

User Story vs Use Case

[User Story vs Use Case](#)

[Similarity and Difference](#)

[Similarity](#)

[Difference](#)

[What is a User Story](#)

[Concept of 3C's](#)

[What is Use Cases](#)

[Detailed Use Case Specification](#)

[Why Need Use Case?](#)

[Integration](#)

[Related Links](#)

"Is a User Story the same thing as a Use Case?" People often ask this question and the dispute on whether an agile team should practice Use Stories vs Use Cases has been around the field for years. Are User Story and Use Case the same thing? If not, which is better? Which one should you use? Or could use both?



Although there are some similarities between User Stories and Use Cases, User Stories and Use Cases are **not interchangeable**; both User Stories and Use Cases identify users and they both describe goal, but they serve different purposes.

User Stories are centered on the result and the benefit of the thing you're describing, whereas Use Cases can be more granular, and describe how your system will act. Is there a place for Use Cases in Agile, or can they be used in conjunction with each other?

This article will tell you the difference between User Stories and Use Cases.

Agile Software for Scrum Teams

Need an agile software solution for product backlog management? Visual Paradigm supports a powerful agile toolset that covers user story mapping, affinity estimation, sprint management, etc. It's powerful but yet easy-to-use, intuitive and, most important, AGILE.

[Free Download \(/download/\)](/download/)

User Stories vs Use Cases

User Stories often start out the same way as Use Cases, in that each describes one way to use the system, is centered around a goal, is written from the perspective of a user, uses the natural language of the business, and - on its own - does not tell the whole story.

User Stories vs Use Cases - The Similarity

If we consider the key component in both approaches:

- User Stories contain, with user role, goal and acceptance criteria.
- Use Cases contain equivalent elements: an actor, flow of events and post conditions respectively (a detailed Use Case template may contain many more other elements).

User Stories vs Use Cases - The Difference

- The details of a User Story may not be documented to the same extreme as a Use Case.
- User Stories deliberately leave out a lot of important details. User Stories are meant to elicit conversations by asking questions during scrum meetings.
- Small increments for getting feedback more frequently, rather than having more detailed up-front requirement specification as in Use Cases.

What is a User Story?

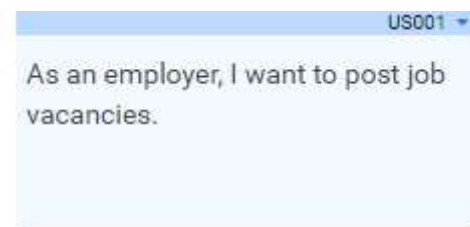
A User Story is a note that captures what a user does or needs to do as part of her work. Each User Story consists of a short description written from user's point of view, with natural language. Unlike the traditional requirement capturing, User Story focuses on what the user needs instead of what the system should deliver. This leaves room for further discussion of solutions and the result of a system that can really fit into the customers' business workflow, solving their operational problems and most importantly adding value to the organization.

Concept of 3C's

The 3C's refer to the three critical aspects of good user stories. The concept was suggested by **Ron Jeffries** (<https://ronjeffries.com/xprog/articles/expcardconversationconfirmation/>), the co-inventor of the user stories practice. Nowadays, when we talk about User Stories, we usually are referring to the kind of User Stories that are composed of these three aspects.

Card

User Stories are written as cards. Each User Story card has a short sentence with just-enough text to remind everyone of what the story is about.



Conversation

Requirements are found and re-fined through a continuous conversations between customers and development team throughout the whole software project. Important ideas and decisions would be discovered and recorded during the stakeholder meetings.

Story Map

Estimate & Spike

Sprint

Configuration

Conversation

Confirmation

Scenario

Storyboard

Design

Reference

History

B

I

A

^

v

☰

☷

Agreement of statement of Purposes for Personal Data required

P1 - Employment information

General employer vs Employment agency

Business req certificate no.

Sector/industry

Company name

Contract details (name, tel, fax, email address)

P2 - Vacancy information

Job title

Job duties

Contract details - perm? temp? summer job?

