

# 7 | Valideren en onderhandelen

Het eerste deel van dit hoofdstuk behandelt het validatieproces en de validatieprincipes waarop dat proces is gebaseerd. Belangrijke onderwerpen zijn de kwaliteitscriteria waaraan requirements moeten voldoen en de validatietechnieken die een requirementsanalist daarvoor gebruikt.

Het tweede deel van dit hoofdstuk gaat over conflicterende requirements en de verschillende type conflicten die tussen belanghebbenden kunnen ontstaan. Hoe een requirementsanalist daarmee omgaat en welke onderhandelingstechnieken hij tot zijn beschikking heeft, komen uitgebreid aan bod.

## 7.1 Positionering

Het valideren van en onderhandelen over requirements is één van de hoofdactiviteiten van requirementsengineering. In hoofdstuk 1 zijn de hoofdactiviteiten kort toegelicht. Deze zijn nogmaals weergegeven in figuur 7.1. Dit hoofdstuk gaat uitgebreid in op de derde hoofdactiviteit.



Figuur 7.1 Activiteiten van requirementsengineering

Tijdens het valideren van requirements stellen de belanghebbenden vast of de requirements voldoen aan de vooraf gedefinieerde kwaliteitscriteria. De belanghebbenden kijken of de gedocumenteerde requirements inhoudelijk overeenkomen met hun eisen en wensen en of ze gebruikt kunnen worden als basis voor het ontwikkelen van de software.

### **Definitie validatie:**

*Het proces waarin gecontroleerd wordt of de gedocumenteerde requirements voldoen aan de behoeften van de belanghebbenden.*

### **Definitie acceptatie:**

*Het proces waarmee wordt beoordeeld of het systeem aan alle requirements voldoet.*

### **Definitie acceptatietest:**

*Een test om te beoordelen of het systeem aan alle requirements voldoet.*

Wanneer requirements niet goed gevalideerd zijn en er fouten in de goedgekeurde requirements blijven zitten, voldoet het opgeleverde systeem niet aan de behoeften of krijgt het project te kampen met hoge herstelkosten en -tijd.

*Definitie fout:*

1. *Defect, fault, bug*: een plek in een artefact dat incorrect is beschreven of weergegeven.
  2. *Error*: een verschil tussen het waargenomen en het gespecificeerde gedrag of resultaat.  
Een error is meestal een signaal voor een defect/fault/bug in een artefact.

*Definitie artefact:*

## Een tussen- of eindresultaat van systeemontwikkeling.

Fouten in requirements hebben grote gevolgen voor de overige ontwikkelactiviteiten. Requirements vormen immers de basis voor de verdere softwareontwikkeling. Hoe later foutieve requirements ontdekt en hersteld worden, hoe groter de gevolgen zijn. De herstelkosten stijgen exponentieel. Alle (tussen)producten die op foutieve requirements zijn gebaseerd moeten aangepast worden. Denk aan ontwerpen, testdocumenten, programmacode, beheer- en gebruikershandleidingen. Een zorgvuldige validatie verdient zich ruimschoots terug.

### Hoogte herstelkosten

Er is meerdere malen onderzoek gedaan naar de hoogte van de herstelkosten bij softwareontwikkeling. Het bekendste onderzoek is dat van Boehm (1981). Hij kwam tot de conclusie dat een fout in de requirements die je pas tijdens de realisatiefase ontdekt 5 tot 10 keer duurder is om te herstellen, dan wanneer je diezelfde fout tijdens het opstellen van de requirements herstelt. Voor een fout die je pas na de ingebruikname van het systeem ontdekt, zijn de herstelkosten maar liefst 100 tot 200 keer hoger dan tijdens de requirementsfase.

Op basis van verschillende onderzoeken komen Leffingwell & Widrig (2003) tot de conclusie dat 25 tot 40% van het totale budget voor maatwerksoftwareontwikkeling opgaat aan het herstellen van fouten in de requirements.

Bij een groot aantal validaties die wij zelf voor klanten hebben uitgevoerd, is de return on investment berekend. Daaruit bleek dat je met iedere euro die je aan validatie uitgeeft tussen de 6 en 11 euro aan herstelkosten utspaart.

### Valideren en verifiëren

We zien dat de termen valideren en verifiëren nogal eens door elkaar worden gebruikt. Toch hebben ze niet dezelfde betekenis. Bij valideren controleer je of je de juiste requirements hebt gedefinieerd (geven ze de behoeften van de belanghebbenden juist weer). Bij verifiëren controleer je of de requirements juist zijn (voldoen de requirements aan de kwaliteitscriteria). Onder de hoofdactiviteit ‘Valideren en onderhandelen’ vallen zowel valideren als verifiëren.

Een gedegen vali-  
gen de kans op h-  
proces helpt de k-

#### ► Betrek de ju

Bij iedere validatie moet de juiste belanghebbenden worden betrokken om de kwaliteit van de totale set aan te kunnen. Eindgebruiker, ontwikkelteam en leveranciers moeten allemaal een feerbaarheid hebben dat de kwaliteit voldoet.

Soms is het nodig betrekken. Aangenomen kost het hen meer benden voor de dag, op een zo hoog niveau.

Externe auditoren  
In sommige gevallen voeren  
vast en kunnen  
vanuit juridische partijen de audit

Het is af te raderen dat de voorwaarden voor de validering niet voldaan zijn. Hij is neer er impliciet.

#### ► Houd foute

Bij het valideren vinden van fouten deze fouten gebeurt software al dezen

## De belanghebber van fouten. Rom

- Efficiënt gel
  - De kans dat
  - De kans dat

## 7.2 Validatieprincipes

Een gedegen validatieproces is gebaseerd op bewezen principes. Deze validatieprincipes verhogen de kans op het identificeren van fouten in de requirements. Een gestructureerd validatieproces helpt de kwaliteit van de requirements te verhogen. De zes validatieprincipes luiden:

### ► Betrek de juiste belanghebbenden

Bij iedere validatieslag dient een requirementsanalist de juiste belanghebbenden te betrekken. Wie de juiste belanghebbenden zijn, hangt af van de requirements die gevalideerd worden en van de kwaliteitsaspecten die aandacht krijgen. Belanghebbenden uit de business valideren de totale set aan requirements onder andere op volledigheid, correctheid en noodzakelijkheid. Eindgebruikers valideren gewoonlijk alleen hun eigen requirements. Leden van het ontwikkelteam valideren de requirements voornamelijk op realiseerbaarheid, consistentie, verifieerbaarheid, compleetheid en voorbarige ontwerpbeslissingen.

Soms is het nodig om externe belanghebbenden van buiten de organisatie bij een validatie te betrekken. Aangezien zij minder bekend zijn met de context van het te ontwikkelen systeem kost het hen meer inspanning om de requirements te valideren. Zeker als de externe belanghebbenden voor de validatie ingehuurd worden, loont het de moeite om de requirements eerst zelf op een zo hoog mogelijk kwaliteitsniveau te brengen.

#### Externe audit

In sommige gevallen is het verstandig of zelfs noodzakelijk om een externe audit uit te laten voeren op de requirements. Hiermee stel je het kwaliteitsniveau van de requirements vast en kun je aantonen dat de kwaliteitsborging goed is ingericht. Dat kan bijvoorbeeld vanuit juridisch oogpunt van belang zijn. Bij uitbesteding op basis van *fixed price* laten de partijen de requirements soms op formele wijze extern beoordelen.

Het is af te raden om een requirementsanalist de door hemzelf opgestelde requirements te laten valideren. Hij is waarschijnlijk blind voor zijn eigen fouten en heeft te veel voorkennis. Wanneer er impliciete kennis wordt verondersteld, merkt de requirementsanalist dat zelf niet op.

### ► Houd foutdetectie en foutcorrectie gescheiden

Bij het valideren van requirements concentreren de belanghebbenden zich uitsluitend op het vinden van fouten. De gevonden fouten worden direct gedocumenteerd. Het herstellen van deze fouten gebeurt achteraf door een requirementsanalist. Deze werkwijze is bij het testen van software al decennia lang gebruikelijk.

De belanghebbenden hoeven zich bij deze werkwijze alleen te concentreren op het detecteren van fouten. Bovendien beoordeelt een requirementsanalist de gevonden fouten nogmaals als hij ze gaat corrigeren. Dit heeft de volgende voordelen:

- Efficiënt gebruiken van de tijd van de belanghebbenden.
- De kans dat fouten over het hoofd worden gezien is kleiner.
- De kans dat onterecht geïdentificeerde fouten worden gecorrigeerd is kleiner.

### ► Valideer vanuit verschillende invalshoeken

Door de te valideren requirements vanuit meerdere gezichtspunten te bekijken, is de kans dat er fouten over het hoofd worden gezien kleiner. Er is een holistisch beeld van de requirements nodig, omdat er veel aspecten en invalshoeken van belang zijn bij softwareontwikkeling. Denk bij invalshoeken voor het valideren van requirements aan:

- Verschillende eenheden of afdelingen binnen de organisatie.
- Verschillende disciplines binnen het ICT-project.
- Verschillende kwaliteitsaspecten.

