification network
- *Agile Project Kick-off Kit* — Boost blog

Work closely together

This might sound obvious but it's worth teasing out, especially for Product Owners working for the first time with a team that is using Scrum.

Scrum embeds the customer in the Scrum team, in the form of the Product Owner.

Colocation is the best way to work together

Ideally the whole Scrum Team sits together. That's because face-to-face communication works best. If you can't sit together all the time, then do it often.

Boost's Rebecca Jones has been both Scrum Master and Product Owner.

"You want to spend as much time in person as possible. It gets the work moving faster, you feel more like a team."

Although the Scrum Guide says the Daily Scrum is just for the Development Team, at Boost we encourage Product Owners to come along each day. Our clients appreciate the insight this gives them.

"It's good to be there with the doers and hear the thinking about what can be achieved and what can't," says John O'Connell from Life Education Trust.

When colocation is not possible, video conferencing gets you face-to-face, if not in the same space.

Get to know your team

"The most successful teams I've seen have gone out for a beer together, gone for a lunch," Rebecca Jones says. "You learn a bit about the people. I think that creating empathy is really important on both sides."

Keep the team together

It takes teams a while to get to know each other. You can make best use of this time investment by not moving people in and out of the team too often.

Get together via video conferencing

At Boost, we use whatever video conferencing (VC) tool is easiest for our clients. Internally we prefer Zoom — we find it's more robust than Skype. We also sometimes use Slack and if clients have a locked-down network and can't download apps, we go for web-based tools like appear.in.

- » *Zoom*
- » *Skype*
- » *Slack*
- » *appear.in*
- » *Comparison of video conferencing tools*

Tips for video conferencing

- Test the mic, camera, speakers and connection beforehand.
- Use a strong, fast internet connection, ideally wired-in ethernet not wifi.
- Aim for even lighting and avoid bright light behind people.
- Keep background noise down.
- Make sure everyone is in shot.
- Look at the camera, it's like looking people in the eye.
- Have a Plan B channel, a phone or email so you can sort out issues.
- Make sure anyone there via VC is included in activities by having someone in the room stand in for them.
- For regular video catch-ups invest in standalone webcams and mics, and a permanent set-up so you don't fluff around each time.

Leverage Scrum roles

A successful Scrum Team makes best use of the benefits built into the breakdown of responsibilities in the Scrum Roles.

These responsibilities can be divided up like this:

- **Product Owner: *Why*** — the vision of the value the product will create
- **Development Team: *What*** — the solution that will achieve this vision
- **Scrum Master: *How*** — the way the Scrum process helps this happen

So let's look at your role as Product Owner, and how you can best work with the other roles.

» *Agile Project Kick-off Kit*

Your work as Product Owner

According to the Scrum Guide, the Product Owner is responsible for maximising the value of the product. To do this you need a clear vision of how the product will help your customers and achieve the strategic goals of your organisation. You can use this vision to inspire the team by depicting the better future you'll be building. Your vision describes the outcomes or impacts, not features. You show the team 'why' you're building the product, they show you 'how' to build it.

This process starts before your first Sprint. Our Agile Project Kick-off Kit explains how you can get the team behind your vision and move from the 'why' to doing the work itself in the space of a day.

This picture of the way you're making the world a better place becomes both a motivational and a decision-making tool.

When questions come up as the project progresses, you and the team can return to the vision. Ask yourselves, 'will it get us closer to the outcome we're after?'.

A key way the Product Owner helps turn the vision into the product itself is by maintaining the Backlog so that it's prioritised and the work to be picked up next is well understood. We'll look in detail at how you maintain the Backlog in later chapters.

Keep in mind that, as Product Owner, you can get the Scrum Master and developers to help with any of your responsibilities, if this is a cost-effective use of their time. However the buck stops with you.

Working with the Scrum Master

The Scrum Master is your first port of call for questions about Scrum best practice. This is especially important when you're new to Scrum.

"When you're new to the Product Owner role, the Scrum Master will work closely with you to make sure you understand your role, and the other roles," says Rebecca.

Make sure the Scrum Master understands the vision. That way they can help you communicate it to the rest of the team.

You can see the range of ways the Product Owner and Scrum Master can work together from the description in the Scrum Guide of the support the Scrum Master can provide the Product Owner. This ranges from, but is not limited to, helping ensure the team understand the product, through Backlog management, to facilitating events.

Working with the Development Team

To find out how Product Owners can best work with the Development Team I quizzed the crew at Boost. What I found could be summed up in one sentence.

> **"Developers work most effectively when a Product Owner knows the product well, and communicates this knowledge responsively and decisively.'**

Know your product

This is the first thing that the developers mentioned.

“A Product Owner can see things a developer won’t see because we live inside the code. They can see the whole user experience,” is how one developer put it.

For ongoing projects or value streams, it’s worth having a flexible, high-level plan for the future.

“What we really like from our Product Owners is that glimpse of where we want to be in six months, so we can all get there together,” Rebecca says.

A developer talked about a time a Product Owner put together a high level roadmap. The team realised there was a better way of achieving their goal, one which cleared away technical debt (outdated, clunky code, tools or infrastructure). The new approach was faster and more stable for end users, and future development will also be faster, so it will cost less to implement new features.

It’s a great example of the collaborative strength of the Agile process. Your product knowledge combined with their technical expertise produces a whole that is greater than the sum of the parts.

Communicate

To communicate the product vision well you don’t need to be a technical expert but you do need a technical appreciation. You need to learn the lingo, develop shared concepts and a shared vocabulary.

It helps to have the patience to keep asking questions until you both understand. Repeating back an explanation in your own words is a good way of making sure you’re on the same page.

This means you need to be good at listening. (“You also need to know when to stop listening,” I was told, “sometimes developers want to go into too much detail.”)

Developers also thrive on feedback. When they’ve got more work to do it’s best if you’re “direct without being negative” and when they’ve nailed it, let them know.

The developers also mentioned the effectiveness of good user stories as a communication tool. We’ll look in more detail at how to develop effective user stories later in the Guide.

Be responsive

This is often called being available. It's also crucial.

Developers frequently have questions or need to get tasks checked and hopefully accepted. The faster you respond, the less downtime there is, ramping up productivity.

Be decisive

To do all this you need to be empowered by your organisation. You need the authority to make decisions. If you have to consult on each one it slows down the work, meaning you get less bang for your buck. Ideally you also have the headspace to make decisions, and are not pulled in too many directions. We look in more detail at how you can ensure you're empowered to succeed in the next chapter on working with stakeholders.

Probably the most important decisions you'll make will be about priorities. You need to be ruthless in pursuing the top priorities first. You need to know what the Minimum Viable Product is and not push for a full-feature, big-bang release because that means you get none of the benefits of the "fast proof", iterative nature of Scrum.

By the way, if you're using Scrum for non-software projects, just think of 'developers' as the experts who turn your vision into reality. The same principles will apply.

Avoid "Scrumishness"

Scrumish is a technical term we just invented. It means not following part of the Scrum framework because it's causing problems, but doing so without solving these underlying problems.

It's fine to adapt Scrum to suit, but if you do this by sweeping problems under the carpet you'll tend to trip yourself up.

For example, if you find the Development Team has only one expert in a particular type of work, as Product Owner you might start assigning them this work. Yes, this would mean the work is done well, but it also means the team is less cross-functional and is no longer self-organising. In turn, this can mean motivation and capability fall. It'd be better to help the team come up with a solution like peer coaching that solves your problem without scrumishness.

» *Agile role transition workshop*

Traditional project management roles and Scrum

Another sign of scrumishness is hanging onto the roles and responsibilities of traditional project management.

There's plenty of room in Scrum for people who have been Project Managers and Business Analysts, but those roles don't exist. The same work is done, but there's no one-to-one match between traditional project management roles and Scrum roles.

Imagine you're a Business Analyst with a manager who sets your priorities, and now you're working as a Product Owner. Deciding priorities is a key part of the Product Owner role so, if your manager still makes these decisions, you add an extra step and risk turning your Sprints into crawls.

Value the Scrum Values

The Scrum Values are ways people interact: with commitment, courage, focus, openness and respect. In a successful Scrum Team, people interact in ways that make the team most productive.

Here's how Scrum.org sums up the Values:

- **Commitment:** People personally commit to achieving the goals of the Scrum Team
- **Courage:** Scrum Team members have courage to do the right thing and work on tough problems
- **Focus:** Everyone focuses on the work of the Sprint and the goals of the Scrum Team
- **Openness:** The Scrum Team and its stakeholders agree to be open about all the work and the challenges with performing the work
- **Respect:** Scrum Team members respect each other to be capable, independent people

» *Scrum Values poster*

Scrum Values and the Product Owner

If you don't think the team or individuals are demonstrating the values, you can always raise this with your Scrum Master or at a Retrospective. And demonstrating the Values yourself means the team is likely to follow suit. Here's how a Product Owner can live up to the Values.

Commitment

You can both show commitment and inspire it in the team, and the two feed into each other.

Keeping the Backlog shipshape demonstrates commitment which the team pick up on.

"When people know exactly what they need to do, it's full steam ahead," Rebecca says. "As soon as the Backlog starts to lag, then the motivation can start to lag as well."

On top of that, if you're enthusiastic about the work, and present a compelling vision, you give the team a strong drive to deliver. A passionate Product Owner is a powerful motivator.

Courage

Your team will often come under external pressure, especially if Scrum is new or not well-understood across the organisation. Senior managers push for pet features mid-Sprint. Other teams try to poach your people or grab their time. Marketers nag you to move up release dates. The Product Owner needs the courage to stick up for the work and the team and the Scrum process. Your Scrum Master will be a key ally in this.

You'll also need to stick up for the customers when working with developers. Experts such as developers often subconsciously assume that others are as technically clued-up as they are, which ain't necessarily so. This is where your in-depth knowledge of your customers, and the way they'll use your product, is vital.

Focus

As mentioned earlier, the Product Owner focuses on the work of the Scrum by being available and responsive.

You'll also want to make sure your Development Team are working full-time on the Scrum. If they have to do other work they'll be less productive. That's because it takes time for our brain to switch tasks. If people outside the Scrum are giving other work to the team, you or the Scrum Master should step in.

Openness

Openness is especially important when it comes time to inspect and adapt. Product Owners should be open when giving and getting feedback.

