

Der Software Testprozess für Manager

„Arbeitstitel“

EARLY DRAFT

Version 0.6

Date: 22. Oktober 2001

Author:

Andreas von Grebmer

Inhaltsverzeichnis

1. Einleitung	5
Teil 1 6	
1.1 Feedback	6
Teil 1	7
2. Der Testprozess (Level 1)	9
2.1 Prozessübergreifende Aktivitäten	12
2.1.1 Defect Tracking.....	12
2.1.2 Change Management.....	15
2.1.3 Namenskonvention.....	15
2.1.4 Abgrenzung.....	16
2.1.5 Test Tools / Werkzeuge.....	18
2.2 Unit Test (Level 2).....	20
2.3 Unit Integration Test (Level 2).....	22
2.4 System Test (Level 2)	23
2.5 Load & Performance Test (Level 2)	23
2.6 User Acceptance Test (Level 2).....	24
2.7 Pilot Test (Level 2)	24
3. Rollen im Testprozess	25
3.1 Schulung.....	28
4. Testdaten	30
4.1 Testdaten für Load & Performance Tests	31
5. Dokumente im Testprozess	33
6. Prozess Kennzahlen (KPI)	35
7. Argumente für das Testen	36
7.1 Fragen, die unbedingt gestellt werden sollten.....	36
7.2 Katastrophen Beispiele	36
7.2.1 CS-Bancomaten ziehen EC-Karten ein (Schweiz)	36
7.2.2 Credit Suisse: Ganztägiger Systemabsturz Basel/Zürich	36
7.2.3 Coop: Online Shops – Supermarkt.....	37
Liebe Kundinnen und Kunden Liebe Besucherinnen und Besucher	37
7.2.4 Der CS-Internetbroker Youtrade ist vorübergehend blockiert gewesen	37
7.2.5 Swisscom Kunden ohne Internet-Zugang.....	38
7.2.6 Softwarepanne bei SBB Billettautomaten.....	38
7.2.7 Bundestag hatte ein Jahr-2000 Problem	38
8. Risiken	40
9. Anforderungen (Anforderungsspezifikation)	41

10. Kosten im Testprozess	42
A. Appendix A	43
A. Abkürzungen	43
B. Referenzen (~ oder Literaturliste)	43
Management und Optimierung des Testprozesses	43
Methodisches Testen von Programmen	43
11. Links	44
11.1 Projektmanagement	44
11.2 Testen von Software	44
11.3 Tools	44
11.4 Workflow	45
11.5 Standards / Standard Organisationen	45
12. Organisationen	46
13. Dokumentenbeispiele	47
14. Stichwortregister (Teil 1 & 2)	48
Teil 2	49
15. TestLex - Test- und Qualitätsmanagement Lexikon	50
15.1 Handhabung	50
15.2 Lexikon	50
Ende	67

TABELLENVERZEICHNIS

TABELLE 1: TESTPROZESS (LEVEL 1)	10
TABELLE 3: FEHLER PRIORITÄTEN	13
TABELLE 4: FEHLER HÄRTE	13
TABELLE 5: DEFECT TRACKING MUSTER PROZESS	14
TABELLE 6: NAMENSKONVENTIONSMODELL	16
TABELLE 7: NAMENSKONVENTIONSBEISPIEL	16
TABELLE 8: WHITE BOX TESTING	20
TABELLE 9: UNIT TEST (LEVEL 2)	21
TABELLE 10: BLACK BOX TESTING	22
TABELLE 11: ROLLEN IM TESTPROZESS	25
TABELLE 12: LOAD & PERFORMANCE BEISPIEL MATRIX	31
TABELLE 13: DOKUMENTE IM TESTPROZESS	33
TABELLE 14: PROZESS KENNZAHLEN (KPI'S)	35
TABELLE 15: RISIKOBEWERTUNG	40

TABELLE 16: ABKÜRZUNGEN 43

TABELLE 17: BÜCHER ZUM THEMA 43

DRAFT

1. Einleitung

Schon wieder ein Buch zum Thema Software Testen ? Ja, weil ich nach allen Büchern, die ich zum Thema Software Testing und Quality Assurance gelesen habe, zum Schluss gekommen bin, dass der Grossteil der Beteiligten weiss, wie man Software testet, aber das vorhandene Wissen im täglichen Betrieb nicht umgesetzt wird. Daher wird dieses Buch Bekanntes nur oberflächlich aufnehmen und sich vielmehr um die Psychologie des Testen kümmern und versuchen, einen pragmatischen Ansatz für die Qualitätssicherung im IT-Umfeld zu vermitteln.

Zielgruppe & Nutzen

- Gesamtprojektleiter zur Gewinnung des Überblicks „QA und Testing“
- Testmanager für die Vertiefung und zum Erhalt neuer Impulse
- Tester für den Gesamtüberblick und Vertiefung einzelner Bereiche des Testens und
- alle als Nachschlagewerk im tägliche Gebrauch
- als Diskussionsgrundlage

Oft streiten sich Projektbeteiligte mit Linienstellen oder Projektmitarbeiter untereinander über Begriffe, wie z.B. Load und Stress Test wochenlang. Würden sie sich an einen Tisch setzen und Ihre Begriffe synchronisieren, so wären alle Unklarheiten innerhalb einer Stunde bereinigt. Es würden Motivationsverlust vermieden, Zeit gespart und Ressourcen geschont.

Kernaussage :

1) Vereinbaren Sie einen Prozess mit allen Beteiligten und halten Sie (trotz aufkommender Hektik) daran fest !

Das Umgehen des Testprozesses (kleiner Freundschaftsdienst !) kann später fatale Folgen haben.

2) Ein Testprozess muss auch ohne Tool „lebensfähig“ sein.

Die Sprache des Buches ist Deutsch. Ich verzichte ausdrücklich darauf, gewisse Ausdrücke auszudeutschen, auch wenn das für Sie inkonsequent wirkt. Das Buch setzt beim Leser ein gewisses IT-Vokabular voraus. Alle nicht sofort erklärten Ausdrücke und Begriffe sind im Teil 2 Lexikon erklärt.

Aus Gründen der einfacheren Lesbarkeit wird immer nur die männliche oder die weibliche Form gewählt.

i Tips; Listet am Ende eines Kapitels nochmals in Kurzform die wesentlichen Punkte auf.

Dieses Buch ist wie folgt aufgeteilt :

Teil 1

Der Testprozess aus Sicht des Autors

Teil 2

TestLex, ein Lexikon zum Thema, mit Erklärungen

1.1 Feedback

Bitte senden Sie Ihre Ergänzungsvorschläge, Feedback, konstruktive Kritik und Lob an

testprocess@guideline.ch

Referenzieren Sie bitte auf Ausgabe, Titel/Untertitel und Seite. Soweit es möglich ist, werden Vorschläge eingearbeitet (Für die Beantwortung aller Anfragen kann ich aber nicht garantieren).

Teil 1

DRAFT

2. Der Testprozess (Level 1)

Definition : Unter „Testprozess“ verstehe ich die Abfolge aller Aktivitäten in einem IT-Projekt, die dazu dienen, die Qualität, d.h. die Erfüllung der Anforderungen zu beweisen und zu verbessern. Dieser Level 1 Prozess zerfällt in Unterprozesse Level 2, diese wiederum in „Arbeitsanweisung (Level 3)“. Unter Umständen kann eine noch feinere Gliederung Sinn machen.

Mit dem Testprozess Level 1 können Sie sich praktisch an jedes Projekt wagen, ob e-commerce Plattform oder Toaster Produktion. Durch Tailoring bauen Sie sich den Prozess, mit dem Ihr Projekt leben kann.

i Die Darstellung des Prozesses findet hier phasenähnlich statt, d.h. in Teststufen. Der Software Life Cycle ist aber in der Regel interaktiv. Stellen Sie sich den Prozess einfach in der Anwendung so vor :

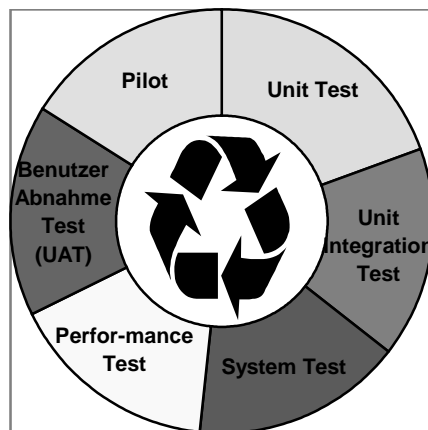


Tabelle 1: Testprozess (Level 1)

Testprozess Level 1

Phasen-und Stufen übergreifende Tätigkeiten	Projektmanagement					
	Quality Management					
	Risk Management					
	Defect & Change Management					
Teststufen →	Sub-Prozess: <i>Unit Test</i>	Sub-Prozess: <i>Unit Integration</i>	Sub-Prozess: <i>System Test</i>	Sub-Prozess: <i>Performance</i>	Sub-Prozess: <i>UAT</i>	Sub-Prozess: <i>Pilot</i>
Verantwortliche Rolle	Entwicklung	Entwicklung Test Team	Test Team Betreiber	Test Team	Auftraggeber Betreiber	Auftraggeber Benutzervertreter
Phase-Input	ungetestete Unit	einzel getestete Units	Unit getesteter Release	System getesteter Release	Performance getesteter Release	Release mit Benutzer Freigabe
Phase Aktivitäten Testarten ↓	Code Review Schnittstellen Test Schnittstellen Performance	Konfigurationstest	Installation De-installation (Fallback) Konfigurations Test Regressions Test Security Test Dokumentation Backup Test Recovery Test Fail Over Test Funktionale Tests End of Period Test Link Test	Load Test Stress Test Volume Test Break Test Performance Profiling	Abnahme durch Benutzervertreter	Tagesgeschäft in begrenztem System mit eingeschränktem Benutzerkreis Basis: Use Case Beschreibungen
Phasen-Output	getestete Unit	Unit getesteter Release	System getesteter Release	Performance getesteter Release	Release mit Benutzer Freigabe	Release mit Freigabe für Produktion

Diese Grafik vermittelt einen Überblick über die Aktivitäten, Abhängigkeiten und deren Zeitdimension im Testprozess.

DRAFT

2.1 Prozessübergreifende Aktivitäten

Folgende Tätigkeiten sind Testphasen übergreifend und müssen mit alle Phasenverantwortlichen abgestimmt werden. Wie bei der Festlegung der Rollen sind diese Definitionen bei einem zweiten Projekt, mit den gleichen Teammitgliedern, kein Thema mehr. Für die Akzeptanz ist es wichtig, dass wo nötig und gewünscht Adaptionen vorgenommen werden.

2.1.1 Defect Tracking

Mit dem Defect Tracking Prozess legen Sie fest, wie auftretende Fehler behandelt und bis zu einer Lösung weiterverfolgt werden. Durch die Priorisierung legen Sie die Geschwindigkeit der Behebung fest. Severity (Fehlerhärte) zeigt die Tragweite, d.h. die Auswirkung des Fehler oder Fehlverhaltens auf. Aus psychologischen oder Motivations-Gründen, empfiehlt es sich, wenn Sie dem Prozess des Defect oder Error Tracking einen anderen Namen, wie z.B. „Issue Tracking“ geben. Sie brechen so den „passiven“ Widerstand der Person, die sich dem Problem annimmt, denn in den meisten Fehler, ist es die gleiche Person, die diesen Programmteil auch entworfen hat. Das Problem wird als Herausforderung gesehen.

Wichtig ist eine konsequente Verfolgung der Fehler und eine einheitliche Terminologie. Alle Fehler der Prioritäten 1-3 werden verfolgt. Sie können durchaus auch sprechende Prioritäten wie LOW, MEDIUM, HIGH vergeben.

Kleinere Projekte können sicher mit dieser Priorisierung leben. Für grössere Projekte, oder bei Bedarf kann auch eine feinere Aufteilung mit Severity (=Stärke/Heftigkeit) eingeführt werden.

Um eine hohe Akzeptanz zu erreichen, muss das Werkzeug von allen Projektmitarbeitern benutzbar sein. Hier treten häufig Probleme auf, wenn mehrere Firmen an einem Projekt arbeiten. Führende Hersteller in diesem Bereich, bieten aber schon Ihre Werkzeuge mit Webzugriff an.

Stellen Sie Ihre Erfahrungen, d.h. gefundene Fehler, Workarounds etc. der Supportorganisation des späteren Betreibers zur Verfügung. Im Idealfall verfügt die Supportorganisation über das gleiche Tool und hat Zugriff auf das gleiche Fehler-Repository (Fehler-Datenbank) wie die Testorganisation.

Tabelle 2: Fehler Prioritäten

Priorität	Beschreibung
1 Immediately	Funktion beeinträchtigt Lauffähigkeit/Sicherheit des Systems, Systemabsturz. Test wird abgebrochen und der Release zurückgewiesen bzw. Test wird nach Fehlerbehebung wieder aufgenommen.
2 Same Release	Fehler innerhalb Funktion/Modul. Funktion oder Modul wird nicht freigegeben bzw. Kunde/Betreiber muss dieses Fehlverhalten explizit „akzeptieren“.
3 Next Release	Normaler Fehler, Funktion oder Teilprogramm kann trotzdem benutzt werden. Workaround für dieses Fehlverhalten ist vorhanden.
4 info or Remark only	Unübliches Verhalten, Verbesserungsvorschlag, Kosmetische oder Schreibfehler

Tabelle 3: Fehler Härte

Severity	Beschreibung
High or System harmed	Alle Benutzer sind von diesem Fehlverhalten betroffen/können diese Funktion nicht löschen. System ist beeinträchtigt.
Midium or Inoperable Application	Applikation oder Funktion kann nicht ausgeführt werden. Systemstabilität ist nicht beeinträchtigt. Es gibt keinen Workaround für den Fehler.
Low or Formal Error	Applikation oder Funktion kann mit gewissen Einschränkungen ausgeführt werden oder es gibt einen Workaround für den Fehler. Systemstabilität ist nicht beeinträchtigt.
Infor- mational	Kosmetischer Hinweis, Verbesserungsvorschlag, Anpassung an Benutzererwartungen, Anpassungen an Best Practice.

Durch Verwendung von Laufnummern und bei konsequentem Defect Tracking sind Sie jederzeit in der Lage herauszufinden, in welchem Release der Fehler gefunden wurde (von wem etc.) und mit welchem Release er behoben wurde. Schleicht sich der gleiche Fehler nochmals ein, so finden Sie den Lösungsansatz viel schneller. Durch die

Einteilung in Fehlerklassen können Sie auch die kritischen Stellen in Ihrer Lösung identifizieren und die Testgewichtung entsprechend erhöhen.

i Tips

- Testprozess = Keep it simple !
- Kommunizieren Sie den Prozess
- Ziehen Sie die Beteiligten beim Prozessdesign mit ein
- Stellen Sie bekannte Fehler und Workarounds der Supportorganisation zur Verfügung (Fehler-Repository)

Der IEEE Standard 1044 beschreibt die Vergabe von Severity-Werten wie folgt:

Grafik

2.1.1.1 Defect Tracking Muster Prozess

Der Defect Tracking (Fehler Verfolgung) Prozess läuft üblicherweise in fünf Schritten ab. Dieser Musterprozess entspricht dem, was Defect Tracking Werkzeuge nach ihrer Grundinstallation abbilden. Eine Anpassung der beteiligten Rollen und Stati ist möglich.

