

Oei, ik groei!

Hoe ziet een goed stuk software er uit? Goede software is stabiel, onderhoudbaar, gebruiksvriendelijk, vrij van bugs, schaalbaar, platform onafhankelijk (hoera, Java!), ... en nog zo wat eigenschappen. Er is echter één bijzonder soort software waarvoor alle genoemde eigenschappen nauwelijks belangrijk zijn, maar waarvoor hele andere kwaliteitscriteria gelden.

Die ene soort is de software die nog niet klaar is, software die nog afgemaakt moet worden! We kunnen het vergelijken met mensen. Wat zijn de belangrijkste kwaliteiten van mensen? Hard werken, eerlijkheid, goede communicatievaardigheden, trouw? Dat zijn kwaliteiten van volwassen mensen. Voor mensen die nog niet 'af' zijn – kinderen – hebben we hele andere kwaliteitscriteria: goed groeien en veel leren. Veel meer hoeft een kind niet te laten zien. Software die nog niet klaar is – kind-software – moet ook vooral goed zijn in groeien en leren, maar wat houdt dat in voor software?

Naast kind-software is er nog een categorie systemen waarvoor kwaliteitscriteria gelden die doen denken aan groeien en leren: simulatiesystemen. Dat zijn systemen die niet stabiel hoeven te zijn. Gebruiksvriendelijkheid, stabiliteit, betrouwbaarheid, integratiemogelijkheden en performance zijn niet relevant. Simulatiesystemen staan los van de wereld van de 'echte' systemen. Ze

worden gebruikt om te leren hoe een echt systeem moet werken. Simulatiesystemen moeten kunnen groeien, ze moeten steeds nieuwe formules, modellen of algoritmen kunnen doorrekenen. Het gekke is dat simulatiesystemen vaak gezien worden als 'leuk' en 'grappig' maar ook wordt er op neergekeken als 'niet serieus', 'het is natuurlijk niet écht ...'. Net alsof het om kinderen gaat!

'Software die nog niet klaar is' zou simulatiesoftware moeten zijn. Software die laat zien wat het uiteindelijke volwassen systeem straks gaat doen, zonder de zorgen om gebruiksvriendelijkheid, performance of onderhoudbaarheid. Dat komt allemaal later nog wel. Simulatiesoftware is geen prototype-software, maar simuleert de kern van de zaak: hoe het magazijnbeheer verloopt, hoe de transacties afgewerkt worden of hoe de nieuwe polissen verkocht worden aan een klant. Simulatiesoftware geeft de mogelijkheid om die kern van de zaak te veranderen: wat gebeurt er als we het magazijn twee keer zo hoog maken? Wat gebeurt er als we dat nieuwe verzekeringsproduct introduceren?

Hoe gaan we van kind-software of een simulatiesysteem naar een echt productiesysteem? Van kind naar volwassene? Hoe ziet puberale software er uit? Puberale software is lelijke software: die wil zich niet laten zien, sluit zich op in zijn puberkamer. Tijdens de pubertijd wordt alles grondig verbouwd.

Ineens moet aan alle kwaliteitscriteria voldaan worden, die voor volwassen software gelden. Al die eigenschappen zoals stabiliteit en gebruiksvriendelijkheid moeten toegevoegd worden zonder de kern van het kind-systeem aan te tasten. Het kind-systeem heeft niet voor niets geleerd. De eind-simulatie van het kind wordt omgebouwd tot volwassene.

Hebben we wel kind-software nodig? Kunnen we niet meteen software maken die wel al klaar is? Wat mij betreft proberen we al heel lang vol te houden dat we in een keer software kunnen maken die helemaal klaar is en moeten we dat maar eens opgeven. We zijn met z'n allen druk bezig om een stuk software helemaal uit te denken voor we aan het programmeren gaan. In veel gevallen wordt meer dan de helft van het 'bouwen' van software besteed aan het vooraf bedenken hoe die software zou moeten werken. In feite bouwen we die software al drie keer in ons hoofd of op papier voordat we aan het echte programmeren beginnen. Software wordt veel te ingewikkeld om in je hoofd te bouwen, dat gaat niet meer. Dat hoofd van ons wordt zo langzamerhand een sterk beperkende factor. Software moet spelen, groeien, puisten krijgen en dan tot bloei komen als een volwassen systeem.

*Daan Kalmeijer is consultant bij CIBIT
(daan@cibit.nl)*