And if you take some time to make your feedback specific and actionable your developers will be able to implement it faster. And the sooner you can provide feedback the easier it will be to implement.

A committed Development Team will also always be looking for the best way to do things.

“If the team has a new way of doing something or they’ve had some new ideas, you want to be open to thinking about other solutions,” says Rebecca.

Respect

As well as treating your team mates with respect, respect their expertise. Scrum is based on the idea of the right people making the right decisions.

This means that the Development Team decides:

- how to implement your vision (so your Backlog should describe outcomes not features)
- how much effort is involved (so don’t override their estimates)
- what items they forecast completing for each Sprint (so don’t pressure them to do more)
- and how the work will be tackled by the team (so don’t assign tasks).

Of course, if you don’t understand why a decision has been made, you absolutely should ask. Agile is all about the conversation.

Stand atop the pillars

The three pillars of Scrum are transparency, inspection and adaptation. Together these make it easier for the right decisions to be made at the right time.

"The best part about Agile is that day-to-day knowledge of what is happening," says Jess Limbrick, a Product Owner for New Zealand's Life Education Trust. "Being included throughout the entire process gives you the ability to respond to change really, really quickly," she says.

Transparency

Transparency means the work is clear and visible. It's clear because you build a shared understanding of what's involved. This might be through team-forming activities such as creating a Team Charter, or Scrum processes like Sprint Planning, Refinement or agreeing on your Definition of Done. And it's visible because the Artifacts make it so.

Transparency helps build trust. "Without trust," Rebecca says, "you're not going to be a high-performing Scrum Team."

As Product Owner, you're responsible for the Product Backlog, so you'll want to make sure the team can access it at any time.

You'll also be able to make use of the other tools that make the work visible.

At Boost we aim for maximum overview for minimum overhead (in line with the Agile principle of simplicity — the art of maximizing the amount of work not done). In practice this often boils down to a visible, physical Scrum board, and the digital equivalent in which we also track Velocity (the average amount of work completed over a number of Sprints).

"The tools are there for the team. They're a good gauge for the Product Owner, but they shouldn't be used against the team which I think sometimes a lot of tools are," says Rebecca. "For example, Velocity helps teams know how much they should commit to. It's also helpful for the Product Owner to know how much work needs to be ready for the team to pick up."

Inspection and adaptation

Regularly checking in on your product and process enables the iterative improvement at the heart of Scrum, and a successful Scrum Team.

Effective Product Owners take advantage of these look and learn loops. They are engaged at Retrospectives. They prepare for the Review so they can learn as much as possible from that Increment of the Product, ideally getting stakeholders along to give their feedback.

Deliver value

A practical way for the Product Owner to ensure the team delivers maximum value is by delivering working solutions at the end of each Sprint. You want the Increment at the end of the Sprint to be releasable, even if you choose not to release it.

In order to do this, make sure your Definition of Done will produce a releasable solution. For software this might include code review, testing, integration and documentation. For a non-software project like a marketing brochure it might cover things like proofreading, sign-off and a print-ready PDF.

If you don't release you still get value from what you can learn from, for example, beta testing. If you do release, you deliver value to your customers and you gain value from what you learn about their use of your product in the wild.

Thus, with each Sprint, you take a step closer to achieving your vision, helping you make a bigger impact and a better world.

Next we look at where the Scrum Team sits in the world. How do Product Owners work with the wider business and beyond?

Working with Stakeholders

The key feature of successful Scrum projects is having a Product Owner who has been empowered by their stakeholders.

Section contents

Who are your Stakeholders?

In Scrum, a Stakeholder is anyone with a vested interest in the product who is not part of the Scrum Team.

As Product Owner, you can think of stakeholders as anyone with an interest in or an influence on the product. These are the people who'll help you discover, develop, release, support and promote the product.

These Stakeholders include:

- customers
- key decision-makers in your organisation
- project sponsors
- colleagues who work with and understand your customers
- people from each part of the business needed to deliver the product
- professional worrywarts like legal (sorry, those in risk-sensitive roles)
- external parties such as donors or regulators.

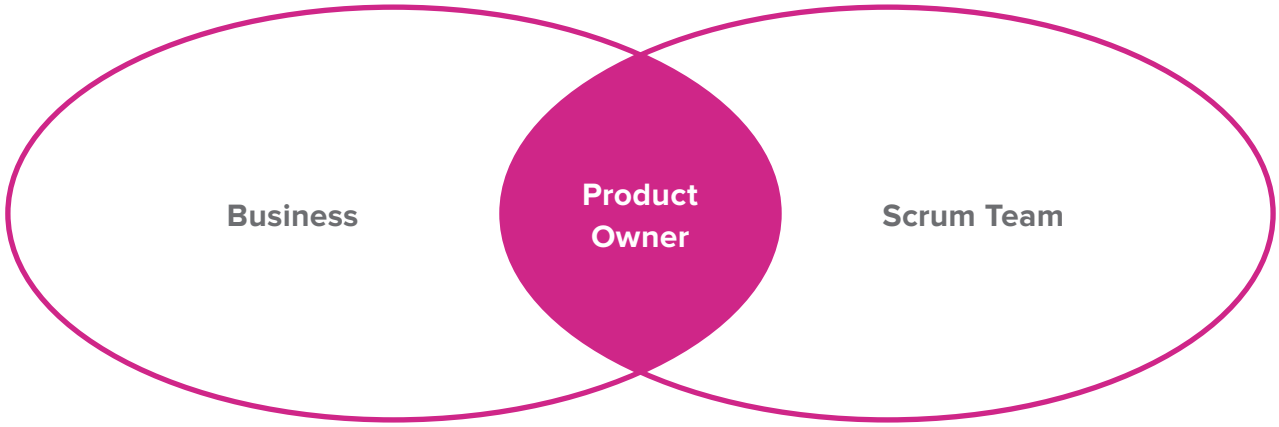
» *The GO Product Roadmap* — Roman Pichler

The Product Owner is a bridge between the business and the builders. You make sure both Stakeholders and Scrum team get what they need to succeed.

Learn how to team up with Stakeholders to help your Scrum deliver maximum value. This chapter shows you how to:

- set expectations about Scrum and the product
- plan how you'll work with different types of stakeholders
- get the support you need to succeed.

It'll be followed by a chapter on one specific and important way you work with stakeholders: making sure you're building what your customers need, a.k.a. product discovery.



Set expectations for the product and process

If Agile and Scrum are not well understood in your organisation, make sure all stakeholders understand why you're working that way and what's in it for them. Let them know:

- how and what you'll communicate
- what the deliverables and artifacts will be
- what meetings or events they might attend
- their level of engagement, including:
  - what you'll want from them
  - when
  - for how long.

Sharing the 'What is Scrum?' chapter of this Guide may help with this.

Getting key stakeholders involved in the discovery process gives them direct experience of the process and input into the product.

Jess Limbrick is Project and Partnerships Manager at Life Education Trust, a charity working to empower and educate Kiwi children to make healthy choices. She's been Product Owner for a number of Scrum projects. For Jess, a key stakeholder is the Trust's CEO John O'Connell. John was heavily involved in the discovery phase of the Trust's first Agile project. This meant he had clear expectations of what the work would be and how it would run.

"Through his involvement at that first stage, it's easy to explain things because he understands how it all works," Jess says.

In our next chapter, we'll show how to use a discovery process to set expectations for the product.

Set expectations for success

There are lots of ways to measure success. Success Sliders are a good way of getting agreement on conflicting goals (time, budget, scope, quality etc). Getting this agreement helps you confirm how you'll know if your project is a success.

Rob Thomsett introduced the idea of project Success Sliders in his book Radical Project Management.

How to run the Success Sliders activity

Step 1
On a poster or whiteboard, draw a grid of 6 rows by 5 columns.

Step 2
Label the rows with your success factors. In the example below we've chosen common factors for software development projects.

Number the columns 1–5, with 5 being the most important.

Step 3
Place post-it notes in column 3 of each row.

Step 4
Tell everyone to work together to decide which column each post-it should go in. More important factors go under the high numbers, less important under the low. You can have more than one post-it in each column. The column that each success factor sits in gives it a numerical value.

Here's the catch: the total value must equal 18.

This means that if you raise the value of one factor you'll have to lower the value of another. The team has to agree how to balance the values for each factor to stay within the magic number.

Putting a limit on the total value reflects the real-world constraints that all projects face.

» *Online Success Sliders tool*

Set expectations for failure

In Scrum you learn by getting solutions in front of your customers. Sometimes these solutions won't give customers exactly what they need. Let your stakeholders know that these failures are not only expected, they are beneficial. Each failure can be a win, because of the knowledge you gain. If there are no failures, you're probably not discovering anything valuable.

Clarify what you need

As well as clarifying these expectations above, you'll also want to make clear what you'll need for the project to be as successful as possible.

This includes things like a space for a colocated Scrum Team, tools such as physical whiteboards and digital Agile project management software, along with your communication tools.

Empowerment — The key thing you need from your stakeholders

In our experience, the key difference between successful and unsuccessful Scrum projects comes down to how empowered the Product Owner is.

With empowerment comes:

- time
- authority
- the vision.

