

Contents

Ce este cadrul de lucru SCRUM?	2
Practicile SCRUM	2
<i>Scrum roles</i>	2
Product owner	3
ScrumMaster	4
Echipa de dezvoltare	6
<i>SCRUM Activities</i>	7
The Sprint	7
Sprint Planning	9
Daily Scrum	10
Sprint Review	10
Sprint Retrospective	11
<i>SCRUM Artifacts</i>	12
Product Backlog	12
Sprint Backlog	13
Increment and Definition of Done	13
Bibliografie	15

Ce este cadrul de lucru SCRUM?

SCRUM nu este un proces standardizat unde poți urma o serie de pași succesivi care să garanteze că va produce într-un anumit timp și cu un anumit buget, un produs de înaltă calitate care va fi pe placul clienților. În schimb, SCRUM este un cadru de lucru simplu care se bazează pe un set de valori, principii și practici pentru a ajuta oamenii, echipele și organizațiile să genereze valoare folosind soluții adaptive pentru rezolvarea problemelor complexe.

Practicile SCRUM

Practicile SCRUM sunt grupate în unități de bază cum ar fi roluri, activități, artefacte, precum și reguli asociate acestora. (Figura 1)

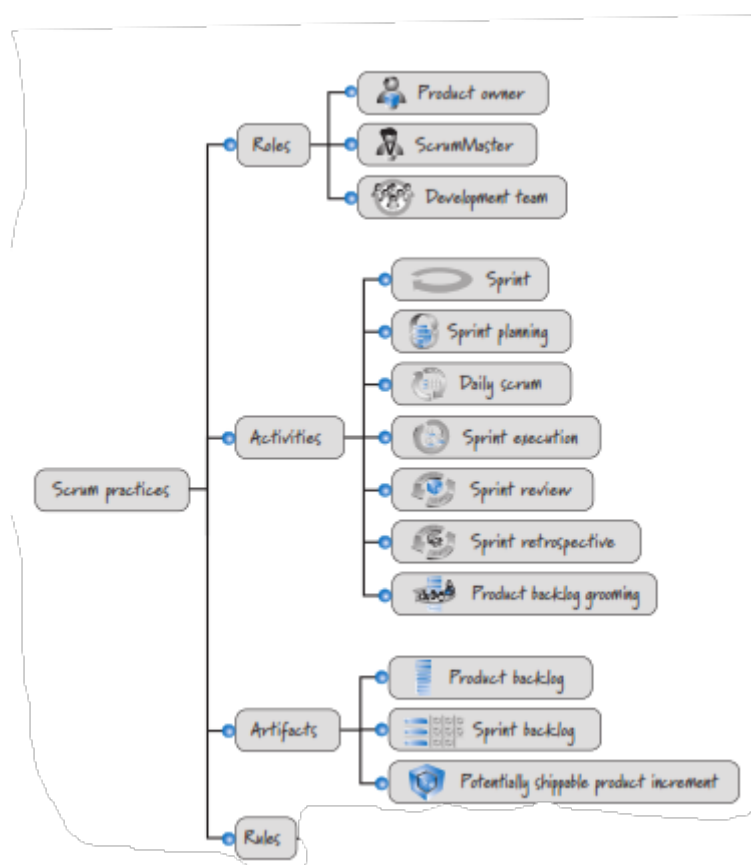


Figura 1 SCRUM Practices

Scrum roles

Dezvoltarea SCRUM constă în una sau mai multe echipe SCRUM (SCRUM Teams), fiecare dintre acestea îndeplinind cele trei responsabilități sau roluri SCRUM: product owner, ScrumMaster și development team (Figura 2). În utilizarea SCRUM se pot defini și alte roluri, însă cadrul SCRUM solicită numai cele trei roluri menționate anterior. În cadrul SCRUM Team,

nu exista sub-echipe sau ierarhii, ci exista o unitate de profesionisti menita sa realizeze un singur obiectiv la un moment dat, denumit Product Goal.

