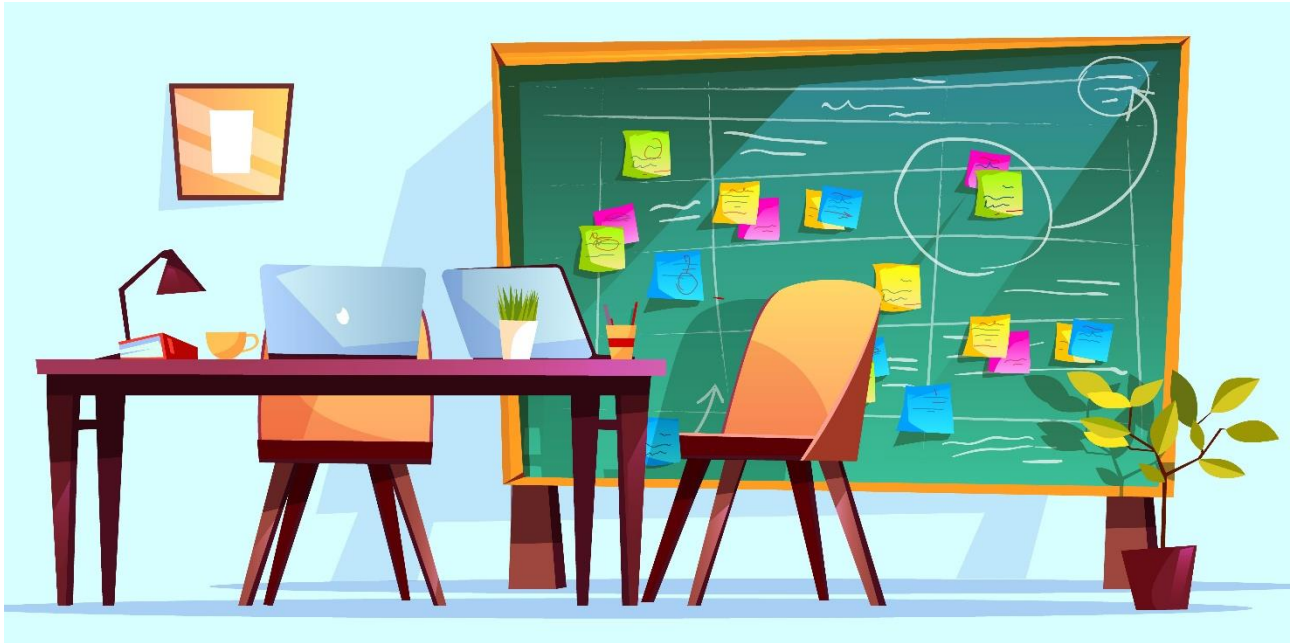




Modellen voor Agile Werken

Veel gebruikte modellen en overzichten voor Agile en Scrum



Willem Salentijn



Het plaatje op de voorkant Designed by Vectorpouch

Inhoud

Inleiding.....	3
1. Het Agile Manifesto.....	4
2. The New New Product Development Game	5
3. Agile methodologieën	6
4. Scrum Framework	7
5. Scrum Waarden.....	8
6. User Story.....	9
7. Epic, User Story, Task	10
8. Kanban.....	11
9. Burndown	12
10. Hoorn van onzekerheid	13
11. Scrum poker (planning poker).....	14
12. Daily Scrum (Standup)	15
13. Niko Niko kalender	16
14. Backlog	17
15. Scrum of scrums (Meta Scrum)	18
16. Nexus™	19
17. LeSS (Large-Scale Scrum).....	20
18. SAFe® (Scaled Agile Framework)	21
19. Subway map	22
20. Wat mensen motiveert	23
Zelf aan de slag?	24

Niets uit deze uitgave mag worden openbaar gemaakt en/of verveelvoudigd door middel van druk, fotokopie, microfilm en op welke andere wijze dan ook zonder voorafgaande schriftelijke toestemming van de auteur.



Inleiding

Begin deze eeuw werd ik als projectmanager ermee geconfronteerd dat er situaties zijn waarin je niet in een aantal logische stappen van idee tot eindproduct kon komen. In deze concrete situaties was het product al verkocht, maar hadden we nog niets. Het idee van de *Kraamkamer* werd geboren en in plaats van eerst te documenteren en vervolgens te maken deed ik met mijn team dat precies andersom. Mijn samenvatting van Agile is dan ook Focus, Focus op wat echt waarde toevoegt en als je dat hebt, begin je met documenteren, namelijk voor de gebruikersdocumentatie.

De principes van Agile en Scrum zijn eenvoudig, maar daarom niet minder complex in de toepassing. Vooral als je gewend bent op een bepaalde manier te werken. Voor werken met Agile en Scrum geldt dat de complexiteit zit in de eenvoud. Die is soms lastig te vatten en bij 5ST3PS lopen we er bijna dagelijks tegenaan dat mensen en organisaties het hele Agile en Scrum onnodig moeilijk maken. Daarom als eindejaarscadeau van het team van 5ST3PS dit modellenboek met 20 modellen en toepassingen.

Veel plezier!

Willem Salentijn, ook namens de rest van het team van 5ST3PS



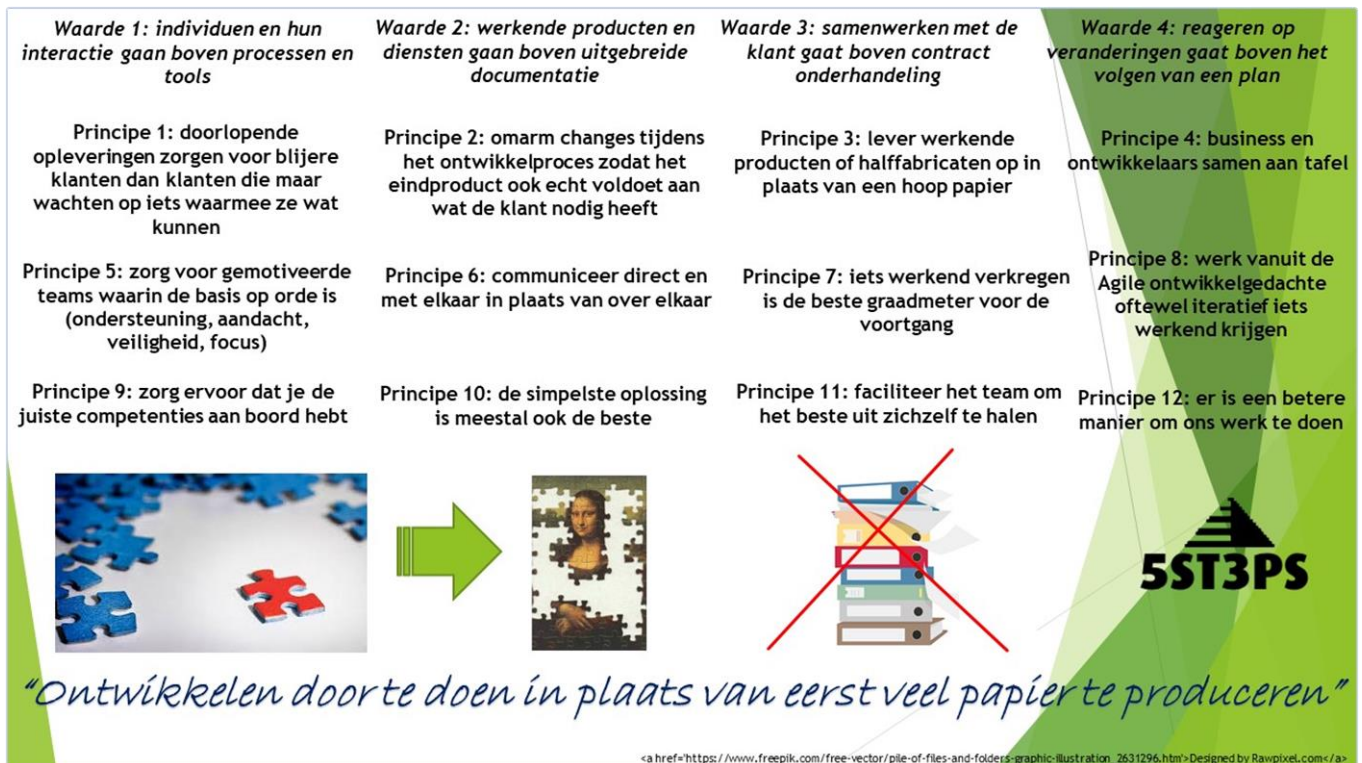
1. Het Agile Manifesto

Oorsprong

In Februari 2001 kwamen 17 ontwikkelaars samen in een ski resort in Utah. Op dat moment waren er een aantal methoden voor Agile ontwikkelen, zoals *Extreme Programming* en *Scrum*. Deze ontwikkelaars wilden tot een integratie komen en zowaar het Agile Manifesto was geboren. Het Agile Manifesto kent 4 waarden en 12 principes.