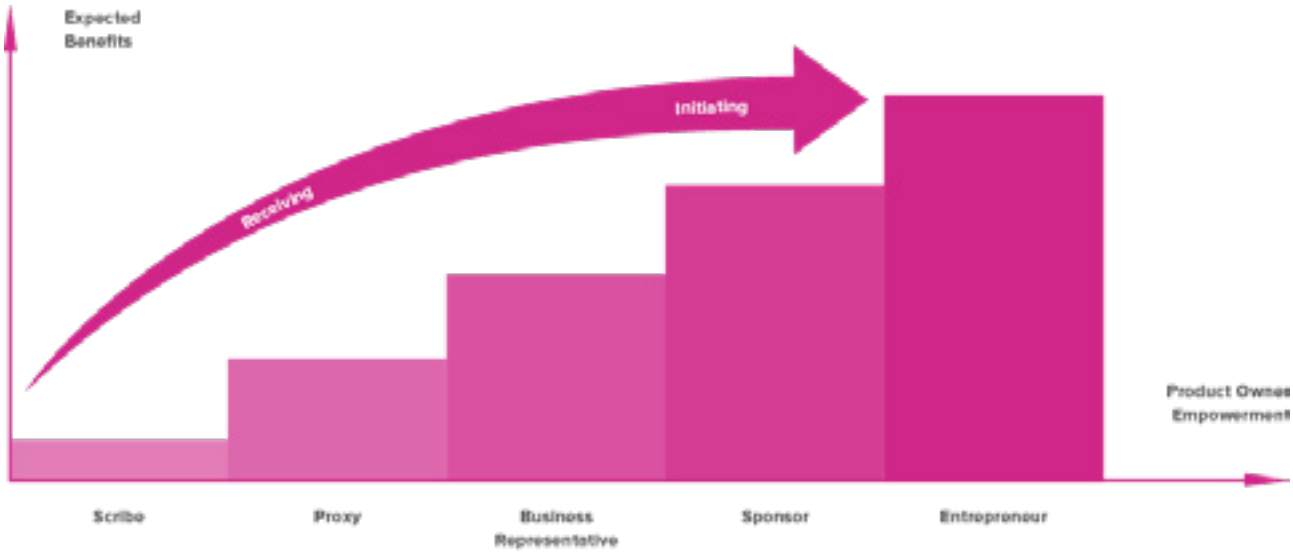
The more time you have to devote to the Scrum team, authority you have to make quick decisions, and ability you have to drive the vision, the more successful the project is likely to be. We'll look in more detail about how you develop the vision in the next chapter on product discovery.

"I know definitely that if I can put in a little bit more time into the project, we get a better outcome," says Jess Limbrick.

At the Trust they try to limit Jess's non-project workload for the duration of the project.

"We look at what other things coming up that I can either delay slightly or hand off," she says.

Because Scrum relies on the power of self-organising teams, it works best when the Product Owner is also self-directed rather than being a conduit for — and waiting on — decisions made by others



The Product Owner empowerment spectrum

You can see this as a spectrum: Scribe > Proxy > Business representative > Sponsor > Entrepreneur. The further along the list you sit the more empowered you are to drive the product, and the more successful the product is likely to be.

Because the extra cost to an organisation of having an empowered Product Owner tends to be small, relative to the overall cost of a project, and the benefits tend to be large, it makes sound financial sense.

How an empowered Product Owner makes decisions

While it's vital for a Product Owner to be decisive, this is not always easy, even if you've been empowered. You'll often have many stakeholders who have even more ideas about your product. If you don't listen you'll miss opportunities and lose buy-in. If you listen to everyone you'll end up with a product that tries to be all things to all people and so satisfies no-one.

To help manage this tension:

- take the time to talk to your stakeholders
- record their input so they see they've been heard (via the Backlog for example)
- base your decisions on how well suggestions help deliver the vision
- make decisions quickly so you don't slow down the Scrum Team.

Types of Stakeholders

Here are some ways to think about the different types of Stakeholders in Scrum.

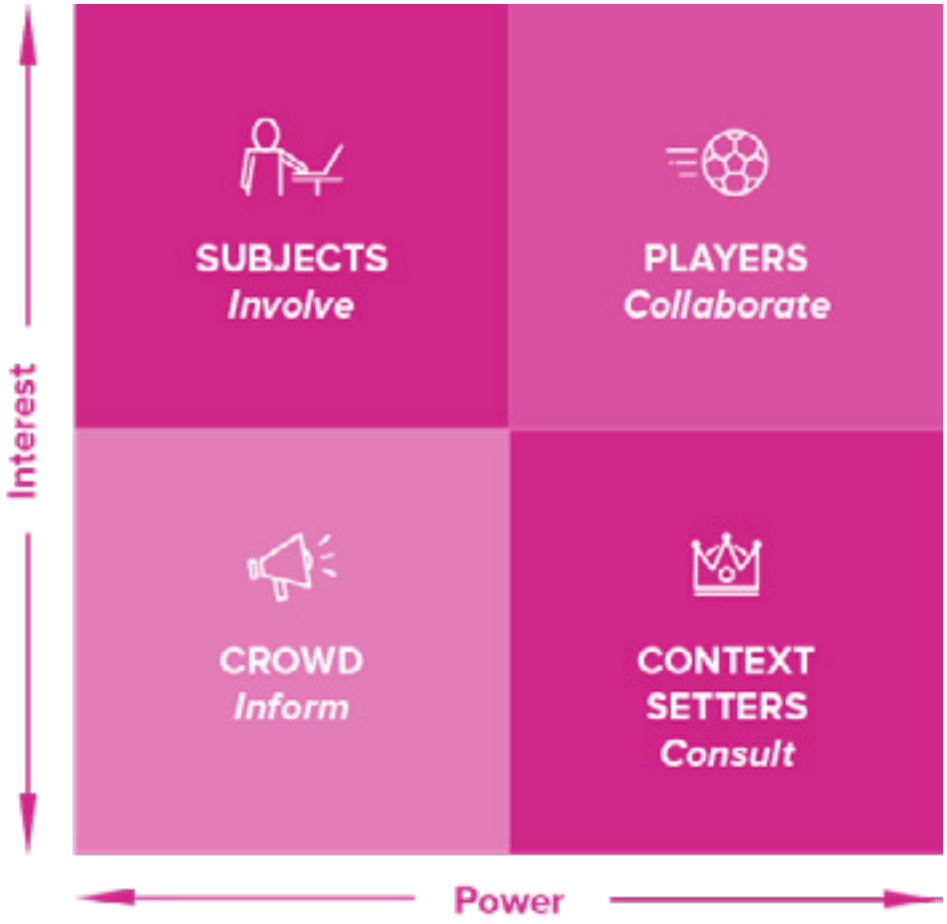
Customers (and their proxies)

Customers are your most important Stakeholder. They're almost a category in their own right. Without customers there's no product. The key way you work with customers is by finding out what they need.

Customer proxies

Within your organisation there are a number of people who understand what makes your customers tick. Customer support, contact centres, front of house, and sales staff talk to customers daily and develop a good understanding of their needs and pain points. Your marketing, user research and analytics experts dig into customer behaviour and motivations.

These people tend to have a vested interest in your product. They'll be supporting or selling the product so will want to understand it well, and be confident it delivers what your customers need. The same goes for people who support internal products or systems. This means that customer proxies are valuable allies.



Mapping Stakeholders to plan how you'll work with them

Product Ownership specialist Roman Pichler recommends breaking down Stakeholders by interest and power, using the grid set out by Colin Eden and Fran Ackermann in their book *Making Strategy.*

Write out your Stakeholders (or groups of Stakeholders) on post-its and place these on this grid, with high interest high power stakeholders top right, low interest low power examples bottom right and so on.

You can then base how you work with each group by their place on the grid.

Collaborate with the Players

Treat the high interest, high power Players as your partners, and collaborate closely with them. Try to get them along to your discovery sessions to understand your product and also to Sprint Reviews to see the progress.

Because knowledge is power, you can think of the customer proxies as Players.

Consult the Context Setters

Low interest but high power Context Setters influence the environment in which your work takes place. This might include things like its priority, visibility or how it's resourced. Their power makes it a priority to spend time with them, listening to their views and keeping them informed. Focus on how the project impacts (and ideally benefits) their area of influence, not on the nitty gritty details.

Involve the Subjects

Subjects have high interest but low power. Often their interest comes from being affected by the product. This makes them keen to influence so give them opportunities to provide feedback, such as getting them along to Sprint Reviews.

Inform the Crowd

As the Crowd have low interest and low power, the amount of time you invest in them can also be low. Keep them informed through regular updates via whatever broad-reach communication tool your organisation prefers.

Planning your work with Stakeholders

You can help out your stakeholders in Scrum by highlighting what's coming up for them and the product.

The more time you're going to want from people, the more warning they'll need. And the further you get into the project, the more detail you'll be able to give them.

The one Scrum Event that stakeholders can attend is the Sprint Review. This is their chance to follow the progress and provide feedback. It's also your chance to demonstrate the value you're delivering (not just features you've built, but the outcomes you're achieving). Since people, especially Players, tend to be busy, it helps to get the Review in their calendars well in advance.

Prepare for Sprint Events

As Product Owner, you'll often want stakeholders to give you input into the Backlog (though the final decisions are yours). It helps to book in time to gather this input before you have your Sprint Planning and Refinement meetings with the Scrum Team.

"Where things cross over with other people's work, I try to make sure that we were a couple of steps ahead of what the Scrum team was going to need," says Jess.

» *Product Roadmap template*
— Roman Pichler

Chart the path ahead with a Product roadmap

To make it easier to communicate your future plans for the product to your stakeholders and Scrum Team, it helps to map out a high-level plan.

Focusing your roadmap on the goals that each step in your plan aims to achieve makes it easier to show stakeholders the benefits of investing time and money. It also gives your Scrum Team motivation for each stage and an opportunity to think ahead.

Working with stakeholders — the Life Education experience

Jess has been Product Owner for a number of different products and projects. This means she may not be the owner of a system before the project starts. She works with the owner of that system during the discovery process and can bring them along to Sprint Reviews to get feedback. Working closely with them means that it's easy to hand back the system once the project is over.

As each phase of work has been completed successfully, it's easier to get the time to devote to the Scrum Team, time from stakeholders, and authority to make decisions.

"I think the success of the past makes it a lot easier as you continue," said Jess.

Working with external stakeholders like donors and regulators

A new project that the Trust is working on has external stakeholders in the form of corporate donors. While on a day-to-day basis they have low interest and low power, they have high interest in the results of their support and their donations are vital for delivering the product.

“Our corporate dollar obviously sits in a totally different space because they’re very much interested in what you’re doing, in how they can tell the story of how they’re helping,” Jess says.

“We already have a quarterly update process where we keep them in the loop with how things are progressing,” she says. “After that we’ll probably do something with the kids, getting them in front of a camera and letting them talk about what they’ve learned.”

Having an inspiring vision for your work also helps get donors on board initially.

Because they are very interested in seeing the value their dollar is delivering it can be worth inviting donors along to Sprint Reviews.

Like donors, regulators can also be extremely influential, and tend to be more interested in the final product than the ongoing process.

