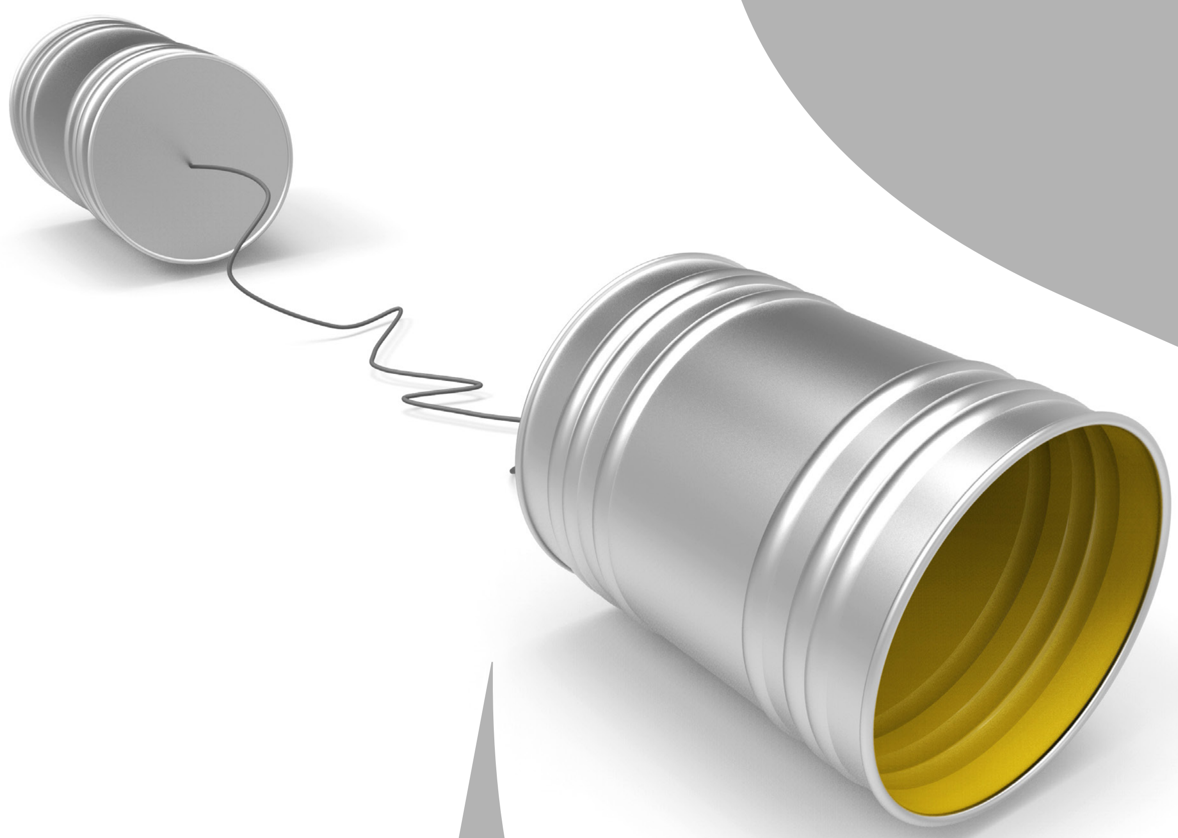
Mike Giannakopoulos
Nikos Vasileiou

Remote Scrum Guide



July 2022, version 1.0

Team O`clock

.TOC

When people say “Agile development” they are usually talking about releasing value in small, frequent increments. But to get there, a team must gradually build the habits, and establish the proper culture that will create a cadence of continuous value delivery through inspection and adaptation.

There are tons of Agile frameworks out there. Some are built for small teams, some for big enterprises with thousands of employees. However, one of the most well established frameworks is undoubtedly Scrum. Scrum has gained tremendous momentum during the last few decades, as one of the easiest frameworks to begin with.

This book is not the Scrum Guide and is not starting from scratch. We are making the assumption that the reader has some experience and understanding of the Scrum framework. In this book, we’ll remind the core Scrum concepts and focus on how Scrum can be applied for remote teams.

Everything shared here is tool-agnostic. We won’t promote ours or any other services. Furthermore, all the information shared is play-tested either by us or other teams we collaborate with.

Let’s get started.

