

celerity



The Role of the Product Owner:

From Mystery to Mastery

The Role of the Product Owner:
From Mystery to Mastery
By Todd Miller

The Product Owner's key responsibilities, at the highest level, are **driving team productivity, building trust with the stakeholders, and serving the users.**

I. Introduction

Many Agile projects suffer from poor or absent Product Ownership, often because the role is complex, challenging, and fraught with unclear expectations and responsibilities.

It's a recipe for confusion over how to best perform the role to generate successful outcomes for the team and product.

Because the role of the Product Owner is largely a mystery, we developed this guide to help Agile teams and organizations better understand responsibilities and drive toward more effective leadership.

In this whitepaper, you'll find foundational advice for masterful Product Ownership based on the experiences of our seasoned Agile experts. We explore how to navigate among the three key constituent groups in an Agile project so that their needs are balanced against the demands of development, and against one another. As mastery becomes a reality, you'll be able to produce a product your team enjoys developing and customers love using.

Table of Contents

I. Introduction	2
II. The Mystery	3
III. The Mastery	4
IV. The Opportunity	9
V. Do's and Don'ts	10
VI. Tools and Terms	11
VII. Contributors	12



A masterful Product Owner works to optimize the effectiveness of the three key constituent groups in a balancing act that requires thoughtful attention to each group.

II. The Mystery

In our many years of leveraging Agile development for ourselves and our clients, we've learned that one of the most difficult aspects of the Product Owner role is unraveling the mystery around how much time to spend with different people during product development. We've simplified this problem by identifying three constituent groups that demand attention during an Agile project: **the team, the stakeholders, and the users (or customers).**

A successful Product Owner must master the art of optimizing each group's effectiveness; strong, balanced relationships are critical to project success. When a Product Owner concentrates on one constituent group at the expense of the others, the result is often undesirable: a product runs over budget, takes longer than expected to produce, or never makes it to completion.

Spending the right amount of time with each group is essential. In the pages ahead, we describe the key activities a Product Owner must **engage in with the team, stakeholders, and users to produce a great product that is received well and created with enthusiasm.**



III. The Mastery: Balancing Constituent Needs

The Team

The Agile team is essential to product development; their iterative schedule and ability to solve problems in real time have revolutionized delivery expectations across the digital space. A productive team relies on a Product Owner to determine priorities in the Product Backlog so that they can self-organize around problems and work productively.

To encourage this momentum, efficient Product Owners focus their time on the Product Backlog and support the team in the Sprint Backlog as needed. It's not unusual for a Product Owner to sway to the extreme of absenteeism or, conversely, slip into a state of micromanagement.

That's why a Product Owner will best **accelerate the team's productivity** when they master these four concepts:

1. Focus on the Product Backlog
2. Empower the team
3. Help - don't hinder - self-organization
4. Facilitate cross-communication

1. Focus on the Product Backlog

The Product Owner must coordinate processes from the Product Backlog and let the Development team govern activities in the Sprint Backlog. Understanding how to leverage the Product Backlog is key to maintaining a healthy Product Owner-team relationship.

The team is a hungry wolf pack that the Product Owner feeds by **living and breathing in the Product Backlog**. Time in this space can be used to develop compelling user stories, guide product direction during Backlog Refinement sessions, and contribute to software craftsmanship by not accepting low quality or unfinished work.

2. Empower the Team

Because the Product Owner lives mostly in the Product Backlog, they must **master the art of empowering the team so that they can function self-sufficiently**. This means trusting the team to determine the fate of their Sprint Backlog, listening to the team when they have input, and being available when requirements need clarification.

The team usually has a better grasp of the time and effort tasks require and interference from the Product Owner can derail the notion of empowerment. Once the team commits to what they think they can accomplish in a Sprint, the Product Owner should avoid questioning the team's hourly estimates and instead focus on pushing for a high-quality solution and sustainable pace. Preventing the accrual of technical debt is critical for momentum, but a Product Owner shouldn't have to micro-manage them to get there.

