

Artikel vom Montag den 03.11.2008

## Die zehn größten Fehler im Software-Test-Management

*Fehlerhafte Software kann das Geschäft jeder Firma ins Stolpern bringen.  
Nur wer seine Anwendungen richtig testet,  
vermindert das Risiko und sorgt damit für mehr Stabilität.*

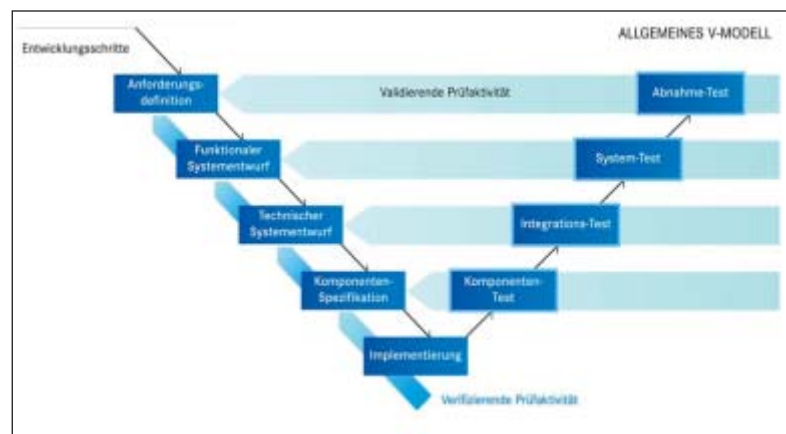
VON ERIC JOCHUM

Software muss heute höchsten Anforderungen genügen: die zugrunde liegenden IT-Infrastrukturen werden immer komplexer, die Qualitätsansprüche wachsen, und der Zeitdruck steigt. Obwohl der Geschäftserfolg immer stärker von der eingesetzten IT abhängt, kämpfen viele Unternehmen mit fehlerhafter Software, überschrittenen Kostenbudgets oder nicht eingehaltenen Projektlaufzeiten.

Mit Hilfe eines Software-Test-Managements lassen sich diese Probleme vermeiden. Fehler können zu jedem Zeitpunkt des Software-Lebenszyklus erkannt und behoben werden. Die folgenden Erfahrungswerte zeigen Ihnen, welche Fehler Sie vermeiden sollten:

### Kein Test-Management

Der erste Fehler, den Unternehmen in Sachen Test-Management machen, ist so einfach wie nahe liegend: Sie haben keines. Test-Management wird in den meisten Fällen weder methodisch geplant noch umgesetzt. Oft denkt sich der verantwortliche Projektleiter erst am Ende des Entwicklungsvorhabens eine Handvoll Testfälle aus und überprüft lediglich stichprobenartig und unsystematisch einige Funktionen der Software.



**Entwicklungsbegleitendes Testen anhand des V-Modells:  
Die Testaktivitäten der verschiedenen Teststufen integrieren diese mit den Entwicklungsstufen.**

Meist werden an diesem Punkt zu viele Fehler gefunden, um diese rechtzeitig zu korrigieren und das Projekt noch innerhalb der geplanten Laufzeit fertig zu stellen. Obwohl die Software längst eingeführt werden sollte, müssen nun erst noch Fehleranalysen und Fehlerkorrekturen betrieben werden. Dies erzeugt deutlich mehr Aufwand als ursprünglich geschätzt. Zudem bleibt meist keine Zeit mehr, die Korrekturen sowie die gesamte Software erneut zu testen. Das Ergebnis bleibt somit weiterhin risikobehaftet.

Manche IT-Abteilungen setzen zwar das Thema Test auf ihre Agenda, vergessen in ihrer Kalkulation aber die Korrek-

turen - ein weiterer Faktor, der versteckte Kosten und Zeitverluste innerhalb eines Projekts bedeuten kann. Es sollte deshalb nicht nur der Test eingeplant werden, sondern auch die Korrektur von Fehlern sowie ein entsprechender Regressionstest. Darunter versteht man die Wiederholung aller Testfälle oder einer Teilmenge, um mögliche Nebenwirkungen von Modifikationen in unveränderten Teilen der Software aufzuspüren.

