**Scrum Introduction**

Over the years, a major evolution in Scrum and its applications has been witnessed in the IT and Non-IT sectors. This has mandated an end-to-end understanding of the Scrum framework and the associated methodologies.

**What is Scrum and Agile?**

Scrum is a framework for developing, delivering, and sustaining complex products. This Guide contains the definition of Scrum. This definition consists of Scrum’s roles, events, artifacts, and the rules that bind them together.

Agile software development refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.

## What does Scrum stand for?

 “Systematic Customer Resolution Unraveling Meeting”

## Define Scrum - What does Scrum mean?

Scrum (n): A framework within which people can address complex adaptive problems, while

Productively and creatively delivering products of the highest possible value.

Scrum is:

• Lightweight

• Simple to understand

• Difficult to master

## ****What is Scrum Methodology****

A simple framework for working on complex products, Scrum facilitates effective team collaboration. It, therefore, boosts team performance and productivity many times. Very often, Scrum is found to be identified as a ‘**methodology**’. But it is usually recommended to think of it as a framework for managing complex processes.

**An iterative incremental framework**Scrum is an iterative incremental framework for effectively managing product development.

**Iterative:**Scrum is known as an iterative framework because it makes progress towards the goal through successive refinements. The development team takes the first major step in a project. Based on the collected requirements they write the code and are aware that it might be weak in some areas. After this, the team iteratively refines those specific areas until the product is satisfactory. In each successive iteration, further details are added and the software is improved. Usually, the work of each iteration is improved in the upcoming iterations.

**Incremental:**Scrum is called an incremental process because here the software is built and delivered in pieces. Each increment represents a complete subset of the final software to be delivered. In a typical Scrum environment, each increment is fully coded and tested. Simply put, “completed” work is delivered throughout the project.

## Scrum Best Practices

The development team can create quality products by following the Scrum practices mentioned below:

* Define requirement ‘just-in-time’ to keep the product features more relevant.
* Take Product Owner’s/Project Owner's feedback daily.
* Regular Sprint Reviews should happen.
* The Scrum team should arrange an event called Sprint Retrospective to improve how they work.
* Arranging offline meetings to carry out face-to-face conversations.
* Don’t burn out the team members.
* Trust the team members.
* Respect the balance between the team members’ personal and professional lives to ease the work.

### ****Why Do We Use Scrum ?****

At the onset of any project implementation, the concerned team has to make a vital decision- “Which development methodology can be used?” This is a serious matter of discourse. If you are working on any methodology for the first time, a clear definition of the development methodology is required. The team members should follow this definition and all its mandates strictly. This is nothing but the best way to organize the working of software development of all sorts. Clearly note that is NOT just another technical approach or simply a style of project management. Going forward, you will often find these terms being used interchangeably, which is downright wrong!

But why are we discussing this?

Because it works just the same way for Scrum.

One of the long-lived methodologies is the Waterfall model which had dominated software development up until recently. The primary limitation of this model is the wrong “assumption” that all the requirements are clearly understood at the beginning of the project.

The Waterfall model was formally introduced for the first time in the year 1970 by Winston W. Royce who wrote-

“I believe in this concept, but the implementation described above is risky and invites failure.”

**Drawbacks of the waterfall:**

* In the Waterfall model, we can only move to the next phase once we fully complete one phase. Collecting requirements are the most crucial phase for any project. This can be the most challenging, as customers may miss some crucial details. It is difficult from the customer’s point of view as the customer cannot figure out the sketch of the end product from the requirements document.
* Changes made in the business or project requirements are difficult to update due to this fixed approach. Changes can be done, but not without any important corrections in the timeline and overall budget.
* Finally, user acceptance testing and system testing are done only at the completion of the project lifecycle. The impact of bugs that are identified could be significant and difficult to correct without extending the budget and project timeline.

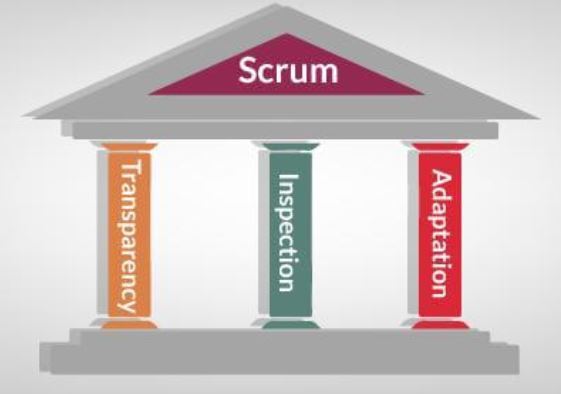
**To overcome these drawbacks in the waterfall development model, we adopt Scrum.**

Scrum is one of the most popular Agile methodologies evolving in all aspects. It is an adaptive, fast, flexible, incremental, iterative, and effective methodology that was designed to deliver outstanding values quickly throughout a project. Scrum is responsible to maintain transparency of communication throughout the continuous progress of a project.

