

Table des matières

Avant-propos	1
Chapitre 1. Principes fondamentaux.	5
1.1. Pourquoi les tests sont-ils nécessaires ? (FL 1.2).	5
1.1.1. Contexte des systèmes logiciels	5
1.1.2. Cause des défauts logiciels	7
1.1.3. Rôle du test dans la vie des logiciels.	9
1.1.4. Tests et qualité.	9
1.1.5. Un peu de terminologie	10
1.2. Qu'est-ce que le test ? (FL 1.1)	11
1.2.1. Origine des défauts	12
1.2.2. Objectifs habituels des tests	13
1.2.3. Exemples d'objectifs de test.	13
1.2.4. Test et débogage	14
1.3. Paradoxes et grands principes (FL 1.3)	15
1.3.1. Démontrer la présence de défauts	15
1.3.2. Impossibilité de tests exhaustifs	15
1.3.3. Tester le plus tôt possible	16
1.3.4. Agrégation des défauts.	16
1.3.5. Paradoxe du pesticide	17
1.3.6. Dépendance envers le contexte	17
1.3.7. L'absence de défauts	18
1.4. Processus de test fondamentaux (FL 1.4)	18
1.4.1. Planification	19
1.4.2. Contrôle	20
1.4.3. Analyse de la base de test	20

1.4.4. Conception des tests	22
1.4.5. Implémentation des tests	23
1.4.6. Exécution des tests	24
1.4.7. Analyse des critères de sortie	25
1.4.8. <i>Reporting</i>	25
1.4.9. Activités de clôture	26
1.5. Psychologie des tests (FL 1.5).	27
1.5.1. Niveaux d'indépendance.	28
1.5.2. Adaptation aux objectifs	29
1.5.3. Testeur : destructif ou constructif ?	30
1.5.4. Compétences relationnelles	30
1.5.5. Besoin de modifier les perspectives	31
1.6. Synthèse du chapitre	32
1.7. Exemples de questions d'examens	33

Chapitre 2. Les tests dans le cycle de vie des logiciels 39

2.1. Modèles de cycles de développement (FL 2.1).	39
2.1.1. Modèles séquentiels	40
2.1.1.1. Modèle en cascade	40
2.1.1.2. Modèle en V.	41
2.1.1.3. Modèle en W et en VVV	42
2.1.2. Modèles itératifs.	42
2.1.2.1. Modèle en spirale.	44
2.1.3. Modèle incrémental.	45
2.1.4. RAD.	48
2.1.5. Méthodes agiles	49
2.1.5.1. EVO	50
2.1.5.2. Scrum.	51
2.1.5.3. Développement dirigé par les tests (TDD).	53
2.1.6. Sélection du modèle de développement	53
2.1.7. Positionnement des tests	55
2.2. Niveaux de tests (FL 2.2).	56
2.2.1. Test de composants	57
2.2.2. Test d'intégration	58
2.2.2.1. Intégration de type <i>Big Bang</i>	59
2.2.2.2. Intégration de bas en haut	59
2.2.2.3. Intégration de haut en bas	59
2.2.2.4. Autres types d'intégration.	60
2.2.3. Tests système	60
2.2.4. Tests d'acceptation	62

2.2.5. Autres niveaux	64
2.3. Types de tests (FL 2.3)	64
2.3.1. Tests fonctionnels	66
2.3.2. Tests non fonctionnels	67
2.3.3. Tests basés sur la structure ou l'architecture du logiciel	70
2.3.4. Tests liés aux changements	71
2.3.5. Comparaisons et exemples	72
2.4. Tests et maintenance (FL 2.4)	73
2.4.1. Contexte de maintenance	74
2.4.2. Maintenance évolutive	75
2.4.3. Maintenance corrective	76
2.4.4. Mise au rebut et remplacement	76
2.4.5. Politiques de tests de régression	77
2.4.5.1. Efficacité des tests de régression.	78
2.4.5.2. Politique de stratégie de régression	79
2.4.6. Contexte d'acceptation et de validation de SLA	80
2.5. Les oracles	80
2.5.1. Problématique des oracles	81
2.5.2. Sources d'oracles	81
2.5.3. Utilisation d'un oracle	82
2.5.3.1. Préconditions	82
2.5.3.2. Postconditions.	82
2.6. Cas particuliers.	83
2.6.1. Tests de performances	83
2.6.2. Tests de maintenabilité.	83
2.7. Synthèse du chapitre	84
2.8. Exemples de questions d'examens	85

Chapitre 3. Techniques statiques 91

3.1. Techniques statiques et processus de test (FL 3.1)	91
3.2. Processus de revue (FL 3.2)	94
3.2.1. Types de revues	95
3.2.1.1. Revues informelles.	95
3.2.1.2. Revues de gestion	95
3.2.1.3. Revues techniques	97
3.2.1.4. Inspections.	97
3.2.1.5. Relectures techniques	98
3.2.1.6. Audits	99
3.2.2. Rôles et responsabilités lors de revues.	100
3.2.2.1. Revues de gestion	100

