

Basis standalone

Testen en verbeteren

hoofdstuk

2

Ontwikkelmethoden





Algemene informatie

Onderwerp	Ontwikkelmethoden
Leerdoel(en)	<ol style="list-style-type: none">1. De student maakt kennis met diverse veelgebruikte ontwikkelmethoden.2. De student verzamelt informatie en scheidt hoofd- van bijzaken.3. De student werkt samen in een klein team.4. De student plant werkzaamheden in een samenwerktool.5. De student maakt een powerpointpresentatie en presenteert deze aan de klas.
Vereiste voorkennis	Alle kennis van TV van T1-4 en H1 van T7
Kwalificatiedossier	<ul style="list-style-type: none"><input type="checkbox"/> B1-K1-W1: Plant werkzaamheden en bewaakt de voortgang<input type="checkbox"/> B1-K1-W2: Ontwerpt software<input type="checkbox"/> B1-K1-W3: Realiseert (onderdelen van) software<input checked="" type="checkbox"/> B1-K1-W4: Test software<input checked="" type="checkbox"/> B1-K1-W5: Doet verbetervoorstellen voor de software <input checked="" type="checkbox"/> B1-K2-W1: Voert overleg<input checked="" type="checkbox"/> B1-K2-W2: Presenteert het opgeleverde werk<input checked="" type="checkbox"/> B1-K2-W3: Reflecteert op het werk



Inhoudsopgave

Algemene informatie	2
Inhoudsopgave	3
Inhoud	4
Waterfall	4
Agile	5
Scrum.....	6
OTAP	6



Al zo lang mensen samenwerken wordt geprobeerd de samenwerking te coördineren, als mede de producten waar naartoe gewerkt wordt. Binnen de ICT is dat niet anders. Sinds decennia zijn er bepaalde methoden veel gebruikt. Zoals ook de inhoud van software verandert, zo verandert ook de inhoud van de ontwikkelmethoden. Hier gaan we dit hoofdstuk verder op in.

Inhoud

Waterfall

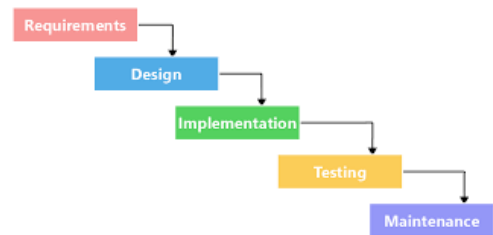
De watervalmethode is een methode die als sinds jaar en dag wordt gebruikt in de softwareontwikkeling. Ook vandaag de dag wordt deze methode nog steeds heel veel gebruikt.



Het idee van de watervalmethode is dat je project uit verschillende fasen bestaat. Elke fase is een voortvloeisel uit de vorige fase en start wanneer de vorige fase (bijna) is afgerond.

Veelgebruikte fasen zijn:

- Specificeren
- Ontwerpen
- Bouwen en implementeren
- Testen
- Onderhoud



Vaak start het proces met een inventarisatie. Er komt bijvoorbeeld van een opdrachtgever een vraag om een bepaald soort software. De requirements worden in kaart gebracht met documenten als een Programma van Eisen. Met een Functioneel – en Technisch Ontwerp wordt heel specifiek de software ontworpen. Ontwikkelteams gaan vervolgens die ontwerpen omzetten in de software. Wanneer alles af is wordt er getest en, zo nodig, aangepast. Aan het eind wordt het product opgeleverd en blijven de softwareontwikkelaars aanwezig bij het onderhoud van de software.

