Best Practices for Agile and Outsourced QA Testing

 tirrellp (http://www.payton-consulting.com/author/tirrellp/)   Testing (http://www.payton-consulting.com/category/testing/)

Tweet

(http://www.payton-consulting.com/wp-content/uploads/2015/03/shutterstock_258929474.jpg)

Outsourcing was one of the most powerful technology trends of the 90s and 2000s. Its promise was clear: Knowledge work with knowledgeable people at a lower cost than having the work done in the United States. Many larger companies got the message at the same time and outsourced en masse. As of 2003, McKinsey reported in its article *Offshoring: Is it a win-win game (http://www.mckinsey.com/insights/employment_and_growth/offshoring_is_it_a_win-win_game)?*:

> Offshoring will allow the US to capture economic value through multiple channels: Reduced costs, new revenues, repatriated earnings, and redeployed labor

Meanwhile, at around the same time, Agile and Scrum were gaining steam, albeit starting from smaller firms and working up the value chain. This brings us to today where organizations want to become more Agile but still continue to work in a geographically distributed environment.

While Agile purists tend to be against outsourced QA testing, its a stark reality for many companies. The decision to outsource may have long predated the decision to transition to Agile. Given this constraint, its much more productive to talk about how some companies are getting the best of both worlds. Here are 3 best practices for getting the most from your Agile efforts and your outsourced qa testing.

(http://www.payton-consulting.com/wp-content/uploads/2015/03/shutterstock_33331030.jpg)

Improve communication and decrease hand-offs

Use video technology to bring the distributed team together. While email is perfectly fine to fill in the gaps and provide additional detail. There is nothing that can replace face to face interaction. Using tools like Google Hangouts or Skype, you can approximate the high bandwidth communication that you get with co-located teams.

A headache associated with geographical distribution is meeting times. A solution is to alternate the time of the team's daily standup every sprint. One sprint use a time that is convenient to the onshore team (and let offshore suffer late nights/early mornings). The next sprint use a time that is convenient to the offshore team (and let the onshore team suffer late nights/early mornings).

Julian Clayton, VP of Product at FieldLens (http://fieldlens.com/), a communications platform for the construction industry, remarks, "do not underestimate the need for
an internal QA lead. Even with only moderately complex products it can be nearly impossible for an external lead to keep up with the day to day
changes in the fast pace of an agile environment"

(http://www.payton-consulting.com/wp-content/uploads/2014/11/shutterstock_200797052-e1420915896685.jpg)

Create a clear separation of testing concerns

A challenge with marrying Agile practices and offshore arrangements is that when there is a single deliverable that needs to be developed (onshore) and tested (offshore), you are adding in another step to be able to deliver a "potentially shippable" product. With every issue, you add 24 hours of delay. No good.

The alternative is to create a clear separation of testing concerns. Engage onshore developers for **verification** testing via automated unit tests (http://en.wikipedia.org/wiki/Unit_testing) (using an xUnit (http://en.wikipedia.org/wiki/XUnit)framework). Engage offshore testers for **validation**

testing (e.g., performance/integration/exploratory testing). This will eliminate waiting for initial verification testing, but still ensure the more robust validation testing still happens, using your offshore testing experts.

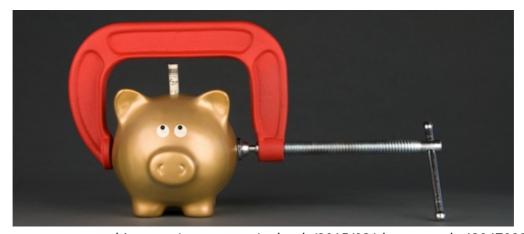(http://www.payton-consulting.com/wp-content/uploads/2015/03/shutterstock_119131810.jpg)

Enforce modern engineering practices

Companies tend to focus more on "process" based solutions when there are many technical solutions to enable you to get the most from your offshore partnership. Two crucial engineering practices are continuous integration and automated unit testing.

Continuous integration describes a practice by which code is integrated and a new integrated build is created upon each check-in. This ensures the system is always stable and any integration related issues are immediately visible. Tools such as Jenkins, Bamboo, CircleCI, Team Foundation Server can help with continuous integration.

Automated unit testing is a practice whereby developers create code to test the code they have written. This creates an ever-growing regression suite that grows as the code base grows. When you combine it with continuous integration, you create an extremely robust "always on" quality framework that runs and reruns every automated unit test every time any code is checked in.

If you want to enable the best practice of "follow the sun" workflow, continuous integration and automated unit testing are must-haves.

(http://www.payton-consulting.com/wp-content/uploads/2015/03/shutterstock_43047022.jpg)

Doing more with less

Agile and outsourcing are not as diametrically opposed as they first appear. Both practices reflect companies' attempts to do more with less. While one is rooted in predictive planning and the other is rooted in adaptive planning, there are still ways to make them work better together. ∧