



Precies volgens plan!

Succesvolle IT-projecten met heldere requirements

Mark Hoogveld
Jan Willem Knop
Marcel Schaar

Precies volgens plan!

Succesvolle IT-projecten met heldere requirements

Mark Hoogveld

Jan Willem Knop

Marcel Schaar



Inhoud

	Voorwoord	7
1	Succesvolle IT-projecten	11
1.1	Succesvolle IT-projecten	12
1.2	Big Hitters	14
1.3	Requirements en belanghebbenden	15
1.4	Requirements op niveau	18
1.5	Requirements engineering in de ideale wereld	19
1.6	Utopia	23
2	Requirements engineering in de echte wereld	25
2.1	Onvolledige en onjuiste doelstellingen	26
2.2	Oplossingsgericht denken	27
2.3	Belanghebbenden gemist	27
2.4	Geen tijd nemen	28
2.5	Geen belang hechten	28
2.6	Geen mandaat hebben	29
2.7	Andermans belang vertegenwoordigen	30
2.8	Tegenstrijdige belangen	30
2.9	Te veel bureaucratie	31
2.10	Geen hergebruik van requirements	32
2.11	Ontelbaar aantal requirements	32
2.12	Geen akkoord geven	33
2.13	Geen prioriteiten stellen	33

2.14	De Big Hitters in de echte wereld	34
3	Succes met requirements	37
4	Organiseer je proces	43
4.1	Duidelijke rollen en verantwoordelijkheden	43
4.2	Schoenmaker, blijf bij je leest!	45
4.3	Rolverdeling	47
4.4	Rolverdeling in de echte wereld	48
5	Wees pragmatisch	53
5.1	Het bepalen van de complexiteit	55
5.2	Pragmatiek op maat	57
5.3	Goed is goed genoeg	64
6	Wees probleemgericht	67
6.1	Zoek de kans	68
6.2	Formuleer doelen	71
6.3	Zoek de oplossing binnen de grenzen	73
7	Leg requirements vast	75
7.1	Sterk en compleet	76
7.2	Stop de verspilling	78
8	Betrek de business	81
9	Big Hitters	83
	Dankwoord	87
	Bijlage	89
	Over Valori	89
	Over de auteurs	91
	Referenties	93
	Index	95

Succesvolle IT-projecten

Organisaties zijn tegenwoordig in sterke mate afhankelijk van IT-systemen. Een organisatie als Artsen zonder Grenzen kan bijvoorbeeld zijn artsen niet bevoorraden zonder communicatiesysteem. Een luchtvaartmaatschappij ontvangt zijn grootste inkomsten met online boekingen en zelfs in de zorgsector, van oudsher een atechnische branche, rukt het elektronisch patiëntendossier op.

Ook veranderingen zijn in onze samenleving en binnen organisaties een constante. Deze veranderingen maken, breken en bevestigen het succes van een organisatie. Veranderingen in de IT hebben rechtstreeks invloed op de bedrijfsvoering van de organisatie en dragen daardoor bij aan de efficiëntie en het succes van de onderneming. Grote veranderingen vinden veelal plaats middels programma's en projecten. Die programma's en projecten zijn succesvol als ze leiden tot succesvolle veranderingen en daarmee bijdragen aan een succesvolle organisatie.

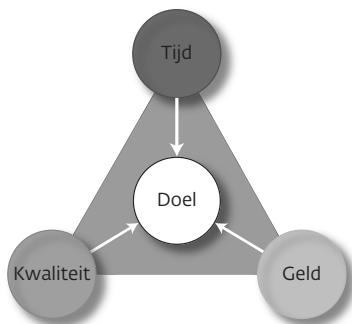
« **The only constant I am sure of is this accelerating rate of change – Peter Gabriel**

In dit boek nemen we projecten met een belangrijke IT-component onder de loep. We kijken wanneer deze succesvol zijn, welke fac-

toren dit succes bepalen en of succesvol requirements engineering hieraan bijdraagt. Deze succesfactoren zijn universeel toepasbaar. Ze komen voor in IT-projecten die volgens de ‘klassieke’ watervalmethodes of AGILE ontwikkelmethoden worden uitgevoerd. Ze zijn toepasbaar wanneer het gaat om het zelf bouwen van software, aanschaffen van standaardapplicaties of creëren van functionaliteit. Omdat deze succesfactoren voor al deze ontwikkelprocessen van toepassing zijn, maken we er in dit boek geen onderscheid tussen.

