

IV. DOCUMENTS DE LA CAMPAGNE SUR L'AGILITÉ

WATERVAL



Lang, lang geleden (en ook nog een beetje vandaag) werkten we als een **waterval**. Projecten werden opgesplitst in verschillende fases. Elke fase werd volledig afgewerkt voor de volgende fase kon starten.

REQUIREMENTS ANALYSE

DESIGN

CODE

INTEGRATIE

TESTING

UITROL



Een aantal softwareontwikkelaars vond de waterval maar niets.

TAAQA! NIET CREATIEF! BUREAUCRATISCH!

Hun redenering: hoe kan je nu al weten wat in de volgende fase van het project precies nodig zal zijn?

Bij elke stap die je zet loer je bij, weet je beter waar je naartoe moet en hoe je dat moet aanpakken. Dus bedachten de softwareontwikkelaars een andere manier van werken, die ze **agile** noemden.

"Today is the dumbest day of the rest of our project"

Hun principes legden ze vast in het **Manifesto for Agile Software Development**. Daarin verkiezen ze hun manier van werken boven de waterval.

FLEXIBEL

SOFTWARE DIE WERKT

UITGEBREIDE DOCUMENTATIE

SAMENWERKING MET DE OPDRACHTGEVER

ONDERHANDELINGEN OVER HET CONTRACT

MENSEN EN INTERACTIE

PROCESSEN EN TOOLS

INSPLEEN OP VERANDERING

DE UITVOERING VAN EEN PLAN

DE WERELD VERANDERT, BEDRIJVEN VERANDEREN



Maar wat hebben wij nu met **software** te maken...? Agile gaat veel verder dan IT en wordt meer en meer op grote schaal toegepast in grote bedrijven.



37% SNELLERE TIME-TO-MARKET

36% FLEXIBELER OMGAAN MET VERANDERENDE PRIORITEITEN

27% STUGENDE PRODUCTIVITEIT

24% STUGENDE KWALITEIT

24% BETERE AFSTEMMING VAN IT- EN BUSINESSDOELSTELLINGEN

Agile is populair, maar vraagt wel een andere **mindset**



Om die reden is het **niet gemakkelijk** om agile te beginnen werken. Een ommekeer naar agile werken zal de **zwakke punten** in een organisatie blootleggen. Worden die zwakke punten aangepakt, dan kan agile een **verrijking** betekenen voor een bedrijf.

"Agile means cultural, operational and business challenges"

DE GEHEIMEN VAN HET SCRUM TEAM



HET SCRUM TEAM

Agile werken gebeurt in kleine teams, '**scrums**' genaamd. Een scrum telt maar 3 soorten leden:

1. PRODUCT OWNER
2. TEAM
3. SCRUM MASTER

MAAR WIE IS DAN DE BAAS?

Iedereen! En niemand. In een scrum is er ruimte voor iedereen om initiatief te nemen en om natuurlijk leiderschap te ontwikkelen. De verantwoordelijkheden zijn verdeeld over product owner en team, met de scrum master als facilitator.

AGILE

Hoe je kan werken in groep en ook echt iets realiseren

"Power isn't the most powerful type of influence"

ZO GAAT HET IN ZIJN WERK

INTAKE
business bepaalt requirements

BACKLOG WORKSHOP
product owner en team bepalen **epics** (Epics = "Everything we might ever do")

PLANNING
epics worden opgedeeld in kleinere, behapbare user stories die gerangschikt worden volgens **prioriteit**. De user stories worden verdeeld in kleine taken, die **verdeeld** worden onder de team members.

SPRINTS
het team selecteert de user stories met de hoogste prioriteit om in een eerste, **korte tijdsperiode (sprint)** te behandelen. Een sprint duurt hoogstens enkele weken. Elke taak ondergaat planning, analyse, ontwerp, presentatie, testing en beoordeling. Aan het einde van een sprint moet er een **bruikbaar product** op tafel liggen.

