

The Scrum Guide™

Den ultimative guide til Scrum:
Spilletts regler

November 2017



Jeff Sutherland

Ken Schwaber

DANISH

Udviklet og vedligeholdt af Scrum-skaberne: Ken Schwaber og Jeff Sutherland

Indholdsfortegnelse

Formålet med Scrum Guiden	3
Definition af Scrum	3
Anvendelse af Scrum	4
Scrum teori	4
Scrum værdier	5
Scrum Team	6
Product Owner	6
Development Team	7
Scrum Master	7
Scrum begivenheder	9
Sprint	9
Sprint Planning	10
Daily Scrum	12
Sprint Review	13
Sprint Retrospective	14
Scrum artefakter	14
Product Backlog	15
Sprint Backlog	16
Increment	17
Gennemsigtighed af artefakter	17
Definition af "Done"	18
Konklusion	19
Anerkendelser	19
Individer	19
Historie	19
Oversættelse	19
Ændringer mellem	20
2016 og 2017 Scrum Guide	20

Formålet med Scrum Guiden

Scrum er et framework, der kan benyttes i udvikling, levering og fastholdelse af komplekse produkter. Denne guide indeholder definitionen af Scrum. Definitionen består af Scrum's roller, begivenheder, artefakter, og de regler der binder disse sammen. Ken Schwaber og Jeff Sutherland udviklede Scrum. Denne Scrum guide er skrevet af og stillet til rådighed af dem. Sammen, står de bag Scrum guiden.

Definition af Scrum

Scrum (sb): Et framework hvori mennesker kan adressere komplekse problemer, samtidig med at der fokuseres på produktivitet og kreativt leveres produkter med højest mulig værdi.

Scrum er:

- Letvægt
- Nemt at forstå
- Svært at beherske

Scrum er et proces framework, der har været anvendt til at styre arbejdet på komplekse produkter siden starten af 1990'erne. Scrum er ikke en proces, teknik eller en endegyldig metode; men skal mere ses som et framework, hvori man kan anvende forskellige processer og teknikker. Scrum udstiller, hvor det står dårligt til med arbejdsteknikker og virksomhedens evne til at styre produktudvikling, så det er muligt løbende at forbedre produkt, team og de omgivelser man arbejder i.

Scrum frameworket består af Scrum Teams og deres tilhørende roller, begivenheder, artefakter samt regler. Hver komponent i frameworket har et specifikt formål og er essentiel for Scrum's succes og brug.

Reglerne i Scrum sammenknytter roller, begivenheder og artefakter samt styrer relationen og interaktionen mellem disse. Reglerne i Scrum er beskrevet undervejs i dette dokument.

Specifikke taktikker for at anvende Scrum frameworket i forskellige situationer varierer og er beskrevet andetsteds.

Anvendelse af Scrum

Scrum blev oprindeligt designet til styring og udvikling af produkter. Fra begyndelsen af 1990'erne er Scrum blevet anvendt bredt over hele verden til at:

1. Forske i og identifikation af levedygtige markeder, teknologier og produkt egenskaber;
2. Udvikle og forbedre produkter;
3. Frigive produkter og opdateringer så ofte som flere gange om dagen;
4. Udvikle og fastholde Cloud platform (online, sikker, on-demand) og andre driftsmiljøer der anvendes af produkter; og,
5. Opretholde og forny produkter.

Scrum er blevet brugt til at udvikle software, hardware, indlejret software, netværk af interagerende funktioner, autonome køretøjer, skoler, regeringer, marketing, ledelse af organisationernes drift og næsten alt, hvad vi bruger i vores dagligdag, som enkeltpersoner og samfund.

Idet kompleksiteten i teknologi, markeder og miljø og deres interaktioner er steget hurtigt, er Scrums evne til at håndtere kompleksitet påvist dagligt.

Scrum har vist sig særligt effektiv i iterativ og inkrementel videnoverførsel. Scrum anvendes nu i vid udstrækning til produkter, tjenester og forvaltningen af den overordnede organisation.

Essensen af Scrum er et lille team af individer. Det enkelte team er meget fleksibelt og adaptivt. Disse styrker fortsætter med at fungere for både enkle, flere, mange og netværk af teams, der udvikler, frigiver, drifter og opretholder arbejdet for tusindvis af individer. De samarbejder gennem sofistikerede udviklingsarkitekturer og miljøer til.

Når ordet "udvikling" bliver brugt i Scrum Guiden refererer de til kompleks arbejde som de eksempler der er beskrevet herover.

Scrum teori

Scrum er baseret på teorien bag empirisk proceskontrol, eller empiri. Empiri definerer, at viden kommer fra erfaring og at tage beslutninger baseret på hvad der er kendt. Scrum implementerer en iterativ, trinvis tilgang til at optimere forudsigeligheden og at kontrollere risici.

Tre grundstøtter er fundamentet for enhver implementering af empirisk proceskontrol: gennemsigtighed, inspektion og tilpasning.

Gennemsigtighed

Essentielle dele af processen skal være synlig for dem der har ansvaret for at frembringer resultatet. Gennemsigtighed kræver at disse dele overholder en fælles standard, så de der observerer processen, har det samme syn på hvad der observeres.

Eksempel

- Et fælles sprog skal anvendes af alle når man taler om processen; og
- En fælles definition af "Done" skal være delt af de der udfører arbejdet og dem der skal inspicere det udarbejdede produkt inkrement.

Inspektion

Brugere af Scrum skal ofte inspicere Scrum artefakter og fremdriften mod et Sprint Goal for at identificere uønskede afvigelser. Inspektion bør ikke foregå så ofte at det går ud over arbejdet. Inspektioner er mest nyttige, når de gennemføres nøjsomt af erfarne reviewere, i tilknytning til det arbejde der udføres.