**Benefits of using Scrum:**

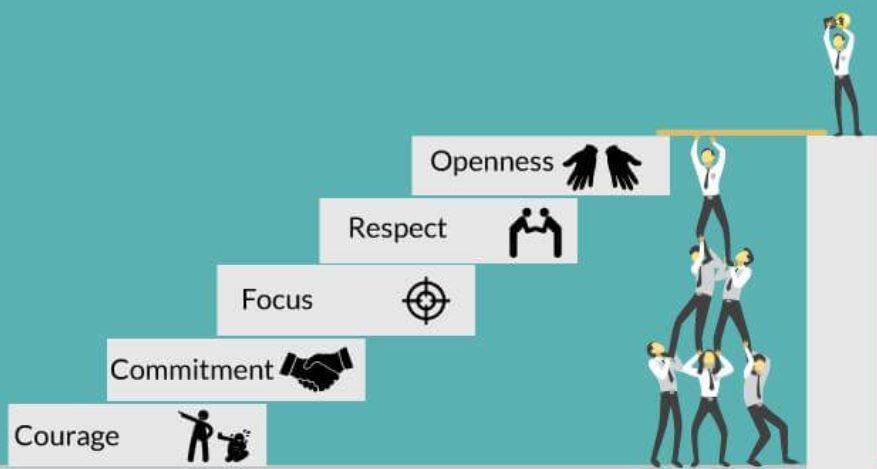
* Scrum makes frequent collaboration among the team members that leads to interpersonal relationships and trust among them.
* Completion of work using the definition of done addresses the development, integration, testing, and documentation with production.
* Conducting daily Scrum retrospective allows the Scrum teams to improve the efficiency of work with Scrum factors.
* Providing quick delivery of software product in short iterations.
* Updating and reviewing according to the client’s requirements.
* Simple in understanding but following the process may be difficult.
* Involving in the sprint review meetings with stakeholder improves the team output.

## THREE PILLARS OF SCRUM



## Transparency, Inspection, and Adaptation are the pillars that underlie the empirical process control theory that’s at the heart of Scrum. “Inspect and adapt” is a slogan on some Scrum teams — it’s what we do with results, failures, processes, market and stakeholder input, and basically everything else.

## The core scrum values upon which the Scrum framework is based on:



**1. Commitment:**

Scrum teams must be committed to progress and willing to have practical objectives and stick to them. This is a team activity where you are a part of a team, and you are accountable to work together and to conform to your commitments.  
  
This value can be expressed in three ways-

* **Sprint-based commitment-**Sprint reflects the realistic goals and a short time duration to achieve these goals. Thus, the team has to be committed to their tasks in order to achieve the Sprint goals.
* **Commitment as a team-**As a team, you need to welcome the changes and represent the adaptability. As a team, you can achieve the project goals in smaller chunks. In case of any issue raised, the team can gather and discuss with each other to come up with a concrete solution to tackle the issue.
* **Commitment as an individual-**Being an individual in a team, you should contribute as much as you can to achieve the Sprint goal. This represents your commitment as an individual.

### ****2. Focus:****

An iterative-incremental approach and timely delivery in Scrum helps to keep us stay focussed towards the project goal. Once the requirements are clear and the goal is set, the most effective way to attain the goal is to be goal-oriented. This motivates you for delivering faster, better and yield more.

### ****3. Openness:****

The Scrum induction requires transparency and openness. We need to investigate reality with a specific end goal to make sensible adjustments. Team members should be open about their work, progress, what they learned and the issues they are facing.  As a team member, you should be open towards sharing a feedback and gaining from each other.

### ****4. Respect:****

As a part of the Scrum team, you should respect team members, their decisions. As an efficient team member, you should also respect diversity. You need to equally respect your users by resolving their problems. As a responsible team member, you need to completely adhere to the Scrum framework and the associated Scrum roles. As a part of the Scrum team, you should respect each other’s skills, knowledge, and insights.