Tabelle 4: Defect Tracking Muster Prozess

GRAFIK

Nicht aufgeführt in dieser Darstellung ist zum Beispiel die Umwandlung eines gefundenen Fehlers in einen Change Request. Oft werden Change Request mit einem eigenen Werkzeug verwaltet.

2.1.2 Change Management

Es gibt keinen funktionierenden Testprozess ohne dass ein Change- oder Release Management implementiert wird. Werden Sie als Testmanager zu einem Projekt geholt, so sollte die Frage nach dem Change Management zu einer der ersten gehören. Der Change Manager ist Ihr Zulieferer. Ist diese Rolle in einem Projekt nicht vorgesehen und Sie übernehmen die Rolle des Testmanagers, so sind Sie gleichzeitig auch Change- und Release-Manager. Die ist auch ein 100 % Job in den meisten Fällen. Stellen Sie sich also auf wenig Freizeit ein bei dieser Konstellation, bzw. bestehen Sie auf der Rolle und ihrer Besetzung.

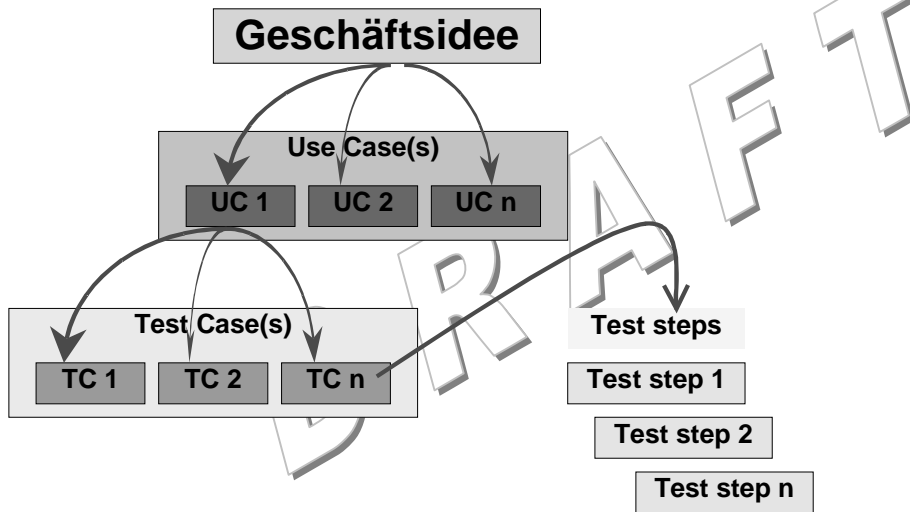
2.1.3 Namenskonvention

Innerhalb des Projektes muss eine einheitliche Namenskonvention für Dokumente, Klassen, Meilensteine, Objekte etc. bestehen. Diese Namenskonvention ist eine Tätigkeit „der ersten Stunde“. Nachträglich eingeführte Namenskonventionen sind oft schwer zu realisieren.

Es ist schwer, ein allgemein gültiges Modell zur Einführung der Namenskonvention zu erarbeiten, da die Konventionen so vielfältig sind wie die Problemlösungsansätze. Am besten erarbeiten Sie Ihren Ansatz in einem Workshop mit Teilnehmern aus allen Teilprojekten. Der Standard für die Namensvergabe muss im Projekthandbuch aufgeführt werden.

Hier ein Beispiel eines Namenskonventionsmodell:

Tabelle 5: Namenskonventionsmodell



Die Umsetzung würde dann etwa so aussehen :

Tabelle 6: Namenskonventionsbeispiel

Was	Beschreibung
UC001_Kleinkreditrechner	Namensgebung für den Use Case/Applikation Kleinkreditrechner
UC001_Grenzwertoben oder UC001_001_Grenzwertoben	Namensgebung für den Testcase „oberer Grenzwert“ des Kleinkreditrechner
UC001_GrenzwertOben_UT	Namensgebung für den Testcase „oberer Grenzwert“ des Kleinkreditrechner, zugeordnet zur Teststufe UNITTEST

Sie sehen, die Namensgebung kann beliebig fein sein.

i Tips

- Die Namensgebung muss rückwärtskompatibel sein. D.h. man sieht dem Test Case an, zu welchem Use Case er gehört.
- Namensgebungen müssen skalierbar, d.h. zukunftsorientiert sein

2.1.4 Abgrenzung

Das Abgrenzen (Was liefern Sie ?, Was liefern Sie nicht ?) des Testprozesses bzw. Ihres Prozessteils ist sehr wichtig. Erstens werden Missverständnisse vermieden und

zweites entstehen keine falschen Erwartungen bei anderen Beteiligten über das, was Sie liefern werden.

Ausgeklammerte Testkomponenten sollten Sie nicht einfach im luftleeren Raum stehen lassen. Wenn Sie sie nicht testen, dann muss es jemand anderes tun. Kümmern Sie sich darum, dass der Komponententest adressiert wird.

Mögliche Komponenten sind:

- Hardware
- Software
- Betriebssystem (OS)
- Dokumentation
- Netzwerk

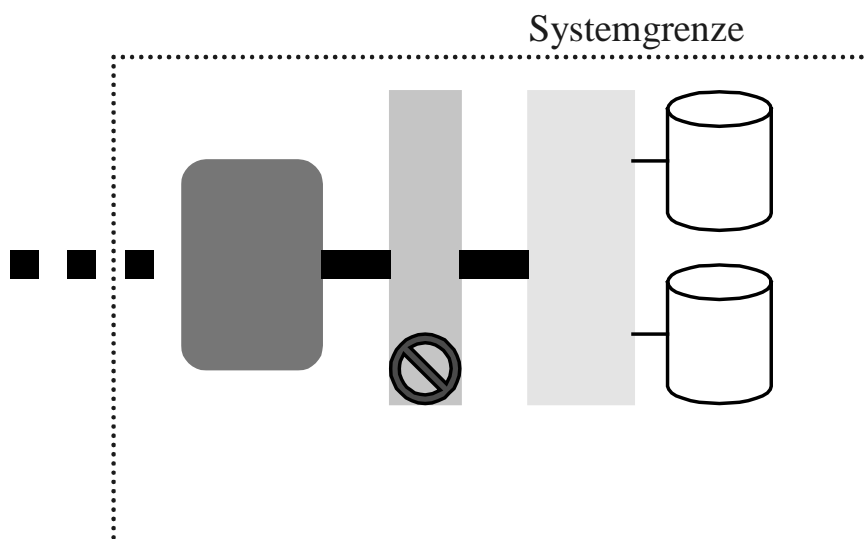
i Tips

- Kommunizieren Sie klar, was von Ihnen zu erwarten ist !
- Kommunizieren Sie klar, was nicht von Ihnen zu erwarten ist !
- Nehmen Sie Abgrenzung in die Teststrategie oder Testplan auf !
-

2.1.4.1 Beispiel Abgrenzung des Testprozesses

Getestet werden nur alle Komponenten innerhalb der Systemgrenze. Auch innerhalb der Systemgrenze können einzelne Komponenten ausgeklammert werden (⊘).

Definition : Die Systemgrenze.....



2.1.5 Test Tools / Werkzeuge

Wie in allen anderen Bereichen gibt es Werkzeuge, die den Testprozess automatisieren. Sie können ihn aber nicht ersetzen. Der Prozess muss zuerst bestehen, bevor Sie über die Evaluierung oder die Anschaffung der Werkzeuge nachdenken.

Die Auswahl und der Einsatz von Test Werkzeugen wird in anderen Büchern erschöpfend aufgezeigt (Siehe Literaturliste). Neben den in diesem Kapitel knapp gehaltenen Informationen finden Sie auf Seite 44 sind noch Links zum Thema.

Im wesentlichen gibt es drei Arten (Kategorien) von Testwerkzeugen:

- **Testverwaltungs-Werkzeuge;** Erfassen und Verwalten der Anforderungen, Erfassen und Verwalten der Testfälle, Testlauf Verwaltung, Defect Tracking, Reporting und Analyse, Rollenkonzept.
- **Regression-Testwerkzeuge;** Skripts von sogenannte Record & Replay Werkzeugen, sollten ohne viele Eingriffe aufgezeichnet werden können. Der Aufnahmemodus muss wählbar sein, zwischen objektorientiert und Koordinaten basierend. Sie automatisieren manuelle Testfälle indem Sie Skripts aufzeichnen und diese Skripts im Testverwaltungs-Werkzeug ablegen, d.h. einem Use Case zuordnen. Wichtig ist auch, dass die Skripts mit Testdaten dynamisiert werden können. Diese Testdatenerstellung erfolgt komfortabel mittels Agenten. Entwickler beherrschen diese Werkzeuge meist sehr schnell, da sie auf gängigen Programmiersprachen aufsetzen.
- **Performance-Testwerkzeuge;** Erzeugen einer gewünschten Anzahl von gleichzeitigen Benutzeraktionen auf Ihrem Zielsystem. Gleich wie beim Regression-Testwerkzeug werden die Benutzerbewegungen aufgezeichnet. Mittels virtuellen Benutzern (VU) wird das gleiche Skript beliebig oft, parallel oder Zeit versetzt gestartet/abgespielt. Durch Kombination von verschiedenen Skripts (Use Cases) erhalten Sie einen realistischen Bild, wie die Applikation (inkl. System) unter der „simulierten“ Last reagiert. Die Kosten richten sich bei den meisten Herstellern nach der Anzahl der VU. Schnell werden fünf- oder sechsstelligen Beträge erreicht.

Achten Sie unbedingt darauf, dass die Werkzeuge integrierbar sind. Werkzeuge eines Hersteller greifen meist verzahnt ineinander. Insellösungen bringen Sie nicht weiter, d.h. ein Regression-Testwerkzeug, dessen auftretende Fehler nicht direkt in das Defect Tracking Werkzeug übernommen werden können, verursachen unnötige Mehraufwände.

Tips

- Liste Werkzeugen siehe Seite 44
- Verwenden Sie integrierte Werkzeuge (alle drei oben genannten greifen ineinander)

- Ihr Prozess muss auch ohne Werkzeuge praktikierbar sein !
- „A fool with a tool is still a fool“
- Trennen Sie positive und negative Testfälle

DRAFT

2.2 Unit Test (Level 2)

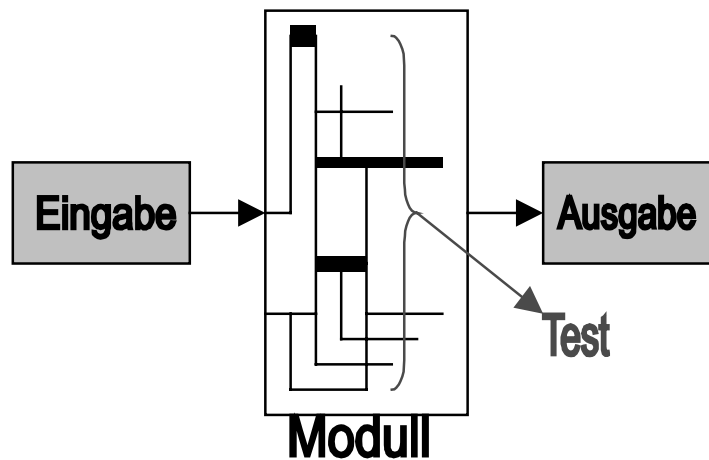
Der Unit Test basiert auf der methodischen Umsetzung von Entwicklerrichtlinien und Methoden, d.h. der aufgeführte Unit Test kann nur erfolgreich umgesetzt werden, wenn folgende Voraussetzungen erfüllt sind:

- Entwicklungsrichtlinien sind vorhanden und up to date, Verantwortlicher benannt
- Entwicklungsleiter wirkt als Mentor des Unit Tests
- Entwickler wurden für den Prozess geschult
- Prozesseinhaltung wird überwacht, Verantwortlicher benannt

Das Fehlen einer dieser Punkte stellt einen noch so raffinierten Unit Test Prozess in Frage.

Unit Tests sind in der Regel „White- oder Glas-Box-Tests“, d.h. die Programmlogik wird geprüft.

Tabelle 7: White Box Testing



daraus ergeben sich folgende Testbereiche:

- Standardeinhaltung; Kommentarzeilen, Schleifen, etc.
- Funktionskontrolle
- Codeänderungen gegenüber letzter (oder bestimmter Version); Verifikation mittels eines Werkzeugs wie z.B. Visual Source Safe oder RATIONAL® Pure Coverage.
- Laufzeitfehler und Speicher Verwendung; Verifikation mittels eines Werkzeugs wie z.B. RATIONAL® Purify.
- Performance Verhalten; Verifikation mittels eines Werkzeugs wie z.B. RATIONAL® Quantify.

Tabelle 8: Unit Test (Level 2)

Prozessfluss	Bemerkungen	Rollen	Dokumente, Daten, Tools ← input, → output
<pre> graph TD Start([Start]) --> ER[Entwicklungsrichtlinien] ER --> Box[] Box --> Diamond{ } </pre>	<p style="text-align: center; font-size: 2em; opacity: 0.5;">DRAFT</p>		<ul style="list-style-type: none"> ← input (used to produce an output) → output (resulting from an input) ←→input & output (template or form completed to a document or filled in form) ← Lessons Learned Berichte → Development Guideline



-

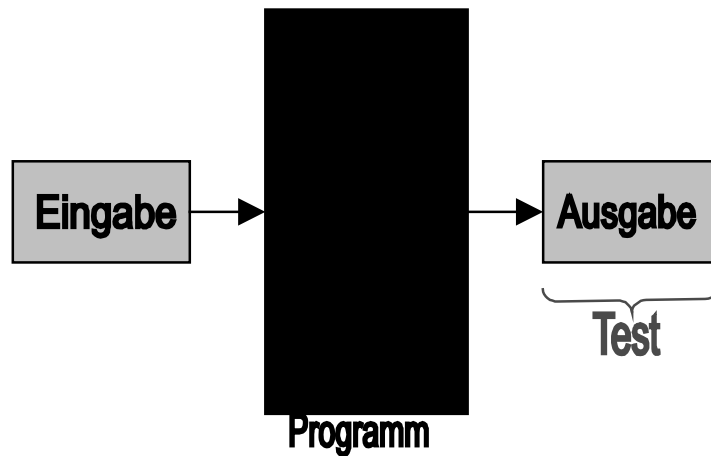
2.3 Unit Integration Test (Level 2)

Hier wird

.....

Unit Integration Tests sind in der Regel „Black-Box-Tests“, d.h. die Programmlogik wird nicht mehr geprüft. Die Tests sind ergebnisorientiert.

Tabelle 9: Black Box Testing



daraus ergeben sich folgende Testbereiche:

- Funktionskontrolle
- Anforderungs-/Spezifikationsüberprüfung
- Stabilität
- Interoperabilität
- Performance Verhalten

.....

i Tips

-

2.4 System Test (Level 2)

Hier wird

Tips

-

2.5 Load & Performance Test (Level 2)

An dieser Stelle entstehen oft Diskussionen mit Fachleuten die sagen: „Aber Performance Test ist doch keine Teststufe, sondern eine Testart, die innerhalb des Systemtests durchgeführt wird!“ Prinzipiell ist diese Aussage völlig richtig, aber mit dem Wandel in der IT hin zu Netzanwendungen wird dieser Test immer aufwendiger und immer wichtiger. Ausserdem ist das benötigte Know-how für die Vorbereitung und Konzeptionierung, sowie für die Durchführung mit Test Werkzeugen nur sehr rar vorhanden. Fast alle anderen Testaktivitäten können von angeleiteten Berufsleuten durchgeführt werden, bei Performance Test hört das Laienspiel aber auf. Hier brauchen Sie Profis. Ein Performance Test ist eine eigenständige Teststufe im Testprozess.

