



‘Een vrij triviale gereedschapskist’

Als softwareontwikkelaar vind je het belangrijk om ‘goede’ code te schrijven. Maar wat is goede code? Wat maakt code slecht en wat maakt het goed? Vaak worden ontwerppatronen toegepast om generieke problemen op te lossen. Dit is echter niet altijd mogelijk. Er bestaat niet voor iedere situatie waar je tijdens het programmeren mee te maken krijgt een ontwerppatroon, maar wat doe je dan? Het boek ‘Implementation Patterns’ van Kent Beck beoogt de nodige handvatten te bieden voor het schrijven van ‘goede’ code.

De essentie van het boek is gericht op het schrijven van leesbare – en daarmee ‘relatief goedkoop’ onderhoudbare – code, waarbij naamgeving centraal staat. Er zijn twee belangrijke elementen waar je je als programmeur aan dient te houden voor het schrijven van leesbare code. Deze zijn een drietal waarden (values) en een zestal principes (principles).

De drie waarden (values) die je als programmeur hoog in het vaandel dient te hebben zijn communicatie, eenvoud en flexibiliteit. Via code communiceer je met andere ontwikkelaars, maar ook met jezelf als je later je eigen code weer verder uitbreidt of aanpast. Na een eerste oplevering van de code gaat de meeste tijd – en dus kosten – zitten in het onderhoud. Als je code eenvoudig te volgen is, niet complex is opgezet en makkelijk aanpasbaar dan zal dit ten goede komen aan de totale onderhoudskosten van code wijzigingen. Het boek beschrijft de drie waarden als spelregels waar je je aan moet houden voor het schrijven van dergelijke code, maar ook hoe je dit in de praktijk doet.

De zes principes zijn meer specifiek gericht op programmeren. Denk er hierbij aan dat een methode geen onverwachte bijwerkingen mag hebben; dubbele code moet worden voorkomen; gerelateerde logica en data dicht bij elkaar moet worden gehouden; ook code met dezelfde wijzigingsfrequentie moet bij elkaar worden gehouden; code symmetrisch leesbaar moet zijn; waar mogelijk code declaratief uitdrukken (annotaties);

De beschreven waarden en principes komen ongetwijfeld als vanzelfsprekend over op iedere lezer met

enige programmeerervaring. De eerste hoofdstukken van het boek gaan in op deze waarden en principes en op de motivatie erachter. De rest van de hoofdstukken gaan in op vrij triviale zaken die komen kijken bij het programmeren in een objectgeoriënteerde taal, met name Java. In feite wordt ingegaan op de basisaspecten van de Java-programmeertaal en deels de API. De structuren waarmee je een applicatie opbouwt: klassen, interfaces, collecties, overerving en afgeleide structuren hiervan passeren de revue en daar zit helaas weinig spanning in.

Aangezien het grootste deel van het boek aandacht besteed aan deze triviale dingen, is het minder waardevol voor de meer ervaren (Java) ontwikkelaar. Het kan van pas komen om de puntjes op de i te zetten van dingen waar je grotendeels al bekend mee bent. Dit terwijl de titel van het boek anders doet vermoeden. Het is een heel aardig boek om gelezen te hebben, maar absoluut geen must voor de doorgewinterde (Java) programmeur. «



Bob Schonenberg
is Software Ontwikkelaar
bij Info Support.

Titel: Implementation Patterns
Auteur: Kent Beck
ISBN-13: 978-0321413093
Uitgever: Addison-Wesley-Professional

Waardering

