

# De Zen van Doen, of: hoe shit happens

Door Lykle de Vries - [@lykle](#) - [lykle@thesisone.com](mailto:lykle@thesisone.com)

## Gewoon aan de slag gaan

Het mantra van de huidige tijd lijkt wel te zijn: doe dingen! Doe veel dingen! Denk niet te lang na, onderzoek niet te veel, maar ga direct aan de slag. Wat je onderweg leert is nuttiger dan zowat alles wat je van te voren kunt onderzoeken.

*Ever tried? Ever failed? No matter.  
Try again. Fail again. Fail better.  
- Samuel Beckett*

Het is een levenshouding die past bij idioten en bij mensen die een startup beginnen. Het is een aantrekkelijke levenshouding die goed past bij het masculiene 'ronkende' actiegerichte opgezweepte 'entrepreneurial' denken dat nog steeds in de mode is. Want gewoon iets gaan doen is voor vrijwel iedereen mogelijk. Geen machines om te kopen, geen banken om een investering bij los te peuteren. Geen zwartkijkers om te moeten overtuigen. Gewoon lekker zelf beginnen. Waar wacht je nog op?

## De democratisering van actie

Laten we wel wezen, het is ook (zeker in de westerse wereld) oneindig veel makkelijker geworden om iets te gaan doen. We zitten niet meer vast in de verplichtende structuren en zuilen van vroeger. We zijn hoger opgeleid, we hebben betere middelen tot onze beschikking en veel van wat je nodig hebt om iets te maken dat er nog niet was is nu bereikbaar voor mensen zonder (oud) geld of een groot netwerk. Veel van de traditionele poortwachters die vroeger bepaalden wat wel en niet een groter publiek bereikte, zijn bovendien ook verdwenen. Dus: alle ruimte voor het individu en diens acties!

## Los van verantwoordelijkheid

Denken vanuit een startup is heel aantrekkelijk. Er is (nog) niemand afhankelijk van je, dus je kunt op ieder willekeurig moment besluiten 'dat het anders moet'. Je kunt net zo lang en zo hard doorwerken als je zelf wilt. Je zit zelf aan het stuur. Voor veel mensen geeft het veel energie om met anderen door te bikkelen en elke dag of week weer een nieuwe stap te kunnen zetten in de ontwikkeling van je product of dienst. Dan ben je dus ineens niet meer helemaal vrij van verantwoordelijkheden. Maar de spirit van de startup is er vaak één van alles of niets. Wie niet mee kan komen, die flikker je gewoon weer uit je team. Dus die verantwoordelijkheid ten opzichte van je teamleden, die valt wel mee.

## Omdat het er nog niet is, weet niemand dat ze het willen

Het andere argument dat vaak gegeven wordt wanneer mensen gewoon maar wat gaan doen is één van vraag en aanbod. Henry Ford schijnt gezegd te hebben dat mensen een sneller paard wensten wanneer hij ze vroeg waar ze behoefte aan hadden. Paarden bestonden al, en waren derhalve het logische referentiekader voor mensen. Via die weg zou hij natuurlijk nooit de auto gerealiseerd hebben gekregen, want die bestond nog niet (Behalve in het hoofd van mensen zoals Henry Ford). Dus moet je mensen eerst iets laten zien

voordat ze kunnen bedenken dat ze nou juist dat nodig hebben wat jij voor ze gemaakt hebt. Een prima argument om vooral maar gewoon te beginnen met maken.

### **Als het mislukt, kun je in ieder geval zeggen dat je wat gedaan hebt**

De tijdsgeest waardeert snelheid. We hebben het niet zo op mensen die eerst (lang) nadenken, en pas wat gaan doen wanneer ze bedacht hebben dat het een goed idee is om iets te gaan doen. Dat was vroeger, toen het nog veel geld kostte om dingen uit te proberen, bijvoorbeeld omdat je er machines en fabrieken voor nodig hadden die eerst nog gebouwd moesten worden. Dat is nu niet meer zo. Nu zijn die obstakels weg. Informatie reist de wereld rond op lichtsnelheid. Dus we hebben ook niet de luxe meer om lang na te denken, we moeten direct handelen om te voorkomen dat een ander sneller is. Dat leidt ertoe dat sommige mensen maar liever snel verkeerd beslissen, dan te langzaam goed beslissen. De rationale achteraf is dan “dat we wel wat geleerd hebben”. En dan is het alweer goed.

