



For the record

Sarah Salzman, technology support manager for Compuware UK and Ireland, explains her view of usability testing and how how test automation tools can help with it

As testers we know why quality assurance is important. We understand that coding errors or glitches in the way one component interacts with another can bring down an entire application and result in an organisation losing thousands of pounds. Therefore it's not surprising that a great deal of testing effort is put into making sure applications are reliable, robust and will perform well under strain. However it shouldn't be forgotten that even the most resilient application, containing code written using best practices, could be perceived as failing if it is not designed in the way it works for the users. At the end of the day what's the point in building an application if people find it too difficult to work with? Most organisations understand this. In fact ease of use is often the first thing on a customer requirements list; however it's also a factor that is often neglected during the development cycle.

This is where usability testing can help, because it ensures applications are tested with the user in mind. Essentially, it is about looking at software design and trying to ensure applications are designed in a way that makes them easy to learn and use. This might seem like an obvious thing to do, but many development projects do not include any usability testing. This is understandable as both development and testing teams are under more pressure than ever to go live with applications, however if businesses want to ensure that they see the benefits of investing in and developing new applications; this is an area they simply cannot overlook.

If you need convincing on usability testing then simply consider the fact that we all come in different shapes and sizes. Our physical and psychological make up will impact the way we use a computer application. In some cases, culture and background may also be influences to consider. Maybe, most importantly, the experience of end users using an application may vary dramatically and this will without doubt impact how easy or difficult they find it to navigate and use. For example, an experienced computer user may decide to tab through the different fields in a CRM system, whereas a novice may choose to use the mouse. It is these types of differences in

human to computer interaction which can be captured and analysed in usability testing. Subsequently trends can be identified in relation to how people want to interact with the application and these trends can be then fed back to the development teams who are designing the user interface.

How do you carry out usability testing?

Opinions may vary on how to carry out usability testing, but to me the starting point should always be to find a group of users who are inexperienced with the application. At the very least you should bring in users who have the same experience levels of those people that will be using the application in the production environment.

You should also try to ensure the user group reflects the ultimate user base of the application in relation to factors such as sex, age, and background. However, in many cases it may be difficult to predict the attributes of the end users of the application being developed and therefore in this situation it is important to use a diverse range of users in the usability testing process.

Once you have your sample user base, you should analyse the functional specification for the application and pick a range of standard tasks end users will be carrying out in the application on a regular basis. If you use manual testing methods it is at this stage that you would ask your user group to carry out the tasks. As testers you would try to note down how users are approaching the task and how they are navigating through the application. Basically from start to finish you would note down the steps the user takes within the application to carry out and complete the set task. The problem with this is that is virtually impossible for you to note down every single interaction the user has with the application and even if you can this is very time consuming, especially if you are using a large user group.

An alternative method could be to create test cases for the tasks you have chosen to base your usability testing on and then use a test automation tool. Understandably, you may be wondering how you use a tool such as

this in usability testing, as the traditional use for these tools is to repeatedly test whether different functionality within the application works. Well, the trick is to adapt your testing tool and use it in record mode. As users carry out the test cases or tasks you have given them, the test automation tool will record every interaction they have with the application. The data recorded by the testing tool will be collected within a test script, which can then be analysed.

Testers should look for patterns in how the application is being navigated and utilised. Were there parts of the task that took users longer to complete than expected? If so, why was this the case? Is it because they found it hard to find a particular function in the application? Did they go to the wrong part of the application? Was key information not prominent enough? By analysing the test scripts and asking these types of questions, testers can highlight problematic areas in the design of the application and ask developers to refine or rectify these issues.

When do you carry out usability testing?

