

SCRUM ceļvedis

Spēles noteikumi

Izstrādājuši un apkopjuši: Kens Švābers (Ken Schwaber) un Džefs Sazerlands (Jeff Sutherland)

© 1991-2013 Ken Schwaber and Jeff Sutherland, All Rights Reserved

© Tulkojums latviešu valodā: SIA Autentica, 2013

Saturs

Tulkojuma priekšvārds	4
<i>Scrum</i> ceļveža nolūks	5
<i>Scrum</i> definīcija	5
<i>Scrum</i> teorija	5
Caurskatāmība	5
Pārbaude	6
Pielāgošana	6
<i>Scrum</i> komanda	6
Produkta īpašnieks	6
Izstrādes komanda	7
Izstrādes komandas lielums	8
<i>Scrum</i> Komandas kapteinis	8
<i>Scrum</i> Komandas kapteiņa atbalsts Produkta īpašniekam	8
<i>Scrum</i> Komandas kapteiņa atbalsts Izstrādes komandai	8
<i>Scrum</i> Komandas kapteiņa atbalsts Organizācijai.....	9
<i>Scrum</i> notikumi	9
Sprints	9
Sprinta atcelšana.....	10
Sprinta plānošana	10
Pirmais jautājums: Ko šajā sprintā var izdarīt?	10
Otrā daļa: Kā izvēlētais darbs tiks paveikts?	11
Sprinta mērķis	11
Ikdienas sanākšana	12
Sprinta apskate	12
Sprinta retrospekcija	13
<i>Scrum</i> artefakti.....	14
Produkta darbu uzkrājums.....	14

Virzības uz mērķi uzraudzība.....	15
Sprinta uzkrājums	15
Sprinta progresu mērījumi	16
Programmatūras Papildinājums.....	16
Artefaktu caurskatāmība	16
Pabeigtības definīcija jeb Ko nozīmē „Pabeigts” darbs.....	16
Beigu piezīme.....	17
Pateicība.....	17
Cilvēkiem.....	17
Vēsture.....	17

Tulkojuma priekšvārds

Tā kā ikdienā – būsim godīgi! – mēs, izstrādātāji, īpaši nedomājam, kā vienu vai otru angļu valodā radušos, varētu pat teikt, dzimušu, terminu lietot latviski, tad pirmajā brīdī kāds tulkojums var šķist nepierasts, pat nepareizs. Jo vairāk domāsim, kā precīzāk nosaukt kādu lietu vai jēdzienu (nevis tikai burtiski iztulkosim vai tieši pārnesīsim no angļu valodas), jo labāk izpratīsim tās nozīmi un jēgu.

Termins angļu valodā	Tulkojums	Tulkojuma varianti, piezīmes
Adaptation	Pielāgošana	
Daily Scrum	Ikdienas sanāksšanas	
Development Team	Izstrādes komanda	
Done	Pabeigts	Pabeigts darbs kontekstā ar komandas darbu pabeigtības definīciju. Variants: Gatavs
Increment	(Produkta) Papildinājums	
Inspections	Pārbaudes	
Product Backlog	Produkta darbu uzkrājums	
Product Owner	Produkta īpašnieks	
Scrum Artefacts	Scrum artefakti	
Scrum Events	Scrum notikumi	
Scrum Framework	Scrum ietvars	
Scrum Master	Scrum komandas kapteinis	Komandas kapteinis kā sporta komandā. Variants: Scrum koordinators.
Scrum Team	Scrum komanda	
Sprint	Sprints	
Sprint Backlog	Sprinta darbu uzkrājums	
Sprint Goal	Sprinta mērķis	
Sprint Planning Meeting	Sprinta plānošana	
Sprint Retrospective	Sprinta retrospekcija	
Sprint Review	Sprinta apskate	
Transparency	Caurskatāmība	Varianti: caurredzamība, pārskatāmība.

Scrum ceļveža nolūks

Scrum ir komplicētu risinājumu izstrādes un uzturēšanas ietvars. Ceļvedis iekļauj *Scrum* definīciju, kas sastāv no *Scrum*'a lomām, darbībām, artefaktiem, kā arī noteikumiem, kas nosaka visu *Scrum* daļu mijiedarbību.

Scrum ietvars ir izstrādāts un *Scrum* ceļvedis ir tapis Kena Švābera un Džefa Sazerlanda kopsadarbībā.

Scrum definīcija

Scrum (definīcija): Ietvars, kurā cilvēki var risināt sarežģītas adaptīvas problēmas, vienlaikus nodrošinot iespējami vērtīga risinājumu produktīvu un radošu piegādi.

Scrum ir:

- relatīvi viegls un nesarežģīts,
- vienkārši saprotams,
- grūti pārvaldāms.

Scrum ir procesu ietvars, kas tiek lietots sarežģītu produktu izstrādē kopš 20.gs. 90. gadu sākuma. *Scrum* nav process vai tehnika produkta radīšanai, drīzāk tas ir karkass, kura ietvaros var izmantot dažādus procesus un tehnikas. Pielietojot *Scrum*, kļūst redzama projektā izmantoto produktu vadības un izstrādes prakšu relatīvā lietderība un iedarbība, kā arī iespējas tos uzlabot.

Scrum ietvars sastāv no *Scrum* Komandām un tām piesaistītajām lomām, darbībām, notikumiem, artefaktiem un noteikumiem. Katrs ietvara komponents realizē noteiktu nolūku un ir būtiski nepieciešams *Scrum* veiksmīgai izmantošanai.

Scrum noteikumi apvieno notikumus, lomas, artefaktus, tā pārvaldot to mijiedarbību un ietekmes. *Scrum* noteikumi ir aprakstīti šī dokumenta pamattekstā.

Ir iespējamas dažādas *Scrum* ietvara lietošanas stratēģijas; tās ir aprakstītas citos avotos.

Scrum teorija

Scrum ir veidots uz pieredzē balstītas procesu kontroles teorijas jeb empīrisma bāzes. Empīrisms paredz, ka zināšanas tiek iegūtas caur pieredzi un lēmumi tiek pieņemti, balstoties uz zināmo (faktiem). *Scrum* paredz iteratīvas, inkrementālas pieejas pielietošanu risku pārvaldībai un paredzamības uzlabošanai.