### ****5. Courage:****

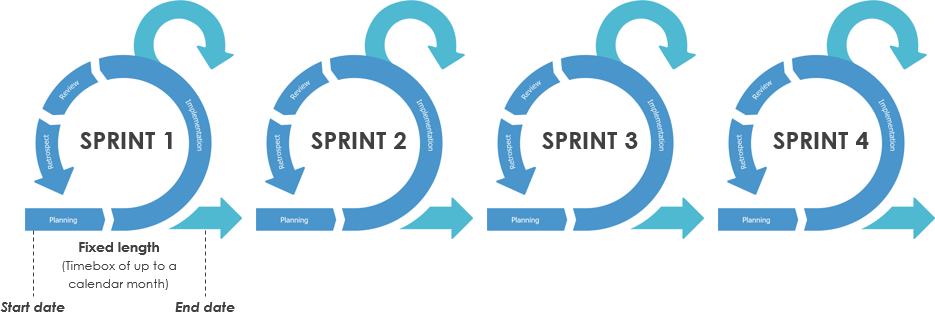
Adaptability to change forms the bedrock of any Scrum project and to accept a change, courage is needed.  
  
Scrum is all about taking risks and finding out an optimized solution. The Scrum team is allowed to think of different approaches to workshop the best and most appropriate solutions. In order to implement new things to the project, we need to explain these new ideas to the team.  
  
By adopting these 5 Scrum values in the project, eventually, you are making your team follow the basic Scrum tenets. Following these values helps in harmonizing the team so they cooperate with each other to create unique ideas for enhanced results.

## SCRUM CEREMONIES

* What Is Sprint?
* Daily Scrum
* Sprint Review
* Sprint Retrospective
* Sprint Planning

**What Is Sprint?**

A **Sprint** is the heart of Scrum with short regular iterations with no longer a bigger time-box of 2-4 weeks. The **Scrum Sprint** goal is to maintain short iterations with enough time between one week to one month to keep the Scrum team focused on potential delivery of a product increment.



Each**Sprint**will be considered as a project with a duration of less than one month. Like projects, Sprints are used to achieve a successful product. Every sprint requires to reach the goal with guidelines for planning, design, and what needs to be built to achieve the resultant increment. We have a limited capacity of Sprints to be done in a one-month calendar. If the sprint’s horizon is too long, we may have to design requirements in bulk, more complexity, and additional time consumption. All of these will increase the chances of risk.

### ****Purpose of the Scrum Sprint:****

**1. Timeboxed:**

From the concept of timeboxing, the Sprint’s are fixed with a characteristic of time-management approach that helps to manage the scope and organize the work performance. Each and every sprint takes place to fit in a fixed time frame with specific start and end dates called a timebox.

**2. Establishes a WIP Limit:**

**Timeboxing** is a technique to keep the **limit of WIP** (work in process). This WIP presents an inventory of work that begins but is not finished yet. If we fail to manage it in a proper way, then it leads to serious economic consequences. Timeboxing maintains a **WIP limit** for each sprint.

**3. Forces Prioritization**

Timeboxing forces us to prioritize and perform a short amount of work that helps the team to focus more. This improves our concentration on getting something valuable (project) done quickly.

**4. Demonstrates Progress**

Timeboxing helps in the illustration of relevant improvements by validating and completing the important pieces of work by a known fixed date.

It also helps the team members to know exactly what needs to be completed to deliver all the features.

**5. Avoid Unnecessary Perfections**

Timeboxing helps in avoiding the unnecessary perfections. At one time we all have to spend a lot of effort in trying to get something “perfect” or to do “gold cover plating”

**6. Motivates Closure**

Timeboxing also motivates to complete the sprint within a fixed time. It is a known fact that at the end of a sprint it brings a tough deadline that will motivate team members to contribute themselves to complete the work on time.

**7. Improves Predictability**

Timeboxing enhances predictability. Even though we can’t forecast exactly about the work that can be completed in a year but it is completely reasonable to predict the work what we can complete in the coming short sprints.

**8. Short Duration**

We get more benefits with short-duration sprints as the progress can be tracked within short intervals of time.

**9. Ease of Planning**

Short-duration sprints will be easier to make a plan. It is easier to plan a few weeks of work instead of having six months of work. Also, planning on such short-duration horizons requires fewer efforts when compared to more longer-horizon planning.

**10. Fast Feedback**

The Short-duration sprint generates quick feedback. During each sprint, we create working software environment to have the possibility to adapt and inspect what we are doing in building the product.

## ****Sprint cancellation:****

The Sprint cancellations consume more resources and time, potential rework and also, everyone should make a regroup in planning to start newsprint.

* The Sprint cancellation can be done before the time-box is completed
* If all the tasks are completed in the Sprint, it reaches the Sprint goal
* The Sprint gets canceled when it does not have a fixed length
* It can also be done when the Sprint goal becomes obsolete.

