

38 questions to help you **avoid**

Agile Imposters

2.0

Revised and expanded

Now featuring the answers
that you'll want to listen for!



Find the people
you need to lead your
agile development

by **Stefan Wolpers**



HANDS-ON AGILE™ fieldnotes™

a new **Age of Product** series

38 questions to help you **avoid**

Agile Imposters

2.0

Revised and expanded
Now featuring the answers
that you'll want to listen for!

by **Stefan Wolpers**

Contents

Introduction	02
About us	04
The role of the Scrum Master	06
Backlog refinement and estimation	19
Sprint planning	26
Standups	35
Retrospectives	43
Conclusions	50

Introduction

Maybe “agile” in general is a fad as opposed to a trend. But whatever the case, we can say for sure that scrum is now very popular in software development. Demand for seasoned scrum practitioners and the entry of new professionals into the market are both on the rise.

If you are looking to fill a position for a Scrum Master or Agile Coach in your organization, you will find the following 38 interview questions useful in identifying the right candidate. These questions are derived from the author’s ten years of practical experience with XP and scrum, serving both as product owner and Scrum Master, and throughout this time interviewing dozens of candidates.

These questions are not enough to turn an inexperienced interviewer into an agile expert. But in the hands of a seasoned practitioner, they will help to determine whether the interviewee has actually been working in the agile trenches — or whether they’re an **Agile Imposter**.

Regarding Agile Imposters 2.0

Following public discussion on [LinkedIn](#), we — that is, Andreea Tomoiaga and I — decided it would be helpful to provide examples of, and guidance interpreting, the answers that we believe would indicate a suitable candidate for a role as Scrum

Master or Agile Coach. This version 2.0 of the Agile Imposters fieldnotes is the outcome of that.

Being cognisant of what to listen for in a candidate's answers allows an interviewer to more quickly understand not only a candidate's familiarity with scrum, but also his or her agile mindset.

With respect to this version 2.0 of the text, please keep in mind that:

- The examples and guidance reflect the personal experience of the authors, and may not be valid for every organization. What works for another organization may not work for yours.
- Multiple choice questions are insufficient when you need to identify a candidate's agile mindset, given the complexity of applying agile to an — or any — organization.
- Andreea and I share a holistic view on agile methodologies: **agile equals product discovery** (what to build) **plus product delivery** (how to build it).

The author's original blog post introducing the first version of the Agile Imposters fieldnotes is [available online](#) via Age of Product.

About us



@stefanw



/in/stefanwolpers



stefan@age-of-product.com

Stefan Wolpers

Author

Stefan has worked many years as a product manager and Agile Coach (Scrum, Lean Startup, Lean Change). He's founded multiple companies and has led the development of B2B and B2C software for multiple startups — including a former Google subsidiary. Starting into software circa 1990, by 1996 he had launched one of the first ecommerce products to feature SAP R/3 connectivity. Other ventures followed, and in 2011 he founded [Startup Camp Berlin](#) — now one of the largest and most active startup conferences in Germany. His latest project, [Age of Product](#), is about lessons learned and best practices for how to identify what product to build and how to build it efficiently. Age of Product has been hosting events and workshops in Berlin since 2015, and plans for a new industry conference may be in the works! Read more about Stefan at [LinkedIn](#) and [Scrum Alliance](#).



Andreea Tomoiaga

Contributor

Andreea has been involved in delivering software products both large and small since 2001, frequently working with distributed teams. She is passionate about agile, and always interested in learning something new. Read about Andreea, her work in agile, and her articles on the subject at [Scrum Alliance](#), [LinkedIn](#), and her own [web site](#).

 [/in/andreeatomoiaaga](https://www.linkedin.com/in/andreeatomoiaaga)

The role of the Scrum Master

Background

- Scrum is not a methodology, but a framework. There are no rules that apply to each and every scenario, just best practices that have worked in other organizations before.
- You need to determine on your own what works for your organization. Doing so is a process, not a destination.
- Being a Scrum Master or Agile Coach is primarily about leadership and coaching. It is not a management role.
- Being a Scrum Master or Agile Coach is definitely not about enforcing processes.
- Scrum is not designed for bean counters. Some metrics are helpful to understand the health of a scrum team but, generally, insisting on “meeting KPIs” (e.g. commitments vs. velocity) does not help.
- Scrum does not have much to say about the process that enables the product owner to fill the backlog with valuable, usable, and feasible user stories. Product discovery based on Lean UX, Design Thinking, or Lean Startup may

help this process. A good Scrum Master wants her team to be a part of this process — for example, by participating in user interviews or running experiments.

