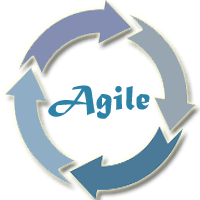
**AGILE**

Agile is an iterative approach of software development methodology using short iterations of 1 to 4 weeks. Using Agile methodology, the software is distributed with fastest and fewer changes. The advantages of agile methodology are customer satisfaction by rapid, continuous development and delivery of useful software.



## Prerequisite

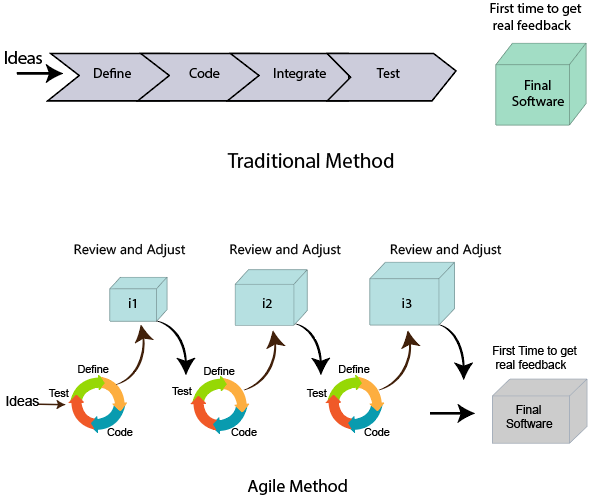
Before learning agile methodology, you should have a basic understanding of software engineering or software development life cycle.

# What is Agile Methodology?

An agile methodology is an iterative approach to software development. Each iteration of agile methodology takes a short time interval of 1 to 4 weeks. The agile development process is aligned to deliver the changing business requirement. It distributes the software with faster and fewer changes.

The single-phase software development takes 6 to 18 months. In single-phase development, all the requirement gathering and risks management factors are predicted initially.

The agile software development process frequently takes the feedback of workable product. The workable product is delivered within 1 to 4 weeks of iteration.



## Roles in Agile

There are two different roles in a Agile methodology. These are the Scrum Master and Product Owner.

### 1. Scrum Master

The Scrum Master is a team leader and facility provider who helps the team member to follow agile practices, so that the team member meets their commitments and customers requirements. The scrum master plays the following responsibilities:

* They enable the close co-operation between all the roles and functions.
* They remove all the blocks which occur.
* They safeguard the team from any disturbances.
* They work with the organization to track the progress and processes of the company.
* They ensure that Agile Inspect & Adapt processes are leveraged correctly which includes
  + Planned meetings
  + Daily stand-ups
  + Demo
  + Review
  + Retrospective meetings, and
  + Facilitate team meetings and decision-making process.

### 2. Product Owner

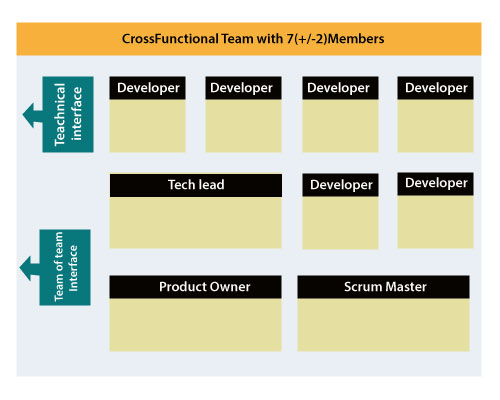
The Product Owner is one who runs the product from a business perspective. The Product Owner plays the following responsibilities:

* He defines the requirements and prioritizes their values.
* He sets the release date and contents.
* He takes an active role in iteration and releasing planning meetings.
* He ensures that the team is working on the most valued requirement.
* He represents the voice of the customer.
* He accepts the user stories that meet the definition of done and defined acceptance criteria.

## Cross-functional team

Every agile team contains self-sufficient team with 5 to 9 team members. The average experience of each member ranges from 6 to 10 years. The agile team contains 3 to 4 developers, 1 tester, 1 technical lead, 1 scrum master and 1 product owner.

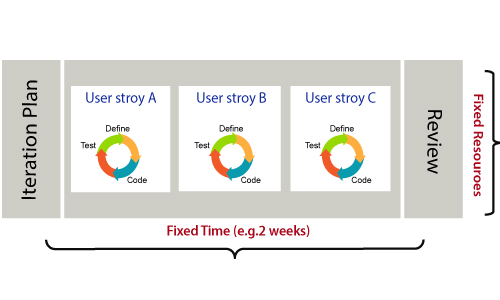
The Scrum master and Product owner are considered as a part of Team Interface, on the other hand remaining members are the part of Technical Interface.



## How an Agile Team plan their work?

An Agile methodology is not a specific set of ceremonies or specific development techniques. Rather, it is a group of methodologies that demonstrate a commitment to tight feedback cycles and continuous improvement. An Agile team works in iterations to deliver the customer requirement, and each iteration takes 10 to 15 days. However, the original Agile Manifesto didn't set the time period of two-week iterations or an ideal team size.

Each user requirement is a planned based and their backlog prioritization and size. The team decides, how much scope they have and how many hours available with each team to their planed task perform.



## What is a user requirement?

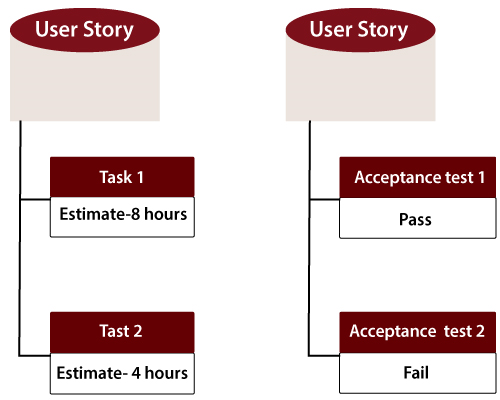
The user requirement defines the requirements of the user in terms of functionalities. There may be of two type of functionality.

* As a <User Role> I want <Functionality> so that <Business Value>
* In order to <Business value> as a <User Role> I want <Functionality>.

During software release planning, a rough estimate is given to a user requirement using relative scale points. During iteration planning, the requirement is broken down into tasks.

### Relation between User requirement and Task

* User requirement talks about what is to be done. It defines the needs of users.
* Task talks about how it is to be done. It defines how functionality is implemented.
* User requirements are implemented by tasks. Every requirement is gathering as the task.
* User requirement is divided into different tasks when it is planned in current iteration.
* User tasks are estimated in hours based, generally it is between 2 to 12 hours.
* Requirements are validated using acceptance test.



## When the requirement is completed

The Agile team decides the meaning of task done. There may be different criteria for it:

* When the entire task (development, testing) are completed.
* When all the acceptance tests are running and are passed.
* When no defects found.
* Product owner has accepted the requirement.
* When the software product is delivered to the end user.

## What is Software Acceptance Criteria?

Acceptance Criteria is defined as the functionality, behavior, and performance required by a product owner. It defines what is to be done so that the developer knows when a user requirement is complete.

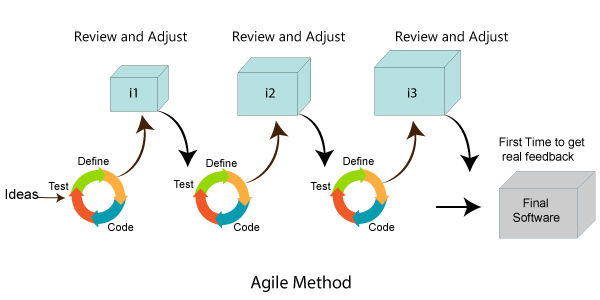
# 

# Advantage of Agile Methodology

There are various advantages of using agile methodology over traditional waterfall model or others. Agile development methodology and testing practices have worked wonders for numerous organizations with positive aspects. Its positive aspects are not hidden, it is very much visible in the organization.

## Advantages of Agile Methodology

1. Customer satisfaction is rapid, continuous development and delivery of useful software.
2. Customer, Developer, and Product Owner interact regularly to emphasize rather than processes and tools.
3. Product is developed fast and frequently delivered (weeks rather than months.)
4. A face-to-face conversation is the best form of communication.
5. It continuously gave attention to technical excellence and good design.
6. Daily and close cooperation between business people and developers.
7. Regular adaptation to changing circumstances.
8. Even late changes in requirements are welcomed.

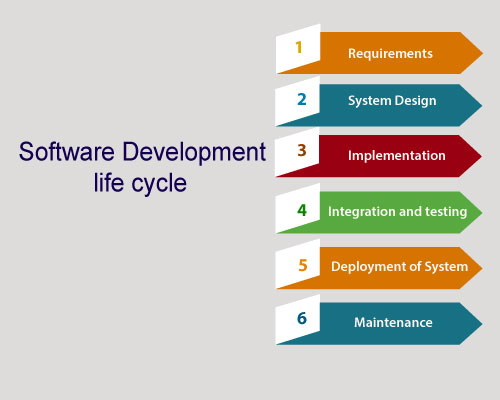


## Disadvantages of Agile methodology:

1. It is not useful for small development projects.
2. There is a lack of intensity on necessary designing and documentation.
3. It requires an expert project member to take crucial decisions in the meeting.
4. Cost of Agile development methodology is slightly more as compared to other development methodology.
5. The project can quickly go out off track if the project manager is not clear about requirements and what outcome he/she wants.

## Advantages of the Waterfall Model:

1. It is one of the easiest and traditional model to manage. Because of its traditional development nature, each phase has specific deliverables and a review process.
2. It works well in smaller size projects where requirements are easily understandable.
3. It has a faster product delivery model.
4. There are well-documented process and results.
5. Easily adaptable method for shifting teams
6. This project management methodology is beneficial to manage dependencies.