(Overigens: diezelfde snelheid en actiegerichtheid gaat ten koste van ons vermogen om écht over iets na te denken. Om te extrapoleren, af te wegen, scenario's door te denken. We worden er beroerde “levensschakers” door).

## **Overzicht houden**

Er was een tijd dat ik een meester was in het programma Microsoft Project. Ik gebruikte het om grote projecten op te hakken in mijlpalen en 'deliverables'. Die deliverables hakte ik dan weer op in onderdelen, die ik weer ophakte in taken. Taken die gedaan moesten worden door mijn team-leden. Door het totale werk op die manier op te breken in stukjes, ontstond een overzicht van wat er, door wie, in welke volgorde, gedaan moest worden. Superhandig.

Behalve dat het nooit ging zoals ik van te voren dacht. Hoe goed ik ook met mijn team mijn best had gedaan om ons nieuwe product tot in het kleinste detail door te denken, en te bepalen hoe we dat dan zouden realiseren, altijd ging er iets mis in de planning. Omdat er iemand ziek werd en langer over zijn taak deed. Omdat een partij waarvan we afhankelijk waren besloot om dingen anders te gaan doen. Omdat er nieuwe wet- en regelgeving kwam waar we rekening mee moesten houden. Omdat we al doende ontdekten dat ons product helemaal niet gebouwd kon worden zoals we van tevoren gedacht hadden.

Duizend-en-één redenen konden er zijn waardoor dat fantastische projectoverzicht aan gort ging. Natuurlijk deed ik mijn best om gedurende het project bij te houden hoe het werkelijk ging. Wekelijks was er overleg tussen mij en de teamleden, en zij hadden op hun beurt weer overleggen met allerlei anderen. Zo hielden we de vinger aan de pols. Maar nooit leverde dat een tijdsbesparing voor het project op. Altijd liep het project uit ten opzichte van de oorspronkelijke planning. En hoewel ik voor mijn gevoel een compleet overzicht had, leek het wel of ik niets te zeggen had over de snelheid waarmee het project af kwam.

## **Motiveren**

Het was in diezelfde tijd dat ik mijn team-leden kon motiveren met de simpele mededeling “dat het hun werk was”. Ze waren tenslotte aangenomen om die taken te vervullen. De meesten van hen accepteerden dat ook. Ze deden hun ding, kregen hun salaris op hun rekening en gingen buiten werktijd andere dingen doen. Zoals voetballen, of stappen en veel

te laat thuis komen. Dat leverde niet alleen stoere verhalen op op de maandagochtend (die ervoor zorgden dat mijn teamleden elkaar persoonlijk een beetje leerden kennen), maar ook risico's voor mijn project in de vorm van blessures en katers. Omdat ze betaald kregen voor het werk dat ze deden en niet voor wat ze in hun vrije tijd deden, was het erg moeilijk om daar wat van te zeggen.

Sterker nog: dat werkte averechts. Toen één van mijn team-leden op een gegeven moment tekenen begon te vertonen van burn-out en ik hem adviseerde om even een maand rust te nemen, werd ik verontwaardigd opgebeld door zijn vrouw. Of ik wel wist hoe belangrijk zijn inkomen voor hun gezin was. Ze had dus kennelijk liever een man met burn-out thuis en geld op de rekening, dan een gezonde echtgenoot. (Overigens was de man niet bij mij in dienst, maar gedetacheerd. Zijn inkomen had hij dus sowieso wel behouden wanneer hij een maand rustiger aan gedaan zou hebben).

