

Agile Projecten

De veranderende rol van de Projectmanager



Door: Iwan Eising, december 2016

Introductie

Het straatbeeld in het IT-landschap is aan het veranderen. Ontwikkelingen als ‘breedband internet’, ‘de Cloud¹’, ‘mobile web’, en ‘internet of things (IoT)’, hebben ervoor gezorgd dat we anders tegen IT aan zijn gaan kijken en vooral hoe we software zijn gaan ontwikkelen.

Het falen van grote IT-projecten en daarmee gepaard gaande mediaspektakel, de miljoenen die schijnbaar verkwist worden aan overheidsprojecten en de opkomst van ‘startups’ vereisen dat we wendbaarder omgaan met softwareontwikkelingen.

Een belangrijke verandering in IT-projecten vindt plaats in het projectmanagement. Waterval wordt steeds meer ingewisseld voor Agile. De rol van de projectmanager is steeds minder relevant en zien we langzamerhand verdwijnen. Spraken we tot voor kort nog van een projectmanager in softwareprojecten, tegenwoordig hebben we het over Product Owners en Scrum-masters. Maar is de rol van de projectmanager echt overbodig geworden in de wereld van zelfsturende teams en productorganisaties?

De definitie van Cloud is helder beschreven door NIST, de belangrijkste karakteristieken zijn hieronder gegeven.

Essential Characteristics of the Cloud by NIST¹

- *On-demand self-service.*
- *Broad network access.*
- *Resource pooling.*
- *Rapid elasticity.*
- *Measured service.*

Historie

In de afgelopen 20 jaar is de Waterval methode in diskrediet geraakt en zijn we hippe nieuwe manieren van software ontwikkelen gaan omarmen. Manieren als Agile, Scrum en DevOps. Het zijn de nieuwe *silver bullets* die alle problemen oplossen. Na de technische zilveren kogels *EAI (Enterprise Application Integration)*, *SOA (Service Oriented Architecture)* en *BPM (Business Process Modeling)*, die ook allemaal niet zaligmakend bleken te zijn, zijn we organisatorische veranderingen gaan nastreven. *Lean* is momenteel de meest voorkomende verandering in (IT-)organisaties. Door de toepassing van concepten uit Lean Manufacturing in IT-processen worden efficiëntie slagen gemaakt om softwareontwikkeling goedkoper te maken.

Al deze veranderingen en ontwikkelingen resulteren in een situatie waarin traditionele softwareprojecten steeds meer verdwijnen. De rol van de traditionele projectleider komt daardoor steeds vaker te vervallen. In sommige gevallen wordt er binnen organisaties al niet meer over de rol ‘projectleider’ gesproken.

¹ Voor een goede en vooral gangbare definitie van Cloud en Cloud computing hanteer ik de definitie van NIST: <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>

In dit artikel leest u mijn observaties en bevindingen als IT-Architect en Strateeg bij verschillende uiteenlopende bedrijven. Van startups met enkele werknemers tot conglomeraten van enkele duizenden werknemers in de IT in een verscheidenheid van markten en branches. Wat u niet leest is de theorie zoals die in allerlei boeken, magazines, nieuwsbrieven en websites staat beschreven. Dit artikel is geschreven vanuit de praktijk. Theorie is ter illustratie en niet zozeer ter onderbouwing. De reden hiervoor is dat de praktijk altijd weerbarstiger is dan de theorie. Los daarvan, de theorie is gebaseerd op modellen, en modellen zijn vaak een slap aftreksel van de werkelijkheid.

Controverse

Recentelijk heb ik een artikel voor mijn blog afgerond met als titel *“De teloorgang van de projectleider in de DevOps wereld”*. De strekking van het artikel is dat de projectleider, door de komst van DevOps, zijn beste tijd wel gehad heeft. Hoewel gechargeerd, is de scheidslijn tussen project en operatie diffuus geworden door de komst van Agile werkwijzen.

De Cloud

Door de komst van ‘de Cloud’ is het mogelijk om een internationaal opererend bedrijf te worden vanuit de zolderkamer. Investerings in IT-infrastructuur is door de Cloud niet meer nodig. Bedrijven als Amazon, Microsoft en Google bieden een infrastructuur dienst aan die met het succes van hun klanten meegroeit. Met zogenaamde free-tiers en andere startup-programma’s (bv Amazon Activate, Microsoft BizSpark en Google Launchpad) is er geen reden om niet vanuit de Cloud te opereren.

Kijken we naar het globale ‘startup-klimaat’, dan zie we dat in de periode 1995 – 2005 het ondenkbaar was om zonder een flinke financiële injectie van een investeerder een internetbedrijf te starten. Sinds de komst van de eerste echt publieke Cloud, Amazon AWS, in 2005 is het mogelijk om voor een paar tientjes per maand een dienst te lanceren. Wanneer het succes groot is, kan er vanwege de elasticiteit van de Cloud en het on-demand self-service karakter van de Cloud extra capaciteit worden bijgeschakeld. Daarbij betaal je net als met de electriciteitsrekening alleen dat wat je daadwerkelijk gebruikt. Het is niet langer noodzakelijk dat de leverancier van een IT-dienst ‘deep pockets’ heeft.