## Disadvantages of Waterfall Model:

1. It is not an ideal model to develop a large scale project size.
2. It requires a clear-cut requirement at the beginning time; otherwise, it may lead to a less effective method.
3. It is difficult to move back to make changes in the previous phase.
4. The testing process starts once development is completed. Hence, it has high chances of bugs to be found later in project development. Due to this, it is costly to fix.

## Compression between the Agile methodology and Waterfall model:

|  |  |
| --- | --- |
| **Agile methodology** | **Waterfall model** |
| It follows the incremental approach. | It is a sequential design process. |
| It divides the project development lifecycle into a sprint. | The software development process is divided into distinct phases. |
| Agile methodology is a flexible methodology. | The Waterfall is a structured software development methodology. |
| Agile is the collection of many different projects. | It is completed as one single project. |
| The test plan is reviewed after each sprint | Test plan is reviewed after complete development. |
| Testing team can take part in the requirements change phase without problems. | It is difficult for the test to initiate any change in needs. |

# 

# 

# Agile Manifesto

In February 2001, at the Snowbird resort in Utah, a team of 17 software developers met to discuss lightweight development methods. The result of their meeting was the following Agile Manifesto for software development:-

We are uncovering the better ways of developing software by doing it and helping others to do it. Through this meeting, we have come to value -

* Individuals and interactions over Processes and tools.
* Working software over comprehensive documentation.
* Customers are collaboration over contact negotiation.
* Responding to change over following a plan.

So that, while there is value in the items on the right, we value the items on the left more.

## The Twelve Principle of Agile Manifesto

**1.Customer Satisfaction:**

Manifesto provides high priority to satisfy the costumer's requirements. This is done through early and continuous delivery of valuable software.

**2.Welcome Change:**

Making changes during software development is common and inevitable. Every changing requirement should be welcome, even in the late development phase. Agile process works to increase the customers' competitive advantage.

**3.Deliver the Working Software:**

Deliver the working software frequently, ranging from a few weeks to a few months with considering the shortest time period.

**4.Collaboration:**

Business people (Scrum Master and Project Owner) and developers must work together during the entire life of a project development phase.

**5.Motivation:**

 Projects should be build around motivated team members. Provide such environment that supports individual team members and trust them. It makes them feel responsible for getting the job done thoroughly.

**6.Face-to-face Conversation:**

 Face-to-face conversation between Scrum Master and development team and between the Scrum Master and customers for the most efficient and effective method of conveying information to and within a development team.

**7.Measure the Progress as per the Working Software:**

 The working software is the key and primary measure of the progress.

**8.Maintain Constant Pace:**

The aim of agile development is sustainable development. All the businesses and users should be able to maintain a constant pace with the project.

**9.Monitoring:** Pay regular attention to technical excellence and good design to maximize agility.

**10.Simplicity:**

Keep things simple and use simple terms to measure the work that is not completed.

**11.Self-organized Teams:**

 The Agile team should be self-organized. They should not be depending heavily on other teams because the best architectures, requirements, and designs emerge from self-organized teams.

**12.Review the Work Regularly:**

The work should be reviewed at regular intervals, so that the team can reflect on how to become more productive and adjust its behaviour accordingly.

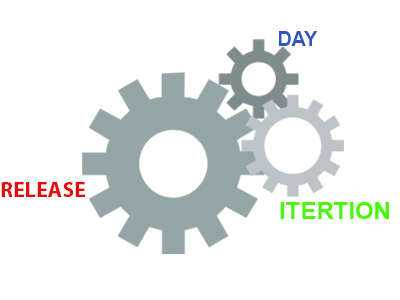
# 

# Agile Characteristics

The product developed under agile methodology has seen several important characteristics that are given below.

### Agile Development Releases and Fixed-Length Iterations

The agile software development method is based on two central units of delivery: release and iteration. A single version consists of several iterations phase. Each iteration consists of its micro-project. The different functions of agile development like defects, enhancement requests and other work items are organized, estimated, and prioritized, and then assigned to release.



### Agile Development Delivers-Working, Tested Software

The primary measure of the agile development team is to deliver working, progress and tested feature software. Working features serve as the basis for enabling and improving customer feedback. It also serve as team collaboration, and overall project visibility. They provide such evidence so that both the system and the project are on track.

At every step of product development, the team continuously works to assemble on the best business solution. This is done using the latest input from users, customers, and other stakeholders.

### Value-Driven Development

Agile development methodology focuses really on delivering business value early and continuously. It is measured by running tested software. The development team focuses on product features as the central unit of planning, tracking, and delivery.

As the development goes on from iteration to iteration, the team tracks how many product are running, tested features they are delivering.

### Continuous (Adaptive) Planning

As the project launches, the development team does just more planning to get going with the initial iteration and, if it is appropriate, to lay out a high-level release plan of features. The single iteration leads the key to continuous planning.

As the iteration starts, the team choose a set of features to implement, determines and estimates each technical task for each feature.

### Multi-Level Planning in Agile Development

The continuous planning impacts much more significant result if it occurs on at least two levels:

* At the release level, the development team identifies and prioritize the features they must have, would like to have, and they can do within the deadline.
* At the iteration level, development team picks and plan for the next batch of features to implement, in priority order. If the product features are too large to estimated or delivered within a single iteration, the development team break them down further.

### Relative Estimation

Several agile development teams use the practice of relative estimation for features to accelerate planning. It removes unnecessary complexity. The development team selects a few (3-5) relative estimation categories, or buckets, and estimates all features in terms of these categories.

The concept of relative estimation or/and predefined estimation buckets that prevent the team from wasting time on debating. When the product feature exceeds an agreed maximum estimate, then it should be further broken down into multiple features.

### Emergent Feature Discovery

As disputed to spending weeks or months, analyzing the requirements before initiating development, agile development projects quickly prioritized and estimated features, and then refine the details when required. The feature of the product is described in more detail between customers, testers, and developers working together.

### Continuous Testing

Using continuous testing of software product, we determine the progress and prevent defects. We handle the running and tested features. Using continuous testing, we can reduce the failure risk in the project.

### Continuous Improvement

Continuous testing and constant improvement are correlated with each other. While continuous testing, if we found any bugs or project failure, we continuously improve that bugs immediately. We continuously refine both the project and the system.

### Small, Cross-functional Teams

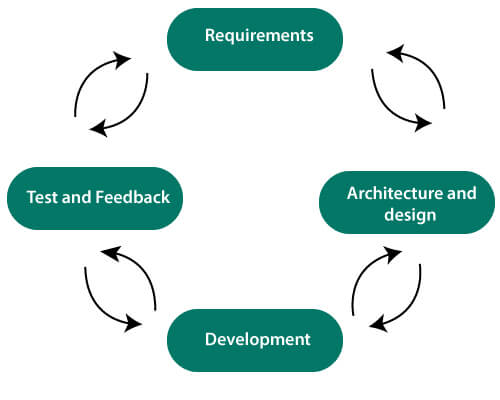
The incremental software product is delivered at every iteration. The development teams must also be cross-functional to be successful in developing the valuable software.

# 

# Agile Software Development Life Cycle (SDLC)

**Software development life cycle (SDLC)** is a phenomenon to **design**, **develop** and, **test** high-quality software. The primary aim of SDLC is to produce high-quality software that fulfills the customer requirement within times and cost estimates.

**Agile Software Development Life Cycle (SDLC)** is the combination of both iterative and incremental process models. It focuses on process adaptability and customer satisfaction by rapid delivery of working software product. Agile SDLC breaks down the product into small incremental builds. These builds are provided into iterations.



In the agile SDLC development process, the customer is able to see the result and understand whether he/she is satisfied with it or not. This is one of the advantages of the agile SDLC model. One of its disadvantages is the absence of defined requirements so, it is difficult to estimate the resources and development cost.

**Each iteration of agile SDLC consists of cross-functional teams working on various phases:**

1. Requirement gathering and analysis
2. Design the requirements
3. Construction/ iteration
4. Deployment
5. Testing
6. Feedback

### Requirements gathering and analysis

In this phase, you must define the requirements. You should explain business opportunities and plan the time and effort needed to build the project. Based on this information, you can evaluate technical and economic feasibility.

### Design the requirements

When you have identified the project, work with stakeholders to define requirements. You can use the user flow diagram or the high-level UML diagram to show the work of new features and show how it will apply to your existing system.

### Construction/ Iteration

When the team defines the requirements, the work begins. The designers and developers start working on their project. The aims of designers and developers deploy the working product within the estimated time. The product will go into various stages of improvement, so it includes simple, minimal functionality.

### Deployment

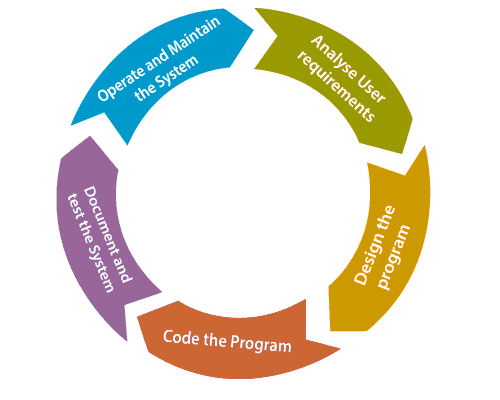
In this phase, the team issues a product for the user's work environment.

### Testing

In this phase, the Quality Assurance team examine the product's performance and look for the bug.

### Feedback

After releasing of the product, the last step is to feedback it. In this step, the team receives feedback about the product and works through the feedback.