Jebkuras uz zināšanām pamatotas procesu kontroles ieviešanas pamatā ir pārskatāmība, pārbaude (apskate) un pielāgošana.

Caurskatāmība

Procesa nozīmīgākajām daļām (aspektiem) ir jābūt zināmām un viegli pieejamām visiem, kas ir atbildīgi par šī procesa rezultātu. Caurskatāmība nozīmē, ka ir izmantoti zināmi standarti, definējot procesa daļas, lai tās varētu vienādi interpretēt visi iesaistītie.

Piemēram:

- Visiem procesā iesaistītajiem vienādi jāinterpretē jēdzieni un valoda, kas tiek izmantota procesa definēšanā, kā arī
- Darba veicējiem (izstrādātājiem) un šī darba akceptētājiem jābūt kopīgai izpratnei par jēdziena „Pabeigts” nozīmi.

Pārbaude

Scrum lietotājiem regulāri jāpārbauda *Scrum* artefakti, kā arī virzība uz mērķi, lai atklātu iespējamās novirzes no vēlamā. Pārbaudēm nav jābūt pārāk biežām, lai tās neietekmētu darba gaitu. Pārbaudes ir lietderīgākas, ja pārbaudu veicēji ir pieredzējuši un rūpīgi, kā arī zinoši konkrētajā darba areālā.

Pielāgošana

Ja pārbaudītājs atklāj, ka vienā vai vairākās procesa daļās ir novirzes ārpus akceptējamām robežām un ka izstrādājams produkts nebūs akceptējams, tad ir jākorrigē process vai procesa rezultātā izstrādātais. Korekcijas jāveic pēc iespējas ātrāk, lai samazinātu turpmākās novirzes.

Scrum apraksta četras formālās pārbaudu un pielāgošanas iespējas, tās aprakstītas šī dokumenta nodaļā „*Scrum* notikumi”:

- Sprinta plānošana,
- Ikdienas *Scrum* jeb Ikdienas sanāksšanas,
- Sprinta apskate,
- Sprinta retrospekcija.

Scrum komanda

Scrum komandu veido Produkta īpašnieks, Izstrādes komanda un *Scrum* Komandas kapteinis. *Scrum* komandas ir pašorganizētas un daudzfunkcionālas. Pašorganizētas komandas pašas izvēlas, kā vislabāk paveikt savu darbu, nevis to nosaka kāds ārpus Komandas esošs vadītājs. Daudzfunkcionālā komandā ir visas nepieciešamās kompetences, lai veiktu darbu neatkarīgi no ārpus komandas esošām personām. *Scrum* komandas modelis ir veidots tā, lai pilnveidotu tās elastīgumu, radītspēju un ražīgumu.

Scrum Komandas risinājumu piegādā iteratīvi un inkrementāli, tā paverot plašākas iespējas atgriezeniskās saites iegūšanai. Inkrementāla „Pabeigtu” produktu piegāde nodrošina, ka vienmēr ir pieejama potenciāli lietderīga strādājoša produkta versija.

Produkta īpašnieks

Produkta īpašnieks ir atbildīgs par produkta vērtības palielināšanu un Izstrādes komandas darbu. Veidi, kā to panākt, ir ļoti dažādi un atkarīgi no organizācijas, *Scrum* komandas, kā arī konkrētajām iesaistītajām personām.

Produkta īpašnieks ir vienpersoniski atbildīgs par Produkta darbu uzkrājumu. Produkta darbu uzkrājuma pārvaldība nozīmē:

- Skaidri izteikt un formulēt katru Uzkrājumā iekļauto vienību;
- Sakārtot Produkta darba uzkrājuma ierakstus tādā secībā, lai vislabāk sasniegtu mērķus un izpildītu uzdevumu;
- Nodrošināt Izstrādes komandas veiktā darba vērtību;
- Nodrošināt, ka Produkta darbu uzkrājums ir pieejams, caurskatāms un visiem saprotams, kā arī rāda, kādus darbus *Scrum* komanda veiks nākamās;
- Nodrošināt, ka Izstrādes komanda saprot Produkta darbu uzkrājuma ierakstus tādā detalizācijas pakāpē, kāds konkrētajā gadījumā nepieciešams.

Iepriekš minētos darbus var veikt pats Produkta īpašnieks vai arī uzticēt tos veikt Izstrādes komandai. Tomēr arī šādā gadījumā par tiem atbildīgs ir Produkta īpašnieks.

Produkta īpašnieks ir viens cilvēks, nevis personu grupa. Produkta īpašnieks Produkta darbu uzkrājumā var atspoguļot arī citu iesaistīto vēlmēs, bet, ja kāds vēlas mainīt Uzkrājumā iekļauto darbu prioritāti, tad viņam par to ir jāpārlicina Produkta īpašnieks.

Lai Produkta īpašnieka darbība būtu veiksmīga, viņa/viņas lēmumi ir jārespektē visai organizācijai. Produkta īpašnieka pieņemtie lēmumi ir redzami Produkta darbu uzkrājuma saturā un darbu secībā. Nevienam nav tiesību likt Izstrādes komandai veikt darbus no cita prasību saraksta, izņemot Produkta uzkrājumu, savukārt Izstrādes komandai nav ļauts darboties pēc kāda cita vēlmēm, izņemot Produkta īpašnieka.

Izstrādes komanda

Izstrādes komandu veido speciālisti, kas strādā, lai katra sprinta beigās izveidotu un potenciāli piegādātu produkta Papildinājumu, kurā iekļauti pabeigtie darbi. Produkta Papildinājumu veido tikai Izstrādes komandas locekļi.

Izstrādes komandas struktūrai un organizācijas atbalstam ir jābūt tādām, lai komanda pati varētu organizēt un pārvaldīt savu darbu. Rezultējošā sinerģija pilnveido Izstrādes komandas kopējo lietderību un efektivitāti.