3. Help - Don't Hinder - Self-Organization

It's important for a Product Owner to help the team self-organize because self-organization sustains the heartbeat of a happy team. A Product Owner must **avoid disrupting the daily Scrum** to task-check or get status updates because that causes a negative reaction on self-organization. Over-maintenance of the daily Scrum, due to its frequency, also takes time away from a Product Owner's responsibilities to other constituent groups, putting project success in jeopardy.

What is self-organization? Consider a rugby team and its coach. A rugby coach shouldn't run onto the pitch in the middle of a Scrum to shout at the team about technique. That would be distracting, ruin momentum, and interrupt the team's potential to solve the problem on their own. Similarly, a Product Owner shouldn't interfere as the team moves the software 'ball' down the 'pitch' in an Agile Sprint.

4. Facilitate Cross-Communication

A successful Agile Product Owner **facilitates the team's ability to talk directly to stakeholders and users**. Providing the team with access to the right connections and identifying preferred channels of communication generates stakeholder-customer-team feedback loops. An alternative option is to practice "Just in Time" reviews as Sprint work becomes complete.

A Product Owner must be aware that teams can get overwhelmed by this direct communication, despite its benefits. They'll need to work with the ScrumMaster to monitor any unproductive distractions.



You've reached the end of this section.
Complete the check point before proceeding.

Check Point:

Are you committed to driving team productivity?

- I understand my role in the Product Backlog
- I trust the team to determine the course of their Sprint
- I can clarify requirements without micromanaging
- I let the Development team be in charge of the Daily Scrum
- I find ways to bring the team and other constituent groups together



The Stakeholders

Stakeholders in an Agile project are people who have an interest in a project's result and can impact development on multiple fronts. This may include the executives that pay for it, salespeople who work with the end-user or customer, or backend office personnel whose jobs may be affected by the outcome.

A Product Owner should **develop trusting relationships with the stakeholders**, but this is easier said than done because these relationships must be maintained delicately. Product Owners may find themselves dodging stakeholders to avoid disruption or, at the other end of the spectrum, gauging all activity with metrics to appease this constituent group.

Four key strategies a Product Owner can employ to stay balanced and masterfully build trust with stakeholders are:

1. Include stakeholders in Sprint Reviews
2. Identify risk and work transparently
3. Make educated commitments
4. Determine best metrics to communicate progress

1. Include Stakeholders in Sprint Reviews

Stakeholders often want a voice in the direction of a product. **A forward-thinking Product Owner prepares to take their feedback and have the Product Backlog reflect the outcome of those conversations.** One consistent way to receive feedback is to invite stakeholders to Sprint Reviews so that the Product Owner can get more frequent incremental feedback.

It's impossible for a Product Owner to make every decision without additional insight and stakeholders often have a good understanding of customer value. Of course, not all feedback is good feedback, so a Product Owner should be careful to always discern helpful from unhelpful advice.

2. Identify Risk and Work Transparently

Although stakeholders will be involved in Sprint Reviews, it's essential to bring them in at other times if circumstances unexpectedly evolve. Stumbling blocks during development are typically unavoidable, so it's critical for a Product Owner to **identify risks and be open about impediments** before they become complicated conversations.

Agile projects naturally require that risk is reduced as quickly as possible. By socializing risk and working creatively with the constituent groups to resolve it, a Product Owner will diminish the need for tough conversations with stakeholders down the road.

Product Owners may find that a Risks and Impediments Backlog makes this kind of transparency easier to monitor. Once implemented, the backlog can also be used to track inter-team dependencies when scaling.

3. Make Educated Commitments

Nurturing a habit of frequent and open communication with stakeholders is an excellent step toward a trusting relationship, but knowing how to follow up with those conversations can make all the difference. When making commitments to stakeholders, a Product Owner must **first consult the team.**



Making uneducated commitments will put the Product Owner and team in a difficult position. As important as Product Owners are to the Agile process, they may not know the effort involved in a given requirement, so this concept acts as a kind of ‘reputation insurance.’

