

Spis treści

Spis ilustracji	XXVII
Spis tabel	XXXIV
Spis listingów	XXXIX
Znaki handlowe	XLI
Przedmowa do drugiego wydania	XLIII
Wstęp	XLV
CZĘŚĆ I. PODSTAWY TESTOWANIA	1
1. Wprowadzenie do testowania	3
1.1. Dlaczego testowanie jest niezbędne	3
1.2. Definicja testowania	4
1.3. Słynne przykłady awarii oprogramowania	7
1.4. Rys historyczny	11
1.5. Ogólne zasady testowania	14
1.6. Jak dużo testować i kiedy skończyć?	18
1.7. Psychologia testowania	19
1.8. Kodeks etyczny testera	21
1.9. Role związane z testowaniem	22
2. Podstawowe definicje	25
3. Proces testowy	32
3.1. Podstawowy proces testowy	33
3.1.1. Planowanie	34
3.1.2. Monitorowanie i nadzór	36
3.1.3. Analiza testów	36
3.1.4. Projektowanie testów	39
3.1.5. Implementacja testów	43
3.1.6. Wykonanie testów	46
3.1.7. Ocena spełnienia kryteriów wyjścia oraz raportowanie	49
3.1.8. Czynności zamykające testowanie	50
3.2. Proces testowy wg ISO/IEC/IEEE 29119	52
4. Testowanie w cyklu życia oprogramowania	55
4.1. Modele wytwarzania oprogramowania	55

4.1.1. Model kaskadowy	56
4.1.2. Model V	57
4.1.3. Model W	58
4.1.4. <i>Rational Unified Process</i> (RUP)	59
4.1.5. <i>Rapid Application Development</i> (RAD)	60
4.1.6. Model spiralny Boehma	61
4.1.7. Metodyki zwinne	62
4.1.8. Metodologia Cleanroom	68
4.2. Weryfikacja i walidacja	70
4.3. Poziomy testów	72
4.3.1. Testy jednostkowe	73
4.3.2. Testy integracyjne	74
4.3.3. Testy systemowe	78
4.3.4. Testy akceptacyjne	79
4.3.5. Pozostałe poziomy testów	80
4.4. Typy testów	82
4.4.1. Testy funkcjonalne	83
4.4.2. Testy niefunkcjonalne	83
4.4.3. Testy strukturalne	84
4.4.4. Testy związane ze zmianami	85
4.5. Poziomy a typy testów	86
5. Testowanie a architektura systemu informatycznego	87
5.1. Architektura warstwowa	88
5.2. Model-Widok-Kontroler	91
5.3. Architektura typu microkernel	92
5.4. Mikrouslugi	94
5.5. Systemy wbudowane	95
CZĘŚĆ II. TECHNIKI PROJEKTOWANIA TESTÓW	97
6. Testowanie oparte na modelu	99
6.1. Cechy dobrego modelu	99
6.2. Taksonomia modeli	101
6.3. Przykład wykorzystania modelu	103
6.4. Modele działania oprogramowania	106
6.4.1. Graf przepływu sterowania	107
6.4.2. Ograniczenia w stosowaniu grafu przepływu sterowania	112
6.4.3. Graf przepływu danych	113
6.4.4. Ścieżki, ścieżki testowe i ścieżki nieosiągalne	115
7. Techniki testowania statycznego	117
7.1. Przeglądy	118
7.1.1. Proces dla testowania statycznego	120
7.1.2. Metody sprawdzania oraz możliwe wyniki przeglądu	121
7.1.3. Role	122
7.1.4. Aspekt psychologiczny przeglądów	123
7.1.5. Typy przeglądów	124

7.1.6. Biznesowa wartość przeglądów	133
7.1.7. Wdrażanie przeglądów	134
7.1.8. Kryteria sukcesu przeglądów	136
7.2. Analiza statyczna	137
7.2.1. Analiza przepływu sterowania	137
7.2.2. Poprawność sekwencji operacji	138
7.2.3. Analiza przepływu danych	140
7.2.4. Narzędzia do parsowania kodu	142
7.2.5. Testowanie zgodności ze standardami oprogramowania	144
7.2.6. Metryki złożoności kodu	144
7.2.7. Formalne dowodzenie poprawności	147
7.2.8. Symboliczne wykonywanie kodu	150
7.2.9. Analiza statyczna strony internetowej	153
7.2.10. Grafy wywołań	154
8. Analiza dynamiczna	159
8.1. Wykrywanie wycieków pamięci	161
8.2. Wykrywanie dzikich i wiszących wskaźników	162
8.3. Błędy API	163
8.4. Analiza wydajności (<i>profiling</i>)	164
9. Techniki oparte na specyfikacji (czarnoskrzynkowe)	168
9.1. Podział na klasy równoważności	169
9.1.1. Opis metody	169
9.1.2. Formalna definicja podziału	171
9.1.3. Poprawne i niepoprawne klasy równoważności	172
9.1.4. Procedura tworzenia przypadków testowych	173
9.1.5. Przykład	175
9.1.6. Przykład śledzenia artefaktów procesu testowego	179
9.2. Analiza wartości brzegowych	181
9.2.1. Opis metody	181
9.2.2. Metody dwóch oraz trzech wartości granicznych	182
9.2.3. Które wartości rozważać jako brzegowe?	185
9.2.4. Przypadek zmiennych ciągłych	188
9.2.5. Przykład	188
9.3. Tablice decyzyjne	191
9.3.1. Opis metody	191
9.3.2. Wartości nieistotne i minimalizacja tablicy decyzyjnej	195
9.3.3. Przykład	199
9.4. Grafy przyczynowo-skutkowe	201
9.4.1. Opis metody	201
9.4.2. Przekształcanie między grafami P-S i tablicami decyzyjnymi	204
9.4.3. Metoda śledzenia wstecznego - redukcja liczby testów	207
9.4.4. Przykład	211
9.5. Testowanie przejść między stanami	212
9.5.1. Opis metody	212
9.5.2. Tabelaryczne reprezentacje przejść	216
9.5.3. Kryteria pokrycia dla maszyny stanowej	218

