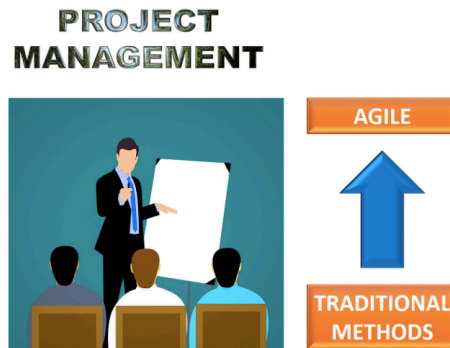


Agile Project Management

Software projects face the risk of failure during the life cycle due to misuse of resources, wrong prioritization and inaccurate understanding of customer demands. This has led companies to experiment with a variety of new methods to produce more productive projects. Agile is a new approach to project management methods resulting from these experiments. This method is based on an incremental and iterative approach that helps teams deliver a product to their customers faster and more desirable. In this respect, the framework of agile promises **significant advantages over traditional methods**. In recent years, most companies have adopted the agile approach, which they find more successful than traditional methods. In this context, we can say that almost every software developer has used the agile method in some form.



Dividing Product Development Life Cycle into Small Components



Agile provides a light framework for development teams. That helps them focus on fast delivery while maintaining functionality. In Agile, the product development life cycle is divided into small components (called iterations), so the product can be easily and rapidly developed and tested. In this way, changes can be made without having to wait for the final product.

Customer's Desired Requirements



The agile method provides that the product is optimized throughout the development process. Using **iterative planning** and getting **feedback** ensures that a delivered product is tailored to the customer's requirements. The status of the project is measured and evaluated throughout the process to easily adapt to changing needs. Measurement and evaluation provide accurate and early visibility into the development of each project.

<https://www.youtube.com/watch?v=mfQIjf2TPdk>

The Agile Manifesto

In 2001, 17 independent Software Leaders met in America for brainstorming. They aimed to find out how to develop software better by using different knowledge and approaches. After two days of brainstorming, they released the Agile Manifesto.

The Agile Manifesto was a powerful statement, carefully crafted using only 68 words. Everyone agreed that the Agile Manifesto was both short and authoritative. While traditional methods advocated a stable plan and avoided changes, the manifesto focused on people, communication, the product, and flexibility.

The Agile Manifesto basically mentions the following four values:

1. Individuals and interactions over processes and tools:

Valuing people more highly than processes or tools is easy to understand. Because it is the human who responds to business needs and drives the development process.

2. Working software over comprehensive documentation:

Historically, great amounts of time have been spent on documenting the product for development and ultimate delivery. Technical specifications and requirements, interface design documents, test plans, documentation plans, and approvals required for each. The Agile Manifesto values documentation, but it values working software much more.

3. Customer collaboration over contract negotiation:

With traditional development models such as Waterfall, customers negotiate the requirements for the product, usually in great detail, prior to any work starts. This means the customer is involved in the process of development before development began and after it was completed, but not during the process. However, the Agile Manifesto describes a customer who is engaged and collaborates throughout the development process.

4. Responding to change over following a plan:

Traditional software development methods regard change as an expense, so it is to be avoided. With Agile, the shortness of an iteration cycle means priorities can be shifted from iteration to iteration and new features can be inserted into the next iteration. Agile's view is that changes always improve a project and provide additional values.

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

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

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

The Agile Manifesto

Agile Principles

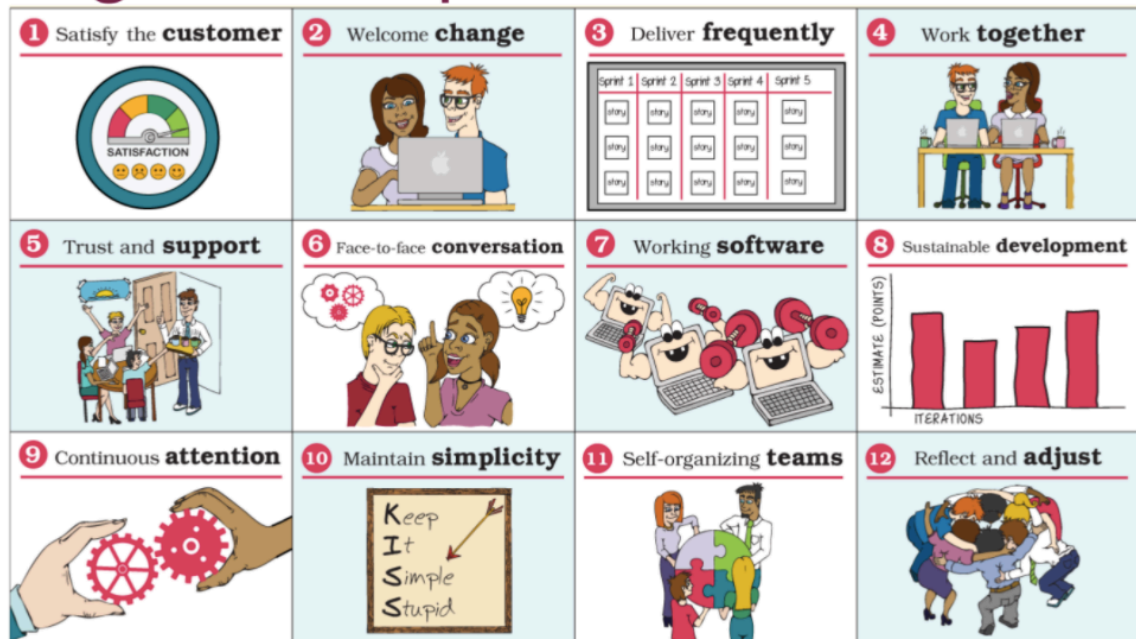
In the months following the publication of the Agile Manifesto, the original signatories continued to communicate. They augmented the four values of the manifesto with the following 12 principles.

Our highest priority is to **satisfy the customer** through early and continuous delivery of valuable software.