## Agile SDLC Process Flow

1. **Concept:** Project are imagined and prioritized.
2. **Inception:** Team members are created, funding is put in place, and basic environments and requirements are discussed.
3. **Iteration/Constriction:** The software development team works to deliver working software. It is based on requirement and feedback.
4. **Release:** Perform quality assurance (QA) testing, provides internal and external training, documentation development, and final version of iteration into the product.
5. **Production:** It is ongoing support of the software.

## Advantages of Agile SDLC

1. Project is divided into short and transparent iterations.
2. It has a flexible change process.
3. It minimizes the risk of software development.
4. Quick release of the first product version.
5. The correctness of functional requirement is implemented into the development process.
6. Customer can see the result and understand whether he/she is satisfied with it or not.

## Disadvantages of Agile SDLC

1. The development team should be highly professional and client-oriented.
2. New requirement may be a conflict with the existing architecture.
3. With further correction and change, there may be chances that the project will cross the expected time.
4. There may be difficult to estimate the final coast of the project due to constant iteration.
5. A defined requirement is absent.

# 

# 

# Agile Project Management

Agile project management is an interactive approach to manage software development. The agile project management focuses on continuous releases and covers customer feedback with every iteration.

Traditionally the agile project management is classified into two frameworks: **scrum** and **kanban**. The [scrum framework](https://www.javatpoint.com/agile-scrum) focused fixed-length project iterations, whereas [kanban framework](https://www.javatpoint.com/agile-kanban) focused on continuous releases. After competition of project first iteration (or steps) project management activity immediately moves on to the next.

## History of Agile Project Management

## Agile project management is rapidly rising in the 21st century. It is used for software development projects and other IT initiatives.

However, from the mid-20th century, the concept of continuous development has taken various forms. For example, there was James Martin's **Rapid Iterative Production Prototyping (RIPP)**, an approach that served as the premise for the 1991 book **Rapid Application Development (RAD)**.

The agile project management framework which has emerged in most recent years is known as Scrum. This methodology features works on the development team to create a product backlog. It also creates a prioritized list of the features, functionalities, and fixes required to deliver a successful software system. The scrum team offers the pieces of a task in rapid increments.

## How Agile Project Management works

The agile project management calls for teams to regularly evaluate cost and time as they move through their work. They use velocity, burnup and burndown charts to measure their work, rather than Gantt charts and project milestones to track progress.

The agile team practices to continuous development and continuous integration using technology that automates steps to speed up the release and use of products.

The presence and participation of the project manager are not required in agile project management. Although the presence of the project manager is essential for success under the traditional (waterfall model) project delivery. The role of the project manager is to distribute task among team members. However, the project manager is not obsolete in agile project management, and many organizations use them in a large, more complex project. The organization mostly places them in the project coordinator role.

Agile Project Management demands that team members know how to work in this new agile methodology. The team member must be able to coordinate with each other, as well as with users.

# SCRUM

# Related image

**Scrum is a framework** that helps agile teams to work together. Using it, the team members can deliver and sustain the complex product. It encourages the team to learn through practice, self-organize while working on the problem. Scum is a work done through the framework and continuously shipping values to customers.

It is the most frequent software that is used by the development team. Its principle and lessons can be applied to all kinds of teamwork. Its policy and experiences is a reason of popularity of Scrum framework. The Scrum describes a set of tools, meetings, and roles that help the teams structure. It also manages the work done by the team

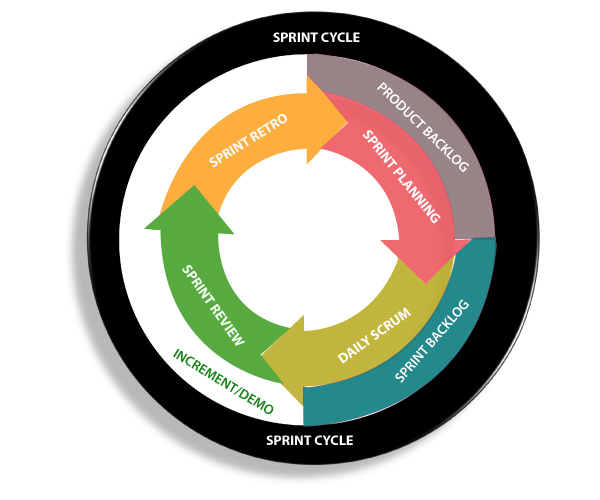
## The framework

Scrum and agile are not the same thing because Scrum focused on continuous improvement, which is a core foundation of agile. Scrum framework focuses on ongoing getting work done.

## What are sprints?

With scrum, a product is built in a series of repetition called **sprints**. It breaks down big complex projects into bite-size pieces. It makes projects more manageable, allows teams to ship high quality, work faster, and more frequently. The sprints give them more flexibility to adapt to the changes.

Sprints are a short, time-boxed period for Scrum team that works to complete a set amount of work. Sprints are the core component of Scrum and agile methodology. The right sprints will help our agile team to ship better software.



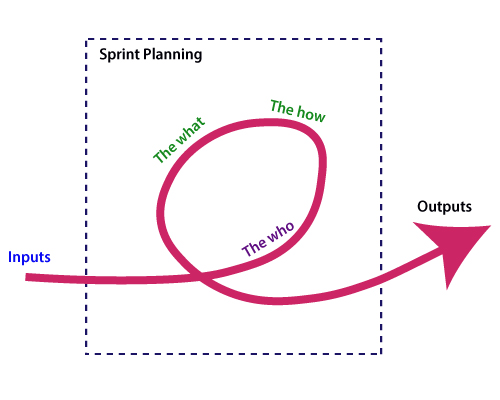
**What is sprint plan?**

Sprint plan is an action in Scrum that kicks off the sprint. The primary purpose of sprint plan is to define what can deliver in the sprint. It also focuses on how the work will be achieved. It is done in combination with the whole Scrum team members.

The sprint is a set of the period where all the work to be done. Before we start the development, we have to set up the sprint. We need to describe how long time is required to achieve the sprint goal and where we are going to start.

## Factors affecting Sprint planning

* **The What:** The product owner describes the goal of the sprint and the backlog items which contribute to achieve that goal.
* **The How:** Agile development team plans its necessary work on how to achieve and deliver the sprint goal.
* **The Who:** The product owner defines the goal based on the value that the customers seek. And the developer needs to understand how they can or cannot deliver that goal.
* **The Inputs:** The product backlog provides the list of input stuff that could potentially be part of the current sprint. The team looks over the existing work done in incremental ways.
* **The Outputs:** The critical outcome of sprint planning is to meet described team goal. The product set the goal of sprint and how they will start working towards the goal.



**What is the product backlog?**

A product backlog is a registered list of work for the development team. It is driven from the roadmap and its requirements. The essential task is represented at the top of the product backlog so that the team member knows what to deliver first. The developer team doesn't work through the backlog from the product owner's side and product owner doesn't push the work to the developer team. The developer team pulls work from the product backlog.

## Backlog starts with the two "R"s

The fundamental product backlog is provided by a team's **roadmap** and **requirements**. Roadmap repetition breaks down into several epics, and each epic will have several requirements and user stories.

The product owner organizes each of the customer stories into a single list. This story is organized for the development team. The product owner chooses to deliver first complete epic.

## The factors that influence a product owner's prioritization

* Priority of customer
* Importance of getting feedback
* Relative implementation difficulty
* Symbiotic relationships between work items

**KANBAN**

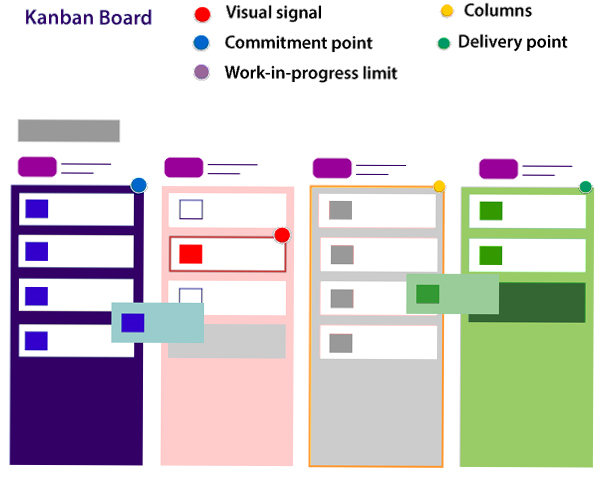
Kanban is a popular framework which is used to implement agile software development. It takes real time communication of capacity and complete transparency of work. The work items are represented in a kanban board visually, allowing team members to see the state of every piece of work at any time.

## Boards

The kanban board is the agile project management tool that designed the necessary visualized work, limited work-in-progress, and maximizes flow (or efficiency). It uses cards, columns, and provides continuous improvement to help technology and service teams who commit the right amount of work and get it done.

### Elements of a kanban board

A person called David Anderson divides the kanban board into five different components. These are Visual signals, columns, work-in-progress limits, a commitment point, and a delivery point.



**1.Visual Signals:** The kanban board is a visual card (stickies, tickets, or otherwise). Kanban team write their projects and work items onto cards, usually per person each card. For agile teams, each card could encapsulate into one user story. Once the board completed, this visual team helps team members and stock members quickly to understand what the team is working.

**2.Columns:** The column represents the specific activities that compose a "workflow" together. The card flows through a workflow until its completion. The workflow may be a simple as "To Do," "In Progress," "Complete," or much more complicated.

**3.Work in progress (WIP) Limits:** The work in progress limits are the maximum number of cards which can be in one column. This is at any given time. It gives the alert signal that you committed too much work.

**4.Commitment point:** Kanban teams also maintain a backlog for their board. This is where the customers and team member put ideas for projects that the team can pick up. The team members pick up plans when they are ready. The committed point is a movement where the design is picked up by the team, and work starts on the project.

5.**Delivery point:** It is the end point of a kanban team's workflow. Mostly the delivery point for every team is when the product and services are handed to the customer.