Izstrādes komandu raksturo:

- Tā ir pašorganizēta, neviens (pat ne *Scrum* Komandas kapteinis) Izstrādes komandai nenorāda, kā Produkta darbu uzkrājumu pārvērst potenciāli piegādājamā funkcionalitātes papildinājumā;
- Izstrādes komanda ir daudzfunkcionāla, tai ir visas zināšanas, kas komandai nepieciešamas, lai izveidotu produkta Papildinājumu;
- *Scrum* neatzīst citus Izstrādes komandas locekļu amata nosaukumus kā vien Izstrādātājs, neatkarīgi no katra veicamā darba; nav paredzēti nekādi šī likuma izņēmumi;
- *Scrum* neatzīst īpašu apakšgrupu izdalīšanu Izstrādes komandā, neatkarīgi no noteiktiem darbu veidiem, kas jādara, piemēram, testēšanu vai biznesa analīzi; nav paredzēti nekādi šī likuma izņēmumi;
- Katram Izstrādes komandas loceklim var būt īpašas prasmes un fokusa zonas, bet atbildība ir visai Izstrādes komandai kopā.

Izstrādes komandas lielums

Optimāls izstrādes komandas lielums ir gana mazs, lai komanda joprojām būtu mobila, un gana liels, lai tā varētu paveikt nozīmīgu darbu. Ja Izstrādes komandā ir mazāk par trīs locekļiem, tajā ir mazāka mijiedarbība un līdz ar to arī mazāks produktivitātes ieguvums. Mazākā Izstrādes komandā var būt prasmju ierobežojums, kas savukārt var traucēt Izstrādes komandai sagatavot potenciāli piegādājamu produkta Papildinājumu. Komandā, kurā ir vairāk par deviņiem locekļiem, ir nepieciešama pārāk liela koordinēšana. Jo lielāka komanda, jo tā veido komplicētāku sistēmu un procesus, tādēļ lielas izstrādes komandas pārvaldīšana, balstoties uz empīriskiem faktiem, ir pārāk sarežģīta. Ja Produkta īpašnieks un *Scrum* Komandas kapteinis nerealizē Izstrādes darbus no projekta Darbu uzkrājuma, tad viņu lomas izstrādes komandas lieluma aprēķinā netiek iekļautas.

Scrum Komandas kapteinis

Scrum Komandas kapteinis ir atbildīgs, lai *Scrum* tiktu saprasts un iedzīvināts darbībā. *Scrum* Komandas kapteinis to nodrošina, pārlicinoties, ka *Scrum* Komanda seko *Scrum* teorijai, praksei un noteikumiem.

Scrum Komandas kapteinis ir *Scrum* Komandas padotais un līderis vienlaikus. *Scrum* Komandas kapteinis palīdz tiem, kas ir ārpus *Scrum* Komandas, saprast, kuras mijiedarbības ar *Scrum* komandu ir vajadzīgas un noderīgas, kuras – nē. *Scrum* Komandas kapteinis palīdz ikvienam mainīt šīs mijiedarbības, lai palielinātu *Scrum* komandas radītā produkta vērtību.

Scrum Komandas kapteiņa atbalsts Produkta īpašniekam

Scrum Komandas kapteinis palīdz Produkta īpašniekam dažādos veidos, tai skaitā:

- Atrodot tehnikas/metodes efektīvākai Produkta darbu uzkrājuma pārvaldīšanai;
- Palīdzot Izstrādes komandai saprast vajadzību pēc skaidra un kodolīga Produkta darbu uzkrājuma;
- Izprotot produkta plānošanu empīriskā vidē;
- Nodrošinot, ka Produkta īpašnieks zina, kā sakārtot Produkta darbu uzkrājumu, lai no tā gūtu maksimālu labumu;
- Izprotot un praktizējot spējās izstrādes iemaņas;
- Sekmējot *Scrum* norises (pēc pieprasījuma vai nepieciešamības).

Scrum Komandas kapteiņa atbalsts Izstrādes komandai

Scrum Komandas kapteinis palīdz Izstrādes komandai dažādos veidos:

- Trenējot Izstrādes komandu pašorganizēties un pilnveidot savu daudzfunkcionalitāti;
- Palīdzot Izstrādes komandai radīt augstas vērtības produktus;
- Novēršot šķēršļus Izstrādes komandas virzībai;
- Sekmējot *Scrum* norises (pēc pieprasījuma vai nepieciešamības);
- Apmācot Izstrādes komandu tādos organizatoriskos apstākļos, kuros *Scrum* vēl nav pilnībā ieviests un saprasts.

Scrum Komandas kapteiņa atbalsts Organizācijai

Scrum Komandas kapteinis palīdz Organizācijai dažādos veidos:

- Vadot un apmācot Organizāciju *Scrum* ieviešanā;
- Plānojot *Scrum* implementācijas organizācijas iekšienē;
- Palīdzot darbiniekiem un pārējām ieinteresētajām pusēm saprast un iedzīvināt *Scrum*, kā arī empīriskā balstītu produktu izstrādi;
- Ierosinot organizācijā izmaiņas, kas palielinātu *Scrum* komandas produktivitāti;
- Sadarbojoties ar citiem *Scrum* Komandas kapteiniem, vairojot organizācijā ieviestā *Scrum* efektivitāti.

Scrum notikumi

Scrum metodika paredz noteiktu pasākumu kopumu, lai nodrošinātu norišu regularitāti un samazinātu neplānotu sanāksmju nepieciešamību. *SCRUM* paredzētie pasākumi ir ar noteiktu maksimālo garumu. Tas nodrošina, ka plānošanas aktivitātēm ir paredzēts noteikts laiks, tā nepieļaujot lieku laika patēriņu plānošanas procesā.

Papildus Sprintam kā tādām, kā ietvaros tiek organizēti visi notikumi, katrs *Scrum* notikums ir formāla iespēja kaut ko pārbaudīt un pielāgot. Šie notikumi ir veidoti tā, lai nodrošinātu nepieciešamo caurskatāmību un pārbaudes. Procesā neiekļaujot kādu no *Scrum* pasākumiem, samazinās caurskatāmība, kā arī iespējas veikt pārbaudes un nepieciešamos pielāgojumus.

Sprints

Sprints ir *Scrum*'a ritmiski pukstošā sirds: tas ir mēnesi vai mazāk ilgs periods, kura laikā tiek sagatavots pabeigts, lietojams un potenciāli piegādājams produkta Papildinājums. Visā izstrādes laikā Sprinti ir nemainīga garuma. Jauns Sprint sākās uzreiz pēc iepriekšējā Sprinta pabeigšanas.

Sprintu veido un tas ietver Sprinta plānošanu, ikdienas sanāksšanas, izstrādes darbus, Sprinta apskati un Sprinta retrospekciju.