Toepassing

De waarden en principes van het Agile Manifesto zijn vooral een basis voor het denken in iteratief ontwikkelen. Tegenover het iteratief ontwikkelen staat het ontwikkelen in waterval, waarbij je telkens een stap moet hebben afgerond voordat je de volgende stap neemt. Het Agile Manifesto geeft houvast voor het ontwikkelen in iteraties waarbij je verschillende keren door dezelfde fase kunt gaan.



Figuur 1. Het Agile Manifesto

Meer lezen? <https://agilemanifesto.org/>



2. The New New Product Development Game

Oorsprong

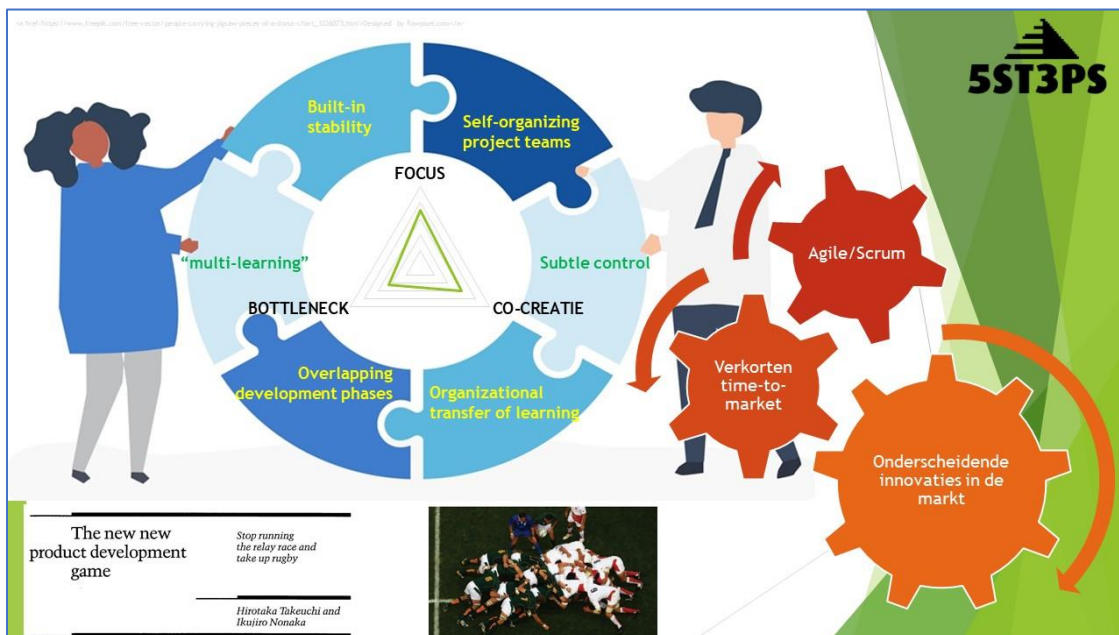
Het begrip Scrum werd gebruikt als analogie voor iteratief ontwikkelen door Takuchi en Nonaka in hun baanbrekende artikel over New New Product Development in 1986. In dit artikel kwamen de heren tot de conclusie dat de ontwikkelingen zo snel gaan dat je op een andere manier innoveren moet faciliteren. Dus niet in kop-en-staart trajecten waarbij je gebaseerd op uitgebreide studies en plannen toewerkt naar een product gereed voor marktintroductie, maar veel meer iteratief waarbij de ontwikkelaars elkaar de bal toespelen en samen bepalen hoe ze richting het einddoel komen. Vandaar de analogie met een Scrum.

Toepassing

In samenhang met het Agile Manifesto het uitleggen van de kenmerken van iteratief ontwikkelen tegenover die van waterval methodieken en meer specifiek New Product development (NPD).

Deze kenmerken zijn

1. Built-in instability (je hebt een doel maar weet bij start nog niet hoe je daar komt)
2. Self-organizing project teams (teams organiseren zichzelf)
3. Overlapping development phases (ontwikkelfases overlappen elkaar)
4. "Multilearning" (leren van elkaar en met elkaar)
5. Subtle control (visueel management aanwezig zodat transparant is wat ieder doet)
6. Organizational transfer of learning (leren van elkaar)



Figuur 2. The New New Product development Game

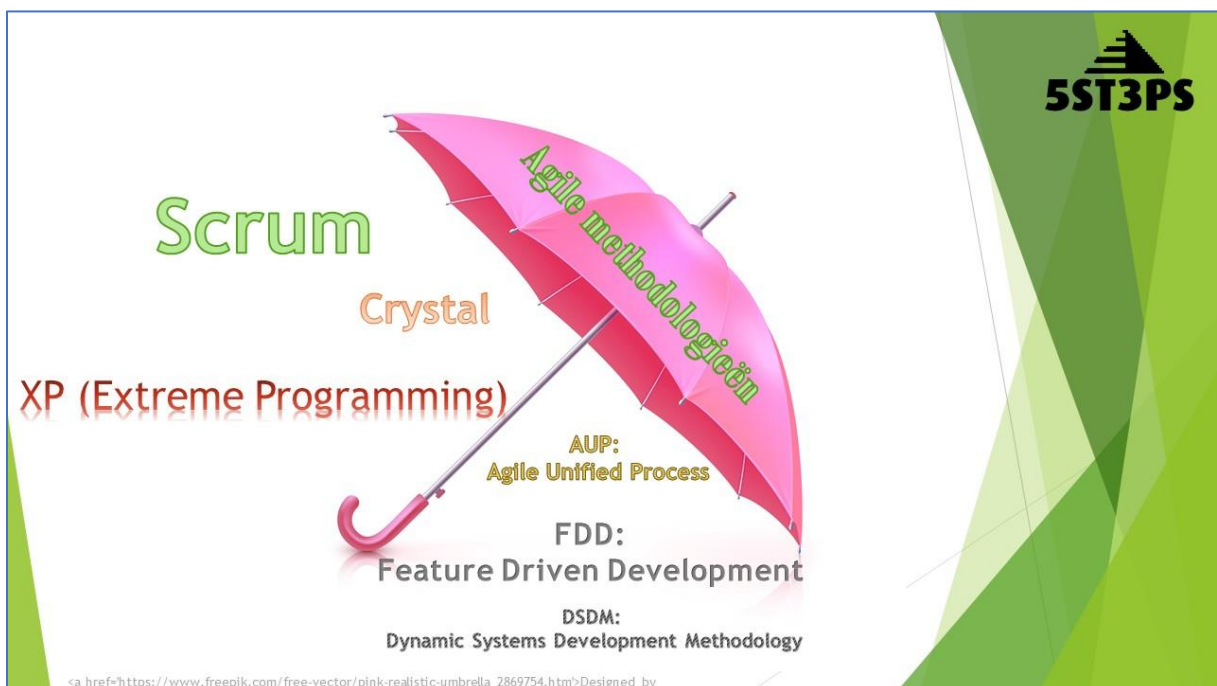
3. Agile methodologieën

Oorsprong

Agile is een benadering en is synoniem voor het iteratief ontwikkelen van functionaliteit. Iteratief ontwikkelen staat tegenover traditionele ontwikkeltechnieken waarin stap-voor-stap vooruit wordt gewerkt om de functionaliteit op te leveren. Hierin wordt pas een volgende stap gezet als de vorige klaar is. Daarom wordt er ook vaak gesproken over waterval methodieken. Net zoals in een waterval kun je maar in één richting vooruit en kun je niet terug. Iteratieve en incrementele ontwikkelmethoden zijn in ieder geval terug te voeren naar IBM in 1957. In de jaren negentig van de vorige eeuw kregen Agile ontwikkelmethoden steeds meer aandacht door de opkomst van automatisering en de problemen in IT-ontwikkel projecten.

Toepassing

In elke methode zitten elementen die je kunt toepassen in het iteratief of incrementeel ontwikkelen. Zo kent *Crystal* een score toe aan het ontwikkelproject bepaald door de omvang en het risico op systemschade. Deze scores zijn de verschillende kleuren die kristallen vertegenwoordigen. *Feature-Driven-Development* heeft een focus op het ontwikkelen van features die samen dan het systeem maken wat ontwikkeld wordt. *Dynamic Systems Development Method* gaat onder andere uit van het werken in time-boxes en de 80/20 regel. *Agile Unified Process* heeft onder andere als uitgangspunt dat de teamleden alle soorten werkzaamheden moeten kunnen uitvoeren, van ontwerpen en configureren tot testen en implementeren. Uit *Extreme Programming* komt het principe van pair programming. Door twee ontwikkelaars samen te laten werken in bijvoorbeeld code schrijven, wordt de kwaliteit beter en betaalt deze inspanning zich vanzelf terug.



