

DE SLEUTEL TOT PROJECTMANAGEMENT

AgilePM® ontsloten

Richard Weber & Rolf Teekens

De Sleutel tot Projectmanagement AgilePM[®] ontsloten

Richard Weber & Rolf Teekens

Auteurs: Richard Weber & Rolf Teekens
Redactie: Katja Urban
Vormgeving & cover ontwerp: Stef Ringoot
Illustraties: Richard Weber
Uitgever: Key Result Publishing, Almere
4de druk, April 2024
ISBN 9789081482769 (softcover)
ISBN 9789081482776 (ebook pdf)
NUR 801



Gebaseerd op: Het officiële AgilePM handboek

”AgilePM® - Agile Project Management Handbook v2”

Uitgave oktober 2014

Agile Project Management is a trade mark of The APM Group Limited. The APMG-International AgilePM and Swirl Device logo is a trade mark of The APM Group Limited.

DSDM®, Atern® and AgilePM® are registered trade marks of Dynamic Systems Development Method Limited in the United Kingdom and other countries.

PRINCE2® is a registered trade mark of AXELOS Limited.

Diagrammen gemarkeerd met ‘Bron: AgilePM Handboek’ zijn gebaseerd op het officiële handboek ‘Agile Project Management Handbook v2’ van oktober 2014. ©DSDM Consortium 2014. Reproduced under licence from The APM Group Limited.

Copyright © April 2024. Key Result BV, tenzij anders aangegeven.

Alle rechten voorbehouden. Niets uit deze uitgave mag worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand of openbaar gemaakt, in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door fotokopieën, opnamen of enige andere manier, zonder voorafgaande schriftelijke toestemming van de uitgever.

Eind december 2012 begon het weer te knagen: we wilden een nieuw boek schrijven. Inmiddels waren we actief met AgilePM aan de slag. In gesprek met cursisten kwamen we tot de conclusie dat een nieuw, Nederlandstalig studieboek over AgilePM een geweldig idee zou zijn. Een handzaam, beknopt boekje met alle belangrijke aspecten. Bestemd voor iedereen die meer wil weten over Agile in projecten. Toegankelijk en aantrekkelijk geschreven en met mooie praktijkervaringen.

Pffft... waar waren we aan begonnen? Net als de vorige keer kostte het weer veel tijd. Schrijversblokkades, leesblindheid, verschil in schrijfstijl, het is ons allemaal overkomen. Ondanks de moeilijke momenten bleven we enthousiast doorwerken. En dat enthousiasme bleek aanstekelijk: relaties overstelpten ons desgevraagd met prachtige, illustratieve praktijkervaringen. Tot in de laatste week kwamen nog mooie verhalen binnen.

We zagen de inhoud en vormgeving onder onze handen groeien en wisten daardoor weer waar we voor buffelden: anderen wegwijs maken in het mooie vak dat projectmanagement heet. En dan overheerst het gevoel van blijheid en tevredenheid. Met trots presenteerden we onze nieuwste titel en inmiddels de 4^{de} herziene versie:

De Sleutel tot Projectmanagement - AgilePM® ontsloten

Allereerst en vooral willen wij onze partners en kinderen bedanken voor hun eindeloze geduld. Vele avonden en zelfs vakantiedagen hebben zij ons moeten missen voor het goede doel.

Onze dank gaat ook uit naar Katja Urban voor haar input, correctie en finale controle. En natuurlijk Stef Ringoot voor de nieuwe vormgeving. Beiden hebben deze 4^{de} versie tot een echte verbetering gemaakt. Dank.

Tot slot willen wij ook alle relaties, collega's, en cursisten bedanken voor hun input. Velen van hen hebben een prominente plek gekregen in het boek en komen persoonlijk aan het woord.

Richard en Rolf

Voor je ligt ‘De Sleutel tot Projectmanagement - AgilePM® ontsloten’, een handzame en praktische uitgave van de theoretische grondslag van Agile Project Management. Het is het eerste Nederlandstalige studieboek bestemd voor iedereen die in projecten agile wil toepassen of die zich voorbereidt op het AgilePM® Foundation examen. Tegelijkertijd is het boek ook een ideaal naslagwerk voor wie in de praktijk met AgilePM of Scrum werkt.

AgilePM is van oorsprong gebaseerd op DSDM Atern, een aanpak primair gericht op softwareontwikkeling; net als Scrum dus sterk gericht op ICT-projecten. Is AgilePM dan typisch weer zo’n ICT-bevlieging? Absoluut niet! Met de komst van het DSDM Agile Project Framework (2014) - een doorontwikkeling van DSDM Atern - zijn veel ICT-specifieke elementen verdwenen. Wij juichen dat toe, want daarmee is AgilePM ook veel toegankelijker geworden voor niet ICT-projecten. Als rasechte projectmanagers hebben wij op voorhand geen oordeel over een methode of aanpak: alles heeft voor- en nadelen. Wat het beste is voor het project en de klant, staat voorop. Het merendeel van AgilePM is zeer geschikt om toe te passen in andere branches en projecten, uiteraard wel afgestemd op de behoeften van het project. Maar dat geldt voor elke aanpak.

In dit boek vormt de theorie uiteraard de basis, maar deze is verrijkt met duidelijk herkenbare verdiepingen en praktijkervaringen. De stof is daarmee begrijpelijk en toegankelijk geworden. Het handzame uiterlijk geeft je misschien de indruk dat het in één avondje uit te lezen valt. We hopen je ertoe te verleiden om er meer tijd aan te besteden.

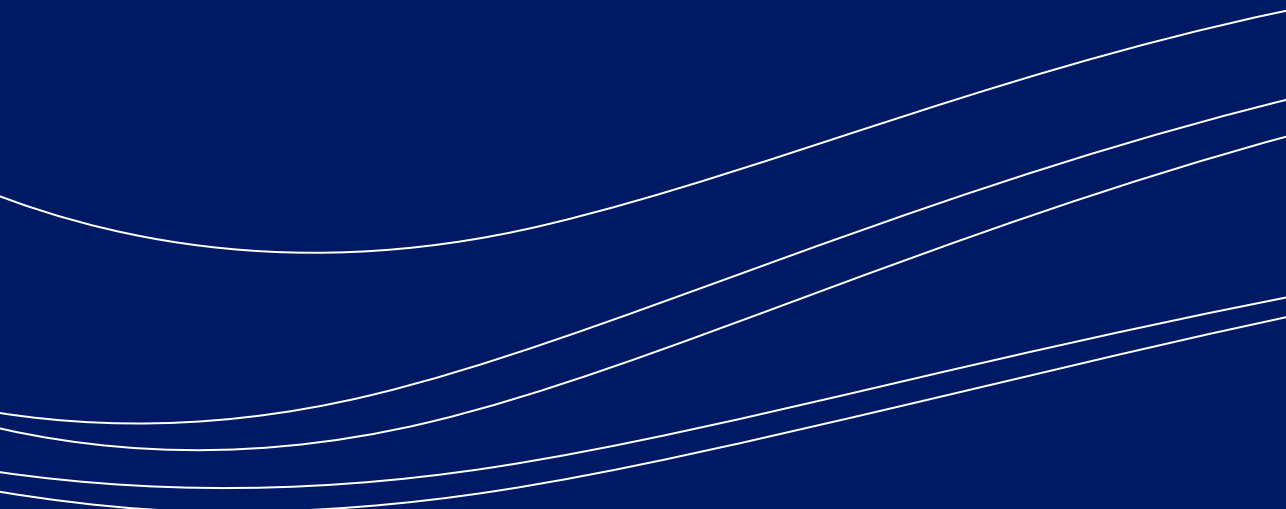
DSDM voerde diverse verbeteringen door met de komst van het Agile Project Framework. Tegelijkertijd veranderden ook de exameneisen voor AgilePM Foundation. Maar we zijn ook kritisch. Opvallend genoeg besteedt het Agile Project Framework geen enkele aandacht meer aan configuratiemanagement. Terwijl versiebeheer, informatiebeveiliging en continuïteit juist bij de agile manier van werken super belangrijk zijn. Daarom handhaven wij configuratiemanagement wel. Ook risicomanagement behoudt zijn plek, maar is in dit boek meer een verdiepingsonderwerp. Deze hoofdstukken zijn gemarkeerd met een ©: voor het Foundation-examen niet echt relevant, maar ze geven een volledig beeld van hoe je agile praktisch toepast.

Ga met dit boek de uitdaging aan project met agile succesvoller te maken en laat ons weten hoe dat voor jou werkt!

Richard Weber & Rolf Teekens

Almere, maart 2024

Overzicht	9
1. Inleiding	10
2. Over AgilePM	20
3. Voorbereiden op AgilePM	33
Projectorganisatie	37
4. Mensen	38
Projectlevenscyclus	59
5. Processen	60
6. Producten	65
Werkwijzen	75
7. Timeboxing	76
8. MoSCoW prioritering	81
9. Iteratief ontwikkelen	86
10. Workshops	90
11. Modelleren	93
Projectmanagement - Planning	95
12. Plannen	96
13. Eisen	101
14. Schatten	106
Projectmanagement - Control	111
15. Voortgangsbewaking	112
16. Projectbeheersing	115
17. Kwaliteit leveren	122
18. Testen	128
Projectmanagement - Extra	133
19. Risicomanagement ©	134
20. Configuratiemanagement ©	137
Bijlagen	143
Bijlage A. Producten & Verantwoordelijkheden	144
Bijlage B. Vragenlijst Projectaanpak	145
Bijlage C. Termen: Scrum versus AgilePM	147



Overzicht

1. Inleiding

1.1 Het Agile Manifest

On February 11-13, 2001, at The Lodge at Snowbird ski resort in the Wasatch mountains of Utah, seventeen people met to talk, ski, relax, and try to find common ground and of course, to eat. Representatives from Extreme Programming, SCRUM, DSDM, Adaptive Software Development, Crystal, Feature-Driven Development, Pragmatic Programming, and others sympathetic to the need for an alternative to documentation driven, heavyweight software development processes convened. What emerged was the Agile Software Development Manifesto.