- The team’s communication with stakeholders should not be run through a gatekeeper (e.g. solely through the product owner). Doing so hurts transparency and negatively effects the team’s performance. Sprint demos are a good way to not only present the value delivered by the team during the previous sprint, but also to stay in close contact with stakeholders.

Question 01

The Agile Manifesto says “people over processes”. Isn’t the Scrum Master, a role meant to enforce the process, therefore a contradiction?

Answers to Question 01

A Scrum Master does not wield any real authority; the scrum team does not report to them. This question, therefore, is meant to discern whether the candidate understands that their role is to lead — as opposed to manage — the team. This question should also help to discern why a candidate is interested in the role of a Scrum Master in the first place. Acceptable answers include:

- I am the facilitator for the scrum team. It's my job to make them successful.
- I am neither a project manager, nor a people manager. I remove impediments, but I do not tell people what to do.
- I am the scrum team's coach, encouraging them to excel as an agile team.

Question 02

What would you consider good indicators that agile is working for your organization, and that your efforts are succeeding?

Answers to Question 02

There is no general definition of "agile success" that can be used to measure an organization's agility. Every organization must develop its own criteria. However, although mostly indirect, there are various indicators that may be useful in discerning the rate of success. These include:

- An increasing team velocity (prior to each new plateau).
- Improved team happiness: less fluctuation; more referrals from team members to fill openings.

- Increased competitiveness in the war for (tech) talent: more senior people are willing to join the organization.
- Products delivered to customers feature better value proposition — resulting, for example, in higher retention rates, better conversion rates, increased life-time value, et cetera.
- An increase in software quality (less technical debt, less bugs, and generally better maintainability).
- Reduced lead time from validated idea to shipped product.
- Reduced cycle time for hypothesis validation.
- Less allocation of resources on low value products.
- A better standing of the IT team among stakeholders.
- Increased participation of stakeholders in agile meetings, for example, during the sprint demo.

Question 03

Are there any common metrics that you would track? If so, which metrics would you track and for what purpose?

Answers to Question 03

This question is, in part, a subset of the previous question. However, and especially when concerning the change management process itself, there are effects that could be measured with a scoring model. Some of these effects are:

- The ability to respond to change and produce valuable code. In other words: the ability to break down features.
- The duality of planning at both release and sprint level.
- The flexibility to adapt to changing facts, time boxes, and continuous delivery.
- That teams are bidding on stories, and have the freedom to select their approach to solving them.
- The creation of a culture of shared learning, and the continuous demonstration of features delivered.

In order to capture these developments, a scoring model can be devised based

on the agile maturity in the organization so that these qualitative aspects can be quantified and thus compared.

Usually — before introducing an agile framework within an organization — it is useful to gather the status quo (the baseline) of these dimensions and review their evolution through time.

These metrics should be continuously addressed throughout the agile journey (via surveys with team members in agile departments, for example).

Question 04

Your team is constantly failing to meet commitments, and its performance velocity is volatile. What might the possible reasons be? How would you address this issue?

Answers to Question 04

This question addresses various issues. There are many possible factors that might combine to make a team's velocity volatile. Such factors might include:

- New team members are being onboarded
- Team members are leaving
- Levels of seniority

- Working in uncharted territory
- Working with legacy code, probably undocumented
- Unexpected technical debt
- Holidays and sick leave
- Executive intervention

If the team's commitment is frequently too aggressive, this could indicate that user stories are poorly prepared — e.g. the definition of ready is not met — which makes the stories hard to estimate. Conversely, their projects might suffer from poorly documented legacy code, excessive technical debt, or just too much buggy and poorly written code — all of which make estimation very hard.

Organizations must be vigilant and not fall for the fallacy that agile is working because commitment and velocity are aligned. Cooking the agile books is easy to do!

The obsession with commitment matching velocity

Read more at [Age of Product](#)



Question 05

Should the scrum team become involved in the product discovery process, and if so, how?

Answers to Question 05

The scrum team should be involved in the product discovery process as early as possible. There are two principal reasons for this:

- The sooner the engineers participate in the product discovery process, the lesser the chances are that solutions are pursued that are technically not viable, or that would not result in a return on investment.
- Early involvement ensures that the scrum team and the product owner develop a shared understanding and ownership of what is to be built. This significantly improves the likelihood that resources are allocated to the most important issues, maximizing value for the customer and mitigating investment risk.