9.5.4. Diagram maszyny stanowej w UML	222
9.5.5. Przykład	223
9.6. Kategoria-podział (<i>Category-Partition</i>)	226
9.6.1. Opis metody	226
9.6.2. Przykład	227
9.7. Drzewa klasyfikacji	232
9.7.1. Opis metody	232
9.7.2. Budowa drzewa klasyfikacji	232
9.7.3. Asortyment produktów programowych i model cech	234
9.7.4. Przykład	237
9.8. Metody kombinacyjne	239
9.8.1. Opis metody	239
9.8.2. Each Choice	240
9.8.3. Base Choice	241
9.8.4. Multiple Base Choice	242
9.8.5. Pair-wise testing	243
9.8.6. Pokrycie n -tupletów (n -wise)	249
9.8.7. Pełne pokrycie kombinatoryczne	250
9.8.8. Subsumpcja kryteriów kombinacyjnych	250
9.9. Testowanie dziedziny	251
9.9.1. Opis metody	251
9.9.2. Hiperpłaszczyzny i podprzestrzenie	251
9.9.3. Wyznaczanie hiperpłaszczyzny przez punkty	253
9.9.4. Punkty IN, OUT, ON i OFF	254
9.9.5. Strategia IN-OUT dla testowania dziedziny	255
9.9.6. Strategia N-ON x M-OFF dla testowania wartości brzegowych	255
9.9.7. Ograniczenia nieliniowe	257
9.9.8. Przykład	258
9.10. Testowanie oparte na przypadkach użycia	261
9.10.1. Opis metody	261
9.10.2. Przykład	262
9.11. Testowanie oparte na scenariuszach	267
9.11.1. Opis metody	267
9.11.2. Przykład	267
9.12. Testowanie oparte na historyjkach użytkownika	272
9.12.1. Opis metody	272
9.12.2. Przykład	274
9.13. Testowanie losowe	276
9.13.1. Opis metody	276
9.13.2. Wady i zalety testowania losowego	276
9.13.3. Automatyzacja i problem wyroczni	277
9.13.4. <i>Adaptive Random Testing</i> (ART)	278
9.13.5. Losowanie danych wejściowych	279
9.13.6. Przykład	280
9.14. Testowanie oparte na składni	282
9.14.1. Opis metody	282
9.14.2. Notacja Backusa-Naura (BNF)	283
9.14.3. Tworzenie przypadków testowych	284

9.14.4. Przykład	285
9.15. Testowanie CRUD	287
9.15.1. Opis metody	287
9.15.2. Rozszerzenie metody	288
9.15.3 Przykład	290
9.16. Wybór i łączenie technik ze sobą	293
10. Techniki oparte na strukturze (białoskrzynkowe)	294
10.1. Testowanie instrukcji	295
10.1.1. Opis metody	295
10.1.2. Przykład	297
10.2. Testowanie gałęzi	299
10.2.1. Opis metody	299
10.2.2. Przykład	300
10.3. Testowanie decyzji	302
10.3.1. Opis metody	302
10.3.2. Testowanie decyzji a testowanie gałęzi	304
10.3.3. Przykład	305
10.4. Testowanie warunków	308
10.4.1. Opis metody	308
10.4.2. Przykład	310
10.5. Testowanie warunków/decyzji	311
10.5.1. Opis metody	311
10.5.2. Przykład	312
10.6. Testowanie wielokrotnych warunków	313
10.6.1. Opis metody	313
10.6.2. Zwarcie (semantyka <i>short-circuit</i>)	314
10.6.3. Nieosiągalne kombinacje warunków	315
10.6.4. Przykład	316
10.7. Testowanie warunków znaczących (MC/DC)	318
10.7.1. Opis metody	318
10.7.2. Dwa warianty kryterium MC/DC	319
10.7.3. Skorelowane a ściśle pokrycie warunków znaczących	321
10.7.4. Wyznaczanie wartości warunków pobocznych dla warunku znaczącego	323
10.7.5. Przykład	325
10.8. Pokrycie MUMCUT oraz kryteria z nim związane	327
10.8.1. Opis metody	327
10.8.2. Kryteria MUTP, MNFP, CUTPNFP i MUMCUT	327
10.8.3. Przykład	328
10.8.4. Zdolność metody MUMCUT do wykrywania określonych typów błędów	329
10.9. Testowanie pętli	330
10.9.1. Opis metody	330
10.9.2. Pętle zagnieżdżone	331
10.9.3. Testowanie wzorców pętli	331
10.9.4. Przykład	334
10.10. Liniowa sekwencja kodu i skok (LSKiS)	336

