

Common Agile Terms & Definitions

Term	Description	Additional Details
Agile	Agile is a descriptor of any system, process, framework, or technique used to organize work and develop functionality that reflects agile values and principles as defined in the agile manifesto.	See http://agilemanifesto.org
Automated testing	The use of software (separate from the functionality being tested) to replace the manual process of executing tests and comparing test results with predicted results.	Teams can work at a more rapid and sustainable pace when tests are automated rather than manual.
Burndown chart	A burndown chart is a graphical display illustrating how much work is remaining in a sprint or a release.	Burndown charts display remaining story points for sprints and releases, but also show remaining hours for sprints.
Continuous delivery	A software development approach for producing releasable functionality in short cycles so it may be released at any time to the end customer.	
Continuous deployment	A software development practice of releasing new code into production as it is developed, integrated and tested rather than after the sprint has completed.	
Continuous integration	A software development practice of merging and testing new code with the existing code base continually as it is checked in by developers, rather than at the end of a development cycle.	
Cross-functional team	A group of people who collectively possess all the skills necessary to plan, execute and deploy an idea of functionality to market.	
Daily scrum	The daily scrum is a 15-minute daily meeting for inspecting progress toward the sprint goal, coordinating the priorities for the day, and raising impediments and risks.	The developers decide the structure. The scrum master coordinates on impediments. The product owner provides support.
Definition of done	Formal description of quality measures required to have a releasable increment at the end of each sprint. It is the scrum team's agreement and commitment to the increment.	May be different for sprints vs. releases. Product backlog items that do not meet the definition of done are not released or presented at the sprint review.
Developer	A scrum team member who can do at least one skill to contribute to achieving the product goal. Developers on a cross-functional scrum team are expert in at least one skill but can contribute many skills to the team.	"T-shaped" is often used to describe a developer who has depth in at least one skill, but breadth across all skills required for the team to develop a product. "M-shaped" developers have depth in more than one skill.

Common Agile Terms & Definitions

Term	Description	Additional Details
Epic	An epic is a very large user story containing multiple channels of value of a given feature of functionality, and is eventually broken down into smaller stories. Epics are often used as placeholders for new ideas that have not been thought out fully.	At the release level, product backlog items should be sized as an epic or smaller. At Platinum Edge, we consider a requirement with a development team estimate of (Fibonacci) 13, 21 or 34 to be an epic, and an appropriate size to be pulled into a release.
Feature	Features are high-level product backlog items that get broken down into epics and user stories as they increase in priority.	Features are high level and sized appropriately for the roadmap level. At Platinum Edge, we consider a requirement with a development team estimate of (Fibonacci) 55, 89 or 144 to be a feature, and an appropriate size to be on the product roadmap.
Impediment	Impediments are anything that prevents the scrum team from meeting their potential. If organizational, it is the scrum master's responsibility to eliminate and prevent it. If it is internal to the team, then they themselves work together to remove it.	Also known as a roadblock.
Product	A product is a vehicle to deliver value. It has a clear boundary, known stakeholders, well-defined users or customers. A product could be a service, a physical product, or something more abstract.	Other definitions of product may exist, but this is the Scrum Guide's definition.
Product backlog	The product backlog is an ordered list of what is needed to create the product (requirements, maintenance items, team improvements). The product goal is the scrum team's commitment. The product backlog emerges and defines the product goal.	The product owner owns the product backlog. Anyone may add items to it, but the product owner decides the priority. Product backlog items usually include: 1) description, 2) order, 3) estimate, 4) value
Product goal	The product goal is the long-term objective for the scrum team and describes a future state of the product which can serve as a target for the scrum team. The scrum team commits to the product goal to ensure transparency and focus against which progress can be measured.	The product goal is in the product backlog. The rest of the product backlog emerges to define "what" will fulfill the product goal.
Product owner	The product owner sets the direction for the product, organizes the product backlog, accepts or rejects work results throughout the sprint and adjusts features and priorities for future sprints as necessary.	The product owner is a dedicated peer level member of the scrum team.
Product roadmap	Holistic, yet digestible view of the features that will enable the product goal.	The product roadmap is the first "cut" of the product backlog.

Common Agile Terms & Definitions

Term	Description	Additional Details
Product team	The product team consists of: <ul style="list-style-type: none"> • Scrum team (product owner, developers, scrum master) • Stakeholders (including sponsor) 	
Project	A temporary endeavor undertaken to create a unique product, service or result. A project is temporary in that it has a defined beginning and end in time, and therefore defined scope and resources.	Other definitions of project may exist, but this is the Project Management Institute's definition.
Relative estimation	The effort associated with requirements relative to the estimates given for other requirements. Two common techniques for relative estimation are affinity estimation and estimation poker.	Common practice is to use relative estimations (e.g. story points), rather than absolute estimations (e.g. hours) for long-range planning.
Release	The transition of an increment of potentially shippable product from the developers to the customer for use. Releases typically happen when one or more sprints have resulted in the product having enough value to deliver it to the customer.	Scrum teams using continuous delivery release functionality to the customer continuously, as much as multiple times every day.
Requirement	Requirements are any item on the product backlog that describes work the developers need to do to deliver valuable functionality to the end customer.	Requirements may be broken down by features or epics and may also follow the user stories pattern.
Scrum	A lightweight, agile framework for 1) organizing work so the development organization knows whether their processes are working for them, and 2) exposing progress on specific functionality as it is developed.	Scrum is founded on empirical process control and lean thinking.
Scrum artifacts	There are 3 scrum artifacts that represent work or value, and maximize transparency to enable inspection and adaptation: <ul style="list-style-type: none"> • Product Backlog • Sprint Backlog • Increment 	Each artifact contains a commitment to ensure it provides information that enhances transparency and focus against which progress can be measured: <ul style="list-style-type: none"> • Product Backlog > Product Goal • Sprint Backlog > Sprint Goal • Increment > Definition of Done
Scrum events	There are 5 scrum events that are opportunities to inspect and adapt scrum artifacts: <ul style="list-style-type: none"> • Sprint • Sprint planning • Daily scrum • Sprint review • Sprint retrospective 	Optimally, all events are held at the same time and place to reduce complexity. Timeboxing each event can be helpful to ensure focus and effectiveness. Product planning (product goal & roadmap) and release planning are also commonly used agile artifacts to increase success of product teams.

