

Table des matières

Remerciements	1
Avant-propos	3
Chapitre 1. Introduction	5
1.1. Définition	5
1.2. Pourquoi et pour qui est cet ouvrage ?	7
1.2.1. Pourquoi cet ouvrage ?	7
1.2.2. Pour qui est cet ouvrage ?	8
1.2.3. Organisation de cet ouvrage	9
1.3. Exemples	10
1.4. Limitations	11
1.5. Pourquoi tester ?	12
1.6. MOA et MOE	13
1.7. Défis principaux	13
1.7.1. Complexité accrue	14
1.7.1.1. Simple	15
1.7.1.2. Compliqué	15
1.7.1.3. Complexe	16
1.7.1.4. Chaotique	16
1.7.2. Taux d'échecs important	16
1.7.3. Visibilité limitée	17
1.7.4. Multisource et complexité	17
1.7.5. Politique multi-entreprises	18
1.7.5.1. Durée de vie	18
1.7.5.2. Mode de conception	19

1.7.5.3. Criticité	19
1.7.5.4. Responsabilités	19
1.7.5.5. Confidentialité	20
1.7.6. Multiples niveaux de test	20
1.7.7. Suivi de contrats, mesures, <i>reporting</i> , pénalités	23
1.7.7.1. SLA et pénalités	23
1.7.7.2. Mesures, métriques et <i>reporting</i>	24
1.7.8. Intégration et environnements de test	24
1.7.9. Disponibilité des composants	25
1.7.10. Combinatoire et couverture	26
1.7.11. Qualité des données	26
1.7.12. Flux, pivots et conversions de données	27
1.7.13. Évolutions et transition	28
1.7.14. Historique et historisation	29
1.7.15. Charlatans	29
Chapitre 2. Cycles de développement logiciel	33
2.1. Cycles de développement séquentiel	34
2.1.1. Cascade	35
2.1.2. Cycle en V	36
2.1.3. Spirale et prototypage	38
2.1.4. Challenges des développements séquentiels	40
2.2. Cycles de développement incrémental	40
2.2.1. Challenges des développements incrémentaux	41
2.3. Cycles de développement agile	41
2.3.1. Manifeste Agile	43
2.3.2. eXtreme Programming	45
2.3.3. Défis des cycles itératifs	47
2.3.3.1. Schéma Scrum	48
2.3.3.2. Schéma Nexus	50
2.3.3.3. Schéma SAFe	51
2.3.3.4. Quelques étapes SAFe	53
2.3.4. Le <i>lean</i>	54
2.3.5. DevOps et livraisons continues	56
2.3.6. Challenges des développements agiles	59
2.4. Acquisition	61
2.5. Maintenances	61
2.6. Et la réalité dans tout cela ?	63

Chapitre 3. Politique et stratégie de test	67
3.1. Politique de test	67
3.1.1. Rédaction de la politique de test	68
3.1.2. Périmètre de la politique de test	68
3.1.3. Applicabilité de la politique de test	69
3.2. Stratégie de test	69
3.2.1. Contenu d'une stratégie de test	71
3.2.2. Stratégies de test et taylorisme	73
3.2.3. Types de stratégie de test	74
3.2.3.1. Stratégies analytiques	74
3.2.3.2. Stratégies méthodiques	76
3.2.3.3. Stratégies réactives	76
3.2.3.4. Stratégies consultatives	77
3.2.4. Stratégie de test et environnements	78
3.3. Choix d'une stratégie de test.	79
3.3.1. Complétude de la stratégie	79
3.3.2. Points importants de la stratégie	80
3.3.3. Suivi de la stratégie	81
3.3.4. <i>Shift left</i> , coûts et délais	83
3.3.5. Stratégie « optimale »	84
3.3.6. Assurer le succès	86
3.3.7. Pourquoi plusieurs itérations de test ?	86
3.3.8. Prévision d'avancement	89
3.3.9. Améliorations continues	90
 Chapitre 4. Méthodologies de test.	 91
4.1. Tests basés sur les risques (RBT)	91
4.1.1. Hypothèses du RBT	92
4.1.2. Méthodologie RBT	92
4.1.2.1. Effort de test et niveau de risque	94
4.1.2.2. Risques et dépendances	96
4.1.2.3. Couverture des risques.	97
4.1.3. RBT <i>versus</i> RRBT	98
4.1.4. Réactions face aux risques.	99
4.1.5. Calcul du risque	99
4.1.5.1. Mesure du risque (valeurs 1 à 4).	100
4.1.5.2. Mesure du risque (valeurs 1 à 5).	102
4.1.6. Synthèse RBT	106