Tilpasning

Hvis en reviewer beslutter, at et eller flere aspekter af en proces falder uden for de acceptable grænser, og at det resulterende produkt vil blive uacceptabelt, så skal processen eller det der produceres justeres. En justering skal gennemføres så snart det er muligt, for at forhindre yderligere afvigelser.

Scrum har fire formelle muligheder for inspektion og tilpasning, som beskrevet i afsnittet om *Scrum begivenheder*:

- Sprint Planning
- Daily Scrum
- Sprint Review
- Sprint Retrospective

Scrum værdier

Når værdierne som engagement, mod, fokus, åbenhed og respekt er omfavnet og anvendt af Scrum Teamet, vil Scrum grundpiller som gennemsigtighed, inspektion og tilpasning komme til live og skabe tillid til alle. Scrum Team medlemmer lærer og udforsker disse værdier, når de arbejder med Scrum begivenheder, roller og artefakter.

Succesfuld brug af Scrum afhænger af at folk bliver dygtigere til at efterleve disse fem værdier. Folk forpligter sig personligt til at nå målene for Scrum teamet. De Scrum Team medlemmer har modet til at gøre det rigtige og arbejde på vanskelige problemer. Alle fokuserer på arbejdet i Sprintet og målene for Scrum teamet. Scrum Team og dets interessenter er enige om at være åbne om alt arbejde og udfordringerne med at udføre arbejdet. Scrum Team medlemmer respekterer hinanden for at være dygtige, uafhængige mennesker.

Scrum Team

Scrum Teamet består af en Product Owner, et Development Team og en Scrum Master. Scrum Teams er selvorganiserende og tværfaglige. Selvorganiserende teams vælger, hvordan de bedst kan udføre deres arbejde, snarere end at blive instrueret af andre uden for teamet. Tværfaglige teams besidder alle nødvendige kompetencer, for at de kan udføre arbejdet uden at være afhængige af andre udenfor teamet. Team modellen i Scrum er designet til at optimere fleksibilitet, kreativitet og produktivitet. Scrum Teamet har vist at de er blevet mere effektive indenfor de nævnte anvendelser og alt komplekst arbejde

Scrum Teams leverer et produkt iterativt og trinvist, hvilket maksimerer mulighederne for feedback. Trinvis leverance af et "Done" produkt sikrer, at en potentielt brugbar version af det fungerende produkt altid er tilgængelig.

Product Owner

Product Owner er ansvarlig for at maksimere værdien af produktet som er resultatet af arbejdet i Development Teamet. Hvordan dette gøres, kan variere meget på tværs af organisationer, Scrum Teams og enkeltpersoner.

Product Owner er den eneste person, der er ansvarlig for administration af Product Backloggen. Product Backlog administration omfatter:

- At beskrive Product Backlog items tydeligt.
- Ændring på rækkefølgen af Product Backlog items således, at de bedst opfylder mål og missioner.
- Optimering af værdien af det arbejde Development Teamet udfører.
- Sikring af, at Product Backloggen er synlig, transparent og klar for alle, og samtidig viser, hvad Scrum Teamet arbejder på som det næste; og

Sikring af, at Development Teamet forstår Product Backlog items på det nødvendige niveau.

Product Owner kan selv udføre ovenstående arbejde eller få Development Teamet til at gøre det, men Product Owner forbliver den ansvarlige.

Product Owner er én person, ikke et udvalg. Product Owner kan i Product Backloggen repræsentere ønskerne fra et udvalg, men dem, der ønsker at ændre prioriteten af et Product Backlog item, skal overbevise Product Owner.

For at Product Owner kan få succes, skal hele organisationen respektere hans eller hendes beslutninger. Product Owners beslutninger er synlige i indholdet og rækkefølgen af Product Backloggen. Ingen har ret til at kræve, at Development Teamet skal arbejde ud fra andre krav end Product Backloggen.

Development Team

Development Teamet består af professionelle, der udfører arbejdet ved at levere et Increment af et "Done" produkt der er potentiel klar til frigivelser efter hvert Sprint. Et "Done" Increment er påkrævet til Sprint Review. Det er kun medlemmer af Development Teamet, der bidrager til Incrementet.

Development Teams er struktureret og bemyndiget af organisationen til at organisere og styre deres eget arbejde. Den deraf følgende synergi optimerer Development Teams samlede produktivitet og effektivitet.

Development Teams har følgende karakteristika:

- De er selvorganiserende. Ingen (ikke engang Scrum Master) fortæller Development Teamet, hvordan de omformer Product Backloggen til et Increment der er potentielt klar til frigivelse.
- Development Teams er tværfaglige med alle de færdigheder, som er nødvendige, for at de, som et team, kan levere en Increment af produktet.
- Scrum anerkender ikke andre titler end Developer for medlemmerne i Development Teamet, uanset hvilket arbejde en person udfører.
- Scrum anerkender ikke sub-teams i Development Teamet, uanset at der kan være behov for forskellige domæner fx test, arkitektur, drift eller forretningsanalyse. Der er ingen undtagelser for denne regel. Og,
- Individuelle Development Team medlemmer kan have specielle færdigheder og fokusområde, men ansvaret ligger hos Development Teamet som en samlet enhed.

Development Team størrelse

Optimalt set er Development Teamet ikke større end, at det kan vedblive med at være adræt, og ikke mindre end, at det kan afslutte betydningsfuldt arbejde. Færre end tre Development Team medlemmer kan mindske samspillet og resultere i mindre produktivetsgevinster. Mindre Development Teams kan opleve begrænsninger i forhold til deres færdigheder i løbet af et Sprint, hvilket kan forårsage, at Development Teamet ikke er i stand til at levere et Increment, der er potentielklar til frigivelse. Mere end ni medlemmer kræver for meget koordination. Store Development Teams skaber for megen kompleksitet til at en empirisk proces er brugbar. Product Owner og Scrum Master er ikke medtaget i denne optælling, medmindre de også udfører arbejde i Sprint Backloggen.