Figuur 3. Agile Methoden

4. Scrum Framework

Oorsprong

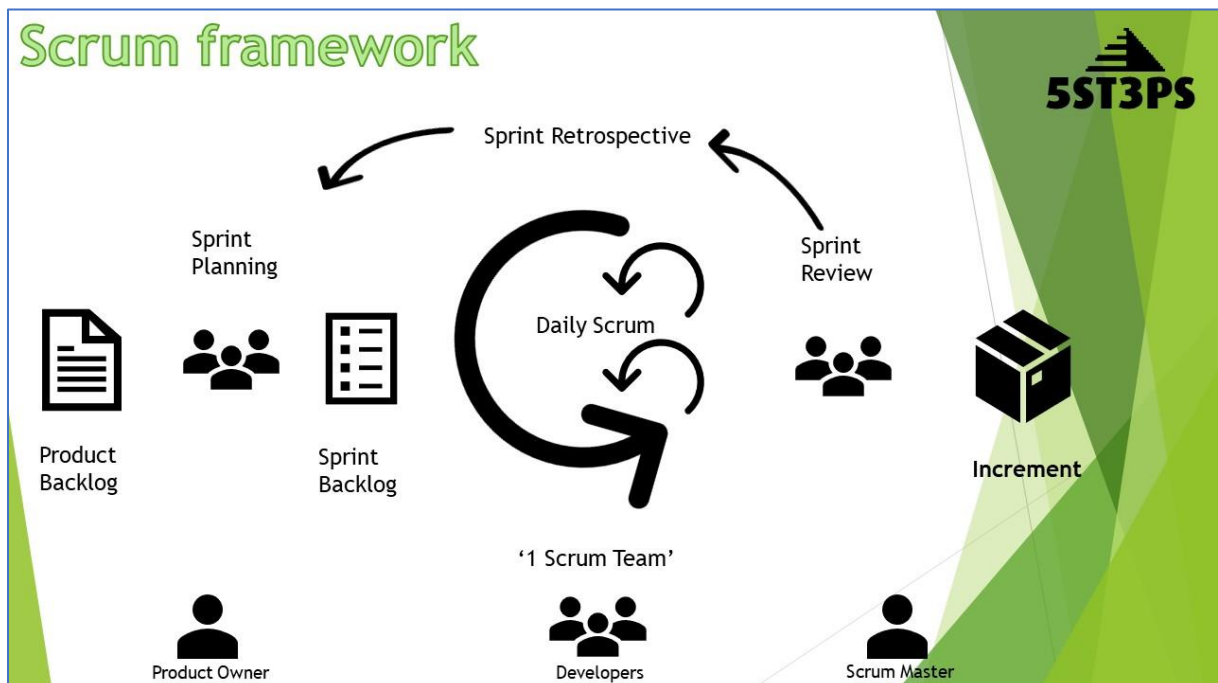
Scrum is een framework welke beschreven is door Ken Schwaber en Jeff Sutherland in de Scrum Guide. Het framework zelf werd door Schwaber en Sutherland voor het eerst gepresenteerd in 1995 op een IT conferentie. In dit framework heb je een aantal rollen, een aantal gebeurtenissen (events) en een aantal 'artifacts'. Het framework geeft richting aan wat je moet faciliteren om op een Scrum manier te werken.

Toepassing

Het Scrum Team bestaat uit een Product Owner, de Developers en een Scrum Master. De Product Owner is verantwoordelijk voor het maximaliseren van de waarde die het resultaat is van wat de Developers hebben ontwikkeld. De Developers zijn dan ook verantwoordelijk voor de waardecreatie. De Scrum Master coacht en begeleidt de manier van werken.

Scrum schrijft vier formele gelegenheden voor ten behoeve van inspectie en aanpassing. Deze zijn de Sprint Planning, Daily Scrum, Sprint Review en de Sprint Retrospective.

De 'artefacten' van Scrum vertegenwoordigen werk of waarde die transparantie bieden en bijdragen aan het eindproduct, welke de belangrijkste waarde vertegenwoordigt. Deze 'artefacten' zijn de Product Backlog, de Sprint Backlog en het Increment.



Figuur 4. Scrum Framework

Meer lezen? <https://www.scrumguides.org/>



5. Scrum Waarden

Oorsprong

In 2016 werden de Scrum Waarden toegevoegd aan de Scrum Guide. De Scrum Waarden beschrijven de zachtere kant van Scrum en zijn te beschouwen als randvoorwaarden en een mindset om Scrum te kunnen toepassen.

Toepassing

De waarden die geborgd moeten zijn in de manier van werken zijn

- ✓ Moed: teamleden moeten de veiligheid hebben om hun hoofd boven het spreekwoordelijke maaiveld uit te steken
- ✓ Focus: teamleden moeten gefocust zijn op werk in de sprint en niet afgeleid worden door andere dingen
- ✓ Commitment: teamleden moeten betrokken zijn bij het team en het te bereiken resultaat
- ✓ Respect: iedereen in een team is gelijkwaardig en de werklast moet evenredig verdeeld zijn
- ✓ Openness: iedereen is transparant en er zijn geen verborgen agenda's



Figuur 5. Scrum Waarden, bron <https://www.scrum.org/resources/scrum-values-poster>



6. User Story

Oorsprong

Het begrip User Story werd voor het eerst gebruikt door Alistair Cockburn in 1998. In Extreme Programming, een Agile ontwikkelmethode, is een user story onderdeel van de planning game. Een user story is vooral geschreven vanuit het perspectief van de afnemer of de gebruiker. Ontwikkelaars gaan vervolgens aan de slag om invulling te geven aan de behoefte die is beschreven in de user story.

Toepassing

De meest gebruikte vorm van user stories bestaan uit drie delen, namelijk wie iets wilt in welke rol, wat diegene wilt en waarom. De Definition of Done (DoD) is het acceptatie criterium welke aangeeft wanneer een user story dan ook daadwerkelijk ingevuld is.

A template for writing a user story. It is titled "User Story" and contains four lines of text, each followed by a dotted line for input. The text is: "Als (rol)", "Wil ik (functionaliteit)", "Zodat (waarom)", and "Ik weet dat deze functionaliteit klaar is als ik". The form is set against a background of overlapping green and yellow geometric shapes. The 5ST3PS logo is visible in the top right corner of the form area.

Figuur 6. Template User Story

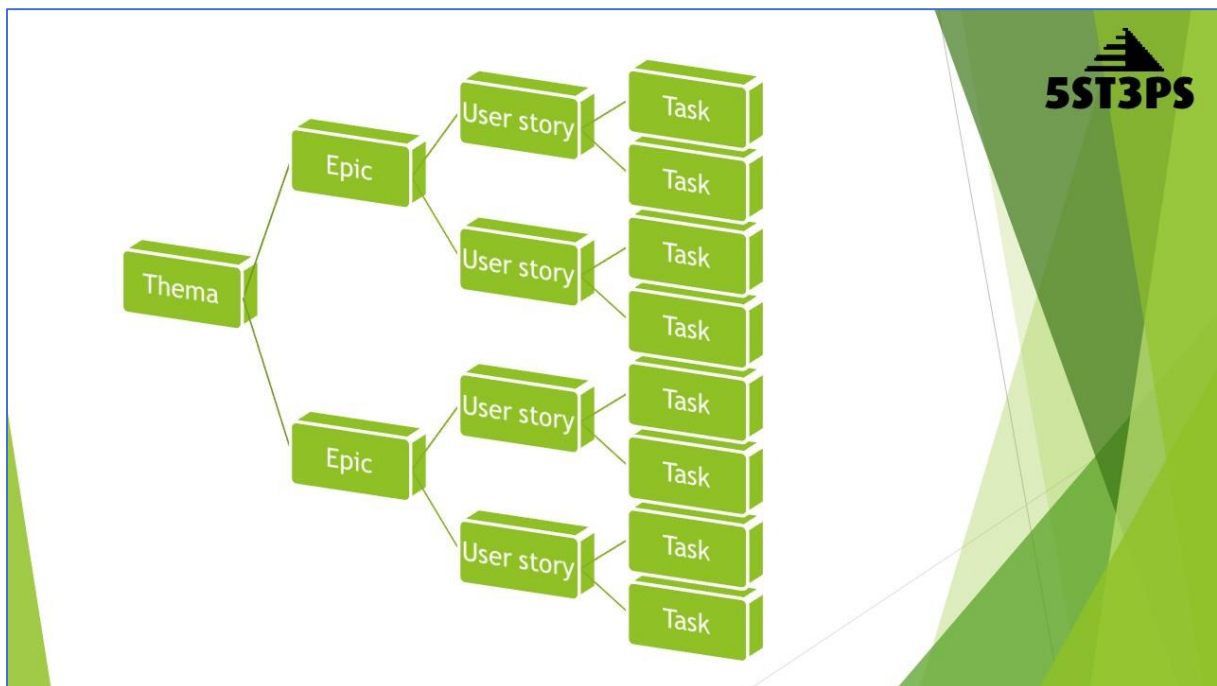
7. Epic, User Story, Task

Oorsprong

In Agile draait het om het invullen van een klantbehoefte. Deze klant kan trouwens zowel intern als extern zijn. De klantbehoefte op zichzelf kunnen onderdeel zijn van een groter geheel. Mike Cohn (2004) introduceerde daarom het concept van de epic als een all-over user story. De verdere breakdown van user stories gebeurt in tasks. Een task kan door een ontwikkelaar in één keer worden uitgevoerd.