4.1.7. Ouvrages supplémentaires	107
4.2. Tests basés sur les exigences (TBX)	107
4.2.1. Hypothèses du TBX	108
4.2.2. Méthodologie TBX	109
4.2.3. Mode de calcul TBX	109
4.2.3.1. Propositions de priorisation	109
4.2.3.2. Charge de travail <i>versus</i> impact métier	110
4.2.4. Synthèse TBX	110
4.3. Tests basés sur les standards (TBS) et tests systématiques	111
4.3.1. Hypothèses du TBS	111
4.3.2. Mode de calcul TBS	111
4.3.3. Synthèse TBS	112
4.4. Tests basés sur les modèles (MBT)	112
4.4.1. Hypothèses du MBT	113
4.4.2. Mode de calcul MBT	114
4.4.3. Synthèse MBT	114
4.5. Tests dans les méthodologies agiles	115
4.5.1. Méthodologies de test agiles ?	115
4.5.2. Couverture des tests	116
4.5.2.1. Classes, objets et méthodes	116
4.5.2.2. État des objets et transitions d'états	117
4.5.2.3. Tests unitaires	117
4.5.2.4. Tests d'intégration	118
4.5.2.5. Intégration continue	119
4.5.2.6. Tests d'acceptation automatisés	120
4.5.2.7. Tests des systèmes	122
4.5.2.8. Tests des API	122
4.5.2.9. Tests et containers	123
4.5.3. Hypothèses	123
4.5.3.1. eXtreme Programming	123
4.5.3.2. Scrum et Nexus	124
4.5.3.3. SAFe	124
4.5.3.4. Kanban	124
4.5.4. Mode de calcul	125
4.5.4.1. eXtreme Programming	125
4.5.4.2. Scrum et Nexus	126
4.5.4.3. SAFe	126
4.5.4.4. Kanban	126
4.5.5. Synthèse	127
4.6. Choix d'une méthodologie multiniveau	127

4.6.1. Hypothèses	128
4.6.2. Mode de calcul	129
4.7. De la conception aux livraisons	130
Chapitre 5. Caractéristiques qualité	133
5.1. Caractéristiques qualité produit	134
5.2. Qualité en utilisation	137
5.3. Qualité pour les acquéreurs	138
5.4. Qualité pour les fournisseurs.	138
5.5. Qualité pour les utilisateurs	139
5.6. Impact de la qualité sur la criticité et la priorité	140
5.7. Démontrer les caractéristiques qualité	140
5.7.1. Deux écoles	140
5.7.2. Preuves IADT	141
5.7.3. Autres réflexions	142
Chapitre 6. Niveaux de test	145
6.1. Éléments génériques d'un niveau de test	147
6.1.1. Impacts des cycles de développement	148
6.1.2. Méthodes et techniques	149
6.1.3. Principes fondamentaux	149
6.1.3.1. Préparation (<i>setup</i>)	150
6.1.3.2. Exécution (<i>run</i>)	151
6.1.3.3. Nettoyage (<i>teardown</i>)	152
6.2. Tests unitaires	152
6.3. Tests d'intégration de composants	154
6.3.1. Types d'interfaces à intégrer	155
6.3.2. Challenge des intégrations.	156
6.3.3. Modèles d'intégration	157
6.3.4. Tests d'intégration matériel-logiciel	158
6.4. Tests des composants	159
6.5. Tests d'intégration de composants plus larges	159
6.6. Tests système.	161
6.7. Tests d'acceptation ou « recette » fonctionnelle	163
6.8. Particularités de certains systèmes	164
6.8.1. Systèmes à sécurité critique	164
6.8.2. Systèmes aéronautiques	165
6.8.3. Confidentialité et sécurisation des données.	165