1.1 Succesvolle IT-projecten

We noemen een (IT-)project succesvol als het de informatievoorziening zodanig heeft veranderd dat deze de beoogde bijdrage aan de bedrijfsdoelstellingen levert. Dat deze bijdrage binnen de grenzen van tijd, geld en kwaliteit (inclusief de functionaliteit) gerealiseerd dient te worden is evident. We noemen dit de duivelsdriehoek.



Figuur 1.1 Duivelsdriehoek tijd, geld en kwaliteit (inclusief functionaliteit)

Bij bedrijfsdoelstellingen denken we bij organisaties vaak aan continuïteit. Maar je kunt ook andere doelen op het oog hebben, zoals voldoende winst, het vergroten van het marktaandeel, een kortere

‘time to market’ voor nieuwe producten of bijvoorbeeld een kostenreductie bij de productie van bestaande producten.



Scannen en geautomatiseerd verwerken

Bij een grote zorgverzekeraar is een project uitgevoerd om polisaanvragen te scannen om zo een kostenreductie te realiseren en in piekperiodes de maximale verwerkingsduur niet te overschrijden.

Na de eerste iteratie bleek dat 60 procent van de polisaanvragen aanmerkelijk sneller verwerkt kon worden. Door meer aandacht te besteden aan het controleproces, waarbij een medewerker de twijfelgevallen controleert en waar nodig corrigeert, is het percentage formulieren dat sneller verwerkt kan worden opgekrikt naar 90-95 procent. Deze verbetering werd vooral bereikt dankzij het plaatsen van grote breedbeeldschermen waarop de gescande aanvraag en de controlegegevens naast elkaar zichtbaar zijn.

Het resultaat was dat de polissen die aan het einde van de keten worden verstuurd, ook in de eindejaarspiek binnen de norm voor maximale verwerkingsduur vielen!

Bijkomend voordeel was dat er minder klachten over te laat afgeleverde polissen binnenkwamen en dus besparingen op het callcenter zijn gerealiseerd.

Wanneer beschouwen we projecten als een succes?

In zijn promotieonderzoek naar *Succes and Failure Factors in ICT projects: A Dutch Perspective* hanteert Aart J. van Dijk (2009) de volgende definitie, die wij graag overnemen. Een project is succesvol als:

- het doel van de verandering wordt bereikt door realisatie van de afgesproken kwaliteit (inclusief functionaliteit);
- de geplande doorlooptijd met maximaal 50 procent wordt overschreden, geaccordeerde scopewijzigingen uitgezonderd;
- de kosten voor de verandering met maximaal 50 procent worden overschreden, geaccordeerde scopewijzigingen uitgezonderd, de zogenaamde overrun (zie hoofdstuk 4).

1.2 Big Hitters

In zijn studie beschrijft Van Dijk de zogenaamde Big Hitters: de factoren van IT-projecten die bepalend zijn voor een succesvolle afloop. De studie vat succesfactoren samen die onder anderen naar voren zijn gebracht door Capers Jones, John Smith en Peter Noordam. Geef aandacht aan deze Big Hitters en je maakt grote kans op een succes, verwaarloos er één van, en de kans dat je project niet succesvol wordt is haast een zekerheid. De zeven Big Hitters zijn:

1. Goed projectmanagement
2. Realistische deadlines
3. Goede communicatie
4. Sterk en compleet requirementsdocument
5. Voldoende betrokkenheid van toekomstige gebruikers
6. Betrokkenheid en commitment van senior management
7. Voldoende professionaliteit (professionals)

Kijken we naar de lijst met Big Hitters, dan zien we dat requirements, het onderwerp van dit boek, expliciet genoemd worden. Maar ze spelen ook een cruciale rol bij alle andere Big Hitters. Waarom dat zo is, komt in de rest van dit boekje aan de orde. Voor dat we dat doen, leggen we echter eerst even uit wat we verstaan onder 'requirements' en 'requirements engineering'.

1.3 Requirements en belanghebbenden

Requirements zijn eisen die belanghebbenden stellen ten aanzien van een product. Dit kan een fysiek product zijn, maar ook een dienst, organisatie, proces, functionaliteit of systeem.

