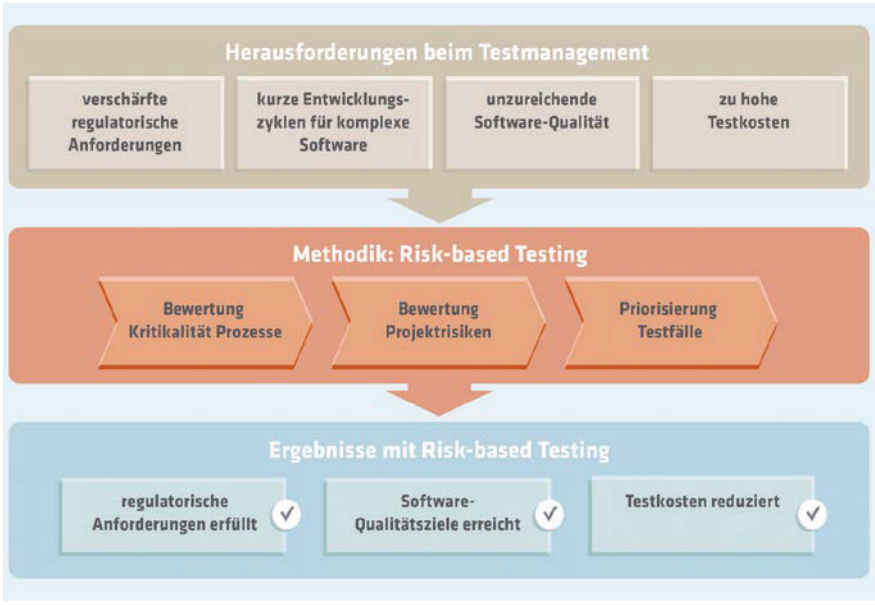
Praxis-Tipps aus Sicht des Testmanagements

Pragmatischer Umgang mit den verschärften regulatorischen Anforderungen an die Banken-IT. Seit Veröffentlichung des Rundschreibens „10/2012 (BA) – Mindestanforderungen an das Risikomanagement MaRisk" sowie den Erläuterungen in 12/2012 durch die Bundesanstalt für Finanzdienstleistungsaufsicht (BaFin) gibt es in der IT von Banken eine Vielzahl von umzusetzenden Maßnahmen, um insbesondere die Anforderungen aus AT 7.2 und AT 7.3 zu erfüllen.

Autor:
Dieter Koenen,
Manager Consulting Services bei der innobis AG

Angekündigte Sonderprüfungen nach § 44 KWG Absatz 1 (Kreditwesengesetz) fordern gleichermaßen Business- und IT-Verantwortliche heraus. Zahlreiche Formalismen und administrative Hürden sind zusätzlich zur hohen Belastung im Tagesgeschäft zu bewältigen.

Natürlich sind die regulatorischen Vorgaben zu erfüllen, aber nicht selten schießen die Maßnahmen auch übers Ziel hinaus. Wichtig ist es, einen angemessenen Umsetzungspfad zu finden. Das Testmanagement bildet dabei die zentrale Klammer zwischen Fachbereich, Entwicklung und Betrieb. Im Test vorgeschriebene Maßnahmen führen zu erheblichen Seiteneffekten in den oben genannten Bereichen. Nachstehend werden pragmatische und praxiserprobte Ansätze entlang des Software-Entwicklungsprozesses skizziert, mit denen sich die regulatorischen Anforderungen aus Sicht des Testmanagements erfüllen lassen. Anhand der Checkliste rechts kann verifiziert werden, welche Komponenten noch zu implementieren sind.



Die Methodik Risk-based Testing beim Testmanagement *Quelle: innobis*

Konzeption

Neue oder modifizierte Software muss getestet werden. Die fachliche Ausprägung der Testfälle unmittelbar vor der Testphase durch die Fachbereiche gestaltet sich oft als zäher Prozess. Viel besser ist es, in der Konzeptionsphase für jedes formulierte Requirement auch eine Beschreibung des Testszenarios einzufordern. Das Thema Rollen und Berechtigungen wird üblicherweise eher stiefmütterlich behandelt. Auch hier sollte die IT schon in der Konzeptionsphase auf die korrekte und vollständige Definition, auf Funktionstrennung sowie auf die Formulierung entsprechender Testszenarien drängen.