## Kanban vs Scrum board

The following are the differences between Kanban and Scrum board:

|  |  |
| --- | --- |
| **Kanban** | **Scrum** |
| Kanban is an ongoing process. | Scrum sprints have a start and stop dates |
| Kanban has no formal roles. | Role is clearly defined of each team in the scrum (product owner, development team, and scrum master). Both teams are self-organized. |
| A kanban board is used throughout the lifecycle of a project | Scrum board is cleared and recycled after each sprint. |
| This board is more flexible with regards to tasks and timing. Its task can be reprioritized, reassigned, or updated as needed. | This board has the number of tasks and a strict deadline to complete them. |

## Kanban

Kanban is the combined methodology for implementing agile software development. Like scrumban, it is a combination of both scrum and kanban (mixed) methodology.

# Difference between Agile and Scrum (Agile vs Scrum)

[Agile](https://www.javatpoint.com/agile-methodology) is an iterative approach of software development methodology using short iterations of 1 to 4 weeks. Due to the agile methodology, the development process is aligned to deliver the changing business requirement. Using Agile methodology, the software is distributed with fasterand fewer changes.

[Scrum](https://www.javatpoint.com/agile-scrum) is a framework of agile that helps agile teams to work together. Using it, the team members development, deliver and sustain the complex product. It encourages the team to learn through practice, self-organize while working on the problem. Scum is a work done through the framework and continuously shipping values to customers.

## Agile

1. Agile is an **iterative and incremental approach** to software development methodology.
2. In this approach, the **leadership** plays an important role.
3. Agile software development is highly suitable for the **medium or large project**.
4. **Flexibility** is the most significant advantage of agile as it quickly reacts to changes.
5. Agile involves **face-to-face communication** and collaboration between the members of various cross-functional teams.
6. Agile development needs **frequent delivery** to the end user for their feedback.
7. In this development, each step like requirements, analysis, design, are **continually monitored** during the lifecycle.
8. The **project leader** takes cares of all the tasks in the agile method.
9. End-user may give their **feedback during the development** process. So, the end product will be more useful.
10. **Delivery and update** of the software are taking place regularly.
11. Design and execution should be kept **simple**.
12. The priority of agile development is always to satisfy the customer by providing **continuous delivery** of valuable software.
13. Working software is the most **fundamental measure** of progress.
14. It is best to have **face-to-face communication** to get as close to the project goal as possible.

## Scrum

1. Scrum is a framework of agile methodology. In which **incremental builds** are delivered to end user in every two to three weeks.
2. Scrum's team is **self-organized**, cross-functional team.
3. Scrum is used in the project where the requirement **rapidly** changes. v
4. A compared to agile it is more **rigid**. So that there are no chances of frequent change.
5. In **daily stand up meeting** the teamwork is achieved with a fixed role assigned to team members, scrum master, and product owner.
6. **No need to change many more** while implementing scrum process.
7. In this process, a **build is delivered** after each sprint to the client for their feedback.
8. After every sprint a demonstration of functionality is provided. So that the **regular feedback** can be taken before next sprint.
9. There is no team leader, so the **entire team handles the issues** or problems.
10. When the team completes the **current sprint activity**, then the next sprint is planned.
11. Design and execution can be **innovative and experimental**.
12. The **daily sprint meeting** is organized to review the feedback to decide the future progress of the project.
13. Working software is **not a fundamental measure**.
14. The target of the Scrum team is to deliver **maximum business value**.

# 

# Agile Daily Stand-up

Agile daily stand-up is termed as the day-to-day status meeting on the project of the members of the agile team. The daily meeting of the agile team discussed the forum for regular updates as well as the problems of team members. It focuses on addressing the issues and tries to solve the issues quickly. The daily stand-up is the regular practice, no matter how an agile team is established regardless of its office location.

## What is Daily Stand-up?

The daily stand-up is a daily status meeting of the agile team member. This meeting roughly takes 12 to 18 minutes (an average of 15 minutes).

Each member of the team has to answer three important questions

1. What he/she did yesterday?
2. What he/she will do today?
3. The problem he/she is facing . . . He/she blocked due to. . .

The daily stand-up is done for a day-to-day status update. The meeting of team members with the product owner can be scheduled at different time. The participants in the stand-up meetings only stand instead of sitting so that the meetings get finished quickly.

## Important of Stand-up:

The importance of having a daily stand-up in agile are as follows:

* The team can evaluate the progress report daily.
* The team member discusses all the progress and the commitments he/she made for the day.
* The members can also see whether they can deliver the project as per the iteration plan or not.
* Stand-up provides visibility to the team on any delay that occurs due to some obstacles.

## Who Attends a Stand-up?

* The project owner, scrum master, and the delivery team should attend the stand-up regularly.
* Customers and Stakeholders are encouraged to participate in the meeting, and they act as an observer. However, they are not supposed to participate in stand-ups.
* The responsibility of scrum master is to take note each team member's queries and the problems they are facing.

## Geographically Dispersed Teams

A stand-up meeting is done in different ways depends upon the working time zone.

* On the rotation basis, select a member who can take the stand-up meeting of teams located in different time zones.
* A separate team has a separate stand-up meeting.
* Daily update the status of the stand-up in a tool such as SharePoint, Rally, Wikis, etc.
* There are varieties of communication tools ready like video conferencing, instant messengers, conference call, and other knowledge sharing tools.

# 

# Agile Definition of Done

Agile Definition of **done** is defined into three different stages called User Story (Requirement), Iteration, and product Release. These are given below:

## User Story (requirement)

A user story is a requirement which is formulated into few sentences. The user requirement is the everyday language of user. This user story should be completed within iteration. The user story is done when

* All the related code and documentation have been checked-in.
* The product passed all the processes of unit test.
* All the processes of the acceptance test case have been moved.
* The product owner must have accepted the story.
* The help text (documentation) is written.

## Iteration

An iteration is a time-based collection of a user story. It works on the defected product and accepted within the release of a product. Iteration is defined at the time of the iteration planning meeting and completed within the iteration demo and review meeting. The iteration is also known as a sprint. The repetition is required when:

* Performance of the product has been tested.
* Product backup is complete.
* User requirement has been accepted or moved for the next iteration.
* Defected product has been fixed or postponed for the next iteration.

## Release

The product release is a major occasion that represents an internal and external delivery of work. It also tests the version of the product or system. The product release is done when:

* The system is stress tested.
* Performance is high.
* Contain the security validation in the product.
* Disaster recovery plan is tested.

# 

# Agile Design

Design plays a vital role in any software development process. The agile team also focuses on "what to do about design" because of the following four factors:

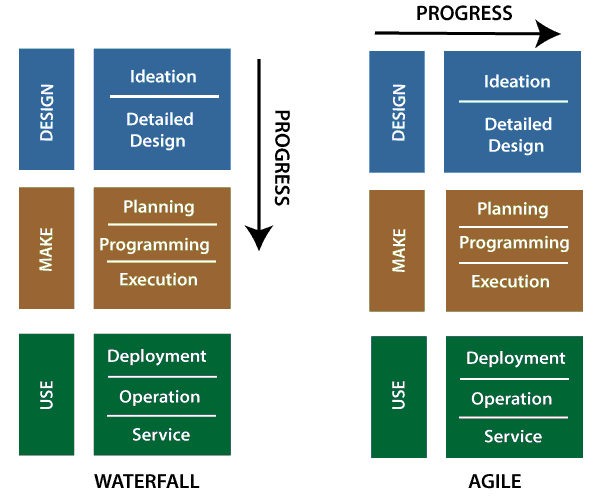
* Many crucial factors focus on loyal designs during the planning process. Design forces towards waterfall culture throughout product implementation.
* Designers also interact with a cross team for a limited time.
* Designers don't always have an easy way to report feedback to the engineering team.
* The presentation and logic layers are not still transparent. They are not separated clearly in the code base, making style changes difficult.

## The product design process and customer interview

The agile is divided into several methodologies and processes. These methodologies and processes keep the iterative and free-flowing nature of the technique at their core. The agile design and development methodology used especially in engineering development, and this process called Scrum.

Customer interviews can be an informative part of the project design phase. We will have several of those "light bulb" movements during interviews. It encourages the people who are interviewing with other members of the team (engineering, marketing, design, etc.)

There are several resources that are available on which we conduct an interview- the logistics, methods, and techniques.



## The customer interview pyramid:

Atlassian is a simple framework that helps in building the customer interview pyramid. This pyramid looks like as



**Communication Observation:** At the bottom of the pyramid, we will get the very minimum. We should all come back from an interview and be able to list observations as we don't need any experience to regulate what you've seen.

**Interpret problems:** Above the Communication Observation, it is an interpret problem. It is explaining the user's behavior and grouping them with an over-arching problem statement.

**Connecting opportunities:** This is the peak of the pyramid where the most value comes in combining the problem with potential opportunities or related patterns. This helps influence a roadmap and make decisions about what to tackle next.

# 

# Agile Software development

Agile development is more than a framework such as Kanban, Scrum, and Extreme Programming of Feature-Driven Development (FDD). It is more than practice, such as planning, test-driven development, planning sessions, stand-ups, and sprints.

Agile software development contains the set of frameworks, so it is called as an umbrella term. These frameworks are based on the values and principles expressed in [Agile manifesto.](https://www.javatpoint.com/agile-manifesto).

The things that separate agile from other approaches to software development are the focus on the people doing the work and how they work together. The agile software development communities focus on collaboration and the self-organizing team.

Mostly the team and organization start work doing on agile software development, and then they focus on practices that help with collaboration and organizing the work.

## How to Be an Awesome Agile Developer

The agile team developers focus on sustainable development ? not heroics. Software sustainability is good estimation, effective branching strategies for managing code. This code is executed by automated testing to protect the quality, and continuous deployment to get fast feedback from users. Agile development is a continuous deployment to get quick feedback from users.