Scrum Master

Scrum Master er ansvarlig for at promovere og supportere Scrum som defineret i denne guide. Scrum Master gør dette ved at hjælpe alle med at forstå Scrum teori, praksis, regler og værdier.

Scrum Master er en tjenende leder for Scrum teamet. Scrum Master hjælper dem udenfor Scrum Teamet med at forstå hvilken interaktion med Scrum Teamet, der er nyttige, og hvilke, der ikke er nyttige. Scrum Master hjælper alle med at ændre dette samspil for at optimere den værdi, der skabes af Scrum Teamet.

©2017 Ken Schwaber and Jeff Sutherland. Offered for license under the Attribution Share-Alike license of Creative Commons, accessible at <http://creativecommons.org/licenses/by-sa/4.0/legalcode> and also described in summary form at <http://creativecommons.org/licenses/by-sa/4.0/>. By utilizing this Scrum Guide you acknowledge and agree that you have read and agree to be bound by the terms of the Attribution Share-Alike license of Creative Commons.

Scrum Masters ydelser overfor Product Owner

Scrum Master tjener Product Owner på flere måder, herunder:

- Sikre at mål, scope og produkt domæne er forstået af alle i Scrum Teamet så godt som muligt
- Finder teknikker til effektiv styring af Product Backlog.
- Hjælper Scrum Team til at forstå behovet for at Product Backlog items er klare og præcist formulerede.
- Forståelse af produktplanlægning i et empirisk miljø.
- Sikrer at Product Owner ved hvordan man arrangerer Product Backlog for at opnå maksimal værdi.
- Forståelse og praktisering af agilitet og
- Faciliterer ønskede eller nødvendige Scrum begivenheder.

Scrum Masters ydelser overfor Development Team

Scrum Master tjener Development Team på flere måder, herunder:

- Coaching af Development Teamet i selvorganisering og tværfaglighed.
- Hjælper Development Teamet mod at skabe produkter af høj værdi.
- Fjerner forhindringer for Development Teamets fremdrift.
- Faciliterer ønskede eller nødvendige Scrum begivenheder, og
- Coacher Development Teamet i organisatoriske miljøer, hvor Scrum endnu ikke er fuldt implementeret og forstået.

Scrum Masters ydelser overfor organisationen

Scrum Master tjener organisationen på flere måder, herunder:

- Leder og coacher organisationen i tilpasningen til Scrum.
- Planlægning af Scrum implementeringer i organisationen.
- Hjælpe medarbejdere og interessenter med at forstå og indføre Scrum og empirisk produktudvikling.
- Forårsager ændringer, der øger produktiviteten af Scrum Teamet, og
- Arbejder med andre Scrum Masters for at forøge effektiviteten af anvendelsen af Scrum i organisationen.

Scrum begivenheder

Faste begivenheder anvendes i Scrum for at skabe regelmæssighed og for at minimere behovet for møder, som ikke er defineret i Scrum. Scrum bruger tidsbegrænsede begivenheder, således at enhver begivenhed har en maksimal varighed. Når et Sprint er begyndt er længden fastlagt og kan ikke afkortes eller forlænges. De resterende begivenheder kan afsluttes når deres formål er opnået. Dette sikrer, at en passende mængde tid bliver brugt uden, at der tillades spildtid i processen.

Udover selve Sprintet selv, som er en beholder for alle andre begivenheder, er enhver begivenhed i Scrum en formel mulighed for at inspicere og tilpasse noget. Disse begivenheder er specielt designet til at give kritisk åbenhed og inspektion. Undladelse af nogen af disse begivenheder resulterer i reduceret gennemsigtighed og er en tabt mulighed for at inspicere og tilpasse.

Sprint

Hjertet i Scrum er Sprintet, en tidsafgrænset periode på en måned eller mindre, hvor en "Done", brugbar del af produkt Incrementet bliver udarbejdet og er potentiel klar til at blive frigivet. Sprints har en konsistent varighed igennem et udviklingsforløb. Et nyt Sprint starter umiddelbart efter afslutningen af det foregående Sprint.

Sprint indeholder og består af Sprint Planning, Daily Scrums, selve udviklingsarbejdet, Sprint Review og Sprint Retrospective.

Under Sprintet:

- Må der ikke foretages nogen ændringer, som kan bringe Sprintet Goal i fare.
- Må kvalitetsmålene ikke mindskes, og
- Scope kan præciseres og genforhandles mellem Product Owner og Development Team, når man lærer nyt.

Hvert Sprint kan betragtes som et projekt af maksimalt en måneds varighed. Ligesom projekter bliver Sprints brugt til at udrette noget. Hver Sprint har en mål for, hvad der skal bygges, et design og en fleksibel plan, der vejleder om, hvordan man bygger det, selve udviklingsarbejdet og det resulterende produkt Increment.

Sprints er begrænsede til en kalendermåned. Når et Sprints varighed er for langt, kan definitionen af, hvad der bliver bygget, ændre sig, kompleksiteten kan stige, og risikoen kan øges. Sprints muliggør forudsigelighed ved at sikre kontrol og tilpasning af fremskridt i retning af et mål mindst hver kalendermåned. Sprints begrænser også de økonomiske risici til en kalendermåned.

Afbrydelse af et Sprint

Et Sprint kan afbrydes før tid. Kun Product Owner har myndighed til at afbryde et Sprint, selv om han eller hun kan være under indflydelse fra interessenter, Development Team eller Scrum Master.

Et Sprint bør afbrydes, hvis Sprint Goal er forældet. Dette kan forekomme, hvis virksomheden skifter retning eller, hvis markedet eller teknologien ændrer sig. Generelt bør et sprint afbrydes, hvis det pga. omstændigheder ikke længere giver mening. Men, pga. den korte varighed af sprints, giver en afbrydelse sjældent mening.

