The Tester Who Came In from the Cold

by Lisa Crispin

The stereotypical tester is a lonely, unloved soul waiting for code to drop "over the wall" for testing. She skulks about, searching vainly for someone who cares about actually fixing the bugs. QA teams often labor in their own little gulag, far from the programmers writing the code and even farther from the business people who thought up the requirements. Relationships between testing and coding teams often border on frosty.

Luckily for all us testers, the wall has started to come down, especially in organizations that have embraced agile principles, values, and practices. Countless developers have been "test-infected" and are eager to collaborate with professional testers to deliver the best possible product. The "whole team" approach—where everyone involved in producing software takes responsibility for quality and testing—has proven effective in creating high-quality, bug-free, testable code. Testers are finding the courage to break down the wall and contribute their valuable skills throughout the development lifecycle. Working closely with stakeholders and developers, testers find opportunities to learn and apply new tools and techniques.

As a testing and quality assurance professional, what can you do to come in where it's warm? As a manager, especially in a shop that's moving toward agile development, what can you do to help your QA team enjoy participating as valued members of a quality-focused development team? As a programmer, how can you get testers engaged in your development process?

Find a Common Language

Some of us may have a deep understanding of the business domain, but a tenuous grip on object-oriented design concepts. Others may dream in Java, but only know a narrow part of the application on which they're working. How can we collaborate if we can't communicate?

Examples are a good way. Ask business experts to illustrate desired system behavior with concrete examples. Have them draw on the whiteboard while you and your fellow developers, QA specialists, and DBAs ask questions. Explain a piece of functionality to a programmer using an executable test.

"Testers are finding the courage to break down the wall and contribute their valuable skills throughout the development lifecycle."

Make Quality the Goal

Five years ago, my team's manager asked us to deliver the best possible software—code we'd be proud to show off. He told us not to worry about deadlines or quantity, but to take the time to figure out how we could do the best job. This was hard to believe, but with mutual trust and a lot of time, it happened. Since we took time to do things right, we were able to "work smarter" and save time over the long term. We also had more fun.

Testers can't come in from the cold without support from their managers. It takes time and training to learn how to work with people in other roles and to learn new technical skills. It takes a strong manager to let the team self-organize, solve its own problems, and continually find ways to do a better job.

Address Physical Logistics

For people to work together, they have to be together. If you're on a test team that sits on a different floor from the developers, get up from your desk and go find them—or lure them to your area with chocolate. Plenty of teams have ripped out walls or tiny cubes and made areas where they can collaborate.

This is tougher if you're in different time zones, but technology has come to our rescue. Experiment with the ever-improving tools that allow remote collaboration. Experiment with working different hours so you can at least talk or videoconference with remote team members.

Let Go of Your Fear

Some testers fear losing their identity if they suddenly become part of a team with many roles working together. Some people see developers writing unit tests and think, "If the programmers are doing the testing, what's left for testers?" And it's perfectly reasonable for a QA manager to look at the new "agile" organization and wonder, "Is there a place for me?"

No matter what your development methodology, you still need good people to produce good software—and you need a mix of skills. Testers still need managers to make sure they have the training and support they need. Programmer testing does help deliver defect-free code, but we still need to find out what the business needs and make sure the code provides it. It takes a village to deliver value to the business. Knowing you're not alone will give you courage.

Get Up and Try It

Testers, get up from your desk right now and find something you can do to help your team—virtual or otherwise. Ask a programmer if there's some testable chunk of code ready to look at. Ask a domain expert for some examples of how the new feature will be used. Get online or go to a local user group meeting and learn a new skill that helps you add value to your team.

Those of you who aren't testers, think about how your job might be made easier and how your product might be made better if testers participate from the start of each development cycle.

Break down that wall! {end}