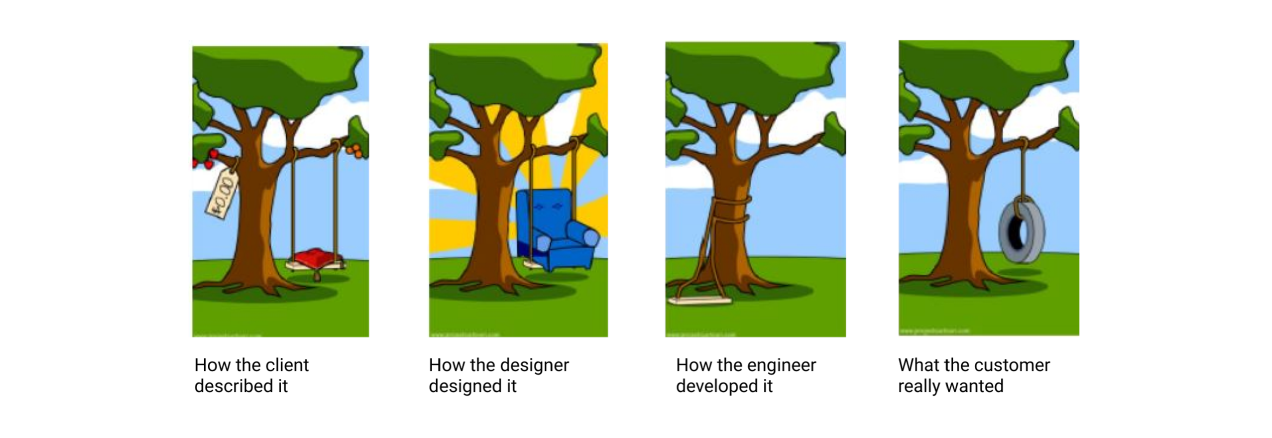
## What Are Acceptance Criteria?

Acceptance criteria are the predefined requirements that must be met, taking all possible scenarios into account, to consider a user story to be finished.

In other words, they specify the conditions under which a user story can be said to be ‘done’. When all the criteria are met, the team can set the task aside and move on to the next story.

**Acceptance criteria are used to:**

* Manage expectations: Acceptance stories clearly define the boundaries of the task. Usually, acceptance criteria are testable, with yes/no or pass/fail results that leave no room for misinterpretation.
* Arrive at a shared understanding with the client: There have been instances where the client might feel that they needed more from the feature, and that it does not meet their requirement in its entirety. Well documented acceptance criteria will address all such ambiguity.
* Spell out the functionality for tests: The defined criteria will help to check whether the system is performing in line with the expectations.
* Work on estimates: When the team is clear about the boundaries of each task, they will be in a position to make accurate estimates.



The image above, which needs no explanation, shows what could happen when the acceptance criteria are not well defined! Each of the outcomes has a tree, a rope and a swing; but they are a far cry from the poor customer’s ask.

## How Should You Format User Story Acceptance Criteria?

There are two commonly used formats for acceptance criteria:

