

Scrum Guide

Wednesday, May 27, 2020 12:06

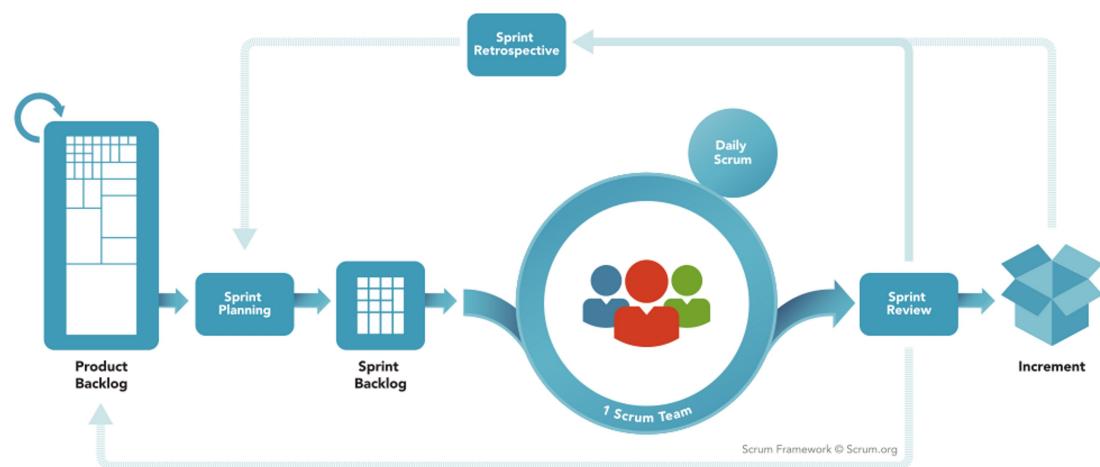
- ★ **Definition of Scrum:** A framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.

The Scrum Framework

The framework consists of:

- Scrum Teams
- roles
- events
- artifacts
- rules - bind together the roles, events, and artifacts, governing the relationships and interaction between them.

SCRUM FRAMEWORK



Roles	Artifacts	Events
<ul style="list-style-type: none">• Product owner• Development team• Scrum master	<ul style="list-style-type: none">• Increment• Product backlog• Sprint backlog	<ul style="list-style-type: none">• Sprint planning• Sprint review• Sprint retrospective• Daily scrum

Scrum Theory

Scrum is founded on empirical process control theory, or *empiricism*. It uses an interative, incremental approach to optimize predictability and control risk.

Empiricism (n): Knowledge comes from experience and making decisions based on what is known.

3 Scrum Pillars of Empirical Process Control

1. Transparency

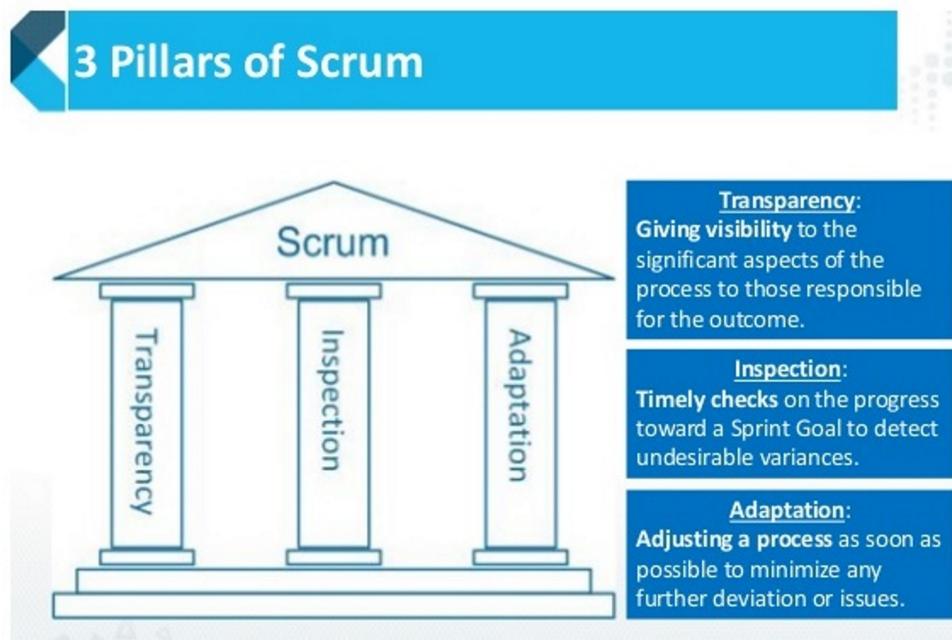
- a. Important aspects of the process must be visible to those responsible for the outcome (Ex: project manager).
- b. These aspects must be defined by a common standard so observers share a common understanding of what is being seen.
- c. Need a common language for the process, a common definition of "Done".