### ► Verander de documentatievorm

In sommige gevallen is het door het omzetten van de requirements naar een andere documentatievorm, eenvoudiger om fouten te vinden. Dat komt omdat iedere vorm van documenteren voor- en nadelen heeft. Natuurlijke taal heeft bijvoorbeeld als nadeel dat het al snel ambigu ofwel dubbelzinnig is. Door een conceptueel model te maken van tekstueel beschreven requirements komen requirements aan het licht die voor meerdere uitleg vatbaar zijn. Hoewel een formele taal voor veel belanghebbenden moeilijk te begrijpen is, helpt het bijvoorbeeld om te valideren of er geen requirements ontbreken bij een ingewikkeld algoritme of complexe structuur.

### ► Maak een begin met vervolgproducten

Requirements dienen onder andere als basis voor ontwerp- en testproducten en voor gebruikershandleidingen. Het vervaardigen van deze producten is een effectieve manier om de kwaliteit van requirements te valideren. Het is daarom aan te bevelen om al tijdens het valideren van requirements een begin te maken met enkele onderdelen van deze producten. Een tester kan bijvoorbeeld van een complexe use case alvast globale test cases maken om te valideren of de kwaliteit van de use case-specificatie voldoende is. Nadeel van dit validatieprincipe is dat het veel tijd in beslag neemt.

#### Verwachtingen

Wanneer je bij het valideren van requirements alvast vervolgproducten laat maken, loop je het risico dat men denkt dat de requirements al definitief zijn. Benadruk daarom met welk doel je de vervolgproducten laat maken. Communiceer bijvoorbeeld duidelijk dat voor bepaalde requirements alvast het ontwerp gemaakt wordt, om te valideren of er geen requirements ontbreken.

### ► Valideer herhaaldelijk

Een requirementsanalist laat de vastgelegde requirements op meerdere momenten valideren. Dit doet hij niet alleen aan het einde van de requirementsanalyse maar ook tussentijds. Bijvoorbeeld na afloop van een interview of workshop of nadat de requirements voor een bepaald onderdeel of deelsysteem zijn opgesteld. Sommige validaties zijn erg formeel en andere meer informeel. Bijvoorbeeld een collegiale review uitgevoerd door een andere requirementsanalist is doorgaans informeel.

Wanneer er voor  
en goedgekeurde  

- een innovati
- een langdure
- een onbekan
- vroegtijdig g
- ontwikkeling
- hergebruik v

## 7.3

Requirements w  
teria.

**Definitie beoorde**  
*Het valideren*  
*begrijpelijk.*

Dit geldt voor in  
van het require  
effectieve validati  
liteit is daarbij va  
de doelen van he

- Inhoud  
Zijn alle req
- Documentat  
Zijn alle req
- Overeenstem  
Zijn alle bel
- conflicten o

bekijken, is de kans dat ereld van de requirementswareontwikkeling. Denk

aan een andere documentenform van documenteren dat het al snel ambitieel beschreven vatbaar zijn. Hoewel helpt het bijvoorbeeld algoritme of complexe

producten en voor gebrui-  
tieve manier om de om al tijdens het vali-  
an deze producten. Een cases maken om te valide-  
an dit validatieprincipe is

ducten laat maken, loop je nadruk daarom met welk eeld duidelijk dat voor valideren of er geen requi-

re momenten valideren.  
ar ook tussentijds. Bij requirements voor een bepaald formeel en andere meerdere requirementsanalist is

Wanneer er voortschrijdend inzicht optreedt, is het noodzakelijk om de al eerder gevalideerde en goedgekeurde requirements opnieuw te laten valideren. Dit is onder andere het geval bij:

- een innovatief project;
- een langdurend project;
- een onbekend domein;
- vroegtijdig goedgekeurde requirements;
- ontwikkelingen in de context van het project;
- hergebruik van requirements.

### 7.3 Kwaliteitsaspecten

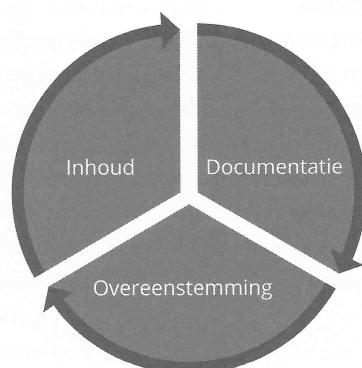
Requirements worden alleen goedgekeurd als ze voldoen aan vooraf gedefinieerde kwaliteitscriteria.

#### *Definitie beoordelen (van requirements):*

*Het valideren en tevens controleren van requirements op kwaliteitscriteria als eenduidig en begrijpelijk.*

Dit geldt voor individuele requirements en voor de set aan requirements als geheel. In het begin van het requirementstraject worden concrete kwaliteitscriteria vastgesteld om een objectieve en effectieve validatie mogelijk te maken. Rekening houden met alle relevante aspecten van kwaliteit is daarbij van belang. Figuur 7.2 laat deze kwaliteitsaspecten zien. Ze komen overeen met de doelen van het valideren van requirements:

- Inhoud  
Zijn alle requirements geëliciteerd en gedocumenteerd en hebben ze het juiste detailniveau?
- Documentatie  
Zijn alle requirements volgens de afgesproken documentatierichtlijnen vastgelegd?
- Overeenstemming  
Zijn alle belanghebbenden het eens met de gedocumenteerde requirements en zijn alle conflicten opgelost?



Figuur 7.2 Kwaliteitsaspecten van requirements

## Inhoud

Dit kwaliteitsaspect gaat over de inhoud en betekenis van requirements. Aan dit kwaliteitsaspect denken de meeste mensen als eerste. De in hoofdstuk 4 beschreven kwaliteitscriteria zijn criteria op het aspect inhoud. Dit geldt voor zowel de kwaliteitscriteria voor individuele requirements als voor requirementsdocumenten. Bij het valideren van requirements op inhoud laat een requirementsanalist onder meer de volgende kwaliteitscriteria beoordelen:

- **Volledig**  
Zowel de totale set aan requirements als de individuele requirements moeten volledig gedocumenteerd zijn. Er mogen geen requirements ontbreken en van iedere requirement moet alle relevante informatie aanwezig zijn.
- **Traceerbaar**  
Niet alleen de requirements, maar ook de requirementsdocumenten moeten traceerbaar zijn. De relaties met onder andere de bronnen en voorliggende documenten zijn dan inzichtelijk gemaakt.
- **Correct ofwel toereikend**  
De requirements moeten de behoeften van de belanghebbenden juist weergeven en zich daartoe beperken.
- **Consistent**  
Het moet mogelijk zijn om alle requirements te implementeren. Ze mogen onderling of in zichzelf geen tegenstrijdigheden bevatten.
- **Geen voorbarige ontwerpbeslissingen**  
In de requirements mogen geen oplossingen verborgen zitten. Eventuele vooraf opgelegde (technische) oplossingen moet een requirementsanalist explicet maken en als randvoorraad opeisen.
- **Verifieerbaar ofwel meetbaar**  
Het moet mogelijk zijn om acceptatie- en testcriteria te definiëren voor alle requirements, zodat aangetoond kan worden dat de software aan de requirements voldoet.
- **Noodzakelijk**  
Iedere requirement moet bijdragen aan de te behalen doelen.

### Oplossingen in de requirements

Requirements gaan alleen over wat er gerealiseerd moet worden, en waarom daar behoefte aan is. Requirements zeggen niets over hoe dat moet gebeuren. Alleen aan welke eisen een oplossing moet voldoen. Toch vinden veel belanghebbenden en ook requirementsanalisten het lastig om requirements vrij te houden van oplossingen. Mensen staan nu eenmaal niet lang stil bij de behoeften die ze hebben, maar vragen direct om de oplossing die ze voor ogen hebben. Een administrateur vraagt bijvoorbeeld een overzicht van alle achterstallige betalingen. Hij benoemt niet explicet dat hij bijvoorbeeld het versturen van aanmaningen efficiënter wil inrichten of dat hij een oplossing zoekt om het naleven van betalingstermijnen te bevorderen.

De reden om requirements oplossingsvrij te formuleren is dat je anders het risico loopt dat er alternatieven over het hoofd worden zien. In het voorbeeld hierboven blijft de administrateur zelf aanmaningen versturen, terwijl met een oplossingsvrije requirement over het efficiënter inrichten van het proces, misschien het automatisch versturen van aanmaningen

als oplossing

Of een requirement  
controleerde  
waarschijnlijker  
aan de staken  
ment na

## Documentatie

Dit kwaliteitsaspek  
de afgesproken

### Definitie confor

De mate wa

### Definitie standa

Een uniform

Wanneer de req

- het lastiger
- op specifiek
- belanghebbende
- er relevante
- belanghebbende
- logische plannen

Bij het valideren  
op onderstaande

- Conform de  
Is bij het va  
(inter)natio  
(zie hoofdstu
- Conform de  
Zijn bij het  
graafidelijns  
Is bij voorber
- Begrijpelijk  
Zijn de requi  
relevante te
- Eenduidig  
Zijn de requi  
benden moe
- Conform de  
Zijn bij het  
genomen? I

dit kwaliteitscriteria zijn individuele requirements op inhoud laat volledig vere requirementen traceerbaar en zijn dan gegeven en zich onderling of in vooraf opgelegde als randvoorwaarden requirements, et. daar behoeft welke eisen een rementsanalisten nu eenmaal niet dat die ze voor de achterstallige aan aanmaningen betalingstermijnen risico loopt dat lijft de administratief over het aan aanmaningen

als oplossing uit de bus komt.