10.10.1. Opis metody	336
10.10.2. Przykład	338
10.10.3. LSKiS a DD-ścieżki i bloki podstawowe	340
10.11. Testowanie ścieżek pierwszych	342
10.11.1. Opis metody	342
10.11.2. Algorytm wyznaczania ścieżek pierwszych	344
10.11.3. Przykład	345
10.12. Analiza ścieżek	348
10.12.1. Wstęp	348
10.12.2. Testowanie wszystkich ścieżek	349
10.12.3. Ścieżki liniowo niezależne i testowanie ścieżek bazowych	349
10.12.4. Wyznaczanie ścieżek bazowych	353
10.12.5. Przykład	355
10.12.6. Inne kryteria pokrycia związane z testowaniem ścieżek	357
10.13. Testowanie przepływu danych	358
10.13.1. Wstęp	358
10.13.2. Pokrycie wszystkich definicji (<i>all-defs</i>)	360
10.13.3. Pokrycie wszystkich użyć (<i>all-uses</i>)	361
10.13.4. Pokrycie wszystkich du-ścieżek (<i>all-du-paths</i>)	361
10.13.5. Uwagi o kryteriach pokrycia przepływu danych	362
10.13.6. Przykład	362
10.14. Objazdy i ścieżki poboczne (<i>detours, sidetrips</i>)	366
10.15. Testowanie mutacyjne	367
10.15.1. Wstęp	367
10.15.2. Rodzaje mutantów	371
10.15.3. Proces testowania mutacyjnego	372
10.15.4. Operatory mutacyjne	373
10.16. Subsumpcja kryteriów	378
11. Techniki oparte na defektach i na doświadczeniu	379
11.1. Wstrzykiwanie błędów	380
11.2. Taksonomie	382
11.3. Zgadywanie błędów	385
11.3.1. Opis metody	385
11.3.2. Przykład	385
11.4. Testowanie oparte na liście kontrolnej	386
11.5. Testowanie eksploracyjne	387
11.5.1. Opis metody	387
11.5.2. Przykład sesji testowania eksploracyjnego	389
11.6. Ataki usterkowe	392
11.7. Testowanie ad hoc	394
12. Wybór odpowiednich technik	395
13. Priorytetyzacja przypadków testowych	398
13.1. Wprowadzenie	398
13.2. Ocena priorytetyzacji - miara APFD	399
13.3. Techniki priorytetyzacji	401

14. Testowanie systemów specyficznych	403
14.1. Testowanie aplikacji mobilnych	403
14.2. Testowanie aplikacji webowych	404
14.3. Testowanie graficznego interfejsu użytkownika (GUI)	406
14.4. Testowanie API	407
14.5. Testowanie aplikacji opartych na danych	409
14.6. Testowanie gier komputerowych	410
 CZĘŚĆ III. TESTOWANIE CHARAKTERYSTYK JAKOŚCIOWYCH	 413
15. Model jakości według ISO 9126	415
16. Modele jakości według ISO 25010	417
17. Testowanie jakości użytkowej	421
17.1. Testowanie efektywności (<i>effectiveness</i>)	421
17.2. Testowanie wydajności (<i>efficiency</i>)	421
17.3. Testowanie satysfakcji (<i>satisfaction</i>)	422
17.3.1. Przydatność (<i>usefulness</i>)	422
17.3.2. Zaufanie (<i>trust</i>)	422
17.3.3. Przyjemność (<i>pleasure</i>)	423
17.3.4. Komfort (<i>comfort</i>)	423
17.4. Testowanie wolności od ryzyka (<i>freedom from risk</i>)	423
17.4.1. Ryzyko ekonomiczne (<i>economic risk</i>)	423
17.4.2. Ryzyko dotyczące zdrowia i bezpieczeństwa (<i>health and safety risk</i>)	424
17.4.3. Ryzyko związane ze środowiskiem (<i>environmental risk</i>)	424
17.5. Testowanie kontekstu użycia (<i>context coverage</i>)	424
17.5.1. Zupełność kontekstu (<i>context completeness</i>)	424
17.5.2. Elastyczność (<i>flexibility</i>)	425
 18. Testowanie jakości produktu	 426
18.1. Testowanie funkcjonalnej przydatności (<i>functional suitability</i>)	426
18.1.1. Zupełność funkcjonalności (<i>functional completeness</i>)	427
18.1.2. Poprawność funkcjonalności (<i>functional correctness</i>)	427
18.1.3. Stosowność funkcjonalności (<i>functional appropriateness</i>)	428
18.2. Testowanie wydajności w działaniu (<i>performance efficiency</i>)	428
18.2.1. Zachowanie w czasie (<i>time behaviour</i>)	429
18.2.2. Zużycie zasobów (<i>resource utilization</i>)	429
18.2.3. Pojemność (<i>capacity</i>)	430
18.2.4. Techniki testowania wydajności	430
18.3. Testowanie zgodności (<i>compatibility</i>)	433
18.3.1. Współistnienie (<i>co-existence</i>)	433
18.3.2. Współdziałanie (<i>interoperability</i>)	434
18.3.3. Przykład	434
18.4. Testowanie użyteczności (<i>usability</i>)	435
18.4.1. Zrozumiałość (<i>appropriateness recognizability</i>)	437