DAILY SCRUM

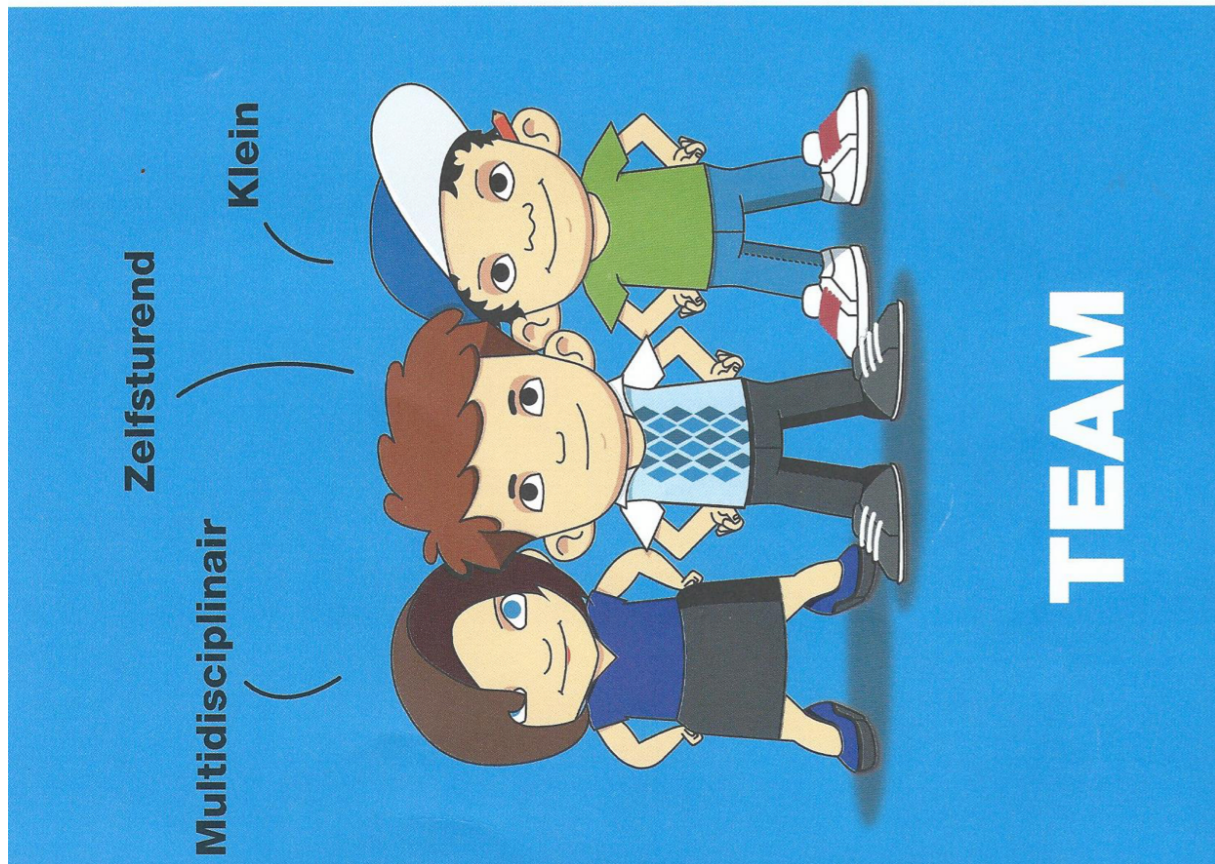
teamleden **rapporteren** aan elkaar en zetten samenwerkingen op. Ze geven aan wat ze gisteren gedaan hebben, wat ze vandaag zullen doen en wat hen mogelijk tegenhoudt.

