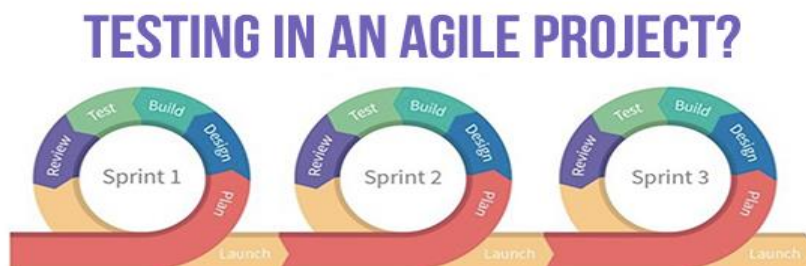


Agile

- Agile mostly used in service-based company
 - In Agile, customer can request for changing in requirement at any point of dev phase.
 - These changes not affect the development of another module.
 - In Agile requirements break into small small pieces and then team work on it.
 - In V model after requirement get fixed the whole team focus on whole product development but in agile methodology, requirements are changing frequently and more focus on module dependent requirement so it's a module driven methodology.
 - Also called as Agile Model (Scrum) / Multi Iterative life cycle
- Agile (means – ability to move quickly and easily)
- Multi Iterative life cycle is the way by which the project is broken up in several smaller pieces. Each iteration is called a Sprint
 - Agile Methods break the product into small incremental builds. These builds are provided in the iterations



Example-

Let's assume we have to create a large website, and the website has 12 web pages that are going to be a part of whole flow .

In an Agile environment, we will work on one page at a time and the iteration is going to be completed in the period of a month. So, at the end of the month there is always a deliverable product that the end user could use and provide its feedback.

- At the end of the sprints you are getting a product

Scrum is one of the version / way / principle / frameworks of Agile

- Around 90 % projects are implemented in Scrum and a popular way to implement agile
- Within Scrum, each iteration is called a Sprint
- After completion of Sprint, software / project that would be delivered to the customer and then customer gives his feedback.
- Customer is always involve in the development method and they give the feedback before the actual end product unlike waterfall model where customer has zero visibility.
- In Agile you have maximum visibility , every 3 weeks , every 4 weeks depending the length of the iteration that the customer chooses , he gets the piece of the software at the end of the sprint

Flow like-

Process is broken into small pieces-

Plan → Build → Test → Review (Sprint 1)

Plan → Build → Test → Review (Sprint 2)

Several increment releases called sprints

Within Scrum each iteration is called as sprint



How do we break the whole thing in?

-In Agile , we write User Stories

-User Stories are the way of which you are going to provide the requirements.

-User Story is contains – what is feature that is going to be developed.

-Each user story is a compact of feature that is going to be develop a piece of a code and at the end project goes live.

-These user stories are **kept is a backlog**

The first is the Product backlog . The product backlog practically has all the user stories (not only the first sprint, second sprint ...)

-Product backlog are the total requirements for the whole project.

-It includes requirement for all modules.

-Product owner gather requirement from the stakeholder

-Product owner is team member for sprint piling meeting.

-Product owner is responsible for creating product backlog.

Sprint Backlog-

-Created by product Owner

-Sprint backlog contains story means complete detail info for the requirement which are require for module development.

-Only the periodized stories are taken in Sprint backlog after sprint planning.

-Prioritization is basically the way by which we are going to assign a level of importance to each user story. Usually done with Product owner and Customer

- Everyone wants to make sure most important and most relevant user stories get executed in the first sprint itself

-Now, Prioritization is done, and let's say we are in sprint 1

How do we keep a track of for the stories on development

- This can be done with daily status call or meeting with each team members and i.e. **daily scrum calls /stand up meeting/status meeting.**
- It is basically a 15 min meeting where the whole agile team is present there.
- The product owner is also present
- Each individual is responsible to answer **three questions - what did you do yesterday, what are you planning to do today and are there any road blocks (impediments / problems)**

Q-Agenda of daily scrum?

A-What is the progress of Project.