Sprinta laikā:

- Netiek veiktas nekādas izmaiņas, kas varētu ietekmēt Sprinta mērķi;
- Kvalitātes mērķi nesamazinās;
- Ja ir iegūta kāda papildus informācija, Produkta ģipša un Izstrādes komanda var pārskatīt un precizēt sprinta sfēru.

Katrs Sprint var tik uzskatīts par projektu, ne garāku par vienu mēnesi. Līdzīgi kā projekts, Sprinti ir paredzēti, lai tā laikā kaut ko noteiktu sasniegtu vai paveiktu. Katrā Sprintā ir noteikts, kas ir jāizstrādā, izstrādes pamatnostādņu projektējums un elastīgs plāns, pats izstrādes darbs, kā arī tā rezultātā radies produkts.

Sprinta garums ir ierobežots – ne ilgāk kā viens kalendārais mēnesis. Ja Sprints ir pārāk garš, var mainīties definējums tam, kas ir jāizstrādā, var pieaugt sarežģītība, kā arī palielināties riski. Sprintā paredzamība tiek nodrošināta, pārbaudot virzību uz mērķi un veicot atbilstošus pielāgojumus vismaz katru kalendāro mēnesi. Sprints arī limitē risku līdz viena kalendārā mēneša izmaksām.

Sprinta atcelšana

Sprintu var atcelt, līdz iestāties tā beigu datums. Tikai Produkta ģipšniekam ir tiesības atcelt Sprintu, tomēr viņš/viņa to var darīt arī citu ieinteresēto personu, Izstrādes komandas vai *Scrum* Komandas kapteiņa iespaidā.

Sprints var tikt atcelts, ja Sprinta mērķis kļūst neaktuāls, piemēram, ja uzņēmums maina virzienu vai ja mainās tirgus vai tehnoloģiskie nosacījumi. Kopumā, Sprints ir jāatceļ, ja tas esošajos apstākļos kļūst bezjēdzīgs. Tomēr, tā kā Sprints ir relatīvi īss, tā atcelšanai reti ir liela jēga.

Kad Sprints ir atcelts, tiek pārskatīti visi projekta uzkrājuma izdarītie darbi un darbi statusā „Pabeigts”. Ja daļa darbu ir potenciāli piegādājami, Produkta ģipšnieks parasti piegādi akceptē. Visi nerealizētie uzkrājuma darbi tiek atkārtoti novērtēti un iekļauti atpakaļ produkta darbu uzkrājumā. To realizācijā ieguldītais darbs ātri zaudē vērtību un tādēļ ir regulāri pārvērtējams.

Sprinta atcelšana patērē resursus, jo visiem ir jāpārgrupējas Sprinta plānošanai jaunā Sprinta uzsākšanai. Sprinta atcelšana ir ārkārtas pasākums, tā nereti ir traumatiska *Scrum* komandai.

Sprinta plānošana

Darbi, kas veicami konkrētajā Sprintā, tiek saplānoti Sprinta plānošanā. Plāns tiek veidots radošā visas *Scrum* komandas kopsadarbībā.

Sprinta plānošana ir laikā ierobežota – mēneša garai Sprinta plānošanai paredzētas ne vairāk kā astoņas stundas. Īsākiem sprintiem arī plānošana parasti ir īsāka. *Scrum* komandas kapteinis pārliecinās, ka plānošana notiek un ka visi dalībnieki saprot tās mērķi. *Scrum* komandas kapteinis māca izstrādes komandu to atcerēties visā plānotajā laikā.

Sprinta plānošana atbild uz šādiem jautājumiem:

- Kas Sprinta rezultātā varētu tikt iekļauts Papildinājumā?
- Kā tiks realizēts plānotais Papildinājums?

Pirmais jautājums: Ko šajā sprintā var izdarīt?

Izstrādes komanda plāno, kāda funkcionalitāte tiks izstrādātā Sprinta laikā. Produkta ģipšnieks ar pārējo *Scrum* komandu pārrunā un vienojas par mērķi, kas jāsasniedz Sprinta laikā, kā arī par tiem produkta darbu uzkrājuma darbiem, kuru realizēšana palīdzētu sasniegt Sprinta mērķi. Visa *Scrum* komanda sadarbojas, lai saprastu Sprinta uzdevumus.

Informācija, uz ko balstīties plānošanas laikā, ir produkta darbu uzkrājums, pēdējais produkta Papildinājums, prognozējamā Izstrādes komandas jauda Sprinta laikā, kā arī Izstrādes komandas

veiktspēja pēdējā Sprinta laikā. Cik darbus no produkta darbu uzkrājums iekļaut Sprintā, ir atkarīgs tikai un vienīgi no Izstrādes komandas. Tikai Izstrādes komanda var noteikt, kas ir paveicams gaidāmajā Sprintā.

Kad Izstrādes komanda ir aplēsusī, kurus uzkrājuma darbus tā realizēs Sprinta laikā, *Scrum* komanda noformulē Sprinta mērķi. Sprinta mērķis ir mērķis, kas tiks sasniegts Sprintā, realizējot izvēlētos darbus no produkta darbu uzkrājuma, un tas visā Sprinta laikā sniedz norādes Izstrādes komandai, kādēļ Papildinājums tiek veidots.

Otrā daļa: Kā izvēlētais darbs tiks paveikts?

Kad Sprinta mērķi ir uzstādīti un Sprintā veicamie darbi atlasīti, Izstrādes komanda izlemj, kādā veidā Sprinta laikā izvēlēta funkcionalitāte tiks pārvērsta pabeigtā programmatūras Papildinājumā. Izvēlētie Projekta uzkrājuma darbi, kā arī to piegādes plāns kopā veido Sprinta uzkrājumu.

Izstrādes komanda parasti sāk ar sistēmas un darbu, kas nepieciešams, lai pārvērstu Produkta uzkrājuma ierakstus strādājošā programmatūras Papildinājumā, uzmetuma izveidošanu. Darbi var būt dažāda lieluma un sarežģītības. Tomēr Sprinta plānošanā tiek saplānots tāds kopējais darba daudzums, kādu Izstrādes komanda var paveikt Sprinta laikā. Darbi, kas Sprintā tiek plānoti kā pirmie, plānošanas sanāksmes beigās tiek sadalīti mazākos darbos, paveicamos vienā dienā vai īsākā laikā. Izstrādes komanda ir pašorganizējoša, tādēļ gan plānošanā, gan arī Sprinta laikā pati izvēlas un uzsāk darbus no Sprinta uzkrājuma.

