Interview Questions for IT Manager

# 

**Scrum Master Interview Questions and Answers**

[Agile & Scrum-Related]

**1.Explain to me the Agile mindset in your own terms?**

If you can’t answer this question on your own then it’s best to go over the Agile Manifesto a couple of times.

We are uncovering better ways of developing  
software by doing it and helping others do it.  
Through this work we have come to value:  
**Individuals and interactions over processes and tools  
Working software over comprehensive documentation  
Customer collaboration over contract negotiation  
Responding to change over following a plan**  
That is, while there is value in the items on  
the right, we value the items on the left more.

Our highest priority is to satisfy the customer  
through early and continuous delivery  
of valuable software.

Welcome changing requirements, even late in  
development. Agile processes harness change for  
the customer's competitive advantage.

Deliver working software frequently, from a  
couple of weeks to a couple of months, with a  
preference to the shorter timescale.

Business people and developers must work  
together daily throughout the project.

Build projects around motivated individuals.  
Give them the environment and support they need,  
and trust them to get the job done.

The most efficient and effective method of  
conveying information to and within a development  
team is face-to-face conversation.

Working software is the primary measure of progress.

Agile processes promote sustainable development.  
The sponsors, developers, and users should be able  
to maintain a constant pace indefinitely.

Continuous attention to technical excellence  
and good design enhances agility.

Simplicity--the art of maximizing the amount  
of work not done--is essential.

The best architectures, requirements, and designs  
emerge from self-organizing teams.

At regular intervals, the team reflects on how  
to become more effective, then tunes and adjusts  
its behavior accordingly.

2.**How can you say that Agile is working in your organization?**

 When the products you deliver to the end-user result in higher customer retention rates, upsells, and more customer acquisition.

 When the team genuinely feels happy, it can be considered that you are doing alright. There are no team members complaining about work environments. Nobody is leaving (or getting fired) the team.

 Lots of submitted job applications can also be considered as a metric for a well-performing Agile organization.

 If there aren’t piles of technical debt, bugs aren’t present, and there’s less time spent on maintenance, you can say that Agile is going well in your company.

**3.What is the difference between Agile and Scrum (another newbie question)?**

If you still struggle with explaining in simple terms, here’s a more straightforward definition that can make things easier for you: Agile is a philosophy. Scrum is a framework and subset of Agile.

Agile describes a set of ideals and values. In contrast, Scrum gives you principles through which you can make your project management easier.

**4.What’s your criticism of Scrum?**

This can be quite an interesting question as it’s quite unusual. It shows your point of view and rationalization process to a degree. That’s something you should personally answer without having someone else tell you the “correct” answer. Some developers consider the [daily scrums](https://www.scrum.org/resources/what-is-a-daily-scrum) to be a waste of time to a certain degree. Others believe that sprints can be managed better allocating more time to fixing the technical debt.

[The Role of the Scrum Master]

4.How do you demonstrate servant leadership?

As a scrum master in a development team, the last thing you want to do is blindly follow directions.

Before taking on a project, always assess the challenges it might bring to the table and how that may benefit the development team in its path to eternal growth.

“[Blood and Thunder!](https://www.wowhead.com/quest=25949/blood-and-thunder#criteria-of)” - in other words, be loyal to your team and try to win their trust. Put your team on a pedestal. Protect your fellow warriors from unreasonable deadlines or interruptions during project development. Acknowledge the team members’ successes. But don’t try to protect them from the consequences of their own behavior.

To ensure that the developers have smooth work progress without many hurdles along the way, you need to ensure that all task details are conveyed and understood by each member of the team.

Spend time with the PO reviewing, testing, and refining all project requirements. Nothing should slip through the cracks.

5.What makes a good team?

A good team is a team in which:

 A developer doesn’t think, on their way back home, “What a waste of time… I can no longer stand Joe, damn… what a BS team.” but instead says “Alright, let me think how I can solve this complicated bug that everybody else struggles with.”

 No one is [siloed](https://blog.trello.com/tips-to-improve-cross-team-collaboration#:~:text=The%20silo%20effect%20occurs%20when,productivity%20suffers%20because%20of%20it.). Instead, everybody exchanges ideas for solving mutual bugs.

 You won’t be afraid to ask for help because you’venever witnessed teammates being scoffed and sniped at for asking for such.

 Others don’t take credit for your work.