## ****Output:****

At the end of the sprint, the team submits the completed work to the project mentor and the project mentor verifies the criteria established at the sprint planning meeting to either accept or reject the work. Sprint goals can be achieved by predicting the inspection and adaptation to get an idea of the progress.

**2. Daily Scrum:**

Daily Scrum is also called the Daily Stand-up meeting. The purpose of the daily Scrum is to know the answers to the three questions about what one did yesterday, is doing today, and what the obstacles are.

**3. Sprint Review meeting:**

This meeting is held once each Sprint is finished. In this meeting, the team demonstrates the work that they have produced during the Sprint. This meeting can be attended by the team members, the Scrum Master, the Project Owner, and the Stakeholder.

**4. Sprint Retrospective:**

This is the last event that wraps up a Sprint cycle. It is held for the teams to share their knowledge earned to enhance the team’s interaction for upcoming Sprint cycles.

**5. Sprint Planning Meeting:**

In this meeting, the objectives of the forthcoming Sprint are defined. This meeting is held towards the start of each Sprint. The Project Owner, Scrum Master, and the team members can be a part of the Sprint Planning meeting.

## ****Scrum Rules****

Presenting a list of Scrum rules, that need to be followed within a Scrum frameworkin software development:

### ****Sprint related rules:****

* + Sprint length should be long enough to deliver meaningful chunks of work and short enough to ease the planning process.
  + Every Sprint includes Sprint Planning and Sprint Planning meeting must be time-boxed to 2 hours or a week of Sprint length.
  + Every Sprint is of the same length.
  + Every Sprint should be of 4 weeks or less in duration.
  + Potentially shippable product’ must be an outcome at the end of each Sprint.

### ****Project Backlog related rules:****

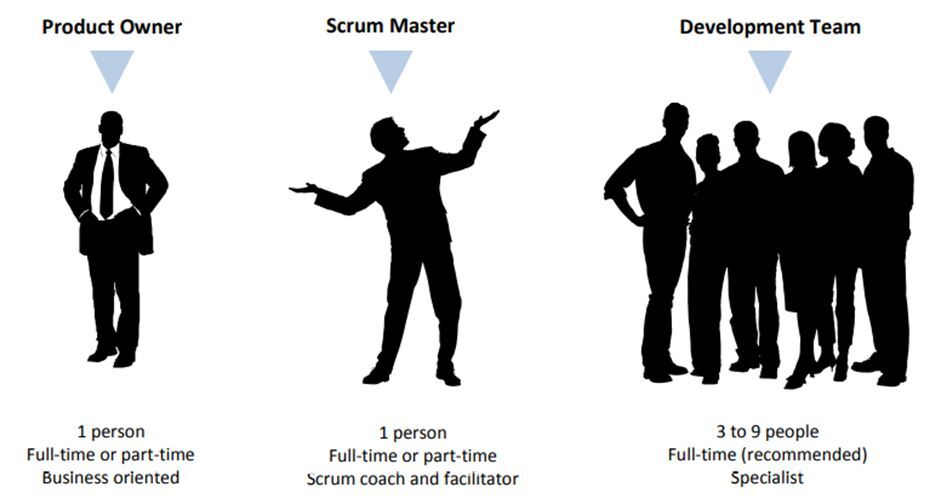
* + All the Project Backlog items (PBI) mentioned in the Project should be related to the same project.
  + Two PBIs can’t have the same position in the Project Backlog.
  + PBIs are conveyed in the form of user stories.

### ****Scrum rules related to the team members role:****

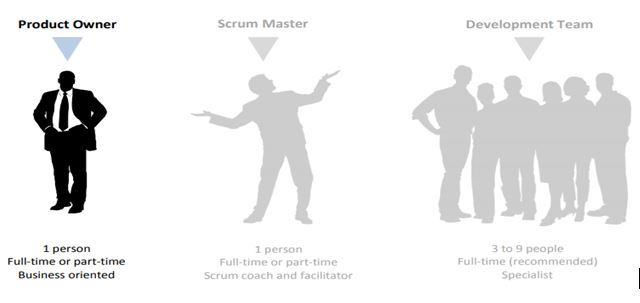
* + As a team member, one should not miss any Scrum Events.
  + As a team member, he/she should work collaboratively to follow the ‘Definition of Done’.

