**Definition of “Done”**

When a Product Backlog item or an Increment is described as “Done”, everyone must understand what “Done” means. Although this varies 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.[[1]](#footnote-1)

The same definition guides the Development Team in knowing how many Product Backlog items it can select during a Sprint Planning. The purpose of each Sprint is to deliver Increments of potentially releasable functionality that adhere to the Scrum Team’s current definition of “Done.” [[2]](#footnote-2)

Development Teams deliver an Increment of product functionality every Sprint. This Increment is useable, so a Product Owner may choose to immediately release it. If the definition of "done" for an increment is part of the conventions, standards or guidelines of the development organization, all Scrum Teams must follow it as a minimum. If "done" for an increment is not a convention of the development organization, the Development Team of the Scrum Team must define a definition of “done” appropriate for the product. If there are multiple Scrum Teams working on the system or product release, the developer teams on all of the Scrum Teams must mutually define the definition of “Done.”[[3]](#footnote-3)

Each Increment is additive to all prior Increments and thoroughly tested, ensuring that all Increments work together.[[4]](#footnote-4)

As Scrum Teams mature, it is expected that their definitions of “Done” will expand to include more stringent criteria for higher quality. Any one product or system should have a definition of “Done” that is a standard for any work done on it.[[5]](#footnote-5)

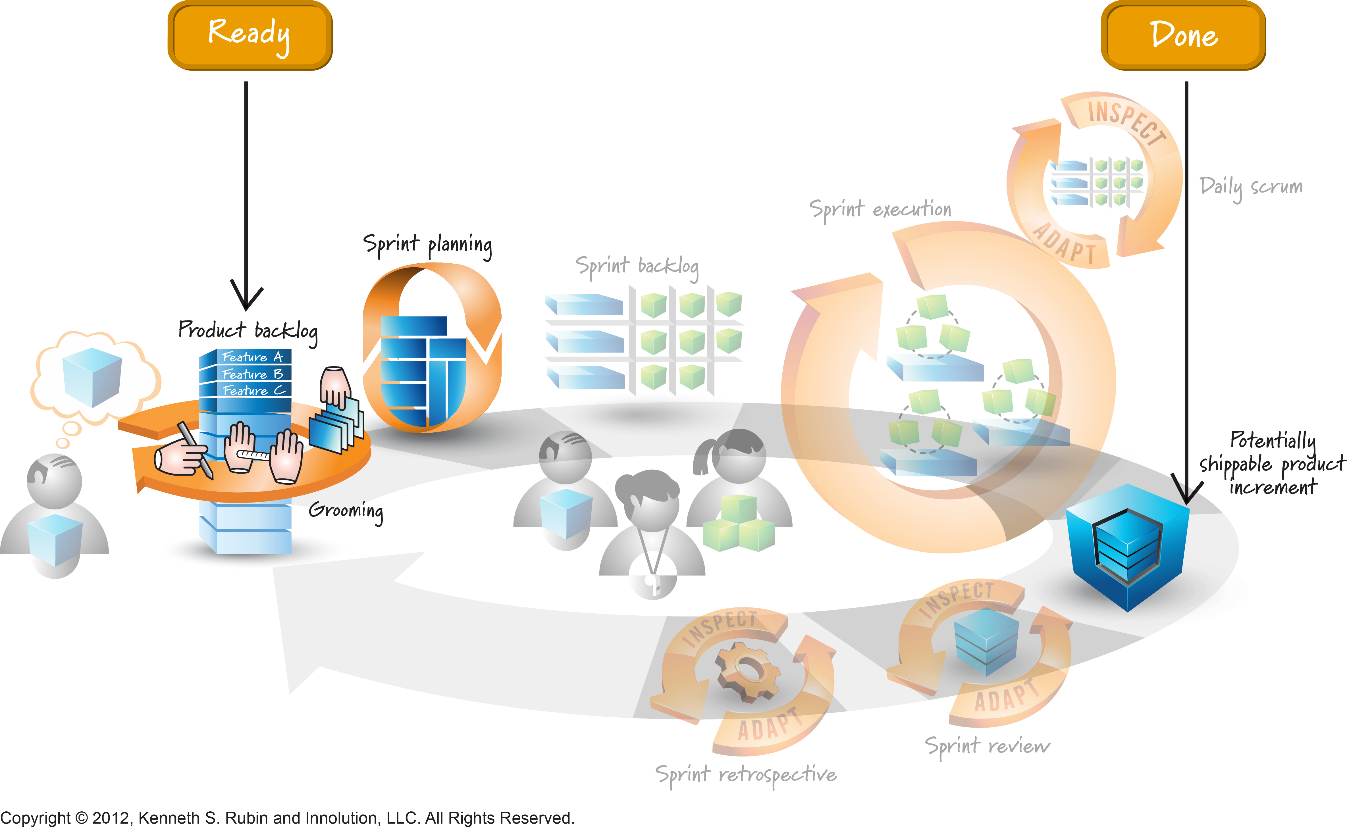
Note that there is a significant difference in user story acceptance criteria and user story definition of done. Acceptance criteria are used to make sure you built the right thing: ‘It was broken and now it is not’ or ‘It didn’t exist now it does’. Done criteria are used to make sure you completed the work with quality.