De 17 onafhankelijke denkers en gebruikers van methodieken voor softwareontwikkeling en programmeren waren het over veel dingen niet eens. Maar ze bereikten consensus over vier belangrijke waarden.

Manifest voor Agile Software Ontwikkeling

Wij laten zien dat er betere manieren zijn om software te ontwikkelen, door in de praktijk aan te tonen dat dit werkt en door anderen ermee te helpen. Daarom verkiezen we:

Mensen en hun onderlinge interactie	boven	Processen en tools
Werkende software	boven	Allesomvattende documentatie
Samenwerking met de klant	boven	Contractonderhandelingen
Inspelen op verandering	boven	Een plan volgen

Hoewel wij waardering hebben voor al hetgeen aan de rechterkant staat vermeld, hechten wij méér waarde aan wat aan de linkerkzijde wordt genoemd.

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

In het Agile Manifest staat dat de ondertekenaars betere manieren van softwareontwikkeling aan het ontdekken zijn door deze manieren in de praktijk te gebruiken en anderen te helpen om ze toe te passen. “Het mag niet meer voorkomen, dat theoretici bepalen hoe praktijkmensen zaken moeten aanpakken. In de praktijk vinden we uit wat werkt en de academische wereld is er om ons te bestuderen en te verklaren waarom het werkt. We luisteren vooral naar mensen die daadwerkelijk software ontwikkelen in de praktijk!”

Een korte toelichting op het Agile Manifest:

Mensen en hun onderlinge interactie boven processen en tools

Mensen die het gehele proces meemaken, creëren de meest succesvolle software. Mensen die in een omgeving werken van vertrouwen en respect, waar relevante informatie vrij en transparant is, kunnen tot excellente interacties komen. Processen en gereedschappen zijn minder belangrijk voor succes dan goede mensen en excellente interactie.

Werkende software boven allesomvattende documentatie

In ICT-projecten ontstaat waarde pas, nadat software is opgeleverd. De focus hoort dan ook te liggen op werkende software. Documentatie is waardevol voor succes, maar werkende software is onvoorwaardelijk. Daarmee meet je voortgang en valideer je aannames.

Samenwerking met de klant boven contractonderhandelingen

Het doel is: waarde creëren voor de klant. Dit doe je, door zeer nauw met de klant samen te werken tijdens de ontwikkeling. Nieuwe inzichten van de klant verwerk je zo snel en effectief mogelijk. Het formele contractuele proces is belangrijk, maar voor succes is het veel belangrijker dat je effectief samenwerkt met je klant en het meest waardevolle realiseert.

Inspelen op verandering boven het volgen van een plan

Gedurende een project ga je heel veel leren. Aan het begin van een project weet je immers het minste. Vroegtijdig en frequent aannames valideren, geeft nieuwe kennis die je verwelkomt en gebruikt. Het plan is belangrijk om een gemeenschappelijke richting te creëren, maar voor succes is effectief omgaan met veranderingen en nieuwe inzichten veel belangrijker.

De 'Agile-gedachte' leidde tot enkele concepten die je regelmatig zult tegenkomen:

Timeboxing	Timeboxing is een techniek voor het managen van tijd en een belangrijk hulpmiddel om dingen gedaan te krijgen. Binnen de beschikbare hoeveelheid tijd, de zogenaamde 'timebox', doe je je uiterste best om zoveel mogelijk eisen en wensen werkend op te leveren. Na afloop van de timebox lever je producten op die af zijn. Producten die niet (geheel) af zijn, niet. De keus is dan om die eisen en wensen te laten vallen of om ze in een volgende timebox mee te nemen. Je zult eisen en wensen dus moeten prioriteren.
Iteratief ontwikkelen	Een product ontwikkel je meestal niet in één keer direct conform de verwachting. Met iteratief ontwikkelen, werk je in kleine stapjes aan het product, test je de werking en verifieer je regelmatig of het conform verwachting verloopt, voordat je verder gaat. Het product komt zo al doende kort-cyclisch tot stand.
Incrementeel ontwikkelen	Bij incrementeel ontwikkelen deel je het project op in deelproducten die je afzonderlijk van elkaar oplevert. Zo kun je feedback uit de praktijk gelijk toepassen, een deeloplossing eerder in gebruik nemen en eerder beginnen met waarde realiseren.

Deze drie begrippen zijn nauw met elkaar verbonden en gebruik je meestal in combinatie.

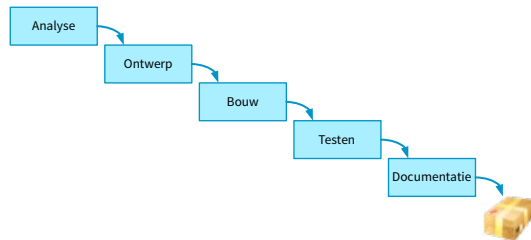
1.2 Historie

Met wat meer inzicht in de historische context van softwareontwikkeling kun je deze concepten op waarde schatten. Tenslotte liggen daar ook de wortels van Agile. Later zullen we laten zien dat de principes van Agile zonder al te veel moeite of creativiteit breder bruikbaar zijn buiten softwareontwikkeling, zoals bij productontwikkeling, reclame en innovatie. En aangezien bijna ieder project ook raakvlakken heeft met ICT, is de stap niet erg groot.

Softwareontwikkeling kent ruwweg drie primaire benaderingen: lineair, iteratief en incrementeel.

Lineair

Het project kent opeenvolgende fases, waarbij beperkte overlap en terugkoppeling mogelijk is. De nadruk ligt op planning, tijd, deadlines, budget en één geïmplementeerd systeem opleveren. Het project blijft tijdens de uitvoering strak onder controle door uitgebreide documentatie, formele reviews en autorisatie aan het einde van iedere fase en voor aanvang van de volgende fase. De Watervalmethode en System Development Methodology (SDM) zijn hiervan de meest bekende voorbeelden. De benadering is top-down: eerst alle eisen tot in detail uitwerken en dan pas ontwikkelen. Een volgende fase begint pas als de voorgaande is afgerond en goedgekeurd. Velen ervaren de lineaire benadering als star en bureaucratisch, al zie je de watervalmethode ook vandaag de dag nog veel terug in projecten.



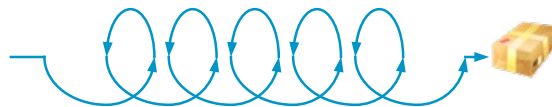
Figuur 1: De lineaire benadering

Iteratief

De iteratieve benadering (cyclisch of spiraalmodel) ontstond als reactie op de lineaire benadering. Deze combineert zowel ontwerp, prototyping als realisatie door telkens een onderdeel van de software aan te pakken. Iedere cyclus doorloopt dezelfde stappen voor ieder deelproduct en op ieder niveau van uitwerking, zonder onderscheid tussen een conceptbeschrijving van de werking of programmeren van deelapplicaties. Het product groeit zo in cycli naar het eindproduct. Dit biedt meer flexibiliteit, snellere terugkoppeling en betere aansluiting bij wat de gebruiker/afnemer wil. Rapid Application Development (RAD) volgt de iteratieve benadering. RAD is - voor zover bekend - de eerste methode die timeboxing toepast. Meer over timeboxing volgt later.

i·te·ra·tie (de; znw.)

1. herhaalde set van stappen.



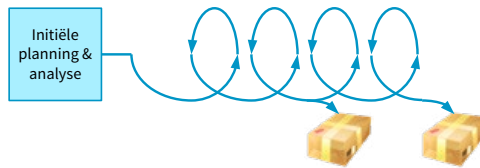
Figuur 2: De iteratieve benadering

Incrementeel

Als snel na de opkomst van RAD ontstonden mengvormen. De incrementele benadering combineert het beste van de lineaire en de iteratieve benadering. Het kan dan gaan om een serie fases, waarin je in iedere fase volgens de watervalmethode werkt en per fase de oplossing uitbreidt. Een andere combinatie is dat je concept, eisen en architectuur eerst volgens de lineaire aanpak bepaalt. Vervolgens ga je met een iteratieve aanpak aan de slag en dat leidt dan bijvoorbeeld met prototyping en doorontwikkeling tot het eindproduct. DSDM en Scrum zijn voorbeelden van deze incrementele benadering.

in-cre-men-teel (bnw.)

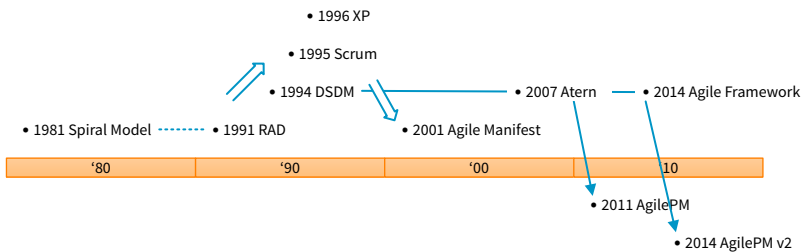
1. kleinschalige innovatie door iets nieuws toe te voegen aan iets bestaands.
2. stapsgewijze verandering zonder het einddoel expliciet te noemen.



Figuur 3: De incrementele benadering

Zet deze ontwikkelmethodes op een tijdslijn en je ziet de ontstaansgeschiedenis van agile met de meest bekende methodes.