### ****Scrum rules related to the Scrum Master (SM) role:****

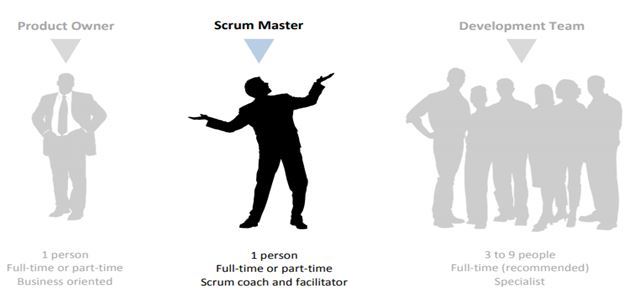
* + As a Scrum Master, he or she has authority on following the right way to implement the Scrum process.
  + Scrum Master has to ensure the timeboxes within a team.
* **SCRUM ROLES**



* **Role 1: The Product Owner**

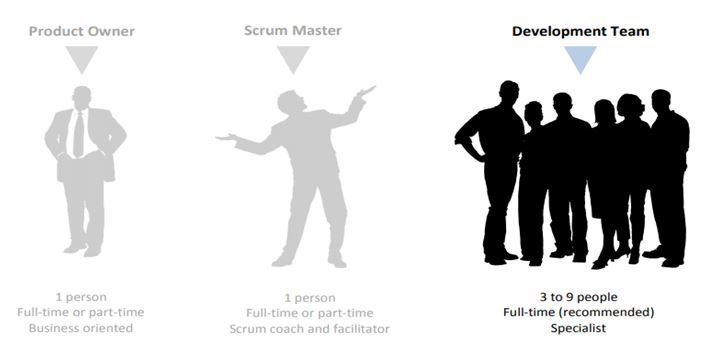


* The Product Owner is responsible for maximizing the value of the product resulting from work of the Development Team. How this is done may vary widely across organizations, Scrum Teams, and individuals.
* The Product Owner is the sole person responsible for managing the Product Backlog. Product Backlog 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.
* The Product Owner may do the above work, or have the Development Team do it.
* However, the Product Owner remains accountable. The Product Owner is one person, not a committee. The Product Owner may represent the desires of a committee in the Product Backlog, but those wanting to change a Product Backlog item’s priority must address the Product Owner.
* **Role 2: The Scrum Master**



* Scrum Masters are those who fully understand Scrum, and help the Scrum Team by coaching them, and ensuring that all Scrum processes are implemented correctly.
* The Scrum Master is a management position, which manages the Scrum process, rather than the Scrum Team. He/she is a servant-leader for the Scrum Team.
* Besides ensuring that the Development Team understands and uses Scrum correctly, the Scrum Master also tries to remove impediments to the Development Team, facilitates their events, and trains or coaches them.
* The Scrum Masters help the Product Owners too, by helping or consulting them on finding techniques, communicating information, and facilitating related events.
* The responsibilities of the Scrum Masters are not limited to the Scrum Team. They should also help those outside the Scrum Team understand the appropriate interactions with the Scrum Team to maximize the value created by the Scrum Team.
* The Scrum Master usually leads the organization in its effort to adopt Scrum. It is possible for a single person to be both Scrum Master, and a member of the Development Team, although this is not recommended. Being a Scrum Master of a project might not occupy 100% of the time of a person; in this case, the best solution is to assign that same person as the Scrum Master in more than one project, rather than making them a member of the Development Team.

**Role 3: The Development Team**



Members of the Development Team are application area experts that are responsible for delivering backlog items, and managing their own efforts.

* They should be cross-functional; being capable of doing the A to Z of the creation of each Product Backlog item. They should be self-organized; find their own way instead of receiving orders. They should be aligned with the goal of the project instead of working blindly.
* A task might be assigned to a single member throughout the Sprint, but the whole Development Team will be responsible and accountable for that task; no individual owns any task.
* The Development Team delivers the final product of the project in step by step Increments, as defined in the Product Backlog. They always work in a product-based way.
* It is highly recommended for members of the Development Team to work full-time in a single project, to stay focused and agile. The composition of the Development Team should not change so often. If there is a need to change team members, then this change should not happen during a Sprint and there will be a short-term decrease in productivity when the composition of the team changes.

## Scrum Master Service to the Product Owner

The Scrum Master serves the Product Owner in several ways, including:

• 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,

## Scrum Master Service to the Development Team

The Scrum Master serves the Development Team in several ways, including:

• 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.

## Scrum Master Service to the Organization

The Scrum Master serves the organization in several ways, including:

• 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 Artifacts

**Scrum Artifacts Overview**

The term “Artifact” in archaeology refers to an object made by a human being. The word artifact roughly translates to ‘The Work of Art’. So, artifacts are nothing but the documents we make either in a tool to get a solution to our problem or art of work that inspires to resolve.

