

Wat is ICT meer dan de koffieautomaat?

Jurri Ross neemt na 30 jaar afscheid van de ICT. Jurri staat bij vriend en vijand bekend om zijn ongezouten meningen. We spraken met hem over wat er goed en wat er fout gaat in ons vakgebied.

door Reinoud de Leve en Hans Siebering

Je neemt nu afscheid van de ICT. Hoe kijk jij aan tegen de ontwikkelingen in ons vak?

Iedereen kent wel die cartoon van de schommel. De oorspronkelijke cartoon bestond geloof ik uit zes plaatjes. Inmiddels zijn er wel twaalf verschillende plaatjes van betrokkenen die op een andere manier naar de schommel kijken. Dat is wat er de laatste decennia met ons vak is gebeurd. In plaats van dat we de communicatie eenvoudiger en directer hebben gemaakt, hebben we het allemaal nog ingewikkelder gemaakt.

Als je dat zegt, zou je blij moeten zijn met de ontwikkelingen van Scrum en agile. Zij zijn ook voor minder rollen.

Dat is zeker een goed element van Scrum. Toch is Scrum meer een managementmethode. Scrum gaat vooral over hoe je een softwareproject organiseert en lost in wezen het communicatieprobleem – dat volgens mij een veel fundamenteeler probleem is – niet op. In de oude watervalmethode was het probleem dat we vaak veel te laat opleverden. We leverden toen functionaliteit die gebaseerd was op de situatie van twee jaar geleden. Met Scrum leveren we iedere twee of drie weken nieuwe functionaliteit op. Dat zijn alleen vaak zulke kleine brokjes functionaliteit dat de business er in de praktijk niet zo veel mee kan. In beide gevallen dwingen we de business om mee te gaan in onze manier van werken. Ik zeg daarmee niet dat Scrum verkeerd is. Net zo min als waterval verkeerd zou zijn. Ik pleit ervoor om het strakke regime los te laten en de planning af te laten hangen van het op te leveren product.

Daarnaast moet je bij Scrum continu in de gaten houden dat je niet bezig bent met suboptimalisatie.

Hoe komt het dat onze methodes zo weinig doeltreffend zijn?

Het belangrijkste probleem is onze communicatie. We communiceren nog steeds te wiskundig, te logisch, te veel vanuit een softwaresysteem denkend. We missen vaak het empathisch vermogen om werkelijk te begrijpen wat de problemen zijn van de mensen die dagelijks met onze systemen moeten werken. We zouden onze taal



veel meer moeten aanpassen aan de gebruikers. Dat we een model van de wereld maken, is nodig; zonder dat kun je geen ICT bedrijven. Maar te vaak gebruiken we het model alsof het de wereld is. En daar loopt het mis. ICT-ers lijken vaak niet te willen begrijpen hoe de wereld echt werkt. Als ik bij een gebruiker aan kom met een UML diagram van zijn omgeving, heb ik een grote kans dat hij het niet begrijpt. Dat maakt het onbegrip tussen de gebruiker en ons alleen maar groter. Ik gebruik maar een paar simpele diagramtechnieken, omdat iedereen die begrijpt. Stroomdiagrammen met rechthoekjes en pijltjes. Use case diagrammen met includes en extends gaan me al veel te ver.

Samenhangend met het communicatieprobleem hebben we een attitudeprobleem: we vinden onszelf veel te belangrijk. Als collectief en als individu. Het wiel wordt steeds opnieuw uitgevonden. Objectoriëntatie en

serviceoriëntatie zouden ons de mogelijkheid moeten bieden om op de schouders van reuzen te gaan staan, maar dat doen we niet. We willen niet iets gebruiken dat door iemand anders ontwikkeld is. Waarom? Het kan zijn omdat we een hekel hebben aan die ander of omdat we het veel te leuk vinden om het zelf te maken. Andere argumenten zijn dat “onze gegevens te gevoelig” zijn of dat “wat wij doen uniek” is. Wat mij betreft zijn dit allemaal kulargumenten. Alsof wij ICT-ers zo belangrijk zijn. Het bestaansrecht van ICT is het ondersteunen van de business. Wat is ICT meer dan de koffieautomaat?