1. **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.
2. Deliver working software frequently, from a couple of weeks to a couple of months, with preference to the **shorter timescale**.
3. Business people and developers must **work together** daily throughout the project.
4. Build projects around **motivated individuals**. Give them the environment and support they need, and trust them to get the job done.
5. The most efficient and effective method of conveying information to and within a development team is **face to face conversation**.
6. Working software is the primary **measure of progress**.
7. Agile processes promote **sustainable development**. The sponsors, developers, and users should be able to maintain a **constant pace** indefinitely.
8. **Continuous attention** to technical excellence and **good design** enhances agility.
9. Simplicity -the art of **maximizing the amount of work** not done- is essential.
10. The best architectures, requirements, and designs emerge from **self-organizing teams**.
11. At **regular intervals**, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

So far you have learned about the agile method. In the next lesson, you will watch a comprehensive interactive video about agile.

► 12 Agile Principles



Q: Do you know about Agile Manifesto & its Principles? Explain in brief.

A: There are four values in the manifesto. Individuals and interactions, working software, customer collaboration and responding to the changes are the values. Stemming from these values there are 12 principles in agile. These principles can be summarized as to satisfy the customer, to welcome changing requirements, good cooperation between business people and developers (working together), face to face conversation, motivated individuals and simplicity.

- Interview Q&A

Building the Right Product

It can be said that the agile method helps companies build the **right product**. Rather than attempting to sell the product before it is released, the agile method encourages teams to improve the product during its development. This helps the company to be in the marketplace as profitable as possible. That retains the vital business value and guarantees the desirable product. This is the reason why the agile method is an appealing option for both partners and developers.

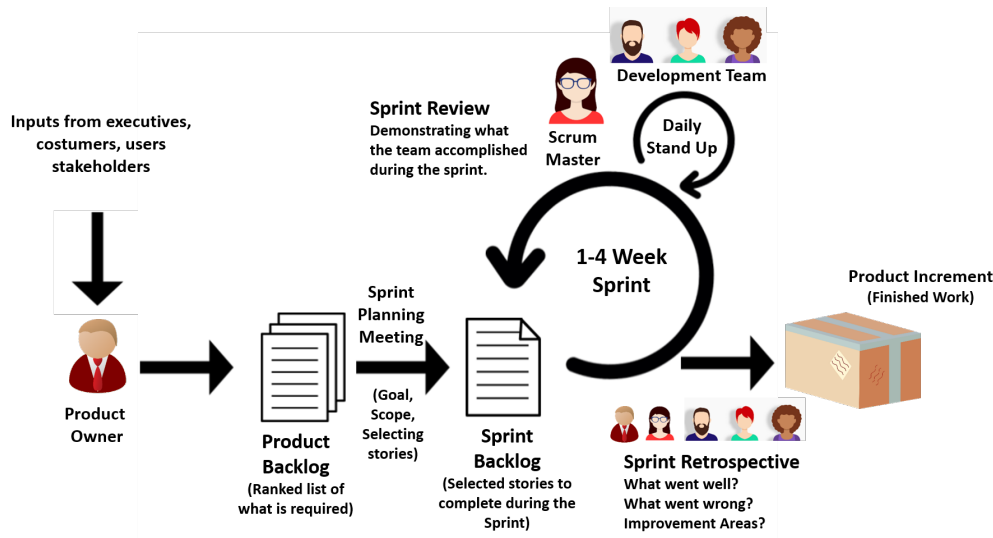
Top Methodologies Used to Implement Agile

There are several different methodologies used to implement agile. In the following lessons, we will examine in detail the two most common agile methodologies: **Scrum** and **Kanban**. Now we are starting with an interactive video about Scrum.

<https://www.youtube.com/watch?v=2Vt7Ik8Ublw&feature=youtu.be>

Scrum Framework and Sprint Concept

Scrum is the most common methodology to implement the Agile. It is an iterative development model used in complex software development processes. In scrum, larger projects are divided into smaller parts that can be managed with **sprints**. Sprints are the periods from one to four weeks. It can be even a few days when needed. Steady sprint length reduces variability; a scrum team can safely predict what they can do on each sprint based on what they have done in previous sprints. The implementation of sprints allows scrum teams to make arrangements for instant improvement, rather than at the end of the project. At the end of each sprint, something remarkable is revealed. For gaining feedback from users or investors, the product produced during each iteration should be demonstrated. The scrum framework defines specific roles, artifacts, and activities for projects. The following figure shows all of these components of the scrum framework that we will discuss later one by one.



The Scrum Framework



Tips:

- Sprint is also called iteration.

Q: What is the duration of a scrum sprint?

A: It depends on the number of people in the development team and the size of the project. In general, a scrum sprint is completed in 1-4 weeks.

- Interview Q&A

Scrum Roles

There are three main roles in scrum projects. These are the **Product Owner**, **Scrum Master**, and **Development Team**.

Product Owner (PO) is the business representative in the team and speaks for the needs of the project for maximizing the value delivered in each sprint. The product owner represents stakeholders and is the voice of the customer. Therefore, the product owner works together with stakeholders and prioritizes the product requirements.

Scrum Master coaches the team, protects the team from organizational distraction, clears any obstacles encountered and helps team members focus on what they do. Scrum Master ensures that scrum is understood well by the team members and it is working properly. Scrum Master constantly improves the team's environment. While the product owner has a directing role, Scrum Master has an enabling role in a scrum team.

A **Development team** usually consists of 3-9 people and performs daily tasks. The team is project-oriented and dedicated to the success of the project. Each team member is very talented that is, the team members are skilled in certain subjects. Each member can do more than one job on the project. Discipline and integrity are the key terms for a successful team.



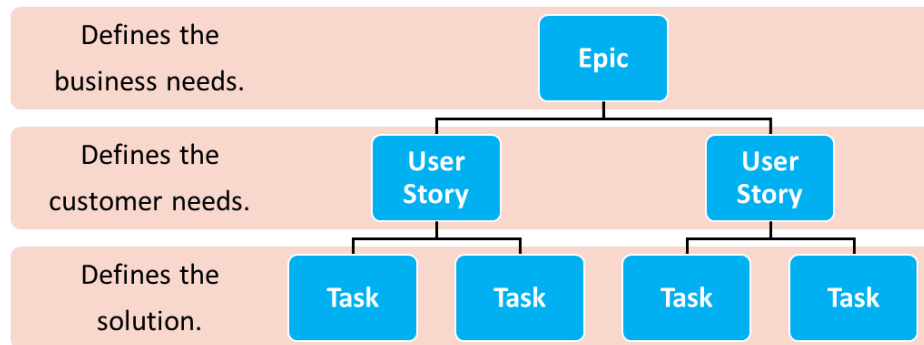
Scrum Roles Scrum Roles

Q: What is the role of the Scrum Master?