Entwicklung

Aktuelle und möglichst schlank gehaltene Entwicklungsrichtlinien mit Namenskonventionen, Richtlinien für den Umgang mit kritischen Programmiertechniken sowie Hinweisen für optimierten Programm-Code sind inzwischen ein Muss. Weiterhin empfiehlt sich ein integriertes Auftrags- und Transportverwaltungssystem. Für SAP-Anwender bietet sich hier die Komponente SAP Change Request Management

(ChaRM) des Solution Managers mit dedizierten Genehmigungsverfahren, strikten Rollentrennungen sowie dem revisionssicheren Nachweis der Transportkette von Entwicklungs- über Test- und Abnahmesysteme bis hin zur Produktion an. Nicht viele Entwicklungsbereiche nehmen sich die Zeit für Code-Reviews im Vier-Augen-Prinzip. Auch hier gibt es inzwischen gute Werkzeuge auf dem Markt, welche den Programm-Code u.a. auf sicherheitsrelevante Schwachstellen überprüfen und bei Regelverletzungen den Transport in Qualitätssicherungs- oder Produktivsysteme verhindern. Im SAP-Umfeld gelten der Code Profiler von Virtual Forge sowie der SAP Code Vulnerability Scanner als führend. Eingangsvoraussetzung für den Fachtest sind dokumentierte Modultests durch die Entwicklung sowie die Benennung von Testeinschränkungen wie noch nicht fertiggestellte Funktionen oder bereits bekannte Fehler. Auch hier bieten sich unterstützende Werkzeuge an. Die ChaRM-Komponente des SAP Solution Managers ermöglicht beispielsweise die Dokumentation und den Nachweis von Modultests.

Testmanagement

Als allererstes empfiehlt es sich, die Testmanagement-Prozesse und die Rollen in den Test- und Abnahmeprozessen eindeutig zu definieren sowie Templates z.B. für Testkonzepte oder Testfallbeschreibungen zur Verfügung zu stellen. Auch hier gilt der Grundsatz: „Weniger ist mehr“. Lange Prosa-Passagen werden nicht gelesen. Besser sind Checklisten, Tabellen und Grafiken. Übrigens ist, um die regulatorischen Vorgaben zu erfüllen, die strikte Rollentrennung zwischen Fachbereich, Entwicklung und Test wichtig. Excel-basierte Testmanagement-Verfahren haben sich überholt. Tests und Abweichungen sind nachvollziehbar und revisionssicher zu dokumentieren und aufzubewahren. Der Einsatz von integrierten Testmanagement-Tools wie das HP Application Lifecycle Management (HP ALM) Quality Center, IBM Rational Quality Manager oder die SAP Solution Manager Test Workbench ist obligatorisch. Aus Sicherheitsgründen dürfen in nicht zentral gemanagten Testumgebungen keine sensiblen (insbesondere auch personenbezogene) Daten verwendet werden. So sind beispielsweise Geschäftspartnerdaten zu anonymisieren. Für den Zugriff auf Testdaten durch Tester (und ggf. durch Entwickler für Fehleranalysen) ist ein Prozess zu definieren. Die Vergabe von User-IDs und Berechtigungen ist zu protokollieren. Stark an Bedeutung gewinnt das Risiko-Management. So muss die Bank auch nachweisen, dass Testrisiken wie die zu späte Bereitstellung der zu testenden Software oder eine fehlende Testinfrastruktur aufgenommen und bewertet sowie entsprechende Mitigationsmaßnahmen durchgeführt und verfolgt werden.