Bij elke sprint ...
... kan de samenstelling van de **team members** wijzigen ivd de aan te pakken taken
... kunnen er **taken** bijkomen, veranderen of een andere prioriteit krijgen

SPRINT REVIEW

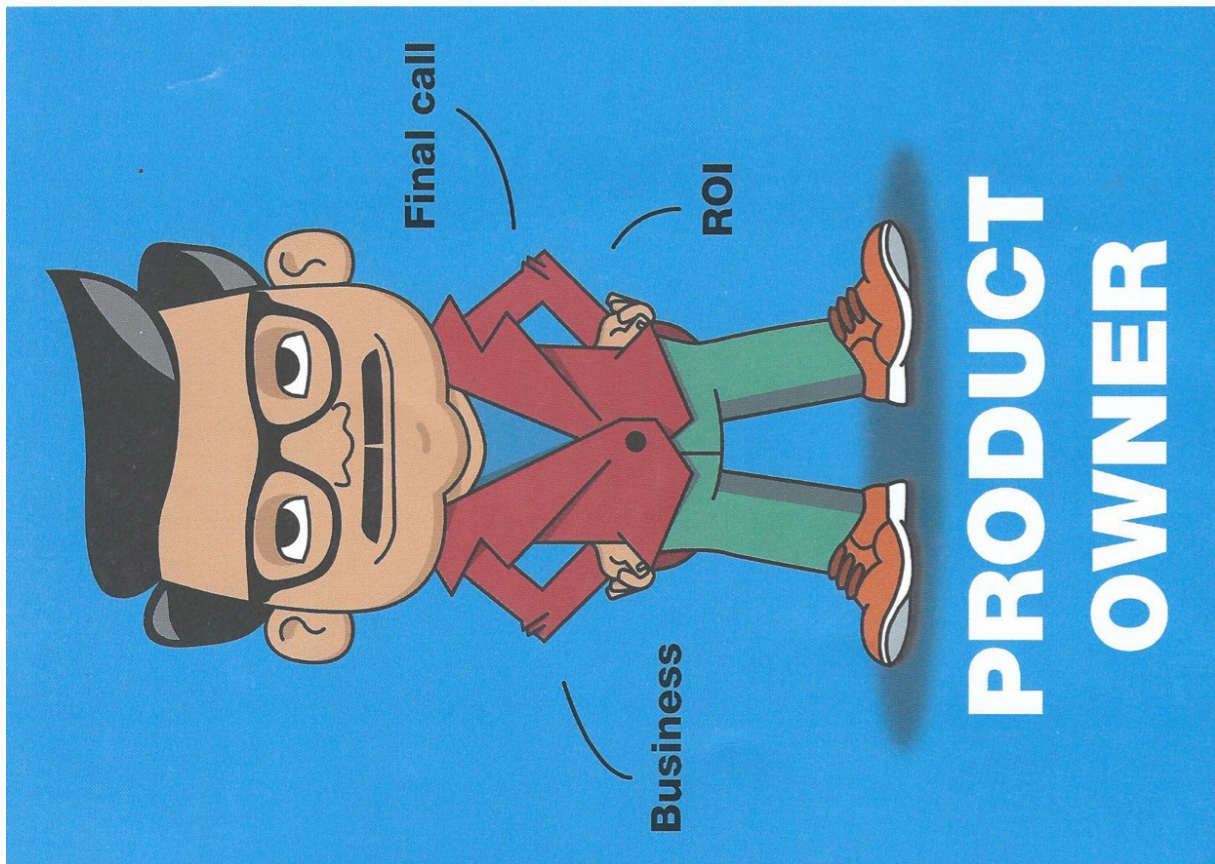
teamleden presenteren gedaan werk aan de product owner en andere stakeholders en verzamelen **feedback**. Product owner bepaalt welk werk 'done' is. Wat niet 'done' is, komt terug op de backlog om later opnieuw op te nemen in een volgende sprint.

RETROSPECTIVE
team verzamelt ideeën ter **verbetering** van het proces



Het **team** creëert oplossingen voor de product owner. Het team bestaat uit verschillende **professionals** die intensief samenwerken aan eenzelfde einddoel. De teamleden rapporteren dagelijks aan elkaar in de **daily stand-ups** en schakelen elkaars hulp in wanneer nodig. Het team **bepaalt zelf** welke taken ze opnemen in welke sprint en welk teamlid welke taken uitvoert.