## Kleinere stappen zetten

De manier waarop ik in het verleden mijn projecten beheerde, was logisch in een tijd dat we dachten dat dingen volgordelijk uitgewerkt konden worden. Eerst een idee, dan een ontwerp, dan een work-breakdown-structure, en dan een gereed eindproduct. We noemen dat ook wel het waterval-model. In essentie betekent het dat je verschillende fases benoemt waar je chronologisch doorheen moet om het gewenste eindresultaat te bereiken. Een belangrijk argument om projecten zo aan te pakken was altijd dat je in de ontwerp-fase goedkoper van gedachten kunt veranderen dan in de productiefase. Immers: wanneer je het product eenmaal uit de machines komt, kost het veel meer moeite om een detail aan te passen dan wanneer het product alleen nog maar een lijn op een tekening is. Wederom: erg logisch wanneer je grote investeringen moet doen alvorens te kunnen gaan produceren.

Een interessant alternatief voor de waterval-methode is de agile/scrum-aanpak. Bij die aanpak werk je niet je volledige product van tevoren in detail uit, maar doe je dat in kleinere stappen. Stappen waarin je het volledige ontwerp en bouw-proces doorloopt in een korte periode. En daarna nog een keer, en nog een keer. Net zolang tot de tijd of het geld op is (of niets meer kunt verbeteren aan het eindresultaat). Met de komst van kwalitatief goede 3D-printers komt deze aanpak ineens ook binnen handbereik van de traditionele productontwikkeling.

Je zet die stappen samen met een team bestaande uit alle relevante belanghebbenden: ontwerpers, bouwers, testers en klanten. Bij de eerste iteratie maak je samen met je team de eerste basis-functies, test je ze en lever je de eerste versie van het product af. Met minimale functionaliteit. In de tweede ronde voeg je de volgende functies toe, in overleg met je team. En zo verder. Het grote voordeel van deze aanpak is dat ontwerp en realisatie heel dicht op elkaar zitten, en er in de tussentijd vrijwel niets kan veranderen. Bovendien kun je zo eenvoudig bijsturen wanneer je ontdekt dat iets niet te realiseren valt op de manier die je van tevoren bedacht had.

De agile/scrum-aanpak vereist wel een hele andere manier van denken dan de waterval-methode. Want je kunt je opdrachtgever niet meer exact beloven wat hij krijgt. Dat bepaal je immers onderweg. Wél kun je je opdrachtgever beloven dat je iets oplevert dat

volledig werkt en waar de klanten/gebruikers vanaf dag één hun stem in hebben gehad. Op diezelfde manier kun je jezelf ook niet exact beloven wat je gaat maken. Dat bepaal je namelijk binnen het team, aan de hand van doelstellingen en kaders die richting geven.

Het kan simpeler. Door een doel te stellen en te bepalen wat het eerste ding is dat je moet doen om in de richting van je doel te bewegen. En als dat gedaan is kies je het volgende ding dat je moet doen om een stapje dichterbij je doel te komen. Enzovoorts. De online wereld is ondertussen vergeven van slimme tools om deze manier van werken te ondersteunen. Denk aan gedeelde to-do-lijsten tot hele online projectmanagement-omgevingen die de agile aanpak ondersteunen. Maar vergeet ook niet de fysieke bijeenkomst waarbij je de gedachten en besluiten gewoon op papier zet en mee naar huis neemt.

## Dingen samen doen

In mijn eentje kan ik maar een bepaald aantal dingen goed voor elkaar krijgen. Voor al het andere ontbreekt me de tijd, expertise, handvaardigheid of kennis. Of combinaties daarvan. Het is dus waarschijnlijk slim om niet alles in mijn eentje te proberen te doen maar mensen te vinden die mij kunnen helpen met andere vaardigheden, kennis en ervaring dan ikzelf heb. Daarmee loop ik gelijk tegen een flinke barriere aan. Nadat ik heb toegegeven dat ik niet alles zelf kan, zal ik mensen moeten vinden die iets anders kunnen, maar waarmee ik wel samen wil werken. En zij met mij. Wanneer ik over geld beschik, kan ik die mensen wellicht op de arbeidsmarkt vinden en in dienst nemen. Maar of ik er nou geld aan uit kan geven of niet, zal ik er echter op letten of die mensen (in mijn ogen) bij mij passen en het doel dat ik voor ogen heb.