Building buy-in

> **Much of the work of the Product Owner is about building buy-in with the project. The great thing is that Scrum and Agile help with this.**

Because Agile projects focus on regularly delivering working solutions you can regularly demonstrate the value of the product and project. Every Sprint you have a Review in which you can not just tell Stakeholders what you've achieved, you can show them.

The Scrum pillar of Transparency means you display your progress via physical and digital Scrum boards for all to see, furthering their buy-in.

Since Scrum favours face-to-face communication, you can answer questions and tease out implications in real time rather than in dribs and drabs, through things like email or reports.

The way you carry out your discovery work also helps you build buy-in. You focus on outcomes, not features, helping you come up with an inspiring vision that motivates the organisation. Discovery artifacts such as your Pragmatic Personas, Press Release and Elevator Pitch are powerful communication tools. Having key metrics focuses priorities and lets stakeholders track progress at a glance.

In our next chapter, we'll detail how to do this discovery work.

Product Discovery

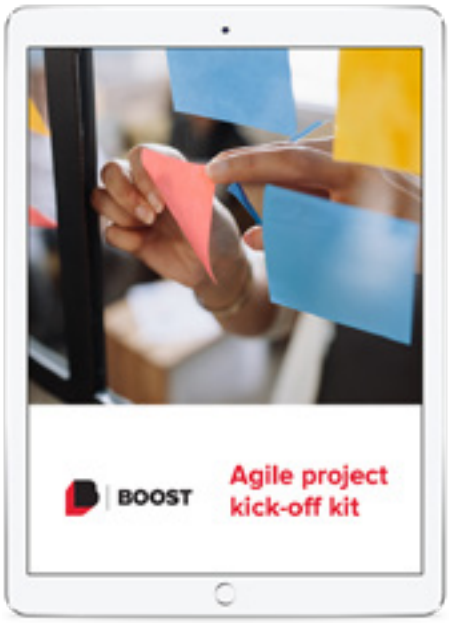
Product discovery is the art and science of finding what product your customers need and how they want to use it.

Section contents

Project Kick-off Kit: Discovery in a day

With some projects, time, money and team schedules mean you want to complete your discovery work in a concentrated burst. With this in mind, we've created the Agile Project Kick-off Kit.

» *Download the Agile Project Kick-off Kit*



This one-day product discovery workshop helps you:

- align both your Scrum Team and stakeholders behind the product vision
- get clear about the benefits you'll bring your customers
- prioritise only those features that will bring these benefits
- set expectations of what you'll deliver
- tackle the prototyping and testing in your early development Sprints.

Discovery exercises from the Kit:

- Vision Presentation
- Press Release
- Elevator Pitch
- Pragmatic Personas
- User Story Mapping
- Prioritising your Minimum Viable Product

» *Product Vision Board* — Roman Pichler

» *Opportunity Canvas* — Jeff Patton

» *Rocket Surgery Made Easy* — Steve Krug's guide to usability testing

» *How to run a Google Sprint* — Boost blog

» *Idea generation methods* — Interaction Design Foundation

» *The Sprint Book* — Jake Knapp

Get a step-by-step guide to creating a product that makes a difference to the lives of your customers and achieves the goals of your organisation.

Product discovery starts with an idea. It might be yours, a colleague's or come from customer feedback. You might be creating software or something else entirely. Your customers may be internal, external or a mix of both. Either way, as Product Owner it's your baby. Here's how you can find out if the idea has real value, and how to realise maximum value if it does.

To do this you come up with your:

- Product vision
- Product strategy
- Prototype solutions
- Minimum Viable Product (MVP).

As you go, remember the ABC of Agile: Always Be Capturing. Record the results and artifacts of your product discovery work. You'll want to refer back to them when you're building the product. Make them visible. Share them with your stakeholders, it's a great way to keep them in the loop.

Your product discovery team

Product discovery involves both the business and the builders, stakeholders and the Scrum Team. Like most team sports, it's OK to sub in and out. Not everyone needs to be involved all the time.

Stakeholders

The stakeholders with a stake in product discovery are:

- high interest, high power "players"
- customer proxies — stakeholders who understand your customers
- customers.

Scrum team

The more you involve the Development Team and Scrum Master in product discovery, the better they understand and are invested in the product. That's because there's nothing like face-to-face time with customers to get a direct and in-depth understanding of what they need.

Sometimes it can be hard to have all the Scrum Team involved in all the discovery work. If, for example, you're partnering with a development company you might not be able to get the Scrum Team along the whole time.

If you're running the one-day discovery workshop we cover in the Agile Project Kick-off Kit, it's well worth getting the Scrum Team along. You may also be able to get some or all of them along to the customer research or prototyping phases (especially if you do this via the intensive five-day Google Design Sprint — we cover the Google Sprint later in this chapter).

Dual track — discovery in tandem with development

Discovery doesn't end when development begins. As you deliver each Increment you get new opportunities to learn about what your customers need. So you can think of discovery and development as intersecting cycles that feed into each other.

Sometimes you may need a burst of product discovery work. This is obviously the case with a new product. And sometimes an existing product needs a shake-up to keep it competitive

Product vision

The Product vision explains why you're building the product.

It describes how your product will make the world a better place, showing how it will help your customers and achieve the strategic goals of your organisation.

This provides both a destination and inspiration. It helps get stakeholders and the Scrum Team behind the product.

In our Kick-off Kit we do this via a Vision Presentation. This then gets fleshed out, along with the strategy, through the Press Release and Elevator Pitch exercises.

» *Vision Presentation*

» *Press Release*

» *Elevator Pitch*

Getting one of your head honchos to present the vision shows that the C-suite are behind the project and takes advantage of their experience communicating why work is important.

You can show how your product will help achieve the goals of your organisation by using the language of your overall strategy, and tying it back to what matters to your stakeholders.

For example, Life Education Trust were building a system to help them make a fundamental, mission-critical change in the way they worked. The key stakeholders were their educators, a change-averse set of internal customers.

"The key word in our vision is empowerment. We are trying to give kids the knowledge to make their own decisions about their health and wellbeing," says Jess Limbrick, Product Owner for the work.

"We had to let them know why we needed to make the change, why it was going to be better, why it delivered a better outcome for kids. Because that's ultimately why educators are with us."

You can sum up your vision in a one-page Product Vision Board.

» *Product Vision Board* — Roman Pichler

Product strategy

Your strategy is the way you'll achieve your vision by giving a set of customers something they need that they can't get elsewhere.

In other words it wraps up and defines your:

- vision
- target customer
- point of difference in the market.

To do this at a kick-off workshop we often create:

- a Press Release — imagine the better world you'll create and work back to identify how you achieved this
- Pragmatic Personas — understand the priority customers you're targeting
- an Elevator Pitch — distil the Press Release and identify the product's competitive edge in the marketplace.

It always pays to keep tabs on your competitors. But this is especially important as you develop your strategy.

You need to understand the market your product will be entering.

» *Press Release*

» *Pragmatic Personas*

» *Elevator Pitch*

Metrics — how you'll measure if the product strategy works

Choosing your key metrics helps you clarify what value is, prioritise work that delivers this and measure how well you're doing.

At Boost we use a single metric that we call One Number. Having a single target is great for communicating with stakeholders. We put it up on a big screen, so it can demonstrate what's been achieved and build buy-in.

Jess Limbrick from Life Education found it worked well for them. "It was clear,

concise and easy to keep our focus narrowed to achieving a key performance indicator. The live number was visible to the whole team which helped with buy-in."

Places like Google, Facebook and eBay go for between two and six metrics, using something called OKRs — Objectives and Key Results. These follow this pattern:

- Objective
  - Metric 1 — why you're measuring it
  - Metric 2 — why you're measuring it
  - etc

For example:

- Double sales of our widget within 12 months
  - Increase daily traffic to our website to 1000 by September — so we can gather more leads
  - Net Promoter score over 90 by July — in order to increase referrals
  - etc

To maximise the value you create, set yourself ambitious targets.

Opportunity Canvas

You can sum all this up on an Opportunity Canvas.

» *Opportunity Canvas* — Jeff Patton

Prototypes — create and test solution options

Sometimes the outline, if not the detail, of your product is self-evident. But often there are a number of ways you can achieve your vision and implement your strategy.

Prototyping is a way to test that the assumptions you've made about your customers and your product in your strategy are correct without the cost of building the full product.

Early on there's much you don't know. Is there in fact an opportunity and, if so, what solution will make the most of it?

You can work this out by combining ideas from Design Thinking and Lean. You:

- understand your customers
- get clear about the problem you're solving
- come up with a range of solutions
- prototype solutions
- test the prototypes
- repeat.

(In Design Thinking these steps are called: Empathise, Define, Ideate, Prototype and Test.)

You can do some or all of this before your first Sprint or in your early Sprints.

Google Design Sprint

One of the most well-supported ways of doing this is the Google Design Sprint (it's different to the Scrum Sprint but has the same Agile underpinning). The Google Sprint is a tried-and-tested, tightly-structured five-day process for creating and testing solutions.

If you can't get a team together for five solid days, we cover how you can achieve the same goals over a longer term below. Even if you're not using a Google Sprint, you can pick up good tips on how to come up with and test solutions by reading the Sprint book or our day-by-day guide.

» ***The Sprint Book*** — Jake Knapp

» ***How to run a Google Sprint*** — Boost blog

Understand your customers

If you and your product discovery team don't already have a good understanding of your customers, here are places to look and approaches to try:

- Existing customer research or testing
- User statistics, analytics and search logs
- Call centre or helpdesk questions — what problems keep coming up?
- Surveys
- Focus groups
- Usability testing an existing or competitor's product
- Customer interviews

These activities give you a detailed insight into your customers that you can use to create your Personas.

Face-to-face works best

For Life Education, Jess says that talking to their internal customers face-to-face got results.

“We knew that face-to-face we would be able to answer a whole lot more of their questions and gauge their reactions,” she says.

Usability testing and interviews are both face-to-face and one-on-one, which works especially well. Often you’ll combine the two. If you can put people at ease and dig into their needs and experiences in a conversational way, these two approaches can give you great insights.

To find out how to do usability testing, read Steve “Don’t Make Me Think” Krug’s classic — Rocket Surgery Made Easy.

Any time you have contact with customers, it’s worth asking if they’re happy to help you later. For example, if you run an online survey, why not ask participants if you can contact them later for user research. It’s a good way of building up a customer research mailing list.