Toepassing

In de Backlog staan de user stories. Deze zijn onderdeel van een grotere user story, de epic. User stories op zichzelf worden weer verder afgepeld in tasks. Zowel Epics, als user stories als tasks gaan over hetzelfde alleen dan elk op een ander detail niveau. Uiteindelijk doel is dat in de teams de tasks worden uitgevoerd en die samen optellen tot de invulling van een user stories, terwijl de user stories samen voorzien in een epic die je als organisatie een voordeel oplevert.



Figuur 7. Hiërarchische boom voor user stories

Meer lezen? Cohn, M. (2004). User stories applied: For agile software development. Addison-Wesley Professional.

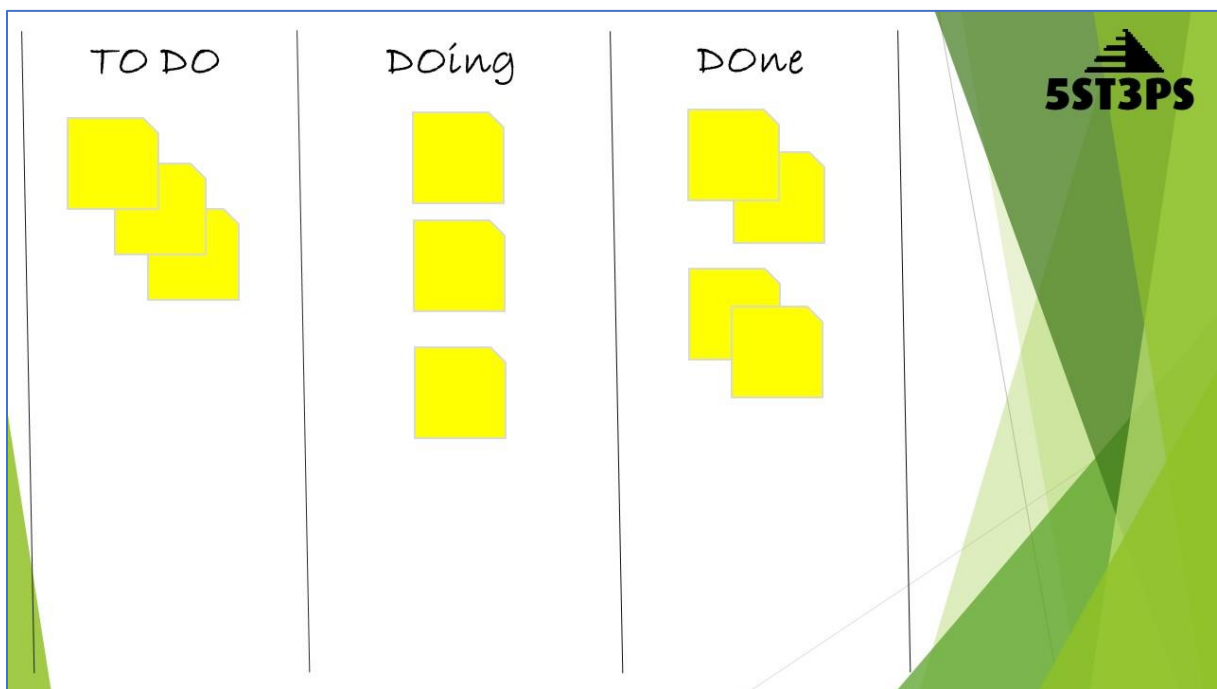
8. Kanban

Oorsprong

De Kanban is een Lean concept en betekent visueel teken. In het Toyota Productie Systeem is een Kanban een teken dat je iets moet doen en geeft de Kanban aan hoewel werk er onderhanden is (Work-in-Progress). Door het gebruiken van een Kanban blijft de productielijn continu in beweging (continuous flow). In Agile wordt Kanban gebruikt om als team gericht bezig te zijn met de ontwikkeltaken. Door de taken op een bord te schrijven (Kanban bord) is duidelijk wat het team moet doen en is het transparant wie waarmee bezig is en wat er afgerond is.

Toepassing

De taken in de sprint worden opgeschreven op het Kanban bord (er zijn uiteraard ook programma's hiervoor) en iedereen pakt 1 taak naar zich toe (Doing) en zet deze als deze klaar is op 'Done'



Figuur 8. Kanban board

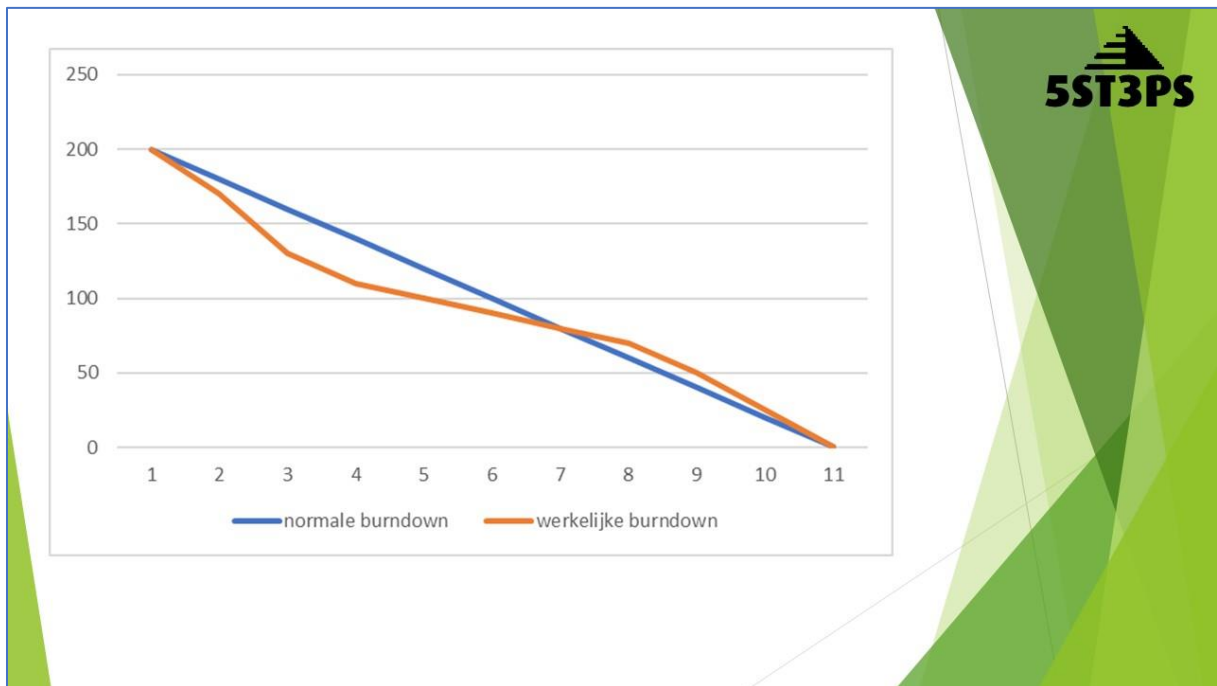
9. Burndown

Oorsprong

In 2000 ontwikkelde Ken Schwaber een toolkit om Scrum teams te ondersteunen. Onderdeel hiervan was de Burndown, waarmee je kunt visualiseren in welk tempo een Scrum team werk oplevert.

Toepassing

Aan elke Sprint koppel je de taken die worden uitgevoerd die samen de user story of user stories maken die worden opgeleverd. Hieraan koppel je de inspanning door bijvoorbeeld Scrum Poker. De inspanningen tel je op en elke keer als er iets gereed wordt gemeld, haal je het aantal punten van het totaal af. Zo krijg je een neergaande lijn welke dan de burndown van de sprint voorstelt. De snelheid van het team (velocity) kun je dan bepalen door het aantal punten wat is opgeleverd te delen door het aantal dagen waarin het werk is gedaan.