Working hrs

Salary - basic salary, double pay, other benefits

Edu level

Working exp?

Confirmation

Confirmation is also known as the acceptance criteria of the User Story. During the discussion of requirements, the customers tells the analyst not only what he/she wants, but also confirming under what conditions and criteria the working software would be accepted or rejected. The cases defined are written as confirmation. Note that confirmation focuses on verifying the correctness of work of the corresponding User Story. It is not an integration testing.

Story Map

Estimate & Spike

Sprint

Configuration

Conversation

Confirmation (0/10)

Scenario

Storyboard

Design

Reference

History

B I A ^ v

A summary of vacancy is displayed by submitting the form with all compulsory fields filled.

1. Click on **Post Vacancy**

Pre-Vacancy Order Form presented

2. Select **General Employer** for **Employer Type**

3. Click **Next**

Vacancy Order Form presented

4. Fill in all fields marked *

5. Click **Next**

Summary of vacancy presented

A summary of vacancy is displayed by submitting the form with all fields filled.

The field 'General employer vs Employment agency' will be focused by submitting the form with this field unspecified.

The field 'Business registration certificate no.' will be focused by submitting the form with this field unspecified.

The field 'Sector/Industry' will be focused by submitting the form with this field unspecified.

The field 'Company name' will be focused by submitting the form with this field unspecified.

The field 'Contact details' will be focused by submitting the form with this field unspecified.

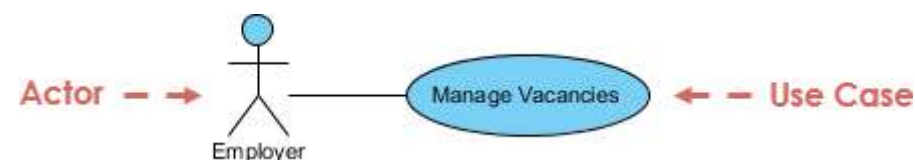
The field 'Job title' will be focused by submitting the form with this field unspecified.

The field 'Job duties' will be focused by submitting the form with this field unspecified.

The field 'Contract details' will be focused by submitting the form with this field unspecified.

What is Use Cases?

Use Cases, introduced by Ivar Jacobson (https://en.wikipedia.org/wiki/Ivar_Jacobson) more than 20 years ago, are used to capture user (actor) point of view while describing functional requirements of the system. They describe the step by step process a user goes through to complete that goal using a software system.



A Use Case is a description of all the ways an end-user wants to "use" a system. Use Cases capture all the possible ways the user and system can interact that result in the user achieving the goal. They also capture all the things that can go wrong along the way that prevent the user from achieving the goal.

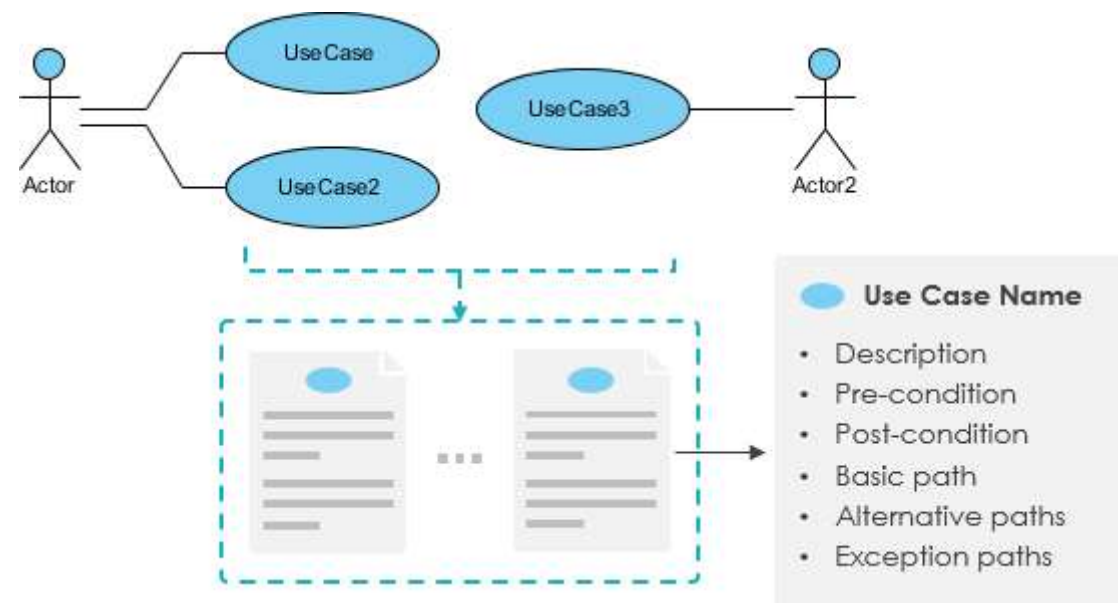
A Use-Case model consists of a number of model elements. The most important model elements are:

- Use Cases,
- Actors

- and the relationships between them.

Detailed Use Case Specification

A Use Case Specification is a textual description of the functionality provided by the system. It captures actor-system interaction. That is, it specifies how a user interacts with a system and how the system responds to the user actions. It is often phrased in the form of a dialog between the actor and the system. The Use Case Specification is represented in the Use Case Diagram by an oval, and is what most people think of when they hear the term Use Case.



Why We Still Need Use Cases?

Alistair Cockburn (<http://alistair.cockburn.us/>), explains that he sees (with companies he consults to) three main problems with User Stories:

1. Lack of context (what's the largest goal)
2. Sense of completeness that you covered all bases relating to a goal.
3. No mechanism for looking ahead at upcoming work.

Integrate Use Case, User Story and Story Mapping techniques

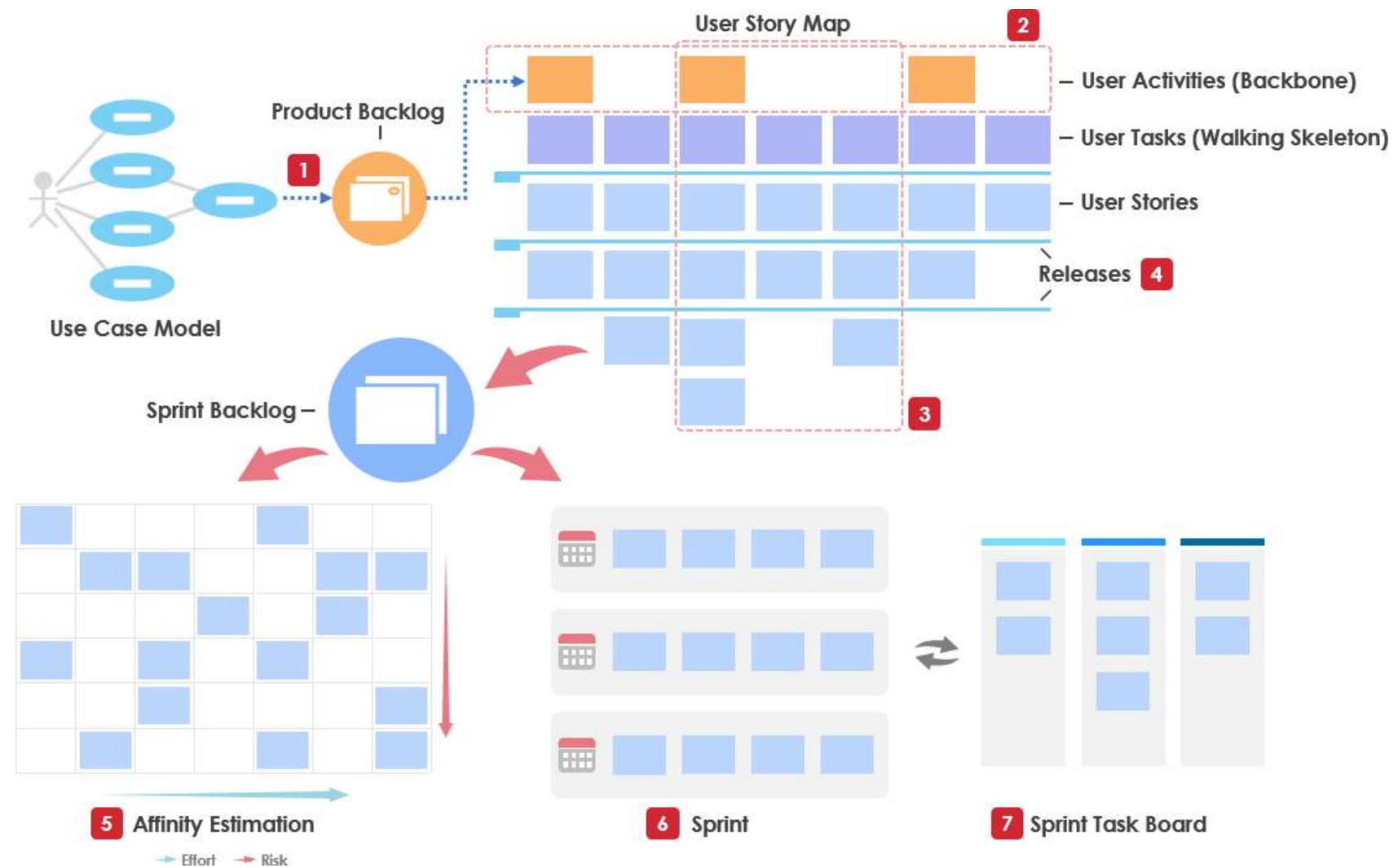
Visual Paradigm ([/](#)) provides a complete agile environment that integrates Use Case, User Story, story mapping, affinity estimation, and Kanban into a completely seamless and automated end-to-end process. This process can address the shortcoming of what Alistair mentioned above with the User Story technique by