Of een requirement oplossingsvrij is en er dus geen (impliciete) oplossing in zit, kun je controleren door je af te vragen of het ook op een andere manier kan. Als dat zo is, heb je waarschijnlijk een oplossing beschreven in plaats van een requirement. Door vervolgens aan de stakeholder te vragen waarom hij die oplossing wil, haal je de achterliggende requirement naar boven.

### Documentatie

Dit kwaliteitsaspect gaat over documentatierichtlijnen. De requirementsdocumenten moeten de afgesproken opbouw, regels en structuur volgen.

#### *Definitie conformiteit (van requirements):*

*De mate waarin een requirementsspecificatie voldoet aan de voorschriften in een standaard.*

#### *Definitie standaard (norm):*

*Een uniform voorschrift om iets te interpreteren, te maken of uit te voeren.*

Wanneer de requirementsdocumenten niet voldoen aan de richtlijnen is de kans groter dat:

- het lastiger is om de requirements te implementeren, omdat bepaalde ontwikkelactiviteiten op specifieke documentatiestandaarden zijn gebaseerd;
- belanghebbenden de requirements niet of verkeerd begrijpen;
- er relevante informatie ontbreekt;
- belanghebbenden requirements over het hoofd zien, omdat ze niet op een logische plaats staan.

Bij het valideren van requirements laat een requirementsanalist de documenten onder andere op onderstaande kwaliteitscriteria beoordelen:

- Conform documentatiestandaarden  
Is bij het vastleggen van de requirements gebruik gemaakt van de organisatie-interne en (inter)nationale standaarden? Denk bijvoorbeeld aan standaardzinssjablonen (zie hoofdstuk 5) en modelleertalen (zie hoofdstuk 6).
- Conform documentatiestructuur  
Zijn bij het opstellen van het requirementsdocument de afgesproken hoofdstuk- en paragraafindeling gehanteerd en zijn ze op de juiste manier toegepast?  
Is bijvoorbeeld het sjabloon voor de use case-specificaties correct gebruikt.
- Begrijpelijkheid  
Zijn de requirementsdocumenten voor alle belanghebbenden goed te begrijpen en zijn de relevante termen in de begrippenlijst gedefinieerd?
- Eenduidig ofwel ondubbelzinnig  
Zijn de requirementsdocumenten voor niet meer dan één uitleg vatbaar? Alle belanghebbenden moeten de documenten op dezelfde manier interpreteren.
- Conform documentatieregels  
Zijn bij het vastleggen van de requirements alle documentatie- en syntaxregels in acht genomen? Is bijvoorbeeld de correcte notatiemijze voor *include*-relaties gebruikt.

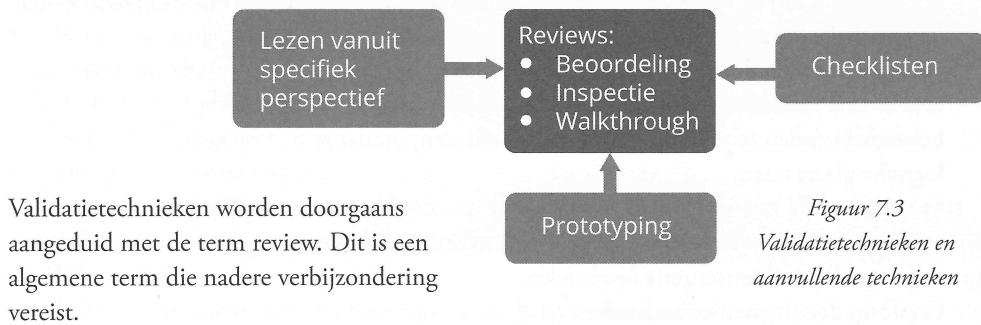
## Overeenstemming

Dit kwaliteitsaspect gaat over het goedkeuren van requirements door de belanghebbenden, ook nadat er voortschrijdend inzicht is opgetreden. Bij het valideren van requirements op het aspect overeenstemming laat een requirementsanalist de volgende kwaliteitscriteria beoordelen:

- Goedgekeurd  
Zijn de relevante belanghebbenden het eens met alle requirements?
- Goedgekeurd na wijzigingen  
Zijn de gewijzigde requirements opnieuw goedgekeurd door alle relevante belanghebbenden?
- Conflicten opgelost  
Zijn er geen onopgeloste conflicten meer over de requirements? Zijn bijvoorbeeld alle meningsverschillen en conflicten, die tijdens validatie naar boven zijn gekomen, opgelost?

## 7.4 Validatietechnieken

Om de requirements effectief te valideren zijn verschillende, deels elkaar aanvullende technieken, vorhanden. Deze zijn weergegeven in figuur 7.3. Een requirementsanalist kiest bij iedere validatie de best passende techniek. Dit is onder meer afhankelijk van de projectomstandigheden en of het om een tussentijdse of een finale evaluatie gaat.



Figuur 7.3  
Validatietechnieken en  
aanvullende technieken

### Definitie review:

*Een formeel georganiseerde controle van een (tussen)product door een groep experts.*

### Reviews

De meest gebruikte verschijningsvormen van reviews zijn:

#### ► Beoordeling

Bij deze reviewtechniek laat een requirementsanalist de requirements beoordelen door een derde, op basis van diens expertise. Een collegiale review door een andere requirementsanalist is de meest gebruikte vorm van beoordelen.

De reviewer bekijkt of de vastgelegde requirements voldoen aan de afgesproken kwaliteitscriteria en plaatst zijn opmerkingen erbij.

Meermaals laten beoordelen

We adviseren om je producten al in een vroeg stadium en regelmatig te laten beoordelen

door belanghebbenden  
insteken af  
van een intake  
een collega

#### ► Inspectie

Dit is een formele inspectie van de requirements (een review) genoemd.

**Definitie inspectie:**  
Een review waarbij de kwaliteitscriteria worden beoordeeld.

Een inspectie is een finieerde rollen gescheiden' kunnen zien met da

Plann

- Planning  
Dit is de voorbereiding bijeenkomsten en projectmanagement.

Praktische tips:  
We geven enige praktische tips.

- Stuur de documenten naar deelnemers voor bijeenkomsten in de geheugen.
- Je moet voor deelnemers in hun agenda voorbereid zijn.
- Voorzichtigheid verwijzen naar deelname van alle deelnemers.
- Laat deelnemers een tester inzetten.

hebbenden, ook  
nts op het aspect  
oordelen:

orbeeld alle  
men, opgelost?

llende technie-  
t kiest bij iedere  
tomstandighe-

## Checklisten

Figuur 7.3  
tiettechnieken en  
llende technieken

erts.

n door een  
remensanalist is

kwaliteitscrite-

en beoordelen

door belanghebbenden en collegarequirementsanalisten. Dit kun je formeel of informeel insteken afhankelijk van de bedrijfscultuur en de situatie. Laat bijvoorbeeld het resultaat van een interview of workshop door de deelnemers beoordelen en laat tussentijds af en toe een collega je werk reviewen.

### ► Inspectie

Dit is een formele reviewtechniek waarbij belanghebbenden in een gezamenlijke bijeenkomst de requirements valideren. Een inspectie wordt ook wel een inhoudelijke review (*technical review*) genoemd.

#### **Definitie inspectie:**

*Een reviewtechniek waarin een groep experts het (tussen)product beoordeelt op de afgesproken kwaliteitscriteria. De bevindingen van de experts worden vervolgens verzameld en geconsolideerd.*

Een inspectie is opgedeeld in fasen en verloopt volgens een gestructureerd proces met gedefinieerde rollen. Het eerder genoemde validatieprincipe ‘Houd foutdetectie en foutcorrectie gescheiden’ komt hierin explicet tot uiting. Figuur 7.4 laat een overzicht van het inspectieproces zien met daarin de fasen van een inspectie.



Figuur 7.4 Het inspectieproces

- **Planning**

Dit is de voorbereidingsfase. Het doel van de inspectie wordt explicet gemaakt en de bijeenkomst gepland. Concreet betekent dit het vaststellen van de te inspecteren requirementsproducten, de deelnemers, hun rollen, de locatie en het tijdstip van de bijeenkomst.

#### Praktische tips voorbereidingsfase

We geven enkele tips om de effectiviteit van je inspectie te verhogen:

- Stuur ongeveer vijf werkdagen voor de geplande bijeenkomst de te reviewen producten naar de reviewers. Doordat de belanghebbenden de review dan enkele dagen voor de bijeenkomst uitvoeren, zitten de requirements tijdens de bijeenkomst nog vers in het geheugen.
- Je moet de review uiteraard wel een paar weken van tevoren samen met de uitnodiging voor deelname aan de inspectie aankondigen, zodat reviewers tijd kunnen reserveren in hun agenda.
- Voorzie daar waar mogelijk de te reviewen producten van regelnummers, dit maakt verwijzen naar plaatsen in het document eenvoudig. Bovendien zijn de bevindingen van alle reviewers dan snel te sorteren naar de betreffende plaats in het document, dit versnelt de verwerking achteraf.
- Laat de gedetailleerde requirements in ieder geval door een ontwikkelaar en een tester inspecteren. Zij gebruiken de specificaties tenslotte als uitgangsdocumentatie.