**What is Scrum Artifact?**

The **Scrum Artifacts** provide the key information to the Scrum team members and Stakeholders regarding how the product is developed and activities to be done, and what activities are being planned in the project.

### ****1. The Product Backlog****

The Product Backlog is arranged in an ordered list of prioritized features that are needed as a part of the end product. It is the main source of requirements to update or include that gets reflected on the product. The Product Backlog is a highly visible artifact at the heart of the Scrum framework that is accessible for all the projects.

### ****2. The Sprint Backlog****

The Sprint Backlog consists of a list of tasks that the Scrum team has to accomplish by the end of the Sprint. During the Sprint planning meeting, the Scrum team selects priority items set by the Product Owner from the product backlog list items, usually in the form of user stories. The Sprint Backlog is a planned process contains complete information that helps to understand the changes done in the development clearly during the Daily Scrum. These Sprint Backlog items are modified by the development team during the Sprint.

### ****3. The Product Increment****

The word “Increment” itself describes its essence to increase to the next level. The Increment is a step towards a goal or vision. The Product Increment is composed of a list of Product Backlog items completed during the Sprint and also with the previous Sprints. By the end of Sprint, each backlog item must be completed in conformance with the Scrum team specified by the definition-of-done checklist.

### ****Artifact 4: Definition of “Done”****

When a Product Backlog item or an Increment is described as “Done”, everyone must understand what “Done” means. Although this may vary significantly per Scrum Team, members must have a shared understanding of what it means for work to be complete, to ensure transparency. This is the definition of “Done” for the Scrum Team and is used to assess when work is complete on the product Increment.

### ****Artifact 5: Monitoring Progress toward a Goal****

At any point in time, the total work remaining to reach a goal can be summed. The Product Owner tracks this total work remaining at least every Sprint Review. The Product Owner compares this amount with work remaining at previous Sprint Reviews to assess progress toward completing projected work by the desired time for the goal. This information is made transparent to all stakeholders.

### 

### ****Artifact 6: Monitoring Sprint Progress****

At any point in time in a Sprint, the total work remaining in the Sprint Backlog can be summed. The Development Team tracks this total work remaining at least for every Daily Scrum to project the likelihood of achieving the Sprint Goal. By tracking the remaining work throughout the Sprint, the Development Team can manage its progress.

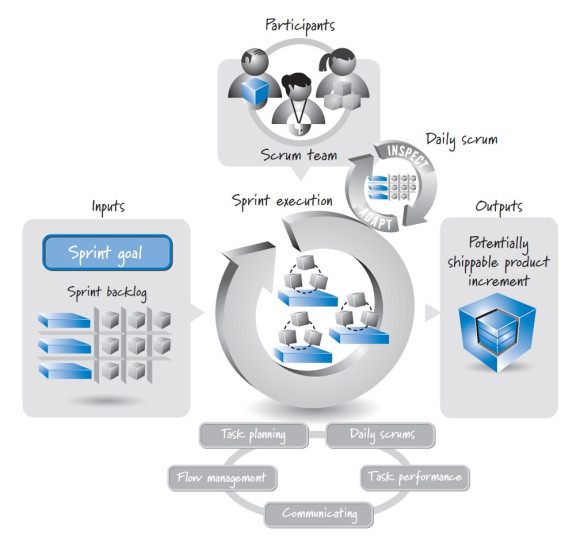
## New imageSprint Execution

Sprint Execution is performed during each Sprint by the Scrum team to meet the Sprint goal. This tutorial centers around the standards and practices that guide how the Scrum team oversees, plans, performs, and discusses during Sprint execution

### 

### ****How Does Sprint Execution work?****

Sprint goal and Sprint Backlog are the inputs to the Sprint Execution process. These inputs are generated during a Sprint Planning. Sprint Execution process includes task planning, performing, managing the tasks, attending daily stand-ups, and communicating with the Scrum teams. The outcome of Sprint Execution is a potentially shippable product increment, formed from a list of product backlog items by meeting the team members’ definition of done.