REMOTE TEAMS AND SCRUM

What is **a Scrum team?**

A Scrum team is a group of people using the Scrum framework to perform their work. In the next sections, we'll describe in detail the Scrum methodologies. Let's see in brief the roles of people in a Scrum team:

- **The Development Team** handles the implementation of the actual work to be delivered within the sprint.
- **The Product Owner** is responsible for splitting and prioritizing the work in small increments. This way the Development Team can release fast, what values more.
- **The Scrum Master** is ensuring that the Scrum team has all the tools and space needed to perform their work. Validates that Scrum principles are followed and safeguarded. Usually, the Scrum Master is a member of the Scrum team.

As a Scrum team, it is important to define the stakeholders you are producing value for! Scrum started as a method for developers and products, so stakeholders were the actual users using the product. Nowadays, the Scrum methodology is used by more teams, so stakeholders can vary and even be people from within the same organization.

What is **a Remote team?**

A remote team's core characteristics are:

Team members are located in different places, e.g., working from home, different office spaces, or even different countries.

Team members with different working hours, different time zones, or different schedules.

The remote team still collaborates toward common goals. Team members need to coordinate and work together to achieve these goals. But, being remote has a few challenges of its own. And the team must enhance their ways with habits that improve their work:

- **Members work independently.** Each member has an autonomous task to work on.
- **Asynchronous communication.** When a member is sending a message to the team, the default is to wait for some time for a response. The response will not be instant – synchronous. And that should be ok.
- **Good planning is required.** The team will need frequent sync-ups to align and collaborate on the delivery of work.

A remote team is relying more on tools that assist in working from a distance. Tools that help with async communication, collaboration, and sharing of information.

Having the right tools in place helps a remote team build habits and excel in their craft.

Challenges

Working in a remote team imposes a new set of challenges to overcome:

- Establishing a dedicated workspace at home is one of the biggest challenges when working remotely. The ideal setup is a separate room or desk that defines a “work setting”.
- Colleague availability defaults to “not here right now”. It is best to avoid assumptions on availability, and plan ahead for any connection and questions with colleagues. Set a meeting time, or share your input with a loose timeframe for an answer – instead of “right now”.
- Updates and syncing with the team default to interactions with no physical presence. This means that the team either follows the “not here right now” rule or sets meetings to get in sync.
- Reporting on progress requires more effort, as people work alone. The team needs to find the balance and flexibility to control reporting. Tipping the scale towards too many meetings, or too few interactions, results in a lack of focus.
- The proper work/life balance is harder to manage when working from home. With no clear distinction between office and home, people tend to work more or lose focus. In both cases, a personal schedule and discipline can help.
- The switch from an in-office to a remote setting makes people feel lonely. Lack of physical interaction and water-cooler discussions need additional effort to overcome.

Opportunities

The remote setup offers some benefits to teams working that way:

- People are happier when working remotely. The reasons vary from less commuting time to schedule flexibility.
- Less distraction and more focus, since employees control their time and distractions. Focus time can help in delivering better work and being more productive.
- Enabling diverse collaboration is not bound to the office room or cube space. Working online, you collaborate with people from other departments on new ideas, enabling true agility.
- Asynchronous work is an enabler for many initiatives that can work in parallel. In an async workflow, people can collaborate over documents, comments, and code. Meetings and calls are for a few hard decisions and brainstorming. This way, many projects can progress in parallel.
- Remote work is location-agnostic, meaning that people can work from anywhere. This flexibility enables team diversity in hiring. Also, the location flexibility allows employees to work from anywhere. So, a business-vacation blended scheme is possible.

SCRUM ARTIFACTS ON REMOTE SETTING

What is **a Scrum artifact?**

Scrum defines some artifacts to represent work done: The Product Backlog, the Sprint Backlog, and the Increments.

The Product Backlog is where Product Owners brainstorm and organize the product evolution. Product Backlog Items (PBIs) are sorted from the most important to the least important. This is a never-ending process, as Product Owners discover more opportunities ahead. Prioritization helps a company focus on delivering value vs. creating features with zero or little traction. This way, customer satisfaction and excitement go up; leading to further monetary growth.

The Sprint Backlog is more technical. It is a combination of a Sprint Goal and a set of Product Backlog Items selected for the sprint in place. The Sprint Goal defines a clear target to achieve within the timeframe of the sprint. A Product Backlog Item in a sprint may be broken down into tasks or subtasks. Each task describes the small steps required toward fulfilling the Sprint Goal. The outcome of the Sprint Backlog is a Product Increment.