Requirements zijn niets anders dan een communicatiemiddel, een gezamenlijk vastgelegd referentiekader. Met requirements is het projectteam in staat om een eindproduct te realiseren dat voldoet aan de eisen van de belanghebbenden.

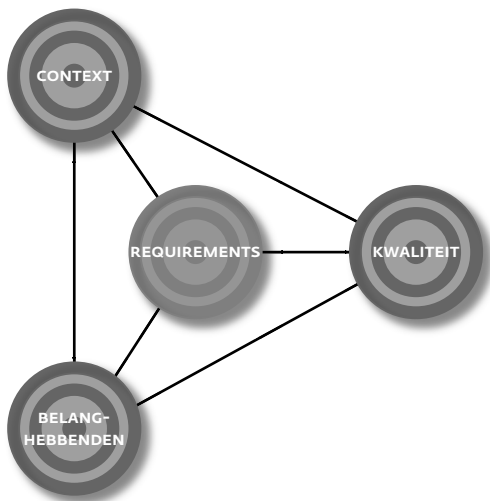
Om aantoonbaar te maken dat het eindproduct voldoet aan de requirements geven de belanghebbenden aan welke requirements zij dermate belangrijk vinden, dat ze in het eindproduct gecontroleerd moeten worden om tot acceptatie over te kunnen gaan. Deze requirements noemen we acceptatiecriteria.

Belanghebbenden zijn in te delen in drie groepen:

- huidige belanghebbenden;
- toekomstige belanghebbenden;
- transitiebelanghebbenden (projectbelanghebbenden).

Binnen genoemde groepen kan een deel van of kunnen alle onderstaande soorten belanghebbenden voorkomen. De gebruikers van het product en hun managers bijvoorbeeld hebben er belang bij dat het product aan hun verwachtingen voldoet, vaak wordt deze groep belanghebbenden aangeduid als de ‘business’. Even belangrijke groepen belanghebbenden bevinden zich echter binnen het IT- en Informatie Management (IM)-domein: de informatiearchitect of IT-architect bijvoorbeeld die er belang bij heeft dat het product voldoet aan de architectuurrichtlijnen, zoals de gegevensarchitectuur en het applicatielandschap. De derde groep zijn de testers, die willen dat het product testbaar is, of de beheerders, die beheer(s)baarheid nastreven.

Maar naast de *belanghebbenden* zijn er nog twee andere pijlers die belangrijk zijn om requirements goed te kunnen beschrijven: *kwaliteit* en *context* (zie figuur 1.2).



Figuur 1.2 Drie pijlers van requirements

Kwaliteit (inclusief functionaliteit)

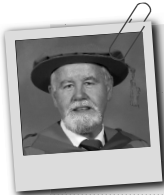
Requirements omschrijven wanneer het product, in de ogen van de belanghebbenden, goed genoeg is. Ze definiëren de kwaliteit. De kwaliteitseisen zijn meestal afgeleiden van de algemene kwaliteitseigenschappen die producten kunnen hebben. Zoals bijvoorbeeld functionaliteit, veiligheid, robuustheid, gebruikersvriendelijkheid, et cetera.

Context

Deze kwaliteitseigenschappen krijgen daarbij een specifieke betekenis binnen de context van het product. In de context van het ontwikkelen van een informatiesysteem kan de kwaliteitseigen-

schap 'veiligheid' bijvoorbeeld resulteren in het requirement 'Onbevoegden mogen geen toegang hebben tot het informatiesysteem'. Binnen de context van een gastank denken we veel meer aan explosiegevaar. De werkelijke invulling van de kwaliteitseigenschappen dient dus samen met de belanghebbenden te worden gedaan en te worden voorzien van hun context.