Produkta īpašnieks var palīdzēt precizēt izvēlētos Produkta darbu uzkrājuma darbus, kā arī atrast nepieciešamos kompromisus. Ja Izstrādes komanda konstatē, ka tai ir pārāk maz vai pārāk daudz darba, tā kopā ar Produkta īpašnieku var pārskatīt Sprinta uzkrājumu. Izstrādes komanda var uzaicināt plānošanā piedalīties arī citus cilvēkus, lai saņemtu padomus par sfēru vai izmantotajām tehnoloģijām.

Sprinta plānošanas beigās Izstrādes komandai jāspēj Produkta īpašniekam un *SCRUM* Komandas kapteinim izskaidrot, kādā veidā tā kā pašorganizējoša komanda ir plānojusi sasniegt Sprinta mērķus un izveidot paredzēto programmatūras Papildinājumu.

Sprinta mērķis

Sprinta mērķis nosaka, kas Izstrādes komandai jāsasniedz Sprinta laikā, realizējot izvēlētos darbus. Tas Izstrādes komandai atgādina, kāpēc programmatūras Papildinājums tiek izstrādāts. Sprinta mērķi rada Sprinta plānošanas laikā. Tas Sprinta ietvaros izstrādes komandai dod zināmu elastīgumu pret izstrādājamo funkcionalitāti. Izvēlēto darbu realizācija kopā nodrošina vienu lielāku funkciju vai procesu, kas arī var tik noteikts par Sprinta mērķi. Sprinta mērķis var būt jebkura cita sakarība kas liek Izstrādes komandai vairāk strādāt kopā, nevis balstoties uz atsevišķām iniciatīvām.

Strādājot Izstrādes komanda vienmēr patur prātā Sprinta mērķi, jo tā izstrādā funkcionalitāti un ievieš tehnoloģijas tieši Sprinta mērķa sasniegšanai. Ja darbs izrādās citāds, nekā Izstrādes komanda gaidījusi, tad tā, sadarbojoties ar Produkta īpašnieku, Sprinta laikā pārskatīja Sprinta darbu saraksta sfēru.

Ikdienas sanāksšana

Ikdienas sanāksšana ir ne vairāk kā 15 minūšu gara Izstrādes komandas tikšanās, lai sinhronizētu komandas dalībnieku aktivitātes un radītu plānu turpmākajām 24 stundām. To panāk, pārskatot kopš iepriekšējās ikdienas sanāksšanas paveikto, kā arī prognozējot to, kas būtu jāizdara līdz nākamajai sanāksšanai.

Vienkāršības dēļ ikdienas sanāksšanas vienmēr notiek tajā pašā laikā un vietā. Tās laikā katrs Izstrādes komandas dalībnieks izstāsta:

- Ko es izdarīju vakar, kas palīdzētu Izstrādes komandai sasniegt Sprinta mērķi?
- Ko es darīšu šodien, lai palīdzētu Izstrādes komandai sasniegt Sprinta mērķi?
- Vai es redzu kādus šķēršļus, kas kavē mani vai Izstrādes komandu Sprinta mērķa sasniegšanā?

Izstrādes komanda izmanto Ikdienas sanāksšanas, lai novērtētu virzību Sprinta mērķa sasniegšanai un Sprinta uzkrājuma realizēšanai. Ikdienas sanāksšanas veicina iespējamību, ka Izstrādes komanda Sprinta mērķi sasniegs. Katru dienu Izstrādes komandai jāizprot, kā tā domā strādāt kopā kā pašorganizēta komanda, lai sasniegtu Sprinta mērķi un Sprinta beigās radītu iecerēto programmatūras Papildinājumu. Izstrādes komanda bieži uzreiz pēc ikdienas sanāksšanas sapulcējas, lai sīkāk izdiskutētu, adaptētu vai pārplānotu atlikušos Sprinta darbus.

Scrum Komandas kapteinis nodrošina, ka Izstrādes komandā notiek ikdienas sanāksšanas, bet Izstrādes komanda savukārt ir atbildīga par šo sanāksmju vadīšanu. *Scrum* Komandas kapteinis māca Izstrādes komandai, kā nepārsniegt plānoto sanāksšanas laiku – 15 minūtes.

Scrum Komandas kapteinis īsteno likuma ievērošanu, ka tikai Izstrādes komandas locekļi piedalās ikdienas sanāksšanās.

Ikdienas sanāksšanas uzlabo komunikāciju komandā, samazina citu tikšanos nepieciešamību, identificē un novērš traucēkļus izstrādē, uzsver un veicina ātru lēmumu pieņemšanu, kā arī uzlabo visas izstrādes komandas zināšanas par projektu kopumā. Ikdienas sanāksšana ir svarīgākā pārbaudes un pielāgojumu veikšanas sanāksme projektā.

Sprinta apskate

Sprinta apskate notiek Sprinta beigās, lai pārbaudītu sprintā radīto programmatūras Papildinājumu un atjaunotu Produkta darbu uzkrājumu, ja tas ir nepieciešams. Sprinta apskates laikā *Scrum* komanda un citas iesaistītās personas vienojas par to, kas ir izdarīts Sprinta laikā. Balstoties uz to, kā arī citām izmaiņām, kas Sprinta laikā veiktas Produkta uzkrājumā, dalībnieki vienojas par to, kas darāms tālāk, lai iegūtu labāko iespējamo rezultātu. Sprinta apskate ir neformāla sanāksme, nevis tikšanās par darbu aktuālo statusu, bet produkta Papildinājuma demonstrēšana ir domāta, lai iegūtu atgriezenisko saiti un atsauksmes, kā arī veicinātu tālāku sadarbību.

Mēnesi ilga Sprinta apskate ir ne garāka par 4 stundām; Ja Sprints ir īsāks par mēnesi, arī apskatei paredzētais laiks parasti ir īsāks. *Scrum* Komandas kapteinis nodrošina, ka notiek Sprinta apskates un ka

klātesošie saprot tās nolūku. *Scrum* Komandas kapteinis māca visus iesaistītos, kā nepārsniegt plānoto tikšanās laiku.