The "**iron triangle**" is a project management system in which all the developers should know about the project scope, schedule, and quality development.

## Journey to a stress-free software release:

The success measure of an agile team is that when the working software product is released to the customer. But some time, it is found that the software teams feel the terrible experience at the time of validating the completed issues against artifacts. The code review might be missing. The complete code is not being merged, builds for merged code fail, etc.

### Factors that build success software release:

**Code best practice:** it will improve the ability to deliver a quality product. The code review is essential before providing the product, and monitoring and fixing the declining builds will assure faster time to release.

**Set up and maximize Jira Software's release hub:** The team focus on setup the Jira software's release hub. It saves the working hours by allowing the release hub to provide a clear picture of progress status and release.

**Automation from build code to release:** The complete automation from build code to releasing the version straight from release hub.

## Why code reviews matter:

Code review is an essential part of the software development before releasing to customer. It helps developers to learn the code base, as well as help them to learn a new technology that grows their skill sets.

**What is exactly a code review?** When the developer team finishes their working on an issue, other developers pay attention to the code and consider questions like:

* Is there any accessible, logical error in the code?
* Is there any module that takes an outside requirement, and all the cases are fully implemented?
* Is the new automated test sufficient for the new code? Is there any requirement to rewrite in the existing automated tests for changing the code?
* Is the new code conforming to current style guidelines?

# 

# Product Management

Product management is the organizational function which guides every step to product lifecycle. The product lifecycle starts from development to positioning and pricing. Its focus on the product and its customer first and foremost.

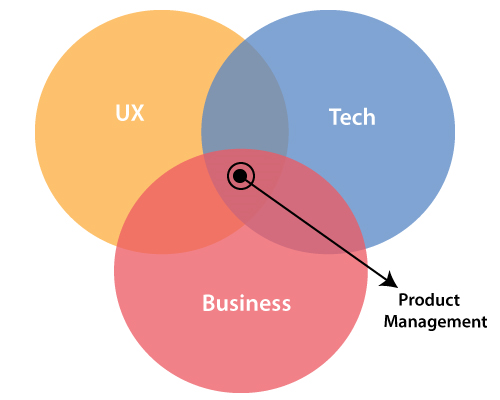
The product entirely focuses on the customer. Product teams routinely ship, better designed, and higher performing product. The product team members daily work with product managers and have interviewed dozens of their role and responsibility. In product management, there is no such one way to apply any principle. Every product has its own goals and challenges which require a unique and customized approach to product management.

## Factor affecting product management

**Business:** Product management helps teams to achieve their business objective by minimizing the communication gap between product developments, design, the customer, and the company.

**User Experience:** Product management concentrate on the user experience (UX) that represents the customer within an organization. Better UX is focus manifests itself.

**Technology:** Product management is a day to day activity in the engineering department. The accurate understanding of computer science is paramount.



## What is Agile Product Management?

Agile product management is about guiding software development, product management through multiple iterations. As agile programs are more fluid than traditional approaches so that agile product management is a more flexible approach.

Agile product managers are more integrated towards technology team than business teams. The product management is supported by the management team and Product Marketing Managers to round out the product discipline. The product manager work over marketing data and business objective.

## Product Roadmap

The product roadmap is a shared resource of truth. It is the outline of the vision, priorities, direction, and progress of product over time. The roadmap is the plan of action which aligns the organization around short-term and long-term goals for the product or project. The roadmap also plans how the product and project will be achieved.

The item on the roadmap should be clearly linked to your product strategy. It should also be responsive to changes in customer feedback and the competitive landscape. The product owner uses this roadmap to collaborate with their teams and build consent on how a product will grow and shift over time. This roadmap provides a team to keep everyone on the same page and gain context for their everyday work and future direction.

## Tips for presenting Product Roadmaps

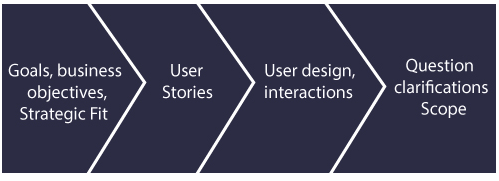
The presentation of the roadmap can be a tedious task for both developed and product managers. Several tips used to present the product roadmap are given below:

1. Set the right context
2. Consider commitments carefully
3. Make realistic plans
4. Think big, start small
5. Build a business case
6. Balance mundane with motivating
7. Roll with the punches
8. Be open and honest

## Agile Product Requirement

Building a high and new product requires research and comprehensive planning. But a question arises from where to start? The product manager generally begins with the product requirements document (PRD).

A product requirement document defines the product that we are building. Product requirement outlines the product's ambition, its features, functionalities, and behaviour.



After collecting all the product requirements, product manager shared it with stakeholders - business and technical teams. They help in building, launch, or market the product.

# 

# Agile Scale

In an Agile methodology, there are two popular frameworks- **scrum** and **kanban**. A team level uses scrum and kanban as a framework. As their popularity increases, the industries start to scale agile to suit larger organizations. There are two popular methods emerge to facilitate, these are **a scrum of scrums**, and the **Scaled Agile Framework (SAFe)**. The scrum and kanban are high starting points for scaling agile within an organization.

## Scrum of Scrums

It is the most attractive, agile framework for individual teams. When several scrum team works together on a big project, the scrum of scrums is the next step for scaling agile. The most crucial component of the scrum of scrums is a multi-team stand-up. It is a small meeting for scrum masters to talk about the agile process.

Select a member from each team to get a start, and each team represents them at the scrum of scrums, admirably someone in a technical role. It is a domestic meeting where the scrum master helps to facilitate the stand-up, but it is run just like any other team stand-up.

## Scaled Agile Framework (SAFe)

Scaled Agile Framework (SAFe) is another way to scale agile in large organizations. According to the Pioneered by Dean Leffingwell, it takes the most structured approach to scale agile than scrum of scrums. It describes three levels in the organization: portfolio, program, and team. Such structure typically appeals to larger organizations, because Scaled Agile Framework (SAFe) employs a tiered approach for the delivery of work.

The large area of SAFe is related to work, called themes, map to business epics, and architectural stories.

## Managing an agile portfolio

Agile can perform a large scale portfolio with many teams and a lot of developers. One of the examples is Netflix that uses the phrase "highly aligned, loosely coupled" to describe the well maintained agile development across a large organization.

## Expanding agile practice across the organization.

The successful company that runs agile at scale level has three common factors:

First, the entire program is iterative. The traditional is portfolio management which is focused on the top-down approach and take long periods. However, the administration takes the concept of build-measure-learn cycles for individual agile teams and applies it on a larger scale. The agile team uses modern design and share findings on a regular accent. It leads the tremendous flexibility.

Second, the organization share knowledge and break barriers between organizational silos. It also communicates across the portfolio. At the team level, similar agile ceremonies, context needs to share regularly throughout the organization so that goals, stumbling blocks, and progress are transparent for everyone.

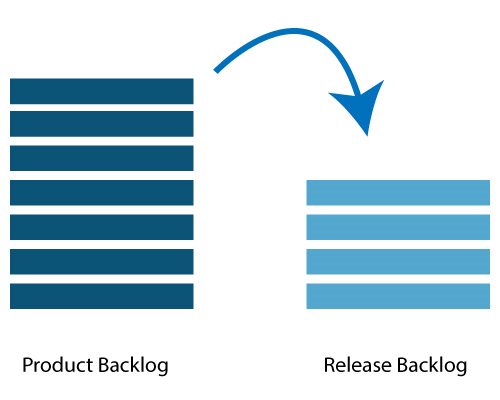
Third, the agile company makes the frequent release product (early and often) across the portfolio, even if a release involves the work of multiple programs.

# 

# 

# Agile Release Planning

The primary purpose of release planning is to make a plan to deliver an increment to the product. It is done in the interval of every 2 to 3 months.



## Who is involved in releasing the plan?

Following person are involved in product releasing plan- Scrum Master, Product Owner, Agile Development Team, Stakeholders.

* **Scrum Master:** The Scrum Master is a team leader and facility provider who helps the team member to follow agile practices so that they can meet their commitments and customers requirements.
* **Product Owner:** The Product Owner is one who runs the product from a business perspective. He defines the requirements and prioritizes their values.
* **Agile Development Team:** Agile development team provides the judgment on the technical feasibilities or any dependencies.
* **Stakeholders:** Stakeholders are the customers, subject matter, program manager act as advisers in decisions which are made around the release planning.

## Prerequisites of Planning:

The prerequisites of release planning are as follows:

* A Product Owner manages the ranked product backlog. While releasing the product, the Product owner feels to include five to ten features at the period of product release.
* High- level vision
* Market and Business objective
* Team's input according to capabilities, known velocity, or about any technical challenge.
* Acknowledged about the new product backlog items are needed

## Material Required:

## The list of materials that is required for the releasing planning is as follows:

* Flip charts, markers, whiteboards/li>
* Posted agenda, purpose/li>
* Projector for sharing the data/tools of computers required during a planning meeting/li>
* Planning data

## Planning Data:

The list of data needed during release planning are as follow:

* Previous iteration data or release planning requests
* Actions plans of previous release/iteration
* Features or defects to be considered
* Organizational and personal calendars
* Velocity from previous releases/ estimates

## Output:

Following are the output of a release planning:

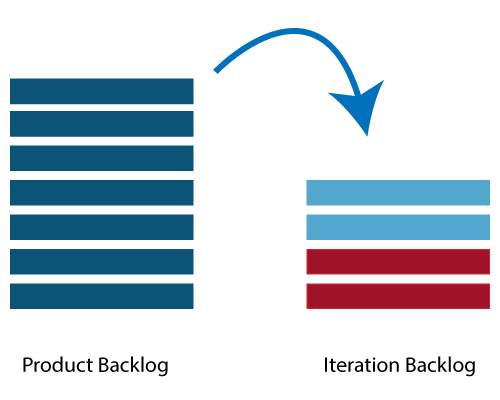
* Release plan
* Commitment
* Issues, dependencies, concerns, and assumptions which are to be monitored
* It suggests improving future release planning

## Planning Agenda

* **Opening ceremony:** Welcome message, review purpose, organizing tools, and introduction to business sponsors.
* **Product Vision and Roadmap:** It shows a broad picture of the product.
* **Review previous releases:** Product planning agenda discussed on any item which can impact the plan.
* **Product release name/theme:** It inspects the current status of roadmap themes and makes the necessary adjustment if any.
* **Issues and concerns:** In the agenda, we check any concern or issue, and then record them.
* **Review and Update the Definition of Done:** Review the product build or definition of done and make appropriate changes based on technology.
* **Retrospect:** Require feedback from participants to make the meeting successful.
* **Close:** Celebrate success.

