Welcome to the overview of how an agile team functions to the best of it’s abilities. I like to start out by getting a sense of what you know already so that I can skim the parts you have a great sense of, and where to go into more detail.

I’d like to remind you of ground rules. If we cannot trust that we will keep to these rules then I’ll ask you to leave now.

* We come from a place where we want to listen;
* We realize that we all come from a place of good;
* We allow ourselves to be open to the ideas presented;
* We will respect each other as it relates to differences of opinions, to time, and to their feelings;
* I will listen to what you have to say;
* I will be willing to be challenged and change my mind when it makes sense;
* I will do my best to explain things, while realizing I may be imperfect in that explanation;
* I will be respectful to each and every one of you as I realize you are coming from a good place.

One of the rules from every Global Scrum Gathering I’ve attended is this: The Rule of Two Feet. If you no longer find this useful to you then please leave. I don’t want you to be here if you cannot be present in this space. To use something from T-Mobile – Be Here Now; barring an emergency, please keep your phones and laptops away. They become a distraction from the goals and may be distracting to those around you.

True / False / It Depends

1. The Scrum team decides how they will operate
2. The Scrum team is empowered to make decisions
3. The Scrum team should choose their work
4. The stakeholders should ask the developers how the work is going
5. The Scrum Master should be providing updates to the customer directly
6. The Scrum Master coaches the team on the process
7. The Product Owner sets the priority of the backlog
8. The Developers do not prioritize the backlog
9. The Scrum Master runs the Scrum Ceremonies
10. The HR Manager prioritizes the backlog
11. Periodic testing Sprints are the preferred mechanism for ensuring high quality.
12. The Development Team needs full details of each PBI in order to estimate the size or complexity of the item.
13. The Scrum Team can request an Abnormal Termination if the Sprint Plan is significantly invalidated or interrupted by external factors.
14. Management can replace one PBI in the Sprint Backlog with another during the Sprint.
15. The ScrumMaster assigns tasks to Team members.
16. For most efficient flow, the Development Team should route all Product Owner questions through the ScrumMaster.
17. The ScrumMaster is responsible for protecting the Development Team from both internal and external distractions.
18. During a Sprint, anyone with sufficient formal authority in the organization may require the Development Team to take on additional tasks.
19. Scrum allows individual team members to be specialists.
20. With the self-organizing team concept, there is no longer any need for managers.
21. It Depends – how the team functions within the structure is ok. By example, the Scrum Master has the requirement to make sure that the process is followed, however *how* the team chooses to implement the process is ok. The SM builds the house, the Team decides what color to paint and where the furniture goes.
22. True. Within the team they can make decisions as to who may do the work, determine what they can accomplish and the team agreements
23. True. Team members, not the SM or anyone else make the decision. The HR manager may influence by providing input to team members for furtherance of their career goals or to stretch themselves. However, if they or the team needs them to focus on what they can do best, they make the decision.
24. It depends. There are times when that direct communication may be needed or desired. Team members should feel empowered to do that, however if the Scrum Master finds that it is a distraction to the team or the accomplishment of the sprint goal, they should step in. At no point should the stakeholder introduce a change from an agreed to PBI; if an error was made in the requirement, the may story close and a new story would be prioritized.
25. True. The Scrum Master using boards provides the team’s progress. However, if stakeholders need a quick update and the product owner isn’t present then this can be done.
26. True.
27. True. They prioritize the backlog based upon business priorities and other factors.
28. True. They may provide feedback or influence the Product Owner’s Prioritization; however, they should choose the work based upon priority barring a technical hurdle in accomplishing the work.
29. False. The Scrum Master ensures that the ceremonies are scheduled and that they occur. They can facilitate the meetings and help keep them focused, however the team can run the meeting as appropriate. **Note:**It is incumbent on the Scrum Master to run the meeting if the team isn’t stepping up or is a lack of maturity to accomplish the task. Examples would be if the team members are not allowing others to make their points at a retrospective, they start giving status updates during the stand up, or are breaking the timeboxes for these ceremonies.
30. False, any manager – regardless of their role – goes through the Product Owner.
31. True
32. False. The team should have a ‘good enough’ understanding of the requirement including a description, purpose, and acceptance criteria so that they can accomplish the work. If important details are missing, they should be provided prior to estimation and bringing into the sprint backlog.
33. True. If the majority of the work cannot be accomplished, they can request the termination and the PO, SM, and team make the call to end the sprint. If the sprint is ended, a retrospective ***must*** occur to determine how/why it occurred and how to reduce the chances of it from happening again.
34. False. The Product Owner owns the backlog, they can influence the decision and if the PO agrees they can bring it to the team and determine if it’s possible. In this case, the team decides on the work as the PO may not be aware of its implications.
35. False. Team members are empowered to make those decisions.
36. False. Scrum is about open communication and transparency. If there are questions, the team is empowered to ask them. However, the SM could be part of them, but at a minimum advised that they occurred as it may be a process problem in the case of a misunderstood story. The SM could then put it on the list of things to possibly discuss during the retro.
37. True. The SM is responsible for protecting the team in this manner and holding the team to their commitment because they are minimizing these interruptions.
38. False. Formal authority has no place in Scrum, this is part of the cultural shift in the organization.
39. True. While we want there to be cross-functional teams, people may have natural specialties based upon their experience. However, they can strive to make sure that the knowledge passes on to the rest of the team so we just don’t have a team of experts.
40. False. The concept of self-organization is limited to the Development Team deciding how best to realize the Sprint Goal. Self-organization does not extend to administrative or human resources matters.