Involving engineers early in the process also ensures their buy-in, encouraging a higher level of commitment and engendering a willingness to participate in all of the phases of a project's development. Ultimately, such involvement leads to motivation, and the scrum team should thus be motivated to participate when changes or a new direction are needed to achieve the goals defined for a sprint or product release.

Question 06

The product owner role is a bottleneck by design. How can you support the product owner so that they can maximize value?

Answers to Question 06

This is a rephrasing of the previous question. Again, the candidate should focus on explaining why involving the scrum team early on in the product discovery process is beneficial for both the product owner and the organization. The team wins together, or the team loses together.

Question 07

How do you ensure that the scrum team has access to a project's stakeholders?

Answers to Question 07

The candidate should understand that there is no way to ensure access to stakeholders. All that can be done is to encourage stakeholders to engage in meaningful communication by being transparent and helpful. Sprint demos are a

useful mechanism for this, often promoting better relationships between different departments and business units — improving a scrum team’s access to their projects’ stakeholders.

Question 08

How do you spread an agile mindset across different departments and throughout a company, and what is your strategy when coaching non-IT stakeholders?

Answers to Question 08

There are various ways of engaging stakeholders with scrum:

- Product and Engineering teams can produce evidence, in presentations or otherwise, proving to stakeholders that scrum is significantly reducing the lead time from idea to product launch.
- Product and Engineering teams can demonstrate that scrum mitigates risk, e.g. regarding the prediction of when new features could be made available, thus contributing to other departments’ successes in planning and execution.
- A scrum team can be transparent with respect to their work and proactively engage stakeholders by inviting them to meetings, sprint demos, and other

events where the team communicates their activity or progress.

- One hands-on approach is to organize workshops designed to teach agile techniques for non-technical colleagues.

App prototyping with absolute beginners

Read more at Age of Product



Question 09

How would you introduce scrum to senior executives?

Answers to Question 09

This is a deliberately open question meant to encourage discussion. The candidate should be concerned with how to spread an agile mindset throughout an organization, or, more particularly, how to create a learning organization that embraces experimentation in order to identify the best product for its customers.

Practical considerations might include organizing workshops for C-level management — applying scrum at the executive level has been successful in the past. Executives — and even, potentially, key directors — can gain first-hand experience

with agile methodologies if organized as a scrum team.

There are no answers to this question easily labeled “right” or “wrong”. Best practices will — and should be — tied to an organization’s culture, size, and the industry it’s operating in.

Question 10

You’ve already provided your project’s stakeholders with training in scrum. After the initial phase of trying to apply the concepts, when the very first obstacles are encountered, some of these stakeholders begin to resist continued adoption. What is your strategy for and experience in handling these situations?

Answers to Question 10

This is another open question, meant to encourage an exchange of ideas about and lessons learned when overcoming resistance to agile within an organization. Familiarity with agile failure patterns common in many organizations will demonstrate a candidate's experience — we have published a [list of agile failure patterns](#) at Age of Product.

Agile failure patterns in organizations

Read more at [Age of Product](#)



Backlog refinement and estimation

2

Background

- Estimation and backlog refinement are essential tasks for every team. Although the product owner — at least officially — is in charge of keeping the backlog at ‘peak value delivery’, he or she will need the assistance of the entire scrum team to do so.
- In the ideal scenario, this requires a cross-functional and colocated scrum team that can work independently from other scrum teams.
- The reality in most cases is, however, that the team might be dependent on deliveries from other teams (e.g. API endpoints) or deliverables from the UX/UI department.
- There are two essential ingredients for good team performance and velocity:
 - 01 **Writing the user story as a team:** The product owner explains why something should be built and provides the necessary background, e.g: market intelligence, results from experiments, user interviews, statistical data. They then create a corresponding user story (or stories) in a joint effort with their team to achieve a shared understanding of the ‘why’ and ‘how’ — and a shared sense of ownership among team members.

- 02 **Sharing a Definition of Ready:** The scrum team and the product owner also need to agree on a “Definition of Ready” for user stories to ensure a flow of well-drafted stories for the development process. It is an agreement on what needs to be provided to call a user story ready for estimation. If one of these requirements is not met, a user story isn’t ready for estimation. A user story without a previous estimation is not ready to become a part of a sprint backlog as the team won’t be able to commit to an unknown entity in the sprint. The scrum team must learn to say “No”.
- A well-groomed product backlog probably has user stories detailed for about 2 or 3 sprints, less than half of which probably meet the “Definition of Ready” criteria of the team. There may be some additional user stories that no one except the product owner is working on at the moment.