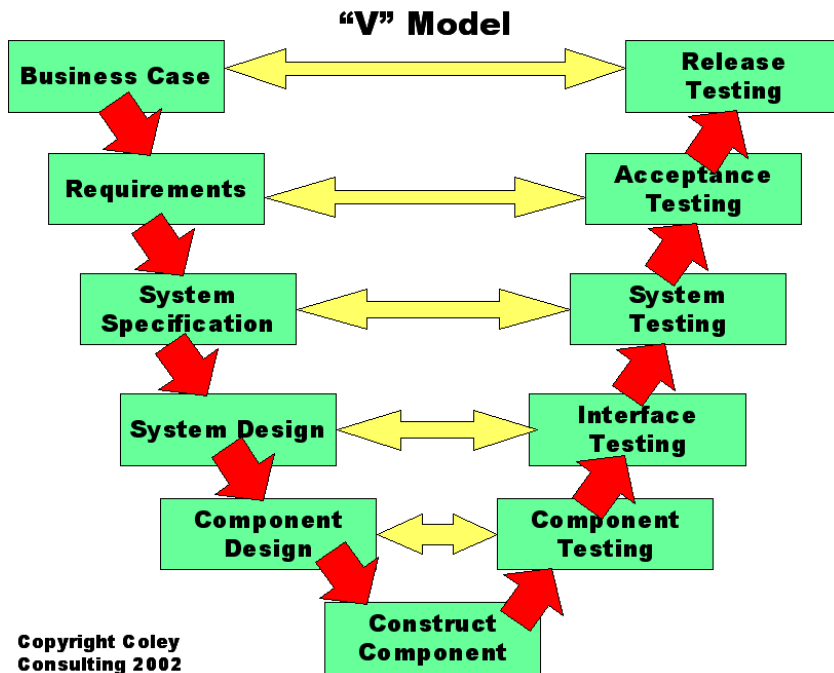
Heel logisch, zou je zeggen. Vooral omdat je misschien bepaalde documenten al van naam kent. Maar dit model heeft ook nadelen. Want tegenwoordig verandert er snel heel veel binnen de softwarewereld, maar ook bij wensen en eisen van gebruikers. Wat nu wanneer een opdrachtgever gedurende het proces ineens andere wensen en eisen krijgt? Zijn publiek kan immers ook ineens andere wensen en eisen krijgen.

Of wat nu wanneer je een opdrachtgever krijgt die zelfs aan het *begin* nog niet eens weet wat hij/zij wil? Maar dat er tijdens de softwareontwikkeling pas duidelijk wordt waar het naartoe gaat? Wat wanneer je pas met het testen er achter komt dat bepaalde systemen toch helemaal niet goed werken. Je bent dan al heel erg ver in het proces en moet dan mogelijk weer 'terug naar de tekentafel' en raakt dus heel veel tijd kwijt.

Wat zelfs nog wel eens een probleem bleek te zijn is dat de mensen uit de verschillende fasen ook een andere kijk op de software konden hebben. De development, operations en quality



assurance kregen het nog wel eens aan de stok met elkaar. Dat waren problemen waar veel ontwikkelaars tegenaan liepen.



Het V-model, een variant van de waterfall method waarin het testen wordt gelinkt aan de verschillende voorbereidende fasen

Agile

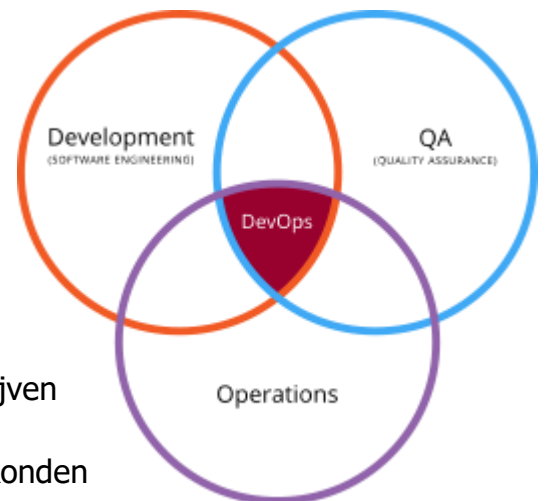
Doe dit nu zelf:

→ Bekijk [Agile Rhapsody](#)

Waar veel bedrijven op een gegeven moment meer behoefte aan hadden, was aan een minder starre ontwikkelmethode. Ze hadden behoefte aan meer beweging, aan een 'lenige' manier van werken in tegenstelling tot de stijve watervalmethode.

Agile is een Engels woord en het betekent ook 'lenigheid'. Bedrijven bedachten dat het handiger was wanneer het hele proces de opdrachtgever betrokken zou zijn tijdens het proces. Hierdoor konden veranderingen in wensen ook gelijk worden meegenomen. Door te gaan ontwikkelen in kleine stappen, telkens naar een werkende volgende versie van de software, werden de risico's enorm beperkt. Testen kon dan ook steeds plaatsvinden op een dergelijke 'iteratie'.

Communiceren deed men liever dagelijks mondeling in plaats van allerlei details vastleggen in grote logge documenten. Daarnaast werden teams samengevoegd: Quality Assurance, Development en Operations werden samen DevOps.