Wenn Sie sich fragen, ob diese Testart nötig ist, dann erinnern Sie sich daran, dass speziell bei e-commerce Anwendungen, die Zahl der Benutzer explodieren kann. Das Web ist potentiell Ihre grösste Filiale. Betreiben Sie also den Aufwand entsprechend.

Laut dem Rational Unified Process (RUP) Framework zerfällt der Performance in :

- **Benchmark Testing**; Vergleicht die Performance eines neuen Tests (noch nie durchgeführten) gegen ein bestehendes System(-teil).
- **Contention Test**; Verifiziert, das ein System Resource eine adequate Anzahl von Request abarbeiten kann. z.B. 30 DB Zugriffe in 10 Sekunden.
- **Performance Profiling**; Bei gleichbleibender Belastung wird die Konfiguration des System verändert, um zu sehen, ob Verbesserungen eintreten
- **Load Testing**; Bei gleichbleibender Konfiguration des System wird das Belastungsprofil (Anzahl Benutzer, Anzahl Transaktionen) verändert, um zu sehen, ob Veränderungen eintreten.
- **Stress Testing**; Bei gleichbleibender Konfiguration des System wird das Belastungsprofil (Anzahl Benutzer, Anzahl Transaktionen) annormal angehoben, um zu sehen, wann das System zusammenbricht.

Für die Umsetzung erstellen Sie zuerst ein Konzept welches folgende Punkte klärt:

- Welche Tests werden durchgeführt (Metriken, Anforderungen)
- Auf welchem System werden die Tests durchgeführt
- Unter welchen Bedingungen werden die Tests durchgeführt
- Wer führt die Tests durch

- Welche Werkzeuge werden eingesetzt
- Testdatenbereitstellung
- Test Dokumentation

.... INJEKTOREN

..... KOSTEN

..... GRAFIK PROZESS

.....

 Tips

-

2.6 User Acceptance Test (Level 2)

Hier wird

 Tips

-

2.7 Pilot Test (Level 2)

Hier wird

 Tips

-

3. Rollen im Testprozess

Definition : Eine Rolle.....

Alle am Testprozess beteiligte Personen übernehmen eine Rolle, die nicht immer vollamtlich sein muss. Wichtig ist, dass die Rolle klar definiert ist und von der betreffenden Person akzeptiert wird. Dieses Kapitel dient als roter Faden für die Erstellung der Rollen in Ihrem Projekt bzw. zur Erstellung von Rollenbeschreibungen in Ihrem Projektmanagement Prozess. Es ist durchaus möglich, dass Sie bei Ihrem Projekt zu dem Schluss kommen, die eine oder andere Rolle nicht zu benötigen.

Der Tester muss die Fähigkeit haben, die Kundensicht anzunehmen und diese zu vertreten und zu verteidigen. Dabei muss er nicht gleichermassen detaillierten Kenntnisse wie ein Benutzervertreter haben, sondern ein Verständnis der Sache. Tester müssen vom Charakter her „Nein-Sager“ sein, trotzdem aber flexibel genug, um einen Kompromiss zu erzielen, der der Sache dient.

Schon beim zweiten Projekt, in dem viele Leute aus dem ersten Projekt wieder zusammenarbeiten, gibt es fast keine Diskussionen über diese Rollen und deren Aufgabengebiete mehr.

Tabelle 10: Rollen im Testprozess

Rolle	Job Beschrieb
Architekt	<p>Syn. → Systemarchitekt, Systemdesigner, Designer</p> <p>Der Architekt legt die verwendeten Technologien und deren Zusammensetzung fest, die zur Erreichung der Auftragsziele notwendig sind.</p> <p>Kontaktrolle:</p> <ul style="list-style-type: none"> ▪ Projektleiter ▪ Entwickler (Chefentwickler) ▪ Externe Berater <p>Tätigkeiten:</p> <ul style="list-style-type: none"> • Definition der Systemarchitektur (logische und physische) • Grafische Darstellung • Erstellen Argumentebilanz, falls mehrere Varianten möglich sind
Auftraggeber	<p>Syn. → Client, Business, Business representative</p> <p>Dem Auftraggeber obliegt die Pflicht, sicherzustellen, dass er das bekommt, was er sich vorstellt. Da er dies in vielen Fällen wegen der hohen Komplexität der Aufgabenstellung nicht selber bewältigen kann, benennt er Benutzervertreter, die diese Aufgabe für ihn</p>

	<p>übernehmen. Die Benennung der Benutzervertreter und die Bereitstellung der Ressourcen (Verfügbarkeit in Absprache mit dem Linienvorgesetzten) liegt in seiner Verantwort und nicht in der des Projektleiters. Der Auftraggeber kann seine Pflichten an ein Steeringkomitee delegieren.</p> <p>Kontaktrolle:</p> <ul style="list-style-type: none"> ▪ Projektleiter ▪ Benutzervertreter ▪ Steeringkomitee <p>Tätigkeiten:</p> <ul style="list-style-type: none"> • Benennen der Benutzervertreter • Benennen der Mitglieder des Steeringkomitee • Abnahme der Teststrategie • Abnahme des Testplans • Abnahme von Meilensteinen
Benutzervertreter	<p>Syn. → Acceptance User, BV, Super User, Power User</p> <p>Wird teilweise oder voll von der Linie, dem Projekt zur Verfügung gestellt. Erarbeitet Testdrehbücher, kontrolliert Testszenarien.</p> <p>Kontaktrolle:</p> <ul style="list-style-type: none"> ▪ Test Manager <p>Tätigkeiten:</p> <ul style="list-style-type: none"> • Erstellen der Abnahmekriterien • Durchführen der Abnahmetests • Einbringen von Änderungsanträgen • Kontrolle der Requirements
Change Manager	<p>Syn. → Configuration Manager, Release Manager, Problem Manager</p> <p>Ist in den meisten Projekten ein Vollzeitjob. Der Change- und Releasemanager muss administrativ arbeiten, Änderungen und Releases verwalten. Technisches Verständnis ist zweitrangig. In ganz grossen Projekten oder reinen Linienorganisationen kann das Aufgabengebiet „Configuration Management“ und das Aufgabengebiet „Release Management“ völlig getrennt, durch zwei Rollen wahrgenommen werden.</p> <p>Kontaktrolle:</p> <ul style="list-style-type: none"> ▪ Projektleiter ▪ Entwickler <p>Tätigkeiten:</p> <ul style="list-style-type: none"> • Triggern der Änderungen • Zusammenstellen von Releases (Softwarestände) • Zusammentragen von Configurationsständen (Configuration

	<p>Management)</p> <ul style="list-style-type: none"> • Release Notes erstellen • Freigabe des Release nach Smoke Test • Umgebungsverwaltung (Entwicklungs-, Test- und Produktionssystem) • Publikation der Release Freigabe, z.B. durch Rundschreiben
Entwickler	<p>Syn. → Developer, Programmierer</p> <p>Mitglied des Entwicklungsteams. Muss durch Schulung für das Thema Qualitätssicherung sensibilisiert werden.</p> <p>Kontaktrolle:</p> <ul style="list-style-type: none"> ▪ Projektleiter ▪ <p>Tätigkeiten:</p> <ul style="list-style-type: none"> • Unit Test • Pult Test
Projektleiter	<p>Syn. → Projekt Manager</p> <p>Ist vom Auftraggeber eingesetzt, um das Projekt im Zeit und Budgetrahmen abzuschliessen.</p> <p>Kontaktrolle:</p> <ul style="list-style-type: none"> ▪ alle <p>Tätigkeiten:</p> <ul style="list-style-type: none"> • Benennen des Test Manager • Kontrolle des Testplans • ?
Tester	<p>Syn. → Test Executer</p> <p>Der Tester ist für das operative Testgeschäft zuständig.</p> <p>Kontaktrolle:</p> <ul style="list-style-type: none"> ▪ Test Manager ▪ Entwickler <p>Tätigkeiten:</p> <ul style="list-style-type: none"> • Einrichten Testinfrastruktur • Erstellen der Testfälle • Hilfe bei Testplan Erstellung
Test Manager	<p>Syn. → Test Co-Ordinator</p> <p>Der Test Manager ist meistens ein Teilprojektleiter in einem Grossprojekt. Er setzt den Testprozess zusammen mit seinem Team auf. Er stösst meist früh zu dem Projektteam und kennt daher meist die „geschichtlichen“ Hintergründe des Projekts. Die zu späte Besetzung dieser Rolle führt meist zu hektische Betriebsamkeit,</p>

	<p>bringt aber für die Qualität des Produkts wenig. Am Anfang des Projekt, noch während der Architekturphase, kann ein Testmanager den grösstmöglich Qualitätsnutzen für das Projekt erzielen.</p> <p>Kontaktrolle:</p> <ul style="list-style-type: none"> ▪ alle <p>Tätigkeiten:</p> <ul style="list-style-type: none"> • Benennen der Testteam Leiter (Teilprojektleiter) • Benennen der Tester • Erstellen des Testplans • Monitoring des Testfortschritts • Reporting • Eskalation der Ausnahmefälle •
Test Tool Engineer	<p>Syn. → Tool Consultant</p> <p>Der Tester Tool Engineer ist ein Tester, der sich auf ein oder mehrere Werkzeuge spezialisiert hat und Methodiken des Testens kennt. Diese Kenntnisse können auch fallweise und über einen begrenzten Zeitraum von dem Hersteller des betreffenden Werkzeuges gemietet werden.</p> <p>Kontaktrolle:</p> <ul style="list-style-type: none"> ▪ Tester ▪ Test Manager <p>Tätigkeiten:</p> <ul style="list-style-type: none"> • Einrichten Testinfrastruktur • Einrichten Testwerkzeug • Unterstützung Testwerkzeug • Testdurchführung

In grösseren Organisationen oder Firmen, die viel Software Entwicklungsprojekte durchführen, lohnt sich eventuell die Schaffung eines „Competence Centers“, das sich nur mit Testing und Qualitätssicherung beschäftigt.

3.1 Schulung

Die Einarbeitungsphase von neuen Testteam Mitgliedern ist sehr wichtig. Erhalten Sie Ihre Testteam Mitglieder nicht von einem Competence center, so müssen Sie auch einen gewissen Aufwand für die Schulung & Weiterbildung betreiben.

Stellen Sie einem neuen Teammitglied einen Mentor für die Anfangsphase zur Verfügung.

....Dokumente

.... externe Schulung

.... lessons learned

DRAFT

4. Testdaten

Kernaussage : Wo Input nötig ist, werden Testdaten gebraucht.

Testdatenmanagement ist wohl die meist unterschätzte Disziplin beim Testen von Software. Oft wird der Aufwand für das Erstellen und Pflegen von Testdaten völlig unterschätzt. In einem umfangreichen e-commerce Projekt habe ich erlebt, dass mehr als zehn Leute für das Datendesign zuständig waren, aber gerade nur eine Person für die Testdaten. Das heisst, eine Person versucht die gleiche Arbeit zu tun wie es die anderen zehn Entwickler tun.

Will man ein komplettes Banking System testen, so braucht man Kunden, diese Kunden brauchen Konten (in diverse Währungen) und Portfolios (in diverse Währungen), diese Kunden brauchen eine Kreditlimite, ein Rating (Kundenbewertung), einen Kundensachbearbeiter, ein Filiale, ein Verhaltensprofil.

Tätigt dieser Kunde nun einen Aktienkauf, so muss zuerst das Konto mit Guthaben vorhanden sein, dann muss zumindest ein leeres Portfolio vorhanden sein. Will man eine Pan-Europäische Bank mit 500'000 Kunden Testen, dann stehen Sie vor einer Herausforderung.

Aufgrund dieser Bewegungsdaten möchten Sie dann alle „End of“ Jobs durchspielen und die Daten verifizieren.

Zugleich meldet das MIS (Management Information System) einen Bedarf zum Testen der spezifizierten Reports an und

Dieses Beispiel, das zugegebenermassen nicht ganz alltäglich ist, zeigt die Dimensionen in denen Sie sich als Testdaten Verwalter bewegen. Und wir haben noch nicht von Fall Back Szenarien, Hot Swapable Systems und Load Balancing geredet !

Tips

- Der Datenkatalog muss die Basis für die Testdaten Erstellung bilden !
- ?

4.1 Testdaten für Load & Performance Tests

Die Bereitstellung der Load & Performance Testdaten stellt wieder etwas andere Anforderungen als die normalen Test Daten. Die Daten müssen nur noch bedingt „logisch“ sein, dafür steigt die Menge der Daten an.

Die genaue Zahl der benötigten Testdaten entspringt den Business Requirments bzw. eine Load & Performance Test Strategie. Ich nehme nochmals das Beispiel mit der Pan-Europäische Bank mit 500'000 Kunden auf.

Bevor Sie mit den Tests bzw. deren Vorbereitungen beginnen, müssen Sie mit dem Kunden im Minimum eine Matrix festlegen, die für das Beispiel etwa so aussehen könnte:

Tabelle 11: Load & Performance Beispiel Matrix

Load & Performance Volumetric	Tag 1	6 Monate	1 Jahr	5 Jahre
Kunden				
Kunden (Spitzenwert = normal * 10)				
Kunden Transaktionen pro Tag (mit Log in)				
Besucher Transaktionen pro Tag (ohne Log in)				
Total Transaktionen pro Tag				
Transaktionen pro Stunde [Tag/8 Arbeitsstunden]				
~				
zusammengesetzt aus folgenden Geschäftstransaktionen :				
Transaktion # 1				
Transaktion # 2				
Transaktion # 3				
Transaktion # 4				
Transaktion # 5				
Transaktion # 6				
Transaktion # 7				
Transaktion # 8				
Transaktion # 9				
Transaktion # 10				

Mit dieser Matrix wissen Sie, was an Arbeit auf Sie zukommt. Testwerkzeug, Testdaten, Tester, Zeitaufwand usw. können jetzt geschätzt werden.

Warum genau vier Durchläufe, warum zehn Transaktionen werden Sie fragen. Beide Zahlen sollten dem effektiv erwarteten Aufkommen, also der erwarteten Realität so nah wie möglich kommen. Die Anzahl Kunden werden während bzw. vor dem Test in die Datenbank geladen, die Anzahl Transaktionen werden simuliert während der

festgelegten Testdauer. Also, wie gesagt, es ist nur noch eine bedingte Logik der Daten nötig, nämlich in soweit, dass die Transaktionen # 1 –10 teilweise Abhängig sind, so das diese Abhängigkeiten von den Testdaten und vom Testablauf beachtet werden müssen.

Zum Beispiel ist Transaktion # 1 ein „Aktienverkauf“ für Kunde xy. Dies setzt voraus, das Kunde xy:

- existiert im Kundenstamm
- xy ein Portfolio besitzt
- das Portfolio die besagte Aktie in der richtigen Anzahl enthält
- xy ein Konto für die Gutschrift in der richtigen Währung besitzt
- die Daten für den Verkaufauftrag vorhanden sind

d.h. vor Transaktion # 1 müssen n Transaktionen laufen und zwar in der richtigen Reihenfolge.

Kernaussage : Testdaten können „verbrennen“, d.h. ist eine gewisse Transaktion einmal ausgeführt worden, kann die gleiche Transaktion nicht in jedem Fall nochmals ausführen, weil z.B. keine Aktien mehr im Portfolio sind. Es müssen neue Datensätze verwendet werden, oder das Gesamtsystem muss in einen früheren Zustand zurück versetzt werden. Der Aufwand ist nicht zu unterschätzen.

