

Inhaltsverzeichnis

Teil I

Was versteht man unter automatisiertem Testen?

1.	Entstehung und Entwicklung des automatisierten Testens	3
1.1	Automatisiertes Testen	3
1.2	Hintergründe zum Testen von Software	5
1.3	Die Automated Test Life-Cycle Methodology (ATLM)	8
1.3.1	Die Entscheidung zum automatisierten Testen	12
1.3.2	Auswahl von Testwerkzeugen	13
1.3.3	Einführung des automatisierten Testens	13
1.3.4	Planung, Design und Entwicklung der Tests	14
1.3.5	Ausführung und Verwaltung von Tests	15
1.3.6	Prüfung und Beurteilung des Testprogramms	16
1.4	Die Rolle der ATLM beim Testen von Software	16
1.4.1	Der Zusammenhang zwischen der ATLM und dem Lebenszyklus der Systementwicklung.....	16
1.4.2	Test Maturity Model (TMM) – Erweiterung durch ein Reifegradmodell für das automatisierte Testen von Software	17
1.4.3	Entwicklung automatisierter Tests	23
1.4.4	Testaufwand	26
1.5	Karriereaussichten als Softwaretester	27
1.6	Zusammenfassung	31
1.7	Literaturhinweise	33
2.	Die Entscheidung zum automatisierten Testen	34
2.1	Ausräumen falscher Erwartungen an automatisierte Tests	37
2.1.1	Erstellen eines Plans für das automatisierte Testen	37
2.1.2	Das ultimative Testwerkzeug	38
2.1.3	Unmittelbare Verringerung des Testaufwands	39
2.1.4	Unmittelbare Verkürzung des Zeitplans	39
2.1.5	Benutzerfreundlichkeit des Testwerkzeugs	40
2.1.6	Universelle Anwendbarkeit der Testautomatisierung	40
2.1.7	100%-Tests	42
2.2	Vorteile des automatisierten Testens	43

XVI Inhaltsverzeichnis

2.2.1	Erstellung eines zuverlässigen Systems	44
2.2.2	Verbesserung der Testqualität	51
2.2.3	Verringerung des Testaufwands und Minimierung des Zeitplans	57
2.3	Unterstützung durch die Geschäftsführung	64
2.3.1	Vorschlagen eines Testwerkzeugs	66
2.4	Zusammenfassung	77
2.5	Literaturhinweise	78
3.	Bewertung und Auswahl von Testwerkzeugen	79
3.1	Die Systementwicklungsumgebung der Organisation	83
3.1.1	Externe Informationen von Geschäftsführung, Personal und Endbenutzern	83
3.1.2	Werkzeugkriterien vor dem Hintergrund der Systementwicklungsumgebung	85
3.1.3	Güte der Softwarequalität	86
3.1.4	Vorliegende Problembereiche	87
3.1.5	Einschränkungen im Budget	87
3.1.6	Testarten	87
3.1.7	Langfristige Aspekte der Investition	88
3.1.8	Einführung des Testwerkzeugs	88
3.1.9	Vermeiden von Mängeln	88
3.2	Werkzeuge zur Unterstützung des Testlebenszyklus	89
3.2.1	Werkzeuge zur Geschäftsanalyse	93
3.2.2	Werkzeuge zum Definieren der Anforderungen	95
3.2.3	Werkzeuge für Analyse und Design	97
3.2.4	Programmierwerkzeuge	99
3.2.5	Metrikwerkzeuge	101
3.2.6	Andere Werkzeuge für den Testlebenszyklus	102
3.2.7	Testwerkzeuge	102
3.3	Suche nach geeigneten Testwerkzeugen	105
3.3.1	Verbesserungsmöglichkeiten	106
3.4	Definition von Bewertungsbereichen	114
3.5	Bewertung von Werkzeugen in der Praxis	116
3.5.1	Bewertungsbericht	118
3.5.2	Lizenzvereinbarungen	120
3.6	Zusammenfassung	121
3.7	Literaturhinweise	122

