

Uit de annalen van de analyse

We zijn gewend om vooruit te kijken. Door de grote snelheid waarmee veranderingen zich voltrekken, lijken ideeën van 5 jaar geleden al achterhaald. Toch is echte vooruitgang alleen maar mogelijk, wanneer je het verleden kent en begrijpt. Alleen al om niet dezelfde fouten te maken. Als we de oude wijn in oude zakken laten, ontstaat er in nieuwe zakken ruimte voor nieuwe wijn. Je bent een genie wanneer je het wiel uitvindt, maar als je geweten had dat het al bestond, had je ruimte gehad om iets echt nieuws uit te vinden. Daarom kijken we af en toe achterom.

door Hans Siebering

Okee, achterom kijken. Maar waar kijk je dan? Twee leden van de redactie bleken twintig jaar geleden vanuit de behoefte aan houvast in de aanpak van requirements vraagstukken hetzelfde boek te hebben aangeschaft: Requirements Engineering – A good practice guide van Ian Sommerville en Pete Sawyer. Laten we eens kijken wat we nu nog aan dit boek kunnen hebben. Vakliteratuur lees je tenslotte niet voor de stijl, maar voor de ideeën. Als die ideeën achterhaald zijn of in recente vakliteratuur ook goed verwoord zijn, dan verliest oude vakliteratuur, in tegenstelling tot 'literaire literatuur', zijn waarde.

Context

Het boek is geschreven in 1997. Er was in die tijd een sterk bewustzijn dat de kwaliteit van de requirements en de doorlooptijd van het ontwikkelproces belangrijke succesfactoren waren. Het doel van het boek is de kwaliteit van requirements te verbeteren. Het is geen leerboek, maar een handboek, gericht op praktische oplossingen. De structuur van het boek is heel simpel. Het is niet meer dan een verzameling richtlijnen, die het resultaat zijn van een Europees project voor de verbetering van de betrouwbaarheid van het vak.

Requirements Engineering was een nieuwe term. Daarvóór heette het vak Systems Analyst, in het Nederlands heel verwarrend vaak systeemontwerper genoemd. De term engineering is trouwens ook verwarrend, omdat het geen technisch vak is. Tegenwoordig worden de termen business -, informatie - en soms requirements analist op verschillende plekken gebruikt, met grotendeels dezelfde betekenis.

De 66 richtlijnen zijn verdeeld over 8 hoofdstukken, geclassificeerd in drie moeilijkheidsgraden. Er is een top 10 van richtlijnen, die je bij voorkeur als eerste implementeert. Bij elke richtlijn is aangegeven wat het belangrijkste voordeel is, wat de invoering kost en wat de

toepassing kost. Er zijn weinig afhankelijkheden tussen de richtlijnen, waardoor ze in elke gewenste volgorde kunnen worden ingevoerd.

Guideline	Description	Page
3.1	Define a standard document structure	41
3.8	Make the document easy to change	60
9.1	Uniquely identify each requirement	218
9.2	Define policies for requirements management	221
6.1	Define standard templates for requirements description	144
6.2	Use language simply, consistently and concisely	147
8.2	Organise formal requirements inspections	195
8.4	Define validation checklists	200
5.2	Use checklists for requirements analysis	117
5.4	Plan for conflicts and conflict resolution	125

Figure 2.7 Top ten guidelines

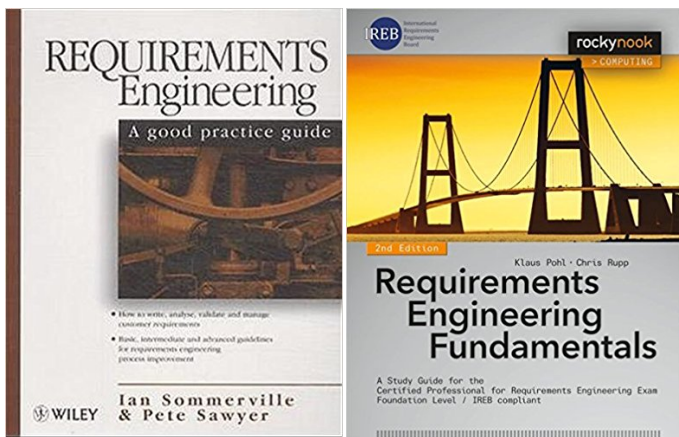
Voor deze eenvoudige structuur is gekozen, omdat die zich leent voor het geleidelijk verbeteren van het requirementsproces. Geleidelijke verandering verdiende de voorkeur boven een snelle verandering. Die zou te grote bedrijfsrisico's met zich meebrengen, omdat er niet voldoende ervaring was met procesverandering. Dit is, gezien vanuit 2018, een verrassend bescheiden opstelling, vergeleken met de voortvarende manier waarop agile methoden enkele jaren later ingevoerd zijn. De komst van scaling agile raamwerken wijst erop dat de huiver voor het veranderen van processen de laatste jaren nog verder afgenomen is, omdat daarbij niet alleen het softwareontwikkelproces, maar de hele organisatie wordt meegenomen.

