

# Sprint Retrospective

---

- What is it?
  - 1-2 hour meeting following each Sprint Demo
  - Attended by Product Owner, Team, ScrumMaster
  - Usually a neutral person will be invited in to facilitate
  - What's working and what could work better

8/26/2023 12:50 PM

vijaynathani.github.io

Page 1

Time boxed to 3 hours.

A rough guide to timings is a team need 30 minutes retrospective time per week under review so using this formula allow 2 hours for a monthly retrospective and a whole day for a retrospective of a several months work.

Team and ScrumMaster talk about the last Sprint.

This meeting may not be attended by Product Owner, if the team feels it will be counter-productive.

Any other person can be invited if the Team so wishes.

# Process – 6 steps

---

- Security first
- Collect facts
- Ask: What went well?
- Ask: What can be improved?
- Who is in control?
- Prioritize

# Step 1: Security First

---

- "Regardless of what we discover, we understand and truly believe:
  - that everyone did the best job they could,
  - given what they knew at the time,
  - their skills and abilities,
  - the resources available and
  - the situation at hand." - Kerth (2001).

Everyone in the world is doing the best, given their level of consciousness. The way to improve then is to improve our level of consciousness i.e. to become more aware.

8/26/2023 12:50 PM

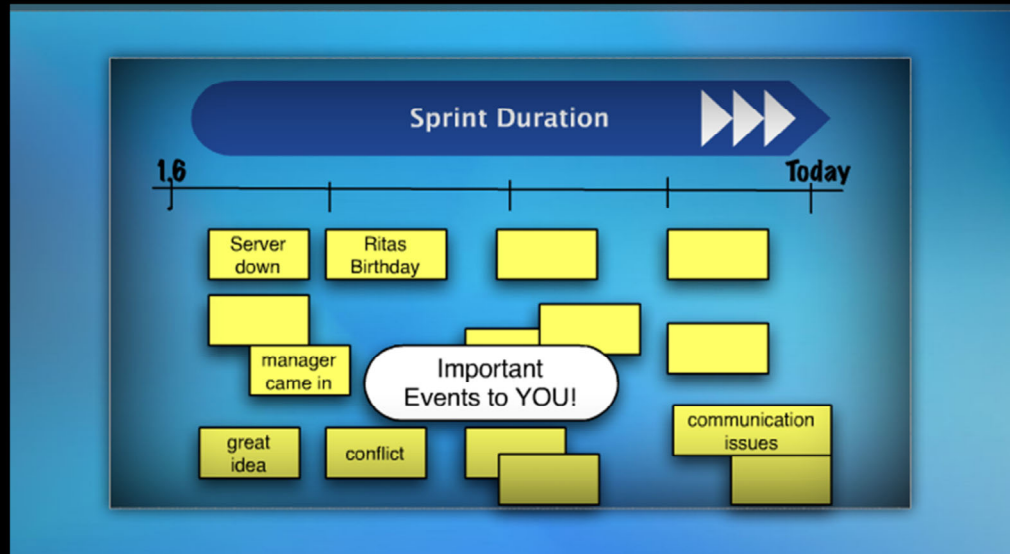
vijaynathani.github.io

Page 3

Keep this kind of message on a flipchart at the start of meeting.

Keep some distance for this meeting: Do not do it immediately after the sprint review. Don't invite senior management. Use a dedicated room (no disturbances).