Daarnaast is het verstandig om een softwarearchitect, een collegarequirementsanalist en een kwaliteitsmedewerker je requirements te laten inspecteren.

- Overzicht geven

Dit is een kick-off-bijeenkomst waarin de requirementsanalist een algemene toelichting geeft op de te inspecteren producten. Hij geeft achtergrondinformatie en maakt duidelijk welke aannames hij heeft gedaan.

- Fouten zoeken

Deze fase kunnen de reviewers individueel of in een gezamenlijke bijeenkomst uitvoeren. De reviewers lezen de te inspecteren producten grondig door en leggen de gevonden fouten vast.

#### Individueel en gezamenlijk

We raden aan om de requirementsproducten zowel individueel als in een gezamenlijke bijeenkomst te laten beoordelen. De individuele beoordeling is dan ter voorbereiding op de groepsbijeenkomst. Ook Wieggers (2003) stelt dat de individuele beoordeling een essentieel onderdeel van een inspectie is. Het maakt een inspectie de meest effectieve manier van valideren.

De groepsbijeenkomst wordt gewoonlijk foutregistratiemeeting genoemd. Tijdens deze meeting lopen de deelnemers gezamenlijk door de requirementsproducten aan de hand van de gedefinieerde kwaliteitscriteria. Ze bespreken de bevindingen en stellen de zwaarte van de bevindingen vast. Het resultaat van deze bijeenkomst is een lijst met gevonden fouten.

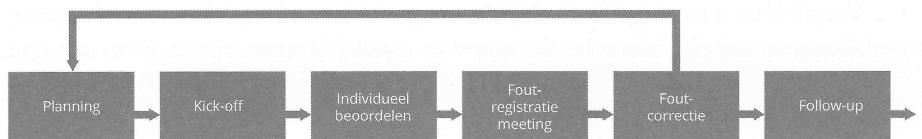
- Fouten verzamelen

In een gezamenlijke bijeenkomst worden de gevonden fouten besproken en vastgesteld of het daadwerkelijk fouten zijn. Reviewer kunnen immers van onjuiste aannames en verkeerde interpretaties uitgegaan zijn. Het resultaat van deze fase is een definitieve foutenlijst met daarin tevens maatregelen om de fouten te corrigeren.

#### Fagan-inspecties

Michael Fagan (1979) is de grondlegger van inspecties van softwareontwikkelproducten. Ze staan daarom bekend onder de naam Fagan-inspecties. Gilb en Graham (1993) hebben het verder uitgewerkt in hun boek 'Software Inspections'. Deze inspecties zijn tevens opgenomen in de IEEE norm 1028 (1997). Het Nederlandstalige boek 'Reviews in de praktijk' gaat uitgebreid in op de verschillende reviewvormen waaronder de inspectie (Cannegieter e.a., 2008).

Een Fagan-inspectie bestaat uit zes fasen. Ze zijn in onderstaande figuur weergegeven.



Meer informatie over Fagan-inspecties vind je op [www.gripoprequirements.nl](http://www.gripoprequirements.nl).

- Praktische We geven
- Beper inspec van de
- Vraag bevin dubbe status
- Spreek aan en nemer immer
- Laat e bende de bew bevin
- Laat e den, b meetin
- Weers requir zich be
- Beper bevin aan in inspec
- Zorg c blijft. volger

Bij een inspectie

- Organisat De organisat
- Moderator De moderat
- de bijeenk
- Auteur Tijdens de requiremen
- Voorlezer Tijdens de H
- Het is daa doorgaans

### Praktische tips foutregistratiemeeting

We geven een aantal tips om de effectiviteit van de foutregistratiemeeting te verhogen:

- Beperk het aantal reviewers op een foutregistratiemeeting tot maximaal zes. Een inspectie met een grotere groep blijkt minder resultaat op te leveren dan het verdelen van de belanghebbenden over twee parallelle foutregistratiemeetings (Wiegers, 2001).
- Vraag de reviewers voorafgaand aan de foutregistratiemeeting hun voorlopige bevindingen op te sturen naar de moderator. Hij kan de voorlopige bevindingen ondubbel en ze opnemen in een overzicht. Dit vergemakkelijkt het vastleggen van de status en prioriteit van de bevindingen.
- Spreek belanghebbenden die het document niet individueel beoordeeld hebben daarop aan en verplaats de bijeenkomst in het uiterste geval naar een later tijdstip. Als deelnemers ongestraft de individuele beoordeling achterwege kunnen laten, verschijnen ze immers vaker onvoorbereid op een inspectiebijeenkomst.
- Laat een foutregistratiemeeting maximaal twee uur duren. Daarna zijn de belanghebbenden minder alert en vinden ze substantieel minder fouten (Wiegers, 2003). Ga de bevindingen en oplossingen niet uitgebreid bespreken, maar stel alleen vast of een bevinding juist is.
- Laat een foutregistratiemeeting niet uitlopen. De kans is dan groot dat belanghebbenden, bij volgende meetings minder kritisch zijn, omdat ze willen voorkomen dat de meeting wederom uitloopt. Plan een extra meeting als er meer tijd nodig blijkt te zijn.
- Weersta de neiging om je te gaan verdedigen als de reviewers kritiek hebben op je requirements. Je vraagt ze immers om naar fouten te zoeken. De reviewers kunnen zich bezwaard gaan voelen om kritiek te leveren als je steeds verdedigt.
- Beperk indien nodig de inspectie tot de voornaamste requirements en bespreek alleen bevindingen die volgens de reviewers belangrijk zijn. Als belanghebbenden te veel tijd aan inspecties moeten besteden daalt hun motivatie en daarmee de effectiviteit van de inspecties.
- Zorg dat de aandacht van de reviewers op het inhoudelijk kwaliteitsaspect gericht blijft. Laat daarom geen producten inspecteren die nog spelfouten bevatten of niet volgens de afgesproken standaarden opgesteld zijn.

Bij een inspectie zijn de volgende rollen belegd:

- Organisator  
De organisator bereidt de inspectie voor en organiseert de bijeenkomsten.
- Moderator  
De moderator leidt de inspectie in goede banen. Hij bewaakt alle fasen en faciliteert de bijeenkomsten. Het is aan te raden om een neutrale moderator te kiezen.
- Auteur  
Tijdens de kick-off licht de auteur van de te inspecteren requirementsproducten de requirements toe. Eén of meer requirementsanalisten vervullen deze rol.
- Voorlezer  
Tijdens de inspectiebijeenkomst leest de voorlezer de requirements één voor één op. Hij licht ze in eigen woorden toe, zodat interpretatieverschillen aan het licht komen. Het is daarom belangrijk om een neutrale voorlezer te kiezen. De moderator neemt doorgaans deze rol op zich.

ementsanalist

e toelichting  
maakt duidelijk

mst uitvoeren.  
evonden fouten

ezamenlijke  
rbereiding op de  
ng een essentieel  
manier van

Tijdens deze  
aan de hand van  
de zwaarte van  
vonden fouten.

n vastgesteld of  
ames en ver-  
itieve foutenlijst

kelproducten.  
n (1993) hebben  
n tevens opge-  
ws in de praktijk'  
ie (Cannegieter

ergegeven.

- Reviewers  
De taak van de reviewers is het inspecteren van de requirementsproducten, het vinden van fouten en het rapporteren van hun bevindingen.
- Notulist  
De notulist legt tijdens de inspectiebijeenkomst de gevonden fouten en de vervolgacties vast.

#### Rollen combineren

Het is niet nodig en niet verstandig om iedere rol door afzonderlijke personen te laten vervullen. Het aantal betrokken personen wordt anders al snel te groot. Je hebt immers meerdere reviewers nodig. Het ligt voor de hand om zelf naast de auteursrol tevens de rol van notulist op je te nemen. In Fagan-inspecties is de rol van moderator meer omvattend. Hij organiseert ook de bijeenkomsten en loopt tijdens de foutregistratiemeeting door de requirementsproducten heen. De rollen organisator en voorlezer komen in Fagan-inspecties niet voor.

We zien regelmatig dat de auteur tevens de rol van moderator op zich neemt. Dit raden we af, omdat de auteur niet neutraal is en bewust of onbewust een te grote stempel op de foutdetectie drukt. Bovendien is de auteur inhoudelijk betrokken. Het is dan lastig om voldoende aandacht aan het proces te schenken.

#### Wanneer inspecties gebruiken?

Deze formele reviewtechniek is vooral geschikt voor de afsluitende review aan het einde van een requirements(deel)fase. In deze review vraag je een eendoordeel over de requirements en hoop je dat de belanghebbenden hun formele goedkeuring geven.

Een inspectie vergt meer tijd van de belanghebbenden dan de andere reviewtechnieken. Dit is bij veel projecten de reden om inspecties achterwege te laten en te kiezen voor een andere reviewtechniek. Realiseer je daarbij dat (goed uitgevoerde) inspecties zich terugverdienen, omdat ze meer fouten detecteren.