Figuur 9. Burndown

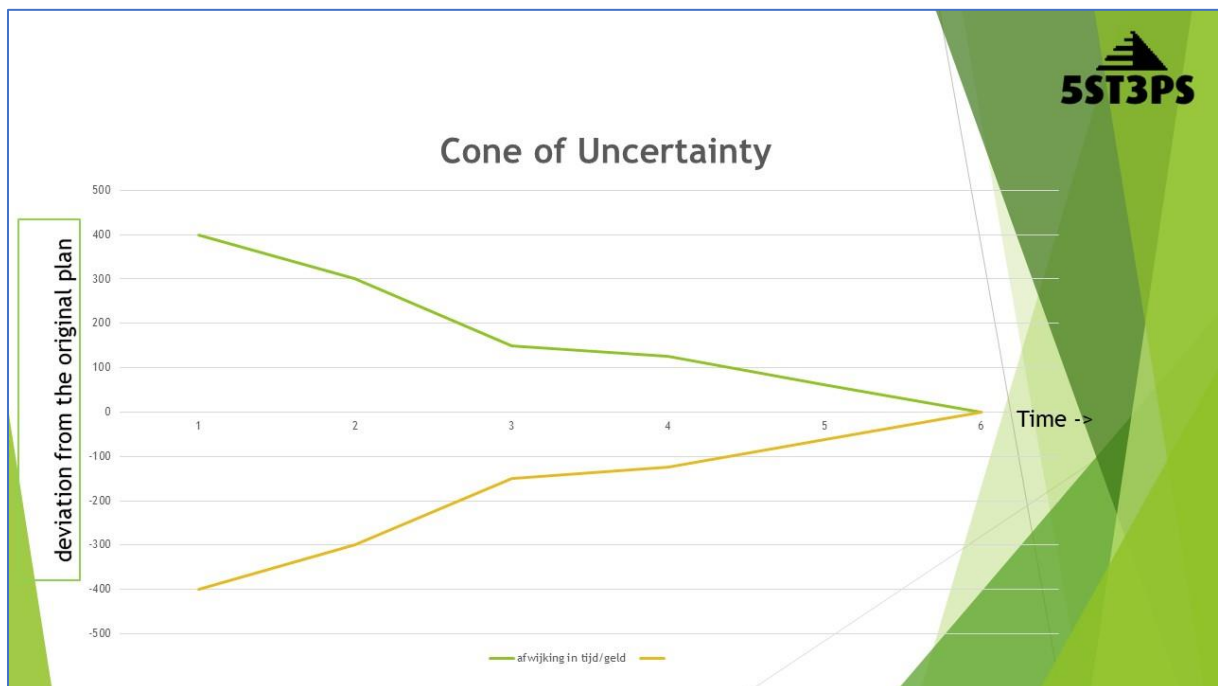
10. Hoorn van onzekerheid

Oorsprong

De hoorn van onzekerheid (Cone of Uncertainty) geeft aan welke de marge is die je bij de start van iets nieuws moet aanhouden. Het concept dateert reeds uit 1958 (Gorey) en werd opgepikt door Barry Boehm in 1981 als een manier om in ontwikkeltrajecten om te gaan met de onzekerheid die inherent is aan ontwikkeling. De term Cone of Uncertainty werd voor het eerst genoemd in 1997 in de Software Project Survival Guide van McConnell.

Toepassing

Bij de start van een sprint is het zeker met iets geheel nieuws en een geheel nieuw team onduidelijk hoe snel nu iets wordt opgeleverd. Daarom heeft het weinig zin om op moment 0 een concrete inschatting te maken. Op basis van empirische gegevens duurt het voor iets geheel nieuws drie sprints om een nauwkeurige inschatting te maken. Hiervoor wordt de burndown van het team gebruikt zodat je na drie sprint een vrij nauwkeurige voorspelling kunt doen hoeveel er in welk tempo wordt opgebrand.



Figuur 10. Hoorn van onzekerheid

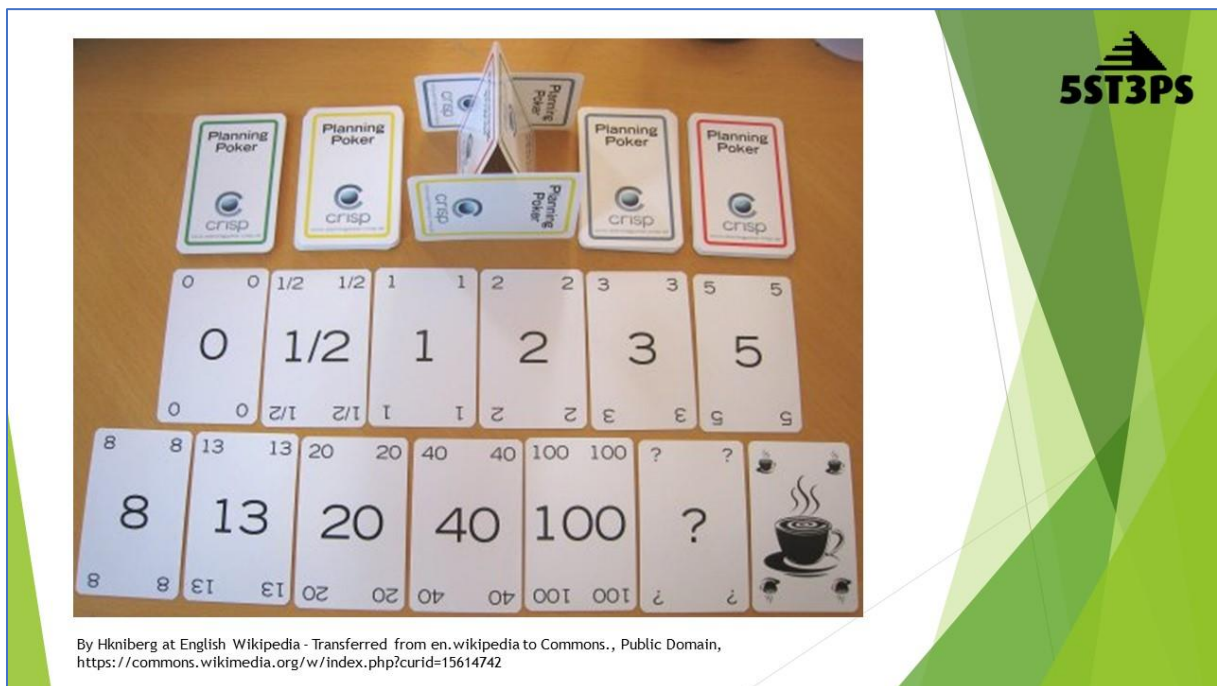
11. Scrum poker (planning poker)

Oorsprong

In 1950 werd de Delphi methode ontwikkeld om uitspraken te kunnen doen over toekomstige gebeurtenissen. In deze methode geven een aantal experts een oordeel en vervolgens kunnen de experts (anoniem) feedback op elkaars oordelen te geven. Dit proces herhaalt zich totdat er overeenstemming is. Mike Cohn (2005) gebruikte deze methode om in Agile ontwikkeltrajecten inschattingen te kunnen doen.

Toepassing

Als het team aan de slag gaat met de planning (Sprint Planning) dan licht de Product Owner de user story toe die gerealiseerd moet worden. De ontwikkelaars bepalen de benodigde taken en vervolgens pakt elk teamlid een kaart uit het scrum poker (of planning poker) welke de inspanning of zwaarte voorstelt voor die taak. Een nul staat voor 'reeds gedaan' en een 100 staat bij wijze van spreken voor de reis naar de maan. Een vraagteken staat voor geen idee. Doordat ieder voor zichzelf bepaalt welke de inspanning is en vervolgens de kaarten aan elkaar laat zien, krijg je een consensus over de inspanning en gaan gesprekken over de verschillen in de inschatting. De Scrum Master faciliteert dit.



Figuur 11. Scrum Poker of Planning Poker

Meer lezen? Cohn, M. (2005). Agile estimating and planning. Pearson Education.

12. Daily Scrum (Standup)

Oorsprong

Ken Schwaber en Jeff Sutherland hadden oorspronkelijk drie vragen voor de dagelijkse meeting met het Scrum team, namelijk:

1. Wat heb je gisteren gedaan?
2. Wat ga je vandaag doen?
3. Wat houdt je tegen om jouw doel te bereiken?

Door de eerste vraag krijg je antwoord op wat er is afgerond. Op de tweede vraag krijg je antwoord op wat je vandaag gaat doen. En door de derde vraag kun je eventuele hindernissen of impediments uit te weg ruimen.

In 2017 maakten de heren de vragen optioneel, omdat ze zagen dat teveel teams de antwoorden verplicht opdreunen en het doel van de vragen juist is om het gesprek te krijgen hoe je als team het resultaat gaat bereiken.

Toepassing

Bij de standup wordt bepaald wat al klaar is, wat nu gedaan moet worden en welke hindernissen weggenomen moeten worden. Dat kan gefaciliteerd worden door de drie oorspronkelijke vragen zoals in vorige sectie beschreven. Een alternatief is de shocktherapie zoals beschreven door onder anderen Sutherland (2009) waarin maar 1 vraag wordt gesteld, *Waarom is het eindresultaat van deze sprint nog niet klaar?*