3.2.2.2. Revues techniques	101
3.2.2.3. Inspections.	101
3.2.2.4. Relectures techniques	102
3.2.2.5. Audits	102
3.2.3. Phases des revues	103
3.2.3.1. Revues de gestion	104
3.2.3.2. Revues techniques	106
3.2.3.3. Inspections.	108
3.2.3.4. Relecture technique	111
3.2.3.5. Audits	113
3.2.4. S’assurer du succès des revues	116
3.2.5. Comparaison des types de revues	118
3.3. Analyse statique outillée	120
3.3.1. Types d’analyses statiques.	120
3.3.1.1. Modélisation	121
3.3.1.2. Analyse statique du code	122
3.3.1.3. Analyse d’architecture.	123
3.3.2. Types de défauts identifiables.	126
3.3.3. Analyse du flot de données	127
3.3.3.1. Modéliser le flot de données	129
3.3.3.2. Tester avec le flot de données	130
3.3.3.3. Hypothèses et limitations	132
3.4. Valeur ajoutée des activités statiques.	132
3.5. Synthèse du chapitre	133
3.6. Exemples de questions d’examens	134

Chapitre 4. Technique de conception de tests 139

4.1. Processus de développement des tests	141
4.1.1. Terminologie.	141
4.1.2. Traçabilité	142
4.2. Catégories de techniques de test (FL 4.1)	145
4.2.1. Boîte noire, blanche ou grise	145
4.2.2. Techniques basées sur l’expérience	147
4.2.3. Caractéristiques des tests	147
4.2.4. Limitations et hypothèses	148
4.2.4.1. Tests basés sur les spécifications	148
4.3. Techniques boîte noire (FL 4.2).	149
4.3.1. Partitions d’équivalence (FL 4.2.1)	150
4.3.1.1. Tester avec les partitions d’équivalence	154
4.3.1.2. Hypothèses et limitations	155

4.3.1.3. Intérêt des partitions d'équivalence	156
4.3.2. Analyse des valeurs limites (FL 4.2.2)	156
4.3.2.1. Tester avec les valeurs limites	159
4.3.2.2. Hypothèses et limitations	159
4.3.2.3. Intérêt des valeurs limites	159
4.3.3. Tables de décisions (FL 4.2.3)	160
4.3.3.1. Identification des conditions	161
4.3.3.2. Calcul du nombre de combinaisons et introduction des données	161
4.3.3.3. Tester avec les tables de décisions.	164
4.3.3.4. Hypothèses et limitations	165
4.3.3.5. Intérêt des tables de décisions	165
4.3.4. Autres techniques combinatoires	165
4.3.5. Diagrammes d'états et transitions d'états (FL 4.2.4)	166
4.3.5.1. Tester avec les transitions d'états	172
4.3.5.2. Hypothèses et limitations	173
4.3.5.3. Intérêt des transitions d'états	173
4.3.6. Cas d'utilisation (FL 4.2.5)	173
4.3.6.1. Tester avec les cas d'utilisation	174
4.3.6.2. Hypothèses et limitations	175
4.3.6.3. Intérêt des cas d'utilisation	175
4.3.7. Limitations et hypothèses	176
4.4. Activités basées sur la structure (FL 4.3)	176
4.4.1. Couverture des instructions (FL 4.3.1)	178
4.4.1.1. Tester avec la couverture des instructions	179
4.4.1.2. Hypothèses et limitations	181
4.4.1.3. Intérêt de la couverture des instructions	183
4.4.2. Couverture des branches et décisions (FL 4.3.2).	183
4.4.2.1. Tester avec la couverture des branches et des décisions . . .	184
4.4.2.2. Complexité cyclomatique	185
4.4.2.3. Hypothèses et limitations	186
4.4.2.4. Intérêt de la couverture des branches et des décisions	186
4.4.3. Autres types de couverture	187
4.4.3.1. Décisions et conditions	187
4.4.3.2. Tester avec la couverture MC/DC	188
4.4.3.3. Hypothèses et limitations	188
4.4.3.4. Intérêt de la couverture MC/DC	189
4.4.4. Limitations et hypothèses des tests basés sur la structure	189
4.4.5. Niveau de couverture et critères de sortie.	189
4.5. Techniques basées sur l'expérience (FL 4.4)	190
4.5.1. Attaques	190

- 4.5.1.1. Interfaces utilisateur 191
- 4.5.1.2. Données et calculs 191
- 4.5.1.3. Interfaces système 192
- 4.5.2. Taxonomies d’erreurs 192
- 4.5.3. Estimation d’erreurs et tests *ad hoc* 193
- 4.5.4. Tests exploratoires 193
 - 4.5.4.1. Test exploratoire basé sur des scénarios 194
- 4.5.5. Tests basés sur des *checklists* 195
- 4.5.6. Hypothèses et limitations 195
- 4.6. Quelle(s) technique(s) sélectionner ? 196
- 4.7. Synthèse du chapitre 197
- 4.8. Exemples de questions d’examens 198