Question 11

The product owner for your team normally turns stakeholder requirements documents into tickets, and asks you to estimate each. Are you okay with that procedure?

Answers to Question 11

A Scrum Master should not accept such a procedure, as it’s nothing more than a waterfall process dressed-up in a pseudo-agile methodology.

If the entire organization is supposed to focus on delivering value to its customers, it is essential that any process involving “requirements” being handed down to the engineers by a JIRA monkey be abandoned. Instead, the organization should start including everyone in the product discovery process — thereby establishing a shared understanding of what needs to be built.

Question 12

What kind of information would you require from the product owner to provide your team with an update on the product and market situation?

Answers to Question 12

Any information suitable to provide the team, with an understanding why something is of value to customers. This may be of a quantitative nature, e.g. analytical data describing how a process is utilized. It also may be of a qualitative nature, such as transcripts, screencasts, or videos from a user testing session.

Question 13

Who should be writing user stories?

Answers to Question 13

Writing user stories should be a joint effort made by the entire scrum team. If it's not, the team might not feel that they have ownership of the stories — inevitably leading to less or no buy-in, reduced motivation... and ultimately a lower-quality product.

Question 14

What does a good user story look like? What is its structure?

Answers to Question 14

A good user story:

- has a description,
- defines acceptance criteria,
- can be delivered within a single sprint,

- has all UI deliverables available,
- has all (probable) dependencies identified,
- defines performance criteria,
- defines tracking criteria, and
- is estimated by the team.

Question 15

What should a “Definition of Ready” consist of?

Answers to Question 15

A “Definition of Ready” is an agreement between the scrum team and the product owner about what must be included in a user story before the story can be considered ready for estimation. **It defines what a good user story looks like.**

We outlined what a good user story should include in our analysis of the possible answers to [Question 14](#). Another approach, however, is to use a framework for how we think about user stories. One such framework is [the INVEST mnemonic by Bill Wake](#), which specifies:

- ✓ **Independent:** The user story should be self-contained, in a way that there is no inherent dependency on another user story.

- ✓ **Negotiable:** User stories, until they become part of an iteration, can always be changed and rewritten.
- ✓ **Valuable:** A user story must deliver value to the end user.
- ✓ **Estimable:** You must always be able to estimate the size of a user story.
- ✓ **Small:** User stories should not be so big as to become impossible to plan, task, and prioritize with a certain level of certainty.
- ✓ **Testable:** The user story or its related description must provide the necessary information to make test development possible.

Question 16

Why aren't user stories simply estimated in man-hours?

Answers to Question 16

It's okay to use hours instead of points when the team and the organization can deal with it. However, some organizations will fall back into waterfall planning when work is estimated in hours.

Estimating in hours might also be tricky when:

- legacy software is involved,
- the team is facing significant technical debt, or
- a team is composed of mostly junior members.

Story points are more suitable in these situations because the points more accurately reflect both required effort and complexity. This is a result of points being used not only for estimation, but also as a means to measure and convey commitments and team velocity.

Question 17

The product owner of your scrum team tends to add ideas of all kinds to the backlog so to remember to work on them at a later stage. Over time, this has led to over 200 tickets in various stages. What's your take on this? Can a scrum team work on 200 tickets?

Answers to Question 17

Any product backlog larger than the scope of two or three sprints is not manageable. Misusing a backlog by adding hundreds of items to it is a clear sign that the product owner needs help from the team to cope with the influx of ideas, suggestions, and requirements — and avoid misallocating resources. This support is what the candidate should offer.

Sprint planning

Background

- Traditionally, the product owner would explain high-level user stories of high value from the product backlog to the scrum team. The team would then turn these into more detailed user stories, and estimate them.
- It turns out, though, that working on user stories in separate backlog refinement and estimation meetings — before the sprint planning — actually improves the quality of the stories and thus the outcome of the team’s work.
- A good practice is to run weekly product backlog refinement and estimation sessions, and only allow user stories that meet the “Definition of Ready” standard of the scrum team into the sprint planning.
- With all uncertainty eliminated, the sprint planning then creates a sense of ownership among the members of a scrum team for the sprint backlog items, as the team now can make a valid commitment.
- Sprint planning is consequently divided into two parts:

Sprint planning I: The product owner presents her choice of the most valuable user stories from the product backlog to the scrum team, and the team selects

— taking circumstances into consideration, for example available capacity — those stories it can commit to deliver by the end of the sprint.

