



Sander Hoogendoorn
www.sanderhoogendoorn.com

Vanmorgen bracht ik met mijn jongste zoontje Boet – net drie – een bezoek aan de dichtstbijzijnde kinderboerderij. Zijn eerste bezoek aan een kinderboerderij. Een openbaring. Toen ik hem na afloop, vlak voor zijn middagdutje vroeg of hij het leuk had gevonden, antwoordde hij heel gedecideerd: “Ik vond het niet leuk in de kinderboerderij. Er waren geen leeuwen, er waren geen tijgers en er was geen krokodil.” Teveel dierentuinen gezien. Een duidelijk geval van gemankeerd verwachtingsmanagement.

Verwachtingen

Het werkelijke leven biedt zo alweer een knappe metafoor voor het vakgebied softwaredevelopment. Hoe vaak gebeurt het niet dat de klant niet datgene krijgt wat hij eigenlijk nodig heeft. Misschien mag ik – alhoewel het inmiddels toch langzamerhand bekend mag worden verwacht – het onderwerp nog eens terugbrengen op het watervalmodel. Naar aanleiding van een recent seminar dat ik verzorg over agile software development herlas ik onlangs nog eens het prachtige white paper *Managing the Development of Large Software Systems* van Winston Royce dat hij publiceerde in de Proceedings of IEEE Westcon in het legendarische jaartal 1970. Dit beroemde artikel wordt nog altijd gezien als de bakermat van het watervalmodel. En inderdaad, in het white paper treffen we de bekende afbeelding van het model, met de fasenanalyse, ontwerp, bouw en test. Destijds baanbrekend.

De strekking van *Managing the Development of Large Software Systems* werd pas goed duidelijk toen het Amerikaanse ministerie van defensie door een engineer op basis van het white paper de standaard voor software development liet uitschrijven, onder de nietsverhullende titel DOD-STD-2167. Het is deze standaard die model heeft gestaan voor de verspreiding van het watervalmodel in tal van overheidsinstellingen, banken, verzekeraars en andere bedrijven die zich in 1970 al druk maakten over het uitvoeren van softwareontwikkelingprojecten.

Helaas, helaas voor ons vakgebied zag de auteur van DOD-STD-2167 een aantal cruciale opmerkingen van Winston Royce over het hoofd. Ik citeer. “*If the computer program in question is being developed for the first time, arrange matters so that the version finally delivered to the customer is actually the second version.*” Wat Winston Royce eigenlijk beoogde met zijn white paper is dat de eerste versie van software zelden volledig

en correct is, en dat je er te allen tijde rekening mee moet houden dat er verbeteringen en wijzigingen komen. Precies datgene wat in een traditioneel project wat getracht uit te bannen. Winston Royce zegt dat wijzigingen onvermijdelijk zijn als “*there are novel elements and unknown factors.*” En zeg nu zelf, wanneer is dat nu niet het geval in een normaal project? Royce promoot dan ook een *do-it-twice approach*. Later, in interviews verklaart hij zich groot voorstander van iteratief ontwikkelen en zegt hij dat het hem altijd heeft gespeten dat het ministerie van defensie zijn white paper zo beperkt heeft geïnterpreteerd. En datzelfde zegt overigens ook de engineer die DOD-STD-2167 heeft gefabriceerd.

En toch. Tijdens een workshop UML die ik eerder deze week deed, gaan er nog altijd stemmen op voor waterval. “Wij doen al jaren watervalprojecten, en met succes,” spreekt een deelnemer zich uit. Terug naar het onderwerp van vandaag. Verwachtingsmanagement. Uit onderzoek blijkt dat tijdens een gemiddeld project de requirements met zo’n 25% wijzigen – ja, ook in watervalprojecten. Een onderzoek dat in 1999 werd uitgevoerd over ruim vierhonderd DOD-STD-2167-projecten toont zelfs aan dat 45% van de van te voren opgestelde requirements (zoals dat hoort in waterval) nooit gebruikt wordt, en 19% maar zelden. Overigens beschouwt datzelfde onderzoek ruim 75% van deze vierhonderd projecten als gefaald. Het is maar dat je het weet. Te vaak een dierentuin beloofd, maar een kinderboerderij gekregen. «

Waardering ★★★★★

Auteur: Winston Royce
Titel: Managing the Development of Large Software Systems