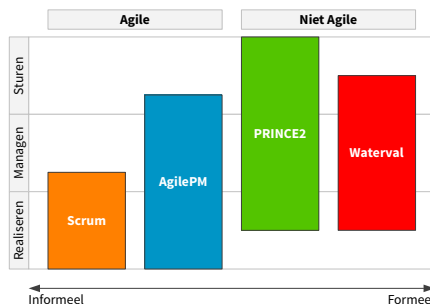
Overzicht



Figuur 4: Historisch perspectief Agile & AgilePM

1.3 Agile & Projectmanagement

Er zijn verschillende methoden of raamwerken die agile zijn. Agile is dus veel meer een paraplu dan een methode. Is welke smaak je kiest, een kwestie van persoonlijke voorkeur? Vanuit het oogpunt van projectmanagement is er wel degelijk verschil. Dit wordt duidelijk, wanneer we ook PRINCE2 en de klassieke watervalmethode in het plaatje meenemen.



Figuur 5: Positionering van AgilePM

Kijk je naar Scrum (enorm populair bij ICT-projecten), dan tref je daar geen projectmanagement in aan. Het team is immers zelfsturend en ‘empowered’, zoals ze het zo mooi zeggen. Scrum gaat daarmee uit van de ideale situatie: het team kan, mag en wil. Dat gaat prima in een omgeving die niet veel aanvullende eisen stelt (bijvoorbeeld wet- en regelgeving) en binnen een project dat niet groot is qua omvang en aantal betrokken partijen. Ook heeft het zijn voordelen om grote, complexe projecten op te delen in kleinere deelprojecten: meer overzicht, eerder resultaat.

Het DSDM Agile Project Framework is een raamwerk voor agile projectrealisatie waarmee ook in grotere projecten je de juiste oplossing op tijd kunt leveren. Waar Scrum meer uitgaat van minimalisme, gaat het DSDM Agile Framework juist meer uit van de realiteit. Bij grotere projecten met meerdere teams is afstemming tussen teams en enige sturing hard nodig door afhankelijkheden. Bovendien kan wet- en regelgeving soms projecten tot meer vastlegging dwingen. Projectmanagement komt daarom in het DSDM Agile Project Framework wel degelijk aan bod.

Juist door de behoefte aan projectmanagement en de roep vanuit de markt om andersoortige projecten ‘agile’ aan te pakken, is AgilePM ontstaan: een subset van het DSDM Agile Project Framework, specifiek voor projectmanagers. Vanaf nu hanteren we daarom het woord ‘AgilePM’.

Net zoals bij vele andere methoden geldt ook bij AgilePM dat het gezonde verstand moet leiden: gebruik wat echt nodig is en laat weg wat overbodig is. Maar: ontkracht de kern van het gedachtegoed van agile niet. Sommige aspecten van AgilePM zijn namelijk essentieel om succesvol te kunnen zijn, zoals timeboxing, iteratief en incrementeel realiseren, ruimte voor eigen verantwoordelijkheid en zelforganisatie, en veel en rijke communicatie.

1.4 AgilePM versus Klassiek

Een belangrijk verschil tussen AgilePM en de klassieke benadering is de wijze waarop het omgaat met de stuurvariabelen. Stuurvariabelen zijn die aspecten die je tijdens een project actief moet controleren en managen om het project succesvol te maken.

De vier primaire stuurvariabelen zijn:

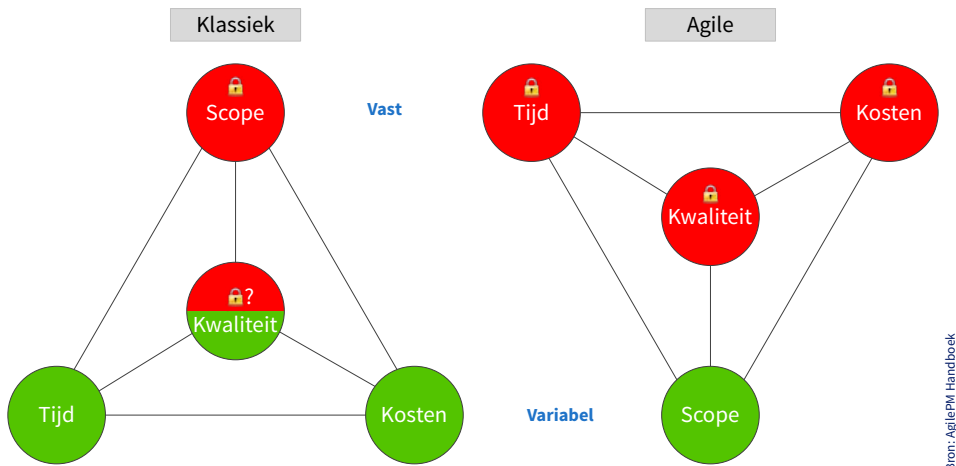
- Tijd (doorlooptijd en inzet)
- Kosten (geld, budget)
- Kwaliteit (opleveringsniveau)
- Scope (eisen & wensen, requirements, features)

Deze vier stuurvariabelen zijn nauw met elkaar verbonden. Wil je een hogere kwaliteit? Dan kost dat tijd en geld. Wil je meer functionaliteit? Dan kost dat ook meer tijd en geld. Het liefst zou je al deze variabelen fixeren: lekker zeker. Dat kan wel, maar dan komt het risico veelal bij de leverancier te liggen en die zal dat tot uitdrukking brengen in zijn prijs. Wie betaalt dat? Je kent vast de volgende gezegdes wel: “goedkoop is duurkoop” of “voor een dubbeltje op de eerste rang willen zitten”. Kortom: alles vast zetten, is geen optie. Het is beter van tevoren af te spreken om in gesprek te blijven met elkaar, als de stuurvariabelen gaandeweg veranderen.

In de klassieke benadering en zelfs bij PRINCE2 is het startpunt de scope: je brengt in kaart wat de klant wil en vertaalt dat naar specificaties. In de watervalmethode zit zelfs een fase waarin je de specificaties in detail uitwerkt. Vervolgens koppel je daar criteria aan (kwaliteit) en tot slot volgt de planning (tijd) en de begroting (geld). Daarmee lijkt je vervolgens eenvoudig een beslissing te kunnen nemen of de investering in het project rendabel is. Dat lijkt zo...

Deze aanpak met gedetailleerde specificaties noemen we Big Design Up Front (BDUF). Helaas zit het venijn in die details: het helder en concreet krijgen kost veel tijd. En in de praktijk blijkt vaak dat de eisen en wensen tijdens het project alsnog wijzigen. Dit heeft veel consequenties voor tijd, budget en/of kwaliteit. Schijnzekerheid dus.

Bij tegenslagen kijk je vaak eerst naar wat je aan extra mensen in kunt zetten: effect op tijd en geld. Of stel je de kritische vraag of uitgebreid testen wel echt nodig is: effect op kwaliteit. En als er werk opnieuw moet, dan heeft dat zelfs effect op tijd, kosten én kwaliteit. Verkleinen van de scope blijkt een lastige keus: je krijgt dan minder dan wat je vooraf afsprak. Dus kies je liever voor iets meer betalen en er langer over doen.



Figuur 6: Stuurvariabelen – Klassiek & Agile

Bij AgilePM draai je het om: wat kun je in het project realiseren binnen het beschikbare budget, binnen de gestelde periode en conform het afgesproken kwaliteitsniveau? En dan blijft er maar één stuurvariabele over: scope. Hoe werkt dit?

Bij aanvang van het project spreek je een deadline en een budget af; die zijn heilig. En ook spreek je af dat je alleen conform de afgesproken kwaliteit oplevert. Pas dan breng je de eisen en wensen in kaart, maar zo globaal, dat slechts met enige zekerheid valt aan te geven dat het project haalbaar is: “Het zou moeten kunnen”. Zodra tijdens het project blijkt dat één van de drie gefixeerde variabelen (deadline, budget en kwaliteit) in gevaar komt, dan los je dat op door iets niet te doen. Het minst belangrijk item van dat moment valt domweg van het bordje.

AgilePM volgt daarmee de ‘Enough Design Up Front’ (EDUF) benadering: op hoofdlijnen de scope afstemmen, zodat je grof kunt schatten en begroten. Per timebox werk je vervolgens de details voor die periode verder uit. Dit betekent dat je eisen en wensen moet prioriteren: op hoofd niveau voor het project of per increment en op detailniveau voor een timebox, maar altijd naar noodzakelijkheid en toegevoegde waarde. Tenslotte wil je als opdrachtgever/klant zoveel mogelijk toegevoegde waarde behalen en zo min mogelijk wensen laten vallen.

Tot slot bestaat er ook nog ‘No Design Up Front’ (NDUF). Dat is hetzelfde als experimenteren zonder plan of “We zien wel waar we uitkomen”. Voor de meeste projecten is dit veel te risicovol en veel organisaties zullen moeite hebben met die onzekerheid en ongecontroleerdheid.

1.5 Het nut van AgilePM

Naar de problemen in projecten met de klassieke aanpak is veel onderzoek gedaan. We geven de top 10 van meest voorkomende problemen:

1. Communicatieproblemen
2. Te late oplevering
3. Verkeerde oplossing
4. Wijzigende scope
5. Topkwaliteit met een gouden randje
6. Technische showstoppers
7. Kostenoverschrijding
8. Vertraagde of verminderde Return on Investment (ROI)
9. Resource issues
10. Conflicterende prioriteiten

Deze opsomming komt je vast wel bekend voor. Maar wat zit er achter? En hoe gaat AgilePM met deze valkuilen om? Laten we de problemen eens punt voor punt bekijken.