Chapitre 5. Gestion des tests 207

- 5.1. Politiques d’organisation des tests (FL 5.1) 207
 - 5.1.1. Niveaux d’indépendance. 207
 - 5.1.2. Rôles et responsabilités 210
 - 5.1.3. Aspects humains et contractuels 212
- 5.2. Planifier et estimer la charge de test (FL 5.2). 213
 - 5.2.1. Activités d’organisation et d’évaluation. 216
 - 5.2.1.1. Facteurs liés aux processus 220
 - 5.2.1.2. Facteurs matériels 221
 - 5.2.1.3. Facteurs humains 221
 - 5.2.1.4. Facteurs liés au contexte 222
 - 5.2.1.5. Estimation basée sur les métriques 224
 - 5.2.1.6. Estimation basée sur l’expérience 225
 - 5.2.1.7. Estimation dans le monde agile 225
 - 5.2.2. Activités de planification et d’ordonnancement 226
 - 5.2.3. Documentation des tests 227
 - 5.2.3.1. Plan de test 228
 - 5.2.3.2. Dossier de test. 229
 - 5.2.3.3. Cas de test 230
 - 5.2.3.4. Procédures de test 230
 - 5.2.3.5. Rapport de test 231
 - 5.2.4. Critères de début et de fin d’activité de test (FL 5.2.3) 232
 - 5.2.4.1. Critères de début d’activités de test (DOR) 232
 - 5.2.4.2. Critères de continuation 233
 - 5.2.4.3. Critères de fin d’activités de test (DOD) 234
- 5.3. Suivi et contrôle des activités de test (FL 5.3) 236
- 5.4. *Reporting* 240

5.4.1. Que rapporter, à qui, comment ?	241
5.4.2. Statistiques et graphiques	243
5.5. Processus transverses	246
5.5.1. Définition des données de test	246
5.5.2. Gestion de configuration (FL 5.4)	247
5.5.3. Gestion des changements et modifications	248
5.6. Gestion des risques (FL 5.5).	248
5.6.1. Principes de gestion des risques	249
5.6.2. Risques projet et risques produit	253
5.6.3. Introduction à la gestion des risques	254
5.6.3.1. Mesure des fréquences.	254
5.6.3.2. Identifier les risques	255
5.6.3.3. Priorisation des risques	255
5.6.3.4. Les tests basés sur les risques	256
5.6.3.5. Priorisation tout au long du projet	256
5.6.3.6. Risques principaux	257
5.7. Gestion des anomalies ou défauts (FL 5.6)	257
5.7.1. Introduction à la gestion des anomalies	258
5.7.1.1. Cycle de vie des anomalies	258
5.7.1.2. Classification des anomalies	258
5.7.2. Identification des anomalies.	259
5.7.2.1. Taxonomies d'erreurs	260
5.7.2.2. Description des anomalies	260
5.7.2.3. Triage.	262
5.7.2.4. Attributs des rapports d'anomalies	262
5.7.3. Actions appliquées aux anomalies	264
5.7.4. Disposition des anomalies	264
5.8. Synthèse du chapitre	265
5.9. Exemples de questions d'examens	266

Chapitre 6. Outils et automatisation 277

6.1. Typologie d'outils (FL 6.1)	277
6.1.1. Types de classifications d'outils	278
6.1.2. Outils de support à la gestion des tests	278
6.1.3. Outils de support à la gestion des exigences	279
6.1.4. Outils de support aux tests statiques	279
6.1.5. Outils de modélisation	280
6.1.6. Outils de support à la conception des tests et de données de test.	280
6.1.7. Outils de support à l'exécution des tests	281

6.1.8. Outils de support à l'environnement des tests	281
6.1.9. Outils de comparaison des données de test	282
6.1.10. Outils de mesure de la couverture des tests	282
6.1.11. Autres outils de support des tests	283
6.2. Problèmes liés aux outils (FL 6.2)	283
6.2.1. Avantages et risques liés aux outils	284
6.2.2. Considérations spécifiques pour certains outils	285
6.2.2.1. Spécificités des outils d'exécution des tests	286
6.2.2.2. Spécificités des outils de gestion des tests	288
6.2.2.3. Spécificités des outils d'analyse statique	289
6.3. Sélection et introduction d'outil (FL 6.2.1)	289
6.3.1. Principes majeurs	290
6.3.2. Processus de sélection d'outil	291
6.3.2.1. Évaluation des objectifs	291
6.3.2.2. Sélection d'une présélection	291
6.3.2.3. Choix de l'outil	292
6.3.3. Implémentation d'un outil	293
6.3.4. Acheter ou concevoir ses outils ?	294
6.3.4.1. Logiciels libres	294
6.3.4.2. Logiciels du commerce	295
6.3.4.3. Développer ses propres outils	295
6.4. Synthèse du chapitre	295
6.5. Exemples de questions d'examens	296
Annexe 1. Examen blanc	299
Annexe 2. Corrigés des questions	311
Glossaire	315
Bibliographie	325
Index	329