

Table of Contents

What is a Definition of Ready (DoR)? - SSA	3
Why Do We Have a Definition of Ready?	3
What Is Typically Captured as a "Ready" User Story on the Story Card?	4
What Tips Do You Have To Help Develop the Definition of Ready?	5
How Do We Know Definition of Ready Is Being Successfully Applied?	5
What Do We Not Want To See When Applying Definition of Ready?	6
Want to Know More?	6

What is a Definition of Ready (DoR)? - SSA

The Definition of Ready (DoR) is part of a working agreement that specifies the conditions that need to be met for a Story before it can be accepted into an Iteration or Sprint. The DoR is typically based on the INVEST criteria for a Story.

The Definition of Ready is an analog to the Definition of Done (DoD), but is aimed at the earlier part of a story's life cycle.

Why Do We Have a Definition of Ready?

“The secret of success is to be ready when your opportunity comes” - Benjamin Disraeli

Simply getting your stories ready will have an immediate and dramatic impact on the Team's productivity; the data indicates at least a 2X improvement in Team performance. The reason for this improvement is:

- Teams avoid beginning work on stories that do not have clearly defined completion criteria. Without a clear completion criteria Teams will waste time on costly back-and-forth discussions as work progresses or, worse, will result in significant rework when work does not meet actual need.
- By having a DoR Teams are able to push back on starting work on ill-defined Stories.

[→ Read more...](#)

How do we know if we have a “good” User Story? People use the mnemonic to make sure they have good Stories to work with – INVEST. INVEST stands for:

- **I**ndependant: Work item is modular and can be delivered with no (or at least manageable) dependencies.
- **N**egotiable: A basic solution is articulated, with room for options.
- **V**aluable: The value is clearly articulated. This is defined in the “so that” part of the User Story.
- **E**stimable: Provides just enough information so it can be sized compared to other work (relative sizing).
- **S**mall: Small enough to be completed in 1 Iteration (Sprint). Some call this attribute “sized to fit”.
- **T**estable: Acceptance criteria has been developed and are understood by the Team.

The initial draft of a Story will not have all these characteristics. Rather the mnemonic guides the discussion to improve our understanding of the Story. Many teams establish a [Definition of Ready \(DoR\)](#) criteria for a Story. In other words we are “READY” when the Story has the INVEST characteristics. Then as Story bubbles up the Backlog and it becomes a candidate for work, they refine the Story so that it has these characteristics helping to increase overall understanding.

[→ Read more...](#)

The Product Owner is responsible for putting the Features and Stories in the Backlog. The Team works with the Product Owner during Backlog Refinement to help Product Owners get the stories into actionable shape.

Backlog refinement is usually the first time that the Team can work though an understanding of dependencies associated with the work, be they internal (e.g. availability of specialists or systems) or external (e.g. need for something from a supplier).

The main thing to achieve is to ensure that we understand how the dependency lines up with Iteration (Sprint) timing. For example, if an external dependency with a supplier is expected to be resolved in 2 months time, then it probably doesn't make sense to prioritize our work so that it would be scheduled for the upcoming iteration. You might as well wait until the value can be delivered and, in the meantime, deliver other valuable stuff (control WIP by "stop starting, start finishing").

What Is Typically Captured as a "Ready" User Story on the Story Card?

The way to think about the information on a card is that, while you do not list out all the details you have, you use the card to capture the basics, to act as a reminder and to provide information on where to find more details if required. One way to think about the level of information is to think about an old library system to find books - an index card would allow you to find the information you need. And don't forget, a User Story is a "reminder to have a conversation".