In short, the Product Backlog serves as the “Why” and the “What” of the next thing to build. The Sprint Backlog describes the “How”.

Building good habits over maintaining a proper Backlog is key for the tribal nature of the Scrum team. It is the heart of communication. It is the campfire, where members of the tribe get together to share progress. So, keeping the Backlog information clear and in one single place, ensures that the whole team is on the same page at all times. Imagine the communication breakdown if, for example, information was scattered all over the place: in emails, chat threads, paper, or even verbal communication.

Managing **a remote backlog**

A team's Backlog within the office can be as simple as sticky notes on a whiteboard. A few vertical lines, with labels to describe the state (e.g., To-do, Doing, Done), and the team can add notes, move them around, or even discard what looks irrelevant or outdated. In general, a very fun and engaging way to stay on track and deliver a successful Sprint goal.

The purpose of the Backlog is to act as the one and only reference point for the entire team. The place where team members can see progress and refocus if needed. However, in a remote setting, the team needs to emulate the Backlog in a digital way, by making use of a digital board.

For a remote team, a digital online board is not the only way to get the job done. There are tons of online services and tools that do backlog management online, such as Jira, Trello, Asana, PivotalTracker, etc., to name a few.

The digital board system can notify the Scrum team of changes via emails or other notifications. Designs, documents, or links, make the digital backlog item rich enough for triaging and implementation by the team.

Keeping it all **async**

Having a detailed and up-to-date backlog is key for working within a remote team over separate time zones. The switch to async work may involve colleagues starting work much later or much earlier in the day. In this setting, ensuring a proper hand-off of your part of the job is critical. More often than ever, a remote colleague might depend on your deliverable to get on with their work.

Backlog management software supports comments for asking and answering questions and reporting progress. It's crucial for a remote team to know the state of each item in the backlog, flag items that need attention, or view items with impediments.

Keeping the information IN the backlog makes it easier for your team to ensure that no data is lost. Avoid the temptation to share progress in a private chat message, a video call, or something that will make the information leak from the whole team.

Describe **PBIs as user stories**

It is strongly encouraged to use the user story format to describe Product Backlog Items. User stories are a great tool for Product Owners, fostering value-driven thinking. In a nutshell, the format of a user story is:

As a ...(who)... I would like to ...(what)..., so that ...(why)...

The user story format holds important information that helps the team make decisions, design a feature, and take action. It is crucial to have all the elements of a user story in place. A brief overview of each user story element is:

- **"As a...(who)"** part defines the person that will benefit when this story is delivered. That person could be a persona or a specific role using your service.
- **"I would like to...(what)"** part describes the action that the person will do. Defining the action in a way that does not relate to your service, helps the team explore more solutions, e.g., instead of "I would like to click the submit button..." write "I would like to enter my data".
- **"So that ... (why)"** part defines the goal, the reason, behind the person's action. This part of the user story is critical, as the team could

Best practices

identify alternative ways into achieving the same goal. For example, in order for a person to be always aware of changes in a software service, the solution could vary from mobile notifications, emails, or even an in-app banner.

If a user story is big enough, the Scrum team can break it down into smaller user stories, to ensure proper delivery within the sprint timeframe.

User stories are then broken down into technical tasks by the Development Team.

For example, let's say that we have the following user story:

As a user, I would like to manage my notification settings so that I can get less distracted by pings throughout the day

This can further break down into a set of technical or non-technical tasks for the Sprint Backlog:

- Add notification setting fields in the database
- Create a notification settings page in the user interface
- Add server API endpoints, to update the notification settings
- Update user documentation with instructions

Notify our users about the update

Best practices

Keep **the backlog lean**

As the product matures, more feedback is gathered, and more customer requests are coming. This is where the backlog starts to grow. A big backlog is bad for the business. It may stall forward-thinking and further product discovery.

Usually, an item in the backlog creates the obligation for action and closure. However, this is not always the case. If you feel courageous enough, a radical practice is to remove all the old items and keep the backlog lean. After all, if an item consistently comes fort by your users, it probably sounds like a good priority for the next sprint.

In short, keep the skeletons away. Create room for forward-thinking and more product discovery.

Best practices

DAILY SCRUM FOR REMOTE TEAMS

What is **a Daily Scrum?**

Daily Scrum, as described in the official Scrum Guide, is a short meeting for team members to update on the progress of their work towards their common goal. Daily Scrum started as a Development Team ceremony, but it can become more inclusive, or be used by teams outside Scrum as a habit.

