

# Inhoud

Voorwoord 7

## DEEL 1 INTRODUCTIE

- 1 Inleiding 13
  - 1.1 Problemen in IT-projecten 16
- 2 De oorzaak achter de problemen 19
  - 2.1 Vier projectdimensies 20
    - 2.1.1 Gemeenschappelijk doel 20
    - 2.1.2 Juiste gereedschap 21
    - 2.1.3 Communicatie en samenwerking 22
    - 2.1.4 Complexiteit 23
  - 2.2 Wat kunt u verwachten in dit boek? 24

## DEEL 2 VIER DIMENSIES IN IT-PROJECTEN

- 3 Gemeenschappelijk doel 27
  - 3.1 Het ontbrekende doel 28
  - 3.2 Hoe formuleer ik een doel? 29
    - 3.2.1 Bedrijfsdoelstellingen 30
    - 3.2.2 IT-doelstellingen 32
    - 3.2.3 IT-strategie 33
    - 3.2.4 IT-architectuur 38
    - 3.2.5 IT-operatie 42
  - 3.3 Hoe wordt het doel een gemeenschappelijk doel? 43
  - 3.4 Conclusie gemeenschappelijk doel 44
- 4 Het juiste gereedschap 47
  - 4.1 De mythe van de softwarefabriek 48
  - 4.2 Projectmanagement en projectaanpak 52
  - 4.3 Projectondersteuning en faciliteiten 59
  - 4.4 Slim plannen 64

- 4.5 Functional Prototyping 71
- 4.6 Outsourcing en offshoring 76
- 4.7 Conclusie 82

## **5 Samenwerking en communicatie 83**

- 5.1 Vertrouwen 84
- 5.2 Gelijkwaardige relatie 85
- 5.3 Verborgene agenda's 92
- 5.4 Persoonlijke relaties 94
- 5.5 Preferred suppliers 95
- 5.6 Contractvormen 98
- 5.7 Gezamenlijke teams 102
- 5.8 Conclusie 104

## **6 Complexiteit 105**

- 6.1 Wat verstaan we onder complexiteit? 108
- 6.2 Complexiteit van organisatie 108
- 6.3 Complexiteit van oplossing 111
- 6.4 Conclusie 116

## **DEEL 3 VERTALING NAAR IT-PROJECTAANPAK**

## **7 Projectaanpak 121**

- 7.1 Fase 1: De briefingfase 123
- 7.2 Fase 2: De prototypefase 126
- 7.3 Fase 3: De oplossingsfase 130
- 7.4 Fase 4: De integratiefase 132
- 7.5 Conclusie 135

## **8 De vier dimensies meten 137**

- 8.1 De briefingfase 137
- 8.2 De prototypefase 145
- 8.3 De oplossingsfase 153
- 8.4 De integratiefase 161
- 8.5 Totaalscore van uw project 168

## **9 Conclusies 171**

## **Over de auteurs 175**

## De oorzaak achter de problemen

Zoals gezegd zijn er talloze problemen die bij falende IT-projecten kunnen worden geïdentificeerd. Voor al deze problemen is vast ook wel weer een oplossing te bedenken. Het is zeker niet onze bedoeling om voor ieder van de door ons geïdentificeerde problemen in IT-projecten een oplossing te gaan bedenken.

Wij willen u meer inzicht geven in de oorzaken die uiteindelijk ten grondslag liggen aan de problemen. Door uw IT-projecten beter of anders te organiseren kunt u de oorzaken van veel problemen wegnemen of kunt u de symptomen van veel problemen makkelijker herleiden naar de werkelijke oorzaak. Dit voorkomt dat u kostbare tijd moet besteden aan symptoombestrijding in plaats van aan structurele verbetering.

Wij hebben bij het analyseren van onze ervaringen van de afgelopen 15 jaar, waarin we zelf met onze voeten 'in de modder' hebben gestaan, een breed perspectief gehanteerd. Wij hebben gezamenlijk meer dan 25 jaar ervaring met IT-projecten in grote en middelgrote ondernemingen. Deze ervaring hebben we opgedaan in verschillende rollen, van ontwikkelaar tot adviseur, van projectleider tot externe auditor. Vaak ook als toeschouwer.