A: Scrum Master coaches the team, protects the team from organizational distraction, clears any obstacles encountered and helps team members focus on what they do. Scrum Master ensures that scrum is understood well by the team members and it is working properly. Scrum Master constantly improves the team's environment. While the product owner has a directing role, Scrum Master has an enabling role in a scrum team. - **Interview Q&A**

Epic, User Story and Task Concepts

The three basic terms used in scrum projects are epic, user story and task. In order to fully comprehend a scrum project, it is very important to understand what these concepts are. The hierarchy between these concepts is explained as follows.



Hierarchy Between Epic, User Story and Task

Epic

An epic refers to a set of jobs that cannot be easily achieved in a single sprint. It usually takes months to perform an epic. Normally, it refers to a series of requirements that have not yet been split into user stories. We can also consider the epic as one of the major goals for your agile team to work on, yet not simplified and divided into various tasks.

Epics are generally large-scaled and do not contain details. They must be divided into multiple user stories before being worked on. Therefore, they are located at the top of the hierarchy. An epic could be a feature of the product, customer demand or business requirement.

Examples of Epics:

- As a bank, we want a facial recognition system in our branches.
- As the marketing department, we want a mobile application and a website to reach more customers.

User Story

At the hierarchy, the user story is between the epic and the task. It describes a software feature from the customer's perspective and includes the **type of user**, **what they want**, and **why they want it**. Therefore, it answers the 'who', 'what' and 'why' in a simple language. The product owner has the responsibility of user stories.

Leaving out the technical aspect, it should describe the behavior from a user's perspective.

Tips:

- Template of a User Story:
As a < type of user >, I want < some goal > so that < some reason >

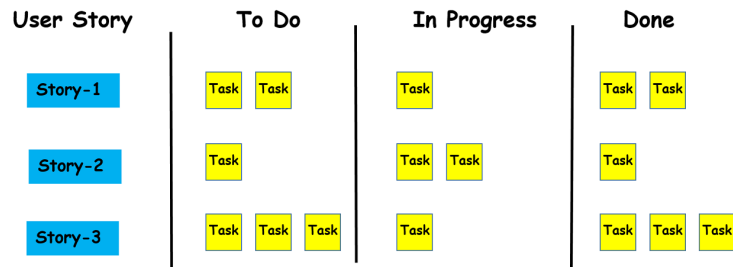
Examples of User Stories:

- As a registered user, I want to add items to the cart so that I can purchase multiple items at once.
- As a student, I want to apply for the exam online so that I can save time.

Q: What is a User Story? **- Interview Question**

Task

Tasks are detailed pieces of work that are necessary to realize a user story. In other words, they define the solutions for customer needs. The time for tasks can range from a couple of hours to several hours and assigned to a team member. Tasks are displayed on a scrum board for easy tracking, as shown in the figure below.

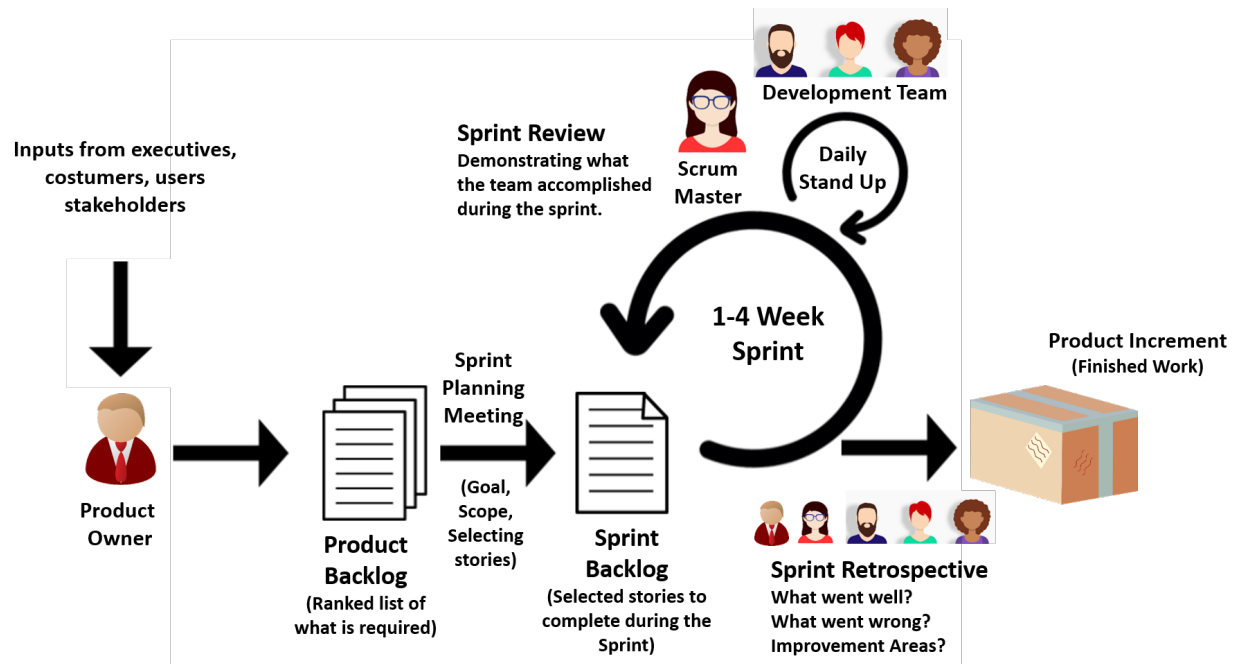


The scrum board helps to visualize the tasks and usually contains cards or post-it notes on a whiteboard. The board is usually divided into three categories: *to do*, *in progress*, and *done*. The team members update the board by moving task cards through the columns on the board.

Scrum Artifacts

Introduction

In scrum, artifacts serve to capture the common understanding of the team. The Scrum framework defines three major artifacts. These concepts are **product backlog**, **sprint backlog**, and **product increment**.



The Scrum Framework

Product Backlog

The product backlog refers to the **list of everything that needs to be done to complete the project**. Beside all user stories, it also includes technical tasks. The product backlog is the responsibility of the product owner. The product owner fulfills this responsibility by creating the product backlog, **prioritizing** the requirements in the product backlog list and constantly updating this list. The product owner updates the product backlog because once a story is completed, it should be removed from the list. Sometimes, however, new stories are added, as the project grows.