1. Communicatieproblemen

Slechte communicatie is een veel genoemde oorzaak van problemen in projecten. Niet onbegrijpelijk als je je realiseert, dat de communicatie exponentieel toeneemt met het aantal stakeholders. Hoe groter het projectteam, hoe meer communicatie en afstemming nodig is en hoe groter de kans op ruis, onbegrip en misvattingen. Maar pas op: communicatie is een containerbegrip. Wanneer communicatie het probleem is, schuurt meestal een dieper liggende oorzaak. Denk aan angst om elkaar niet aan te spreken, teamleden die niet met elkaar overweg kunnen, onjuiste informatie geven of elkaar niet op de hoogte houden van veranderingen.

AgilePM stimuleert regelmatige, rijke en face-to-face communicatie in en rondom het project. De discussie die je met een klant krijgt als je iets laat zien (zoals een model, concept of prototype), levert meer op dan een document van 100 pagina's. AgilePM gaat daarnaast uit van kleine teams met een duidelijk omschreven rolverdeling. De rolverdeling helpt een nieuw team zich snel te vormen, terwijl het principe van zelfsturing bijdraagt aan de onderlinge verbondenheid, eigen initiatief en verantwoordelijkheid. De dagelijkse werkbijeenkomsten (Daily Stand-ups) en de workshops maken de communicatielijnen kort. Daarnaast zorgen ze voor laagdrempelige, interpersoonlijke interactie.

2. Te late oplevering

Veel projecten lopen uit met alle gevolgen van dien. Slechte schattingen kunnen daar de oorzaak van zijn, maar dat hoeft niet. Schatten is een vak op zich: je kijkt naar de toekomst en probeert te voorspellen hoe die eruit ziet. In alle eerlijkheid moeten we bekennen dat nagenoeg al onze projecten ook niet 100% volgens plan verlopen. Iets nieuws maken met een nieuw team in een nieuwe omgeving maakt schatten niet eenvoudig.

Beter is het om de verwachtingen meteen te managen: je kunt niet alles voorzien en wijzigingen zijn onvermijdelijk. Maar door met vaste deadlines te werken, timeboxen te gebruiken en de eisen en wensen te prioriteren, weet de klant zeker dat hij iets krijgt wat toegevoegde waarde heeft. En met iedere volgende iteratie, timebox en increment, wordt de planning stapsgewijs steeds nauwkeuriger en kun je gericht in gesprek blijven over aanpassingen en gevolgen.

[3. Verkeerde oplossing](#)

Wie heeft het niet meegemaakt: je bestelt een nieuw product en het voldoet bij gebruik toch niet helemaal aan je verwachting. Met projecten is het niet anders. Veel problemen worden pas zichtbaar bij het gebruik van de oplossing. Een feature die net iets anders werkt of juist ontbreekt. Features die je toch niet gaat gebruiken. Een kostbaar probleem, want hoe later dit aan het licht komt, hoe meer moeite het kost om het probleem op te lossen.

De directe betrokkenheid van de klant/gebruiker in het projectteam en het gebruik van modellen en prototypes, maakt dat fouten in de oplossing of verschillen van interpretatie van eisen snel op tafel komen.

[4. Wijzigende scope](#)

Dit probleem staat ook bekend als 'scope creep' en voortschrijdend inzicht. In een stabiele omgeving is duidelijkheid over wat men wil. Dan kun je eisen en wensen op voorhand goed in detail uitwerken. Maar in veel omgevingen is dat nu juist het probleem: de tijd dringt en de omgeving verandert continu. De oplossing en/of gebruikte techniek is zo complex dat een uitgebreide, gedetailleerde specificatie te lang duurt en later waarschijnlijk toch zal wijzigen.

AgilePM gaat ervan uit dat wijzigingen onvermijdelijk zijn en kiest de middenweg: formuleer wel eisen en wensen op hoofdlijnen, maar werk details pas zo dicht mogelijk bij de realisatie uit. Van een trage en stroperige wijzigingsprocedure is ook geen sprake. Het team is immers zelfsturend en empowered om alle detailbeslissingen zelf te nemen.

[5. Topkwaliteit met gouden randjes](#)

Menige oplossing of eindproduct krijgt nog de nodige franjes mee. Je wilt laten zien dat je iets oplevert dat er ook nog fantastisch uitziet. In de basis is dit geen verkeerde gedachte, maar zorgen die franjes niet voor meer kwetsbaarheid, onderhoud en doorlooptijd? En welke toegevoegde waarde leveren die franjes nu eigenlijk?

Goed is goed genoeg. Door te prioriteren op 'business value' en binnen de timebox alleen die delen te maken die benefits of waarde leveren, zorg je ervoor dat het team de focus behoudt en niet onnodig tijd verspilt aan nutteloze extraatjes.

[6. Technische showstoppers](#)

Er is tegenwoordig nauwelijks meer een project te bedenken zonder een ICT- of technische component. Niet zelden zijn die projecten nieuw en complex. De kans is best groot dat zich vroeg of laat - meestal laat - een technisch onoverkomelijk probleem voordoet.

Dat je bij projecten met een hoge 'nieuw' factor tegen dit soort problemen aanloopt, is onvermijdelijk. Dat hoort nu eenmaal bij innoveren. Met AgilePM zorg je er wel voor dat dit zo vroeg mogelijk aan het licht komt. Zogenaamde 'proof of concepts' of prototypes toepassen in combinatie met testen parallel aan ontwikkeling, brengt het principe 'fail fast' in de praktijk. Daardoor kun je vroegtijdig signaleren en zo mogelijk aanpassen (of stoppen).

[7. Kostenoverschrijding](#)

Bij de watervalmethode is de kans op kostenoverschrijding vrij groot. Fouten die je laat in het traject ontdekt, zijn immers moeilijk op te lossen. En daar betaal je een prijs voor. Naast correctiewerk komt daar vaak nog de tijdsvertraging bij en ook daar zijn kosten aan verbonden.

Door binnen AgilePM timeboxing te gebruiken, fixeer je niet alleen de tijd, maar ook het budget. Kostenoverschrijding is niet mogelijk. Maar ook Agile projecten kennen toch tegenslagen en dus vertragingen en hogere kosten? Tegenslagen wel, alleen los je die niet op door het budget los te laten, maar door één of meerdere lager geprioriteerde eisen en wensen niet te realiseren. Bovendien deel je een project op in meerdere, kleine timeboxen. Tegenvallers worden daardoor heel snel zichtbaar, waardoor verwachtingen en werkelijkheid nooit ver uiteen liggen.

8. Vertraagde of verminderde Return on Investment (ROI)

Over het algemeen geldt dat je pas toegevoegde waarde realiseert, nadat je het eindproduct in gebruik neemt. Bij de klassieke benadering is pas helemaal aan het einde van het project sprake van een gereed product; pas dan is het eindproduct 100% klaar. Een vertraging en/of kostenstijging betekent onmiddellijk dat terugverdienen later begint en langer gaat duren.

AgilePM ondervangt dit op twee manieren. Enerzijds realiseer je eisen en wensen met een hoge toegevoegde waarde zo vroeg mogelijk in het project. Anderzijds streef je ernaar ook tussentijds ‘gereed deelproduct’ op te leveren, zodat de organisatie alvast kan profiteren van de bijbehorende toegevoegde waarde.

9. Resource issues

Je kent het probleem vast wel: je komt mensen te kort, ze zijn te onervaren of door vertraging in het project niet meer beschikbaar (want al ingepland voor een ander project). Tel daarbij op dat je in de klassieke benadering werk functioneel opdeelt (bijvoorbeeld specificeren, bouwen, testen, implementeren) en achter elkaar plaatst. De kans is groot dat je niet alles van tevoren goed overziet en dat aspecten tussen wal en schip raken.

Maar eerlijk is eerlijk, AgilePM zorgt ook niet voor meer ervaring of een grotere, beschikbare capaciteit. Wat AgilePM wel doet is, timeboxing toepassen. En dat betekent dat je geen deadlines overschrijdt en betrokkenen niet langer aan het project werken dan afgesproken. Daarmee kun je veel stabiel en voorspelbaarder resources aan projecten toekennen.

Verder gaat AgilePM uit van kleine, stabiele, multidisciplinaire, zelfsturende teams. Beknopte rolbeschrijvingen wijzen teamleden bij aanvang voldoende snel hun weg en voorkomen dat zaken onafgedekt blijven. Tegelijkertijd creëert AgilePM met dergelijke teams een omgeving waarin een klein team zich kan ontplooiën, talenten kan benutten en waarderen en waarin resultaten zichtbaar zijn. Ervaring groeit door samen te werken, elkaar te stimuleren en te ondersteunen, zeker als een team langer samenwerkt aan één of meer projecten.

10. Conflicterende prioriteiten

Een terugkerende ergernis van veel projectmanagers is de strijd om schaarse resources. Heb je net afspraken gemaakt over de inzet van een specialist, blijkt hij toch niet beschikbaar. “Sorry, druk druk druk”, “Ander werk liep wat uit” of “Er kwam wat tussendoor”. Herkenbaar?

De meest ideale oplossing is teamleden fulltime inzetten. Dit heeft ook de voorkeur van AgilePM. Helaas is dit lang niet altijd haalbaar, dus dan maar een vaste inzet; bijvoorbeeld drie dagen per week. Daarnaast is de strategische aansluiting belangrijk. Als voor teamleden en stakeholders op de verschillende niveaus in de projectorganisatie helder is welke belangen prioriteit hebben, kan het team snel en adequaat reageren op belangenconflicten. Workshops kunnen een hulpmiddel zijn om dergelijke problemen te bespreken en beheersbaar te maken.