### Komplexität und Vorbereitungs-dauer werden unterschätzt

Viele Unternehmen machen noch immer den typischen Fehler, zu spät mit dem Testen zu beginnen. Oft handeln sie dabei

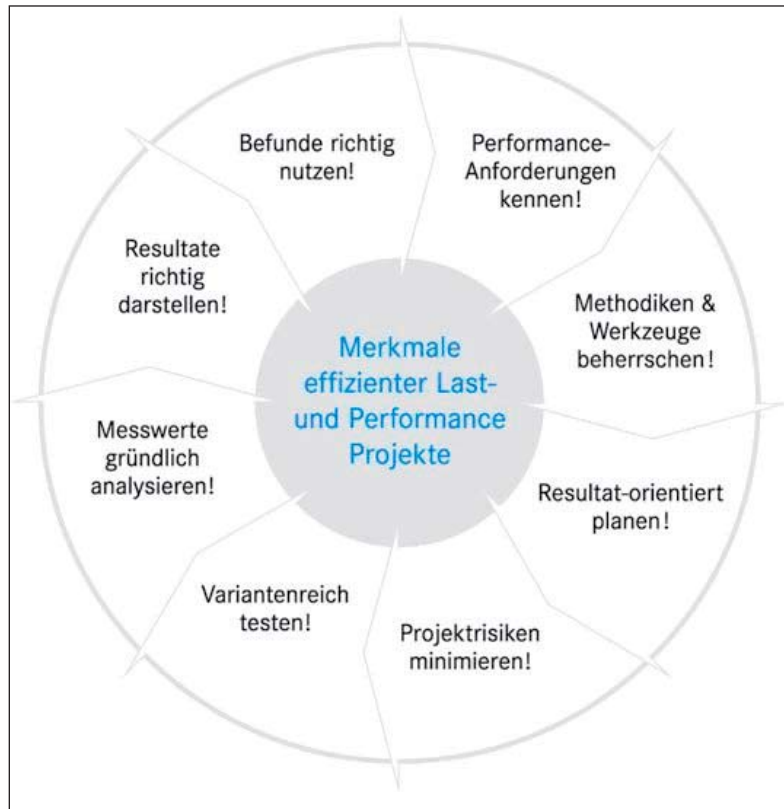
auch nach dem Irrglauben, dass erst ein großer Teil der Software erstellt sein muss, um diesen auch testen zu können. Dadurch werden allerdings Testaktivitäten versäumt, die bereits sehr früh möglich sind. Die Folge: Fehler werden erst zum Projektende entdeckt.

Mit einem systematischen Test-Management kann von Beginn der Entwicklung an und das ganze Projekt begleitend getestet werden. Das heißt, jede Entwicklungsstufe ist in ihrer entsprechenden Teststufe integriert. Dadurch lassen sich Fehler schnell in den Entwicklungsstufen finden, in denen sie entstanden sind. Auf diese Weise ist eine weniger aufwändige Korrektur möglich, und folgende Entwicklungsstufen bauen nicht auf unentdeckten Fehlern auf. Darüber hinaus lassen sich bereits während der Projektlaufzeit Qualitätsaussagen zum fertigen Produkt treffen.

**Fehlendes und unzureichendes Testdaten-Management**

Der wichtigste Schritt des Testdaten-Managements ist die komplette Erfassung der vollständigen Testdaten. Einige Unternehmen beschränken sich an dieser Stelle darauf, den Testfall zu beschreiben, und vergessen darüber hinaus, ein vollständiges Datenset zu definieren, welches für die spätere Erfassung notwendig ist. Kernpunkt für den Erfolg ist es, Testdaten dauerhaft zu verwalten. Dazu zählt genauso, neue Daten zu ergänzen, wie auch Daten durch neue Zusammenstellungen zu optimieren.

Kann man während des Testens nicht auf den entsprechenden Datenbestand oder nur auf alte Daten zurückgreifen, so führt das zu erneutem Zeitverzug und möglichen Fehlerquellen. Zudem ist es Aufgabe des Testdaten-Managements, zu wissen, welche Daten an welchen Tagen welche Ergebnisse auslösen können und welche Ergebnisse erwartet werden dürfen. Ein entsprechendes Risiko-Management ist damit gewährleistet, weshalb es unbedingt notwendig ist, über die gesamte Projektlaufzeit entsprechende Ressourcen einzuplanen.



**Merkmale effizienter Last- und Performance-Tests.**

**Fehlende Testautomatisierung**

Bei der Testautomatisierung unterscheidet man zwischen der codenahen Testautomatisierung und der GUI-Testautomatisierung. Unternehmen machen dabei die Fehler, entweder keine codenahe Testautomatisierung anzusetzen oder die GUI-Testautomatisierung zu früh zu beginnen. Verzichtet man auf eine codenahe Testautomatisierung, verschenken die Firmen eine preiswerte Möglichkeit, bereits zeitig Aufschlüsse zur Qualität des Systems zu erhalten.