Når et Sprint bliver afbrudt, skal alle afsluttede og "Done" Product Backlog items reviews. Hvis en del af arbejdet er potentielt klar til at blive frigivet, vil Product Owner typisk acceptere det. Alle ufuldstændige Product Backlog items bliver restestimeret og tilføjet Product Backloggen igen. Det udførte arbejde på disse forgår hurtigt og må ofte reestimeres.

Afbrydelse af et Sprint koster ressourcer, da alle deltager i Sprint Planning for at starte et nyt Sprint. Sprint afbrydelser er ofte traumatisk for Scrum Teamet og er meget usædvanlige.

Sprint Planning

Det arbejde, der skal udføres i Sprint, er planlagt på Sprint Planning. Planen bliver skabt igennem et samarbejde mellem hele Scrum Teamet.

Sprint Planning er tidsbegrænset til otte timer for et Sprint på en måned. For kortere Sprints er begivenheden sædvanligvis kortere. Scrum Master sørger for at begivenheden afholdes og at deltagerne forstår dets formål. Scrum Master lærer Scrum Teamet at afvikle begivenheden indenfor tidsbegrænsningen.

Sprint Planning giver svar på følgende:

- Hvad kan leveres i Incrementet som følge af det forestående Sprint?
- Hvordan vil det arbejde, der skal udføres for at levere et Increment, blive udført?

Første del: Hvad kan leveres i dette Sprint?

Development Teamet arbejder med at forudsige hvilken funktionalitet, der vil blive udviklet i løbet af Sprintet. Product Owner drøfter det mål som Sprintet skal opnå og de Product Backlog items der skal leveres for at opnå målet. Hele Scrum Teamet samarbejder om at forstå det arbejde der skal udvikles i Sprintet.

Input til dette møde er Product Backloggen, seneste produkt Increment, Development Teamets forventede kapacitet i løbet af Sprintet og Development Teamets tidligere performance. Antallet af items, der bliver udvalgt fra Product Backloggen til det kommende Sprint, bestemmes alene af Development Teamet. Kun Development Teamet kan vurdere, hvad det kan udrette i løbet af det kommende Sprint.

Under Sprint Planning udarbejder Scrum Teamet endvidere et Sprint Goal. Sprint Goal er en målsætning, som vil blive opfyldt indenfor Sprintet igennem implementeringen af Product Backlog, og det giver vejledning til Development Teamet om, hvorfor de bygger dette Increment.

Anden del: Hvordan vil det valgte arbejde blive færdiggjort?

Efter at have fastlagt Sprint Goal og valgt hvilke Product Items, der skal medtages i Sprintet, beslutter Development Teamet, hvordan det vil levere dette som "Done" produkt Increment i Sprintet. De Product Backlog items, der er valgt i Sprintet, samt planen for hvordan man leverer, bliver kaldt Sprint Backlog.

Development Teamet starter som regel med at designe systemet og det arbejde, der kræves for at omforme Product Backlog til en fungerende produkt Increment. Arbejdet kan være af varierende størrelse eller estimeret indsats. Men der planlægges tilstrækkeligt i Sprint Planning til, at Development Teamet kan forudse, at de kan klare det i kommende Sprint. Det arbejde, der er planlagt af Development Teamet til de første dage af Sprintet, nedbrydes, inden udgangen af dette møde, til enheder ofte med en varighed på en dag eller mindre. Development Teamet organiserer selv, hvordan de udfører arbejdet i Sprint Backlog, både i løbet af Sprint Planning og efter behov igennem hele Sprintet.

Product Owner kan hjælpe med at præcisere de udvalgte Product Backlog Items og hjælpe med afvejninger. Hvis Development Teamet fastslår, at det har for meget eller for lidt arbejde, kan det genforhandle Sprint Backlog med Product Owner. Development Teamet kan også vælge at invitere andre mennesker til at deltage for at give teknisk eller domænespecifik vejledning.

Ved udgangen af Sprint Planning skal Development Teamet være i stand til at forklare Product Owner og Scrum Master, hvordan man agter at arbejde som et selvorganiserende team for at opnå Sprint Goal og udvikle det forventede produkt Increment.

Sprint Goal

Sprint Goal er et mål for Sprintet, der kan opnås gennem implementering af Product Backlog. Det giver vejledning til Development Teamet om, hvorfor de udvikler produkt Incrementet. Målet bliver fastlagt ved Sprint Planning. Sprint Goal giver Development Teamet nogen fleksibilitet med hensyn til den funktionalitet der skal leveres i Sprintet. De valgte Product Backlog items giver en sammenhængende funktionalitet, som kan være Sprint Goal. Sprint Goal kan være enhver anden sammenhæng der får Development Teamet til at arbejde sammen i stedet for på separate initiativer uden fælles mål.

Mens Development Teamet arbejder, holder det Sprint Goal for øje. For at opfylde Sprint Goal implementeres funktionalitet og teknologi. Hvis arbejdet viser sig at være anderledes end Development Teamet havde forventet, så samarbejder det med Product Owner om at forhandle omfanget af Sprint Backlog, der skal implementeres inden for Sprint.

Daily Scrum

Daily Scrum er en begivenhed, der er tidsbegrænset til 15 minutter. Daily Scrum afholdes hver dag i Sprintet. Her planlægger Development Teamet arbejdet der skal udføres i de næste 24 timer. Dette optimerer team samarbejde og effektivitet ved at inspicere arbejdet siden seneste Daily Scrum og forudsige det fremtidige Sprint arbejde. Daily Scrum afholdes på samme tid og sted hver dag for at reducere kompleksiteten.