Roles

Scrum Master

* Defender of the Process
  + Done by educating and coaching the team and leading by example
  + Done by educating the product owner
* Defending the Team
  + Helping to eliminate roadblocks where possible
  + Helping to decrease interrupts
  + Being an information radiator by keeping externally facing reports up to date
* Holding the Team accountable
  + Making sure the team meets their commitments to increase trust
  + Watching team members hold each other accountable
  + Making sure they work on the agreed to prioritization
* Empower the team
  + While the team needs to work within the framework, allow them to assign work
  + Ensure they are heard and where it makes sense go with a change to test it out

Product Owner

* Voice of the customer
  + Has a clear understanding of customer requests
  + Can speak for the customer when they aren’t around
  + Works with the customer to write/review stories for the team to work
* Owner of the Backlog
  + Responsible for prioritizing the work
  + Responsible for taking team input, understanding its impacts, and prioritizing correctly
  + Communicating the backlog to all interested parties and can defend the prioritization

SCRUM Team

* Understanding the Technologies
  + Taking the time to review or POC items which are new
  + Informing the Product Owner and Scrum Master how this impacts the backlog.
  + Seeing if there are new ways to do the work.
* Owning the work
  + Making sure they understand acceptance criteria
  + Making a commitment to complete the story
  + Working with others to increase their cross-knowledge
* A team of specialists and generalists
  + Can have expert knowledge
  + Pairing so that others can gain the knowledge
  + Need to be able to set up testing environments to make sure the code is stable
  + If they are a tester, need to gain knowledge of what is reasonable to test
* Innovate
  + Find ways to work better
  + Find other ways to accomplish the customer goals
  + Be willing to adapt to the process or provide positive feedback to improve the process.

Getting Things Done

**True/False**

Agile/Scrum lets us get things done faster:

False – Agile’s purpose is to find what doesn’t work fast by producing quick feedback loops with the customer.

Agile/Scrum doesn’t provide exact time of completion

True – much like ‘traditional’ project management there isn’t a way to provide a fixed date of completion. Agile can provide a decent range of the delivery – and is built to deliver incrementally providing value as soon as it is completed.

Agile does use estimation and a dynamic ‘velocity’ so that as things change we can provide more immediate updates as to the estimated completion date.

Agile/Scrum handles change easily

True – Unlike other methods, we embrace change – and can do it with a lesser amount of ‘paperwork.’ Having a clear goal, acceptance criteria, and vision is still important.

**However,** if the delivery window still needs to be maintained then something must come out of the project work and that needs to be decided up front.

Agile/Scrum handles interruptions easily

False – Interruptions, by their definition, are detractors to the team’s progress and have nothing to do with the sprint goal or prioritization.

While some interruptions are unavoidable, therefore the Scrum Master and Product Owner need to be part of that discussion. By being in two-week sprints, the question could be – does this need to be done now or can it wait a week for us to start? Because of way that people work, if the interruption is early in the sprint it is easier to minimize the impact. Towards the end of the sprint where there is a push for code quality, integration testing, and others the interruptions could have a very detrimental impact to the customers and are discouraged.

**How to we predict**

Prediction is done via a velocity based upon the teams estimate via story points. A story point is a measure of time plus complexity (or risk). It is beyond the scope to discuss story points, but realize that they are like any sort of measurement; 2 is twice the effort of 1, 5 is more than 3 times the effort of 1, but less than 5 times the effort.

After three sprints (6 weeks) the team will have done enough work to get a good average of the story points per two-week sprint. As the team will do an order of magnitude estimate of the work when they put it into the backlog. At that point it is simple division. Likewise, if there is a certain singular feature-set we want to bring out *and* the entire team were devoted to it, it’s another simple division.

We don’t predict based upon single people because something of higher priority may arise that takes that person off the work, which is why it’s a team commitment.

Realize that the average story points would include *recorded* interruptions. You’d have to do more math to remove those interruptions from the average to get a more accurate estimate. This is why reducing unplanned work is important, if we plan it we get more accurate predictions.

“But we need to tell customers when it will get done.”

This is a true statement; how did you tell that this in the past? Were we always on time? Did we pad our dates so we’d build in that space for uncertainty? Were we being truthful or transparent to the customer? With frequent updates to the customer throughout the process – at a minimum of every two weeks – they can make the determination if they want to call it “done” and deliver that work. If they don’t want to call it early they have immediate feedback as to where the project stands.

We can, with some degree of accuracy, let customers know where we are at and when we might get it done. This may sound like a cop-out, but it brings us to this point.

**We Estimate at the Last Responsible Moment**

With agile, given that the landscape from the customer changes frequently, we don’t spend a large amount of time doing *detailed* estimation. We still need to do some estimation up front, but (again) the is an order of magnitude estimate *after* we have determined the most important stuff to do. The items we think would be part of the first 6 weeks would be broken down to their logical, deliverable, work products and estimated accordingly.

Generally, we do not go into detailed break downs further down the backlog *unless* we know that an item will require stories that must be done early on. If there is a need, there must be an acceptance that some of the work may be wasteful if the client needs change, technologies change, or T-Mobile goes another direction.

“When can we expect this”

This is the most difficult question