Sprint Backlog

The sprint backlog can be defined as a subset of the product backlog. The sprint backlog is generated from the product backlog during the sprint planning meeting at the beginning of each sprint. The user stories selected from the product backlog, which will be completed during the sprint constitute the sprint backlog.

The sprint backlog is **not a flexible list** like a product backlog. That means the sprint backlog is unchanged during the sprint period. Once agreed upon in the sprint planning meeting, the stories, and steps to complete them remain stable during the sprint length. If there are stories left still unfinished by the end of the sprint, they will be added back to the product backlog and addressed during the next sprint.

Product Increment

The Product Increment is the sum of all the product backlog items finished during the sprint. In other words, by the end of each sprint, the development team creates a new software that gets built into the main product and this new software is called product increment. The product increment aims to invest in **small amounts** in the new features of the main product. This helps to shorten the time before receiving feedback.

As the name implies, product increment continues to increase within the subsequent sprints. That means each product increment includes all the previous sprint increment values as it is cumulative. It should be a **working product** at the end of each sprint, which means that the product should be ready for shipping. But that doesn't mean it can be released in the market. It should go through a detailed testing phase and be of high quality before releasing it in the market.

Q: Explain the term 'increment' in Scrum.

A: The Product Increment is the sum of all the product backlog items finished during the sprint. In other words, by the end of each sprint, the development team creates a new software that gets built into the main product and this new software is called product increment. The product increment aims to invest in small amounts in the new features of the main product. This helps to shorten the time before receiving feedback. As the name implies, product increment continues to increase within the subsequent sprints. That means each product increment includes all the previous sprint increment values as it is cumulative.

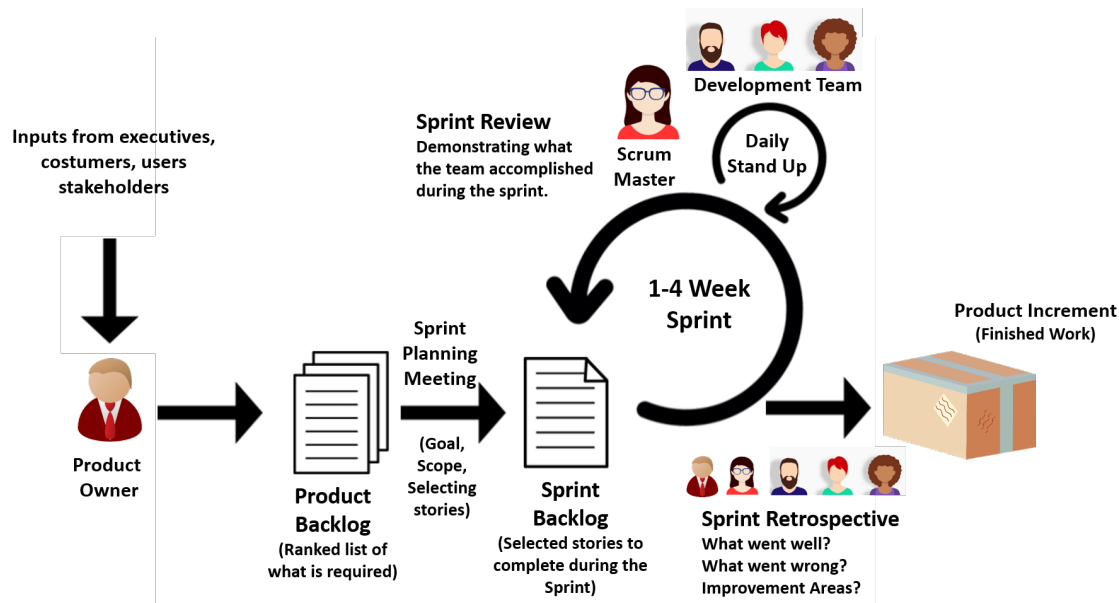
- Interview Q&A

Scrum Ceremonies

List of Ceremonies in Scrum

In the scrum process, a series of meetings called ceremonies are held regularly. These ceremonies aim to minimize the need for meetings that are not defined in the scrum and to ensure regularity. These ceremonies are:

1. Sprint Planning Meeting,
2. Grooming Meeting,
3. Daily Stand Up Meeting or Daily Scrum,
4. Sprint Review Meeting,
5. Sprint Retrospective Meeting.



The Scrum Framework

Q: What are different ceremonies and their importance in Scrum?

A: In the scrum process, a series of meetings called ceremonies are held regularly. These ceremonies aim to minimize the need for meetings that are not defined in the scrum and to ensure regularity. These ceremonies are: 1. Sprint Planning Meeting, 2. Grooming Meeting, 3. Daily Stand Up Meeting or Daily Scrum, 4. Sprint Review Meeting, 5. Sprint Retrospective Meeting.

- Interview Q&A

Sprint Planning Meeting

This activity takes place at the start of each sprint and all the scrum roles take part in this meeting. Normally, it lasts for one or two hours. In sprint planning meetings, the scrum team decides the scope and the goals of the sprint.

In this respect, the product owner presents to the team the prioritized list of the user stories from the product backlog. The team members discuss the stories and predict how much work the team can perform in the coming sprint. Then they select some of the stories to complete during the sprint. These selected stories constitute the sprint backlog.

Grooming Meeting

The backlog grooming meeting is held to **review the backlog** and to ensure the backlog contains the appropriate items. The product owner, scrum master and some or all of the development team participate in the meeting.

Reviewing the backlog items involves; rewriting some items to be more expressive, splitting big ones into smaller items, deleting old or no more need items. That helps the team keep the product backlog tidy and getting backlog items ready for upcoming sprints. At the end of the grooming meeting, the team has a fresh prioritized product backlog.



Tips:

- Apart from the grooming meeting, the other four meetings are held at specific times. For example, daily stand-up meetings are held every day, and others are held at the beginning or end of each sprint. However, no specific date has been allocated for the grooming meeting. It is usually arranged in the middle of the sprint period.

Daily Stand Up Meeting

The daily stand up meeting is an internal meeting for the development team. The scrum master ensures that the team has the meeting, but the development team itself is responsible for conducting the daily stand up meeting.

Regardless of the length of the sprint, the maximum time allocated for this event is 15 minutes. The members of the development team give three statements about

- What they completed yesterday,
- What they will focus on today,
- What are the things impeding them?