1. Wat heb je gisteren gedaan?
2. Wat ga je vandaag doen?
3. Wat houdt je tegen om jouw doel te bereiken?

Figuur 12. Drie vragen voor de daily scrum of standup



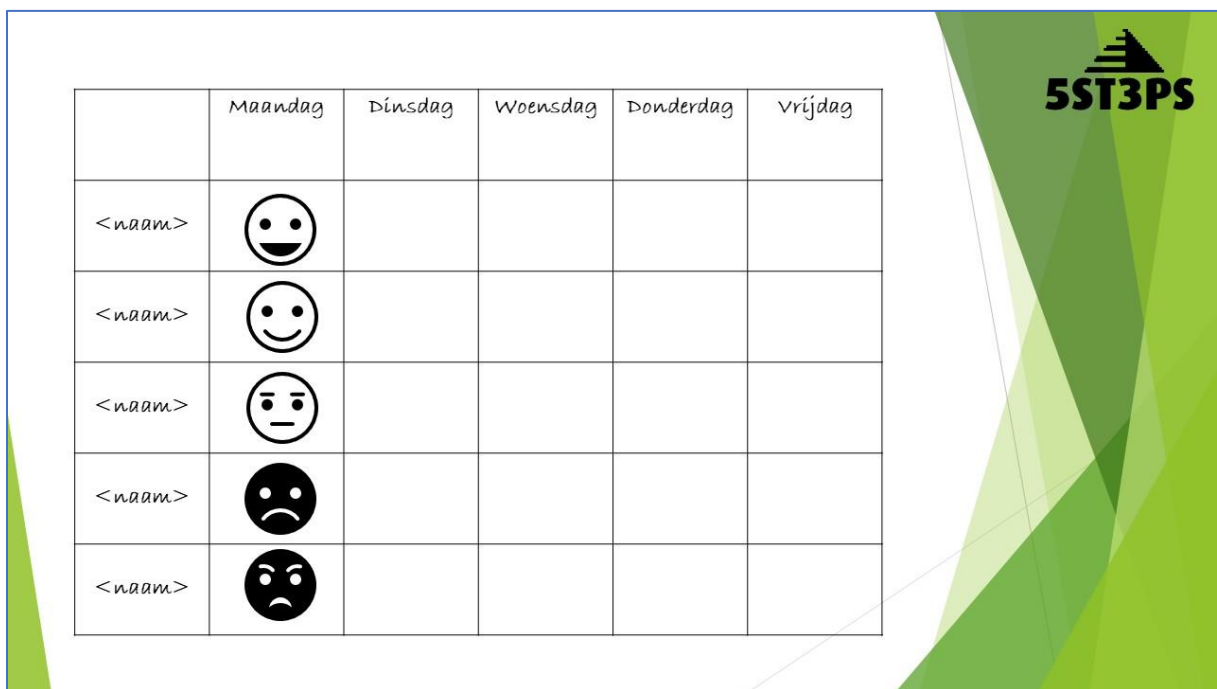
13. Niko Niko kalender

Oorsprong

Niko is Japans voor 'lach'. Bevlogen medewerkers maken minder fouten, zijn productiever en hebben betere ideeën. Een manier om de bevoegenheid van de medewerkers te visualiseren is de Niko-Niko kalender. Een voorloper van de Niko-Niko kalender is de energie seismograaf, welke het energie niveau aangeeft in de groep.

Toepassing

Aan het einde van de dag geeft elk teamlid een smiley voor hoe hij terugkijkt op die dag. Een big smile staat uiteraard voor boven verwachting waarde gecreëerd en een donkere smiley geeft aan dat de dag niet heeft opgeleverd wat vooraf was gedacht. Dan kan dat een hindernis zijn die tijdens de daily scrum kan worden besproken om zo in teamverband op te lossen.

A grid for a Niko-Niko calendar. The grid has 6 columns and 6 rows. The columns are labeled 'Maandag', 'Dinsdag', 'Woensdag', 'Donderdag', and 'Vrijdag'. The rows are labeled with '<naam>' in the first column. The first row of the grid is empty. The second row has a big smiley face in the 'Maandag' column. The third row has a medium smiley face in the 'Maandag' column. The fourth row has a neutral face in the 'Maandag' column. The fifth row has a sad face in the 'Maandag' column. The sixth row has a very sad face in the 'Maandag' column. All other cells in the grid are empty. The grid is set against a background with a green geometric pattern and the 5ST3PS logo in the top right corner.

	Maandag	Dinsdag	Woensdag	Donderdag	Vrijdag
<naam>	😊				
<naam>	🙂				
<naam>	😐				
<naam>	☹️				
<naam>	😡				

Figuur 13. Niko-Niko kalender

14. Backlog

Oorsprong

In zijn simpelste vorm is een Backlog een *To Do list* die wordt beheerd door de Product Owner. In de Backlog staan alle zaken die waarde toevoegen en je als organisatie een voordeel opleveren in de slag om de klant en de strijd met concurrenten. Nu waarde zich vertaalt in een product waarvoor de klant bereid is te betalen, hebben we het over een Product Backlog en is een subset hiervan de Sprint Backlog. De Sprint Backlog bestaat uit de user stories die in een sprint worden afgerond.

Toepassing

De Product Owner (PO) beheert de Product Backlog en zet er de epics en user stories in. Daaraan geeft de PO een prioriteit. Het team bepaalt de inspanning (zie eerdere item over Scrum Poker). Als op een user story niet de impact bepaald kan worden, dan moet het team deze user story verder onderzoeken. Een user story waarvan de inspanning nog niet bepaald kan worden en waarvoor het team nader onderzoek moet doen, heet een Spike. Een Spike moet de uitzondering zijn.



Figuur 14. Backlog

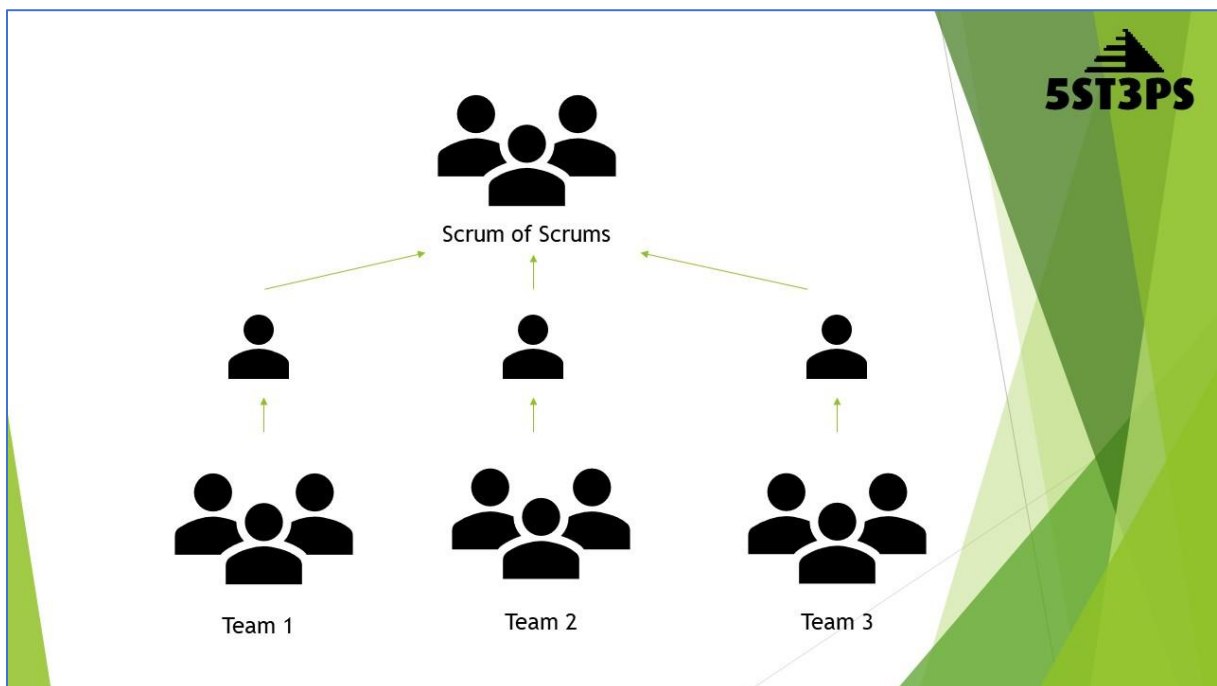
15. Scrum of scrums (Meta Scrum)

Oorsprong

In Scrum werkt in de ideale situatie iedereen dedicated aan een project. Nu je onderling afhankelijkheden hebt tussen scrum teams, beschreef Jeff Sutherland (2001) een model om deze afhankelijkheden te coördineren. Dit model is de Scrum of Scrums of Meta Scrum.

Toepassing

Elk scrum team levert 1 ambassadeur, dat kan de Scrum Master zijn of een ontwikkelaar, aan een meta team, de Scrum of scrums. In deze meeting worden de onderlinge afhankelijkheden besproken en benoemd. Deze afhankelijkheden en hindernissen (impediments) worden in een eigen backlog bijgehouden. Over deze impediments worden afspraken gemaakt en opgevolgd.