Dit blijkt onder andere uit het onderzoek van Jones (1996). Hij concludeerde dat bij informele reviews 20% tot 40% van de fouten gevonden worden en bij formele reviews 45% tot 70%.

#### ► Walkthrough

Deze reviewtechniek is een minder formele variant van inspecties. De belanghebbenden lopen de requirementsproducten gezamenlijk door op zoek naar requirements die niet voldoen aan de kwaliteitscriteria. Het proces is eenvoudiger en minder strikt dan bij een inspectie.

#### *Definitie walkthrough:*

*Een reviewtechniek waarbij de auteur van een (tussen)product een groep experts systematisch door het (tussen)product leidt. De bevindingen van de experts worden vervolgens verzameld en geconsolideerd.*

De deelnemers sproducten alva bijeenkomst leidt. Hij licht de req daarop als ze tw leiden, is het so

Bij een *walkthru*

- Reviewers
- Auteur
- Notulist

Wanneer u Je kunt bet laat review woordigers reviewen. E heen leiden kiezen zijn ject.

Reviewen i Reviews of kelteam vra mogelijk is stories goed sprint (vaak de belangrijk wikkeld de so

Reviewen e agile-project opgesteld e team) verfij van de gebu

User stories We komen requiremen we in dat g gegeven (hi

Aanvullende Om de effectivit requirementen

De deelnemers bereiden zich voor op de *walkthrough*-bijeenkomst door de requirement-sproducten alvast te lezen en te beoordelen op de belangrijkste kwaliteitscriteria. Tijdens de bijeenkomst leidt een requirementsanalist de deelnemers stap voor stap door zijn producten. Hij licht de requirements toe en verstrekkt eventueel extra informatie. De deelnemers reageren daarop als ze twijfelen aan de kwaliteit van de requirement. Om de discussies in goede banen te leiden, is het soms verstandig om daar een moderator voor in te schakelen.

Bij een *walkthrough* zijn ten minste de onderstaande rollen vertegenwoordigd:

- Reviewers
- Auteur
- Notulist

#### Wanneer *walkthrough* toepassen?

Je kunt beter een *walkthrough* gebruiken in plaats van een inspectie, als je de requirements laat reviewen door belanghebbenden uit de business, zoals gebruikers en klantvertegenwoordigers. Zij zijn niet gewend om requirementsdocumenten te lezen en requirements te reviewen. Bij een *walkthrough* kun je de reviewers begeleiden, beter door de documenten heen leiden en ingaan op vragen en discussies. Andere redenen om voor een *walkthrough* te kiezen zijn een zeer informele organisatiecultuur en reviews vroeg in het requirementsproject.

#### Reviewen in agile-projecten

Reviews of beter feedback op tussenresultaten spelen in agile een cruciale rol. Het ontwikkelteam vraagt feedback op de reeds ontwikkelde software, zo veel en zo vaak als praktisch mogelijk is. Tijdens de sprint kijkt de *product owner* geregeld mee om te zien of de *user-stories* goed geïmplementeerd worden. De sprint review meeting aan het einde van iedere sprint (vaak demo genoemd) heeft als voornaamste doel het verzamelen van feedback van de belanghebbenden uit de business. Het is dus niet alleen een demonstratie van de ontwikkelde software.

Reviewen en goedkeuren van requirements (meestal in de vorm van user stories) is in agile-projecten niet nodig. De requirements worden immers niet in een afzonderlijke fase opgesteld en goedgekeurd, maar ontstaan geleidelijk. De *product owner* (en het ontwikkelteam) verfijnen de requirements voortdurend en maken daarbij gebruik van de feedback van de gebruikers op de ontwikkelde software.

#### User stories reviewen

We komen geregeld organisaties en projecten tegen die op de traditionele manier met requirements omgaan en tevens user stories gebruiken. De user stories laten reviewen raden we in dat geval wel aan. Dit komt het meeste voor bij formele organisaties met strak vormgegeven (hiërarchische) bevoegdheden en bij (Europees) aanbestede opdrachten.

### Aanvullende technieken

Om de effectiviteit van de validatie te verhogen zijn aanvullende technieken vorhanden. Een requirementsanalist combineert een reviewtechniek vaak met één van onderstaande technieken

### ► Lezen vanuit een specifiek perspectief

Bij deze validatietechniek reviewen de belanghebbenden de requirements vanuit een bepaalde invalshoek. Iedere deelnemer krijgt vooraf een perspectief toegewezen, ook wel reviewrol genoemd. Hierbij ontvangt hij validatie-instructies en een checklist met de voor dat perspectief voornaamste vragen of onderwerpen die door de requirements geadresseerd dienen te worden.

Enkele voordelen van perspectief gebaseerd lezen zijn:

- Door de sterke focus ontdekken de belanghebbenden meer fouten.
- Een belanghebbende is minder tijd kwijt als hij alleen vanuit een specifiek perspectief reviewt.
- De kans is kleiner dat belangrijke aspecten over het hoofd worden gezien, omdat een requirementsanalist vooraf de relevante perspectieven definieert en daarvoor checklisten opstelt.

Een requirementsanalist stelt de relevante perspectieven vast en wijst ze toe aan de deelnemers. Hierbij kiest hij eerst een geschikte as waارlangs hij de relevante invalshoeken indeelt. De meest voor de hand liggende indeling is die naar groepen belanghebbenden. Denk dan bijvoorbeeld aan het perspectief van:

- Gebruikers(groepen)
  - Weerspiegelen de requirements alle gebruikersbehoeften?
- Klanten
  - Weerspiegelen de requirements de behoeften van onze klanten?
- Ontwikkelaars
  - Bevatten de requirements alle informatie die nodig is voor het ontwikkelen van de software?
- Testers
  - Bevatten de requirements de benodigde informatie voor het testen van de ontwikkelde software?

#### Perspectieven toewijzen

Wijs voor ieder perspectief bij voorkeur een belanghebbende uit de betreffende groep toe. Helaas is dat niet altijd mogelijk. Voor het vertegenwoordigen van de klantbelangen is vaak een vertegenwoordiger uit de eigen organisatie het best haalbare.

Een andere mogelijkheid om de perspectieven in te delen is een indeling naar de kwaliteitsaspecten van requirements:

- Inhoud
  - Voldoen de requirements en de requirementsproducten aan de inhoudelijke kwaliteitscriteria?
- Documentatie
  - Voldoen de requirements en de requirementsproducten aan alle documentatierichtlijnen?
- Overeenstemming
  - Zijn alle requirements door de juiste belanghebbenden goedgekeurd?

Lezen vanuit een specifiek perspectief is naast een aanvullende validatietechniek tevens een elicitiatietechniek (zie hoofdstuk 3).

### ► Prototyping

Met een prototype kan de belanghebbenden hun oordeel geven over de functionaliteit van het systeem. Op deze manier kunnen de belanghebbenden direct zien of de functionaliteit voldoet aan hun verwachtingen. En omdat de belanghebbenden direct kunnen zien of de functionaliteit voldoet aan hun verwachtingen, kunnen zij direct feedback geven. Dit kan helpen bij het vinden van problemen en het verbeteren van de functionaliteit.

Er zijn verschillende manieren om een prototype te maken. De meest gebruikte manier is de quick-and-dirty methode. Hierbij wordt een prototype snel gemaakt en uitgeprobeerd. De belanghebbenden kunnen dan direct feedback geven en de functionaliteit kan worden aangepast.

Tools voor prototyping:

- Voor het maken van prototypes zijn er verschillende tools beschikbaar, zoals Microsoft Visio of de open-source tool Dia. Deze tools maken het mogelijk om prototypes te maken zonder veel berekening.

Bij sommige tools is het mogelijk om prototypes snel te maken. Dit kan handig zijn voor de ontwikkeling van prototypes.

Het latere reviewen van het prototype:

- Instructies voor de belanghebbenden
  - Om het prototype te begrijpen en te evalueren.
- Validatiesessie
  - De belanghebbenden kunnen de functionaliteit van het prototype evalueren en feedback geven.
- Een checklist
  - De belanghebbenden kunnen de functionaliteit van het prototype evalueren en feedback geven.

Bij het valideren van het prototype is het belangrijk dat de belanghebbenden kunnen verrassen door de functionaliteit van het prototype.

### ► Prototyping

Met een prototype krijgen de belanghebbenden een concreet beeld van het toekomstige systeem. Op deze manier zijn ze beter in staat om zich een voorstelling van de oplossing te maken en om afwijkingen van de gewenste werkwijze te identificeren. Uit onderzoek van Jones (1998) blijkt dat prototyping de meest effectieve manier is om fouten in requirements te vinden. Aangezien met het ontwikkelen van een prototype extra tijd en geld is gemoeid, is het zelden wenselijk om alle requirements met behulp van een prototype te laten valideren.

Er zijn verschillende soorten prototypen te onderscheiden op basis van de vervolgstappen, namelijk wegwerp- en evolutionaire prototypen. Voor evolutionaire prototypen geldt dat ze steeds verder uitgebouwd en verbeterd worden om uiteindelijk onderdeel van het opgeleverde systeem uit te maken. Wegwerpprototypen worden daarentegen voor een specifiek doel ontwikkeld en daarna niet meer gebruikt.