Sprint planning II: The scrum team adds details to the user stories in the sprint backlog, e.g. splitting them into tasks, identifying parts that need further clarification, and agreeing about who will be working on which tasks. The product owner does not necessarily need to participate in the Sprint planning II, but does need to be on standby for questions.

- If the user story preparation process is handled well, the entire sprint planning may take no longer than 2 or 3 hours.
- A productive sprint planning requires a healthy team. Dysfunctional teams will not achieve the required level of cooperation. It will be a futile and painful exercise.
- A team should usually avoid allocating more than 75 percent of their capacity to new user stories. Bugs, refactoring, and research also require regular attention to avoid building-up technical debt.
- Insufficiently prepared user stories seriously hamper the flow and efficiency of the scrum team. These should never be selected for the sprint backlog, but instead sorted out during backlog refinement and estimation meetings.

Question 18

How can you as a Scrum Master contribute to sprint planning in such a way that the team is working only on the most valuable user stories?

Answers to Question 18

The scrum team should never select user stories according to any kind of ranking established by the product owner or anyone else. Sprint planning is determining the most efficient path to value creation. The best way to accomplish this is to ensure that:

- The scrum team is involved in the product discovery process at an early stage.
- The product backlog refinement process is well understood by both the scrum team and the product owner. (This can be supported, for example, by the creation of a “Definition of Ready” standard for user stories.)
- All user stories are created in a collaborative effort between the product owner and the scrum team. (A shared understanding and joint ownership is the goal.)

Question 19

With what metrics would you assess the value of a user story?
What metrics would not be acceptable?

Answers to Question 19

Answers to this question should focus on delivering value to customers and, by extension, the company or organization. Measuring potential technical debt, with the objective of minimizing it, is one example of a good metric.

Question 20

How do you facilitate user story selection in a way that the most valuable stories are chosen without overruling the team's prerogative to define their own commitments?

Answers to Question 20

If a team is involved early enough in the process, for example by jointly creating user stories with the product owner, or during product discovery, any kind of guidance will be unnecessary. Most teams will often autonomously pick the best user stories

for a given sprint.

If a team resorts to “cherry picking” user stories during sprint planning, however, then the backlog refinement process needs to be seriously addressed, as the product owner is likely picking user stories that are not maximizing customer value.

Question 21

How much capacity would you consider adequate for refactoring? Fixing important bugs? Exploring new technologies or ideas?

Answers to Question 21

Setting aside the most critical and urgent tasks (such as the web site being offline), the 15–10–5 approach is generally a good rule of thumb: dedicate 15% of a team’s capacity to technical debt, 10% to bugs, and 5% to explorative spikes. You may, of course, deviate from this when it comes to an individual sprint. But, generally, maintaining these allocations will satisfy the maintenance requirements of most software applications.

Question 22

How do you deal with a product owner that assigns user stories or tasks to individual team members?

Answers to Question 22

Assigning tasks to individual team members does not work at all and needs to be stopped. The assignment of user stories is the scrum team's prerogative. Preventing individual task assignment, if likely to occur, should be one of the Scrum Master's most pressing concerns.

Question 23

How do you deal with team members "cherry picking" tasks?

Answers to Question 23

A scrum team should generally have autonomy in how its members choose to distribute tasks. Maybe the presumed "cherry picking" of tasks by individual team members is a crucial part of the team's path to performance.

However, if team members complain about this happening, the scrum master needs to address the issue. Additional training might enable some team members to accommodate a greater variety of tasks. Or, perhaps, someone needs to be gently pushed out of their comfort zone so that they learn to choose different kinds of tasks than the kinds they currently choose.

Question 24

A user story is lacking the final designs, but the design department promises to deliver on day two of the upcoming sprint. The product owner of your scrum team is fine with that, and pushed to have the user story put in the sprint backlog. What's your take?

Answers to Question 24

The answer to this depends upon the team's situation and experience: If the design department will definitely deliver (because they have done so in the past); if the user story is high value and can be accomplished within the sprint despite its UI deliverables arriving late; and if the team agrees — it's probably an acceptable exception.

Beware, however, that this kind of exception does not become the “new normal” — allowing the organization to bypass the backlog refinement and sprint planning

process. The candidate should be aware that such a situation would not be tenable. Furthermore, if implementation of a user story subjected to such an exception fails, no one will bother to read the fine print and acknowledge that an exception had been made; instead, they will most likely view the agile process itself as having failed.

