XBOSOFT
Software Quality Improvement

Scrum Test Planning

What goes into a scrum test plan?

With scrum, one of the popular flavors of agile, the entire team works together with one goal, working software, at the end of each sprint. Scrum teams are ideally collocated, so the flow of information is uninterrupted, but does this mean there is no need for a test plan?

In traditional software development project life cycles, test plans play a very important role. In the initial phase of the project, the testing team sits together and discusses the testing scope based on the requirements specification to ensure that all critical features mentioned in the specification will be tested. The testing team also discusses who tests what and the timing of each test phase in coordination with development. The output of this process is the test plan. This whitepaper discusses test planning as part of scrum, a popular agile methodology. As part of the paper, we discuss test planning and scrum in general, and then testing methodologies and test planning particular to scrum.

Test Plan

What is a test plan? A test plan is a document that systematically describes the testing approach. The team needs the test plan because they want to make the following testing components clear:

Test Coverage: What do we test and what is the extent of our testing? Typically included are functionalities of the SUT (System under Test) and any other regulatory standards that the system must comply with (i.e. information privacy, safety standards). In traditional waterfall oriented development projects many parts of the test coverage and methodologies are decided at the beginning, before development work starts.

With so many organizations moving toward agile, many are changing their processes and are wondering how QA fits into the agile development process.

In Agile development, Software is developed in incremental, rapid cycles. This results in small incremental releases with each release building on previous functionality. So rather than huge test plan documents, we have incremental test plans as well.

Test Method: What methodologies shall be used during the testing period and what process is applied? How will the test cases be generated? Do we need a smoke test before every regression test? Do we automate our regression tests? How to decide when the testing phase is finished? By the number of remaining defects or by timeline? What are our acceptance criteria or how do we make the go/no go decision?

Test Responsibility: Who tests what? How are results reported? This is not only about the role in the testing organization and the task assignment, but may also include other responsibilities such as collecting and maintaining test data.

This is a simple way to categorize information in a test plan document. However, in agile, and in particular scrum (Agile also includes other methodologies such as Kanban and various hybrids), a formal test plan document is not always necessary. Scrum handles elements in testing plan in a different way.

Scrum Basics

What is scrum? Scrum is one of many agile methodologies that values customer involvement and fast response as a key element and success factor. Scrum allows a team to quickly adapt to changing requirements from customers and involves the client as early as possible in the development process by delivering finished pieces of software every 2-4 weeks thereby facilitating user feedback early on, rather than at the end of the development cycle. Some common concepts in Scrum include the product owner, scrum team, user stories, and sprint.

When requirements change, the scrum process gives you the ability to adapt to these changes. So you can't have a huge rigid test plan. Scrum testing needs to be as agile as the requirements themselves.

Product Owner and Scrum team: In a scrum team, the customer either directly participates as the product owner or has a representative standing in as product owner. In many organizations, the product manager takes on this role. The product owner is responsible for collecting requirements from stakeholders and explaining these requirements to the scrum team. The scrum team is responsible for implementing the requirements as delineated and agreed upon during each sprint planning meeting. The scrum team consists of all team members including requirements analysts, business analysts, developers, testers, etc.

User Stories: A user story in scrum is a way to express requirements from the customer point of view. User stories follow the format 'As a (user) I want to….'. There is an entire science behind writing good user stories that are not only quick and easy to write, but also easily understood by development staff. This simple and direct format is also a good basis for testing.

Sprint: A sprint is a fixed working period of between 2-4 weeks and is actually a re-packaging of a similar concept called time-boxing. During each sprint, the scrum team focuses on finishing the user stories that were included in the sprint backlog (work items for that 2-4 week time period). At the end of any given sprint, a working product should be delivered by the team to the product owner that encompasses the functionality and features in those user stories.

Sprint Planning Meeting: At the beginning of each sprint, all team members together plan the work for the upcoming sprint. During the planning meeting the team:

a) Receives guidance and explanation of each user story from the product owner, who also answers questions. After the scope of the stories becomes clear, the team then estimates their effort: i.e. the required time to develop and test this functionality. The testing method and testing effort is also included in the estimation session. This is where agile poker comes in for effort estimation.
b) Moves selected user stories from the product backlog to the sprint backlog (work to be completed in that sprint) based on knowledge of past sprint velocity and priority.
c) Decomposes the user story into detailed tasks, which includes the tasks for writing test specifications and test execution.

Testing in Scrum

Test Coverage in Scrum: In scrum, all customers' requirements are documented and stored as user stories, and prioritized by the product owners and put onto the product backlog. All user stories should be developed and tested against the requirements of the customer. Only the user stories that must be finished in the next 2-4 weeks are moved from product backlog to the sprint backlog, so the testing team spends most of their effort on those particular user stories. Therefore, there is not only '**general test coverage'** for the whole project, but also '**dedicated test coverage'** for each particular sprint.