2. Inspection

- a. Scrum users must frequently inspect Scrum artifacts and progress toward a Sprint Goal to detect undesirable variances.
- b. Inspection should not be so frequent as to get in the way of work.
- c. Inspections are best performed by skilled inspectors at the point of work.

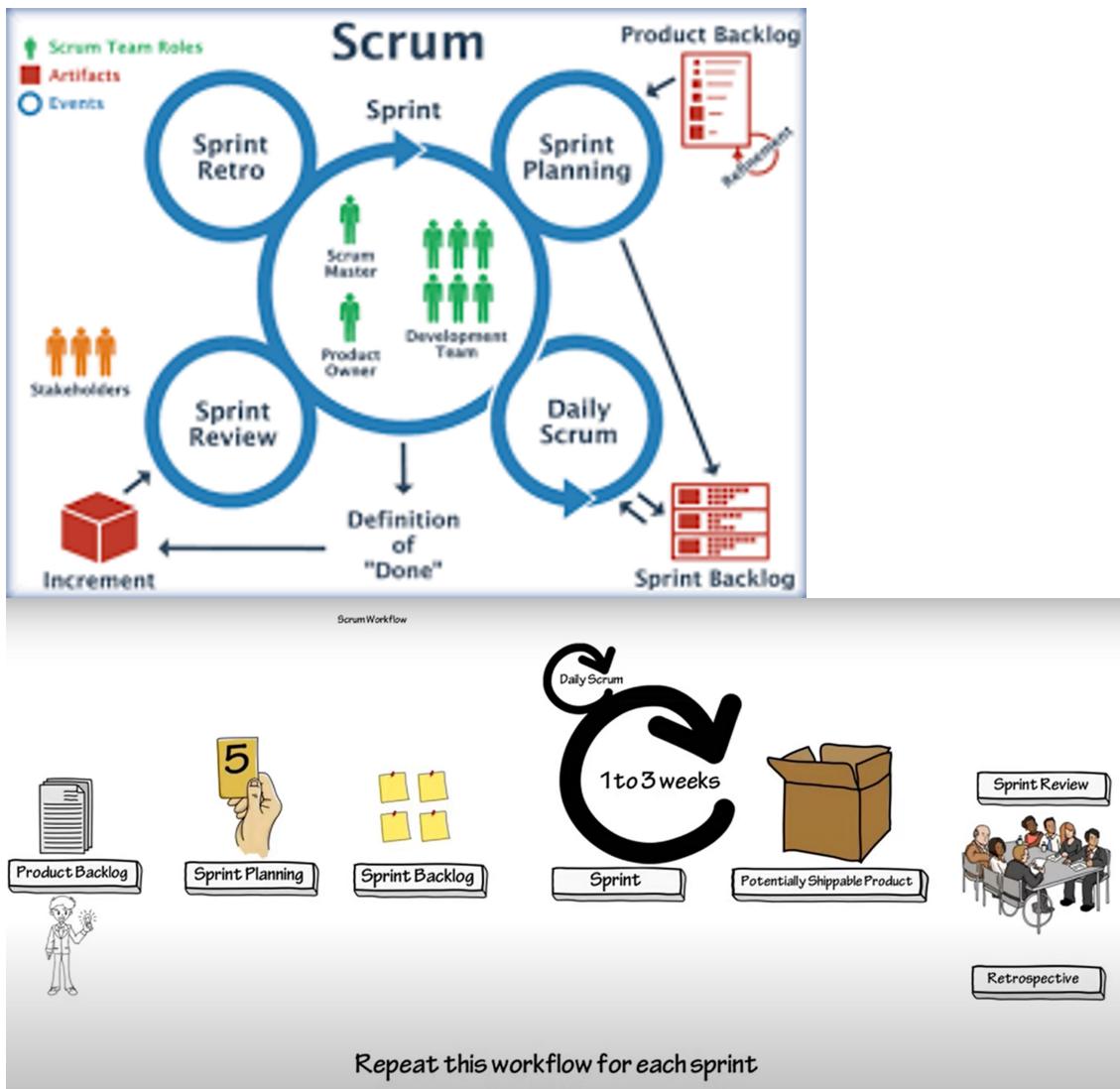
3. Adaptation

- a. If one or more aspects of a process deviate outside acceptable limits, the process or material being processed must be adjusted.
- b. Adjustments must be made as soon as possible to minimize further deviation.



