Professional Scrum Trainer

ScrumCrazy.com Services

- Scrum Training
- Scrum Coaching
- Technical Coaching
- Quick Advice/Assessment
- Details of Coaching and Consulting services
- Contact Us

Story Testing Patterns Summary

| Edit | 0 | 0 | 26 | ... |

Pattern	Generally Good For...	Generally Bad For...
"Test that..."	- Beginning Story Testers - Simple Tests - Tests hard to describe using the other patterns	- Experienced Story Testers that know a better pattern. - Tests with a lot of setup logic or behavior logic(try a different pattern) - Tests where behavior depends on numerous test inputs
Given/When/Then	- Tests that require - a lot of preconditions or setup, OR - setup that is important or easily forgotten - Tests that have a specific, non obvious trigger - Tests where there are few expected outputs	- Tests that have unimportant/simple /obvious preconditions - Tests where there are multiple different inputs and multiple different outputs - Tests where a single Given/When/Then only describes one of numerous very similar test scenarios
Specification By Example - Conceptual or Concrete	- Tests that have numerous: - Inputs that affect output behavior - Outputs/expected behaviors - Tests where it's important to test a lot of different data scenarios - Tests where the trigger event is somewhat obvious - Any test where it seems like a table would be useful to: - describe the test better, or - help explore all of the possible inputs and outputs for a test.	- Simple tests - Tests that are more about verifying simple UI behavior - For instance – "Test that an error message is displayed when the user enters an incorrect password." - Test where there is really only one input or precondition
Bullet Points	*Teams that are highly co-located with PO* - Stories that are very small(2-3 days) - Tests that are very simple - Tests with fairly obvious expected behavior	*Distributed Teams* - Stories that are large (which is a bad habit anyway) - Tests that are not simple - Tests with non-obvious expected behavior
"Test With..."	- Teams that are highly co-located with PO - Stories that are very small(2-3 days) - Tests that are very simple - Tests with fairly obvious expected behavior	- Distributed Teams - Stories that are large (which is a bad habit anyway) - Tests that are not simple - Tests with non-obvious expected behavior
Flow Charts	Tests where the flow of behavior is very complex, and easier to represent with a series of successive questions/answers	Generally bad for everything else.
State Diagrams	Tests where a system object can go through numerous (often workflow related) states	Generally bad for everything else.

Remember to strongly prefer index cards(5x8), wiki's, and whiteboards over ALM tools and other electronic documents/tools.