Development Teamet bruger Daily Scrum til at vurdere fremskridt mod Sprint Goal og inspicere, hvordan fremdriftens tendens er mod færdiggørelsen af arbejdet i Sprint Backloggen. Daily Scrums optimerer sandsynligheden for, at Development Teamet når Sprint Goalet. Development Teamet bør, hver dag, være i stand til at forklare Product Owner og Scrum Master, hvordan de agter at arbejde sammen som et selvorganiserende team for at nå Sprint Goal og skabe den forventede produkt Increment i den resterende del af Sprintet.

Strukturen af mødet fastlægges af Development Teamet og kan varetages på forskellige måder hvis det fokuserer på fremdrift mod at nå Sprint Goal. I nogle Development Teams anvendes spørgsmål, andre anvender mere diskussionspræget dialog. Herunder er eksempler på hvad der kan anvendes:

- Hvad bidrog jeg med i går der hjalp Development Teamet med at opnå Sprint Goal?
- Hvad vil jeg bidrage med i dag så Development Teamet kan nå Sprint Goal?
- Ser jeg nogen udfordringer, der forhindre mig eller Development Teamet i at nå Sprint Goal?

Development Teamet eller medlemmer i teamet mødes ofte umiddelbart efter Daily Scrum for diskutere detaljer, eller for at tilpasse eller genplanlægge resten af Sprintets arbejde.

Scrum Master sikrer, at Development Teamet holder mødet, men Development Teamet er ansvarlig for gennemførelsen af Daily Scrum. Scrum Master lærer Development Teamet at holde Daily Scrum inden for tidsbegrænsningen på 15 minutter.

Daily Scrum er et internt møde for Development Teamet. Hvis andre deltager sikrer Scrum Master at de ikke forstyrrer mødet.

Daily Scrums forbedrer kommunikation, eliminerer andre møder, identificerer og fjerner forhindringer for fremdriften, fremhæver og fremmer en hurtig beslutningsproces og forbedrer Development Teamets vidensniveau. Dette er et vigtigt møde ift. inspektion og tilpasning.

Sprint Review

Et Sprint Review afholdes ved slutningen af hvert Sprint for at inspicere Incrementet og for at tilpasse Product Backlog - hvis nødvendigt. Under Sprint Review, vil Scrum Teamet og interessenter samarbejde om at vurdere, hvad der er afsluttet i Sprintet. Ud fra dette, og ud fra evt. ændringer til Product Backlog undervejs i Sprintet, vil deltagerne samarbejde om den næste funktionalitet som kan blive færdig. Dette er et uformelt møde, ikke et statusmøde og præsentationen af Incrementet har til hensigt at give umiddelbar feedback og tilskynde til samarbejde.

Sprint Review er maksimalt et fire timers møde for et Sprint på en måned. For kortere Sprints er begivenheden sædvanligvis kortere. Scrum Master sørger for at begivenheden afholdes og at deltagerne forstår dets formål. Scrum Master lærer alle deltagere at overholde begivenhedens tidsbegrænsning.

Sprint Review inkluderer følgende elementer:

- Deltagere inkluderer Scrum Teamet og nøgle interessenter inviteret af Product Owner.
- Product Owner forklarer hvilke Product Backlog items der er "Done" og hvilke der ikke er "Done".
- Development Teamet drøfter, hvad der er gået godt i Sprintet, hvilket problemer der er opstået og hvordan disse problemer er blevet løst.
- Development Teamet præsenterer "Done" funktionalitet og svarer på spørgsmål om Incrementet.
- Product Owner drøfter Product Backlog som den nu ser ud. Han eller hun forudsiger sandsynlige mål og release datoer ud fra fremdrift indtil nu. Han eller hun forudsiger mulige datoer for færdiggørelse baseret på fremdriften indtil videre – om nødvendigt.
- Alle drøfter hvad der skal foregå herefter, så Sprint Review giver værdifuld input til efterfølgende Sprint Planning.
- Review af, hvordan markedet eller potentiel brug af produktet kan have ændret, hvad der er det mest værdifuldt at udvikle herefter, og
- Review af tidsplan, budget, potentielle egenskaber og markedsituation for de næste forventede versioner af funktionalitet eller egenskaber af produktet.

Resultatet af Sprint Review er en revideret Product Backlog, der viser de sandsynlige Product Backlog items for næste Sprint. Product Backlog kan også blive revideret for at tage højde for nye forretningsmuligheder.

Sprint Retrospective

Sprint Retrospective er en mulighed for Scrum Teamet til at inspicere sig selv og udarbejde en plan for forbedringer der kan effektueres i løbet af det næste Sprint.

Sprint Retrospective forekommer efter Sprint Review og inden næste Sprint Planning. Sprint Retrospective er maksimalt et tre timers møde for et Sprint på en måned. For kortere Sprints er begivenheden sædvanligvis kortere. Scrum Master sørger for at begivenheden afholdes og at deltagerne forstår dets formål.

Scrum Master sikrer at møde er positivt og produktivt. Scrum Master lærer Scrum Teamet at afvikle begivenheden indenfor tidsbegrænsningen. Scrum Master deltager som ligeværdigt medlem i mødet ift. ansvarlighed overfor Scrum processen.

Formålet med Sprint Retrospective er at:

- Inspicerer hvordan Sprintet er gået i forhold til personer, relationer, proces og værktøjer.
- Identificere og ordne de større ting der er gået godt samt mulige forbedringer; og
- Udarbejde en plan for implementering af forbedringer til hvordan Scrum Teamet arbejder.

Scrum Master opfordrer Scrum Teamet til, indenfor Scrum proces frameworket, at forbedre deres udviklingsproces og praktikker for at gøre dem mere effektive og glade i næste Sprint. Gennem hver Sprint Retrospective planlægger Scrum Teamet måder at øge produktkvaliteten på at forbedre arbejdsprocesser eller ved at tilpasse definition af "Done" hvis der er behov for det og dette ikke konflikter med produkt eller organisatoriske standarder.

