**SDLC:**

1. **Waterfall** model (**systems development life cycle**): also called sequential model and
   1. Best for a small project
   2. Best for the project, which all requirement & concept is clear
   3. Easy & fast for development
   4. **Process/Methods**: Req. Analysis, Design, Implementation, Testing, Deployment, Maintenance,
   5. Callengage: Add new feacture, Transparent, Ontime Tracking, Changes in between
2. **Agile =>**

[**https://agilemanifesto.org/principles.html**](https://agilemanifesto.org/principles.html)

[**https://blog.hubspot.com/service/scrum-master-product-owner**](https://blog.hubspot.com/service/scrum-master-product-owner)

[**https://www.atlassian.com/agile/scrum/sprint-planning**](https://www.atlassian.com/agile/scrum/sprint-planning)

[**https://www.atlassian.com/agile/scrum/scrum-master**](https://www.atlassian.com/agile/scrum/scrum-master)

[**https://www.youtube.com/watch?v=KL2AAWTdwB8&t=6s&ab\_channel=AmitTechnologies**](https://www.youtube.com/watch?v=KL2AAWTdwB8&t=6s&ab_channel=AmitTechnologies)

**The Agile methodology:** is a way to manage a project by breaking it up into several phases. It involves constant collaboration with stakeholders and continuous improvement at every stage. Once the work begins, teams cycle through a process of planning, executing, and evaluating.

**Agile** is a software development methodologies

**Agile** is an iterative approach to project management and software development that helps teams to deliver valuable & faster product to the customers.

## **Agile** **Methodology:** Scrum & kanban

**Agile Principle / Manifesto**

1. Customer satisfaction
2. Welcome changes
3. Deliver Frequently
4. Working together
5. Motivated team
6. Face to face
7. Working software
8. Constant pace
9. Good design
10. Simplicity
11. Self-organization
12. Reflect & Adjustment

## **Jira**

Jira Software is an agile project management tool that supports any agile methodology, be it scrum, kanban, or your own unique flavor. From agile boards, backlogs, roadmaps, reports, to integrations and add-ons you can plan track, and manage all your agile software development projects from a single tool. It's a software development tool used by an agile team.

**Feature of Agile**

1. **Plan:**  Create user stories and issues, plan sprints, and distribute tasks across your software team.
2. **Track:** Prioritize and discuss your team’s work in full context with complete visibility.
3. **Release:** Ship with confidence and sanity knowing the information you have is always up-to-date.
4. **Report:** Improve team performance based on real-time, visual data that your team can put to use.
5. k...
6. **Scrum:**

**Scrum is a framework** that helps teams work together.

**Import Point/Step/Process:** Product Backlog, EPIC, Version, Sprint, Sprint Planing, Sprint Backlog, Active sprint, Sprint Reviee, Retrospective, story, task/subtask, bug/defacut, ToDo list, InProcess, Testing, Done, Complete, Duration. scrum board, Scrum master

The key **difference between Agile and Scrum** is that while **Agile** is a project management philosophy that utilizes a core set of values or principles, **Scrum** is a specific **Agile** methodology that is used to facilitate a project.

* 1. Best for a larger product
  2. Best for project, where changes are frequently changes
  3. Product is developed in incremental/phase base
  4. **Process:** 
     1. Product Backlog, Sprints(Sprints planning & sprite backlog(current sprint/task ongoing)), Scrum Team, Increment version, Review.
     2. **Product Backlog**: Numbers of requirement to do.

Product owner specifies/ mentions all requirements, which need to be done & achieved for product/software. A product backlog is a prioritized list of work for the development team that is derived from the roadmap and its requirements. The most important items are shown at the top of the product backlog so the team knows what to deliver first.

* + 1. **Sprints**: Product backlog brack into multiple sprint.

Sprints are at the very heart of scrum and agile methodologies, and getting sprints right will help your agile team ship better software with fewer headaches. Sprints make projects more manageable, allow teams to ship high-quality work faster and more frequently, and give them more flexibility to adapt to change. Sprint contains/collection of one or more **issues(**Story/Bug/Task**)**.

* + 1. **Sprints Planing:** Decide which/when/how to do particular sprint.

The purpose of sprint planning is to define what/who/how/when can be delivered in the sprint and how that work will be achieved. Sprint planning is done in collaboration with the whole scrum team.