- So it is basically a status update.
- Scrum master takes care the track of how many tasks are developed as per plan and he/she try to resolve the blockers if any within the team)

Q-What is Epic?

A-Scrum Epic in Agile Methodology is a large body of work that can be broken down into a number of smaller stories. An Epic can be spread across sprints and even across agile teams.

An Epic can be a high-level description of what the client wants, and accordingly, it has some value attached to it. As we mentioned, an Epic is a high-level requirement, hence its scope can change over the course of time.

Like Modules or main big requirements we can say Epic.

Under Epic multiple User stories linked.

Epics are a helpful way to organize your work and to create a hierarchy. The idea is to break down the work into shippable pieces so that large projects can actually get done on a regular basis



Scrum Master creates a Burn Down Chart

A Burn Down Chart is a graphical representation of what was the work that was required to be executed yesterday , what got completed

Roles –

1. Product owner -

- Product owner gather requirement from the stakeholder
- Product owner is team member for sprint planning meeting.
- Product owner is responsible for creating product backlog.
- Responsible for defining the features in the product

2.Scrum master -

- Protecting the team and a process . Keeps the things flowing and clear the blockers from team members.

3. Team – Developers + Testers + anyone who helps in developing the product .

- Group of professionals who deliver the product (developers, programmers, designers).
- The complete their work within the sprint or cycle.

4.Stakeholder-

- Stakeholder is customer.
- In agile methodology customer is the member of top most body.

-At any phase like in dev,testing or at requirement they can change the requirement.

-they have bunch of requirements related to the project

Artifacts / documents-(All details are above)

- 1.User Stories
- 2.Product backlog
- 3.Sprint backlog
- 4.Burm down chart

Scrum events / Ceremonies / discussion / Meetings -

- **Sprint planning:** Where the entire Scrum team sit together—at the beginning of every Sprint—and decide which story need to be taken on next sprint and discuss with the estimations.
Mostly Product owner, development lead or sr dev and test lead or sr tester do the estimations for any user stories.
- **Daily Scrum:** 15-minute time boxed meeting which held every day of the Sprint, where-
Each individual is responsible to answer three questions -
what did you do yesterday,
what are you planning to do today and
are there any road blocks (impediments / problems)
- **Sprint review:** An informal meeting held at the end of every Sprint where the Scrum team discuss feedback.
- **Sprint retrospective:** A meeting where the Scrum team discuss about the of the previous Sprint like how it went,any positive done,any negative done,any improvements etc and establishes improvements for the next Sprint.

Difference between V and Agile model

V model	Agile model
Customer	Stakeholder
Business Analyst	Product Owner
Business Requirement Specification	Product Backlog
Software Requirement Specification	Sprint Backlog
Functional Requirement Specification	User Stories
Release	Sprint

Advantages of Agile model:

- High amount of visibility to the customer.
- Customers, developers and testers constantly interact with each other.
- Working software is delivered frequently (weeks rather than months).
- Face-to-face conversation is the best form of communication.
- Close daily cooperation between business people and developers.
- Continuous attention to technical excellence and good design.
- Regular adaptation to changing circumstances.
- Even late changes in requirements are welcomed.

Disadvantages of Agile model:

- In case of some software deliverables, especially the large ones, it is difficult to assess the estimations for requirement at the beginning of the software development life cycle.

- lack of documentation.
- The project can easily get taken off track if the customer representative is not clear about the requirement.
- Maximum times senior programmers are capable of taking the kind of decisions required during the development process

When to use Agile Model:

- When new changes are needed to be implemented.
- Both system developers and stakeholders alike, find they also get more freedom of time and options than if the software was developed

Popular Frameworks of Agile Methods/types of agile methodology:

- Scrum
- eXtreme Programming (XP)
- Dynamic Systems Development Method (DDSM)
- Feature Driven Development (FDD)
- Adaptive Software Development (ASD)
- The Crystal Method
- Lean Software Development (LSD)
- Disciplined Agile (DA)
- Scaled Agile Framework (SAFe)
- Rapid Application Development (RAD)