*In these troubled, uncertain times, we don't need more command and control; we need better means to engage everyone's intelligence in solving challenges and crises as they arise.*

*Margaret - J. Wheatley*

Nou zal het per persoon verschillen hoe makkelijk je met anderen samen werkt, en op welke manier je dat graag doet. Wanneer je niet zo van confrontaties houdt, zoals ik, is het best moeilijk om een ander aan te spreken op resultaten waarvan je denkt dat ze beter zouden kunnen. Zet ik onze relatie dan niet teveel onder druk? Bovendien vind ik het best moeilijk om die erfenis van vroeger (het vooraf uitdenken en ontwerpen van zaken) los te laten en een ander met onderdelen daarvan te vertrouwen. Dus ik vind het ingewikkeld wanneer ik een idee deel, en dan van een ander een reactie krijg waarmee dat idee een beetje veranderd wordt. Kan ik me dan nog net zo betrokken voelen bij het product als ik oorspronkelijk betrokken was bij het idee voor het product?

Wie wel makkelijk delegeert en confronteert is in mijn ogen sneller succesvol in het realiseren van zaken die hij/zij niet zelf kan of wil doen. Er zit continu spanning op het geven van verantwoordelijkheid aan een ander en jouw eigen beeld van hoe een goed eindresultaat eruit zou moeten zien. Die spanning is voor mij pas functioneel te gebruiken wanneer ik de ander goed heb leren kennen en vertrouwen. Wanneer ik van de ander weet wat zijn of haar motivaties zijn om mee te doen en heb geleerd hoe ik de ander kan inspireren, wordt het voor mij 'leuk' om samen te werken.

## Hippie-style werken: de doe maar, zie maar-aanpak

Het is makkelijk om, naar aanleiding van al het voorgaande, een zwart/wit-beeld te creëren van twee verschillende manieren om dingen voor elkaar te krijgen. Aan de ene kant hebben we dan de helder geformuleerde doelstellingen die omgezet worden in een nauwkeurig ontwerp en verdeeld worden over professionals die ieder hun eigen expertise inzetten om het eindresultaat volgens een strakke planning voor elkaar te krijgen. Aan de andere kant staat dan de tijdelijke groep van wat wazige idee-gedreven individuen die iets bijdragen wanneer ze kunnen en wel zien of het datgene oplevert wat ze ervan gehoopt hadden. En hoewel je als lezer hopelijk begrijpt dat die tweede zienswijze evenmin 'waar' is als de eerste, denk ik wel dat, in algemene zin, wij de nieuwe manieren van samenwerken toch nog maar ingewikkeld vinden en al snel in die tweede, wazigere hoek neigen te zetten.

Dat is natuurlijk ook niet zo gek. Want we zijn opgeleid op scholen en in onderwijssystemen die veel waarde hechten aan logica, ratio, ordening en volgorde. Geen wonder ook dat zoiets als Het Nieuwe Werken ook zoveel weerstand treft. Want hoe moeilijk is het niet om datgene wat je geleerd hebt, los te laten? Maar de crux zit hem wel daar, in die verandering van kijken naar hoe je dingen gedaan krijgt.

Ja, het is best hippie-achtig om te werken vanuit een eigen, innerlijke motivatie. Schijnbaar los van de realiteit, schijnbaar gedreven door iets anders dan commercieel belang. Nee, het is niet hippie-achtig om dat samen met anderen te doen met als bedoeling om iets waardevols toe te voegen aan deze wereld. Maar dat het hippie-achtig voélt in vergelijking met wat we ooit geleerd hebben, dat verbaast me niets. Het is echter geen doe maar, zie maar-aanpak die tot resultaten leidt. Het creëren van vertrouwen en begrip legt de basis om in een team samen te kunnen werken. Om je eigen bijdrage te kunnen leveren, op jouw manier en met gebruikmaking van jouw kwaliteiten moet je weten hoe de ander jouw inspanning zal waarderen. En je moet een reden hebben om te willen bijdragen.

