

# The New Role of the Agile Tester - The General Contractor

**Melissa Tondi**

melissa.tondi@gmail.com

## Abstract

One of the most misunderstood testing activities within the SDLC (Software Development Lifecycle) has been Regression testing. Historically, we build in days—or sometimes weeks—to account for regression testing, usually at the end of a sprint or while preparing for a big release. Most of the time, it is considered solely QA's responsibility. In some cases, the delivery team may swarm the work and execute test cases at QA's direction to accelerate the duration of the cycle, but the team usually expects QA to "own" the effort.

## Biography

Melissa Tondi has spent most of her career working within software testing teams. She is in Quality Engineering leadership at E\*TRADE and a Principal Consultant at Disrupt Testing, where she assists companies to continuously improve the pursuit of quality software—from design to delivery and everything in between. In her software test and quality engineering careers, Melissa has focused on building and organizing teams around three major tenets—efficiency, innovation, and culture – and uses the Greatest Common Denominator (GCD) approach for determining ways in which team members can assess, implement and report on day to day activities so the gap between need and value is as small as possible.

# 1 An Outdated Idea

The tradition of QA being solely or primarily responsible for regression testing is outdated. Don't get me wrong - during this time, Developers are helping where they can -likely supporting QA's efforts by being available to assist in fixing bugs testers find and perhaps executing test cases at QA's request. However, unless or until there are bugs to be fixed, Developers are usually disengaged from the detailed Regression testing and the valuable information that is being gathered by QA during this time - arguably, the most important part of this activity.

Meanwhile, during the Regression testing cycle QA is able to gather new information that the rest of the team may be missing, but because Developers are working on new work during this time and QA is heads-down in Regression, we've removed QA from being involved in any of the valuable collaboration efforts that happen to kick off new work. QA then must play catch-up to work they should have been integrally involved with from the start. The more unique Regression cycles scheduled in this outdated way, the more we create a chasm between Developers and QAs. This results in lost time, inefficiencies and ultimately creates a crutch.

Regression testing can and should be owned by either the delivery team as a whole, or Developers. When there's no firm charter of QA's responsibilities, a gap is created within the delivery team and, more than likely, QA will fill that gap and assume the ownership for all Regression testing. That's bad for QA because this takes us away from other important activities – like testing and interacting with the software to advocate for the users. It's also bad for Developers because the crutch of QA creates a false sense of ownership where they may no longer be responsible for a deeper understanding of how code additions and/or changes may affect other areas of the software.

It's time for a new way of thinking on what QA's role is within a delivery team, address the Regression testing crutch and other outdated assumptions on QA's responsibility, and recognize when that assumption becomes a crutch.

## 2 The New Role of QA – The General Contractor, Not Inspector

As IT professionals, we often *loosely* compare software development to building construction. In the construction trade you have a general contractor (GC), skilled tradespeople, architects, project managers, inspectors, and general laborers.

For example, a skilled tradesperson, like the plumber, comes into a building and ensures that the seals are tight, the drains are free, that water flows in and out of the structure as expected, and that everything is up to code. The structure of the design is assumed to be sound. Plumbers are not expected to provide input into the design, color or features; they focus exclusively on the plumbing.

If you compare Developers to skilled tradespeople, there are many similarities. They are usually deep experts in their own areas, but rarely have that same expertise across other trades. Each trade has an inspector who are trade experts, and there are usually many inspectors involved in a project. The

inspector role is the role that ensures that trade code has been followed and serves as an independent and separate entity that certifies that something has met code.

In IT, the long-held perception is that QA is the inspector – the role that checks and signs off that requirements are met. However, they are not experts in every area the way a team of construction inspectors are. I'd like to introduce another opinion when making the comparison – QA as a General Contractor (GC). Think about it: QA pros are familiar with multiple components of the software development cycle, QA, like the general contractor, works with the owner/user throughout the course of the project and helps vet out requirements and specific needs every bit as much as a general contractor does when working with a homeowner. In addition, the GC ensures that the desired level of quality has been met, including design and user experience and offers timely and valuable feedback back to the tradespeople that add to the overall success of the project.

When comparing QA with the inspector role, it instills a false sense of non-ownership with Development and perhaps assumes that they cannot be trusted to check their own work against requirements. If another team (QA) is in place to check that activities have been done correctly (or according to building code) by tradespeople, then we may remove accountability and create an “Us vs. them” mentality – and ultimately a crutch.

### **3 Start Acting Like the General Contractor**

The role of an inspector is to ensure the code has been followed; QA, as General Contractor, ensures that the desired level of quality has been met, including design and user experience., and to provide recommendations on how to improve. QA's role is to not necessarily make sure the electrician is installing each outlet correctly (although, they're certainly capable of this), but to provide valuable feedback on the functionality, experience, and that the needs of the client and industry have been satisfied. QA as GC provides consultation through most milestones – refinement, acceptance criteria, retrospectives, etc. They are also able to examine the situation from a 10,000-foot view and see the entirety of the project, incorporating all aspects that affect it. As general contractor, QA depends on the expertise of each “trade,” in this case the owner of each component of the software cycle, and fosters relationships between each critical member of the team. In addition, the General Contractor also helps guide conversations within the team and consults with them to understand what they can be doing on their own to ensure quality, collaborate on other quality metrics, and strategically promotes these topics before work is begun.

That's why QA , using the example above, should not be solely responsible for regression testing: QA is not an inspector to ensure our team counterparts did what they were supposed to do and that it's up to code, but explores the experience from a broader, more user-based vantage point.

When you take QA out of the analysis and refinement of new work and have them focus solely on an activity like Regression testing, you silo them to the Inspector role only and place them in a situation where they cannot be consultative when new work is kicking off. By shifting QA's role to that of a GC, you let the process and workflow naturally tap in to their general, and sometimes deep, expertise of the “tradespeople” but also let them flourish and advocate for not only the software's functionality, but the overall experience as well.