

Inhoudsopgave

Voorwoord 9

Hoofdstuk 1 Introductie 11

- 1.1 Over softwareontwikkeling 11
- 1.2 Samenvatting 14
- 1.3 Vragen 14
- 1.4 Oefeningen 14

Hoofdstuk 2 Use cases 15

- 2.1 Inleiding 15
- 2.2 Een eenvoudige use case 15
- 2.3 Meer dan één use case 17
- 2.4 Use cases van Messenger 19
- 2.5 Oefeningen 22
- 2.6 Use cases van een studentenadministratie 23
- 2.7 Use-casebeschrijving 25
- 2.8 Use case en activiteitendiagram 28
- 2.9 Waarom zijn use cases belangrijk? 32
- 2.10 Drie toepassingen 32
- 2.11 De magnetron 32
- 2.12 De bibliotheek 33
- 2.13 Sokoban 36
- 2.14 Gebruik van een UML-tool 38
- 2.15 Referenties 39
- 2.16 Samenvatting 39
- 2.17 Vragen 40
- 2.18 Oefeningen 40

Hoofdstuk 3 Domeinklassen 43

- 3.1 Inleiding 43
- 3.2 Hoe vind ik domeinklassen? 43
- 3.3 Magnetron: analyse van de use-casebeschrijving 44
- 3.4 De bibliotheek 50
- 3.5 Oefening 51
- 3.6 Domeinklassen en de bibliotheek 51

- 3.7 Oefening 53
- 3.8 Associaties tussen de domeinklassen 53
- 3.9 Multipliciteiten 57
- 3.10 Oefeningen 60
- 3.11 Domeinmodel van bibliotheek met multipliciteiten 60
- 3.12 Objectendiagram 61
- 3.13 Domeinklassen van Sokoban 63
- 3.14 Samenvatting 65
- 3.15 Vragen 66
- 3.16 Oefeningen 66

Hoofdstuk 4 Klassen in UML en Java 69

- 4.1 Inleiding 69
- 4.2 Het diagram van een enkele klasse 69
- 4.3 Associatie tussen twee klassen 72
- 4.4 Multipliciteit 75
- 4.5 Oefeningen 78
- 4.6 Afhankelijkheid 79
- 4.7 Polymorfie 80
- 4.8 Oefeningen 90
- 4.9 Abstracte klassen 92
- 4.10 Oefening 94
- 4.11 Interfaces 95
- 4.12 Polymorfie met een interface 97
- 4.13 Oefening 103
- 4.14 Interface, abstracte klasse of beide? 103
- 4.15 Samenvatting 105
- 4.16 Vragen 106
- 4.17 Oefeningen 107

Hoofdstuk 5 Sequentie- en toestandsdiagrammen 109

- 5.1 Inleiding 109
- 5.2 De magnetron 109
- 5.3 Sequentiediagrammen 114
- 5.4 Sequentiediagram voor de magnetron 117
- 5.5 Oefening 128
- 5.6 Sequentiediagram voor Sokoban 128
- 5.7 Een toestandsdiagram 132
- 5.8 Samenvatting 135
- 5.9 Vragen 135
- 5.10 Oefeningen 135

Hoofdstuk 6 Testen 137

- 6.1 Inleiding 137
- 6.2 Het doel van testen 138
- 6.3 Het testen van een systeem 139
- 6.4 Het ontwerpen van tests 140
- 6.5 Code coverage 141
- 6.6 Paden testen 143
- 6.7 Oefeningen 148
- 6.8 Bepalen van de cyclomatische complexiteit 149
- 6.9 Oefeningen 155
- 6.10 Procescyclustest 156
- 6.11 JUnit 160
- 6.12 Testen met JUnit 162
- 6.13 De methoden van Assert 176
- 6.14 Referenties 178
- 6.15 Samenvatting 178
- 6.16 Vragen 179
- 6.17 Oefeningen 180

Hoofdstuk 7 Ontwikkelen 181

- 7.1 Inleiding 181
- 7.2 Sokoban, versie 1 181
- 7.3 Sokoban, versie 2 188
- 7.4 Sokoban, versie 3 193
- 7.5 Sokoban, versie 4 197
- 7.6 Sokoban, versie 5 202
- 7.7 Terugblik 209
- 7.8 Referenties 209
- 7.9 Samenvatting 209
- 7.10 Vragen 210
- 7.11 Oefeningen 210

Hoofdstuk 8 Model-view-controller 211

- 8.1 Inleiding 211
- 8.2 Een console-applicatie 211
- 8.3 Het model 213
- 8.4 Model en grafische gebruikersinterface 216
- 8.5 Observable-observer 222
- 8.6 Een MVC-implementatie 224
- 8.7 Een tweede view 228
- 8.8 MVC en Sokoban 232
- 8.9 Referenties 238

- 8.10 Samenvatting 238
- 8.11 Vragen 239
- 8.12 Oefeningen 239

Hoofdstuk 9 Ontwerppatronen 241

- 9.1 Inleiding 241
- 9.2 Command 241
- 9.3 Oefening 248
- 9.4 Memento 248
- 9.5 Oefening 250
- 9.6 Bescherming van de interne toestand 250
- 9.7 Command en Memento combineren 259
- 9.8 Packagediagram 261
- 9.9 Referenties 264
- 9.10 Samenvatting 264
- 9.11 Vragen 264
- 9.12 Oefeningen 266

Hoofdstuk 10 Persistentie 267

- 10.1 Inleiding 267
- 10.2 Tekstbestand 268
- 10.3 Oefeningen 271
- 10.4 Objecten wegschrijven 272
- 10.5 Oefening 275
- 10.6 Een relationele database 275
- 10.7 Het domein- en het ER-model 278
- 10.8 Oefening 280
- 10.9 Omzetten van klassendiagram naar relationeel schema 280
- 10.10 JDBC 284
- 10.11 Drielagenarchitectuur 285
- 10.12 Hibernate 285
- 10.13 Referenties 287
- 10.14 Samenvatting 287
- 10.15 Vragen 288

Index 289