Common Agile Terms & Definitions

Term	Description	Additional Details
Scrum master	The scrum master is a dedicated scrum team member responsible for making the process run smoothly through 1) coaching product team members and enforcing the rules of scrum, 2) removing obstacles impacting productivity, and 3) protecting the team from external distractions.	Just like an aeronautical engineer removes drag on an aircraft, the scrum master removes organizational drag on the scrum team. Scrum masters are proactive in not only removing impediments, but also in preventing them.
Scrum mentor	Someone experienced with scrum, but not actively working as a member of the scrum team, who works alongside the scrum team to provide in-the-moment course correction while the team is learning and/or improving playing scrum.	Sometimes referred to as a scrum coach or agile coach.
Scrum accountabilities (commonly referred to as roles)	There are 3 scrum accountabilities: <ul style="list-style-type: none"> • Product owner • Developer • Scrum master 	Each member of the scrum team is a peer to the other scrum team members. No one manages anyone else on the scrum team.
Scrum team	The scrum team is a self-managed, self-organized and cross-functional group of people who do the hands-on work of elaborating, designing, developing, documenting, testing and integrating functionality. They are responsible for producing the product and must also have the authority to make decisions about how to perform the work.	The three accountabilities (roles) on a scrum team include product owner, developer and scrum master.
Self-organizing	A model in which the people performing the work determine how the work is organized and allocated.	
Self-managing	A management approach of accountability based on results rather than process compliance whereby team members decide what to work on and how to accomplish their work.	Self-managing teams often have higher levels of accountability because it is at the peer level.
Sprint	Sprints are consistent, fixed-length and timeboxed iterations, the basic development cycles for a product. Sprints are one month or less, usually 1-2 weeks, and sometimes as little as one day.	Sprints should be consistent in length to enable a scrum team to forecast release and product completion based on empirical data.
Sprint backlog	The sprint backlog is the subset of product backlog items selected for the current sprint that are needed to achieve the scrum team's sprint goal commitment, broken down into the tasks necessary to accomplish them.	The developers own the sprint backlog, meaning only developers can add tasks to the sprint backlog to accomplish the sprint goal, but the product owner does not add additional scope through user stories to the sprint backlog.

Common Agile Terms & Definitions

Term	Description	Additional Details
Sprint goal	The single objective for the sprint and creates coherence and focus, encouraging the scrum team to work together rather than on separate initiatives. It is created during sprint planning.	Although the sprint goal is a commitment by the developers, it provides flexibility in terms of the exact work needed to achieve it.
Sprint planning	Sprint planning is when the sprint goal is set, and the product backlog items are selected for the immediate sprint.	
Sprint review	Sprint review is held at the end of the sprint. The product owner reviews the sprint goal, and the developers demonstrate to the stakeholders the increment (releasable functionality) developed during the sprint. The product owner collects feedback from stakeholders and updates the product backlog based on that feedback.	Sprint review is when the product team inspects and adapts the product.
Sprint retrospective	Sprint retrospective is when the scrum team reviews their experience of the sprint, evaluates opportunities for improvement and determines an action plan for implementing improvements. Areas of focus may include, but not limited to, quality practices, environment, communication, collaboration, processes, tools, skills, etc.	Sprint retrospective is when the scrum team inspects and adapts their processes and environment. Only the scrum team attends sprint retrospective. It should be the entire scrum team, including the product owner.
Stakeholders	Stakeholders are anyone who can impact a product or on whom the scrum team relies for support.	Stakeholders may include the sponsor, executives, managers, auditors, legal, security specialists, enterprise architects, support, sales, marketing, etc.
Story points	Story points are used to give relative estimates of effort to product backlog items. Story points are based on the Fibonacci sequence, in which each number is the sum of the two previous numbers. As the numbers progress, the gaps between numbers increase to acknowledge the decrease in accuracy in predicting larger chunks of work.	Fibonacci sequence: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144 At Platinum Edge, requirements with estimates 1-8 can be brought into a sprint, 13-34 can be brought into a release and 55-144 are appropriate for the product roadmap.
Task	Tasks are single small items of work (usually a day or less in duration) that help a particular user story reach completion.	Tasks are only planned for the current sprint (not for future sprints) and are estimated in hours in order to generate the sprint burndown chart.
Task board	A task board is an information radiator displaying the sprint goal, user stories and tasks of a sprint. It shows the status of each sprint backlog item (e.g. to do, doing, accept, done).	The task board is a type of kanban board. The daily scrum meeting is often held around the task board, being visible to everyone.

Common Agile Terms & Definitions

Term	Description	Additional Details
User story	A user story is a pattern used to describe the requirement of a single channel of value of functionality from an end-user perspective. User stories serve the scrum team as a reminder to have collaborative discussions about how to deliver the functionality to the end customer.	Format: As a <user>, I want to <action> so that <benefit>. Acceptance criteria: When I do this <>, this happens <>.
Velocity	A work output measure based on a specific team's performance, usually measured as the sum of all completed user story points in the sprint.	Velocity is not a metric to be used for comparing productivity between teams. It is also not a goal. It is a historical fact used as an input to empirically forecast a product's schedule and cost.