Get clear about the problem you’re solving

Define the people your solution will help and the problem you’ll solve for them. Following the formula set out by the Design Thinking pioneers at the Stanford design school can help with this:

How can we help [type of customer],

To [accomplish a goal or perform an activity],

When [What makes it hard? What makes it rewarding?]

Note down the assumptions you’ve made in defining this problem so you can test them later.

- *Rocket Surgery Made Easy* — Steve Krug
- *Conversational interviews* — CrazyEgg

Come up with a range of solutions

Consider multiple solutions to solve your customers' problems. Your first solution may not be the best. To come up with a range of solutions, you're after idea-generation techniques that:

- are quick, collaborative, visual and tactile
- help you think outside the box
- give you time to think quietly
- avoid groupthink.

Once you have a set of solutions, pick the best. Using the designated decider and structured voting from the Google Sprint can help you come up with quick, focussed decisions. Check our Google Sprint how-to guide to see how this works.

» *Games for fresh thinking and ideas* — Gamestorming.com

» *Idea generation* — Interaction Design Foundation

» *How to run a Google Sprint* — Boost blog

Create prototypes

> **Create the simplest prototypes you can to test the assumptions most likely to scupper your solutions.**

These assumptions might be about your customers, your solution or your business environment. One way of digging into assumptions is to think about risks. What could go wrong that would make the product a failure?

Identify the riskiest assumptions and test these first. For instance, if your target customers don't have the problem you think they do, there's no point in testing whether your product solves it.

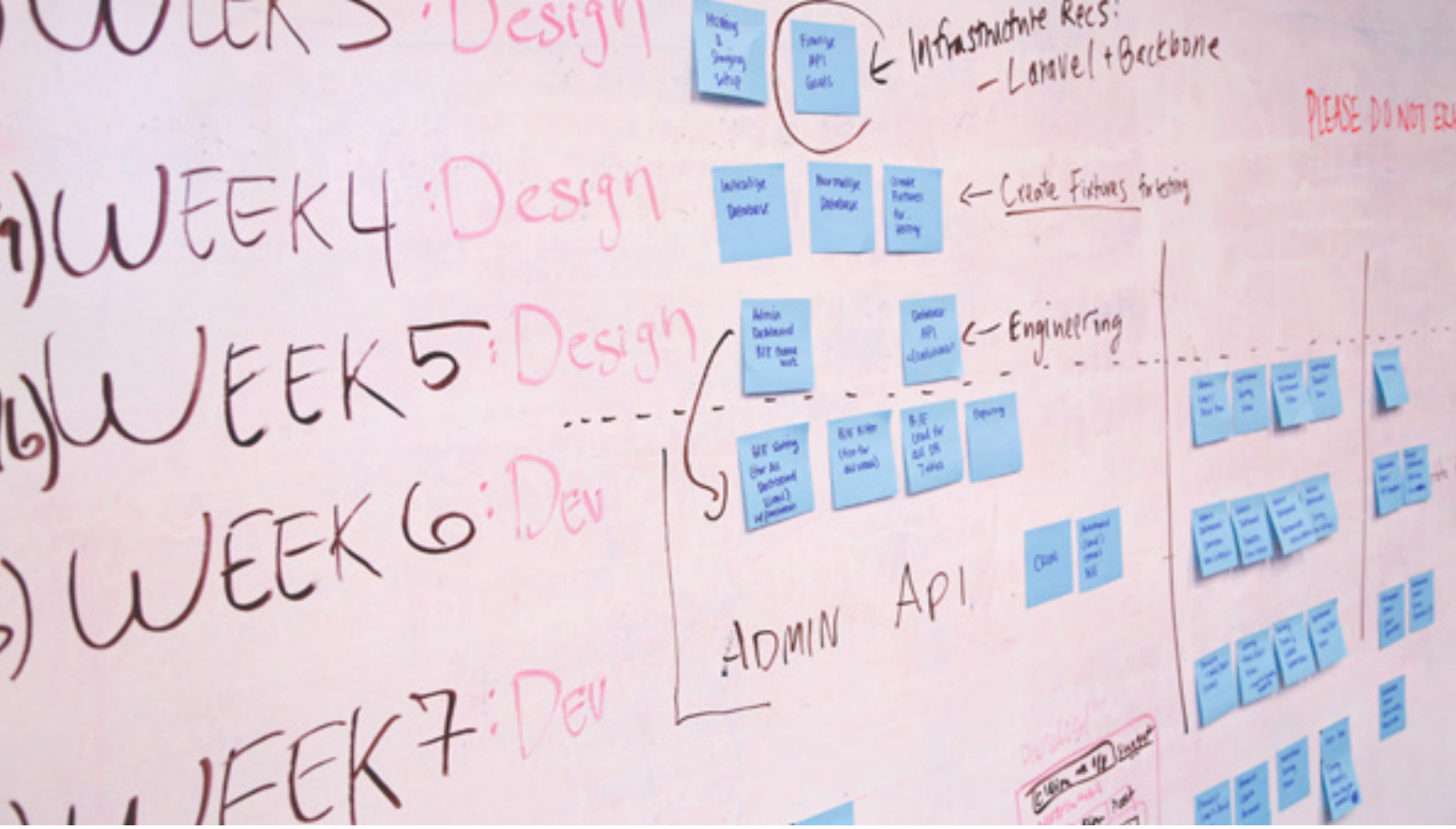
Start with the simplest, lowest-fidelity prototype you need to test these assumptions. That way you haven't put in unnecessary effort if your solution fails the test. In the Sprint book they recommend using a presentation package like Keynote or PowerPoint to create basic, clickable prototypes.

Test prototypes

In Rocket Surgery Made Easy, Steve Krug shows how to test prototypes at different levels of fidelity, from paper napkin sketches through to working products.

Here are a couple of other simple techniques for early prototypes:

- Describe the solution to customers and gauge their reaction.
- Mock-up a landing page promoting the product to see if people are convinced to click through.



Rocket Surgery also covers how to recruit test subjects. If you've already collected a mailing list of customers who have agreed to help, you're off to a flying start.

You, the Scrum Team and your discovery stakeholders should watch the tests together. Seeing is believing and nothing better demonstrates the gap between our expectations and how customers actually use our products than watching them in action.

Jess from Life Education has started working on a new, public-facing product for kids. Testing the interactive prototype threw up plenty of surprises.

"Some of the things that I thought that might not be obvious were. Then there were some other things that I thought were obvious that weren't," Jess says.

Test in pairs. That way one can concentrate on building a rapport with the test subject, the other on taking notes. Everyone else can watch via video conferencing.

Meet straight after watching the test and agree on your actions.

Repeat

Feed what you learn about your customers into your next prototype, or your Minimum Viable Product.

Minimum Viable Product — finding it via User Story Mapping

To get to the point you can start building your product you need to map out how your customers will use it, and prioritise the minimum features needed to make this experience worthwhile: your Minimum Viable Product.

User Story Mapping

User Story Mapping guides you to the point that you can start writing user stories and gives you a visual chart showing the structure of your stories. These stories and this structure will guide your development work. We cover how to create a User Story Map in our Kick-off Kit.

User Stories

A user story is a short description of something your customer will do when using your product. The user story is written:

- from the point of view of a person using your product
- in the language that your customers would use.

Because the user story is written in language anyone can understand, and focuses on the customer and the benefit they get, it's a useful tool for communicating with stakeholders. It's clear why you're doing the work.

In a later chapter we'll go into more detail about how to get the most from user stories in Scrum.

» *User Story Mapping*

» *Prioritising your Minimum Viable Product*

Testing your MVP

Getting your early MVP in front of your customers is great but you can't rely on them to give you feedback unprompted, as Jess from Life Education discovered.

"We probably overestimated the likelihood of the educators telling us that something didn't work," says Jess. "It wasn't until we had more directed conversations that stuff really started coming up."

Krug's Rocket Surgery Made Easy shows how to have these directed conversations.

When to release your product

Once you have a set of features coming together as a unified whole that allows users to reach a goal they want to reach, you're ready to release a new product. Or, for existing products, you're ready when you have additional features that will make existing customers happier or attract new customers.

Before you release, prepare how you're going to learn from the live product.

Learning from your product once it's released

Once live, keep an eye on your analytics and bug reports. Talk to your customer service people about the questions they're getting from users. Provide a way for your customers to give you feedback on your product. Ask those who give feedback or suggest features if they're interested in getting sneak previews of new features. Use this group for your beta testing of yet-to-be-released Increments.

You'll also need to keep tabs on your key metrics. Have you hit your stretch targets? If not, what might you do in upcoming Sprints to get there?

Inspect and adapt your product discovery work

In Scrum you're always looking at the work you've done and trying to find ways to do it better.

With this in mind, you could focus a Retrospective on your product discovery work or even have a specific discovery retro. Document a couple of key actions that will improve your future discovery work. Then celebrate your successes!

The Product Backlog

Keep the Product Backlog in priority order with the top priority items ready for the Development Team.

Section contents

The Product Owner is responsible for the Product Backlog in Scrum. Find out what this means in practice.

> **The Product Backlog is an ordered list of everything that is known to be needed in the product. It is the single source of requirements for any changes to be made to the product.**
>
> – The Scrum Guide

Key characteristics of the Backlog

The Backlog is:

- the Product Owner's responsibility
- a team effort
- ordered with the highest priority items at the top
- more detailed the further up you go
- always changing.

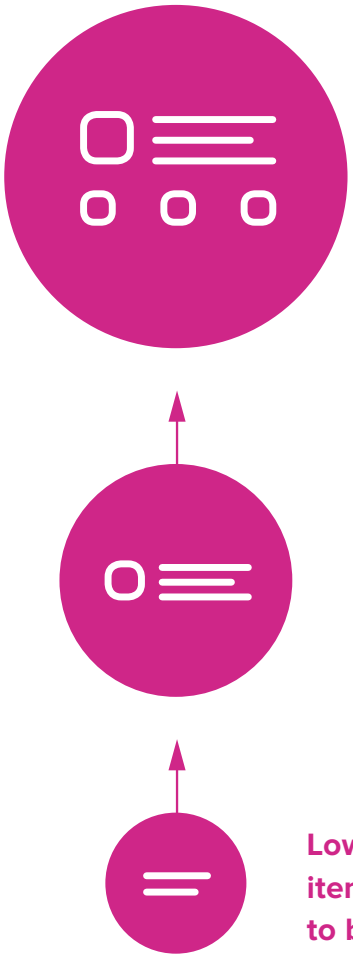
As Product Owner you decide what goes into the backlog and where it sits in the priority order. You do this with the help of your discovery stakeholders and the Development Team but the buck stops with you. Because it's a team effort you need to make sure all involved can access the Backlog.