Geen nieuw probleem

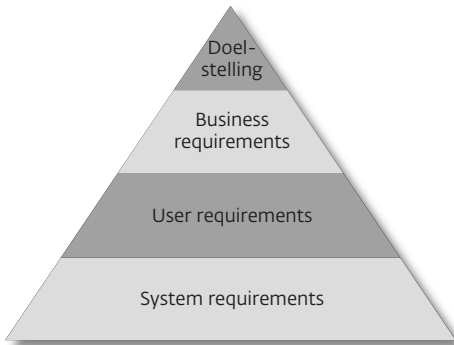


'Ik weet nog niet wat voor huis ik wil hebben, maar begint u maar vast.' Zou u zo'n opdracht geven of als aannemer aanvaarden? Vast niet. Waarom starten ICT-projecten dan soms wel op die manier? Het is niet altijd eenvoudig om wensen en eisen te formuleren. Een voorbeeld uit mijn 46-jarige ICT-praktijk: in 1970 kreeg ik als systeemontwerper bij de TU-Delft de uitdagende opdracht om een verkeersdetectiesysteem te bouwen. De requirements waren globaal en beperkt. Om inzicht te krijgen in die requirements heb ik mij, omdat ik nog geen rijbewijs had, laten rondrijden op allerlei wegen, weefvakken, enzovoorts. Dat gaf mij een goed beeld van waaraan het systeem zou moeten voldoen. Is er in 2011 veel veranderd? Niet echt. Zeer recentelijk ben ik als IT-auditor betrokken geweest bij een omvangrijk ICT-onderzoek in de publieke sector. Als onderzoekers hebben we enkele avond- en nachtdiensten meegedraaid en met de medewerkers de wensen/eisen bediscussieerd. Voor ICT-ers geldt mijns inziens ook dat de aangedragen requirements moeten worden bestudeerd op duidelijkheid, volledigheid en juistheid. Dit geldt eveneens voor moderne ontwikkelingsmethoden. Motto: 'Een opdrachtgever die niet weet wat hij wil, krijgt iets anders! Een ICT-er die niet precies weet wat hij moet maken, maakt het verkeerde product!'

Dr. Ir. Aart J. van Dijk EMITA RE is een gedreven informaticus met meer dan 40 jaar praktijkervaring. Hij heeft meerdere boeken op zijn naam staan en kan de praktijk vanuit de theorie onderbouwen.

1.4 Requirements op niveau

Requirements hebben een nare eigenschap. Ze kunnen voldoen aan alle in de vorige paragraaf genoemde eigenschappen en toch niet bruikbaar zijn om de gewenste verandering te bereiken. Daarvoor is er nog één eigenschap van essentieel belang: het niveau van het requirement. In theorie is het aantal niveaus afhankelijk van het aantal keren dat je de waaromvraag kunt beantwoorden. Waarom is het requirement van belang (de rationale, zie ook paragraaf 7.1)? In de praktijk is het handig om requirements in te delen in vier niveaus, waarbij de technische niveauperdeling even buiten beschouwing is gelaten.



Figuur 1.3 Niveaus van requirements

- Het hoogste niveau van requirements zijn *bedrijfsdoelstellingen*: wat wil het bedrijf bereiken?
- Een niveau daaronder treffen we zogenaamde *business requirements* aan: aan welke eisen moeten onze organisatie en bedrijfsprocessen voldoen om de bedrijfsdoelstellingen te realiseren?
- Weer een niveau lager zien we *user requirements*: aan welke eisen moet een deelproces voldoen en wat zijn de bijbehorende

functionaliteiten, bijvoorbeeld de marketing-, sales- of financiële processen van de organisatie?

- Nog een niveau lager zijn er de *system requirements*: aan welke eisen moeten de IT-systemen voldoen die de functionaliteiten moeten ondersteunen?

Nu we alle eigenschappen van requirements hebben besproken is het tijd om eens naar het voortbrengingsproces te kijken, requirements engineering.

Requirements engineering is het proces dat leidt tot de vastgelegde en vastgestelde requirements. In paragraaf 1.5 wordt dit proces aan de hand van een voorbeeld in de ideale wereld toegelicht.

1.5 Requirements engineering in de ideale wereld

In de ideale wereld is er een opdrachtgever die belang heeft bij het bereiken van een bepaalde doelstelling. De opdrachtgever geeft de requirements engineer opdracht om zijn doelstellingen en business requirements SMART¹ te beschrijven. De businessarchitect/businessanalist werkt diverse oplossingsrichtingen uit met advies. De opdrachtgever maakt een keuze en geeft de projectmanager de opdracht om een oplossing te realiseren waarmee de doelstelling bereikt kan worden. De projectmanager stemt de opdracht af met de opdrachtgever, formuleert een concrete projectopdracht en maakt een plan voor de uitvoering van het project. Een van de werkpakketten in het plan is het verzamelen van de wensen en eisen ten aanzien van de oplossing, eerst de user requirements en later de system requirements. De projectmanager geeft aan de requirements engineer de opdracht om dit werkpakket uit te voeren.