1.6 Voordelen van AgilePM

Laten we de voordelen van AgilePM nog eens samenvatten:

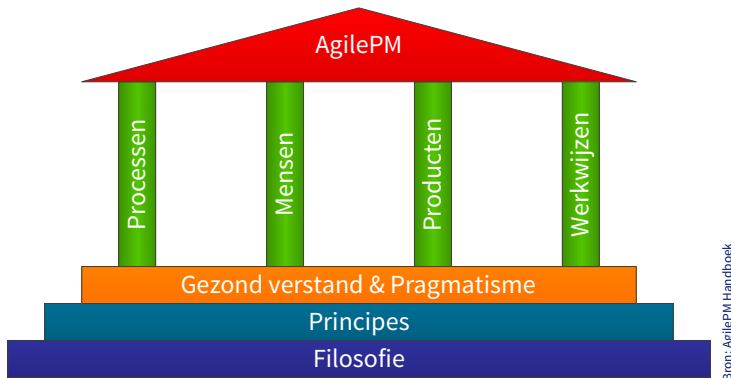
- Gebruikers voelen zich eigenaar van de oplossing en omarmen deze, doordat ze nauw betrokken zijn bij het project.
- Er is een drastisch grotere kans dat de oplossing is wat ook andere stakeholders verwachten, doordat ook zij nauw betrokken zijn, door korte, cyclische opleveringen en directe communicatielijnen.
- Uitrollen en in gebruik nemen verloopt hierdoor veel soepeler.
- Mede door goed te prioriteren en daadwerkelijk binnen tijd en budget te leveren, zal de uiteindelijke oplossing met meer zekerheid voldoen aan de echte behoefte van de business: toegevoegde waarde genereren.
- Teamleden kunnen collega-gebruikers beter informeren en praktijkgericht trainen om de oplossing te gebruiken.

AgilePM lost dus verschillende, veel voorkomende problemen met projecten op. Maar daarnaast levert het voorspelbaarder kwalitatieve oplossingen op tijd en binnen budget. Dat zal voor vele organisaties een diepgewortelde wens zijn. Zolang alle betrokken zich blijven realiseren dat projectwerk nog steeds mensenwerk is, valt er veel te winnen.

2. Over AgilePM

2.1 Structuur AgilePM

De structuur van AgilePM kun je je het beste voorstellen als een Romeinse tempel met fundering, vloer, dekvloer, kolommen en dak.



Figuur 7: De structuur van AgilePM

De filosofie bestaat uit vijf uitgangspunten waarin het Agile-gedachtegoed ligt besloten: de fundering voor ieder Agile-project. Daarop ligt de vloer: acht principes die de basis vormen voor het succes van het project. De dekvloer zorgt voor agility (wendbaarheid) in de toepassing, zodat de kolommen licht blijven. Vervolgens staan op de vloer vier dragende kolommen: de processen, de mensen, de producten, de werkwijzen. Werkwijzen worden in AgilePM ook wel techniek genoemd. En tot slot het dak: AgilePM is succesvol toegepast op projecten.

Overzicht

2.2 Filosofie

Wie het Agile Manifest ernaast legt, zal zien dat de filosofie van AgilePM daar veel overeenkomsten mee vertoont. Op detailniveau zijn er nuanceverschillen, maar het gedachtegoed van Agile is duidelijk herkenbaar.

fi·lo·so·fie (de; znw.)

1. denkwijze, benadering.

AgilePM filosofie:

De beste zakelijke waarde ontstaat, wanneer projecten zijn afgestemd op duidelijke zakelijke doelstellingen, met regelmaat tastbaar resultaat opleveren en draaien om samenwerking tussen gemotiveerde mensen met voldoende bevoegdheid en autonomie.

Dit is bereikbaar, als alle stakeholders:

1. de bedrijfsdoelstellingen kennen, begrijpen en onderschrijven;
2. 'empowered' zijn om binnen grenzen zelfstandig te opereren en beslissingen te nemen;
3. nauw samenwerken om een juiste en passende oplossing te leveren;
4. nauw samenwerken om de oplossing op tijd op te leveren en volgens de prioriteiten zoals de business die bepaalde;
5. wijzigingen accepteren als onvermijdelijk, omdat hun inzichten zich gaandeweg zullen verdiepen.

2.3 Principes

De filosofie is vertaald in de vloer met haar acht principes. Deze principes zijn leidend voor het gedrag, de houding en de mindset van je team, wil het consequent leveren en presteren. Wanneer het geheel of gedeeltelijk niet wil of kan voldoen aan deze principes, verzwakt de filosofie en verkleint je kans op succes.



1. Focus op de behoeften van de business



2. Lever op tijd



3. Werk samen



4. Doe geen concessies aan kwaliteit



5. Bouw incrementeel vanaf een stevige basis



6. Ontwikkel iteratief



7. Communiceer duidelijk en continu



8. Laat zichtbare beheersing zien

Overzicht

Klinkt goed en logisch, toch? Maar wat houdt ieder principe nu precies in en welke toegevoegde waarde heeft het voor je project? Laten we de principes eens stap voor stap doornemen.



1. Focus op de behoeften van de business

Om te kunnen focussen op de bedrijfsbehoefte, moet je eerst weten wat die is. Zorg er daarom voor dat je team de werkelijke bedrijfsprioriteiten begrijpt. Vraag bij aanvang van het project aan de business wat de bedrijfsprioriteiten zijn en gebruik ze bij het prioriteren van de eisen en wensen. Herhaal de vraag als je een deelproduct oplevert en op het punt staat te beginnen met de volgende timebox. Dan kun je met de prioriteiten rekening houden in je Realisatieplan en opeenvolgende Timeboxplannen.

De bedrijfsbehoefte moet overigens ook blijken uit de Businesscase, waarin de verwachte benefits van het project staan - liefst ook in financiële termen. Zoek hiervoor sponsorschap en commitment vanuit de business.

Zoals we hebben gezien, is de scope variabel bij AgilePM. Maar dat betekent niet dat je de klant in onzekerheid moet laten zitten. Garandeer wat je minimaal in de oplossing zult meenemen: de minimaal vereiste subset (Engels: Minimum Usable Subset). Meer hierover in het hoofdstuk over MoSCoW-prioritering.

= aantoonbare toegevoegde waarde leveren



2. Lever op tijd

Op tijd leveren is een belangrijke afspraak om vertrouwen te kweken: de deadline is heilig, doe wat je zegt te doen. En dat vereist (zelf)discipline.

Dat bereik je, door timeboxing toe te passen op al het werk. Of het nu gaat om een vergadering, een workshop, een demonstratie of een release: timeboxing helpt je scherp te blijven en je aandacht te richten op wat nu nodig is. De prioriteiten van de business zijn daarbij leidend.

Lever ook regelmatig iets tastbaars of bruikbaar op, wat echt toegevoegde waarde heeft. Gebruikers zien dan al snel concrete resultaten en dat haalt de zorg om een eventuele vertraging weg.

= vertrouwen verdienen



3. Werk samen

Werken in projecten betekent samenwerken: niet alleen met elkaar in dezelfde ruimte werken aan hetzelfde, maar vooral elkaar actief stimuleren en helpen. Dat betekent dat je begrip hebt voor en respect toont aan je teamleden, dat je initiatief en verantwoordelijkheid neemt en een ander aanspreekt op zijn verantwoordelijkheid.

Een projectteam is multidisciplinair: teamleden komen zowel bij de klant als de leverancier vandaan en hebben ieder hun eigen specialisme en persoonlijkheid. Daarbij is altijd sprake van aanvulling: leden zijn complementair. Ieder lid voegt iets extra's toe aan het team, maar heeft tegelijkertijd ook overeenkomsten met een ander lid.

Een team dat goed samenwerkt, is meer dan de som der delen. Samenwerking vergroot het begrip, de snelheid en de wendbaarheid. Dit stimuleer je door eenheid te brengen in het team en de cultuur en daarnaast actieve betrokkenheid van stakeholders te verlangen in en rond het team.

= bouwen van een excellerend team dat de klus klaart



4. Doe geen concessies aan kwaliteit

Wat kwaliteit is, spreek je af aan het begin van het project: wat betekent 'goed genoeg' en wanneer is iets af? Vervolgens houd je je ook aan die acceptatiecriteria. Zo voorkom je dat kwaliteit variabel wordt. Doe liever een paar dingen goed, dan veel dingen half.

Test en controleer alles grondig, vroeg en continue. Wacht dus niet met controleren tot iets af is, maar begin direct bij de bouw. Kijk waar je kwaliteitscontroles kunt inbouwen in het product en voer (her)controles doorlopend uit.

Kwaliteit betekent ook dat je kiest voor voldoende ontwerp, documentatie en testen. Niet te veel en niet te weinig.

= een oplossing leveren die 'goed genoeg' is



5. Bouw incrementeel vanaf een stevige basis

Ieder project heeft een degelijke start nodig: een gezamenlijk vertrekpunt en een ingericht project als basis. Een duidelijk beeld van het probleem en de oplossing is voldoende, zonder te verzuipen in verlamme details.

Vervolgens bouw je incrementeel en lever je een oplossing op die telkens een stukje uitgebreider en beter is: een increment [Woorddefinitie: increment]. De klant neemt dit in gebruik en levert je nuttige terugkoppeling uit de praktijk. Die ervaringen verwerk je in het volgende increment, waardoor je eerder benefits realiseert en problemen eerder signaleert en oplost. Klant blij, leverancier blij.

Zorg daarnaast voor regelmatige (her)beoordeling en bevestiging van de projectrichting en de geldende prioriteiten. Niet alleen weet je zo zeker dat je nog steeds aan de juiste oplossing werkt, maar je weet ook dat het project gerechtvaardigd is.