4 Formal Scrum Events for Inspection and Adaptation

1. Sprint Planning
2. Daily Scrum
3. Sprint Review
4. Sprint Retrospective



5 Scrum Values

1. Commitment - to achieving the goals of the Scrum Team.
2. Courage - to do the right thing and work on tough problems.
3. Focus - on the work of the Sprint and the goals of the Scrum Team.
4. Openness - about all the work and the challenges with performing the work.
5. Respect - each other to be capable, independent people.

When these 5 Scrum values are adopted and lived by the Scrum Team, the 3 Scrum pillars come to life and **trust** is built.



© Scrum.org

3 Scrum Roles in the Scrum Team

1. *Product Owner* (one person)
 - a. Responsible for maximizing the value of the product resulting from work of the Development Team.
 - b. Is the sole person responsible for managing the Product Backlog.
 - c. Product Backlog management includes:
 - Clearly expressing Product Backlog items;
 - Ordering the items in the Product Backlog to best achieve goals and missions;
 - Optimizing the value of the work the Development Team performs;
 - Ensuring that the Product Backlog is visible, transparent, and clear to all, and shows what the Scrum Team will work on next; and,
 - Ensuring the Development Team understands items in the Product Backlog to the level needed.
 - d. The Product Owner may do the above work or have the Dev Team do it. However, the Product owner remains accountable.
 - e. For the Product Owner to succeed, the entire organization must respect his/her decisions.
2. *Development Team*
 - a. Consists of professionals who do the work of delivering a potentially releasable Increment of "Done" product at the end of each Sprint.
 - b. A "Done" Increment is required at the Sprint Review. Only members of the Dev Team create the Increment.
 - c. Dev Teams are structured and empowered by the organization to organize and manage their own work. This optimizes the Dev Team's overall efficiency and effectiveness.
 - d. Characteristics of Development Teams:
 - Self-organizing. No one (not even the Scrum Master) tells the Development Team how to turn Product Backlog into Increments of potentially releasable functionality;
 - Development Teams are cross-functional, with all the skills as a team necessary to create a product Increment;
 - Scrum recognizes no titles for Development Team members, regardless of the work being performed by the person;
 - Scrum recognizes no sub-teams in the Development Team, regardless of domains that need to be addressed like testing, architecture, operations, or business analysis; and,

- Individual Dev Team members may have specialized skills and areas of focus, but accountability belongs to the Dev Team as a whole.
- e. Development Team Size: 7 +/- 2 members.
 - i. No less than 3, no more than 9.
 - ii. The Product Owner and Scrum Master are not included in this count unless they are also executing the work of the Sprint Backlog.
- 3. Scrum Master
 - a. Responsible for promoting and supporting Scrum as defined in the Scrum Guide.
 - b. Scrum Masters help everyone understand Scrum theory, practices, rules, and values.
 - c. The Scrum Master is a *servant-leader* for the Scrum Team.
 - d. They help those outside the Scrum Team understand which of the interactions with the Scrum Team are helpful and which aren't. This helps to maximize the value created by the Scrum Team.
 - e. Scrum Master Service to the Product Owner
 - Ensuring that goals, scope, and product domain are understood by everyone on the Scrum Team as well as possible;
 - Finding techniques for effective Product Backlog management;
 - Helping the Scrum Team understand the need for clear and concise Product Backlog items;
 - Understanding product planning in an empirical environment;
 - Ensuring the Product Owner knows how to arrange the Product Backlog to maximize value;
 - Understanding and practicing agility; and,
 - Facilitating Scrum events as requested or needed.
 - f. Scrum Master Service to the Development Team
 - Coaching the Development Team in self-organization and cross-functionality;
 - Helping the Development Team to create high-value products;
 - Removing impediments to the Development Team's progress;
 - Facilitating Scrum events as requested or needed; and,
 - Coaching the Development Team in organizational environments in which Scrum is not yet fully adopted and understood.
 - g. Scrum Master Service to the Organization
 - Leading and coaching the organization in its Scrum adoption;
 - Planning Scrum implementations within the organization;
 - Helping employees and stakeholders understand and enact Scrum and empirical product development;
 - Causing change that increases the productivity of the Scrum Team; and,
 - Working with other Scrum Masters to increase the effectiveness of the application of Scrum in the organization.