¹ SMART is een acroniem dat staat voor: *Specifiek, Meetbaar, Acceptabel, Realistisch en Traceerbaar.*

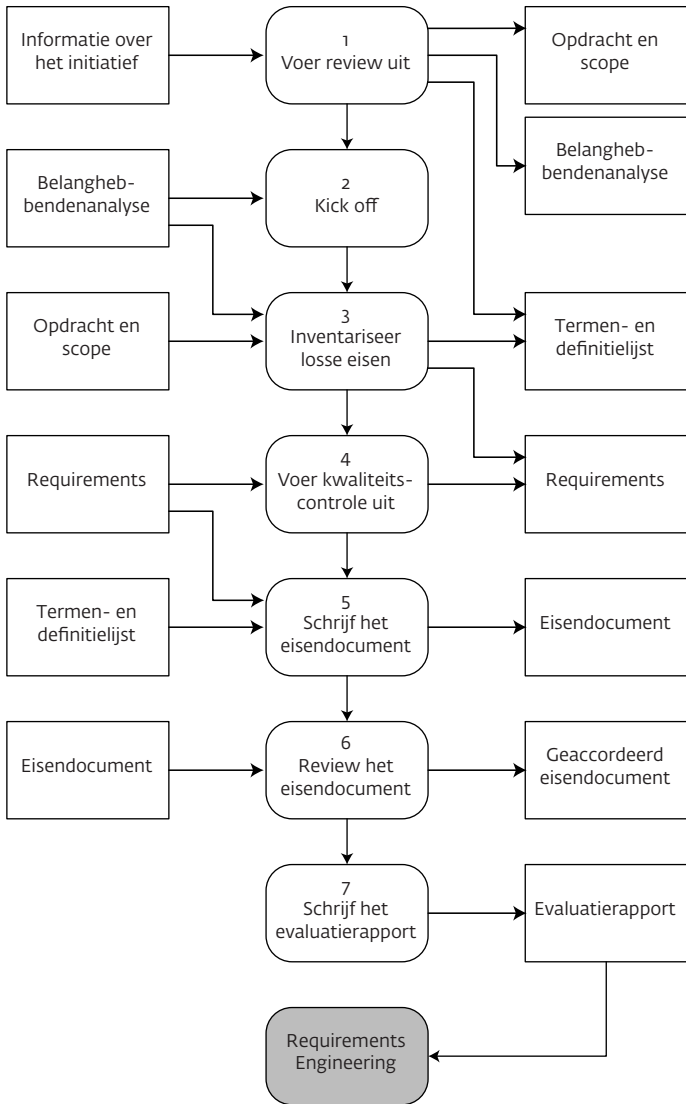
Zoals je kunt zien wordt de requirements engineer meerdere keren ingezet. Het proces dat hij doorloopt ziet er gelukkig steeds hetzelfde uit.

De requirements engineer neemt de opdracht, de ‘informatie over het initiatief’, aan en voert een review uit op deze opdracht. In overleg met de opdrachtgever wordt de opdracht en de scope afgestemd. Tevens start de requirements engineer met het inventariseren van de belanghebbenden en hun belang, de belanghebbendenanalyse. Tot slot legt de requirements engineer een termen- en definitielijst aan. Dit is *stap 1* ‘voer review uit’ in het REQUIRESmart® procesmodel (zie figuur 1.4).

Stap 2 is de ‘kick-off’. Op basis van de belanghebbendenanalyse vindt er een kick-off plaats met de belanghebbenden. Tijdens de kick-off zijn alle belanghebbenden aanwezig en wordt de opdracht van de requirements engineer uitgelegd. Na de kick-off zijn alle belanghebbenden op de hoogte van de opdracht en zien ze de toegevoegde waarde van requirements.

Stap 3. Vervolgens vormen de opdracht en de scope de basis waarop de belanghebbenden separaat geïnterviewd worden door de requirements engineer. Dit is *stap 3* ‘inventariseer losse eisen’. Tijdens de interviews verzamelt de requirements engineer alle requirements.

Een onafhankelijke belanghebbende die belang heeft bij de kwaliteit van de requirements, de kwaliteitsmanager, voert een kwaliteitscontrole uit. De kwaliteit is uitstekend. Dit is *stap 4* ‘voer kwaliteitscontrole uit’.