Wendbaarheid

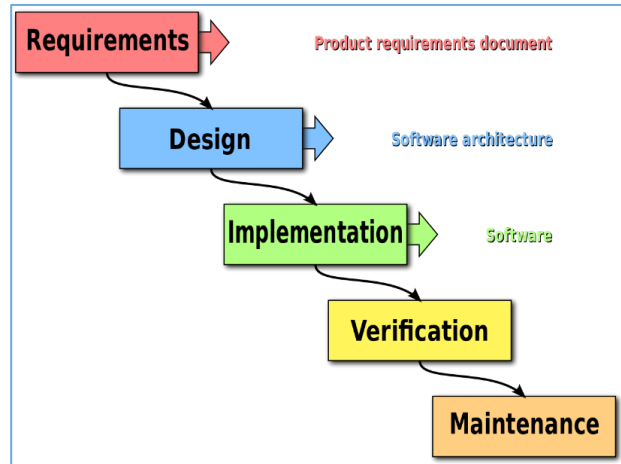
Door de elkaar steeds sneller opvolgende ontwikkelingen in de digitale wereld, moeten bedrijven zich steeds sneller kunnen aanpassen om de boot niet te missen. Product lanceringen mogen geen jaren meer duren. De consumentenmarkt moet op steeds kortere termijn van nieuwe mogelijkheden worden voorzien. Kleine wendbare bedrijven hebben hierbij een oneerlijke concurrentiepositie, een gebrek aan bureaucratie maakt het gebrek aan financiële middelen meer dan goed. Door de komst van de Cloud geldt; *“Hoe wendbaarder hoe succesvoller”*. Sinds 2010 realiseren grote bedrijven zoals financiële dienstverleners, mediabedrijven en tot op zekere hoogte overheden zich dat ze zich moeten aanpassen. Willen ze niet onder de voet gelopen worden door kleinere spelers, zullen ze wendbaarder moeten worden. Zoals Werner Vogel, CTO van Amazon, het plastisch stelt:

“Death by a thousand cuts.”

We zien dat bedrijven als ING, KLM, Wehkamp en Essent, het traditionele pad van de Waterval projecten en de grote meerjarige programma’s verlaten en ‘Agile’ zijn gaan werken. Daarmee is vaak ook de rol van de projectleider binnen deze organisaties radicaal veranderd. Je zou kunnen zeggen dat de projectleider zoals we die kennen niet meer terug te vinden is.

Waterval

De Waterval methode is eigenlijk de oervorm van projecten in de IT. Dat komt voort uit het feit dat IT-projecten in beginsel werden uitgevoerd met de kennis en ervaring uit niet IT-projecten. De eerste IT-projecten zijn hardware-projecten waarbij de software als integraal onderdeel van de hardware werd ontwikkeld. Bij de introductie van generieke hardware wordt software onafhankelijk van de onderliggende hardware ontwikkeld. Bij zowel softwareprojecten als hardware projecten wordt de Waterval methode gehanteerd.



De Waterval methode refereert aan de grafiek die je kunt tekenen voor projecten die volgens deze methode worden gerealiseerd. Na een analyse van het idee wordt de oplossing gespecificeerd. Resultierend in de requirements. Vervolgens wordt er een ontwerp voor de oplossing gemaakt. Deze wordt vervolgens geïmplementeerd door het projectteam. Als laatste wordt de oplossing in productie genomen en vindt er onderhoud plaats. In het diagram hierboven is dit proces te zien. We zien de toepassing van dit proces vaak als de ontwikkelkosten relatief hoog zijn. Bij de ontwikkeling van fysieke producten, zoals computer hardware, is dit vaak het geval. De investering wordt terugverdiend door zo snel mogelijk zoveel mogelijk eenheden te verkopen.

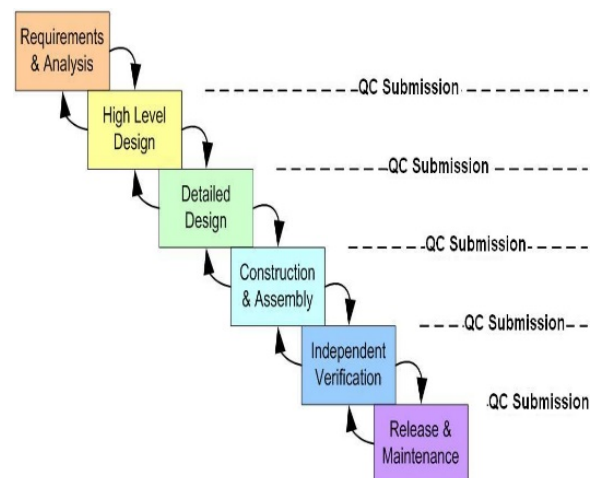
Bij de ontwikkeling van standaard-software kunnen we ervanuit gaan dat de ontwikkeling van de software tussen de 9 en 18 maanden zal duren. Vaak met uitschieters naar boven. Projecten waarbij de tijdslijn langer dan 18 maanden is noemen we al snel programma's. Die vervolgens worden opgedeeld in meerdere nauw met elkaar samenhangende projecten. De verschillende projecten worden dan weer volgens de Waterval methode gerealiseerd.

We zien dat bedrijven waarin de IT-organisatie is georganiseerd in functionele teams ook sterk de neiging hebben om volgens de Waterval methode te (blijven) werken. Deze teams optimaliseren hun eigen processen met als doel zo efficiënt mogelijk te presteren binnen het softwareontwikkeling proces.

Iteratief

Door het falen² van Waterval projecten, zien we vooral in de jaren 90 het iteratief uitvoeren van IT-projecten de kop opsteken. De notie van voortschrijdend inzicht is de belangrijkste drijfveer voor het omarmen van het iteratieve proces.

