Q9

* Story won’t be marked as done as QA identified defect while testing and it’s not meeting to acceptance criteria.
* If QA tests the US on last day, then when will be Iteration review for that user story with PO. Team is running late as per time line.
* US will be can be consider as pillow-over and pushed to the next iteration after having discussion with PO.
* One case can be possible if US will be in working situation and missing small requirement which is not breaking the flow then log a defect and continue. [Mostly won’t suggested and acceptable by PO]

Q10

* Would be 2 things: Team doesn’t have right skill-set OR Team isn’t committed to work.
* Need to keep track the progress of work day-to-day. Which will give me the clear picture, why this problem.
* If something is going in wrong direction, then I need to guide them and back to the track.
* Make team to be aware for Scrum Values.

Q11

* While working on development, technical debt should not grow.
* Will keep work in technical debt in-between iteration.
* If some scope change/US pushed to next iteration with respect to another dependency/technical architecture issue. These things will give opportunity for team to work on technical debt in parallel.

Q12

* If some internal team would be known before then share work with other teams or Get more people to make the things done on right time.
* Something won’t go well then I have to reply to the senior management with valid reason.
* Things won’t be in my hand, it’s team work and we have to gone through the process.

# Important Points

* Implementing Scrum is not good enough until we make use of Scrum Values.
* Scrum Master shouldn’t say “NO”. Come up with approach to do the things in alternate way with same quality delivery.
* Scrum Master should always say Good things first about team then the improvement areas.
* Scrum Master do conflict resolvement and not the people management.
* Agile coach is like mirror, how are you doing so will give suggestion with respect to process but not the technical solution.
* Solution would be given by consultant.
* Scrum Master/Agile Coach won’t provide solution to the problem. They just mentor and guide the team.
* Scrum Master is dog sheep.
* Scrum core values:
  + Built-in quality
  + Program Execution
  + Alignment
  + Transparency
* Kanban (KAN: Visual || BAN: Card) WIP (Work in-progress) || WL (Work Limit)

STOP Starting and START Finishing

# Action Items

* Ad-hoc work slow-down the work progress. i.e. Ad-hoc can be quick estimation, PROD defect etc
* If delivery isn’t going good, then scrum master will make aware to leadership with facts (Why and What)
* Scrum Master responsibility to help leadership for better delivery instead of only doing scrum ceremonies.

# Iteration

* Better to do re-planning instead of sticking to the plan. (Sometimes need to accept change requirement in between iteration)
* Architect would be part of the team and own task as required. Even through architect effort would not be included in capacity of the team.
* PO and SMEs can join iteration planning as per availability.

# Program Increment

* Acceptance criteria would be defining by PO prior to PI session for feature/US.
* Mostly we won’t wait for PI session to split features into US. Same thing can be done while grooming session.
* High level estimation used to be done or can be done prior PI session. (Grooming session)

# Leadership Team

* Agile is ready to change but leadership won’t change but expect the faster growth. (Here problem with the top management team)

# Agile Team

* The more will interact with customer (PO) then will give more clarity to the customer as what they need. That’s why we have iterative model and changes comes in-between iteration.
* Don’t work beyond capacity else quality will get reduce.

# Facts or Timelines

* Scrum (1-4 weeks for sprint) || SAFe (2 weeks for iteration)
* We can have max 20% enabler at time.
* Architect team run ahead of iteration to provide technical base.
* Automation used to be run behind of iteration.
* If cycle time getting reduce means team productivity got improved. Team productivity can’t be define by velocity.

# ART

* Release manager who works at program level, will decide when need to keep build and when code will move from base environment to PROD. i.e. complete release plan.

# Definitions

## Release Capacity

How much time required to release a planned product (Which involve steps first to last)

## Focus Capacity

How much really development effort needed for product development.

## Velocity

Capacity of team to deliver business values.

Define the limit of work can be done by a team in each iteration.

## Load

How much team has committed to do in one iteration out of velocity?

## Nexus

Single product development at a time. Max can have 12 teams with total of 100 people.

## ART

Will have multiple train. Each train can have max 13 teams. Each train can have 50-125 people. Total ART can have 2000 people.

## Refactor

Can be technical debt. Go back to the coded user story and re-code it to reuse further or maintainability purpose.

## Spike

Something we do for future. R&D or POC.

## Stretch Item

Will be having some dependency or uncertainty to do. If things are good, then only will do else not.

Note: Suggested to keep stretch item as additional so utilize team bandwidth without diverting from team velocity. It’s low confident item.

## Risk

Types of Risk

* Resolved
* Owned
* Accepted
* Mitigated