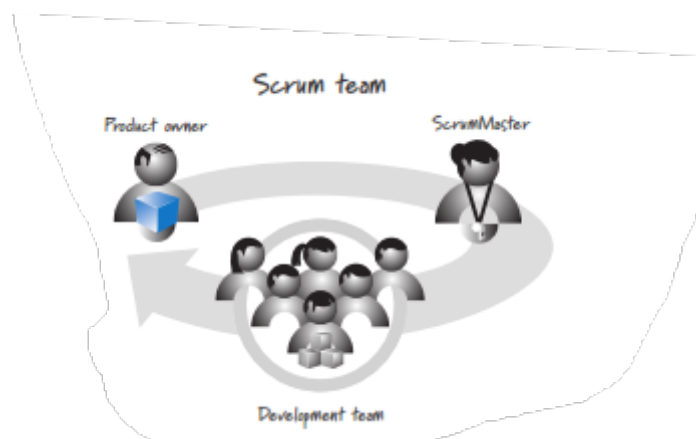


Figura 2 Scrum roles

Product owner

Un Product Owner este singura autoritate responsabila pentru a decide functionalitatile si caracteristicile a ceea ce se va dezvolta, precum si ordinea in care se va realiza dezvoltarea acestora. Acesta trebuie sa priveasca in cel putin doua directii simultan. Pe de o parte, un Product Owner trebuie sa inteleaga nevoile si prioritatile organizationale ale partilor interesate, clientilor si utilizatorilor atat de bine incat sa actioneze in numele acestora. Pe de alta parte, are scopul de a mentine o comunicare constanta si activa cu echipa de dezvoltare, precum si de a le transmite acestora o viziune clara asupra scopurilor pe care acestia le au de indeplinit, respectiv ce au de dezvoltat precum si pasii sau ordinea necesara pentru dezvoltare. De asemenea, un Product Owner trebuie sa se asigure ca functionalitatile si criteriile de acceptare a caracteristicilor sunt specificate, precum si ca testele care verifica aceste criterii sunt realizate pentru a verifica daca aceste functionalitati si caracteristici sunt corecte si complete. Astfel, un Product Owner este responsabil de intregul succes asupra solutiei de dezvoltare sau mentenanta.

Nu conteaza daca se pune accentul pe un produs extern sau pe o aplicatie interna; un Product Owner tot are obligatia sa se asigure ca atinge obiectivul de a realiza cea mai valoroasa munca posibila, acest lucru incluzand si accentul pe munca tehnica.

Product Owner este o persoana cu roluri depline si responsabilitati semnificative. In anumite situatii acesta poate delega din sarcini altor persoane, insa tot el ramane responsabil pentru sarcinile delegate.

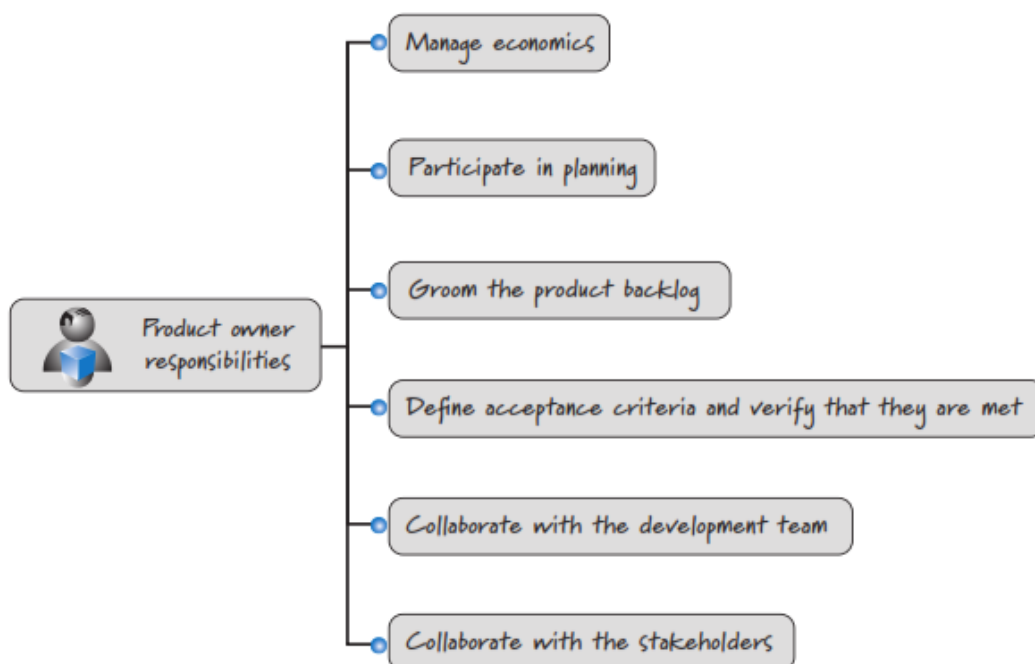


Figura 3 *Principalele responsabilitati ale unui Product Owner*

De asemenea, una dintre atributiile unui Product Owner este aceea de a gestiona eficient Product Backlog, prin:

- dezvoltarea si comunicarea explicita a Product Goal-ului;
- crearea si comunicarea clara a elementelor din Product Backlog;
- ordonarea elementelor din Product Backlog;
- asigurarea faptului ca intregul Product Backlog este transparent vizibil si inteles;

Un Product Owner poate reprezenta cerintele mai multor parti interesate in Product Backlog. Daca cei interesati doresc sa schimbe componenta sau ordinea elementelor in Product Backlog trebuie sa il convinga pe Product Owner.

ScrumMaster

ScrumMaster este unul dintre cele trei roluri cheie care alcatuiesc fiecare echipa SCRUM. In timp ce un Product Owner este axat pe construirea unui produs adecvat, un Scrum Master are ca scop principal sa ajute pe toata lumea sa inteleaga si sa adopte valorile, principiile si practicile SCRUM. Un ScrumMaster actioneaza ca un arbitru intre echipa de dezvoltare si Product Owner. Un ScrumMaster actioneaza totodata si ca un adevarat lider ajutand echipa SCRUM si restul organizatiei sa traseze directii clare pentru dezvoltarea propriilor performante de inalta calitate in conformitate cu cadrul de lucru SCRUM.

