



Consultants



Five risky testing mistakes

$$E[e^{-sX_{+,i,u}} | A(X_{-,i,u}) = k] = \sum_{i=1}^k \left[\frac{1}{2Li!} \frac{d^i}{dz^i} \Big|_{z=0} \left(\frac{G_p^*(z)}{z} \right) + \frac{1}{2L(k-i)!} \frac{d^{k-i}}{dz^{k-i}} \right]$$



Five risky testing mistakes

We are often called in by CIOs to serve as their ‘clean up crew.’ What this means is that a testing project has gone off the rails and a vendor needs to come in and get the project back on track and the major bugs cleaned up. Thankfully, many of the issues are preventable. Quite often, the problems arise from a lack of compliance to testing fundamentals. Below are five of the most common oversights in testing projects.

1. Neglecting testing altogether

In many IT projects, proper testing is regrettably an afterthought. More often than not, this is a sin of omission than commission. IT and business managers tend to focus first more on immediate activities within the software development lifecycle. Testing, on the other hand, gets less attention for a number of understandable reasons. As opposed to early development activities like requirements gathering and architecting, testing comes later in the software development lifecycle, subsequently getting less attention (i.e. the out of sight, out of mind effect). Furthermore, managers tend to focus first on leveraging internal tools and people versus outsourced and variable testing resources and; the testing function is traditionally not seen by many IT managers as an interesting activity or a priority undertaking. Neglecting testing is almost guaranteed to increase the number of defects in a new application or product and lead to lower than expected performance. CIOs can improve testing importance and function by paying more attention to the principles of software quality assurance principles and expressly linking poor testing outcomes with business and IT investment performance.

2. Lack of a testing strategy or plan

Testing is too vital an activity to undertake without a blueprint, metrics or end goal. A lack of a strategy and plan increases the chances the testing effort will lack the necessary scope, be rushed for time or lead to wasted resources. Ad hoc testing efforts can be effective at finding obvious bugs but will not necessarily find defects that are implied or implicit within the requirements. A good testing plan will also ensure the testing effort is properly paced – either too fast where defects are missed or too slow where the projects is unnecessarily delayed.

3. Insufficient time



Given the realities of the SDLC and business environment, it is very common for testing to absorb schedule delays and then be executed in a rushed, haphazard fashion. A lack of time increases the chances the testing tasks will be poorly executed and many bugs slip through. The key to maximizing software quality is to involve the testing team in the architecture definition and coding process as early as possible. Ideally, tests should be designed before coding begins. This gives the developer ideas of what will be tested so they can design and code the quality and performance enhancements.

4. Poor resources

High performance testing is part science and part art. Not only must testers be skilled in a particular application or tool but they should also possess an innate skepticism, problem solving bent and understanding of the industry and/or application domain. Unfortunately, many companies and testing firms focus too much on minimizing cost or getting rapid scalability at the expense of attracting and cultivating quality talent. Failure to adequately consider tester productivity and experience with a product's desired feature/functions will often result in lower quality testing and productivity.

5. Ignoring real life

Many project teams make the mistake of wrapping up testing once an application or product is released. This is problematic when quality is paramount. Testing should never be wrapped up too quickly before early user feedback can be gathered. Many serious bugs will emerge after the application is launched. Prudent companies will consider production releases as 'beta' quality and continue to run regression tests to ensure total quality.

A related mistake occurs when firms test only in controlled environments. As every company knows, customers will use the product in a variety of different circumstances, environments and conditions. CIOs should always look to conduct a portion of their testing in the 'real world.'

Of course, having enough testers and technology 'when needed as needed' is also crucial to project success. However, the number of bodies and tools is often not enough. Companies need to ensure they don't self-inflict their pain. As an experienced 'clean up crew', we have learned that testing performance can improve by paying closer attention to testing fundamentals and seeing testing as a vital part of a well-oiled quality assurance program.



For further information please contact:

Harry France
Sr. VP Sales
(519) 575-0314
hfrance@qaconsultants.ca