#### Tools voor het maken van prototypen

Voor het maken van een prototype kun je uit verschillende tools kiezen. Met bijvoorbeeld Visio of de opensourcetool Pencil kun je eenvoudig schermverlopen simuleren. Het is ook mogelijk om met Powerpoint een grove opzet van het systeem te maken. Voor simulatie van berekeningen en beslisbomen is een spreadsheet geschikt.

Bij sommige ontwikkelomgevingen wordt zelfs programmatuur meegeleverd waarin je snel schermen en schermverlopen kunt maken. Voordeel hiervan is dat het prototype er dan realistisch uitziet en grotendeels overeenkomt met het uiteindelijke systeem. Nadeel daarvan is dat gebruikers en opdrachtgevers dan al snel de indruk hebben dat het systeem vrijwel klaar is. Er ontstaat dan gemakkelijk onbegrip over de onverwacht lange tijd die het ontwikkelen van het uiteindelijke systeem vergt.

Het laten reviewen van een prototype vergt een gedegen voorbereiding. Alleen het ontwikkelen van het prototype is niet toereikend. Een requirementsanalist bereidt de validatie op de volgende manieren voor:

- Instructies maken voor het gebruik van het prototype  
Om het prototype zelfstandig te kunnen uitproberen hebben de belanghebbenden duidelijke instructies of een gebruikershandleiding nodig.
- Validatiescenario's opstellen  
De belanghebbenden krijgen een aantal cases die ze met het prototype moeten doorlopen bij het valideren van de requirements. Bijvoorbeeld een case met specifieke data of over een toekomstig operationeel proces.
- Een checklist opstellen met validatiecriteria  
De belanghebbenden beoordelen het prototype aan de hand van een lijst met vragen over de relevante kwaliteitskenmerken.

Bij het valideren voeren de belanghebbenden de voorbereide scenario's uit. Daarnaast is het van belang dat ze zelf andere scenario's uitproberen en met het prototype experimenteren. Hierdoor kunnen verrassende fouten boven water komen. De belanghebbenden moeten dan wel de scope van het prototype en de daarin opgenomen requirements kennen. Ze kunnen dan gerichter

zoeken naar ontbrekende functionaliteit en beoordelen of de functionaliteit die ze missen binnen de scope valt. De belanghebbenden noteren de ontbrekende en onjuiste requirements en hun ervaringen met het prototype.

Voor het verkrijgen van objectieve resultaten is het noodzakelijk dat de belanghebbenden de validatie zelfstandig uitvoeren. De requirementsanalist blijft op de achtergrond. Hij observeert de belanghebbenden en noteert interessante zaken in observatienotulen. Denk hierbij aan een onverwachte actie van een belanghebbende of aan momenten waarop hij twijfelt. Dit zijn signalen voor ontbrekende of onjuist geïmplementeerde requirements. Het maken van video-opnamen is een andere manier om dit soort signalen te ontdekken.

Na afloop consolideert een requirementsanalist de notities van de belanghebbenden en zijn eigen observatienotulen. Als er veel of grote fouten gevonden zijn, laat een requirementsanalist het prototype aanpassen en een tweede validatieslag uitvoeren. Op basis van het validatieresultaat past hij de requirementsdocumentatie aan.

#### ► Checklisten

Checklisten zijn een effectief hulpmiddel om te voorkomen dat belangrijke punten over het hoofd worden gezien. Bij het valideren van requirements zijn checklisten met vragen over de relevante kwaliteitskenmerken gebruikelijk. Dit kunnen dwingende of minder dwingende vragenlijsten zijn. Een checklist kan namelijk verplicht en optioneel te valideren punten bevatten en eventueel een vaste validatievolgorde voorschrijven. Hoe strikter de checklist is, hoe gecontroleerder het validatieproces verloopt en hoe beter de resultaten van de individuele belanghebbenden te vergelijken zijn.

Het maken van een checklist is geen eenmalige actie. Een requirementsanalist onderhoudt de checklist en blijft alert op mogelijkheden om vragen duidelijker te formuleren, toe te voegen of te verwijderen. Daarnaast mogen checklisten niet te lang zijn of zich richten op futiliteiten. Het verdient aanbeveling om de vragenlijst eenvoudig en compact te houden, zodat een belanghebbende hem niet voortdurend hoeft te raadplegen terwijl hij aan het valideren is. Een requirementsanalist moet daarbij te algemene of te abstracte vragen vermijden. Een vraag als ‘Zijn de requirements correct geformuleerd?’ is bijvoorbeeld niet specifiek genoeg.

#### Opstellen checklisten

Tips voor het opzetten en onderhouden van checklisten vind je in ‘Reviews in de praktijk’ (Cannegieter e.a., 2008). Voorbeelden van checklisten vind je op [www.gripoprequirements.nl](http://www.gripoprequirements.nl).

## 7.5 Omgaan met conflicterende requirements

Bij het opstellen van requirements spelen uiteenlopende belangen een rol. De wensen en behoeften van de belanghebbenden lopen zelden synchroon. De belanghebbenden moeten het immers eens worden over de complete set aan requirements. Een requirementsanalist streeft dan ook naar een gemeenschappelijk en gedragen begrip van de requirements. Wanneer de belanghebbenden niet tot overeenstemming kunnen komen, ontstaan conflicten. Conflicten over

de requirements  
requirementsanalist  
groot dat bepaal

Wanneer conflic  
weergegeven acti

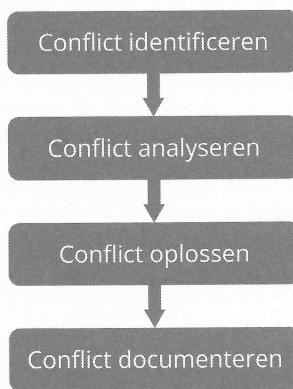
- Om conflicteren  
durend alert op  
moment in het
- Tijdens het  
dat verschill
- Tijdens het  
vast, dat de  
requiremen
- Tijdens het  
zijn met req
- Tijdens het  
kunnen me

Aan conflicteren  
vinden van een  
mentsanalist an  
volgende paragr  
van toepassing z

Bij het zoeken n  
den. Dit zijn ni  
en zienswijzen v  
nieuw conflic  
dien heeft dit ee

de requirements vormen een bedreiging voor het succes van het project. Wanneer een requirementsanalist niet op de juiste manier omgaat met conflicterende requirements is de kans groot dat bepaalde belanghebbenden het project niet langer steunen.

Wanneer conflicterende requirements ontstaan, voert een requirementsanalist de in figuur 7.5 weergegeven activiteiten uit.



Figuur 7.5 Activiteiten omgaan met conflicten

Om conflicterende requirements vroegtijdig te identificeren is een requirementsanalist voortdurend alert op het ontstaan ervan. Conflicterende requirements kunnen immers op ieder moment in het requirementsproces ontstaan:

- Tijdens het eliciteren van requirements constateert een requirementsanalist bijvoorbeeld, dat verschillende afdelingen binnen de organisatie elkaar tegenspreken.
- Tijdens het documenteren van requirements stelt een requirementsanalist bijvoorbeeld vast, dat de nieuwe informatie van een gebruiker in strijd is met eerder gedocumenteerde requirements van een andere gebruikersgroep.
- Tijdens het valideren van requirements, blijkt een manager het bijvoorbeeld niet eens te zijn met requirements van een stafafdeling.
- Tijdens het beoordelen van een wijzigingsverzoek, blijkt de indiener niet uit de voeten te kunnen met een breed gedragen requirement.

Aan conflicterende requirements kunnen allerlei oorzaken ten grondslag liggen. Voor het vinden van een passende oplossing is het kennen van die oorzaak cruciaal. Een requirementsanalist analyseert daarom eerst om wat voor type conflict het gaat. Deze typen zijn in de volgende paragraaf beschreven. Het is niet uitzonderlijk dat er tegelijkertijd meerdere oorzaken van toepassing zijn.

Bij het zoeken naar een oplossing betreft een requirementsanalist alle relevante belanghebbenden. Dit zijn niet alleen de belanghebbenden die een conflict hebben. Wanneer de standpunten en zienswijzen van andere belanghebbenden buiten beeld blijven, is er een reële kans op een nieuw conflict nadat overeenstemming over de conflicterende requirements is bereikt. Bovendien heeft dit een negatieve invloed op de bereidheid van belanghebbenden om medewerking

aan het project te verlenen. De technieken die een requirementsanalist gebruikt bij het zoeken naar een oplossing voor conflicterende requirements zijn in paragraaf 7.7 beschreven.

#### Procesafspraken maken

Begin voordat je inhoudelijk op het conflict ingaat met het maken van afspraken over het te volgen proces. Spreek met de belanghebbenden af welke stappen jullie gaan volgen om het conflict op te lossen en wat de tijdslijn is. Spreek verder af welke onderhandelingstechnieken de voorkeur hebben en op welke wijze wordt gehandeld als jullie geen oplossing voor het conflict kunnen vinden.

