Knights of Columbus Product Backlog Grooming and Story Decomposition Guide

November 15, 2017

Emily Hall

Agile Coach

Knights of Columbus

# Document History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Editor | Summary |
| 1.0 | 11/15/2017 | Emily Hall | Initial Draft |

Contents

[Document History 1](#_Toc498508277)

[Summary 1](#_Toc498508278)

[Goals of Weekly Backlog Grooming 1](#_Toc498508279)

[CRUD/Operations 2](#_Toc498508280)

[Workflow Steps 2](#_Toc498508281)

[Business Rules 2](#_Toc498508282)

[Test Cases 2](#_Toc498508283)

[Input Options/Platform 2](#_Toc498508284)

[DataTypes or Parameters 2](#_Toc498508285)

[Roles 2](#_Toc498508286)

[Sources 3](#_Toc498508287)

# 

# Summary

Decomposing stories into smaller stories can help a team plan more effectively in the context of a Sprint time box. Some teams have a limited charter and need to decompose a product backlog story to define work within their chartered scope (for example, enabling single sign-on for a larger customer facing system). Estimation against user stories is more accurate when stories are smaller.

The decomposition of stories can be accomplished using a number of methods outlined below.

Backlog Grooming

# The Weekly Backlog Grooming Meeting

## Goals of the Weekly Backlog Grooming Meeting

1. Come out with at least a few backlog items that are right sized, estimatable, and well formed.
   1. **Right Sized:** approximately two to five person days to build.
   2. **Estimatable:** the dev team is moderately comfortable with making an estimate (the work involved is clear enough, the unknowns are identified enough, and the path to completion is clear enough).
   3. **Well formed:** based on the words used, the story can be deemed to either be right sized and estimatable, or not.
2. Getting to a deep enough understanding of the backlog item(s) so that the dev team could accept it into a sprint with a moment's notice.
3. Splitting the Product Backlog Items (user stories) to support the team’s design and planning needs.
4. Have enough high priority items finely refined to fill about two sprints.
5. Define testing criteria for user stories.

## Rules of the Weekly Backlog Grooming Meeting

1. Do not over-discuss or over-design.
2. Stick to the stories at hand unless a high risk needs to be surfaced. The Product Owner owns the business priorities, the team works from the top of the backlog.

# CRUD/Operations

CRUD; create, read, update, delete; is a method of breaking a user story down by action. Is something created or deleted? Is something read or update? This level of story decomposition can assist the team in reducing a larger story to a smaller one by examining the underlying action behind achieving customer value.

# Workflow Steps

Define the user story as a workflow with an ordered list of steps to achieve the outcome. If each step can be describe from the user perspective, do so. If a step needs to be described in terms of technical detail, that is also acceptable, but only if it cannot be described from the user perspective. By ensure that we keep the user in mind during the design phase, we ensure that we are empowering the Product Owner to control the prioritization of the backlog as well as ensuring that our work remains focused on delivering customer value.

# Business Rules

Individual business rules can be identified as user stories. In many cases, business rules will have a different actor and reasoning. Business rules user stories are often testing driven.

# Test Cases

These stories are usually written in an if/when scenario. This strategy often results in application of business rules, workflow steps, and input operations.

# Input Options/Platform

If functionality is available on a variety of platforms, user stories can be differentiated by platform to focus efforts on the more important platforms first.

# DataTypes or Parameters

By breaking down stories by DataTypes or Parameters in use we can focus on a smaller amount of functionality. If a story allow a user to interact with the system in a number of different ways resulting in a number of different results (like interacting with a search engine) each data type interaction should be described using its own story. Additionally, if a number of different data types can be returned to a user as a result of a single story, each of those results should be described as a story. This allow the product owner to prioritize the most important results.

# Roles

If a story would or could behave differently based on a user’s role--that could be broken out as a separate story.

# Sources

Rupani, Bharti. (June 29, 2015). Make Your User Stories Manageable Through Story Decomposition. *excella consulting.* Retrieved from <https://www.excella.com/insights/make-your-agile-user-stories-manageable-through-story-decomposition>.

Verwijs, Christiaan. (2015). 10 useful strategies for breaking down large User Stories (and a cheat sheet). *Agilistic.* Retrieved from <https://blog.agilistic.nl/10-useful-strategies-for-breaking-down-large-user-stories-and-a-cheatsheet/>