Eén van de gevolgen van deze attitude is, dat het geheugen van ons vak erg kort is. De meeste van de zogenaamde innovaties waren oude wijn in nieuwe zakken. In de jaren tachtig hadden we modulair programmeren, in de jaren negentig component based development en in de jaren nul service oriented architecture. Zoek de verschillen.

Ook het geheugen binnen een project of een applicatiegroep is vaak niet goed. We documenteren meestal slecht. Omdat een werkend systeem boven documentatie gaat, is documenteren vaak een sluitpost van het project en wordt het meestal daarna ook nog eens helemaal niet onderhouden. Als er daarna een probleem ontstaat, laten we iemand in India in de code duiken, omdat wij te duur zijn om onze eigen erfenis te ontcijferen.

Kun je voorbeelden geven van hoe het misloopt?

- In de jaren negentig gingen we ergens object georiënteerd werken. Collega's stelden voor om per object een scherm te maken. Dat was logisch, want dan bevatte een scherm de gegevens die bij elkaar hoorden. Dat een gebruiker voor het afhandelen van één transactie tien schermen door moest, werd vergeten. De gebruiker haakte af. De manier waarop “logica” toegepast werd, leidde ertoe dat de gebruiker de dienaar werd van het systeem. Maar het moet natuurlijk andersom: het systeem moet de dienaar zijn van de gebruiker.
- We gingen RUP doen. We maakten use cases. Beschrijvingen, die voor de gebruiker een black box waren. En vervolgens dwongen we hem te tekenen voor iets, dat hij niet begreep. Ook hier was de mislukking al ingebouwd.

We weten wel dat we moeten communiceren. Dat houdt in dat we praten met de gebruiker. Maar dat doen we niet, we interviewen hem of, erger nog, we praten tegen hem. Dat is een heel beperkte vorm van communicatie. Zo dansen we om de essentie heen. Met rampen tot gevolg.

Wat is er nodig om het in de toekomst beter te doen?

Het ICT vak draait in essentie om communicatie. Communicatie kan niet zonder empathie. En daar ontbreekt het aan. Nog altijd ligt de nadruk op logische, rationele vaardigheden. Op vaardigheden, die belangrijk zijn in de exacte wetenschappen. Maar de wereld, die we moeten ondersteunen, is heel vaak niet logisch en niet rationeel. Emoties en persoonlijke belangen spelen altijd een rol. De vaardigheden om daarmee om te gaan, ontbreken grotendeels. We hebben beide soorten vaardigheden nodig: we moeten logisch denken én empathisch kunnen zijn. De gebruiker staat in de kou. We hebben empathie nodig. We moeten ons verplaatsen in de gebruiker. Onszelf de vraag stellen: hoe zou ik zijn werk willen doen? Wij moeten de stap naar de gebruiker maken, niet andersom.

Het goede nieuws is, dat empathie een vaardigheid is, die de meeste mensen kunt ontwikkelen.

Het goede nieuws is, dat empathie een vaardigheid is, die de meeste mensen kunt ontwikkelen. Je kunt leren je te verplaatsen in de ander. Er liggen nog steeds grote kansen voor ons vak, maar die kunnen we pas pakken wanneer we begrijpen waar de klant mee bezig is. We moeten ons verdiepen in het domein van de klant. En dat kan alleen maar door onze empathische vaardigheden te ontwikkelen en te gebruiken. Daarvoor is een cultuuromslag nodig in ons vak. Die omslag zal pas komen, wanneer het belang daarvan ingezien wordt. Vooralsnog is snel scoren nog veel te vaak de drijfveer.