The scrum master deals with the removal of blocking issues. Some blocking issues may also need to be removed by the product owner.

Sprint Review Meeting

Sprint review meeting takes place after each sprint. The participants of this meeting are typically the product owner, the development team, the scrum master, customers and developers from other projects. In this meeting, the development team demonstrates to the whole organization **what they accomplished** during the sprint and **receives feedback**. It is important to receive feedback because that helps the product owner to update the product backlog and consider the scope of the next sprint.

The sprint review meeting is usually held informally, typically with rules that prohibit the use of PowerPoint slides and not allowing more than a couple of hours of preparation time for the meeting.

Sprint Retrospective Meeting

Like sprint review meetings this meeting is also held at the end of each sprint. The attendees of this meeting are the development team, the scrum master and the product owner. In this meeting, all the participants discuss:

- What went well in the previous sprint?
- What didn't work well?

- What are the improvement areas to increase team performance?

A two-hour retrospective meeting is enough for a two-week sprint. The aim of the retrospectives is not to listen to complaints but to find **effective solutions to problems** and to develop action plans.

Q: What is the purpose of a retrospective?

A: Like sprint review meetings this meeting is also held at the end of each sprint. The attendees of this meeting are the development team, the scrum master and the product owner. In this meeting, all the participants discuss:

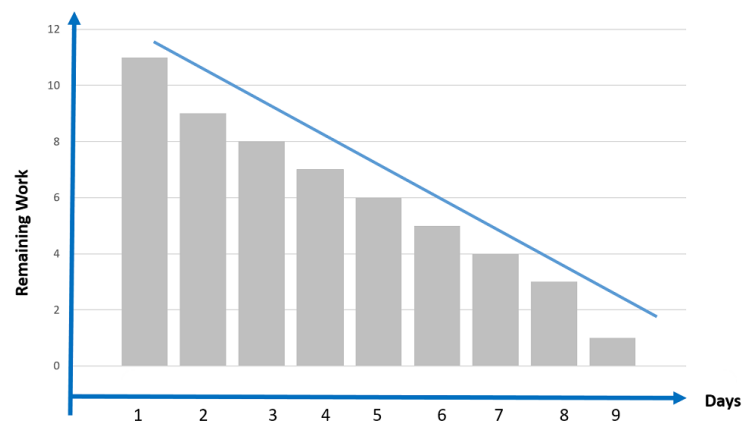
- What went well in the previous sprint?
- What didn't work well?
- What are the improvement areas to increase team performance?

- Interview Q&A

Other Artifacts in Scrum

Burndown Chart

In addition to the before-mentioned main artifacts, scrum projects also include some other artifacts, such as a burndown chart, timeboxing, story point or velocity.



An Example of Burndown Chart

A burndown chart is a graphical demonstration of work left to do versus time. In the chart, the vertical axis usually represents the remaining work while the horizontal axis represents the time or days. It is useful for forecasting when all of the work will be completed. A simple example of a burndown chart is given above.

Q: What is the use of burn-down charts?

A: A burn-down chart demonstrates the amount of work remained to complete a project. So, the burn-down chart is used to trace the progress of a project.

- Interview Q&A

Timeboxing

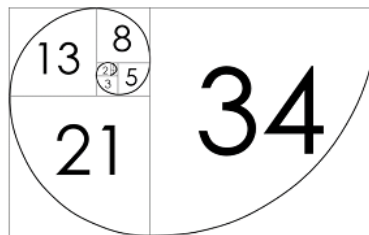


A timebox is a period of time in which a team works to achieve a goal. Instead of waiting until the goal is reached, the timeboxing approach stops when the time limit is reached. The timeboxing aims to define and limit the amount of time dedicated to a certain event. Scrum framework uses timeboxing as a tool for all of the scrum activities for concretely defining ambiguous tasks.

Story Point

Story Points are decided upon and used by individual scrum teams. A Story Point is a relative unit of measure to provide relative predictions of effort for completing tasks or user stories.

A Story Point provides an easier estimation to the team. Rather than assessing a product backlog item and estimating it in hours, teams consider only how much effort this item will require, relative to other product backlog items.



FibonacciNumbersFibonacciNumbers

In other words, a story point is a numeric value that indicates the difficulty level of the user story. Before the development team makes an estimation, story points are assigned to each user story using the **Fibonacci numbers** (1, 2, 3, 5, 8, 13, 21, 34...). A story that is assigned 2 story points should be twice as difficult as a story that is assigned 1 story point.

Q: Why aren't user stories simply estimated in man-hours?

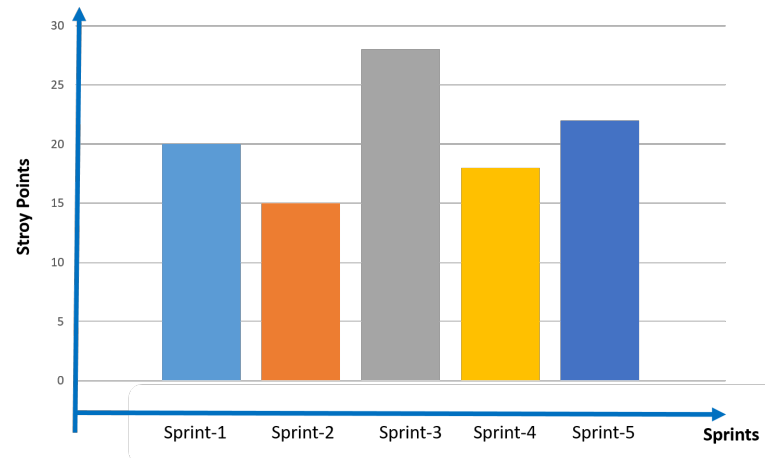
A: Estimation of user stories on the basis of man-hours is possible but not preferred. Because in that case, you won't be able to concentrate on the quality product to be delivered to the customer. In addition to that, you will concentrate on the cost and budget of the management while using man-hours. Rather than man-hours, story points are used, as it provides a complete idea about both the complexity of work and required efforts.

- Interview Q&A

Velocity

Velocity is an indication of the average amount of items from the product backlog turned into a product increment during each sprint. In other words, it is a measure of the amount of work the development team can handle during a single sprint. In scrum, velocity is the key metric.