Sprinta apskates iekļauj šādus elementus:

- Apskatēs piedalās *Scrum* komanda, kā arī pārējās ieinteresētās personas, ko uzaicinājis Produkta īpašnieks;
- Produkta īpašnieks identificē, kuri darbi ir un kuri nav „Pabeigti”;
- Izstrādes komanda izrunā, kas tai Sprinta laikā veicies labi, ar kādām problēmām tā saskārusies un kā tās tika risinātas;
- Izstrādes komanda demonstrē darbus, kas ir „Pabeigti”, un atbild uz jautājumiem par sagatavoto programmatūras Papildinājumu;
- Produkta īpašnieks runā par Produkta darbu uzkrājuma pašreizējo stāvokli, kā arī izvirza vēlamos gala datumus, balstoties uz līdzšinējo darbu progresu (ja tas nepieciešams);
- Visi sanākušie sadarbojas, lai vienotos, ko darīt tālāk, tādējādi Sprinta apskate ir nozīmīgs ieguldījums nākamā Sprinta plānošanai;
- Pārskats, kā tirgus vai potenciālā produkta izmantošana varētu mainīties, kas ir vērtīgākais, ko varētu darīt nākamā;
- Pārskats par laika grafiku, budžetu, potenciālajām iespējām un tirgu nākamajam paredzamajam Produkta papildinājumam.

Sprinta apskates rezultātā ir pārskatīts Produkta darbu uzkrājums, kas nosaka iespējamās nākamā Sprinta darbus. Produkta darbu uzkrājums var būt mainīts pilnībā, atbilstoši aktuālajām iespējām un vajadzībām.

Sprinta retrospekcija

Sprinta retrospekcija ir *Scrum* komandas iespēja pārbaudīt pašai sevi un izveidot nākamajos Sprintos ieviešamo uzlabojumu plānu.

Sprinta retrospekcija notiek pēc Sprinta apskates, bet pirms nākamā Sprinta plānošanas. Mēnesi garam sprintam retrospekcijas ilgums ir trīs stundas, īsākiem sprintiem tā parasti ir īsāka. *Scrum* Komandas kapteinis nodrošina, ka Sprinta retrospekcijas notiek un ka klātesošie saprot tās nolūku. *Scrum* Komandas kapteinis māca visus iesaistītos, kā nepārsniegt plānoto tikšanās laiku. Retrospekcijā *Scrum* Komandas kapteinis piedalās kā vienlīdzīgs komandas loceklis attiecībā pret atbildību par *Scrum* procesiem.

Sprinta retrospekcijas mērķi:

- Pārskatīt, kāda pagājušā Sprintā bija sadarbība ar cilvēkiem, savstarpējās attiecības, procesu norisi un rīku izmantošana;
- Identificēt un sakārtot svarīgākās lietas, kas veicās labi, kā arī potenciālos uzlabojumus;
- Atbilstoši *Scrum* komandas darba stilam izveidot uzlabojumu ieviešanas plānu.

Scrum Komandas kapteinis iedrošina *Scrum* komandu *Scrum* procesu ietvaros uzlabot tās izstrādes procesu un izmantotās prakses, lai nākamā sprinta laikā tie būtu vēl efektīvāki un patīkamāki. Katrā Sprinta retrospekcijā *Scrum* komanda domā veidus, kā uzlabot produkta kvalitāti, papildinot un piemērojot "Pabeigts" definīciju, lai tā būtu piemērota šai komandai.

Sprinta retrospekcijas beigās *Scrum* komandai ir jābūt identificētiem uzlabojumiem, kas ieviešami nākamajā Sprintā. To ieviešana nākamajā Sprintā faktiski ir *Scrum* komandas pašpielāgošanās atbilstoši veiktajai pašpārbaudei. Lai arī uzlabojumus var ieviest jebkurā brīdī, Sprinta retrospekcija piedāvā formālu iespēju fokusēties uz pārbaudes un pielāgojumu veikšanu.

Scrum artefakti

Scrum artefakti reprezentē darbu vai citas vērtības, lai nodrošinātu procesu caurskatāmību, kā arī veicinātu pārbaudi un nepieciešamo pielāgojumu veikšanu. *Scrum* artefakti ir speciāli radīti tādi, lai uzlabotu svarīgākās informācijas caurskatāmību un lai visi tos saprastu vienādi.

Produkta darbu uzkrājums

Produkta darbu uzkrājums ir sakārtots saraksts, kurā iekļauts viss, kas varētu būt nepieciešams produkta radīšanai, un ir vienīgais prasību avots jebkuras produkta izmaiņas vai papildinājuma veikšanai. Par Produkta darbu uzkrājumu, ieskaitot tā saturu, pieejamību un sakārtojumu, ir atbildīgs Produkta īpašnieks.

Produkta darbu uzkrājums nekad nav pilnībā pabeigts un gatavs. Pirmās versijas aptver tikai sākotnēji zināmās un labāk saprotamās prasības. Produkta darbu uzkrājums attīstās un izvēršas vienlaikus ar pašu produktu un apstākļiem, kādos tas tiks lietots. Produkta darbu uzkrājums ir dinamisks, tas nepārtraukti mainās, identificējot, kas produktam nepieciešams, lai tas būtu piemērots, konkurētspējīgs un noderīgs. Produkta darbu uzkrājums eksistē tik ilgi, cik ilgi eksistē pats Produkts.

Produkta darbu uzkrājums iekļauj visas produkta iespējas, funkcijas, prasības, uzlabojumus un labojumus, un tas nosaka nākamajās produkta piegādēs veicamās izmaiņas. Produkta darbu uzkrājumā par katru ierakstu ir šāda informācija: apraksts, secība sarakstā, novērtējums un vērtība .

Tā kā procesa laikā produkts tiek lietots un pieaug tā apjoms, kā arī par to tiek saņemts tirgus novērtējums, Produkta darbu uzkrājums kļūst arvien lielāks, apjomīgāks un izsmeļošāks. Prasības ir nepārtraukti mainīgas, tādēļ Produkta darbu uzkrājums ir dzīvs projekta artefakts. Biznesa prasību, tirgus stāvokļa, tehnoloģiju vai citu apstākļu izmaiņas vai radīt izmaiņas Produkta darbu uzkrājumā.