Scrum Roles: A different way of thinking, a better way to drive success

Scrum roles differ from traditional project roles.
By collaborating, a Scrum team delivers more business value, faster.



Scrum Teams are self-organizing and cross-functional.

Self-organizing teams choose how best to accomplish their work, rather than being directed by others outside the team.

Cross-functional teams have all competencies needed to accomplish the work without depending on others not part of the team.

The team model is designed to optimize flexibility, creativity, and productivity.

Teams deliver products iteratively and incrementally, maximizing opportunities for feedback. Incremental deliveries of "Done" product ensure a potentially useful version of the working product is always available.

6 Scrum Events

- The Sprint
 - usually 2 weeks
 - A time-box of one month or less during which a "Done", useable, and potentially releaseable product increment is created.
 - Consistent durations throughout a development effort.
 - A new Sprint starts immediately after the conclusion of the previous Sprint.
 - Sprints consist of Sprint Planning, Daily Scrums, the development work, the Sprint Review, and the Sprint Retrospective.
 - Each Sprint has a built in goal, a design and flexible plan that will guide building it, the work, and the resultant product increment.
 - Sprints help to limit risk.
 - A Sprint can be cancelled before the Sprint time-box is over.
 - Only the Product Owner has the authority to cancel the Sprint.

- A Sprint would be cancelled if the Sprint Goal becomes obsolete, but that rarely ever happens.
- Sprint Planning
 - The work to be performed in the Sprint is planned at the Sprint Planning.
 - This plan is created by the collaborative work of the entire Scrum Team.
 - The Scrum Master ensures that the event takes place and their attendants understand its purpose. The Scrum Master teaches the Scrum Team to keep it within the time-box.
 - Sprint Planning answers the following:
 - What can be delivered in the increment resulting from the upcoming Sprint?
 - How will the work needed to deliver the increment be achieved?
 - Only the Development Team can assess what it can accomplish over the upcoming Sprint.
 - During Sprint Planning the Scrum Team also crafts a Sprint Goal. The Sprint Goal is an objective that will be met within the Sprint through the implementation of the Product Backlog, and it provides guidance to the Development Team on why it is building the Increment.
 - The Product Backlog items selected for this Sprint plus the plan for delivering them is called the **Sprint Backlog**.
- Sprint Goal
 - An objective that will be met within the Sprint through the implementation of the Product Backlog.
 - Provides guidance to the Development Team on why it is building the Increment.
 - Created during the Sprint Planning meeting.
 - The selected Product Backlog items deliver one coherent function, which can be the Sprint Goal.
 - The Dev Team always keeps the Sprint Goal in mind when working and aim to satisfy the requirements of the Sprint Goal by the end of the Sprint.
- Daily Scrum
 - A 15-minute time-boxed event for the Dev Team.
 - The Daily Scrum is held every day of the Sprint.
 - At it, the Dev Team plans work for the next 24 hours.
 - The meeting is held at the same time and place each day to reduce complexity.
 - **3 important questions should be asked at the Daily Scrum:**
 - What did I do yesterday that helped the Development Team meet the Sprint Goal?
 - What will I do today to help the Development Team meet the Sprint Goal?
 - Do I see any impediment that prevents me or the Development Team from meeting the Sprint Goal?
 - The Daily Scrum is used to inspect the Dev Team's progress toward the Sprint Goal and improve communication. As a result, it optimizes the probability that the Dev Team will meet the Sprint Goal.
 - The Dev Team is responsible for conducting the Daily Scrum.
- Sprint Review
 - A Sprint Review is held at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed.
 - During the Sprint Review, the Scrum Team and stakeholders collaborate about what was done in the Sprint.
 - The focus is on what could be done to optimize value.
 - This is an informal meeting, not a status meeting.
 - At most a 4-hour meeting for 1-month Sprints.
 - The Scrum master ensures that the event takes place and that attendees understand its purpose.
 - The Sprint Review includes the following elements:
 - Attendees include the Scrum Team and key stakeholders invited by the Product Owner;