Gedurende het afgelopen jaar zijn wij bezig geweest met het in kaart brengen van de grootste problemen die wij tegenkwamen in IT-projecten bij onze klanten, maar ook die waarover we vernamen in publicaties en de media. We hebben hier een uitgebreide analyse op losgelaten en het is gelukt om de problemen te categoriseren en te rangschikken.

Uit de analyse is gebleken dat vrijwel alle problemen die zich voordoen bij falende IT-projecten, globaal door vier elementen kunnen worden veroorzaakt. In ons model zullen we deze vier elementaire oorzaken de **vier dimensies** noemen.

Deze dimensies zijn:

1. **Gemeenschappelijk doel:** Het is niet bij alle projectdeelnemers duidelijk wat het doel is van een project en het bredere bedrijfsdoel daarachter.
2. **Juiste gereedschap:** Specifieke problemen vereisen specifieke aanpakken. De middelen die worden ingezet om een project uit te voeren zijn niet altijd geschikt voor het type probleem.
3. **Communicatie en samenwerking:** Er is onvoldoende samenwerking en communicatie tussen deelnemers en belanghebbers van een project.
4. **Complexiteit:** Er wordt onnodig veel complexiteit in een project geïntroduceerd.

## 2.1 Vier projectdimensies

In deze paragraaf worden de vier projectdimensies kort besproken. Later in dit boek zullen we nog nader op de dimensies ingaan.

### 2.1.1 *Gemeenschappelijk doel*

Wanneer je een willekeurig lid van een projectteam vraagt waarom het project wordt uitgevoerd, is de conclusie vaak dat het achterliggende bedrijfsdoel geheel of gedeeltelijk uit het oog verloren is. Wie het boek *The Goal* van Eli Goldratt gelezen heeft weet hoe belangrijk het is duidelijk te hebben wat je doel is. Wat je meestal ziet is dat projectleden (soms met moeite) nog wel weten wat het doel is van het specifieke onderdeel van het project waaraan zij werken, maar dat ze het achterliggende bedrijfsdoel waar het allemaal om gaat niet kennen. We kunnen dit vergelijken met een voetbalelftal. Het gemeenschappelijke doel van het elftal is om meer doelpunten te maken dan de tegenstander. Dus een verdediger die alleen maar goed verdedigt is aardig, maar hij zal ook moeten bijdragen aan het faciliteren van het scorend vermogen van de spitsen. Andersom is een spits die uitsluitend probeert te scoren ook niet effectief. Hij zal ook soms moeten meeverdedigen om te voorkomen dat de tegenstander meer doelpunten scoort dan zijn eigen team. Zo heeft iedereen in het team zijn eigen belang en verantwoordelijkheid, maar ze zullen allemaal wel het

gemeenschappelijke doel moeten kennen en nastreven om succesvol te kunnen zijn.

Het niet of onvoldoende beseft hebben van het achterliggende doel van een project kan een belangrijke bron zijn van verkeerde beslissingen die tijdens het project genomen worden. Hoe kan je de optimale architectuur kiezen, de meest passende projectaanpak of de meest geschikte projectstructuur als je niet precies weet wat het gemeenschappelijke doel is? Het verbaast ons nog regelmatig hoe bijvoorbeeld IT-architecten uitsluitend op basis van technische gronden architectuurbeslissingen nemen zonder de achterliggende bedrijfsdoelstellingen hierbij in acht te nemen. Dit resulteert in de welbekende 'technische feestjes' waarbij de techneuten heel enthousiast werken aan een oplossing die technisch misschien heel goed in elkaar zit, maar de bedrijfsdoelstellingen uiteindelijk niet of onvoldoende ondersteunt. Mede hierdoor zien we over de gehele linie zeer groot onbegrip bij meer bedrijfsgeoriënteerde afdelingen over de aansluiting van de IT-afdeling bij de wensen vanuit 'de bedrijfsvoering'.