4. Determine the Best Metrics to Communicate Progress

Metrics can give an Agile Product Owner the edge they need to appease stakeholders, but it’s important to find the right metrics to communicate progress as Agile tools are frequently implemented incorrectly.

For example, Velocity is a commonly misused and miscommunicated metric and has been the cause of an untold number of miserable teams and stakeholders. Velocity is not an evaluation metric for punishing or rewarding a team. It’s a predictive measurement for forecasting the future. Misusing it with the stakeholders increases workplace vulnerability and creates a hostile work environment. This will cause gaming of estimations, which decreases predictability of future efforts.

We’ve successfully used Release Burn Up or Burn Down Charts to mitigate risk and visualize progress to stakeholders. Working demos, however, are the best visual evidence of software that a Product Owner can give to stakeholders. Impact mapping and user story mapping tools are also great visual ways to be transparent with stakeholders and build a Product Backlog. A Sprint Burn Down, while a great instrument for cross-functional teams to identify Sprint goal realization, shouldn’t be used as an administrative reporting tool or to display day-to-day progress to the stakeholders.



You’ve reached the end of this section.
Complete the check point before proceeding.

Check Point:

Are you prepared to build trust with the stakeholders?

- I can distinguish between helpful feedback and misguided advice
- I’m comfortable openly discussing risk with stakeholders as I recognize it
- I consult with my team before making promises to the stakeholders
- I understand how to use metrics to my advantage



The Users

It's easy to get caught up in the stakeholder's goals or the team's momentum, but a Product Owner can't afford to lose focus on the users, or customers, who will interact with the completed product. This is especially important if the product's intended purpose is to build revenue. In any case, a Product Owner requires user feedback to validate assumptions made during development.

To **best serve the users**, a masterful Product Owner needs to:

1. Leverage the expertise of User Experience (UX) professionals
2. Maximize value with Lean tactics
3. Embrace Minimum Viable Products (MVPs) and Minimum Marketable Products (MMPs)
4. Illicit feedback

1. Leverage the Expertise of UX Professionals

A Product Owner that has access to UX professionals should **take time to understand what they do and try to digest the results of their studies**. These team members can put you in the minds of end-users, which is invaluable when designing exceptional customer experiences.

As forward thinkers, they'll usually work 1-2 Sprints ahead of the rest of the team. A Product Owner should ensure that they don't move too far ahead because that can lead to stale requirements akin to WaterScrum, undermining Agile's favored emergent requirements.

2. Maximize Value with Lean Tactics

Users want great products, and they want them now. Mastering Lean tactics is essential to uncovering fast, valuable delivery to users. They allow Product Owners to **find and eliminate waste at all costs**.

The key is to maximize value in all aspects of your operation, whether that be through customer-centric prioritization of the Product Backlog or by identifying operational inefficiencies. Product Owners may view this as operating in a constant sense of danger, and that's a good thing. Possible questions to ask to stay on track:

- What's slowing down the team?
- Can I quantify user value on this feature?
- Is what we're building high quality?
- Are we accruing technical debt?

A Product Owner that serves users relentlessly drives toward delivering early and often. Act-Learn-Build is one philosophy that's popular in the Lean community because it encourages validated learning, which clears up assumptions about a product that can impact future user adoption.

3. Embrace MVPs and MMPs

A proactive Product Owner is comfortable with MVP (the smallest thing to build to begin validated learning) and MMP (the least a user will accept to adopt the product). They'll often need to drive goals so that the team reaches an MVP or MMP before moving to other product features. This will help to **separate needs over wants**, which benefits all constituent groups.

A successful Product Owner creates an environment where the team loves building the product, the customers love using the product, and stakeholders feel they influenced the product.