The following sample information follows the expectations from established as a result of following the [the INVEST criteria](#), assuming you are capturing the results in a work management system (e.g. Jira, VersionOne, Rally, etc.):

- Title: 3-10 word descriptive title that clearly describes the outcome; make it meaningful. Experience shows that cards (and tools) get very "busy" if you use the user voice format of the story to reference it. After spending time with the story, people will naturally use short cuts to reference the information. This is the short cut and is used to reference the requirement in discussions and meetings. Titles work best when it completes the statement:
 - "I wish I had..." or
 - "I wish the system would..." Has a short, 3-10 word descriptive title that clearly describes the outcome; make it meaningful
- User Voice: Has a "user voice" form of the requirement ("As a <role> I want <capability> so that <I get this value>")
- Acceptance Criteria: Has an acceptance criteria that helps us understand when we are done. Where possible these should be defined with examples (i.e. mostly written in Gherkin "Given <system is in this state> when <I take this action> then <I expect to see this result>")
- Type: The issue type (story, defect, etc.) is correct for the work

- Estimate: Has an estimate based on completing to Definition of Done, and used by the Team to understand how big the piece of work is so they can plan appropriately. For Stories that are about to be worked on the estimate small enough to fit into an Iteration (Sprint) (estimate is typically sized below some threshold that means something to the Team e.g. 8's and below will fit into an Iteration)
- Dependencies: Have been identified and priority aligns with Iteration (Sprint) timing
- Prioritized: Items “expected” for the upcoming Iteration (Sprint) are prioritized appropriately, including improvement items from the Retrospective. In other words they are at the top of the Team (Product) Backlog

If you are doing this with physical 5×3” index cards, the back of the card will typically have the acceptance criteria and other notes that come up during the discussion, while the front will have the title, description, estimates, etc.

What Tips Do You Have To Help Develop the Definition of Ready?

Here are some tips and tricks to help you develop your Definition of Ready practice:

- The Product Owner should work with the Agile Team to establish their initial Definition of Ready. This should be a collaborative effort.
- The Backlog Refinement event is the main event to work stories to Definition of Ready
- Think about creating a Ready (sub-)state on your Team Kanban board to reduce the amount of time you revisit items as part of refinement and visually see that there is sufficient backlog for the Team.
- Define your exit criteria for the step before commitment to work based on your Definition of Ready
- Ensure Teams and Product Owner regularly discuss their Definition of Ready. After all, this is a working agreement aimed at improving the flow of value and so will need revision as things change.
- Use time-boxes to limit how much time is spent collaborating on a Story, and treat items with lots of unknowns as “homework”. For example, perhaps set up a Spike (Enabler) story if there is requirements or implementation uncertainty, and bring results back to the next Iteration Planning or Backlog Refinement event.

How Do We Know Definition of Ready Is Being Successfully Applied?

We know the Definition of Ready is being successfully applied when:

- There is a smooth flow of top priority (business perspective) Stories to the Teams
- For each Team there is enough Ready Backlog for at least 2-3 of Iterations (Sprints) at any given time.
- Story is well understood by the Team and uses language everyone comprehends (e.g. avoid acronyms, jargon, and superfluous words)
- Iteration (Sprint) planning results in a plan that seems relatively low-risk
- Clear understanding of how the results can be demonstrated to customers and other stakeholders to solicit feedback during an Iteration (Sprint) Review.
 - NOTE: Think of the user voice format and acceptance criteria as the “script” that you will use when you demonstrate the story at the Iteration Demo.
- Team is set up to up to quality Stories, with limited requirement defects
- Team consistently adheres to Definition of Ready criteria

What Do We Not Want To See When Applying Definition of Ready?

The Definition of Ready is not being used well when:

- The Product Owner is expected to come up independently with Definition of Ready stories.
- The Team is required to accept Stories for work that do not meet Definition of Ready.
- Too much time is spent on each Story to get it Ready.
- The process becomes a “stage gate” whereby it literally gets in the way of the smooth flow value, or Teams use it as an excuse not to work on things. Remember, this is all about the smooth flow of value and collaboration!

Want to Know More?

- [\(Agile Alliance\) What is Definition of Ready?](#)
- [\(Scrum Inc\) Definition of Ready](#)
- [Gherkin](#) as a way of specifying acceptance criteria, and [Cucumber](#) as a way to implement automated testing of Gherkin.

[FAQ](#), [DoR](#), [Ready](#), [SubjectSpecificArticle](#), [UserStory](#), [Practice](#), [OnePager](#)

From: <https://www.hanssamios.com/dokuwiki/> - Hans Samios' Personal Lean-Agile Knowledge Base

Permanent link: https://www.hanssamios.com/dokuwiki/what_is_a_definition_of_ready_dor_ssa

Last update: 2022/02/23 07:03