Typischerweise kommt es in frühen Projektphasen noch zu Änderungen in den spezifizierten Masken, da Anforderungen künftiger Benutzer übersehen wurden oder sich nachträglich noch ändern. Wird eine GUI-Automatisierung zu früh angesetzt, muss sie alle Änderungen der Masken nachziehen. Dies treibt den Wartungsaufwand der GUI-Automatisierung in die Höhe.

Optimal wäre der Einsatz einer frühen codenahen Testautomatisierung sowie einer am Projektende eingesetzten GUI-Testautomatisierung. Auf diese Weise lässt sich ein Großteil der

bestehenden Funktionen mit einer daraufhin erstellten Automatisierung kostengünstig in Form von Regressionstests prüfen.

An dieser Stelle spielt auch das erwähnte Testdaten-Management eine wichtige Rolle. Eine Automatisierung, egal ob codenahe oder auf das GUI bezogen, verlangt einen immer wieder auf den Ursprungszustand einstellbaren Testdatenbestand. Nur dadurch liefert das zu testende System für die Automatisierung vergleichbare Ergebnisse.

**Das benötigte Know-how wird unterschätzt**

Wenn Unternehmen ihre Software testen, so begehen sie häufig den Fehler, die Entwickler der Software auch als Tester einzusetzen. Dies liegt zwar nahe, denn für die Firmen sind die Softwareentwickler zugleich die Experten der Software. Die Erfahrung hat jedoch gezeigt, dass den Entwicklern der objektive Blick auf die von ihnen entwickelte und zu testende Software fehlt. Sie neigen dazu, nur die Funktionen zu testen, für die sie das Produkt entwickelt haben. Negativfälle, in denen An-

wender falsche, nicht nachvollziehbare Eingaben machen, werden von ihnen in aller Regel nicht berücksichtigt. Im Übrigen verfügen die meisten Entwickler auch nicht über das spezifische Wissen eines Testers, da es in der Ausbildung meist nicht berücksichtigt wird.

Das Test-Management zeigt zwar Defizite in der Qualität der Software auf, viele Unternehmen tun sich aber schwer, den daraus resultierenden Nutzen zu quantifizieren. Oft müssen erst Fehler in der Produktion auftreten, damit entsprechende Budgets eingerichtet werden. Dementsprechend sollten die Verantwortlichen die Komplexität und das notwendige Know-how frühzeitig identifizieren und handeln, bevor Fehler auftreten.

Entwicklung und Testen von Software sind hochkomplexe Bereiche, die - wenn sie Hand in Hand arbeiten - maßgeblich zur Gesamtqualität beitragen. Eine unabhängige Testabteilung ergänzt die Softwareentwicklung, niemals kann jedoch eine Abteilung die andere ersetzen.

#### **Verzicht auf methodisches Vorgehen**

In der Praxis testen Unternehmen häufig ziel- und planlos. Viele manuelle Schritte, Redundanzen, unterschiedliche Dokumentationen und daraus resultierende Mehrkosten sind an der Tagesordnung. Intuitiv versuchen Tester, die richtigen Testfälle zu ermitteln. Dabei werden Testfälle übergangen, redundant erstellt, oder man findet nicht die richtigen beziehungsweise wirklich wichtigen. Das Ergebnis: Ein effektives Testen wird unmöglich.

Mit einem methodischen Vorgehen lassen sich dagegen die richtigen Testfälle in der richtigen Zahl finden. Die Unternehmen können damit auch den Aufwand für das Testen und den Support maßgeblich senken. Das bedeutet höhere Qualität, geringere Kosten, verbesserte Dokumentation der Systemanforderungen, eine geringere Ausfallrate und schließlich zufriedener Kunden.

Experten haben sich unter

dem Dach des Internationalen Software Testing Qualification Board (ISTQB) zusammengeschlossen, um dieses methodische Vorgehen in die Ausbildung für professionelle Softwaretester zu integrieren. Gute Tester sind heute schwer zu finden, mit einem ISTQB-Zertifikat hat man eine Basis geschaffen, die Qualität der Tester zu belegen.

#### **Kein Tool-Einsatz oder die falschen Tools**

Oft betrachten Unternehmen den frühen Einsatz von Werkzeugen als Allheilmittel für all ihre Testprobleme. Dies ist jedoch ein Irrglaube. An erster Stelle jeder Optimierung des Test-Managements steht die Ausarbeitung der unternehmerischen Prozesse und Methoden. Erst wenn diese definiert sind, können Werkzeuge ausgewählt werden, die optimal zu diesen Methoden und Prozessen passen. Für kleine Unternehmen können sich hier mehrere Einzellösungen genauso sinnvoll ergänzen, wie sich möglicherweise für große Unternehmen integrierte Tool-Suites als richtig erweisen.