Wanneer een oplossing is gevonden voor de conflicterende requirements, documenteert een requirementsanalist alle relevante informatie over het conflict. Op deze manier is het mogelijk om later terug te halen hoe het conflict is opgelost. Dit is van belang als later blijkt dat de gekozen oplossing toch onwenselijk is of als opnieuw een conflict over hetzelfde onderwerp ontstaat. Een requirementsanalist legt alle relevante informatie vast:

- Oorzaak van het conflict
- Betrokken belanghebbenden
- Meningen van de belanghebbenden
- Gehanteerde aanpak en onderhandelingstechniek(en)
- Overwogen alternatieven
- Beslissing
- Onderbouwing van de beslissing

#### Gedocumenteerde oplossing verifiëren

Vergeet niet om de informatie die je over het conflict hebt vastgelegd, voor te leggen aan de belanghebbenden. Je laat zo verifiëren of je alles correct hebt vastgelegd. Bovendien haal je op deze manier eventuele interpretatieverschillen over de overeengekomen oplossing naar boven.

## 7.6 Type conflicten

Een conflict over requirements kan allerlei oorzaken hebben. De onderliggende oorzaak is in belangrijke mate van invloed op de aard en het verloop van het conflict. Het is daarom niet verwonderlijk dat Moore (2003) die oorzaken als basis heeft gebruikt voor zijn typering van conflicten. Hij onderscheidt de volgende typen conflicten:

- Inhoudelijke conflicten  
Aan dit type conflict ligt een verschil in informatie ten grondslag. Eén of meer belanghebbenden zijn verkeerd of onvoldoende geïnformeerd of ze interpreteren de informatie niet hetzelfde. Hierdoor is een verschil van mening ontstaan, dat door het verstrekken van de juiste informatie of bewijzen relatief eenvoudig op te lossen is. Een voorbeeld hiervan is een conflict over de tijd die klanten bereid zijn te wachten, voordat een website respons geeft op input van de klant.
- Belangenconflicten  
Wanneer belanghebbenden tegenstrijdige belangen hebben of verschillende doelen nastreven zijn conflicterende requirements bijna onvermijdelijk. Dit speelt bijvoorbeeld als een

- commerciële delivery-managers
- Waardencoördinatoren  
Soms is de cultuurveranderingen en bijvoorbeeld hebben staatssecretarissen een voorkeur voor een bepaald type hierbij hoort een soonechte regeling die zijn soms verschillende machtsstrijd.
- Relatiecoördinatoren  
Bij dit type hierbij hoort een soonechte regeling die zijn soms verschillende machtsstrijd.
- Structuurcoördinatoren  
Bij dit type hierbij hoort een soonechte regeling die zijn soms verschillende machtsstrijd.

Rol requirementsanalisten  
Voor zover dat de requirementsanalisten belanghebbenden goed eindigen.

Als je bij handelingen gesprekken dat ook niet brengen. Soms

## 7.7

Bij het zoeken naar een goede oplossing kan de requirementsanalist gebruik maken van een andere techniek. Dit kan voor de keuze van de bestemmingen en de cultuur.

► **Overeenstemming**  
De partijen wisselen ervaringen. Als dit geslaagd. Er zijn verschillende mogelijkheden.

kt bij het zoeken  
chreven.

spraken over het  
gaan volgen om  
erhandelingstech-  
geen oplossing

umenteert een  
r is het mogelijk  
olijkt dat de geko-  
nderwerp ontstaat.

r te leggen aan de  
ovendien haal je  
oplossing naar

e oorzaak is in  
daarom niet  
typering van

neer belang-  
de informatie  
verstreken  
voorbeld hier  
en website

doelen nastre-  
orbeeld als een

commercieel manager de nieuwe software zo snel mogelijk op de markt wil brengen en een delivery-manager het vooral belangrijk vindt dat de software stabiel en goed getest is.

- Waardenconflicten

Soms is de oorzaak van een conflict te herleiden naar een verschil in normen en waarden. Cultuurverschillen en persoonlijke overtuigingen leiden soms tot onverenigbare requirements en hardnekkige conflicten. Een requirement als het printen van rapporten kan bijvoorbeeld op verzet stuiten van belanghebbenden die duurzaamheid hoog in het vaandel hebben staan.

- Relatieconflicten

Bij dit type conflict ligt de oorzaak in de persoonlijke verhoudingen. De emoties kunnen hierbij hoog oplopen. Niet onenigheid over de requirements zelf maar de verstoerde persoonlijke relatie tussen belanghebbenden is dan het echte probleem. Die verstoerde relaties zijn soms voor de start van het project al ontstaan. Om elkaar dwars te zitten of een machtsstrijd uit te vechten, verwerpen de belanghebbenden elkaars requirements.

- Structuurconflict

Bij dit type conflict acht een belanghebbende een andere belanghebbende niet competent genoeg. Hij is ervan overtuigd dat de ander niet in staat is om een goed oordeel te vellen en wijst daarom zijn requirements zondermeer af. Dit speelt bijvoorbeeld bij managers die van hun ondergeschikten geen inspraak of invloed dulden en bij experts die zichzelf superieur vinden.

#### Rol requirementsanalist

Voor zover dat nodig is voor het opstellen van requirements, behoort het tot je taak als requirementsanalist om het voortouw te nemen bij het oplossen van conflicten tussen belanghebbenden. Toch zijn er ook situaties waarin je de autoriteit mist om dat tot een goed einde te brengen. Dat is al snel het geval bij een relatieconflict of een structuurconflict.

Als je bij het oplossen van een conflict te weinig voortgang boekt met inhoudelijke gesprekken, is het verstandig om de oorzaak van het conflict bespreekbaar te maken. Werkt dat ook niet, dan is er waarschijnlijk meer autoriteit nodig om de partijen tot elkaar te brengen. Schakel in dat geval de projectmanager of opdrachtgever in.

## 7.7 Onderhandelingstechnieken

Bij het zoeken naar een oplossing voor conflicterende requirements maakt een requirementsanalist gebruik van onderhandelingstechnieken. Als de situatie daarom vraagt switcht hij naar een andere techniek of gebruikt hij een combinatie van technieken. Aspecten die bepalend zijn voor de keuze van een onderhandelingstechniek zijn het soort conflict, de projectomstandigheden en de cultuur van de organisatie. De bekendste onderhandelingstechnieken zijn:

### ► Overeenstemming bereiken

De partijen wisselen informatie en argumenten uit om de anderen van hun standpunt te overtuigen. Als dit lukt en de partijen komen allemaal tot hetzelfde inzicht, dan is deze techniek geslaagd. Er zijn geen verliezers, alle partijen zijn het eens met de oplossing.

**Wanneer toepassen**

Het spreekt voor zich dat deze onderhandelingstechniek tot het meest bevredigende resultaat leidt. Probeer daarom eerst om overeenstemming tussen alle partijen te bereiken. Dat heeft alleen kans van slagen als het conflict nog niet hoog is opgelopen en de partijen bereid zijn naar elkaar te luisteren. Inhoudelijke conflicten lenen zich het beste voor deze onderhandelingstechniek. Vaak kun je belangenconflicten en zelfs waardenconflicten oplossen door de partijen inzicht te geven in elkaars drijfveren. Ze komen dan tot een compromis of zelfs tot overeenstemming.

**► Compromis sluiten**

De partijen zoeken naar een oplossing die (in enige mate) recht doet aan alle belangen. Het is een proces van geven en nemen, iedereen krijgt een beetje zijn zin. Een compromis is een samensmelting van onderdelen uit alternatieve oplossingen of een volledig nieuwe oplossing.

**► Stemmen**

De partijen brengen allemaal een stem uit op één van de voorliggende oplossingsalternatieven. Iedereen legt zich neer bij het alternatief dat de meeste stemmen krijgt. Hoewel het voor sommige partijen even slikken zal zijn, is de oplossing vanwege het democratisch proces vaak toch verdedigbaar en acceptabel voor alle partijen.

**Stok achter de deur**

Als partijen eindeloos blijven discussiëren, kun je deze techniek als stok achter de deur gebruiken. Je spreekt dan bijvoorbeeld aan het begin van een onderhandelingsbijeenkomst af dat er gestemd gaat worden als na 2 uur nog geen akkoord bereikt is.

**Stemmen: meerdere mogelijkheden**

Naast iedere partij een stem te laten uitbrengen op het alternatief van zijn voorkeur, kun je allerlei varianten gebruiken. Denk eens aan variaties in:

- Het aantal stemmen of punten die men over de alternatieven mag verdelen. Bijvoorbeeld dat iedereen in totaal drie punten mag geven en ze naar keuze over één, twee of drie alternatieven mogen verdelen.
- Het aantal stemrondes. Bijvoorbeeld alleen de alternatieven met meer dan 25% van de stemmen gaan door naar de volgende stemronde.
- De zwaarte van de stemmen. Bijvoorbeeld een stem van een *key user* telt dubbel.
- Het aantal alternatieven dat de partijen moeten kiezen. Bijvoorbeeld een top drie kiezen.

**► Definiëren van varianten**

De partijen zoeken naar een manier om conflicterende requirements toch naast elkaar te implementeren. Ze bekijken of het mogelijk is om het systeem instelbaar te maken en definiëren daartoe varianten die met behulp van systeeminstellingen of parameters te implementeren zijn.