Het achterliggende idee van iteratief werken is de kans op falen binnen projecten minimaliseren. Kortom, dat door tussentijdse kwaliteitsborging de algehele kwaliteit verbeterd wordt. Hoe eerder ontdekt wordt



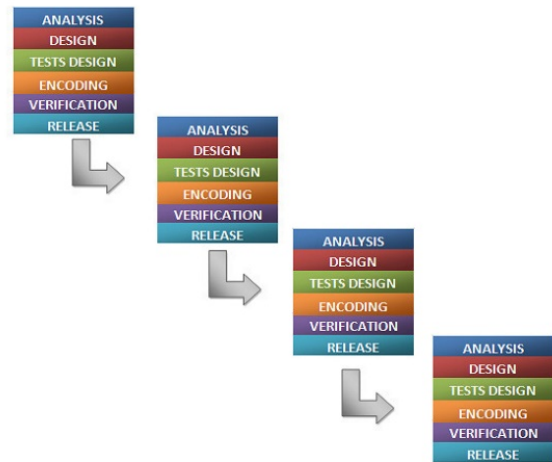
² Projecten falen wanneer ze niet op tijd worden opgeleverd, dan wel niet met het initiële budget uitkomen en daardoor een onacceptabele ROI krijgen. Dat laatste is een significante nuancering.

dat het resultaat niet haalbaar is, des te goedkoper de corrigerende actie. Net als bij de Waterval methode is ook de iteratieve werkwijze een directieve manier van omgaan met IT. In essentie betreft het niet meer dan een verfijning van de Waterval methode, de watervallen zijn kleiner geworden.

Agile

Agile werken is een manier van werken die uit de frustratie van softwareontwikkelaars naar voren is gekomen. Frustratie over het feit dat ze niet tot nauwelijks software ontwikkelen waar gebruik van kan worden gemaakt.

Agile methoden, zoals Scrum, gaan ervanuit dat de klant, of de gebruiker, 'wispelturig' is. Of liever gezegd, naarmate de ontwikkeling van de software vordert, verandert de wens van de gebruiker. Wat eerst van groot belang leek te zijn, kan na verloop van tijd een achterhaalde eis zijn. Of een ander softwaresysteem biedt reeds een vergelijkbare oplossing. Ook kan veranderde wet- en regelgeving de prioriteit van bepaalde functionaliteit beïnvloeden. Bij Scrum, net als andere Agile methoden, ligt de nadruk op werkende software met waarde voor de business. Er wordt daarom aan die requirements gewerkt waarmee de meeste waarde voor het bedrijf wordt gecreëerd. Na een eerste oplevering, de MVP, worden nieuwe functionaliteiten op regelmatige basis toegevoegd. Met andere woorden een requirement, vaak feature genoemd, wordt volledig uitgewerkt tot het punt dat de software in productie zou kunnen worden genomen. Daarna wordt er aan een nieuwe feature gewerkt.



Het grote voordeel van Agile methoden is dat op ieder gewenst moment het project stop gezet kan worden. De investering hoeft dan niet te worden gezien als desinvesteringen, de gerealiseerde software is tenslotte voorzien van de belangrijkste features, die productiekwaliteit hebben. Bij de verschillende Waterval varianten is er bij een vroegtijdige beëindiging van het project nog geen sprake van bruikbare software en er is dus wel van een desinvestering.

Agile Manifesto

Op de volgende bladzijde staat het manifest van de Agile beweging³, dit manifest is in 2001 opgesteld en gepubliceerd door toonaangevende personen uit de softwareontwikkeling. Dat laatste is van belang omdat het manifest geschreven is vanuit een software-engineering perspectief in plaats van een projectmanagement perspectief. De implicaties voor projectmanagement en projectmanagers zijn nergens geformuleerd noch meegenomen. Het uitgangspunt van dit manifest is betere software ontwikkelen binnen IT-projecten om zo investeringen in IT minder vaak te laten resulteren in desinvesteringen.

³ <http://Agilemanifesto.org>

De eenvoud van het manifest is de kracht van de werkwijze. In een aantal simpele uitspraken wordt exact uit de doeken gedaan hoe men desinvesteringen in de IT tegen wil gaan. En daarmee loopt het ook meteen behoorlijk spaak. Allereerst wordt het manifest vaak niet goed gelezen. Daar waar 'over' staat, wordt in veel gevallen 'instead' gelezen. De voordelen zullen hierdoor niet of nauwelijks worden behaald en de tegenstanders krijgen een 'zie je nou wel' momentje. Het manifest biedt daarbij ook geen handvatten voor hoe je nu Agile zou moeten werken.

Vanaf 2005 zien we dat binnen kleine teams volgens Agile principes wordt gewerkt bij de ontwikkeling van internettoepassingen. Toepassingen die mee moeten kunnen veranderen

met de grillige karakter van het internet. Rond 2010 gaan ook grote bedrijven Agile werken en omdat deze bedrijven daarnaast druk zijn met het outsource van hun IT, worden project aanbesteed met de eis dat ze Agile worden uitgevoerd. De onderliggende contracten zijn echter gebaseerd op traditionele softwareontwikkeling. In de contractering wordt nog steeds uitgegaan van Waterval projecten terwijl de uitvoering volgens Agile methoden wordt gedaan. Deze mismatch tussen contract en uitvoering resulteert in nagenoeg alle gevallen tot voortijdig afbreken van de samenwerking tussen opdrachtgever en –nemer. Pas recentelijk zien we dat Agile projecten worden gedaan op basis van Agile contracten.

Scrum

Scrum is een van de meest populaire manieren om Agile te werken. Het beschrijft een algemene werkwijze met zijn eigen rituelen en projectorganisatie. Omdat Scrum een methode is die door en voor softwareontwikkelaars is bedacht, vindt het veel aftrek in softwareprojecten. Vooral het recalcitrante karakter van Agile en Scrum maakt dat het veel wordt toegepast in teams die werken aan nieuwbouw trajecten. Daarbij is het niet conformeren aan staande processen des te interessanter voor relatief jonge softwareontwikkelaars. Wie wil er tenslotte bij een stoffige organisatie werken die nog in monolieten denkt en waar openlijk wordt toegegeven dat COBOL nog lang niet dood is?⁴

Scrum is sinds enkele jaren min of meer gemeengoed in de softwareontwikkeling. Vooral met de adoptie van Scrum door grote bedrijven die hun websites en onlinediensten ontwikkelen in Scrum-teams, kan worden gezegd dat Scrum de Agile werkwijze van het moment is.