Figura 4 ilustreaza principalele responsabilitati pe care le indeplineste un ScrumMaster:

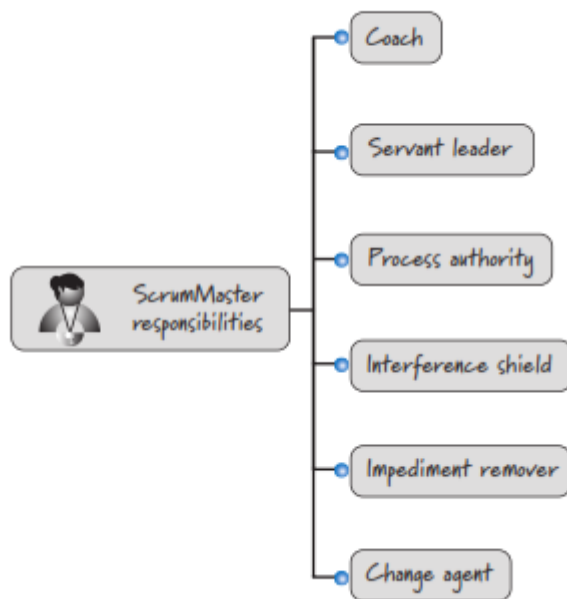


Figura 4 Principalele responsabilitati ale unui ScrumMaster

Coach

Un ScrumMaster este un arbitru agil pentru echipa SCRUM – atat pentru echipa dedezvoltare cat si Product Owner, astfel acesta poate inlatura barierele intre roluri si sa permita Product Owner-ului sa conduca in mod direct dezvoltarea.

Analog unui arbitru al unei echipe sport, ScrumMaster observa in ce mod echipa utilizeaza SCRUM si face tot ce ii sta in putinta sa o ajute sa atinga urmatorul nivel de performanta. In cazul in care echipa nu este capabila sa rezolve problemele aparute, acesta este prezent pentru a rezolva personal problemele.

Printre principalele abilitati ale unui ScrumMaster putem mentiona abilitati de colaborare, de protectie, de informare, de transparenta, de rabdare, de adresare constata de intrebari.

Servant Leader

Un ScrumMaster este descris adesea ca un leader intotdeauna la dispozitia echipei SCRUM (servant leader). Chiar si atunci cand acesta actioneaza ca un arbitru de echipa, un ScrumMaster este primordial la dispozitia echipei SCRUM, asigurandu-se ca nevoile de inalta prioritate ale acesteia vor fi indeplinite. Un servant leader nu va inteba niciodata: „Ce veti face astazi pentru mine?”, in schimb acesta va inteba: „Ce pot face astazi pentru a va ajuta astfel incat echipa sa fie mai eficienta?”.

Process Authority

Prin aceasta responsabilitate, un ScrumMaster este imputernicit sa se asigure ca echipa SCRUM promulga si adera la valorile, principiile si practicile SCRUM impreuna cu abodarele specifice ecipei SCRUM. ScrumMaster ajuta in mod continuu echipa SCRUM sa imbunatateasca prrocesul, oricand e posibil sa maximizeze valoarea generata.

„Autoritate” în acest context nu este echivalent cu autoritatea pe care o exercita un manager functional sau un manager de proiect. De exemplu, un ScrumMaster nu poate face angajari sau concedieri si nu poate spune echipei ce sacini sa indeplineasca si cum sa le indeplineasca. In schimb, un ScrumMaster ajuta echipa sa defineasca si sa adere la propriul sau proces cu scopul de se asigura ca lucruile vor fi realizate.

Interference Shield

Un ScrumMaster protejeaza echipa de dezvoltare de interferentele cu mediul extern pentru ca acestia sa ramana axati pe generarea de valoare cu fiecare sprint. Astfel, acesta actioneaza ca un interceptor.

Impediment Remover

Un ScrumMaster este responsabil pentru eliminarea impedimentelor care inhiba productivitatea echipei.

Change Agent

Un ScrumMaster trebuie sa ajute sa fie depasite atat impedimente de functionare cat si sa ajute oamenii sa isi formeze opinii si sa aiba anumite viziuni. Un ScrumMaster ii ajuta pe ceilalti sa inteleaga nevoia de schimbare, impactul pe care il are SCRUM in afara echipei SCRUM , precum si gama larga de beneficii pe care SCRUM o poate indeplini. De asemenea, un ScrumMaster se asigura ca schimbarile efective se vor realiza la toate nivelele organizatiei, permitand nu doar un succes pe termen scurt, ba mai mult, beneficii pe termen lung ale utilizarii SCRUM.

Echipa de dezvoltare

Abordarile dezvoltarii unui software traditional definesc diverse tipuri de joburi, cum ar fi arhitect, programator, tester, administrator de baze de date, designer UI si asa mai departe. SCRUM defineste rolul echipei de dezvoltare care este o combinatie incucisata din aceste tipuri de persoane. In particular, echipa de dezvoltare este una dintre cele trei roluri cheie care alcatuiesc fiecare echipa SCRUM. Membrii echipei de dezvoltare, colectiv, au abilitatile necesare pentru sa genereze valoare conform cerintelor unui Product Owner.