**richting bepalen/vasthouden en snel benefits realiseren
= vertrouwen kweken**



6. Ontwikkel iteratief

Iteratief ontwikkelen zit in de genen van Agile: je begint globaal en al doende en lerende werk je cyclisch en convergerend [Woorddefinitie: convergeren] toe naar de uiteindelijke oplossing. Aan het begin van iedere iteratie doe je voldoende aan ontwerp om een goede start met de bouw te maken.

Accepteer dat details pas later duidelijk worden en dat wijzigingen onvermijdelijk zijn. Zonder verandering zal je oplossing zich ook niet verder ontwikkelen. Ontwikkel alle producten iteratief en houd de cycli kort. Dan profiteer je van snelle terugkoppeling en ontdek je fouten eerder. Wees creatief, experimenteer, ervaar en leer.

= ontdekken en verbeteren



7. Communiceer duidelijk en continue

Goede communicatie is essentieel om ruis en misverstanden te voorkomen. Zorg daarom zoveel mogelijk voor informele face-to-face communicatie op alle niveaus. Gebruik liever korte, energieke workshops dan lange vergaderingen.

Houd dagelijks een korte werkbespreking. Problemen en knelpunten signaleer je zo heel snel en kan het team direct oppakken. Probeer waar mogelijk gebruik te maken van 'rijke' technieken zoals modellen, prototypes en schetsen. Zo kun je delen van de oplossing gemakkelijker demonstreren en maak je het visueel en tastbaar voor de gebruiker. Bovendien zeggen plaatjes meer dan duizend woorden. En is documentatie nodig? Zorg dan dat die op orde is, maar kort en bondig.

= iedereen geïnformeerd en betrokken houden



8. Laat zichtbare beheersing zien

Vertrouwen is goed, maar meten is weten. Omdat je iteratief werkt met een beperkt ontwerp als vertrekpunt, bestaat bij stakeholders vaak wel enige zorg. Vertrouwen moet je winnen en dat doe je door planning en voortgang zichtbaar te maken voor iedereen. Je stakeholders mogen bijvoorbeeld rustig je dagelijkse Stand-up bijwonen. Dat is transparantie en daarvoor is lef nodig. Pas de formaliteit van voortgangsbewaking en -rapportage wel aan op wat echt nodig is.

Bewaak de voortgang van het werk op het niveau van opgeleverde producten in plaats van op het niveau van afgeronde activiteiten. Je klant kan zich daar veel gemakkelijker een beeld bij vormen. Naast de voortgang van het werk, houd je ook de voortgang en de ontwikkeling van de Businesscase bij. Zo evalueer je de doorlopende projectrechtvaardiging aan de hand van bedrijfsdoelen.

Je managet de omgeving en omstandigheden proactief en stimuleert de zelfsturing binnen het team.

= voortgang zichtbaar maken en vertrouwen tonen

2.4 Gezond verstand & Pragmatisme

De basis ligt er nu. Maar om de vier pijlers niet te zwaar en te log te maken, dien je deze met gezond verstand praktisch passend te maken voor het project. Het 'doen' heeft de voorkeur boven 'eerst alles tot in detail doordenken en uitwerken'. Dit ethos zorgt ervoor dat mensen en hun onderlinge interactie voorrang blijven krijgen boven processen en tools. Bureaucratie en te veel overhead sluit je zo zoveel mogelijk uit.

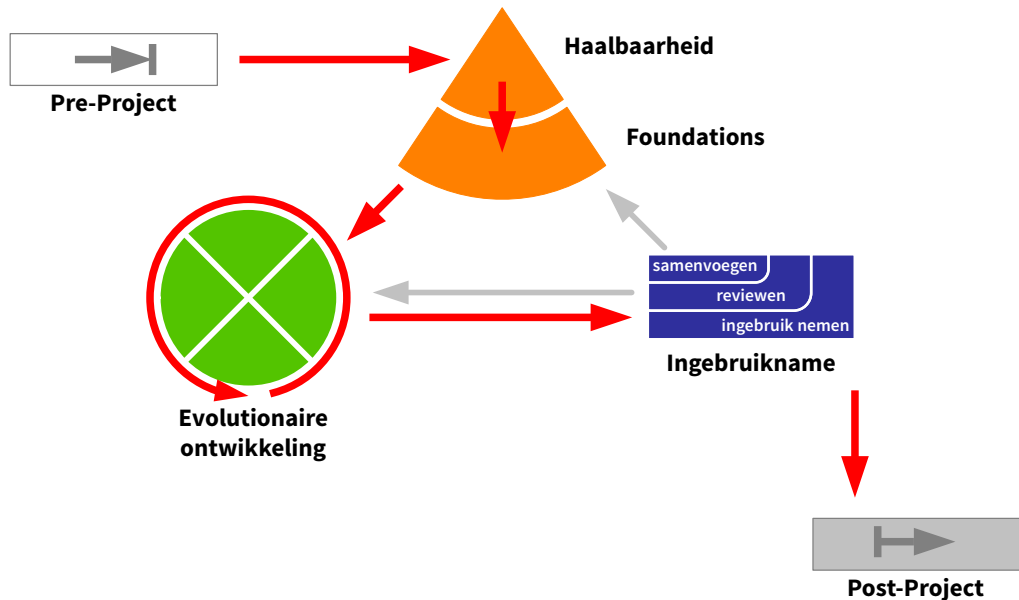
2.5 Processen

In tegenstelling tot andere Agile benaderingen bestrijkt AgilePM de hele projectlevenscyclus en niet alleen de bouw of ontwikkeling. Deze levenscyclus is iteratief en incrementeel, zodat je de oplossing in delen oplevert en stap voor stap groeit. AgilePM onderkent daarin ook de noodzaak voor projectmanagement en integreert de verschillende disciplines die daarbij horen.

Ook zorgt de methode ervoor dat de toegevoegde waarde van het project duidelijk en de voorgestelde oplossing haalbaar is. En het zorgt voor een degelijke basis vóór de start van het inhoudelijke werk. Daarbij pas je de techniek timeboxing veelvuldig toe.

Processen heten binnen AgilePM ook wel fases. Overigens niet te verwarren met de fases in de watervalmethode, waarbij je de volgende fase niet mag beginnen voordat je de voorgaande hebt afgerond en goedgekeurd. Bij AgilePM is het mogelijk sommige fases samen te voegen en in de tijd tegelijk en zelfs door dezelfde teamleden te laten uitvoeren. Het is maar hoe je het inricht voor je specifieke project. Projecten blijven maatwerk.

De zes processen of fases zijn:



Bron: AgilePM Handboek

Figuur 8: Het procesmodel van AgilePM

Overzicht



[Pre-Project \(Pr\)](#)

De fase waarin het project ontstaat en je afspreekt wat het vertrekpunt is voor het project. Deze fase valt buiten het project.



[Haalbaarheid \(Ha\)](#)

Over het algemeen een korte periode waarin je de haalbaarheid en de rechtvaardiging onderzoekt en vastlegt.



[Foundations \(Fo\)](#)

In deze belangrijke fase zoek je het project uit en richt je het in. Er moet voldoende gezamenlijk inzicht zijn om op hoofdlijnen de scope, de technische elementen, de standaarden en procedures af te spreken, voordat je begint met bouwen en ontwikkelen.



[Evolutionaire Ontwikkeling \(Eo\)](#)

Een iteratieve bouw- en ontwikkel fase, waarin je de oplossing stap voor stap laat groeien door kerntechnieken als Iteratief Ontwikkelen, Prioritering en Timeboxing toe te passen.



[Ingebruikname \(In\)](#)

De fase van het project waarin je ieder increment (de output van één of meerdere timeboxen) oplevert voor operationeel gebruik. De (deel)oplossing gaat dus live.



[Post-Project \(Po\)](#)

De periode waarin je controleert in welke mate je de benefits daadwerkelijk realiseerde. Deze fase valt buiten het project.

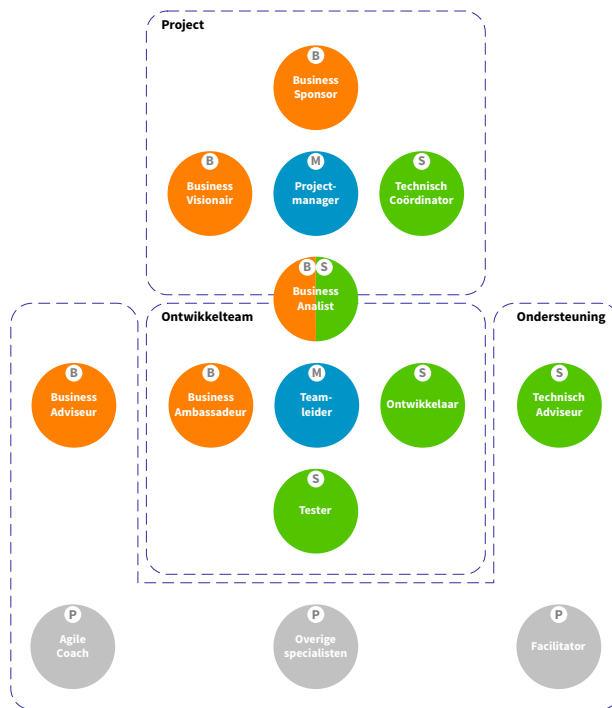
2.6 Mensen

AgilePM doet een voorstel voor een generieke structuur van de projectorganisatie en de rollen die daarin aanwezig zijn. Daarbij gaat AgilePM uit van een project van enige omvang. Laat je dus niet afschrikken door het aantal rollen of de naamgeving. AgilePM gaat uit van rollen, ingedeeld naar competentie/bijdrage aan het project. Teamleden kunnen meerdere rollen vervullen of samen met een ander lid dezelfde rol vervullen. Sommige sectoren gebruiken andere namen voor vergelijkbare taakinhoud. De kern is dat je geen taken en verantwoordelijkheden over het hoofd ziet, taken in je project evenwichtig verdeelt en betrokkenheid en input waarborgt.