Manifesto for Agile Software Development

WE ARE UNCOVERING BETTER WAYS OF DEVELOPING SOFTWARE BY DOING IT AND HELPING OTHERS DO IT.

THROUGH THIS WORK WE HAVE COME TO VALUE:

INDIVIDUALS AND INTERACTIONS OVER PROCESSES AND TOOLS

WORKING SOFTWARE OVER COMPREHENSIVE DOCUMENTATION

CUSTOMER COLLABORATION OVER CONTRACT NEGOTIATION

RESPONDING TO CHANGE OVER FOLLOWING A PLAN

THAT IS, WHILE THERE IS VALUE IN THE ITEMS ON

⁴ Een belangrijke reden voor bedrijven als ING en KLM om 'all the way' te gaan is om jong talent te trekken, talent dat tot voor kort niet meer bij een grote organisatie wilde werken.

Een belangrijke reden voor deze grote bedrijven om Scrum te omarmen is het feit dat het 'boek van Scrum' een duidelijke werkwijze voor Scrum-teams definieert. Er wordt in de 'Official Scrum Guide'⁵ tot in de kleinste details uitgelegd hoe je Scrum zou kunnen doen. Dit maakt het mogelijk om Scrum op grote schaal te introduceren in grote organisaties. Iets wat veel lastiger is wanneer een methode leunt op best-practices. Echter de tekst uit de Scrum boeken wordt letterlijk gevolgd bij de implementatie, met een falende IT-organisatie tot gevolg. Scrum is ondanks de duidelijke verwoording van implementatie, een denkwijze. Eentje die tot doel heeft desinvesteringen in IT te minimaliseren. Dat door de toepassing van een 'lightweight' en 'Simple to understand'⁶ methode, de overhead van IT-projecten niet langer de resultaten van het project in de weg staat. Een overhead die er nog wel is, omdat Scrum alleen de softwareontwikkeling zelf betreft en niet of nauwelijks het voortraject of de in beheername adresseert.

Scrum vs. Prince2

Net als Scrum is Prince2 ook een werkwijze en net als Prince2, moet je Scrum ook tunen naar de eigen organisatie. Voor beide methoden wordt dit vaak vergeten. Scrum gaat daarbij zover dat ieder Scrum-team zijn eigen interpretatie van Scrum zou moeten hanteren. Er wordt gestreefd naar een zo hoog mogelijke mate van autonomie van een team.

In mijn ervaring worden projectleiders net zo vaak op Prince2 als op Scrum cursus gestuurd met het idee dat na certificering de projectleider weet hoe het project geleid moet worden. Uiteraard is dit nagenoeg nooit het geval. Dit omdat men de theorie leert en niet de praktijk ervaart. Anders dan bijvoorbeeld bij LEAN waarbij alleen het eerste certificeringsniveau zonder praktijkervaring kan worden behaald. Volgende niveaus vereisen een aantoonbare praktijkervaring naast theoretische kennis van de materie.

Zowel Scrum als Prince2 hebben tot doel desinvesteringen in IT te minimaliseren. Het grote verschil tussen Scrum en Prince2 is echter dat Prince2 als uitgangspunt heeft dat '*control*' de sleutel tot succes is en Scrum uitgaat van '*governance*'⁷ als de sleutel tot succes. We moeten ons realiseren dat Prince2 is ontwikkeld vanuit het perspectief van projectmanagement; Door projectmanagement te verbeteren Waterval projecten minder vaak te laten falen. Er is expliciet niet gekeken naar hoe de softwareontwikkeling beter gedaan kan worden. Scrum doet het precies andersom.

De conclusie is dan ook dat Scrum en Prince2 goed samen kunnen gaan. Iets wat je in veel grotere organisaties ziet gebeuren sinds ze enkele jaren geleden Agile methoden zijn gaan omarmen. IT-projecten worden volgens de Prince2 manier opgezet, vaak volgens bestaande processen, en uiteindelijk wanneer het softwareprojecten betreft volgens Scrum gerealiseerd⁸.

⁵ <http://www.scrumguides.org/scrum-guide.html>

⁶ Neem notie van het feit dat Scrum helaas ook 'Difficult to master' is.

⁷ Door duidelijke richtlijnen omtrent hoe om te gaan met voortschrijdend inzicht, veranderende prioriteiten en nieuwe eisen en wensen van de klant en/of gebruiker maakt Scrum het proces wendbaar en Prince2 controleerbaar.

⁸ Zie een heldere uiteenzetting over Prince2 en Scrum en hoe ze complementair zijn op <http://www.projectmanagementexpert.nl/scrum-versus-prince2/>

Project Management

Waterval Projecten

Waterval projecten worden gemanaged door projectmanagers. Na een gedegen analyse van de requirements neemt de projectmanager de taak op zich om deze te realiseren en vervolgens beschikbaar te stellen aan de gebruiker. Als onderdeel van in productiename, zal de projectmanager de software opleveren aan een beheer-organisatie.

In meer complexe projecten wordt de projectmanager vaak bijgestaan door een technisch projectleider. In situaties waar een project onderdeel uitmaakt van een groter programma, valt de projectmanager vaak hiërarchisch onder een programmamanager. De rol van de projectmanager is voornamelijk het faciliteren van het uitvoerende team. De coördinatie van activiteiten van het uitvoerende team en andere functionele teams uit de organisatie valt binnen deze rol.