Nereti viena produkta izstrādē piedalās vairākas *Scrum* komandas. Lai aprakstītu nākamās projekta darbus, tiek izmantots viens Produkta darbu uzkrājums, tam pievienojot papildus saraksta atribūtus darbu grupēšanai.

Produkta darbu uzkrājuma pilnveidošana ietver darbu detalizāciju, sakārtošanu un prioritizēšanu, kā arī novērtējuma pievienošanu - tas ir nepārtraukts process, kurā darbu detalizācijā sadarbojas Produkta īpašnieks un Izstrādes komanda. Produkta darbu uzkrājuma pilnveidošanas laikā darbi tiek pārskatīti un

mainīti. Scrum komanda izvēlas, kad un kā ir veicama Produkta darbu uzkrājuma pilnveidošana. Tā jebkurā gadījumā parasti nepatērē vairāk kā 10% no Izstrādes komandas kopējās kapacitātes. Neatkarīgi no tā, darbus jebkurā laikā var atjaunināt un papildināt Produkta īpašnieks vai kāds cits pēc Produkta īpašnieka ieskata.

Produkta darbu uzkrājumā augšgalā sakārtotie darbi ir skaidrāki un detalizētāki nekā saraksta lejasdaļā esošie. Precīzāks novērtējums tiek veikts, pamatojoties uz lielāku skaidrību un detalizāciju; jo zemāks sakārtojums, jo mazāk detaļu. Tie Produkta uzkrājuma darbi, pie kuriem Izstrādes komanda strādās nākamajā Sprintā, ir sadalīti mazākos darbos, lai katru no tiem varētu saprātīgi pabeigt Sprinta laikā. Šādi sadalītie darbi Produkta uzkrājumā tiek marķēti kā „sagatavots” vai „gatavs izstrādei” un tos ir iespējams izvēlēties Sprinta plānošanas laikā. Produkta uzkrājuma ieraksti parasti iegūst šādu caurskatāmības pakāpi iepriekš aprakstītās pilnveidošanas rezultātā.

Izstrādes komanda ir atbildīga par visu darbu novērtējumiem. Produkta īpašnieks var ietekmēt Izstrādes komandas domas, palīdzot izprast neskaidro un atrodot kompromisus, bet gala novērtējumu nosaka tikai tie cilvēki, kas veiks konkrētos darbus.

Virzības uz mērķi uzraudzība

Jebkurā laika brīdī ir jābūt iespējai saskaitīt kopējā darba apjomu, kas vēl izdarāms mērķa sasniegšanai.

Produkta īpašnieks nosaka kopējo atlikušo darbu vismaz uz katru Sprinta apskati. Produkta īpašnieks salīdzina šo vērtību ar atlikušā darba kopējo apjomu uz iepriekšējā Sprinta apskates brīdi, lai noteiktu padarītā darba progresu attiecībā pret vēlamo laiku mērķa sasniegšanai. Šī informācija ir caurskatāma visām ieinteresētajām pusēm.

Nākotnes virzības ātruma noteikšanai tiek lietotas dažādas projektīvās metodes, piemēram, tendenču grafiki, līknes un citas projektīvās metodes; tās ir pierādījušas savu lietderīgumu. Tomēr tās neaizstāj praktiskās pieredzes nozīmi. Sarežģītos apstākļos nav zināms, kas notiks, tikai tas, kas noticis, ir izmantojums tālredzīgu nākotnes lēmumu pieņemšanai.

Sprinta uzkrājums

Sprinta uzkrājums ir no Produkta uzkrājuma atlasītu darbu komplekts konkrētajam Sprintam kopā ar izveidojamā produkta Papildinājuma piegādes plānu un realizējamā Sprinta mērķi. Sprinta uzkrājums ir Izstrādes komandas prognoze, kāda funkcionalitāte tiks iekļauta nākamajā produkta Papildinājumā un kāds darbs nepieciešams, lai šo funkcionalitāti piegādātu.

Sprinta uzkrājums parāda visus tos veicamos darbus, ko Izstrādes komanda atzinusi par nepieciešamiem, lai sasniegtu Sprinta mērķi.

Sprinta darbu saraksts ir pietiekami detalizēts plāns, kura izmaiņu gaitu var saprast lkdienas sanāksmās. Izstrādes komanda korigē Sprinta uzkrājumu visu Sprinta laiku, un Sprinta uzkrājums Sprinta laikā atjaunojas. Tas notiek, Izstrādes komandai strādājot kā plānots un arvien vairāk uzzinot par darbu, kādu nepieciešams veikt, lai sasniegtu Sprinta mērķi.

Ja Sprinta mērķa sasniegšanai ir nepieciešams kāds papildus darbs, Izstrādes komanda to pievieno Sprinta darbu uzkrājumam. Līdzko darbs ir izdarīts vai pabeigts, tā tiek pārrēķināts atlikušais nepieciešamais darbs. Ja kāds plāna elements ir iekļauts uzkrājumā lieki, to izņem. Tikai Izstrādes komanda Sprinta laikā var mainīt savu Sprinta uzkrājumu. Sprinta uzkrājums ir labi pamanāma reālā laika ilustrācija Izstrādes komandas plānotajam Sprinta laikā paveicamajam darbam, un tā pieder tikai un vienīgi Izstrādes komandai.

Sprinta progresa mērījumi

Jebkurā Sprinta brīdī jāspēj aprēķināt kopējais Sprinta uzkrājuma atlikušais darbs.

Izstrādes komanda seko līdzī kopējā atlikušā darba apjomam vismaz katrā lkdienas sanāksmā un atbilstoši novērtē iespējamību sasniegt Sprinta mērķi. Visa Sprinta laikā sekojot, cik darba atlicis, Izstrādes komanda var vadīt Sprinta progresu.

Programmatūras Papildinājums

Programmatūras Papildinājums ir visu šī Sprinta laikā pabeigto Produkta uzkrājuma darbu un visu iepriekšējo Papildinājuma vērtību summa. Katra Sprinta beigās ir jāizveido jauns „Pabeigts” programmatūras Papildinājums, kas nozīmē, ka tam jābūt lietojamam (izmantojamam) un jāatbilst *Scrum* komandas definētajam pabeigtības definīcijai. Tam jābūt lietojamam neatkarīgi no tā, vai Produkta īpašnieks izvēlas vai neizvēlas to piegādāt.