Kosten  
Neem bij de overwegin worden, ma

**► Overrulen**

Als er geen tijd is is overrulen waar doorslaggevend. boven de requiren Bij een gelijke po

**► Alle feiten a**

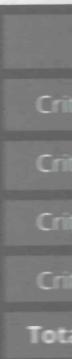
De partijen zette het conflict nadelen feiten vast door z gen.

**► Categoriseren**

Om een alternatief heeft plus en minus. D (verder) onderzo

**► Beslissingsn**

De partijen zette singscriterium voorbeeld in figuur hoogste totaalsco



### Kosten

Neem bij deze optie de hogere kosten voor het ontwikkelen en beheren van het systeem in de overwegingen mee. Er moet niet alleen extra functionaliteit ontwikkeld en onderhouden worden, maar door het instelbaar maken neemt de complexiteit van de software toe.

#### ► Overrulen

Als er geen tijd is om te onderhandelen of als de andere technieken geen succes hebben gehad, is overrulen waarschijnlijk de enige optie. Hierbij is de hiërarchische positie van de partijen doorslaggevend. De requirements van de partij met de hoogste hiërarchische positie, gaan boven de requirements van de andere partijen zonder rekening te houden met hun bezwaren. Bij een gelijke positie beslist een belanghebbende op een hogere positie.

#### ► Alle feiten afwegen

De partijen zetten zoveel mogelijk feiten op een rij door alle factoren die van invloed zijn op het conflict nader te onderzoeken. Vervolgens stellen ze de mate van relevantie van de gevonden feiten vast door ze te prioriteren. Deze informatie vormt de basis voor verdere onderhandelingen.

#### ► Categoriseren in plus/minus/interessant

Om een alternatief goed te kunnen beoordelen onderzoeken de partijen welke gevolgen een alternatief heeft. Positieve en negatieve gevolgen plaatsen ze respectievelijk in de categorieën plus en minus. De overige gevolgen vallen onder de categorie interessant en moeten nog (verder) onderzocht worden.

#### ► Beslissingsmatrix invullen

De partijen zetten alle oplossingsalternatieven in een kolom van de matrix. Voor ieder beslissingscriterium voegen ze een rij toe. Daarna scoren ze de alternatieven op alle criteria. In het voorbeeld in figuur 7.6 is een puntenschaal van één tot vijf gehanteerd. Het alternatief met de hoogste totaalscore, alternatief 2 in het voorbeeld, wordt gekozen.

	Alternatief 1	Alternatief 2	Alternatief 3
Criterium 1	5	3	1
Criterium 2	2	5	3
Criterium 3	1	4	3
Criterium 4	4	5	4
Totaal score	12	17	11

Figuur 7.6 Voorbeeld beslissingsmatrix

### Wegingsfactor

Als de beslissingscriteria niet allemaal even belangrijk zijn, kun je aan ieder criterium een wegingsfactor toekennen. Om de totaalscore te berekenen vermenigvuldig je iedere individuele score met de bijbehorende wegingsfactor en tel je per alternatief de uitkomsten bij elkaar op. In onderstaande beslissingsmatrix is een extra kolom met wegingsfactoren aan de matrix in figuur 7.6 toegevoegd.

	Wegingsfactor	Alternatief 1	Alternatief 2	Alternatief 3
Criterium 1	1	5	3	1
Criterium 2	0,2	2	5	3
Criterium 3	0,2	1	4	3
Criterium 4	0,5	4	5	4
Totaal score		7,6	7,3	4,2

De totaalscore van alternatief 1 is als volgt berekend:

$$1*5 + 0,2*2 + 0,2*1 + 0,5*4 = 7,6 \text{ punten}$$

## 7.8 Samenvatting

Tijdens het valideren van requirements stellen de belanghebbenden vast of de requirements voldoen aan hun behoeften en aan de vooraf gedefinieerde kwaliteitscriteria. Het is van belang om daarbij rekening te houden met de kwaliteitsaspecten inhoud, documentatie en overeenstemming. Het aspect inhoud, dat over de betekenis van de requirements gaat, krijgt gewoonlijk de meeste aandacht. De requirementsdocumenten moeten daarnaast de documentatierichtlijnen volgen en door de belanghebbenden goedgekeurd zijn.

Een gestructureerd validatieproces helpt de kwaliteit van de requirements te verhogen. Dit proces is gebaseerd op bewezen validatieprincipes waaronder het scheiden van foutdetectie en foutcorrectie.

Er bestaan verschillende technieken om requirements te valideren, vaak reviews genoemd. De meest gebruikte validatietechnieken zijn beoordeling, inspectie en *walkthrough*. Deze worden vaak aangevuld met technieken als prototyping, checklisten en lezen vanuit een specifiek perspectief.

Wanneer de belanghebbenden niet tot overeenstemming kunnen komen, ontstaan conflicten. Het is aan een requirementsanalist om de belanghebbenden op één lijn te brengen en een oplossing voor conflicterende requirements te zoeken. Hij houdt hiertoe rekening met de achterliggende oorzaak van het conflict en stelt vast om wat voor type conflict het gaat. Dat kan bijvoorbeeld een inhoudelijk conflict, een belangconflict of een relatieconflict zijn. Een requirementsanalist kiest dan een daarbij passende onderhandelingstechniek.

### Oefen

1. Welke twee van de volgende opties zijn correct?

- a. Overeenstemming
- b. Documentatie
- c. Mogelijkheid
- d. Inhoud
- e. Betrekking

2. Je bent een projectmanager. De klant heeft de volgende vraag: 'Wat is de belangrijkste plaats voor de documentatie in de requirements?'

- a. Nadat de requirements zijn opgesteld
- b. Meerdere mogelijkheden
- c. Voordat de requirements zijn opgesteld
- d. Voor het ontwerp

3. Je bent een projectmanager. De klant heeft de volgende vraag: 'Wat is de belangrijkste plaats voor de documentatie in de requirements?'

- a. Zonder belanghebbenden
- b. De acceptatiefase
- c. De requirements fase
- d. Het achterliggende oorzaak

## Oefenvragen

1. Welke twee van de volgende kwaliteitsaspecten zijn geen kwaliteitsaspecten voor het valideren van requirements?
  - a. Overeenstemming
  - b. Documentatie
  - c. Mogelijkheden
  - d. Inhoud
  - e. Betrek de juist belanghebbenden
2. Je bent een requirementsanalist bij een omvangrijk en langdurig project. De projectmanager vraagt je wanneer de requirements gevalideerd moeten worden. Wanneer zou de validatie plaats moeten vinden?
  - a. Nadat de requirements zijn opgenomen in het requirementsmanagementtool.
  - b. Meerdere keren.
  - c. Voordat ieder requirement versie 1.0 krijgt.
  - d. Voor het ondertekenen van de requirements door de projectmanager.
3. Je bent een requirementsanalist in een project dat onder zware tijdsdruk staat. De projectmanager wil de onderhandelingen over de requirements overslaan. Je wilt aan de projectmanager duidelijk maken wat daarvan de risico's zijn. Welk argument kun je daarvoor niet gebruiken?
  - a. Zonder overeenstemming over de requirements riskeer je een conflict tussen belanghebbenden.
  - b. De acceptatie van het systeem wordt bedreigd.
  - c. De requirements kunnen niet worden goedgekeurd want de kwaliteit is niet vastgesteld.
  - d. Het achterwege laten van de onderhandelingen kan leiden tot hogere kosten dan de kosten van de onderhandelingen zelf.

4. Welke drie van onderstaande technieken zijn geen validatietechnieken?

- a. Lezen vanuit specifiek perspectief
- b. Beoordeling
- c. Revisie
- d. Vragenlijsten
- e. Inspectie
- f. Perspectieven wijzigen
- g. Walkthrough
- h. Prototyping

5. Er is een conflict tussen twee security-specialisten in het project. Als requirementsanalist is het je taak om dit conflict op te lossen. Je hebt geprobeerd om binnen de beschikbare tijd een compromis te sluiten, maar die poging is mislukt. Wat is je beste volgende stap?

- a. Je beslist wat de beste oplossing is en legt dit vast in de requirementsdocumentatie.
- b. Je beslist om de onderhandelingen te stoppen (omdat er geen tijd meer is) en laat de beslissing over aan de projectmanager.
- c. Je dwingt de twee *security-specialisten* om tot overeenstemming te komen.
- d. Je schakelt een derde partij met beslissingsbevoegdheid in om een beslissing over het conflict te nemen.
- e. Je beslist dat de twee tegenstrijdige requirements allebei geïmplementeerd moeten worden.

# 8 | R

De zes voorname beschreven. Het die attributen ee riteren van requi

In een hoofdstuk baarheidsrelaties van een proces

## 8.1

Het managen van hoofdstuk 1 zijn ring zijn weergege 8 gaat uitgebreid

Het doel van requ volgorde juist en

**Definitie requirements.**  
Het proces van (tussen)produ ruments.

Voor goed requ lijk. Daarnaast di zogenaamde requ informatie over in gelegde informatie kunnen gebruiken