The Daily Scrum is a 15-minute meeting, where each team member shares their updates. In its most common format, participants are covering up to the following three questions:

- What work has been done since the last meeting.
- What they are currently working on.
- What are the impediments blocking their progress, if any.

It is suggested that only members working towards the team's goal take part in the Daily Scrum, regardless of role.

Teams that properly execute the Daily Scrum in their ceremonies, tend to improve faster. Misalignments and blockers can surface early in the day, and the whole team can take prompt action on resolving those issues and get work done. In time, the team can learn to perform the ceremony efficiently, eliminating the need for additional clarification meetings.

How remote work **differs on a daily basis**

Working with a remote team dictates a different way of collaborating with teammates:

- As individuals plan their working day, there is a big chance of solo work and limited colleague interactions, due to little time overlaps.
- People can focus more on a task, due to fewer distractions. That's because the default for remote work is to assume that colleagues are not instantly available.
- Considering the solo work and minimal overlap, remote work requires better planning and alignment. This way, people know what to do next, once the task they currently work on is complete.

All the points above converge on the need for better alignment and clarity on a team level. This way all members know where they are heading, which is the priority, and who they are expecting answers from to move on with their work.

Performing **a Daily Scrum**

Based on team location and way of work, there are different ways to perform the Daily Scrum. Regardless of how the Daily Scrum is performed, its value and existence are crucial. Some teams perform the Daily Scrum every weekday, while other teams perform it two or three times a week.

A synchronous Daily Scrum call, for teams working in the same office or at the same time zone. The team is usually having a scheduled 15-minute Daily Scrum call, taking rounds and sharing updates. This meeting is usually performed at the start of the working day to boost alignment and focus on upcoming work.

An asynchronous Daily Scrum session, for teams distributed in various time zones or different work schedules. Members share updates over an online tool (e.g., a document, a chat application, or a service). The team can read all notes and respond via chat or comments for extra clarifications or help. The asynchronous Daily Scrum has a longer duration, giving time for people in different time zones to add their updates. The duration can vary from 4 to 12 hours, depending on team distribution.

A hybrid Daily Scrum, having both asynchronous and synchronous parts. The hybrid option is for teams working in different time zones with some time overlap. Like the asynchronous session, members add their notes in an online tool at their pace. Moreover, the team has a scheduled call to align on the shared notes, share comments, and provide clarifications.

Challenges

The Daily Scrum is a habit-forming ritual. As teams get comfortable performing the Daily Scrum, it is easy to fall back on behaviours that diminish the meeting's value. With diminishing returns, the team feels demotivated on performing the Daily Scrum. So at any point, the team should try to apply corrective action to get the most benefits possible from this ritual. Some challenges are:

- The reporting trap for the Daily Scrum meeting. The presence of a team leader in the Daily Scrum can make members feel like they are reporting to that person. Reporting trap makes people share details on their work, skipping any issues requiring help from their colleagues.
- Team members focus on sharing their part without engaging with their colleagues' updates. This is more problematic on an asynchronous Daily Scrum, where members skip reading notes. Indicators of this behaviour are little progress on work, some tasks staying behind schedule, and extra calls to cover the missing parts.
- The solution trap for the Daily Scrum meeting. This is happening when an impediment suddenly triggers a solution discussion within the scope of the Daily Scrum ritual. The solution trap leads to disengagement by non-involved members and extends the meeting duration. This problem is affecting teams that are performing a synchronous Daily Scrum in a call.

Opportunities

When done right, the Daily Scrum has an additive positive effect on the teams, due to its recurring nature. Two of the basic benefits offered are:

- Fast alignment and corrections in case the team changes direction. Since the Daily Scrum is periodic, misunderstandings or changes of direction are shared on the spot, keeping everyone on the same page.
- Focus and commitment are at the forefront, keeping the team aligned on the work that needs their attention. So, the team focuses on delivering the most valuable work in order of priority.

RETROSPECTIVE FOR REMOTE TEAMS

What is **a Retrospective?**

According to the Scrum Guide, a Sprint Retrospective is an inspection of "how the last Sprint went regarding individuals, interactions, processes, tools, and their Definition of Done". The purpose of the meeting is to plan ways that increase the quality and effectiveness of a team.

Extending the above definition beyond the sprint case, more teams can reap the benefits. So, a retrospective is an inspection of the past period in regard to individuals, interactions, processes, tools, and the goals that a team has set.