# Agile Iteration Planning

The primary purpose of iteration planning is for the team. The team should be a complete set of the top-ranked product backlog items. The completion of top ranked product backlog is a commitment in the time needed on the length of iteration and team velocity.



## Who involved in the iteration planning?

**Scrum Master:** The Scrum Master is a team leader and facility provider. He helps the team member to follow agile practices so that they can meet their commitments and customers requirements.

**Product Owner:** The Product Owner deals with a complete view of the product backlog and their acceptance criteria.

**Agile Development Team:** Agile delivery defines their tasks and sets the effort. The effort is to estimate the requirements to fulfill the commitment.

## Prerequisites of Planning

* The items in the product backlog are sized and have a relative story point assigned.
* The product owner gave the ranking to the portfolio items.
* Acceptance criteria of each portfolio item is clearly stated.

## Planning Process

**Iteration planning involved the following steps:**

* Determines how many requirements (stories) are fit in an iteration.
* Break this requirement into tasks. Assign each task to their owners.
* Each task is set to some estimated time.
* These estimates help the team members to check how many hours for each member will be required to iterate.
* Team members are assigned tasks by seeing their velocity or capacity. Due to this, the team member is not overburdened.

## Velocity Calculation

The agile team calculates the velocity based on the previous iterations. A velocity is an average number of units that required finishing user stories in the iteration. Assume that, a team took 10,12,8 story points in each iteration for the previous three iterations, this shows that the team can take 10 as velocity for the next iteration.

Planned velocity tells the team how many user requirements can be completed in the current iteration. If the team instantly finishes the work assigned, then more user requirements can be pulled in. Otherwise, the requirement can be moved out too to the next iteration.

## Task capacity

Three factors determine the capacity of the team:

* Total number of ideal working hours in a day
* A person gives total days in each iteration
* Percentage of time a member is entirely available for the team.

Considered a team has 6 members, committed to work full time of 8 hours a day on a project. And no member is on leave during iteration, and then the task capacity for two-week iteration will be- 6 x 8 x 10 = 480 hours

## Iteration Planning Steps

* Product Owner describes the highest ranked item of the product backlog.
* Team member describes the tasks required to complete the item.
* Team members own the tasks.
* The team member estimates their own time to finish each task.
* The above steps are repeated for all the items in the iteration.
* If any member is overloaded with work, then his/her tasks distributed among other team members.

# 

# Agile Product Backlog

The agile product backlog in [Scrum](https://www.javatpoint.com/agile-scrum) is a list of prioritized features. It contains a short description of all the functionalities desired in the product. In usual scenario, items should be broken down into user stories. Commonly, a Scrum team and its product owner write everything that they can think for agile backlog prioritization.

## Why Product Backlog is Important?

* The backlog is prepared to provide an estimate of each feature.
* It helps in the planning of the product's roadmap.
* It helps in the re-ranking the features of the product by adding more value to it.
* It assists in determining the priority of the product first. The team member works first on the higher prioritize product.

## Characteristics of Product Backlog

Each product should have its own product backlog. It can be a set of large to very large features.

Multiple team members can work on a single product backlog.

Ranking of product is based on the technical value, business value, risk management, or strategic fitness.

Highest priorities items are decomposed into smaller stories during release planning. This is because they can be completed in future iterations.

## The Product Backlog comprises the following different types of items:

* Features
* Bugs
* Technical work
* Knowledge acquisition

# 

# Agile Tools

In agile development, leading as project management is not the easiest job. Jumping between your daily scrums to your next sprint, it causes hard to focus on the work. The agile development tools fulfill your needs, and does it for you.

**There are several agile tools available in the market. Some of them are listed below:**

**JIRA Agile**

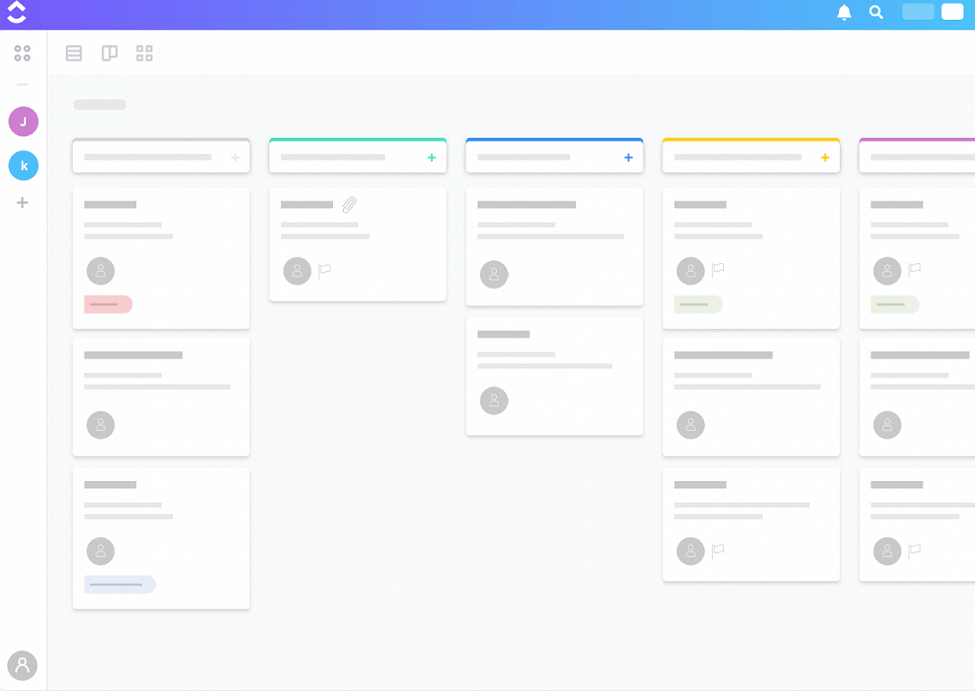
### Agile Tools

**Jira** is a tool developed by Australian Company **Atlassian**. It is used for **issue tracking, bug tracking, and project management**. The bugs and issues are related to your software and Mobile apps. The Jira dashboard consists of many useful functions and features. This function and features make secure handling of issues.

### Agile Software Features:

* Issue tracking
* Bug tracking
* Boards
* Epics
* Custom fields

### ClickUp

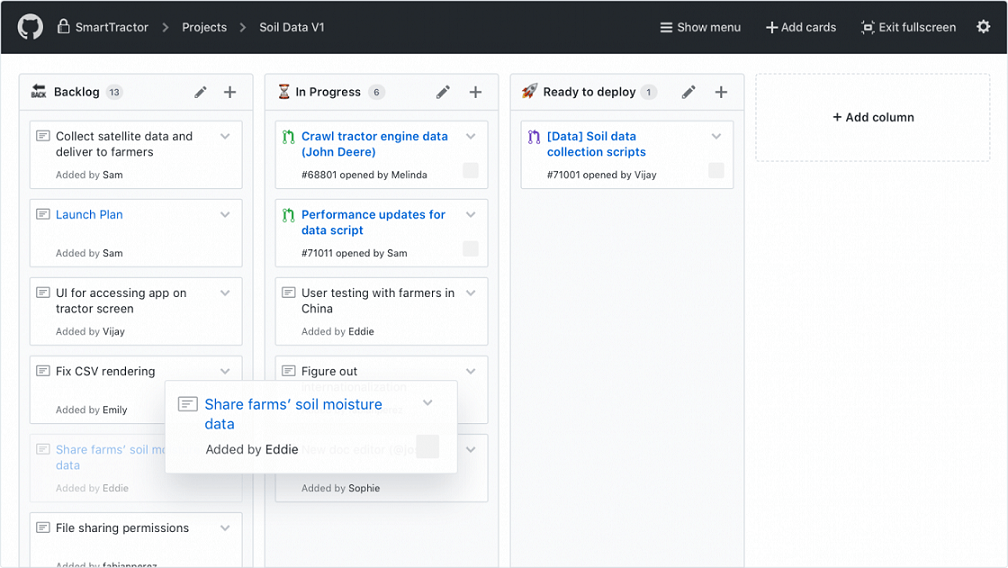


**ClickUp** is one of the ultimate agile management tools. It is used for anyone who uses agile methodology. It is the only project management tool whose goal is "to move quickly and easily". **CuickUp** is in the hand of some of the most famous agile team, including **Google** and **Apple!** It is a free forever plan, so, the team can get their hand of ClickUp.