Chapitre 7. Documentation des tests	167
7.1. Objectifs de la documentation	168
7.2. Plan de construction de la conformité (PCC)	169
7.3. Articulation de la documentation de test.	169
7.4. Politique de test	170
7.5. Stratégie de test	171
7.6. Plan de test maître (PTM)	172
7.7. Plan de test de niveau	175
7.8. Dossier de test	175
7.9. Spécifications de cas de test	176
7.10. Spécifications de procédure de test	176
7.11. Spécifications de données de test	177
7.12. Spécifications d'environnement de test.	177
7.13. <i>Reporting</i> et suivi d'avancement	178
7.14. Documentation de projet	178
7.15. Autres livrables.	179
Chapitre 8. Reporting	181
8.1. Introduction.	181
8.2. Parties prenantes	183
8.3. Qualité du produit	184
8.4. Coût des anomalies	185
8.5. Périodicité du <i>reporting</i>	186
8.6. Avancement des tests et interprétation	186
8.6.1. Couverture des exigences	187
8.6.2. Couverture des risques	188
8.6.2.1. Top 10	189
8.6.2.2. Comparaison des projets	189
8.6.2.3. Comparaison des composants	189
8.6.3. Couverture des composants ou des fonctionnalités	191
8.7. Avancement et anomalies	192
8.7.1. Découverte des anomalies	192
8.7.2. Correction des anomalies	193
8.7.3. <i>Backlog</i> des anomalies	194
8.7.4. Nombre de réouvertures d'anomalies	195
8.8. Efficacité et efficacité des activités de test	196
8.9. Améliorations continues	197
8.9.1. Mise en place d'améliorations continues	197
8.9.1.1. Identification des points d'amélioration	197

8.9.1.2. Mesure des objectifs d'amélioration, KPI	198
8.9.1.3. Définition des actions d'amélioration.	199
8.9.1.4. Mesure des améliorations	199
8.10. Points d'attention du <i>reporting</i>	200
8.10.1. Audience	200
8.10.2. Utilisation.	201
8.10.3. Impartialité	202
8.10.4. Évolution du <i>reporting</i>	202
8.10.5. <i>Reporting</i> Scrum.	203
8.10.6. <i>Reporting</i> Kanban	204
8.10.7. <i>Reporting</i> de conception des tests.	204
8.10.8. <i>Reporting</i> d'exécution	205
8.10.9. <i>Reporting</i> des anomalies logicielles	206
8.10.10. <i>Reporting</i> d'avancement TAU/UAT	209
8.10.11. <i>Reporting</i> pour les parties prenantes	210
8.10.11.1. Hiérarchie.	210
8.10.11.2. Développement	210
8.10.11.3. Utilisateurs	211
Chapitre 9. Techniques de test	213
9.1. Typologies de test	213
9.1.1. Tests statiques et revues	214
9.1.2. Tests non fonctionnels	214
9.2. Introduction aux techniques de test.	215
9.3. CRUD	216
9.4. Chemins (<i>path</i>).	216
9.4.1. Fonctionnement	218
9.4.1.1. Chemins fonctionnels (boîte noire)	218
9.4.1.2. Chemins techniques (boîte blanche).	220
9.4.2. Couverture	220
9.4.3. Limitations et risques.	221
9.5. Test des partitions d'équivalence (EP)	221
9.5.1. Objectifs	223
9.5.2. Fonctionnement	224
9.5.3. Couverture	224
9.5.4. Limitations et risques.	224
9.6. Test des valeurs limites (BVA)	224
9.6.1. Objectifs	225
9.6.2. Fonctionnement	225