Figuur 9 toont de rollen ingedeeld per focusgebied en hun globale onderlinge positie in het project.

Het schema gebruikt kleuren om de interessegebieden duidelijk te maken. De oranje rollen hebben een business/gebruikers focus (**B**), de groene rollen een specialistische/technische focus (**S**), de blauwe rollen een (project)management focus (**M**) en de grijze rollen een procesondersteunende focus (**P**).

De projectorganisatie kent drie categorieën of focusgebieden: Project, Ontwikkelteam(s) en Ondersteuning. Het projectniveau is een managersniveau. Deze rollen worden vervuld door mensen die rapporteren aan een stuurgroep of daar lid van zijn.



Bron: AgilePM Handboek

Figuur 9: Rollen in een AgilePM-project

Hieronder volgen de belangrijkste verantwoordelijkheden van iedere rol. In het hoofdstuk over Projectorganisatie gaan we daar dieper op in. **Let op:** voor enkele rollen kiezen wij een andere vertaling, omdat we de letterlijke vertaling te sector specifiek of te verwarrend vinden. De officiële termen zoals die ook in de examens staan, zie je in het rood eronder vermeld.

Project

Rol	Kern verantwoordelijkheden
Business Sponsor	Project champion en eigenaar van de Businesscase: stelt budget en resources veilig, garandeert effectieve besluitvorming en handelt escalaties snel af.
Projectmanager	Zorgt voor planning op hoofdlijnen: bewaakt de voortgang, configuratiebeheer en de beschikbaarheid van resources. Managet risico's, escaleert knelpunten en focust op zorgen voor een goede werkomgeving.
Business Visionair	Eigenaar van de visie van de business en ziet toe op de impact van de bredere business verandering: bewaakt de voortgang ten opzichte van de visie. Is daarmee medebepaler van de requirements op hoofdlijnen en geeft input aan het ontwerp en tijdens reviewsessies. Eigenaar van de in gebruik genomen oplossing.
Technisch Coördinator	Accordeert en bewaakt de technische architectuur: identificeert en managet technische risico's, adviseert over en coördineert de technische teamactiviteiten. Ziet er daarnaast op toe dat de oplossing aan niet-functionele eisen voldoet.
Business Analyst	Rechterhand van Projectmanager en brengt de business en technisch inhoudelijke aspecten bij elkaar. Verbindingsrol tussen project en ontwikkelteam.

Overzicht

Ontwikkelteam(s)

Rol	Kern verantwoordelijkheden
Business Analyst	Ziet erop toe dat gevolgen van dagelijkse beslissingen op de business worden overzien en overwogen: ondersteunt daartoe de communicatie tussen de business en de technische rollen van het team. Managet alle benodigde producten gerelateerd aan de eisen.
Teamleider	Zorgt dat het team gefocust blijft op tijdige oplevering: managet de details van een timebox en de dagelijkse activiteiten. Ziet er tevens op toe dat test- en reviewactiviteiten worden gepland en uitgevoerd. Stimuleert actieve bijdrage van het hele team.
Business Ambassadeur	Brengt het gezichtspunt van de business in voor de dagelijkse besluitvorming: beschrijft business scenario's om de oplossing te ontwerpen en te testen. Draagt bij aan alle eisen, ontwerp- en reviewsessies en borgt dat de oplossing juist is. Coördineert acceptatie door de business.
Ontwikkelaar	Ontwikkelt de oplossing en participeert volledig in alle relevante activiteiten voor kwaliteitsborging.
Tester	Werkt samen met de business rollen aan het definiëren van testscenario's. Voert de volledige technische testen uit en rapporteert resultaten aan de Teamleider en de Technisch Coördinator.

Rol	Kern verantwoordelijkheden
Business Adviseur	Geeft specifieke input of adviseert het team over een specifiek business gerelateerd onderwerp (bijvoorbeeld een accountant of juridisch adviseur).
Technisch Adviseur	Geeft input en adviseert het team over specifieke technische aspecten.
Overige Specialisten (Geen officiële rol meer)	Specialisten die kort, veelal op ad hoc basis uitvoerend (ondersteunend) werk verrichten. Inzet op projectniveau is ook mogelijk, maar het werk heeft dan een meer adviserend, verifiërend karakter.
Agile Coach (DSDM Coach)	Helpt teams voor wie AgilePM nieuw is, om het maximale uit de methode te halen. Stemt AgilePM af op de behoefte van het project.
Facilitator (Workshop Facilitator)	Onafhankelijke manager en organisator van workshops: verantwoordelijk voor de context en het proces rondom een workshop, niet voor de inhoud.

2.7 Werkwijzen

Veel projecten hebben last van overheadkosten. Wanneer je de vijf technieken van AgilePM goed toepast, schuilt er een enorm kracht in: je kunt die overheadkosten minimaliseren en tegelijkertijd op tijd leveren en snel toegevoegde waarde realiseren. Wie wil dat nu niet?

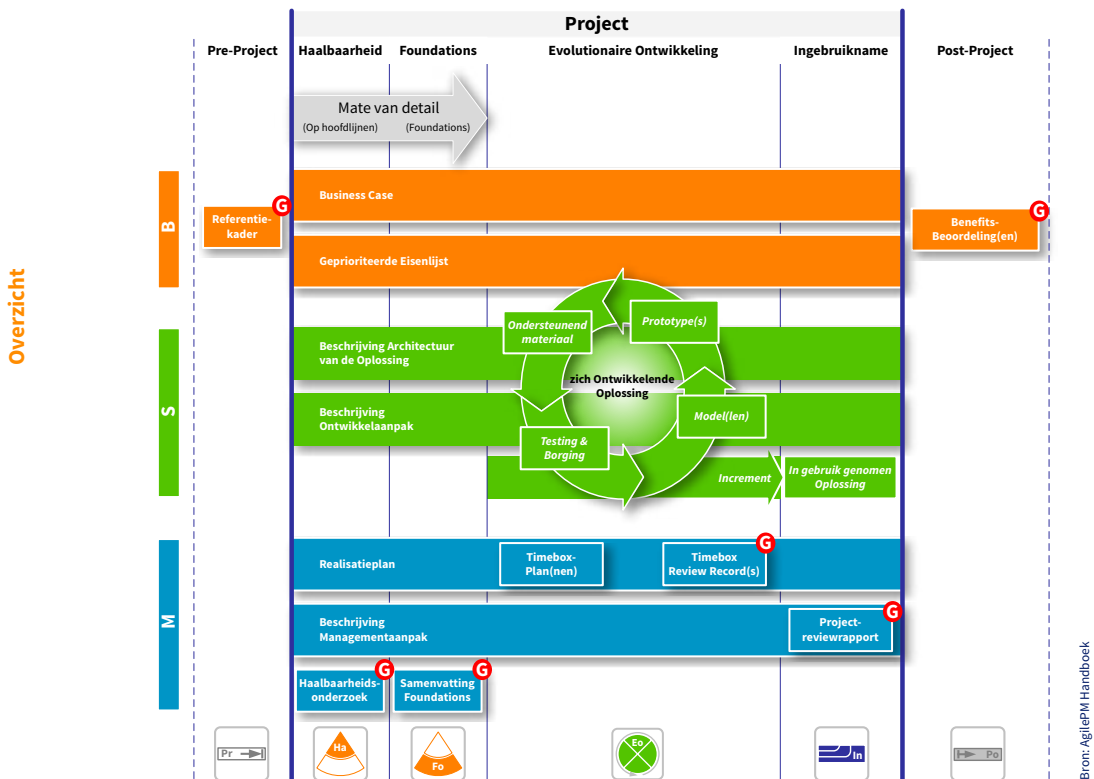
Timeboxing	Het opdelen van het werk in korte tijdsperioden. Doel: met enkele korte iteraties zoveel mogelijk werk binnen de gestelde tijd en het vastgestelde budget afkrijgen - t enminste de minimaal vereiste subset. De techniek zorgt ervoor dat je altijd op tijd levert.
Iteratief Ontwikkelen	Door de stapsgewijze verfijning en detaillering laat je een oplossing evolueren van idee naar een opgeleverd en werkend product.
Prioriteren	Eisen op project-, increment en timebox-niveau prioriteren en de planning van timeboxen toetsen op haalbaarheid. AgilePM gebruikt hiervoor de MoSCoW-regel: eisen met een hoge prioriteit ('Must haves') krijgen voorrang op wensen ('Should haves' en 'Could haves'). Deze aanpak helpt je bij het zo snel mogelijk realiseren van toegevoegde waarde.
Workshops	Workshops kunnen veel bijdragen aan het groepsproces, zeker als een onafhankelijke facilitator die leidt. Denk bijvoorbeeld aan sessies voor besluitvorming, probleemoplossing en brainstorming.
Modelleren	Modelleren is een techniek om eisen boven tafel te krijgen, verwachtingen te managen, haalbaarheid te toetsen en resultaten te demonstreren. Je kunt denken aan schaalmodellen, voorbeeldschermen, proof-of-concept en prototypes om resultaten te visualiseren en tastbaar te maken.

2.8 Producten

Iedere fase in de levenscyclus kent één of meerdere deliverables of producten. Niet alle producten zijn nodig voor ieder project en ook de mate van formaliteit en vastlegging varieert. Dit is afhankelijk van project- en organisatiefactoren, zoals contractuele relaties, bedrijfsstandaarden, wet- en regelgeving en bedrijfscultuur.