At the end of each sprint, velocity is calculated by summing up the story points for completed user stories. Points from not completed or partially completed user stories should not be considered in calculating velocity. After calculating the velocity at the end of the first sprint, the team can make a new estimation of how long the project will take to complete.



Velocity in Scrum Sprints

Suppose the total value of story points for the remaining user stories is 80, and the total value of story points for the completed user stories in the first sprint is 20. That means the team needs 4 more sprints to complete the whole project.

Q: What do you know about the Velocity in Scrum?

- Interview Question

Top Methodologies Used to Implement Agile

As you recall, Scrum and Kanban are the two most common agile methodologies to implement Agile. So far you have learned something about Scrum. And now you will learn something about Kanban.

Introduction to Kanban

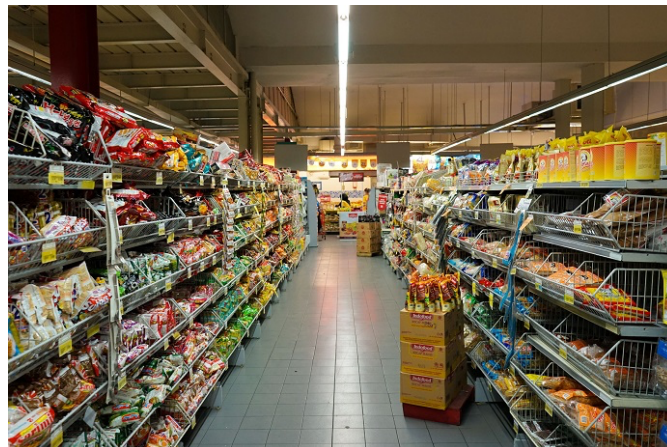
Scrum is the most common way to implement the agile, however, kanban is another popular methodology for implementing the agile system in the business.

The word "kanban" is in Japanese and can be translated as "the card you can see" or "signboard". As the name implies, it is a visual framework used to implement agile and shows **what to produce, when to produce it, and how much to produce**.

Kanban focuses on **visualizing the workflow process**. The primary goal of kanban is to identify potential flaws in the process and to correct them to ensure that works progress at the desired pace.

A Brief History of Kanban

Kanban's story dates back to the 1940s. During these years, Toyota updated its production method based on the model that supermarkets use to manage stocks on shelves. Supermarkets stock enough products to meet consumer demand. This is a method that optimizes the flow between the supermarket and the consumer. Since inventory levels match the consumption rate, the supermarket stores the optimum quantity of products at any given time.



Supermarket Shelves

Toyota adopted the system used by the supermarket. To monitor capacity levels in real-time, the company started using a card called "kanban" between different production teams. When a bin of material used on the production line was unloaded, a kanban was sent to the warehouse, explaining which material was required. With the new materials provided by the warehouse, production continued without interruption, and the process monitored continuously.

The kanban method controls the entire chain from production to the end consumer. In this way, it prevents both the failure of supply and the over-stocking of the goods at various stages of the production process. This requires continuous monitoring of the process.

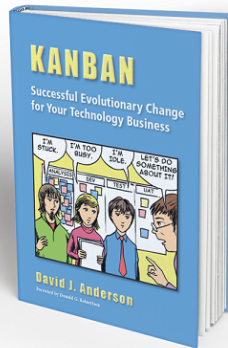
In the following lesson, you will watch an interactive video about the kanban process.

<https://www.youtube.com/watch?v=iVaFVa7HYj4&feature=youtu.be>

Kanban in Software Development

Introduction

Although its story dates back to the 1940s, the implementation of kanban in the world of computer technologies is a relatively new topic. In 2004, David J. Anderson introduced the idea of using the kanban concept for software development, and in 2010 he wrote a book named "Kanban: Successfully Evolutionary Change for your Technology Business ". Kanban's use in software development begins after these attempts by David J. Anderson.



Q: Explain what is Kanban.

A: A Kanban is like a flash card carrying all the information about the current status of your work and the work required to be done on the product at each stage of the software development process.

- Interview Q&A

The Book written by David J. Anderson

Comparison of Kanban and Scrum



It is not easy to completely separate the kanban and scrum methods. You remember the scrum board. For example, most people believe that scrum teams use a kanban board in the scrum process. On the other hand, there are some fundamental differences between the two methods.

- There are no certain time limits in kanban while sprints in scrum have a start and end dates,
- While there are rigid deadlines for the tasks on the scrum board, kanban is more flexible in this regard,
- In contrast to designated roles in a scrum project like product owner, scrum master, and development team, there are no certain roles in kanban,
- The team rearranges the scrum board after each sprint while the same kanban board is used throughout the project.

Q: Describe the places where 'Scrum' and 'Kanban' are used?

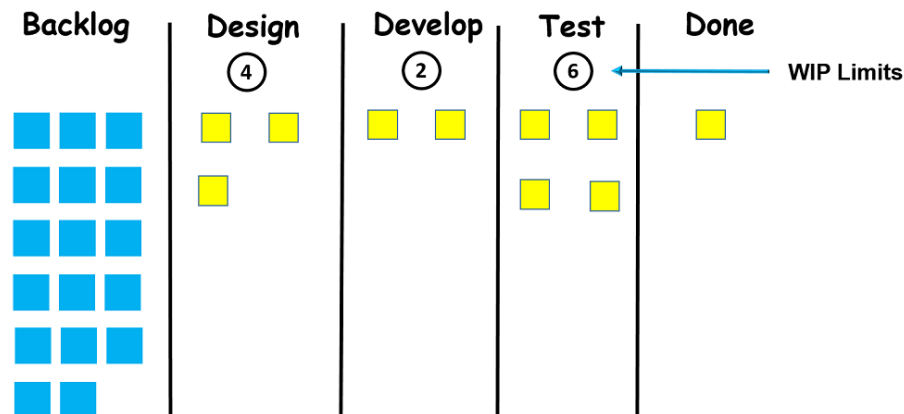
A: Scrum is a better choice when you need a more prominent process. However, if you want improvement in running the process without much changes in the whole scenario, you should use Kanban.

- Interview Q&A

The Kanban Board

A kanban board is a tool designed to visualize the work and increase efficiency using cards and columns. As you remember the word "kanban" means "the card you can see", therefore the board on which cards are attached is the main item of a kanban project.

There are three main components of a kanban board: **visual signals**, **columns** and **work-in-progress limits (WIP)**.