* + 1. **Task/Subtask:** Sprint brack into multiple Task & Task brack into multiple subtask.
    2. **Sprints Review:** Demostract the work with team & ask feedback.

A sprint review is about demonstrating the hard work of the entire team. Team members gather around a desk for informal demos and describe the work they’ve done for that iteration. It’s a time to ask questions, try new features, and give feedback. Sharing in success is an important part of building an agile team.

* + 1. **Self Review:**
    2. **Defect/Bug:**
    3. **Standups:** For software teams, a stand-up is like a sports team’s huddle(gathering सभा). It will be on a daily basis for approx 1 5mintus discussion about, what was your work status yesterday, which type of issue/task is blocking you.
    4. **Increment:** Sprint/task wise build called **potentially shippable product.**
  1. **Product Owners:**  manage the **Product Backlog(**All Requirement**), Release Management(**when build will release**), Stakeholder management(**users, customers, governance, and organizational leadership**),** and ensure the company gains maximum value from the product. Product Owners plan and prioritize work for the product's Scrum teams. But, to do this, they need to work behind the scenes with internal and external stakeholders to create a perfect product roadmap. This involves interviewing customers, reviewing product feedback, analyzing market trends, and working with upper management to approve a product vision.

**Product Owner(Role & Not owner of the product):** PO having good insight on customer and Manage StackHolder, Communicate with stackHolder, Manage Product Backlog, Manage Release. they realing knowing what the product should be doing & communicate with different product stakeholder to understand the requirement and thay bring the requirement to development team.

Here are some responsibilities for Product Owners.

* + 1. Create and maintain the product backlog.
    2. Work with the Product Manager to create a product vision and roadmap.
    3. Collaborate with the Scrum Master to ensure the product's development aligns with its original vision.
    4. Ensure the product backlog is updated and available to the entire development team.
    5. Work across departments and prioritize tasks for the Scrum Master based on stakeholder needs.
    6. Evaluate progress throughout the development process.
  1. **Scrum Masters:** are responsible for implementing the Scrum approach with the development and engineering teams. This makes their job description fairly straightforward. Here's an example we pulled from Southern New Hampshire University. A Scrum Master leads the Agile development team and supports the Product Owner by relaying updates to relevant employees. If the roadmap or backlog changes(by-product owner) along the way, the Scrum Master will notify the development team about the update. Scum Masters must possess in-depth knowledge of the Agile Methodology and be capable of communicating its best practices to the rest of the team. This requires them to be excellent communicators and gifted mentors. Scrum Master must justify the task and explain why it's relevant to the product's vision. They also need leadership skills so they can set attainable goals for their employees. This will keep people motivated and prevent teams from taking shortcuts during the development process.

Below are some of the common responsibilities for the Scrum Master position.

* + 1. Plan and execute the Agile Methodology with the Scrum development team.
    2. Monitor the sprint's progress and remove roadblocks impeding the product's development.
    3. Work with the Product Owner to make sure the product backlog is up to date.
    4. Communicate changes in the product backlog to the development team.
    5. Motivate the development team to complete tasks on time.
    6. Report on the success of the sprint.
    7. **Transparent:** must be transparent with Product Owner and development teams. To effectively inspect and adapt it is important that the right people can see what is going on. The scrum master is tasked with ensuring that the scrum team works in a transparent way
    8. **Bord Adminstator:** Work as the administrator of the scrum board. Ensure that cards are up to date and the scrum tool, Jira software is working well.
    9. **Iteration/sprint planning meetings:** Sprint planning is an event in scrum that kicks off the sprint. The purpose of sprint planning is to define what can be delivered in the sprint and how that work will be achieved. Sprint planning is done in collaboration with the whole scrum team. the sprint planning meeting for a two-week sprint would be no longer **than two hours**.
    10. **Daily standup:** Stand-up is a daily meeting that involves the core team: product owners, developers, and the scrum master. It will be **15 to 15 minutes** discussion and meeting discussion involves like
        1. What did I work on yesterday?
        2. What am I working on today?
        3. What issues are blocking me?
    11. **Sprint Review:** Team members gather around a desk for informal demos and describe the work they’ve done for that iteration. It’s a time to ask questions, try new features, and give feedback. Sharing in success is an important part of building an agile team.
    12. **Retrospective:** Ask the feedback/improvment with teams
    13. **1 To 1 Meeting:**
    14. **Reporting**
    15. **Blocker**
    16. **Team Motivation**
    17. **Coach when required & Work as protector of team**
  1. **Team(Development Team):** Including all types of people including designers, Writers, Programmers, etc. team will self-organize so they can make decisions to get work done.