Met 5 à 9 leden is een scrum team klein. Zelfs voor heel **grote projecten** worden de teams niet groter. In plaats daarvan komen er andere teams bij, die allemaal werken aan delen van het eindproduct. Elk team vaardigt één persoon af die enkel keren per week afstemt met de andere teams en het werk coördineert. Op die manier is het mogelijk dat enkele honderden mensen samenwerken in scrum teams.



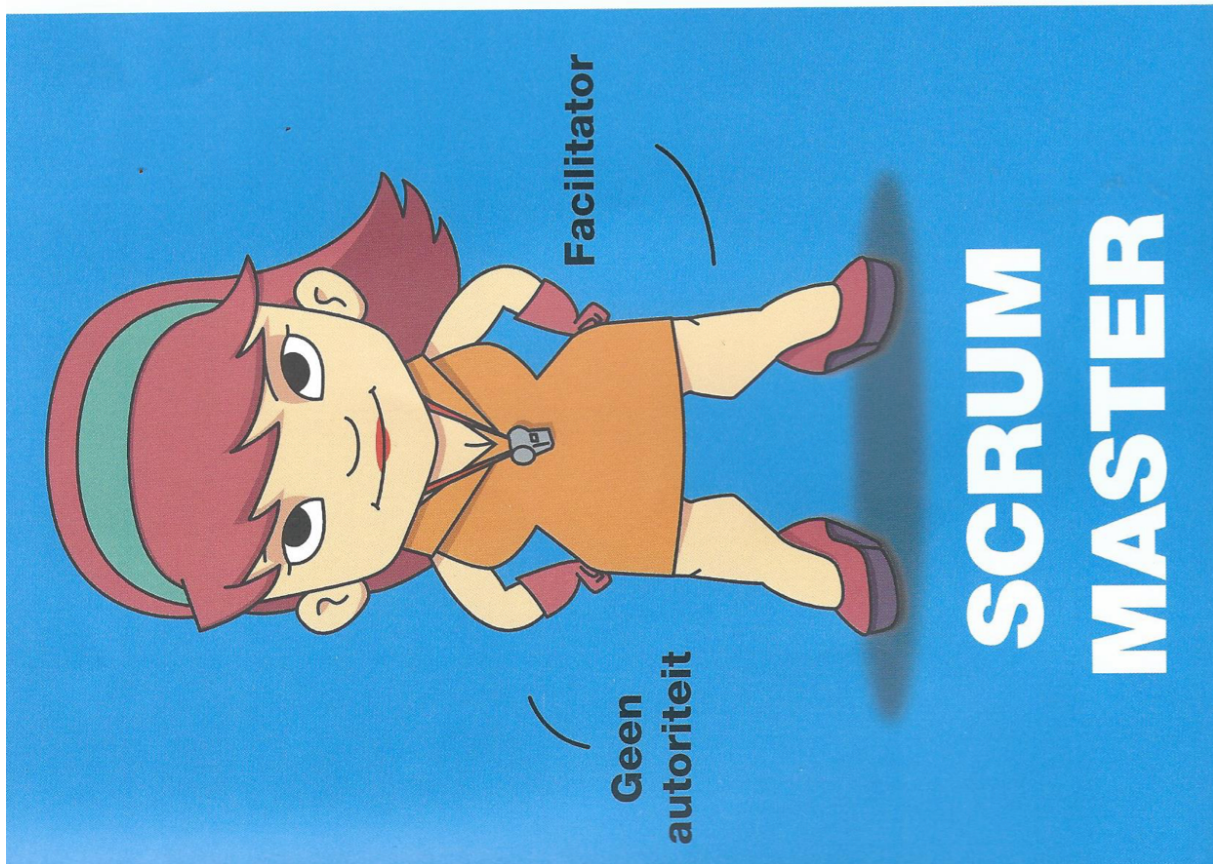
De **product owner** vertegenwoordigt de business en is de belangrijkste stakeholder van het project. Hij of zij is een persoon die heel goed weet wat de business nodig heeft en die deze **visie** kan overbrengen naar het scrum team. Hij is verantwoordelijk voor de **return on investment** van het scrum team en heeft de **final call** over wat de requirements zijn en welke prioriteit ze hebben. Daarom speelt de product owner een cruciale rol tijdens de backlog workshop (requirements bepalen) en tijdens de sprint planning (prioritiseren).

