



‘Bruikbare en direct toepasbare ideeën voor elke ontwikkelaar’

In ‘Growing Object-Oriented Software, Guided by Tests’ leggen TDD- (Test Driven Development) en mocking pioniers Steve Freeman en Nat Pryce uit hoe goed gebruik van TDD en mocktesten leidt tot betere software. Ze laten met behulp van overtuigende voorbeelden zien hoe deze technieken toegepast moeten worden in de praktijk. De voorbeelden zijn geschreven in Java en de schrijvers maken in hun testcode veelvuldig gebruik van de JUnit4, Hamcrest en jMock2 frameworks.

Volgens de schrijvers ontstaat goede en onderhoudbare object georiënteerde software door het schrijven van een falende test, voordat de bijbehorende productiecode geschreven wordt. Bij het schrijven van de productiecode zou men zich vervolgens moeten laten leiden door wat de (mock) testen te vertellen hebben. Wat Freeman en Pryce onder goede software verstaan, lichten ze toe aan het begin van het boek. Kort samengevat pleiten ze voor objecten die elkaar berichten sturen in een ‘tell, don’t ask’ stijl. Hun designfocus is dus sterk gericht op communicatie tussen objecten, op hoe groepen objecten samen een taak volbrengen. Het mocken van objecten is essentieel in het testen van deze communicatie.

Het boek bevat talloze adviezen en richtlijnen voor testen in het algemeen en TDD in het bijzonder. De adviezen zijn goed onderbouwd, soms met redeneringen, vaak ook met code. Het boek staat vol met expressieve code als: `assertThat(instruments, hasItem(instrumentWithPrice(greaterThan(81))))`; De schrijvers leggen uitvoerig uit hoe de lezer zelf ook zulke expressieve code kan produceren en geven keer op keer het goede voorbeeld. Gedurende een tiental hoofdstukken bouwen ze een illustratieve applicatie waarmee ze op overtuigende wijze laten zien hoe ze zich met behulp van de testen naar een goed design laten leiden. Ook de minder getrainde ontwikkelaar zal de werkwijze, designideeën en programmeerstijl van de schrijvers al lezende steeds beter weten te waarderen. Voor degenen die nog niet eerder met jMock2 en Hamcrest gewerkt hebben is het raadzaam om eerst de appendices te lezen waarin deze frameworks kort maar krachtig toegelicht worden.

Van schoolverlaters tot ontwikkelaars met jarenlange ervaring: iedereen zal bruikbare en direct toepasbare ideeën in dit boek vinden. Ontwikkelaars die op zoek zijn naar de ultieme testnaam conventie, naar manieren om complexe testdata onderhoudbaar op te zetten, naar tips om expressievere (test)code te schrijven, naar meer kennis over het testen van threads en asynchrone code, naar goede voorbeelden van het gebruik van jMock2, JUnit4 en Hamcrest of naar algemene testadviezen en richtlijnen, zullen in dit boek uitstekende antwoorden vinden.

De test- en codeveteranen Freeman en Pryce hebben met dit boek een belangrijke bijdrage geleverd aan ons vakgebied. Op overtuigende wijze tonen ze aan dat de belangrijkste problemen rondom mocking en TDD opgelost zijn. Het zijn inmiddels goed gedocumenteerde, gepolijste technieken geworden. Dit boek zal zeker bijdragen aan de kennis over en verspreiding van deze technieken.

TDD en mocking zijn technieken die niet mogen ontbreken in de gereedschapskist van een (java)ontwikkelaar. Voor de ontwikkelaars onder ons die zich voorbij het basisoniveau in deze technieken willen verdiepen en verbeteren is dit boek een absolute aanrader. «

Titel: Growing Object-Oriented Software
Auteur: Steve Freeman en Nat Pryce
ISBN-13: 978-0-321-50362-6
Uitgever: Addison-Wesley

Waardering



Maarten Metz
is software ontwikkelaar bij Logica.