- 9.6.3. Couverture 225
- 9.6.4. Limitations et risques. 225
- 9.7. Test des tables de décision (DTT) 225
 - 9.7.1. Objectifs 226
 - 9.7.2. Fonctionnement 226
 - 9.7.2.1. Exemple 226
 - 9.7.3. Couverture 229
 - 9.7.4. Limitations et risques. 229
- 9.8. Test des cas d'utilisation (UCT). 229
 - 9.8.1. Objectifs 229
 - 9.8.2. Fonctionnement 230
 - 9.8.3. Couverture 231
 - 9.8.4. Limitations et risques. 231
- 9.9. Combinaison de données (DCOT) 231
 - 9.9.1. Objectifs 231
 - 9.9.2. Fonctionnement 231
 - 9.9.3. Couverture 232
 - 9.9.4. Défi 232
- 9.10. Cycle de vie des données (DCYT) 232
 - 9.10.1. Objectifs 232
 - 9.10.2. Fonctionnement 232
 - 9.10.3. Couverture 233
 - 9.10.4. Défi 233
- 9.11. Test exploratoire (ET). 233
 - 9.11.1. Objectifs 234
 - 9.11.2. Fonctionnement 234
 - 9.11.3. Couverture 234
 - 9.11.4. Limitations et risques. 235
- 9.12. États et transitions (STT) 236
 - 9.12.1. Objectifs 236
 - 9.12.2. Fonctionnement 236
 - 9.12.3. Couverture 237
- 9.13. Tests du cycle de processus (PCT) 237
 - 9.13.1. Objectifs 237
 - 9.13.2. Fonctionnement 237
 - 9.13.3. Couverture 238
 - 9.13.4. Limitations et risques. 238
- 9.14. Test de vie réelle (RLT). 239
 - 9.14.1. Objectifs 239
 - 9.14.2. Fonctionnement 239
 - 9.14.3. Couverture 240

9.14.4. Limitations et risques	240
9.15. Tests particuliers	241
9.15.1. Tests de (non-)régression (TNR)	241
9.15.2. Tests automatisés	243
9.15.3. Tests de performances	244
9.15.4. Tests de sécurité	244
9.16. Explosion combinatoire	246
9.16.1. Tableaux orthogonaux (OAT)	246
9.16.2. Arbres de classification (CTT)	247
9.16.3. Tests des domaines (DOM)	250
9.16.4. <i>Built in</i> tests (BIT, IBIT, CBIT et PBIT)	250
Chapitre 10. Tests statiques, revues et inspections	253
10.1. Qu'est-ce donc ?	255
10.2. Revues ou test ?	256
10.2.1. Qu'est-ce qu'une revue ?	256
10.2.2. Que peut-on soumettre à des revues ?	256
10.3. Types et formalisme des revues	257
10.3.1. Revues informelles ou <i>ad hoc</i>	259
10.3.2. Revues techniques.	259
10.3.3. Revues basées sur des <i>checklists</i>	260
10.3.4. Revues basées sur des scénarios	261
10.3.5. Revues basées sur des perspectives (PBR)	261
10.3.6. Revues basées sur les rôles.	261
10.3.7. Relectures techniques.	261
10.3.8. Inspections	262
10.3.9. Revue de jalon	262
10.3.10. Revue de pairs	262
10.4. Mise en œuvre des revues	263
10.5. <i>Checklists</i> de revues	263
10.5.1. Revues et point de vue	264
10.5.2. <i>Checklist</i> pour revue de spécifications ou d'exigences	264
10.5.3. <i>Checklist</i> pour revue d'architecture.	266
10.5.4. <i>Checklist</i> pour revue de conception de haut niveau	268
10.5.5. <i>Checklist</i> pour revue de conception détaillée (CDR)	269
10.5.6. <i>Checklist</i> pour revue de code	271
10.6. Taxonomies de défauts	273
10.7. Rentabilité des revues	274
10.8. Analyses de sécurité (<i>safety</i>)	275

Glossaire	277
Bibliographie	285
Index	293
Sommaire de <i>Les tests de logiciels 1</i>	297
Sommaire de <i>Les tests de logiciels 3</i>	299