18.4.2. Łatwość nauki (<i>learnability</i>)	438
18.4.3. Łatwość użycia (<i>operability</i>)	438
18.4.4. Ochrona przed błędami użytkownika (<i>user error protection</i>)	438
18.4.5. Estetyka interfejsu użytkownika (<i>user interface aesthetics</i>)	439
18.4.6. Dostęp (<i>accessibility</i>)	439
18.5. Heurystyki dotyczące użyteczności	440
18.5.1. Heurystyki Nielsena	440
18.5.2. Laboratorium badania użyteczności (<i>usability lab</i>)	446
18.6. Testowanie niezawodności (<i>reliability</i>)	448
18.6.1. Dojrzałość (<i>maturity</i>)	449
18.6.2. Odporność na błędy (<i>fault tolerance, robustness</i>)	451
18.6.3. Odtwarzalność (<i>recoverability</i>)	451
18.6.4. Dostępność (<i>availability</i>)	452
18.7. Testowanie zabezpieczeń (<i>security</i>)	452
18.7.1. Poufność (<i>confidentiality</i>)	453
18.7.2. Integralność (<i>integrity</i>)	453
18.7.3. Niezaprzeczalność (<i>non-repudiation</i>)	453
18.7.4. Odpowiedzialność (<i>accountability</i>)	453
18.7.5. Uwierzytelnianie (<i>authenticity</i>)	454
18.8. Testowanie pielęgnowalności (<i>maintainability</i>)	454
18.8.1. Modularność (<i>modularity</i>)	454
18.8.2. Powtórne użycie (<i>reusability</i>)	456
18.8.3. Analizowalność (<i>analyzability</i>)	456
18.8.4. Modyfikowalność (<i>modifiability</i>)	457
18.8.5. Testowalność (<i>testability</i>)	457
18.9. Testowanie przenaszalności (<i>portability</i>)	460
18.9.1. Adaptowalność (<i>adaptability</i>)	460
18.9.2. Instalowalność (<i>installability</i>)	461
18.9.3. Zastępowalność (<i>replaceability</i>)	462
19. Testowanie jakości danych	463
19.1. Model jakości danych	463
19.2. Charakterystyki inherentne	464
19.2.1. Dokładność (<i>accuracy</i>)	464
19.2.2. Zupełność (<i>completeness</i>)	465
19.2.3. Spójność (<i>consistency</i>)	466
19.2.4. Wiarygodność (<i>credibility</i>)	466
19.2.5. Aktualność (<i>currentness</i>)	466
19.3. Charakterystyki inherentne i zależne od systemu	467
19.3.1. Dostępność (<i>accessibility</i>)	467
19.3.2. Zgodność (<i>compliance</i>)	467
19.3.3. Poufność (<i>confidentiality</i>)	468
19.3.4. Wydajność (<i>efficiency</i>)	468
19.3.5. Precyzja (<i>precision</i>)	469
19.3.6. Identyfikowalność (<i>traceability</i>)	469
19.3.7. Zrozumiałość (<i>understandability</i>)	470
19.4. Charakterystyki zależne od systemu	470
19.4.1. Dostępność (<i>availability</i>)	470

19.4.2. Przenaszalność (<i>portability</i>)	470
19.4.3. Odtwarzalność (<i>recoverability</i>)	470
CZĘŚĆ IV. ZARZĄDZANIE TESTOWANIEM	473
20. Zarządzanie testowaniem w kontekście	475
20.1. Kontekst ograniczeń projektowych	476
20.2. Kontekst interesariuszy procesu testowego	476
20.3. Kontekst produkcji oprogramowania	478
20.4. Kontekst cyklu życia oprogramowania	480
20.5. Kontekst testów	480
20.6. Kontekst czynnika ludzkiego	481
21. Testowanie oparte na ryzyku	482
21.1. Czym jest ryzyko?	483
21.2. Zalety testowania opartego na ryzyku	486
21.3. Rodzaje ryzyka	488
21.4. Zarządzanie ryzykiem w cyklu życia	489
21.5. Identyfikacja ryzyka	491
21.5.1. Analiza interesariuszy (<i>stakeholder analysis</i>)	491
21.5.2. Technika „władza versus zainteresowanie”	493
21.5.3. Techniki identyfikacji ryzyk	494
21.5.4. Przykłady ryzyk produktowych	496
21.6. Analiza ryzyka	498
21.6.1. Klasyfikacja ryzyk	498
21.6.2. Czynniki wpływające na prawdopodobieństwo i wpływ ryzyka	499
21.6.3. Ilościowa i jakościowa ocena ryzyka produktowego	500
21.6.4. Osiąganie konsensusu w procesie decyzyjnym	501
21.6.5. Priorytetyzacja ryzyk	502
21.7. Łagodzenie ryzyka	504
21.7.1. Sposoby łagodzenia ryzyka	504
21.7.2. Łagodzenie ryzyka przez testowanie	505
21.7.3. Estymacja kosztów łagodzenia ryzyka	507
21.8. Monitorowanie ryzyka	509
21.8.1. Macierz identyfikowalności ryzyk	510
21.8.2. Aktualizacja ryzyk oraz ich parametrów	511
21.8.3. Raportowanie	512
21.9. Techniki analizy ryzyka	513
21.9.1. VRAM (<i>Pragmatic Risk Analysis and Management</i>)	514
21.9.2. SST (<i>Systematic Software Testing</i>)	516
21.9.3. Przykład: zastosowanie SST do systemu ELROJ	519
21.9.4. PRisMa (<i>Product Risk Management</i>)	520
21.9.5. Przykład: zastosowanie PRisMa do systemu ELROJ	528
21.9.6. Analiza zagrożeń (<i>hazard analysis</i>)	531
21.9.7. Koszt ekspozycji ryzyka (<i>cost of exposure</i>)	533
21.9.8. FMEA (<i>Failure Mode and Effect Analysis</i>)	535
21.9.9. Przykład: zastosowanie FMEA do systemu ELROJ	537
21.9.10. QFD (<i>Quality Function Deployment</i>)	538