### 2.1.2 *Juiste gereedschap*

Bij grote IT-dienstverleners en interne IT-afdelingen van grote organisaties zie je vaak dat ze een standaard projectaanpak hebben die ze voor alle projecten hanteren. Of de projecten nu groot, klein, langlopend of kortlopend zijn. De projectaanpak is altijd hetzelfde. Steeds vaker zien wij bij grotere IT-dienstverleners de fabrieksmatige aanpak van IT-projecten. Hierin is weinig tot geen ruimte voor eigen interpretaties en initiatief bij de leden van een projectteam. Op de fabrieksmatige aanpak komen wij later in dit boek uitgebreid terug. In de praktijk zien wij vaak dat eenzelfde (fabrieksmatige) aanpak voor alle projecten niet optimaal werkt en zelfs de oorzaak kan zijn dat een project faalt.

Wij vergelijken dit ook wel met een timmerfabriek waarin uitsluitend gewerkt wordt met hamers als gereedschap. Hiermee kunnen bepaalde problemen perfect opgelost worden. Het slaan van een spijker in een stuk hout bijvoorbeeld. Voor andere problemen is het gereedschap echter niet geschikt, zoals het indraaien van een schroef. Je kan met alle geweld een eind komen door met een hamer te proberen een schroef ergens in te rammen, maar op de lange termijn is het geen

goede oplossing voor het probleem. Zelfs bij het inslaan van een spijker in een stuk hout kunt u zich voorstellen dat er soms een grote hamer nodig is en soms een kleine. Dit is afhankelijk van het type spijker en soort hout. In IT-projecten zien we helaas vaak de timmerfabriek terug met alleen maar hamers, het liefst zo groot mogelijk. Specifieke problemen vergen echter specifieke manieren van oplossen. In een klein, kortlopend project heeft het vaak geen zin om de complete Princez papierwinkel op te gaan zetten. Tegen de tijd dat de volledige methode is opgezet, is het budget op. In dit soort projecten volstaan een korte beschrijving van het project en de uit te voeren stappen inclusief een begroting en wellicht een risicoanalyse. Bij grotere projecten zullen soms complete referentiearchitecturen moeten worden uitgewerkt of een projectstart architectuurdocument worden geschreven. Waar het om gaat bij deze dimensie van IT-projecten is dat niet ieder project succesvol uitgevoerd kan worden met exact dezelfde projectaanpak en *deliverables*. Per project zal bekeken moeten worden welke aanpak het meest geschikt is voor het type probleem dat moet worden opgelost.

### 2.1.3 *Communicatie en samenwerking*

Bij grotere IT-projecten is het gebruikelijk om vooraf uitgebreide contracten op te stellen met afspraken over doelen en scopebepaling. Het risico hierbij is dat er vaak juristen aan te pas komen om deze contracten op te stellen. Nu is het onze ervaring dat juristen meestal heel slim zijn, maar dat ze over het algemeen weinig inhoudelijke kennis van de problematiek rondom een IT-project hebben. Regelmatig zie je in zo'n situatie dat alle risico's contractueel bij de opdrachtnemer worden gelegd (of de partij met de zwakste jurist). Het project moet het liefst tegen een vaste prijs en met een vaste datum van oplevering uitgevoerd worden, uiteraard tegen een zo laag mogelijke prijs. In deze gevallen is van samenwerking in de praktijk vaak nauwelijks meer sprake. De belangen zijn zo uit elkaar gedreven dat partijen elkaar eerder tegenwerken dan dat ze samenwerken aan een zo goed mogelijk projectresultaat.

Ter illustratie van het belang van samenwerking beschrijven we een fenomeen dat we een tijdje geleden in de politiek hebben gezien en dat vanuit Amerika overgewaaid is naar Nederland. Binnen een politieke

partij konden leiderskandidaten zichzelf kandidaat stellen en een eigen campagne voeren om gekozen te worden tot de nieuwe partijleider. Wat je zag gebeuren is dat men elkaar tijdens de verkiezingscampagne voor van alles en nog wat uitmaakte en dat het de partij meer uit elkaar dreef dat tot elkaar bracht. Na de leidersverkiezing bleek het erg lastig voor de gekozen leider om weer gezamenlijk verder te gaan met de overige niet-gekozen kandidaten in dezelfde partij. Van samenwerking was in de partij geen sprake meer en in de opiniepeilingen deed de partij het dan ook slecht. Het heeft lang geduurd voordat deze partij weer wat was opgekrabbeld en wij vermoeden dat ze deze manier om een nieuwe leider te kiezen niet snel weer zullen toepassen.