5. Dokumente im Testprozess

Nachfolgend finden sie eine Liste mit Dokumenten, die Ihnen einen Überblick über die Dokumente im Testprozess verschafft.

Tabelle 12: Dokumente im Testprozess

Phase	Dokument	Wer	Bemerkung
Initialisierung	Test Strategie	Test Manager ① ?
Initialisierung	Test Plan	Test Manager	Rollen und ev. Rollenbesetzung, Zeitplan ① Als Basis dient die Teststrategy
Vorstudie			
Vorstudie	Systemarchitektur	Architekt ①
Vorstudie	Risikoanalyse	Projektleiter Testmanager	① Testaufwand für Bereiche muss durch Risikobewertung gerechtfertigt werden
Vorstudie		 ①
Vorstudie	Qualitätsplan	Qualitätsmanager Testmanager	Reviewplanung, Messverfahren, Rollen ①
Vorstudie	Projektplanung	Projektleiter ①
Vorstudie		 ①
Vorstudie		 ①
Vorstudie	Use Case Beschreibungen	Kunde Entwickler	Muss Mengengerüste enthalten, bzw. diese müssen separat verfügbar sein. ①
Vorstudie		 ①
Vorstudie		 ①
Hauptstudie		 ①
Hauptstudie	Data Dictionary	Architekt Entwickler ①
Hauptstudie		 ①
Hauptstudie		 ①
Detailstudie		 ①
Detailstudie	Benutzerhandbücher	 ①
Detailstudie		 ①
Einführung		 ①
Einführung		 ①
Einführung		 ①
Einführung		 ①
Übergabe		 ①
Übergabe		 ①

Übergabe		 ⓘ
Übergabe		 ⓘ
Übergabe		 ⓘ

Wenn Sie in mehr als einem Projekt mitgearbeitet haben, dann wissen Sie, dass viele dieser Dokumente, die sicher Sinn machen, manchmal nur als Template auf dem Projektdokumentenserver liegen. Dass etwas nicht, bzw. meistens nicht richtig funktioniert, ist noch kein Grund, das Sie nicht versuchen sollten, diese Dokumente zu bekommen. Von einem Flugzeugmechaniker verlangen Sie ja auch, das er volles Engagement liefert, obwohl jeder weiss, dass es keine absolute Sicherheit gibt.

Methodisches Testen macht nur begrenzt Sinn, wenn das gesamte Projekt nicht methodisch geführt wird. Firmen, die als Kernkompetenz das Abwickeln von Projekten betreiben sollten sicher einen Projekt Management Prozess haben. Dieser Prozess schreibt Vorgehensweisen, Methoden, Meilensteine, Lieferobjekte etc. vor. Diese Frameworks können eigene Entwicklungen, oder noch besser Adaptionen von bestehenden Musterprozessen sein. Stellvertretend sei hier RUP (Rational Unified Process) genannt, den es sowohl in Deutsch, wie auch in Englisch gibt; siehe www.rational.com.

ⓘ Tips

- Definieren Sie gewisse Dokumente als Meilensteine
- Veröffentlichen Sie regelmässig Lessons Learned Berichte zum Thema Software Testing
- Veröffentlichen Sie regelmässig Lessons Learned Berichte zum Thema Projekt Management
-

6. Prozess Kennzahlen (KPI)

Prozess Kennzahlen sind wichtige Hilfsmittel für die Fortschrittsüberwachung und die Messung der Produktequalität. Gut gewählte Kennzahlen lassen auch Knowledge Transfer im Sinne einer lernenden Organisation zu. Wenn Sie zum Beispiel die Function Point Methode (Schätzmethode siehe ????????) für die Aufwandschätzung verwenden, dann benötigen Sie Erfahrungszahlen, um möglichst genau schätzen zu können.

Leider entsteht beim Daten sammeln immer wieder der Eindruck, das dies zur Messung der persönlichen Leistung einzelner Mitarbeiter geschieht. Dieser Irrtum kann und muss durch gute Kommunikation vermieden werden, da sonst ein „Graben“ zwischen Test und Entwicklung oder zwischen Chef-Entwickler und Entwickler entsteht, der sich langfristig kontraproduktiv auswirkt.

Tabelle 13: Prozess Kennzahlen (KPI's)

Kennzahl	Formel	Bemerkung
	[Input]	[Input]
Testabdeckung %	$\frac{\text{Anzahl UC} \times \text{erwartete Anzahl Tests pro UC}}{\text{Anzahl aktuelle TestCases}}$	Diese Kennzahl ist ein wichtiger Indikator in der Vorbereitungsphase.

.... Metriken

Tips

- Machen Sie transparent, was Sie messen !
- Machen Sie transparent, wie Sie messen !
- Machen Sie transparent, warum Sie messen !
- Die Produktequalität ist das Ziel der Messung
- .

7. Argumente für das Testen

7.1 Fragen, die unbedingt gestellt werden sollten

Im Kundenkontakt oder auch innerhalb eines Projektteams wird früher oder später das Testen bez. der Aufwand für das Testen in Frage gestellt. Hier möchte ich Ihnen einige Fragen auflisten, die Sie dann den Verantwortlichen oder dem Kunden, der die Kosten senken möchte, stellen sollten.

- Was passiert, wenn die Komponente/das Programm in der Produktion versagt (Auswirkung) ?
- Ist ein Versagen Existenz bedrohend ?
- Können wir uns den Imageverlust, selbst für einen kleinen Fehler leisten ? („Die können nicht einmal eine Webseite betreiben, und jetzt wollen die mein Geld haben ?“)
- Kann ein Fehler Schadenersatzansprüche nach sich ziehen ?
- .
- .

7.2 Katastrophen Beispiele

Im Internet finden sich unter den Begriffen „Softwarepanne“, „Software Ausfall“ etc. viele Beispiele zu diesem Thema.

7.2.1 CS-Bancomaten ziehen EC-Karten ein (Schweiz)

Quelle : Meldung Teletext SFDRS 24.12.00 21:46

Nach den Pannen bei der Telekurs AG mit ECdirect vom Samstag sind am Sonntag an den Bancomaten der Credit Suisse CS alle EC-Karten eingezogen worden. Die CS-Kundenhotline bestätigte eine entsprechende Meldung von "Radio 24". Der Fehler habe um 20.40 Uhr behoben werden können. Grund für die Panne war ein **Systemupdate** bei CS, das unerwarteterweise das Bancomatensystem in Mitleidenschaft gezogen habe, hiess es. Am Samstag hatte die Fehlmanipulation eines Roboters im Rechenzentrum der Telekurs AG von 13.15 bis 15.25 Uhr den Zahlungsverkehr lahmgelegt. 100'000 ECdirect-Tansaktionen waren betroffen.

7.2.2 Credit Suisse: Ganztägiger Systemabsturz Basel/Zürich

Quelle : Baz 17.5.2001 bot

Credit Suisse: Ganztägiger Systemabsturz Basel/Zürich. Was Matthias Friedli, Pressesprecher der Credit Suisse, gestern mit «einem kleinen Rumpfer in unserem System» abtun wollte, war in Wirklichkeit ein grosser Systemausfall: Die Kunden der Credit Suisse konnten gestern schweizweit weder an den Geldautomaten Geld

beziehen noch Geschäfte über das Internet tätigen. Wie Friedli auf Anfrage der BaZ erklärte, kam es bei der CS zu einem «nicht alltäglichen technischen Datenbank-Problem». Folge: Funktionen mit Zugriffen auf die Kunden-Datenbank waren nicht mehr möglich. Mit der EC-Karte habe der Geldbezug hingegen funktioniert. Entdeckt wurde der Systemausfall um 10.00 Uhr. Danach wurden alle Systeme heruntergefahren und frisch aufgestartet, was bis in den Abend hinein andauerte, «da die Systeme sehr komplex verknüpft sind», so Friedli. Sicherheitsprobleme hätten dabei aber keine bestanden. Schaden sei gemäss seinen Informationen keiner entstanden. Einzig Unannehmlichkeiten hätten für die Kunden - wie auch für das CS-Personal - aufgrund des Systemabsturzes bestanden, so Friedli.

7.2.3 Coop: Online Shops – Supermarkt

Quelle : 03.09.2001 Original Webseite www.coop.ch/supermarkt

Liebe Kundinnen und Kunden Liebe Besucherinnen und Besucher

Vielen Dank für Ihren Besuch. Alle wollen den neuen Internet Shop von Coop besichtigen! Deshalb hat sich vor ihm eine Warteschlange gebildet. Das Team arbeitet an Lösungen, um den Ansturm zu bewältigen.

Wir versuchen den Shop sobald wie möglich wieder zu öffnen.

Selbstverständlich können Sie sich trotzdem Ihre Einkäufe bequem nach Hause liefern lassen. Bestellen Sie einfach per Telefon: 0848 847 848

Diese Meldung wird regelmässig aktualisiert. Letzte Aktualisierung : Samstag, 1. September 2001, 17:53

Wir danken für Ihr Verständnis.

7.2.4 Der CS-Internetbroker Youtrade ist vorübergehend blockiert gewesen

QUELLE: BAZ ONLINE 21.09.2001 - 17:15 Uhr sda

Der CS-Internetbroker Youtrade ist vorübergehend blockiert gewesen ZÜRICH - Der Internet-Broker der Credit Suisse, Youtrade, ist während rund 24 Stunden unterbrochen gewesen. Von gestern nachmittag bis heute um 16 Uhr waren online keine Transaktionen möglich. Ursache sei ein technisches Problem im Netzwerk von Youtrade gewesen. Dies bestätigte ein CS-Sprecher. Kunden hätten aber telefonisch Transaktionen tätigen können.

Die CS entschuldige sich bei den rund 28 000 Benutzerinnen und Benutzern von Youtrade für die Panne, die ausgerechnet mit massiven Kursverlusten an den Börsen zusammenfiel.

7.2.5 Swisscom Kunden ohne Internet-Zugang

QUELLE: BAZ ONLINE 04.10.2001 - 08:53 Uhr SDA/BAZ

Tausende von Swisscom-Kunden waren am Mittwoch zwischen 14.30 und 17 Uhr vom Internet abgeschnitten. Ursache war eine Softwarepanne. Sie konnte am erst späten Nachmittag wieder behoben werden. Betroffen waren alle Internetbenutzer, die über die Nummer 0840 auf das Internet gelangen, wie Swisscom-Sprecher Christian Neuhaus sagte.

7.2.6 Softwarepanne bei SBB Billettautomaten

QUELLE: 58. Sitzung des Gemeinderates von Zürich Mittwoch, 14. Juli 1999, 20.45 Uhr, im Rathaus

Geschäft 1357. 99/348 Von Christian Mettler (SVP) ist am 14.7.1999 folgende Interpellation eingereicht worden:

Am 7. Juli 1999 hat eine Softwarepanne den Verkauf von Halbtaxbilletten und Monatskarten lahmgelegt. Betroffen waren nicht nur 400 SBB- und 280 Bahnhöfe von Privatbahnen, sondern auch die Verkaufsstellen des ZVV. Nach kurzem Unterbruch konnten die 20 umsatzstärksten Bahnhöfe der Schweiz wieder normal arbeiten.

In diesem Zusammenhang bitte ich den Stadtrat um die Beantwortung der folgenden Fragen:

1. Warum konnten die Verkaufsstellen in der Stadt zum Teil erst nach 24 Stunden wieder Monatskarten und Abonnements ausgeben?
2. Wie hoch sind bei den VBZ die geschätzten Mindereinnahmen und Folgekosten?
3. Wer deckt die entstandenen Unkosten und Mindereinnahmen der Verkaufsstellen der VBZ?
4. Welche Massnahmen sind getroffen worden, um bei einer erneuten Computerpanne unabhängig handeln zu können?
5. Weshalb können die Abonnementsausgaben nicht losgelöst vom Zentralcomputer erfolgen?
6. Wer betreut die Software und Hardware der Ticket-Shops der VBZ?

7.2.7 Bundestag hatte ein Jahr-2000 Problem

QUELLE: Textarchiv Berliner Zeitung, 14.01.2000, Lokales, Autor Ulrich Paul

Beim Bundestag hat es zum Jahreswechsel bei den Computern Probleme durch die Datumsumstellung gegeben. Der Bonner Bundestagsabgeordnete Norbert Hauser (CDU) berichtete erst jetzt, dass ein "teilweiser Ausfall unserer Software die gesamte elektronische Kommunikation via E-Mail" lahm gelegt habe. Dadurch sei es für die Wahlkreisbüros eine Woche unmöglich gewesen, elektronische Post zu öffnen oder zu verschicken. Weil er oft E-Mail benutze, bedeute der Ausfall der Software "eine erhebliche Behinderung" seiner Arbeit, kritisierte Hauser. Nachdem er den Direktor des Bundestags auf die Missstände aufmerksam gemacht habe, sei das Problem behoben worden. Bundestagssprecher Hartwig Bierhoff bestätigte den Vorfall. "Es war ein kleines Jahr-2000-Problem". Hauser habe das Problem aber erst bemerkt, als die Schadensbeseitigung bereits angelaufen sei.

8. Risiken

Risikobewertung zeigt Ihnen, wo das Testen am wichtigsten ist. Ein identifiziertes Risiko, dessen Eintrittswahrscheinlichkeit „hoch“ und dessen Auswirkung „hoch“ ist, muss sicher die volle Aufmerksamkeit des Testteams haben.

Risikoanalysen werden wenn überhaupt auf der Basis des Gesamtprojekts durchgeführt. Sollten Sie Einfluss auf die Risikoanalyse haben, so drängen Sie darauf, dass sie in sinnvolle Bereiche unterteilt wird. Hier einige Beispiele für solche Bereichsunterteilung für ein **e-Commerce Projekt**; Gesamtprojekt, Mitarbeiter, Funktion (Use Case), Security, Netz, Applikationsserver usw.

Eine so unterteilte Risikoanalyse gibt Ihnen schnell Aufschluss über Ihren besonderen Aufmerksamkeitsbedarf.

Tabelle 14: Risikobewertung

No	Bereich	Risiko	Eintrittswahrscheinlichkeit	Auswirkung
1	Mitarbeiter	Programmierressourcen können nicht rechtzeitig gefunden werden	mittel	hoch
2	Budget	Ungenauere Kostenkalkulation (falsche Anwendung der Schätzmethode)	mittel	mittel
3	Termine	Unrealistische Termine	gross	hoch
4	Termine	Terminabhängigkeiten (Kritischer Pfad)	gross	mittel
5	Zulieferer	Abhängigkeit von Zulieferern (Termine/Know How)	gross	mittel

Tips

- Die Risikoanalyse sollte immer **eine** der Grundlagen Ihrer Testfallerstellung sein.
- Gibt es keine Risikoanalyse, so drängen Sie darauf, dass eine erstellt wird. Erstellen Sie diese notfalls selber. Publizieren Sie Ihre Risikoliste.
- ABC-Analyse Methode kann sehr hilfreich sein.

9. Anforderungen (Anforderungsspezifikation)

Seit dem V-Modell bilden die Anforderungen (requirements) eine Basis für die Testfallerstellung.

Die Qualität der Anforderungen ist ein Schlüsselfaktor für den Erfolg des Projekts. Meist werden diese in Textform oder in UML Format von einem Design Team erstellt. Und an diesem Punkt passieren die meisten Fehler, die sich später in massiven Kosten für die Fehlerbehebung oder gar für eine Re-Design eines Teiles oder der ganzen Applikation niederschlagen.