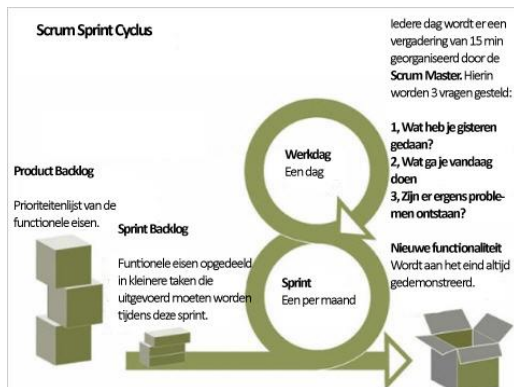
Doe dit nu zelf:

- Bekijk eens de volgende filmpjes over DevOps:
- DevOps in 5 minuten: klik [HIER](#).
 - DevOps bij de politie: klik [HIER](#).
 - Azure DevOps: klik [HIER](#).

Er is niet één agile-methode hetzelfde, net als dat er ook niet één watervalmethode hetzelfde is. Er zijn allemaal diverse manieren bedacht om agile te werken.

Scrum

Misschien wel de meest bekende variant van Agile werken: de [Scrum](#). De term komt uit de rugbywereld. Een spelhervatting gebeurt dan vaak met een scrum: Twee groepen spelers gaan dan voorovergebogen tegen elkaar induwen. Deze sterke samenwerking was inspiratie voor de bedenkers van de scrummethode. Hoewel scrum ook weer overal anders gebeurt zijn er ook veel overeenkomsten. Zo is er met scrum eigenlijk altijd:



- Een [scrum master](#), product owner, development team
- Sprints
- [Een daily standup](#)
- [Scrum poker](#)
- [Product Backlog](#)

OTAP

OTAP staat voor Ontwikkel, Test, Accepteer, Productie. Deze methode is in te passen in vele ontwikkelstrategieën, waaronder Waterfall, Scrum / Agile, DevOps. Deze methode wordt nog veel gebruikt.

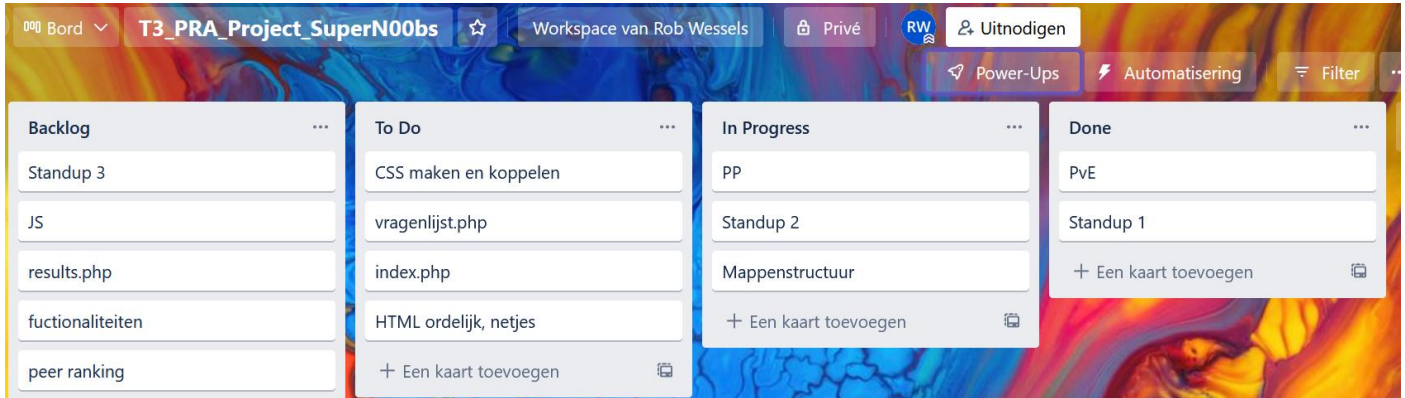
Doe dit nu zelf:

- Bekijk [deze site](#) voor meer info over OTAP.



Trello

Trello wordt gebruikt bij opdracht 2.1. Hier een voorbeeld van de kolomnamen die je daarvoor moet maken.



Je kunt nu **oefening 2.1** maken.