



Agile: Much more than a development architecture.

Summary:

This document shows the advantages of Agile for QA and Testing, and how important is the communication in this type of environment.

Agile: Much More Than a Development Architecture

by Benjamin R. Waymire

Agile development is nothing new. In its purest form, Agile offers a simple, manageable architecture for development teams to rapidly deliver *project value* in a constantly evolving business and technical environment. But as we all know (and have learned from the more traditional waterfall management model), complex management methodologies for development teams seldom flourish without challenges or even failures. Using Agile, today's organizations are working to drastically reducing the overall risk associated with software development, but this requires much more than establishing a simple architecture.

So what does it require? Agile requires discipline and self-management. Organizations not only have to have a process, they must actually follow it as well. Development, business analysis, QA, and testing teams (called scrum teams) must be able to function and communicate as if they are all in the same building. In today's business world, teams that are critical to the overall success of a project are often not in the same building. Given this, resources and communications are more critical than ever.

Here are a few vital success components that are often overlooked initially in traditional Agile environments:

- Allocation of resources must be defined very clearly.
- Proactive communication is critical. Organizations with an off-shore presence, especially, must have a well-prepared communications plan for projects:
 - Who is involved?
 - What are the specific roles and responsibilities?
 - What are the project dependencies?
 - What is the customer's business environment?

Ultimately, there is no analysis so in-depth or discerning that will produce the same quality results as simply leveraging customer feedback and requiring process-driven, open communications.

Agile vs. Waterfall

Agile methods differ substantially from the traditional waterfall management methodologies and are far superior. For example, Agile teams do not need a year's worth of requirement analysis and design before coding begins. Instead, the majority of the testing can be automated easily in fixed requirements cases. Even in environments with the most fluid requirements, customers will almost always

exceed earned benefits by investing in automation from the beginning. A cost/benefit analysis is well advised to truly quantify automation.

A Case Study: Communications and Agile

Let's briefly examine a recent case study of which I was involved. A customer migrated from Waterfall to Agile having a corporate development team ratio of approximately 75% onsite to 25% off-shore. Internal business analysts worked directly with the customer business analysts and product owners to review Product Backlog Items (PBIs, or change requests). Upon review completion, the PBIs were supported and appropriately prioritized and QA and the business analysts agreed on the test scenarios and technical clarifications. During actual test execution, the analysts worked closely with the business to get appropriate prioritization for outstanding defects and their impact on the overall business workflow.

What was the end result? Significant cost reductions for the program's quality assurance with absolutely no adverse effect on...well...quality! Adequate testing approaches were also developed for integrating systems with different delivery mechanisms (Agile vs. Waterfall).

Why was this the end result? The team incorporated a superior architecture (Agile), established close-knit communications, project dependencies, and expectations from the beginning. When proactive communication covers all aspects of the project cycle (analyses, preparation, acceptance, deployment and maintenance) within the Agile environment, success can be achieved in even the most highly fluid Agile environments – and in limited timeframes. I cannot understate how important this is, and also how often quality communication in Agile is simply neglected.

Final Thought

It is important to be clear on this point: Agile will not turn a failing project into a successful project. However, combined with an effective communication plan it will help an organization quickly identify that a project is, in fact, failing and help figure out why it's failing. Fundamentally, this is how an organization can get back on course before it's too late; saving precious time and big dollars.

About The Author

Benjamin R. Waymire is a freelance journalist and subject matter expert in Software Development, Quality Assurance and Testing methodologies for Allied Testing, a leading specialist QA and testing firm with the focus on the capital markets, trading, and finance industries. To learn more about Allied Testing's products and services, please visit <http://www.AlliedTesting.com>.