Candidates may either accept or reject exceptions to the agile process, but they should be able to analyze the situation and explain the collateral damage that the scrum team may be exposed to.

Question 25

A member of the scrum team does not want to participate in the sprint planning and considers the meetings a waste of time. How do you deal with this attitude?

Answers to Question 25

This kind of passive-aggressive behaviour is not necessarily a problem particular to scrum. It is, however, toxic, and will affect both team-building and performance.

If a member of a scrum team is behaving in this way, the Scrum Master will need to take action, as such behaviour can neither be ignored nor tolerated if the team is to continue functioning. Dealing with it may require an escalating approach:

- As a first step, the Scrum Master might address the team member privately to discuss their reservations. Perhaps they just need more coaching or a longer training period to change their attitude.
- As a second step, the entire team could make it an issue during one or more retrospectives — and offer support to the team member.
- If there is still no change of attitude, a meeting with the team member and his or her manager is advisable.
- Finally, if no change can be achieved, the team member should be reassigned — potentially to another, probably non-agile team. Alternatively, a kanban team that would not force the team member out of their comfort zone could be tried.

Scrum is not meant for everybody.

Standups

Background

- Standups are well suited to discuss the state of progress: is all going as planned, or does the team need to adjust?
- It is also a time when the scrum team can meet and communicate with the project's stakeholders.
- Standups cannot fix a dysfunctional organization or team, an inadequate backlog, a sprint planning gone wrong, low-quality user stories, a missing product vision... you get the idea.
- Standups are good if the team is already collaborating well and the basics are in order: backlog building, estimation and refinement, sprint planning, et cetera.
- However, the more experienced a team is, the better the internal communication, and the more the standup will seem a time consuming ritual of little value. A two person team does not necessarily need a formal standup; in this case, meeting for coffee might be a practical alternative.
- If impediments are not communicated before the next standup, something is wrong within the team. Maybe it is more a group than a team?

- Offline boards are great, as physically taking a card and moving it instills a certain ownership of a user story. If you have to let go of either an online or offline board, consider letting go of the online board if you're a colocated team.

Question 26

Would you recommend formal standups for all teams no matter their size or experience level?

Answers to Question 26

In answering this question, the candidate should exhibit common sense. Standups are an important part of scrum, but not all standups need to be formal. A small, experienced, colocated team may use a coffee break for their standup. However, taking that kind of relaxed approach to standups for a large team with several junior members would probably achieve nothing if it doesn't first descend into chaos. For large teams, a formal meeting is needed to provide format and guidance.

For distributed teams who need to "dial in" and, for obvious reasons, can't meet for coffee, a formal standup is necessary insofar as the standup, due to technical constraints, must be scheduled and conducted in an organized fashion.

Question 27

Do you expect experienced team members to wait until the next standup to ask for help overcoming an impediment?

Answers to Question 27

Certainly not. Waiting for the next standup delays progress.

When facing impediments, if team members are waiting for the next standup before asking for help, it's likely that the “team” is more of a group — and the scrum master has team-building work to do.

Question 28

How do you handle team members that “lead” standups, turning the event into a reporting session for themselves?

Answers to Question 28

All teams go through [Tuckman's stages of group development](#): norming, storming, forming, performing. [Scrum teams included.](#)

Although there are not officially any leadership roles within a scrum team, someone is likely to assume that role. This might happen because of his or her (technical) expertise, communication skills, or level of engagement. It's important, however, that this does not result in other team members reporting to this person. The Scrum Master must therefore be vigilant and intervene if necessary to ensure team members communicate — during standups or otherwise — as required by scrum.

Question 29

How do you manage team members that consider standups to be a waste of time and who are therefore either late, uncooperative, or who simply don't attend?

Answers to Question 29

The kind of passive-aggressive behaviour depicted by this question is not necessarily a problem particular to scrum. It is, however, toxic, and will affect both team-building and performance.

This question addresses the same problem as [Question 25](#), and an appropriate answer should address the same points.