Running inspection meetings, like the retrospective, is vital for all teams that want to evolve. A successful retrospective promotes open communication and improves teamwork among peers. With better communication and collaboration, a team is happier and produces positive outcomes.

Performing **a retrospective**

Before starting a retrospective, the team needs to define:

- The scope of the retrospective in terms of both period and topic of discussion. With a clear scope, the team can focus on the important stuff, and leave non-relevant remarks outside the discussion.
- The retrospective activity theme, where the team agrees on the form of the retrospective. The purpose of the activity is to help team members to structure and share their thoughts. There is a multitude of activities to choose from by browsing the Internet, or you can create your own.

Performing a retrospective meeting consists of some steps:

- **Adding notes**, where participants add their notes on interactions, processes, tools, and goals of the said period.
- **Grouping notes**, where participants share notes and clarify any understandable notes. Additionally, some notes are grouped together since they are common.
- **Voting notes**, where each participant casts votes on the most significant notes already shared.
- **Discussion and action items**, where the team discusses the most voted notes and takes some action items that are helping the team improve.

The most typical way to perform a retrospective with a team is **in sync**. When all team members share the same physical room, write notes on post-its and stick them on a board. If team members are remote, the retrospective is done over a call, and notes are added to a digital board. A synchronous retrospective lasts from one up to three hours, depending on the topic in discussion.

A team can perform a retrospective **fully asynchronously**, too. In that case, all instructions and retrospective notes live in the digital space. A member of the team is the facilitator. The facilitator is responsible to organize the retrospective by sending emails to the team to:

- **Define the scope of the meeting.**
- **Set the retrospective activity.**
- **Share the digital space where the retrospective will take place.**
- **Time-box the duration for each step.**

All team members join the digital space to share their notes and vote. It is best to use a chat channel to ask for clarifications on notes, and to perform an asynchronous discussion on action items. A fully asynchronous retrospective can take more than a day or two and could span a full week.

A third option is **hybrid**, with some parts performed asynchronously and others in sync. A big part of the retrospective is for individual contribution, where members can take part at their pace. The team can start adding their notes, get clarifications, and vote asynchronously. For any clarifications, the team can use a chat channel; if there is ambiguity on many notes, the team can jump on a call to clarify together. With these steps done, the team can get to a meeting to discuss the emerging themes and take action items. Having a hybrid retrospective can take up to two days. Set a day and a half for the asynchronous note-taking and voting, and have a one-hour call to discuss and take action items.

Challenges

Having a retrospective has a variety of challenges. One thing to keep in mind is securing the psychological safety of all participants. Avoid any personal blaming and finger-pointing, focus instead on ways to improve as a team. Your team goal is the collective effort, achieved through collaboration and support.

Some more things to keep in mind depending on the way you conduct a retrospective are:

- When meeting in person, keep the discussion focused on the important topics. In a retrospective, topics with varied importance for each person pop up. Deep diving into each topic early on can lead you to an analysis paralysis discussion. To avoid that, stick to clarifications only before discussing the most voted items.
- An asynchronous meeting, spanning many days or hours, makes people disengage from sharing notes and following up. Additionally, not seeing colleagues makes the meeting impersonal. To improve this, have reminders or time-boxed sessions on sharing notes.
- With an asynchronous part and then meeting for a discussion, teams might rush to resolutions. To avoid missing important notes, try walking through all the asynchronous notes as a team.

Opportunities

The retrospective meeting is an opportunity for team bonding, growth, and alignment. The evolution of Agile and Scrum also affected the retrospective meeting. Retrospective activities are part of that evolution. Using storytelling or analogies, an activity helps the team exchange notes and ideas.

Through a variety of retrospective activities, teams can focus on improving specific skillsets. Here are a few categories to help your team goals:

- **Standard activities** focus on the team's strengths and weaknesses. Standard activities work great for a set period or completed work; e.g., a quarter, or a feature completion.
- **Playful activities** use analogies, fables, or fantasy storylines. Playful activities are celebratory, unifying, and inspiring the team over the completed work.
- **Futurespective activities** to analyze upcoming challenges. Futurespectives are good to identify areas for improvement, potential flaws, and upcoming work.

Since the retrospective format is very flexible, you can create a custom activity on your team's needs. Custom activities are more relatable to your team's status.