Folgende Punkte minimieren dieses Risiko gegen Null:

- Anforderungen müssen messbar sein; Begriffe wie Benutzerfreundlich oder leicht erlernbar müssen erklärt sein, wie und woran soll gemessen werden → den messbar = testbar !
- Der Kunde muss die Anforderungen verifizieren (in einer Notation, die er versteht)
- ...

-> Notation in Lexikon

-> Bezug zu Test (TAF) Automation Framework, Requirments Fehler werden spät entdeckt und kosten sehr will !!!

Tips

- Der Kunde muss die Anforderungen verifizieren (in einer Notation, die er versteht)
-

10. Kosten im Testprozess

Dieses Kapitel steht bewusst an dieser Stelle. Jetzt kennen Sie viele Fakten und Gründe, warum gewisse Aktivitäten im Testprozess unumgänglich sind.

Personal, HW, SW -> Bezug zur Risikoanalyse !

Tips

- Beobachten Sie den schleichenden Funktionszuwachs mittels Konfigurationsmanagement
- Planen Sie Testkosten von Anfang an
- Kalkulieren Sie Testkosten realistisch (Verifikation durch Dritte) !
-

A. Appendix A

A. Abkürzungen

Tabelle 15: Abkürzungen

Abkürzung	Definition

A.....

B. Referenzen (~ oder Literaturliste)

Folgende Bücher

Tabelle 16: Bücher zum Thema

Titel, Detailangaben
Management und Optimierung des Testprozesses Pol, Koomen, Spillner: Ein praktischer Leitfaden für erfolgreiches Testen von Software mit TPI und TMap. 2000. XVI, 543 S. ISBN: 3-932588-65-7- DPUNKT VERLAG
Methodisches Testen von Programmen Der Schwerpunkt der Darstellung liegt auf der Entwicklung von effektiven Testfällen. Darüber hinaus werden auch ökonomische und psychologische Aspekte von Programmtests berücksichtigt sowie Managementinformationen, Testwerkzeuge, High-Order, Testing, Fehlerbehebung und Code-Inspektionen. Autor: G.J. Myers 7. Auflage ISBN: 3-486-25634-3

11. Links

Diese Links wurden zum Zeitpunkt der letzten Auflage des Buches getestet. Eine aktuelle Link Liste finden Sie unter www.guideline.ch/links_d.html.

11.1 Projektmanagement

Name	Inhalt	URL
Checklisten	Universität Stuttgart; Checklisten für den Software-Entwicklungsprozess	http://www.informatik.uni-stuttgart.de/ifi/se/service/checklists/index.html
ERNI Consulting AG	Erfahrungsberichte im PDF Format	http://www.erni.ch
Schweiss Projekt	Was jeder Projektleiter wissen muss	www.scheissprojekt.de
Scheuring	MS Projekt Zusatz Software	www.scheuring.ch

11.2 Testen von Software

Name	Inhalt	URL
CCTA Library	IT Infrastructure Library developed by CCTA, the UK Government's Central Computer and Telecommunications Agency.	http://www.proactive-sv.com.au/public.htm
QUALITY TECHNIQUES NEWSLETTER	Newsletter	http://www.soft.com/News/QTN-Online/subscribe.html
SQE	Kurse, Newsletter	http://www.sqe-europe.com
StickyMinds.com	Regelmässige Informationen	http://www.stickyminds.com
Testing Foundations	Test Consulting	http://www.testing.com

11.3 Tools

Name	Inhalt	URL
Astra SiteManager®	Mercury® Astra SiteManager testet dead Links	http://tryandbuy.mercuryinteractive.com/cgi-bin/portal/trynbuy/index.jsp
AutoTester®		http://www.autotester.com
Compuware®	Tool Manufacturer	http://www.compuware.com
Mercury®	Tool Manufacturer	http://www-svca.mercuryinteractive.com/
Cyrano Webtester®	Tool für Web Applikationen	http://www.cyrano.com/
eTest Suite	Performance Tool	http://www.empirix.com
eValid	Web Applikation Tool	http://www.e-valid.com
Rational®	Tool Manufacturer; most intergrated Tool suite.	http://www.rational.com/products/siteload/index.jsp
Saturn	Performance Tool	http://www.zott.net
Segue®	Segue Software, Inc., the e-business reliability experts	http://www.segue.com
TestView WebLoad	Performance Tool	http://www.radview.com
TPC Benchmark™	Performance Tool für Web Applikationen	http://www.tpc.org/tpcw/
TPNS		http://www.tpns.com
Trans Century Enterprise Tester		http://www.emeraldsoft.com
Übersicht	Übersicht von Link Test Werkzeugen	http://www.elsop.com/wrc/comp_ls.htm
Web Application Stress	Performance Tool	http://www.microsoft.com

11.4 Workflow

Name	Inhalt	URL
Projektstudie	Workflow Management in Gießereien	http://www-wi.cs.uni-magdeburg.de/~kroetz/wfpage/wfm_page.html

11.5 Standards / Standard Organisationen

Name	Inhalt	URL
ASQF Erlangen	Arbeitskreis Software-Qualität Franken e.V	http://www.asqf.de
ETSI	European Telecommunication Standards Institute	http://www.etsi.org
FPM	Foundation for Performance Measurement Web Site	http://www.fpm.com
ICSTEST	International Conference on Software Testing	http://www.icstest.com/
IEEE		http://www.ieee.org
IEEE Standards		http://www.standards.ieee.org
ITU	International Telecommunication Union	http://www.itu.int
Software Productivity Consortium	TAF-Standard	https://www.software.org
The Open Group		http://www.opengroup.org

12. Organisationen

IT Organisationen im deutschen Sprachraum, die sich „Englisch“ als Norm für Dokumente und Prozesse auferlegt haben, sind nach meiner Erfahrung viel offener gegenüber externen Beratern und Mitarbeitern (nicht nur aus dem englischen Sprachraum), sondern reagieren auch wesentlich flexibler auf bei Umstellungen. Ein schlechtes (erlebtes) Beispiel ist die super organisierte IT-Organisation, die Probleme hat, eine Testsoftware, die es nur in der englischen Version gibt, zu installieren. Der ungeübte Mitarbeiter wird dann auch grosse Mühe haben, wenn er mit der „amerikanischen“ Supporthotline zu tun hat.

13. Dokumentenbeispiele

DRAFT

14. Stichwortregister (Teil 1 & 2)

→ Erstellen mit ALT+UMSCHALTEN+X

DRAFT

Teil 2

DRAFT

15. TestLex - Test- und Qualitätsmanagement Lexikon

Dieser Teil soll Fachausdrücke aus dem Bereich QM und Software Testing verständlich erklären.

Im wesentlichen empfehle ich die Verwendung von Begriffen basierend auf der **ISO 8402**, damit sich neue Mitarbeiter und externe Ansprechpartner schneller mit Ihrem Prozess vertraut machen können.

15.1 Handhabung

Benutzen Sie diesen Teil als Nachschlagewerk. Ergänzungen und Anregungen siehe Kapitel Feedback.

Alphabetisch geordnet, Leerschlag und Bindestrich werden als nicht existent betrachtet, d.h. **Contention Test** kommt vor **Content Management**.

15.2 Lexikon

Start

DRAFT

ABC-Analyse

→ allgemein

eng. -

Besagt, dass 20% einer Menge 80 % Einfluss haben (A), 30% haben 50 % Einfluss (B) und 50 % lediglich 5 % (C). Es gilt also, sich auf die A Produkte/Risiken etc. zu konzentrieren.

Abnahmekriterium

→ allgemein

eng. acceptance criterion

Synonym Akzeptanzkriterium. Diejenigen Kriterien, die ein SW-Produkt erfüllen muss, um einen mit dem Auftraggeber festgelegten Abnahme-Test zu bestehen oder den Fertigstellungs-Anforderungen zu genügen.

Abnahme Test

→ allgemein

eng. acceptance test

Synonym Akzeptanz Test. Formeller Test zu dem Zweck, festzustellen, ob ein SW-Produkt die daran gestellten Abnahme-Kriterien erfüllt. Er bildet den Nachweis der Erfüllung der vereinbarten Anforderungen gegenüber dem Auftraggeber.

Acceptance User, siehe **Power User**

Access Concentrator

→ allgemein

deu. -

Gegenstelle der PPPoE-Verbindung, entspricht dem POP bei Wählverbindungen, daher auch ‚Breitband-POP‘.

ADSL

→ allgemein

eng. Asymmetric Digital Subscriber Line

Ermöglicht hohe Übertragungsraten auf den alten Kupferleitung mit bis zu 6 MBit/s. Im Frequenzbereich zwischen 138 kHz bis 1,1 MHz. Rund 140 mal schneller als eine herkömmliche 56K-Modemverbindung. Geringer Montageaufwand. Gleichzeitiges telefonieren möglich.

Akzentanzkriterium, siehe **Abnahmekriterium**

Akzeptanz Test, siehe **Abnahme Test**

Anforderung

→ V-Modell

eng. requirement

Messbare Eigenschaften, Forderungen und Merkmale. Anforderungen müssen auf ihre Erfüllung geprüft werden können.

Application Service Provider, siehe **ASP**

Application Service Providing, siehe **ASP**

Architekt

→ Rollen im Testprozess

eng. architect, designer

Definition der Systemarchitektur (logische und physische).

Artefakt

→ allgemein

eng. artifact

Lieferobjekte (Lieferergebnis) im Entwicklungsprozess, z.B. ein Testkonzept, Testplan, Systemarchitektur etc. Ausdruck wird auch im RUP verwendet.

ASP

→ allgemein

eng. Application Service Provider

eng. Application Service Providing

Ein Dienstleister, der Anwendungssoftware vermietet, welche auf Servern in einem Rechenzentrum gehostet wird und auf die online, unabhängig vom Betriebssystem, zugegriffen werden kann. Der Aufwand für Installation, Update, Hardware Pflege etc. wird für den ASP Kunden reduziert. Die Technik „Application Service Providing“ wird ebenfalls mit der Abkürzung ASP bezeichnet, was zu Verwirrungen führt.

Assessment criteria, siehe **Prüfschärfe**

Auftraggeber

→ Rollen im Testprozess

eng. client

Erteilung des Auftrages und Bereitstellung der benötigten Ressourcen. Abnahme von Teil- und Endprodukten.

B2B

→ allgemein

eng. Business to Business

Elektronischer Internethandel von Waren und Dienstleistungen zwischen Unternehmen, oft über B2B-Marktplätze.

B2C

→ allgemein

eng. Business to Consumer

Elektronischer Internethandel zwischen Unternehmen und Endverbraucher.

Benchmark Testing

→ RUP – Performance Testing

deu. -

Vergleicht die Performance eines neuen Tests (noch nie durchgeführten) gegen ein bestehendes System(-teil).

Benutzerakzeptanz Test, siehe **UAT**

Benutzerfreundlichkeit

→ V-Modell

eng. usability

Eigenschaft einer Software, die es erlaubt, die Aufwände für Einarbeitung oder Benutzung so gering wie möglich zu halten. Nach [ISO 9126] ist Benutzerfreundlichkeit ein Qualitätsmerkmal.

Es umfasst die Untermerkmale
- Verstehbarkeit (Understandability)
- Erlernbarkeit (Learnability)
- Handhabbarkeit (Operability)

Benutzervertreter, siehe **Power User**

Betriebsqualität

→ V-Modell

eng. ?

Mass für die Einhaltung der Sicherheitsanforderungen während des Betriebs eines IT-Systems, insbesondere in Ausnahmesituationen wie z. B. Fehler, Wartung.

Black Box Test

→ allgemein

deu. -

Das Resultat des Test wird kontrolliert, unabhängig, von der Programmlogik. Wie das Resultat erzielt wird, wird nicht kontrolliert. Das Gegenteil ist White Box Testing.

Business Excellence

→ allgemein

eng. -

deu. Spitzenleistungen eines Unternehmens
Business Excellence bezeichnet langfristige und nachhaltige Spitzenleistungen eines Unternehmens im Führen der Organisation und im Erzielen von Ergebnissen. Exzellente Ergebnisse werden - nach der Definition der European Foundation for Quality Management EFQM - im Hinblick auf Geschäftsergebnisse, Kunden, Mitarbeiter und Gesellschaft durch eine Führung erzielt, welche aktiv dafür sorgt, dass die Politik und Strategie, die Mitarbeiter, die Zusammenarbeit mit Partnern, der Einsatz der Ressourcen und die Unternehmensprozesse kundenorientiert entwickelt, geführt und stetig verbessert werden. Als Ergebnisse resultieren überdurchschnittliche Kunden-, Mitarbeiterzufriedenheit und Beiträge an die Gesellschaft, was letztendlich zu hervorragenden Geschäftsergebnissen führt.

Business to Business, siehe **B2B**

Business to Consumer, siehe **B2C**

BV, Benutzervertreter siehe **Power User**

C2C

→ allgemein

eng. Consumer to Consumer

Elektronischer Internethandel zwischen Privatpersonen (Endverbrauchern). Meistens via Auktions- oder Tauschportale.

Change Manager

→ Rollen im Testprozess

deu. -

Verwaltet Änderungen, d.h. Änderungen am System oder Änderungen an der Softwarezusammensetzung (Release). Diese

Zwei Aufgaben können in zwei Rollen zerfallen, den „Configuration-“ und den „Releasemanager“

Cluster, siehe **Test Cluster**

CM, siehe **Content Management**

Configuration, siehe **Konfigurierung**

Configuration Management, siehe **Konfigurationsmanagement**

Configuration Manager, siehe **Change Manager**

Consumer to Consumer, siehe **C2C**

Contention Test

→ RUP – Performance Testing

deu. -

Verifiziert, das ein System resource eine adequate Anzahl von Request abarbeiten kann. z.B. 30 DB Zugriffe in 10 Sekunden.

Content Management

→ Entwicklung

deu. Inhaltsverwaltung

CM dient zur Verwaltung und Bereitstellung von Inhalten. Es unterstützt bei der Erzeugung von Inhalten, hilft, die Inhalte in das entsprechende Format (HTML, PDF etc.) zu bringen, führt durch den Editing-Prozess. Durch einen implementierten Redaktionsprozess gelangt der neue Inhalt vom Autor über den Redaktor auf die Web-Seite.

CPM, siehe **Kritischer Pfad**

CRM

→ Testprozess

eng. Customer Relationship Management

deu. Kundenbeziehungsmanagement (IBM)

Bezeichnet sowohl die CRM Strategie, die die Pflege optimaler Kundenbeziehungen zum Ziel hat, als auch die (Software-)Lösung, mit der dieses Ziel erreicht werden soll.

Debugging

→ allgemein

deu. Fehlerbehebung, Fehlerkorrektur

Feststellen bzw. einkreisen eines Fehlers und dessen Behebung. Diese Tätigkeit wird vom Entwickler durchgeführt. Die Fehlerbehebung ist nicht Aufgabe des Testers.

Deliverable, siehe **Artefakt**

Design, siehe **Entwurf**

Developer, siehe **Entwickler**

Dienstleistung

→ ISO
eng. services
Immaterielles Ergebnis eines oder mehrerer (Teil-)Prozesse, welches einen Kontakt mit Kunden beinhaltet (ISO 9000-2000, Konzepte und Vokabular).

DNS, siehe **Domain Name Service**

Domain Name Service (DNS)

→ allgemein
deu. -
Dienst zur Umsetzung von Klartext Adressen (www.guideline.ch) in IP-Adressen (1xx.xxx.xxx.xxx).