De rol van de projectmanager is dus zowel intern (het team) als extern (de organisatie) gericht. De projectmanager is daarbij budgethouder voor het project en is in veel gevallen verantwoordelijk voor de samenstelling van zijn team. De hiermee gepaard gaande consequenties voor het budget vallen ook onder de verantwoordelijkheid van de projectmanager.

Om de resultaten van de projectmanager en het uitvoerende team te kunnen beoordelen worden er KPI's (Key Performances Indicators) afgesproken. Veel voorkomende KPI's zijn:

- Mate van budgetoverschrijding.
- Mate van tijdoverschrijding.
- Aantal defecten gevonden in de verschillende fasen van het project (test-, acceptatie en productiefase)

Vaak wordt de projectmanager ook verantwoordelijk gehouden voor de kwaliteit van de oplevering. Deze wordt aan de hand van het aantal defecten in de eerste periode na oplevering bepaald. Dit is om te borgen dat de software niet 'over de muur wordt gesmeten' en de beheer organisatie dus met 'de rommel' van het projectteam wordt opgezadeld.

Agile Projecten

Agile projecten vallen onder de verantwoordelijkheid van de productmanagers, of in Scrum termen; Product Owners. Deze wordt bijgestaan door de Scrum-master. De Product Owner is anders dan de projectmanager, ook verantwoordelijk voor het verkrijgen van de requirements. Eventuele verduidelijking van requirements valt de Product Owner ook ten deel. Daarnaast is de Product Owner verantwoordelijk voor het succes van de software in productie. Deze verantwoordelijkheid betreft niet

Project Manager

<https://nl.wikipedia.org/wiki/Projectleider>

- Een **projectleider of projectmanager** is iemand die verantwoordelijk is voor het succesvol realiseren van een project. Een projectleider realiseert het project door het coördineren van en/of aansturen van projectmedewerkers, het projectmanagement.
- Projectleider is een generieke rol, die men in verschillende vakgebieden tegenkomt. Zo bestaan er bijvoorbeeld projectleiders binnen civiele techniek, bouwkunde en ICT.
- In het verleden waren projectleiders veelal gepromoveerde specialisten op hun vakgebied. Tegenwoordig wordt de rol van projectleider meer gezien als een apart specialisme.

alleen het aantal defecten in de eerste periode na in productie name, maar gedurende de volledige levenscyclus van de software. De bruikbaarheid van de software en waarde creatie voor de business behoren tot de succesfactoren van de software en zijn daarmee de verantwoordelijkheid van de Product Owner.

De rol van Product Owner is niet nieuw en is praktisch niet te onderscheiden van de rol van productmanager in meer traditionele omgevingen. Het onderscheidende aspect van de Product Owner in Agile projecten is de actieve betrokkenheid tijdens de ontwikkeling van de software. De Product Owner bewaakt op dagelijkse basis de kwaliteit en bruikbaarheid van de software. Hij heeft het mandaat om te bepalen wat er gemaakt moet worden en wanneer dit van acceptabele kwaliteit is.

De Product Owner is geen projectmanager en zal in beginsel daarmee ook niet de coördinerende taken van de projectmanager vervullen. Hoewel stakeholder management wel een onderdeel van de rol van Product Owner is, beperkt dit zich voornamelijk tot het verkrijgen van (heldere) requirements. De Scrum-master vervult echter de intern gerichte coördinerende rol van de projectmanager. Het team wordt in de werkzaamheden bijgestaan door de Scrum-master, hij faciliteert het team en draagt zorg voor de typische 'rituelen' die Scrum rijk is. Zoals de dagelijkse stand-up meeting, de Sprintreview en -demo, de Retrospectives etc. De extern gerichte taken vinden we niet benoemd in Scrum. Vanuit de gedachte dat een Scrum-team zichzelf volledig moet kunnen bedruipen en een verregaande mate van autonomie heeft, is er in theorie weinig tot geen behoefte aan een rol die de externe factoren van een project coördineert.

Product Owner

<http://www.nationaleberoepengids.nl/Productmanager>

De Productmanager [red. Product Owner] is verantwoordelijk voor het succes van een bepaald product en heeft zeggenschap over de ontwikkeling, productie en marketing daarvan. Zodoende begeleidt de Productmanager het product vanaf het begin en weet deze dus ook enorm veel over het product te vertellen. Het is de taak van de Productmanager om de commerciële, lange termijn doelen in het oog te houden en het product op de beste manier te verkopen. De functie van een Productmanager kan op vele manieren ingevuld worden, maar de kern is altijd het verzekeren van succes voor een bepaald product, door middel van de ontwikkeling, de marketing, maar ook de goede budgettering. Al deze elementen waar een Productmanager op moet letten, zorgen er uiteindelijk voor dat het product of de productlijn rendabel is en dat ook blijft.