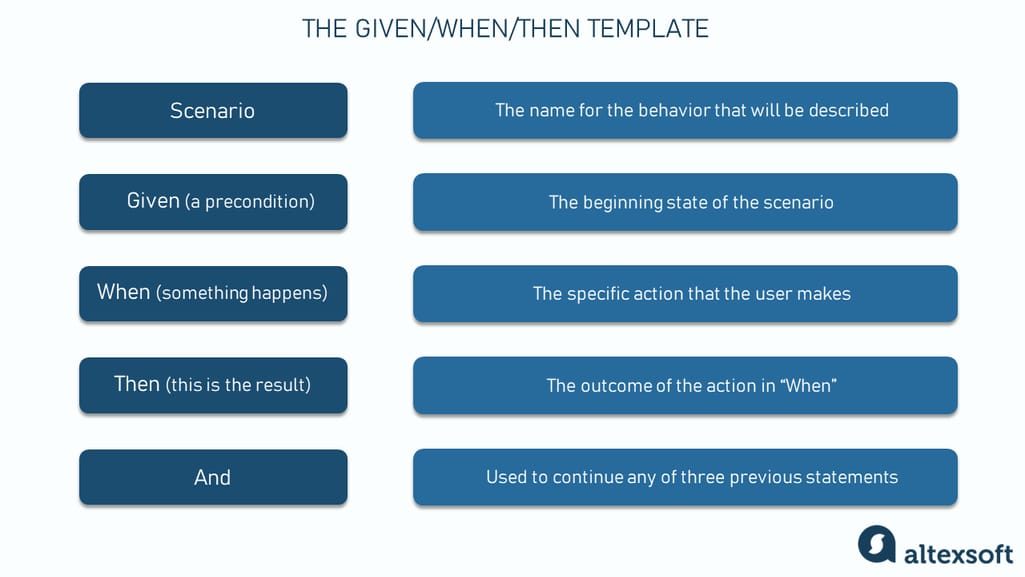
1. **Given/When/Then**

For a user story that typically follows this format:

As a (intended user), I want to (intended action), so that (goal/outcome of action).

…the acceptance criteria would be like this:

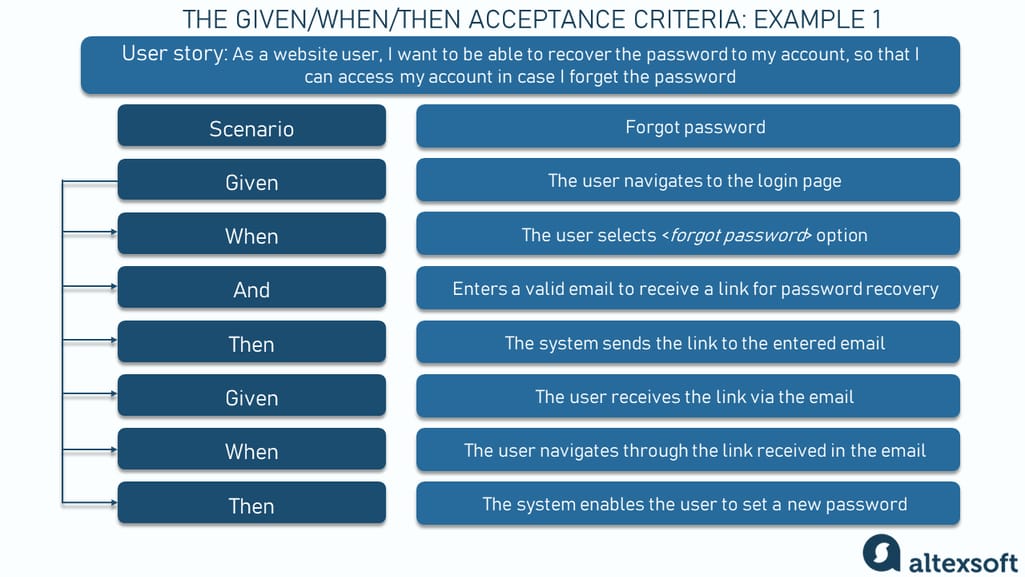
Scenario: (explain scenario). Given (how things begin), when (action taken), then (outcome of taking action).



By utilizing this template, teams can articulate acceptance criteria more precisely, ensuring clear and efficient communication between developers, testers, and stakeholders.

**EXAMPLE:**

**User story:** As a website user, I want to be able to recover the password to my account, so that I will be able to access my account in case I forgot the password.



**“Recovering the password” acceptance criteria example**

**Scenario:** Forgot password

* **Given:** The user navigates to the login page
* **When:**The user selects <*forgot password*> option
* **And:** Enters a valid email to receive a link for password recovery
* **Then:** The system sends the link to the entered email
* **Given:** The user receives the link via the email
* **When:**The user navigates through the link received in the email
* **Then:** The system enables the user to set a new password

**EXAMPLE 2:**

* User story: As an online buyer, I want to add a book to my shopping cart, so that I can purchase it.
* Acceptance criteria:  Given that I have shortlisted three books in my wish list, when I click on one book, then it gets added to my shopping cart.

1. **Verification List**

The team makes a verification checklist, defining a list of pass/fail or yes/no statements that will mark the functionality as complete.

Whatever format you choose, it should be something that the team is comfortable working with.

