The guide to retrospectives – remote or in person

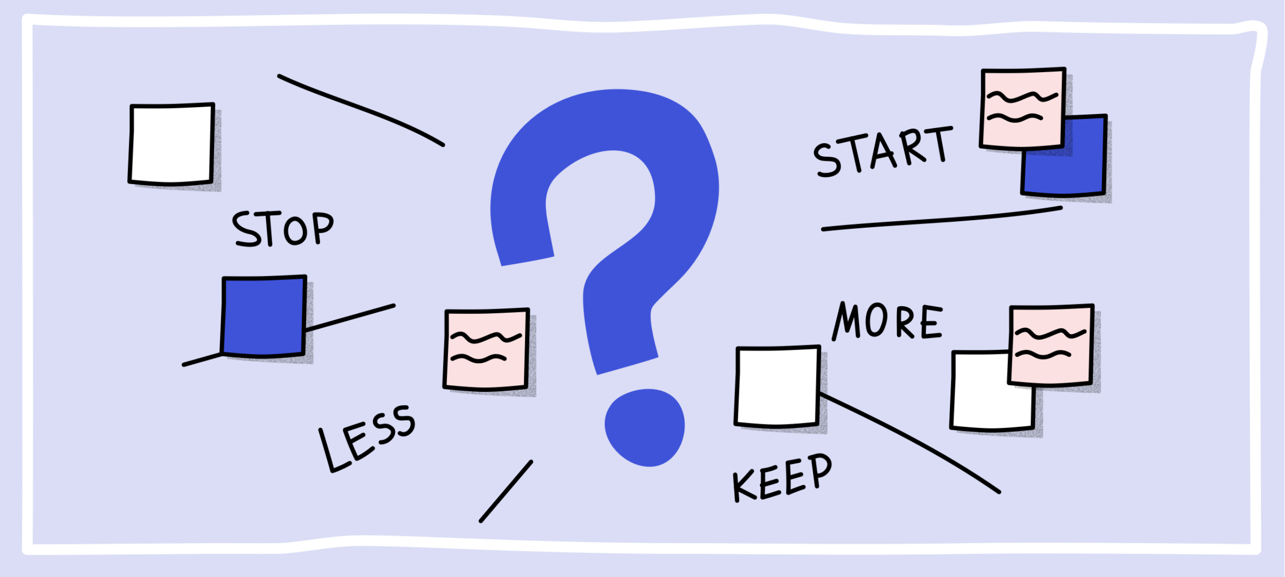
Michael de le Maza
Agile Coach

What is a sprint re...

Sometimes it feels like your team is on a hamster wheel. You’re constantly producing, the wheel keeps spinning, your legs keep moving faster, and eventually, you end up burnt out and exhausted.

But, what if you were to step off that wheel periodically? Maybe you’d realize that a little grease makes things a whole lot easier. Or perhaps you’d figure out a way to add a motor – meaning you didn’t need to work nearly as hard.

Those periodic opportunities to evaluate your work are the magic behind retrospectives. Let’s talk about what they are and how they can save your team a whole lot of fruitless spinning.



What is a retrospective?

A retrospective is a meeting where a team examines how it's working together with the goal of improving their future collaborations.

You might also hear retrospective meetings referred to as Scrum retrospectives, sprint retrospectives, or even post-mortems. Agile teams perform retrospectives at the end of sprints – which are short periods of time (or timeboxes) in which teams complete a set amount of work.

Retrospectives give teams information about what went well (and what didn’t) that they can use to improve their next sprints.

Why retrospectives are a crucial tool for

teams

When the average organization invests upwards of 15% of their collective time in meetings, I understand if you’re hesitant to pull your team together for yet another sit-down. However, agile retrospectives are well worth the time and effort. Here’s why:

1. Teams can speak their minds

You might have a few vocal team members who are comfortable speaking up when they hit a bump in the road. But, you might have even more team members who just keep their heads down in their work – even if something isn’t working quite right.

A retrospective meeting invites all of your team members to chime in with their insights. What went well in your last sprint? What could’ve gone better? How will you act on that information?

Your retrospective provides an opportunity for everyone to voice their opinions (and not just your most extroverted team members).

2. Challenges are addressed

We all know the sentiment that goes something like, “Insanity is doing the same thing over and over again and expecting a different result.”

It seems sensible on paper, yet it’s a trap that many teams fall into. You get so caught up in getting stuff done, you don’t often step back and evaluate *how* you get stuff done. There are likely sticking points and tensions you aren’t even aware of.

Those all bubble to the surface in project retrospectives. This isn’t just a vent session – it’s a chance for your team to address those challenges and work together more efficiently and effectively.

3. Collaborations improve

That brings us to the biggest benefit of retrospectives. When your agile team takes advantage of these regular intervals to evaluate their collaborations, their processes and output get better and better.

They’ll continue to identify areas for improvement and then actually act on them. The result? An unstoppable team.

How to perform a sprint retrospective in 6 steps

Now that you’re armed with the questions you want to ask, you’re ready to pull your retrospective meeting together. Here are six steps to make it happen.

1. Use Miro’s free retrospective templates

You don’t need to start from scratch. Miro’s retrospective templates will help you discuss positive and negative feedback and plan improvements. Here's one the Miro team loves to use:

See more retrospective templates >>

2. Schedule your sprints

In order to figure out when you’ll be hosting your sprint retrospective meeting, you need to know when your sprints are happening. Schedule those out on your team’s calendar. If you aren’t sure how long you’ll need, two weeks per sprint is a solid starting point. Generally, sprints shouldn’t be longer than a month.