Bewährt hat sich der risikobasierte Testansatz. Prozesse werden in diesem Verfahren nach Kritikalität wie Aufrufhäufigkeit oder Sicherheitsanforderungen sowie hinsichtlich Änderungen und Anpassungen im laufenden Testvorhaben bewertet (s. Schaubild „Die Methodik Risk-based Testing beim Testmanagement“). Daraus leiten sich der Umfang und die Priorität der involvierten Testobjekte beziehungsweise Testfälle ab. Der Fokus verschiebt sich damit risikogetrieben auf die wirklich wichtigen Tests. In der Verantwortung des Testmanagements liegt der Nachweis, dass Pflichtergebnistypen wie ein Testkonzept oder eine Testplanung erstellt, obligatorische Tests wie Security- oder Disaster-Recovery-Tests durchgeführt (bzw. die Nichtausführung begründet) oder der Abnahmeprozess ordnungsgemäß durchlaufen wurden. Hier bietet sich ein Checklisten-gestütztes Internes Kontrollsystem (IKS) an. Bei der Implementierung zählen definierte Übergabeverfahren bei strikter Rollentrennung zwischen Entwicklung und Betrieb. Für den Betrieb ist ein möglichst toolgestütztes Information-Security-Managementsystem zu etablieren und zu betreiben. Aus Sicht des Tests kann dann beispielsweise auf regelmäßige Standverfahren für ein Disaster Recovery verwiesen werden.

Fazit

Die oben aufgeführten Maßnahmen zur Erfüllung der regulatorischen Anforderungen sind zwingend umzusetzen. Empfohlen werden eine ganzheitliche Betrachtung sowie die Implementierung von pragmatischen und praxisbewährten Ansätzen. Anhand der Checkliste (siehe unten) kann verifiziert werden, welche Bausteine noch einzuführen sind. Weiterhin sollte auf Standards wie die BSI-Standards 100-1 bis 100-4, ISO 29119 (Test) oder ITIL gesetzt werden.

Nr.	Komponente	Nicht erfüllt	Teil-weise erfüllt	Voll-ständig erfüllt
1.	Ist die Definition von Testszenarien für Requirements obligatorisch?			
2.	Werden Änderungen an Rollen und Berechtigungen beschrieben?			
3.	Gibt es (aktuelle) Entwicklungsrichtlinien?			
4.	Ist ein integriertes Auftrags- und Transportsystem implementiert?			
5.	Sind die Rollen beim Roll-out von Software strikt zwischen Entwicklung und Betrieb getrennt			
6.	Werden Code-Reviews durchgeführt?			
7.	Gibt es unterstützende Qualitätssicherungswerkzeuge in der Entwicklung?			
8.	Werden Modultests durchgeführt und dokumentiert?			
9.	Sind die Testprozesse und die Rollen im Testmanagement beschrieben?			
10.	Gibt es im Test eine strikte Rollentrennung zwischen Fachbereich, Entwicklung und Test?			
11.	Steht ein integriertes Testmanagement-Werkzeug zur Verfügung?			
12.	Werden sensible Daten in Testsystemen geschützt bzw. anonymisiert?			
13.	Gibt es ein Verfahren zur Vergabe von User-IDs und Berechtigungen für Testsysteme?			
14.	Werden die Testrisiken dokumentiert, bewertet und verfolgt?			
15.	Wird die Durchführung von Testfällen priorisiert (Risk-based Testing)?			
16.	Wird die Erstellung von Pflicht-Ergebnistypen sowie von obligatorischen Tests systematisch kontrolliert (IKS-Checkliste)?			
17.	Gibt es einen definierten Prozess für den Zugriff auf sensible Testdaten?			
18.	Ist der Übergabeprozess in die Produktion definiert?			
19.	Ist ein Information-Security-Management-System (inkl. Notfall-Konzepte etc.) etabliert?			
20.	Werden Standards wie ITIL, BSI oder DIN ISO Normen eingesetzt?			

Diese Checkliste (kein Anspruch auf Vollständigkeit), kann verwendet werden, um einen ersten Überblick zu gewinnen, ob und welche Komponenten implementiert sind