## Step 2: Timeline



8/26/2023 12:50 PM

vijaynathani.github.io

Page 4

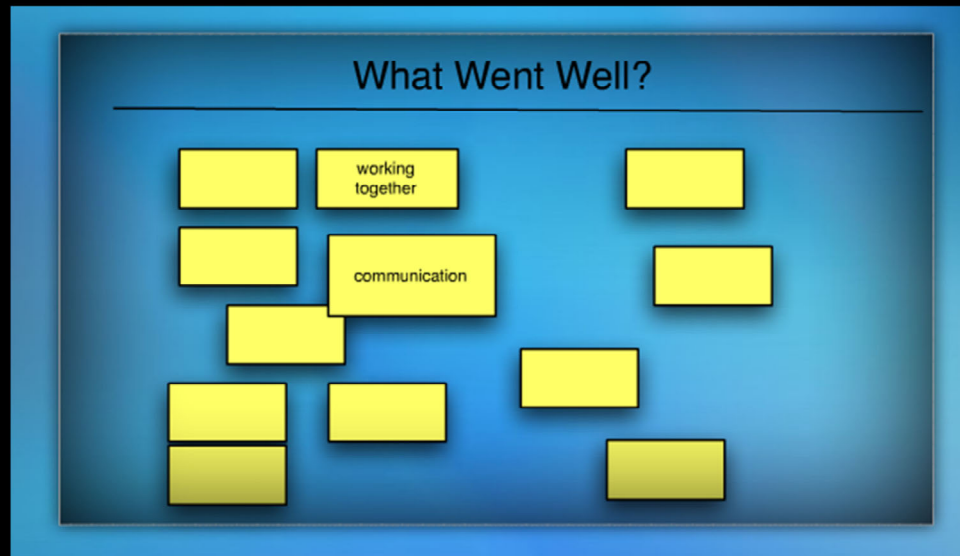
This step takes 10 to 30 minutes.

Here we talk about facts or events, not emotions. Everyone says items that are important for them, not for others.

Why do this? Because people tend to forget the items that have occurred or we are not sure whether everyone noticed an important event e.g. One team member had to sit late to finish his/her task. Everyone else went home early.

Each person says a short story about the event and puts the post-it at that date.

## Step 3



8/26/2023 12:50 PM

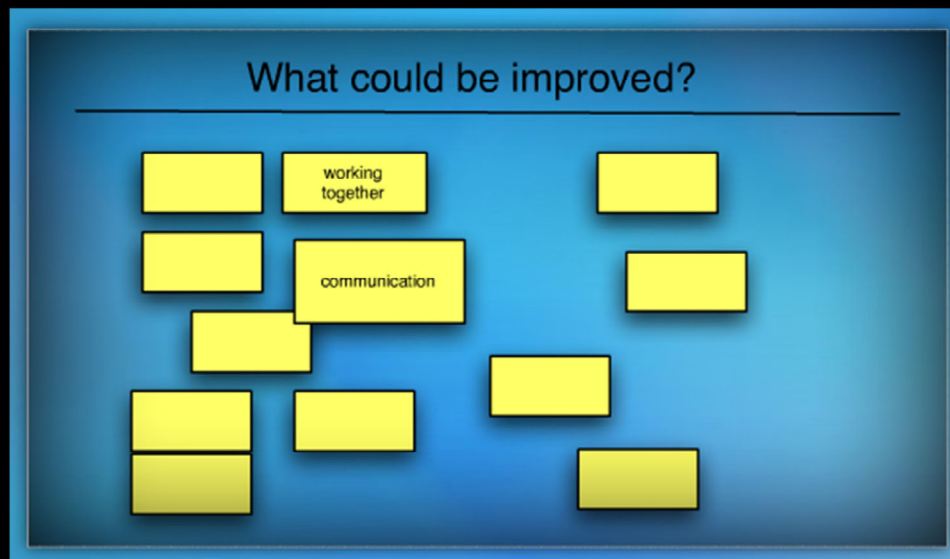
vijaynathani.github.io

Page 5

Everyone comes in good mood.

The sprint was great because .....

## Step 4



8/26/2023 12:50 PM

[vijaynathani.github.io](https://vijaynathani.github.io)

Page 6

Is there anything that we would repeat, if we have the opportunity. If yes then "No improvement"

If no, then what can be improved.