- The Product Owner explains what Product Backlog items have been “Done” and what has not been “Done”;
- The Development Team discusses what went well during the Sprint, what problems it ran into, and how those problems were solved;
- The Development Team demonstrates the work that it has “Done” and answers questions about the Increment;
- The Product Owner discusses the Product Backlog as it stands. He or she projects likely target and delivery dates based on progress to date (if needed);
- The entire group collaborates on what to do next, so that the Sprint Review provides valuable input to subsequent Sprint Planning;
- Review of how the marketplace or potential use of the product might have changed what is the most valuable thing to do next; and,
- Review of the timeline, budget, potential capabilities, and marketplace for the next anticipated releases of functionality or capability of the product.
- Sprint Retrospective
 - The Sprint Retrospective is an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint.
 - Occurs after the Sprint Review and before the next Sprint Planning.
 - At most a 3-hour meeting for 1-month Sprints.
 - The Scrum Master ensures that the event takes place and that attendants understand its purpose. He/She ensures that the meeting is positive, productive and within the time-box.
 - Purpose of the Sprint Retrospective is to:
 - Inspect how the last Sprint went with regards to people, relationships, process, and tools;
 - Identify and order the major items that went well and potential improvements; and,
 - Create a plan for implementing improvements to the way the Scrum Team does its work.
 - During the meeting, the Scrum Team plans ways to increase product quality by improving work processes or adapting the definition of "Done".
 - The Sprint Retrospective provides a formal opportunity to focus on inspection and adaptation.

3 Scrum Artifacts

Scrum artifacts represent work or value to provide transparency and opportunities for inspection and adaptation.

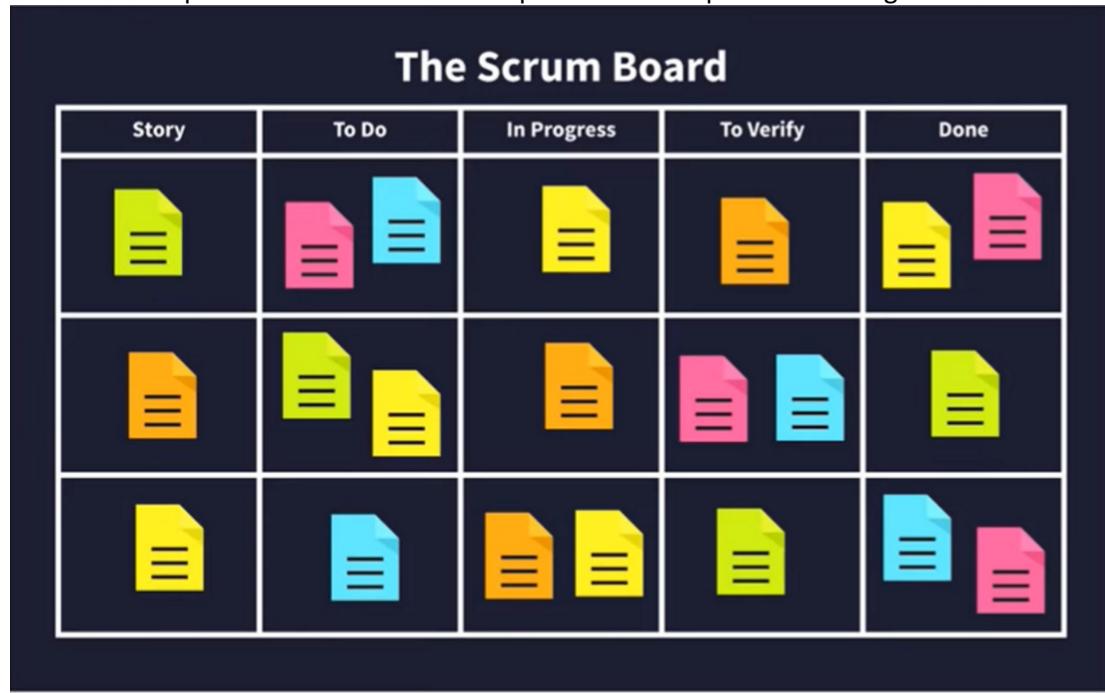
- *Product Backlog*
 - The Product Backlog is an ordered list of everything that is known to be needed in the product. This list of requirements is acquired from user stories of the product written by the user.