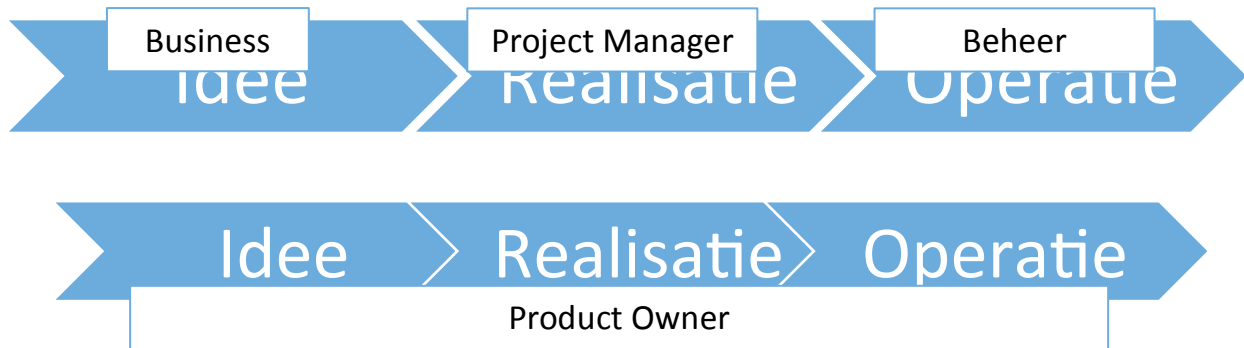
In organisaties waar Waterval projecten worden gedaan is de mate van verzuiling binnen de IT-organisatie erg hoog. Functionele teams adresseren specifieke aspecten van het IT-landschap en werkzaamheden worden door teams van functionele experts uitgevoerd. In Agile omgevingen wordt juist gestreefd naar multidisciplinaire teams die deze werkzaamheden volledig autonoom kunnen uitvoeren. Dat betreft bijvoorbeeld de inrichting van infrastructuur, de uitvoering van tests, of het beheer van databases. Organisaties die Waterval vervangen door Agile hebben echter de traditionele expertise teams in hun IT-organisatie. Hier zien we dus nog steeds de behoefte aan een projectmanager die de coördinatie tussen werkzaamheden van het projectteam en de functionele teams op zich neemt. De projectmanager is echter niet verantwoordelijk voor het (uitblijven van) succes van het project. Hij is ook niet langer budgetverantwoordelijk voor de ontwikkeling van de software. Dit ligt bij de Product Owner, deze is verantwoordelijk voor de ontwikkeling van de software, het product.

Om de resultaten van de Product Owner te kunnen beoordelen worden er KVI's (Key Value Indicators) afgesproken. De Product Owner wordt ultimo verantwoordelijk geacht voor het slagen of falen van het project aan de hand van deze KVI's. Veel voorkomende KVI's zijn:

- Mate van terugverdienen van de investeringen, ROI
- Snelheid waarmee de velocity⁹ van het Scrum-team of de Scrum-teams stabiliseert

Verandering in Focus

Er is dus een belangrijke verandering in focus tussen de projectmanager en de Product Owner. Het meest in het oog springende is misschien wel het verschil tussen KPI enerzijds en KVI anderzijds. Dit verschil hangt nauw samen met het verschil in verantwoordelijkheid. Voor de projectmanager is de scope 'het project'. Voor de Product Owner is de scope 'het product', cradle-to-grave dus.



Het belangrijkste verschil in focus is daarmee dus het verschil tussen 'Want' en 'Need'. De projectmanager richt zich op wat de klant wil, en de Product Owner op dat wat de klant nodig heeft. De projectmanager zal samen met het projectteam een kritische blik werpen op requirements maar dan vooral in termen van kosten (tijd en geld) en haalbaarheid. Dit is de expertise van de projectmanager, zo kostenefficiënt mogelijk de gevraagde functionaliteit opleveren. De Product Owner zal samen met het projectteam een kritische blik werpen op requirements maar dan vooral in termen van prioriteit, business waarde en bruikbaarheid. De expertise van de Product Owner is, zo kostenefficiënt mogelijk business waarde te creëren. De Product Owner kijkt hierbij naar ROI en probeert deze te optimaliseren.

Verandering in Organisatie

Kijken we naar de organisatorische kant van het verhaal, dan zien we de volgende verschillen tussen de projectmanager en de Product Owner:

Projectmanager	Product Owner
De projectmanager komt uit de IT-organisatie. De projectmanager moet begrijpen hoe je een IT-oplossing binnen de organisatie in productie krijgt.	De Product Owner komt uit de business. De Product Owner moet begrijpen met welk(e) product(en) business waarde gecreëerd kan worden.
De projectmanager levert de oplossing niet zozeer op aan de klant of gebruiker maar aan het beheer. De exploitatie van de oplossing ligt bij de beheerorganisatie.	De Product Owner levert de oplossing aan de gebruiker. Met de belofte dat het product gebruikt kan blijven worden, ergo de Product Owner draagt zorg voor de juiste werking van het product gedurende de periode dat het product in gebruik is.
De projectmanager zal worden beoordeeld door het IT-management aan de hand van de KPI's.	De Product Owner zal worden beoordeeld door het management aan de hand van de KVI's.

⁹ Velocity is a metric that predicts how much work an Agile software development team can successfully complete within a two-week sprint (or similar time-boxed period). (<http://whatis.techtarget.com/definition/Agile-velocity>)

Tenslotte

De rol van de projectmanager is niet van het toneel verdwenen. Waterval projecten blijven bestaan en IT-organisaties zullen nog lang zijn opgedeeld in functionele teams. De coördinerende rol van de projectmanager zal dus nodig blijven. Wel zien we dat organisaties die al geruime tijd bezig zijn met Agile en Scrum inmiddels afscheid van de Scrum-master nemen. De Scrum-teams zijn ervaren genoeg dat de rol van Scrum-master niet meer nodig is. In deze organisaties zijn projectmanagers nog steeds nodig, omdat er vanwege de verzuiling nog steeds de coördinerende rol nodig is.

Met de komst van DevOps zien we wel dat er aan de verzuiling in IT-organisaties een einde komt. Multidisciplinaire teams die ook taken van de beheerorganisatie uitvoeren werken aan producten, cradle-to-grave. Door de komst van deze productteams is de projectmanager in softwareland niet meer nodig. Projectportfolio's worden inmiddels al ingeruild voor productportfolio's en IT verschuift steeds vaker weer naar de business. Het product-denken begint ook in grote organisaties een feit te worden, IT als geldmachine in plaats van kostenpost.