### ****1. Sprint Execution Planning:****

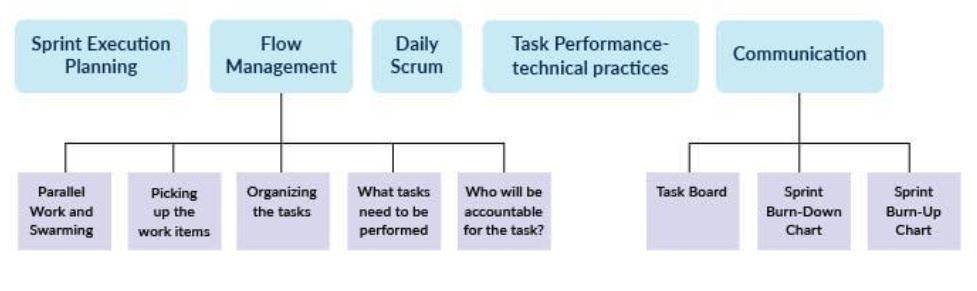
During sprint planning, the team creates a prioritised plan, called Sprint Backlog for achieving the Sprint goal. While task planning, team members perform ‘just-in-time’ task-level planning to increase the task performance, if needed.

### ****2. Flow Management****

Basically, a team should be able to manage the workflow throughout the Sprint to fulfil a Sprint goal. The workflow management includes the decisions on the amount of work the team can perform parallelly, from which task to start, organizing the tasks, which work to perform, and who will be accountable for the tasks. Let’s dive deeper into these.

**➢Parallel Work and Swarming-**  
  
The team must be self-organized to decide how many tasks the members can perform parallely. Because, working on too many tasks at a time can overburden a team and working on just one item at a time is time-consuming. The team should be balanced enough to perform all the tasks, restricting the T-shaped skills from burn-out. This way will result in not only reducing the time but also delivering a maximum value during a Sprint.  
Swarming is accomplishing one unfinished item at a time with available capacity, instead of switching on to new items. Finally, teams have to identify the most important items to deliver the maximum value at the end of each Sprint.

**➢Picking up the work items-**  
  
The easiest way to pick up the work items is to choose the highest priority item specified by the Product Owner (PO). Unfortunately, in some cases it doesn’t work. Because, dependencies or the team members’ skills might be in a different order. The development team can choose the work items appropriately.  
  
**➢Organizing the tasks-**  
  
The amount of work can be picked up from a value and delivery-focused work items. The developer and tester pair up and work in a highly interleaved manner with rapidly creating test cycles, codes, test execution, and test and code refinement. This approach helps in uninterrupted workflow, supports fast feedback, and enables skilled team members to get the work done by organizing the tasks.  
  
**➢What tasks need to be performed-**  
  
The Scrum teams are self-organized teams. They decide their tasks to complete the Product backlog items. The Product Owners and Stakeholders define the scope of the feature and create the acceptance criteria. The team and the Product Owner work collaboratively to ensure that the technical decisions are made in an economically sensible way.  
 **➢Who will be accountable for the task?**  
  
A team member who can get the task done properly and correctly is the best to handle the task. In case that team member is not available, the team itself should be able to choose the next best person.



### ****3. Daily Scrum:****

Daily Scrum is a 15-minute ‘inspect and adapt’ activity, that takes place anywhere in the workplace and once in every 24 hours. The aim of this meeting is to help the teams achieve the Sprint goal faster. Also, Daily Scrum shares the big picture of the current status of the Sprint, how much to work, which work items to start working on, which is the best way to follow, and how to organize the work among the team members. The Daily Scrum is crucial for the flow management.

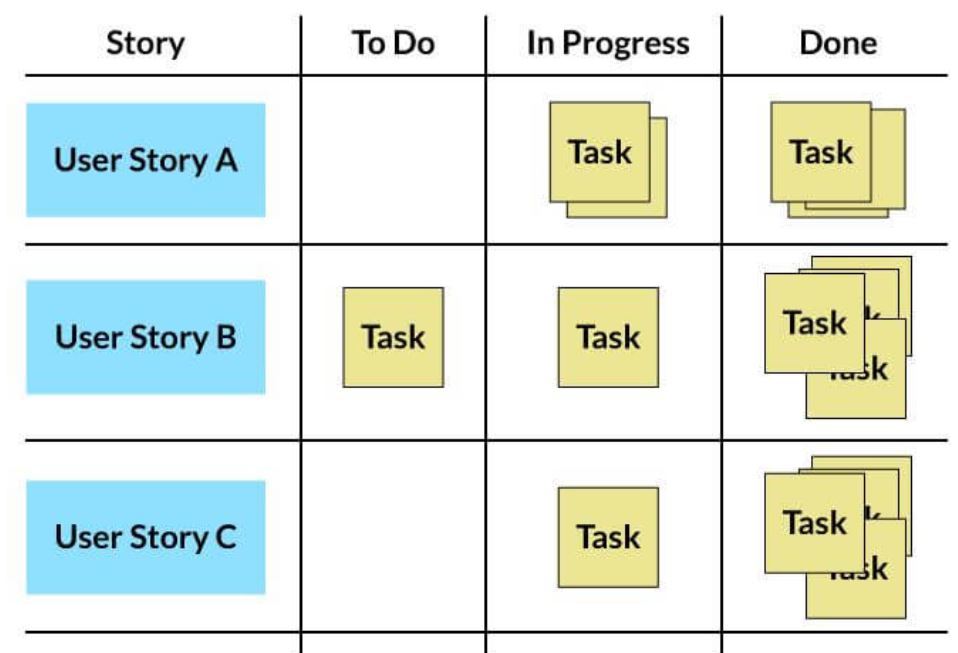
### ****4. Task Performance- technical practices:****

