

<Agile Methodologies>

Agile Testing: Agile is a practice that a QA follows in a dynamic environment where testing requirements keep changing according to the customer needs. It is done parallel to dev activity where QA team receives small piece of code from the dev team for testing.

Scrum: Basically, Scrum is grouping some people. And who are involved in this? There is main 4 guys are there as below:

- a. Scrum Master
- b. Product Owner
- c. Dev Team
- d. QA Team

Note: Agile is a practice that follows the principles of Agile Software development process. And the principle of Agile Software testing is-

1. **Testing moves the project forward** – Continuous testing is the only way to ensure continuous progress.
2. **Testing is not a phase** – Agile team tests alongside the development team to ensure that the features implemented during a given iteration are actually done.
3. **Everyone tests** – In agile testing, the entire team including analysts, developers, and testers test the application. After every iteration, even the customer performs the User Acceptance Testing.
4. **Keep the Code Clean** – The defects are fixed as they are raised within the same iteration.
5. **Lightweight / Less documentation**
6. **Test Driven Development Approach (TDD):** Test Cases are written along with the requirements. Hence, development can be driven by testing. This approach is called Test Driven Development (TDD) and Acceptance Test Driven Development (ATDD).

Important Note: What is the difference between Agile and Scrum?

Ans: Agile is a practice and Scrum is a model. Or you can say Scrum is group of people. So, these scrum people do the Agile practice and that Agile practice follows some principles of Agile software development Approach. And those principles are:

- a. Continuous testing is the only way to ensure the continuous progress.
- b. Testing is not a Separate phase.
- c. Everyone tests in Agile environment.
- d. Very Less documentation.
- e. And the most important thing they follow the TDD approach. Hence, Development can be driven by testing.

Sprint: 14 days or 2 weeks or 10 working days.

⇒ We get business requirements in the form of User Story.

Epic: Collection of large number of features or user stories.

Feature: Group of multiple user stories.

User story: the Actual business requirements. and BA/ BUSniess analyst provides the User story. and we discuss with BA to clarify the requirements or user story. so that we can start our test case development process.

Daily stand up: It is daily standup meeting for 15 minutes happens on daily basis.

Product backlog: It is the pending items where all the epics/user stories are there to be developed in the upcoming sprints.

Sprint backlog: It is the action items for the current sprint basically we pull the user stories from product backlog to the current sprint backlog.

Agile ceremonies: There are 4 agile ceremonies below:

1. **Sprint Planning:** development team, scrum master, product owner.==> Involve entire team to discuss that how much work the team can complete from the product backlog. That body of work then becomes the sprint backlog.
2. **Daily Stand-Up:** development team, scrum master, product owner ==> Stand-up is designed to quickly inform everyone of what's going on across the team.

What did I complete yesterday?

What will I work on today?

Am I blocked by anything?

Sprint Review: development team, scrum master, product owner==> Iteration review is a time to showcase the work of the team. They can be in a casual format like "demo Fridays"

Sprint Retrospective: So basically, we focus on three things in this meeting:

1. What went wrong during the sprint life cycle.
2. What went right during the sprint life cycle.
3. What could we do better for the upcoming sprints.

Velocity and Capacity: **Velocity** is based on actual points completed, which is typically an average of all previous sprints. **Velocity** is used to plan how many products backlog items the team should bring into the next sprint. **Capacity** is how much availability the team has for the sprint.

Pair Programming: **Pair programming** is an **agile** software development technique in which two **programmers** work together at one workstation. The one writes code while the other review his code. Or They can switch their roles accordingly.

Impediments: An **Impediment** is anything that keeps the Team from getting work Done and that slows Velocity. **Impediments** come in many forms: a sick team member, a missing resource, lack of

management support or even a cold team room. If it's blocking the team from doing its work, it's an **Impediment**.

“**Impediments** are anything that can stop the progress of your project or user story or prevent it from getting finished.”

Burn up and Burn Down chart: A **burn-down chart** shows the amount of work remaining on a project (the remaining effort), whereas a **burn-up chart** shows how much work has been completed and the total scope of the project.

Build breaker: An inadvertent mistake by a software developer that sometimes stops the **build** process, or causes unacceptable warnings, and/ or failures in the automated test environments, is known as a '**Build Breaker**'.

Grooming in Agile: Backlog **grooming**, also referred to as backlog refinement or story time, is a recurring event for **agile** product development teams. The primary purpose of a backlog **grooming** session is to ensure the next few sprints worth of user stories in the product backlog are prepared for sprint planning.

Zero sprint – Zero Sprint can be defined as the preparation step of the first sprint in Agile. There are some activities that are required to be done before actually starting the project. These activities are considered as the Zero sprint; the examples of such activities are – setting the environment for development, preparation of backlogs etc.

Spike – Spike is the type of story that can be taken between the sprints. Spikes are commonly used for the activities related to the design or technical issues such as research, design, prototyping, and exploration. There are two types of spikes – functional spikes and technical spikes.

What are the different roles in Scrum?

The three scrum roles i.e., Scrum Master, Product Owner and Team should be explained with the details of few primary responsibilities of each role.

Product owner – A product owner is actually the stakeholder of the project. He represents the project requirements before the team. He is responsible to have a vision of what to build and convey his detailed vision to the team. He is the starting point of an agile scrum software development project.

Scrum team – Scrum team is formed by the collective contribution of individuals who perform for the accomplishment of a particular project. There are around 10 people in scrum team 6 Dev and 4 QA

Scrum master – Scrum master is the leader and the coach for the scrum team who checks whether the scrum team is executing committed tasks properly. He is also responsible to increase the efficiency and productivity of the team so that they can achieve the sprint goal effectively.

What do you know about “Planning Poker” technique?

Answer: Planning poker, also known as Scrum Poker, is a card-based agile technique that is used for planning and estimation. To start a session of planning poker technique, the agile user story is read by the product owner. The steps performed in the poker planning technique are –

1. Each estimator has a deck of poker cards with the values such as 0, 1, 2, 3, 5, and so on, to denote story points, ideal days or something else that the team uses for estimation.
2. Each estimator has a discussion with the product owner and then privately selects a card on the basis of their independent estimation.
3. If the cards with same value are selected by all estimators, it is considered as an estimate. If not, the estimator discusses the high and low value of their estimates.
4. Then again, each estimator privately selects a card and reveals. This process of poker planning is repeated to reach a general agreement.