21.9.11. Przykład: zastosowanie QFD do systemu ELROJ	541
21.9.12. FTA (<i>Fault Tree Analysis</i>)	543
21.9.13. Przykład: zastosowanie FTA do systemu ELROJ	547
21.10. TMap (Test Management Approach)	549
21.11. TestGoal - testowanie oparte na wynikach	550
22. Pozostałe strategie testowania	553
22.1. Testowanie oparte na wymaganiach	553
22.1.1. Testowanie wymagań	554
22.1.2. Projektowanie testów opartych na wymaganiach	557
22.1.3. Przykład: obliczanie punktacji w grze w kręgle	558
22.2. Podejście oparte na modelu (profile operacyjne)	560
22.3. Podejście metodyczne (listy kontrolne)	562
22.4. Podejście oparte na standardzie	563
22.5. Inne podejścia	565
22.5.1. Podejście reaktywne	565
22.5.2. Podejście <i>good enough</i>	566
22.5.3. Podejście konsultacyjne	566
23. Dokumentacja w zarządzaniu testowaniem	567
23.1. Dokumenty organizacyjnego procesu testowego	570
23.1.1. Polityka testów	570
23.1.2. Przykład polityki testów	571
23.1.3. Organizacyjna strategia testowania	573
23.1.4. Przykład organizacyjnej strategii testowania	575
23.2. Dokumenty procesu zarządzania testowaniem	578
23.2.1. Plan testów (<i>test plan</i>)	578
23.2.2. Przykład planu testów	580
23.2.3. Jednopoziomowy plan testów (<i>level test plan</i>)	586
23.2.4. Przykład jednopoziomowego planu testów	588
23.2.5. Raport o stanie testów (<i>test status report</i>)	591
23.2.6. Przykład raportu o stanie testów	591
23.2.7. Raport końcowy z testowania (<i>test completion report</i>)	592
23.2.8. Przykład raportu końcowego z testowania	593
23.3. Dokumenty dynamicznych procesów testowych	594
23.3.1. Specyfikacja testów (<i>test design specification</i>)	594
23.3.2. Przykład specyfikacji testów	595
23.3.3. Specyfikacja przypadku testowego (<i>test case specification</i>)	596
23.3.4. Przykład specyfikacji przypadku testowego	597
23.3.5. Specyfikacja procedury testowej (<i>test procedure specification</i>)	597
23.3.6. Przykład specyfikacji procedury testowej	598
23.3.7. Wymagania co do danych testowych (<i>test data requirements</i>)	599
23.3.8. Przykład wymagania co do danych testowych	600
23.3.9. Wymagania co do środowiska testowego (<i>test environment requirements</i>)	601
23.3.10. Przykład wymagań co do środowiska testowego	602
23.3.11. Raport o gotowości danych testowych (<i>test data readiness report</i>)	603

23.3.12. Raport o gotowości środowiska testowego (<i>test environment readiness report</i>)	603
23.3.13. Otrzymane wyniki (<i>actual results</i>)	603
23.3.14. Wynik testu (<i>test result</i>)	603
23.3.15. Dziennik wykonania testów (<i>test execution log</i>)	604
23.3.16. Raport o incydencie (<i>test incident report</i>)	604
23.3.17. Raport z sesji testowania eksploracyjnego (<i>exploratory testing session report</i>)	605
23.3.18. Przykład raportu z sesji testowania eksploracyjnego	605
24. Szacowanie testów	607
24.1. Czynniki wpływające na szacowanie	608
24.2. Techniki szacowania	610
24.2.1. Intuicja, zgadywanie, doświadczenie	610
24.2.2. Estymacja przez analogię	611
24.2.3. Struktura podziału prac (<i>Work Breakdown Structure, WBS</i>)	611
24.2.4. Estymacja grupowa	612
24.2.5. Dane przemysłowe	613
24.2.6. Analiza punktów testowych (<i>Test Point Analysis</i>)	614
24.2.7. Modele matematyczne (parametryczne)	619
24.3. Negocjacje i redukcja zakresu testów	619
25. Nadzór i kontrola postępu testów	621
25.1. Wprowadzenie	621
25.2. Przykłady metryk	624
25.2.1. Metryki ryzyka produktowego	625
25.2.2. Metryki defektów	625
25.2.3. Metryki przypadków testowych	628
25.2.4. Metryki pokrycia	629
25.2.5. Metryki pewności	630
25.3. Zarządzanie testowaniem opartym na sesji	631
26. Biznesowa wartość testowania	633
26.1. Wprowadzenie	633
26.2. Koszt jakości	634
26.3. Wartość ekonomiczna oprogramowania i problemy z nią związane	637
26.4. Dług technologiczny w kontekście testowania	638
27. Testowanie rozproszone, zakontraktowane i zewnętrzne	640
27.1. Zespoły w ramach organizacji	641
27.2. Zespoły w ramach innych oddziałów organizacji	642
27.3. Dostawcy sprzętu i oprogramowania	643
27.4. Dostawcy usług testowych	643
27.5. TaaS (<i>Testing as a Service</i>)	643
28. Zarządzanie wdrażaniem standardów przemysłowych	646
29. Zarządzanie incydentami	647

29.1. Cykl życia defektu	648
29.2. Atrybuty defektu i ODC	650
29.2.1. Atrybuty defektów	650
29.2.2. ODC (<i>Orthogonal Defect Classification</i>)	650
29.2.3. Przykładowe analizy przy użyciu ODC	651
29.3. Metryki zarządzania incydentami	653
29.4. Zawartość raportu o incydencie	655
29.5. Komunikowanie incydentów	656

CZĘŚĆ V. LUDZIE I NARZĘDZIA **657**

30. Ludzie i ich kompetencje - tworzenie zespołu **659**

30.1. Budowanie zespołu testowego	660
30.1.1. Opis stanowiska	661
30.1.2. Analiza CV	663
30.1.3. Rozmowa kwalifikacyjna	663
30.1.4. Asymilacja nowego członka zespołu	666
30.1.5. Zakończenie zatrudnienia	666
30.2. Rozwój zespołu testowego	667
30.2.1. Indywidualny rozwój poszczególnych członków zespołu	667
30.2.2. Wyznaczanie celów	668
30.2.3. Dynamika zespołu testowego	669
30.2.4. Określanie ról i odpowiedzialności	671
30.2.5. Umiejętności zespołu jako całości: <i>gap analysis</i>	672
30.2.6. Indywidualne osobowości i role w zespole	674
30.2.7. Rozwój umiejętności i szkolenia	677
30.2.8. Mentoring	678
30.2.9. Okresowa ocena członków zespołu	678
30.3. Przywództwo	679
30.3.1. Zarządzanie a przywództwo	679
30.3.2. Model zmiany według Satir	679
30.3.3. Cechy charakteru lidera i zasady przywództwa	680
30.3.4. Informowanie i komunikacja	681
30.3.5. Lojalność, zaufanie i odpowiedzialność	682
30.3.6. Budowanie zespołu	682
30.4. Poziomy niezależności zespołu testowego	682
30.5. Komunikacja	684