The development team’s responsibilities include:

1. Delivering the work through the sprint.
2. To ensure transparency during the sprint they meet daily at the daily scrum ( sometimes called a standup).
3. **Kanban:** Kanban is a popular framework used to implement agile and DevOps software development. It requires real-time communication of capacity and full transparency of work. Work items are represented visually on a kanban board, allowing team members to see the state of every piece of work at any time.

Kanban is a visual method for [managing workflow](https://www.planview.com/products-solutions/products/leankit/) at the individual, team, and even organizational level. Implementing software increment on Kanban Method is a pull based system, that help the team to Continue the delivery in a sustainable pace, with in capacity. Its reduce waste of efforts and time. To Maintain this, It needs to follow the basic principles of Kanban as below.

**1. Visualize Work:** Visual model of Kanban Board of work and its workflow make the scope and capaicty transparent, it helps to observe and Inspect the flow of work moving backlog to done. This makes the work visible—along with blockers, bottlenecks and queues and upcoming work, Which helps the team to make strategy of working on exiting work or bring new work to in Progress.

**2. Limit Work in Progress:** The Team mutually defines a Limit for all “work in progress” Columns in Kanban Board, such as Analysis, Development, Testing etc. This WIP limit implements the Pull based system, as work can be pulled to the current column from previous column only if the total number of work under the column is less than its limit.

This helps balance the flow-based approach so teams don’t start and commit to too much work at once. Its reduce waste and help the team to focusing on finishing first and starting later.

**3. Focus on flow of Work:** To Complete a work, and add a value it has to pass through multiple stages of its development phase. Like Analysis, Development, Testing , Review etc. To get the effective benefit of Kanban the Team needs to focus on flow of work from its initiation to completion. By following above 2 principles helps achieve focus on flow.

Focus on the workflow leads the team to visualize upcoming bottlenecks to act on. so that the flow remains. Team frequently makes strategy of working on in progress wor item to optimize the flow

1. **Kanban Bord:** Imagine a whiteboard, divided into vertical lanes. Each lane represents a step in your process, from **“To Do,” to “Doing,” to “Done.”** A kanban board is an agile project management tool designed to help visualize work, limit work-in-progress, and maximize efficiency (or flow). Kanban boards use cards, columns, and continuous improvement to help technology and service teams commit to the right amount of work, and get it done! kanban boards can be broken down into five components
2. **Visual Signals** — One of the first things you’ll notice about a kanban board are the visual cards (stickies, tickets, or otherwise). Kanban teams write all of their projects and work items onto cards, usually one per card. For agile teams, each card could encapsulate one [user story](https://www.atlassian.com/agile/project-management/user-stories). Once on the board, these visual signals help teammates and stakeholders quickly understand what the team is working on.
3. **Columns** — Another hallmark of the kanban board are the columns. Each column represents a specific activity that together compose a “workflow”. Cards flow through the workflow until completion. [Workflows](https://www.atlassian.com/agile/project-management/workflow) can be as simple as “To Do,” “In Progress,” “Complete,” or much more complex.
4. **Work In Progress (WIP) Limits** — WIP limits are the maximum number of cards(Analysis, InProgrss, Testing, Reviewing) that can be in one column at any given time. Like A column with a WIP limit of three cannot have more than three cards in it.
5. **Commitment point** — Kanban teams often have a backlog for their board. This is where customers and teammates put ideas for projects that the team can pick up when they are ready. The commitment point is the moment when an idea is picked up by the team and work starts on the project.
6. **Delivery point** — The delivery point is the end of a kanban team’s workflow. For most teams, the delivery point is when the product or service is in the hands of the customer. The team’s goal is to take cards from the commitment point to the delivery point as fast as possible.
7. **kanban card:** The kanban card is an essential component of kanban, a work management framework which helps you visualize your work, limit work in progress (WIP), and maximize efficiency (or flow). Each kanban card represents a single work item as it moves through various stages of completion which are represented on either a physical or virtual kanban board.
8. Kanban Plan: Kanplan is a mixed methodology for practicing agile software development. Like scrumban, it combines features from both scrum and kanban. Kanplan is ideal for teams who want the ability to backlog groom, but don’t want to work in sprints