If a member of a scrum team is exhibiting such an attitude, the scrum master will need to take action. Such behaviour can neither be ignored nor tolerated if the team

is to continue functioning. Dealing with it may require an escalating approach:

- As a first step, the Scrum Master might address the team member privately to discuss their reservations. Perhaps they just need more coaching or a longer training period to change their attitude.
- As a second step, the entire team could make it an issue during one or more retrospectives — and offer support to the team member.
- If there is still no change of attitude, a meeting with the team member and his or her manager is advisable.
- Finally, if no change can be achieved, the team member should be reassigned — potentially to another, probably non-agile team. Alternatively, a kanban team that would not force the team member out of their comfort zone could be tried.

Scrum is not meant for everybody.

Question 30

The standups of your scrum team are not attended by any stakeholder. How do you change that?

Answers to Question 30

This can quickly turn into a philosophical discussion. Should stakeholders be allowed to participate in daily scrums? If they do, is it a form of reporting — circumventing scrum rules?

If some adaptation of scrum is working in your organization — that's a good thing. Don't rule out stakeholders being allowed to participate in the daily standups if the team is okay with that. In fact, if stakeholders attend standups regularly, this will invariably and significantly improve communication between the team and the stakeholders — another good thing.

But how does a Scrum Master encourage stakeholders to attend standups? Make it worth their while. A Scrum Master can do this in any number of ways. They might offer stakeholders the opportunity to learn early details of a new product or feature if they attend. They might also choose to give stakeholders the opportunity to ask questions of the engineers directly, without going through the product owner.

Question 31

How do you approach standups with distributed teams?

Answers to Question 31

Standups with distributed teams are not much different to standups with colocated teams — except that sharing board activity may require video conferencing when the team is working with offline boards that mirror each other.

If you are using task management or planning software like JIRA (or any other cloud-based application), board updates are generally easy for each team member to follow on-screen. With such online software in place, a Skype call or Google Hangout may suffice.

Question 32

Can you draw an example of an offline kanban board for a scrum team right now?

Answers to Question 32

The qualifier “kanban” is, of course, a teaser. But any Scrum Master should be able to draw a simple board.

The necessary elements of any board are columns (or rows) for:

- Backlog

-
- In progress
 - Code review
 - Quality assurance
 - Done

Additional information may be included on or attached to any kind of board:

- Sprint or meeting dates
- User Acceptance Tests (UAT)
- A Definition of Ready
- A burndown chart (progress and work remaining over time)
- A parking lot (topics for future discussion)

Retrospectives

5

Background

- Retrospectives will only work to improve the team's collaboration and performance if they are a safe place to provide honest yet constructive feedback.
- Don't blame others. Everyone should be focused on how to improve the situation.
- Some teams include the product owner right away. Others insist that the product owner should be explicitly invited.
- Change locations regularly. It helps not to hold the retrospective at the team's workplace. Any amount of distance makes reflection easier, and being in a new place helps to prevent boredom and team members "checking out" completely.
- Change the format regularly. Running the same format more than twice is not recommended.
- Make it a smartphone-, tablet-, and laptop-free zone so that nobody is distracted and everybody can focus on contributing to the round.
- Document all issues, even if just temporarily using Post-it notes. However, it's

always better to keep a formal document or file.

- The typical set of questions includes: “What went right?”, “What went wrong?”, and “What is there to improve?”.
- An alternative set of questions is the starfish retrospective: “What to introduce?”, “What to keep doing?”, and “What to stop doing?” — eventually enhanced by “What to do more of?” and “What to do less of?”.
- Another alternative is the “Mad–Sad–Glad” technique. This technique works best following either:
 - longer intervals (e.g. at the end of the year),
 - major changes,
 - suffering a drawback or encountering pressure, and
 - outstanding achievements made by the team.

Usually this kind of retrospective encourages emotional self expression, thereby helping to uncover points of frustration so that strategies may be devised to overcome them.

- Action items and improvement tasks:
 - 01 Make action items falsifiable. “Do X more often” does not meet that criteria.
 - 02 Make a single team member responsible for an action item.

- 03 Don't forget to check on the action items identified during the last retrospective.
- 04 It is useful to put action items on a kanban board because this makes it easier to track progress.

Question 33

Who should participate in a retrospective?

Answers to Question 33

Although not mandatory, it's generally best practice to include the product owner in retrospectives — they are an important part of the “team”, both practically and functionally. Some scrum teams, however, may not always want their product owners to participate — a team's wishes should be considered.

Anyone else outside of the immediate scrum team, and particularly managers of team members, should not be invited to participate.

Question 34

Do you check the team's health during a retrospective, or is doing so unnecessary? If you do, how would you go about it?