Cele mai multe dintre organizatii obisnuiesc sa imparta diferitele roluri ale jobului in roluri specifice echipei. Aceste organizatii pot avea o echipa de designeri, una de dezvoltare si alta de tester.

In SCRUM, echipa de dezvoltare trebuie sa realizeze intreaga munca pentru a produce una sau mai multe functionalitati ale fiecarui sprint, incluzand designul, dezvoltarea, integrarea si testarea functionalitatilor. Astfel, este nevoie de o echipa care are abilitatile necesare sa indeplineasca toate aceste atributii.

Uneori organizatiile incearca sa mentina separata o echipa de testare sau QA in timp ce realizeaza echipa SCRUM. Uneori a avea o echipa care se bazeaza exclusiv pe testare poate fi necesara, ca de exemplu in cazul in care exista cerinte specifice de reglementare care sa prevada ca este necesara o echipa care sa indeplineasca anumite criterii de testare. Cu toate acestea, in cele mai multe situatii nu exista o asemenea necesitate. Testarea poate fi pe deplin inclusa in

munca realizata cu fiecare sprint. Prin urmare, echipa de dezvoltare care realizeaza munca in acel sprint va trebui sa realizeze si testarea.

Ori de cate ori este posibil, ar trebui sa fie create echipe care sa indeplineasca functionalitati incrucisate. Divizand munca in diferite roluri specifice echipei este un impediment in utilizarea cu succes a SCRUM. Pentru o divizare in diferite roluri specifice echipei trebuie sa existe nevoi reale.

Principalele responsabilitati ale echipei de dezvoltare

Figura 5 ilustreaza principalele responsabilitati ale echipei de dezvoltare.

Printe acestea putem enumera:

- Crearea unui plan pentru sprint, denumit Sprint backlog;
- Asigurarea calitatii prin aderarea la „Definition of Done”, respectiv ce inseamna „realizat”;
- Analizarea si adaptarea zilnica a planului lor prin „Sprint Goal”;
- Raspunderea profesionala pentru rezultatele obtinute;

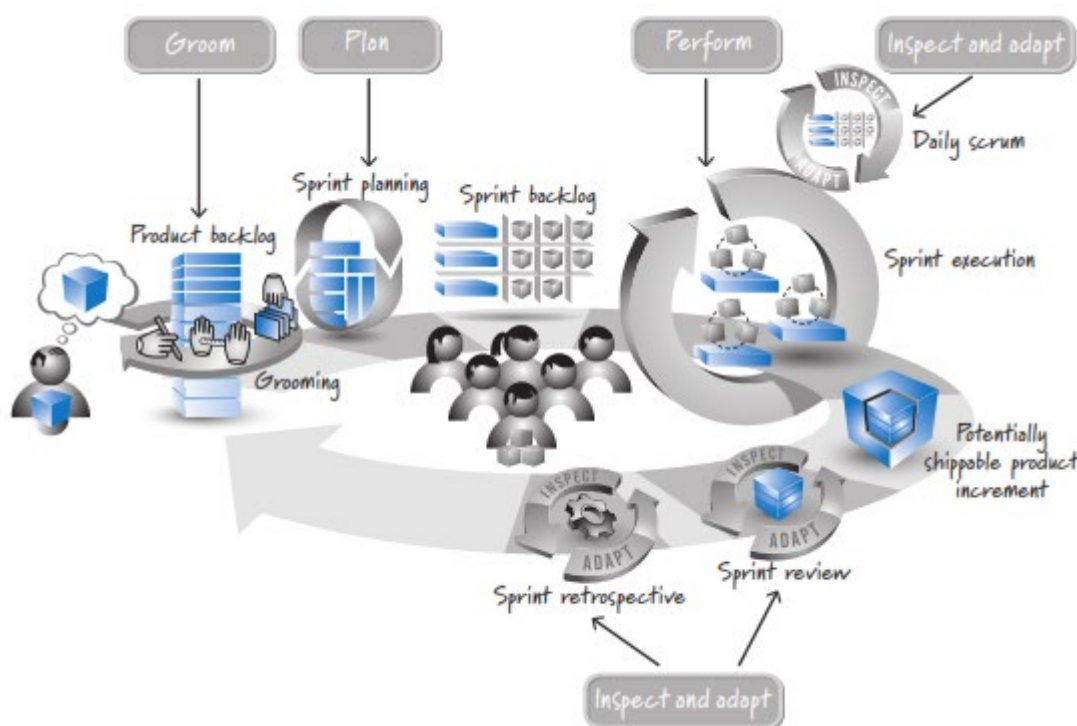


Figura 5 *Principalele responsabilitati ale echipei de dezvoltare cu privire la activitatile SCRUM*

SCRUM Activities

The Sprint

In SCRUM, munca este indeplinita prin iteratii sau cicluri lunare, denumite sprinturi. Munca realizata in fiecare sprint trebuie sa creeze valoare tangibila pentru clienti sau utilizatori.

Sprinturile au durata fixa de pana la o luna, cu data fixa de inceput si data fixa de finalizare pentru a asigura consistenta. Un sprint nou incepe imediat dupa finalizarea celui anterior.