 Everybody takes responsibility for their actions and no blame or finger-pointing is present.

6. **What would you do when the team doesn’t like your ideas and instead suggests new ones?**

A great behavioral question. A reasonable thing to do would be to ask for a cogent explanation from your team to back up their ideas. If their explanation makes sense, and the whole team is backing it up, maybe you should replace your old idea with the new one your team suggests.

Otherwise, explain to the team why their suggestions aren’t as good as yours and gently tell them that changes won’t be initiated.

7.How would you handle a situation where someone leaves early without notifying anybody?

Obviously, this is not typical behavior. Leaving early without letting anybody know can seem unfair to your whole team. Therefore, you need to make it clear:

1. When someone leaves early, they should inform the team and give a legitimate reason for that.

2. Team members who leave early should still be available on Slack or other communication channels if their help is needed.

8.How should a Scrum Master communicate with a Product Owner?

Always be honest and sincere. Both you and the PO should be aligned in your goals while developing the product backlog which will, later on, influence the project’s success.

9.Should the development team be involved in the early phases of planning?

Absolutely. When you involve the whole team early on, you make sure that the developers are fully committed to completing the project with success. The development team’s participation in the early discovery phases can significantly help with time and cost estimates as well.

10.How would you spread Scrum across your organization?

By talking to other stakeholders in different departments and being transparent about your Agile progress. Another quite effective way would be to invite stakeholders to Scrum reviews so they can see for themselves the efficiency of the team.

11.As a new scrum master, what challenges are you aptly to encounter?

One of the biggest challenges you might stumble upon as a beginner scrum master is burn-out. The tons of meetings you have to attend, emails you need to send, and reports you should create can seem like energy-vampires at first.

Keeping the development team and other stakeholders on the same page can also be a difficult task. For this purpose, you need to create thoroughly-detailed reports to convey the project’s progress to the stakeholders. Keep task details clear and transparent in the backlog so that the scrum team can explicitly see the next step ahead of them.

[Product Backlog]

12.Who is writing the user stories?

In writing user stories, all three parties (the PO, the Scrum Master, and the development team) make contributions. If the development team is isolated from the user story creation process, it might lack the motivation to complete the project successfully, which can result in a lower-quality product.

13.What’s the structure of a user story?

 A user-story description.

 Acceptance criteria.

 Performance criteria.

 Tracking criteria.

 Implementable within the Sprint.

 All UI deliverables are available.

14.Why aren’t user stories estimated in person-hours?

Because it shifts the team focus from value creation to saving costs. Estimating man-hours involves an unparalleled level of precision which can be time-consuming as well.

The product owner is constantly adding items throughout the sprint. Is that a good or a bad thing?

This is also a great behavioral interview question. Any backlog that goes beyond the scope of 2 to 3 sprints can become an unmanageable fire dumpster.

Adding hundreds of items just for the sake of adding them will lead to nothing else but chaos among the team members. Therefore, it’s the Scrum Master’s job to help the PO structure the [product backlog](https://www.agilealliance.org/glossary/backlog/#:~:text=A%20product%20backlog%20is%20a,that%20a%20team%20works%20on.) in a way that will be easily manageable and won’t overwhelm the dev team with a myriad of tasks.

[Sprint Planning]

15.How can you help the team to work on the most important stories first?

The Product Owner is ultimately the one who identifies [user stories](https://www.atlassian.com/agile/project-management/user-stories) and makes an ordered list of items for the team to follow. It's the scrum master job (along with the PO) to ensure that no secondary user stories are ranked above primary ones. To help the teamwork on the most important user stories, the scrum master can include developers in the initial planning stages when user stories are assembled.

16.How would you assess the value of a story?

 Profits.

 Cost reduction.

 Net promoting scores above 8 (NPS).

 Higher conversion rates (customer sign-ups).

17.How much team capacity is reasonable to allocate for refactoring and fixing bugs?

A good rule of thumb is no more than 30% of the team's capacity should go into fixing bugs and exploring new technologies or ideas. Say, 15% can go into managing technical debt, 10% into fixing bugs, and the rest 5% into developing new ideas.

18.A team member is cherry-picking tasks? How do you deal with that?

The development team has the ultimate authority to select the tasks it's going to be working on. However, if developers are cherry-picking secondary tasks before 1-st class items, they should be advised otherwise. If the team has been working only on secondary tasks and running out of time, you won't be able to ship an MVP when the time comes. Pair programming can be a reasonable solution to solving this problem.