The Kanban Board

Visual Signals: Project and work items are written onto cards by kanban teams. For an agile project, each card can correspond to a different user story or a task. Those cards help the team quickly understand **what they working** on and **what is the current situation** of the project.

Columns: Each **stage of a project** is demonstrated with a column. Those stages together compose a **workflow**. Visual signals or cards go through the workflow until they are completed.

WIP Limits: The maximum number of cards that can be in a column at any given time is called WIP limits. WIP limits are written **on the top of each column** on the board.

Types of Kanban Boards

A kanban board can be a physical whiteboard or a digital board. Because of its size, a whiteboard can easily give you an overview of the current status. A physical kanban board is flexible; it is quite convenient to draw new lines and add new columns if required. It also allows the whole team to stand around it, understand the process of the project. If the whiteboard has wheels, you can move it to the meeting rooms. On the other hand, if your team is not working in the same office they can use kanban digital boards remotely.

Q: How does Kanban improve visibility?

A: Kanban uses digital or physical boards to demonstrate the team's workflow. The tasks demonstrated by cards move from left to right representing the progress. So at any given time, the organization can see progress, capacity, productivity, and efficiency.

- Interview Q&A

Principles and Practices of Kanban

Kanban has adopted **Four Foundational Principles** and **Six Core Practices** to manage the workflow and increase productivity. The four principles of kanban are:



Principles of Kanban

Start with what you are doing now: Do not make instant changes to the existing setup or process. Kanban must be directly applied to the existing workflow. Necessary changes should be made slowly and gradually, so, the team feels comfortable.

Agree to pursue incremental, evolutionary change: Make minor incremental changes rather than major changes that might lead to resistance within the team.

Initially, respect current roles, responsibilities, and job-titles: You do not need to modify your existing roles and functions that perform well. The team will coordinate and implement the necessary changes to the roles and titles. These three principles help managers overcome the expected emotional resistance and fear of change.

Encourage acts of leadership at all levels: Constant improvement at all levels of the organization is encouraged. It is desirable that all team members produce ideas, show leadership and contribute to continuous improvement..

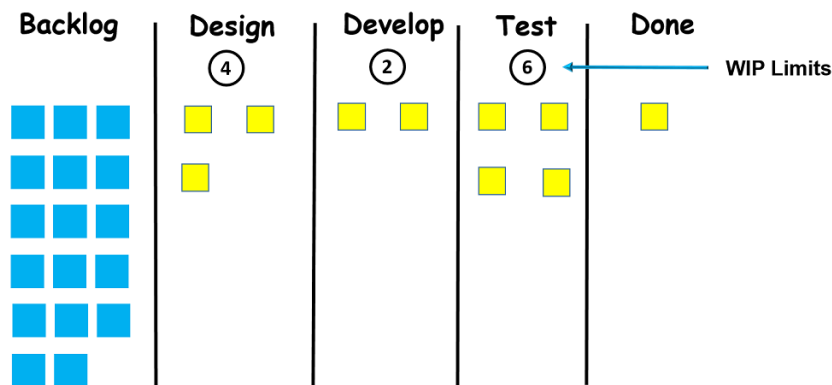
Besides the abovementioned four principles, there are six core practices in kanban. When you understand these six practices well, you will learn about the kanban process. Because these six practices explain the kanban process.



Visualization is the first step to start with kanban. You need to visualize the current workflow on either a physical whiteboard or a digital Kanban board. This board can be simple or detailed depending on the complexity of the work. When you visualize the process, it can be seen what you and your team are currently doing.



Limiting Work in Progress (WIP) is the second practice in the kanban. The maximum number of cards that can be in a column at any given time is called WIP limits. When you assign a limit to each column, your team doesn't work more than they can handle.



In other words, you expect your team **to complete the current work first** before taking up a new one. So, you create capacity in the system and new jobs can be received by the team. Initially, it may not be so easy to predict the WIP limits of your team. Therefore, to start with no WIP limits can be considered. After making observations and getting sufficient data about your team you can define WIP limits for each stage in the workflow. Or you can start with a WIP limit of 1 to 1.5 times the number of people taking part in each stage or each column.

Q: Ideally, how WIP limit is calculated with respect to team size?

A: You can start with a WIP limit of 1 to 1.5 times the number of people taking part in each stage or each column. For example, if team size is 4 in a particular stage, max 6 items can be in progress at any given time. **Interview Q&A**

Managing Flow

After visualization and limiting each stage, the third practice relates to improving the flow of the process. Managing and improving flow is the key point of your kanban system after applying the first two practices. Because there are two options at this point. You will either come across a smooth flow within WIP limits or you will observe the works start piling up. This affects the delivery time of the products.



Related to the delivery time of the product a new concept called **lead time** comes into the picture. Lead time is the period between creating a task in your workflow and its final departure from the kanban board. Therefore, Kanban helps you monitor workflow, identify blockages, and make adjustments to improve the flow for reducing the lead time. As you manage the flow and improve it, your team's pace becomes more predictable. This makes it easy for you to make commitments to your customers. And it is very important to make reliable commitments in the kanban philosophy.

Q: What is lead time in Kanban?

A: Lead time is the period between creating a task in your workflow and its final departure from the kanban board. Therefore, kanban helps you monitor workflow, identify blockages, and make adjustments to improve the flow for reducing the lead time. - **Interview Q&A**

Making Process Policies Explicit

Making process policies, basic principles and working methods explicit is the fourth practice of kanban. Being explicit and clear on these issues allows the team to follow the process easily and make proposals for the necessary improvements. In other words, the team members who are not clear about the existing process can not improve the system.

Implementing Feedback Loops

During the third practice, we discussed reducing the lead time by managing the flow. However, rapid delivery can not solve all the problems. You should also know what your customers and the end-users think, and how much the product contributes to the value of your company.



This is where fifth practice comes into the picture, you need to get feedback from people not involved in your system. Feedback loops are also required within a system to ensure that the desired functionality is provided with the high quality. That's where different kinds of tests come out. Automated continuously running tests are preferred as they shorten feedback loops.

Improving Collaboratively, Evolving Experimentally (Using the Scientific Methods)

Because the kanban is an evolutionary improvement process you should evaluate your system continuously and improve it constantly. That allows you to adopt minor changes in the process and improve the workflow at a constant pace. While improving your workflow, kanban encourages you to use scientific methods like you form a hypothesis and test it.