Sprinturile de scurta durata furnizeaza mai multe beneficii:

- Sunt mai usor de planificat;
- Genereaza feed-back-uri rapide care ajuta la eliminarea abordarilor de dezvoltare nefavorabile, deoarece in timpul fiecarui sprint cream un software, iar ulterior avem posibilitatea sa analizam si sa adaptam ceea ce am construit si modul in care am construit.
- Nu doar ca imbunatatesc economiile printr-un feed-back mai rapid, ci se imbunatatesc si randamentul investitiilor in ansamblu.
- Ajuta la mentinerea entuziasmului participantilor la un nivel inalt prin livrarea produsului muncii frecvent. Multumirea pentru livrarea produsului muncii cu o frecventa cescuta ajuta la imbunatatirea interesului si a dorintei de a continua munca in vederea indeplinirii scopului final.
- Furnizeaza multiple puncte de control semnificative.

Durata consecventa

Ca regula, intr-un efort de dezvoltare dat, o echipa poate alege o durata consecventa pentru sprinturile sale si sa nu le schimbe atata timp cat nu exista un motiv bine intemeiat, ca de exemplu:

- scaderea sprintului de la 4 saptamani la 2 saptamani cu scopul de a obtine un feed-back mai frecvent, dar se doreste realizarea acestui lucru numai pentru cateva saptamani inainte de a lua decizia finala;

-sarbatorile anuale sau finalul de an fiscal fac ca sprintul de 3 saptamani sa fie mai practic in aceasta situatie decat un sprint de 2 saptamani.

Sprinturile care au aceeasi durata furnizeaza un ritm regulat, predictibil pentru efortul de dezvoltare SCRUM.

De asemenea, utilizarea aceleiasi durate ajuta la splificarea planificarii activitatii. Cand toate sprinturile sunt de aceeasi durata echipa se familiarizeaza cu cantitatea de munca pe care trebuie sa o duca la bun sfarsit intr-un sprint.

In timpul Sprint-ului:

- O regula SCRUM importanta este ca o data stabilit scopul sprintului si executia sprint a inceput nu mai este permisa nicio modificare care sa afecteze Sprint Goal-ul(fiecare sprint poate fi descris printr-un sprint goal - care sa descrie scopul businessului si valoarea sprintului)
- Mentine durata consecventa a sprintului, daca nu exista motive clare de modificare a acesteia;
- Nu se reduce calitatea;
- Se imbunatatesc Product Backlog dupa cum este necesar;

- Scopul poate fi clarificat si renegociat cu Product Owner-ul conform cunostintelor dobandite;

O situatie in care Sprint Goal poate fi anulat este aceea ca acesta devine depasit. Anularea poate fi realizata numai de catre Product Owner.

Sprint Planning

Fiecare sprint incepe cu o planificare a sprintului (sprint planning), un moment in care echipa SCRUM se intalneste pentru a stabili scopul sprintului si a determina ce anume se va furniza de-a lungul sprintului ce va urma.

Un Product Backlog poate consta in mai multe saptamani si luni de munca, ceea ce inseamna ca este mult mai mult decat poate fi cuprins intr-un scurt sprint. Pentru a stabili cel mai important subset de elemente de Product Backlog ce vor fi construite in urmatorul sprint, echipa SCRUM stabileste planificarea sprintului. In timpul planificarii sprintului echipa SCRUM stabileste care sunt scopurile sprintului, iar echipa de dezvoltare stabileste elementele specifice Product Backlog-ului care sunt aliniate scopului si pe care le pot furniza in mod real pana la finalul sprintului. Pentru a obtine incredere in ceea ce va livra, echipa de dezvoltare creaza un plan pentru modul in care va duce la bun sfarsit elementele din Product Backlog. Elementele din Product Backlog, impreuna cu planul formeaza sprint backlog.

Cand se realizeaza un Sprint Planning se pun urmatoarele intebari:

1.Care este valoarea sprintului?

Product Owner-ul propune modul in care valoarea si utilitatea produsului pot creste in Sprint-ul curent. Echipa SCRUM colaboreaza pentru a defini Sprint Goal-ul si ulterior comunica valoarea Sprint-ului celorlalte parti interesate. Sprint Goal-ul trebuie finalizat inainte de sfarsitul sesiunii de Sprint Planning.

2.Ce se poate realiza in acest sprint?

Echipa de dezvoltare impreuna cu Product Owner-ul selecteaza elemente din Product Backlog pentru a le include in Sprint-ul curent. In timpul acestui proces echipa SCRUM poate imbunatati aceste elemente, ceea ce sporeste intelegerea si increderea.

Reprezinta o adevarata provocare sa stabilesti cantitatea de munca care poate fi realizata intr-un Sprint. Cu toate acestea, cu cat echipa de dezvoltare cunoaste performanta din trecut, capacitatea lor viitoare, precum si Definition of Done (DOD), cu atat vor avea mai multa incredere in previziunile lor pentru Sprint.

3.Cum se va realiza ceea ce s-a stabilit?

Pentru fiecare element selectat din Product Backlog, echipa de dezvoltare stabileste ce este de facut pentru a crea un increment conform Definition of Done. Astfel, pentru aceasta echipa de dezvoltare stabileste modul in care descompun elementele din Product Backlog in activitati mai mici de o zi sau mai putin.