[Daily Scrums]

18.How do you handle developers that act as leaders and turn daily scrums into reporting sessions?

As you may know, no leadership roles are present in Scrum. Yet, it's not uncommon for some developers to showcase leadership skills (superior technical knowledge, great social skills, or deeper level of engagement) and act as "mentors" to the rest of the team. While that can be acceptable, what's not acceptable is to ask other developers for reports. This can reduce the chemistry between team members and result in a productivity stagnation.

*“People who enjoy meetings should not be in charge of anything.” – Thomas Sowell*

19.How do you handle developers who consider daily Scrum to be a waste of time and therefore avoid them?

This can be quite a severe problem for the team environment. Such members can turn out to be quite toxic at their core and convey only negative vibes, ruining the team's spirit and productivity.

Here's how you can address the situation:

1. Privately discuss, with the team member in question, their motives and attitudes. Try to figure out what's going on in their head. Maybe they need to be further trained on the principles of Scrum.

2. If no change has occurred in the developer's attitude afterward, it's advisable to schedule a meeting with the department manager.

3. If you still can't align the particular developer's vision with your own and the team's, then it might be reasonable to reassign the developer to a different team, a Kanban team for example, where they won't be forced to step out of their comfort zone.

20.What if no stakeholders are present at the daily scrums?

Some people think stakeholders participating in daily scrums can turn the meeting into a reporting session. Not always true, though. In fact, it can improve communication between the development team and upper management.

21.How do you handle daily scrums with remote teams?

The only way you can approach this challenge and overcome it is, ostensibly, through video communication. Slack can be a perfect tool for leading your video calls with distributed teams. It's important to note that the scrum team's size shouldn't go over ten people (recommended scrum team size is from 5 to 8 members). Bigger teams may require you to switch over to the SAFe framework. Getting familiar with SAFe, however, is a different topic.

[Sprint Retrospectives]

22.Who are the people that should take part in the Scrum retrospective?

[Sprint Retrospectives](https://www.scrum.org/resources/what-is-a-sprint-retrospective), according to the Scrum guide, are made only for the scrum team, the product owner, as well as the scrum master.

23.Should you check up on the team’s health (energy) during a retrospective?

It’s useful to measure the team’s health during sprint retrospectives in terms of their engagement and satisfaction. This can influence their productivity which respectively impacts the quality of the final product. You can measure the development team’s health by sending out an anonymous 2-minutes questionnaire to team members before the retrospective. A preferable questionnaire is one that features a simple scale from 1 (terrible) to 5 (awesome) as optional answers every question.

24.How can you avoid your retrospectives getting overcome by boredom?

The scrum master can acknowledge the developers’ suggestions for conducting retrospectives and thus encourage them to participate actively.

However, not every single meeting will go as you’ve planned it. Sometimes, retrospectives can be boring - accept it and move on. Next time try a different approach.

[Transition to Scrum]

24.How would you handle a Scrum transition of an organization that was heavily reliant on Waterfall before?

For the Waterfall-to-Scrum transition to happen, there needs to be experienced people in the team who have been practicing Scrum previously or are quite familiar with it to initiate the transition. Something else you can do to smooth out the transition would be to upgrade the project management tools you are using. Such tools (advocates of Scrum) can ease your team communication while accelerating the pace at which progress moves towards the goal line.

You can also conduct Scrum-related workshops, on-site practices & exercises about user stories, time & cost estimations, collaboration tools, and other tasks.

25.How can you prepare for a Scrum transition?

The most important thing when transitioning to Scrum is to first listen and observe everything and everybody. Try to conduct as many interviews as you can with literally everybody in your organization - QA, UX, UI designers, product managers, developers, etc.

This will allow you to identify existing underlying problems in your organization.

Once you have a clear view of the things that may act as impediments in your Scrum transition, you can address and eradicate them. This process will typically take between four and twelve weeks, depending on the size of your company.

[Scrum Anti-Patterns]

26.What makes a bad Scrum Master?

When you, as a scrum master, allow stakeholders to disrupt the development team’s workflow, you aren’t doing your job right.

You have to keep your team strictly focused on the tasks they are working on. Don’t allow stakeholders to turn daily scrums into reporting sessions - it kills productive collaboration.

