**Agile**

Put simply, **Agile** development is an approach to managing software development teams and projects. Agile is a time boxed, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver it all at once near the end. These incremental steps are known as **sprints**. It is an alternative to more traditional sequential development strategies that helps teams respond better to unpredictable occurrences.

**Agile** refers to a set of “methods and practices based on the values and principles expressed in the Agile Manifesto,” which includes things like collaboration, self-organization, and cross functionality of teams.

The [Agile Manifesto](https://agilemanifesto.org/) placed a new emphasis on communication and collaboration, functioning software, team self organization, and the flexibility to adapt to emerging business realities.

**10 Key Principles of Agile**

1. Active user involvement is imperative  
2. The team must be empowered to make decisions  
3. Requirements evolve but the timescale is fixed  
4. Capture requirements at a high level; lightweight & visual  
5. Develop small, incremental releases and iterate  
6. Focus on frequent delivery of products  
7.  Complete each feature before moving on to the next  
8. Apply the 80/20 rule (80% of results will come from just 20% of the action)  
9. Testing is integrated throughout the project lifecycle – test early and often  
10. A collaborative & cooperative approach between all stakeholders is essential

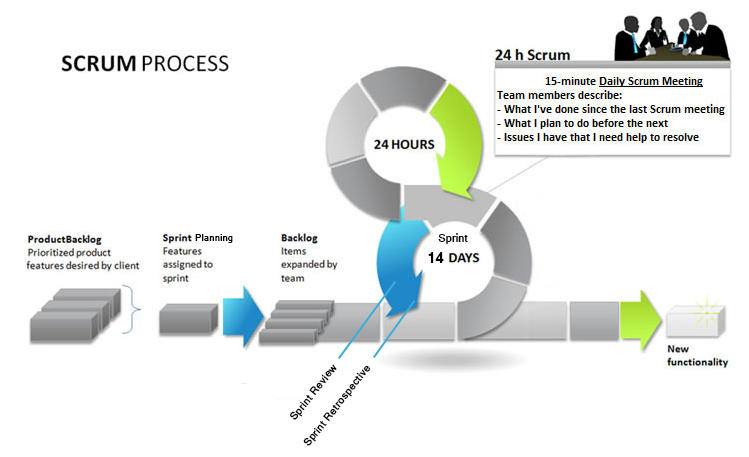
The Agile Manifesto doesn’t provide concrete steps. Organizations usually seek more specific methods within the Agile movement.

**Scrum**

**Scrum** is also an agile development method, which concentrates particularly on how to manage tasks within a team-based development environment.  Scrum is the most popular and widely adopted agile method. Scrum is a framework that is used to implement Agile development.

Scrum emphasizes decision making from real-world results rather than speculation. Time is divided into short work cadences, known as sprints, typically one week or two weeks long. The product is kept in a potentially shippable (properly integrated and tested) state at all times. At the end of each sprint, stakeholders and team members meet to see a demonstrated potentially shippable product increment and plan its next steps.

Scrum is a simple set of roles, responsibilities, and meetings that never change. By removing unnecessary unpredictability, we’re better able to cope with the necessary unpredictability of continuous discovery and learning.



**Scrum Roles**

* **Product Owner:** The Product Owner should be a person with vision, authority, and availability. The Product Owner is responsible for continuously communicating the vision and priorities to the development team from the stakeholders (people who have hired your company to develop). Because Scrum values self-organization among teams, a Product Owner must fight the urge to micro-manage. At the same time, Product Owners must be available to answer questions from the team.
* **Scrum Master:** The Scrum Master acts as a facilitator for the Product Owner and the team. The Scrum Master does not manage the team. The Scrum Master works to remove any impediments that are obstructing the team from achieving its sprint goals. This helps the team remain creative and productive while making sure its successes are visible to the Product Owner. The Scrum Master also works to advise the Product Owner about how to maximize efforts for the team.
* **Team:** According to Scrum’s founder, “the team is utterly self managing.” The development team is responsible for self organizing to complete work. A Scrum development team contains about seven fully dedicated members (officially 3-9), ideally in one team room protected from outside distractions. For software projects, a typical team includes a mix of software engineers, architects, programmers, analysts, QA experts, testers, and UI designers. Each sprint, the team is responsible for determining how it will accomplish the work to be completed. The team has autonomy and responsibility to meet the goals of the sprint.

To use a ship as an example of the scrum roles: the **captain is the Product Owner** that steers the direction and keeps the boat on track. The **Team is the ship itself** with all of its components. The **Scrum Master is the mechanic** that ensures that all parts of the ship have the ability to work effectively.

**Scrum Overview**

<https://www.mountaingoatsoftware.com/agile/scrum>

**Scrum Terminology**

<https://www.mountaingoatsoftware.com/agile/scrum/overview>

**Scrum Board**

Breaks down the features into individual tasks (post-its) and assigns them to 3 main areas: **Not Started** on left, **In Progress** in the middle and **Complete** on the right.



**Mad, Sad, Glad**

Mad, Sad, Glad is a technique used in the sprint retrospective stage where the team is asked to compile their emotions on the previous sprint into post-its on a board. The idea being that reflecting on the positive and negative aspects of the past will improve the processes of the future sprints.

<https://www.retrium.com/resources/techniques/mad-sad-glad>