Sommige producten zijn specifiek voor een bepaalde fase, andere kunnen zich blijven ontwikkelen in opvolgende fases. Figuur 10 toont de standaard flow van de producten door de levenscyclus van een project. Net zoals bij de rollen van de projectorganisatie zijn de producten specifiek voor een bepaalde doelgroep. Met de **B/oranje** voor business-/ gebruikersproducten, **M/turquoise** voor managementproducten en **S/specialistisch** voor specialistische producten.

De producten gemarkeerd met een **G** zijn governance producten: bedoeld voor besluitvorming of controle. Deze producten hebben over het algemeen een formele status.



Figuur 10: Belangrijke producten in de projectlevenscyclus

AgilePM staat toe dat teamleden zelf beslissen hoe ze producten maken en wat de inhoud is. Aanpassing aan omgeving en omstandigheden is zo bijna altijd mogelijk en vaak ook noodzakelijk. Soms zijn nagenoeg alle producten nodig, soms volstaat het te werken met slechts een eisenlijst (AgilePM: Geprioriteerde Eisenlijst) en de oplossing (AgilePM: zich Ontwikkelende Oplossing). Zo'n minimalistische invulling is goed vergelijkbaar met Scrum.

Voor nu gaan we ervan uit dat dit plaatje voldoende is voor een eerste beeldvorming over de producten. Het hoofdstuk over Informatie & Documentatie gaat hier dieper op in.

Aan de hand van een concreet voorbeeld laten we zien hoe je met AgilePM een project in grote lijnen indeelt.

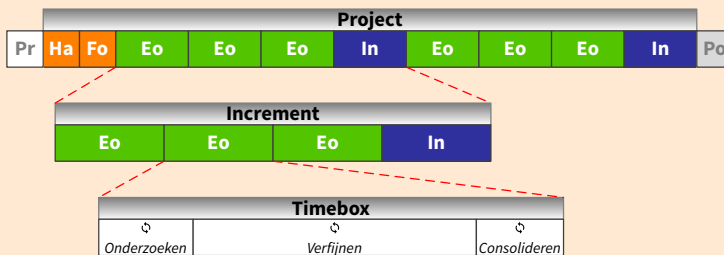


Praktijkvoorbeeld - Nieuwe Website Sportvereniging

De voorzitter van de hardloopervereniging 'SnelSnellerSnelst' wil graag een nieuwe website. De oude is niet meer van deze tijd. En na een aantal crashes is informatie verloren gegaan en is onderhoud zeer lastig. Recentelijk heeft de bond waarbij de vereniging is aangesloten, ook een webkoppeling vrijgegeven waardoor het inschrijven voor officiële wedstrijden veel eenvoudiger wordt. De wedstrijdsecretaris is dolgelukkig. Dit scheelt hem handenvol typewerk. Door een gecertificeerde applicatie te gebruiken voor het beheer en onderhoud van de website, is het ook mogelijk de ledenadministratie en -facturatie verder te automatiseren: een grote wens van de nieuwe penningmeester.

Alles in één keer bouwen en opleveren vinden de betrokkenen geen goed idee. Liever stap voor stap opleveren, zodat ze beloftes aan de leden nakomen, dan weer de mooiste en beste site bouwen die nooit echt af komt. Ze delen het project op in twee increments en gebruiken gestructureerde timeboxen. Let op: dit is een voorbeeld. Iemand anders had misschien voor een andere indeling gekozen.

- Basis website met informatie (Datum gereed: 1/9)
 - Nieuwe look & feel
 - Invoering nieuw Content Management Systeem (CMS)
 - Centrale ledenadministratie
- Backend voor leden (Datum gereed: 1/10)
 - Nieuwe look & feel
 - Beheer eigen gegevens, plaatsen nieuws, berichten, e.d.



Figuur 11: De relatie tussen project, increment en timebox

Increment 1 bestaat uit 4 timeboxen: **3x** ontwikkeling, waarvan de eerste twee voor verkenning en ontwerp (CMS leren kennen en uitproberen) en de derde voor de bouw, en **1x** ingebruikname. Daarmee is de basis website live en centrale ledenadministratie ingericht.

Increment 2 bestaat ook uit 4 timeboxen: **3x** ontwikkeling, waarvan de eerste voor ontwerp (we zijn nu bekend met het CMS), tweemaal bouwen inclusief datamigratie/conversie, en **1x** ingebruikname. Daarmee is de backend live.

Kort samengevat:

- ieder project bestaat uit 1 of meer increments met een (deel)oplossing aan het eind;
- iedere increment bestaat uit 1 of meerdere timeboxen;
- iedere timebox kent minimaal 3 iteraties: onderzoeken, verfijnen en consolideren.

2.9 Ervaringsdeskundigen

Tot zover een eerste indruk van de theorie van AgilePM. Hoe je AgilePM vervolgens afstemt op je project en projectomgeving, willen we je graag laten vertellen door ervaringsdeskundigen. Zij zijn werkzaam in verschillende sectoren en omgevingen en willen graag hun ervaringen met agile in projecten met je delen. Dat varieert van do's en don'ts tot tips en trucks. Tenslotte gaat theorie pas leven als je ermee aan de slag gaat. Verder in het boek komen zij persoonlijk aan het woord, maar hier stellen we graag hun organisaties vast aan je voor.

Fortes Solutions BV verkort doorlooptijden van projecten en het verminderen kosten en risico's in projectorganisaties. Daar zijn we specialisten in. Hiervoor ontwikkelen wij gebruikersvriendelijke webbased applicaties, met als vlaggeschip 'Principle Toolbox': de tool voor het professioneel managen van projecten, programma's en portfolio's.



Onze software is gebaseerd op concepten die wij ontwikkelden in nauwe samenwerking met gebruikers en partners. We gebruiken Agile om onze eigen producten continu te verbeteren en meer waarde voor onze klanten toe te voegen.

Ortec Finance is specialist in het meten en managen van risico en rendement. Wij leveren wereldwijd technologie en advies op het gebied van risk en return management. Onze softwaremodellen combinerend met adviesdiensten, helpen we investeerders financiële beslissingen te nemen en de resultaten daarvan te monitoren.



Onze Software Development afdeling gebruikt sinds enkele jaren Agile, gebaseerd op de Scrum-methodiek. Slechts op enkele punten volgen we onze eigen variant. Sinds we dit doen, kunnen we de updates op onze software frequenter en met een verbeterde kwaliteit leveren.

Nobel: een full service ICT-partner met een bijzondere visie. Wij zijn ervan overtuigd dat organisaties die willen blijven groeien en excelleren, moeten inspelen op de snel veranderende markt.



Daarbij zien wij ICT als bepalende factor om deze veranderingen te volgen en noodzakelijke aanpassingen mogelijk te maken.

Ons belangrijkste aanbod aan onze klanten is daarom: het vermogen om te veranderen door samenwerking in ICT. Agile Project Management versterkt in onze projecten die samenwerking en vergroot de toegevoegde waarde voor onze klanten.

WE ARE - WHAT'S NEXT. We nemen onze klanten mee op een reis vol digitale ontdekkingen. Van 'getting the basics right' tot het uitrollen van grote en innovatieve, digitale strategieën.



Het maakt ons niet uit waar de reis begint, zolang we maar het gedeelde uitgangspunt hebben: de overtuiging dat de mix van creativiteit, technologie en media, merken speciaal, deelbaar en waardevoller kan maken. Betrokkenheid van onze klanten staat bij ons hoog in het vaandel. We doen het namelijk nooit alleen; samen gaan we op zoek naar What's Next. De agile-aanpak is voor ons dan ook heel gewoon.

Technologie om mensen te helpen leren. Wij geloven dat er maar een manier is om mensen te helpen te leren: met gepersonaliseerd leren!



Met aNewSpring bieden wij een adaptief leerplatform waarop je eenvoudig cursussen kunt creëren en leveren die zich aanpassen aan de individuele leerbehoefte van de deelnemer. Leren gaat zo gemakkelijker, sneller en beter.

- Wij bieden een adaptief leerplatform met maandelijks nieuwe functionaliteiten
- Innovatie doen we samen met onze klanten voor maximale toegevoegde waarde

We gebruiken delen van Scrum om op een agile-wijze nieuwe releases te bouwen en in productie te nemen.

Klein. Het nieuwe groot. We leven in een geweldige tijd. Een tijd waarin iedereen bij kan dragen aan een andere economie. Groener. Menselijker. Innovatiever.



Wat je elke dag doet, maakt verschil. Wat je wel of niet koopt. Hoe je werkt. Hoe je reist. Al die handelingen samen zorgen voor de grote, de echte verandering. Zo maken we van klein: het nieuwe groot.

Triodos Bank is een van de duurzaamste banken ter wereld. De Financial Times riep ons in 2009 uit tot Sustainable Bank of the Year. Onze missie is om uw geld te laten werken aan positieve maatschappelijke, ecologische en culturele veranderingen.

Onze afdeling ICT gebruikt Agile / Scrum voor nieuwe ontwikkelingen.

De basis voor de activiteiten van NS Reizigers: Alles wat met het rijden van treinen en service-verlening naar de klant te maken heeft.



NS Reizigers streeft naar:

- treinen op tijd laten rijden
- voldoende en comfortabel materieel
- goede service en informatie
- adequate opvang bij verstoringen

De klant wil kwaliteit en daarom werken wij de komende jaren aan alle verbeteringen die voor 'de reis van de klant' nodig zijn. Op een gemiddelde werkdag maken meer dan één miljoen klanten per dag gebruik van onze diensten en reizen met de trein.

Sinds enkele jaren passen we Agile meer en meer toe - veelal Scrum aangepast aan de omstandigheden.