31. Techniki pracy grupowej **685**

31.1. Proces podejmowania decyzji	686
31.2. Techniki wspomagania kreatywności	686
31.2.1. Burza mózgów	686
31.2.2. NGT (<i>Nominal Group Technique</i>)	687
31.2.3. Metoda analogii	688
31.2.4. JAD (<i>Joint Application Development</i>)	688
31.3. Techniki porządkowania i priorytetyzacji	688
31.3.1. Diagram podobieństwa (<i>affinity diagram</i>)	688
31.3.2. Macierz i graf priorytetyzacji	689

31.3.3. Mapa myśli	691
31.4. Techniki szacowania i oceny	691
31.4.1. Wielokrotne głosowanie	691
31.4.2. Metoda delficka i Wideband Delphi	692
31.4.3. Poker planistyczny (<i>planning poker</i>)	695
31.4.4. Model Saaty'ego (<i>Analytic Hierarchy Process</i>)	696
31.5. Definiowanie problemu, przyczyny źródłowej lub możliwości	698
31.5.1. Analiza pola sił (<i>force field analysis</i>)	698
31.5.2. Immersja	699
31.5.3. Diagram rybiej ości (diagram Ishikawy)	700
32. Testowanie wspierane narzędziami	702
32.1. Podstawowe zagadnienia związane z użyciem narzędzi	703
32.1.1. Wybór odpowiedniego narzędzia	704
32.1.2. Koszty wdrożenia narzędzia	705
32.1.3. Ryzyka związane z wdrożeniem narzędzia	706
32.1.4. Korzyści z wdrożenia narzędzia	707
32.1.5. Strategie wdrażania automatyzacji	708
32.1.6. Integracja i wymiana informacji między narzędziami	708
32.1.7. Klasyfikacja narzędzi testowych	709
32.2. Automatyzacja testów	711
32.2.1. Czynniki sukcesu udanej automatyzacji	711
32.2.2. Metryki dla automatyzacji testów	712
32.3. Genetyczna architektura automatyzacji testów	713
32.3.1. Warstwa generowania testów	715
32.3.2. Warstwa definiowania testów	715
32.3.3. Warstwa wykonania testów	715
32.3.4. Warstwa adaptacji testów	716
32.3.5. Zarządzanie konfiguracją	716
32.4. Automatyczna generacja danych testowych	717
32.5. Metody i techniki automatyzacji testów	719
32.5.1. Kroki procesu projektowania architektury	719
32.5.2. Podejścia do automatyzacji przypadków testowych	723
32.5.3. Podejście nagraj i odtwórz	724
32.5.4. Skrypty linearne	728
32.5.5. Skrypty zorganizowane	731
32.5.6. Testowanie oparte na danych	733
32.5.7. Testowanie oparte na słowach kluczowych	735
32.5.8. Testowanie oparte na procesie	739
32.5.9. Testowanie oparte na modelu	741
32.5.10. Języki i notacje dla definicji testów	742
32.6. Katalog narzędzi testowych	749
32.6.1. Narzędzia do zarządzania testami	750
32.6.2. Narzędzia do wykonywania/organizacji testów	750
32.6.3. Narzędzia do testowania mutacyjnego, posiewu usterek i wstrzykiwania błędów	751
32.6.4. Narzędzia do testów bezpieczeństwa	751
32.6.5. Symulatory i emulatory	751

32.6.6. Narzędzia do analizy statycznej i dynamicznej	752
32.6.7. Narzędzia do testów wydajnościowych	753
32.6.8. Narzędzia typu <i>capture & replay</i>	753
32.6.9. Narzędzia do testów jednostkowych	753
32.6.10. Testowanie w metodyce BDD	754
32.6.11. Narzędzia do testowania opartego na modelu	755
32.6.12. Narzędzia do śledzenia defektów, zarządzania incydentami i raportowania	755
32.7. Wdrażanie narzędzi w organizacji	756
32.7.1. Cykl życia narzędzia	756
32.7.2. Użycie narzędzi <i>open source</i>	756
32.7.3. Przykładowe problemy związane z użyciem narzędzi	758
33. Rozwój zawodowy testera	759
33.1. Charakterystyka testera idealnego	759
33.2. Edukacja, zdobywanie wiedzy	764
33.3. Czerpanie inspiracji	766
CZĘŚĆ VI. UDOSKONALANIE PROCESU TESTOWEGO	769
34. Kontekst udoskonalania procesu	771
34.1. Po co udoskonaląć?	771
34.2. Co można udoskonalić?	773
34.3. Spojrzenia na jakość	775
34.3.1. Jakość oparta na produkcji	775
34.3.2. Jakość oparta na użytkowniku	775
34.3.3. Jakość oparta na wytwarzaniu	776
34.3.4. Jakość oparta na wartości	776
34.3.5. Jakość transcendentna	776
34.4. Generyczny proces udoskonalania	777
34.4.1. Cykl Deminga-Shewarta (PDCA)	777
34.4.2. IDEAL	779
34.4.3. Podstawowe zasady doskonałości (<i>Fundamental Concepts of Excellence</i>)	781
34.5. Przegląd podejść do udoskonalania	782
34.5.1. Podejścia oparte na modelu	782
34.5.2. Podejścia analityczne	785
34.5.3. Podejścia hybrydowe	785
34.5.4. Inne podejścia do udoskonalania procesu testowego	785
35. Udoskonalanie oparte na modelu	788
35.1. Wprowadzenie	788
35.2. Udoskonalanie procesów organizacyjnych i biznesowych	789
35.2.1. TQM (<i>Total Quality Management</i>)	790
35.2.2. ISO 9000	791
35.2.3. EFQM Excellence Model	791
35.2.4. Six Sigma	792
35.2.5. Lean	793