Daily Scrum

Daily Scrum este o activitate critica, cu frecventa zilnica ce se inspecteaza si adapteaza cu scopul de a ajuta echipa sa gaseasca un flux mai flexibil si mai usor spre o solutie. Astfel, daily scrum este o activitate de 15 minute care are loc o data la 24 de ore. Daily scrum serveste ca o inspectie, sincronizare si un plan zilnic de adaptare a activitatii care ajuta o echipa care se organizeaza singura sa isi imbunatateasca activitatea.

Sprint Review

Un Sprint Review are loc la finalul unui sprint, moment in care echipa SCRUM, impreuna cu partile interesate evalueaza rezultatul. Aceasta este o alta intalnire importanta pentru evaluare si adaptare. Un Sprint Review dureaza maxim 4 ore in cazul evaluarii unui sprint de o luna. Pentru sprinturi mai scurte de o luna, evenimentul este adesea mai scurt de 4 ore.

Cel mai frecvent, un Product Owner este cel care stie care sunt partile interesate sa participe la eveniment. Numai partile interesate cheie sau cele relevante vor fi invitate. Parti interesate cheie, sunt cele mai importante parti interesate asupra produsului la momentul evaluarii unui sprint dat. Un produs are mai multe parti interesate in realizarea acestuia, insa nu toti dintre acestia vor fi interesati de rezultatul unui sprint particular, in consecinta acestia nu vor trebuie sa participe la intalnirile care nu prezinta interes. Cu alte cuvinte, nu fiecare modificare este importanta pentru cineva. De asemenea, alte parti care iau parte la intalnire sunt cei din echipa de dezvoltare si Scrum Master-ul. Rolul celui din urma este de a usura intalnirea si de a se asigura ca toti inteleg scopul. Product Owner-ul trebuie sa fie neaparat prezent la intalnire. Acesta nu poate delega o alta persoana sa il reprezinte.

Aceasta intalnire este un eveniment foarte important care arata rezultatele muncii, lucru care caracterizeaza Product Ownerul. Rezultatul sprintului este in esenta ceea ce a reusit echipa SCRUM sa faca in cadrul sprintului. Acesta poate fi product incrementul care indeplineste sprint goal-ul sau orice alt tip de rezultat pe care l-au produs in sprintul curent.

In cadrul intalnirii trebuie discutate atat partile care au mers bine si au avut rezultatele dorite, cat si partile care nu au mers suficient de bine, precum si dificultatile intampinate.

In cazul in care rezultatul unui sprint l-ar surprinde pe Product Owner ar insemna ca acesta a pierdut colaborarea cu echipa in timpul sprintului.

Una dintre partile cele mai importante ale intalnirii este data de momentul in care echipa SCRUM incearca sa demonstreze ca munca lor este realizata cu succes in conformitate cu Definition of Done pentru realizarea product incrementului. Aceasta demonstratie asupra realizarii incrementului se realizeaza cu scopul de a obtine feed-back si de a incuraja colaborarea cu privire la ceea ce urmeaza de facut. Echipa SCRUM nu trebuie sa se limiteze numai la a face o prezentare, ci trebuie sa implice partile interesate in testarea efectiva a produsului pentru a intelege cat mai bine ce a fost construit.

In cazul in care partile interesate observa ceva important care trebuie reparat, echipa SCRUM poate realiza acest lucru in urmatorul sprint si va realiza cat mai repede un nou increment.

Aceasta intalnire este o buna oportunitate pentru partile interesate sa puna intrebari sau sa propuna modificari sau functionalitati noi. Acestia vor trebui de asemenea sa inteleaga care au fost schimbarile din mediul de activitate de la ultimul sprint. Mediul de activitate in esenta

semnifica orice ar putea impacta produsul, ca de exemplu: modificari de piata ca cerere scazuta a clientilor, noi oportunitati, competitori care dezvoltă alt produs, noi reglementari ca impacteaza produsul. Astfel, adesea aceasta intalnire conduce la revizuirea product backlogului, modificari bazate pe feedback sau pe modificarile mediului de activitate.

Ce se intampla in cazul in care sunt elemente din product backlog care nu sunt inca realizate sau nu sunt realizate pe deplin? In primul rand, ceea ce nu a fost finalizat nu va fi demonstrat in cadrul intalnirii si nu va face parte din product increment, orice lucru nefinalizat va fi reinclus in product backlog si Product Owner-ul va decide ce este de facut mai departe.

Sprint Retrospective

Sprint Retrospective este ultimul eveniment aferent sprintului si se numeste „*The Retro*”. In timp ce in intalnirea din Sprint Review, focusul a fost pus pe evaluarea si adaptarea produsului, scopul Sprint Retrospective este de a evalua si adapta dezvoltarea procesului intr-o maniera care creste calitatea si eficienta.

Numai echipele care reflecta asupra a ceea ce s-a intamplat si identifica ce poate fi imbunatatit in viitor pot deveni mai eficiente in timp.