Performing **a retrospective with a remote team**

A successful retrospective for a remote team leverages all the characteristics of the remote setting. You will need a tool that allows participants to:

- Contribute independently.
- Participate asynchronously.
- Have access to all information shared by their peers.
- Group for a synchronous discussion when needed.

With all the above in place, you set the conditions to turn the retrospective ceremony into a team habit. Here are the steps that will help you get there.

For starters, you will need to set up the retrospective for your remote team. Setting up a retrospective is important to think about the period you will be inspecting with the team; is it for the last weeks, or the period it took you to get to a milestone? Going past that, you choose from a set of retrospective activities the one that suits your needs. There are activities for future thinking (futurespectives), reflection, celebration, etc. All activities are promoting open communication and team bonding so that each team member can freely share findings.

When you are ready with the setup, you can invite your team members into adding their notes. You can do that via email or through your company's chat application. It is important to educate the team on the period of inspection, and the specifics of the retrospective activity.

At this point, each team member can enter the retrospective to add their notes. Since the team works remotely, you can inform them to add notes within a specific period, spanning from a few hours to some days. There is no need for everyone to be on a call or a meeting to add their notes, each one can do it on their time. When the specified time has passed, you can share the last ping – via email or a message – and then inform your peers that the retrospective is moving on to the next step.

With all notes added, the retrospective proceeds to the next step, sharing notes and clarifications. At this point, all the notes are visible to the whole team. Participants can identify notes that are similar and need grouping, or obscure and require clarifications. You can ping the person who added the note for clarifications, via email or a direct message.

There are two options for handling this step as a team:

- Continue asynchronous, and set a facilitator to group and clarify notes on behalf of the team.
- Have a synchronous meeting, to group and clarify notes together.

When the grouping step is complete, you can ping your colleagues to cast their votes on the items of importance. Each member can cast their votes on their time, asynchronously. Doing that async and not in a meeting is excluding any bias due to talking while casting votes. You can set a specific time limit for the voting step.

With all team members done voting, you can proceed to the final step of the retrospective, the discussion. This is the only step of the retrospective that the remote team needs to get together and talk. You can send a meeting invite to the team and share the voting results. Sharing the voting results beforehand provides space for review.

When you all meet, make sure to follow the etiquette of online meetings:

- Provide space for everybody to share their thoughts on the topic of discussion.
- Try not to interrupt the speaker.
- Keep the discussion within the topic.
- Keep the discussion within the specified time.

The retrospective format can lead you to long discussions. The topics are always relevant to the team and everyone feels strongly about each item.

Please remember to:

- Be polite.
- Keep the discussion around team improvements, and avoid going personal.
- Focus on getting up to 3 action items, no more.

When you finally conclude with the discussion and action items, the retrospective is considered complete. You will need to inform all participants about the resulting action items and plan how you can start addressing each one. To achieve that you can use helpers as tasks in a ticketing system (e.g., Jira, Asana, Trello), reminders in your company messaging app, or email reminders.

SPRINT PLANNING FOR **REMOTE TEAMS**

What is **a sprint planning?**

The Spring planning is the ceremony that kicks-off the next upcoming sprint. It is the event where the whole Scrum Team debates over the work that will be the focus point of the next few weeks, depending on the sprint duration and the sprint goal.

Sprint planning happens at the beginning of the sprint. However alot of teams are preparing the backlog and remove uncertainty during a ceremony called the Backlog Refinement Meeting. During this section we will focus on the key elements of a remote backlog refinement process. These tips, however, can be applied during the Spring planning as well.

What is **a backlog refinement session?**

Backlog refinement is an agile ceremony, usually taking place at the mid of the sprint. The Product Owner or Scrum Master facilitates the meeting. The purpose is to prepare the backlog for the upcoming sprint planning. Backlog refinement ensures that important tasks stay on top of the list. With team discussion, there is alignment over the feasibility and effort of each task.

During a backlog refinement meeting, team members discuss the user stories or tasks over the following scopes:

- Keep most important items to the top, as candidates for the upcoming sprint planning.
- Remove unrelated or obsolete items.
- Create new user stories if needed.
- Break down items into smaller ones, to keep the scope small enough to match the sprint's length.
- Align the whole team over the effort required to get things done.

What a backlog refinement session offers is a shared understanding of upcoming work for the entire team. For example, a Product Owner can see the technical limitations or technical refactoring required to deliver a new feature. Or, for an engineer to understand functional requirements or needs for infrastructure upgrades.