Is met de verschuiving van IT terug naar de business een eind gekomen aan de CIO? In mei 2012, verscheen een artikel op Apptio¹⁰ waarin het einde van de CIO zoals we die tegenwoordig kennen zal verdwijnen. Een jaar later verscheen op InfoWorld.com een artikel¹¹ over de veranderende rol van de CIO waarin feitelijk wordt gesteld dat de CIO zijn beste tijd heeft gehad. En in juni van dit jaar verscheen op de website redtoo.com een artikel¹² dat uit de doeken doet hoe de opkomst van DevOps de rol van de CIO binnen organisaties ondermijnd. Hierin wordt gewag gemaakt van BizDevOps, business, development en operations samen in een team. Hoewel ik de introductie van de term "*BizDevOps*" meer illustratief vind voor het niet begrijpen van wat DevOps is, is de strekking van het artikel op redtoo.com absoluut interessant. Slaan we Gartner eropna, dan is de CIO, maar ook de projectmanager een rol die in Mode 1 van hun bimodal organisatiemodel te vinden is¹³. Daarmee dus een rol die past bij de 'oude' wereld. Een wereld waarin we software ontwikkelen met een lange levenscyclus en waarin het niet noodzakelijk is om deze, regelmatig, aan te passen. Een onhoudbare situatie gezien de recente veranderingen in wet- en regelgeving. Naleving van GDPR (General Data Protection Regulation)¹⁴ wordt steeds strenger gecontroleerd en andere wet- en regelgeving¹⁵ omtrent data wordt steeds relevanter door de opkomst van Big-Data. Allemaal ontwikkelingen die wendbaarheid vereisen. Ontwikkelingen die in de tijd waarin de monolieten van de Mode 1 wereld werden gemaakt niet meegenomen zijn in de requirements van de Waterval projecten.

¹⁰ <http://www.apptio.com/blog/end-cio>

¹¹ <http://www.infoworld.com/article/2612642/it-strategy/the-end-of-the-cio-as-we-know-it---and-it-feels-fine.html>

¹² <https://www.redtoo.com/ch/blog/devops-the-end-of-the-cio-organization/>

¹³ <https://itarchitectureandstrategy.blogspot.nl/2016/11/the-bimodal-misconception-why-gartner.html>

¹⁴ https://en.wikipedia.org/wiki/General_Data_Protection_Regulation

¹⁵ Naast de bekende regelgeving omtrent privacy, is ook de archiefwet, het recht 'om vergeten te worden' en allerlei fiscale wetgeving van belang.

Conclusie

In de sterk veranderende wereld van de softwareontwikkeling zien we dat bedrijven steeds vaker Waterval verruilen voor Agile methoden om hun projecten in te richten. De opmars van Agile gaat zo ver dat de verzuiling van IT-organisaties uit de afgelopen 50 jaar teniet wordt gedaan en plaats maakt voor multidisciplinaire teams. De rol van de projectmanager wordt hierdoor steeds minder relevant. In organisaties waarin van projectteams naar productteam wordt overgestapt zien we de projectmanager helemaal van het toneel verdwijnen. Is er voor deze rol dan nog wel een plek in de IT? Waterval en verzuiling zullen blijven bestaan en daarmee ook de noodzaak tot coördinerende rollen. De aanwezigheid van projectmanagers in softwareontwikkeling verschuift echter wel van mainstream naar niche.

Referenties en verklaring van terminologie

BPM – Business Process Modeling

Business Process Modeling (vaak aangeduid met het Engelse woord Business process modeling (BPM)) is een verzameling methoden en technieken om bedrijfsprocessen in kaart te brengen. Deze modellen worden gebruikt in de bedrijfskunde met name in het projectmanagement en in de analyse en ontwerp van informatiesystemen.

<https://nl.wikipedia.org/wiki/Procesmodellering>

Cloud Computing

Cloud computing is het via een netwerk – vaak het internet – op aanvraag beschikbaar stellen van hardware, software en gegevens, ongeveer zoals elektriciteit uit het lichtnet. De term is afkomstig uit de schematechnieken uit de informatica, waar een groot, decentraal netwerk (zoals het internet) met behulp van een wolk wordt aangeduid.

https://nl.wikipedia.org/wiki/Cloud_computing

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

<http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>

EAI – Enterprise Application Integration

Enterprise Application Integration (EAI) is een term die vooral in de bedrijfswereld wordt gebruikt om technologie aan te duiden die bedoeld is om verschillende bedrijfsapplicaties te laten samenwerken.

https://nl.wikipedia.org/wiki/Enterprise_application_integration

ESB – Enterprise Service Bus

Een Enterprise Service Bus (ESB) is een architecturale softwareconstructie (pattern) waarmee de communicatie tussen de afnemers van diensten (“service”) en aanbieders hiervan, vereenvoudigd wordt. De ESB biedt hiertoe aan de kant van de aanvrager een met de aanvrager afgesproken interface

aan, dit kan een webservice zijn, maar bijvoorbeeld ook een SMTP (e-mail) interface. Aan de kant van de aanbieder zal de ESB via de interface die met de aanbieder is afgesproken communiceren. Zo kan het dus gebeuren dat een aanvrager van een dienst op een compleet andere wijze met de ESB communiceert dan de ESB met de aanbieder. Onderstaande figuur geeft dit schematisch weer.

https://nl.wikipedia.org/wiki/Enterprise_service_bus

GDPR – General Data Protection Regulation

The General Data Protection Regulation (GDPR) (Regulation (EU) 2016/679) is a Regulation by which the European Parliament, the Council and the European Commission intend to strengthen and unify data protection for individuals within the European Union (EU). It also addresses export of personal data outside the EU. The primary objectives of the GDPR are to give citizens back the control of their personal data and to simplify the regulatory environment for international business by unifying the regulation within the EU. When the GDPR takes effect it will replace the data protection directive (officially Directive 95/46/EC) from 1995. Perhaps confusingly for some, there is a new directive as well as a new regulation; it will apply to police procedures, which will continue to vary from one Member State to the other.