Ved slutningen af Sprint Retrospective bør Scrum Teamet have identificeret de forbedringer de vil implementere i næste Sprint. Implementeringen af disse forbedringer i næste Sprint er Scrum Teamets tilpasning i forhold til deres egen inspicering. Selv om forbedringer kan implementeres på ethvert tidspunkt så giver Sprint Retrospective den formelle mulighed for at fokusere på inspicering og tilpasning.

Scrum artefakter

Scrum's artefakter repræsenterer arbejde eller værdi som på forskellig måde er værdifuld for at give gennemsigtighed og mulighed for inspicering og tilpasning. Scrum artefakter er specielt designet til at maksimere gennemsigtighed af de nøgleinformationer der er nødvendige for at sikre at alle har den samme opfattelse af artefakter.

Product Backlog

Product Backlog er en ordnet liste af alt der forventes at være nødvendig for produktet. Det er den eneste kilde til krav for alle ændringer af produktet. Product Owner er ansvarlig for Product Backlog, herunder indhold, tilgængelighed og rækkefølge.

En Product Backlog er aldrig færdig. Den tidligste form af Product Backlog indeholder den indledende viden og de bedst forståede krav på dette tidspunkt. Product Backlog udvikler sig over tid i takt med at produktet og omgivelserne, hvori den anvendes, forandre sig. Product Backlog er dynamisk; den forandre sig konstant for at repræsentere et produkt, der er passende, konkurrencedygtigt og nyttigt. En Product Backlog eksisterer når produktet eksisterer.

Product Backlog lister alle funktioner, krav, udvidelser, og fejlrettelser som udgør de ændringer, der skal implementeres i produktet i fremtidige releases. Product Backlog items indeholder beskrivelse, rækkefølge, estimat og værdi. Product Backlog items indeholder ofte en test beskrivelse der skal kunne gennemføres med succes når item er Done.

Når et produkt anvendes og opnår værdi, og omgivelserne giver feedback så bliver Product Backlog en større og mere udtømmende liste. Krav vil altid forandre sig og Product Backlog er derfor et levende artefakt. Ændringer i forretningsmæssig krav, marked eller teknologi kan forårsage ændringer i Product Backlog.

Multiple Scrum Teams arbejder ofte sammen om et fælles produkt. Én Product Backlog benyttes til at beskrive det forventede arbejde på produktet. En Product Backlog attribut kan anvendes til at grupperer items.

Product Backlog detaljering er den aktivitet hvor detaljer tilføjes og hvor estimer og rækkefølge bliver tildelt hvert item i Product Backlog. Dette er en løbende proces hvor Product Owner og Development Teamet samarbejder om detaljerne i Product Backlog items. Under Product Backlog detaljering reviews og revideres Product Backlog items. Scrum Teamet bestemmer hvornår og hvordan detaljering foregår. Detaljering tager sædvanligvis ikke mere end 10% af Development Teamets kapacitet. Product Backlog items kan dog til enhver tid opdateres af Product Owner eller efter aftale med Product Owner.

Product Backlog items, der er højere rangerede, er mere klare og mere detaljeret end de der er lavere rangerede. Mere præcise estimer udarbejdes på baggrund af større klarhed og øgede detaljer; jo lavere rangeret, jo mindre detalje. Product Backlog items, der skal behandles af Development Teamet i det næste Sprint er detaljeret, idet de er opdelt, så hver enkelt del forventes at kunne blive "Done" indenfor Sprintets tidsmæssige afgrænsning. Disse Product Backlog items anses for at være "Ready" til udvælgelse i Sprint Planning. Product Backlog items opnår sædvanligvis denne form for gennemsigtighed i løbet af detaljeringsprocessen.

Development Teamet er ansvarlig for alle estimater. Product Owner kan påvirke Development Teamet ved at hjælpe dem med at forstå og prioritere mellem afvejninger. Med de personer, der skal udføre det konkrete arbejde, er dem der giver de endelige estimater.

Monitorering af fremdrift mod mål

På et hvilket som helst tidspunkt er det muligt at opgøre, hvad der mangler for at nå et mål. Product Owner monitorerer denne totale mængde af arbejde mindst ved hvert Sprint Review. Product Owner sammenligner dette med mængden af resterende arbejde ved forrige Sprints afslutning, for at vurdere fremdriften mod at afslutte det planlagte arbejde til den planlagte tid og dermed mod at opnå målet. Denne information gøres synlig for alle interessenter.

Forskellige praktikker til at forudsige trends og fremdrift har været anvende fx burn-downs, burn-ups, eller cumulative flows. Disse praktikker har vist sig brugbare, men kan ikke erstatte vigtigheden af empiri. I komplekse miljøer er det, der kommer til at foregå, ukendt. Kun hvad der allerede har hændt, kan anvendes til fremadrettet beslutningstagning.

Sprint Backlog

Sprint Backlog består af de dele af Product Backlog items som er udvalgt til Sprintet, samt en plan for at levere funktionalitet og opnå Sprint Goal. Sprint Backlog indeholder Development Teamets forventning til, hvilken funktionalitet der vil kunne leveres i det næste Sprint og det nødvendige arbejde, der skal til for at levere denne funktionalitet som et "Done" Increment.

Sprint Backlog visualiserer al det arbejde som Development Teamet identificerer som nødvendigt for at opnå Sprint Goal. For at sikre løbende forbedringer indeholder det mindst en høj prioritet proces forbedring identificeret ved det seneste Retrospective møde.

Sprint Backlog er en plan med tilstrækkelig detaljer til, at ændringer i fremdrift kan forstås på Daily Scrum. Development Teamet modificerer Sprint Backlog gennem hele Sprintet, og Sprint Backlog opstår under Sprintet. Denne opståen forekommer når Development Teamet arbejder sig gennem planen og får større viden om det arbejde, der er krævet for at nå Sprint Goal.