The items at the top need to be refined so that they are ready to be brought from the Backlog into the Sprint. Before the start of each Sprint you'll make sure the highest priority stories are at the top and add enough detail to give the Development

Team a starting point for a discussion about what the item involves. You'll add more detail as a result of that discussion. This means the Backlog is like an inverted pyramid, with the highest priority items at the top having the greatest detail.

As you learn more about the product, you can tweak the priority order and add more detail to items that have moved to the top.

Highest priority items at the top of the backlog have the most detail



Lower priority items don't need to be as detailed

User stories as Product Backlog items

The Scrum Guide doesn't mention user stories. The individual items in the Product Backlog could be things like requirements, use cases, specifications, bug fixes, maintenance tasks or user stories. Most Scrum Projects opt for user stories because they are a great way of keeping the customer front of mind and helping Product Owners maximise the value delivered.

Here at Boost we usually go for user stories too, so that's what we're going to cover here. While the details may change if your Product Backlog items are not stories, the basic principles are the same.

Where do Backlogs come from?

The items in your Backlog start as a user need that you've identified in your product discovery work. Maybe you ran a User Story Mapping exercise. Or perhaps you've learnt something new about your customers' needs from a previous Increment or had a feature suggested by a stakeholder. Often this will just be a stub — a brief, meaningful label.

Tools for managing your Backlog

We often get asked which tool should be used to manage the Product Backlog. Where you keep your Backlog doesn't matter as much as what you do with it.

Your Backlog may be a box of index cards. For smaller projects a shared text document or spreadsheet can work well. There are also many digital tools, including CA Agile Central (formerly Rally), PivotalTracker, JIRA and Trello.

- *CA Agile Central*
- *PivotalTracker*
- *Jira*
- *Trello*

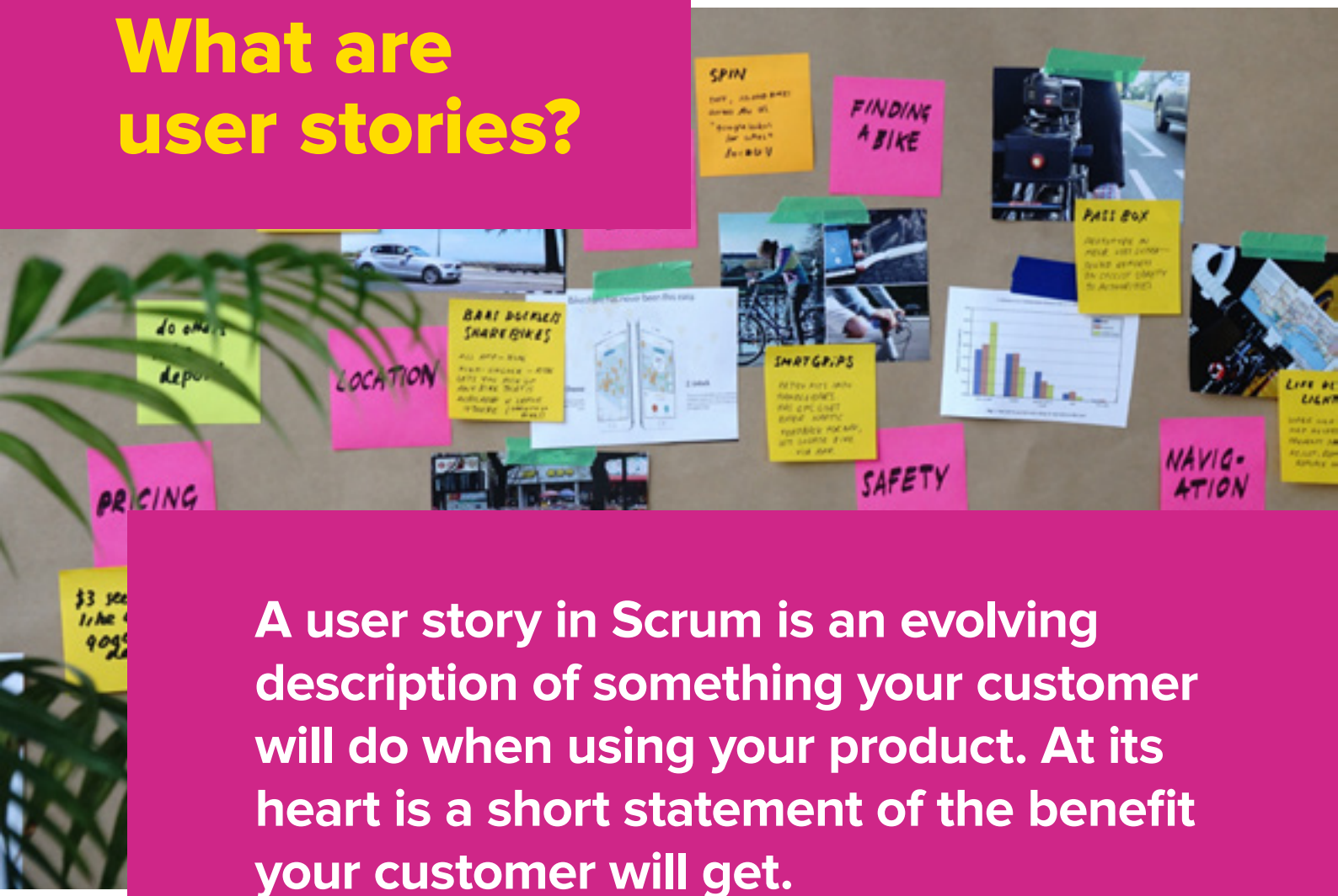
- *Ten product backlog tips*
  — Roman Pichler

User stories in Scrum

Find out how Product Owners can best specify the Product Backlog through User Stories. Get a template and tips for collaboratively creating effective User Stories.

Section contents

What are user stories?



A user story in Scrum is an evolving description of something your customer will do when using your product. At its heart is a short statement of the benefit your customer will get.

This is:

written from the point of view of the customer

in the language the customer would use.

The story of user stories

The origin story of user stories shows that they're all about the excitement of delivering products customers love.

Kent Beck told Jeff Patton that he came up with the idea of user stories when he was thinking about the stories users tell when they find that software is helping them out in cool ways:

"I type in the zip code and it automatically fills in the city and state without me having to touch a button!"

"If you can tell stories about what the software does and generate energy and interest and a vision in your listener's mind," Beck said, "then why not tell stories before the software does it?"

> **User stories start simple and get more detailed as they move up the priority list, getting closer to being built.**

- *Writing effective user stories* — Boost blog
- *Acceptance criteria for user stories* — Boost blog
- *Prioritisation and estimation techniques* — Boost blog
- *Non-functional requirements as user stories* — Mike Cohn

Anatomy of a user story

A user story doesn't start with all the elements below, they get added as you go.

#0067

ID number

A optional unique ID, often assigned as you add the story to your digital story tracking tool.

HELLO I'M

Name

Also known as the stub, the name is a clear, concise, specific label. Pick a name that will make it clear what story you're referring to when you're discussing your stories

Description of the customer benefit

A simple statement of the story's who-what-why, using this structure:

As [actor] I want [action] so that [achievement].

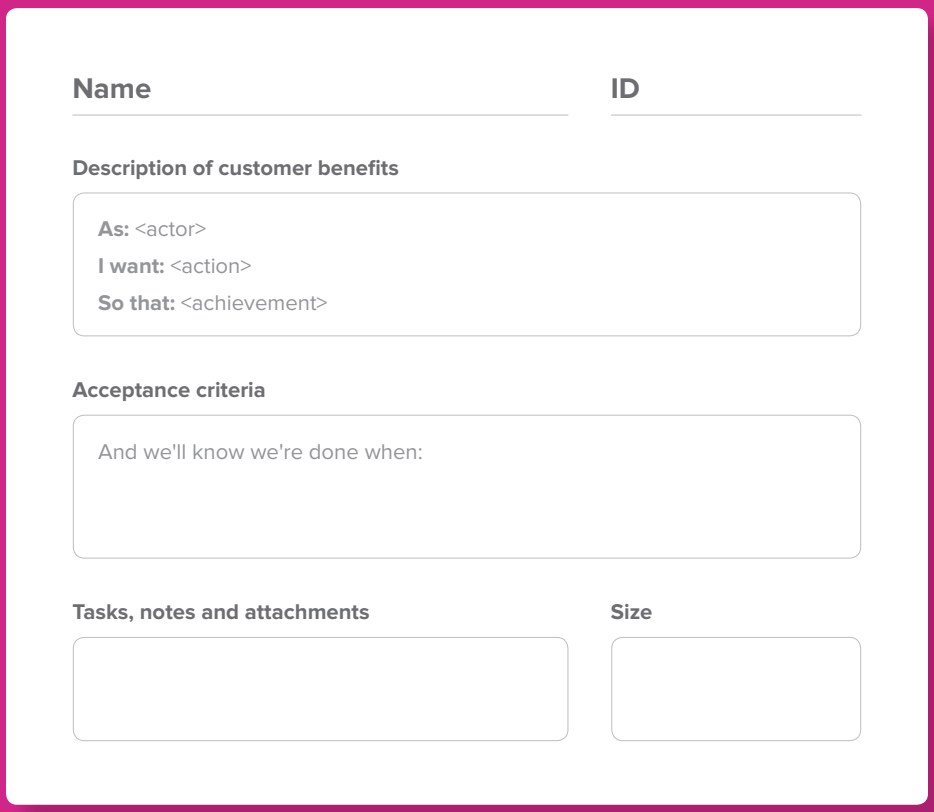
The **actor** will usually be one of your personas. This makes sure you have a good idea of who you are completing the user story for.

The **action** describes what will happen. It doesn't describe how it will happen because that's the domain of your Development Team.

The **achievement** explains the benefit the actor receives from the action. It shows why they want the action to take place and conveys the value of the story.