The development team should have technical knowledge of what they perform. For example, if you are implementing Scrum in software development, then the team members must be good at the programming languages, technical practices like continuous integration, automated testing, refactoring, test-driven development, and so on. Teams with weak technical skills generally fail to achieve the long-term benefits of the Scrum methodology.

**5. Communication:**

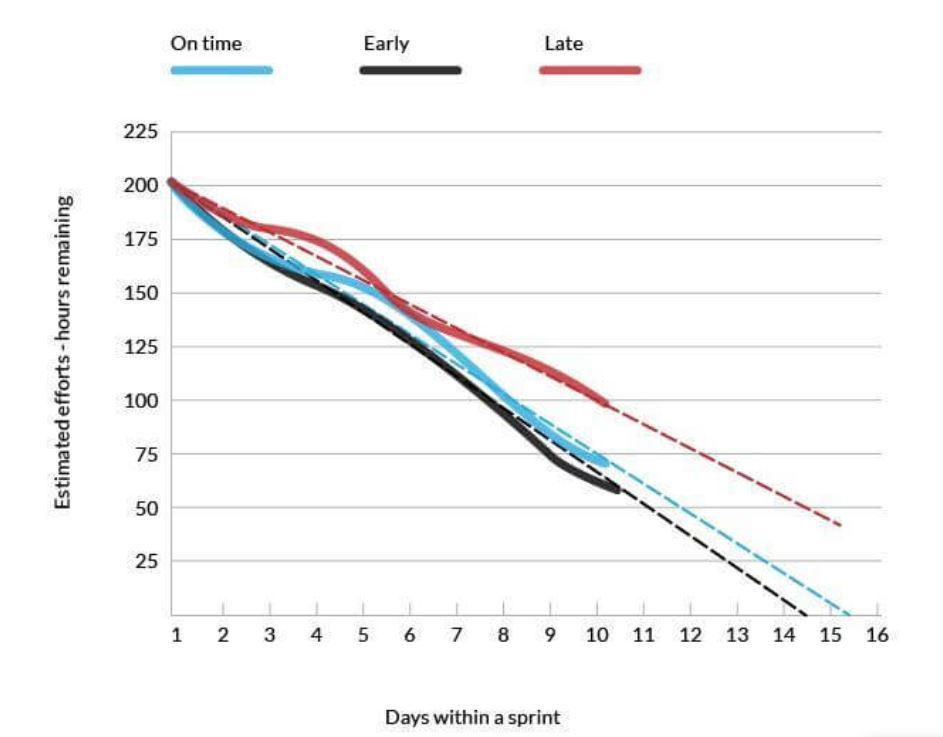
Communication is necessary to effectively communicate a progress within a team. Communication can be done in 3 ways:

**➢Task Board**



The task board eases the communication on the sprint progress within the team members. The task board represents the user stories, a list of tasks. The tasks remain in the ‘To-do’ column till the team starts working on them. Once the team starts working on the tasks, that task is mentioned under the ‘In-progress’ column, and eventually to the ‘done’ column. The team can get the status of work easily just by glancing at the board.

**➢Sprint Burn-Down Chart**  
  
The sprint burndown chart helps in tracking a progress and predicting when the work will be completed. The chart tracks how many hours of efforts are remaining to accomplish each task per day.  
  
The team updates its ‘in-progress’ tasks with an estimate of how much work is remaining, each day. It adds these task hours to the total number of estimated hours for any tasks that haven't started yet and plots the results on the sprint burndown chart, as given below.



**➢Sprint Burn-Up Chart**  
  
The Sprint Burn-up chart is the other way to track a progress through a Sprint. These charts are plotted using user stories. Each day, the sum of completed product backlog items to date, as measured in story points, is charted. Burn up charts allow team members to see how product backlog items are flowing during the sprint.