4. Illicit Feedback from End-Users

The most important way for a Product Owner to serve users is to **talk to them so that their feedback is always recognized**. A Product Owner might include them in Sprint Reviews, create customer focus groups, and creatively engage them across development. This can be done within the boundaries of a product by adding feedback mechanisms, measurements on feature usefulness, or telemetry tools.

The Product Owner is the team's liaison to the customer and should spend sufficient time with users to direct the team toward delivering a great product.



You've reached the end of this section.
Complete the check point before proceeding.

IV. The Opportunity

The role of the Agile Product Owner doesn't have to be a mystery. While they wear many hats during product development, the role confusion Product Owners experience stems from a misunderstanding of how to balance time across the three distinct constituent groups.

As Product Owners become more comfortable with their key responsibilities, they'll learn to master harmonious relationships across the groups in an ongoing process that evolves with the life of the product.

It's important to acknowledge that time spent with the three groups, while balanced, will almost never be equal because every project is unique; the team, stakeholders, and users will require varying degrees of attention throughout the product lifecycle.

An astute Product Owner recognizes this and sees it not as a limitation but as an opportunity to strategically drive team productivity, build trust with stakeholders, and ultimately serve the users. The result will be a product that people love to build and, most importantly, love to use.

Check Point:

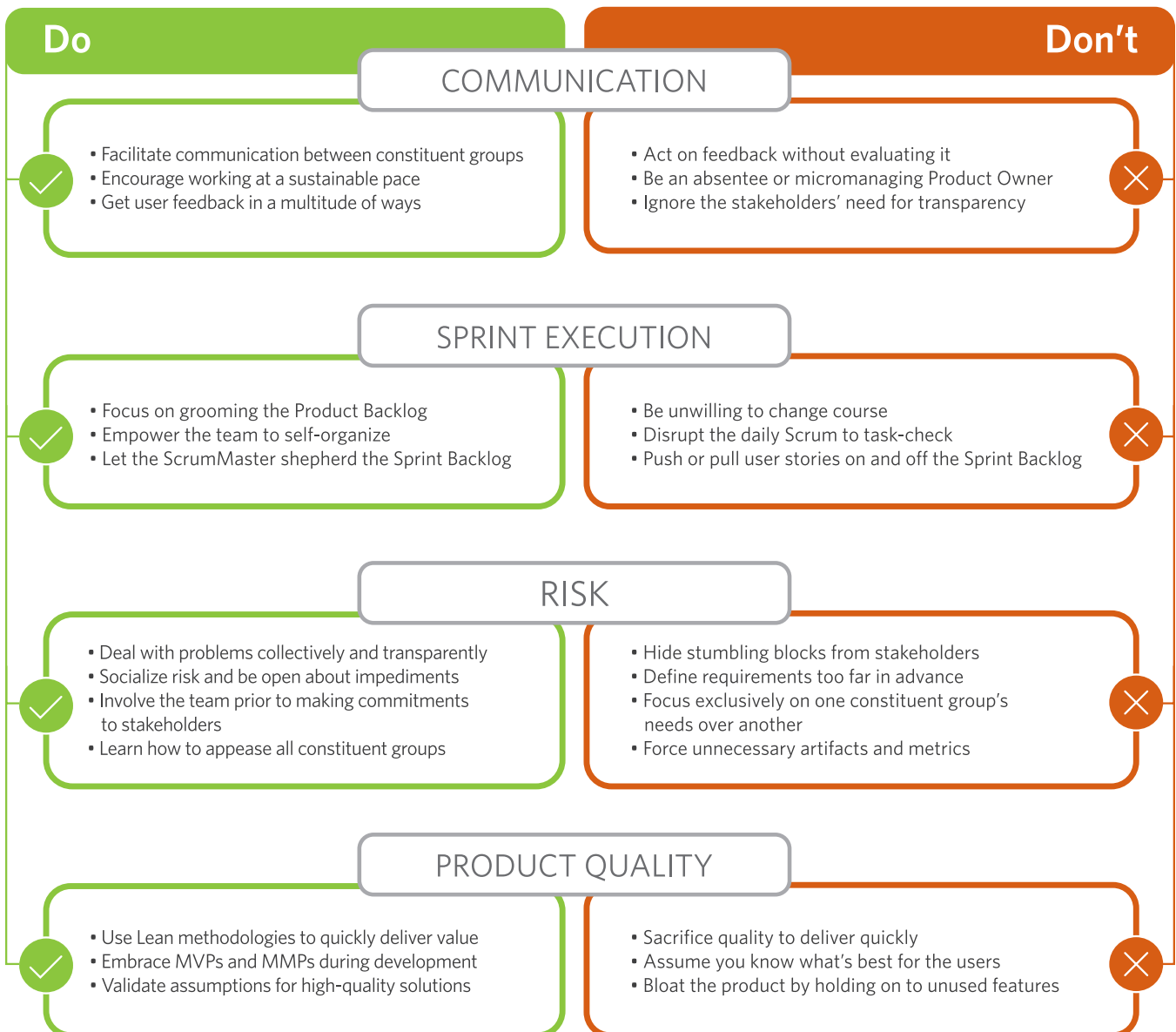
Are you ready to serve the users?