### Agile Software Features:

* Create epics
* Use story points
* Analyze sprint performance
* Time estimates
* Start and due dates
* Time tracking

### Github



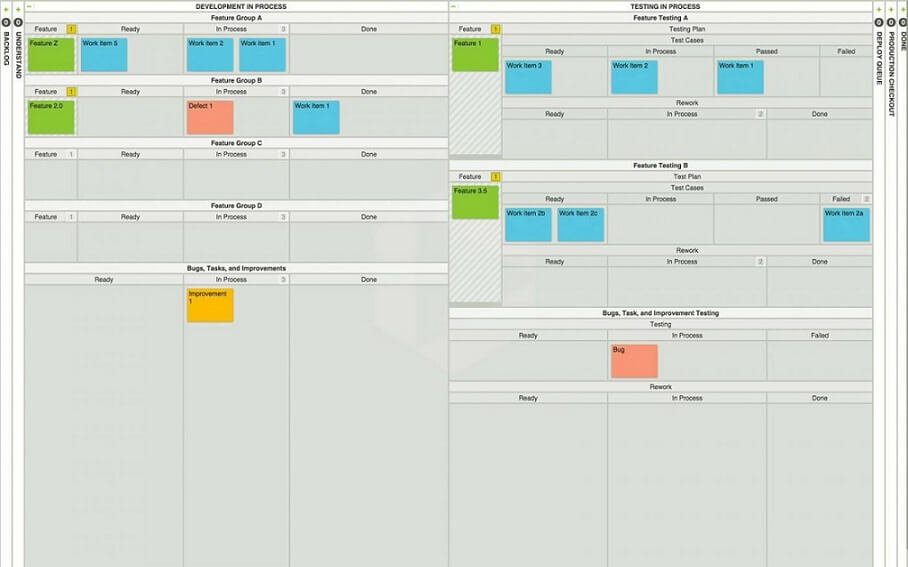
**Github** is one of the largest hosted Git serve where the developers can store all of their codes for a vast number of projects there. The Github provides such a facility of record edits across an entire team in **real time**. Github is also integrated with many other tools so, many people such as developer and product owner can work on the same code at the same time.

The project manager can make the Github work for their team. It includes lots of project management tools which help him to inspect what the development team is working on.

### Agile Software Features:

* Issue tracking
* Mentions
* Labels
* Link issues and pull requests

### LeanKit



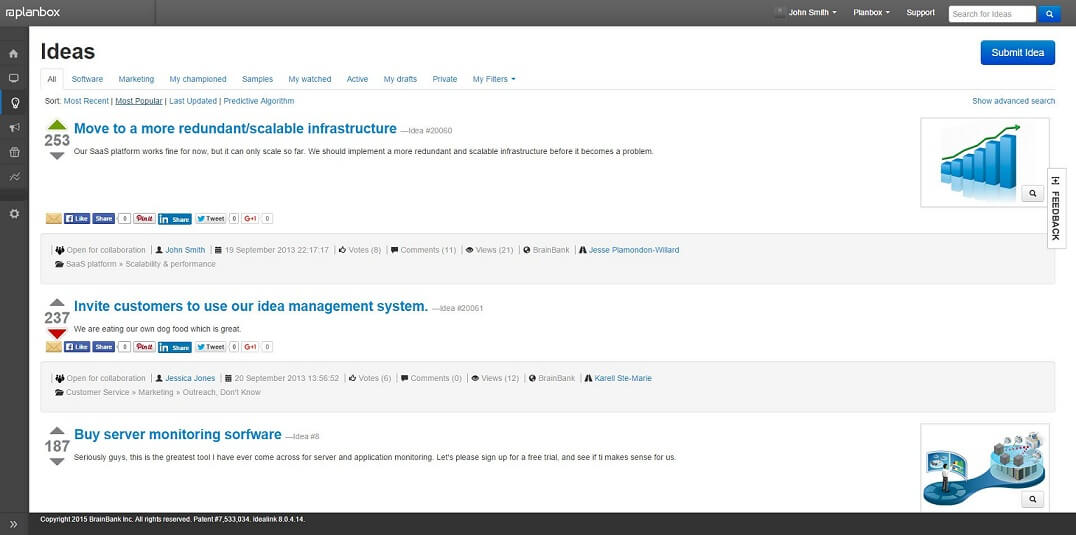
**LeanKit** is the ultimate management tool for a **Kanban** board on the agile progress for your sprints. It uses cards to represent the work items and live statuses of team member. It work perfect for the remote employees to ensure everyone can see the Kanban board in real time. It prevents the same task to complete twice and make sure the whole team remains on the same page.

LeanKit work well for cross-functional team which is benefit for Scrum or Kanban boards.

### Agile Software Features:

* Board view templates
* Track issues and bugs
* Manage project portfolios
* Lean metrics and reporting

### Planbox



Planbox is a tool that tracks the process of burndown charts. Using this everyone knows how far you are from the Sprint?s completion/goal. Burndown charts are most important part of the agile cycle. Planbox also integrates the customer bug reports, and fixes, making it useful for a wide range of users.

It has an advance reporting features which make it easy to review the status and areas where improvement is needed at Daily Scrum.

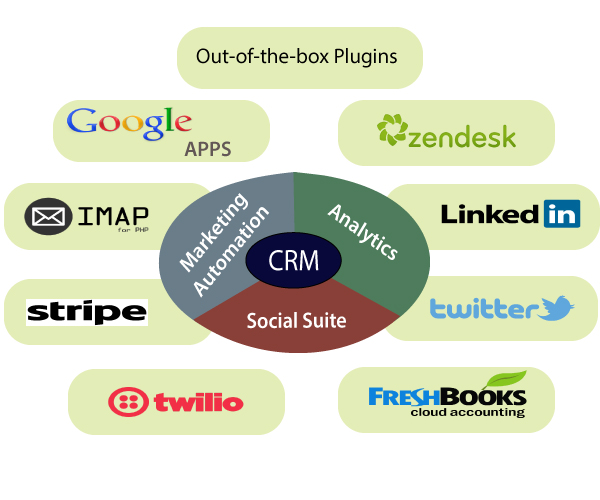
# 

# Agile CRM

Agile CRM (Customer Relationship Management) is an All-in-One CRM with Sales, Service automation, and Marketing in a single platform. It consists of sales tracking, marketing automation, contact management, web analytics, telephony, two-way emails, and helpdesk with a simple, clean, and modern interface.

Customer Relationship Management or CRM is software that helps in managing the brand's engagement with your current and future customer.

The CRM software allows you to build, organize, and present database of your customer information. This information can be updated by you and your team when the new data is discovered. It is a central storehouse of all your customer and prospect information which facilitates your team to be organized and more productive. It also allows teamwork among teams and provides management to deeper judgment into individual performance and overall growth of the business.



## Why we built Agile CRM

As an entrepreneur, everyone wants to receive positive feedback and increasing the new product's success. From the first few customers, you engage them with their name, knowing them well enough to talk to them multiple times a day. After that, they may help you to share your product to where it stands. Then, you will get more success, but more trouble also to manage the communication with all of your customers.

## Need of Agile CRM

The primary requirement of Agile CRM is broken down into the following points:

1. **Easy-to-use marketing automation:** This is easy to use because not every owner of a small businesses should need to understand the technical details of automation.
2. **A manageable CRM:** it is easy to use and affordable.
3. **Telephony:** There was probably nothing more annoying than seeing someone's phone number on CRM and then dialing their number on our phone. We spent a lot of time in solving our cell phone plans too!
4. **Social suite:** Social media playing a vital role in CRM marketing and linking customer with it. If the business owner doesn't incorporate with social media at the very beginning, then they would be left behind.
5. **Full, two-way email integration:** The owner wants to integrate personal emails fully into the CRM. So, they can send, receive, and view emails between the owner and their customers inside the CRM itself.

## Benefit of Agile CRM

The significant benefit of CRM is that the business moves to the centralized platform to store its data. It makes easy access to information from one common source. Due to the presence of best CRM, the organization gets confident to pay attention to their customer without any additional cost.

Before the availability of CRM, the data has scattered across spreadsheets, documents, address book, notebook, and email system. The CRM, simplify this tedious process, and data are access through a centralized location.

# 

# Agile Certification in 2019 (Updated)

In this era, only the traditional method of software development is not enough. Agile is the need of most organization. The demand for agile professionals is increasing in IT industries. This demand brings about the need for agile certifications.

If you are an agile professional who wants to validate your skills and level up the career, you should get one of the top agile certifications. Choose a write certification for yourself, check your strength, and go ahead. An agile certification will provide you global acceptance and make you stand out of the crowd.

Many different institutions and education bodies provide the number of agile certifications. It is a tedious task to find out one of which could help you to level up your career.

In this article, we will discuss some of the top agile certifications course provided by different institutions in the market.

## Why Agile Certifications?

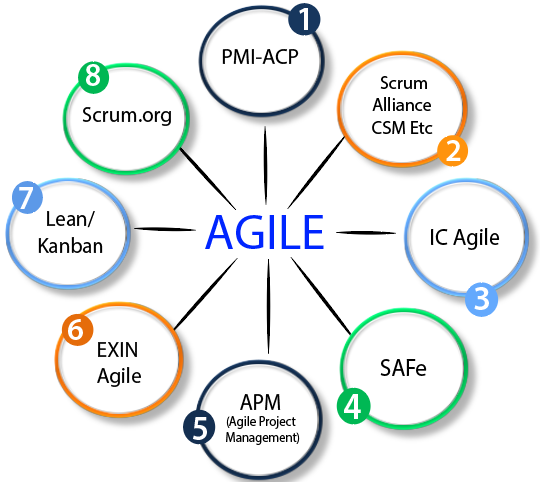
Now, agile is matured similarly to other domains, like Project Management & Service Management. The organization is looking for agile certification and experience from the candidates. Due to this reason, agile professionals aspire for the certifications. Using this certification, they can gain all the benefits when they plan to switch jobs or pursue a career in the agile domain.