## Shit happens omdat het moet

En daar zit hem denk ik het geheim van dingen doen, en shit voor elkaar krijgen. Wanneer ik niet alleen kan vertellen wát ik voor elkaar wil krijgen, maar ook wáárom, creëer ik manieren voor anderen om aan te haken bij wat ik wil. Ze kunnen dan begrijpen op welke manier hun ervaring en expertise (hun wát) een plek kan krijgen in het project, maar ze kunnen dan ook begrijpen op welke manier ik hun resultaten zal beschouwen en beoordelen. Op mijn beurt zal ik me moeten interesseren in de manieren waarop de anderen hun informatie willen ontvangen, op welke manier ze (meer of minder) aangestuurd willen worden en wat hun motivatie(s) zijn om mee te doen.

De Zen van dingen voor elkaar krijgen is vooral de acceptatie dat je van te voren niet zeker kunt weten op welke manier je het resultaat zult bereiken wanneer je dat samen met anderen doet. Het is de kunst om te leren verwoorden wát je wilt bereiken, en het wáárom daarvan te leren delen met jouw team. Dat kan inhouden dat er van alles aan de manier waarop het resultaat gerealiseerd wordt (het hóe) anders gaat dan je zelf voor ogen had, maar dat is eigenlijk nooit zo erg als je van te voren misschien dacht. Sterker nog: de 'zen' van succesvolle projectregisseurs is wellicht dat ze een soort 'force field' neerleggen waarbinnen

dingen zich kunnen realiseren, en tegelijkertijd voorkomen dat de teamleden volledig ontspoord raken. Ik ben geen psycholoog en ook geen neuroloog, maar ik zou me kunnen voorstellen dat anderen het doorhebben wanneer ze iemand met een dergelijk 'force field' tegenkomen, en daarom met het project mee willen doen.

Maar laat ik niet te zweverig eindigen. Het geheim van dingen voor elkaar krijgen, de kunst om shit te laten gebeuren zit hem in het weten wat wel en niet handig is om te doen:

- ***Het is in het algemeen wel nuttig om plannen te maken, maar vooral als denkproces.***

Hoe hangen dingen met elkaar samen? Wat gebeurt er als externe parameters veranderen (scenario's, gevoeligheidsanalyses)? Als je daar bedreven in bent, kun je sneller en beter reageren op veranderingen - zelfs op veranderingen die je niet precies hebt voorzien en doordacht. Je overziet het speelveld dan gewoon beter.

- ***Soms is het beter om even stil te staan dan keihard een nogal willekeurige kant op te rennen.***

Vooraf om te zien hoe bepaalde ontwikkelingen uitpakken of om op een andere manier onzekerheid te reduceren.

- ***In de fysieke wereld is agile/scrum (nog) vaak gewoon erg onpraktisch.***

Als je van de ene kant van de rivier naar de andere wilt kun je in de definitie- en ontwerpfasen naar hartenlust itereren en simuleren (brug, tunnel of hyperloop? hefbrug, hangbrug of draaibrug?), maar zodra je begint te bouwen is dat niet meer handig.

- ***Een onderneming (in de zin van: iets nieuws gaan doen) is nog weer anders. Een onderneming heeft een open einde, is nooit af.***

Daar is klein beginnen en zo snel mogelijk verbeteren van levensbelang. En dat organiseren (anderen laten meewerken) kan eigenlijk alleen door te sturen op de intentie - iedereen moet heel goed snappen wat de bedoeling is, anders weten ze niet wat de goede kant is.

Sturen op intentie, houd die term in de gaten. Zodra we de huidige golf van 'werken met talent' een beetje te boven zijn, gaan we (leren) sturen op intentie. Wie dit stuk gelezen heeft, weet vandaag al wat hem of haar te doen staat om succesvol invloed te hebben op de wereld van morgen.