Figuur 15. Scrum of scrums



16. Nexus™

Oorsprong

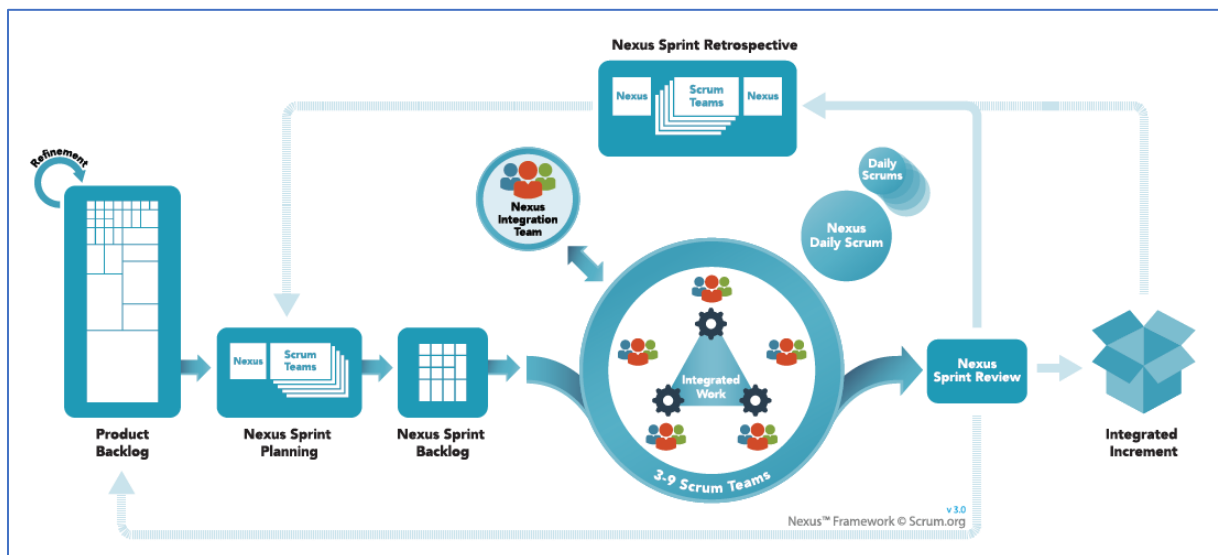
Nexus™ is een framework welke is bedacht door Ken Schwaber en gepubliceerd in 2015. Nexus™ is een framework naast Scrum welke antwoord geeft op de vraag hoe je met verschillende Scrum teams werkt om samen resultaat op te leveren. Uit de Nexus guide (2018) ontleen we de volgende definities:

Nexus (n): a relationship or connection between people or things

Nexus is a framework consisting of roles, events, artifacts, and rules that bind and weave together the work of approximately three to nine Scrum Teams working on a single Product Backlog to build an Integrated Increment that meets a goal.

Toepassing

Net zoals bij Scrum bestaat het Nexus™ framework uit een aantal rollen, artefacten en ceremonies. In de rollen is de rol van de *Nexus Integration Team Members* net iets anders dan die van de ontwikkelaars in het Scrum team. Deze *Nexus Integration Team Members* moeten ervoor zorgen dat de increments waarvoor onderling moet worden samen gewerkt, worden opgeleverd aan het einde van de sprint. Naast de Daily Scrums is er dan ook een *Daily Nexus Scrum meeting*.



Figuur 16. Nexus™ Framework for Scaling Scrum <https://www.scrum.org/resources/online-nexus-guide>

Meer lezen? Schwaber, K. (2015). Nexus Guide. The Definitive Guide to Nexus: The exoskeleton of scaled Scrum development. PDF). scrum. org.

17. LeSS (Large-Scale Scrum)

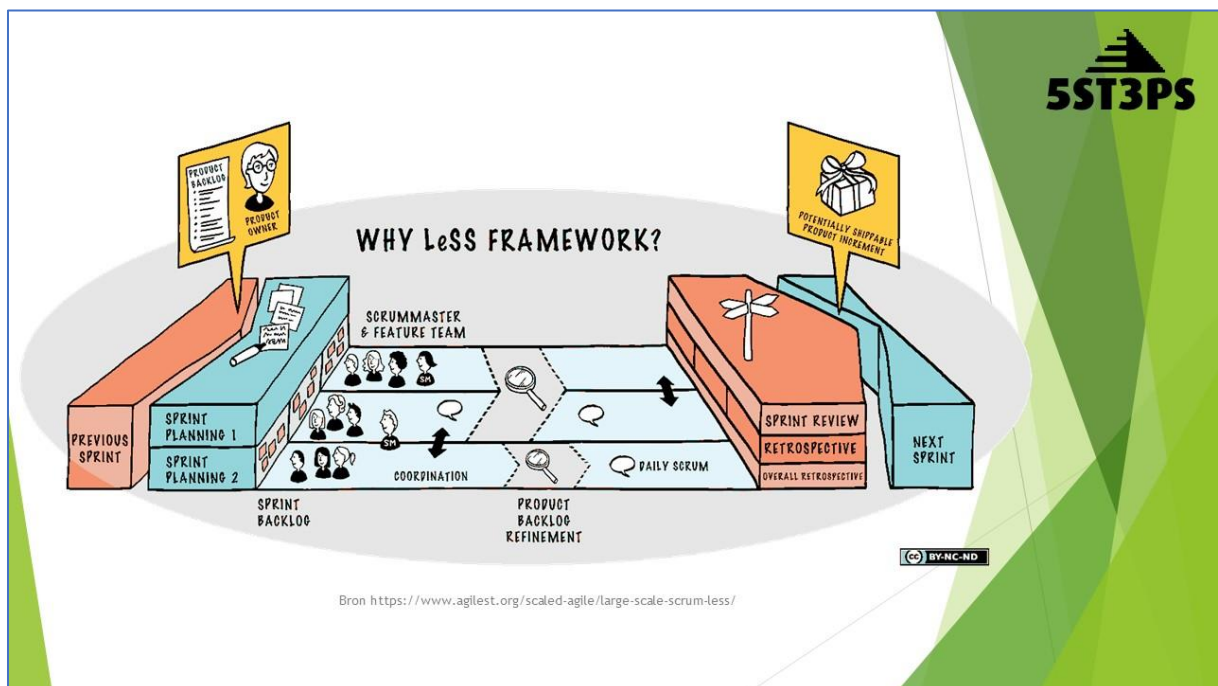
Oorsprong

Bas Vodde en Craig Larman ontwikkelden in 2015 het LeSS framework om projecten op grote schaal te kunnen managen. Het (basis) Scrum framework is heel erg gericht op een enkel project, terwijl in een organisatie meerdere projecten naast elkaar lopen en vaak ook samen een bepaald resultaat moeten opleveren.

Toepassing

LeSS (Large-Scale Scrum) is te beschouwen als een toevoeging op het Scrum framework zodat de meeste principes nog steeds van toepassing zijn. Verschillen zitten in de volgende zaken:

De eerste sprint planning is een geïntegreerde planning met alle teams. De tweede sprint planning is voor elk team, maar wel tegelijkertijd zodat er onderling afstemming kan plaatsvinden. Bij de Daily Scrum kunnen ontwikkelaars van andere Scrum teams aanwezig zijn. Er is een overall Backlog die beknopt aangeeft welke de user stories zijn van elk team. Refinements gebeuren dan ook bij voorkeur met meerdere teams om zo van elkaar te leren. In de verschillende reviews Review zijn voor dezelfde reden meerdere mensen uit verschillende teams aanwezig. De Retrospective daarentegen is wel gemeenschappelijk voor alle teams.



Figuur 17. LeSS Framework

Meer lezen? Larman, C., & Vodde, B. (2016). Large-scale scrum: More with LeSS. Addison-Wesley Professional.