Answers to Question 34

Measuring a team's health is a good way to get an idea about current levels of engagement and satisfaction, and is a useful method for discerning trends. A simple scale from 1 (terrible) through 2 (poor), 3 (neutral), 4 (good), to 5 (excellent) will do the job — and requires just two minutes to complete.

Question 35

What retrospective formats have you used in the past?

Answers to Question 35

With retrospectives, one size does not fit all. Consequently, there are various retrospective formats in common use:

- ✓ **The classic format:**
 - What did we do well?
 - What should we have done better?

- ✓ **The boat format:**
 - What's pushing us forward?
 - What's holding us back?

- ✓ **The starfish retrospective:**
 - Start doing ...
 - Do less of ...
 - Do more of ...
 - Stop doing ...
 - Continue doing ...

Each serves different situations. The candidate should have applied more than one of these formats in the past and be able to share their reasons for having done so.

Question 36

How do you prevent boredom during retrospectives?

Answers to Question 36

Any of numerous possible variations may be used to prevent boredom during retrospectives. A different location; a different format; shortening or lengthening the allotted time box. Scrum masters can also use the team's choice of action items to encourage and structure discussions around the issues that the team themselves thus identified — creating engagement through acknowledgement.

There is no single correct answer to this question. What's important is that the candidate acknowledges that boredom might become an issue, and that there are ways to deal with it.

Question 37

Your team is picking reasonable action items, but is later not delivering on them. How do you handle this?

Answers to Question 37

If action items are picked but no one is later delivering on their commitments, the Scrum Master needs to follow up.

If this is happening because of some impediment external to the team, the Scrum Master needs to address and eliminate the cause. The team can then catch up

during a later sprint.

However, if there is no external impediment, the problem is likely due to motivation, attitude, or personal issues within the team. In this latter case, the Scrum Master needs to provide those team members responsible with the encouragement or motivation necessary to overcome the problem, and see that they deliver on their commitments.

If the problem can not ultimately be resolved, picking action items becomes a useless exercise — and the team will suffer as a result.

Question 38

How would you recommend following-up on action items?

Answers to Question 38

A good follow-up practice is to start talking about the status of the action items of the last retrospective — before picking new ones. If actions items from the last retrospective haven't been accomplished, the scrum team needs to understand why they failed and prevent that failure from happening again.

Conclusions

My advice for the interview is to move as fast as possible from the meta- to the ground-level. Don't waste too much time discussing the advantages of agile methodologies or other theoretical concepts. Two or three questions from each of the sets in this fieldguide will provide more than enough ground for an engaging 60 minute conversation.

Scrum has always been a hands-on business, and to be successful in this a candidate needs to have a passion for getting their hands dirty. While the basic rules are trivial, getting a group of individuals with different backgrounds, levels of engagement, and personal agendas to form and perform as a team is a complex task — as you might say is always the case when humans and communication are involved.

The larger an organization is, and the more management levels there are, the more likely failure is lurking around the corner. In these cases you should usually choose the pragmatic veteran who has experienced failure in previous projects and carries the scars to prove it. Being a “Certified Scrum Master” — or having any such kind of certification — does not guarantee success.

Help fix Agile

by joining the



Agile Clinic™

our group on LinkedIn for
Agile practitioners



[linkedin.com/groups/8435015](https://www.linkedin.com/groups/8435015)



AGE of PRODUCT

INVENT FOR YOUR CUSTOMERS

www.age-of-product.com

Copyright ©2015 by Berlin Product People GmbH

“Age of Product” and the “Hands-on Agile Fieldnotes” series title are trademarks of Berlin Product People GmbH. All rights reserved. No part of this publication or its text may be reproduced by any means or otherwise distributed or made available without the prior written permission of Berlin Product People GmbH.

If you would like permission to reprint or publish online any part or all of this publication or its text, write to us at info@berlin-product-people.com addressed “Attention: Permissions Request”.

While the advice and information herein are believed to be true and accurate at the date of publication, neither the author nor the editor nor the publisher can accept legal responsibility for errors or omissions that may have been made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Editing, typesetting, and design by: Jourdan Ritchey
Published and distributed by: Berlin Product People GmbH

Berlin Product People GmbH

Borsigstrasse 8
10115 Berlin
Germany

Registered at: AG Charlottenburg (HRB 160341 B)
Managing Director: Kral Wolpers
VAT identification number: DE227895748

info@berlin-product-people.com
telephone: +49 30 24781647
fax: +49 30 247816479

HANDS-ON AGILE™ fieldnotes™