Downstream

→ allgemein
deu. Abwärts-Strom
Übertragungsrichtung aus dem Internet zum PC.

Dynamischer Test

→ allgemein
eng. Dynamic Test
Werden zur Programmlaufzeit mit dem Ziel durchgeführt, ein konkretes Fehlverhalten des Programms zu provozieren und damit den fehlerhaften Code zu lokalisieren.

E2E

→ Systemtest
eng. End to End
deu. Schlüssel Kennzahlen
Das Testen von Funktionen (Business/Datenbanken) von der Frontend/GUI Eingabe über die Verarbeitung bis zur erwarteten Ausgabe.

Einstufung

→ ISO/IEC 9126
eng. rating
Die Abbildung eines Messwertes auf den entsprechenden Gütebereich. Damit wird der Gütebereich der Software für ein ausgewähltes Qualitätsmerkmal bestimmt.

Einwahlpunkt, siehe **Point of Presence**

End to End Test, siehe **E2E**

Entwickler

→ Rollen im Testprozess
eng. Developer
Mitglied des Entwicklungsteams. Führt Unit und Pult Test durch.

Entwicklungsumgebung

→ Rollen im Testprozess
eng. development or engineering environment
Hardware, Software und organisatorische Massnahmen (Vorgehensweise, Konzepte, Methoden), die bei der Systementwicklung in einem Projekt verwendet werden.

Entwurf

→ IEEE
eng. design
Einerseits der Prozess, bei dem die Architektur, Komponenten, Module, Schnittstellen, Prüfverfahren und Daten eines (Software-) Systems so definiert werden, dass sie festgelegte Anforderungen erfüllen. Andererseits das Ergebnis des Entwurfsprozesses.

ETSI

→ Organisationen
eng. -
European Telecommunication Standards Institute, siehe www.etsi.org.

European Telecommunication Standards Institute, siehe **ETSI**

Evaluation

→ Projekt Management
eng. evaluation
Prüfung und Bewertung eines Systems anhand der IT-Sicherheitskriterien. Die Evaluationsstellen können neben der Evaluierung eines fertigen Produkts auch begleitende Evaluationen durchführen, d.h. die einzelnen Prüfschritte werden während des Entwicklungsprozesses durchgeführt, sobald die entsprechenden Zwischenergebnisse den Zustand „akzeptiert“ haben.

Evaluationsbericht

→ Projekt Management
eng. evaluation report
Das Dokument, in dem das Ergebnis der Evaluation des IT-Systems und die daraus resultierende Bewertung nach Einzelaspekten aufgegliedert niedergelegt ist.

Evaluationshandbuch

→ Projekt Management
eng. evaluation manual
Richtlinien zur Vorgehensweise bei der Evaluation nach den IT-Sicherheitskriterien. Die CIP-Titelaufnahme der Deutschen Bibliothek für das Evaluationshandbuch lautet: IT-Evaluationshandbuch: Handbuch für die Prüfung der Sicherheit von Systemen der Informationstechnik (IT), herausgegeben von der ZSI-Zentralstelle für Sicherheit in der Informationstechnik im Auftrag der

Bundesregierung. 1. Fassung vom 22. Februar 1990. Köln: Bundesanzeiger, 1990.

Fehler

→ allgemein

eng. error, fault, bug

Ein Fehler liegt vor, wenn die tatsächlichen Eigenschaften einer Funktionseinheit von dem als korrekt festgelegten oder in einer Spezifikation geforderten Verhalten oder Zustand abweichen. Bemerkung: Im englischen Sprachgebrauch bezeichnet *fault* (auch *bug*) das Vorhandensein eines Fehlers in einer Funktionseinheit, durch den das Auftreten fehlerhafter Ergebnisse (*error*) verursacht werden kann.

Fehlerpriorisierung, siehe **Priorität**

FireWire®, siehe **IEEE1394**

Funktionalität

→ ISO 9126

eng. functionality

Eine Menge von Eigenschaften, die sich auf das Vorhandensein eines Satzes von Funktionen und deren festgelegte Eigenschaften auswirken. Die Funktionen sind jene, die die festgelegten oder vorausgesetzten Erfordernisse erfüllen. Diese Menge von Eigenschaften charakterisiert, was eine Funktionseinheit zur Erfüllung von Erfordernissen tut, während die anderen Qualitätsmerkmale hauptsächlich charakterisieren, wann und wie sie das tut.

Glas Box Test, siehe **White Box Test**

GUI

→ allgemein

eng. graphic user interface

deu. Benutzeroberfläche

Eingabe und Bildschirmmasken einer Applikation.

HTML

→ allgemein

eng. Hyper Text Markup Language

Seitenbeschreibungssprache für die Erstellung einer Seite im Internet (WWW). Durch die laufende Erweiterung gibt es mehrere Standardversionen.

IEEE

→ IEEE

eng. The Institute of Electrical and Electronics Engineers inc.

Vereinigung für Standards im Elektronik Bereich. Weltweit aufgeteilt in zehn Regionen. Standards, Mitgliedschaft und Informationen siehe www.ieee.org.

IEEE1394

→ IEEE

eng. FireWire®

Industrie Standard für FireWire®, High Performance Serial Bus. (ausbauen!)

International Telecommunication Union, siehe **ITU**

ISO

→ allgemein

eng. International Organization for Standardization

1947 gegründete Standard Organisation mit 140 Mitgliedsländern. Siehe auch www.iso.org.

ISO13406-2

→ ISO Standard

eng. -

Norm für Flachdisplays. Unter anderem Prüfung in Pixelfehlerklassen und Blickwinkelklasse.

ISO8402

→ ISO Standard

eng. -

Norm für Qualitätsmanagement Begriffe.

ITU

→ Organisationen

eng. International Telecommunication Union

International Telecommunication Union. Siehe auch www.itu.int.

Konfigurationsmanagement

→ V-Modell

eng. configuration management

Submodell, in dem die Verwaltung der während der Systementwicklung anfallenden Produkte geregelt wird. Ihre verschiedenen Versionen und Beziehungen zueinander werden konsistent gehalten, Änderungen überwacht und Zugriffe kontrolliert.

Konfigurierung

→ V-Modell

eng. configuration

Wahl einer der für ein IT-System möglichen Ausprägungen (hard- und softwareseitig).

Konformität

→ Testprozess

eng. -

Erfüllung von vorgegebenen Anforderungen. Das Konformitätszertifikat ist ein Dokument, welches nach den Regeln des Zertifizierungssystems ausgestellt wird und Sicherheit liefert, dass ein ordnungsgemäß identifiziertes Produkt oder eine Dienstleistung, ein Prozess oder ein Qualitätsmanagement-

System bestimmte Standards erfüllt (ISO 9000-2000, Konzepte und Vokabular).

KPI

→ Testprozess

eng. Key Performance Indicator

deu. Schlüssel Kennzahlen

Kennzahlen, die wichtige Daten liefern, die für die Messung des Testfortschritts, der Testabdeckung wichtig sind.

Kritischer Pfad

→ Testprozess

eng. critical path

Weg auf dem die Abhängigkeiten der Einzelnen Lieferobjekte keine Verzögerung zulassen, d.h. sequentiell abgearbeitet werden müssen. Toleranz = 0. Beruhend auf der CPM (critical path method) für die Zeitplanung.

Kunden (interne, externe)

→ allgemein

eng. client, customer

Im Zentrum des Qualitätsmanagement steht die optimale Erfüllung von Kundenbedürfnissen. Neben dem eigentlichen Kunden einer Unternehmung (externe Kunden) sind auch die unternehmensinternen Empfänger von Arbeitsergebnissen Kunden (interne Kunden). Jedes Arbeitsergebnis hat einen Abnehmer oder Kunden. Jeder Mitarbeiter muss seine internen und externen Kunden und deren Anforderungen an seine Arbeitsergebnisse (Produkte, Dienstleistungen) kennen und sich mit ihnen über die optimale Erfüllung deren Kundenbedürfnisse auseinandersetzen.

Kundenbedürfnisse

→ Testprozess

eng. -

(Kunden-)Bedürfnisse bezeichnen allgemein einen Zustand des Mangels oder des Fehlens von etwas, dessen Behebung (Befriedigung) verlangt wird. Bedürfnisse können sich im Laufe der Zeit verändern, aber auch von Situation zu Situation; es werden neue geweckt, bisherige verschwinden. Bedürfnisse haben von Person zu Person unterschiedliche Bedeutung. Sie bilden eine Art Hierarchie, die wiederum von Person zu Person, aber auch von Situation zu Situation und von Zeit zu Zeit sich ändern kann. Bedürfnisse können in bewusste und nicht bewusste Bedürfnisse unterschieden werden. Aus diesem Grunde ist es nicht immer einfach, Bedürfnisse in Erfahrung zu bringen. Wenn Kundenanforderungen erfüllt sind, besteht Kundenzufriedenheit (ISO).

Kundenorientierung

→ Testprozess

eng. -

Über die Qualität von Produkten und Dienstleistungen entscheiden die Kunden. Kundenloyalität, Kundenbindung und Marktanteil werden am besten durch eine klare Ausrichtung auf die Bedürfnisse von bestehenden und potenziellen Kunden optimiert. Unter Kundenorientierung wird die Ausrichtung sämtlicher Tätigkeiten und Abläufe eines Unternehmens auf die Anforderungen und Erwartungen seiner Kunden verstanden. Grundlage bildet die Einbeziehung der Kundensicht in das Verständnis von Qualität (ISO).

Kundenzufriedenheit

→ Testprozess

eng. -

Die optimale Befriedigung der Bedürfnisse und Erwartungen der Kunden; wenn die Kundenanforderungen erfüllt sind, besteht Kundenzufriedenheit (ISO).

Leistungserstellungskette

→ ISO

eng. Supply Chain Management

Die Herstellung von Produkten und die Erbringung von Dienstleistungen durch ein Unternehmen bestehen aus einer Abfolge von verschiedenen Tätigkeiten, die durch verschiedene Leistungsersteller erbracht werden. Jeder dieser Leistungsersteller ist dabei zugleich auch -empfänger (interner Kunde), da er die Ergebnisse der Arbeiten eines vorgängigen Leistungserstellers bezieht. Damit eine möglichst optimale Qualität in diesem Ablauf resultiert, ist es notwendig, dass sich Leistungsersteller und -empfänger über die Anforderungen und die erbrachten Leistungen unterhalten und die Tätigkeiten entsprechend abstimmen. Am Anfang und Ende einer Leistungserstellungskette steht letztendlich immer der Kunde des Unternehmens.

Lessons Learned

→ allgemein

deu. Erfahrungsberichte

Gemachte Erfahrungen werden in Form von Berichten, Intra- oder Internetpublikation oder in einer Datenbank zur Verfügung gestellt. Andere sollen von den Erfahrungen profitieren können.

Life Cycle

→ allgemein

deu. Lebenszyklus

Der Zeitraum beginnend bei der Idee zu einem Software-Produkt und endend, wenn das Produkt aus der Nutzung genommen wird. Der Software Life Cycle wird zeitlich in einzelne aufeinanderfolgende Phasen unterteilt.

Load Testing

→ RUP – Performance Testing

deu. -

Bei gleichbleibender Konfiguration des System wird das Belastungsprofil (Anzahl Benutzer, Anzahl Transaktionen) verändert, um zu sehen, ob Veränderungen eintreten.

Maximum Segment Size

→ allgemein

deu. -

Grösste Nutzdatenmenge, die in ein TCP-Paket passt.

Maximum Transfer Unit

→ allgemein

deu. -

Grösstes Datenpaket, das das medium transportieren kann (bei PPPoE 1492 Bytes).

Modul

→ allgemein

eng. module

Zu programmierender Software-Baustein, der nicht weiter zerlegt werden soll. Ein Modul wird anhand von Kriterien wie Abgeschlossenheit, Geheimnisprinzip, Datenabstraktion, Kapselung, Schnittstellenspezifikation, Schnittstellenminimalität, Überschaubarkeit, Testbarkeit gebildet.

MSS, siehe **Maximum Segment Size**

MTU, siehe **Maximum Transfer Unit**

Normen

→ ISO

eng. norm, standard

Anerkannte, als verbindlich geltende Regeln, die gleichzeitig auch einen Massstab zur Bestimmung der Einhaltung vorgeben. In den 80er Jahren des vorletzten Jahrhunderts wurden erstmals Qualitätsnormen für Managementsysteme definiert und international standardisiert.

Objektivität

→ allgemein

eng. objectivity

Eine Prüfung wird objektiv durchgeführt, wenn das Ergebnis nachweislich auf Tatsachen beruht und nicht durch Vorurteile, Meinungen oder Gefühle der Prüfer gefärbt ist und nicht durch Interessen anderer beeinflusst wird. Dieser Begriff ist in allen Arten von Prüfungsvorgängen (Sicherheitsbewertung, QS-Prüfungen, Konformitätsnachweis usw.) von Bedeutung. Unparteilichkeit kommt dem Begriff Objektivität sehr nahe.

Performance Profiling

→ RUP – Performance Testing

deu. -

Bei gleichbleibender Belastung wird die Konfiguration des System verändert, um zu sehen, ob Verbesserungen eintreten.

Point of Presence

→ allgemein

deu. Einwahlpunkt

Einwahlpunkt eines providers, prüft die Zugangsdaten und leitet den Datenverkehr weiter.

Point to Point Protocol (PPP)

→ allgemein

deu. -

Zugangsprotokoll der meisten DSL-Provider, definiert in RFC 2516.

POP, siehe **Point of Presence**

Power User

→ Rollen im Testprozess

deu. Benutzervertreter

Vom Auftraggeber der Entwicklung oder vom Applikationsinhaber bestimmter Benutzer, der die Interessen des Auftraggebers wahr nimmt. Führt die funktionalen Tests der Anwendung durch. Hat tiefe Kenntnisse der Anwendung und des Geschäftsfeldes, für das die Anwendung bestimmt ist. Kann auch während den Unit Tests zur Verifikation einer Lösung/GUI hinzugezogen werden.

PPP, siehe **Point to Point Protocol**

Priorität

→ Defect Tracking

eng. priority

Legt die Geschwindigkeit der Behebung fest. z.B. 1,2,3,4 oder hoch, mittel, tief. Siehe auch Severity

Problem Manager

→ Rollen im Testprozess

deu./eng.

Entweder mit den Aufgaben des Configuration Manager und/oder des Release Managers, sowie mit Problembehebung betraut, oder ausschliesslich mit der Problembehebung betraut. Siehe **Change Manager**.

Produkt

→ ISO

eng. product

Ergebnis eines oder mehrerer (Teil-)Prozesse. Produkte werden gegenüber Dienstleistungen als materielle Ergebnisse beschrieben, Dienstleistungen hingegen sind immaterielle Ergebnisse eines oder mehrerer (Teil-

)Prozesse. ISO 9000-2000 definiert Produkte sowohl als materielle als auch immaterielle Ergebnisse von Prozessen (ISO 9000-2000, Konzepte und Vokabular).

Programmierer siehe **Entwickler**

Projektleiter

→ Rollen im Testprozess

eng. project manager

Vom Auftraggeber mit der Durchführung eines Projekts im Zeit und Budgetrahmen beauftragt.