Når nyt arbejde bliver nødvendigt tilføjes det til Sprint Backlog af Development Teamet. Når arbejde er udført eller færdiggjort opdateres estimatet for det tilbageværende arbejde. Når delelementer i planen anses for unødvendige så slettes disse fra planen. Det er alene Development Teamet der kan ændre deres Sprint Backlog i løbet af et Sprint. Sprint Backlog er et ekstremt synlig, realtidsbillede af det arbejde som Development Teamet planlægger at gennemføre i løbet af Sprintet, og det tilhører udelukkende Development Teamet.

Monitorering af fremdrift i Sprintet

På et hvilket som helst tidspunkt i Sprintet er det muligt at opgøre, hvad der mangler at blive udført fra Sprint Backloggen. Development Teamet monitorerer denne totale mængde af arbejde mindst ved hvert Daily Scrum for at vurdere sandsynligheden for, at kunne nå Sprint Goal. Ved at monitorere det resterende arbejde løbende gennem Sprintet kan Development Teamet styre dets fremdrift.

Increment

Et Increment er summen af alle de Product Backlog items som er afsluttet i løbet af Sprintet og værdien af Incrementer af alle tidligere Sprint. Ved slutningen af et Sprint skal det nye Increment være i tilstanden "Done", hvilket betyder at det er i en brugbar stand og opfylder Scrum Teamets definition af "Done". Et Increment er en samling af inspiserbart færdigt arbejde som understøtter empirisme ved slutningen af Sprintet. Incrementet er et skridt på vej mod visionen eller målet. Incrementet skal være brugbart uanset om Product Owner beslutter sig for at frigive det til brugerne.

Gennemsigtighed af artefakter

Scrum bygger på gennemsigtighed. Beslutninger om at optimere værdi og kontrollere risici baseres på den oplevede tilstand af artefakter. Når gennemsigtighed er komplet tages beslutninger på en sundt grundlag. Når gennemsigtighed er ukomplet kan beslutninger være fejlagtige, værdi kan forringes og risici øges.

Scrum Master skal arbejde sammen med Product Owner, Development Team og andre involverede for at forstå om artefakter er totalt gennemsigtige. Der findes praktikker der håndterer ukomplet gennemsigtighed. Hvis komplet gennemsigtighed ikke er til stede skal Scrum Master hjælpe alle med at benytte de mest anvendelige praktikker til at forbedre gennemsigtighed. En Scrum Master kan identificere ukomplet gennemsigtighed ved at inspicere artefakter, identificere mønstre, lytte indgående til hvad der siges og ved at opdage forskelle mellem det forventede og de reelle resultat.

Scrum Masters opgave er at arbejde sammen med Scrum Team og organisationen for at øge gennemsigtighed af artefakter. Dette arbejde består sædvanligvis i læring, overtalelser og forandringer. Gennemsigtighed kommer ikke i løbet af en nat – det er en rejse.

Definition af "Done"

Når et Product Backlog item eller et Increment beskrives som "Done", er det nødvendigt at alle har den samme opfattelse af hvad dette betyder. "Done" kan variere væsentligt fra Scrum Team til Scrum Team. For at sikre gennemsigtighed må deltagerne i Scrum Teamet derfor have en fælles forståelse af, hvad det betyder at arbejdet er færdigt. Dette betegnes som definition af "Done" for Scrum Teamet og benyttes til at vurdere hvornår arbejdet er færdigt på det enkelte produkt Increment.

Den samme definition guider Development Teamet i at vide, hvor mange Product Backlog items Teamet kan vælge til Sprint Planning. Formålet med hvert Sprint er at levere Increments af produktet, der er potentiel klar til at blive frigivet og som opfylder Scrum Teamets nuværende definition af "Done."

Development Teamet leverer et Increment af produktet i hvert Sprint. Disse Increments er brugbare. Derfor kan en Product Owner vælge at frigive funktionaliteten med det samme. Hvis definitionen af "Done" for et Increment er med i konventioner, standarder eller guidelines for udviklingsorganisationen skal alle Scrum Teams som minimum følge denne definition.

Hvis "Done" for et Increment **ikke** er en konvention for en udviklingsorganisation skal Development Teamet af et Scrum Team vedtage en definition af "Done" der er passende for produktet. Hvis der er flere Scrum Teams der arbejder på samme løsning eller produkt release skal Development Teams fra alle Scrum Teams i fællesskab udarbejde en definition af "Done".

Ethvert Increment lægger funktionalitet oveni tidligere Increments, og er nøje gennemtestet, således at det sikres at alle Increments virker sammen.

I takt med at Scrum Teamet bliver mere modent forventes det at deres kriterier for definition af "Done" bliver mere og mere fokuseret på at levere højere kvalitet. Når en ny definition tages i brug kan den afsløre arbejde der skal udføres for tidligere Increments der var "Done". Ethvert produkt eller system bør have en definition af "Done" som er den standard for det arbejde der udføres.

Konklusion

Scrum er frit og stilles til rådighed i denne guide. Scrum's roller, begivenheder, artefakter og regler er obligatoriske. Selvom det er muligt at implementere dele af Scrum, så er resultatet ikke Scrum. Scrum eksisterer kun i sin helhed og fungerer godt som en container for andre teknikker, metoder og praktikker.

Anerkendelser

Individer

Af de tusinder af individer der har bidraget til Scrum, vil vi gerne fremhæve de der var essentielle ved begyndelsen. Jeff Sutherland, arbejdede med Jeff McKenna og John Scumniotales, og Ken Schwaber, arbejdede med Mike Smith og Chris Martin og de arbejdede alle sammen. Mange andre har bidraget i de følgende år og uden deres hjælp ville Scrum ikke have nået så langt som det er i dag.