Teil II

Einführung automatisierter Tests in ein Projekt

4.	Einführung des automatisierten Testens	125
4.1	Analyse des Testprozesses	128
4.1.1	Überprüfung des bestehenden Prozesses	130
4.1.2	Testziele	135
4.1.3	Teststrategien	141
4.2	Überlegungen zum Testwerkzeug	154
4.2.1	Überprüfung projektspezifischer Systemanforderungen	157
4.2.2	Überblick über die zu testende Anwendung	158
4.2.3	Überprüfung des Projektzeitplans	160
4.2.4	Überprüfung der Kompatibilität des Testwerkzeugs	161
4.2.5	Vorführung des Werkzeugs für das Projektteam	163
4.2.6	Unterstützungsprofil für ein Testwerkzeug	164
4.2.7	Überprüfung der Schulungsanforderungen	166
4.3	Zusammenfassung	166
4.4	Literaturhinweise	167
5.	Testteam-Management	169
5.1	Organisatorische Struktur eines Testteams	172
5.1.1	Durchgangs-Testteam	173
5.1.2	Zentrales Testteam	174
5.1.3	UVV-Testteam	176
5.1.4	Team für Systemmethodik und Tests	177
5.1.5	Zusammenfassung zu Testteams	178
5.2	Aufgaben im Rahmen des Testprogramms	181
5.3	Umfang des Testaufwands	189
5.3.1	Methoden zum Bestimmen der Testteamgröße	191
5.3.2	Die verhältnisorientierte Methode	192
5.3.3	Die Prozentmethode	193
5.3.4	Die Testverfahrenmethode	194
5.3.5	Die Aufgabenplanungsmethode	195
5.3.6	Faktoren mit Einfluss auf den Testaufwand	197
5.4	Einstellen von Testingenieuren	199
5.4.1	Qualitäten von Testingenieuren	200
5.4.2	Zusammensetzung des Testteams	202
5.4.3	Stellenausschreibung	205
5.4.4	Einstellungsaktivitäten	205
5.4.5	Finden von Testingenieuren	207
5.4.6	Gespräche mit Testingenieuren	208
5.4.7	Charakteristika des optimalen Bewerbers	210
5.5	Rollen und Verantwortlichkeiten	212
5.6	Zusammenfassung	217
5.7	Literaturhinweise	218

Teil III

Planung und Vorbereitung der Tests

6.	Testplanung: Intelligente Anwendung von Testverfahren	221
6.1	Testplanungsaktivitäten	223
6.2	Der Anwendungsbereich des Testprogramms	228
6.2.1	Systembeschreibung	230
6.2.2	Kritische und riskante Funktionen	230
6.2.3	Testziele und -strategien	230
6.2.4	Testwerkzeuge	231
6.2.5	Parameter des Testprogramms	231
6.2.6	Verifizierungsmethoden	234
6.2.7	Definition der Testanforderungen	235
6.3	Testanforderungsmanagement	237
6.3.1	Werkzeuge für das Anforderungsmanagement	238
6.3.2	Beurteilen der Risiken von Testanforderungen	241
6.3.3	Festlegen von Testprioritäten	241
6.3.4	Anforderungsverfolgbarkeitstabelle	242
6.4	Ereignisse, Aktivitäten und Dokumentation des Testprogramms	245
6.4.1	Ereignisse	245
6.4.2	Aktivitäten	246
6.4.3	Dokumentation	246
6.5	Die Testumgebung	247
6.5.1	Vorbereitung der Testumgebung	248
6.5.2	Integration und Einrichtung der Testumgebung	250
6.6	Der Testplan	251
6.6.1	Kriterien für den Testabschluss/die Akzeptanz der Testergebnisse	254
6.6.2	Beispiel eines Testplans	255
6.7	Zusammenfassung	255
6.8	Literaturhinweise	257
7.	Testanalyse und -design	258
7.1	Analyse der Testanforderungen	261
7.1.1	Testanalyse auf Entwicklungsebene (Struktureller Ansatz)	262
7.1.2	Testanalyse auf Systemebene (Verhaltensorientierter Ansatz)	264
7.2	Design des Testprogramms	268
7.2.1	Designmodelle für Testprogramme	269
7.2.2	White-Box-Techniken (Tests auf Entwicklungsebene)	273
7.2.3	Black-Box-Techniken (Tests auf Systemebene)	281
7.2.4	Dokumentation des Testdesigns	293
7.3	Entwerfen von Testverfahren	295
7.3.1	Definition von Testverfahren	296
7.3.2	Automatisierte oder manuelle Tests?	302
7.3.3	Standards für das Design automatisierter Tests	308
7.3.4	Richtlinien für das Design manueller Tests	315