Projekt Manager, siehe **Projektleiter**

Prozess

→ ISO

eng. process

Ein Prozess ist ein Arbeitsablauf, der als Bündel von Aktivitäten aufgrund von Inputs für Kunden ein immaterielles oder materielles Ergebnis von Wert (Output, Leistung) erzeugt. Umfassende Geschäftsprozesse lassen sich in operative Teilprozesse mit internen oder externen Kunden unterteilen. Es können verschiedene Arten von Prozessen unterschieden werden: Geschäftsprozesse, Führungsprozesse, Supportprozesse usw.

Prozessbeherrschung

→ ISO

eng. -

Die Merkmale eines Prozesses ändern praktisch nicht oder nur in beabsichtigter Weise und in definierten Grenzen. Dies bedeutet, ein Prozess ist so stabil, dass die von ihm erzeugten Produkte über die Zeit fehlerfrei bleiben. Die Sorgfalt der Planung, die umfassende Ermittlung der Einflussgrößen und die Sicherung der Prozessfähigkeit stellen unverzichtbare Voraussetzungen dar. Das eigentliche Ziel des Prozessmanagement ist der beherrschte Prozess. Im Qualitätsmanagement wird zusätzlich die ständige Verbesserung und Weiterentwicklung der Prozesse angestrebt.

Prozessmanagement

→ ISO

eng. process management

Prozessmanagement umfasst die zielgerichtete Planung, Organisation und Kontrolle der Arbeitsabläufe eines Unternehmens hinsichtlich Qualität, Zeit, Kosten und Kundenzufriedenheit. Dabei erfolgt die Aufgabenteilung im gesamten Unternehmen nach einer durch Leistungserstellungsketten vorgegebenen Prozessnotwendigkeit. Prozessmanagement-Konzepte gibt es seit Anfang der 80er Jahre des letzten Jahrhunderts.

Prozessorientierung

→ ISO

eng. process oriented

Wie ein Unternehmen die Prozesse identifiziert, überprüft und ändert um seine Geschäftstätigkeit ständig zu verbessern und erfolgreicher zu gestalten. In jedem Unternehmen gibt es ein Netzwerk von Prozessen, die geführt und verbessert werden. Unter den Prozessen gibt es einige, die für den Unternehmenserfolg ausschlaggebend sind. Diese Prozesse müssen identifiziert werden; sie überschreiten Abteilungs- und Funktionsgrenzen und erfordern besondere Beachtung.

Prozessprüfung

→ allgemein

eng. process assessment

Qualitätsprüfung an einem Prozess bzw. an einer Tätigkeit anhand der Merkmale des Prozesses bzw. der Tätigkeit selbst. Prozessprüfungen dienen unter anderem der Verfahrensüberwachung. Der Prozess in dem hier verwendeten Sinne unterscheidet sich von dem „Software-Prozess“ innerhalb der operativen Software.

Prozessverantwortlicher

→ allgemein

eng. process responsible

Person, welche für einen bestimmten Prozess verantwortlich ist. Seine Aufgaben sind die Koordination aller Massnahmen und der beteiligten Bereiche bei der Einführung und dem Betrieb eines Prozesses, die Planung und Überwachung eines Prozesses, die Einleitung von Verbesserungsmaßnahmen, die Bestimmung der Einflussgrößen und die Berichterstattung.

Prüfkriterien

→ V-Modell

eng. assessment criteria

Prüfkriterien sind Fragestellungen, die durch eine Prüfung geklärt werden sollen. Bei der Prüfung von Papierprodukten sollten als Prüfkriterien Checklisten verwendet werden.

Prüfplan

→ V-Modell 97

eng. test plan, assessment plan

Der Prüfplan definiert organisatorische Regelungen zum Ablauf der Prüfung, nämlich zeitliche und personelle Planung sowie die für die Durchführung erforderliche Umgebung (Werkzeuge, Rechner, etc.). Im Prüfplan ist festgelegt, welche Produkte und Aktivitäten in welchem Zustand wann, von wem und womit zu prüfen sind. Der Testplan ist eine spezielle Form

des Prüfplans, bei dem die gewählte Prüfmethode der Test ist.

Prüfprotokoll

→ V-Modell

eng. assessment report, test report

Das Prüfprotokoll enthält die vom Prüfer verfassten Aufzeichnungen über den Verlauf der Prüfung, vor allem die Gegenüberstellung von erwartetem und erzieltm Ergebnis. Die Aussagen im Prüfprotokoll können im Rahmen der Auswertung um Analysen über mögliche Ursachen des Fehlverhaltens und um vorgeschlagene Massnahmen ergänzt werden.

Prüfschärfe

→ V-Modell

eng. assessment criteria

Die Prüfschärfe ist ein Mass für die Gründlichkeit einer Prüfung. Sie wird bestimmt vom Umfang der Prüfaufgaben laut Prüfspezifikation sowie von der Genauigkeit, mit der die Prüfergebnisse analysiert und bewertet werden.

Pulttest

→ Unittest

eng. Desk Test

Am Arbeitstisch, vom Entwickler selber durchgeführte Codedurchsicht oder Ausführen von Programmen oder Programmteilen. Ist für gute Entwickler selbstverständlich.

Qualität

→ ISO

eng. quality

Qualität ist die optimale Erfüllung der Bedürfnisse interner und externer Kunden, das Vermeiden von Fehlern, die effiziente Leistungserstellung und das Bestreben, ständig besser zu werden. Qualität heisst nicht «übertriebenes Streben nach Perfektion».

Qualitätsansprüche

→ ISO

eng. -

Um den Bedürfnissen der Kunden gerecht zu werden, werden an Produkte, Dienstleistungen und Prozesse entsprechende Anforderungen gestellt. Die Qualität von Produkten und Dienstleistungen ist sowohl an intern vorgegebene als auch vom Kunden verlangte Erfordernisse gebunden. Die Kundenforderungen drücken sich in Qualitätsansprüchen aus; wenn diese erfüllt werden, entsteht Kundenzufriedenheit (ISO 9000-2000, Konzepte und Vokabular).

Qualitätsaudit

→ ISO

eng. quality audit

Beurteilung der Wirksamkeit des Qualitätssicherungssystems oder seiner Elemente durch eine unabhängige systematische Untersuchung. Die Beurteilung umfasst die Prüfung, inwieweit die einschlägigen QS-Verfahrensanweisungen befolgt werden und ob diese zweckmässig sind. Gegebenenfalls folgen daraus Hinweise auf nötige Verbesserungen und Korrekturen.

Es wird unterschieden zwischen

- Systemaudit
- Verfahrensaudit und
- Produktaudit.

Von Verfahrensaudit oder von Produktaudit wird gesprochen, wenn die Wirksamkeit von Elementen des Qualitätssicherungssystems anhand von Verfahren oder Produkten beurteilt wird. Von Systemaudit spricht man, wenn die Wirksamkeit des Qualitätssicherungssystems als Ganzes beurteilt wird. Qualitätsaudits werden - je nach Veranlassung durch die Leitung der Organisation oder durch Vertragspartner/Dritte - für interne oder für externe Zwecke durchgeführt (Internes Qualitätsaudit/Externes Qualitätsaudit).

Qualitätsgrösse

→ ISO 8402, DIN 55350

eng. -

Eigenschaft eines Produkts, die unmittelbar messbar ist (siehe Qualitätsmass) und zur Beurteilung von Qualitätsmerkmalen geeignet ist (z. B. lines of code).

Qualitätskosten

→ ISO 8402, DIN 55350

eng. quality costs

Kosten, die vorwiegend durch Qualitätsanforderungen verursacht sind, das heisst: Kosten, die durch Tätigkeiten der Fehlerverhütung (Fehlerverhütungskosten) und durch planmässige Qualitätsprüfungen (Prüfkosten) verursacht sind. Üblicherweise werden auch die Kosten aufgrund intern oder extern festgestellter Fehler (Fehlerkosten) zu den Qualitätskosten gezählt. Diese wurden hier bewusst aus der Definition genommen.

Qualitätslenkung

→ ISO 8402, DIN 55350

eng. quality management

Die vorbeugenden, überwachenden und korrigierenden Techniken und Tätigkeiten bei der Realisierung eines Produkts mit dem Ziel, die Qualitätsanforderungen zu erfüllen.

Qualitätsmanagement

→ ISO

eng. quality management

Das Qualitätsmanagement beinhaltet alle Tätigkeiten einer Organisation, welche die Qualitätspolitik, die Qualitätsziele und die entsprechenden Verantwortungen umfassen. Qualitätspolitik und -ziele werden durch Qualitätsplanung, Qualitätslenkung, Qualitätssicherung und Qualitätsverbesserung verwirklicht, mit dem generellen Ziel, die Kundenzufriedenheit und den langfristigen Erfolg einer Organisation zu sichern und zu verbessern. Häufig entsteht Verwirrung im deutschen Sprachraum, da «Qualitätssicherung» ebenfalls als umfassende Benennung von «Qualitätsmanagement» verwendet wird; Qualitätssicherung ist ein wichtiger Teilaspekt des Qualitätsmanagement, der sich auf die Erfüllung der Qualitätsanforderungen konzentriert.

Qualitätsmanagement-Handbuch

→ ISO

eng. quality management guideline

Beinhaltet die Beschreibung eines Qualitätsmanagement-Systems und gibt die grundsätzliche Einstellung des Management sowie ihre Absichten und Massnahmen zur Sicherung und Weiterentwicklung der Qualität des Unternehmens wieder.

Qualitätsmanagement-System

→ ISO

eng/deu.

Das Qualitätsmanagement-System ist Teil eines Managementsystems, welcher das Erstellen und Erfüllen der Qualitätspolitik und Qualitätsziele zum Ziel hat und umfasst die zur Verwirklichung erforderliche Organisationsstruktur, die notwendigen Verfahren, Prozesse und Mittel (ISO 9000-2000, Konzepte und Vokabular).

Qualitätsmasse

→ V-Modell

eng. quality metric

Messbare Größen, die Rückschlüsse auf die Ausprägung bestimmter Qualitätsmerkmale gestatten. (Beispiele: Antwortzeit für Funktion ... ist kleiner als ... Sekunden, Länge des Moduls ... ist kleiner als ... Seiten)

Qualitätsmerkmale

→ ISO

eng. ?

Ein Merkmal ist eine Eigenschaft, die das Unterscheiden und Erkennen von Einheiten ermöglicht. Wenn ein solches Merkmal die Qualität bestimmt, handelt es sich um ein Qualitätsmerkmal oder Qualitätskriterium.

Qualitätsnachweis

→ ISO

eng. ?

Produktbezogene Qualitätsaufzeichnung, die als Nachweis dafür dient, dass die Qualitätsanforderung an ein materielles, ein immaterielles oder ein kombiniertes Angebotsprodukt erfüllt ist (Deutsche Gesellschaft für Qualität, 1995).

Qualitätsplanung

→ ISO

eng. quality planing

Teil des Qualitätsmanagement-Systems, welcher sich mit dem Aufstellen und Auswerten der Qualitätspolitik, den Qualitätszielen und den Qualitätsanforderungen befasst. Die Qualitätsplanung beinhaltet die vorbeugende Fehlerverhütung durch Analyse und Planung.

Qualitätspolitik

→ ISO

eng. quality policy

Die Qualitätspolitik ist Teil der Unternehmenspolitik. Sie umfasst alle Absichten und Anordnungen einer Unternehmung bezogen auf die Erfüllung der Qualitätsanforderungen (Qualitätsziele). Die Qualitätspolitik wird durch die Qualitätsplanung, die Qualitätslenkung, durch das Qualitätsmanagement-System und durch Qualitätsverbesserungen in die Praxis umgesetzt.

Qualitätsprinzipien

→ ISO

eng. ?.

Die acht Prinzipien sind: (1) Kundenorientierte Organisation, (2) Aktivitäten der Führung, (3) Beteiligung der Mitarbeiter, (4) Vorgehen mittels Prozessen, (5) systematisches Vorgehen, (6) ständige Verbesserung, (7) sachliches Vorgehen bei Beschlussfassung, (8) für beide Seiten vorteilhafte Lieferantenbeziehungen. Diese Prinzipien mögen als selbstverständlich erscheinen, die systematische Umsetzung in die Praxis ist jedoch für alle Verantwortlichen eine tägliche Herausforderung.

Qualitätsprüfung

→ ISO

eng. ?.

Feststellen, inwieweit eine Einheit vorgegebene Qualitätsanforderungen erfüllt. Betrachtungseinheit deckt auch Prozesse ab.

Qualitätssicherung

→ DIN ISO 8402

eng. quality assurance

Alle geplanten und systematischen Tätigkeiten, die innerhalb des Qualitätsmanagements verwirklicht sind und die wie erforderlich dargelegt werden, um angemessenes Vertrauen

zu schaffen, dass eine Einheit die Qualitätsforderung erfüllen wird.

Qualitätssicherungs-Plan

→ DIN 55350

eng. quality plan

Dokument, welches die speziellen Elemente der Qualitätssicherung sowie die Zuständigkeiten, sachlichen Mittel und Tätigkeiten festlegt, die für ein Projekt vorgesehen sind.

Qualitätsstufe

→ allgemein

eng. confidence level

Bezeichnet die Zusammenfassung einer Reihe von Sicherheits-Bewertungskriterien, die ein System erfüllen muss, um diese Kennzeichnung tragen zu dürfen. Die Qualitätsstufen bauen insofern aufeinander auf, als die Bewertungskriterien einer „niederen“ Stufe in denjenigen einer „höheren“ mit enthalten sind.

Qualitätswesen

→ DGQ

eng. quality department

Aufbauorganisatorische Einheit, die sich vorwiegend mit Qualitätssicherung befasst.

Qualitätszertifikat

→ ISO

eng. ?

Verwandte Begriffe: Qualitätssystemzertifikat, Qualitätsmanagement-Systemzertifikat. Eine dritte Partei gibt eine schriftliche Zusicherung, dass ein Produkt oder eine Dienstleistung, ein Prozess oder ein Qualitätsmanagement-System vorgegebene Anforderungen erfüllt. Bei Erfüllung der Anforderungen wird ein Qualitätszertifikat ausgestellt (ISO 9000-2000, Konzepte und Vokabular).

Qualitätsziele

→ ISO

eng. ?.

Im Rahmen der Qualitätspolitik zu erreichender Zustand eines Unternehmens (ISO 9000-2000, Konzepte und Vokabular).

Rating, siehe Einstufung

Regressionstest

→ Testprozess, V-Modell 97

eng. regression test

deu. Wiederholtest

Wiederholung eines bereits vollständig durchgeführten Tests, z. B. aufgrund von Pflege, Änderung und Korrektur, wobei zur Vereinfachung der Testauswertung die Ergebnisse des Regressionstests mit den

Ergebnissen des vorausgegangenen Tests verglichen werden.

Release Manager, siehe Change Manager

Report, siehe Prüfprotokoll

Review

→ ISO

eng/deu.

Die Ergebnisse einer Organisationseinheit, die Arbeiten eines Projektes usw. werden systematisch bestimmt und bewertet, um zu beurteilen, ob die vorgegebenen Ziele erreicht werden.

RUP

→ Testprozess

eng/deu.

RATIONAL© UNIFIED PROCESS, Prozess Frame Work von Rational zur strukturierten Durchführung von Entwicklungsprojekten. Ist anpassbar und in mehreren Sprachen verfügbar.

SCM, siehe Supply Chain Management oder Leistungserstellungskette

Security requirement, siehe Sicherheitsanforderung

Severity

→ Defect Tracking

deu. Stärke, Heftigkeit

Neben der Fehlerpriorisierung in 1,2,3,4 oder hoch, mittel, tief, kann auch noch eine zweite Skalierung vorgenommen werden, um die Auswirkung des Fehler auszudrücken.