Acceptance criteria

Acceptance criteria define how will you know the story is done. These help the Development Team because they give more detail about how they'll deliver the benefit to your customer. And they help you as Product Owner because they show what you need to look for when you're checking the work.



Tasks, notes and attachments

Tasks are a breakdown of the steps needed to complete the story. The team come up with and record these.

Notes might include whether the story is part of an Epic, any dependencies and so on.

The team can also ask the Product Owner to attach things like wireframes, workflow diagrams, use cases etc to the story.

Estimation

This is the Development Team's estimate of the relative size of the story. It ensures they get a shared understanding of the effort and complexity in each story and lets them judge which stories they forecast they'll complete in a Sprint.

Status

What stage is the story in its journey from 'not started' to 'done'? You might record this by updating it in your digital tool and moving the story along a physical board.

Example user story

Imagine you're building a movie theatre website. Here's one story you might have:

Subscribe to the Cinerama newsletter ID 0032

Description of customer benefits

As Mandy Moviebuff **I want to** be able to subscribe to the Cinerama newsletter **so I can** get excited about the upcoming attractions, competitions and offers at my favourite movie theatre.

Acceptance criteria

- call to action content is displayed
- user can subscribe by supplying their email address
- user's details are stored in a database
- spam protection is in place
- user gets a friendly thank you message

Tasks, notes and attachments

Tasks:

- implement call to action
- embed javascript subscribe snippet from our newsletter tool button
- implement spam protection
- build thank you message

Notes:

- Epic: Create Cinerama News page on website

Attachments:

- Text for the call to action and thank you message

Size

M

Status

In progress

The stages user stories go through

User stories sit in the priority order you as Product Owner gave them in the Product Backlog.

> **User stories start simple and get more detailed as they move up the priority list, getting closer to being built.**

That way you don't waste effort elaborating user stories that you later learn are not needed.

Ron Jeffries described the three critical elements of user stories as Card, Conversation and Confirmation, and Jeff Patton has extended these into a five-stage loop:

Card: Write the outline of your story idea on a card (or add it to your digital tool).

Conversation: Discuss how it will be achieved with the Development Team.

Confirmation: Agree on the approach and how you'll all know it's done. Note this as the acceptance criteria.

Construction: The Development Team build and test the user story as part of the Sprint Increment.

Consequences: Show it off and see what you can learn from it.

INVEST in good user stories

You can remember the criteria for good user stories through Bill Wake's INVEST formula. User stories should be:

Independent: The story doesn't rely on other stories getting done. If stories are heavily dependent it's hard to plan, prioritise and estimate them. You can often solve this by breaking up stories differently or grouping them together.

Negotiable: As Product Owner you draft your user story card to start a conversation with the Development Team, not to detail exactly what they'll do. And the story may evolve as it gets developed.

Valuable: The story needs to spell out how the work will benefit your customer. When you're thinking about this, keep in mind how it helps you achieve the product Vision and strategy you came up with in discovery.

Estimable: If the card and the conversation don't give the Development Team enough information to estimate the size of the story, they can't pull it into the Sprint.

Small: You should be able to complete your stories in one Sprint. Otherwise it can't be part of a potentially releasable increment.

Testable: You need to know what you're going to test when the Development Team send it to you for acceptance.

Refining your Backlog of user stories

Stories need to be ready for the Development Team before the Sprint starts. The Scrum Guide calls this refinement (it used to be called grooming). The Guide doesn't spell out how or when this happens. At Boost we schedule in a refinement meeting towards the end of the current sprint.

» *Prioritisation and estimation techniques*
— Boost blog

Next we'll cover a summary of what goes into refinement.

Before your refinement meeting

As Product Owner you need to make sure that the stories at the top of the Backlog are ready for refinement. You'll work with your discovery stakeholders to flesh out and prioritise the stories. You can also get the Scrum Team to help you write them — but it's up to you to make it happen.

Before your refinement meeting, the Product Owner:

- makes sure they are in priority order
- adds the description of the customer benefits to the stub
- drafts the acceptance criteria
- adds any notes or attachments that will help the team understand the story.

During the refinement meeting

During the refinement meeting the Development Team:

- quizzes you to clarify the story
- discusses how they would complete it
- reviews its priority
- suggests changes so it meets the INVEST criteria
- estimates its size.

You add the results of these discussions to the story.

You may also want to have a Definition of Ready, a list of criteria the story has to meet before it can be brought into the Sprint. For example, this might include things like the need for stories to:

- follow INVEST criteria
- include acceptance criteria
- provide any content that's needed
- include any external contact details.

User stories during the Sprint

Sprint Planning

In this meeting the whole Scrum Team plan what you'll achieve in the Sprint, and how you'll achieve it. The more you've done in Refinement, the easier your Sprint Planning will be. This gives your Development Team more time to work out how they'll do it. Having discussed it in refining and estimated its size they can make the call about whether to add it to the Sprint Backlog.

Stories in the Sprint Backlog are often added to a physical board so everyone can see at a glance where they're at and how much work-in-progress there is at any time.

Daily Scrum

As they coordinate the day's work at the Daily Scrum, the Development Team will note progress made on stories, their next steps and anything that's stopping the work so that these blockers can be resolved.

Acceptance

When the story is done, the Development Team will let you know the story is ready for you to check if it meets the acceptance criteria.

Review

At the review the Development Team will demonstrate all the completed stories in action.

You'll probably have stakeholders at the Review. They won't have been involved in the whole conversation about the story so you'll need to explain why the story was a priority and what it achieved. Luckily the benefit statement sums this up in a simple description!

Feedback you get from the Review, and what you learn from getting the new Increment in front of your customers, may suggest you need a new story or changes to existing stories, including their priority. The Product Owner needs to make sure that any stories that have moved the top of the backlog are ready for the refinement meeting.

Retrospective

When you look back to identify ways you can improve, keep in mind if there any opportunities to develop your stories more effectively.

Pros and cons of user stories in Scrum

Understanding the benefits of user stories in Scrum helps you make sure your stories give you these benefits. And knowing the downsides helps you mitigate them.

Pros include the way that user stories:

- focus on the benefit to the customer
- minimise upfront effort
- evolve as you learn more
- make use of face-to-face communication
- have a common structure making it easier to compare, prioritise and estimate stories
- use language anyone can understand making them good for communicating with stakeholders.

» *Non-functional requirements as user stories* — Mike Cohn

Cons include:

1. It can be hard to write them for non-functional requirements, backend or maintenance tasks.
2. They can seem like a slightly casual way to capture requirements for people who are new to Scrum.
3. The fact that you repeat the process means it can become a rote task. This can lead to people cutting corners.

To resolve these you can:

1. Think of the ultimate benefit of these non-functional requirements or maintenance tasks. You should be able to express this in your user stories.
2. Show stakeholders stories at the end of their evolution so they can see the level of detail stories capture and how they are structured.
3. Vary how you do them, and who does what. Maintain the motivation by harking back to the vision, and how the stories are going to have an impact for the better.

Make multiple Product Owners work: A case study

Scrum specifies a single Product Owner but you're considering more. See how the National Library made having multiple Product Owners work.

Section contents

National Library had more work than one product owner could manage and still be able to spend time in the business.

The National Library is tasked with helping Kiwis capitalise on the collective knowledge of the nation. Within National Library, DigitalNZ was set up to make New Zealand's digital content easier to find, share and use.

Both National Library and DigitalNZ have their own websites providing access to catalogues, collections, services and other resources.

Boost are the National Library's development partner. We provide their Scrum Masters and Development Teams. They currently have two Scrum Teams.

Each of these Scrum Teams has one Scrum Master, three or four developers and, contrary to Scrum doctrine, at least three Product Owners.

Scrum: Only One Product Owner

According to the Scrum Guide "the Product Owner is the person responsible for managing the Product Backlog". The Guide is specific: "The Product Owner is one person, not a committee."

That's because having a committee reduces transparency and slows inspection and adaptation. The Backlog describes the priorities for the work but having multiple Product Owners can muddy priorities. With multiple Product Owners you have no single source of truth so it's not always clear who to talk to. And a committee is less likely to give quick feedback, slowing work down.

"It's very hard to listen to multiple voices," says Rebecca Jones, Boost Scrum Master for our National Library work.

So if you're going to have multiple Product Owners, you need to make sure you do it in a way that maintains transparency and aids inspection and adaptation.

- *National Library website*
- *DigitalNZ website*
- *AnyQuestions website*
- *Supplejack* — Boost blog
- *Scaling the Product Owner* — Roman Pichler
- *The Chief Product Owner on Large Agile Projects* — Mike Cohn

The problem: Large ongoing projects

National Library had more work than one Product Owner could manage, and still spend time in the business. Spending that time gives Product Owners the crucial understanding of customer needs.

“If we had one Product Owner, they would spend all their time being a Product Owner, creating stories, sizing stories, doing refinement. Eventually, they’d become disconnected with the actual business that they were supposed to be the Product Owner for,” says James Robertson, DigitalNZ Systems Manager.

Context: Why it might work

The National Library work had particular characteristics that fed into their decision to use multiple Product Owners.

Continuous value stream for an established product

The DigitalNZ and National Library websites and tools already exist. In fact they are the product of literally hundreds of Sprints. The work now is more a continuous value stream than a discrete series of projects.

Product ownership expert Roman Pichler says new products should have a single Product Owner but established products can need more. Young products need rapid decisions as you adapt to the large amounts you learn from getting them in front of customers. A single Product Owner is best-suited to this. Mature products grow and require more development than one person can stay on top of. At the same time, they tend to change less, and less often, reducing the potential drag caused by shared decision-making.

» *Scaling the Product Owner*
— Roman Pichler

If you have discrete pieces of work you'll often want to split those off into separate projects. Indeed this is what the National Library did when they wanted to refresh their Any Questions website for example.

Because it's an established project, the National Library have a team experienced in Scrum. When James started a few years ago the team had processes in place to make having multiple Product Owners work. But it would be much harder to set up these processes with a green team working on a greenfields project.

Clear split of responsibilities

For the National Library, the current state of play involves two Scrum Teams. One of these is for the National Library site and services, the other for DigitalNZ. Both Scrum Teams have multiple Product Owners, each responsible for a different stream of the work.