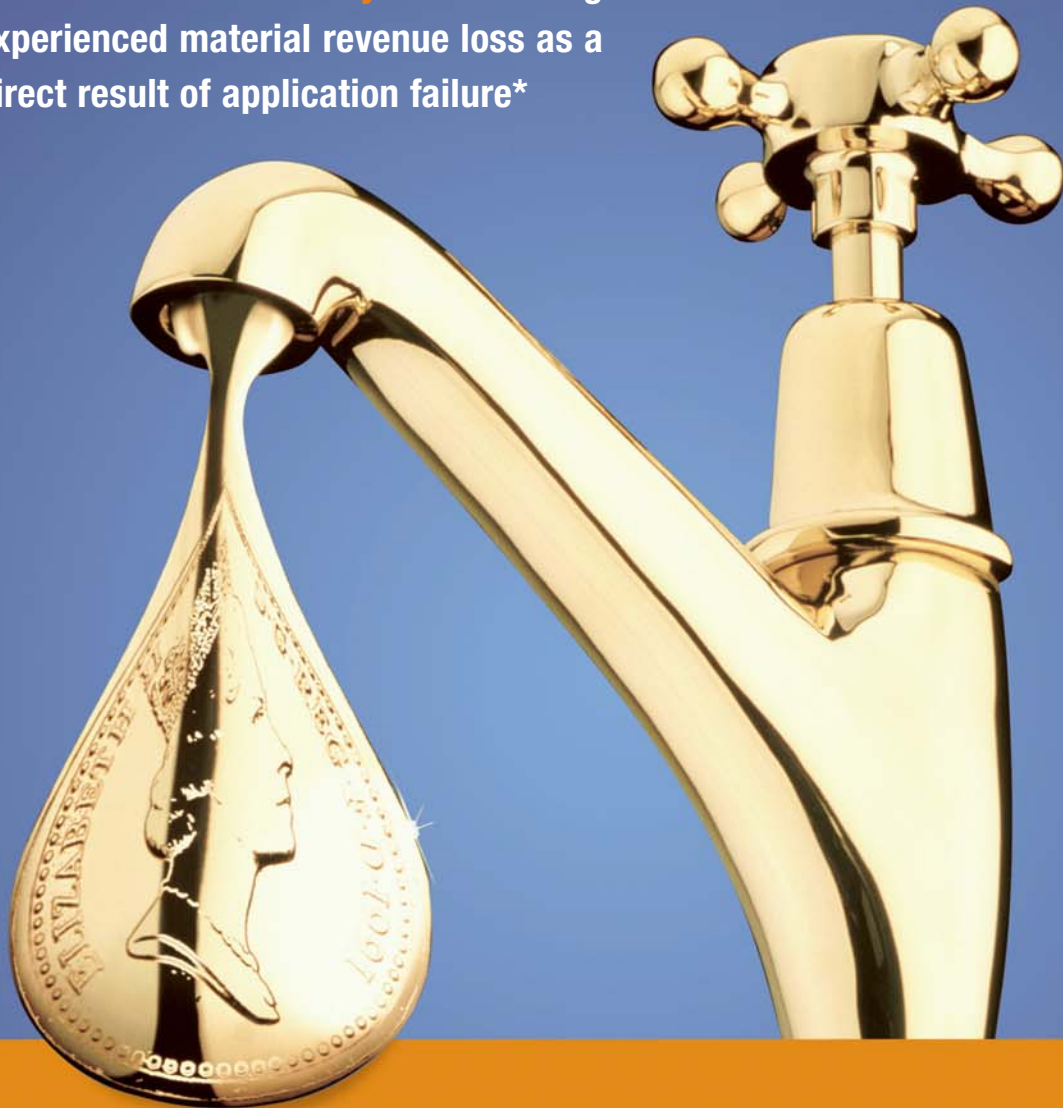
Usability testing should be started as early in the development cycle as possible. You should try to ensure that it is completed once a prototype of the application is built. It should then be repeated when major sections of the application are complete and again when the entire application is finished.

In addition, if the application is going to be maintained and updated then it is advisable to carry out usability testing before upgrades are rolled out to ensure changes to the way people interact with the application can be fed into the design of the upgrade.

Although usability testing takes time, money and resource, an understanding of how users interact with the systems we build is invaluable. Everyone involved in the development and deployment of an application should remember this and also never forget that the usability of a system will make or break the uptake of it. PT

Are your applications leaking money?

64% of IT Executives say Yes... having experienced material revenue loss as a direct result of application failure*



In today's competitive environment, IT departments need to release increasingly rich feature sets across complex distributed infrastructures. To reduce the risk of costly errors, analysts such as Forrester Research and Patricia Seybold, recommend an Automated Software Quality (ASQ) solution.

To learn how your software projects can be a third less expensive** and to download your ASQ information pack with Patricia Seybold white paper, visit:

www.compuware.co.uk/money

* Forrester Research - 2003 ** Patricia Seybold Group - 2003

COMPUWARE®
www.compuware.co.uk