35.3. Udoskonalanie procesu produkcji oprogramowania	794
35.3.1. CMMI	794
35.3.2. ISO/IEC 15504	796
35.3.3. ITIL (<i>Information Technology Infrastructure Library</i>)	796
35.3.4. TSP (<i>Team Software Process</i>)	797
35.3.5. BOOTSTRAP	797
35.4. Udoskonalanie procesu testowego - modele referencyjne procesu	798
35.4.1. TPI Next (<i>Test Process Improvement</i>)	798
35.4.2. TMMi (<i>Test Maturity Model integration</i>)	806
35.4.3. Porównanie TPI Next i TMMi	816
35.5. Udoskonalanie procesu testowego - modele referencyjne zawartości	816
35.5.1. STEP (<i>Systematic Test and Evaluation Process</i>)	816
35.5.2. CTP (<i>Critical Testing Processes</i>)	820
36. Podejście analityczne	822
36.1. Wprowadzenie	822
36.2. Analiza przyczynowa	822
36.2.1. Opis metody	822
36.2.2. Wybór elementów do analizy przyczynowej	823
36.2.3. Zebranie i zorganizowanie informacji	831
36.2.4. Identyfikacja przyczyny źródłowej przez analizę zebranych informacji	832
36.2.5. Wyciągnięcie wniosków	833
36.3. Podejście GQM (<i>Goal-Question-Metric</i>)	834
36.3.1. Opis metody	834
36.3.2. Fazy GQM	834
36.3.3. Dwa paradygmaty metody GQM	835
36.3.4. Wzorzec definiowania celu	837
36.3.5. Siedem pytań	837
36.3.6. Przykład	838
36.4. Miary, metryki i wskaźniki	839
37. Wybór metody usprawniania	840
38. Proces udoskonalania	842
38.1. Wprowadzenie	842
38.2. Rozpoczęcie procesu doskonalenia	843
38.2.1. Określenie powodu doskonalenia (stymulacja do zmiany)	843
38.2.2. Ustanowienie celów dla doskonalenia testowania	844
38.2.3. Określenie kontekstu	847
38.2.4. Pozyskanie wsparcia	848
38.2.5. Stworzenie infrastruktury dla procesu udoskonalania	848
38.3. Diagnozowanie aktualnej sytuacji	849
38.3.1. Scharakteryzowanie obecnego oraz pożądanego stanu procesu	849
38.3.2. Rekomendacje akcji naprawczych	851
38.4. Ustanowienie planu doskonalenia procesu testowego	851
38.4.1. Ustanowienie priorytetów dla wdrażania planu doskonalenia	851

38.4.2. Opracowanie podejścia do wdrożenia	852
38.4.3. Zaplanowanie działań związanych z wdrożeniem	853
38.5. Działanie w celu wdrożenia udoskonaleń	853
38.5.1. Stworzenie rozwiązania	854
38.5.2. Rozwiązania pilotażowe/testowe	854
38.5.3. Doprecyzowanie rozwiązania	855
38.5.4. Zaimplementowanie rozwiązania	855
38.6. Wyciąganie wniosków z projektu doskonalenia testów	855
38.6.1. Analiza i weryfikacja przeprowadzonych działań	855
38.6.2. Propozycje przyszłych rozwiązań	856
39. Organizacja, role i umiejętności	857
39.1. Organizacja	857
39.1.1. Zakres działań GPT	857
39.1.2. Organizacja Grupy procesu testowego	858
39.1.3. Właściwości Grupy procesu testowego	859
39.2. Role i umiejętności	859
39.2.1. Doskonalący proces testowy	859
39.2.2. Główny oceniający	861
39.2.3. Oceniający	861
39.2.4. Umiejętności doskonalącego proces testowy	862
40. Czynniki sukcesu	865
CZĘŚĆ VII. JAKOŚĆ OPROGRAMOWANIA	867
41. Czym jest jakość oprogramowania?	869
41.1. Testowanie oprogramowania a jakość oprogramowania	869
41.2. Model Kano	870
41.3. Dojrzałość procesu i standardy jakości	871
41.3.1. SPR	872
41.3.2. Ocena Malcolma Baldridge'a	873
41.4. Co mierzyć, jak mierzyć i po co mierzyć?	873
42. Podstawy teorii pomiarów	875
42.1. Metryka, miara, wskaźnik, pomiar	875
42.2. Skale pomiarowe	877
42.2.1. Skala nominalna	877
42.2.2. Skala porządkowa	878
42.2.3. Skala interwałowa	879
42.2.4. Skala stosunkowa	879
42.2.5. Podsumowanie rodzajów skal pomiarowych	879
42.3. Typy metryk	880
42.3.1. Metryka bezpośrednia (podstawowa)	880
42.3.2. Suma/różnica	880
42.3.3. Stosunek	880
42.3.4. Proporcja	881
42.3.5. Odsetek	881