3. Invite your team

A retrospective should happen at the end of each sprint. Put them on the calendar now and add your team members to that meeting event, so you make sure it happens. Your entire team should participate.

4. Ask the right questions

Good news: I’ve already outlined those above. Start with those to get the conversation underway, but be prepared that other talking points will come up. If the team starts to explore an unnecessary tangent, steer them back to evaluating your past sprint and improving your next one.

5. Work to improve your processes

Following your retrospective, your agile team will move into their next sprint. This is not only a time when they’ll be getting more work done, but also improving their processes and interactions based on what was discussed in your retrospective. Make sure you’re actually turning those ideas into action.

6. Discuss how it went

At the end of your next sprint comes another retrospective, where you can discuss how any changes you made panned out. It’s like a snowball that keeps on rolling – your collaborations will continue to improve.

If your team points to the same challenges in this retrospective? That’s a sign that either you didn’t follow through on addressing the issue, or what you tried to implement wasn’t successful.

Beware of the barriers to a successful retrospective

Following the above steps will help you get your retrospective rolling. But, keep your eye out for a few common roadblocks that could throw your retrospective meeting off track.

People

At the end of a sprint, team members might feel a little burned out. Now you’re asking them to share honest and comprehensive feedback, which can be a little stressful. Encourage them to get candid and set that example yourself.

Processes

Would you go to the gym and do the same exercise over and over again? Probably not. Yet, that’s the approach so many teams take to their retrospectives. While templates are helpful, there isn’t a rinse-and-repeat process or retrospective format. Be prepared to probe your systems and processes in different ways to get the most helpful insights.

Tools

Technology is on your side, so use a tool that allows every team member to access and add thoughts and ideas in an easy and non-confrontational structure. Intuitive tools that recognize and highlight contributions can help drive engagement on remote and in-person teams.

Time

Your retrospective shouldn’t be an afterthought. In order for it to be beneficial, you need to dedicate adequate time to it. You don’t want to rush to deliver – you want to make time to increase your *capacity* to deliver. That’s not a five-minute conversation, so leave your team plenty of time to dig deep.

Questions your team should ask (and answer) during your retrospective

So, you’ve reached the end of the sprint and are planning to do a retrospective. If you’re expecting your team to roll up their sleeves and start picking apart their own collaborations, you’re going to be disappointed.

The Scrum Master needs to lead this conversation, and they’ll want to come prepared with some questions to get the discussion rolling. Here are four questions your team should be sure to address.

1. What went well?

Start with the positive and have your team outline all of the things they were pleased with during this sprint. Did they fix more bugs than they thought they would? Did they solve a difficult problem? Navigate around a roadblock? **These are things you'll continue doing.**

2. What didn't go well?

This one isn't as fun to answer, but it can be enlightening. Your team should share their frustrations and challenges from this past sprint – and they should do so in a constructive and respectful way, of course. **These are things you'll stop doing.**

3. What ideas do we have for next time?

Your retrospectives are about continuous improvement. This question will help your team figure out how they'll apply what they learned to their next sprints, so things can run even smoother. **These are things you'll continue doing.**

4. How will we implement these actions?

If you answer the above questions, all you have is a bunch of information. In order for it to be valuable, you need to put it into action. That’s what this question is all about. It requires that your team identifies what you’ll actually *do* with the things you’ve uncovered. **These are your action items.**

4 tips for mastering your next retrospective

If you want to make the most of your retrospective meeting (and we know you do), here are a few tips to ensure you have an honest and value-packed conversation.

1. Bring back the icebreaker

Your team already knows each other, but that doesn’t mean that an icebreaker isn’t valuable. Have everybody name the funniest moment of the project or do something totally off-the-wall like asking them to share one word that captures how they feel that day.

Doing this sets a positive vibe and reminds everybody that you’re a team, and this isn’t a finger-pointing session. Additionally, quality retrospectives depend on engagement, and icebreakers are extremely successful tools for getting everybody involved.

2. Gather feedback before the meeting

If you blindside your team with questions and requests for feedback in the actual retrospective, prepare for a lot of crickets and blank looks. Empower them to come prepared to that meeting with valuable insights by gathering feedback ahead of time.

Setting up a whiteboard tool like Miro gives team members a place to drop all of their thoughts and talking points on virtual sticky notes. Plus, it's approachable. It can be hard for people to speak up when they feel their feedback is critical, and this online option can feel a lot more manageable.

3. Ask probing questions

While you might start with some surface-level observations, sprint retrospectives are meant to dig below the surface so problems are addressed at their deepest level. You want to get down to root causes.

So, for example, if your team brings up that one of the sprint's challenges was the number of bugs they had to deal with, your first inclination might be to ask, “How can we address bugs more efficiently moving forward?” You’d be even better off asking, “Why do we have so many bugs in the first place?”

4. Keep the iteration independent

It can be deceptively easy to refer to the past. Getting stuck on old problems, conversations, and projects can muck up the analysis of the current iteration. Stay focused on your single sprint. It’ll make the retrospective far more manageable, while also ensuring any new team members (who don’t have the context from previous sprints) are able to form independent thoughts and chime in with their perspectives.

Now that you have the basic tips and tricks of running a sprint retrospective, learn how to master the art of a remote retro in the next chapter of this guide.