7.3.5	Detailliertes Testdesign	318
7.3.6	Anforderungen an die Testdaten	320
7.4	Zusammenfassung	326
7.5	Literaturhinweise	328
8.	Entwicklung von Tests	329
8.1	Die Entwicklungsarchitektur für Tests	332
8.1.1	Die technische Umgebung	333
8.1.2	Überprüfen der Einsatzbereitschaft der Testumgebung	336
8.1.3	Analyse der Wiederverwendbarkeit vorhandener Automatisierungskomponenten	336
8.1.4	Zeitplan für die Entwicklung/Durchführung von Testverfahren	338
8.1.5	Analyse der Modularitätsbeziehungen	341
8.1.6	Erläuterung der Beispieltabelle für Modularitätsbeziehungen	345
8.1.7	Kalibrierung des Testwerkzeugs	348
8.1.8	Lösungen für Kompatibilitätsprobleme	349
8.1.9	Manuelle Durchführung von Testverfahren	351
8.1.10	Inspektionen von Testverfahren – Peer Reviews	352
8.1.11	Konfigurationsmanagement für Testverfahren	352
8.2	Richtlinien für das Entwickeln von Tests	354
8.2.1	Der Übergang vom Design zur Entwicklung	356
8.2.2	Wieder verwendbare Testverfahren	358
8.2.3	Wartungsfreundliche Testverfahren	367
8.2.4	Andere Richtlinien	386
8.3	Automatisierungsinfrastruktur	389
8.3.1	Tabellengestützte Testautomatisierung	390
8.3.2	Skript zum automatisierten Einrichten der PC-Umgebung	392
8.3.3	Optionen zum automatisierten Aufzeichnen	393
8.3.4	Anmeldefunktion	394
8.3.5	Funktion zum Beenden	394
8.3.6	Navigation	395
8.3.7	Verifizieren von GUI-Standards	395
8.3.8	Smoke-Test	396
8.3.9	Routinen zur Fehlerprotokollierung	397
8.3.10	Verifizierungsskript für die Hilfe-Funktion	398
8.3.11	Funktionen für zeitgesteuerte Hinweisfenster	399
8.3.12	Fortgeschrittene mathematische Funktionen	399
8.4	Zusammenfassung	399
8.5	Literaturhinweise	401

Teil IV Testdurchführung und -überprüfung

9.	Testdurchführung	405
9.1	Durchführung und Bewertung von Testphasen	407
9.1.1	Durchführung und Bewertung von Einheitentests	408
9.1.2	Durchführung und Bewertung von Integrationstests	411
9.1.3	Durchführung und Bewertung von Systemtests	413
9.1.4	Ergebnisanalyse bei Regressionstests	416
9.1.5	Durchführung und Bewertung von Benutzerakzeptanztests	417
9.2	Fehlerverfolgung und Umgang mit neuen Softwareversionen	418
9.2.1	Fehlerlebenszyklusmodell	424
9.3	Verfolgung des Testprogrammstatus	424
9.3.1	Ertragsverwaltung	427
9.3.2	Sammeln und Analysieren von Testmetriken	431
9.4	Zusammenfassung	439
9.5	Referenzen	441
10.	Überprüfung und Bewertung des Testprogramms	442
10.1	Korrektur- und Verbesserungsmaßnahmen	444
10.2	Kosten/Nutzen-Analyse des Testprogramms	457
10.3	Zusammenfassung	468
10.4	Literaturhinweise	469

Teil V Anhang

A.	Testen von Anforderungen	473
A.1	Ansatz zum Testen von Anforderungen	473
	Inhaltsangabe	473
	Das Qualitätstor	474
	Eine Anforderung messbar machen	474
	Quantifizierbare Anforderungen	475
	Nicht quantifizierbare Anforderungen	475
	Verfolgen	476
	Verständlichkeit und Konsistenz	477
	Vollständigkeit	478
	Relevanz	480
	Anforderung oder Lösung?	481
	Bewertung durch Beteiligte	482
	Verfolgbarkeit	482
	Ordnung in einer unordentlichen Welt	483
	Schlussfolgerungen	485
	Literaturhinweise	485
	Anmerkung	486