Wanneer wij het in dit boek hebben over samenwerking, bedoelen we in eerste instantie de samenwerking tussen opdrachtgever en opdrachtnemer. Hierin zien wij de grootste en meest risicovolle problemen. Uiteraard is ook de samenwerking tussen de verschillende stakeholders in en om het project van groot belang.

#### 2.1.4 *Complexiteit*

In IT-projecten hebben we met z'n allen vaak de neiging om zaken onnodig complex te maken. Een mooi voorbeeld is de Roer- en de Swalmentunnel die op 1 december 2009 na twee jaar vertraging (en ruimschoots boven het oorspronkelijke budget) officieel in gebruik is genomen. In Nederland hebben we toch al aardig wat ervaring met het bouwen van tunnels en de bijbehorende IT-systemen om bijvoorbeeld het verkeer veilig te regelen. Toch heeft men in dit project bedacht om een ingewikkeld nieuw systeem te gaan gebruiken dat uiteindelijk een grote bron van ellende is geworden. De verantwoordelijke minister (destijds Camiel Eurlings) heeft inmiddels dit soort complexe niet-bewezen en dure tunnelsystemen afgezworen en al een aantal malen publiekelijk geadviseerd nooit meer in overheidsprojecten dergelijke complexe onbewezen tunnelsystemen te gebruiken. Wij denken dat dit een heel verstandig advies is.

Bij het bedenken van een niet-complexe oplossing is de valkuil dat er vaak snel geconcludeerd wordt dat de oplossing te kort door de bocht gaat. Of dat men er onvoldoende draagvlak voor krijgt omdat de besluitvormers niet kunnen geloven dat een dergelijk eenvoudige

oplossing het complexe probleem prima kan oplossen. Toch is het van groot belang een oplossing zo eenvoudig mogelijk te houden om de kans op succes zo groot mogelijk te maken.

Het probleem achter te complexe oplossingen zit vaak in de totstandkoming ervan. Het gebeurt vaak dat men begint met een klein eenvoudig systeem en daar in de loop der jaren telkens een klein stukje complexiteit aan toevoegt. Wanneer u verzuimt om periodiek een stapje terug te doen om de complexiteit te kunnen overzien en eventueel te reduceren, loopt u het risico dat oplossingen al dan niet bewust zo complex worden dat de kans op instabiliteit toeneemt, de time-to-market verslechtert en onderhoudskosten de pan uit rijzen.

## 2.2 Wat kunt u verwachten in dit boek?

In het vervolg van dit boek zullen we de vier elementaire dimensies van IT-projecten nader onderzoeken en uitwerken. U zult zien dat alle problemen in IT-projecten op de een of andere manier zijn te relateren aan een van deze vier dimensies. Aan de hand van dit boek leert u hoe u focus kunt houden op deze vier elementaire dimensies; als u een van de dimensies verwaarloost, zult u de symptomen van een falend project snel herkennen. Alle belangrijke en actuele onderwerpen die spelen rond IT-projecten zullen in dit boek de revue passeren: *outsourcing* en *offshoring*, financieringsmodellen, preferred suppliermodellen, projectmanagement, communicatie- en verwachtingsmanagement, werken onder architectuur enzovoort. Het doel van dit boek is u te helpen specifieke problemen in uw eigen IT-projecten te herkennen en deze te plaatsen binnen het kader van de vier dimensies. Zodra u weet in welk van de vier dimensies het probleem valt, zult u met behulp van de hulpmiddelen die in dit boek geboden worden, in staat zijn de problemen structureel aan te pakken waardoor de succesansen van uw IT-projecten gegarandeerd zullen toenemen.