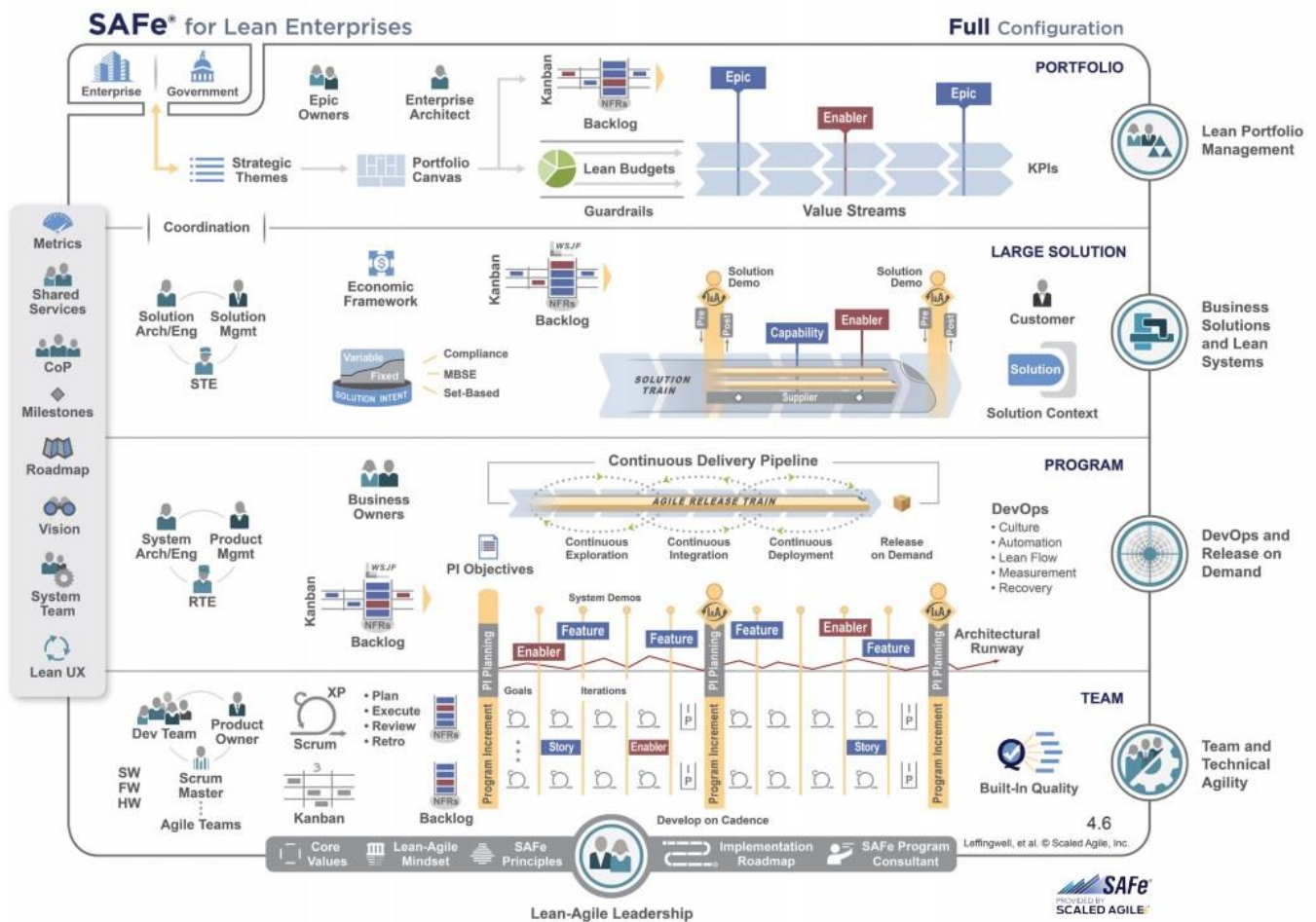


18. SAFe® (Scaled Agile Framework)

Oorsprong

In de basisvorm in Scrum heel erg gericht op het projectmatig ontwikkelen voor één resultaat. Nu in organisaties er meerdere resultaatpaden naast elkaar lopen, is een vraag hoe je Scrum opschaalt. Methoden als Scrum of Scrums, Nexus™ en LeSS geven daar invulling aan. Naast het opschalen van Scrum is een ander probleem dat organisaties traditioneel in een hiërarchie zijn georganiseerd en je in Scrum daarmee eigenlijk geen rekening houdt nu de waarde wordt bepaald door de Product Owner, het Ontwikkelteam de waarde realiseert en de Scrum Master in de werkwijze coacht. Zeker voor grote organisaties die verschillende hiërarchische lagen kennen, vergt dit veel afstemming en onderling begrip. SAFe is een raamwerk welke heel erg lijkt op de opzet in organisaties met hiërarchische lagen en is daarom ook populair bij grote organisaties.

Toepassing



Figuur 18. SAFe 4.6. bron <https://www.scaledagileframework.com/> Hier kun je ook meer lezen over SAFe.



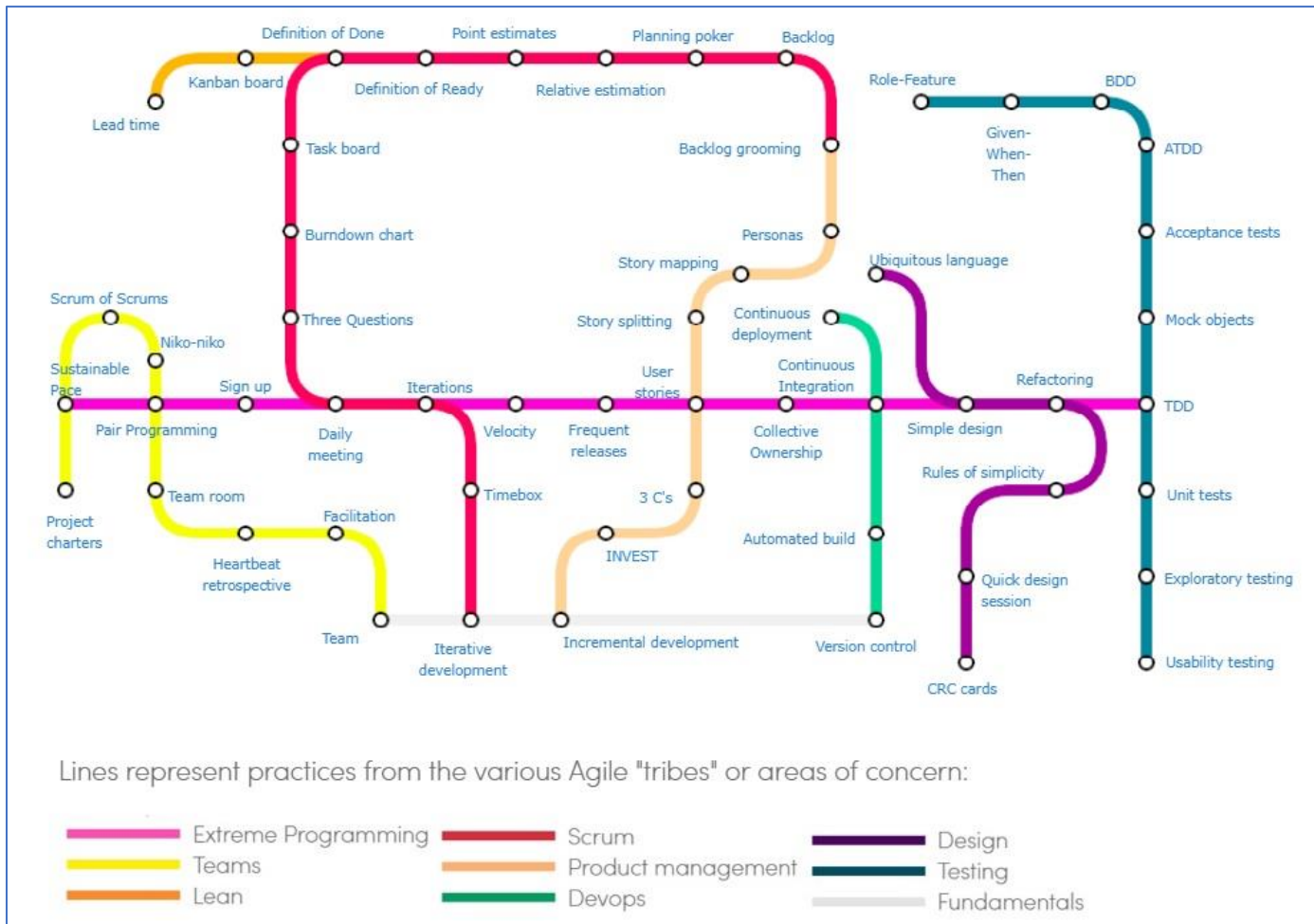
19. Subway map

Oorsprong

Agile Alliance is een non-profit organisatie voor Agile ontwikkelaars en is opgericht in 2001. Agile Alliance heeft een Subway map ontwikkeld waarin de verschillende toepassingen in de Agile ontwikkelfamilie zijn terug te vinden.

Toepassing

In de Subway map heb je de verschillende begrippen in onderling overzicht. Het geeft overzicht aan de beroepsbeoefenaar.



Figuur 19. Subway map bron <https://www.agilealliance.org/>

20. Wat mensen motiveert

Oorsprong

In het boek Drive (2009) beschrijft Daniel Pink wat mensen en vooral professionals werkelijk motiveert. Oude theorieën over wat mensen beweegt zijn vooral gebaseerd op straf en beloning. Dit zijn extrinsieke motivatoren en hebben veel minder effect dan intrinsieke motivatoren. Uiteraard moeten mensen voldoende betaald worden, maar factoren zoals zelfstandigheid, meesterschap en zingeving zijn veel belangrijker in het motiveren en presteren van mensen.

Toepassing

De drie factoren die Pink benoemt, autonomie, zingeving en Meesterschap staan feitelijk ook centraal in het hele Agile werken. Als facilitator moet je er voor zorgen dat:

- de ontwikkelaars zelfstandig hun werk kunnen organiseren en uitvoeren (autonomie)
- de ontwikkelaars zien dat ze direct bijdragen aan een hoger doel en klantwaarde creëren (zingeving)
- de ontwikkelaars kunnen experimenteren en steeds beter worden in wat ze doen (Meesterschap)



Figuur 20. Wat mensen motiveert



Zelf aan de slag?

Als je niet bekend bent met de termen uit Agile, dan is een basiscursus aan te raden. Als je wilt leren om de waarde van organisaties te maximaliseren, dan is de Product Owner de beste keuze. Voor het begeleiden is de Scrum Master de beste keuze. Zelf raad ik aan om kennis te hebben van zowel de Body of Knowledge van de Product Owner als de Scrum Master om zo een compleet beeld te hebben. Verdiepingsopleidingen zitten in het opschalen van Agile en Scrum zoals SAFe of in het implementeren van DevOps.

Op Springest kun je aanbieders vergelijken en ons aanbod vind je op www.springest.nl/5st3ps



Beste Nieuwe Opleider van Nederland



Top 2 Beste Opleider voor Kwaliteitsmanagement