DO'S

- Requirements bepalen
- Prioriteiten aangeven
- Team motiveren
- Actief betrokken bij team, proces en product
- Contact en communicatie met andere stakeholders

DON'TS

- Sprints plannen: het team kan beter dan wie ook inschatten hoeveel werk er nodig is om aan bepaalde requirements te voldoen
- Nieuwe requirements creëren tijdens een sprint: verandering is oké en wordt zelfs aangemoedigd, maar kan enkel tussen twee sprints



De **scrum master** staat ter beschikking van het team. Hij of zij **faciliteert** het werk van de teamleden door hen te beschermen van afleidingsen en onderbrekingen, obstakels te verwijderen, mensen uit te leggen hoe scrum werkt, timings te bewaken, de backlog overzichtelijk te houden enz. Zo zorgt de scrum master ervoor dat het team zijn ware **potentieel** bereikt. De rol van scrum master kan opgenomen worden door een apart persoon, maar ook door een van de teamleden.

DO'S

- Team beschermen
- Team coachen
- Scrum prediken

DON'TS

- Bevelen: de scrum master is een 'dienende leider'. Hij of zij heeft geen autoriteit over het team, maar kan hen wel vooruit helpen door de omstandigheden zo gunstig mogelijk te maken

AGILE LEXICON

Agile

Agile is een flexibele manier van werken: de vraag van de opdrachtgever en de voorgestelde oplossing van het projectteam evolueren voortdurend. Het team is multidisciplinair en stuurt zichzelf. Bedoeling is om al vroeg in het proces werkende oplossingen op te leveren.

De principes van Agile liggen vast in het Agile Manifesto (2001).

Agile Manifesto

Het Agile Manifesto werd in 2001 opgesteld door enkele softwareontwikkelaars die een lichte, flexibele manier van werken zochten.

In het Manifest verkiezen ze

- mensen en interactie boven processen en tools
- software die werkt boven uitgebreide documentatie
- samenwerking met de opdrachtgever boven onderhandelingen over het contract
- kunnen inspelen op verandering boven de uitvoering van een plan

Het onderliggende idee onder deze nieuwe manier van werken is dat je bij de start van een project nooit alle requirements kan kennen of verzamelen. Tijdens het project duiken nieuwe requirements op of blijken vooropgestelde requirements plots niet meer zo belangrijk. Daarom is het belangrijk om alle stakeholders (gebruiker, consument ...) continu te betrekken in het proces en snel antwoord te bieden op veranderingen.

Het Agile Manifesto verankert 12 principes:

1. Klantentevredenheid door snelle oplevering van resultaten
2. Openstaan voor veranderende requirements, zelfs later in het proces
3. Frequent bruikbare producten opleveren (om de 4 weken of sneller)
4. Nauwe, dagelijkse samenwerking tussen business en creatie
5. Projecten zijn opgebouwd rond gemotiveerde, betrouwbare individuen
6. Face-to-face overleg is de beste vorm van communicatie
7. Vooruitgang van het project meten adhv de hoeveelheid werkende producten
8. Projectteam is bereid om voortdurend in ontwikkeling te zijn
9. Nadruk op technische uitmuntendheid en goed design
10. Eenvoud is essentieel
11. Projectteam is zelfsturend
12. Regelmatige aanpassing aan veranderende omstandigheden

Backlog

Een lijst met items die een bedrijf wil realiseren. Items zijn groot en veelomvattend ('epics') of opgedeeld in kleinere, behapbare items ('user stories').

Backlog Refinement

Workshop waarin team en product owner samen de items van de backlog overlopen en eventuele verduidelijken, aanpassen, opnieuw prioriteren ...

Methodes om prioriteiten toe te kennen kunnen zijn:

- Moscow
- Fit/Gap
- Story Points

Daily Stand-up (or Daily Scrum)

Korte teammeeting waarin elk teamlid antwoord geeft op volgende vragen:

- Welk werk heb ik gedaan sinds de laatste meeting?
- Welk werk zal ik doen tegen de volgende meeting?
- Waar loop ik vast? Wiens hulp heb ik nodig?