Auch ein zu später Werkzeugeinsatz kann die Effektivität und Effizienz der Testmethoden und -prozesse einschränken. Häufig versuchen die Unternehmen allzu lange, bestehende Prozesse in bestehende generische Tool-Einsätze, wie zum Beispiel Excel oder Word von Microsoft, hineinzuprogrammieren. Fazit: Nur die richtigen Werkzeuge zum richtigen Zeitpunkt führen zur optimalen Effektivitäts- und Effizienzsteigerung.

#### **Falsche Testfälle oder falsche Anzahl von Testfällen**

Unerfahrene Tester und eine fehlende systematische Vorgehensweise führen zu redundanten und fehlenden Testfällen. Zu viele Testfälle können einerseits Redundanzen bedeuten. Die Firmen verschenken damit Zeit und finanzielle Mittel. Zu wenige oder falsche Testfälle führen andererseits zu einer unzureichenden Testabdeckung. Es bleiben ungetestete Funktionen zurück, mit dem Ri-



**Eric Jochum, Bereichsleiter für Test-Management bei beck et al.:** „Software muss in hochkomplexen und unternehmens-kritischen Bereichen funktionieren. Ein Testen in jeder Entwicklungsstufe ist da nicht mehr wegzudenken.“

siko, dass mögliche Fehler in diesen Funktionen in die Produktion gelangen.

Ein systematisch-methodisches Vorgehen stellt sicher, dass die richtigen Testfälle ermittelt werden. Zu den methodischen Vorgehensweisen werden analytische, heuristische wie auch systematische Testfall-Ermittlungsmethoden gezählt.

#### **Last- und Performance-Test erst am Projektende**

Last- und Performance-Tests sind ein wichtiges Instrument, um das System mit Blick auf das zu erwartende Benutzerverhalten zu prüfen. Ziel ist es, die Skalierbarkeit verschiedener Funktionen der Applikation zu ermitteln. Gerade die Lasttests sind aufwändig zu planen und zu betreiben. Unternehmen neigen deshalb oft dazu, diese Tests erst am Ende eines Projektes anzusetzen. Jedoch lässt sich bereits während der Entwicklung testen, welcher Last das fertige Softwareprodukt einmal wird standhalten können.

Probleme sind fast immer in der Softwarearchitektur begründet, die ganz zu Beginn des Projektes entworfen wird. Testet man nun erst am Ende eines Entwicklungsvorhabens und findet solche Fehler verspätet,

bedarf es eines großen Zeit- und Kostenaufwands, da man wieder am Anfang einhaken muss. Das kann einen zusätzlichen Zeitverlust sowie Mehrkosten bedeuten, welche die Verantwortlichen in aller Regel nicht einkalkuliert haben. Last- und Performance-Tests müssen deshalb immer von Beginn an konzipiert und entwicklungsbegleitend eingeplant werden.

#### **Die Folgen eines falschen Test-Managements werden unterschätzt**

Was bedeutet eine fehlerhaft laufende Software für ein Unternehmen? Was bringt eine verspätete Einführung an versteckten Kosten mit sich? Wie wirken sich Lücken in der Software auf die Produktivität der Mitarbeiter

und schließlich das Erreichen der Geschäftsziele aus? All das sind Fragen, denen sich die Verantwortlichen in den Unternehmen stellen müssen. Software darf heute nicht mehr nur vom Blickwinkel der IT aus betrachtet werden, denn sie hängt direkt mit den meisten Geschäftsprozessen des Unternehmens zusammen. Damit wird sie zu einem der kritischen Faktoren für Erfolg oder Misserfolg.

Unterschätzen die Verantwortlichen das und dulden mögliche Fehler, Verzögerungen oder Ausfälle, die durch Test-Management vermeidbar gewesen wären, schaden sie damit der Bilanz und dem eigenen Ansehen. Letztlich kann sogar der Verlust von Kunden drohen. Deshalb sollten Unternehmen

immer eine Test-Management-Organisation aufbauen, die eigenständig und unabhängig von der Entwicklung arbeitet und damit vor den genannten Problemen schützt.

#### **Fazit**

Um den hohen Anforderungen gerecht zu werden, welche heute an die Software in hochkomplexen und unternehmenskritischen Bereichen gestellt werden, ist ein Testen in jeder Entwicklungsstufe nicht mehr wegzudenken. Mit einem strukturierten Test-Management sowie qualifizierten Testern und Test-Managern sind Unternehmen diesen Anforderungen schon heute gewachsen.◆