supplementing Use Case and story mapping tools. The other useful agile tools are also incorporated for all the needs to manage your agile projects faster, better and smarter.

The concept map below gives an overview of the agile tools supported by Visual Paradigm.

1. Send requirements from visual model as product backlog item (for use in story map construction)
2. User activities in story map, which represents a large system context as a whole
3. Vertical structuring of activities, tasks and stories - Completeness of backlog
4. Release management
5. Estimate user stories based on their development effort and risk
6. Manage development activities with sprint
7. Track progress with sprint task board

Point 1 to 3 are tools for supplementing the short coming of user stories. The other user agile tools are listed in Point 4 to 7.



Ready for Agile?

Want an agile tool that can manage your scrum projects well? Visual Paradigm features a user story mapping tool, Affinity Estimation tool, sprint management tool, and task management.

Try it Free ([/download/](#))

Related Links

1. [Professional agile software tool with story mapping, affinity estimation and more \(/features/agile-user-story-mapping-tool\)](#).
2. [Try Visual Paradigm FREE \(/download/\)](#).

Turn every software project into a successful one.

Try Visual Paradigm Free (/download/)

_(/)	<u>Product</u>	<u>Support</u>	<u>Learn</u>	<u>About Us</u>
	<u>Features</u> (/features/)	<u>Forums</u> (http://forums.visual-paradigm.com/)	<u>Community Circle</u> (https://circle.visual-paradigm.com/)	<u>Visual Paradigm</u> (/aboutus/)
	<u>Editions</u> (/editions/)			<u>Newsroom</u> (/aboutus/newsreleases/)
	<u>Try Now</u> (/download/)	<u>Request Help</u> (/support/#support-form)	<u>Know-how</u> (https://knowhow.visual-paradigm.com/)	(https://www.youtube.com/user/VisualParadigm/)
	<u>Pricing</u> (/shop/)			
	<u>Visual Paradigm Online</u> (https://online.visual-paradigm.com/)	<u>Customer Service</u> (https://cs.visual-paradigm.com/)	<u>Demo Videos</u> (/features/demo/)	<u>Academic Partnership</u> (/partner/academic/)
			<u>Tutorials</u> (/tutorials/)	
			<u>Documents</u> (/support/documents/)	

(<https://www.youtube.com/user/VisualParadigm>)



(<https://www.instagram.com/visualparadigm>)