A bad scrum master will sweep team conflicts or problems under the rug hoping they’d just disappear with time passing. On the contrary, this can cause more significant problems further down the line.

Developers are acting as bullies during the sprint retrospective, belittling other team members. This is actually the scrum master’s problem as it shows they aren’t interested in the team’s well being. The scrum retrospective should feel like a safe place for all members, including introverts, where they can share their struggles and improve.

26. What can go wrong in a retrospective and how to avoid it?

Don’t force team members to participate in the scrum retrospectives. It’s probably the scrum master’s fault for not making the meeting worth the team members’ time.

Developers should be fueled by motivation & inspiration, not by fear and order.

Don’t postpone items for the next retrospective.

Suppose there’s a blame culture going on during the sprint retrospective. In that case, you should take measures to prevent that from happening further down the road. Otherwise, your team might completely break up.

27.How can you (as a Scrum Master) identify where you need to improve?

Be susceptible to feedback. Ask your team and stakeholders what you can do to help them (and thus improve yourself as well).

Experiment with your team’s suggestions. Search for innovative solutions to lead your team.

Maybe having a fully dedicated retrospective meeting spanning a few hours can be much more useful than your past attempts at stuffing everything that happened during the sprint in a 10-minute meeting.

[Agile Metrics]

28. 2Your team is failing to meet deadlines, and its velocity is volatile. What are the most probable reasons for that? How would you handle such a situation?

Many factors make a Scrum Team’s velocity volatile:

 New team members unfamiliar with the environment and the organization’s ways of doing things.

 Developers leaving the team or getting fired.

 The team working on projects that aren’t anything similar to what they’ve encountered before.

 Unexpected technical debt and bugs.

 Holidays, sick leaves, maternity leaves.

 Scope changes.

 The product backlog is just a mess. Developers struggle with selecting primary tasks and instead end up working on secondary assignments.

29. What are suitable Agile metrics for leading Scrum?

First, you need to use metrics that concern the team as a whole and avoid individual metrics.

Second, consider context as well. For example, members being on holiday or sick leaves can result in changes in your metrics.

Examples of suitable agile metrics are:

 Lead time.

 Cycle time.

 Many defects escaping to production.

 The ratio of fixing work to create new value.

30. What are the roles in Scrum?  
Scrum prescribes only three roles: the Product Owner, Scrum Master, and the Delivery Team. These roles should ideally be cross-functional and not shared among other projects. Many Scrum Masters have not had the opportunity to work with a team that was cross-functional or dedicated due to the organization’s resistance or inability to allow for what some refer to as a “luxury.” This question may lead the interviewer to ask how you would handle working with a team that did not have a designer or tester within the team or how you would handle a team that was not dedicated. Be ready!

31.What is the Daily Stand-Up?

One of the interview questions on Agile is sure to be about the Daily Stand-Up. The answer? Every day, preferably in the morning, the team meets for no more than 15 minutes to answer three questions:  
  
What did you do yesterday?  
What do you plan on doing today?  
Are there any blocks or impediments that keep you from doing your work?  
  
This Scrum ceremony is not meant to be a status meeting for stakeholders, but a way to energize the team and get them to set the focus for the day.

1. Describe what happens in the sprint planning meeting.In Sprint Planning, the Product Owner presents the goal of the sprint and discusses the high priority product backlog items. The Delivery team then chooses the amount of work for the next sprint.

2. What is the role of the Scrum Master?Here’s how to handle a Scrum Master interview question like this: The Scrum Master serves the team and shields them from any distractions that could prevent them from completing a sprint goal. They also remove blocks, teach the team to become self-organized, and serve as a coach who teaches Agile and Scrum values and principles.

Name some other Agile frameworks.

There are other frameworks in addition to Scrum, such as Kanban, Test Driven Development, and Feature-Driven Development. Mention frameworks you have followed and provide scenarios.

What type of metrics or reports have you used?

Sprint, release burn-down and burn-up charts are standard reports. Most companies also want to understand how many stories were committed versus completed per sprint and the number of defects identified post-release to production.

. What is a burn-down chart?

A burn-down chart displays the amount of work a team has burned through—such as hours during the sprint. Discuss how you have used these in the past.

2.

What is retrospective?

A retrospective is a meeting to inspect and adapt the process. This Agile methodology interview question is looking for the many ways to conduct a retrospective—so be ready to explain one or two formats.