Artefaktu caurskatāmība

Scrum paļaujas uz caurskatāmību. Lēmumi par rezultāta uzlabošanu un risku kontroli tiek pieņemti, balstoties uz esošo artefaktu stāvokli. Tādējādi gadījumā, ja caurskatāmība ir pilnīga, šiem lēmumiem ir stabils pamats. Ja artefakti nav pilnībā pārskatāmi un caurredzami, šie lēmumi var būt kļūdaini, vērtība var samazināties un riski – pieaugt.

Scrum komandas kapteinim jāstrādā kopā ar Produkta īpašnieku, Izstrādes komandu un citām iesaistītajām pusēm, lai saprastu, vai artefakti ir pilnībā skaidri un caurskatāmi. Eksistē dažādas prakses, kā novērst artefaktu nepārskatāmību, Scrum komandas kapteiņa uzdevums ir palīdzēt ikvienam atrast un izmantot piemērotāko praksi, ja pārskatāmība nav pilnīga. Scrum komandas kapteinis var atklāt nepietiekamu pārskatāmību, pārbaudot artefaktus, saprotot modeļus, uzmanīgi klausoties, kas tiek pateikts, kā arī uztverot atšķirības starp gaidīto un faktisko rezultātu.

Scrum komandas kapteiņa darbs ir strādāt kopā ar Scrum komandu un uzņēmumu, lai palielinātu artefaktu pārskatāmību. Šis darbs parasti ietver mācīšanos, pārliecināšanu un izmaiņas. Caurskatāmība neiestājas vienas nakts laikā, tas ir ceļš.

Pabeigtības definīcija jeb Ko nozīmē „Pabeigts” darbs

Kad Produkta darbu uzkrājuma vienība vai programmatūras Papildinājums ir noteikts kā „Pabeigts”, katram ir jāsaprot, ko nozīmē „Pabeigts”. Lai arī katrā *Scrum* komandā izpratne ir atšķirīga, visiem vienas

komandas locekļiem ir jābūt vienotai izpratnei, ko nozīmē pabeigts darbs, tādējādi nodrošinot caurskatāmības principu. Šī vienotā izpratne arī ir *Scrum* komandas „Pabeigtības definīcija” un tiek lietota, lai noteiktu, kad darbs ir gatavs iekļaušanai produkta Papildinājumā.

Šī pati definīcija palīdz Izstrādes komandai noteikt, cik daudz Produkta uzkrājuma darbu tā var izvēlēties Sprinta plānošanas laikā. Katra Sprinta mērķis ir sagatavot tādu potenciāli piegādājamās funkcionalitātes Papildinājumu, kas atbilst *Scrum* komandas aktuālajai pabeigtības definīcijai.

Izstrādes komanda sagatavo produkta funkcionalitātes Papildinājumu katrā Sprintā. Programmatūras papildinājums ir gatavs lietošanai, tādējādi Produkta ģpašnieks var izlemt to nekavējoties piegādāt. Ja Programmatūras papildinājuma „Pabeigts” definīcija ir izstrādes organizācijas vienošanās, standarta vai vadlīniju sastāvdaļa, tad visām *Scrum* komandām tā jāievēro vismaz kā minimālās prasības savas komandas „Pabeigts” definīšanā. Ja Papildinājuma „Pabeigts” definīcija **nav** ietverta izstrādes organizācijas konvencijās, tad *Scrum* komandas Izstrādes komandai jārada „Pabeigts” definīcija, kas vislabāk atbilst produktam. Ja pie produkta vai sistēmas laidiena strādā vairākas *Scrum* komandas, tad „pabeigts” definīcija ir jāveido kopā visu *Scrum* komandu Izstrādes komandām.

Katrs Papildinājums ir papildinošs visiem iepriekšējiem Papildinājumiem (t.i. katrs Papildinājums satur arī visus iepriekšējos Papildinājumus) un ir pilnībā testēts, pārliecinoties, ka visi Papildinājumi kopā strādā.

Scrum komandai nobriestot, ir sagaidāms, ka tās pabeigtības definīcija paplašināsies, iekļaujot stingrākus nosacījumus kvalitātes uzlabošanai. Katram produktam vai sistēmai ir jābūt definētam, ko nozīmē katrs „Pabeigts” darbs pie šī produkta vai sistēmas.

Beigu piezīme

Scrum ir pieejams bez maksas un tiek piedāvāts šajā ceļvedī. *Scrum* lomas, artefakti, notikumi un noteikumi ir nemainīgi un, lai arī ir iespējams ieviest tikai daļu no *Scrum*, iegūtais rezultāts nebūs *Scrum*. *Scrum* eksistē tikai tā pilnīgumā un labi funkcionē kā bāze citām tehnikām, metodoloģijām un praksēm.

Pateicība

Cilvēkiem

No tiem tūkstošiem cilvēku, kas ir veicinājuši *Scrum* attīstību, mums jāizceļ tie, kuri bija visaktīvākie pirmajos desmit gados. Pirmais bija Džefs Sazerlands (Jeff Sutherland), kurš strādāja ar Džefu Makkenu (Jeff McKenna), un Kens Švābers (Ken Schwaber), strādājot ar Maiku Smitu (Mike Smith) un Krisu Mārtinu (Chris Martin). Nākamajos gados *Scrum* tālāk lietoja un attīstīja daudzi citi, un bez viņu palīdzības *Scrum* nebūtu tik rafinēts, kāds tas ir šodien.

Vēsture

Kens Švābers un Džefs Sazerlands plašāku publiku ar *Scrum* pirmo reizi iepazīstināja 1995.gadā OOPSLA konferencē. Viņu prezentācija dokumentēja būtiskāko, ko Kens un Džefs bija iemācījušies *Scrum* lietošanas pirmajos gados.

Scrum vēsture nu jau ir gana gara. Lai godinātu pirmo vietu, kur *Scrum* tika lietots un pilnveidots, mēs pieminam *Individual, Inc.*, *Fidelity Investments*, un *IDX* (tagad: *GE Medical*).

Scrum ceļvedis dokumentē Džefa Sazerlanda un Kena Švābera *Scrum* meklējumus un darbību vairāk kā divdesmit gadu garumā. Citi avoti jūs iepazīstinās ar modeļiem, procesus un atziņas, kas papildina *Scrum* ietvaru. Tas uzlabo produktivitāti, vērtību, radošumu un lepnumu.