De notie van procesvolwassenheid wordt geïntroduceerd, gebaseerd op de opzet van CMM, van het Amerikaanse Ministerie van Defensie, dat zich focust op het softwareontwikkelproces.

Geschiedenis

ISO 9000 en dus ook CMM bestonden al en de richtlijnen in het boek zijn in lijn met beide. Object oriëntatie was er al een jaar of tien en er is al sprake van use cases (scenario's). Op basis van het boek Rapid Application Development van James Martin was de agile methode DSDM ontwikkeld, maar die had lang niet zo'n hoge vlucht genomen als Scrum in latere jaren. Het concept user story werd een paar jaar later bedacht. De wereld was dus nog tamelijk waternoot.

IIBA en IREB bestonden nog niet en BCS hield zich nog weinig bezig met business analyse. RUP bestond nog niet onder die naam en UML werd rond dezelfde tijd door de OMG geadopteerd.



Vergelijking

Om de waarde voor ons vak in de huidige tijd te kunnen beoordelen heb ik een vergelijking gemaakt met 'het IREB boek' Requirements Engineering Fundamentals van Klaus Pohl en Chris Rupp.

Wat meteen in het oog springt is dat het IREB boek meer hiërarchie en structuur heeft. Het heeft één inhoudsopgave waar alle paragrafen in staan, terwijl in de Good Practice Guide elk hoofdstuk een aparte inhoudsopgave van de richtlijnen heeft.

Beide boeken vinden het belangrijk dat requirements documenten een standaard formaat hebben, dat past in de organisatie en het project. Ze gaan in op het gebruik van taal, templates en diagrammen. Ze gaan allebei in op het toepassen van modelleringstechnieken, zoals het use case model en het klassediagram. Beide boeken benadrukken dat je er bij het opstellen rekening mee moet houden dat er de documenten verschillende

Beide boeken benadrukken dat je er bij het opstellen rekening mee moet houden dat er de documenten verschillende doelgroepen hebben, die elk hun eigen verwachtingen hebben bij het lezen.

doelgroepen hebben, die elk hun eigen verwachtingen hebben bij het lezen.

De Good Practice Guide besteedt meer aandacht aan het gevoelig zijn voor politieke overwegingen binnen organisaties. Verder bespreken beide boeken diverse elicitatietechnieken, zoals het vinden van stakeholders en andere bronnen van requirements, het kijken vanuit verschillende gezichtspunten, het gebruik van prototypes en scenario's (Use Cases) en hergebruik van requirements. De Good Practice Guide gaat in een apart hoofdstuk dieper in op de analyse van verschillende gezichtspunten.

Beide boeken gaan uitgebreid in op het onderhandelen, valideren en prioriteren van requirements: waar liggen de systeemgrenzen, hoe gaan we om met conflicten, wat zijn de risico's? Het gebruik van checklists en interactiematrices om conflicterende en overlappende requirements op te sporen.

Ook aan requirements management besteden beide boeken gelukkig veel aandacht: unieke identificatie van requirements, traceability en management van wijzigingen aan requirements.

De Good Practice Guide besteedt een lang hoofdstuk aan requirements aan kritieke systemen (hoge betrouwbaarheid, beschikbaarheid, onderhoudbaarheid en beveiligbaarheid). Hierin wordt aandacht besteed aan het verzamelen van incidenten, de analyse van gevaren en het implementeren van het hieruit geleerde in het systeem en de cultuur.

Nadat je het boek hebt leren kennen vind je snel wat je zoekt, dankzij de hoofdstukindeling, de moeilijkheidsgraden en de top 10.

Conclusie

Mijn eerste indruk was dat het IREB boek toegankelijker is en gemakkelijker in het gebruik. En dat in de helft van de bladzijden! Maar na het bestuderen van de verschillen zie ik dat de Good Practice Guide juist heel toegankelijk is. Nadat je het boek hebt leren kennen vind je snel wat je zoekt, dankzij de hoofdstukindeling, de moeilijkheidsgraden en de top 10. En het gaat op een aantal onderwerpen dieper in dan het IREB boek. Daardoor is de Good Practice Guide beter bruikbaar om het vak te leren kennen. Bovendien leest de Good Practice Guide gemakkelijker. Dat laatste komt waarschijnlijk doordat de auteurs allebei 'native speakers' zijn, waar de IREB auteurs Duits als moedertaal hebben. Door de duidelijke structuur van de Good Practice Guide is het op ieder moment glashelder of je richtlijnen of overwegingen aan het lezen bent. Het boek is nooit geactualiseerd, maar kan volgens mij nog steeds goed van pas komen, bijvoorbeeld tijdens een sprint retrospective. En niet alleen daar.