Considering whether or not your software company should hire a dedicated team of testers? Here are the Top 5 Wrong Reasons why you shouldn’t.

We Have a Beta Program

Some people feel that the best way to debug a system is to ship it to your customers and wait for trouble tickets. I don’t know how things work in your industry, but as far as I’m concerned, finding new customers is hard enough. I definitely don’t want to make matters worse by shipping them buggy software.

I’ve spent the last five years in the telecommunications industry where the standard for reliability is 5-9. If I even suggested to my customers that my software was in its beta phase, they’d hang up on me immediately. I’m positive the same is true in any industry.

Imagine if I published articles that were not only full of grammatical errors but also missed a complete sentence here and there. Would you bother sending me an email to inform me that my articles are defective? Would you recommend my newsletter to your peers? I didn’t think so! So imagine how your customers feel when mission critical software breaks and crashes on them.

Developers Will Get Lazy

Some managers feel that developers will get lazy if they know someone else is responsible for testing their code. Developers are either lazy or they’re not. Someone who takes pride in his work will rigorously test his code regardless of whether or not you have a dedicated team of testers.

If your developers are lazy, don't blame your testers. Blame the developers! Not hiring a team of dedicated testers won't improve the situation. It'll actually make the problem worse because your sloppy developer's code will find itself in your customer's hand instead of your testers' lab.

We Can't Afford Testers

If you can afford testing, then you can afford testers. Basic economics - and common sense - prove that it’s more economical and efficient to hire specialists than generalists, as long as you can keep the specialists busy.

If it takes you 50 person-months to develop software, you will (statistically speaking) spend 25 person-months testing and validating it. Which do you think is more economical? Assigning 25 person-months of testing to developers or testers?
As explained in quality is job #1, one of my previous articles which lists reasons why software companies should hire a dedicated team of testers, professional QA people are more efficient at verifying software than developers are. It’s their job! Keep the 1:3 ratio in mind and hire one tester for every three developers, even if it means getting rid of your bottom-of-the-barrel developers to maintain a balanced human capital budget.

Testers Find Too Many Bugs

This excuse sounds more absurd than it actually is. I agree that in some cases testers report bugs that are valueless. Enter non-ASCII characters in one field, ASCII in another, stick a screwdriver in your DVD drive, do the hokey pokey and hit the enter key, and the system will return an error. Who cares? No customer will ever come across this scenario in a live deployment. (Actually, if I showed you some of the defects our customers are entering in our trouble ticketing system, you’d be surprised!)

If you think your testers are really finding too many insignificant bugs, give them guidance. Don’t ridicule their effort or disregard their problem reports. Explain the use cases you’re trying to satisfy and the known (and acceptable) limitations of the system.

We Can't Find and Keep Any Good Testers

I truly sympathise with those who bring up this point because it’s true. Good testers are really hard to find and the first-class ones often apply to transfer to your product development team. Still, this is not a reason to give up on hiring a dedicated team of testers.

Here are three suggestions to keep your testers on your QA team for as long as possible:

- When hiring testers, look for people who have previously held a QA role for at least a year. Beware of new graduates and others who will accept any position to get a foot your organisation’s door. Chances are more likely that they will ask for a transfer as soon as their probation period is over.

- Offer your testers a competitive salary. Some companies tend to pay their testers way less than their developers. As a result, testers request a transfer simply because they’d like to earn more money, and you can’t blame them. Level your pay scales and you’ll give them one more reason to stay on your QA team.

- Allow testers to develop their technical skills. Give them the opportunity to write automated test scripts, install and configure test networks, and take design/development courses. Providing these extra benefits will entice your testers to stay in their role for a longer period. And when they do decide they want to move to your development team, they’re going to be better prepared.

Conclusion

After reading dozens of opinions on the subject, I’m still convinced that having a dedicated team of testers is well worth the investment. You can disagree with me, but if you do, make sure it’s for the right reasons.

Luc Richard holds an MBA with a major in high technology. For the past ten years, he’s been managing the development of software applications.