# Mark every item

---

- C – Caused by Scrum
- E – Exposed by Scrum
- U – Unrelated to Scrum

8/26/2023 12:50 PM

vijaynathani.github.io

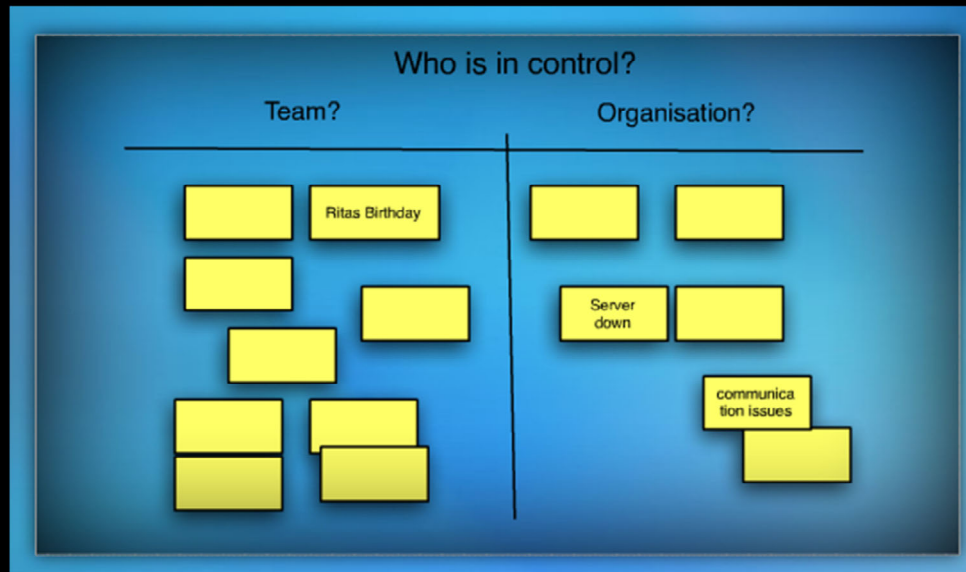
Page 7

At the end of the meeting, mark each item on the What's Working / What Could Work Better lists as either:

C = caused by Scrum (would not be happening without Scrum)  
E = exposed by Scrum (would be there even if team were not using Scrum)  
U = unrelated to Scrum (like the weather)

Then, add up the C, E, and U in each column

## Step 5: Items to Improve



8/26/2023 12:50 PM

vijaynathani.github.io

Page 8

Who can make that improvement? Team or Organization.



# Sample Picture



If the team is distributed, then consider tools like IdeaBoardz, Pointing Poker, Stromboard.

## Part 1

Create 3 large lists (whiteboard or flipchart)

What's working

What's could work better

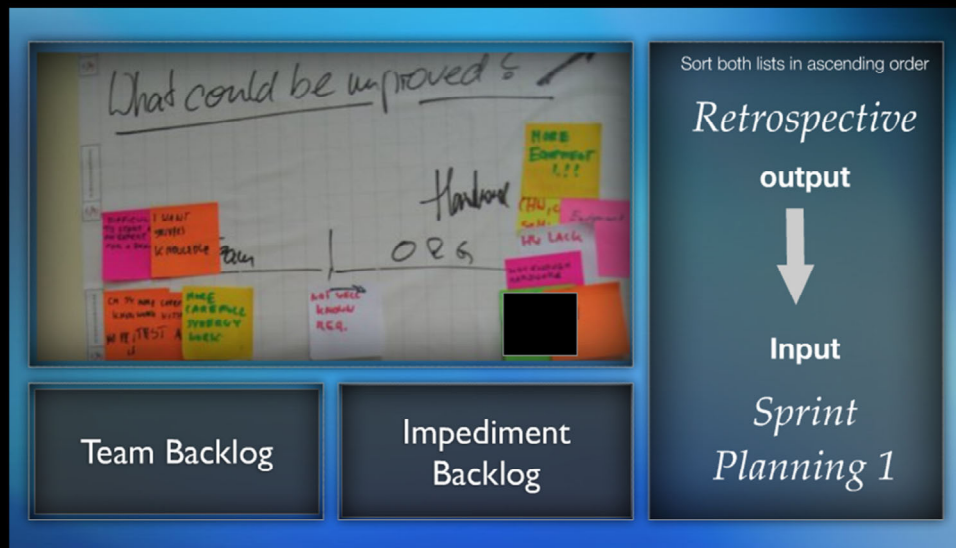
Things to try in the next Sprint

Go around the room, and give each person an opportunity to add 1 or more items to the 3 lists

If people agree with something already on the lists, put a tick mark next to them

Select a subset of the "Things to try..." list to try in the next Sprint (ScrumMaster responsible for tracking this)

# Output should be used



8/26/2023 12:50 PM

vijaynathani.github.io

Page 10

Team backlog is for the team to improve in the next sprint. E.g. Move one story to done and then move to another. So that testers can begin testing the done story. Impediment backlog is for the ScrumMaster to follow up e.g. If team is changing again and again, the velocity will be low.

# Retrospective