Definition of Done / Ready

De Definition of Done bepaalt wanneer een user story afgewerkt is. Elk team stelt zijn eigen definitie op. Elk teamlid begrijpt en hanteert de definitie.

Voorbeeld: 'Done' betekent: gecodeerd volgens de standaarden, geïmplementeerd en getest met volledige test automation, geïntegreerd en gedocumenteerd.

De Definition of Ready bepaalt wanneer een user story klaar is om opgenomen te worden in een sprint of wanneer aan alle voorwaarden voldaan is om een sprint te starten.

Epic

Een epic is een grote user story die niet in één sprint kan worden uitgevoerd. Daarom wordt een epic opgedeeld in verschillende user stories.

Voorbeeld: Voor de epic 'Als kandidaat wil ik via een zoekopdracht in Google kunnen zoeken naar vacatures zodat ik meteen een passende job binnen Colruyt Group vind' moeten verschillende taken uitgevoerd worden.

Happiness Metric

Met de Happiness Metric meet je hoe tevreden de teamleden zijn over het verloop van het project.

(Shippable) Increment

Het 'product' dat het scrum team aan het einde van een (of meerdere) sprint(s) oplevert. Het product voldoet aan de vooraf afgesproken 'definition of done' en moet bruikbaar zijn.

Moscow Analyse

Moscow is een methode om prioriteiten toe te kennen aan items uit de backlog. Elk item wordt ingedeeld bij 'must', 'should', 'could' of 'won't'.

M MUST

Item dat absoluut opgenomen moet worden in het eindproduct om dat product een succes te kunnen noemen.

S SHOULD

Item met hoge prioriteit dat, indien mogelijk, opgenomen moet worden in het eindproduct.

C COULD

Item dat wenselijk is maar niet noodzakelijk. Wordt opgenomen in het eindproduct als er tijd en middelen voor zijn.

W WON'T

Item dat volgens de stakeholders niet opgenomen moet worden in het eindproduct, al kunnen ze in de toekomst altijd op deze beslissing terugkomen.

Planning Poker

Planning Poker is een methode om in te schatten hoeveel werk vereist is om een item uit de Product Backlog te realiseren. Elk teamlid geeft dmv een scorekaart aan hoeveel werk een item volgens hem inhoudt. Verschillen in die inschatting worden besproken tot het team een consensus bereikt. Het is vooral de discussie die ervoor zorgt dat iedereen betrokken is.

Portfolio Backlog

High-level backlog met epics die de product owner in de toekomst gerealiseerd wil zien.

Product Backlog

De Product Backlog is een lijst met het nodige werk voor een product. Alle items in de Backlog zijn voorzien van een prioriteit (bv dmv Moscow of Story Points) en van een inschatting qua hoeveelheid werk (bv dmv Planning Poker, Fit/Gap).

Product Backlog Items

Alles wat op een Product Backlog staat is een (product backlog) item. Product Backlog Items hebben een beschrijving, schatting van werkvolume en een prioriteit.

Product Backlog Beheer

Goed Product Backlog Beheer omvat:

- Helder omschrijven van Product Backlog Items
- Ordenen van Product Backlog Items ifv bedrijfsdoelstellingen
- Optimaliseren van de waarde van het werk dat in een sprint gebeurt
- Zorgen voor transparantie, duidelijkheid en zichtbaarheid van de Product Backlog
- Ervoor zorgen dat het team de Product Backlog Items (en hun plaats op de lijst) begrijpt tot op het niveau dat nodig is

Product Owner

De Product Owner is één persoon die de belangen van alle stakeholders vertegenwoordigt. Hij of zij is verantwoordelijk voor het maximaliseren van de waarde die door het scrum team wordt opgeleverd. De Product Owner is verantwoordelijk voor het bijhouden en beheer van de Product Backlog.

Zie ook rolbeschrijvingen.

Release Burn-down chart

Een grafiek die aangeeft hoeveel van een release nog gerealiseerd moet worden. De grafiek loopt af naar een vooraf besproken doel.