- It is the single source of requirements for any changes to be made to the product.
- The Product Owner is responsible for the Product Backlog, including its content, availability,

and ordering.

- The Product Backlog is dynamic; it constantly changes to identify what the product needs to be appropriate, competitive, and useful.
- The Backlog lists all features, functions, requirements, enhancements, and fixes that constitute the changes to be made to the product in future releases.
- Requirements never stop changing, so the Product Backlog is a living artifact.
- The number of items selected from the Product Backlog for the Sprint is solely up to the Development Team.
- Multiple Scrum Teams often work on the same Product Backlog.
- *Product Backlog refinement*: the act of adding detail, estimates, and order to items in the Product Backlog.
- Product Backlog items that can be "Done" by the Dev Team within one Sprint are deemed "Ready" for selection in a Sprint Planning.
- *Sprint Backlog*
 - The selected Product Backlog items for this Sprint + the plan for delivering the product Increment and realizing the Sprint Goal = *Sprint Backlog*
 - The Sprint Backlog makes visible all the work that the Dev Team identifies as necessary to meet the Sprint Goal.
 - It contains enough detail that changes in progress can be understood in the Daily Scrum.
 - The Sprint Backlog shows what work is needed to deliver the functionality into a "Done" Increment.
 - The Dev Team modifies the Sprint Backlog throughout the Sprint. Elements are added or removed as necessary.
 - To ensure continuous improvement, it includes at least one high priority process improvement identified in the previous Retrospective meeting.



- *Increment*

- The Increment is the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints.
- At the end of a Sprint, the new Increment must be "Done" (usable condition).
- An increment is a body of inspectable, done work that supports empiricism at the end of the Sprint.
- The increment is a step toward a vision or goal.

Artifact Transparency

- Maintaining Transparency
 - Scrum relies on transparency.
 - Decisions to optimize value and control risk are made based on the perceived state of the artifacts.
 - If artifacts are not completely transparent, decisions can be flawed, value may diminish and risk may increase.
 - The Scrum Master, Product Owner and Development Team must work together to make the artifacts completely transparent.
 - A Scrum Master can detect incomplete transparency by noticing the difference between expected and real results.
 - It is the Scrum Master's job to work with the Scrum Team to increase the transparency of the artifacts.
 - The path to transparency usually involves learning, convincing, and change.
- Definition of "Done"
 - When a Product Backlog item or an Increment is described as "Done", everyone must understand what "Done" means.
 - Members of the Scrum Team must have a shared understanding of what it means for work to be complete.
 - If there are multiple Scrum Teams, they must mutually define the definition of "Done".
 - Have a clear, stringent definition of "Done" so as to assess when work is complete on the product Increment.
 - Each Increment is additive to all prior Increments and thoroughly tested, ensuring that all Increments work together.

Who, What, When, Purpose (of Scrum Events)

Sprint Planning:

- Who: Entire Scrum Team (optional stakeholders)
- When: At the beginning of the Sprint.
- Duration: 8 hours for a 1-month Sprint.
- Purpose: To identify which items from the Product Backlog to add to the Sprint Backlog. To set a Sprint Goal.

Daily Scrum:

- Who: Dev Team, Scrum Master
- When: Everyday of the Sprint.
- Duration: 15 minutes.
- Purpose: Inspect progress towards the Sprint Goal, adapt Sprint Backlog, improve communications, eliminate other meetings, remove impediments, etc. This is an inspect and adapt meeting.

Sprint Review:

- Who: Scrum Team, Stakeholders
- When: End of the Sprint, but prior to Retrospective.
- Duration: 4 hour meeting for a 1-month Sprint.
- Purpose: Inspect Product Increment & adapt Product Backlog. Elicit feedback and foster collaboration between Scrum Team and Stakeholders.

