

User Story

A clear, concise description for a desirable user need or outcome

SIMPLIFIED, ONE-PAGE SUMMARY (SOS)



WHAT is a User Story?

A **User Story** is a concise description of a desired outcome written from the user's perspective. User stories define what the user needs, enabling the team to deliver value in an incremental way. "Story" refers to the narrative that explains the customer need. [e.g. "What is the story behind this request?"]¹

WHY is a User Story valuable?

- User stories focus and align the team on what matters most and reduces the risk of delivering something that the user does not want or need.
- Because user stories are designed as small chunks of work that deliver value, they can be easily reprioritized and changed. This makes them well-suited for complex, dynamic environments instead of defining all requirements up front in advance.

HOW TO START¹

Often helpful for newly formed teams, this thought exercise helps weigh the components of a user story. It's not essential and experienced teams often discontinue it over time.



- "As a..."** | Defines the user role, providing context for the outcome that best suites their specific needs
- "...I want..."** | Describes the user's desired outcome—regardless of how that goal might be accomplished
- "...so that..."** | Conveys why the outcome is important and identifies what success looks like to the user

THE I.N.V.E.S.T. TEST²

A good user story should be...

- INDEPENDENT** | If released, it would provide incremental value on its own
- NEGOTIABLE** | Leaves room to negotiate the methods used to produce the outcome
- VALUABLE** | The value is clear for the end user
- ESTIMABLE** | Provides enough information to size the story compared to other work
- SMALL** | Small enough to be completed in one iteration (e.g. one sprint)
- TESTABLE** | It's clear when the outcome has been achieved and the work is complete



THE THREE "C"s³

- Card** | Concise description and rationale that fits on an index card
- Conversation** | Starts the discussion around possible solution options
- Confirmation** | Be certain we all understand what it means to be done

ANATOMY OF A USER STORY

- TITLE**
A few words that clearly summarize the desired outcome
- DESCRIPTION**
Describe the desired outcome (Use the "As a...I want... so that..." format if it helps.)
- ACCEPTANCE CRITERIA**
Helps the team understand what's required to deliver this outcome Given, when, then may be helpful.

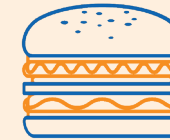
Determine Customer Order History

As an existing customer, I want to be able to view recently purchased items so that I can easily re-order them.

Acceptance Criteria:

1. *Given a customer placed an order <= 6 months ago, when they click the [My Previous Orders] link, then display their prior orders <= 6 months*
2. *Given a previously ordered item is displayed and in stock, when a customer clicks [buy again], then add the item to their shopping cart*

THE CHEESEBURGER ANALOGY⁴



How to Split Stories Vertically

As a restaurant customer, I want a delicious cheeseburger, so that I can satisfy my hunger and enjoy a tasty treat.

- ▶ The team realizes they cannot build the entire cheeseburger in one iteration, so we need to "split" the story into smaller chunks.
- ▶ Our first inclination might be to build the bun first to build the infrastructure for the rest of the cheeseburger. But what if we build the entire bun and our customer tells us they are on a gluten-free diet?
- ▶ Rather than risk wasting time building all of one component, we build just enough bun, patty, cheese, lettuce, etc to build one "bite" for our customer

DEFINITION OF SUCCESS

- Meets all INVEST mnemonic criteria
- Well understood by the team and uses language everyone comprehends (e.g. avoid acronyms, jargon, and superfluous words)
- Always aligns to the product vision, goals and objectives
- Results can be demonstrated to customers to solicit feedback
- Makes incremental progress toward the overall outcome or objective
- Sets the team up to create defect-free and production viable solutions

TIPS & TRAPS

- Split big stories vertically to avoid separation of design, build, and test
- Ensure each story can deliver customer value and do not simply describe team tasks
- Start by limiting story scope to no more than four hours of actual development work that can be done over just a few days and adjust as needed
- Everyone contributes to user story creation, not just the product owner
- If a story requires extensive acceptance criteria, it is likely the story needs to be split
- Any story can also include the detailed tasks required to build the solution, but tasks are optional and should include whatever is helpful for the team

RECOMMENDED RESOURCES

- User Stories Applied by Mike Cohn bit.ly/5whyz002
- Story Mapping by Jeff Patton bit.ly/5whyz003
- Story Splitting Patterns by Agile for All bit.ly/5whyz004
- User Story Splitting with SPIDER bit.ly/5whyz005