**Agile certifications provide the following benefits:**

* Better Visibility
* Better Salary
* Better Credibility
* Keeping up to the current market trend



**Top 5 Agile Certifications**



There are lots of certifications available in the market to select the gaining credentials in the agile domain. Due to lots of certification available, you get confused about which agile certification to be chosen at what point of time.

Below diagram illustrate the various certifications available in the market. You have to choose one of the best agile certifications, which is good for you.

**The top 5 agile certifications which are best for agile professionals who want to build their career with agile methodology.**

1. PMI-ACP (Agile Certified Practitioner)
2. Scrum Alliance (Certified Scrum Master/Certified Scrum Product Owner/Certified Scrum Developer)
3. Scrum Alliance (Certified Scrum Professional)
4. Scrum.org (Professional Scrum Master/Professional Scrum Product Owner/Professional Scrum Developer -1)
5. SAFe Scaled Agilists

## PMI Agile Certified Practitioner (PMI-ACP)

The **Agile Certificate Practitioner** certificate is one of the best project management certifications. It is provided by the Project Management Institute (PMI). PMI-ACP certificate is designed for the project management professionals who are using agile methodology. This certificate is also designed for that professional who is practicing their projects or planning to move to the agile.

## Scrum Alliance (Certified Scrum Developer/Certified Scrum Master/Certified Scrum Product Owner)

Scrum Alliance (scrumalliance.org) is a favorite organization for those who are practicing Scrum in their project. There are 3 popular certifications provided by Scrum Alliance for Agile Scrum professional, and these are: **Certified Scrum Developer**, **Certified Scrum Master**, and **Certified Scrum Product Owner**.

## Scrum Alliance (Certified Scrum Professional)

The Scrum Alliance provides a professional level certificate known as a **Certified Scrum Professional** certificate. It is one of the top agile certifications. It is also the top project management certifications. This certificate recognizes the ability of professional in advance practices, roles, artifacts, and procedures of the Scrum Framework.

## Scrum.org (Professional Scrum Developer-I/ Professional Scrum Master/ Professional Scrum Product Owner)

Scrum.org provides the professional level scrum certificate. This certificate validates the candidate's knowledge in the understanding of Scrum. This also checks how candidate applies scrum practices and framework in real-world situations. It offers certification for all the scrum developers, scrum masters, and scrum product owners. There is no need to join anyone for this certification training from scrum.org to achieve these certificates; instead, they need compulsory to pass the certification exam.

## SAFe Scaled Agilist

SAFe Scaled Agilist is also called as SAFe Agilist or Scaled Agilist. It is a two-day training course provided by the training providers. The candidates of this certification are more than 5 years of experience in software development, testing, project management, business analysis, or scrum. In this course, the candidates get detailed information about Lean-Agile leadership principles. Candidates also learn how to launch SAFe in agile.

**Top 20 Agile Interview Questions and Answers**

### 1) What is an agile or agile methodology?

Agile is an iterative approach of software development methodology using short iterations of 1 to 4 weeks. Due to the agile methodology, the development process is aligned to deliver the changing business requirement.

### 2) What are some quality strategies of agile?

Some quality strategies of agile are:

* Iteration
* Re-factoring
* Dynamic code analysis
* Short feedback cycles
* Reviews and inspection
* Standards and guidelines
* Milestone reviews

### 3) What are an agile manifesto and its principle?

Agile manifesto uncovers the better way of developing software by doing it and helping others to do it. Agile has 4 manifesto and 12 principles which defines:

* Individuals and interactions, i.e., self-motivating and self-organized should be encouraged.
* Demonstrate the working software at regular intervals with comprehensive documentation.
* Customers are collaboration over contact negotiation.
* Responding to change over following a plan.

**The principles of agile manifesto are-**

1. **Customer Satisfaction:** Manifesto provides high priority to satisfy the costumer's requirements. Customer satisfaction is done through early and continuous delivery of valuable software.
2. **Welcome Change:** Making change during software development is common and inevitable. Every changing requirement should be welcome, even in the late development phase. Agile process is used to increase the customer's competitive advantage.
3. **Deliver the Working Software:** Deliver the working software frequently, ranging from a few weeks to a few months with considering the shortest period.
4. **Collaboration:** Business people (Scrum Master and Project Owner) and developers must work together during the entire life of a project development phase.
5. **Motivation:** Projects should be build around motivated team members. Provide such environment that supports individual team members and trust them. It makes them feel responsible for getting the job done thoroughly.
6. **Face-to-face Conversation:** Face-to-face conversation between Scrum Master and development team and between the Scrum Master and customers for the most efficient and effective method of conveying information to and within a development team.
7. **Measure the Progress as per the Working Software:** The working software is the key and primary measure of the progress.
8. **Maintain Constant Pace:** The aim of agile development is sustainable development. All the businesses and users should be able to maintain a constant pace with the project.
9. **Monitoring:** Pay regular attention to technical excellence and good design to maximize agility.
10. **Simplicity:** Keep things simple and use simple terms to measure the work that is not completed.
11. **Self-organized Teams:** The Agile team should be self-organized. They should not be depending heavily on other teams because the best architectures, requirements, and designs emerge from self-organized teams.
12. **Review the Work Regularly:** The work should be reviewed at regular intervals, so that the team can reflect on how to become more productive and adjust its behaviour accordingly.

### 4) Is there any disadvantage of the agile model (SDLC)?

Disadvantages of Agile SDLC:

1. The development team should be highly professional and client-oriented.
2. New requirement may be a conflict with the existing architecture.
3. With further correction and change, there may be chances that the project will cross the expected time.
4. There may be difficult to estimate the final coast of the project due to constant iteration.
5. A defined requirement is absent.

### 5) What are the burn-up and burn-down chart?

The burn-up chart depicts the amount of work done in the project, whereas the burn-down chart illustrates the amount of work remaining in the project. Thus, the burn-up and burn-down are used to describe the progress report of the project.

### 6) What do you understand by Daily Stand-Up?

The daily stand-up is the day-to-day meeting (mostly in the morning) in which the whole team meets around 15 minutes to find the answer for the following three questions:

* What was done yesterday?
* What is your plan for today?
* Is there any obstacle that restricts you to complete your task?

### 7) What do you understand about Scrum?

**Scrum is a framework** that helps agile teams work together to develop, deliver, and sustain the complex product in the shortest time. The product provides by scrum team in this shortest period is known as a **sprint**.

### 8) What are the different roles in Scrum?

There are three different roles in scrum. These are the Scrum Master, Product Owner, Agile Development Team:

* **Scrum Master:** The Scrum Master is a team leader and facility provider who help the team member to follow agile practices so that they can meet their commitments and customers requirements.
* **Product Owner:** The Product Owner is one who runs the product from a business perspective. He defines the requirements and prioritizes their values.
* **Agile Development Team:** Agile development team provides the judgment on the technical feasibilities or any dependencies.

### 9) What are the responsibilities of the Scrum Master?

The critical responsibility of Scrum Master includes:

* Tracking and monitoring project development.
* Understanding the user requirement correctly.
* Work to obtain the project properly.
* Improving the performance of the team.
* Organized meetings and resolve issues.
* Communicate and report to the customer and development team.

### 10) What are different ceremonies and their importance in Scrum?

To clearly express the Scrum planning, Scrum review, Scrum Daily stand up, and scrum retrospective is the purpose of the ceremony. The importance of these ceremonies is to use sprint as per your project.

### 11) What do you know about Scrum ban?

Scrum-ban is a Scrum and Kanban-based model for software development. This model is used in the project that needs continuous maintenance, various programming error, or some sudden changes.

### 12) What do you understand by the term agile testing?

The agile testing is the software testing process which is fully based on the principle of agile software development. It is the iterative approach where the user story becomes the output of the collaboration between the product owner and the development team.

### 13) What are the major principles of agile testing?

Some of the essential principles of agile testing are:

* Customer satisfaction
* Face to face communication
* Sustainable development
* Continuous feedback
* Quick respond to changes
* Successive improvement
* Self-organized
* Focus on essence
* Error-free clean node
* Collective work

### 14) What are the skills of a good agile tester?

The agile tester is one who implements the principle of agile software development principles for software testing. An excellent agile tester has the following skills:

* He must be familiar with the principles and concept of agile.
* He must be excellent communication skill to communicate with the team and the clients.
* He can set the priority of a task according to customer requirements.
* He should able to understand the customer requirement properly.
* He should understand the project risk due to changing demand.

### 15) Name the agile frameworks.

Some of the agile frameworks are:

* Scrum
* Kanban
* Feature Driven Development
* Test Driven Development

### 16) Is it ever suggested to use waterfall over Scrum? If yes, explain when.

Yes, sometimes we use waterfall module over scrum. This is because when the client requirement is simple, small, well-defined, fully understood, predictable, and the subject does not change until the project complete.

### 17) Name some methodologies and development where you have used the agile model.

While answering this type of question, keep in mind to mention those methodologies from which you are familiar whit. Some of the methodologies where agile is used are:

* Crystal methodologies
* Lean software development
* Dynamic development
* Feature-driven development

### 18) What was the length of sprints/iterations in your project?

It is a common question for experienced people. The idea behind is to judge in which kind of environment you have worked? There will be follow up of the question that the length fixed in the beginning and never changed? Did you try with less than this length or more than that?

### 19) What is the difference between the agile & traditional way of working?

The traditional way of development is that which follow the sequential where design -> development -> testing etc. is performed whereas, in agile development, all of this is done in every iteration/sprint.

### 20) Why does Scrum encourage the use of automated testing for projects?

Due to faster possible delivery of the project, the Scrum development encourages to use automated (automated performance or automated regression) testing. While answering this question, you should explain some tools that you have used for automated testing.