Sprint Retrospective este un eveniment intern al echipei SCRUM, in care nu sunt implicate parti externe cum ar fi partile interesate sau managerii, deoarece participarea acestora ar inhiba echipa sa discute deschis despre problemele pe care le-au constatat si ar reduce din eficienta intalnirii.

Scrum Master-ul actioneaza in vederea facilitarii intalnirii, asigurandu-se ca intalnirea este pozitiva si productiva.

Durata evenimentului este de maxim trei ore pentru sprinturi care tin mai putin de o luna.

Acest eveniment nu este numai pentru echipa de dezvoltare, ci trebuie sa participe toti membrii echipei SCRUM. Nu exista sugestii clare asupra desfasurarii si structurarii intalnirii, ci se pune accent pe scopul si rezultatul dorit. Scopul este de a analiza modul in care s-a desfasurat ultimul sprint cu privire la indivizi, interactiuni, procese, instrumente precum si la Definition of Done.

Mediul de desfasurare al evenimentului trebuie sa fie unul pozitiv, fara se arate cu degetul spre altii sau sa fie blamati si sa se exprime frustrare. Rolul Scrum Master-ului este individual sa ajute echipa sa obtina un castig din aceasta intalnire.

Ca prim pas, echipa aduna reflectii cu privire la ce a mers bine si ce poate fi imbunatatit. In cazul in care ceva nu a mers bine in timpul sprintului, este important sa se identifice cauza. Si aproape intotdeauna este ceva de discutat, ca de exemplu de ce a esuat echipa in a indeplini sprint goal-ul, cum au afectat anumite impedimente previzionarea si asa mai departe. Din aceste motive, acest eveniment este obligatoriu.

Indiferent de cat de experimentata este echipa, este intotdeauna nevoie de analiza a ceea ce a fost in sprintul tocmai incheiat. Echipa SCRUM va discuta despre ce a identificat, despre asteptari de a intelege diferite perspective, despre identificarea partii care nu a mers bine si despre posibilele imbunatatiri.

Este la fel de important sa se identifice ce a mers bine astfel incat echipa sa aplice aceste lucruri pe imbunatatirile pe care le identifica si urmeaza sa le implementeze. Echipa SCRUM

poate implementa aceste imbunatatiri in product backlog, dar ideal ar fi sa adauge una sau mai multe imbunatatiri in urmatorul sprint backlog.

In cazul in care sunt identificate imbunatatiri, insa aceste nu sunt implementate nicaieri nu se va schimba nimic in imbunatatirea calitatii muncii pe care o livreaza.

Definition of Done poate fi modificata pentru a include noi conventii, standarde sau ghiduri sau sa le faca pe cele existente mai restrictive, insa aceasta nu trebuie creata sau actualizata cu fiecare sprint.

SCRUM Artifacts

In scrum un artefact este ceva care este creat de cei care utilizeaza SCRUM. Ghidul SCRUM defineste trei artefacte: product backlog, sprint backlog si increment, adesea denumit si incrementul produsului – product increment.

Product Backlog

In SCRUM, product backlog este un artefact proiectat pentru a furniza transparenta si oportunitati de evaluare si adaptare.

Product backlog este o lista ordonata a ceea ce este necesar pentru produs. Backlog-ul poate contine noi caracteristici sau imbunatatiri pentru repararea celor existente sau alte modificari care trebuie sa fie realizate asupra produsului.

Caracteristicile elementelor din backlog depind de produs sau de domeniul unde munca de dezvoltare are loc, in mod uzual un element va avea atribute ca descrierea, ordinea, marimea si valoarea. Elementele din product backlog adesea includ criterii de acceptare care vor ajuta testele daca munca realizata este completa.

Singura persoana responsabila pentru product backlog este Product Owner-ul. Managementul product backlog-ului este activitatea prin care se fac modificari asupra product backlog-ului.

Deci cum este mai exact product backlog-ul creat si gestionat?

Totul incepe cu un scop sau o viziune. Product Owner-ul trebuie sa defineasca un obiectiv pe care trebuie sa il indeplineasca produsul. Pentru asta trebuie sa inteleaga cerintele afacerii prin colaborarea constanta cu persoanele interesate, sa cunoasca conditiile pietei si sa ia in considerare alte surse relevante. Proprietarii privati vor adauga caracteristici sau idei la product backlog asa cum sunt relevante pentru produs, insa continutul product backlogului trebuie sa fie aliniat cu protocolul.

Gestionarea product backlog-ului este un proces in derulare, pe cand Product Owner-ul este responsabil de product backlog si poate dispune, adauga sau elimina elemente in product backlog de cate ori trebuie. Este foarte important sa se lucreze foarte strans cu echipa de dezvoltare. In timpul product backlog-ului, Product Owner colaboreaza cu echipa de dezvoltare pentru a sparge elementele mari in unele mai mici cu scopul de a fi mai usor de descris si dezvoltat. Cu cat elementele sunt mai mici, cu atat este mai bine.

Acesta va lucra la adaugarea detaliilor, marimii si ordinii product backlogului.

Marimea unui element arata cat este de mare sau mic efortul de a duce la bun sfarsit munca. Procesul de determinare al marimi unui element este numit dimensionare sau estimare. Echipa de dezvoltare este responsabila de dimensionare si are ultimul cuvânt. Product Owner-

ul si echipa de dezvoltare colaboreaza pentru a intelege elementele si adesea cauta cai de a reduce complexitatea prin realizarea unor compromisuri.