The regulation was adopted on 27 April 2016. It enters into application 25 May 2018 after a two-year transition period and, unlike a Directive, it does not require any enabling legislation to be passed by national governments.

https://en.wikipedia.org/wiki/General_Data_Protection_Regulation

LEAN

Lean manufacturing of lean production, beter bekend als lean, is een managementfilosofie in het vakgebied van operations management die erop gericht is om maximale waarde voor de klant te realiseren met zo min mogelijk verspilling. Door verspillingen te elimineren gaan de operationele kosten omlaag, wat in het algemeen leidt tot een verbetering van het bedrijfsresultaat.

https://nl.wikipedia.org/wiki/Lean_manufacturing

MVP – Minimal Viable Product

In product development, the minimum viable product (MVP) is a product with just enough features to gather validated learning about the product and its continued development. Gathering insights from an MVP is often less expensive than developing a product with more features, which increase costs and risk if the product fails, for example, due to incorrect assumptions. The term was coined and defined by Frank Robinson, and popularized by Steve Blank, and Eric Ries. It may also involve carrying out market analysis beforehand.

https://en.wikipedia.org/wiki/Minimum_viable_product

Scrum

Scrum is een flexibele manier om (software)producten te maken. Er wordt gewerkt in multidisciplinaire teams die in korte sprints, met een vaste lengte van 1 tot 4 weken, werkende (software) producten opleveren. Scrum is een term die afkomstig is uit de rugbysport. Bij een scrum probeert een team samen een doel te bereiken en de wedstrijd te winnen. Samenwerking is heel belangrijk en men moet snel kunnen inspelen op veranderende omstandigheden.

Scrum wordt vaak gebruikt bij producten waarvan de klant c.q. gebruiker nog niet goed weet wat hij wil en waarbij men al doende leert om de eisen en wensen beter te beschrijven en in bruikbare producten om te zetten. Vaak weet men pas wat men wil als men het eerste product, het prototype, ziet en dan worden alsnog de eisen aangepast. Scrum heeft de flexibiliteit om met laat wijzigende eisen en wensen om te gaan. Scrum valt onder de Agile-softwareontwikkeling.

[https://nl.wikipedia.org/wiki/Scrum_\(softwareontwikkelmethode\)](https://nl.wikipedia.org/wiki/Scrum_(softwareontwikkelmethode))

SOA – Service Oriented Architecture

Service-oriëntatie, vertaling van *Service-oriented architecture* (SOA), is een architectuurmodel, geen technologie op zich. Centraal bestaat een SOA-opgebouwd systeem uit servicecontracten. Hierbij is sprake van afnemers van diensten en leveranciers.

<https://nl.wikipedia.org/wiki/Service-ori%C3%ABntatie>

XML – eXtensible Markup Language

XML, Extensible Markup Language (XML) is een standaard van het World Wide Web Consortium voor de syntaxis van formele opmaaktalen waarmee men gestructureerde gegevens kan weergeven in de vorm van platte tekst. Deze presentatie is zowel machineleesbaar als leesbaar voor de mens. Het XML-formaat wordt gebruikt om gegevens op te slaan (zoals in het OpenDocument-formaat) en om gegevens over het internet te versturen.

https://nl.wikipedia.org/wiki/Extensible_Markup_Language

Aanbevolen literatuur

Naast de linkjes en referenties in dit artikel kan ik de volgende boeken en websites aanraden.

Literatuur

- [The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win](#) (Gene Kim)
- [The Goal: A Process of Ongoing Improvement](#) (Eliyahu M. Goldratt)
- [Critical Chain: A Business Novel](#) (Eliyahu M. Goldratt)
- [The Agile Architecture Revolution: How Cloud Computing, REST-Based SOA, and Mobile Computing Are Changing Enterprise IT](#) (Jason Bloomberg)
- [The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses](#) (Eric Ries)
- [The CIO Paradox: Battling the Contradictions of IT Leadership](#) (Martha Heller)

Websites

- <http://zapthink.com/>
- <https://intellyx.com/>
- <http://itarchitectureandstrategy.blogspot.nl/>
- <http://Agilemanifesto.org/>
- <http://www.scrumguides.org/scrum-guide.html>

De auteur

Iwan Eising, is getrouwd met een fantastische vrouw en vader van twee uitzonderlijk leuke zoontjes. Hij verdeelt zijn vrije tijd onder water tussen de vissen, in de bioscoop tussen de cinefielen, online tussen de gamers of op de Columbuschool in Almere tussen hoogbegaafde kinderen waar hij assisteert bij de lessen science. Dit allemaal als hij niet zelf aan het programmeren is. Zijn devies is: *“Dat wat er ontworpen wordt moet ook gebouwd kunnen worden.”*

Hij is te vinden op Twitter (@Arc_E_Tect) en op LinkedIn (<https://nl.linkedin.com/in/iwaneising>), Hij publiceert regelmatig via zijn blog Arc-E-Tect (<https://itarchitectureandstrategy.blogspot.nl/>)



Voor dit artikel ligt het copyright bij de auteur, Iwan Eising. De gehanteerde licentie is een Creative Commons licentie ([CC BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/))