Sprint Retrospective:

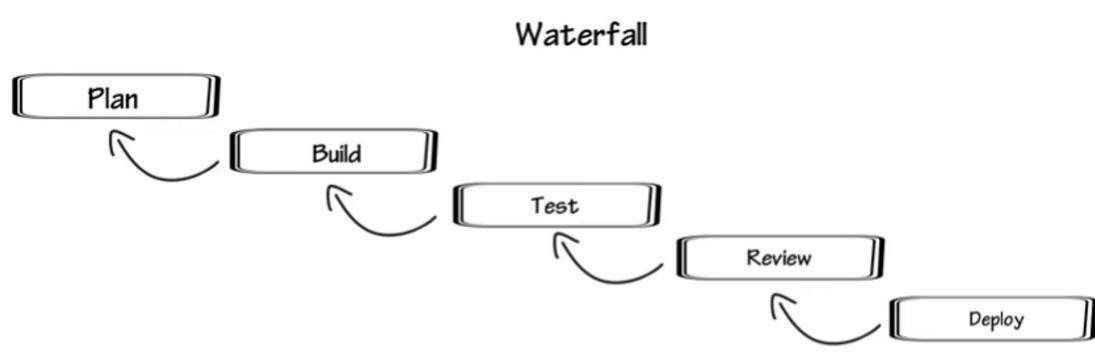
- Who: Scrum Team
- When: After the Sprint Review. Last meeting that happens in the Sprint.

- Duration: 3 hour meeting for a 1-month Sprint.
- Purpose:
 - Inspect how the last Sprint went with regards to people, relationships, process, and tools;
 - Identify and order the major items that went well and potential improvements; and,
 - Create a plan for implementing improvements to the way the Scrum Team does its work.

Increment - Potentially Shippable Product

Continuous Delivery Pipeline

- Continuous Exploration (CE)
- Continuous Integration (CI)
- Continuous Deployment (CD)
- Release on Demand



PROS AND CONS OF WATERFALL

PROS	CONS
1. Clear goals and directions	1. Time consuming
2. Progress can be easily measured	2. Customer dissatisfaction
3. Saves time and money	3. Risk factor

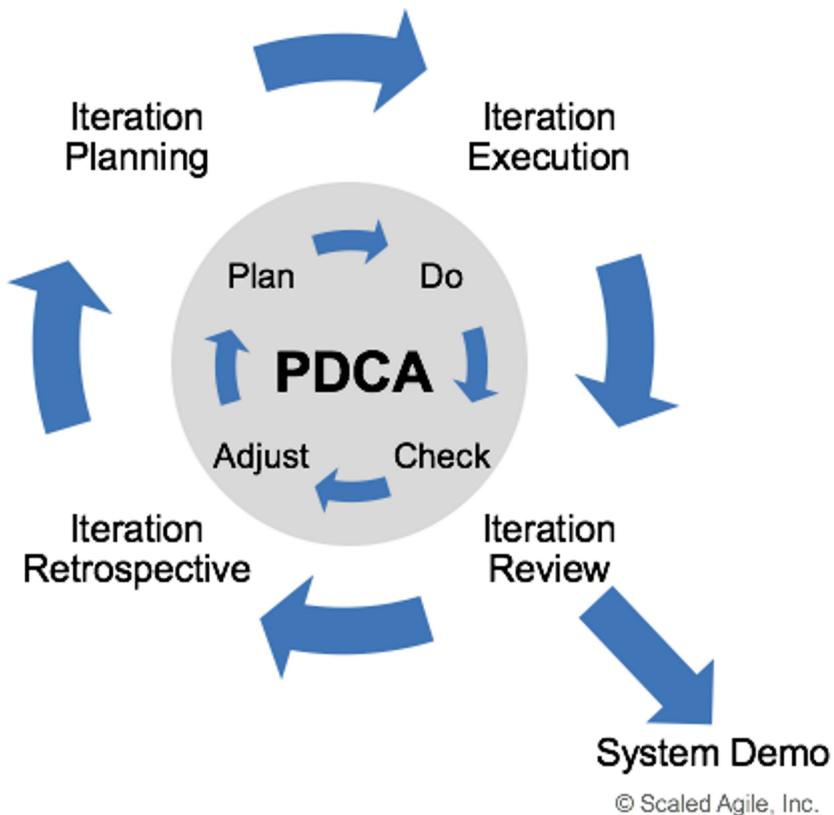
AGILE

edureka!



PROS AND CONS OF AGILE

PROS	CONS
1. High customer involvement	1. Dedicated team
2. Lesser risk	2. Hard to execute
3. Improved quality	3. Documentation can be ignored



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.

