

Weet Alice waar ze naartoe wil?

Peter Koelewijns liedje 'Alice, who the fuck is Alice?' uit 1995 diende als een grap op 'Living next door to Alice', een song van Smokie uit 1976. Maar die Alice bedoel ik niet. Nee, ik bedoel die van Lewis Carroll. Weet u nog? Dat mooie kinderboekje van hem uit 1865: Alice in Wonderland. Daarin vraagt Alice aan de Cheshire-kat welke weg ze zal nemen. Als Alice zegt dat ze niet weet waar ze naartoe wil, antwoordt de kat dat het dan niet uitmaakt welke weg ze neemt. Elke weg is goed.

En zo staan architecten dagelijks voor een twee-, drie- of meersprong. Technisch zijn we verantwoordelijk voor de realisatie van een product of systeem. Hierin komt een grote combinatie van subdoelen samen. Natuurlijk moet het op tijd klaar zijn en mag het niet te veel kosten. De gebruiker moet er intuïtief mee overweg kunnen. Het ontwikkelbudget kent zijn grenzen. Hoe zit het met onderhoud na ingebruikname? En het installeren in de klantomgeving, de aansluiting met de bestaande infrastructuur? De veiligheid van het product, de wetgeving, het energieverbruik, de privacy en beschikbaarheid van 24x7 zonder uitval, en ga zo maar door.

Architecten nemen haast continu beslissingen op basis van onvolledige informatie. Ze moeten dus voortdurend een weg kiezen. Dat is lastig en daarmee ook een pracht van een uitdaging. Beste Alice, pardon, archi-

tect: weet welke weg je het beste kunt kiezen! Mijn advies is een iteratieve weg langs de kwaliteitseisen. Ik zal uitleggen wat ik daarmee bedoel.

Bij kwaliteitseisen moet je denken aan *portability*, *usability*, *reliability*, *availability*, *testability*, *configurability*, *liability*, *installability*, *serviceability* en nog een aantal 'ilities' die ook wel de niet-functionele requirements heten. Als het iets is waar we als architecten verantwoorde-

lijk voor zijn, dan zijn het wel deze dingen.

Ik hoor je vragen: 'Maar hoe zit het dan met de functionaliteit?' Helemaal gelijk in. Functionaliteit hoort er ook bij. Maar dat komt wel goed. Daar herinnert onze omgeving ons vanaf het begin van een project voortdurend aan. De 'ilities' niet. Die manifesteren zich pas aan het eind van het traject. 'Fijn dat het sorteren goed werkt, maar wat een slechte performance', krijgen architecten dan te horen. Of: 'Bij het installeren kunnen we zo veel parameters instellen dat het uren kost voordat het

een beetje begint te werken.' Of: 'Aan het eind van de productielijn is het apparaat helemaal niet fatsoenlijk te testen.' En wat te denken van: 'Er is geen diagnostiek beschikbaar als er een fout optreedt.' En: 'Na integratie loopt het geheugen helemaal vol'. Van die dingen.

Laten we die als architect nu eens echt lekker in het begin al in ons ontwikkeltraject meenemen. Niet pas aan het eind, nee, vanaf het begin. Nog mooier is het als we het ook in een leuk iteratief traject verpakken. Niet alleen de functionaliteit stapje voor stapje iteratief ontwikkelen, maar ook alle non-functionals iteratief meenemen.

Laten we het volgende voorbeeld nemen: we kunnen naar een volgend kanaal zappen met onze digitale tv, maar het kost nu twee seconden en hij houdt het 187 zaps vol zonder te crashen. Dan weten we in ieder geval dat we eerst aandacht moeten gaan besteden aan robuustheid voordat we bijvoorbeeld picture-in-picture gaan toevoegen. Een tv moet toch wel minimaal veertig duizend keer achter elkaar kunnen zappen zonder om te vallen. De hoeveelheid werk om productstabieleit te verbeteren groeit namelijk exponentieel met de hoeveelheid functionaliteit.

Moraal van het verhaal: start zo vroeg mogelijk met het meten van kwaliteitsaspecten, dan weten we in ieder geval of we de goede kant opgaan of dat we moeten bijsturen. Uiteindelijk hebben we dan de Cheshire-kat helemaal niet nodig.



Ger Schoeber