Release Burn-up chart

Een grafiek die aangeeft hoeveel van een release al gerealiseerd is. Elke sprint komt er een increment bij waardoor de grafiek groeit en dichterbij het einddoel komt.

Requirements

Alle eisen en wensen die een gebruiker stelt aan een product. Dit kunnen functionele eisen zijn, gebruikers-eisen, business requirements ...

Scrum

Scrum is een methode uit Agile Ontwikkeling om complexe projecten te beheersen. Er wordt gewerkt in multidisciplinaire teams die in korte sprints, van 1 tot 4 weken, een werkend product opleveren. De term Scrum is afkomstig uit de rugbysport waarbij een team samen een doel probeert te bereiken (of in dat geval wedstrijd winnen) waarbij voor elke fase de tactiek besproken wordt in een 'scrum'.

Scrum wordt vaak gebruikt bij (digitale) producten waarvan klant (of eindgebruiker) nog niet helemaal precies weet wat hij wilt en waarbij men doorheen het project de requirements beter beschrijft.

Scrum Board

Het Scrum Board is een bord waarop alle Sprint Backlog Items hangen. Deze zijn verdeeld over 3 kolommen: To Do, In Progress en Done. De belangrijkste items (of onderverdeling naar taken) hangen bovenaan. Dit bord geeft in één oogopslag een status van het project. Elke dag bespreekt het scrum team de status in de daily stand-up.

Scrum Master

De Scrum Master is verantwoordelijk voor het scrumproces. Hij zorgt ervoor dat de scrum op de juiste manier ingezet wordt om deze techniek maximaal te laten renderen. De Scrum Master heeft geen leidende functie, wel een ondersteunende. De rol kan opgenomen worden door een teamlid.

Sprint

Een timebox van 1 tot 4 weken waarin een werkend product (of prototype) wordt opgeleverd dat voldoet aan de 'definition of done'. Alle sprints van een project hebben dezelfde lengte en volgen elkaar (bijna) direct op.

Sprint Planning Meeting

Tijdens de Sprint Planning Meeting bepaalt het team gezamenlijk welk werk uitgevoerd zal worden tijdens de komende sprint.

Sprint Retrospective

Een sprint wordt afgesloten met een 'Retrospective'. Tijdens deze vergadering kijkt het team terug op het werkproces, de samenwerking en kwaliteit van de afgelopen sprint met als doel aan te pakken wat niet goed is verlopen.

Story Points

Story Points zijn, net als Planning Poker, een middel om het 'gewicht' van bepaalde user stories in te schatten. De punten (bv 1, 2, 4, 8 en 16) slaan dus niet op het aantal te presteren werkuren, maar geven eerder aan hoe de te verrichten taken zich tot elkaar verhouden op vlak van werkvolume.

(Scrum) Team

Een Scrum Team is multidisciplinair en zelfsturend. Dit betekent dat het team in staat is alle aspecten van het project (van ontwerp tot en met oplevering) te verzorgen. Het team is zelf verantwoordelijk voor het inplannen en uitvoeren van het werk tijdens een sprint.

Team Velocity

De gemiddelde hoeveelheid werk die het team per sprint kan verzetten. Het gemiddelde komt voort uit voortschrijdend inzicht doorheen verschillende sprints.

User Experience

User Experience (UX) omvat alle aspecten van interacties tussen eindgebruikers en een organisatie. In Agile Development is UX vaak verantwoordelijk voor het in kaart brengen van de requirements.

User Story

Een user story beschrijft de vereiste functionaliteiten van het eindproduct. Uit elke user story kan je afleiden welke stakeholder gediend is met de functionaliteit, wat hij precies verwacht en waarom. User stories zijn opgebouwd volgens een vast stramien: 'Als [type gebruiker] wil ik [iets kunnen] zodat ik [een behoefte kan vervullen]'. Het aanhouden van deze structuur is belangrijk om de 'waarom' van een bepaalde functionaliteit voor een eindgebruiker niet uit het oog te verliezen.