Sharing different perspectives – from product, development, QA, and DevOps – helps to create a common understanding of a work task. With more clarity, the Product Owner can understand the effort required for each task. With all this data in mind, the Product Owner can take an informed decision over the value vs. effort, and prioritize or deprioritize a user story for work.

Performing **a backlog refinement session**

Usually, a backlog refinement session is done over a synchronous meeting. The whole team gathers, and the Product Owner brings items for discussion and clarification on the expected outcome.

When a task is clear, the team can size the effort on delivering that task from the backlog. The sizing process has the following outcomes:

- It fosters discussions to better understand the requirements of the goal.
- It surfaces different perspectives on the implementation difficulty. For example, a member might say that a task is “easy” to build but forget to include QA or updates to the documentation.
- Hard to complete tasks, break down into smaller independent tasks. Small independent tasks ensure continuous delivery and incremental releases. A continuous delivery flow is reducing the development risk, shows progress over time, and makes the team happy with the work done.

The most common sizing technique is the planning poker activity. During the activity, the Development Team is estimating the effort required over a metric, for example, “small”, “medium”, “large”, “extra-large”, or using a Fibonacci scaling, such as 0.5, 1, 3, 5, 8, 13, 100. When all members align over the sizing effort for a task, this task is refined and the team moves on to the next task. In cases of big diversity in effort sizing, for example, if someone says “small” and another says “large”, a discussion starts to clarify further.

In a remote setting, the activity is asynchronous, to span many timezones. In that case, the estimation can happen over a tool, like Slack or Microsoft Teams. In such cases, team members estimate a set of user stories at their pace. Then, a discussion can take place asynchronously as well, if required.

A combination of sync and async refinement process is also possible. Starting off by providing a set of tasks to estimate at their time. When everyone has voted, a synchronous meeting takes place to discuss uncertain items and come to a common plan.

Challenges

The backlog refinement session has a few key challenges for both the Product Owner and the Development Team:

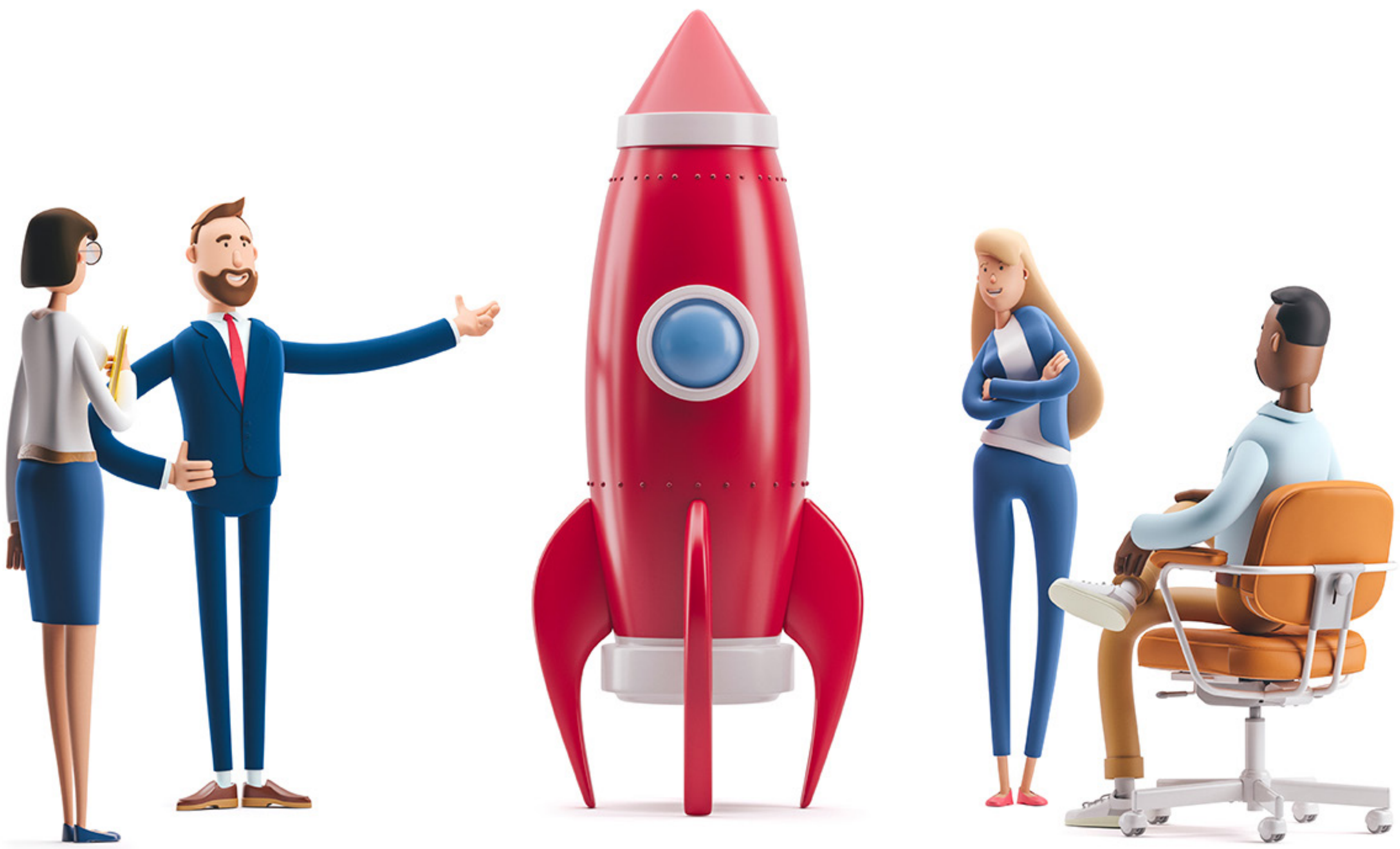
- Decide on priorities based on the customer value vs. the effort required to get it done.
- Be able to size the complexity and the effort based on what is known. Are there rabbit holes ahead? Is there unknown technical debt lurking around the corner, putting the endeavour at risk?
- Ensuring that user stories or tasks are clear enough so that everybody on the team has a common understanding of what is at stake during the discussion.
- Remove bias over the sizing of the work. Everyone should be able to equally contribute from their perspectives, without senior people exclusively setting the tone, and driving the discussion.