There is [another] key difference between “Done Criteria” and “Acceptance Criteria”. While Acceptance Criteria are unique for individual User Stories, Done Criteria are a set of rules that are applicable to all User Stories in a given Sprint. General Done Criteria could include any of the following[[6]](#footnote-6):

* Reviewed by other team members
* Completed unit testing of the User Story
* Completion of quality assurance tests
* Completion of all documentation related to the User Story
* All issues are fixed
* Successful demonstration to stakeholders and/or business representatives

As with the Acceptance Criteria, all conditions of the Done Criteria must be satisfied for the User Story to be considered ‘Done’.

Definition of Done [is] the criteria that the product increment must fulfill at the end of a sprint. These usually include high-priority product backlog items implemented as executable software that is thoroughly tested, free of defects, refactored, and adequately documented so that the increment could be shipped.[[7]](#footnote-7)

[[8]](#footnote-8)

The Scrum Team should use a checklist of the general Done Criteria to ensure a task is finished and the result meets the Definition of Done (DoD). A clear Definition of Done is critical because it helps remove ambiguity and allows the team to adhere to required quality norms. The definition of ’Done’ is typically determined and documented by the Scrum Guidance Body[[9]](#footnote-9).

The required records and data to comply with the project’s [‘Done’] documentation requirements can be generated as the team proceeds through Sprints and Releases[[10]](#footnote-10).

The inclusion of activities such as holding review meetings and writing design documents can help ensure compliance with internal and external quality standards. The basic principles of Scrum such as short iterations, incremental building, customer involvement, adaptation to changing requirements, and constantly adjusting scope, time, and cost within the project will still apply[[11]](#footnote-11).

**Self-Assessment**

**The Significance of “Done”:** Define the role of “done,” and describe the importance of having a potentially shippable product increment at the end of each Sprint in order to optimize business value, mitigate risk, and maximize transparency.[[12]](#footnote-12)

**Definition of Done (DoD):** Identify that Scrum’s minimal DoD means potentially shippable. If the Team is using anything other than that minimal DoD, it should be explicitly captured.[[13]](#footnote-13)

**Understanding the DoD:** Analyze the consequences of an inadequate DoD for the Team, the product, and the organization, and identify that any “undone” Product Backlog must be returned to the Product Backlog. Analyze the consequences of having a product in an “unstable/undefined” state due to accumulated “undone” work.[[14]](#footnote-14)

**Understand Quality**: Explain that quality is defined in the definition of done. It is no longer a variable but fixed.[[15]](#footnote-15)

1. Scrum Guide July 2013, [www.scrumguides.org](http://www.scrumguides.org) [↑](#footnote-ref-1)
2. SG [↑](#footnote-ref-2)
3. SG [↑](#footnote-ref-3)
4. SG [↑](#footnote-ref-4)
5. SG [↑](#footnote-ref-5)
6. Scrum Book of Knowledge, [www.scrumstudy.com](http://www.scrumstudy.com) [↑](#footnote-ref-6)
7. CSPO Learning Objectives, [www.scrumalliance.org](http://www.scrumalliance.org) [↑](#footnote-ref-7)
8. Agilexicon, <http://www.innolution.com/resources/visual-agilexicon-attribution> [↑](#footnote-ref-8)
9. SBOK [↑](#footnote-ref-9)
10. SBOK [↑](#footnote-ref-10)
11. SBOK [↑](#footnote-ref-11)
12. CSM Learning Objectives, [www.scrumalliance.org](http://www.scrumalliance.org) [↑](#footnote-ref-12)
13. CSM [↑](#footnote-ref-13)
14. CSM [↑](#footnote-ref-14)
15. CSPO [↑](#footnote-ref-15)