JOIN OVER 100,000 PROS! Join Log In | Member Benefits

STICKYMINDS

Agile Testing DevOps Security AI

# What Agile QA Really Does: Testing Requirements

ARTICLE

By Steve Berczuk | November 28, 2010

SUMMARY

Teams transitioning to agile struggle with understanding the role of QA and testing. Much of what you read about agile focuses developer testing. Every project I've worked on which had good QA people had a much higher velocity, and higher quality. And there are limits to what developer Unit Testing can test.

Teams transitioning to agile struggle with understanding the role of QA and testing. Much of what you read about agile focuses developer testing. Every project I've worked on which had good QA people had a much higher velocity, and higher quality. And there are limits to what developer Unit Testing can test.

In an "ideal" agile environment a feature goes from User Story, to Use Case, to implementation. With tests along the way, you should be fairly confident that by the time you mark a story done that it meets the technical requirements you were working from. If someone is testing your application after this point, and the developer tests are good, it seems like they should be testing something other than the code, which has already been tested.

Even though it's important that your QA team does not become the place where "testing is done" as that will sidetrack an agile project very quickly, it is possible that the QA Team is testing (ideally in collaboration with developers) things that the team decided that developers could not test completely. Also, in practice, problems sometimes slip through developer tests, and the QA team is then testing the developer tests to give feedback on the types of things the developer tests need to be better at.

Aside from "catching the errors that slipped through" a QA team doing exploratory testing is also exploring novel paths through the application, and may discover problems during

We use cookies to ensure you get the best experience on our website.

Accept

Read More

*In effect, exploratory testing is testing the requirements for completeness.*

If you have a QA team, and they are doing exploratory testing, they are really testing:

- Requirements: Finding interactions between features and components that were not defined or understood when coding and developer testing began.
- Developer Tests: Identifying where developer tests were not as good as they could have been (and how they could be better). This is a good use of automated "integration" tests.
- Systems tests that might be hard to test in a developer context.This might be the one set of tests that are the primary domain of QA.

And, as I've said in other places,  QA team can be extremely valuable testing user stories and requirements before they make it into a sprint backlog, by providing feedback about testability and precision. The QA team can also add value by testing how useful a product that meets the specification actually is.

Of course, this is an idealization.  But if you are disciplined about doing developer testing, you can still get a lot of value from exploratory testing, whether the people doing them are dedicated QA people, or people who are filling the role.

**Topics:** agile transition,

## ABOUT THE AUTHOR

**Steve Berczuk**

Steve Berczuk is a Principal Software Engineer with experience as a manager, Scrum Master and technical lead in Boston, MA. The author of *Software Configuration Management Patterns: Effective Teamwork, Practical Integration*, he is a recognized expert in software configuration management and agile software development. Steve is passionate about helping teams work effectively to produce quality software. He has an M.S. in operations research from Stanford University and an S.B. in Electrical Engineering from MIT, and is a Certified ScrumMaster. Contact Steve at steve@berczuk.com or visit berczuk.com and follow his blog at blog.berczuk.com.

## UPCOMING WEBINAR

We use cookies to ensure you get the best experience on our website.

## LETS HANG!

## FEATURED RESOURCES

SECURITY

**Shifting Security Left: DevSecOps eGuide**

White Paper | StickyMinds

Log In or Register to post comments

## USER COMMENTS

**0 comments**

## YOU MAY ALSO LIKE

SECURITY

**Agentic AI Opens Up New Security Flaws**

Nahla Davies | August 27, 2025

We use cookies to ensure you get the best experience on our website.

SECURITY

**Countering the Threat of Ransomware-as-a-Service (RaaS)**

Nahla Davies | August 22, 2025

SECURITY

**Building Trust in Blockchain: The Role of Smart Contract Testing**

Ekaterina Egorova | November 18, 2024


**Join the crew for FREE! Access exclusive content, contribute, and more.**

See All Member Benefits





We use cookies to ensure you get the best experience on our website.