Historie

Ken Schwaber og Jeff Sutherland arbejdede på Scrum indtil 1995 hvor de co-præsenterede Scrum på OOPSLA konferencen i 1995. Denne præsentation dokumenterede, den erfaring som Ken og Jeff havde opnået igennem årene forinden og offentliggjorde derved den første formelle definition af Scrum.

Scrum's historie er beskrevet andetsteds. For at anerkende de første steder hvor Scrum blev afprøvet og finjusteret, nævner vi her Individual, Inc., Newspaper, Fidelity Investments, og IDX (nu GE Medical).

Scrum guiden dokumenterer Scrum, som er udviklet og vedligeholdt gennem mere end 20 år af Jeff Sutherland og Ken Schwaber. Andre kilder beskriver mønstre, processer, og indsigt der komplementerer Scrum frameworket. Disse kan optimere produktivitet, værdi, kreativitet og tilfredsstillelse ved resultatet.

Oversættelse

Denne guide er blevet oversat fra den originale engelske version, som er stillet til rådighed af forfatterne nævnt herover. Bidragsydere til oversættelsen inkluderer Stig Efsen.

Oversætters navn: Stig Efsen

Primær email: stig.efsen@gmail.com

LinkedIn: <https://www.linkedin.com/in/efsen/>

Ændringer mellem 2016 og 2017 Scrum Guide

1. Tilføjet sektion: Anvendelse af Scrum

Scrum blev oprindeligt designet til styring og udvikling af produkter. Fra begyndelsen af 1990'erne er Scrum blevet anvendt bredt over hele verden til at:

1. Forske i og identifikation af levedygtige markeder, teknologier og produkt egenskaber;
2. Udvikle og forbedre produkter;
3. Frigive produkter og opdateringer så ofte som flere gange om dagen;
4. Udvikle og fastholde Cloud platform (online, sikker, on-demand) og andre driftsmiljøer der anvendes af produkter; og,
5. Opretholde og forny produkter.

Scrum er blevet brugt til at udvikle software, hardware, indlejret software, netværk af interagerende funktioner, autonome køretøjer, skoler, regeringer, marketing, ledelse af organisationernes drift og næsten alt, hvad vi bruger i vores dagligdag, som enkeltpersoner og samfund.

Idet kompleksiteten i teknologi, markeder og miljø og deres interaktioner er steget hurtigt, er Scrums evne til at håndtere kompleksitet påvist dagligt.

Scrum har vist sig særligt effektiv i iterativ og inkrementel vidensoverførsel. Scrum anvendes nu i vid udstrækning til produkter, tjenester og forvaltningen af den overordnede organisation.

Essensen af Scrum er et lille team af individer. Det enkelte team er meget fleksibelt og adaptivt. Disse styrker fortsætter med at fungere for både enkle, flere, mange og netværk af teams, der udvikler, frigiver, drifter og opretholder arbejdet for tusindvis af individer. De samarbejder gennem sofistikerede udviklingsarkitekturer og miljøer til.

Når ordet "udvikling" bliver brugt i Scrum Guiden refererer de til kompleks arbejde som de eksempler der er beskrevet herover.

2. Ændret formulering i sektion: Scrum Master, for at give mere klarhed om rollen

Scrum Master er ansvarlig for at promovere og supportere Scrum som defineret i denne guide. Scrum Master gør dette ved at hjælpe alle med at forstå Scrum teori, praksis, regler og værdier.

Scrum Master er en tjenende leder for Scrum teamet. Scrum Master hjælper dem udenfor Scrum Teamet med at forstå hvilken interaktion med Scrum Teamet, der er nyttige, og hvilke, der ikke er nyttige. Scrum Master hjælper alle med at ændre dette samspil for at optimere den værdi, der skabes af Scrum Teamet.

3. Tilføjet følgende til: Scrum Master ydelser overfor Product Owner

Sikre at mål, scope og produkt domæne er forstået af alle i Scrum Teamet så godt som muligt

4. Opdateret første afsnit i sektion: Daily Scrum, til:

Daily Scrum er en begivenhed, der er tidsbegrænset til 15 minutter. Daily Scrum afholdes hver dag i Sprintet. Her planlægger Development Teamet arbejdet der skal udføres i de næste 24 timer. Dette optimerer team samarbejde og effektivitet ved at inspicere arbejdet siden seneste Daily Scrum og forudsige det fremtidige Sprint arbejde. Daily Scrum afholdes på samme tid og sted hver dag for at reducere kompleksiteten.

5. Opdateret sektion: Daily Scrum for at præcisere målet for Daily Scrum inklusive dette afsnit:

Strukturen af mødet fastlægges af Development Teamet og kan varetages på forskellige måder hvis det fokuserer på fremdrift mod at nå Sprint Goal. I nogle Development Teams anvendes spørgsmål, andre anvender mere diskussionspræget dialog. Herunder er eksempler på hvad der kan anvendes:

- Hvad bidrog jeg med i går der hjalp Development Teamet med at opnå Sprint Goal?
- Hvad vil jeg bidrage med i dag så Development Teamet kan nå Sprint Goal?
- Ser jeg nogen udfordringer, der forhindre mig eller Development Teamet i at nå Sprint Goal?

6. Tilføjet klarhed vedr. time boxes

Brugt ordet "maksimalt" for at fjerne alle spørgsmål om at en begivenhed skal have en bestemt længde og i stedet tydeliggjort, at det er den maksimale tid der er afsat.

7. Tilføjet følgende til sektion: Sprint Backlog

For at sikre løbende forbedringer indeholder det mindst en høj prioritet proces forbedring identificeret ved det seneste Retrospective møde.

8. Tilføjet klarhed til sektion: Increment

Et Increment er en samling af inspiserbart færdigt arbejde som understøtter empirisme ved slutningen af Sprintet. Incrementet er et skridt på vej mod visionen eller målet.