Opportunities

In a remote setting, challenges have the potential of opportunities. Given the async nature of a remote team, written clarity over the work to be done is becoming more important than ever:

- Using digital backlog management software, the Product Owner can describe user stories. With clarity, there are no loose ends, so the Development Team understands better how to refine the work for that user story.
- The team can use digital planning poker tools, to size items, without creating bias.
- The Development Team can “buy” time to prepare beforehand, and make a better contribution to the refinement process.

SPRINT REVIEW FOR REMOTE TEAMS



What is **a sprint review?**

A sprint review is a meeting held at the end of a sprint. It is the opportunity for the Scrum team to demonstrate the outcome of their hard work during the sprint. In this meeting, both the Scrum team and other stakeholders participate and share their feedback on the work done. This review is helping shape upcoming work and eventually plan ahead for the next sprint.

In a remote setting, sprint reviews happen online. Having a remote sprint review has some major opportunities for the team. Including stakeholders outside the company is much easier in a remote setting, as these people will probably be away from the office.

A remote sprint review also promotes the agile approach of working with customers on specific projects, versus trending to the more casual waterfall approach. Finally, it is way easier to include clients by incrementally sharing progress either via an online meeting or offline, as we can see below.

Synchronous **remote sprint review**

When both the Scrum team and the stakeholders are in similar time zones, the event can happen via an online meeting. Zoom, Google Meet, Microsoft Teams Calls, and other tools, are there to help an online meeting.

The Scrum team can share their screen and demonstrate the progress through a presentation or a live demo. Get feedback, keep notes, or debate towards the best next version of the feature in development.

Asynchronous **sprint review**

When team members or stakeholders are in different time zones, this is where an asynchronous sprint review can kick in.

The Scrum team can record a video of the delivered increment, covering any potential open questions and the decisions making process. Fortunately, there are tons of technologies to easily use this method. Tools like Loom can make it super easy to record and share an online presentation.

An alternative approach is to share a document on the delivered increment, offering details, designs, and attached screenshots. However, sharing a prerecorded video presentation is preferable.

With the sprint review material at hand, stakeholders can get back with comments and feedback online, at their pace. This can be done either through email or within a chat application, such as Slack or Microsoft Teams using threaded discussions.

You've read the whole guide! Thank you!

We would love to hear your experience and feedback on the contents of this book by dropping us an email at hello@teamoclock.com.

We will try to keep the content updated to reflect the reality of the world we live.

Team O`clock