Then you make changes depending on the results of the test. If you are implementing agile principles, your main task is to continuously evaluate your process and improve it when needed. You can observe and measure the impact of the changes that you make by scientific methods. By using these methods, you can assess whether a change helps you improve or not. Then you make a decision about whether to try some other solutions or to keep the current status.

Q: Are there any drawbacks of the Agile model? If yes, please explain.

A: Yes, there are some drawbacks of the Agile method, some of them are as follows:

- 1- It is not easy to make an estimation of the effort required to complete a task. It becomes more complex in the case of large projects as it becomes difficult to make a prediction about the total effort required.
- 2- In case the desired requirements of the client are not understood properly, the final project will not meet the customer requirements. Thus, this will lead to customer dissatisfaction.
- 3- Only the leader who has considerable experience in the Agile model is capable to take important decisions. The team members with less or no experience are not involved in the decision-making process, thus they don't have a chance to advance their knowledge.

- Interview Q&A

Interview Questions About Agile, Scrum and Kanban

Interview Questions

What is the duration of a scrum sprint? -

It depends on the number of people in the development team and the size of project. In general, a scrum sprint is completed in 1-4 weeks.

What do you know about the Velocity in Scrum?

At the end of each sprint, velocity is calculated by summing up the story points for completed user stories. Points from not completed or partially completed user stories should not be considered in calculating velocity. After calculating the velocity at the end of the first sprint, the team can make a new estimation of how long the project will take to complete. Suppose the total story points for the remaining user stories is 80, and the total story points for the completed user stories in the first sprint is 20. That means the team needs 4 more sprints to complete the whole project.

Do you know about Agile Manifesto & its Principles? Explain in brief.

There are four values in the manifesto. Individuals and interactions, working software, customer collaboration and responding to the changes are the values. Stemming from these values there are 12 principles in agile. These principles can be summarized as: to satisfy the customer, to welcome changing requirements, good cooperation between business people and developers (working together), face to face conversation, motivated individuals and simplicity.

What is the use of burn-down charts?

A burn-down chart demonstrates the amount of work remained to complete a project. So, the burn-down chart is used to trace the progress of a project.

What is the role of the Scrum Master?

Scrum Master coaches the team, protects the team from organizational distraction, clears any obstacles encountered and helps team members focus on what they do. Scrum master ensures that scrum is understood well by the team members and it is working properly. Scrum master constantly improves the team's environment. While product owner has a directing role, scrum master has an enabling role in a scrum team.

What are different ceremonies and their importance in Scrum?

In the scrum process, a series of meetings called ceremonies are held regularly. These ceremonies aim to minimize the need for meetings that are not defined in the scrum and to ensure regularity. These ceremonies are:

-
1. Sprint Planning Meeting,
 2. Grooming Meeting,
 3. Daily Stand Up Meeting or Daily Scrum,
 4. Sprint Review Meeting,
 5. Sprint Retrospective Meeting.

What is the scrum of scrums?

Suppose there are 6 teams working on a project and each team has 6 members. Each team leads its own special scrum meeting. However, in order to coordinate and communicate with different teams, it is required to organize a separate scrum meeting. The meeting organized to hold coordination between scrum teams is called the scrum of scrums. There is one team leader from every group, known as ambassador, who is responsible to represent his team in the scrum of scrums.

Explain the term 'increment' in Scrum.

The Product Increment is the sum of all the product backlog items finished during the sprint. In other words, by the end of each sprint, the development team creates a new software that gets built into the main product and this new software is called product increment. The product increment aims to invest in small amounts in the new features of the main product. This helps to shorten the time before receiving feedback. As the name implies, product increment continues to increase within the subsequent sprints. That means each product increment includes all the previous sprint increment values as it is cumulative.

Describe the places where 'Scrum' and 'Kanban' are used?

Scrum is a better choice when you need a more prominent process. However, if you want improvement in running the process without much changes in the whole scenario, you should use Kanban.

What is a user story?

It describes a software feature from the customer's perspective and includes the type of user, what they want, and why they want it. Therefore, it answers the 'who', 'what' and 'why' in a simple language. The product owner has the responsibility of user stories. Leaving out the technical aspect, it should describe the behavior from a user's perspective. Examples of User Story: As a registered user, I want to add items to the cart so that I can purchase multiple items at once.

Why aren't user stories simply estimated in man-hours?

Estimation of user stories on the basis of man-hours is possible but not preferred. Because in that case, you won't be able to concentrate on the quality product to be delivered to the customer. In addition to that, you will concentrate on the cost and budget of the management while using man-hours. Rather than man-hours, story points are used, as it provides a complete idea about both the complexity of work and required efforts.

Explain what is Kanban.

A Kanban is like a flash card carrying all the information about the current status of your work and the work required to be done on the product at each stage of the software development process.

How does Kanban improve visibility?

Kanban uses digital or physical boards to demonstrate the team's workflow. The tasks demonstrated by cards move from left to right representing the progress. So at any given time, the organization can see progress, capacity, productivity, and efficiency.

Ideally, how WIP limit is calculated with respect to team size?

You can start with a WIP limit of 1 to 1.5 times the number of people taking part in each stage or each column. For example, if team size is 4 in a particular stage, max 6 items can be in progress at any given time.

What is lead time in Kanban?

Lead time is the period between creating a task in your workflow and its final departure from the kanban board. Therefore, kanban helps you monitor workflow, identify blockages, and make adjustments to improve the flow for reducing the lead time.

Are there any drawbacks of the Agile model? If yes, please explain.

Yes, there are some drawbacks of the Agile method, some of them are as follows:

- It is not easy to make an estimation of the effort required to complete a task. It becomes more complex in the case of large projects as it becomes difficult to make a prediction about the total effort required.
- In case the desired requirements of the client are not understood properly, the final project will not meet the customer requirements. Thus, this will lead to customer dissatisfaction.
- Only the leader who has considerable experience in the Agile model is capable to take important decisions. The team members with less or no experience are not involved in the decision-making process, thus they don't have a chance to advance their knowledge.

What is the purpose of a retrospective?

Like sprint review meetings this meeting is also held at the end of each sprint. The attendees of this meeting are the development team, the scrum master and the product owner. In this meeting, all the participants discuss:

- What went well in the previous sprint?
- What didn't work well?
- What are the improvement areas to increase team performance?