B.	Werkzeuge zur Unterstützung des Testlebenszyklus	487
B.1	Einführung	487
B.2	Geschäftsanalyse	492
B.2.1	Werkzeuge zur Geschäftsmodellierung	492
B.2.2	Werkzeuge für das Konfigurationsmanagement	495
B.2.3	Fehlerverfolgungswerkzeuge	497
B.2.4	Verwaltung technischer Reviews	504
B.2.5	Dokumentationsgeneratoren	505
B.3	Anforderungsdefinition	506
B.3.1	Werkzeuge für das Anforderungsmanagement	506
B.3.2	Werkzeuge zur Anforderungsverifikation	509
B.3.3	Anwendungsfallgeneratoren	509
B.4	Analyse und Design	511
B.4.1	Werkzeuge für visuelles Modellieren	511
B.4.2	Struktur-, Fluss- und Sequenzdiagramme	513
B.4.3	Generatoren für Testverfahren	515
B.5	Programmierung	516
B.5.1	Werkzeuge zur Syntaxüberprüfung/Debugger	516
B.5.2	Werkzeuge zur Erkennung von Speicherengpässen und Laufzeitfehlern	516
B.5.3	Werkzeuge zur Codeüberprüfung	519
B.5.4	Werkzeuge für die statische und dynamische Analyse	520
B.5.5	Werkzeuge für den Einheitentest	524
B.6	Metrikwerkzeuge	525
B.6.1	Werkzeuge zur Analyse der Code- bzw. Testabdeckung und zur Codeinstrumentalisierung	525
B.6.2	Werkzeuge zur Usability-Beurteilung	533
B.7	Werkzeuge zur Testunterstützung	534
B.7.1	Testdatengeneratoren	534
B.7.2	Werkzeuge für den Dateivergleich	535
B.7.3	Simulationswerkzeuge	536
B.8	Testphase	538
B.8.1	Werkzeuge für das Testmanagement	538
B.8.2	Werkzeuge zum Testen von Netzwerken	538
B.8.3	Werkzeuge zum Testen von GUI-Anwendungen	539
B.8.4	Werkzeuge für Belastungs-/Leistungstests	544
B.8.5	Werkzeuge zum Testen von Web-Anwendungen	549
B.8.6	Jahr-2000-Testwerkzeuge	550
B.9	Weitere Hersteller von Testwerkzeugen	552
	Literaturhinweise	552
C.	Die Karriere des Testingenieurs	553
C.1	Technische Kenntnisse	557
C.2	Testprozesse	559
C.3	Teamarbeit	563

XXII Inhaltsverzeichnis

C.4	Technische Leitung	564
C.5	Test-/Projektmanagement	565
C.6	Geschäfts-/Produktmanagement	567
	Literaturhinweise	568
D.	Beispiel eines Testplans	569
D.1	Einleitung	572
D.1.1	Absicht	572
D.1.2	Hintergrund	573
D.1.3	Systemübersicht	573
D.1.4	Relevante Dokumente	575
D.1.5	Übergeordneter Zeitplan	577
D.2	Aufgaben und Verantwortlichkeiten	579
D.2.1	Projektorganisation	579
D.2.2	Projektaufgaben und -verantwortlichkeiten	580
D.2.3	Struktur der Testaufgaben	583
D.2.4	Testteamressourcen	590
D.3	Testprogramm	591
D.3.1	Umfang	591
D.3.2	Testansatz	593
D.3.3	Teststrategien	597
D.3.4	Automatisierte Werkzeuge	600
D.3.5	Qualifikationsmethoden	601
D.3.6	Testanforderungen	602
D.3.7	Testdesign	603
D.3.8	Testentwicklung	607
D.4	Testumgebung	609
D.4.1	Testumgebungskonfiguration	609
D.4.2	Testdaten	611
D.5	Testdurchführung	613
D.5.1	Testprogrammberichte	613
D.5.2	Testprogrammmetriken	614
D.5.3	Fehlerverfolgung	615
D.5.4	Konfigurationsmanagement	617
D.6	Detaillierter Testzeitplan	617
	Anhang	
D.A	Entwicklungsrichtlinien für Testverfahren	618
D.B	Tabelle der Testverifizierungszusammenfassung	619
D.C	Testverfahren und -skripts	621
E.	Empfohlene Vorgehensweisen	622
E.1	Dokumentierter Prozess	623
E.2	Umgang mit Erwartungen	624
E.3	Pilotprojekt	624

E.4	Überprüfung der Testwerkzeugkompatibilität	625
E.5	Aktualisierung von Testwerkzeugen	625
E.6	Festlegen einer Baseline für Systemeinrichtung und -konfiguration	627
E.7	Softwareinstallationen in der Testumgebungs-Baseline	627
E.8	Übergeordnete Ziele des Testprogramms	627
E.9	Die Automatisierung einfach halten	628
E.10	Standards für das Design und die Entwicklung von Testverfahren	628
E.11	Automatisierte oder manuelle Tests?	629
E.12	Wiederverwendungsanalyse	629
E.13	Kommunikation des Testteams mit anderen Teams	630
E.14	Verträglichkeit der Zeitpläne	630
E.15	Einbeziehung des Kunden	631
E.16	Fehlerdokumentation und Fehlerberichte	632
E.17	Fürsprecher und Experten im automatisierten Testen	632
E.18	Zuordnungen innerhalb des Testteams	632
E.19	Beteiligung von Benutzergruppen	633
E.20	Vorschläge für die Verbesserung von Testwerkzeugen	633
E.21	Beta-Tests	634
E.22	Expertenwissen	634
	Stichwortverzeichnis	635