- I have access to UX experts that strategize with users in mind
- I use Lean methodologies to minimize wasted time and resources
- I see the value in MVPs and MMPs
- I use feedback mechanisms to gauge user experience



V. Do's and Don'ts

Here are some critical Do's and Don'ts to help you navigate the Product Owner role successfully:



VI. Tools and Terms

Through our experience working with Agile development methodologies, we've found that a keen understanding of Agile tools and strategies is beneficial to all constituent groups.

As the Product Owner, remember to demonstrate situational awareness to know which strategies work best for each group and be flexible when choosing metrics because priorities can change quickly and it will be necessary to respond accordingly.

We've listed 10 definitions of key terms that can enhance how you operate with the three constituent groups:

Product Vision: The first step in an Agile software development project. The product vision can be thought of as the product's "Elevator Pitch," describing the purpose of the product and the desired outcome. A Vision Board visually concentrates/materializes the vision to reinforce project goals. Roman Pichler offers a free template of this tool.

Non-Functional Requirements: Constraints or system attributes on a given product. Non-functional requirements should be identified at the beginning of a project and kept in mind throughout development to ensure project success.

Sprint Goal Definitions: Overarching goals of a Sprint as defined by the Scrum team during Sprint planning. Sprint goal definitions help the team focus on the greater goals of their work during a Sprint.

Chapters: High-level goals developed by the Product Owner, generally feature-based, that the team hopes to accomplish over the course of 2-3 Sprints. Chapters can be redefined at any time to accommodate the direction of development.

Prioritization: The act of ranking the work items on the Product Backlog, with the most valuable being first on the list. The Product Owner is ultimately responsible for the priority of work and the order in which completion should occur.

Product Backlog: The artifact where all requirements on a Scrum project exist, including features, bugs, technical work, and knowledge acquisition. The Product Owner holds authority over this artifact and can add, remove, and re-prioritize its content.

Sprint Backlog: The Sprint Backlog is a subset of the Product Backlog, reflecting what a team has committed to completing in a given Sprint. The team holds authority over this backlog, and the ScrumMaster helps the team remove impediments and self-organize around its contents.

Impact Mapping: A business' greater objectives for the project demonstrated by outlining assumptions of the product. Impact mapping keeps the team focused on solving business problems during development and helps stakeholders measure a product's business impact after release.

Story Mapping: A grid or tree organizing user stories. Story mapping allows teams to visualize the functionality users require in software and release iterations accordingly.

WaterScrum/ScrumBut: Semi-Agile frameworks that result from teams that are only partially committed to Scrum. WaterScrum, or ScrumBut, usually involves some retention of Waterfall processes.

Credits:

Todd Miller - Author & Agile Guru

Alyson Mullee - Copy Editor

Brad Blyzwick - Designer



The Role of the Product Owner:
From Mystery to Mastery