Figuur 1.4 REQUIRESmart® proces voor requirements engineering

In *stap 5* ‘schrijf het eisendocument’ voegt de requirements engineer de requirements van de afzonderlijke belanghebbenden samen en komt zo tot een volledige en juiste set requirements, het eisendocument. De requirements engineer presenteert het eisendocument aan de belanghebbenden en opdrachtgever en vraagt de groep om goedkeuring ervan.

Stap 6. Onder leiding van de kwaliteitsmanager reviewen de belanghebbenden en de opdrachtgever het eisendocument en keuren het goed. De requirements engineer levert het eisendocument op aan de projectmanager. Dit is *stap 6* ‘review het eisendocument’.

Stap 7. Tot slot organiseert de requirements engineer een evaluatie van het gevolgde proces en met de belanghebbenden. Dit is *stap 7* ‘schrijf het evaluatierapport’. De requirements engineer rapporteert over de positieve en verbeterpunten in het requirements engineeringproces aan de projectmanager. De projectmanager neemt het resultaat van het werkpakket in ontvangst en tekent het af.

De goedgekeurde set requirements vormt de basis voor de volgende fasen van het project. Ze zijn het uitgangspunt voor analyses, ontwerpen, de te bouwen oplossing, de acceptatiecriteria ten behoeve van testen, et cetera.

De requirements engineer speelt gedurende de vervolgfases een rol bij het overdragen van de requirements aan diegenen die deze nodig hebben voor het uitvoeren van hun werkpakketten.

Tevens beheert de requirements engineer de opgeleverde set van requirements. Hierdoor wordt geborgd dat de uiteindelijke oplossing voldoet aan de eisen van de belanghebbenden en het resultaat bijdraagt aan de doelstelling van de opdrachtgever.

1.6 Utopia

In de praktijk verloopt requirements engineering zelden zo utopisch als in de vorige paragraaf is beschreven. Reden om in de rest van dit boek stil te staan bij hoe requirements engineering er in de echte wereld uit ziet.

We weten inmiddels dat requirements engineering een belangrijk ingrediënt is van een succesvol project. In het voorbeeld uit de ideale wereld is al een aantal van de zogenoemde Big Hitters voorbijgekomen die bepalend zijn voor het succes van een project. Bijvoorbeeld een goed projectmanagement of de betrokkenheid van gebruikers.

Om als succesfactor gewicht in de schaal te leggen dient requirements engineering wel succesvol te worden uitgevoerd. In het volgende hoofdstuk gaan we daarom dieper in op de oorzaken waardoor requirements engineering in de praktijk minder vlekkeloos verloopt. In de hoofdstukken daarna lees je hoe je de belangrijkste valkuilen kunt omzeilen en hoe je er, mocht je er al ingestapt zijn, alsnog kunt uitklimmen.

IT-projecten zijn niet altijd zo succesvol als ze horen te zijn. Onduidelijke of onvolledige requirements zijn regelmatig de oorzaak van tegenvallende resultaten. Requirements vormen het uitgangspunt voor de gewenste oplossing. Een fout in dit uitgangspunt kan ertoe leiden dat er precies volgens plan wordt gebouwd, maar dat de oplossing toch tegenvalt.

Precies volgens plan! geeft de lezer de juiste handvatten om requirements engineering zo in te richten dat er een oplossing wordt gerealiseerd die rendement oplevert voor de business en die voldoet aan de verwachtingen. Dit boek bevat een helder overzicht van de problemen die de lezer in de praktijk kan tegenkomen, wat de oorzaken hiervan zijn en welke gevolgen deze hebben. Aan de hand van praktische tips en praktijkvoorbeelden leer je problemen vroegtijdig signaleren en voorkomen. Duidelijk wordt hoe Requirements Engineering kan bijdragen aan het succes van jouw project.

Precies volgens plan! is het tweede boek in de Valori-reeks: *De echte wereld*. Deze reeks laat zien dat je door de mens centraal te zetten, methoden en middelen met maximaal succes kunt inzetten. Eerder verscheen *De held die voor mijn nachtrust zorgt*.



Over de auteurs

Mark Hoogveld, Jan Willem Knop en Marcel Schaar zijn ervaren business analisten en requirements engineers en verstaan de kunst om organisaties te helpen hun werkelijke behoefte te vertalen naar goede IT-oplossingen.



ISBN 978 90 12 58306 0

NUR 980



ACADEMIC
SERVICE



9 789012 583060

www.academic-service.nl