Sicherheitsanforderung

→ Testprozess

eng. security requirement

Sicherheitsanforderungen ergeben sich nach der Durchführung der Risikoanalyse daraus, dass gegen die Bedrohungen, von denen unannehmbar hohe Risiken ausgehen, etwas unternommen werden muss.

Sicherheitspolitik

→ V-Modell

eng. (system) security policy

Eine Sicherheitspolitik beschreibt grundlegende sicherheitsbezogene Ziele einer Organisation sowie Regeln und Randbedingungen, mit denen den als bedeutsam angesehenen allgemeinen Bedrohungen ihrer Systeme entgegengewirkt wird. Aus der Sicherheitspolitik lassen sich Sicherheitsanforderungen ableiten. Die Übersetzung von system security policy ins Deutsche mit Sicherheitspolitik ist unzutreffend,

hat sich aber so eingebürgert. Es wird empfohlen, den deutschen Ausdruck in dieser Bedeutung nicht zu verwenden, um Verwechslungen mit der nationalen Sicherheitspolitik der Bundesregierung zu vermeiden. Massnahmen können Zugangskontrollen, Passwörter, Schulungen etc. sein.

Smoke Test

→ Testkonzept
deu. Schnelltest

Ein Release wird angeliefert, installiert und schnell und formlos "angetestet". Wird "Rauch" gefunden, geht man dem Problem nach und weisst die Lieferung unter Umständen zurück. Das System wird so nicht unnötig mit einem schlechten Produkt belegt, bzw. Ressourcen werden nicht verschwendet, für unnötige Tests.

Software

→ allgemein
eng. specification
Programme für Datenverarbeitungssysteme, die zusammen mit deren Eigenschaften zusätzliche Betriebsarten oder Anwendungsarten ermöglichen. Software umfasst die Computerprogramme mit den zugehörigen Daten und der Dokumentation.

Software Life Cycle, siehe Life Cycle

Spezifikation

→ Projekt Management
eng. specification
Ein Dokument, welches alle (Nutzer-) Anforderungen an das System oder an Teile des Systems, die vorgesehene Architektur, das geplante funktionale Verhalten oder andere (qualitative) Eigenschaften genau und nachvollziehbar beschreibt. Man unterscheidet zwischen Anforderungs- und Designspezifikation. Spezifikation wird auch synonym für die Aktivität zur Erstellung dieses Dokuments gebraucht. Unterschiedlichen Systemebenen entsprechen auch unterschiedliche Spezifikationsebenen, die einen unterschiedlichen Detaillierungsgrad aufweisen (z. B. Top-level-Spezifikation, Feinspezifikation).

Statische Testverfahren

→ allgemein
eng. -
Zum Beispiel Walkthrough, Inspektion oder Statistiken. Es werden keine Programme ausgeführt. Programmfehler sind dabei nur bedingt erkennbar. Die Einhaltung von Richtlinien wird überprüft.

Stress Testing

→ RUP – Performance Testing
deu. -
Bei gleichbleibender Konfiguration des System wird das Belastungsprofil (Anzahl Benutzer, Anzahl Transaktionen) annormal angehoben, um zu sehen, wann das System zusammenbricht.

Super User, siehe Power User

Supply Chain Management

→ allgemein
deu. -
Planung, Organisation und Durchführung aller Versorgungsprozesse entlang der vernetzten Wertschöpfungskette.

TAF, siehe Test Automation Framework

TCO

→ allgemein
eng. Total Cost of Ownership
Gesamtkosten der Installation, des Betriebs und der Pflege eines PC-Arbeitsplatzes (im Unternehmen).

TCP, siehe Transmission Control Protocol

Test

→ allgemein
eng. test
Methode zur dynamischen Prüfung eines Produkts mit dem Ziel, Abweichungen im erwarteten Verhalten des Produkts vom tatsächlichen Verhalten des Produkts festzustellen, so dass Korrekturmaßnahmen eingeleitet werden können.

Test, formeller

→ allgemein
eng. test, formal
Ein Test unter Obhut und den Vorschriften einer Instanz. Der Test hat die Eigenschaft der Nachvollziehbarkeit. Vorgaben, Eingaben und Ergebnisse müssen protokolliert sein. Siehe Informeller Test.

Test, informeller

→ allgemein
eng. test, informal
Test anhand dessen sich der Entwickler selbst davon überzeugt, dass das von ihm erstellte Produkt Anforderungen erfüllt. Siehe Formeller Test.

Testabdeckung

→ V-Modell
eng. test coverage
Wichtige Kennzahl zur Ermittlung der Tests gegenüber den Anforderungen. Kann bei Verwendung eines Testwerkzeugs real time

ermittelt werden. Ausmass der Testfälle, mit denen ein Programm oder Programmteil getestet wird. Man unterscheidet zwischen den Abdeckungen C0 bis C7, wobei die gebräuchlichsten Testabdeckungen C0, C1 und C2 sind.

C0 :Jede Anweisung mindestens einmal durchlaufen.

C1 :Jeder Programmzweig wird mindestens einmal durchlaufen (z. B. erfordert die bedingte Anweisung if....then...endif zwei Testfälle).

C2 :Bei zusammengesetzten Bedingungen in bedingten Anweisungen wird jede Bedingung (wahr, falsch) getestet.

Test Automation Framework (TAF)

→ allgemein

eng/deu.

Test Standard, der sich auf das Aufspüren von Requirement Fehlern und das Automatisieren von Tests konzentriert. <http://www.software.org>.

Testbett, siehe **Test Harness**

Test Cluster

→ Testprozess

deu. Testart

Zusammenfassung von Testfällen ähnlichen Typs, z.B. Tests Zahlungsverkehr, Tests Benutzeroberfläche, Tests Berechnungen usw. für die Durchführung und Auswertung der Tests.

Test Co-ordinator, siehe **Test Manager**

Testdaten

→ V-Modell 97

eng. test data

Eingabe zum Zweck der Prüfung eines Systems, eines Moduls oder einer Komponente. Darunter sind aber auch Daten zum Aufsetzen der Testumgebung, Simulationsdaten, und Daten aus der realen Umgebung (Stammdaten) zu verstehen. Testfälle gehören zu einem oder mehreren Testfällen. Es können unterschiedliche Testdatensätze für die einzelnen Teststufen/Testphasen benötigt werden.

Tester

→ Rollen im Testprozess

eng. tester

Beauftragt mit der Erstellung und Durchführung von Testfällen.

Test Executer, siehe **Tester**

Testfall

→ V-Modell 97

eng. test case

Eine spezifische Menge von Testdaten, zugehörigen Durchführungsanweisungen und Sollergebnissen, entwickelt für eine bestimmte Prüfabsicht.

Test Harness

→ Testprozess

deu. Testbett, Testtreiber

Softwaresimulation einer Systemumgebung bzw. deren Schnittstellen (Emulation Schnittstellen). Wird meist aus Kostengründen verwendet zum Beispiel um HOST Transfer zu einem Applikationsserver zu testen.

Test Manager

→ Rollen im Testprozess

eng. test manager

Meist leitet er das Teilprojekt Testing innerhalb eines Entwicklungsprojekts. Erstellung und Implementierung des Testprozesses.

Testplan, siehe **Prüfplan**

Testprozedur

→ Testprozess

eng. test procedure

Die Testprozedur ist eine spezielle Form der Prüfprozedur, bei der die gewählte Prüfmethode der Test ist.

Testreport, siehe **Prüfprotokoll**

Test Script Language (TSL)

→ Test Werkzeuge

deu. Testbett, Testtreiber

An C angelehnte Test Programmiersprache.

Test Tool Engineer

→ Rollen im Testprozess

deu. -

Tester mit Spezialkenntnissen für ein oder mehrere Testwerkzeuge. Kennt die Testmethoden.

Testware

→ allgemein

eng. testware

Sammelbezeichnung aller Elemente Software und Hardware für Test (ohne die zu Testende Software/Prüfling): Testdokumente, Testdaten, Testprogramme, Testumgebung, Simulatoren, Tester,

Tool Consultant, siehe **Test Tool Engineer**

Total Cost of Ownership, siehe **TCO**

Total Quality Management (TQM)

→ ISO

eng/deu.

Total Quality Management ist eine ganzheitliche qualitätsorientierte Unternehmensführung und -kultur, die weit über das traditionelle Konzept der Qualitätssicherung und -kontrolle hinausgeht und alle Organisationsprozesse als Mittel zur Erfüllung von Kundenanforderungen begreift.

TQM, siehe **Total Quality Management**

Transmission Control Protocol (TCP)

→ allgemein

deu. -

Internet-Protokoll, das die Übertragung jedes Pakets sicherstellt. Eingesetzt u.a. für HTTP, FTP, E-Mail.

Trouble Ticket

→ allgemein

deu. Fehlermeldung

Bei aufgegebenen Fehlern erstellt ein Supportmitarbeiter ein „Trouble Ticket“, d.h. der Fehler wird nachvollziehbar von der Aufgabe bis zur Schliessung. Eskalationswege und Lösungen werden genauso aufgenommen wie die benötigten Aufwände. Meistens verwenden Supportorganisationen eine Software, die die Tickets verwaltet. Trouble Tickets dürfen nicht mit dem Defect Tracking Prozess in einem Entwicklungsprojekt verwechselt werden.

TSL, siehe **Test Script Language**

TTCN-3

→ allgemein

deu. -

„Tree and Tabular Combined Notation, Edition 3“ vom European Telecommunication Standard Institute (ETSI). Eine Spezifikations- und Implementierungssprache für Softwaretests. Auch grafisches Format zur Erstellung von Testfällen mittels Sequenz Diagrammen (verschiedene Notationen). ITU Standard Z.140. Eingesetzt zum Testen von IPv6. Siehe www.etsi.org.

UDP, siehe **User Datagram Protocol**

Upstream

→ allgemein

deu. Aufwärtsstrom

Übertragungsrichtung vom PC ins Internet.

Usability, siehe **Benutzerfreundlichkeit**

User Datagram Protocol (UDP)

→ allgemein

deu. -

Ungesichertes Internet-Protokoll. Verbreitet bei Online-Spielen und Media-Streaming z.B. Quicktime, real-Player, Windows Streaming.

Validation, siehe **Validierung**

Validierung

→ ISO 8402

eng. validation

Bestätigen aufgrund einer Untersuchung und durch Führung eines Nachweises, dass die besonderen Forderungen für einen speziell vorgesehenen Gebrauch (Erwartungshaltung des Anwenders) erfüllt worden sind. Eine Validierung kann durchgeführt werden

- in Form einer Prüfung des Endproduktes gegen die Erwartung des Anwenders, - in Form einer Simulation auf der Basis von Zwischenergebnissen, bei der der Anwender einbezogen wird und -als Prüfschritt im Rahmen einer Durchführungsentscheidung am Ende einer Entwicklungsstufe bei inkrementeller Systementwicklung.

Verifikation

→ allgemein

eng. verification, proof of correctness

Nachweis der Korrektheit von Programmen mit formalen Mitteln (z. B. weakest precondition Kalkül) auf der Basis formaler Spezifikation.

Verifizierung

→ ISO 8402

eng. verification

Bestätigen aufgrund einer Untersuchung und durch Führung eines Nachweises, dass die festgelegten Forderungen erfüllt worden sind. Die Verifizierung kann als Prozess- oder Produktprüfung durchgeführt werden.

Verifikationsbedingung

→ allgemein

eng. verification condition

Bei der Anwendung von „Verifikationsmethoden“ erzeugte Formeln der Prädikatenlogik, deren Gültigkeit nachgewiesen werden muss.

Verlässlichkeit

→ allgemein

eng. dependability

Die Übereinstimmung zwischen erwarteter (spezifizierter) und erbrachter Leistung eines Dienstes oder Systems. Zusammenfassender Begriff für Aspekte der Sicherheit (security und safety), Zuverlässigkeit (reliability) und Verfügbarkeit (availability).

Version

→ V-Modell

eng. version

Unter „Versionen“ werden zeitlich nacheinander liegende Ausprägungen eines Produktes verstanden. Alle Produkte einer Konfiguration

(Software, Dokumente) besitzen eine individuelle Versionsangabe (Beispiele: a, b, c; 1.2, 1.3, 1.4). Jede formale oder inhaltliche Änderung an einem Konfigurationsprodukt bewirkt eine Erhöhung der Versionsangabe. Die Versionen müssen verwaltet und veröffentlicht werden, siehe Konfigurationsmanagement(?).

Vier Augen Prinzip

→ Unittest

eng. ?

Ich gebe meinen Code/Dokument einer anderen Person aus dem Team zur Kontrolle. Die Kontrolle ist formlos und das Ergebnis wird nicht rapportiert. Diese Massnahme erhöht die Qualität, die ein Entwickler Team abliefern kann, da die Barriere der eigenen Blindheit überwunden wird.

V-Modell

→ V-Modell

eng. lifecycle methodology

Regelungen, die die Gesamtheit aller Aktivitäten und Produkte und deren logische Abhängigkeiten bei der Entwicklung und Pflege/Änderung von Systemen, deren Aufgabenerfüllung vorwiegend durch den Einsatz von IT realisiert wird, im Bereich der Bundesverwaltung (Deutschland) festlegen.

VRML

→ Entwicklung

eng. Virtual Reality Modeling Language

VRML ist eine Szenenbeschreibungssprache zur Definition dreidimensionaler, interaktiver Web-Inhalte. Wurde 1997 ein ISO-Standard. Einsatz für 3D-Web Communities, 3D-Objekte und -Environments.

White Box Test

→ allgemein

deu. Glass-Box-Testing

Das Resultat des Test wird kontrolliert, spezielles Augenmerk gilt aber der Programmlogik. Wie das Resultat erzielt wird ist wichtig. Wegen der Komplexität wird dieser Test fast immer von Entwicklern gemacht. Das Gegenteil ist Black Box Testing.

UAT

→ UAT

eng. User Acceptance Test

deu. Benutzerakzeptanz Test

Der spätere Benutzer, bzw. delegierte Vertreter der Endbenutzer prüfen das Lieferobjekt gegen die Spezifikation. Hierfür werden Testfälle erarbeitet und durchgeführt. Fehler werden aufgenommen, priorisiert und dem Change Management Prozess zugeführt.

User Acceptance Test, siehe **UAT**

XML

→ Entwicklung

eng. Extended Markup Language

Ist eine standardisierte „Metasprache“ zur Definition individueller Seitenbeschreibungssprachen, mit denen strukturierte Dokumenteninformationen ausgezeichnet werden. XML ist wie HTML eine verknüpfungsorientierte Sprache für Web-Inhalte.

Zertifikation

→ V-Modell

eng. certification

Verwaltungsmassnahmen zur Erlangung der Zulassung einer Funktionseinheit durch eine unabhängige Organisation zu einem bestimmten Einsatzzweck.

Zertifizierungsstelle

→ allgemein

eng. certification body

Eine Organisation oder Institution, die von einer für Zertifizierungen zuständigen, nationalen oder internationalen Koordinierungsstelle zur Vergabe von Zertifikaten, z.B. für evaluierte Produkte, berechtigt wurde.

Zuverlässigkeit

→ ISO

eng. ?

Die Wahrscheinlichkeit, dass eine Einheit (Produkt, Dienstleistung) unter festgelegten Bedingungen während einer bestimmten Zeitdauer funktionsfähig und von unveränderter Qualität bleibt.

DRAFT

Ende

DRAFT