42.3.6. Miary iloczynowe	881
42.3.7. Tempo	882
42.4. Spójność i odpowiedniość pomiaru	882
42.5. Błędy pomiarowe	884
42.6. Podstawowe zasady analizy danych	885
42.6.1. Miary tendencji centralnej	885
42.6.2. Miary rozproszenia	887
42.6.3. Korelacja i regresja liniowa	888
42.6.4. Przyczynowość	893
43. Narzędzia kontroli jakości	895
43.1. Klasyfikacja narzędzi	895
43.2. Rodzaje narzędzi oraz obszary ich zastosowań	896
43.3. Statystyczna kontrola procesu	896
43.3.1. Wykres przebiegu	903
43.3.2. Karty kontrolne	904
43.4. Wykres czasu cyklu	910
43.5. Narzędzia analizy i zapobiegania przyczynom źródłowym	910
43.5.1. 5 pytań „dlaczego?” i diagram „why-why”	911
43.5.2. Macierz „jest-nie jest”	912
43.5.3. Kaizen	913
43.5.4. Poka yoke	913
44. Metryki wielkości oprogramowania	915
44.1. Metryki wolumenowe	915
44.1.1. LOC	916
44.1.2. Współczynnik produktywności języka	918
44.1.3. Pomiar specyfikacji i projektu	919
44.2. Metryki funkcjonalności	920
44.2.1. Punkty funkcyjne	920
44.2.2. Punkty obiektowe i rozszerzone punkty obiektowe	927
44.2.3. Punkty cech	927
44.2.4. Punkty przypadków użycia	928
45. Metryki charakterystyk jakościowych	931
45.1. Metryki dla funkcjonalności	931
45.2. Metryki dla niezawodności	932
45.3. Metryki dla użyteczności	933
45.4. Metryki dla wydajności	934
45.5. Metryki dla pielęgnowalności	935
45.6. Metryki dla przenaszalności	936
46. Metryki złożoności oprogramowania	937
46.1. Metryki Halsteada	937
46.2. Złożoność cyklomatyczna McCabe'a	939
46.2.1. Gęstość złożoności cyklomatycznej	941
46.2.2. ECC (Essential Cyclomatic Complexity)	941
46.3. Konstrukcje składniowe	944

46.4. Metryki struktur	944
46.5. Metryki złożoności systemu	945
46.5.1. Indeks utrzymywalności	945
46.5.2. Metryka złożoności systemu Agrestiego-Carda-Glassa	946
46.6. Metryki obiektowe	948
46.6.1. Metryki Lorenza	948
46.6.2. Metryki CK	948
46.7. Metryki złożoności dokumentacji	952
46.8. Metryki złożoności algorytmicznej	953
47. Metryki i modele wysiłku	954
47.1. Modele oparte na zgadywaniu i intuicji	955
47.2. Modele oparte na dekompozycji	955
47.3. Modele oparte na wiedzy eksperckiej	956
47.4. Modele oparte na benchmarkach	956
47.5. Modele oparte na porównaniu	958
47.5.1. Porównanie proste (naiwne)	958
47.5.2. Porównanie z uwzględnieniem różnic	959
47.6. Modele parametryczne	960
47.6.1. Tworzenie własnego modelu	962
47.6.2. Model COCOMO II	963
47.7. Łączenie modeli i uwagi na temat estymacji	966
48. Metryki i modele dla defektów	968
48.1. Natura defektów	970
48.2. Metryki defektów	971
48.3. Modele statyczne defektów	973
48.3.1. Model wprowadzania/usuwania defektów	973
48.3.2. Model fazowy	975
48.3.3. Model dwufazowy	976
48.3.4. Efektywność usuwania defektów i powstrzymanie fazowe	977
48.3.5. Modele zmian w kodzie	981
48.4. Modele dynamiczne defektów	983
48.4.1. Model Rayleigha	983
48.4.2. Model wykładniczy i S-kształtny	991
48.4.3. Model COQUALMO	992
48.5. Analiza mutacyjna	994
48.6. Metryki dynamicznej stylometrii	995
49. Metryki i modele przyrostu niezawodności	997
49.1. Wprowadzenie	997
49.2. Matematyczne podstawy teorii niezawodności	1000
49.2.1. Funkcja niezawodności i funkcja awarii	1000
49.2.2. Metryka MTTF (średniego czasu do awarii)	1000
49.2.3. Rozkłady prawdopodobieństwa modelujące występowanie awarii	1001
49.2.4. Rozkład wykładniczy i jego związek z metryką MTTF	1001
49.2.5. Funkcja częstości awarii oraz ryzyko (<i>hazard rate</i>)	1002
49.2.6. Prawdopodobieństwo awarii do czasu t	1003

49.3. Modele przyrostu niezawodności	1004
49.3.1. Model Jelinskiego-Morandy	1005
49.3.2. Model niedoskonałego debugowania Goela-Okumoto	1009
49.3.3. Niejednorodny model procesu Poissona Goela-Okumoto	1009
49.3.4. Logarytmiczny model Poissona czasu wykonywania Musy-Okumoto	1010
49.3.5. Model S-kształtny	1011
49.3.6. Inne modele niezawodności	1011
50. Metryki i modele dostępności	1013
50.1. Dostępność	1013
50.1.1. Dostępność ciągła a dostępność wysoka	1014
50.1.2. Metody zwiększające dostępność systemu	1015
50.1.3. Ilościowa miara dostępności i jej obliczanie	1016
50.2. Odmładzanie oprogramowania	1017
50.2.1. Powody degradacji i sposoby odmładzania oprogramowania	1017
50.2.2. Wpływ odmładzania na dostępność systemu	1018
51. Metryki dla procesu testowego	1020
52. Metryki zadowolenia klienta	1024
52.1. Proces pomiaru zadowolenia klienta	1024
52.1.1. Wybór metody i sposobu przeprowadzenia badania	1024
52.1.2. Opracowanie ankiety/kwestionariusza lub innego narzędzia badań	1026
52.1.3. Wybór metody próbkowania	1026
52.1.4. Wybór rozmiaru próby	1027
52.1.5. Zebranie, opracowanie i analiza danych	1029
52.2. Dział wsparcia klienta	1031
53. Sposób prezentowania danych	1033
53.1. Prezentowanie danych graficznych	1033
53.2. Prezentowanie metryk	1036
Dodatek A. Specyfikacja programu ELROJ	1039
Dodatek B. Normy i standardy	1043
Dodatek C. Matematyczny i teoretyczno-informatyczny	1047
Dodatek D. Informacja o certyfikacji i egzaminach	1059
Bibliografia	1061
Skorowidz	1078