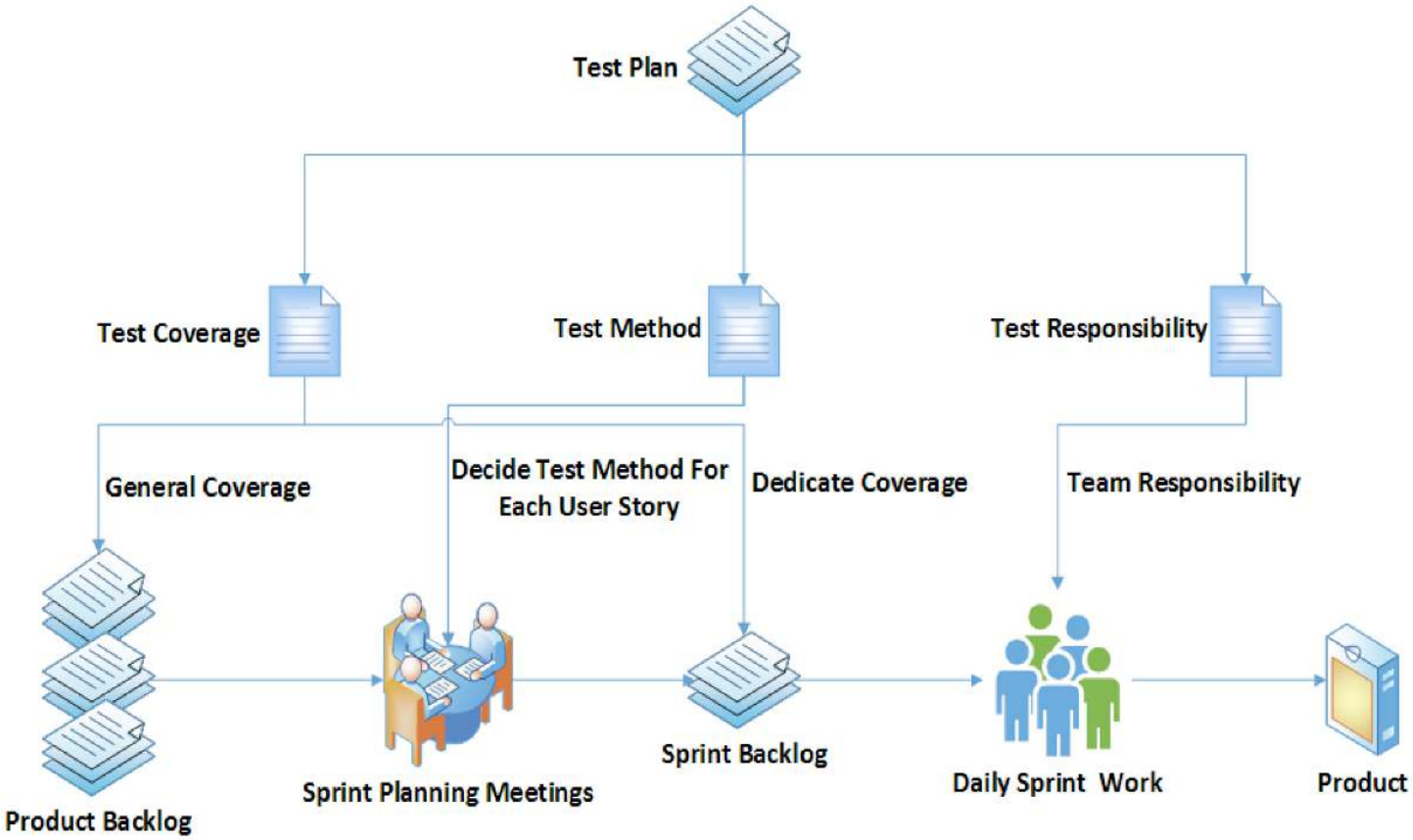
Test Method in Scrum: As mentioned earlier, the scrum team analyzes each user story and breaks it down into detailed tasks. For example, a user story which concerns both the UI in the browser as well as the back-end server application may contain three development tasks: develop UI, connect the web service and develop routines to maintain the database. In the meantime, different testing tasks are also created. A manual testing

When implementing Agile Testing, keep your feedback loop short. As part of your process, talk about defects immediately with developers. If you need clarification regarding requirements, talk to the Product Owner. Then adapt your test plan and testing accordingly, on the spot.

"In agile, everyone takes ownership at driving toward a high quality product, but that doesn't mean everyone does everything. People should still do what they are best at doing, but also be willing to jump in and help out where they can, and be cognizant of what need to be done outside their own role."

task is created for testing the UI and web service connections between the internal application and an external service while a simple automation test may be applied to the database testing task. Some tasks may aim at refactoring code and reducing technical debt without implementing any new functionality. In this case, sometimes the most effective testing method for these stories are automated regression tests. To summarize: in scrum, testing methods used depend on the nature and elements of the user stories.

Test Responsibility in Scrum: 'Traditional' development teams usually include a dedicated team of testers. This is not the case for pure scrum teams. In scrum, all team members are striving for on-time and quality delivery and there is no clear segregation or differentiation between tester and developer roles. Everyone should pick up what he/she can do. That is in theory, but in reality, experienced testers usually bring a lot of additional testing experience that the developers do not have. Additionally, testers bring a different viewpoint than a developer and they are also usually able to act as the bridge between developer and customers (product owners). Members who have testing experience can easily understand the user stories from the customer view while they also understand some development concepts. Therefore, experienced testers can help a scrum team eliminate miscommunication and gaps that often occur between the customers and the developers.



Conclusion

As we've discussed, traditional/waterfall projects usually dictate the need for a detailed test plan document because they want to set up a very clear view of the testing scope, testing method and testing responsibility in the beginning of a project. However, scrum, one of the many flavors of agile, involves and applies these concepts in a more incremental way which can be more effective and natural in that test planning actually occurs with each sprint. The sprint planning meeting helps the team decide the testing scope and method for each sprint while adapting to the current situation. Therefore, there is less necessity to create a formal test plan document at the beginning of the project. Rather, each sprint has testing tasks as part of the sprint plan. Additionally, one of the most important scrum credos is that testing is everyone's work and that quality is the responsibility of everyone.