

Introduction to Requirements in Agile Processes

CS 2720

Many agile processes use *user stories* to capture requirements. Ron Jeffries defined three critical aspects of “social” user stories:

- *Card*
- *Conversation*
- *Confirmation*

Mike Cohn suggests using the [following template](#) for user stories:

As a «role» I want «feature» so that «reason».

Acceptance Criteria:

«approx. 3-5 “conditions of satisfaction”»

Sample User Story

¹ As a vice president of marketing, I want to select a holiday season to be used when reviewing the performance of past advertising campaigns so that I can identify profitable ones.

Acceptance Criteria:

- Make sure it works with major retail holidays: Christmas, Easter, Mother's Day, Father's Day, New Year's Day
- Support holidays that span two calendar years (none span three)
- Holiday seasons can be set from one holiday to the next (such as Thanksgiving to Christmas)
- Holiday seasons can be set to be a number of days prior to the holiday

¹From Mike Cohn's [User Stories](#)

² As a club member, I can cancel a reservation so that I do not get charged the full amount for a room I don't use.

Acceptance Criteria:

- A non-premium member is charged 10% for a same-day cancellation
- A premium member can cancel the same day without a fee
- An e-mail confirmation is sent
- The hotel is notified of any cancellation

²Modified from Mike Cohn's [Introduction to User Stories](#)

Sample Workflow

- User stories are written and placed in a *product backlog*. Stories are continuously tracked on the (*story, task, agile*) *board*.
- Through discussion with the *product owner*, a user story may be classified as an *epic* and divided later into smaller user stories.
- The team and the product owner select important user stories for the next iteration and place them in the *sprint backlog*.
- Stories in the *sprint backlog* are expanded into one or more *sprint tasks* by the team.
- The product owner meets with the team in the *sprint review meeting* to verify the story was completed.

What's a Good User Story?

While we don't need the formalisms of a plan-driven approach, a good user story still has some requirements.

Bill Wake suggests the **INVEST** criteria:

- *Independent*
- *Negotiable*
- *Valuable*
- *Estimable*
- *Small*
- *Testable*

What's a Good Task?

You have likely already heard the **SMART criteria** that Bill Wake suggests for tasks:

- *Specific*
- *Measurable*
- *Achievable*
- *Relevant*
- *Time-boxed*