Principles behind the Agile Manifesto

We follow these principles:

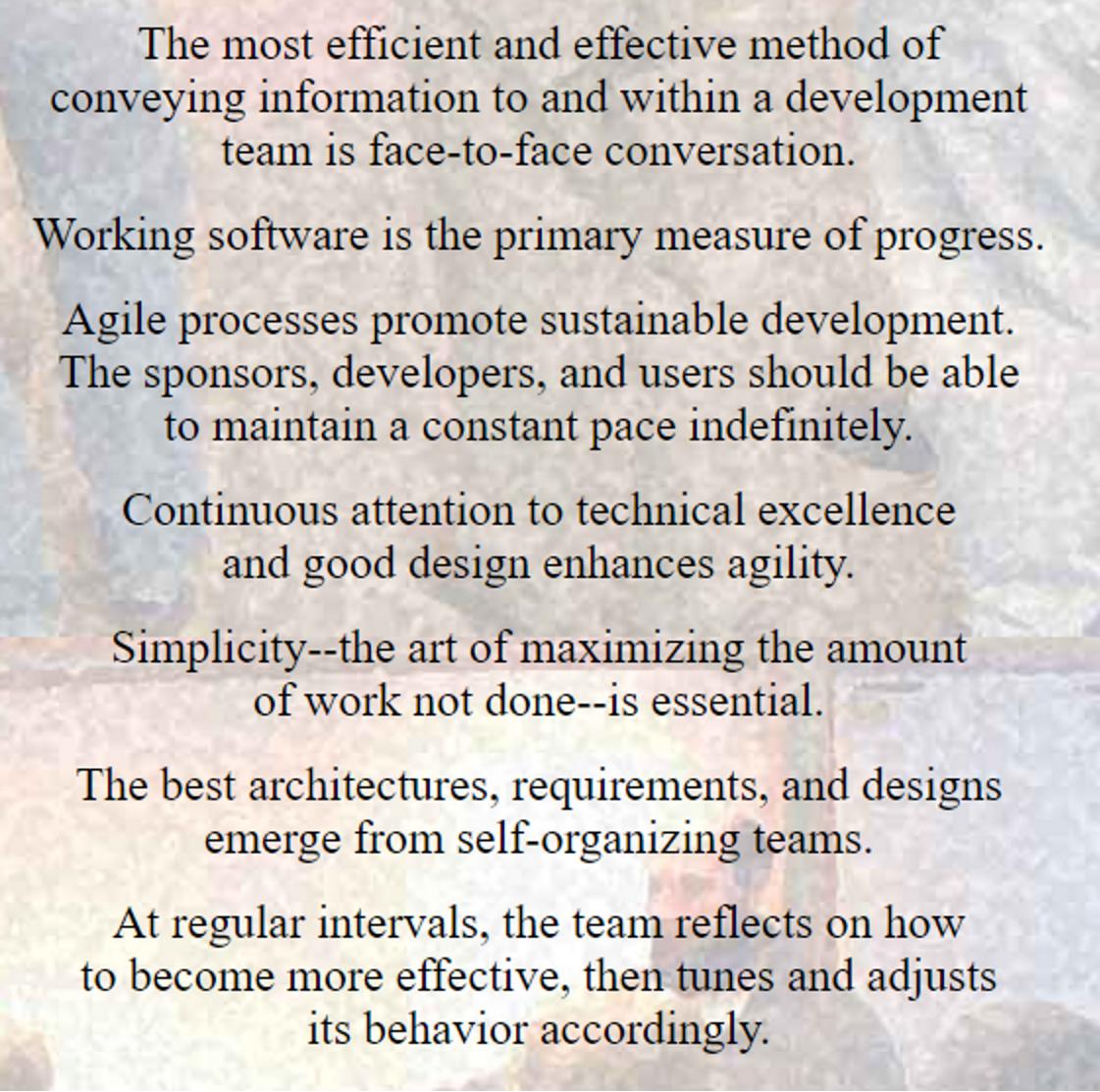
Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Business people and developers must work together daily throughout the project.

Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.



The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Working software is the primary measure of progress.

Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Continuous attention to technical excellence and good design enhances agility.

Simplicity--the art of maximizing the amount of work not done--is essential.

The best architectures, requirements, and designs emerge from self-organizing teams.

At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

References:

- <https://www.scrum.org/>
- <https://www.scrumalliance.org/>
- <https://www.scaledagileframework.com/>
- <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>
- <https://scrumguides.org/scrum-guide.html>
- <https://agilemanifesto.org/>
- <http://agilemanifesto.org/principles.html>