

Writing Contracts for Agile Development

by Mike Cohn • 18 Comments • originally published in Gantthead on 2006-09-01

User stories have become an increasingly popular way for agile teams to express requirements. Teams like that working with user stories shifts the emphasis from writing about requirements to talking about them. Project customers like that they are not expected to do the impossible and identify all needs upfront. Users of a product built with user stories like that the product is more likely to meet their needs than one built using a requirements technique more focused on writing than talking.

However, even though user stories cause teams and customers to talk more, there is often much still that needs to be written. Contract or outsourced development is one situation in which a team will need to augment conversations with documentation. This article will show one technique I've used for writing contracts to cover contracted agile development.

A few years ago I worked on a system intended for use by broadcasters during the summer Olympics. Sportscasters, writers, and others would use the system to dredge up facts about competitors in the various events. A sample scenario was that a sportscaster who would be covering the 400-meter hurdles later in the day would use this system to research past performance and find interesting personal facts to interject into the commentary. The system was designed as a large web of information: a user could, for example, start with the event (400-meter hurdles) and drill down from country to athlete. Or starting with an athlete a user could navigate to all events that athlete would compete in and then to another athlete in one of those events, and so on. This application was outsourced to my team which developed it using Scrum, one of the most popular agile processes

Requirements were written as user stories using this template:

As a , I want [so that].

This template allows user stories to clearly identify the user who wants to accomplish something, what it is that she wants to accomplish, and optionally why. There were a handful of

user types on this system but for this example we only need to know about two of them: viewers and content providers. A viewer was any user who was primarily looking at data in the system. This includes the sportscasters and researchers discussed earlier. Content providers were the writers who put all of the data into the system in advance of its use. Some of the stories identified for this system were:

1. As a viewer I can view details about a specific athlete.
2. As a viewer viewing an athlete, I can easily navigate to the events he's in and a list of other athletes on his team.
3. As a viewer I can bookmark athletes of interest.
4. As a viewer I can easily return to athletes I've bookmarked.
5. As a viewer I can highlight portions of an athlete's profile so that when I next view the profile I can see items of most interest to me.
6. As a content provider, I can have basic formatting control over how information is displayed.

It would have been challenging and risky for us to estimate work based solely on these user stories and then to commit to a fixed price contract. We needed more information, which we could get through conversations with our customer. But we also needed to document some key assumptions and decisions that would result from those conversations. What we did was identify a small number of high level acceptance criteria for each user story. Today I refer to these as conditions of satisfaction for the story. We then produced a requirements documents that included each user story along with its conditions of satisfaction. Here's how the first story above was augmented with its conditions of satisfaction:

1. As a viewer I can view details about a specific athlete.
 - Name (multiple, could be long, include pronunciation)
 - Nickname (include pronunciation)
 - Prior performance at Olympics
 - World and Olympic records held
 - Interesting anecdotes
 - Citizenship, where trained, coach (link)

6. As a content provider I can have basic formatting control over how information is displayed.

- Italics
- Bold
- Bulleted and numbered lists
- Not WYSIWYG but can click a button to see a preview

The conversations that led the team to this list were held before the contract was written and before coding began. Discussions of story #6 were particularly important because the team was able to rule out the WYSIWYG (What You See Is What You Get) editor and save the customer money with a preview button instead. They also determined that very simple formatting was all that was required. Font styles, sizes, and colors, for example, were not needed. This helped avoid a potential conflict that could have resulted in a money-losing low bid or an unnecessarily high bid for the project that the customer would have rejected.

Sometimes a user story that appears very straightforward hides some very risky assumptions. Discussing the conditions of satisfaction can help identify these assumptions so that the developer and customer can be more confident that they are in agreement and that the contract covers the work expected. An example of this can be seen in this user story:

5. As a user I can highlight portions of an athlete's profile so that when I next view the profile I can see items of most interest to me.

At first glance this story seemed clear. The developers were planning an interface that would allow a viewer to select a yellow highlighting marker tool and indicate as many regions as desired. However, our conversations with the customer pointed to more work than that. Like all of our discussions to flesh out details of a user story we asked the customer questions like:

- How will we know we're done?
- When we demo this story to you what would you like to see so that you know we've done what you expect?

This led to a discussion of requirements that the customer hadn't even thought about. Some of the highlights should work precisely as we'd planned, but now the customer also wanted "global

highlights” made by one user but visible to all. This led to a discussion of security privileges (should all users see all highlights?) and other topics. In the end we settled on a yellow highlighter tool for private highlights and a blue tool for global highlights. The story and its conditions of satisfaction appeared in our contract as:

5. As a user I can highlight portions of an athlete’s profile so that when I next view the profile I can see items of most interest to me.

- Two types of highlighting—yellow (private) and blue (global)
- Highlights made during one session are visible in later sessions

Naturally, collecting these conditions of satisfaction can be time-consuming. This is somewhat mitigated by the fact that we are not trying to drive every bit of uncertainty from the requirements as that would be impossible. Instead, we’re trying to drive out sufficient uncertainty that each party to the contract can feel comfortable with the amount of risk being assumed. In my experience taking user stories down to the level of conditions of satisfaction is often useful on the first contract between a developer and customer. For subsequent projects—if the two have learned how to work together and have established a level of trust—contracts may only need to include the user stories.

Of course, there’s more to writing a good contract for an agile project than just including conditions of satisfaction for the user stories. A good contract will also align incentives between the parties. Additionally, the team must have a good approach to estimating that provides a high likelihood of a profitable project. Starting with user stories augmented with their conditions of satisfaction, however, is a good start.

Posted: September 1, 2006

Tagged: user stories, sprinting, teams, estimating, sprint planning, customers, agile processes