Take the DigitalNZ Scrum Team for example. Product ownership is effectively split between three Product Owners like this:

Rowan: DigitalNZ website and service — the features, functionality and usability of the website itself

Dan: DigitalNZ Supplejack API — Supplejack harvests content for the DigitalNZ website

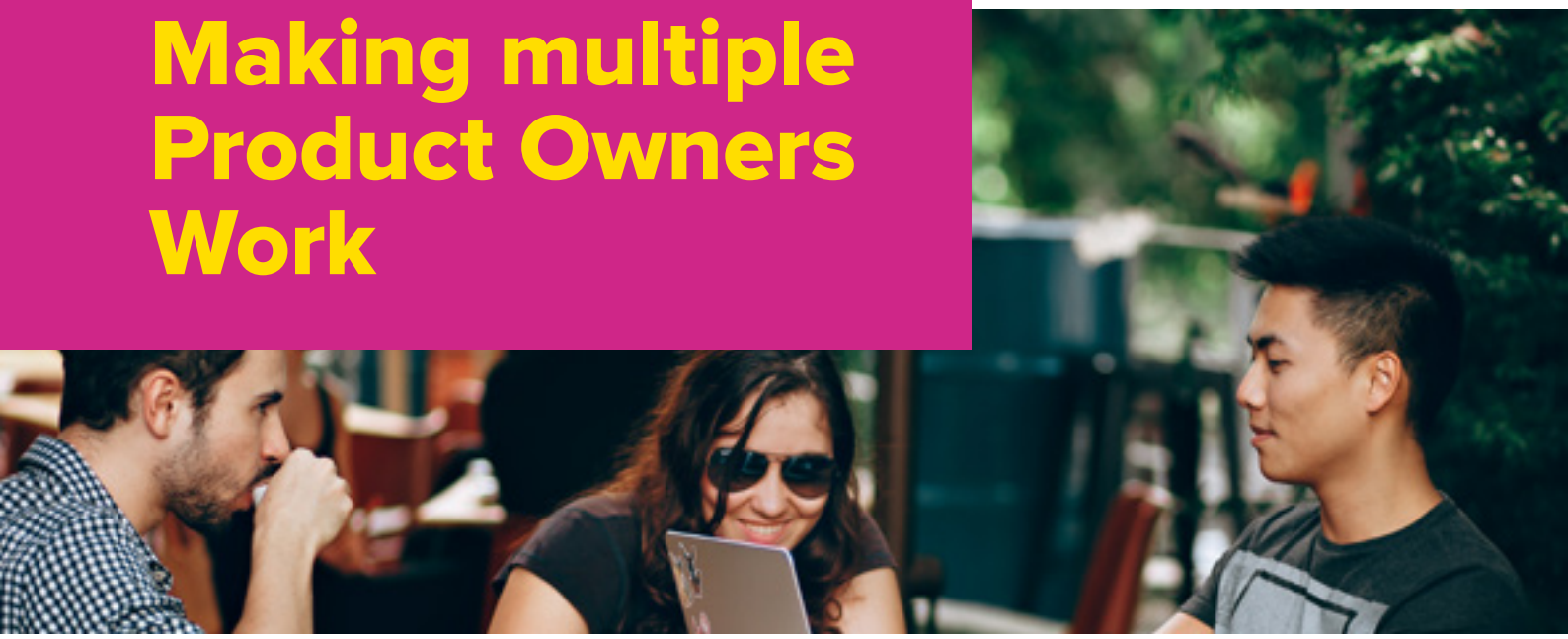
James: Infrastructure — servers and core software that the applications sit on top of

“Having split Product Owners works because we have split the products clearly,” James says. “It’s quite clear to the team who’s responsible for which area, so there is never any question about who to talk to.”

Constrained budget

If you have this kind of clear split, one option is to have one Scrum Team for each stream. The downside is that each team needs a Scrum Master and ideally at least three developers. This can cost more money. If you can afford it, consider this approach. You will need to invest a bit of time in coordinating the different teams but this should be less work than the constant coordination required when you combine Product Owners within a single team.

Making multiple Product Owners Work



Here's how the National Library have made multiple Product Owners work.

Set up a Super Product Owner

Initially the National Library had one "super Product Owner", what Mike Cohn calls a Chief Product Owner.

"He had oversight over all the different streams of work and would ensure that everything had appropriate level of business value relative to cost, and that we were getting the right blend of work from the different streams," says James.

» *The Chief Product Owner on Large Agile Projects* — Mike Cohn

"He'd look at the Backlog for the next sprint before it came out, discuss it with the various Product Owners, perhaps tweak the priority order, or question whether a story should be done now, or done at all. It was a way of managing that natural tension between different people wanting to get their own way."

Then organisational responsibilities changed and this was no longer possible.

Use budgets to share development effort

In order to continue giving each stream a fair share of the development effort, each Product Owner now has their own budget and they decide on the priorities for spending that budget.

Using budgets in this way is a bit of a blunt tool so the Product Owners have some flexibility to juggle the work.

Coordinate, communicate, collaborate

“Making multiple Product Owners work needs an extra level of coordination, communication and collaboration.

The National Library Product Owners discuss and coordinate their work internally as well as with the team.

“The Product Owners support each other as a team and put effort into working well together,” Scrum Master Rebecca says. “Because they work very closely they understand each other’s priorities. They’ll talk about it as a group first and decide what the overarching business priorities are before they push their own work.”

“You have to be a bit adaptable,” James says, “you have to understand that you’re not the only king in the room.”

You do spend some time twiddling your thumbs in meetings while the other Product Owners discuss their work with the team.

“That’s a pretty minor downside,” he says, “and the advantage is that the Product Owners are aware of what other work is going on.”

James also says it’s best if you don’t weigh in too often on work for the other Product Owners.

“Learning to bite your tongue is reasonably important,” he says.

Like Boost, the National Library team are mainly in Wellington, but some of the Product Owners are also based in other cities.

“The extra communication’s made fairly easy by the online tools that we use, like Slack and appear.in.” James says.

» *Slack*

» *appear.in*

Spend time with the team

A lot of this collaboration happens naturally as a result of all the Product Owners coming along to all the Scrum Events.

Along with Sprint Planning and refinement, Retrospectives and Reviews, they all also join the Daily Scrum. For those in Wellington, this often means appearing in person. For those outside Welly, it’s via video conference.

Both Scrum Teams come together for a combined Review. This gives everyone insight into the all the work and offers additional perspectives for feedback. It's also a chance for the whole crew to celebrate the impact the teams have had, the benefits delivered to the National Library and the people of New Zealand.

Knowing that face-to-face communication works best, the Product Owners are also looking to spend more time in person here at Boost.

Be available

Because the Product Owners also work in the business they're not full time. That means they have to make a special effort to be available, to check stories that are up for acceptance and to respond to questions. It's a bit of a juggling act.

Adapt the Backlog

For the National Library work we use Pivotal Tracker as our digital tool for managing the Backlog, along with a physical Scrum board. We have a single Backlog for all the work of each of the Scrum Teams, meaning each Backlog contains stories for multiple Product Owners.

Each story specifies which Product Owner was the Requester so the developers always know who to talk to about the story.

» *Pivotal Tracker*

Deliver value

Spreading stories amongst three streams might make it harder to deliver a potentially shippable product.

For this work, most stories deliver value straight away. That's because the Definition of Done means that each story is fully tested and integrated into the existing project, and because few of the stories are dependent on other work to be released.

When one stream has a chunk of work that can only deliver value if a number of large stories are completed in a Sprint, the Product Owners can juggle the amount of work for each stream to enable this.

Other benefits of multiple Product Owners

The Product Owners get additional benefits beyond being able to spend time in the business:

Specialisation — They get to become more of an expert in their stream than they could if their attention was spread across streams.

Shared load — They don't need to come up with a Sprint's worth of stories on their own.

Wider view of the business — Working on day-to-day basis with their fellow Product Owners gives them a broader view of the business context than they'd have if they were in a separate team.

Some cover during absences — Because the team keep up with each other's work they can sometimes check stories or answer questions, though their specialisations limit this.

Other impacts of multiple Product Owners

With extra Product Owners, everyone has to work extra hard to keep meetings within their timeboxes. Having multiple Product Owners can also affect different team members in different ways.

Impacts on the Development Team

While Product Owners can specialise, a cross-functional development team needs to be able to work on all the streams.

"Having three different streams getting fed through one team means there's quite a lot of context-switching for people, which can be a challenge for efficiency. Some devs prefer to get in the headspace of just one type of work and stick to that for at least a Sprint," James says. "Other devs have said the opposite, that they actually enjoy the variety."

"Within the team, they have the opportunity to manage themselves. If one person feels like they'll benefit from working exclusively on this stream of work, then as long as their teammates are happy with that, then we're happy with that."

Impacts on the Scrum Master

Rebecca has found very few issues as Scrum Master.

“It’s just a couple more relationships to build, a bit more admin, just making sure that everybody can attend the meetings,” she says. “It’s also harder sometimes when Product Owners are remote, making them feel engaged.”

“It feels like we’re one team,” she says. “That’s because I trust that they trust each other.”

Thinking of having multiple Product Owners?

You should probably think again.

Often organisations are tempted to have multiple Product Owners for reasons that don’t fit with the Scrum framework, such as giving stakeholders a say in the work.

“Ask yourself, ‘Do we absolutely need multiple Product Owners?’. I would say most of the time you don’t,” Rebecca says.

But if you can’t work without having multiple Product Owners, we hope this case study helps you do it in a way that gives the impact you’re looking for.

So, what makes a kick-ass Product Owner?

The Product Owner role is central to Scrum.

It requires a wide range of skills. You need to be good at collaborating but willing to make the big calls. You must divide your time between understanding the business and working with your Scrum Team. You need to be good at communicating — able to listen closely and explain yourself clearly. You have to be open to change but must stick to the principles and values of Agile through thick and thin.

All this can be challenging. But, once you get into the swing of things, and really starting kicking ass, it can also be enormously rewarding.

We hope this Guide helps bring you these rewards.

If you've any questions, feedback, or just want to share your Scrum experiences, please drop us a line:

info@boost.co.nz

+64 4 939 0062

twitter.com/boostnz

facebook.com/BoostNewZealand

Level 5, 57-59 Courtenay Place, Wellington