Product backlogul este emergent si niciodata complet sau final. Modificarile din product backlog se bazeaza pe cerintele afacerii, conditiile pietei si alti factori relevanti.

Elementele din product backlog care pot fi realizate in urmatorul sprint sunt socotite ca pregatite pentru selectie, adica sunt suficient de mici pentru a se potrivi intr-un sprint, suficient de detaliate si pregatite astfel incat echipa de dezvoltare sa poata incepe imediat sa lucreze la acestea. Cu alte cuvinte, un element din product backlog este pregatit cand echipa SCRUM considera ca poate fi facut.

Sprint Backlog

In timpul intalnirii de sprint planning, Product Ownerul explica ce este de intreprins in continuare pentru a creste valoarea produsului si intreaga echipa SCRUM formuleaza un sprint goal.

Prin colaborarea cu Product Owner-ul, echipa de dezvoltare va decide care element incepand cu cel din varful product backlog-ului va fi adaugat in sprint backlog.

Sprint backlog va contine de asemenea un plan pentru livrarea incrementului si realizarea sprint goal-ului.

Diferenta intre product backlog si sprint backlog:

- Un product backlog este o lista ordonata de idei si caracteristici pe care produsul trebuie sau le poate avea la dispozitie pentru a le indeplini;
- Un sprint backlog este creat la inceputul fiecarui sprint si contine doar o parte din elementele product backlog-ului, respectiv elementele ce vor fi incluse in sprintul curent, precum si un plan de livrare a functionalitatilor si a sprint goal-ului. Planul este practic o descompunere a fiecarui product backlog in unitati mai mici care permit dezvoltatorilor sa realizeze incrementul, unitati denumite taskuri.

Cand dezvoltatorii selecteaza elementele din product backlog, ei cred ca le pot realiza in acel sprint. Astfel, practic creaza o previzionare a ceea ce vor livra. Previzionarea sprintului nu este o garantie, o promisiune sau un angajament. In timpul sprintului se pot intampla lucruri neasteptate.

Sprint backlog face transparenta toata munca pe care dezvoltatorii o socotesc necesara.

Astfel, se poate spune ca sprint backlog este un artefact temporar care exista numai de-a lungul sprintului. Fiecare sprint are un nou sprint backlog. Toata munca care a ramas nefinalizata in sprint backlog va fi reintrodusa in product backlog. Ulterior, Product Owner va decide ce este de facut in continuare.

Dezvoltatorii detin controlul asupra sprint backlog-ului. Acestia vor putea modifica sprint backlog-ul de-a lungul sprintului dupa cum considera necesar pentru a indeplini sprint goal-ul.

Increment and Definition of Done

Incrementul este o noua versiune a produsului care se adauga la incrementurile anterioare rezultate din sprinturi anterioare.

Dezvoltatorii doresc sa livreze cel putin un nou increment cu fiecare sprint. Fiecare increment este o versiune imbunatatita si utilizabila a produsului. Este libertatea Product Owner-ului sa decida daca si cand va realiza incrementul. Cu toate acestea, incrementul trebuie sa fie utilizabil fara sa solicite o munca aditionala, cum ar fi documente de testare. Cand un element din product backlog sau increment este considerat complet sau facut (done), toata lumea trebuie sa inteleaga semnificatia lui done.

Uneori organizatia va defini standarde de calitate si acestea trebuie sa fie urmate pentru a face mai usor pentru toata lumea sa inteleaga ce inseamna munca indeplinita.

Compania trebuie sa creeze Definition of Done care poate fi utilizata sa se evalueze daca munca a fost realizata conform criteriilor de calitate agreeate.

In SCRUM, fiecare sprint trebuie sa creeze cel putin un increment care trebuie sa adere la Definition of Done. Daca un element din Product Backlog al sprintului nu respecta Definition of Done, acesta nu va fi inclus in increment.

Cu cat Definition of Done contine mai multe criterii stricte, cu atat se asigura o calitate mai mare. In timpul Sprint retrospective, echipa SCRUM planifica cum sa creasca calitatea produselor si o modalitate de a reusi acest lucru este prin adaptarea Definition of Done. Nu exista standarde pe care sa le contina Definition of Done. Ceea ce va contine difera de la o echipa la alta si de la un produs la altul.

Prin utilizarea unei Definition of Done se urmareste:

- transparenta in realizarea incrementului;
- ghidarea echipei de dezvoltare cu privire la cate elemente din product backlog sa aleaga in cadrul intalnirii de sprint planning;
- angajament pentru increment, respectiv cu cat este finalizat mai rapid un element din product backlog si respecta Definition of Done, se creeaza un nou increment.

Bibliografie

Rubin, Kenneth S. - *Essential SCRUM – A practical guide to the most popular agile process*, 2013, Pearson Education, Inc.

Ken Schwaber, Jeff Sutherland – „ *The Scrum Guide - The Definitive Guide to Scrum: The Rules of the Game*, November 2020 <https://www.scrum.org/resources/scrum-guide>